



UNIVERSIDAD TÉCNICA DE AMBATO

**FACULTAD DE INGENIERÍA EN SISTEMAS, ELECTRÓNICA E
INDUSTRIAL
CARRERA DE INGENIERÍA EN SISTEMAS COMPUTACIONALES E
INFORMÁTICOS**

Tema:

**APLICACIÓN WEB PROGRESIVA (PWA) PARA EL CONTROL DE
INVENTARIOS EN LA EMPRESA FERRIMAR.**

Trabajo de Titulación Modalidad: Proyecto de Investigación, presentado
previo a la obtención de título de Ingeniera en Sistemas Computacionales e
Informáticos.

ÁREA: Software

LÍNEA DE INVESTIGACIÓN: Desarrollo de Software

AUTOR: Erika Camila Martinez Shaca

TUTOR: Ing. Mg. Carlos Núñez

AMBATO – ECUADOR

marzo - 2023

APROBACIÓN DEL TUTOR

En calidad de tutor del Trabajo de Titulación con el tema: APLICACIÓN WEB PROGRESIVA (PWA) PARA EL CONTROL DE INVENTARIOS EN LA EMPRESA FERRIMAR, desarrollado bajo la modalidad Proyecto de Investigación por la señorita Erika Camila Martínez Shaca, estudiante de la Carrera de Ingeniería en Sistemas Computacionales e Informáticos, de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial, de la Universidad Técnica de Ambato, me permito indicar que la estudiante ha sido tutorada durante todo el desarrollo del trabajo hasta su conclusión, de acuerdo a lo dispuesto en el Artículo 15 del Reglamento para obtener el Título de Tercer Nivel, de Grado de la Universidad Técnica de Ambato, y el numeral 7.4 del respectivo instructivo.

Ambato, marzo 2023.

Ing. Mg. Carlos Núñez

TUTOR

AUTORÍA

El presente Proyecto de Investigación titulado: APLICACIÓN WEB PROGRESIVA (PWA) PARA EL CONTROL DE INVENTARIOS EN LA EMPRESA FERRIMAR es absolutamente original, auténtico y personal. En tal virtud, el contenido, efectos legales y académicos que se desprenden del mismo son de exclusiva responsabilidad del autor.

Ambato, marzo 2023.



Erika Camila Martínez Shaca

C.C. 1803626199

AUTOR

DERECHOS DE AUTOR

Autorizo a la Universidad Técnica de Ambato, para que haga uso de este Trabajo de Titulación como un documento disponible para la lectura, consulta y procesos de investigación.

Cedo los derechos de mi Trabajo de Titulación en favor de la Universidad Técnica de Ambato, con fines de difusión pública. Además, autorizo su reproducción total o parcial dentro de las regulaciones de la institución.

Ambato, marzo 2023



Erika Camila Martínez Shaca

C.C. 1803626199

AUTOR

APROBACIÓN DE TRIBUNAL DE GRADO

En calidad de par calificador del Informe Final del Trabajo de Titulación presentado por la señorita Erika Camila Martínez Shaca, estudiante de la Carrera de Ingeniería en Sistemas Computacionales e Informáticos, de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial, bajo la Modalidad Proyecto de Investigación, titulado **APLICACIÓN WEB PROGRESIVA (PWA) PARA EL CONTROL DE INVENTARIOS EN LA EMPRESA FERRIMAR**, nos permitimos informar que el trabajo ha sido revisado y calificado de acuerdo al Artículo 17 del Reglamento para obtener el Título de Tercer Nivel, de Grado de la Universidad Técnica de Ambato, y al numeral 7.6 del respectivo instructivo. Para cuya constancia suscribimos, conjuntamente con la señora Presidente del Tribunal.

Ambato, marzo 2023.

Ing. Pilar Urrutia, Mg.
PRESIDENTA DEL TRIBUNAL

Ing. Franklin Mayorga.
PROFESOR CALIFICADOR

Ing. Leonardo David Torres.
PROFESOR CALIFICADOR

DEDICATORIA

Todo este proyecto, este sueño, y esta lucha, se los entregó a mis padres, gracias papá y mamá por ayudarme a llegar hasta aquí, gracias por cada gota de sudor, por cada lagrimea, y por cada sonrisa que han puesto para que hoy su hija este cumpliendo uno de sus sueños; no me alcanzaría la vida para pagarles todo el esfuerzo que ustedes han puesto en esto, pero les aseguro que todo ha valido la pena, espero que se sientan orgullosos de mí, porque todo lo que soy ahora es un reflejo de los excelentes padres que me criaron y que me enseñaron que con trabajo duro puedo obtener grandes logros, y hoy con orgullo puedo decirles “papá y mamá, lo logré”.

Erika Camila Martínez

AGRADECIMIENTO

Todo este trayecto ha estado lleno de altas y bajas, ha sido un camino complejo, pero con la ayuda de mis personas incondicionales hemos logrado llegar hasta este día. Quiero empezar agradeciendo a los protagonistas principales de este sueño, mis padres Francisco y Lastenia, el hecho de que yo haya llegado hasta aquí y está parada frente a todos ustedes se los debo a ellos, que me han apoyado desde mis primeros pasos hasta hoy que estoy a puertas de ser una profesional; a mi hermana Carolina, que ha sido mi compañía diaria; a todos mis profesores de la Universidad Técnica de Ambato que me han ayudado a cursar esta majestuosa carrera, el haberme compartido de sus conocimientos hace que me convierta en el gran ser humano que está presente hoy con ustedes; y mis amigos que hice durante la carrera, que han sido los que han luchado conmigo este título y también son los que me recuerdan día a día que esta meta está cada vez más cerca; y no quiero culminar sin antes agradecer a la Universidad Técnica de Ambato, la que ha sido mi hogar todos estos años, la que me abrió las puertas y me lleno de esperanzas de crecer tanto académicamente como persona, y en la cual me encuentro el día hoy cerrando uno más de mis más grandes sueños y anhelos. Gracias a todos ustedes por haber escrito conmigo este capítulo de mi vida, los llevo en mi corazón eternamente.

Erika Camila Martínez Shaca

ÍNDICE DE CONTENIDOS

APROBACIÓN DEL TUTOR.....	2
AUTORÍA.....	¡Error! Marcador no definido.
DERECHOS DE AUTOR.....	3
APROBACIÓN DE TRIBUNAL DE GRADO.....	5
DEDICATORIA.....	6
AGRADECIMIENTO.....	7
ÍNDICE DE TABLAS.....	11
ÍNDICE DE IMÁGENES.....	16
RESUMEN EJECUTIVO.....	18
ABSTRACT.....	19
CAPITULO I.....	20
MARCO TEÓRICO.....	20
1.1. Tema de investigación.....	20
1.2. Antecedentes investigativos.....	20
1.2.1. Contextualización del problema.....	20
1.2.2. Fundamentación Teórica.....	23
1.3. Objetivos.....	34

1.3.1.	Objetivo General	34
1.3.2.	Objetivos Específicos.....	34
CAPITULO II		35
METODOLOGÍA		35
2.1.	Materiales.....	35
2.2.	Métodos.....	40
2.2.1.	Modalidad de la investigación	40
2.2.2.	Población y muestra	40
2.2.3.	Recolección de información.....	41
2.2.4.	Procesamiento y análisis de datos	52
CAPITULO III		53
RESULTADOS Y DISCUSION.....		53
3.1.	Análisis y discusión de resultados.....	53
3.1.1.	Desarrollo de la propuesta.....	68
3.1.1.1.	Fase de planificación.....	68
3.1.1.2.	Fase de diseño	101
3.1.1.3.	Fase de desarrollo.....	118
3.1.1.4.	Fase de pruebas	137
3.1.1.5.	Fase de lanzamiento	145
CAPITULO IV		148

CONCLUSIONES Y RECOMENDACIONES.....	148
4.1. Conclusiones	148
4.2. Recomendaciones.....	148
MATERIALES DE REFERENCIA.....	150
Referencias Bibliográficas	150
Anexos.....	154

ÍNDICE DE TABLAS

Tabla 1: Ventajas y Desventajas de una PWA	33
Tabla 2: Comparación entre Aplicación Web, Nativa y Progresiva	34
Tabla 3: Formato de Cuestionario de entrevista.....	36
Tabla 4: Cuestionario de entrevista (adicionales)	37
Tabla 5: Formato de Ficha de Observación de Campo	38
Tabla 6: Formato de Ficha de Contenidos.....	39
Tabla 7: Población.....	40
Tabla 8: Recolección de información - Entrevista	45
Tabla 9: Recolección de información – Observación de campo	47
Tabla 10: Recolección de información – Ficha de Contenido No. 1	48
Tabla 11: Recolección de información – Ficha de Contenido No. 2	49
Tabla 12: Recolección de información – Ficha de Contenido No. 3	50
Tabla 13: Recolección de información – Ficha de Contenido No. 4	50
Tabla 14: Recolección de información – Ficha de Contenido No. 5	51
Tabla 15: Comparación de framework de frontend.....	59
Tabla 16: Comparación de framework de backend.....	61
Tabla 17: Comparación de bases de datos.....	64
Tabla 18: Comparación de metodologías ágiles.....	67
Tabla 19: Definición de roles de usuario.....	70
Tabla 20: Modelo de Historia de Usuario	73
Tabla 21: Historia de Usuario 01	74
Tabla 22: Historia de Usuario 02	75

Tabla 23: Historia de Usuario 03	75
Tabla 24: Historia de Usuario 04	75
Tabla 25: Historia de Usuario 05	76
Tabla 26: Historia de Usuario 06	76
Tabla 27: Historia de Usuario 07	77
Tabla 28: Historia de Usuario 08	77
Tabla 29: Historia de Usuario 09	77
Tabla 30: Historia de Usuario 10	78
Tabla 31: Historia de Usuario 11	78
Tabla 32: Historia de Usuario 12	79
Tabla 33: Historia de Usuario 13	79
Tabla 34: Historia de Usuario 14	79
Tabla 35: Historia de Usuario 15	80
Tabla 36: Tarea 001.....	81
Tabla 37: Tarea 002.....	81
Tabla 38: Tarea 003.....	82
Tabla 39: Tarea 004.....	82
Tabla 40: Tarea 005.....	82
Tabla 41: Tarea 006.....	83
Tabla 42: Tarea 007.....	83
Tabla 43: Tarea 008.....	83
Tabla 44: Tarea 009.....	84
Tabla 45: Tarea 010.....	84
Tabla 46: Tarea 011.....	84
Tabla 47: Tarea 012.....	84

Tabla 48: Tarea 013.....	85
Tabla 49: Tarea 014.....	85
Tabla 50: Tarea 015.....	85
Tabla 51: Tarea 016.....	86
Tabla 52: Tarea 017.....	86
Tabla 53: Tarea 018.....	86
Tabla 54: Tarea 019.....	87
Tabla 55: Tarea 020.....	87
Tabla 56: Tarea 021.....	87
Tabla 57: Tarea 022.....	88
Tabla 58: Tarea 023.....	88
Tabla 59: Tarea 024.....	88
Tabla 60: Tarea 025.....	89
Tabla 61: Tarea 026.....	89
Tabla 62: Tarea 027.....	89
Tabla 63: Tarea 028.....	90
Tabla 64: Tarea 029.....	90
Tabla 65: Tarea 030.....	90
Tabla 66: Tarea 031.....	91
Tabla 67: Tarea 032.....	91
Tabla 68: Tarea 033.....	91
Tabla 69: Tarea 034.....	92
Tabla 70: Tarea 035.....	92
Tabla 71: Tarea 036.....	92
Tabla 72: Tarea 037.....	93

Tabla 73: Tarea 038.....	93
Tabla 74: Tarea 039.....	93
Tabla 75: Tarea 040.....	94
Tabla 76: Tarea 041.....	94
Tabla 77: Tarea 042.....	94
Tabla 78: Tarea 043.....	95
Tabla 79: Tarea 044.....	95
Tabla 80: Tarea 045.....	95
Tabla 81: Tarea 046.....	96
Tabla 82: Tarea 047.....	96
Tabla 83: Tarea 048.....	96
Tabla 84: Tarea 049.....	97
Tabla 85: Plan de iteraciones general.....	98
Tabla 86: Plan de iteración 01	98
Tabla 87: Plan de iteración 02.....	98
Tabla 88: Plan de iteración 03.....	99
Tabla 89: Plan de iteración 04.....	99
Tabla 90: Plan de iteración 05.....	99
Tabla 91: Plan de iteración 06.....	99
Tabla 92: Plan de entrega	100
Tabla 93: Metáfora	102
Tabla 94: Tarjeta CRC, Ingresar al aplicativo.....	104
Tabla 95: Tarjeta CRC, Ingresar mercadería.....	104
Tabla 96: Tarjeta CRC, Gestionar Precios	104
Tabla 97: Tarjeta CRC, Generar proformas/facturas	105

Tabla 98: Tarjeta CRC, CRUD proveedores.....	105
Tabla 99: Tarjetas CRC, CRUD clientes.....	105
Tabla 100: Tarjeta CRC, CRUD productos.....	106
Tabla 101: Tarjeta CRC, CRUD usuarios	106
Tabla 102: Tarjetas CRC, Generar reportes	106
Tabla 103: Formato tarjeta de prueba de aceptación.....	137
Tabla 104: Prueba de aceptación 01	138
Tabla 105: Prueba de aceptación 02.....	139
Tabla 106: Prueba de aceptación 03.....	139
Tabla 107: Prueba de aceptación 04.....	140
Tabla 108: Prueba de aceptación 05.....	141
Tabla 109: Prueba de aceptación 06.....	141
Tabla 110: Prueba de aceptación 07.....	142
Tabla 111: Prueba de aceptación 08.....	143
Tabla 112: Prueba de aceptación 09.....	143
Tabla 113: Prueba de aceptación 010.....	143
Tabla 114: Test prueba de usabilidad.....	145

ÍNDICE DE IMÁGENES

Imagen 1: Interfaz de Visual Studio Code	29
Imagen 2: Arquitectura de una PWA	31
Imagen 3: Archivos de un componente Angular.....	55
Imagen 4: Arquitectura MVC Angular	55
Imagen 5: Data-Binding unidireccional y bidireccional	56
Imagen 6: Diagrama de fases de la metodología Extreme Programming	68
Imagen 7: Diagrama de proceso de Compras a proveedores en Ferrimar.....	71
Imagen 8: Diagrama de proceso de Ventas en Ferrimar	72
Imagen 9: Diagrama Entidad-Relación de la Base de Datos	109
Imagen 10: Boceto de diseño Login Usuario	110
Imagen 11: Boceto de diseño Pantalla de Inicio	111
Imagen 12: Boceto de diseño Pantalla de Proveedores.....	112
Imagen 13: Boceto de diseño Pantalla Clientes	113
Imagen 14: Boceto de diseño Pantalla de Productos.....	114
Imagen 15: Boceto de diseño Pantalla de Stock	114
Imagen 16: Boceto de diseño Pantalla Facturas.....	115
Imagen 17: Boceto de diseño Pantalla Proformas.....	116
Imagen 18: Boceto de diseño Pantalla de Reportes	117
Imagen 19: Pantalla final login	121
Imagen 20: Pantalla final lista productos	127
Imagen 21: Pantalla final modal productos.....	129
Imagen 22: Pantalla final lista stock.....	132

Imagen 23: Panel de control de XAMPP	146
Imagen 24: Directorio de XAMPP	146
Imagen 25: Aplicación instalada localmente.....	147

RESUMEN EJECUTIVO

Hoy en día existen una gran variedad de tipos de aplicaciones que han ido mejorando en el transcurso del tiempo a medida que las expectativas y necesidades de los usuarios crecen, estas aplicaciones deben ser adaptadas a todo tipo de empresas.

El presente trabajo está basado en la gestión que maneja la ferretería Ferrimar para el control de su inventario, dado que la información que posee es plenamente manual, propensa a errores humanos y pérdida de información. Se centra también en el control de acceso a los datos y módulos mediante roles y permisos de usuario, encapsulando de esta manera la información sensible y confidencial.

La Aplicación Web Progresiva fue desarrollada con el framework Angular en su versión 13.3.9, que está basado en el lenguaje TypeScript y posee una gran variedad de librerías para dinamizar el desarrollo frontend. Por otra parte, el backend se desarrolló con NodeJS en la versión 16.15.1 con el cual se logró establecer un servidor robusto confiable y con respuestas inmediatas y comunicación directa al servidor de base de datos MySQL Workbench 8.0.

La metodología aplicada para el presente proyecto fue Extreme Programming (XP), ya que al ser de tipo ágil responde rápidamente a los cambios con los requerimientos del cliente necesarios para establecer el correcto funcionamiento de la aplicación en desarrollo.

El sistema facilita el correcto manejo de cantidades y precios de productos dentro de la ferretería Ferrimar, minimizando así los errores cometidos por parte de los empleados y reduciendo costos en procesos como logística y comercio.

Palabras clave: Aplicación web progresiva, Angular, metodologías de desarrollo

ABSTRACT

Today there are a wide variety of types of applications that have been improving over time as the expectations and needs of users grow, these applications must be adapted to all types of companies.

The present work is based on the management that Ferrimar hardware store handles for the control of its inventory, given that the information it has is fully manual, prone to human error and loss of information. It also focuses on controlling access to data and modules through roles and user permissions, thus encapsulating sensitive and confidential information.

The Progressive Web Application was developed with the Angular framework in its version 13.3.9, which is based on the TypeScript language and has a great variety of libraries to dynamize the frontend development. On the other hand, the backend was developed with NodeJS in version 16.15.1 with which it was possible to establish a reliable and robust server with immediate responses and direct communication to the MySQL Workbench 8.0 database server.

The methodology applied for the present project was Extreme Programming (XP), since being of agile type it responds quickly to the changes with the requirements of the client necessary to establish the correct operation of the application in development.

The system facilitates the correct management of quantities and prices of products within the Ferrimar hardware store, thus minimizing errors made by employees and reducing costs in processes such as logistics and commerce.

Keywords: Progressive web application, Angular, development methodologies.

CAPITULO I

MARCO TEÓRICO

1.1. Tema de investigación

APLICACIÓN WEB PROGRESIVA (PWA) PARA EL CONTROL DE INVENTARIOS EN LA EMPRESA FERRIMAR.

1.2. Antecedentes investigativos

1.2.1. Contextualización del problema

Hoy en día, el entorno de empresas y economía global es un tema un tanto complicado en el cual desenvolverse, una de las causas más representativas de esta desorientación es que se carece de referentes, no queda claro el modelo a seguir, y también, que existen muchas maneras de hacer funcionar una empresa, que tal vez en el pasado fueron ejemplares y hoy son expuestas como antimodelos. [1]

La actividad empresarial óptima se caracteriza por generar productos o servicios basados en la gestión del conocimiento y economía creativa. El factor humano está enfocado hacia un desarrollo continuo y explotar el conocimiento técnico, aplicar tecnologías de información, gestión y comunicaron. [1]

Según A. Fierro y F. Fierro:

“El éxito de la empresa comercial comienza y termina con el buen manejo y control de inventarios y la cartera derivada de los mismos, con el apoyo logístico que cumplen los proveedores oportunos.” [2]

El estudio de los inventarios merece una especial atención debido a la gran cantidad de recursos financieros que las empresas invierten en ellos, identificándose como el factor generador de las utilidades en cualquier actividad mercantil. [2]

Al realizar actividades económicas, las empresas requieren de sistemas automatizados para el control y manejo de su stock, esto representa una de las inversiones más importantes que requiere con relación al resto de sus activos ya que engloba tanto los procesos de ventas como de compras. [3]

La carencia de una herramienta para administrar de una manera adecuada la información acerca del stock en la ferretería Ferrimar, impide que la empresa pueda tomar decisiones acertadas, causando un mal manejo de mercadería, generando menor rentabilidad y mayores pérdidas económicas.

Lo que se busca con este proyecto es solventar esta carencia de Software y ayudar en la toma de decisiones.

Delimitación

Línea de investigación: Desarrollo de Software.

Sublínea de investigación: Aplicaciones Web.

Delimitación espacial: Ferretería Ferrimar, ubicada en la parroquia Huachi Grande, Ambato.

Delimitación temporal: La presente investigación se desarrollará durante el periodo octubre 2022 – febrero 2023.

Justificación

La presente investigación tiene como finalidad la solución a la problemática que presenta la entidad, que es la deficiente gestión y control para los materiales y mercadería, mediante el estudio de Aplicaciones Web Progresivas (PWA) y formas

viables de aplicarlas para desarrollar herramientas de gestión [4]. Surge de la interrogante ¿Qué debería mejorar dentro de la entidad para una mejor administración?

Es importante porque para empresas como Ferrimar, que se dedican a la comercialización de productos es necesario un control ordenado de los mismos de tal forma que sea fácil y accesible conocer sus estados ya sea para gestionar mejor sus ventas o pedidos, tomar decisiones a presente o a futuro y la realización de informes.

Se dice que las empresas que tienen un buen desempeño logístico y logran una ventaja competitiva, están proporcionando un nivel de servicio superior a la de sus clientes [5] [6].

El principal beneficiario de la presente investigación será la empresa, dado que, con una mejor gestión de inventarios y bodega tendrán en conocimiento información exacta y veraz en tiempo real acerca de los productos que se encuentren en stock. Esto les ayudará en una mejor toma de decisiones.

En consecuencia, representa también un beneficio para los clientes debido que, se les brindará mejores ofertas de productos y entregas eficientes en los tiempos estimados.

Factibilidad técnica. -

Es factible realizar la presente investigación ya que se cuentan con los recursos tecnológicos y también acceso a la información para el desarrollo del sistema planteado.

Factibilidad operativa. -

Por parte de la entidad se cuenta con el apoyo total, permitiendo el acceso a la información necesaria para el diseño del sistema web asegurando así buenos resultados de acuerdo con los requisitos.

Factibilidad económica. -

Económicamente el proyecto es factible ya que los costos que implican el desarrollo e implantación del sistema y tiempo son asumidos por el investigador.

1.2.2. Fundamentación Teórica

Estudio del arte

Luego de revisar los diferentes trabajos de investigación realizados en las distintas universidades del Ecuador y a nivel internacional acerca de aplicaciones PWA y automatización y sistemas en inventario y o bodegas en las empresas, se pueden mencionar los siguientes:

Jessica Moreta Romero de la Universidad Técnica de Ambato, en su trabajo de titulación llamado “Sistema de control y seguimiento de los procesos de recursos humanos para los empleados de la Empresa Eléctrica Ambato Regional Centro Norte S.A. mediante una aplicación web progresiva (PWA)”, donde su objetivo principal fue desarrollar un sistema para el control y seguimiento de procesos de recursos humanos mediante una PWA, aplico una encuesta y entrevistas con el fin de recolectar información y determinar los requerimientos funcionales para la aplicación. Una vez culminado el desarrollo llego a la conclusión que la aplicación web progresiva simplificó los procesos que se ejercían desde el departamento de recursos humanos. [7]

El trabajo de titulación presentado por Wilmer Oñate Calderón en 2020, titulado “Implementación de una aplicación web progresiva para la gestión de pruebas de simulación para el ingreso a instituciones militares y universidades en el centro de capacitación y nivelación académica smartel”, de la Universidad Técnica de Ambato, el autor realizó una investigación previa acerca de tecnologías para el desarrollo de páginas web progresivas, comparando así Vue, React, Angular, etc. Escogiendo Angular ya que demostró ser un framework más robusto y escalable, obteniendo así ventaja sobre los otros. [8]

El trabajo de titulación de Michael Raúl Zambrano, con el título “Desarrollo de una tienda virtual progressive web apps (PWA) para gestionar las ventas de los productos en la empresa Garcés Torres (GT) Jean’s cupido.” (2021), para la Universidad Técnica de Ambato. Donde se realizó un estudio de la mejor herramienta PWA y su aplicación. Donde comparo parámetros como la documentación disponible, el tipo de enlace de datos, reutilización de código, el modelo con que trabajan, entre otros, estos datos también se tomaran en cuenta para realizar la comparativa en la presente investigación.

El autor aquí también recomienda la aplicación de las PWA para gestión de ventas en empresas mediante una metodología de desarrollo ágil. [9]

La tesis de Luis Eduardo Aldas Aldaz con el título “Sistema web para el control de facturación e inventario de medicamentos y bienes en el hospital regional docente Ambato” (2013), de la Universidad Técnica de Ambato. Aquí concluye que el control de facturación e inventario en la entidad mediante el manejo de tecnología potenció el nivel de control sobre sus procesos y que toda la información que maneja las entidades ya sean públicas o privadas es importante a la hora de tomar decisiones. [10] Por último, el proyecto técnico previo a la obtención de ingeniera en sistemas de Brigitte Lucas Vega en 2017, titulado “Desarrollo e implementación de Aplicación Web para el Control de Inventario del local comercial Maquinas Hidalgo” para la universidad Politécnica Salesiana. En sus objetivos están la generación de informes de la mercadería y sus movimientos, lo cual se logró con resultados satisfactorios, lo cual asegura que es una aplicación contribuyente a la mejora del local comercial y se disminuyó la carga operativa. También los informes desarrollados en la Aplicación intervienen en la toma de decisiones a tiempo. [11]

Control de inventarios

La gestión de materiales y bodega es un eslabón importante en una empresa ya que su misión es conseguir que todos los materiales estén en cantidad y calidad disponibles para ser despachados en los plazos planificados. Se trata de cumplir con las expectativas del cliente con el menor costo posible.

Existen actividades importantes para la adecuada gestión de inventario, entre las cuales están: [5]

- El análisis de demanda.
- Clasificación de los productos.
- Políticas de inventario.
- Análisis de proveedores.
- Análisis y gestión de almacenamiento.

- Gestión de transporte.

Control interno de inventario

Es el conjunto de procedimientos, normas y métodos que se utilizan dentro de una entidad para planificar y controlar sus productos o materiales, de manera que esta pueda funcionar eficazmente. Esto ayuda a estimar cuando se debe reabastecer de mercancía y llevar de una mejor manera la información. [12]

Métodos de control de inventario

Existen múltiples maneras de llevar un control de inventario, por ejemplo, se puede clasificar el o los productos según su naturaleza, es decir, si hacen referencia a materia prima, productos en proceso, y productos terminados.

Otra manera de llevar el control es según el proceso logístico, es decir que existirá inventario en bodega, pero también habrá material en proceso de producción o que se encuentran dentro de una red logística.

Por último, el control según la funcionalidad del producto, que significa realizar un registro de los productos que serán comercializados de manera normal, y también el registro para cubrir en caso de problemas o fallas con el producto. [12]

Gestión vs Control de Inventario

Son dos conceptos similares sin embargo poseen una leve diferencia, mientras que la gestión de inventario implica el movimiento general de productos y materia a través de la cadena de suministro, el control de inventario se ocupa únicamente de las existencias de productos en un almacén controlar el estado en que se encuentran.

El manejo de inventario involucra acciones como la predicción de demanda de los clientes, esto quiere decir que dentro de esta área se estudiará el movimiento con el

que entran y salen los productos, por ende, puede llevar control de pedidos y compras. Adicionalmente se monitoriza el orden y la gestión de los productos dentro de la cadena de suministro, es decir desde que ingresas a la bodega hasta el momento en que se da uso a los productos o se entrega a un cliente.

Por otra parte, el control de inventario se encarga de la verificación de calidad y almacenamiento en si de los materiales o productos. Se verifica la cantidad existente, sus fechas de caducidad, problemas de almacenamiento, etc.

Gestión de la información

Se trata de la manipulación correcta de la información y datos que se posee, con calidad y exactitud para servir a los objetivos de una organización y llegar a un fin común. En una empresa no solo es importante el manejo de sus bienes materiales, sino también de como maneja la información ya sea interna, externa o corporativa.

Las principales funciones de la Gestión Información van desde la determinación de las necesidades en correspondencia a funciones y actividades, la mejora continua de canales y acceso a la información, y el empleo eficiente de los recursos. [13]

Teoría de Bodegas

Generalmente, se define una bodega como un espacio para almacenar mercancías, y o productos que pueden ser materia prima, producto semiterminado o terminado. En este espacio los productos se conservan en condiciones óptimas y esperan a ser despachados.

Existen principios que se consideran genéricos, que se pueden aplicar para cualquier tipo de almacén que son:

Los productos deben estar siempre bajo la responsabilidad de un custodio o supervisor, que será el encargado de mantener el orden y seguridad de la mercadería.

Toda entrada o salida de productos debe tener su debida documentación autorizada y registro.

Se deben repartir las funciones dentro del almacén de forma equitativa entre los empleados, estas tareas pueden ser la recepción, revisión, registro, almacenamiento, etc.

La empresa debe llevar un registro diario de todas las entradas y salidas de los productos, esto es para llevar un mejor control y un stock actualizado.

La manera en que están ubicados los productos debe ser intuitiva y fácil de reconocer, por lo que se sugiere colocar los productos de acuerdo con su categorización.

Para todos los productos se debe asignar una identificación única, la cual debe estar codificada de manera estándar.

Existen bodegas manuales y bodegas automatizadas. [15]

Bodegas Manuales

Las bodegas manuales se definen como aquellas que en su mayor porcentaje es operada por personal humano, el cual debe estar capacitado para la correcta recepción, almacenaje, manipulación y mantenimiento de los productos. [15]

Básicamente las funciones físicas del personal mencionado deben ser:

- Descarga de mercadería de proveedores.
- Verificar la cantidad y la calidad de los artículos nuevos
- Colocar la mercadería en el lugar asignado
- Realizar lista de pedidos o productos faltantes
- Despacho de productos

Además, deberá también realizar actividades administrativas dentro del control de inventario como:

- Documentación de la recepción de materiales
- Actualización en los registros de inventario el ingreso de productos

- Actualización de salida de productos
- Notificación a contabilidad del movimiento de caja y productos.

