

Trabajo de grado



**SISTEMA DE INFORMACIÓN DE FERRETERÍAS BAJO RÉGIMEN SIMPLE DE
TRIBUTACIÓN**

Aldemar Vivas Amarillo - 1968062-3743

Documento presentado al programa
Ingeniería en sistemas
como requisito para optar por el título de
Ingeniero

Noviembre - 2024
Facultad de Ingeniería

Trabajo de grado Universidad del valle

**SISTEMA DE INFORMACIÓN DE FERRETERÍAS BAJO RÉGIMEN SIMPLE DE
TRIBUTACIÓN**

Aldemar Vivas Amarillo - 1968062-3743

Director: Msg. Juan Pablo Pinillos Reina

Documento presentado al programa
Ingeniería en sistemas
como requisito para optar por el título de
Ingeniero

Noviembre - 2024
Facultad de Ingeniería

Índice general

1. Resumen.....	2
2. Abstract	3
3. Introducción.....	4
1 Descripción general	4
2 Problema de investigación.....	6
2.1 Descripción del problema	6
2.2 Formulación del problema.....	10
2.3 Justificación del problema	11
2.4 Marco contextual	15
4. Marco referencial	21
0.1 Marco Teórico.....	21
0.2 Estado del Arte	25
0.3 Antecedentes	30
0.4 Marco Conceptual	34
0.5 Estándar IEEE 12207	38
1 Alcance del proyecto.....	39
1.1 Declaración de Alcance.....	39

1.2	Objetivos	42
1.3	Metodología de Desarrollo de Software	46
1.4	Metodología de Gestión de Actividades	50
5.	Desarrollo de proyecto.....	52
1	Planificación.....	54
1.1	Principios de Gestión de Calidad Aplicados al Producto	54
1.2	Plan de Comunicaciones con el Cliente	55
1.3	Criterios de Aceptación del Cliente	57
1.4	Fases del Proyecto y Aplicación del Estándar IEEE 12207	58
1.5	Procesos generales de ferreterías.....	60
1.6	Particularización al RST	61
1.7	Módulos	63
1.8	Requerimientos	70
2	Diseño	76
2.1	Principios tenidos en cuenta en el diseño	77
2.2	Tecnologías.....	80
2.3	Diagramas.....	89
2.4	Mockups.....	97
3	Codificación	100
3.1	Cumplimiento de requerimientos	102
3.2	Métricas	109
4	Pruebas	112
4.1	Contról de versiones	112
4.2	Plan de Pruebas	114
4.3	Casos de prueba	115

4.4	Pruebas Unitarias	119
4.5	Pruebas de Integración	121
4.6	Pruebas Funcionales	122
4.7	Pruebas de Rendimiento	122
4.8	Pruebas de usabilidad	124
4.9	Resultados de la encuesta	131
4.10	Resultados y Observaciones	133
4.11	Auditorías de calidad	134
6.	Resultados	142
1	Resultados obtenidos	142
1.1	Desarrollo Integral del Sistema de Información	142
1.2	Seguridad y Control de Acceso	142
1.3	Pruebas de Usabilidad y Validación con Usuarios Finales	143
1.4	Validación Técnica y Cumplimiento de Requisitos	143
1.5	Módulo de Ventas y Gestión de Clientes	144
1.6	Optimización del Módulo de Inventarios	144
1.7	Gestión de Proveedores y Reabastecimiento	145
1.8	Módulo de Costos y Administración de Gastos	145
2	Trabajos futuros	146
2.1	Integración de Facturación Electrónica	146
2.2	Desarrollo de un Módulo de Análisis Predictivo	146
2.3	Expansión de Funcionalidades de Reportes	147
2.4	Desarrollo de una Aplicación Móvil	147
2.5	Implementación de un Módulo de Fidelización de Clientes	147
2.6	Optimización del Rendimiento y Seguridad del Sistema	148

3	Dificultades.....	149
3.1	Adaptación al Régimen Simple de Tributación.....	149
3.2	Implementación de Reportes de Ventas y Costos	149
3.3	Manejo de Inventario y Restock de Productos	150
3.4	Desafíos con la Interfaz de Usuario.....	150
3.5	Pruebas de Desempeño y Escalabilidad.....	150
4	Comparativa entre los resultados obtenidos y los objetivos planteados	152
7.	Conclusiones	155
8.	Bibliografía.....	158
1	Referencias	158
2	Encuesta del trabajo de campo	164
2.1	Checklist para la Validación de Entregas de Requerimientos.....	165
9.	Anexos.....	166
1	Requerimientos Funcionales.....	166
2	Repositorio	171
3	Instrumento de evaluación de satisfacción del usuario	172
4	Resultados de encuesta	175
5	Mockups	179
6	Evidencias	181

Índice de tablas

4.1	Resultados esperados de cada objetivo. Fuente: Elaboración propia.	45
4.2	Evaluación de Metodologías de Desarrollo de Software. Fuente: Elaboración Propia.	48
5.1	REQUERIMIENTO 1	71
5.2	REQUERIMIENTO 2	72
5.3	REQUERIMIENTO 3	72
5.4	Requerimientos no funcionales	75
5.6	Registro de interacciones de los usuarios durante las pruebas de usabilidad	129
6.1	Resultados esperados de cada objetivo. Fuente: Elaboración propia.	154
9.1	REQUERIMIENTO 4	166
9.2	REQUERIMIENTO 5	166
9.3	REQUERIMIENTO 6	167
9.4	REQUERIMIENTO 7	167
9.5	REQUERIMIENTO 8	167
9.6	REQUERIMIENTO 9	168
9.7	REQUERIMIENTO 10	168
9.8	REQUERIMIENTO 11	168
9.9	REQUERIMIENTO 12	169
9.10	REQUERIMIENTO 13	169
9.11	REQUERIMIENTO 14	169
9.12	REQUERIMIENTO 15	170

9.13 REQUERIMIENTO 16.....	170
9.14 REQUERIMIENTO 17.....	170

ÍNDICE DE FIGURAS

4.1	Tabla adaptada de selección de metodología de desarrollo de software	47
5.1	Tabla de cálculo de impuesto RST.	61
5.2	Diagrama ilustrativo de arquitectura cliente servidor	88
5.3	Diagrama de casos de uso para el usuario vendedor.....	90
5.4	Diagrama de casos de uso para el usuario inventario.....	90
5.5	Diagrama de casos de uso para el usuario administrador	91
5.6	Diagrama de secuencia de ventas	93
5.7	Diagrama de secuencia de inventario	94
5.8	Diagrama de secuencia de clientes	95
5.9	Diagramas de entidad de relación	96
5.10	Vista de los mockups del módulo de Inventario.....	97
5.11	Vista de los mockups de edición de colaborador.	98
5.12	Vista de los mockups del módulo de ventas.....	99
5.13	Captura de tablero Kanban del proyecto.....	106
5.14	Captura de fragmento de código del proyecto.....	107
5.15	Captura de fragmento de código del proyecto.....	108
5.16	Captura de registro de commit de GitHub.....	110
5.17	Captura de las pruebas unitarias del módulo de ventas.....	120
5.18	Captura de las pruebas unitarias del módulo de inventarios	120
5.19	Captura de las pruebas unitarias del módulo de contabilidad	121
5.20	Gráficas de prueba de rendimiento de Locust	123

5.21	Reporte de prueba de rendimiento de Locust.....	124
5.22	Respuesta de pregunta de encuesta de usabilidad	131
5.23	Respuesta de pregunta de encuesta de usabilidad	131
5.24	Respuesta de pregunta de encuesta de usabilidad	132
9.1	Respuesta de pregunta de encuesta de usabilidad	175
9.2	Respuesta de pregunta de encuesta de usabilidad	175
9.3	Respuesta de pregunta de encuesta de usabilidad	176
9.4	Respuesta de pregunta de encuesta de usabilidad	176
9.5	Respuesta de pregunta de encuesta de usabilidad	177
9.6	Respuesta de pregunta de encuesta de usabilidad	177
9.7	Respuesta de pregunta de encuesta de usabilidad	178
9.8	Evidencia de pruebas de usabilidad con usuarios.....	181
9.9	Evidencia de pruebas de usabilidad con usuarios.....	182
9.10	Evidencia de pruebas de usabilidad con usuarios.....	183
9.11	Evidencia de pruebas de usabilidad con usuarios.....	184
9.12	Evidencia de pruebas de usabilidad con usuarios.....	185
9.13	Evidencia de pruebas de usabilidad con usuarios.....	186

Resumen

El presente trabajo de grado tuvo como objetivo el desarrollo de un Sistema de Información para Ferreterías bajo el Régimen Simple de Tributación, diseñado para mejorar la gestión administrativa, financiera y contable de las ferreterías que operan bajo este régimen en Colombia. Este sistema permite una optimización en el manejo de inventarios, ventas, costos, clientes, proveedores y facturación, cumpliendo con las normativas fiscales del régimen tributario simplificado. El proyecto se basa en la implementación de tecnologías como Django y PostgreSQL, ofreciendo una plataforma intuitiva y eficiente para la gestión diaria de las ferreterías. Además, incorpora módulos clave, tales como la generación de reportes de ventas, el control de productos y el cálculo de impuestos, adaptados a las características particulares del régimen simple de tributación. Este sistema no solo busca mejorar la eficiencia operativa de las ferreterías, sino también garantizar el cumplimiento de las obligaciones fiscales de manera ágil y precisa. El desarrollo de este sistema contribuye a la digitalización y automatización de procesos en pequeñas y medianas empresas del sector ferretero, facilitando su adaptación a las exigencias fiscales del país.

Palabras clave: Ferretería, Régimen simple de tributación, sistema de información, inventario, Módulos.

Abstract

The objective of this thesis is to develop an Information System for Hardware Stores under the Simple Tax Regime, designed to improve the administrative, financial and accounting management of hardware stores operating under this regime in Colombia. This system allows for optimization in the management of inventories, sales, costs, customers, suppliers and billing, complying with the tax regulations of the simplified tax regime. The project is based on the implementation of technologies such as Django and PostgreSQL, offering an intuitive and efficient platform for the daily management of hardware stores. In addition, it incorporates key modules, such as the generation of sales reports, product control and tax calculation, adapted to the particular characteristics of the simple tax regime. This system not only seeks to improve the operational efficiency of hardware stores, but also to ensure compliance with tax obligations in an agile and precise manner. The development of this system contributes to the digitalization and automation of processes in small and medium-sized companies in the hardware sector, facilitating their adaptation to the country's tax requirements.

Keywords: Hardware store, Simple tax regime, information system, inventory, Modules.

Introducción

1 Descripción general

En la era digital actual, la tecnología se ha convertido en un factor de transformación esencial, impulsando la eficiencia, la competitividad y la optimización en diversos sectores económicos. Todas las industrias, independientemente de su relación directa con el desarrollo de software, han adoptado soluciones tecnológicas para mejorar sus procesos. Esta constante evolución empresarial ha convertido la adaptación tecnológica en un requisito fundamental para el éxito.

Sin embargo, no todos los sectores han logrado integrarse plenamente en esta transformación digital. Un caso particular es el de las ferreterías que operan bajo el régimen simple de tributación. Estas empresas desempeñan un papel crucial en la economía, proporcionando materiales esenciales para la construcción y renovación, pero enfrentan dificultades significativas debido a la falta de herramientas tecnológicas adecuadas. La ausencia de soluciones accesibles y especializadas dificulta la gestión eficiente del inventario, la atención al cliente y el seguimiento preciso de sus transacciones y cuentas.

En este contexto, el presente estudio se enfoca en analizar esta problemática y desarrollar una solución tecnológica que permita a estas ferreterías optimizar sus operaciones. A lo largo del documento, se presentará un diagnóstico detallado del estado actual de la digitalización en este sector, seguido de una propuesta de sistema de información adaptado a sus necesidades. Se abordarán aspectos clave como la gestión de inventarios, ventas y facturación, además de evaluar el impacto que la implementación de esta solución podría tener en la competitividad y sostenibilidad de estas empresas.

Así, este estudio no solo busca cerrar la brecha tecnológica que limita a las ferreterías de régimen simple, sino también contribuir al fortalecimiento de un sector esencial para la economía. Con ello, se espera que estas empresas puedan aprovechar las ventajas de la tecnología para mejorar su desempeño y responder eficazmente a las exigencias del mercado actual.

2 Problema de investigación

2.1 Descripción del problema

En muchas ocasiones, existen ferreterías cabo sus procesos de manera análoga y la falta de modernización de los mismos. En muchos casos, se utiliza un libro de contabilidad donde se registran todos los movimientos financieros de la empresa, desde pagos a proveedores hasta movimientos de caja menor. Este enfoque hace que el proceso sea ineficiente y propenso a errores, ya que al ser en papel, no hay nada parametrizado y puede cambiarse en cualquier momento, sin mencionar que este método es propenso a pérdidas de información.

Un método comúnmente utilizado en las ferreterías pequeñas es el cálculo mental, que es totalmente informal y altamente propenso a pérdidas económicas al no tener un registro de las necesidades y gastos del establecimiento. Además, es poco eficiente en cuanto al conocimiento de los productos que se están agotando o no se están vendiendo.

Otra herramienta ampliamente utilizada para mantener la contabilidad, existencias y pagos de los establecimientos es Microsoft Excel, especialmente en su versión de escritorio. Este software ofrece una hoja de cálculo compuesta por filas y columnas que forman celdas, en las cuales se puede ingresar información tanto numérica como alfabética. Su flexibilidad permite que se adapte a una gran variedad de necesidades dentro de las ferreterías, desde el control de inventario hasta la gestión de ventas y reportes financieros.

Sin embargo, su aprovechamiento óptimo depende del conocimiento del usuario, y aquí es donde surge un desafío. Los principales usuarios en las ferreterías suelen ser administradores, vendedores y empleados encargados de la gestión operativa y financiera del negocio. En su mayoría, se trata de personas con experiencia en el sector ferretero, pero con conocimientos informáticos limitados. Muchos han aprendido a manejar Excel de manera autodidacta o con conocimientos básicos transmitidos por colegas, lo que genera un uso restringido de sus funcionalidades.

La edad de los usuarios varía, pero es común encontrar administradores entre los 40 y 60 años, quienes prefieren métodos tradicionales como registros en papel o sistemas muy básicos en Excel. Por otro lado, los empleados más jóvenes (20 a 35 años) suelen tener mayor familiaridad con herramientas digitales, pero no siempre cuentan con formación específica en gestión empresarial o análisis de datos. Esta brecha de conocimiento influye en la eficiencia con la que se usa Excel en estos negocios, pues muchas de sus funciones avanzadas —como el uso de tablas dinámicas, automatización con macros o la validación de datos— quedan infrautilizadas.

Esta falta de información, que debería ser consultable tanto de manera gráfica como en forma de datos, disminuye la satisfacción del cliente, dificultando el crecimiento del negocio.

Finalmente, existe el método más óptimo que involucra el uso de software especializado para negocios, con el objetivo de facilitar su administración. Este recurso ofrece una amplia variedad de herramientas que permiten mejorar la satisfacción del cliente, fomentando así el crecimiento de estas empresas.

El problema en el contexto de los softwares diseñados específicamente para ferreterías que operan bajo el régimen simple de tributación plantea un desafío significativo debido a la falta de herramientas tecnológicas accesibles en términos de su alto costo comercial, además de su alta curva de aprendizaje.

Estas ferreterías, que prestan servicios a clientes bajo este régimen, suelen encontrarse en una situación desfavorecida al ver que las soluciones existentes en el mercado del software si bien cumplen con los requerimientos necesarios para hacer que una ferretería opere de una manera correcta, estas soluciones carecen de simplicidad que hacen que este proceso sea cómodo para el tipo de usuarios que común se encuentran en las ferreterías.

Este vacío tecnológico se traduce en una serie de complicaciones que afectan tanto la gestión interna de estas ferreterías como la experiencia del cliente. La falta de un sistema adecuado de gestión de inventario puede llevar a la pérdida de ventas debido a la falta de productos disponibles o al exceso de existencias que ocupan espacio y generan costos innecesarios. Además, la carencia de un sistema eficiente de puntos de venta puede hacer que las transacciones sean lentas y propensas a errores, afectando la satisfacción del cliente.

Es importante señalar que el seguimiento de cuentas y la contabilidad son aspectos críticos de cualquier negocio, pero muchas ferreterías como pueden ser el caso de las mas accesibles al publico general (ferreterías de barrio y negocios emergentes) En el régimen simple de tributación carecen de las herramientas necesarias para llevar un registro adecuado de sus transacciones y cuentas pendientes. Esto puede dar lugar a problemas financieros y dificultades para mantener un flujo de efectivo constante. Todo

esto se traduce en una experiencia deficiente tanto para los propietarios de las ferreterías como para los clientes que dependen de estos establecimientos para sus necesidades de construcción y reparación.

Quizás, si existiera una solución tecnológica diseñada específicamente para las necesidades de las ferreterías que operan bajo el régimen simple de tributación, muchas de estas problemáticas podrían resolverse de manera efectiva. Un software intuitivo, accesible y optimizado para este tipo de negocios permitiría mejorar la eficiencia operativa, reducir errores en los registros financieros y agilizar las transacciones en los puntos de venta.

2.2 Formulación del problema

¿Cómo apoyar la gestión de procesos de las ferreterías de régimen simple de tributación ?

2.3 Justificación del problema

La falta de información actualizada sobre el estado del sector ferretero motivó la realización de un análisis de campo, cuyo propósito fue contextualizar el problema y definir una solución alineada con las necesidades específicas del sector en Tuluá. Este análisis desempeñó un papel clave en el proceso metodológico, ya que permitió identificar datos relevantes sobre los desafíos operativos y el impacto de la competencia en la industria.

El trabajo de campo permitió visualizar con mayor claridad el problema a abordar, proporcionando datos relevantes sobre la importancia y los desafíos del sector ferretero en Tuluá. Uno de los hallazgos clave es la contribución económica de estas empresas al comercio local. Según datos de la Cámara de Comercio de Tuluá, las ferreterías y los establecimientos relacionados con la construcción representan aproximadamente el 6 % del total de empresas en la ciudad. Esta cifra demuestra el papel fundamental que desempeñan en la actividad económica y su impacto en el Producto Interno Bruto (PIB) local.

Además de su relevancia económica, las ferreterías son una fuente importante de empleo en la comunidad. En promedio, cada una de estas empresas cuenta con dos o tres empleados, lo que, en términos de impacto social, representa un sustento directo para aproximadamente 240 personas en la ciudad. Este dato resalta la importancia de fortalecer el sector, ya que cualquier mejora en su eficiencia operativa puede traducirse en una mayor estabilidad laboral y mejores condiciones para los trabajadores.

Sin embargo, a pesar de su importancia, las ferreterías enfrentan desafíos

operativos significativos. La investigación identificó problemas recurrentes en la gestión de inventarios, dificultades para mejorar las ventas y carencias en la administración de clientes. En particular, una encuesta realizada a 20 ferreterías en Tuluá reveló que el 70 % de ellas considera que la gestión de inventarios es uno de los principales obstáculos para su operación eficiente. Esta situación resalta la necesidad de implementar herramientas tecnológicas que optimicen estos procesos y reduzcan la carga administrativa.

Por otro lado, la competencia en el sector ha ido en aumento, representando un reto adicional para las ferreterías locales. Datos recopilados de dueños y empleados indican que el número de establecimientos ferreteros en Tuluá ha crecido un 16 % en los últimos cinco años. Este incremento refleja un mercado cada vez más saturado, donde la presencia de grandes cadenas comerciales dificulta la sostenibilidad de los negocios más pequeños. Ante este panorama, mejorar la eficiencia operativa y desarrollar estrategias de diferenciación se vuelven factores clave para mantener la competitividad en el sector.

La Ingeniería de Sistemas puede contribuir significativamente a resolver este problema mediante el diseño e implementación de una solución tecnológica integral que aborde los desafíos identificados. Por medio de la automatización de procesos, un sistema de información puede mejorar la gestión de inventarios, simplificar las operaciones administrativas y reducir errores humanos. Además, la incorporación de herramientas de análisis de datos permitirá a los dueños de ferreterías comprender mejor sus ventas, identificar productos de alta rotación y optimizar su inventario. Por otro lado, la solución también facilita la conectividad y la escalabilidad, lo que posibilita a los negocios adaptarse a las demandas del mercado y evolucionar con el tiempo.

En este sentido, la solución desarrollada en este trabajo se enfoca en un sistema de información especializado para el sector ferretero bajo el régimen simple de tributación. Esto se justifica porque responde directamente a las necesidades específicas de las ferreterías de Tuluá, basándose en el trabajo de campo realizado. Asimismo, se trata de una solución viable, ya que la tecnología necesaria para su desarrollo está al alcance de las ferreterías locales y se adapta a sus limitaciones económicas y tecnológicas. Aunado a ello, el sistema está diseñado para ser escalable, lo que asegura su capacidad de crecimiento y adaptación a las necesidades futuras del sector.

Por otro lado, si no se hubiera abordado este problema, las ferreterías locales aún carecerían de una solución que les permitiera reducir altos niveles de ineficiencia en la gestión administrativa y fiscal, lo que continuaría generando errores en el cumplimiento tributario y una mala gestión de inventarios. Esto podría haber derivado en una mayor presión competitiva de las grandes cadenas comerciales y, en consecuencia, en el cierre de negocios pequeños que no logran adaptarse al entorno cambiante del mercado

Lo novedoso de este proyecto radica en su enfoque integral y adaptado al Régimen Simple de Tributación. Un aspecto que no ha sido suficientemente abordado en las soluciones comerciales existentes es la integración efectiva de la gestión fiscal simplificada para empresas acogidas a este régimen en Colombia.

La mayoría de las soluciones disponibles se centran en procesos generales de contabilidad o gestión de inventarios, sin considerar las especificidades fiscales de este modelo tributario. Este vacío obliga a los empresarios ferreteros a recurrir a diversas herramientas, muchas veces no interconectadas, para cumplir con sus obligaciones

fiscales, lo que genera ineficiencias operativas y un mayor riesgo de error en el cumplimiento tributario.

En particular, este sistema no solo automatiza procesos, sino que también incorpora una gestión fiscal simplificada, orientada a facilitar el cumplimiento tributario de los dueños de ferreterías. Además, incluye una gestión de inventarios en tiempo real, que proporciona una visión actualizada y precisa de las existencias.

De igual manera, el sistema ofrece herramientas para el análisis operativo y competitivo, permitiendo evaluar el impacto de la competencia y diseñar estrategias de diferenciación.

Por lo tanto, esta solución llena vacíos importantes en las herramientas actuales, ya que aborda de manera integral los desafíos específicos que enfrentan las ferreterías bajo el Régimen Simple de Tributación, un aspecto que ha sido insuficientemente tratado por los sistemas comerciales disponibles. Las soluciones existentes, aunque efectivas en ciertos aspectos de gestión administrativa, no logran integrar de manera eficiente la gestión fiscal simplificada y la optimización de inventarios en tiempo real, dos aspectos clave para las ferreterías en Tuluá.

2.4 Marco contextual

El sector ferretero en Colombia se ha consolidado como un pilar fundamental en el desarrollo de la industria de la construcción, remodelación y mantenimiento. Este sector experimenta un crecimiento constante debido a la expansión de la construcción tanto residencial como industrial, que genera una demanda estable de materiales y herramientas ferreteras.

Con el fin de proporcionar un contexto sólido y contar con información actualizada que sirviera de base para el proceso metodológico del proyecto, además de comprender en profundidad las necesidades de los usuarios y las particularidades de su operativa diaria, se decidió llevar a cabo un trabajo de campo. Esta fase resultó crucial, ya que permitió obtener datos reales sobre el funcionamiento de las ferreterías y los desafíos que enfrentan en la gestión de sus procesos administrativos y comerciales. Para ello, se realizó un estudio con una muestra representativa de 20 ferreterías en la ciudad de Tuluá, Valle del Cauca.

El trabajo de campo no solo aportó insumos clave para la toma de decisiones en el desarrollo del sistema de información, sino que también permitió identificar problemas recurrentes que enfrentan estos establecimientos, como la falta de automatización en sus operaciones, errores en el cálculo de impuestos y dificultades en el control de stock.

Gracias a este análisis detallado, el diseño del sistema pudo orientarse de manera más efectiva hacia la solución de problemas reales, asegurando que la plataforma desarrollada respondiera de manera eficiente a las necesidades específicas del sector ferretero en el contexto del RST.

Para garantizar que la entrevista proporcionara datos útiles y específicos, se diseñó un cuestionario basado en criterios clave. En primer lugar, se priorizó la relevancia para el proyecto, formulando preguntas orientadas a conocer las dificultades actuales de las ferreterías. Además, se aseguró la claridad y concisión de cada pregunta, evitando términos técnicos complejos para facilitar la comprensión y permitir respuestas detalladas. También se optó por preguntas abiertas en lugar de preguntas cerradas, con el fin de otorgar flexibilidad en las respuestas y evitar limitaciones en la información recolectada. Finalmente, se diseñaron preguntas que garantizaran una cobertura integral de los aspectos fundamentales de los resultados de la gestión ferretera, extrayendo datos cuantitativos de la entrevista.

Con la finalidad de dar validez metodológica a este instrumento la entrevista, la entrevista fue sometida a la revisión de un experto, el cual dio su aval para continuar con el proceso y así poder comenzar la encuesta a los administradores del sector ferretero.

Además del trabajo de campo anteriormente mencionado, se buscó información relevante en bases de datos académicas, estudios recientes y estadísticas con el fin de complementar la información y de esta manera conocer el contexto actual del sector ferretero.

A continuación, se procede a dar el contexto de la situación actual del sector ferretero bajo las fuentes mencionadas anteriormente. Cabe mencionar que se puede evidenciar las preguntas realizadas en la sección de anexos.

Los empleados de ferreterías conforman un grupo diverso en cuanto a edad, género y nivel educativo, aunque existen ciertos patrones comunes en el sector. En su mayoría,

se trata de hombres adultos jóvenes y de mediana edad, con edades que oscilan entre los 25 y 50 años. Esto se debe a que muchas de las tareas dentro de una ferretería requieren esfuerzo físico, conocimientos técnicos sobre herramientas y materiales de construcción, así como experiencia en trabajos manuales. Sin embargo, se ha observado presencia de mujeres en funciones administrativas y de atención al cliente.

En términos de formación académica, gran parte de los empleados de ferreterías cuentan con estudios de educación secundaria, y en algunos casos, formación técnica en áreas relacionadas con la construcción, la electricidad o la mecánica. Muchos de ellos han adquirido su conocimiento a través de la experiencia laboral, lo que les permite asesorar a los clientes con información práctica y soluciones efectivas

Otro aspecto relevante es la experiencia laboral previa. La mayoría de los empleados de ferreterías han trabajado anteriormente en sectores relacionados, como la construcción, la carpintería, la plomería o la electricidad. Esto les permite tener un conocimiento práctico sobre los productos y su uso, facilitando una atención al cliente más efectiva. En contraste, los trabajadores más jóvenes o recién ingresados suelen iniciar con tareas básicas como organización de inventario y manejo de caja, para luego adquirir mayores responsabilidades con el tiempo.

En cuanto a la jornada laboral, es común que los empleados trabajen en horarios extendidos, incluyendo fines de semana, debido a que las ferreterías suelen operar en horarios amplios para atender a trabajadores del sector de la construcción y otros clientes. Esta dinámica exige que los empleados sean personas activas, con buena disposición para el trabajo físico y la interacción con el público.

Finalmente, los empleados de ferreterías destacaron por su capacidad de servicio al cliente y habilidades de negociación. La venta de productos ferreteros no solo implica registrar compras, sino también brindar asesoramiento, explicar diferencias entre materiales y herramientas, y en muchos casos, ayudar a los clientes a encontrar la mejor solución según su necesidad y presupuesto. Esta cercanía con el cliente y su conocimiento empírico hacen de los empleados de ferreterías una parte fundamental del éxito de estos negocios.

A pesar de los desafíos derivados de la volatilidad económica y la inflación, la industria sigue mostrando signos positivos de recuperación y expansión, especialmente en las áreas urbanas donde se desarrollan proyectos de gran envergadura aportando un 2,5 % al Producto Interno Bruto del país según FENALCO.

Una característica destacada del sector ferretero colombiano es la diversificación de su oferta de productos. Tradicionalmente centrado en herramientas manuales y materiales de construcción, el mercado ha evolucionado para incluir productos especializados como equipos de protección personal, tecnología para la construcción y soluciones ecológicas. Esta expansión de la oferta responde a las crecientes necesidades de los consumidores, quienes buscan productos más variados y específicos para cada tipo de proyecto. De esta manera, las ferreterías han logrado adaptarse a las nuevas demandas del mercado y ofrecer soluciones más completas a sus clientes.

La competencia en el sector ferretero colombiano es alta, con una notable presencia de grandes cadenas de ferreterías como Homecenter y Easy, que dominan el mercado gracias a su capacidad para ofrecer precios competitivos, una amplia gama de productos y promociones atractivas. Sin embargo, las ferreterías independientes siguen siendo un

actor relevante, especialmente en regiones más pequeñas o rurales, donde mantienen su influencia. Esta competencia ha impulsado la mejora continua en la calidad del servicio y la oferta de productos, lo que beneficia a los consumidores al tener acceso a una mayor variedad y mejores condiciones de compra.

Un aspecto crucial para las empresas del sector ferretero, en particular las pequeñas y medianas, es la implementación del régimen simple de tributación, que ha sido introducido como una medida para simplificar los procesos fiscales y reducir la carga tributaria. Este régimen, destinado principalmente a los pequeños empresarios, permite una tributación más sencilla y accesible. Para las ferreterías que operan bajo este régimen, se presenta una oportunidad para reducir costos fiscales, lo que les permite ser más competitivas frente a grandes cadenas. Sin embargo, para acceder a este beneficio, las ferreterías deben cumplir con ciertos requisitos, como no superar los ingresos brutos anuales a los establecidos por la ley. Además, las empresas deben operar dentro de actividades permitidas por la normatividad y no estar involucradas en sectores excluidos, como los relacionados con la explotación minera o el negocio de la importación de productos de lujo. Otro aspecto clave del régimen simple es que ofrece tasas de tributación progresivas, lo que significa que a medida que aumentan los ingresos de la empresa, la tasa impositiva también lo hace, pero de manera gradual, lo que permite una mayor equidad fiscal. La implementación del régimen simple ha sido clave para incentivar la formalización de negocios en el sector ferretero, favoreciendo a aquellos que aún operan de manera informal y necesitan simplificar sus obligaciones tributarias.

