

Facultad de Tecnología Informática

Ingeniería en Sistemas Informáticos

**Trabajo de Campo I**



|  |  |  |  |
| --- | --- | --- | --- |
| **TRABAJO PRÁCTICO INTEGRADOR:** **Cheeselogix** | | | |
| **Docente** | Christian Chamula | | |
| **Alumno** | Santillán Menéndez Juan Cruz | | |
| **Sede/Comisión** | Centro – 3°J | **Turno:** | Noche |
| **Año:** | 2023 | **Versión** | 1.0 |

## 

## **ÍNDICE**

Contenido

[2](#_Toc175511494)

[**ÍNDICE** 2](#_Toc175511495)

[**HISTORIAL DE REVISIÓN** 3](#_Toc175511496)

[**G00. Descripción Global del Producto** 4](#_Toc175511501)

[**G01. Propósito** 4](#_Toc175511502)

[**G02. Descripción funcional del producto y Alcance** 5](#_Toc175511503)

[**G02.1. Descripción funcional del producto RFN1** 5](#_Toc175511504)

[**G02.2. Descripción funcional del producto RFN2** 5](#_Toc175511505)

[**G03. Definiciones, Acrónimos, y Abreviaciones** 6](#_Toc175511506)

[**G04. Descripción de las personas participantes (Roles)** 8](#_Toc175511507)

[**G05. Otros Requisitos** 9](#_Toc175511508)

[**G06. Diagrama de clases parcial de todos los módulos implementado** 11](#_Toc175511509)

[**G07. Modelo de datos parcial de todos los módulos implementados** 12](#_Toc175511510)

## **HISTORIAL DE REVISIÓN**

|  |  |  |  |
| --- | --- | --- | --- |
| Fecha | Versión | Autor | Descripción |
| 25/08/2024 | 1.0 | Juan Cruz Santillán | Desarrollo desde el G0 al G7 |
| 31/08/2024 | 1.1 | Juan Cruz Santillán | Planteamiento N01 |
| 07/09/2024 | 1.2 | Juan Cruz Santillán | Desarrollo N01 |
| 14/09/2024 | 1.3 | Juan Cruz Santillán | Desarrollo N01 |
| 21/09/2024 | 1.4 | Juan Cruz Santillán | Refinamiento N01 |
| 05/10/2024 | 1.5 | Juan Cruz Santillán | Desarrollo desde el T01 al T03 |
| 09/10/2024 | 1.6 | Juan Cruz Santillán | Refinamiento T01 a T03 |

## **G00. Descripción Global del Producto**

## **G01. Propósito**

La empresa **Embutearte** ha contratado a **Crosstech**Solutions para desarrollar un software que simplifique y automatice las operaciones clave en la distribución de lácteos y embutidos. **Crosstech**Solutions se especializa en soluciones tecnológicas diseñadas para mejorar la eficiencia operativa y la satisfacción del cliente, asegurando una gestión más efectiva de los recursos y operaciones comerciales.

Como parte de este proyecto, **Crosstech**Solutions ha identificado las siguientes áreas críticas en la cadena de valor de **Embutearte**:

* **Compras a proveedores**
* **Control de Inventario**
* **Ventas en tienda**

A partir de estas áreas, se definirán los requerimientos funcionales más importantes para el software.

## **G02. Descripción funcional del producto y Alcance**

## **G02.1. Descripción funcional del producto RFN1**

**RFN1. Gestión de compras de insumos:**

El sistema deberá gestionar el ciclo de vida de las compras de mercadería a proveedores, desde la generación de órdenes de compra hasta su recepción y actualización del inventario. Las funcionalidades incluirán la creación y seguimiento de órdenes de compra (pendiente, aprobada, abonada, recibida), facilitando un control eficiente de la mercadería recibida y su correcta asignación en los sectores correspondientes del almacén.

Este proceso no solo optimizará la organización de productos lácteos y embutidos, sino que también permitirá un seguimiento preciso de la disponibilidad de stock, mejorando la eficiencia operativa y garantizando un control detallado sobre el inventario de la distribuidora.

## **G02.2. Descripción funcional del producto RFN2**

**RFN2. Gestión de ventas en tienda:**

El sistema gestionará eficientemente el proceso de ventas en la tienda. Permitirá la verificación de stock en tiempo real antes de confirmar una venta, asegurando que solo se vendan productos disponibles. Cada transacción se registrará en el sistema, generando automáticamente una factura que incluirá los detalles del cliente y los productos vendidos. Tras cada venta, el inventario se actualizará, reflejando los movimientos de stock.

## **G03. Definiciones, Acrónimos, y Abreviaciones**

* **Unidad de Producto:** Un único artículo que forma parte del inventario de mercadería.
* **Caja de Producto:** Contenedor que agrupa una cantidad específica de unidades de producto, facilitando su transporte y almacenamiento.
* **Tipo de Producto:** Clasificación del producto según su naturaleza, ya sea embutido o lácteo.
* **Número** **de Etiqueta (NE):** Código único impreso en cada Caja/Unidad de Producto para su identificación y trazabilidad en el sistema.
* **Sectores (Zonas de Almacenamiento):** División del almacén en áreas específicas para organizar la mercadería según categorías, como temperatura, tipo de producto, o fecha de vencimiento.
* **Almacén:** Espacio físico destinado al almacenamiento de la mercadería, organizado en sectores para facilitar la gestión del stock.
* **Vencimiento (Fecha de Vencimiento):** Fecha límite hasta la cual un producto es apto para el consumo, registrada y controlada en el sistema para asegurar la frescura de los productos.
* **Lote:** Conjunto de unidades de producto fabricadas en las mismas condiciones y en el mismo período de tiempo, identificado por un número de lote.
* **Factura:** Documento legal que detalla la transacción comercial entre el proveedor y el comprador.
* **Orden de Compra:** Documento emitido por el comprador que solicita la adquisición de productos o servicios al proveedor.

|  |  |  |
| --- | --- | --- |
| PERSONAL | Descripción | RESPONSABILIDAD |
| Alex Turner | VENDEDOR | Es responsable de atender a los clientes en el punto de venta, asesorarlos en la selección de productos, y garantizar una experiencia de compra satisfactoria. Sus tareas incluyen llenar el carrito de compras y seleccionar los productos según los pedidos de los clientes. |
| Delfina de Vigan | CAJERO | Se encarga de procesar las ventas en el punto de venta, lo que incluye generar las facturas y cobrar los montos correspondientes a cada transacción, asegurando un manejo adecuado del efectivo y otros medios de pago. |
| Monk Luffy | ENCARGADO DE DEPOSITO | Es responsable de recibir los productos que llegan al almacén, verificando su estado y organizándolos adecuadamente. Además, se encarga de la entrega de productos a los clientes. |
| Gonzalo Martinez | GERENTE | Dirige las operaciones generales de la distribuidora, toma decisiones estratégicas y asegura el cumplimiento de los objetivos de negocio. Sus responsabilidades incluyen la selección de proveedores, el registro de nuevos proveedores en el sistema y la incorporación de nuevos productos al inventario. |
| Marcelo Barovero | ENCARGADO DE COMPRAS | Es la persona responsable de solicitar los productos y materiales necesarios para la distribuidora. Se encarga de negociar con los proveedores, hacer seguimiento a los pedidos y asegurarse de que se cumplan los plazos de entrega. |