Bodegas Automatizadas

Son bodegas que requieren de la menor intervención humana posible, ya que poseen sistemas de control automatizados que controlan el inventario en general y optimizan realizando procesos con mayor rapidez. El costo de estos puede ser mucho menor a llevar una bodega manual, esto depende de distintos agentes como el número de artículos, número de clientela, volumen de ventas. [15]

Visual Studio Code

Visual Studio Code es un editor de código creado por Microsoft es libre y multiplataforma, no debe confundirse con Visual Studio. Este software se encuentra disponible tanto para Windows, como para Linux y macOS. Cuenta con varias opciones de personalización, además de múltiples extensiones que ayudan a las necesidades del desarrollador, como cambiar atajos, la apariencia o accesos rápidos.

Visual Code además trabaja de forma integrada con Git sin necesidad de agregar ningún plugin o extensión, lo cual le da una gran ventaja a desarrolladores que trabajan con sistemas de control de versiones para una mejor organización de proyectos. [16]

Ventanas y funcionalidad

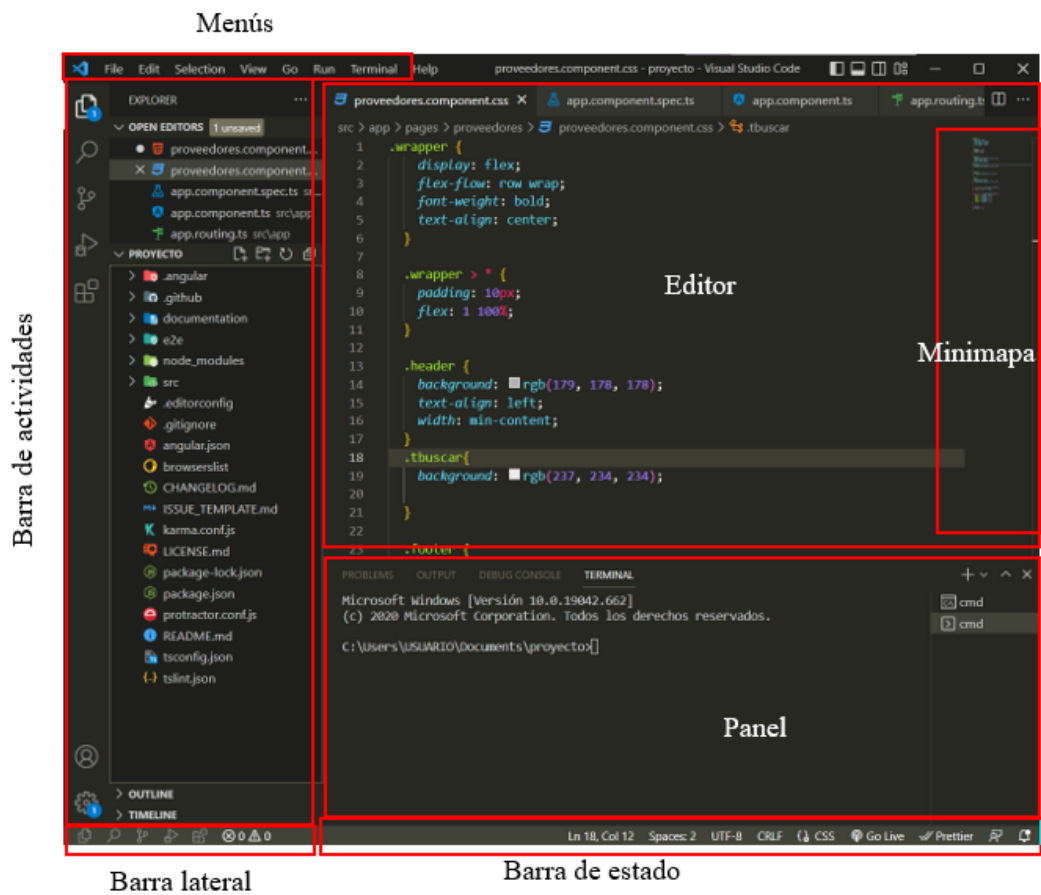


Imagen 1: Interfaz de Visual Studio Code
Elaborado por: Investigador

- En la parte superior izquierda de la venta se encuentra los distintos menús con los que cuenta el editor.
- En la barra de actividades se puede encontrar el proyecto en el que se está trabajando y, en su parte superior indica las pestañas que están abiertas.
- Dentro de la pantalla de editor es donde se va mostrando el código de cada uno de los archivos que el usuario seleccione, en este mismo espacio, pero a la derecha se ubica un Mini mapa donde se puede ver en qué parte del archivo se encuentra editando el código
- Por último, en el panel es donde se despliegan los mensajes de problemas, devolución de errores, consola o los terminales.

Aplicación Web

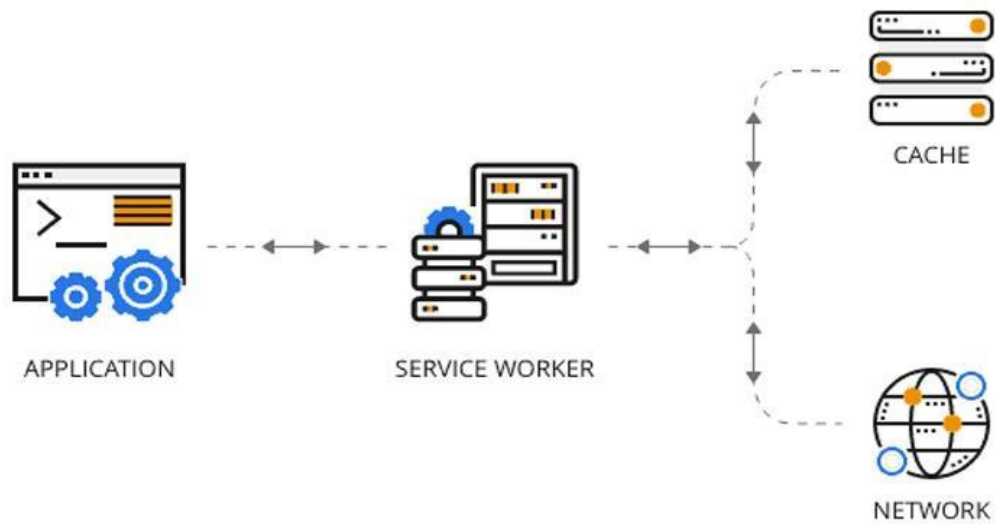
Son aquellas aplicaciones software creadas e instaladas sobre un servidor en internet o sobre una intranet, mas no sobre el sistema operativo. Es muy similar a páginas web, sin embargo, estas brindan respuestas a casos muy particulares. Se pueden utilizar sobre cualquier navegador web, independientemente del sistema operativo. Esta característica ayuda en la rapidez de la obtención y gestión de información. [17]

Aplicaciones Web Progresivas (PWA)

Las Aplicaciones Web Progresivas son una combinación entre aplicaciones web y aplicaciones nativas, ya que hacen uso de *service workers* (script que se ejecuta en segundo plano y permite implementar funcionalidades que no requiere de una página web ni interacción con el usuario [18] entre otras tecnologías para que puedan seguir ejecutándose en segundo plano. Pueden llevar a cabo tareas de aplicaciones nativas como el envío de notificaciones o funcionamiento sin conexión a internet. [19]

Componentes de las PWA

Sus principales componentes son Service Worker, App Shell, Web App Manifest y Push Notification, todas estas en conjunto permiten que la aplicación pueda cumplir cualquier tipo de funcionalidad. [13]



*Imagen 2: Arquitectura de una PWA
Fuente: [20]*

1.- Como se puede ver en la *Imagen 2*, un componente importante es el Service worker que es o son archivos JavaScript, que se ejecutan en segundo plano y por ellos no interactúan directamente con el objeto window o document, sino tienen acceso a los recursos es decir eventos de instalación, peticiones a recursos web, memoria caché, etc. Tienen la responsabilidad de determinar la información que será almacenada en la cache del dispositivo y que información debería pasar con peticiones web en caso de que no exista conexión a internet. [21]

2.- Application Shell (app Shell) contiene los recursos locales que necesita la PWA para cargar la estructura de la interfaz de usuario para que al momento que se pierda la conectividad con internet, esta siga funcionando y mostrándose al usuario. No es un requisito necesario para construir un PWA, pero si es de utilidad ya que provee un mejor rendimiento. [21]

3.- El Web App Manifest

Esta particularidad está muy enfocada a los dispositivos móviles. Según Dennis Sheppard (2017) con esta herramienta se puede especificar cosas como los iconos con el que se guardará la aplicación en la pantalla de inicio, configurar temas visuales e incluso una interfaz para que se muestre en lo que tarda en cargar el navegador, esto

evitará pantallas en blanco. Todas estas configuraciones se deben detallar en archivo JSON (JavaScript Object Notation) y referenciarlo al archivo HTML de la app. [13]

4.- Notificaciones Push

Estas notificaciones se originan en el servidor, entonces el navegador no necesita ser abierto para recibir dichas notificaciones. Soy muy útiles ya que provee al cliente de información pertinente, por ejemplo, un recordatorio para una reunión. Después el usuario decide si abrir o no la aplicación a través de la notificación.

Características de las PWA

Se construyen utilizando estándares de desarrollo web, es decir que, con las bases de HTML (HyperText Markup Language), CSS (Cascading Style Sheets) y JavaScript junto con algunas APIS (Application Programming Interfaces) se logra construir una PWA.

Sin embargo, no todas las aplicaciones web pueden ser aplicaciones PWA, ya que las PWA utilizan el protocolo HTTP (HyperText Transfer Protocol) encriptado, es decir, hacen uso de HTTPS (HyperText Transfer Protocol Secure)

Incrementan su funcionalidad, esto quiere decir que a medida que se utilizan pueden irse haciendo más eficaces, optimizando recursos y memoria.

También poseen las cualidades más destacables de las aplicaciones nativas como ser responsive, y poder ejecutarse y adaptarse en dispositivos móviles como celulares o tablets y también en computadoras ya sean laptops o de escritorio, estilo y navegación propias de aplicaciones, ejecución offline lo cual se consigue mediante el uso de la memoria caché.

Se pueden añadir a la pantalla de inicio, tanto en equipos de cómputo como en celulares, dando así la apariencia de ser una aplicación instalada.

Son como su nombre lo indica, progresivas, que quiere decir que se adaptan a cada actualización de los navegadores en los que vaya a correr la app.

También cabe destacar que se instalan directamente desde el navegador web y no dependen de tiendas de Apple o Android, facilitando así también la publicación de las mismas apps y acortando tiempos. [22] [23]

Con las definiciones expuestas se concluye que:

PWA	
VENTAJAS	DESVENTAJAS
- Comportamiento de una aplicación nativa	- No tiene acceso a las funciones del dispositivo.
- Fácil instalación	- Mayor consumo de energía
- Multidispositivo	- Mayor curva de aprendizaje
- Son apps seguras	

*Tabla 1: Ventajas y Desventajas de una PWA
Elaborado por: Investigador*

Comparación entre aplicación Web, Aplicación Web Progresiva y aplicación Nativa

En la *Tabla 2* que se muestra a continuación, se realizó una comparación entre las aplicaciones web, aplicaciones nativas y aplicaciones web progresivas.

	Usabilidad	Desarrollo	Apariencia	Instalación	Acceso a internet
App web	Accesible desde cualquier navegador	usa HTML, CSS, JS	Parecido a una app nativa	No	Si
App nativa	Sistema operativo móvil específico	Lenguajes y SDK (Software Development Kit)	Proporcionada por el fabricante del SO (Sistema Operativo)	Si, desde la tienda de cada dispositivo	No

Aplicación Web Progresiva	Navegadores específicos	Frameworks y tecnologías web	Parecida a una app nativa	No	No
---------------------------	-------------------------	------------------------------	---------------------------	----	----

*Tabla 2: Comparación entre Aplicación Web, Nativa y Progresiva
Elaborado por: Investigador*

Como se puede observar, las aplicaciones Web Progresivas combina características entre las dos otras opciones, por lo que se comporta como una página web, pero al mismo tiempo hace uso de diferentes tecnologías que permite utilizarla como una app nativa. Todo esto proporcionara un mejor rendimiento y experiencia con el usuario. [24]

1.3.Objetivos

1.3.1. Objetivo General

Implantar una Aplicación Web Progresiva (PWA) para el control de inventarios para la empresa Ferrimar.

1.3.2. Objetivos Específicos

- Determinar los procesos involucrados en el control de inventarios dentro de la entidad.
- Investigar las tecnologías web para crear Aplicaciones Web Progresivas para el desarrollo del proyecto.
- Desarrollar una Aplicación Web Progresiva de acuerdo con la lógica de negocio que permita mejorar el manejo de inventarios y material en la ferretería Ferrimar.

CAPITULO II

METODOLOGÍA

2.1. Materiales

Para la recolección de información se utilizaron tres técnicas: entrevista, observación de campo y recolección de documentación. En primer lugar, se aplicó una entrevista a todo el personal de la empresa, luego se combinó con una ficha técnica de observación de campo y para finalizar, la recolección de documentación, con la finalidad de abarcar todos los detalles que involucran el manejo de inventario dentro de la entidad.

1. Entrevista

La entrevista consta de dos partes, la primera engloba todo lo que se requiere conocer acerca de procesos y procedimientos. Y la segunda parte, contiene preguntas acerca del manejo de productos y mercadería.



UNIVERSIDAD TÉCNICA DE AMBATO
FACULTA DE INGENIERÍA EN SISTEMAS
ELECTRÓNICA E INDUSTRIAL



Objetivo: Recolectar la mayor cantidad de información acerca del proceso de compra, venta y gestión de artículos dentro de la entidad.

N.º	Pregunta
Sobre procesos:	
1	¿Cree que la empresa maneja adecuadamente un control de inventarios-bodega?
2	¿Cuál es el procedimiento para manejar el inventario?
3	¿Existe un control para el ingreso y salida de productos?

4	Existe el control sobre el préstamo/alquiler de máquinas?
5	¿Cuál es el procedimiento para la compra de materiales?
6	¿Cuál es el proceso cuando se detecta uno o varios faltantes con respecto a la compra de productos?
7	¿Existe un documento/manual que describa los procesos de recepción almacenamiento y conservación de material?
8	¿Cuándo se efectúa una devolución de material cual es el proceso de devolución o de dada de baja?
Sobre manejo de mercadería:	
9	¿Se efectúa de forma periódica comprobaciones físicas de las existencias?
10	¿Se encuentra debidamente asegurados los inventarios, teniendo él cuenta el valor de estos? / Como se encuentra ordenado los materiales
11	¿Existe un tratamiento contable para el inventario obsoleto?
12	¿Existe un sistema de codificación para sus productos? ¿Cual?
13	¿Existe alguna restricción de ingreso de mercadería?
14	¿Se tiene de alguna forma identificados los productos que más flujo tienen en la empresa?
15	¿Se encuentran identificadas las áreas de los productos de alguna manera?
16	¿Al recibir productos se toma en cuenta el estado y ensamblaje en que llegan?

*Tabla 3: Formato de Cuestionario de entrevista
Elaborado por: Investigador*

En adición, se realizarán tres preguntas más al contador de la empresa, referentes a su área, que son:

N.º	Pregunta
1	¿Cuál es la forma de cálculo del precio de los productos?
2	¿Existen productos con y sin IVA? ¿Cuáles son?
3	¿Cómo es el formato de factura?

*Tabla 4: Cuestionario de entrevista (adicionales)
Elaborado por: Investigador*

2. Observación de campo

Se decidió complementar la entrevista con una observación de campo ya que permite adicionar información que va tomando en cuenta el investigador durante la visita a la empresa. A continuación, se presenta la ficha que se utilizó para llevar a cabo la observación de campo.



UNIVERSIDAD TÉCNICA DE AMBATO
FACULTA DE INGENIERÍA EN SISTEMAS
ELECTRÓNICA E INDUSTRIAL



FICHA DE OBSERVACIÓN DE CAMPO

Objetivo: Complementar información a la entrevista y realizar la descripción completa del lugar y las funciones de las personas involucradas.

Tema de investigación:	
Datos del Investigador:	
Lugar:	
Fecha y Hora:	
Categorías	Descriptorios

¿Qué persona realiza la facturación?	
¿Cuántas personas se encuentran?	
¿Dónde se lleva el control de productos?	
¿Cómo se realiza un pedido o proforma?	
¿Se manejan tarifas extras en la facturación?	
¿Se manejan descuentos facturación?	
¿Qué se hace con las facturas anuladas?	
¿Existen servicios adicionales?	
Conclusiones	

*Tabla 5: Formato de Ficha de Observación de Campo
Elaborado por: Investigador*

3. Recolección de documentación.

Con el fin de tomar muestras reales de la documentación que se maneja en la empresa se recolectaron:

- Factura de compra
- Factura de venta
- Proforma
- Lista de inventario

A continuación, se presenta el formato de ficha de contenidos que se utilizó para la recolección documental.

Ficha de Contenido			
Número:		Categoría:	
Descriptores			
Observación			

*Tabla 6: Formato de Ficha de Contenidos
Elaborado por: Investigador*

En el presente trabajo investigativo también se utilizó fuentes confiables de información, tales como repositorios académicos, libros, revistas científicas, sitios web, trabajos de titulación en el área de informática.

2.2. Métodos

2.2.1. Modalidad de la investigación

- Investigación Bibliográfica

La presente investigación es bibliográfica debido a que se toma como apoyo tesis del área informática, libros, documentos técnicos, revistas, artículos existentes para la elaboración del marco teórico sobre generación de aplicaciones web progresivas.

- Investigación de Campo

La investigación es también de campo debido a que se busca obtener información correspondiente a los procedimientos y herramientas para el control de materiales dentro de bodega con el personal involucrado en el tema.

2.2.2. Población y muestra

Población

A continuación, se presenta la población total que es el grupo de personas que trabajan en la empresa Ferrimar.

No.	Tipo de población	Descripción	Número	Porcentaje
1	Gerente	Persona encargada de la empresa	1	20%
2	Empleados	Personas que realizan los pedidos, manejo de productos, y manejan bodega	2	40%
3	Ayudante de bodega	Persona que ayuda en el despacho de material	1	20%
4	Contador	Persona que maneja el área contable	1	20%
TOTAL			5	100%

*Tabla 7: Población
Elaborado por: Investigador*

Se trabajó con la totalidad del universo sin que sea necesario sacar muestras representativas, ya que no sobrepasa los cien elementos.

2.2.3. Recolección de información

La entrevista estuvo dirigida específicamente a los usuarios que harán uso de la aplicación, para lo que fue necesario utilizar como instrumento el cuestionario de preguntas abiertas.

El objetivo principal es recolectar la mayor cantidad de información acerca del proceso de compra, venta y gestión de artículos dentro de la entidad.

La entrevista se realiza en las instalaciones de Ferrimar, el día 3 de junio de 2022, obteniendo los resultados descritos en la siguiente tabla:

Pregunta		Resultado	Conclusión
Procesos			
1	¿Cree que la empresa maneja adecuadamente un control de inventarios-bodega?	No se tiene un control adecuado de cuanto material sale o entra y cuanto se tiene disponible. Esto lleva a la adquisición innecesaria de algunos productos o, por otro lado, el desabastecimiento y pérdida de clientes.	La empresa lleva la mayoría de los controles manualmente y mediante resultados de facturación El gerente no puede visualizar un resumen o un estado de ventas a tiempo real
2	¿Cuál es el procedimiento para manejar el inventario?	Mediante la revisión de facturas de compra. Se sabe los productos que se posee en stock revisando las ultimas facturas de compra, o de manera visual en el almacén.	
3	¿Existe un control para el ingreso y salida de productos?	Si, mediante las facturas de compra y venta. Regularmente no se realizan comprobaciones de que el material que se compra se encuentre en bodega. Pero se pueden llegar a hacer estimación mediante la cantidad de comprados-vendidos en las facturas.	
4	Existe el control sobre el préstamo/alquiler de máquinas?	No, debido a que no se realiza muy frecuentemente. Debería llevarse un registro ya que existen algunas herramientas que se prestan, pero a veces los clientes no las devuelven.	
5	¿Cuál es el procedimiento para la compra de materiales?	Se anota los productos que se van terminando o que se requieren y se revisa con los proveedores los precios normalmente los últimos días de la semana. Luego se realiza el pedido y en fin de mes se pagan todas las facturas.	

6	¿Cuál es el proceso cuando se detecta uno o varios faltantes con respecto a la compra de productos?	Se comunica al proveedor para informar del faltante. Y el proveedor lo hace llegar antes de la próxima compra.	
7	¿Existe un documento/manual que describa los procesos de recepción almacenamiento y conservación de material?	No. Al llegar el material o productos nuevos, se cuenta la cantidad y revisión superficial para verificar el buen estado. Se revisan aspectos como las características (que sean las que están facturadas), que los empaques lleguen en buen estado o no estén abiertos.	
8	¿Cuándo se efectúa una devolución de material cual es el proceso de devolución o de dada de baja?	Se contacta al proveedor y se realiza una nota de crédito a favor y el producto es devuelto con los mismos transportistas.	
Manejo de mercadería			
9	¿Se efectúa de forma periódica comprobaciones físicas de las existencias?	No. Se puede dar el caso de que el gerente o empleados requieran saber la existencia para realizar el análisis del producto	Existe La posibilidad de cometer errores
10	¿Se encuentra debidamente asegurados los inventarios, teniendo él cuenta el valor de estos? / Como se encuentra ordenado los materiales	Si, los productos o materiales que son más costosos o pequeños se encuentran dentro del establecimiento y de un pequeño cerramiento donde solo accede el personal, mientras que el material más grande y pesado se encuentra fuera del establecimiento	humanos durante el control de inventario, por lo que se encuentra
11	¿Existe un tratamiento contable para el inventario obsoleto?	No, se encuentra almacenado.	muy necesario automatizar dicho

12	¿Existe un sistema de codificación para sus productos? ¿Cual?	No. Los productos deberían identificarse primero por nombre de producto, marca, modelo y luego las demás características, ya que no todos se pueden identificar de la misma manera.	proceso y llevar controles de mercadería en una aplicación. Como se puede observar en el Anexo C, la entidad maneja una lista en Excel de los productos existentes.
13	¿Existe alguna restricción de ingreso de mercadería?	Productos en mal estado se pueden devolver con el transportista que los trajo. Se podría implementar restricciones como productos que no se han vendido en x tiempo no comprarlos	
14	¿Se tiene de alguna forma identificados los productos que más flujo tienen en la empresa?	Si, mediante facturación.	
15	¿Se encuentran identificadas las áreas de los productos de alguna manera?	No, únicamente por el tipo de producto se van poniendo en su lugar	
16	¿Al recibir productos se toma en cuenta el estado y ensamblaje en que llegan?	Si, se realiza una revisión al llegar los transportistas y si todo está acorde a lo pedido se aceptan los productos.	
Preguntas Contables			
1	¿Cuál es la forma de cálculo del precio de los productos?	Varia depende el producto, se les asigna un porcentaje de ganancia y en otros se le aumenta un valor aproximado.	Se deben definir los precios de los productos y estos pueden ser
2	¿Existen productos con y sin IVA? ¿Cuáles son?	La mayoría de los productos tienen IVA 12%, a excepción de algunos productos agrícolas como palas, ozes y madera.	

3	Formato de factura.	Contiene un encabezado con el número de factura que proporciona SRI y datos de la empresa. Le siguen los datos del cliente y por último los productos vendidos, es bastante simple.	variables al momento de facturar.
---	---------------------	---	-----------------------------------

*Tabla 8: Recolección de información - Entrevista
Elaborado por: Investigador*

La observación de campo se realizó el jueves 2 de junio de 2022, su objetivo fue complementar información a la entrevista y realizar la descripción completa del lugar y las funciones de las personas involucradas en las actividades. Una vez realizada, se obtuvo la siguiente información:

Tema de investigación:	Recolección de información adicional sobre los procesos en el control de inventario en Ferrimar
Datos del Investigador:	Camila Martinez
Lugar:	Ambato, instalaciones de Ferretería Ferrimar
Fecha y Hora:	Junio 2, 2022 - 09:30
Desarrollo	
¿Qué persona realiza la facturación?	No todas las personas en la empresa pueden realizar facturas de venta solo el gerente o el empleado.
¿Cuántas personas se encuentran?	El contador no se encuentra en las instalaciones, únicamente acude los fines de mes y cuando se requiera realizar algún trámite o revisión de cuentas.
	Se encuentran tres personas, el gerente, un empleado y un ayudante de bodega. Los empleados se van turnando según el trabajo que se tenga planificado ese día.
¿Dónde se lleva el control de productos?	Las anotaciones de material faltante o defectuoso se van realizando en un cuaderno, Las facturas de compra y venta se almacenan en carpetas y se separan por meses, para después entregarlas al contador.
¿Cómo se realiza un pedido o proforma?	Los clientes pueden realizar pedidos con anticipación y este material es registrado en una proforma para posteriormente, ser facturado.

¿Se manejan tarifas extras en la facturación?	Cuando el cliente pide que el material sea enviado a algún lugar, se cobra una tarifa adicional por flete.
¿Se manejan descuentos facturación?	Para clientes exclusivos por compras al por mayor, o a criterio del gerente se manejan descuentos en algunos productos.
¿Qué se hace con las facturas anuladas?	Existe la posibilidad de anular facturas de venta, esto se da en la mayoría de los casos porque se calcularon mal los precios. La factura se debe tachar y se deja en el facturero.
¿Existen servicios adicionales?	Puede ingresar o salir mercadería sin el documento de factura, se da en casos puntuales. Por ejemplo: en ingreso cuando llega mercadería de promoción que no está detallada en las facturas de compra. Y en salida cuando se regalan productos por una compra.
Conclusiones	
Cada uno de los empleados de Ferrimar tienen establecidas sus funciones, sin embargo, todos son procesos manuales que son susceptibles a errores humanos. Cuentan con un listado donde se encuentran todos los productos que se maneja, sin embargo, no está actualizado en cuanto a cantidades debido a que no se lleva un control periódico.	

*Tabla 9: Recolección de información – Observación de campo
Elaborado por: Investigador*

En la recolección de información se obtuvo:

- Muestra de factura de compra. (Anexo A, B)
- Muestra de factura de venta. (Anexo C)
- Muestra de proforma. (Anexo D)
- Listado actual del inventario. (Anexo E)

Ficha de Contenido			
Número:	1	Categoría:	Factura de compra 1
Descriptores			
<p>Proveedor: MEGAKONS S.A</p> <p>Numero de Factura: 001-003-000223301</p> <p>Campos de detalle compra: Contiene el código del producto, detalle, unidad, cantidad, precio unitario, descuento y total</p> <p>Cálculo de totales: Contiene el cálculo de tarifa 0%, tarifa 12%, la suma de descuentos, subtotal, IVA 12% y total a pagar.</p> <p>Adicionales: La factura tiene campos para firmar como entregada la factura y un campo adicional llamado OBSERVACIONES.</p>			
Observación			
<p>Se debe tener en cuenta como la empresa calcula el total de compra ya que este valor depende de varios cálculos previos.</p> <p>Ajustar los campos que poseen los productos para poder ingresar de manera homogénea las facturas de compra.</p>			

*Tabla 10: Recolección de información – Ficha de Contenido No. 1
Elaborado por: Investigador*

Ficha de Contenido			
Número:	2	Categoría:	Factura de compra 2
Descriptores			
<p>Proveedor: IMPORTADOR FERRETERO TRUJILLO CIA.LTDA</p> <p>Número de Factura: 006-001-000042609</p> <p>Campos de detalle compra:</p>			

<p>Contiene código del producto, cantidad, unidad, bodega, descripción, precio, descuento, precio con descuento aplicado, valor de descuento y subtotal</p> <p>Cálculo de totales:</p> <p>La empresa calcula el total de la factura con los campos venta con tarifa 0%, venta con tarifa 12%, IVA 12% y total.</p> <p>Adicionales:</p> <p>Contiene los campos para firma del destinatario y el despachador.</p>
Observación

*Tabla 11: Recolección de información – Ficha de Contenido No. 2
Elaborado por: Investigador*

Ficha de Contenido	
Número:	3
Categoría:	Factura de venta
Descriptores	
<p>Datos del cliente:</p> <p>Se requiere los campos nombres, apellidos, dirección, RUC o cédula de identidad, teléfono y guía de remisión.</p> <p>Número de Factura:</p> <p>001-001-0010273</p> <p>Campos de detalle compra:</p> <p>Contiene la cantidad, descripción, valor unitario y valor total</p> <p>Cálculo de totales:</p> <p>Contiene el cálculo de subtotal de tarifa 12%, subtotal de tarifa 0%, descuento, valor por servicio de transporte, total IVA 12% y valor total de la venta.</p> <p>Adicionales:</p> <p>Contiene un casillero para indicar la forma de pago del cliente, firmas de entrega y recepción de la factura.</p>	
Observación	
<p>El gerente indica que la empresa no hace uso del campo de forma de pago, y tampoco del descuento y tarifa de servicio de transporte, ya que estos son calculados dentro del detalle de la misma factura. En cuanto a los datos que se encuentran en el</p>	

encabezado de factura tampoco hacen uso del número de guía de remisión debido a que el local no maneja ese tipo de documento.

*Tabla 12: Recolección de información – Ficha de Contenido No. 3
Elaborado por: Investigador*

Ficha de Contenido			
Número:	4	Categoría:	Proforma
Descriptores			
Datos del cliente: Se requiere los campos nombres, apellidos, dirección, RUC o cédula de identidad y número de teléfono.			
Número de Proforma: 000 001 564			
Fecha: 07-Jun-2022			
Campos de detalle compra: Contiene los campos cantidad, descripción, valor unitario y valor total			
Cálculo de totales: Contiene el campo de tarifa de envío, descuento y valor total.			
Adicionales: Contiene un logotipo, el cual no coincide con el que se encuentra en las facturas de venta que se generan.			
Observación			
El gerente informa que las proformas se generan mediante una plantilla de Word y que no es muy usual que los clientes pidan estas. Adicional se debe solicitar o generar un logotipo para hacer uso de este tanto en el sistema con en los documentos que se genere y así mantener estandarizadas las presentaciones.			