Además de la simplificación de los impuestos, el régimen simple también ofrece beneficios adicionales como la posibilidad de acceder a descuentos en algunos

impuestos, lo que representa un alivio económico importante para las pequeñas ferreterías que operan en un mercado competitivo. La implementación de esta normativa ha sido fundamental para fomentar la formalización de muchas ferreterías que antes operaban en la informalidad, dándoles acceso a beneficios fiscales y facilitando su participación en el mercado nacional.

En los últimos años, el sector ha experimentado una transformación digital significativa. La adopción del comercio electrónico por parte de muchas ferreterías ha permitido a los consumidores realizar sus compras de manera más cómoda y eficiente, lo que ha aumentado la competitividad y la accesibilidad del mercado. El crecimiento de las plataformas de ventas digitales, junto con el uso de aplicaciones móviles para facilitar el acceso a productos y servicios, ha sido clave para mantener la relevancia del sector en un contexto cada vez más digitalizado.

Sin embargo, el sector ferretero enfrenta varios retos. Uno de los más destacados es el aumento en los costos de los materiales, que ha sido impulsado por la inflación y la escasez de algunos insumos. Este incremento en los costos afecta tanto a los proveedores como a los consumidores, creando una presión sobre los márgenes de ganancia de las empresas del sector. Además, las dificultades logísticas y de distribución, especialmente en áreas más alejadas de las grandes ciudades, aumentan los costos operativos y afectan la eficiencia del servicio.

Marco referencial

0.1 Marco Teórico

0.1.1 *Teoría General de Sistemas*

La Teoría General de Sistemas (TGS) fue propuesta por Ludwig von Bertalanffy en la década de 1940 como un enfoque interdisciplinario para comprender sistemas complejos en diversas disciplinas [2]. Según esta teoría, un sistema es un conjunto de elementos interrelacionados que trabajan en conjunto para alcanzar un objetivo común [2]. La TGS se ha aplicado ampliamente en diversas áreas, incluida la gestión empresarial, la biología, la informática y la ingeniería de software.

En el contexto de este trabajo, se considera la ferretería como un sistema, lo que implica que posee entradas, procesos y salidas que interactúan para lograr la eficiencia operativa. La aplicación de la TGS permite analizar la estructura y funcionamiento de la ferretería, facilitando el diseño de un sistema de información que optimice sus procesos internos y contribuya a su organización.

0.1.2 Teoría de la Usabilidad

La usabilidad es un concepto fundamental en la interacción humano-computadora y se refiere a la facilidad con la que los usuarios pueden aprender y utilizar un sistema para alcanzar sus objetivos de manera eficiente y satisfactoria [3]. Según Jakob Nielsen, la usabilidad se basa en cinco atributos principales:

- Aprendizaje rápido: La interfaz debe permitir que los nuevos usuarios se familiaricen con el sistema en poco tiempo.
- Eficiencia: Los usuarios experimentados deben poder completar sus tareas de manera rápida.
- Memorabilidad: Después de un período sin uso, los usuarios deberían poder recordar cómo interactuar con el sistema sin dificultad.
- Errores mínimos: Se debe reducir la posibilidad de errores y, en caso de que ocurran, permitir una fácil recuperación.
- Satisfacción: El sistema debe ser agradable de usar, proporcionando una experiencia positiva. En el contexto de este trabajo de grado, la usabilidad es un aspecto crítico, ya que los usuarios del software serán empleados de ferreterías, quienes pueden no tener experiencia previa en el uso de sistemas informáticos avanzados.

Aplicar principios de usabilidad en el desarrollo del software garantizará que los empleados de la ferretería puedan operar el sistema sin dificultades, optimizando los procesos de ventas y gestión de inventario. Esto se alinea con el enfoque de diseño

centrado en el usuario [3], que destaca la importancia de adaptar la interfaz a las necesidades y habilidades de los usuarios finales.

0.1.3 *Teoría de Bases de Datos*

La teoría de bases de datos es un campo fundamental en informática y gestión de la información. Proporciona un marco teórico y práctico para diseñar, almacenar, recuperar y gestionar datos de manera eficiente y estructurada. Los modelos de datos, como el Modelo Entidad-Relación (ER) y el Modelo Relacional, definen la estructura lógica de la base de datos, incluyendo cómo se almacenan, organizan y relacionan los datos[4].

La integridad de la información es un concepto central en la teoría de bases de datos, asegurando la precisión y consistencia de los mismos en el modelo mediante restricciones como claves primarias y foráneas, que garantizan la integridad referencial. Las transacciones, unidades de trabajo en una base de datos, son el enfoque de la teoría de transacciones, que se centra en garantizar la consistencia y la atomicidad de las operaciones en la base de datos. Por ejemplo, la atomicidad asegura que una transacción se considere una operación única e indivisible[5].

Para concluir, las teorías presentadas en este marco teórico están interrelacionadas y son fundamentales para el desarrollo de un sistema de información efectivo para las ferreterías bajo el régimen simple de tributación. La Teoría General de Sistemas establece una visión global de la ferretería como un sistema interconectado, donde todas las partes trabajan juntas para alcanzar un objetivo común, lo que proporciona una base para entender cómo interactúan los diversos componentes del sistema. La

Teoría de la Usabilidad complementa esta perspectiva al centrarse en la importancia de garantizar que los empleados de la ferretería, quienes serán los usuarios del sistema, puedan interactuar con él de manera efectiva, incluso si no tienen experiencia previa con software. Aquí, la usabilidad se convierte en un factor clave para el éxito del sistema, asegurando que el diseño sea accesible y fácil de usar. Finalmente, la Teoría de Bases de Datos proporciona las bases para la gestión de la información dentro del sistema, asegurando que los datos relacionados con los productos, ventas y otras operaciones de la ferretería se manejen de manera eficiente, precisa y coherente. La integración de estas tres teorías permitirá construir un sistema que no solo funcione bien a nivel técnico, sino que también sea práctico y accesible para los usuarios finales, mientras asegura la correcta gestión de la información. En conjunto, estas teorías forman un enfoque integral y coordinado, esencial para el desarrollo de un software eficaz y funcional en el contexto de una ferretería.

0.2 Estado del Arte

Es característico que en la actual era digital las diferentes industrias y establecimientos comerciales alrededor del mundo se esfuercen por mantenerse a la vanguardia en temas tecnológicos. Estos esfuerzos permiten mejorar los procesos y operaciones dentro de las empresas. En el marco del presente trabajo de investigación, se ha tomado como referente a las ferreterías de Tuluá que operan bajo el régimen simple de tributación. Estas son esenciales para la economía popular del municipio porque proporcionan materiales esenciales para el sector de la construcción, que contribuye notablemente a la expansión y mejoramiento de las edificaciones del entorno urbano y rural.

A pesar de su importancia, estas ferreterías se enfrentan a desafíos significativos y carecen de las herramientas tecnológicas necesarias para optimizar sus operaciones. Esto hace necesario un software que busque mejorar los procesos de inventario, y el sistema de compra y venta de los materiales, lo que podría generar numerosos beneficios para el establecimiento.

A continuación, se seleccionaron un total de 5 documentos que en conjunto constituyen el estado del arte de la presente investigación. Tres de los trabajos citados corresponden a trabajos de grado y dos se constituyen como artículos de revista. Estos documentos respaldan teóricamente la necesidad de softwares diseñados a medida de establecimientos comerciales con características similares a las ferreterías que operan bajo el régimen simple de tributación. Todos los documentos seleccionados fueron elaborados en América Latina durante los últimos 5 años, es decir, entre los años 2019-2024.

El primer trabajo seleccionado para este análisis se titula “Implementación de un sistema control de almacén en la ferretería Tomasini - Piura; 2020” [6], elaborado por Socola Lozada (2022) para optar al título de Ingeniería de Sistemas en la Universidad Católica Los Ángeles de Chimbote. Es importante mencionar que la línea de investigación bajo la cual se desarrolló esta investigación es la Ingeniería de Software. El objetivo principal estuvo orientado en la implementación de un sistema de control para el almacén de la ferretería TOMASINI ubicada en la ciudad de Piura, Perú, de forma que se lograra mejorar el proceso de ingreso, salida y registro de todo tipo de materiales disponibles en la ferretería.

La metodología utilizada para el estudio fue cuantitativa y descriptiva, con diseño no experimental de corte transversal. Este trabajo tiene algunas similitudes en cuanto a la línea de investigación y la metodología propuesta. Además, la ferretería TOMASINI, al momento del estudio, presentaba las mismas dificultades observadas en las ferreterías que operan bajo el régimen simple de tributación en el municipio de Tuluá. Estas incluyen un control impreciso de su inventario, falta de registro ordenado de entradas, salidas o gastos, lo que causa confusión al momento de realizar cuentas o administrar el inventario. Además, los registros eran manejados manualmente en hojas de cuaderno, lo que genera pérdidas de información.

La situación fue analizada mediante una encuesta con una población muestral de 11 personas. En la dimensión uno de la encuesta, el 64.00

El segundo trabajo de investigación, publicado en 2021, fue realizado por Héctor Andrés Hernández, Yeimi Lorena Cruz Gil, María Daniela Puentes Saavedra y Diego Ernesto Mendoza Patiño. Titulado “Diseño de un sistema de gestión de inventarios

para el almacén TÉCNITALLER S.A.S de la ciudad de Neiva-Huila, Colombia” [7], este trabajo tuvo como objetivo diseñar un sistema de gestión de inventarios para el almacén Técnitaller S.A.S en Neiva. Este almacén enfrentaba ineficiencias en la administración de sus inventarios, lo que generaba reprocesos, faltantes, obsolescencia, bajo nivel de servicio y daño en mercancías.

La metodología empleada fue un estudio no experimental-transversal y descriptivo, que comenzó con un diagnóstico del área, seguido de la elaboración de la propuesta de gestión de inventarios y su análisis económico. Como resultado, se determinó que el modelo a implementar integraba un sistema de información con 4 módulos de gestión: Compras, almacén, inventarios y despacho. La implementación de este sistema permitiría optimizar la gestión de inventarios y mejorar la operación en general, reflejándose en menores costos para mantener, pedir y almacenar el inventario.

La tercera investigación, elaborada en 2021 por Andrés Felipe Rojas Pinzón para obtener el título de Ingeniería de Sistemas en la Universidad de los Andes, se titula “Gestión administrativa y contable para las pequeñas y medianas empresas del sector de transporte de carga por carretera” [8]. Aunque el sector al cual se aplica este estudio no corresponde al mismo que se pretende en esta investigación de trabajo de grado, es un referente importante. El transporte de carga por carretera en Colombia es uno de los gremios más importantes de la economía colombiana.

Este trabajo destaca que muchas empresas de transporte por carretera no cuentan con un software que facilite la toma de decisiones según el comportamiento del negocio. Como resultado, los pequeños transportadores no tienen herramientas para llevar sus gastos de manera organizada ni sacar provecho de los datos que recolectan, lo que los

coloca en desventaja competitiva. Esto motivó el desarrollo de una solución orientada a la gestión de los gastos de la empresa transportadora y la presentación de los mismos según parámetros elegidos por el usuario. Los informes generados pueden ser exportados a Excel para que el dueño de la empresa tenga información valiosa para la toma de decisiones y mejorar la competitividad en el sector del transporte.

El cuarto documento en esta recopilación se titula “Conceptos económicos de innovación en la valoración tecnológica de MiPymes colombianas desarrolladoras de software”, elaborado por María Isabel Díaz Vega y Manuel José Ospina Ospina. Publicado en la revista virtual del Sena, Vol. 87, Núm. 2 (2023) [9], este artículo sostiene que algunas micro, pequeñas y medianas empresas del sector de software en Colombia han desaparecido debido a la falta o uso incorrecto de metodologías de gestión de proyectos, la falta de acceso a capitales de riesgo y la inadecuada valoración de productos.

Mediante una revisión de literatura financiera, económica y de estimación del software por encargo, la investigación busca identificar conceptos de la economía de la innovación que inciden en la valoración tecnológica de productos de las MiPymes dedicadas al desarrollo de software por encargo en Colombia. Este estudio subraya la importancia de involucrar conceptos como bienes públicos y privados, costos de investigación y desarrollo, el valor generado, el precio del mercado, las externalidades, el cambio técnico y el crecimiento económico en los modelos de valoración tecnológica.

Finalmente, el quinto estudio titulado “Desarrolladores de software en tiempos de pandemia” [10] fue publicado en 2024 por Jhonatan Sánchez Pimiento y Luz Arias Casallas. Este trabajo hace referencia a las dificultades y oportunidades que surgieron

durante la emergencia sanitaria del COVID-19. En Colombia, las pequeñas empresas (PYMES) fueron las más afectadas por la pandemia debido a que no contaban con plataformas tecnológicas que les permitieran integrarse a la digitalización.

Por lo tanto, las empresas del sector tecnológico tuvieron que suplir las necesidades evidenciadas en diferentes sectores productivos. Este fenómeno resultó en un crecimiento significativo en las áreas de tecnología (software y hardware). Sin embargo, también se evidenció un déficit de desarrolladores debido a la gran cantidad de proyectos surgidos, lo que subraya la necesidad de analizar los lenguajes de programación en relación con las oportunidades laborales y de mejora en las empresas de todos los sectores productivos.

0.3 Antecedentes

En el contexto de este trabajo de grado, fue fundamental incluir una sección de antecedentes con el objetivo de establecer una base clara y contar con referencias sólidas sobre las tendencias actuales del mercado ferretero. Esta sección busca analizar sistemas existentes que podrían considerarse competencia para el software que se pretende desarrollar, identificando tanto sus fortalezas como sus debilidades. Este análisis permitió determinar oportunidades de mejora que otras soluciones no ofrecen a sus clientes y que podrían servir de inspiración o diferenciación para la propuesta.

Para garantizar un enfoque riguroso y relevante, esta sección documentó las ventajas, desventajas, y vacíos de las soluciones existentes, con base en fuentes confiables y soportadas bibliográficamente. Se incluirán sistemas ERP que, aunque no estén específicamente diseñados para empresas del sector de ferreterías, tienen características similares a la solución planteada, como la gestión empresarial integral y la adaptabilidad. Esto invita a realizar búsquedas más específicas utilizando palabras clave que permitan recopilar información precisa sobre el sector.

0.3.1 *Manager*

Manager es un software ampliamente utilizado en diversos sectores productivos. Se clasifica como un ERP (Enterprise Resource Planning) que se destaca por su capacidad de adaptarse a casi cualquier tipo de negocio. Este sistema ofrece una variedad de módulos que pueden añadirse o eliminarse según las necesidades específicas de la empresa, lo que lo convierte en una herramienta flexible y escalable.

Entre sus versiones especializadas se incluyen módulos para contabilidad general, producción, supermercados, restaurantes, ferreterías, acueductos, nómina, recursos humanos, entre otros[11].

Ventajas:

- Amplia adaptabilidad a diferentes tipos de negocios.
- Modularidad, permitiendo personalizar las funcionalidades según las necesidades.
- Diversificación de aplicaciones, lo que le permite abarcar un mercado amplio.

Desventajas:

- Aunque sus tarifas oscilan entre 336.000 y 841.000 anuales por dispositivo, lo que puede parecer accesible, el costo puede incrementarse significativamente si se necesita implementar en múltiples dispositivos.
- Puede ser genérico para sectores específicos, dejando vacíos en funcionalidades clave para ciertas industrias, como la ferretera.

Este sistema es una referencia valiosa debido a su enfoque en la flexibilidad y modularidad, características que podrían ser aprovechadas en el diseño de la solución propuesta.

0.3.2 *Siigo*

Siigo es una empresa colombiana reconocida por ofrecer soluciones ERP dirigidas a pequeñas y medianas empresas (PYMEs). Su enfoque se centra en proporcionar herramientas integrales para la gestión empresarial, incluyendo áreas como contabilidad, facturación, inventario, nómina, entre otras [12].

Ventajas:

- Adaptabilidad a diversos sectores, permitiendo una personalización según las necesidades específicas del cliente.
- Uso de tecnologías en la nube, facilitando el acceso remoto y la colaboración entre usuarios.
- Actualizaciones regulares que aseguran el cumplimiento de las normativas y las necesidades del mercado.

Desventajas:

- Aunque es flexible, su enfoque generalizado podría dejar vacíos en funcionalidades específicas para sectores particulares, como el de ferreterías.
- Dependencia de la conectividad a internet, lo que podría representar una limitante para algunas empresas en ubicaciones con acceso limitado.

El análisis de Siigo es relevante para identificar cómo la integración de tecnologías en

la nube y un enfoque centrado en las necesidades de las PYMEs pueden servir como insumos para la solución propuesta, pero también destaca la importancia de ajustar estas herramientas a las necesidades específicas del sector de ferreterías.

0.4 Marco Conceptual

0.4.1 *Sistema de Información*

Un sistema de información se refiere a una herramienta tecnológica que facilita las tareas de un sector productivo, como puede ser el caso de las ferreterías de régimen simple de tributación en este trabajo de grado. Utiliza la información del usuario y la transforma de manera que este pueda tomar decisiones basadas en los datos proporcionados por el software de manera intuitiva [13].

0.4.2 *Sistema de Información web*

El sistema de información web se refiere a un sistema de información que se ejecuta en un servidor remoto y se tiene una conexión al mismo, este posee la capacidad de procesamiento, el almacenamiento y la memoria en un lugar distinto al cliente que proporciona las entradas.[31]

0.4.3 *Régimen simple de tributación*

El Régimen Simple de Tributación - RST es un sistema de impuestos que favorece al establecimiento generando un bajo valor al pagar los impuestos de manera unificada, brindando tarifas bajas las cuales se liquidan según el nivel de ingresos y al que pueden acceder de forma voluntaria, que se declara anualmente y se paga con anticipos bimestrales a través de los sistemas electrónicos de la DIAN (Dirección de Impuestos y Aduanas Nacionales. 2024) [14].

0.4.4 *Software para Ferreterías*

Cuando se desarrolla software para ferreterías, se espera que incluya funcionalidades clave como seguimiento de inventario, ventas, compras y gestión de proveedores. Estas funcionalidades impactan tanto a los empleados como a los clientes, mejorando la atención y servicio. Para ferreterías de régimen de régimen simple de tributación, se busca personalizar estas funcionalidades para adaptarse a sus necesidades específicas, evitando aspectos innecesarios como la generación de facturas para entidades de impuestos o pagos de prestaciones sociales.

0.4.5 *Sistemas de Gestión Empresarial (ERP)*

Un sistema de gestión empresarial (ERP, por sus siglas en inglés) es un software diseñado para mejorar la productividad de un negocio al tener toda la información centralizada. Los ERP tienen características fundamentales como flexibilidad, escalabilidad, adaptación y expansión de uso a través de medios digitales (Zabala et al., 2021) [15].

0.4.6 *Inventario*

El inventario consiste en las herramientas y productos disponibles en un almacén o depósito para ser utilizados según las necesidades. Puede incluir materia prima, equipos, herramientas y materiales directos. En el contexto de este trabajo de grado, usaremos la definición de que un inventario es un software que en el que se reflejan las existencias del inventario físico y así tener una noción clara de lo que se posee y lo que falta [16].

0.4.7 Módulos del Software

Los módulos del software son funcionalidades específicas en forma de código que se agregan o eliminan según sea necesario. Estas son unidades individuales y autónomas dentro de un programa o sistema más grande, permitiendo escalabilidad y un mayor control de los fallos en el software

0.4.8 Requerimientos del Software

Los requerimientos del software son indicaciones especificadas por el cliente para que el producto entregado sea acorde a sus necesidades. Estos son repasados, evaluados y aprobados sistemáticamente, buscando que se reflejen en el software en forma de pruebas para evitar ambigüedades (Huarcaya Zapana et al., 2021) [17].

0.4.9 Interfaz de Usuario (UI)

La interfaz de usuario (UI) es la parte visual de un programa informático. Incluye elementos con los que el usuario puede interactuar digitalmente, como botones, listas, imágenes, formularios y textos. También abarca características visuales como colores, fuentes y tamaños de letra, entre otros.

0.4.10 Experiencia del Usuario (UX)

La experiencia de usuario (UX) se refiere al nivel de satisfacción del usuario mientras utiliza un software. Considera factores como el tamaño de letra, la posición de los botones y la fuente del texto, ya que estos aspectos afectan la facilidad de uso del software.

0.4.11 Integración de Sistemas

La integración de sistemas se refiere a cómo se comunican los diferentes componentes del sistema. Por ejemplo, la conexión entre los módulos de facturación y el módulo de nómina de empleados. Esto se utiliza para evaluar la optimización del sistema (Pérez et al., 2022) [18].

0.4.12 Escalabilidad

La escalabilidad es una característica del software que mide su capacidad para crecer a medida que el negocio lo hace. Un software es escalable si permite agregar o modificar funcionalidades de manera sencilla e intuitiva, sin que el tiempo de desarrollo sea un obstáculo para el crecimiento de la empresa.

0.5 Estándar IEEE 12207

Para la gestión del ciclo de vida del software en este trabajo de grado, se adoptó el **estándar IEEE 12207**, el cual define un conjunto de procesos, actividades y tareas que permiten estructurar el desarrollo, mantenimiento y operación de un sistema de software [30]. Este estándar proporciona un marco para garantizar la calidad en todas las fases del proyecto, desde la planificación hasta la implementación y pruebas.

1 Alcance del proyecto

1.1 Declaración de Alcance

El propósito principal de este proyecto fue diseñar, desarrollar e implementar un sistema de software de gestión empresarial completo y especializado, dirigido a las ferreterías que operan bajo el régimen simple de tributación en la ciudad de Tuluá, Valle del Cauca. A través de este desarrollo, se buscó mejorar la eficiencia operativa, apoyar la toma de decisiones estratégicas, garantizar el cumplimiento normativo y aumentar la calidad del servicio ofrecido a los clientes. De este modo, las ferreterías pueden posicionarse como negocios más competitivos y preparados para enfrentar los desafíos del mercado.

El sistema lleva a cabo diversas tareas específicas orientadas a satisfacer las necesidades operativas y estratégicas de las ferreterías. Entre estas tareas se encuentran la gestión del inventario, que permitirá controlar el stock de productos, actualizar automáticamente las existencias y generar alertas de bajo inventario; la gestión de ventas, que incluirá el registro de transacciones, la emisión de facturas, la recolección de las mismas; y la gestión de clientes, mediante la cual se almacena información de los clientes, se rastreará su historial de compras y se facilitará una atención personalizada. Además, el sistema automatiza procesos relacionados con el cumplimiento tributario, como los cálculos necesarios para el régimen simple de tributación y la generación de informes fiscales. También incluye herramientas de análisis y reportes que proporcionarán información clave sobre desempeño, ventas y finanzas para respaldar la toma de decisiones.

Para garantizar un funcionamiento efectivo, el sistema está compuesto por módulos especializados que abarcarán diferentes áreas de gestión empresarial. Estos módulos incluyen un módulo de inventarios para el control de productos y existencias, un módulo de ventas para gestionar transacciones y facturación, un módulo de clientes que permitirá organizar la información y el historial de los usuarios, un módulo contable para generar reportes financieros y cumplir con las obligaciones tributarias, y un panel de administración que ofrece una interfaz centralizada para visualizar datos clave y personalizar configuraciones.

En cuanto a sus límites, el sistema es una aplicación web compatible con navegadores modernos y adaptada para dispositivos móviles, pero no se desarrolla aplicaciones nativas para sistemas operativos como iOS o Android. Tampoco integra funcionalidades avanzadas de comercio electrónico ni conexión con plataformas externas como marketplaces. Los datos se almacenan en una base de datos relacional que asegurará la integridad y disponibilidad de la información.

Algunas funciones están fuera del alcance del proyecto debido a las limitaciones de tiempo y presupuesto. Entre ellas, se encuentran el desarrollo de aplicaciones móviles nativas, la gestión avanzada de nómina y la capacitación intensiva de los usuarios finales. Estas exclusiones están justificadas porque el enfoque está en priorizar las funcionalidades más relevantes para el objetivo del sistema, optimizando los recursos disponibles.

Para asegurar que el sistema se alinee con las necesidades específicas del sector, se recopiló información mediante entrevistas y encuestas a propietarios y empleados de ferreterías, revisiones de normativas legales aplicables al régimen simple de tributación

y análisis de sistemas similares en el mercado. Además, la especificación del software fue validada a través de prototipos interactivos y pruebas piloto en empresas seleccionadas, asegurando que la solución cumpla con las expectativas y ofrezca beneficios tangibles.

1.2 Objetivos

1.2.1 *Objetivo General*

Implementar un sistema de información de gestión empresarial enfocado en las ferreterías de régimen simple de tributación de la ciudad de Tuluá, Valle del Cauca.

1.2.2 *Objetivos Específicos*

- Levantar requisitos funcionales y no funcionales analizando las necesidades de una ferretería, identificando oportunidades de mejora que sirvan como base para la implementación del sistema de gestión empresarial.
- Diseñar una solución que se adapte a las necesidades específicas de una ferretería de régimen simple de tributación, asegurando la satisfacción de dichas necesidades.
- Desarrollar una solución que cumpla los estándares de calidad de la IEEE que permita a una ferretería comenzar sus operaciones de manera eficiente.
- Realizar pruebas exhaustivas para identificar posibles errores en la solución, al mismo tiempo que se busca oportunidades de mejora continua.

1.2.3 *Resultados esperados*

En la sección de objetivos se plantea unos ítems a los que se va a llegar para considerar el proyecto como exitoso. Cada uno de esos objetivos se planteó con un resultado esperado que va a servir para dar valor al proyecto. En la siguiente tabla se presentan dichos resultados esperados de cada objetivo.

Objetivo	Resultado Esperado
Levantar requisitos funcionales y no funcionales analizando las necesidades de una ferretería, identificando oportunidades de mejora que sirvan como base para la implementación del sistema de gestión empresarial.	Levantar unos requerimientos funcionales y no funcionales para su proximo desarrollo.
Diseñar una solución que se adapte a las necesidades específicas de una ferretería de régimen simple de tributación, asegurando la satisfacción de dichas necesidades	Hacer que el software que se va a desarrollar próximamente sea cómodo para el usuario, realizando una en de manera que se pueda aumentar la productividad del usuario mientras usa la solución.
Desarrollar una solución que cumpla los estandares de calidad de la IEEE que permita a una ferretería comenzar sus operaciones de manera eficiente.	Lograr que el software desarrollado sea funcional y de valor a la ferretería.
Realizar pruebas exhaustivas para identificar posibles errores en la solución, al mismo tiempo que se busca oportunidades de mejora continua.	Perfeccionar el software desarrollado para evitar errores a la hora de que el usuario utilice la aplicación identificando asperctos de mejora e implementarlos.

Tabla 4.1: Resultados esperados de cada objetivo. Fuente: Elaboración propia.

1.3 Metodología de Desarrollo de Software

En el apartado de las metodologías de desarrollo de software, se procedió a comparar tres metodologías diferentes para tener opciones y, de esta manera, elegir la más apropiada para este proyecto.

La importancia de elegir una metodología de desarrollo de software correcta radica en que esta facilitará grandemente el trabajo. Además de proporcionar un marco de trabajo claro y conciso, la construcción de software se convierte en seguir una serie de instrucciones en cuanto al tiempo y forma de realizar las tareas. Esto no implica que se deba seguir la metodología de manera rígida, todo lo contrario. Se debe procurar que la metodología se adapte al programador o equipo de desarrollo de manera óptima para lograr la mejor experiencia posible.

Para la comparación de las herramientas, se utilizó los criterios de selección de metodologías de Rosales, P., Salas, J., y Tinoco, O. (2010) [19]. Estos criterios consisten en tener en cuenta diferentes aspectos, como los que se muestra en la imagen:

Criterio	Descripción	Estrategia
Presencia en internet	Mayor presencia en motores de búsqueda en internet	Usar motores como: Google Scholar, y Google Trends.
Documentación	Volumen de libros o papers en inglés o español referentes a la metodología	Usar catálogos de revistas indexadas tales como: scopus, pubindex; también búsquedas en sistemas digitales de bibliotecas. Scholar
Certificación	Aquella metodología en la que se puede certificar y por ende garantizar el cumplimiento de sus prácticas.	Búsqueda de empresas certificadoras de la implementación de la metodología con motores tales como: Google, Altavista y Yahoo.
Training	Aquella metodología para la que existe instituciones que ofrecen programas de entrenamiento referentes a la metodología.	Búsqueda de empresas con motores tales como: Google, Altavista y Yahoo.
Comunidades	Hay formada una comunidad de profesionales alrededor de la metodología o está asociada a Agile Alliance	Verificar si la metodología está asociada a Agile Alliance o tiene su propia comunidad.

Fuente: Adaptado a partir de (Rosales, Salas y Tinoco, 2010)

Figura 4.1: Tabla adaptada de selección de metodología de desarrollo de software

A partir de los criterios presentados anteriormente, se eligieron tres alternativas distintas de metodologías de desarrollo de software, las cuales son: Test Driven Development (TDD), Extreme Programming y Waterfall (cascada). se procederá al análisis correspondiente y como resultado de ello poseemos la siguiente tabla:

Metodología	Presencia	Doc	Certificación	Training	Comunidad	Total
Test Driven Development	2	1	2	2	3	10
Extreme Programming	3	3	2	3	3	14
Waterfall (cascada)	1	2	2	1	1	7

Tabla 4.2: Evaluación de Metodologías de Desarrollo de Software. Fuente: Elaboración Propia.

Analizando los resultados de la tabla diligenciada bajo los criterios presentados anteriormente se llega a la conclusión de que la metodología a elección será extreme programming ya que esta es la metodología que posee mas puntos al final. Extreme programming es una metodología de desarrollo de software propuesta por Kent Beck a fines de la década de 1990 basada en el desarrollo ágil que busca la mejora continua del software siempre teniendo en cuenta la calidad del producto [20].

Al ser extreme programming una metodología de desarrollo ágil, alinea sus principios con los valores fundamentales de la agilidad, como la flexibilidad, la colaboración y la entrega incremental. La capacidad de adaptarse rápidamente a los cambios en los requisitos del cliente es uno de los aspectos distintivos de XP [21]. A través de la iteración continua y la participación del cliente, la metodología se centra en la entrega de valor de manera constante y sostenible a lo largo del tiempo.

Otro principio clave es la realización de pruebas continuas, que implica la creación y ejecución constante de pruebas automatizadas. Este enfoque garantiza que el código sea robusto y funcional en todo momento, permitiendo la detección temprana de posibles problemas y facilitando la corrección rápida [21].

