

Componente Frontend del Proyecto Formativo y Proyectos de Clase (listas de chequeo)

Evidencia: GA7-220501096-AA4-EV03

Formación: Tecnólogo en Análisis y Desarrollo de Software

Aprendiz: Jefferson Harbey Mendez Castellanos

Instructor: John Alejandro Niño Tambo

Ficha: 2977395

Fecha: 11 – 11 - 2025

Tabla de contenido

Introducción	3
Objetivo.....	3
Introducción al proyecto	3
Descripción técnica de los cambios realizados	5
Barra de navegación izquierda (sidebar).....	5
Contenedor de pedidos en menú	6
Mejoras visuales a la interfaz inventario	7
Catálogo de bebidas	8
Componentes reutilizables	8
Mejoras en feedback implementadas	10
Diseño responsive mejorado	11
Interfaces dinámicas agregadas.....	14
Uso de Git para control de versiones	15
Otras mejoras Frontend implementadas o corregidas	16
Conclusiones	17

Introducción

La estructura frontend es clave para cualquier proyecto de software, es por eso, que en esta evidencia se va a documentar el trabajo que se viene realizando en el frontend de nuestro proyecto, el “Software de Gestión Administrativa Matrona”, el cual presenta mejoras en rendimiento, usabilidad, experiencia del usuario, accesibilidad a todas sus funciones, y adaptabilidad a muchos tipos de pantallas con un diseño responsive. Además, se ha logrado la optimización en el código con la implementación de JINJA2, tecnología usada para la creación de componentes reutilizables la cual hace una combinación perfecta con FastAPI para reutilizar código dinámicamente en cualquier plantilla de nuestro frontend. Adicionalmente se busca consolidar conocimientos en el uso de git, para control de versiones, trabajando en ramas secundarias y manejo de commits. Al final de este trabajo se espera una estructura frontend del proyecto matrona moderna y profesional.

Objetivo

Mejorar el estilo visual del sistema, manteniendo coherencia con colores, estilos en componentes. Mejorar el diseño responsive para que todas las interfaces sean adaptables a múltiples pantallas sin presentar errores en sus componentes internos como tarjetas, tablas etc. Implementar JINJA2 para crear componentes reutilizables, optimizando y eliminando la repetición de múltiples líneas de código. Utilizar Git y GitHub para el control de versiones, creando ramas secundarias para trabajar en ellas varias funcionalidades sin tocar la rama principal y verificar la consistencia de enlaces de navegación.

Introducción al proyecto

El software de Gestión Administrativa Matrona tiene como objetivo la optimización y automatización de las labores administrativas de la empresa Matrona, el sistema es diseñado con

una arquitectura modular, y de acuerdo a los requerimientos del cliente, presenta las siguientes funcionalidades:

- Usuarios: registro de usuarios con sus datos, autenticación para uso del sistema, y roles administrador/cliente/empleado.
- Modulo para Inventario: control de cervezas producidas, permite agregar, sumar al stock, editar, eliminar y ver detalles de cada cerveza.
- Modulo materiales: permite control de existencia de materiales para producción y envasado de cerveza; guarda, edita, y elimina materiales.
- Modulo contabilidad: información contable con datos de ventas por fecha, cliente, tipo de cerveza, y cantidades vendidas, información del total de presupuesto y grafico de ventas para análisis de datos.
- Proveedores: información de los proveedores de materia prima, facilitando el acceso a datos de contacto.
- Catalogo de bebidas: el cliente reserva su pedido de manera anticipada, sea por unidad, por sixpack o por cajas, el catálogo muestra la información de cada bebida completa.
- Pedidos: el sistema procesa pedidos y los muestra en el panel principal, en donde el administrador confirmará cada vez que se entregue un pedido para posteriormente ser agregado a contabilidad actualizando los datos de las ventas.
- Empleado: el sistema permite funciones específicas para empleados, como materiales, inventario, y proveedores.

- Adicionalmente presenta seguridad con autenticación con JWT y encriptación de contraseñas con bcrypt lo cual lo hace un sistema seguro y confiable.

