

UNIVERSIDAD CATÓLICA SANTO TORIBIO DE MOGROVEJO
FACULTAD DE INGENIERÍA
ESCUELA DE INGENIERÍA DE SISTEMAS Y COMPUTACIÓN



**APLICATIVO MÓVIL CON TECNOLOGÍA ANDROID UTILIZANDO
UN MÉTODO DE COSTEO DE INVENTARIO PARA APOYAR EL
PROCESO DE VENTAS EN LA EMPRESA DICOPERSAC CHICLAYO**

**TESIS PARA OPTAR EL TÍTULO DE
INGENIERO DE SISTEMAS Y COMPUTACIÓN**

**AUTOR
CARLOS ALBERTO EFFIO GONZALES**

**ASESOR
HUILDER JUANITO MERA MONTENEGRO**

<https://orcid.org/0000-0001-6830-5415>

Chiclayo, 2022

**APLICATIVO MÓVIL CON TECNOLOGÍA ANDROID
UTILIZANDO UN MÉTODO DE COSTEO DE INVENTARIO
PARA APOYAR EL PROCESO DE VENTAS EN LA EMPRESA
DICOPERSAC CHICLAYO**

PRESENTADA POR:
CARLOS ALBERTO EFFIO GONZALES

A la Facultad de Ingeniería de la
Universidad Católica Santo Toribio de Mogrovejo
para optar el Título de

INGENIERO DE SISTEMAS Y COMPUTACIÓN

APROBADA POR:

María Ysabel Aranguri Garcia
PRESIDENTE

Zelada Valdevieso Hector Miguel
SECRETARIO

Huiler Juanito Mera Montenegro
VOCAL

DEDICATORIA

El presente trabajo está dedicado a mi esposa Lady e hijas Kristell y Aslesh por ser mi apoyo a lo largo de mi viaje universitario y durante toda mi vida y para mis padres.

A Dios padre, amigos y familiares que contribuyeron en la confianza para poder alcanzar mis logros y metas en esta etapa de formación profesional.

AGRADECIMIENTOS

En primer lugar a Dios, a mi familia que me han permitido cumplir de manera satisfactoria cada etapa profesional.

A mis asesores y profesores, quienes me han guiado en el complicado proceso que no ha sido nada fácil, pero gracias a su ayuda he podido lograr con mis objetivos de ser profesional.

También quiero agradecer a los directivos y plana docente de la prestigiosa universidad por la orientación y guía brindada.

ÍNDICE

RESUMEN.....	11
ABSTRACT.....	12
INTRODUCCIÓN	13
I. MARCO TEÓRICO.....	16
1.1. ANTECEDENTES	16
1.1.1. ANTECEDENTES INTERNACIONALES	16
1.1.2. ANTECEDENTES NACIONALES	17
1.1.3. ANTECEDENTES LOCALES.....	19
1.2. BASES TEÓRICO CIENTÍFICAS	21
1.2.1. INVENTARIO.....	21
1.2.2. ANÁLISIS DE INVENTARIOS	21
1.2.3. DETERMINACIÓN DEL PUNTO DE REORDEN.....	23
1.2.4. EXISTENCIAS DE RESERVA O SEGURIDAD DE INVENTARIOS.....	24
1.2.5. CONTROL DE INVENTARIOS JUSTO A TIEMPO	25
1.2.6. MÉTODOS DE EVALUACIÓN DE INVENTARIOS	26
1.2.7. MÉT.ODOS DE EVALUACIÓN DE INVENTARIOS	26
1.2.7. SUSTENTO DE LA ELECCIÓN DEL MÉTODO DE COSTEO.....	26
II. METODOLOGÍA	29
2.1. TIPO Y NIVEL DE INVESTIGACIÓN	29
2.1.1. TIPO DE INVESTIGACIÓN.....	29
2.1.2. DISEÑO Y NIVEL DE INVESTIGACIÓN.....	29
2.2. POBLACIÓN, MUESTRA Y MUESTREO	30
2.2.1. POBLACIÓN	30
2.2.2. MUESTRA	30
2.2.3. MUESTREO	31
2.3. CRITERIOS DE SELECCIÓN.....	31
2.4. OPERACIONALIZACIÓN DE VARIABLES	31
2.4.1. VAR IABLES	31
2.4.2. ÍN DICAD ORES (OPER ACIO NALIZA CIÓN DE VAR IA BLES)	32
2.5. TÉCNICAS E INSTRUMENTOS DE RECOLECCIÓN DE DATOS.....	33

2.6.	PROCEDIMIENTOS	33
2.6.1.	METODOLOGÍA DE DESARROLLO	33
2.6.2.	ANÁLISIS DE RIESGOS	34
2.6.3.	PRODUCTO ACREDITABLE	35
2.6.4.	MANUAL DE USUARIO.....	35
2.7.	PLAN DE PROCESAMIENTO Y ANÁLISIS DE DATOS	35
2.8.	MATRIZ DE CONSISTENCIA	36
2.9.	CONSIDERACIONES ÉTICAS	37
III.	RESULTADOS	37
3.1.	EN BASE A LA METODOLOGÍA UTILIZADA.....	37
3.1.1.	ITERACIÓN #1: PLANIFICACIÓN DEL PROYECTO	37
3.1.2.	ITERACIÓN #2: ANÁLISIS PREL DE REQ– MOD DE NEGOCIO	38
3.1.3.	ITERACIÓN # 3: ANÁLISIS PRELIMINAR DE REQ.	57
3.1.4.	ITERACIÓN # 4: ANÁLISIS.....	68
3.1.5.	ITERACIÓN # 5: DISEÑO	73
3.2.	EN BASE A LOS OBJETIVOS DE LA INVESTIGACIÓN.....	103
3.2.1.	DETERMINAR EL MÉTODO DE COSTEO DE INVENTARIO QUE SE APLICARÁ EN EL DESARROLLO DEL SOFTWARE	103
3.2.2.	DESARROLLAR LA APLICACIÓN MÓVIL SEGÚN LOS REQUERIMIENTOS DEFINIDOS Y EL MÉTODO DE COSTEO SELECCIONADO	103
3.2.3.	VALIDAR LA FUNCIONALIDAD DE LA APLICACIÓN MÓVIL, SEGÚN LOS REQUERIMIENTOS DEFINIDOS CON LOS USUARIOS INTERESADOS.	104
IV.	CONCLUSIONES	106
V.	RECOMENDACIONES	107
VI.	REFERENCIAS.....	108
VII.	ANEXOS.....	110

LISTA DE TABLAS

T A B L A 1 : MÉT.ODOS DE EVALUACIÓN DE INVENTARIO	26
TABLA 2: CONTRATACIÓN DE HIPÓTESIS	29
TA BL A 3: TÉCNICAS E INSTRUMENTOS DE RECOLECCIÓN DE DAT.OS.....	33
TABLA 4: MA-TRIZ DE CONSISTENCIA	36
TABLA 5: CRITERIOS ÉTICOS.....	37
TA~BLA 6: IDENTIFICACIÓN DE REQUERIMIENTO FUNCIONAL INICIAR SESIÓN	38
TA~BLA 7: IDENTIFICACIÓN DE REQUERIMIENTO FUNCIONAL MENÚ PRINCIPAL.....	38
TA ~BLA 8: IDENTIFICACIÓN DE RE QUE RI MIENTO FU NCIONAL MI S PED IDO	39
TA~BLA 9: IDENTIFICACIÓN DE REQUERIMIENTO FUNCIONAL REGISTRAR CLIENTE	39
T A~BLA 10: IDENTIFICACIÓN DE REQUERIMIENTO FUNCIONAL REGISTRAR PRODUCTO	39
TA~BLA 11: IDENTIFICACIÓN DE REQUERIMIENTO FUNCIONAL REGISTRAR CATEGORÍA DE PRODUCTO	40
TA~BLA 12: IDENTIFICACIÓN DE REQUERIMIENTO FUNCIONAL REGISTRAR MARCA DE PRODUCTO	40
TA~BLA 13: IDENTIFICACIÓN DE REQUERIMIENTO FUNCIONAL DEVOLUCIONES DE COMPRAS (FACTURA-BOLETAS)	40
TA~BLA 14: IDENTIFICACIÓN DE REQUERIMIENTO FUNCIONAL COMPRA REGISTRADAS	41
TA. ~BLA 15: IDENTIFICACIÓN DE REQUERIMIENTO FUNCIONAL VENTAS REGISTRADAS	41
TA. ~BLA 16: IDENTIFICACIÓN DE REQUERIMIENTO FUNCIONAL KARDEX	41
TA~BLA 17: IDENTIFICACIÓN DE RE.QUERIMIENTO NO FUN.CIONAL DE COM.PATIBILIDAD DE S.O.....	42
TA~BLA 18: IDENTIFICACI.ÓN DE REQUERIMIENTO NO FUNCIONAL DE COMPATIBILIDAD DE RESOLUCIÓN	42
TA~BLA 19: IDENTIFICACIÓN DE REQUERIMIENTO NO FUNCIONAL DE INTERFAZ SENCILLA E INTUITIVA	43
TABLA 20: MÓDULO INICIAR SESIÓN VENTAS	58
TA~BLA 21: MÓDULO REGISTRAR NUEVO CLIENTE.....	58
TABLA 22: MÓDULO CONSULTAR PRODUCTO DE VENDEDOR.....	59
TABLA 23: MÓDULO REALIZAR GESTIÓN DE PEDIDO	59
TABLA 24: MÓDULO INICIAR SESIÓN CONTROL INTERNO	60
TA~BLA 25: MÓD.ULO REGISTRAR COMPRA	60
TA~BLA 26: MÓDULO REGISTRAR VENTA.....	60
TABLA 27: MÓDULO ADMINISTRAR KARDEX	61

LISTA DE FIGURAS

FIG. 1 NIVEL ÓPTIMO D.E INVENTARIO	22
FIG. 2 INTERESES DEL NIVEL ÓPTIMO DE INVENTARIO	23
FIG. 3 ARQUITECTURA ANDROID	28
FIG. 4: DIAGRAMA DE CONTEXTO DEL NEGOCIO	43
FIG. 5: DIAGRAMA DE CASO DE USO DE NEGOCIO DE COMPRAS.....	44
FIG. 6: DIAGRAMA DE CASO DE USO DE NEGOCIO DE VENTAS	44
FIG. 7: DIAGRAMA DE CASO DE USO DE NEGOCIO DE ALMACÉN.....	45
FIG. 8: DIAGRAMA DE ACTIVIDADES DE VERIFICAR STOCK DE COMPRAS	46
FIG. 9: DIAGRAMA DE ACTIVIDADES DE SOLICITAR PRODUCTOS DE COMPRAS	46
FIG. 10: DIAGRAMA DE ACTIVIDADES DE REGISTRAR COMPRA.....	47
FIG. 11: DIAGRAMA DE ACTIVIDADES DE REALIZAR DEVOLUCIÓN.....	47
FIG. 12: DIAGRAMA DE ACTIVIDADES DE REGISTRAR CLIENTE DE VENTAS.....	48
FIG. 13: DIAGRAMA DE ACTIVIDADES DE CONSULTAR PRODUCTO DE VENTAS	48
FIG. 14: DIAGRAMA DE ACTIVIDADES DE REGISTRAR VENTA DE VENTAS.....	49
FIG. 15: DIAGRAMA DE ACTIVIDADES DE PAGAR VENTA DE VENTAS.....	49
FIG. 16: DIAGRAMA DE ACTIVIDADES DE REQUERIMIENTO DE PEDIDOS DE ALMACÉN	50
FIG. 17: DIAGRAMA DE ACTIVIDADES DE REQUERIMIENTO DE INVENTARIO	50
FIG. 18: DIAGRAMA DE OBJETOS DE VERIFICAR STOCK.....	51
FIG. 19: DIAGRAMA DE OBJETOS PARCIAL PRODUCTO DE VERIFICAR STOCK	51
FIG. 20: DIAGRAMA DE OBJETOS DE SOLICITAR PRODUCTOS	51
FIG. 21: DIAGRAMA DE OBJETOS PARCIAL DE SOLICITAR PRODUCTOS.....	52
FIG. 22: DIAGRAMA DE OBJETOS DE VERIFICAR Y PAGAR PRODUCTO DE COMPRAS Y DEVOLUCIÓN...	52
FIG. 23: DIAGRAMA DE OBJETOS PARCIAL DE VERIFICAR Y PAGAR PRODUCTO DE COMPRAS Y DEVOLUCIÓN	52
FIG. 24: DIAGRAMA DE OBJETOS PARCIAL DE REGISTRAR CLIENTE	53
FIG. 25: DIAGRAMA DE OBJETOS PARCIAL DE REGISTRAR CLIENTE	53
FIG. 26: DIAGRAMA DE OBJETOS DE CONSULTAR PRODUCTO	53
FIG. 27 DIAGRAMA DE OBJETOS PARCIAL DE CONSULTAR PRODUCTO DE VENTAS	53
FIG. 28: DIAGRAMA DE OBJETOS DE REGISTRAR VENTAS	54
FIG. 29: DIAGRAMA DE OBJETOS PARCIAL DE REGISTRAR VENTA.....	54
FIG. 30: DIAGRAMA DE OBJETOS DE PAGAR VENTAS	54
FIG. 31: DIAGRAMA DE OBJETOS PARCIAL DE PAGAR VENTAS.....	55
FIG. 32: DIAGRAMA DE OBJETOS DE REQUERIMIENTO DE PEDIDOS.....	55
FIG. 33: DIAGRAMA DE OBJETOS PARCIAL DE REQUERIMIENTO DE PEDIDOS	55
FIG. 34: DIAGRAMA DE OBJETOS DE INVENTARIO.	56
FIG. 35: DIAGRAMA DE OBJETO PARCIAL DE INVENTARIO.....	56
FIG. 36: DIAGRAMA DE OBJETOS GENERAL DE COMPRAS.....	56
FIG. 37: DIAGRAMA DE OBJETOS GENERAL DE VENTAS	57
FIG. 38: DIAGRAMA DE OBJETOS GENERAL DE ALMACÉN.....	57
FIG. 39: DIAGRAMA DE CONTEXTO DE REQUERIMIENTOS.....	57
FIG. 40: DIAGRAMA DE CASO DE USO DE TOMA PEDIDOS.	58
FIG. 41: DIAGRAMA DE CASO DE USO DE CONTROL INTERNO.....	59
FIG. 42: DIAGRAMA DE ACTIVIDADES DE INICIAR SESIÓN.....	61
FIG. 43: DIAGRAMA DE ACTIVIDADES DE REGISTRAR CLIENTE DE TOMA PEDIDOS	62
FIG. 44: DIAGRAMA DE ACTIVIDADES DE CONSULTAR PRODUCTO DE TOMA PEDIDOS	62
FIG. 45: DIAGRAMA DE ACTIVIDADES DE VENTA DE GESTIONAR VENTA.....	63
FIG. 46: DIAGRAMA DE ACTIVIDADES DE MODIFIAR VENTA DE GESTIONAR VENTA	63
FIG. 47: DIAGRAMA DE ACTIVIDADES RECHAZAR PEDIDO	64
FIG. 48: DIAGRAMA DE ACTIVIDADES DE INICIAR SESIÓN DE CONTROL INTERNO	64
FIG. 49: DIAGRAMA DE ACTIVIDADES DE REGISTRAR COMPRA DE CONTROL INTERNO.....	65
FIG. 50: DIAGRAMA DE ACTIVIDADES DE REGISTRAR PRODUCTO DE CONTROL INTERNO.....	65

FIG. 51: DIAGRAMA DE ACTIVIDADES DE KARDEX DE ADMINISTRAR KARDEX.....	66
FIG. 52: DIAGRAMA DE OBJETOS PARCIAL DE INICIAR SESIÓN DE VENTAS.....	66
FIG. 53: DIAGRAMA DE OBJETOS PARCIAL DE REGISTRAR CLIENTE DE VENTAS.....	66
FIG. 54: DIAGRAMA DE OBJETOS PARCIAL DE CONSULTAR PRODUCTO DE VENTAS	66
FIG. 55: DIAGRAMA DE OBJETOS PARCIAL DE GESTIONAR PEDIDO DE VENTAS	67
FIG. 56: DIAGRAMA DE OBJETOS PARCIAL DE INICIAR SESIÓN DE CONTROL INTERNO	67
FIG. 57: DIAGRAMA DE OBJETOS PARCIAL DE REGISTRAR COMPRA DE CONTROL INTERNO	67
FIG. 58: DIAGRAMA DE OBJETOS PARCIAL DE REGISTRAR PRODUCTO DE CONTROL INTERNO.	67
FIG. 59: DIAGRAMA DE OBJETOS PARCIAL DE ADMINISTRAR KARDEX DE CONTROL INTERNO.	67
FIG. 60: DIAGRAMA DE OBJETOS GENERAL DE VENTA.	68
FIG. 61: DIAGRAMA DE OBJETOS GENERAL DE CONTROL INTERNO	68
FIG. 62: DIAGRAMA DE PAQUETES DE ANÁLISIS.....	68
FIG. 63: DIAGRAMA DE CASO DE USO DE ANÁLISIS DE VENTAS.....	69
FIG. 64: DIAGRAMA DE CASO DE USO DE ANÁLISIS DE PA CONTROL INTERNO	69
FIG. 65: DIAGRAMA DE CLASES DE ANÁLISIS DE INICIO DE SESIÓN DE PAVENTAS	69
FIG. 66: DIAGRAMA DE CLASES DE ANÁLISIS DE REGISTRAR CLIENTE DE PAVENTAS	70
FIG. 67: DIAGRAMA DE CLASES DE ANÁLISIS DE CONSULTAR PRODUCTO DE PAVENTAS	70
FIG. 68: DIAGRAMA DE CLASES DE ANÁLISIS DE GESTIONAR PEDIDO DE TOMA PEDIDOS.....	70
FIG. 69: DIAGRAMA DE CLASES DE ANÁLISIS DE INICIAR SESIÓN DE PA CONTROL INTERNO	70
FIG. 70: DIAGRAMA DE CLASES DE ANÁLISIS DE CREAR REPARTO DE PA CONTROL INTERNO	71
FIG. 71: DIAGRAMA DE CLASES DE ANÁLISIS DE GESTIONAR PRODUCTO DE PA CONTROL INTERNO.....	71
FIG. 72: DIAGRAMA DE CLASES DE ANÁLISIS DE ADMINISTRAR KARDEX DE PA CONTROL INTERNO	71
FIG. 73: DIAGRAMA DE CLASES PARCIAL INICIAR SESIÓN DE VENTAS.	71
FIG. 74: DIAGRAMA DE CLASES PARCIAL GESTIONAR CLIENTE DE VENTAS.....	72
FIG. 75: DIAGRAMA DE CLASES PARCIAL CONSULTAR PRODUCTO DE VENTAS	72
FIG. 76: DIAGRAMA DE CLASES PARCIAL GESTIONAR PEDIDO DE VENTAS	72
FIG. 77: DIAGRAMA DE CLASES PARCIAL INICIAR SESIÓN DE CONTROL INTERNO.....	72
FIG. 78: DIAGRAMA DE CLASES PARCIAL GESTIONAR COMPRA DE CONTROL INTERNO	72
FIG. 79: DIAGRAMA DE CLASES PARCIAL GESTIONAR PRODUCTO DE CONTROL INTERNO.	73
FIG. 80: DIAGRAMA DE CLASES PARCIAL ADMINISTRAR KARDEX DE CONTROL INTERNO.	73
FIG. 81: DIAGRAMA DE CLASES GENERAL DE ANÁLISIS.....	73
FIG. 82: DIAGRAMA DE CONTEXTO DE SUBSISTEMA DE DISEÑO	73
FIG. 83: DIAGRAMA DE CONTEXTO DE REALIZACIÓN DE CASO DE USO DE VENTAS.	74
FIG. 84: DIAGRAMA DE CONTEXTO DE REALIZACIÓN DE CASO DE USO DE CONTROL INTERNO	74
FIG. 85: DIAGRAMA DE SECUENCIA DE INICIO DE SESIÓN DE VENTAS.....	74
FIG. 86: DIAGRAMA DE SECUENCIA DE GESTIONAR CLIENTE DE VENTA.....	75
FIG. 87: DIAGRAMA DE SECUENCIA DE CONSULTAR PRODUCTO DE VENTA.....	75
FIG. 88: DIAGRAMA DE SECUENCIA DE GESTIONAR PEDIDO DE VENTAS.....	76
FIG. 89: DIAGRAMA DE SECUENCIA DE INICIAR SESIÓN DE CONTROL INTERNO	76
FIG. 90: DIAGRAMA DE SECUENCIA DE GESTIONAR COMPRA CONTROL INTERNO	77
FIG. 91: DIAGRAMA DE SECUENCIA DE REGISTRAR PRODUCTO DE CONTROL INTERNO	77
FIG. 92: DIAGRAMA DE SECUENCIA DE ADMINISTRAR KARDEX DE CONTROL INTERNO.....	78
FIG. 93: DIAGRAMA PARCIAL DE CLASES DE DISEÑO DE INICIAR SESIÓN DE VENTA.....	78
FIG. 94: DIAGRAMA PARCIAL DE CLASES DE GESTIONAR CLIENTE DE VENTAS.....	78
FIG. 95: DIAGRAMA PARCIAL DE CLASES DE DISEÑO DE CONSULTAR PRODUCTO DE VENTAS.....	79
FIG. 96: DIAGRAMA PARCIAL DE CLASES DE GESTIONAR PEDIDO DE VENTA	79
FIG. 97: DIAGRAMA PARCIAL DE CLASES DE INICIAR SESIÓN DE CONTROL INTERNO	79
FIG. 98: DIAGRAMA PARCIAL DE CLASES DE GESTIONAR COMPRA DE CONTROL INTERNO	80
FIG. 99: DIAGRAMA DE CLASES DE DISEÑO GENERAL.....	80
FIG. 100: DIAGRAMA DE DISEÑO DE BASE DE DATOS LÓGICA.....	81
FIG. 101: DIAGRAMA DE DISEÑO DE BASE DE DATOS FÍSICO.....	81
FIG. 102: DIAGRAMA DE DISEÑO DE ESTADOS	82
FIG. 103: DIAGRAMA DE DESPLIEGUE.....	82
FIG. 104: DIAGRAMA DE COMPONENTES	82
FIG. 105: INTERFAZ DE INICIO DE SESIÓN VENTA ANDROID –.....	81
FIG. 107: INTERFAZ DE INICIO DE SESIÓN –.....	111
FIG. 108: INTERFAZ LISTA DE PROVEEDORES-REGISTRAR PROVEEDORES.....	111

FIG. 109: INTERFAZ LISTA DE CLIENTES- REGISTRAR CLIENTE – ELABORACIÓN PROPIA	111
FIG. 110: INTERFAZ LISTA DE PRODUCTOS-REGISTRAR PRODUCTOS- ELABORACIÓN PROPIA.....	112
FIG. 111: INTERFAZ LISTA DE PRODUCTOS – ELABORACIÓN PROPIA	112
FIG. 113: INTERFAZ DE DEVOLUCIONES - ELABORACIÓN PROPIA	113
FIG. 114: 1NTERFAZ DE COMPRAS REGISTRAS-ELABORACIÓN PROPIA	114
FIG. 115: 1NTERFAZ DE VENTAS REGISTRADA- ELABORACIÓN PROPIA	114
FIG. 116: 1NTERFAZ DE VENTAS REGISTRADA.....	115
FIG. 117: 1NTERFAZ DE VENTAS REGISTRADA.....	115

RESUMEN

Actualmente en la empresa regional DICOPERSAC distribuidora y comercializadora de golosinas y abarrotes de marcas comerciales como: Herbi, Adams, Hornimans, Chocolates Guré, Docile, etc., viene presentando problemas dentro de los procesos de las ventas que se realizan del día a día que luego se ven reflejados en el control de inventario (Kardex), tales como demoras en consultar stock de productos (entre 5 a 10 minutos), mal control de inventario, demora en la venta y atención del cliente, adicional a esto el mal control y seguimiento de productos vencidos o deteriorados el cual tiene pérdida de 1,520 soles semanales que anualmente llegan a oscilar entre 72,960.00 soles., su amplia cobertura en el mercado ha permitido identificar las deficiencias operativas en el inventario, por tal razón se propuso la alternativa de desarrollar un aplicativo móvil con tecnología android utilizando un método de costeo de inventario para apoyar el proceso de ventas de la empresa Dicopersac.

Este método costeo de inventario, permite dar a conocer la salida de inventarios que ingresó primero, generando así un orden más adecuado para los negocios que venden productos con fecha de vencimiento. Para lo cual se presentó la investigación que tiene como su principal objetivo en poder mejorar el proceso de ventas a través de la implementación de una aplicación móvil con tecnología android utilizando el método de costeo de inventario, teniendo como objetivos específicos definir los requerimientos según la información proporcionada por la empresa, determinar el método que se aplicará para el desarrollo del software, realizar el análisis y el diseño del sistema en función a los requerimientos definidos y el método establecido, elaborar el App Móvil según los requerimientos definidos y aplicando el método seleccionado y validar la funcionalidad acuerdo a los requerimientos definidos, logrando cumplir los objetivos de la tesis y con las pruebas pertinentes del software. Utilicé la metodología RUP para el proyecto, la cual satisface las necesidades de contexto, tiempo y desarrollo del proyecto mejorando los problemas de control de mercadería.

Palabras clave: Método de costeo, Gestión de Inventarios, Tecnología Android, Aplicativo Móvil.

ABSTRACT

Currently in the regional company DICOPERSAC, distributor and marketer of sweets and groceries of commercial brands such as: Herbi, Adams, Hornimans, Chocolates Guré, Docile, etc., it has been presenting problems within the sales processes that are carried out on a daily basis and in inventory control (Kardex), such as delays in checking stock of products (between 5 to 10 minutes), poor inventory control, delay in sales and customer service, in addition to this poor control and monitoring of expired products or deteriorated which has a loss of 1,520 soles per week that annually range between 72,960.00 soles, its wide coverage in the market has allowed to identify the operational deficiencies in the inventory, for this reason the alternative of developing a mobile application with android technology was proposed using an inventory costing method to support the sales process of the Dicopersac company.

This inventory costing method makes it possible to disclose the inventory output that entered first, thus generating a more appropriate order for businesses that sell products with an expiration date. For which the research was presented that has as its main objective to be able to improve the sales process through the implementation of a mobile application with Android technology using the inventory costing method, having as specific objectives to define the requirements according to the information provided by the company, determine the method to be applied for the development of the software, carry out the analysis and design of the system based on the defined requirements and the established method, develop the Mobile App according to the defined requirements and applying the selected method and Validate the functionality of the mobile application according to the defined requirements, achieving the objectives of the thesis and with the relevant software tests. I used the RUP methodology for the project, which satisfies the context, time and development needs of the project and improves merchandise control problems.

keywords: Costing method, Inventory Management, Android Technology, Mobile Application.

I. INTRODUCCIÓN

Global Research Marketing (GRM) señala que el 67% de los clientes manifiestan que tienen mala experiencia al momento de realizar los servicios de las compras, siendo así que muchos de ellos no vuelven a comprar en el mismo lugar, mientras tanto un 60% de los clientes solicitan el libro de reclamos por la mala atención al público. Es así que las empresas buscan realizar innovaciones tecnológicas que permitan mejorar sus procesos especialmente el de ventas y control de inventarios. Debido a que uno de los aspectos por la que un cliente no regresaría a comprar sería por la demora en la atención con un 73.4 % según estudios realizados en el 2018. [1]

Actualmente muchas de las empresas buscan el mejoramiento del día a día y poder brindar mejor servicio de atención de los clientes, empleando nuevas tecnologías y estrategias. El autor Brian Halligan, director ejecutivo de Hubspot, señala que "el 80 % de las inversiones en tecnología de las empresas deberían centrarse en mejorar la experiencia del usuario". [2] Por lo tanto, el objetivo es expandir y mantener la presencia en los mercados existentes y entrar en nuevas direcciones, crear y lanzar productos exitosos, desarrollar marcas líderes y lograr el máximo alcance geográfico de presencia en cada punto de venta. [3].

La Sección de Información Mercados del CCL Centro de Comercio Exterior (CCEX), el Perú importó alrededor de 2.300 toneladas de dulces de chocolate con un valor de importación total de \$10 millones en 2016. En solo cinco países, esta proporción ronda el 73%. de las importaciones totales de chocolate y productos similares: USA 18%, Chile 14%, Brasil 12%, México 10% y Colombia 10%. [4]

Según Niels Segersbol, director general de Grams, mayorista de golosinas peruana, que anualmente se consumen unos 3,5 kg per cápita en forma de golosinas, lo que representa una gran oportunidad de crecimiento en este rubro. [5]

En otros países de la región, como México, Brasil o Argentina, el costo de los dulces por persona es mucho mayor, esto se debe a factores como el estilo de vida y mayor ingreso per cápita” - confirmó Segersbol [6].

En la región de Lambayeque, provincia de Chiclayo la empresa DICOPERSAC viene registrando pérdidas económicas por el mal control en su proceso de ventas

de producto e inventarios, tales como demoras en consultar stock de productos (entre 5 a 10 minutos), mal control de inventario (Kardex), demoras en las ventas y atención del cliente, adicional a esto el mal control y seguimiento de productos vencidos o deteriorados el cual tiene pérdida de 1,520 soles semanales que anualmente llegan a oscilar entre 72,960.00 soles. Es así que por tales pérdidas la empresa está en la búsqueda de soluciones haciendo uso de tecnologías que permitan mejorar los procesos: como primer objetivo se busca determinar el método de inventario, tomando en cuenta los requerimientos según la información brindada por la empresa tomando en cuenta el estado situacional del sistema, también se busca determinar el método de costeo, que se va a utilizar para implementar la aplicación móvil sistema de inventarios más apropiado para uso comercial en la empresa DICOPERSAC Chiclayo.

Al analizar el estado actual de la problemática de la empresa con los procesos de venta, se ha planteado la siguiente interrogante de investigación: ¿De qué manera se debe utilizar la tecnología para apoyar al proceso de las ventas en la empresa DICOPERSAC?

En respuesta a la interrogante planteada, se propone el implementar una aplicación móvil utilizando el método de costeo de inventarios más conocido como método PEPS de inventario.

La presente investigación, su principal objetivo es implementar un aplicativo móvil con tecnología android utilizando un método de costeo de inventario para apoyar el proceso de ventas en la empresa DICOPERSAC Chiclayo, teniendo como objetivos específicos determinar el método de costeo de inventario que se aplicará en el desarrollo Empresa DICOPERSAC Chiclayo, desarrollar una aplicación móvil según los requerimientos que han sido definidos teniendo en cuenta el método de costeo seleccionado, y validar la funcionalidad de la aplicación móvil de acuerdo a los requerimientos definidos por los usuarios interesados.

En lo científico, la investigación servirá como fuente y base para otras investigaciones que precisen desarrollar la tecnología Android. La gran cantidad de usuarios y la importancia de los dispositivos móviles para estos hacen que esta área sea interesante para investigar y estudiar el presente tema. De esta manera utilizaremos el método científico, con el fin de incrementar nuestros conocimientos y en consecuencia dar respuesta a nuestro problema planteado.

En lo económico, la implementación del Aplicativo Móvil con Tecnología Android en la empresa DICOPERSAC, desea disminuir tiempo y reducir costos operativos de la empresa donde el Supervisor del almacén se encargará del control de ingreso y salida de los productos sin pérdida de tiempo, ahorrando tiempo y dinero incrementando de esta manera las ventas a su distinguida clientela. La realización de esta implementación del proyecto significa una inversión que puede verse reflejada y recuperada en un corto y mediano plazo.

En lo Social, con el advenimiento de esta nueva aplicación móvil, la atención al cliente de DICOPERSAC puede ser mejor y más eficiente, aumentando la capacidad de atención a los clientes, reduciendo el esfuerzo de realizar las operaciones diarias del día a día de cada persona que está dentro de la empresa.

En lo tecnológico, el uso de la tecnología Android a través de los dispositivos móviles permitirá registrar de manera rápida y precisa las mercancías que ingresan y las mercancías que salen, y la cantidad de mercancías recibidas y liberadas, donde se verá reflejada en el almacén, permitiendo una atención rápida a los proveedores, así como los clientes que llegan hacer sus compras en la misma distribuidora.

En lo personal es necesario considerar que existe una preocupación por el crecimiento de la competencia en el mercado local, muchas empresas informales promocionan diversos productos a bajo costo, evadiendo impuestos al estado y creándose una pérdida de clientes para las empresas formales. Sin embargo para mantenerse en el mercado es necesario aplicar estrategias para mejorar las ventas y recuperar clientes, además el uso de la tecnología favorece los medios de difusión a bajo costo y sin contrato de promoción es decir no hay fecha de caducidad, nuestro propósito es lograr disponer el uso de la tecnología, diseñado en tecnología android, donde se pueda realizar transacciones en la BD(Base-datos) de inventarios del negocio y procesar información con facilidad, ahorrando tiempo, trabajo y dinero.

II. MARCO TEÓRICO

2.1. Antecedentes

Los siguientes predecesores que fueron considerados en este estudio

2.1.1. Antecedentes internacionales

Mairena y Vallejos [7], en su tesis denominada “Evaluación de los procesos de control de inventario y facturación para su automatización en farmacia Siloé, según las observaciones encontradas en la presente tesis es que la información se distribuye de manera completa e incorrecta, por lo tanto, cada vez que compra, depende de la búsqueda de todas las cuentas y boletos anteriores, y las ventas se registran en los registros del archivo, de estas actividades diarias que se realizan lentamente. No dan como resultado datos reales, siendo para ello necesarios para un buen control los datos.

La relación que guarda con la presente tesis, es que ambos proyectos tienen la necesidad de implementar una aplicación de escritorio y un aplicativo android que permiten llevar el control dentro del almacén, siendo la mejor opción y la más adecuada para el entorno y control actual, esta sensata herramienta garantizará una mayor seguridad en el procesamiento de datos.

Guadalupe y Balarezo [8], en su tesis denominada “Análisis, diseño e implementación de un Sistema de control de Inventario para el área de Producción de la Facultad de Ciencias Químicas de la Universidad de Guayaquil” La falta de automatización en la gestión de inventarios es un gran problema porque genera pérdida de materia prima por falta de control sobre sus niveles de inventario y la mala gestión de los mismos, hace que no se mejoren los niveles de venta, así como también la falta de información en tiempo real sobre la cantidad de materiales que son usados para fabricar los productos.

La relación que guarda con la presente tesis, es que ambos proyectos no cuenten con la automatización de información en

tiempo real es por ello que surge la necesidad de poder implementar un sistema de inventario para poder controlar la materia prima y los productos terminados que son utilizados para la elaboración de los mismos, permitiendo obtener información de manera oportuna.

Velásquez y Zeledón [9], en su tesis “Sistema de inventario y facturación de la tienda de accesorios de computadoras y celulares “Decosys”) en una empresa que vende accesorios informáticos y móviles. Nova SIS se desarrolla de acuerdo con el enfoque del modelo repetitivo con la finalidad de poder cumplir con las expectativas de los clientes mediante un prototipo permitiendo los mejoramientos de control del inventario y facturación de forma eficiente.

La relación que guarda con la presente tesis es que ambas investigaciones automatizan su información mediante un sistema de control, lo que redundará en una mejor gestión de la información para la toma de decisiones empresariales.

