

Building a Web Development Portfolio



- A curated, deployed site showing who you are and what you've built



- The single most important asset for aspiring developers in 2026



- Hiring shifted: evidence of problem-solving and trajectory, not just skill



- Today: positioning, project selection, case studies, building, outreach



The 2026 Portfolio Standard



- Recruiters scan portfolios in under a minute, judging live demos and code activity



- Key signals: learning trajectory, finishing projects, and explaining decisions



- Red flags: tutorial-only projects, broken links, and inflated experience



Positioning Yourself for a Niche



- 'Generic coder' fails in 2026: indistinguishable portfolios blend together



- Pick a niche: accessibility, e-commerce, or AI-integrated products



- One-sentence positioning: your stack + the problems you solve



- Tailor every project and case study to your target industry

GENERIC CODER

Blends in. Easily replaceable.



VS.

NICHE DEVELOPER

Distinct. Valuable. In demand.



ACCESSIBILITY



E-COMMERCE



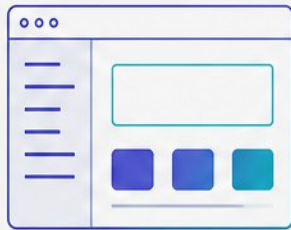
AI-INTEGRATED
PRODUCTS



MODERN STACK

Choosing Projects That Prove Ability

- - 3–5 high-quality, problem-solving projects beat 10 tutorial clones
- - Show range: a full-stack CRUD app, an AI feature, a personal tool
- - Build for a problem you actually face—real decisions, real outcomes
- - Frame capstones and redesigns as original work: decisions, not demos
- - Every featured project must have a live demo and clean README



Full-Stack CRUD App

End-to-end app with complete data flows

`</>`
LIVE DEMO


CLEAN README



AI Feature

Intelligent feature that solves a real need

`</>`
LIVE DEMO


CLEAN README



Personal Tool

Utility built to simplify a daily problem

`</>`
LIVE DEMO


CLEAN README

**AVOID
TUTORIAL
CLONES**



Structuring a Compelling Case Study



- Core narrative: problem, constraints, approach, tradeoffs, outcome, and reflection



- Show technical depth to hiring managers while staying clear for non-technical reviewers



