



**UNIVERSIDAD POLITÉCNICA SALESIANA
SEDE QUITO**

CARRERA DE COMPUTACIÓN

**DESARROLLO E IMPLEMENTACIÓN DE UNA PLATAFORMA MODULAR
PARA LA OPTIMIZACIÓN DE PROCESOS ADMINISTRATIVOS DE UNA TIENDA
EN LÍNEA**

Trabajo de titulación previo a la obtención del
Título de Ingeniera en Ciencias de la Computación.

AUTORA: SAMI GABRIELA ALDAZ CABRERA

TUTOR: RODRIGO EFRAÍN TUFIÑO CÁRDENAS

Quito – Ecuador
2026

CERTIFICADO DE RESPONSABILIDAD Y AUTORÍA DEL TRABAJO DE TITULACIÓN

Yo, Sami Gabriela Aldaz Cabrera con documento de identificación N.º 1756136170 manifiesto que:

Soy La autora y responsable del presente trabajo; y, autorizo a que sin fines de lucro la Universidad Politécnica Salesiana pueda usar, difundir, reproducir o publicar de manera total o parcial el presente trabajo de titulación.

Quito, 03 de marzo de 2026

Atentamente,



Sami Gabriela Aldaz Cabrera
1756136170

CERTIFICADO DE CESIÓN DE DERECHOS DE AUTOR DEL TRABAJO DE TITULACIÓN A LA UNIVERSIDAD POLITÉCNICA SALESIANA

Yo, Sami Gabriela Aldaz Cabrera con documento de identificación N.º 1756136170, expreso mi voluntad y por medio del presente documento cedo a la Universidad Politécnica Salesiana la titularidad sobre los derechos patrimoniales en virtud de que soy autora del Proyecto Técnico: “Desarrollo e implementación de una plataforma modular para la optimización de procesos administrativos de una tienda en línea”, el cuál ha sido desarrollado para optar por el título de: Ingeniera en Ciencias de la Computación, en la Universidad Politécnica Salesiana, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente.

En concordancia con lo manifestado, suscribo este documento en el momento que hago la entrega del trabajo final en formato digital a la Biblioteca de la Universidad Politécnica Salesiana.

Quito, 03 de marzo de 2026

Atentamente,



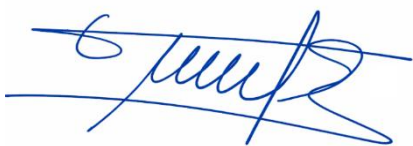
Sami Gabriela Aldaz Cabrera
1756136170

CERTIFICADO DE DIRECCIÓN DEL TRABAJO DE TITULACIÓN

Yo, Rodrigo Efraín Tufiño Cárdenas con documento de identificación N.º 1717646390, docente de la Universidad Politécnica Salesiana, declaro que bajo mi tutoría fue desarrollado el trabajo de titulación: DESARROLLO E IMPLEMENTACIÓN DE UNA PLATAFORMA MODULAR PARA LA OPTIMIZACIÓN DE PROCESOS ADMINISTRATIVOS DE UNA TIENDA EN LÍNEA, realizado por Sami Gabriela Aldaz Cabrera con documento de identificación N.º 1756136170, obteniendo como resultado final el trabajo de titulación bajo la opción Proyecto Técnico que cumple con todos los requisitos determinados por la Universidad Politécnica Salesiana.

Quito, 03 de marzo de 2026

Atentamente,



Ing. Rodrigo Efraín Tufiño Cárdenas, MSc.
1717646390

DEDICATORIA

Dedico este trabajo, con todo mi corazón, a mi familia.

Principalmente a mi papá Fabricio, por ser mi apoyo incondicional y por ser mi pilar y permitirme experimentar esta bonita etapa, la universidad, por todas las enseñanzas y aprendizajes. Gracias por confiar siempre en mí, por alentarme en cada paso, por ayudarme a desenvolverme sola y enfrentar cada reto por más difícil que este sea. Aunque a veces cometí errores, siempre estuvo a mi lado, celebrando mis logros y enseñándome a aprender de mis equivocaciones.

A mi mamá Sarita, por ser mi mejor amiga, por motivarme a levantarme cada día y seguir adelante e impulsarme a vivir nuevas experiencias, algunas que nunca imaginé, por animarme en mis mejores y peores momentos, porque siempre ha estado para mí, por ser mi cómplice y sobre todo por el amor que sé que me tiene.

A mi hermano Fernando, quien próximamente vivirá y disfrutará de una nueva experiencia, la universidad. Gracias por apoyarme cada día; te espera algo muy bonito y amistades que durarán para siempre, aunque a veces peleamos, siempre estamos juntos y contamos uno con el otro, a pesar de todo siempre disfrutaremos y nos divertiremos juntos.

A mi hermana Valeria, la más pequeña de la casa, quien cada día me recordó que, aunque sea universitaria, no dejó que me olvidará que pro dentro suelo ser una niña. Compartir tiempo con ella es disfrutar momentos únicos e irrepetibles.

Dedico también este trabajo a alguien muy especial que, aunque ya no esté presente y no pudo verme alcanzar esta meta, sé que está orgulloso de mí y me acompaña siempre.

Cada uno de ustedes han sido parte fundamental de este logro. Tengo a los mejores padres y hermanos del mundo, porque tienen un amor incondicional, me comprenden y constantemente me apoyan en los momentos difíciles.

Sami Gabriela Aldaz Cabrera

AGRADECIMIENTO

A mi familia, por ser mi pilar incondicional, por su apoyo constante, su amor y por acompañarme en cada etapa de este camino, incluso en los momentos más difíciles.

A mis amigos, que a veces están un poco locos, pero que hicieron de estos semestres una experiencia inolvidable, llena de risas, escapadas y momentos únicos que siempre llevaré en el corazón.

Agradezco infinitamente a todas las personas que conocí y sigo conociendo a lo largo de mi vida, especialmente, durante mi carrera universitaria. Con cada una de ellas voy aprendiendo, riendo, viviendo momentos y experiencias que marcan mi formación personal y profesional. A quienes hoy ya no están cerca por cualquier circunstancia de la vida o por el destino, les agradezco de igual manera, porque los bonitos recuerdos permanecerán hoy y siempre.

Agradezco de manera muy especial a una persona muy importante para mí, quien me enseñó que la universidad no solo es exámenes, estudios, deberes y proyectos, sino también un espacio para vivir momentos únicos, divertidos y especiales junto a las personas que más quieres.

Finalmente, a mi tutor por su apoyo constante, por su guía y por demostrarme que en la universidad existen docentes que creen en sus estudiantes, los alientan e impulsan a seguir adelante. Haber iniciado este camino con él desde el primer semestre y culminarlo teniéndolo como tutor en este último semestre refleja su compromiso, dedicación y el profundo amor que siente por la enseñanza.

Sami Gabriela Aldaz Cabrera

ÍNDICE DE CONTENIDOS

CAPÍTULO I

<i>ANTECEDENTES Y GENERALIDADES</i>	<i>1</i>
1.1. Introducción	1
1.2. Problema De Estudio.....	1
1.3. Antecedentes.....	2
1.4. Justificación.....	3
1.5. Objetivos	4
1.5.1. Objetivo General	4
1.5.2. Objetivos Específicos.....	4

CAPÍTULO II

<i>MARCO TEÓRICO</i>	<i>5</i>
2.1. Aplicaciones Modulares	5
2.1.1. Tipos de Aplicaciones Modulares	5
2.2. Comercio Electrónico.....	6
2.2.1. Clasificación Modelos de Negocio.....	6
2.2.2. Tienda en Línea	7
2.2.3. LEMP sobre Ubuntu Server 24.04 LTS.....	7
2.2.4. Nginx como Servidor Web	8
2.2.5. MariaDB.....	8
2.3. Entornos Virtuales con Python	9
2.4. WordPress.....	10
2.5. WooCommerce	10
2.6. API REST	10
2.7. Python.....	11
2.8. PySide6.....	11
2.9. Qt Designer	12
2.10. Requests	12
2.11. OpenPyXL.....	13

2.12. XlsxWriter	13
2.13. GitHub	14
<i>CAPITULO III</i>	
<i>METODOLOGÍA</i>	15
3.1. Tipo de Metodología de Investigación	15
3.2. Técnicas de Recolección de Datos	15
3.3. Fases Metodológicas	16
3.3.1. Fase 1: Análisis de procesos y requerimientos técnicos	16
3.3.2. Fase 2: Configuración del entorno de pruebas	16
3.3.3. Fase 3: Desarrollo ágil de módulos funcionales (Scrum).....	17
3.3.4. Fase 4: Integración del sistema	18
3.3.5. Fase 5: Pruebas funcionales en entorno simulado (Scrum).....	18
<i>CAPITULO IV</i>	
<i>RESULTADOS</i>	19
4.1. Análisis	19
4.1.1. Propósito.....	19
4.1.2. Alcance	19
4.1.3. Perspectiva del Sistema	21
4.1.4. Usuario Operador de la Tienda.....	21
4.1.5. Restricciones Generales	21
4.1.6. Suposiciones y Dependencias	22
4.1.7. Requerimientos Funcionales (RF).....	22
4.1.8. Requerimientos No Funcionales (NFR)	25
4.1.9. Requerimientos de Dominio (DR)	26
4.2. Diseño.....	26
4.2.1. Diagrama de Despliegue	27
4.2.2. Funcionalidades del Sistema	28
4.2.3. Wireframes	33
4.3. Desarrollo	37
4.3.1. Sprint 1: Análisis de procesos y requerimientos técnicos	38

4.3.2. Sprint 2: Configuración del entorno de pruebas.....	38
4.3.3. Sprint 3: Integración del sistema	39
4.3.4. Sprint 4: Desarrollo ágil de módulos funcionales	41
4.3.5. Sprint 5: Pruebas funcionales en entorno simulado	46
CONCLUSIONES	57
RECOMENDACIONES	58
REFERENCIAS	59

ÍNDICE DE TABLAS

Tabla 1 Definiciones, Acrónimos y Abreviaturas.....	20
Tabla 2 Restricciones Generales.....	22
Tabla 3 Requerimientos Funcionales Módulo de Reporte Ventas.....	23
Tabla 4 Requerimientos Funcionales Módulo de Inventario.....	23
Tabla 5 Requerimientos Funcionales Módulo de Actualización Productos	24
Tabla 6 Requerimientos Funcionales Módulo Lista de Distribuidores	24
Tabla 7 Requerimientos No Funcionales.....	25
Tabla 8 Requerimientos de Dominio	26

ÍNDICE DE FIGURAS

Figura 1 Diagrama Actualización de Productos.....	27
Figura 2 Diagrama de Despliegue.....	28
Figura 3 Funcionalidad de PyWoo	29
Figura 4 Funcionalidad Ingreso de Credenciales	29
Figura 5 Funcionalidad Aplicación Conexión con la Tienda en Línea	30
Figura 6 Funcionalidad Menú	30
Figura 7 Funcionalidad de Reporte Ventas	31
Figura 8 Funcionalidad de Actualización de Productos	31
Figura 9 Funcionalidad de Inventario	32
Figura 10 Funcionalidad de Lista de Distribuidores	32
Figura 11 Wireframe Menú	33
Figura 12 Wireframe Reporte Ventas	34
Figura 13 Wireframe de Actualización de Productos.....	35
Figura 14 Wireframe de Inventario	36
Figura 15 Wireframe Lista de Distribuidores.....	37
Figura 16 <i>Entorno de Pruebas - Moda F&S Estilo</i>	39

Figura 17 Arquitectura General del Sistema PyWoo.....	40
Figura 18 Configuración de API REST	42
Figura 19 Conexión con la API REST de WooCommerce.....	43
Figura 20 Módulo Menú Principal.....	43
Figura 21 Módulo Reporte de Ventas.....	44
Figura 22 Módulo Inventario	44
Figura 23 Módulo Actualizar Productos	45
Figura 24 Módulo Lista Distribuidores.....	45
Figura 25 Tienda en Línea Moda F&S Estilo	47
Figura 26 Ingreso de Credenciales	48
Figura 27 Menú Principal del Sistema	49
Figura 28 Módulo de Reporte Ventas.....	50
Figura 29 Archivo Exportado de Reporte Ventas.....	50
Figura 30 Formato Archivo de Actualización de Productos	51
Figura 31 Producto No Actualizado	51
Figura 32 Módulo Actualización de Productos	52
Figura 33 Producto Actualizado.....	52

Figura 34 Archivo Exportado Actualizar Productos	53
Figura 35 Módulo de Inventario	54
Figura 36 Archivo Exportado Inventario	54
Figura 37 Módulo Lista de Distribuidores	55
Figura 38 Archivo exportado Lista de Distribuidores	56
Figura 39 Usuario Ejcutando PyWoo.....	56

RESUMEN

El objetivo de este proyecto de titulación fue diseñar e implementar una plataforma modular para optimizar los procesos administrativos de una tienda online basada en WooCommerce. Se trata la problemática de los negocios que tienen tiendas en líneas, se gestionan los procesos administrativos como los reportes de ventas, la actualización de productos, el inventario, lista de productos para distribuidores según scripts técnicos independientes, lo que hace que haya una gran dependencia del personal especializado e impide la participación de los usuarios que no tienen conocimientos técnicos.

La solución presentada en este trabajo, la denominada PyWoo, es una aplicación de escritorio desarrollada en lenguaje Python y PySide6; en la cual se ha implementado la integración con Woo Commerce utilizando la API REST; centralizándose en módulos funcionales los procesos administrativos, diseñados con uso y modularidad, que permiten la automatización segura y eficiente de las críticas tareas de negocio. El sistema fue desarrollado mediante una metodología mixta que combina el modelo en cascada para las fases de análisis y planificación y el marco ágil Scrum para las fases de desarrollo, pruebas e implementación.

El sistema fue validado mediante un entorno de prueba al que se le asignó una tienda online real. Con esto se pudo verificar la interoperabilidad con WooCommerce, reducir los errores operativos y mejorar la eficiencia de la ejecución de tareas administrativas. Los resultados muestran que PyWoo facilita las tareas administrativas, reduce la dependencia de usuarios técnicos y es una herramienta intuitiva, escalable y de código abierto, adecuada para microempresas y pequeñas empresas de comercio electrónico.

Palabras clave: WordPress, WooCommerce, tienda en línea, automatización, modular.

ABSTRACT

The objective of this degree project was to design and implement a modular platform to optimize the administrative processes of an online store based on WooCommerce. It addresses the problem faced by businesses with online stores, where administrative processes such as sales reports, product updates, inventory, and product lists for distributors are managed according to independent technical scripts, which creates a heavy reliance on specialized personnel and prevents the participation of users who do not have technical knowledge.

The solution presented in this work, called PyWoo, is a desktop application developed in Python and PySide6 language, in which integration with WooCommerce has been implemented using the REST API. It centralizes administrative processes in functional modules, designed for ease of use and modularity, allowing for the secure and efficient automation of critical business tasks. The system was developed using a mixed methodology that combines the waterfall model for the analysis and planning phases and the agile Scrum framework for the development, testing, and implementation phases.

The system was validated using a test environment assigned to a real online store. This made it possible to verify interoperability with WooCommerce, reduce operational errors, and improve the efficiency of administrative task execution. The results show that PyWoo facilitates administrative tasks, reduces dependence on technical users, and is an intuitive, scalable, open-source tool suitable for micro and small e-commerce businesses.

Keywords: WordPress, WooCommerce, online store, automation, modular.

CAPÍTULO I

ANTECEDENTES Y GENERALIDADES

1.1. Introducción

En este capítulo se abordan los antecedentes y la justificación para el desarrollo y la implementación de una solución tecnológica, con el fin de optimizar procesos administrativos en una tienda en línea. En AV Electronics, se identifican las limitaciones operativas de los procesos, dependientes de una persona técnica. La problemática justifica la necesidad de una plataforma modular que simplifique la gestión administrativa, mejore la eficiencia operativa y permita la participación de usuarios técnicos y no técnicos.

1.2. Problema De Estudio

AV Electronics es una empresa dedicada a comercializar productos electrónicos a través de e-commerce basada en la plataforma WooCommerce, su principal forma de ventas. En la empresa se gestionan distintos procesos administrativos, se generan reportes de ventas, se actualizan los productos, se gestiona el inventario y la lista de distribuidores mediante scripts independientes. Si bien estas soluciones permiten evitar la ejecución completamente manual de las tareas de la tienda, su implementación se caracteriza por una alta dificultad técnica, lo que ocasiona una escasa adaptación de procesos y una fuerte necesidad de una persona técnica que se encargue de la operación.

Esta situación limita la ejecución eficiente de los procesos administrativos, impidiendo que otros usuarios se encarguen de la gestión de la tienda en línea. Como consecuencia, las actividades del negocio se vuelven poco intuitivas, afectando directamente el rendimiento operativo de la empresa. La falta de interfaz intuitiva y los scripts dispersos provocan un uso ineficiente del tiempo, dificulta la lectura de la información y aumenta la probabilidad de errores derivados de ejecuciones o configuraciones incorrectas. Esto limita la facilidad de realizar cambios en el inventario, el análisis de las ventas a los usuarios, entre otros.

El problema encontrado no está relacionado con la falta de automatización, sino con la falta de una plataforma modular, que se encuentre orientada al usuario, para no depender de un

usuario técnico y permitir la simplificación de la ejecución de los procesos administrativos. En este contexto, surge la necesidad de una solución que mejore la operación y la plataforma pueda ser utilizada por usuarios técnicos y no técnicos, de esa manera, se garantiza la continuidad del negocio, reducir el tiempo de ejecución y se gestiona de una forma más accesible e intuitiva.

Investigaciones similares como las de (Guandinango & Guamanzara, 2021), quienes desarrollaron una solución libre para automatizar el proceso de importación de productos y sincronización con WooCommerce, evidencian la urgencia de estas herramientas en entornos comerciales reales. De igual forma, (Moyón, 2022) propuso una aplicación web de facturación electrónica integrada a WooCommerce para mejorar la gestión tributaria en tiendas ecuatorianas.

1.3. Antecedentes

El crecimiento del comercio electrónico ha llevado, entre otras, a empresas del tipo AV Electronics a introducir plataformas para llevar a cabo la integración de sus operaciones comerciales de forma flexible y a coste bajo. En este sentido, WooCommerce, y su relación con WordPress, cumple el objetivo de una de las plataformas más utilizadas, así como por su capacidad de adaptación a distintas tipologías de negocio. Sin embargo, su uso habitual exige múltiples procesos que, en muchos casos, no están completamente desarrollados ni integrados.

La automatización de tareas administrativas ha sido resuelta mediante scripts independientes diseñados para resolver necesidades específicas en la tienda online, pero estas no estaban pensadas para el uso de personas sin capacidad técnica, lo que limita su utilización y no favorece su integración entre procesos administrativos. Esto evidencia una solución del tipo técnica a líneas de procesos de ejecución, priorizando la ejecución funcional a la del usuario final.

Existen diferentes estudios en relación a la transformación digital por medio de automatización de las microempresas que nos indican que la automatización poco "acompañada" por interfaces de usuario intuitivas y arquitecturas modulares que ayuden tanto en el mantenimiento como en su uso, pueden llegar a ser un cuello de botella. De igual forma, los estudios académicos indican que la centralización de los procesos administrativos mediante sistemas modulares puede llegar a mejorar la eficiencia operativa, disminuir los errores e

incluso facilitar la capacitación del personal, independientemente de que este formado en técnica o no.

