



Centro de Bachillerato Tecnológico Agropecuario No. 186

"Lic. Andrés Quintana Roo"

CC7. 23D7A0186P

"2024, Año de Francisco Villa, el revolucionario del pueblo"

Realizado por: Camila Islas Romero, Yadira Jazmín Cocom Poot, Ailed Guadalupe Che Chi, Samantha Isabela Kuk Hernández y Yamilex Kuk Chable



Manual para crear y diseñar páginas web



Docente: Jorge Alejandro Soberanis Teran.

ASIGNATURA: TEMAS DE FILOSOFIA

Fecha de entrega: Kantunilkin, 11 de marzo de 2025

1. Introducción.....	3
2. Planificación del proyecto.....	4
3. Diseño y creación del contenido.....	6
4. Diseño de la interfaz.....	12
5. Desarrollo del sitio web.....	14
6. Pruebas y Lanzamiento.....	21
7. Conclusión.....	22

En este manual, te guiare paso a paso para crear una pagina web desde cero. Ya que es un proceso que requiere planificación, diseño y desarrollo, aprenderás los conceptos básicos de diseño web y estructura y funcionalidad, así como las mejores practicas para crear una experiencia de usuario optima.

Nuestro principal objetivo con este manual es facilitar la comprensión y desarrollo de las paginas web para cualquier persona que esté interesado en desarrollar. Asimismo, recalcar la importancia que tiene en este proyecto la inteligencia artificial (IA), ya que ha transformado la sociedad moderna impactando múltiples áreas como la educación, la salud, la economía y la tecnología porque permite realizar tareas repetitivas con mayor rapidez y precisión que los humanos.

La aplicación en el diseño web no solo mejora la experiencia del usuario como, sino que también optimiza los procesos de desarrollo y mantenimiento, facilitando la creación de plataformas innovadoras y accesibles para todas.

Capítulo 1.- Planificación del Proyecto de Diseño y Creación de Páginas Web

Realizado por: Camila Islas Romero

1.1. Definición de Objetivos

1.1.1.-Objetivo General: Crear una página web que ofrezca [especificar el servicio o producto] y que sea accesible, atractiva y funcional para los usuarios.

- Aumentar la visibilidad en línea a través de un diseño optimizado para SEO.
- Mejorar la experiencia del usuario mediante una interfaz intuitiva y responsiva.
- Generar leads o ventas mediante llamados a la acción claros y efectivos.
- Fomentar el engagement a través de contenido relevante y actualizado.

1.2. Identificación del Público Objetivo

- Demografía
- Edad: (Especificar rango de edad)
- Género: (Especificar género si es relevante)
- Ubicación: (Especificar áreas geográficas)
- Psicografía
- Intereses: (Especificar intereses relevantes al producto/servicio)
- Comportamiento en línea: [Cómo interactúan con sitios web similares, redes sociales, etc.]

1.2.1 Necesidades y Problemas

¿Qué necesidades específicas busca satisfacer el público objetivo?

¿Qué problemas enfrentan que tu página web puede ayudar a resolver?

1.3. Análisis de Competencia

- Identificación de Competidores
- Listar al menos 3-5 competidores directos en el mismo nicho.
- Evaluación de Sitios Web Competidores.
- Diseño visual: ¿Cómo es su estética? ¿Qué colores y tipografías utilizan?
- Funcionalidad: ¿Qué características ofrecen? (formularios, chat en vivo, etc.)
- Contenido: ¿Qué tipo de contenido publican? (blogs, videos, guías)
- Estrategias de marketing: ¿Cómo se promocionan en redes sociales y otros canales?
- Fortalezas y Debilidades:
 - Identificar las fortalezas clave que tienen los competidores.
 - Señalar debilidades o áreas donde puedes destacar (por ejemplo, mejor atención al cliente o contenido más relevante).

1.4. Creación de un Mapa del Sitio

El mapa del sitio es una representación visual o estructurada de las páginas que contendrá tu sitio web.

