- **Others:** If you prefer React-based workflows, Gatsby or Next.js can static-export a site, but they include heavy JS bundles and are less "minimal". A small portfolio doesn't require these – a simpler SSG (as above) will be easier to maintain and far faster. Similarly, older generators like Middleman or Metalsmith work too, but Jekyll/Eleventy are more than sufficient.

Example: *Fernfolio* is an Eleventy starter portfolio that showcases the power of a static site. It's a minimal template integrated with Netlify CMS for easy content editing [9]. It has no JS framework, uses responsive images, and scores high on Lighthouse (performance audits) out of the box [10]. This template (and others on sites like Jamstack Themes) could be a great starting point. You can customize the design while inheriting good practices (optimized images, SCSS support, minified assets, etc.). *Screenshot:* below is a preview of Fernfolio's design and structure:

Fernfolio – an Eleventy + Netlify CMS portfolio template (demo). It emphasizes speed (no front-end framework) and provides a clean project listing and blog section [9] [10].

All the above frameworks generate static files that can be hosted on GitHub Pages (or Netlify/Vercel). If using Jekyll, GH Pages can build it automatically on each push. For others, you might use GitHub Actions to build and deploy, or simply build locally and push the output to a `gh-pages` branch. The effort to set this up is one-time; afterwards your update workflow is just editing content and rebuilding.

2. Design Patterns & UX for Showcasing Creative Work

To give your portfolio a **high-end, yet minimal and clear feel**, consider these design and UX improvements:

- **Minimal, Text-Forward Layout:** Emulate the style of Chloe Scheffe's site (which you admire) by making text content the hero on your homepage. For example, Chloe's homepage lists each project in a simple table: showing the *Title, type (kind), client, year, and a text link for a thumbnail* [11]. This kind of list-oriented layout is clean and scannable. You could list your projects with brief descriptors (e.g. project name, category like "Brand Identity" or "Web App", client or personal project, year). Keep typography elegant but not over-designed – the focus should be on content. A sans-serif font with good readability or a mix of sans for body and a display font for headers can work. Ensure consistent styling for titles, metadata, etc., possibly mirroring a print-like simplicity (Chloe's site, for instance, looks almost like a text index of an archive).

- **Toggleable List vs. Image Grid View:** Implement a view toggle so users can switch between a text-only list of projects and an image preview grid. This caters to different browsing preferences – some users prefer reading project titles and info in a list, others are drawn to visual thumbnails [12]. A list view is efficient for scanning and reading; it shows more items on screen and follows a natural reading pattern (users read in an "F-shaped" flow) [13]. This is great when users are looking for specific project names or info. In contrast, a grid view is more visual and engaging, ideal for showcasing your design thumbnails – *images dominate and user attention is spread evenly across items* [14]. By providing both, you give visitors control: they can browse in a text-focused way or visually.

  Toggle Implementation: The toggle can be a simple button or link (e.g. "≡ List / □ Grid"). Technically, you have a few options: - **CSS/JS Toggle:** Load all content on one page – use CSS classes