Aunque la automatización permite resolver problemas de una forma muy rápida, y menos eficiente a la larga gracias a la concentración de la gestión administrativa del personal técnico que la abren una brecha para los usuarios que no poseen dicha técnica. La falta de una interfaz amigable suma el hecho de no tener flujos de trabajo conocidos, y esto supone un hecho limitante del crecimiento organizacional y limitante del uso de tecnología en los microempresas. La utilización de aplicaciones modulares que huyen del usuario permiten proseguir con la gestión administrativa, mejoran la eficiencia operativa y minimizan la posibilidad de errores.

1.4. Justificación

La utilización de una aplicación modular orientada a la administración del comercio electrónico se presenta como respuesta a una necesidad cada vez más importante, la de poder implementar un proceso de gestión administrativa que simplifique y optimice el proceso de funcionamiento del negocio. La empresa, aunque dispone de la posibilidad de automatizar la ejecución de tareas, las tiene implementadas mediante la utilización de script técnicos independientes por lo que representa una carga para el responsable de la tienda en línea y también supone un descenso en la autonomía para llevar a cabo la gestión del negocio por parte de la persona que tiene que realizar dichas tareas al depender muchas veces de personas con la suficiente experiencia para poner en marcha esas pequeñas automatizaciones.

El poder implementar PyWoo hace posible que se integren procesos administrativos como la generación de informes de ventas, en la actualización de productos, en el control de inventarios o en la administración de los distribuidores gracias a una aplicación única estructurada en módulos funcionales que permite la ejecución de tareas de forma centralizada y simplificada, evita el tiempo destinado a tareas repetitivas y reduce la posibilidad de errores cometidos por la ejecución de procesos de forma aislada, de esta forma se favorece la idea de poder ser un mismo usuario el que pueda ejercer tareas administrativas que tareas operativas y redistribuir el trabajo asumiendo un menor porcentaje de dependencia de usuarios técnicos.

En el proyecto, en la parte académica incorpora conceptos fundamentales relacionados con la integración de sistemas, la automatización de procesos, el diseño de arquitecturas modulares y la aplicación de principios de la ingeniería de software. PyWoo al integrarse con API REST de WooCommerce responde a la problemática, ya que se automatizan procesos de una forma eficiente y sin necesidad de ingresar a la plataforma de WordPress, mejorando totalmente la parte administrativa como el incluir la tecnología en el comercio electrónico.

1.5. Objetivos

1.5.1. Objetivo General

Diseñar e implementar una plataforma tecnológica modular que automatice procesos administrativos en una tienda en línea basada en WooCommerce, mediante el desarrollo de módulos funcionales orientados a la gestión eficiente de productos, inventario y ventas, con el fin de reducir tareas manuales y facilitar su uso por parte de usuarios sin conocimientos técnicos.

1.5.2. Objetivos Específicos

Analizar los procesos administrativos de una tienda en línea para identificar oportunidades de optimización.

Diseñar y desarrollar módulos de automatización para el registro de productos, gestión de inventario y reporte de ventas, accesibles a usuarios sin conocimientos técnicos.

Integrar los módulos desarrollados con la plataforma WooCommerce, garantizando la interoperabilidad y coherencia del sistema.

Evaluar y desplegar la solución en un entorno real conectado a WooCommerce, verificando su funcionamiento mediante pruebas representativas.

CAPÍTULO II

MARCO TEÓRICO

El capítulo presenta los principios conceptuales y las tecnologías que fundamentan el diseño y desarrollo del sistema PyWoo, dirigido a la mejora de los procesos administrativos de una tienda en línea. La solución que propone se fundamenta en una arquitectura modular de software que entrelaza tecnologías propias del comercio electrónico, herramientas de desarrollo en Python, componentes para la gestión de entornos virtualizados y componentes del control de versiones. Pero también se analizan los modelos de comercio electrónico y los modelos de aplicación en microempresas y empresas pequeñas que en la mayor parte de los casos no tienen personal técnico especializado y en el caso general tienen una dependencia tecnológica, lo que lleva a una baja eficiencia operativa.

2.1. Aplicaciones Modulares

Según Sommerville (2016) las aplicaciones modulares son programas se que dividen en distintas partes, cada una de esas partes hace algo, si llega a caerse un módulo al ser divididos el otro sigue funcionando. Agrupa toda la lógica en un grupo de código, pero al ser dividido permite que cada módulo se desarrolla, se implementa, se prueba, se mantiene de forma independiente, pero contribuye a todo el sistema como tal. En el programa se utiliza aplicaciones modulares para que el programa funcione para distintos procesos, pero con instrucciones específicas para cada módulo reduciendo la complejidad.

2.1.1. Tipos de Aplicaciones Modulares

Según Sommerville (2016) existen distintas aplicaciones, aplicaciones en computadora, en internet a nivel web, en el celular, cada una para un fin específico, en el caso de aplicaciones en computadora se utilizan más en procesos administrativos, al ser de escritorio ofrece un mayor control al momento de ejecutar el programa y una experiencia de usuario más estable. Las aplicaciones web, son sistemas que permiten el acceso remoto y se utilizan desde cualquier navegador, lo que si hace uso de internet. Las aplicaciones móviles priorizan el acceso inmediato y la portabilidad desde dispositivos como teléfonos o tablets. El programa se hace

uso de aplicaciones de escritorio porque es para un fin administrativo que necesita ser ejecutado en ese momento y para el usuario sea intuitivo de utilizar.

2.2. Comercio Electrónico

Según Vásquez (2025) el comercio electrónico corresponde a la compra y venta de servicios o productos desde plataformas web, sin necesidad de ir a una tienda física. Dentro de la plataforma se permite la integración de varios procesos como ventas, pagos, facturación, marketing y el análisis de datos todo desde una sola aplicación como internet. El comercio electrónico ha crecido en una gran medida, solo con la necesidad de internet se realizan compras desde dispositivos de móviles o equipos

El comercio electrónico o e-commerce, corresponde a la generación de transacciones comerciales en el ámbito digital, especialmente en Internet. Vásquez (2025) define el comercio electrónico como la actividad de ejecutar transacciones comerciales por medio de la Web, de Internet y de las aplicaciones de los dispositivos móviles. Dicho modelo permite la integración de procesos de ventas, pagos, logística, marketing y análisis de datos dentro de una única plataforma digital.

La gestión administrativa genera gran cantidad de información en el comercio electrónico e información que ha de ser tratada y gestionada de forma eficaz y eficiente. Las plataformas que implementan comercio electrónico suelen proporcionar interfaces administrativas, pero no siempre se adaptan a las necesidades de tipo específico de negocio, llegando a propiciar el desarrollo de soluciones externas mediante el uso de scripts o herramientas informáticas aisladas, lo que provoca una fragmentación, una mayor complejidad operativa, una dependencia de técnico especializado y la necesidad de contar con soluciones que centralicen y simplifiquen la gestión administrativa, garantizando la correcta integridad de los datos.

2.2.1. Clasificación Modelos de Negocio

El comercio electrónico se puede clasificar siguiendo la relación entre los actores que intervienen en la transacción. Entre los modelos más generalizados podemos encontrar negocio a consumidor, negocio a negocio, consumidor a consumidor, consumidor a negocio y gobierno

a ciudadano. Cada uno de estos modelos presenta unas características propias que serán bien diferentes entre sí, tales como los volúmenes de transacciones que les son propios, el tipo de usuarios o la complejidad operativa.

Las micro y pequeñas empresas suelen trabajar con bastante frecuencia bajo una estructura B2C B2B, que mezcla las ventas directas hacia el consumidor real y las ventas gestionando distribuidores o mayoristas. Este alto grado de complejidad administrativa es porque requiere implementar diferentes esquemas de costos, inventario y reportes. La correcta identificación del modelo de negocio es crucial para diseñar sistemas que sean coherentes y que sirvan a los objetivos internos de la empresa y a su mejora progresiva.

2.2.2. Tienda en Línea

La tienda en línea es una página web a la que se puede acceder en cualquier momento y en cualquier sitio. Dicha disponibilidad es la que les permite tener acceso a un número mayor de clientes porque tampoco necesitan disponer de una tienda física donde los clientes puedan realizar sus compras. Los comerciantes de comercio electrónico pueden vender a un público global creando un espacio de mercado que se extiende más allá de las fronteras tradicionales (Gonzales & Herrera, 2025). Estas organizaciones para comercializar bienes o servicios disminuyen gastos innecesarios por tener que hacerse cargo de un local físico, no tienen restricción de horarios ya que funcionan las veinticuatro horas del día, los siete días de la semana, presentan una gama de bienes o de servicios comercializables. Los comerciantes construyen estos tipos de negocios para que se adapten a cualquier dispositivo.

2.2.3. LEMP sobre Ubuntu Server 24.04 LTS

La distribución GNU / Linux Ubuntu Server 24.04 LTS tiene un servidor, y se destaca por servidores hecho a medida para satisfacer las necesidades de entorno de servidor, las cuales deben ser: estabilidad, seguridad y soporte en el tiempo. Las distribuciones de Linux permiten un alto grado de personalización del entorno de servidor de forma tal que puede adaptarse de forma sencilla a los requerimientos concretos de cualquier comercio electrónico web.

La arquitectura de tipo LEMP (Linux, Nginx, MariaDB y PHP) permite un uso altamente optimizado del servidor y mantener un rendimiento adecuado de los servicios web

en funcionamiento permanente. Según DigitalOcean (2025), Ubuntu 24.04 LTS cuenta con cinco años de soporte oficial que se pueden extender a un máximo de doce años con Ubuntu Pro de forma que se asegura suficiente soporte mediante actualizaciones de seguridad y mantenimiento en el tiempo de la compatibilidad tecnológica del software a lo largo del ciclo de vida del proyecto.

2.2.4. Nginx como Servidor Web

Es un servidor web que permite administrar la gran cantidad de datos, proporciona tiempos de respuesta más eficientes y veloces gracias a que permite gestionar conexiones concurrentes más eficientemente. Cuando hay múltiples usuarios conectados a la vez, la página no tiende a colapsar y por esta razón seguirá funcionando. "NGINX utiliza una arquitectura asincrónica, no bloqueante y de un solo subproceso, impulsada por eventos la cual puede gestionar millas de conexiones simultáneas con un bajo consumo de memoria " (NGINX, 2025). Dado que se basa en una arquitectura de este tipo, se conseguirá activar WordPress en la medida de que se ejecuten todos los datos que requieren dicha aplicación. Nginx es un servidor web capaz de gestionar, de forma eficiente, múltiples conexiones simultáneas. Su arquitectura asincrónica y orientada a eventos permite gestionar grandes volúmenes de solicitudes al mismo tiempo y con un consumo bajo de memoria, por lo que se puede afirmar que es adecuado para aplicaciones web de gran concurrencia.

2.2.5. MariaDB

MariaDB es un sistema de gestión de bases de datos relacional de código abierto, derivado de MySQL, que se utiliza ampliamente en aplicaciones web y plataformas de comercio electrónico. Cuando se produce totalmente la compatibilidad con MySQL, prácticamente la totalidad de aplicaciones que se han diseñado para este sistema pueden funcionar por encima de MariaDB sin que su lógica de acceso a datos tenga que ser alterada. Las características más destacadas de MariaDB son la libertad de elegir motores de almacenamiento, la optimización del rendimiento y la gestión eficaz de grandes cantidades de datos, lo cual la convierte en la solución a escoger en una aplicación como WooCommerce, donde la integridad en la e información y la disponibilidad de esta son todo un reto. (MariaDB Foundation, 2025).

La (MariaDB Foundation, 2025) la versión 10.11 es una edición de tipo LTS que constituye una evolución del motor además de tener mecanismos que proporcionan cifrado a los datos en reposo. Se trata de software libre por lo que no habrá costes asociados a las licencias repartidas por los fabricantes y, además, se garantizará en todo momento la disponibilidad de esta información, así como la consistencia de la misma.

PHP es un lenguaje de programación interpretado de propósito general que se ejecuta del lado del servidor y está especialmente diseñado para el desarrollo de aplicaciones web dinámicas y generación de contenido en tiempo real. La ejecución eficiente de scripts PHP se realiza mediante PHP-FPM, el cual se encarga de crear, reutilizar y administrar procesos, optimizando el uso de recursos en servidores web como Nginx. (PHP Group, 2025).

En el ámbito del comercio electrónico, WooCommerce es una extensión construida sobre WordPress que, en su arquitectura, está directamente ligada a PHP para funcionar. Las operaciones que el sistema ejecuta de forma dinámica: cargar productos en la tienda, procesar el pedido y comunicarse con la base de datos, etc. son todas realizadas mediante código PHP en el servidor. Según la documentación técnica oficial de WooCommerce indica que PHP constituye la tecnología principal para la ejecución del núcleo del sistema y la integración de extensiones, permitiendo que la plataforma responda de forma dinámica y segura a las solicitudes de los usuarios dentro de un entorno web basado en WordPress (Raicem, 2025).

2.3. Entornos Virtuales con Python

La idea que se persigue al utilizar entornos de trabajo también es la de aislar las librerías que únicamente son necesarias para un determinado proyecto y eso equivale a una copia y un árbol de directorios limpios de un nuevo entorno dentro de la misma instalación de python; “Un entorno virtual es, para nosotros, un árbol de directorios independientes que contiene la instalación de python para una determinada versión del lenguaje y la selección de paquetes adicionales” (Python Software Foundation, 2025). Lo necesario para crear el entorno es `venv` y lo principal es activarlo. Se debe tomar en cuenta que se debe crear el archivo *requirements.txt* con todas las librerías y versiones necesarias para el funcionamiento del proyecto. Los entornos virtuales en Python permiten aislar las dependencias específicas de un proyecto, evitando conflictos entre librerías y versiones del lenguaje. Esta práctica resulta fundamental para garantizar la reproducibilidad y estabilidad del entorno de desarrollo.

El entorno virtual se define como un directorio independiente que contiene una instalación de python para una versión del lenguaje determinado y con los paquetes que son necesarios para el proyecto (Python, sf). En el desarrollo del sistema PyWoo se utilizó el módulo venv y con un archivo requerimientos.txt para documentar las dependencias empleadas.

2.4. WordPress

WordPress es una herramienta en línea diseñada específicamente para la creación de tiendas virtuales y para gestionar todo el contenido. Su estructura es sencilla, pudiendo ir desde las páginas de inicio a las diferentes formas de agregar productos. “El editor de bloques de WordPress le permite insertar, reorganizar y diseñar contenido multimedia con muy pocos conocimientos técnicos” (WordPress.org, 2025). La plataforma permite usar su API REST, que permite trabajar en la parte externa, dado que con tener las funcionalidades por medio de complementos hace que numerosos procesos no sean tan manuales y cambiar los temas

2.5. WooCommerce

WooCommerce es un complemento de comercio electrónico de WordPress. Tiene la capacidad de gestionar la creación de inventarios, productos, pedidos, facturación, a través de diferentes procesos "WooCommerce añade funcionalidad de comercio electrónico a WordPress utilizando tipos de entradas personalizadas y almacenando los datos de pedidos en tablas estándar de WordPress o tablas personalizadas" (WooCommerce, 2025). Es un motor de ventas que se considera no tan automático pero que permite realizar compras a través de redes digitales.

2.6. API REST

Una interfaz de programación de aplicaciones de estilo REST es una arquitectura para la comunicación entre sistemas que se hace por medio de solicitudes HTTP para acceder a los recursos identificados mediante URLs utilizando operaciones estándar como la de GET, POST, PUT o DELETE.

Según la documentación oficial de WooCommerce, " La REST API de WooCommerce permite leer y escribir diferentes partes de datos de WooCommerce como productos, pedidos, clientes y mucho más. " (WooCommerce, 2025). Esta API REST se accede a través de endpoints

que comienzan con el prefijo wp-json/wc/v3 y que requiere autenticación a través de clave de consumidor y clave secreta generados en la consola de configuración del propio WooCommerce.

2.7. Python

Python es un lenguaje de programación de alto nivel que tiene una sintaxis clara, una buena legibilidad y un amplio ecosistema de librerías. Estas características lo convierten en una buena opción para el desarrollo de aplicaciones orientadas a la automatización, integración de sistemas y creación de interfaces de usuario. Python permite implementar soluciones complejas conservando un código comprensible, circunstancia que es muy importante sobre todo en los proyectos en los que se busca minimizar la dependencia del personal técnico con conocimientos especializados.

La facilidad de Python a la hora de trabajar con servicios web mediante APIs REST, la manipulación de estructuras de archivos y el desarrollo de aplicaciones multiplataforma de escritorio, junto con su comunidad muy activa y su documentación rigurosamente establecida favorecen su uso en el medio académico y profesional. Desde una perspectiva de ingeniería de software, contribuye a aumentar la productividad del proceso de desarrollo y a disminuir el número de errores vinculados a la complejidad del código.

2.8. PySide6

PySide6 es una librería para construir interfaces gráficas que permite programar aplicaciones en Python con un estado alternativo a través de interfaces gráficas. Al ser de código abierto no es necesario pagar por la licencia. Esta librería es el enfoque de Qt para quienes usan Python para construir una arquitectura que utilice Qt. “Qt para Python es una opción oficial para enlazarse a las API de Qt6 desde Python” (Qt Group, 2025) . Las aplicaciones de escritorio están indicadas para el personal administrativo de diversas empresas. Al ser portátil puede convertirse en un solo ejecutable para un usuario determinado sin requerir instalar otra aplicación.

2.9. Qt Designer

Qt Designer es una herramienta gráfica que hace parte del ecosistema Qt y que permite crear interfaces de usuario de forma visual, es decir, sin escribir líneas de código. Con esta herramienta es posible hacer ventanas, formularios y componentes de interacción arrastrables, haciendo que la construcción de interfaces gráficas coherentes y utilizables con el entorno Qt durante la codificación sea más rápida (Qt Python, s. f.) fue utilizado para diseñar las interfaces del sistema de escritorio, priorizando la claridad visual y la facilidad de uso.

Qt Designer se utiliza para diseñar las interfaces del sistema de escritorio considerando la facilidad visual y la facilidad de uso. La utilización de Qt Designer permite poder diferenciar el diseño visual de la lógica de programación y esto mejora la organización del código a la hora de programar y facilitar así futuras modificaciones de la interfaz. En ingeniería de software, separar la vista de la lógica de programación contribuye a la mejor mantenibilidad del sistema y mejora, así como la experiencia del usuario.

2.10. Requests

La librería Requests es una de las herramientas que más se utilizan en Python para realizar solicitudes HTTP de forma sencilla y legible. Requests abstrae la complejidad de conexión, encabezados y respuestas HTTP, permitiendo que el desarrollador se preocupe de la lógica del negocio. Requests soporta los métodos estándar GET, POST, PUT y DELETE, lo que la hace adecuada para interactuar con servicios web y APIs REST.

El uso de Requests mejora algo la estabilidad y la confiabilidad de la aplicación, dado que gestiona de forma adecuada las excepciones por error de conexión, los tiempos de espera y la validación de respuestas. Desde el punto de vista técnico, Requests permite hacer integraciones dignas de este nombre, pero sin agregar más dependencias complejas a la solución, de tal forma que hace más fácil los objetivos del proyecto de simplificar la parte administrativa de WooCommerce sin tener que someter al usuario final a la complejidad técnica implicada en este tipo de automatización.

2.11. OpenPyXL

El proceso de archivos estructurados es otro de los elementos que ayuda a automatizar la parte administrativa hacia el usuario final, dado que a través de OpenPyXL el sistema comprueba la estructura de los archivos que el usuario sube, revisa la existencia de algunos campos obligatorios como el SKU, y procesa los valores de precios y de stock, todo ello antes de aplicar los cambios sobre la tienda. Esta comprobación previa evita el origen de errores operativos y desgasta menos el almacenamiento de información en WooCommerce.

