

INSTITUTO TECNOLÓGICO DE MINATITLÁN

INGENIERÍA EN SISTEMAS COMPUTACIONALES

“Proyecto SIABA”

PRESENTA:

- 18230721 - Dominguez Tovar Israel David
- 18230724 - Gamas Michael
- 18230748 - Pozos Miranda Luis Eduardo
- 17230772 - Rodriguez Gonzalez Aleydis Gabriel



Minatitlán, Ver., a 26 de septiembre del 2021

INDICE

PRESENTACIÓN DEL PROYECTO.....	1
INFORMACIÓN DEL PROYECTO.....	1
ANÁLISIS DE RECOPIACIÓN DE LA INFORMACIÓN.....	2
ENTREVISTA.....	2
REVISIÓN DE REGISTROS.....	2
REQUISITOS FUNCIONALES Y NO FUNCIONALES.....	6
LISTA DE REQUERIMIENTOS FUNCIONALES Y NO FUNCIONALES.....	6
MATRIZ DE TRAZABILIDAD.....	7
DOCUMENTACIÓN DE REQUISITOS.....	9
DIAGRAMA DEL DOMINIO (DIAGRAMA DE CLASES).....	15
DC REPUVE.....	15
MODELO DEL NEGOCIO (DIAGRAMAS DE CASO DE USO).....	27
DCU CONSULTAS.....	27
ACTORES.....	28
CASOS DE USO.....	29
DCU GESTIONAR INVENTARIO.....	31
ACTORES.....	31
CASOS DE USO.....	32
DCU INICIO DE SESIÓN.....	35
ACTORES.....	35
DCU REGISTRO.....	38
ACTORES.....	38
CASOS DE USO.....	39
ESTUDIO DE FACTIBILIDAD.....	41
FACTIBILIDAD ECONÓMICA.....	41

FACTIBILIDAD TÉCNICA.....	42
FACTIBILIDAD OPERATIVA.....	43
COSTO-BENEFICIO DEL SISTEMA.....	44
ANEXOS.....	45
ANEXO A.....	45

Presentación del proyecto

Información del proyecto

Título formal del proyecto: Sistema de abarrotes

Título informal del proyecto: SIABA

Objetivo general: Diseñar e implementar un sistema que permita gestionar de forma eficiente las entradas y salidas de productos dentro de la microempresa “Pily” para la automatización del inventario.

Objetivos específicos:

- Analizar los requerimientos de sistematización dentro de la microempresa “Pily” para la gestión de inventarios del almacén.
- Diseñar una base de datos y la interfaz del sistema de inventarios dentro de la microempresa “Pily” para el gestionamiento de los productos.
- Desarrollar una aplicación para crear, actualizar y eliminar la información de la base de datos, para mejorar la eficacia en el proceso de ventas o pedidos de los productos.
- Implementar el sistema SIABA para el gestionamiento del inventario y las entradas y salidas de los productos en la microempresa “Pily”.

Descripción del proyecto:

La microempresa “Pily” lleva un control manual, donde es anotado todo en hojas recicladas, las hojas tienen de encabezado los productos en existencia dentro del almacén y el total de los productos que hay (stock).

Va comparando los productos que hay con los que se vendieron, los empleados tienen el deber de anotar todo lo que vendieron en cantidad y con su respectivo precio, para que el administrador sume el total de lo vendido y verifique que este coincida con el dinero que está en caja al momento de hacer el corte, dicho corte es realizado diariamente.

En el control de proveedores, cada proveedor le proporciona un ticket al administrador, donde se muestra los productos y la cantidad que el proveedor está entregando, también se lleva el control de los tickets clasificándolos en una libreta (Barcel, Sabritas, Ricolino, Coca-Cola, etc).

Todos los artículos que ingresen se van al almacén por lo que tienen que ser agregados en la hoja donde maneja el stock.

Análisis de recopilación de la información.

Entrevista

Se aplicó una entrevista al administrador de la microempresa “Pily” **Ver anexo A.** Analizamos que el cliente tiene problemas para administrar su negocio de manera eficaz y eficiente, actualmente cuenta con un formato que elaboró en papel y lápiz, dónde apunta cantidad, producto y costo, este mismo es el formato para el conteo de inventario.

Ademas, tiene un formato para las ventas del negocio, ahí lleva el registro diario pagina por pagina. Al hacer la entrevista nos dimos cuenta de las necesidades de nuestro cliente, recopilamos información para el desarrollo del sistema.

Ahora sabemos como se hacen todos los procesos del negocio, quien es el responsable de esos procesos, que información debe almacenar y información acerca de la interfaz del sistema.

Revisión de registros

En la figura 1 notamos que el cliente en hojas recicladas, lleva un formato de inventario, en dónde en el encabezado, él maneja el nombre del artículo y debajo, va poniendo el número de artículos que se encuentran al finalizar el día. El cliente cuenta lo de su almacén y lo que tiene en exhibición.

ESQUEMA DE MAYOR										NOMBRE DEL ALUMNO									
Inventario										FECHA									
chips	takis	tostitos	tostitos	tostitos	wapa	runes	Chipo	Big	pop	Caca	dulce	Maxi	Galle						
12	12	2	2	2	3	3	2	8	2	15	47	5	20	4	2				
12	12	2	2	2	3	3	2	8	2	14	44	5	20	4	2				
11	12	2	2	2	3	3	2	8	2	13	43	5	19	4	2				
12	12	2	4	3	3	3	3	8	1	15	51	3	10	4	2				
11	11	1	3	3	3	3	3	8	1	14	49	3	9	4	2				
11	10	1	3	3	3	2	3	8	2	13	46	3	20	3	4				
11	10	1	1	3	3	2	3	6	2	13	44	3	18	3	4				

Figura 1. Registro de inventario.

De ahí va comparando con su lista de ventas diarias para hacer un arqueo de las ventas diarias, y que cuadre con la existencia del producto, en la Figura 2, podemos notar como lleva a cabo sus registros de ventas diarias.

Sabado 7 Mayo 2021

Sabritas y Galletas

- 1 Sabritas = 13 ✓
- 1 Emparedor = 13 ✓
- 1 Tostitos = 10 ✓
- 1 Ruffles = 13 ✓
- 1 Papiros mados = 13 ✓
- 1 big mix = 12 ✓
- 1 Pizetazo = 13 ✓
- 1 Tostitos = 10 ✓
- 1 dulce = 1 ✓
- 1 Galleta = 13 ✓

~~832~~
832

Bebidas y caracateres.

- 2 benetton \$ 6 = 12 ✓
- 1 concordador = 7 ✓
- 1 mandet = 20 ✓
- 1 Pizetazo = 6 ✓
- 1 mathy = 5 ✓
- 1 Pizetazo = 13 ✓

1 mandet 500 = 20 ✓

2 Coca 1 1/4

3 Coca 2.5 L.

Bebidas y Aguas

- 2 benetton 10 ✓
- 1 Agua 1 L. = 12 ✓
- 1 mandet 20 ✓
- 1 Coca 400 = 10 ✓
- 1 Agua 1 L. = 12 ✓
- 1 Coca 1 1/4 = 19 ✓
- 1 Coca 2 L R = 28 ✓
- 1 Coca 2.5 L R = 28 ✓
- 1 benetton = 13 ✓
- 1 Coca 3 L R = 33 ✓
- 4 Coca 500m Vidn = 48 ✓
- 1 Coca 600 = 15 ✓
- 1 Coca 600 = 15 ✓
- 1 Coca 1 1/4 = 19 ✓
- 1 mandet = 20 ✓
- 1 Coca 3 L = 33 ✓
- 1 Coca 2.5 = 28 ✓
- 1 Cigarras R = 10 ✓
- 1 Cajetilla H.R = 70 ✓
- 6 Vasos = 6 ✓
- 2 Vasos = 2 ✓
- 1 Coca 1 1/4 = 19 ✓
- 1 mandet = 20 ✓
- 1 Coca 2 = 33 ✓
- 1 Cajetilla = 70 ✓
- 1 benetton = 5 ✓
- 1 Coca 2.5 = 28 ✓
- 1 Coca 3 L = 33 ✓
- 1 Tostitos = 22 ✓
- 1 Tostitos = 13 ✓
- 1 Coca 3 L R = 33 ✓
- 1 Pizetazo = 13 ✓

681

855

Figura 2. Registro de ventas.

