

# Cómo pensar y programar: CatPhotoApp + Café Menu (Guía con ejemplos)

Objetivo: que cualquiera pueda entender “para qué sirve cada cosa” y replicar estos patrones en sus propios proyectos. Usaremos explicaciones claras y fragmentos de código cortos para fijar ideas.

- HTML define el significado y la estructura (qué es cada cosa).
- CSS define la apariencia (cómo se ve).
- Primero semántica y accesibilidad; luego estilos; después interacción.

# Proyecto 1 — CatPhotoApp: estructura y semántica

La página se organiza en secciones temáticas dentro de un contenido principal. Esta estructura ayuda a lectores de pantalla y a mantener el proyecto ordenado.

## Contenido principal (main) y secciones

El contenedor principal agrupa todo el contenido central; cada tema va en su propia sección con un subtítulo claro.

```
<main>
  <section>
    <h2>Cat Photos</h2>
    ...
  </section>
  <section>
    <h2>Cat Lists</h2>
    ...
  </section>
</main>
```

Los títulos están jerarquizados: un título general de la página y subtítulos por sección. Esta jerarquía crea una “tabla de contenidos mental”.

```
<h1>CatPhotoApp</h1>  <!-- título principal -->
<h2>Cat Photos</h2>    <!-- subtítulo de sección -->
<h3>Things cats love:</h3> <!-- subtítulo más específico -->
```

## CatPhotoApp: enlaces e imágenes con propósito

Los enlaces conectan a recursos. Si un enlace te saca del sitio actual, puedes abrirlo en una pestaña nueva.

```
<a href="https://freecatphotoapp.com">galería</a>
<a target="_blank" href="...">abrir en otra pestaña</a>
```

Las imágenes incluyen una descripción alternativa breve que explica su función o contenido; esto es vital para accesibilidad y cuando la imagen no carga.

```

```

Cuando una imagen está dentro de un enlace, se vuelve clicable: el usuario puede “hacer clic en la imagen” para ir al destino.

```
<a href="https://freecatphotoapp.com">
  
</a>
```

# CatPhotoApp: listas y figuras con leyenda

Usamos listas no ordenadas para colecciones sin orden y listas numeradas para rankings o secuencias.

```
<ul>
  <li>catnip</li>
  <li>laser pointers</li>
  <li>lasagna</li>
</ul>

<ol>
  <li>flea treatment</li>
  <li>thunder</li>
  <li>other cats</li>
</ol>
```

Una figura agrupa una imagen y su leyenda. Esto comunica que el texto explica esa imagen concreta.

```
<figure>
  
  <figcaption>Cats love lasagna.</figcaption>
</figure>
```

El énfasis textual mejora la comprensión: énfasis leve y énfasis fuerte comunican diferentes niveles de importancia para lectores y tecnologías asistivas.

```
<figcaption>Cats <em>love</em> lasagna.</figcaption>
<figcaption>Cats <strong>hate</strong> other cats.</figcaption>
```

## CatPhotoApp: formularios accesibles y claros

Los formularios se leen mejor si agrupamos preguntas relacionadas y ponemos un título para cada grupo. Eso mejora la navegación con teclado y la comprensión.

```
<form action="https://.../submit-cat-photo">

  <fieldset>
    <legend>¿Interior o exterior?</legend>
    <label><input type="radio" name="indoor-outdoor" value="indoor" checked> Interior</label>
    <label><input type="radio" name="indoor-outdoor" value="outdoor"> Exterior</label>
  </fieldset>

  <fieldset>
    <legend>Personalidad</legend>
    <input id="loving" type="checkbox" name="personality" value="loving" checked>
    <label for="loving">Cariñoso</label>
    ...
  </fieldset>

  <input type="text" name="catphotourl" placeholder="URL de la foto" required>
  <button type="submit">Enviar</button>
</form>
```

- Selección única (radios): comparten el mismo nombre de grupo; solo una respuesta es válida.
- Selección múltiple (checkboxes): comparten nombre, pero se pueden marcar varias opciones.
- Etiquetas asociadas: cada control tiene su texto; mejora el foco y el clic.
- Validación nativa: campos obligatorios o con formato sugerido reducen errores.