La integración continua complementa esta práctica al asegurar que los cambios en el código se fusionen de manera regular y se sometan a pruebas automáticas para verificar la estabilidad del sistema. Esta automatización contribuye a la eficiencia del proceso de desarrollo y reduce el riesgo de conflictos en la integración de código.

1.4 Metodología de Gestión de Actividades

Para optimizar el tiempo disponible y asegurar una gestión eficiente de las tareas durante el desarrollo de este trabajo de grado, se ha considerado necesario implementar una metodología de gestión de actividades. Esta metodología debe permitir tener una visión clara de las tareas en curso, las que faltan por realizar y las que ya se han completado.

A continuación, se presentan algunas alternativas que se evaluaron para la gestión de actividades:

- **Método de la lista de tareas:** Consiste en crear una lista de actividades que se deben realizar, con una simple marcación de tareas completadas y tareas pendientes.
- **Método de la matriz Eisenhower:** Utiliza una matriz de 2x2 para clasificar las tareas según su urgencia e importancia, lo que ayuda a priorizar las actividades que deben realizarse de inmediato y las que pueden delegarse o posponerse [32].
- **Método Kanban:** Se enfoca en visualizar el flujo de trabajo mediante un tablero con columnas que representan las diferentes fases del proyecto (Por hacer, En progreso, Hecho), lo que permite un seguimiento en tiempo real del avance de las tareas. [33]

Para seleccionar la metodología más adecuada, se tuvo en cuenta la necesidad de tener un sistema visual y flexible que permita un seguimiento claro y constante de

las actividades. El modelo Kanban fue elegido como la metodología base debido a su capacidad para proporcionar una visión clara y dinámica del estado de las tareas, permitiendo gestionar de forma eficiente las actividades en curso, las pendientes y las completadas. Además, se valora la flexibilidad de esta metodología, que permite ajustes y mejoras a medida que el proyecto avanza.

Cabe destacar que, a lo largo del desarrollo del trabajo, la metodología elegida no será rígida, sino que se someterá a una retroalimentación constante para optimizar la organización y asegurar el cumplimiento eficiente de los objetivos.

Desarrollo de proyecto

Para garantizar un desarrollo estructurado y eficiente, el proyecto se divide en cuatro fases principales: planificación, diseño, codificación y pruebas. Cada una de estas etapas desempeña un papel crucial en la construcción del sistema, desde la conceptualización inicial hasta la entrega de un producto funcional y confiable. A continuación, se procede a describir brevemente lo que se encontrará en cada uno de los apartados.

Fase de Planificación La fase de planificación comprende todo el proceso previo al desarrollo del proyecto. En esta etapa, se define el alcance del sistema, se identifican los requerimientos funcionales y no funcionales, se realiza un análisis del entorno y de las necesidades del usuario, y se establecen los recursos necesarios para la implementación.

Fase de Diseño Durante esta etapa, se definieron las arquitecturas del sistema y la interfaz de usuario. Se construyeron diagramas de flujo, casos de uso, y modelos de datos que representan las funcionalidades principales del sistema. El diseño tiene como objetivo asegurar que la solución sea intuitiva, escalable y alineada con los estándares de calidad del software. En esta fase, también se seleccionaron las tecnologías que serán utilizadas, como frameworks de desarrollo y bases de datos.

Fase de Codificación En la fase de codificación se materializa el diseño planteado. Aquí, se implementaron las funcionalidades del sistema siguiendo buenas prácticas de programación, con énfasis en la modularidad, el uso de patrones de diseño, y el cumplimiento de los requerimientos definidos. Se priorizó la creación de un sistema eficiente, seguro y de fácil mantenimiento.

Fase de Pruebas Finalmente, la fase de pruebas garantizó la calidad del sistema desarrollado. Se ejecutan pruebas funcionales, de integración, de rendimiento y de seguridad, con el fin de identificar y corregir posibles errores antes de la implementación final. Esta etapa asegura que el sistema cumpla con los estándares establecidos y satisfaga las expectativas del usuario.

1 Planificación

La fase de planificación es fundamental en el desarrollo de cualquier proyecto, ya que establece las bases sobre las cuales se edificará el sistema. En esta etapa, se llevó a cabo un análisis detallado de los requerimientos del sistema, tanto funcionales como no funcionales, y se definen los alcances de cada uno de los módulos de manera mas detallada. Es el momento de identificar las necesidades del usuario.

Esta sección se centró en detallar cada uno de los pasos previos al desarrollo del sistema, proporcionando un enfoque claro y organizado para la implementación exitosa del sistema de información para ferreterías bajo el RST. Se abordaron los aspectos clave de la recopilación de datos, el análisis de las necesidades del usuario, y el diseño de una hoja de ruta que guiará el proyecto durante su ciclo de vida.

1.1 Principios de Gestión de Calidad Aplicados al Producto

Para garantizar un software de alta calidad, se aplicaron los siguientes principios desde la fase de planificación y análisis de requisitos:

- **Confiabilidad:** Se refiere a la capacidad del sistema para operar sin fallos en un período determinado. Se mide a través de pruebas de estabilidad y tolerancia a fallos, asegurando que el software pueda manejar cargas de trabajo sin interrupciones inesperadas.
- **Eficiencia:** Evalúa el rendimiento del sistema en términos de tiempos de respuesta y uso óptimo de los recursos. Un software eficiente debe minimizar el consumo de

memoria y procesamiento sin comprometer la funcionalidad.

- **Seguridad:** Abarca la protección de la información y la prevención de accesos no autorizados. Se implementan medidas como autenticación robusta, cifrado de datos y control de accesos para mitigar riesgos de ataques informáticos.
- **Usabilidad:** Garantiza que la interfaz y la experiencia de usuario sean intuitivas y accesibles. Se considera la facilidad de aprendizaje, la reducción de errores y la eficiencia en la interacción con el sistema.
- **Mantenibilidad:** Se enfoca en la facilidad con la que el software puede ser actualizado, corregido y mejorado. Para ello, se aplican buenas prácticas de codificación, documentación clara y una arquitectura modular que facilite modificaciones futuras.

1.2 Plan de Comunicaciones con el Cliente

Con el fin de obtener conocimiento de operaciones de las ferreterías, se decidió trabajar para un cliente específico y de esta manera generalizar para que este sistema pueda operar en cualquier establecimiento de ferretería bajo régimen simple de tributación.

El cliente en el cual se basó el estudio fue Matering SAS, una empresa fundada en Palmira, valle del cauca en 2021 la cual efectivamente está bajo el RST, lo que permitió complementar el enfoque del proyecto y garantizar la relevancia de las soluciones implementadas, ya que el sistema de información diseñado específicamente para empresas bajo el Régimen Simple de Tributación (RST) debía cumplir con las normativas y beneficios asociados a este régimen.

Por lo tanto, para garantizar la satisfacción de Matering SAS y asegurar que FerreCloud cumple con los requisitos establecidos, se implementó un plan de comunicación estructurado en todas las fases del desarrollo. Dado que se adoptó una metodología ágil con entregas funcionales constantes, la retroalimentación del cliente fue un factor clave en la evolución del sistema. A continuación, se describe el proceso de comunicación en cada fase:

1.2.1 Fase de Levantamiento de Requisitos

- Se realizaron reuniones iniciales con los representantes de Matering SAS para comprender sus necesidades y definir los requerimientos del sistema.
- Se estableció un documento de requisitos funcionales y no funcionales, validado por el cliente antes de iniciar el desarrollo.
- Se habilitó un canal de comunicación permanente (correo electrónico y WhatsApp) para atender dudas y recibir observaciones sobre los requerimientos.

1.2.2 Fase de Desarrollo e Implementación

- Se realizaron entregas parciales y funcionales del sistema en cada sprint, permitiendo que Matering SAS probara las funcionalidades en un entorno controlado.
- Se llevaron a cabo reuniones de revisión con el cliente al finalizar cada sprint para presentar avances, resolver inquietudes y ajustar funcionalidades según la retroalimentación recibida.

- Se utilizaron checklist para validar las entregas de los requerimientos (La checklist se evidencia en la sección de anexos).

1.2.3 Fase de Pruebas y Validación

- Se implementó FerreCloud en un entorno real dentro de Matering SAS, permitiendo que los empleados probaran el sistema en sus operaciones diarias.
- Se aplicaron encuestas y se recopilaban observaciones sobre la usabilidad y el rendimiento del sistema.
- Se documentaron problemas detectados y se estableció un plan de corrección con seguimiento conjunto con el cliente.

Este plan de comunicación permitió mantener una colaboración continua con Matering SAS, asegurando que FerreCloud cumpliera con sus expectativas y requerimientos.

1.3 Criterios de Aceptación del Cliente

Para formalizar la aceptación de cada entrega, se establecieron los siguientes mecanismos de validación con Matering SAS:

- **Pruebas funcionales:** Cada módulo entregado fue probado por el administrador de Matering SAS para verificar que cumpliera con los requisitos definidos en la fase de levantamiento de requerimientos.

- **Checklist de validación:** Se estableció una lista de criterios basada en los requerimientos funcionales y no funcionales para cada módulo, la cual debía ser aprobada por el cliente antes de su implementación definitiva.
- **Uso en entorno controlado:** La validación final del sistema se realizó en un entorno controlado dentro de Matering SAS, asegurando que la solución fuera efectiva en sus operaciones diarias antes de su aceptación final.

Estos mecanismos garantizaron que el cliente estuviera involucrado en todo el proceso de desarrollo y que FerreCloud cumpliera con sus expectativas y necesidades operativas.

1.4 Fases del Proyecto y Aplicación del Estándar IEEE 12207

A continuación se procede a describir el proceso en el que se garantizó el estándar IEEE 12207 dentro de cada una de las fases del proyecto, asegurando la calidad en cada una de ellas.

1.4.1 Fase de Planificación y Análisis de Requisitos

- Se establecieron los requisitos funcionales y no funcionales del sistema.
- Se documentó la planificación del desarrollo siguiendo prácticas de *gestión de proyectos de software*.
- Se aplicaron principios de *gestión de la calidad* desde el inicio, definiendo criterios de validación y aceptación.

1.4.2 *Fase de Diseño*

- Se estructuró la arquitectura del sistema con base en principios de modularidad y mantenibilidad.
- Se utilizó un control de versiones y gestión de configuración para documentar y asegurar la trazabilidad del diseño.

1.4.3 *Fase de Codificación*

- Se implementó un control de calidad en la escritura del código, aplicando *revisiones de código y pruebas unitarias*.
- Se aplicaron herramientas de gestión de cambios para garantizar la estabilidad del software.

1.4.4 *Fase de Pruebas y Validación*

- Se implementaron pruebas funcionales y de integración para verificar el cumplimiento de los requisitos.
- Se realizó una validación del sistema mediante pruebas de usuario y pruebas de carga.

Este enfoque garantizó que el sistema desarrollado cumpla con los principios de calidad establecidos, asegurando su funcionalidad, mantenibilidad y eficiencia en el contexto de la gestión de ferreterías bajo el régimen simple de tributación.

1.5 Procesos generales de ferreterías

En el marco de este trabajo de grado, el primer paso para una planificación eficiente fue conocer los procesos generales de una ferretería para después particularizar con base a las necesidades del RST, por lo tanto, con base a la entrevista realizada al administrador de **Matering SAS**, entre los procesos generales de una ferretería se incluyen la gestión de inventarios, lo cual es un apartado clave, ya que permite registrar las entradas de productos provenientes de los proveedores y las salidas correspondientes a las ventas o usos internos. Este proceso incluye la verificación constante de las existencias para evitar desabastecimientos y el exceso de productos, así como la clasificación y organización de los artículos en categorías específicas (herramientas, materiales de construcción, electricidad, etc.) para facilitar su localización.

En cuanto a las ventas, estas se realizan principalmente de manera directa, con interacción en el mostrador, donde se factura y se entregan los productos al cliente.

El proceso de compras y reabastecimiento implica la gestión de relaciones con proveedores, la elaboración de órdenes de compra basadas en las necesidades del negocio y la recepción de mercancía, donde se inspeccionan y registran los productos entregados. Este proceso asegura que el inventario esté alineado con la demanda del mercado.

La atención al cliente es otro pilar fundamental. Incluye la resolución de devoluciones de productos defectuosos o insatisfactorios, el registro de clientes para posterior revisión.

Finalmente, la administración general abarca el control del personal, lo cual incluye las personas vinculadas con el establecimiento además del registro contable.

1.6 Particularización al RST

Según el administrador de Matering SAS, ferreterías acogidas al Régimen Simple de Tributación presentan características específicas que impactan directamente en sus procesos. En el área de ventas, la integración del IVA simplificado es un aspecto clave, aunque este no se pague por medio del RST, es necesario excluirla del cálculo fiscal de manera automatizada para efectos prácticos.

La gestión financiera y fiscal se caracteriza por la simplicidad, gracias a la estructura del régimen que combina varios impuestos en un único pago bimestral. Esto reduce significativamente la carga administrativa, permitiendo generar reportes básicos que cumplen con las normativas legales sin la necesidad de sistemas complejos, a continuación, se procede a describir el cálculo del RST según la DIAN:

Rangos		Grupos / tarifas				Rangos		Grupo / tarifas		Rangos		Grupo / tarifas	
Ingresos brutos		G. 1. Tiendas pequeñas, mini-mercados, micro-mercados y peluquería:	G. 2. Actividades comerciales al por mayor y detal; servicios técnicos y mecánicos en los que predomina el factor material sobre el intelectual, los electricistas, los albañiles, los servicios de construcción y los talleres mecánicos de vehículos y electrodomésticos; actividades industriales, incluidas las de agro-industria, mini-industria y micro-industria; actividades de telecomunicaciones y las demás actividades no incluidas en los siguientes numerales	G.3. Actividades de expendio de comidas y bebidas, y actividades de transporte	G. 4. Educación y actividades de atención de la salud humana y de asistencia social	Ingresos brutos		G. 5. Servicios profesionales, de consultoría y científicos en los que predomine el factor intelectual sobre el material, incluidos los servicios de profesiones liberales	Ingresos brutos		G. 6. Reciclaje, recuperación de materiales y recolección de desechos		
Igual o superior a	Inferior a					Igual o superior a	Inferior a		Igual o superior a	Inferior a			
0	6.000					0	6.000		0	100.000			
6.000	15.000					6.000	12.000						
15.000	30.000												
30.000	100.000												
		1.2%	1.6%	3.1%	3.7%	0	6.000	7.3%					
		2.8%	2.0%	3.4%	5.0%			8.3%					
		4.4%	3.5%	4.0%	5.4%								
		5.6%	4.5%	4.5%	5.9%								
Liquidar el RST es SIMPLE: 1. Seleccione su grupo 2.Ingresos brutos X tarifa = Impuesto a pagar													

Figura 5.1: Tabla de cálculo de impuesto RST.

Para el RST existe la tabla de liquidación, la cual tiene una sola fórmula, la cual se muestra en la parte inferior de la Figura 5.1, esta incluye un grupo, que en este caso se tomará el G2 ya que las ferreterías son parte de este y así extrayendo la tarifa dependiendo la cantidad de Unidad de Valor Tributario (UVT)¹; Cabe resaltar que esta tarifa es variable, por lo tanto debe ser parametrizable para próximos cambios fiscales. Consiguiente a esto se calculan los ingresos brutos, lo cual se va a calcular de manera automática por medio del sistema. Teniendo en cuenta las dos variables se procede a calcular el impuesto a pagar bajo el RST.

En cuanto al cálculo de los ingresos brutos se tomó como referencia el simulador de RST que está disponible en la página web de la DIAN, lo cual tiene en cuenta los descuentos que establece la DIAN para tener un correcto cálculo de los ingresos brutos.

De esta manera el Administrador de la ferretería pudo tener un calculo fiscal automático, evitando la recolección manual de facturas, cálculo de ingresos brutos y descuentos relacionados al RST.

¹UVT: medida de valor que permite ajustar los valores contenidos en las disposiciones relativas a los impuestos y obligaciones administrados por la Unidad Administrativa Especial Dirección de Impuestos y Aduanas Nacionales

El desarrollo de un sistema de información eficiente y funcional requiere una estructura modular que permita la gestión y operación de sus diversos componentes de manera independiente y coordinada. En el trabajo de grado titulado "Sistema de información para ferreterías bajo régimen simple de tributación" se ha desarrollado un software el cual se llamó **FerreCloud**, el cual se ha diseñado utilizando un enfoque modular, asegurando así una mayor flexibilidad, escalabilidad y mantenibilidad.

1.7 Módulos

Cada módulo del sistema FerreCloud ha sido cuidadosamente desarrollado para cumplir con requisitos específicos y abordar distintas áreas funcionales esenciales para la operación diaria de una ferretería. La modularidad del sistema no solo facilita su desarrollo y mantenimiento, sino que también permite una mejor adaptación a las necesidades cambiantes del negocio.

Para este contexto se crearon 6 grandes áreas para abarcar completamente el funcionamiento de un establecimiento objetivo los cuales se describen a continuación

1. Gestión de inventario
2. Gestión de proveedores
3. Gestión de clientes
4. Gestión de impuestos
5. Gestión de ventas
6. Gestión de empleados

A continuación se procede a describir los objetivos de cada uno de los módulos del software FerreCloud:

1.7.1 *Gestión de inventario*

El módulo de gestión de inventario fue una parte esencial de todo establecimiento dedicado a la venta y suministro de productos físicos debido a que con base a este se pueden tomar decisiones que afecten en gran medida al establecimiento en cuestión; Algunos de estos apartados pueden ser en primer lugar, el control preciso de las existencias que permite llevar un registro detallado de los artículos disponibles, lo que ayuda a evitar faltantes o el exceso de productos. Esto asegura que siempre haya suficiente stock para satisfacer la demanda sin caer en la llegada de productos que existen en inventario, lo que podría generar pérdidas.

Además, un módulo de inventario facilitó la gestión eficiente de los productos al permitir su organización por categorías, proveedores, precios y otras características. Esto simplifica el manejo del inventario, haciéndolo más accesible y fácil de gestionar.

Implementar un módulo de inventario también ahorra tiempo y reduce costos al automatizar procesos que, de otra manera, serían manuales, como el conteo y la actualización de existencias. Esto minimiza errores y libera tiempo para que el personal se concentre en otras tareas importantes.

Finalmente, el análisis de ventas y de productos es un aspecto crucial. Un buen módulo de inventario ofrece reportes detallados sobre los artículos más vendidos, las ganancias obtenidas y la rotación de inventarios, lo que facilita la planificación y la

estrategia de ventas.

1.7.2 Gestión de proveedores

El módulo de Gestión de proveedores de FerreCloud se enfocó en optimizar la gestión de las relaciones comerciales con los distintos proveedores y mejorar el abastecimiento de productos. En primer lugar, un módulo de este tipo permite organizar y centralizar la información de los proveedores, lo que facilita el acceso a datos clave como contactos. Esto mejoró la eficiencia en las negociaciones y reduce el tiempo invertido en buscar o gestionar información.

Otro objetivo importante es mejorar la planificación y el control de las compras. Al integrar el módulo con el sistema de inventario, fué posible monitorizar los niveles de stock cuando ciertos productos alcancen un nivel mínimo. Esto asegura un flujo constante de abastecimiento y minimiza el riesgo de faltantes en el inventario.

Un módulo de proveedores también contribuye a mejorar la comunicación y el seguimiento de pedidos. apartados como las fechas de entrega y pagos permite un seguimiento más preciso y evita errores o retrasos que podrían afectar las operaciones de la ferretería.

Finalmente, uno de los objetivos clave es mantener una relación fluida y duradera con los proveedores. Al contar con un sistema que gestiona de manera eficiente las órdenes de compra el establecimiento que utilice FerreCloud puede fortalecer sus relaciones comerciales, lo que podría derivar en mejores descuentos o plazos de entrega más favorables.

1.7.3 *Gestión de clientes*

Los objetivos de implementar un módulo de clientes para una ferretería son esenciales para mejorar la gestión de relaciones con los mismos y optimizar las ventas. En primer lugar, uno de los principales objetivos es centralizar la información de los clientes, lo que permite registrar datos relevantes como nombres, correos electrónicos, direcciones, contactos e historial de compras. Esta base de datos facilita el acceso a información clave y ayuda a personalizar la atención al cliente.

Otro objetivo importante es mejorar la experiencia del cliente. Al contar con un historial de compras que permite al usuario del sistema realizar devoluciones de manera rápida y cómoda para el cliente

Finalmente, un módulo de clientes bien implementado no solo mejora la gestión interna de la ferretería, sino que también contribuye a aumentar la satisfacción del cliente, optimizar las ventas y fortalecer la relación comercial con los clientes, lo que a su vez impulsa el crecimiento del negocio.

1.7.4 *Gestión de impuestos*

El objetivo general del módulo de gestión de impuestos en FerreCloud es facilitar la adaptación de la contabilidad de la empresa al Régimen Simple de Tributación (RST). Este sistema se diseña para simplificar la carga fiscal de las ferreterías, promoviendo una gestión más eficiente y menos dispendiosa de las obligaciones tributarias.

Dentro de este módulo, se busca reducir significativamente el tiempo y el esfuerzo invertido en la búsqueda y recopilación de información fiscal consolidando toda esta información en una sola tabla centralizada. Esto no solo facilitará el acceso a los datos relevantes, sino que también permitirá realizar análisis más exhaustivos sobre la situación fiscal de la ferretería. La tabla incluirá información clave relacionada con el RST, como el total de ingresos, los costos deducibles, las retenciones aplicables y otros aspectos que impactan en la declaración de impuestos.

1.7.5 *Gestión de ventas*

módulo de ventas de FerreCloud es esencial para mejorar la gestión de los procesos comerciales y optimizar la atención al cliente. El objetivo principal del presente módulo es el de agilizar el proceso de ventas. Esto implica registrar las transacciones de manera rápida y precisa, reduciendo errores humanos y tiempos de espera en el mostrador, lo que mejora la experiencia del cliente y la eficiencia operativa.

Además, un módulo de ventas tiene como objetivo llevar un control detallado de las ventas realizadas. Esto incluye registrar información sobre los productos vendidos, cantidades y precios, así como los datos del cliente. Esta información centralizada permite un acceso rápido a los historiales de compra y facilita la generación de reportes de ventas en tiempo real.

Otro objetivo clave es integrar la gestión de inventarios con las ventas. De este modo, cada venta registrada impacta automáticamente en el inventario, reduciendo los niveles de stock de los productos vendidos y facilitando la actualización en tiempo real de las existencias. Esta integración ayuda a prevenir ventas de productos agotados, mejorando

la precisión del inventario y evitando inconvenientes con los clientes.

Además, un objetivo importante es optimizar el flujo de caja y la gestión de pagos. El módulo es esencial llevar un registro de los pagos recibidos y pendientes, gestionar diferentes métodos de pago y controlar las cuentas por cobrar de clientes frecuentes o mayoristas. Esto ayuda a mantener una liquidez adecuada y una mejor organización financiera.

Finalmente, un módulo de ventas busca mejorar la experiencia del cliente. Al contar con un sistema centralizado y automatizado, se pueden ofrecer promociones personalizadas, descuentos por fidelidad y recomendaciones de productos relacionados. Esto contribuye a incrementar la satisfacción del cliente y a fomentar su lealtad hacia la ferretería.

En conjunto, un módulo de ventas de Ferrecloud fortalece la relación con los clientes y proporciona datos valiosos para el crecimiento del negocio.

1.7.6 Gestión de empleados

Los objetivos de implementar un módulo de empleados para una FerreCloud es centralizar la información de los empleados. Esto implica registrar datos esenciales como nombres, cargos, cajas de compensación, remuneraciones, historial de desempeño y datos de contacto, lo cual facilita la administración del personal y el acceso a información relevante de manera rápida y organizada.

Finalmente, un objetivo clave es asegurar el cumplimiento de normativas laborales.

Un módulo de empleados bien diseñado facilita el control del cumplimiento de regulaciones, como el pago de prestaciones, vacaciones, horas extras y demás obligaciones legales. Esto no solo reduce riesgos legales, sino que también ayuda a mantener un ambiente laboral justo y motivador para los empleados.

1.8 Requerimientos

La técnica utilizada para la recolección de requerimientos en el desarrollo del Sistema de Información de Ferreterías bajo Régimen Simple de Tributación fue la organización de Focus Groups. Esta técnica se centró en reunir a un pequeño grupo de usuarios finales representativos (como administradores, vendedores y personal contable de la ferretería) para discutir sus necesidades, preocupaciones y expectativas respecto al sistema.

Descripción de la Técnica Formación del Grupo: Se seleccionaron de manera deliberada a participantes que fueran clave para el funcionamiento de la ferretería, tales como los encargados de ventas, el personal encargado de la contabilidad y administración, y los proveedores (Bodega). Este grupo debía ser representativo de las diversas funciones que el sistema debe cubrir.

Discusión Guiada: Un facilitador (que en este caso fue el responsable de la recolección de los requerimientos) guió la sesión de discusión utilizando una serie de preguntas abiertas que permitieron a los participantes expresar sus opiniones sobre el sistema que necesitaban. Se abordaron temas como:

- ¿Qué funcionalidades consideran esenciales para la gestión de ventas e inventarios?
- ¿Qué procesos actuales del negocio se desean mejorar?
- ¿Cuáles son los mayores desafíos que enfrentan en la contabilidad y generación de reportes fiscales?
- ¿Qué características de interfaz consideran más útiles para facilitar el trabajo

diario?

Análisis de Necesidades y Prioridades: Durante la discusión, los participantes también tuvieron la oportunidad de priorizar los requerimientos según su importancia. Se permitió que los usuarios comentaran y se intercalaran entre ellos, enriqueciendo la conversación y ofreciendo diferentes perspectivas de los mismos problemas.

Documentación y Validación: Tras la sesión de Focus Group, los resultados fueron documentados y se presentó un resumen de los requerimientos prioritarios. Esto fue validado por los mismos participantes y, si era necesario, se organizó una segunda sesión para afinar los detalles.

1.8.1 *Requerimientos funcionales*

Con base en lo mencionado anteriormente, se procederá con el planteamiento de requisitos funcionales, clasificando todos los requisitos con uno o más de ellos.

Tabla 5.1: REQUERIMIENTO 1

Nombre	Ver el inventario completo
Propósito	Los usuarios pueden ver las existencias de los productos para ofrecer a los clientes
Descripción	En la sección de inventario o en la propia pagina de ventas el usuario puede ver las existencias de los productos para de esta manera tener un claro conocimiento de que productos se pueden ofrecer a los clientes
Entrada	No aplica
Salida	Lista de los productos en existencia

Tabla 5.2: REQUERIMIENTO 2

Nombre	Ver el inventario completo
Propósito	Los usuarios pueden ver las existencias de los productos para ofrecer a los clientes
Descripción	En la sección de inventario o en la propia pagina de ventas el usuario puede ver las existencias de los productos para de esta manera tener un claro conocimiento de que productos se pueden ofrecer a los clientes
Entrada	No aplica
Salida	Lista de los productos en existencia

Tabla 5.3: REQUERIMIENTO 3

Nombre	Crear un nuevo producto en inventario
Propósito	El encargado crea un nuevo producto para ser visto en el inventario y de esta forma ofrecerle a los clientes
Descripción	En el módulo de inventario se instancia un botón que abre un formulario para crear un nuevo producto en inventario y de esta manera poder ingresar existencias del mismo
Entrada	Datos del producto
Salida	Producto creado en el inventario exitosamente

1.8.2 *Requerimientos no funcionales*

En el marco de este trabajo de grado, titulado "Sistema de información para ferreterías bajo régimen simple de tributación", se identificaron y detallaron los requisitos no funcionales que son esenciales para la construcción y operación exitosa del sistema. Estos requisitos no funcionales, aunque no especifican funcionalidades particulares, fueron cruciales para asegurar que el sistema sea robusto, eficiente y capaz de satisfacer las expectativas de los usuarios y las demandas operacionales de la ferretería. A continuación, se describen los principales requisitos no funcionales considerados para este proyecto:

Requerimiento	Descripción	Objetivo	Criterio de aceptación	Estrategia de implementación
Escalabilidad	El sistema debe manejar aumentos en la carga de trabajo sin degradación significativa del rendimiento.	Garantizar que el sistema crezca con la ferretería, soportando más usuarios, transacciones y datos.	Soporte para al menos el doble del tráfico inicial sin pérdida perceptible de rendimiento.	Uso de tecnologías escalables, como bases de datos distribuidas y servidores con capacidad de balanceo de carga.

Requerimiento	Descripción	Objetivo	Criterio de aceptación	Estrategia de implementación
Seguridad	Protección de datos para garantizar confidencialidad, integridad y disponibilidad.	Prevenir accesos no autorizados y proteger los datos de la ferretería y sus clientes.	Autenticación exitosa mediante métodos robustos (e.g., doble factor) y cifrado de información sensible.	Implementación de protocolos como HTTPS, almacenamiento cifrado y autenticación segura (e.g., JWT o OAuth).
Usabilidad	Interfaz amigable y fácil de usar, diseñada para diferentes niveles de habilidad técnica.	Facilitar la adopción del sistema por parte de los usuarios con mínima capacitación.	Usuarios completan las tareas principales en menos de 3 intentos y proporcionan retroalimentación positiva (>80 %) sobre la experiencia.	Diseño centrado en el usuario (UX/UI), pruebas con usuarios finales, y flujos de trabajo claros y consistentes.

Requerimiento	Descripción	Objetivo	Criterio de aceptación	Estrategia de implementación
Disponibilidad	Minimización del tiempo de inactividad del sistema.	Garantizar el acceso continuo al sistema, incluso en caso de fallos o desastres.	Tiempo de inactividad mensual menor al 0.5 %.	Implementación de redundancia en servidores y bases de datos, con un plan de recuperación ante desastres documentado.
Portabilidad	Capacidad de operar en múltiples entornos y dispositivos sin modificaciones extensivas.	Asegurar que el sistema pueda evolucionar con las necesidades cambiantes y adaptarse a diferentes infraestructuras tecnológicas.	Acceso exitoso desde computadoras, tabletas y teléfonos móviles sin problemas de compatibilidad.	Uso de tecnologías multiplataforma (e.g., frameworks web responsivos), y compatibilidad con entornos locales y en la nube.

Tabla 5.4: Requerimientos no funcionales

2 Diseño

En esta sección, se detalló el proceso de diseño del sistema de información especializado para el sector ferretero bajo el régimen simple de tributación, centrándose en las decisiones tecnológicas y metodológicas que guiarán su implementación. Primero, se explicará la elección del lenguaje de programación y el framework adecuados, tomando en cuenta factores como la facilidad de mantenimiento, escalabilidad y compatibilidad con las necesidades del sistema. Esta elección es fundamental para garantizar el rendimiento y la flexibilidad del sistema a medida que evoluciona.

