

# Portfoli-you

**“A portfolio about you, by you”**

*CAPSTONE PROJECT PROPOSAL*

Ethan D. Townsend

## Table of Contents

<b>Overview</b>	<b>3</b>
Description	3-4
Why	4
Takeaway	5
<b>Features</b>	<b>6</b>
Core Requirements	6-11
Stretch Goals	8-13
<b>Technical Process</b>	<b>14</b>
Tech Stack	14-15
Proposed Schedule	15-17
<b>Conclusion</b>	<b>18</b>

---

# Overview

## Description

On average, recruiters spend only 6–8 seconds scanning a resume, but that jumps to roughly 3 minutes when reviewing a portfolio website([StandOut CV](#) and [UX Design CC](#)). That extra attention can be the difference in standing out, or falling into the crowd. **Portfoli-you** addresses this by giving anyone—regardless of technical background—a fast, visual way to build a compelling portfolio website. It's a desktop web-app with drag-and-drop, portfolio-focused widgets so users can design freely without touching backend code, then deploy via built-in guides to GitHub Pages, quick-share tunneling, or static export to take their site anywhere.

I'll distribute **Portfoli-you** from a dedicated download page on [ethantownsend.dev](#) with signed Windows/macOS installers, published checksums, a changelog, system requirements, and a privacy/security overview. The app will ship with an auto-update channel for incremental releases (with offline installers available) and step-by-step documentation for each supported publish path so users can get live quickly.

The product is local-first with optional cloud. Without signing in, users can create unlimited local projects, use drag-and-drop, preview, and export or publish to their own hosting with no lock-in. Signing in will unlock cloud conveniences such project sync and backups, with the optional, business-model idea of adding basic analytics, shared template gallery, and more.

To keep hosted costs predictable and make the business model clear, I'll apply sensible quotas to cloud features: for the MVP free tier, up to **1 cloud project per account** and **~500 MB**

of hosted assets (local projects remain unlimited). Paid tiers (e.g., Pro/Teams) can raise these limits and add collaboration, custom domains (SG-05), priority build queues, advanced analytics (SG-04), and support SLAs. These starting limits will be refined with real usage and validation.

As more and more people enter the workforce, it becomes increasingly harder to break from the mold of the standard resume and showcase their unique talents, skills, and experience all in one viewing. **Portfoli-you** is that solution

---

## Why

I'm committed to building a project I can be proud of, one that stretches my skills and stands on its own as a credible product. I'll treat it as a venture, not a class exercise, grounding decisions in practical business thinking: clear value proposition, basic market validation, sustainable pricing, and a realistic plan for acquiring and retaining users. The goal is to ship something usable, measurable, and capable of growing beyond the classroom. I'll prioritize clean architecture, maintainable code, testing, security, and clear documentation, iterating based on feedback and data to demonstrate engineering maturity, product sense, and delivery.

---

## Takeaway

By the end of this project, I will have built and shipped a full-fledged product end-to-end. I'll learn Electron and stand up a backend-driven update pipeline so anyone can install the app and receive updates globally. I'll deliver a real solution that outperforms existing options, with security, authentication, and support handled correctly. I'll own the codebase and be proud not just of the final result but of the craftsmanship and independence behind it. I'll make the product business-model ready and investor-friendly, and I'll instrument analytics to prove value, especially for users with physical portfolios or those hesitant to start a digital one, so I can demonstrate real adoption and impact. I will have made something I am both proud of, but can continue to be proud of. I want this product to be future-proof and business ready.

