

UNIVERSIDAD DEL VALLE



INGENIERIA DE SISTEMAS INFORMATICOS

MANUAL DE USUARIO

ESTUDIANTE:

- VICHINI FERNÁNDEZ MIJHAIL ALVARO

DOCENTE: ING. MIRANDA ORDONEZ HENRY

MATERIA: PROGRAMACION WEB I

GRUPO: C

TEMA: PAGINA WEB

LA PAZ-BOLIVIA

2025

Tabla de contenido

1. Introducción.....	3
2. Objetivo del Proyecto.....	4
3. Tecnologías Utilizadas	4
4. Estructura del Proyecto.....	5
5. Explicación Detallada de Cada Módulo.....	6
5.1 Página Principal (index.html).....	6
5.2 Módulo de Menú (menu.html)	6
5.3 Módulo de Novedades (novedades.html).....	7
5.4 Módulo de Compra (comprar.html).....	7
6. JavaScript – Lógica del Sistema.....	7
6.1 Carga de Productos.....	7
6.2 Parámetros de la URL	7
6.3 Control de Cantidad.....	8
6.4 Modales.....	8
6.5 Validación Básica.....	8
7. Estilos CSS – Diseño y Responsividad	8
8. Simulación del Pago.....	9
9. Pruebas Realizadas	9
10. Conclusiones del Proyecto.....	10

MANUAL TÉCNICO DEL PROYECTO WEB – RESTAURANTE

1. Introducción

Este manual técnico fue elaborado para describir de forma detallada y entendible el funcionamiento interno del proyecto web de un restaurante desarrollado como parte de la materia de Programación Web. El propósito principal es explicar cómo está construido el sitio, qué tecnologías utiliza y cómo se organiza tanto a nivel visual como funcional.

Este documento está escrito desde la perspectiva de un estudiante que realizó el proyecto con las herramientas vistas en clase, por lo que el lenguaje es claro, directo y enfocado en el aprendizaje. También se incluyen explicaciones de por qué se eligió cada estructura y qué problemas se intentaron resolver con el diseño actual.

El proyecto consiste en un sitio web que muestra un catálogo de comida (hamburguesas, tacos y desayunos), un módulo de menú donde se puede filtrar por categorías, una sección de novedades y una página individual de compra donde se simula el pago por tarjeta. Todo el proyecto está hecho únicamente con HTML, CSS y JavaScript, sin backend ni bases de datos reales.

2. Objetivo del Proyecto

El objetivo principal de este proyecto fue crear una página web funcional, visualmente atractiva y fácil de usar, que simule el sistema de un restaurante donde los usuarios puedan navegar entre productos, ver detalles y realizar una compra simulada.

Objetivos específicos:

- Aplicar los conocimientos de HTML, CSS y JavaScript aprendidos durante el semestre.
- Implementar un diseño responsive que se adapte a distintas pantallas como celulares y computadoras.
- Simular un flujo de compra entendible para el usuario.
- Practicar el manejo de eventos, dinámicas con el DOM, modales, y URL con parámetros.
- Organizar el código en archivos separados para mantener el orden del proyecto.

3. Tecnologías Utilizadas

El proyecto se desarrolló usando únicamente tecnologías del lado del cliente, ya que la asignatura se enfoca en Programación Web Frontend.

Tecnologías empleadas:

- HTML5: para la estructura principal de cada página.
- CSS3: para todos los estilos visuales, animaciones, modales y diseño responsive.
- JavaScript Vanilla: para manejar la lógica del menú, cargar productos dinámicamente, abrir modales, validar datos y mostrar el total a pagar.

- Google Fonts (Poppins): se usó para mejorar la estética del texto.
- SwiperJS: utilizado para crear el slider principal de portada.

No se usó PHP, bases de datos ni frameworks avanzados, ya que no formaban parte del requisito del proyecto.

4. Estructura del Proyecto

La estructura del proyecto está organizada en carpetas para mantener ordenados los estilos, scripts y recursos del sitio. De esta forma, cualquier persona que abra la carpeta puede entender rápidamente dónde se encuentra cada parte del sistema.

Estructura de carpetas:

css/

estilos.css

compra.css

menu.css

novedades.css

js/

compra.js

menu.js

images/

(contiene las imágenes de los productos)

index.html

menu.html

novedades.html

comprar.html

- Los archivos HTML representan cada sección del sitio.
- Los CSS están divididos según la página para facilitar la edición.
- Los JS se encargan de la lógica correspondiente a su módulo.

5. Explicación Detallada de Cada Módulo

Aquí se explica cómo funciona cada una de las partes del proyecto.