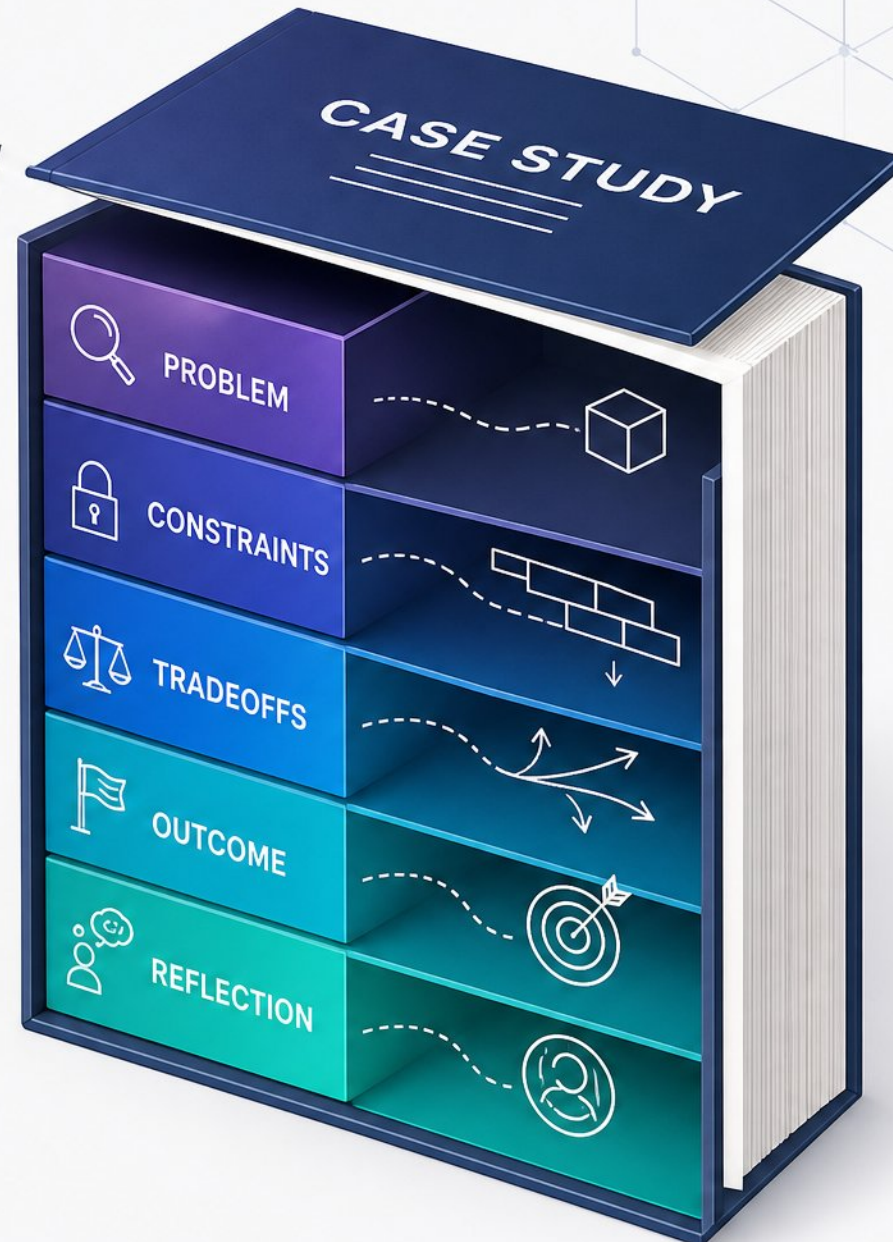
- Document failures and tradeoffs; signal senior-level thinking



- Explain why choices were made; not just what was built



- Frame bootcamp capstones and client work as original, decision-driven projects



Designing for the 55-Second Scan



