

Pet Care

Sistema de Gestión para
Veterinarias

Lenguajes de última Generación

Profesor: Gonzalo Facundo Arigos

Grupo 5

Alumnos: Mauricio Damian
Carambia, Enzo Hernan Mansilla,
Marcelo David De Angelis, Maria
Sol García Frolik y María Belén La
Cava

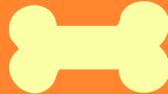




Pet Care

Sistema de Gestión para Veterinarias

Mauricio Damian Carambia, Enzo Hernan Mansilla, Marcelo David De Angelis, Maria Sol García Frolik y María Belén La Cava



Descripción del sistema

El sistema PetCare es una solución integral diseñada para la gestión eficiente de una veterinaria. Está construido con una arquitectura en capas que separa las responsabilidades en diferentes módulos: Interfaz de Usuario (UI), Lógica de Negocio (BLL), Acceso a Datos (DAL) y Entidades del Dominio (Entity). Esta estructura modular facilita el mantenimiento, la escalabilidad y la comprensión del sistema.

En la capa de entidades, el sistema define las principales clases del dominio: Cita, Cliente, Empleado, Factura, Mascota, Producto y Veterinario. Cada clase representa un objeto del sistema con sus atributos y relaciones. Por ejemplo, la clase Cita gestiona los datos relacionados con las citas programadas, incluyendo atributos como fecha, hora, veterinario y mascota asociados. La clase Cliente contiene la información personal de los clientes, como nombre, dirección, teléfono y correo electrónico. Del mismo modo, cada clase encapsula la lógica específica de sus datos.

La capa de acceso a datos (DAL) implementa la interacción con la base de datos a través de consultas SQL directas y manejo de conexiones utilizando SqlConnection. Esta capa contiene métodos CRUD (Crear, Leer, Actualizar, Eliminar) para todas las entidades, asegurando que los datos sean persistentes y consistentes. Las operaciones incluyen validaciones básicas para prevenir errores en las transacciones.

La lógica del negocio (BLL) es el núcleo funcional del sistema. Aquí se implementan las reglas y restricciones propias de la veterinaria. Por ejemplo, se valida que las citas sean programadas en horarios permitidos y con fechas futuras. Los datos del cliente deben ser completos y válidos, incluyendo formatos correctos de teléfono y correo electrónico. Además, las facturas requieren la validación de clientes, productos y cantidades válidas antes de su emisión. Esta capa también maneja transacciones utilizando TransactionScope para garantizar la atomicidad en operaciones complejas como la creación de múltiples citas o la modificación de datos relacionados.

La interfaz de usuario (UI) se implementa como una serie de formularios Windows Forms. Estos formularios permiten a los usuarios interactuar fácilmente con el sistema mediante operaciones CRUD para cada entidad. Por ejemplo, el formulario FrmCita permite a los usuarios gestionar citas, incluyendo la selección de mascotas y veterinarios asociados, mientras que FrmCliente permite registrar y editar la información de los clientes. Los formularios están diseñados para mostrar mensajes claros de error y éxito, facilitando la experiencia del usuario.

El flujo principal del sistema comienza con el inicio de sesión a través de FrmLogin, donde los empleados ingresan sus credenciales para acceder a las funcionalidades. Una vez dentro, el usuario puede gestionar clientes, mascotas, productos, veterinarios, facturas y citas. Cada operación está cuidadosamente diseñada para cumplir con las necesidades de una veterinaria, desde la programación de citas con validación de horarios hasta la emisión de facturas con cálculos automáticos de totales basados en productos y cantidades seleccionadas.

El sistema PetCare se destaca por su robustez y flexibilidad, cubriendo las principales necesidades de una veterinaria y asegurando un manejo eficiente de los datos. Su estructura modular no solo facilita el mantenimiento, sino que también permite agregar nuevas funcionalidades conforme a los requisitos futuros del negocio.