A continuación, se abordará la arquitectura general del sistema, que define cómo se estructuraron los componentes y cómo interactúan entre sí, garantizando que el sistema sea robusto, eficiente y fácil de expandir en el futuro. Se discutieron las decisiones sobre la base de datos seleccionada, considerando la necesidad de manejar grandes volúmenes de información relacionada con inventarios, ventas y registros fiscales de manera eficiente y segura.

Además, se presentaron los diagramas de casos de uso, que ofrecen una representación visual de las interacciones clave entre los usuarios y el sistema, permitiendo comprender cómo se llevó a cabo cada función principal del sistema, desde la gestión de inventarios hasta la generación de reportes fiscales. Los casos de uso proporcionan una visión clara de las funcionalidades que el sistema debe cumplir y sirven como base para el desarrollo de la interfaz de usuario.

Se incluye también un modelo de entidad-relación, que describe las relaciones entre

las diferentes entidades del sistema, como productos, clientes, proveedores, ventas y facturas, lo que permite visualizar la estructura de datos necesaria para soportar las operaciones del sistema. Este modelo es esencial para asegurar la integridad y coherencia de la información almacenada en la base de datos.

Por último, se presentaron los mockups, que proporcionaron una representación visual preliminar de la interfaz de usuario, ayudando a anticipar cómo los usuarios interactúan con el sistema y garantizando que la experiencia de uso sea intuitiva y amigable. Estos elementos visuales son fundamentales para asegurar que el diseño no solo sea funcional, sino también accesible y fácil de utilizar para los dueños de las ferreterías.

El diseño detallado de estas componentes es crucial para asegurar que el sistema sea efectivo, adaptable a las necesidades específicas del sector ferretero, y capaz de mejorar significativamente la gestión operativa y fiscal de las ferreterías bajo el régimen simple de tributación. Este enfoque integral sienta las bases para una implementación exitosa y para el cumplimiento de los objetivos establecidos en el proyecto.

2.1 Principios tenidos en cuenta en el diseño

El diseño de FerreCloud se fundamentó en principios de usabilidad y en un enfoque de diseño centrado en el usuario, garantizando que el sistema sea accesible, intuitivo y funcional para los empleados de ferreterías. A continuación, se detallan los principios considerados durante el desarrollo del sistema.

2.1.1 Principios de Usabilidad

Siguiendo los principios de usabilidad de Jakob Nielsen [3], el diseño y la evaluación de FerreCloud se fundamentaron en los siguientes aspectos esenciales para garantizar una experiencia de usuario eficiente y satisfactoria:

- **Visibilidad del estado del sistema:** El sistema proporciona retroalimentación clara y constante sobre las acciones realizadas, mostrando confirmaciones de operaciones exitosas y alertas en caso de errores.
- **Correspondencia entre el sistema y el mundo real:** Se emplean términos, iconografía y estructuras familiares para los empleados de ferreterías, minimizando la necesidad de adaptación y facilitando la interacción con el software.
- **Consistencia y estándares:** La interfaz mantiene una estructura uniforme en toda la plataforma, asegurando que elementos similares se comporten de manera predecible y facilitando el aprendizaje del sistema.
- **Prevención de errores:** Se diseñaron validaciones en formularios y restricciones en los datos ingresados para evitar registros incorrectos, reduciendo la posibilidad de errores humanos.
- **Reconocimiento en lugar de recuerdo:** Se optimizó la interfaz para que los usuarios puedan reconocer opciones y funcionalidades sin necesidad de recordar comandos o secuencias complejas.
- **Diseño estético y minimalista:** Se evitó la sobrecarga de información, priorizando una presentación clara y jerárquica de los elementos más relevantes.

Estos principios garantizaron que FerreCloud no solo cumpliera con su propósito funcional, sino que también ofreciera una experiencia de usuario fluida y accesible.

2.1.2 Diseño Centrado en el Usuario

El desarrollo de FerreCloud siguió un enfoque de diseño centrado en el usuario, asegurando que la plataforma respondiera a las necesidades reales de los empleados de ferreterías, quienes pueden no tener amplia experiencia con sistemas digitales. Para ello, se consideraron los siguientes aspectos:

- **Comprensión del usuario:** Se llevaron a cabo observaciones y entrevistas con empleados de ferreterías para conocer sus tareas diarias, dificultades con la tecnología y necesidades específicas.
- **Interfaz intuitiva:** Se utilizaron términos y elementos gráficos familiares para los usuarios, reduciendo la necesidad de capacitación y facilitando el uso del sistema desde el primer contacto.
- **Evaluación mediante pruebas de usuario:** Durante el desarrollo, se realizaron pruebas con empleados de ferreterías para detectar posibles dificultades y mejorar la usabilidad del sistema con base en sus comentarios.

Este enfoque permitió que FerreCloud sea una herramienta accesible y eficiente, facilitando la digitalización de procesos sin representar una barrera tecnológica para los usuarios.

2.2 Tecnologías

En esta sección se describen las tecnologías utilizadas en el trabajo de grado y su justificación correspondiente

2.2.1 *Lenguaje de programación*

La selección de Python para el proyecto de grado fue estratégica, principalmente por su versatilidad, escalabilidad y condición de tecnología de vanguardia. La flexibilidad de Python permite la creación y el desarrollo rápidos de prototipos, algo crucial para un proyecto con requisitos dinámicos y objetivos en evolución. Su extenso ecosistema de bibliotecas permite a los desarrolladores implementar funcionalidades complejas de manera eficiente, ahorrando tiempo y esfuerzo. Además, la legibilidad y la sintaxis limpia de Python agilizan la colaboración entre los miembros del equipo, mejorando la productividad y el mantenimiento del código durante todo el ciclo de vida del proyecto.

Además, la escalabilidad de Python lo convierte en una opción ideal para proyectos con diferentes cargas de datos y demandas computacionales. Ya sea que se manejen tareas de procesamiento de datos a pequeña escala o se administren conjuntos de datos a gran escala, la solidez de Python garantiza un rendimiento y una confiabilidad óptimos. Su soporte para procesamiento paralelo y computación distribuida facilita una escalabilidad perfecta, lo que permite que el proyecto de grado se adapte a los requisitos cambiantes y se adapte al crecimiento futuro sin esfuerzo [23].

Además, la posición de Python como tecnología de vanguardia subraya su

relevancia en los panoramas de desarrollo de software contemporáneos. Con su adopción generalizada en todas las industrias y su papel fundamental en diversos sistemas de información actuales, Python equipa a los desarrolladores con las herramientas y técnicas necesarias para abordar los desafíos modernos e innovar de manera efectiva. Al aprovechar las funciones avanzadas de Python y mantenerse al tanto de los últimos desarrollos en el lenguaje, el proyecto de grado puede mantenerse a la vanguardia y ofrecer resultados impactantes que resuenan con las necesidades y expectativas cambiantes de la industria.

2.2.2 *Framework*

Elegir Django para un proyecto de grado en sistemas de información ofrece numerosas ventajas alineadas con los objetivos y requisitos del proyecto. Django, un marco web Python de alto nivel, es reconocido por su solidez, características de seguridad y capacidades de rápido desarrollo, lo que lo convierte en una opción ideal para crear sistemas de información sofisticados [24]. Aquí hay varias razones por las que se podría elegir Django para tal proyecto:

Desarrollo rápido: La filosofía de "baterías incluidas" de Django proporciona a los desarrolladores un conjunto completo de herramientas y bibliotecas, lo que les permite crear rápidamente aplicaciones web con todas las funciones. Su autenticación incorporada, su interfaz de administración y su ORM (mapeo relacional de objetos) aceleran significativamente el proceso de desarrollo, lo que permite al equipo del proyecto concentrarse en implementar funcionalidades principales en lugar de reinventar la rueda.

Escalabilidad: La escalabilidad de Django está bien documentada, lo que lo hace adecuado para proyectos de diversos tamaños y complejidades. Su capacidad para manejar mucho tráfico y grandes conjuntos de datos garantiza que el sistema de información pueda crecer sin problemas a medida que aumentan las demandas de los usuarios con el tiempo. Además, el soporte de Django para mecanismos de almacenamiento en caché, técnicas de optimización de bases de datos y ejecución de tareas asincrónicas mejora aún más la escalabilidad, garantizando un rendimiento óptimo incluso bajo cargas pesadas.

Seguridad: La seguridad es primordial para cualquier sistema de información, especialmente aquellos que tratan con datos sensibles. Django viene con numerosas funciones de seguridad integradas, incluida la protección contra vulnerabilidades web comunes, como la inyección SQL, secuencias de comandos entre sitios (XSS) y falsificación de solicitudes entre sitios (CSRF). Además, el sistema de autenticación de Django, sus sólidas capacidades de gestión de usuarios y el cumplimiento de las mejores prácticas en seguridad web ayudan a mitigar los riesgos y salvaguardar la integridad y confidencialidad de los datos del sistema [25].

Comunidad y ecosistema: Django cuenta con una comunidad vibrante y solidaria de desarrolladores, contribuyentes y entusiastas. Esta comunidad activa garantiza soporte continuo, actualizaciones frecuentes y una gran cantidad de paquetes y extensiones de terceros que amplían la funcionalidad de Django. Aprovechar este rico ecosistema permite al equipo del proyecto acceder a una amplia gama de recursos, documentación y soluciones impulsadas por la comunidad, acelerando así los esfuerzos de desarrollo y resolución de problemas.

Documentación y recursos de aprendizaje: Django ofrece documentación completa y abundantes recursos de aprendizaje, haciéndolo accesible a desarrolladores de todos los niveles. Ya sean programadores novatos o veteranos, los miembros del equipo pueden familiarizarse rápidamente con los conceptos, convenciones y mejores prácticas de Django, lo que facilita la colaboración fluida y el intercambio de conocimientos dentro del equipo del proyecto.

2.2.3 Base de datos

PostgreSQL se presenta como una opción sólida y confiable para el desarrollo de un sistema de información orientado a ferreterías bajo el régimen simple. Una de sus principales fortalezas radica en su robustez y fiabilidad, reconocidas a nivel global, lo que lo convierte en una base de datos idónea para gestionar datos críticos en un entorno comercial. La integridad de los datos es fundamental en sistemas que manejan transacciones financieras, inventarios y relaciones con proveedores y clientes, elementos clave en el contexto de una ferretería.

Además, PostgreSQL ofrece soporte completo para transacciones ACID (Atomicidad, Consistencia, Aislamiento y Durabilidad), lo que garantiza que las operaciones se ejecuten de manera segura y coherente, sin riesgo de corrupción o pérdida de datos [26]. Esto es esencial en un sistema donde las transacciones de ventas, compras y actualizaciones de inventario deben realizarse con precisión para evitar discrepancias que puedan impactar la administración del negocio.

La capacidad de PostgreSQL para gestionar modelos de datos complejos es otro factor determinante. Este sistema de gestión de bases de datos relacionales permite trabajar con claves foráneas y realizar uniones complejas, lo cual es crucial para un sistema de gestión de ferreterías que maneja relaciones entre productos, clientes, proveedores y transacciones. Esta flexibilidad permite diseñar un esquema de base de datos que represente fielmente la estructura operativa de la ferretería, facilitando la gestión integral del negocio.

Por otro lado, PostgreSQL destaca por su extensibilidad, permitiendo a los

desarrolladores agregar tipos de datos, operadores y funciones personalizadas. Esto es particularmente útil en un sistema para ferreterías, donde las necesidades específicas del negocio, como cálculos de impuestos o descuentos, pueden requerir personalizaciones que otros sistemas de bases de datos no ofrecen de manera nativa. Esta capacidad de adaptación asegura que el sistema pueda evolucionar junto con las necesidades del negocio.

A nivel de rendimiento, PostgreSQL maneja de manera eficiente grandes volúmenes de datos, lo cual es relevante en un entorno donde se procesan y almacenan múltiples transacciones y registros a lo largo del tiempo. En un sistema de información para ferreterías, que puede acumular grandes cantidades de datos sobre productos, ventas y clientes, el rendimiento de la base de datos es un aspecto crucial para garantizar la velocidad y eficiencia del sistema.

La amplia comunidad de usuarios y desarrolladores de PostgreSQL garantiza un soporte continuo y una constante mejora de la plataforma. Este respaldo es especialmente útil en un proyecto académico, donde la documentación y el acceso a recursos de aprendizaje son fundamentales. Asimismo, el licenciamiento gratuito y de código abierto de PostgreSQL reduce los costos de implementación, lo que lo convierte en una opción económicamente viable tanto para el desarrollo académico como para su uso en pequeñas y medianas empresas, como las ferreterías.

Finalmente, PostgreSQL se integra de manera eficiente con tecnologías como Django, el framework utilizado en este proyecto. Esta integración simplifica el desarrollo y despliegue del sistema de información, permitiendo una gestión fluida de la base de datos y facilitando el uso de herramientas como PgAdmin para su

administración. En resumen, PostgreSQL es una opción estratégica que ofrece un equilibrio entre robustez, flexibilidad y costos, ajustándose perfectamente a las necesidades de un sistema de información para ferreterías bajo el régimen simple.

2.2.4 *Arquitectura*

Desarrollar un sistema de información utilizando una arquitectura cliente-servidor ofrece varias ventajas significativas, lo que lo convierte en una opción convincente para muchos proyectos, incluidos aquellos en un contexto académico o educativo.

- **Escalabilidad:** Las arquitecturas cliente-servidor permiten una implementación escalable, donde varios clientes pueden conectarse a un servidor central. Esta escalabilidad garantiza que el sistema pueda adaptarse a un número creciente de usuarios o cargas de datos cada vez mayores sin una degradación significativa del rendimiento.
- **Gestión Centralizada:** En una arquitectura cliente-servidor, los datos y la lógica empresarial están centralizados en el servidor, lo que facilita la gestión y el mantenimiento. Las actualizaciones, parches y mejoras de seguridad se pueden aplicar de forma centralizada, lo que reduce la sobrecarga de administrar instalaciones de clientes individuales.
- **Seguridad:** Las arquitecturas cliente-servidor suelen ofrecer una mejor seguridad en comparación con los sistemas peer-to-peer o independientes. Los controles de acceso, los mecanismos de autenticación y el cifrado se pueden implementar de manera más efectiva en un entorno centralizado. Esto garantiza que los datos confidenciales estén protegidos y solo sean accesibles para usuarios autorizados, lo cual es crucial para los sistemas de información que manejan información personal o confidencial.
- **Concurrencia y rendimiento:** Al centralizar el almacenamiento y el

procesamiento de datos en un servidor, las arquitecturas cliente-servidor pueden gestionar mejor la concurrencia y optimizar el rendimiento. El servidor puede manejar múltiples solicitudes de clientes simultáneamente, utilizando los recursos de manera eficiente y minimizando los tiempos de respuesta. Esta gestión de concurrencia es esencial para sistemas con múltiples usuarios que acceden y modifican datos simultáneamente.

- **Compatibilidad multiplataforma:** Las arquitecturas cliente-servidor permiten la compatibilidad multiplataforma, lo que permite a los clientes ejecutarse en varios dispositivos y sistemas operativos mientras se conectan a un servidor central. Esta flexibilidad garantiza que los usuarios puedan acceder al sistema desde diferentes dispositivos sin requerir configuraciones específicas de hardware o software.

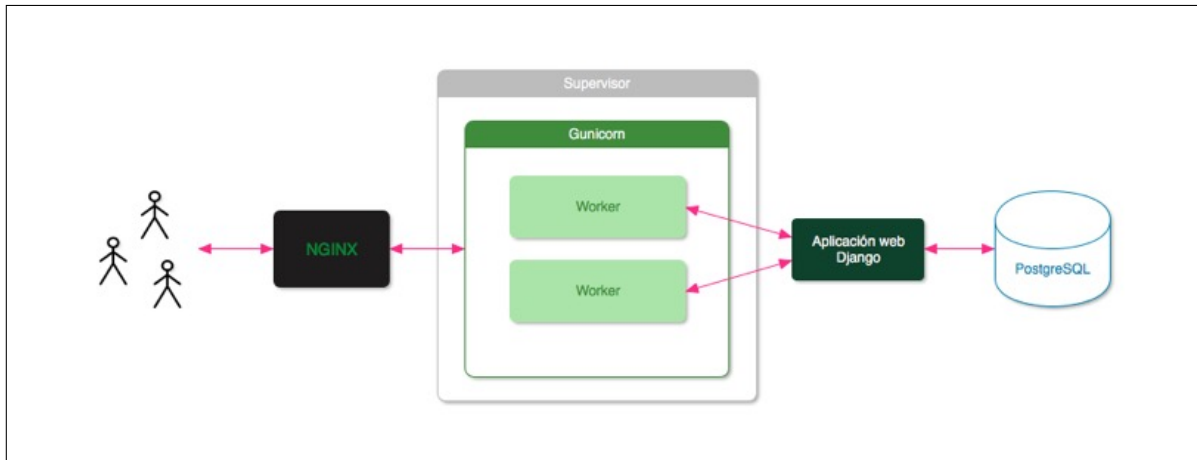


Figura 5.2: Diagrama ilustrativo de arquitectura cliente servidor

2.3 Diagramas

En el contexto de FerreCloud, los diagramas de modelado son herramientas esenciales que facilitan la comprensión y el diseño de la estructura y funcionalidad de una aplicación. Este trabajo de grado titulado "Sistema de información para ferreterías bajo régimen simple de tributación" se enfoca en la creación de un sistema eficiente y simplificado para la gestión de inventarios, ventas y tributación en ferreterías que operan bajo este régimen fiscal particular.

Para estructurar y clarificar los distintos componentes y funcionalidades del sistema, se han elaborado diversos diagramas que proporcionan una visión detallada y comprensible de su arquitectura y funcionamiento. A continuación, se presentan los diagramas de casos de uso, el modelo de entidad-relación (ER) y el diagrama de clases, los cuales en conjunto ofrecen una visión integral del sistema.

2.3.1 *Diagrama de casos de uso*

Los diagramas de casos de uso ilustran las interacciones entre los usuarios (actores) y el sistema, definiendo las principales funcionalidades que se deben implementar. Estos diagramas ayudan a identificar y especificar los requisitos del sistema, asegurando que todas las necesidades de los usuarios sean contempladas [27]. En el caso de nuestro sistema, los actores principales incluyen el administrador de la ferretería, el vendedor y el empleado de bodega, cada uno con diferentes roles y responsabilidades dentro del sistema, a continuación se procede a mostrar los casos de uso de cada uno de los actores.

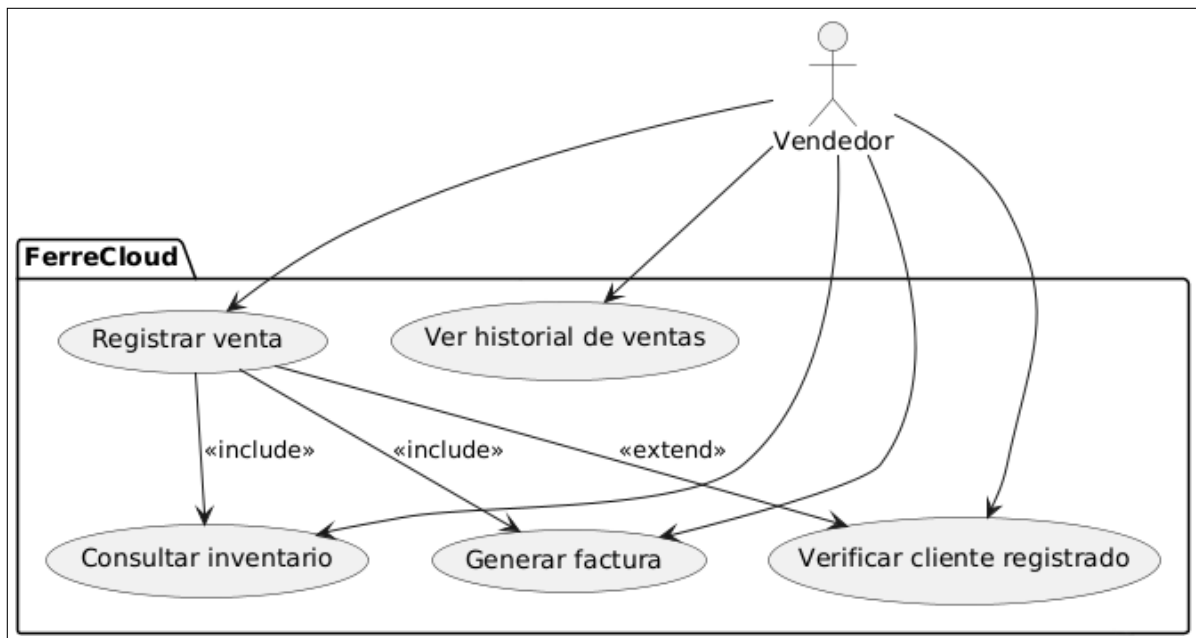


Figura 5.3: Diagrama de casos de uso para el usuario vendedor

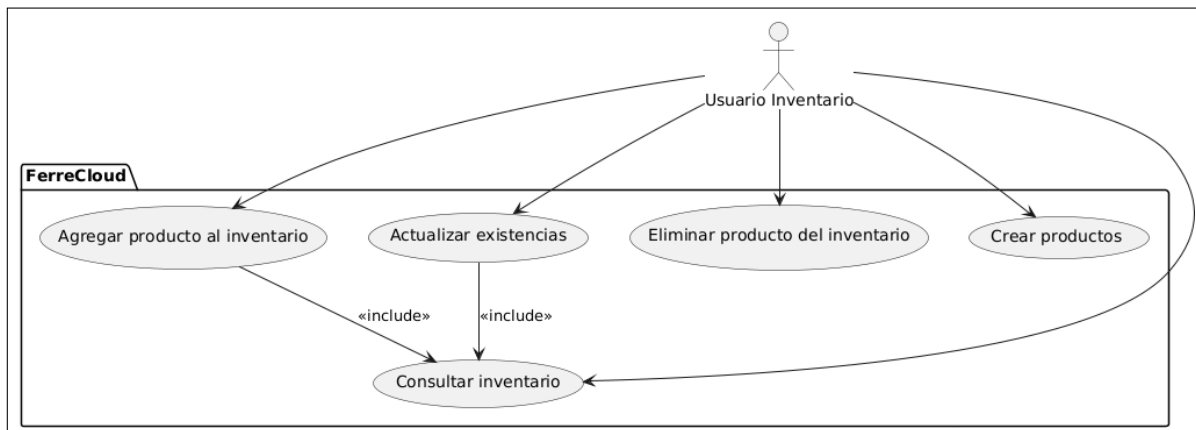


Figura 5.4: Diagrama de casos de uso para el usuario inventario

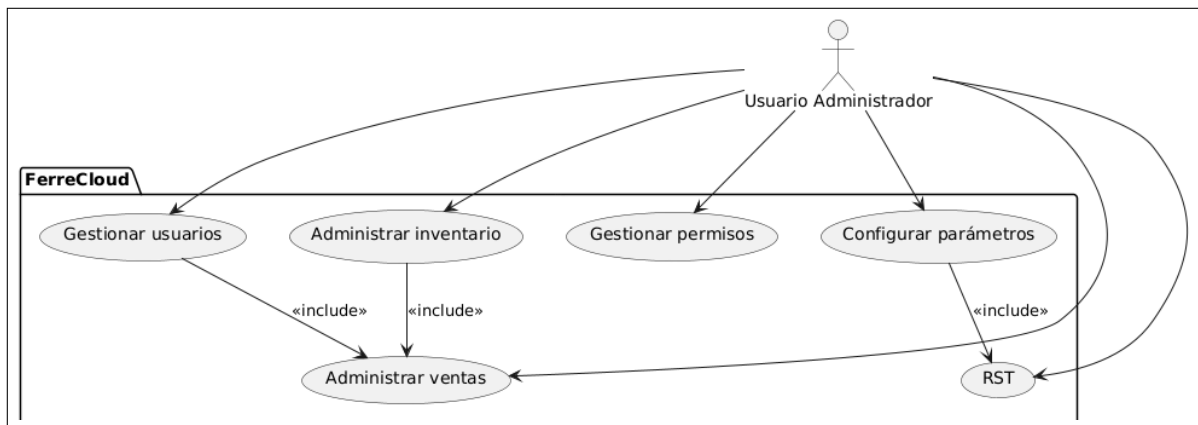


Figura 5.5: Diagrama de casos de uso para el usuario administrador

2.3.2 *Diagrama de secuencia*

Los diagramas de secuencia son una herramienta fundamental en el modelado de sistemas, ya que permiten visualizar cómo interactúan los componentes de un sistema a lo largo del tiempo. En el contexto de FerreCloud, los diagramas de secuencia proporcionan una representación detallada de los flujos de interacción entre los distintos actores y el sistema en los procesos clave del negocio. Estos diagramas ayudan a comprender el comportamiento de la aplicación en escenarios específicos [27], como la gestión de ventas, el registro de clientes y la administración de inventarios, entre otros. A continuación, se presentarán los diagramas de secuencia que describen el comportamiento de las funciones más importantes dentro de FerreCloud, facilitando la comprensión de cómo las solicitudes y respuestas son manejadas en el sistema, lo que contribuye a una mejor implementación y mantenimiento del software.