- Simple, fast, instantly legible design beats creative flair



- Lead reviewers to your best 2-3 projects first



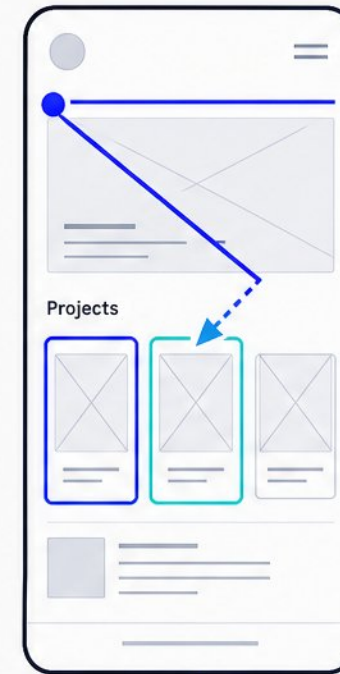
- Test mobile, accessibility, and keyboard navigation



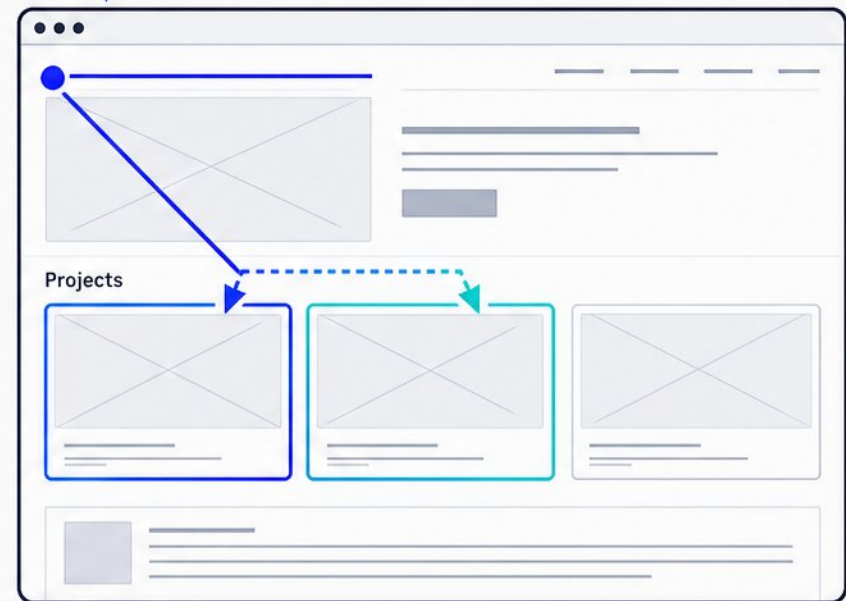
- Aim for Lighthouse 90+; LCP under 2.5s, zero CLS