*Tabla 13: Recolección de información – Ficha de Contenido No. 4
Elaborado por: Investigador*

Ficha de Contenido			
Número:	5	Categoría:	Inventario

Descriptores
Posee los campos categoría, marca, nombre, medidas, unidad de medida, cantidad disponible, precio de venta, total.
Observación
Las categorías sirven para filtrar y buscar de una mejor manera los productos. El listado no incluye el precio de compra del producto debido a que estos cambian constantemente.

*Tabla 14: Recolección de información – Ficha de Contenido No. 5
Elaborado por: Investigador*

2.2.4. Procesamiento y análisis de datos

En base a los resultados de la aplicación de los instrumentos de recolección de información, se pudo concluir que:

- Los procesos que manejan y modifican la cantidad de productos y stock en la ferretería Ferrimar son las compras y las ventas. Estos procesos se realizan de manera manual, y las cantidades se van sumando o restando del archivo que manejan como inventario, por lo que se requiere automatizar con ayuda de software para reducir tiempo de operación y errores de cálculo.
- Al llevar el registro manual y poco periódico del inventario, se comenten errores de cálculo y la información que se posee no es confiable.
- La ferretería puede ingresar mercadería sin un número de factura debido a que en ocasiones llegan productos como parte de promociones o regalo. De igual manera puede disminuir mercadería como parte de donaciones o para uso dentro de la misma.
- Se considera necesario llevar un control de stock y precios, tanto de compra como de ventas para tener al día la información de capital que se encuentra.
- Se requiere de una aplicación que sea accesible desde distintos dispositivos ya que no todos los usuarios poseen el mismo.
- Se requiere definir roles de usuario con los distintos permisos que tendrá cada uno, debido que no todos realizan las mismas actividades.
- El gerente/propietario de la Ferretería Ferrimar necesita visualizar reportes que ayuden en toma de decisiones, ya que al momento no se tiene la posibilidad de analizar el estado de los productos que se posee en stock.

CAPITULO III

RESULTADOS Y DISCUSION

3.1. Análisis y discusión de resultados

Framework para desarrollar Aplicaciones Web Progresivas (frontend)

Un framework es una herramienta que facilita el desarrollo de páginas web, permitirá crear un proyecto en menos tiempo y con un código más limpio. Contiene una especie de plantilla con la estructura base de un aplicativo web, y a partir de ahí el programador pueden ir modificando y añadiendo detalles a su esqueleto.

Lo que ofrece la mayoría de frameworks es un patrón de diseño para el desarrollo, en la mayoría se trata del patrón MVC (Modelo-Vista-Controlador), dicho patrón divide el desarrollo en tres capas. Donde el modelo maneja los datos e información del aplicativo, junto con sus reglas de negocio. Segundo, la vista, que representa las interfaces y como se presentan los datos a los usuarios. Y, por último, el controlador que es la capa en la que se procesan peticiones y consultas, es decir aquí se realiza el intercambio de información. [25]

Para poder escoger un framework, como desarrollador se debe tomar en cuenta aspectos como la demanda en el mercado laboral, la durabilidad, y el ajuste al trabajo que se requiera. En la presente investigación se analizarán 3 de ellos que según Daniel Ortega Delgado [26], se encuentran en el top seis de frameworks más solicitados.

Angular

Es un framework basado en JavaScript. Fue lanzado por Google en el año 2009 y se considera una de las herramientas más estables, ya que no ha sufrido cambios críticos en las actualizaciones de los últimos cinco años. Se basa en un modelo MVC y da la posibilidad de manipular los datos del lado del cliente. [27]

Usa un sistema de módulos que, pese a que no son estrictamente obligatorios, es muy recomendado utilizarlos, pueden ayudar al programador a separar el código en diferentes archivos, haciendo así software más manejable.

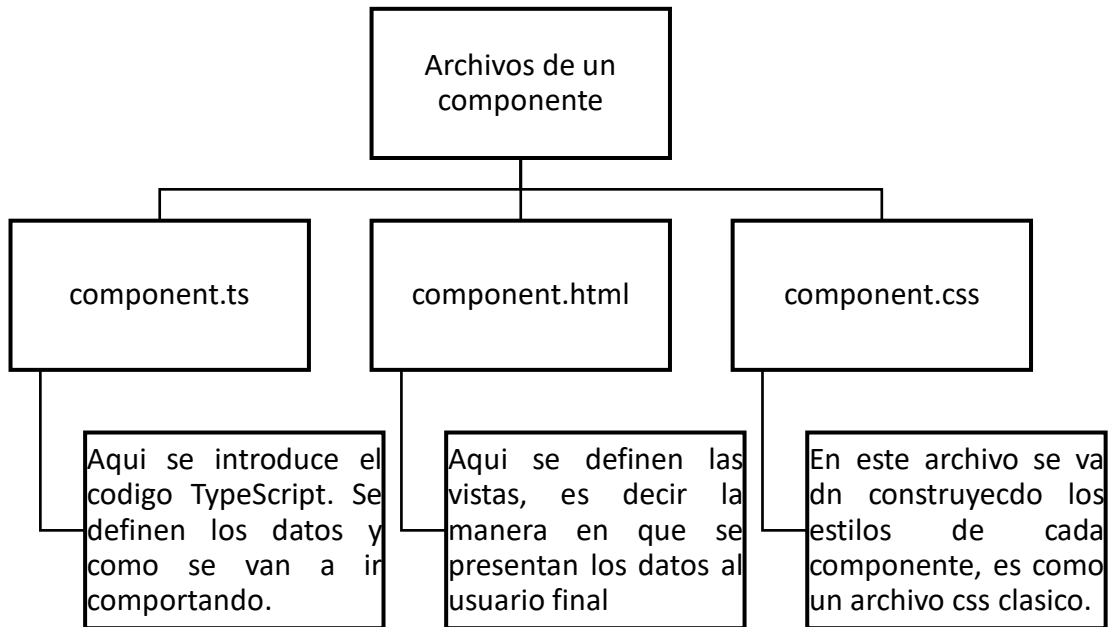
Su característica más innovadora es que permite al usuario actualizaciones de las aplicaciones web a tiempo real entre diferentes dispositivos. [26] Además, por su estabilidad y la gran acogida que ha tenido a lo largo del mundo es el framework con más apoyo comunitario en web, y soporte.

Tiene algunas otras características como:

- Comúnmente es utilizado para desarrollar aplicaciones multiplataforma.
- Angular está diseñado para que el programador sea forzado a escribir código limpio. Esto conlleva a que el código sea más lógico y pueda ser reutilizable.
- Al tener el respaldo de Google se va posicionando entre los frameworks más robustos.
- Tiene la función de enlace de datos bidireccional entre el modelo y los componentes de la vista.
- Además, tiene una característica en la que se pueden crear etiquetas HTML personalizadas que pueden actuar como pequeños widgets, esto tiene el nombre de directivas [28]

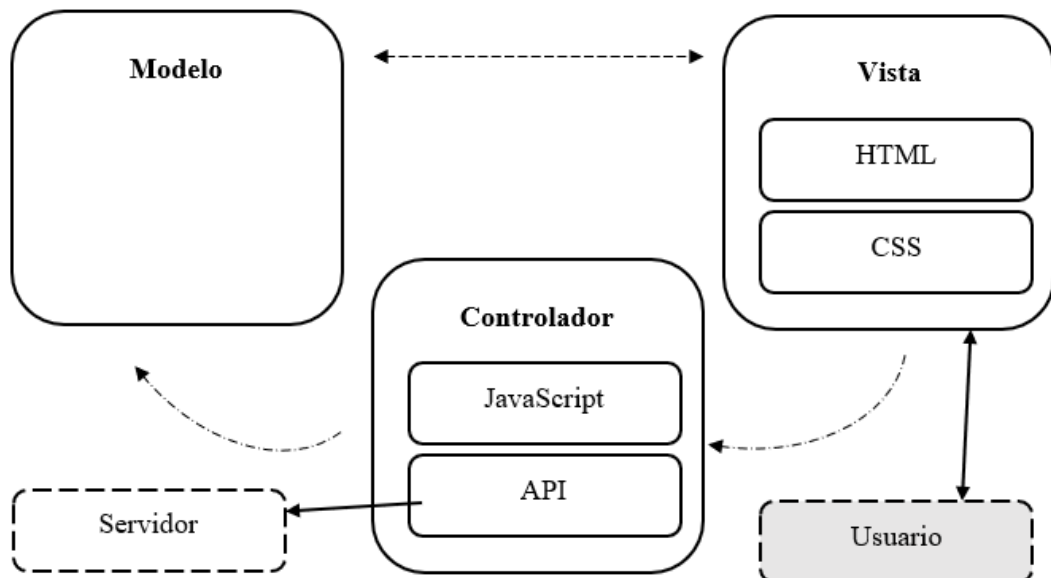
Arquitectura de angular

Los elementos principales de angular se llaman componentes, los cuales se ordenan con tres archivos principales.



*Imagen 3: Archivos de un componente Angular
Elaborado por: Investigador*

Ya descritos los componentes, se deduce que angular trabaja con el modelo MVC (Modelo-Vista-Controlador), funcionando de la siguiente manera:



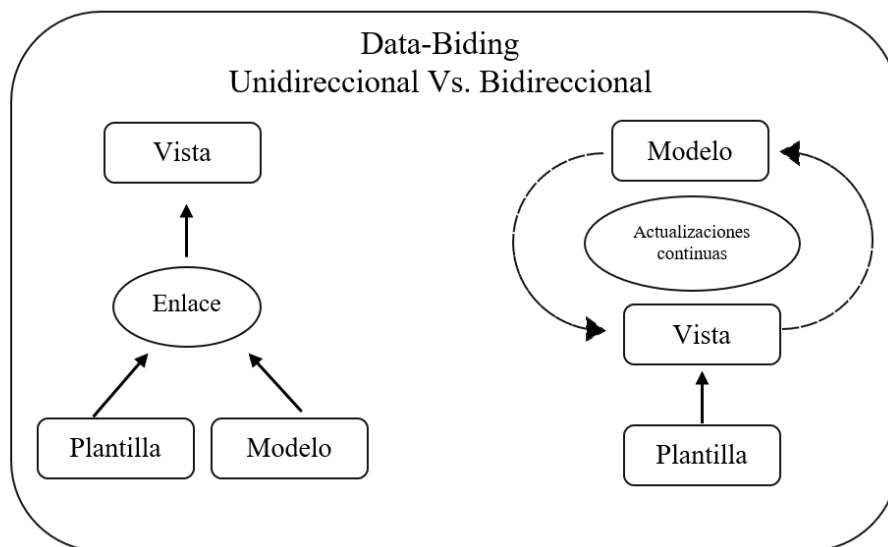
*Imagen 4: Arquitectura MVC Angular
Elaborado por: Investigador*

Ember JS

Es un marco cliente de código abierto lanzado en 2011, al igual que los demás y permite a desarrolladoras crear aplicaciones y páginas web basadas en HTML. Entidades como Apple o GitHub han trabajado con esta herramienta [29]. Proporciona gestión de datos y flujo de operaciones.

Contiene componentes que se dividen en dos partes JavaScript, la primera es una plantilla y la segunda es un archivo que describe el comportamiento de los componentes. Además, proporciona gestión en todas sus versiones y herramientas de actualizaciones junto a potentes guías, herramientas y documentación.

Su arquitectura se adapta a MVC. Por lo que permite crear aplicaciones escalables y de fácil mantenimiento. Ésta también ofrece un data-biding bidireccional que, al contrario del unidireccional, éste lo que hace es la sincronización automática de datos entre el modelo y la vista y viceversa, en la siguiente Imagen se comparan los dos modelos de data-biding.



*Imagen 5: Data-Binding unidireccional y bidireccional
Elaborado por: Investigador*

La diferencia con Angular es que Ember obliga al desarrollador a escribir código dentro de su propia estructura, es decir se deben seguir sus directrices de programación, desde la nomenclatura de las vistas y controladores, esto se puede considerar como un punto en contra para escoger esta herramienta, pese a que está diseñado así con el fin de que el mismo se encargue de generar automáticamente los objetos cuando se requieran. Ember puede generar controladores, modelos y vistas junto a su comportamiento implícitamente. [30]

No obstante, tiene ciertas desventajas:

- Es un framework más pesado
- Puede resultar dificultoso de aprender por lo que no se recomienda para proyectos pequeños
- No existen actualizaciones y mejoras constantes por lo que se debe esperar bastante tiempo para una nueva versión

Vue JS

Vue es un marco JavaScript ‘progresivo’ esto quiere decir que con Vue se puede crear todo tipo de desarrollos. Fue lanzado en 2014, que se destaca debido a que se apoya en una API simple, por lo que es un framework más ligero que angular, sin embargo, es capaz de crear interfaces y elementos dinámicos muy elaboradas. También ofrece libertad para utilizar módulos y librerías de apoyo para desarrollo web. Al igual que los frameworks mencionados anteriormente, posee un data-binding bidireccional, por lo cual la parte visual de la página se actualizará sin que el desarrollador deba intervenir de manera manual.

Se considera Vue como un framework sencillo, con una curva de aprendizaje baja, esto le ha dado una gran acogida sobre todo en países asiáticos como China, por lo que la mayoría de las comunidades de soporte de este framework está en idioma chino. Empresas como Xiaomi y AliBaba se han acogido a este framework.

Al ser un framework creado recientemente y al ser independiente no cuenta con el soporte de empresas grandes como Google o Facebook y está es la razón por la cual es alejado de grandes proyectos, a pesar de que cuenta con un servicio eficiente.

de interactuar con la base de datos, recibe las peticiones y contiene la lógica para poder devolver los datos correctos a la parte de cliente. [31]

Características de VueJS

- Reactividad: Ofrece componentes visuales de forma reactiva. Estos componentes reaccionan ante eventos sin que el rendimiento sea afectado
- Su API es fácil de utilizar
- Ocupa un DOM Virtual, es decir que utiliza una copia en caché y la va modificando según necesite
- Sigue un flujo unidireccional de data-binding para la comunicación entre componentes, y bidireccional para la comunicación entre modelos.
- Tiene soporte para TypeScript, y también se puede extender mediante plugins
- Plantillas: admite archivos JSX o HTML válidos para utilizarlas como plantillas.

El número de tecnologías que ofrecen este servicio son variadas, tal puede ser el caso de lenguajes como PHP junto a un framework por ejemplo Laravel o symfony, Python con Django, .NET, JavaScript con NodeJs etc.

La idea de separar en partes un proyecto es mantener funciones y código separado, llevando así una programación más limpia y de mayor facilidad de mantenimiento, de esta manera el frontend se encarga de recibir las peticiones, y el backend de procesarlas.

Análisis comparativo entre frameworks de frontend

		Angular	Ember	VueJS
Características	Routing	Requiere de un controlador o plantilla base.	Posee un enrutamiento más robusto.	Requiere de una librería para poder implementar rutas

	Tipo de proyectos	Cualquier tipo de aplicación web y PWA	Se usa en páginas modernas, tanto de escritorio como móviles.	Aplicaciones de frontend
	Arquitectura	Modelo-Vista-Controlador	Modelo-Vista-Componente	Modelo-Vista-Controlador
	Lenguaje	JavaScript, TypeScript	JavaScript	Javascript
	Documentación	Buena	Buena	Buena

*Tabla 15: Comparación de framework de frontend
Elaborado por: Investigador*

Determinación de framework de desarrollo frontend.

Al realizar el análisis entre los frameworks de frontend opensource, planteados en la *Tabla 15* del presente capítulo y las actividades que se van a desarrollar, se ha determinado que el framework que se utilizará en el presente proyecto es Angular, ya que se ajusta tanto a las necesidades del cliente que es brindar una aplicación multiplataforma, como del desarrollador que es utilizar un framework con una curva de aprendizaje baja y código reutilizable.

La ventaja que presenta sobre los demás frameworks es que es escalable y se puede convertir fácilmente una aplicación angular a una PWA, que es lo que se busca y el objetivo principal del proyecto. Usa programación TypeScript, la cual es una extensión de JavaScript que agrega características importantes a las aplicaciones como promesas, o programación asíncrona. Además, tal ser una tecnología que ha ganado bastante popularidad, posee una gran cantidad de documentación y soporte en línea.

Por otra parte, para el diseño de interfaces se utiliza HTML y CSS, lo cual hace que código sea reutilizable, y de fácil actualización y mantenimiento.

Framework de desarrollo backend

El backend de una aplicación es el conjunto de lógica y programación que trabajan del lado del servidor. El objetivo de este es que el proceso de desarrollo de software sea más acelerado, separando así el código de cliente con el de servidor. El backend se encarga de realizar peticiones, obtener y enviar información a la base de datos, así como también acciones de lógica de negocio.

Laravel

Se trata de un framework basado en PHP, el cual brinda herramientas y recursos variados para el desarrollo de aplicaciones modernas. En los últimos años ha crecido bastante su comunidad y popularidad. Facilita las tareas más habituales en proyectos de pequeño, mediano y gran alcance como lo es el enrutamiento, autenticación de seguridad, almacenamiento, etc.

Django

Es uno de los frameworks más populares, es gratuito y de código abierto. Se basa en lenguaje Python. Maneja un conjunto de formularios los cuales hacen que sea idóneo para el desarrollo *full-stack*. Por otra parte, sus componentes son sustituibles e intercambiables para facilitar la escalabilidad.

NodeJS

NodeJS fue creado por desarrolladores de JavaScript, fue diseñado con un modelo de entrada y salida controlado por eventos para hacerlo más eficiente. Al igual que las tecnologías analizadas anteriormente, esta también cuenta con una gran comunidad y ha tenido mucha acogida en el desarrollo de software. Está en constante actualización de características y permite crear REST APIs robustas, donde los usuarios pueden configurar rutas y el intercambio de solicitudes entre frontend y la base de datos.

Análisis comparativo entre frameworks de backend

A continuación, en la *Tabla 9* se analizarán alguna de las características de las tecnologías OpenSource de backend más utilizadas. [32]

	Laravel	Django	NodeJS
Lenguaje de programación	PHP	Python	JavaScript
Año de lanzamiento	2011	2005	2009
Curva de aprendizaje	Alta	Baja	Media
Arquitectura	Modelo Vista Controlador (MVC)	Modelo Vista Template (MVT)	Modelo Vista Controlador (MVC)
Soporte y Documentación	Si	Si	Si
Ventajas	Plataforma full-stack	Alta seguridad y actualizaciones	Funciona también como servidor
	Fácil trazabilidad en errores de código	Modelo de búsquedas en base de datos sencillo	Se pueden crear aplicaciones tiempo real

*Tabla 16: Comparación de framework de backend
Elaborado por: Investigador*

Es necesario establecer un lenguaje de backend acorde al sistema que se va a desarrollar ya que será el responsable de comunicar los datos entre la base de datos y la aplicación de usuario. Básicamente hace que la lógica de una página web funcione.

Determinación de framework de desarrollo backend.

Al realizar el análisis entre los frameworks de backend, planteados en la *Tabla 16* y los requerimientos del sistema, se estableció que se trabajará con tecnología NodeJS. El hecho de que su código se pueda programar en JavaScript y estar utilizando este

mismo lenguaje de lado de cliente hará que la transferencia de datos entre estos puntos sea más rápida y por lo tanto reduce los tiempos de trabajo. En adición tiene mayor popularidad que las demás opciones, por lo que cuenta con mayor documentación, soporte, etc., lo cual será de gran apoyo y ayuda para la resolución de problemas que se presenten durante la programación.

Base de datos

Una base de datos es un conjunto organizado de información que se almacenada de una forma relacionada y estructurada, para garantizar la integridad de datos. Se componen de un lenguaje de definición de datos, lenguaje de manipulación de datos y el lenguaje de consulta.

Por otra parte, los gestores de bases de datos son programas que permiten administrar los accesos a las bases de datos, con la finalidad de servir como una interfaz entre el usuario y la información.

SQL Server

Características

SQL Server es una base de datos relacional de código cerrado, se vende bajo licencia ya que es desarrollada por la empresa Microsoft. Es apta para para entornos empresariales que utilizan .NET. Inicialmente era compatible solo con Windows, pero debido a su gran acogida, se lanzó una versión para Linux.

Al ser una herramienta destinada para utilizar en un ámbito empresarial posee varios sistemas avanzados de análisis, informes y reportes de bases de datos. Su mayor característica para destacar es que ofrece una alta disponibilidad, ya que permite un mayor tiempo de actividad continua y a la vez conmutación de datos más rápida.

MySQL

Características

Es una base de datos relacional, basada en SQL, es de código abierto y actualmente pertenece a Oracle. Se desarrollo a mediados de los años noventa, por lo que es uno de los gestores con mayor trayectoria en el mercado, también está calificado como una de las bases de datos de código abierto más popular del mundo.

Es multiplataforma, por lo que está disponible para usar en Windows, MacOS, Linux, Solaris, etc. Es muy utilizado para proyectos basados en PHP. Está programada en su mayoría en lenguaje C y C++.

Es una herramienta sencilla y fácil de instalar, y al ser de código abierto, posee gran cantidad de documentación y soporte en línea.

PostgreSQL

Características

Es un sistema de base de datos relacional de código abierto y totalmente gratuito. Con PostgreSQL es posible ejecutar con tanto como consultas relacionales en SQL, como consultas no relaciones, las cuales hacen uso de JSON.

Una de sus más grandes características es la capacidad que posee para el manejo de grandes volúmenes de datos, lo cual le brinda una ventaja sobre el resto de bases de datos. Tiene también compatibilidad total con el sistema ACID (atomicity, consistency, isolation y durability)

Análisis comparativo de Bases de Datos

	MySQL	SQL Server	PostgreSQL
Licencia	Open Source, es gratuito, aunque también posee una versión empresarial, la cual si tiene costo y herramientas adicionales	Es de pago bajo licencia	No tiene ningún costo de licencia, cualquier persona es libre de descargar y darle uso sin ningún costo

Sistemas Operativos compatibles	FreeBSD Linux OS X Solaris Windows	Linux, Windows	Compatible con distintos sistemas operativos y servidores web como Apache
Motores	Se puede utilizar distintos motores como InnoDB o MyISAM	Únicamente utiliza el motor que brinda Microsoft	Existen varios motores que brindan el servicio de manejar bases de datos hechas en PostgreSQL
Relaciones entre tablas	Se relacionan con el uso de llaves primarias y foráneas (Tablas relacionales)	Se relacionan con el uso de llaves primarias y foráneas (Tablas relacionales)	Se puede manejar bases de datos relacionales y no relacionales
Transacciones, vistas, procedimientos almacenados, triggers	Si	Si	Si
Reportes	No	Si	Necesita de herramientas adicionales para generar reportes

*Tabla 17: Comparación de bases de datos
Elaborado por: Investigador*

Determinación de Gestor de base de datos.

La base de datos que se utilizará será MySQL, la cual se adapta al presente proyecto al ser relacional, y es uno de los gestores de base de datos más populares y gratuita.

Ofrece las características necesarias para proyectos profesionales. Asegura la integridad y confiabilidad de datos, tiene una instalación sencilla y no consume muchos recursos del servidor.

Metodologías de desarrollo

La Metodología de desarrollo de software es un conjunto de prácticas y métodos que permite el cumplimiento de todas las etapas del ciclo de vida de un software y alcanzar un objetivo. Es importante elegir una metodología acorde al proyecto debido a que ésta establece las actividades a seguir y la manera en que se deben realizar.

Existen dos tipos de metodologías: las ágiles y las tradicionales y su diferencia radica en la planificación que requiere cada una, mientras que las metodologías tradicionales se centran en la definición y preparación de costos, alcances y tiempos, las metodologías ágiles tienen un enfoque de trabajo en equipo y la presentación de buenos resultados, esta última también es más flexible ante cambios. [33]

Scrum

Scrum se puede aplicar en proyectos complejos, que se desarrollarán en entornos dinámicos y variables. Es idóneo para el trabajo colaborativo en equipos, en los cuales se pueden dar cambios y deben ser resueltos con rapidez. Hoy en día no solo se aplica para el desarrollo de software sino en otras ramas como en educación, marketing, política, etc.

Extreme Programming

Es una metodología ágil creada a finales de los años noventa exclusivamente para el desarrollo de software. Se basa en el seguimiento constante y la priorización de actividades y su principal objetivo es realizar ciclos de entrega veloces y eficaces. Aquí se realizan revisiones de código y pruebas constantes para lograr una buena

adaptación. Se asemeja con Scrum ya que se organizan en sprints basadas en historias de usuarios, sin embargo, Extreme Programming (XP) es más rígido.

Kanban

Es plenamente una metodología visual, donde se puede organizar los procesos y los trabajos continuos. Los actores pueden visualizar el progreso de las tareas hasta que son finalizadas a tiempo real. El tablero más sencillo se compone de los estados ‘Solicitado’, ‘En curso’, ‘Hecho’. Uno de sus principios básicos es gestionarse en el trabajo, mas no en los trabajadores, significa que se enfocaran en la entrega oportuna al cliente, comprender con exactitud sus necesidades, esto dará como resultado la calidad de servicios prestados.

Análisis comparativo entre metodologías de desarrollo

A continuación, se presenta una tabla comparativa con las principales metodologías ágiles junto con sus características:

	Scrum	Extreme Programming (XP)	Kanban
Método de trabajo	Se divide el proyecto en bloques o sprints de un tiempo especificado y se revisan continuamente	Se divide el proyecto en fases en las que se debe completar un ciclo completo de análisis, desarrollo y pruebas	Se utilizan técnicas visuales para el seguimiento de las actividades (las principales son: pendientes, en proceso y culminadas)

Enfoque	Ejecución de sprints	Procesos incrementales	Gestión de tareas y desarrollo incremental
Tamaño de proyecto	Aplicable en todos	Proyectos medianos y grandes	Medianos y grandes
Roles	Scrum Master Product Owner Agile Coach	Cliente Programador Encargado de Pruebas (Tester) Encargado de Seguimiento (Tracker) Entrenador (Coach) Gestor (Big Boss)	Service Request Manager Service Delivery Manager

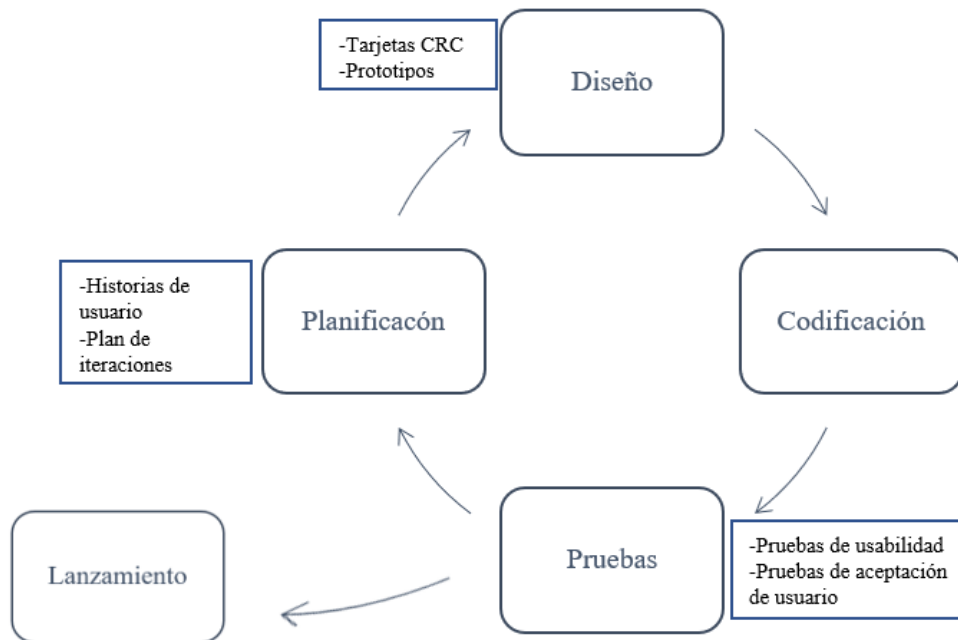
*Tabla 18: Comparación de metodologías ágiles
Elaborado por: Investigador*

Determinación de metodología de desarrollo.

Después de realizar la comparación de características de las metodologías ágiles más utilizadas en la *Tabla 12*, se concluye que en el presente proyecto se aplicará a la metodología Extreme Programming (XP). Para lo cual se deberá dividir el proyecto en pequeñas iteraciones o fases en las que se de completar un ciclo de análisis desarrollo y pruebas, esto ayuda a la revisión y retroalimentación de los avances, además de la corrección a tiempo de los errores. Es adecuado para el proyecto ya que posee iteraciones en las que se va revisando junto con la persona con rol de tester y el coach. Además, los avances y se pueden ir corrigiendo durante el proceso de desarrollo.

3.1.1. Desarrollo de la propuesta

Según la metodología XP se debe cumplir con un conjunto de fases que son: planificación, diseño, codificación, pruebas, lanzamiento. En cada una de estas fases también hay que cumplir una serie de pasos para que todas las etapas de desarrollo de software se cumplan y se maneje de manera ordenada el proyecto.



*Imagen 6: Diagrama de fases de la metodología Extreme Programming
Elaborado por: Investigador*

Ferretería Ferrimar brindó todas las facilidades para realizar un levantamiento de información adecuado y con esto, obtener los requerimientos necesarios.

3.1.1.1. Fase de planificación

En la fase de planificación, los usuarios de Ferrimar plantean a grandes rasgos los roles de usuario, que son de interés para la primera entrega del producto. A su vez, el desarrollador se familiariza con las herramientas, tecnologías y prácticas que se

utilizarán según el análisis del proyecto. Aquí construirá un prototipo de entre las posibilidades de arquitectura de la aplicación.