- Página Principal
- Acerca de Nosotros
- Servicios/Productos
- Servicio/Product 1
- Servicio/Product 2
- Servicio/Product 3
- Blog
- Categoría 1
- Categoría 2
- Artículo Destacado
- Testimonios
- Contacto

- Formulario de Contacto
- Mapa/Ubicación

Este mapa puede expandirse según las necesidades específicas del proyecto. Cada sección debe tener enlaces claros para facilitar la navegación.

Capítulo 2: Diseño y creación de contenido

Realizado por: Samantha Isabela Kuk Hernandez

2.1. Crear un diseño visual atractivo

2.1.1: Definir el objetivo y el público objetivo

- Identificar el propósito de la página web: ¿Qué es lo que se quiere comunicar o vender?
- Definir el público objetivo: ¿Quiénes son los visitantes potenciales de la página web?
- Investigar las preferencias y necesidades del público objetivo: ¿Qué tipo de contenido y diseño les atrae?

2.1.2: Crear un esquema de color

- Seleccionar un esquema de color que se adapte al objetivo y al público objetivo: ¿Qué colores se asocian con la marca o el tema?
- Crear un esquema de color coherente: ¿Cómo se pueden combinar los colores para crear un efecto visual atractivo?
- Considerar la accesibilidad y la legibilidad: ¿Cómo se pueden asegurar de que el texto sea legible sobre el fondo?

2.1.3 Seleccionar tipografías

- Seleccionar tipografías que se adapten al objetivo y al público objetivo: ¿Qué tipo de letra se asocia con la marca o el tema?
- Crear un esquema de tipografías coherente: ¿Cómo se pueden combinar las tipografías para crear un efecto visual atractivo?
- Considerar la legibilidad y la accesibilidad: ¿Cómo se pueden asegurar de que el texto sea legible en diferentes dispositivos y tamaños de pantalla?

2.1.4: Crear un diseño de layout

- Crear un esquema de layout que se adapte al objetivo y al público objetivo: ¿Cómo se pueden organizar los elementos en la página para crear un efecto visual atractivo?
- Considerar la jerarquía y la estructura: ¿Cómo se pueden organizar los elementos para crear una jerarquía clara y una estructura lógica?
- Crear un diseño de layout adaptable: ¿Cómo se pueden asegurar de que el diseño se adapte a diferentes dispositivos y tamaños de pantalla?

2.1.5: Agregar elementos visuales

- Agregar imágenes y gráficos que se adapten al objetivo y al público objetivo*: ¿Qué tipo de imágenes y gráficos se asocian con la marca o el tema?
- Crear un esquema de elementos visuales coherente: ¿Cómo se pueden combinar las imágenes y gráficos para crear un efecto visual atractivo?
- Considerar la carga y el rendimiento: ¿Cómo se pueden asegurar de que las imágenes y gráficos no afecten negativamente la carga y el rendimiento de la página?

2.1.6: Revisar y ajustar

- Revisar el diseño para asegurarse de que se adapte al objetivo y al público objetivo: ¿El diseño es coherente con la marca y el tema?
- Ajustar el diseño según sea necesario: ¿Qué cambios se pueden hacer para mejorar el diseño y hacer que sea más atractivo?
- Crear un diseño que sea adaptable y escalable*: ¿Cómo se pueden asegurar de que el diseño se adapte a diferentes dispositivos y tamaños de pantalla, y que sea escalable para futuras necesidades?

2.2. Desarrollar contenido de calidad

2.2.1 Definir el objetivo y el público objetivo

- Identificar el propósito de la página web: ¿Qué es lo que se quiere comunicar o vender?
- Definir el público objetivo: ¿Quiénes son los visitantes potenciales de la página web?
- Investigar las necesidades y preferencias del público objetivo: ¿Qué tipo de contenido les interesa y les es útil?

2.2.2: Crear un plan de contenido

- Determinar el tipo de contenido: ¿Qué tipo de contenido se necesita? (artículos, videos, imágenes, etc.)
- Crear un calendario de publicación: ¿Cuándo se publicará el contenido?
- Asignar responsabilidades: ¿Quién será responsable de crear y publicar el contenido?

