Site Development

- Figma to Dev (HTML & Elementor) Handoff Quick Sheet

Figma to Dev (HTML & Elementor) Handoff Quick Sheet

This guide ensures our Figma designs are structured in a way that makes development (whether custom HTML or WordPress/Elementor) fast, accurate, and scalable.

1. Placeholder Content Rules

To make it easy for developers to identify what is final content and what needs to be swapped out or removed in production, follow these rules:

- **Placeholder Images:** Use **placehold.co** for all non-final images.
- *How to use:* You can use the URL format `[https://placehold.co/WIDTHxHEIGHT](https://placehold.co/WIDTHxHEIGHT)` (e.g., `[https://placehold.co/600x400](https://placehold.co/600x400)`). You can either run a Figma plugin that inserts these directly, or place a text layer over a gray box that clearly states the placeholder URL so the dev knows the exact dimensions to build for.
- **Placeholder Text:**
- For generic body copy, use standard **Lorem Ipsum**.
- For specific headings, dynamic content (like blog feeds), or functional copy, prefix the text with `**[placeholder]**`.
- *Example:* `[placeholder] 5 Tips for Web Design` or `[placeholder] User Testimonial here`.
- *Why?* Developers can do a global site search for the exact string `[placeholder]` before launch to ensure no dummy data made it to production.

2. Indicating Interactions & Animations

Developers need to know *how* things move, not just how they look. When designing elements like hovers, tabs, and accordions, use **Figma Interactive Components** or visual redlining.

- **Hover States (Buttons, Links, Cards):**
  - **Best Practice:** Create a Component Set with two variants: `Default` and `Hover`. Link them in the Prototype tab using "While hovering."
  - **Handoff Note:** If there is a specific CSS transition you want (e.g., *scale up slightly, ease-in 0.3s*), leave a sticky note next to the component explicitly stating the timing and effect.
- **Accordions (FAQs, Menus):**
  - **Best Practice:** Always design at least two states: `Collapsed` and `Expanded`.
  - **Handoff Note:** Show the padding and spacing *inside* the expanded state. For Elementor, note if you expect the accordion to push content down or overlay it.
- **Tabs:**
  - **Best Practice:** Design the `Active` and `Inactive` tab UI.
  - **Handoff Note:** Create a frame showing the content for "Tab 1" and a separate frame showing "Tab 2" to demonstrate how the layout handles different content heights.

3. Structural Best Practices (The "Flexbox" Mindset)

Elementor and modern HTML rely heavily on CSS Flexbox. Your Figma files should reflect this.

- **Use Auto Layout (Shift + A) for EVERYTHING:** Auto layout forces you to design with HTML/CSS constraints. If a button has `20px` padding in Auto Layout, the developer can copy those exact padding values into Elementor.
- **Spacing:** Rely on fixed gaps in Auto Layout rather than dragging elements around.
- **Group Like Containers:** Structure your layers logically. (e.g., `Section` > `Container` > `Row` > `Column`). This directly translates to how an Elementor page is built.

4. Design System & Global Styles

- **Typography:** Do not use loose text styling. Define **Figma Text Styles** (e.g., `H1`, `H2`, `Body Text`, `Button`). These map directly to Elementor's Global Fonts.
- **Colors:** Define **Figma Color Styles** (e.g., `Primary`, `Secondary`, `Accent`, `Text-Dark`). This prevents the developer from dealing with 15 slightly different shades of blue.

5. Developer Communication

- **Layer Naming:** Avoid `Frame 1432` or `Rectangle 5`. Use semantic names like `Hero Section`, `Pricing Card`, or `Submit Button`.
- **Dev Mode Notes:** Use Figma's annotation tools or a bright red "Sticky Note" component to leave explicit instructions (e.g., "*Devs: Build this using the Elementor Pro Posts Widget*" or "*This header should be sticky on scroll*").