**G04. Descripción de las personas participantes (Roles)**

|  |  |  |
| --- | --- | --- |
| PERSONAL | ROL | ACCESO |
| Alex Turner | VENDEDOR | RFN2 |
| Delfina de Vigan | CAJERO | RFN2 |
| Monk Luffy | ENCARGADO DE DEPOSITO | RFN1  RFN2 |
| Marcelo Barovero | ENCARGADO DE COMPRAS | RFN1 |
| Gonzalo Martinez | GERENTE | RFN1 |
| Eduardo Sacheri | CLIENTE | RFN2 |
| Marilina Bertoldi | ANALISTA FUNCIONAL | RFN1  RFN2 |
| Gustavo Balvorín | ADMINISTRADOR DE SISTEMA | RFN1  RFN2 |
| Martin Rolle | ANALISTA PROGRAMADOR | RFN1  RFN2 |

## **G05. Otros Requisitos**

* **Del Producto:**
* **Estándares Aplicables:**
  + El sistema debe tener un **menú horizontal** en la parte superior, donde se listarán las principales funcionalidades, como la gestión de compras, ventas, inventario, y proveedores.
  + **Logo de la distribuidora** como fondo en la pantalla principal, proporcionando una identidad visual consistente con la marca.
  + Una **barra inferior** mostrará información relevante como el nombre del usuario, el área de trabajo actual (compras, ventas, etc.), y la fecha y hora actuales.
* **Requisitos de Desempeño:**
  + El sistema debe asegurar un **tiempo de respuesta rápido**, especialmente en funciones críticas como la actualización de stock y generación de facturas.
  + **Escalabilidad**: Debe poder manejar un creciente número de transacciones y datos sin comprometer el rendimiento.
  + **Fiabilidad**: El sistema debe ser robusto y capaz de operar sin fallos que interrumpan las operaciones diarias.
  + **Capacidad de procesamiento** adecuada para manejar múltiples usuarios y operaciones simultáneas sin deterioro en el rendimiento.
* **Requisitos de Sistema:**
  + Debe contar con **capacidad de recuperación en caso de fallos**.
  + **Eficiencia en la utilización de recursos**.
  + El **hardware debe cumplir con los requisitos mínimos** necesarios para ejecutar el software de manera eficiente y fluida.
  + El software debe ser **compatible con la infraestructura de red existente** y ser capaz de integrarse con otros sistemas y dispositivos utilizados en la distribuidora.
* **Requisitos de Entorno:**
  + **Compatibilidad con el sistema operativo Windows**, asegurando que el software funcione correctamente en las versiones soportadas.
  + El sistema debe contar con una **conexión de red confiable** y una velocidad de conexión adecuada para garantizar un rendimiento óptimo del software.
* **De Documentación:**
* **Manual del Usuario:** Debe ser una guía de uso clara y detallada del software, diseñada específicamente para los usuarios de la distribuidora. El manual debe ser fácil de entender y contener información detallada sobre cómo utilizar todas las funciones del software, incluyendo la gestión de compras, ventas, inventario, y facturación.
* **Ayuda en Línea:** Dentro del menú, en el botón de ayuda, se podrá acceder a una sección de asistencia en línea. Al hacer clic en esta opción, los usuarios podrán encontrar respuestas a sus preguntas y solucionar problemas de forma autónoma. La ayuda en línea debe estar organizada por módulos y funcionalidades para facilitar la búsqueda de información.
* **Guía de Instalación y Configuración:** En formato PDF, se proporcionará una guía con instrucciones claras y detalladas sobre cómo instalar y configurar el software correctamente. Esta guía debe cubrir desde la instalación inicial hasta la configuración de las distintas funcionalidades del sistema.
* **Archivo LEEME:** Un documento de texto que contenga información importante sobre el sistema, incluyendo una descripción general del software, requisitos del sistema, y una guía básica de instalación y funcionamiento.