Levantamiento de la información

Con la entrevista y la observación de campo aplicada a los empleados de Ferrimar se logró conocer los procesos que manejan y la manera en que se desenvuelven. Se llegó a la conclusión que manejan un stock manual, por lo cual pueden existir errores humanos, atrasando las actividades. Esto se podrá automatizar de la manera planteada desde realizar un listado de pedidos hasta facturar una proforma.

Definición de roles

Para aplicar la metodología XP, se requiere definir de manera clara los roles y funciones dentro del proyecto.

Rol	Nombre	Función
Programador	Camila Martinez	Responsable en desarrollar la Aplicación Web Progresiva, cumplirá con las exigencias de los usuarios.
Coach	Ing. Carlos Núñez	Responsable de realizar el seguimiento de procesos y su cumplimiento
Big Boss	Francisco Martinez	Es el gerente del establecimiento. Es responsable de la coordinación y las revisiones de avances.

Tester	Francisco Martinez	Responsable de validar el aplicativo, ejecutar pruebas regularmente y difundir los resultados.
--------	--------------------	--

*Tabla 19: Definición de roles de usuario
Elaborado por: Investigador*

Modelado de procesos en el manejo de inventario en Ferrimar

Tras el análisis se concluye que, para la correcta administración del inventario en la ferretería Ferrimar, se debe tomar en cuenta dos procesos que son: la entrada y salida de productos. A continuación, se procederá a modelar los procesos mencionados.

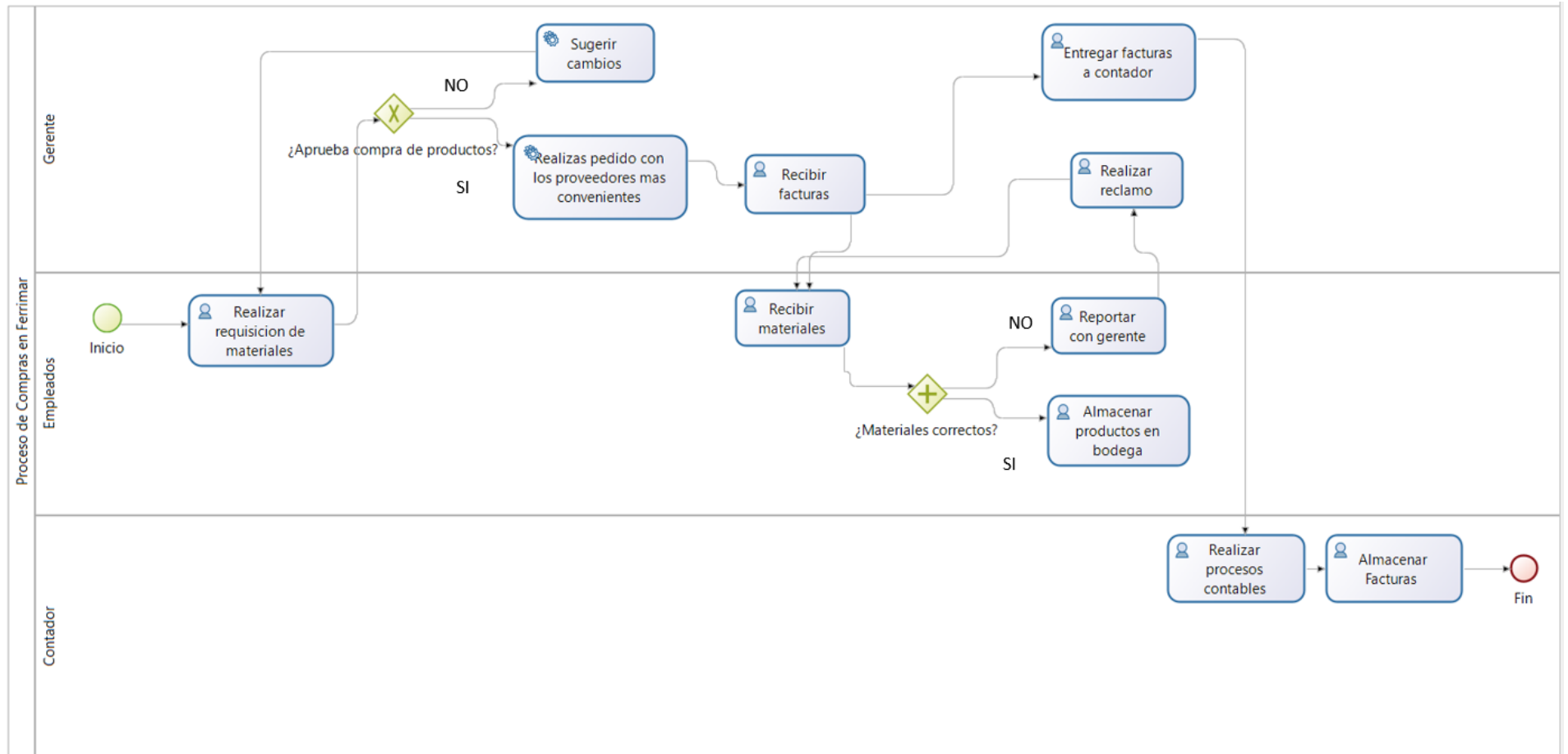


Imagen 7: Diagrama de proceso de Compras a proveedores en Ferrimar
Elaborado por: Investigador

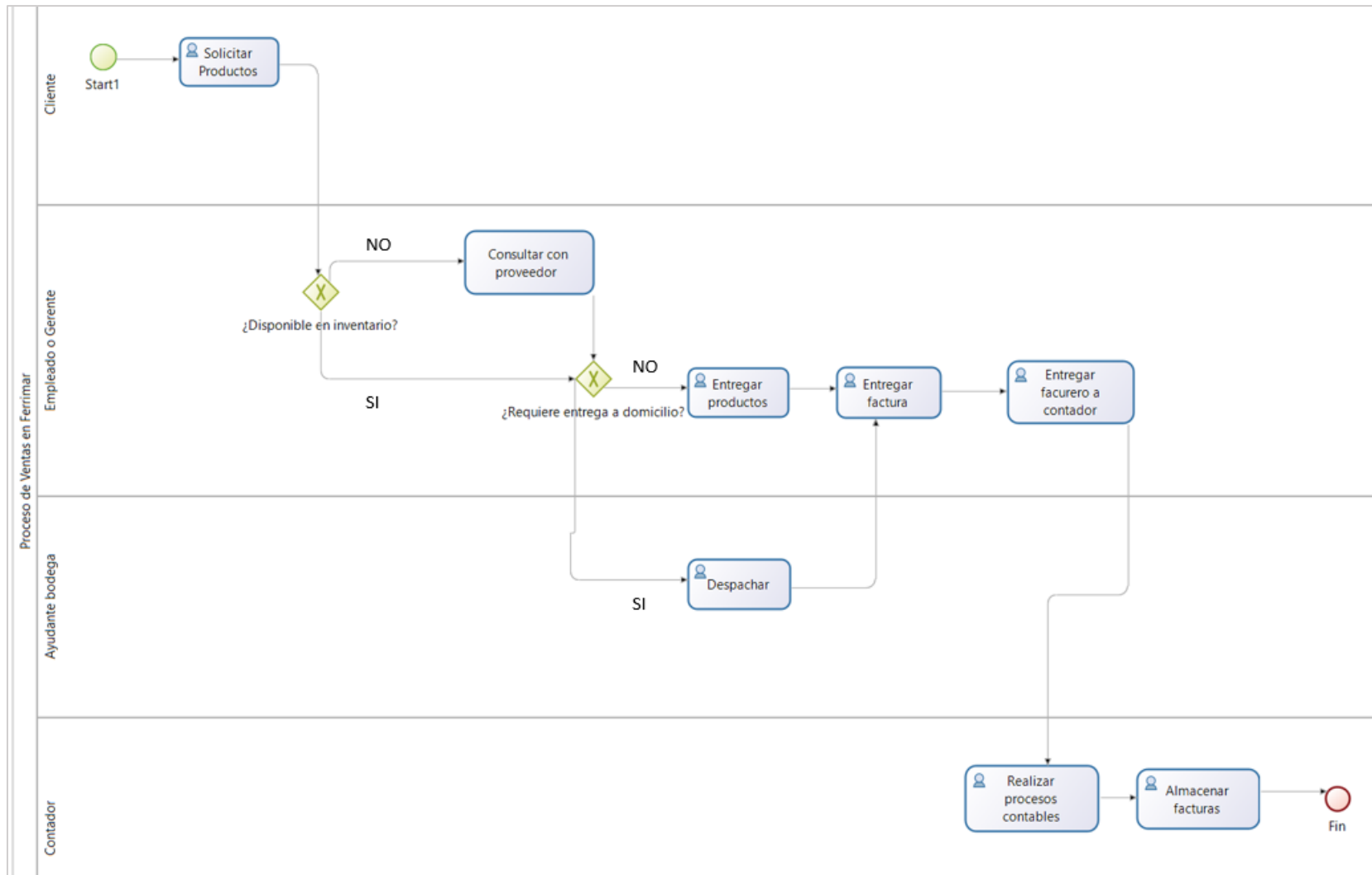


Imagen 8: Diagrama de proceso de Ventas en Ferrimar
Elaborado por: Investigador

Historias de usuarios

Las historias de usuarios son a breves rasgos, explicaciones generales de cada función o modulo del software, contiene datos como prioridad, riesgo, responsable, etc. Toda esta información es necesaria para satisfacer las necesidades del usuario final.

Para la elaboración de cada una de las historias de usuario, se utilizará el siguiente modelo:

Historia de Usuario	
Numero:	Usuario:
Nombre de historia:	
Prioridad en el negocio:	Puntos estimados
Riesgo del desarrollo	iteración asignada:
Responsable:	
Descripción:	
Observación:	

*Tabla 20: Modelo de Historia de Usuario
Elaborado por: Investigador*

Donde:

- **Numero**

Es el número de identificación de cada historia.

- **Usuario**

Para que usuario/os va dirigida la historia.

- **Nombre**

Breve título de la historia de usuario

- **Prioridad en el negocio**

Valor comprendido entre Alto-Medio-Bajo para cualificar la prioridad que tendrá para el usuario.

- **Puntos estimados**

El número de días aproximados que en los que se llevará a cabo la tarea de la historia.

- **Riesgo del desarrollo**

Valor comprendido entre Alto-Medio-Bajo para cualificar el riesgo del desarrollo de la historia.

- **Iteración asignada**

Numero de iteración asignada a la historia

- **Responsable**

Persona asignada a realizar la tarea descrita.

- **Descripción**

Se detalla información del requerimiento.

- **Observación**

Detalles extras para tomar en cuenta.

Todas las historias de usuario realizadas para el presente proyecto se detallan a continuación.

Historia de Usuario	
Numero: 01	Usuario: Desarrollador
Nombre de historia: Análisis y modelamiento de la base de datos	
Prioridad en el negocio: Alta	Puntos estimados: 1
Riesgo del desarrollo: Alta	Iteración asignada: 1
Responsable: Camila Martinez	
Descripción: Se analizará y se modelará la base de datos en base a los actores e información que se requiere guardar.	
Observación: Ninguna	

*Tabla 21: Historia de Usuario 01
Elaborado por: Investigador*

Historia de Usuario	
Numero: 02	Usuario: Desarrollador
Nombre de historia: Definición de Roles y Usuarios	
Prioridad en el negocio: Media	Puntos estimados: 1
Riesgo del desarrollo: Media	Iteración asignada: 1

Responsable: Camila Martinez
Descripción: Se establecerán los permisos y acciones que podrán realizar cada uno de los actores dentro de la institución
Observación: Revisar junto con Gerente de Ferrimar, tabla de roles se encuentra en el <u>Anexo X</u>

*Tabla 22: Historia de Usuario 02
Elaborado por: Investigador*

Historia de Usuario	
Numero: 03	Usuario: Desarrollador
Nombre de historia: Interfaz inicio y general	
Prioridad en el negocio: Media	Puntos estimados: 2
Riesgo del desarrollo: Baja	Iteración asignada: 2
Responsable: Camila Martinez	
Descripción: Se diseña la interfaz general, los colores y estilos que se maneja en el aplicativo.	
Observación: Se debe solicitar el logo que maneja Ferrimar	

*Tabla 23: Historia de Usuario 03
Elaborado por: Investigador*

Historia de Usuario	
Numero: 04	Usuario: Desarrollador
Nombre de historia: Interfaz proveedores	
Prioridad en el negocio: Media	Puntos estimados: 2
Riesgo del desarrollo: Baja	Iteración asignada: 2
Responsable: Camila Martinez	
Descripción: Se diseñará una interfaz intuitiva para el usuario, que muestre los proveedores y según el tipo de permiso que posea, las opciones de editar, eliminar o adicionar un nuevo proveedor.	
Observación: Ninguna	

*Tabla 24: Historia de Usuario 04
Elaborado por: Investigador*

Historia de Usuario	
Numero: 05	Usuario: Desarrollador
Nombre de historia: Interfaz clientes	
Prioridad en el negocio: Media	Puntos estimados: 2
Riesgo del desarrollo: Baja	Iteración asignada: 2
Responsable: Camila Martinez	
Descripción: Se diseñará una interfaz intuitiva para el usuario, que muestre los clientes de Ferrimar y según el tipo de permiso que posea, las opciones de editar, adicionar un nuevo cliente.	
Observación: Los clientes no se pueden eliminar, solo se darán de baja para poseer un registro histórico del cliente.	

*Tabla 25: Historia de Usuario 05
Elaborado por: Investigador*

Historia de Usuario	
Numero: 06	Usuario: Desarrollador
Nombre de historia: Interfaz administración de productos	
Prioridad en el negocio: Media	Puntos estimados: 3
Riesgo del desarrollo: Baja	Iteración asignada: 3
Responsable: Camila Martinez	
Descripción: Debe ser una pantalla sencilla, donde se listen todos los productos y todas las características que estos poseen.	
Observación: El precio de los productos pueden ser variables.	

*Tabla 26: Historia de Usuario 06
Elaborado por: Investigador*

Historia de Usuario	
Numero: 07	Usuario: Desarrollador
Nombre de historia: Interfaces salida productos	
Prioridad en el negocio: Media	Puntos estimados: 3
Riesgo del desarrollo: Baja	Iteración asignada: 3
Responsable: Camila Martinez	

Descripción: Estas pantallas embarcarán facturación, salida sin documento y proformas de productos.
Observación: Al ser varias pantallas, se debe tomar en cuenta tiempo adicional en la planificación

*Tabla 27: Historia de Usuario 07
Elaborado por: Investigador*

Historia de Usuario	
Numero: 08	Usuario: Desarrollador
Nombre de historia: Interfaz de reportes	
Prioridad en el negocio: Media	Puntos estimados: 2
Riesgo del desarrollo: Baja	Iteración asignada: 4
Responsable: Camila Martinez	
Descripción: Se debe establecer los reportes que requiere la empresa y según eso, realizar el diseño de pantalla.	
Observación: Realizar junto el gerente y contador de Ferrimar.	

*Tabla 28: Historia de Usuario 08
Elaborado por: Investigador*

Historia de Usuario	
Numero: 09	Usuario: Desarrollador
Nombre de historia: Interfaz administración de usuarios	
Prioridad en el negocio: Media	Puntos estimados: 1
Riesgo del desarrollo: Baja	Iteración asignada: 4
Responsable: Camila Martinez	
Descripción: Solo el gerente, y el administrador del sistema tiene acceso al control de usuarios. Aquí se podrá adicionar, editar o dar de baja usuarios del sistema.	
Observación: Ninguna	

*Tabla 29: Historia de Usuario 09
Elaborado por: Investigador*

Historia de Usuario	
Numero: 10	Usuario: Desarrollador
Nombre de historia: Desarrollo de API proveedores	
Prioridad en el negocio: Alta	Puntos estimados: 3
Riesgo del desarrollo: Alta	Iteración asignada: 5
Responsable: Camila Martinez	
Descripción: Desarrollar la API REST de los proveedores con los métodos GET, POST, PUT para la interacción con la interfaz diseñada.	
Observación: Los proveedores no se eliminarán, se darán de baja para tener un histórico e información de proveedores que se manejaban con anterioridad.	

*Tabla 30: Historia de Usuario 10
Elaborado por: Investigador*

Historia de Usuario	
Numero: 11	Usuario: Desarrollador
Nombre de historia: Desarrollo de API clientes	
Prioridad en el negocio: Alta	Puntos estimados: 3
Riesgo del desarrollo: Alta	Iteración asignada: 5
Responsable: Camila Martinez	
Descripción: Desarrollar la API REST de los clientes con los métodos GET, POST, PUT para la interacción con la interfaz diseñada.	
Observación:	

*Tabla 31: Historia de Usuario 11
Elaborado por: Investigador*

Historia de Usuario	
Numero: 12	Usuario: Desarrollador
Nombre de historia: Desarrollo de API productos	
Prioridad en el negocio: Alta	Puntos estimados: 3
Riesgo del desarrollo: Alta	Iteración asignada: 5
Responsable: Camila Martinez	

Descripción: Desarrollar la API REST de los productos con los métodos GET, POST, PUT para la interacción con la interfaz diseñada.
Observación:

*Tabla 32: Historia de Usuario 12
Elaborado por: Investigador*

Historia de Usuario	
Numero: 13	Usuario: Desarrollador
Nombre de historia: Desarrollo de API usuarios	
Prioridad en el negocio: Alta	Puntos estimados: 3
Riesgo del desarrollo: Alta	Iteración asignada: 5
Responsable: Camila Martinez	
Descripción: Desarrollar la API REST de los usuarios con los métodos GET, POST, PUT para la interacción con la interfaz diseñada.	
Observación: Los usuarios del sistema no se eliminarán, estos solo se darán de baja para llevar un control del histórico de cuentas de usuarios.	

*Tabla 33: Historia de Usuario 13
Elaborado por: Investigador*

Historia de Usuario	
Numero: 14	Usuario: Desarrollador
Nombre de historia: Desarrollo de reportes	
Prioridad en el negocio: Alta	Puntos estimados: 3
Riesgo del desarrollo: Alta	Iteración asignada: 5
Responsable: Camila Martinez	
Descripción: Según los reportes solicitados, realizar la programación adecuada para comunicarse con las interfaces y devolver los datos necesarios.	
Observación: Ninguna	

*Tabla 34: Historia de Usuario 14
Elaborado por: Investigador*

Historia de Usuario	
Numero: 15	Usuario: Desarrollador
Nombre de historia: Pruebas finales	
Prioridad en el negocio: Alta	Puntos estimados: 1
Riesgo del desarrollo: Alta	Iteración asignada: 6
Responsable: Camila Martinez	
Descripción: Se realiza el testeo de todas las funciones y reportes junto con todo el personal involucrado en los procesos.	
Observación: Estas son las pruebas finales, después de cada iteración también se deberán realizar las respectivas pruebas	

*Tabla 35: Historia de Usuario 15
Elaborado por: Investigador*

Tareas

Las tareas son actividades detalladas que se realizan en cada historia de usuario, aquí se define el tipo de actividad, y las fechas en la que se estima realizar la tarea y una descripción de esta. Las tareas son supervisadas paralelamente a la entrega.

Tarea	Responsable: Camila Martinez
N.º Tarea: T001	Código Historia: 01
Tipo Tarea: Análisis	Puntos estimados: 1
Nombre Tarea:	Análisis de la Base de datos
Fecha de inicio: junio 2022	Fecha Fin: junio 2022
Descripción:	Se analizarán cuáles serán las tablas más importantes y como interactúan entre ellas.

Tabla 36: Tarea 001

Elaborado por: Investigador

Tarea	Responsable: Camila Martinez
N.º Tarea: T002	Código Historia: 01
Tipo Tarea: Análisis	Puntos estimados: 1
Nombre Tarea:	Modelamiento físico de la Base de datos
Fecha de inicio: junio 2022	Fecha Fin: junio 2022
Descripción:	Se empieza a hacer un primer modelamiento con tablas, atributos y relaciones.

Tabla 37: Tarea 002

Elaborado por: Investigador

Tarea	Responsable: Camila Martinez
N.º Tarea: T003	Código Historia: 01
Tipo Tarea: Análisis	Puntos estimados: 1
Nombre Tarea:	Modelamiento en MySQL y pruebas de funcionamiento
Fecha de inicio: julio 2022	Fecha Fin: julio 2022

Descripción:	Se crea la base de datos en MySQL y se realizan pequeñas pruebas para verificar el funcionamiento.
--------------	--

Tabla 38: Tarea 003

Elaborado por: Investigador

Tarea	Responsable: Camila Martinez
N.º Tarea: T004	Código Historia: 02
Tipo Tarea: Análisis	Puntos estimados: 1
Nombre Tarea:	Análisis y definición de roles
Fecha de inicio: julio 2022	Fecha Fin: julio 2022
Descripción:	Según el levantamiento de información realizado, se analiza y se establece roles y funciones para los distintos usuarios que harán uso del sistema.

Tabla 39: Tarea 004

Elaborado por: Investigador

Tarea	Responsable: Camila Martinez
N.º Tarea: T005	Código Historia: 03
Tipo Tarea: Diseño	Puntos estimados: 1
Nombre Tarea:	Diseño pantalla master
Fecha de inicio: agosto 2022	Fecha Fin: septiembre 2022
Descripción:	Se establecen los colores, formas y menú que se manejaran el sistema

Tabla 40: Tarea 005

Elaborado por: Investigador

Tarea	Responsable: Camila Martinez	
N.º Tarea: T006	Código Historia: 03	
Tipo Tarea: Diseño	Puntos estimados: 1	
Nombre Tarea:	Diseño pantalla inicio	
Fecha de inicio: septiembre 2022	Fecha Fin: septiembre 2022	
Descripción:	Se empieza por diseñar el dashboard del sistema y las funciones que tendrá.	

*Tabla 41: Tarea 006
Elaborado por: Investigador*

Tarea	Responsable: Camila Martinez	
N.º Tarea: T007	Código Historia: 04	
Tipo Tarea: Diseño	Puntos estimados: 1	
Nombre Tarea:	Interfaz de listado proveedores, filtros	
Fecha de inicio: septiembre 2022	Fecha Fin: septiembre 2022	
Descripción:	Se modela en la interfaz principal la pantalla de proveedores y filtros que tendrá	

*Tabla 42: Tarea 007
Elaborado por: Investigador*

Tarea	Responsable: Camila Martinez	
N.º Tarea: T008	Código Historia: 04	
Tipo Tarea: Diseño	Puntos estimados: 1	
Nombre Tarea:	Interfaz de adición de proveedores	
Fecha de inicio: septiembre 2022	Fecha Fin: septiembre 2022	
Descripción:	Se modela el formulario para agregar proveedores	

*Tabla 43: Tarea 008
Elaborado por: Investigador*

Tarea	Responsable: Camila Martinez	
N.º Tarea: T009	Código Historia: 04	
Tipo Tarea: Diseño	Puntos estimados: 1	
Nombre Tarea:	Interfaz de edición de proveedores	
Fecha de inicio: septiembre 2022	Fecha Fin: septiembre 2022	

Descripción:	Se modela el formulario para editar proveedores
--------------	---

Tabla 44: Tarea 009

Elaborado por: Investigador

Tarea	Responsable: Camila Martinez	
N.º Tarea: T010	Código Historia: 05	
Tipo Tarea: Diseño	Puntos estimados: 1	
Nombre Tarea:	Interfaz de listado de clientes y filtros	
Fecha de inicio: septiembre 2022	Fecha Fin: septiembre 2022	
Descripción:	Se modela en la interfaz principal la pantalla de clientes y filtros que tendrá	

Tabla 45: Tarea 010

Elaborado por: Investigador

Tarea	Responsable: Camila Martinez	
N.º Tarea: T011	Código Historia: 05	
Tipo Tarea: Diseño	Puntos estimados: 1	
Nombre Tarea:	Interfaz de adición de clientes	
Fecha de inicio: septiembre 2022	Fecha Fin: septiembre 2022	
Descripción:	Se modela el formulario para agregar clientes	

Tabla 46: Tarea 011

Elaborado por: Investigador

Tarea	Responsable: Camila Martinez	
N.º Tarea: T012	Código Historia: 05	
Tipo Tarea: Diseño	Puntos estimados: 1	
Nombre Tarea:	Interfaz de edición de clientes	
Fecha de inicio: septiembre 2022	Fecha Fin: septiembre 2022	
Descripción:	Se modela el formulario para editar clientes	

Tabla 47: Tarea 012

Elaborado por: Investigador

Tarea	Responsable: Camila Martinez	
N.º Tarea: T013	Código Historia: 06	
Tipo Tarea: Diseño	Puntos estimados: 1	
Nombre Tarea:	Interfaz de listado de productos, consulta y filtros	
Fecha de inicio: septiembre 2022	Fecha Fin: septiembre 2022	
Descripción:	Se modela en la interfaz principal la pantalla de productos, stock y filtros que tendrá	

Tabla 48: Tarea 013

Elaborado por: Investigador

Tarea	Responsable: Camila Martinez	
N.º Tarea: T014	Código Historia: 06	
Tipo Tarea: Diseño	Puntos estimados: 1	
Nombre Tarea:	Interfaz de adición de productos	
Fecha de inicio: septiembre 2022	Fecha Fin: octubre 2022	
Descripción:	Se modela el formulario para agregar productos y sus características.	

Tabla 49: Tarea 014

Elaborado por: Investigador

Tarea	Responsable: Camila Martinez	
N.º Tarea: T015	Código Historia: 06	
Tipo Tarea: Diseño	Puntos estimados: 1	
Nombre Tarea:	Interfaz de edición de productos	
Fecha de inicio: octubre 2022	Fecha Fin: octubre 2022	
Descripción:	Se modela el formulario para editar productos y sus características.	

Tabla 50: Tarea 015

Elaborado por: Investigador

Tarea	Responsable: Camila Martínez
N.º Tarea: T016	Código Historia: 06
Tipo Tarea: Diseño	Puntos estimados: 1
Nombre Tarea:	Interfaz de stock
Fecha de inicio: octubre 2022	Fecha Fin: octubre 2022
Descripción:	Interfaz de consulta de la tabla stock y detalles del producto

*Tabla 51: Tarea 016
Elaborado por: Investigador*

Tarea	Responsable: Camila Martínez
N.º Tarea: T017	Código Historia: 06
Tipo Tarea: Diseño	Puntos estimados: 1
Nombre Tarea:	Interfaz de productos, consulta y filtros
Fecha de inicio: octubre 2022	Fecha Fin: octubre 2022
Descripción:	Se modela en la interfaz principal la pantalla de productos, stock y filtros que tendrá

*Tabla 52: Tarea 017
Elaborado por: Investigador*

Tarea	Responsable: Camila Martínez
N.º Tarea: T018	Código Historia: 07
Tipo Tarea: Diseño	Puntos estimados: 1
Nombre Tarea:	Interfaz de proformas
Fecha de inicio: octubre 2022	Fecha Fin: octubre 2022
Descripción:	Se modela en la interfaz principal la pantalla de proformas y filtros que tendrá

*Tabla 53: Tarea 018
Elaborado por: Investigador*

Tarea	Responsable: Camila Martínez	
N.º Tarea: T019	Código Historia: 07	
Tipo Tarea: Diseño	Puntos estimados: 1	
Nombre Tarea:	Interfaz de facturación	
Fecha de inicio: octubre 2022	Fecha Fin: octubre 2022	
Descripción:	Se modela en la pantalla principal la pantalla para crear facturas de venta a clientes, y los filtros.	