2.1.2. Antecedentes Nacionales

Quintanilla [10], en su tesis “Desarrollo del sistema de control de inventarios”, manifiesta que el principal problema es que no existe información en tiempo real y un control total sobre el movimiento de productos en el almacén y la distribución de herramientas. El objetivo general de este proyecto es llevar un registro de las entradas y salidas de materiales, herramientas y libros para garantizar un control adecuado de la ferretería y la librería. Frente a esta realidad problemática surge la necesidad de desarrollar la metodología PMBOOK, dando como resultado la planificación-diseño-desarrollo e implementación, permitiendo una mejor gestión de dispositivos y librerías, control total del movimiento e inventario en tiempo real, proporcionando de esta manera una nueva herramienta de trabajo para los empleados del Área de Control Conjunto, que permita reducir los tiempos de

reacción y mejorar la controlabilidad dentro del área de control patrimonial.

Oscanoa [11], refiere en su tesis “Desarrollo de una aplicación Android para incrementar la participación en medios digitales”, en nuestro país el uso masivo de dispositivos móviles como en todos los sectores socioeconómicos se convierte en una opción interesante como medio de comunicación y promoción de marca en medios una aplicación móvil sobre la plataforma Android para digitales. Este trabajo tiene como objetivo desarrollar contribuir al logro de las metas de la Universidad Privada del Norte en el sector digital. Hoy en día, con el avance de la tecnología, los dispositivos móviles nos permiten acceder a Internet para que podamos enviar y recibir información más rápido, en cualquier momento y en cualquier lugar.

Cornejo [12], “Análisis, diseño e implementación de una aplicación para administrar y consultar avisos clasificados para tabletas android” Este proyecto incluye análisis, diseño e implementación. Aplicación de gestión de publicaciones y asesoramiento sobre notificaciones secretas estructurado para tabletas con sistemas operativos de Android, centrándose en todos los tipos. Permitiendo al usuario la interacción con la interfaz de manera fácil en usar -con interfaces totalmente amigables, amable y fácil de entender por intuición, y también combina las funciones más utilizadas Herramientas (sitios web, aplicaciones, periódicos, etc.) Mantequilla para notificar las notificaciones y poder clasificarlos.

2.1.3. Antecedentes locales

Oblitas [13], “Aplicación móvil multiplataforma como guía para orientar al turista en su estadía”, manifiesta la importancia del turismo siendo una de las áreas de actividad económica que ha crecido de forma espectacular en los últimos años, Es por esto que el turismo se considera una alternativa viable para el desarrollo del Perú, eligiendo el turismo cultural como la mejor opción para la región Lambayeque, permitiendo a los visitantes conocer destinos turísticos, gastronómicos y de alojamiento. Ante la realidad que vivimos y a la problemática presentada se descubre que han pasado demasiado tiempo para referirse a destinos turísticos, restaurantes o apartamentos, el nivel de insatisfacción de los turistas por la falta de información, es por ello se pretende implementa una aplicación móvil que sirve como guía para acortar el tiempo promedio que un turista para consultar con la geografía en tiempo real, obteniendo como resultados los de stinos tu rístic os, restaurantes y vida; Aumento del nivel de satisfacción de los turistas, aumentando el porcentaje de turistas con información útil y aumentando el número de ubicaciones promedio que los turistas visitan de acuerdo con sus preferencias.

Bustamante & Lozano [14], en su tesis denominada “desarrollo de aplicación web basado en el modelo de revisión continua y utilizando la tecnología rfid para mejorar la gestión de inventarios de vehículos automotores menores” manifiesta que la presente tesis está enfocado al desarrollo de una aplicación web para mejorar la gestión de inventarios en Lima Motor S.R.L utilizando tecnología de identificación por radiofrecuencia (RFID), siendo su principal problema el proceso automatizado de la información para gestionar el inventario de vehículos de pasajeros, realizándose de forma manual, surgiendo la necesidad de poder automatizarlos y aumentar la eficiencia. El objetivo fue desarrollar una aplicación web basada en el modelo de Vista Continua y tecnología RFID utilizada para mejorar la gestión de inventario de vehículos pequeños en Lima Mo

tor S.R.L, permitiendo el tránsito de vehículos entre puntos de entrega de las .ventas, se registren sus importaciones y exportaciones, más aun la actualización del inventario e informe el tiempo de respuesta y el inventario de manera rápida y eficaz. Finalmente, se comprobó la hipótesis comparando los resultados de la aplicación web con registros previamente procesados manualmente. Con base en los resultados de las pruebas realizadas con el uso de la aplicación web, se encontró que el nivel de confianza es del 95%.

Vargas [15], “Aplicación web adaptativa para mejorar el control de inventarios de confecciones “angiara”, utilizando la planificación de los recursos de materiales”, empresa de confecciones de todo tipo, se ocupa de la venta local y nacional con una cartera de clientes consolidada; Su modelo de negocio se basa en producir prendas y materias primas de alta calidad. Varios temas se relacionan con los procesos internos, especialmente la planificación de recursos. Su principal objetivo es desarrollar un modelo de lote económico apoyado por un sistema de red adaptativo para mejorar la gestión de inventario, utilizando métodos de recopilación de datos como encuestas, entrevistas y fichas de observación como medio para extraer información e interpretarla. Su principal problema es analizar la planificación y control de inventario de los recursos de producción, ya que genera deficiencias en la coordinación de entregas, por falta de política e información sobre el inventario adecuado en el almacén de suministros teniendo como propuesta el desarrollo de una aplicación web para agilizar la gestión de inventario de la empresa para mejorar la eficiencia operativa y un mejor control de inventario.

2.2. Bases Teórico Científicas

2.2.1. Inventario

El inventario es el stock de bienes que se mantienen para su uso o ventas en el futuro. La gestión de inventario implica mantener estos bienes disponibles cuando necesitan ser usados o vendidos, en base a políticas que determinan cuándo y cuánto inventario debe reponerse. [16]

La gestión de inventarios se enfoca en cuatro (04) aspectos básicos:

1. Número de unidades que deberán producirse en un momento dado.
2. A qué hora se debe hacer el inventario.
3. ¿Qué elementos del inventario merecen una atención especial? y
4. ¿Podemos protegernos de los cambios en los costos de los artículos en los stock?

La gestión de inventario se trata de asegurar el inventario necesario para operar su negocio al menor costo posible.

El propósito de la gestión de inventarios tiene dos aspectos contrapuestos: por un lado, es necesario minimizar la inversión en reservas, ya que los recursos que no se destinan para este fin se pueden invertir en proyectos suficientes, por otro lado, que no se pueden financiar. Por otro lado, debe asegurarse de que la empresa tenga suficiente inventario para satisfacer la demanda a medida que ocurre y mantener la producción y las ventas funcionando sin problemas.

Por lo tanto, la gestión de inventario tiene como objetivo armonizar o equilibrar los siguientes objetivos:

- ✓ Máxima atención al cliente.
- ✓ Maximizar la eficiencia de los departamentos de compras y la calidad de producción.
- ✓ Minimizar sus inversiones en el inventario.

2.2.2. Análisis de Inventarios

Para lograr la eficiencia de los recursos en el manejo de materia prima y que el producto final de un valor adecuado a este concepto, cada empresa, institución u organización debe establecer una

política de gestión de inventarios que tenga en cuenta los términos de su consecución de los objetivos de la empresa. [17]

Según lo mencionado por el autor, una política eficaz de gestión de inventario, es aquella que determina el nivel óptimo de inversión en inventario y asegura que el nivel óptimo se alcance a través del control.

Las clasificaciones de acciones al nivel de inventarios más comunes son:

Materia Prima: Este es un producto que no ha sufrido ningún procesamiento.

Producto en Proceso: Inventario en proceso o pendiente en la línea de producción.

Componentes de Nivel Alto: Son las piezas o ensamblajes que están listos para ensamblarse en un producto terminado.

Productos Terminados: El producto que ha completado el ciclo de producción y está listo para ser entregado al consumidor.

2.2.2.1. Nivel Óptimo de Inventario

Es un nivel que permite cubrir plenamente las necesidades de la empresa con una mínima inversión. [17]

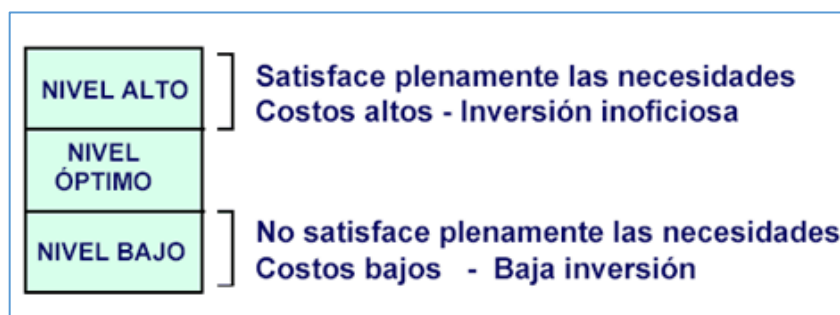


Fig. 1 Nivel Óptimo de Inventario [1]

Existen diferentes áreas dentro de la empresa que tienen diferentes necesidades de nivel de inventarios:

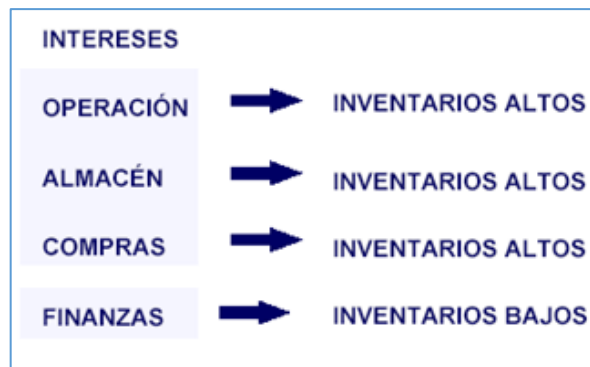


Fig. 2 Intereses del Nivel Óptimo de Inventario [2]

2.2.3. Determinación del punto de reorden

La puntuación de reorden refleja el uso diario del inventario de la empresa y la cantidad de días en que se realizó y recibió la orden, donde cada área debe realizar un pedido antes de que se agote el stock actual, teniendo en cuenta la cantidad de días que tardará el proveedor en recibir y procesar el pedido, así como el tiempo que tardará en enviarse la mercancía. [17]

El punto de reorden, es considerado como el nivel crítico de inventario que debe ser manejado por las empresas industriales, lo que significa que hay una señal para el departamento responsable de ordenar que cierta cantidad de material o producto haya alcanzado un cierto nivel y debe ser nuevo pedido.

Existen muchas formas de indicar un punto de reorden, que van desde una etiqueta, papel, tarjeta o cupón colocado en un gabinete de almacenamiento o en una pila de bolsas que indica que se debe realizar un nuevo pedido, hasta métodos más complejos.

Algunas herramientas para gestionar estos inventarios son:

- La requisición viajera: El objetivo de esta herramienta es ahorrar mucho trabajo administrativo ya que los puntos de control y las aprobaciones se configuran con anticipación para que se realicen nuevos pedidos a través de esta instalación y los materiales o artículos no estén disponibles en las empresas.

Hay dos sistemas principales que utilizan el transporte de propulsión para complementar.

- Órdenes o pedidos fijos. El objetivo es realizar un pedido cuando el stock suficiente cumpla con el requisito máximo que puede existir durante el tiempo que tarda en llegar un nuevo pedido.
- Resurtidos periódicos Este sistema es muy común en la mayoría de los casos cuando se establece un control de inventario ilimitado. La idea principal de este sistema es el conocimiento del inventario.[\[17\]](#)

Para conseguir el punto de reorden debemos utilizar la siguiente fórmula:

$$ROP = d * L$$

Donde:

R = Punto de reorden

L = Tiempo de entrega

d= periodo de tiempo promedio (unidades por día, semana, mes)

2.2.4. Existencias de reserva o seguridad de inventarios

La mayoría de las empresas a nivel mundial necesitan mantener existencias de seguridad que permitan mantenerse al día con una demanda superior a la esperada. Estas provisiones se realizan para paliar choques o situaciones derivadas de cambios impredecibles en los requerimientos de bienes. [\[17\]](#)

El inventario a veces se almacena como productos semiacabados para equilibrar las necesidades de producción de los diferentes procesos o partes que componen la producción para que los programas de producción se puedan ajustar y el inventario se reponga de manera oportuna.

Por lo general, no es posible predecir todos los problemas y fluctuaciones que pueden surgir con la demanda, aunque es completamente cierto que las empresas deben tener reservas de existencias si no quieren tener clientes insatisfechos.[\[18\]](#)

2.2.5. Control de Inventarios justo a tiempo

En el control de inventarios justo a tiempo, es que el inventario se compra y se pone en producción de proveedores muy confiables y en un sistema de gestión de inventario eficiente [\[18\]](#)

Una empresa puede reducir la producción en el proceso de producción a través de una mejor gestión, incluidos los factores internos. Si es necesario, se pueden reducir. Esto requiere un suministro muy eficiente de las materias primas requeridas debido a una mayor eficiencia interna, pero en mayor medida a factores externos.

Gracias a la cooperación, incluidos los proveedores confiables, se puede reducir la cantidad de materias primas, como en el caso de los productos terminados, se puede decir que con su rápida reposición, se reduce el costo de la falta de inventario y al mismo tiempo inventario.

La propuesta presentada tiene como objetivo mejorar la gestión de inventarios aumentando el nivel de inventario óptimo de la empresa y está interesada en aumentar el nivel de ventas y control de la empresa.

2.2.6. Métodos de evaluación de inventarios

Entre los métodos de valoración de costeos más importantes para tenemos [19]:

Tabla 1: Métodos de evaluación de inventario

MÉTODOS	DEFINICIÓN	VENTAJAS	DESVENTAJAS
COSTO PROMEDIO	El método de costo promedio ponderado, a menudo denominado método de costo promedio, se basa en el costo promedio ponderado del inventario durante un período de tiempo. Este método trata el costo unitario como el costo unitario promedio del período, es decir, si el precio unitario disminuye o aumenta durante el período, valor medio de estos costes. El costo promedio se determina de la siguiente manera: Dividir el costo de los artículos disponibles para la venta (inventario al inicio de la compra) por el número de unidades disponibles. [19]	<ul style="list-style-type: none"> Fácil de aplicar En una economía inflacionaria, representa un nivel razonable de utilidad porque promedia los costos pasados y presentes. [19] 	<ul style="list-style-type: none"> No permite llevar un control detallado del costo de la mercadería entrante y saliente. [19]
PEPS o FIFO	Según este método consisten en primeras entradas, primeras salidas, donde las empresa deben realizar un seguimiento del costo de cada artículo comprado de su inventario. El precio unitario utilizado para calcular el inventario final puede ser diferente del precio unitario utilizado para calcular el costo de los bienes vendidos. Según FIFO, el primer costo en el inventario es el primer costo de los bienes vendidos, de ahí el nombre Primero en entrar, primero en salir. [19]	<ul style="list-style-type: none"> Las existencias al final del período se valoran al precio de las últimas compras El costo del inventario vendido está determinado por el precio de compra inicial. Por tanto, los gastos incluidos en la cuenta de pérdidas y ganancias son menores reconocida por otros métodos de valoración Cuanto menores sean los gastos en la cuenta de pérdidas y ganancias, mayor será el margen de beneficio bruto en comparación con la obtenida por otros métodos[19] 	<ul style="list-style-type: none"> Cuanto mayor sea la utilidad, mayores serán los pagos de impuestos. En una economía inflacionaria, la utilidad puede exagerarse porque compara el precio de compra anterior con el precio de venta actual. [19]
UEPS o LIFO	El método últimas entradas, primeras salidas, depende del precio de compra de las acciones individuales. Bajo este método, el costo final del inventario es el primero que se carga al costo de los bienes vendidos. Este método ha sido desaprobado por las Normas Internacionales de Contabilidad (NIIF).[19]	<ul style="list-style-type: none"> Compare los costos actuales con los precios de venta actuales, para que pueda los resultados reflejan una menor usabilidad que otros métodos. Menos comodidades le permiten reducir sus impuestos [19] 	<ul style="list-style-type: none"> La aplicación y operación del Kardex son más complejas que otros métodos. Las existencias al final del período se valúan al costo por qué no reflejan honestamente el valor real del inventario final.[19]

2.2.7. Sustento de la elección del método de costeos

El método de costeos de inventario utilizado para este proyecto de tesis fue el del Método FIFO el cual tiene valor contable debido a que el inventario de la empresa se reconocerá al último costo de compra o producción y el costo de venta resultante será el precio correspondiente al inventario inicial, según consta por la NIF (Normalización de la Información Financiera), adicionalmente la aplicación podrá emitir un Kardex utilizando una hoja de cálculo de Excel, esta hoja de cálculo será exportada desde la aplicación en cualquier momento cuando lo solicite(n) el(los) administrador(es) y propietario(s).

2.2.8. Marcos de trabajo para el desarrollo de aplicaciones móviles

2.2.8.1. Tipos de aplicaciones Móviles

Una aplicación móvil es una colección de instrucciones lógicas, procedimientos, reglas, documentos, datos e información relacionada que funciona específicamente en dispositivos móviles como teléfonos inteligentes, televisores inteligentes, mesas de computadoras, relojes y otros dispositivos. La ventaja principal de contar con una aplicación, es que facilita los trámites y actividades que el usuario realiza de forma habitual.

Las aplicaciones móviles deben estar diseñadas específicamente para usarse en cualquier momento y en cualquier lugar, y deben cambiar continuamente para adaptarse a las necesidades del usuario.

Hay 3 tipos de aplicaciones móviles:

- Aplicacion Nativas
- Aplicacion We b
- Aplicacion Híbridas

El propósito de esta tesis, se desarrollará una aplicación móvil para mejorar el proceso de ventas del control interno [\[20\]](#)

2.2.8.2. Sistema operativo Móviles

Actualmente existe diversidad de sistemas, siendo lo más importantes:

- Android (con más del 80% del mercado): Además de ser el más utilizado, también es apreciado por una gran cantidad de usuarios por su flexibilidad y facilidad de uso de los servicios integrados de Google. [\[22\]](#)
- IOS (con más del 10% del mercado): muy apreciado por su seguridad, compatibilidad con el hardware de Apple y facilidad de uso. [\[22\]](#)
- Windows Phone: lo que es más importante, es compatible con la mayoría de los programas que se ejecutan en

Windows, incluida la popular suite de Microsoft Office. [22]

- Blackberry: diseñado para uso comercial. Es un sistema operativo principalmente para mensajería instantánea, correo electrónico y otras funciones relacionadas con la comunicación. [22]

- Symbian: Sistema operativo confiable (comercializado por más de 10 años), batería de larga duración y sistema multitarea bien desarrollado. [22]

El sistema operativo utilizado en este proyecto será Android, ya que cubre el 80% del mercado actual en el que se desarrollará la aplicación.

2.2.8.3. Arquitectura Android

Como se mencionó, Android es una plataforma para dispositivos móviles que incluye el sistema operativo, el middleware y las aplicaciones básicas de usuario. [21]

Las siguientes líneas dan una descripción general de la arquitectura utilizada en Android: [21]



Fig. 3 Arquitectura Android [3]

III. METODOLOGÍA

3.1. Tipo y nivel de investigación

3.1.1. Tipo de investigación

La investigación para el presente trabajo, este tipo tecnológica aplicada, debido a que se emplea la tecnología en tiempo real, en este caso en una tienda comercial, para optimizarlos procesos de gestión de almacén para apoyar en el proceso de ventas utilizando el método PEPS, a través, del uso de dispositivos móviles, mediante el uso de la aplicación móvil android implementada.

3.1.2. Diseño y Nivel de investigación

Es un estudio de tipo experimental, donde una variable independiente (aplicativo móvil con tecnología android utilizando el método costeo de inventario), se manipula y se pretende establecer el posible efecto sobre la variable dependiente (Proceso de Ventas).

El nivel de investigación del presente trabajo es pre-experimental, cuyo grado de control es mínimo. Se ha realizado una sola prueba en las oficinas de la empresa.

Tabla 2: Contratación De Hipótesis

Variable Independiente	Estímulo	Variable Dependiente
<u>O1:</u> Aplicativo móvil con tecnología Android utilizando un Método de Costeo de inventario	<u>X:</u> Diseño de aplicación móvil	<u>O2:</u> Procesos de venta.

Dónde:

O1 = Aplicativo móvil con tecnología Android utilizando un Método de Costeo de inventario.

O2 = Procesos de venta.

X = Diseño de aplicación móvil

3.2. Población, muestra y muestreo

3.2.1. Población

Tiene como sustento a una cartera de clientes de 850 clientes activos.

3.2.2. Muestra

La muestra es obtenida a partir de la fórmula del procedimiento para estimar el tamaño de una muestra representativa de una población finita, determinada por Bernal [1]:

$$n = \frac{N \cdot Z^2 \cdot p \cdot q}{d^2 \cdot (N - 1) + Z^2 \cdot p \cdot q}$$

Dónde:

n = Tamaño de la muestra por estimar.

Z = 1.96 (Si la seguridad es de 95%)

N = Total de población 850 clientes activos.

p = Proporción esperada (en este caso 5% = 0.05)

q = 1-p (en este caso 1 – 0.05 = 0.95)

d = Precisión (en este caso 5%) 0.05

- **Muestra 1**

La muestra de la población 1 incluye:

$$\frac{850 \cdot 1.96^2 \cdot 0.05 \cdot 0.95}{0.05^2 (850 - 1) + 1.96^2 \cdot 0.05 \cdot 0.95} = 67$$

$n = 67$ clientes activos

3.2.3. Muestreo

Como método de muestreo se utilizaron estudios documentales y observacionales.

3.3. Criterios de selección

La delimitación de la población es tener la base de datos de las mercancías de las venta, sus precios y la disposición en almacén. Además, entre el proceso de pedido, pago y entrega del producto se necesita llamar a la oficina y realizar las transacciones necesarias, sin embargo, el tiempo de espera muchas veces permite que la venta se pierda y la empresa logre ser más eficiente en sus gestiones administrativas.

3.4. Operacionalización de variables

3.4.1. Variables

3.4.1.1. Variable independiente

Aplicativo móvil con tecnología Android utilizando un Método de Costeo de inventario.

3.4.1.2. Variable dependiente

Proceso de ventas.

3.4.2. Indicadores (Operacionalización de variables)

Objetivos específicos	Indicador(es)	Definición conceptual	Unidad de medida	Instrumento	Definición operacional
- Determinar el método de costeo de inventario que se aplicará en el desarrollo del software	Cantidad de métodos evaluados	Elección entre los métodos de inventario que mejor se adapte al problema del negocio	Porcentaje efectividad	Encuesta	Suma de puntaje / total.
-Desarrollar la aplicación móvil según los requerimientos definidos y el método de costeo seleccionado.	Grado de implementación los requerimientos funcionales definidos. Grado de implementación del método.	Puesta en marcha del aplicativo	Porcentaje satisfacción	Encuesta	$(N^{\circ} \text{ datos} / \text{total}) * 100$
-Validar la funcionalidad de la aplicación móvil, según los requerimientos definidos con los usuarios interesados.	Grado de satisfacción de los usuarios	Validar el cumplimiento de los objetivos.	Porcentaje satisfacción	Encuesta	$(N^{\circ} \text{ datos}) \text{ Suma}$

3.5. Técnicas e instrumentos de recolección de datos

El proceso de análisis y obtención de información se hizo a través de test con el fin de conocer la situación actual de los procesos de compra, la programación de los pedidos y las entregas de los productos.

Tabla 3: TÉCNICAS E INSTRUMENTOS DE RECOLECCIÓN DE DATOS

Técnicas	Instrumentos	Elementos de la población	Propósito
Test	Cuestionario	Cartera de clientes	Eficiencia económica (Inversión)
Análisis documentario	Fichas o guías de observación	Inventario	Promoción a las ventas (Recuperación)
Test	Cuestionario	Cartera de clientes	Eficiencia económica (Inversión)

3.6. Procedimientos

3.6.1. Metodología de desarrollo (MD)

Iteraciones del MD, caso RUP:

1. Iteración #1: Planif. del proy. (Plan de sistemas)

En esta iteración se desarrollaron las siguientes actividades:

- Estudio de Documentación
- Toma de datos y tiempos
- Estimar cronogramas y Costos

2. Iteración #2: Análisis Preliminar de Req. - Modelado de Negocio - Diagrama de Contexto (DC)

En esta iteración se desarrollaron las siguientes actividades:

- DC del negocio
- DCUN del negocio
 - Diagrama-actividades-negocio
 - Diagrama-objetos- negocio

3. Iteración #3: Análisis Preliminar de Requerimientos – Casos de Uso

En esta iteración se desarrollaron las siguientes actividades:

- Diag. Contexto (DC)
- Diag. casos uso (DCUN Por paquete)
 - Diag. de actividades

- Diagrama de objetos

- Diagrama-objetos-general (Modelo/dominio)

4. Iteración #4: Análisis

En esta iteración se desarrollaron las siguientes actividades:

- Diagrama-contexto-análisis(DCA)
- Diagrama-Realización- casos de uso análisis(DRCUA)
 - Diagrama clases-análisis(DCA)
 - Diagrama-clases- parciales(DCP)
- Diagrama- clases- General(DCG)

5. Iteración #5: Diseño

En esta iteración se desarrollaron las siguientes actividades:

- Subsistema de diseño en función a paquetes
- Diagrama-Realizaciones-casos de uso diseño(DRCUD)
 - Diagrama-clases-diseño(DCUD)
 - Diagrama-Secuencia(DS)
- Diagrama-clases-diseño-General(DCDG)
- Diseño de Base de datos
 - Base/Datos Lógica (BD-LOG)
 - Base/Datos Física(BD-FIS)
- Diagramas-estados(DE)
- Diseño-Interfaces(DI)
- Diagrama-Despliegue(DD)

6. Iteración #6: Implementación y Prueba

En esta iteración se desarrollaron las siguientes actividades:

- Diseño-Interfaces(DI)
- Diagrama-Despliegue(DD)
- Diagrama-componentes(DC)
- Pseudocódigo

3.6.2. Análisis de riesgos

El análisis de riesgos se efectuó con la finalidad de identificar las fases, entregables y objetivos afectados en el desarrollo de tesis.

Ver *Anexo N° 01*.

3.6.3. Producto acreditable

1. Interfaces

Se construyeron las Interfaces del Aplicativo Móvil haciendo uso del lenguaje Android la misma que se presentan en el **Anexo N° 01: Manual de Usuario**

2. Arquitectura

Se diseñó una arquitectura idónea para el funcionamiento del aplicativo móvil se detalla en el **Iteración #6: Capítulo IV. Resultados.**

3. Infraestructura tecnológica

Considerando la arquitectura anteriormente descrita, se definen las características de cada uno de sus componentes en el **Iteración #6: Capítulo IV. Resultados.**

3.6.4. Manual de usuario

El manual del usuario está diseñada para ayudar a los mismos usuarios tal como se describe en *Anexo N° 99*.

3.7. Plan de procesamiento y análisis de datos

Para el procesamiento de datos se realizó los siguientes procedimientos:

- Después de la aplicación de la encuesta se procedió a registrarlos datos utilizando el programa Excel se ordena la posición (tabulación) para el diseño de tablas y gráficos.
- El Análisis de datos aplicamos el programa estadístico Statical Package for the Social Sciense (SPSS 20) desarrollamos los siguientes procesos: Estructura de base de datos SPSS, Declaración de variables, migración de datos fuentes proveniente del programa Excel, luego se calculó la estadística descriptiva y el Análisis de fiabilidad.
- Preparación del informe según los datos obtenidos utilizamos el programa Word, exportamos los datos del SPSS e interpretamos los resultados de acuerdo a los resultados, finalmente obtenemos las conclusiones.

La información será presentada en tablas estadísticas, con sus respectivos gráficos y los estadígrafos más relevantes y representativos luego de aplicar el pre y post test.

3.8. Matriz de Consistencia

Tabla 4: MATRIZ DE CONSISTENCIA

PROBLEMA	OBJETIVOS	HIPOTESIS	VARIABLES
<u>FORMULACION DEL PROBLEMA</u>	<u>OBJETIVO GENERAL</u>	<u>HIPOTESIS</u>	<u>VARIABLES DE ESTUDIO</u>
¿De qué manera se debe utilizar la tecnología para mejorar el proceso de ventas en la empresa DICOPERSAC?	Desarrollar una aplicación móvil con Tecnología Android utilizando un método de costeo de inventario, para apoyar el proceso de ventas en la empresa Dicopersac Chiclayo	Con la implementación de una aplicación móvil con Tecnología Android utilizando un método de costeo de inventario se apoyará el proceso de ventas en la empresa Dicopersac Chiclayo	VARIABLE INDEPENDIENTE Aplicativo móvil con tecnología Android utilizando un Método de costeo de inventario. VARIABLE DEPENDIENTE Proceso de ventas
<u>OBJETIVOS ESPECIFICOS</u>	<u>DESCRIPCION DEL LOGRO DE LOS OBJETIVOS ESPECIFICOS</u>		<u>INDICADORES</u>
- Determinar el método de costeo de inventario que se aplicará en el desarrollo del software	Establecer el mejor método de inventario para la empresa		• Cantidad de métodos evaluados.
- Desarrollar la aplicación móvil según los requerimientos definidos y el método de costeo seleccionado.	Establecer el adecuado método de costeo para el inventario para el mejor análisis y diseño de la metodología RUP.		• Porcentaje de requerimientos especificados respecto a los requerimientos definidos.
- Validar la funcionalidad de la aplicación móvil, según los requerimientos definidos con los usuarios interesados.	Sustentar la funcionalidad y cumplimiento del requerimiento de la aplicación desarrollada.		• Grado de implementación del método.

3.9. Consideraciones éticas

Tenemos presente algunos criterios éticos a la hora del diseño del estudio y en el momento de la recolección de datos de la investigación.

Tabla 5: CRITERIOS ÉTICOS.

Criterio	Descripción
El consentimiento informado	Los participantes de la encuesta estuvieron de acuerdo en brindarnos la información requerida conociendo sus derechos y responsabilidades de las mismas
La confidencialidad	Asegurar la protección de la identidad de las personas que participan como informantes de la investigación.
Observación participante	Responsabilidad ética por los efectos y consecuencias que puede derivarse de la interacción establecida con los sujetos participantes del estudio.
Entrevistas	No se deben provocar actitudes que condicionen la respuesta de los participantes.

IV. RESULTADOS

4.1. En base a la metodología utilizada

4.1.1. Iteración #1: Planificación del proyecto (Plan de sistemas)

En esta iteración se desarrollaron las siguientes actividades:

Descripción general de la empresa

En la región de Lambayeque, provincia de Chiclayo la empresa DICOPERSAC viene registrando pérdidas económicas por el mal control en su proceso de ventas de producto e inventarios de almacén, es así que la empresa está en la búsqueda de soluciones tecnológicas que permitan mejorar estos procesos: como primer objetivo se busca determinar el método de inventario, tomando en cuenta el estado situacional del sistema, también se busca determinar el método de costeo, que se va a utilizar para implementar la aplicación móvil sistema de inventarios más apropiado para uso comercial en la empresa DICOPERSAC Chiclayo.

4.1.2. Iteración #2: Análisis Preliminar de Req. -Modelo/ Negocio.

- **Requerimientos Funcionales**

En las siguientes tablas corresponden a los Requerimientos Funcionales :

Tabla 6: IDENTIFICACIÓN DE REQUERIMIENTO FUNCIONAL INICIAR SESIÓN

Identificación R-F-01					
Título	Iniciar Sesión				
Descripción	El Aplicativo permite iniciar sesión solo a usuarios autorizados				
Prioridad	*	Alta	Estabilidad	*	Si
		Media			No
		Baja	Verificabilidad	*	Alta
Claridad	*	Alta			Media
		Media			Baja
		Baja			
Necesidad	*	Esencial		Deseable	Opcional

FUENTE: ELABORACIÓN PROPIA

Tabla 7: IDENTIFICACIÓN DE REQUERIMIENTO FUNCIONAL MENÚ PRINCIPAL

Identificación R-F-02					
Título	Menú Principal				
Descripción	La aplicación dispondrá de un menú principal, donde el usuario podrá elegir entre las opciones: - Mis Pedidos - Nuevo Pedido. - Nuevo Cliente.				
Prioridad	*	Alta	Estabilidad	*	Si
		Media			No
		Baja	Verificabilidad	*	Alta
Claridad	*	Alta			Media
		Media			Baja
		Baja			
Necesidad	*	Esencial		Deseable	Opcional

FUENTE: ELABORACIÓN PROPIA

Tabla 8: IDENTIFICACIÓN DE REQUERIMIENTO FUNCIONAL MIS PEDIDO

Identificación R-F-03					
Título	Módulo Mis Pedido				
Descripción	Este módulo permite visualizar el estado de los pedidos realizado por el vendedor				
Prioridad	*	Alta	Estabilidad	*	Si
		Media			No
		Baja		Verificabilidad	*
Claridad	*	Alta			Media
		Media			Baja
		Baja			
Necesidad	*	Esencial		Deseable	Opcional

FUENTE: ELABORACIÓN PROPIA

Tabla 9: IDENTIFICACIÓN DE REQUERIMIENTO FUNCIONAL REGISTRAR CLIENTE

Identificación R-F-04					
Título	Módulo Registrar Cliente				
Descripción	Este módulo permite realizar el registro de un nuevo cliente				
Prioridad	*	Alta	Estabilidad	*	Si
		Media			No
		Baja		Verificabilidad	*
Claridad	*	Alta			Media
		Media			Baja
		Baja			
Necesidad	*	Esencial		Deseable	Opcional

FUENTE: ELABORACIÓN PROPIA

Tabla 10: IDENTIFICACIÓN DE REQUERIMIENTO FUNCIONAL REGISTRAR PRODUCTO

Identificación R-F-05					
Título	Módulo Registrar Producto				
Descripción	Este módulo permite realizar el registro de un nuevo producto				
Prioridad	*	Alta	Estabilidad	*	Si
		Media			No
		Baja		Verificabilidad	*
Claridad	*	Alta			Media
		Media			Baja
		Baja			
Necesidad	*	Esencial		Deseable	Opcional

FUENTE: ELABORACIÓN PROPIA

Tabla 11: IDENTIFICACIÓN DE REQUERIMIENTO FUNCIONAL REGISTRAR CATEGORÍA DE PRODUCTO

Identificación R-F-06					
Título	Módulo Registrar Categoría de Producto				
Descripción	Este módulo permite registrar categoría de productos				
Prioridad	*	Alta	Estabilidad	*	Si
		Media			No
		Baja	Verificabilidad	*	Alta
Claridad	*	Alta			Media
		Media			Baja
		Baja			
Necesidad	*	Esencial		Deseable	Opcional

FUENTE: ELABORACIÓN PROPIA

Tabla 12: IDENTIFICACIÓN DE REQUERIMIENTO FUNCIONAL REGISTRAR MARCA DE PRODUCTO

Identificación R-F-07					
Título	Módulo registrar marca de producto				
Descripción	Este módulo permite registrar marca de producto				
Prioridad	*	Alta	Estabilidad	*	Si
		Media			No
		Baja	Verificabilidad	*	Alta
Claridad	*	Alta			Media
		Media			Baja
		Baja			
Necesidad	*	Esencial		Deseable	Opcional

FUENTE: ELABORACIÓN PROPIA

Tabla 13: IDENTIFICACIÓN DE REQUERIMIENTO FUNCIONAL DEVOLUCIONES DE COMPRAS (FACTURA-BOLETAS)

Identificación R-F-08					
Título	Módulo visualizar las devoluciones de compras				
Descripción	Este módulo permite devoluciones de compras (factura-boletas)				
Prioridad	*	Alta	Estabilidad	*	Si
		Media			No
		Baja	Verificabilidad	*	Alta
Claridad	*	Alta			Media
		Media			Baja
		Baja			
Necesidad	*	Esencial		Deseable	Opcional

FUENTE: ELABORACIÓN PROPIA

Tabla 14: IDENTIFICACIÓN DE REQUERIMIENTO FUNCIONAL COMPRA REGISTRADAS

Identificación R-F-09					
Título	Módulo visualizar compras registradas				
Descripción	Este módulo permitirá visualizar las compras registradas				
Prioridad	*	Alta	Estabilidad	*	Si
		Media			No
		Baja	Verificabilidad	*	Alta
Claridad	*	Alta			Media
		Media			Baja
		Baja			
Necesidad	*	Esencial		Deseable	Opcional

FUENTE: ELABORACIÓN PROPIA

Tabla 15: IDENTIFICACIÓN DE REQUERIMIENTO FUNCIONAL VENTAS REGISTRADAS

Identificación R-F-10					
Título	Módulo visualizar ventas registradas				
Descripción	Este módulo permite visualizar las ventas registradas				
Prioridad	*	Alta	Estabilidad	*	Si
		Media			No
		Baja	Verificabilidad	*	Alta
Claridad	*	Alta			Media
		Media			Baja
		Baja			
Necesidad	*	Esencial		Deseable	Opcional

FUENTE: ELABORACIÓN PROPIA

Tabla 16: IDENTIFICACIÓN DE REQUERIMIENTO FUNCIONAL KARDEX

Identificación R-F-11					
Título	Módulo visualizar Kardex de cada producto				
Descripción	Este módulo permite visualizar el Kardex				
Prioridad	*	Alta	Estabilidad	*	Si
		Media			No
		Baja	Verificabilidad	*	Alta
Claridad	*	Alta			Media
		Media			Baja
		Baja			
Necesidad	*	Esencial		Deseable	Opcional

FUENTE: ELABORACIÓN PROPIA

- **Requerimientos No Funcionales**

A continuación, se muestra los requerimientos no funcionales para el desarrollo del aplicativo.

