

UNIVERSIDAD TECNICA DEL NORTE



FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS CARRERA DE INGENIERÍA EN SISTEMAS COMPUTACIONALES

TEMA:

DESARROLLO DE UN SISTEMA DE GESTIÓN Y CONTROL DE PROCESOS PARA LA MICROEMPRESA BAZAR Y PAPELERÍA SAN ANTONIO UTILIZANDO SPRING BOOT.

Trabajo de grado previo a la obtención del título de Ingeniera
en Sistemas Computacionales.

AUTORA:

Paola Alexandra Tito Chulde.

DIRECTOR:

MSc. Fausto Alberto Salazar Fierro.

IBARRA – 2020.



UNIVERSIDAD TÉCNICA DEL NORTE

BIBLIOTECA UNIVERSITARIA

AUTORIZACIÓN DE USO Y PUBLICACIÓN A FAVOR DE LA UNIVERSIDAD TÉCNICA DEL NORTE

1. IDENTIFICACIÓN DE LA OBRA

En cumplimiento del Art. 144 de la Ley de Educación Superior, hago la entrega del presente trabajo a la Universidad Técnica del Norte para que sea publicado en el Repositorio Digital Institucional, para lo cual pongo a disposición la siguiente información:

DATOS DE CONTACTO			
CÉDULA DE IDENTIDAD:	100349348-1		
APELLIDOS Y NOMBRES:	TITO CHULDE PAOLA ALEXANDRA		
DIRECCIÓN:	SAN ANTONIO DE IBARRA, CAMILO POMPEYO Y 20 DE JULIO		
EMAIL:	patitoc@utn.edu.ec		
TELÉFONO FIJO:	062 933 566	TELÉFONO MÓVIL:	0993908694

DATOS DE LA OBRA	
TÍTULO:	DESARROLLO DE UN SISTEMA DE GESTIÓN Y CONTROL DE PROCESOS PARA LA MICROEMPRESA BAZAR Y PAPELERÍA SAN ANTONIO UTILIZANDO SPRING BOOT.
AUTOR (ES):	TITO CHULDE PAOLA ALEXANDRA
FECHA: DD/MM/AAAA	28/05/2020
SOLO PARA TRABAJOS DE GRADO	
PROGRAMA:	<input checked="" type="checkbox"/> PREGRADO <input type="checkbox"/> POSGRADO
TITULO POR EL QUE OPTA:	INGENIERA EN SISTEMAS COMPUTACIONALES
ASESOR /DIRECTOR:	MSC. FAUSTO SALAZAR

2. CONSTANCIAS

La autora manifiesta que la obra objeto de la presente autorización es original y se la desarrolló, sin violar derechos de autor de terceros, por lo tanto, la obra es original y que es la titular de los derechos patrimoniales, por lo que asume la responsabilidad sobre el contenido de la misma y saldrá en defensa de la Universidad en caso de reclamación por parte de terceros.

Ibarra, a los 28 días del mes de mayo del 2020

LA AUTORA:



PAOLA ALEXANDRA TITO CHULDE
CI. 100349348-1



UNIVERSIDAD TÉCNICA DEL NORTE

FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

Ibarra, 28 de mayo del 2020

CERTIFICACIÓN DEL DIRECTOR

Por medio del presente, yo MSc. Fausto Salazar, certifico que la Sra. Paola Alexandra Tito Chulde, portadora de la cédula de identidad Nro. 100349348-1. Ha trabajado en el desarrollo del proyecto de trabajo de grado denominado **“DESARROLLO DE UN SISTEMA DE GESTIÓN Y CONTROL DE PROCESOS PARA LA MICROEMPRESA BAZAR Y PAPELERÍA SAN ANTONIO UTILIZANDO SPRING BOOT”**, previo a la obtención del título de Ingeniera en Sistemas Computacionales, lo cual ha realizado en su totalidad con responsabilidad.

Es todo cuanto puedo certificar en honor a la verdad.

Atentamente,

MSc. Fausto Salazar
DIRECTOR DE TESIS



**PAPELERIA
SAN ANTONIO**
LÍDERES EN CALIDAD Y ECONOMÍA

SRA. GLADYS VIZCAINO
PROPIETARIA
095 915 7847
062 932 241
Camilo Pompeyo Guzman
(Junto al Colegio Daniel Reyes)

Ibarra, 18 de mayo de 2020

CERTIFICADO

La microempresa Bazar y Papelería San Antonio, certifica que la Sra. PAOLA ALEXANDRA TITO CHULDE, con número de cédula 100349348-1, quien desarrollo su trabajo de titulación con el tema "DESARROLLO DE UN SISTEMA DE GESTIÓN Y CONTROL DE PROCESOS PARA LA MICROEMPRESA BAZAR Y PAPELERÍA SAN ANTONIO UTILIZANDO SPRING BOOT". Nos es grato informar que el aplicativo ha sido culminando en su totalidad y cumpliendo con los requisitos solicitados.

Por lo que se recibe el proyecto como culminado, nos comprometemos a continuar utilizando el mencionado aplicativo en beneficio de nuestra microempresa.

Es todo cuanto puedo certificar, facultando a la interesada hacer uso de este certificado como estime conveniente.

Atentamente.

Sra. Gladys Vizcaino,

Propietaria de Bazar Y Papelería San Antonio.



DEDICATORIA

El presente trabajo le dedico a mis padres Rosa y Antonio, porque gracias a su esfuerzo, apoyo y sacrificio he logrado cumplir una etapa más en mi vida.

A mis queridos hermanos Andrés y David quienes siempre han estado a mi lado en las buenas y malas brindándome todo su apoyo.

A mis demás familiares y amigos que confiaron en mí, para todos ellos dedico esta tesis por apoyarme y confiar en mí incondicionalmente.

Paola Tito.

AGRADECIMIENTOS

Agradezco a Dios por todas las bendiciones recibidas por ser la fortaleza en los momentos de debilidad y ser el inspirador para seguir adelante.

A mis padres por apoyarme en todo lo que estuvo a su alcance, por estar siempre a mi lado, apoyarme y orientarme durante toda mi vida, por motivarme en los momentos más difíciles y siempre confiar en mí.

A mi tutor Msc. Fausto Salazar y asesores quienes me brindaron su voluntad, su paciencia y sus conocimientos para que pueda finalizar con éxito este trabajo.

A todos los docentes quienes me instruyeron para que sea un profesional con altos conocimientos académicos y éticos.

Paola Tito.

ÍNDICE DE CONTENIDOS

1. IDENTIFICACIÓN DE LA OBRA	ii
2. CONSTANCIAS	iii
DEDICATORIA	vi
AGRADECIMIENTOS	vii
ÍNDICE DE CONTENIDOS	viii
ÍNDICE DE TABLAS	xi
ÍNDICE DE FIGURAS	xii
RESUMEN	xiv
ABSTRACT	xv
INTRODUCCIÓN	1
Problema	1
Antecedentes	1
Situación Actual	1
Prospectiva	2
Planteamiento del Problema	2
Objetivos	3
Objetivo General	3
Objetivos Específicos	3
Alcance	4
Justificación	5
CAPÍTULO 1	7
1. Marco teórico	7
1.1 Microempresa Bazar y Papelería San Antonio	7
1.1.1 Introducción de la microempresa	7
1.1.2 Proyección del sistema de gestión y control de procesos	7
1.1.3 Procesos manuales de la microempresa	7
1.2 Spring framework	8
1.2.1 Características de Spring Framework	9
1.2.2 Módulos	9
1.2.3 Ventajas de Spring Framework	11

1.3	Spring Boot.	12
1.3.1	Características de Spring Boot.	12
1.3.2	Dependencias de Spring Boot.	13
1.3.3	Acceso a Datos con Spring Boot.....	14
1.3.4	Gestor de Proyectos Maven.	14
1.4	Framework Angular.	16
1.4.1	TypeScript.....	16
1.4.2	Servicios Web REST	17
1.4.3	Arquitectura Angular.....	17
1.4.4	Características de Angular.	18
1.5	Metodología Scrum.	19
1.5.1	Pilares fundamentales de Scrum.....	19
1.5.2	Ciclo de Vida de Scrum.....	20
1.5.3	Características de Scrum.	21
1.6	Norma ISO/IEC 25010.....	22
1.6.1	Modelo de Calidad en Uso.	22
CAPÍTULO 2.....		24
2.	Desarrollo.....	24
2.1.	Fase 1 Inicialización.	24
2.1.1	Definición de Roles de Scrum.....	24
2.1.2.	Product Backlog	24
2.2.	Fase 2. - Planificación y estimación.....	25
2.2.1.	Historias de usuarios.	25
2.2.2.	Aprobación del Sprint	31
2.2.3.	Estimación de tareas por Sprint.....	31
2.3.	Fase 3. - Proceso de desarrollo del sistema web.	36
2.3.1.	Modelo de la base de datos del sistema web.....	37
2.3.2.	Prototipo del sistema web.....	39
CAPÍTULO 3.....		53
3.	Resultados	53

3.1 Verificación de Resultados.	53
3.1.1 Medición del Modelo de Calidad en Uso.....	53
3.1.2 Evaluación del Modelo de Calidad en Uso.....	56
3.1.3 Característica: Satisfacción	56
3.1.4 Resultado de Evaluación.	58
3.2 Análisis de Interpretación de Resultados.....	59
3.3 Análisis de Impacto.	59
3.3.1. Impacto Económico.	60
3.3.2 Impactos Ecológico	61
3.3.3. Impacto Tecnológico.....	61
CONCLUSIONES.	62
RECOMENDACIONES.	63
REFERENCIAS BIBLIOGRÁFICAS.....	64
ANEXOS.....	67

ÍNDICE DE TABLAS

Tabla 1: Definición de Roles de Scrum	24
Tabla 2: Definición del Product Backlog.....	25
Tabla 3: Historia de Usuario 1	26
Tabla 4: Historia de Usuario 2.....	26
Tabla 5: Historia de Usuario 3.....	26
Tabla 6: Historia de Usuario 4.....	27
Tabla 7 : Historia de Usuario 5.....	27
Tabla 8 : Historia de Usuario 8.....	28
Tabla 9: Historia de Usuario 7.....	28
Tabla 10: Historia de Usuario 8.....	29
Tabla 11 : Historia de Usuario 9.....	30
Tabla 12 . Historia de Usuario 10.....	30
Tabla 13. Aprobación del Sprint	31
Tabla 14: Sprint 1.....	32
Tabla 15: Sprint 2.....	32
Tabla 16: Sprint 3.....	34
Tabla 17 : Sprint 4.....	35
Tabla 18: Sprint 5.....	36
Tabla 19: Preguntas de Cuestionario.....	55
Tabla 20 : Medición de Satisfacción.....	56
Tabla 21: Resultados de la Encuesta.....	56
Tabla 22: Métrica Utilidad.....	57
Tabla 23: Métrica Confianza.	57
Tabla 24: Métrica Comodidad	58
Tabla 25: Resultados de Evaluación.....	58
Tabla 26 : Calificación de impacto.....	60
Tabla 27: Impacto Económico.....	60
Tabla 28: Impacto Ecológico.....	61

ÍNDICE DE FIGURAS

Fig. 1: Árbol del problemas	3
Fig. 2: Arquitectura.....	4
Fig. 3: Funcionalidad del Sistema de Gestión y control de Procesos	5
Fig. 4: Módulos de la Arquitectura de Spring.....	10
Fig. 5: Dependencias del Spring Boot.	13
Fig. 6: Clase JpaRepository.	14
Fig. 7: Arquitectura Framework Angular	17
Fig. 8: Pilares Fundamentales de Scrum.....	19
Fig. 9: Ciclo De Vida metodología Scrum.....	20
Fig. 10: Fases de Metodología Scrum.....	21
Fig. 11: Clasificación de calidad del producto Software.....	22
Fig. 12: Calidad de Uso Norma 25010	23
Fig. 13: Base de Datos Módulo de Ventas e inventario	37
Fig. 14 : Diagrama de Componentes.....	38
Fig. 15: Casos de Uso del Login	39
Fig. 16: Página Inicio de Sesión.....	39
Fig. 17: Caso de Usos Operador.....	40
Fig. 18: Página principal del Operador.	41
Fig. 19: Página Crear producto	41
Fig. 20: Página de Lista de Proveedores.....	42
Fig. 21: Página Crear Proveedor.....	42
Fig. 22: Página Lista de Compras.	43
Fig. 23: Crear Compra	43
Fig. 24: Página Detalle Compra	44
Fig. 25: Página Lista de Pago	44
Fig. 26: Página Detalle Pago.....	45
Fig. 27: Página de lista Pedidos.....	45
Fig. 28: Página de kardex.	46
Fig. 29: Página de Reportes.....	46
Fig. 30: Caso de Usos Empleado.....	47
Fig. 31: Página Principal de Empleado.....	47
Fig. 32: Página Stock	48
Fig. 33: Página Clientes	48

Fig. 34: Página Ventas.....	49
Fig. 35: Página Crear Factura.	49
Fig. 36: Página Detalle factura.	50
Fig. 37: Caso de Usos Administrador.....	50
Fig. 38: Página del Administrador.	51
Fig. 39. Página Administrador de Usuario.	51
Fig. 40: Bloquear Usuario	52
Fig. 41: Matriz de Calidad De Software.	53
Fig. 42: Tipo de Software.	54
Fig. 43: Característica de Calidad.	54
Fig. 44: Resultado de Evaluación de Calidad en uso	58
Fig. 45: Niveles de Puntuación.....	59
Fig. 46: Matriz de Evaluación de Calidad en Uso.	69

RESUMEN

El presente documento de Trabajo de Grado tiene como propósito realizar la creación de un sistema de gestión y control de procesos que permitirá mejorar la productividad en la micro empresa Bazar Y Papelería San Antonio.

Por lo tanto, se desarrolló una aplicación web a medida que automatice el proceso de ventas e inventario de la micro empresa, utilizando las herramientas de Framework Spring Boot para el (Back-End), Framework Angular para el (Front-End) y MySql como gestor de base de datos, mediante la utilización de la metodología SCRUM.

Para el cumplimiento del presente proyecto se estructuró de la siguiente manera:

Parte introductoria, se identificó el problema estableciendo los antecedentes, la situación actual, la prospectiva, el planteamiento del problema, se definió los objetivos, la delimitación del alcance y la justificación del proyecto propuesto.

Capítulo uno, se define la parte teórica del proyecto que consta de conceptos fundamentales, detalles de la micro empresa con sus procesos, herramientas tecnológicas y la metodología para el desarrollo del sistema como es la metodología SCRUM.

Capítulo dos, se detalla el desarrollo del módulo con la metodología de desarrollo ágil Scrum dividida en Sprints y la implementación del sistema en la micro empresa.

Capítulo tres, se realiza la validación de resultados del sistema aplicando la norma ISO/IE 25010 con la validación de característica Satisfacción y sub característica utilidad con la ayuda de la encuesta Usabilidad SUS (System Usability Scale).

Finalmente se detalla las conclusiones a las que se llegó una vez finalizado el proyecto y las recomendaciones dadas para futuras investigaciones.

ABSTRACT

The purpose of this undergraduate work document is to create a Process Management and Control System that will improve productivity in "Bazar Y Papelería San Antonio" micro company.

Therefore, a web application was developed to automate the sales and inventory process of the micro company, using Spring Boot Framework tools for Back-End, Angular Framework for Front-End, MySql as database manager and the SCRUM methodology.

To fulfill this project, it was structured as follows: Introductory part, the approach to the problem was identified by establishing the background, current situation and prospective, also objectives, scope and justification were established to the proposed project.

Chapter one, the theoretical part of the project that consists of fundamental concepts, details about the micro company with its processes, technological tools and the methodology for the development of the system such as the SCRUM methodology were defined.

Chapter two, the development of the module with the Agile Development Methodology Scrum divided into Sprints and the implementation of the system in the micro enterprise were detailed.

Chapter three, the system results validation was carried out applying the ISO / IEC 25010 standard, by taking into account "Usability" characteristic in the context of "Appropriateness recognizability" subcharacteristic, with the help of SUS (System Usability Scale) questionnaire.

Finally, the conclusions reached at the end of the project were detailed and the recommendations given for future research were elaborated.

INTRODUCCIÓN

Problema.

Antecedentes.

Bazar y Papelería San Antonio es una microempresa dedicada a la comercialización de artículos de papelería, bazar y materiales de artes plásticas; todo este tiempo se ha manejado de manera tradicional ingresando información de manera manual, lo que no permite llevar los procesos de manera adecuada ya que no están automatizados; toda esta información se encuentra guardada en hojas de cálculo.

El desarrollo de aplicaciones web evoluciona cada día con el avance de nuevas tecnologías, frameworks¹ y herramientas para el desarrollo de software²; permitiendo implementar funcionalidades que son de uso común en las aplicaciones y procesos que ya han sido desarrollados y probados(Sanz & Silva, 2014).

A nivel empresarial se realiza desarrollo de software para suplir las necesidades de automatización de procesos, este desarrollo ha seguido las tendencias tradicionales o ya conocidas por los desarrolladores, en donde las últimas décadas, las arquitecturas de software han avanzado desde los sistemas monolíticos³ abriendo la puerta a nuevas alternativas enfocada a los microservicios⁴.

Spring Boot es una alternativa viable, debido a que es un entorno ágil y más productivo, porque permite a los desarrolladores trabajar en un marco basado en microservicios; centrándose directamente en la lógica del negocio evitando las configuraciones iniciales del proyecto(Gajewski & Zabierowski, 2019).

Situación Actual.

Se ha observado que la microempresa “Bazar y Papelería San Antonio” actualmente maneja mucha información para cada proceso de ventas e inventarios, creando muchos documentos físicos siendo expuestos a pérdida de información, ya que no cuenta con un software que permita llevar un control de las tareas de cada proceso que realizan los empleados.

¹ Framework: es un entorno de trabajo que sirve de base en el desarrollo de aplicativos.

² Software: es un programa con procedimientos que permite realizar tareas en un sistema informático.

³ Monolíticos: sistemas que están escritos en un solo código o espacio lógico.

⁴ Microservicios: conjunto de pequeños servicios.

Si bien, la microempresa no utiliza tecnologías informáticas, la mayoría de negocios similares ya se ha implementado herramientas de software que faciliten el desempeño de las tareas.

Una de las tendencias en desarrollo de sistemas empresariales es el framework de Spring Boot, es por ello que surge la necesidad de explorar todas sus funcionalidades, con el fin de analizar todas las herramientas orientadas al desarrollo web.

Prospectiva.

Este aplicativo brindará apoyo informático para el registro de datos de la microempresa, además las herramientas a utilizar servirán para facilitar la gestión de los procesos con una interfaz amigable evitando confusión al usuario, igualmente el software será capaz de brindar seguridad de información mediante el uso de autenticación de usuarios.

Poseer un sistema que facilite la gestión de procesos permite optimizar el tiempo en cada proceso, además se podrá realizar un análisis en base a la información registrada en la aplicación.

Mediante el análisis que se realizará para el desarrollo de este proyecto, se generará la documentación necesaria para quienes deseen o necesiten crear sitios webs con la herramienta Spring Boot.

Con el desarrollo del sistema web se quiere demostrar que este tipo de tecnología ayuda a agilizar el desarrollo, y, sobre todo, que nos permite crear aplicaciones muy ligeras que consumen menor número de recursos.

Planteamiento del Problema.

El presente trabajo de investigación, parte de un análisis de la problemática existente en la microempresa Bazar y papelería San Antonio, donde no existe un sistema automatizado de gestión y control de procesos que permita a los empleados realizar sus funciones de manera ágil y oportuna.