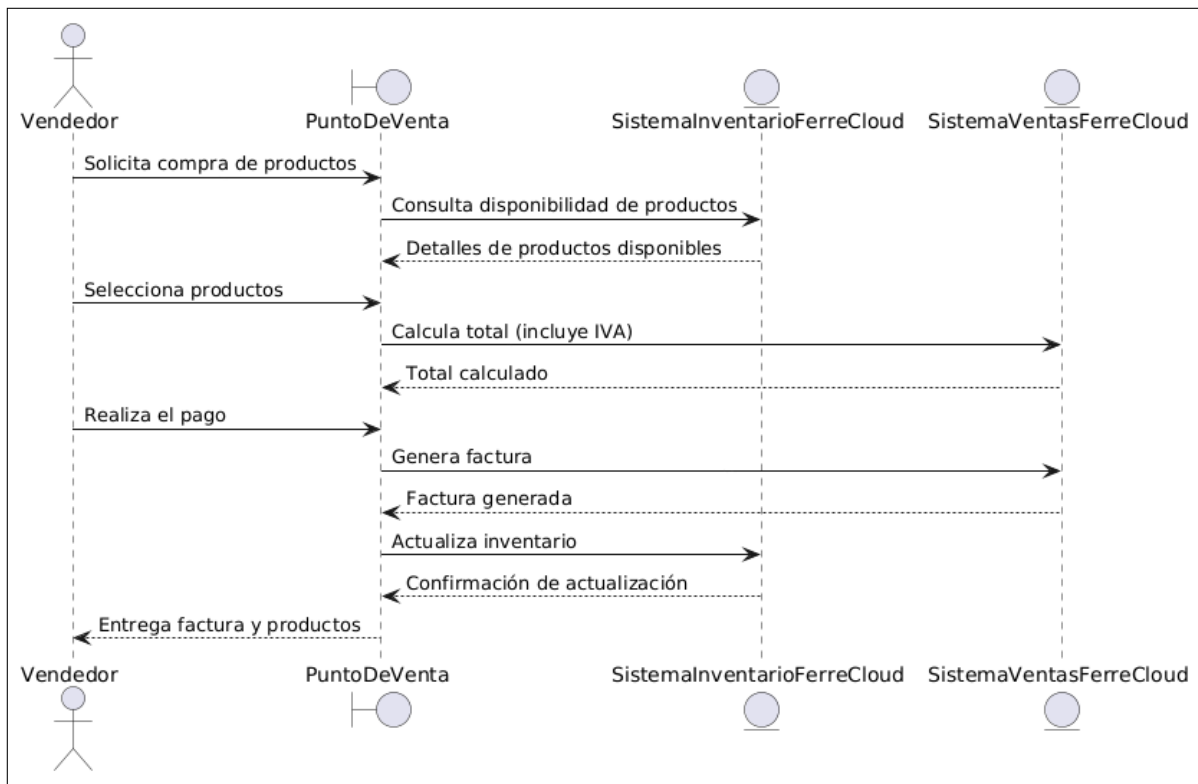


Figura 5.6: Diagrama de secuencia de ventas

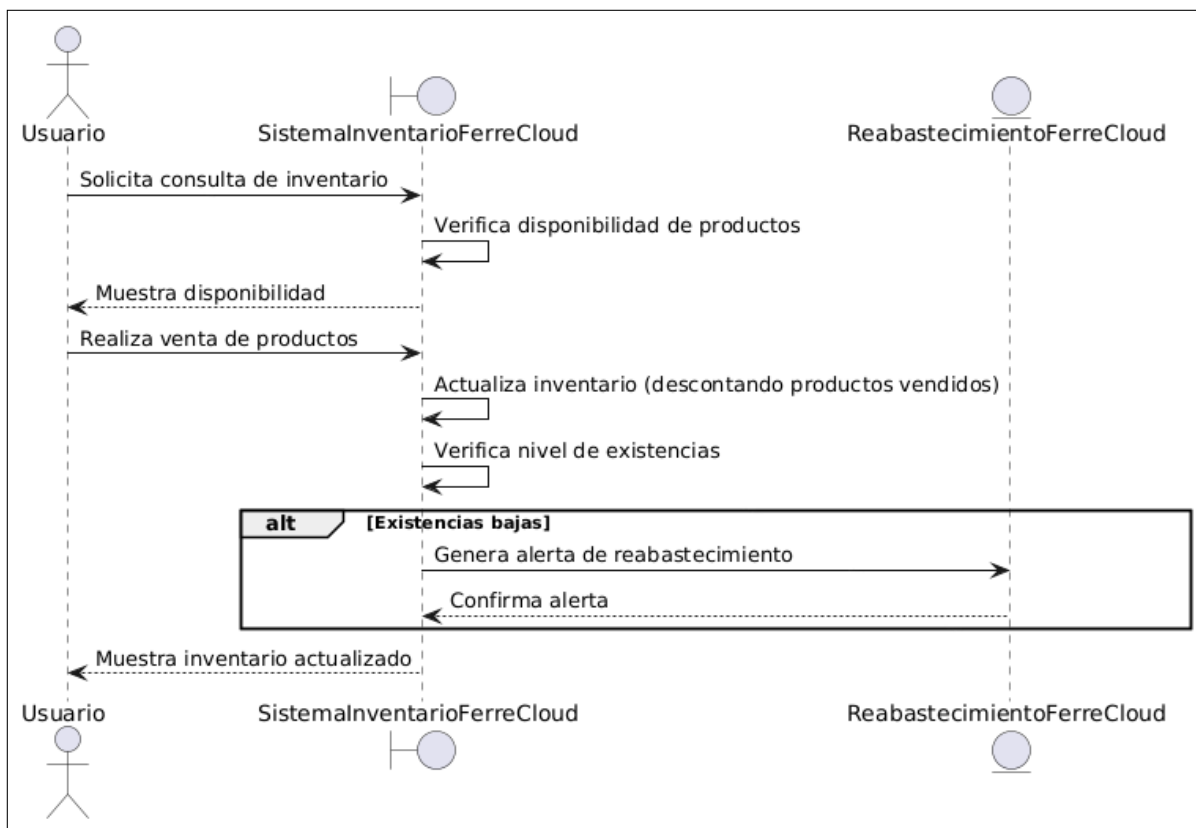


Figura 5.7: Diagrama de secuencia de inventario

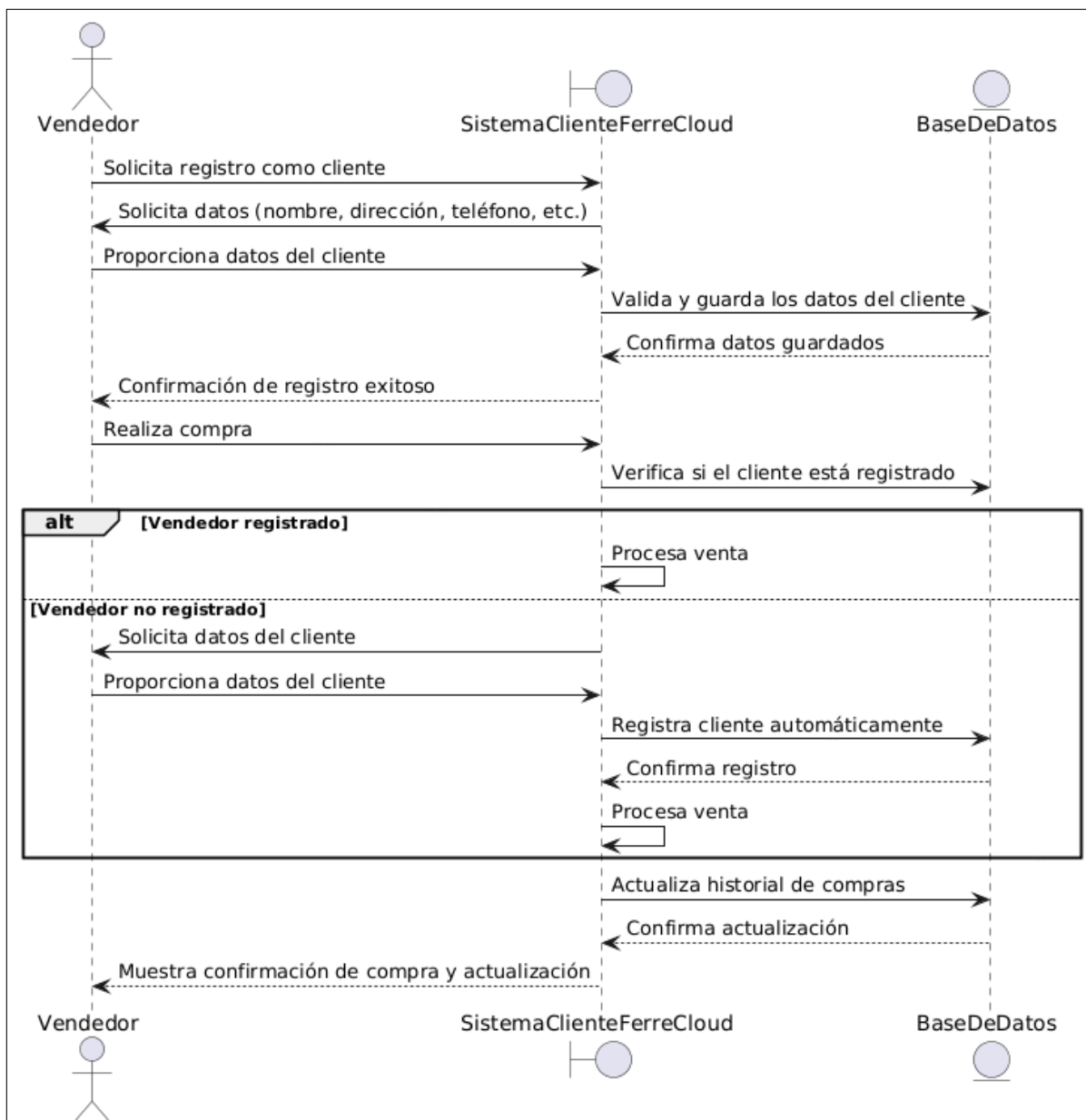


Figura 5.8: Diagrama de secuencia de clientes

2.3.3 Modelo de entidad de relación

El modelo de entidad-relación es una representación gráfica que describe la estructura lógica de la base de datos del sistema. Este diagrama muestra las entidades relevantes, sus atributos y las relaciones entre ellas. Para el "Sistema de información para ferreterías bajo régimen simple de tributación", las entidades clave incluyen productos, clientes, ventas, proveedores y transacciones tributarias. Este modelo es crucial para diseñar una base de datos que soporte eficientemente las operaciones del sistema.

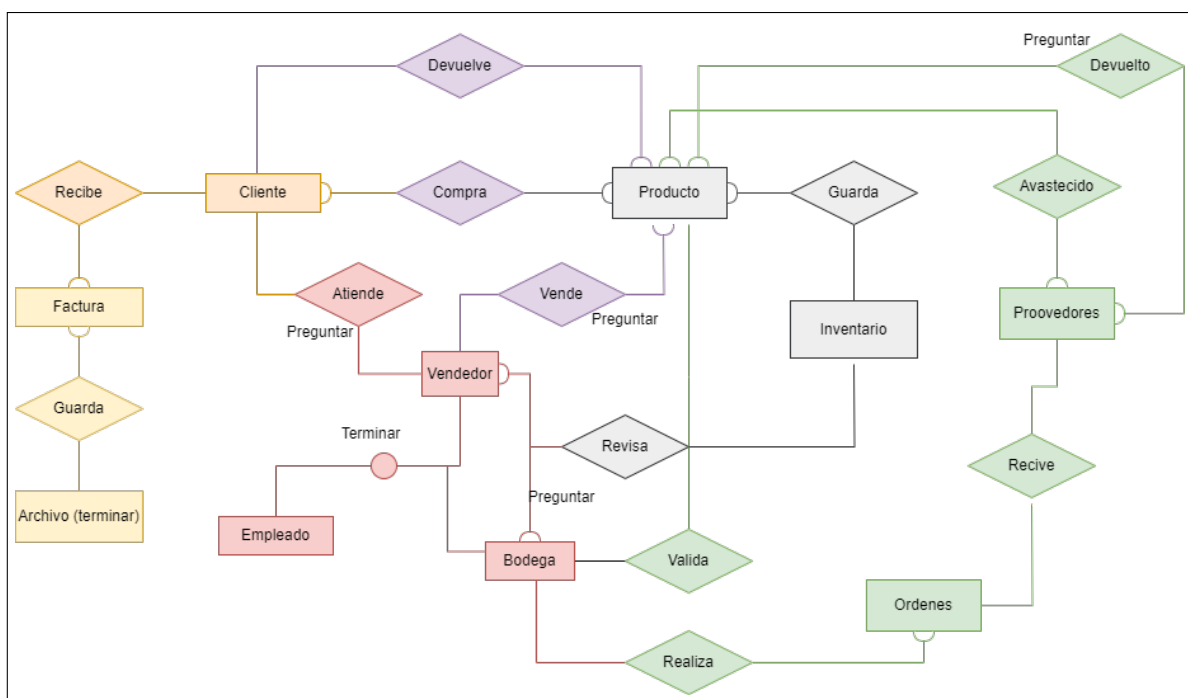


Figura 5.9: Diagramas de entidad de relación

2.4 Mockups

En esta sección, se presentan los mockups clave de la aplicación, incluyendo las pantallas principales y los módulos más representativos. Cada mockup ha sido diseñado para reflejar los requisitos específicos del proyecto, asegurando que el sistema cumpla con las necesidades operativas y funcionales previamente establecidas. A través de estas representaciones, se pueden identificar mejoras en el flujo de trabajo, facilitar el desarrollo teniendo una base visual la cual representar en forma de código.

A continuación se muestran algunas imágenes correspondientes a los Mockups del SGI. El resto de los Mockups se aprecian en la sección anexos

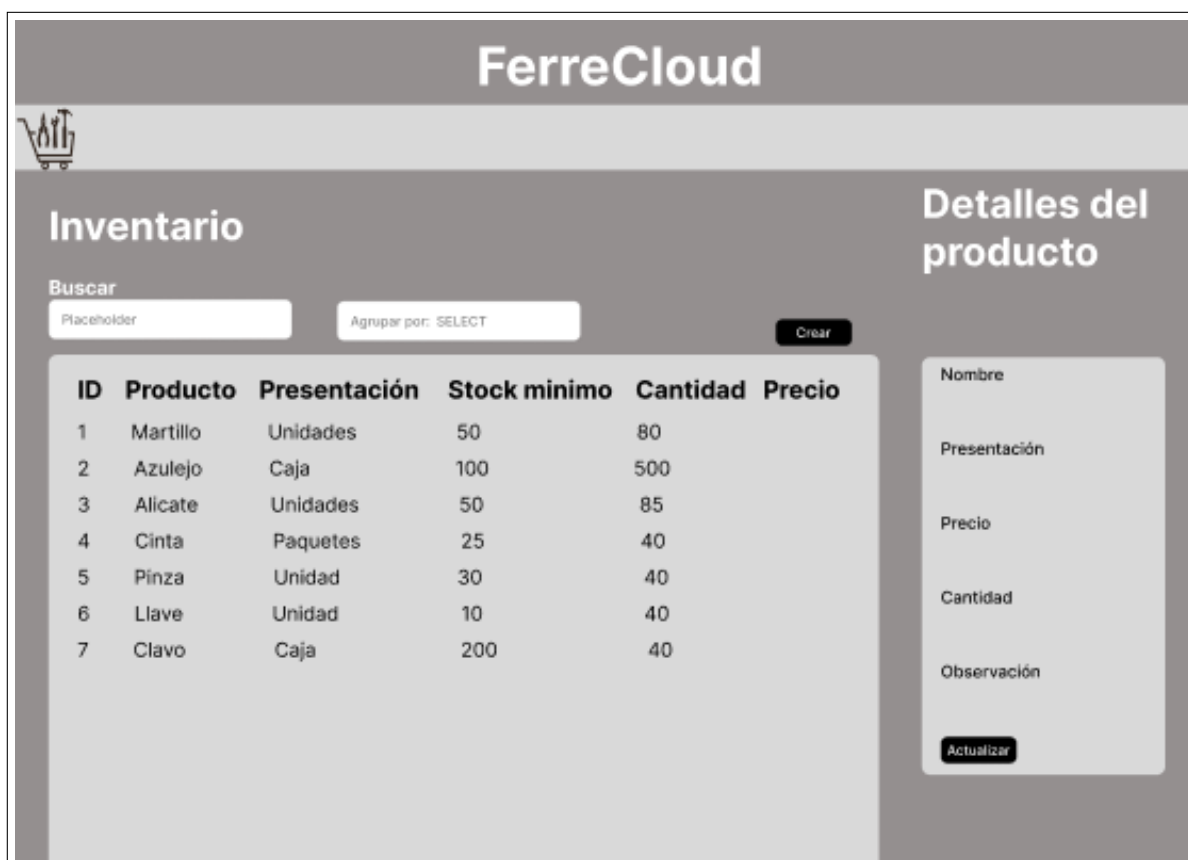


Figura 5.10: Vista de los mockups del módulo de Inventario.

FerreCloud

Colaborador

Documento: Placeholder

Nombre: Placeholder

Telefono: Placeholder

Dirección: Placeholder

Email: Placeholder

Salario: Placeholder

Celular: Placeholder

Caja de compensación: Placeholder

Cargo: Placeholder

Estado: ☒ Habilitado ☐ Deshabilitado

Guardar **Eliminar**

Información de contacto: Teléfono: XXX-XXX-XXXX | Correo electrónico: info@ferreteria.com

Figura 5.11: Vista de los mockups de edición de colaborador.



Figura 5.12: Vista de los mockups del módulo de ventas.

Para ver el repositorio completo de mockups, dirigirse a la sección de Anexos

3 Codificación

El proceso de codificación de Ferrecloud fue una fase clave en el desarrollo del proyecto, ya que permitió la implementación de las funcionalidades esenciales para la gestión de las operaciones de las ferreterías bajo el marco legal colombiano. Este proceso se dividió en varias etapas, que incluyeron la aplicación de la planificación, el diseño de la arquitectura del sistema, el aprendizaje de las tecnologías seleccionadas, el desarrollo de cada módulo y, las pruebas continuas de funcionalidad y rendimiento.

La codificación del sistema se centró en asegurar que las funcionalidades fueran implementadas de manera eficiente, modular y escalable. A lo largo del desarrollo, se hizo uso de prácticas de programación modernas, adoptando prácticas que permitieran un mantenimiento sencillo y la integración de nuevas características en el futuro. Además, se tomó en cuenta la necesidad de cumplir con los requerimientos fiscales del régimen simple de tributación, lo que implicó una integración precisa de cálculos de impuestos, generación de facturas electrónicas y gestión de inventarios.

La elección previa de herramientas y tecnologías fue un factor decisivo en la construcción de un sistema robusto y adaptable a las necesidades del usuario. Se utilizó el framework Django, un entorno de desarrollo ágil y seguro para aplicaciones web, que facilitó la creación de una plataforma confiable y de alto rendimiento. A su vez, se implementaron prácticas de control de versiones mediante Git, lo que permitió un trabajo seguro realizando un control adecuado de los cambios realizados en el código.

El proceso de codificación no solo implicó la implementación de las funcionalidades

solicitadas, sino también la constante iteración y mejora del código a través de pruebas unitarias y de integración, con el objetivo de garantizar un funcionamiento correcto y la satisfacción de los usuarios finales. Además, se priorizó la creación de una interfaz de usuario intuitiva, que facilitara la interacción con el sistema y mejorara la experiencia general de los usuarios.

3.1 Cumplimiento de requerimientos

El cumplimiento de los requerimientos funcionales del *Sistema de Información de Ferreterías bajo Régimen Simple de Tributación* se organizó y gestionó a través de la metodología de gestión de actividades seleccionada previamente, la cual fué el tablero Kanban, lo que permitió un control eficiente de las tareas y una ejecución ágil y escalable del proyecto. A continuación, se detalla cómo se abordaron los requerimientos utilizando este enfoque de gestión de tareas:

3.1.1 Sprints y Ciclos de Trabajo

El proyecto se organizó en sprints, que son ciclos de trabajo que permitieron entregar avances parciales y funcionales del sistema de manera continua. Cada sprint tuvo una duración establecida de una a dos semanas y un conjunto claro de tareas a realizar, relacionadas con los requerimientos funcionales del sistema. Estos sprints fueron fundamentales para mantener un ritmo constante de desarrollo y asegurar que se cumplieran los plazos establecidos.

Cada sprint comenzó con una planificación, en la que se definieron las tareas prioritarias a realizar según los requerimientos y el impacto de cada funcionalidad en el sistema. Una vez que el sprint inició, se trabajó de manera concentrada en las tareas asignadas, con revisiones periódicas para evaluar el progreso y hacer ajustes según fuera necesario. Al final de cada sprint, se realizó una reunión de revisión con el cliente, lo que permitió evaluar los avances, resolver dudas y ajustar los requerimientos de acuerdo con la retroalimentación recibida.

Durante los sprints, el tablero Kanban fue clave para seguir el progreso de cada tarea. Se utilizó para gestionar de manera eficiente el flujo de trabajo y asegurar que las tareas se movieran de acuerdo con su estado (desde la columna de “Por Hacer” hasta “Hecho”). El proceso de revisión continua durante cada sprint garantizó que se mantuvieran altos estándares de calidad y que se cumplieran los objetivos de cada fase del proyecto.

Además, los sprints permitieron ajustar el rumbo del proyecto en tiempo real, brindando la flexibilidad necesaria para adaptarse a nuevas necesidades o cambios en los requerimientos, lo cual es uno de los beneficios clave de trabajar con una metodología ágil. Al final de cada ciclo, se consolidó el trabajo realizado, lo que permitió avanzar con mayor claridad hacia el siguiente conjunto de tareas del proyecto.

3.1.2 Definición de Tareas por Requerimiento Funcional

Cada requerimiento funcional fue asignando en tareas individuales que representaban su implementación y posteriormente se colocó en el tablero Kanban en la columna de “*Por Hacer*”.

3.1.3 Priorización de Tareas

A través de una revisión continua de los requerimientos y los avances del proyecto, las tareas fueron priorizadas de acuerdo con su impacto en el cumplimiento de los objetivos del sistema. Las tareas críticas que formaban la columna vertebral del sistema, como la

integración de las ventas y la generación de facturas, se asignaron una alta prioridad y se movieron rápidamente a la columna de *“En Proceso”*.

3.1.4 Columna de *“En Proceso”* (Desarrollo Activo)

Las tareas que estaban en desarrollo pasaron a la columna de *“En Proceso”*. Durante esta fase, se trabajó en la implementación de las funcionalidades específicas según los requerimientos. El tablero Kanban permitió visualizar en todo momento el estado de cada tarea, lo que facilitó la colaboración entre el estudiante y el director lo que permitió detectar posibles bloqueos de manera temprana. Por ejemplo, en la implementación de la *“gestión de ventas”*, se tuvo que ajustar el cálculo del IVA de acuerdo con la parametrización, lo que llevó a mover esa tarea a la columna de *“En Revisión”*.

3.1.5 Revisión y Validación

Una vez que las tareas fueron desarrolladas, se movieron a la columna de *“Revisión”*, donde se realizaron pruebas para verificar que cada funcionalidad cumpliera con los requerimientos. Este proceso fue fundamental para asegurar que la funcionalidad estuviera alineada con las expectativas del cliente y que no se introdujeran errores en el sistema. Los resultados de las pruebas se documentaron en el tablero, y cualquier error identificado se retroalimentó al equipo de desarrollo, quien volvió a trabajar en la tarea para corregir los problemas antes de su aprobación final.

3.1.6 *Columna de “Hecho” (Finalización de Funcionalidades)*

Después de la revisión y validación, las tareas que cumplían con los requerimientos funcionales fueron movidas a la columna de *“Hecho”*. Estas tareas representaban funcionalidades completamente implementadas y validadas, como la correcta emisión de facturas o el cálculo del IVA. La organización del tablero Kanban permitió verificar que todos los aspectos funcionales del sistema estuvieran completos antes de la entrega final del proyecto.

3.1.7 *Iteración y Mejoras Continuas*

A lo largo del ciclo de vida del proyecto, el tablero Kanban facilitó la iteración continua. Las tareas que requerían ajustes adicionales o mejoras basadas en comentarios de director o pruebas de usuario se movieron de nuevo a las columnas de *“En Proceso”* o *“En Revisión”*, garantizando una retroalimentación constante y la mejora continua del sistema.

3.1.8 *Transparencia y Colaboración*

Gracias a la estructura visual del tablero Kanban, todos los miembros del equipo pudieron tener una visión clara del estado del proyecto. Esto permitió una comunicación constante y una toma de decisiones más rápida, lo que contribuyó al cumplimiento puntual de los requerimientos funcionales establecidos para el sistema.

En conclusión, el uso de un tablero Kanban facilitó la planificación, seguimiento y

ejecución de las tareas necesarias para cumplir con los requerimientos funcionales de Ferrecloud. Este enfoque no solo mejoró la organización del equipo de trabajo, sino que también aseguró que cada funcionalidad fuera implementada de manera eficiente y conforme a las necesidades del proyecto.

A continuación, se procede a evidenciar el tablero Kanban realizado en Notion a una final media del proyecto:

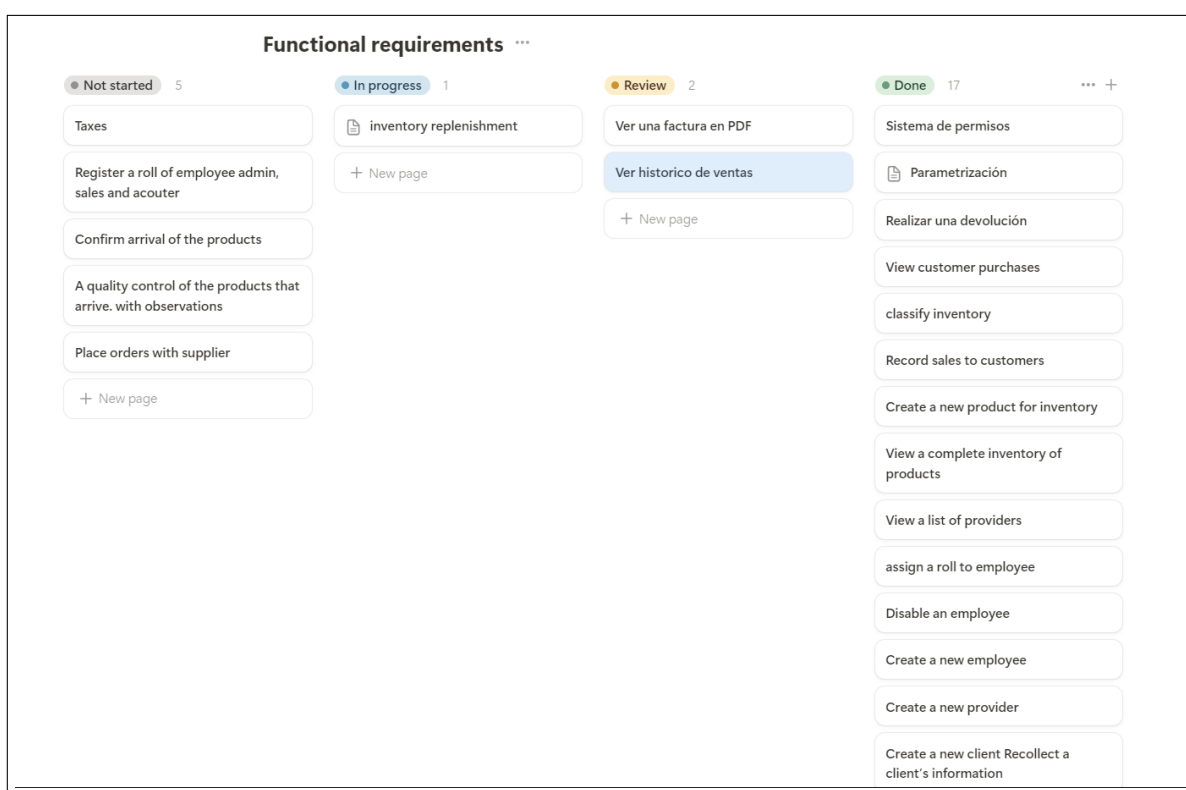


Figura 5.13: Captura de tablero Kanban del proyecto.

Para ilustrar el estilo de programación utilizado en el desarrollo del sistema, a continuación, se presenta un fragmento de código que demuestra la implementación de la lógica para procesar el carrito de compras. Este ejemplo refleja un enfoque estructurado y claro, donde se prioriza la gestión adecuada del inventario, el cálculo

preciso del total de la compra y el manejo de errores mediante validaciones específicas. Además, se emplea el modelo de datos de Django para garantizar la integridad de la información y facilitar la interacción con la base de datos.

```
def procesar_carrito(carrito, iva_tasa):
    total = Decimal('0.00')
    detalles = []
    for item in carrito:
        try:
            producto = Producto.objects.get(id=item['id'])
            cantidad = Decimal(item['cantidad'])
            if producto.cantidad >= cantidad:
                producto.cantidad -= cantidad
                producto.save()
                total += producto.precio * cantidad
                detalles.append({
                    'producto': producto,
                    'cantidad': cantidad,
                    'precio_unitario': producto.precio
                })
            else:
                return None, f"No hay suficiente inventario para el producto: {producto.nombre}"
        except Producto.DoesNotExist:
            return None, f"El producto con ID {item['id']} no existe."
    return {'total': total, 'detalles': detalles}, None
```

Figura 5.14: Captura de fragmento de código del proyecto.

La función `recibirOrden` en Django permite gestionar la recepción de órdenes de reabastecimiento, asegurando el control de calidad de los productos antes de su ingreso al inventario. Cuando se recibe una solicitud POST, se procesan los detalles de calidad enviados desde el frontend en formato JSON. Si un producto no cumple con los estándares de calidad, se registran observaciones y se marca la orden con estado de Rechazada". En caso contrario, los productos aprobados se agregan al inventario y la orden se marca como Recibida". Esta implementación garantiza un control eficiente de la calidad en el proceso de reabastecimiento.

```
@login_required
def recibirOrden(request, orden_id):
    orden = get_object_or_404(Reabastecimiento, id=orden_id, eliminado=False)

    if request.method == "POST":
        # Obtener detalles de control de calidad
        detalles_calidad = json.loads(request.body) # Se espera un JSON desde el frontend
        errores = []

        for detalle in detalles_calidad:
            detalle_obj = get_object_or_404(ReabastecimientoDetalle, id=detalle['detalle_id'])
            detalle_obj.calidad = detalle['calidad'] # Guardar el estado de calidad
            detalle_obj.observaciones = detalle.get('observaciones', '')
            detalle_obj.save()

            if not detalle['calidad']: # Si calidad es False, marcar como problema
                errores.append(f"Producto {detalle_obj.producto.nombre} tiene problemas de calidad.")
            else:
                # Si el producto pasa el control de calidad, actualizar el inventario
                producto = detalle_obj.producto
                producto.cantidad += detalle_obj.cantidad # Aumenta la cantidad disponible en inventario
                producto.save()

        # Si hubo errores, guardar la orden como "Rechazada" (estado 2), si no, "Recibida" (estado 1)
        orden.estado = 2 if errores else 1
        orden.save()

        return JsonResponse({'success': True, 'errores': errores})

    return render(request, 'recibirOrden.html', {'orden': orden})
```

Figura 5.15: Captura de fragmento de código del proyecto.

3.2 Métricas

En el desarrollo de proyectos de ingeniería de sistemas, la evaluación y medición de resultados se convierten en aspectos fundamentales para garantizar la calidad y efectividad de las soluciones propuestas. Es en este contexto donde las métricas desempeñan un papel crucial, proporcionando indicadores cuantitativos que permiten medir el rendimiento, la eficiencia, la calidad del software, y el cumplimiento de los objetivos establecidos. La correcta definición y aplicación de métricas no solo facilita el seguimiento del progreso del proyecto, sino que también sirve como herramienta de apoyo en la toma de decisiones, la gestión de riesgos y la mejora continua post proyecto.

En este trabajo de grado, se exploraron diversas métricas clave utilizadas para evaluar el desempeño del sistema propuesto, abarcando aspectos como la rotación de código, la tasa de fallas, tiempo medio de detección, tiempo medio entre fallas, tiempo medio de resolución. La selección y aplicación de estas métricas se fundamenta en los **ATRIBUTOS CALIDAD ISO 9126**. A lo largo de esta sección, se busca resaltar la importancia de estas métricas en la aplicación en el desarrollo del prototipo planteado.

3.2.1 Rotación de código

Tal y como lo plantea A. Navarro ^{.Es} la cantidad de líneas de código agregadas, modificadas o eliminadas de una base de código durante un período de tiempo"[28], de esta misma manera se tomó el tiempo entre sprints para calcular por medio de los

commits de github, las líneas modificadas en ese determinado lapso de tiempo y así determinar el avance del proyecto en ese sprint. Como se evidencia en el ejemplo.

Teniendo en cuenta la fórmula:

$$\text{rotación de código} = \frac{\text{Agregadas} + \text{Modificadas} + \text{Eliminadas}}{\text{Tiempo}}$$

Y el registro de commit:

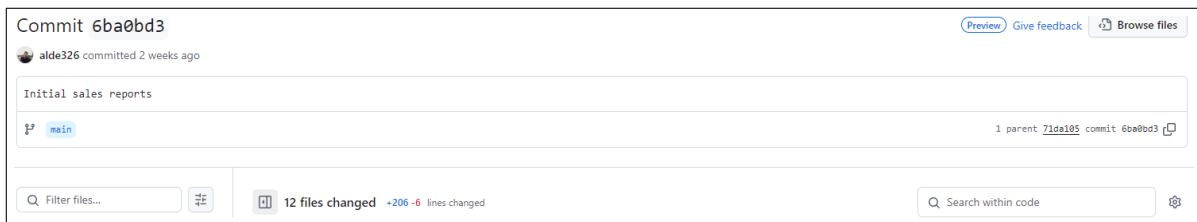


Figura 5.16: Captura de registro de commit de GitHub

$$\text{rotación de código} = \frac{212}{7} = 30.2$$

Este resultado de 30.2 nos brinda una referencia para saber que tanto se avanzó en el código respecto al sprint anterior.

3.2.2 Tasa de fallas

La tasa de fallas es un referente para saber la calidad de un producto de software en el cual se toman las variables del número de fallas encontradas y un lapso de tiempo

determinado [29](En el marco de este proyecto, se usó el tiempo de sprint). Cabe resaltar que una tasa de fallas alta es equivalente a una falta de calidad en el mismo.

Fórmula:

$$\text{Tasa de fallas} = \frac{\text{Numero de fallas encontradas}}{\text{Tiempo}}$$

En este caso se presentan dos aplicaciones de la fórmula, una a etapas iniciales de la implementación y otra en etapas finales de la misma.

Sprint 9

$$\text{Tasa de fallas} = \frac{12}{7} = 1.71$$

Sprint 20

$$\text{Tasa de fallas} = \frac{3}{7} = 0.4$$

Con la comparación de estas fórmulas, se evidencia la mejoría de FerreCloud en términos de calidad, haciendo que el software sea confiable en vista del usuario.

4 Pruebas

En esta sección se no solo se describe el proceso de pruebas llevado a cabo para asegurar que Ferrecloud funcione correctamente, en vez de esto se realizó un plan de aseguramiento de calidad que complementa las pruebas y de esta manera cumplir con los estándares de calidad que fueron propuestas para este proyecto y de esta manera cumplir con los requisitos establecidos. El objetivo del plan de aseguramiento de calidad es identificar y corregir errores, evaluar el rendimiento, y asegurar la usabilidad del sistema en un entorno de producción.