Tabla 17: IDENTIFICACIÓN DE REQUERIMIENTO NO FUNCIONAL DE COMPATIBILIDAD DE S.O

Identificación R-NF-01					
Título	Compatibilidad con Sistemas Operativos				
Descripción	El sistema deberá ser compatible con las diferentes versiones de Android				
Prioridad	*	Alta	Estabilidad	*	Si
		Media			No
		Baja	Verificabilidad	*	Alta
Claridad	*	Alta			Media
		Media			Baja
		Baja			
Necesidad	*	Esencial		Deseable	Opcional

FUENTE: ELABORACIÓN PROPIA

Tabla 18: IDENTIFICACIÓN DE REQUERIMIENTO NO FUNCIONAL DE COMPATIBILIDAD DE RESOLUCIÓN

Identificación R-NF-02					
Título	Compatibilidad con Resoluciones				
Descripción	La aplicación será compatible con diferentes resoluciones de pantalla.				
Prioridad	*	Alta	Estabilidad	*	Si
		Media			No
		Baja	Verificabilidad	*	Alta
Claridad	*	Alta			Media
		Media			Baja
		Baja			
Necesidad	*	Esencial		Deseable	Opcional

FUENTE: ELABORACIÓN PROPIA

Tabla 19: IDENTIFICACIÓN DE REQUERIMIENTO NO FUNCIONAL DE INTERFAZ SENCILLA E INTUITIVA

Identificación R-NF-03					
Título	Interfaz sencilla e intuitiva				
Descripción	La aplicación deberá ser sencilla e intuitiva para los usuarios con pocos conocimientos tecnológicos y que pueda ser usada sin esfuerzo.				
Prioridad	*	Alta	Estabilidad	*	Si
		Media			No
		Baja	Verificabilidad	*	Alta
Claridad	*	Alta			Media
		Media			Baja
		Baja			
Necesidad	*	Esencial		Deseable	Opcional

FUENTE: ELABORACIÓN PROPIA

- **Modelado del Negocio**

Capítulo II: Modelado Del Negocio

- **Diagrama Contexto Negocio(DCN)**

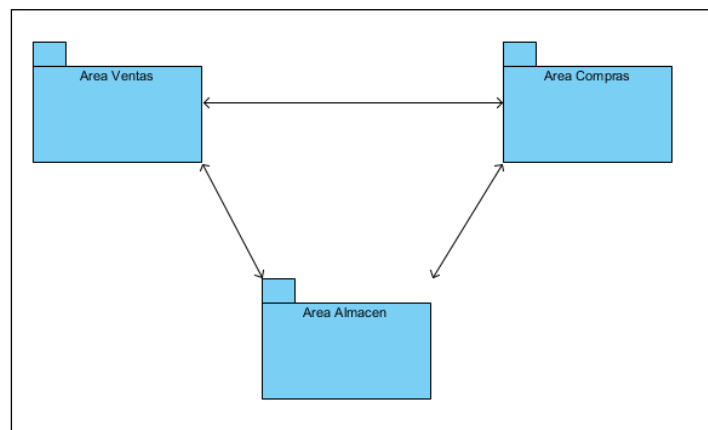


Fig. 4: Diagrama de contexto del Negocio

- **Diagrama casos uso negocio (DCUN)**

- **Compras**

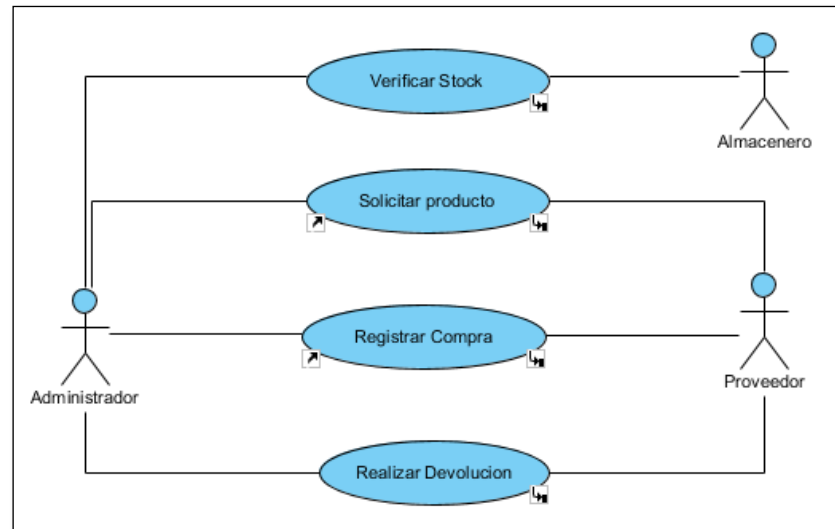


Fig. 5: Diagrama de caso de uso de negocio de compras

○ **Ventas**

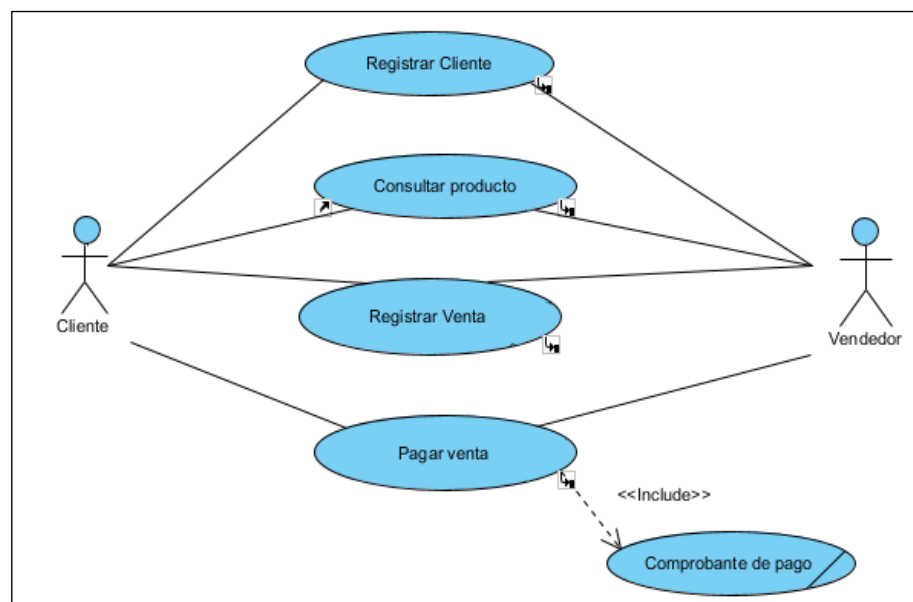


Fig. 6: Diagrama-caso- uso-negocio-de-ventas(DCUNV)

○ **Almacen**

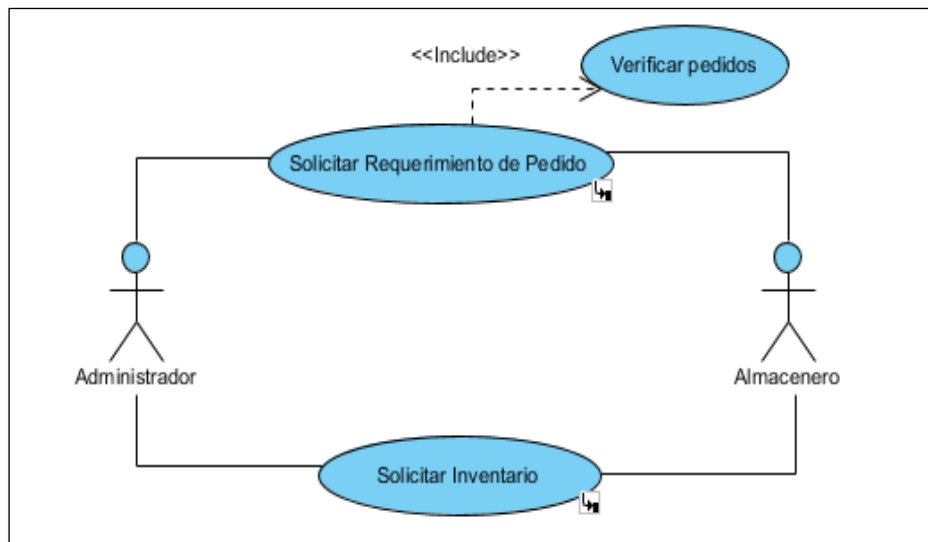


Fig. 7: Diagrama-caso-uso- negocio-de-almacén(DCUNA)

- **Diagrama-de-actividades**
 - **Compras**
 - **Verificar Stock**

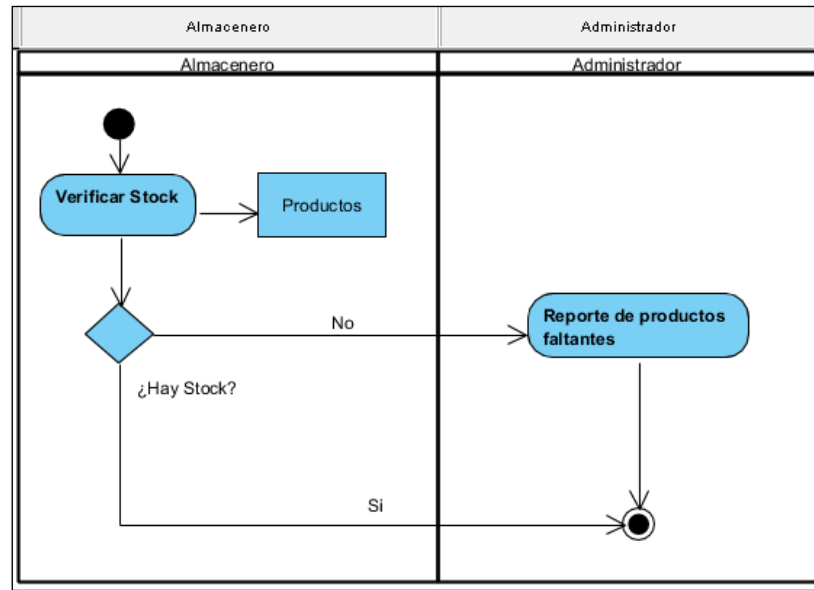


Fig. 8: Diagrama de actividades de verificar stock de compras

- **Solicitar Productos**

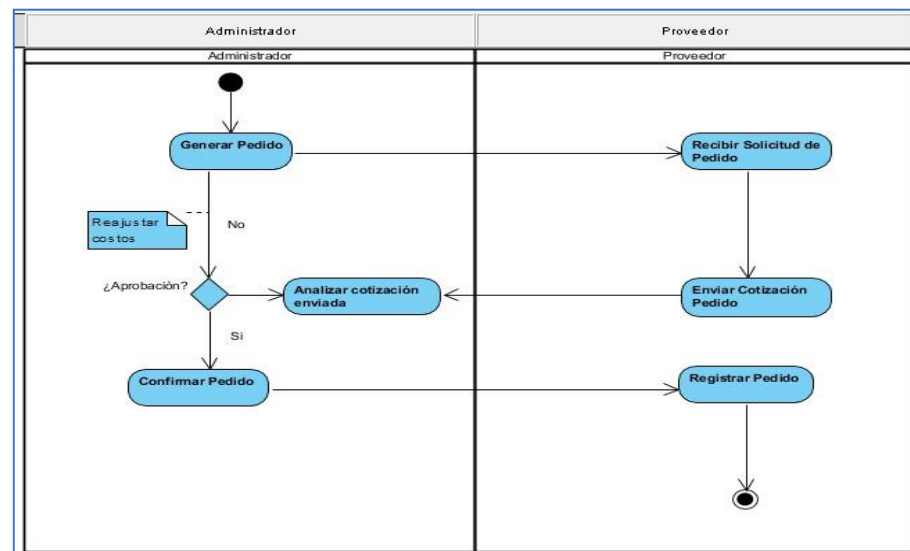


Fig. 9: Diagrama de actividades de solicitar productos de compras

- **Registrar Compra**

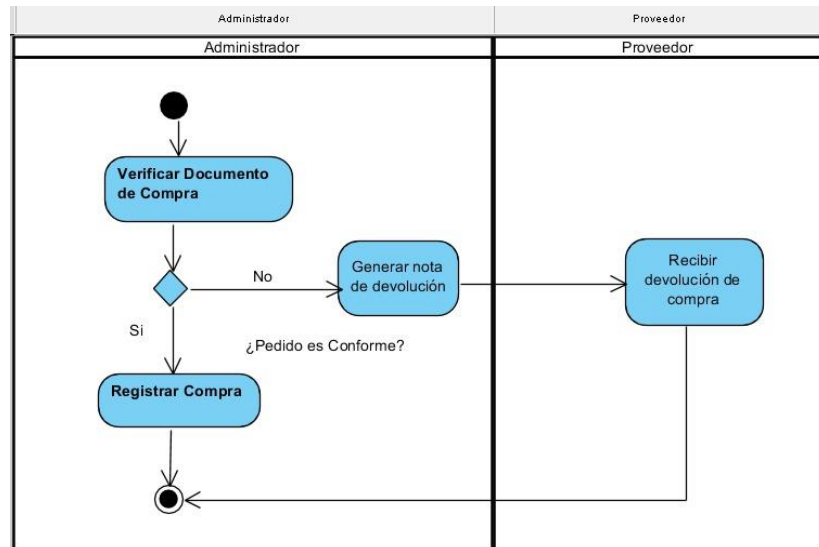


Fig. 10: Diagrama de actividades de registrar compra

- **Realizar Devolución**

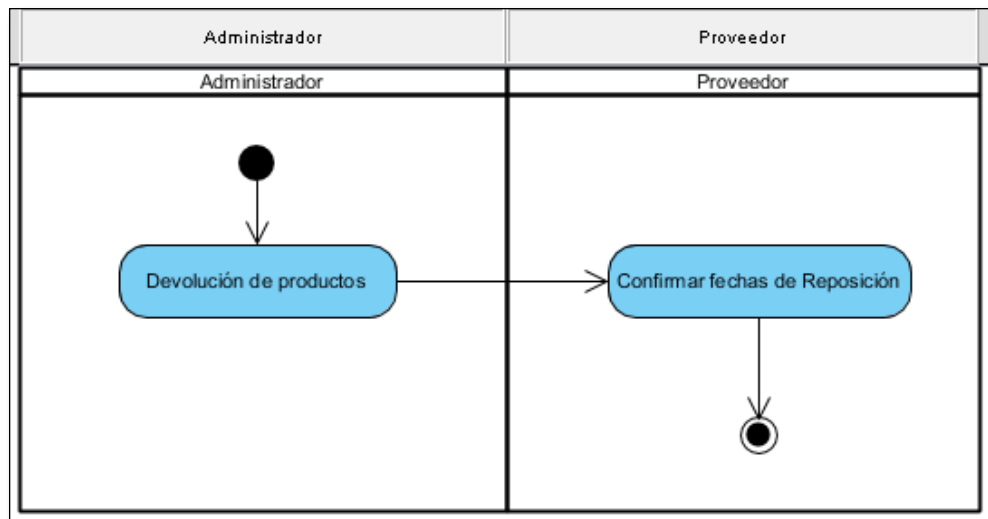


Fig. 11: Diagrama-actividades-realizar-devolución

- **Ventas**

- **Registrar Cliente**

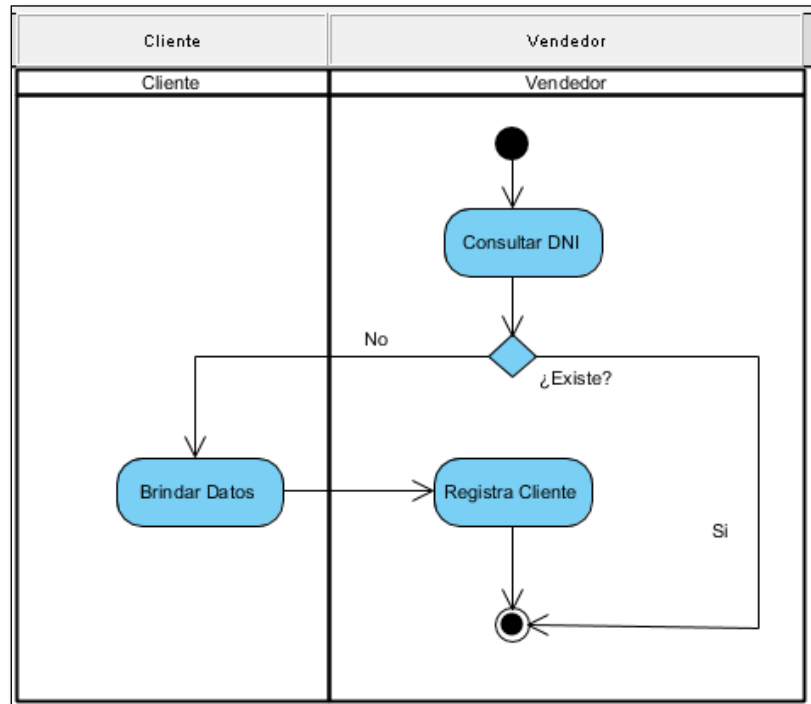


Fig. 12: Diagrama de actividades de registrar cliente de ventas

- **Consultar Producto**

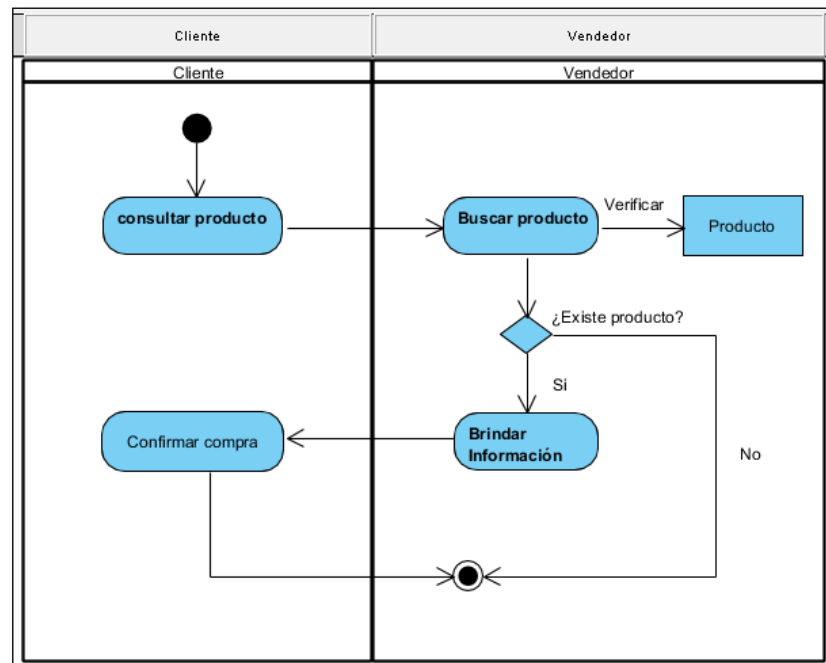


Fig. 13: Diagrama-actividades-consultar-producto-ventas

- **Registro Venta**

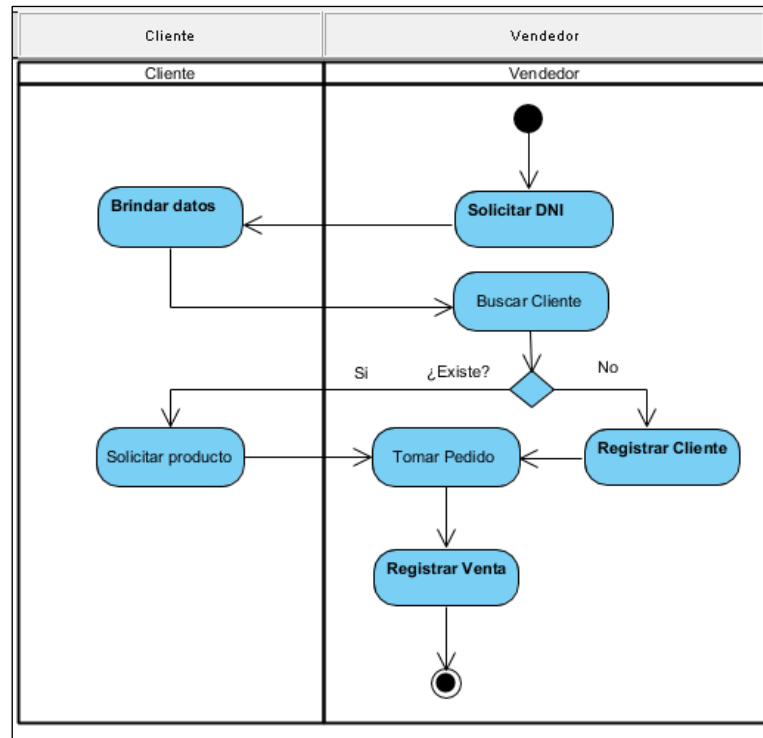


Fig. 14: Diagrama- actividades-registrar- venta-ventas

- **Pagar Venta**

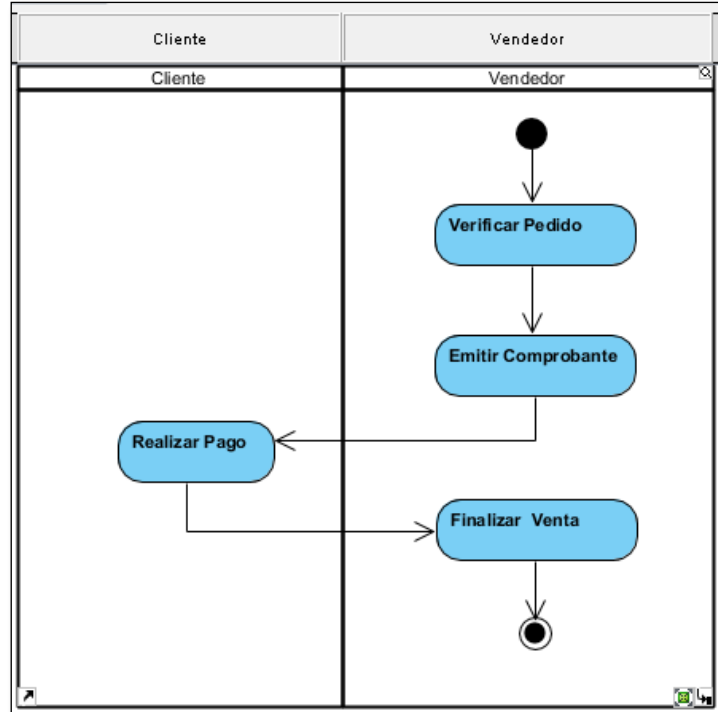


Fig. 15: Diagrama de actividades de pagar venta de ventas

- **Almacen**
- **Requerimiento de Pedidos**

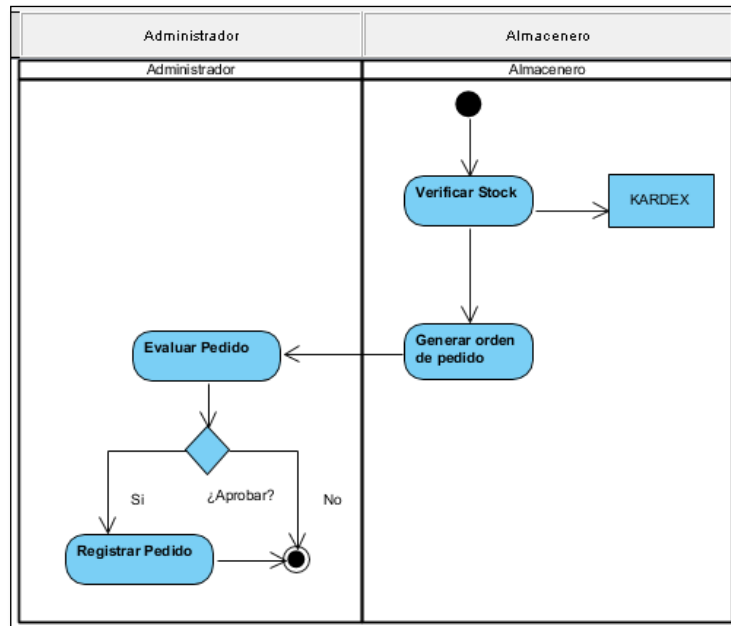


Fig. 16: Diagrama- actividades-requerimiento-pedidos-almacén

○ **Inventario**

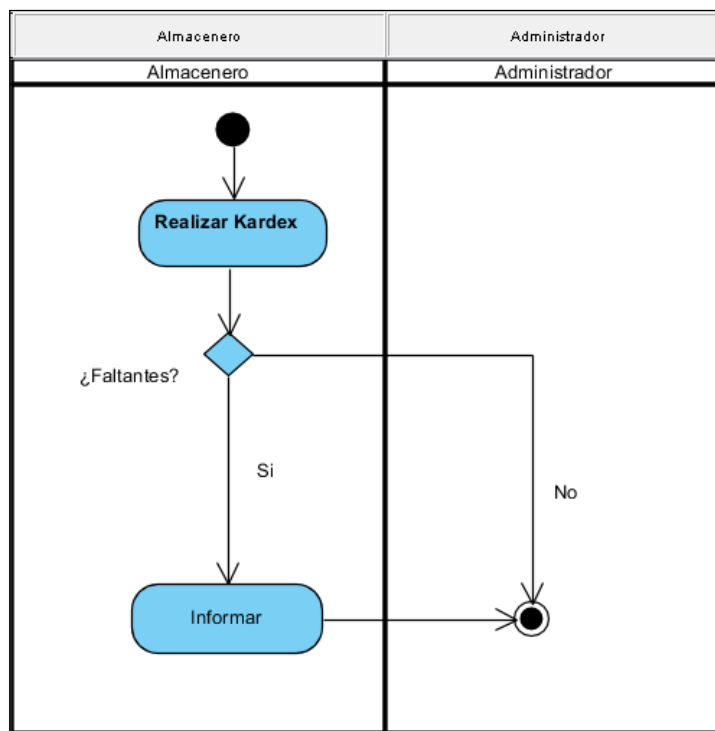


Fig. 17: Diagrama-actividades-requerimiento- inventario

○ **Diagrama / objetos**

○ **Compras**

○ **Verificar Stock**

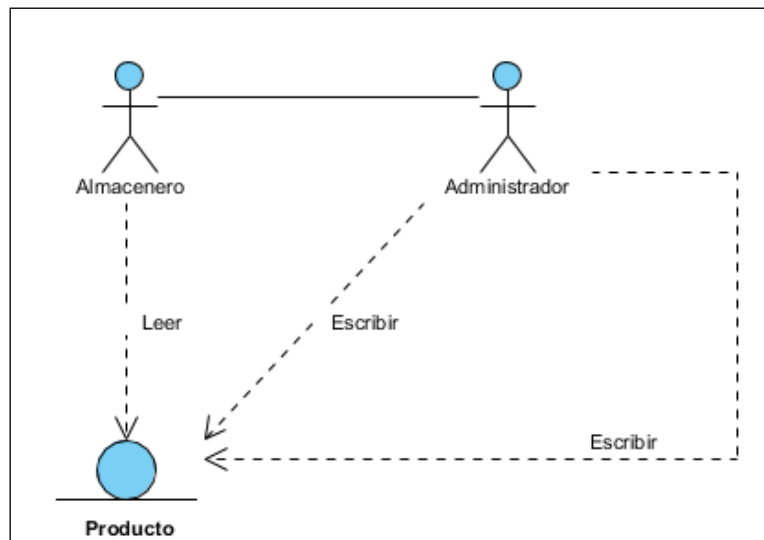


Fig. 18: Diagrama- objetos- verificar- stock

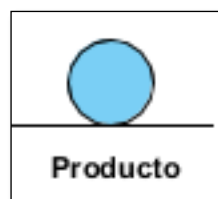


Fig. 19: Diagrama - objetos- parcial - producto de verificar stock

○ **Solicitar Productos**

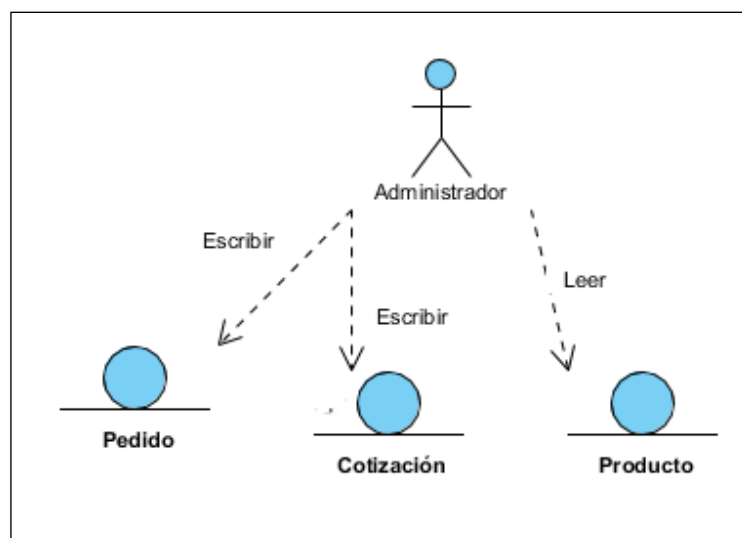


Fig. 20: : Diagrama - objetos- solicitar- productos

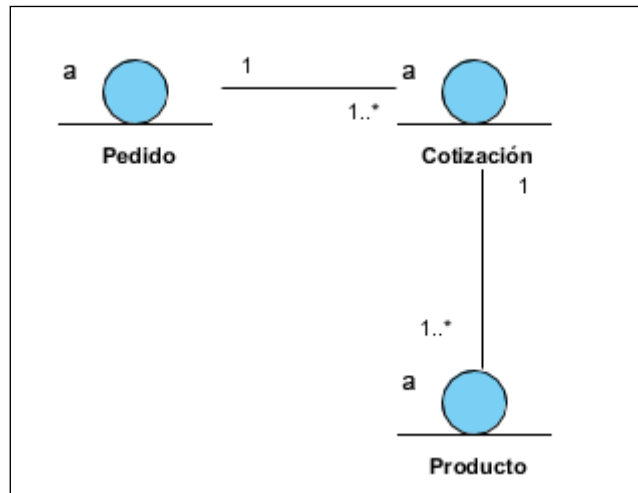


Fig. 21: Diagrama - objetos- parcial - solicitar productos

○ **Registrar Compra - Devolución**

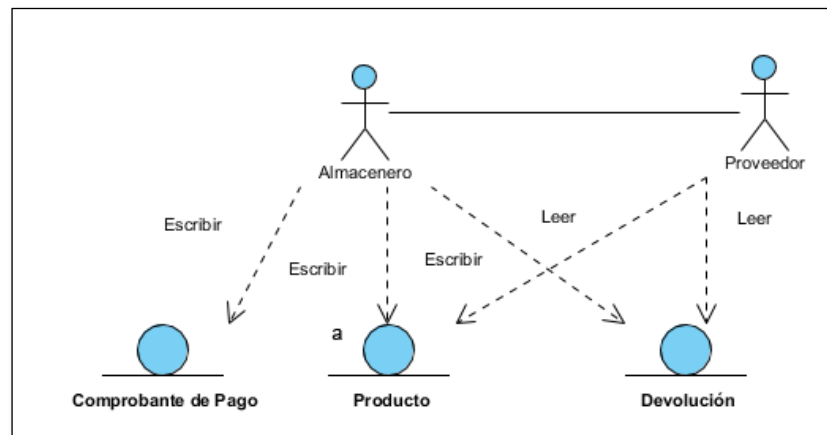


Fig. 22: Diagrama de objetos de verificar y pagar producto de compras y devolución

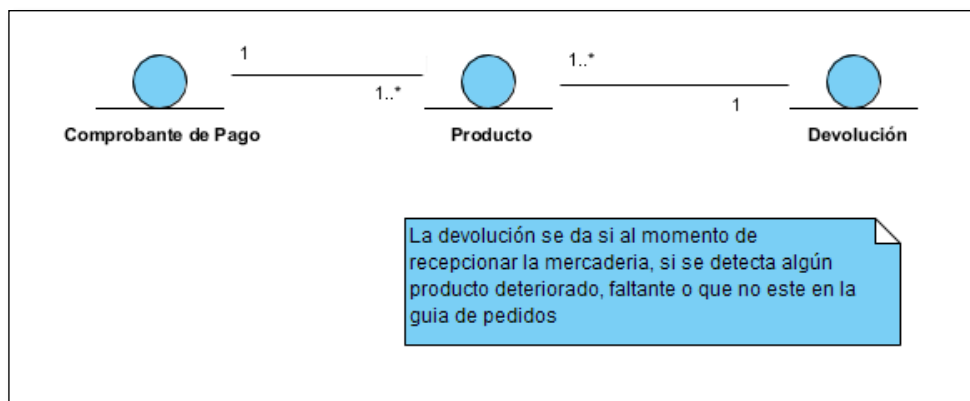


Fig. 23: Diagrama de objetos parcial de verificar y pagar producto de compras y devolución