Los empleados desempeñan varias actividades diarias entre ellas facturación, ingreso y despacho de mercadería, los que contienen documentación archivada en hojas de cálculo generando desorganización y pérdida de información que al final de día no cuenta con un reporte de toda la actividad provocando un caos e insatisfacción a los propietarios; en la figura 1 se muestra los efecto y causas que tiene el problema a resolver.

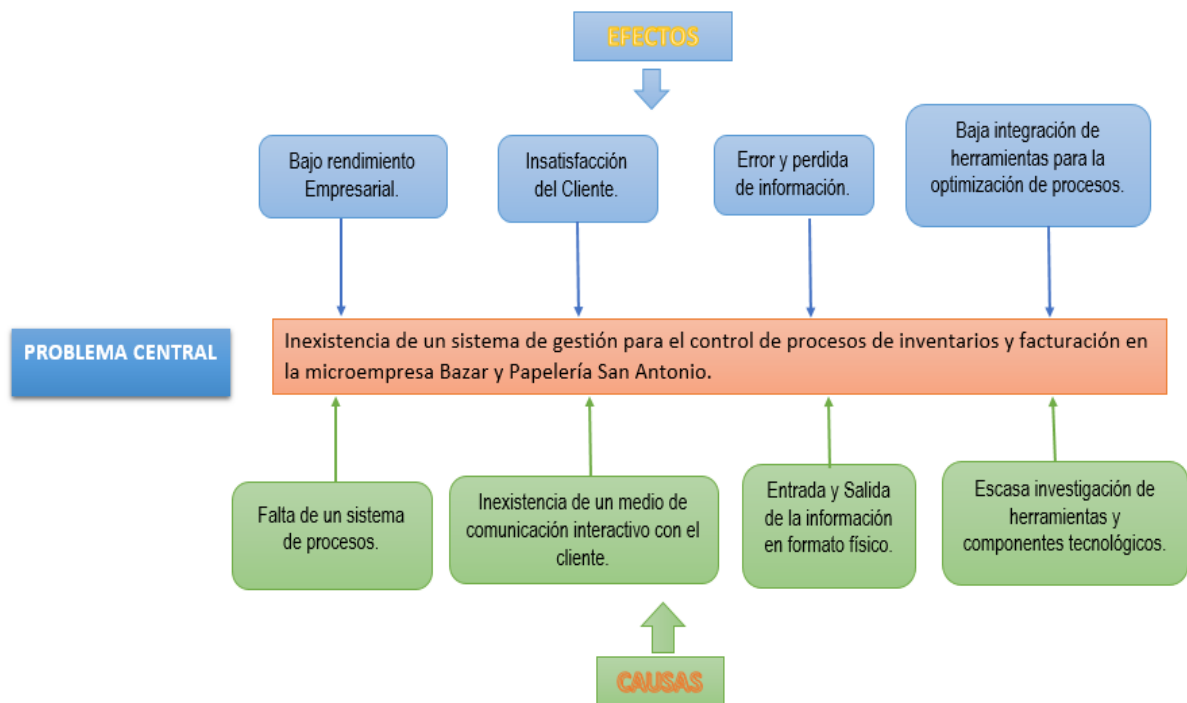


Fig. 1: Árbol del problemas.
Fuente: Propia

Por lo tanto, analizando estos aspectos se puede realizar la siguiente interrogante:
¿Qué beneficios traerá para la empresa Bazar y Papelería San Antonio adquirir un sistema de gestión y control de procesos?

Objetivos

Objetivo General

Desarrollar un sistema web que permita automatizar los procesos de gestión y control de la empresa Bazar y Papelería San Antonio utilizando Spring Boot.

Objetivos Específicos

- Realizar un marco teórico sobre el framework de desarrollo Spring Boot y las herramientas de apoyo.
- Desarrollar una aplicación web para la empresa Bazar y Papelería San Antonio utilizando la metodología SCRUM.
- Verificar los resultados mediante la evaluación de calidad de uso, con la norma ISO 25010 con la característica de satisfacción y sub característica de utilidad.

Alcance

El presente estudio tiene como finalidad analizar las características y funcionalidades del framework Spring Boot, permitiendo la automatización de los procesos de la empresa. El aplicativo se mostrarán todos los beneficios que proporciona la tecnología Spring Boot, este sistema web se realizará utilizando el IDE de desarrollo eclipse, la base de datos será MySQL, para el desarrollo del aplicativo se utilizará herramientas de software libre evitando costos de licenciamiento.

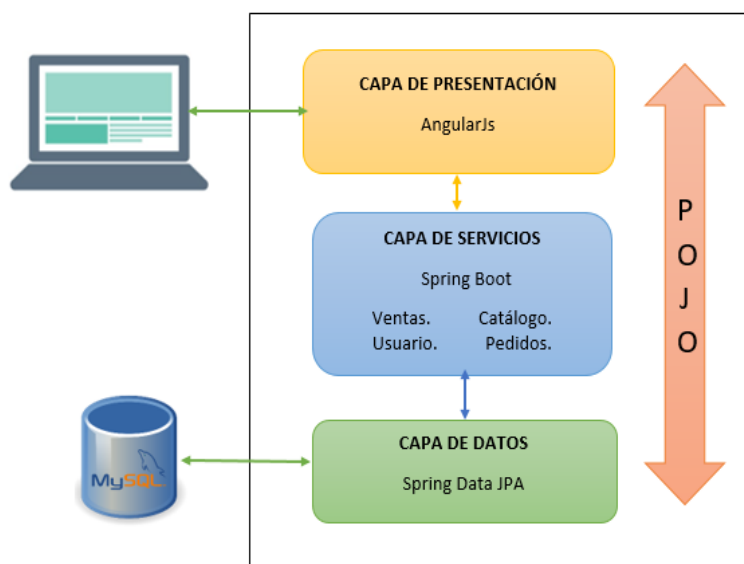


Fig. 2: Arquitectura.
Fuente: Propia.

- Capa de Presentación: interactúa con el cliente del servicio mediante las funciones de la aplicación, para esta capa se utilizará el frameworks AngularJs⁵ para comunicarse con la capa de servicio.
- Capa de servicio: se encarga de procesar y enviar datos por medio de servicios Rest⁶, en esta capa se utilizará el frameworks Spring Boot.
- Capa de datos: en esta capa se trabajará con el framework Spring Data JPA, para la persistencia de datos.
- Adicionalmente utilizaremos POJO es un tipo de objeto que transporta la información entre capas mediante contenedores de datos(Webb et al., 2018).

⁵ AngularJs: es un framework de aplicaciones web.

⁶ Servicios Rest: es una interfaz para conectar varios sistemas basados en protocolos.

El Sistema de gestión y control de proceso está dividido en 2 partes:

SISTEMA DE GESTIÓN Y CONTROL DE PROCESOS.

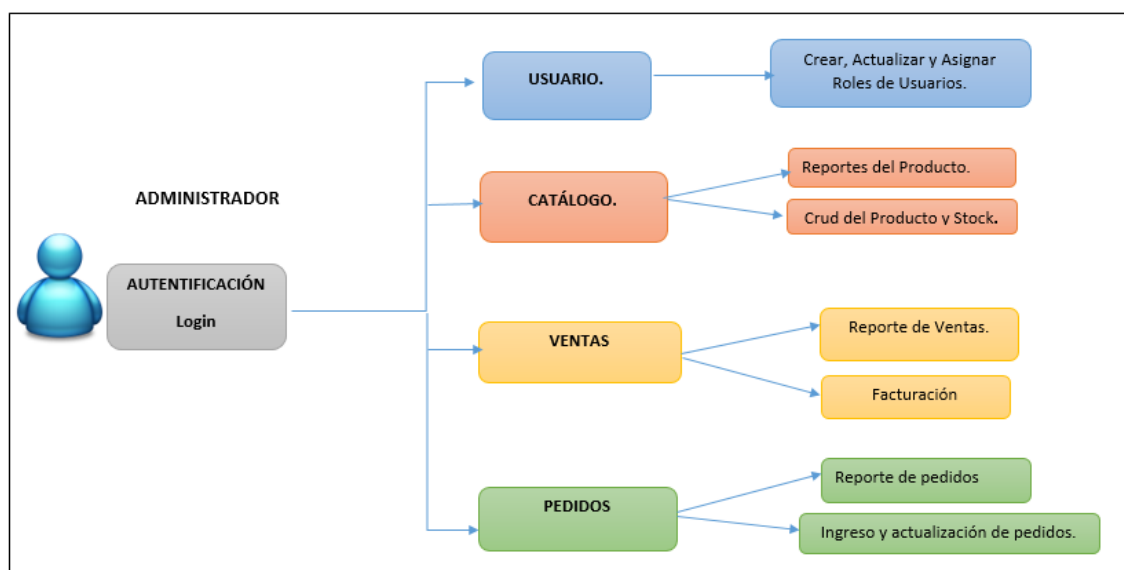


Fig. 3: Funcionalidad del Sistema de Gestión y control de Procesos
Fuente: Propia

- **Módulo de autenticación**

Este módulo tiene como funcionalidad principal el restringir el ingreso al sistema por parte de usuarios externos no registrados, también limitará las funcionalidades que tiene el sistema según el rol asignado previamente al usuario.

- **Módulo de Administración.**

Este módulo será el encargado de administrar y procesar toda la información y archivos almacenados por los usuarios.

- Usuario
- Catalogo
- Ventas
- Pedidos.

Justificación.

La implementación de este proyecto se justifica al no existir aplicaciones o sistemas de procesos automatizados en Bazar y Papelería San Antonio, tomando en cuenta que la

microempresa cuenta con computadores y tecnología que permita la implementación del sistema.

El presente proyecto tiene un enfoque a los siguientes Objetivos de Desarrollo Sostenible planteados por la ONU y UNESCO (Naciones Unidas, 2018):

9.b Apoyar el desarrollo de tecnologías, la investigación y la innovación nacionales en los países en desarrollo, incluso garantizando un entorno normativo propicio a la diversificación industrial y la adición de valor a los productos básicos, entre otras cosas.

9.c Aumentar significativamente el acceso a la tecnología de la información y las comunicaciones y esforzarse por proporcionar acceso universal y asequible a Internet en los países menos adelantados de aquí a 2020(Naciones Unidas/CEPAL, 2019).

Justificación Social

Este proyecto será de gran utilidad para el grupo de trabajo de bazar y Papelería San Antonio ya que contarán con un sistema de procesos y gestión automatizados, además la ayudará a mantener una mejor organización de las actividades en las que están involucrados.

La documentación generada en el desarrollo de este proyecto de investigación será de ayuda para los estudiantes que deseen desarrollar una aplicación utilizando las herramientas de Spring Boot.

Justificación Económica

El análisis de datos y registro de información son muy valiosos para la microempresa por lo que la implementación del Sistema de gestión y control de procesos es una herramienta bastante útil en agilizar los procesos y disminuir las tareas del empleado y propietarios del negocio, siendo así los beneficiarios directos, y como beneficiarios indirectos el cliente al realizar compras más rápidas sin pérdida de tiempo.

Trabajar con herramientas de software libre y gratuito ayuda a evitar gastos de licenciamiento, Spring Boot evita pérdida de tiempo en desarrollo por la reutilización de código generando reducción de costos en el Cliente.

Justificación Tecnológico

Es importante utilizar las herramientas tecnológicas que están en constante evolución para desarrollar sistemas óptimos y actualizar los conocimientos para optimizar el desarrollo de aplicaciones y sistemas web.

CAPÍTULO 1

1. Marco teórico

1.1 Microempresa Bazar y Papelería San Antonio.

1.1.1 Introducción de la microempresa.

Bazar y Papelería San Antonio es una microempresa que se encuentra en el mercado desde el año 1995, está dedicada a la comercialización de artículos de bazar, papelería, librería, material de pintura y artes plásticas; con los años fue creciendo y adquiriendo más mercadería, obteniendo una mayor variedad de productos, ampliando el mercado a ventas de libros y aparatos tecnológicos.

Se encuentra ubicada en la provincia de Imbabura, cantón Ibarra, parroquia de San Antonio, en las calles Camilo Pompeyo y Sucre; ha venido trabajando en sus procesos de modo tradicional, no existe ningún sistema informático que ayude en la automatización procesos.

1.1.2 Proyección del sistema de gestión y control de procesos.

El sistema permitirá registrar todos los documentos correspondientes a inventarios, pedido y ventas, de esta forma se pretende optimizar los procesos y se facilite la gestión y seguimiento de cada uno de ellos, los beneficios que se obtendrá:

- Registro de la mercadería.
- Información de existencia de productos actualizado.
- Ventas.
- Seguimiento del estado del pedido.
- Documentos legibles.

1.1.3 Procesos manuales de la microempresa.

Atención al cliente: Se detalla toda la información correspondiente a los productos, dando a conocer las características y precios.

Salida de productos: Para este proceso es necesario tener un listado actualizado de productos, esto permitirá a los vendedores ver las existencias y ofrecer productos a los clientes.

Facturación: Una vez seleccionado el producto se llena el detalle de la factura, y se realiza el cobro, el proceso se lo realiza de forma manual. Una vez realizada una venta se procede a guardar la información en un papel de anotaciones.

Fin de la jornada: Se verifican las ventas realizadas en el día y se procede a cuadre de caja.

Stock de productos: En esta sección se encuentra todo el inventario de productos, detallado con entradas y salidas.

1.2 Spring framework.

Spring es un marco de trabajo utilizado para el desarrollo de aplicaciones de código abierto, que brinda un modelo de programación orientada a objetos para la plataforma Java⁷. Este framework de programación tiene por finalidad facilitar la construcción de software empresarial, impulsado en una metodología de trabajo ágil y eficiente, permitiendo el desarrollo de aplicaciones ligeras (César, 2018).

Está configurado mediante los archivos XML⁸ y peticiones HTTP⁹ mediante un patrón de diseño que encapsula algoritmos¹⁰ relacionados en subclases, con la finalidad de re utilizar los algoritmos(Campoverde, 2019).

Uno de los aspectos principales de Spring es que se encuentra basado en la técnica de Inversión de Control y la inyección de dependencias.

Inversión de Control (IoC): es un método de programación donde el flujo de ejecución se invierte, utilizado para gestionar las relaciones del objeto, es decir, proporciona un contenedor que maneja todo lo que se hace con los objetos permitiéndole resolver dependencias entre los servicios que el contenedor controla(Gutiérrez, Verduzco, & Farías, 2015).

Inyección de dependencia (DI): es un patrón de diseño de software que funciona con un objeto que se encarga de construir las dependencias que una clase necesite, permitiendo introducir comportamientos a componentes, siendo así el mecanismo para implementar IoC(Crusoveanu, 2019).

⁷ Java: lenguaje de programación orientada a objetos.

⁸ Lenguaje de Marcado Extensible (XML): que es un lenguaje de etiquetas que define un conjunto de reglas para la codificación de documentos.

⁹ Protocolo de Transferencia de Hiper Textos (Http): es la transferencia de información entre diferentes servicios y los clientes que utilizan páginas web.

¹⁰ Algoritmos: es un conjunto de instrucciones o reglas que permite, solucionar un problema o procesar datos y llevar a cabo otras tareas.

1.2.1 Características de Spring Framework.

A continuación, se describirán algunas características de Spring (Pérez, 2015) :

- Es un frameworks liviano y ligero debido a su implementación POJO¹¹.
- IoC configura los componentes de la aplicación mediante la inyección de dependencias.
- Utiliza inyección de dependencia para separar el código de negocio.
- Trabaja con base de datos relacionales.
- Utiliza estándares, protocolos, herramientas y prácticas para la configuración de seguridad y acceso.
- Ayuda a generar rápidamente aplicaciones.
- Maneja Programación Orientada a Aspectos (AOP) para organizar el código en los problemas y aspectos del negocio.
- Está orientado a peticiones hacia los controladores con una funcionalidad especial que permite el desarrollo de proyectos web.
- Esta organizado de manera modular.
- Spring no obliga al programador a heredar ninguna clase ni a implementar ninguna interfaz.
- Tiene amplia compatibilidad para la integración con otros frameworks y librerías de uso común.

1.2.2 Módulos.

EL framework Spring está compuesto por una serie de módulos que brindan a los programadores los medios y servicios necesarios para desarrollar aplicaciones empresariales de acuerdo a sus necesidades. Es importante establecer que no es necesario trabajar con todos los módulos, el programador tiene la libertad de utilizar las herramientas que sea necesario (Gajewski & Zabierowski, 2019).

En la Figura 6 se observar los módulos que componen el marco de trabajo Spring obtenido en la página web oficial del framework:

¹¹ Plain Old Java Object (POJO) Objeto Java Plano Antiguo: es una instancia de una clase que no extiende ni implementa nada en especial.

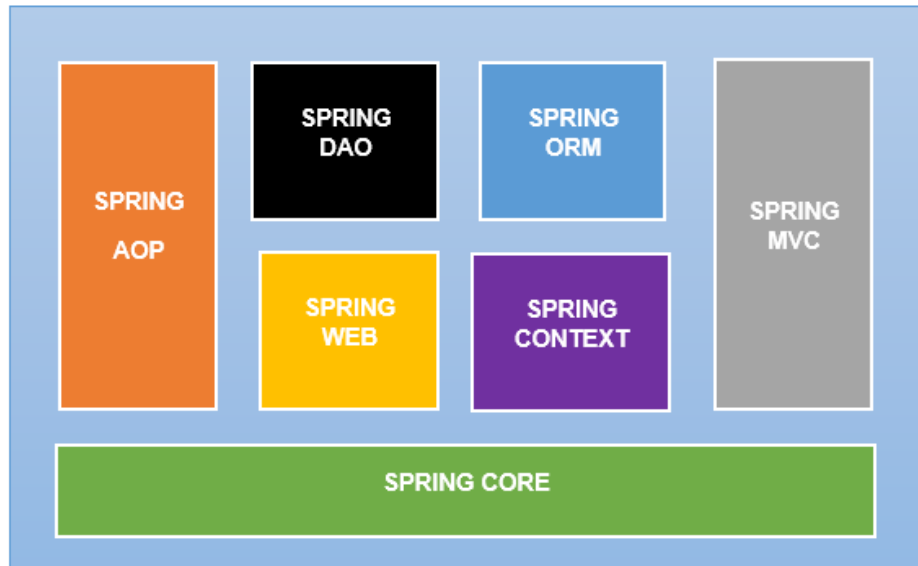


Fig. 4: Módulos de la Arquitectura de Spring.
Fuente: Adaptado de Spring.org

Spring Core: es contenedor fundamental de Spring se caracteriza por crear muchos tipos de Bean que se encarga de manejar la relación entre ellos; el cual utiliza el patrón IoC y configura los objetos a través de Inyección de dependencia(Esquivel, 2016).

Spring Context: es donde se realiza la configuración del framework por medio de un archivo XML, proporciona métodos de acceso a los objetos, además, permite internacionalización, propagación de eventos, recarga de recurso y la creación transparente de contextos(Ibáñez & Lévano, 2016).

Spring Programación Orientada a Objetos (AOP): es un módulo que provee soporte de la programación orientada a objetos de una manera muy similar a la inyección de dependencias, el módulo permite el desacoplamiento de los objetos java se puede utilizar para:

- Persistencia
- Manejo de transacciones
- Seguridad
- Logging (Esquivel, 2016).

Spring Mapeo Objeto Relacional ORM: en este módulo de integración con frameworks permitiendo manejo de sesiones, recursos, transacciones y excepciones; a su

vez da soporte a los frameworks ORM más completo entre ellos Hibérnate, JPA y Apache OBJ(Ibáñez & Lévano, 2016).

