**Logotipo, nombre de la empresa

Descripción generada automáticamente con confianza media**

**PROFESORES:**

**Leonel Jimenez Gamboa**

**ALUMNO:**

**Agustin Fabian Ricci**

Contenido

[**(G01)** Proposito: 7](#_Toc178705723)

[**(G02)** Descripción funcional del producto y Alcance: 8](#_Toc178705724)

[PN1 Gestion de ventas de la tienda: 8](#_Toc178705725)

[**RFN2.** Gestion de Compras de articulos para todos los puntos de venta 9](#_Toc178705726)

[PN2 Gestion de Compras de artículos para todos los puntos de venta: 9](#_Toc178705727)

[(G03) Definiciones,Acronimos,Abreviaturas 10](#_Toc178705728)

[Definiciones: 10](#_Toc178705729)

[Acronimo: 11](#_Toc178705730)

[Abreviaturas: 12](#_Toc178705731)

[(G04) TABLA DE ACTORES. 12](#_Toc178705732)

[(G05) Otros Requisitos. 14](#_Toc178705733)

[**(G06.1)** 16](#_Toc178705734)

[(G06.1.2) Capa de Entidad Sobre Negocios 17](#_Toc178705735)

[(G06.1.3) Capa de Negocios Sobre El Negocio 18](#_Toc178705736)

[(G06.2) Capa de Servicios General 19](#_Toc178705737)

[(G06.2.1) Capa de Entidad de Servicios 20](#_Toc178705738)

[(G06.2.2) Capa de Negocios de Servicios 21](#_Toc178705739)

[(G06.3) Capa de Datos 21](#_Toc178705740)

[(G07). DER COMPLETO 23](#_Toc178705741)

[(Punto 3.1 / N01-A ) Actores de PN1. 24](#_Toc178705742)

[DIAGRAMA DE ROLES N0.1-B 24](#_Toc178705743)

[(Punto 3.2 / N02-A ) 25](#_Toc178705744)

[Diagrama de Roles / N02-B 25](#_Toc178705745)

[DIAGRAMA DE SECUENCIA DE ROLES/N01-C 26](#_Toc178705746)

[DIAGRAMA DE SECUENCIA DE ROLES/N02 27](#_Toc178705747)

[DIAGRAMA ECS.(Punto 3.1/N01-B) 27](#_Toc178705748)

[DIAGRAMA ECS.(Punto 3.1/N02-B) 27](#_Toc178705749)

[DIAGRAMA DE PROCESOS.(Punto 4/N01-C) 28](#_Toc178705750)

[DIAGRAMA DE PROCESOS.(Punto 4/N0.2-C) 29](#_Toc178705751)

[(Punto 3.4 / N01-D Carpeta) 30](#_Toc178705752)

[(Punto 3.4 /N02-D Carpeta) 30](#_Toc178705753)

[N02.1 Diseño del negocio(RFN1-Gestion de ventas de la tienda) 31](#_Toc178705754)

[N02.1.1-Diagrama de casos de uso (general) 31](#_Toc178705755)

[N02.1.2 - Especificación de casos de uso 32](#_Toc178705756)

[N02.1.2.1 - CU01 – Registrar Cliente 32](#_Toc178705757)

[N02.1.3.2 - DS “CU01- Registrar Cliente” 33](#_Toc178705758)

[N02.1.4 - Diagrama de Clases - “CU01 - Registrar Cliente” 34](#_Toc178705759)

[N02.1.4.2 - DC “CU01- Registrar Cliente” 34](#_Toc178705760)

[N02.1.6 – Diagrama de Actividad “CU01 – Registrar Cliente” 35](#_Toc178705761)

[N02.1.6.2 – DER “CU01 – Registrar Cliente 36](#_Toc178705762)

[N02.1.2.1 – “CU02 – Seleccionar Producto” 37](#_Toc178705763)

[N02.1.4 - Diagrama de Clases - “CU03 -Llenar carrito” 44](#_Toc178705764)

[N02.1.4 - Diagrama de Clases - “CU04 – Generar Factura” 49](#_Toc178705765)

[N02.1.4 - Diagrama de Clases - “CU05 -Cobrar Venta” 53](#_Toc178705766)

[N02.1.4 - Diagrama de Clases - “CU06 -Despachar Producto” 58](#_Toc178705767)

[N02.2 - Diseño del negocio(RFN2- Gestion de compras de Producto) 60](#_Toc178705768)

[N02.2.1 Diagrama de casos de uso General 60](#_Toc178705769)

[N02.2 1.2 Especificación del CU01 Generar orden de Compra 61](#_Toc178705770)

[N02.2 1.3 Diagrama de secuencia del CU01 Generar orden de Compra 62](#_Toc178705771)

[N02.2 1.4 Diagrama de Clases del CU01 Generar orden de Compra 63](#_Toc178705772)

[N02.2 1.6 DER CU01Generar orden de Compra 63](#_Toc178705773)

[N02.2 2.2 Especificación del CU02 Generar Solicitud de cotizacion 64](#_Toc178705774)

[N02.2 2.3 Diagrama de secuencia del CU02 Generar solicitud de cotizacion. 65](#_Toc178705775)

[N02.2 2.4 Diagrama de Clases del CU02 Generar solicitud de cotizacion 66](#_Toc178705776)

[N02.2 2.6 DER CU02 Generar Solicitud Cotización 67](#_Toc178705777)

[N02.2 1.7 GUI CU02 Generar Solicitud Cotización 67](#_Toc178705778)

[N02.2 3.2 Especificación del CU03 Pagar 68](#_Toc178705779)

[N02.2 3.3 Diagrama de secuencia del CU03 Pagar 69](#_Toc178705780)

[N02.2 3.4 Diagrama de clases del CU03 Pagar 70](#_Toc178705781)

[N02.2 3.6 DER CU03 Pagar 70](#_Toc178705782)

[N02.2 4.2 Especificación del CU04 Gestion de Productos 71](#_Toc178705783)

[N02.2 4.3 Diagrama de secuencia del CU10 Gestión de Productos 72](#_Toc178705784)

[N02.2 4.4 Diagrama de clases del CU04 Gestión de Productos 73](#_Toc178705785)

[N02.2 4.6 DER CU10 Gestión de Productos 73](#_Toc178705786)

[N02.2 4.7 GUI CU04 Gestión de Productos 73](#_Toc178705787)

[N02.2 5.2 Especificación del CU05 Registro del proveedor 74](#_Toc178705788)

[N02.2 5.3 Diagrama de secuencia del CU08 Registro del proveedor 75](#_Toc178705789)

[N02.2 5.4 Diagrama de clases CU05 Registro de proveedor 75](#_Toc178705790)

[N02.2 6.2 Especificación del CU06 Registro del proveedor 76](#_Toc178705791)

[N02.2 6.3 Diagrama de secuencia del CU06 Pre-registro de Proveedor 77](#_Toc178705792)

[N02.2 6.4 Diagrama de Clases del CU06 Pre-registro de Proveedor 78](#_Toc178705793)

[N02.2 6.5 DER CU06 Pre-registro de proveedor 78](#_Toc178705794)

[N02.2 6.7 GUI CU06 Pre-registro de proveedor 78](#_Toc178705795)

[T01- Arquitectura Base 79](#_Toc178705796)

[T01.1 – Diseño de la Arquitectura. (Diagrama de capas) 79](#_Toc178705797)

[T01.2 Captura de pantalla de las capas creadas en el IDE 80](#_Toc178705798)

[T01.3 – Diseño modelo del DS de Persistencia de Datos 81](#_Toc178705799)

[T01.4 – Diseño modelo del DS de Consulta de Datos 82](#_Toc178705800)

[T01.5 – Diagrama Jerarquico de la estructuracion del menu principal 83](#_Toc178705801)

[T01.6 – Captura de pantalla del Menu Principal 83](#_Toc178705802)

[T02. Gestion de Usuario 84](#_Toc178705803)

[T02.1 – Crear Usuario 84](#_Toc178705804)

[T02.2 - Login 90](#_Toc178705805)

[T02.3- Desbloquear Usuario 97](#_Toc178705806)

[T02.4 - Cambiar Clave 102](#_Toc178705807)

[T02.5 - Logout 108](#_Toc178705808)

[T03. Encriptación 113](#_Toc178705809)

[T03.1 - Irreversible 113](#_Toc178705810)

[T03.2 - Reversible (Explicación Campo Sensible BD - Caja Negra) 116](#_Toc178705811)

[T04 Crear Perfil 118](#_Toc178705812)

[T04.1 Descomposicion funcional 118](#_Toc178705813)

[T04.2 – Crear Familias 124](#_Toc178705814)

[T05 Idioma 129](#_Toc178705815)

[(T06) Bitacora de Eventos 136](#_Toc178705816)

[Diagrama de Secuencia de Gestion de Bitacora EVENTOS 136](#_Toc178705817)

[Diagrama de Clases de Gestion de Bitacora EVENTOS 137](#_Toc178705818)

[Diagrama de Entidad Relacion de Bitacora EVENTOS 138](#_Toc178705819)

[Auditoria bitacora de eventos. 138](#_Toc178705820)

[Descripcion funcional 138](#_Toc178705821)

[Diagrama de Casos de uso 140](#_Toc178705822)

[(T06.1.3) Diagrama de Secuencia CU01-Auditar Bitacora 141](#_Toc178705823)

[(T06.1.4) Diagrama de Clases CU01-Auditar Bitacora 142](#_Toc178705824)

[(T06.1.4) Diagrama Entidad Relacion CU01-Auditar Bitacora 142](#_Toc178705825)

[Bitacora de Cambios. 143](#_Toc178705826)

[Descripción Funcional: 143](#_Toc178705827)

[Diagrama Actividad Bitacora de Cambios 144](#_Toc178705828)

[Auditoria bitacora de Cambios. 145](#_Toc178705829)

[Descripcion funcional 145](#_Toc178705830)

[Diagrama de Casos de uso 146](#_Toc178705831)

[Especificacion de Casos de uso 147](#_Toc178705832)

[Diagrama de secuencia de CU 01 Auditar Bitacora Historico 148](#_Toc178705833)

[Diagrama de clases de CU-01 Auditar Bitacora Historico 149](#_Toc178705834)

[Diagrama Entidad Relacion Bitacora de Historicos 149](#_Toc178705835)

[T07 -Gestion de respaldo. 150](#_Toc178705836)

[Descripcion funcional 150](#_Toc178705837)

[Casos de uso. 150](#_Toc178705838)

[Especificacion de CU. 151](#_Toc178705839)

[Diagrama de secuencia. 152](#_Toc178705840)

[Diagrama de clases. 153](#_Toc178705841)

[Diagrama de Entidad Relacion (CU01-Gestionar Respaldo) 153](#_Toc178705842)

[GUI. 154](#_Toc178705843)

[A.03 Serializacion 155](#_Toc178705844)

[(A03.1)Descripcion Funcional: 155](#_Toc178705845)

[(A03.2)Diagrama de casos de uso: 155](#_Toc178705846)

[(A03-1) Especificacion de caso de uso CU-01 Serializar-Desserializar 156](#_Toc178705847)

[(A03-2) Diagrama de secuencia CU-01 Serializar-Desserializar 157](#_Toc178705848)

[(A03-3) Diagrama de clases CU01 Gestion Serializacion 158](#_Toc178705849)

[(A03.4) Diagrama Entidad Relacion CU01 Gestion Serializacion 158](#_Toc178705850)

[(A03.5) GUI CU01 Gestion Serializacion 159](#_Toc178705851)

## **(G01)** Proposito:

Productos OSC se embarcará en un proyecto integral para gestionar las ventas y despachos de sus productos de limpieza vehicular. Este sistema estará diseñado para adaptarse al mercado local, con múltiples puntos de venta y despacho. Los clientes podrán efectuar compras en cualquier punto y recibir sus pedidos donde haya disponibilidad de stock. Cada punto de venta contará con personal capacitado para atender a los clientes, registrar información esencial, administrar inventario, gestionar pedidos y despachos. Además, se integrarán funciones de facturación y gestión de clientes para brindar una experiencia de compra satisfactoria y completa.

## **(G02)** Descripción funcional del producto y Alcance:

RFN1: Gestion de ventas de la tienda: En cada sucursal de la empresa la compra se puede realizar la compra de los productos, la gestion va a requerir los datos del cliente, los detalles de los productos que el cliente quiere, la verificacion de dichos productos, la realizacion de la factura de la venta, la cobranza de la misma y el despacho de la venta realizada.

#### PN1 Gestion de ventas de la tienda:

1. El cliente le comunica al vendedor que productos desea adquirir,le comunica la informacion relevante sobre el producto como (modelo,marca,tipo,cantidad,etc), el vendedor verifica que los productos solicitados estan en stock, en caso de que no esten se le sugiere otros productos similares.
2. El vendedor verifica que los productos solicitados como (stock, modelo , marca,codigo de producto, tipo) estén disponibles en la tienda. En caso de que no estén disponibles, le ofrece alternativas similares disponibles en la tienda.
3. El vendedor una vez que verifica los productos, debe completar en un carrito de compras los productos solicitados por el cliente con la informacion detallada como (Modelo,marca,tipo,cantidad,etc) y debe corroborar que el cliente este de acuerdo con los productos marcados.
4. Terminada la verificacion del carrito, el vendedor debe cerrar el pedido del carrito, el vendedor procede a solicitar al cliente datos relevantes sobre el como(Nombre,Edad,Apellido,Mail,Dni).
5. Con los datos del cliente, el vendedor procede a realizar la factura de la venta y ofrece los distintos medios de pago para abonar la compra, de forma que se puede realizar el pago con efectivo, tarjeta de debito o tarjeta de credito y se le solicita datos como(Numero de tarjeta,codigo de seguridad, vencimiento,nombre que figura en la tarjeta,Ciudad,Provincia,CP), sabiendo que se puede pagar hasta con dos medios de pago, pero con tarjeta de credito tendra un

recargo de 10%.

1. Se procede a pedir al cliente en caso de que abone con tarjeta, los datos como (Nombre completo, monto a cancelar,banco asociado).
2. Una vez cobrado y cancelado el total de la venta, se imprimen dos facturas de compra y se piden los datos para dicha factura como (nombre del cliente, apellido, número de teléfono, DNI, nombre del vendedor, CUIL, código postal, dirección, teléfono, tipo de factura, subtotal, impuestos, forma de pago). Una copia se queda en la tienda y otra se entrega al cliente. Una vez entregada la compra al cliente, el vendedor debe marcar en el sistema que dicha compra fue entregada, generándose automáticamente un recibo de la compra en caso de que se desee cambiar algún producto por falla del mismo.

#### **RFN2.** Gestion de Compras de articulos para todos los puntos de venta

El sistema debe permitir a los usuarios gestionar la compra de artículos de limpieza a todos los proveedores. Esto incluye la capacidad de enviar una solicitud de cotizacion, para posteriormente aceptarla o no, en caso de aceptar se genera una órden de compra, seleccionar los proveedores adecuados, gestionar las cantidades de productos a comprar, y actualizar automáticamente el inventario una vez que se confirme la recepción de los productos. además de proporcionar un historial de compras por proveedor y por artículo, tambien incluye la opcion de que en caso de que el stock de algun producto del inventario baje hasta menos de un punto determinado, emite una alerta de reposicion de stock para el encargado de compras.

#### PN2 Gestion de Compras de artículos para todos los puntos de venta:

1. **Solicitar Cotizacion:**

El suministrador Inicia el proceso de compra, accede al formulario de productos revisando una grilla que muestra todos los productos con stock por debajo de 25. Alli genera una cotizacion a todos los proveedores con (Cantidad por producto, Modelo, Marca, Tamaño, Tipo,Fecha de Envio,Condiciones de Pago,Fecha esperada de respuesta) de la empresa(Email,Razon social, Cuil, Contacto,Direccion), del proveedor(Email,Cuil, Razon Social, Direccion,Telefono).

En caso de que no tenga el proveedor **registrado**, se registra al proveedor en el formulario de proveedores, con los siguientes datos (Email,Cuil, Razon Social, Direccion,Telefono,DNI).

1. **Evaluacion y selección de Proveedores:**

El suministrador recibe las cotizaciones de los proveedores y evalua los productos de las cotizaciones mirando(Precio,Codigo producto, cantidad, modelo), seleccionando los proveedores que se desea generar una orden de compra seleccionando(Nombre, CUIL,Direccion,Dni) en caso de que no desee un proveedor simplemente no se selecciona.