- Target 100 on Accessibility; test with axe DevTools

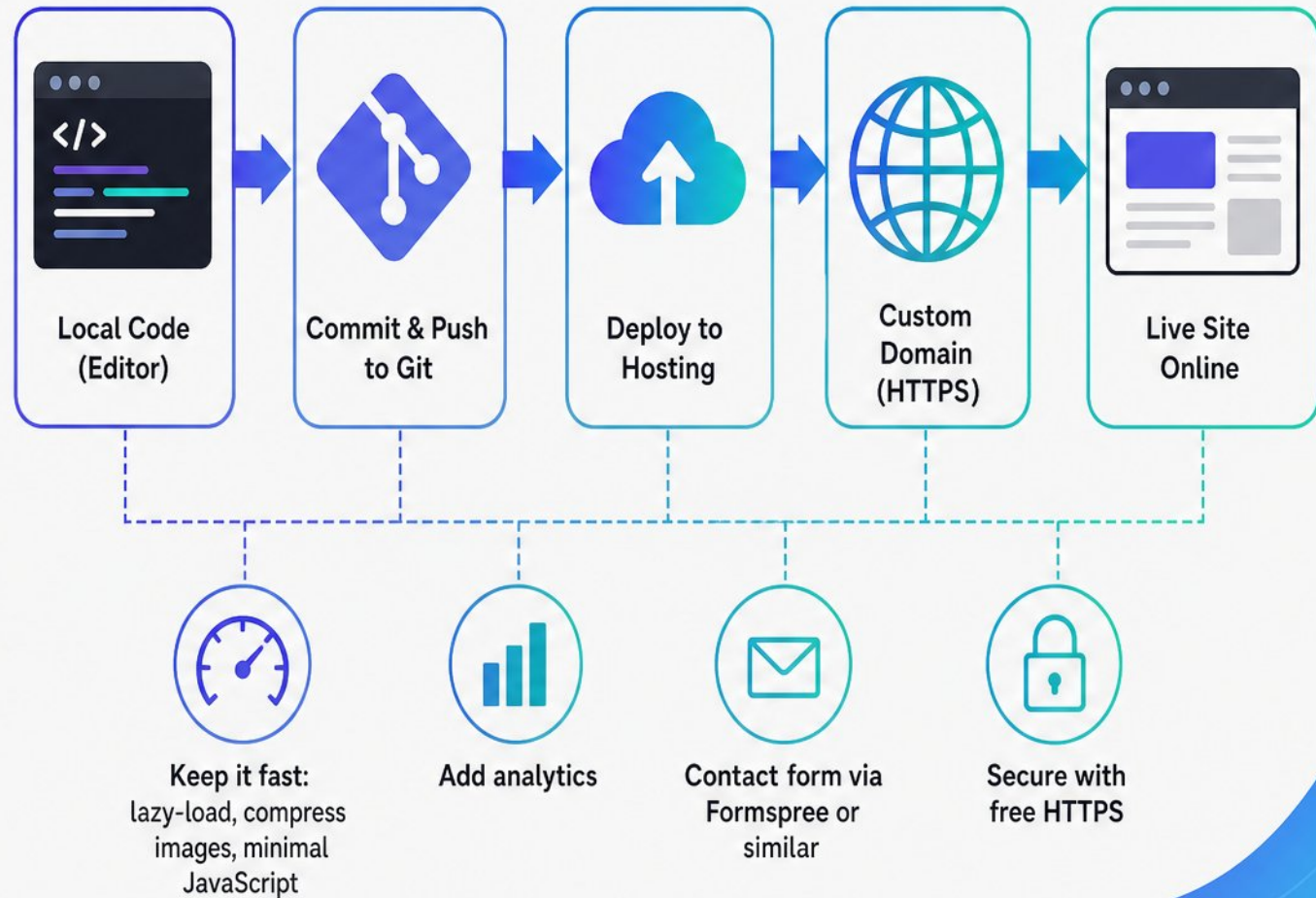


SCAN PATH



Building and Deploying the Site

- Choose a stack matching your target role: static, Next.js, Astro, or vanilla
- Deploy free on Vercel, Netlify, GitHub Pages, or Cloudflare Pages
- Connect a custom domain with free HTTPS via your host
- Add analytics and a contact form via Formspree or similar
- Keep it fast: lazy-load, compress images, minimal JavaScript



Making GitHub Work for You



- Recruiters scan pinned repos, READMEs, and recent activity



- Six strong repos beat forty abandoned ones



- Each README needs a live demo and clear problem



- Clean commit history shows consistency and care



- Profile README acts as a recruiter-facing cover letter

Pinned Repositories

Recent Activity

Profile README

Hi, I'm a builder.
I build practical tools that solve real problems.

```
const problem = "Real world";  
const solution = "Clear impact";  
build(problem, solution);
```



Showcase your best work up front.



Fresh activity signals ongoing engagement.



Clear problem. Live demo. Real impact.



Curate with intention. Quality, clarity, and consistency win opportunities.

Writing an About Page That Builds Trust

- ✓ - Keep it to 2-3 sentences: who you are, your focus, and what you're looking for
- ✓ - Show trajectory and growth; avoid pretending to be senior
- ✓ - Mention your current stack and the problems you solve
- ✓ - Include email, GitHub, and LinkedIn; skip the contact form
- ✓ - End with a clear call-to-action for outreach