---

## Features

### Core Requirements (ordered by importance)

#	Requirement	User Story (short)	Acceptance Criteria (done when...)	Est. (hours)
CR-01	<u>Local-first (No account required)</u>	<i>As a user, I can use <b>Portfoli-you</b> entirely offline without creating an account.</i>	Onboarding offers Start locally / Connect account; Start locally routes to Home; cloud-only actions show locked state; local data persists across restarts.	6
CR-02	<u>Registration (Google &amp; Microsoft)</u>	<i>As a new user who wants cloud features, I can register by connecting a Google or Microsoft account.</i>	From onboarding/Settings I can Connect account via Google/Microsoft OAuth; first successful OAuth creates a user record; redirect to Home with cloud features enabled; cancel returns to local-only.	5
CR-03	<u>Login (Google &amp; Microsoft)</u>	<i>As a returning cloud user, I can log in with Google or Microsoft to access my synced content.</i>	From onboarding/Settings I can Sign in via Google/Microsoft; existing user loads profile and cloud projects; auth persists; expired tokens prompt re-auth without blocking local use.	5

<b>CR-04</b>	<u>Sign-out &amp; Mode Switching</u>	<i>As a user, I can sign out to return to local-only mode without losing local data.</i>	Sign out clears tokens and relocks cloud features; local projects remain; UI shows mode; cloud actions while signed out prompt to sign in.	<b>4</b>
<b>CR-05</b>	<u>Sidebar (Shell UI)</u>	<i>As a user, I can access primary areas from a persistent sidebar.</i>	Left vertical sidebar with Home, Modify, Deploy; Account & Settings pinned at bottom; active tab clearly highlighted; basic keyboard focus order works.	<b>2</b>
<b>CR-06</b>	<u>Navigation &amp; Routing</u>	<i>As a user, I can navigate between areas and routes behave predictably.</i>	Clicking tabs loads Home/Modify/Deploy views; back/forward works; selected tab/route persists across restart; cloud-only routes show a sign-in prompt when signed out.	<b>3</b>
<b>CR-07</b>	<u>Home/Dashboard</u>	<i>I see my portfolios and quick actions.</i>	List loads from local store; if signed in & sync on, cloud projects also show; Create/New works offline; Modify/Deploy routes; cloud actions gated when signed out.	<b>5</b>

<b>CR-08</b>	<u>Cloud Sync &amp; Limits</u> (Optional)	<i>My data can sync to the cloud with limits (1 portfolio/user).</i>	Applies only when signed in; cloud portfolios in Firestore; hard limit: 1/user; friendly “limit reached”; deleting a cloud project frees the slot; local projects not counted.	<b>6</b>
<b>CR-09</b>	<u>Editor: Widget System &amp; Functionality</u>	<i>Cloud Sync &amp; Limits</i> (Optional)	DnD add/reorder/duplicate/remove; snap-to-grid; undo/redo ( $\geq 10$ ); max 2 pages; widgets honor theme; properties in Content Panel; schema saved locally and syncs if enabled; existing projects load after new widget types added.	<b>16</b>
<b>CR-10</b>	<u>Editor: Drag-and-Drop Builder (2 pages max)</u>	<i>I can build a portfolio without code.</i>	Add/reorder/remove sections; curated widgets (text, image, carousel, video, link/API); max 2 pages; snap-to-grid; undo/redo; offline OK.	<b>25</b>
<b>CR-11</b>	<u>Content Panel + Validation</u>	<i>I can edit fields safely.</i>	Right-panel fields reflect selected widget/section, required prompts, URL checks; instant apply.	<b>10</b>
<b>CR-12</b>	<u>Live Preview + Responsive Toggles</u>	<i>I preview desktop/tablet/phone.</i>	Hot-reload edits; 3 breakpoints toggle; no manual refresh.	<b>10</b>