1. **Generar Orden de Compra:** El Suministrador revisa de cada solicitud los productos(Marca, Modelo,Tamaño,Tipo, Precio Unitario), compara los precios de cada proveedor, a la hora de generar la orden de compra, se envia una aprobacion con los productos que se elijeron previamente, indicando(Producto, Modelo, Tamaño,Tipo, Cantidad,Fecha Confirmacion,Referencia Cotizacion,Nombre de la Empresa, telefono, Razon Social,Email, Direccion de Envio) del receptor (Nombre,cuil, telefono).

**Aclaracion:** El paso 3 puede pasar con cualquier cantidad de productos con cada proveedor.

1. **Confirmacion orden de Compra:** Cuando el proveedor recibe la orden de compra, responde con una confirmacion que incluye detalles especificos para poder asegurar que ambas partes estan alineadas, como(Fecha confirmacion,Referencia orden compra, Estado de Orden, Fecha entrega, Tiempo Preparacion, Metodo de envio,Direcion Entrega Confirmada).
2. **Realizar pago de Compra:**

El administrador realiza el pago de dicha orden hacia el proveedor, una ves realizado el pago y aprobado, se envia (Fecha del Pago, Monto Pagado, Metodo de Pago, Numero de Transaccion, Numero de factura asociada, confirmacion de terminos de pago, Adjunto de Comprobante de pago).

1. **Confirmacion de Pago:** Una ves que el proveedor recibe el pago enviado anteriormente, adjunta documentacion para confirmar que el pago ha sido recibido y validado (Estado de Orden, Terminos de garantia, Politica de devolucion y reemplazo, Monto del Pago, Fecha Transaccion) donde esta informacion es recibida por el suministrador.
2. **Recepcion de Productos:** Una ves que la orden de compra ha sido pagada y el pago ha sido administrado por el proveedor, se procede a recibir los productos solicitados con una factura (Numero de factura, fecha emision, fecha recepcion de productos) del proveedor (Nombre, direccion,telefono, Cuit), del comprador (Cuit, direccion, dni) y de los productos ( Codigo de producto, descripcion, cantidad, Precio unitario,subtotal, total impuestos, descuento aplicado por cantidad comprada).

## (G03) Definiciones,Acronimos,Abreviaturas

#### Definiciones:

* **Producto:** Un bien tangible ofrecido para la venta que puede satisfacer una necesidad o deseo del cliente.
* **Limpieza de autos:** Proceso de limpiar y mantener la apariencia de un vehículo, tanto por dentro como por fuera.
* **Detalle de autos (Auto Detailing):** Proceso de realizar una limpieza, restauración y acabado de alta calidad del vehículo, tanto en el interior como en el exterior, para producir un nivel de detalle y limpieza que normalmente no se alcanza en la limpieza regular del vehículo.
* **Sistema de ventas:** Conjunto de procesos y herramientas utilizadas para gestionar las ventas de productos y servicios.
* **Punto de venta (POS):** Lugar donde se realizan las transacciones de ventas, ya sea en una tienda física o en línea.
* **Carrito:** Conjunto de articulos que se suman para realizar la venta del cliente.
* **Proveedor:** Es la entidad que se encarga de vender productos automoviles a la empresa ProductosOSC.
* **IVA (Impuesto al Valor Agregado):** Es un impuesto que se aplica en casi todas las
* etapas de la cadena de producción y comercialización de bienes y servicios. El IVA

es un porcentaje que se añade al precio final de un producto o servicio y que el

consumidor final debe pagar.

* **AFIP (Administración Federal de Ingresos Públicos):** Es el ente gubernamental

encargado de la recaudación de impuestos, el control aduanero y la seguridad social

en Argentina. AFIP regula y fiscaliza las actividades comerciales y de servicios para

asegurar el cumplimiento de las obligaciones fiscales.

* **Ganancias (Impuesto a las Ganancias):** Es un impuesto que se aplica a los ingresos

obtenidos por personas físicas y jurídicas. La alícuota y las condiciones varían

dependiendo del tipo de contribuyente y de los ingresos obtenidos.

* **Ingresos Brutos:** Es un impuesto provincial que grava el total de los ingresos

obtenidos por las actividades comerciales, industriales y de servicios.

#### Acronimo:

* **OSC:** Nombre de la empresa (no necesariamente un acrónimo, pero utilizado como tal para referirse a Productos OSC).
* **POS (Point of Sale):** Punto de venta.
* **PV** - **Punto de Venta**: Ubicaciones físicas o digitales donde se venden productos.
* **OC** - **Orden de Compra**: Documento que representa la compra de productos a un proveedor.
* **FP** - **Forma de Pago**: Método utilizado por el cliente para realizar la transacción (tarjeta, efectivo, etc.).
* **PC** - **Proceso de Compra**: Etapas involucradas en la adquisición de productos para el inventario.
* **CD** - **Control de Distribución**: Supervisión de la entrega y redistribución de productos entre puntos de venta.
* **GV** - **Gestión de Ventas**: Actividades relacionadas con la venta de productos a los clientes.
* **INV** - **Inventario**: Registro de productos disponibles en los puntos de venta.
* **PM** - **Proveedor de Mercancías**: Empresas o personas que suministran productos de limpieza a Productos OSC.
* **DDP** - **Distribución de Productos**: Proceso de entrega de productos entre los puntos de venta o a los clientes finales.
* **PR** - **Proveedor Regular**: Un proveedor habitual con el que se establecen acuerdos a largo plazo.
* **CT** - **Costo Total**: El costo total asociado a la compra y distribución de productos en los puntos de venta.
* **MC** - **Margen Comercial**: Diferencia entre el costo de los productos y su precio de venta.
* **FC** - **Facturación Cliente**: Proceso de emisión de facturas a los clientes después de realizar una venta.
* **DP** - **Demanda de Producto**: La cantidad de productos solicitados por los clientes en un período específico.

#### Abreviaturas:

* **ID:** Identificación de productos.
* **Cant:** Cantidad de productos
* **Prod:** Producto.
* **Cat:** Categoría de productos.
* **Inv:** Inventario de productos
* **Desc:** Descuento de productos.
* **Tr:** Transacción bancaria realizara en la venta de los productos.
* **Ac:** Cuenta bancaria.
* **GB :** GigaByte.
* **Prov:** Proveedor de Productos.
* **Fac :** Factura de productos solicitados.
* **Cot:** La cotizacion, es decir el proceso de pedir los productos faltantes a los proveedores.
* **TP:** termino de pago en una compra realizada.
* **Ret:** Retorno de mercancia en caso de que sea una devolucion.

## (G04) TABLA DE ACTORES.

|  |  |  |
| --- | --- | --- |
| PERSONAL | ROL/Descripcion | Responsabilidad/Acceso |
| Martin Garcia | Vendedor/ Se encarga de atender a los clientes y de llenar el carrito. | RFN1-Seleccionar Productos  RFN1-Llenar Carrito  RFN1 - Despachar Productos según la factura. |
| Carlos Perez | Gerente encargado de las areas de ventas, compras, productos, reportes, bitacoras. | RFN1-RFN2 |
| Javier Gonzalez | Cajero Recibe el dinero de las facturas y comprueba que el pago se realizo con éxito | RFN1-Facturar |
| Sandro Diaz | Analista Funcional empleado encargado de analizar y realizar el correcto funcionamiento de los requisitos funcionale 1 y 2 | RFN1-RFN2 |
| Maria Sanchez | Analista Programador empleado que se encarga de programar los requisitos 1 y 2 | RFN1-RFN2 |
| Pepe Mujica | Product Owner Es el dueño de el proyecto y encargado de aprobar todo lo que se vaya haciendo del mismo | RFN1-RFN2 |
| Agustin Ricci | Scrum Master encargado del equipo que lleva adelante la creacion del proyecto | RFN1-RFN2 |
| San Martin | Suministrador El encargado de gestionar la accion de venta hacia el proveedor y revisar que productos faltan en stock | RFN2 |
| Hurley gomez | Despachador: El encargado de comprobar que los productos de la factura coinciden con los entregados | RFN2 |
| Franco Colapinto | Administrador El administrador se encarga de toda la area de entregas y ventas | RFN2 |

## (G05) Otros Requisitos.

Se documentarán los requisitos para cada uno de los siguientes puntos.

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

**Del producto**

Estándares

* ISO-9000/9001: Gestión y aseguramiento de calidad. Modelo de

aseguramiento de la calidad en diseño, desarrollo, producción,

instalación y servicio.

* El sistema debe tener una interfaz de usuario amigable.
* El sistema debe tener el logo de la empresa en algún lugar
* El sistema debe ser responsivo.

Requisitos de sistema

* Sistema Operativo Windows 10/11
* NET Framework versión 4.7.2
* Procesador con al menos 2Ghz
* 4GB de memoria RAM
* 1GB de disco para la instalación del software
* TB para almacenar los registros en la base de datos y bitácora
* SQL Server 2019 o superior

Requisitos de Desempeño

* Para que el sistema funcione correctamente, se deben respetar los

requisitos previamente indicados en “Requisitos del sistema”

Requisitos de Entorno Requisitos de Documentación:

* Se recomienda instalar y ejecutar el sistema como administrador y en un servidor de una red LAN para tener una conexión mas eficiente.

**De documentación**

Guías de Instalación, Configuración, y Fichero Léame

Para poder comenzar la instalación, es necesario descargar el instalador

y descomprimirlo. A continuación, deberá ejecutar el archivo ‘setup.exe’

con permisos de administrador.

Posteriormente deberá indicar dónde desea que se instale el sistema. Al

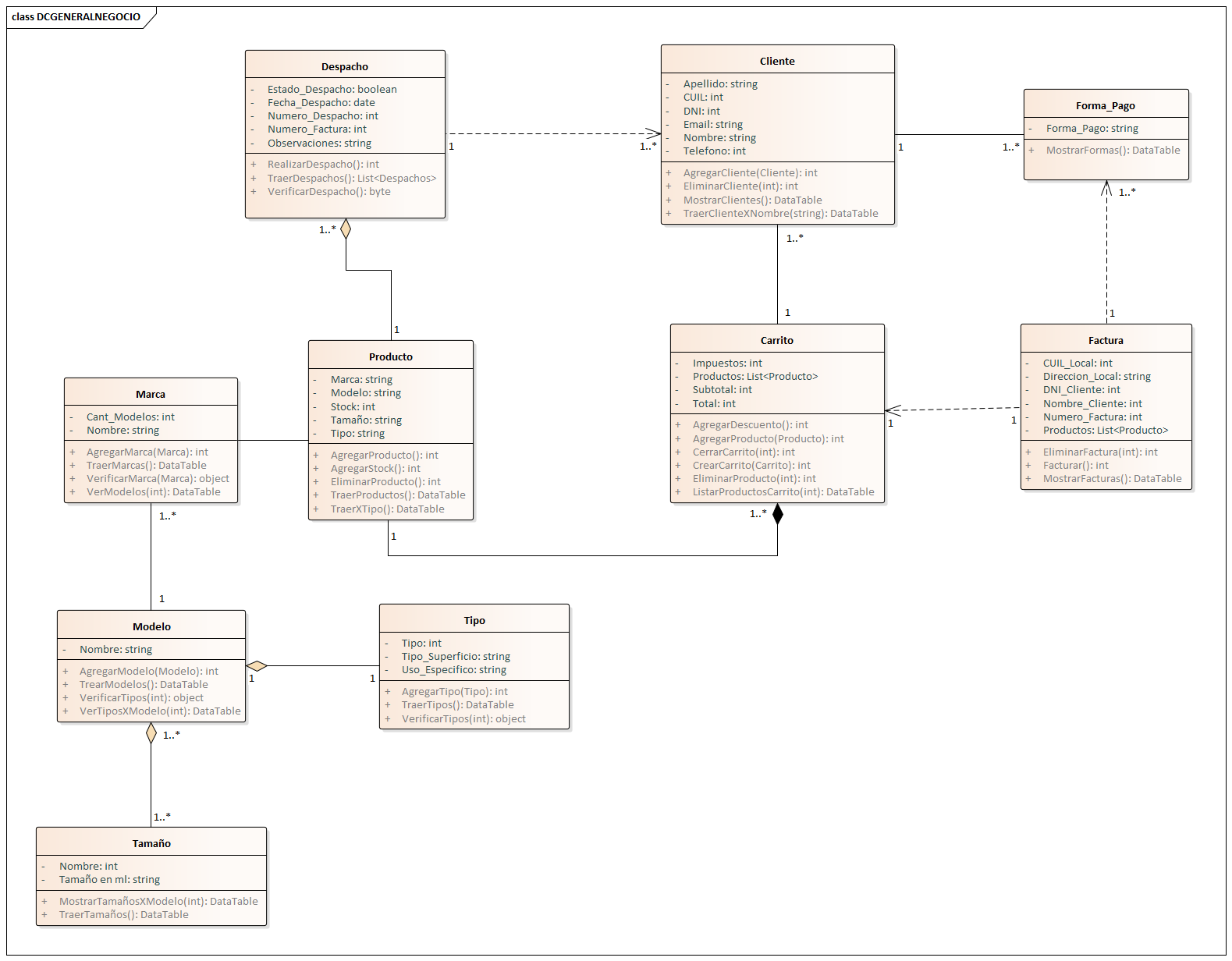
tocar en continuar empezará la instalación. Una vez terminada la misma

se le consultará si desea eliminar el instalador y si reiniciará la

computadora en el momento o más tarde para poder finalizar con la

instalación llevada a cabo.