Diagrama de Base De Datos

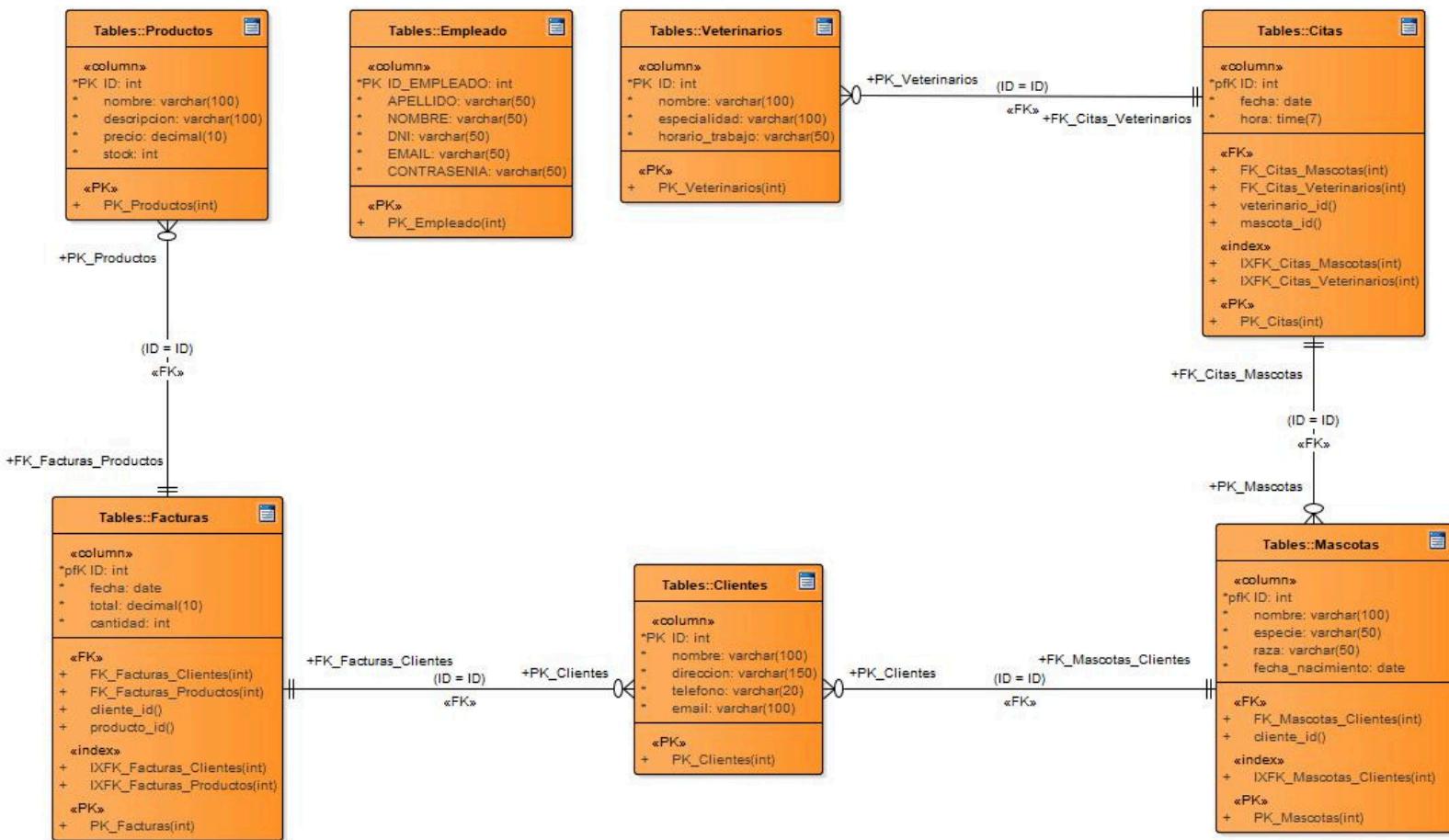