La utilización de estas herramientas permite a los usuarios no técnicos trabajar con formatos de salida de información habituales, como pueden ser las hojas de cálculo Excel, pero para nada interactuar con una base de datos o ejecutar scripts técnicos. Desde el punto de vista académico, la integración de Pandas y OpenPyXL es la demostración de una aplicación práctica de librerías que está plenamente consolidada científicamente, para resolver problemas prácticos que tienen que ver con el uso de la gestión administrativa.

2.12. XlsxWriter

La función de la exportación de datos es básica en la actualidad mayoría de sistemas de gestión administrativa, dado que permite aprovechar dicha información al poder ser utilizada por distintas aplicaciones. En el caso de gestión administrativa, a menudo los formatos CSV y XLSX constituyen normas o estándares de exportación por su potencial de ser leídas por la mayor parte de productos de software de uso personal, además de ser manipulables por usuarios con distintos niveles de conocimiento técnico(XlsxWriter, 2025). Por medio de los formatos de exportación simplemente se permite el intercambio de información estructurada sin la necesidad de programas concretos, lo que resulta de especial interés para la microempresa y la pequeña empresa.

La implementación de mecanismos de exportación permite que la información generada por el sistema pueda ser analizada posteriormente, archivada como respaldo administrativo o utilizada para la toma de decisiones estratégicas. El formato CSV permite transferir información tabulada debido a su simplicidad y ligereza, mientras que el formato XLSX aumenta dichas capacidades debido a que organiza la información en hojas de cálculo con un formato, estilo y estructuras más sofisticadas.

2.13. GitHub

GitHub es la herramienta que permite gestionar repositorios Git, así como, todo un conjunto de herramientas para la gestión de proyectos de software. Sin embargo, GitHub, más allá de ser un repositorio remoto, tiene características propias como el control de incidencias, la posibilidad de documentación con ficheros README y la visualización de los cambios en el histórico. De esta forma, GitHub se convierte en una herramienta muy utilizada en el ámbito académico y profesional por su versatilidad.

En el caso del proyecto PyWoo, se utilizó GitHub como repositorio giratoria central para gestionar el código fuente y la documentación técnica del sistema. Con esta herramienta se tuvo copias de seguridad integradas normalmente y se tuvo el progreso del desarrollo a lo largo del tiempo. Esta práctica es especialmente relevante en el proceso de un trabajo de titulación por el objetivo de mantener la transparencia, la trazabilidad y la reproducibilidad del proceso de desarrollo.

CAPITULO III

METODOLOGÍA

En esta sección se da cuenta de la metodología de trabajo que ofrece la oportunidad de estudiar la estructura, análisis, diseño, desarrollo e integración de una plataforma modular ejecutada mediante pruebas que permitan la conexión API REST y la cual está orientada a la mejora de procesos administrativos.

3.1. Tipo de Metodología de Investigación

La realización del proyecto en un entorno real en producción, favoreció el poder analizar procesos administrativos y las carencias consecuencia de la dependencia de usuarios sin conocimiento técnico que dependían del personal especializado. Con una buena lectura del análisis y a partir de este se diseñó una solución basada en una aplicación de trabajo que no compromete la integridad y el funcionamiento del sitio de tienda en línea.

La investigación de campo permitió analizar la ineficiencia provocada por la carencia de una aplicación administrativa conforme a los usuarios finales. A partir del análisis se diseñó una solución orientada a usuarios no técnicos, con la finalidad de reducir errores y tiempos de ejecución.

3.2. Técnicas de Recolección de Datos

Las técnicas que se utilizaron en este proyecto permitieron analizar la funcionalidad del sistema y la eficiencia operativa de la aplicación desarrollada. Se llevó a cabo el análisis de la estructura de WooCommerce como API REST y los scripts para la parte administrativa de la tienda. Para este caso el diseño del estudio tiene una relevancia y ni que comentar que resulta conveniente estudiar cómo es la ejecución de tareas antes y después de utilizar la aplicación modular.

3.3. Fases Metodológicas

La realización de la aplicación determinó una estructura a través de fases metodológicas, en las que se combinó el modelo cascada con Scrum para que la metodología fuera mucho más ágil y flexible. De esta forma el proyecto se desarrolló en una forma ordenada, tomando en consideración la evaluación y el análisis, el nivel de planificación y la propia realización de cada una de las fases.

Cada fase se definió en función de la ejecución progresiva del proyecto, es decir, desde el análisis inicial, hasta la validación definitiva de la solución; y se lograron desarrollar pruebas con la tienda en línea, para mejorar la parte administrativa.

3.3.1. Fase 1: Análisis de procesos y requerimientos técnicos

Esta primera fase, trató del análisis de cada uno de los procesos administrativos de la tienda, la definición de los requerimientos funcionales y no funcionales, tomando en cuenta la estructura del modelo cascada, esto permitió que se evidencie que, aunque existen procesos para ayudar en la parte administrativa carece de una interfaz intuitiva para mejor facilidad para el usuario.

Se tomó en cuenta las tecnologías necesarias para levantar el proyecto, se consideró integrar WooCommerce con su API REST, utilizando software libre y la implementación de GNU/Linux. Se tomó en cuenta la seguridad, el rendimiento, la estabilidad y mantenibilidad que debe tener la aplicación. Se asegura que el desarrollo de este proyecto esté alineado a las necesidades de la tienda.

3.3.2. Fase 2: Configuración del entorno de pruebas

En esta fase se empezó configurando un entorno de pruebas, diseñado para replicar el funcionamiento operativo de una tienda en línea real, con el objetivo de validar el desarrollo de la plataforma sin comprometer datos productivos ni interferir con la operación del negocio. Es importante aclarar que AV Electronics constituye el caso de estudio real del presente proyecto, mientras que la tienda Moda F&S Estilo fue utilizada exclusivamente como entorno de pruebas simulado, permitiendo realizar validaciones técnicas de manera segura.

Realizar la prueba del entorno de pruebas para el sistema, en el cual se desplegó un servidor virtualizado en la Universidad Politécnica Salesiana con Ubuntu Server fue el entorno utilizado. Donde se instalaron y configuraron los servicios necesarios para hacer funcionar la tienda online: Nginx como servidor web, PHP como lenguaje de ejecución del lado servidor, y MariaDB como un gestor de base de datos SQL. Así mismo se implementó WordPress junto al plugin WooCommerce, en el que se construyó la tienda de pruebas utilizada para verificar las pruebas funcionales del sistema.

En este entorno se replicó la estructura típica de una tienda online, incorporando productos simples y productos variables, configurando inventarios y simulando la función de realizar pedidos de compra, lo que permitió establecer un entorno de pruebas que representara el entorno real de la tienda online y validar así la comunicación entre la tienda y la API REST de WooCommerce. Con este entorno se pudieron verificar las pruebas de funcionalidad sobre los diferentes módulos de la plataforma, validando la consulta, actualización y el procesamiento de la información administrativa sin afectar el funcionamiento de la tienda de producción.

3.3.3. Fase 3: Desarrollo ágil de módulos funcionales (Scrum)

En esta fase se desarrollaron los módulos asociados a los diferentes procesos administrativos del sistema fueron realizados así mismo en esta fase, empleando el marco de trabajo SCRUM como metodología de trabajo en la fase de desarrollo. Se priorizaron funcionalidades tales como la actualización de productos, la gestión de inventario, la generación de informes de ventas o generación de listas de productos para los distribuidores A la vez. A la par, se realizó la validación de cada módulo, tanto a nivel de código como de interfaz gráfica, desarrollando siempre interfaces orientadas a los usuarios sin conocimientos técnicos, que permitieran disminuir la dependencia de tareas administradas por personal especializado y la posible ejecución de tales procesos administrativos. En esta fase se realizaron ajustes iterativos en la lógica de negocio y la experiencia de usuario, garantizando que existiera coherencia entre funcionalidad, usabilidad y los requisitos definidos como tales.

En línea con el enfoque ágil que se había aplicado, las funcionalidades desarrolladas en esta fase fueron verificadas igualmente a medida de su integración en el sistema PyWoo, lo que permitió comprobar su funcionamiento de forma conjunta, su interoperabilidad y su comunicación con los servicios del sistema central. Dichas verificaciones se consolidaban en la

fase de integración del sistema, en el que se validaba el comportamiento general de la plataforma en un entorno controlado.

3.3.4. Fase 4: Integración del sistema

En esta etapa, una vez que se ha trabajado y activado cada uno de los procesos, comienza el momento de la puesta en común de todos, respetando, en todo momento, el modelo lógico comentado anteriormente, vista, controlador; y se insistía en la interoperabilidad de los diferentes componentes que lo componen, entre módulos, interacciones y operaciones que le son propias. Se llegó a estudiar la interfaz de cada uno de los procesos, como se daría la navegación entre módulos y acabó por tener que la conexión de credenciales de una tienda con API REST hiciera el momento propicio para llevar a cabo la ejecución de la aplicación. Se llegó a plantear que el mismo sea un sistema eficiente y, sobre todo operativo, de forma que se le pueda hacer uso en cualquier momento del e-commerce real.

3.3.5. Fase 5: Pruebas funcionales en entorno simulado (Scrum)

En esta fase se realizaron pruebas de funcionalidad utilizando la tienda en línea que se implementó anteriormente en un servidor de la universidad. Estas pruebas se ejecutaron para probar la funcionalidad de cada uno de los módulos, de cómo la conexión entre la tienda en línea y WooCommerce hizo que se evidencie el funcionamiento de cada módulo, al momento de que algo fallaba se realizaban ajustes para que la aplicación funcione correctamente y evidenciar la actualización de productos, el inventario, el reporte de ventas y la lista de precios para distribuidores.

CAPITULO IV

RESULTADOS

Este capítulo se centra en el análisis, el desarrollo y la implementación de la aplicación PyWoo, esta plataforma fue diseñada para optimizar los procesos administrativos. Esta solución contempla un sistema de autenticación mediante las credenciales de API REST de la tienda en línea. Los módulos analizados corresponden a: reportes de ventas dentro de rangos de fechas definidas, visualizar el inventario, actualizar productos y lista de productos para distribuidores con cálculos automáticos de precios y descuentos. Tomando en cuenta que cada módulo debe ser intuitivo y escalable, analizando los resultados beneficiosos que esta aplicación puede generar.

4.1. Análisis

4.1.1. Propósito

Se identifican y definen los requerimientos funcionales, no funcionales y de dominio, los cuales garantizan el desarrollo del sistema, que será capaz de optimizar procesos administrativos. Se contempla que el usuario tiene una capacidad no tan técnica, es por eso que se tomó en cuenta la eficiencia, la seguridad y la usabilidad de la aplicación. Se establecieron lineamientos que aseguren a funcionalidad del sistema. Se priorizó que la aplicación sea intuitiva para el usuario o cualquier persona que se quede a cargo de las operaciones y estas no sean tan complejas, de esa manera se reducen los errores al momento de realizar los procesos.

4.1.2. Alcance

El alcance del proyecto comprende el diseño y desarrollo de una aplicación modular destinada a automatizar los principales procesos administrativos de la tienda en línea, entre los que se incluyen la gestión de inventarios, la actualización de productos, la generación de reportes de ventas y la elaboración de listados de productos para distribuidores. Para el cumplimiento de los objetivos propuestos, se contempló la implementación de un entorno de pruebas que permita validar tanto el funcionamiento individual de cada módulo como su correcta integración con la tienda en producción, mediante el uso de la API REST de

WooCommerce. No se consideró la realización de aplicaciones de teléfonos móviles o de la ampliación de las funcionalidades propias del sistema básico porque la filosofía del proyecto se centraba en trasladar lo comunicado a los usuarios que poseen pocos conocimientos técnicos para resumir y cancelar procedimientos administrativos. A raíz de esta premisa base, la aplicación se configura como un sistema de soporte que añade contenido a la operativa de la tienda al efecto de proporcionar un procesamiento más eficiente e intuitivo. Contexto del Sistema.

El sistema PyWoo fundamenta su funcionamiento en principios basados en las ciencias de la computación y / o bien de la ingeniería del software, con la finalidad de mitigar una dependencia técnica de la operación administrativa de la tienda en línea. Su contexto de funcionamiento está caracterizado por una interacción directa en la práctica con la API REST de WooCommerce, permitiendo así el acceso a información en tiempo real que resultan ser relevantes como pueden ser informes de ventas, actualización de productos, niveles de stock e información destinada preferentemente a los distribuidores.

PyWoo es una solución intermedia entre la ejecución de scripts que ayuda a mitigar los errores al ejecutar las operaciones y gestionar los niveles de eficiencia de los procesos mediante la separación de los módulos y pocas funciones designadas.

Tabla 1

Definiciones, Acrónimos y Abreviaturas

Abreviatura / Término	Definición
<i>API REST</i>	Interfaz de Programación de Aplicaciones con arquitectura REST. Mecanismo principal de interoperabilidad.
<i>RF</i>	Requerimiento Funcional.
<i>NFR</i>	Requerimiento No Funcional.
<i>Scrum</i>	Marco de trabajo ágil utilizado en las fases de desarrollo y pruebas.
<i>SRS</i>	Documento de Especificación de Requerimientos de Software.
<i>API</i>	Interfaz para Programación de Aplicaciones que permite la comunicación entre sistemas de software.
<i>UI</i>	Interfaz de usuario. Elementos visuales con los que interactúa el usuario.
<i>UX</i>	Grado de satisfacción y facilidad de uso del sistema por parte del usuario.
<i>GUI</i>	Interfaz gráfica de usuario que permite la interacción por medio de ventanas, botones, formularios.
<i>SKU</i>	Código único que se utiliza para identificar un producto que está en el inventario.
<i>LEMP</i>	Linux, Nginx, MariaDB y PHP. Conjunto de tecnologías de servidores web.
<i>Plugin</i>	Extensión que añade funcionalidades adicionales a una aplicación.

Nota. Listado de términos clave utilizados en el documento. Elaborado por: La autora.

4.1.3. Perspectiva del Sistema

PyWoo es un sistema complementario para ampliar las capacidades administrativas de WooCommerce, sin la necesidad de ingresar a la plataforma de WordPress y realizar cualquier proceso pendiente desde afuera, además de añadir capacidades de automatización modularidad. La aplicación no almacena datos en modo persistente o fuera del sistema, ya que todas las operaciones se efectúan directamente sobre la tienda en línea, garantizando que la información procesada o que se encuentra ya en la tienda sea reflejada en tiempo real al momento de la ejecución de cualquiera de los módulos. Este diseño de arquitectura permite un estrecho control sobre la información del negocio, pero a la vez permite que se puedan realizar futuras ampliaciones sin afectar la operatividad de la tienda, en el caso de agregar otro módulo no perjudicaría la estructura general de todo el sistema.

4.1.4. Usuario Operador de la Tienda

El usuario principal de la plataforma corresponde al operador de la tienda a cargo de inventario, actualizar productos, el reporte de ventas y la lista de distribuidores. Su nivel técnico se considera medio, ya que ha utilizado aplicaciones de ofimática y básicas de software, de tal forma que la interfaz de PyWoo fue diseñada de forma clara, intuitiva y fácil de entender, sin que sea necesario realizar comandos complejos o ejecutar scripts que se encuentren de forma independiente. Se ha minimizado la posibilidad de errores, pero que, si sea muy fácil de utilizar, sobre todo porque al ser intuitiva no es necesario tener conocimientos tan técnicos para utilizar la aplicación.

4.1.5. Restricciones Generales

Las restricciones generales identificadas durante el desarrollo garantizaron la viabilidad técnica y operativa del proyecto. Se destaca que el uso exclusivo de software libre, la integración obligatoria de la API REST de WooCommerce para el funcionamiento de la aplicación, la combinación de metodologías de desarrollo en cascada y ágil para las distintas fases del proyecto. Estas limitaciones han permitido la implementación del sistema y la arquitectura del programa y de esta forma sea seguro y eficiente.

Tabla 2*Restricciones Generales*

ID	Restricción	Tipo	Justificación
R-01	Tecnología de Software Libre	Diseño y Desarrollo	El sistema deberá implementarse utilizando exclusivamente software libre.
R-02	Integración Única	Interoperabilidad	El sistema deberá integrarse obligatoriamente con la API REST de una instancia activa de la tienda en línea.
R-03	Metodología Mixta	Metodología	El desarrollo deberá seguir el modelo en cascada para fases iniciales y un marco ágil para las fases de construcción y prueba.

Nota. Se presenta los límites tecnológicos, metodológicos y económicos. Elaborado por: La autora.

4.1.6. Suposiciones y Dependencias

Se asumió que los equipos sobre los que se ejecuta la plataforma cuentan con el hardware y software mínimo que garantizan la ejecución de la misma. Se ha encontrado que los precios y el stock de los productos son modificables de forma efectiva por medio de la API REST y sin acceso a la misma, las funcionalidades no se puedan ejecutar. En esta medida, la plataforma actualiza datos directamente en WooCommerce sin almacenamiento propio, lo que simplifica la arquitectura asegura que la información expuesta corresponda al estado actual de la tienda en línea en tiempo real.

4.1.7. Requerimientos Funcionales (RF)

Los requerimientos funcionales se organizaron por módulo, se prioriza la automatización de procesos críticos y la obtención de resultados precisos con poca intervención del usuario. Es importante destacar el reporte de ventas por rangos de fecha, visualizar y exportar el inventario, actualizar productos mediante archivos, generar listas de productos para distribuidores con precios y descuentos calculados automáticamente. Para cada una de las funcionalidades implementadas se ha tenido mucho en cuenta que la complejidad de su uso sea la más pequeño posible, de forma que se reduzcan los errores y se asegure que los resultados sean coherentes con la información disponible en WooCommerce.

Tabla 3*Requerimientos Funcionales Módulo de Reporte Ventas*

ID	Tipo	Enunciado	Racional	Criterio de Verificación
RF-01	Funcional	El sistema deberá permitir al usuario seleccionar un rango de fechas entre la Fecha de Inicio y la Fecha Fin para la consulta de ventas.	Permite análisis flexible de periodos específicos.	El sistema acepta y procesa rangos válidos sin error.
RF-02	Funcional	El sistema deberá consultar la API REST de WooCommerce para recuperar todas las órdenes dentro del rango definido.	Garantiza integridad de datos con la tienda.	Se recuperan todas las órdenes del rango.
RF-03	Funcional	El sistema deberá mostrar automáticamente los resultados en una tabla al presionar “Generar Reporte”.	Evita pasos innecesarios para el usuario.	La tabla se llena sin acciones adicionales.
RF-04	Funcional	El sistema deberá mostrar en el reporte: SKU, nombre del producto, cantidad, precio, total y método de pago.	Permite análisis completo de ventas.	La tabla contiene todas las columnas requeridas.
RF-05	Funcional	El sistema deberá calcular automáticamente totales y ganancias.	Reduce errores al reducir el cálculo manual.	Los valores coinciden con los datos procesados.
RF-06	Funcional	El sistema deberá permitir la exportación del reporte a Excel mediante el botón “Exportar”.	Facilita análisis externo.	El archivo Excel se genera correctamente.

Nota. Se presenta los requerimientos funcionales definidos para el módulo de Reporte Ventas.
Elaborado por: La autora.

Tabla 4*Requerimientos Funcionales Módulo de Inventario*

ID	Tipo	Enunciado	Racional	Criterio de Verificación
RF-01	Funcional	El sistema deberá consultar el inventario completo de productos desde WooCommerce.	Permite visión global del stock.	La tabla muestra todos los productos.
RF-02	Funcional	El sistema deberá identificar productos con stock bajo o sin stock.	Apoya la toma de decisiones.	Los productos se clasifican correctamente.
RF-03	Funcional	El sistema deberá mostrar automáticamente los resultados al finalizar la consulta.	Interfaz clara y directa.	No existen botones “Detalles”.
RF-04	Funcional	El sistema deberá permitir la exportación del inventario a Excel mediante el botón “Exportar”.	Facilita análisis externo.	El archivo Excel se genera correctamente.