**(G06.1) Diagrama de clases general del negocio**

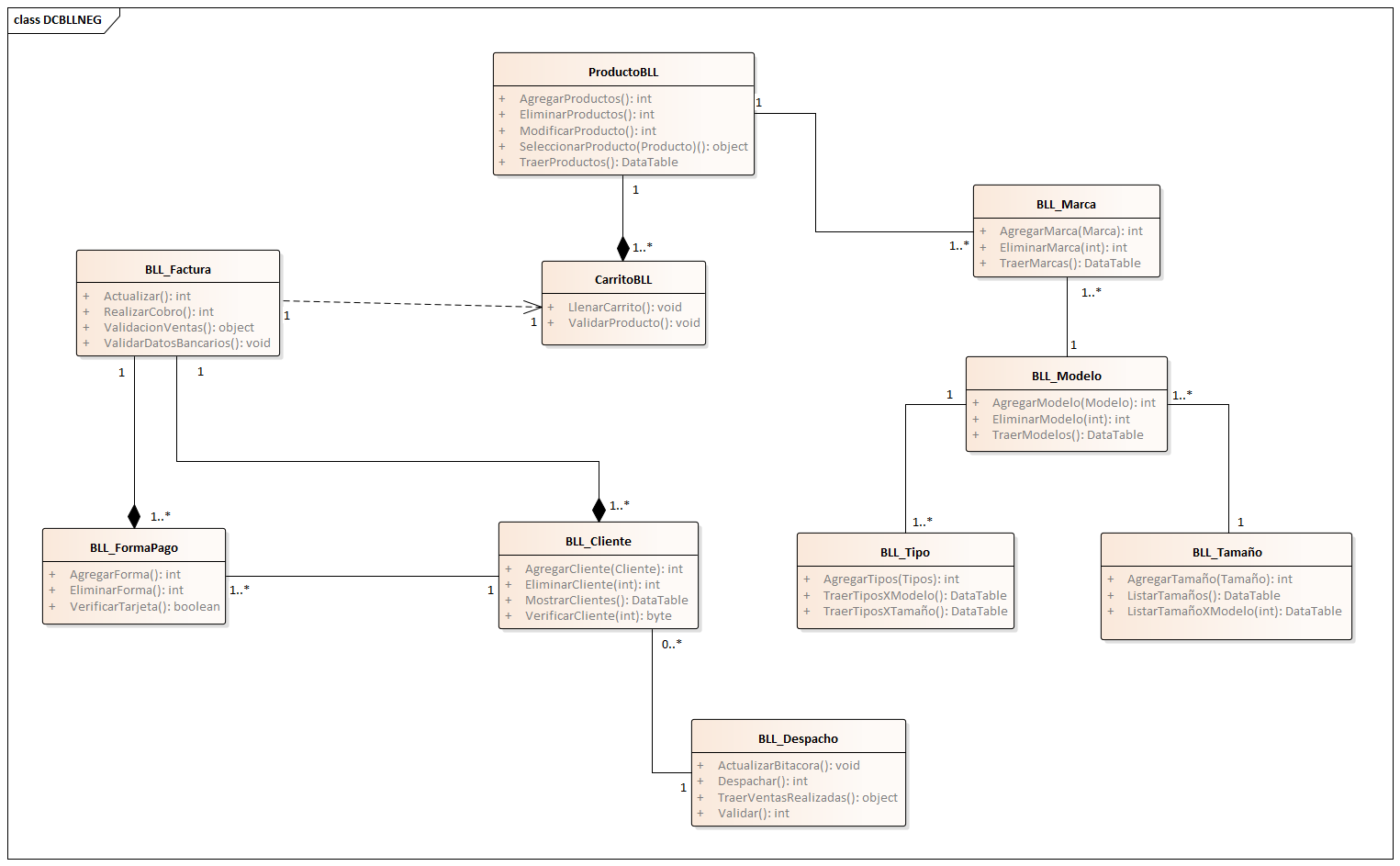
****

#### (G06.1.2) Capa de Entidad Sobre Negocios

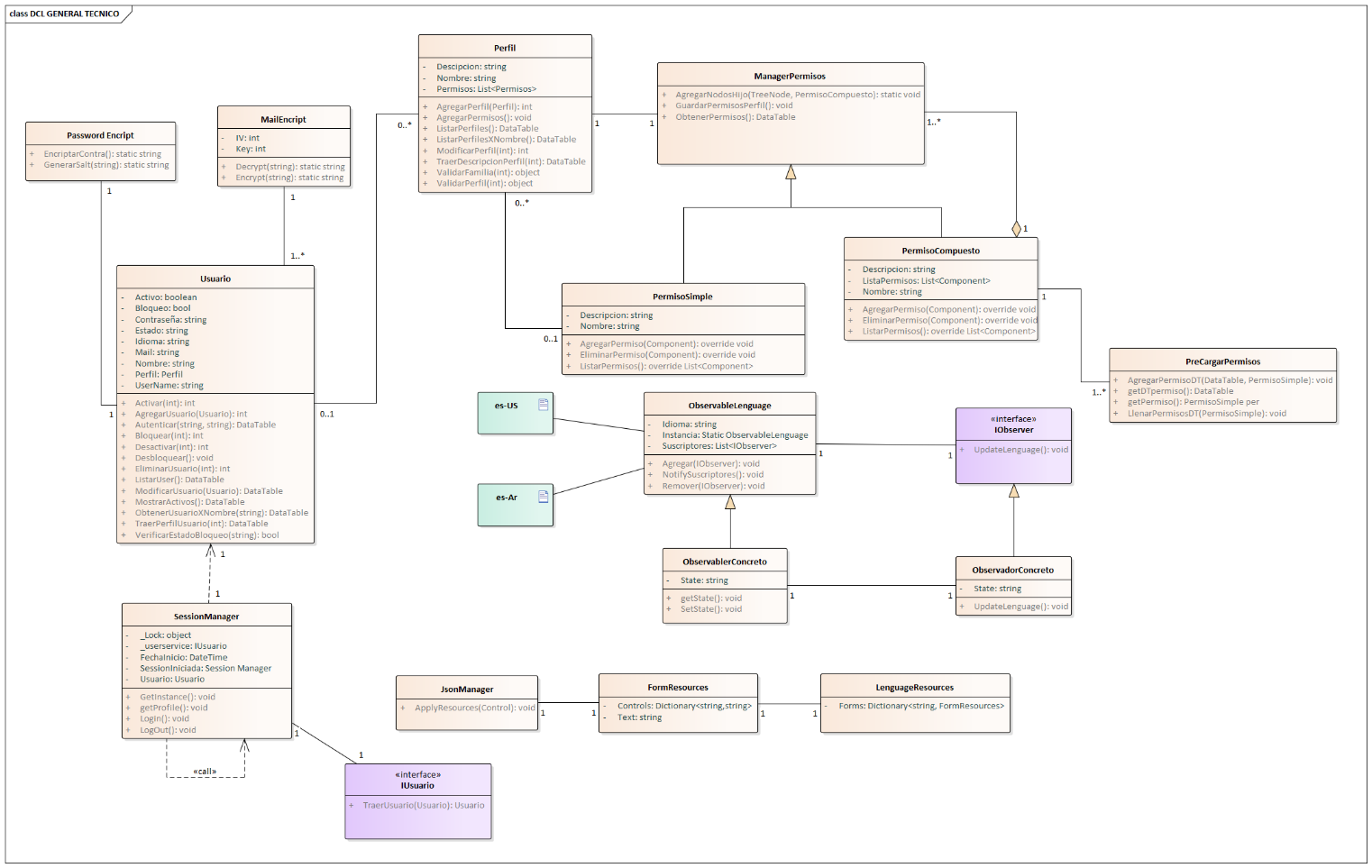
**Diagrama

Descripción generada automáticamente**

#### (G06.1.3) Capa de Negocios Sobre El Negocio

****

#### (G06.2) Capa de Servicios General

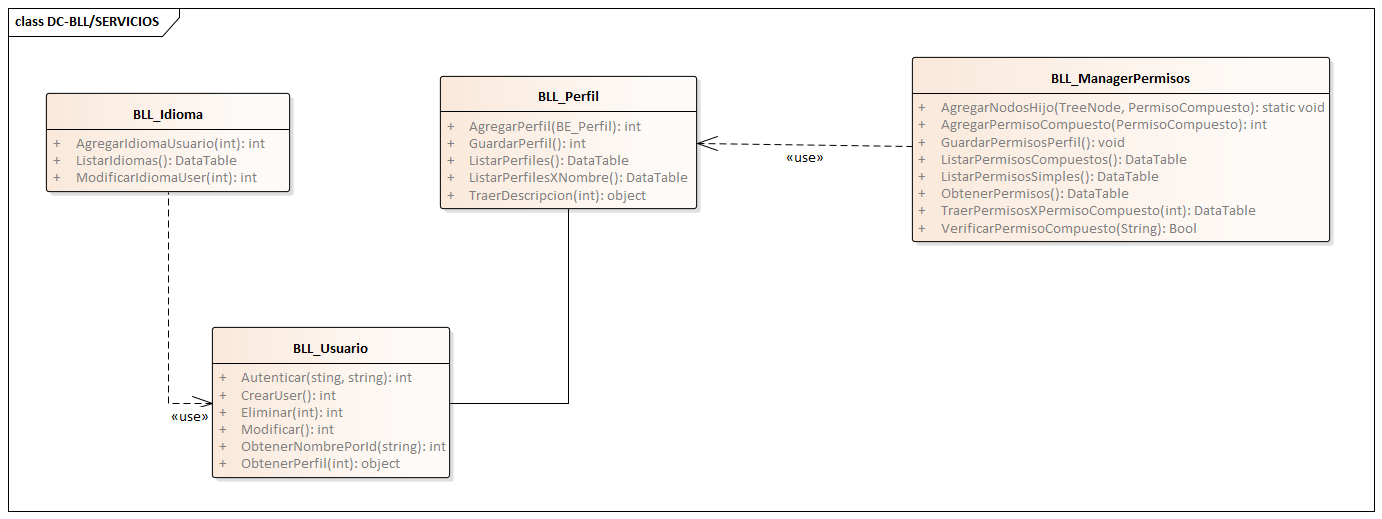
****

#### (G06.2.1) Capa de Entidad de Servicios

Diagrama

Descripción generada automáticamente

#### (G06.2.2) Capa de Negocios de Servicios



#### (G06.3) Capa de Datos

Modificar esta diagrama de capa de datos.

# (G07). DER COMPLETO

## (Punto 3.1 / N01-A ) Actores de PN1.

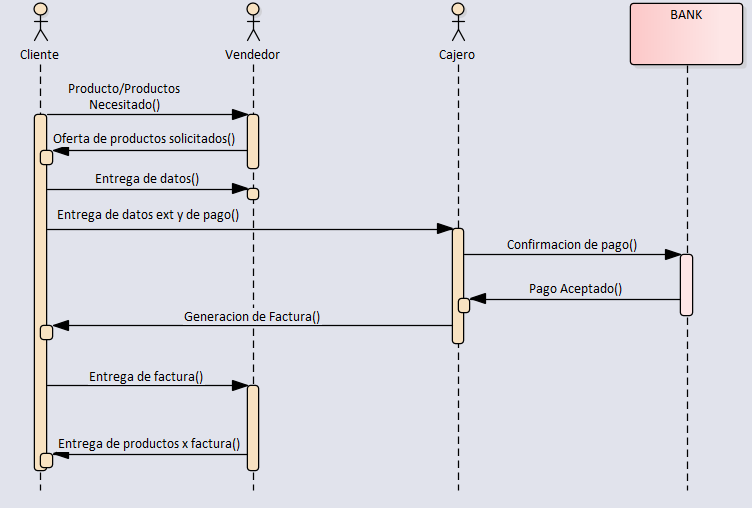
* Cliente (Persona- No es actor directo- No usa GUI- Fuente de informacion)
* Vendedor(Persona – Primario- Usa GUI)
* Cajero(Persona-Primario-Usa GUI)
* Banco(Sistema- Se conecta con el sistema-Remoto)

#### DIAGRAMA DE ROLES N0.1-B

Diagrama

Descripción generada automáticamente

#### DIAGRAMA DE SECUENCIA DE ROLES/N01-C



#### DIAGRAMA ECS.(Punto 3.1/N01-B)

Diagrama

Descripción generada automáticamente

#### DIAGRAMA DE PROCESOS.(Punto 4/N01-C)

Diagrama

Descripción generada automáticamente

#### (Punto 3.4 / N01-D Carpeta)

Diagrama

Descripción generada automáticamente

# N02.1 Diseño del negocio(RFN1-Gestion de ventas de la tienda)

Se procede a especificar con mayor detalle el punto G02, documentando el requerimiento funcional RFN1 - “Gestión de ventas de la tienda”.

#### N02.1.1-Diagrama de casos de uso (general)

Diagrama

Descripción generada automáticamente

#### N02.1.2 - Especificación de casos de uso

#### N02.1.2.1 - CU01 – Registrar Cliente

|  |
| --- |
| **Id y Nombre:** CU01- Registrar Cliente |
| **Objetivo:** Registrar a el cliente para que se pueda asociar una compra con el cliente. |
| **Actor Principal:**Cajero,Cliente |
| **Pre-Condiciones:** El vendedor debe estar logueado en el sistema. |
| **Puntos de Extension:-** |
| **Disparador:**  El cajero presiona el boton “Registrar cliente” en la seccion “Registrar” de la pantalla “Clientes” |
| **Post Condiciones:** El Cajero registro al cliente |
| **Escenario Principal:**  1.El Cajero presiona el boton “Registrar cliente” en la seccion “Registrar” de la pantalla “Clientes”.  2. El sistema carga la pantalla de la seccion “Registrar” que nos permite registrar al cliente.  3. El Cajero completa los campos de ,(Nombre, Apellido,Mail,Dni,Email) y presiona el boton “Registrar Cliente”.  4. El sistema valida que el cliente no este registrado anteriormente mediante el campo “DNI”.  5. El sistema actualiza la bitacora.  6. El sistema Muestra un mensaje “Cliente registrado correctamente”. |
| **Escenario Alternativo:**  2.1. El Cajero por indicacion del cliente decide no realizar ninguna compra y no continua con la registracion.  2.2 El Cajero muestra el menu principal.  4.1 El sistema muestra un mensaje de “Clienta registrado”. |

#### N02.1.3.2 - DS “CU01- Registrar Cliente”

Diagrama

Descripción generada automáticamente

#### N02.1.4 - Diagrama de Clases - “CU01 - Registrar Cliente”

#### N02.1.4.2 - DC “CU01- Registrar Cliente”

Diagrama

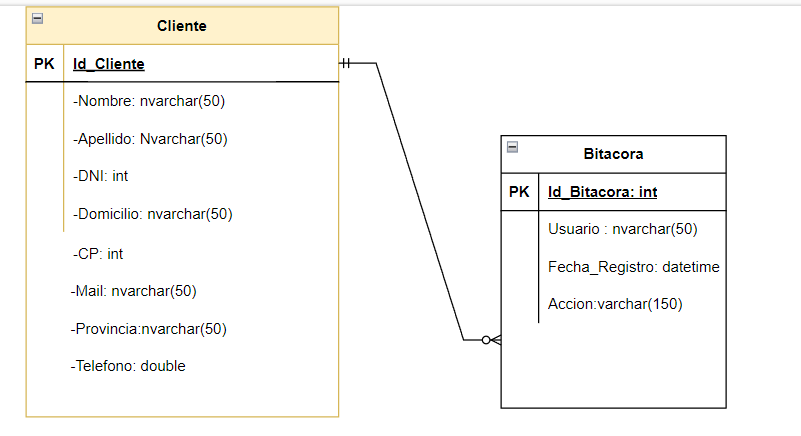
Descripción generada automáticamente

#### N02.1.6 – Diagrama de Actividad “CU01 – Registrar Cliente”

Diagrama

Descripción generada automáticamente

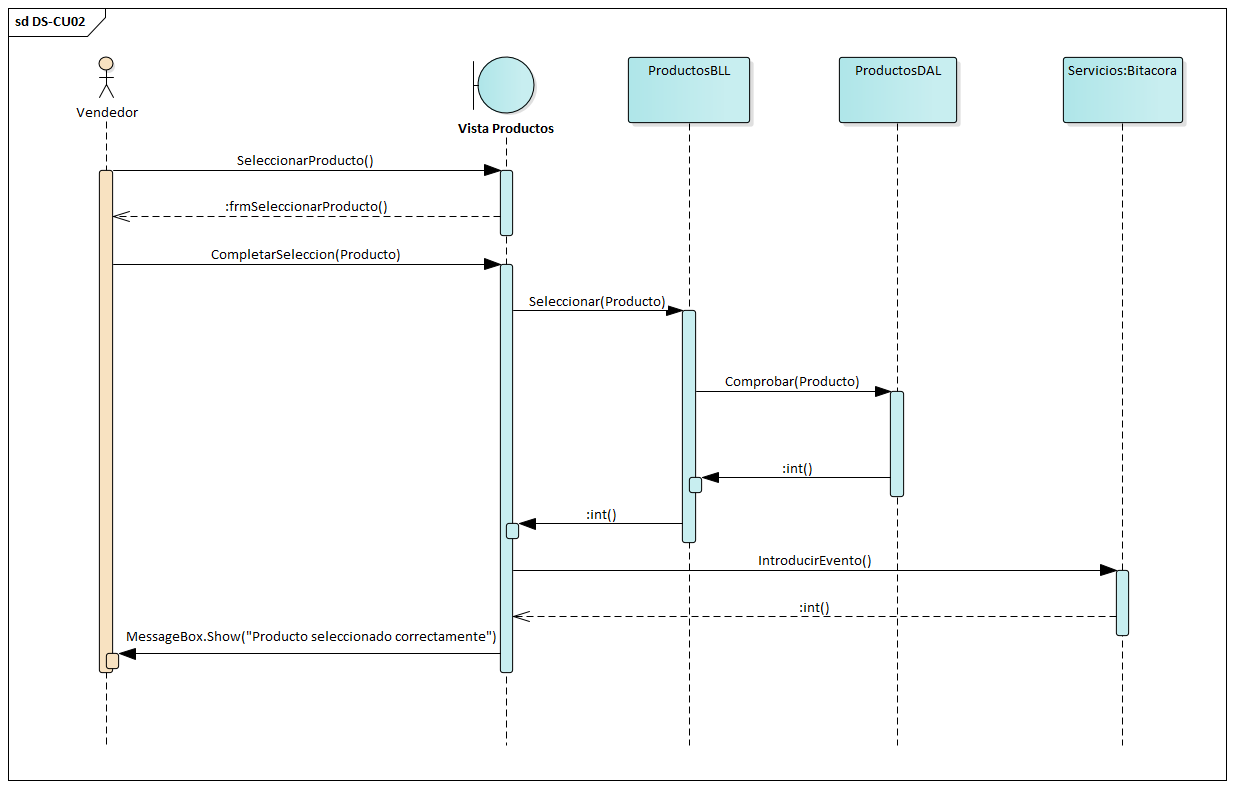
#### N02.1.6.2 – DER “CU01 – Registrar Cliente