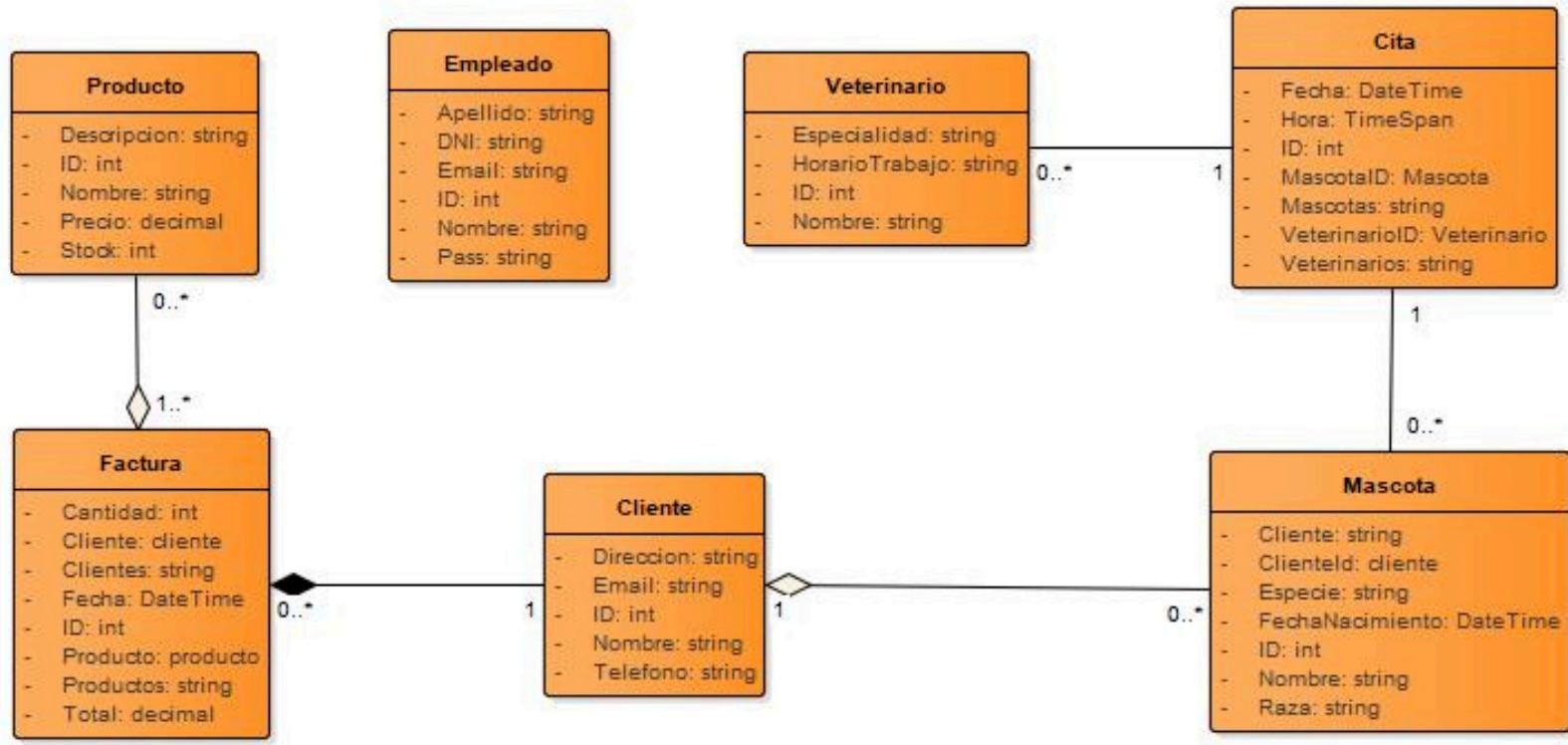


