

**Proyecto:**

“Sistema Integral de Gestión de Locales Informáticos.”

**Carrera:** Tecnicatura Universitaria en Programación

**Cátedra: Metodología de sistemas**

**Profesor:**

* Magariños, Luis

**Integrantes:**

* Ermeti Carlos

AÑO 2022

Índice:

1. Introducción: 1

1.1 Presentación del proyecto 2

1.2 Análisis del sistema 3

1.3 Objetivos del sistema 4

1.4 Límites 5

1.5 Descripción conceptual del proyecto 6

1. Requerimientos: 7

2.1 Requerimientos funcionales 8

2.2 Consultas e informes 9

2.3 Requerimientos no funcionales 10

1. Diagrama de Gantt 11

3.1 Diagramas de caso de uso 12

3.2 Especificaciones de casos de uso 13

1. Diseño del sistema 14

4.1

4.2 Simbología utilizada 15

4.3 Procesos 16

4.3.1 Proceso Login de Usuario 17

4.3.2 Diagrama de FlowChart 18

4.3.3 Rutinas 19

4.3.4 Pantallas 20

4.3.5 Proceso Gestion de Empleado 21

4.3.6 Diagrama de FlowChart 22

4.3.7 Rutinas 22 4.3.8 Pantallas 23

4.4.1 Proceso gestión de Cliente 24

4.4.2 Diagrama de FlowChart 25

4.4.3 Rutinas 26

4.4.4 Pantallas 27

4.5.1 Proceso gestión de Soporte 28

4.4.2 Diagrama de FlowChart 29

4.4.3 Rutinas 30

4.4.4 Pantallas 31

4.5.1 Proceso gestión de ventas 32

4.5.2 Diagrama de FlowChart 33

4.5.3 Rutinas 34

4.5.4 Pantallas 35

4.6.1 Proceso gestión de proveedores 36

4.6.2 Diagrama de FlowChart 37

4.6.3 Rutinas 38

4.6.4 Pantallas 39

4.7.1 Proceso gestión de productos 40

4.7.2 Diagrama de FlowChart 41

4.7.3 Rutinas 42

4.7.4 Pantallas 43

4.8.1 Proceso gestión de compras 44

4.8.2 Diagrama de FlowChart 45

4.8.3 Rutinas 46

4.8.4 Pantallas 47

1. Esquema de base de datos : 48

5.1.1 Esquema de base de datos 49

5.2.1 Diccionario de datos 50

Introducción:

Presentación del Proyecto

Proyecto: Sistema Integral de Gestión de Locales Informáticos.

El siguiente proyecto se realizó en la empresa llamada Soft Planet, que se dedica al servicio técnico de equipos informáticos y distribución de equipamiento, tanto software como hardware para puntos de venta, de la ciudad de Concordia, Entre Ríos.

El cliente necesita unificar los procesos de su actividad comercial a través de un sistema integral el cual se encargue de almacenar los artículos que trabaja, su stock y venta, proyección de implementaciones, servicios y un seguimiento del servicio tanto editable por el área técnica como así también de solo lectura para el cliente.

* Análisis del Sistema:

Objetivos

Mejorar la gestión y eficiencia de la compra, venta, almacenamiento de mercadería  y servicio técnico de equipos a clientes consumidores finales.

Límites del sistema

Los límites establecidos serán desde la compra o soporte pedido por un cliente hasta la facturación.

Descripción Conceptual del Dominio

Para comenzar con el servicio de Soft Planet, el cliente puede solicitar tanto soporte referente a equipos informáticos como también la compra de uno o más productos.

También Herramienta computacional que nos permita registrar tanto  elementos para la venta(Stock) cómo realizar el seguimiento de servicios técnicos brindados.

Requerimientos:

Requerimientos Funcionales: Procesamiento

1- Gestión de soporte técnico de equipos.

2- Gestión de ventas

3- Gestión de compras

4- Carga de empleados

5- Carga de clientes

6- Carga de proveedores

7- Carga de productos

Consulta e informes:

1- Generación de reportes de venta

2- Informe de estado de equipo en soporte

3- Emisión de comprobantes de compra

4- Emisión de comprobantes de venta

5- Listado de productos con quiebre de stock o mínimo

6- Generación historial de clientes

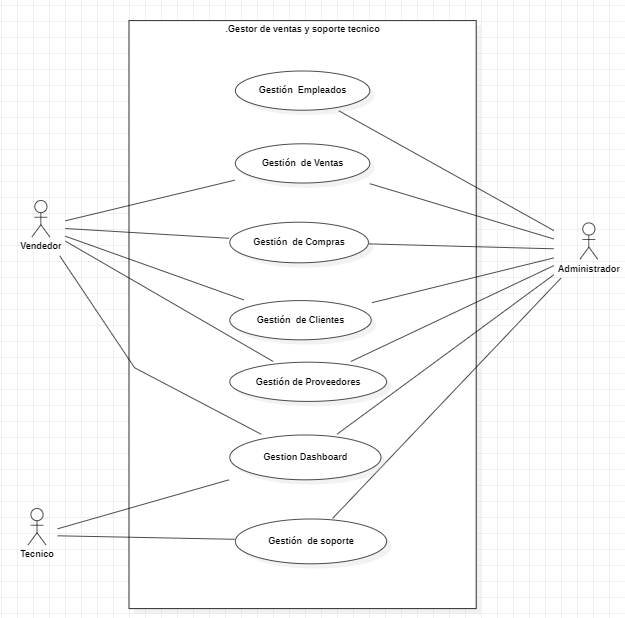
Requerimientos No Funcionales:

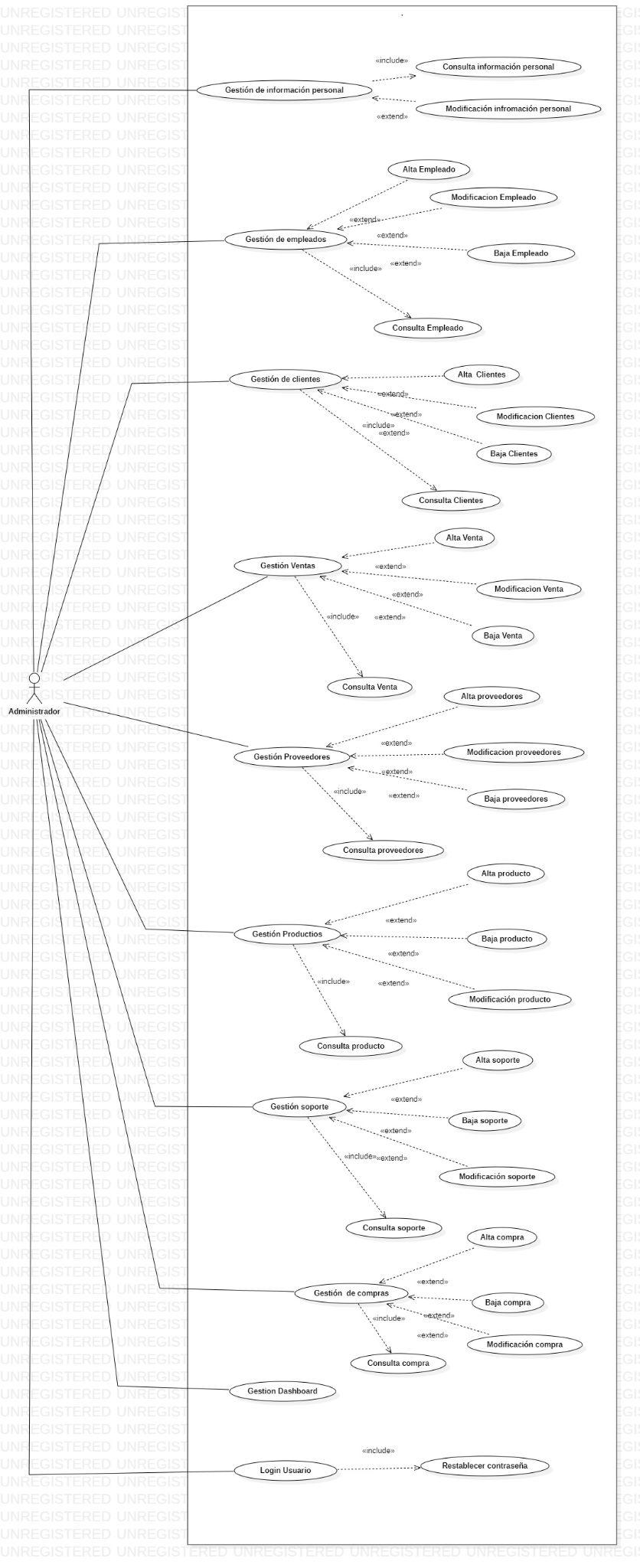
* Seguridad.
* Accesibilidad.
* Usabilidad.
* Estabilidad.
* Portabilidad.
* Escalabilidad.
* Mantenibilidad.

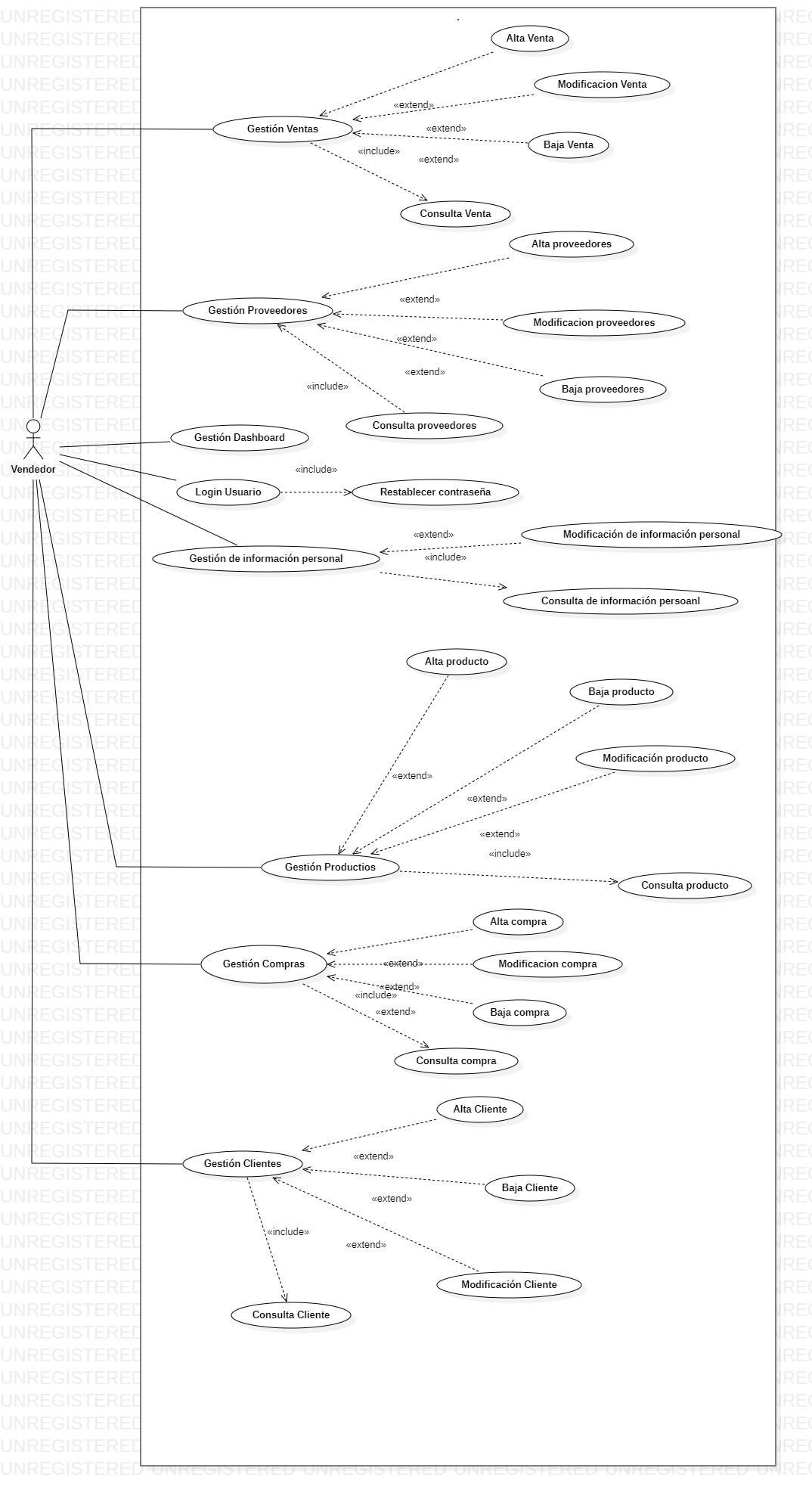
Diagrama de Gantt

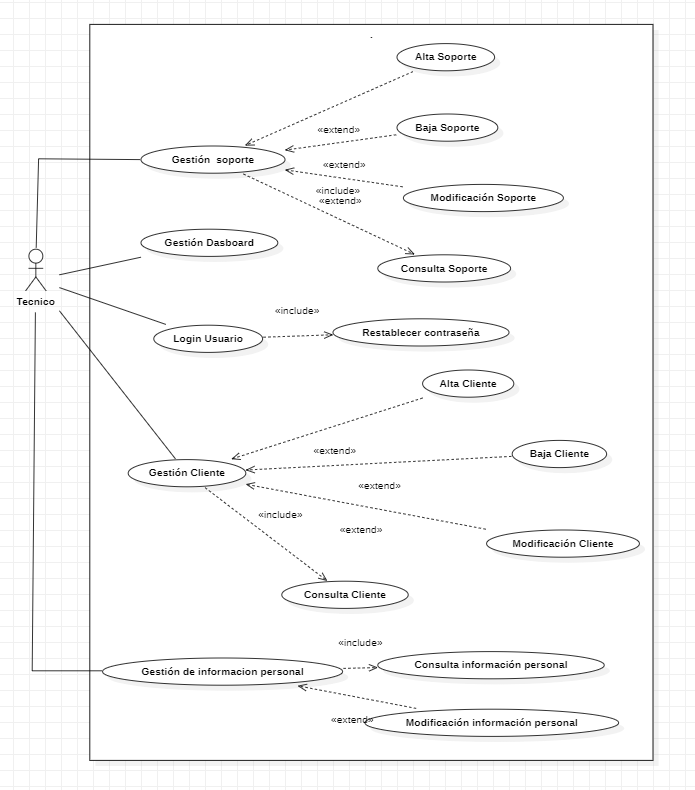
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| SEMANAS | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 |
| ACTIVIDADES |
| Análisis de situación |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Presentación del proyecto |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Límites |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Objetivos |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Dominio del sistema |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Diagramas de caso de uso |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Requerimientos |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Especificaciones de casos de uso |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Diagrama de clases |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Procesos |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Prioridades de implantación |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Desarrollo |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Plan de implantación |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

Diagrama de Casos De Usos









Especificaciones de casos de uso:

Caso de uso: Login de Usuario

* Actor: usuario.
* Descripción: el caso de uso permite el ingreso al sistema.
* Flujo básico:
  + El usuario ingresa el nombre de usuario elegido junto a su contraseña.
  + El sistema valida los datos mediante una base de datos.
  + El sistema autoriza el ingreso a la aplicación.
* Flujos Alternos:
  + En el caso que la pagina no permita el ingreso:
    - Se notificará mediante un mensaje que el nombre de usuario y/o contraseña son incorrectos. Luego se le permitirá volver a intentar el ingreso de datos en el formulario.
* Puntos de inclusión:
  + Restablecer contraseña: el usuario puede restablecer su contraseña en el caso que se la olvide.
* Puntos de extensión: no tiene.
* Notas: no tiene.

Caso de uso: Gestión Empleados

* Actor: Usuario
* Descripción: el caso de uso permite al Usuario gestionar su cuenta.
* Flujo básico:
  + El sistema mostrará en pantalla el dashboard correspondiente a su rol donde el Usuario podrá realizar las operaciones que solicite.
* Flujos Alternos:
  + En caso que no haya una sesión activa:
    - Si no hay una sesión activa al momento de solicitar este caso de uso se mostrará en pantalla un mensaje avisando al usuario que debe loguearse para poder ingresar a dicha sección.
* Puntos de inclusión:
  + Consulta de Ventas:
    - El usuario podrá consultar la cantidad de ventas, compras, servicios de soporte realizados .
  + Consulta Soporte:
    - El usuario podrá consultar los soportes y ventas realizadas e imprimir un detalle
  + Consulta de Usuarios:
    - El usuario podrá consultar según su rol los datos de otros usuarios del sistema.
  + Consulta de Stock:
    - El usuario podrá consultar el estado del stock.
* Puntos de extensión: no tiene
* Notas: no tiene.

