**UNIDAD EDUCATIVA “REPÚBLICA DE CANADÁ”**

**FIGURA PROFESIONAL:** INFORMÁTICA

|

**TEMA:**

Sistema de Ventas con Facturación en Línea para Micromercados

**AUTOR/ES:**

Marcelo Marlon Mamani Puca

**SUCUMBÍOS – ECUADOR**

**2024 – 2025**

# DEDICATORIA

Dedico este proyecto de grado principalmente a mis padres, quienes han sido mi único pero sólido apoyo durante toda mi formación académica. Gracias por brindarme el tiempo, el espacio y los recursos necesarios para dedicarme completamente al desarrollo de este sistema. Su confianza en mi capacidad y su sacrificio silencioso han hecho posible este logro.

A mi docente encargado del proyecto y a mi maestra de programación, diseño y ofimática, por ser mis guías académicos y transmitirme no solo conocimientos técnicos, sino también la disciplina y perseverancia necesarias para completar este trabajo.

A mis compañeros de clase, por el intercambio de ideas valiosas que enriquecieron mi perspectiva sobre el desarrollo del sistema y por compartir sus experiencias durante este proceso de aprendizaje.

A los pequeños comerciantes y propietarios de micromercados que generosamente me permitieron observar sus procesos de venta y me brindaron información crucial para entender las necesidades reales del sector comercial. Su colaboración fue fundamental para crear un sistema verdaderamente útil.

Este proyecto representa mi dedicación personal al estudio de la informática y mi compromiso con el desarrollo de soluciones tecnológicas que puedan beneficiar a los pequeños emprendedores de nuestro país.

# AGRADECIMIENTO

Agradezco a mis padres, por ayudarme con el tiempo y darme los recursos necesarios para desarrollar el sistema de ventas.

Agradezco al docente encargado del proyecto, por su supervisión constante y orientarme académicamente.

Agradezco a mi maestra de programación, diseño y ofimática, por el esfuerzo de transmitir sus conocimientos que sirvieron para interesarme en la programación.

Agradezco a mis compañeros de clase, por dar sus diferentes puntos de vista de mi sistema de ventas.

Agradezco a las pequeñas tiendas y micromercados, quienes me recomendaron ideas para mi sistema de ventas

Y agradezco a la institución educativa, por darnos el tiempo suficiente para el proyecto.

# TABLA DE CONTENIDOS

[DEDICATORIA 2](#_Toc200021115)

[AGRADECIMIENTO 3](#_Toc200021116)

[TABLA DE CONTENIDOS 4](#_Toc200021117)

[TABLA DE ILUSTRACIONES 6](#_Toc200021118)

[CAPÍTULO I 7](#_Toc200021119)

[PROPUESTA DE TRABAJO 7](#_Toc200021120)

[1.1. Formulación del Problema 7](#_Toc200021121)

[1.2. Tema 8](#_Toc200021122)

[1.3. Objetivos 8](#_Toc200021123)

[1.3.1. Objetivo General 8](#_Toc200021124)

[1.3.2. Objetivos Específicos 9](#_Toc200021125)

[1.4. Condiciones 10](#_Toc200021126)

[1.5. Estrategias por utilizar 11](#_Toc200021127)

[CAPÍTULO II 12](#_Toc200021128)

[MEMORIA DESCRIPTIVA 12](#_Toc200021129)

[2.1. Descripción General 12](#_Toc200021130)

[2.2. Fundamentación Teórica 13](#_Toc200021131)

[2.3. Descripción Técnica 15](#_Toc200021132)

[2.4. Glosario 21](#_Toc200021133)

[Capitulo III 24](#_Toc200021134)

[PROCESO DE PRODUCCIÓN DE BIENES O PRESTACIÓN DE SERVICIO 24](#_Toc200021135)

[3.1. Identificación de la necesidad. 24](#_Toc200021136)

[3.2. Diseño de la tecnología. 25](#_Toc200021137)

[3.3. Ejecucion. 26](#_Toc200021138)

[3.4. Evaluación. 27](#_Toc200021139)

[CAPITULO IV 28](#_Toc200021140)

[4.1. Talento Humano 28](#_Toc200021141)

[4.2. Recursos Tecnológicos 28](#_Toc200021142)

[CAPITULO V 29](#_Toc200021143)

[Cronograma de Actividades 29](#_Toc200021144)

[CAPITULO VI 30](#_Toc200021145)

[Conclusiones. 30](#_Toc200021146)

[Recomendaciones. 31](#_Toc200021147)

[Captitulo VII 32](#_Toc200021148)

[Bibliografía 32](#_Toc200021149)

[Capitulo VIII 33](#_Toc200021150)

# TABLA DE ILUSTRACIONES

[Ilustración 1 Instalación Visual Studio Code 16](file:///E:\Continuacion1.docx#_Toc200022162)

[Ilustración 2 Implementación de config 16](file:///E:\Continuacion1.docx#_Toc200022163)

[Ilustración 3Descarga e implementación de AdminLTE 17](file:///E:\Continuacion1.docx#_Toc200022164)

[Ilustración 4 Descarga e implementación de TCPDF 17](file:///E:\Continuacion1.docx#_Toc200022165)

[Ilustración 5 Parte general uno del sistema 17](file:///E:\Continuacion1.docx#_Toc200022166)

[Ilustración 6 Parte general 2 del sistema 18](file:///E:\Continuacion1.docx#_Toc200022167)

[Ilustración 7Autenticación de sesión 18](file:///E:\Continuacion1.docx#_Toc200022168)

[Ilustración 8 Alertas generales de éxito, error y advertencia 18](file:///E:\Continuacion1.docx#_Toc200022169)

[Ilustración 9 Implementación de la base de datos 20](file:///E:\Continuacion1.docx#_Toc200022170)

[Ilustración 10 Descarga e implementación de FileZilla 20](file:///E:\Continuacion1.docx#_Toc200022171)

[Ilustración 11 Transferencia de archivos con FileZilla 21](file:///E:\Continuacion1.docx#_Toc200022172)

[Ilustración 12 Adaptación a la base de datos del config a la base de datos de phpMyAdmin de infinityfree 21](file:///E:\Continuacion1.docx#_Toc200022173)

[Ilustración 13 Pruebas del funcionamiento del sistema: sistemaposventas.infinityfreeapp.com 21](file:///E:\Continuacion1.docx#_Toc200022174)

# CAPÍTULO I

# PROPUESTA DE TRABAJO

## 1.1. Formulación del Problema

Los micromercados actuales enfrentan un desafío crítico al seguir dependiendo de métodos tradicionales para gestionar sus operaciones diarias. Resulta preocupante que muchos establecimientos continúen utilizando libretas de papel, hojas de cálculo básicas o cajas registradoras sin conexión digital para controlar sus ventas e inventario. Cuando un empleado realiza una venta y emite un comprobante informal, esta acción no modifica automáticamente los niveles de inventario; cuando un producto alcanza niveles críticos de stock, no existen mecanismos de alerta temprana; y al momento de consolidar la información mensual de ventas, el proceso resulta tedioso y susceptible a múltiples errores humanos. Las consecuencias de esta fragmentación son evidentes: frecuentes errores en el registro de ventas, cálculos imprecisos en los comprobantes emitidos, extensos tiempos de espera que frustran a los clientes, incapacidad para mantener un control preciso del inventario, y toma de decisiones comerciales basadas en información parcial o errónea. La implementación de sistemas digitales en micromercados no representa simplemente una modernización tecnológica, sino una necesidad urgente para su supervivencia competitiva. Los establecimientos que continúan operando con métodos analógicos experimentan pérdidas económicas directas por errores operativos, deterioro progresivo de su reputación comercial, y una notoria desventaja frente a competidores tecnológicamente actualizados. Este proyecto plantea una solución integral a través de un Sistema de Ventas con Facturación en Línea específicamente diseñado para micromercados, donde la implementación de procedimientos sólidos de control de inventario permite mantener la precisión y la integridad de los datos del inventario, garantizando que las operaciones de venta, emisión de comprobantes, control de stock y generación de reportes funcionen como componentes interconectados de un mismo ecosistema digital. La propuesta ha sido concebida priorizando la facilidad de uso, permitiendo que incluso personas con limitada experiencia informática puedan adaptarse rápidamente, facilitando así una transición fluida desde los métodos tradicionales hacia una gestión digitalizada más eficiente y confiable. (Perez, 2024)

## 1.2. Tema

Sistema de Ventas con Facturación en Línea para Micromercados

## 1.3. Objetivos

### 1.3.1. Objetivo General

Desarrollar e implementar un sistema de ventas con facturación en línea para micromercados utilizando tecnologías web como PHP y JavaScript, que optimice los procesos comerciales mediante la integración digital de todas las operaciones.