2.2.3: Desarrollar contenido de calidad

- Investigar y recopilar información: ¿Qué información es relevante y útil para el público objetivo?
- Crear contenido original y único: ¿Cómo se puede crear contenido que sea diferente y mejor que el de la competencia?
- Utilizar un lenguaje claro y conciso: ¿Cómo se puede comunicar la información de manera clara y fácil de entender?

2.2.4: Optimizar el contenido para motores de búsqueda

- Investigar palabras clave: ¿Qué palabras clave se relacionan con el contenido y el público objetivo?
- Utilizar palabras clave de manera estratégica: ¿Cómo se pueden utilizar las palabras clave en el título, la descripción y el contenido?
- Crear contenido que sea relevante y útil: ¿Cómo se puede crear contenido que sea relevante y útil para el público objetivo?

2.2 5: Revisar y actualizar el contenido

- Revisar el contenido regularmente: ¿Cómo se puede asegurar de que el contenido sea actual y relevante?

- Actualizar el contenido según sea necesario: ¿Cómo se puede actualizar el contenido para reflejar cambios en la industria o la empresa?
- Eliminar contenido obsoleto o irrelevante: ¿Cómo se puede eliminar contenido que ya no sea útil o relevante?

2.3 Crear un sistema de navegación intuitivo

2.3.1: Planificar la estructura de la página web

- Definir el objetivo y el público objetivo: ¿Qué es lo que se quiere comunicar o vender?
- Crear un mapa de sitio: ¿Cómo se pueden organizar las secciones y subsecciones de la página web?
- Identificar las páginas clave: ¿Cuáles son las páginas más importantes de la página web?

2.3.2: Diseñar la navegación principal

- Crear un menú de navegación: ¿Cómo se pueden organizar las opciones de navegación en un menú?
- Utilizar etiquetas claras y concisas: ¿Cómo se pueden utilizar etiquetas que sean fáciles de entender y que reflejen el contenido de la página?
- Crear un sistema de navegación jerárquico: ¿Cómo se pueden organizar las opciones de navegación en una estructura jerárquica?

2.3.3: Agregar navegación secundaria

- Crear enlaces internos: ¿Cómo se pueden agregar enlaces a otras páginas de la página web?
- Utilizar breadcrumbs: ¿Cómo se pueden utilizar breadcrumbs para mostrar la ruta de navegación actual?
- Crear un sistema de navegación por categorías: ¿Cómo se pueden organizar las opciones de navegación por categorías?

2.3.4: Optimizar la navegación para dispositivos móviles

- Crear un diseño adaptable: ¿Cómo se puede crear un diseño que se adapte a diferentes tamaños de pantalla y dispositivos?
- Utilizar un menú de navegación desplegable: ¿Cómo se puede utilizar un menú de navegación desplegable para ahorrar espacio en la pantalla?
- Crear un sistema de navegación táctil: ¿Cómo se puede crear un sistema de navegación que sea fácil de usar en dispositivos táctiles?

2.3.5: Probar y refinar la navegación

- Probar la navegación en diferentes dispositivos y navegadores: ¿Cómo se puede asegurar de que la navegación funcione correctamente en diferentes dispositivos y navegadores?
- Recopilar retroalimentación de los usuarios: ¿Cómo se puede recopilar retroalimentación de los usuarios para mejorar la navegación?
- Refinar la navegación según sea necesario: ¿Cómo se puede refinar la navegación para mejorar la experiencia del usuario?

2.4. Agregar elementos multimedia y interactivos

2.4.1: Planificar el contenido multimedia y interactivo

- Definir el objetivo del contenido: ¿Qué se quiere lograr con el contenido multimedia y interactivo?
- Identificar el público objetivo: ¿Quién es el público objetivo del contenido?
- Determinar el tipo de contenido: ¿Qué tipo de contenido se quiere agregar? (vídeos, imágenes, audio, juegos, etc.)

2.4.2: Agregar imágenes y gráficos

- Utilizar imágenes de alta calidad: ¿Cómo se pueden utilizar imágenes de alta calidad para mejorar la apariencia de la página web?
- Optimizar las imágenes para la web: ¿Cómo se pueden optimizar las imágenes para reducir el tamaño de archivo y mejorar la carga de la página?