Caso de uso: Gestión Clientes

* Actor: Usuario.
* Descripción: el caso de uso permite al Usuario registrar, consultar , modificar y dar de baja clientes.
* Flujo básico:
  + El Usuario ingresa a la sección de gestión correspondiente .
  + Según la acción que desea realizar, clickea en el botón correspondiente.
* Flujos Alternos:
  + En caso que no haya una sesión activa:
    - Si no hay una sesión activa al momento de solicitar este caso de uso se mostrará en pantalla un mensaje avisando al usuario que debe loguearse para poder ingresar a dicha sección.
* Puntos de inclusión:
  + Consulta:
    - El Usuario podrá consultar los clientes y proveedores con el fin de verificar su estado.
* Puntos de extensión:
  + Alta:
    - El Usuario podrá dar de alta a otros clientes.
  + Baja:
    - El Usuario podrá dar de baja clientes y proveedores cuando lo desee.
* Notas: no tiene.

Caso de uso: Gestión Venta

* Actor: Usuario.
* Descripción: el Usuario podrá registrar y consultar las ventas realizadas.
* Flujo básico:
  + El Usuario selecciona el botón dependiendo si desea registrar o consultar ventas.
  + El sistema cargará los datos de la base de datos y los mostrará en pantalla.
  + El Usuario hará los cambios pertinentes dependiendo la opción que haya elegido.
* Flujos Alternos:
  + En caso que no haya una sesión activa:
    - Si no hay una sesión activa al momento de solicitar este caso de uso se mostrará en pantalla un mensaje avisando al usuario que debe loguearse para poder ingresar a dicha sección.
* Puntos de inclusión:
  + Consulta:
    - El Usuario podrá consultar las compras y soporte de cada cliente.
* Puntos de extensión:
  + Alta:
    - El Usuario tendrá la posibilidad de registrar ventas y soporte, una vez que el cliente haya abonado.
  + Baja:
    - El Usuario podrá dar de baja la Venta y soporte de un cliente cuando lo desee.
* Notas: no tiene.

Caso de uso: Gestión Compra

* Actor: Usuario.
* Descripción: el Usuario podrá registrar y consultar las compras realizadas.
* Flujo básico:
  + El Usuario selecciona el botón dependiendo si desea registrar o consultar compras .
  + El sistema cargará los datos de la base de datos y los mostrará en pantalla.
  + El Usuario hará los cambios pertinentes dependiendo la opción que haya elegido.
* Flujos Alternos:
  + En caso que no haya una sesión activa:
    - Si no hay una sesión activa al momento de solicitar este caso de uso se mostrará en pantalla un mensaje avisando al usuario que debe loguearse para poder ingresar a dicha sección.
* Puntos de inclusión:
  + Consulta:
    - El Usuario podrá consultar las compras realizadas.
* Puntos de extensión:
  + Alta:
    - El Usuario tendrá la posibilidad de registrar compras.
  + Baja:
    - El Usuario podrá dar de baja las compras cuando lo desee.
* Notas: no tiene.

Caso de uso: Gestión Soporte

* Actor: Usuario.
* Descripción: el caso de uso permite al Usuario registrar, consultar y modificar soporte.
* Flujo básico:
  + El Usuario ingresa a la sección de soporte.
  + Dependiendo de la acción que desee realizar, seleccione el botón correspondiente.
* Flujos Alternos:
  + En caso que no haya una sesión activa:
    - Si no hay una sesión activa al momento de solicitar este caso de uso se mostrará en pantalla un mensaje avisando al usuario que debe loguearse para poder ingresar a dicha sección.
* Puntos de inclusión:
  + Consulta:
    - El Usuario podrá consultar los soportes que hay registrados junto a los clientes a los que pertenecen .
* Puntos de extensión:
  + Alta:
    - El Usuario podrá dar de alta un soporte.
  + Baja:
    - El Usuario podrá dar de baja un soporte.
  + Modificar:
    - El Usuario podrá modificar la información de un soporte.
* Notas: no tiene.

Caso de uso: Gestión Producto

* Actor: Usuario.
* Descripción: el caso de uso permite al Usuario registrar, consultar y modificar productos.
* Flujo básico:
  + El Usuario ingresa a la sección de producto.
  + Dependiendo de la acción que desee realizar, seleccione el botón correspondiente.
* Flujos Alternos:
  + En caso que no haya una sesión activa:
    - Si no hay una sesión activa al momento de solicitar este caso de uso se mostrará en pantalla un mensaje avisando al usuario que debe loguearse para poder ingresar a dicha sección.
* Puntos de inclusión:
  + Consulta:
    - El Usuario podrá consultar los productos que hay registrados.
* Puntos de extensión:
  + Alta:
    - El Usuario podrá dar de alta un producto.
  + Baja:
    - El Usuario podrá dar de baja un producto.
  + Modificar:
    - El Usuario podrá modificar la información de un producto.
* Notas: no tiene.

Caso de uso: Gestión Dashboard

Actor: Usuario

Descripción: el caso de uso permite al usuario gestionar el sistema según el rol que cumplan en el sistema.

Flujo básico:

* + Según el rol que cumpla el usuario verá la pantalla principal del sistema.
  + Según la acción que desee realizar, el Usuario, tendrá en un sidebar los botones que lo llevarán a dichas acciones.

Flujos Alternos:

* + En caso que no haya una sesión activa:
    - Si no hay una sesión activa al momento de solicitar este caso de uso se mostrará en pantalla un mensaje avisando al usuario que debe loguearse para poder ingresar a dicha sección.

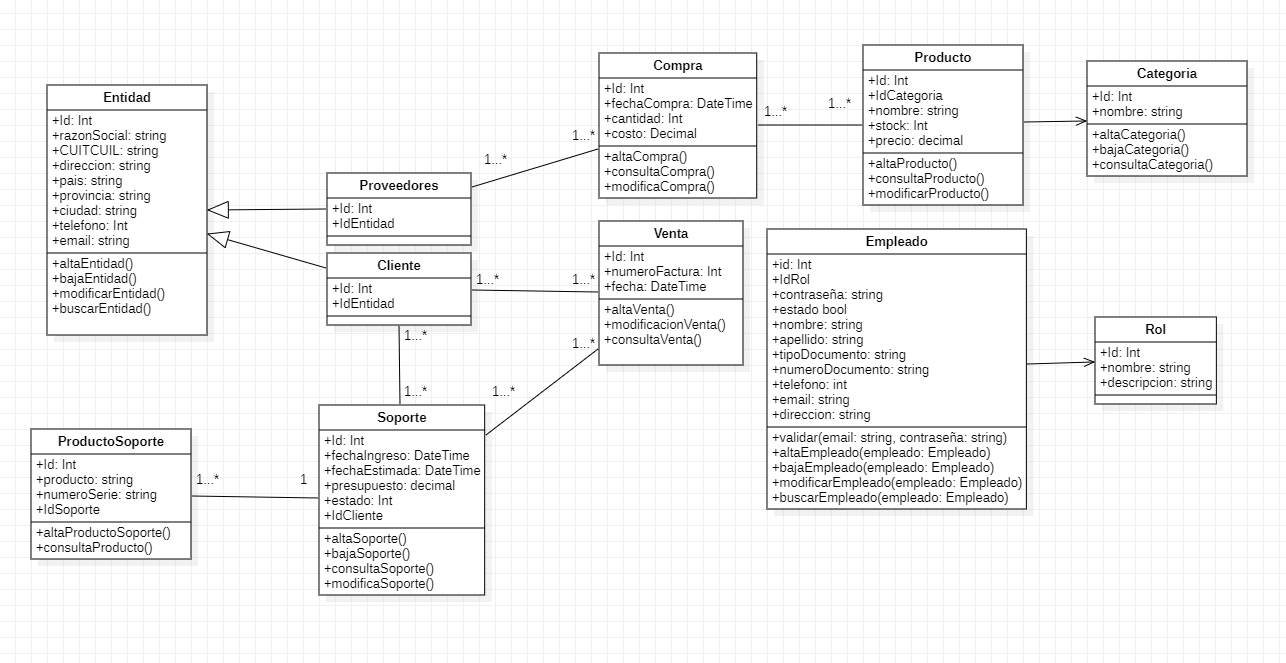
Puntos de inclusión: no tiene.

Puntos de extensión: no tiene.

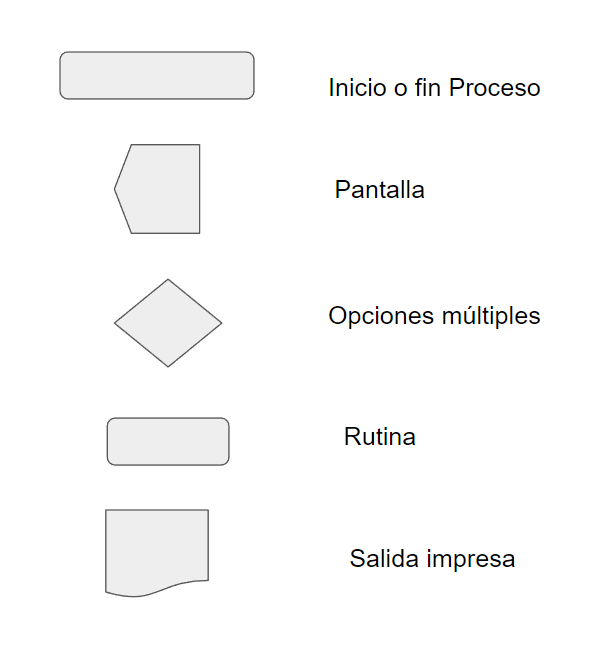
Notas: no tiene.

* Diseño del Sistema:

***Diagrama de Clases:***

******

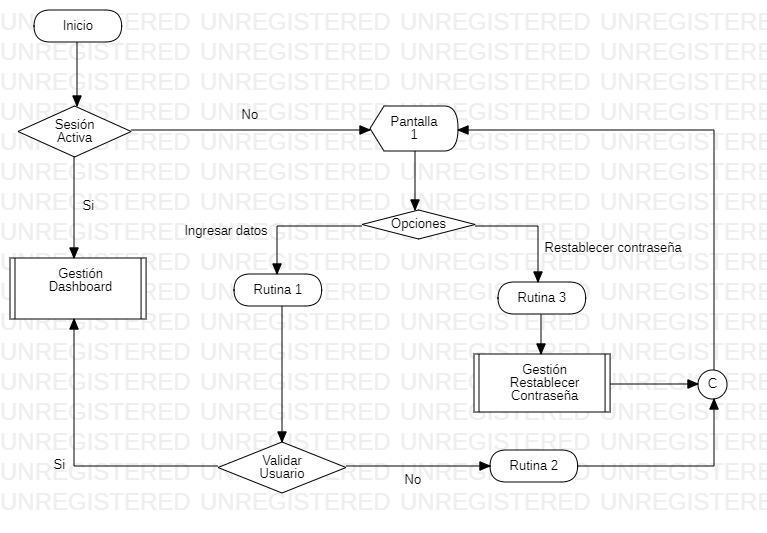
***Simbología Utilizada***



**Procesos :**

**Proceso Login de Usuario**

1. FlowChart : LogIn



**Rutinas**

public class **Roles** {

private int idRol;

private string nombreRol;

private string descripcionRol;

public **Rol** (int idRol, string nombreRol,string descripcionRol) {

this.idRol = idRol;

this.nombreRol = nombreRol;

this.descripcionRol = descripcionRol;

}

public int Rol::getidRol () {

retornar this.idRol;

}

public string Rol::getNombreRol () {

retornar this.nombreRol;

}

}

public class **Empleados** {

public int id;

public Int idRol;

public string usuario;

public string contraseña;

public string nombre;

public string apellido;

public string tipoDocumento;

public string numeroDocumento;

public string telefono;

public string email;

public string direccion;

public bool estado;

public **Empleados** (int id,string usuario, string nombre, string apellido, char numeroDocumneto,string tipoDocuemnto, string telefono, string email,string direccion, string contraseña, bool estado)

{

this.id = id;

this.usuario = usuario

this.contraseña = contraseña;

this.nombre = nombre;

this.apellido = apellido;

this.tipoDocumento = tipoDocuemnto

this. numeroDocumento= numeroDocumento;

this.telefono = telefono;

this.email = email;

this.direccion = direccion;

this.estado = estado;

}

}

public **buscarEmpleado** (String usuario) {

array L\_empleado;

L\_ empleado = SELECT \* FROM empleados WHERE empleados.usuario == usuario

Si L\_empleado(0).usuario != "" entonces

retornar L\_empleado;

FinSi

}

public **validar** (string usuario, string contraseña)

{

this.valido = falso;

buscarEmpleado(usuario);

Si L\_empleado(0).usuario != ""

Si L\_empleado(0).contraseña = contraseña entonces

this.valido = verdadero;

this.idEmpleado = L\_empleado(0).id ;

this.idRol = L\_empleado(0).idRol;

Sino

mostrar mensaje("Usuario o Contraseña inválidos");

Fin Si

Sino

mostrar mensaje("Debe introducir un usuario y contraseña");

Fin Si

this.valido;

}

public **altaEmpleado** (string usuario) {

buscarEmpleado(usuario);

Si L\_Empleado(0).usuario = "" entonces

CREATE TABLE IF NOT EXISTS empleado

( id INT AUTO\_INCREMENT PRIMARY KEY,usuario VARCHAR,contraseña VARCHAR, nombre VARCHAR, apellido VARCHAR,tipoDocumento VARCHAR, numeroDocumneto VARCHAR, telefono INT, email VARCHAR,direccion VARCHAR, estado bit,idRol int)

INSERT INTO empleados(usuario,contraseña,nombre, apellido,tipoDocumento, numeroDocumneto, telefono, email,direccion,estado) VALUES(this.usuario ,this.usuario,this.nombre , this.apellido , this.tipoDocuemtno ,this.numeroDocumneto, this.telefono,this.email, this.direccion,this.estado,this.idRol);

}

public **bajaEmpleado** (string usuario) {

buscarEmpleado(usuario);

Si L\_empleado(0).usuario != "" entonces

Update FROM empleados set estado = “Baja ”WHERE L\_empleados(0).idEmpleado = idEmpleado

}

public **modificarEmpleado** (string usuario) {

buscarEmpleado(usuario);

Si L\_Empleado(0).usuario != "" entonces

CREATE PROCEDURE IF NOT EXISTS **modificarEmpleado**

(

@idEmpleado int,

@usuario varchar (50),

@contraseña varchar(50),

@idrol int,

@correo varchar(50),

@nombre varchar (50),

@apellido varchar (50),

@tipoDocuemento varchar (50),

@numeroDocumento varchar (50),

@Direccion varchar (50),

@email varchar(50),

@estado bit

)

as

update empleados set @usuario = usuario, contraeña = @contraseña,idRol = @idrol nombre = @nombre,apellido = @apellido,email = @correo,tipoDocumento = @tipoDocuemento, numeroDocumento = @numeroDocumento, direccion = @Direccion, email = @email , @estado = estado where id = @idEmpleado

go

}

public **restablecerContraseña** (string email) {

T\_empleados = SELECT \* FROM empleados where email == this.email

Si T\_empleados != “” entonces

Enviar email de restablecimiento

sino

mostrar mensaje: “ Verificar correo ingresado”

fin si

fin si

}

public class **Cliente** extends Entidad

**{**

private int idCliente;

private int idEntidad;

public Cliente(int idCliente, int Clase\_idClase)

{

this.idCliente = idCliente; this.idClase = idClase;

}

public int Cliente::getidCliente ()

{

retornar this.idCliente;

}

public **buscarCliente** (Cliente cliente) {

array L\_clientes;

L\_ clientes = SELECT \* FROM clientes, entidad

INNER JOIN clientes ON clientes.IdEntidad = Entidad.id

WHERE clientes IN (SELECT \* FROM entidad WHERE entidad. CUIT/CUIL= CUIT/CUIL)

Si L\_clientes(0). CUIT/CUIL!= "" entonces

retornar L\_clientes;