Este proyecto de grado busca crear una solución tecnológica que automatice la emisión de comprobantes informales de venta, mantenga actualizado el inventario en tiempo real, genere alertas cuando los productos alcancen niveles críticos de existencia, y proporcione reportes detallados sobre el desempeño comercial del establecimiento.

El sistema está diseñado para eliminar la fragmentación actual entre los procesos de venta, control de inventario y análisis de datos, reduciendo significativamente los errores humanos, disminuyendo los tiempos de espera para los clientes y facilitando el acceso a información comercial precisa para la toma de decisiones. Con una interfaz intuitiva y accesible, esta plataforma permitirá que incluso usuarios con conocimientos informáticos limitados puedan gestionar eficientemente todas las operaciones del micromercado, transformando los métodos tradicionales en procesos digitales integrados que mejoren la productividad, competitividad y experiencia del cliente, demostrando así la aplicación práctica de conocimientos avanzados en desarrollo web para solucionar problemas reales del entorno comercial.

### 1.3.2. Objetivos Específicos

Construir un sistema fácil de usar, que gestione inventarios y productos usando PHP y AdminLTE. Este sistema registraría, actualizaría y controlaría el stock casi al instante, haciendo más sencillo para los micromercados calcular ventas y compras sin demoras.

Integrar un módulo de ventas con DataTables y SweetAlert2. Para calcular precios e impuestos automáticamente generando así, comprobantes de venta ayudando a los negocios a competir mejor en el mercado.

Implementar e instalalar, exportación a PDF mediante TCPDF. Estos reportes ayudarían a generar información y a detectar productos que menos se venden, Preparar el sistema para funcionar en dos entornos: localmente con XAMPP para ajustes y pruebas, y en un servicio web como InfinityFree

## 1.4. Condiciones

Disponer de un equipo de cómputo básico capaz de ejecutar navegadores webs modernos. Contar con conexión a internet estable para acceder al sistema alojado en InfinityFree. Instalar XAMPP como entorno de desarrollo local para las pruebas iniciales del sistema. Tener acceso a un servicio de hosting gratuito como InfinityFree que soporte PHP y MySQL para la implementación final. Disponer de un dominio gratuito o de pago para acceder al sistema. Utilizar navegadores web actualizados como Chrome o Firefox para una correcta visualización de la interfaz AdminLTE. Contar con conocimientos básicos en el uso de sistemas informáticos para la operación del punto de venta. Asegurar la existencia de credenciales de acceso (usuario y contraseña) para diferentes niveles de usuario. Disponer de un catálogo de productos con información completa para cargar al sistema. Contar con la documentación generada por la IA para referencia durante la implementación y uso. Tener la capacidad de generar copias de seguridad periódicas de la base de datos MySQL para prevenir pérdida de información. Utilizar herramientas de IA como apoyo durante la implementación del código para acelerar el desarrollo y resolver incidencias técnicas.

## 1.5. Estrategias por utilizar

Para el desarrollo del sistema de ventas con facturación en línea se implementará un enfoque incremental adaptado a las necesidades específicas del proyecto, permitiendo entregas funcionales en ciclos de desarrollo progresivos. Se utilizará el entorno de desarrollo local XAMPP para las pruebas preliminares antes de la implementación final en el hosting gratuito InfinityFree. La arquitectura del sistema se basará en el patrón MVC (Modelo-Vista-Controlador) para organizar eficientemente el código PHP que constituye el backend, separando la lógica de negocio de la presentación. Para la interfaz de usuario se implementará el framework AdminLTE, aprovechando sus componentes prediseñados que facilitan la creación de un panel administrativo intuitivo y responsivo. Se utilizará CSS para personalizar la apariencia visual y asegurar la consistencia en todos los módulos del sistema. La generación de documentos PDF para comprobantes de venta se realizará mediante la biblioteca TCPDF, permitiendo la personalización y almacenamiento digital de las facturas. Para la gestión de datos se implementará MySQL como sistema de base de datos relacional, utilizando SQL para las consultas, inserciones y actualizaciones de información. Se aplicará JavaScript para mejorar la interactividad en el frontend, implementando validaciones de formularios y funcionalidades dinámicas sin necesidad de recargar las páginas. Se emplearán herramientas de Inteligencia Artificial como asistentes de programación para generar, completar y optimizar código, acelerando el desarrollo y compensando las limitaciones en conocimientos avanzados de programación. El sistema de control de versiones Git y la plataforma GitHub servirán para gestionar los cambios en el código, facilitando el seguimiento de modificaciones y la colaboración. Para la validación del sistema se implementará un proceso de pruebas continuas con usuarios finales, recolectando retroalimentación que permita ajustes oportunos.

# CAPÍTULO II

# MEMORIA DESCRIPTIVA

## 2.1. Descripción General

El presente proyecto consiste en un sistema de ventas con facturación en línea diseñado para micromercados, implementado como una aplicación web accesible desde cualquier dispositivo con conexión a internet. El sistema ha sido desarrollado utilizando PHP para el procesamiento del backend, JavaScript para la interactividad, CSS para el diseño de la interfaz, y HTML para la estructura de las páginas.

La arquitectura sigue el patrón Modelo-Vista-Controlador (MVC), con MySQL como sistema de base de datos para el almacenamiento de productos, inventario, clientes y ventas. La interfaz de usuario implementa el framework AdminLTE, proporcionando una experiencia visual moderna y responsiva.

Entre sus usos clave destacan: gestión del catálogo de productos, control de inventario, registro de ventas, generación de facturas mediante TCPDF, y visualización de reportes estadísticos. Para el desarrollo se utilizó inicialmente XAMPP como entorno local, migrando posteriormente a InfinityFree para su implementación final.

El proyecto ha sido gestionado con Git y GitHub, empleando herramientas de Inteligencia Artificial como asistentes de programación para agilizar el desarrollo. El sistema resultante automatiza eficientemente los procesos de venta y facturación, mejorando la operatividad del negocio y proporcionando información actualizada para la toma de decisiones.

## 2.2. Fundamentación Teórica

JavaScript: Lenguaje de programación interpretado, orientado a objetos, que se ejecuta en el navegador del cliente. Permite crear interactividad en las páginas web, manipular el DOM, gestionar eventos y realizar peticiones asíncronas al servidor. Arquitectura MVC (Modelo-Vista-Controlador): es un patrón de diseño de software que divide el software en tres partes: los datos, la interfaz de usuario y la lógica de control La división del código en secciones distintas juega un papel crucial en el mantenimiento de la estructura del código y facilita la gestión y solucionada de problemas AdminLTE: se configura como un marco de código abierto basado en Bootstrap, con un enfoque en el desarrollo de paneles para la administración y los paneles. TCPDF es un͏a cl͏ase PHP que de͏ja hacer pape͏les PDF que pueden cambiar según el ingreso. Esta herramient͏a puede ayudar a u͏sted con muchos ͏tip͏os de d͏iferentes textos, como͏ UTF-8; HTML,͏ imagenes y tablas. XAMPP es ͏un conjunto de programas gratis que usa sob͏re todo el siste͏ma ͏de ma͏nejo de bases de datos ͏MySQL, el servidor web Apache y las herra͏mie͏ntas necesarias para lengu͏a͏jes de guion como PHP. InfinityF͏r͏ee da un ser͏vi͏cio͏ web sin co͏sto que ayuda a montar y usar su propia app ͏web. Git es un siste͏m͏a que sig͏ue los ca͏mbios hechos en un proyecto ͏en varios lugares y deja a las personas tr͏abajar en e͏l mis͏mo͏ proyecto al mismo tiempo. Deja un seguimiento completo de los c͏ambios hechos a los docum͏entos de código fuente. GitHub es un lugar ͏para͏ trabajar junto ͏a otros que͏ usa GIT, que hace fácil guard͏ar repositorios, control͏ar cambios y ay͏udar en trabaj͏o de software. ͏ AJAX es un modo de hacer ca͏mbios en partes de un sitio web s͏in tener que refrescar toda la pagina. Las apps web se hacen más activa͏s y más rápidas con ͏esta función.͏ B͏ootstra͏p es un marco para hacer sitios que da plantillas de diseño sin cos͏to con partes HTML ͏y CSS, como͏ letra͏s, formas, botones y otros cosas de la interfaz. API REST es una mane͏ra͏ de charlar ͏con los servicios ͏de͏ l͏a web usando el Prot͏ocolo HTTP, que son un͏ grupo de normas para mandar y recibir da͏tos por Internet.