## **G06. Diagrama de clases parcial de todos los módulos implementado**

**Diagrama general de clases del negocio**

Diagrama

Descripción generada automáticamente

**Diagrama general de clases de la parte técnica y servicios**

Diagrama

Descripción generada automáticamente

**Diagrama de clases por capas - Negocio**

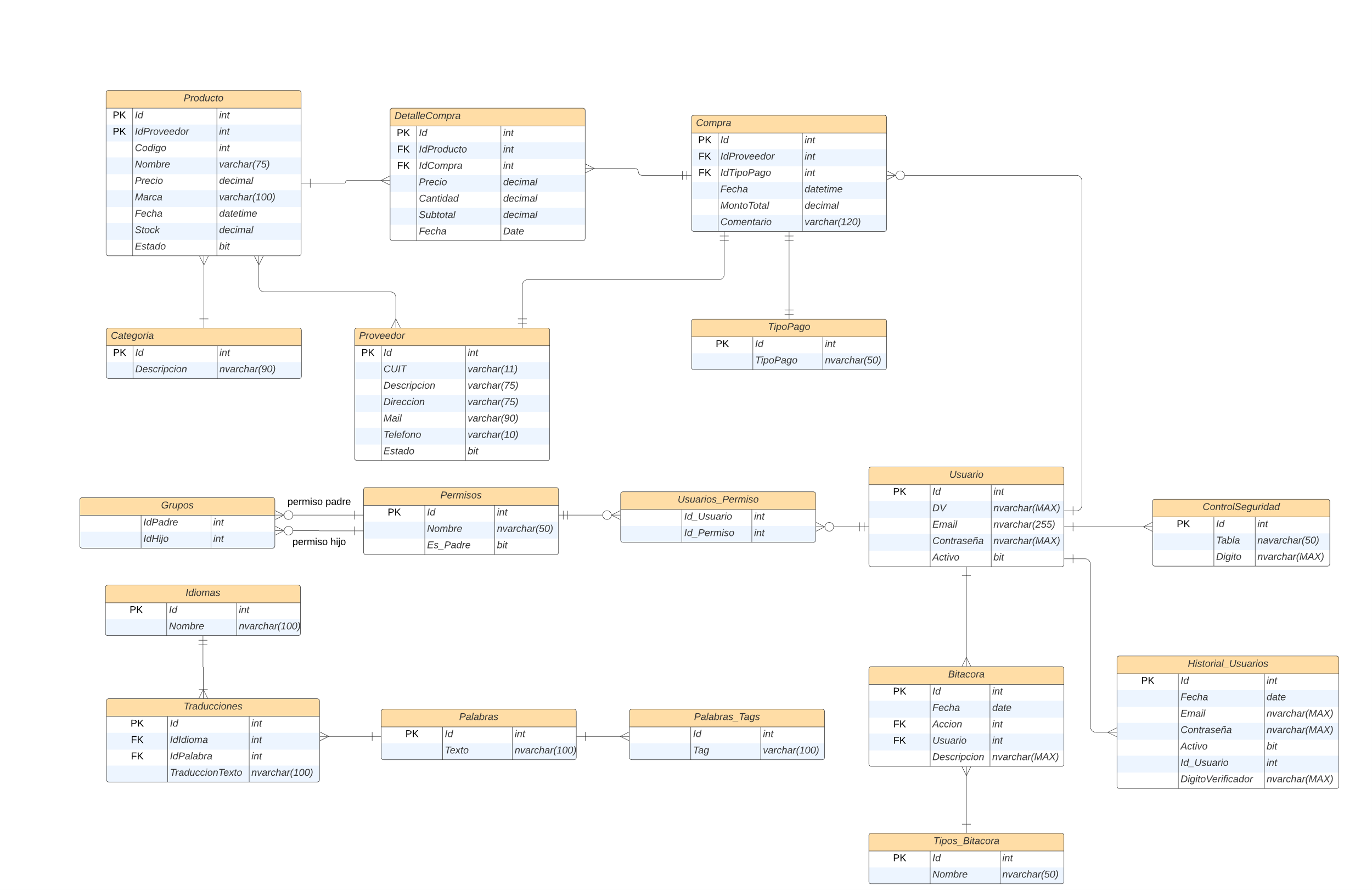
Diagrama

Descripción generada automáticamente

**Diagrama de clases por capas - Tecnica**

Diagrama, Esquemático

Descripción generada automáticamente



## **G07. Modelo de datos parcial de todos los módulos implementados**

## **G07. Modelo de datos parcial de todos los módulos implementados**

**N01. Especificación funcional por proceso de negocio**

**Proceso de Negocio: Gestión de compras de productos**

**0. DETALLE**

1. El encargado de compras ingresará a la pestaña Compras y seleccionará la opción realizar compra. En esta sección, podrá generar una orden de compra. Seleccionará la categoría del producto que desea adquirir (Fiambres, Quesos, Encurtidos, Almacén). Podrá buscar el producto por nombre, seleccionar su marca y proveedor, y visualizar el precio, que podrá modificar si es necesario. También tendrá la opción de ingresar nuevos productos junto con su respectiva marca, precio y proveedor, si no se encuentran precargados en el sistema.
2. Una vez seleccionado el producto, se deberá especificar la cantidad deseada (expresada en Kg o unidades) y el método de pago (Efectivo, Crédito, Débito, Transferencia o Mercado Pago). Después de completar estos campos, el encargado deberá presionar el botón “Agregar a Lista” para añadir el producto a la lista de compras.
3. El encargado de compras podrá seguir agregando productos de un mismo proveedor a la lista o quitar elementos según sea necesario. Una vez finalizada la lista, podrá generar la orden de compra mediante el botón “Generar Orden de Compra”. Esto creará un detalle asociado a esa orden y se registrarán las características más relevantes: detalle de la orden, total, método de pago, fecha, proveedor y estado de la orden.
4. En la opción Control, el Gerente tendrá la posibilidad de acceder a Evaluar Órdenes de Compra. El gerente es responsable de aprobar o rechazar las solicitudes de alta de productos y proveedores, así como las órdenes de compra, cambiando su estado a aprobada o rechazada.
5. Si la orden de compra incluye un producto o proveedor que no está registrado en el sistema, se generará una alerta. Al ingresar la compra, el sistema permitirá al encargado de compras registrar los nuevos productos o proveedores.
6. Si la orden de compra ha sido aprobada, el encargado de compras podrá visualizarla en una lista de Órdenes pendientes. Mediante el botón “Realizar Pago”, podrá efectuar el pago y cambiar el estado de la orden a “Abonada”.
7. Una vez que el pedido llegue, el encargado de depósito (repositor) revisará el ingreso de los productos. Indicará en el sistema si el pedido llegó en orden o si hubo algún problema, y podrá dejar comentarios en caso de que algo no esté bien.
8. **Identificación de roles intervinientes:**

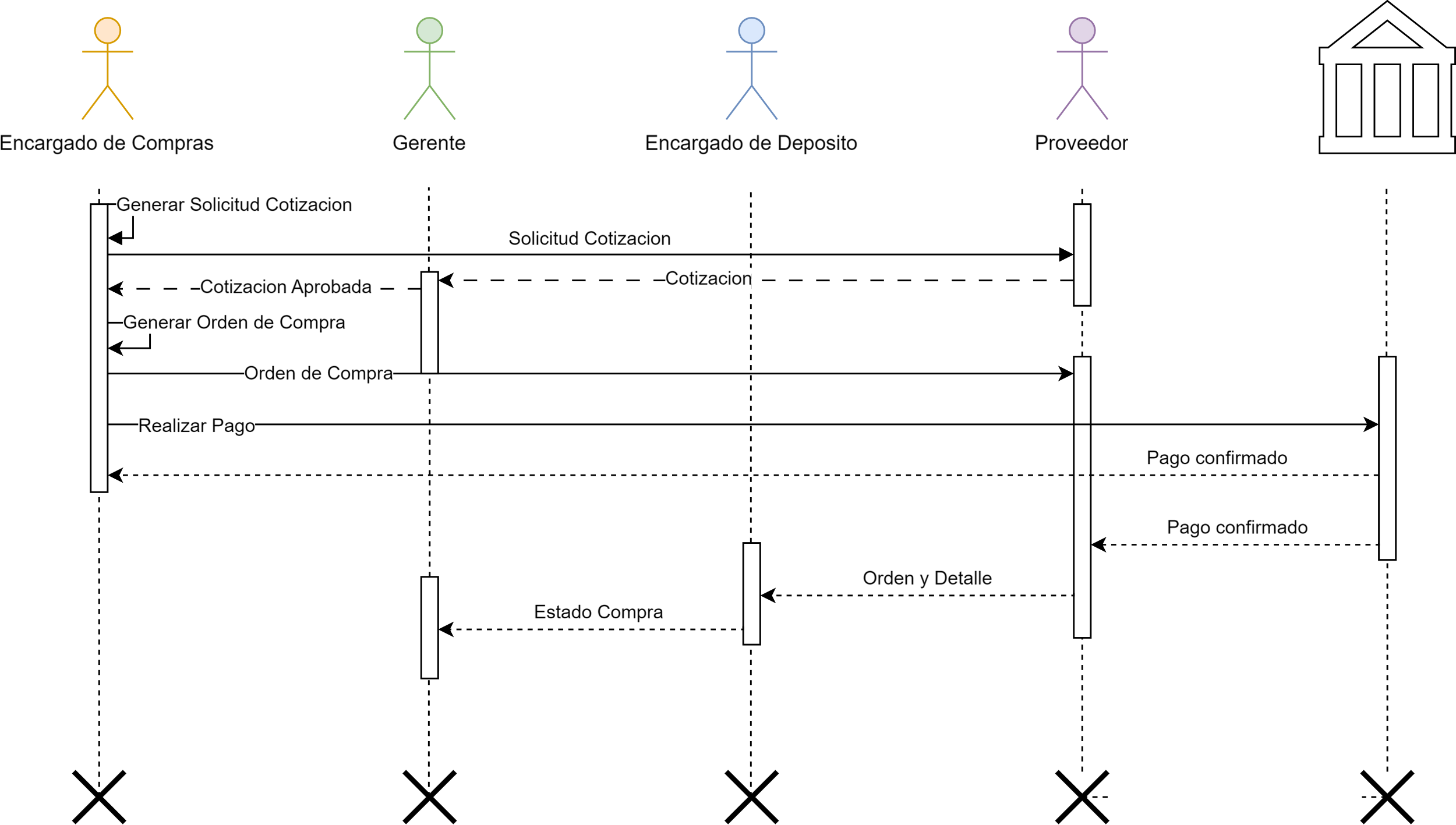
* Vendedor (Persona – no es actor directo – no usa GUI - Solicitante)
* Gerente (Persona – Primario – Usa GUI – Aprobador)
* Encargado de Compras (Persona – Primario – Usa GUI)
* Encargado de Deposito (Persona – Secundario – Usa GUI)
* Banco (Sistema- Se conecta con el sistema de compras - Remoto)
* Proveedor (Persona – no es actor directo – no usa GUI - Solicitante)