FinSi

}

public **altaCliente** (Cliente cliente) {

**buscarCliente**(cliente);

Si L\_Cliente(0). CUIT/CUIL= "" entonces

CREATE TABLE IF NOT EXISTS Entidad

( id INT AUTO\_INCREMENT PRIMARY KEY, RazonSocial VARCHAR, CUIT/CUIL VARCHAR, direccion VARCHAR, pais VARCHAR, provincia VARCHAR, ciudad VARCHAR, telefono INT, email VARCHAR)

CREATE TABLE IF NOT EXISTS clientes

( id INT AUTO\_INCREMENT PRIMARY KEY, FOREIGN KEY (cliente\_idCliente) REFERENCES cliente (idCliente), estado bit);

INSERT INTO Entidad(RazonSocial, CUIT/CUIL, direccion , pais,provincia,ciudad,telefono,email) VALUES(this.RazonSocial,this.CUIT/CUIL,this.direccion,this.pais,this.provincia,this.ciudad,this.email);

INSERT INTO clientes (IdEntidad) VALUES( this.IdEntidad, this.estado); }

public **bajaCliente** (Cliente cliente)

{

**buscarCliente**(Cliente.CUIT/CUIL);

Si L\_Cliente(0). CUIT/CUIL!= "" entonces

Update FROM clientes set estado = “Baja ”WHERE L\_clientes(0).id = id

}

public **modificarCliente** (Cliente cliente) {

**buscarCliente**(Cliente.CUIT/CUIL);

Si L\_Cliente(0). CUIT/CUIL!= "" entonces

CREATE PROCEDURE IF NOT EXISTS **modificarCliente**

(

@idCliente int,

@idEntidad int,

@razonSocial varchar(50)

@CUIT/CUIL varchar(50),

@direccion varchar (50),

@ciudad varchar (50),

@provincia varchar (50),

@pais varchar (50),

@telefono int,

@email varchar(50)

)

as

update clientes idEntidad= @idEntidad set a where id =@ idCliente

update entidad set razonSocial = @razonSocial, CUIT/CUIL= @CUIT/CUIL, direccion = @Direccion, ciudad= @ciudad, provincia= @provincia , pais =@pais where id = @idEntidad

go

}

**}**

**RUTINA 1**: validar(usuario,contraseña);

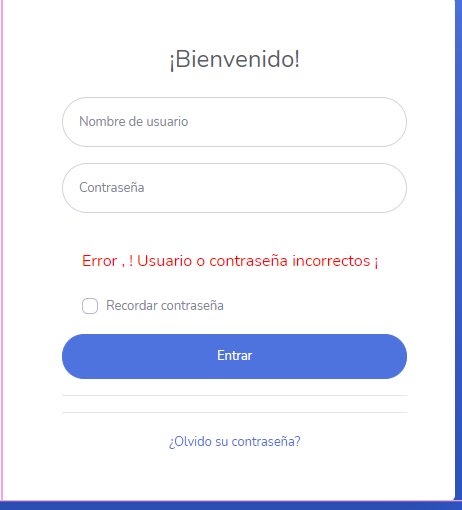
**RUTINA 2**: Muestra mensaje ("El usuario y/o contraseña son inválidos. Vuelva a intentarlo") txtUsuario = ""; txtPass = "";

**RUTINA 3:** Llama al subproceso Gestión Restablecer contraseña.

1. ➔Pantallas: Pantalla Login

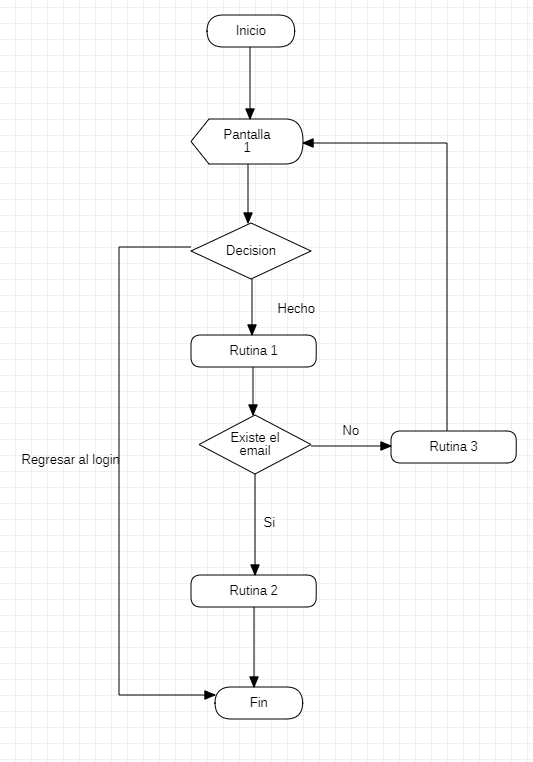


1. ➔Pantallas: Pantalla Error



**Proceso Restablecer contraseña**

1. **FlowChart : Restablecer contraseña**



➔Rutinas:

**RUTINA 1: Verifica que el email ingresado exista en el sistema.**

**RUTINA 2: Se envía un email con una contraseña provisoria.**

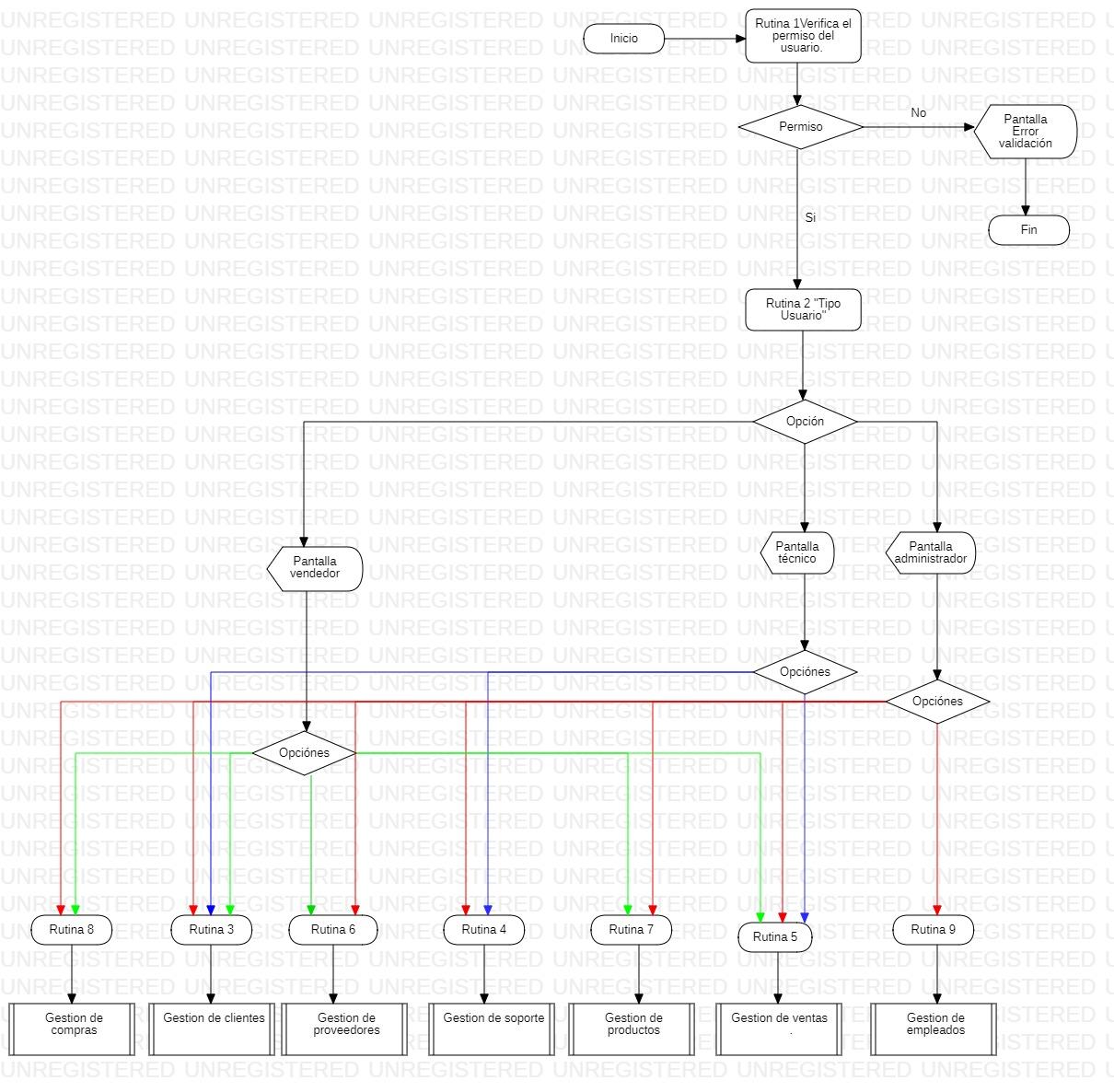
**RUTINA 3: Muestra un mensaje de error “El email ingresado no existe” y limpia la caja de texto.**

➔Pantallas: Pantalla 1 (Restablecer contraseña):

****

**Proceso Gestión Dashboard**

1. **FlowChart : Dashboard**



**Rutinas:**

**RUTINA 1: Verifica el permiso del usuario.**

**RUTINA 2: Según el tipo de empleado determina qué rol cumple en el sistema otorgando los privilegios necesarios.**

**RUTINA 3: Llama al subproceso Gestión Clientes.**

**RUTINA 4: Llama al subproceso Gestión Soporte.**

**RUTINA 5: Llama al subproceso Gestión Ventas.**

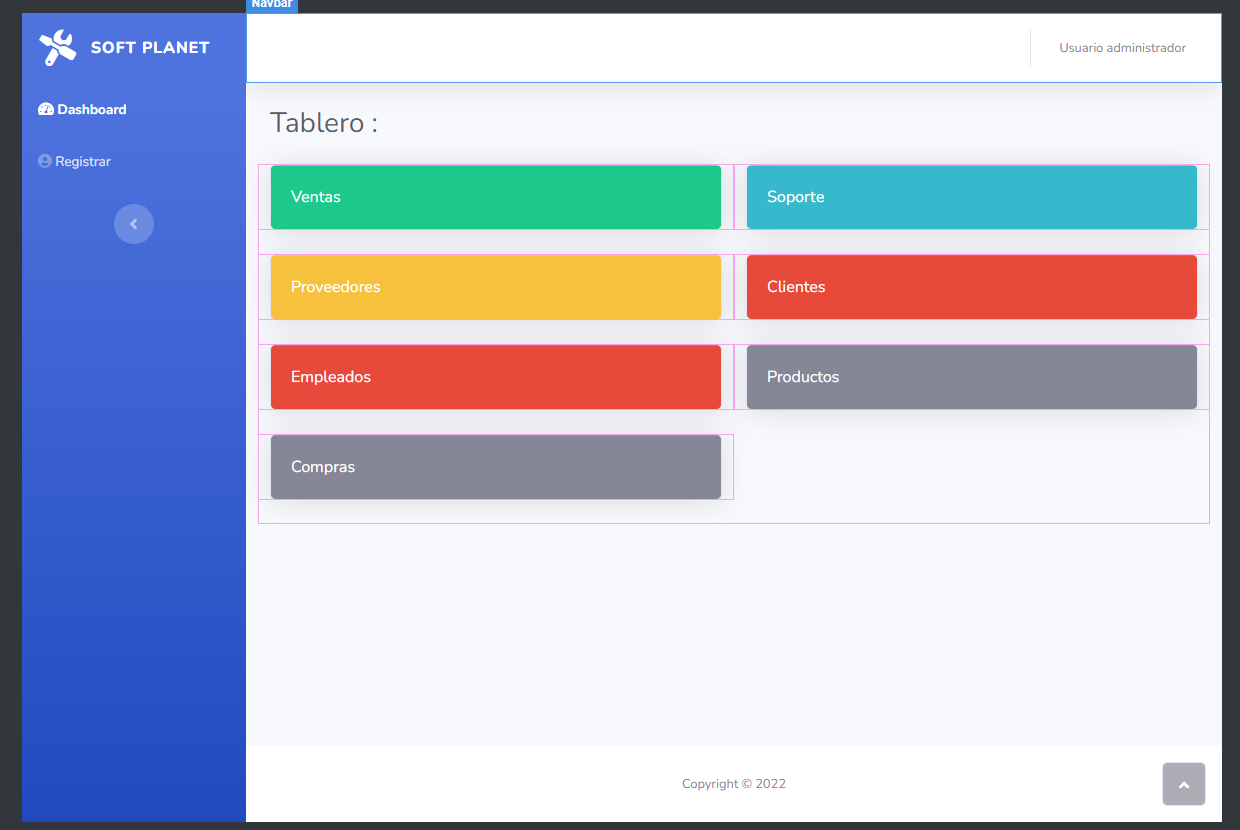
**RUTINA 6: Llama al subproceso Gestión Proveedores.**

**RUTINA 7: Llama al subproceso Gestión Productos.**

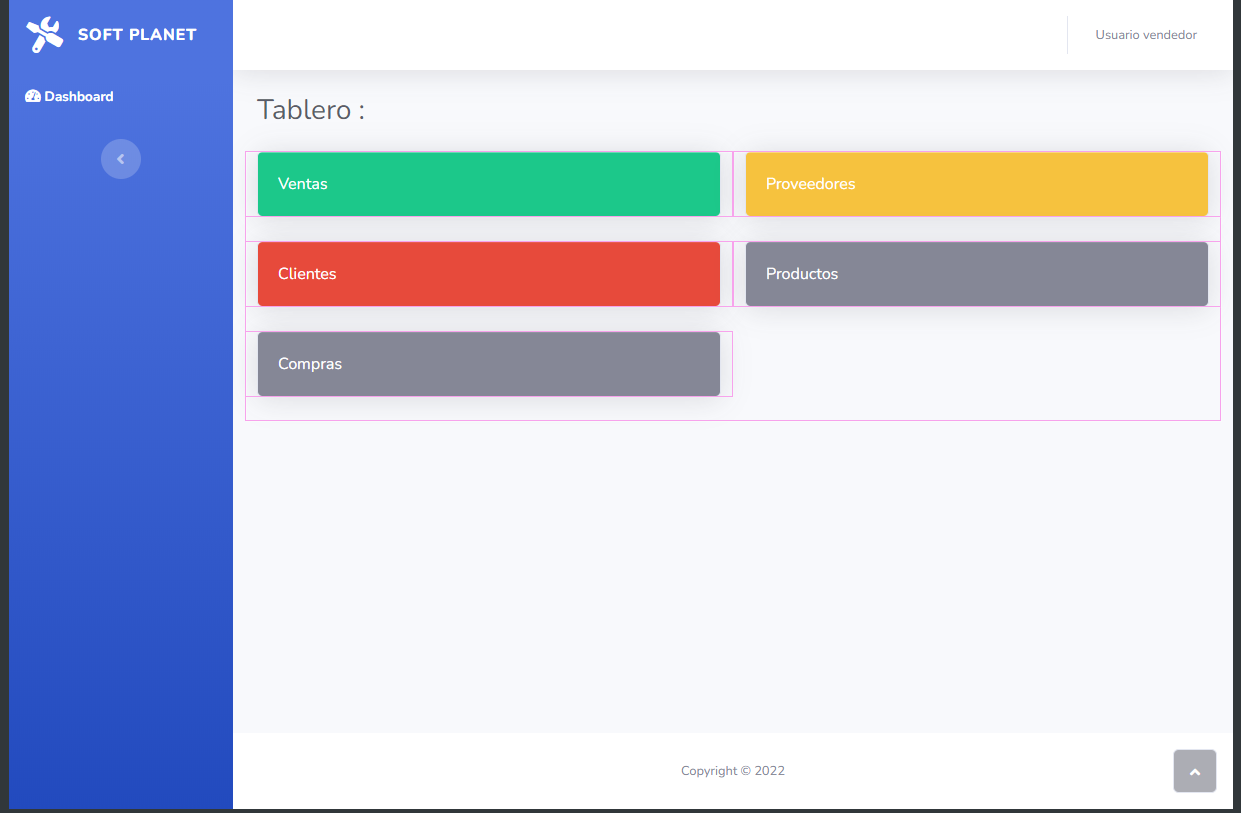
**RUTINA 8: Llama al subproceso Gestión Compras.**