Nota. Se presenta los requerimientos funcionales definidos para el módulo de Inventario.
Elaborado por: La autora.

Tabla 5*Requerimientos Funcionales Módulo de Actualización Productos*

ID	Tipo	Enunciado	Racional	Criterio de Verificación
RF-01	Funcional	El sistema deberá permitir la carga de archivos CSV o XLSX con SKU, precio y/o stock.	Automatiza actualizaciones masivas.	El archivo se valida correctamente.
RF-02	Funcional	El sistema deberá validar la estructura del archivo antes de procesar.	Previene errores críticos.	Archivos inválidos no se procesan.
RF-03	Funcional	El sistema deberá mostrar una ventana modal de progreso durante la actualización.	Proporciona retroalimentación visual.	El progreso se visualiza correctamente.
RF-04	Funcional	El sistema deberá aplicar cambios solo tras confirmación implícita del proceso.	Control operativo seguro.	Solo los SKUs válidos se actualizan.
RF-05	Funcional	El sistema deberá permitir la exportación del inventario a Excel mediante el botón “Exportar”.	Facilita análisis externo.	El archivo Excel se genera correctamente.

Nota. Se presenta los requerimientos funcionales definidos para el módulo de Actualización de Productos. Elaborado por: La autora.

Tabla 6*Requerimientos Funcionales Módulo Lista de Distribuidores*

ID	Tipo	Enunciado	Racional	Criterio de Verificación
RF-01	Funcional	El sistema deberá generar una lista de productos elegibles para distribuidores.	Automatiza la creación de listas comerciales.	La tabla se genera correctamente.
RF-02	Funcional	El sistema deberá calcular precios de distribuidor aplicando reglas de descuento.	Reduce errores manuales.	Los cálculos coinciden con la fórmula definida.
RF-03	Funcional	El sistema deberá permitir exportar la lista en dos formatos: interno y externo.	Diferencia uso interno/externo.	Ambos archivos se generan correctamente.
RF-04	Funcional	El sistema deberá mostrar automáticamente la tabla al finalizar el cálculo.	Flujo de trabajo continuo.	No se requieren botones adicionales.
RF-05	Funcional	El sistema deberá permitir la exportación del inventario a Excel mediante el botón “Exportar”.	Facilita análisis externo.	El archivo Excel se genera correctamente.

Nota. Se presenta los requerimientos funcionales definidos para el módulo de la Lista de Distribuidores. Elaborado por: La autora.

4.1.8. Requerimientos No Funcionales (NFR)

Los requerimientos no funcionales se enfocaron en la usabilidad, rendimiento, seguridad y mantenibilidad del sistema. La interfaz fue desarrollada para ser intuitiva, con retroalimentación visual y mensajes claros; las operaciones críticas se registran para auditoría; las actualizaciones masivas se ejecutan en tiempos mínimos; y el código sigue estándares de documentación que facilitan su mantenimiento y futura escalabilidad. Estas medidas aseguran que PyWoo no solo cumpla con los objetivos funcionales, sino que también proporcione una experiencia de usuario confiable y eficiente.

Tabla 7

Requerimientos No Funcionales

ID	Tipo	Enunciado	Racional	Criterio de Verificación
NFR-01	Usabilidad	La interfaz deberá ser clara, simple e intuitiva, comparable al uso de una aplicación básica de ofimática.	El usuario no posee conocimientos técnicos avanzados.	El usuario completa tareas de una forma intuitiva.
NFR-02	Usabilidad	El sistema deberá contar con un menú principal centralizado para acceder a todos los módulos.	Facilita la navegación y reduce confusión.	El menú permite acceso directo a cada módulo.
NFR-03	Usabilidad	El sistema deberá utilizar terminología descriptiva y no técnica en botones y mensajes.	Mejora la comprensión del usuario.	Todos los textos son comprensibles.
NFR-04	Usabilidad	El sistema deberá mostrar mensajes de error claros y orientados a la acción correctiva.	Facilita la resolución de problemas.	El mensaje indica causa y solución.
NFR-05	Rendimiento	El sistema deberá generar reportes.	Garantiza eficiencia operativa.	Pruebas de rendimiento cumplen el tiempo.
NFR-06	Rendimiento	El sistema deberá procesar actualizaciones masivas de todos los SKU en menos de 5 segundos.	Automatización eficiente.	Tiempo de ejecución validado.
NFR-07	Rendimiento	El sistema no deberá afectar el rendimiento de la tienda en línea.	Protege la experiencia del cliente final.	Monitoreo sin degradación significativa.
NFR-08	Seguridad	El sistema deberá registrar las operaciones masivas realizadas.	Permite auditoría y control.	Registro generado correctamente.
NFR-09	Seguridad	La comunicación con la API deberá realizarse mediante HTTPS.	Protege datos sensibles.	Conexión cifrada validada.

Nota. Se presenta el diagrama de flujo de la autenticación y conexión de la API. Elaborado por: La autora.

4.1.9. Requerimientos de Dominio (DR)

Se establecieron reglas específicas que limitan la operación del sistema a software libre, un único stock por SKU, cumplimiento de normas de privacidad y seguridad de datos, y dependencia exclusiva de la API REST de WooCommerce. Estas condiciones garantizan coherencia, seguridad y viabilidad del sistema dentro del contexto operativo de la tienda.

Tabla 8

Requerimientos de Dominio

ID	Tipo	Enunciado	Racional	Criterio de Verificación
DR-01	Tecnología	El sistema deberá operar exclusivamente sobre software libre.	Restricción presupuestaria del proyecto.	No se utilizan licencias propietarias.
DR-02	Negocio	El inventario deberá basarse en un único stock general por SKU.	Simplifica la gestión operativa.	La consulta usa un solo campo de stock.
DR-03	Legal	El sistema deberá cumplir principios de privacidad y seguridad de datos.	Cumplimiento ético y legal.	Auditoría de manejo de datos.
DR-04	Integración	El sistema dependerá exclusivamente de la API REST de WooCommerce.	Garantiza coherencia de datos.	Sin API no se ejecutan procesos.

Nota. Se presenta las reglas del dominio consideradas para la operación del sistema. Elaborado por: La autora.

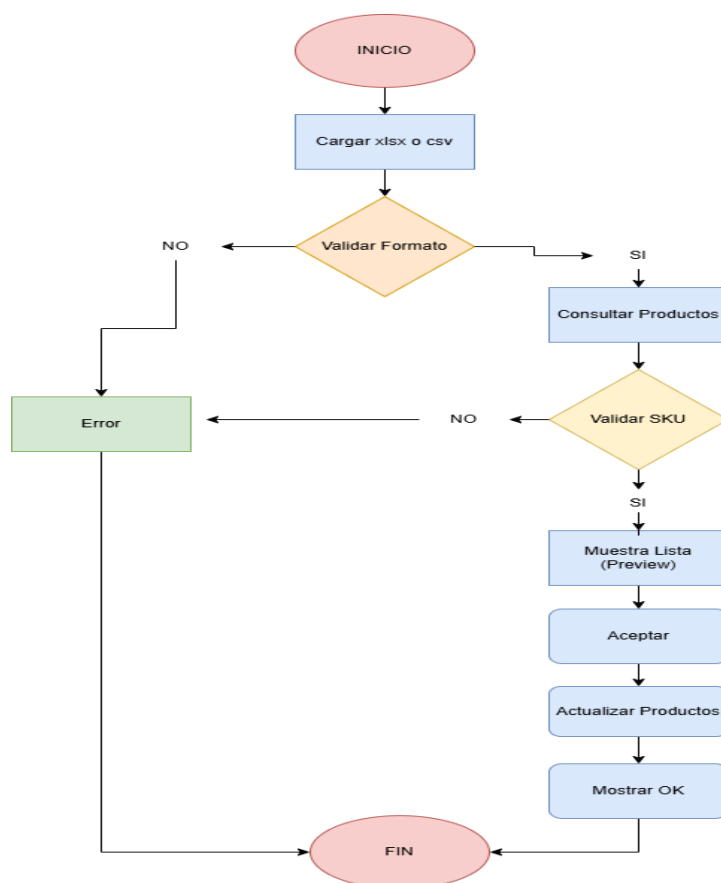
4.2. Diseño

En este subcapítulo se analizó el diseño y modelamiento de la aplicación. Se incluyeron diagramas y el diseño de interfaz abstracto. Todo lo relacionado con el diseño y modelamiento de la aplicación.

Este diagrama representa el flujo del proceso de actualización de productos. Comienza con la carga de un archivo XLSX o CSV, luego el sistema valida su formato y, si hay errores, detiene el proceso. Si el archivo es correcto, se consultan los productos en WooCommerce y se validan los SKU. Después se muestra una lista de previsualización para que el usuario confirme los cambios y, una vez aceptados, se actualizan los productos y se finaliza el proceso como se muestra en la Figura 1.

Figura 1

Diagrama Actualización de Productos



Nota. Se presenta el diagrama de actualización de productos. Elaborado por: La autora.

4.2.1. Diagrama de Despliegue

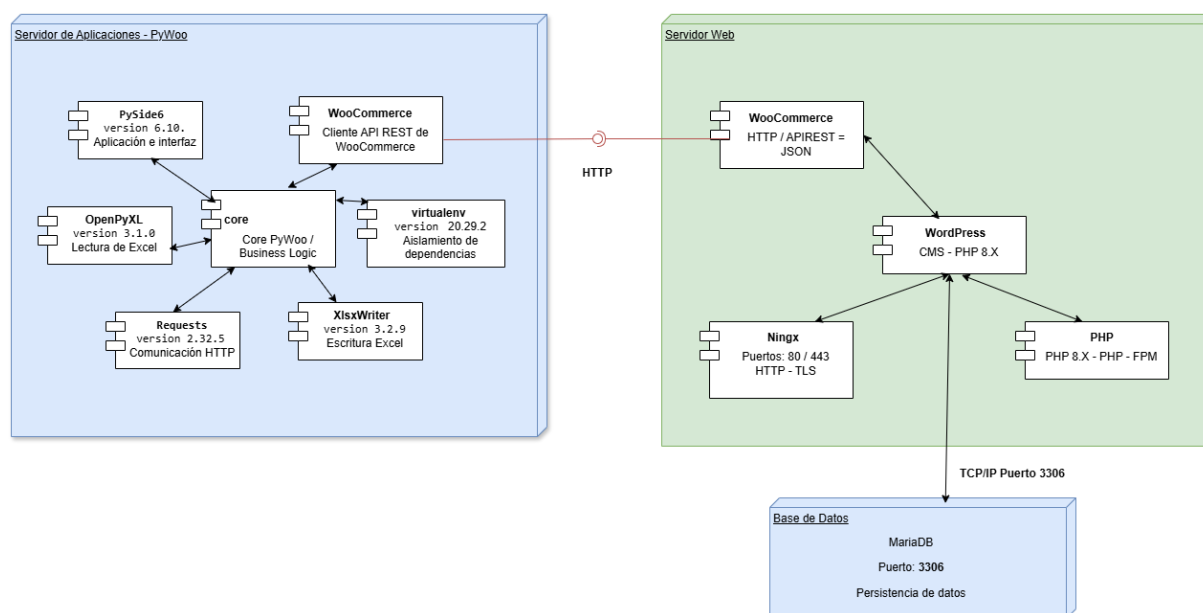
La Figura 2 representa la arquitectura general del sistema, donde se evidencia una separación clara entre el cliente y el servidor. En el lado del cliente se encuentra la aplicación PyWoo, compuesta por una interfaz gráfica desarrollada con PySide6 y un núcleo lógico llamado core que centraliza la comunicación, validaciones y procesamiento de datos. Este núcleo utiliza librerías externas como requests para la comunicación HTTP con WooCommerce y herramientas de exportación como xlsxwriter.

En el lado del servidor se observa la plataforma WooCommerce integrada sobre WordPress, la cual gestiona la información persistente en una base de datos MariaDB. Esta

arquitectura cliente-servidor permite independencia tecnológica, reutilización de código y escalabilidad del sistema.

Figura 2

Diagrama de Despliegue



Nota. Se presenta el diagrama de despliegue. Elaborado por: La autora.

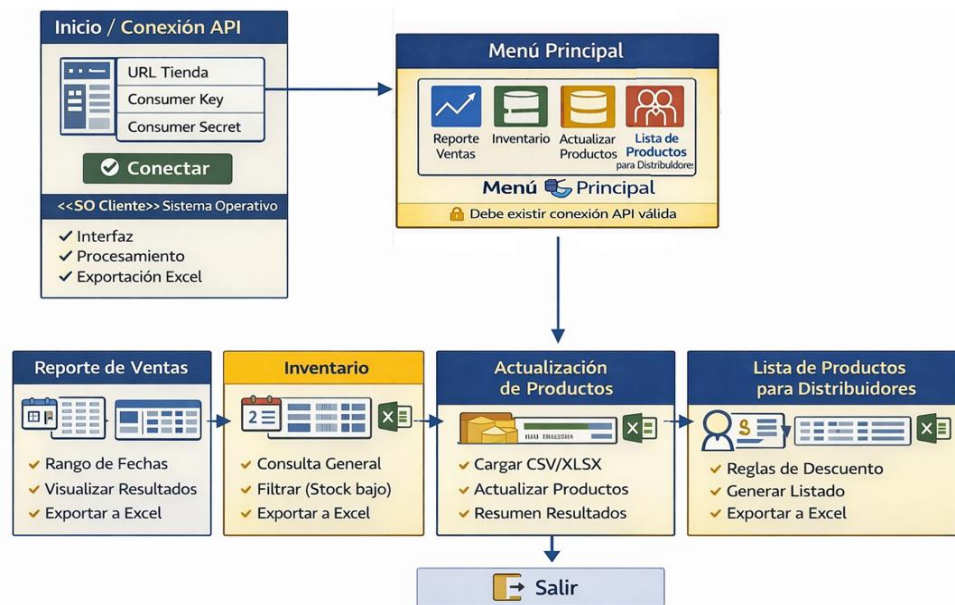
4.2.2. Funcionalidades del Sistema

La solución PyWoo se apoya en una arquitectura modular, siguiendo el principio de separación de responsabilidades, que permite llevar a cabo un mantenimiento correcto, una adecuada escalabilidad y una buena reutilización. La aplicación está construida en el lenguaje Python como lenguaje base y PySide6 como framework para la creación de las interfaces gráficas que permiten crear interfaces de usuario modernas y multiplataforma.

La Figura 3 muestra el funcionamiento de cada uno de los módulos. Cada uno de los módulos depende de una conexión válida con la API de WooCommerce y utilizan el cliente unificado del sistema, de esta forma se consigue coherencia, se minimiza la redundancia y se potencia la mantenibilidad del mismo.

Figura 3

Funcionalidad de PyWoo

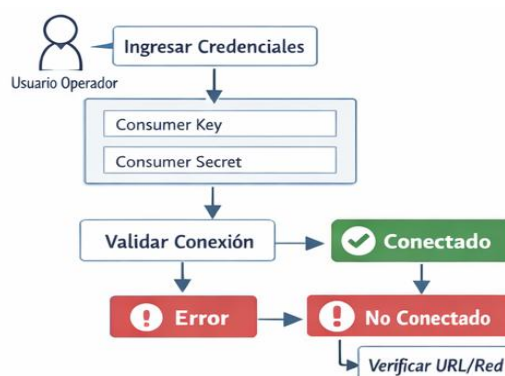


Nota. Se presenta el funcionamiento de la aplicación. Elaborado por: La autora.

La función que se utiliza para la aplicación de la funcionalidad de PyWoo es ingreso correcto de las credenciales de la tienda tal como apareció en la Figura 4, como el elemento más importante para poder aplicar la conexión correspondiente a la API REST de WooCommerce y que únicamente podrá tener lugar con una entrada de APIKEY que está autorizada e ingresar al menú de la tienda.

Figura 4

Funcionalidad Ingreso de Credenciales



Nota. Se presenta el ingreso de credenciales. Elaborado por: La autora.

Después de que se haya realizado el ingreso de las credenciales, la herramienta PyWoo establece la conexión con la interfaz de programación de aplicaciones REST de WooCommerce que establece la comunicación directa con la tienda en línea, permitiendo realizar la consulta y la actualización de varios datos relevantes en tiempo real. Durante dicho proceso, el sistema comprueba la respuesta que devuelve la API a través del protocolo HTTPS, lo que garantiza que se establece la conexión en su totalidad y que esta cumple con los criterios de seguridad requeridos, tal y como se ilustra en la Figura 5.

Figura 5

Funcionalidad Aplicación Conexión con la Tienda en Línea

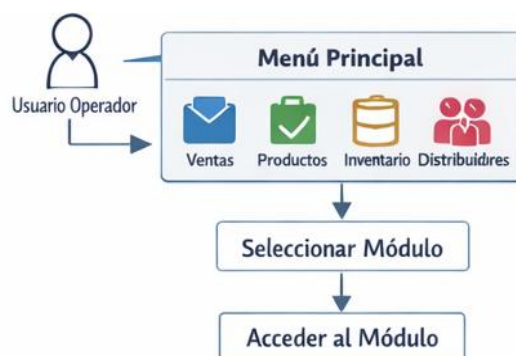


Nota. Se presenta la conexión con API REST y la tienda prototipo. Elaborado por: La autora.

El establece conexión y accede a un menú central que concentra todas las funcionalidades de PyWoo. Este menú ha sido diseñado bajo criterios de usabilidad, con una interfaz gráfica clara, botones con descripciones y modularidad que permite operar en reporte de ventas, inventario, actualización de productos y gestión de distribuidores, tal como se puede observar en la Figura 6.

Figura 6

Funcionalidad Menú



Nota. Se presenta el funcionamiento de menú. Elaborado por: La autora.

El módulo de ventas permite consultar los datos desde la API REST y mostrados en forma de tabla, que definirá atributos como SKU de la tienda. Además, el propio módulo ofrece la posibilidad de exportar la información en formato Excel, ya sea para su análisis extensivo utilizando herramientas de trabajo alternativas, o para facilitar su uso como respaldo documental, tal y como se observa en la Figura 7.

Figura 7

Funcionalidad de Reporte Ventas



Nota. Se presenta el funcionamiento de reporte ventas. Elaborado por: La autora.

El módulo que permite la actualización de productos permite aplicar una serie de modificaciones a un determinado conjunto de productos cuya identificación se realice mediante el SKU, ya sea la importación de un fichero mediante su correspondiente formato CSV o XLSX. Con ello se ayuda a minimizar el riesgo de tener datos no válidos en la información relacionada con los productos, tal como se puede apreciar en la Figura 8.

Figura 8

Funcionalidad de Actualización de Productos



Nota. Se presenta el funcionamiento de actualización de productos. Elaborado por: La autora.

Desde el módulo dedicado al inventario se puede observar tanto el stock disponible como el stock no disponible en la tienda online. La información aparece en tablas, ya que nos muestra un producto simple y una opción de producto variable, a partir del filtrado por la opción de exportar la información, tal como podemos comprobar en la Figura 9, se logra el análisis del inventario se pueda hacer desde la propia aplicación y no sea necesario ir a WordPress.

Figura 9

Funcionalidad de Inventario



Nota. Se presenta el funcionamiento de inventario. Elaborado por: La autora.

El funcionamiento de la lista de distribuidores, también conocida como lista generada de forma automática para elaborar las listas de productos de venta mayorista, basándose en las anteriores reglas de descuento y distinguiendo el formato interno del formato externo. Este módulo permite la homogeneidad de los cálculos de precio, optimiza el flujo de trabajo del procedimiento administrativo y permite la elaboración de las listas de la forma adecuada para los distribuidores para los usuarios no expertos. Lo que hace referencia a lista de distribuidores se incluye en la Figura 10.