****

#### N02.1.2.1 – “CU02 – Seleccionar Producto”

|  |
| --- |
| **Id y Nombre:** CU02- Seleccionar Producto |
| **Objetivo:** Seleccionar producto para introducir dentro de un carrito. |
| **Actor Principal:** Vendedor,Cliente |
| **Pre-Condiciones:** El vendedor debe estar logueado en el sistema y el cliente debe estar registrado. |
| **Puntos de Extension:** CU03 Llenar carrito |
| **Disparador:**  El vendedor apreta el boton de “Seleccionar productos” dentro de la seccion “Productos”. |
| **Post Condiciones:** El vendedor selecciona un producto. |
| **Escenario Principal:**  1.El Vendedor apreta el boton “Seleccionar productos” dentro de la seccion “Productos”.  2. El sistema carga la pantalla de la seccion “Productos”, mostrando todos los productos disponibles para ser seleccionados.  3. El Vendedor hace click sobre cada producto que desee seleccionar y selecciona las opciones de dicho producto como(Tipo,Tamaño,Cantidad).  4. El sistema selecciona ese producto.  5. El sistema actualiza la bitacora.  6. El sistema muestra un mensaje “El producto ha sido seleccionado correctamente” |
| **Escenario Alternativo:**  2.1 El vendedor sale de la seccion Productos ya que el cliente se arrepiente de seleccionar algun producto para la compra.  6.1 El sistema muestra un mensaje “El producto no pudo ser seleccionado” ya que ha habido un problema con la conexión con el servidor. |

**N02.1.3.2 - DS “CU02- Seleccionar Producto”**



**N02.1.4.2 - DC “CU02 -Seleccionar Producto”**

**Diagrama

Descripción generada automáticamente**

N02.1.6 – Diagrama de Actividad “CU02 –Seleccionar Producto”

Diagrama

Descripción generada automáticamente

**N02.1.6.2 – DER “CU02 – Seleccionar Producto**

**Diagrama

Descripción generada automáticamente**

**N02.1.3.2 - CU “CU03- Llenar carrito”**

|  |
| --- |
| **Id y Nombre:** CU03- Llenar carrito |
| **Objetivo:** El objetivo de esta especificacion es llenar el carrito con productos seleccionados. |
| **Actor Principal:** Vendedor,Cliente |
| **Pre-Condiciones:** El vendedor debe estar logueado en el sistema y el cliente debe estar registrado. |
| **Puntos de Extension:** CU02 Seleccionar producto |
| **Disparador:**  El vendedor Apreta el boton de “Llenar carrito” en la seccion de productos. |
| **Post Condiciones:** El vendedor llena el carrito de compras. |
| **Escenario Principal:**  1.El Vendedor apreta el boton “Llenar carrito” dentro de la seccion productos.  2. El sistema carga la pantalla de la seccion “Carrito” , donde estan todos los productos seleccionados previamente.  3. El Vendedor hace click sobre el boton “Cerrar carrito de compra”.  4. El sistema cierra el carrito.  5. El sistema valida que todos los productos seleccionados del carrito estan disponibles para la venta.  6. El sistema actualiza la bitacora.  7. El sistema Muestra un mensaje “Carrito cerrado correctamente”. |
| **Escenario Alternativo:**   * 1. El vendedor no apreta ningun boton ya que el cliente se arrepiente de realizar la compra y no desea llenar ningun carrito.   4.1 El sistema no puede cerrar el carrito ya que hay un problema con la conexión al servidor.  6.1 El sistema no puede actualizar la bitacora ya que hay un problema con la validacion del carrito.  7.1 El sistema que se muestra es “El carrito no pudo ser cerrado correctamente”, debido a que hay un problema con la validacion del carrito o con la conexión con el servidor. |

**N02.1.3.2 - DS “CU03- Llenar carrito”**

**Diagrama

Descripción generada automáticamente**

#### N02.1.4 - Diagrama de Clases - “CU03 -Llenar carrito”

**N02.1.4.2 - DC “CU03 -Llenar Carrito”**

**Diagrama

Descripción generada automáticamente**

N02.1.6 – Diagrama de Actividad “CU03 – Llenar Carrito”

Diagrama

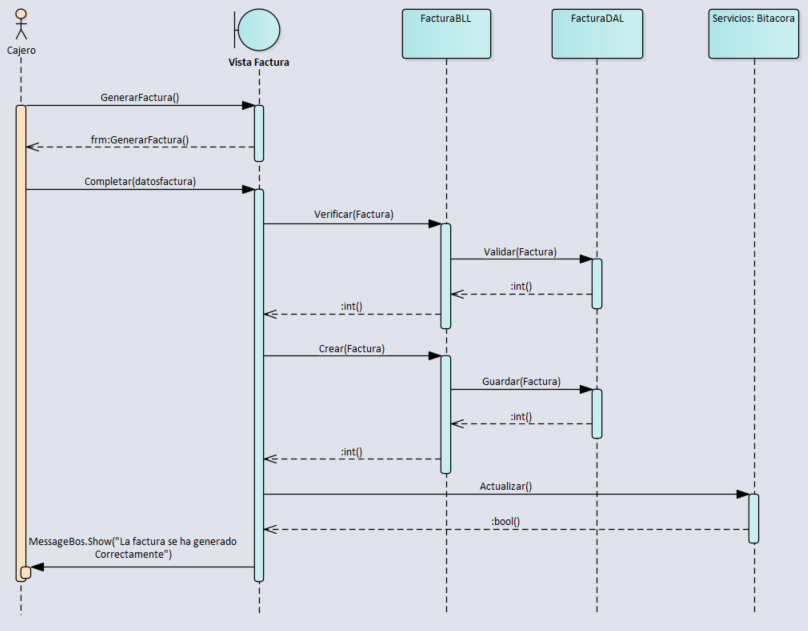
Descripción generada automáticamente

**Captura de pantalla de computadora

Descripción generada automáticamenteN02.1.6.2 – DER “CU03 – Llenar Carrito**

**N02.1.3.2 - CU “CU04-Generar Factura”**

|  |
| --- |
| **Id y Nombre:** CU04- Generar Factura |
| **Objetivo:** Se desea generar una factura para una venta ya realizada. |
| **Actor Principal:**Cajero,Cliente |
| **Pre-Condiciones:** El cajero debe estar logueado en el sistema. |
| **Puntos de Extension:-** Registrar Cliente – Cobrar Venta |
| **Disparador:**  El cajero presiona el boton “ Generar factura “ que se encuentra en la seccion de facturas |
| **Post Condiciones:** El Cajero genero la factura de una venta realizada correctamente |
| **Escenario Principal:**  1.El Cajero presiona el boton “Generar factura” en la seccion de Ventas.  2. El sistema carga la pantalla de la seccion “Facturas” donde se muestran unas opciones a completar obligatoriamente como(Tipo de factura, ventas generadas, cantidad de copias,Nombre del que recibe la factura, dni).  3. El Cajero selecciona la venta por la cual desea realizar la factura.  4. El sistema valida que la factura no se haya realizado ya anteriormente.  5. El sistema actualiza la bitacora  6. El sistema Muestra un mensaje “Factura generada correctamente”. |
| **Escenario Alternativo:**  1.1 El Cajero por orden del cliente decide no generar la factura y cancela la operación.  3.1 El Sistema no muestra ninguna venta debido a un error con la conexión al servidor.  4.1 El sistema muestra un mensaje de “Factura generada”. |

**N02.1.3.2 - DS “CU04- Generar Factura”**

#### N02.1.4 - Diagrama de Clases - “CU04 – Generar Factura”

N02.1.4.2 - DC “CU04 – Generar Factura”

Diagrama

Descripción generada automáticamente

**N02.1.5.2 – Diagrama de Actividad “CU04 – Generar Factura”**

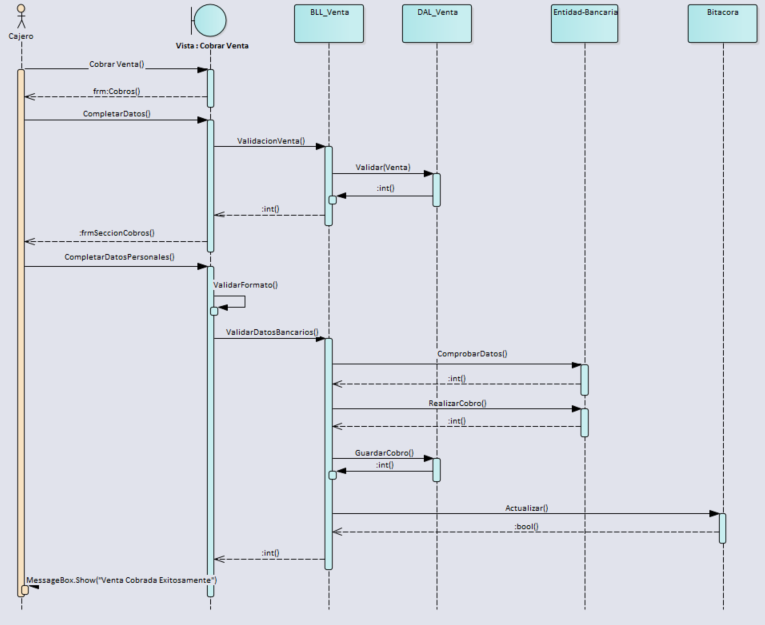
**N02.1.6.2 – DER “CU04 – Generar Factura”**

**Diagrama

Descripción generada automáticamente**

**N02.1.3.2 - CU “CU05- Cobrar Venta”**

|  |
| --- |
| **Id y Nombre:** CU05- Cobrar Factura |
| **Objetivo:** Se desea cobrar el dinero de una Factura ya realizada. |
| **Actor Principal:**Cajero,Cliente |
| **Pre-Condiciones:** El cajero debe estar logueado en el sistema. |
| **Puntos de Extension:-** Registrar Cliente – Generar Factura |
| **Disparador:**  El cajero presiona el boton “cobrar factura” en la seccion de facturas |
| **Post Condiciones:** El Cajero genero la factura de una venta realizada correctamente |
| **Escenario Principal:**  1.El Cajero presiona el boton “Cobrar Factura” de la seccion de Factura.  2. El sistema carga la pantalla de la seccion de cobros donde se pueden ver los cobros realizados y los cobros a realizar.  3. El Cajero selecciona el cobro de la Factura a realizar  4. El sistema valida que la Factura no ha sido cobrado anteriormente.  5. El sistema muestra la seccion de cobros con las opciones a completar para realizar el cobro como(Numero de tarjeta de credito,DNI,clave de seguridad de la tarjeta).  6. El Cajero completa todos los datos para realizar el cobro de la venta y apreta el boton “Pagar”  7. El sistema valida los datos bancarios.  8. El sistema actualiza la bitacora.  9. El sistema muestra un mensaje “La venta ha sido cobrada exitosamente”. |
| **Escenario Alternativo:**   * 1. El Cajero sale al menu principal ya que el cliente se arrepiente de realizar la compra.   4.1 El sistema valida la venta y ya ha sido cobrada por lo que manda un mensaje a la pantalla diciendo “La venta ya ha sido cobrada”.  7.1 El sistema muestra un mensaje de error debido a un problema con la conexión al servidor bancario. |

**N02.1.3.2 - DS “CU05-Cobrar Venta”**

#### N02.1.4 - Diagrama de Clases - “CU05 -Cobrar Venta”

N02.1.4.2 - DC “CU05 – Cobrar Venta”

Diagrama

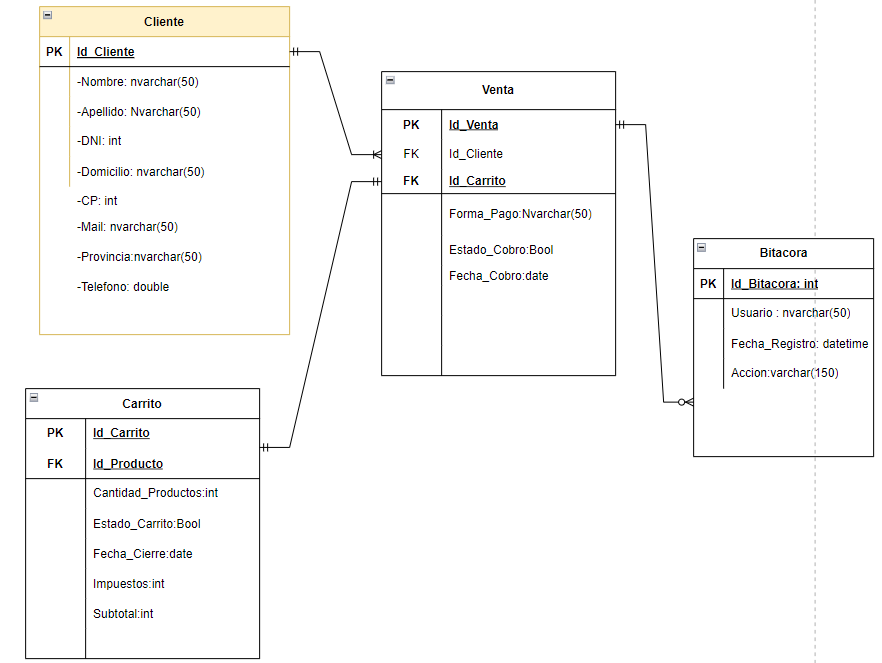
Descripción generada automáticamente

N02.1.5.2 – Diagrama de Actividad “CU05 – Cobrar Venta”

Diagrama

Descripción generada automáticamente

**N02.1.6.2 – DER “CU05 – Cobrar Venta”**

****

**N02.1.3.2 - CU “CU05- Cobrar Venta”**

|  |
| --- |
| **Id y Nombre:** CU06 – Despachar Producto |
| **Objetivo:** Se desea despachar los productos de una venta. |
| **Actor Principal:** Vendedor,Cliente |
| **Pre-Condiciones:** El cajero debe estar logueado en el sistema, debe haber una venta ya realizada y cobrada, debe haber stock de los productos solicitados en la venta, todos los productos de la vente deben estar en el mismo punto de retiro. |
| **Puntos de Extension: -** |
| **Disparador:**  El Vendedor presiona el boton “Despachar venta” en la seccion Despachos. |
| **Post Condiciones:** El vendedor despacha los productos solicitos por el cliente de manera exitosa. |
| **Escenario Principal:**  1.El Vendedor presiona el boton “Despachar venta” en la seccion de venta  2. El sistema carga la pantalla de la seccion de despachos donde se pueden ver todas las ventas realizadas y cobradas.  3. El Vendedor selecciona la venta que desea despachar y presiona el boton “Despachar”.  4. El sistema valida que la factura ha sido cobrada anteriormente.  5. El sistema valida que todos los productos de la venta seleccionada se encuentra en el mismo punto.  6. El Vendedor entrega los productos al cliente.  7. El sistema actualiza el stock de los productos.  8. El sistema actualiza la bitacora.  9. El sistema muestra un mensaje “El despacho de los productos se ha realizado con exito”. |
| **Escenario Alternativo:**   * 1. El Cajero sale al menu principal ya que el cliente se arrepiente de realizar el despacho en el local seleccionado.   4.1 El sistema encuentra que la venta seleccionada no ha sido cobrada previamente.  5.1 El sistema encuentra que faltan productos en el punto de venta seleccionado para el despacho de productos. |

Diagrama

Descripción generada automáticamente**N02.1.3.2 - DS “CU06-Despachar Productos”**

#### N02.1.4 - Diagrama de Clases - “CU06 -Despachar Producto”

N02.1.4.2 - DC “CU06 – Despachar Producto”

Diagrama

Descripción generada automáticamente

**N02.1.6.2 – DER “CU06 – Despachar Productos”**

**Diagrama

Descripción generada automáticamente**

(Punto 3.2 / N02-A ) Actores de PN2.