**RUTINA 9: Llama al subproceso Gestión Empleados.**

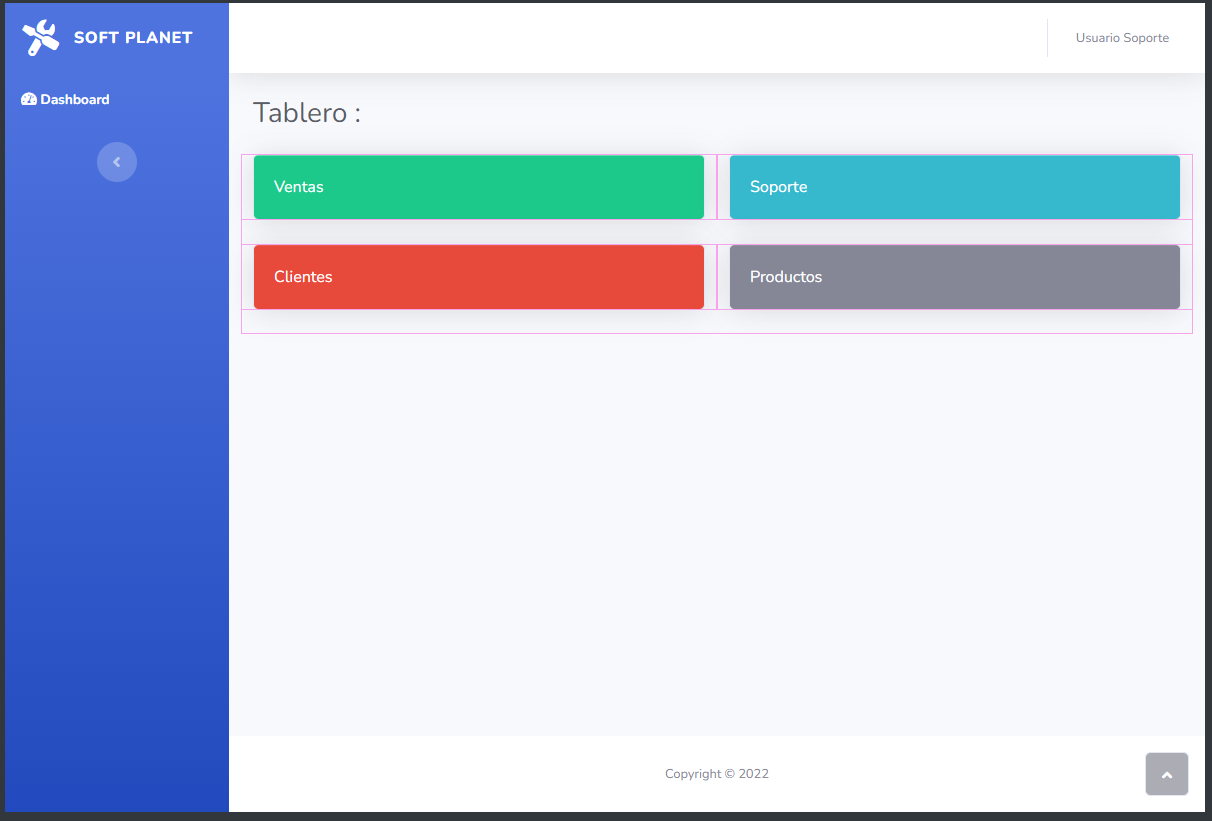
➔Pantallas: Pantalla Administrador

****

➔Pantallas:Pantalla vendedor



➔Pantallas: Pantalla Técnico:

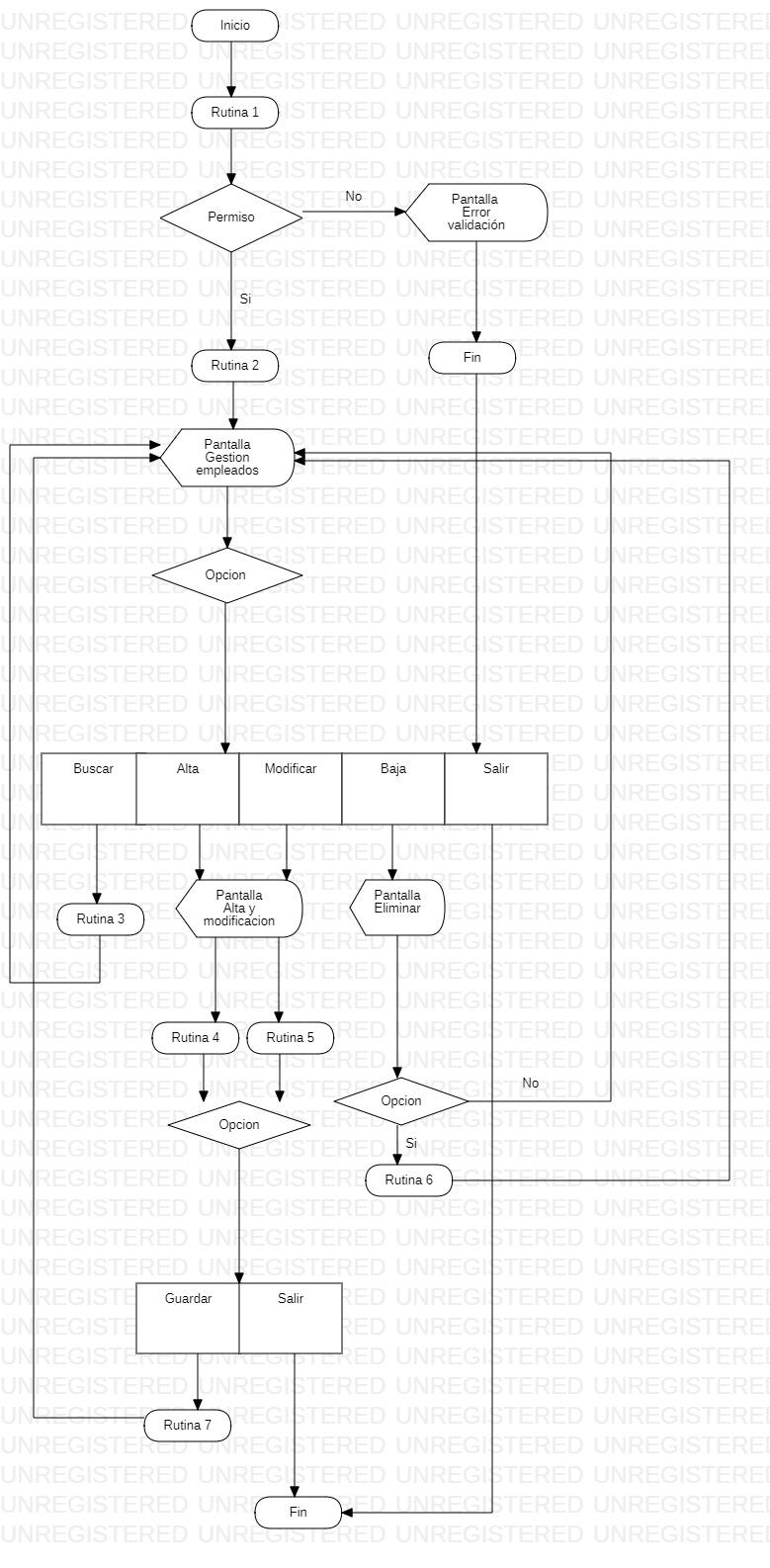
****

➔Pantallas: Error validación :



**Proceso ABM Empleado:**

1. **FlowChart : Gestion Empleados**



**Rutinas AMB Empleado**

➔Rutinas:

**RUTINA 1: Verifica que el usuario tenga permiso para ingresar a dicho proceso.**

**RUTINA 2: Muestra una grilla con los datos de la tabla empleado con sus respectivos botones de acciones.**

**RUTINA 3: Busca y selecciona un empleado por su nombre.**

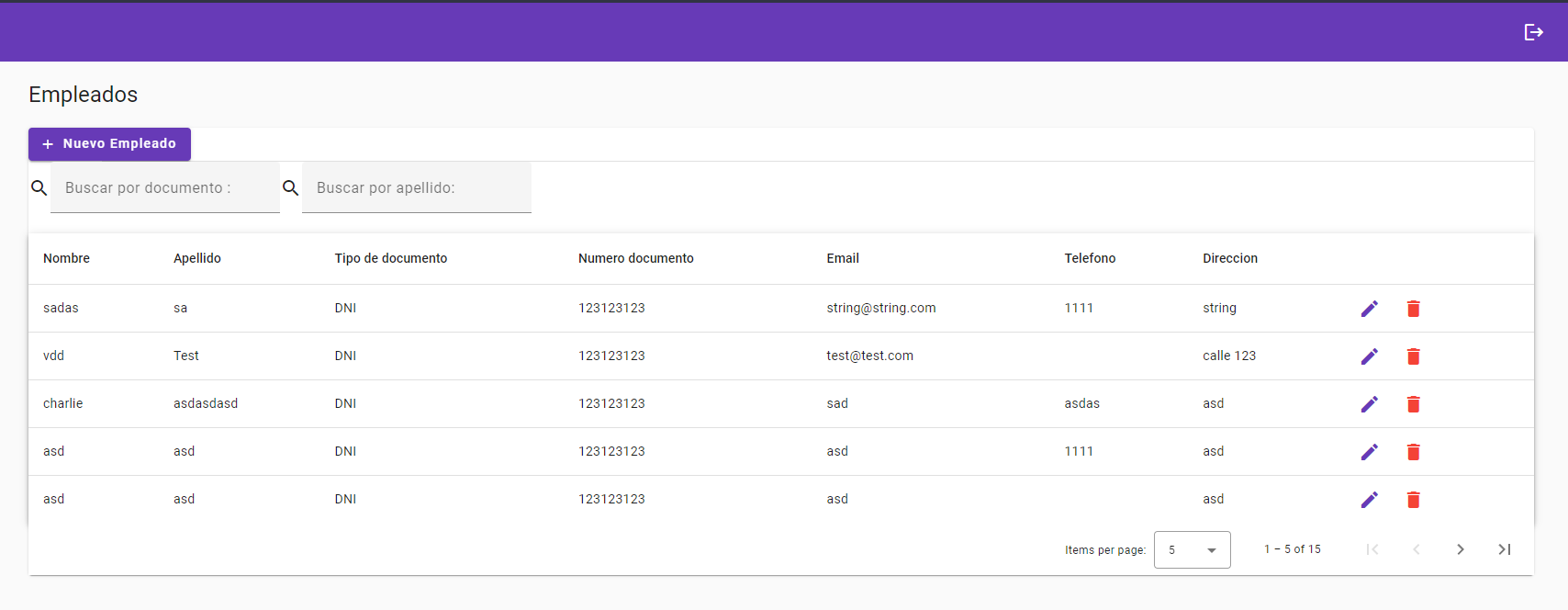
**RUTINA 4: Muestra los campos del formulario con los datos del registro seleccionado para actualizarlos.**

**RUTINA 5:Muestra un formulario vacío para el ingreso de datos del registro de empleado .**

**RUTINA 6:Borra y actualiza el registro seleccionado en la tabla y la base de datos**

**RUTINA 7: Graba o actualiza los registros del empleado en la base de datos.**

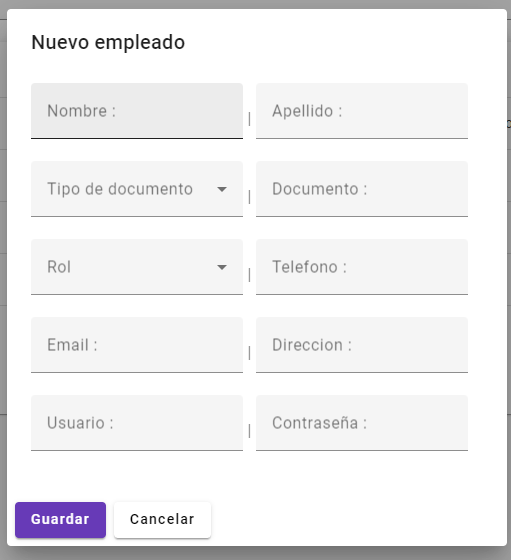
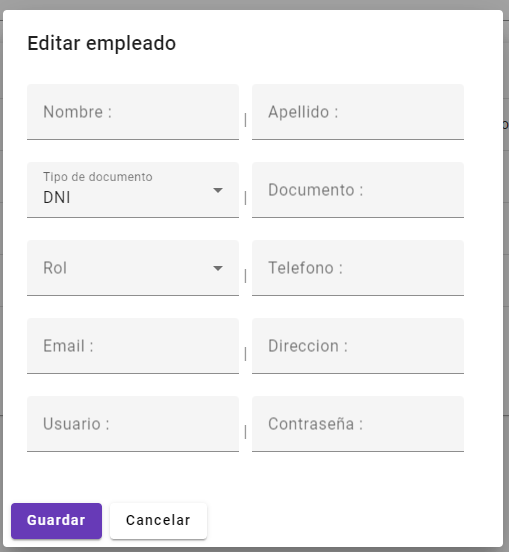
➔Pantallas: Gestión Empleados



➔Pantallas:Eliminar



➔Pantallas: Alta-Modificación

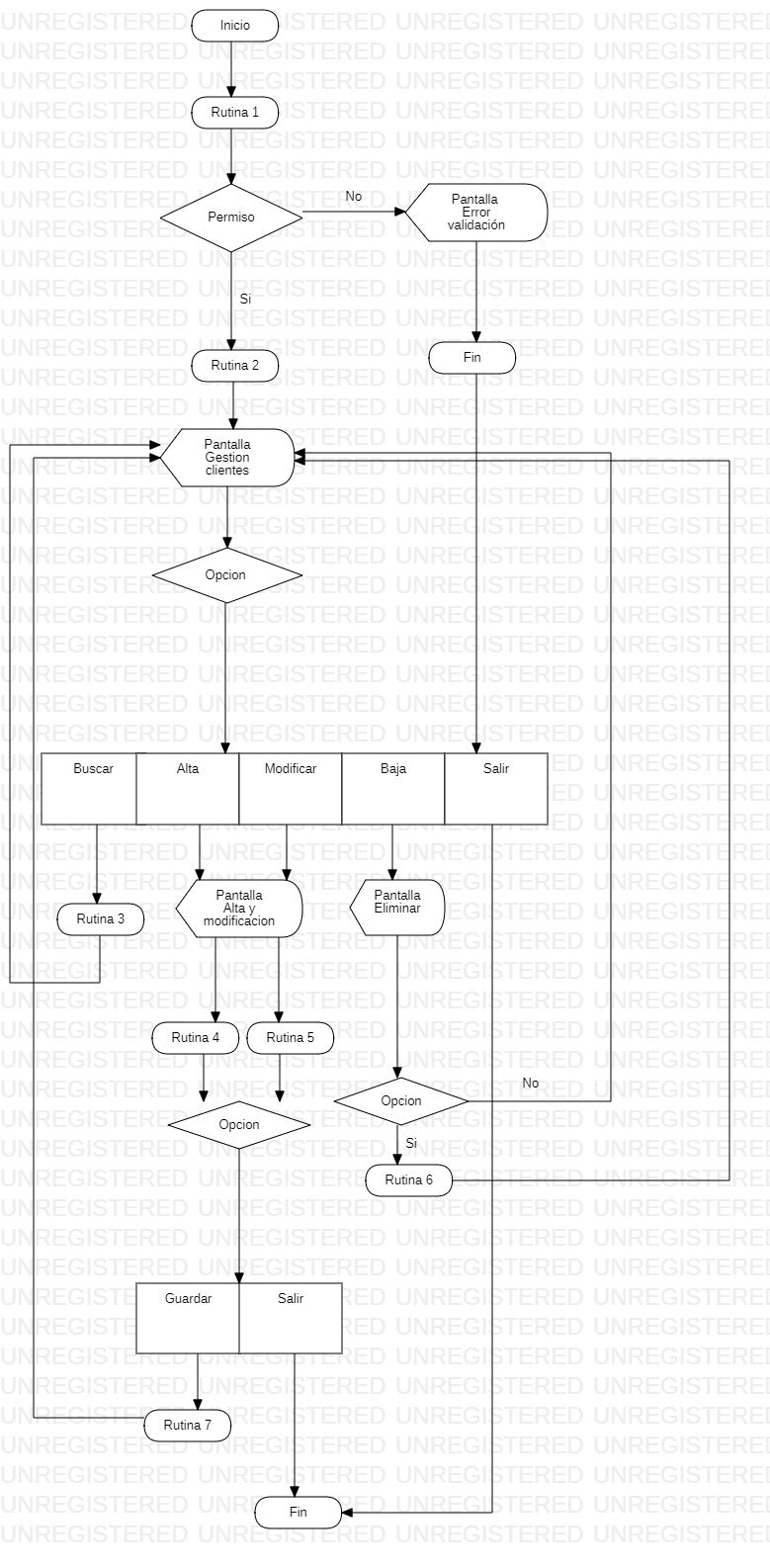
****

➔Pantallas: Error validación :