<b>CR-13</b>	<u>Themes &amp; Templates</u> (light)	<i>I start fast and restyle globally.</i>	≥2 starter templates; ≥3 themes; global color/typography edit; persists per project locally; syncs if enabled.	<b>5</b>
<b>CR-14</b>	<u>Autosave &amp; Recovery</u>	<i>I don't lose work.</i>	Autosave ≤30s; manual save; last-session recovery; local revision history (≥5); if signed in, queued changes flush when online.	<b>10</b>
<b>CR-15</b>	<u>Asset Manager (Images)</u>	<i>I manage images in one place.</i>	Central list; upload/replace; basic resize/compress; alt-text; dedupe; assets stored locally; if sync on, mirror to cloud; offline uploads queue.	<b>8</b>
<b>CR-16</b>	<u>Local Serve + LAN Preview</u>	<i>I can test locally before deployment.</i>	One-click “Start Local”; shows localhost & LAN URL; reachable from another device on the same network.	<b>8</b>
<b>CR-17a</b>	<u>Quick Share via Tunnel</u>	<i>I can share a live preview quickly.</i>	One-click tunnel (localtunnel/cloudflared); temporary public URL; Stop kills tunnel; status visible; no account required.	<b>10</b>
<b>CR-17b</b>	<u>GitHub Pages Deploy</u>	<i>I can publish globally via GitHub.</i>	Build static site; connect GH account; push to <a href="#">gh-pages</a> ; store chosen deploy method if signs in; show public URL.	<b>12</b>

CR-18	<u>Static Export (ZIP)</u>	<i>I can take my site anywhere.</i>	Export full bundle (HTML/CSS/JS/assets) to folder/ZIP; runs on plain static server.	10
CR-19	<u>Deployment Status Badges</u>	<i>I see what's live at a glance.</i>	Per-portfolio badges: [DEPLOYED], [DEPLOYED (TUNNEL)], or none; HTTP ping on open/deploy; shows last-known status when offline.	2
CR-20	<u>Desktop Installer (Win/macOS) + Auto-Update &amp; Project Migration</u>	<i>As a user, I can download/install Portfoli-you, and it auto-updates; my existing projects keep working after updates.</i>	Win/macOS installers; auto-update check; apply updates without reinstall; schemaVersion migration with local backup/log; "About & Updates" shows versions/notes and supports rollback.	12
CR-21	<u>Baseline A11y &amp; SEO</u>	<i>My site isn't broken for users/SEO.</i>	Flags missing alt text/low contrast; set title/description/social preview image.	4
CR-22	<u>Account &amp; Settings</u>	<i>I manage my account and quotas.</i>	Shows mode (Local-only vs Connected as <name>); view cloud usage/limits only when signed in; sign out; per-project Sync toggle; Delete cloud data (only if connected); Reset local cache.	5
CR-23	<u>Project Import/Export (.portfoliyou)</u>	<i>I can move full projects between machines.</i>	Export/import complete project to a single file; works offline; versioned/migratable.	8

## Stretch Goals (ordered by importance)

#	Stretch Goal	Scope (what's included)	Acceptance Criteria (Done when...)	Deps	Est. (hours)
SG-01	<u>Interactive Onboarding &amp; Tutorial</u>	First-run guided tour, sample project autoloading, context tips (editor, deploy), “Replay Tutorial” button on Home.	New user completes guided flow to: Create → Modify (add widget) → Preview (3 breakpoints) → Deploy (local/tunnel). “Replay” restarts without breaking current work.	CR-01, 02,03,05,07,12	12
SG-02	<u>Approved API Widgets Pack</u>	Add whitelisted embeds: GitHub repo list, YouTube playlist, Dev.to/Medium feed. Per-widget caching and safe	Each widget renders with a configured handle/playlist; export still works; if rate-limited, the widget displays a graceful	CR-05, 10,14	12

		fallbacks.	fallback.		
<b>SG-03</b>	<u>AI Assistant</u> <u>(MVP)</u>	Inline suggestions: (1) layout (section order/spacing), (2) copy (headline/bio), (3) improvement tips. Preview → Apply → Undo. Rate-limited calls; opt-in per project.	Each suggestion type can be generated, previewed, applied, and reverted; edits appear in history; no blocking if AI is unavailable.	<i>CR-05, 06,07,09</i>	<b>30</b>
<b>SG-04</b>	<u>Deployment</u> <u>Analytics</u> <u>(Provider-agnostic)</u>	Lightweight client tracker for pageviews/uniques, top pages, last 7/30 days, referrers; uptime pings (HTTP 200/timeout); privacy toggle.	After enabling analytics, the dashboard shows counts for a deployed site (GitHub Pages, OR Tunnel, OR <b>Portfoli-you</b> Host). Turning off removes the tracker from export/build.	<i>CR-11–15</i>	<b>16</b>
<b>SG-05</b>	<u>Portfoli-you</u> <u>Branded Hosting</u> <u>(Enterprise Preview)</u>	One managed environment: subdomain per site ( <b>*.portfoliyou.app</b> ), auto-HTTPS, atomic deploys, status + analytics page, usage caps (bandwidth/requests).  Simple org billing stub	User can deploy to branded host, gets live subdomain with valid cert; status shows “Up”/“Down” and last deploy time; analytics populate; delete removes site + data.	<i>CR-13–15</i>	<b>30</b>