Descripción técnica de los cambios realizados

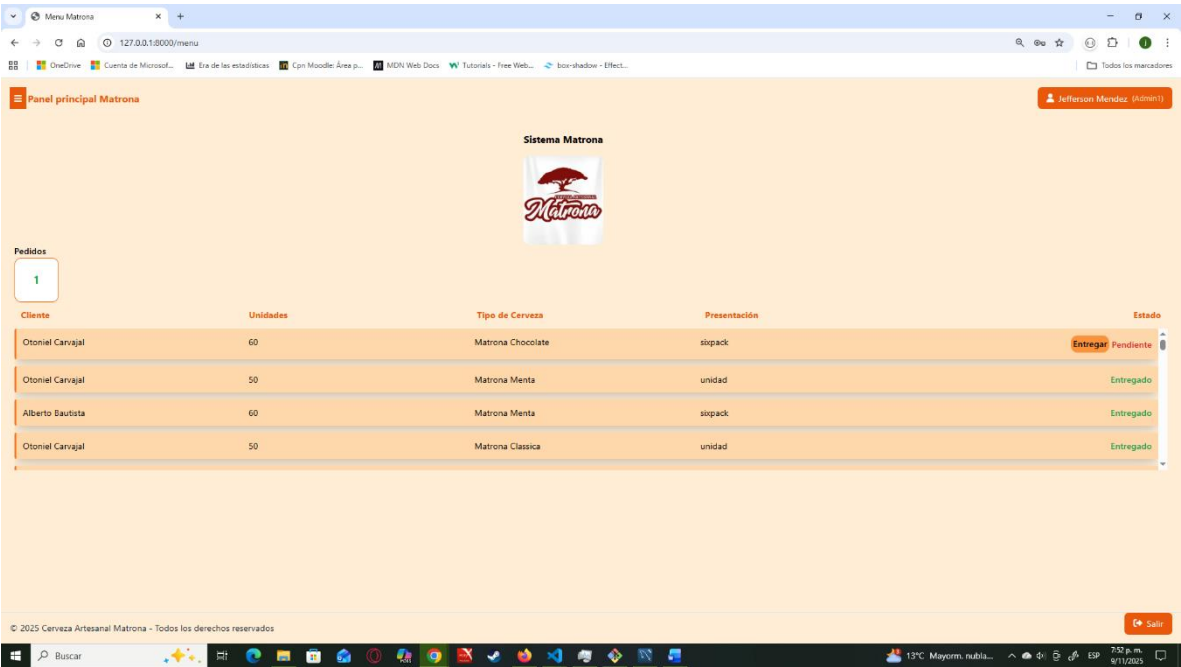
Barra de navegación izquierda (sidebar)

Se realizó un trabajo especial a la barra lateral, logrando ocultarla al presionar cualquier lugar vacío de la pantalla, sin romper la estructura de los demás componentes frontend, y se añadió un botón menú hamburguesa que al detectar un evento click, vuelve a activar la barra de funciones, logrando de esta manera un estilo moderno y optimizando el espacio lo que beneficia enormemente la usabilidad en pantallas pequeñas.

Antes

Panel principal Matrona				
<div>Sistema Matrona</div>				
Pedidos				
1				
Cliente	Unidades	Tipo de Cerveza	Presentación	Estado
Agustin Gelvez	6	Matrona Frutas	sixpack	Entregar Pendiente
Agustin Gelvez	10	Matrona Frutos Rojos	unidad	Entregado
Agustin Gelvez	36	Chocolate 3	unidad	Entregado
Otoniel Carvajal	2	Chocolate 3	unidad	Entregado
Otoniel Carvajal	6	Chocolate 3	sixpack	Entregado
Otoniel Carvajal	1	Matrona Clasica	unidad	Entregado
Agustin Gelvez	0	Sin nombre	N/A	Entregado
Otoniel Carvajal	0	Sin nombre	N/A	Entregado

Ahora



Contenedor de pedidos en menú

Se limito el alto del contenedor que muestra la lista de pedidos a un máximo de 256 px, se añadió overflow para navegar con facilidad por la lista de pedidos.

Antes



Ahora

Pedidos

1

Cliente	Unidades	Tipo de Cerveza	Presentación	Estado
Otoniel Carvajal	60	Matrona Chocolate	sipack	Entregar Pendiente
Otoniel Carvajal	50	Matrona Menta	unidad	Entregado
Alberto Bautista	60	Matrona Menta	sipack	Entregado
Otoniel Carvajal	50	Matrona Clasica	unidad	Entregado

© 2025 Cerveza Artesanal Matrona - Todos los derechos reservados

Salir

Mejoras visuales a la interfaz inventario

Se eliminaron elementos obsoletos, y se realizó mejora visual al componente que muestra detalles de cada cerveza en el inventario.

Antes

Menu Principal

Inventario

Materiales

Contabilidad

Catálogo de Productos

Cotizaciones

Empleados

Clientes

Proveedores

Panel de Administración Inventario

Gestión de Inventario

Stock de Cervezas

Agregar Nueva Cerveza

Ultimo Movimiento

Matrona Clasica	1441 Unidades	Modificar	+ Detalles	+ Agregar stock	Eliminar
Matrona Cafe	1000 Unidades	Modificar	+ Detalles	+ Agregar stock	Eliminar
Matrona Durazno	2300 Unidades	Modificar	+ Detalles	+ Agregar stock	Eliminar
Matrona Frutos Rojos	840 Unidades	Modificar	+ Detalles	+ Agregar stock	Eliminar
Matrona Ciruela	835 Unidades	Modificar	+ Detalles	+ Agregar stock	Eliminar
Matrona Chocolate	100 Unidades	Modificar	+ Detalles	+ Agregar stock	Eliminar
Matrona Menta	166 Unidades	Modificar	+ Detalles	+ Agregar stock	Eliminar