Spring Objeto de Acceso a Datos (DAO): permite extraer el código de acceso a datos de una manera simple y limpia, facilita la implementación de los objetos a la capa de acceso a datos. Ayuda a simplificar el código, reduciendo el código repetitivo y cuenta con manejo de transacciones a través de AOP (Esquivel, 2016).

Spring Web: módulo que aporta clases especiales orientadas al desarrollo web como ficheros, inicialización del contenedor IoC y contenido de la aplicación web(Ibáñez & Lévano, 2016).

Spring Modelo Vista Controlador (MVC): una implementación del conocido por el patrón de diseño para el desarrollo de aplicaciones web mediante la creación de API¹² y Servlets¹³ (Mak & Mak, 2018).

1.2.3 Ventajas de Spring Framework

A continuación se describirán algunas características de Spring (César, 2018):

- Soporte para la gestión de transacciones.
- Spring promueve la reutilización de código.
- Soporte para interacción con diferentes bases de datos.
- Está diseñado para que las aplicaciones que son construidas con Spring sean fáciles de testear.
- Soporte para Inyección de Dependencia.
- Lógica aplicable con POJO.
- Su código fuente es más simplificado y menor dificultad en los ajustes.
- Soporte para el estilo de servicios web RES.

¹² API: es la interfaz de programación de aplicaciones.

¹³ Servlets: son módulos java que nos sirven para extender las capacidades de los servidores web.

1.3 Spring Boot.

Spring es un marco de trabajo para aplicación web, que se ha popularizado entre los programadores, con el tiempo los proyectos basados en él fueron generando inconvenientes por un gran número de archivos XML que tenían que estar correctamente configurados para que los componentes individuales y la aplicación funcionarían correctamente (Webb et al., 2018).

Esto ha causado muchas frustraciones entre los programadores al invertir tiempo que puede ser valioso en la lógica de negocios. Por lo tanto, se creó Spring Boot un módulo desarrollado a partir de Spring, que nace con la finalidad de simplificar aún más la programación de aplicaciones basadas en el framework, ayudando al programador solo centrarse en el desarrollo de la solución del problema, olvidándose por completo la compleja configuración inicial que actualmente tiene Spring (Gajewski & Zabierowski, 2019).

Spring Boot simplifica enormemente tareas convencionales como son el despliegue del servidor, el cual ya se encuentra embebido dentro del proyecto que por defecto es un Tomcat permitiendo al proyecto se ejecute en un Jar. Las configuraciones se realizan automáticamente como el gestor de base de datos, el caso que trabaje con varias bases de datos diferentes lo único que debe hacer el programador es proporcionar una dirección IP; también permite usar las herramientas de gestor de proyectos como Maven y Gradle mediante la administración de dependencias (Suryotrisongko, Jayanto, & Tjahyanto, 2017).

1.3.1 Características de Spring Boot.

Spring Boot centra su éxito en las siguientes características que lo hacen fácil de utilizar (Martín, 2018):

- Cuenta con un completo módulo de auto configuración y validación en todos los aspectos de la aplicación.
- Trabaja en base a una gestión de dependencias, es decir, solo determina qué tipo de proyecto utilizara y él se encarga de resolver todas las librerías y dependencias que necesita sin necesidad de especificar su versión para que la aplicación funcione.

- Permite ejecutar aplicaciones Stand-alone¹⁴ y web, ya que permite desplegar las aplicaciones mediante un servidor web integrado.
- Cuenta con la incorporación directa de servidores web como Apache Tomcat, eliminando la necesidad de incluir archivos WAR¹⁵.
- Tiene métricas de servicios que permite consultar el estado de la aplicación, la disponibilidad y detalle de los Beans¹⁶.
- Es un frameworks extensible ya que permite la creación de complementos, los cuales facilitan aún más el desarrollo de las aplicaciones (Martín, 2018).

1.3.2 Dependencias de Spring Boot.

Spring Boot trabaja mediante dependencias facilitando el trabajo al desarrollador las cuales se pueden incluir manualmente, pero esto resultaría un tanto innecesaria ya que el framework permite iniciar con dos dependencias esenciales en proyectos serían starter y las dependencias de testing.” (Pérez Martínez & Altadill Izura, 2018).

```

<dependency>
  <groupId>org.springframework.boot</groupId>
  <artifactId>spring-boot-starter</artifactId>
</dependency>
<dependency>
  <groupId>org.springframework.boot</groupId>
  <artifactId>spring-boot-starter-test</artifactId>
  <scope>test</scope>
</dependency>

```

Fig. 5: Dependencias del Spring Boot.
Fuente. Adaptación del Archivo pom.xml

Dependencia Start: permite al proyecto hacer uso de algunas funcionalidades de Spring Boot que están embebidas. Estos grupos de funcionalidades son denominados Starters.(Bruno, 2019)

Dependencia Test: disponemos de una serie de clases y anotaciones de utilidad que nos servirán para desarrollar tests de integración de forma sencilla verificando el código si realmente funciona correctamente.(Ottinger & Lombardi, 2019).

¹⁴ Stand-alone: es a una aplicación que puede ser utilizada en cualquier computadora sin necesidad de ser instalada.

¹⁵ Archivo de Aplicación Web (WAR): es un archivo que contiene la estructura de directorios y ficheros de una aplicación web.

¹⁶ Beans: es un objeto que se instancia y administrado por un contenedor Spring IoC.

1.3.3 Acceso a Datos con Spring Boot

Una de las características de las aplicaciones empresariales es que pueden gestionar los datos de la empresa almacenados en un SGBD (Software de Gestión de Base de Datos). Generalmente esto se puede lograr cuando el código tiene acceso a un origen de datos, enviando consultas y procesando los resultados.

“Spring facilita enormemente el acceso a datos adoptando las técnicas de los frameworks ORM como Hibérnate, donde los datos de un SGBD se traducen a colecciones de objetos y viceversa(Pérez Martínez & Altadill Izura, 2018).

Spring Boot ofrece un fantástico soporte para acceder a los datos con JPA a través de sus interfaces, del tipo Repository, esto se provee el paquete Spring Data Rest con el cual con apenas código podremos crear una API para acceder a las tablas de nuestra base de datos(Pérez Martínez & Altadill Izura, 2018).

```
import org.springframework.data.jpa.repository.JpaRepository;

public interface IClienteDao extends JpaRepository<Cliente, Long>{
    public Cliente findByCedula(String cedula);
}
```

Fig. 6: Clase JpaRepository.
Fuente: Adaptación clase Spring Boot

1.3.4 Gestor de Proyectos Maven.

Maven es un software libre de Apache que se utiliza para administrar las dependencias de los proyectos, con la integración de plugins o complementos mediante el archivo POM¹⁷, en dicho archivo se definen las instrucciones para compilar el proyecto, con librerías, compilación y empaquetado del código dentro de la estructura del JAR¹⁸, además, se determina una lista de dependencias las cuales se actualizan automáticamente dentro del proyecto (Gómez, 2016).

¹⁷ POM: que es archivo principal de un proyecto Maven. Contiene información acerca del proyecto, fuentes, test, dependencias, plugins y versión.

¹⁸JAR: es un tipo de archivo que permite ejecutar aplicaciones y herramientas escritas en el JAVA.

Los proyectos desarrollados en Maven se identifican en un archivo XML, puede ejecutarse nativamente en el computador o también puede integrarse en la mayoría de IDEs¹⁹, y tiene la capacidad de trabajar en red (Campoverde, 2019).

1.3.4.1 Características de Maven.

Entre las características más importantes que tiene Maven son (Flex, 2018):

- Es posible trabajar en Maven sin necesidad de estar conectado a internet solo con realizar una pequeña configuración en el fichero POM.
- Construye nuestro código fuente del proyecto.
- Maven se basa en patrones y en estándares, esto permite a los desarrolladores moverse entre proyectos y no necesitan saber cómo compilar o empaquetar, esto mejora el mantenimiento y la reusabilidad.
- Permite descargar plantillas de proyectos para tener la estructura ya creada.
- Crea una web con la documentación del proyecto.
- Maven realiza la gestión de librerías, incluso teniendo en cuenta las dependencias transitivas.
- Trabaja con un modelo comprensible para la creación de proyectos y el mantenimiento de ellos.
- Nos proporciona un acceso inmediato a nuevas funcionalidades requiriendo un esfuerzo muy pequeño de configuración.
- Maven permite una sencilla gestión simultánea de varios proyectos.

1.3.4.2 Ventajas de Maven.

Entre las ventajas más importantes que tiene Maven son (Campoverde, 2019):

- Estandarización en el sistema de compilación.
- Fácil configuración para construir artefactos.
- Gestionamiento de dependencias, esta es una gran ventaja debido a que Maven gestionará las dependencias del programador.
- Facilitación de pruebas como parte del ciclo de vida del proyecto.

¹⁹IDEs Entorno de Desarrollo Integrado, es un programa informático que brinda servicios completos al desarrollador de software.

- Disponibilidad de complementos, Maven dispone de una variedad de complementos y plugins para ejecutar tareas diversas, solo es necesario configurar el archivo POM o en su defecto descargar dichos plugins de acuerdo al requerimiento del programador.
- Organización y estandarización en la estructura de ficheros del proyecto.
- Facilita el modularidad y el versionamiento de los proyectos lo que permite trabajar con varios grupos de programadores en un mismo desarrollo.
- Dispone de un enorme repositorio de librerías Open Source en constante actualización, de forma que los desarrolladores pueden acceder a las versiones más actualizadas de las mismas.

1.4 Framework Angular.

Las aplicaciones de hoy requieren software completo debido a las demandas de usabilidad, adaptabilidad y dinamismo. Esto es difícil de lograr usando la interfaz de usuario convencional basado en componentes. Es por ello a lo largo de los años van saliendo nuevas herramientas de desarrollo con mayores funcionalidades entre ellas esta Angular (Gudiño, 2018).

Angular es un framework de código abierto, creado por Google en el año 2009, utilizado principalmente para desarrollar aplicaciones web del lado del cliente, permitiendo simplificar la creación de aplicaciones cliente en HTML y en un lenguaje como TypeScript (Valencia, 2018).

1.4.1 TypeScript

Es un lenguaje de programación de código abierto creado por Microsoft, trabaja como un compilador encargado de traducir instrucciones de un lenguaje a otro, ofrece soporte para las diferentes funciones presentadas por JavaScript para el desarrollo de componentes sólidos y entrega de desarrollo de aplicaciones de óptimas condiciones y de alta confianza. (Valverde Ramos & Mora de Fuentes, 2014)

TypeScript una vez compilado de forma limpia y simple JavaScript puede ser ejecutado desde cualquier navegador web (Microsoft, 2020).

1.4.2 Servicios Web REST

Transferencia de estado representacional, presenta una arquitectura basada en el protocolo Http en la que está presente el cliente/servidor, todos los recursos se encuentran de forma única por medio de URLs.

La arquitectura REST presenta cuatro tipos de operaciones para compartir y manipular recursos(Cardador, 2014).

- Read (Leer): El cliente envía una petición al servidor para leer una determinada información de un recurso.
- Create (Crear): El cliente envía una petición al servidor para la creación de un nuevo recurso en el sistema.
- Delete (Eliminar): El cliente envía una petición al servidor, esta acción permite eliminar un recurso del sistema o del servidor.
- Update (Actualizar): El cliente puede realizar una petición al servidor para modificar un recurso en el sistema.

Para realizar cualquier tipo de operaciones necesaria se ejecutarán por el protocolo HTTP que permitirá operaciones como (Cardador, 2014):

- GET (leer y consultar),
- PUT (editar),
- POST (crear),
- DELETE (eliminar)

1.4.3 Arquitectura Angular.

En la figura 7, se puede observar gráficamente la Arquitectura de Angular:

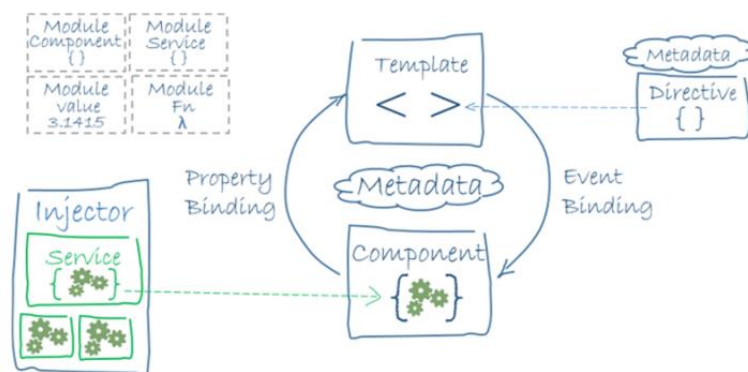


Fig. 7: Arquitectura Framework Angular
Fuente: Powered by Google, 2017.

Módulos: es un contenedor diseñado para diferentes partes de una aplicación, permite una agrupación lógica del código declarado por componentes, servicios, filtros, directivas, etc. (Figueroa, 2018).

Componentes: es un controlador de código re utilizable que va ligado a una vista. Está estructurado de 3 archivos CSS, HTML y TypeScript; y cada componente define una clase que contiene data y lógica, y está vinculada al HTML²⁰ (Proaño, 2019).

Data binding: es un enlace de datos, que permite la sincronización automática entre la vista y el modelo, todo cambio que realice en el modelo se actualiza en la vista y viceversa (MAFLA, 2019).

Directivas: son instrucciones para manejar el DOM (Modelo de Objetos del Documento) están formadas por sintaxis del HTML. Esta directiva se crea a nivel de un atributo o clase, se encuentran anteceditas por un asterisco. Ejemplo : *ngIf (Proaño, 2019).

Servicios e Inyección de Dependencias: permite manejar las clases de los componentes de forma ligera y eficiente mediante mecanismos que proporcionan nuevas instancias de una clase con las dependencias (Proaño, 2019).

1.4.4 Características de Angular.

Angular es una plataforma de desarrollo de software que utiliza estándares web. Incluye una gran cantidad de características esenciales entre ellas están: (Figueroa, 2018)

- Framework para construir SPA²¹.
- Genera directivas o componentes para la creación de aplicaciones web.
- Los componentes de angular permiten re utilizar su código en otro proyecto de angular.
- El desarrollador crea vistas con HTML Templates mediante atributos propios de Angular.
- Permite crear animaciones complejas y de alto rendimiento, desarrollando aplicaciones ágiles.
- El desarrollador escribe Componentes con clases TypeScript para gestionar el contenido de la vista.

²⁰ Template HTML: es una mezcla de HTML con Angular tags personalizados, permite definir la vista de un componente.

²¹ Aplicación de Página Única (SPA): traslada gran parte de la lógica de presentación y negocios al navegador

- La lógica de la aplicación a nivel cliente se escribe con clases denominadas Services.
- Todos los componentes, y Services se crean dentro de módulos.

1.5 Metodología Scrum.

Existen numerosas metodologías de desarrollo que están enfocadas para mejorar el proceso de desarrollo de sistemas. Cada una ha proporcionado mejoras significativas en la productividad, entre ellas esta Scrum, un modelo para trabajar en equipo a partir de Sprints, centrado en la obtención de resultados y responder a las exigencias de los requerimientos. (MAFLA, 2019)

El objetivo principal de Scrum es trabajar con el programador y con las nuevas prácticas de desarrollo, mediante el fundamento de la teoría de control empírico de procesos, esto asegura que el conocimiento procede de la experiencia y poder tomar decisiones basándose en lo conocido (Navarro, Fernández, & Morales, 2013).

1.5.1 Pilares fundamentales de Scrum.

La metodología de desarrollo se base en 3 fundamentos como se observa el fig.:

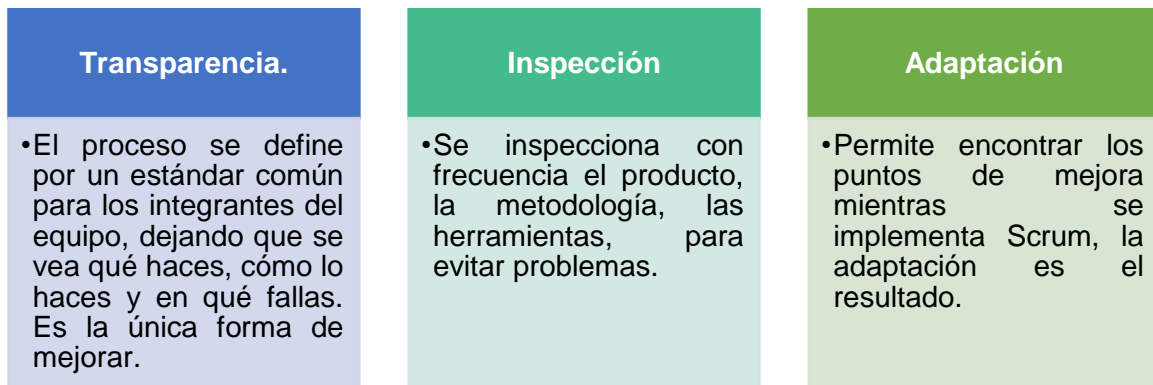


Fig. 8: Pilares Fundamentales de Scrum
Fuente: (Schwaber, Ken; Sutherland, 2013)

Roles de Scrum.

La metodología Scrum tiene roles y responsabilidades, asignados a sus procesos de desarrollo entre ellos están (Navarro, 2013):

Customer: el cliente participa en las tareas que involucran la lista Product Backlog.

Management: tiene la responsabilidad de controlar el progreso del desarrollo del aplicativo y participa en la selección de objetivos y requerimientos.

Product Owner: es el encargado de informar la visión del proyecto al equipo, él tiene autoridad sobre los tres roles, también es el representante del negocio.

Scrum Master: persona que lidera el equipo, este papel hace que cumpla las actividades de acuerdo al cronograma, a la vez facilita las reuniones como líder.

Scrum Team: es el equipo del proyecto que tiene la autoridad para decidir cómo organizarse para cumplir con los objetivos. Se encuentra organizado entre 5 y 10 personas cada una especializado, por ejemplo: Programadores, Diseñadores de Interfaz de usuario, etc.

1.5.2 Ciclo de Vida de Scrum.

Scrum cuenta con varias etapas de desarrollo como se muestra en la figura 9.



Fig. 9: Ciclo De Vida metodología Scrum
Fuente: (Equipo de desarrollo de SOFTENG, 2013)

Backlog del Producto: lista de características de los requisitos o expectativas del cliente que debe estar compuesto únicamente por funciones; cosas que el cliente pueda entender. Es gestionado por el Product Owner.(Proaño, 2019)

Sprint: Es el conjunto de interacciones del producto que se planifica mediante el tiempo de un mes o menos. Permitiendo inspeccionar y adaptar todos los otros eventos de Scrum. Un Sprint se cancelaría si el objetivo del Sprint llega a quedar obsoleto y el solo el Product Owner tiene la autoridad para ello.(Proaño, 2019)

Backlog del Sprint: proporciona una visión del trabajo a realizar durante el Sprint actual, ayuda a visualizar todo el trabajo pendiente durante un Sprint. Pertenece únicamente al Equipo de Desarrollo (Navarro et al., 2013).

Incremento de Producto: Es la suma de todos los elementos de la Product Backlog y el valor de los incrementos de todos los Sprints anteriores. Debe estar en condiciones de utilizarse sin importar si el Product Owner decide liberarlo o no (Navarro et al., 2013).