- **Ventas**

- **Registrar Cliente**

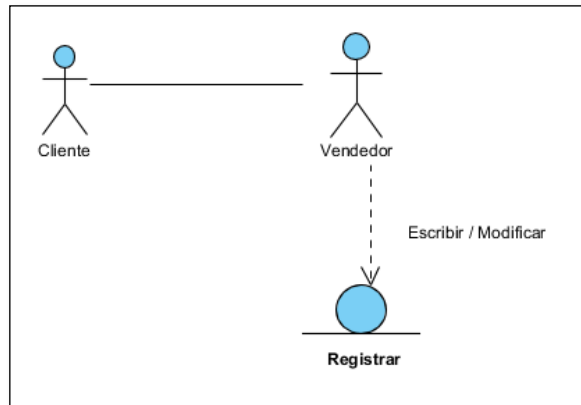


Fig. 24: Diagrama - objetos parcial de registrar cliente

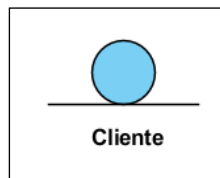


Fig. 25: Diagrama de objetos parcial de registrar cliente

- **Consultar – Producto**

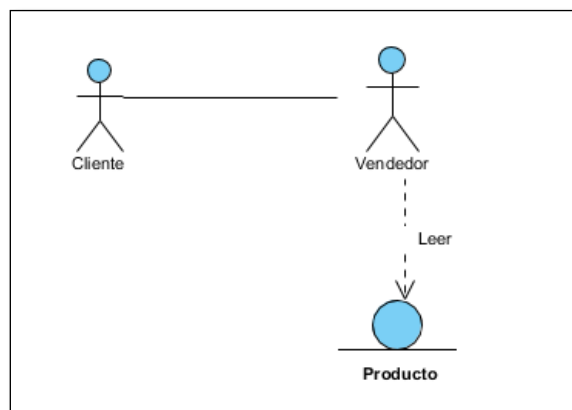


Fig. 26: Diagrama -objetos - consultar- producto

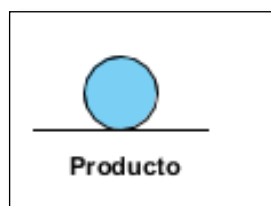


Fig. 27 : Diagrama - objetos - parcial - consultar - producto - ventas

○ Registrar Venta

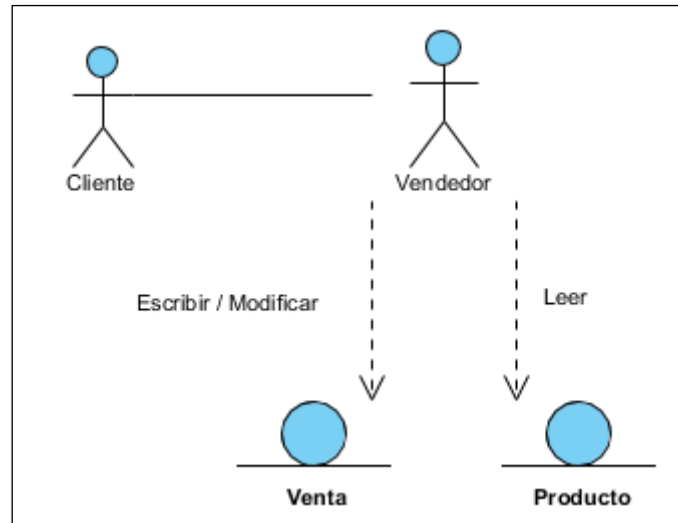


Fig. 28: Diagrama - objetos - registrar ventas

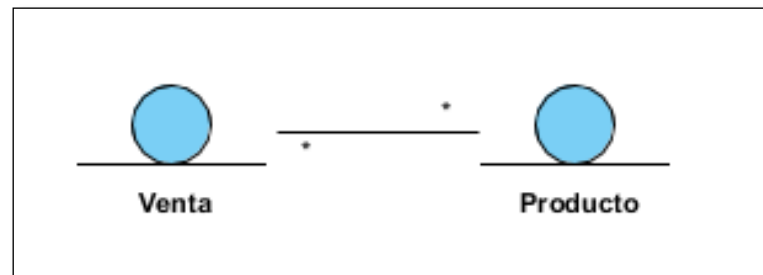


Fig. 29: Diagrama - objetos- parcial - registrar venta

○ Pagar Venta

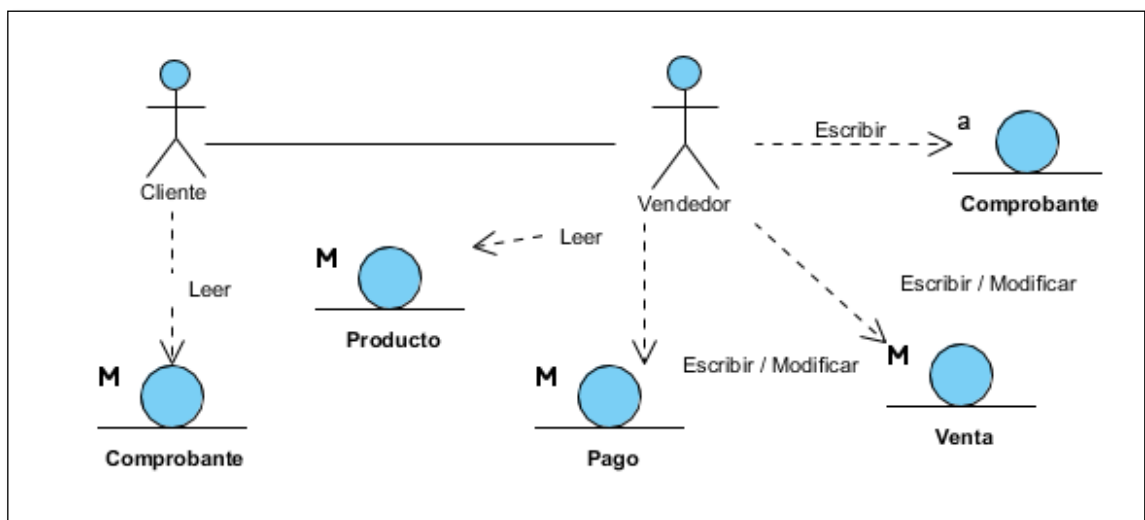


Fig. 30: Diagrama - objetos - pagar ventas

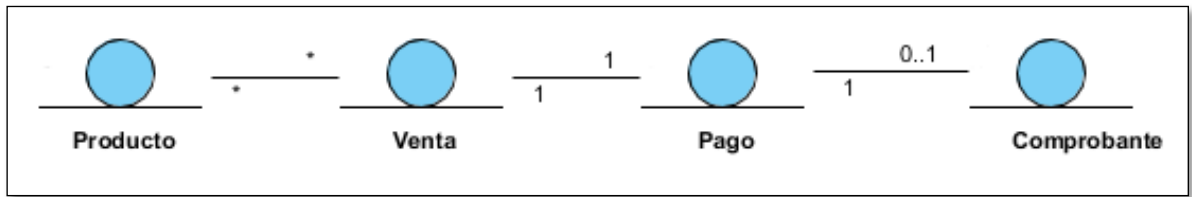


Fig. 31: Diagrama - objetos- parcial - pagar ventas.

○ Almacén

○ Requerimiento de Pedidos

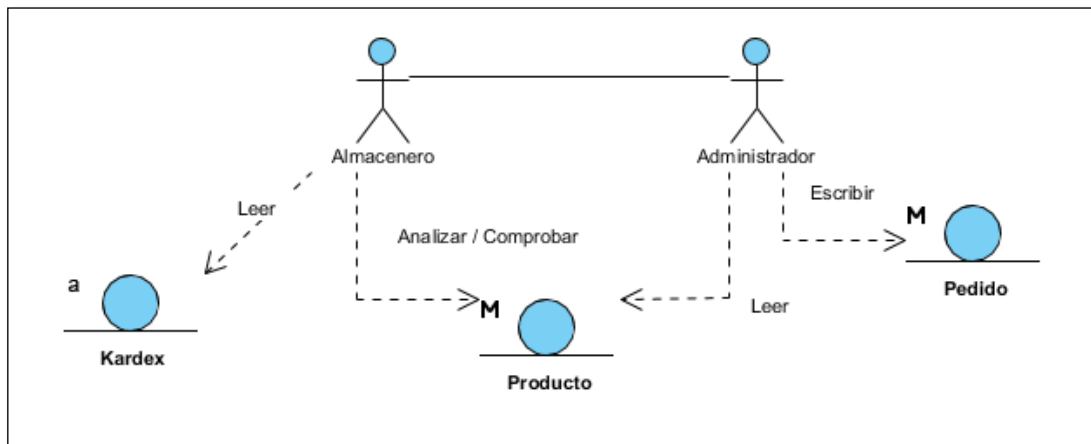


Fig. 32: Diagrama -objetos - requerimiento - pedidos.

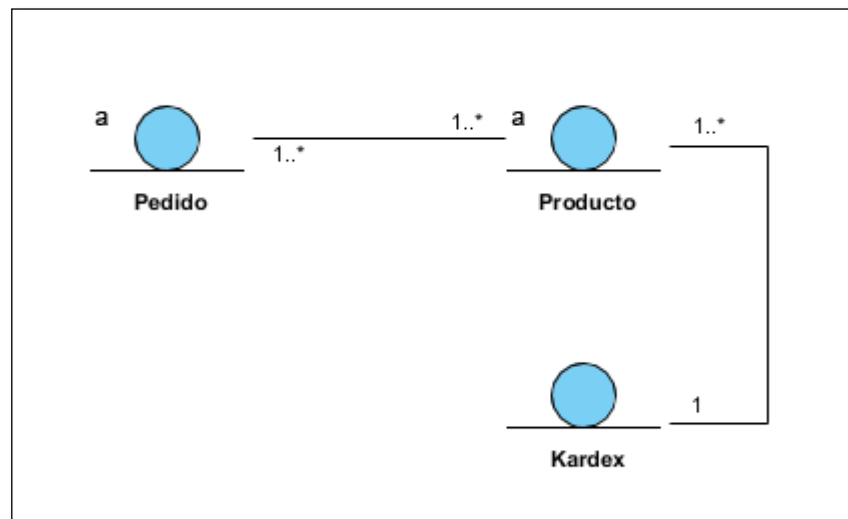


Fig. 33: Diagrama - objetos- parcial - requerimiento de pedidos

○ **Inventario**

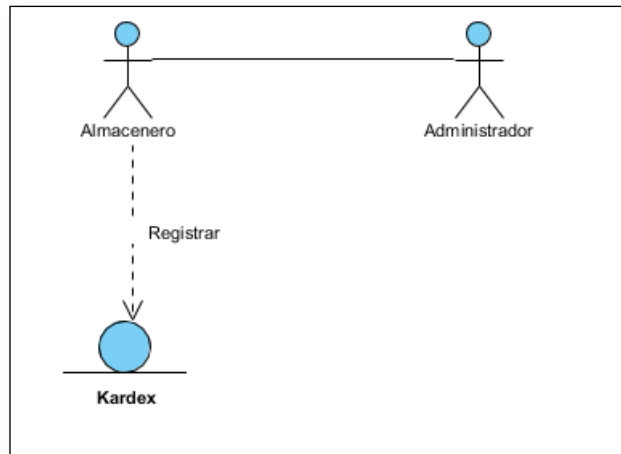


Fig. 34: Diagrama - objetos -inventario.



Fig. 35: Diagrama de objeto parcial de inventario.

○ **Modelo del dominio**

• **Compras**

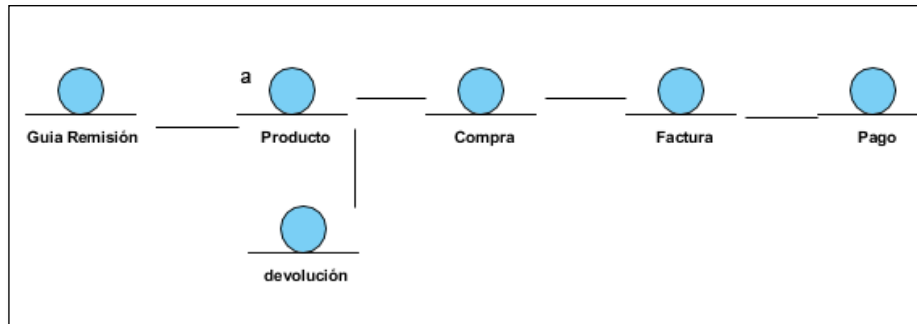


Fig. 36: Diagrama de objetos general de compras.

- **Venta**

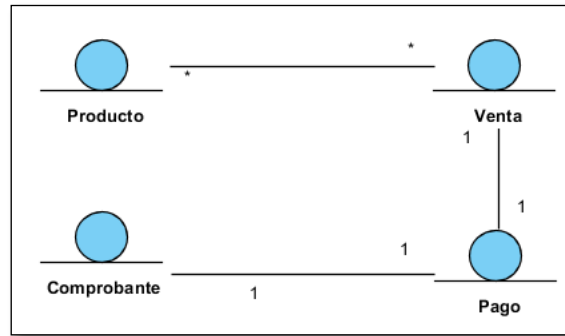


Fig. 37: Diagrama - objetos general - ventas.

- **Almacén**

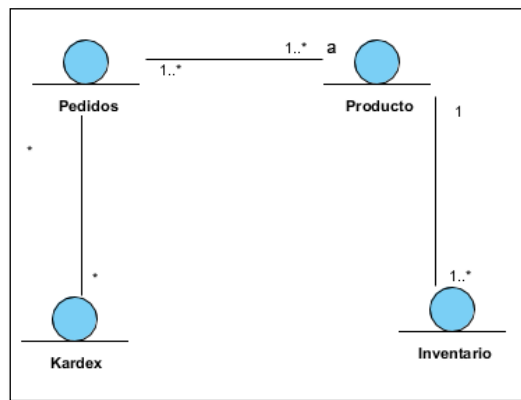


Fig. 38: Diagrama de objetos general de almacén.

4.1.3. Iteración #3: Análisis Preliminar-Requerimientos / Casos-Usos

En esta iteración se desarrollaron las siguientes actividades:

- **Requerimientos**
 - **Diagrama / Contexto**

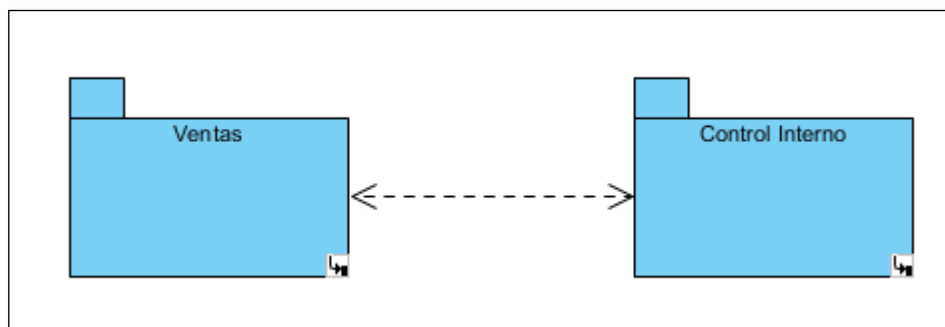


Fig. 39: Diagrama - contexto - requerimientos.

○ **Diagrama / casos /uso**

○ **Ventas**

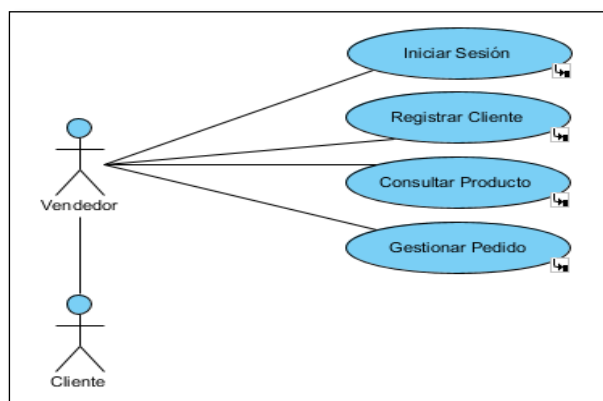


Fig. 40: Diagrama -caso-uso- Toma-pedidos.

Tabla 20: MÓDULO INICIAR SESIÓN VENTAS

Nombres de caso: Inicio de Sesión	01
Objetivo: Interfaz de inicio de sesión	
Actor Principal: Vendedor	
Precondición: inicio de sesión vendedor.	
Flujo Principal: 1. El usuario ingresará datos de Login y contraseña. 2. La aplicación verifica si el usuario existe. 3. En caso estar exista, la aplicación permitirá al usuario mostrar los módulos correspondientes; caso contrario mostrar mensaje usuario no existe.	
Flujo Alternativo: Verificar conexión a internet.	
Post Condición: - Debe iniciar sesión correctamente. - Restringir acceso si no está autorizado.	

FUENTE: ELABORACIÓN PROPIA

Tabla 21: MÓDULO REGISTRAR NUEVO CLIENTE.

Nombres de caso: Registrar Cliente	02
Objetivo: Registrar Nuevo Cliente.	
Actor Principal: Vendedor	
Precondición: cliente no existente, inicio de sesión vendedor.	
Flujo Principal: 1. El usuario ingresará datos de Login y contraseña e ingresará al módulo. 2. Si el cliente no existe se registrar los siguientes datos: Nro. Documento (DNI o RUC) 3. La aplicación verifica que los datos se ingresan correctamente. 4. Presionar el botón para registrar cliente correctamente.	
Flujo Alternativo: Verificar conexión a internet.	
Post Condición: - Registrar el cliente correctamente	

FUENTE: ELABORACIÓN PROPIA

Tabla 22: MÓDULO CONSULTAR PRODUCTO DE VENDEDOR

Nombres de caso: Consultar producto	03
Objetivo: Consulta del Producto.	
Actor Principal: Vendedor	
Precondición: El producto debe estar registrado, inicio de sesión vendedor.	
Flujo Principal: 1. El usuario ingresa datos de Login y contraseña e ingresará al módulo. 2. ingresar en la caja de texto el Nombre del Producto el cual filtrará la información de la lista. 3. Seleccionar el Producto y presionar enter.	
Flujo Alternativo: Verificar conexión a internet.	
Post Condición: - Realizar la consulta y filtrar automáticamente.	

FUENTE: ELABORACIÓN PROPIA

Tabla 23: MÓDULO REALIZAR GESTIÓN DE PEDIDO

Nombres de caso: Gestionar Pedido	04
Objetivo: Realizar la gestión Pedido del Cliente	
Actor Principal: Vendedor	
Precondición: Tener registrado al cliente, stock de producto, inicio de sesión vendedor.	
Flujo Principal: 1. El vendedor inicia sesión en la aplicación. 2. verifica que el cliente este registrado en la BD de lo contrato registra NUEVO Cliente. 3. Selecciona el producto. 4. Verifica Stock e ingresa cantidad. 5. Guarda Pedido.	
Flujo Alternativo: Verificar conexión a internet.	
Post Condición: - No debería permitir generar la venta si no hay stock. - El registro de la gestion debe guardarse correctamente.	

FUENTE: ELABORACIÓN PROPIA

○ **Control Interno**

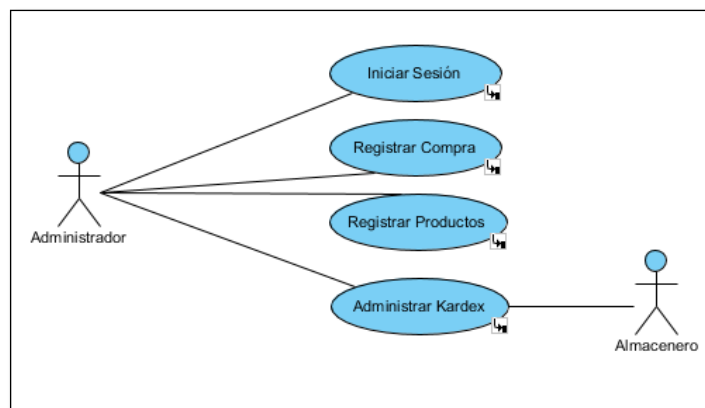


Fig. 41: Diagrama de caso de uso de control interno

Tabla 24: MÓDULO INICIAR SESIÓN CONTROL INTERNO

Nombres de caso: Inicio de Sesión	01
Objetivo: Interfaz de inicio de sesión	
Actor Principal: Administrador	
Precondición: inicio de sesión Administrador.	
Flujo Principal: 1. El usuario ingresará datos de Login y contraseña. 2. La aplicación verifica si el usuario existe. 3. En caso estar exista, la aplicación permitirá al usuario mostrar los módulos correspondientes; caso contrario mostrar mensaje usuario no existe.	
Flujo Alternativo: Verificar conexión a internet.	
Post Condición: - Debe iniciar sesión correctamente. - Restringir acceso si no está autorizado.	

FUENTE: ELABORACIÓN PROPIA

Tabla 25: MÓDULO REGISTRAR COMPRA

Nombres de caso: Registrar Compra	02
Objetivo: Registrar Nuevo Compra.	
Actor Principal: Administrador	
Precondición: cliente no existente, inicio de sesión administrador.	
Flujo Principal: 1. El usuario ingresará datos de Login y contraseña e ingresará al módulo. 2. Registrará los datos del proveedor 3. Selecciona productos. 4. Presionar el botón registrar compra correctamente.	
Flujo Alternativo: Verificar conexión a internet.	
Post Condición: - Registrar el compra correctamente	

FUENTE: ELABORACIÓN PROPIA

Tabla 26: MÓDULO REGISTRAR VENTA.

Nombres de caso: Registrar Venta	02
Objetivo: Registrar Nuevo Venta.	
Actor Principal: Administrador	
Precondición: cliente no existente, inicio de sesión administrador.	
Flujo Principal: 1. El usuario ingresará datos de Login y contraseña e ingresará al módulo. 2. Selecciona cliente 3. Selecciona productos. 4. Presionar el botón registrar venta correctamente.	
Flujo Alternativo: Verificar conexión a internet.	
Post Condición: - Registrar venta correctamente	

FUENTE: ELABORACIÓN PROPIA

Tabla 27: MÓDULO ADMINISTRAR KARDEX

Nombres de caso: Kardex	02
Objetivo: Visualizar Kardex.	
Actor Principal: Almacenero	
Precondición: existencia de productos, existencia de ventas, inicio de sesión Almacenero.	
Flujo Principal: 1. El usuario ingresará datos de Login y contraseña e ingresará al módulo. 2. Selecciona producto 3. Visualiza el inventario con el método PEPS. 4. Presionar el producto se exportara un Excel con la implementario de PEPS del producto.	
Flujo Alternativo: Verificar conexión a internet.	
Post Condición: - Visualizar Kardex de Productos correctamente	

FUENTE: ELABORACIÓN PROPIA

○ **Diagrama de Actividades**

● **Ventas**

● **Iniciar Sesión**

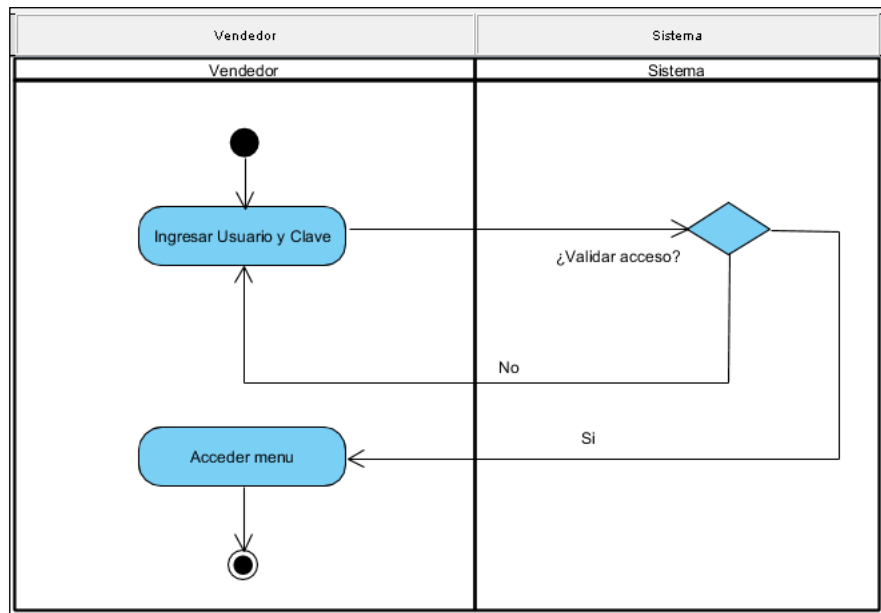


Fig. 42: Diagrama de actividades de iniciar sesión

- **Registrar Cliente**

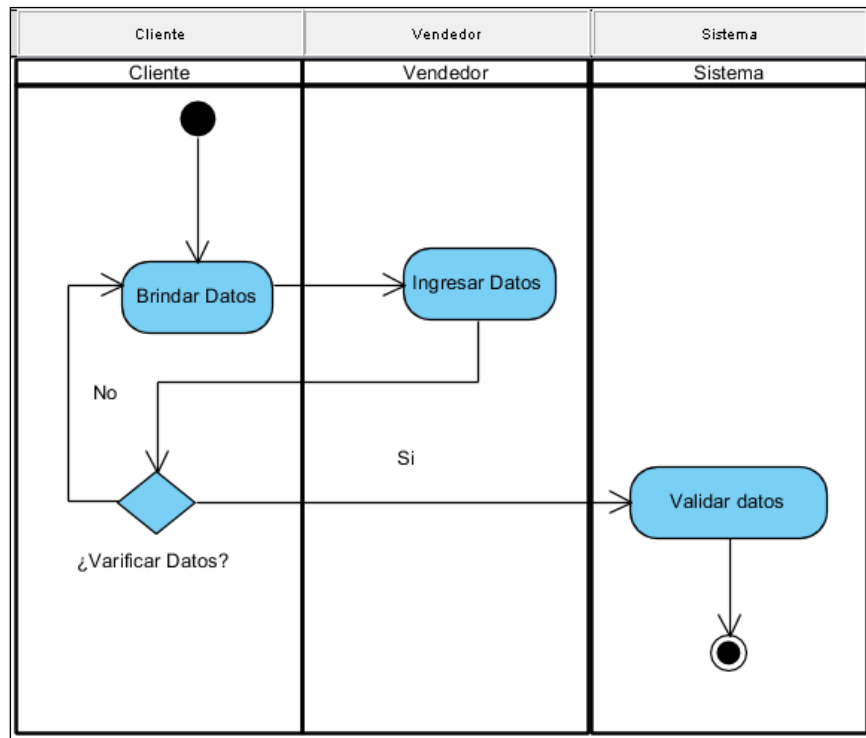


Fig. 43: Diagrama de actividades de registrar cliente de toma pedidos

- **Consultar Producto**

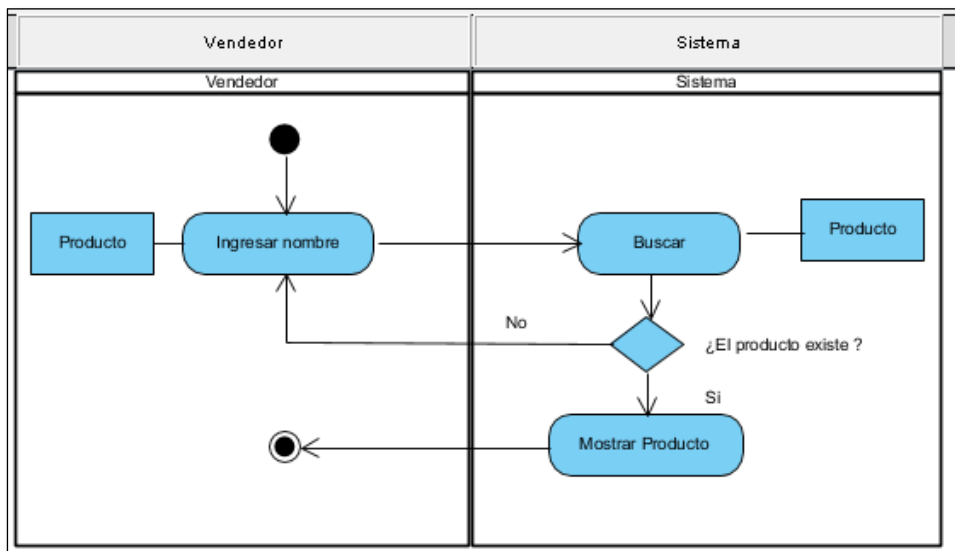


Fig. 44: Diagrama de actividades de consultar producto de toma pedidos

- **Gestionar Pedido**
 - **Venta**

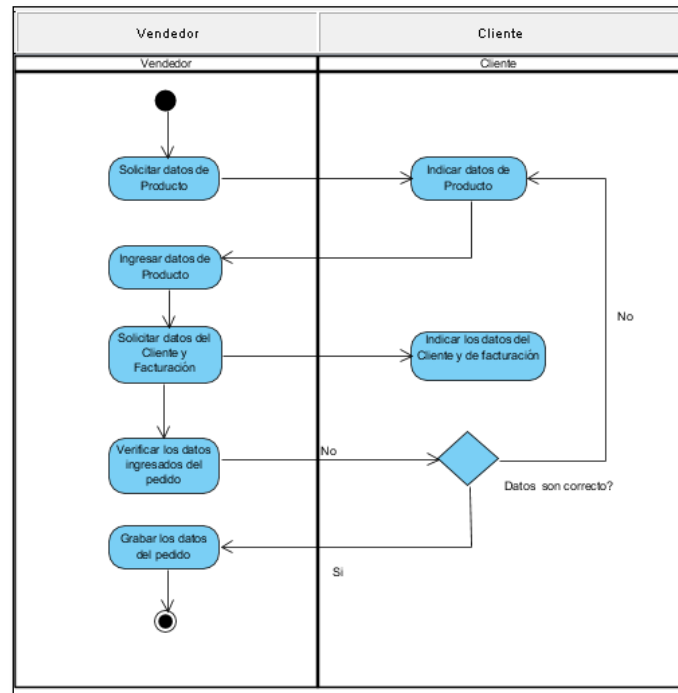


Fig. 45: Diagrama de actividades de venta de gestionar venta

- **Modificar Pedido**

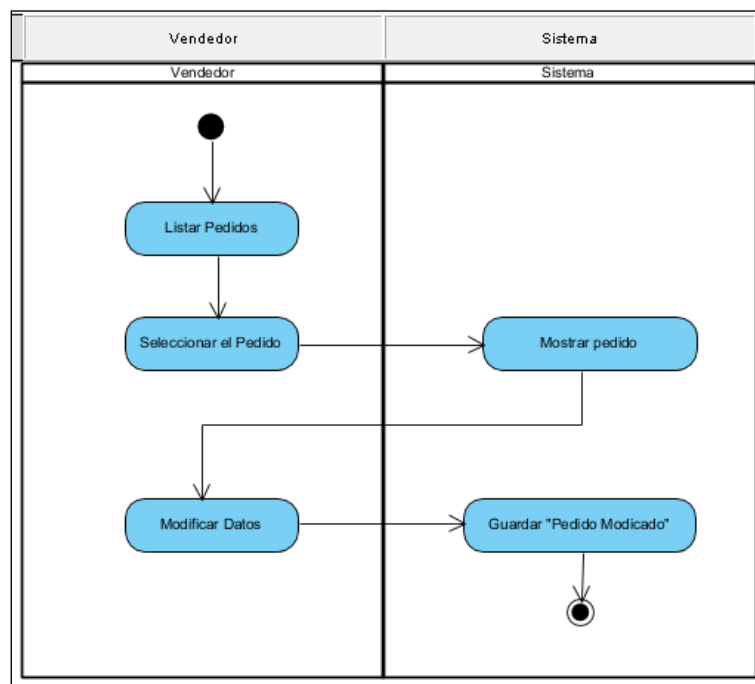


Fig. 46: Diagrama de actividades de modificar venta de gestionar venta

- **Rechazar Pedido**

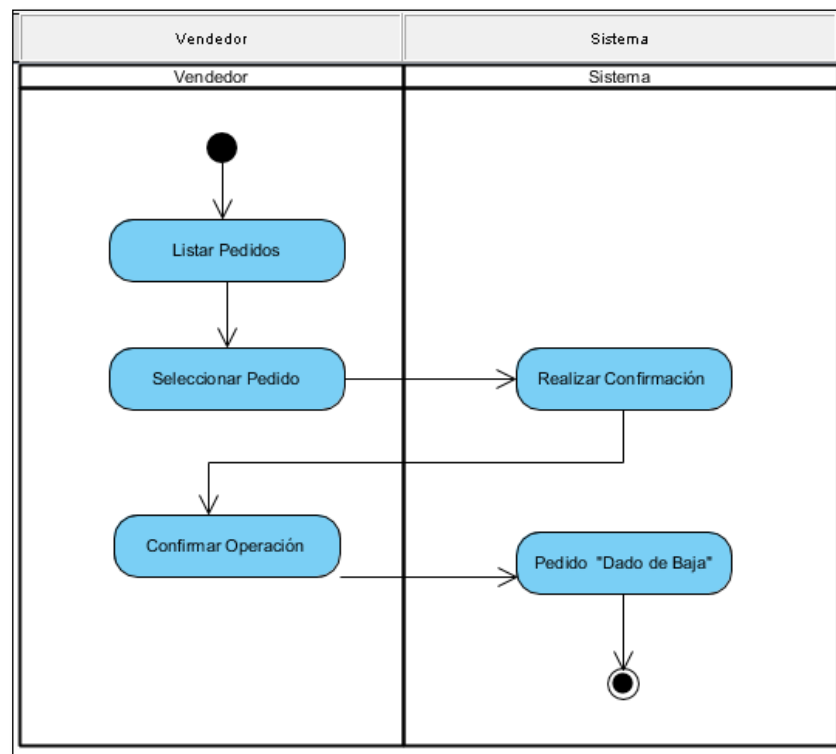


Fig. 47: Diagrama de actividades Rechazar pedido

- **Control Interno**

- **Iniciar Sesión**

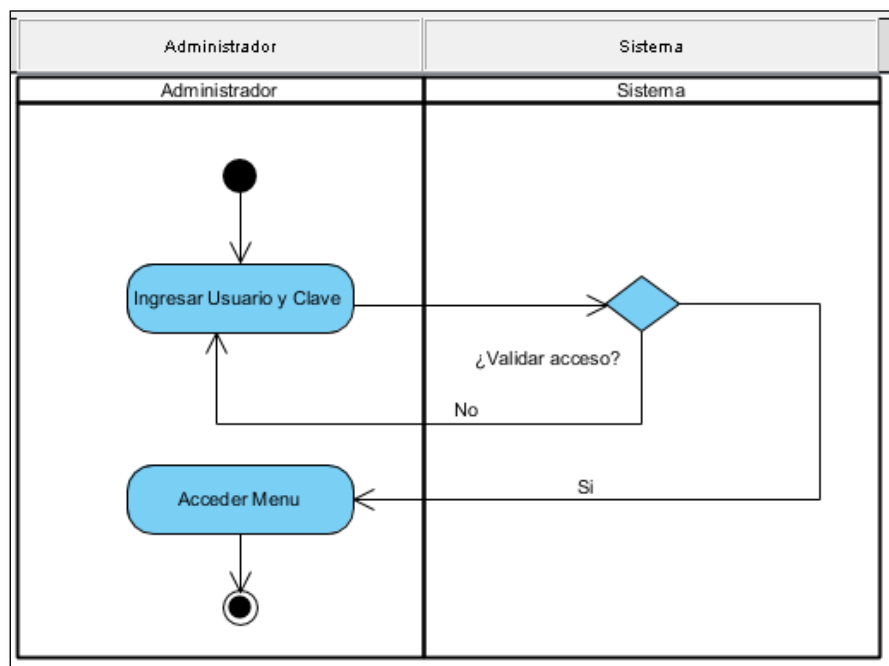


Fig. 48: Diagrama de actividades de iniciar sesión de control interno

- **Registrar Compra**

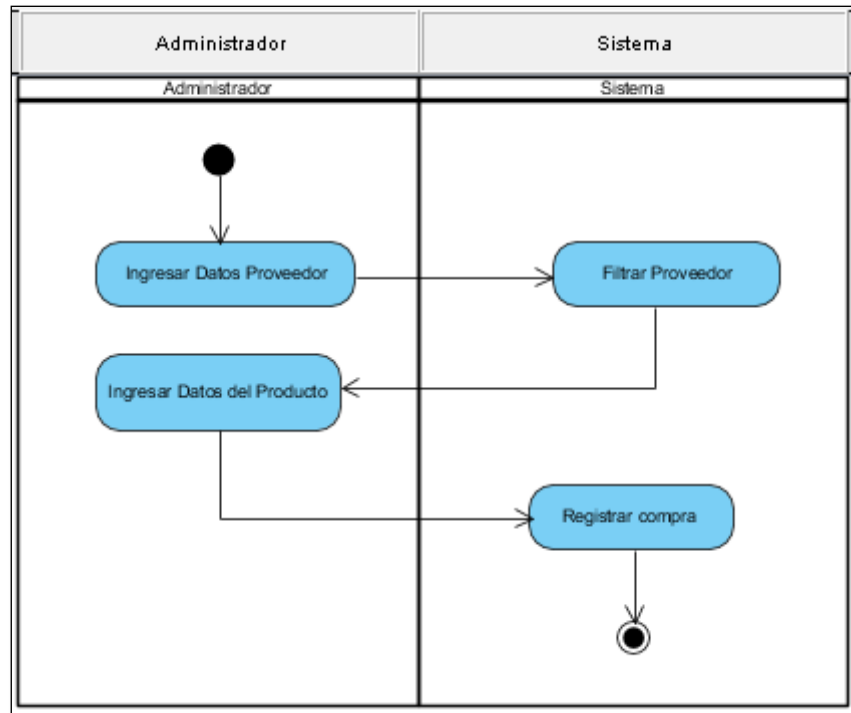


Fig. 49: Diagrama de actividades de registrar compra de control interno

- **Registrar Producto**

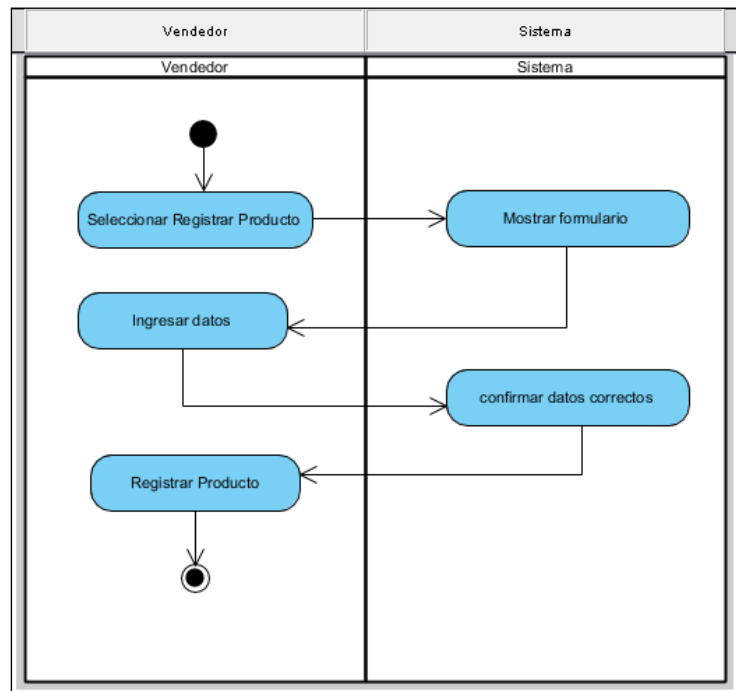


Fig. 50: Diagrama de actividades de registrar producto de control interno.