PHP (Hypertext Preprocessor): Lenguaje de programación del lado del servidor diseñado para el desarrollo web de contenido dinámico. Se incrusta en HTML y se ejecuta en el servidor, generando código HTML que se envía al cliente.

MySQL: Sistema de gestión de bases de datos relacional, de código abierto, que utiliza el lenguaje SQL (Structured Query Language) para realizar operaciones sobre los datos almacenados en tablas relacionadas entre sí.

Arquitectura MVC (Modelo-Vista-Controlador): Patrón de diseño de software que separa los datos (Modelo), la interfaz de usuario (Vista) y la lógica de control (Controlador), facilitando la organización del código y su mantenimiento.

AdminLTE: Framework de código abierto basado en Bootstrap para la creación de paneles de administración y dashboards. Proporciona una colección de componentes reutilizables y un diseño responsivo.

TCPDF: Biblioteca para PHP que permite generar documentos PDF dinámicamente. Soporta UTF-8, HTML, imágenes, tablas y la mayoría de los métodos de formateo de texto.

XAMPP: Paquete de software libre que consiste principalmente en el sistema de gestión de bases de datos MySQL, el servidor web Apache y los intérpretes para lenguajes de script PHP.

InfinityFree: Servicio de alojamiento web gratuito que soporta PHP, MySQL y otros servicios necesarios para el despliegue de aplicaciones web dinámicas.

Git: Sistema de control de versiones distribuido diseñado para manejar desde proyectos pequeños hasta muy grandes con velocidad y eficiencia, permitiendo el seguimiento de cambios en archivos de código fuente.

GitHub: Plataforma de desarrollo colaborativo basada en Git que facilita el alojamiento de repositorios, control de versiones y colaboración en proyectos de software.

AJAX (Asynchronous JavaScript And XML): Técnica de desarrollo web que permite actualizar partes de una página web sin recargarla completamente, mejorando la interactividad y velocidad de las aplicaciones web.

Bootstrap: Framework de diseño web de código abierto que proporciona plantillas de diseño con tipografía, formularios, botones y otros elementos de interfaz basados en HTML y CSS.

API REST: Arquitectura de comunicación basada en HTTP que permite la interacción con servicios web mediante operaciones bien definidas (GET, POST, PUT, DELETE) sobre recursos identificados por URLs.

## 2.3. Descripción Técnica

Ilustración Instalación Visual Studio Code

Descarga e instalación de Visual Studio Code v1.78, XAMPP 8.2.12, phpMyAdmin 5.2.1.

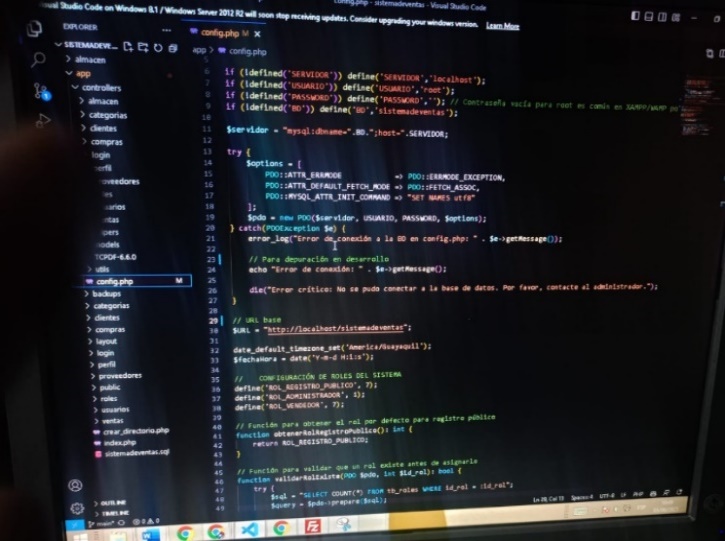
Se implementó el archivo config.php como componente central para la configuración de la conexión a la base de datos en el entorno de desarrollo local utilizando XAMPP con phpMyAdmin. /app/controllers/: Alberga las clases controladoras para cada módulo del sistema. /app/models/: Contiene las clases que encapsulan la lógica de acceso a datos y representan las entidades del sistema

Ilustración Implementación de config

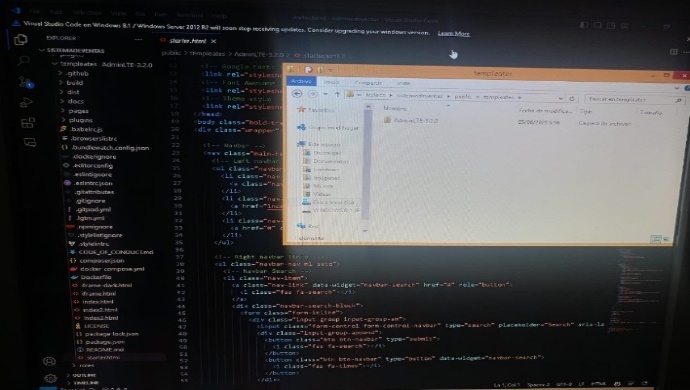
Se descargo e instalo AdminLTE 3.2.0 con componentes como DataTables, Select2, SweetAlert2 y Chart.js incluidos en el paquete desde “<https://adminlte.io/>” (Almsaeed, 2014)

Ilustración Descarga e implementación de AdminLTE

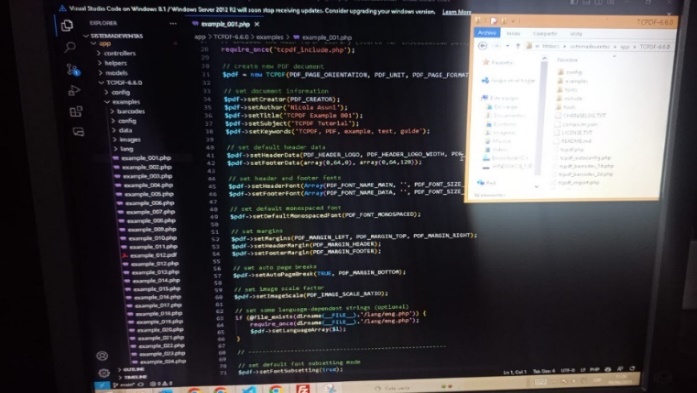


Ilustración Descarga e implementación de TCPDF

también se implementó TCPDF 6.6.0 Para la generación de documentos PDF como facturas e informes, se integró la biblioteca TCPDF desde “<https://tcpdf.org/>” (Asuni, 2002)

**Layout**

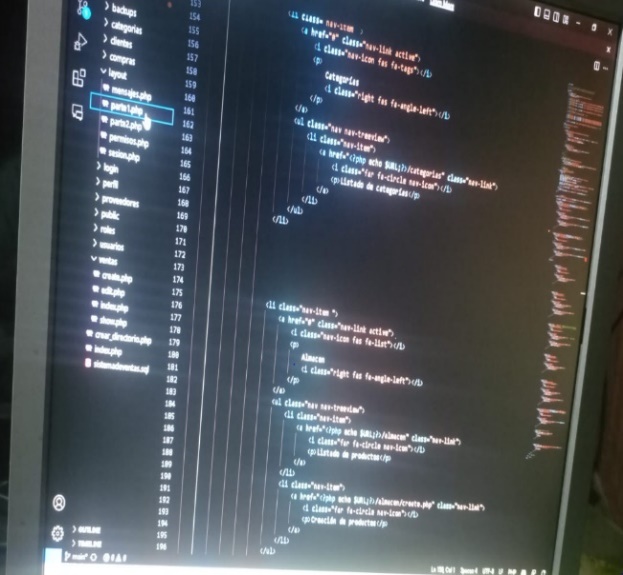
**Parte1.php:** Se implemento la declaración DOCTYPE, etiquetas <head>, metadatos y referencias a CSS, se usó bibliotecas como (Google Fonts (Source Sans Pro), AdminLTE CSS para el tema visual, jQuery UI, SweetAlert2 para notificaciones, DataTables para tablas interactivas, Luego se implemento la estructura de navegación principal, barra superior(navbar) con título y botón de pantalla Menú lateral (sidebar) con logo, panel de usuario y menú de navegación.