Fases de Scrum

Scrum se estructura en tres fases denominadas Pre-Game, Game y Post-Game:

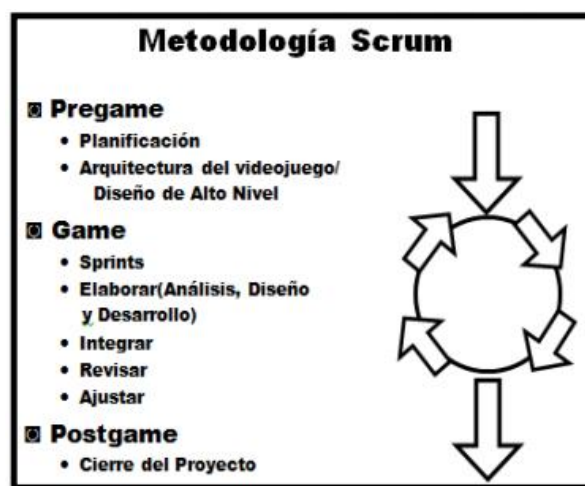


Fig. 10: Fases de Metodología Scrum
Fuente: (Equipo de desarrollo de SOFTENG, 2013)

Pre-Game: En esta fase se define el producto basado en las características conocidas, estimando su tiempo y costo. También se analiza el sistema a construir, se define la arquitectura y se realiza un diseño de alto nivel de la solución.

Game: En esta etapa, se tiene ya el alcance del proyecto bien definido, y por lo tanto una buena idea de lo que realmente es el juego y lo que se debe hacer.

1.5.3 Características de Scrum.

Flexibilidad a cambios: permite realizar cambios de requerimientos durante proceso si el cliente lo ve necesario.(Schwaber, Ken; Sutherland, 2013)

Mayor calidad del software: al trabajar con Sprints al final se obtendrá una versión funcional, que contribuye a entregar un software de calidad.(Schwaber, Ken; Sutherland, 2013)

Transparencia: Consiste en mantener informados a todo el personal de lo que ocurre con el proyecto para establecer una mejor comprensión.

Adaptación: Si los requisitos cambian, el equipo de trabajo debe ajustarse para llevar acabo la ejecución de sprint(Schwaber, Ken; Sutherland, 2013) .

1.6 Norma ISO/IEC 25010.

La norma proporciona el modelo de calidad a seguir para evaluar un producto software contribuyendo a la calidad. En este modelo se determinan las características de calidad que se van a tener en cuenta a la hora de evaluar las propiedades de un componente software determinado.(ISO/IEC-25010, 2019)

Los requisitos son funcionalidad, rendimiento, seguridad, mantenibilidad, etc. los que se encuentran clasificados por las características y subcaracterísticas de calidad como se observa en la figura 11.



Fig. 11: Clasificación de calidad del producto Software
Fuente: (ISO/IEC-25010, 2015)

1.6.1 Modelo de Calidad en Uso.

El modelo de calidad se clasifica en cinco características que tienen que ver con los resultados de la interacción del usuario con el sistema, está determinado por la calidad del software, hardware, diseño de interfaz y características de usuarios y ambiente social. Todos los factores contribuyen a la calidad en uso del sistema. Las características son las que se observa en la figura 12.



Fig. 12: Calidad de Uso Norma 25010
Fuente: ISO/IEC 25010, 2011, pág. 10.

A continuación, se define a cada una de las características y sub características de acuerdo al modelo de calidad de uso la ISO/IEC 25010, tomadas de (ISO/IEC 25010, 2011, págs. 8-21).

- Efectividad: capacidad del sistema software para alcanzar los objetivos o necesidades del usuario, al momento de utilizar el sistema.
- Eficiencia: capacidad del sistema software para alcanzar los objetivos del usuario, utilizando los recursos mínimos.
- Satisfacción: capacidad del sistema software para satisfacer las diferentes necesidades mínimas de los usuarios al utilizarlo. Contempla las siguientes subcaracterísticas:
 - Utilidad: grado en que un usuario es satisfecho cuando logra alcanzar sus objetivos planteados.

CAPÍTULO 2

2. Desarrollo.

Para el desarrollo del software se ha optado por la implementación de la metodología SCRUM, para esto es necesario la estructuración correcta del proyecto, así como la información recabada del proyecto para su respectiva construcción.

Para la realización de los módulos en la micro empresa Bazar y Papelería San Antonio, se desarrollará un sistema informático, para la gestión de los procesos, el aplicativo se integrará con los módulos, ventas, pedidos, compras, kardex los mismos módulos que estarán integrados a sus respectivos reportes.

A continuación, se especifica los requisitos necesarios para el desarrollo del sistema web:

2.1. Fase 1 Inicialización.

2.1.1 Definición de Roles de Scrum

Para el desarrollo de la aplicación se debe asignar roles al equipo de trabajo para un correcto desempeño. En la Tabla 1, se detalla el equipo de trabajo para la metodología SCRUM:

Tabla 1: Definición de Roles de Scrum

Rol	Descripción	Persona
Product Owner	Propietaria de Bazar y Papelería San Antonio.	Gladys Vizcaíno
Scrum master	Director de trabajo de grado.	Msc. Fausto Salazar
Development Team	Tesista	Paola tito

Fuente: Propia

2.1.2. Product Backlog

En este punto se focaliza en listar los requisitos prioritarios para el desarrollo del sistema.

- **Definición del Product Backlog.**

A continuación, se refleja todo lo que el equipo de desarrollo debe realizar en la construcción del software. Ver Tabla 2.

Tabla 2: Definición del Product Backlog

ID	Historia de usuarios	Prioridad	Dificultad	Descripción
1	Levantamiento de requerimiento.	Alta	8	Se establecerá las funciones del sistema.
2	Base de datos.	Alta	8	La base de datos debe ser diseñada para almacenar los datos necesarios para el sistema.
3	Herramientas de Desarrollo.	Alta	5	El software para el desarrollo debe ser gratuito.
4	Gestión del Módulo de productos.	Alta	6	El sistema debe permitir registrar, modificar y gestionar stock de productos.
5	Gestión del Módulo de Clientes y Proveedores.	Medio	5	El sistema debe permitir registrar, modificar y gestionar las facturas, compras asignados al cliente y proveedor.
6	Gestión del Módulo de Ventas.	Alta	6	El sistema debe permitir ingresar facturas y realizar reporte de ventas.
7	Gestión del Módulo de Compras.	Alta	6	El sistema debe permitir ingresar, anular y generar reportes de compras.
8	Gestión del Módulo de Pedidos y Stock	Alta	5	El sistema permitirá administrar los pedidos mediante el stock de producto.
9	Kardex	Medio	4	Se gestionará entrada y salida de la mercadería.
10	Gestión del Módulo de Seguridad	Alta	10	El sistema debe tener control de perfiles de acceso.

Fuente: Propia

2.2. Fase 2. - Planificación y estimación.

2.2.1. Historias de usuarios.

Las historias de usuario se obtuvieron luego de varias reuniones realizadas con la propietaria Gladys Vizcaíno. y Paola Tito (Tesista) con el fin de realizar el levantamiento de requerimientos.

Tabla 3: Historia de Usuario 1

HISTORIA DE USUARIO		
Número: 1	Usuario: Empleado/Operado/Administrador	
Nombre historia: Levantamiento de requerimientos.		
Prioridad: Alta	Riesgo: Alto	Estimación: 8 horas
Descripción:		
<ul style="list-style-type: none"> • Se obtiene los requisitos con una reunión con la dueña de la micro empresa en la que expone todas sus necesidades. • El sistema debe tener una interfaz fácil de usar. • El sistema web debe ser desarrollado con herramientas sin costo. 		
Pruebas de aceptación:		
<ul style="list-style-type: none"> • Levantamiento de requisitos quedan registrados por los presentes, y se estable tiempos de entrega. 		

Fuente: Propia

Tabla 4: Historia de Usuario 2

HISTORIA DE USUARIO		
Número: 2	Usuario: Administrador	
Nombre historia: Base de datos.		
Prioridad: Alta	Riesgo: Alto	Estimación: 10 horas
Descripción:		
<ul style="list-style-type: none"> • Gestor de base de datos gratuito. • La base de datos debe poseer las tablas entidad-relación. 		
Pruebas de aceptación: No existe límite de tablas.		

Fuente: Propia

Tabla 5: Historia de Usuario 3

HISTORIA DE USUARIO	
Número: 3	Usuario: Administrador.

Nombre historia: Herramientas de Desarrollo.

Prioridad: Alto

Riesgo: Alto

Estimación: 8 horas

Descripción: Framework para el desarrollo gratuitos.

Pruebas de aceptación: Herramientas open Source.

Fuente: Propia.

Tabla 6: Historia de Usuario 4

HISTORIA DE USUARIO

Número: 4

Usuario: Administrador/Operador/Empleado

Nombre historia: Gestión del Módulo de productos.

Prioridad: Alta

Riesgo: Alto

Estimación: 10 horas

Descripción:

- El sistema debe permitir ingresar y modificar productos.
 - Se debe mostrar una lista de los productos.
 - Se debe mostrar un detalle Producto.
 - Dar de Baja el Producto.
-

Pruebas de aceptación:

- El sistema mostrará mensajes de alerta de almacenamientos o de errores.
 - Los campos obligatorios no deben ser ingresados vacíos.
 - Si no existe datos en la lista mostrar mensaje.
 - La lista poseerá un criterio de búsqueda.
-

Fuente: Propia

Tabla 7 : Historia de Usuario 5

HISTORIA DE USUARIO

Número: 5

Usuario: Administrador/Empleado

Nombre historia: Gestión del Módulo de Clientes y Proveedores.

Prioridad: Media

Riesgo: Medio

Estimación: 8 horas

Descripción:

- El sistema debe permitir ingresar y modificar a clientes.
- El sistema debe permitir ingresar y modificar a proveedores.
- Se debe mostrar una lista de clientes y proveedores que muestre el detalle de cada uno.

Pruebas de aceptación:

- El sistema mostrará mensajes de alerta de almacenamientos o de errores.
- Los campos obligatorios no deben ser ingresados vacíos.
- Si cédula o ruc se encuentra registrado mostrar mensaje.
- Si no existe datos en la lista mostrar mensaje.
- La lista poseerá un criterio de búsqueda.

Fuente: Propia.

Tabla 8 : Historia de Usuario 8

HISTORIA DE USUARIO

Número: 6

Usuario: Administrador/Empleado

Nombre historia: Gestión del Módulo de Ventas.

Prioridad: Alto

Riesgo: Medio

Estimación: 8 horas

Descripción:

- El sistema debe permitir ingresar y anular facturas.
- Se debe mostrar la lista de las facturas con su respectivo detalle.
- Se imprimirá la factura en pdf.
- Se imprimira en excel las facturas de acuerdo a un criterio de busqueda.

Pruebas de aceptación:

- El sistema mostrará mensajes de alerta de almacenamientos o de errores.
- Los campos obligatorios no deben ser ingresados vacíos.
- Si no existe datos en la lista mostrar mensaje.
- Se anulará la factura antes de las 24 horas creada.
- Si no existe datos en la lista mostrar mensaje.
- La lista poseerá un criterio de búsqueda.

Fuente: Propia.

Tabla 9: Historia de Usuario 7

HISTORIA DE USUARIO

Número: 7

Usuario: Administrador/Operador

Nombre historia: Gestión del Módulo de Compras.

Prioridad: Alto

Riesgo: Medio

Estimación:8 horas

Descripción:

- El sistema debe permitir ingresar y anular compras.
 - El sistema debe ingresar pagos de la compra como abonos.
 - Se debe mostrar la lista de las compras con su respectivo detalle.
 - Se imprimirá la compra en pdf.
 - Se imprimira en excel las compras deacuerdo a un criterio de busqueda.
-

Pruebas de aceptación:

- El sistema mostrará mensajes de alerta de almacenamientos o de errores.
 - Los campos obligatorios no deben ser ingresados vacíos.
 - Si no existe datos en la lista mostrar mensaje.
 - La lista poseerá un criterio de búsqueda.
-

Fuente: Propia.

Tabla 10: Historia de Usuario 8

HISTORIA DE USUARIO

Número: 8

Usuario: Administrador/Operador

Nombre historia: Gestión del Módulo de Pedidos y Stock.

Prioridad: Media

Riesgo: Medio

Estimación:8 horas

Descripción:

- El sistema debe permitir ingresar y actualizar pedidos.
 - El Stock del producto permitirá llevar un control de entrada y salida del productos.
 - Se debe mostrar la lista de los pedidos con su respectivo detalle.
 - El pedido se imprimira en archivo excel.
-

Pruebas de aceptación:

- El sistema mostrará mensajes de alerta de almacenamientos o de errores.
 - Los campos obligatorios no deben ser ingresados vacíos.
 - Si no existe datos en la lista mostrar mensaje.
 - La lista poseerá un criterio de búsqueda.
-

Fuente: Propia.

Tabla 11 : Historia de Usuario 9

HISTORIA DE USUARIO		
Número: 9	Usuario: Administrador/Operador/Empleado	
Nombre historia: Gestión del Módulo Kardex.		
Prioridad: Media	Riesgo: Medio	Estimación: 8 horas
Descripción:		
<ul style="list-style-type: none"> • Filtros de búsqueda en entrada y salida del producto. • Reporte de movimientos del producto. • Imprimir reporte en Excel. 		
Pruebas de aceptación: Los filtros de búsqueda permitirá generar archivos Excel.		
Fuente: Propia.		

Tabla 12 . Historia de Usuario 10

HISTORIA DE USUARIO		
Número: 10	Usuario: Administrador	
Nombre historia: Gestión del Módulo de Seguridad.		
Prioridad: Media	Riesgo: Medio	Estimación: 8 horas
Descripción:		
<ul style="list-style-type: none"> • Se debe permitir acceder únicamente a las funciones del sistema al personal autorizado. • El usuario debe ingresar sus credenciales para ingresar al sistema. • El administrador podrá bloquear usuario y restaurar contraseña. 		
Criterio de validación:		
<ul style="list-style-type: none"> • Si el usuario no existe no debe ingresar al sistema. • El usuario debe acceder a las funciones dependiendo de su rol. 		
Fuente: Propia.		

2.2.2. Aprobación del Sprint

A continuación, se asigna las historias de usuarios a los Sprint, la definición de cada Sprint consiste en estimar un tiempo por semanas. Ver Tabla 13.

Tabla 13. Aprobación del Sprint

ID	Historia	Estimación dificultad	Sprint número	Fecha Sprint
1	Levantamiento de requerimiento.	8	Sprint 1	(20/12/2019)
2	Base de datos.	8		al
3	Herramientas de Desarrollo.	5		(30/12/2019)
4	Gestión del Módulo de productos.	6	Sprint 2	(02/01/2020)
5	Gestión del Módulo de Clientes y Proveedores.	5		al
6	Gestión del Módulo de Ventas.	6	Sprint 3	(15/01/2020)
7	Gestión del Módulo de Compras.	6		al
8	Gestión del Módulo de Pedidos y Stock	5	Sprint 4	(15/01/2020)
9	Kardex	4		al
10	Gestión del Módulo de Seguridad	10	Sprint 5	(06/03/2020)
				al
				(26/03/2020)

Fuente: Propia

2.2.3. Estimación de tareas por Sprint

A continuación, se realiza la estimación de los Sprints, que contienen la fecha inicio y fecha fin, además se establece el tiempo de las tareas por horas.

Tabla 14: Sprint 1

Sprint 1			
Fecha inicio: 20/12/2019			
Fecha fin: 30/12/2019			
ID	Historia de usuario	Tarea	Horas
1	Levantamiento de requerimiento.	Revisión del proceso de administración de escenarios.	120
		Revisión del framework Spring Boot.	16
		Revisión del framework Angular.	16
2	Base de datos.	Modelar base de datos.	45
		Normalizar base de datos.	40
		Pruebas de base de datos.	20
		Reestructurar base de datos.	30
		Configurar conexión base de datos y Spring Boot.	20
3	Herramientas de Desarrollo.	Descargar e instalar JDK versión 8.	2
		Descargar e instalar MySql.	2
		Descargar e instalar Visual Code	2
		Descargar e instalar Node.js y Angular.	2
		Descargar e instalar Spring Tool suite.	2
		Total	317

Tabla 15: Sprint 2

Sprint 2			
Fecha inicio: 02/01/2020			

Fecha fin: 14/01/2020			
ID	Historia de usuario	Tarea	Horas
4	Gestión del Módulo de productos.	Configuración del mapeo en SpringBoot.	5
		Creación y configuración del controller y del repository en SpringBoot.	5
		Creación y configuración de los componentes y servicios en angular.	15
		Creación de las pantallas de ingreso y modificación de las productos.	15
		Mostrar la lista de las productos.	10
5	Gestión del Módulo de clientes y proveedores.	Configuración del mapeo en SpringBoot.	5
		Creación y configuración del controller y del repository en SpringBoot.	5
		Creación y configuración de los componentes y servicios en angular.	15
		Creación de las pantallas de ingreso y modificación de las clientes y proveedores.	15
		Mostrar la lista de clientes y proveedores.	10
		Total	100

Tabla 16: Sprint 3

Sprint 3			
Fecha inicio: 15/01/2020			
Fecha fin: 14/02/2020			
ID	Historia de usuario	Tarea	Horas
6	Gestión del Módulo de ventas.	Configuración del mapeo en SpringBoot.	5
		Creación y configuración del controller y del repository en SpringBoot.	5
		Creación y configuración de los componentes y servicios en angular.	15
		Creación de las pantallas de ingreso de facturas.	15
		Mostrar la lista de facturas.	10
		Generar Reporte de ventas.	10
		Imprimir detalle de Factura	5
7	Gestión del Módulo de clientes y proveedores.	Configuración del mapeo en SpringBoot.	5
		Creación y configuración del controller y del repository en SpringBoot.	5
		Creación y configuración de los componentes y servicios en angular.	15
		Creación de las pantallas de ingreso de compra de mercadería.	15
		Mostrar la lista de compras.	10
		Generar Reporte de compras.	10
		Imprimir detalle de la Compra	5
		Total	130

Tabla 17 : Sprint 4

Sprint 4			
Fecha inicio: 15/02/2020			
Fecha fin: 05/03/2020			
ID	Historia de usuario	Tarea	Horas
6	Gestión del Módulo Pedidos y Stock.	Configuración del mapeo en SpringBoot.	5
		Creación y configuración del controller y del repository en SpringBoot.	5
		Creación y configuración de los componentes y servicios en angular.	15
		Creación de las pantallas de ingreso de Pedidos.	15
		Mostrar la lista de pedidos.	10
		Mostrar Lista de Stock.	10
		Generar Archivo Excel de Stock.	5
7	Gestión del Kardex.	Configuración del mapeo en SpringBoot.	5
		Creación y configuración del controller y del repository en SpringBoot.	5
		Creación y configuración de los componentes y servicios en angular.	15
		Mostrar filtros de búsqueda de productos.	10
		Generar Reporte por detalle de producto.	5
		Imprimir detalle del Kardex	5
Total			110

Tabla 18: Sprint 5

Sprint 5			
Fecha inicio: 06/03/2020			
Fecha fin: 26/03/2020			
ID	Historia de usuario	Tarea	Horas
		Configuración del mapeo en SpringBoot.	5
		Creación y configuración del controller y del repository en SpringBoot.	5
6	Gestión del Módulo de Seguridad	Configuración del mapeo en SpringSecurity.	15
		Creación y configuración de los componentes y servicios en angular.	15
		Control de perfiles de acceso.	100
		Total	140

2.3. Fase 3. - Proceso de desarrollo del sistema web.

A continuación, se especifica algunos pasos del proceso de desarrollo del sistema web.

2.3.1. Modelo de la base de datos del sistema web.

En siguiente modelo indica la estructura lógica de la base de datos del módulo de ventas e inventarios.