Respecto al control de los proveedores, cada uno de ellos le proporcionan un ticket dónde le muestra la cantidad y precio que se adquirieron. La manera de

controlar y archivar esos tickets es guardándolos en una libreta, en dónde los clasifica por la marca del producto.

MINATITLAN
 RFC: PRO-84042358
 Mario Pani 100, Piso 7
 Fecha: 03/04/2021 Hora: 10:11:50
 Bodega: MR0220 Ruta: M00364
 Vendedor: Intelliroute
 Documento: 637530393010302735

Cliente: 220052332
 PILAR NOHEMI MIRANDA LOZANO
 FCO. SARABIA 28-A

ENTREGA: 05/04/2021
 RUTA DE REPARTO: RM0108

OPERACION			
ENTREGA			
CVE	CANT	DESCRIPCION	IMPORTE
360		COCA COLA 600 ML 24 PK	
1/0	298.00	0.00	\$297.99
501		COCA COLA 2.5L RET PET 8PK	
2/0	198.50	0.00	\$397.00
673		CC COLA 1.25 VIDRIO RET 8PK	
2/0	134.50	0.00	\$268.99
2352		CC COLA BOT 3L RET PET 6p	
1/0	177.00	0.00	\$176.99
97182		GENEROSA MANZANA 413 ML 6PK	
1/0	67.20	0.00	\$67.20
97185		GENEROSA DURAZNO 413 ML 6PK	
1/0	67.20	0.00	\$67.20
8/0		TOTAL ENTREGA	\$1,275.37

DEVOLUCION			
CVE	CANT	DESCRIPCION	
170450	2/0	CAJA PLASTICA 8 CAV. 1.2	
170451	2/0	CAJA PLASTICA 8 CAV. 2.5	
170460	2/0	ENV CC 2.5LTS REFPET8	
170464	2/0	ENV VRET CC 1.25L (880GR	
171036	1/0	CAJA PLASTICA 6 CAV. 3L	
171102	1/0	ENV CC 3 L REF PET 6PK	
10/0		TOTAL ENVASE DEVUELTO	

BALANCE TOTAL: CONTADO \$1,275.37

Descuentos aplican antes de impuestos
 Nota de venta no representa saldo
 en estado de cuenta
 Puedes obtener tu factura en el portal
clienteskf.mx/CFDIportalclientes/
 Si tienes problema para facturar marca
 800-2233-672

envia mensaje al correo
suportefacturacion@csc.femsa.com.mx

Quejas Sugerencias Atención a clientes
 CENTRO DE CONTACTO COCA-COLA FEMSA
 800-2233-672

!! GRACIAS POR TU COMPRA !!

FIRMA/SELLO DEL CLIENTE

Figura 3. Tickets de pedido.

Al momento que se adquiere mercancía nueva, el administrador los ingresa a su hoja de inventario, sin embargo, como el cliente no cuenta con una buena administración, divide el almacén en dos partes, los productos exhibidos y los que cuenta en almacén.

ESQUEMA DE MAYOR		ESQUEMA DE MAYOR									
<u>Bolzano</u>		6 6									
<u>Pabellazo</u>		6 4 4 15 13 18 11 9 12 11									
<u>Sabonitas y Puffles</u>		25 24 24 28 26 24 23 21 27 23									
<u>Puntos y roncantes</u>		12 11 11 24 24 23 20 19 21 21									
<u>Fritos y crojitos</u>		1 1 1 8 8 8 7 6 6 6									
<u>Chetos y churros</u>		14 12 11 32 30 28 26 26 34 32									
<u>Yakult</u>		8 8 6 14 14 13 5 5 13 9									
2	Jugo vidno	8 8 7 6 6 6 6 8 8 8									
6	delaware ch.	6 6 6 5 5 5 7 7 7 5									
8	delaware lata	4 3 3 5 5 5 5 5 5 6									
1	coca lata	3 3 3 5 5 4 4 5 5 6									
	pirada	- - - 4 2 10 10 13 10									
	power	5 6 6 3 2 2 7 7 8 8									
	agua lt	11 5 5 4 4 4 5 4 4 4									
87	agua 600	4 4 4 4 4 4 6 6 6 5									
6	Sabor 600	6 4 17 17 16 12 11 15 13									
8	Sabor 2L	2 2 2 3 3 3 3 3 3 3									
21	Sidral 2L	3 3 2 - - 2 2 3 2									
13	coca lt	5 5 5 5 5 5 5 5 5 2									
	coca 600	19 14 14 10 8 8 12 10 18 14									
93	coca 400	6 6 6 8 8 7 3 2 8 8									
273	coca 2.5	8 20 18 8 9 15 8 7 8 7									
7	Coca 3L	7 17 10 8 15 12 6 8 5 6									
	coca 3L desech	2 2 2 3 2 1 3 3 3 3									
6	coca vidrio 500	6 6 6 3 2 16 5 4 5 5									
	coca 11/4	5 11 11 3 8 8 5 2 2 3									

Figura 4. Registro de inventario.

En el negocio son 3 personas los que laboran ahí, el administrador también realiza actividades de empleado, sin embargo, es el único que puede agregar productos nuevos, modificar precios o eliminar productos, además de añadir productos nuevos para el negocio. También es el único que puede dar de baja del inventario mercancía dañada.

Requisitos funcionales y no funcionales

Lista de requerimientos funcionales y no funcionales

REQUERIMIENTOS FUNCIONALES

- Autenticación de usuario.
- Registrar usuarios.
- Consultar información.
- Menú de inicio.
- Gestionar inventario.
- Generar reportes.
- Registro de datos

REQUERIMIENTOS NO FUNCIONALES

- Interfaz del sistema.
- Soporte del sistema.
- Nivel de usuario.
- Confiabilidad del sistema.
- Seguridad de información.

Matriz de trazabilidad

	MATRIZ DE TRAZABILIDAD DE REQUISITOS	
		PAG. 1 DE 1

		Fecha edición	14/09/2021
TÍTULO DEL PROYECTO	Sistema de ventas para abarrotes (SIABA)	Código Proyecto	#####

ESTADO DE LOS REQUISITOS DEL PROYECTO

ID	REQUISITO	TIPO	PRIO	ESTADO	OBJETIVO	ENTREGABLE(s)	ESTADO (Entreg.)	VALIDACIÓN (persona/fecha)
RF01	Autenticación de usuario	RN	Alta	Activo	Restringir las acciones de unos usuarios a otros	Módulo con inicio de sesión		Pilar Nohemi Miranda Lozano
RF02	Registrar usuarios	RP	Alta	Activo	Almacenar información de los usuarios y crear su cuenta para acceder al sistema	Módulo donde se registrarán los usuarios		Pilar Nohemi Miranda Lozano
RF03	Consultar información	RN	Media	Activo	El usuario podrá ver información acerca de su negocio	Módulo donde se generarán las consultas		Pilar Nohemi Miranda Lozano
RF04	Menú de inicio	RP	Media	Activo	Tener un control en los productos de bajo stock	Un menú al iniciar sesión		Pilar Nohemi Miranda Lozano
RF05	Gestionar inventario	RN	Alta	Activo	Tener un control sobre las entradas y	Módulo para automatización		Pilar Nohemi Miranda Lozano

					salidas	de inventario		
RF06	Generar reportes	RN	Alta	Activo	Generar reportes de tablas específicas	Módulo donde se generarán reportes		Pilar Nohemi Miranda Lozano
RF07	Registro de datos	RN	Alta	Activo	Registrar información en tablas específicas	Módulo para el registro general de datos		Pilar Nohemi Miranda Lozano
RFN01	Interfaz del sistema	RP	Baja	Activo	Ser sencilla de usar y conectar con todos los usuarios	Interfaz de usuario		Pilar Nohemi Miranda Lozano
RFN02	Soporte del sistema	RC	Baja	Activo	Ofrecer ayuda a los futuros usuarios	Manual de usuario		Pilar Nohemi Miranda Lozano
RFN03	Nivel de usuario	RN	Alta	Activo	Acceder a la información de acuerdo al rol del usuario	Acceso a la información de acuerdo al usuario		Pilar Nohemi Miranda Lozano
RFN04	Confiabilidad del sistema	RC	Alta	Activo	Un sistema activo	Backup constante		Pilar Nohemi Miranda Lozano
RFN05	Seguridad de información	RC	Alta	Activo	Un sistema donde se almacenen de manera segura documentación, archivos, contraseñas, etc.	Sistema con inicio de sesión		Pilar Nohemi Miranda Lozano