4.1 Contról de versiones

El control de versiones es fundamental para garantizar la trazabilidad de los cambios realizados al Plan de Aseguramiento de Calidad (PAC) a lo largo del desarrollo del proyecto. Este control asegura la disponibilidad de un historial claro de modificaciones, permitiendo identificar las mejoras, correcciones o actualizaciones realizadas.

Herramientas de Control de Versiones Para el manejo y la gestión de versiones del código fuente y documentos relacionados al proyecto, se utiliza el sistema de control de versiones Git, implementado a través de la plataforma GitHub.

Las principales funcionalidades utilizadas son:

- Control de Cambios: Rastreo detallado de modificaciones en el código y documentos.

- **Ramas (Branches):** Uso de ramas específicas para desarrollar nuevas funcionalidades o realizar pruebas antes de integrar los cambios en la rama principal.
- **Etiquetas (Tags):** Marcado de versiones oficiales del proyecto para facilitar su identificación.
- **Revisión y Aprobación:** Cada nueva versión debe ser revisada y aprobada por el director del proyecto antes de su publicación oficial.
- **Frecuencia de Actualizaciones:** Las actualizaciones deben realizarse cada vez que se completen cambios significativos en el documento o código fuente.

4.1.1 *Evolución de Versiones: Del MVP al Producto Final*

El desarrollo del sistema se llevó a cabo de manera iterativa, pasando por diferentes versiones desde el Producto Mínimo Viable (MVP) hasta la versión final completamente funcional. A continuación, se describen las principales fases de evolución del sistema:

- **Versión MVP (Producto Mínimo Viable):** En esta fase inicial, se desarrolló una versión básica del sistema con las funcionalidades esenciales para validar su viabilidad y obtener retroalimentación temprana. Se priorizó la implementación del módulo de ventas y la generación de facturas.
- **Versión Beta:** Tras validar el MVP, se incorporaron mejoras y nuevas funcionalidades, incluyendo la gestión de clientes y la parametrización de impuestos. En esta versión se realizaron pruebas con usuarios clave para identificar oportunidades de optimización.

- **Versión RC (Release Candidate):** Esta versión incluyó correcciones de errores detectados en la fase beta, así como ajustes en la interfaz y optimización del rendimiento del sistema. Se realizaron pruebas exhaustivas para garantizar la estabilidad de la aplicación.
- **Versión Final:** Tras la validación del Release Candidate, se lanzó la versión final del sistema, que incorporó todas las funcionalidades planificadas, con una interfaz optimizada y el cumplimiento total de los requerimientos definidos en el proyecto.

El uso de un enfoque iterativo permitió mejorar continuamente el sistema a lo largo del desarrollo, asegurando que cada versión fuera más estable y funcional que la anterior. Además, el control de versiones facilitó la gestión de cambios, evitando la pérdida de información y permitiendo una implementación organizada y estructurada del proyecto.

4.2 Plan de Pruebas

Para garantizar la calidad del sistema, se empleó una metodología de pruebas que incluyó:

- **Pruebas Unitarias:** Cada módulo y función fueron evaluados de forma independiente para verificar que operen de manera correcta. Esto permitió detectar problemas en una fase temprana del desarrollo.
- **Pruebas de Integración:** Se realizaron para verificar que los distintos módulos del sistema (ventas, inventario, contabilidad, etc.) interactúan correctamente. Estas pruebas son esenciales en un sistema de información complejo, donde la integración entre componentes es clave.

- **Pruebas Funcionales:** Se enfocaron en validar que el sistema cumple con todos los requisitos funcionales especificados, como el registro de ventas, el cálculo de impuestos y la generación de reportes de inventario y ventas.
- **Pruebas de Rendimiento:** Se llevaron a cabo simulando escenarios de alta demanda para evaluar la eficiencia y el tiempo de respuesta del sistema bajo carga, asegurando que pueda manejar el tráfico en un entorno real sin pérdidas de rendimiento significativas.
- **Pruebas de Usabilidad:** Estas pruebas se realizaron con usuarios potenciales para evaluar la facilidad de uso del sistema, identificando mejoras en la interfaz de usuario y en la experiencia general.

4.3 Casos de prueba

En el desarrollo del Sistema de Información de Ferreterías bajo Régimen Simple de Tributación, se llevaron a cabo diversas pruebas para garantizar su correcto funcionamiento, estabilidad y usabilidad. Estas pruebas permitieron identificar y corregir errores en etapas tempranas del desarrollo, asegurar la integración adecuada entre los módulos y evaluar el desempeño del sistema bajo diferentes condiciones de uso. A continuación, se presenta un caso de prueba realizado por cada uno de los tipos de prueba, detallando su objetivo, los datos de entrada, los pasos a seguir y los resultados esperados.

Tipo de Prueba	Objetivo	Pasos	Resultado Esperado
Prueba Unitaria	Verificar que la función <code>test_procesar_carrito</code> calcule correctamente el total de una factura.	■ Definir una lista de productos con precio y cantidad. ■ Llamar a la función con los datos de entrada. ■ Comparar el resultado con el esperado.	El total debe calcularse correctamente incluyendo el IVA. Para productos con valores de prueba.

Tipo de Prueba	Objetivo	Pasos	Resultado Esperado
Prueba de Integración	Comprobar que al realizar una venta, el inventario se actualiza correctamente.	■ Registrar una venta de ciertos productos. ■ Consultar el inventario después de la transacción.	Los productos vendidos deben descontarse del inventario. Ejemplo: Si había 10 martillos y se vendieron 2, el inventario debe mostrar 8 martillos restantes .
Prueba Funcional	Validar que el sistema genera correctamente un reporte de ventas dentro de un rango de fechas.	■ Acceder al módulo de reportes. ■ Seleccionar un rango de fechas. ■ Generar y revisar el informe.	El reporte debe mostrar todas las ventas realizadas dentro del rango seleccionado con detalles como fecha, productos vendidos y totales.

Tipo de Prueba	Objetivo	Pasos	Resultado Esperado
Prueba de Rendimiento	Evaluar la eficiencia del sistema al procesar 500 ventas simultáneas.	■ Simular múltiples usuarios generando ventas concurrentes (ej. con Locust). ■ Medir el tiempo de respuesta y verificar estabilidad del sistema.	El sistema debe procesar todas las ventas sin errores ni bloqueos, con tiempos de respuesta menores a 2 segundos por transacción .

Tipo de Prueba	Objetivo	Pasos	Resultado Esperado
Prueba de Usabilidad	Evaluar la facilidad de uso del formulario de ventas con usuarios sin experiencia.	■ Pedir a usuarios nuevos que registren una venta sin asistencia. ■ Observar dificultades y recopilar retroalimentación.	Los usuarios deben completar la venta sin confusión. Si hay problemas, se identifican mejoras en la interfaz como botones más claros o mensajes de error detallados.

4.4 Pruebas Unitarias

Se desarrollaron pruebas unitarias para cada uno de los módulos clave, incluyendo:

- **Ventas:** Verificación de cálculo de totales, descuentos, y aplicación del IVA en función del régimen simple de tributación.

```
PS C:\Users\alde3\Documents\Workspace\FerreCloud-master> python manage.py test
>>
Found 6 test(s).
Creating test database for alias 'default'...
System check identified no issues (0 silenced).
OK: test_crear_detalles_factura pasó exitosamente
.OK: test_crear_factura pasó exitosamente
.OK: test_obtener_parametro_iva pasó exitosamente
OK: test_obtener_parametro_iva (sin parámetro IVA) pasó exitosamente
.OK: test_procesar_carrito pasó exitosamente
OK: test_procesar_carrito con inventario insuficiente o producto inexistente pasó exitosamente
.OK: test_registrar_cliente pasó exitosamente
OK: test_registrar_cliente (cliente existente) pasó exitosamente
.OK: test_registrar_ingreso pasó exitosamente
.
-----
Ran 6 tests in 0.144s

OK
Destroying test database for alias 'default'...
```

Figura 5.17: Captura de las pruebas unitarias del módulo de ventas

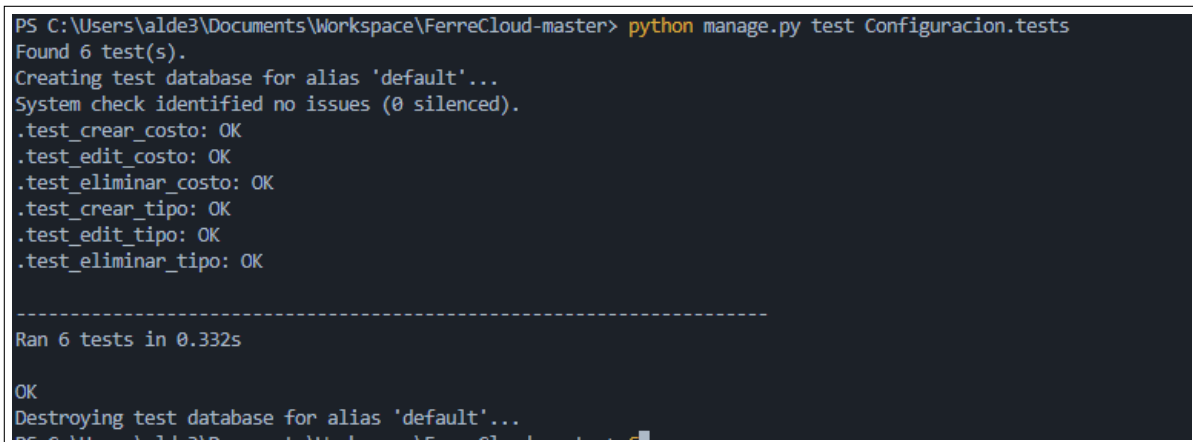
- **Inventario:** Pruebas para asegurar el correcto registro de productos, actualización de stock, y alertas de productos en bajo inventario.

```
PS C:\Users\alde3\Documents\Workspace\FerreCloud-master> python manage.py test Inventario.tests.InventariosViewsTestCase
Found 6 test(s).
Creating test database for alias 'default'...
System check identified no issues (0 silenced).
.test_edit_product_post_valid: OK
.test_eliminar_producto: OK
.test_index_inventarios_permission_denied: OK
.test_index_inventarios_with_permission: OK
.test_new_product_post_valid: OK
.test_product_search: OK
-----
Ran 6 tests in 26.064s

OK
Destroying test database for alias 'default'...
PS C:\Users\alde3\Documents\Workspace\FerreCloud-master>
```

Figura 5.18: Captura de las pruebas unitarias del módulo de inventarios

- **Contabilidad:** Validación de la creación, edición y eliminación de costos, y generación de reportes contables.



```
PS C:\Users\alde3\Documents\Workspace\FerreCloud-master> python manage.py test Configuracion.tests
Found 6 test(s).
Creating test database for alias 'default'...
System check identified no issues (0 silenced).
.test_crear_costo: OK
.test_edit_costo: OK
.test_eliminar_costo: OK
.test_crear_tipo: OK
.test_edit_tipo: OK
.test_eliminar_tipo: OK

-----
Ran 6 tests in 0.332s

OK
Destroying test database for alias 'default'...
```

Figura 5.19: Captura de las pruebas unitarias del módulo de contabilidad

Estas pruebas unitarias se ejecutaron utilizando herramientas de testing de Python y Django, como el módulo `unittest` y `pytest`.

4.5 Pruebas de Integración

Las pruebas de integración se enfocaron en el flujo de trabajo del usuario, probando la interacción entre los módulos:

- **Gestión de Ventas e Inventario:** Verificación de que cada venta registrada actualice correctamente el inventario.
- **Actualización de Cálculo Tributario:** Confirmación de que los valores de IVA y otros impuestos se calculen y actualicen automáticamente en los registros contables del sistema.

4.6 Pruebas Funcionales

Para cada funcionalidad principal del sistema, se crearon escenarios de prueba específicos:

- **Escenario de Registro de Venta:** Se evaluó el proceso de registro de una venta, verificando el cálculo de los totales, la aplicación de descuentos y el cálculo del IVA.
- **Escenario de Reabastecimiento de Inventario:** Simulación de la adición de productos al inventario, asegurando que el stock se actualice correctamente.
- **Generación de Reportes:** Validación de la generación de reportes de ventas, inventario y contabilidad, asegurando la precisión de los datos y su presentación.

4.7 Pruebas de Rendimiento

Se realizaron pruebas de estrés en el sistema para evaluar su rendimiento bajo una carga elevada. Para esto se utilizó Locust, un sistema basado en python para realizar pruebas de rendimiento y generar reportes completos. En primer lugar se seleccionó la cantidad de usuarios que iban a estar realizando solicitudes GET y POST concurrentemente, en el marco de FerreCloud se configuraron 10 debido a que con base a encuesta realizada, es el número máximo de empleados que se tienen en un establecimiento como el nicho objetivo. Los Resultados fueron los siguientes:

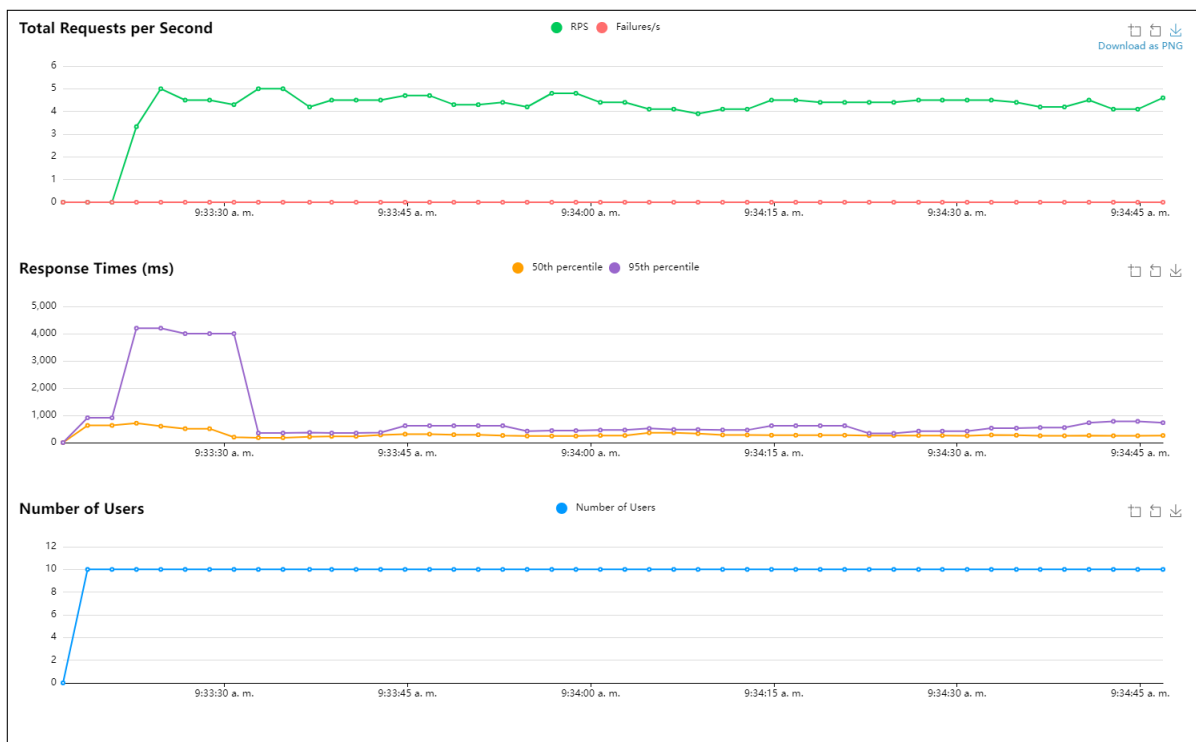


Figura 5.20: Gráficas de prueba de rendimiento de Locust

Cuando Locust entra en ejecución, se evidencian las gráficas de rendimiento, las cuales se evidencia una subida en el tiempo de respuesta al comienzo de la aplicación mas sin embargo no es preocupante debido a que esto representa el login y al renderizar el dashboard inicial, es normal el hecho de que el tiempo de respuesta de FerreCloud sea mas alto respecto al promedio

Type	Name	# Requests	# Fails	Median (ms)	95%ile (ms)	99%ile (ms)	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	Current RPS	Current Failures/s
GET	/	110	0	260	520	780	289.79	149	889	4140	1.4	0
GET	/accounts/login/	10	0	600	910	910	685.94	525	914	4140	0	0
POST	/accounts/login/	10	0	3900	4200	4200	3961.52	3766	4230	4140	0	0
GET	/impuestos/	88	0	270	600	700	298.92	157	703	4140	1.2	0
GET	/inventario/	90	0	250	590	710	288.61	158	706	4140	0.8	0
GET	/ventas/	91	0	250	630	730	297.06	146	730	4140	1.2	0
Aggregated		399	0	260	700	4000	395.15	146	4230	4140	4.6	0

Figura 5.21: Reporte de prueba de rendimiento de Locust

En el reporte se puede evidenciar pruebas de login y rendering de vistas de la aplicación. Se evidencia que ninguno de las solicitudes arrojó error, aprobando exitosamente la prueba.

4.8 Pruebas de usabilidad

Para las pruebas de usabilidad de FerreCloud, se realizaron evaluaciones con usuarios en un entorno real. Para ello, el sistema se implementó en un servidor local de la ferretería **Matering SAS**. Una vez puesto en producción, se solicitó a los colaboradores de la ferretería que utilizaran el sistema con el fin de obtener retroalimentación sobre su usabilidad.

4.8.1 Metodología de evaluación

El proceso de prueba siguió los principios del diseño centrado en el usuario, asegurando que las necesidades y expectativas de los colaboradores de la ferretería

fueran el eje central de la evaluación. Se realizaron las siguientes actividades:

- **Caracterización de los participantes:** Se seleccionaron colaboradores con distintos niveles de experiencia y características demográficas, con el fin de evaluar la usabilidad del sistema en un entorno representativo de los usuarios finales.
 - **Rango de edad:** Los participantes tenían edades comprendidas entre los 22 y 55 años, con una mayor concentración en el rango de 30 a 45 años. Esto permitió evaluar la facilidad de uso del sistema tanto para personas jóvenes con mayor familiaridad con la tecnología como para aquellas con menos experiencia en herramientas digitales.
 - **Nivel educativo:** El nivel de formación académica de los participantes variaba desde educación secundaria hasta formación técnica y universitaria en áreas relacionadas con administración, contabilidad e ingeniería. Esta diversidad permitió evaluar la accesibilidad del sistema para usuarios con distintos niveles de conocimientos técnicos.
 - **Experiencia en el uso de sistemas digitales:** Se incluyeron usuarios con distintos grados de familiaridad con software de gestión empresarial. Aproximadamente un 40 % tenía experiencia previa en el uso de otros sistemas de facturación e inventario, mientras que el 60 % restante no había trabajado antes con herramientas similares, lo que permitió evaluar la curva de aprendizaje de FerreCloud.
 - **Frecuencia de uso de herramientas tecnológicas:** Un 30 % de los participantes utilizaba computadoras o dispositivos digitales con frecuencia en su trabajo diario, mientras que el 70 % restante dependía más de

procesos manuales o tradicionales. Esto permitió evaluar qué tan intuitivo y accesible resultaba FerreCloud para usuarios menos familiarizados con la tecnología.

La variedad de perfiles seleccionados permitió obtener una visión integral sobre la usabilidad del sistema, identificando tanto sus fortalezas como las áreas que requerían ajustes para mejorar la experiencia del usuario.

Cada grupo de participantes brindó una perspectiva única sobre la usabilidad del sistema, permitiendo identificar necesidades específicas y oportunidades de mejora en la interfaz y el flujo de trabajo de FerreCloud.

- **Registro de interacciones:** Se monitoreó el uso del sistema por parte de los participantes, registrando el tiempo necesario para completar tareas específicas.

Usuario	Edad	Nivel educativo	Tiempo de ejecución	Observaciones
U1	25	Técnico en administración	16 minutos	Completa la tarea sin problemas, destaca la rapidez del sistema.

Continúa en la siguiente página

Usuario	Edad	Nivel educativo	Tiempo de ejecución	Observaciones
U2	38	Secundaria completa	9 minutos	Presenta dificultades iniciales con la navegación, pero logra adaptarse tras algunas indicaciones.
U3	42	Secundaria completa	15 minutos	Se adapta rápidamente y encuentra útil la organización del menú principal.
U4	30	Profesional en administración	14 minutos	Navega sin inconvenientes, pero sugiere mejoras en la presentación de reportes.

Continúa en la siguiente página

Usuario	Edad	Nivel educativo	Tiempo de ejecución	Observaciones
U5	50	Secundaria completa	12 minutos	Tarda más de lo esperado en encontrar algunas funciones, recomienda botones más visibles.
U6	28	Ingeniero industrial	17 minutos	Identifica correctamente las secciones, pero menciona que algunos términos pueden ser confusos.
U7	34	Secundaria completa	6 minutos	Considera que el sistema es intuitivo.

Continúa en la siguiente página

Usuario	Edad	Nivel educativo	Tiempo de ejecución	Observaciones
U8	45	Bachillerato incompleto	10 minutos	Requiere apoyo en algunos pasos, menciona que el sistema es funcional pero que necesita más práctica.

Tabla 5.6: Registro de interacciones de los usuarios durante las pruebas de usabilidad

- **Encuesta de satisfacción:** Al finalizar la prueba, los participantes respondieron una encuesta basada en la escala de Likert[34] para evaluar su experiencia con FerreCloud.

4.8.2 Diseño del instrumento de evaluación de la satisfacción del usuario:

Gracias por participar en la prueba de FerreCloud Tu opinión es muy importante para nosotros. Por favor, completa esta breve encuesta para ayudarnos a mejorar:

¿Qué tan fácil fue aprender a usar FerreCloud? *(1 = Muy difícil, 5 = Muy fácil)*

- 1
- 2

- 3

- 4

- 5

¿Qué tan rápido respondía el sistema a tus acciones? *(1 = Muy lento, 5 = Muy rápido)*

- 1

- 2

- 3

- 4

- 5

En caso de desear visualizar la totalidad de la encuesta, por favor dirigirse a la sección de Anexos

4.9 Resultados de la encuesta

¿Qué tan fácil fue aprender a usar FerreCloud? *(1 = Muy difícil, 5 = Muy fácil)*

 Copiar gráfico

5 respuestas

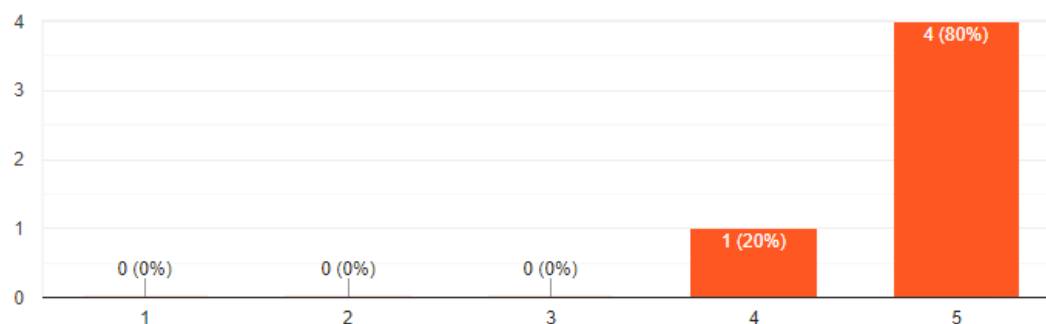


Figura 5.22: Respuesta de pregunta de encuesta de usabilidad

¿Qué módulo utilizaste más durante la prueba? *(Selecciona todas las que correspondan)*

 Copiar gráfico

5 respuestas

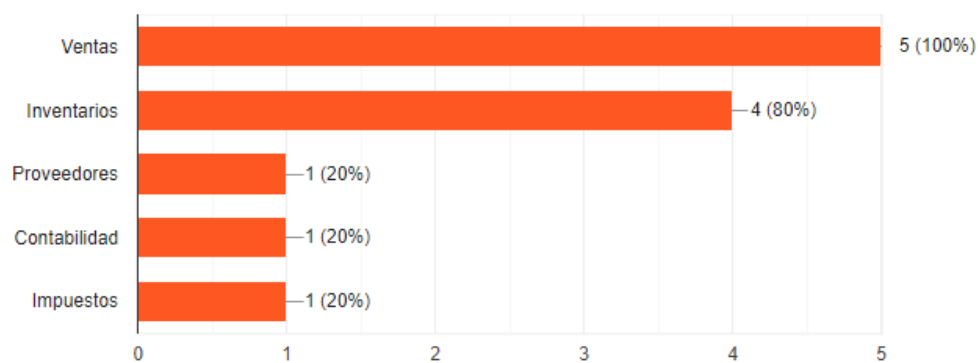


Figura 5.23: Respuesta de pregunta de encuesta de usabilidad

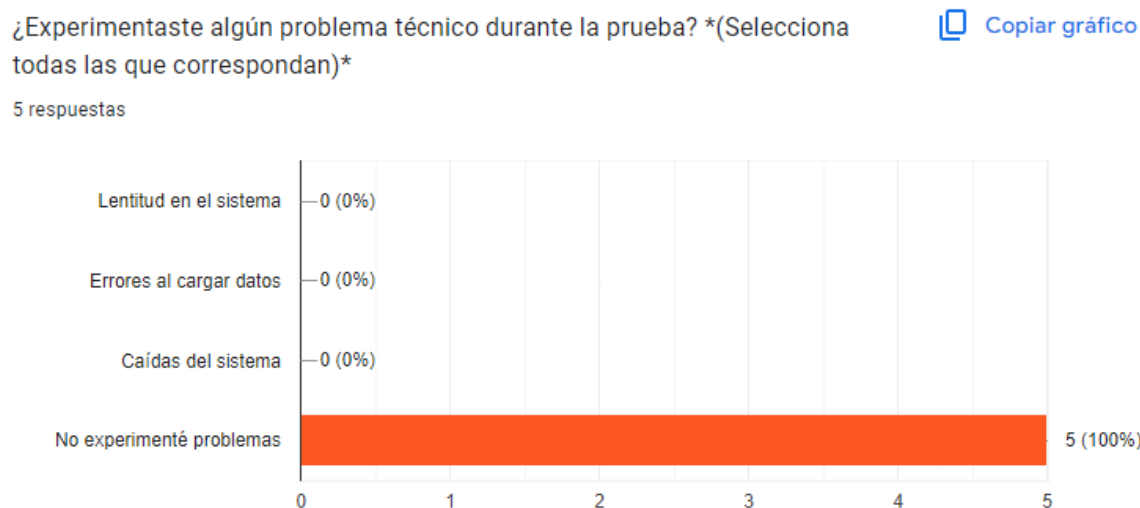


Figura 5.24: Respuesta de pregunta de encuesta de usabilidad

Para visualizar la totalidad de las respuestas, y las evidencias fotográficas de la implementación, visitar la sección de anexos.

En la encuesta se evidenció una percepción favorable en la usabilidad del sistema, destacándose aspectos clave como la comodidad al interactuar con las diferentes funcionalidades, la estabilidad general durante su uso y el flujo de trabajo intuitivo que facilita realizar tareas de manera eficiente. Esto sugiere que el diseño del sistema logra satisfacer las necesidades de los usuarios, brindándoles una experiencia práctica y sin interrupciones significativas.

4.9.1 Resultados de la evaluación de pruebas de usabilidad

Con base en los principios de usabilidad definidos por Jakob Nielsen[3], se analizaron los resultados de las pruebas realizadas en FerreCloud, obteniendo los

siguientes hallazgos:

- **Aprendizaje rápido:** Un 80 % de los participantes consideró que el sistema era fácil de aprender a usar, lo que indica que la interfaz facilita la familiarización de los nuevos usuarios con el software.
- **Eficiencia:** El módulo de ventas fue el más utilizado, seguido por el módulo de inventarios, lo que demuestra que los usuarios pueden completar sus tareas de manera efectiva. Sin embargo, se detectaron problemas de rendimiento en la carga de reportes, afectando la rapidez en la generación de informes.
- **Memorabilidad:** Aunque no se evaluó explícitamente en esta prueba, el alto porcentaje de usuarios que consideraron fácil el uso del sistema sugiere que las interacciones son intuitivas y de fácil recuerdo.
- **Errores mínimos:** El 100 % de los participantes logró completar sus tareas principales sin errores críticos, lo que indica que el sistema minimiza la posibilidad de fallos y permite a los usuarios alcanzar sus objetivos sin inconvenientes.
- **Satisfacción:** Un 80 % de los usuarios recomendaría FerreCloud a otras personas, lo que sugiere una experiencia de uso positiva.

4.10 Resultados y Observaciones

Los resultados de las pruebas indicaron que el sistema cumple con los requisitos funcionales y no presenta errores críticos que puedan afectar su operación en un entorno real. Se detectaron algunas áreas de mejora, como la optimización de

consultas a la base de datos en el módulo de reportes, que fueron ajustadas antes de la fase de implementación.

4.11 Auditorías de calidad

Las auditorías de calidad son herramientas esenciales para garantizar que el desarrollo de FerreCloud cumpla con los estándares de calidad establecidos. Estas auditorías permiten identificar y mitigar riesgos, asegurar el cumplimiento de los requisitos y fomentar una mejora continua en los procesos y entregables del proyecto.

4.11.1 *Proposito de las auditorías*

- Evaluar la adherencia a los estándares y procedimientos definidos en el Plan de Aseguramiento de Calidad (PAC).
- Detectar y registrar desviaciones o no conformidades en los procesos y productos.
- Asegurar la calidad del producto final antes de su entrega.
- Proveer información objetiva y verificable para la toma de decisiones.

4.11.2 *Proceso de auditorías*

Para el marco de este proyecto seguimos el estandar de auditoría de producto, lo cual es un proceso sistemático que evalúa la calidad, conformidad y desempeño de un producto respecto a los estándares, especificaciones o requerimientos establecidos. Este tipo de auditoría se centra exclusivamente en el producto final, y su objetivo principal

es garantizar que cumple con los requisitos del cliente, normativas legales y estándares de calidad.

En primer lugar, se define del alcance, es decir, los aspectos del software que serán auditados, como las funcionalidades específicas, los módulos del sistema o el código fuente. También se establecen los criterios de evaluación, que pueden incluir normativas de desarrollo, buenas prácticas de programación, estándares de seguridad o requisitos específicos del cliente. Además, se asignan roles al equipo siendo el director el auditor y el estudiante el responsable auditado, garantizando que cuenten con las competencias necesarias para realizar la revisión de manera objetiva.