- **Administrar Kardex**

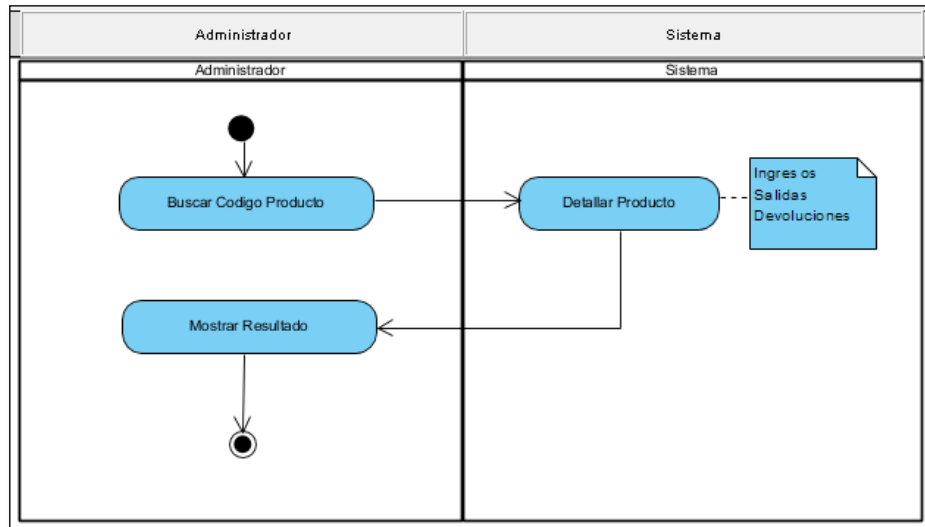


Fig. 51: Diagrama de actividades de kardex de administrar kardex

- **Diagrama de Objetos Clases Parciales**

- **Venta**

- **Inicio Sesión**

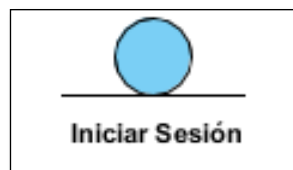


Fig. 52: : Diagrama de objetos parcial de iniciar sesión de ventas

- **Registrar Cliente**



Fig. 53: : Diagrama de objetos parcial de registrar cliente de ventas

- **Consultar Producto**

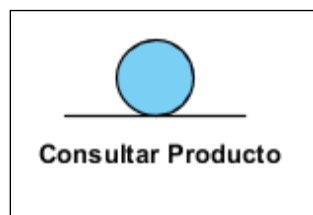


Fig. 54: Diagrama de objetos parcial de consultar producto de ventas

- **Gestionar Pedido**

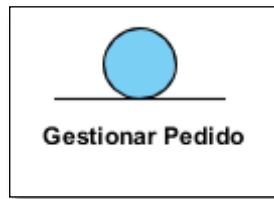


Fig. 55: Diagrama de objetos parcial de gestionar pedido de ventas

- **Control Interno**

- **Iniciar Sesión**

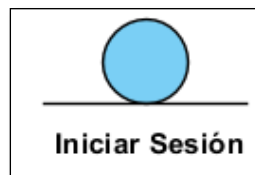


Fig. 56: Diagrama de objetos parcial de iniciar sesión de control interno

- **Registrar Compra**

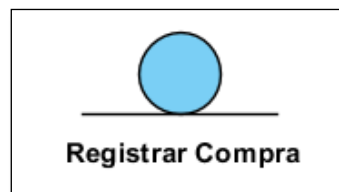


Fig. 57: Diagrama de objetos parcial de registrar compra de control interno

- **Registrar Producto**



Fig. 58: : Diagrama de objetos parcial de registrar producto de control interno.

- **Administrar Kardex**



Fig. 59: Diagrama de objetos parcial de administrar kardex de control interno.

- **Diagrama de Objetos General**

- **Ventas**

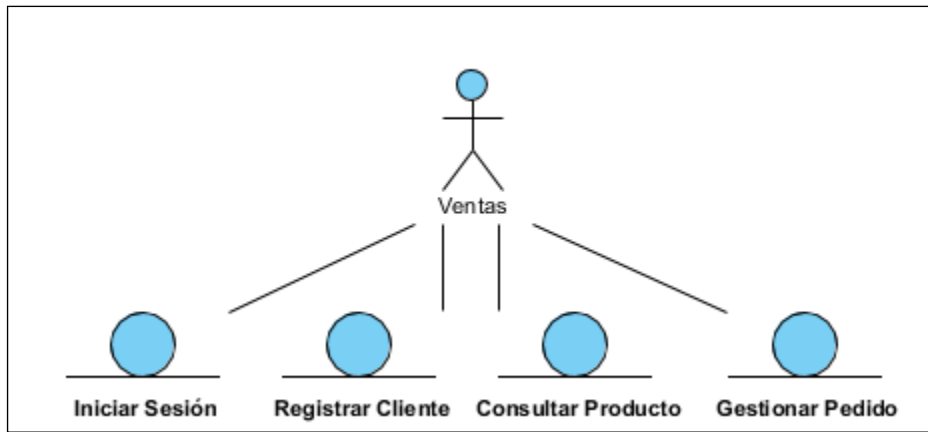


Fig. 60: Diagrama de objetos general de venta.

- **Control Interno**

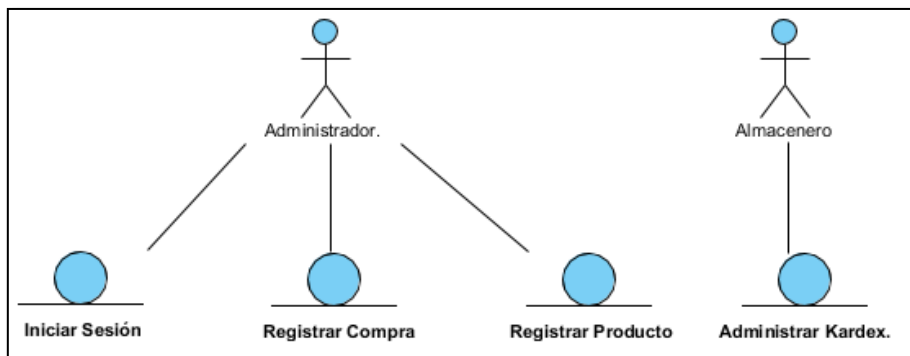


Fig. 61: Diagrama de objetos general de control interno

4.1.4. Iteración #4: Análisis

- **Paquete De Análisis**

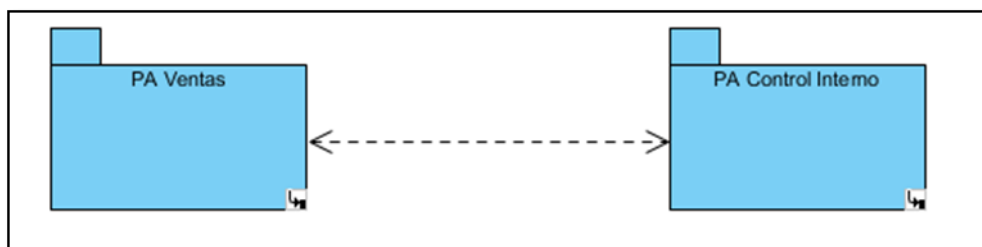


Fig. 62: Diagrama de paquetes de análisis

○ Diagrama de realización de casos de uso de análisis

○ PAventas

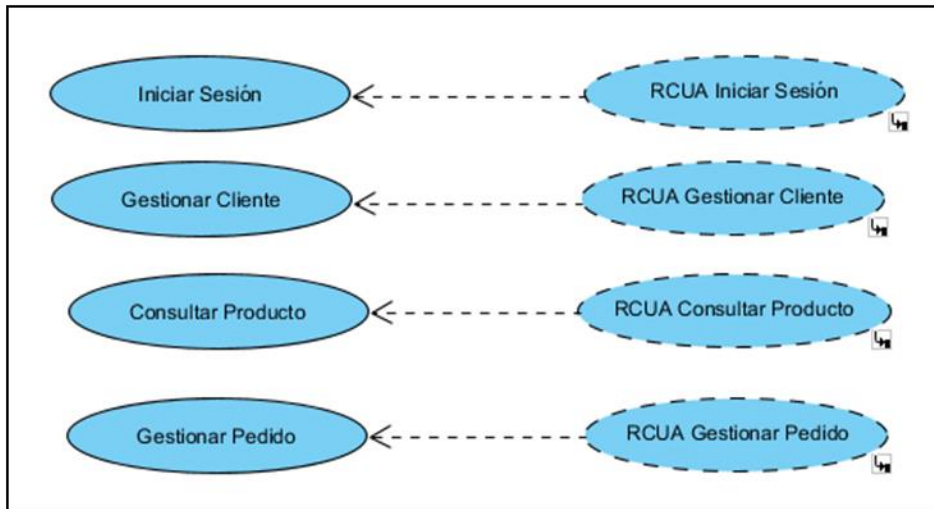


Fig. 63: Diagrama de caso de uso de análisis de ventas

○ PAcontrol Interno

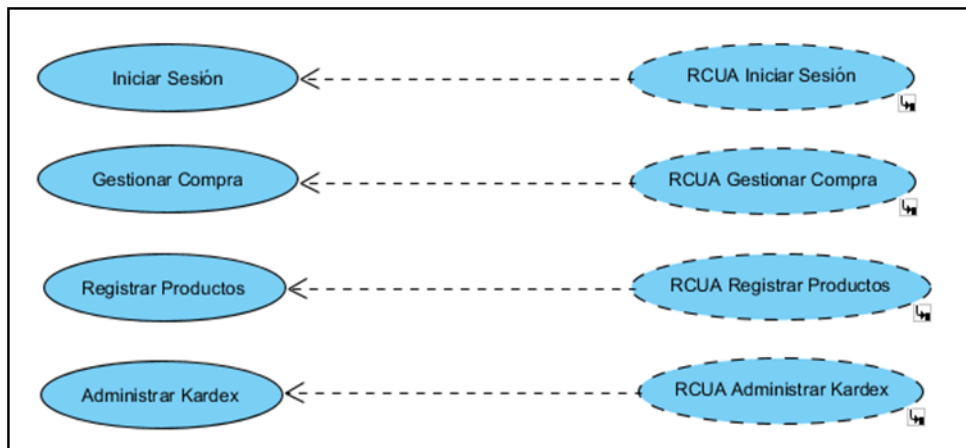


Fig. 64: Diagrama de caso de uso de análisis de PA control interno

Diagrama de clases de análisis

○ PAventas

● Inicio Sesión

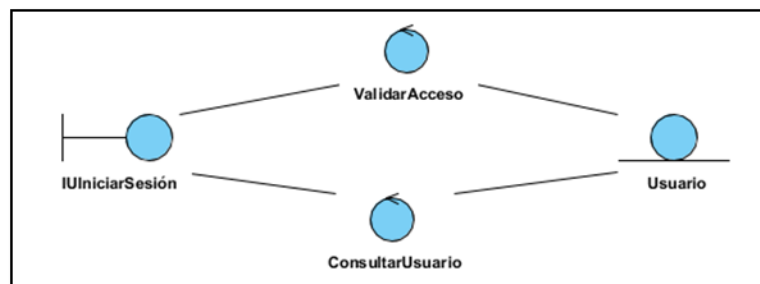


Fig. 65: Diagrama de clases de análisis de inicio de sesión de PAventas

Gestionar Cliente

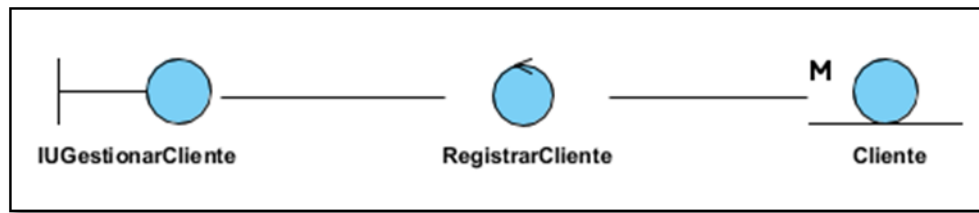


Fig. 66: Diagrama de clases de análisis de Registrar cliente de PAventas

- Consultar Producto

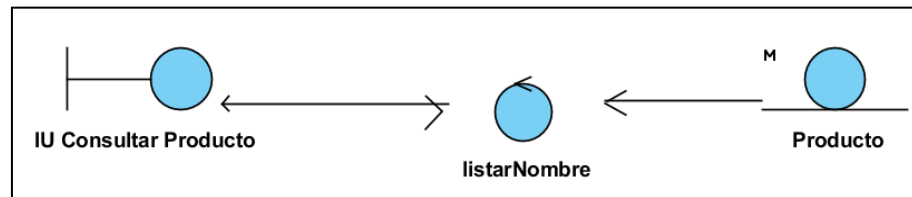


Fig. 67: Diagrama de clases de análisis de consultar producto de PAventas

- Gestionar Pedido

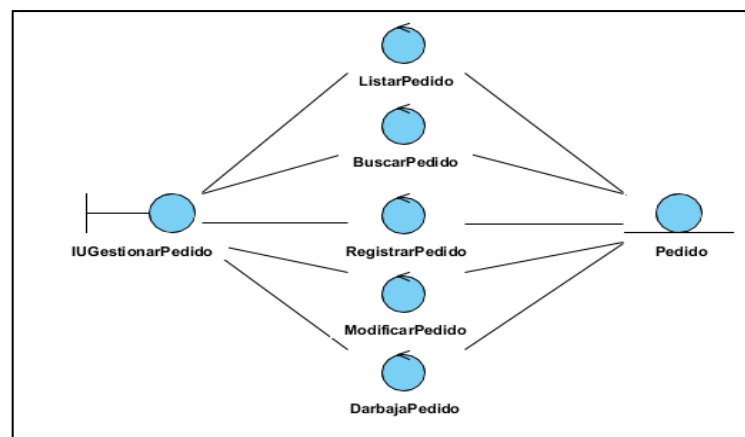


Fig. 68: Diagrama de clases de análisis de gestionar pedido de toma pedidos

- Control Interno

- Iniciar Sesión

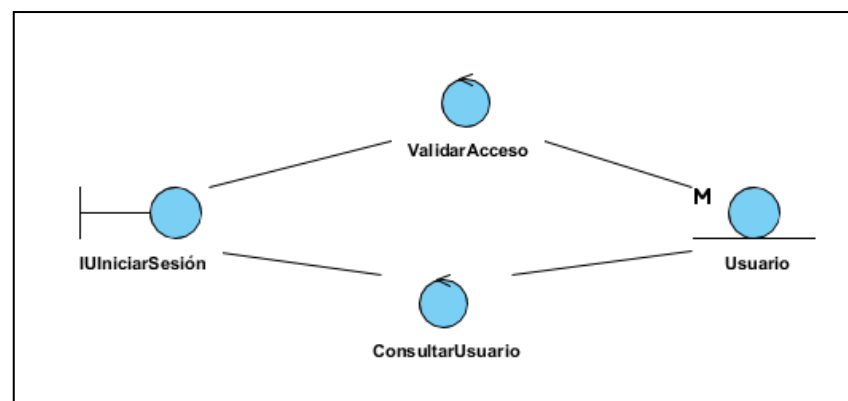


Fig. 69: Diagrama de clases de análisis de iniciar sesión de PA control interno

- **Gestionar Compra**

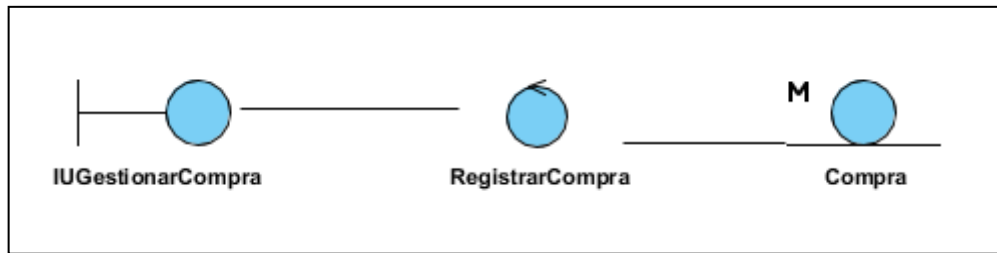


Fig. 70: Diagrama de clases de análisis de crear reparto de PA control interno

- **Gestionar Producto**

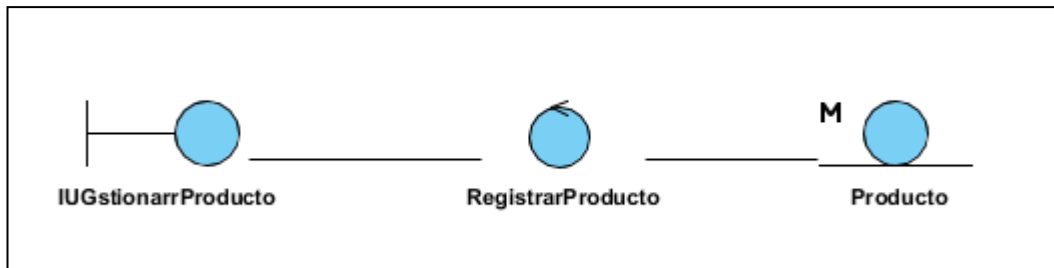


Fig. 71: Diagrama de clases de análisis de Gestionar producto de PA control interno

- **Administrar Kardex**

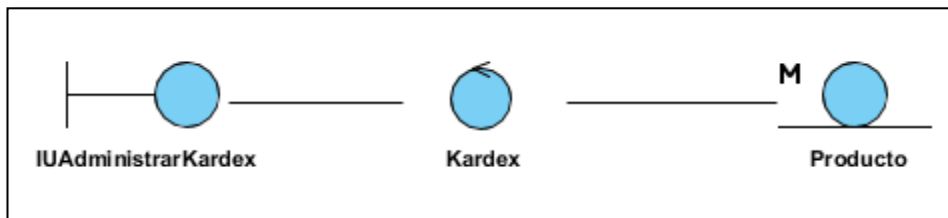


Fig. 72: Diagrama de clases de análisis de administrar kardex de PA Control interno

- **Diagr. de Clases Parciales**

- **Ventas**

- **Inicio Sesión**

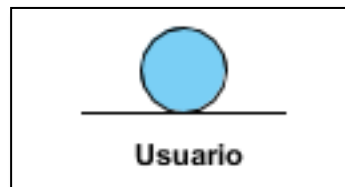


Fig. 73: : Diagrama de clases parcial iniciar sesión de ventas.

- **Gestionar Cliente**

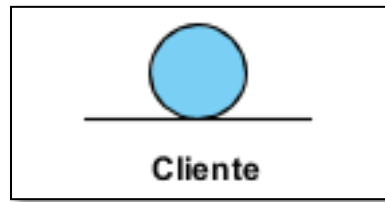


Fig. 74: Diagrama de clases parcial gestionar cliente de ventas

- **Consultar Producto**

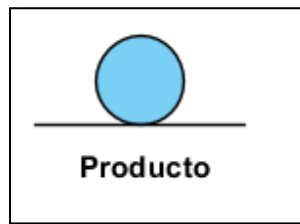


Fig. 75: Diagrama de clases parcial consultar producto de ventas

- **Gestionar Pedido**

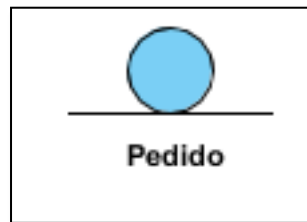


Fig. 76: : Diagrama de clases parcial gestionar pedido de ventas

- **Control Interno**

- **Iniciar Sesión**



Fig. 77: Diagrama de clases parcial iniciar sesión de control interno

- **Gestionar Compra**

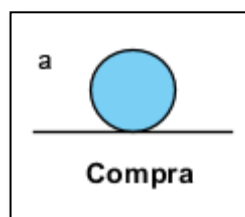


Fig. 78: Diagrama de clases parcial gestionar compra de control interno

- **Gestionar Producto**

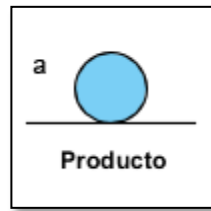


Fig. 79: Diagrama de clases parcial gestionar producto de control interno.

- **Administrar Kardex**

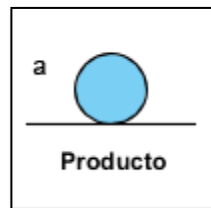


Fig. 80: Diagrama de clases parcial administrar kardex de control interno.

- **Diagrama de Clases General**

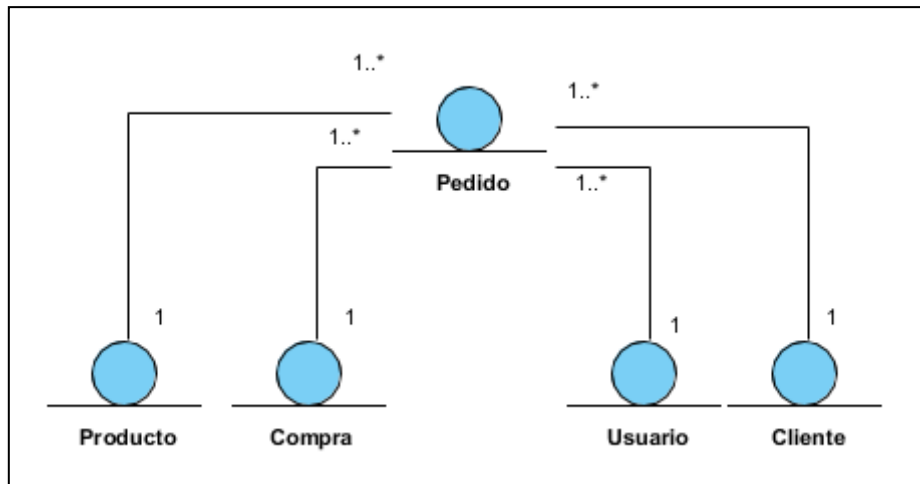


Fig. 81: Diagrama de clases general de análisis.

4.1.5. Iteración #5: Diseño

- **Subsistema De Diseño (En Función A Paquetes)**

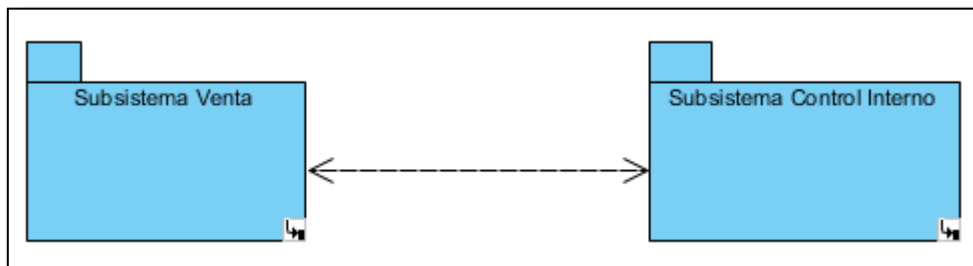


Fig. 82: Diagrama de contexto de subsistema de diseño

○ **Diagrama de realizaciones de casos de uso diseño**

○ **Ventas**

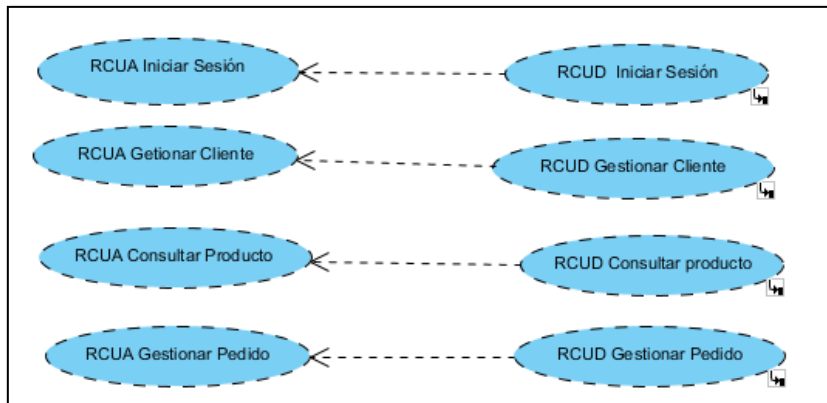


Fig. 83: Dia grama de contexto de realización de caso de uvso de Ventas.

○ **Control Interno**

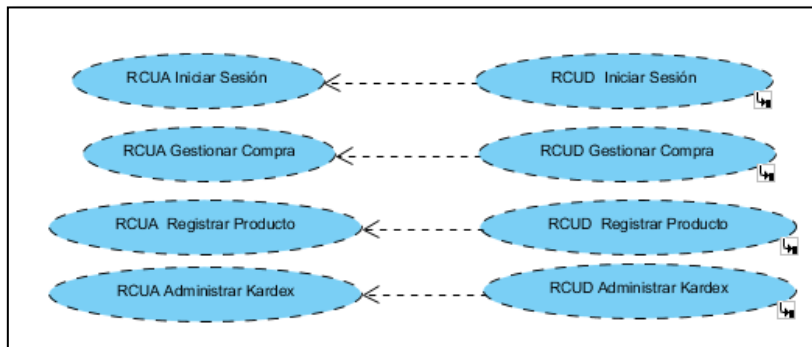


Fig. 84: Diagrama de contexto de realización de caso de uso de control interno

• **Diagrama de Secuencia**

○ **Ventas**

• **Inicio Sesión**

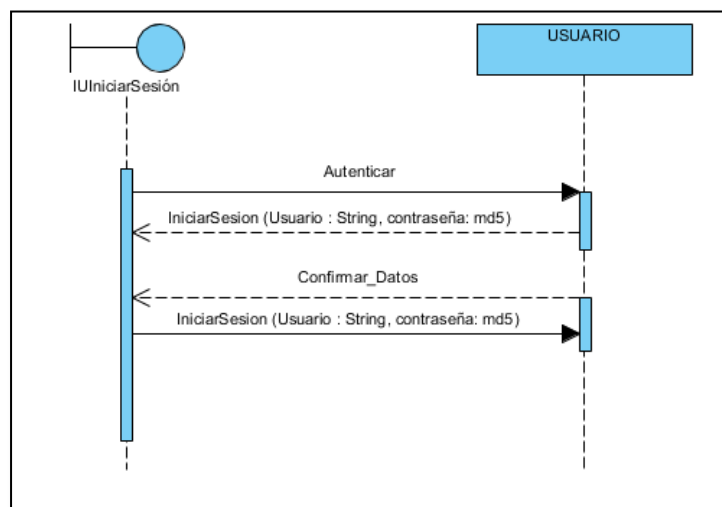


Fig. 85: Diagrama de secuencia de inicio de sesión de ventas.

- **Gestionar/ Cliente**

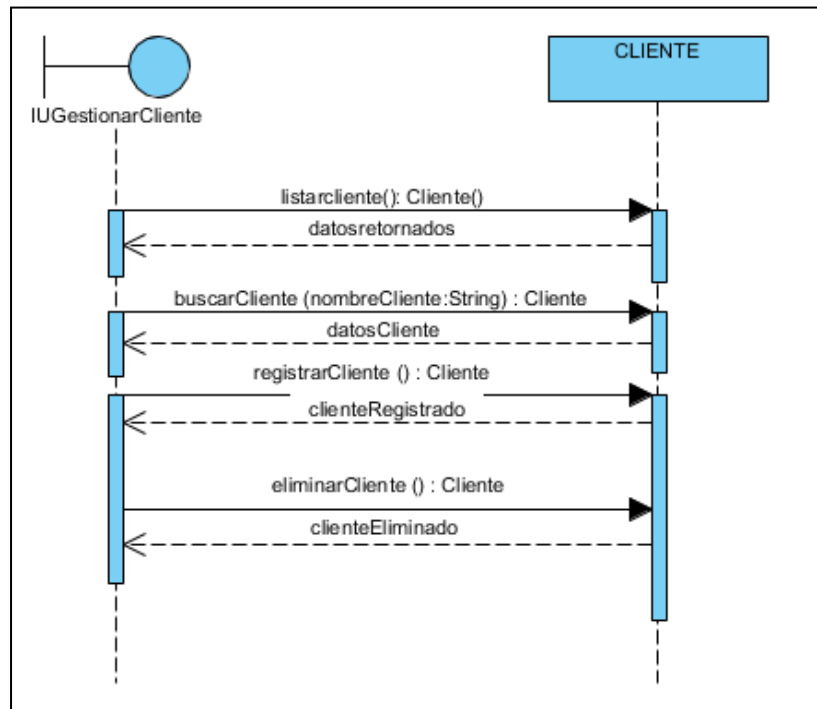


Fig. 86: Diagrama-secuencia de gestionar cliente de venta

- **Consultar/Producto**

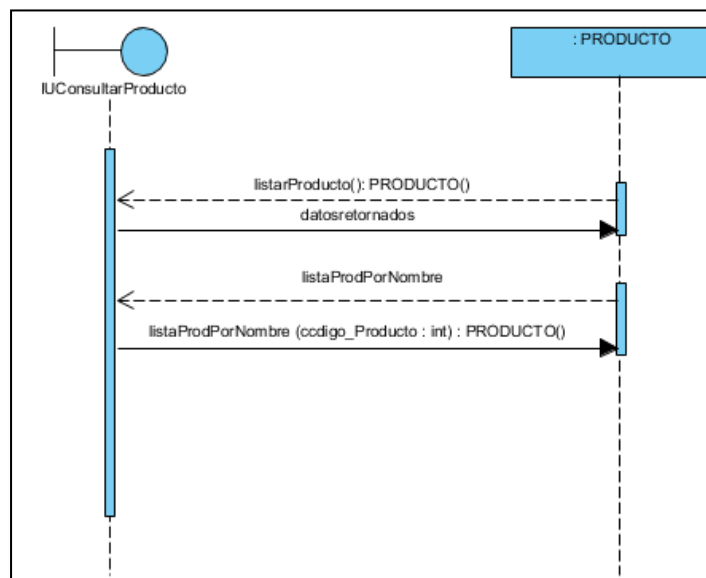


Fig. 87: Diagrama de secuencia de consultar producto de venta.

- **Gestionar / Pedido**

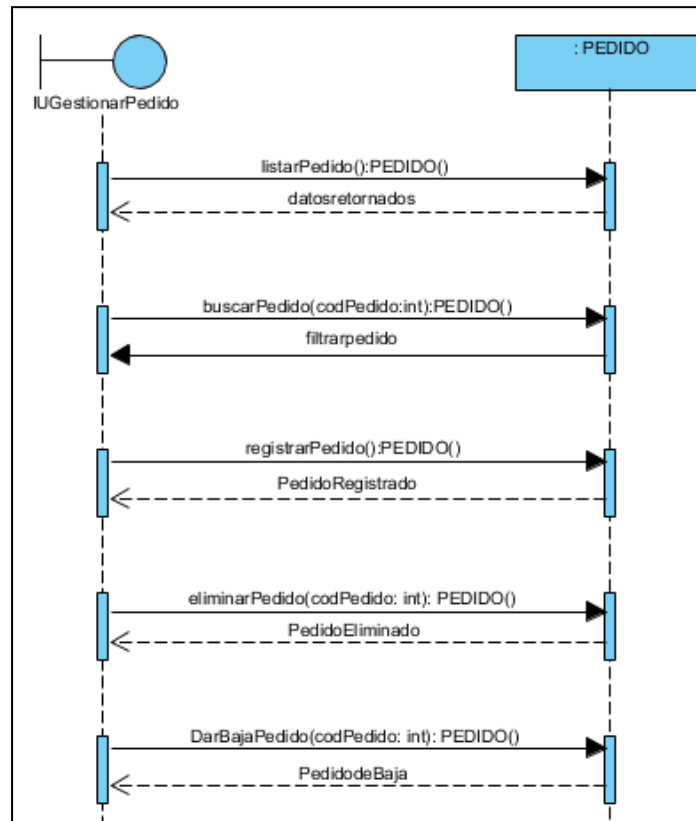


Fig. 88: Diagrama de secuencia -gestionar pedido de ventas.