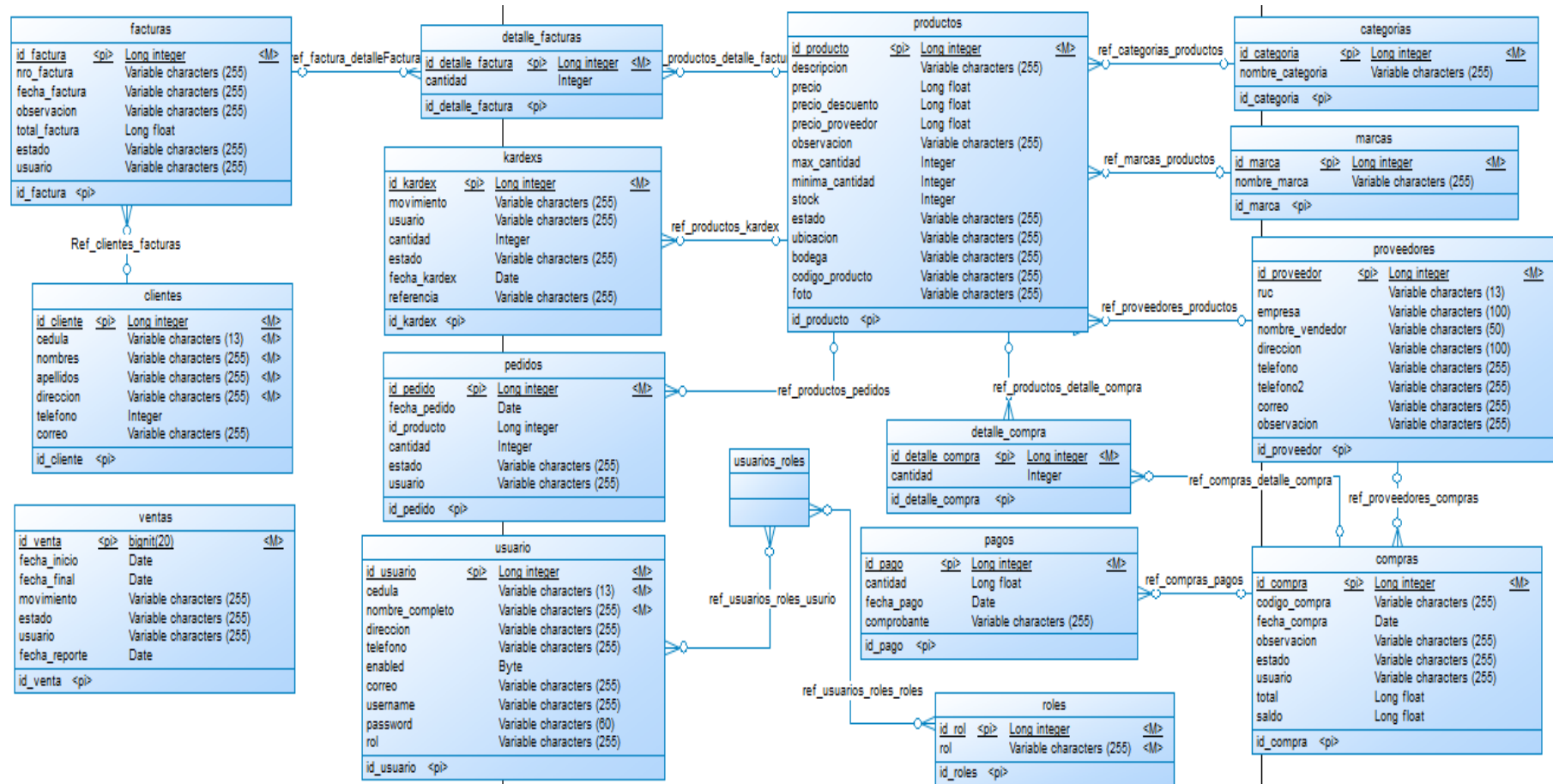


Fig. 13: Base de Datos Módulo de Ventas e inventario

Fuente: Propia.

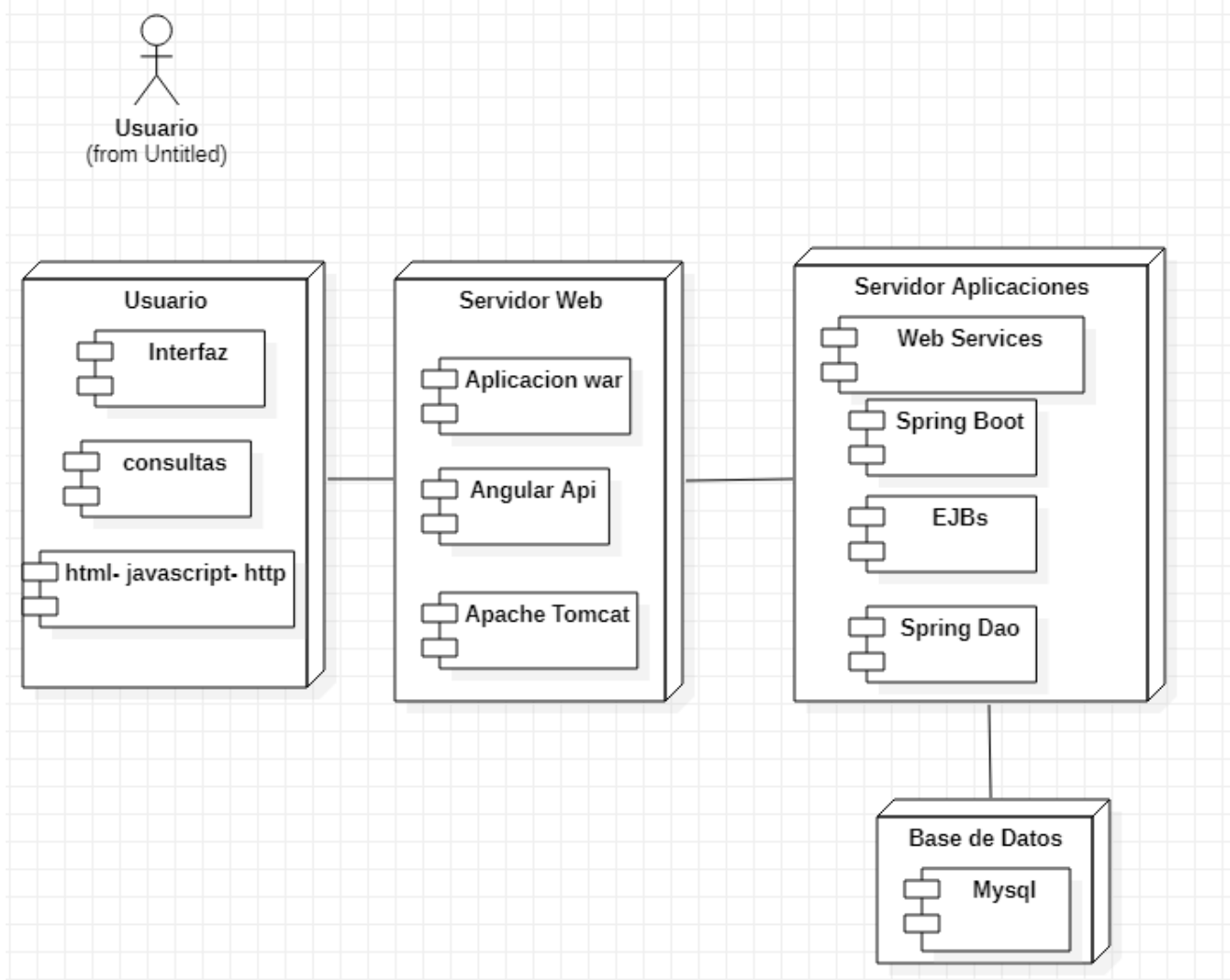


Fig. 14 : Diagrama de Componentes
Fuente: Propia.

2.3.2. Prototipo del sistema web.

A continuación, se indica las capturas de funcionamiento del software.

Inicio de sesión.

Caso de usos del usuario.

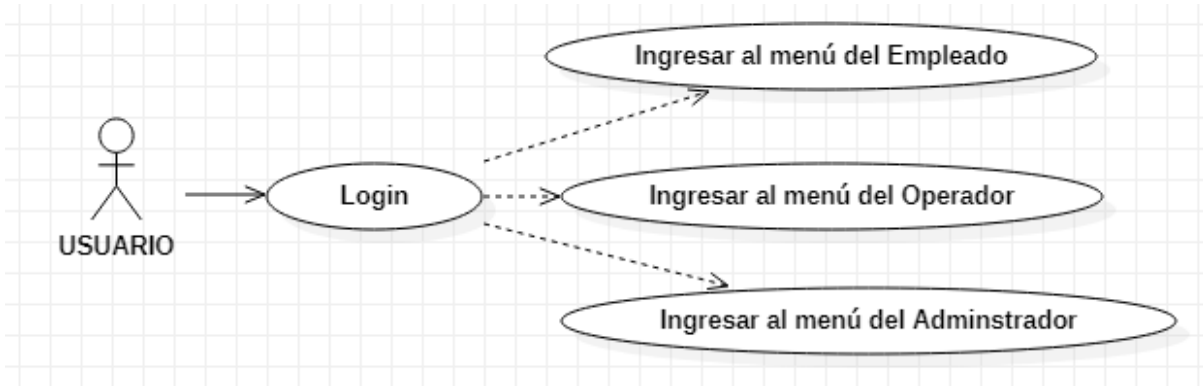
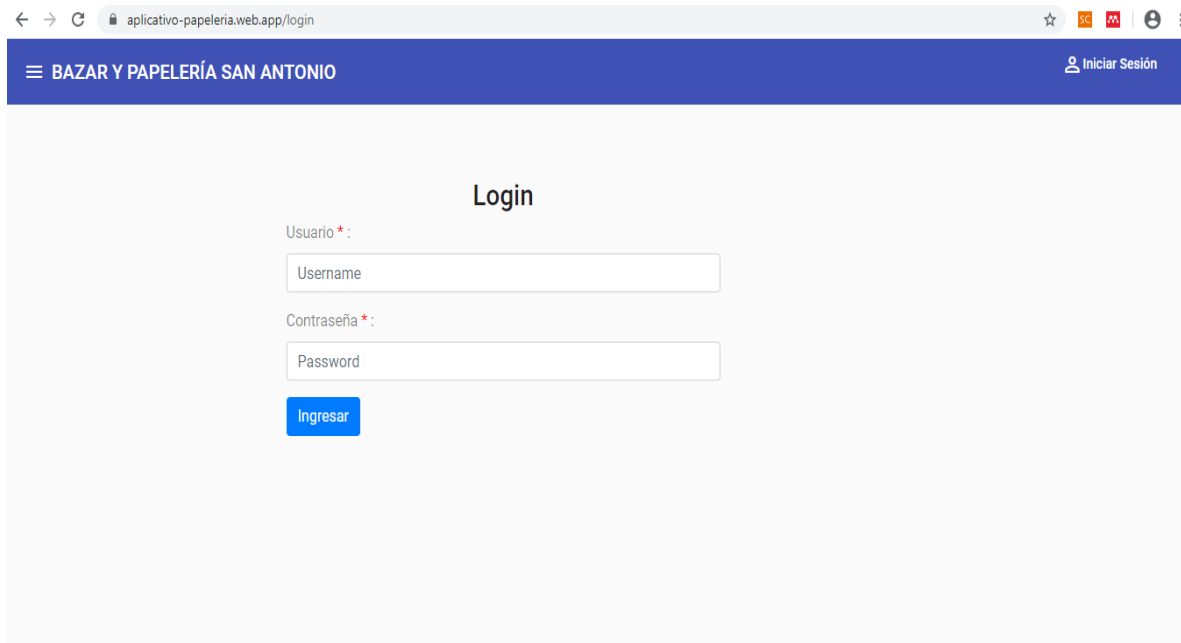


Fig. 15: Casos de Uso del Login
Fuente: Propia.

Para ingresar, el sistema cuenta con un inicio de sesión, donde el usuario debe ingresar sus credenciales. Existen 3 roles: Administrador, Operador y Empleado.



© PAOLA TITO 2020

Fig. 16: Página Inicio de Sesión
Fuente: Propia.

Administración del Operador.

Caso de usos del Operador.

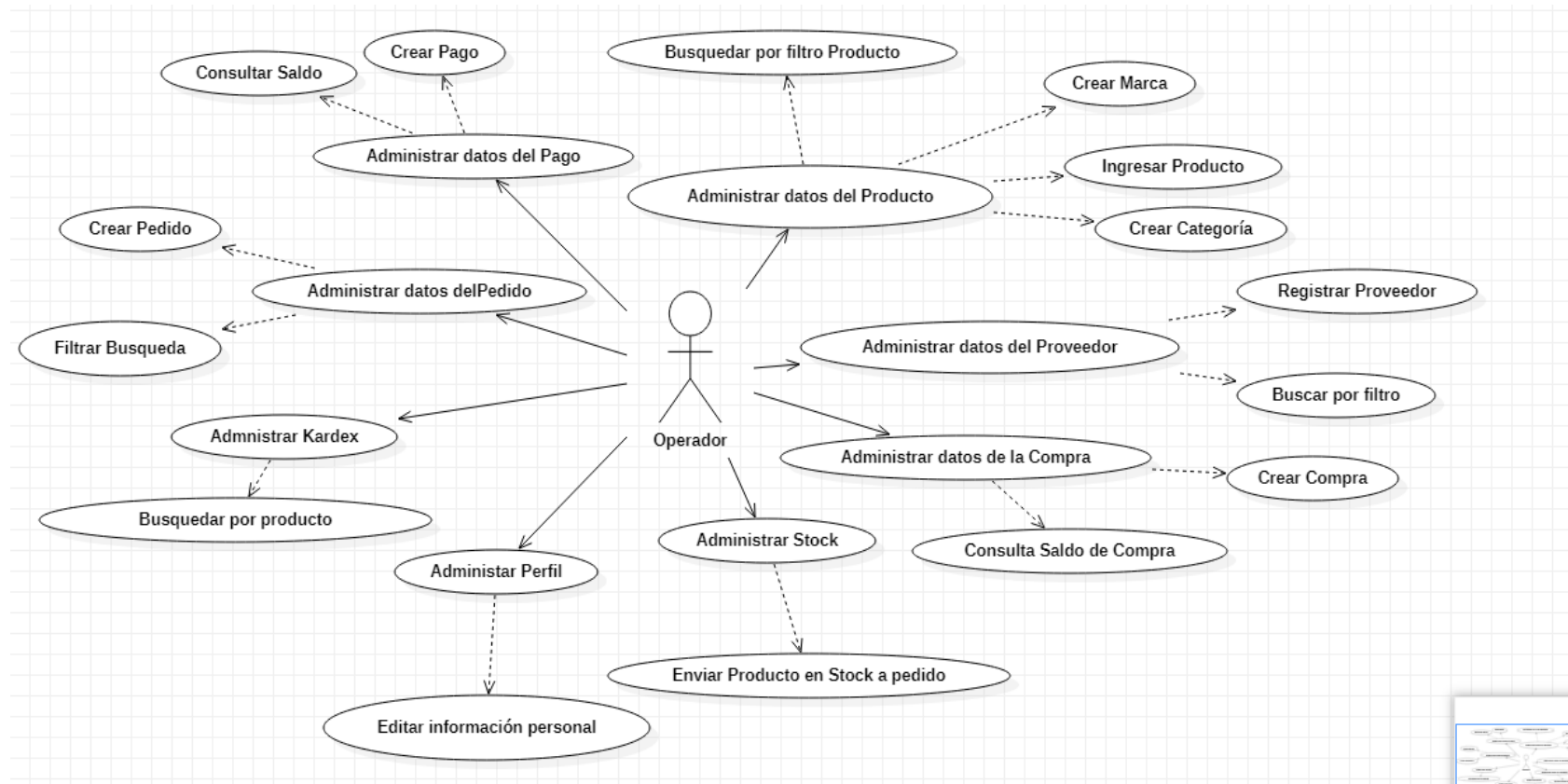


Fig. 17: Caso de Usos Operador.
Fuente: Propia.

En la Fig 18. se muestra la página principal del operador con las funciones al cual él puede acceder. El operador puede acceder a Proveedor, Compras, Kardex, Producto, Stock, Pedido y Reportes.

Menú

- PRODUCTOS
- PROVEEDORES
- COMPRAS
- PAGOS
- PEDIDOS
- KARDEX
- REPORTES

Productos **Nuevo Producto** Stock Marcas Categorías

Lista de Productos

Mostrar Buscar Exportar

Imagen	Código	Descripción	Marca	Proveedor	P. Unitario	P. Descuento	Ubicación	Cant. Mínima	Stock	Estado	Acciones
	COD-55554	Archivador Grande	Utilesa	Dilipa	\$3,80	\$3,50	Bodega 2: Estantería 4	12	0	Nuevo	
	COD-45555	Grapadora Pequeña de Colores	Artesco	Dilipa	\$1,00	\$0,94	Bodega 1: Estantería 12	6	0	Nuevo	
	COD-55665	Esferos punta media Azul	Bic	Dilipa	\$0,35	\$0,30	Bodega 1: Estantería 4	60	0	Nuevo	
	COD-56656	Maza Daz 250 mg	Pelikan	Orgrama	\$1,60	\$1,40	Bodega 1: Estantería 4	24	96	Activo	
	COD-55545	Cuaderno 100 hojas a Cuadros	Bic	Dilipa	\$1,25	\$1,10	Bodega 1: Estantería 12	24	98	Activo	

Primera 1 Última

Fig. 18: Página principal del Operador.
Fuente: Propia.

Productos.

En la Fig 19. se muestra la página para ingresar producto.

Menú

- PRODUCTOS
- PROVEEDORES
- COMPRAS
- PAGOS
- PEDIDOS
- KARDEX
- REPORTES

Productos **Nuevo Producto** Stock Marcas Categorías

Producto

Descripción del Producto * :

Marca * : Categoría * : Proveedor * :

Precio * : Precio Descuento * : Precio Proveedor * :

Cantidad Mínima * : Cantidad Máxima * : Bodega * : Ubicación * :

Observación :

Crear

Fig. 19: Página Crear producto
Fuente: Propia.

Proveedores.

En la Fig 20 y Fig 21. se muestra la página para ingresar, listar proveedores.

Menú

PRODUCTOS

PROVEEDORES

COMPRAS

PAGOS

PEDIDOS

KARDEX

REPORTES

Proveedores [Nuevo Proveedor](#)

Listado de Proveedores

Mostrar Buscar

RUC	EMPRESA	VENDEDOR	TELÉFONO	CORREO	ACCIONES
1001860731	Dilpa	Jorge Lopez	062932241	jorge@gmail.com	
1003493481001	Orgrama	Carlos Maya	062932241	carlos@orgrama.com	

Primera **1** Ultima

Fig. 20: Página de Lista de Proveedores.
Fuente: Propia

Menú

PRODUCTOS

PROVEEDORES

COMPRAS

PAGOS

PEDIDOS

KARDEX

REPORTES

Proveedores [Nuevo Proveedor](#)

Proveedor

RUC * : Nombre de la Empresa * :

Vendedor * : Dirección * :

Correo * : Teléfono * : Teléfono Opcional * :

Observación

Crear

© PAOLA TITO 2020

Fig. 21: Página Crear Proveedor.
Fuente: Propia

Compras.

En la Fig 22,23, 24. se muestra la página para ingresar, listar y reporte de las compras.