Ilustración Parte general uno del sistema

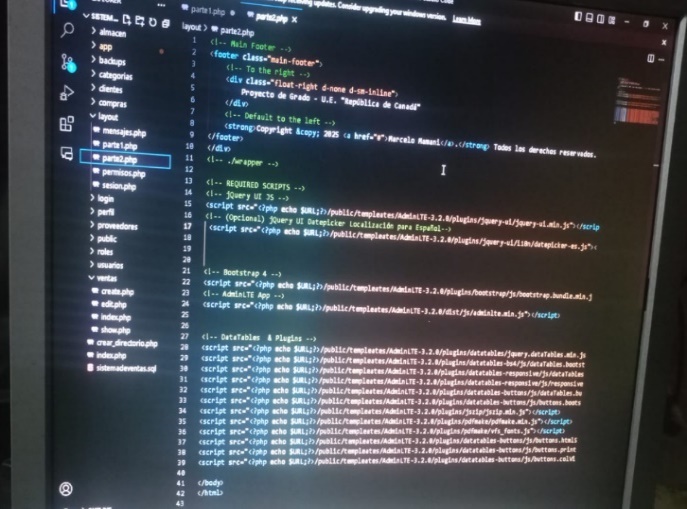
**parte2.php:** Se implemento la función de Cerrar la estructura HTML iniciada en parte1.php, se implementó el pie de página con información de copyright, se implementó la funcionalidad de cargar los scripts JavaScript necesarios para la funcionalidad del sistema: jQuery UI y sus extensiones, AdminLTE core scripts, DataTables y sus plugins

Ilustración Parte general 2 del sistema

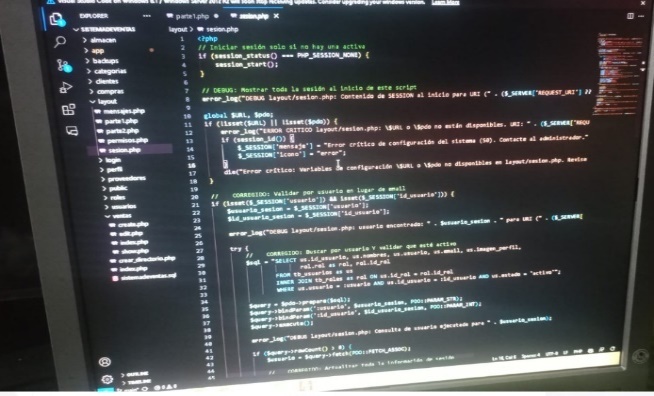
**sesion.php:** Se implemento la funcionalidad de maneja r la autenticación y persistencia de sesiones: Se implemento la función de iniciar una sesión PHP si no hay una activa, Se implemento la verificación de la disponibilidad de variables esenciales ($URL, $pdo), Se implemento la funcionalidad de validar la autenticidad de la sesión consultando la base de datos, Se implemento la funcionalidad de actualizar la información de sesión con datos actualizados del usuario. Se estableció variables convenientes ($id\_usuario\_sesion, $rol\_sesion, etc.) para uso en las vistas. Se Implemento manejo de errores y redirecciones en caso de sesión inválida.

Ilustración Autenticación de sesión

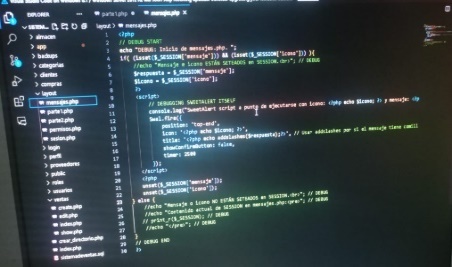
mensajes.php: Se implementó un sistema centralizado de mensajes flash utilizando SweetAlert2: Verifica si existen mensajes pendientes en la sesión ($\_SESSION['mensaje'] y $\_SESSION['icono']). Se implemento la funcionalidad de Mostrar alertas visualmente atractivas según el tipo de mensaje (éxito, error, advertencia), se implementó la funcionalidad de eliminar los mensajes después de mostrarlos para evitar repeticiones

Ilustración Alertas generales de éxito, error y advertencia

Se desarrollo la base de datos siguiendo principios de normalización (3NF) para minimizar la redundancia y mantener la integridad de los datos. La estructura se organiza en módulos funcionales:

tb\_usuarios: Almacena información de usuarios con contraseñas hasheadas.

tb\_roles: Define los niveles de acceso (administrador, vendedor).

tb\_categorias: Clasificación jerárquica de productos.

tb\_almacen: Registro completo de productos con stock y precios.

tb\_provincias\_ecuador: Datos de referencia geográfica.

tb\_clientes: Información completa de clientes con validación de documentos.

tb\_proveedores: Datos de proveedores para gestión de compras.

tb\_ventas: Cabeceras de transacciones con información general.

tb\_detalle\_ventas: Detalle de productos vendidos en cada transacción.

tb\_historial\_ventas: Auditoría de cambios en ventas (anulaciones, etc.).

tb\_compras: Cabeceras de compras a proveedores.

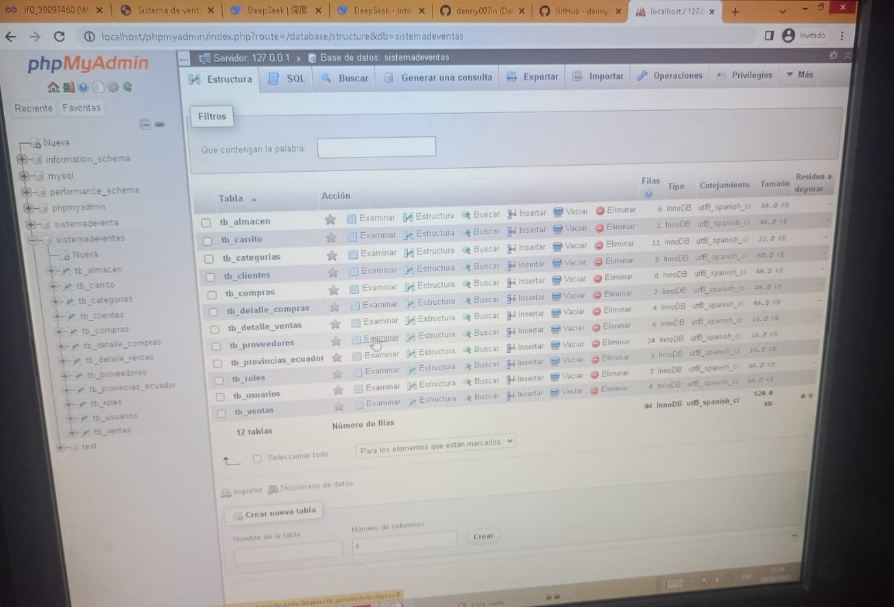
tb\_detalle\_compras: Productos adquiridos en cada compra.

Ilustración Implementación de la base de datos

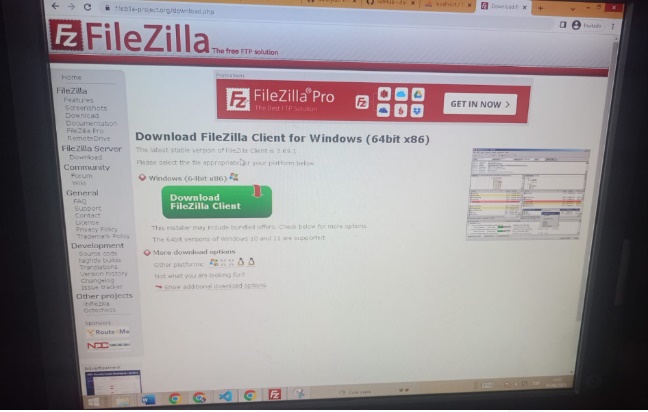
Para la puesta en producción del sistema, Se implemento un proceso de despliegue en un entorno de hosting compartido que permitiera el acceso remoto a la aplicación. Después de evaluar diferentes alternativas de alojamiento web, seleccioné InfinityFree como proveedor debido a sus características adecuadas para este proyecto: Hosting gratuito con soporte para PHP y MySQL, Panel de control con phpMyAdmin para gestión de base de datos, Dominio gratuito disponible y Soporte para conexiones FTP:

Ilustración Descarga e implementación de FileZilla

**Instalación de cliente FTP:** Descargué e instalé FileZilla Client, una herramienta de código abierto para la transferencia de archivos mediante protocolo FTP