Proceso **ABM Clientes:**

1. **FlowChart : Clientes**



**Rutinas ABM Clientes:**

➔Rutinas:

**RUTINA 1: Verifica que el usuario tenga permiso para ingresar a dicho proceso.**

**RUTINA 2: Muestra una grilla con los datos de la tabla cliente con sus respectivos botones de acciones.**

**RUTINA 3: Busca y selecciona un cliente por su nombre.**

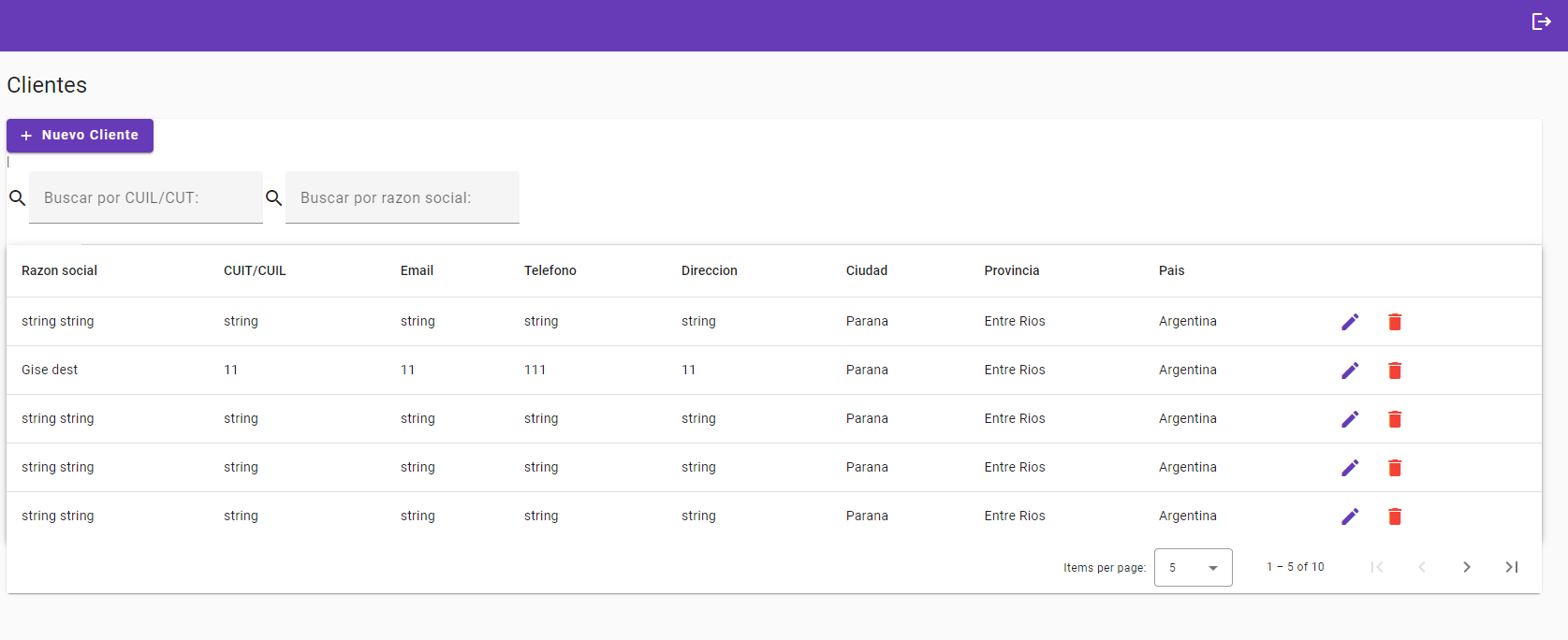
**RUTINA 4: Muestra un formulario vacío para el ingreso de datos del registro de usuario.**

**RUTINA 5:Muestra los campos del formulario con los datos del registro seleccionado para actualizarlos.**

**RUTINA 6: Borra y actualiza el registro seleccionado en la tabla y la base de datos**

**RUTINA 7: Graba o actualiza los registros del usuario en la base de datos.**

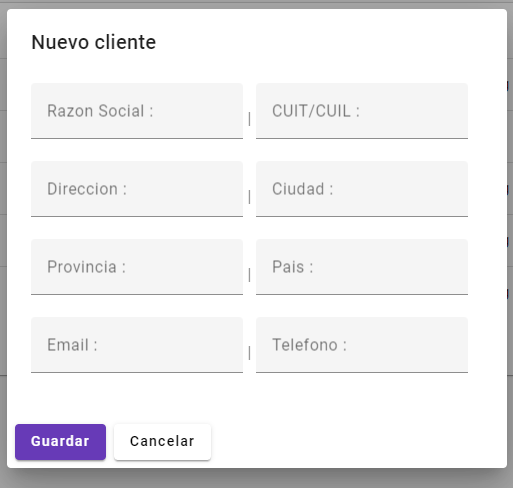
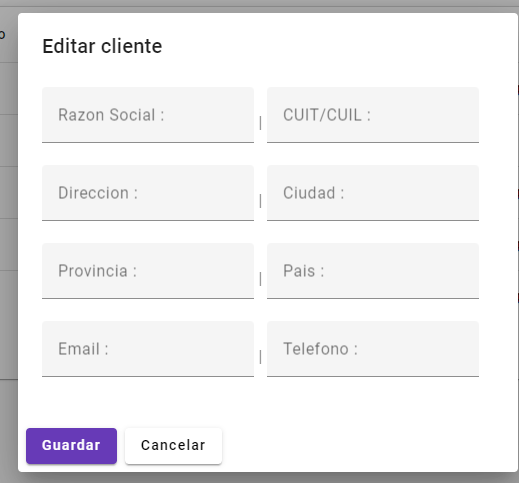
➔Pantallas: Pantalla Gestión Clientes



➔Pantallas: Pantalla Eliminar



➔Pantallas: Pantalla 3 (Alta-Modificación)

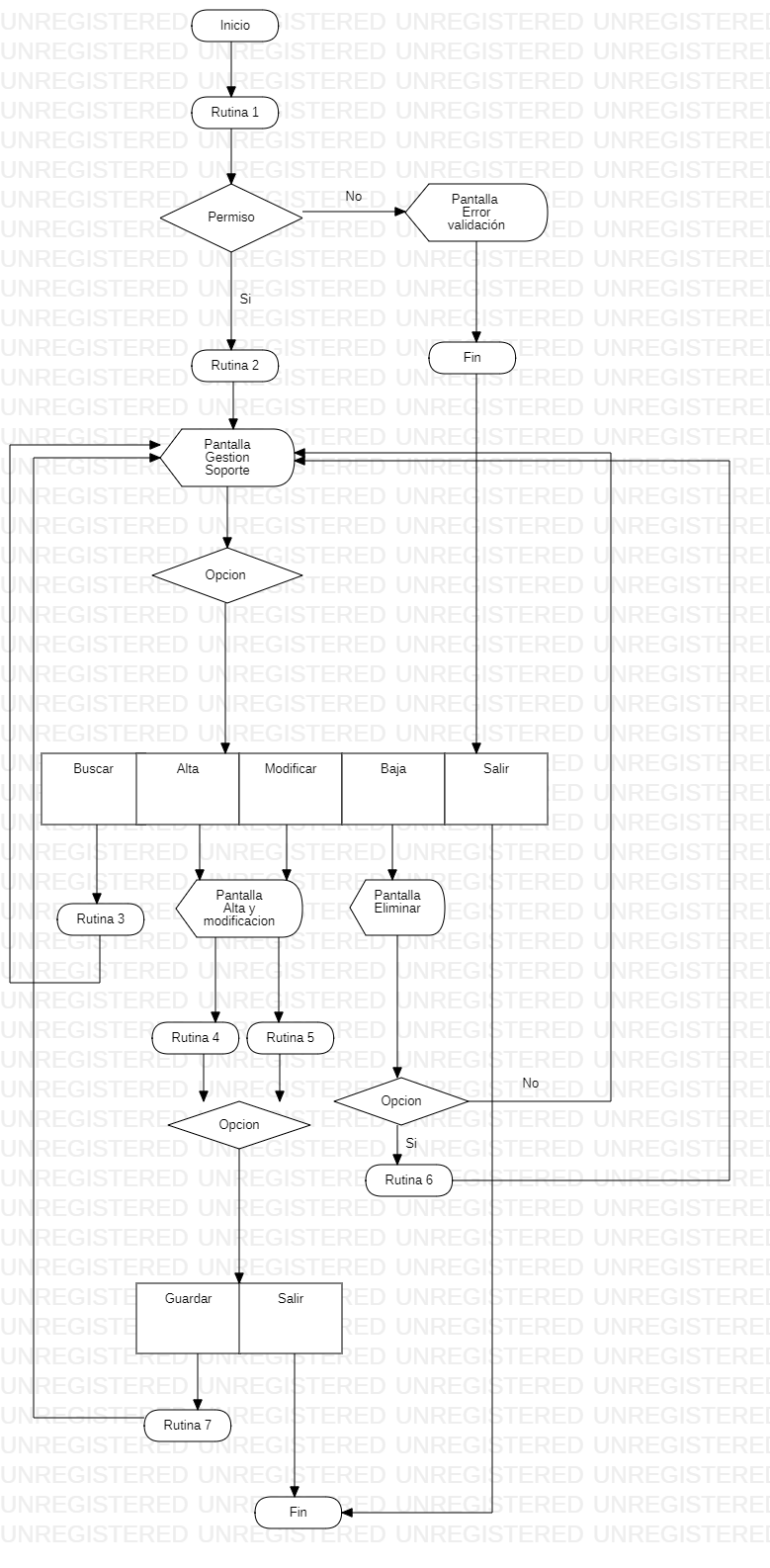
****

➔Pantallas: Error validación :



**Proceso ABM Soporte**

1. **FlowChart : Gestión soporte**



**Rutina: ABM Soporte**

➔Rutinas:

**RUTINA 1: Verifica que el usuario tenga permiso para ingresar a dicho proceso.**

**RUTINA 2: Muestra una grilla con los datos de la tabla soporte con sus respectivos botones de acciones.**

**RUTINA 3: Busca y selecciona un soporte por el cliente .**

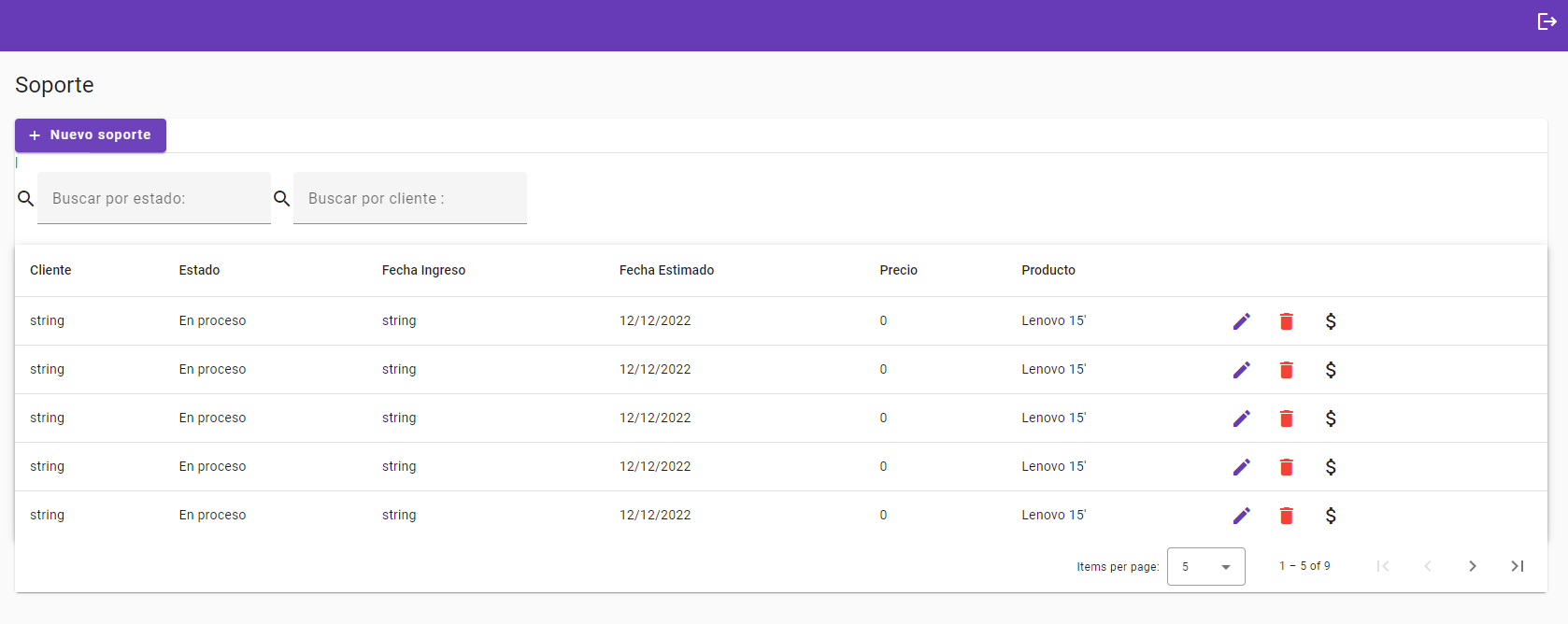
**RUTINA 4: Muestra un formulario vacío para el ingreso de datos del registro de un soporte y soporteDetalle.**