Diagrama de Clases



Manual de Usuario

Este manual está dirigido a los usuarios del sistema PetCare, incluidos:

- **Recepcionistas:** Para gestionar clientes, mascotas y citas.
- **Veterinarios:** Para consultar y administrar su agenda de trabajo.
- **Administradores:** Para supervisar las operaciones generales y gestionar el inventario de productos.

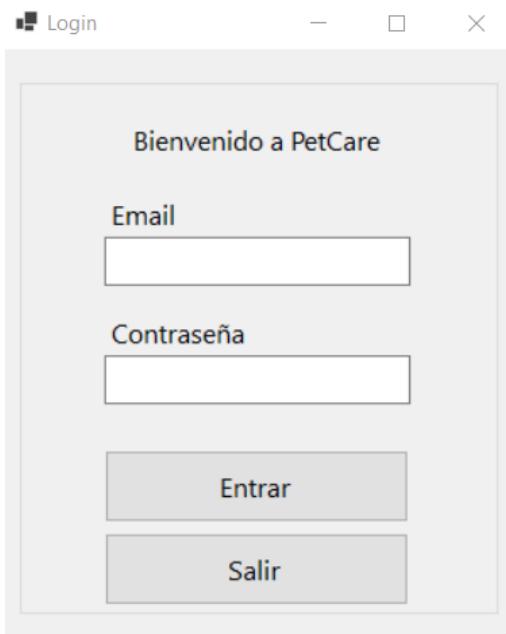
Inicio de Sesión

El sistema PetCare cuenta con una funcionalidad de inicio de sesión para garantizar que solo los empleados autorizados puedan acceder a las funcionalidades del sistema. Este proceso se implementa en el formulario FrmLogin de la capa de interfaz de usuario (UI), que interactúa con la capa de lógica de negocio (BLL) y la capa de acceso a datos (DAL) para validar las credenciales proporcionadas.

Procedimiento de Inicio de Sesión

1. Pantalla de Inicio de Sesión:

- Al iniciar el sistema, se muestra el formulario de login (FrmLogin), donde el usuario debe ingresar su dirección de correo electrónico y contraseña.
- Los campos requeridos son:
 - **Correo electrónico:** Representa la dirección registrada del empleado.
 - **Contraseña:** La clave de acceso asociada al empleado.



1. Validación de Credenciales:

- Al presionar el botón "Iniciar Sesión", el sistema valida los datos ingresados:
 - El formulario llama al método LoginEmpleado de la clase EmpleadoBusiness (capa BLL).
 - Este método verifica las credenciales contra la base de datos utilizando el método ObtenerEmpleadoPorEmail de la clase EmpleadoData (capa DAL).
- Si se encuentra un empleado con el correo electrónico proporcionado y la contraseña coincide:
 - El empleado se autentica correctamente y se devuelve su información.
- Si las credenciales son incorrectas:
 - Se muestra un mensaje de error: "**Email o contraseña incorrectos**".

2. Ingreso Exitoso:

- Cuando las credenciales son válidas:
 - El formulario guarda el empleado autenticado en la propiedad EmpleadoAutenticado.
 - Se cierra el formulario de login y se redirige al menú principal del sistema.
- Esta autenticación permite personalizar el acceso a ciertas funcionalidades según sea necesario.

3. Errores de Inicio de Sesión:

- Si el correo electrónico no está registrado o la contraseña no es correcta, se muestra un mensaje claro al usuario indicando el problema.
- Si ocurre un error en la base de datos durante el proceso, se lanza una excepción con un mensaje general de error.

Gestión de Clientes

El sistema PetCare incluye un módulo de **Gestión de Clientes**, implementado principalmente en el formulario FrmCliente de la capa de interfaz de usuario (UI). Este módulo permite registrar, consultar, modificar y eliminar información de los clientes de la veterinaria, asegurando un manejo eficiente y validado de los datos.

ID	Nombre	Direccion	Telefono
1	Rosales	Misiones 123	1231231231
4	Martinez	Posadas 123	1154564567
6	Garcia	Huergo 23	1154561235
7	Perez	Avellaneda 786	1141594234
8	Gonzalez	Husares 1234	1151641234
15	Peralta	San pedro 123	1163213214

El formulario interactúa con las capas de lógica de negocio (BLL) y acceso a datos (DAL) para procesar y almacenar la información en la base de datos.

Funciones del Módulo de Gestión de Clientes

1. Registro de Clientes

- El usuario puede registrar un nuevo cliente llenando los campos requeridos en el formulario.
- **Campos obligatorios:**
 - **Nombre:** Se requiere un nombre completo.
 - **Dirección:** Es obligatorio proporcionar una dirección válida.
 - **Teléfono:** Debe ser un número numérico con al menos 8 dígitos.
 - **Email:** Se valida el formato del correo electrónico (ejemplo: usuario@dominio.com).
- **Proceso Interno:**
 - Al presionar el botón "Guardar", se crea una instancia de la clase Cliente con los datos ingresados.
 - El método GuardarCliente de la clase ClienteBusiness (BLL) realiza las validaciones correspondientes y, si todo es correcto, llama al método GuardarCliente de la clase ClienteData (DAL) para almacenar el cliente en la base de datos.