Outreach and Applying with a Portfolio



Put portfolio URL in resume header, LinkedIn, and cover letters



Point recruiters to 2–3 specific projects most relevant to the role



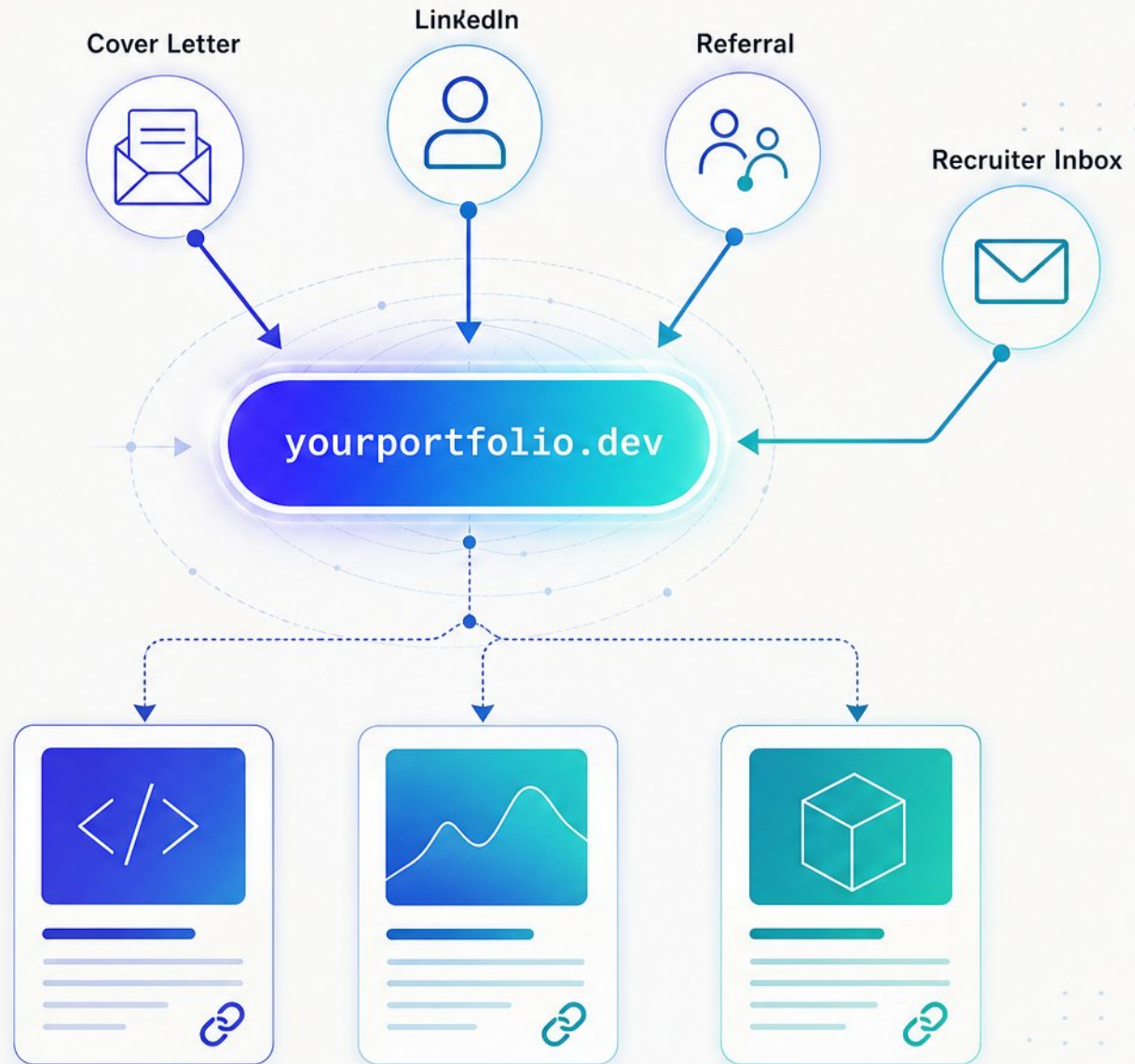
Use portfolio to start conversations with recruiters, communities, referrals



Treat silence or feedback as data—iterate, don't get discouraged



Consistent project titles and links across all channels build trust



Reviewing and Improving Over Time



- Schedule monthly portfolio reviews and update after each new project



- Track analytics to see which case studies get read and which are skipped



- Watch for stale warnings: outdated skills, broken links, or dormant projects



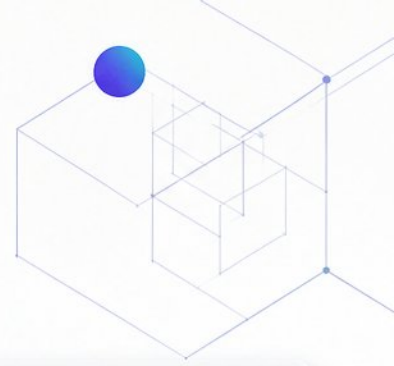
- Small fixes beat redesigns: update case studies, fix links, refresh demos