Tabla 54: Tarea 019

Elaborado por: Investigador

Tarea	Responsable: Camila Martínez	
N.º Tarea: T020	Código Historia: 07	
Tipo Tarea: Diseño	Puntos estimados: 1	
Nombre Tarea:	Interfaz de salida de mercadería sin documento	
Fecha de inicio: octubre 2022	Fecha Fin: octubre 2022	
Descripción:	Se modela en la interfaz principal la pantalla para salida de mercadería, y los filtros	

Tabla 55: Tarea 020

Elaborado por: Investigador

Tarea	Responsable: Camila Martínez	
N.º Tarea: T021	Código Historia: 08	
Tipo Tarea: Diseño	Puntos estimados: 1	
Nombre Tarea:	Interfaz de reporte 1	
Fecha de inicio: octubre 2022	Fecha Fin: octubre 2022	
Descripción:	Se modela en la interfaz principal el primer reporte visual	

Tabla 56: Tarea 021

Elaborado por: Investigador

Tarea	Responsable: Camila Martínez
N.º Tarea: T022	Código Historia: 08
Tipo Tarea: Diseño	Puntos estimados: 1
Nombre Tarea:	Interfaz de reporte 2
Fecha de inicio: octubre 2022	Fecha Fin: octubre 2022
Descripción:	Se modela en la interfaz principal el segundo reporte visual

*Tabla 57: Tarea 022
Elaborado por: Investigador*

Tarea	Responsable: Camila Martínez
N.º Tarea: T023	Código Historia: 09
Tipo Tarea: Diseño	Puntos estimados: 1
Nombre Tarea:	Interfaz configuración de usuario
Fecha de inicio: octubre 2022	Fecha Fin: octubre 2022
Descripción:	Se diseña la pantalla donde se configurarán las credenciales de usuario

*Tabla 58: Tarea 023
Elaborado por: Investigador*

Tarea	Responsable: Camila Martínez
N.º Tarea: T024	Código Historia: 09
Tipo Tarea: Diseño	Puntos estimados: 1
Nombre Tarea:	Interfaz herramientas administrador
Fecha de inicio: octubre 2022	Fecha Fin: octubre 2022
Descripción:	Se diseña la pantalla donde se configurarán las herramientas de administrador

*Tabla 59: Tarea 024
Elaborado por: Investigador*

Tarea	Responsable: Camila Martinez	
N.º Tarea: T025	Código Historia: 10	
Tipo Tarea: Programación	Puntos estimados: 1	
Nombre Tarea:	Tabla proveedores método GET	
Fecha de inicio: noviembre 2022	Fecha Fin: noviembre 2022	
Descripción:	Se crean los métodos para consulta de proveedores	

Tabla 60: Tarea 025

Elaborado por: Investigador

Tarea	Responsable: Camila Martinez	
N.º Tarea: T026	Código Historia: 10	
Tipo Tarea: Programación	Puntos estimados: 1	
Nombre Tarea:	Tabla proveedores método POST	
Fecha de inicio: noviembre 2022	Fecha Fin: noviembre 2022	
Descripción:	Se crean los métodos para agregar de proveedores	

Tabla 61: Tarea 026

Elaborado por: Investigador

Tarea	Responsable: Camila Martinez	
N.º Tarea: T027	Código Historia: 10	
Tipo Tarea: Programación	Puntos estimados: 1	
Nombre Tarea:	Tabla proveedores método PUT	
Fecha de inicio: noviembre 2022	Fecha Fin: noviembre 2022	
Descripción:	Se crean los métodos para actualizar de proveedores	

Tabla 62: Tarea 027

Elaborado por: Investigador

Tarea	Responsable: Camila Martinez	
N.º Tarea: T028	Código Historia: 11	
Tipo Tarea: Programación	Puntos estimados: 1	
Nombre Tarea:	Tabla clientes método GET	
Fecha de inicio: noviembre 2022	Fecha Fin: noviembre 2022	
Descripción:	Se crean los métodos para consulta de clientes	

Tabla 63: Tarea 028

Elaborado por: Investigador

Tarea	Responsable: Camila Martinez	
N.º Tarea: T029	Código Historia: 11	
Tipo Tarea: Programación	Puntos estimados: 2	
Nombre Tarea:	Tabla clientes método POST	
Fecha de inicio: diciembre 2022	Fecha Fin: diciembre 2022	
Descripción:	Se crean los métodos para agregar de clientes	

Tabla 64: Tarea 029

Elaborado por: Investigador

Tarea	Responsable: Camila Martinez	
N.º Tarea: T030	Código Historia: 11	
Tipo Tarea: Programación	Puntos estimados: 2	
Nombre Tarea:	Tabla clientes método PUT	
Fecha de inicio: diciembre 2022	Fecha Fin: diciembre 2022	
Descripción:	Se crean los métodos para editar de clientes	

Tabla 65: Tarea 030

Elaborado por: Investigador

Tarea	Responsable: Camila Martinez	
N.º Tarea: T031	Código Historia: 12	
Tipo Tarea: Programación	Puntos estimados: 2	
Nombre Tarea:	Tabla productos método GET	
Fecha de inicio: diciembre 2022	Fecha Fin: diciembre 2022	
Descripción:	Se crean los métodos para consulta de productos y stock	

*Tabla 66: Tarea 031
Elaborado por: Investigador*

Tarea	Responsable: Camila Martinez	
N.º Tarea: T032	Código Historia: 12	
Tipo Tarea: Programación	Puntos estimados: 2	
Nombre Tarea:	Tabla productos método POST	
Fecha de inicio: diciembre 2022	Fecha Fin: diciembre 2022	
Descripción:	Se crean los métodos para agregar de productos	

*Tabla 67: Tarea 032
Elaborado por: Investigador*

Tarea	Responsable: Camila Martinez	
N.º Tarea: T033	Código Historia: 12	
Tipo Tarea: Programación	Puntos estimados: 2	
Nombre Tarea:	Tabla productos método PUT	
Fecha de inicio: diciembre 2022	Fecha Fin: diciembre 2022	
Descripción:	Se crean los métodos para actualizar productos y stock	

*Tabla 68: Tarea 033
Elaborado por: Investigador*

Tarea	Responsable: Camila Martinez	
N.º Tarea: T034	Código Historia: 12	
Tipo Tarea: Programación	Puntos estimados: 2	
Nombre Tarea:	Tabla compras método GET	
Fecha de inicio: diciembre 2022	Fecha Fin: diciembre 2022	
Descripción:	Se crean los métodos para consulta de facturas de compra	

*Tabla 69: Tarea 034
Elaborado por: Investigador*

Tarea	Responsable: Camila Martinez	
N.º Tarea: T035	Código Historia: 12	
Tipo Tarea: Programación	Puntos estimados: 2	
Nombre Tarea:	Tabla compras método POST	
Fecha de inicio: diciembre 2022	Fecha Fin: diciembre 2022	
Descripción:	Se crean los métodos para agregar de facturas de compra	

*Tabla 70: Tarea 035
Elaborado por: Investigador*

Tarea	Responsable: Camila Martinez	
N.º Tarea: T036	Código Historia: 12	
Tipo Tarea: Programación	Puntos estimados: 2	
Nombre Tarea:	Tabla compras método PUT	
Fecha de inicio: diciembre 2022	Fecha Fin: diciembre 2022	
Descripción:	Se crean los métodos para anular de facturas de compra, ya que estas no se pueden actualizar	

*Tabla 71: Tarea 036
Elaborado por: Investigador*

Tarea	Responsable: Camila Martinez	
N.º Tarea: T037	Código Historia: 12	
Tipo Tarea: Programación	Puntos estimados: 2	
Nombre Tarea:	Tabla proformas método GET	
Fecha de inicio: diciembre 2022	Fecha Fin: diciembre 2022	
Descripción:	Se crean los métodos para visualizar y consultar cotizaciones	

Tabla 72: Tarea 037

Elaborado por: Investigador

Tarea	Responsable: Camila Martinez	
N.º Tarea: T038	Código Historia: 12	
Tipo Tarea: Programación	Puntos estimados: 2	
Nombre Tarea:	Tabla proformas método POST	
Fecha de inicio: diciembre 2022	Fecha Fin: diciembre 2022	
Descripción:	Se crean los métodos para agregar cotizaciones sin afectar al stock de productos	

Tabla 73: Tarea 038

Elaborado por: Investigador

Tarea	Responsable: Camila Martinez	
N.º Tarea: T039	Código Historia: 12	
Tipo Tarea: Programación	Puntos estimados: 2	
Nombre Tarea:	Tabla proformas método PUT	
Fecha de inicio: diciembre 2022	Fecha Fin: diciembre 2022	
Descripción:	Se crean los métodos para editar el estado de las cotizaciones.	

Tabla 74: Tarea 039

Elaborado por: Investigador

Tarea	Responsable: Camila Martinez
N.º Tarea: T040	Código Historia: 12
Tipo Tarea: Programación	Puntos estimados: 2
Nombre Tarea:	Tabla ventas método GET
Fecha de inicio: diciembre 2022	Fecha Fin: diciembre 2022
Descripción:	Se crean los métodos para consulta de facturas de venta

*Tabla 75: Tarea 040
Elaborado por: Investigador*

Tarea	Responsable: Camila Martinez
N.º Tarea: T041	Código Historia: 12
Tipo Tarea: Programación	Puntos estimados: 2
Nombre Tarea:	Tabla ventas método POST
Fecha de inicio: enero 2023	Fecha Fin: enero 2023
Descripción:	Se crean los métodos para agregar nuevas facturas de venta

*Tabla 76: Tarea 041
Elaborado por: Investigador*

Tarea	Responsable: Camila Martinez
N.º Tarea: T042	Código Historia: 12
Tipo Tarea: Programación	Puntos estimados: 2
Nombre Tarea:	Tabla entrada-salida sin documento método GET
Fecha de inicio: enero 2023	Fecha Fin: enero 2023
Descripción:	Se crean los métodos para consulta de entrada o salida de productos sin factura.

*Tabla 77: Tarea 042
Elaborado por: Investigador*

Tarea	Responsable: Camila Martinez
N.º Tarea: T043	Código Historia: 12
Tipo Tarea: Programación	Puntos estimados: 2
Nombre Tarea:	Tabla entrada-salida sin documento método POST
Fecha de inicio: enero 2023	Fecha Fin: enero 2023
Descripción:	Se crean los métodos para agregar o disminuir productos sin factura.

*Tabla 78: Tarea 043
Elaborado por: Investigador*

Tarea	Responsable: Camila Martinez
N.º Tarea: T044	Código Historia: 12
Tipo Tarea: Programación	Puntos estimados: 2
Nombre Tarea:	Tabla entrada-salida sin documento método GET
Fecha de inicio: enero 2023	Fecha Fin: enero 2023
Descripción:	Se crean los métodos para consultar la entrada o salida de productos sin factura

*Tabla 79: Tarea 044
Elaborado por: Investigador*

Tarea	Responsable: Camila Martinez
N.º Tarea: T045	Código Historia: 13
Tipo Tarea: Programación	Puntos estimados: 2
Nombre Tarea:	Tabla usuarios método GET
Fecha de inicio: enero 2023	Fecha Fin: enero 2023
Descripción:	Se crea el método para consultar los usuarios existentes

*Tabla 80: Tarea 045
Elaborado por: Investigador*

Tarea	Responsable: Camila Martinez
N.º Tarea: T046	Código Historia: 13
Tipo Tarea: Programación	Puntos estimados: 2
Nombre Tarea:	Tabla usuarios método POST
Fecha de inicio: enero 2023	Fecha Fin: enero 2023
Descripción:	Se crea el método para agregar nuevos usuarios.

*Tabla 81: Tarea 046
Elaborado por: Investigador*

Tarea	Responsable: Camila Martinez
N.º Tarea: T047	Código Historia: 13
Tipo Tarea: Programación	Puntos estimados: 1
Nombre Tarea:	Tabla usuarios método PUT
Fecha de inicio: enero 2023	Fecha Fin: enero 2023
Descripción:	Se crea el método para actualizar a los usuarios.

*Tabla 82: Tarea 047
Elaborado por: Investigador*

Tarea	Responsable: Camila Martinez
N.º Tarea: T048	Código Historia: 14
Tipo Tarea: Programación	Puntos estimados: 3
Nombre Tarea:	Desarrollo de reportes
Fecha de inicio: enero 2023	Fecha Fin: enero 2023
Descripción:	Se crean los métodos pertinentes para la consulta y filtro de datos que se consideran más importantes en el sistema.

*Tabla 83: Tarea 048
Elaborado por: Investigador*

Tarea	Responsable: Camila Martinez
N.º Tarea: T049	Código Historia: 15
Tipo Tarea: Evaluación	Puntos estimados: 2
Nombre Tarea:	Pruebas generales
Fecha de inicio: enero 2023	Fecha Fin: enero 2023
Descripción:	Se realizan pruebas del sistema.

*Tabla 84: Tarea 049
Elaborado por: Investigador*

Estimación de historias de usuario

Una vez finalizadas las historias de usuario con su respectivo análisis y tareas, se presenta la valoración en la cual se estima 4 horas de trabajo en los días laborables (lunes, martes, miércoles, jueves, viernes), lo cual suma un total de 20 horas a la semana.

Plan de iteraciones

El número de iteraciones del proyecto se estimó y se agrupó según el tipo de tareas a realizar dentro de cada historia, repartiéndose en un total de 6 iteraciones.

Iteración	N.º	Historia de usuario	Tiempo estimado	
			Días	Horas
1	001	Análisis y modelamiento de la base de datos	3	12
1	002	Definición de Roles y Usuarios	1	4
2	003	Interfaz inicio y general	4	16
2	004	Interfaz proveedores	3	12
2	005	Interfaz clientes	3	12
3	006	Interfaz administración de productos	10	40

3	007	Interfaces salida productos	10	40
4	008	Interfaz de reportes	5	20
4	009	Interfaz administración de usuarios	3	12
5	010	Desarrollo de API proveedores	5	20
5	011	Desarrollo de API clientes	5	20
5	012	Desarrollo de API productos	20	80
5	013	Desarrollo de API usuarios	3	12
5	014	Desarrollo de reportes	7	28
6	015	Pruebas finales	3	12
Total			85	340

*Tabla 85: Plan de iteraciones general
Elaborado por: Investigador*

Iteración 01

Iteración	N.º	Historia de usuario	Tiempo estimado	
			Días	Horas
1	001	Análisis y modelamiento de la base de datos	3	12
1	002	Definición de Roles y Usuarios	1	4
Total			4	16

*Tabla 86: Plan de iteración 01
Elaborado por: Investigador*

Iteración 02

Iteración	N.º	Historia de usuario	Tiempo estimado	
			Días	Horas
2	3	Interfaz inicio y general	4	16
2	4	Interfaz proveedores	3	12
2	5	Interfaz clientes	3	12
Total			10	40

*Tabla 87: Plan de iteración 02
Elaborado por: Investigador*

Iteración 03

Iteración	N.º	Historia de usuario	Tiempo estimado	
			Días	Horas
3	6	Interfaz administración de productos	10	40
3	7	Interfaces salida productos	10	40
Total			20	80

*Tabla 88: Plan de iteración 03
Elaborado por: Investigador*

Iteración 04

Iteración	N.º	Historia de usuario	Tiempo estimado	
			Días	Horas
4	8	Interfaz de reportes	5	20
4	9	Interfaz administración de usuarios	3	12
Total			8	32

*Tabla 89: Plan de iteración 04
Elaborado por: Investigador*

Iteración 05

Iteración	N.º	Historia de usuario	Tiempo estimado	
			Días	Horas
5	10	Desarrollo de API proveedores	5	20
5	11	Desarrollo de API clientes	5	20
5	12	Desarrollo de API productos	20	80
5	13	Desarrollo de API usuarios	3	12
5	14	Desarrollo de reportes	7	28
Total			40	160

*Tabla 90: Plan de iteración 05
Elaborado por: Investigador*

Iteración 06

Iteración	N.º	Historia de usuario	Tiempo estimado	
			Días	Horas
6	15	Pruebas finales	3	12
Total			3	12

*Tabla 91: Plan de iteración 06
Elaborado por: Investigador*

Plan de entrega

Para la elaboración del plan de entrega, se toma en cuenta cada una de las iteraciones a las que han sido asignadas cada historia de usuario, de acuerdo con la prioridad que presenta cada una, tomando en cuenta que cada iteración tendrá un tiempo estimado de cuatro semanas, durante los días laborables de la misma.

N.º	Historia de usuario	Tiempo estimado		Iteración asignada						Entrega asignada					
		Días	Horas	1	2	3	4	5	6	1	2	3	4	5	6
1	Análisis y modelamiento de la base de datos	3	12	X						X					
2	Definición de Roles y Usuarios	1	4	X						X					
3	Interfaz inicio y general	4	16		X					X					
4	Interfaz proveedores	3	12		X					X					
5	Interfaz clientes	3	12		X					X					
6	Interfaz administración de productos	10	40			X					X				
7	Interfaces salida productos	10	40			X					X				
8	Interfaz de reportes	5	20				X					X			
9	Interfaz administración de usuarios	3	12				X					X			
10	Desarrollo de API proveedores	5	20					X						X	
11	Desarrollo de API clientes	5	20					X						X	
12	Desarrollo de API productos	20	80					X						X	
13	Desarrollo de API usuarios	3	12					X						X	
14	Desarrollo de reportes	7	28					X						X	
15	Pruebas finales	3	12						X						X

Tabla 92: Plan de entrega
Elaborado por: Investigador

3.1.1.2. Fase de diseño

Metáfora

La aplicación PWA para el control de inventario ayudará a llevar al día el stock que se maneja en la entidad, ayudando a reducir errores humanos. De esta manera se podrán presentar reportes a tiempo real.

Iteración	N.º	Historia	Prioridad	Riesgo	Estado de desarrollo	Pruebas
1	1	Análisis y modelamiento de la base de datos	Alta	Alta	Completo	Aprobado
	2	Definición de Roles y Usuarios	Media	Media	Completo	Aprobado
2	3	Interfaz inicio y general	Media	Baja	Completo	Aprobado
	4	Interfaz proveedores	Media	Baja	Completo	Aprobado
	5	Interfaz clientes	Media	Baja	Completo	Aprobado
3	6	Interfaz administración de productos	Media	Baja	Completo	Aprobado
	7	Interfaces salida productos	Media	Baja	Completo	Aprobado
4	8	Interfaz de reportes	Media	Baja	Completo	Aprobado
	9	Interfaz administración de usuarios	Media	Baja	Completo	Aprobado
5	10	Desarrollo de API proveedores	Alta	Alta	Completo	Aprobado
	11	Desarrollo de API clientes	Alta	Alta	Completo	Aprobado

	12	Desarrollo de API productos	Alta	Alta	Completo	Aprobado
	13	Desarrollo de API usuarios	Alta	Alta	Completo	Aprobado
	14	Desarrollo de reportes	Alta	Alta	Completo	Aprobado
6	15	Pruebas finales	Alta	Alta	Completo	Aprobado

*Tabla 93: Metáfora
Elaborado por: Investigador*

La Aplicación web progresiva (PWA) para el control de inventarios en la empresa FERRIMAR permitirá la consulta a tiempo real del stock desde cualquier dispositivo móvil, esto debido a que no todos los empleados que trabajan ahí pueden hacer uso de un computador, por lo que deberán ocupar sus dispositivos móviles o Tablet.

Consta de varios módulos definidos que son:

- 1. Módulo Proveedores:** Se deberá visualizar, actualizar e ingresar los datos definidos como obligatorios y opcionales de los proveedores para llevar de una mejor manera la entrada de productos.
- 2. Módulo Compras:** Permitirá agregar facturas de compra a proveedores, se podrán anular estas facturas y visualizar en listado. Una vez ejecutada cada tarea, se reflejarán los cambios en la aplicación y en las tablas involucradas.
- 3. Módulo Clientes:** Se deberá visualizar, actualizar e ingresar los datos definidos como obligatorios y opcionales de los clientes para llevar de una mejor manera la salida de productos.
- 4. Módulo Stock y Productos:** El módulo de stock es el central y más importante de la PWA, aquí se mostrará información, actualizará datos de la mercancía. Se debe llevar bien el control ya que un stock actualizado depende de las compras y ventas.
- 5. Módulo Ventas y proformas:** Permitirá generar cotizaciones y, de ser el caso facturar las cotizaciones o crear de cero. Anular facturas y visualizar ventas. Una vez ejecutada cada tarea, se reflejarán los cambios en la aplicación y en las tablas involucradas.

6. **Módulo Reportes:** Muestra la información de ventas y productos según rangos de fechas
7. **Módulo Administración:** Llevar el control de usuarios de la aplicación y permisos.

Tarjetas CRC

Las tarjetas CRC son un conjunto de tarjetas estándar y cada una representará una clase, las responsabilidades que esta tiene y los colaboradores que interactuarán con la misma.

Ingresar al aplicativo	
Responsabilidades	Colaboradores
Ingreso a la aplicación a través de las credenciales de usuario, verificar información en la base de datos.	Capa de acceso de datos
Observaciones: Involucra a la tabla <i>usuario</i>	

*Tabla 94: Tarjeta CRC, Ingresar al aplicativo
Elaborado por: Investigador*

Ingresar mercadería	
Responsabilidades	Colaboradores
Realizar el ingreso de productos por factura o entrada sin documento	Capa de acceso de datos CRUD productos
Observaciones: Se involucran las tablas de compras, detalle de entrada-salida de productos, stock, producto y proveedores.	

*Tabla 95: Tarjeta CRC, Ingresar mercadería
Elaborado por: Investigador*

Gestionar Precios	
Responsabilidades	Colaboradores
Visualizar, editar y agregar precios a los productos existentes Establecer precios por defecto	Capa acceso de datos
Observaciones: Se involucra a la tabla precios de la base de datos, stock y producto.	

*Tabla 96: Tarjeta CRC, Gestionar Precios
Elaborado por: Investigador*

Generar proformas/facturas	
Responsabilidades	Colaboradores
Realizar cotizaciones, ver estado de las cotizaciones Crear facturas o anularlas	Capa acceso de datos CRUD proformas, ventas, salida de material sin documento
Observaciones: Se involucran todas las tablas de venta, proforma, producto y stock	

*Tabla 97: Tarjeta CRC, Generar proformas/facturas
Elaborado por: Investigador*

CRUD proveedores	
Responsabilidades	Colaboradores
Visualizar y gestionar los distintos proveedores que posee la empresa	Capa acceso de datos CRUD proveedores
Observaciones: Involucra la tabla proveedor	

*Tabla 98: Tarjeta CRC, CRUD proveedores
Elaborado por: Investigador*

CRUD clientes	
Responsabilidades	Colaboradores
Visualizar y gestionar los distintos clientes que posee la empresa	Capa acceso de datos CRUD clientes
Observaciones: Involucra únicamente la tabla cliente	

*Tabla 99: Tarjetas CRC, CRUD clientes
Elaborado por: Investigador*

CRUD productos	
Responsabilidades	Colaboradores
Visualizar productos y stock, realizar filtros de productos	Capa acceso de datos
Realizar ingreso de material, tanto por factura como por entrada sin documento	CRUD productos
Observaciones: Involucra las tablas de compra, entrada-salida de productos, venta, unidades, stock.	

*Tabla 100: Tarjeta CRC, CRUD productos
Elaborado por: Investigador*

CRUD usuarios	
Responsabilidades	Colaboradores
Administrar nombres de usuario y contraseña, descripción del empleado.	Capa acceso de datos CRUD usuarios
Observaciones: Cada uno de los usuarios puede gestionar su contraseña, pero únicamente deberá tener acceso a la administración general el gerente. Se involucra la tabla usuario	

*Tabla 101: Tarjeta CRC, CRUD usuarios
Elaborado por: Investigador*

Generar reportes	
Responsabilidades	Colaboradores
Visualizar, generar reportes	Capa acceso de datos
Observaciones: Se genera según las tablas que se relacionan con ventas y productos.	

*Tabla 102: Tarjetas CRC, Generar reportes
Elaborado por: Investigador*

Estructura del proyecto

En el presente proyecto se utilizaron las herramientas más adecuadas para su elaboración, tanto en hardware como en software.

Hardware

Laptop Asus VivoBook X513EA

Procesador i5 11va gen

Memoria RAM 12 GB.

Software

Sistema operativo de 64 bits.

Angular CLI: 13.3.9

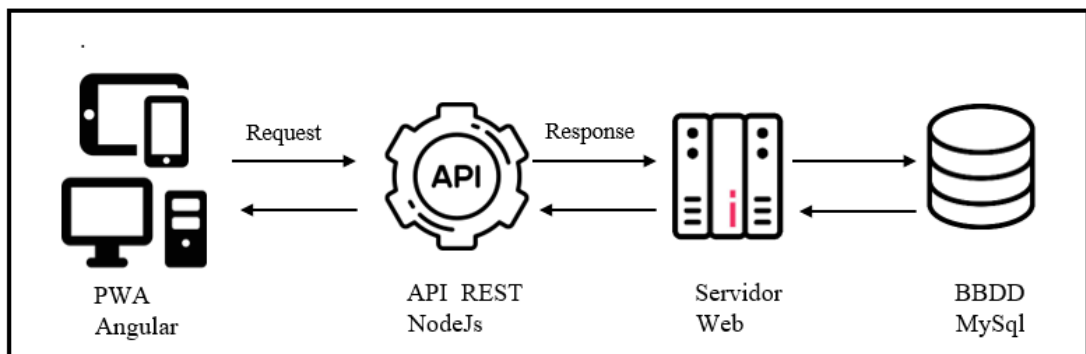
Node 15.16.1

MySQL 8

Editor de código Visual Studio Code versión 1.74.3

Arquitectura del proyecto

A través la definición de la arquitectura de la aplicación, se puede analizar de manera visual como va a funcionar cada una de las partes y conocer la forma en que estas van a interactuar entre ellas. Es necesario ya que se crea una base sólida para el proyecto y se consigue una plataforma escalable.



*Imagen 9: Arquitectura del sistema
Elaborado por: Investigador*

Observando la *Imagen 9*, se puede identificar que se trata de una arquitectura que se compone de la capa de presentación, capa de proceso y la capa de datos. Se implementó de esta manera con la finalidad de facilitar la administración de la aplicación, y en caso de requerir de algún cambio poder encontrar fácilmente el código que lo contiene.

Organización del proyecto

Una vez se crea un proyecto en angular, este ya viene definido con su propia estructura, a la cual se le agregaron algunas clases para una mejor implementación.

e2e: contiene ficheros los cuales realizan test automáticos.

node_modules: contiene las dependencias que el proyecto necesita.

src: es el directorio en donde se agregarán los componentes y módulos, es el más importante. Contiene las carpetas app, assets, enviroments y archivos como favicon.ico, index.htm, main.ts.

editorconfig: contiene la información de configuración de editor de código.

.gitignore: son las carpetas y archivos que debe ignorar el git cuando se agregue el proyecto al repositorio.

angularjson: contiene toda la configuración de Angular como rutas o versiones.

package.json: aquí se almacena la configuración de la aplicación como las dependencias necesarias para su correcta ejecución y sus versiones.

Diseño de la Base de datos

Una vez recolectada toda la información, se procede a diseñar el pase de datos, para lo cual se utilizó MySQL en la versión 8. En el siguiente diagrama se muestra las tablas junto con sus atributos y las relaciones que poseen cada una.

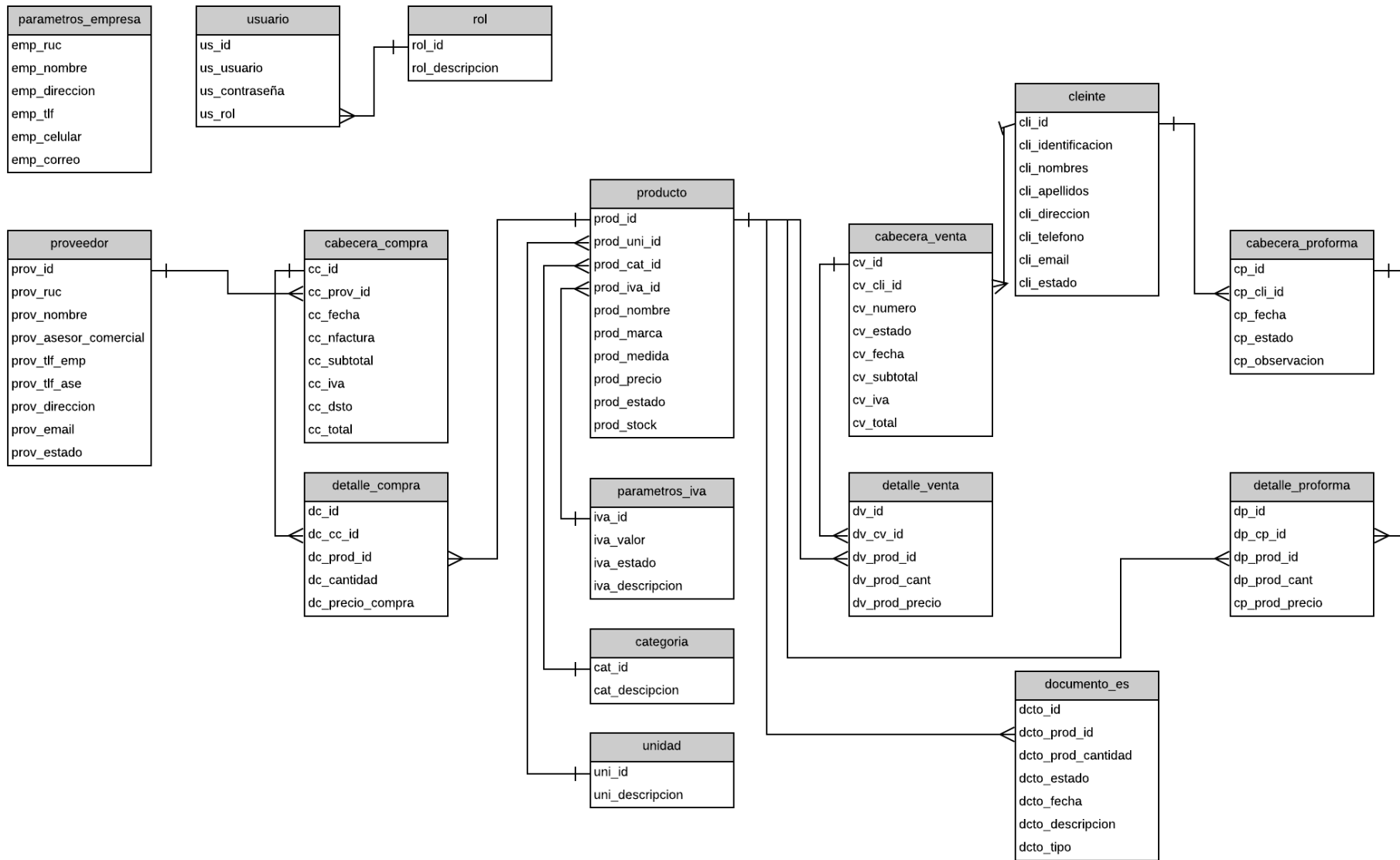


Imagen 9: Diagrama Entidad-Relación de la Base de Datos
Elaborado por: Investigador

Diseño de Interfaces

Es importante el bosquejo de las interfaces debido a que es un primer acercamiento al resultado final que se espera del software. Ayuda a brindar una perspectiva a los usuarios de cómo será el funcionamiento de los procesos y de qué forma va a interactuar el usuario final con las distintas interfaces.

Diseño Login Usuario

Se trata de la primera interfaz de la PWA, el usuario deberá identificarse con sus credenciales, los datos serán comprobados con la base de datos y posteriormente se dará o no el acceso al sistema.