* Despachador (Persona-No es actor directo-No usa GUI)
* Banco(Sistema- Se conecta con el sistema-Remoto)
* Suministrador(Persona-Primario-Usa GUI)
* Administrador(Persona-Primario-Usa GUI)
* Proveedor (Persona-Secundario-No Usa GUI)

*Diagrama de Roles / N02-B*

Diagrama

Descripción generada automáticamente

*DIAGRAMA DE SECUENCIA DE ROLES/N02-C*

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

*DIAGRAMA ECS.(Punto 3.2/N02-B)*

Interfaz de usuario gráfica, Diagrama

Descripción generada automáticamente con confianza media

*DIAGRAMA DE PROCESOS.(Punto 4/N0.2-C)*

Diagrama

Descripción generada automáticamente

*(Punto 3.4 /N02-D Carpeta)*

Diagrama

Descripción generada automáticamente

#### N02.2 - Diseño del negocio(RFN2- Gestion de compras de Producto)

#### N02.2.1 Diagrama de casos de uso General

Diagrama

Descripción generada automáticamente

#### N02.2 1.1 Diagrama de secuencia del CU01 Generar solicitud de cotizacion.

|  |
| --- |
| **Id y Nombre:** CU01-Generar solicitud de cotizacion |
| **Objetivo:** El objetivo es generar una solicitud de cotizacion |
| **Actor Principal:** Encargado de compras |
| **Pre-Condiciones:** - |
| **Puntos de Extension:** - |
| **Disparador:**  El Suministrador realiza la solicitud de cotizacion presionando el boton Solicitar cotizacion. |
| **Post Condiciones:** El Suministrador realizo con éxito la solicitud de cotizacion. |
| **Escenario Principal:**  1.El sistema carga en pantalla el menu principal.  2. El encargado de compras abre el apartado de cotizaciones.  3. El sistema carga el menu de cotizaciones.  4.El encargado revisa en una lista de productos que estan cargados con poco stock <15.  5. El encargado selecciona todos los que desea reponer agregandolos a una lista.  6. El encargado aprieta el boton solicitar cotizacion.  7. El sistema guarda la cotizacion solicitada.  8. El sistema envia al proveedor la cotizacion.  9. El sistema guarda el evento en la bitacora, actualizando la grilla y mostrando dicho evento en ella. |
| **Escenario Alternativo:**  4.1 No hay productos con poco stock por lo que sale y vuelve al menu principal.  5.1 Si el encargado selecciona y aprieta “Eliminar” se elimina el producto de la lista. |

#### N02.2 1.3 Diagrama de secuencia del CU01 Generar solicitud de cotizacion.

Diagrama

Descripción generada automáticamente

#### N02.2 1.4 Diagrama de Clases del CU01 Generar solicitud de cotizacion

Diagrama

Descripción generada automáticamente

#### N02.2 1.6 DER CU01 Generar Solicitud Cotización

Mapa de una computadora

Descripción generada automáticamente con confianza baja

#### N02.2 1.7 GUI CU02 Generar Solicitud Cotización

Interfaz de usuario gráfica

Descripción generada automáticamente

#### N02.2 2.2 Especificación del CU02 Generar orden de Compra

|  |
| --- |
| **Id y Nombre:** CU01-Generar orden de compra |
| **Objetivo:** Se desea Generar una orden de compra |
| **Actor Principal:** Administrador |
| **Pre-Condiciones:** Se necesita realizar una compra a un proveedor. |
| **Puntos de Extension:** CU05-Registrar Proveedor , CU03-Pagar |
| **Disparador:**  El Administrador genera la orden de compra mediante un boton. |
| **Post Condiciones:** El Administrador genero exitosamente la orden de compra. |
| **Escenario Principal:**  1.El Administrador recibe la cotizacion por parte del proveedor y abre el sistema.  2. El sistema carga la seccion de Compras.  3. El Administrador ingresa a la pantalla de Compras y muestra las solicitudes de cotizaciones.  4. El Administrador selecciona la cotizacion que desea aprobar.  5. El administrador aprieta el boton “Aprobar”.  6. El sistema convierte la solicitud en una orden de compra. |
| **Escenario Alternativo:**   * 1. El admin no recibe ninguna cotizacion por lo que no se puede proceder a abrir el sistema.   4.1 El admin revisa las solicitudes y no desea aprobar nada ya que no esta de acuerdo con los precios previamente pactados.  6.1 El sistema por un error de tipeo no pudo cargar correctamente la orden de compra. |

#### N02.2 2.3 Diagrama de secuencia del CU02 Generar orden de Compra

Diagrama

Descripción generada automáticamente con confianza baja

#### N02.2 2.4 Diagrama de Clases del CU02 Generar orden de Compra

Diagrama

Descripción generada automáticamente

#### N02.2 2.6 DER CU02 Generar orden de Compra

Gráfico

Descripción generada automáticamente con confianza baja

#### N02.2 3.2 Especificación del CU03 Pagar

|  |
| --- |
| **Id y Nombre:** CU03 – Pagar |
| **Objetivo:** El objetivo es pagar la orden de compra realizada |
| **Actor Principal:** Administrador |
| **Pre-Condiciones:** Tener una orden de compra generada |
| **Puntos de Extension:** CU01 Generar orden de compra |
| **Disparador:**  El administrador aprieta el boton “pagar” |
| **Post Condiciones**: Se ha realizado el pago con exito |
| **Escenario Principal:**  1.El sistema carga la seccion de pagar.  2. El administrador selecciona la orden de compra aprobada que desea pagar.  3. El sistema realiza y registra la compra.  4.El sistema emite un mensaje de “Compra registrada y realizada”. |
| **Escenario Alternativo:**  3.1 En caso de que el administrador presione “No” el sistema cambiara de aprobada a 0. |

#### N02.2 3.3 Diagrama de secuencia del CU03 Pagar

Diagrama

Descripción generada automáticamente

#### N02.2 3.4 Diagrama de clases del CU03 Pagar

Diagrama

Descripción generada automáticamente

#### N02.2 3.6 DER CU03 Pagar

Tabla

Descripción generada automáticamente

#### N02.2 4.2 Especificación del CU04 Gestion de Reposicion de Productos

|  |
| --- |
| **Id y Nombre:** CU04 – Gestion de Reposicion de Productos |
| **Objetivo:** El objetivo es moficar o agregar algun producto |
| **Actor Principal:** Encargado de entregas |
| **Pre-Condiciones:** Se requiere gestionar los productos |
| **Puntos de Extension: -** |
| **Disparador:**  El encargado de entregas entra al modulo de gestion de productos. |
| **Post Condiciones**: Se han realizado los cambios con éxito. |
| **Escenario Principal:**  1.El sistema carga la seccion de Gestion de productos.  2.En caso de que la intencion sea registrar un nuevo producto el encargado ingresara los datos del producto(stock, codigo, descripcion, precio, etc) y presionara agregar.  3. El sistema registrara el nuevo producto.  4. En caso de que se haya recibido una entrega de un proveedor, y se requiera actualizar el stock de los productos, el encargado ingresará el código de cada producto junto a la cantidad recibida y presionará “Sumar stock”.  5.El sistema suma el stock correspondiente a los productos ingresados. |
| **Escenario Alternativo:**  3.1 En caso de que el producto ya se encuentre registrado se mostrara en la pantalla un mensaje de “producto registrado”.  5.1 En caso de que el producto no exista en el sistema se mostrara un mensaje de “ Producto inexistente”. |

#### N02.2 4.3 Diagrama de secuencia del CU04 Gestión de reposicion de Productos

Diagrama

Descripción generada automáticamente

#### N02.2 4.4 Diagrama de clases del CU04 Gestión de reposicion de Productos

Diagrama

Descripción generada automáticamente

#### N02.2 4.6 DER CU04 Gestión de reposicion de Productos

Tabla, Excel

Descripción generada automáticamente

#### N02.2 4.7 GUI CU04 Gestión de reposicion de Productos

-

#### N02.2 5.2 Especificación del CU05 Registro del proveedor

|  |
| --- |
| **Id y Nombre:** CU05 – Registro del proveedor |
| **Objetivo:** El objetivo es realizar un registro con los datos completos del proveedor. |
| **Actor Principal:** Administrador |
| **Pre-Condiciones:** Se necesita registrar los datos faltantes de un proveedor. |
| **Puntos de Extension:** CU03 Pagar- CU02 Generar orden de compra |
| **Disparador:**  El encargado de entregas entra al modulo de gestion de productos. |
| **Post Condiciones**: Se registro un nuevo proveedor con éxito. |
| **Escenario Principal:**  1.El sistema carga la seccion de Proveedores.  2. El administrador selecciona el proveedor que desea registrar.  3. El encargado completa los datos faltantes y aprieta guardar.  4. El sistema registra los datos del nuevo proveedor. |
| **Escenario Alternativo:**  **-** |

#### N02.2 5.3 Diagrama de secuencia del CU05 Registro del proveedor

Diagrama

Descripción generada automáticamente

#### N02.2 5.4 Diagrama de clases CU05 Registro de proveedor

Diagrama

Descripción generada automáticamente

#### N02.2 5.6 DER CU05 Registro de Proveedor

Tabla

Descripción generada automáticamente

#### N02.2 6.2 Especificación del CU06 Verificar Compra

|  |
| --- |
| **Id y Nombre:** CU06 – Verificar Compra |
| **Objetivo:** El objetivo es verificar la compra realizada |
| **Actor Principal:** Despachador |
| **Pre-Condiciones:** Se debe haber realizado una compra, se debe haber recibido los productos y factura |
| **Puntos de Extension: -** |
| **Disparador:**  El despachador abre la ventana de Verificacion de compras y aprieta el boton verificar compra. |
| **Post Condiciones**: Se verifico una compra con éxito. |
| **Escenario Principal:**  1.El sistema carga la seccion de Verificacion de compras.  2. El Despachador revisa la lista de productos recibidos  3. El despachador aprieta el boton, mostrar orden de compra  4. El sistema muestra la orden de compra con la lista de productos y sus cantidades correspondientes.  5. El despachador aprieta el boton verificar compra.  6. El sistema abre un pop up de verificacion.  7. El despachador aprieta el boton correcto.  8. El sistema guarda la verificacion y cierra el pop up.  9. El sistema guarda el evento en la bitacora.  10. El sistema emite un mensaje “Orden de compra verificada correctamente”. |
| **Escenario Alternativo:**   * 1. El despachador aprieta el boton incorrecto. |

#### N02.2 6.3 Diagrama de secuencia del CU006 Verificar orden de compra.

Diagrama

Descripción generada automáticamente

#### N02.2 6.4 Diagrama de clases CU06 Verificar orden de compra

Diagrama

Descripción generada automáticamente

#### N02.2 6.6 DER CU06 Verificar orden de compra

Interfaz de usuario gráfica

Descripción generada automáticamente con confianza baja

# T01- Arquitectura Base

#### T01.1 – Diseño de la Arquitectura. (Diagrama de capas)

Diagrama

Descripción generada automáticamente

#### T01.2 Captura de pantalla de las capas creadas en el IDE

Texto

Descripción generada automáticamente

#### T01.3 – Diseño modelo del DS de Persistencia de Datos

Diagrama

Descripción generada automáticamente

#### T01.4 – Diseño modelo del DS de Consulta de Datos

Diagrama

Descripción generada automáticamente

#### T01.5 – Diagrama Jerarquico de la estructuracion del menu principal

Diagrama

Descripción generada automáticamente

#### T01.6 – Captura de pantalla del Menu Principal

# T02. Gestion de Usuario

#### T02.1 – Crear Usuario

T02.1.1 – Descomposicion Funcional

* El Administrador ingresa al menú principal, luego aprieta en la opcion “Registrarse”.
* El sistema depliega una pantalla donde se podra llenar todos los datos para el futuro usuario.
* El Usuario completa todos los datos y aprieta el boton “Registrarse”.
* El sistema verifica que no exista un usuario registrado previamente en el sistema con ese nombre y dni.
* El sistema guarda el usuario nuevo en el sistema.
* En caso de no existir, el sistema muestra un mensaje “ se ha registrado en el sistema exitosamente”.
* El sistema actualiza la grilla de la bitacora de eventos mostrando el registro de dicho usuario.
* Finalmente el sistema actualiza la grilla de usuarios del sistema.

T02.1.2 – Diagrama de casos de uso

Diagrama

Descripción generada automáticamente con confianza media

T02.1.3 – Especificacion de casos de uso – “CU07-Crear usuario”

|  |
| --- |
| **Id y Nombre:** CU07 – Crear Usuario |
| **Objetivo:** Se desea crear un usuario en el sistema. |
| **Actor Principal:** Empleado |
| **Pre-Condiciones: -** |
| **Puntos de Extension: -** |
| **Disparador:**  El empleado ingresa a la seccion de “Registrar”. |
| **Post Condiciones:** El empleado creo una nueva cuenta en el sistema. |
| **Escenario Principal:**  1.El usuario con permisos del administrador ingresa al menu principal y selecciona la opcion “Registrarse”.  2. El sistema muestra una pantalla con todas las opciones (Nombre,Contraseña,Mail,Rol,Estado,DNI).  3. El empleado completa todos los datos mostrados en pantalla y posteriormente aprieta el boton “Registrarse”.  4. El sistema valida que los datos ingresados del usuario no existen.  4. El sistema muestra en pantalla un mensaje “Se ha registrado exitosamente en el sistema”.  5. El sistema actualiza la bitacora.  6. El sistema actualiza los usuarios de la grilla en la seccion de “Administracion”. |
| **Escenario Alternativo:**   * 1. El usuario no tiene permisos del administrador por lo que no puede ingresar al sistema.   3.1 El sistema muestra un mensaje “Falta completar datos” debido a que faltan completar datos para la carga del usuario.  4.1 El sistema muestra un mensaje “Usuario existente” debido a que el sistema verifica que el usuario ingresado ya existe en el sistema.  4.2 El sistema muestra un mensaje de “Datos incorrectos” debido a que los tipos de datos no son validos. |

Diagrama

Descripción generada automáticamenteT02.1.4 – Diagrama de Secuencia – “CU07-Crear usuario”

T02.1.4 –Diagrama de Clases– “CU07-Crear usuario”

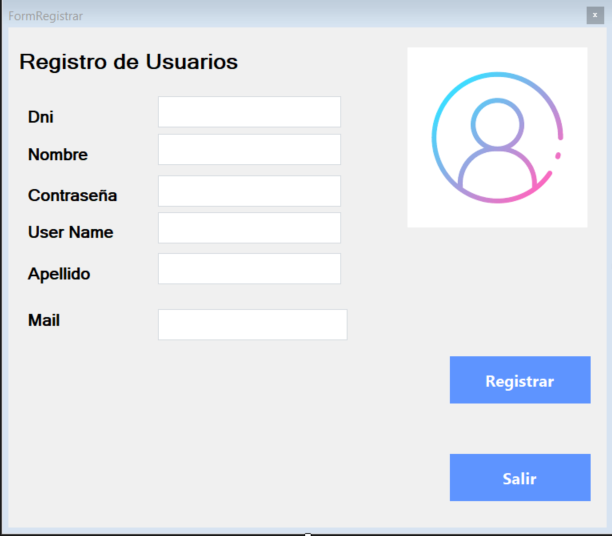
Diagrama

Descripción generada automáticamente

T02.1.5 –Diagrama e Entidad-Relacion – “CU07-Crear usuario”



T02.1.7 – GUI – “CU07-Crear usuario”



#### T02.2 - Login

T02.2.1 - Descomposición funcional

1. El usuario ingresa al menú de Login al iniciar el programa.
2. El usuario ingresa las casillas de “Nombre” y “Contraseña” con sus datos de usuario , posteriormente aprieta el boton de “Iniciar Sesion”.
3. El sistema valida que los datos de entrada no esten vacios y cumplan con ciertos criterios.
4. El sistema verifica los datos introducidos por el usuario contra la base de datos del sistema.
5. Posteriormente el sistema verifica si el usuario esta bloqueado o activo.
6. El sistema gestiona la creacion de una nueva sesion para el usuario autenticado.
7. Se lleva un registro de los intentos fallidos de inicio de sesion y se bloquea la cuenta despues de un numero determinado de intentos.
8. El sistema ingresa el evento a la bitacora, actualizando la grilla para mostrar dicho evento.
9. Posteriormente el sistema proporciona retroalimentacion sobre el estado del login.