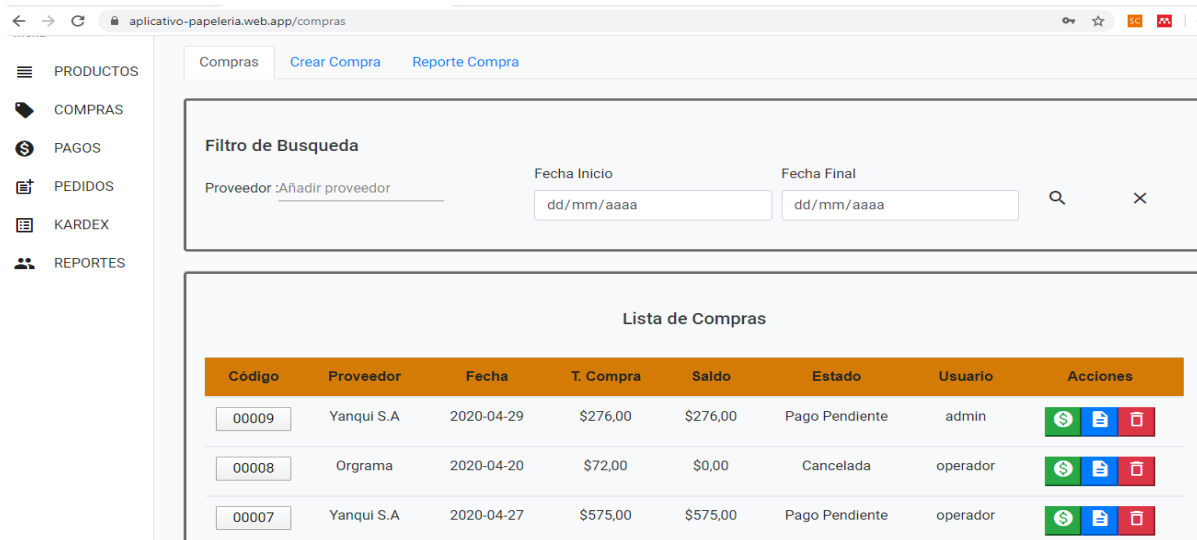


Fig. 22: Página Lista de Compras.
Fuente: Propia.

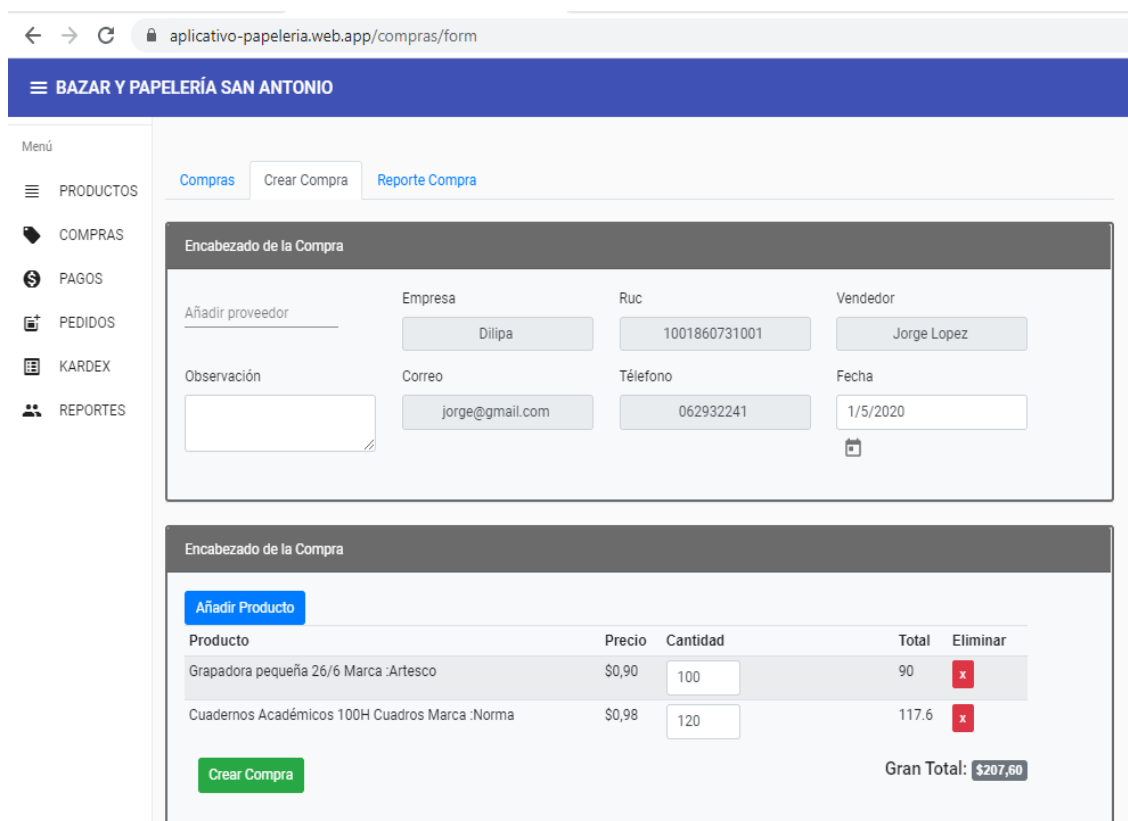


Fig. 23: Crear Compra
Fuente: Propia.

Nro Compra 00009 « Regresar

Datos del Proveedor

Yanqui S.A

Ana Fuentes

Datos de la Compra

Fecha: miércoles 29, abril 2020

Estado: Pago Pendiente

Producto	Precio	Cantidad	Total
Calculadora f-350 Color Negro	\$11,50	24	276

Observaciones

No tiene observación

Total: \$276,00

Fig. 24: Página Detalle Compra
Fuente: Propia.

Pagos.

En la Fig 25 y 26. se muestra la página para ingresar, listar los pagos de las compras

BAZAR Y PAPELERÍA SAN ANTONIO

Menú

- PRODUCTOS
- PROVEEDORES
- COMPRAS
- PAGOS**
- PEDIDOS
- KARDEX
- REPORTES

Pagos [Crear Pagos](#)

Filtro de Búsqueda de Pagos

Search...

Listado de Pagos Exportar

RECIBO	FECHA	PROVEEDOR	CÓD COMPRA	ABONO	ACCIONES
	2020-04-30	Dilipa	00001	48	
	2020-04-30	Dilipa	00001	50	

Primera **1** Última

Fig. 25: Página Lista de Pago
Fuente: Propia

Pago Detalle Regresar

Proveedor: Dilipa

Código Compra: 00001

Fecha: 30/04/2020

Cantidad: 48

COMPROBANTE

CODIGO	CONCEPTO	CANTIDAD	ASIGNACIÓN	DEDUCCION
001	Sueldo	15	7.000,00	
101	Bono, Comision, Prestamo	1	2.350,00	
106	Dias Feriados	1	700,00	

Fig. 26: Página Detalle Pago.
Fuente: Propia.

Pedidos.

En la Fig 27. se muestra la página para ingresar, listar los pedidos.

← → ↻ aplicativo-papeleria.web.app/pedidos

BAZAR Y PAPELERÍA SAN ANTONIO

Menú

- PRODUCTOS
- COMPRAS
- PAGOS
- PEDIDOS
- KARDEX
- REPORTES

Pedidos [Reporte Pedido](#)

Pedido

Añadir producto Cantidad + Crear

Lista de Pedidos

Mostrar Buscar

Exportar

ID	PRODUCTO	MARCA	FECHA	CANTIDAD	ESTADO	USUARIO	ACCIONES
2	Masa Daz Blanca 250 mg	Bic	2020-05-01	48	Enviado	operador	<input type="button" value="✓"/> <input type="button" value="✎"/> <input type="button" value="✖"/>
1	Sacapuntas Maped Metálico	Pelikan	2020-05-01	120	Despachado	admin	<input type="button" value="✓"/> <input type="button" value="✎"/> <input type="button" value="✖"/>

Primera Última

Fig. 27: Página de lista Pedidos
Fuente: Propio.

Kardex.

En la Fig 25. se muestra la página para listar los movimientos del producto.

Fecha	Producto	Movimiento	Estado	Cantidad	Usuario
2020-05-01	Colores Norma doble punta	Entrada	Registrada	36	admin
2020-05-01	Colores Norma doble punta	Salida	Registrada	2	admin
2020-05-01	Colores Norma doble punta	Salida	Registrada	1	andres

Fig. 28: Página de kardex.
Fuente: Propio.

Reportes.

En la Fig 26. se muestra la página de los reportes en ventas y compras.

Fecha Inicio	Fecha Final	Movimiento	Referencia	Total	Usuario	Fecha de Emisión	Acción
2020-04-01	2020-04-30	Compras	Reporte por Fecha	\$1.021,00	operador	2020-05-01	
2020-05-01	2020-05-01	Facturación	Reporte Diario por : andres	\$30,05	andres	2020-05-01	

Fig. 29: Página de Reportes.
Fuente: Propia.

Administración del Empleado.

Casos de Usos Empleado.

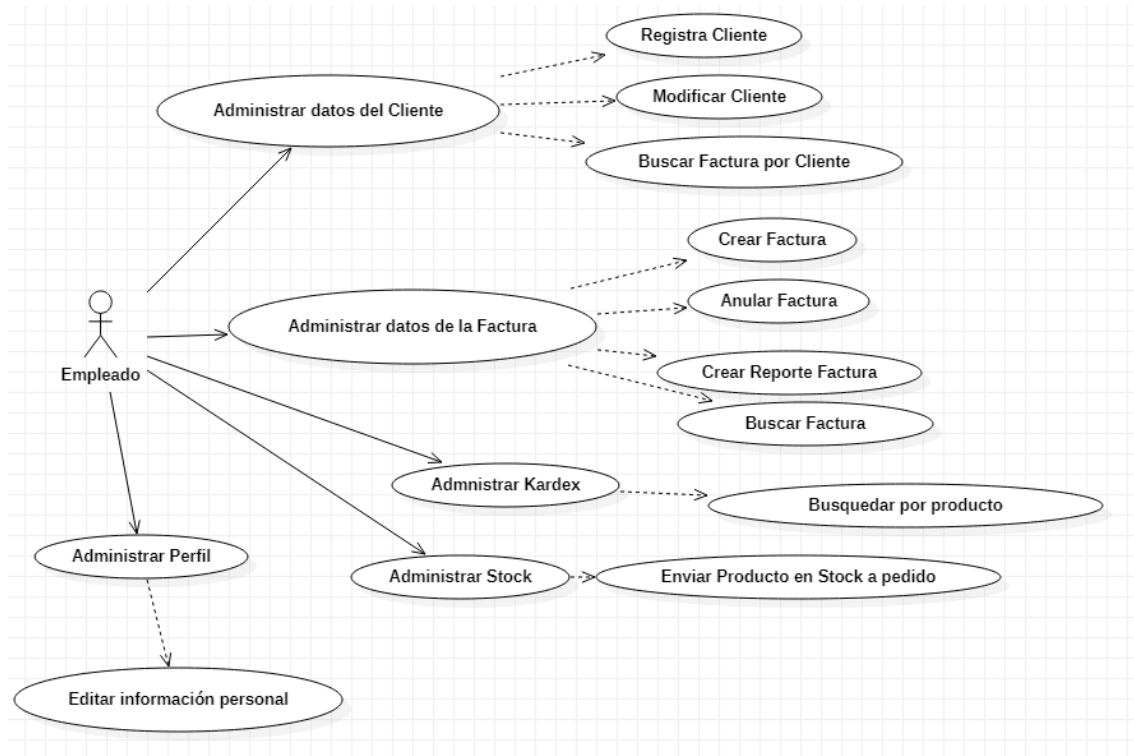


Fig. 30: Caso de Usos Empleado.
Fuente: Propia

En la Fig 31. se muestra la página principal del empleado con las funciones al cual él puede acceder.

Imagen	Código	Descripción	Marca	Proveedor	P. Unitario	P. Descuento	Ubicación	Cant. Mínima	Stock	Estado
	COD-56466	Mouse Inalámbrico	Artesco	Orgrama	\$8,90	\$8,75	Almacen: Vitrina principal	6	105	Activo
	COD-56454	Corrector 9ML	Pelikan	Utilesa S.A	\$1,00	\$0,95	Bodega 1: Estanteria 5, FILA 4	24	0	Nuevo
	COD-55564	Calculadora f-350 Color Negro	Pelikan	Yanqui S.A	\$14,00	\$12,50	Almacen: Vitrina principal	6	124	Activo

Fig. 31: Página Principal de Empleado.
Fuente: Propia.

Stock.

En la Fig 32. se muestra la página para enviar productos a pedidos que se encuentra en Stock.

Menú

- PRODUCTOS
- CLIENTES
- VENTAS
- KARDEX
- REPORTES

Productos Stock

Mostrar 1 entradas

Buscar Exportar

Descripción	Marca	Categoría	Proveedor	Cantidad Mínima	Stock	Ubicación	Estado	Pedido
Masa Daz Blanca 250 mg	Bic	Utiles Escolares	Mercantiles	12	3	Bodega 4: Estantería 5	Pedido Registrado	

Primera 1 Última

Fig. 32: Página Stock
Fuente: Propia

Clientes.

En la Fig 33. se muestra la página para ingresar y listar los clientes.

Menú

- PRODUCTOS
- CLIENTES
- VENTAS
- KARDEX
- REPORTES

Clientes Nuevo Cliente

Mostrar 5 entradas

Buscar

Cédula	Nombres	Dirección	Teléfono	Correo	Editar
1002261533	Ximena Suarez	Ibarra	9999999	ximena@gmail.com	
0400403986	Laura Cevallos	Ibarra	9999999	laura123@gmail.com	
0400157129	Maria Esperanza Chulde Revelo	Ibarra	062343434	mary@gmail.com	
0400510054	Carlos Anibal Chulde Lima	Ibarra el Sagrario	062343434	anibal@gmail.com	
1003947163	Jefferson David Tito Chulde	San Antonio	062933566	david@gmail.com	

Primera 1 2 Última >

Fig. 33: Página Clientes
Fuente: Propia.

Ventas.

En la Fig 34. se muestra la página para ingresar, listar y visualizar el detalle de facturas.

4

The screenshot shows the 'Facturas' page of the 'BAZAR Y PAPELERÍA SAN ANTONIO' application. The page has a sidebar menu with options: PRODUCTOS, CLIENTES, VENTAS, KARDEX, and REPORTE. The main content area includes a search filter and a table of invoices.

Filtro de Búsqueda

Fecha Inicio *: Fecha Desde *: Estado: Usuario:

Lista de Facturas

N° Factura	Fecha	Cédula	Cliente	Estado	Usuario	Acciones
00004	2020-05-01	0400157129	Maria Esperanza Chulde Revelo	Registrada	admin	
00003	2020-05-01	0400403986	Laura Cevallos	Registrada	admin	
00002	2020-05-01	0400157129	Maria Esperanza Chulde Revelo	Registrada	admin	
00001	2020-05-01	1003947163	Jefferson David Tito Chulde	Anulada	admin	

Fig. 34: Página Ventas
Fuente: Propia

The screenshot shows the 'Crear Factura' page of the 'BAZAR Y PAPELERÍA SAN ANTONIO' application. The page has a sidebar menu with options: PRODUCTOS, CLIENTES, VENTAS, KARDEX, and REPORTE. The main content area includes a form for creating a new invoice.

Encabezado de la Factura

Cliente: Cédula o Ruc: Teléfono:

Observación: Dirección: Correo: Fecha:

Añadir Producto

Añadir producto:

Detalle Factura

Producto	Marca	Precio	Cantidad	stock	Total	Eliminar
Esferos Punta Fina Azul	Bic	\$0,30	<input type="text" value="1"/>	215	\$0,30	

Gran Total: **0.3**

Fig. 35: Página Crear Factura.
Fuente: Propia.

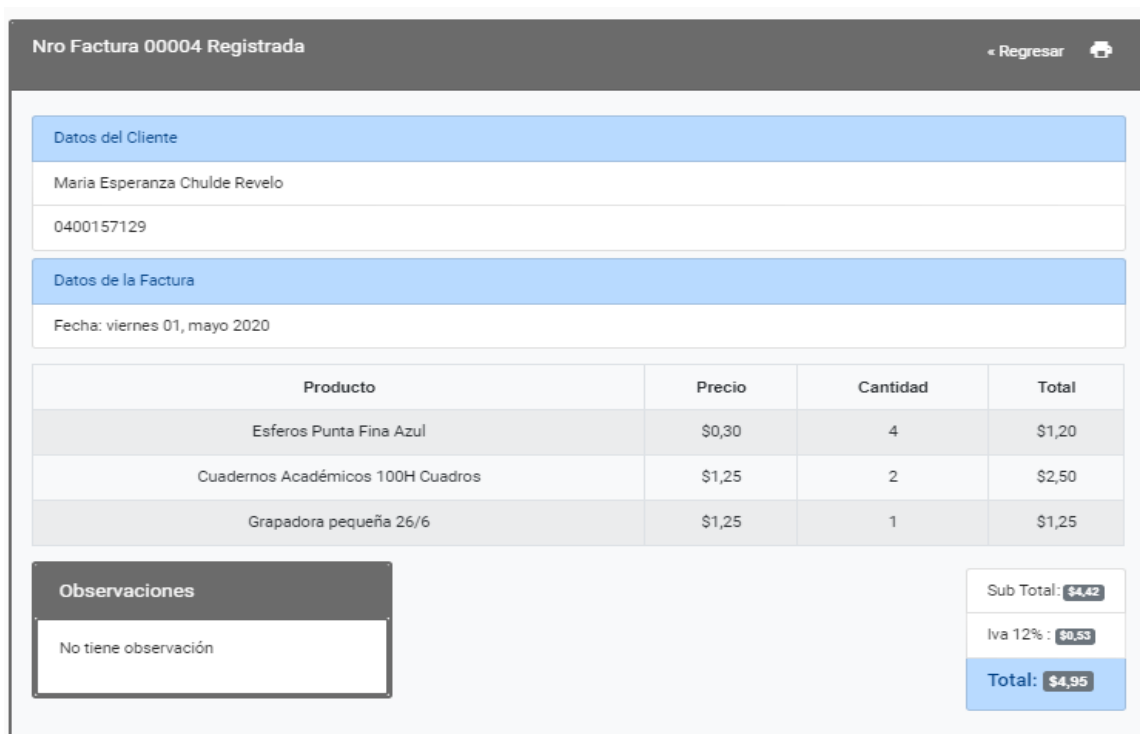


Fig. 36: Página Detalle factura.
Fuente: Propia.

Administrador.

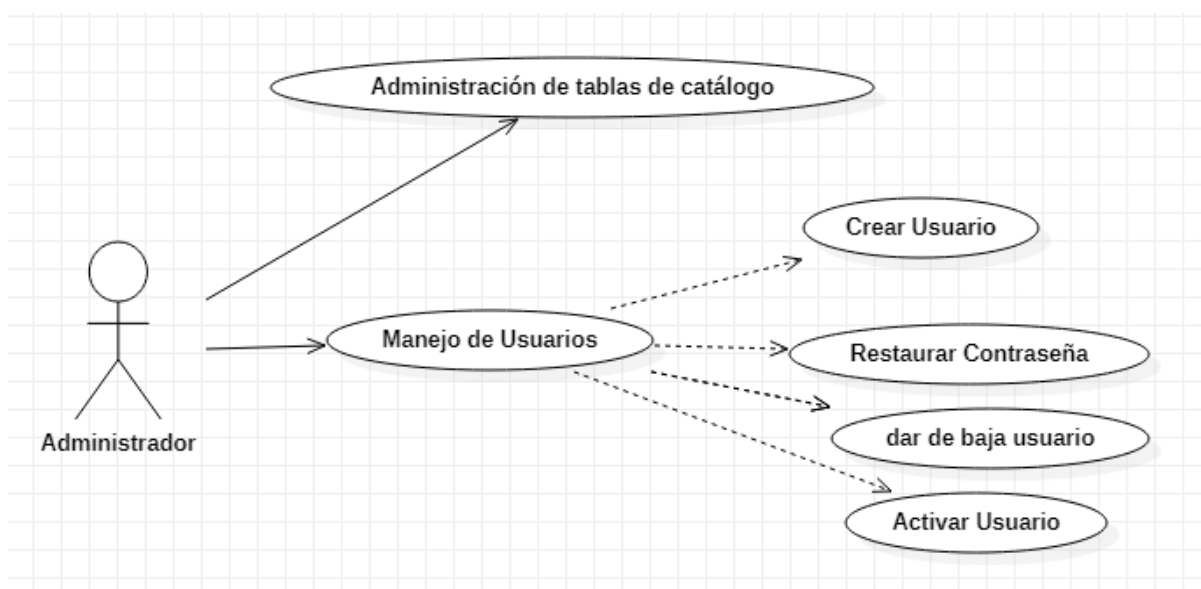


Fig. 37: Caso de Usos Administrador.
Fuente: Propia.