Durante la inspección visual del software, el director evalúa la interfaz de usuario (UI) y la experiencia de usuario (UX). Se busca verificar que la UI sea intuitiva, accesible y que cumpla con los estándares de diseño establecidos. También se revisa que el software tenga una documentación clara, con manuales de usuario actualizados y accesibles, y que los mensajes de error sean apropiados y fáciles de entender.

Por último la verificación funcional se evalúa que el software cumpla con las funcionalidades especificadas en los requisitos. Esto incluye la realización de pruebas para garantizar que el sistema realice las operaciones previstas.

Una vez realizadas todas las pruebas y verificaciones, se documentan los hallazgos de la auditoría, señalando cualquier no conformidad o área de mejora detectada. Los informes detallados incluyen los resultados de las pruebas, las vulnerabilidades de seguridad encontradas, las discrepancias con los requisitos o las buenas prácticas, y cualquier recomendación de mejora e ingresandolas al tablero Kanban.

A partir de los resultados obtenidos, se proponen acciones correctivas. Si se encuentran errores o vulnerabilidades, se sugiere su corrección inmediata, priorizando aquellos que tengan un impacto más significativo en el funcionamiento o la seguridad del software. También se pueden recomendar mejoras en la documentación, el diseño de la interfaz o la experiencia de usuario.

Durante el proyecto hubieron 4 auditorías, cada una de ellas mas enfocada en cada uno de los apartados principales de la misma, las cuales se presentan a continuación:

Entregable	Criterio Evaluado	Herramienta	Fecha
Código Fuente	Estándares de codificación	VS Code	15/09/2024
Interfaz de Usuario	Usabilidad y Accesibilidad	Pruebas manuales	18/10/2024
Pruebas Funcionales	Resolución de casos de uso	Pruebas unitarias y Locust	08/11/2024
Documentación Técnica	Claridad y completitud	Revisión Manual	25/11/2024

Informe Detallado de Auditorías Realizadas

1. Auditoría de Código Fuente

- **Fecha:** 15/09/2024
- **Entregable:** Código Fuente
- **Criterio Evaluado:** Estándares de codificación
- **Herramienta Utilizada:** VS Code

Objetivo: Evaluar si el código fuente sigue los estándares de codificación establecidos para garantizar su calidad, mantenibilidad y coherencia.

Hallazgos:

- El código sigue en general las convenciones de nombres y estructura recomendadas, aunque se encontraron algunas inconsistencias menores en el estilo de la indentación, lo que puede afectar la legibilidad.
- Se detectaron algunas funciones con un tamaño mayor al recomendado, lo que podría dificultar la comprensión y mantenimiento del código a largo plazo.
- Se observaron partes del código que podrían beneficiarse de la refactorización para mejorar su eficiencia y modularidad.

Recomendaciones:

- Ajustar la indentación de algunas líneas de código para cumplir con el estándar establecido.
- Refactorizar las funciones largas y complejas, dividiéndolas en subfunciones más pequeñas.
- Continuar con el uso de herramientas de análisis estático de código para mantener la calidad del código.

2. Auditoría de Interfaz de Usuario

- **Fecha:** 18/10/2024
- **Entregable:** Interfaz de Usuario
- **Criterio Evaluado:** Usabilidad y Accesibilidad
- **Herramienta Utilizada:** Pruebas manuales

Objetivo: Evaluar si la interfaz de usuario cumple con los estándares de usabilidad y accesibilidad establecidos, asegurando que sea intuitiva y fácil de usar para los usuarios finales.

Hallazgos:

- La interfaz es en su mayoría intuitiva y fácil de navegar, pero algunos botones y menús podrían ser más visibles o tener un contraste mayor para mejorar su accesibilidad.
- Se identificaron problemas de accesibilidad para personas con discapacidad visual, como el uso de combinaciones de colores que dificultan la lectura.
- Algunos elementos de la UI, como las notificaciones y los mensajes de error, no son lo suficientemente claros o visibles.

Recomendaciones:

- Mejorar el contraste en los botones y menús para facilitar la lectura, especialmente para personas con discapacidad visual.
- Utilizar colores más accesibles y asegurarse de que el diseño cumpla con las pautas WCAG (Web Content Accessibility Guidelines).
- Asegurarse de que todos los mensajes de error sean descriptivos y fáciles de entender, con instrucciones claras sobre cómo corregir el problema.

3. Auditoría de Pruebas Funcionales

- **Fecha:** 08/11/2024
- **Entregable:** Pruebas Funcionales
- **Criterio Evaluado:** Resolución de casos de uso
- **Herramienta Utilizada:** Pruebas unitarias y Locust

Objetivo: Evaluar si el sistema cumple con los casos de uso especificados y si las pruebas funcionales garantizan que el software realice las operaciones previstas.

Hallazgos:

- La mayoría de los casos de uso se ejecutaron correctamente, pero se encontraron algunas fallas en la validación de datos en los formularios, lo que permite entradas incorrectas en el sistema.

- En las pruebas de rendimiento realizadas con Locust, se detectó que el sistema no soporta un gran volumen de usuarios simultáneos sin degradarse significativamente en términos de respuesta.

Recomendaciones:

- Corregir las validaciones en los formularios para evitar que se ingresen datos incorrectos.
- Mejorar el rendimiento mediante optimizaciones en las consultas a la base de datos y la implementación de técnicas de escalabilidad, como el uso de cachés o balanceo de carga.

4. Auditoría de Documentación Técnica

- **Fecha:** 25/11/2024
- **Entregable:** Documentación Técnica
- **Criterio Evaluado:** Claridad y completitud
- **Herramienta Utilizada:** Revisión Manual

Objetivo: Evaluar si la documentación técnica está clara, completa y es accesible para los usuarios y desarrolladores que la necesiten.

Hallazgos:

- La documentación técnica está bien organizada, pero algunos módulos importantes del sistema no están suficientemente documentados.
- El manual de usuario es claro, pero carece de ejemplos prácticos que ayudarían a los usuarios a entender mejor las funcionalidades del sistema.
- La documentación de la API no está actualizada con los cambios más recientes realizados en el código.

Recomendaciones:

- Completar la documentación de los módulos faltantes para asegurar que todos los aspectos del sistema estén debidamente documentados.
- Incluir más ejemplos prácticos en el manual de usuario para facilitar la comprensión.
- Actualizar la documentación de la API de acuerdo con los últimos cambios realizados en el sistema.

Resultados

1 Resultados obtenidos

1.1 Desarrollo Integral del Sistema de Información

El *Sistema de Información de Ferreterías bajo el Régimen Simple de Tributación* fue desarrollado como una plataforma robusta y escalable que optimiza la gestión operativa de las ferreterías, adaptándose a las normativas fiscales del régimen simple. La estructura modular del sistema permite la incorporación de futuras mejoras y la personalización según las necesidades específicas de cada ferretería. A lo largo del desarrollo, se aplicaron metodologías ágiles para asegurar una entrega constante de funcionalidades, manteniendo un ciclo iterativo de pruebas y retroalimentación.

1.2 Seguridad y Control de Acceso

En términos de seguridad, se implementó un sistema de autenticación de usuarios basado en roles, que garantiza que cada usuario tenga acceso únicamente a las funcionalidades que le corresponden según su perfil. Se definieron distintos niveles de

acceso para administradores, vendedores y personal de inventario, evitando accesos no autorizados a información crítica. Además, se aplicaron medidas de seguridad como el cifrado de contraseñas de Django y la validación de sesiones, garantizando la protección de los datos.

1.3 Pruebas de Usabilidad y Validación con Usuarios Finales

Se llevaron a cabo pruebas de usabilidad con usuarios de distintas ferreterías para validar la efectividad y la experiencia de usuario del sistema. Estas pruebas incluyeron la realización de tareas específicas dentro del sistema, como la gestión de ventas, el registro de clientes, la actualización de inventarios y la emisión de reportes. Los resultados obtenidos indicaron un alto nivel de aceptación y satisfacción, destacándose la facilidad de uso y la claridad en la interfaz de usuario. Además, los comentarios y sugerencias de los usuarios fueron integrados en las iteraciones finales del desarrollo, logrando mejoras en el rendimiento y la accesibilidad.

1.4 Validación Técnica y Cumplimiento de Requisitos

Finalmente, se llevó a cabo un proceso exhaustivo de validación técnica para garantizar que el sistema cumpla con los requisitos establecidos en el proyecto. Esta validación incluyó pruebas de carga y estrés para verificar la estabilidad del sistema en escenarios de alta demanda, pruebas de integración para asegurar la correcta comunicación entre los distintos módulos, y pruebas funcionales para validar cada uno de los casos de uso definidos. Se concluyó que el sistema es capaz de gestionar de manera eficiente las operaciones de una ferretería y que cumple con las normativas del

Régimen Simple de Tributación.

1.5 Módulo de Ventas y Gestión de Clientes

En el módulo de ventas, se implementaron mecanismos de registro ágil y preciso de transacciones comerciales, permitiendo al usuario registrar ventas de manera rápida y evitar errores mediante la validación de datos en tiempo real. La función de cálculo automático del IVA, configurada de acuerdo con el régimen tributario vigente, ofrece un registro transparente de las obligaciones fiscales. Además, se añadieron funcionalidades para aplicar descuentos, gestionar métodos de pago y emitir comprobantes de venta. La gestión de clientes permite registrar nuevos clientes, actualizar su información y visualizar un historial de compras, lo que facilita la fidelización y mejora la atención al cliente.

1.6 Optimización del Módulo de Inventarios

El módulo de inventarios se diseñó para gestionar eficientemente los productos, con una interfaz intuitiva y accesible. Los usuarios pueden registrar productos nuevos, actualizar sus cantidades y recibir alertas cuando un producto alcance un nivel mínimo definido. Esta funcionalidad se complementó con un sistema de categorización que permite organizar los productos por tipo, facilitando su búsqueda y la generación de reportes detallados. Además, se integró un historial de movimientos de inventario que registra las entradas y salidas de productos, lo cual ofrece una visión clara de la disponibilidad en tiempo real.

1.7 Gestión de Proveedores y Reabastecimiento

Se desarrolló un módulo de proveedores que permite registrar y gestionar la información de cada proveedor, incluyendo datos de contacto, condiciones de pago y productos suministrados. Este módulo se conecta directamente con la gestión de inventarios, facilitando la emisión de pedidos de reposición basados en umbrales configurables por el administrador. Además, se añadió la funcionalidad de generar órdenes de compra y mantener un registro de los pedidos realizados, lo cual permite un mejor control de las relaciones comerciales y la planificación de compras.

1.8 Módulo de Costos y Administración de Gastos

La incorporación de un módulo de costos facilita la administración y registro de gastos asociados a la operación de la ferretería, como costos de logística, mantenimiento, y servicios generales. El sistema permite ingresar detalles específicos sobre cada costo, así como su categorización, lo cual facilita la identificación de las áreas de mayor inversión y permite optimizar los recursos. Además, este módulo permite visualizar los costos en un rango de tiempo determinado y realizar comparaciones de periodos, generando información clave para la toma de decisiones.

2 Trabajos futuros

A lo largo del desarrollo del *Sistema de Información de Ferreterías bajo el Régimen Simple de Tributación*, se han identificado varias oportunidades para ampliar y mejorar el sistema en futuras iteraciones. A continuación, se destacan algunas líneas de trabajo que podrían abordarse para enriquecer y fortalecer la plataforma:

2.1 Integración de Facturación Electrónica

Una de las mejoras clave es la implementación de un módulo de facturación electrónica, alineado con los requerimientos legales en Colombia. Este módulo permitiría a las ferreterías emitir facturas electrónicas, optimizando el cumplimiento fiscal y reduciendo el uso de documentos físicos, lo cual contribuiría a una gestión administrativa más eficiente.

2.2 Desarrollo de un Módulo de Análisis Predictivo

Se propone la incorporación de herramientas de análisis predictivo mediante técnicas de *machine learning*. Estas funcionalidades permitirían generar pronósticos de ventas y estimaciones de demanda de productos, facilitando la planificación de inventarios y la toma de decisiones estratégicas basadas en datos históricos y tendencias de mercado.

2.3 Expansión de Funcionalidades de Reportes

Si bien el sistema actual cuenta con funcionalidades básicas de generación de reportes, se recomienda el desarrollo de un módulo avanzado de reportes que permita la personalización de parámetros y la exportación de datos en múltiples formatos. Esto proporcionaría a los usuarios una mayor flexibilidad y profundidad en el análisis de la información operativa y financiera.

2.4 Desarrollo de una Aplicación Móvil

Otra línea de trabajo futura es la creación de una aplicación móvil que permita a los usuarios acceder al sistema desde dispositivos móviles. La aplicación podría incluir funcionalidades clave como la gestión de inventarios, el registro de ventas y la visualización de reportes en tiempo real, mejorando la accesibilidad y la eficiencia en la operación diaria.

2.5 Implementación de un Módulo de Fidelización de Clientes

En aras de mejorar la relación con los clientes, se sugiere el desarrollo de un módulo de fidelización que permita registrar y gestionar programas de beneficios, tales como puntos acumulables o descuentos exclusivos para clientes frecuentes. Esta funcionalidad contribuiría a fomentar la lealtad de los clientes y a aumentar la recurrencia de compras.

2.6 Optimización del Rendimiento y Seguridad del Sistema

Finalmente, se continua trabajando en la optimización del rendimiento del sistema y en la mejora de las medidas de seguridad. Esto incluiría la implementación de técnicas avanzadas de control de acceso, la protección de datos sensibles y la optimización de la velocidad de carga y respuesta del sistema en escenarios de alta demanda.

3 Dificultades

Durante el desarrollo del **Sistema de Información de Ferreterías bajo Régimen Simple de Tributación**, se presentaron varias dificultades técnicas y operativas que afectaron tanto la implementación como la integración de distintas funcionalidades del sistema. Algunas de las principales dificultades fueron las siguientes:

3.1 Adaptación al Régimen Simple de Tributación

Una de las mayores dificultades fue adaptar el sistema a las particularidades del **Régimen Simple de Tributación** que rige para las ferreterías en Colombia. Este régimen establece normas específicas de cálculo de impuestos, retenciones, y registros contables que debían ser correctamente implementadas en el sistema. La complejidad radica en la necesidad de realizar cálculos dinámicos según las ventas y costos, lo que requería un entendimiento profundo de la normativa tributaria y su correcta integración en los módulos de ventas y contabilidad.

3.2 Implementación de Reportes de Ventas y Costos

La creación de reportes detallados sobre ventas y costos fue una tarea desafiante, especialmente por la necesidad de presentar la información en tiempo real y en un formato claro y comprensible para los usuarios. En algunos casos, los cálculos de IVA no se realizaban correctamente debido a los cambios en las tasas impositivas aplicadas a diferentes tipos de productos. Esto requirió ajustar los algoritmos de cálculo para

asegurar que los informes fueran precisos y cumplieran con la normativa vigente.

3.3 Manejo de Inventario y Restock de Productos

El sistema de inventario se vio afectado por la necesidad de gestionar múltiples productos en diversas ubicaciones dentro de la ferretería, lo cual añadió un nivel extra de complejidad en la gestión de stock. Además, la implementación de la funcionalidad para el **restock de productos** a través de proveedores presentó dificultades por la falta de información estandarizada sobre los productos. Fue necesario desarrollar una lógica compleja para automatizar las órdenes de reabastecimiento sin generar errores en las cantidades o tipos de productos solicitados.

3.4 Desafíos con la Interfaz de Usuario

Aunque el sistema fue diseñado para ser intuitivo, algunos usuarios experimentaron dificultades al navegar por la interfaz, especialmente aquellos sin experiencia previa en el manejo de sistemas informáticos. Esto se debió a la complejidad inherente de las funcionalidades requeridas por el sistema, como el manejo de facturación electrónica, la creación de ventas, y la generación de reportes fiscales. Se realizaron varias iteraciones de prueba con usuarios finales para optimizar la experiencia de usuario.

3.5 Pruebas de Desempeño y Escalabilidad

A medida que el sistema se fue desarrollando y probando en diferentes escenarios, surgieron dificultades relacionadas con el **rendimiento** y la **escalabilidad** del software.

Especialmente en las tareas que involucraban grandes volúmenes de datos, como la generación de reportes de ventas anuales o la consulta de inventarios a gran escala, el sistema presentó tiempos de respuesta elevados. Esto obligó a optimizar las consultas a la base de datos y a implementar técnicas de **caching** para mejorar la velocidad de procesamiento.

4 Comparativa entre los resultados obtenidos y los objetivos planteados

En el proyecto "Sistema de Información para Ferreterías bajo Régimen Simple de Tributación", los resultados obtenidos mostraron un avance significativo hacia el cumplimiento de los objetivos planteados. Aunque se presentaron ciertas dificultades a lo largo del desarrollo, estas fueron abordadas de manera proactiva, lo que permitió superar los retos y garantizar un producto funcional y de valor. A continuación, se describe cómo se alcanzaron los objetivos, enfrentando y resolviendo las problemáticas encontradas.

Requerimientos funcionales y no funcionales El levantamiento de los requerimientos funcionales y no funcionales fue un paso crucial en el proyecto. Aunque inicialmente se encontró que algunos requerimientos no funcionales, como la escalabilidad, no estaban completamente definidos, esto se resolvió mediante la implementación de sesiones adicionales de recopilación de información con los stakeholders y consultas con expertos en desarrollo de sistemas. Este enfoque permitió identificar áreas críticas y ajustar el diseño del sistema para satisfacer mejor las necesidades del negocio.

Comodidad y productividad del usuario El objetivo de garantizar una experiencia cómoda para los usuarios, que además incrementara su productividad, se logró al integrar principios de diseño centrado en el usuario. Las dificultades iniciales con la disposición de algunos componentes en la interfaz se resolvieron mediante pruebas iterativas de usabilidad y retroalimentación constante de los usuarios finales. Estas pruebas llevaron a rediseñar elementos clave, como los menús y formularios, resultando en una experiencia de usuario más fluida y eficiente.

Funcionalidad y valor para la ferretería El sistema cumplió su propósito de aportar valor a la ferretería al automatizar procesos manuales y simplificar tareas administrativas. Aunque inicialmente hubo limitaciones en el desarrollo de ciertos módulos avanzados, estas se superaron mediante la priorización estratégica de funcionalidades esenciales y la planificación de futuras iteraciones para incluir características adicionales. Además, la incorporación de reportes básicos, pero efectivos, permitió a los usuarios apreciar el valor inmediato del sistema.

Perfeccionamiento del software y prevención de errores La mejora continua del software fue un enfoque constante durante el desarrollo. Si bien se identificaron algunos errores menores en escenarios específicos, se implementaron ciclos adicionales de pruebas con usuarios reales y herramientas automatizadas para garantizar una mayor estabilidad. Este proceso no solo ayudó a prevenir errores futuros, sino que también mejoró la confianza de los usuarios en el sistema. Además, se estableció un plan de mantenimiento que incluyó la corrección de errores reportados tras el despliegue inicial.

Uso cómodo del software por parte del usuario El objetivo de facilitar un uso cómodo del software fue ampliamente logrado. La incorporación de tutoriales interactivos, una guía de usuario integrada y un diseño claro permitió a los usuarios adaptarse rápidamente al sistema. Las dificultades relacionadas con la curva de aprendizaje inicial se superaron mediante sesiones de capacitación personalizada, lo que no solo mejoró la experiencia del usuario, sino que también aumentó su confianza y satisfacción al utilizar la solución.

A continuación se presenta una tabla comparativa con los objetivos esperados vs los

objetivos obtenidos:

Objetivo esperado	Resultado obtenido
Levantar unos requerimientos funcionales y no funcionales para su próximo desarrollo.	Se levantaron requerimientos funcionales y no funcionales con base a las necesidades esenciales de las ferreterías actuales teniendo en cuenta los principios de que sea escalable, funcional, estable y usable.
Hacer que el software que se va a desarrollar próximamente sea cómodo para el usuario, realizando una en de manera que se pueda aumentar la productividad del usuario mientras usa la solución.	El software desarrollado demostró las encuestas realizadas reflejaron un alto nivel de satisfacción, destacando la comodidad y facilidad de uso de la solución cumpliendo con el objetivo planteado de ofrecer una herramienta práctica y funcional.
Lograr que el software que desarrollemos sea funcional y de valor a la ferretería.	Se logró que el software desarrollado diera valor al establecimiento que lo implementa de forma que este aumenta la productividad en la atención al cliente teniendo un flujo de trabajo preciso y rápido.
Perfeccionar el software desarrollado para evitar errores a la hora de que el usuario utilice la aplicación identificando aspectos de mejora e implementarlos.	Se llevaron a cabo pruebas exhaustivas en todas las funcionalidades del software para asegurar su correcta operación y minimizar al máximo la presencia de errores. Estas pruebas incluyeron tanto escenarios comunes de uso como casos límite.
Lograr que el usuario utilice el software de manera cómoda	Se Desarrolló un software enfocado al usuario, con metáforas que permitan su intuitividad, de forma que el usuario no necesite una capacitación exhaustiva para operar el software en cuestión

Tabla 6.1: Resultados esperados de cada objetivo. Fuente: Elaboración propia.

Conclusiones

El desarrollo del Sistema de Información para Ferreterías bajo el Régimen Simple de Tributación fue un ejercicio riguroso de ingeniería de software, que abarcó desde la planificación inicial hasta la validación final. Este proceso se fundamentó en metodologías estructuradas que garantizaron un diseño robusto, funcional y adaptable a las necesidades específicas del sector ferretero y las exigencias normativas del régimen tributario simplificado.

Durante la fase de análisis, se realizaron entrevistas y levantamientos de requisitos con los principales actores del sector, lo que permitió identificar los procesos críticos de las ferreterías y las problemáticas a resolver. A partir de esta información, se definieron las funcionalidades del sistema y se creó un modelo conceptual que sirvió de base para la arquitectura.

En el diseño, se aplicaron patrones de ingeniería como la arquitectura Cliente-Servidor, lo que permitió mantener una separación clara entre la lógica de negocio, la interfaz de usuario y el manejo de datos. Este enfoque modular facilitó la escalabilidad del sistema y su mantenimiento futuro. La base de datos fue cuidadosamente estructurada, optimizando las relaciones y consultas para garantizar

un alto rendimiento, incluso con grandes volúmenes de datos.

La fase de desarrollo utilizó herramientas y frameworks modernos, lo que no solo aceleró la construcción del sistema, sino que también aseguró su compatibilidad con tecnologías actuales. Se desarrollaron módulos específicos para el manejo de inventarios, ventas y registros contables, integrando cálculos automáticos para impuestos, descuentos y totales. Además, se implementaron medidas de seguridad, como la encriptación de datos sensibles y controles de acceso basados en roles.

El proceso de pruebas fue uno de los pilares del proyecto. Se realizaron pruebas unitarias para validar la funcionalidad individual de cada componente, garantizando que cada módulo cumpliera con su propósito sin errores. Posteriormente, se ejecutaron pruebas de integración para evaluar la interacción entre los diferentes módulos y asegurar que funcionaran de manera coherente. Finalmente, se llevaron a cabo pruebas con usuarios finales, quienes simulaban escenarios reales de uso, identificando posibles puntos de mejora en la interfaz y en los flujos de trabajo.

Estas pruebas exhaustivas no solo permitieron identificar y corregir errores, sino que también contribuyeron a validar la capacidad del sistema para manejar distintas condiciones de uso, desde operaciones sencillas hasta escenarios más complejos. Además, se documentaron todos los casos de prueba y sus resultados, dejando un registro detallado que facilita el mantenimiento y la evolución del sistema.

En conclusión, el desarrollo del sistema es un ejemplo de cómo aplicar principios de ingeniería de software y procesos de validación rigurosos puede garantizar la calidad de una solución tecnológica. Este enfoque metódico permitió construir una herramienta

confiable, eficiente y alineada con las necesidades del sector ferretero, sentando las bases para su crecimiento y adaptación futura.

Bibliografía

1 Referencias

[1] Mundo Ferretero, "La ferretería del futuro: Digitalización, innovación y tradición: La mezcla perfecta," Mundo Ferretero, 2024.
<https://www.mundoferretero.com.mx/index.php/ferreteria/item/1222-la-ferreteria-del-futuro-digitalizacion-innovacion-y-tradicion-la-mezcla-perfecta>.

[2] L. von Bertalanffy, General System Theory: Foundations, Development, Applications, rev. ed. New York, NY, USA: George Braziller, 1968.

[3] J. Nielsen, Usability Engineering. Boston, MA, USA: Academic Press, 1993.

[4] A. Silberschatz, H. F. Korth, y S. Sudarshan, Database System Concepts, 7^a ed. Nueva York: McGraw-Hill, 2020.

[5] R. Elmasri y S. B. Navathe, Fundamentals of Database Systems, 7^a ed. Boston: Pearson, 2015.

[6] S. Lozada, "Implementación de un sistema de control de almacén en la ferretería Tomasini - Piura; 2020," Tesis de Ingeniería de Sistemas, Universidad Católica Los Ángeles de Chimbote, Chimbote, Perú, 2022. [En línea]. Disponible en: <https://repositorio.uladech.edu.pe/handle/20.500.13032/26286>

[7] H. A. Hernández, Y. L. Cruz Gil, M. D. Puentes Saavedra, y D. E. Mendoza Patiño, "Diseño de un sistema de gestión de inventarios para el almacén TÉCNITALLER S.A.S de la ciudad Neiva-Huila, Colombia," 2021. [En línea]. Disponible en: <https://portal.amelica.org/ameli/journal/517/5172288015/html/>.

[8] A. F. Rojas Pinzón, "Gestión administrativa y contable para las pequeñas y medianas empresas del sector de transporte de carga por carretera," Tesis de Ingeniería de Sistemas, Universidad de los Andes, Bogotá, Colombia, 2021. [En línea]. Disponible en: <https://repositorio.uniandes.edu.co/entities/publication/7f9d4687-5c9b-47a0-bea1-d5e97692c7b7>.

[9] M. I. Díaz Vega y M. J. Ospina Ospina, "Conceptos económicos de innovación en la valoración tecnológica de MiPymes colombianas desarrolladoras de software," Revista Virtual del Sena, vol. 87, no. 2, 2023. [En línea]. Disponible en: https://revistas.sena.edu.co/index.php/inf_tec/article/view/conceptos-economicos-de-innovacion-en-la-valoracion-tecnologica.

[10] J. Sánchez Pimiento y L. Arias Casalla, "Desarrolladores de software en tiempos de pandemia," Caoba: Revista de Sistemas, 2024. [En línea]. Disponible en: <https://caoba.sanmateo.edu.co/ojs/index.php/sistemas/article/view/205>.

- [11] ManagerERP - QualityColombia <https://qualitycolombia.com/manager-erp/>
- [12] Siigo - Sistema administrativo y contable <https://www.siigo.com/>
- [13] K. C. Laudon and J. P. Laudon, Sistemas de información gerenciales, 15^a ed. Pearson, 2019.
- [14] Dirección de Impuestos y Aduanas Nacionales (DIAN), Régimen simple de tributación,"Micrositios DIAN. [Enlace] <https://micrositios.dian.gov.co/regimen-simple-tributacion/>
- [15] Zabala, R. M., Granja, L. G., Calderón, H. A., & Velasteguí, L. E. (2021). Efecto en la gestión organizacional y la satisfacción de los usuarios de un sistema informático de planificación de recursos empresariales (ERP) en Riobamba, Ecuador. Información Tecnológica. <https://doi.org/10.4067/s0718-07642021000500101>
- [16] A. T. Chan, Gestión de inventarios y control de existencias en empresas, 2^a ed. McGraw-Hill, 2018.
- [17] Huarcaya Zapana, G. I., Herencia Castro, N., Sarmiento Tico, M. A., Chalco Choquehuanca, E. D., & Ticona Bejarano, A. D. (2021). Aplicación del método Design Thinking en el área de requerimientos de software. Innovación y Software. <https://doi.org/10.48168/innosoft.s5.a37>
- [18] Pérez, J., Agustín, M.-G., & Ledesma, S.-T. (2022). Integración de sistemas. Fundació Universitat Oberta de Catalunya.

[19] P. Salas, J. y Tinoco, O. (2010). Criterios de selección de metodologías de Rosales.

[20] K. Beck, Extreme Programming Explained: Embrace Change, 2nd ed., Addison-Wesley, 2004.

[21] A. Cockburn, Agile Software Development: The Cooperative Game, 2nd ed., Addison-Wesley, 2006.

[22] R. C. Martin, Clean Code: A Handbook of Agile Software Craftsmanship, Prentice Hall, 2008.

[23] Python Software Foundation, "What is Python?", Python.org. <https://www.python.org>.

[24] Código Vanguardia, "Django: Ventajas y características que todo desarrollador debe conocer,"2022. [En línea]. Disponible en: <https://codigovanguardia.com/desarrollo-web/django-ventajas-y-caracteristicas-que-todo-desarrollador-debe-conocer/>.

[25] Django Software Foundation, Django Documentation. Disponible: <https://docs.djangoproject.com/en/5.1/>.

[26] The PostgreSQL Global Development Group, PostgreSQL Documentation: Transactions. [Online]. Available: <https://www.postgresql.org/docs/current/tutorial-transactions.html>.

[27] G. Booch, J. Rumbaugh, and I. Jacobson, The Unified Modeling Language User Guide, 2nd ed. Boston, MA: Addison-Wesley, 2005.

[28] A. Navarro, "Métricas de calidad del software," Blog Innevo, [En línea]. Disponible en: <https://blog.innevo.com/metricas-de-calidad-del-software#:~:text=1.,puede%20indicar%20problemas%20de%20calidad>.

[29] Dirección General de Protección Civil y Emergencias, "Métodos cuantitativos de análisis de riesgos", Protección Civil y Emergencias. [Online]. Available: https://www.proteccioncivil.es/catalogo/carpeta02/carpeta22/guiatec/Metodos_cuantitativos/cuant_232.htm

[30] IEEE/ISO/IEC 12207-2017. [Online]. Available: <https://standards.ieee.org/ieee/12207/5672/>

[31] Mosquera Arboleda, Nelson Enrique, Sistema de información Web. Available: <https://hdl.handle.net/11059/1325>

[32] Asana, "La matriz de Eisenhower: Prioriza tus tareas," *Asana*. <https://asana.com/es/resources/eisenhower-matrix>

[33] Asana, "What is Kanban?," *Asana*. <https://asana.com/resources/what-is-kanban>

[34] D. da Silva, "Qué es escala de Likert y cómo aplicarla", Zendesk, 14 de septiembre de 2024. [En línea]. Disponible en:

<https://www.zendesk.com.mx/blog/que-es-escala-de-likert/>.