Detalles Matrona Clásica

300 Unidades disponibles

5.5 Vol Alcohol

330 Ml de Contenido

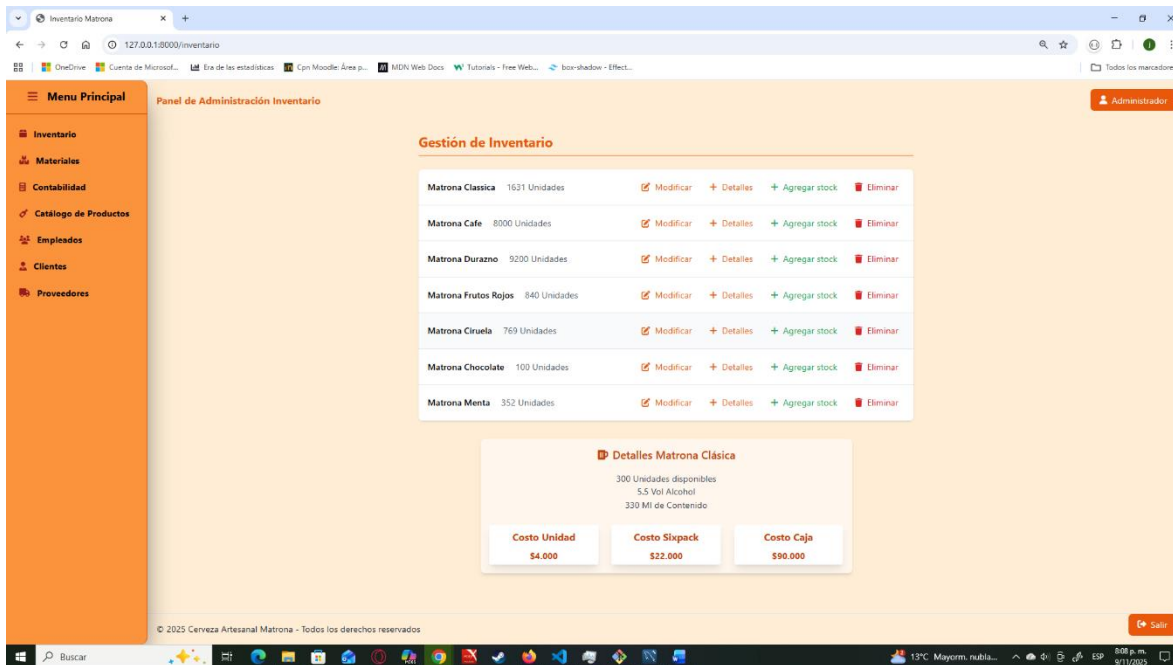
Costo Unidad

Costo Sipack

Costo Caja

Salir

Ahora



Catálogo de bebidas

Se mejoro el estilo visual para que sea más consistente con el estilo general del proyecto, mejorando el color cabecera y pie de pagina

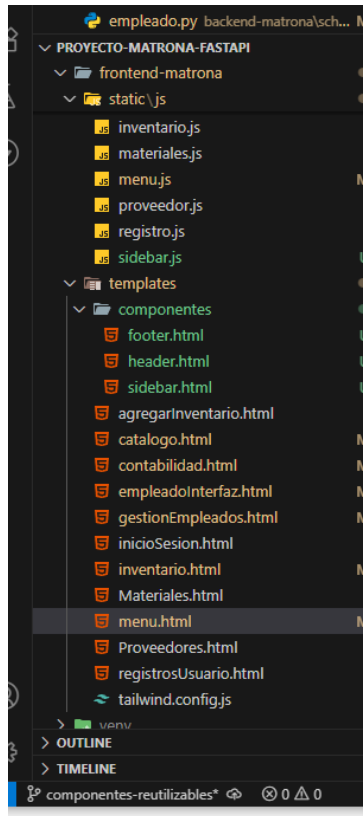
Componentes reutilizables

Se realizo la implementación de Jinja2, el cual es un motor de plantillas para Python que permite crear dinámicamente texto, como HTML, código o mensajes, utilizando plantillas con sintaxis similar a Python, en nuestro proyecto se ha empezado a implementar en la construcción de componentes reutilizables, clave para optimizar código repetido. A continuación, se presenta la descripción de los componentes que han sido separados de las plantillas para ser reutilizados con jinja2 en cualquier lugar de nuestra frontend.

aside.html(barra de navegación): elemento principal del sistema, contiene todas las funcionalidades y esta presenta en la mayoría de las interfaces, ha sido separado contribuyendo al

diseño modular y posteriormente incluido en cada plantilla html, esto evita muchísimas líneas de código logrando así un sistema mantenible y escalable.