- **Control Interno**

- **Inicio/Sesión**

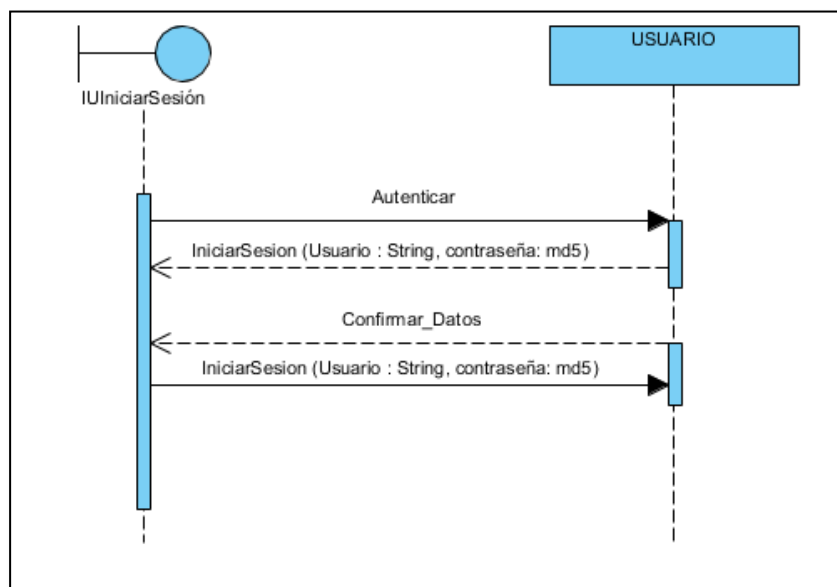


Fig. 89: Diagrama de secuencia de iniciar sesión de control interno

- **Gestionar Compra**

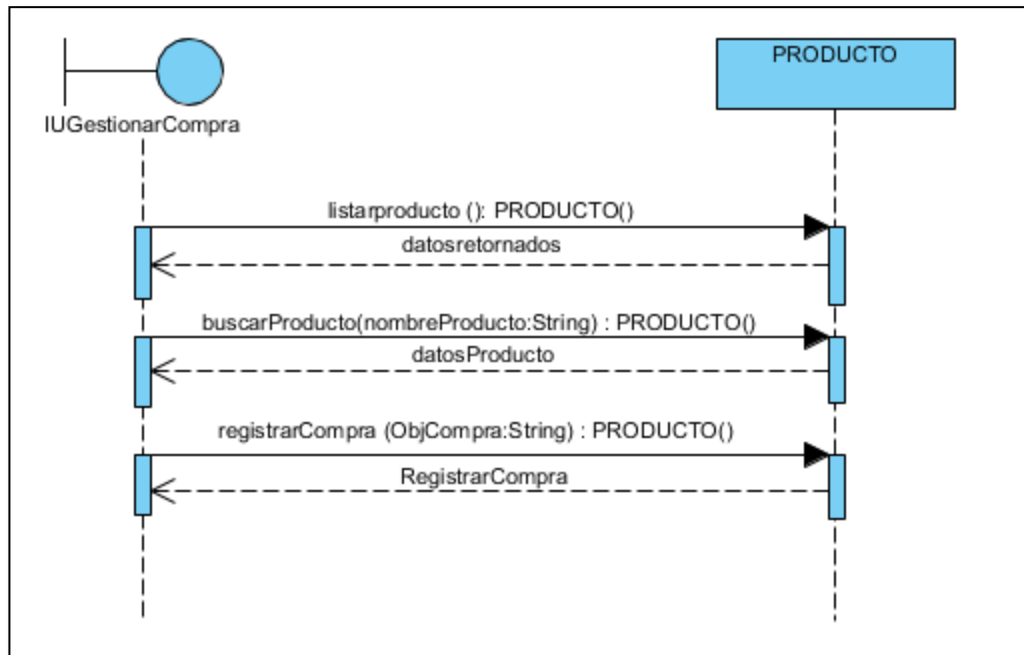


Fig. 90: Diagrama de secuencia de gestionar compra control interno

- **Registrar Producto**

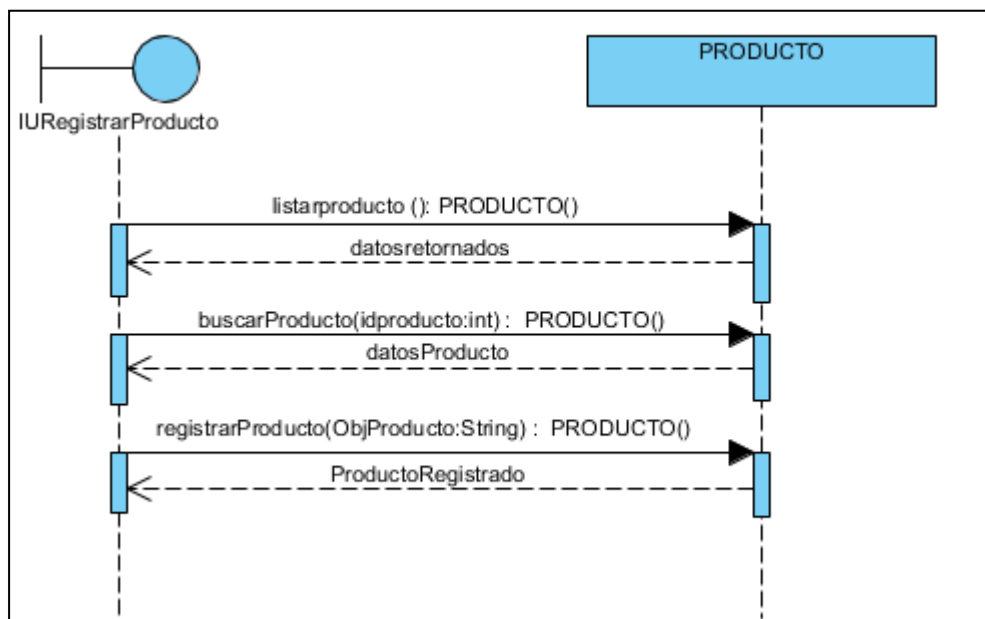


Fig. 91: Diagrama de secuencia de registrar producto de control interno

- **Administrar Kardex**

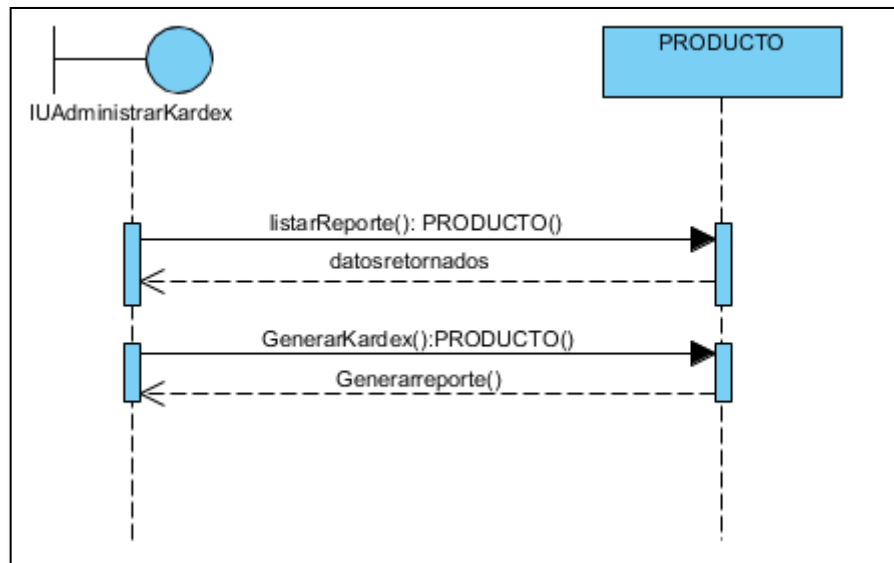


Fig. 92: Diagrama de secuencia de administrar kardex de control interno

- **Diagramas - parciales - clases - Diseño**

- **Ventas**

- **Iniciar/Sesión**

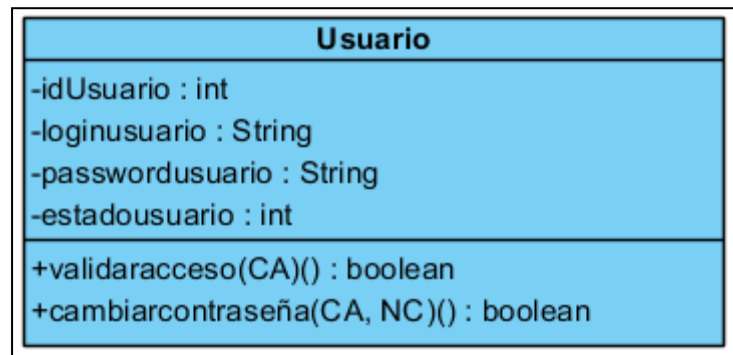


Fig. 93: Diagrama parcial -clases de diseño de iniciar sesión de venta.

- **Gestionar/Cliente**

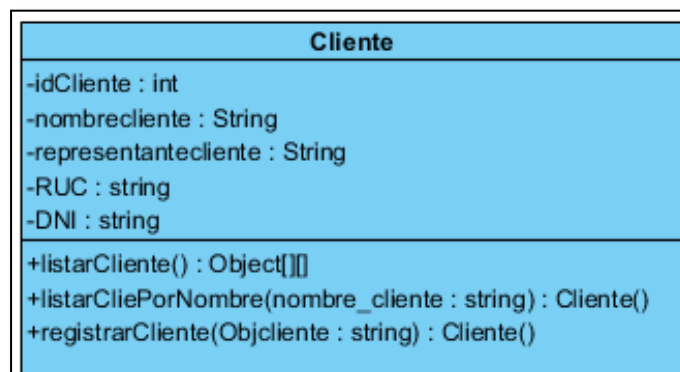


Fig. 94: Diagrama parcial de clases de gestionar cliente de ventas

- **Consultar / Producto**

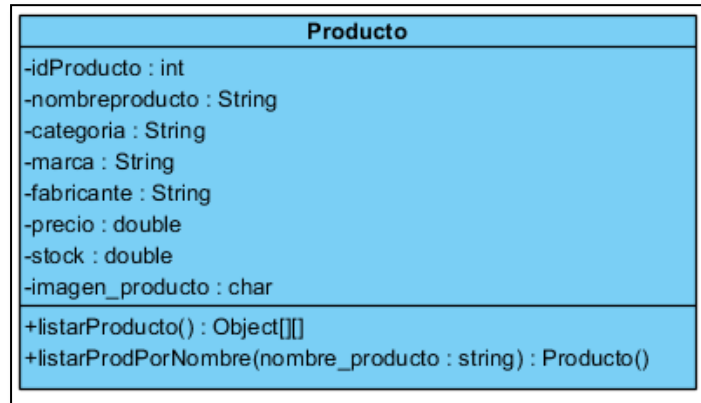


Fig. 95: Diagrama parcial -clases de diseño de consultar - producto -ventas.

- **Gestionar Pedido**

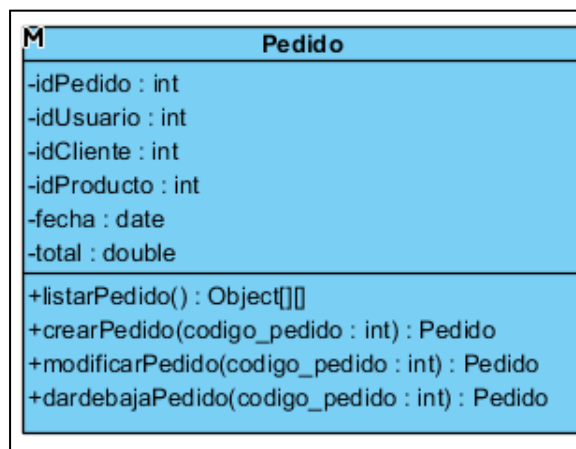


Fig. 96: Diagrama parcial - clases de gestionar pedido -venta

- **Control Interno**

- **Iniciar Sesión**

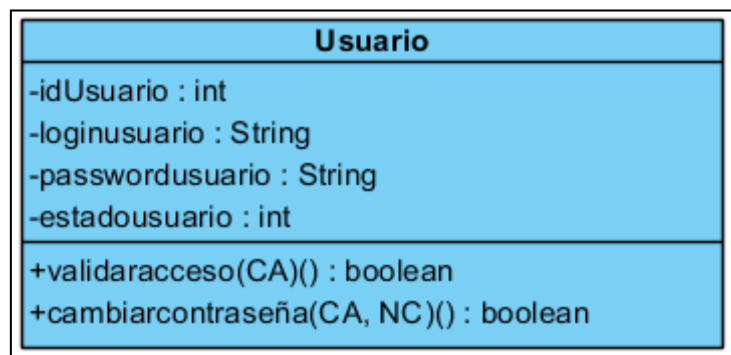


Fig. 97: Diagrama parcial - clases -iniciar sesión - control interno

- Gestionar Compra

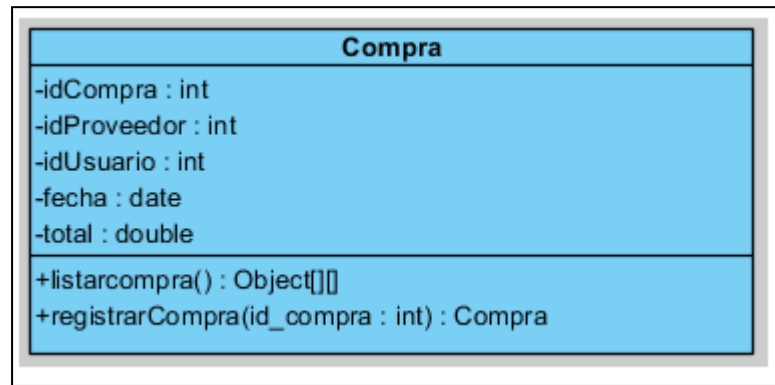


Fig. 98: Diagrama parcial de clases de gestionar compra de control interno

- Diagrama - clases - diseño general

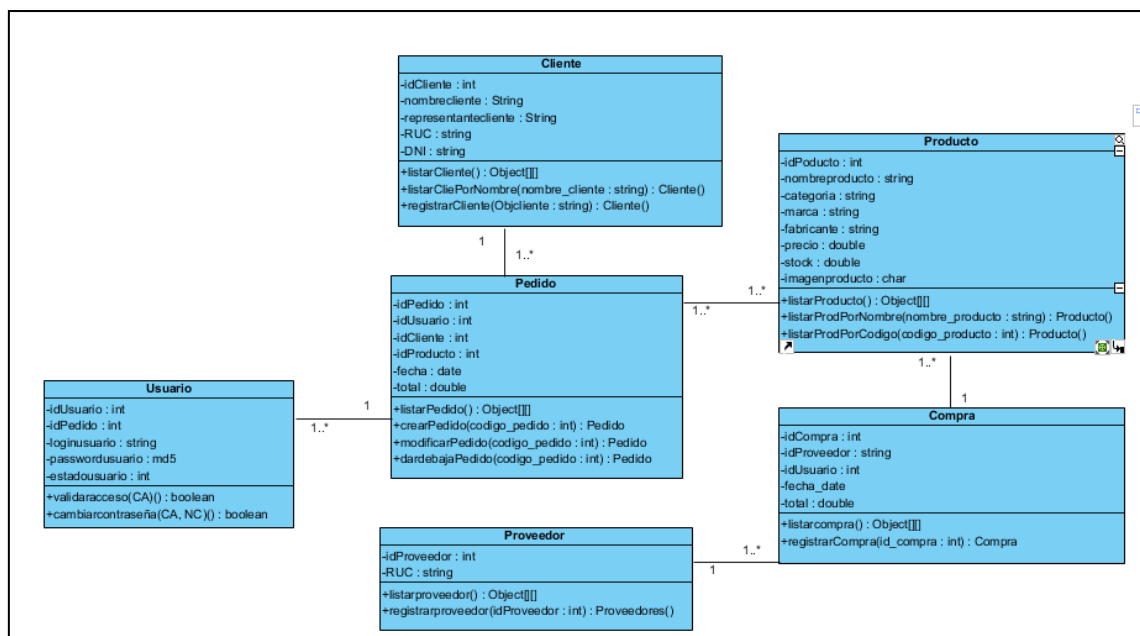


Fig. 99: Diagrama -clases - diseño general

○ **Diseño de Base Datos**

• **Base datos lógica (BDL)**

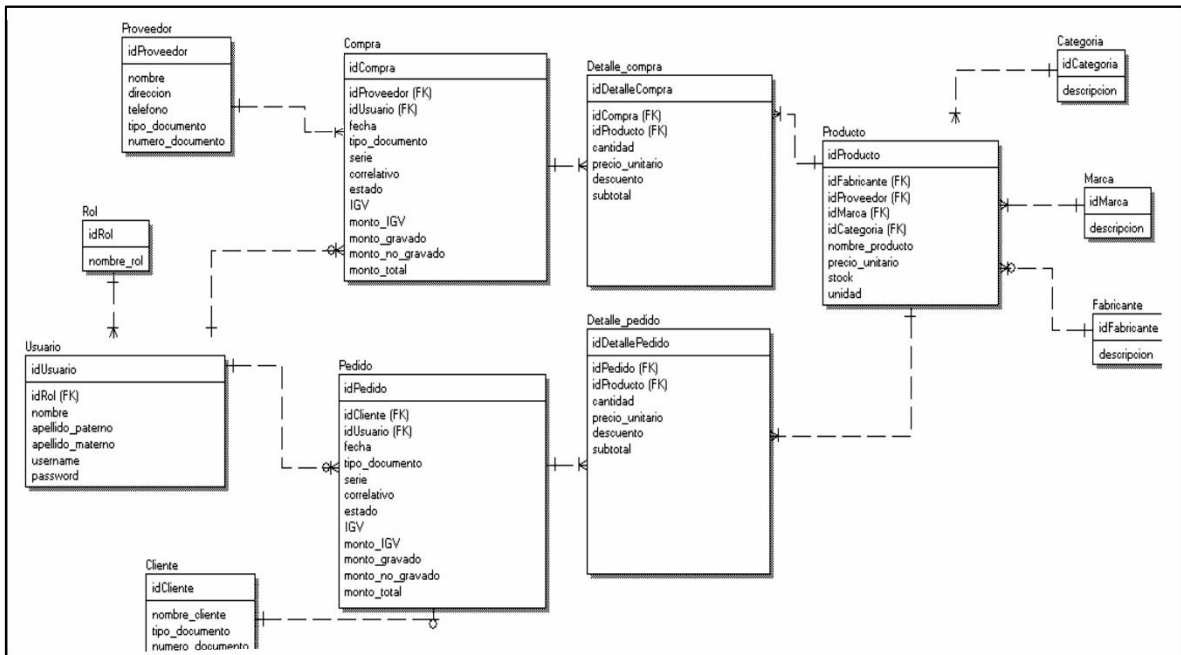


Fig. 100: Diagrama -diseño de base - datos lógica

• **Base-datos-Física(BDF)**

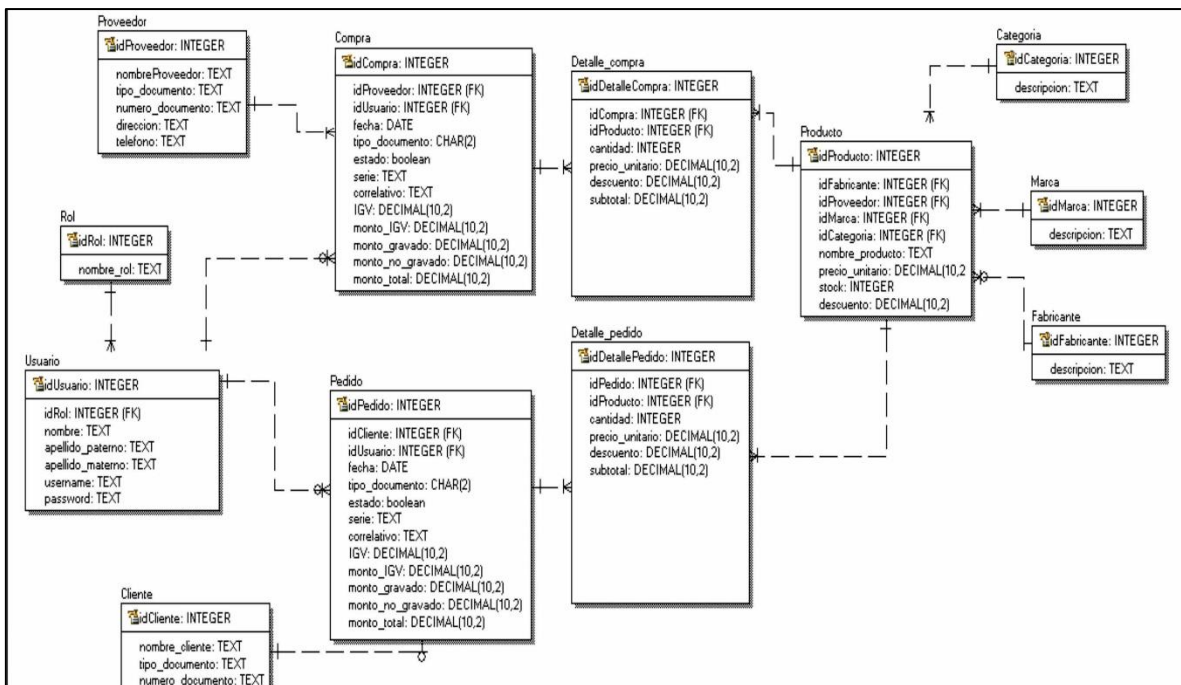


Fig. 101: Diagrama -diseño - BD base de datos físico

- **Diagrama / Estados**

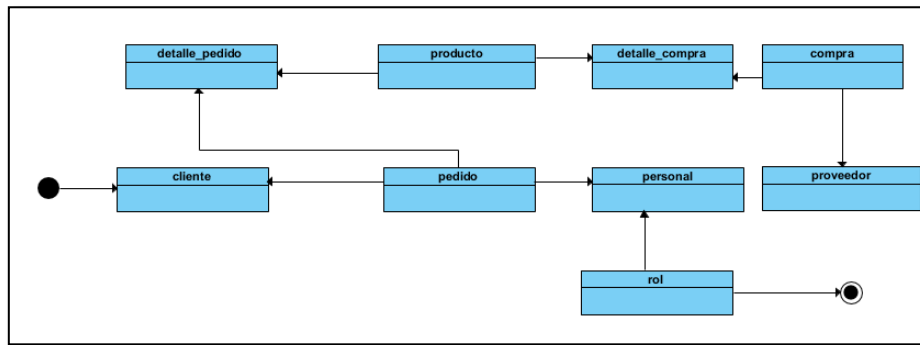


Fig. 102: Diagrama -diseño - estados

Iteración #6: Implementación / Prueba

- **Diseño de Interfaces**

Las vistas o interfaces del aplicativo móvil se encuentran el **Anexo N° 01:**

Manual de Usuario

- **Diagrama / Despliegue**

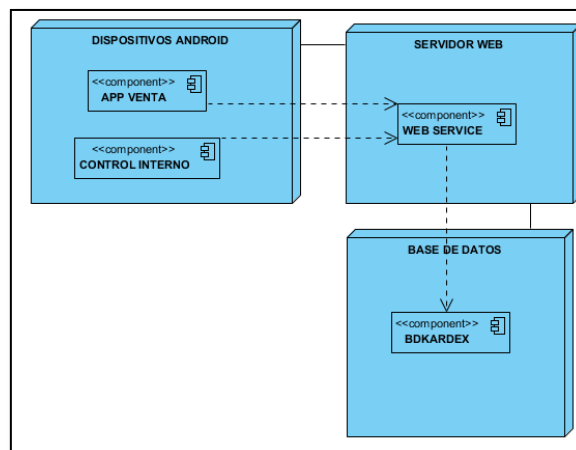


Fig. 103: Diagrama -despliegue.

- **Diagrama / Componentes**

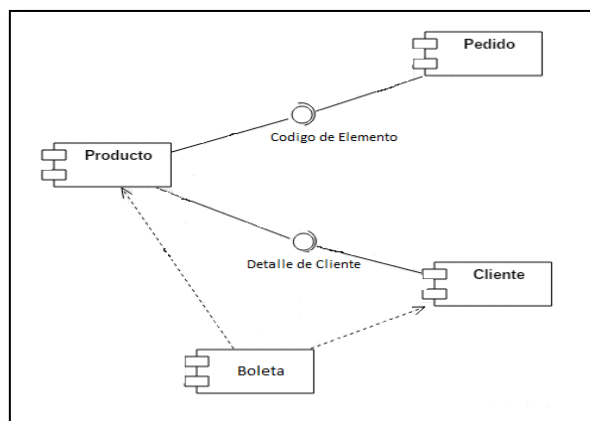


Fig. 104: Diagrama - componentes

CÓDIGO FUENTE PARCIAL DEL PRODUCTO ACREDITABLE

CONEXIÓN

```
<?php
/**
 * Provee las constantes para conectarse a la base de datos
 * Mysql.
 */
define("HOSTNAME", "localhost");// Nombre del host
define("DATABASE", "kardexapp"); // Nombre de la base de datos
define("USERNAME", "root"); // Nombre del usuario
define("PASSWORD", ""); // Nombre de la contraseña
?>
```

Login

```
<?php
/**
 * Clase que envuelve una instancia de la clase PDO
 * para el manejo de la base de datos
 */
require_once 'mysql_login.php';
class Database
{
    /**
     * Única instancia de la clase
     */
    private static $db = null;

    /**
     * Instancia de PDO
     */
    private static $pdo;

    final private function __construct()
    {
        try {
            // Crear nueva conexión PDO
            self::getDb();
        } catch (PDOException $e) {
            // Manejo de excepciones
            echo "error";
        }
    }

    /**
     * Retorna en la única instancia de la clase
     * @return Database|null
     */
    public static function getInstance()
    {
        if (self::$db === null) {
            self::$db = new self();
        }
        return self::$db;
    }
}
```

```

}

/**
 * Crear una nueva conexión PDO basada
 * en los datos de conexión
 * @return PDO Objeto PDO
 */
public function getDb()
{
    if (self::$pdo == null) {
        self::$pdo = new PDO(
            'mysql:host=' . HOSTNAME .
            ';dbname=' . DATABASE,
            USERNAME,
            PASSWORD,
            array(PDO::MYSQL_ATTR_INIT_COMMAND => "SET NAMES utf8")
        );

        // Habilitar excepciones
        self::$pdo->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
    }

    return self::$pdo;
}

/**
 * Evita la clonación del objeto
 */
final protected function __clone()
{
}

function _destructor()
{
    self::$pdo = null;
}
}
?>

```

Servidor

```

package com.app.kardex.kardex.Utils;

public class Servidor {
    //ENLACE BASE DEL SERVIDOR
    /**
     * ESTA VARIABLE DEBE TENER LA DIRECCION IP DE LA COMPUTADORA QUE
     EJECUTA EL XAMMP
     * SI CAMBIAS TU DIRECCION IP, TAMBIEN CAMBIA EL VALOR DE ESTA VARIABLE,
     NO LO OLVIDES
     */
    public static final String SERVIDOR = "http://192.168.43.12/";

    //RUTAS DE PETICIONES HTTP

```

```

//CLIENTE
public static final String ACTUALIZAR_CLIENTE =
"webservice/controlador/cliente/actualizar_cliente.php";
public static final String OBTENER_CLIENTES =
"webservice/controlador/cliente/obtener_clientes.php";
public static final String REGISTRAR_CLIENTE =
"webservice/controlador/cliente/registrar_cliente.php";

//COMPRA
public static final String ELIMINAR_COMPRA =
"webservice/controlador/compra/eliminar_compra.php";
public static final String OBTENER_COMPRAS_REGISTRADAS =
"webservice/controlador/compra/obtener_compras_registrados.php";
public static final String OBTENER_DETALLES_COMPRA =
"webservice/controlador/compra/obtener_detalle_compras.php";
public static final String REGISTRAR_COMPRA =
"webservice/controlador/compra/registrar_compra.php";

//P E D I D O
public static final String ELIMINAR_PEDIDO =
"webservice/controlador/pedido/eliminar_pedido.php";
public static final String OBTENER_PEDIDOS_BOLETA_REGISTRADOS =
"webservice/controlador/pedido/obtener_pedidos_boleta_registrados.php";
public static final String OBTENER_PEDIDOS_FACTURA_REGISTRADOS =
"webservice/controlador/pedido/obtener_pedidos_factura_registrados.php";
public static final String OBTENER_DETALLES_PEDIDO =
"webservice/controlador/pedido/obtener_detalle_pedidos.php";
public static final String REGISTRAR_PEDIDO =
"webservice/controlador/pedido/registrar_pedido.php";

//PRODUCTO
public static final String ACTUALIZAR_PRODUCTO =
"webservice/controlador/producto/actualizar_producto.php";
public static final String OBTENER_PRODUCTOS =
"webservice/controlador/producto/obtener_productos.php";
public static final String REGISTRAR_PRODUCTO =
"webservice/controlador/producto/registrar_producto.php";

//MARCA
public static final String OBTENER_MARCAS =
"webservice/controlador/marca/obtener_marcas.php";
public static final String REGISTRAR_MARCA =
"webservice/controlador/marca/registrar_marca.php";

//FABRICANTE
public static final String OBTENER_FABRICANTES =
"webservice/controlador/fabricante/obtener_fabricantes.php";
public static final String REGISTRAR_FABRICANTE =
"webservice/controlador/fabricante/registrar_fabricante.php";

//CATEGORÍA
public static final String OBTENER_CATEGORIAS =
"webservice/controlador/categoria/obtener_categorias.php";
public static final String REGISTRAR_CATEGORIA =
"webservice/controlador/categoria/registrar_categoria.php";

//PROVEEDOR
public static final String ACTUALIZAR_PROVEEDOR =
"webservice/controlador/proveedor/actualizar_proveedor.php";

```



```

private String idUsuario;

// Lista del modelo
private List<MenuItemKardex> menusKardex;

// Items de option menu
private MenuItem itemListView;
private MenuItem itemGridView;

// Variables
private int contador = 0;
private final int CAMBIAR_A_LIST_VIEW = 0;
private final int CAMBIAR_A_GRID_VIEW = 1;

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_menu);
    ButterKnife.bind(this);
    Toolbar toolbar = (Toolbar) findViewById(R.id.toolbar);
    setSupportActionBar(toolbar);

    // Session class instance
    session = new UserSessionManager(getApplicationContext());
    usuario = session.getUserDetails();
    // get id
    idUsuario = usuario.get(UserSessionManager.KEY_ID);

    // Metodo para mostrar el icono
    mostrarIconBar();
    // Obtenemos el listado de frutas
    menusKardex = getMenus();
    // Declaramos los contenedores de vistas
    listView = (ListView) findViewById(R.id.listView);
    gridView = (GridView) findViewById(R.id.gridView);

    // Adjuntamos el método click para ambas vistas
    listView.setOnItemClickListener(this);
    gridView.setOnItemClickListener(this);

    // Instanciamos el adaptador para cada caso
    adaptadorListView = new AdaptadorGridMenu(this, R.layout.list_view_item_menu,
    menusKardex);
    adaptadorGridView = new AdaptadorGridMenu(this, R.layout.grid_view_item_menu,
    menusKardex);

    // Establecemos el adaptador
    listView.setAdapter(adaptadorListView);
    gridView.setAdapter(adaptadorGridView);

    // Registramos el context menu para ambas vistas
    registerForContextMenu(listView);
    registerForContextMenu(gridView);

    //Toast.makeText(this, "Bienvenido", Toast.LENGTH_SHORT).show();
}

/**
 * Metodo para mostrar el icono de la aplicacion en el action bar

```

```

*/
private void mostrarIconBar() {
    getSupportActionBar().setIcon(R.mipmap.ic_launcher);
    getSupportActionBar().setDisplayUseLogoEnabled(true);
    getSupportActionBar().setDisplayHomeAsUpEnabled(true);
}

/**
 * Regresa una lista de los las opciones del menu
 *
 * @return
 */
private List<MenuitemKardex> getMenus() {
    List<MenuitemKardex> list = new ArrayList<>();
    if (idUserario.equalsIgnoreCase("1")) {
        list.add(new MenuitemKardex("Venta", R.drawable.venta));
        list.add(new MenuitemKardex("Compra", R.drawable.compra));
        list.add(new MenuitemKardex("Kardex", R.drawable.kardex));
        list.add(new MenuitemKardex("Devoluciones", R.drawable.devolucion));
        list.add(new MenuitemKardex("Producto", R.drawable.productp));
        list.add(new MenuitemKardex("Proveedor", R.drawable.proveedor));
        list.add(new MenuitemKardex("Cliente", R.drawable.cliente));
        list.add(new MenuitemKardex("Fabricante", R.drawable.fabrica));
        list.add(new MenuitemKardex("Categoria", R.drawable.fabrica));
        list.add(new MenuitemKardex("Marca", R.drawable.fabrica));
        list.add(new MenuitemKardex("Salir", R.drawable.salir));
    } else if (idUserario.equalsIgnoreCase("2")) {
        list.add(new MenuitemKardex("Venta", R.drawable.venta));
        list.add(new MenuitemKardex("Cliente", R.drawable.cliente));
        list.add(new MenuitemKardex("Salir", R.drawable.salir));
    } else if (idUserario.equalsIgnoreCase("3")) {
        list.add(new MenuitemKardex("Kardex", R.drawable.kardex));
        list.add(new MenuitemKardex("Devoluciones", R.drawable.devolucion));
        list.add(new MenuitemKardex("Proveedor", R.drawable.proveedor));
        list.add(new MenuitemKardex("Salir", R.drawable.salir));
    }

    return list;
}

@Override
public void onItemClick(AdapterView<?> adapterView, View view, int position, long id) {
    clickMenu(menusKardex.get(position));
}

/**
 * @param menuitemKardex objeto modelo Método que notificara con un "Toast" los datos
 de la
 *          fruta seleccionada
 */
private void clickMenu(MenuitemKardex menuitemKardex) {
    switch (menuitemKardex.getTitulo()) {
        case "Venta":
            loadView(new VentaActivity());
            break;
        case "Compra":
            loadView(new CompraActivity());
            break;
        case "Devoluciones":
            loadView(new Devoluciones());
    }
}

```

```

        break;
    case "Kardex":
        loadView(new KardexActivity());
        break;
    case "Producto":
        loadView(new ProductosActivity());
        break;
    case "Proveedor":
        loadView(new ProveedoresActivity());
        break;
    case "Cliente":
        loadView(new ClientesActivity());
        break;
    case "Fabricante":
        loadView(new FabricanteActivity());
        break;
    case "Categoria":
        loadView(new CategoriaActivity());
        break;
    case "Marca":
        loadView(new MarcaActivity());
        break;
    case "Salir":
        dialogoAlertaLogout("¿Esta seguro que desea cerrar sesion?");
        break;
    default:
        break;
    }
}

private void loadView(AppCompatActivity activity) {
    Intent intent = new Intent(getApplicationContext(), activity.getClass());
    intent.addFlags(Intent.FLAG_ACTIVITY_NEW_TASK);
    startActivity(intent);
}

private void logout(AppCompatActivity activity) {
    Intent intent = new Intent(getApplicationContext(), activity.getClass());
    intent.addFlags(Intent.FLAG_ACTIVITY_CLEAR_TOP);
    startActivity(intent);
    finish();
}

@Override
public boolean onCreateOptionsMenu(Menu menu) {
    // Inflamos el option menu con nuestro layout personalizado
    MenuInflater inflater = getMenuInflater();
    inflater.inflate(R.menu.menu_opcion, menu);
    // luego de inflar , recogemos las referencias a los botones que nos interesan
    itemListView = menu.findItem(R.id.list_view);
    itemGridView = menu.findItem(R.id.grid_view);
    return true;
}

@Override
public boolean onOptionsItemSelected(MenuItem item) {
    // Eventos para los clicks en los botones del option menu
    switch (item.getItemId()) {
        case R.id.list_view:
            cambioListGridView(CAMBIAR_A_LIST_VIEW);
    }
}