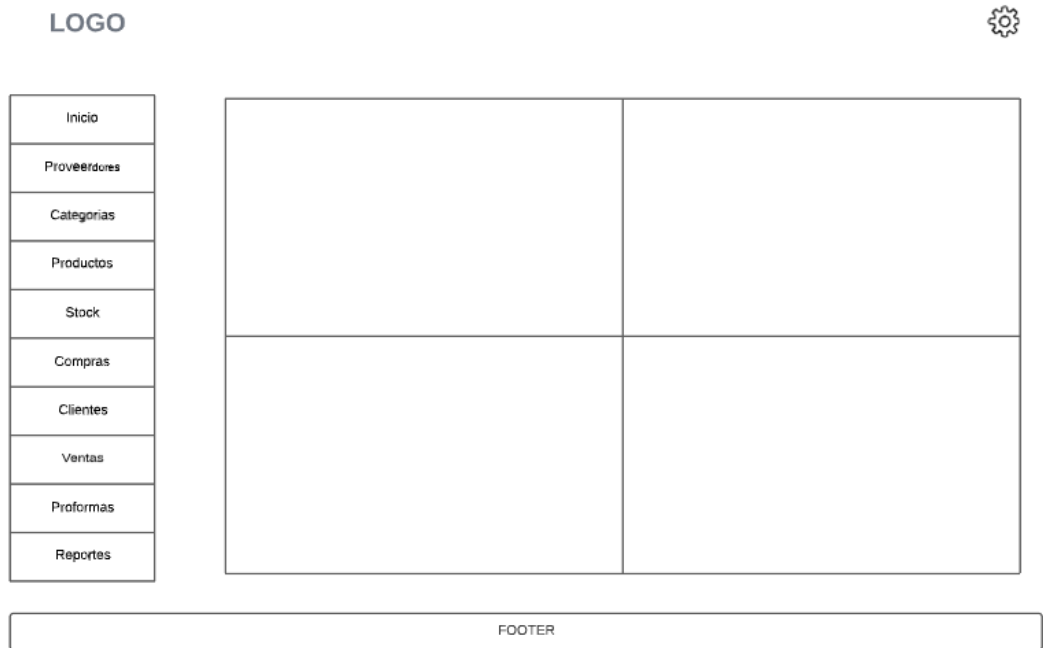


The image shows a wireframe for a user login page. At the top center is a logo consisting of a blue gear with a white cross inside, and the text 'FERRETERÍA FERRIMAR' to its right. Below the logo are three input fields: a single-line field labeled 'Usuario', a double-line field labeled 'Contraseña', and a single-line button labeled 'Entrar'.

*Imagen 10: Boceto de diseño Login Usuario
Elaborado por: Investigador*

Diseño Pantalla de Inicio

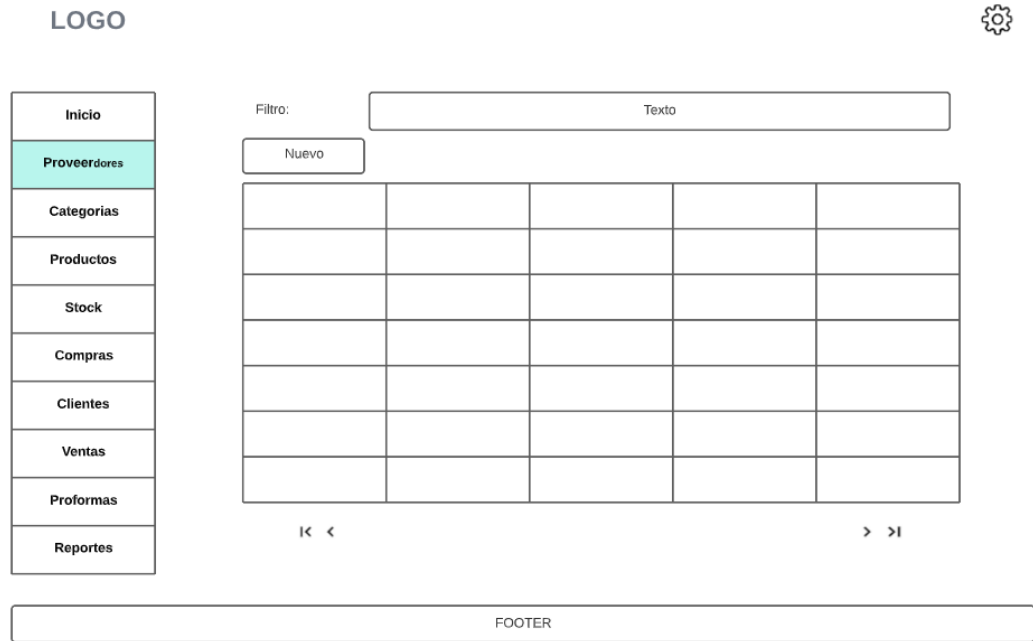
Es una pantalla de inicio, donde se presentará un resumen de datos que se encuentran almacenados en el sistema. Esta pantalla es únicamente informativa, aquí el usuario no deberá interactuar de manera alguna con la base de datos.



*Imagen 11: Boceto de diseño Pantalla de Inicio
Elaborado por: Investigador*

Diseño Pantalla Proveedores

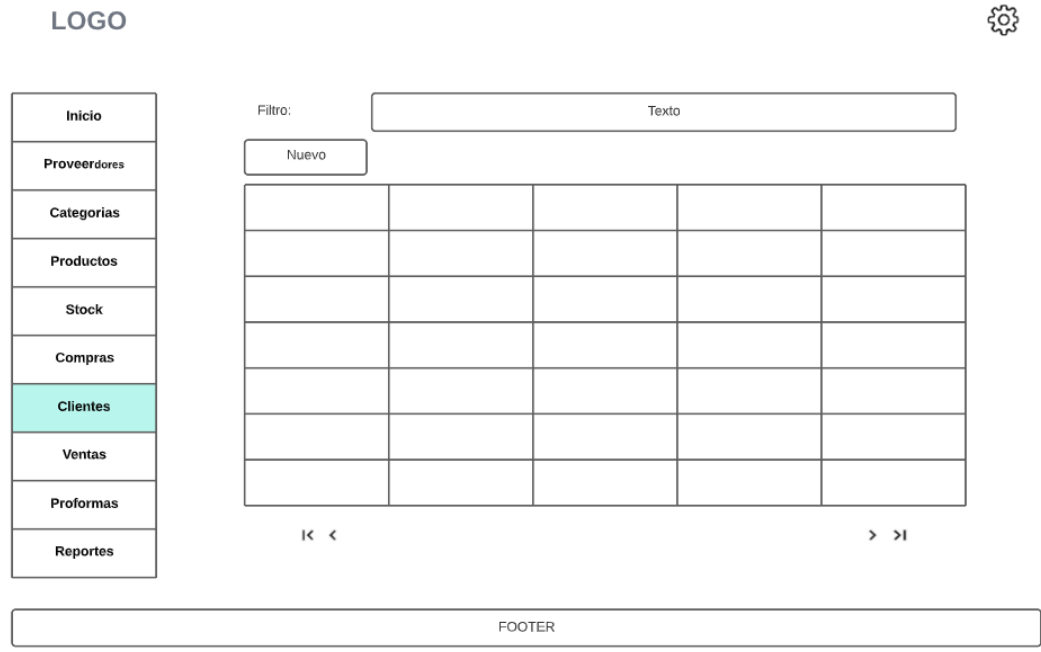
Como segundo ítem del menú se encuentra la interfaz de proveedores, donde se mostrarán todos listados con los datos considerados como más importantes. Se podrá visualizar todos los datos seleccionando una fila. Presenta también los botones de crear, editar y eliminar.



*Imagen 12: Boceto de diseño Pantalla de Proveedores
Elaborado por: Investigador*

Diseño Pantalla Clientes

Se encontrarán listados todos los clientes activos en la base de datos. Presenta también los botones de nuevo, editar y eliminar.



*Imagen 13: Boceto de diseño Pantalla Clientes
Elaborado por: Investigador*

Diseño Pantalla Stock y Mercadería

Es la interfaz más importante, aquí se mostrarán todos los productos ingresados en la base de datos y su información. Tiene las opciones de ingresar mercadería, la cual puede ser mediante una factura o un documento de entrada (no controlado). La opción de ver los precios que posee cada producto y depende del rol que posea el usuario también puede editarlos. Por último, las opciones de editar o eliminar un producto.

LOGO



Inicio	Filtro: <input type="text" value="Texto"/>																																													
Proveedores	<input type="button" value="Nuevo"/>																																													
Categorías	<table border="1"><tr><td></td><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td><td></td></tr></table>																																													
Productos																																														
Stock																																														
Compras																																														
Clientes																																														
Ventas																																														
Proformas																																														
Reportes																																														

⏪ < > ⏩

FOOTER

*Imagen 14: Boceto de diseño Pantalla de Productos
Elaborado por: Investigador*

LOGO



Inicio	Filtro: <input type="text" value="Texto"/>																																													
Proveedores	<input type="button" value="Nuevo"/>																																													
Categorías	<table border="1"><tr><td></td><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td><td></td></tr></table>																																													
Productos																																														
Stock																																														
Compras																																														
Clientes																																														
Ventas																																														
Proformas																																														
Reportes																																														

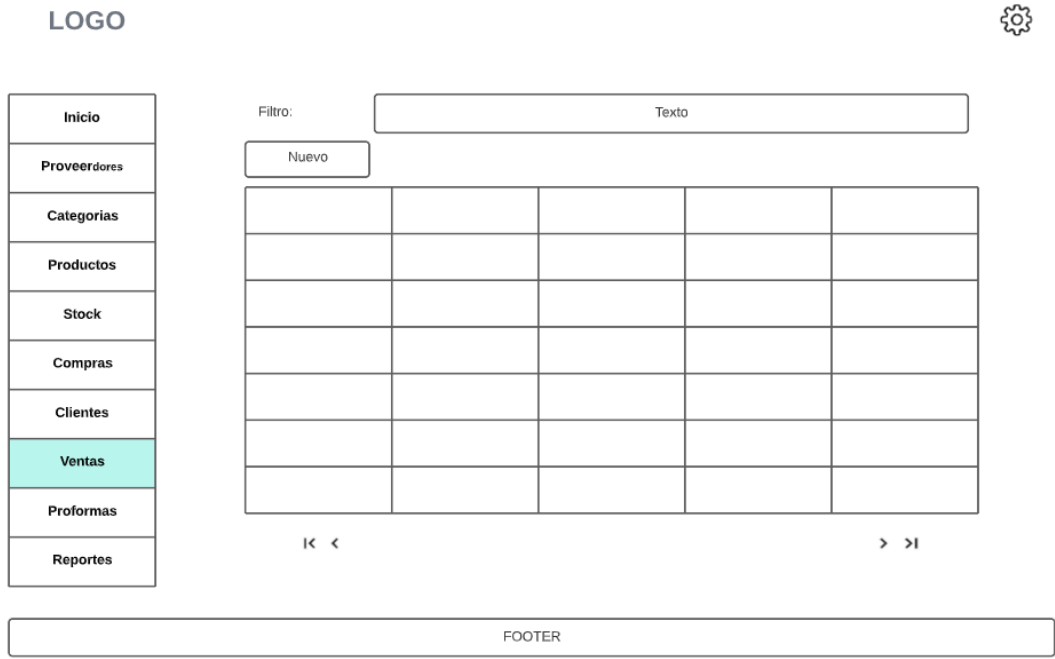
⏪ < > ⏩

FOOTER

*Imagen 15: Boceto de diseño Pantalla de Stock
Elaborado por: Investigador*

Diseño Pantallas Ventas y Proformas

En la opción de ventas se desplegará la interfaz de facturación. Muestra los campos necesarios para crear una nueva factura, si el usuario desea ver el listado completo de las facturas generadas debe hacer clic sobre la opción ´ver facturas´.



*Imagen 16: Boceto de diseño Pantalla Facturas
Elaborado por: Investigador*

En este mismo apartado del menú existirá la opción de proformas, donde el usuario puede generar una nueva cotización o, si desea ver el listado completo de documentos generados deberá hacer clic en el botón de ver proformas. Se podrá también facturar a partir de una proforma generada.

LOGO



Inicio	Filtro: <input type="text" value="Texto"/>																																								
Proveedores	<input type="button" value="Nuevo"/>																																								
Categorías	<table border="1"><tr><td></td><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td><td></td></tr></table>																																								
Productos																																									
Stock																																									
Compras																																									
Cientes																																									
Ventas																																									
Proformas																																									
Reportes																																									

|< < > >|

FOOTER

*Imagen 17: Boceto de diseño Pantalla Proformas
Elaborado por: Investigador*

Diseño Pantalla Reportes

Se tiene planificado crear dos interfaces para reportes en las cuales las pantallas serán similares a las que se muestra a continuación en la Figura 18. Tendrá inicialmente el espacio para ingresar los parámetros y luego la opción de Descargar.

LOGO



*Imagen 18: Boceto de diseño Pantalla de Reportes
Elaborado por: Investigador*

3.1.1.3. Fase de desarrollo

Creación de la Conexión a la base de datos (backend)

Se necesita crear una conexión al gestor MySQL Workbench, para lo cual se utilizó la siguiente configuración:

Archivo .env

```
PORT=3006
DB_NAME="ferrimar"
DB_USER="root"
DB_PASS="109012"
SECRETORPRIVATEKEY=Est03sMyPublicK3y23@2903
```

Archivo dbConnection.ts

```
import dotenv from "dotenv";
import setupModels from "./schemas";
import { Sequelize } from 'sequelize';

dotenv.config();

const dbConnection:Sequelize = new Sequelize(process.env.DB_NAME as
string,process.env.DB_USER as string,process.env.DB_PASS as string, {
  host : "localhost",
  dialect: "mysql",
  port: 3306,
  define: {
    freezeTableName: true,
    timestamps: false,
  }
});

setupModels(dbConnection);
dbConnection.sync();

export default dbConnection
```

Métodos para usuarios

Generar Token

Cuando un usuario inicia sesión en la aplicación utilizando sus credenciales (usuario y contraseña), la aplicación genera un token de sesión único que se asocia con el usuario. Cada vez que el usuario realiza una acción en la aplicación se requiere de dicha

autenticación, por lo que el token se envía junto con la solicitud al servidor para validar la identidad del usuario.

Archivo authRepository.ts

```
import Jwt from "jsonwebtoken"
import bcryptjs from "bcrypt"
import { Role } from "../../common/schemas/rol-schema";

export class DaoAuthenticationRepo{
  validatePassword(password: string,hashPass?:string): boolean {
    const validPass = bcryptjs.compareSync(password,hashPass!);
    if(!validPass){
      return false;
    }
    return true;
  }
  async generateToken (nickname: string,idRol:number) {
    const role = await Role.findByPk(idRol);
    return new Promise((resolve,reject) =>{
      const payload = nickname;
      Jwt.sign({payload},process.env.SECRETORPRIVATEKEY as
string,{
        expiresIn:"1d",
        algorithm:'HS256'
      }),(err,token)=>{
        if(err){
          console.log(err);
          reject('No se pudo generar el token');
        }else{
          const data = {
            nickname,
            message:'genial',
            token,
            role:role?.dataValues.name
          }
          resolve(data);
        }
      })
    })
  }
}
```

Archivo validar-token.ts

```
import { Request, Response } from "express"
```

```

import Jwt from "jsonwebtoken"
import dotenv from 'dotenv';
dotenv.config();

const validarJWT = async (req:Request, res:Response ,next: any)
=>{

    const token = req.header('Authorization');

    if(!token){
        return res.status(401).json({
            msg: 'No se encuentra autorizado para esta petición'
        });
    }
    try {
        Jwt.verify(
            token,
            process.env.SECRETORPRIVATEKEY as string,
            (error: any, decoded) => {
                if (error) {
                    throw new Error("Invalid authorization token");
                }
                console.log(decoded);
                return next();
            },
        );
    } catch (error) {
        console.log(error);
        res.status(401).json({
            msg: 'Token no válido'
        })
    }
    return;
}
export {
    validarJWT
}

```

Luego, en la programación de frontend para el login de usuario el resultado fue el siguiente:



Ferrimar

Nombre Usuario

El nombre de usuario es proporcionado por el administrador.

Contraseña

INGRESAR

*Imagen 19: Pantalla final login
Elaborado por: Investigador*

login.component.html

```
<form>
  <div>
    
    <h1 style="text-align: center;">Ferrimar</h1>

  </div>
  <div class="form-group">
    <label for="exampleInputEmail1">Usuario</label>
    <input type="email" class="form-control"
id="exampleInputEmail1" [(ngModel)]="nickname" aria-
describedby="emailHelp" placeholder="Enter email">
    <small id="emailHelp" class="form-text text-muted">El nombre
de usuario es proporcionado por el administrador.</small>
  </div>
  <div class="form-group">
    <label for="exampleInputPassword1">Contraseña</label>
    <input type="password" class="form-control"
[(ngModel)]="password" id="exampleInputPassword1"
placeholder="Password">
  </div>
  <button type="submit" class="btn btn-primary"
(click)="auth()">Ingresar</button>
</form>
```

CRUD

Las clases genéricas para CRUD son herramientas que se utilizan en la programación orientada a objetos para facilitar el acceso y manejo de datos en una base de datos. Se creó una serie de clases que contienen métodos predefinidos para realizar estas operaciones básicas en la base de datos.

Archivo dao-crud-generico.ts

```
import { Model } from "sequelize";
class DaoCrudGeneric<I> {

  private model!:any;
  private condition?:any;

  constructor(model:Model,condition?:any){
    this.model = model;
    this.condition = condition;
  }

  public async getItem():Promise<I[]>{
    const items = await this.model.findAll();
    return items;
  }
  public async findPkItem(id:number){
    const item = await this.model.findByPk(id);
    return item
  }

  public async saveItem(data:I,transactional?:any){
    const item = await
this.model.create(data,{transaction:transactional});
    return item;
  }

  public async updateItem(data:I,transactional?:any){
    console.log(transactional);

    const item = await this.model.update(data,this.condition);
    return item;
  }

  public async deleteItem(id:number){
    const item = await this.model.findByPk(id);
```

```

        if(!item){
            throw new Error("El item ingresado no existe");
        }
        await item.destroy();

        return true;
    }

    public async deleteByStatus(id:number){
        const item = await this.model.findByPk(id);

        if(!item){
            throw new Error("El item ingresado no existe");
        }
        await item.update({status:"INACTIVO"})

        return true;
    }
}

export default DaoCrudGeneric

```

Con esta clase creada, cada uno de los modulos va gestionando sus operaciones tomando los metodos generales que se encuentran aquí.

Modulo Productos

Este módulo es uno de los más importantes ya que parte de la creación de productos el cálculo de stock de los mismos. Una vez creados ya se pueden ir realizando operaciones como compras y ventas.

Archivo productService.ts

```
import { HttpStatus } from '../../../utils/enums/httpStatusEnum';
import { MessagesSuccess } from
 '../../../utils/enums/messagesSuccessEnum';
import { ServiceResponse } from
 '../../../common/interfaces/httpResponsesInterface';
import { IProduct } from '../../../common/interfaces/product';
import { DaoProductsRepo } from '../repository/daoProductRepository';
import { OperationsInventory } from
 '../../../inventory/utils/operationsInventory';
import { InventoryService } from
 '../../../inventory/services/inventoryService';
export class ProductsService{

    public async getProducts():Promise<ServiceResponse<IProduct>>{
        const daoProducts = new DaoProductsRepo();
        const res          = await daoProducts.getItem();
        const result: ServiceResponse<IProduct> = {
            httpStatus: HttpStatus.OK,
            message: MessagesSuccess.CONSULT,
            listData: res,
        };
        return result;
    }

    public async
saveProducts(data:IProduct,transaction:any):Promise<ServiceResponse<IPro
duct>>{
        const daoProducts = new DaoProductsRepo();
        const existProductByName = await
daoProducts.validExistProduct(data);

        if(existProductByName){
            throw new Error("Ya existe el producto con el nombre
ingresado")
        }
    }
}
```

```

        const res = await daoProducts.saveItem(data,transaction);
        const operationInventory = new OperationsInventory();
        const inventoryBuild =
await operationInventory.buildObjectCreateInventory(res.dataValues.idPr
oduct,data.name);
        const inventoryService = new InventoryService();
        const respInventory = await
inventoryService.saveInventory(inventoryBuild,transaction);
        if(!respInventory) {
            throw new Error("Ha ocurrido un error al crear el producto o
el inventario");
        }

        const result:ServiceResponse<IProduct> ={
            httpStatus:HttpStatus.CREATED,
            message:MessagesSuccess.CREATED,
            data:res.dataValues
        }
        return result;
    }
    public async
updateProducts(data:IProduct):Promise<ServiceResponse<IProduct>>{
        const condition = { where: { idProduct: data.idProduct } };
        const daoProducts = new DaoProductsRepo(condition);
        const existProductById = await
daoProducts.findPkItem(data.idProduct);
        const existProduct = await daoProducts.validExistProduct(data);
        if(!existProductById){
            throw new Error("No se pudo encontrar el producto
ingresado")
        }
        if(existProduct){
            throw new Error("Ya existe el producto con el nombre
ingresado")
        }
        await daoProducts.updateItem(data);
        const inventoryService = new InventoryService();
        await inventoryService.updateName(data.name,data.idProduct);
        const result:ServiceResponse<IProduct> ={
            httpStatus:HttpStatus.OK,
            message:MessagesSuccess.UPDATED
        }
        return result;
    }
    public async
deleteProducts(id:number):Promise<ServiceResponse<IProduct>>{

```

```

const daoProductsRepo = new DaoProductsRepo();
await daoProductsRepo.deleteByStatus(id);

const result: ServiceResponse<IProduct> = {
  httpStatus: HttpStatus.OK,
  message: MessagesSuccess.DELETED,
};
return result;
}

public async findProductById(idProduct:number){
  const daoProductsRepo = new DaoProductsRepo();

  const resp = await daoProductsRepo.findPkItem(idProduct);
  if(!resp){
    throw new Error("El producto ingresado no existe")
  }
  return resp.dataValues;
}


public async findProductByIdService(idProduct:number){
  const daoProductsRepo = new DaoProductsRepo();



  const resp = await daoProductsRepo.findPkItem(idProduct);
  if(!resp){
    throw new Error("El producto ingresado no existe")
  }
  const result:ServiceResponse<IProduct> ={
    httpStatus:HttpStatus.OK,
    data:resp,
    message:MessagesSuccess.CONSLT
  }
  return result;
}
}

```

En la siguiente interfaz se muestra el listado de todos los productos almacenados en la base de datos, contiene la caja de filtro y el botón de añadir, el cual desplegará el modal de edición o creación de productos.

Buscar:

Añadir + 

ID PRODUCTO	NOMBRE	PRECIO	MARCA	ESTADO	ACCIONES
1	ADAPTADOR FLEX	\$1.00	VIKINGO	ACTIVO	 

« 1 (current) »

4 items por página ▾

*Imagen 20: Pantalla final lista productos
Elaborado por: Investigador*

Archivo productos.component.html

```

<table class="tbuscar">
  <tr>
    <td>Buscar:</td>
    <td><input (keyup)="searchProduct($event)" /></td>
  </tr>
</table>

<div class="wrapper">

  <article class="main">

    <div class="content-button">
      <button class="button_add" (click)="addOrEditProduct()">Añadir
        <span class="material-icons" >
          add
        </span>
      </button>
    </div>
    <table class="table table-bordered">
      <thead>
        <tr>
          <th scope="col">Id Producto</th>
          <th scope="col">Nombre</th>
          <th scope="col">Precio</th>
          <th scope="col">Marca</th>
          <th scope="col">Estado</th>
        </tr>
      </thead>
    </table>
  </article>
</div>

```

```

        <th scope="col">Acciones</th>
    </tr>
</thead>
<tbody *ngFor="Let product of searchedProducts">
    <tr>
        <th scope="row">{{product.idProduct}}</th>
        <td>{{product.name}}</td>
        <td>{{product.price | currency}}</td>
        <td>{{product.brand}}</td>
        <td>{{product.status}}</td>
        <td scope="col" >
            <div class="content_button_actions">
                <button class="button_edit"
ngbTooltip="Editar" (click)="addOrEditProduct(product)">
                    <span class="material-icons">
                        mode_edit
                    </span>
                </button>

                <button class="button_delete" ngbTooltip="Eliminar"
(click)="deleteProduct(product.idProduct)">
                    <span class="material-icons">
                        delete
                    </span>
                </button>
            </div>
        </td>
    </tr>
</tbody>

</table>
<div class="d-flex justify-content-between p-2">

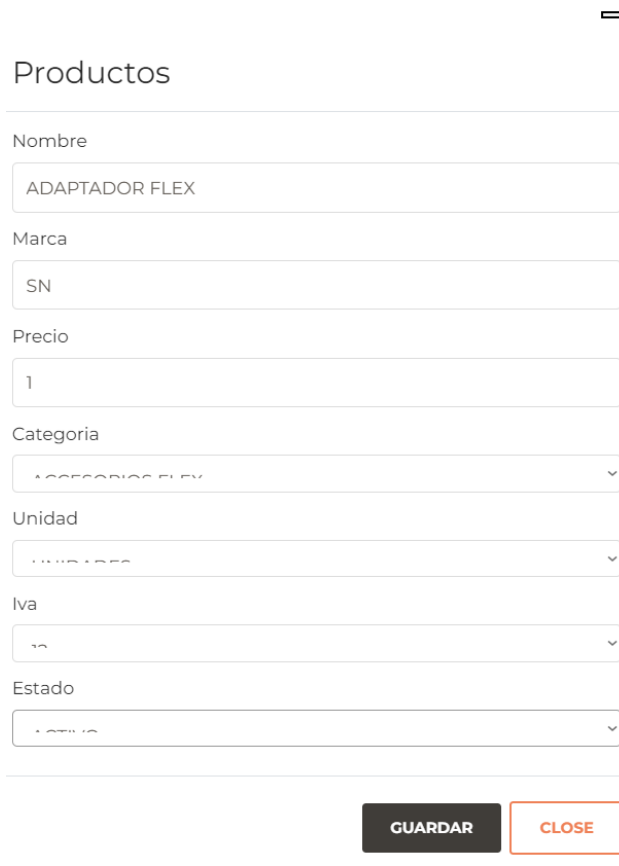
    <ngb-pagination
    [collectionSize]="collectionSize"
    [(page)]="page"
    [pageSize]="pageSize"
    (pageChange)="refreshProducts()"
    >
    </ngb-pagination>
    <select class="form-select" style="width: auto"
[(ngModel)]="pageSize" (ngModelChange)="refreshProducts()">
        <option [ngValue]="2">2 items por página</option>
        <option [ngValue]="4">4 items por página</option>
        <option [ngValue]="6">6 items por página</option>
    </select>

```



```
</div>
</article>
</div>
```

Para la edición y creación de los productos se muestra el siguiente modal con los datos que se requiere.



Productos

Nombre
ADAPTADOR FLEX

Marca
SN

Precio
1

Categoría
ACCESORIOS FLEX

Unidad
UNIDADES

Iva
12

Estado
ACTIVO

GUARDAR CLOSE

*Imagen 21: Pantalla final modal productos
Elaborado por: Investigador*

modal-productos.component.html

```
<div class="modal-header">
  <h4 class="modal-title">Productos</h4>
  <button type="button" class="btn-close" aria-label="Close"
(click)="activeModal.dismiss('Cross click')"></button>
</div>
<div class="modal-body">
  <form [formGroup]="formProduct">
```

```

<div class="form-group">
  <label for="formGroupExampleInput">Nombre</label>
  <input type="text" class="form-control"
id="formGroupExampleInput" formControlName="name" oninput="this.value =
this.value.toUpperCase()" placeholder="Ingrese el nombre del producto">
</div>
<div class="form-group">
  <label for="formGroupExampleInput">Marca</label>
  <input type="text" class="form-control"
id="formGroupExampleInput" formControlName="brand" oninput="this.value
= this.value.toUpperCase()" placeholder="Ingrese la marca del
producto">
</div>

<div class="form-group">
  <label for="formGroupExampleInput">Precio</label>
  <input type="number" class="form-control"
id="formGroupExampleInput" formControlName="price" placeholder="Ingrese
el precio del producto">
</div>

<div class="form-group">
  <label for="inputState" class="form-label">Categoria</label>
  <select id="inputState" class="form-control"
formControlName="category">
    <option selected>Seleccione la categoria</option>
    <option *ngFor="let category of categories"
[value]="category.idCategory">{{category.name}}</option>
  </select>
</div>

<div class="form-group">
  <label for="inputState" class="form-label">Unidad</label>
  <select id="inputState" class="form-control"
formControlName="unity">
    <option selected>Selecciona la unidad</option>
    <option *ngFor="let unity of unities"
[value]="unity.idUnity">{{unity.description}}</option>
  </select>
</div>

<div class="form-group">
  <label for="inputState" class="form-label">Iva</label>
  <select id="inputState" class="form-control"
formControlName="iva">
    <option selected>Selecciona el iva</option>

```

```

        <option *ngFor="Let iva of iva"
[value]="iva.idIva">{{iva.value}}</option>
    </select>
</div>

    <div class="form-group">
        <label for="inputState" class="form-label">Estado</label>
        <select id="inputState" class="form-control"
formControlName="status">
            <option selected>Elige un estado</option>
            <option *ngFor="Let status of statuses"
[value]="status">{{status}}</option>
        </select>
    </div>
</form>
</div>
<div class="modal-footer">
    <button *ngIf="!product" type="button" class="btn colorButton"
[disabled]="formProduct.invalid"
(click)="saveProduct()">Guardar</button>
    <button *ngIf="product" type="button" class="btn colorButton"
[disabled]="formProduct.invalid" (click)="editProduct()">Editar</button>
    <button type="button" class="btn btn-outline-danger"
(click)="activeModal.close('Close click')">Close</button>

</div>

```

Modulo Stock

En la siguiente imagen se muestra la interfaz de listado de todos los productos junto con su cantidad en stock y la suma de precio aproximado en todos los materiales.