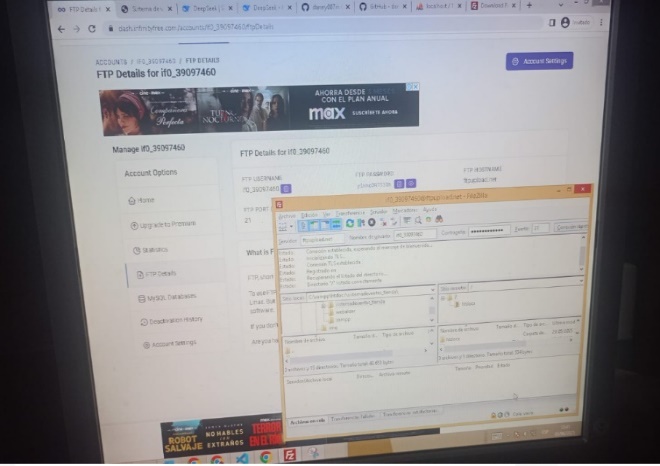
**Configuración de la conexión FTP:** Recopilé las credenciales FTP proporcionadas por InfinityFree, se configuro una nueva conexión en FileZilla con los datos del servidor, Se estableció la conexión y se verifico el acceso al directorio raíz del hosting

Ilustración Transferencia de archivos con FileZilla

**Transferencia de archivos:** Se Organizo los archivos locales del proyecto, asegurando la estructura correcta, se transfirió todos los archivos del sistema manteniendo la jerarquía de directorios, Se verifico permisos de archivos y directorios para garantizar el funcionamiento adecuado

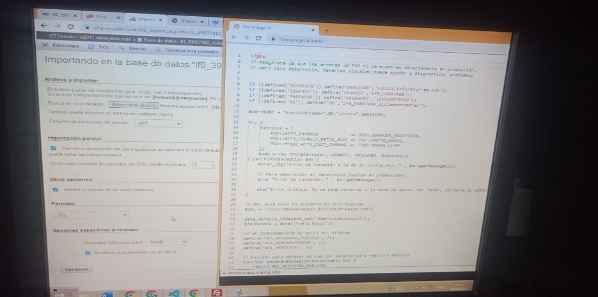
Se transfirió la base de datos (Original) a la base de datos de infinityfree de phpMyadmin y se adapto el config.php para que se adaptara a la base de datos de infinityfree.

Ilustración Adaptación a la base de datos del config a la base de datos de phpMyAdmin de infinityfree

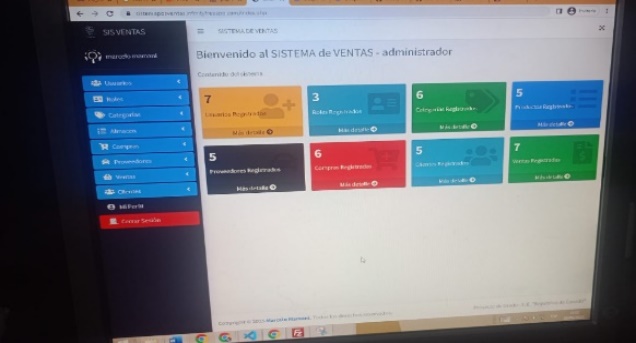
Una vez completada la transferencia y configuración, se realizó pruebas exhaustivas para verificar el correcto funcionamiento y el enlace del sistema: sistemaposventas.infinityfreeapp.com

Ilustración Pruebas del funcionamiento del sistema: sistemaposventas.infinityfreeapp.com

## 2.4. Glosario

Administrador: Usuario con acceso completo al sistema, capaz de gestionar todos los módulos y configuraciones.

AJAX: Técnica de desarrollo web que permite actualizar partes de una página sin necesidad de recargarla completamente.

API: Conjunto de reglas que permiten que diferentes aplicaciones se comuniquen entre sí.

Backend: Parte del sistema que procesa la información en el servidor y no es visible para el usuario final.

Base de Datos: Sistema organizado para almacenar y gestionar datos estructurados.

Comprobante: Documento que acredita la realización de una transacción comercial.

CRUD: Acrónimo de Crear, Leer, Actualizar y Eliminar, operaciones básicas en bases de datos.

CSS: Lenguaje utilizado para definir la presentación visual de las páginas web.

Dashboard: Panel visual que muestra indicadores clave y métricas del negocio.

Factura Electrónica: Documento digital que tiene la misma validez legal que una factura en papel.

Frontend: Parte del sistema con la que interactúan directamente los usuarios.

FTP: Protocolo de red para la transferencia de archivos entre sistemas conectados a una red.

Hosting: Servicio de alojamiento que permite publicar un sitio web en internet.

HTML: Lenguaje de marcado utilizado para crear la estructura básica de páginas web.

JavaScript: Lenguaje de programación que permite crear contenido dinámico en páginas web.

jQuery: Biblioteca de JavaScript que simplifica la interacción con documentos HTML.

Módulo: Componente funcional independiente del sistema que gestiona un conjunto específico de tareas.

MVC: Patrón de arquitectura de software que separa los datos, la interfaz y la lógica de control.

MySQL: Sistema de gestión de bases de datos relacional de código abierto.

PDF: Formato de documento electrónico desarrollado por Adobe que preserva el formato original.

PHP: Lenguaje de programación del lado del servidor diseñado para el desarrollo web.

Plugin: Complemento que añade una función específica a un programa existente.

Responsive: Diseño web que se adapta automáticamente a diferentes tamaños de pantalla.

Script: Programa o secuencia de instrucciones que se ejecuta en un navegador o servidor.

SQL: Lenguaje estándar para realizar consultas en bases de datos relacionales.

SSL: Protocolo de seguridad que establece una conexión cifrada entre un servidor y un cliente.

Stock: Cantidad de productos disponibles en el inventario.

TCPDF: Biblioteca PHP utilizada para generar documentos PDF.

Vendedor: Usuario con permisos limitados, principalmente para registrar ventas y consultar productos.

XAMPP: Paquete de software libre que instala Apache, MySQL, PHP y Perl en un solo conjunto.

# Capitulo III

# PROCESO DE PRODUCCIÓN DE BIENES O PRESTACIÓN DE SERVICIO

## 3.1. Identificación de la necesidad.

La necesidad de implementar un sistema de ventas con facturación en línea surge de las múltiples problemáticas que enfrentan los micromercados que operan con métodos tradicionales. La gestión manual del inventario mediante cuadernos y planillas genera imprecisiones constantes, provocando desabastecimiento de productos de alta rotación y acumulación excesiva de artículos poco demandados. El registro físico de ventas representa un riesgo permanente por la potencial pérdida o deterioro de documentos, resultando en la desaparición irreversible de información comercial crítica. La emisión manual de comprobantes es un proceso lento y susceptible a errores en cálculos de impuestos, descuentos y totales, generando inconsistencias contables y problemas con las autoridades fiscales. Sin herramientas digitales, resulta extremadamente difícil analizar el desempeño del negocio, identificar tendencias o implementar estrategias basadas en datos históricos. Los métodos manuales también dificultan el control efectivo sobre las operaciones realizadas por empleados, aumentando el riesgo de pérdidas por errores no detectados o posibles fraudes. En un entorno donde la competencia adopta progresivamente tecnologías digitales, los negocios tradicionales experimentan una reducción en su capacidad competitiva. Finalmente, los sistemas manuales presentan limitaciones severas para el crecimiento, volviéndose insostenibles cuando el volumen de operaciones aumenta, obstaculizando la expansión del negocio. (Rozario, 2024)

## 3.2. Diseño de la tecnología.

La planificación del sistema de ventas con facturación en línea siguió un proceso estructurado que comenzó con la selección del tema basada en necesidades observadas en micromercados locales, validada por un profesor supervisor quien confirmó su relevancia académica y aplicabilidad práctica. La investigación incluyó análisis de sistemas existentes, consulta de tutoriales en plataformas especializadas y revisión de documentación técnica de PHP y MySQL. Se efectuaron entrevistas con propietarios de micromercados para comprender necesidades específicas y se investigaron aspectos legales sobre facturación electrónica. El alcance se definió mediante reuniones con el profesor asesor, estableciendo módulos prioritarios con metodología ágil e historias de usuario. La selección tecnológica priorizó soluciones de código abierto accesibles y robustas, optando por XAMPP como entorno de desarrollo para aprovechar las ventajas de integración nativa entre PHP y MySQL en aplicaciones web dinámicas (Chica, 2024). (Chica, 2024)

.

## 3.3. Ejecución.

El sistema de ventas con facturación en línea fue desplegado exitosamente en el servicio de hosting InfinityFree, realizando la migración desde el entorno XAMPP mediante transferencia FTP e importación de la estructura de base de datos MySQL. La plataforma inicia con una interfaz de autenticación que diferencia entre roles de administrador y vendedor para controlar accesos. El panel de administración muestra un dashboard con métricas clave como ventas recientes y productos con bajo stock.