# Proyecto 2 — Café Menu: de la estructura al diseño

Objetivo: mostrar una carta de productos ordenada, legible y estética. La estructura separa el contenido por categorías y cada producto se muestra en una fila con su precio.

## Bloque del menú y secciones

Un contenedor envuelve todo el menú para centrarlo, limitar su ancho máximo y aplicar un fondo y un acolchado uniformes. Luego dividimos por categorías (Cafés y Postres).

```
<div class="menu">  
  <section>  
    <h2>Coffee</h2>  
    <article class="item">  
      <p class="flavor">French Vanilla</p>  
      <p class="price">$3.00</p>  
    </article>  
    ...  
  </section>  
  <section>  
    <h2>Desserts</h2>  
    ...  
  </section>  
</div>
```

## Filas de producto

Cada fila es un artículo con dos partes: nombre/descripción y precio. El lector puede barrer rápidamente la columna de precios a la derecha.

```
<article class="item">  
  <p class="dessert">Donut</p>  
  <p class="price">$1.50</p>  
</article>
```



# Café Menu: estilos clave y patrones reutilizables (CSS)

El CSS da forma visual a la página. En este proyecto se usan patrones simples pero potentes: centrado del contenedor, tope de ancho, columnas por fila y estados de enlace.

## Contenedor centrado y ancho legible

Un contenedor suele usar un ancho relativo (porcentaje) con un límite máximo. Así se adapta a móviles y no se vuelve ilegible en pantallas grandes.

```
/* contenedor del menú */  
  
.menu {  
    width: 80%;          /* ancho fluido */  
    max-width: 500px; /* tope legible */  
    margin: 0 auto;    /* centrado horizontal */  
    padding: 20px;      /* aire interior */  
    background: burlywood;  
}  
  
Fila con dos columnas (nombre y precio)
```

Una forma sencilla es usar cajas en línea. La parte del nombre ocupa ~75% y el precio ~25%, con alineaciones opuestas.

```
.item p {  
    display: inline-block;  
    font-size: 18px;  
    margin: 5px 0;  
}  
.dessert, .flavor { width: 75%; text-align: left; }  
.price { width: 25%; text-align: right; }
```

Títulos, separadores e iconos

Los títulos escalan en tamaño para marcar jerarquía. Las líneas horizontales separan secciones; los iconos refuerzan la identidad de cada categoría y se centran fácilmente.

```
h1 { font-size: 40px; }
h2 { font-size: 30px; }
hr { height: 2px; background: #8b4513; border: none; }
img { display: block; margin: 0 auto; }
```

## Estados de enlace

Los enlaces cambian de aspecto al pasar el puntero o activarse; esto aporta feedback visual y mejora la usabilidad.

```
a, a:visited { color: #000; }
a:hover, a:active { color: #8b4513; }
```

# Principios transversales y cómo depurar

- Semántica clara → accesibilidad mejor: estructura el contenido antes de estilizar.
- Nombres de clase descriptivos y consistentes: facilitan mantener y reutilizar.
- Patrón ancho fluido + tope: buen equilibrio para lectura en muchas pantallas.
- Alineaciones coherentes: ojo entrenado del usuario ahorra tiempo al leer precios.
- Textos alternativos e inputs etiquetados: nadie queda fuera de la experiencia.

## Depurar rápido

- Contornos temporales: agrega bordes a bloques para ver su caja real durante el diseño.
- Prueba por teclado: Tab para recorrer controles; el foco debe seguir un orden lógico.
- Revisa contraste: colores legibles en distintos fondos y estados.
- CTRL/CMD+Shift+I: usa las herramientas del navegador para inspeccionar y ajustar estilos en vivo.

Con estos patrones y ejemplos, puedes recrear desde cero ambos proyectos o escalarlos (por ejemplo, convertir filas a un sistema flexible con alineación vertical y gaps).