**RUTINA 5: Muestra los campos del formulario con los datos del registro seleccionado para actualizarlos.**

**RUTINA 6: Borra y actualiza el registro seleccionado en la tabla y la base de datos**

**RUTINA 7: Graba o actualiza los registros del usuario en la base de datos.**

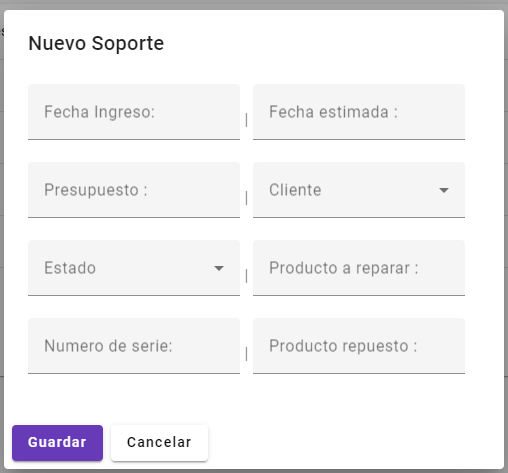
➔Pantallas: Pantalla Gestión Soporte

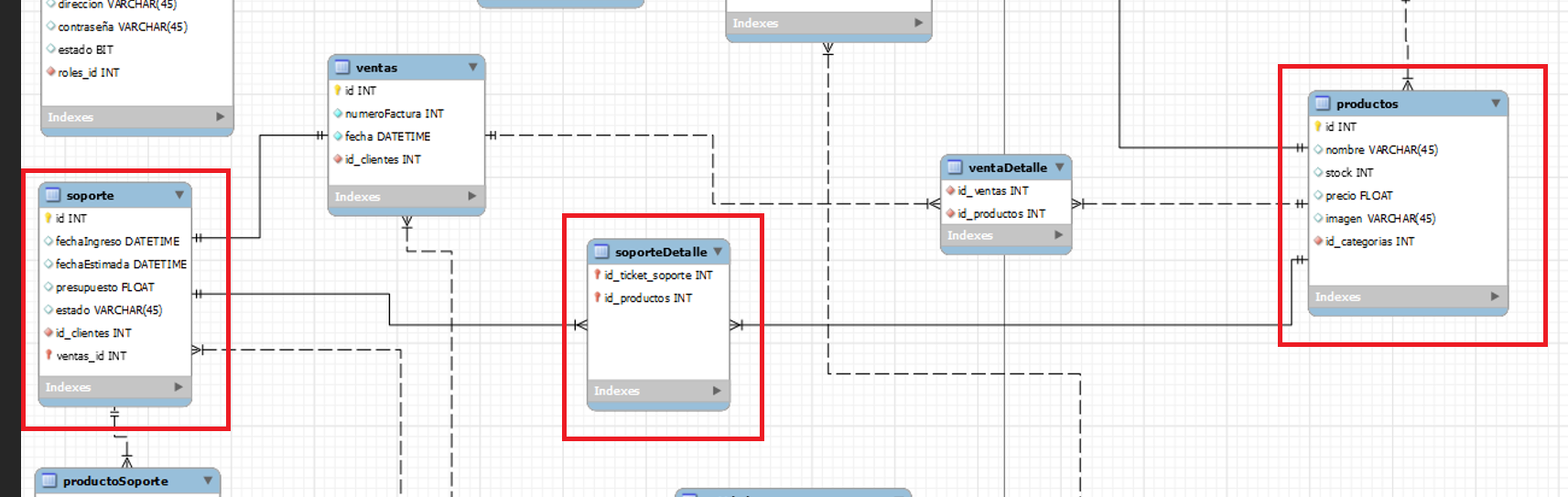


➔Pantallas: Pantalla Eliminar soporte



1. ➔Pantallas: Pantalla Alta - Modificación soporte

****

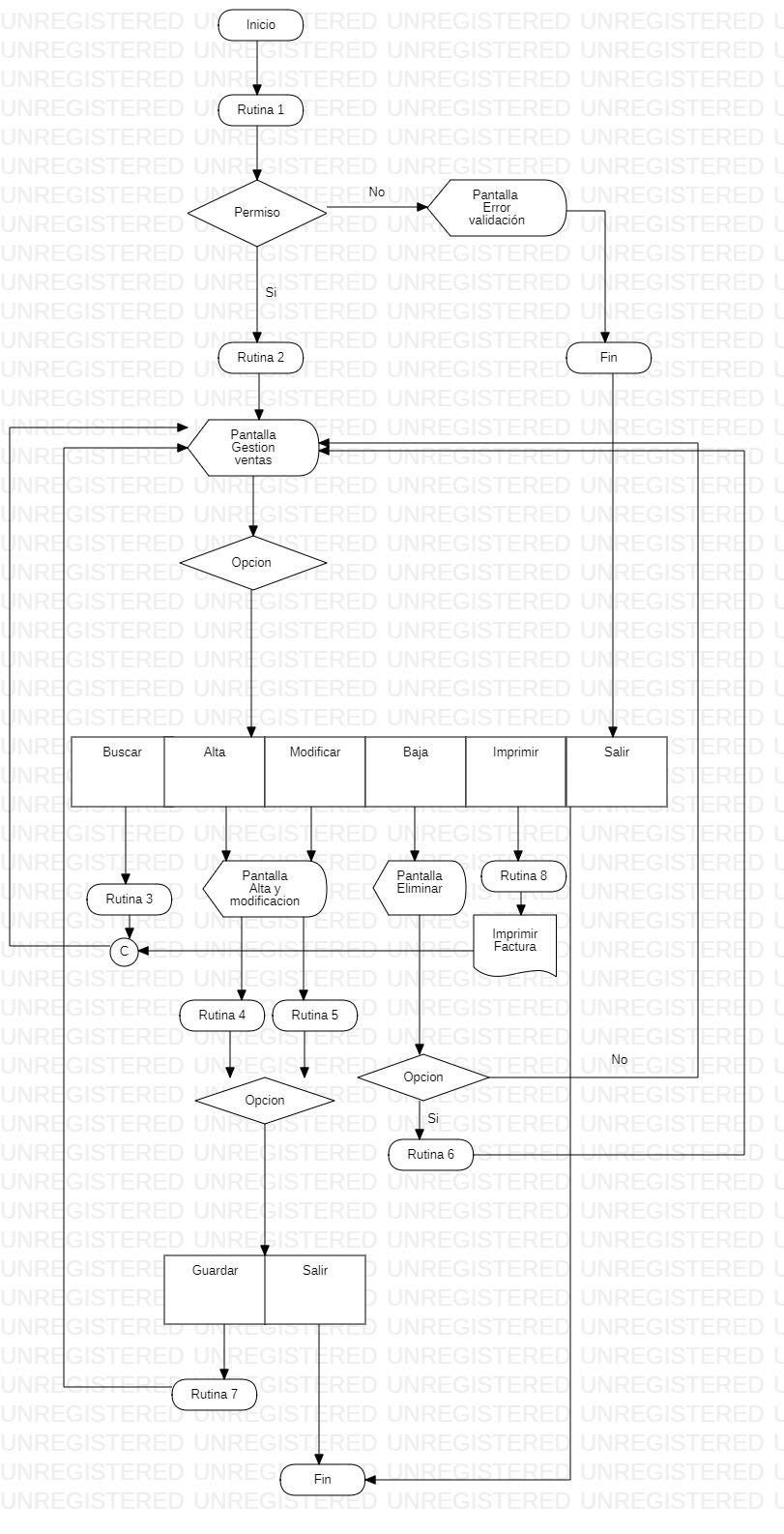


➔Pantallas: Error validación :



**Proceso ABM Ventas**

**FlowChart : Gestion ventas**:



**Rutinas ABM Ventas**

➔Rutinas:

**RUTINA 1: Verifica que el usuario tenga permiso para ingresar a dicho proceso.**

**RUTINA 2: Muestra una grilla con los datos de la tabla ventas ,clientes con sus respectivos botones de acciones.**

**RUTINA 3: Busca y selecciona una venta por cliente.**

**RUTINA 4:Muestra un formulario vacío para el ingreso de datos del registro de una nueva venta.**

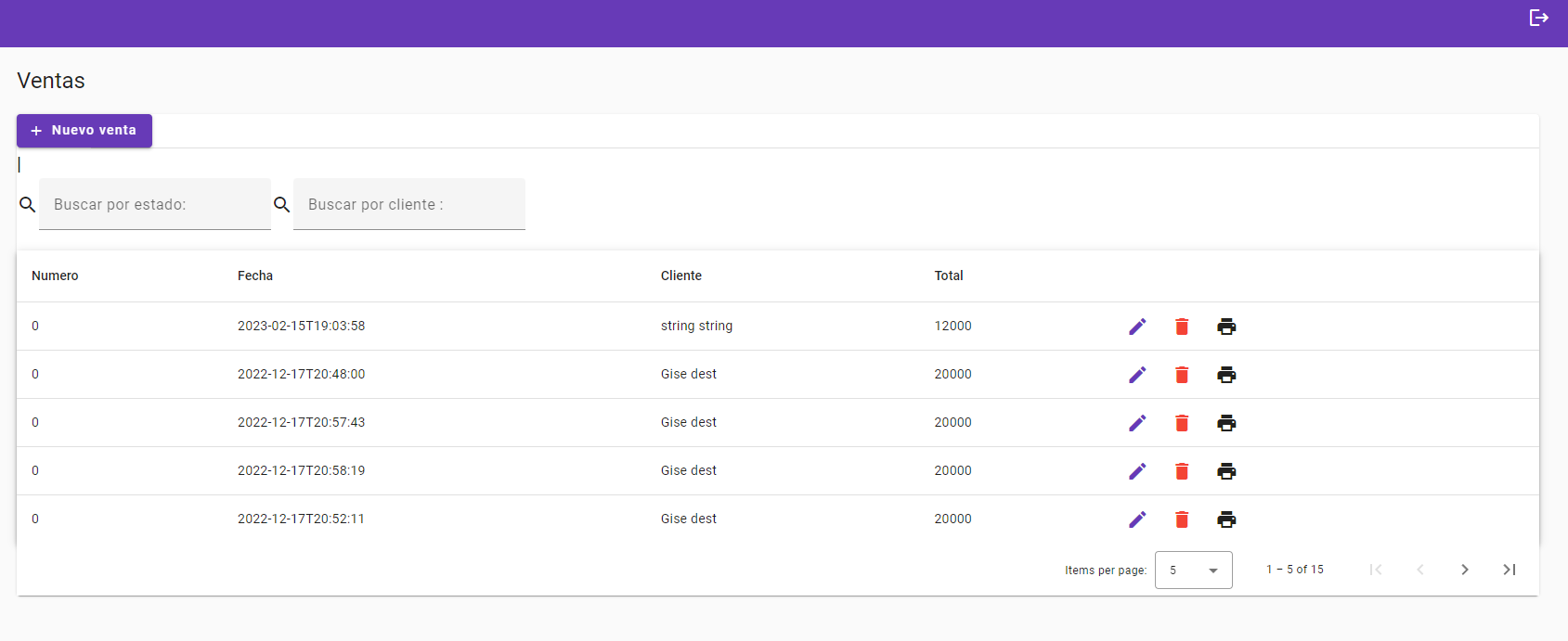
**RUTINA 5:Muestra los campos del formulario con los datos del registro seleccionado para actualizarlos.**

**RUTINA 6:Borra y actualiza el registro seleccionado en la tabla y la base de datos**

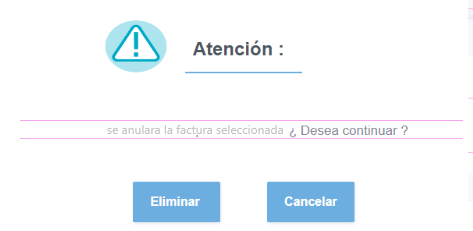
**RUTINA 7: Graba o actualiza los registros de ventas en la base de datos.**

**RUTINA 8: Imprime comprobante de la venta realizada**

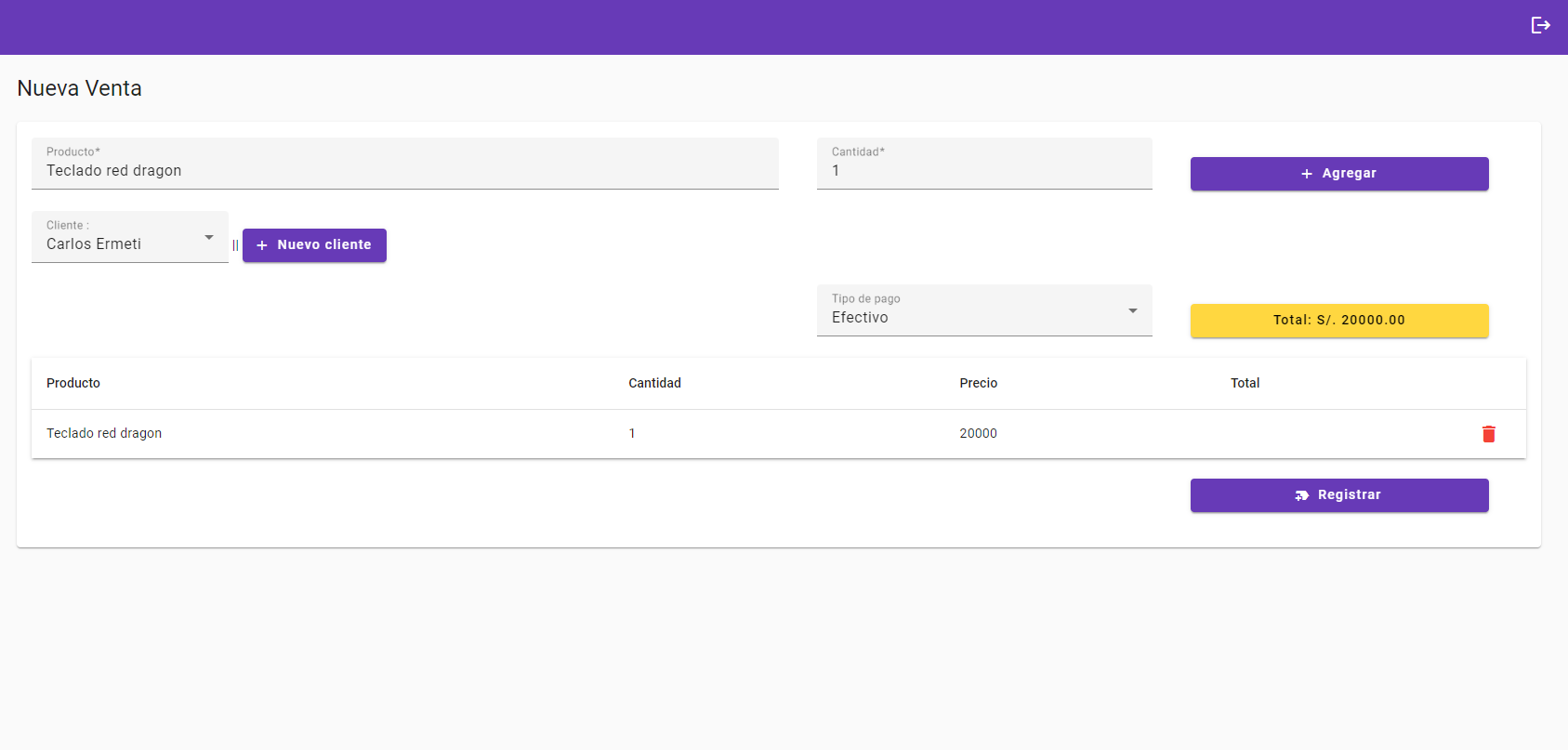
➔Pantallas: Pantalla Gestión ventas



➔Pantallas: Pantalla Eliminar



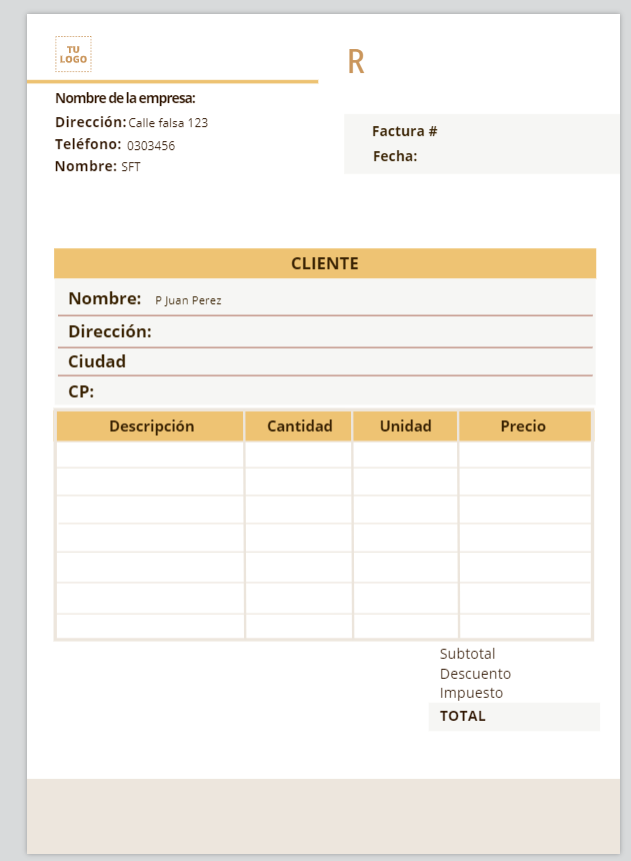
➔Pantallas: Pantalla Alta - Modificación



➔Pantallas: Error validación :