Alta Cliente

Apellido y Nombre	<input type="text"/>
Teléfono	<input type="text"/>
Dirección	<input type="text"/>
Email	<input type="text"/>
<input type="button" value="Guardar"/>	
<input type="button" value="Agregar Borrador"/> <input type="button" value="Confirmar"/>	

2.Modificación de Clientes

- Permite modificar los datos de un cliente existente.
- **Proceso:**
 - El usuario selecciona un cliente de la lista para modificarlo. Los datos del cliente se cargan en los campos del formulario.
 - El usuario edita los datos deseados y presiona el botón "Modificar".
 - Se llama al método ModificarCliente de ClienteBusiness, que realiza las validaciones y actualiza los datos del cliente mediante el método correspondiente en ClienteData.

Modificar Cliente

ID	<input type="text" value="1"/>
Apellido y Nombre	<input type="text" value="Rosales"/>
Dirección	<input type="text" value="Misiones 123"/>
Teléfono	<input type="text" value="1231231231"/>
Email	<input type="text" value="rosales@gmail.com"/>
<input type="button" value="Modificar"/>	

3.Eliminación de Clientes

- Permite eliminar un cliente de la base de datos.
- **Proceso:**
 - El usuario selecciona un cliente de la tabla y presiona el botón "Eliminar".
 - El sistema verifica que el cliente existe llamando al método GetById de ClienteData.
 - Si el cliente existe, se elimina utilizando el método EliminarCliente.

Eliminar Cliente

Cliente	<input type="text" value="1"/>
<input type="button" value="Eliminar"/>	

Validaciones del Módulo

El sistema realiza varias validaciones en los datos del cliente para garantizar la integridad y consistencia de la información:

- El nombre no puede estar vacío.
- La dirección debe ser válida.
- El número de teléfono debe ser numérico y cumplir con un mínimo de 8 caracteres.
- El correo electrónico debe cumplir con un formato válido.
- No se puede modificar ni eliminar un cliente que no exista en la base de datos.

Flujo del Usuario en la Gestión de Clientes

1. **Registro:**
 - El usuario ingresa los datos del cliente en los campos del formulario y presiona "Guardar".
 - Si los datos son válidos, el cliente se almacena en la base de datos y aparece en la tabla.
2. **Consulta:**
 - El usuario presiona "Actualizar" para ver la lista completa de clientes registrados.
 - La tabla se llena automáticamente con los datos de la base de datos.
3. **Modificación:**
 - El usuario selecciona un cliente de la tabla y modifica sus datos en el formulario.
 - Al presionar "Modificar", los cambios se validan y actualizan en la base de datos.
4. **Eliminación:**
 - El usuario selecciona un cliente de la tabla y presiona "Eliminar".
 - El cliente se elimina si no está asociado a otras entidades dependientes.

Gestión de Mascotas

El módulo de **Gestión de Mascotas** en el sistema PetCare permite registrar, consultar, modificar y eliminar información sobre las mascotas asociadas a los clientes de la veterinaria. Este módulo se implementa principalmente en el formulario FrmMascota de la capa de interfaz de usuario (UI) y utiliza las capas de lógica de negocio (BLL) y acceso a datos (DAL) para procesar la información y almacenarla en la base de datos.

Alta Mascota

Nombre	<input type="text"/>
Raza	<input type="text"/>
Especie	<input type="text"/>
Cliente	<input type="text"/> Rosales
Fecha Nacimiento	sábado , 23 de noviembre

Modificar Mascota

Mascota	<input type="text"/> 15
Nombre	<input type="text"/> Firulais
Raza	<input type="text"/> Pitbull
Especie	<input type="text"/> Canino
Cliente	<input type="text"/> Rosales
Fecha Nacimiento	sábado , 23 de noviembre

Eliminar Mascota

Mascotas

ID	Nombre	Especie	Raza	FechaNacimiento	Cliente
15	Firulais	Canino	Pitbull	10/6/2020	Perez
17	Monchi	Felino	Siames	20/11/2024	Martinez
19	Bobby	Ave	Loro	8/1/2019	Rosales
20	Coqui	Canino	Pequines	15/6/2021	Gonzalez