RN	Requerimientos del negocio	RP	Requerimientos del proyecto	RI	Requerimientos de las partes interesadas
RI	Requerimientos de las partes interesadas	RC	Requerimientos de calidad		

Documentación de requisitos

Requisitos Funcionales

Número del requisito:	RF01
Nombre del requisito:	Autenticación de usuario
Características:	Los usuarios deberán identificarse con un usuario y contraseña para acceder al sistema.
Descripción del requisito:	<p>Usuario:</p> <p>Descripción paso a paso de como se autenticará en el sistema:</p> <ol style="list-style-type: none">1. Deberá estar registrado en el sistema, en caso contrario deberá cubrir el RF02.2. Si ya tiene una cuenta, deberá ingresar su usuario: RFC y contraseña.3. Si los datos son correctos, entra al sistema mostrando el menú principal4. Si los datos son incorrectos, tendrá un máximo de 3 intentos para acceder.5. Si supera el numero máximo de intentos, tendrá que llamar al administrador para un cambio de contraseña.
Prioridad del requisito:	Alta/Esencial Media/Deseado Baja/Opcional

Número del requisito:	RF02
Nombre del requisito:	Registrar usuarios
Características:	Los usuarios deberán registrarse en el sistema para tener una cuenta y con esto acceder al sistema.
Descripción del requisito:	El sistema permitirá al usuario (Administrador, empleado) registrarse.

	<ul style="list-style-type: none"> • El usuario administrador debe ingresar datos como: RFC, Nombre, Apellido paterno, Apellido materno, E-mail, Contraseña. • El usuario empleado debe ingresar datos como: RFC, Nombre, Apellido paterno, Apellido materno, Numero telefónico, Calle, No. Calle, Colonia, Ciudad, Copia de INE (PDF), Solicitud de empleo(PDF), Comprobante de domicilio(PDF) y contraseña.
Prioridad del requisito:	Alta/Esencial Media/Deseado Baja/Opcional

Número del requisito:	RF03
Nombre del requisito:	Consultar información
Características:	El sistema ofrecerá información general acerca de los productos, ventas, pedidos y proveedores.
Descripción del requisito:	<p>Consultar productos: Muestra información de todos los productos como su nombre, precio de compra, precio de venta, el stock y unidad de medida.</p> <p>Consultar ventas: permite al usuario ver las ventas realizadas y mostrar datos como fecha, monto total y el usuario que realizo la venta.</p> <p>Consultar pedidos: Muestra a los usuarios información de los pedidos realizados como fecha, monto total, quien realizó el pedido y quien es el proveedor.</p> <p>Consultar proveedores: Permite a los usuarios ver información acerca de los proveedores como el nombre del proveedor.</p> <p>Consulta prestamos: Permite al usuario visualizar los prestamos hechos por los usuarios, mostrando datos como el id del prestamos, concepto del préstamo, monto del préstamo, fecha y hora.</p>
Prioridad del requisito:	Alta/Esencial Media/Deseado Baja/Opcional

Número del requisito:	RF04
Nombre del requisito:	Menú de inicio
Características:	El sistema ofrecerá una pantalla de inicio donde se observarán los productos que tienen un bajo stock.
Descripción del requisito:	Al ingresar al sistema cubriendo el RF01, el usuario visualizará una ventana donde se encontrarán los productos que tienen un bajo stock, mostrando el nombre del producto, el precio de compra y el stock.
Prioridad del requisito:	Alta/Esencial Media/Deseado Baja/Opcional

Número del requisito:	RF05
Nombre del requisito:	Gestionar inventario
Características:	El sistema ofrecerá una automatización al vender algún producto o realizar un pedido.
Descripción del requisito:	Al momento de realizar una venta el sistema automáticamente restará la cantidad vendida del producto en su stock. Al momento de recibir un pedido se dan de alta los productos esto provocará que el stock aumente conforme a la cantidad del producto pedido.
Prioridad del requisito:	Alta/Esencial Media/Deseado Baja/Opcional

Número del requisito:	RF06
Nombre del requisito:	Generar Reportes
Características:	El sistema permitirá la generación de reportes de diferentes tablas.
Descripción del requisito:	<p>El usuario podrá generar reportes de:</p> <ul style="list-style-type: none"> • Ventas • Pedidos • Prestamos • Inventarios
Prioridad del requisito:	Alta/Esencial Media/Deseado Baja/Opcional

Número del requisito:	RF07
Nombre del requisito:	Registro de datos
Características:	El sistema permitirá el registro de datos dentro de las tablas de la base de datos.
Descripción del requisito:	<p>El usuario podrá registrar datos en las tablas de:</p> <ul style="list-style-type: none"> • Unidades de medida • Proveedores • Roles • Ventas • Pedidos • Prestamos • Productos al inventario
Prioridad del requisito:	Alta/Esencial Media/Deseado Baja/Opcional

Requerimientos no funcionales

Número del requisito:	RFN01		
Nombre del requisito:	Interfaz del sistema		
Características:	El sistema presentará una interfaz sencilla para que sea de fácil aprendizaje y manejo a los usuarios del sistema.		
Descripción del requisito:	El sistema tendrá una interfaz de uso intuitiva y sencilla.		
Prioridad del requisito:	Alta/Esencial	Media/Deseado	Baja/Opcional

Número del requisito:	RFN02		
Nombre del requisito:	Soporte del sistema		
Características:	El sistema tendrá un manual de usuario para facilitar el uso del sistema.		
Descripción del requisito:	Le permitirá al usuario disponer de documentación que le permita hacer uso del sistema.		
Prioridad del requisito:	Alta/Esencial	Media/Deseado	Baja/Opcional

Número del requisito:	RFN03		
Nombre del requisito:	Nivel de usuario		
Características:	Garantizar al usuario el acceso de información de acuerdo al nivel que posee.		
Descripción del requisito:	Facilidades y controles para permitir el acceso a la información a los usuarios autorizados con la intención de consultar y modificar la información de acuerdo al usuario.		

