**Level 01 — CSS Motion Essentials (Days 01 – 10)**

**Theme:** Pure‑CSS/Tailwind motions you can master quickly and re‑use everywhere.  
**Daily budget:** 2‑3 h focused work.  
**Outcome:** 4 small cloned UI effects **+** 1 original “About Me” micro‑site that stitches them together.

|  |  |  |  |
| --- | --- | --- | --- |
| **#** | **Project (Clone / Original)** | **Days** | **What You’ll Build** |
| 1 | **Pulse CTA Button** (Clone) | 1 | An attention‑grabbing call‑to‑action that softly “breathes.” |
| 2 | **Fade‑in Hero Section** (Clone) | 2 | Simple hero header that fades + slides up on load. |
| 3 | **Card Hover Lift** (Clone) | 3 | Product/info card that lifts & casts a shadow on hover. |
| 4 | **Sticky Top Navbar** (Clone) | 4 | Navigation bar that sticks once scrolled past hero. |
| 5 | **Mini “About Me” Microsite** (Original) | 05‑07 | One‑page profile assembling all four effects above. |

A rest & reflection cycle follows:

|  |  |
| --- | --- |
| **Day** | **Activity** |
| 8 | **Rest / Inspiration Scroll** — step away, gather inspiration shots on Dribbble/Behance. |
| 9 | **Case Study & README** — write a concise project write‑up, record short GIFs, push to GitHub. |
| 10 | **Level Retro + GSAP Setup** — note what went well, install GSAP starter files ready for Level 02. |

**7/21/2025**

**📘 Project 1 – Pulse CTA Button (Day 01)**

**Goal:** Recreate a circular “Sign Up” button that gently pulses to draw attention.

**Tools/Libraries:**  HTML5, Tailwind CSS, **@keyframes**

**Key Features:**

* Scale + opacity pulse loop (ease‑in‑out)
* Accessible focus outline
* Hover state pauses pulse

**Step‑by‑Step (≈ 2 h)**

1. **Scaffold (15 min)** – index.html with a section.flex.center.✔
2. **Button Markup (10 min)** – <button class="cta-btn">Sign Up</button>. ✔
3. **Tailwind Config (10 min)** – Create custom pulse keyframes in tailwind.config.js.
4. **Animate (40 min)** – Apply animate-pulse-cta utility, tune animation-duration:1.8s. ✔
5. **Hover & Focus (20 min)** – On hover: and focus: add scale-105 & thicker outline. ✔
6. **Accessibility Check (10 min)** – Verify color contrast ≥ 4.5 : 1. ✔
7. **Git Commit (15 min)** – “feat: pulse‑cta‑button”.✔

**📘 Project 2 – Fade‑in Hero Section (Day 02)**

**Goal**:

A hero banner whose heading, sub‑heading, and button fade & slide up sequentially on page load.

**Tools/Libraries**

HTML, Tailwind, CSS variables

**Key Features**

Staggered fade‑in via custom animation-delay

Responsive full‑viewport height

Optional background‑image cover

**Step‑by‑Step (≈ 2 h)**

1. **Layout (20 min)** — min-h-screen grid place-items-center. (using flex) ✔
2. **Content (15 min)** — <h1>, <p>, CTA.✔
3. **Keyframes (20 min)** — from{opacity:0;transform:translateY(20px)} to{...}.
4. **Stagger (25 min)** — Add style="--delay:0.2s" custom property; in CSS animation- delay:var(--delay).✔
5. **Responsive Typography (15 min)** — Use Tailwind’s fluid font utilities.
6. **Commit & Deploy (15 min)** — Push & preview with Vercel.