estructura de archivos

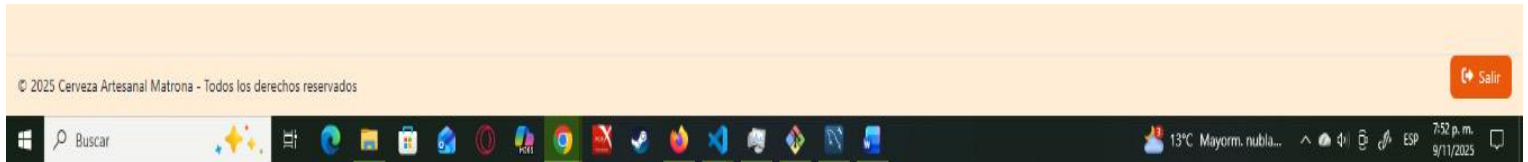


inserción en las plantillas con include

```
frontend-matrona > templates > menu.html > html > body.for
2  <html lang="es">
3  <head>
4
5
6  <title>Menu Matrona</title>
7  <script src="https://cdn.tailwindcss.com"></script>
8  <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/@fortawesome/fontawesome-free@5.15.1/css/all.min.css">
9  </head>
10 <body class="font-sans flex h-screen">
11   <button id="verSidebar" class="hidden p-2">
12     <i class="fas fa-bars"></i>
13   </button>
14   <!-- Barra Lateral -->
15   {%include 'componentes/sidebar.html'%}
16
17   <div class="flex-1 flex flex-col">
18     <!-- Cabeza de página -->
19     <header class="bg-orange-100">
20       <div class="flex justify-between items-center">
21         <h2 class="text-lg font-bold">
22       <div class="flex items-center space-between">
23         <button class="btn-admin flex items-center">
24           <i class="fas fa-user"></i>
25           <span id="nombreUsuario" class="text-sm">
26           <span id="rolUsuario" class="text-sm">
```

header(cabeza de página): componente clave y repetitivo en la mayoría de las interfaces de la misma manera que la barra de navegación se ha aislado en un componente reutilizable

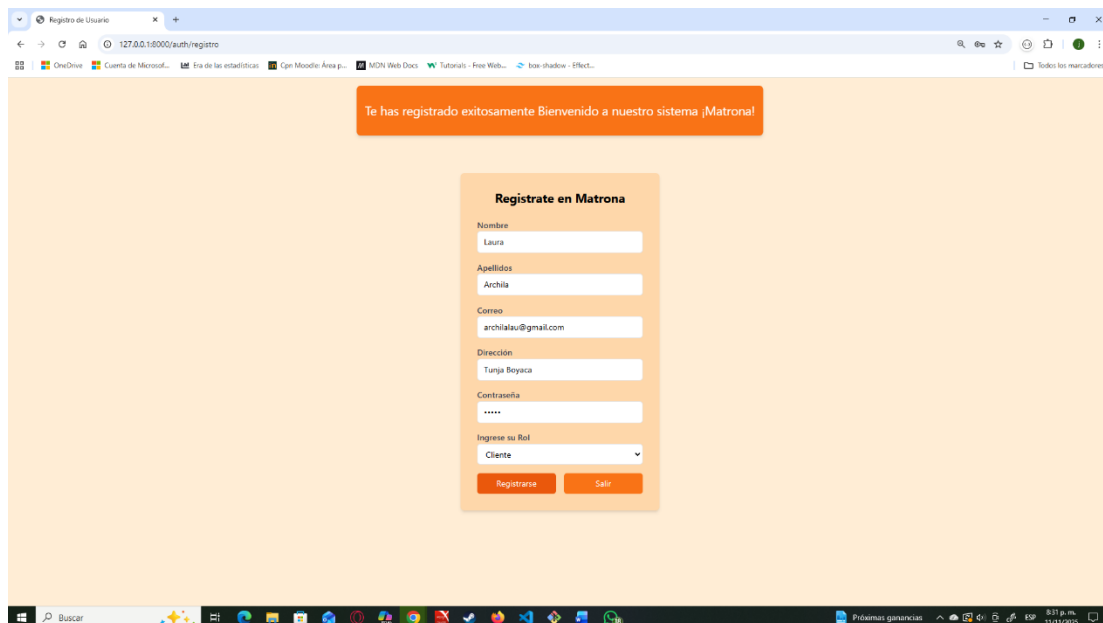
footer(pie de página): componente reutilizable el cual incorpora botón funcional que cierra sesión en cualquier interfaz, y adicionalmente incluye el copyright de la empresa.



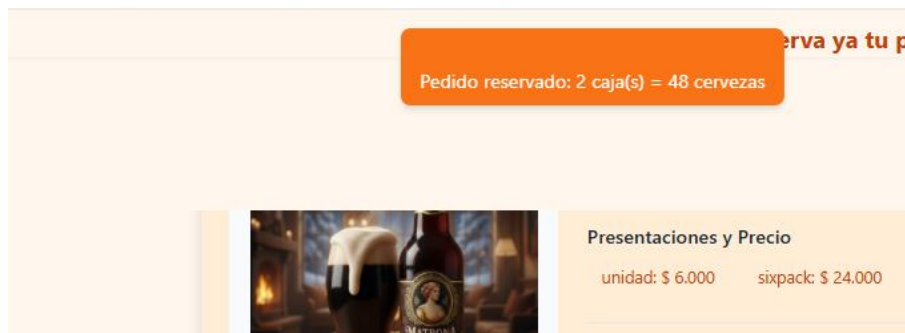
Mejoras en feedback implementadas

Se han implementado mensajes de alerta o de ayuda al usuario empleando estilos del base del proyecto como se muestra a continuación