- Agregar efectos visuales y animaciones: ¿Cómo se pueden agregar efectos visuales y animaciones para hacer que las imágenes sean más atractivas?

2.4.3: Agregar contenido de video

- Crear videos de alta calidad: ¿Cómo se pueden crear videos de alta calidad para agregar a la página web?
- Utilizar plataformas de video en línea: ¿Cómo se pueden utilizar plataformas de video en línea como YouTube o Vimeo para agregar videos a la página web?
- Agregar subtítulos y closed captions: ¿Cómo se pueden agregar subtítulos y closed captions para hacer que los videos sean más accesibles?

2.4.4: Agregar contenido de audio

- Crear archivos de audio de alta calidad: ¿Cómo se pueden crear archivos de audio de alta calidad para agregar a la página web?
- Utilizar plataformas de audio en línea: ¿Cómo se pueden utilizar plataformas de audio en línea como SoundCloud o Spotify para agregar audio a la página web?
- Agregar controles de reproducción: ¿Cómo se pueden agregar controles de reproducción para hacer que el audio sea más interactivo?

2.4.5: Agregar elementos interactivos

- Crear formularios y encuestas: ¿Cómo se pueden crear formularios y encuestas para recopilar información de los visitantes?
- Utilizar herramientas de interacción en línea: ¿Cómo se pueden utilizar herramientas de interacción en línea como chatbots o webinars para agregar interactividad a la página web?

Capítulo 3. Diseño de la Interfaz

Realizado por: Ailed Guadalupe Che Chi

Los principios del diseño visual son las reglas y guías que ayudan a crear una composición visual atractiva y efectiva.

3.1. Principios del Diseño Visual

- **Equilibrio:** La distribución de elementos visuales para crear una sensación de estabilidad y armonía.
- **Contraste:** La utilización de diferentes colores, texturas y tamaños para crear interés visual y destacar elementos importantes.
- **Jerarquía:** La organización de elementos visuales para crear una estructura clara y fácil de seguir.
- **Ritmo:** La creación de un patrón visual que guía al usuario a través de la página web.
- **Unidad:** La creación de una sensación de cohesión y armonía a través de la utilización de elementos visuales consistentes.

3.2 Selección de Colores y Tipografía

La selección de colores y tipografía es fundamental para crear una página web atractiva y efectiva.

- **Colores:** Selecciona colores que se ajusten a la marca y la personalidad de tu página web. Utiliza herramientas como la rueda de colores para seleccionar colores que se complementen entre sí.
- **Tipografía:** Selecciona fuentes que sean claras, legibles y consistentes con la marca y la personalidad de tu página web. Utiliza herramientas como Google Fonts para seleccionar fuentes gratuitas y de alta calidad.

3.3 Diseño Responsivo y Adaptativo

El diseño responsivo y adaptativo es fundamental para crear una página web que se vea bien en diferentes dispositivos y tamaños de pantalla.

Diseño móvil primero: Diseña tu página web para dispositivos móviles primero, y luego ajusta el diseño para dispositivos de escritorio.

- Utiliza media queries: Utiliza media queries para ajustar el diseño de tu página web según el tamaño de la pantalla y el dispositivo.
- Utiliza unidades relativas: Utiliza unidades relativas como porcentajes y ems para ajustar el tamaño de los elementos según el tamaño de la pantalla.

3.4 Prototipos y Wireframes

Los prototipos y wireframes son herramientas fundamentales para diseñar y planificar una página web.

Utiliza herramientas de diseño: Utiliza herramientas de diseño como Sketch, Figma o Adobe XD para crear prototipos y wireframes.

Crea un esquema básico: Crea un esquema básico de la estructura y el diseño de tu página web.

Ajusta y refina: Ajusta y refina tu prototipo y wireframe según sea necesario para asegurarte de que se ajusten a tus necesidades y objetivos.

Capítulo 4.-Desarrollo del sitio web

Realizado por: Yadira Jazmín Cocom Poot

4.1.-Desarrollo Front-end: Utilice HTML, Css JavaScript para crear la interfaz visible del sitio.