Prioridad del requisito:	Alta/Esencial	Media/Deseado	Baja/Opcional
---------------------------------	---------------	---------------	---------------

Número del requisito:	RFN04		
Nombre del requisito:	Confiabilidad del sistema		
Características:	El sistema tendrá que estar en funcionamiento 12 horas los 7 días de la semana.		
Descripción del requisito:	La disponibilidad del sistema debe ser continua durante las 12 horas garantizando contar con una contingencia por una posible falla. Por lo tanto, el sistema contará con un proceso de backup diario.		
Prioridad del requisito:	Alta/Esencial	Media/Deseado	Baja/Opcional

Número del requisito:	RFN05		
Nombre del requisito:	Seguridad de información		
Características:	El sistema garantiza seguridad a los usuarios una seguridad en cuanto a la información que se procede en el sistema.		
Descripción del requisito:	Garantizar la seguridad del sistema con respecto a la información almacenada como documentos, contraseñas y datos personales.		
Prioridad del requisito:	Alta/Esencial	Media/Deseado	Baja/Opcional

Diagrama del dominio (diagrama de clases)

DC REPUVE

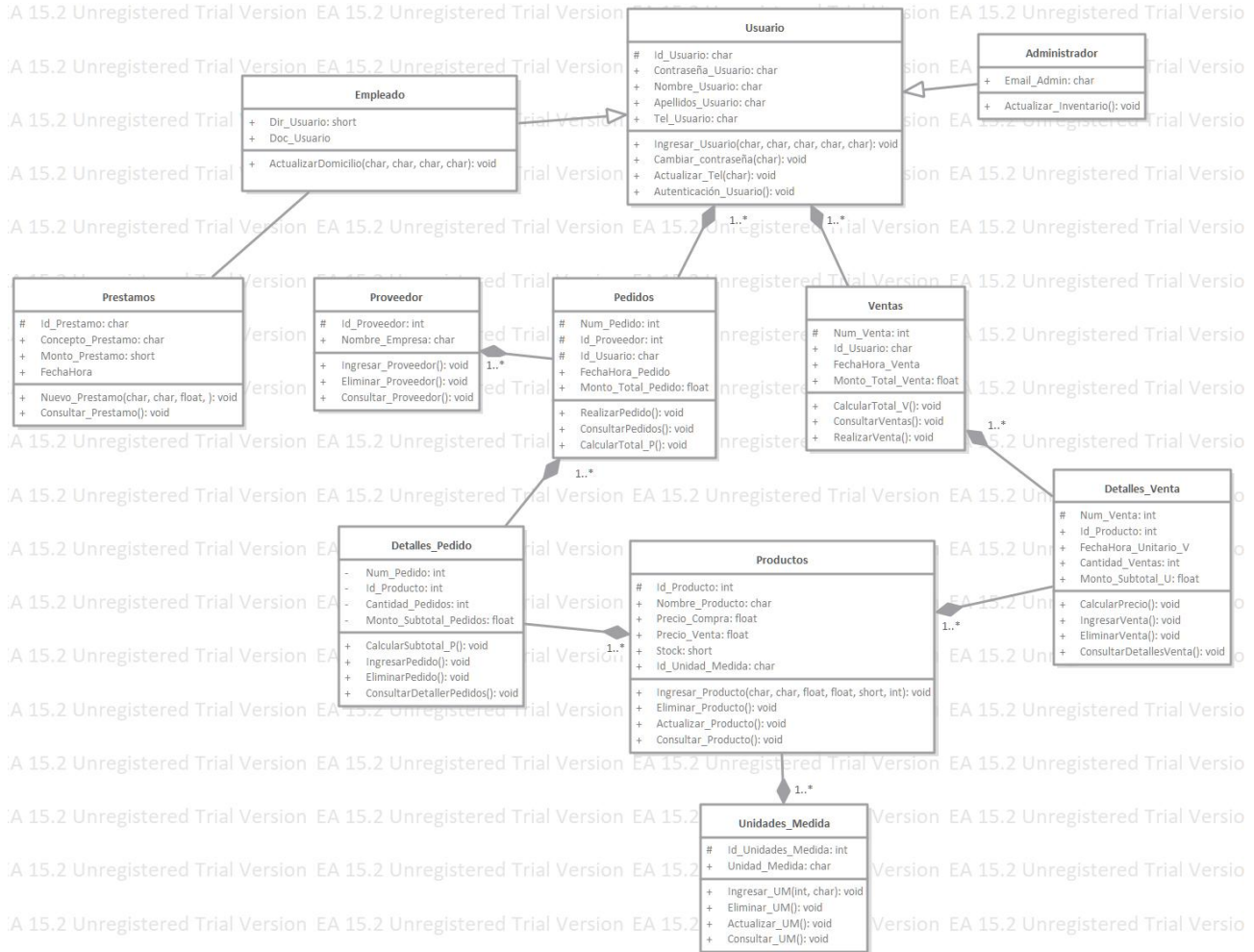


Figura 5. DC REPUVE.

Administrador



Persona dueña del negocio, al ser el administrador tendrá privilegios sobre el empleado





RELACIÓN
 Generalización de Administrador a Usuario [Direction is 'Source -> Destination'.]
ATRIBUTOS
 Email_Admin : char Public [Is static True. Containment is Not Specified.]
MÉTODOS
 Actualizar_Inventario () : void Public [Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]

Detalles_Pedido

Representa el cuerpo de un pedido, con este vemos los detalles de los productos que se pidieron al proveedor

RELACIÓN
 Agregación de Detalles_Pedido a Pedidos [Direction is 'Source -> Destination'.]
 Agregación de Detalles_Pedido a Productos [Direction is 'Source -> Destination'.]
ATRIBUTOS
 Num_Pedido : int Private [Is static True. Containment is Not Specified.]
 Id_Producto : int Private [Is static True. Containment is Not Specified.]

ATRIBUTOS	
 Cantidad_Pedidos : int Private	[Is static True. Containment is Not Specified.]
 Monto_Subtotal_Pedidos : float Private	[Is static True. Containment is Not Specified.]

MÉTODOS	
 CalcularSubtotal_P () : void Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
 IngresarPedido () : void Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
 EliminarPedido () : void Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
 ConsultarDetallerPedidos () : void Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]

Detalles_Venta

Representa el cuerpo de una venta, con esta vemos los detalles de los productos que se vendieron en una venta

RELACIÓN	
 Agregación de Detalles_Venta a Productos	[Direction is 'Source -> Destination'.]
 Agregación de Detalles_Venta a Ventas	[Direction is 'Source -> Destination'.]

ATRIBUTOS	
 Num_Venta : int Protected	[Is static True. Containment is Not Specified.]



ATRIBUTOS	
◆ Id_Producto : int Public	[Is static True. Containment is Not Specified.]
◆ FechaHora_Unitario_V : Public	[Is static True. Containment is Not Specified.]
◆ Cantidad_Ventas : int Public	[Is static True. Containment is Not Specified.]
◆ Monto_Subtotal_U : float Public	[Is static True. Containment is Not Specified.]

MÉTODOS	
◆ CalcularPrecio () : void Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
◆ IngresarVenta () : void Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
◆ EliminarVenta () : void Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
◆ ConsultarDetallesVenta () : void Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]


Empleado

Persona que trabaja para el administrador, tendrá acceso a la información de la base de datos de acuerdo a su nivel de acceso.

RELACIÓN	
← Generalización de Empleado a Usuario	[Direction is 'Source -> Destination'.]

ATRIBUTOS	
 Dir_Usuario : short Public	[Is static True. Containment is Not Specified.]
 Doc_Usuario : Public	[Is static True. Containment is Not Specified.]

ASOCIACIONES	
 Association (direction: Unspecified)	
Source: Public (Class) Empleado	Target: Public (Class) Prestamos


MÉTODOS	
 ActualizarDomicilio (CALLE_USUARIO : char , NO_CALLE_USUARIO : char , COLONIA_USUARIO : char , CIUDAD_USUARIO : char) : void Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]

Pedidos




Representa la cabecera de los pedidos con esta identificamos un pedido

RELACIÓN	
 Agregación de Pedidos a Proveedor	[Direction is 'Source -> Destination'.]
 Agregación de Pedidos a Usuario	[Direction is 'Source -> Destination'.]

RELACIONES	
 Agregación de Detalles_Pedido a Pedidos	[Direction is 'Source -> Destination'.]




ATRIBUTOS	
 Num_Pedido : int Protected	[Is static True. Containment is Not Specified.]

ATRIBUTOS	
 Id_Proveedor : int Protected	[Is static True. Containment is Not Specified.]
 Id_Usuario : char Protected	[Is static True. Containment is Not Specified.]
 FechaHora_Pedido : Public	[Is static True. Containment is Not Specified.]
 Monto_Total_Pedido : float Public	[Is static True. Containment is Not Specified.]