Mensaje de bienvenida al registrarse el usuario



Mensaje de confirmación de pedido



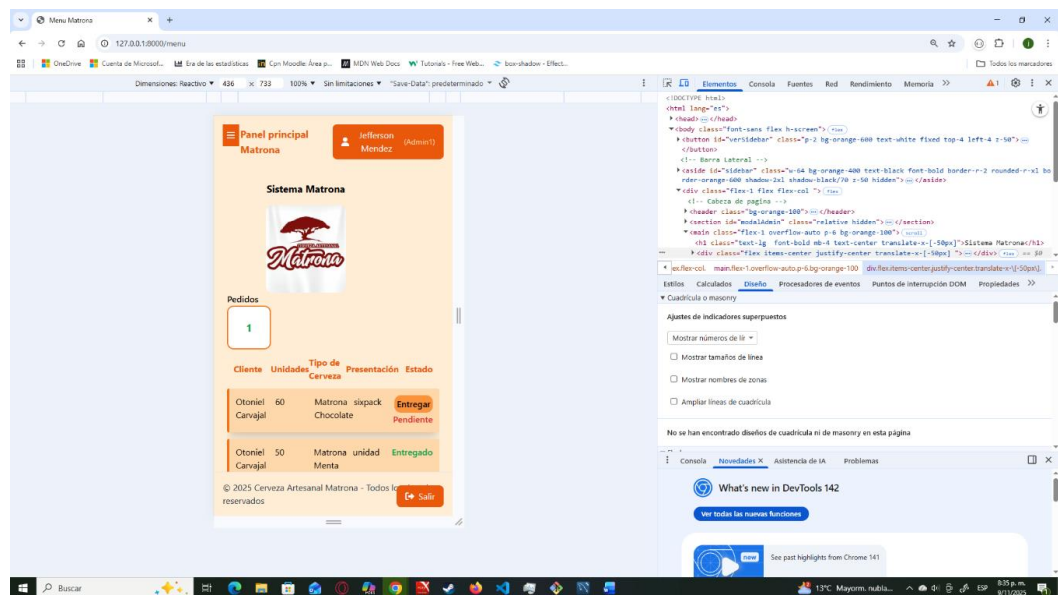
Diseño responsive mejorado

A continuación se presentan las mejoras responsive implementadas para lograr que el sistema sea adaptable a pantallas pequeñas incluso móviles.

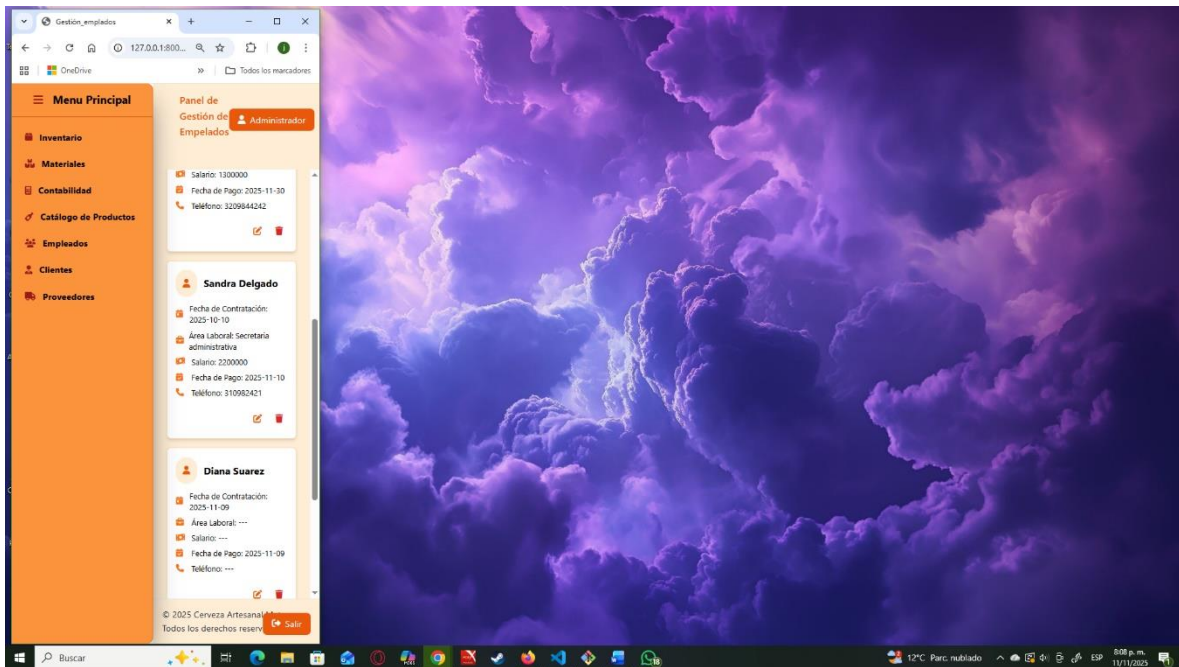
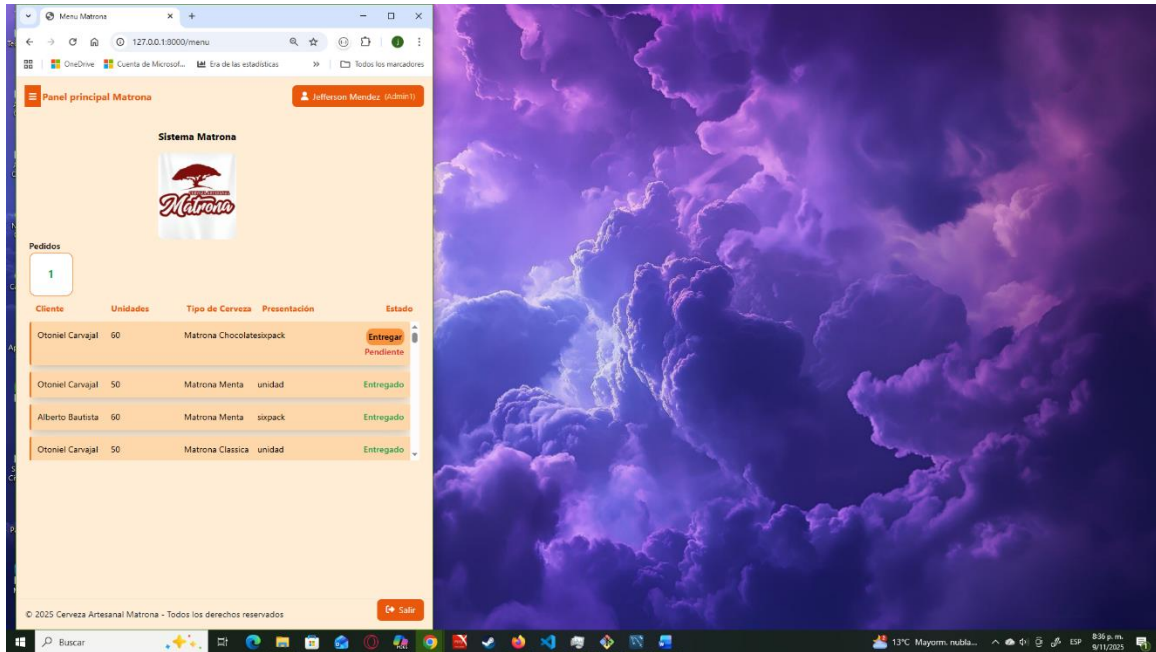
Diseño general