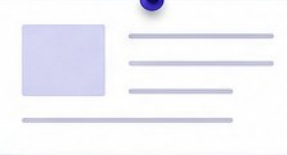
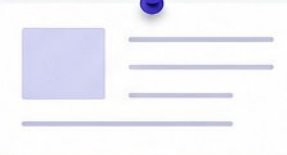
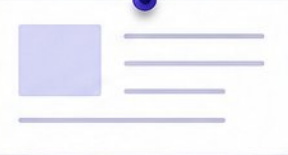
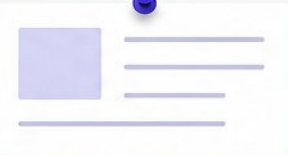
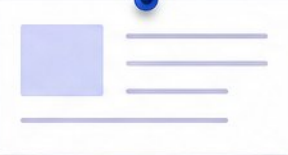
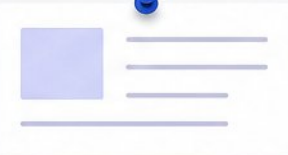
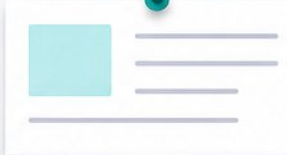
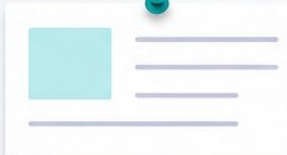


- Full redesign only when the portfolio fails to represent current skills

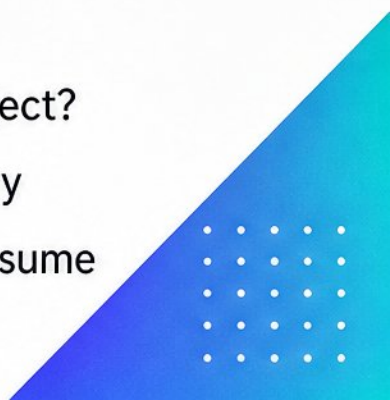
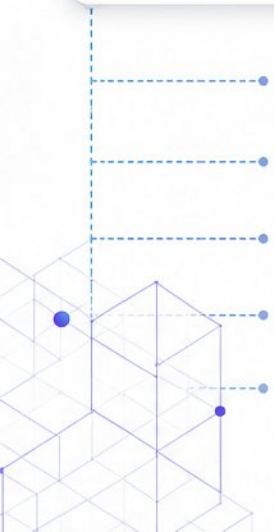


Putting It All Together: Your Portfolio Checklist



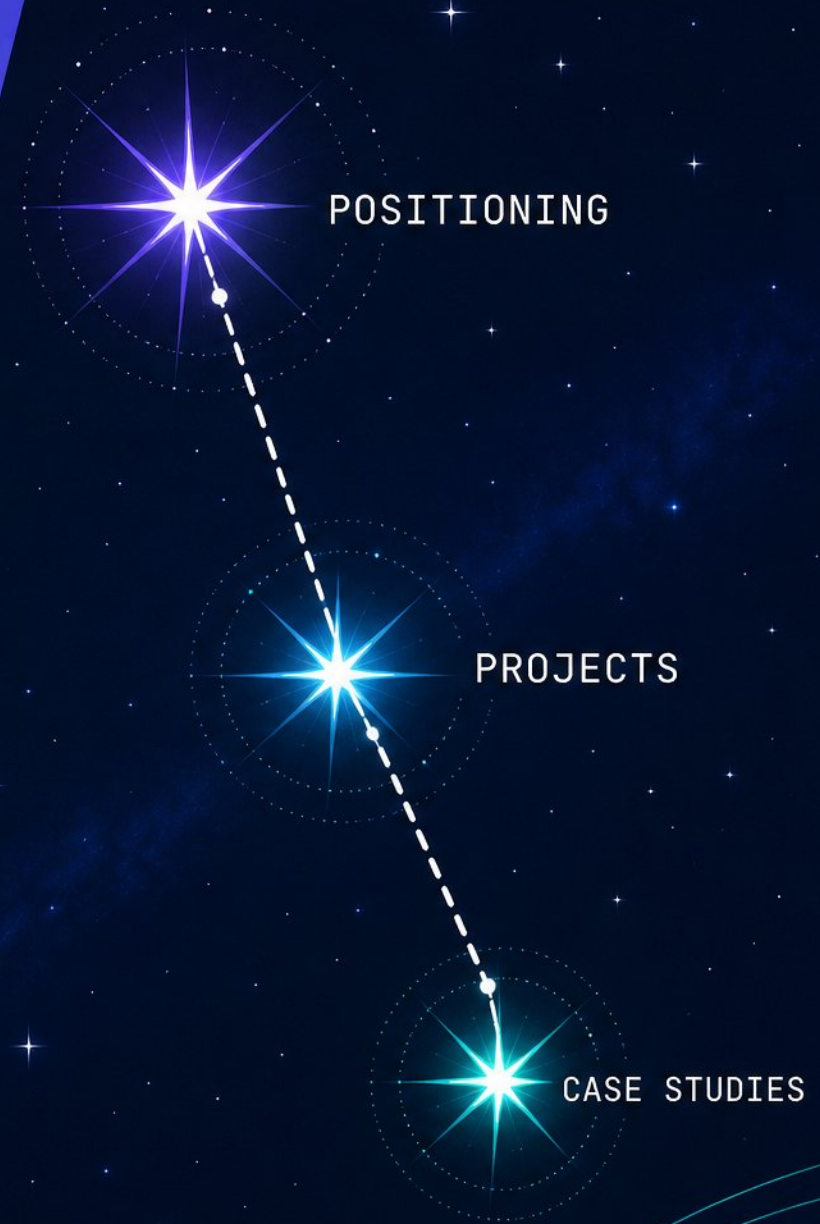
 LIVE DEMOS	 READMEs	 CONTACT INFO	 PERFORMANCE	 MOBILE TESTING
 	 	 	 	 

