

■ Manual de Estandarización Web & Estilo (V1.0)

Proyecto: Portafolio Yurak Chalen

Stack: HTML5, CSS3 (Moderno), Vanilla JS, Bootstrap 5.

1. Identidad Visual (Design System)

La estética se define como "**Industrial Tech Minimalist**". Prioriza la utilidad, el contraste oscuro y la tipografía limpia sobre la decoración excesiva.

1.1 Paleta de Colores (Inmutables)

Nunca uses valores HEX directos en los componentes. Usa siempre las variables CSS.

1.2 Tipografía

- **Principal:** 'Inter', sans-serif. Para todo el UI general.
 - **Código/Terminal:** 'Fira Code', monospace. Exclusivo para la ventana de contacto y snippets de código.
 - **Pesos:**
 - 300 (Light): Cuerpo de texto largo.
 - 700 (Bold): Títulos y Botones.
-

2. Reglas de CSS (Arquitectura)

2.1 Manipulación de Color

Prohibido: Usar opacity para transparencias de fondo (afecta al texto hijo).

Obligatorio: Usar color-mix para variantes de color.

2.2 Curvas de Animación (Motion)

Para mantener la sensación de "fluidez mecánica", todas las transiciones de movimiento (transform, width, height) deben usar esta curva Bézier personalizada, nunca ease o linear por defecto.

2.3 Capas (Z-Index)

Se utiliza un sistema de capas estricto para evitar conflictos de apilamiento (como pasó en el Hero).

- z-index: 1: Elementos base interactivos (botones, inputs).
- z-index: 2: Elementos decorativos que requieren superposición visual (bordes glass, efectos).
- z-index: 10: Elementos flotantes absolutos (pelota del juego).
- z-index: 100: Navegación sticky y controles de carrusel.
- z-index: 9999: Ventana de Terminal Maximizada.

3. Protocolos de JavaScript (Lógica)

3.1 Cero "Spaghetti Code" en HTML

Regla de Oro: Nunca escribir lógica JS dentro de atributos HTML.

-  **Prohibido:** `<button onclick="doSomething()">`
-  **Obligatorio:** Usar `addEventListener` en el archivo `script.js`.

3.2 Delegación de Eventos

No añadas un *listener* a cada elemento de una lista. Añádalo al contenedor padre.

3.3 Separación de Estado y Vista

El DOM no es tu base de datos. Si la terminal tiene datos, estos deben vivir en un objeto JS (`tabManager.tabs`), no solo en el HTML. El HTML se "renderiza" basándose en esos datos.

4. Guía de Componentes (Snippets)

4.1 Botones de Acción (CTA)

Deben seguir siempre la clase `.btn-accent-green`.

- **Estado normal:** Sin sombra, texto oscuro (`--bg-dark`).
- **Hover:** Texto blanco, glow trasero difuminado.

4.2 Texto Interactivo (Glow)

Cualquier párrafo introductorio importante debe llevar la clase `.lead-interactive`.

- **Comportamiento:** Color `--text-muted` por defecto. Al pasar el mouse, cambia a blanco puro `#ffffff` con `text-shadow`.

4.3 Tarjetas (Glass Effect)

Las tarjetas (proyectos, info) usan un borde sutil casi invisible que se ilumina al hover.

5. Accesibilidad (A11y) - NO NEGOCIABLE

Para mantener el estándar profesional, no podemos romper la web para lectores de pantalla.

1. **Elementos Decorativos de Texto:** Si rompes una palabra con `span` para animarla (como "I-nteract-i-vos"), el contenedor padre **debe** tener `aria-label="PalabraCompleta"` y los hijos `aria-hidden="true"`.
 2. **Botones de Iconos:** Todo botón que solo sea un icono (`<i class="bi-x"></i>`) debe tener obligatoriamente un atributo `aria-label="Descripción de la acción"`.
 3. **Contraste:** Al quitar el outline de los inputs (`outline: none`), es obligatorio definir un `border-bottom` de alto contraste en el estado `:focus`.
-

6. Checklist de Validación para Nuevas Secciones

Antes de publicar una nueva sección o feature, responde "Sí" a todo:

1. ☐ ¿He usado variables CSS en lugar de colores HEX?
2. ☐ ¿El HTML está limpio de atributos `onclick`, `onmouseover`, etc.?
3. ☐ ¿Si hay animaciones, usan la curva `cubic-bezier` del sistema?
4. ☐ ¿Los textos animados son legibles para un lector de pantalla?
5. ☐ ¿En móvil, los elementos tienen suficiente espaciado táctil (min 44px)?