MÉTODOS	
 RealizarPedido () : void Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
 ConsultarPedidos () : void Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
 CalcularTotal_P () : void Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]

Prestamos

Representa los prestamos que le hace el administrador al empleado

ATRIBUTOS	
 Id_Prestamo : char Protected	[Is static True. Containment is Not Specified.]
 Concepto_Prestamo : char Public	[Is static True. Containment is Not Specified.]
 Monto_Prestamo : short Public	[Is static True. Containment is Not Specified.]


ATRIBUTOS	
 FechaHora : Public	[Is static True. Containment is Not Specified.]



ASOCIACIONES	
 Association (direction: Unspecified)	
Source: Public (Class) Empleado	Target: Public (Class) Prestamos

MÉTODOS	
 Nuevo_Prestamo (Id_Prestamo : char , Concepto_Prestamo : char , Monto_Prestamo : float , FechaHora_Prestamo :) : void Public	
[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]	
 Consultar_Prestamo () : void Public	
[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]	

Productos

Representa los productos que hay en el negocio

RELACIÓN	
 Agregación de Productos a Unidades_Medida	[Direction is 'Source -> Destination'.]

RELACIONES	
 Agregación de Detalles_Venta a Productos	[Direction is 'Source -> Destination'.]
 Agregación de Detalles_Pedido a Productos	[Direction is 'Source -> Destination'.]

ATRIBUTOS	

ATRIBUTOS	
◆ Id_Producto : int Protected	[Is static True. Containment is Not Specified.]
◆ Nombre_Producto : char Public	[Is static True. Containment is Not Specified.]
◆ Precio_Compra : float Public	[Is static True. Containment is Not Specified.]
◆ Precio_Venta : float Public	[Is static True. Containment is Not Specified.]
◆ Stock : short Public	[Is static True. Containment is Not Specified.]
◆ Id_Unidad_Medida : char Public	[Is static True. Containment is Not Specified.]

MÉTODOS	
◆ Ingresar_Producto (Id_producto : char , Nombre_Producto : char , Precio_Compra : float , Precio_Venta : float , Stock : short , Unidad_Medida : int) : void Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
◆ Eliminar_Producto () : void Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
◆ Actualizar_Producto () : void Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
◆ Consultar_Producto () : void Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]

Proveedor

Empresa que surte los productos por medio de un pedido

RELACIONES	
⇒ Agregación de Pedidos a Proveedor	[Direction is 'Source -> Destination'.]

ATRIBUTOS	
◆ Id_Proveedor : int Protected	[Is static True. Containment is Not Specified.]
◆ Nombre_Empresa : char Public	[Is static True. Containment is Not Specified.]

MÉTODOS	
◆ Ingresar_Proveedor () : void Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
◆ Eliminar_Proveedor () : void Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
◆ Consultar_Proveedor () : void Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]

Unidades_Medida

Representa las unidades de medida de los productos, como si fuera una categoría. Ejemplo: Mililitros, Litros, Piezas, Kilos, etc.

RELACIONES	
⇒ Agregación de Productos a Unidades_Medida	[Direction is 'Source -> Destination'.]

ATRIBUTOS	
◆ Id_Unidades_Medida : int Protected	[Is static True. Containment is Not Specified.]
◆ Unidad_Medida : char Public	[Is static True. Containment is Not Specified.]

MÉTODOS	
◆ Ingresar_UM (Id_Unidad_Medida : int , Unidades_Medida : char) : void Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
◆ Eliminar_UM () : void Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
◆ Actualizar_UM () : void Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
◆ Consultar_UM () : void Public	[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]

Usuario

Representa la persona que es usuario del sistema, está es una generalización de los dos posibles usuarios: Administrador y empleado

RELACIONES	
⇒ Agregación de Ventas a Usuario	[Direction is 'Source -> Destination'.]
⇒ Generalización de Administrador a Usuario	[Direction is 'Source -> Destination'.]
⇒ Generalización de Empleado a Usuario	[Direction is 'Source -> Destination'.]
⇒ Agregación de Pedidos a Usuario	[Direction is 'Source -> Destination'.]

ATRIBUTOS

◆ Id_Usuario : char Protected

[Is static True. Containment is Not Specified.]

◆ Contraseña_Usuario : char Public

[Is static True. Containment is Not Specified.]

◆ Nombre_Usuario : char Public

[Is static True. Containment is Not Specified.]

◆ Apellidos_Usuario : char Public

[Is static True. Containment is Not Specified.]

◆ Tel_Usuario : char Public

[Is static True. Containment is Not Specified.]

MÉTODOS

◆ Ingresar_Usuario (NOMBRE_USUARIO : char , APELLIDO_P_USUARIO : char , APELLIDO_M_USUARIO : char , ID_USUARIO : char , TELEFONO_USUARIO : char) : void Public

[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]

◆ Cambiar_contraseña (NuevaContraseña : char) : void Public

[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]

◆ Actualizar_Tel (NuevoTelefono : char) : void Public

[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]

◆ Autenticación_Usuario () : void Public

[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]

Ventas

Representa la cabecera de las ventas con esta identificamos una venta

RELACIÓN

← Agregación de Ventas a Usuario

[Direction is 'Source -> Destination'.]

RELACIONES

⇒ Agregación de Detalles_Venta a Ventas

[Direction is 'Source -> Destination'.]

ATRIBUTOS

◆ Num_Venta : int Protected

[Is static True. Containment is Not Specified.]

◆ Id_Usuario : char Public

[Is static True. Containment is Not Specified.]

◆ FechaHora_Venta : Public

[Is static True. Containment is Not Specified.]

◆ Monto_Total_Venta : float Public

[Is static True. Containment is Not Specified.]

MÉTODOS

◆ CalcularTotal_V () : void Public

[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]

◆ ConsultarVentas () : void Public

[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]

◆ RealizarVenta () : void Public

[Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]

Modelo del negocio (Diagramas de caso de uso)

DCU Consultas

Diagrama de casos de uso que representa la visualización de información en forma de consultas y reportes

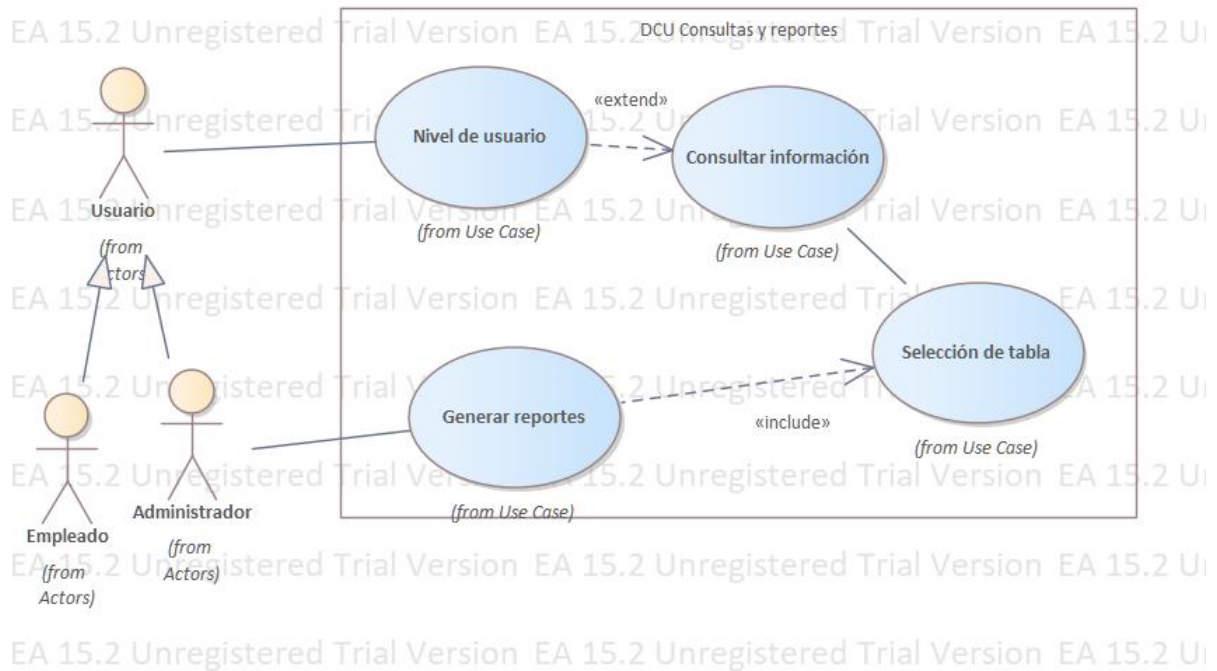


Figura 6. DCU Consultas

Actores

Administrador

Persona dueña del negocio

RELACIÓN	
← Generalización de Administrador a Usuario	[Direction is 'Source -> Destination'.]