- ✓ - Verify every live demo, README, and contact link works
- ✓ - Test performance, mobile responsiveness, and accessibility
- ✓ - Run the 30-second self-scan: can a stranger find your best project?
- ✓ - Confirm each project links to a live URL and matching repository
- ✓ - Share your portfolio URL confidently across LinkedIn and your resume



Key Takeaways

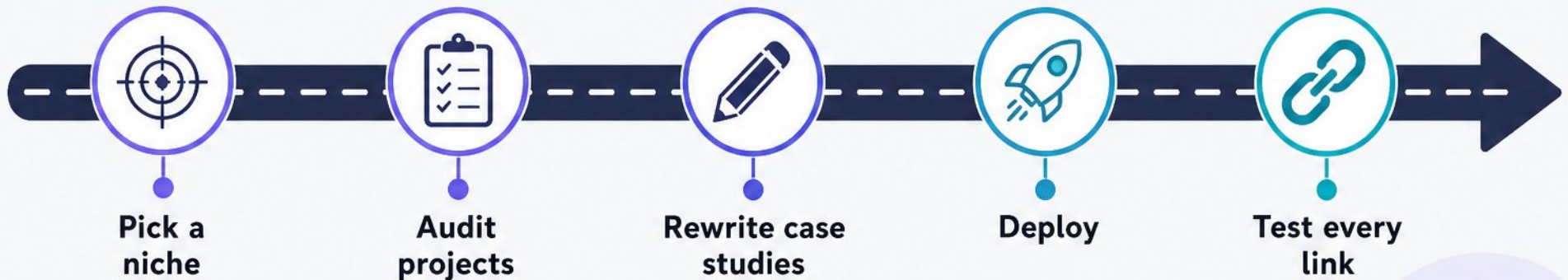
- A portfolio is proof of work: live, deployed, and documented projects
- Position yourself clearly, choose 3-5 projects, and write case studies
- Show 3-5 strong projects that highlight decisions and outcomes
- Keep the site fast, accessible, and mobile-responsive
- Treat portfolio maintenance as part of the job search



Action Plan and Next Steps

WEEK 1

WEEK 2



- > Week 1: Pick a niche, audit projects, rewrite case studies
- > Week 2: Deploy, test every link, share your URL broadly
- > Review monthly: refresh demos, remove outdated work, add new projects
- > Get feedback from mentors, coding communities, and honest self-review
- > Keep commit history active to show continuous learning



PersonWise.

PersonWise • AI digital-human course platform



Scan the code or click anywhere to watch this course live, with voice Q&A

personwise.ai/t/a29f7e27/public