- HTML (Lenguaje de marcado de hipertexto): es el lenguaje de marcado que estructura el contenido de la página web, definiendo elementos como encabezados, párrafos, enlaces e imágenes.
- CSS (Cascading Style Sheets): Se utiliza para estilizar la presentación del contenido HTML, controlando aspectos como colores, tipografías, márgenes y diseño general.
- JavaScript: Este lenguaje de programación permite añadir interactividad a las páginas web, manipulando el DOM (Document Object Model) para responder a acciones del usuario, como clics o desplazamientos.

La estructura del lenguaje HTML para poder desarrollar una pagina web es la que se muestra a continuación:

4.1.1.-Estructura Básica del Lenguaje HTML

```
<!DOCTYPE html>

<html lang="es">

<head>

  <meta charset="UTF-8">

  <meta name="viewport" content="width=device-width, initial-
scale=1.0">

  <title>Mi Página Web</title>

  <link rel="stylesheet" href="styles.css">

</head>

<body>

  <header>

    <h1>Bienvenido a Mi Página Web</h1>
```

```

<nav>
  <ul>
    <li><a href="#about">Sobre mí</a></li>
    <li><a href="#services">Servicios</a></li>
    <li><a href="#contact">Contacto</a></li>
  </ul>
</nav>
</header>

<main>
  <section id="about">
    <h2>Sobre mí</h2>
    <p>Descripción sobre ti o tu negocio.</p>
  </section>

  <section id="services">
    <h2>Servicios</h2>
    <p>Descripción de los servicios ofrecidos.</p>
  </section>

  <section id="contact">
    <h2>Contacto</h2>
    <form action="#" method="post">
      <label for="name">Nombre:</label>
      <input type="text" id="name" name="name" required>
    </form>
  </section>

```

```

        <label for="email">Correo Electrónico:</label>
        <input type="email" id="email" name="email" required>

        <input type="submit" value="Enviar">
    </form>
</section>
</main>

<footer>
    <p>&copy; 2025 Mi Página Web. Todos los derechos
    reservados.</p>
</footer>

<script src="script.js"></script>
</body>
</html>

```

4.1.2.Aplicacion de Estilo CSS

4.1.2.1 Estilo en línea

Este método implica agregar CSS directamente en el elemento HTML utilizando el atributo style. Aunque es sencillo, no se recomienda su uso extensivo debido a problemas de mantenimiento y SEO.

Ejemplo:

```
xml
```

```
<p style="color: green; font-size: 20px;">Este es un párrafo con
estilo en línea.</p>
```

4.1.2.2. CSS interno (incrustado)

El CSS interno se coloca dentro de una etiqueta <style>en la sección <head>del documento HTML. Este método es útil para aplicar estilos en una sola página.

Ejemplo:

```
xml
<!DOCTYPE html>
<html lang="es">
<head>
  <meta charset="UTF-8">
  <title>Ejemplo de CSS Interno</title>
  <style>
    body {
      background-color: lightblue;
    }
    h1 {
      color: red;
      padding: 20px;
    }
  </style>
</head>
<body>
  <h1>Bienvenido a mi página web</h1>
  <p>Este es un párrafo de ejemplo.</p>
</body>
</html>
```

4.1.2.3. CSS externo

El método más recomendado es utilizar un archivo CSS externo. Esto implica crear un archivo separado con extensión .css, que luego se vincula al documento HTML mediante una etiqueta <link> en la sección <head>. Este enfoque promueve la separación de preocupaciones y facilita el mantenimiento.