Figura 10

Funcionalidad de Lista de Distribuidores



Nota. Se presenta el funcionamiento de lista de distribuidores. Elaborado por: La autora.

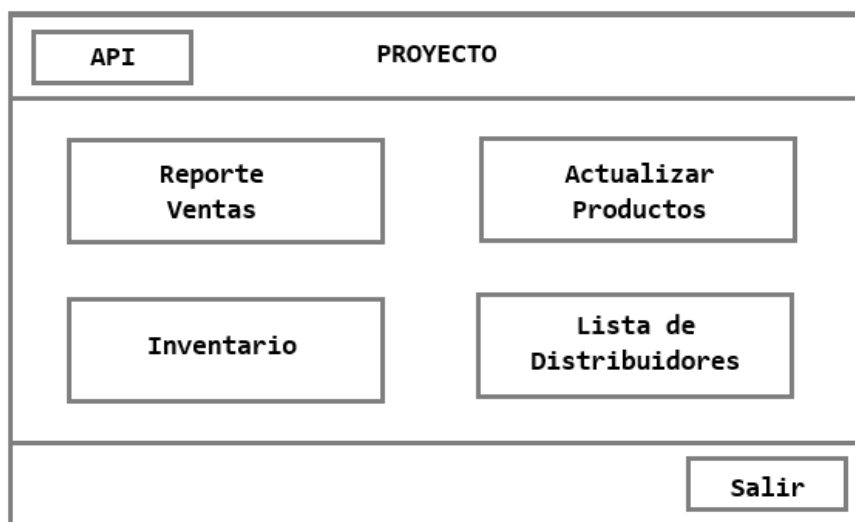
4.2.3. Wireframes

En el diseño visual de la plataforma PyWoo, este fue definido mediante la generación de wireframes en escala de grises, esto es, en su estructura y organización general de la interfaz de usuario, la cual fue utilizada como una guía para evaluar antes, anticipadamente, la usabilidad. En este sentido, los wireframes permitieron garantizar que la interacción pudiera ser comprensible, consistente y plenamente alcanzable por usuarios con un nivel técnico básico. La estrategia metodológica que caracterizó el trabajo en los wireframes planteó la iteración sobre los diseños y la identificación de problemas de navegación o de disposición y distribución de elementos en la interfaz, con la finalidad de que la exposición de cada módulo administrativo fuese clara, precisa y directa.

El wireframe del menú principal de PyWoo agrupa en la misma vista los módulos del sistema: Inventario, Reporte de Ventas, Actualización de Productos y Lista de Distribuidores como se puede apreciar en la Figura 11. La organización general de los módulos permite a los usuarios identificarlos de una forma muy rápida y acceder a la funcionalidad deseada sin ambigüedades. El wireframe es también un diseño visual minimalista, el cual da una experiencia intuitiva para las personas sin conocimientos técnicos y faculta la constitución de un flujo de trabajo lógico al aglutinar las funcionalidades en una sola estructura de menú.

Figura 11

Wireframe Menú



Nota. Se presenta el wireframe del menú principal. Elaborado por: La autora.

El wireframe del módulo de Reporte de Ventas, como se muestra en la Figura 12, fue diseñado para facilitar la consulta y el análisis de las ventas de la tienda dentro de un rango de fechas. La interfaz presenta campos para seleccionar la fecha de inicio y la fecha final, lo que permite al usuario filtrar la información de manera precisa. Además, incorpora una barra de progreso que indica el estado de carga del reporte, brindando retroalimentación visual durante el proceso.

Los resultados se organizan en una tabla de pedidos, lo que mejora la claridad y el orden de la información presentada. Este wireframe prioriza una estructura sencilla e intuitiva, reduciendo la necesidad de pasos innecesarios y evitando complejidad técnica para el usuario. Asimismo, se incluye la opción de exportar los datos a formatos como Excel, lo que facilita el análisis externo y la generación de reportes administrativos de forma rápida y confiable.

Figura 12

Wireframe Reporte Ventas

The wireframe illustrates the 'Reporte Ventas' interface. It features a header with the title 'REPORTE VENTAS' and a date input field labeled 'DD/MM/AA'. Below the header, there are two date input fields labeled 'Desde:' and 'Hasta:', followed by an 'Exportar' button. A 'Progreso:' label is positioned above a horizontal progress bar. Underneath the progress bar is a table with a header row labeled 'Pedidos' and a large empty rectangular area for the table content. At the bottom of the interface, there are two buttons: 'Volver al Menú' on the left and 'Salir' on the right.

Nota. Se presenta el wireframe de reporte ventas. Elaborado por: La autora.

El wireframe de Actualización de Productos, el que el usuario puede realizar la modificación de los productos, mediante la carga del archivo en formato CSV o Excel con la información que desea actualizar. La interfaz computacional que se muestra en la Figura 13, incluso ya incluye la correcta validación de la estructura del archivo y validación de los datos, a fin de poder verificar que los productos que se procesan son productos válidos.

Se incluye también una ventana de progreso que permite ir siguiendo la actualización que se va realizando, y también se diferencian productos simples de productos variables ya que esto ayuda a darles una mejor forma de poder visualizar la misma información. Este diseño hace que el tiempo necesario para la actualización se optimice, haciéndose, así, poca la importancia que tienen unos conocimientos profundos para poder actualizarlos y proporciona la tranquilidad al usuario de poder hacer visibles los cambios que se efectúan con el fin de mejorar la calidad operativa del sistema.

Figura 13

Wireframe de Actualización de Productos

The wireframe is a rectangular interface titled "ACTUALIZAR PRODUCTOS" in the top center. To the right of the title is a date input field labeled "DD/MM/AA". Below the title bar, there are three buttons: "Subir CSV/Excel" on the left, "Procesar" in the center, and "Exportar" on the right. Below these buttons is a progress indicator labeled "Progreso:" followed by a horizontal progress bar. Under the progress bar are two tabs: "Productos Simples" and "Productos Variados". Below the tabs is a large empty rectangular box, likely for a table or list of products. At the bottom of the interface are two buttons: "Volver al Menú" on the left and "Salir" on the right.

Nota. Se presenta el wireframe de actualización de productos. Elaborado por: La autora.

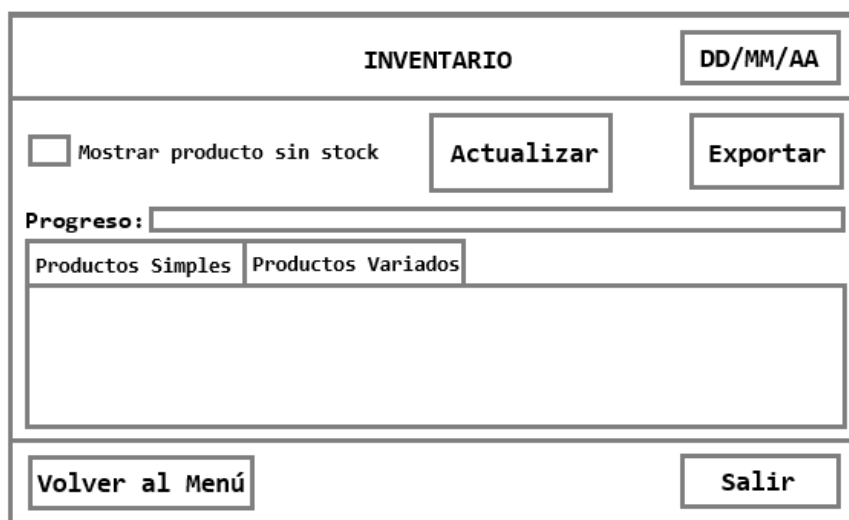
El wireframe de Inventario, el cual se muestra en la Figura 14, se encuentra diseñado para poder consultar y gestionar de forma clara el stock de productos de los disponibles y no disponibles para la tienda. El interfaz presenta elementos de filtro, como por ejemplo visualizar productos de stock 0, además de botones para actualizar la información y exportar los datos generados. En la cantidad de información visualizada dispone de una barra de progreso que indica en qué estado se encuentra la carga de la información, proporcionando retroalimentación visual de la carga.

Los productos se encuentran visualizados a través de una tabla delimitada para productos simples y productos variados, lo que permite una mejor lectura y un mejor control del inventario. Este wireframe nos permite obtener una visión clara del stock sin penetrar dentro

del panel administrativo de WooCommerce, contribuyendo así a la mejora del flujo de trabajo y a la reducción de errores operativos.

Figura 14

Wireframe de Inventario



The wireframe for the 'INVENTARIO' module is structured as follows:

- Header:** A box labeled 'INVENTARIO' on the left and a date input field 'DD/MM/AA' on the right.
- Filters and Actions:** A checkbox labeled 'Mostrar producto sin stock' on the left, and two buttons labeled 'Actualizar' and 'Exportar' on the right.
- Progress Bar:** A label 'Progreso:' followed by a horizontal progress bar.
- Product Type Tabs:** Two tabs labeled 'Productos Simples' and 'Productos Variados'.
- Main Content Area:** A large empty rectangular box for displaying the product list.
- Footer:** Two buttons labeled 'Volver al Menú' and 'Salir'.

Nota. Se presenta el wireframe del inventario. Elaborado por: La autora.

El diseño para el módulo de Lista de Distribuidores que aparece en la Figura 15 muestra una pantalla destinada a la generación de listados de productos con destino distribuidores. El usuario a partir de este wireframe puede, en esta línea, extraer la información de forma automática y visualizarla en forma de listas, separando productos de carácter simple de los productos de carácter variado, facilitando la lectura y el control de la información.

El sistema considera automáticamente los descuentos establecidos generando los precios ajustados sin necesidad de cálculo manual. La interfaz también incluye una barra de progreso que, visualmente, permite hacerse una idea del estado en el que va el proceso, y también un control de exportación a archivo de la información generada para su uso en informes o en comunicaciones comerciales. En resumen, el wireframe da prioridad a la rapidez y eficacia, la claridad de la información y el uso práctico, lo que permite a usuarios con conocimientos de técnica generar listas precisas de forma rápida y fiable.

Figura 15

Wireframe Lista de Distribuidores

The wireframe shows a rectangular interface with a header bar containing the title "LISTA DE DISTRIBUIDORES" and a date input field labeled "DD/MM/AA". Below the header, there are two buttons: "Generar" on the left and "Exportar" on the right. Under these buttons is a progress indicator labeled "Progreso:" followed by a horizontal bar. Below the progress bar are two tabs: "Productos Simples" and "Productos Variados". A large empty rectangular box is positioned below the tabs. At the bottom of the interface, there are two buttons: "Volver al Menú" on the left and "Salir" on the right.

Nota. Se presenta el wireframe de la lista de distribuidores. Elaborado por: La autora.

4.3. Desarrollo

En esta parte se describe el proceso para la construcción del sistema PyWoo, desarrollado bajo un enfoque híbrido que combina el modelo en cascada en las fases iniciales con la metodología ágil Scrum durante el desarrollo e integración de los módulos funcionales. El sistema se validó mediante una tienda prototipo basada en WooCommerce, que sirvió exclusivamente como entorno de pruebas.

Se debe tomar en cuenta, que a lo largo del desarrollo del proyecto Proyecto PyWoo se guardaron los cambios en el repositorio de GitHub. Ver repositorio:

<https://github.com/SamSami25/PyWoo>

A continuación, se evidencia cada iteración con sus respectivas tareas propuestas en función a la metodología propuesta para el desarrollo del sistema.

4.3.1. Sprint 1: Análisis de procesos y requerimientos técnicos

Este sprint corresponde a la primera parte del modelo en cascada, la fase de análisis del problema y los requerimientos del sistema. En esta fase se analizó cómo funciona cada proceso administrativo de la tienda online, se identificaron las tareas que se realizaban de forma automatizada pero independiente, repetitiva y poco eficiente, para cada uno de los módulos como reportes de ventas, actualización de productos, inventario y la elaboración de listas de distribuidores.

A partir del análisis se determinaron los requerimientos funcionales y no funcionales del sistema PyWoo, así como las restricciones técnicas para su implementación. Además, se definió el uso de una tienda prototipo basada en WooCommerce como entorno de pruebas, desplegada sobre un sistema GNU/Linux virtualizado, considerando los servicios necesarios para su operación.

El resultado de este sprint fue la base conceptual del proyecto, se estableció el alcance del sistema, como se va a orientó el diseño y posteriormente el desarrollo de los módulos.

4.3.2. Sprint 2: Configuración del entorno de pruebas

En este sprint se llevó a cabo la configuración del entorno de pruebas, con el objetivo de disponer de una infraestructura controlada para validar el funcionamiento del sistema PyWoo. Para ello, se implementó un servidor virtualizado con sistema operativo GNU/Linux, en el cual se configuraron los servicios necesarios para el despliegue de una tienda en WordPress con el complemento WooCommerce.

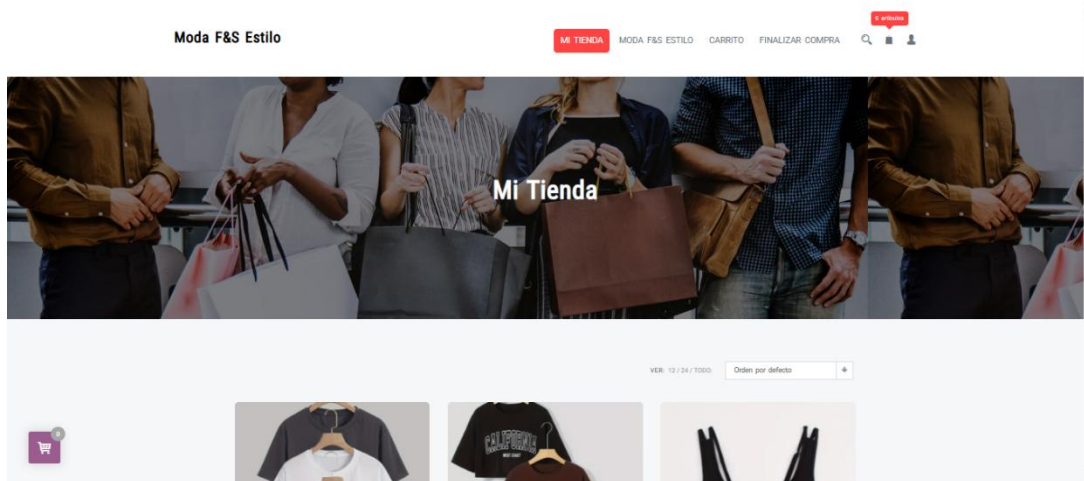
Durante esta fase se realizaron actividades como la actualización del sistema operativo, la configuración de la zona horaria del servidor y la instalación del entorno LEMP, compuesto por Linux, Nginx, MariaDB y PHP, además de herramientas para la administración de la base de datos. Posteriormente, se procedió a la instalación de WordPress, habilitando el acceso a la tienda prototipo a través de un navegador web, como se muestra en la Figura 16.

El entorno configurado permitió simular el funcionamiento de una tienda en línea real y sirvió como base para las pruebas de conexión mediante la API REST de WooCommerce. De

esta manera, se garantizó que los módulos desarrollados en PyWoo fueran implementados y validados bajo condiciones similares a un escenario de producción.

Figura 16

Entorno de Pruebas - Moda F&S Estilo



Nota. Se presenta la tienda en línea que fue considerada entorno de pruebas. Elaborado por: La autora.

4.3.3. Sprint 3: Integración del sistema

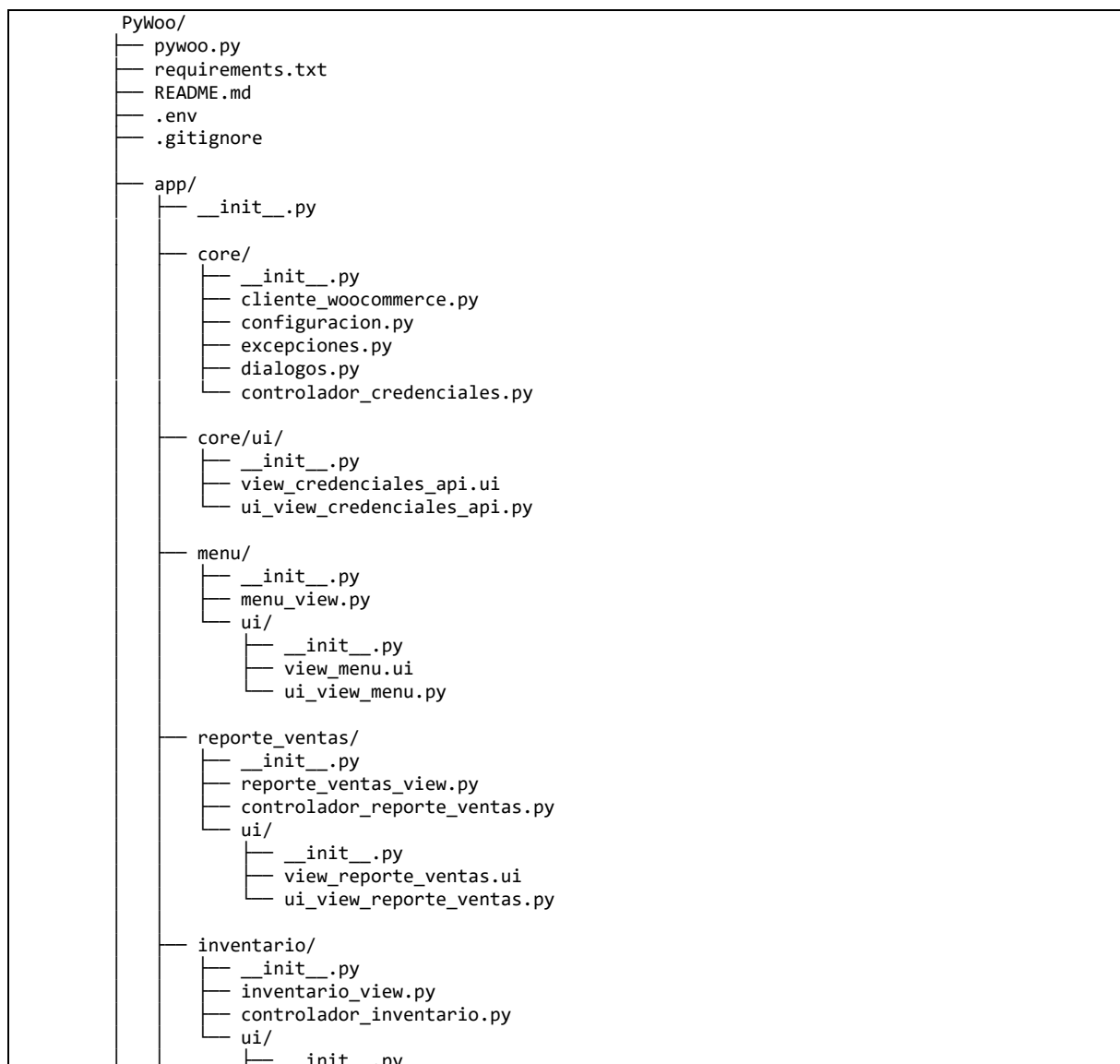
El Sprint 3 estuvo orientado a la integración completa de los módulos desarrollados en el sistema PyWoo. En esta etapa se incorporaron los módulos de Reporte de Ventas, Inventario, Actualización de Productos y Lista de Distribuidores, asegurando que todos operen bajo una lógica común, compartan una interfaz gráfica consistente y utilicen mecanismos unificados de procesamiento y comunicación.