```

```

        return true;
    case R.id.grid_view:
        cambioListGridView(CAMBIAR_A_GRID_VIEW);
        return true;
    default:
        return super.onOptionsItemSelected(item);
    }
}

/**
 * @param opcion <p>Metodo para cambiar entre Grid/List view</p>
 * <p>Recibe constantes CAMBIAR_A_LIST_VIEW o CAMBIAR_A_GRID_VIEW
 como parametro</p>
 */
private void cambioListGridView(int opcion) {
    if (opcion == CAMBIAR_A_LIST_VIEW) {
        //si queremos cambiar a list view, y el list view está en modo invisible...
        if (this.listView.getVisibility() == View.INVISIBLE) {
            // Escondemos el grid view, y enseñamos su boton en el menu de opciones
            this.gridView.setVisibility(View.INVISIBLE);
            this.itemGridView.setVisible(true);
            // no olvidamos enseñar el list view y esconder su boton en el menu de opciones
            this.listView.setVisibility(View.VISIBLE);
            this.itemListView.setVisible(false);
        }
    } else if (opcion == CAMBIAR_A_GRID_VIEW) {
        // Si queremos cambiar a grid view, y el grid view esta en modo invisible...
        if (this.gridView.getVisibility() == View.INVISIBLE) {
            // escondemos el list view, y enseñamos su boton en el menu de opciones
            this.listView.setVisibility(View.INVISIBLE);
            this.itemListView.setVisible(true);
            // no olvidamos enseñar el grid view y esconder su boton en el menu de opciones
            this.gridView.setVisibility(View.VISIBLE);
            this.itemGridView.setVisible(false);
        }
    }
}

public void dialogoAlertaLogout(String mensaje) {
    AlertDialog.Builder builder = new AlertDialog.Builder(this, R.style.alertDialog);

    builder.setMessage(mensaje).setTitle("Advertencia")
        .setCancelable(true)
        .setIcon(R.mipmap.ic_warning)
        .setPositiveButton("Sí", new DialogInterface.OnClickListener() {
            @Override
            public void onClick(DialogInterface dialogInterface, int i) {
                // Clear the User session data
                // and redirect user to LoginActivity
                session.logoutUser();
                finish();
            }
        }).setNegativeButton("Cancelar", null);
    AlertDialog alertDialog = builder.create();
    alertDialog.show();
}

@OnClick(R.id.imgLogo)
public void onViewClicked() {
    Toast.makeText(this, R.string.app_name, Toast.LENGTH_SHORT).show();
}

```

```
}
}
```

BASE DE DATOS

```
kardexapp
-- phpMyAdmin SQL Dump
-- version 4.8.5
-- https://www.phpmyadmin.net/
--

-- Servidor: 127.0.0.1
-- Tiempo de generación: 05-07-2019 a las 08:37:53
-- Versión del servidor: 10.1.38-MariaDB
-- Versión de PHP: 7.1.27
SET
  SQL_MODE = "NO_AUTO_VALUE_ON_ZERO";
SET
  AUTOCOMMIT = 0;
START TRANSACTION
;
SET
  time_zone = "+00:00";
/*!40101
SET
  @OLD_CHARACTER_SET_CLIENT = @@CHARACTER_SET_CLIENT */;
/*!40101
SET
  @OLD_CHARACTER_SET_RESULTS = @@CHARACTER_SET_RESULTS */;
/*!40101
SET
  @OLD_COLLATION_CONNECTION = @@COLLATION_CONNECTION */;
/*!40101
SET NAMES
  utf8mb4 */;
--
-- Base de datos: `kardexapp`
-----

-- ESTRUCTURA DE TABLA PARA LA TABLA `CATEGORIA`
--
CREATE TABLE `categoria` (
  `idCategoria` INT(11) NOT NULL,
  `descripcion` TEXT COLLATE utf8_bin NOT NULL
) ENGINE = InnoDB DEFAULT CHARSET = utf8 COLLATE = utf8_bin;

-- Volcado de datos para la tabla `categoria`

INSERT INTO `categoria` (`idCategoria`, `descripcion`)
VALUES(1, 'Comestible'),(2, 'Chicles'),(3, 'Chocolates'),(4, 'Chupetines y Paletas'),(5,
'Cereales'),(6, 'Galletas Saladas'),(7, 'Galletas Dulce');
-----

-- ESTRUCTURA DE TABLA PARA LA TABLA `CLIENTE`
--
CREATE TABLE `cliente` (
  `idCliente` INT(11) NOT NULL,
  `nombre_cliente` TEXT COLLATE utf8_bin NOT NULL,
  `tipo_documento` TEXT COLLATE utf8_bin,
```

```

    `numero_documento` TEXT COLLATE utf8_bin NOT NULL
) ENGINE = InnoDB DEFAULT CHARSET = utf8 COLLATE = utf8_bin;
--
-- Volcado de datos para la tabla `cliente`
--

INSERT INTO `cliente` (
  `idCliente`,
  `nombre_cliente`,
  `tipo_documento`,
  `numero_documento`
)
VALUES( 1, 'Bodega Don Pepe', 'DNI', '75758489' ),( 2, 'Bodega Don Luchito',
'RUC', '10164398124' ),( 3, 'Bodega La Chotanita', 'RUC', '10273734606' ),( 4,
'Bodega Virgen del Carmen', 'RUC', '10167917254' ),( 5, 'Bodega Bazar Lobito', 'RUC',
'10455784986' ),( 6, 'Mi Dulce Kristell', 'DNI', '10431745762' ),( 7, 'Bodega Mi Buen
Pastor', 'DNI', '40414578' );
-----
--
-- ESTRUCTURA DE TABLA PARA LA TABLA `COMPRA`
--
CREATE TABLE `compra` (
  `idCompra` INT(11) NOT NULL,
  `fecha` DATE NOT NULL,
  `idProveedor` INT(11) NOT NULL,
  `idUsuario` INT(11) NOT NULL,
  `serie` TEXT COLLATE utf8_bin NOT NULL,
  `correlativo` TEXT COLLATE utf8_bin NOT NULL,
  `IGV` DECIMAL(10, 2) NOT NULL,
  `monto_gravado` DECIMAL(10, 2) NOT NULL,
  `estado` TINYINT(1) NOT NULL,
  `tipo_documento` CHAR(2) COLLATE utf8_bin DEFAULT NULL,
  `monto_no_gravado` DECIMAL(10, 2) DEFAULT NULL,
  `monto_IGV` DECIMAL(10, 2) DEFAULT NULL,
  `monto_total` DECIMAL(10, 2) DEFAULT NULL
) ENGINE = InnoDB DEFAULT CHARSET = utf8 COLLATE = utf8_bin;
--
-- Volcado de datos para la tabla `compra`
--

INSERT INTO `compra` (
  `idCompra`, `fecha`, `idProveedor`, `idUsuario`, `serie`, `correlativo`,
  `IGV`, `monto_gravado`, `estado`, `tipo_documento`, `monto_no_gravado`,
  `monto_IGV`, `monto_total` )
VALUES( 1, '2019-07-03', 1, 1, 'F001', '00001', '0.18', '349.50', 1, '20', '0.00',
'62.91', '412.41' ),( 2, '2019-07-03', 1, 1, 'F001', '00002', '0.18', '209.70', 1, '20',
'0.00', '37.75', '247.45' ),( 3, '2019-07-03', 5, 1, 'F001', '00003', '0.18', '750.00',
0, '20', '0.00', '135.00', '885.00' ),( 4, '2019-07-03', 5, 1, 'F001', '00004', '0.18',
'270.00', 1, '20', '0.00', '48.60', '318.60' ),( 5, '2019-07-05', 7, 1, 'F001', '00005',
'0.18', '300.00', 1, '20', '0.00', '54.00', '354.00' ),( 6, '2019-07-05', 7, 1, 'F001',
'00006', '0.18', '150.00', 1, '20', '0.00', '27.00', '177.00' ),( 7, '2019-07-05', 7, 1,
'F001', '00007', '0.18', '450.00', 0, '20', '0.00', '81.00', '531.00' );

```

```

-----
--
-- Estructura Stand-in para la vista `compras`
-- (Véase abajo para la vista actual)
--

```

```

CREATE TABLE `compras` (
  `idCompra` INT(11),
  `idDetalleCompra` INT(11),
  `fecha` DATE,
  `nombre_producto` TEXT,
  `estado` TINYINT(1),
  `cantidad` INT(11),
  `precio_unitario` DECIMAL(10, 2),
  `subtotal` DECIMAL(10, 2)
);
-----

```

-- ESTRUCTURA DE TABLA PARA LA TABLA `DETALLE COMPRA`

```

--
CREATE TABLE `detalle_compra` (
  `idDetalleCompra` INT(11) NOT NULL,
  `cantidad` INT(11) NOT NULL,
  `precio_unitario` DECIMAL(10, 2) NOT NULL,
  `idCompra` INT(11) NOT NULL,
  `idProducto` INT(11) NOT NULL,
  `subtotal` DECIMAL(10, 2) NOT NULL,
  `descuento` DECIMAL(10, 2) DEFAULT NULL
) ENGINE = InnoDB DEFAULT CHARSET = utf8 COLLATE = utf8_bin;
--

```

```

-- Volcado de datos para la tabla `detalle_compra`
--

```

```

INSERT INTO `detalle_compra` ( `idDetalleCompra`, `cantidad`, `precio_unitario`,
`idCompra`, `idProducto`, `subtotal`, `descuento` ) VALUES( 1, 50, '6.99', 1, 1,
'349.50', NULL ),( 2, 30, '6.99', 2, 1, '209.70', NULL ),( 3, 50, '15.00', 3, 6, '750.00',
NULL ),( 4, 15, '18.00', 4, 7, '270.00', NULL ),( 5, 20, '15.00', 5, 9, '300.00', NULL ),(
6, 10, '15.00', 6, 9, '150.00', NULL ),( 7, 30, '15.00', 7, 9, '450.00', NULL );
-----

```

-- ESTRUCTURA DE TABLA PARA LA TABLA `DETALLE PEDIDO`

```

--
CREATE TABLE `detalle_pedido` (
  `idDetallePedido` INT(11) NOT NULL,
  `cantidad` INT(11) NOT NULL,
  `precio_unitario` DECIMAL(10, 2) NOT NULL,
  `subtotal` DECIMAL(10, 2) NOT NULL,
  `idPedido` INT(11) NOT NULL,

```

```

    `idProducto` INT(11) NOT NULL,
    `descuento` DECIMAL(10, 2) DEFAULT NULL
) ENGINE = InnoDB DEFAULT CHARSET = utf8 COLLATE = utf8_bin;
--
-- Volcado de datos para la tabla `detalle_pedido`
--INSERT INTO `detalle_pedido`(
    `idDetallePedido`,
    `cantidad`,
    `precio_unitario`,
    `subtotal`,
    `idPedido`,
    `idProducto`,
    `descuento`
)
VALUES( 1, 20, '6.99', '139.80', 1, 1, NULL),( 2, 20, '5.00', '100.00',
2, 2, NULL),( 3, 50, '18.00', '900.00', 3, 7, NULL),( 4, 20, '6.99',
'139.80', 3, 1, NULL),( 5, 25, '18.00', '450.00', 4, 7, NULL),(6, 10, '6.99',
'69.90', 5, 1, NULL),( 7, 50, '6.99', '349.50', 6, 1, NULL),( 8, 50, '6.99',
'349.50', 7, 1, NULL),( 9, 50, '6.99', '349.50', 8, 1, NULL),( 10, 20,
'15.00', '300.00', 9, 9, NULL),( 11, 20, '15.00', '300.00', 10, 9, NULL
),( 12, 15, '15.00', '225.00', 11, 9, NULL),( 13, 10, '15.00', '150.00', 12,
9, NULL),( 14, 25, '15.00', '375.00', 13, 9, NULL),( 15, 25, '15.00',
'375.00', 14, 9, NULL
);
-----
--
-- Estructura de tabla para la tabla `fabricante`
--
CREATE TABLE `fabricante` (
  `idFabricante` INT(11) NOT NULL,
  `descripcion` TEXT COLLATE utf8_bin NOT NULL
) ENGINE = InnoDB DEFAULT CHARSET = utf8 COLLATE = utf8_bin;
--
-- Volcado de datos para la tabla `fabricante`
--
INSERT INTO `fabricante`(`idFabricante`, `descripcion`)
VALUES(1, 'Costa'),(2, 'Nestle Peru'),(3, 'Confiperu'),(4, 'Arcor'),(5, 'Winters'),(6, 'Kraft
Foods'),(7, 'Victoria');
-----
--
-- Estructura de tabla para la tabla `marca`
--
CREATE TABLE `marca` (
  `idMarca` INT(11) NOT NULL,
  `descripcion` TEXT COLLATE utf8_bin NOT NULL
) ENGINE = InnoDB DEFAULT CHARSET = utf8 COLLATE = utf8_bin;
--
-- Volcado de datos para la tabla `marca`
--
INSERT INTO `marca`(`idMarca`, `descripcion`)
VALUES(1, 'Costa'),(2, 'Chocomel'),(3, 'Cua Cua'),(4, 'Doña Pepa'),(5, 'Arcor'),(6, 'Victoria');

```

```

-----
--
-- Estructura de tabla para la tabla `pedido`
--
CREATE TABLE `pedido` (
  `idPedido` INT(11) NOT NULL,
  `fecha` DATE NOT NULL,
  `idCliente` INT(11) NOT NULL,
  `idUsuario` INT(11) NOT NULL,
  `IGV` DECIMAL(10, 2) NOT NULL,
  `monto_gravado` DECIMAL(10, 2) NOT NULL,
  `serie` TEXT COLLATE utf8_bin NOT NULL,
  `correlativo` TEXT COLLATE utf8_bin NOT NULL,
  `estado` TINYINT(1) NOT NULL,
  `tipo_documento` CHAR(2) COLLATE utf8_bin DEFAULT NULL,
  `monto_IGV` DECIMAL(10, 2) DEFAULT NULL,
  `monto_total` DECIMAL(10, 2) DEFAULT NULL,
  `monto_no_gravado` DECIMAL(10, 2) DEFAULT NULL
) ENGINE = InnoDB DEFAULT CHARSET = utf8 COLLATE = utf8_bin;
--
-- Volcado de datos para la tabla `pedido`
--
INSERT INTO `pedido` (`idPedido`, `fecha`, `idCliente`, `idUsuario`, `IGV`,
`monto_gravado`, `serie`, `correlativo`, `estado`, `tipo_documento`, `monto_IGV`,
`monto_total`, `monto_no_gravado`) VALUES( 1, '2019-07-03', 1, 1, '0.18', '139.80',
'B001', '00001', 1, '10', '25.16', '164.96', '0.00'),( 2, '2019-07-03', 1, 1, '0.18',
'100.00', 'B001', '00002', 1, '10', '18.00', '118.00', '0.00'),( 3, '2019-07-03', 2, 1,
'0.18', '1039.80', 'B001', '00003', 1, '10', '187.16', '1226.96', '0.00'),( 4, '2019-07-04',
2, 1, '0.18', '450.00', 'B001', '00004', 1, '10', '81.00', '531.00', '0.00'),( 5, '2019-07-04',
3, 2, '0.18', '69.90', 'B001', '00001', 1, '10', '12.58', '82.48', '0.00'),( 6, '2019-07-05',
2, 1, '0.18', '349.50', 'B001', '00005', 1, '10', '62.91', '412.41', '0.00'),( 7, '2019-07-
05', 2, 1, '0.18', '349.50', 'B001', '00006', 1, '10', '62.91', '412.41', '0.00'),( 8, '2019-07-05',
2, 1, '0.18', '349.50', 'B001', '00007', 1, '10', '62.91', '412.41', '0.00'),( 9, '2019-
07-05', 1, 1, '0.18', '300.00', 'B001', '00008', 1, '10', '54.00', '354.00', '0.00'),( 10,
'2019-07-05', 1, 1, '0.18', '300.00', 'B001', '00009', 1, '10', '54.00', '354.00',
'0.00'),( 11, '2019-07-05', 3, 1, '0.18', '225.00', 'B001', '00010', 1, '10',
'40.50', '265.50', '0.00'),( 12, '2019-07-05', 2, 1, '0.18', '150.00', 'B001',
'00011', 0, '10', '27.00', '177.00', '0.00'),( 13, '2019-07-05', 5, 1, '0.18',
'375.00', 'B001', '00012', 1, '10', '67.50', '442.50', '0.00'),( 14, '2019-07-05', 5,
1, '0.18', '375.00', 'B001', '00013', 1, '10', '67.50', '442.50', '0.00');
-----
--
-- Estructura Stand-in para la vista `pedidos`
-- (Véase abajo para la vista actual)
--
CREATE TABLE `pedidos` ( `idPedido` INT(11), `idDetallePedido` INT(11), `fecha` DATE,
`producto` TEXT, `estado` TINYINT(1), `cantidad` INT(11), `precio_unitario`
DECIMAL(10, 2), `subtotal` DECIMAL(10, 2));
-----
--
-- ESTRUCTURA DE TABLA PARA LA TABLA `PRODUCTO`
--
CREATE TABLE `producto` (

```

```

`idProducto` INT(11) NOT NULL,
`nombre_producto` TEXT COLLATE utf8_bin NOT NULL,
`precio_unitario` DECIMAL(10, 2) NOT NULL,
`stock` INT(11) NOT NULL,
`descuento` DECIMAL(10, 2) DEFAULT NULL,
`unidad` TEXT COLLATE utf8_bin NOT NULL,
`idCategoria` INT(11) NOT NULL,
`idMarca` INT(11) NOT NULL,
`idProveedor` INT(11) NOT NULL,
`idFabricante` INT(11) NOT NULL
) ENGINE = InnoDB DEFAULT CHARSET = utf8 COLLATE = utf8_bin;
--

-- Volcado de datos para la tabla `producto`
--
INSERT INTO `producto`(`idProducto`, `nombre_producto`, `precio_unitario`,
`stock`, `descuento`, `unidad`, `idCategoria`, `idMarca`, `idProveedor`,
`idFabricante`)VALUES( 1, 'Chocolate Visio', '6.99', 200, '0.00', 'Unidades',
1, 1, 1, 1),( 2, 'galletas', '5.00', 230, '0.00', 'Unidades', 1, 2, 1, 1),(
6, 'Cua Cua x 10 unidades', '15.00', 250, '0.00', 'Unidades', 3, 3, 1, 6
),( 7, 'Doña Pepa X Bolsa', '18.00', 240, '0.00', 'Bolsas', 3, 4, 1, 6),( 9,
'Rellenitas galletas', '15.00', 175, '0.00', 'Unidades', 7, 6, 1, 7);
-----

--
-- ESTRUCTURA DE TABLA PARA LA TABLA `PROVEEDOR`
--
CREATE TABLE `proveedor`(
`idProveedor` INT(11) NOT NULL,
`nombreProveedor` TEXT COLLATE utf8_bin NOT NULL,
`tipo_documento` TEXT COLLATE utf8_bin,
`numero_documento` TEXT COLLATE utf8_bin NOT NULL,
`direccion` TEXT COLLATE utf8_bin,
`telefono` TEXT COLLATE utf8_bin
) ENGINE = InnoDB DEFAULT CHARSET = utf8 COLLATE = utf8_bin;
--

-- Volcado de datos para la tabla `proveedor`
--
INSERT INTO `proveedor`(`idProveedor`, `nombreProveedor`,
`tipo_documento`, `numero_documento`, `direccion`, `telefono`) VALUES( 1,
'Alicorp', 'DNI',
'20100055237', 'Av. Argentina 4793. Callao', '987523654' ),( 2, 'Ambrosoli',
'RUC', '20100060401', 'Av. Santa Felicia Nro 255 Lima-Ate', '970965715' ),( 3,
'Golosinas Trome', 'RUC', '20101088295', 'Av. San Isidro - Lima', '989798554'
),( 4, 'Labocer', 'RUC', '201000521734', 'Av. Grau - San Isidro', '963985274' ),(
5, 'Chocolates del Peru', 'RUC', '20514584789', 'Av. Maquinarias 2360- Lima',
'987852741' ),( 6, 'Golocenter', 'RUC', '20546821235', 'Av. Nicolas de Arriola
Nrov9 - Dpto 306', '013252874' ),( 7, 'Confiperu', 'RUC', '20258908849', 'Av.
Revolucion - ventanilla', '012523083' );
-----

--
-- ESTRUCTURA DE TABLA PARA LA TABLA `ROL`
--
CREATE TABLE `rol`(`idRol` INT(11) NOT NULL,
`nombre_rol` TEXT COLLATE utf8_bin NOT NULL

```

```

) ENGINE = InnoDB DEFAULT CHARSET = utf8 COLLATE = utf8_bin;
--

-- Volcado de datos para la tabla `rol`
--

INSERT INTO `rol`(`idRol`, `nombre_rol`) VALUES(1, 'Administrador'),(2, 'Venta'),(3, 'Almacen');
-----
--

-- ESTRUCTURA DE TABLA PARA LA TABLA `USUARIO`
--

CREATE TABLE `usuario`(
  `idUsuario` INT(11) NOT NULL,
  `nombre` TEXT COLLATE utf8_bin,
  `apellido_paterno` TEXT COLLATE utf8_bin,
  `apellido_materno` TEXT COLLATE utf8_bin,
  `username` TEXT COLLATE utf8_bin NOT NULL,
  `PASSWORD` TEXT COLLATE utf8_bin NOT NULL,
  `idRol` INT(11) NOT NULL
) ENGINE = InnoDB DEFAULT CHARSET = utf8 COLLATE = utf8_bin;
--

-- Volcado de datos para la tabla `usuario`
--

INSERT INTO `usuario`(`idUsuario`, `nombre`, `apellido_paterno`,
`apellido_materno`, `username`, `PASSWORD`, `idRol`) VALUES( 1, 'Carlos',
NULL, NULL, 'admin', 'admin', 1 ),(2, 'venta', "", "", 'venta', 'venta', 2),( 3,
'almacen', "", "", 'almacen', 'almacen', 3);
-----
--

-- Estructura para la vista `compras`
--

DROP TABLE IF EXISTS
  `compras`;
CREATE ALGORITHM = UNDEFINED DEFINER = `root`@`localhost` SQL SECURITY
DEFINER VIEW `compras` AS SELECT
  `compra`.`idCompra` AS `idCompra`,
  `detalle_compra`.`idDetalleCompra` AS `idDetalleCompra`,
  `compra`.`fecha` AS `fecha`,
  `producto`.`nombre_producto` AS `nombre_producto`,
  `compra`.`estado` AS `estado`,
  `detalle_compra`.`cantidad` AS `cantidad`,
  `detalle_compra`.`precio_unitario` AS `precio_unitario`,
  `detalle_compra`.`subtotal` AS `subtotal`
  FROM
  (
    (
      `detalle_compra`
      JOIN `compra` ON
      (
        (
          `compra`.`idCompra` = `detalle_compra`.`idCompra`
        )
      )
    )
  )
)

```

```

JOIN `producto` ON
  (
    (
      `producto`.`idProducto` = `detalle_compra`.`idProducto`
    )
  )
);
-----
--

-- Estructura para la vista `pedidos`
--

DROP TABLE IF EXISTS
`pedidos`;
CREATE ALGORITHM = UNDEFINED DEFINER = `root`@`localhost` SQL SECURITY
DEFINER VIEW `pedidos` AS SELECT
  `pedido`.`idPedido` AS `idPedido`,
  `detalle_pedido`.`idDetallePedido` AS `idDetallePedido`,
  `pedido`.`fecha` AS `fecha`,
  `producto`.`nombre_producto` AS `producto`,
  `pedido`.`estado` AS `estado`,
  `detalle_pedido`.`cantidad` AS `cantidad`,
  `detalle_pedido`.`precio_unitario` AS `precio_unitario`,
  `detalle_pedido`.`subtotal` AS `subtotal`
FROM
  (
    (
      (
        `detalle_pedido`
        JOIN `pedido` ON
          (
            (
              `pedido`.`idPedido` = `detalle_pedido`.`idPedido`
            )
          )
        )
      )
    )
  )
  JOIN `producto` ON
    (
      (
        (
          `producto`.`idProducto` = `detalle_pedido`.`idProducto`
        )
      )
    )
  );
--

-- Índices para tablas volcadas
--

--

-- Indices de la tabla `categoria`
--

ALTER TABLE
`categoria` ADD PRIMARY KEY(`idCategoria`);
--

-- Indices de la tabla `cliente`
--

```

```

ALTER TABLE
  `cliente` ADD PRIMARY KEY(`idCliente`);
--

-- Indices de la tabla `compra`
--

ALTER TABLE
  `compra` ADD PRIMARY KEY(`idCompra`),
  ADD KEY `idProveedor`(`idProveedor`),
  ADD KEY `idUsuario`(`idUsuario`);
--

-- Indices de la tabla `detalle_compra`
--

ALTER TABLE
  `detalle_compra` ADD PRIMARY KEY(`idDetalleCompra`),
  ADD KEY `idCompra`(`idCompra`),
  ADD KEY `idProducto`(`idProducto`);
--

-- Indices de la tabla `detalle_pedido`
--

ALTER TABLE
  `detalle_pedido` ADD PRIMARY KEY(`idDetallePedido`),
  ADD KEY `idPedido`(`idPedido`),
  ADD KEY `idProducto`(`idProducto`);
--

-- Indices de la tabla `fabricante`
--

ALTER TABLE
  `fabricante` ADD PRIMARY KEY(`idFabricante`);
--

-- Indices de la tabla `marca`
--

ALTER TABLE
  `marca` ADD PRIMARY KEY(`idMarca`);
--

-- Indices de la tabla `pedido`
--

ALTER TABLE
  `pedido` ADD PRIMARY KEY(`idPedido`),
  ADD KEY `idCliente`(`idCliente`),
  ADD KEY `idUsuario`(`idUsuario`);
--

-- Indices de la tabla `producto`
--

ALTER TABLE
  `producto` ADD PRIMARY KEY(`idProducto`),
  ADD KEY `idCategoria`(`idCategoria`),
  ADD KEY `idMarca`(`idMarca`),
  ADD KEY `idProveedor`(`idProveedor`),
  ADD KEY `idFabricante`(`idFabricante`);

```

```
--  
  
-- Indices de la tabla `proveedor`  
--  
ALTER TABLE  
  `proveedor` ADD PRIMARY KEY(`idProveedor`);  
--  
-- Indices de la tabla `rol`  
--  
ALTER TABLE  
  `rol` ADD PRIMARY KEY(`idRol`);  
--  
-- Indices de la tabla `usuario`  
--  
ALTER TABLE  
  `usuario` ADD PRIMARY KEY(`idUsuario`),  
  ADD KEY `idRol` (`idRol`);  
--  
  
-- AUTO_INCREMENT de las tablas volcadas  
--  
--  
-- AUTO_INCREMENT de la tabla `categoria`  
--  
ALTER TABLE  
  `categoria` MODIFY `idCategoria` INT(11) NOT NULL AUTO_INCREMENT,  
  AUTO_INCREMENT = 8;  
--  
-- AUTO_INCREMENT de la tabla `cliente`  
--  
ALTER TABLE  
  `cliente` MODIFY `idCliente` INT(11) NOT NULL AUTO_INCREMENT,  
  AUTO_INCREMENT = 8;  
--  
  
-- AUTO_INCREMENT de la tabla `compra`  
--  
ALTER TABLE  
  `compra` MODIFY `idCompra` INT(11) NOT NULL AUTO_INCREMENT,  
  AUTO_INCREMENT = 8;  
--  
  
-- AUTO_INCREMENT de la tabla `detalle_compra`  
--  
ALTER TABLE  
  `detalle_compra` MODIFY `idDetalleCompra` INT(11) NOT NULL AUTO_INCREMENT,  
  AUTO_INCREMENT = 8;  
--  
  
-- AUTO_INCREMENT de la tabla `detalle_pedido`  
--  
ALTER TABLE
```

```

`detalle_pedido` MODIFY `idDetallePedido` INT(11) NOT NULL AUTO_INCREMENT,
AUTO_INCREMENT = 16;
--

-- AUTO_INCREMENT de la tabla `fabricante`
--

ALTER TABLE
`fabricante` MODIFY `idFabricante` INT(11) NOT NULL AUTO_INCREMENT,
AUTO_INCREMENT = 8;
--

-- AUTO_INCREMENT de la tabla `marca`
--

ALTER TABLE
`marca` MODIFY `idMarca` INT(11) NOT NULL AUTO_INCREMENT,
AUTO_INCREMENT = 7;
--

-- AUTO_INCREMENT de la tabla `pedido`
--

ALTER TABLE
`pedido` MODIFY `idPedido` INT(11) NOT NULL AUTO_INCREMENT,
AUTO_INCREMENT = 15;
--

-- AUTO_INCREMENT de la tabla `producto`
--

ALTER TABLE
`producto` MODIFY `idProducto` INT(11) NOT NULL AUTO_INCREMENT,
AUTO_INCREMENT = 10;
-- AUTO_INCREMENT de la tabla `proveedor`

ALTER TABLE
`proveedor` MODIFY `idProveedor` INT(11) NOT NULL AUTO_INCREMENT,
AUTO_INCREMENT = 8;
--

-- AUTO_INCREMENT de la tabla `rol`
--

ALTER TABLE
`rol` MODIFY `idRol` INT(11) NOT NULL AUTO_INCREMENT,
AUTO_INCREMENT = 4;
--

-- AUTO_INCREMENT de la tabla `usuario`
--

ALTER TABLE
`usuario` MODIFY `idUsuario` INT(11) NOT NULL AUTO_INCREMENT,
AUTO_INCREMENT = 4;
--

-- Restricciones para tablas volcadas
-- Filtros para la tabla `compra`
ALTER TABLE
`compra` ADD CONSTRAINT `compra_ibfk_1` FOREIGN KEY(`idProveedor`)
REFERENCES `proveedor`(`idProveedor`),

```

```

    ADD CONSTRAINT `compra_ibfk_2` FOREIGN KEY(`idUsuario`) REFERENCES
`usuario`(`idUsuario`);
--

-- Filtros para la tabla `detalle_compra`
--

ALTER TABLE
`detalle_compra` ADD CONSTRAINT `detalle_compra_ibfk_1` FOREIGN KEY(`idCompra`)
REFERENCES `compra`(`idCompra`),
    ADD CONSTRAINT `detalle_compra_ibfk_2` FOREIGN KEY(`idProducto`) REFERENCES
`producto`(`idProducto`);
--

-- Filtros para la tabla `detalle_pedido`
--

ALTER TABLE
`detalle_pedido` ADD CONSTRAINT `detalle_pedido_ibfk_1` FOREIGN KEY(`idPedido`)
REFERENCES `pedido`(`idPedido`),
    ADD CONSTRAINT `detalle_pedido_ibfk_2` FOREIGN KEY(`idProducto`) REFERENCES
`producto`(`idProducto`);

-- Filtros para la tabla `pedido`

ALTER TABLE
`pedido` ADD CONSTRAINT `pedido_ibfk_1` FOREIGN KEY(`idCliente`) REFERENCES
`cliente`(`idCliente`),
    ADD CONSTRAINT `pedido_ibfk_2` FOREIGN KEY(`idUsuario`) REFERENCES
`usuario`(`idUsuario`);
--

-- Filtros para la tabla `producto`
--

ALTER TABLE
`producto` ADD CONSTRAINT `producto_ibfk_1` FOREIGN KEY(`idCategoria`)
REFERENCES `categoria`(`idCategoria`),
    ADD CONSTRAINT `producto_ibfk_2` FOREIGN KEY(`idMarca`) REFERENCES
`marca`(`idMarca`),
    ADD CONSTRAINT `producto_ibfk_3` FOREIGN KEY(`idProveedor`) REFERENCES
`proveedor`(`idProveedor`),
    ADD CONSTRAINT `producto_ibfk_4` FOREIGN KEY(`idFabricante`) REFERENCES
`fabricante`(`idFabricante`);
ALTER TABLE
`usuario` ADD CONSTRAINT `usuario_ibfk_1` FOREIGN KEY(`idRol`) REFERENCES
`rol`(`idRol`);
COMMIT
;
/*!40101
SET
CHARACTER_SET_CLIENT = @OLD_CHARACTER_SET_CLIENT */;
/*!40101
SET
CHARACTER_SET_RESULTS = @OLD_CHARACTER_SET_RESULTS */;
/*!40101
SET
COLLATION_CONNECTION = @OLD_COLLATION_CONNECTION */;

```

4.2. En base a los objetivos de la investigación

4.2.1. Determinar el método de costeo de inventario que se aplicará en el desarrollo del software

O1: Determinar el método que se aplicará para el desarrollo del software antes de la implementación del aplicativo a desarrollar.

O2: Determinar el método que se aplicará para el desarrollo del software después de utilizar el aplicativo desarrollado.

Se realizó un análisis donde se pudo obtener como resultado lo siguiente:

Antes O1: Para determinar el método que se aplicará en el proceso para el desarrollo del software transaccional es un tiempo promedio de 1 hora.

Después O2: Se realizó el aplicativo con el método de costeo más apropiado conocido como el método P.E.P.S(FIFO), con un tiempo promedio de 2 minutos.

Diferencia (O2 - O1): Para determinar el método que se aplicará para el proceso de desarrollo del software se identificó que el tiempo de duración fue de 1 hora.

Para el desarrollo del aplicativo se evaluó las diferentes alternativa dando como la mejor solución el método PEPS lo cual ha permitido mejor a aproximado de 2 minutos teniendo una diferencia de 57 minutos.

4.2.2. Desarrollar la aplicación móvil según los requerimientos definidos y el método de costeo seleccionado.

O1: No se contaba con el análisis y diseño antes de la implementación del aplicativo a desarrollar.

O2: Se realizó el análisis del diseño y funciones del APP después de utilizar el aplicativo desarrollado.

Se realizó un análisis y diseño donde se pudo obtener como resultado lo siguiente:

Antes O1: Para la realización de su sistema transacción no se había tomado un buen análisis y diseño de la solución.

Después O2: Luego de la implementación de los requerimientos definidos y el método establecido, se realizó en un tiempo promedio de 3 minutos.

Diferencia (O2 - O1): Durante el análisis y diseño de los requerimientos definidos se identificó que el tiempo de duración para los procesos fue de 2 horas. Ahora, luego de la implementación de los requerimientos definidos y el método establecido para los procesos del App se tiene un tiempo de 3 minutos teniendo una diferencia de 1 hora 56 minutos.

4.2.3. Validar la funcionalidad de la aplicación móvil, según los requerimientos definidos con los usuarios interesados.

O1: No se consideraba requerimientos definidos antes de la implementación del aplicativo a desarrollar.

O2: El App cumple con los requerimientos definidos después de utilizar el aplicativo desarrollado.

Se realizó el cumplimiento de los requerimientos definidos obteniendo lo sgte:

Antes O1: El tiempo aprox. era de 10 a 20 minutos para los procesos de venta y Kardex.

Después O2: Se brindó mayor eficacia y rapidez al momento de realizar sus procesos con un tiempo de 2 a 3 minutos.

Diferencia (O2 - O1): Durante el análisis acerca de la problemática y definición de los requerimientos de la empresa se observa que el tiempo en el proceso es de 10 a 20 minutos. Ahora, gracias a la validación de los requerimientos se realiza estos procesos con un tiempo de 3 minutos teniendo una diferencia de 17 minutos.

DISCUSIÓN

En base al Objetivo: “Determinar el método de costeo de inventario que se aplicará en el desarrollo del software”.