5.1 Página Principal (index.html)

Esta página sirve como la portada del restaurante. Se incluyó un slider principal para mostrar productos destacados y captar la atención del usuario desde el inicio. Debajo del slider hay categorías de productos que cambian dinámicamente según la pestaña seleccionada.

5.2 Módulo de Menú (menu.html)

Este módulo muestra todas las comidas del restaurante. Lo importante aquí es que se utilizan botones para filtrar entre hamburguesas, tacos y desayunos. Al presionar un botón, el archivo menu.js genera dinámicamente las tarjetas de los productos sin recargar la página.

5.3 Módulo de Novedades (novedades.html)

Es una página donde se muestran productos nuevos o promocionales. Se usa una cuadrícula CSS para mostrar las tarjetas de forma limpia y ordenada.

5.4 Módulo de Compra (comprar.html)

Esta página recibe datos desde el menú mediante parámetros en la URL. Por ejemplo:

`comprar.html?name=Combo1&price=20&img=foto.jpg`

Luego el archivo compra.js extrae esos valores y los muestra en la página. También se puede seleccionar la cantidad y se actualiza automáticamente el precio. Al presionar “Comprar”, aparece un modal de pago que simula el ingreso de tarjeta.

6. JavaScript – Lógica del Sistema

El JavaScript es fundamental para que el sitio se sienta dinámico.

6.1 Carga de Productos

El archivo menu.js contiene arreglos con la información de cada categoría. Cuando el usuario elige una categoría, se genera el HTML desde cero usando JS, sin necesidad de escribir manualmente todas las tarjetas.

6.2 Parámetros de la URL

El archivo compra.js utiliza URLSearchParams para obtener nombre, descripción, imagen y precio del producto seleccionado.

6.3 Control de Cantidad

Se programaron botones '+' y '-' que cambian el número de productos. Cada vez que cambia la cantidad, el precio se recalcula.

6.4 Modales

Los modales del pago funcionan mostrando y ocultando contenedores con display: block y display: none. Esto permite una simulación simple sin recargar la página.

6.5 Validación Básica

Aunque no es un pago real, se valida que el número de tarjeta tenga al menos 16 dígitos y que el CVV tenga 3, simulando un proceso real.

7. Estilos CSS – Diseño y Responsividad

Para que el proyecto sea atractivo visualmente, se creó un diseño moderno usando colores oscuros y rojos relacionados con comida rápida. Se añadió padding, márgenes, sombras, bordes redondeados y animaciones.

Además, se implementaron media queries para pantallas pequeñas. En celulares:

- El menú se pliega en estilo hamburguesa.
- Los grids se vuelven columnas de 1 elemento.
- Las imágenes se reducen para evitar desplazamientos innecesarios.

El diseño se probó en varios tamaños de pantalla para asegurar que fuera usable en PC, tablet y celular.

8. Simulación del Pago

El proyecto no usa servidores reales ni pasarelas de pago, por lo que la parte de pago es completamente simulada. Sin embargo, se implementó un modal donde el usuario puede escribir sus datos como si fuera una compra real.

Datos simulados:

- Número de tarjeta
- Fecha de expiración
- CVV

Si la información está incompleta, se muestra un mensaje. Al completarse, aparece otro modal indicando "Gracias por su compra". Esto cumple la intención de simular un proceso real sin poner datos verdaderos ni conectarse a nada peligroso.

9. Pruebas Realizadas

Se hicieron varias pruebas manuales durante el desarrollo:

- Probar que los botones del menú cambian correctamente la categoría.
- Revisar que los productos carguen con su imagen, nombre y precio.
- Verificar que los parámetros de la URL se pasen de manera correcta.
- Aumentar y disminuir cantidades y validar que el precio cambie.
- Probar los modales para asegurarse de que abran y cierren.
- Revisar la responsividad desde un teléfono.

— Validar datos en la simulación de pago.

Todo funcionó correctamente según los objetivos estudiantiles planteados.

10. Conclusiones del Proyecto

El proyecto permitió poner en práctica todo lo aprendido durante la materia de Programación Web. Se logró construir una página web completa, con varias secciones, estilos modernos y funcionalidades interactivas realizadas con JavaScript.

También fue una buena experiencia para entender la importancia de dividir el código en archivos separados, escribir funciones reutilizables y pensar en la experiencia del usuario. Aunque el sistema no incluye backend real, cumple perfectamente con el alcance de un estudiante y demuestra conocimientos sólidos de desarrollo web frontend.

Como mejoras futuras, se podría añadir:

- Registro de usuarios
- Sistema real de pedidos
- Base de datos
- Carrito de compras
- Integración con APIs de pago