Se evaluó la interoperabilidad entre los distintos componentes del sistema, verificando que cada módulo accediera de forma controlada a los servicios centrales, tales como la gestión de credenciales, la conexión con la API REST de WooCommerce y el manejo de excepciones. Asimismo, se comprobó la correcta comunicación con la tienda prototipo mediante solicitudes HTTP autenticadas, lo que permitió la consulta y actualización de información en tiempo real de manera segura.

La estructura del proyecto PyWoo se diseñó siguiendo un enfoque modular, estableciendo una clara separación entre la lógica de negocio, la capa de presentación y los controladores asociados a cada funcionalidad. El núcleo del sistema se concentra en el paquete core, responsable de la gestión de configuraciones, la conexión con WooCommerce y el tratamiento de errores. Por su parte, cada módulo funcional dispone de sus propias vistas, controladores y archivos de interfaz, lo que favorece el mantenimiento del sistema, su escalabilidad y la reutilización del código. Esta arquitectura modular facilita una integración eficiente entre los componentes y contribuye a la estabilidad general del sistema, como se muestra en la Figura 17.

Figura 17

Arquitectura General del Sistema PyWoo





Nota. Se presenta la arquitectura de PyWoo. Elaborado por: La autora.

4.3.4. Sprint 4: Desarrollo ágil de módulos funcionales

En este sprint se llevó a cabo la implementación de las interfaces finales del sistema, incorporando un diseño visual a color y garantizando su correcto funcionamiento. En esta etapa, los módulos fueron integrados de manera directa con la API REST de WooCommerce, permitiendo la interacción efectiva entre la aplicación PyWoo y la tienda en línea (*SamSami25 - Repositories*, s. f.).

4.3.4.1. Conexión de WooCommerce API REST

Con el fin de asegurar una comunicación segura entre el sistema PyWoo y la tienda WooCommerce, se desarrolló un módulo específico destinado a la gestión de las credenciales de la API REST. Este componente permite el ingreso, validación y almacenamiento controlado de información sensible, como la URL de la tienda, el Consumer Key y el Consumer Secret, evitando que dichos datos se encuentren directamente incrustados en el código fuente de la aplicación.

Las credenciales son almacenadas en un archivo de configuración: env, lo que posibilita su reutilización por parte de los distintos módulos del sistema y facilita el cambio entre entornos de ejecución sin requerir modificaciones en el código. Este enfoque contribuye a mejorar la seguridad, la mantenibilidad y la flexibilidad del sistema, como se muestra en la Figura 18.

Figura 18

Configuración de API REST

```
class Configuracion:
    def __init__(self):
        load_dotenv(ENV_PATH)

    def obtener_credenciales(self):
        url = os.getenv("WC_URL")
        ck = os.getenv("WC_KEY")
        cs = os.getenv("WC_SECRET")

        if not all([url, ck, cs]):
            raise ConfiguracionError("Credenciales incompletas")

        return url, ck, cs

    def guardar_credenciales(self, url, ck, cs):
        if not url or not ck or not cs:
            raise ConfiguracionError("No se permiten credenciales vacías")

        set_key(ENV_PATH, "WC_URL", url)
        set_key(ENV_PATH, "WC_KEY", ck)
        set_key(ENV_PATH, "WC_SECRET", cs)
```

[PyWoo\core\configuración.py](#)

Nota. Se presenta el módulo de configuración de credenciales WooCommerce. Elaborado por: La autora.

4.3.4.2. Conexión con la API REST de WooCommerce

La comunicación entre PyWoo y WooCommerce se realiza mediante la API REST oficial, utilizando la versión wc/v3. Esta integración permite acceder a recursos como órdenes, productos y clientes mediante solicitudes HTTP autenticadas, garantizando interoperabilidad y compatibilidad con la plataforma WooCommerce como se muestra en la Figura 19.

Figura 19

Conexión con la API REST de WooCommerce

```
class ClienteWooCommerce:
    def __init__(self, url, consumer_key, consumer_secret):
        self.base_url = f"{url.rstrip('/')}/wp-json/wc/v3"
        self.auth = (consumer_key, consumer_secret)

    def obtener_ordenes(self, per_page=100):
        response = Requests.get(
            f"{self.base_url}/orders",
            auth=self.auth,
            params={"per_page": per_page}
        )
        response.raise_for_status()
        return response.json()
```

[PyWoo/app/core/cliente_woocommerce.py](#)

Nota. Se presenta Implementación de cliente REST para WooCommerce. Elaborado por: La autora.

4.3.4.3. Módulo Menú Principal

El menú principal actúa como punto central de navegación del sistema PyWoo. El usuario puede acceder de forma directa a todos los módulos del sistema, garantizando una experiencia de usuario intuitiva y organizada como se muestra en la Figura 20.

Figura 20

Módulo Menú Principal

```
class MenuPrincipal(QMainWindow):
    def __init__(self):
        super().__init__()
        self.ui = Ui_MainW_menu()
        self.ui.setupUi(self)

        self.ui.bt_inventario.clicked.connect(self.abrir_inventario)
        self.ui.bt_reporteventas.clicked.connect(self.abrir_reporte_ventas)
```

[PyWoo/app/menu/menu_view.py](#)

Nota. Se presenta el menú principal de navegación del sistema PyWoo. Elaborado por: La autora.

4.3.4.4. Módulo Reporte de Ventas

El módulo de Reporte de Ventas permite generar informes detallados de las órdenes registradas en WooCommerce dentro de un rango de fechas definido por el usuario. Este módulo facilita el análisis financiero y la toma de decisiones mediante la exportación automática de datos a archivos Excel como se muestra en la Figura 21.

Figura 21

Módulo Reporte de Ventas

```
def obtener_ventas(self, fecha_desde, fecha_hasta):
    ventas = self._cliente.obtener_ordenes(per_page=100)
    desde = datetime.strptime(fecha_desde, "%Y-%m-%d")
    hasta = datetime.strptime(fecha_hasta, "%Y-%m-%d")
    return [
        v for v in ventas
        if desde <= datetime.fromisoformat(v["date_created"].replace("Z", "")) <= hasta
    ]
```

[PyWoo/app/reporte_ventas/controlador_reporte_ventas.py](#)

Nota. Se presenta el controlador del módulo Reporte de Ventas. Elaborado por: La autora.

4.3.4.5. Módulo Inventario

El módulo Inventario permite visualizar el estado actual de los productos almacenados en WooCommerce, filtrando por productos con stock, sin stock o todos. Esta funcionalidad es esencial para la gestión logística y el control de existencias como se muestra la Figura 22.

Figura 22

Módulo Inventario

```
def obtener_productos(self, filtro="todos"):
    productos = self._cliente.obtener_productos()
    if filtro == "con_stock":
        return [p for p in productos if (p.get("stock_quantity") or 0) > 0]
    if filtro == "sin_stock":
        return [p for p in productos if (p.get("stock_quantity") or 0) == 0]
    return productos
```

[PyWoo/app/inventario/controlador_inventario.py](#)

Nota. Se presenta el controlador del módulo Inventario. Elaborado por: La autora.

4.3.4.1. Módulo Actualizar Productos

Este módulo permite actualizar de forma masiva los precios y el stock de productos en WooCommerce mediante archivos Excel o CSV, optimizando procesos que tradicionalmente se realizan de forma manual como se muestra la Figura 23.

Figura 23

Módulo Actualizar Productos

```
def actualizar_productos(self):
    for p in self._productos_archivo:
        self._cliente.actualizar_producto(
            producto_id=p["id"],
            data={
                "regular_price": str(p["price"]),
                "stock_quantity": p["stock_quantity"],
                "manage_stock": True
            }
        )
```

[PyWoo/app/actualizar_productos/controlador actualizar productos.py](#)

Nota. Se presenta el controlador del módulo Actualizar Productos. Elaborado por: La autora.

4.3.4.1. Módulo Lista Distribuidores

El módulo Lista de Distribuidores obtiene información de clientes y productos para generar reportes comerciales destinados a distribuidores, facilitando la gestión de precios y descuentos como se muestra la Figura 24.

Figura 24

Módulo Lista Distribuidores

```
def generar_lista(self):
    clientes = self._cliente.obtener_clientes(per_page=100)
    if not clientes:
        raise PyWooError("No se encontraron distribuidores")
    self._distribuidores = clientes
```

PyWoo/app/lista_distribuidores/controlador_lista_distribuidores.py
--

Nota. Se presenta el controlador del módulo Lista Distribuidores. Elaborado por: La autora.

4.3.5. Sprint 5: Pruebas funcionales en entorno simulado

Durante la implementación de esta sprint, se llevaron a cabo las pruebas funcionales del sistema PyWoo en un entorno de simulación de la tienda en línea Moda F&S Estilo, disponible en el siguiente link:

<https://moda.csuioups.org/>

Se llevaron a cabo las pruebas funcionales del sistema PyWoo en un entorno de simulación de comercio electrónico correspondiente a la tienda online que corresponde a Moda F&S Estilo, la cual ha sido construida sobre la plataforma WooCommerce, y que sólo se ha utilizado para fines de prueba, permitiendo la medición del sistema sin comprometer información real y sin afectar la ejecución de operaciones comerciales activas, lo que garantiza la orden del proceso de validación y la seguridad del mismo.

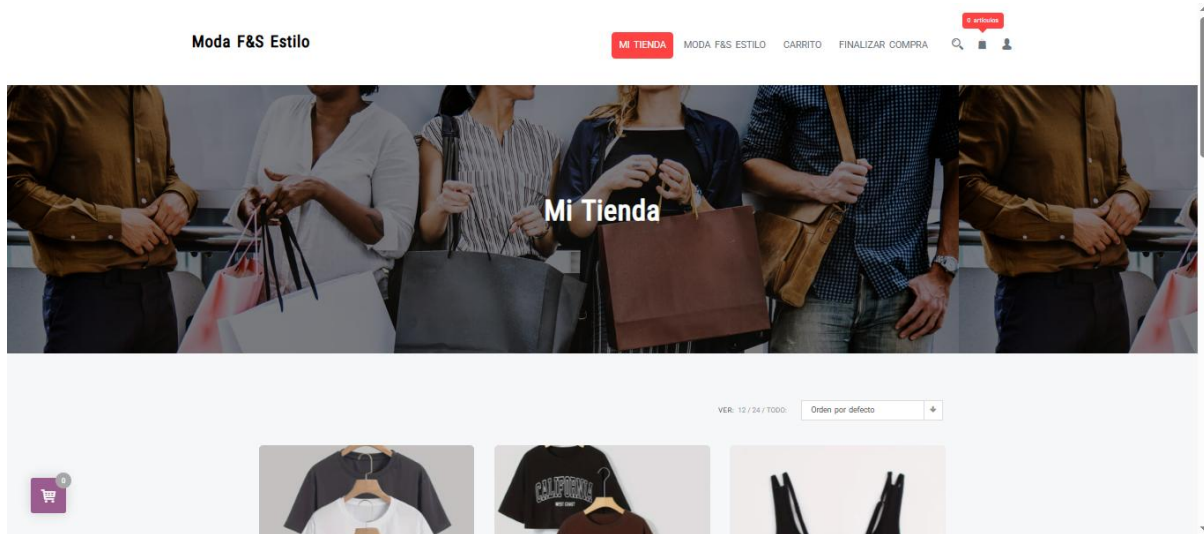
La finalidad del sprint en cuestión era corroborar la existencia de todos los módulos de los que está constituido PyWoo que son: Reporte de Ventas, Inventario, Actualización de Productos y Lista de Productos para Distribuidores además de comprobar que era posible la comunicación con la API REST de WooCommerce mediante peticiones correctamente autenticadas. La realización de las pruebas se centró en la consistencia del sistema, la correcta obtención y actualización de información en tiempo real y, si la información obtenida era fidedigna sería mostrada por la interfaz gráfica y en los ficheros exportados.

La adecuada realización del proceso, pero principalmente el proceso que se presentó permitió hallar posibles incoherencias, validar los workflows que se definieron y de esta manera el sistema funcionaba correctamente ante los variados escenarios de uso. Este hecho dejó al descubierto la consistencia y la fiabilidad de PyWoo en el entorno simulado y, por tanto, el sistema había quedado listo para la validación final y la exposición académica, tal como se observa en la Figura 25.

4.3.5.1. Pruebas

Figura 25

Tienda en Línea Moda F&S Estilo



Nota. Se presenta la tienda en línea que fue considerada entorno de pruebas.
Elaborado por: La autora.

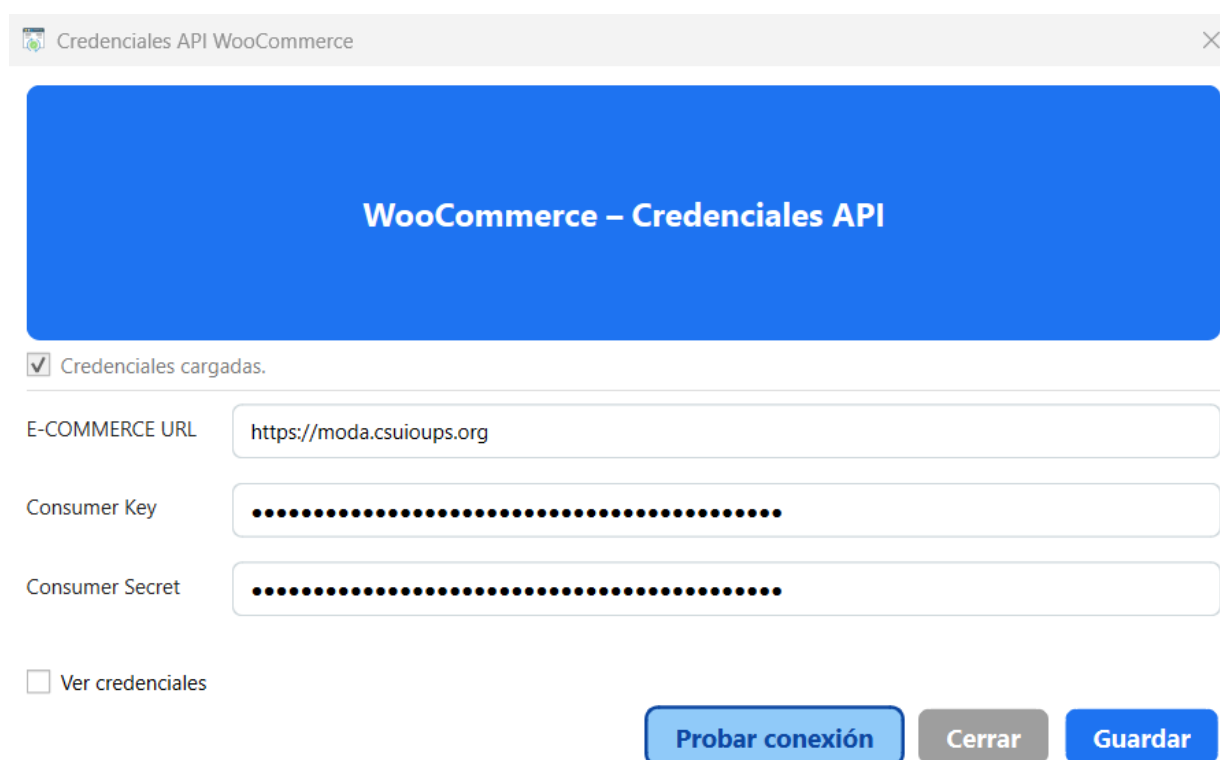
4.3.5.2. Configuración API WooCommerce

Para establecer la comunicación entre el sistema PyWoo y la tienda prototipo desarrollada sobre WooCommerce, se dispone de una interfaz destinada al ingreso de las credenciales de la API de WooCommerce, tal como se ilustra en la Figura 26. En esta ventana, el usuario debe registrar la URL de la tienda, junto con la Consumer Key y el Consumer Secret, credenciales necesarias para la autenticación de las solicitudes realizadas mediante la API REST en su versión wc/v3.

La interfaz incorpora funcionalidades para guardar las credenciales y realizar una prueba de conexión, lo que permite comprobar de forma inmediata la validez de los datos ingresados antes de proceder con el uso del sistema. Las credenciales se almacenan de manera externa al código fuente, lo cual fortalece la seguridad de la información sensible y previene su exposición directa. Este enfoque simplifica la configuración inicial y garantiza que los distintos módulos de PyWoo utilicen una conexión centralizada, segura y reutilizable.

Figura 26

Ingreso de Credenciales



The image shows a web browser window titled "Credenciales API WooCommerce". The main content area has a blue header with the text "WooCommerce – Credenciales API". Below this, there is a checkbox labeled "Credenciales cargadas." which is checked. Underneath, there are three input fields: "E-COMMERCE URL" with the value "https://moda.csuioups.org", "Consumer Key" with masked characters, and "Consumer Secret" with masked characters. At the bottom left, there is an unchecked checkbox labeled "Ver credenciales". At the bottom right, there are three buttons: "Probar conexión" (blue), "Cerrar" (grey), and "Guardar" (blue).

Nota. Se presenta el módulo Ingreso de Credenciales. Elaborado por: La autora.

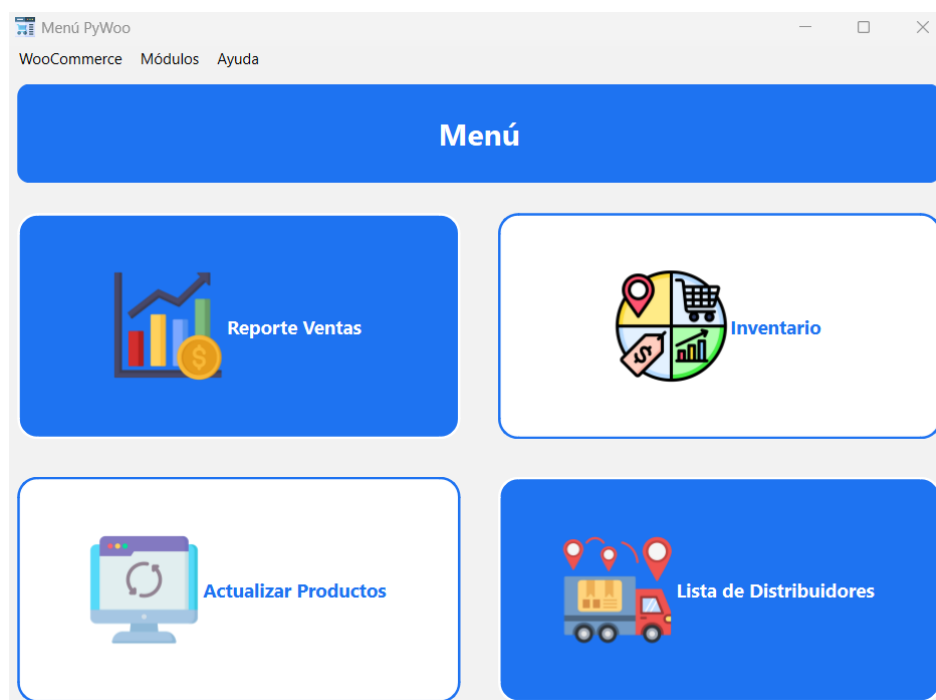
4.3.5.3. Acceso al Menú Principal

La Figura 27 presenta el menú principal de PyWoo, donde se han reagrupado en una sola interfaz las cuatro funcionalidades: Inventario, Reporte de Ventas, Actualizar Productos y Lista Distribuidores. Todos los módulos aparecen representados por botones de tamaño grande y diferentes, que se explican por sí mismos con diferentes iconos y colores, lo que facilita la identificación de la opción que se quiere ejecutar.