T02.2.2 - Diagrama de casos de uso – Login

Diagrama

Descripción generada automáticamente

T02.2.3 - Especificación de casos de uso - “CU08- Realizar Login”

|  |
| --- |
| **Id y Nombre:** CU08 – Log In |
| **Objetivo:** El empleado ingresara los datos para loguearse. |
| **Actor Principal:** Empleado |
| **Pre-Condiciones: -** |
| **Puntos de Extension: -** |
| **Disparador:**  El usuario esta iniciando sesion. |
| **Post Condiciones:** El empleado inicio sesion en el sistema. |
| **Escenario Principal:**  1.El Usuario entra al sistema.  2. El sistema muestra la primer pantalla como “Inicio de Sesion” y muestra los campos nombre y contraseña con distintos botones como (Iniciar Sesion, Registrarse y Salir).  3. El usuario completa todos los campos como (Nombre,Contraseña) y aprieta el boton “Iniciar Sesion”.  4. El sistema valida los campos introducidos.  5.El sistema autentica el nombre y la contraseña.  6.El sistema muestra un mensaje de “Inicio de Sesion Exitoso”. |
| **Escenario Alternativo:**   * 1. El empleado no pude entrar al sistema debido a un error interno.   4.1 El sistema muestra un mensaje de error debido a una inconsistencia con el formato de datos introducidos en los campos.  5.1 El sistema muestra un mensaje de error debido a que el nombre o la contraseña son incorrectos. |

T02.2.4 – Diagrama de Secuencia – “CU08-Login”

Diagrama

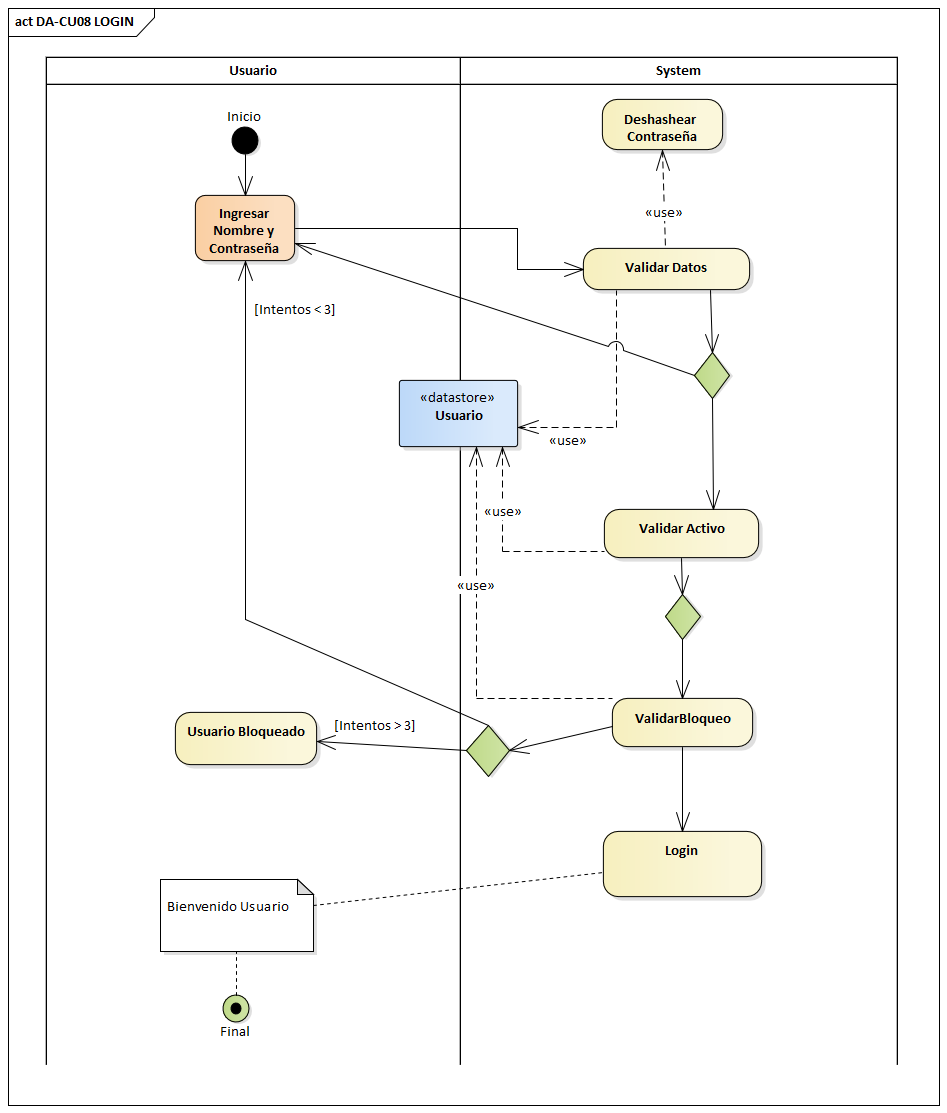
Descripción generada automáticamente

T02.2.5 – Diagrama de Clases – “CU08-Login

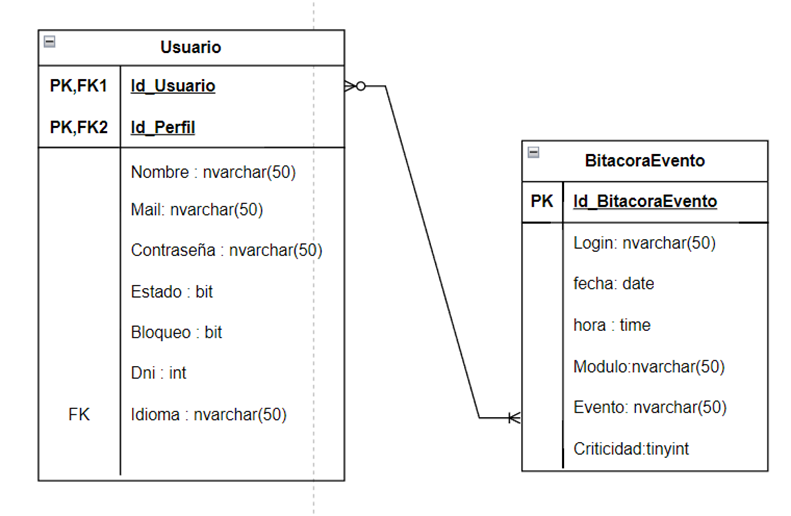
Diagrama

Descripción generada automáticamente

T02.2.6– Diagrama de Actividad – “CU08-Login”



T02.2.7 – Diagrama de Entidad/Relacion – “CU08-Login”



T02.2.8 - GUI - “CU08 - Login”

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

#### T02.3- Desbloquear Usuario

T02.3.1 - Descomposición funcional

1. Se carga el menu de administrador de usuarios.
2. Se muestra todos los usuarios bloqueados y desbloqueados, ademas de los respectivos campos como (Nombre,Dni,UserName).
3. El sistema muestra una grilla con todos los usuarios, viendose asi el usuario con su estado que se desea bloquear
4. Se completan los campos solicitados para realizar el desbloqueo de dicho usuario, se necesita.
5. Una vez completado los campos se procede a verificar que la validacion regex de cada una.
6. Posteriormente se verifica que el usuario existe y que este desbloqueado en el sistema.
7. Una vez verificado se procede a realizar el desbloqueo del usuario.
8. El sistema muestra una retroalimentacion del estado del usuario.
9. El sistema finalmente registra el evento en la bitacora y actualiza la grilla mostrando dicho evento.

T02.3.2 - Especificación de casos de uso - “CU11 – Desbloquear Usuario”

|  |
| --- |
| **Id y Nombre:** CU11 – Desbloquear Usuario |
| **Objetivo:** El Empleado desbloquea un usuario en el sistema. |
| **Actor Principal:** Empleado |
| **Pre-Condiciones:** El usuario debe estar bloqueado en el sistema.  El empleado que realice el desbloqueo de usuario debe tener los permisos por el administrador.  El empleado debe estar logueado en el sistema. |
| **Puntos de Extension: -** |
| **Disparador:**  El empleado realiza el desbloqueo del usuario completando los campos y apretando el boton “Desbloquear Usuario”. |
| **Post Condiciones:** Se ha desbloqueado el usuario en el sistema. |
| **Escenario Principal:**  1.El empleado se loguea en el sistema.  2. El sistema muestra el menu principal.  3. El empleado selecciona el apartado de Administracion de Usuarios.  4. El sistema muestra la interfaz de Administracion de Usuarios.  5. El empleado revisa que en la grilla actualizada de los usuarios se encuentra el usuario con el estado bloqueado.  6.El empleado completa los campos (Nombre,Dni y Username) y apreta el boton desbloquear Usuario.  7. El sistema verifica que los campos esten correctos a traves de validaciones regex  8. El sistema desbloquea el usuario mediante los campos completados anteriormente.  9. El sistema cambia la contraseña a su nombre de usuario por default.  10. El sistema muestra un mensaje de retroalimentacion del estado del usuario.  11. El sistema guarda el evento en la bitacora y actualiza los eventos. |
| **Escenario Alternativo:**  3.1 El empleado no tiene permisos para acceder a la pestaña de administracion por lo que no procede a realizar el desbloqueo.  5.1 El empleado revisa en la grilla de usuarios y verifica que el usuario esta desbloqueado.  7.1 El sistema muestra un mensaje de error debido a que los campos no pasaron las validaciones correspondientes. |

T02.3.4 – Diagrama de Secuencia – “CU11- Desbloquear Usuario”

Diagrama

Descripción generada automáticamente

T02.3.5 – Diagrama de Clases – “CU11-Desbloquear Usuario”

Diagrama

Descripción generada automáticamente

T02.3.7 – Diagrama de Entidad/Relacion – “CU11 Desbloquear Usuario”

Diagrama

Descripción generada automáticamente

#### T02.4 - Cambiar Clave

T02.4.1 - Descomposición funcional

1. Se carga el menu de Inicio de sesion, mostrando los campos de (Nombre,Contraseña), los botones de iniciar sesion, registrarse o cambiar clave.

2. El usuario aprieta el boton de cambiar clave.

3. Se muestra la GUI de cambiar clave con los campos de “Nombre”,”Contraseña”,”Dni”,”Contraseña nueva” con doble campo para verificar que la contraseña introducida para cambiar es la correcta.

4. El usuario completa los datos mostrados en la GUI para cambiar la clave.

5. Se verifica que la cuenta no este bloqueada y se encuentra activa para realizar el cambio de clave.

6. Se autentica el usuario en la base de datos para comprobar que esta registrado.

7. Posteriormente se procede a cambiar la clave de dicho usuario en la base de datos.

8. Posteriormente el sistema proporciona retroalimentacion sobre el estado del login.

9. Finalmente se registra el evento en la bitacora de eventos y se actualiza la grilla

T02.4.2 - Especificación de casos de uso - “CU09 - Cambiar contraseña”

|  |
| --- |
| **Id y Nombre:** CU09 – Cambiar Contraseña |
| **Objetivo:** El administrador cambia la contraseña de dicho usuario |
| **Actor Principal:** Empleado |
| **Pre-Condiciones: -** |
| **Puntos de Extension: -** |
| **Disparador:**  El empleado cambia la contraseña de un usuario determinado. |
| **Post Condiciones:** Se ha modificado la contraseña de un usuario del sistema**.** |
| **Escenario Principal:**  1.El empleado entra al sistema.  2. El sistema muestra la primer pantalla como “Inicio de Sesion” y muestra los campos nombre y contraseña con distintos botones como (Iniciar Sesion, Registrarse, Salir y cambiar clave).  3. El empleado apriete el boton de cambiar clave.  4. El sistema muestra el menu de Cambiar Clave.  5.El empleado completa los campos de (Nombre,Dni) para validar la existencia del usuario, luego completa los campos de contraseña nueva y aprieta el boton “Cambiar Clave”.  6.El sistema valida los datos del usuario.  7. El sistema cambia la contraseña del usuario mencionado en los campos (Dni y nombre).  8. El sistema muestra un mensaje “Se ha actualizado la contraseña del usuario en el sistema”.  9. El sistema registra el evento en la bitacora, actualizando la grilla y mostrando dicho evento en ella. |
| **Escenario Alternativo:**   * 1. El empleado no pude entrar al sistema debido a un error interno.   5.1 El empeado no pudo completar los datos debido a que el usuario se olvido el nombre de su cuenta.  6.1 El sistema valida los datos del usuario pero no encuentra ningun usuario asociado con el nombre o dni mencionado por eso muestra un mensaje “Nombre o Dni incorrectos”. |

T02.4.4 – Diagrama de Secuencia – “CU09-Cambiar Contraseña”

T02.4.5 – Diagrama de Clases – “CU09-Cambiar Contraseña

Diagrama

Descripción generada automáticamente

T02.4.7 – Diagrama de Entidad/Relacion – “CU09-CambiarContraseña”

Diagrama

Descripción generada automáticamente

Agregar Bitacora de eventos al der

T02.4.8 - GUI - “CU09 - Cambiar contraseña”

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

#### T02.5 - Logout

T02.5.1 - Descomposición funcional

1. Para poder cerrar la sesión activa, el usuario presiona la opción “Logout” desde las diferentes GUI como “Menu principal” , “Administracion de usuarios”.

2. El sistema carga una pantalla de confirmación, en caso de que el usuario haya apreta el boton sin intencion de realizar el logout.

3. El sistema carga el GUI de logout, donde te muestra dos botones “Si” en caso de que quiera realizar el logout y “No” para cancelar la operación y volver al menu principal.

4. El usuario selección la operación que desea realizar mediante los dos botones.

5. Al apretar si, el sistema realiza el logout del usuario ingresado previamente en la pantalla del Login.

6. Posteriormente el sistema proporciona retroalimentacion sobre el estado del login.

7. El sistema se cierra por completo al realizar la operación Logout.

8. Finalmente el sistema registra el eventro

T02.5.2 - Especificación de casos de uso - “CU010 - Logout”

Diagrama

Descripción generada automáticamente

T02.5.3 - Especificación de casos de uso - “CU010 - Logout”

|  |
| --- |
| **Id y Nombre:** CU10– Logout |
| **Objetivo:** El Usuario realiza el logout en el sistema. |
| **Actor Principal:**  Usuario |
| **Pre-Condiciones:**  El usuario debe estar logueado en el sistema.  Debe haber una sesion activa para el usuario. |
| **Puntos de Extension: -** |
| **Disparador:**  El usuario decide realizar el logout presionando el boton “Si” de la interfaz de confirmacion de Logout. |
| **Post Condiciones:** El usuario realiza el logout de forma exitosa en el sistema**.** |
| **Escenario Principal:**  1.El Usuario apreta el boton “Salir”  2. El sistema abre la GUI de confirmacion de logout.  3.El usuario selecciona la opcion “Si”.  4. El sistema verifica la sesion del usuario.  5. El sistema realiza el logout en el sistema.  6. El sistema se cierra por completo. |
| **Escenario Alternativo:**  3.1 El usuario se arrepiente y apreta el boton “No” llevandolo al menu principal. |

T02.5.4 - Diagrama de secuencia - “CU010 - Logout”

Diagrama

Descripción generada automáticamente

T02.5.5 - Diagrama de actividad - “CU010 – Logout”

N/A

T02.5.6 - Diagrama de clases - “CU010 - Logout”

Diagrama

Descripción generada automáticamente

T02.5.7 - Diagrama Entidad Relación (DER) - “CU010 - Logout”

N/A

T02.5.8 - GUI - “CU010 - Logout”

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

# T03. Encriptación

#### T03.1 - Irreversible

Se utilizará el método de encriptación irreversible basado en el uso del algoritmo de hash

SHA256 para que las contraseñas de los usuarios estén seguras. Este algoritmo toma una

cadena de texto (contraseña) y la convierte en un valor de longitud fija “Hash”.

Cuando el usuario ingresa la contraseña, se aplica el algoritmo SHA256 para generar un nuevo

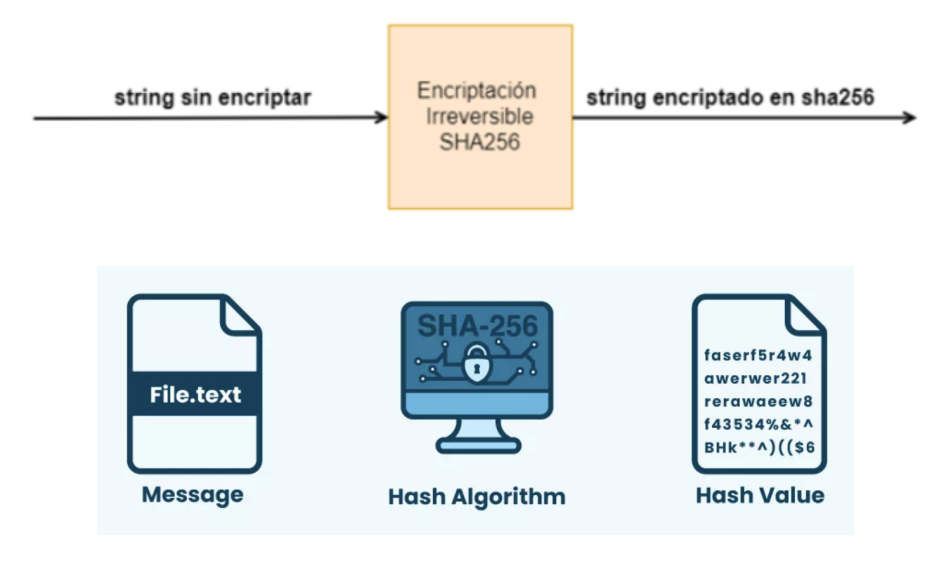
hash, y este lo almacena en la base de datos. Para que el usuario pueda ingresar, deben

coincidir el hash almacenado en la base de datos con el que se generó cuando el usuario inicia

sesión.