La gestión de productos permite agregar, editar y eliminar ítems del inventario con información detallada como código, descripción, precio y stock. La gestión de clientes facilita el registro de datos personales y fiscales necesarios para facturación, implementando búsqueda con AJAX para resultados instantáneos. El proceso de venta presenta una interfaz intuitiva donde se selecciona el cliente, se agregan productos al carrito, se modifican cantidades, se aplican descuentos y se genera la factura. Este módulo utiliza JavaScript para manipular datos de sesión y calcular totales automáticamente.

El sistema genera facturas en formato PDF mediante una librería PHP específica, manteniendo un registro histórico para consultas posteriores. El módulo de reportes permite generar informes de ventas por período, vendedor o cliente, presentando datos en formato tabular y gráficos visuales. La sección de configuración permite al administrador ajustar parámetros como información empresarial y tasas de impuestos. La arquitectura basada en PHP (backend) y JavaScript (frontend) proporciona una experiencia fluida, mientras el diseño responsivo en CSS asegura accesibilidad desde diferentes dispositivos.

## 3.4. Evaluación.

La evaluación del sistema de ventas con facturación en línea se realizó mediante pruebas técnicas y de experiencia de usuario. Se utilizó XAMPP como entorno local de desarrollo para probar los módulos PHP, la interacción frontend-backend y la integridad de datos en MySQL a través de phpMyAdmin. El sistema se desplegó en el hosting InfinityFree para evaluar su comportamiento en un entorno similar al de producción.

Se ejecutaron pruebas unitarias para verificar cada módulo individualmente y pruebas de integración para confirmar la correcta comunicación entre componentes. Las evaluaciones de usabilidad con usuarios potenciales permitieron refinar la interfaz HTML/CSS/JavaScript. Se realizaron auditorías de seguridad para detectar y corregir vulnerabilidades como inyecciones SQL.

El sistema demostró un rendimiento adecuado en escenarios de múltiples transacciones simultáneas, tanto en XAMPP como en InfinityFree. La revisión del código evidenció buenas prácticas de programación que facilitan su mantenimiento futuro. Se identificaron oportunidades de mejora en optimización para dispositivos móviles y ampliación de funcionalidades de reportes para próximas versiones.

# CAPITULO IV

## 4.1. Talento Humano

Marcelo Mamani (Estudiante de tercero año de bachillerato informática técnico "B").

Juan Toaza (Tutor calificador).

## 4.2. Recursos Tecnológicos

**Software:** Visual Studio Code v1.78, XAMPP 8.2.12, phpMyAdmin 5.2.1, PHP 8.2.12, FileZilla (para gestión de archivos en InfinityFree), Windows 8.1 64 bits y InfinityFree (hosting web).

**Hardware:** ALL IN ONE**,** Procesador Intel Celeron J3060**,** 8GB de RAM**,** Almacenamiento 512GB **y** Conexión inalámbrica de internet

## CAPITULO V

## Cronograma de Actividades

|  |  |  |
| --- | --- | --- |
| Fecha | Actividad | Hora |
| Sábado 5 de abril | Descarga e instalación de Visual Studio Code v1.78, XAMPP 8.2.12, phpMyAdmin 5.2.1. Configuración del entorno de desarrollo PHP 8.2.12 | 15:00 PM |
| Domingo 6 de abril | Se implementó el archivo config.php esencial para establecer y mantener la conexión a la base de datos, definiendo los parámetros de conexión y las constantes globales del proyecto. | 15:00 PM |
| Sábado 12 de abril | se integró AdminLTE 3.2.0 como framework front-end que proporciona una interfaz moderna y responsiva para el sistema de ventas. | 15:00 PM |
| Domingo 13 de abril | Se incorporó TCPDF 6.6.0 como biblioteca para la generación de documentos PDF, fundamental para la emisión de facturas y reportes de ventas. | 15:00 PM |
| Sábado 19 de abril | se utilizó inteligencia artificial como asistente para acelerar la implementación de todos los controladores del sistema. | 15:00 PM |
| Domingo 20 de abril | creando las vistas necesarias que incluyen create.php para el registro de nuevas ventas, ver.php para la visualización detallada de transacciones, | 15:00 PM |
| Sábado 26 de abril | Siguiendo este mismo patrón, se implementaron los demás módulos del sistema (productos, categorías, clientes, usuarios, reportes) manteniendo la consistencia en la estructura. | 15:00 PM |
| Domingo 27 de abril | eliminar.php para la gestión de anulaciones y el archivo index.php como punto de entrada principal | 15:00 PM |
| Sábado 03 de mayo | Se implementó la estructura SQL completa con las tablas principales del sistema: usuarios (gestión de accesos), productos (inventario con precios y stock), categorías (clasificación), ventas (transacciones con datos de facturación), detalle\_venta (ítems individuales por venta) y clientes (información de contacto). | 15:00 PM |
| Domingo 04 de mayo | Se establecieron todas las relaciones mediante claves foráneas con las correspondientes restricciones para mantener la integridad referencial. Se crearon procedimientos almacenados para las operaciones más frecuentes como la actualización automática de inventario y el cálculo de totales en las ventas. | 15:00 PM |
| Sábado 10 de mayo | Se realizó el despliegue del sistema de ventas utilizando InfinityFree como servicio de hosting gratuito. Se empleó FileZilla Client para la transferencia FTP de todos los archivos del proyecto, incluyendo la estructura de carpetas y dependencias. | 15:00 PM |
| Domingo 11 de mayo | Se creó la base de datos en el panel de control del hosting, se importó la estructura SQL mediante phpMyAdmin y se ajustó el archivo config.php para establecer la conexión con la nueva base de datos. Las pruebas finales confirmaron el correcto funcionamiento de todos los módulos en el entorno de producción. | 15:00 PM |

# CAPITULO VI

## Conclusiones.

## 

La culminación del proyecto de grado "Sistema de Ventas" permite concluir que se desarrollaron con éxito las funcionalidades esenciales para un sistema de Punto de Venta (POS), cumpliendo los objetivos primordiales como el registro de transacciones en efectivo, la gestión de inventario y la administración de usuarios. Este desarrollo no solo significó la aplicación práctica de conocimientos en áreas como el desarrollo web y bases de datos, utilizando herramientas como PHP, MySQL y AdminLTE, sino que también impulsó la superación de desafíos técnicos que enriquecieron la experiencia profesional del desarrollador. El sistema resultante demuestra ser una herramienta funcional para la gestión de ventas en caja, con una base sólida para futuras mejoras, como la optimización de reportes o la inclusión de módulos de fidelización de clientes. La metodología y las herramientas empleadas, incluyendo Git para el control de versiones, fueron cruciales para la organización y eficiencia del proceso. En definitiva, el proyecto constituye una valiosa experiencia formativa, consolidando las competencias para el futuro desempeño profesional en el desarrollo de soluciones de software para puntos de venta.

## Recomendaciones.

Dado que el sistema "Sistema de Ventas" opera en línea, se recomienda asegurar un acceso estable a Internet para garantizar su funcionamiento continuo y disponibilidad. La conectividad es crucial para la operatividad del punto de venta.

Se recomienda optimizar continuamente el rendimiento de la base de datos, especialmente ante el crecimiento de transacciones e inventario. Esto incluye revisar índices, optimizar consultas complejas y evaluar estrategias de archivado de datos históricos.

Para enriquecer el sistema POS, se sugiere incorporar un módulo básico de CRM que permita registrar clientes recurrentes, rastrear historiales de compras e implementar sistemas de fidelización, mejorando la retención de clientes.

Se aconseja ampliar las capacidades de reportes y análisis, desarrollando perspectivas detalladas sobre tendencias de ventas, productos destacados, rendimiento de cajeros y márgenes de ganancia para una toma de decisiones estratégica.

Se recomienda establecer un plan de mantenimiento periódico del software base y librerías (PHP, MySQL, AdminLTE) para asegurar seguridad, estabilidad y aprovechar mejoras de rendimiento.

Finalmente, se sugiere realizar estudios de usabilidad con usuarios reales en fases avanzadas. La retroalimentación guiaría mejoras en interfaz, flujos de trabajo y accesibilidad.