ID INVENTARIO	STOCK	DESCRIPCIÓN	PRECIO ESTIMADO	ESTADO
1	0	:ADAPTADOR FLEX	\$0.00	ACTIVO

*Imagen 22: Pantalla final lista stock
Elaborado por: Investigador*

```
<table class="tbuscar">
  <tr>
    <td>Buscar:</td>
    <td><input (keyup)="searchStock($event)"/></td>
  </tr>
</table>

<div class="wrapper">

  <article class="main">

    <table class="table table-bordered">
      <thead>
        <tr>
          <th scope="col">Id Inventario</th>
          <th scope="col">Stock</th>
          <th scope="col">Descripción</th>
          <th scope="col">Precio Estimado</th>
          <th scope="col">Estado</th>
        </tr>
      </thead>
      <tbody *ngFor="Let inventory of searchedInventories">
        <tr>
          <th scope="row">{{inventory.idInventory}}</th>
```

```

        <td>{{inventory.stock }}</td>
        <td>{{inventory.description }}</td>
        <td>{{inventory.estimatedPrice | currency}}</td>
        <td>{{inventory.status}}</td>
        <!-- <td >Eliminar</td> -->
    </tr>

</tbody>
</table>
<div class="d-flex justify-content-between p-2">

    <ngb-pagination
    [collectionSize]="collectionSize"
    [(page)]="page"
    [pageSize]="pageSize"
    (pageChange)="refreshInventory()"
    >
</ngb-pagination>
    <select class="form-select" style="width: auto"
    [(ngModel)]="pageSize" (ngModelChange)="refreshInventory()">
        <option [ngValue]="2">2 items por página</option>
        <option [ngValue]="4">4 items por página</option>
        <option [ngValue]="6">6 items por página</option>
    </select>
</div>
</article>
</div>

```

Para los cálculos de stock se realizan operaciones tanto en compras y ventas, por lo cual la adición y sustracción de cantidades en stock se realizan en estas clases.

purchaseDetailService.ts

El siguiente código muestra el código generado para backend, el cual cuando ingresa una compra se adiciona en el stock de cada producto las cantidades compadas.

```

import { IDetailPurchase } from '../../common/interfaces/purchase';
import { DaoPurchaseDetailRepository } from
'../repositories/purchaseDetailRepository';
import { InventoryService } from
'../../inventory/services/inventoryService';
import { ProductsService } from
'../../products/services/productService';

```

```

import { ServiceResponse } from
'../../../../common/interfaces/httpResponsesInterface';
import { HttpStatus } from '../../../../utils/enums/httpStatusEnum';
import { MessagesSuccess } from
'../../../../utils/enums/messagesSuccessEnum';
export class PurchaseDetailService {

    public async buildObjectPurchaseDetail(data: IDetailPurchase[],
idPurchaseHeader: number) {
        data.map(data => data.idPurchaseHeader = idPurchaseHeader);
        return data;
    }

    public async savePurchaseDetail(data:
IDetailPurchase[],transaction:any) {

        const daoDetailPurchase = new DaoPurchaseDetailRepository();
        const inventoryService = new InventoryService();
        const productService = new ProductsService();
        const arrayPromises = await data.map(async data => {
            await productService.findProductById(data.idProduct);
        })
        const resp = Promise.all(arrayPromises).then(async datas => {
            console.log(datas);
            await
daoDetailPurchase.saveMultipleDetails(data,transaction);
            await inventoryService.addProductToInventory(data);
        }).catch(err => this.errorInPromise(err))
        console.log(resp);
        return resp;
    }
    public errorInPromise(err: any) {
        throw new Error(err.message)
    }
}

    public async getById(id:number):Promise<any>{
        const daoDetailPurchase = new DaoPurchaseDetailRepository();
        const res = await daoDetailPurchase.getByIdPurchase(id);

        const result: ServiceResponse<IDetailPurchase> = {
            httpStatus: HttpStatus.OK,
            message: MessagesSuccess.CONSUULT,
            listData: res as unknown as IDetailPurchase[],
        };
        return result;
    }
}

```

purchaseDetailService.ts

El siguiente código muestra el código generado para backend, el cual cuando se crea una venta, se resta en el stock de cada producto las cantidades vendidas.

```
import { InventoryService } from
'../../inventory/services/inventoryService';
import { ProductsService } from
'../../products/services/productService';
import { ISaleDetail } from '../../common/interfaces/sales';
import { DaoSaleDetailRepository } from
'../repository/saleDetailRepository';
import { ServiceResponse } from
'../../../../common/interfaces/httpResponsesInterface';
import { HttpStatus } from '../../../../utils/enums/httpStatusEnum';
import { MessagesSuccess } from
'../../../../utils/enums/messagesSuccessEnum';
export class SaleDetailService {

    public async buildObjectSaleDetail(data: ISaleDetail[],
idSaleHeader: number) {
        data.map(data => data.idSaleHeader = idSaleHeader);
        return data;
    }

    public async saveSaleDetail(data: ISaleDetail[],transaction:any) {

        const daoDetailSale = new DaoSaleDetailRepository();
        const inventoryService = new InventoryService();
        const productService = new ProductsService();
        const arrayPromises = await data.map(async data => {
            await productService.findProductById(data.idProduct);
        })
        const resp = Promise.all(arrayPromises).then(async datas => {
            console.log(datas);
            await daoDetailSale.saveMultipleDetails(data,transaction);
            await
inventoryService.subtractProductOfInventory(data).catch(err =>
this.errorInPromise(err));
        }).catch(err => this.errorInPromise(err))
        console.log(resp);
        return resp;
    }

    public errorInPromise(err: any) {
        throw new Error(err.message)
    }
}
```

```
}  
  
public async getById(id:number):Promise<any>{  
    const daoDetailSale = new DaoSaleDetailRepository();  
    const res = await daoDetailSale.getByIdSale(id);  
  
    const result: ServiceResponse<ISaleDetail> = {  
        httpStatus: HttpStatus.OK,  
        message: MessagesSuccess.CONSUULT,  
        listData: res as unknown as ISaleDetail[],  
    };  
    return result;  
}  
}
```


3.1.1.4.Fase de pruebas

Pruebas de aceptación

La fase de pruebas evalúa y se verifica que la aplicación final funciona de una manera correcta y devuelve los resultados esperados. Se realiza junto con los usuarios finales para obtener la aprobación de estos y se basan en las historias de usuario planteadas.

Para cada una de las pruebas se hará uso del siguiente formato.

Prueba de aceptación			
Numero:		Identificador de Historia:	
Nombre:			
Descripción			
Condición de ejecución:			
Pasos:			
Resultado esperado:			
Resultado:			

*Tabla 103: Formato tarjeta de prueba de aceptación
Elaborado por: Investigador*

Donde:

Número: Representa el número de cada prueba realizada.

Identificador de historia: Es el número de historia de usuario a la que corresponde la prueba.

Nombre: Es la denominación de cada prueba.

Descripción: Breve descripción de la actividad.

Condiciones de Ejecución: Aquí se especifica las condiciones para que la prueba sea considerada como Satisfactoria.

Pasos: Es el conjunto de métodos que hay que seguir para realizar una actividad.

Resultado Esperado: Resultados obtenidos al realizar la prueba.

Evaluación de la prueba: Resultado final, este puede ser calificado como Satisfactorio o No Satisfactorio.

Los resultados obtenidos durante las pruebas de aceptación se muestran a continuación.

Prueba de aceptación			
Numero:	01	Identificador de Historia:	002
Nombre:	Inicio de sesión		
Descripción:	Validación de credenciales		
Condición de ejecución:	El usuario deberá ingresar sus credenciales que son nombre de usuario y contraseña.		
Pasos:	Ingresar en los campos de texto un usuario y una contraseña, luego hacer clic en el botón de ingreso		
Resultado esperado:	Valida si los datos ingresados existen en la base de datos y son correctos. Si esto se cumple entonces da paso al menú principal, sino advierte de que las credenciales son incorrectas.		
Resultado:	Satisfactorio		

*Tabla 104: Prueba de aceptación 01
Elaborado por: Investigador*

Prueba de aceptación			
Numero:	02	Identificador de Historia:	004
Nombre:	Pantalla proveedores		
Descripción:	Gestión de la tabla de proveedores		
Condición de ejecución:	Despliega una interfaz donde se muestren los proveedores existentes listados junto con sus datos. Si desea agregar uno nuevo, aparecerán los campos requeridos, si desea editar se desplegarán los campos completos con la opción de editarlos. Y para eliminar uno, el sistema pedirá la confirmación de la acción		

Pasos:	Presionar sobre el botón de cada opción para verificar la consulta, creación, actualización y eliminación de datos.
Resultado esperado:	La interfaz muestra correctamente el listado, y los campos para agregar, editar. En todas las acciones mostrará un mensaje para verificar si se realizó o no la actividad.
Resultado:	Satisfactorio

Tabla 105: Prueba de aceptación 02

Elaborado por: Investigador

Prueba de aceptación			
Numero:	03	Identificador de Historia:	005
Nombre:	Pantalla clientes		
Descripción:	Gestión de la tabla de clientes		
Condición de ejecución:	La aplicación deberá mostrar una interfaz donde se muestren todos los clientes existentes listados junto con sus datos. Si desea agregar uno nuevo, aparecerán los campos requeridos, si desea editar se desplegarán los campos completos con la opción de editarlos. Y para eliminar uno, el sistema pedirá la confirmación de la acción		
Pasos:	Presionar sobre el botón de cada opción para verificar la consulta, creación, actualización y eliminación de datos.		
Resultado esperado:	La interfaz muestra correctamente el listado, y los campos para agregar, editar. En todas las acciones mostrará un mensaje para verificar si se realizó o no la actividad.		
Resultado:	Satisfactorio		

Tabla 106: Prueba de aceptación 03

Elaborado por: Investigador

Prueba de aceptación			
Numero:	04	Identificador de Historia:	006
Nombre:	Pantalla productos		
Descripción:	Gestión de la tabla de productos		
Condición de ejecución:	La aplicación deberá mostrar una interfaz donde se muestren todos los productos existentes listados junto con sus datos. Si desea agregar uno nuevo, aparecerán los campos requeridos, si desea editar se desplegarán los campos completos con la opción de editarlos. Y para eliminar uno, el sistema pedirá la confirmación de la acción.		
Pasos:	Presionar sobre el botón de cada opción para verificar la consulta, creación, actualización y eliminación de datos.		
Resultado esperado:	La interfaz muestra correctamente el listado, y los campos para agregar, editar. En todas las acciones mostrará un mensaje para verificar si se realizó o no la actividad.		
Resultado:	Satisfactorio		

*Tabla 107: Prueba de aceptación 04
Elaborado por: Investigador*

Prueba de aceptación			
Numero:	05	Identificador de Historia:	006
Nombre:	Pantalla compras		
Descripción:	Gestión de facturas de compra		
Condición de ejecución:	Se muestra una interfaz donde se listen las facturas de compra ingresadas. Si desea agregar uno nuevo, aparecerán los campos requeridos, No existe la opción de editar ya que no se deben manipular los datos de esta tabla, sin embargo, se tiene la opción de anular la factura de compra.		

Pasos:	Presionar sobre el botón de cada opción para verificar la consulta, creación, y anulación del documento.
Resultado esperado:	La interfaz muestra correctamente el listado, y los campos para agregar y eliminar. En todas las acciones mostrará un mensaje para verificar si se realizó o no la actividad.
Resultado:	Satisfactorio

*Tabla 108: Prueba de aceptación 05
Elaborado por: Investigador*

Prueba de aceptación			
Numero:	06	Identificador de Historia:	006
Nombre:	Pantalla documento de entrada/salida		
Descripción:	Gestión de los documentos de entrada/salida		
Condición de ejecución:	Se muestra una interfaz donde se listen los documentos ingresados. Tiene únicamente la opción de agregar y anular.		
Pasos:	Verificar el listado de documentos y su información. Se debe verificar también la funcionalidad de los botones agregar y eliminar.		
Resultado esperado:	La interfaz muestra correctamente el listado, los campos para agregar dentro de una pantalla modal y eliminar. En todas las acciones mostrará un mensaje para verificar si se realizó o no la actividad.		
Resultado:	Satisfactorio		

*Tabla 109: Prueba de aceptación 06
Elaborado por: Investigador*

Prueba de aceptación			
Numero:	07	Identificador de Historia:	007
Nombre:	Pantalla proformas		
Descripción:	Gestión de proformas		

Condición de ejecución:	Se muestra una interfaz inicial donde se listan las cotizaciones generadas. La aplicación deberá mostrar los campos necesarios para crear una nueva y devolver los mensajes adecuados.
Pasos:	Presionar el botón de nuevo para ingresar un nuevo registro. También existe la opción de editar las cotizaciones sin embargo será únicamente de precios. Se pedirá la confirmación de los datos.
Resultado esperado:	La interfaz muestra correctamente el listado, los campos para agregar son los correctos y devuelve los mensajes para indicar si se realizó o no la actividad.
Resultado:	Satisfactorio

*Tabla 110: Prueba de aceptación 07
Elaborado por: Investigador*

Prueba de aceptación			
Numero:	08	Identificador de Historia:	007
Nombre:	Pantalla ventas		
Descripción:	Gestión de facturas de venta		
Condición de ejecución:	La aplicación deberá mostrar una interfaz donde se listen todas las facturas emitidas existentes junto con sus datos. Si desea agregar una nueva, aparecerán los campos requeridos, no existe la opción de editar ya que estas solo tienen la opción de anular. Y para toda acción, el sistema pedirá la confirmación de esta.		
Pasos:	Presionar el botón de nuevo para ingresar un nuevo registro. También existe la opción de anular las facturas, esto devolverá al stock los productos. Se pedirá la confirmación de los datos.		
Resultado esperado:	La interfaz muestra correctamente el listado, los campos para agregar son los correctos y devuelve los mensajes para indicar si se realizó o no la actividad.		

Resultado:	Satisfactorio
------------	---------------

*Tabla 111: Prueba de aceptación 08
Elaborado por: Investigador*

Prueba de aceptación			
Numero:	09	Identificador de Historia:	008
Nombre:	Vista de reporte 1		
Descripción:	Reporte ventas		
Condición de ejecución:	La aplicación debe mostrar según las fechas la información del monto de ventas que se ha generado.		
Pasos:	Indicar el tipo de reporte que desea visualizar en la barra superior		
Resultado esperado:	Si existen datos en la base de datos, la aplicación devolverá de manera grafica la información de una manera ordenada por montos y fechas.		
Resultado:	Satisfactorio		

*Tabla 112: Prueba de aceptación 09
Elaborado por: Investigador*

Prueba de aceptación			
Numero:	010	Identificador de Historia:	008
Nombre:	Vista de reporte 2		
Descripción:	Reporte compras		
Condición de ejecución:	La aplicación debe mostrar según las fechas la información del monto de compras que se ha generado.		
Pasos:	Indicar el tipo de reporte que desea visualizar en la barra superior		
Resultado esperado:	Si existen datos en la base de datos, la aplicación devolverá de manera grafica la información de una manera ordenada por montos y fechas.		
Resultado:	Satisfactorio		

*Tabla 113: Prueba de aceptación 010
Elaborado por: Investigador*

Pruebas de usabilidad

Con el fin de verificar la adaptabilidad de la Aplicación Web Progresiva a los distintos tipos de usuarios, se aplicó el siguiente test obteniendo los resultados a continuación:



UNIVERSIDAD TÉCNICA DE AMBATO
FACULTA DE INGENIERÍA EN SISTEMAS
ELECTRÓNICA E INDUSTRIAL



CARRERA DE INGENIERÍA EN SISTEMAS COMPUTACIONALES E
INFORMÁTICOS

Test de usabilidad realizada al gerente/propietario de la ferretería Ferrimar

Elegir una valoración según el rango que corresponda:

- Totalmente de acuerdo = 5
- De acuerdo = 4
- Regular = 3
- En desacuerdo = 2
- Totalmente en desacuerdo = 1

No.	Afirmación	Evaluación				
		1	2	3	4	5
1	La combinación de colores y estructura de las interfaces es adecuado y agradable para el usuario				X	
2	Los módulos que contiene la aplicación están correctamente indicados y no son ambiguos					X
3	Existen las opciones para deshacer sus pasos al navegar					X
4	La aplicación contiene mensajes de confirmación para prevención de errores					X

5	Cada elemento que se encuentra en las interfaces es relevante					X
6	Se comprende de manera clara el resumen de stock, materiales y precios					X
7	La aplicación indica errores de manera comprensible sin lenguaje técnico				X	
8	Al usuario le gusta manejar la aplicación					X

*Tabla 114: Test prueba de usabilidad
Elaborado por: Investigador*

3.1.1.5.Fase de lanzamiento

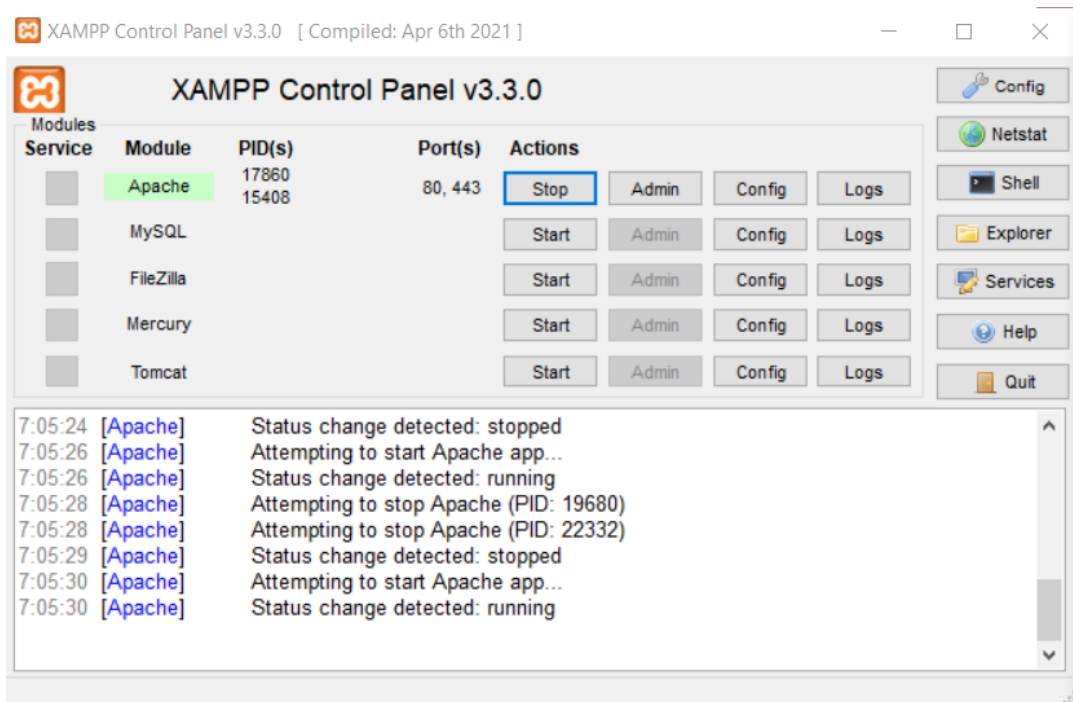
Para implantar la aplicación Web Progresiva es necesario un dominio, ya sea público o local para alojar los archivos. Ferretería Ferrimar indicó que no posee un dominio público por lo que se decidió instalar localmente el sistema, con el fin de que los usuarios hagan uso de la aplicación, y en un futuro, la empresa invierta en un servidor público.

Configuración de equipos

XAMPP es un paquete de software libre y gratuito el cual proporciona un ambiente de servidor web completo para sistemas operativos como Windows, Linux y Mac. Incluye un servidor web Apache por lo que se utilizó esta herramienta para alojar el proyecto.

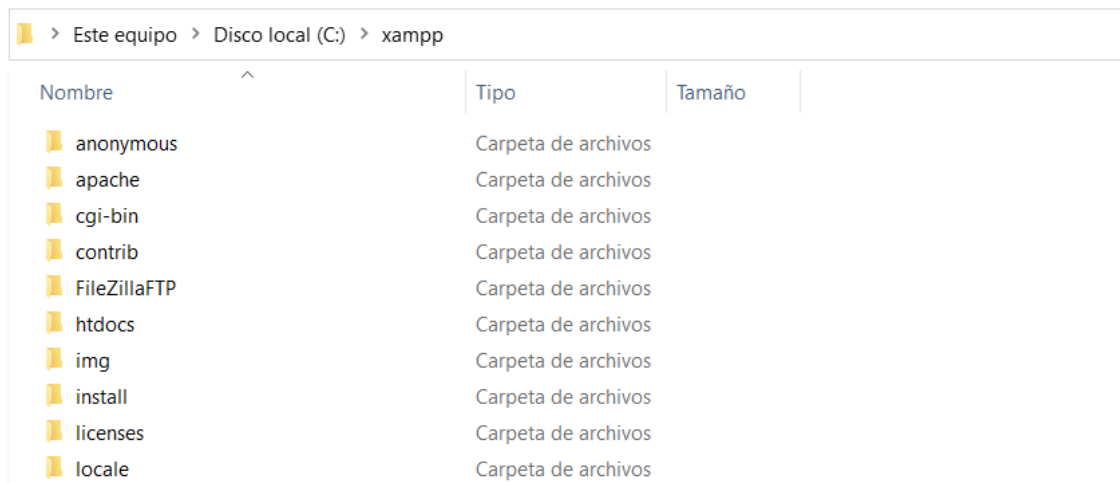
Se descargó la versión de XAMPP 8.0.9 para Windows desde el sitio web oficial: <https://www.apachefriends.org/index.html>. Cuando se ejecutó el archivo descargado para iniciar el instalador de XAMPP, en el proceso de instalación, se seleccionó los componentes Apache, MySQL, PHP, phpMyAdmin y se mantuvo la carpeta por defecto que ofrece el instalador.

Una vez instalada la herramienta se desplegó su panel de control, donde se debe iniciar la opción Apache como se muestra a continuación.



*Imagen 23: Panel de control de XAMPP
Elaborado por: Investigador*

Ingresa a en el icono Explorer, el cual desplegará el directorio local de XAMPP



*Imagen 24: Directorio de XAMPP
Elaborado por: Investigador*

Aquí se encuentra la carpeta *htdocs*, donde se deberá pegar la aplicación, tanto el servidor como el cliente, como se indica en la siguiente imagen.

Nombre	Tipo	Etiquetas	Autores
Backend	Carpeta de archivos		
Frontend	Carpeta de archivos		

*Imagen 25: Aplicación instalada localmente
Elaborado por: Investigador*

CAPITULO IV

CONCLUSIONES Y RECOMENDACIONES

4.1.Conclusiones

Al culminar el presente proyecto se implementó una PWA que permite el control de inventarios en la empresa FERRIMAR, teniendo las siguientes conclusiones:

Las entrevistas realizadas a los empleados de la empresa y la observación de campo que se ejecutó, se obtuvo toda la información necesaria para entender el proceso del manejo de mercadería y de esta manera esquematizar diagramas que ayudaron en análisis para posteriormente ejecutar el desarrollo.

Con ayuda de la aplicación desarrollada, se simplificó el proceso para mantener al día la información de stock y productos que se manejaba en Ferrimar, ya que el proceso era únicamente manual, debido a esto existían errores manuales y calculados en los datos.

Se realizó el estudio de múltiples tecnologías para el desarrollo de Aplicaciones Web Progresivas, eligiendo Angular como framework de desarrollo. Debido a su arquitectura MVC permitió el desarrollo ágil, y ayudó a separar notablemente las tareas que cada componente debe realizar y así lograr una aplicación más limpia y de fácil mantenimiento.

La implantación y aplicación final se desarrollaron con éxito, siendo bien recibida por parte del gerente propietario de ferretería Ferrimar, lo cual pudo ser evidenciado en las pruebas de aceptación de usuario.

4.2.Recomendaciones

Se sugiere seguir evaluando en el futuro la gestión de los demás procesos como control del transporte y logística, seguimiento de empleados, etc. para, de ser el caso agregar estos posibles módulos a la aplicación web ya que sería de gran ayuda gestionar sus actividades a partir de una sola plataforma.

Se recomienda realizar mantenimientos periódicos a la base de datos, además de realizar respaldos, ya que la entidad maneja una gran cantidad de productos y la información que maneja de entrada y salida de mercadería es importante.

La persona que asumirá el rol de administrador de la aplicación será el gerente, por lo que se sugiere una capacitación adecuada del uso del sistema para manejar información adecuada por parte de los empleados.

Se deberá tener los navegadores actualizados para el buen funcionamiento tanto del frontend como backend. Así como también se recomienda tener actualizadas las herramientas de desarrollo, debido a que constantemente los desarrolladores están corrigiendo errores o incluso optimizando tareas y solucionando problemas de compatibilidad.

MATERIALES DE REFERENCIA

Referencias Bibliográficas

- [1] J. Águila y J. Monguet, Por qué algunas empresas tienen éxito y otras no, Barcelona, España: Ediciones Deusto, 2010.
- [2] A. Fierro y F. Fierro, Contabilidad de activos con enfoque NIF para pymes, Bogotá, Colombia: ECOE EDICIONES, 2015.
- [3] adapptative, «Progressive Web Apps (PWA), la revolución tecnológica de la web,» 2 Agosto 2018. [En línea]. Available: <https://adapptative.com/pwa-revolucion-tecnologica/>.
- [4] R. M. Apunte y R. A. Rodríguez, «Diseño y aplicación de sistema de gestión en Inventarios en empresa ecuatoriana,» *Ciencias Holguín*, vol. 22, n° 3, pp. 1-14, 2016.
- [5] O. P. Carlos Veloz, «Métodos para mejorar la eficiencia y la toma de decisiones en la gestión de inventarios,» *Revista Ciencia UNEMI*, vol. 10, n° 22, p. 38, 2017.
- [6] P. Panchi-Mayo, R. Armas-Heredia y F. C. Solórzano, «LOS INVENTARIOS Y EL COSTO DE PRODUCCIÓN EN LAS EMPRESAS INDUSTRIALES DEL ECUADOR,» *ROCA. Revista científico -educacional de la provincia Granma.*, vol. 13, n° 4, pp. 254 - 264, 2017.
- [7] J. D. M. Romero, «SISTEMA DE CONTROL Y SEGUIMIENTO DE LOS PROCESOS DE RECURSOS HUMANOS PARA LOS EMPLEADOS DE LA EMPRESA ELÉCTRICA AMBATO REGIONAL CENTRO NORTE S. A. MEDIANTE UNA APLICACIÓN WEB PROGRESIVA (PWA,» Universidad Técnica de Ambato, Ambato, 2022.
- [8] W. O. Calderón, «IMPLEMENTACIÓN DE UNA APLICACIÓN WEB PROGRESIVA PARA LA GESTIÓN DE PRUEBAS DE SIMULACION PARA EL INGRESO A INSTITUCIONES MILITARES Y UNIVERSIDADES EN EL

- CENTRO DE CAPACITACION Y NIVELACION ACADEMICA SMARTEL,»
Universidad Técnica de Ambato, Ambato, 2020.
- [9] M. R. Zambrano, «DESARROLLO DE UNA TIENDA VIRTUAL PROGRESSIVE WEB APPS (WPA) PARA GESTIONAR LAS VENTAS DE LOS PRODUCTOS EN LA EMPRESA GARCÉS TORRES (GT) JEAN'S CUPIDO,» Universidad Técnica de Ambato, Ambato, 2021.
- [10] L. E. A. Aldaz, «SISTEMA WEB PARA EL CONTROL DE FACTURACIÓN E INVENTARIO DE MEDICAMENTOS Y BIENES EN EL HOSPITAL REGIONAL DOCENTE AMBATO,» Universidad Técnica de Ambato, Ambato, 2013.
- [11] B. L. Vega, «DESARROLLO E IMPLEMENTACION DE APLICACION WEB PARA EL CONTROL DE INVENTARIO DEL LOCAL COMERCIAL MAQUILAS HIDALGO,» Universidad Técnica de Ambato, Ambato, 2017.
- [12] «LOS MÉTODOS DE CONTROL DE INVENTARIOS MÁS USADOS,» ESERP Business & Law School, [En línea]. Available: <https://es.eserp.com/articulos/metodo-control-inventarios/>.
- [13] J. A. Arévalo, «Gestión de la Información, gestión de contenidos y conocimiento,» SIOU, Salamanca, España, 2007.
- [14] M. V. Karina Maldonado, DISEÑO DE LA GESTIÓN LOGÍSTICA DE LA BODEGA DE PRODUCTO TERMINADO DE LA DIVISION DE CARTON EN PAPELERA NACIONAL S.A, Guayaquil: Universidad Politecnica Salesiana, 2011.
- [15] F. Flores, «Qué es Visual Studio Code y qué ventajas ofrece,» OpenWebinars, 22 07 2022. [En línea]. Available: <https://openwebinars.net/blog/que-es-visual-studio-code-y-que-ventajas-ofrece/>.
- [16] S. Baez, «KnowDo,» 20 10 2012. [En línea]. Available: <http://www.knowdo.org/knowledge/39-sistemas-web>.
- [17] A. O. L. P. T. L. C. G. P. P. V. Aguirre, «PWA para unificar el desarrollo Desktop, Web y Mobile,» Instituto de Investigación en Informática LIDI , La Plata, Argentina, 2019.