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente



#### T03.2 - Reversible (Explicación Campo Sensible BD - Caja Negra)

Se utilizará el método de encriptación reversible para ocultar los datos sensibles de la base de

datos. Se llevará a cabo utilizando el método de encriptación y desencriptación AES Advanced

Encryption Standard.

Este método AES permite la encriptación y desencriptación de datos utilizando una clave

secreta compartida. Esto convierte la descripción de un formato encriptado ilegible para

aquellos que no tengan la clave, para proteger la confidencialidad de la base de datos.

El proceso de desencriptación se basa en utilizar la clave secreta que se utilizó durante la

encriptación, permitiendo recuperar la información y mostrarla en un formato legible para los

usuarios autorizados.

La seguridad de esta encriptación depende de la seguridad de la clave utilizada para

desencriptar

Diagrama

Descripción generada automáticamente

Campo de la BD donde se va a aplicar:

Texto

Descripción generada automáticamente con confianza media

# T04 Crear Perfil

#### T04.1 Descomposicion funcional

1. Administrador ingresa al sistema
2. El sistema muestra el menú principal
3. Administrador selecciona la opción para gestionar perfiles
4. El sistema muestra el formulario de creación de perfil
5. Administrador introduce los datos del perfil
6. El sistema valida los datos introducidos
7. El administrador confirma la creación del perfil
8. El sistema guarda el perfil en la base de datos
9. El sistema muestra un mensaje de confirmación

T04.1.2 Diagrama de casos de uso

Diagrama

Descripción generada automáticamente

**T04.1.3 Especificacion de caso de uso**

|  |
| --- |
| **Id y Nombre:** CU13 – Crear Perfil |
| **Objetivo:** El administrador crea un perfil de usuario con un conjunto de permisos especificos. |
| **Actor Principal: Administrador** |
| **Pre-Condiciones:** El administrador debe tener permisos para gestionar los perfiles**.** |
| **Puntos de Extension:** CU011- Gestionar Familias |
| **Disparador:**  El administrador debe haber iniciado sesion en la plataforma |
| **Post Condiciones:** El nuevo perfil se ha creado exitosamente |
| **Escenario Principal:**   1. El usuario selecciona la opción "Gestionar Perfiles" en la interfaz de la plataforma. 2. El sistema muestra la pantalla de gestión de perfiles con una lista de perfiles   existentes, patentes disponibles y familias disponibles.   1. El usuario selecciona un perfil de la lista. 2. El usuario puede agregar una patente al perfil seleccionado. 3. El usuario puede quitar una patente del perfil seleccionado 4. El usuario puede agregar una familia al perfil seleccionado 5. El usuario puede quitar una familia del perfil seleccionado. 6. El usuario guarda los cambios realizados en el perfil. 7. El sistema actualiza el perfil con los cambios realizados y muestra un mensaje de confirmación. |
| **Escenario Alternativo:**  9.1 Ya existe un perfil con ese nombre por lo que el sistema arroja un mensaje de error. |

**T04.1.4 Diagrama de Secuencia CU013 Crear Perfil**

Diagrama

Descripción generada automáticamente

**T04.1.6 Diagrama de clases CU013 Crear Perfil**

**Diagrama

Descripción generada automáticamente**

Diagrama

Descripción generada automáticamenteT04.1.8 Diagrama de Actividad “ Gestionar Perfil”

**T04.5 Diagrama entidad relacion**

**Diagrama

Descripción generada automáticamente**

**T04.1.6 Interfaz De CU013**

**Gráfico, Gráfico en cascada

Descripción generada automáticamente**

# T04.2 – Crear Familias

**T04.2.1 - Descripción funcional**

1. El usuario selecciona la opción "Gestionar Familias" desde la pantalla de gestión de

perfiles.

2. El sistema muestra la pantalla de gestión de familias con una lista de familias

existentes, patentes disponibles y otras familias disponibles.

3. El usuario selecciona una familia de la lista.

4. El usuario puede agregar una patente a la familia seleccionada.

5. El usuario puede quitar una patente de la familia seleccionada.

6. El usuario puede agregar una familia a la familia seleccionada.

7. El usuario puede quitar una familia de la familia seleccionada.

8. El usuario guarda los cambios realizados en la familia.

9. El sistema actualiza la familia con los cambios realizados y muestra un mensaje de

confirmación.

**T04.2.2 – Especificación de caso de uso “Gestionar Familias”**

|  |
| --- |
| **Id y Nombre:** CU14 – Gestionar Familias |
| **Objetivo:** El Administrador es capaz de crear una familia de permisos, añadiendo permisos compuestos y simples a una familia. |
| **Actor Principal: Administrador** |
| **Pre-Condiciones:** El administrador debe tener permisos para gestionar las familias. |
| **Puntos de Extension: -** |
| **Disparador:**  El administrador debe haber iniciado sesion en la plataforma |
| **Post Condiciones:** La nueva familia se ha creado exitosamente con los permisos asignados |
| **Escenario Principal:**  1. El usuario selecciona la opción "Gestionar Perfiles" en la interfaz de la plataforma.  2. El sistema muestra la pantalla de gestión de perfiles con una lista de perfiles  existentes, patentes disponibles y familias disponibles.  3. El usuario selecciona un perfil de la lista.  4. El usuario puede agregar una patente al perfil seleccionado.  5. El usuario puede quitar una patente del perfil seleccionado  6. El usuario puede agregar una familia al perfil seleccionado  7. El usuario puede quitar una familia del perfil seleccionado.  8. El usuario guarda los cambios realizados en el perfil.  9. El sistema actualiza el perfil con los cambios realizados y muestra un mensaje de  confirmación. |
| **Escenario Alternativo:** |

**T04.2.4 Diagrama de Secuencia CU014 Gestionar Perfiles**

**Diagrama

Descripción generada automáticamente**

**T04.2.5 Diagrama de Clases CU014 GestionarFamilias**

Forma

Descripción generada automáticamente con confianza media

**T04.2.6 Diagrama de Entidad Relacion CU014 GestionarFamilias**

**Diagrama

Descripción generada automáticamente**

**T04.2.7 – GUI “Gestionar Familias”**

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

# T05 Idioma

**T05.1. Descomposición funcional**

1. El usuario ingresa al sistema
2. El sistema muestra el formulario de login.
3. El usuario ingresa al sistema introduciendo los datos como(username y contraseña).
4. El sistema valida los datos y abre el menu principal.
5. El usuario en el menu principal elige a traves de una combobox el idioma que desea para el usuario.
6. El sistema valida que el idioma elegido esta en la base de datos.
7. El sistema cambia el idioma de todo el sistema de dicho usuario y es guardado.

**T05.2.Diagrama de caso de uso**

**Diagrama

Descripción generada automáticamente**

**T05.3 Especificacion de caso de uso“Cambiar Idioma”**

|  |
| --- |
| **Id y Nombre:** CU12 – Cambiar Idioma |
| **Objetivo:** El usuario cambia el idioma de todo el sistema. |
| **Actor Principal:** Usuario |
| **Pre-Condiciones: -** |
| **Puntos de Extension: -** |
| **Disparador:**  El usuario cambia el idioma de todo el sistema apretando un boton en la interfaz del menu principal del sistema. |
| **Post Condiciones:** Se ha modificado el idioma de el sistema exitosamente |
| **Escenario Principal:**   1. El sistema muestra la pantalla de menu principal. 2. El sistema carga todos los botones predifinidos de la interfaz principal. 3. El usuario en el menu prinicipal elije el idioma del combobox y aprieta el boton cambiar idioma. 4. El sistema guarda la selección del usuario. 5. El sistema guarda la preferencia de idioma del usuario. 6. El sistema notifica a todos los formularios mediante el uso del patron observer que hubo un cambio en el atributo por lo tanto debe de cambiar la interfaz de usuario. 7. El sistema busca en la carpeta de recursos(.Json) la hoja de recursos con el idioma correspondiente. 8. El sistema trae la hoja especificada a los formularios y actualiza todos los controles mediante un sistema de clave-valor. 9. El sistema actualiza el idioma de los formularios. |
| **Escenario Alternativo:**  3.1 El usuario no aplica los cambios. |

**T05.4 Diagrama de Secuencia**

**Diagrama

Descripción generada automáticamente**

**T05.4 Diagrama de Clases**

**Diagrama

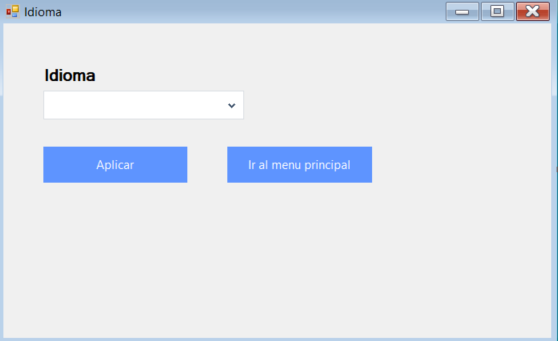
Descripción generada automáticamente**

**T05.5 Diagrama Entidad Relacion**

**Diagrama

Descripción generada automáticamente**

**T05.6 Interfaz de Cambio de idioma**

****

# (T06) Bitacora de Eventos

* La **bitácora de eventos** :
* El usuario con el perfil de administrador podrá ingresar a la sección de bitácora de eventos, la cual muestra en un DataGridView todos los eventos que han sucedido en el sistema.
* Cada registro de bitácora muestra: Login, fecha y hora, Evento, modulo, Descripción del evento, criticidad.
* El Administrador puede filtrar los resultados según sea su elección, rango de tiempo, Login, modulo, Nombre, apellido, criticidad.
* También el Administrador puede descargar el PDF con los eventos filtrados o con todos los eventos que aparezcan en grilla.

#### Diagrama de Secuencia de Gestion de Bitacora EVENTOS

Interfaz de usuario gráfica, Diagrama

Descripción generada automáticamente

#### Diagrama de Clases de Gestion de Bitacora EVENTOS

Diagrama

Descripción generada automáticamente

#### Diagrama de Entidad Relacion de Bitacora EVENTOS

**A screenshot of a computer

Description automatically generated**

### 

### Auditoria bitacora de eventos.

#### Descripcion funcional

**Registro Automático de Eventos:**

**Funcionalidad:** Cada vez que ocurre un evento en el sistema, este se registra automáticamente y de manera instantánea en la tabla de EVENTOS. Este registro incluye detalles clave como la fecha y hora del evento, el módulo donde ocurrió, la operación realizada, el usuario responsable, y el nivel de criticidad asignado.

**Objetivo:** Garantizar que todos los eventos sean capturados en tiempo real para una trazabilidad completa y efectiva del sistema.

**Asignación de Niveles de Criticidad:**

**Funcionalidad:** A cada evento registrado se le asigna un nivel de criticidad según su importancia. Los niveles de criticidad van del 1 al 5, donde 1 indica eventos de máxima importancia y 5 los de menor relevancia.

**Objetivo**: Priorizar la revisión de eventos críticos para una gestión más eficiente y proactiva del sistema.

**Filtrado de Información en la Interfaz:**

**Funcionalidad:** La interfaz de la bitácora permite a los usuarios filtrar los eventos registrados en función de los siguientes criterios:

**Fecha y Hora:** Permite seleccionar un rango de tiempo específico para la consulta de eventos.

**Usuario:** Filtra los eventos por el usuario que los realizó.

**Módulo:** Filtra por el módulo del sistema en el que ocurrió el evento (e.g., Usuarios, Ventas, Compras, etc.).

**Criticidad:** Filtra los eventos según su nivel de criticidad.

**Evento:** Filtra por el tipo de evento (e.g., Login, Logout, Crear Usuario, Cambiar Clave, etc.).

**Objetivo:** Facilitar la búsqueda y revisión de eventos específicos, proporcionando una herramienta poderosa para la auditoría y el análisis del sistema.

Tipificación de Eventos y Módulos:

**Funcionalidad:** Tanto los eventos como los módulos en los que estos ocurren se encuentran tipificados para facilitar la selección y el filtrado en la interfaz.

**Eventos Tipificados:** Login, Logout, Crear Usuario, Cambiar Clave, Registrar Cliente, Generar Carrito, etc.

**Módulos Tipificados:** Usuarios, Ventas, Compras, Maestro, Perfiles, etc.

**Objetivo:** Estandarizar la clasificación y registro de eventos, lo que permite una mejor organización y consulta de la información almacenada.

**Visualización del Nombre y Apellido del Usuario:**

**Funcionalidad:** Al seleccionar un registro en la lista de eventos (grilla) mediante el mouse, se muestran automáticamente en pantalla el nombre y apellido del usuario que realizó el evento. Por defecto, siempre se mostrará la información del primer registro de la lista.

**Objetivo:** Proveer detalles específicos del usuario relacionado con cada evento, mejorando la comprensión y análisis de las operaciones realizadas.

**Persistencia de Datos:**

**Funcionalidad:** Los datos de la bitácora son persistentes; una vez registrados, no pueden ser modificados ni eliminados. Esto garantiza la integridad y la seguridad de la información auditada.

**Objetivo:** Asegurar que toda la información registrada en la bitácora esté disponible para revisiones futuras, cumpliendo con los requisitos de auditoría y cumplimiento normativo.

**Ubicación en el Menú del Administrador:**

**Funcionalidad:** La bitácora de eventos está accesible desde el menú principal del administrador, brindando acceso rápido y fácil para la revisión y gestión de los eventos registrados en el sistema.

**Objetivo:** Facilitar el acceso a la bitácora para los usuarios con permisos de administración, asegurando una gestión eficiente del sistema.

#### Diagrama de Casos de uso

A diagram of a person with a diagram

Description automatically generated

(T06.1.2) Especificacion Caso de Uso Auditar Bitacora

|  |
| --- |
| Caso de uso CU 01 – Auditar Bitácora |
| Descripción:El Administrador ingresa a la sección de bitácora y observa todos los eventos registrados |
| Actor Principal: Administrador |
| Actor Secundario: |
| Precondición: El administrador debe estar logueado y debe tener el perfil de administrador para acceder a la bitácora. |
| Escenario Principal:  1-El Administrador entra al apartado de control de bitácora.  2- El Sistema busca los registros en la base de datos y los cargar en un DataTable.  3- El Sistema Carga los registros para poder verlos en un DataGridView.  4- El Administrador visualiza los registros completos. |
| Escenario alternativo:  4.1 El Administrador aplica filtros según sus preferencias, rango de tiempo, criticidad, usuario.    4.2 El sistema aplica los filtros y refresca la pantalla con los datos solicitados. |
| Postcondición: Se audita la bitácora de eventos. |

#### (T06.1.3) Diagrama de Secuencia CU01-Auditar Bitacora

**A diagram of a company

Description automatically generated**

#### (T06.1.4) Diagrama de Clases CU01-Auditar Bitacora

**A diagram of a computer

Description automatically generated**

#### (T06.1.4) Diagrama Entidad Relacion CU01-Auditar Bitacora

**A screenshot of a computer

Description automatically generated**

## Bitacora de Cambios.

**Registro de Cambios:**

#### Descripción Funcional:

1-EL sistema registra los movimientos o eventos que le suceden a la tabla de productos.

2-Cada Vez que un producto es modificado, creado o eliminado es registrado por un trigger en la tabla Productos\_C.

3-A su vez esos Registros se pueden visualizar en la grilla que muestra el sistema con todos los eventos que sucedieron a esa tabla Productos\_C.

4-El usuario puede filtrar por nombre, código y fecha los registros que se ven en dicha grilla.

5-El sistema cuenta con un botón para activar el producto en el caso de que la actividad de este sea 0 pasa a 1 o todos los que tenían 1 tienen que pasar a 0.

6-Los cambios de dicha actividad tiene que afectar e imparctar en la tabla Productos original.

Lo que se utilizo para esta bitácora fueron triggers:

Los triggers son un disparador que se activan según el evento que se le allá asignado, esto lo que hace es agarrar, Los campos de la tabla Productos y copiarlos en la tabla de Productos\_C.  
Los triggers se pueden utilizar ya sea con INSERT, UPDATE O DELETE.

Para poder registrar todos los cambios de los productos para un histórico.