La interfaz mejora sin duda la experiencia de uso en comparación con la interfaz anterior, ya que se mejora la navegación y se reduce el tiempo que se tarda en llegar a las distintas opciones del proceso. Además, los elementos gráficos interactivos ayudan a que el sistema se convierta en un entorno intuitivo y fácil de utilizar, incluso para las personas que tengan conocimientos técnicos reducidos. En definitiva, el menú principal propone un flujo de trabajo robusto y ágil para gestionar la tienda de una forma cotidiana.

Figura 27

Menú Principal del Sistema



Nota. Se presenta el Menú de PyWoo. Elaborado por: La autora.

4.3.5.4. Reporte Ventas

En el módulo Reporte de Ventas, la opción de seleccionar pertenezcan a un rango de fecha específico que permita generar reportes donde consten las transacciones procesadas en la tienda. Considerado el periodo a analizar, el sistema procesara la información y retornara los resultados del proceso en una tabla correctamente estructurada que contenga los campos de información que le interesa al usuario: fecha de venta, cliente, subtotal, costos de envío, impuestos, descuentos, total, método de pago, utilidad, estado del pedido, tal como se aprecia en la Figura 28.

Mientras se ejecuta el proceso, se cuenta con una barra de progreso que permite informar al usuario del avance, lo que proporcionará más claridad y control sobre la operación. Al mismo tiempo, el sistema permite también exportar la información a un archivo Excel, lo que además de facilitar el análisis externo, permite llevar una copia de la información administrativa, tal como se visualiza en la Figura 29.

Figura 28

Módulo de Reporte Ventas

Fecha	Cliente	Subtotal Envío	IVA	Descuento	Total	Utilidad	Estado	Notas	Pedido Identificación	Correo	Teléfono	Dirección	Ciudad	Cajero
2026-01-19 01:16	Carol Castillo	9.00	0.00	0.00	9.00	0.00	Borrador		242	sami@gmail.com	0981234567	Chillogallo	Quito	
2026-01-19 00:28	Carol Castillo	34.50	2.00	0.00	36.50	0.00	Procesando	Pedido Frágil	241	sami@gmail.com	0981234567	Chillogallo	Quito	
2026-01-10 23:37	Sam Cabrera	38.00	2.00	0.00	40.00	0.00	Procesando		240	sami@gmail.com	0981234567	Quitumbe	Quito	
2026-01-07 17:43		5.00	2.00	0.00	7.00	0.00	Borrador		239					
2026-01-03 22:58	Sam Castillo	60.00	0.00	0.00	60.00	0.00	En espera	Gracias, frágil por favor.	238	sami@gmail.com	0912345678	Quitumbe	Quito	
2025-12-09 13:13	Santiago Iza	106.00	2.00	0.00	108.00	0.00	Procesando	Es para mi novia, pueden tal vez ponerle una dedicatoria.	237	pasante.ituio@coheco.ec	0912345678	La Kennedy	Quito	
2025-12-09 13:08	Valeria Cabrera	88.00	0.00	0.00	88.00	0.00	En espera	Por favor deben llegar empacutados con cuidado.	236	pasante.ituio@coheco.ec	0912345678	Av. Eloy Alfaro	Quito	
2025-12-04 12:29	Sam Coheco	36.50	2.00	0.00	38.50	0.00	Procesando		235	pasante.ituio@coheco.ec	0912345678	Rio Coca	Quito	
2025-03-21 10:15	Samantha Cabrera	40.00	2.00	0.00	42.00	0.00	Procesando	Me encantan!!!!!! <3	233	samitta@gmail.com		Quitumbe	Quito	
2025-03-21 10:12	Beatriz Rodriguez	103.00	2.00	0.00	105.00	0.00	Procesando	Gracias todo esta muy bonito.	232	bettlrodri@gmail.com		La Carolina	Quito	
2025-03-21 10:09	Fernanda Carrera	79.50	0.00	0.00	79.50	0.00	En espera		231	fer.carrera@gmail.com		La Ecuatoriana	Quito	
2025-03-21 10:08	Valeria Sanchez	41.99	2.00	0.00	43.99	0.00	En espera		230	valesath@gmail.com		Quitumbe	Quito	
2025-03-21 10:06	Fernando Aldaz	45.00	2.00	0.00	47.00	0.00	Procesando		229	fernandoaldaz@gmail.com		Martha Bucaram	Quito	
2025-03-20 12:03	Celeste Poma	18.50	2.00	0.00	20.50	0.00	En espera	Muchas gracias	228	celeste@gmail.com		Quitumbe	Quito	
2025-03-20 12:01	David Aldaz	50.00	2.00	0.00	52.00	0.00	Procesando	Me gustaria recibirlo en mi departamento 205	227	david25@gmail.com		Chillogallo	Quito	
2025-03-17 21:54	Samantha Hidalgo	30.00	2.00	0.00	32.00	0.00	Procesando		177	aldazsami@gmail.com		Carapungo	Quito	
2025-03-11 22:43	Carita Suarez	25.00	2.00	0.00	27.00	0.00	En espera		176	carita@nemail.com		Martha Bucaram	Quito	

Nota. Se presenta la ejecución de reporte ventas. Elaborado por: La autora.

Figura 29

Archivo Exportado de Reporte Ventas

	A	B	C	D	E	F	G	H	I	J	K	L	M	
1	Fecha	Cliente	Subtotal	Envío	IVA	Descuento	Total	Utilidad	Estado	Notas	Pedido	Identificación	Correo	Te
28	2025-03-01 21:27	Sami Cabrera	59,98	0,00	0,00	0,00	59,98	0,00	En espera		166		samicabrera@gmail.com	
29	2025-03-01 21:26	Jhoel Paqui	56,00	0,00	0,00	0,00	56,00	0,00	En espera		165		david@gmail.com	
30	2025-03-01 21:25	Jhoel Paqui	36,00	2,00	0,00	0,00	38,00	0,00	Procesando		164		david@gmail.com	
31	2025-02-28 22:02	Santiago Iza	25,00	2,00	0,00	0,00	27,00	0,00	Completado		163		david@gmail.com	
32	2025-02-28 22:01	Valeria Aldaz	30,00	2,00	0,00	0,00	32,00	0,00	Procesando		162		david@gmail.com	
33	2025-02-28 22:00	Deiby Cabrera	99,00	0,00	0,00	0,00	99,00	0,00	En espera		161		david@gmail.com	
34	2025-02-28 21:56	Sarita Suarez	30,00	2,00	0,00	0,00	32,00	0,00	Procesando		160		sami@gmail.com	
35	2025-02-21 02:20	Sarita Suarez	90,00	0,00	0,00	0,00	90,00	0,00	Procesando		159		sami@gmail.com	
36	2025-02-21 02:17	Valeria Cabrera	250,00	2,00	0,00	0,00	252,00	0,00	Procesando		158		sami@gmail.com	
37	2025-02-21 02:16	Santiago Iza	18,00	2,00	0,00	0,00	20,00	0,00	Procesando		157		dpoma@gmail.com	
38	2025-02-21 02:14	David Cabrera	101,00	2,00	0,00	0,00	103,00	0,00	Procesando		156		saldaz@est.ups.edu.ec	
39	2025-02-18 11:18	Sami Aldaz	25,00	2,00	0,00	0,00	27,00	0,00	Procesando		61		aldazsami@gmail.com	
40	Totales		2.063,96	60,00	0,00	0,00	2.123,96	0,00						

Nota. Se presenta el formato de Reporte Ventas. Elaborado por: La autora.

4.3.5.5. Actualización de Productos

En el módulo Actualización de Productos se realizan modificaciones sobre la información de los productos: precios, stock, etc., realizando la carga de archivos en formato CSV o XLSX como se observa en la Figura 30. Al momento de querer actualizar el usuario verifica el SKU en la tienda Figura 31, como en el archivo se sube el SKU el sistema también

comprueba la estructura del archivo cargado y comprueba la existencia de los SKU de los productos en la tienda, asegurando que únicamente se traten los registros válidos.

Para comprobar si el módulo funciona bien se comparó la situación inicial de los productos con los valores obtenidos después de la ejecución del proceso, que verdadera se han actualizado. Finalmente, el sistema permite exportar los resultados a un archivo Excel, como respaldo de la información actualizada, como se puede comprobar en la Figura 34.

Figura 30

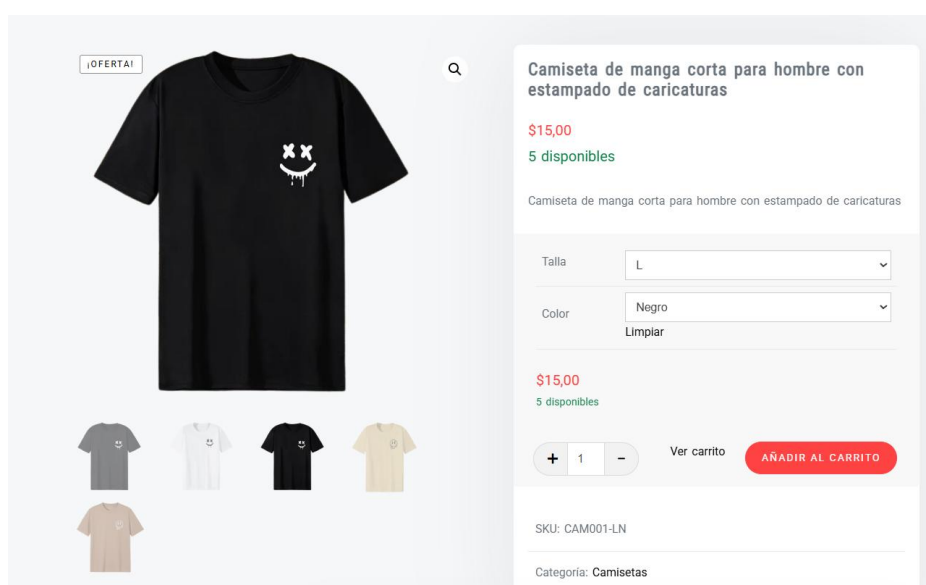
Formato Archivo de Actualización de Productos

	A	B	C	D	E
1	SKU	PRECIO	STOCK		
2	VES005	60.00	20		
3	CHO002		10		
4	ACE001		10		
5	BL004-S		10		
6	CAM001-SB		25		
7	CAM001-LN	20.00	10		
8	EP001		20		
9					
10					

Nota. Se presenta el formato de Actualización de Productos. Elaborado por: La autora.

Figura 31

Producto No Actualizado



Nota. Se presenta el producto a actualizar. Elaborado por: La autora.

Figura 32

Módulo Actualización de Productos

Actualización de Productos

WooCommerce Módulos Ayuda

Subir CSV / Excel Aplicar Cambios Archivo: Actualizar_Productos.xlsx Exportar

Aplicando cambios 7 de 7 100%

Productos Simples Productos Variados

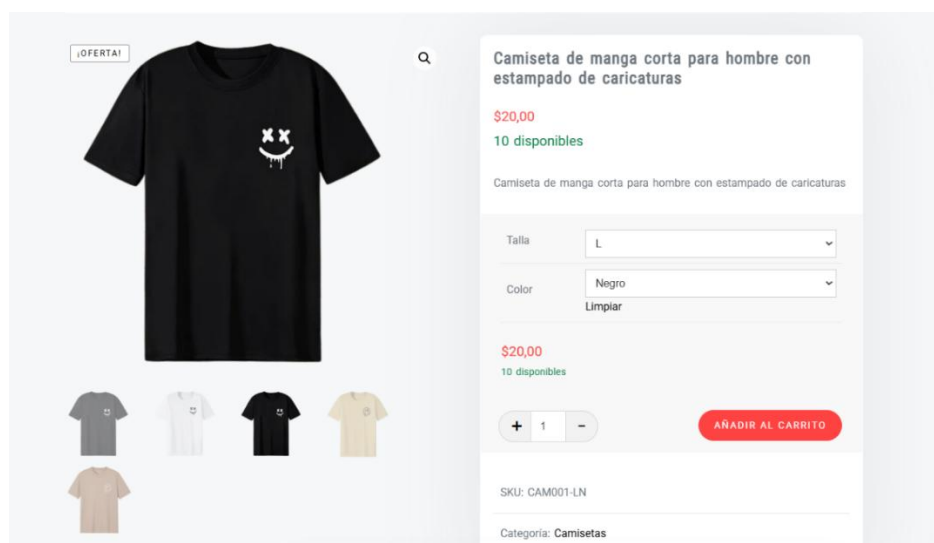
SKU	NOMBRE	CATEGORÍA	STOCK ACTUAL	STOCK NUEVO	PRECIO ACTUAL	PRECIO COMPRA	PRECIO VENTA ACTUAL	PRECIO VENTA NUEVO	ESTADO
JEAD001	Cargo Jeans de Cintura ...	Jeans	11		18	0	22		Sin cambio
ACE002	Pendientes Elegantes de ...	Accesorios	20		3	0	5		Sin cambio
ACE003	Pendientes de Perlas de ...	Accesorios	20		10	0	10		Sin cambio
LIC003	Pantalones de Yoga para ...	Licras	10		15	0	15		Sin cambio
VES002	Vestido sin mangas con ...	Vestidos	5		15	0	15		Sin cambio
CHO002	Chompa Acolchada con ...	Chompas	8	10	25	0	28		OK Actualizado
CHO003	Chompa Impermeable d...	Chompas	8		28	0	28		Sin cambio
VES004	Vestido Negro con Cuel...	Vestidos	4		30	0	30		Sin cambio
CHO001	Chaqueta Sudadera con ...	Chompas	6		29.99	0	29.99		Sin cambio
VES005	Vestido Rojo de Fiesta	Vestidos	10	20	49.99	0	49.99	60	OK Actualizado
CHA001	Chaqueta Crop Top ...	Chaquetas	-4		25	0	28		Sin cambio
CHA002	Chaqueta Cuero de ...	Chaquetas	9		25	0	28		Sin cambio
JEAD004	Mom Jean	Jeans			18	0	18		Sin cambio
JEAD003	Jeans de Denim Estrella ...	Jeans	9		18	0	18		Sin cambio
JEAD002	Jeans de Denim Elásticos...	Jeans	3		18	0	18		Sin cambio
CHA003	Chaqueta de Hombre	Chaquetas	6		28	0	28		Sin cambio

Volver al Menú Productos procesados correctamente

Nota. Se presenta el módulo Actualización de Productos. Elaborado por: La autora.

Figura 33

Producto Actualizado



Nota. Se presenta el producto ya actualizado. Elaborado por: La autora.

Figura 34

Archivo Exportado Actualizar Productos

	A	B	C	D	E	F	G
1	SKU	NOMBRE	CATEGORÍA	STOCK	PRECIO COMPRA	PRECIO VENTA	ESTADO
395	ACE0011	Pulsera Pandora de Corazón	Accesorios		0		Sin cambio
396	ACE0010	Pulsera Pandora de Stitch	Accesorios		0		Sin cambio
397	ACE009	Conjunto Mariposa Reloj y Pulsera	Accesorios		0		Sin cambio
398	ACE007	Bolso para Mujer Estilo Cruzado	Accesorios		0		Sin cambio
399	ACE006	Pulseras Yin y Yang para Compartir	Accesorios		0		Sin cambio
400	ACE005	Pulseras Mariposa para Compartir	Accesorios		0		Sin cambio
401	ACE004	Bolsa Stitch	Accesorios		0		Sin cambio
402	LIC001	Leggings de Cintura Alta y Ajuste Slim para Mujer	Licras		0		Sin cambio
403	ACE001	Mini Impresora Térmica	Accesorios	10	0		OK Actualizado
404	JEA001	Cargo Jeans de Cintura Alta para Mujer	Jeans		0		Sin cambio
405	ACE002	Pendientes Elegantes de Cereza Burdeo	Accesorios		0		Sin cambio
406	ACE003	Pendientes de Perlas de Flores Minimalistas	Accesorios		0		Sin cambio
407	LIC003	Pantalones de Yoga para Mujer	Licras		0		Sin cambio
408	VES002	Vestido sin mangas con estampado floral	Vestidos		0		Sin cambio
409	CHO002	Chompa Acolchada con Capucha para Mujer	Chompas	10	0		OK Actualizado
410	CHO003	Chompa Impermeable de Color Azul	Chompas		0		Sin cambio
411	VES004	Vestido Negro con Cuello en Pecho de Corazón	Vestidos		0		Sin cambio
412	CHO001	Chaqueta Sudadera con Cremallera	Chompas		0		Sin cambio
413	VES005	Vestido Rojo de Fiesta	Vestidos	20	0	60	OK Actualizado
414	CHA001	Chaqueta Crop Top Cuerina de Mujer	Chaquetas		0		Sin cambio
415	CHA002	Chaqueta Cuerina de Mujer	Chaquetas		0		Sin cambio
416	JEA004	Mom Jean	Jeans		0		Sin cambio
417	JEA003	Jeans de Denim Estrella de Cintura Alta	Jeans		0		Sin cambio
418	JEA002	Jeans de Denim Elásticos Casuales	Jeans		0		Sin cambio
419	CHA003	Chaqueta de Hombre	Chaquetas		0		Sin cambio
420							

Nota. Se presenta el formato de Actualización de Productos. Elaborado por: La autora.

4.3.5.6. Inventario

El módulo de Inventario, se visualiza una pantalla sobre el estado del stock de los productos que tenemos habilitados en la tienda online. Puedes consultar la información directamente desde WooCommerce, de forma que la información que se muestre será un reflejo del estado del inventario y no tendrás que ir a la administración de la plataforma tal como corresponde a lo dicho en la Figura 35.

El usuario tiene la posibilidad de consultar de un modo más general ver el listado de todos los productos que tenemos en nuestra base de datos o bien gestionar productos con stock y productos sin stock, contribuyendo a detectar productos críticos o, ahora, productos con bajo stock. La información se presenta en formato de tabla y se dispone de las siguientes columnas, SKU, nombre del producto, categoría, cantidad, precio y estado. El sistema impulsa la exportación del inventario que puedes gestionar con Excel. También puedes tener acceso a tu información y exportar este inventario. Esto se puede observar en la Figura 36. Este módulo nos permite gestionar el inventario y la toma de decisiones administrativas con información no sólo con datos actualizados, sino que también nos garantiza que es fiable.

Figura 35

Módulo de Inventario

SKU	NOMBRE	CATEGORÍA	STOCK	PRECIO	ESTADO
EP001	Paquete 600 Resistencias de Película Metálica 1/4w	Resistencias	20	12,00	En Stock
RE004	Convertidor Voltaje DC-DC Step-Down 3A MP1584EN	Conversores y reguladores	0	1,74	Sin Stock
RE003	Convertidor Voltaje DC-DC Step-Up 2A MT3608	Conversores y reguladores	489	1,74	En Stock
CO022	Módulo Transmisor y Receptor 433MHz RF	Comunicación	15	3,04	En Stock
CID001	HT12D	Circuitos Integrados	14	2,17	En Stock
MI027	Baquelita Perforada 5x7cm	Componentes básicos, Herramientas, Misceláneos	30	1,09	En Stock
SE059	Sensor de Presión Diferencial MPX5500DP	Fuerza y peso	9	33,48	En Stock
MI026	Baquelita Perforada 9x15cm	Componentes básicos, Herramientas, Misceláneos	30	3,04	En Stock
AS033	TFT 3.5 Touch Screen Shield	Shields, TFT	46	26,09	En Stock
CID002	HT12E	Circuitos Integrados	13	2,17	En Stock
AS032	TFT 2.8 Touch Screen Shield	Shields, TFT	0	9,52	Sin Stock
MD011	Mini Driver L298N	Controladores de motor	87	1,74	En Stock
MO021	Micro DC Motor 3V-6V 8000RPM	Motores y accesorios	113	1,30	En Stock
SE058	Módulo Galga Extensiométrica	Fuerza y peso	21	7,00	En Stock
SE057	Sensor Óptico Reflectivo TCRT5000	Distancia, proximidad y reflectivo	82	1,74	En Stock
MI007	40 Cables Dupont Hembra-Hembra 10cm	Cables	149	1,74	En Stock

Nota. Se presenta el módulo Inventario. Elaborado por: La autora.