ASOCIACIONES	
✎ Association (direction: Unspecified)	
Source: Public (Actor) Administrador	Target: Public (UseCase) Generar reportes

Empleado

Persona que trabaja para el administrador

RELACIÓN	
← Generalización de Empleado a Usuario	[Direction is 'Source -> Destination'.]

Usuario

Representa a las personas por igual, estas interactúan con el sistema, hay dos tipos de usuario: El administrador y empleado.

RELACIONES	
⇒ Generalización de Administrador a Usuario	[Direction is 'Source -> Destination'.]
⇒ Generalización de Empleado a Usuario	[Direction is 'Source -> Destination'.]

ASOCIACIONES

 Association (direction: Unspecified)

Source: Public (Actor) Usuario


Target: Public (UseCase) Nivel de usuario

Casos de uso

Consultar información

Representan las consultas que se harán por medio de las tablas de la base de datos

CONECTORES

 **Extend** «extend» Source -> Destination
De: Nivel de usuario : UseCase, Public
a: Consultar información : UseCase, Public

ASOCIACIONES

 Association (direction: Unspecified)


Source: Public (UseCase) Consultar información

Target: Public (UseCase) Selección de tabla

Generar reportes

Representa los reportes que serán generados solo por el administrador.

CONECTORES

 **Include** «include» Source -> Destination
De: Generar reportes : UseCase, Public
a: Selección de tabla : UseCase, Public

ASOCIACIONES

 Association (direction: Unspecified)


Source: Public (Actor) Administrador

Target: Public (UseCase) Generar reportes

Nivel de usuario

Facilidades y controles para permitir el acceso a la información a los usuarios autorizados con la intención de consultar y modificar la información de acuerdo al usuario.

CONECTORES

 **Extend** «extend» Source -> Destination
De: Nivel de usuario : UseCase, Public
a: Consultar información : UseCase, Public


ASOCIACIONES

 Association (direction: Unspecified)
Source: Public (Actor) Usuario Target: Public (UseCase) Nivel de usuario

Selección de tabla

Representa la tabla en la cual se hará una acción ya sea ingresar, consultar, borrar o actualizar.

CONECTORES

 **Include** «include» Source -> Destination
De: Generar reportes : UseCase, Public
a: Selección de tabla : UseCase, Public

ASOCIACIONES

 Association (direction: Unspecified)
Source: Public (UseCase) Consultar información Target: Public (UseCase) Selección de tabla

DCU Gestionar inventario

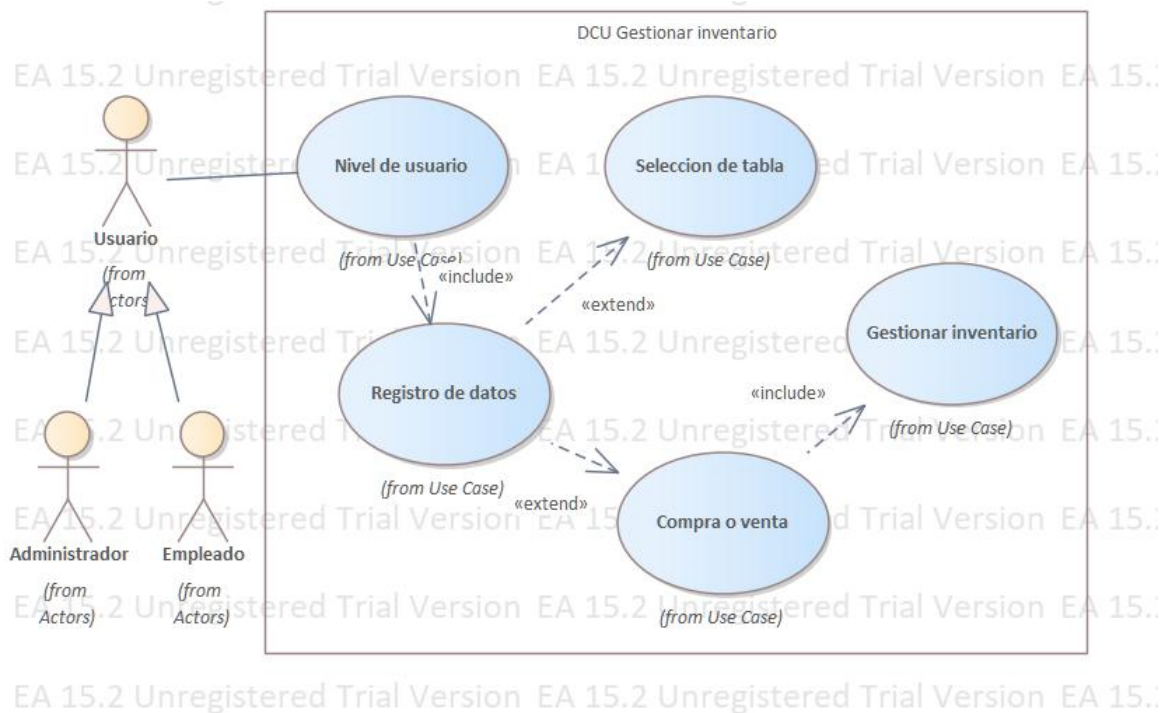


Figura 7. DCU Gestionar inventario

Actores

Administrador

Persona dueña del negocio

RELACIÓN
<p>← Generalización de Administrador a Usuario</p> <p>[Direction is 'Source -> Destination'.]</p>

Empleado

Persona que trabaja para el administrador

RELACIÓN
<p>← Generalización de Empleado a Usuario</p> <p>[Direction is 'Source -> Destination'.]</p>

Usuario

Representa a las personas por igual, estas interactúan con el sistema, hay dos tipos de usuario: El administrador y empleado.



RELACIONES		
⇒ Generalización de Empleado a Usuario		[Direction is 'Source -> Destination'.]
⇒ Generalización de Administrador a Usuario		[Direction is 'Source -> Destination'.]

ASOCIACIONES		
 Association (direction: Unspecified)		
Source: Public (Actor) Usuario		Target: Public (UseCase) Nivel de usuario

Casos de uso

Compra o venta

Representa si el registro de datos es una compra o una venta para que logre el proceso de gestion de inventario

CONECTORES		
 Include «include» Source -> Destination		
De: Compra o venta : UseCase, Public		
a: Gestionar inventario : UseCase, Public		
 Extend «extend» Source -> Destination		
De: Registro de datos : UseCase, Public		
a: Compra o venta : UseCase, Public		


Gestionar inventario

Representa la automatización del inventario, la cual es:

Al momento de realizar una venta el sistema automáticamente restara la cantidad vendida del producto en su stock.

Al momento de recibir un pedido se dan de alta los productos esto provocará que el stock aumente conforme a la cantidad del producto pedido


CONECTORES

 **Include** «include» Source -> Destination
De: Compra o venta : UseCase, Public
a: Gestionar inventario : UseCase, Public

Nivel de usuario

Facilidades y controles para permitir el acceso a la información a los usuarios autorizados con la intención de consultar y modificar la información de acuerdo al usuario.