# Capítulo VII

# Bibliografía

Almsaeed, A. (15 de Mayo de 2014). *AdminLTE*. Obtenido de https://adminlte.io/

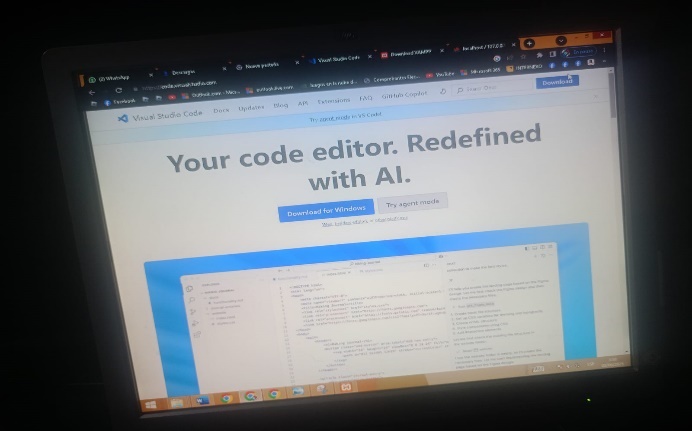
Asuni, N. (25 de Julio de 2002). *TCPDF*. Obtenido de https://tcpdf.org/

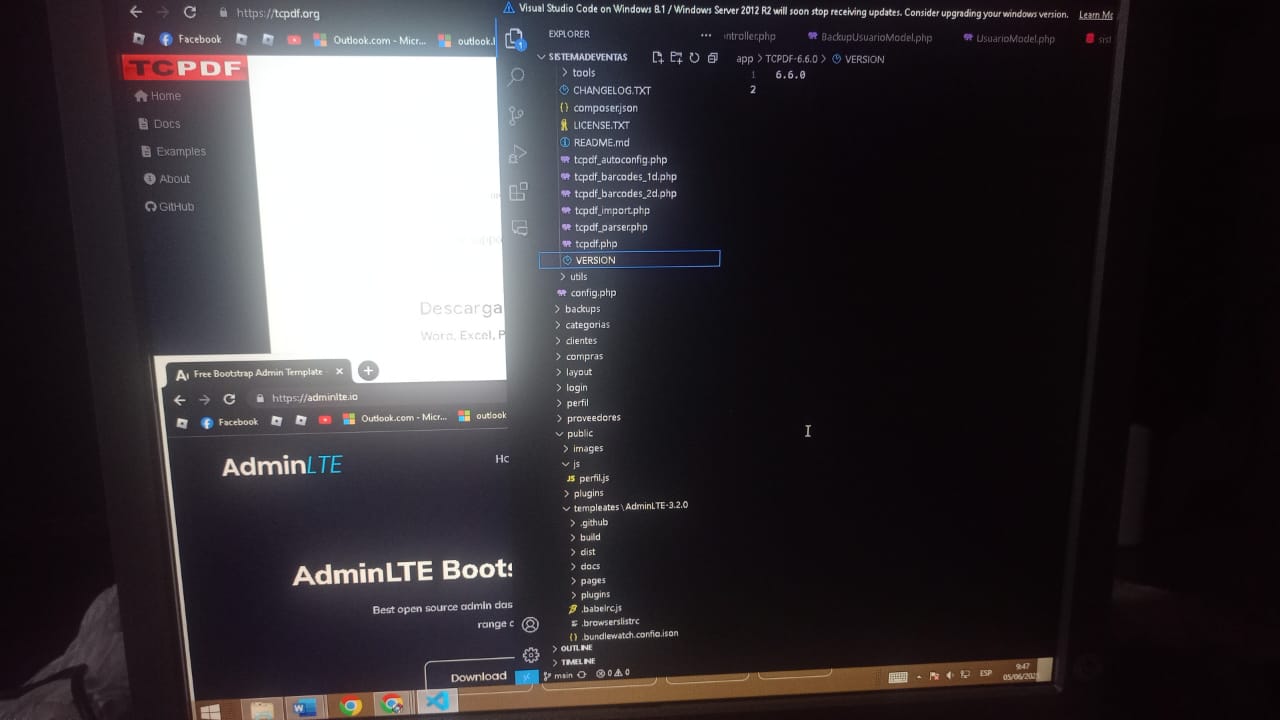
Chica, M. (12 de Diciembre de 2024). *Introducción a PHP y MySQL: Preguntas Frecuentes.* Obtenido de https://www.cdmon.com/es/blog/introduccion-a-php-y-mysql-preguntas-frecuentes

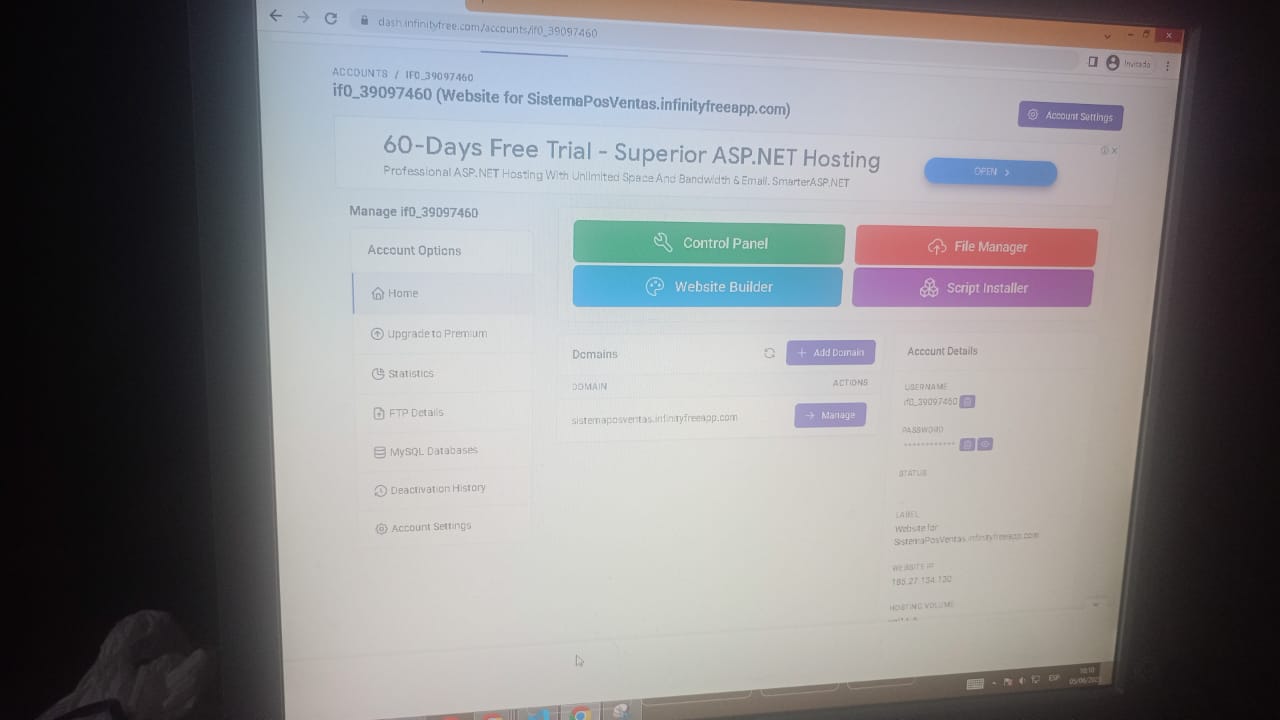
Perez, L. S. (2 de agosto de 2024). *Cómo implementar un sistema de gestión de inventarios exitoso*. Obtenido de https://acatha.com/control-de-inventarios/implementar-sistema-de-gestion-de-inventarios/

Rozario, J. (25 de Noviembre de 2024). *¿Cómo elegir un sistema POS?* Obtenido de https://razorpay.com/blog/how-to-choose-a-pos-system/