Figura 36

Archivo Exportado Inventario

	A	B	C	D	E	F
1	SKU	NOMBRE	CATEGORÍA	STOCK	PRECIO	ESTADO
2	EP001	Paquete 600 Resistencias de Película Metálica 1/4w	Resistencias	20	12,00	En Stock
3	RE004	Convertidor Voltaje DC-DC Step-Down 3A MP1584EN	Conversores y reguladores	0	1,74	Sin Stock
4	RE003	Convertidor Voltaje DC-DC Step-Up 2A MT3608	Conversores y reguladores	489	1,74	En Stock
5	CO022	Módulo Transmisor y Receptor 433MHz RF	Comunicación	15	3,04	En Stock
6	CID001	HT12D	Circuitos Integrados	14	2,17	En Stock
7	MI027	Baquelita Perforada 5x7cm	Componentes básicos, Herramientas	30	1,09	En Stock
8	SE059	Sensor de Presión Diferencial MPX5500DP	Fuerza y peso	9	33,48	En Stock
9	MI026	Baquelita Perforada 9x15cm	Componentes básicos, Herramientas	30	3,04	En Stock
10	AS033	TFT 3.5 Touch Screen Shield	Shields, TFT	46	26,09	En Stock
11	CID002	HT12E	Circuitos Integrados	13	2,17	En Stock
12	AS032	TFT 2.8 Touch Screen Shield	Shields, TFT	0	9,52	Sin Stock
13	MD011	Mini Driver L298N	Controladores de motor	87	1,74	En Stock
14	MO021	Micro DC Motor 3V-6V 8000RPM	Motores y accesorios	113	1,30	En Stock
15	SE058	Módulo Galga Extensiométrica	Fuerza y peso	21	7,00	En Stock
16	SE057	Sensor Óptico Reflectivo TCRT5000	Distancia, proximidad y reflectivo	82	1,74	En Stock
17	MI007	40 Cables Dupont Hembra-Hembra 10cm	Cables	149	1,74	En Stock
18	MI005	40 Cables Dupont Macho-Macho 10cm	Cables	381	1,74	En Stock
19	ME009	Teclado Matricial 4x4 de Plástico	Teclados	27	5,24	En Stock
20	MD010	Driver DRV8825	Controladores de motor	45	3,04	En Stock

Nota. Se presenta el formato exportado de Inventario. Elaborado por: La autora.

4.3.5.7. Lista de Productos para Distribuidores

El módulo Listado de Productos para Distribuidores permite la generación de listados teniendo en cuenta unos parámetros específicos para los distribuidores, que ya consideran las reglas de negocio definidas para el cálculo del precio final y de los descuentos de forma automática, tal como se ve en la Figura 37. La aplicación de los productos es capaz de procesarlos y muestra los resultados en una pantalla que organizan lo que envuelven los datos como SKU, nombre del producto, stock, precio base, descuentos, etc.

Le permite al usuario visualizar la información de cada producto. Además del procedimiento los costos de cada uno, el módulo permite exportar a Excel los resultados, que puede ser utilizado en el desarrollo de procesos comerciales, en la elaboración de presupuestos y cotizaciones de los productos, tal como se ve en la Figura 38. Este mismo diseño nos ha servido para generar listados rápidos y fiables.

Figura 37

Módulo Lista de Distribuidores

SKU	NOMBRE DEL PRODUCTO	VARIACIÓN	STOCK	PVP	PRECIO COMPRA	GANANCIA	DESCUENTO (%)	DESCUENTO (\$)	PVD	OBSERVACIÓN	URL
EP003-680	Resistencia de película metálica 1/4 W	Valor: 680 Ω	15	0.03	0.0	1.0	20.0	0.01	0.02	COMPRA MÍNIMA 2 UNIDADES	https://moda.csuicups.org/producto/...
EP003-510	Resistencia de película metálica 1/4 W	Valor: 510 Ω	9	0.03	0.0	1.0	20.0	0.01	0.02	COMPRA MÍNIMA 2 UNIDADES	https://moda.csuicups.org/producto/...
EP003-330	Resistencia de película metálica 1/4 W	Valor: 330 Ω	995	0.03	0.0	1.0	20.0	0.01	0.02	COMPRA MÍNIMA 2 UNIDADES	https://moda.csuicups.org/producto/...
EP003-220	Resistencia de película metálica 1/4 W	Valor: 220 Ω	890	0.03	0.0	1.0	20.0	0.01	0.02	COMPRA MÍNIMA 2 UNIDADES	https://moda.csuicups.org/producto/...
EP003-200	Resistencia de película metálica 1/4 W	Valor: 200 Ω	2	0.03	0.0	1.0	20.0	0.01	0.02	COMPRA MÍNIMA 2 UNIDADES	https://moda.csuicups.org/producto/...
EP003-22	Resistencia de película metálica 1/4 W	Valor: 22 Ω	8	0.03	0.0	1.0	20.0	0.01	0.02	COMPRA MÍNIMA 2 UNIDADES	https://moda.csuicups.org/producto/...
EP003-10	Resistencia de película metálica 1/4 W	Valor: 10 Ω	10	0.03	0.0	1.0	20.0	0.01	0.02	COMPRA MÍNIMA 2 UNIDADES	https://moda.csuicups.org/producto/...
BL004-S	Blusa Crop Top Color Negro	Tipo: Talla S	10	5.0	5.0	0.0	0.0	0.0	5.0		https://moda.csuicups.org/producto/blusa-...
LIC002-XLN	Pantalones Casuales de Cintura Alta para Mujer con Bolsillo Lateral	Talla: XL Color: negro	10	18.0	20.0	-0.1111	0.0	0.0	18.0		https://moda.csuicups.org/producto/...
LIC002-XLP	Pantalones Casuales de Cintura Alta para Mujer con Bolsillo Lateral	Talla: XL Color: plomo	10	15.0	15.0	0.0	0.0	0.0	15.0		https://moda.csuicups.org/producto/...
LIC002-LN	Pantalones Casuales de Cintura Alta para Mujer con Bolsillo Lateral	Talla: L Color: negro	10	18.0	20.0	-0.1111	0.0	0.0	18.0		https://moda.csuicups.org/producto/...
LIC002-LP	Pantalones Casuales de Cintura Alta para Mujer con Bolsillo Lateral	Talla: L Color: plomo	10	15.0	15.0	0.0	0.0	0.0	15.0		https://moda.csuicups.org/producto/...
LIC002-MN	Pantalones Casuales de Cintura Alta para Mujer con Bolsillo Lateral	Talla: M Color: negro	10	18.0	0.0	1.0	20.0	3.6	14.4	COMPRA MÍNIMA 2 UNIDADES	https://moda.csuicups.org/producto/...
LIC002-MP	Pantalones Casuales de Cintura Alta para Mujer con Bolsillo Lateral	Talla: M Color: plomo	10	15.0	15.0	0.0	0.0	0.0	15.0		https://moda.csuicups.org/producto/...
LIC002-SN	Pantalones Casuales de Cintura Alta para Mujer con Bolsillo Lateral	Talla: S Color: negro	10	18.0	20.0	-0.1111	0.0	0.0	18.0		https://moda.csuicups.org/producto/...
LIC002-SP	Pantalones Casuales de Cintura Alta para Mujer con Bolsillo Lateral	Talla: S Color: plomo	10	15.0	15.0	0.0	0.0	0.0	15.0		https://moda.csuicups.org/producto/...

Nota. Se presenta el módulo Lista Distribuidores. Elaborado por: La autora.

Figura 38

Archivo exportado Lista de Distribuidores

A	B	C	D	E	F	G	H	I	J	K	L
SKU	NOMBRE DEL PRODUCTO	VARIACIÓN	STOCK	PVP	PRECIO COME	GANANCI	DESCUENTO	DESCUENTO	PVD	OBSERVACIÓN	URL
MI027	Baquellita Perforada 5x7cm		30	1,09	0	1	20	0,22	0,87	COMPRA MÍNIMA 2 UNIDADES	https://moda.csiuups.org/producto/baquellita-perforada-5x7cm/
SE059	Sensor de Presión Diferencial MPX5000DP		9	33,48	0	1	20	6,7	26,78	COMPRA MÍNIMA 2 UNIDADES	https://moda.csiuups.org/producto/sensor-de-presion-diferencial-mpx5000dp/
MI026	Baquellita Perforada 9x15cm		30	3,04	0	1	20	0,61	2,43	COMPRA MÍNIMA 2 UNIDADES	https://moda.csiuups.org/producto/baquellita-perforada-9x15cm/
AS033	TFT 3.5 Touch Screen Shield		46	26,09	0	1	20	5,22	20,87	COMPRA MÍNIMA 2 UNIDADES	https://moda.csiuups.org/producto/tft-3-5-touch-screen-shield/
CID002	HT12E		13	2,17	0	1	20	0,43	1,74	COMPRA MÍNIMA 2 UNIDADES	https://moda.csiuups.org/producto/ht12e/
MD011	Mini Driver L298N		87	1,74	0	1	20	0,35	1,39	COMPRA MÍNIMA 2 UNIDADES	https://moda.csiuups.org/producto/mini-driver-l298n/
MD021	Micro DC Motor 3V-6V 8000RPM		113	1,3	0	1	20	0,26	1,04	COMPRA MÍNIMA 2 UNIDADES	https://moda.csiuups.org/producto/micro-dc-motor-3v-6v-8000rpm/
SE008	Módulo Galga Extensiométrica		21	7	0	1	20	1,4	5,6	COMPRA MÍNIMA 2 UNIDADES	https://moda.csiuups.org/producto/modulo-galga-extensiometrica/
SE057	Sensor Óptico Reflectivo TCRT5000		82	1,74	0	1	20	0,35	1,39	COMPRA MÍNIMA 2 UNIDADES	https://moda.csiuups.org/producto/sensor-optico-reflectivo-tcrt5000/
MI007	40 Cables Dupont Hembra-Hembra 10cm		149	1,74	0	1	20	0,35	1,39	COMPRA MÍNIMA 2 UNIDADES	https://moda.csiuups.org/producto/40-cables-dupont-hembra-hembra-10cm/
MI005	40 Cables Dupont Macho-Macho 10cm		381	1,74	0	1	20	0,35	1,39	COMPRA MÍNIMA 2 UNIDADES	https://moda.csiuups.org/producto/40-cables-dupont-macho-macho-10cm/
ME009	Teclado Matricial 4x4 de Plástico		27	5,24	0	1	20	1,05	4,19	COMPRA MÍNIMA 2 UNIDADES	https://moda.csiuups.org/producto/teclado-matricial-4x4-de-plastico/
MD010	Driver DRV825		45	3,04	0	1	20	0,61	2,43	COMPRA MÍNIMA 2 UNIDADES	https://moda.csiuups.org/producto/driver-drv825/
ME010	Teclado Matricial 4x4		30	2,71	0	1	20	0,54	2,17	COMPRA MÍNIMA 2 UNIDADES	https://moda.csiuups.org/producto/teclado-matricial-4x4/
MI006	40 Cables Dupont Macho-Hembra 10cm		272	1,74	0	1	20	0,35	1,39	COMPRA MÍNIMA 2 UNIDADES	https://moda.csiuups.org/producto/40-cables-dupont-macho-hembra-10cm/
ME008	Teclado Matricial 5x4 de Membrana		26	4,35	0	1	20	0,87	3,48	COMPRA MÍNIMA 2 UNIDADES	https://moda.csiuups.org/producto/teclado-matricial-5x4-de-membrana/
MD005	Servo MG996R 360 Grados		45	13,04	0	1	20	2,61	10,43	COMPRA MÍNIMA 2 UNIDADES	https://moda.csiuups.org/producto/servo-mg996r-360-grados/
CO021	Módulo GPS NEO-6M		105	8,7	0	1	20	1,74	6,96	COMPRA MÍNIMA 2 UNIDADES	https://moda.csiuups.org/producto/modulo-gps-neo-6m/
FT006	FingerTech Silver Spark 16mm Gearmotor 33:1		9	27,87	0	1	20	5,57	22,3	COMPRA MÍNIMA 2 UNIDADES	https://moda.csiuups.org/producto/fingertech-silver-spark-16mm-gearmotor-33-1/
SE004	Sensor Voltage 25V		45	2,43	0	1	20	0,49	1,94	COMPRA MÍNIMA 2 UNIDADES	https://moda.csiuups.org/producto/sensor-voltage-25v/
AG020	Mega 2560 R3 CH340		167	21,57	0	1	20	4,31	17,26	COMPRA MÍNIMA 2 UNIDADES	https://moda.csiuups.org/producto/mega-2560-r3-ch340/
SE056	Sensor Sonido Digital		36	1,74	0	1	20	0,35	1,39	COMPRA MÍNIMA 2 UNIDADES	https://moda.csiuups.org/producto/sensor-sonido-digital/
AG019	Uno R3 CH340		474	10,87	0	1	20	2,17	8,7	COMPRA MÍNIMA 2 UNIDADES	https://moda.csiuups.org/producto/uno-r3-ch340/
MI002	Módulo generador de pulso NE555		83	1,52	0	1	20	0,3	1,22	COMPRA MÍNIMA 2 UNIDADES	https://moda.csiuups.org/producto/modulo-generador-de-pulso-ne555/
SE055	Sensor Laser		9	4,08	0	1	20	0,82	3,26	COMPRA MÍNIMA 2 UNIDADES	https://moda.csiuups.org/producto/sensor-laser/

Nota. Se presenta el archivo exportado de Lista de Distribuidores. Elaborado por: La autora.

CONCLUSIONES

La arquitectura que se ha ideado para el sistema PyWoo se basa en un criterio modular, el de la separación de responsabilidades, para poder facilitar la gestión administrativa de una tienda online implementado en WooCommerce. En la solución analizada, se ha visto que varios de los procesos estaban automatizados a través de diversos scripts independientes que si bien es cierto reducían las tareas manuales, tenían un alto grado de complejidad técnica, escasas interfaces robustas y requerían el trabajo exclusivamente de personal especializado. Esta situación limitaba el uso de la tienda y condicionaba en gran medida el uso de conocimiento técnico para trabajar en el entorno administrativo, resultando complicado al mismo tiempo poder delegar las actividades administrativas hacia otros usuarios.

A la vez, se ha diseñado e implementado una plataforma tecnológica modular, denominada PyWoo, para centralizar y simplificar la ejecución de los procesos administrativos más relevantes a través de una interfaz gráfica, clara y accesible. La arquitectura del sistema está organizada en capas bien definidas, y existe una capa de presentación que ejecuta la interacción con el usuario, una capa de control que ejecuta la lógica de negocio de las operaciones y una capa de integración que ejecuta la comunicación con la API REST de WooCommerce. La separación en capas permite que cada componente pueda evolucionar de forma independiente sin afectar al conjunto del software.

La modularidad de la propuesta se observa tanto en la implementación de módulos que controlan funcionalidades específicas, como en la elaboración de reportes de ventas o la actualización de productos, como en la implementación de módulos que tamizan funcionalidades específicas como la gestión de inventario o la elaboración de listas de distribuidores. De esta forma, cada módulo ha sido diseñado como una unidad funcional autónoma, la cual reutiliza componentes comunes como por ejemplo el cliente de conexión a WooCommerce y los mecanismos de configuración centralizados. Esta estrategia ha permitido, por un lado, reducir la redundancia del código y por otro, mantener un comportamiento homogéneo de la experiencia de usuario en todos los módulos. En cuanto a la fabricación de la interfaz gráfica, esta fue ejecutada para que existe una posible reducción de la curva del aprendizaje por parte de personas que dispongan de diferentes grados de experiencia técnica y puedan llevar a cabo tareas administrativas sin la necesidad de recurrir a scripts o configuración avanzada.

RECOMENDACIONES

Antes de empezar con la ejecución en tiendas reales de la aplicación de PyWoo se recomienda que se realicen pruebas adicionales, verifique que funcione cada una de las interfaces, los módulos y demás para la continuidad del negocio, es una forma de analizar como el sistema se comporta con un gran o pequeño porcentaje de datos dependiendo del negocio. Esto va a permitir realizar ajustes que se necesiten hacer en la arquitectura del sistema y de esa manera asegurar que la ampliación sea capaz de adaptarse a distintas tiendas en línea y no afecte la integridad de la información y el funcionamiento del negocio.

En caso de que sea necesario incorporar nuevas funcionalidades, se recomienda aprovechar la arquitectura basada en modelo, vista, controlador, estructura utilizada en el desarrollo del proyecto. Este enfoque favorece la incorporación o eliminación de módulos de manera ordenada y eficiente. Asimismo, el diseño prioriza los procesos administrativos más frecuentes en los comercios electrónicos, lo que contribuye a que la aplicación sea intuitiva, escalable y alineada, garantizando que el sistema pueda adaptarse a distintas tiendas virtuales sin comprometer la integridad de los datos ni la continuidad operativa.

Al utilizar API REST de WooCommerce es recomendable renovar periódicamente las claves de acceso, idealmente cada tres meses, como medida preventiva de seguridad. De igual manera, se sugiere realizar revisiones periódicas del sistema para asegurar su correcto funcionamiento y mejorar la experiencia del usuario. La integración de múltiples módulos permite reducir la dependencia del acceso directo a la consola de WordPress, convirtiendo la aplicación en una herramienta de apoyo para la toma de decisiones administrativas, sin sacrificar la facilidad de uso ni la continuidad del servicio.

REFERENCIAS

- DigitalOcean. (2025). *Cómo instalar Linux, Nginx, MySQL y PHP (pila LEMP) en Ubuntu*.
<https://www.digitalocean.com/community/tutorials/como-instalar-lemp-ubuntu>
- GitHub. (2026). *SamSami25—Repositories*. <https://github.com/SamSami25>
- MariaDB Foundation. (2025, 6 de noviembre). *MariaDB Documentation*.
<https://mariadb.com/docs>
- NGINX, Inc. (2025). *NGINX: Documentación oficial del servidor web*. <https://nginx.org>
- PHP Group. (2025). *Manual de PHP*. <https://www.php.net/manual/es/index.php>
- Python Software Foundation. (2025). *Entornos virtuales y paquetes (venv)*.
<https://docs.python.org/3/tutorial/venv.html>
- Qt Group. (2025). *Qt para Python*. <https://doc.qt.io/qtforpython-6/>
- Raicem. (2025, 29 de octubre). *WooCommerce Server Recommendations*. WooCommerce.
<https://woocommerce.com/document/server-requirements/>
- Sommerville, I. (2016). *Software engineering* (10ª ed.). Pearson.
- Vásquez, K. (2025). *Automatización de procesos administrativos y adopción de nuevas tecnologías: Su influencia en la eficiencia organizacional y la retención de clientes en empresas modernas*. *Ciencias Tecnológicas*, 33(49), 47–56.
[https://doi.org/10.26820/recimundo/9.\(esp\).mayo.2025.47-56](https://doi.org/10.26820/recimundo/9.(esp).mayo.2025.47-56)
- WooCommerce. (2025). *WooCommerce REST API*.
<https://woocommerce.com/document/woocommerce-rest-api/>
- WordPress.org. (2025). *Documentation*. <https://wordpress.org/documentation/>

XlsxWriter. (2025). *XlsxWriter: A Python module for creating Excel XLSX files (versión 3.2.9)*.
<https://xlsxwriter.readthedocs.io/>