**Diagrama de Roles**

Diagrama

Descripción generada automáticamente

**Diagrama de Secuencia de Roles**



1. **Descripción Funcional del Proceso:**

Diagrama

Descripción generada automáticamente

1. **Diagrama de Proceso**

Diagrama

Descripción generada automáticamente

1. **Modelo Conceptual**

Diagrama

Descripción generada automáticamente

**N02.1. Especificaciones de Casos de Uso**

1. **Proceso de Negocio: Gestión de compras de productos**

**DIAGRAMA DE CASOS DE USO**

Diagrama

Descripción generada automáticamente

1. **Especificación CU01 – Generar Orden de Cotización**

|  |
| --- |
| **ID y Nombre**: CU01 – Generar Orden de Cotización |
| **Objetivo**: Detectar productos sin stock o con stock bajo y generar una solicitud de cotización |
|
| **Actor principal**: - Encargado de Compras |
| **Precondiciones**: El usuario debe estar logueado y con los permisos necesarios. |
|
| **Punto de extensión**: - |
| **Disparador**: Se detectan producto/s que deben reponer stock. |
| **Postcondiciones**: Se genera una solicitud de cotización. |
| **Escenario principal**:   1. El usuario ingresa al área de gestión de compras, Generar Solicitud de cotización. Se despliega una grilla con todos los productos y su stock **(Nombre del producto, categoría, stock).** 2. Se ingresa un filtro y se actualizan los datos **(Filtro por nombre, categoría o cantidad de stock)** 3. El usuario realiza el análisis de los productos y selecciona aquellos deseados para realizar una cotización **(Nombre del producto, stock)** 4. Se ingresan en una segunda grilla donde muestra los **productos seleccionados, su stock actual, y una casilla para ingresar la cantidad de productos solicitado** 5. El usuario ingresa la cantidad que se desea solicitar de cada producto **(Cantidad)** 6. El usuario selecciona “Generar Cotización” 7. El sistema genera un documento con la información del negocio, los productos solicitados y la cantidad. 8. Se genera un documento que se le otorga al usuario, conteniendo la información correspondiente a la cotización |
| **Flujos alternativos**:  2.1- No se ingresa ningún filtro o el filtro ingresado no devuelve datos.  3.1- El usuario realiza el análisis de existencia de los productos y decide no realizar una cotización |

Diagrama de secuencia General Solicitud de Cotización

Diagrama

Descripción generada automáticamente

Diagrama de clases Generar Orden de Cotización

Texto

Descripción generada automáticamente

1. **Especificación CU02 – Generar Orden de Compra**

|  |
| --- |
| **ID y Nombre**: **CU02** – Generar Orden de Compra |
| **Objetivo**: Generar la orden de compra para enviar al proveedor |
|
| **Actor principal**: - Encargado de Compras |
| **Precondiciones**: Se analizó la cotización previa y se selecciona generar orden de compra. Usuario logueado y con los permisos necesarios. |
|
| **Punto de extensión**: - CU03 Agregar Proveedor |
| **Disparador**: Se desea generar la orden de compra. |
| **Postcondiciones**: Se envía una orden de compra al proveedor. |
| **Escenario principal**:   1. El usuario selecciona “Generar Orden de Compra” 2. El usuario ingresa los datos de la cotización dentro de la orden de compra **(Datos del proveedor, datos del cliente, detalles de los productos)** 3. El sistema verifica que la cotización sea válida **(Que el proveedor esté registrado en el sistema, que los productos sean válidos)** 4. Se muestra un documento formal con la orden de compra. En donde aparece: **La información de la empresa, la fecha, el número de orden de compra, información del proveedor, descripción de los productos, precios unitarios, cantidad, costo total, impuestos y descuentos, fecha de entrega estimada, información del envío e instrucciones especiales.** 5. El usuario acepta la orden de compra y se genera un documento de **Orden de Compra** con la información proporcionada. 6. El sistema guarda un registro en la base de datos para una nueva compra con el estado de “Pendiente” en la base de datos |
| **Flujos alternativos**:  2.1- No existe el proveedor en el sistema y debe ingresarse – Extend – CU03 Agregar Proveedor  5.1- El usuario rechaza la orden de compra |

Diagrama de secuencia General Generar Orden de Compra

Diagrama

Descripción generada automáticamente

Diagrama de clases Generar Orden de Compra

Diagrama

Descripción generada automáticamente

CU03 – Agregar Proveedor

|  |
| --- |
| **ID y Nombre**: CU03 – Agregar Proveedor |
| **Objetivo**: Generar la orden de compra para enviar al proveedor |
|
| **Actor principal**: - Encargado de Compras |
| **Precondiciones**: Se debe haber generado la cotización. |
| **Punto de extensión**: - |
| **Disparador**: Se desea ingresar un proveedor en el sistema |
| **Postcondiciones**: Se ingresa el proveedor en el sistema. |
| **Escenario principal**:   1. Se ingresa a la pantalla de gestión de Usuarios -> Gestionar Proveedores 2. Se muestra una grilla con todos los proveedores y casillas para agregar un nuevo proveedor **(Documento, Mail, teléfono, Ubicación, Estado)** 3. El usuario ingresa los datos y selecciona aceptar **(Documento, Mail, teléfono, Ubicación, Estado)** 4. El sistema comprueba que los datos ingresados tienen un formato válido **(Documento, Mail, teléfono, Ubicación, Estado)** 5. Se despliega un mensaje confirmando el ingreso de un nuevo proveedor y se actualiza la grilla |
| **Flujos alternativos**:  4.1- Los datos no tienen un formato valido y se notifica al usuario  5.1- Hubo un problema ingresando un nuevo proveedor y se notifica al usuario |