(non-charging).

SG-06	<u>External Host</u>	Persist chosen deploy method;	For a GitHub Pages site, the app	CR-13,	8
	<u>Linking &amp; Deep</u>	show health and “Open	shows the current URL, last	15	
	<u>Links</u>	Dashboard” link for external providers; warn on mismatch between linked URL and exported version.	deploy time (from commit), health check, and a button to open the provider dashboard.		
SG-07	<u>Custom Domains</u>	CNAME connect, domain	User adds a custom domain, sees	SG-04	16
	<u>&amp; HTTPS (for</u>	verification, automatic cert	verified status, and the site serves		
	<u>Branded Host)</u>	provisioning/renewal, redirect from apex to <b>www</b> or vice-versa.	over HTTPS within cert SLA; removal reverts to subdomain.		
SG-08	<u>Advanced</u>	+3 full-site templates, +5	Switching themes/templates never	CR-08	10
	<u>Themes &amp;</u>	themes; theme preset	breaks layout; contrast warnings		
	<u>Template Market</u>	save/load; color-contrast	< 5% of default combinations.		
	<u>(Lite)</u>	guardrails.			

---

# Technical Process

## Tech Stack (\*: Technology to learn)

### Primary stack (JS/TS, Electron-first)

- **Desktop shell:** Electron + TypeScript
  - `electron`, `electron-builder`, `electron-updater`\*
- **UI:** React + Vite + Tailwind + shadcn/ui (+ Radix primitives)
  - `react`, `vite`, `tailwindcss`, `class-variance-authority`,  
`@radix-ui/react-*`
- **State/data fetching/validation:**
  - Global: `zustand`\*
  - Async/cache: `@tanstack/react-query`\*
  - Validation/schemas: `zod`\*
- **Designer (drag & drop + layout):**
  - `@dnd-kit/core` (+ modifiers for snap-to-grid)
  - Optional: `re-resizable` for handles
- **Preview:** In-app React preview that renders the same page model; three controlled breakpoints (no separate bundler needed for MVP)
- **Persistence:**
  - **Cloud:** Firebase Auth (Google & Microsoft) + Firestore (projects/pages/blocks/themes)
  - **Assets:** Firebase Storage (images) with content-hash paths for dedupe

- **Local:** Electron `userData` folder for autosave and **local revision snapshots** (JSON + `archiver` ZIP)
  - **Deploy & share:**
    - **Static export:** write `/dist-site` + ZIP via `archiver`
    - **GitHub Pages:** `octokit` (REST) to create repo/branch and push build artifacts
    - **Local serve + LAN:** tiny static server (`sirv` or `express`)
    - **Tunnel:** `cloudflared` (preferred) with fallback to `localtunnel`
  - **Testing & QA:** `vitest` (unit), `playwright` (E2E smoke); `axe-core` for a11y checks
  - **Packaging & updates:** `electron-builder` (NSIS on Win, dmg/zip on macOS) + `electron-updater` (GitHub Releases provider)
  - **Hardening:** `contextIsolation: true`, no `nodeIntegration` in renderer, IPC allow-lists
- 

## Proposed Schedule: (Sprint Length: 5-6 days, 4-5 hours a day)

### Sprint 0: (Project Setup & Foundations)

- Finalize architecture & tech stack (Electron, React/Vite/Tailwind, Firebase)
- Initialize repository with CI/CD pipeline (GitHub Actions).
- Create an Electron shell with the “Hello World” app.
- Set up Firebase project (Auth, Firestore, Storage).
- Draft schema for `project/page/widget/theme`.