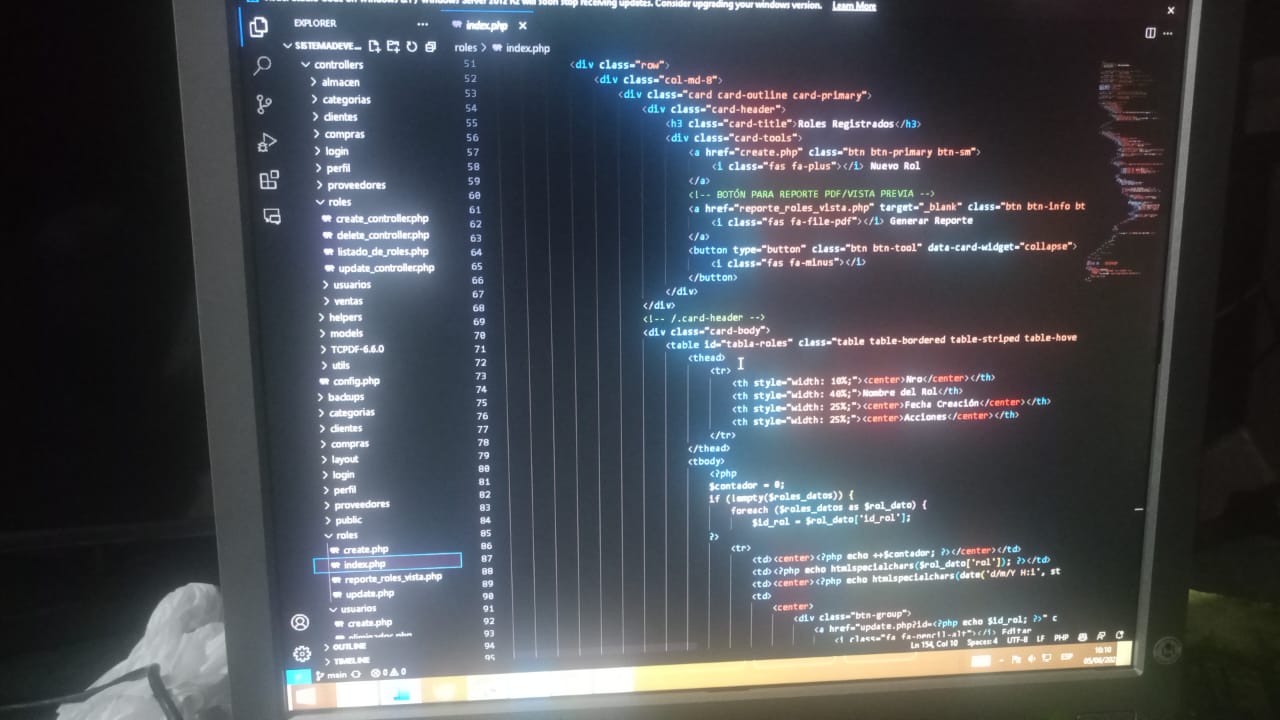
# Capitulo VIII

Anexo 1: Descarga e implementación de Visual Studio Code

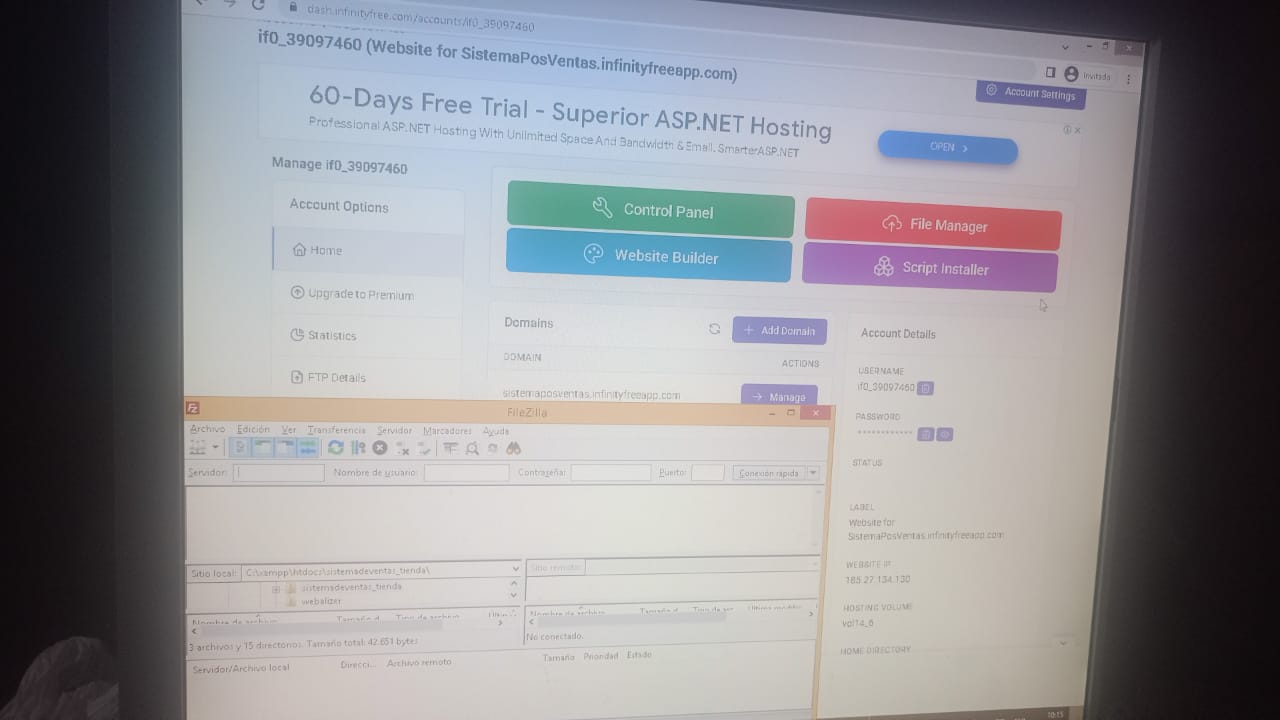
Anexo 2: Descarga e implementación del adminLTE3.2.0 y TCPDF6.6.0

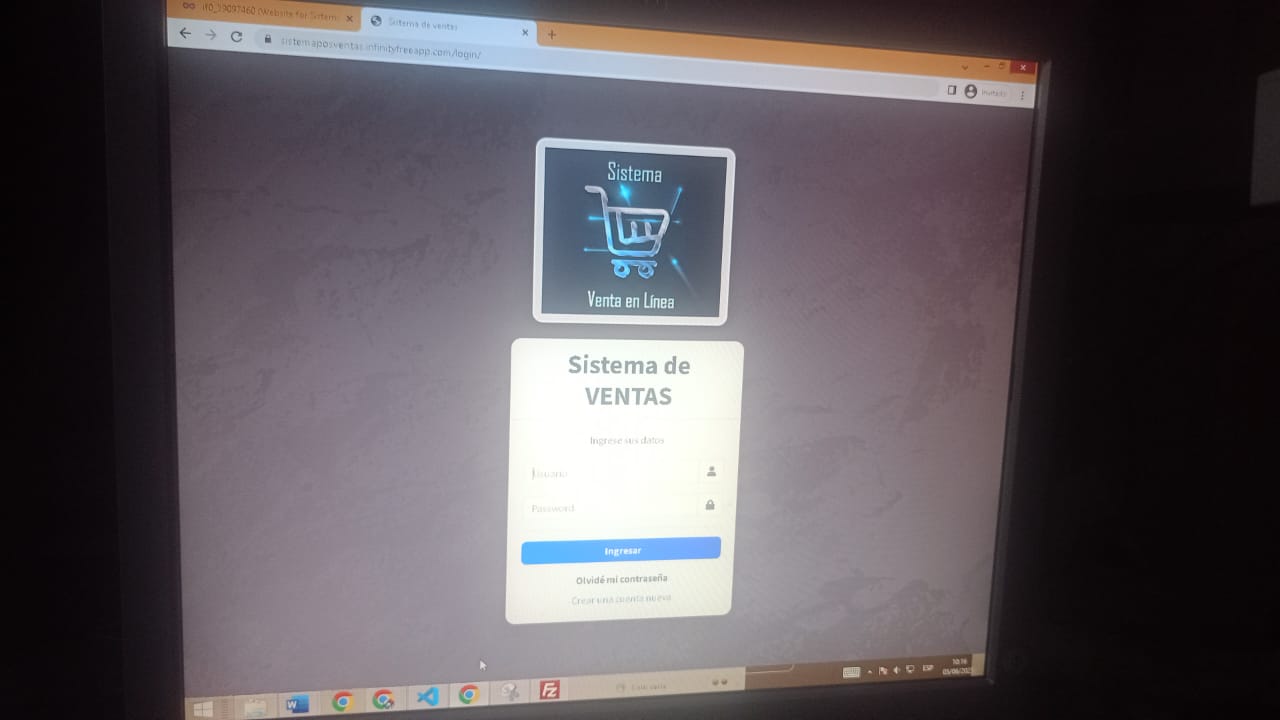
 Anexo 3: se implementó la estructura /app/controllers/ e index como vista

Anexo 4: Terminado todo se implementó con la subida del proyecto a un hosting gratuito



Anexo 5: Se instaló el cliente FTP FileZilla para facilitar la transferencia de archivos del sistema de ventas hacia el servicio de hosting gratuito InfinityFree.



Anexo 6: Se comprobó el funcionamiento del sistema en la web