Barra lateral colapsable generando espacio y facilitando la adaptabilidad a pantallas muy pequeñas, como se ve a continuación al probar con las herramientas para dispositivo de desarrolladores de Chrome.

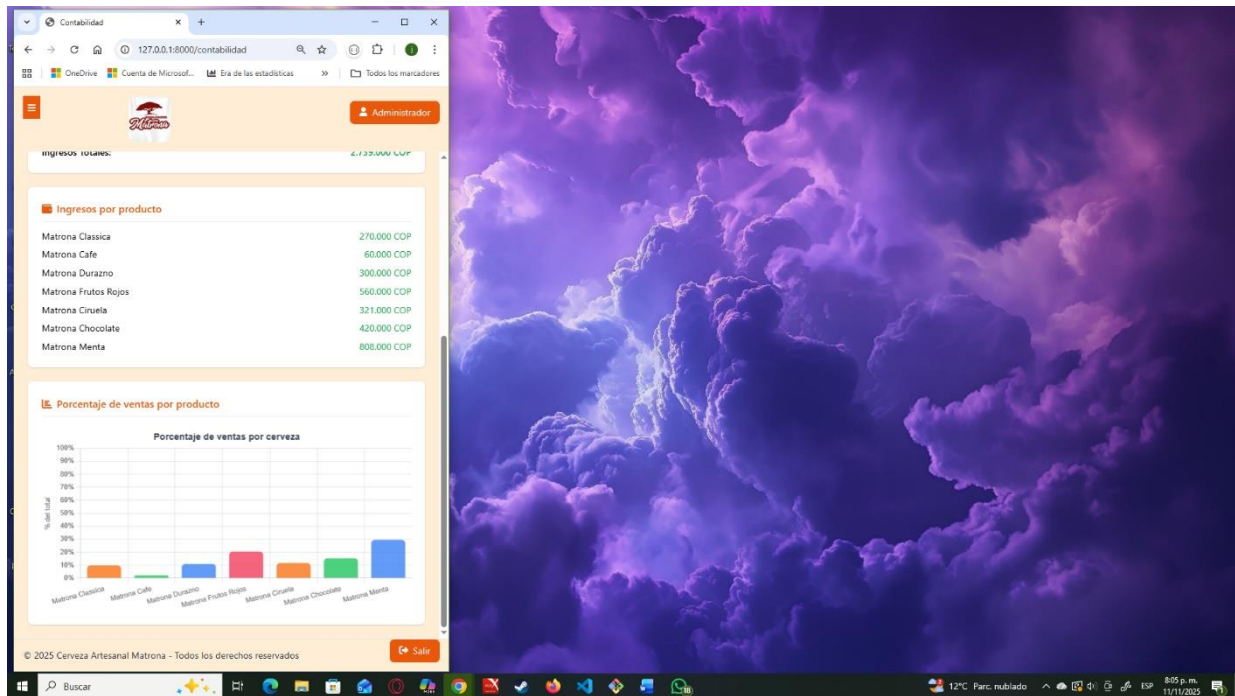
móviles



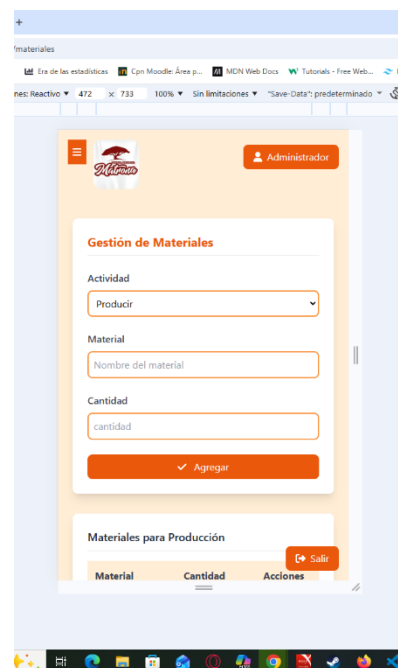
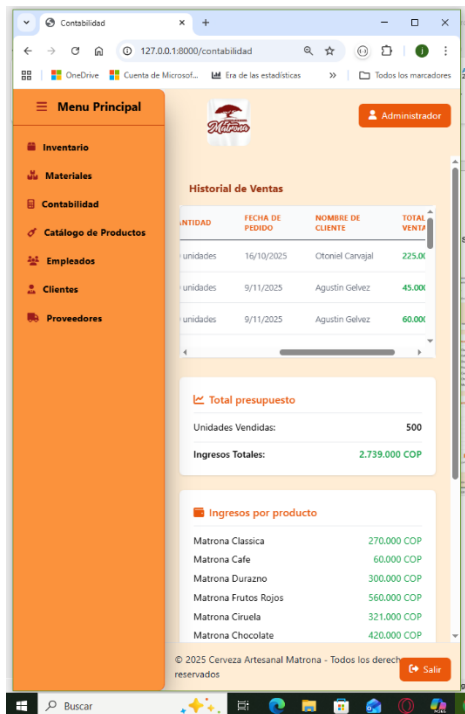
Otras pantallas pequeñas con resultados muy positivos



Tablas y contenedores adaptables adaptables a diferentes resoluciones de pantalla