- [18] I. Ramirez, «¿Qué es una Aplicación Web Progresiva o PWA?,» Xataka, 03 Julio 2018. [En línea]. Available: <https://www.xataka.com/basics/que-es-una-aplicacion-web-progresiva-o-pwa>.
- [19] D. Sheppard, Beginning Progressive Web App Development, Illinois, USA: Apress, 2017.
- [20] Zahabia, «GKM,» 28 Enero 2021. [En línea]. Available: <https://www.gkmit.co/blog/mobile-development/progressive-web-apps-pwas-the-future-of-mobile-web-apps>.
- [21] Google, «Introduction to Progressive Web App Architectures,» 25 02 2021. [En línea]. Available: <https://developers.google.com/web/ilt/pwa/introduction-to-progressive-web-app-architectures>.
- [22] Upplication, «Progressive Web App, La nueva manera de entender las Apps,» 2018. [En línea]. Available: https://www.upplication.com/images/pwa/upplication_PWA_2018_es-ES.pdf.
- [23] A. Eripret, «Progressive Web Apps (PWA) ¿Qué son?,» WPO Web Performance Optimization, [En línea]. Available: <https://seocom.agency/es/blog/introduccion-pwa/>.
- [24] J. C. FATJÓ, «Desarrollo de app web, app móvil, híbrida y PWA. ¿Cuál elegir?,» TRIBALYTE, 15 Enero 2021. [En línea]. Available: https://tech.tribalyte.eu/blog-desarrollo-app-web-movil-hibrida-pwa#Un_error_muy_comun.
- [25] acens, Framework para el desarrollo ágil de, 2014.
- [26] D. Ortego, «OpenWebinars,» 14 Marzo 2017. [En línea]. Available: <https://openwebinars.net/blog/los-6-mejores-frameworks-javascript/>.
- [27] D. Pacheco, «Los 10 frameworks de desarrollo web más solicitados en 2021,» [En línea]. Available: Los 10 frameworks de desarrollo web más solicitados en 2021.
- [28] B. D. C. D. Brad Dayley, Node.js, MongoDB and Angular. Web Development, USA: Pearson Education, Inc, 2018.

- [29] J. H. Skeie, Ember.js in Action, Manning, 2014.
- [30] Catrian, «Facilidad de desarrollo con Ember.js,» 21 Enero 2016. [En línea]. Available: <https://www.catrian.com/facilidad-de-desarrollo-con-ember-js/#:~:text=uno%20de%20ellos.-,Ember.,del%20estado%20de%20la%20aplicación..>
- [31] A. F. Mestres, Introduccion al frontend y backend, Barcelona, España: Universitat Oberta de Catalunya.
- [32] C. Gor, «Which One You Should Opt For In 2022?,» EsparkBiz, 18 Mayo 2020. [En línea]. Available: <https://www.esparkinfo.com/blog/laravel-vs-django-vs-nodejs.html>.
- [33] F. J. Mory, «Metodologías Ágiles vs Tradicionales,» Revista Empresarial & Laboral , [En línea]. Available: <https://revistaempresarial.com/tecnologia/metodologias-agiles-vs-tradicionales/#:~:text=Los%20sistemas%20tradicionales%20se%20centran,los%20clientes%20y%20la%20flexibilidad..>

ANEXO B: MUESTRA DE FACTURA DE COMPRA

"CONTRIBUYENTE ESPECIAL" IMPORTADOR FERRETERO TRUJILLO CIA.LTDA. LET. CIA. LTDA.
EL MUNDO FERRETERO A TU SERVICIO

Quito-Matriz: CHILGALLO - S59 SN Y CALLE E2C, Cel: 0998457447
 Sucursal Guayaquil: TARDUI - VIA DAULE SOLAR 1 Y CALLE 13, Cel: 0967594694
 Sucursal Cuenca: YANUNCAI - LOMA SIN Y DIEGO DE BAZA, Cel: 0992270848
 Sucursal Santo Domingo: RIO VERDE - AV DE EL COOPERATIVISMO SIN Y CALLE 02, Cel: 0990431849

RUC: 1791932455001

C. I.: 006-001-000042609 **FACTURA PROVISIONAL No. 006-001-00**

CLIENTE: MARTINEZ JUAN FRANCISCO COD. CLI: 00601

DIRECCION: HUACHI GRANDE VIA A TISALEO BARRIO EL CENSO CIUDAD: AMBATO

FECHA Y HORA DE AUTORIZ.: -BARRIO EL CE

NUMERO DE AUTORIZACION: 1411202201179193245500120060010000426090004260910 EMISOR: PRODUCCION

CLAVE DE ACCESO: 1411202201179193245500120060010000426090004260910 FECHA DE EMISION: 14/Noviembre/2022

FORMAS DE PAGO: TRANS CON UTILIZACION SISTEMA FINANCIERO 333,44 E-juanfrancismartinez@hotmail.com TIPO DE PAGO: -Tipo_cli: 98

0042609

RUC: 1802699247001

TELF: 032441892

VENDEDOR: JY

EMISION: INDF. SISTEMA

TIPO DE PAGO: -Tipo_cli: 98

CODIGO.....	CANTIDAD.	UNID.	ED DESCRIPCION.....	PRECIO...	DSCTO %	FREC-DSCTO	VAL_DSCTO	SUBTOTAL....
CL-CL050X2.8	2.00	CJA	01-ABELCA CLAV 50X2.80 2X12 C/C 2"	48.5625	9.00 %	44.19	8.74	88.39 49.50
CL-CL065X3.45	1.00	CJA	01-ABELCA CLAV 65X3.45 21/2X10C/C 1/2"	45.5000	8.00 %	41.86	3.64	41.86 46.88
155653	3.00	UND	01-BLL PALA CDR NYPLAST CLASTIC	8.9018	0.00 %	8.90	0.00	26.71 9.97
BR-BR0506	12.00	UND	01-BROCHA CHMO C/BLANCO 3"	0.8639	0.00 %	0.88	0.00	10.61 1.00
GR-E266-03	6.00	UND	01-FV J/ANG.INDD.E266.03DHC/M 16	9.9536	7.00 %	9.26	4.18	55.54 10.70
GR-J60001	6.00	UND	01-FV J/ANG.LAV.E262.05DH C/M 16	9.9536	7.00 %	9.26	4.18	55.54
GN-CHDQ/L	24.00	PAR	01 GUANTE COMANDO NITRILLO L	0.7946	0.00 %	0.79	0.00	19.07 0.890

ESTE ES UN DOCUMENTO INTERNO NO VÁLIDO PARA EFECTOS TRIBUTARIOS Y PODRÁ CONSULTAR SU FACTURA RELACIONADA EN ESTE DOCUMENTO EN EL PORTAL DEL SRI.

FAVOR CANCELAR ÚNICAMENTE CON CHEQUE CRUZADO A FAVOR DE LA EMPRESA - EXIJA SU RECIBO DE COBRO, ÚNICO DOCUMENTO PARA LA CANCELACIÓN DE LA MISMA.

Me comprometo a pagar IMPORTADOR FERRETERO TRUJILLO CIA. LTDA. el valor de la FACTURA relacionada con este documento interno en las condiciones de estos establecimientos. En caso de mera pagara intereses máximo permitido por la Ley. La mercadería antes mencionada, ha recibido conforme y en perfectas condiciones, declaro no tener ningún reclamo que hacer por el estado de conservación ni por la cantidad de las mismas. Las mercaderías de la presente factura es de propiedad de IMPORTADOR FERRETERO TRUJILLO CIA. LTDA. Quien se reserva el dominio de las mismas hasta su cancelación total y viajan por cuenta y riesgo del comprador no aceptamos ningún tipo de reclamo por mermas, pérdidas, roturas u otros daños después de que las mercaderías hayan salido de nuestras bodegas. La devolución de las mercaderías es inmediata (24 horas), caso contrario le daremos por recibida a conformarse, solo en perfecto estado, embalaje original y con todos sus accesorios.


	<p>VENTA CON TARIFA 0% 0.00</p> <p>VENTA CON TARIFA 2% 297.71</p> <p>IVA 2% 35.73</p> <p>TOTAL 333.44</p>
--	--

DESTINATARIO DESPACHADOR

NETBOOK -14/11/22-08:26:40-MIRIAN WACA N 14/11/22 14:09:07 REMITIR RETENCION A: cartera_ambato@importadortrujillo.com.ec

USUARIO:

ANEXO C: MUESTRA DE FACTURA DE VENTA



Ferretería FERRIMAR

MARTINEZ MANCHENO JUAN FRANCISCO

Dirección: Barrio Jacinto Buchelli
Calle Alaska s/n y Vía a Tisaleo
Tel. 03-2441892 - 0998802054
e-mail: pfranciscomartinez@hotmail.com
Ambato - Ecuador

RUC. 1802699247001

FACTURA
001-001-

0010273

AUT. S.R.I. 1130166516


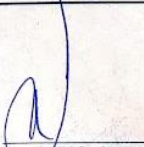
CONTRIBUYENTE RÉGIMEN RIMPE


CIUDAD	DÍA	MES	AÑO
AMBATO	30	07	2022

Cliente Sr(es): CONSONACIO SALVA UTA

Dirección: AMBATO FRANCO BAULA Y HERMANO MIGUEL

Ruc / CI: 1891307100001 Telf.: 032559734 Guía de Remisión:

CANT.	DESCRIPCIÓN	V. UNIT.	V. TOTAL
5	VOLOQUETAS RIPIO TRITURADO	120,00	600,00
5	VOLOQUETAS ANEXA NEGRA	120,00	600,00
200	SACOS CEMENTO CHINGONAZO	7,60	1520,00
2500	FLORES HACIZO DE 12 cm	0,40	1000,00
5	MALLA ANEXA 5-5-15	63,60	308,30
10	99 VANILLA ADELCA 12x12	55,90	559,00
5	99 VANILLA ADELCA 8x12	55,90	279,50
5	99 VANILLA ADELCA 10x12	55,95	279,25
1	ROLLO ALAMBRE NEGRO # 18	38,00	38,00
1	BARRA "HERRAGÓN"	22,00	22,00
1	PICO "BELLOTA"	13,00	13,00
1	PALA "BELLOTA"	13,00	13,00
1	DISCO CORTE HOANIGOM 7"	9,51	9,51
FORMA DE PAGO		SUB TOTAL 12 % 11868,90	
Efectivo <input type="checkbox"/>		SUB TOTAL 0 %	
Dinero <input type="checkbox"/>		DESCUENTO	
Electrónico <input type="checkbox"/>		S. TRANSPORTE	
Tarjeta de Crédito / Débito <input type="checkbox"/>		IVA 12 % 584,26	
Otros <input type="checkbox"/>		VALOR TOTAL 5453,06	
ORIGINAL ADQUIRENTE		<div style="display: flex; justify-content: space-around;"> <div style="text-align: center;">  <p>Entregado Conforme</p> </div> <div style="text-align: center;">  <p>Recibido Conforme</p> </div> </div>	
COPIA EMISOR			


CORDENO ALVARADO JUIS MIGUEL / RUC. 1801812725901 / AUT. 1262
 Cel: 994485333 * E-mail: offfstetun@hotmail.com * Impreso del 10151 al 10450
 Fecha de Autorización 06 / Julio / 2022 * Valido hasta el 06 / Julio / 2023

ANEXO D: MUESTRA DE PROFORMA FERRIMAR



Todo para su construcción

RUC: 1802699247001
Dir.: Huachi Grande; Av. Alaska S/N.
Ambato, Ecuador
244 1892 – 0998802054

PROFORMA N.º 000 001 564

FECHA: 07-Jun-2022

DATOS DEL CLIENTE

CI: 1802613206

Nombre: Carlos Salinas

Telf.: 0986383402

Dirección: Huachi Grande, Ambato

CANT.	DESCRIPCIÓN	V.UNITARIO	V.TOTAL
160	METROS PORCELANATO 50X5 BLU	\$ 21,00	\$ 3.360,00
32	METROS CERAMICA PISO COCINA	\$ 16,00	\$ 512,00
42	METROS CERAMICA PARED COCINA	\$ 14,80	\$ 621,60
64	METROS CERAMICA PISO BAÑO	\$ 17,00	\$ 1.088,00
45	METROS CERAMICA PARED BAÑO	\$ 17,00	\$ 765,00
4	INODORO BAÑO ROMA BLANCO	\$ 170,00	\$ 680,00
1	FREGADERO 2 POZOS FV	\$ 280,00	\$ 280,00
1	PIANCHA GRANITO COCINA	\$ 380,00	\$ 380,00
20	TUBO ROSCA 1/2 PL	\$ 9,00	\$ 180,00
9	TUBO CUATRITUBO 1/2	\$ 10,00	\$ 90,00
6	LLAVE ANGULAR LAV FV	\$ 14,00	\$ 84,00
4	LLAVE ANGULAR IMDO FV	\$ 14,00	\$ 56,00
70	CODOS 1/2 X90 PLA	\$ 0,60	\$ 42,00
30	UNIONES 1/2 PLA	\$ 0,90	\$ 27,00
27	TEE 1/2	\$ 0,80	\$ 21,60
10	LLAVES PASO 1/2 FV	\$ 15,00	\$ 150,00
5	SELLAPLUS 100CC	\$ 9,50	\$ 47,50
20	TEFLONES AA	\$ 1,00	\$ 20,00
10	UNIVERSALES 1/2 PLA	\$ 1,50	\$ 15,00
120	CEMENTO SF LVALGRE 50Kg	\$ 7,90	\$ 948,00
80	EMPASTES INTERIOR SIKA 20Kg	\$ 13,00	\$ 1.040,00
19	EMPASTES EXTERIOR SIKA 20Kg	\$ 26,00	\$ 494,00
75	BONDÉX PREMIUM 20Kg	\$ 14,00	\$ 1.050,00
1	CALFON INSTAMATIG 26 lt	\$ 380,00	\$ 380,00
3	CABIE SOLIDO #10 100m.	\$ 130,00	\$ 390,00
6	CABIE SOLIDO #10 100m.	\$ 60,00	\$ 360,00
4	CABIE SOLIDO #10 100m.	\$ 45,00	\$ 180,00
3	CAJA TERMICA 4 cc	\$ 45,00	\$ 135,00
4	BREAKER 40A P	\$ 8,50	\$ 34,00
4	BREAKER 50A P	\$ 8,75	\$ 35,00
4	MEZCLADORA DUCHA	\$ 87,00	\$ 348,00
18	CARBONATO T.A 25Kg	\$ 9,00	\$ 162,00
3	RESAF LEX CAMECA	\$ 95,00	\$ 285,00
4	PINTURA SATINADA BLANCA CANECA	\$ 145,00	\$ 580,00
2	TINA BAÑO CORONA	\$ 474,00	\$ 948,00
3	CABINA BAÑO	\$ 387,00	\$ 1.161,00
8	CHAPAS POMO GEO	\$ 21,00	\$ 168,00
4	MUEBLE BAÑO ESPEJO INEC	\$ 680,00	\$ 2.720,00
	GASTOS DE ENVÍO	\$ -	
	DESCTO.	\$ -	
TOTAL			\$19.837,70

**J. Francisco Martínez
GERENTE**

Esta proforma tiene una validez de 30 días hábiles.

ANEXO E: MUESTRA DE INVENTARIO MANUAL EN EXCEL DE FERRIMAR

	A	B	C	D	E	G	H	I
	INVENTARIO 2022							
	CATEGORIA	MARCA	NOMBRE	MEDIDAS/ESPECIFICACIONES	U.MEDIDA	CANTIDAD	PRECIO VENTA PUBLICO	TOTAL
4	FERRETERIA	TITAN	ABRAZADERA 10-06		UNIDADES	52	\$ 0,50	26,00
5	FERRETERIA	TITAN	ABRAZADERA 30-08		UNIDADES	238	\$ 0,85	202,30
15	FERRETERIA	SN	ABRAZADERA BLANCA CHINA	3/4	UNIDADES	118	\$ 0,20	23,60
16	FERRETERIA	SN	ABRAZADERA BLANCA CHINA	1 3/4 PULG.	UNIDADES	127	\$ 0,60	76,20
17	FERRETERIA	SN	ABRAZADERA BLANCA CHINA	2 PULG.	UNIDADES	164	\$ 0,40	65,60
18	FERRETERIA	SN	ABRAZADERA EMT	1 PULG.	UNIDADES	60	\$ 0,45	27,00
19	FERRETERIA	SN	ABRAZADERA EMT	1/2	UNIDADES	74	\$ 0,07	5,18
20	FERRETERIA	SN	ABRAZADERA EMT	1 PULG.	UNIDADES	177	\$ 0,25	44,25
21	FERRETERIA	SN	ABRAZADERA EMT	3/4	UNIDADES	180	\$ 0,10	18,00
22	FERRETERIA	TITAN	ABRAZADERA INDUSTRIAL	2 3/4	UNIDADES	4	\$ 4,00	16,00
23	FERRETERIA	TITAN	ABRAZADERA INDUSTRIAL	1 3/4 PULG.	UNIDADES	7	\$ 3,50	24,50
24	FERRETERIA	SN	ABRAZADERA PARA CABLE DE HIERRO	3/8	UNIDADES	67	\$ 0,40	26,80
25	FERRETERIA	TITAN	ABRAZADERA PEQUENA		UNIDADES	27	\$ 0,30	8,10
26	VALVULAS Y GRIFERIA	FV	ACCESORIO DE BANO PORCELANA BLANCO		UNIDADES	1	\$	-
27	FERRETERIA	WD-40	ACEITE	191 ml	UNIDADES	3	\$ 5,50	16,50
28	FERRETERIA	CASTROL	ACEITE 2 TIEMPOS	1 LITRO	UNIDADES	1	\$ 5,50	5,50
29	FERRETERIA	SN	ACEITE 3 EN 1 PEQUENO		UNIDADES	12	\$ 1,00	12,00
30	FERRETERIA	SN	ACEITERO METALICO ROJO		UNIDADES	2	\$ 3,50	7,00
31	CEMENTO Y EMPASTES	SIKA	ACELERANTE PARA HORMIGON 161 HE	4 KILOS	UNIDADES	0	\$ 11,50	
32	CEMENTO Y EMPASTES	SIKA	ACELERANTE PARA HORMIGON 161 HE	2 KILOS	UNIDADES	3	\$ 6,25	18,75
33	TUBERIAS Y ACCESORIOS	IPS	ACOPLE DE TANQUE	1/2	UNIDADES	1	\$ 2,50	2,50
34	TUBERIAS Y ACCESORIOS	IPS	ACOPLE DE TANQUE	1 PULG.	UNIDADES	1	\$ 4,50	4,50
35	VALVULAS Y GRIFERIA	SN	ACOPLE MEDIDOR DE AGUA BRONCE 2 JUEGOS		UNIDADES	2	\$ 3,50	7,00
36	VALVULAS Y GRIFERIA	SN	ACOPLES DE CAUCHO	1 1/4	UNIDADES	16	\$ 0,70	11,20
37	ACCESORIOS FLEX	SN	ADAPTADOR FLEX	1 PULG.	UNIDADES	13	\$ 1,00	13,00
38	ACCESORIOS FLEX	SN	ADAPTADOR FLEX	3/4	UNIDADES	15	\$ 0,75	11,25
39	ACCESORIOS FLEX	SN	ADAPTADOR FLEX	1 1/2	UNIDADES	18	\$ 0,90	16,20
40	ACCESORIOS FLEX	SN	ADAPTADOR FLEX	1 1/4	UNIDADES	27	\$ 1,00	27,00



MANUAL DE USUARIO
SISTEMA PARA EL MANEJO DE
INVENTARIO

Versión: 1.0.0
Elaborado por: Camila Martinez
Fecha de revisión: febrero 15, 2023

ANEXO F: MANUAL DE USUARIO



Manual de usuario para Aplicación de control de Inventario Ferrimar

Manual de instrucciones

- 2023 -



MANUAL DE USUARIO
SISTEMA PARA EL MANEJO DE
INVENTARIO

Versión: 1.0.0

Elaborado por: Camila Martínez

Fecha de revisión: febrero 15, 2023

Contenido

Inicio de Sesión	3
Gestión de usuarios	3
Módulo Proveedores	5
Módulo Clientes	7
Módulo Productos	10
Módulo Stock	12
Agregar Compras	12



MANUAL DE USUARIO SISTEMA PARA EL MANEJO DE INVENTARIO

Versión: 1.0.0
Elaborado por: Camila Martinez
Fecha de revisión: febrero 15, 2023

Inicio de Sesión

Una vez el administrador del sistema le haya proporcionado sus credenciales que son usuario y contraseña, puede hacer uso del sistema.



Ferrimar

Usuario

Contraseña

ACCEDER

Gestión de usuarios

La gestión de usuarios es un módulo al cual únicamente tiene acceso el administrador del sistema, que es el rol que asumirá el gerente. Aquí se pueden agregar nuevos usuarios o, por el contrario, eliminarlos.

Nombre de usuario:

[Añadir +](#)

ID USER	NOMBRE	CONTRASEÑA	ROL	ESTADO	ACCIONES
2	PCUEVA	\$2b\$10\$p.HATERem0bOM5InBsw03OKTjJwi3MMUaPDgTRRcENvH/JVYRZQeW	EMPLEADO	ACTIVO	
3	CAMARTINEZ	\$2b\$10\$Ke3btNYZYlcWqIEHb1UVL6tSA4YLazbSiIJ3Wg2UYIMXEvyI5Hxe	GERENTE	ACTIVO	

< 1 (current) >

4 items por página ▾

En la parte superior de la interfaz se encuentra la caja de texto para realizar búsquedas rápidas de usuario. Se puede filtrar según el nombre. Se aconseja al administrador crear los usuarios de acuerdo con un estándar, es decir la inicial de su nombre + el apellido. De esta manera se tendrá homogeneidad en las credenciales.



Nombre de usuario:

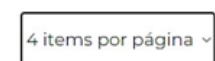
En la esquina superior derecha se encuentra el botón añadir, el cual despliega el modal para agregar nuevos usuarios



En todas las filas de la tabla, se encuentra el botón eliminar, que sirve para dar de baja un usuario



Debido a que se pueden crear gran cantidad de datos, existe la facilidad de agrupar la tabla según los datos, el siguiente botón brindara la opción de agrupar en 4, 8 o 12 filas la tabla.



4 items por página ▾

Una vez se presiona el botón de añadir, se despliega el siguiente modal, en el cual se debe ingresar los datos nombre, contraseña, estado (el cual siempre será activo) y el rol que cumplirá. Los roles disponibles son GERENTE, EMPLEADO, BODEGUERO, CONTADOR.



MANUAL DE USUARIO SISTEMA PARA EL MANEJO DE INVENTARIO

Versión: 1.0.0

Elaborado por: Camila Martinez

Fecha de revisión: febrero 15, 2023

Usuarios

Nombre

Contraseña

Estado

Rol

ACEPTAR CLOSE

Módulo Proveedores

El presente módulo está restringido para algunos roles de usuario, por lo tanto, si no aparece en el menú inicial se debe a que el usuario no tiene los permisos necesarios.

Nombre o RUC:

Añadir +

ID PROVEEDOR	RUC	NOMBRE	DIRECCIÓN	TELEFONO	ESTADO	ACCIONES
1	1803112018001	IMPORTADORA TRUJILLO	QUITO	0998058947	ACTIVO	
2	1802699247001	MADYS SA	AMBATO	0998058741	ACTIVO	
3	1954784233001	PROFERMACO	RIOBAMBA	0998058947	ACTIVO	
4	1325478966001	MEGAPROFER	AMBATO	0998054781	ACTIVO	

« 1 (current) »

4 items por página ▾

En la parte superior de la interfaz se encuentra la caja de texto para realizar búsquedas rápidas de usuario. Se puede filtrar según el nombre de la empresa o según el RUC.

Nombre o RUC:

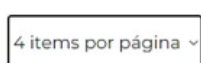
En la esquina superior derecha se encuentra el botón añadir, el cual despliega el modal para agregar nuevos proveedores



En todas las filas de la tabla, se encuentran en la columna acciones, los botones de editar y eliminar, que sirve para actualizar o cambiar información del proveedor o darlo de baja.



Debido a que se pueden crear gran cantidad de datos, existe la facilidad de agrupar la tabla según los datos, el siguiente botón brindara la opción de agrupar en 4, 8 o 12 filas la tabla.



Una vez se presiona el botón de añadir, se despliega el siguiente modal, en el cual se debe ingresar todos los datos de la compañía o empresa que son RUC, nombre, asesor comercial, dirección, teléfono del proveedor y del asesor, email.



MANUAL DE USUARIO
SISTEMA PARA EL MANEJO DE
INVENTARIO

Versión: 1.0.0
Elaborado por: Camila Martinez
Fecha de revisión: febrero 15, 2023

Proveedor

Ruc
1803112018001

Nombre
IMPORTADORA TRUJILLO

Asesor Comercial
SOFIA CASTRO

Dirección
QUITO

Telefono de Proveedor
0998058947

Telefono de Asesor
055478142

Email
MTRUJILLO@TRUJILLO.COM

Estado
COTACACHI

GUARDAR **CLOSE**

Una vez se revisó que los datos sean los correctos, el usuario puede hacer clic en guardar, caso contrario, para cancelar la acción debe hacer clic en cerrar.



Módulo Clientes

El presente módulo está restringido para algunos roles de usuario, por lo tanto, si no aparece en el menú inicial se debe a que el usuario no tiene los permisos necesarios.

Nombre o RUC:

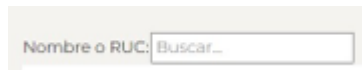
Añadir +

ID CLIENTE	IDENTIFICACIÓN	NOMBRES	APELLIDOS	DIRECCIÓN	TELEFONO	ESTADO	ACCIONES
1	1807040533	JOSE LEONARDO	MERA TORRES	PATATE	0995024785	ACTIVO	
2	1620425187	CAROLINA	RODRIGUEZ	AMBATO	0998052361	ACTIVO	

« 1 (current) »

4 items por página ▾

En la parte superior de la interfaz se encuentra la caja de texto para realizar búsquedas rápidas de usuario. Se puede filtrar según el nombre del cliente, los apellidos, CI o RUC.



Nombre o RUC:

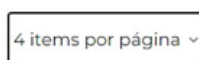
En la esquina superior derecha se encuentra el botón añadir, el cual despliega el modal para agregar nuevos clientes.



En todas las filas de la tabla, se encuentran en la columna acciones, los botones de editar y eliminar, que sirve para actualizar o cambiar información del cliente o darlo de baja.



Debido a que se pueden crear gran cantidad de datos, existe la facilidad de agrupar la tabla según los datos, el siguiente botón brindara la opción de agrupar en 4, 8 o 12 filas la tabla.



4 items por página ▾



MANUAL DE USUARIO
SISTEMA PARA EL MANEJO DE
INVENTARIO

Versión: 1.0.0
Elaborado por: Camila Martinez
Fecha de revisión: febrero 15, 2023

Una vez se presiona el botón de añadir, se despliega el siguiente modal, en el cual se debe ingresar todos los datos del cliente de la ferretería, el cual puede identificarse por RUC o cédula de identidad, deberá ingresar también nombres, apellidos, dirección, un número telefónico, y el email.

Cientes

Identificación

Nombres

Apellidos

Dirección

Teléfono

Email

Estado

GUARDAR **CLOSE**

Cuando se revisó que los datos sean los correctos, el usuario puede hacer clic en guardar.

GUARDAR

Módulo Productos

Para que un producto se refleje en stock, primero debe ser creado en la interfaz de productos para saber que consta en el inventario de la ferretería, aquí se tomaran datos más específicos del producto.

A continuación se presenta la interfaz donde se listan todos los productos que han sido ingresados a la base de datos



Buscar:

Añadir + 

ID PRODUCTO	NOMBRE	PRECIO	MARCA	ESTADO	ACCIONES
1	ADAPTADOR FLEX	\$1.00	VIKINGO	ACTIVO	 

« 1 (current) »

4 ítems por página ▾

En la parte superior de la interfaz se encuentra la caja de texto para realizar búsquedas rápidas. Se puede filtrar según el nombre del producto o la marca.

Buscar:

En la esquina superior derecha se encuentra el botón añadir, el cual despliega el modal para agregar nuevos productos.



En todas las filas de la tabla, se encuentran en la columna acciones, los botones de editar y eliminar, que sirve para actualizar o cambiar información del material o darlo de baja.



MANUAL DE USUARIO
SISTEMA PARA EL MANEJO DE
INVENTARIO

Versión: 1.0.0
Elaborado por: Camila Martinez
Fecha de revisión: febrero 15, 2023



Cuando presiona el botón de añadir, se despliega el siguiente modal, en el cual se debe ingresar todos los datos requeridos que son, nombre de producto, marca, precio aproximado de venta, a que tipo de categoría pertenece, tipo de IVA que grava dicho producto.

Productos

Nombre

Marca

Precio

Categoría

Unidad

Iva

Estado

Módulo Stock

Es el módulo más importante de la aplicación web debido a que aquí se refleja lo que se tiene al momento en stock en toda la bodega. Todos los usuarios tienen acceso a la interfaz de stock.

En esta interfaz se despliega los datos que se quiere conocer de los productos existentes al momento en stock, es decir la cantidad existente, los detalles del producto, y la cantidad en dólares estimada que se tiene de ese material. Los cálculos se realizan de manera automática una vez se agregan facturas de compra o de venta por lo que esta interfaz no tiene opciones de edición, eliminación u otros.



ID INVENTARIO	STOCK	DESCRIPCIÓN	PRECIO ESTIMADO	ESTADO
1	0	:ADAPTADOR FLEX	\$0.00	ACTIVO

« 1 (current) »

4 items por página ▾

Agregar Compras

Es importante mantener al día las facturas que llegan de productos arribados, por lo que hay que ingresarlas en el módulo de compras.

El siguiente modal muestra los datos que se deberán ingresar para crear facturas. Los cuales son número de factura, el proveedor el cual está facturando, los productos, cantidades, precio de compra. Automáticamente el sistema calculará el porcentaje IVA 12% de los productos que hayan sido ingresados como obligados a pagar IVA.



MANUAL DE USUARIO
SISTEMA PARA EL MANEJO DE
INVENTARIO

Versión: 1.0.0
Elaborado por: Camila Martinez
Fecha de revisión: febrero 15, 2023

Número de factura

Ingrese el número de factura

Email

Ingrese el email del proveedor

Dsto

Ingrese el descuento

Proveedor

Productos

Cantidad

Ingrese la cantidad

Precio Unitario

Ingrese el precio de compra

Iva

Añadir pedido +

Detalle del pedido

#	PRODUCTO	CANTIDAD	PRECIO UNITARIO	VALOR PARCIAL	ELIMINAR
---	----------	----------	-----------------	---------------	----------

Subtotal:0

Iva:0

Descuento:0

Total:0

Guardar pedido +