Para cumplir con el presente objetivo, se realizó el análisis de los tres métodos de costeo existentes, logrando definir la utilización del método P.E.P.S para la tesis. Según anexo 3 nos muestra la aprobación del uso de P.E.P.S con un grado de aprobación 4.85 con puntaje entre 10 y 9.

En base al Objetivo: “Desarrollar la aplicación móvil según los requerimientos definidos y el método de costeo seleccionado.”

Para cumplir con este objetivo, se realizó el diseño de los diagramas de análisis que están conformados por el Diagr. Contexto(DC), diagrama de realización de casos de uso y diagrama de clases generales y para la parte del diseño se desarrolló los diagramas de sub sistema de diseño de funciones a paquetes, diagrama de realización de caso de uso de diseño, diagrama de clases de diseño, base de datos física y lógica, diagrama de estado e interfaces y diagrama de despliegues. Según anexo 3 muestra un 99.954 % como grado de satisfacción de funciones del aplicativo por parte de los usuarios.

En base al Objetivo: “Validar la funcionalidad de la aplicación móvil, según los requerimientos definidos con los usuarios interesados.”.

Para cumplir con este objetivo, se realizó los diseños de interfaces, diagramas de despliegues, componentes y pseudocódigo.

Además de realizar las pruebas funcionales y de sistema como carga, rendimiento, volumen, seguridad y usuario. Según anexo 2 muestra un 100.00 % como grado de calidad de software con un promedio de 2.6 minutos en cada cumplimiento de objetivos.

V. CONCLUSIONES

1. Se ha llevado a cabo un análisis de los métodos de costeo más utilizados PEPS, UEPS y costo promedio, llegando a la conclusión que el método más apropiado es el método PEPS ya que nos permite saber el costo por separado de los movimientos del almacén tanto en entradas como en salidas.
2. Se ha implementado la totalidad de los requerimientos funcionales identificados y además se ha implementado el método PEPS logrando ser una aplicación que cubre las necesidades de la empresa.
3. Se ha mostrado la funcionalidad de la aplicación de acuerdo a los requerimientos definidos pudiendo afirmar que el aplicativo cumple al 100 % los requerimientos de la empresa reduciendo en 3 minutos los procesos.

VI. RECOMENDACIONES

1. Para llevar a cabo la puesta en marcha la empresa, deberá hacer la compra de equipos de celulares, dominio y hosting.
2. Se recomienda solo instalar el aplicativo a personal autorizado evitando así filtración de información.
3. Brindar un monitoreo continuo para evitar el incumplimiento de malos procesos y mejorar así el control interno.
4. Capacitar a cada uno los empleados sobre el aplicativo y el uso de la nueva tecnología.
5. En el caso que existan nuevos cambios según requerimientos por parte de la empresa hacia la aplicación móvil, se deberá actualizar nuevas versiones que permitan el correcto funcionamiento.

VII. LISTA DE REFERENCIAS

- [1] Gestión, 2018. Nivel Óptimo de Inventario <https://www.slideshare.net/ManuelRojas49/nivel-optimo-de-inventario>
- [2] Reportedigital. 2018. Cómo emplear las nuevas tecnologías en el servicio al cliente, <https://reportedigital.com/transformacion-digital/nuevas-tecnologias>
- [3] Software de comunicaciones, 2014. Arquitectura Android, <https://sites.google.com/site/swcuc3m/home/android/generalidades/2-2-arquitectura-de-android>
- [4] Manuel Rojas, 2018. Nivel óptimo de inventario, <https://www.slideshare.net/ManuelRojas49/nivel-optimo-de-inventario>
- [5] Revista el Comercio, 2016. "Consumo per cápita de golosinas en Perú es de 3,5 kilos al año", El Comercio, 2019. <https://elcomercio.pe/economia/peru/consumo-per-capita-golosinas-peru-3-5-kilos-ano-225536-noticia/>
- [6] Isabel Cristina, José Boada. 2017. Ruta de soluciones para la gestión de inventarios en pymes del sector retail que comercialicen productos de alto volumen, con miras a respaldar su crecimiento en ventas. Revista Espacios, 2017. <https://www.revistaespacios.com/a17v38n53/17385313.html>
- [7] Doyler Gamaliel Mairena Delgado, Marvin Enmanuel Vallejos Méndez. 2017. Evaluación de los procesos de control de inventario y facturación para su automatización en farmacia Siloé. Universidad Nacional Autónoma de Nicaragua. Managua.
- [8] Iris Jazmin Guadalupe Balarezo. 2015. Análisis, diseño e implementación de un Sistema de control de Inventario. Universidad de Guayaquil. Ecuador.
- [9] Leydi Roxana Velásquez Valle, Carlos Alberto Zeledón Bustillo. 2014. Sistema de inventario y facturación de la tienda de accesorios de computadoras y celulares "Decosys". Universidad Nacional Autónoma de Nicaragua. Nicaragua.
- [10] Manuel Quitanilla Uribe. 2014. Desarrollo del Sstema de Conrol de Inentarios de la Unversidad de San Marín de Pores. Título Pregrado.
- [11] David Oscana. 2016. Dearrollo de un Aplcación Anroid para inrementar la paticipación en medios digitales de una Unversidad Privada. Título pregrado.
- [12] Claudia María del Pilar Zapata Del Río. 2013. Análisis, diseño e implementación de una aplicación para administrar y consultar avisos clasificados para tabletas android., Título pregrado.
- [13] Almendra Solange de María Oblitas Guevara. 2016. Aplicación Móvil. Guía para orientar al turista en su estadía por la región Lambayeque. Título pregrado.
- [14] Esther Elizabeth Bustamante Gamarra, Klary Marlit Lozano Ruiz. 2015. Desarrollo de aplicación web basado en el modelo de revisión continua y utilizando la tecnología RFID para mejorar la gestión de inventarios de vehículos automotores menores en la Empresa Lima Motor S.R.L , Título de pregrado.
- [15] Vargas Apaestegui David Alexis. 2018. Aplicación web adaptativa para mejorar el control de inventarios de confecciones "ANGIARA". Utilizando la planificación de los recursos de materiales. Tesis Pregrado.

- [16] Finney – Miller.1999. CURSO DE CONTABILIDAD INTERMEDIA I. Tomos 3 y 4. Biblioteca de Contabilidad Superior. Teoría y Material de Práctica.
- [17] Jonson, Robert W. 2016. ADMINISTRACION FINANCIERA – Capítulo Administración de Inventarios.
- [18] JQuery mobile. 2014. About: JQuery Mobile. 2016. <http://jquerymobile.com/about/>.
- [19] EVA Software & Hardware Design. 2019. "Conoce los tipos de app que existen y para qué sirven - EVA Software & Hardware Design". <http://evainnova.com/app/conoce-los-tipos-de-app-que-existen-y-para-que-sirven/>.
- [20] AreaTecnologia. 2016. Sistemas Operativos Moviles Para Celulares, Areatecnologia.com, 2019. [Online]. Available: <https://www.areatecnologia.com/informatica/sistemas-operativos-moviles.html>.
- [21] Servisoftcorp.2017. Definición y cómo funcionan las aplicaciones móviles, <https://servisoftcorp.com/definicion-y-como-funcionan-las-aplicaciones-moviles/>
- [22] GenBeta. 2011. JQuery Mobile: Introducción al desarrollo web para móviles. 27 de Abril de 2011. <http://www.genbetadev.com/frameworks/jquerymobile-introduccion-al-desarrollo-web-para-moviles>.
- [23] Vittone, José, y Javier Cuello. 2015. Diseñando apps para móviles.
- [24] Pressman, Roger S. 2010 Ingeniería del Software. Un enfoque práctico
- [25] Dpto de Ciencias de la Computación e I.A. 2010.Artículo, Universidad de Granada.

VIII. ANEXOS

ANEXO N° 01: MANUAL DE USUARIO

PROTOTIPO DEL DISEÑO

A continuación, las presentes imágenes son referenciales, donde detallaremos las funciones que debe realizar el aplicativo móvil con tecnología Android, utilizando un sistema de costeo de inventario que permitirá el apoyo para el proceso de ventas en la empresa Dicopersac Chiclayo, Presentamos a continuación cada uno de las ventanas con sus respectivos controles que se serán utilizados en el presente sistema.

Cada representante de venta podrá visualizar dos elementos importantes tales como:

- ✓ Objetos: Es el centro de atención del usuario (Cualquier objeto que pueda manipular)
- ✓ Acciones: Permiten al representante crear o manipular objetos

ACCESO AL SISTEMA

Usuario Ventas

Al ingresar a la interfaz como ventas, podemos visualizar el panel de control los siguientes accesos, permitiendo realizar los siguientes movimientos como: Ventas, Compras, Kardex, Devoluciones, Producto, Proveedor, Cliente, Fabricante, Categoría y Marca.

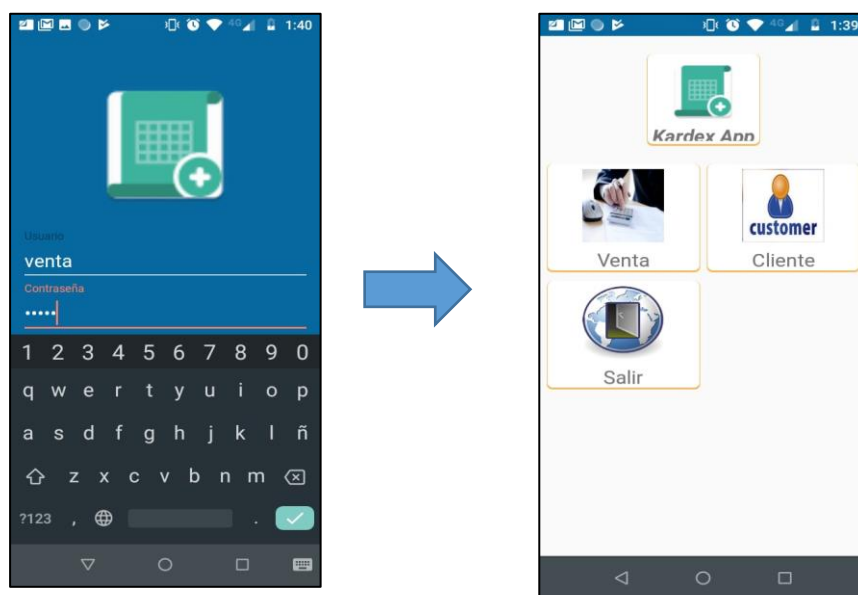


Fig. 105: Interfaz de inicio de Sesión Venta Android –

Usuario Almacen

AL ingresar a la interfaz como Almacén, podemos visualizar el panel de control los siguientes accesos, permitiendo realizar los siguientes movimientos como : Ventas, Compras, Kardex, Devoluciones, Producto, Proveedor, Cliente, Fabricante, Categoría y Marca

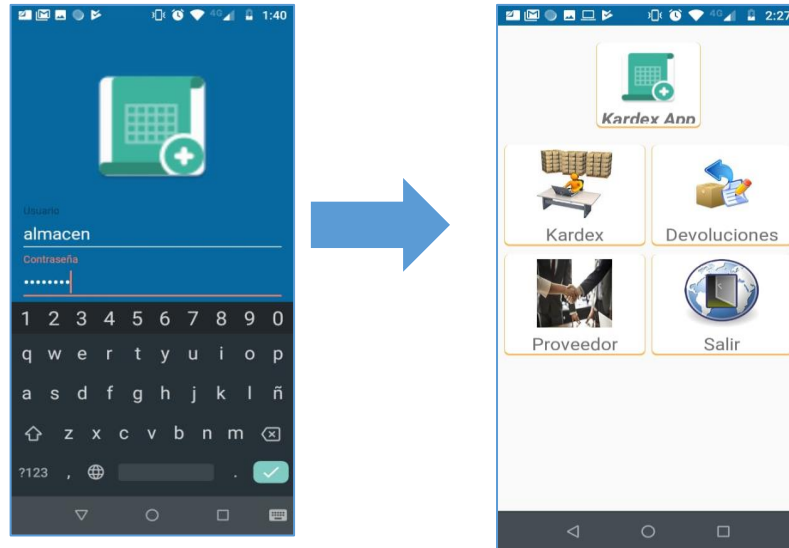


Fig. 106 Interfaz de inicio de sesión Almacén Android –Elaboración propia

Usuario Administrador

AL ingresar a la interfaz como administrador, podemos visualizar el panel de control, permitiendo realizar los siguientes movimientos como : Ventas, Compras, Kardex, Devoluciones, Producto, Proveedor, Cliente, Fabricante, Categoría y Marca.

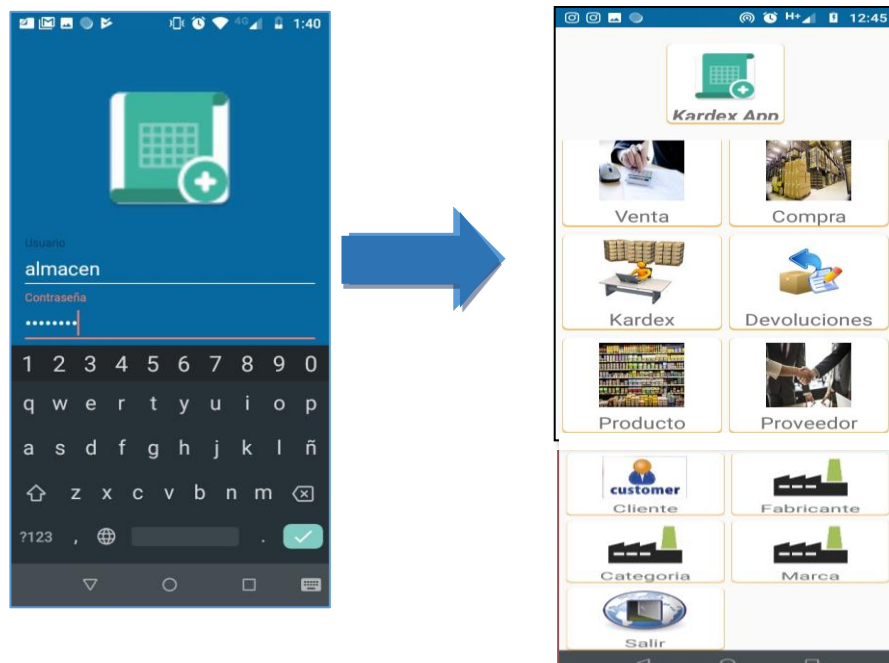


Fig. 107: Interfaz de inicio de sesión – Elaboración propia

REGISTRAR PROVEEDOR

Esta interfaz permite visualizar la lista de proveedores (Interfaz izquierda) así como crear nuevos proveedores (interfaz Derecha), editar, considerando como campo obligatorio el nombre del proveedor, dirección, RUC y su teléfono.

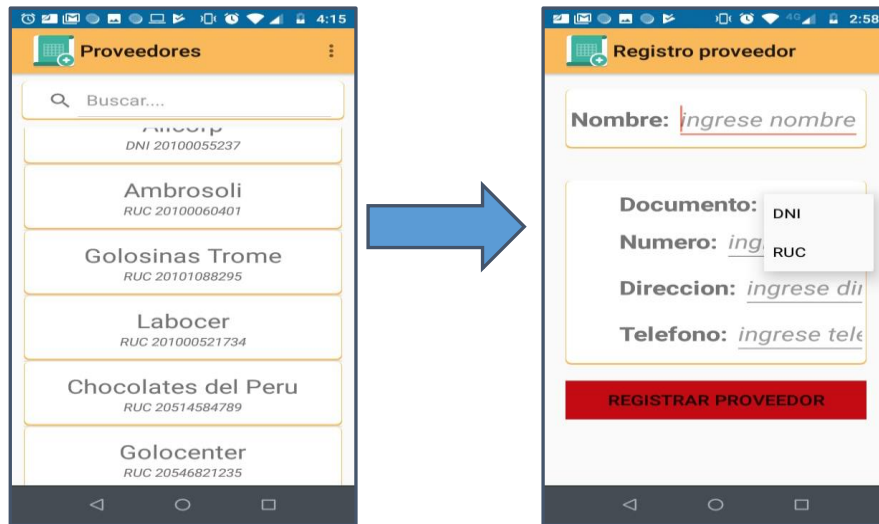


Fig. 108: Interfaz lista de Proveedores-Registrar Proveedores –Elaboración propia

REGISTRA CLIENTE

Esta interfaz permite visualizar la lista de clientes-Bodegas (Interfaz izquierda), así como crear nuevos clientes (interfaz Central), considerando como campo obligatorio el tipo de documento ya sea RUC o DNI, el nombre del cliente (Interfaz Derecha).

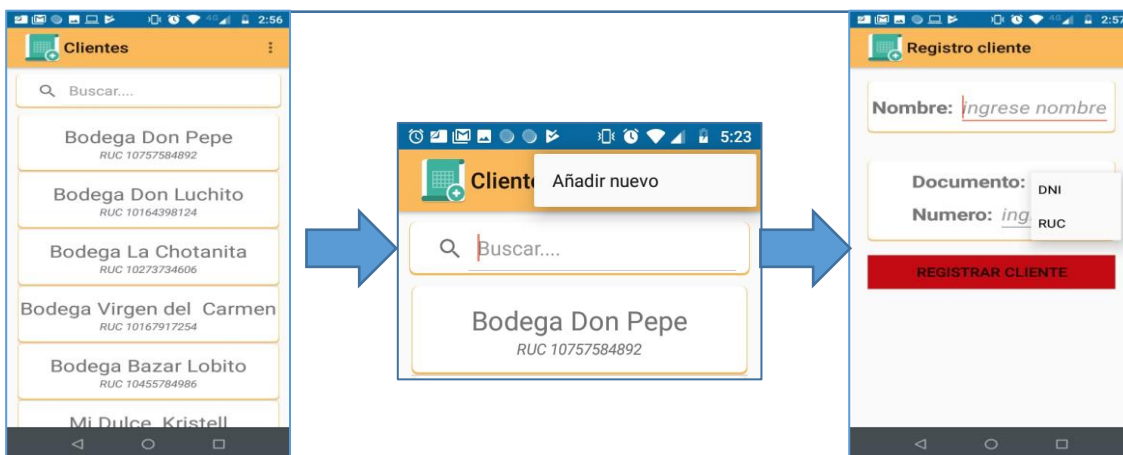


Fig. 109: Interfaz lista de clientes- Registrar Cliente – Elaboración propia

REGISTRAR PRODUCTO

Esta interfaz permite visualizar el listado de los productos existentes en el almacén con su respectivo stock de cada producto, (Interfaz izquierda), así como crear nuevos productos (interfaz central), considerando como campos obligatorios los siguientes datos: Categoría, Marca, Fabricante, Unidad de medida, precio y cantidad de cada producto (Interfaz Derecha).

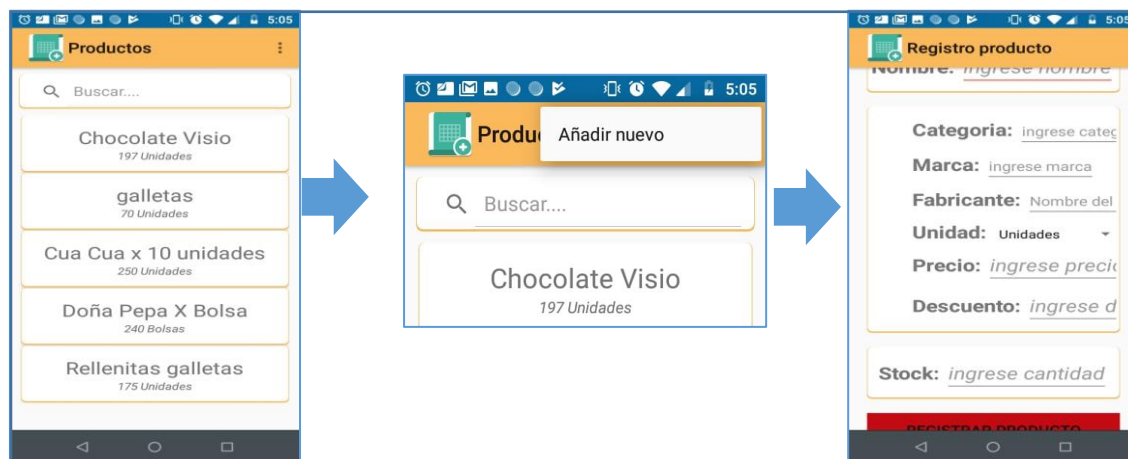


Fig. 110: Interfaz lista de productos-Registrar Productos- Elaboración propia

REGISTRAR CATEGORIA DE PRODUCTOS

Esta interfaz permite visualizar el listado de las categorías de los productos (Interfaz izquierda), así como crear nuevas categorías (interfaz central), considerando como campo obligatorios el siguiente dato : Marca de Categoría de cada producto según el proveedor(Interfaz Derecha)

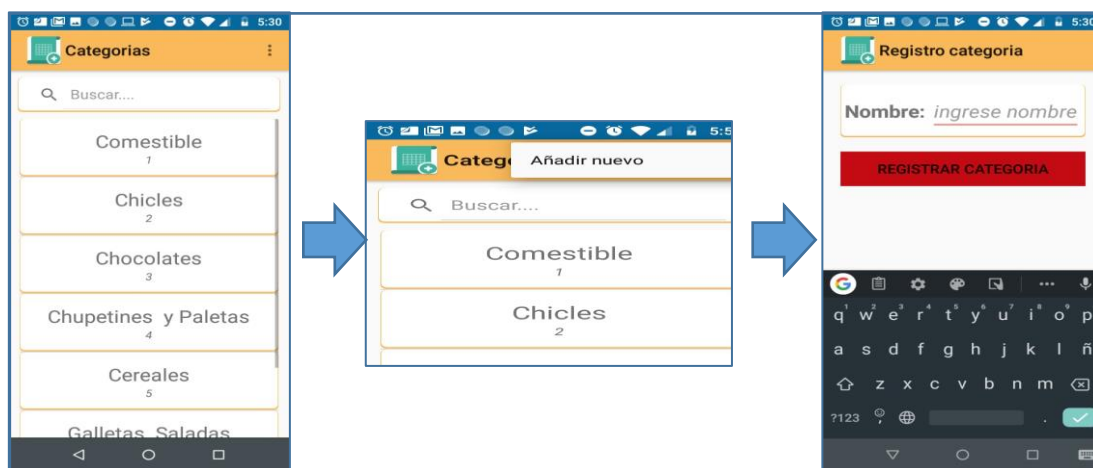


Fig. 111: Interfaz lista de productos –Elaboración propia

REGISTRAR MARCA DE PRODUCTO

Esta interfaz permite visualizar el listado de las marcas de los productos (Interfaz izquierda), así como crear nuevos productos (interfaz central), considerando como campo obligatorio el siguiente dato : Marca de Categoría de cada producto según el proveedor (Interfaz Derecha).

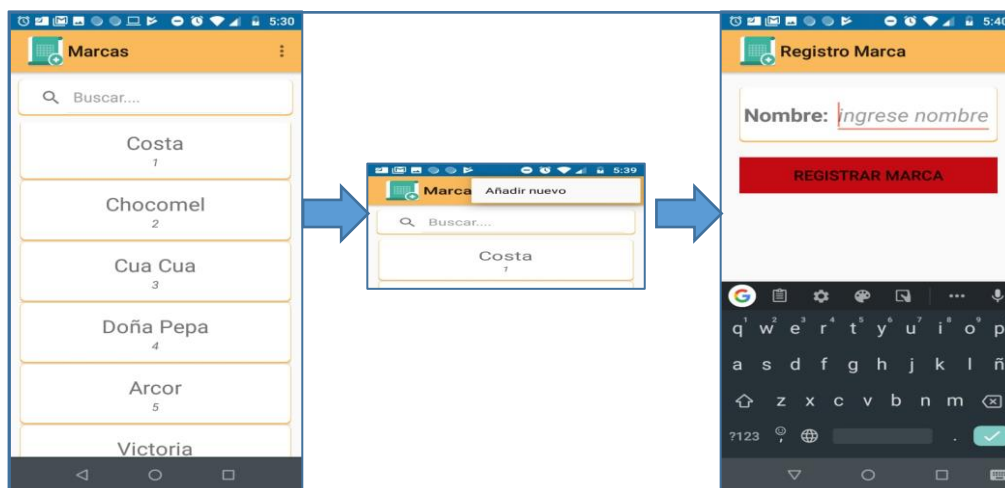


Fig. 112: Interfaz de marcas de productos- Elaboración propia

DEVOLUCIONES DE COMPRAS (FACTURA-BOLETAS)

Al ingresar a esta interfaz podemos visualizar dos opciones :

- a) Compras Registradas
- b) Ventas Registradas



Fig. 113: Interfaz de devoluciones - Elaboración propia

A.- COMPRAS REGISTRADAS .- Esta interfaz permite visualizar la opción de compras registradas de los productos en el almacén mediante factura (Interfaz izquierda). Al seleccionar la opción Compras registradas podemos visualizar un listado de las Facturas de la cual se han realizado las compras de los productos (Interfaz Central). En caso de que exista devolución se puede eliminar la Factura según el Numero de Factura (toda la factura) como política de la empresa (Interfaz Derecha)



Fig. 114: Interfaz de Compras registras-Elaboración propia

B.- VENTAS REGISTRADAS .- Esta interfaz permite visualizar dos opciones: Boletas de Ventas y Facturas de venta de los productos en el almacén (Interfaz izquierda). Al seleccionar la opción Boleta de Ventas podemos visualizar un listado de las Boletas emitidas al realizar las ventas (Interfaz Central). En caso de que exista devolución de venta se puede eliminar la Boleta según el Numero de Boleta (toda la Boleta) como política de la empresa(Interfaz Derecha)

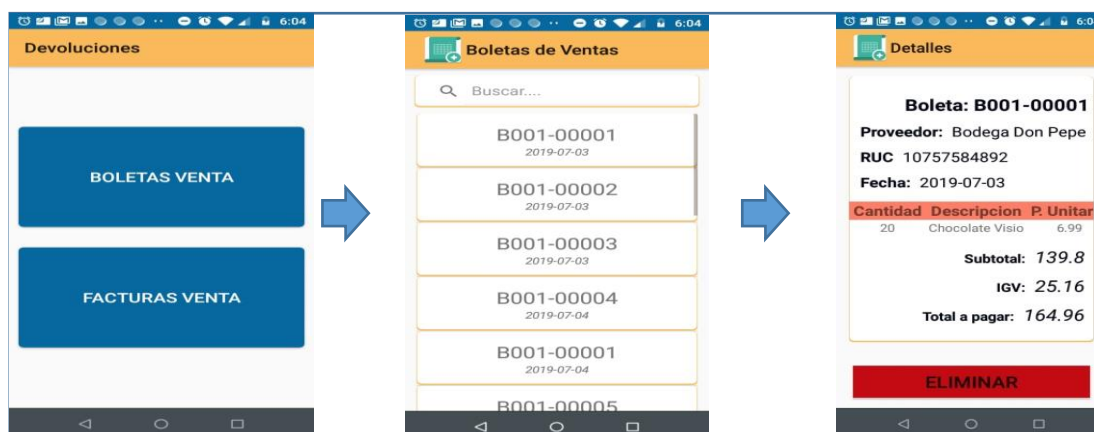


Fig. 115: Interfaz de Ventas registrada- Elaboración propia

KARDEX

Esta interfaz permite visualizar el listado de los productos existentes en el almacén con su stock de cada uno (Interfaz izquierda).



Fig. 116: Interfaz de Ventas registrada

AL seleccionar o al buscar cualquiera de los productos debera abrimos un excel que permita ver la cantidades de compras y salidas de los productos haciendo uso del Método PEPS.

FECHA	TIPO	entradas			salida			existencias		
		CANTIDAD COMPRA	costo unitario	TOTAL COMPRA	CANTIDAD COMPRA	costo unitario	TOTAL COMPRA	CANTIDAD	costo unitario	TOTAL COMPRA
2019-07-01	Saldo Inicial							250	15	3750
2019-07-05	Compra 30 unidades x 15 Soles según F/ 001	30	15	450				250	15	3750
2019-07-05	Venta de 20 unidades por 20soles				20	15	300	30	15	450
2019-07-05	Compra de 10 Unidades x 15 soles	10	15	150				230	15	3450
								30	15	450
2019-07-05	Venta de 25 unidades x 15 soles				25	15	375	10	15	150
								30	15	450
2019-07-05	Devolucion de compra de 30 unidades según F/001	-30	15	-450				10	15	150
								175	15	2625
								30	15	450
								10	15	150
	Venta de 15 unidades x 18 soles según Bol-002				15	15	225	30	15	450
								10	15	150
2019-07-05	venta de 25 unidades x 18 soles				25	15	375	30	15	450
								10	15	150
	Devolucion de 15 unidades por 15 soles según Bol -002				-15	15	-225	30	15	450
								10	15	150
2019-07-05	VENDO 25 UNIDADES POR 15 SOLES				25	15	375	30	15	450
								10	15	150
								125	15	1875

*Fig. 117: Interfaz de Ventas registrada
Android - Elaboración propia*

ANEXO N° 01: INSTRUMENTOS DE RECOLECCIÓN DE DATOS

N° Usuarios	Grado de cumplimiento funciones del aplicativo antes de implementar el aplicativo	Grado de cumplimiento funciones del aplicativo después de implementar el aplicativo
1	60'	5'
2	40'	4'
3	60'	5'
4	50'	4'
5	50'	5'
6	60'	5'
7	60'	5'
8	50'	5'
9	40'	5'
10	40'	5'
Tiempo Promedio		4'8''
Grado de cumplimiento		90.400

N° Usuarios	Grado de satisfacción de los usuarios antes de implementar el aplicativo	Grado de satisfacción de los usuarios después de implementar el aplicativo
1	60'	5'
2	60'	2'
3	60'	2'
4	120'	2'
5	60'	2'
6	120'	2'
7	60'	5'
8	100'	5'
9	60'	5'
10	60'	5'
Tiempo Promedio		3'5''
Grado de satisfacción		99.954

N° Usuarios	Grado cumplimiento de objetivos antes de implementar el aplicativo	Grado cumplimiento de objetivos después de implementar el aplicativo
1	120'	3'
2	120'	3'
3	60'	2'
4	120'	2'
5	80'	2'
6	120'	2'
7	120'	3'
8	120'	3'
9	70'	3'
10	60'	3'
Tiempo Promedio		2'6''
Grado de calidad de software		99.00

ANEXO N° 02: ENCUESTA

1. El resultado del aplicativo logro sus expectativas.
 - a. Si
 - b. No
 - c. Otro (.....).

2. El desarrollo del aplicativo llegó a los resultados esperados, logrando mejorar los procesos.
 - a. Si
 - b. No
 - c. Otro (.....).

3. El aplicativo determinó ser la mejor solución a los problemas de mi empresa.
 - a. Si.
 - b. No.

4. El aplicativo logro los objetivos de su trabajo.
 - a. Si
 - b. No

5. Califica al aplicativo en una escala del 1 al 5 donde el 1 representa la más baja calificación y el 5 la más alta.
 - a. 1
 - b. 2
 - c. 3
 - d. 4
 - e. 5

RESULTADO: Listado pruebas funcionales.- las pruebas funcionales se centran en verificar el funcionamiento de los requerimientos de la aplicación desarrollada, cuadros detallados Iteración #3: Análisis Preliminar de Requerimientos – Casos de Uso

Id	Nombre	Cumple	No Cumple	%
01	Inicio de Sesión Vendedor	SI	-	12.5
01	Inicio de Sesión Administrador	SI	-	12.5
02	Registrar Cliente	SI	-	12.5
02	Registrar Compra	SI	-	12.5
02	Registrar Venta	SI	-	12.5
02	Kardex	SI		12.5
03	Consultar producto	SI	-	12.5
04	Gestionar Pedido	SI		12.5
Porcentaje				100 %

ANEXO N° 03: INSTRUMENTOS DE RECOLECCIÓN DE DATOS

METODO CUALITATIVO POR PUNTOS							
Factor	Peso	GRUPO A		GRUPO B		GRUPO C	
		Calificación	Ponderación	Calificación	Ponderación	Calificación	Ponderación
COSTO PROMEDIO	15.00%	2	0.3	3	0.45	2	0.3
PEPS o FIFO	70.00%	5	3.5	5	3.5	5	3.5
UEPS o LIFO	15.00%	1	0.15	2	0.3	2	0.3
Totales	100.00%		3.95		4.25		4.1

Se realizaron tres grupos de encuestas

Formula Ponderación: $\text{Peso} / \text{Calificación} = \text{Ponderación}$

**ANEXO N° 04. CONSTANCIA DE APROBACIÓN DEL PRODUCTO
ACREDITABLE DE LA ENTIDAD DONDE SE EJECUTÓ LA TESIS**

CONSTANCIA DE APROBACIÓN

Por el presente documento, la Sra. Coronel Perales Carmen, en calidad de Gerente General, con domicilio Fiscal Cal. Manuel Suarez Nro. 133-Chiclayo, de la empresa comercial Dicopersac:

**HACE CONSTANCIA DEL PRODUCTO
ACREDITABLE PARA SU TESIS**

EL Sr. **EFFIO GONZALES Carlos Alberto**, de la Universidad Santo Toribio de Mogrovejo", identificado con DNI **40478699**, se presentó a nuestro establecimiento para dar solución a la problemática que se viene presentando en nuestra empresa referente al SISTEMA DE INVENTARIO.

El Sr. Effio, ha sido autorizado para realizar el levantamiento de información en materia de estudio para su tesis denominado "**APLICATIVO MÓVIL CON TECNOLOGÍA ANDROID UTILIZANDO EL MÉTODO PEPS DE INVENTARIO PARA APOYAR EL PROCESO DE VENTAS EN LA EMPRESA DICOPERSAC CHICLAYO**", dejando constancia que el producto acreditable tiene la aprobación correspondiente.

Se expide el presente documento para los fines que se estime conveniente

ANEXO 05: ANÁLISIS RIESGOS

– Matriz salvaguarda de riesgos

Entre los planes de mitigación para superar riesgos identificados en esta etapa se mencionan:

Fuente: Elaboración propia

Código del riesgo	Amenaza / Oportunidad	Descripción del riesgo	Fase	Nivel de riesgo	Tipo de respuesta	Responsable	Plan de mitigación
RE1 – 001	Amenaza	Definir los requerimientos y elección del método que se aplicará para el desarrollo de aplicativo del producto software	1	MUY ALTO	Salvaguarda	Tesista	<ul style="list-style-type: none"> ✓ Investigación de métodos ✓ importancia ✓ ventajas ✓ grado de eficiencia en los procesos
RE1 – 002	Amenaza / Oportunidad	Realizar el análisis y el diseño del sistema en función a los requerimientos que se han podido identificar en el negocio	2	ALTO	Salvaguarda	Tesista	<ul style="list-style-type: none"> ✓ desarrollo del análisis de los requerimientos ✓ diseño de prototipo del diseño ✓ asegurar cumplimiento y de App
RE1 – 004	Amenaza / Oportunidad	Elaborar el App Móvil utilizando la metodología PEPS.	3	MEDIO	Salvaguarda	Tesista	<ul style="list-style-type: none"> ✓ validar el cumplimiento de objetivos ✓ validar la satisfacción de los usuarios ✓ Desarrollar la programación ✓ Desarrollar base de datos ✓ Establecer funciones
RE1 – 003	Amenaza / Oportunidad	Evaluar el cumplimiento de los objetivos de la tesis con las pruebas pertinentes del software.	4	BAJO	Salvaguarda	Tesista	<ul style="list-style-type: none"> ✓ Comprobar la funcionalidad ✓ Satisfacer los objetivos ✓ Evaluar la calidad del software