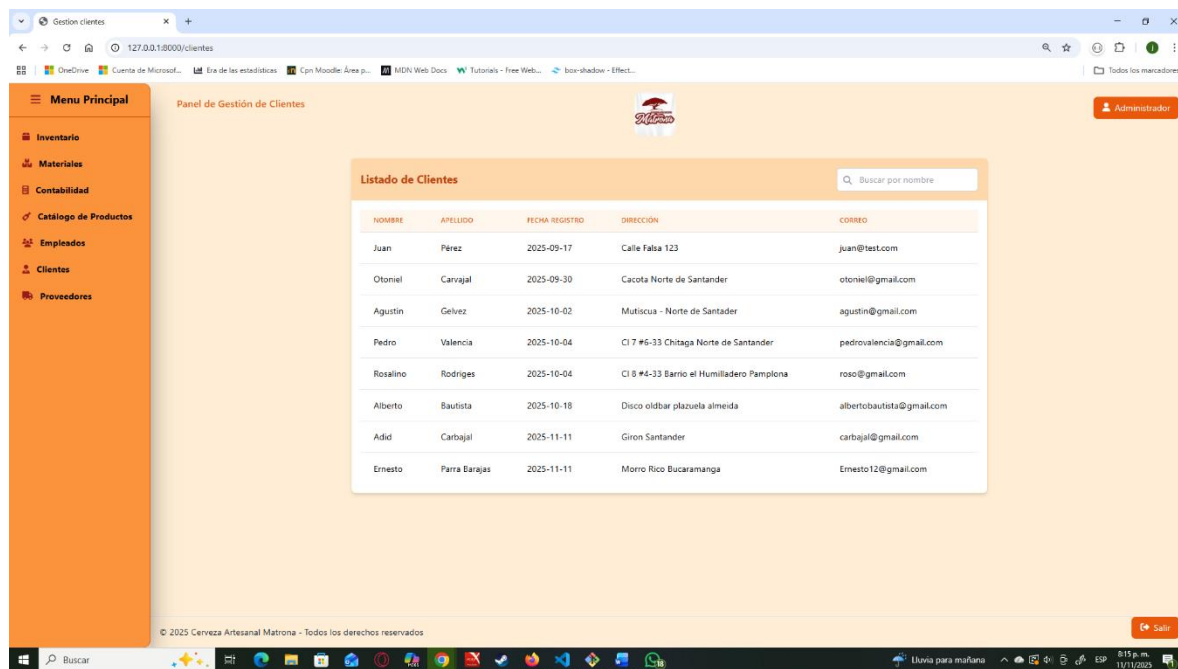


barras de navegación “overflow” para fácil desplazamiento en pantallas pequeñas

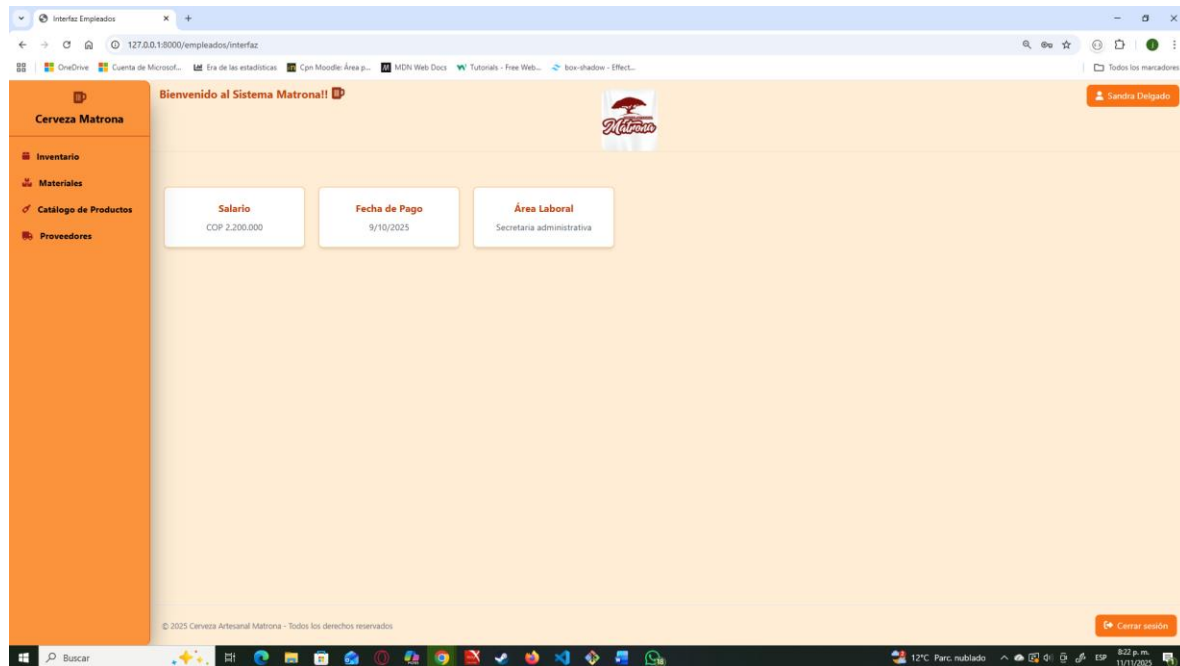


Interfaces dinámicas agregadas

gestionClientes: esta interfaz cumple la función de listar todos los clientes registrados en la base de datos y mostrarlos dinámicamente en una tabla.



empleadosInterfaz: esta interfaz es la interfaz principal del empleado de la empresa, en ella encontrara las funciones que el sistema le permite y sus datos personales, también renderizada dinámicamente con los datos reales del empleado que obtenemos con el fetch a nuestra API de FastAPI, obteniendo así los datos del token luego de la autenticación JWT de cada usuario, de esta manera logramos mostrar los datos de cada usuario logeado.



Uso de Git para control de versiones

Para el trabajo realizado al frontend, se ha creado una rama secundaria Git para trabajar en esta rama manteniendo el sistema de la rama principal main, intacto y completamente funcional evitando posibles errores graves que rompan el sistema por completo.

```
MINGW64/c/Users/Jefferson/proyecto-matrona-fastapi
Jefferson@DESKTOP-R403HBD MINGW64 ~/proyecto-matrona-fastapi (componentes-reutilizables)
$ git branch
* componentes-reutilizables
main

Jefferson@DESKTOP-R403HBD MINGW64 ~/proyecto-matrona-fastapi (componentes-reutilizables)
$
```


Posteriormente al estar totalmente verificado el funcionamiento del sistema se ha creado el commit, se fusionaron las ramas y se ha enviado la actualización al repositorio remoto del proyecto en GitHub.

Otras mejoras Frontend implementadas o corregidas

- ✓ Corrección de tipografía