En la Fig 38. se muestra la página principal del administrador él puede acceder a todas las funciones, adicional crear y administrar usuarios.

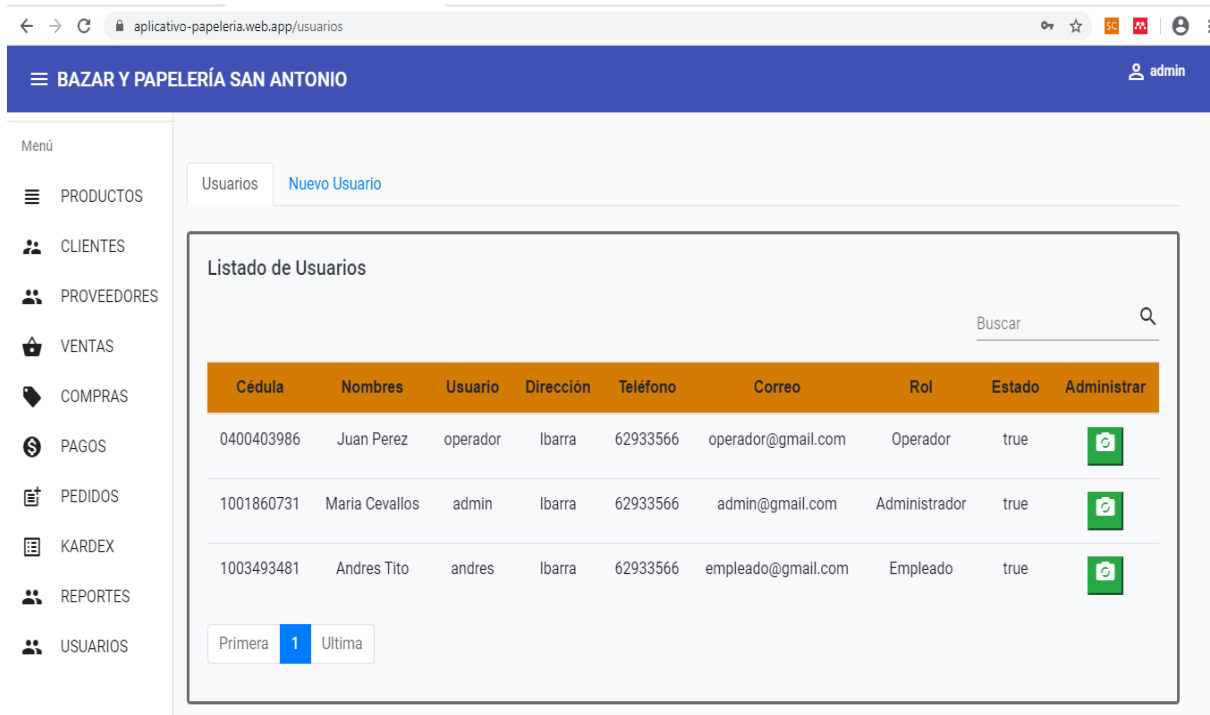


Fig. 38: Página del Administrador.
Fuente: Propia.

Fig 1:

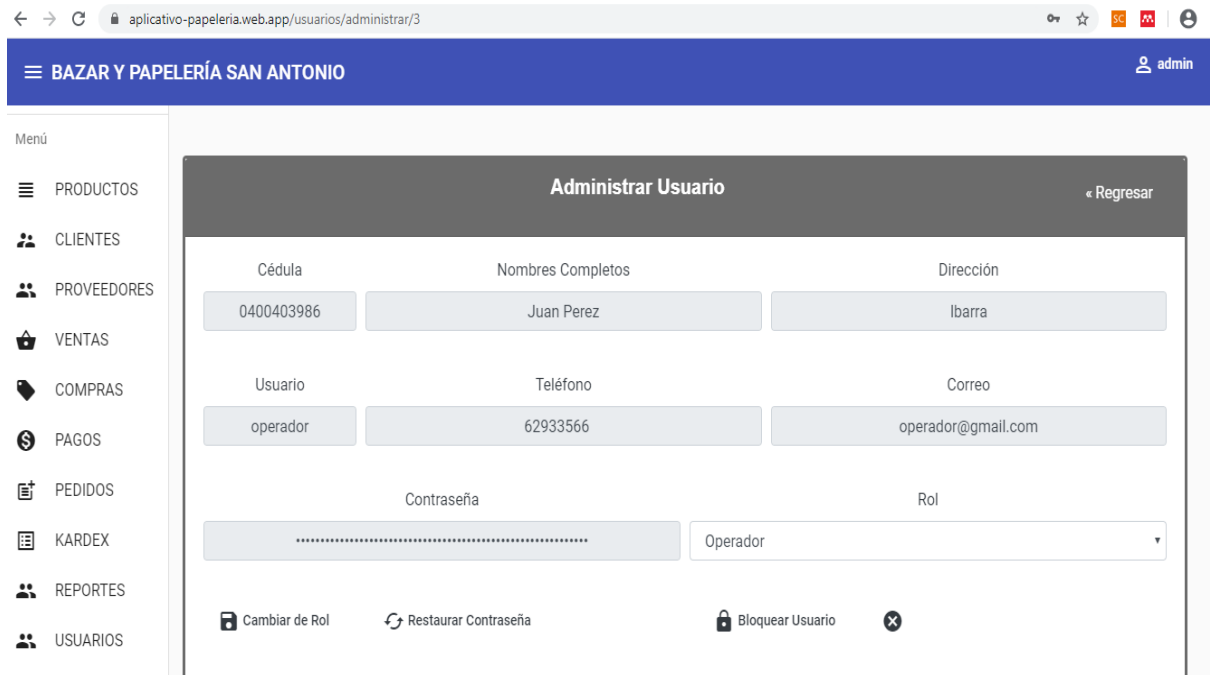


Fig. 39. Página Administrador de Usuario.
Fuente: Propia.

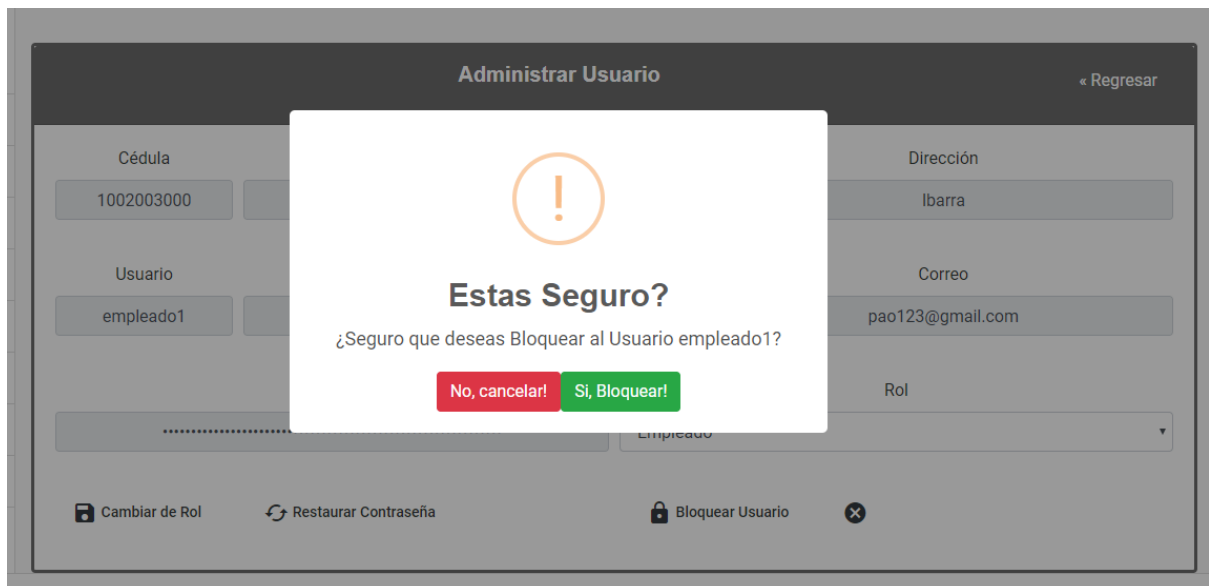


Fig. 40: Bloquear Usuario
Fuente: Propia.

CAPÍTULO 3

3. Resultados

3.1 Verificación de Resultados.

Después del desarrollo e implementación de sistema, se realiza la etapa de pruebas, terminado el proceso de desarrollo y pruebas, se definió en conjunto con el Product Owner y el Scrum Master aplicar la matriz de medición de la calidad en uso conforme indica la norma ISO/IEC 25010, que muestra la manera como medir cada una de las subcaracterísticas del modelo de calidad.

Se identificó las características y subcaracterísticas más importantes en el contexto que se desarrolló el proyecto. La característica escogida es satisfacción y subcaracterística utilidad.

3.1.1 Medición del Modelo de Calidad en Uso.

- **Datos Informativos del Sistema.**

La información ingresada en la matriz permite identificar de manera objetiva el producto de software para aplicar la evaluación.

MATRIZ DE CALIDAD DE SOFTWARE		
1. DATOS INFORMATIVOS:		
Fecha:	25/3/2020	
Institución:	BAZAR Y PAPELERÍA SAN ANTONIO	
Nombre del Software:	DESARROLLO DE UN SISTEMA DE GESTIÓN Y CONTROL DE PROCESOS PARA LA MICROEMPRESA BAZAR Y PAPELERÍA SAN ANTONIO UTILIZANDO SPRING BOOT.	
OBJETIVOS GENERALES DEL SOFTWARE		
Desarrollar un sistema web que permita automatizar los procesos de gestión y control de la empresa Bazar y Papelería San Antonio utilizando Spring Boot		
OBJETIVOS ESPECÍFICOS DEL SOFTWARE		
<ul style="list-style-type: none">• Realizar un marco teórico sobre el framework de desarrollo Spring Boot y las herramientas de apoyo.• Desarrollar una aplicación web para la empresa Bazar y Papelería San Antonio utilizando la metodología SCRUM.• Verificar los resultados mediante la evaluación de calidad de uso, con la norma ISO 25010 con la característica de satisfacción y sub característica de utilidad.		
PARTICIPANTES		
Cargo	Nombre	Unidad
Desarrolladora	Paola Tito	

Fig. 41:Matriz de Calidad De Software.
Fuente: Adaptada (Vaca, 2017)

- Selección del tipo del producto de software.

En esta parte se determinó el tipo del producto de software en el que encaja la aplicación para aplicar la evaluación.

2. TIPO DE PRODUCTO SOFTWARE		
Producto	Clasificación de producto	Selección
Página Web (PW)	Estática	
	Animada	
	Dinámica	
	Portal Web	
	Tienda Virtual o Comercio Electrónico	
	Página Web con Gestor de Contenido	
	Página Web 2.0	
Base de Datos (BDD)	BDD jerárquica	
	BDD de red	
	BDD transaccional	
	BDD relacional	
	BDD multidimensional	
	BDD orientado a objetos	
	BDD documental	
Software de Aplicación (SA)	BDD deductiva	
	SA de productividad (editores de texto)	
	SA de entretenimiento (videojuegos)	
	SA de negocios (ERP)	
	SA de educación (programas interactivos de aprendizaje)	
	SA de tecnología (control de sistemas, médicas, etc.)	X

Fig. 42: Tipo de Software.
Fuente: Adaptada (Vaca, 2017)

- Selección de la característica y subcaracterísticas a evaluar

El parámetro de interés que se utilizó para la evaluación es la satisfacción en uso con un nivel de importancia alta.

8. SUBCARACTERÍSTICAS DE CALIDAD EN USO				
Característica	Subcaracterística	Nivel de Importancia	%	Total Característica
C1 - Efectividad	Efectividad	No aplica		0%
C2 - Eficiencia	Eficiencia	No aplica		0%
C3 - Satisfacción	Utilidad	Alta	100%	100%
C4 - Libertad de Riesgo	Libertad del riesgo económico	No aplica		0%
	Libertad del riesgo de salud y seguridad	No aplica		
	Libertad del riesgo ambiental	No aplica		
C5 - Cobertura de contexto	Completitud de Contexto	No aplica		0%
	Flexibilidad	No aplica		

Fig. 43: Característica de Calidad.
Fuente: Adaptada (Vaca, 2017)

- **Definición de la Muestra.**

Para obtener los elementos de las métricas se estableció el instrumento de encuesta SUS (System Usability Scale - Escala de Usabilidad del Sistema), para medir la satisfacción del usuario. La muestra se seleccionó mediante un muestreo no probabilístico por conveniencia, seleccionando a todo el personal que labora en la micro empresa.

- **Encuesta SUS.**

SUS propone 10 preguntas para medir la usabilidad de un sistema, las cuales están en una escala Likert, donde 1 significa Muy en desacuerdo y 5 Muy de acuerdo (Reyes, 2017).

Estas preguntas permitieron levantar datos para medir la subcaracterística utilidad. A continuación, se detalla el enunciado de cada una de ellas:

Tabla 19: Preguntas de Cuestionario.

Pregunta	Enunciado
Pregunta 1	¿Considera usted qué usaría este sistema frecuentemente?.
Pregunta 2	¿Considera usted qué el sistema fue fácil de usar?
Pregunta 3	¿Considera usted qué el sistema fue fácil de usar?
Pregunta 4	¿Considera usted qué necesitaría ayuda de una persona con conocimientos técnicos para usar este sistema?
Pregunta 5	¿Considera usted qué las funciones de este sistema están bien integradas?
Pregunta 6	¿Considera usted qué el sistema es inconsistente?.
Pregunta 7	¿Considera usted qué la mayoría de la gente aprendería a usar este sistema en forma rápida?.
Pregunta 8	¿Considera usted qué el sistema es difícil de usar?
Pregunta 9	¿Se siente confiado al usar el sistema?
Pregunta 10	¿Considera usted qué necesita aprender muchas cosas tecnológicas antes de usar el sistema?

Fuente: Propia.

3.1.2 Evaluación del Modelo de Calidad en Uso.

Para evaluar el modelo de calidad, primero se tabuló los datos obtenidos de la encuesta SUS, luego se aplicó las métricas establecidas. A continuación, se detalla las mediciones del modelo establecido.

3.1.3 Característica: Satisfacción

- **Subcaracterística: Utilidad.**

- **Métrica Nivel de Satisfacción**

La sub-característica de utilidad determina la relación entre el número de usuarios satisfechos con el uso del producto y el número total de usuarios, Las preguntas de la encuesta SUS seleccionadas para recolectar datos de medición de la “Utilidad” fueron las preguntas 2, 6, 8 y 10. Utilizando la escala de Likert.

Tabla 20 : Medición de Satisfacción

Respuesta	Escala
Muy de acuerdo	5
Algo de acuerdo	4
Ni de acuerdo ni en desacuerdo	3
Algo desacuerdo	2
Muy en desacuerdo	1

Fuente: Escala Likert.

Para obtener los usuarios satisfechos se realizó el cálculo por cada pregunta mediante la suma de todas las respuestas de los individuos dividido para 5, que es el máximo valor de la escala con el cual se obtiene el valor de usuarios satisfechos, al final se obtiene el promedio de las 4 preguntas.

$$\text{Pregunta} = \text{sumaRespuestas} / 5 = \text{satisfechos.}$$

Tabla 21: Resultados de la Encuesta.

Pregunta	Suma de Pregunta	Máximo Valor	Satisfechos
Pregunta 2	14	5	2,8
Pregunta 6	11	5	2,2
Pregunta 8	24	5	4,8
Pregunta 10	9	5	1,8
Total			$(11,6 / 4) = 2,9$

Fuente: Propia.

Para obtener el valor total de los sujetos satisfechos se calculó el promedio de las preguntas.

Formula: $x = \frac{A}{B}$

Tabla 22: Métrica Utilidad.

Métrica	Utilidad	
Elementos de los Datos	Descripción	Valor
A	Número de usuarios satisfechos	2,9
B	Número total de usuarios	5

Fuente: Propia.

Remplazando en la formula se obtiene: $x = \frac{2,9}{5} = 0,58$

○ **Métrica Confianza.**

Determina el grado de confianza de los usuarios, basado en las quejas presentadas por la falla del producto.

Formula : $x = \frac{A}{B}$

: $C = 1 - X$

Tabla 23: Métrica Confianza.

Métrica	Confianza	
Elementos de los Datos	Descripción	Valor
A	Número de quejas presentadas	0
B	Número total de usuarios	5

Fuente: Propia.

Remplazando en la formula se obtiene: $x = \frac{0}{5} = 0$

$C = 1 - 0 = 1$

○ **Métrica Comodidad.**

Determina el grado de facilidad y poco esfuerzo en el uso del producto y se la obtiene por medio del cálculo de los valores ponderados de la escala de respuesta de las preguntas de la encuesta asociadas a esta subcaracterística, preguntas elegidas 3 y 5. Para obtener el valor de cada elemento de dato, se multiplica la ponderación por el total de respuestas y se divide para 10 que es el total de usuarios de las dos preguntas.

$$\text{Formula : } x = A + B + C + D + E$$

Tabla 24: Métrica Comodidad

Métrica	Comodidad			
Elementos de los Datos	Descripción	Ponderación	Total de Respuesta	Valor
A	Muy de acuerdo	1	6	0,6
B	Algo de acuerdo	0.8	4	0,32
C	Ni de acuerdo ni en desacuerdo	0.6	0	0
D	Algo desacuerdo	0.4	0	0
E	Muy en desacuerdo	0.2	0	0

Fuente: Propia.

$$x = 0,6 + 0,32 + 0 + 0 + 0 = 0,92$$

3.1.4 Resultado de Evaluación.

Después de la validación de calidad a la aplicación, se obtuvieron los resultados como se observa en la figura xx se tiene un resumen el nivel y porcentaje de importancia que se asignó a la característica que se evaluó, la calificación final que se consiguió por métricas de evaluación y la calidad total de la aplicación.

Tabla 25: Resultados de Evaluación.

Característica	Sub Característica	Métrica	Peso	Medición	Resultado	Total
Satisfacción	Utilidad	Nivel de Satisfacción	30%	0,58	17,40 %	77 %
		Confianza	40%	0,8	32 %	
		Comodidad	30%	0,92	27,6 %	

Fuente: Propia.