**📘 Project 3 – Card Hover Lift (Day 03)**

| **Goal** | **Tools/Libraries** | **Key Features** |
| --- | --- | --- |
| A card that elevates on hover with shadow & transform, revealing subtle info. | HTML, Tailwind transition utilities | • transition-transform & ease-[cubic‑bezier]• Shadow goes from sm → xl• Prefers‑reduced‑motion fallback |

**Step‑by‑Step (≈ 2 h)**

1. **Structure (20 min)** — Container with three .card items.
2. **Base Style (20 min)** — Rounded corners, image top, text block bottom.
3. **Hover State (25 min)** — hover:scale-105 hover:shadow-xl.
4. **Focus‑within (15 min)** — Same effect for keyboard nav.
5. **Motion‑safe (10 min)** — Wrap transform in @media (prefers-reduced-motion:no-preference).
6. **Doc & Commit (15 min)**.

**📘 Project 4 – Sticky Top Navbar (Day 04)**

| **Goal** | **Tools/Libraries** | **Key Features** |
| --- | --- | --- |
| Navbar that becomes fixed with subtle backdrop blur once you scroll past the hero. | Tailwind, **Intersection Observer** | • Adds backdrop-blur & bg-white/80 when stuck• Simple scroll detection in ≤ 20 LoC JS• Mobile hamburger (static) |

**Step‑by‑Step (≈ 2 h)**

1. **Markup (20 min)** — Hero followed by long filler content.
2. **Nav Style (20 min)** — Flex logo + links, transition background.
3. **JS Observer (30 min)** — Observe hero sentinel; toggle .is-sticky.
4. **Backdrop Blur (10 min)** — Tailwind class backdrop-blur-md.
5. **Test Mobile (15 min)** — Ensure sticky works on iOS Safari.
6. **Commit (10 min)**.

**🎨 Project 5 – Original “About Me” Microsite (Days 05‑07)**

| **Goal** | **Tools/Libraries** | **Key Features / Animations** |
| --- | --- | --- |
| Combine *all four* effects in a single‑page personal profile; add light content & palette. | HTML5, Tailwind, custom CSS keyframes, Netlify/Vercel | • Hero uses Fade‑in  • CTA pulse button links to email  • Portfolio cards use Hover Lift• Navbar sticks on scroll• Responsive & accessible |

**Day 05 — Layout & Content *(≈ 2 h)***

1. **Wireframe quickly** on paper/Figma (15 min).
2. **Scaffold pages**: sections — Hero, About, Projects, Contact (30 min).
3. **Import components** from previous days (40 min).
4. **Fill basic copy & placeholder images** (35 min).

**Day 06 — Integrate Animations *(≈ 2 h)***

1. **Bring in fade‑in hero keyframes**; adjust delays (30 min).
2. **Place CTA pulse button** in Contact section (20 min).
3. **Apply card hover lift** to Projects grid (30 min).
4. **Hook up sticky navbar JS** (25 min).
5. **Contrast & A11y pass** (15 min).

**Day 07 — Polish & Deploy *(≈ 2 h)***

1. **Set color scheme** with Tailwind :root custom properties (20 min).
2. **Responsive tweaks** for ≤ 400 px width (25 min).
3. **Performance pass** – compress images, remove unused CSS (25 min).
4. **Deploy on Netlify**; set custom sub‑domain (20 min).
5. **Record 10‑sec GIF demo** with ScreenToGif for README (15 min).
6. **Final commit + tag v1.0** (15 min).

**🛋️ Day 08 — Rest / Inspiration**

* Go outside, sketch ideas, scroll curated inspiration (max 30 min).
* Do **not** code.

**📝 Day 09 — Case Study & README**

* Write **WHAT, WHY, HOW, RESULTS** in Markdown.
* Embed hero GIF & Core Web Vitals screenshot.
* Push to GitHub Pages.

**🔧 Day 10 — Retrospective & GSAP Setup**

* List 3 wins & 3 pain points.
* Install **GSAP** via npm or CDN; fork the official CodePen starter.
* Create level-02 folder to stay organized.

**✅ Deliverables for Level 01**

1. **/level‑01/** folder with 5 sub‑projects in Git.
2. Live “About Me” microsite URL.
3. README with GIFs + reflection.
4. Prepared GSAP boilerplate for Level 02.

Stay disciplined with the 2‑3 h cap, and you’ll roll into Level 02 ready for timeline‑based JavaScript animation. Good luck—ship something small **every day**!