2 Encuesta del trabajo de campo

- ¿Cuales son los desafíos principales a los que se enfrenta el establecimiento a nivel administrativo?
- ¿Cuales son los desafíos principales a los que se enfrenta el establecimiento a nivel operativo?
- ¿Cuántos colaboradores tiene el establecimiento?
- ¿Cuales son los ingresos mensuales estimados de su establecimiento?
- ¿Cual es la mayor dificultad que tiene relacionado al régimen simple de tributación?

2.1 Checklist para la Validación de Entregas de Requerimientos

Para garantizar que las entregas parciales del sistema cumplieran con los requerimientos establecidos, se utilizó la siguiente checklist de validación:

■ Revisión de Requerimientos Funcionales:

- Todos los requerimientos funcionales fueron implementados correctamente.
- Las funcionalidades entregadas cumplen con los criterios de aceptación definidos.
- Las funcionalidades fueron probadas en un entorno controlado por el cliente.

■ Revisión de Requerimientos No Funcionales:

- El sistema muestra los niveles adecuados de rendimiento (tiempos de respuesta, uso de recursos).
- Se implementaron medidas de seguridad necesarias para proteger los datos.
- La interfaz del usuario es intuitiva y fácil de navegar.

■ Revisión de Feedback del Cliente:

- Se recopilaron comentarios del cliente sobre las funcionalidades entregadas.
- Los comentarios fueron analizados y, si es necesario, se realizaron ajustes en las funcionalidades.

Anexos

1 Requerimientos Funcionales

Tabla 9.1: REQUERIMIENTO 4

Nombre	Eliminar producto de inventario
Propósito	Eliminar el producto de inventario cuando este no se va a ofrecer mas en el establecimiento
Descripción	Dentro de la lista de los productos de inventario, en cada uno de los items se muestra un botón en forma de X que abre una ventana de confirmación que al ser aceptada hace que desaparezca el elemento de la lista, haciendo que este producto no aparezca mas en inventario
Entrada	Producto seleccionado
Salida	Producto eliminado de inventario exitosamente

Tabla 9.2: REQUERIMIENTO 5

Nombre	Buscar un producto en inventario
Propósito	Cuando los usuarios necesitan buscar rápidamente un producto dentro de las existencias del inventario, se escribe el producto a buscar y de esta manera aparece en la lista de búsqueda
Descripción	En el modulo de inventario se instancia un finder en la parte superior de este en el cual se puede escribir la palabra clave y de la misma manera se hace la consulta en la base de datos del producto especifico
Entrada	Palabra clave
Salida	Existencia del producto en inventario

Tabla 9.3: REQUERIMIENTO 6

Nombre	Clasificación de inventario
Propósito	Realizar una clasificación de inventario manera rápida
Descripción	Cuando los usuarios necesitan ver las existencias de los productos de su misma clase, se crea una clasificación que lo que hace es hacer la consulta basada en el tipo de clasificación que el usuario necesita
Entrada	Tipo de clasificación
Salida	Inventario clasificado exitosamente

Tabla 9.4: REQUERIMIENTO 7

Nombre	Reabastecer inventario
Propósito	En caso de disminuir de gran manera las existencias de un producto se muestra una lista de dichos productos y permite realizar ordenes de compra para reabastecerlos
Descripción	En una vista aparte del inventario se muestran los productos bajos de existencias, los cuales muestra que se necesitan ser reabastecidos y de esta manera permite a usuario realizar ordenes de compra
Entrada	No aplica
Salida	Vista de productos a reabastecer

Tabla 9.5: REQUERIMIENTO 8

Nombre	Lista de proveedores
Propósito	Establecer un registro de proveedores vinculados al establecimiento
Descripción	En una vista se muestran la lista completa de proveedores que suplen el establecimiento de productos
Entrada	No aplica
Salida	Vista con la lista completa de proveedores

Tabla 9.6: REQUERIMIENTO 9

Nombre	Crear un nuevo proveedor
Propósito	En caso de que el establecimiento necesite un nuevo proveedor se es necesario crearlo para así mismo verlo en la lista de proveedores
Descripción	Dentro del módulo de proveedores se tiene un botón de crear un nuevo proveedor que con este se abre un formulario y al llenarlo se crea y se muestra en la lista de proveedores
Entrada	Datos del proveedor
Salida	Proveedor creado exitosamente

Tabla 9.7: REQUERIMIENTO 10

Nombre	Realizar ordenes de compra
Propósito	Permitir a los usuarios realizar ordenes de compra de los productos creados y bajos en existencias
Descripción	En la vista de productos bajos de existencia se tiene un botón de .orden de compra que este abre un formulario con los datos necesarios para llevar un registro de lo que el establecimiento pide a los proveedores"
Entrada	Datos de la orden
Salida	Vista de productos a reabastecer

Tabla 9.8: REQUERIMIENTO 11

Nombre	Confirmar la llegada de productos
Propósito	El usuario puede registrar la llegada de los productos para llevar un registro de los mismos
Descripción	Después de crear una orden de compra se tiene el estado de la orden, por lo cual se tiene una variable que indica si ha llegado el producto al establecimiento, por lo tanto, se muestra aun botón que confirma la llegada del producto
Entrada	Producto seleccionado
Salida	Vista de productos a reabastecer

Tabla 9.9: REQUERIMIENTO 12

Nombre	Análisis de control de calidad
Propósito	Llevar un registro de calidad de los productos que llegan al establecimiento
Descripción	Luego de registrar la llegada de los productos se abre un formulario para establecer un análisis de control de calidad para realizar preguntas clave sobre el estado de los nuevos productos
Entrada	No aplica
Salida	Formulario de calidad

Tabla 9.10: REQUERIMIENTO 13

Nombre	Crear nuevo cliente
Propósito	Llevar registro de los clientes que compran regularmente en el establecimiento
Descripción	Cuando el cliente compra regularmente en el establecimiento debe tener una opción de crearlo para que con ese registro, se pueda llevar registro de las compras que hace cada cliente en caso tal de que se quiera reclamar por alguna compra
Entrada	Datos del cliente
Salida	Cliente creado exitosamente

Tabla 9.11: REQUERIMIENTO 14

Nombre	Ver compras de los clientes
Propósito	Llevar un registro de compras que realizan los clientes regulares
Descripción	Luego de haber registrado un cliente nuevo, cada compra se relaciona a un cliente en específico, para poder ver que compras se hacen a que clientes
Entrada	Nombre del cliente
Salida	Lista de compras del cliente

Tabla 9.12: REQUERIMIENTO 15

Nombre	Crear un nuevo empleado
Propósito	Llevar un registro de los colaboradores que ingresan a la organización
Descripción	En caso de aprobar la contratación de un colaborador, se ingresa al sistema con usuario y contraseña para que este pueda tener un login en el sistema
Entrada	Datos del empleado
Salida	Empleado creado exitosamente

Tabla 9.13: REQUERIMIENTO 16

Nombre	Deshabilitar un nuevo empleado
Propósito	Remover empleados del sistema
Descripción	En caso de que un colaborador deje de trabajar en el establecimiento se debe remover su usuario de los sistemas, de esta manera, se instancia un botón de remover usuario en la vista de editar usuario.
Entrada	No aplica
Salida	Empleado deshabilitado exitosamente.

Tabla 9.14: REQUERIMIENTO 17

Nombre	Realizar una venta
Propósito	Realizar ventas a los clientes finales
Descripción	Cuando llega un cliente y desea comprar un producto del establecimiento, se hace la lista de cosas por comprar y con el precio sumado de ellos, se descuenta del inventario
Entrada	Productos deseados
Salida	Venta realizada exitosamente

2 Repositorio

Repositorio de GitHub: <https://github.com/alde326/FerreCloud2>

3 Instrumento de evaluación de satisfacción del usuario

1. *Información General*

1. **¿Qué módulo utilizaste más durante la prueba?** *(Selecciona todas las que correspondan)*

- Ventas
- Inventarios
- Proveedores
- Contabilidad
- Reportes
- Otros: _____

2. **¿Cuánto tiempo estimas que dedicaste a probar FerreCloud?**

- Menos de 1 hora
- Entre 1 y 2 horas
- Más de 2 horas

2. *Facilidad de Uso*

1. **¿Pudiste completar las tareas principales que intentaste realizar?**

- Sí, todas
- Sí, la mayoría
- No, tuve problemas

3. Rendimiento y Estabilidad

1. **¿Experimentaste algún problema técnico durante la prueba?** *(Selecciona todas las que correspondan)*

- Lentitud en el sistema
- Errores al cargar datos
- Caídas del sistema
- No experimenté problemas

2. **¿Qué tan rápido respondía el sistema a tus acciones?** *(1 = Muy lento, 5 = Muy rápido)*

- 1
- 2
- 3
- 4
- 5

4. Impresiones Generales

1. **¿Qué tan satisfecho estás con tu experiencia general en esta prueba?** *(1 = Muy insatisfecho, 5 = Muy satisfecho)*

- 1
- 2

- 3

- 4

- 5

2. ¿Recomendarías FerreCloud a otras personas basándote en esta prueba?

- Sí

- No

- Tal vez

5. Comentarios Adicionales

1. ¿Qué funcionalidades te parecieron más útiles durante la prueba?

2. ¿Qué sugerencias tienes para mejorar FerreCloud?

Link de la encuesta: <https://forms.gle/svXHFtfQCEYuxQhJ6>

4 Resultados de encuesta

¿Qué tan rápido respondía el sistema a tus acciones? *(1 = Muy lento, 5 = Muy rápido)*

 Copiar gráfico

5 respuestas

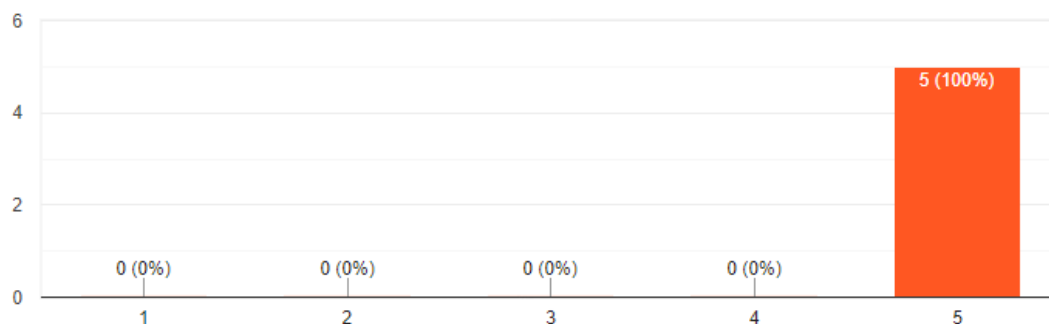


Figura 9.1: Respuesta de pregunta de encuesta de usabilidad

¿Recomendarías FerreCloud a otras personas basándote en esta prueba?

 Copiar gráfico

5 respuestas

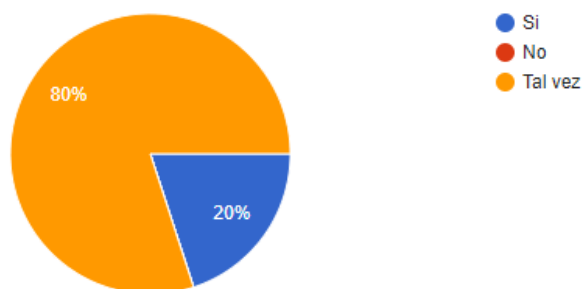


Figura 9.2: Respuesta de pregunta de encuesta de usabilidad

¿Cuánto tiempo estimas que dedicaste a probar FerreCloud?

 Copiar gráfico

5 respuestas



Figura 9.3: Respuesta de pregunta de encuesta de usabilidad

¿Pudiste completar las tareas principales que intentaste realizar?

 Copiar gráfico

5 respuestas

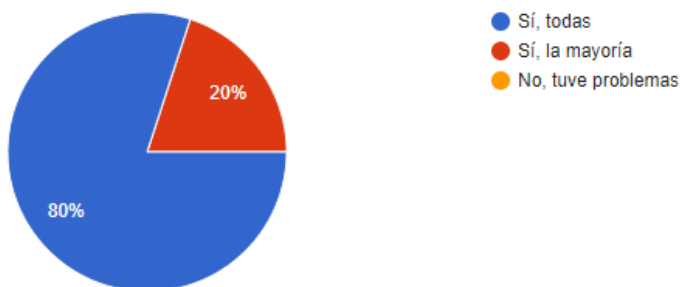


Figura 9.4: Respuesta de pregunta de encuesta de usabilidad

¿Qué funcionalidades te parecieron más útiles durante la prueba?

5 respuestas

Me gustó la parte de ventas, muy facil y rápido
Me gustaron mucho los reportes de contabilidad
muy completo todo
El inventario dentro de ventas
Es bastante sencillo de usar

Figura 9.5: Respuesta de pregunta de encuesta de usabilidad

¿Qué tan satisfecho estás con tu experiencia general en esta prueba?

 Copiar gráfico

(1 = Muy insatisfecho, 5 = Muy satisfecho)

5 respuestas

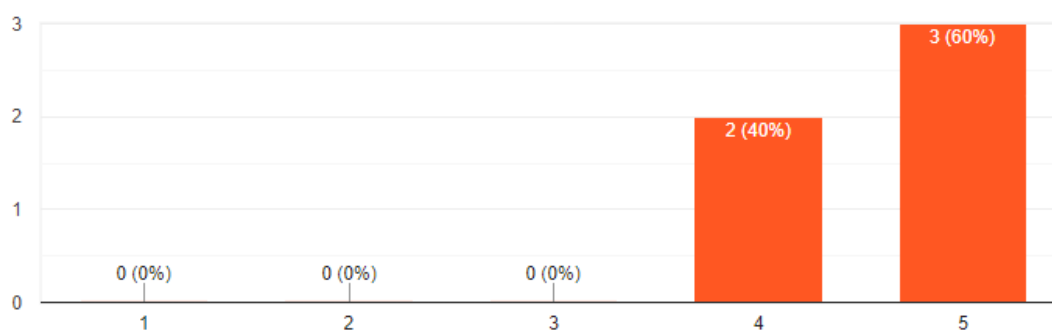


Figura 9.6: Respuesta de pregunta de encuesta de usabilidad

¿Qué sugerencias tienes para mejorar FerreCloud?

5 respuestas

Muy buen trabajo
Sería bueno poner facturación electrónica porque hay clientes que lo piden
ninguna
información mas detallada del inventario de ventas
Ninguna

Figura 9.7: Respuesta de pregunta de encuesta de usabilidad

5 Mockups

Anexo vínculo al repositorio de mockups completos realizados en Figma, de los cuales se basó la implementación de FerreCloud:

Repositorio de Figma: <https://www.figma.com/design/UfO5b2xgsCjykLTDHn8T59/FerreCloud?node-id=0-1&t=0F1KSuHNlaLT1fFE-1>

6 Evidencias

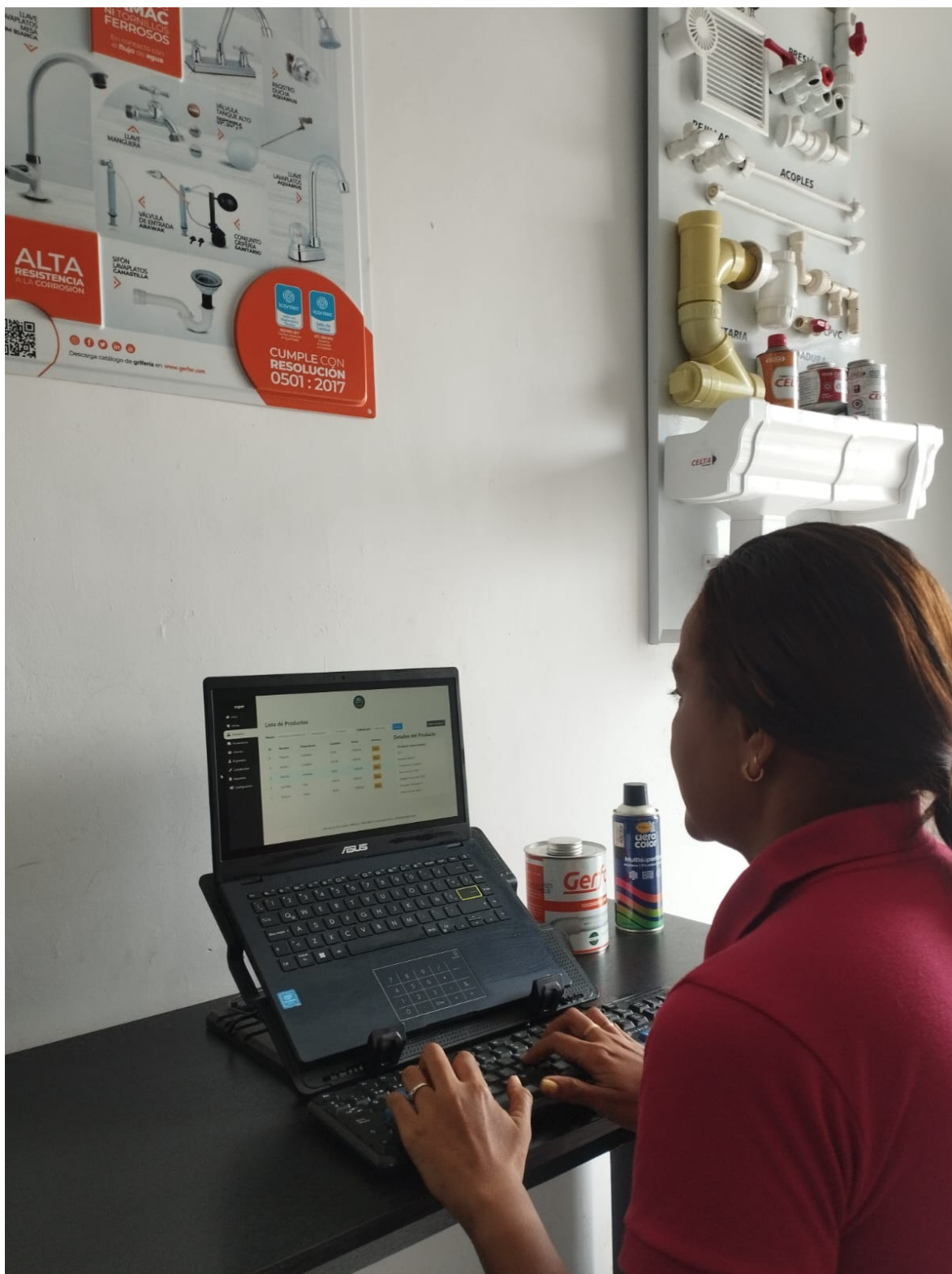


Figura 9.8: Evidencia de pruebas de usabilidad con usuarios

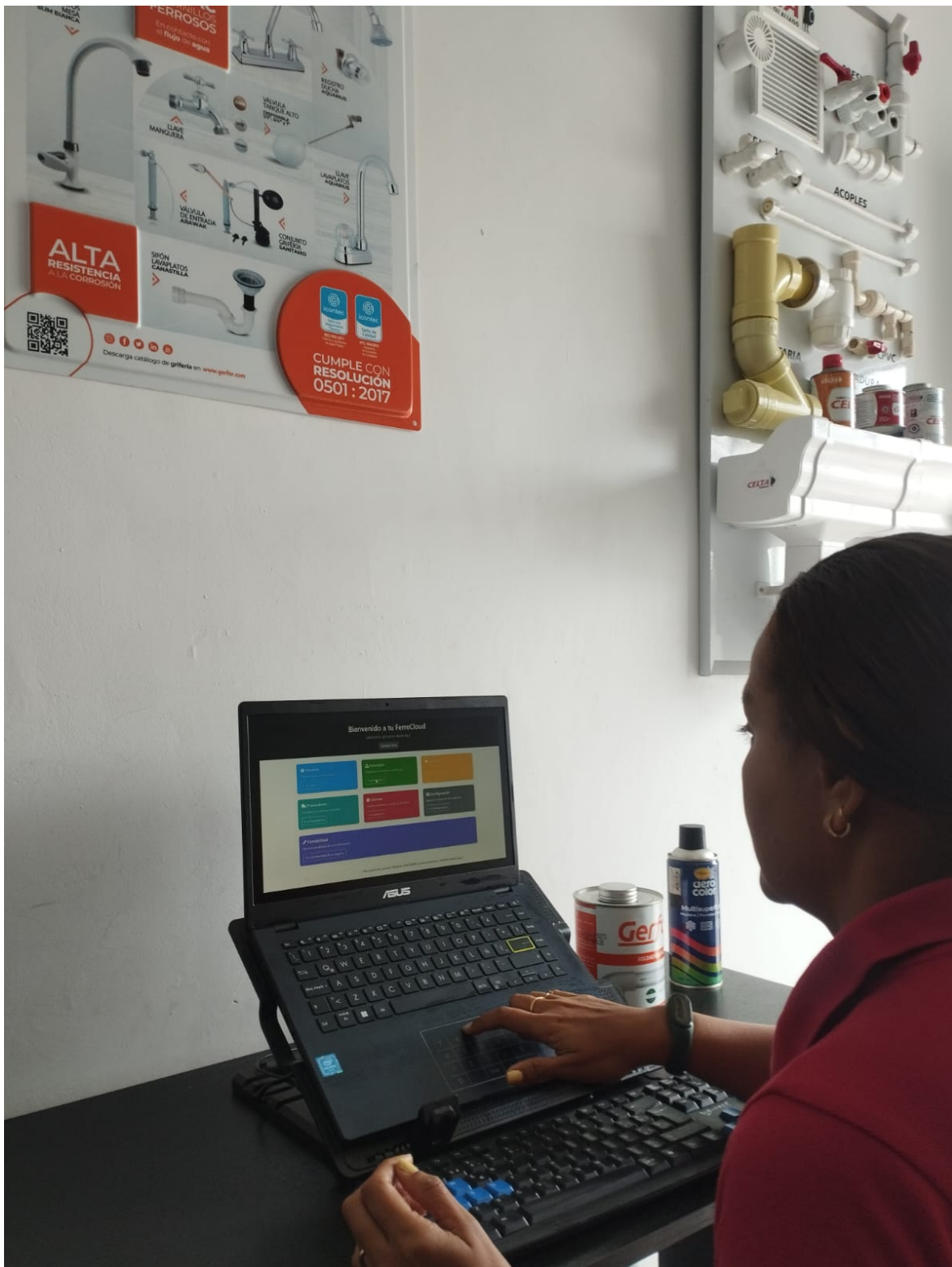


Figura 9.9: Evidencia de pruebas de usabilidad con usuarios



Figura 9.10: Evidencia de pruebas de usabilidad con usuarios

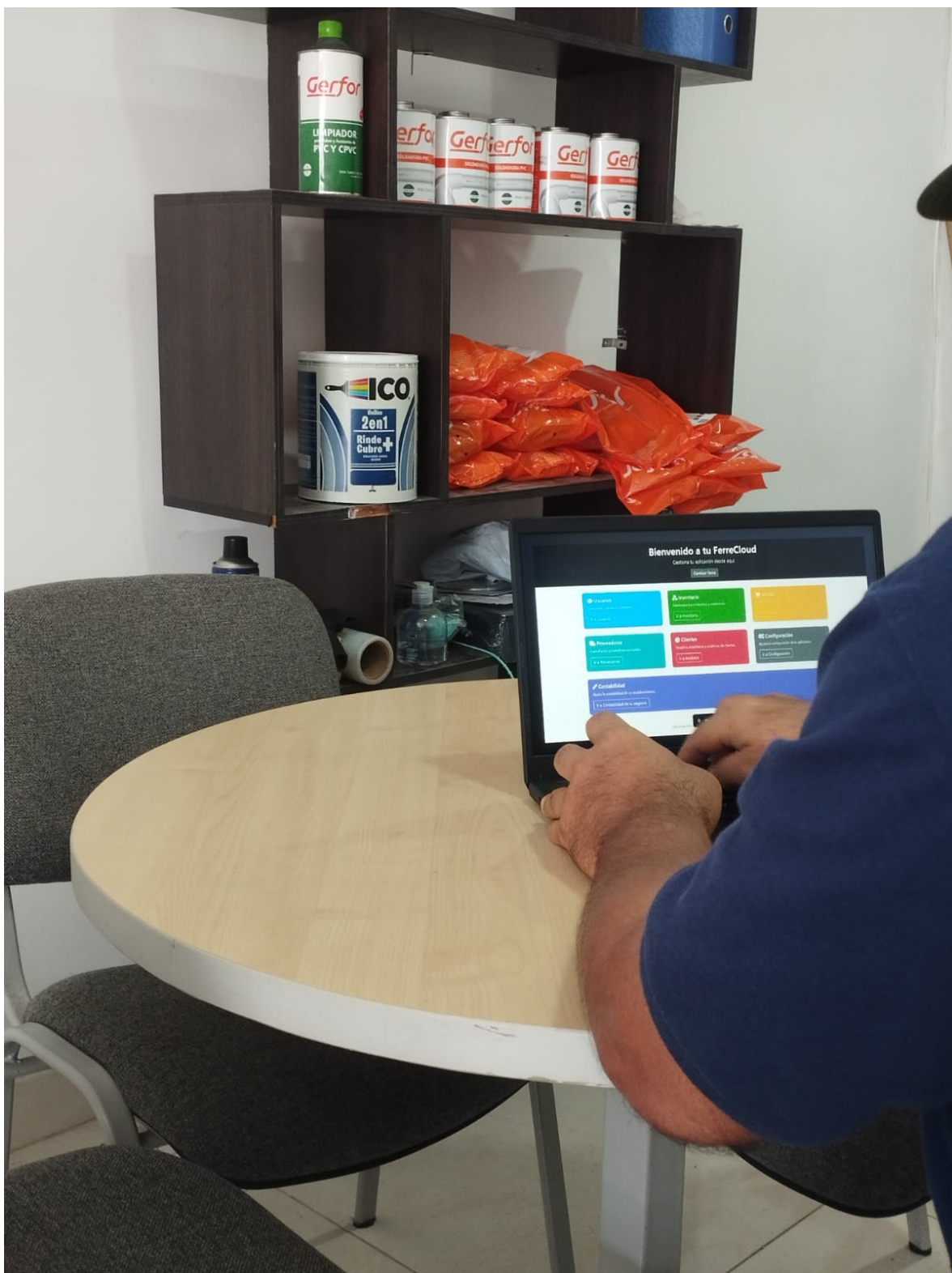


Figura 9.11: Evidencia de pruebas de usabilidad con usuarios



Figura 9.12: Evidencia de pruebas de usabilidad con usuarios

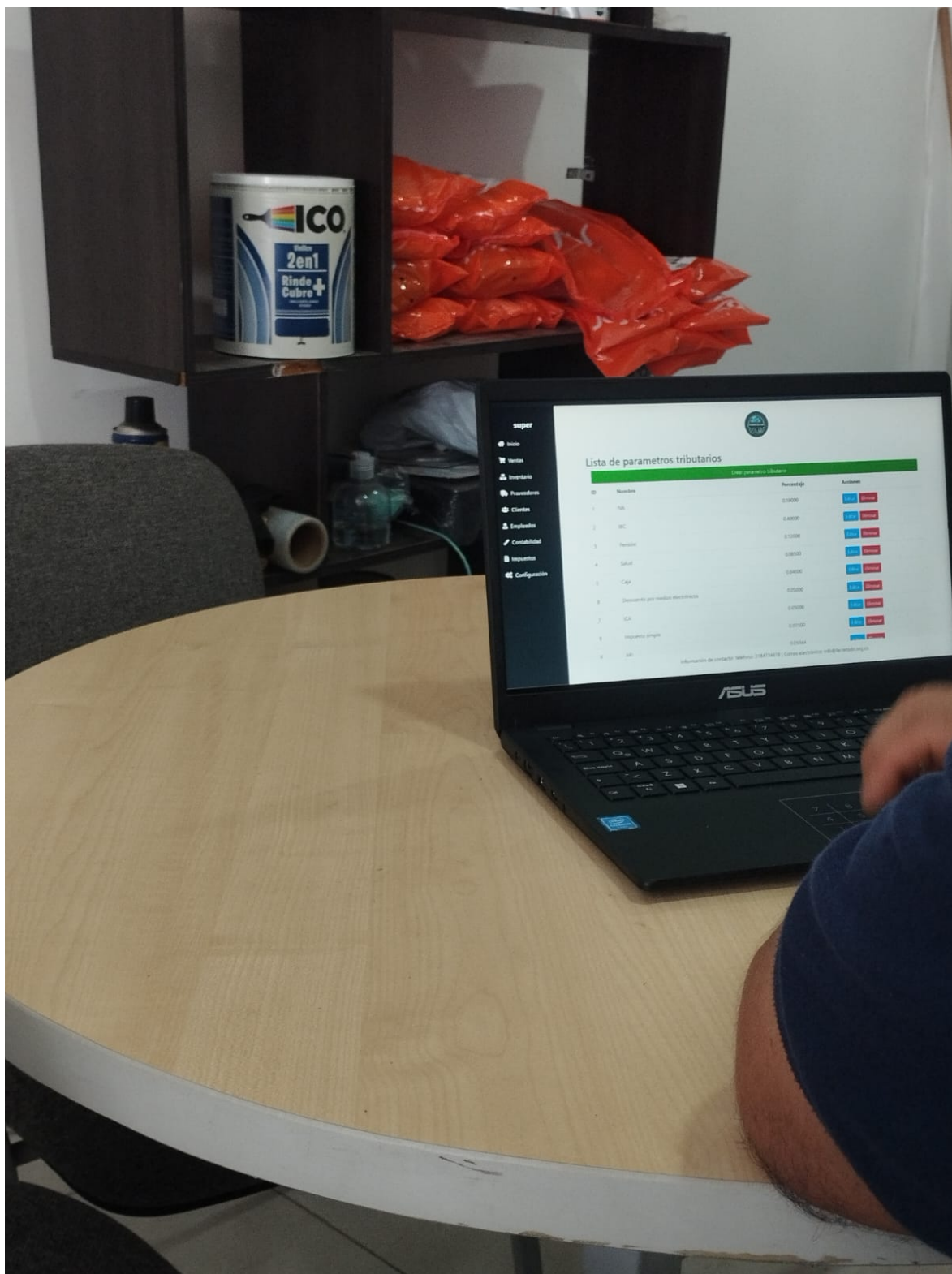


Figura 9.13: Evidencia de pruebas de usabilidad con usuarios