**Sprint 1: (*Authentication & Navigation*)**

- **CR-01:** Implement Google & Microsoft OAuth sign-in/out.
- **CR-02:** Build sidebar/navigation skeleton (Home, Modify, Deploy)
- Routing between pages (Dashboard, Editor, Deploy).

**Sprint 2: (*Dashboard & Project Management*)**

- **CR-03:** Home/Dashboard page with portfolios list & patch-notes/tutorial slot.
- **CR-04:** Cloud projects + hard limit (2 per user).
- Project CRUD in Firestore (create, read, delete).

**Sprint 3: (*Editor Foundations*)**

- **CR-05:** Widget system (add, reorder, duplicate, remove widgets).
- **CR-06:** Drag-and-drop builder (2 pages, snap-to-grid).
- Undo/redo stack ( $\geq 10$  steps).
- Persist editor state in Firestore.

**Sprint 4: (*Content Editing & Preview*)**

- **CR-07:** Content panel with validation (URL checks, required fields).
- **CR-08:** Live preview with responsive toggles (desktop/tablet/phone).
- **CR-09:** Themes & templates ( $\geq 2$  templates,  $\geq 3$  themes).

**Sprint 5: (*Autosave & Asset Management*)**

- **CR-10:** Autosave, manual save, and local revision history.
- **CR-11:** Asset manager (image upload, replace, resize/compress).
- Link assets to Firebase Storage with deduplication.

**Sprint 6: (*Local Testing & Quick Share*)**

- **CR-12:** Local serve + LAN preview (tiny static server).
- **CR-13:** Quick share via tunnel (cloudflared/localtunnel).
- Show tunnel status badges.
- Verify LAN + tunnel access works across devices.

**Sprint 7: (*Deployment Options*)**



- **CR-14:** GitHub Pages deploy (push to **gh-pages** branch).
- **CR-15:** Static export (downloadable ZIP of site).
- **CR-16:** Deployment status badges (live/offline).
- Error handling for deployment failures.

### **Sprint 8:** *(Desktop Installer & Updates)*

- **CR-17:** Desktop installer for Windows/macOS (electron-builder).
- Auto-update integration with GitHub Releases.
- Schema migrations with versioning + backups.
- “About & Updates” screen with release notes.

### **Sprint 9:** *(Finalization & QA)*

- **CR-18:** Baseline accessibility & SEO checks.
- **CR-19:** Account & Settings page (email, quotas, delete account).
- QA testing (unit, E2E smoke tests).
- Bug fixes, UI polish, documentation.
- (Optional) Stretch goals: onboarding tutorial, API widgets.

### **Sprint 10:** *(Presentation & Delivery)*

- Package final build for Win/macOS.
  - Run demo: Auth → Create → Edit → Preview → Deploy.
  - Submit final documentation (README, setup guide, user guide).
  - Prepare and deliver the final presentation.
-

## Conclusion

**Portfoli-you** is more than just a capstone project—it is a fully envisioned product that solves a real problem for students, professionals, and job seekers alike. In today’s competitive job market, a static resume is no longer enough to stand out. By offering users a simple, powerful, and accessible way to create professional portfolio websites, **Portfoli-you** directly addresses this challenge, empowering individuals to showcase their skills, experiences, and personal brand in a more dynamic and impactful format.

Through its desktop application model with no account requirements, intuitive drag-and-drop editor, and flexible deployment options, **Portfoli-you** lowers the barrier to entry for web presence creation, making it possible for anyone, regardless of technical background, to build and share a unique digital identity. The emphasis on usability, maintainability, and scalability ensures that this product is not just a classroom exercise but a foundation for future growth.

This project represents a culmination of both technical and professional development: combining strong engineering practices, clear product thinking, and practical business awareness. By delivering **Portfoli-you**, I will demonstrate my ability to design, build, and ship a product that is polished, purposeful, and capable of real-world impact.

---