Phase 0 — Project Skeleton & File Structure  
  
Goal: Create the clean starting point.  
  
Root directory: Portfolio/  
  
Sub-folders:  
  
Portfolio/  
├── frontend/ # React + Vite + Tailwind app  
├── backend/ # Cloudflare Workers, API routes, DB  
├── docs/ # Phase logs, project notes  
├── wrangler.toml # Cloudflare config  
├── package.json # Root scripts  
└── README.md # Project overview  
  
  
Initialize Git + GitHub repo (proof of version control).  
  
Initialize monorepo (can be Yarn workspaces or PNPM, but plain npm with separate package.jsons works too).  
  
🎨 Phase 1 — Frontend Bootstrap  
  
Goal: Stand up the Netflix-style frontend.  
  
Inside frontend/:  
  
npm create vite@latest frontend -- --template react  
cd frontend  
npm install -D tailwindcss postcss autoprefixer  
npx tailwindcss init -p  
  
  
Configure Tailwind:  
  
// tailwind.config.js  
export default {  
 content: ["./index.html", "./src/\*\*/\*.{js,jsx,ts,tsx}"],  
 theme: { extend: {} },  
 plugins: [],  
}  
  
  
Base components scaffold:  
  
frontend/src/components/  
├── NavBar.jsx # horizontal semi-translucent bar  
├── SideBar.jsx # collapsible vertical nav  
├── Carousel.jsx # rotating hero cards  
├── ProjectCard.jsx # reusable Netflix-like tiles  
└── Layout.jsx # wrapper with nav + sidebar logic  
  
  
Set up routing:  
  
Use react-router-dom.  
  
Pages: Home.jsx, Project.jsx (individual project detail w/ live app + code).  
  
🖼 Phase 2 — Netflix-like UI Layout  
  
Goal: Nail the skeleton visual language.  
  
Horizontal NavBar:  
  
Semi-transparent (bg-black/60, sticky).  
  
Priority on scroll.  
  
Vertical SideBar:  
  
Hidden by default.  
  
Slide-in on hamburger button click (use framer-motion for smooth animation).  
  
Hero Carousel (below navbar):  
  
Cards swipeable left/right (desktop arrow buttons + mobile touch support).  
  
Each card = intro/about/skills/goals.  
  
Project Feed:  
  
Grid or horizontal row sections (like Netflix).  
  
Each card links to its own Project page.  
  
⚙️ Phase 3 — Backend Skeleton  
  
Goal: Prep Cloudflare for dynamic content.  
  
Inside backend/:  
  
backend/  
├── functions/  
│ ├── api/  
│ │ └── v1/  
│ │ ├── projects.ts # GET list of projects  
│ │ └── project.ts # GET single project by id/slug  
└── migrations/  
 └── 001\_init.sql  
  
  
wrangler.toml binds:  
  
[[d1\_databases]]  
binding = "DB"  
database\_name = "portfolio-db"  
database\_id = "auto-generated-id"  
  
  
Migration example (skeleton only):  
  
CREATE TABLE projects (  
 id INTEGER PRIMARY KEY AUTOINCREMENT,  
 title TEXT NOT NULL,  
 slug TEXT UNIQUE NOT NULL,  
 description TEXT,  
 demo\_url TEXT, -- where live app is hosted  
 github\_url TEXT, -- link to GitHub repo  
 created\_at DATETIME DEFAULT CURRENT\_TIMESTAMP  
);  
  
  
(no seed data yet, but structure is ready)  
  
🔌 Phase 4 — API Integration  
  
Goal: Wire frontend to backend.  
  
Create src/services/api.js:  
  
export async function fetchProjects() {  
 const res = await fetch("/api/v1/projects");  
 return res.json();  
}  
  
export async function fetchProject(slug) {  
 const res = await fetch(`/api/v1/project?slug=${slug}`);  
 return res.json();  
}  
  
  
Home.jsx: calls fetchProjects() and renders ProjectCard list.  
  
Project.jsx: calls fetchProject(slug), displays embedded demo (<iframe> or <video> preview), GitHub link, and description.  
  
📱 Phase 5 — Responsiveness & UX Polish  
  
Goal: Make it feel modern and fluid.  
  
Tailwind breakpoints (sm, md, lg) for feed and cards.  
  
Swipe support for carousel (use react-swipeable).  
  
Smooth transitions (Framer Motion).  
  
Ensure horizontal nav always wins over sidebar (z-index priority).  
  
🔒 Phase 6 — Deployment Pipeline  
  
Goal: Deploy live on Cloudflare Pages + Workers.  
  
frontend/ → Cloudflare Pages.  
  
backend/ → Cloudflare Workers (via Wrangler).  
  
Configure routing so /api/\* requests route to Workers.  
  
Auto-deploy with GitHub Actions (push → Cloudflare).  
  
📝 Phase 7 — Content Management  
  
Goal: Make adding projects simple.  
  
Option A (short term): manually add rows via wrangler d1 execute.  
  
Option B (later): build a tiny /admin page protected by secret key for adding/updating projects.  
  
🌟 Phase 8 — Portfolio Enhancements  
  
Ideas once skeleton is live:  
  
Add filters (by tech: React, Node, Cloudflare).  
  
Add tags/skills badges on each project card.  
  
Add blog posts/articles feed.  
  
Animate hero cards like Netflix trailers.  
  
⚡ Summary:  
  
Phase 0–2 = frontend Netflix-style UI skeleton.  
  
Phase 3–4 = backend DB + API, no seed data yet.  
  
Phase 5 = responsive polish.  
  
Phase 6 = deploy on Cloudflare.  
  
Phase 7–8 = enhancements + CMS.