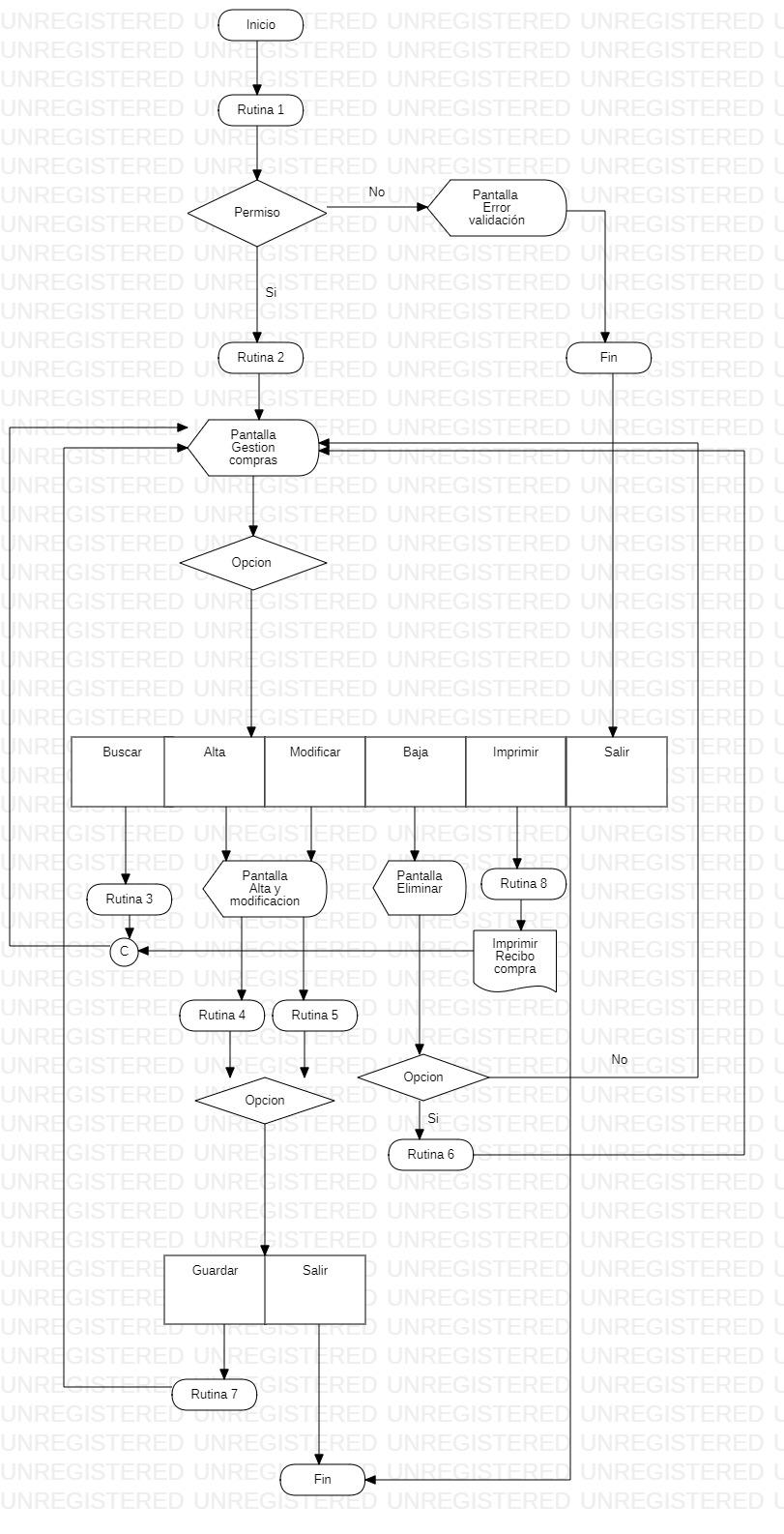


➔Salida Impresa :



**Proceso ABM Compras**

1. **FlowChart : Gestion compras**



**Rutinas ABM Compras**

➔Rutinas:

**RUTINA 1: Verifica que el usuario tenga permiso para ingresar a dicho proceso.**

**RUTINA 2: Muestra una grilla con los datos de la tabla compras ,proveedores con sus respectivos botones de acciones.**

**RUTINA 3: Busca y selecciona una compra por proveedor.**

**RUTINA 4:Muestra un formulario vacío para el ingreso de datos del registro de una nueva compra.**

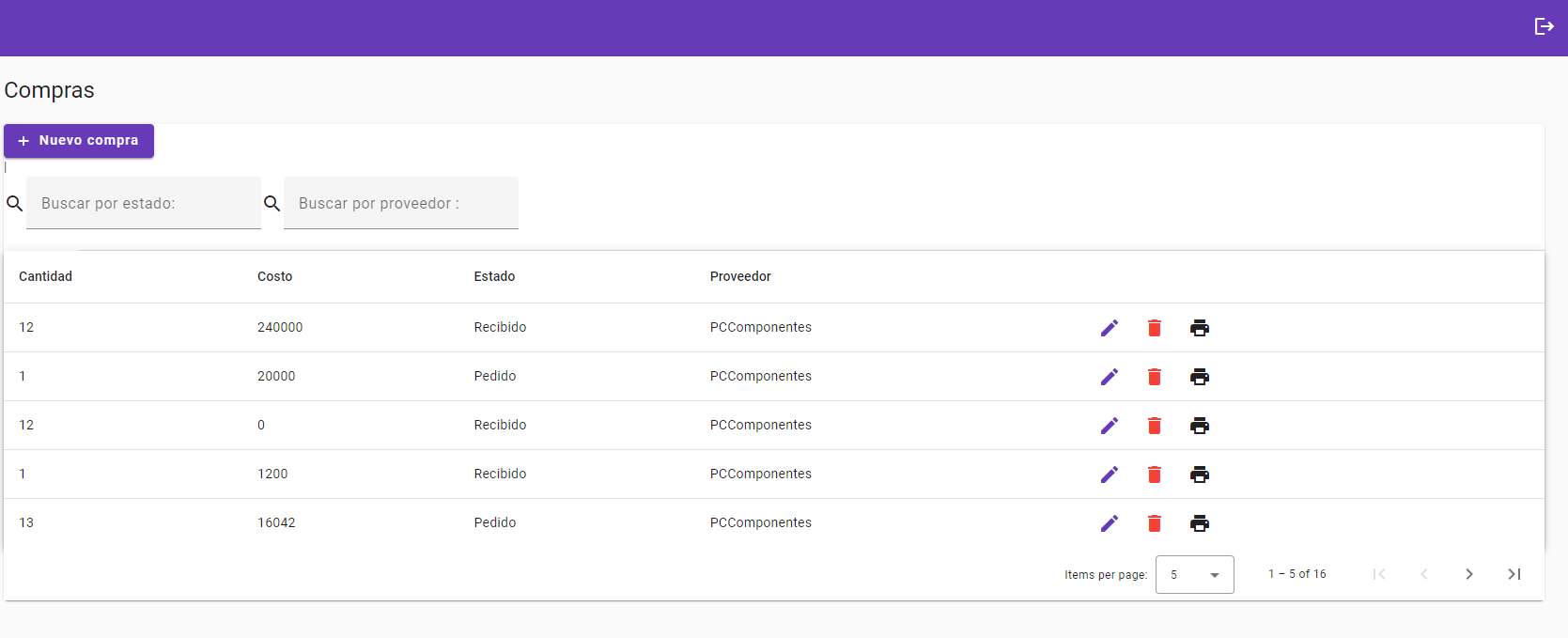
**RUTINA 5:Muestra los campos del formulario con los datos del registro seleccionado para actualizarlos.**

**RUTINA 6:Borra y actualiza el registro seleccionado en la tabla y la base de datos**

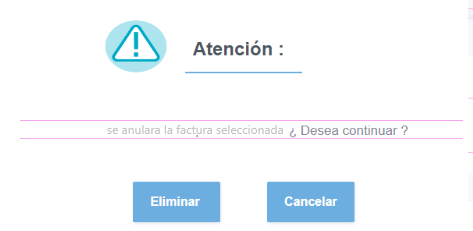
**RUTINA 7: Graba o actualiza los registros de ventas en la base de datos.**

**RUTINA 8: Imprime comprobante de la compra realizada**

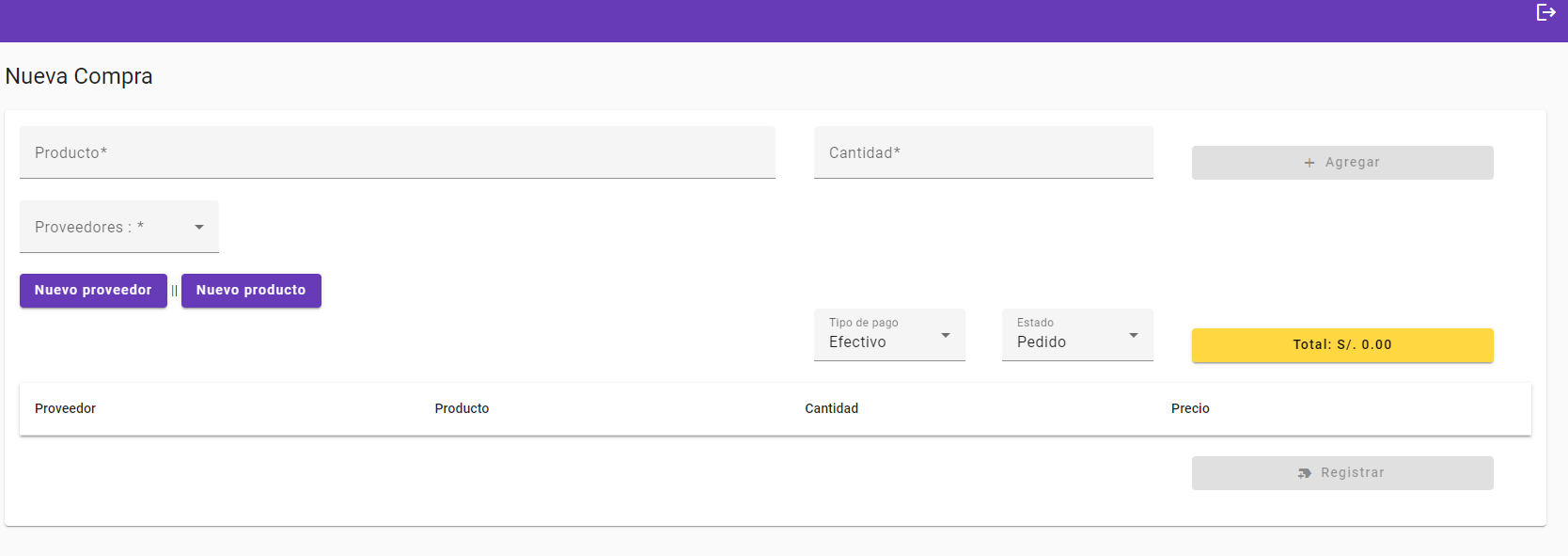
➔Pantallas: Pantalla Gestión compra

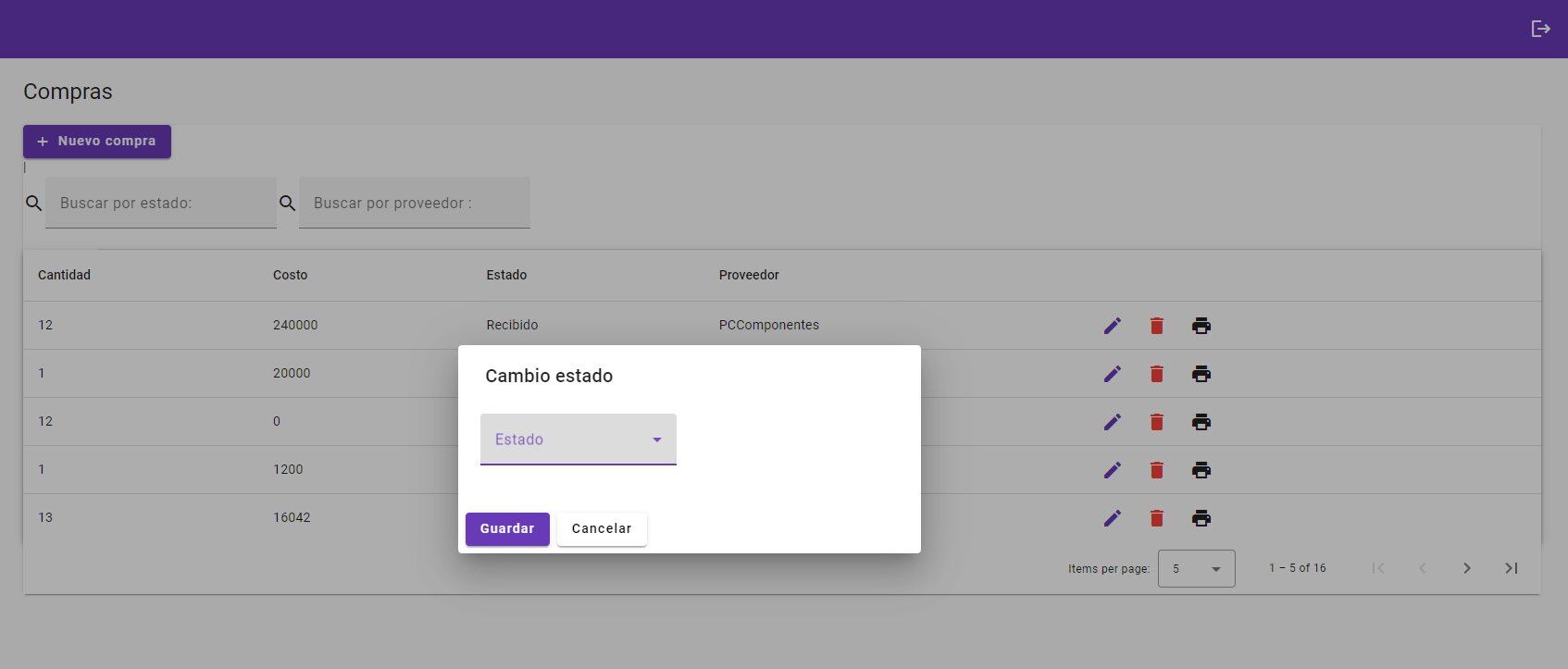


➔Pantallas: Pantalla Eliminar



➔Pantallas: Pantalla Alta - Modificación

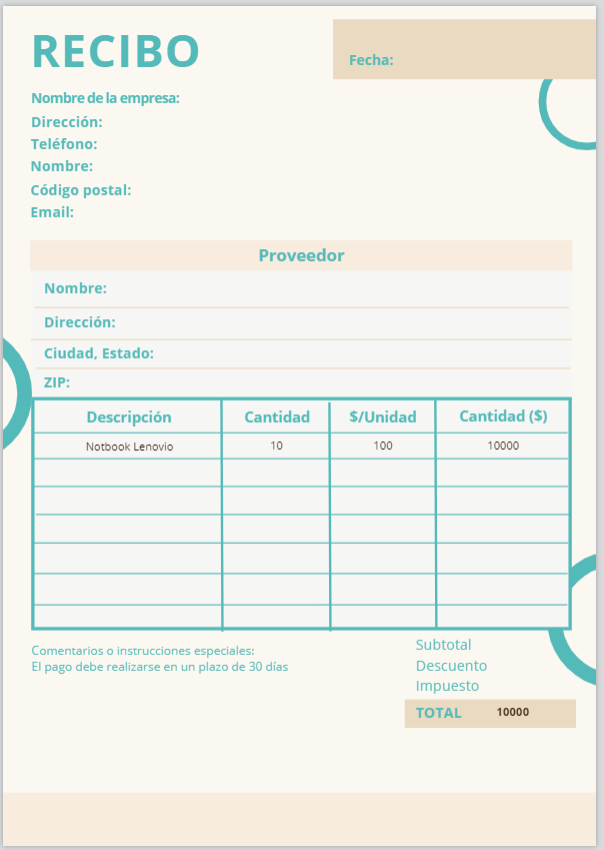




➔Pantallas: Error validación :