Funciones del Módulo de Gestión de Mascotas

1. Registro de Mascotas

- Los usuarios pueden registrar una mascota asociándola a un cliente existente.
- **Campos obligatorios:**
 - **Nombre:** El nombre de la mascota.
 - **Especie:** Tipo de mascota (e.g., perro, gato).
 - **Raza:** Especificación de la raza de la mascota.
 - **Fecha de Nacimiento:** Fecha de nacimiento de la mascota.
 - **Cliente Asociado:** Seleccionado desde un combo box que lista los clientes registrados.
- **Proceso Interno:**
 - El usuario llena los datos requeridos en el formulario y presiona el botón "Guardar".
 - Se crea una instancia de la clase Mascota con los datos ingresados.
 - El método GuardarMascota de la clase MascotaBusiness realiza las validaciones necesarias y, si todo es correcto, guarda los datos en la base de datos utilizando el método correspondiente de MascotaData.

Alta Mascota

Nombre	<input type="text"/>
Raza	<input type="text"/>
Especie	<input type="text"/>
Cliente	<input type="text" value="Rosales"/>
Fecha Nacimiento	<input type="text" value="sábado , 23 de noviembre"/>
<input type="button" value="Agregar Borrador"/> <input type="button" value="Confirmar"/> <input type="button" value="Guardar"/>	

2.Modificación de Mascotas

- Permite modificar los datos de una mascota existente.
- **Proceso:**
 - El usuario selecciona una mascota de la tabla para cargar sus datos en los campos del formulario.
 - Se editan los campos deseados y se presiona el botón "Modificar".
 - El sistema valida los datos y utiliza el método ModificarMascota de MascotaBusiness para actualizar la información en la base de datos mediante MascotaData.

Modificar Mascota

Mascota	<input type="text" value="15"/>
Nombre	<input type="text" value="Firulais"/>
Raza	<input type="text" value="Pitbull"/>
Especie	<input type="text" value="Canino"/>
Cliente	<input type="text" value="Rosales"/>
Fecha Nacimiento	<input type="text" value="sábado , 23 de noviembre"/>
<input type="button" value="Modificar"/>	

3.Eliminación de Mascotas

- Permite eliminar una mascota de la base de datos.
- **Proceso:**
 - El usuario selecciona una mascota de la lista o ingresa su ID en el campo correspondiente.
 - Al presionar el botón "Eliminar", se verifica que la mascota existe mediante el método GetById de MascotaData.

- Si la mascota existe, se elimina junto con las citas asociadas utilizando el método EliminarMascota.

The image shows a user interface for deleting a pet. It consists of a light gray rectangular box with a thin black border. At the top left, the text "Eliminar Mascota" is displayed in a small, dark font. Below this, there is a horizontal layout. On the left, the word "Mascota" is followed by a rectangular input field. To the right of the input field is a large, prominent blue rectangular button with the word "Eliminar" in white. The overall design is clean and functional, typical of a software application's confirmation dialog.

Validaciones del Módulo

El sistema aplica varias validaciones para garantizar la consistencia de los datos:

- El nombre, especie y raza de la mascota no pueden estar vacíos.
- La fecha de nacimiento debe ser válida y no puede ser futura.
- Una mascota debe estar asociada a un cliente existente.
- No se puede eliminar una mascota que no exista en la base de datos.

Flujo del Usuario en la Gestión de Mascotas

1. Registro:

- El usuario ingresa los datos de la mascota y selecciona un cliente asociado desde el combo box.
- Presiona "Guardar" y, si los datos son válidos, la mascota se registra en la base de datos y se muestra en la tabla.

2. Consulta:

- El usuario presiona "Actualizar" para cargar todas las mascotas registradas.
- Los datos se muestran en una tabla donde se incluyen opciones para seleccionar y editar mascotas.

3. Modificación:

- El usuario selecciona una mascota de la tabla y modifica sus datos en los campos del formulario.
- Al presionar "Modificar", los cambios se validan y actualizan en la base de datos.

4. Eliminación:

- El usuario selecciona una mascota de la tabla o ingresa su ID y presiona "Eliminar".
- La mascota se elimina junto con sus citas asociadas, si las tiene.