Diagrama de Secuencia Agregar Proveedor

Diagrama

Descripción generada automáticamente

Diagrama de Clases Agregar Proveedor

Diagrama

Descripción generada automáticamente

CU2 – Realizar Pago

|  |
| --- |
| **ID y Nombre**: CU02 – Realizar Pago |
| **Objetivo**: Si la orden de compra se encuentra aprobada, El encargado de compras podrá visualizarla en una lista de Ordenes pendientes, y mediante el botón “Realizar Pago”, hacer efectivo el pago y cambiar el estado a finalizada. |
|
| **Actor principal**: - Encargado de Compras |
| **Precondiciones**: Se debe contar con una orden de compra aprobada por el gerente |
|
| **Punto de extensión**: - |
| **Disparador**: El Gerente aprobó una orden de compra |
| **Postcondiciones**: Recibe un comprobante de pago |
| **Escenario principal**:   1. El encargado de compras dentro de la pantalla “Compras” vera una lista de   órdenes y sus estados, deberá seleccionar la compra que desea abonar.   1. El sistema verificara que el estado sea “aprobado” y habilitara el botón   “Realizar pago”.   1. Para concretar la compra el encargado lo hará mediante el botón “realizar   pago”.   1. Ingresará los datos bancarios correspondientes al medio de pago previamente seleccionado y mediante el botón confirmar enviará el monto a una entidad bancaria. 2. El sistema verificara si es posible realizar el pago 3. El estado de la compra cambia de pendiente a abonado. |
| **Flujos alternativos**:  5.1- Mensaje de error por fondos insuficientes.  6.1- Mensaje de error falló la transacción. |

Diagrama de clases – Realizar pago

Diagrama

Descripción generada automáticamente

Diagrama de Secuencia CU2 – Realizar Pago

Diagrama

Descripción generada automáticamente

CU05 – Registrar Recepción

|  |
| --- |
| **ID y Nombre**: CU05 – Verificación Productos |
| **Objetivo**: Verificar los productos de una orden de compra |
|
| **Actor principal**: - Encargado de deposito |
| **Precondiciones**: Se debe haber generado una orden de compra, usuario logueado en el sistema con los permisos necesarios |
|
| **Punto de extensión**: - |
| **Disparador**: Se llega a la fecha de entrega de productos y se debe recibir la compra |
| **Postcondiciones**: Productos ingresados en el sistema. |
| **Escenario principal**:   1. El usuario accede al sistema y selecciona la opción Productos -> "Compras ". 2. El sistema despliega una lista de las órdenes de compra pendientes de recepción, mostrando los detalles de la orden de compra como fecha, proveedor, productos solicitados y cantidades. 3. El usuario selecciona la orden de compra que corresponde a los productos recibidos. 4. El sistema carga automáticamente los detalles de la orden de compra, incluyendo los productos solicitados y sus cantidades. 5. El usuario verifica visualmente los productos físicos recibidos y los compara con la información de la orden de compra. 6. Los productos coinciden con lo solicitado y el usuario procede a confirmar la recepción. 7. El sistema actualiza la compra como finalizada. El sistema actualiza productos y stock |
| **Flujos alternativos**:  5.1- si el usuario identifica alguna discrepancia entre los productos físicos y la orden de compra, puede registrar estas discrepancias en el sistema y cambiar la cantidad de productos recibida. |

Diagrama de secuencia Verificación Productos

Diagrama

Descripción generada automáticamente

Diagrama de clases – Registrar recepción.

Diagrama

Descripción generada automáticamente

CU06 – Evaluar Solicitud de cotización

|  |
| --- |
| **ID y Nombre**: CU06 – Evaluar Solicitudes de cotización |
| **Objetivo**: El Gerente podrá evaluar Solicitudes de cotización |
|
| **Actor principal**: - Gerente |
| **Precondiciones**: Debe haber Solicitudes de cotización con estado pendiente. |
|
| **Punto de extensión**: CU07 – Evaluar Altas |
| **Disparador**: Se detectan producto/s que deben reponer stock. |
| **Postcondiciones**: El Gerente necesita aprobar una solicitud de cotización. |
| **Escenario principal**:   1. El Gerente seleccionara “Evaluar Solicitudes de Cotización” dentro de las opciones de control. 2. Dentro de la pantalla Evaluación de Solicitudes de Cotización, verá un listado con las Solicitudes de Cotización y podrá filtrarlas por los estados “pendiente” o “finalizado”. 3. Cuando selecciona una Solicitud de Cotización Pendiente verá en otro listado el detalle de la solicitud (“Producto”, “Marca”, “Precio”, “Cantidad”). El sistema verificara que la solicitud se encuentre pendiente y habilitara los botones “Aprobar” y “Rechazar”. 4. El sistema verificara que el proveedor y producto se encuentren entre los proveedores y productos, y que mantenga su precio con estado “Aprobado”. 5. El gerente Aprueba la Solicitud de Cotización y el sistema cambia el estado a Aprobada, el sistema actualiza la lista quitando la Solicitud de Cotización. 6. El Gerente recibe un mensaje de “Cotización Aprobada” |
| **Flujos alternativos**:  3.1.1 Si selecciona una orden de compra con estado “Finalizado” 🡪 CU07  4.1.2 Si la Solicitud de Cotizacion se encuentra pendiente y tiene un Proveedor rechazado se alertará mediante una notificación.  4.1.3 Si la Solicitud de Cotizacion se encuentra pendiente y tiene un nuevo producto se alertará mediante una notificación.  4.1.4 Si la Solicitudes de Cotizacion se encuentra pendiente y tiene un producto rechazado se alertará mediante una notificación. |

Diagrama de clases – Evaluar Solicitud de Cotizacion

Diagrama

Descripción generada automáticamente

Diagrama de secuencia – CU06 Evaluar Solicitud de cotización

Gráfico

Descripción generada automáticamente con confianza media

CU07 – Evaluar Altas

|  |
| --- |
| **ID y Nombre**: CU07 – Evaluar Altas |
| **Objetivo**: El Gerente evalúa el estado de Solicitudes de cotización Finalizadas pudiendo registrar nuevos proveedores o productos |
|
| **Actor principal**: - Gerente |
| **Precondiciones**: Debe haber solicitudes de Solicitudes de cotización con estado Finalizado.. |
|
| **Punto de extensión**: |
| **Disparador**: El Gerente desea registrar un nuevo proveedor o producto nuevo. |
| **Postcondiciones**: El Gerente necesita aprobar una solicitud de cotización. |
| **Escenario principal**:   1. El Gerente filtra las Solicitudes de cotización por estado “Finalizado” 2. Cuando selecciona una Solicitudes de cotización verá en otro listado el detalle de la compra (“Producto”, “Marca”, “Precio”, “Cantidad”). El sistema verificara que el proveedor y producto no se encuentren entre los productos y proveedores con estado “Aprobado”. 3. Si la orden de compra se encuentra Finalizada y tiene un nuevo Proveedor se alertará mediante una notificación indicando el estado de entrega y se habilitaran los botones “Registrar Proveedor” y “Rechazar Proveedor”. Si la solicitud tiene un nuevo Producto se alertará mediante una notificación y se habilitaran los botones “Registrar Producto” y “Rechazar Producto”. 4. El Gerente selecciona la opción de “Registrar”. 5. El sistema cambia el estado del Producto a Proveedor de “Pendiente” a “Aprobado” |
| **Flujos alternativos**:  4.1 El Gerente selecciona la opción de “Rechazar”.  5.1 El sistema cambia el estado del Producto a Proveedor de “Pendiente” a “Rechazado” |