- ✓ Corrección a los colores de interfaces, con cuidado especial para no perder la coherencia entre pantallas manteniendo el estilo de colores base
- ✓ Verificación de enlaces de navegación
- ✓ Mensajes de alerta en acciones post uso de inputs
- ✓ Validación para campos input, de acuerdo a los tipos de datos permitidos en la base de datos

Conclusiones

La implementación de JINJA2, motor de plantillas de Python en nuestro proyecto nos ha brindado múltiples beneficios, se ha logrado crear componentes reutilizables de manera sencilla, además, nos permite trabajar con variables y funciones de nuestro backend “FastAPI” utilizándolas al renderizar cada plantilla, esto nos abre un camino con infinitas posibilidades de mejora en la optimización del código logrando un proyecto robusto, mantenible y escalable. Por último, se concluye el trabajo con resultados positivos, se presenta un frontend responsive, altamente adaptable a pantallas de diferentes resoluciones, incluso móviles, manteniendo la consistencia entre elementos sin romper el estilo visual esto gracias al uso de flexbox y otras propiedades css claves para lograr esta adaptabilidad, además, se hicieron mejoras en validaciones de inputs y mensajes de feedback, se probó la navegación entre pantallas corrigiendo enlaces rotos y se ajustó el estilo visual. De esta manera continuamos en la consolidación en las diferentes funciones de este gran proyecto, el cual presenta mejoras significativas en todos los aspectos, gracias al trabajo diario realizado en este caso al Frontend.

Link del repositorio del proyecto en GitHub: <https://github.com/Jeffersonmendezgh/Software-de-Gestion-Administrativa-Matrona.git>