Portfolio Teardown - Weyehn Reeves (Recent CS Grad)

September 15, 2025 - Prepared for hiring manager review and candidate iteration.

Snapshot (what was reviewed)

Live site: weyehnr.github.io/My-Profolio/ - the slug is misspelled ("Profolio"), which hurts credibility.

Public signals: LinkedIn (UMass Amherst) and GitHub show early front-end projects. These should be featured cohesively on the site. Profiles:

GitHub: WeyehnR

LinkedIn

P0 - fix these immediately

1   Rename the site slug from "My-Profolio" to "portfolio" (or buy a clean domain like weyehn.dev and point GitHub Pages to it). Docs: GitHub Pages custom domain.

2   Above-the-fold clarity: replace generic headline with a value proposition tailored to junior roles: "Frontend developer (React + TypeScript). I ship accessible, tested interfaces that load in <1s on 3G."

3   One-page resume link (PDF) - put a "Download resume" button in the header and footer. Keep ATS-friendly formatting.

4   Show three focused case studies only (GreatFrontEnd + two more that demonstrate state management, accessibility, and tests). Challenges: greatfrontend.com/projects.

5   Contact that converts: a "Hire me" button (mailto) and a short contact form using Formspree or Netlify Forms.

What to show (and how)

Case study template (use this structure on the portfolio and in each repo README):

1   Context: who needed what in one or two sentences.

2   Constraints: e.g., 3G performance budget, WCAG AA, no UI kit.

3   Key decisions: routing, state strategy, component composition, accessibility patterns.

4   Testing: unit (Jest), component (React Testing Library), accessibility (axe).

5   Performance: before/after Lighthouse scores and what moved the needle.

6   Result: 20-second demo video or GIF; links to live demo and code.

7   Snippet: 15-20 lines showing a real abstraction (custom hook, reducer, or accessible widget).

UX, accessibility, and performance - quick wins

Accessibility (non-negotiable)

1   Use semantic landmarks: ; add a skip link.

2   Visible focus styles; color contrast >= 4.5:1; meaningful alt text; use aria-live where content updates.

3   Use real buttons for actions and real links for navigation; associate labels with inputs (no placeholder-as-label).

```
<a class="skip-link" href="#main">Skip to content</a>
<main id="main">...</main>

<style>
  .skip-link { position: absolute; left: -999px; }
  .skip-link:focus { left: 1rem; top: 1rem; padding: .5rem; background: #fff; }
  :focus-visible { outline: 3px solid currentColor; outline-offset: 2px; }
</style>
```

Performance

1   Inline critical CSS for the hero; defer the rest.

2   Use modern images: srcset + sizes + loading="lazy"; compress aggressively (~100 KB hero, <50 KB cards).

3   Limit fonts: system stack or one webfont with font-display: swap.

4   Prevent layout shift: give media fixed width/height or an aspect-ratio.

```
<img
  src="img/card-640.jpg"
  srcset="img/card-320.jpg 320w, img/card-640.jpg 640w, img/card-1280.jpg 1280w"
  sizes="(max-width: 600px) 90vw, 400px"
  width="400" height="300" alt="Collections grid preview" loading="lazy">
```

SEO and social cards (so your link preview looks good)

Add a descriptive , meta description, Open Graph & Twitter tags, and JSON-LD 'Person' data. Example:

```
<head>
  <title>Weyehn Reeves - Frontend Developer (React, TypeScript, A11y)</title>
  <meta name="description" content="UMass Amherst CS grad building fast, accessible React apps. S
  <meta property="og:type" content="website">
  <meta property="og:title" content="Weyehn Reeves - Frontend Developer">
  <meta property="og:description" content="Fast, accessible React apps. Case studies + code.">
```

```
<meta property="og:url" content="https://weyehn.dev">
<meta property="og:image" content="https://weyehn.dev/og-card.jpg">
<meta name="twitter:card" content="summary_large_image">
<link rel="canonical" href="https://weyehn.dev">
<link rel="icon" href="/favicon.ico">
<script type="application/ld+json">
{
  "@context":"https://schema.org",
  "@type":"Person",
  "name":"Weyehn Reeves",
  "url":"https://weyehn.dev",
  "sameAs":[
    "https://github.com/WeyehnR",
    "https://www.linkedin.com/in/weyehn-reeves-169b60213"
  ],
  "jobTitle":"Frontend Developer",
  "alumniOf":"University of Massachusetts Amherst"
}
</script>
</head>
```

GitHub polish (recruiters click through)

1   Rename repos with typos (e.g., html-sample-profolio -> html-sample-portfolio) and add one-line summaries.

2   Pin six repos that reflect the stack you want to use professionally (React/TypeScript/a11y/testing).

3   Standardize READMEs: banner, live link, stack badges, case-study narrative, how to run, 'What I learned'.

4   Add basic CI with GitHub Actions to run lint, tests, and deploy Pages on main. Docs: GitHub Actions.

```
name: ci
on: [push, pull_request]
jobs:
  build:
    runs-on: ubuntu-latest
    steps:
      - uses: actions/checkout@v4
      - uses: actions/setup-node@v4
        with: { node-version: 20 }
      - run: npm ci
      - run: npm run lint && npm test -- --ci
  deploy:
    needs: build
    if: github.ref == 'refs/heads/main'
    runs-on: ubuntu-latest
    steps:
      - uses: actions/checkout@v4
      - uses: actions/setup-node@v4
        with: { node-version: 20 }
```

```
- run: npm ci && npm run build
- uses: actions/upload-pages-artifact@v3
  with: { path: dist }
- uses: actions/deploy-pages@v4
```

Content and copy

Hero copy: "I build fast, accessible React apps. Here are 3 case studies with code, tests, and performance numbers."

Project card copy: Responsive product collections grid. Role: dev. Stack: HTML/CSS. Notes: semantic layout, grid/flex, no CLS. Include links to GreatFrontEnd challenge and live code.

About: three lines max - what you like building, how you work, what you're learning next. Mention UMass for credibility, not as the headline.

Short, blunt hireability scorecard

1   First impression / professionalism: 5/10 (spelling + generic messaging hurt).

2   Projects shown / relevance: 6/10 (good starts; need deeper narrative + tests).

3   Accessibility & performance signals: Incomplete (treat as missing until demonstrated).

4   Conversion (resume + contact): Incomplete (add both).

5   GitHub polish: 6/10 (rename typos, pin strategically, add CI).

Ship the P0s and you're realistically at 8/10 for junior interviews.

7-day plan

1   Day 1-2: Domain + slug rename + hero rewrite + resume button.

2   Day 3: Convert three projects into full case studies; record a 20-second silent demo for each.

3   Day 4: Add accessibility and performance fixes (skip link, contrast, lazy images, font swap).

4   Day 5: SEO/OpenGraph + JSON-LD (use the snippet above).

5   Day 6: GitHub cleanup (rename, pin, READMEs, CI).

6   Day 7: Mobile dry-run + Lighthouse; ask 3 engineers for feedback; iterate.

Live links (clickable)

https://weyehnr.github.io/My-Profolio/

https://github.com/WeyehnR

https://www.linkedin.com/in/weyehn-reeves-169b60213

https://www.greatfrontend.com/projects

https://www.frontendmentor.io/

https://developers.google.com/web/tools/lighthouse

https://www.deque.com/axe/