Diagrama de clases – Evaluar Altas

Diagrama

Descripción generada automáticamente

Diagrama de secuencia – CU07 Evaluar Altas

Diagrama, Esquemático

Descripción generada automáticamente

Diagrama de componentes Diagrama

Descripción generada automáticamente

## **T00. DOCUMENTOS DE ASPECTOS TÉCNICOS QUE PROVEE EL SISTEMA DE INFORMACIÓN**

1. **Domain (Dominio - BE)**

* **Descripción**: La capa de Dominio o Entidades (**BE - Business Entities**) contiene las clases que representan los objetos de negocio en la aplicación. Estos objetos reflejan las estructuras y los conceptos que se manejan en la lógica del negocio.
* **Responsabilidades**:
  + Definir la estructura de los datos de negocio a través de clases y objetos que representan las entidades principales.
  + Actuar como contenedores de datos sin incluir lógica de negocio compleja, pero pueden contener métodos de validación básica o transformación de datos.

1. **UI (Interfaz de Usuario)**

* **Descripción**: La capa de UI se encarga de la interacción con el usuario. Aquí se encuentran los formularios y componentes visuales que permiten a los usuarios interactuar con la aplicación.
* **Responsabilidades**:
  + Presentar datos de manera visual y amigable para el usuario.
  + Capturar las entradas del usuario y enviarlas a las capas adecuadas para su procesamiento.
  + Actualizar la interfaz con los resultados o mensajes relevantes provenientes de las capas de negocio.

1. **Aplicacion (Capa de Lógica de Negocio - BLL)**

* **Descripción**: Esta capa contiene la Lógica de Negocio (BLL - Business Logic Layer) de la aplicación. Es la responsable de procesar datos, aplicar reglas de negocio, y coordinar las interacciones entre las capas de Entidades, la UI y la capa de Servicios.
* **Responsabilidades**:
  + Implementar reglas de negocio y procesos específicos de la aplicación.
  + Validar y transformar datos antes de enviarlos o almacenarlos.
  + Coordinar las interacciones entre la UI, las entidades (BE) y los servicios, asegurando que las operaciones se ejecuten de manera adecuada.

1. **Mapper (Mapeador - MPP)**

* **Descripción:** La capa de Mapper se encarga de transformar y mapear los datos entre las entidades de negocio (BE) y las estructuras utilizadas en las capas superiores de la aplicación, como la capa de lógica de negocios (BLL) o la capa de presentación (UI). Facilita la transferencia de información entre estas capas de manera estructurada y coherente.
* **Responsabilidades:**
  + Convertir y adaptar los datos entre las entidades de negocio (Idioma, Permiso, etc.) y los objetos que manejan las capas de la aplicación, asegurando que los datos se adapten correctamente para su procesamiento en cada capa.
  + Ejecutar operaciones de ABML (Agregar, Modificar, Borrar, Listar) en la base de datos utilizando procedimientos almacenados y parámetros específicos asegurando la integridad y consistencia de los datos en las diferentes operaciones.
  + Manejar la transformación de resultados de consultas de bases de datos (DataTable a listas de objetos) para que sean utilizados por otras capas de la aplicación de manera eficiente y sin errores.

1. **Servicios (Internos y Externos)**

* **Descripción:** La capa de Servicios proporciona funcionalidades específicas y reutilizables que apoyan a otras partes de la aplicación. Estos servicios pueden ser internos, encapsulando lógica común y utilitaria, o externos, gestionando la comunicación con servicios externos o bases de datos.
* **Responsabilidades:**
* **Servicios Internos:**
  + Encapsular lógica común o funcionalidades reutilizables como el hashing de contraseñas y la generación de dígitos verificadores, asegurando la seguridad y consistencia en la validación de entidades (CalcularDigitoVerificadorHorizontal).
  + Implementar métodos utilitarios para manipulación y validación de datos, como en el caso de la clase Seguridad, que ofrece servicios de autenticación y cálculo de integridad de datos.
* **Servicios Externos:**
  + Manejar la conexión y la interacción con la base de datos (por ejemplo, con la clase Conexion que gestiona operaciones de lectura y escritura usando procedimientos almacenados). Garantizar que estas interacciones sean seguras, eficientes y transaccionales, asegurando el rollback en caso de errores.
  + Proveer métodos para la lectura y escritura de datos en la base de datos, transformando consultas y resultados en objetos y estructuras que puedan ser utilizados por las capas superiores de la aplicación.

DS de persistencia de datos

Diagrama

Descripción generada automáticamente

DS de Consulta de datos

Diagrama

Descripción generada automáticamente

T02.1 Gestión de Log In / Log Out del Sistema

**Descomposición Funcional Login:**

1. Para acceder al sistema, El usuario ingresa un nombre de usuario y una contraseña.
2. El sistema verifica que el usuario y contraseña ingresados sean correctos y se encuentren en la base de datos. Si los datos son incorrectos, se mostrará un mensaje de error y se dará la opción de volver a ingresar las credenciales.
3. Si un usuario ingresa información incorrecta tres veces seguidas, su cuenta quedará bloqueada.
4. Si un usuario está bloqueado, no podrá iniciar sesión incluso si ingresa la información correcta.
5. Si un usuario ingresa correctamente su información de inicio de sesión (Y no está bloqueado), se inicia sesión y el contador de intentos se reiniciará a cero.
6. Se mostrarán solo las funcionalidades a las que tengan acceso.

**Diagrama de caso de uso Login:**

Imagen que contiene Gráfico de burbujas

Descripción generada automáticamente

**Especificación funcional Login:**

|  |
| --- |
| **ID y Nombre: CDU01** – Iniciar Sesión |
| **Objetivo:** El usuario necesitaIngresar al sistema para operar el negocio, |
| **Actor principal:** Empleado |  |
| **Precondiciones:** Usuario desbloqueado.No debe haber una sesión ya iniciada |  |
| **Punto de extensión: CDU02 – Cambiar Idioma** |  |
| **Disparador:** El usuario necesita realizar una venta o una compra en el sistema. |  |
| **Postcondiciones:** El usuario se ingresa al sistema y tiene acceso a las actividades que le corresponden. |  |
|  |
| **Escenario principal:**   1. Se muestra la pantalla de Login, donde el usuario ingresa sus datos (Email y Contraseña). 2. Al hacer clic en ingresar, Se comprueba que el usuario no esté ya logueado 3. Se encripta la contraseña y se validan las credenciales con las de la base de datos. 4. Los datos del usuario son correctos. 5. Se inicia sesión en el sistema, se muestra la pantalla principal con los accesos correspondientes según su perfil 6. El sistema realiza la gestión del digito verificador, y si es correcto, se concede el acceso al sistema |  |
|  |
|  |
|  |
|  |
|  |
|  |
| **Flujos alternativos:**   * 1. El usuario decide cambiar de idioma – Extends CDU02 – Cambiar idioma   3.1- Hay un error de conexión con la base de datos. Se informa por pantalla que hubo un error  3.2- Hay un error en las credenciales ingresadas o el usuario no existe, se notifica.  3.3- En caso de que se ingrese un usuario válido con la contraseña correcta pero dicho usuario esté bloqueado, se le notificará al usuario que debe ponerse en contacto con el administrador y se regresará al punto dos.  3.4- Si se ingresan las credenciales correctas y el usuario no está bloqueado, pero ya hay una sesión abierta, se le comunicará al usuario y se regresará al punto dos.  7.1- Si se detecta una discrepancia en alguna tabla que es administrada por el dígito verificador, el sistema denegará el acceso a todos los usuarios y únicamente permitirá que el administrador ingrese para resolver el problema. |  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |

1. Diagrama de clases

Diagrama

Descripción generada automáticamente

1. DER

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

1. Secuencia

Diagrama, Esquemático

Descripción generada automáticamente

Diagrama de actividad LogIn

Diagrama

Descripción generada automáticamente

# T02.5 Gestión de Logout

**Descomposición funcional del Logout**

1. Para cerrar sesión, el usuario hará clic en el botón “Cerrar sesión” ubicado en el panel lateral
2. El sistema muestra una ventana la cual contiene un botón de cancelar y otro de aceptar
3. El usuario confirma cerrar sesión.
4. Se muestra la pantalla de Login

Diagrama de casos de uso: 

**Especificación funcional CDU03 – Realizar Logout:**

|  |
| --- |
| **ID y Nombre: CDU03** – Realizar Logout |
| **Objetivo:** Cerrar sesión en el sistema |
| **Actor principal:** Usuario |  |
| **Precondiciones** El usuario debe estar autenticado en el sistema. |  |
| **Punto de extensión: -** |  |
| **Disparador:** El usuario debe salir del sistema |  |
| **Postcondiciones:** Se cierra sesión y se borra caché |  |
| **Escenario principal:**   1. El sistema muestra la ventana de confirmación para cerrar sesión. 2. El usuario presiona el botón cerrar sesión 3. Se muestra la GUI de cerrar sesión la cual tiene el botón de cancelar y aceptar 4. Al presionar aceptar, se cierra sesión y se borra cache 5. Se eliminan todos los datos de la sesión 6. Se muestra la pantalla de Login |  |
|  |
|  |
|  |
| **Flujos alternativos:**  3.1. Se presiona cancelar y se continua con la sesión |  |

Diagrama de secuencia LogOut

Diagrama, Dibujo de ingeniería

Descripción generada automáticamente

Diagrama de actividad:

Diagrama

Descripción generada automáticamente

T03. Gestión de Encriptado

**Objetivo**

El principal objetivo de la encriptación en nuestro sistema es asegurar la confidencialidad y la integridad de las contraseñas de los usuarios. Para ello, utilizamos el algoritmo SHA-256, que es considerablemente más seguro que MD5 debido a su robustez y resistencia. Esta medida de seguridad es esencial para proteger las contraseñas almacenadas en la base de datos y garantizar la seguridad de las credenciales de acceso de nuestros usuarios.

**Descripción detallada del funcionamiento**

El proceso de encriptación es crucial y se implementa en dos momentos específicos dentro de nuestro sistema:

1. Durante el registro de un nuevo usuario:

* Cuando un nuevo usuario se registra, elige una contraseña que se encripta antes de ser almacenada en la base de datos. Este proceso comienza en la capa de lógica de negocio (BLL\_USUARIO), donde la contraseña proporcionada por el usuario es pasada al método Seguridad.HashPassword. Este método utiliza SHA-256 para convertir la contraseña en texto plano a una forma encriptada y segura.
* La contraseña encriptada se guarda entonces en la base de datos, asegurando que ninguna contraseña en texto plano sea almacenada o transmitida más allá de este punto.

1. Durante el inicio de sesión de un usuario existente:

* Al iniciar sesión, la contraseña ingresada por el usuario se encripta nuevamente utilizando el mismo método SHA-256 para asegurar que la comparación con la contraseña almacenada en la base de datos sea precisa y segura.
* Esta verificación se realiza dentro del método LogearConCredencialesde BLL\_USUARIO, donde se compara la contraseña cifrada proporcionada con la versión cifrada almacenada en la base de datos. Si coinciden, el acceso es concedido, y el usuario puede entrar al sistema.

**Implementación adicional:**

Para reforzar la seguridad, todas las acciones relacionadas con el inicio-cierre de sesión se registran en la bitácora del sistema utilizando BLL\_BITACORA. Esto incluye la creación de registros de usuario y los intentos de inicio de sesión, proporcionando una pista de análisis que puede ser utilizada para revisar actividades sospechosas o para diagnósticos de seguridad

Diagrama de Clases

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Diagrama de secuencia

Diagrama

Descripción generada automáticamente

T04. Gestión de Perfiles

**Crear Perfil**

1. **Descripción Funcional**

1. El usuario con perfil de Administrador ingresa a la pestaña Administración del menú superior o de la barra lateral y dentro de la lista desplegable ingresa a Gestión de Permisos.

2. Dentro de la pantalla gestión de permisos, podrá visualizar a través de un treeview los permisos y los grupos de permisos ya existentes, podrá ir seleccionando los diferentes grupos de permisos a través de un combobox para visualizarlos tambien.

3. El usuario podrá crear un nuevo grupo de permisos a través de ingresar el nombre en el textBox y luego clickear en el botón Crear Grupo, luego deberá limpiar el árbol, seleccionar el permiso en el comboBox ‘permisos’, agregarlo por primera vez y asociarle un permiso individual o un grupo de permisos.

4. Una vez generado el grupo deseado el usuario administrador podrá asignar o desasignar grupos seleccionados de un combobox a usuarios seleccionados a través de otro combobox.

**Diagrama de Casos de Uso**

Diagrama

Descripción generada automáticamente con confianza media

**Especificación de Casos de Uso CU040 – Gestión de permisos**

|  |
| --- |
| **ID y Nombre: CDU040** – Gestión de Permisos |
| **Objetivo:** Se agrega una familia (grupo) y se le asignan patentes (permisos) |
| **Actor principal:** Administrador |  |
| **Precondiciones:** Se debe tener permiso de administrador y la familia no debe existir. |  |
| **Punto de extensión: -** |  |
| **Disparador:** Se necesita crear un nuevo grupo de permisos. |  |
| **Postcondiciones:** El usuario se ingresa al sistema y tiene acceso a las actividades que le corresponden. |  |
|  |
| **Escenario principal:**  1. El usuario ingresa a “Gestión de Permisos”.  2. Completa el TextBox nombre con el nombre que desea asignarle al nuevo grupo de Permisos y selecciona crear Grupo  3. El sistema verifica que el nombre esté disponible y si lo está lo persiste en la base de datos y actualiza el comboBox con todos los grupos  4. Luego se “Limpia el Árbol” y se le asigna al nuevo grupo creado sus permisos correspondientes.  4. Cuando se cargan nuevamente las familias, se genera un listado de objetos composite, de los que heredan las familias y se le agregan como composite hijo las patentes relacionadas.  6. Cuando un grupo es agregado, el sistema persiste en una tabla en la base de datos el id de grupo y el id de permiso asignados, y actualiza. En el treeview correspondiente según el grupo seleccionado en el combobox se visualizará los permisos asignados.  7. Una vez finalizada la gestión el usuario deberá cerrar la pantalla |  |
|  |
|  |
|  |
|  |
|  |
|  |
| **Flujos alternativos:**  3.2 No se pudo agregar por error de persistencia  **PostCondiciones:**  Se generó un nuevo grupo de permisos |  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |

**DER Permisos:**

Diagrama

Descripción generada automáticamente

**Clases Permisos:**

Diagrama

Descripción generada automáticamente

### **T04a. Alta**

#### a. **Objetivo**

El objetivo de la gestión de alta de usuarios es permitir la creación segura y eficaz de nuevas cuentas de usuario dentro del sistema. Este proceso no solo incluye la captura y el almacenamiento de datos del usuario, sino que también asegura que estos datos sean validados y cifrados adecuadamente para mantener la seguridad y la integridad del sistema.

#### b. **Descripción detallada de cómo funciona**

El proceso de alta de usuarios comienza cuando un usuario potencial llena un formulario proporcionando su correo electrónico y contraseña.

1. **Validación de entrada:**

Inicialmente, el sistema verifica que el correo electrónico y la contraseña cumplen con los criterios especificados, como el formato correcto del correo electrónico. Esta validación ayuda a prevenir errores comunes y aumenta la seguridad del usuario.

1. **Encriptación de Contraseña:**

Utilizando la clase Seguridad, la contraseña ingresada por el usuario se encripta utilizando el algoritmo SHA-256. Este paso es crucial para asegurar que las contraseñas almacenadas en la base de datos no sean legibles ni accesibles en su forma original.

1. **Verificación de Unicidad:**

Antes de proceder con el almacenamiento, el sistema, a través del MPP\_USUARIO, verifica que no existe ya un usuario con el mismo correo electrónico. Esta comprobación asegura la unicidad de las cuentas y evita duplicaciones.

1. **Registro en la Base de Datos:**

Si la verificación es exitosa y no se encuentran duplicados, el MPP\_USUARIO procede a registrar el nuevo usuario en la base de datos. Este paso incluye pasar las propiedades del usuario, como el correo electrónico y la contraseña cifrada, a la función correspondiente de la capa de acceso a datos ( DAL), que finalmente guarda los datos en la base de datos.

1. **Confirmación:**

Una vez completado el registro, el sistema devuelve una confirmación al usuario indicando que la cuenta ha sido creada exitosamente. Además, se registra este evento en el sistema de bitácora a través de MPP\_BITACORApara auditorías y seguimiento de actividades.

c. Diagrama de Clases (PARA TODOS EN EL ABM)

Diagrama

Descripción generada automáticamente

d. Diagrama de Entidad Relación

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

e. Diagrama de Secuencia

Diagrama

Descripción generada automáticamente

### **T05. Gestión de Múltiples Idiomas**

### **Objetivo**

El principal objetivo de la gestión de múltiples idiomas en nuestro sistema es proporcionar una interfaz de usuario adaptable a diferentes idiomas, mejorando así la accesibilidad y la experiencia del usuario. Esto permite a los usuarios seleccionar su idioma preferido y ver toda la interfaz en dicho idioma, facilitando su interacción con la aplicación.

### **Descripción detallada del funcionamiento**

La gestión de múltiples idiomas se implementa a través de varios componentes clave dentro del sistema, que incluyen la capa de presentación (formularios), la capa de lógica de negocio (BLL), y la capa de acceso a datos (MPP). Aquí se explica cómo se realiza esta gestión paso a paso:

1. **Definición y Almacenamiento de Idiomas y Traducciones:**

**Tablas y Procedimientos Almacenados:**

* Idiomas almacena los diferentes idiomas disponibles con su respectivo identificador y nombre.
* Palabras almacena todas las palabras y frases que se pueden traducir, junto con un identificador único.
* Traducciones almacena las traducciones de cada palabra para cada idioma, relacionando los identificadores de Idiomas y Palabras con el texto traducido.

**Procedimientos Almacenados:**

* Guardar\_Traduccion: Este procedimiento se encarga de insertar o actualizar una traducción en la tabla Traducciones. Primero, verifica si la palabra ya existe en la tabla Palabras. Si no existe, la inserta y luego guarda la traducción.
* ListarTraduccionesPorIdioma: Este procedimiento recupera todas las traducciones para un idioma específico, uniendo las tablas Traducciones y Palabras para obtener el texto de la palabra y su traducción.
* Borrar\_Traduccion: Este procedimiento elimina una traducción específica de la tabla Traducciones.

1. **Carga de Idiomas y Traducciones:**

**Formulario de Configuración de Idiomas:**

En el formulario ConfigurarIdioma, se cargan todos los idiomas disponibles y se muestran en un ComboBox. El usuario puede seleccionar un idioma para visualizar y gestionar sus traducciones.

Las traducciones existentes para el idioma seleccionado se muestran en un DataGridView. Los usuarios pueden agregar, modificar o eliminar traducciones directamente desde esta interfaz.

Al agregar o modificar una traducción, se utiliza la capa de lógica de negocio (BLL\_TRADUCCION) que interactúa con la capa de acceso a datos (MPP\_TRADUCCION) para actualizar la base de datos.

1. **Actualización de la Interfaz:**

**Patrón Observer:**

El patrón Observer se implementa a través de SessionManager, que notifica a todos los formularios observadores cuando se cambia el idioma.

Los formularios registran observadores a SessionManager y actualizan sus textos cuando se selecciona un nuevo idioma. Este mecanismo asegura que cualquier cambio de idioma se propague inmediatamente a todas las interfaces de usuario abiertas.

**Selección de Idioma:**

Al seleccionar un idioma en el ComboBox de idiomas, se notifica a SessionManager que cambia el idioma actual. SessionManager entonces llama al método Actualizar en todos los observadores registrados.

El método Actualizar recorre todos los controles del formulario y cambia su texto según las traducciones disponibles para el idioma seleccionado, utilizando los tags definidos en cada control para identificar las palabras a traducir.

1. **Implementación adicional:**

**Registro en la Bitácora:**

Todas las acciones relacionadas con la gestión de idiomas y traducciones se registran en la bitácora del sistema utilizando BLL\_BITACORA. Esto incluye la creación, modificación y eliminación de traducciones, así como los cambios de idioma. Este registro proporciona una pista de auditoría que puede ser utilizada para revisar cambios y actividades sospechosas.

### **Diagrama de Clases**

Diagrama

Descripción generada automáticamente

1. **Diagrama de Entidad Relación**

Diagrama

Descripción generada automáticamente

1. **Diagrama de Secuencia**

Diagrama

Descripción generada automáticamente