CONECTORES

 **Include** «include» Source -> Destination
De: Nivel de usuario : UseCase, Public
a: Registro de datos : UseCase, Public

ASOCIACIONES

 Association (direction: Unspecified)

Source: Public (Actor) Usuario

Target: Public (UseCase) Nivel de usuario

Registro de datos

Representan los datos que se registrarán en las tablas de la base de datos del sistema

CONECTORES	
 Extend «extend» Source -> Destination	
De:	Registro de datos : UseCase, Public
a:	Compra o venta : UseCase, Public
 Extend «extend» Source -> Destination	
De:	Registro de datos : UseCase, Public
a:	Selección de tabla : UseCase, Public
 Include «include» Source -> Destination	
De:	Nivel de usuario : UseCase, Public
a:	Registro de datos : UseCase, Public

Selección de tabla

Representa la tabla en la cual se hará una acción ya sea ingresar, consultar, borrar o actualizar.

CONECTORES	
 Extend «extend» Source -> Destination	
De:	Registro de datos : UseCase, Public
a:	Selección de tabla : UseCase, Public

DCU Inicio de sesión

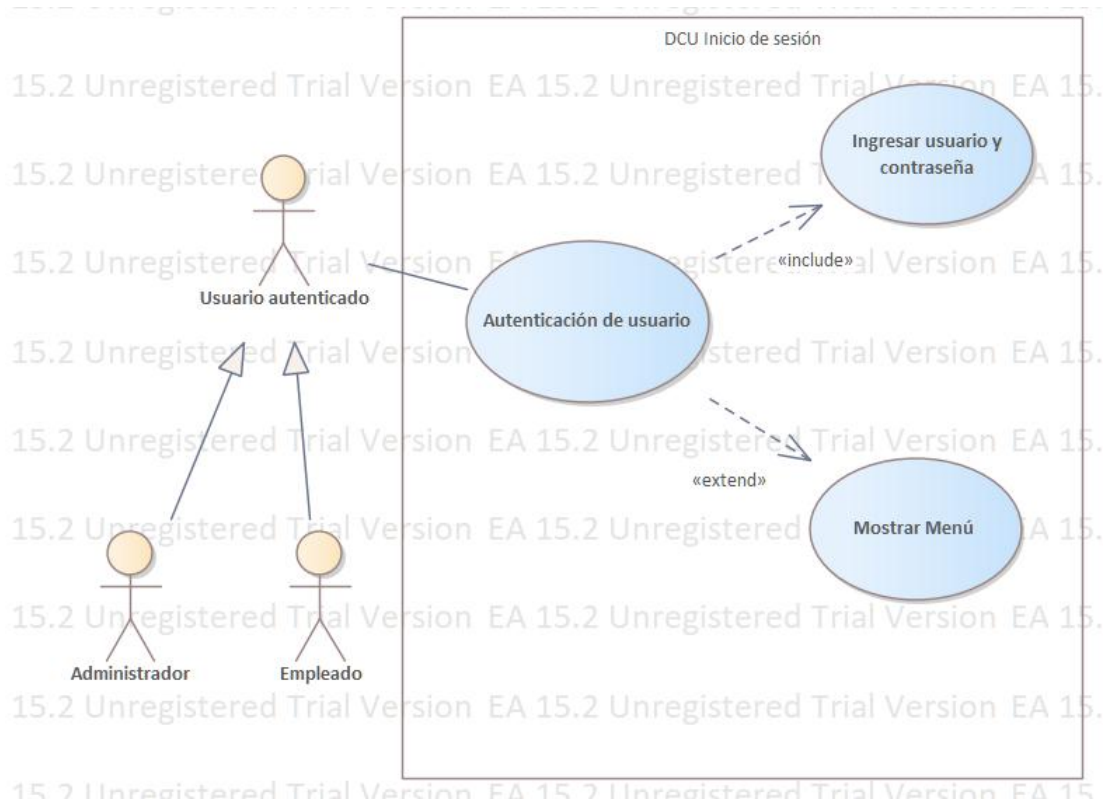


Figura 8. DCU Inicio de sesión

Actores

Administrador

Persona dueña del negocio


RELACIÓN

← Generalización de Administrador a Usuario autenticado

[Direction is 'Source -> Destination'.]



Empleado

Persona que trabaja para el administrador

RELACIÓN	
 Generalización de Empleado a Usuario autenticado	[Direction is 'Source -> Destination'.]

Usuario autenticado

Representa al usuario que registró correctamente sus credenciales al momento del inicio de sesión

RELACIONES	
 Generalización de Empleado a Usuario autenticado	[Direction is 'Source -> Destination'.]
 Generalización de Administrador a Usuario autenticado	[Direction is 'Source -> Destination'.]


ASOCIACIONES	
 Association (direction: Unspecified)	
Source: Public (Actor) Usuario autenticado	Target: Public (UseCase) Autenticación de usuario


Casos de usos

Autenticación de usuario

Representa que las credenciales de los usuarios sean correctas

CONECTORES

 **Include** «include» Source -> Destination
De: Autenticación de usuario : UseCase, Public
a: Ingresar usuario y contraseña : UseCase, Public

 **Extend** «extend» Source -> Destination
De: Autenticación de usuario : UseCase, Public
a: Mostrar Menú : UseCase, Public

ASOCIACIONES

 Association (direction: Unspecified)


Source: Public (Actor) Usuario autenticado

Target: Public (UseCase) Autenticación de usuario

Ingresar usuario y contraseña

El ser un usuario autenticado conlleva el haberse registrado antes, en ese momento especificamos un id de usuario y una contraseña, los cuales representa este caso de uso


CONECTORES

 **Include** «include» Source -> Destination
De: Autenticación de usuario : UseCase, Public
a: Ingresar usuario y contraseña : UseCase, Public

Mostrar Menú

el usuario visualizará una ventana donde se encontrarán los productos que tienen un bajo stock, mostrando el nombre del producto, el precio de compra y el stock.

CONECTORES

 **Extend** «extend» Source -> Destination
De: Autenticación de usuario : UseCase, Public
a: Mostrar Menú : UseCase, Public

DCU Registro

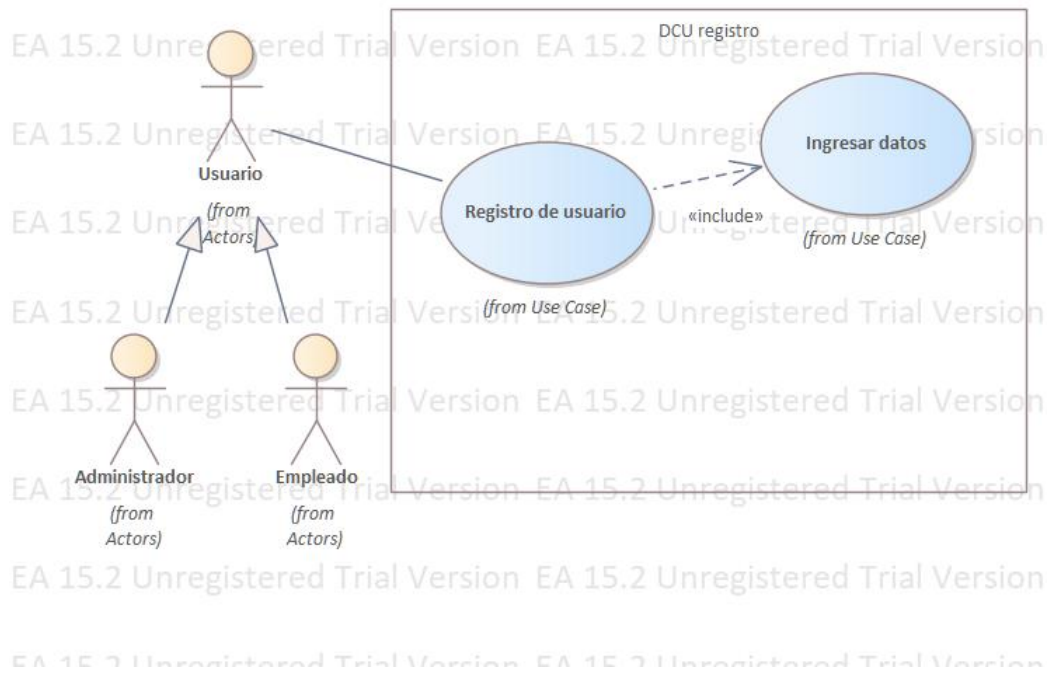


Figura 9. DCU Registro

Actores

Administrador

Persona dueña del negocio

RELACIÓN	
← Generalización de Administrador a Usuario	[Direction is 'Source -> Destination'.]