RESULTADOS DE EVALUACIÓN DE CALIDAD EN USO							
	Características	Valor Parcial Total (/10)	Nivel de Importancia	Porcentaje de Importancia	Valor Final	Calidad Parcial del Sistema (/10)	Calidad Total del Sistema (/10)
CALIDAD EN USO	Efectividad	0,00	0	0%	0,00	7,7	7,7
	Eficiencia	0,00	0	0%	0,00		
	Satisfacción	7,36	Alta	100%	0,77		
	Libertad de Riesgo	0,00	0	0%	0,00		
	Cobertura de contexto	0,00	0	0%	0,00		

Fig. 44: Resultado de Evaluación de Calidad en uso
Fuente: Adaptada (Vaca, 2017)

3.2 Análisis de Interpretación de Resultados.

Para analizar los resultados, se tomó en cuenta la escala de evaluación descrita en la norma ISO/IEC 25040, la cual dice que la evaluación de un producto es realizada según el contexto en el que se desarrolla, por lo cual, la Fig 45 indica los niveles de puntuación para las métricas. En el caso de la evaluación realizada, todas las métricas están con la valoración de satisfactorio. (Suryotrisongko et al., 2017) (ISO/IEC-25040, 2011)



Fig. 45: Niveles de Puntuación.
Fuente: Adaptada de (ISO/IEC-25040, 2011)

Una vez concluido la validación de los resultados obtenidos se muestra que la aplicación realizada cumple satisfactoriamente los parámetros que brinda la norma ISO/IEC 25010. De acuerdo a la tabla, sobre el resultado de la característica evaluada en la aplicación se puede deducir lo siguiente:

El aplicativo desarrollado para el control y gestión de procesos de papelería San Antonio cumple el 77 % en satisfacción al usuario, lo que quiere decir está dentro del rango Muy Bueno, llegando a ser aceptable y pueden usarlo sin asistencia de un experto, cumple correctamente con los objetivos planteados. Además de tener una eficiencia aceptable en comparación con un usuario experto.

3.3 Análisis de Impacto.

En el presente trabajo se ha visto oportuno realizar un análisis de impactos en distintas áreas como: ecológico, económico y tecnológico, en la Tabla, se plantea la calificación según el impacto positivo o negativo, los cuales serán valorados de la siguiente manera:

Tabla 26 : Calificación de impacto.

Valor	Impacto
-3	Impacto alto Negativo
-2	Impacto medio Negativo
-1	Impacto bajo Negativo
0	No hay Impacto
1	Impacto bajo Positivo
2	Impacto medio Positivo
3	Impacto alto Positivo

Fuente: (Posso, 2011).

Se realizará una matriz que contendrá de manera horizontal el nivel de impacto, mientras que de manera vertical se establece una serie de indicadores que detallarán información específica del área analizada, para obtener información que permita matemáticamente saber cuál es el nivel de impacto en el proyecto. Los valores se han considerado tomando en cuenta la información recopilada del aplicativo.

3.3.1. Impacto Económico.

Tabla 27: Impacto Económico

Indicador	Nivel de Impacto								Total
	-3	-2	-1	0	1	2	3		
Incremento en Ventas							x		3
Incremento productividad							x		3
Total							6		6

Fuente: Propia.

Suma del valor de impacto económico: 6/2

Total, del nivel de impacto económico: 3

Nivel de Impacto económico: Alto positivo.

Análisis.

El impacto económico es muy importante dentro de la realización de un trabajo, en este caso el aplicativo cumple con las expectativas de generar más ingresos económicos a la micro empresa al realizar sus procesos de forma ágil y ordenada.

3.3.2 Impactos Ecológico

Tabla 28: Impacto Ecológico.

Indicador \ Nivel de Impacto	-3	-2	-1	0	1	2	3	Total
Reducción del uso de papel						x		2
Mejora del medio Ambiente						x		2
Total						4		4

Fuente: Propia.

Suma del valor de impacto ecológico: 4/2

Total, del nivel de impacto ecológico: 2

Nivel de Impacto ecológico: Medio positivo.

Análisis.

El impacto ambiental del proyecto incidirá, a corto, mediano y largo plazo, es necesario reducir el uso de papel en impresiones de comprobantes. Por otro lado, en cuanto al indicador de mejora del medio ambiente no tiene un impacto tan positivo debido a que al utilizar el aplicativo es necesario un aparato electrónico, éste necesita que la batería esté cargada, lo que resulta no tan beneficioso para el ambiente el uso constante de energía eléctrica.

3.3.3. Impacto Tecnológico.

Indicador \ Nivel de Impacto	-3	-2	-1	0	1	2	3	Total
Uso de nueva tecnología							x	3
Transferencia tecnológica							x	3
Total							6	6

Fuente: Propia.

Suma del valor de impacto tecnológico: 6/2

Total, del nivel de impacto tecnológico: 3

Nivel de Impacto tecnológico: Alto positivo.

Análisis.

El impacto tecnológico es un aspecto muy importante en la actualidad, y para la realización del proyecto se empleó el uso de una tecnología que con el paso del tiempo ha ido evolucionando de manera positiva.

CONCLUSIONES.

Se comprobó que los procesos realizados por la micro empresa Bazar y Papelería San Antonio, permiten llevar un mejor manejo de proceso, ya que con el ingreso de la información y los reportes generados se conoce el estado del producto, generando mayor conformidad a los trabajadores.

Las herramientas tecnológicas definidas para el entorno de desarrollo de la aplicación web Framework Spring Boot como Backend y Framework Angular como Frontend permitió realizar el trabajo de una manera ágil, debido que ofrecen una variedad de complementos que pueden ser instalados y aplicados fácilmente.

La metodología ágil Scrum, facilitó el desarrollo e implantación del proyecto, gracias a la forma rápida y estructurada de los Sprints, obteniendo como resultado final un aplicativo de calidad que cumple con los requerimientos y tiempos establecidos por el cliente.

Con la aplicación de la norma ISO/IEC 25010 se evaluó la calidad en uso del sistema web mediante la característica de satisfacción y sub característica de utilidad obteniendo como resultado un puntaje de 77 %, garantizando que los usuarios se sienten muy satisfecho con el software, ya cuenta con todos los requisitos acordados al inicio del proyecto.

RECOMENDACIONES.

Emplear sistemas informáticos en todas las empresa o negocios para eliminar los procesos manuales y el uso excesivo de papel. Gracias al uso de estos sistemas se puede realizar las mismas tareas de una forma ágil y sin pérdida de registros, esto es muy beneficioso al momento de optimizar el tiempo y los recursos de una empresa.

Se recomienda mantener una actualización de revisiones correspondiente a los Frameworks Spring Boot y Angular para el correcto funcionamiento de los mismo, al ser software libre presentan actualizaciones constantes para mejorar el rendimiento y optimización al momento del desarrollo de una aplicación.

Aplicar roles de test en la metodología Scrum permitió facilitar el cumplimiento de los requerimientos, es importante tomar en cuenta todos los requerimientos establecidos para evitar el retraso de entrega de cada Sprint.

Se recomienda aplicar una o más características que proporciona la ISO 25010, ya que la aplicación de un estándar internacional en un producto software garantiza la calidad del mismo y por consiguiente la satisfacción del usuario final.

REFERENCIAS BIBLIOGRÁFICAS.

- Bruno, L. (2019). Spring Data. *Journal of Chemical Information and Modeling*, 53(9), 1689–1699. <https://doi.org/10.1017/CBO9781107415324.004>
- Campoverde, U. G. A. (2019). ESTUDIO DE LA HERRAMIENTA MAVEN COMO GESTOR DE PROYECTOS SPRING MVC CON EL CASO DE USO APLICACIÓN PARA COMERCIO ELECTRÓNICO.
- Cardador, C. A. L. (2014). Implantación de aplicaciones web en entornos internet, intranet y extranet. Retrieved May 4, 2020, from https://books.google.com.ec/books?id=Lj91CQAAQBAJ&pg=PT4&source=gbs_toc_r&cad=3#v=onepage&q=%09REST&f=false
- César, B. A. (2018). DESARROLLO DE UNA PLATAFORMA WEB BAJO EL FRAMEWORK SPRING DE JAVA PARA LABORATORIOS VIRTUALES. Universidad de Piura.
- Crusoveanu, L. (2019). Inversion of Control and Dependency Injection with Spring. Retrieved December 14, 2019, from 16 de octubre de 2019 website: <https://www.baeldung.com/inversion-control-and-dependency-injection-in-spring>
- Equipo de desarrollo de SOFTENG, C. (2013). Metodología SCRUM para desarrollo de software a medida. *Metodologías Para El Desarrollo de Software*. Retrieved from <https://www.softeng.es/es-es/empresa/metodologias-de-trabajo/metodologia-scrum.html>
- Esquivel, E. A. (2016). Aplicación web de gestión de una academia con Spring Framework. *Universidad de Sevilla*.
- Figueroa, C. W. E. (2018). SISTEMA DE FACTURACIÓN UTILIZANDO LOS FRAMEWORK ANGULAR 5, LARAVEL 5.5 PARA LA JUNTA ADMINISTRADORA DE AGUA POTABLE SAN JOSÉ DE CHORLAVÍ DE LA CIUDAD DE IBARRA – ECUADOR. <https://doi.org/10.1051/matecconf/201712107005>
- Flex, A. (2018). *Marco de Desarrollo de la Junta de Andalucía Accesibilidad para contenidos multimedia*. 1–2.
- Gajewski, M., & Zabierowski, W. (2019). Analysis and comparison of the spring framework and play framework performance, used to create web applications in Java. *2019 IEEE 15th International Conference on the Perspective Technologies and Methods in MEMS Design, MEMSTECH 2019 - Proceedings*, 170–173. <https://doi.org/10.1109/MEMSTECH.2019.8817390>
- Gómez, A. C. (2016). *Gestión de Proyectos del Servicio Integrado de Empleo de la UPV*. Universidad Politécnica de Valencia.
- Gudiño, Q. A. R. (2018). ESTUDIO DE INTEGRACIÓN DE LOS FRAMEWORKS ANGULAR 4 Y YII2 , ORIENTADO A SERVICIOS REST , QUE PERMITAN LA GESTIÓN Y CONTROL DE INVENTARIOS PARA MEJORAR LA PRODUCTIVIDAD EN LA EMPRESA INDUXION.
- Gutiérrez, M. C. E., Verduzco, R. J. A., & Farías, M. N. (2015). Inyección de Dependencias en el Lenguaje de Programación Go. *RIDE Revista Iberoamericana Para La Investigación y El Desarrollo Educativo*, 5. Retrieved from <http://www.redalyc.org/articulo.oa?id=498150318017>
- Ibáñez, F. M., & Lévano, J. A. (2016). DISEÑO E IMPLEMENTACIÓN DE UNA HERRAMIENTA DE VISUALIZACIÓN PARA ANÁLISIS EN TIEMPO REAL DE REDES

SDN/OPENFLOW.

- ISO/IEC-25010. (2015). Iso 25010. Retrieved May 14, 2020, from Iso 25000 website: <https://iso25000.com/index.php/normas-iso-25000/iso-25010>
- ISO/IEC-25010. (2019). Iso 25010. *Iso25000.Com*, 1–7. Retrieved from <https://iso25000.com/index.php/normas-iso-25000/iso-25010>
- ISO/IEC-25040. (2011). ISO/IEC 25040:2011(en), Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — Evaluation process. Retrieved May 14, 2020, from 2011 website: <https://www.iso.org/obp/ui/#iso:std:iso-iec:25040:ed-1:v1:en>
- MAFLA, S. M. F. (2019). *COMPARATIVA DE LOS FRAMEWORKS ANGULAR Y PRIMEFACES PARA EL DESARROLLO DEL APLICATIVO CONTROL DE MATERIA PRIMA EN LA EMPRESA MASTERCUBOX S.A., UTILIZANDO LA METODOLOGÍA SCRUM*”.
- Mak, G., & Mak, G. (2018). Spring MVC Framework. *Spring Recipes*, 321–393. https://doi.org/10.1007/978-1-4302-0623-1_10
- Martín, E. C. J. (2018). SportMeet: Aplicación Web para la gestión de partidas deportivas usando Spring. Universidad de Sevilla.
- Microsoft. (2020). TypeScript - JavaScript that scales. Retrieved May 4, 2020, from <https://www.typescriptlang.org/>
- Naciones Unidas/CEPAL. (2019). La Agenda 2030 y los Objetivos de Desarrollo Sostenible: una oportunidad para América Latina y el Caribe. Objetivos, metas e indicadores mundiales. In *Publicación de las Naciones Unidas*. Retrieved from https://repositorio.cepal.org/bitstream/handle/11362/40155/24/S1801141_es.pdf
- Navarro, A. C., Fernández, J. D. M., & Morales, J. V. (2013). Revisión de metodologías ágiles para el desarrollo de software. *PROSPECTIVA*, 11, 30–39. <https://doi.org/10.1097/GCO.0b013e3283630e10>
- Ottinger, J. B., & Lombardi, A. (2019). Spring Boot. In *Beginning Spring 5* (pp. 169–205). https://doi.org/10.1007/978-1-4842-4486-9_7
- Pérez, M. E. (2015). Desarrollo de aplicaciones mediante framework de spring. In *Sextil Online LLC*. Retrieved from https://books.google.com.ec/books?hl=es&lr=&id=UCpADwAAQBAJ&oi=fnd&pg=PT12&dq=CARACTERISTICAS+spring+framework&ots=DV3w0lzn6p&sig=F4JdpCeOfEau7KYez7KQh4jqVZ8&redir_esc=y#v=onepage&q=CARACTERISTICAS spring framework&f=false
- Pérez Martínez, E., & Altadill Izura, P. X. (2018). *Spring 5*. Anaya Multimedia.
- Posso, M. Á. (2011). *Metodología para el trabajo de Grado* (1st ed.; C. E. del L.-N. de Pichincha, Ed.).
- Proaño, Y. C. F. (2019). DESARROLLO DE UN SISTEMA WEB PARA EL CONTROL DE LA PRODUCCIÓN DE UN INVERNADERO DE TOMATE EN EL CANTÓN COTACACHI, UTILIZANDO EL FRAMEWORK ANGULAR VERSIÓN 4. *Journal of Chemical Information and Modeling*, 53(9), 1689–1699. <https://doi.org/10.1017/CBO9781107415324.004>
- Reyes, E. (2017). *LÍNEA DE INVESTIGACIÓN: Aplicación de normas y estándares en un proceso de desarrollo de software*.
- Sanz, L. F., & Silva, P. B. (2014). Gestión de riesgos en proyectos de desarrollo de software en España: Estudio de la situación. Retrieved December 14, 2019, from Revista Facultad

- de Ingenieria website: <https://www.redalyc.org/articulo.oa?id=43030033021>
- Schwaber, Ken; Sutherland, J. (2013). *La Guía de Scrum*.
- Suryotrisongko, H., Jayanto, D. P., & Tjahyanto, A. (2017). Design and Development of Backend Application for Public Complaint Systems Using Microservice Spring Boot. *Procedia Computer Science*, 124, 736–743. <https://doi.org/10.1016/j.procs.2017.12.212>
- Vaca, T. N. (2017). MODELO DE CALIDAD DE SOFTWARE APLICADO AL MÓDULO DE TALENTO HUMANO DEL SISTEMA INFORMÁTICO INTEGRADO UNIVERSITARIO – UTN. In *Universidad Técnica del Norte*.
- Valencia, A. D. G. (2018). *Análisis de frameworks de desarrollo de api rest y su impacto en el rendimiento de aplicaciones web con arquitectura Spa*. Retrieved from <http://repositorio.utn.edu.ec/handle/123456789/8264>
- Valverde Ramos, E., & Mora de Fuentes, P. H. (2014). *Introducción a TypeScript*. Retrieved from <http://fanta.56k.es/libros/tecnicos/Manual-TypeScript.pdf>
- Webb, P., Syer, D., Long, J., Nicoll, S., Winch, R., Wilkinson, A., ... Documentation, I. S. B. (2018). *Spring Boot Reference Guide Table of Contents*.

ANEXOS

Anexo A: Encuesta SUS.

1.- ¿Considera usted qué usaría este sistema frecuentemente?

- Muy de acuerdo
- Algo acuerdo
- Ni de acuerdo ni en desacuerdo
- Algo desacuerdo
- Muy en desacuerdo.

2.- ¿Considera usted qué el sistema fue fácil de usar?

- Muy de acuerdo
- Algo acuerdo
- Ni de acuerdo ni en desacuerdo
- Algo desacuerdo
- Muy en desacuerdo.

3.- ¿Considera usted qué el sistema fue fácil de usar?

- Muy de acuerdo
- Algo acuerdo
- Ni de acuerdo ni en desacuerdo
- Algo desacuerdo
- Muy en desacuerdo.

4.- ¿Considera usted qué necesitaría ayuda de una persona con conocimientos técnicos para usar este sistema?

- Muy de acuerdo
- Algo acuerdo
- Ni de acuerdo ni en desacuerdo
- Algo desacuerdo
- Muy en desacuerdo.

5.- ¿Considera usted qué las funciones de este sistema están bien integradas?

- Muy de acuerdo
- Algo acuerdo
- Ni de acuerdo ni en desacuerdo
- Algo desacuerdo
- Muy en desacuerdo.

6.- ¿Considera usted qué el sistema es inconsistente?

- Muy de acuerdo
- Algo acuerdo
- Ni de acuerdo ni en desacuerdo
- Algo desacuerdo
- Muy en desacuerdo.

7.- ¿Considera usted que la mayoría de la gente aprendería a usar este sistema en forma rápida?

- Muy de acuerdo
- Algo acuerdo
- Ni de acuerdo ni en desacuerdo
- Algo desacuerdo
- Muy en desacuerdo.

8.- ¿Considera usted que el sistema es difícil de usar?

- Muy de acuerdo
- Algo acuerdo
- Ni de acuerdo ni en desacuerdo
- Algo desacuerdo
- Muy en desacuerdo.

9.- ¿Se siente confiado al usar el sistema?

- Muy de acuerdo
- Algo acuerdo
- Ni de acuerdo ni en desacuerdo
- Algo desacuerdo
- Muy en desacuerdo.

10.- ¿Considera usted que necesita aprender muchas cosas tecnológicas antes de usar el sistema?

- Muy de acuerdo
- Algo acuerdo
- Ni de acuerdo ni en desacuerdo
- Algo desacuerdo
- Muy en desacuerdo.
-

Anexo B: Matriz de Evaluación de calidad en Uso

EVALUACIÓN DE CALIDAD EN USO																		
Características	Subcaracterísticas	Métrica	Propósito-métrica	Método de aplicación	Fase ciclo de vida de calidad del producto	Fórmula / Variables	Peor caso	Valor Deseado	Aplica	Variables			Valor Obtenido	Valor Métrica / 10	Final Subcaracterística	Total Característica	Final Característica	Calidad de Uso del Sistema
										A	B	T	X					
Satisfacción	Utilidad	Uso discrecional de las funciones	¿Qué tan satisfecho está el usuario?	Realizar un cuestionario sobre el nivel de satisfacción sobre el sistema.	Uso	$X = A/B$ A= Número de preguntas con respuesta satisfactorias B = Número total de preguntas realizadas en el cuestionario. Dónde: $B > 0$	0	1	SI	2,9	5		0,58	5,80	7,70	7,70	7,70	7,70
		Porcentaje de quejas de los clientes	¿Qué porcentaje de los usuarios optan por utilizar las funciones sistema?	Observación de uso	Uso	$X = A/B$ A= Número de funciones específicas del software que se utilizan B= Número total de funciones que están destinados a ser usados Dónde: $B > 0$	0	1	SI	4	5		0,80	8,00				
		Retorno de la Inversión (ROI)	¿Cuáles el porcentaje de quejas realizadas por los clientes?	Contar el número de clientes que se quejan y contar el número total de clientes	Uso	$X = A/B$ A = Número de clientes que se quejan B= Número total de clientes Dónde: $B > 0$	1	0	SI	1	5		0,20	9,20				

Fig. 46:Matriz de Evaluación de Calidad en Uso.

Fuente: Adaptada (Vaca, 2017)