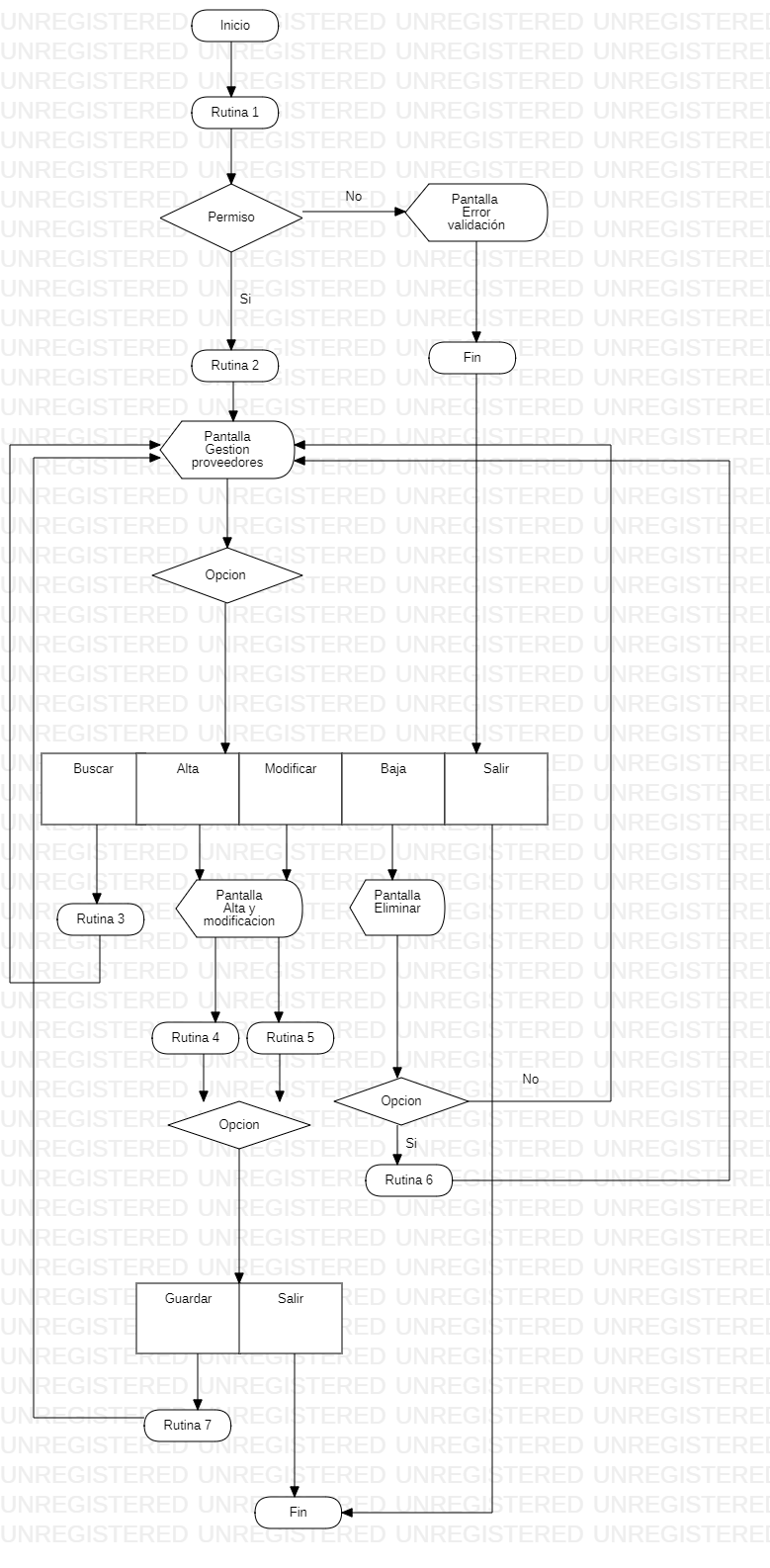


➔Pantallas: Salida Impresa



**Proceso ABM Proveedores**

FlowChart :Gestión  **Proveedores :**



**Rutinas ABM PROVEEDORES**

➔Rutinas:

**RUTINA 1: Verifica que el usuario tenga permiso para ingresar a dicho proceso.**

**RUTINA 2: Muestra una grilla con los datos de la tabla proveedores con sus respectivos botones de acciones.**

**RUTINA 3: Busca y selecciona un proveedor por su nombre.**

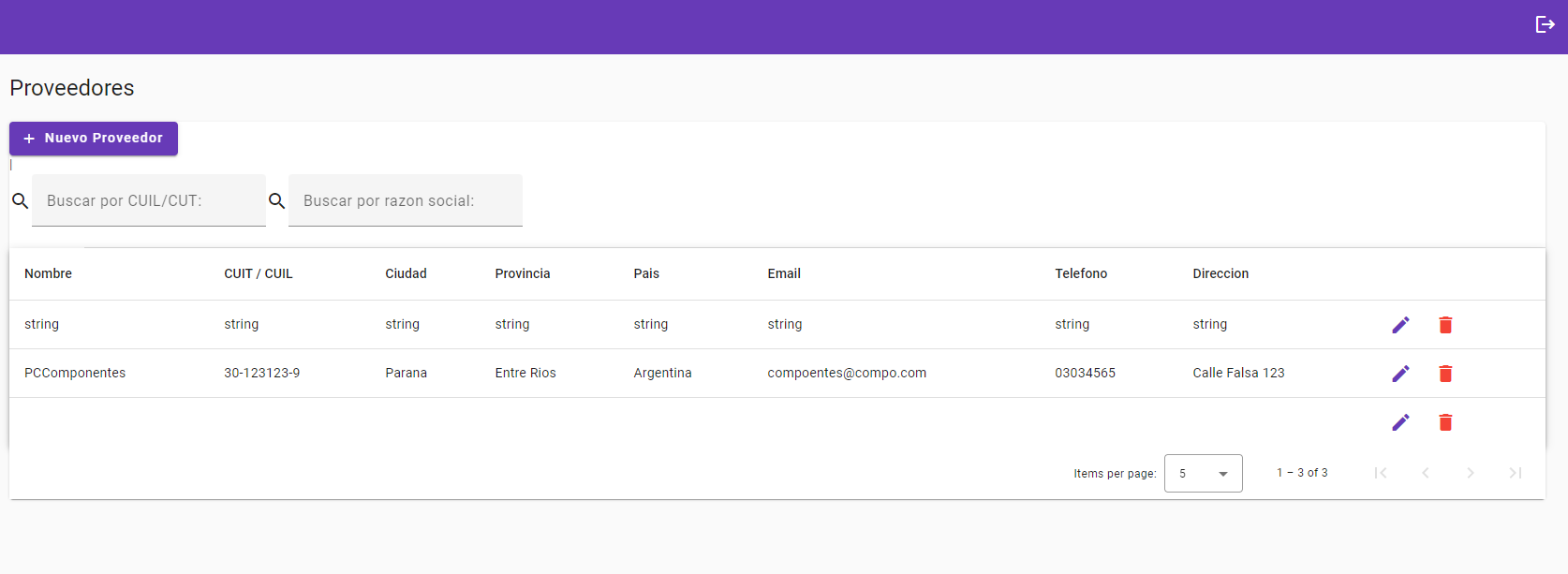
**RUTINA 4:Muestra un formulario vacío para el ingreso de datos del registro de proveedores.**

**RUTINA 5: Muestra los campos del formulario con los datos del registro seleccionado para actualizarlos.**

**RUTINA 6: Borra y actualiza el registro seleccionado en la tabla y la base de datos**

**RUTINA 7: Graba o actualiza los registros del usuario en la base de datos.**

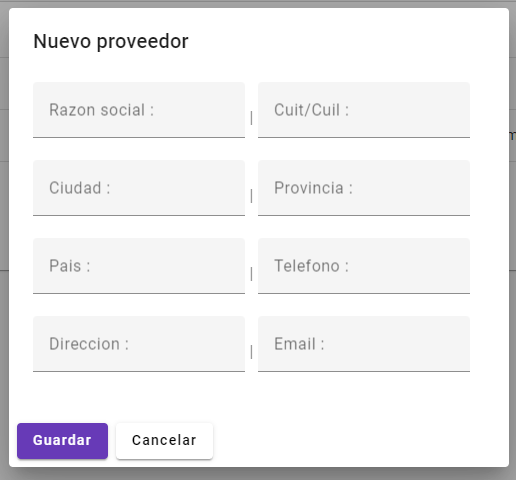
➔Pantallas: Pantalla Gestión Proveedores



➔Pantallas: Pantalla Eliminar



➔Pantallas: Pantalla Alta-Modificación

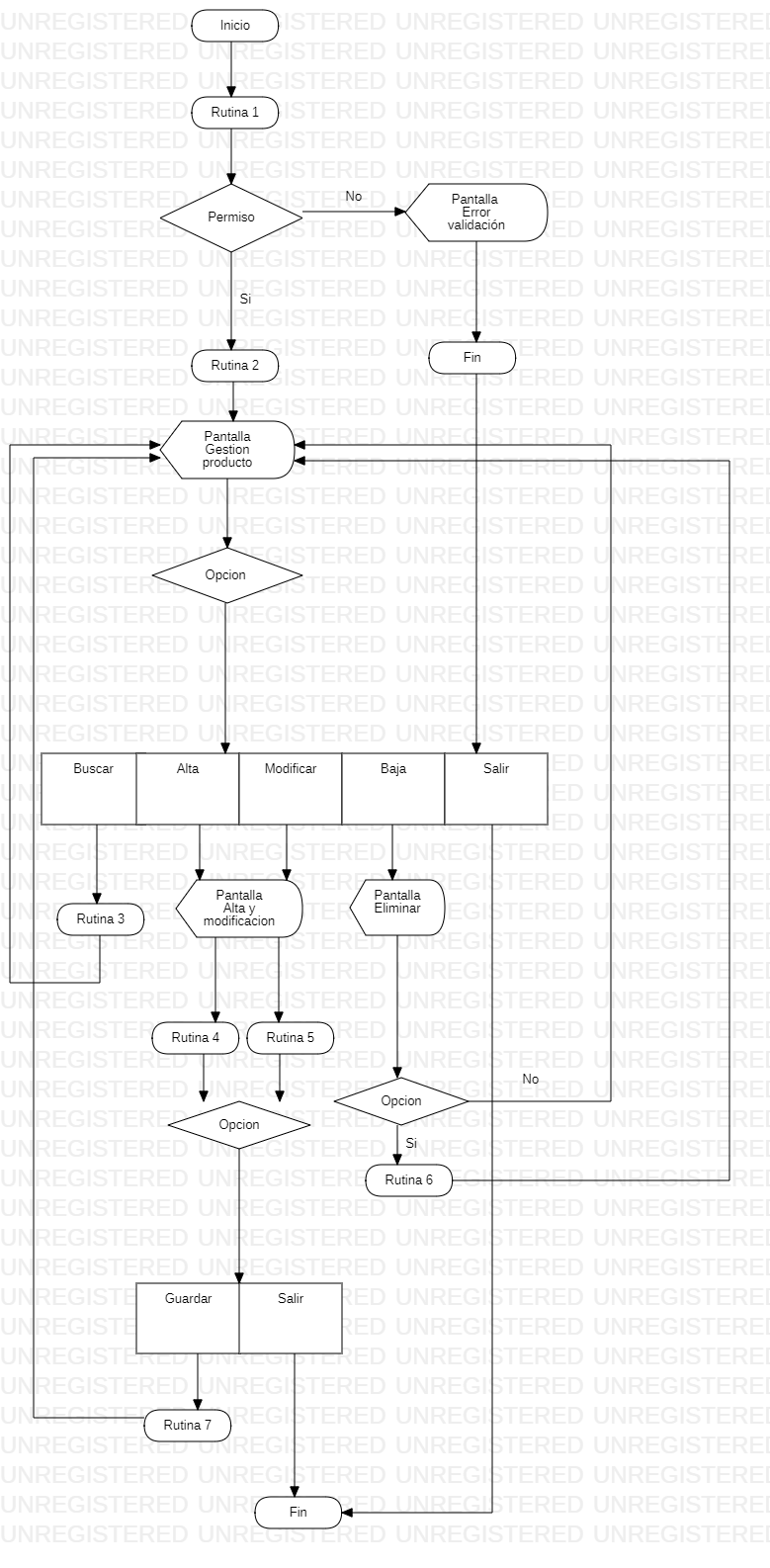
****

➔Pantallas: Error validación :



**Proceso ABM productos**

1. **FlowChart : Gestión productos**



➔Rutinas:

**RUTINA 1: Verifica que el usuario tenga permiso para ingresar a dicho proceso.**

**RUTINA 2: Muestra una grilla con los datos de la tabla producto y proveedor con sus respectivos botones de acciones.**

**RUTINA 3: Busca y selecciona un producto por su categoría o proveedor.**

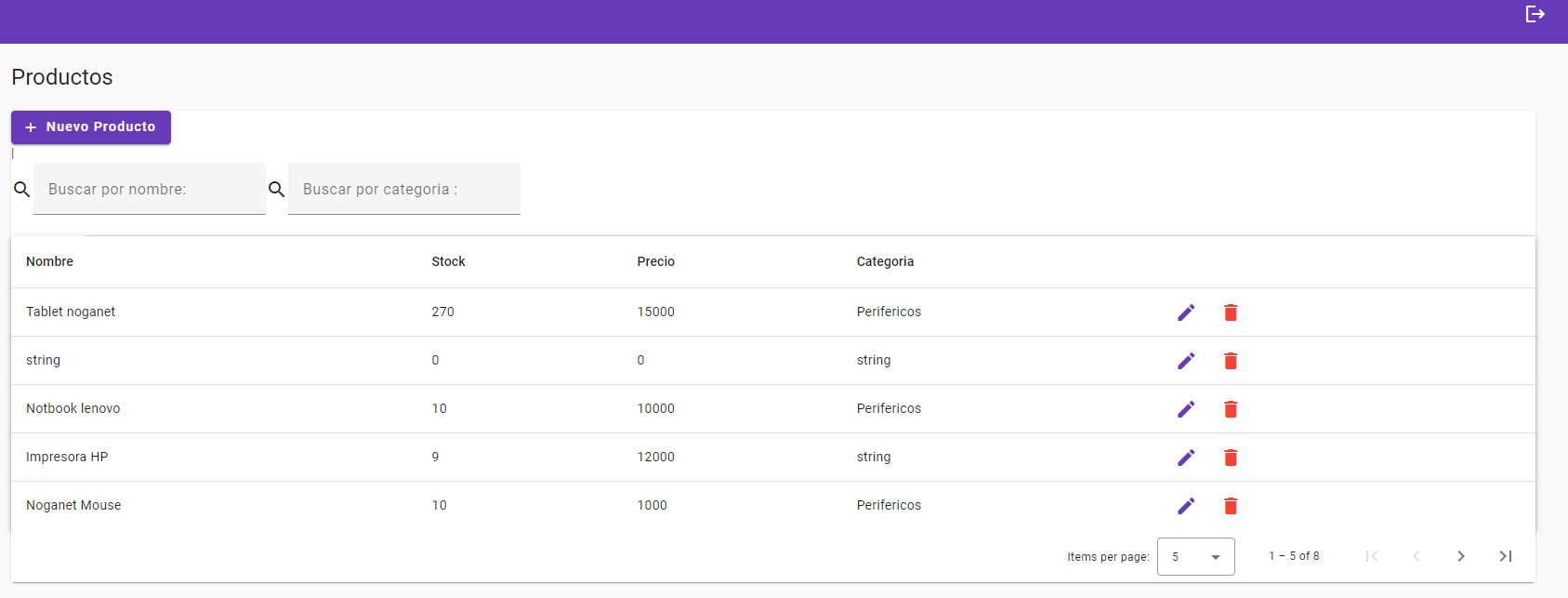
**RUTINA 4:Muestra un formulario vacío para el ingreso de datos del registro de producto.**

**RUTINA 5: Muestra los campos del formulario con los datos del registro seleccionado para actualizarlos.**

**RUTINA 6: Borra y actualiza el registro seleccionado en la tabla y la base de datos**

**RUTINA 7: Graba o actualiza los registros del producto en la base de datos.**

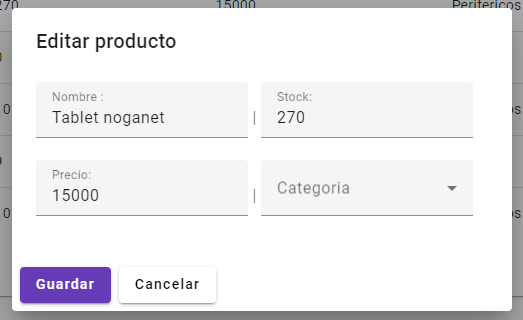
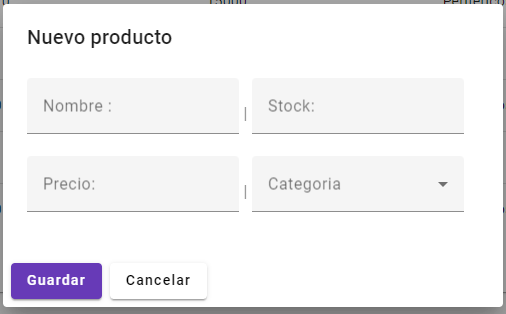
➔Pantallas: Pantalla Gestión Productos



➔Pantallas: Pantalla **Eliminar**



➔Pantallas: Pantalla **Alta- Modificación produtos**



➔Pantallas: Error validación :



**Base de Datos**

* + Esquema de Base de Datos