Empleado

Persona que trabaja para el administrador

RELACIÓN	
← Generalización de Empleado a Usuario	[Direction is 'Source -> Destination'.]

Usuario

Representa a las personas por igual, estas interactúan con el sistema, hay dos tipos de usuario: El administrador y empleado.

RELACIONES	
⇒ Generalización de Empleado a Usuario	[Direction is 'Source -> Destination'.]
⇒ Generalización de Administrador a Usuario	[Direction is 'Source -> Destination'.]

ASOCIACIONES	
 Association (direction: Unspecified)	
Source: Public (Actor) Usuario	Target: Public (UseCase) Registro de usuario

Casos de uso

Ingresar datos

Al momento de registrarse el usuario deberá ingresar datos


- El usuario administrador debe ingresar datos como: RFC, Nombre, Apellido paterno, Apellido materno, E-mail, Contraseña.
- El usuario empleado debe ingresar datos como: RFC, Nombre, Apellido paterno, Apellido materno, Numero telefónico, Calle, No. Calle, Colonia, Ciudad, Copia de INE (PDF), Solicitud de empleo(PDF), Comprobante de domicilio(PDF) y contraseña.

CONECTORES	
 Include «include» Source -> Destination	
De: Registro de usuario : UseCase, Public	
a: Ingresar datos : UseCase, Public	

Registro de usuario

Es el momento en el que el usuario se registra al sistema para poder ser un usuario autenticado.

CONECTORES

 **Include** «include» Source -> Destination
De: Registro de usuario : UseCase, Public
a: Ingresar datos : UseCase, Public

ASOCIACIONES

 Association (direction: Unspecified)

Source: Public (Actor) Usuario

Target: Public (UseCase) Registro de usuario

Estudio de factibilidad

Factibilidad económica

HARDWARE

Teniendo en cuenta que la administradora posee el equipo necesario, los costos asociados a hardware son nulos.

Componente	Especificaciones mínimas	Costos
Memoria RAM	2 GB	\$399
Tarjeta de video	Estándar	\$1183
Procesador del equipo	Intel celeron	\$840
Disco duro	256 gb	\$650
Teclado	Estándar	\$100
Monitor	800x600	\$900
Impresora	Estándar	\$800
Lector código de barras	Estándar	\$462

SOFTWARE

Teniendo en cuenta que los equipos poseen el software necesario y apropiado sin la necesidad de recurrir a gastos extras o adquisición de alguna licencia, podemos afirmar que los costos asociados a software son nulos.

Componente	Requisitos del sistema
Sistema Operativo	Windows, MacOS o Linux.
Navegador Web	Google Chrome, Microsoft Edge, Mozilla Firefox.

Factibilidad técnica

HARDWARE

Componente	Especificaciones mínimas
Memoria RAM	2 GB
Tarjeta de video	Estándar
Procesador del equipo	Intel celeron
Disco duro	256 gb
Teclado	Estándar
Monitor	800x600
Impresora	Estándar
Lector código de barras	Estándar

SOFTWARE

Componente	Requisitos del sistema
Sistema Operativo	Windows, MacOS o Linux.
Navegador Web	Google Chrome, Microsoft Edge, Mozilla Firefox.

Actualmente, la empresa cuenta con la tecnología tanto en el hardware como en el software necesaria, basándonos en la factibilidad técnica podemos decir que es factible por realizar el proyecto.

Factibilidad operativa

Para la microempresa Pily, la necesidad de realizar un cambio en la situación actual es un hecho concreto, la manera en la que se esta gestionando la entrada y salida de sus productos provoca muchas dificultades en los tiempos que genera este proceso. Esto conlleva a que se generara la necesidad de automatizar el proceso de registro y salida de productos.

Teniendo esta información, expresada en reuniones y entrevistas con la dueña de la microempresa, se ha planteado una solución que consistirá en la creación de un sistema automatizado, el cual **SIABA** pretende ser un apoyo en gran parte de la administración que se produce actualmente.

Con el fin de desarrollar un software de calidad, se garantiza un buen funcionamiento del sistema y una excelente percepción por parte del usuario, donde estará presente una interfaz con un diseño minimalista y de uso intuitivo, que facilite la comprensión del sistema y que sea fácil de manejar, provocando que el usuario pueda familiarizarse con el sistema en muy poco tiempo.

El sistema será encargado de automatizar los procesos de entrada y salida del inventario, lo cual va a generar los siguientes aspectos:

- Se reduce la carga de trabajo, lo que permite desempeñar de mejor forma sus labores.
- Aumenta la confianza de los clientes con la empresa al trabajar con nuevas tecnologías de información (TI).
- Se disminuye el tiempo en el proceso de gestionar los productos.
- Permite revisar la base de datos para identificar que productos se encuentran disponibles.

Teniendo en cuenta los impactos positivos que genera el sistema, la microempresa Pily está de acuerdo con la realización de este, pues se encuentra beneficiado tras el desarrollo del proyecto.

Basándonos en la factibilidad operacional podemos decir que es factible realizar el proyecto.

Costo-beneficio del sistema

Nuestro cliente ya cuenta con casi todo a nivel técnico, lo único con lo que no cuenta es con el lector de código de barras, por lo tanto, su inversión se recuperaría en menos de un año, al ser el sistema enfocado en software libre, no usamos licencias que aumenten el precio, con esto el precio final de nuestro sistema incluyendo el lector de código de barras sería de 10,000 pesos.

A tener en cuenta que nuestro cliente gasta aproximadamente 850 por mes en hojas, tinta, plumas, carpetas y corrector. Por lo tanto, el proyecto es factible.

Anexos

Anexo A.

1. ¿Para qué requiere del uso de un software en su microempresa?
2. ¿Quién es el responsable de realizar las ventas de los productos?
3. ¿Existe un formato designado al momento de realizar una venta?
4. ¿Cuál es el procedimiento al momento de realizar una venta?
5. ¿Quién es el que realiza los pedidos a los proveedores?
6. ¿Quién es el responsable de registrar las entradas del inventario?
7. ¿Puede describir el proceso de registrar los productos?
8. ¿Quién realiza el conteo físico de inventario?
9. ¿Cuenta con algún formato donde se lleve el seguimiento de su inventario?
10. ¿Puede describir la manera en la que realiza el seguimiento de su inventario?
11. ¿Qué tiempo le tomaría aproximadamente contar su inventario actual?
12. ¿Qué tan frecuente realiza el proceso de conteo físico de inventario?
13. ¿Cuenta con algún tipo de identificación de sus productos? ¿Quiere algún tipo de identificación en específico?
14. ¿Qué información requiere de sus productos?
15. ¿Maneja productos a granel?
16. ¿Cuenta con algún tipo de identificación para sus empleado (código o número)? ¿le gustaría alguno?
17. ¿Qué información cuenta de sus empleados (dirección, número de teléfono, etc.)?

18. ¿Requiere llevar un control sobre el salario?
19. ¿Dispone con un equipo de cómputo en la microempresa?
20. ¿Que tipo de equipo? R/ Es un equipo de escritorio marca Lenovo.
21. ¿Sus empleados tienen experiencia en el uso de los equipos de cómputo?
22. ¿Cuántos empleados tiene a su cargo?
23. ¿Existe algún factor que le dificulte el trabajo a su empleado?
24. ¿Cuenta con un logo en su microempresa?, ¿Requiere de uno?
25. ¿Requiere de algunos colores en específico en la interfaz de su sistema?
26. ¿De las siguientes imágenes cual es mas de su agrado?