#### Diagrama Actividad Bitacora de Cambios

Diagrama

Descripción generada automáticamente

### Auditoria bitacora de Cambios.

#### Descripcion funcional

**Inicio de Sesión:** Acceder al sistema y navegar al "Menú Maestro".

**Acceso a la Bitácora:** Seleccionar la "Bitácora de Cambios" para visualizar el historial.

**Filtrado de Información:** Aplicar filtros según los criterios deseados.

**Revisión de Registros Activos:** Identificar y revisar los registros activos.

**Actualizar Estado:** Modificar y guardar cambios en el estado de actividad.

**Impresión y Exportación:** Imprimir o exportar la bitácora de cambios según sea necesario.

**Registro de Cambios en la Tabla PRODUCTOS\_C:**

**Funcionalidad:** Cada vez que se realice un cambio en la tabla PRODUCTOS (ya sea una actualización, adición o eliminación), se generará automáticamente un registro en la tabla PRODUCTOS\_C. Este registro incluirá detalles como el código del producto (Cod\_Prod), la fecha y hora del cambio, el nombre del producto, la cantidad existente, la descripción, y un indicador de actividad (Act).

**Persistencia de Cambios:**

**Funcionalidad:** La tabla PRODUCTOS\_C será utilizada para persistir todos los cambios ocurridos en la tabla PRODUCTOS. Cada vez que se modifique un producto, se creará un nuevo registro en PRODUCTOS\_C, con el campo Act indicando si el registro es el más reciente (activo) o si es un registro histórico (inactivo).

**Identificación de Registros Activos:**

**Funcionalidad:** En la tabla PRODUCTOS\_C, un solo registro por cada producto será marcado como activo (Act = 1), representando la información más reciente y actualizada. Este mecanismo asegura que solo un registro esté marcado como activo para cada producto, mientras que los demás se mantendrán como históricos.

**Mantenimiento de la Tabla PRODUCTOS:**

**Funcionalidad:** La tabla PRODUCTOS solo almacenará información de productos activos. Cualquier cambio en la data de PRODUCTOS (agregación, modificación o eliminación) resultará en la creación de un nuevo registro en PRODUCTOS\_C con Act = 1.

**Construcción del Servicio con Triggers:**

**Funcionalidad:** Este servicio se construirá utilizando Triggers en la base de datos. Los Triggers garantizarán que cada cambio en PRODUCTOS se refleje inmediatamente en PRODUCTOS\_C, automatizando el proceso de registro de cambios.

**Ubicación en el Menú Maestro:**

**Funcionalidad:** La bitácora de cambios será accesible desde el menú Maestro del sistema, permitiendo a los administradores y usuarios autorizados acceder rápidamente al historial de cambios y realizar las auditorías necesarias.

#### Diagrama de Casos de uso

A diagram of a person with a diagram

Description automatically generated

#### Especificacion de Casos de uso

|  |  |
| --- | --- |
| *Caso de uso CU 01 – Auditar bitacora de Cambios* | |
| *Descripción:* | **El Administrador ingresa a la seccion de bitacora de historicos y observa todos los cambios registrados** |
| *Actor Principal:* | Administrador. |
| *Actor Secundario:* |  |
| *Precondiciones:* | **El Administrador debe de estar logeado en sistema. El Administrador debe de tener el perfil correspondiente para observar bitacora de cambios.** |
| *Escenario Principal:* | 1. **El Administrador ingresa a la sección de control de bitacora de historico.** 2. **El Administrador selecciona la tabla de la cual quiere ver su historico.** 3. **El sistema muestra en un datagrid view los registros completos de la tabla seleccionada con su historico y validando cual es el registro vigente.** 4. **El Administrador visualiza los registros completos.** |
| *Extensiones o Flujos Alternativos:* | **\*. Se cierra el sistema.**  **4.1 El Administrador realiza un rollback de un registro.**  **4.2 El Administrador selecciona un registro y acciona el boton del rollback.**  **4.3 El Sistema toma las llaves foreaneas del historico y realiza un update en la tabla original, dejando otro registro como vigente.** |
| *Postcondiciones:* | El Administrador restauro un registro correctamente. |

#### Diagrama de secuencia de CU 01 Auditar Bitacora Historico

Diagrama

Descripción generada automáticamente

#### Diagrama de clases de CU-01 Auditar Bitacora Historico

Diagrama

Descripción generada automáticamente

#### Diagrama Entidad Relacion Bitacora de Historicos

Interfaz de usuario gráfica, Diagrama, Texto, Aplicación

Descripción generada automáticamente

# T07 -Gestion de respaldo.

## Descripcion funcional

1. El administrador ingresa a la seccion de respaldos
2. El administrador inicia el proceso de backup, selecciona el directorio para generar el respaldo del sistema.
3. El administrador confirma la operacion, y el sistema guarda el respaldo correctamente.
4. El administrador inica el proceso de restauracion de base de datos, selecciona el directorio donde se encuentra el respaldo, confirma la restauracion del sistema.

## Casos de uso.

A diagram of a diagram

Description automatically generated

## Especificacion de CU.

|  |  |
| --- | --- |
| *Caso de uso CU 01 – Gestionar respaldo* | |
| *Descripción:* | **El Administrador ingresa a la seccion de respaldos y genera o restaura respaldos del sistema.** |
| *Actor Principal:* | Administrador. |
| *Actor Secundario:* |  |
| *Precondiciones:* | **El Administrador debe de estar logeado en sistema. El Administrador debe de tener el perfil correspondiente.** |
| *Escenario Principal:* | 1. **El Administrador ingresa a la sección de respaldos.** 2. **El sistema muestra dos grupos de controles, generacion de respaldos y restauracion de respaldos.** 3. **El administrador realiza la operacion deseada.** 4. **El adminsitrador confirma y aplica los cambios.** |
| *Extensiones o Flujos Alternativos:* | **\*. Se cierra el sistema.**  **4.A Administrador realiza Backup**   1. **El administrador selecciona el boton para elegir la ubicacion donde guardara el backup** 2. **El sistema muestra las carpetas del sistema con un openfileDialog** 3. **El administrador elige una ubicacion y confirma.** 4. **El sistema guarda la ubicion seleccionada en el un textbox** 5. **El administrador acciona la opcion de generar Backup.** 6. **El sistema genera el backup.**   **4.B Administrador realiza Restore**   1. **El administrador selecciona la ubicacion donde tiene su backup** 2. **El sistema guarda la ubicion seleccionada en el un textbox** 3. **El administrador acciona la opcion de generar realizar un restore del sistema.** 4. **El sistema realiza el restore.** |
| *Postcondiciones:* |  |

## Diagrama de secuencia.

**A diagram of a computer program

Description automatically generated**

## Diagrama de clases.

A screenshot of a computer

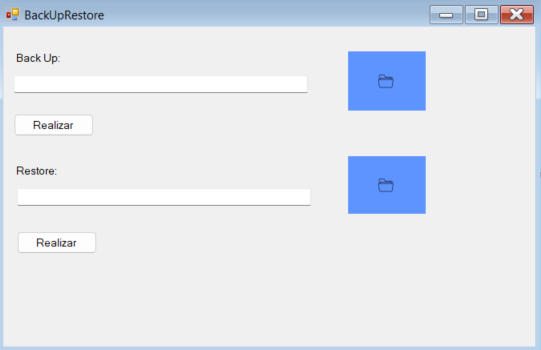
Description automatically generated

#### Diagrama de Entidad Relacion (CU01-Gestionar Respaldo)

Diagrama

Descripción generada automáticamente con confianza media

## GUI.



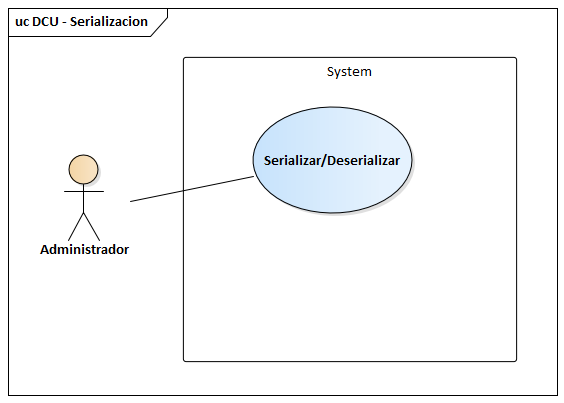
# A.03 Serializacion

#### (A03.1)Descripcion Funcional:

El objetivo es implementar una funcionalidad que permita serializar y deserializar datos de clientes a partir de una interfaz gráfica (GUI). Los datos de clientes se mostrarán en un *Grid* y se podrá realizar la serialización o deserialización de estos datos directamente desde allí.

1. Serializar Clientes:
   * El usuario selecciona la opción de serializar desde la pantalla principal. Los datos que están en el *Grid* (lista de clientes) se serializan en un formato específico (ejemplo: JSON, XML o binario) y se almacenan en un archivo en el sistema local.
2. Deserializar Clientes:
   * El usuario selecciona la opción de deserializar datos. El sistema toma un archivo serializado previamente (en el formato soportado) y lo deserializa para mostrar los datos en el *Grid*.
3. Limpiar Grid:
   * Opción para limpiar la lista de clientes del *Grid*. Esto será útil antes de realizar una operación de deserialización para evitar confusión con datos previamente cargados.

#### (A03.2)Diagrama de casos de uso:

****

#### (A03-1) Especificacion de caso de uso CU-01 Serializar-Desserializar

|  |  |
| --- | --- |
| *Caso de uso CU 01 – Serializacion-Desserializacion* | |
| *Descripción:* | **El Administrador ingresa a la seccion de ABM Clientes** |
| *Actor Principal:* | Administrador. |
| *Actor Secundario:* |  |
| *Precondiciones:* | **El Administrador debe de estar logeado en sistema. El Administrador debe de tener el perfil correspondiente.** |
| *Escenario Principal:* | 1. **El Administrador entra a la pantalla de Clientes** 2. **El sistema muestra do botones, serializar y des-serializar** 3. **El administrador realiza la operacion deseada.** 4. **El adminsitrador confirma y aplica los cambios.** |
| *Extensiones o Flujos Alternativos:* | **\*. Se cierra el sistema.**  **4.A Administrador realiza Serializar**  **5. El Administrador tilda la opcion JSON o XML para el tipo de archivo que elige para realizar la serializacion**  **6. El Administrador Aprieta el boton Serializar.**  **7. El sistema abre una pestaña de ruta de archivos.**  **8. El Administrador escribe el nombre del archivo que desea guardar**  **9. El sistema serializa la lista de clientes que figura en el DataGridView y lo guarda con la opcion tildada y el nombre de archivo que fue elegido por el admin.**  **10. El sistema muestra un mensaje de resultado sobre la operación realizada**  **4.B Administrador realiza Des-Serializacion**   1. **El Administrador Selecciona la opcion des-serializar.** 2. **El sistema abre el explorador de archivos para que el administrador selecciona el archivo que desea des-serialiar.** 3. **El Administrador selecciona el archivo que desea.** 4. **El sistema comprueba que el archivo es JSON-XML** 5. **El sistema des-serializa en el formato correspondiente** 6. **El sistema una vez des-serializado convierte todo a una datagrid view.** |
| *Postcondiciones:* |  |

#### (A03-2) Diagrama de secuencia CU-01 Serializar-Desserializar

**Diagrama

Descripción generada automáticamente**

#### (A03-3) Diagrama de clases CU01 Gestion Serializacion

Diagrama

Descripción generada automáticamente

#### (A03.4) Diagrama Entidad Relacion CU01 Gestion Serializacion

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Imagen que contiene Forma

Descripción generada automáticamente

#### (A03.5) GUI CU01 Gestion Serializacion

Interfaz de usuario gráfica

Descripción generada automáticamente

##### T08 Digito Verificador

#### T08.1 Descripcion Funcional:

* El proceso funciona al actualizar o persistir datos en las tablas del negocio. Se recalculan los digitos verificadores horizontales y verticales del sistema, los cuales se almacenan posteriormente en una tabla especifica.
* El calculo realizado del digito verificador horizontal se realiza concatenando todos los datos de cada fila y aplicando la funcion de encriptacion SHA-256. Este valor encriptado se concatena progresivamente hasta completar el proceso de todas las filas de la tabla. Al finalizar, se aplica nuevamente la funcion sobre los valores encriptados, obteniendo asi el digito verificador horizontal.
* El Calculo del digito verificador vertical se genera se genera concatenando todos los datos de cada columna de la tabla y aplicando la función SHA-256. Este valor encriptado se concatena de igual manera hasta que se completa el proceso para todas las columnas de la tabla. Al final, se aplica la misma función sobre estos valores encriptados, resultando en el dígito verificador vertical de la tabla.
* Al terminar de calcular los dígitos verificadores horizontales y verticales de las tablas, estos se concatenan y se aplica nuevamente la función SHA-256, de esta forma obteniendo el digito verificador horizontal y vertical global del sistema.

#### T08.1.2-Diagrama de Secuencia

Diagrama

Descripción generada automáticamente

#### T08.1.3-Diagrama de clases

Diagrama

Descripción generada automáticamente

#### T08.1.4-DER

Tabla

Descripción generada automáticamente

#### T08.2 Diagrama de casos de uso

#### 

Diagrama

Descripción generada automáticamente

#### T08.1.1 Especificacion de CU01 Login

|  |
| --- |
| **Id y Nombre:** CU01– Login |
| **Objetivo:** Se desea realizar una revision del digito verificador. |
| **Actor Principal:** Administrador |
| **Pre-Condiciones: -** |
| **Puntos de Extension: -** |
| **Disparador:**  El administrador presiona la opcion “ingresar” en la pantalla de incio de sesion. |
| **Post Condiciones:** El sistema validó que coincidan el digito verificador generado con el de la base de datos |
| **Escenario Principal:**   1. El usuario debe haber validado sus credenciales exitosamente. 2. El usuario presiona el botón “Ingresar” en la pantalla de inicio de sesión. 3. El sistema realiza el cálculo del digito verificador. 4. El sistema obtiene el digito verificador que se encuentra almacenado en la base de datos. 5. El sistema compara los dígitos verificadores. 6. El sistema permite finalizar el proceso de inicio de sesión. |
| **Escenario Alternativo:**  5.1 El sistema muestra una pantalla de error con el mensaje “El sistema no se encuentra disponible en estos momentos. Contacte al administrador.”  5.2 El sistema redirecciona a la pantalla de inicio de sesión. |

#### T08.1.2 Diagrama de secuencia de CU01 Login

Diagrama

Descripción generada automáticamente

#### T08.1.3 Diagrama de clases de CU01 Login

Diagrama

Descripción generada automáticamente

#### T08.1.4 DER de CU01 Login

Tabla

Descripción generada automáticamente

#### T08.2.1 Especificacion de CU02 Reparar Inconsistencia

|  |
| --- |
| **Id y Nombre:** CU02– Reparar Inconsistencia |
| **Objetivo:** Se desea realizar una revision del digito verificador. |
| **Actor Principal:** Usuario |
| **Pre-Condiciones: -** |
| **Puntos de Extension: -** |
| **Disparador:**  El administrador presiona la opcion “ingresar” en la pantalla de incio de sesion. |
| **Post Condiciones:** El sistema realiza el recalculo del digito verificador o realiza un restore. |
| **Escenario Principal:**   1. El Administrador debe haber validado sus credenciales exitosamente. 2. El Administrador presiona el botón “Ingresar” en la pantalla de inicio de sesión. 3. El sistema detecta que los dígitos verificadores generados con los de la base de datos no coinciden. 4. El sistema muestra una pantalla al administrador, dando la opción de recalcular el digito verificador o realizar un restore. 5. El administrador selecciona la opción deseada.   6. El sistema permite el ingreso al menú principal al administrador. |
| **Escenario Alternativo:**   * 5.1.1 El administrador selecciona la opción recalcular digito verificador * 5.1.2 El sistema realiza el recalculo del digito verificador y lo guarda en la base de datos * 5.1.3 El sistema redirecciona al punto 6. * 5.2.1 El administrador selecciona la opción de realizar restore. * 5.2.2 El sistema permite al administrador seleccionar un archivo .bak para realizar restore. * 5.2.3 El administrador selecciona archivo .bak * 5.2.4 El sistema realiza el restore de la base de datos * 5.2.5 El sistema redirecciona al punto 6. |

#### T08.2.2 Diagrama de secuencia del CU02 Reparar inconsistencia

Diagrama, Esquemático

Descripción generada automáticamente

#### T08.2.3 Diagrama de clases CU02 Reparar Inconsistencia

Diagrama

Descripción generada automáticamente

#### T08.2.4 DER CU02 Reparar Inconsistencia

Diagrama

Descripción generada automáticamente con confianza media