Ejemplo:

1. Archivo CSS (styles.css):

```
css

body {
    background-color: lightgray;
}

h1 {
    color: blue;
    padding: 15px;
}
```

2. Archivo HTML (index.html):

```
xml

<!DOCTYPE html>
<html lang="es">
<head>
    <meta charset="UTF-8">
    <title>Ejemplo de CSS Externo</title>
    <link rel="stylesheet" href="styles.css">
</head>
<body>
    <h1>Bienvenido a mi página web</h1>
```

```
<p>Este es un párrafo de ejemplo. </p>
</body>
</html>
```

4.1.3.-Interactividad con JavaScript

Puedes agregar interactividad usando JavaScript. Aquí hay un ejemplo simple:

Javascript

```
document. addEventListener ('DOMContentLoaded', function () {
    const form = document. query Selector('form');

    form.addEventListener('submit', function(event) {
        event. preventDefault (); // Evita que se envíe el formulario
        alert('Formulario enviado con éxito');
    });
});
```

4.2Desarrollo Back-end: Implementa funcionalidades del servidor y bases de datos necesarias para gestionar datos e interacciones.

El desarrollo back-end se encarga de la lógica, el almacenamiento y la gestión de datos en una aplicación web. Es responsable de manejar la comunicación entre el servidor y el cliente, procesando solicitudes y enviando respuestas adecuadas. Esto incluye:

- **Gestión de bases de datos** : Almacenar y recuperar datos.
- **Autenticación de usuarios** : Validar credenciales y gestionar sesiones.
- **Procesamiento de formularios** : Recibir y manejar datos enviados por los usuarios.

4.2.1. Lenguajes de Programación Comunes en el Back-End

Los desarrolladores back-end utilizan varios lenguajes de programación para crear aplicaciones web, entre los más populares se encuentran:

- **Python:** Usado con frameworks como Django y Flask.
- **JavaScript (Node.js):** permite ejecutar JavaScript en el servidor.
- **PHP:** Muy utilizado para desarrollo web, especialmente en sistemas como WordPress.
- **Ruby:** Usado con Ruby on Rails.
- **Java:** Común en aplicaciones empresariales.

4.3. Integración de características clave: Agregue formularios de contacto, herramientas de análisis y otros elementos interactivos según sea necesario.

Ejemplo:

- **Imágenes:**

xml

```

```

- **Vídeos :**

xml

```
<video controls>
```

```
  <source src="video.mp4" type="video/mp4">
```

Tu navegador no soporta el video.

```
</video>
```

Capítulo 5. Pruebas y Lanzamiento

Realizado por: Yamilex Ku Chable

5.1. Pruebas

- Verificar la funcionalidad de todos los elementos: Asegúrate de que todos los botones, enlaces y formularios funcionen correctamente. Realice pruebas exhaustivas en cada página del sitio para identificar y corregir errores.
- Pruebas de navegación: Confirma que la navegación sea intuitiva y que los usuarios puedan acceder a todas las secciones sin problemas.

5.1.1. Pruebas de compatibilidad

- Revisar en diferentes navegadores: Asegúrese de que el sitio se vea y funcione bien en los principales navegadores (Chrome, Firefox, Safari, Edge).
- Pruebas en dispositivos móviles: Verifica que el diseño sea responsivo y se adapte adecuadamente a diferentes tamaños de pantalla.

5.1.2.- Pruebas de Velocidad y Rendimiento

- Medir la velocidad de carga: Utilice herramientas como Google PageSpeed Insights para evaluar el tiempo de carga del sitio y obtener recomendaciones para mejorar el rendimiento.
- Optimizar recursos : Minimiza archivos CSS y JavaScript, y optimiza imágenes para reducir el tiempo de carga.

5.2. Lanzamiento y Monitoreo

- Preparar el lanzamiento : Una vez completadas las pruebas, asegúrese de que todo esté documentado y centralizado. Desactiva herramientas de depuración y realiza una última revisión general.
- Anuncio del lanzamiento : Comunica el lanzamiento a través de tus canales de marketing (redes sociales, correo electrónico).

- Monitoreo post-lanzamiento: Después del lanzamiento, supervise el rendimiento del sitio utilizando herramientas analíticas. Esté atento a cualquier problema que pueda surgir y esté preparado para realizar ajustes rápidamente.

En conclusión, crear una página web efectiva y atractiva implica definir objetivos claros desde principios hasta la implementación de diseños, desarrollo y estilos con atención a detalle cada a paso del camino a un sitio funcional visualmente atractivo.

Siguiendo los pasos de este manual podrás crear una web que te atraiga o que requieras.