Gestión de Veterinarios

El módulo de **Gestión de Veterinarios** en el sistema PetCare permite registrar, consultar, modificar y eliminar información sobre los veterinarios que trabajan en la clínica. Este módulo se implementa en el formulario FrmVeterinario de la capa de interfaz de usuario (UI) y utiliza las capas de lógica de negocio (BLL) y acceso a datos (DAL) para procesar y almacenar la información de manera eficiente.

Alta Veterinario

Apellido y Nombre

Especialidad

Horario

Agregar Borrador **Confirmar** **Guardar**

Veterinarios

	ID	Nombre	Especialidad	HorarioTrabajo
▶	9	Marcos	Radiólogo	7 a 10
	10	Juan	Cirujano	11 a 16
	11	Romina	Cardiología	10 a 14
	12	Rocío	Cardiología	12 a 17

Modificar Veterinario

Veterinario

Apellido y Nombre

Especialidad

Horario

Modificar

Eliminar Veterinario

Veterinario

Eliminar

Cerrar

Funciones del Módulo de Gestión de Veterinarios

1. Registro de Veterinarios

- Permite registrar un nuevo veterinario en el sistema, incluyendo información básica como su nombre, especialidad y horario de trabajo.
- **Campos Obligatorios:**
 - **Nombre:** El nombre completo del veterinario.
 - **Especialidad:** Área de especialización del veterinario.
 - **Horario de Trabajo:** Horario disponible para consultas y citas.
- **Proceso Interno:**
 - El usuario llena los datos en el formulario y presiona el botón "Guardar".
 - Se crea una instancia de la clase Veterinario con los datos ingresados.
 - El método GuardarVeterinario de la clase VeterinarioBusiness valida la información y, si es correcta, la guarda en la base de datos mediante el método correspondiente en VeterinarioData.

Alta Veterinario

Apellido y Nombre

Especialidad

Horario

Agregar Borrador **Confirmar** **Guardar**

2.Modificación de Veterinarios

- Permite actualizar la información de un veterinario existente.
- **Proceso:**
 - El usuario selecciona un veterinario de la lista para cargar sus datos en los campos del formulario.
 - Edita los campos deseados y presiona el botón "Modificar".
 - El método ModificarVeterinario de VeterinarioBusiness valida los datos y actualiza la información en la base de datos mediante VeterinarioData.

Modificar Veterinario	
Veterinario	9
Apellido y Nombre	Marcos
Especialidad	Radiologo
Horario	7 a 10
Modificar	

3.Eliminación de Veterinarios

- Permite eliminar un veterinario del sistema.
- **Proceso:**
 - El usuario selecciona un veterinario de la lista o ingresa su ID en el campo correspondiente.
 - Al presionar "Eliminar", se verifica que el veterinario existe mediante el método GetById de VeterinarioData.
 - Si existe, se elimina utilizando el método EliminarVeterinario.

Eliminar Veterinario	
Veterinario	9
Eliminar	

Validaciones del Módulo

El sistema implementa varias validaciones para garantizar la integridad de los datos:

- El nombre, especialidad y horario de trabajo no pueden estar vacíos.
- No se puede modificar o eliminar un veterinario que no exista en la base de datos.
- Antes de eliminar un veterinario, se eliminan todas las citas asociadas a él para evitar inconsistencias.

Flujo del Usuario en la Gestión de Veterinarios

1. Registro:

- El usuario ingresa los datos requeridos del veterinario en el formulario.
- Presiona "Guardar" y, si los datos son válidos, el veterinario se registra en la base de datos y se muestra en la tabla.

2. Consulta:

- El usuario presiona "Actualizar" para cargar la lista de veterinarios registrados.
- Los datos se muestran en un DataGridView donde pueden seleccionarse para modificaciones o eliminaciones.

3. Modificación:

- El usuario selecciona un veterinario de la lista para editar sus datos en el formulario.
- Presiona "Modificar" y los cambios se validan y actualizan en la base de datos.

4. Eliminación:

- El usuario selecciona un veterinario de la lista o ingresa su ID y presiona "Eliminar".
- Se eliminan las citas asociadas al veterinario y luego su registro.

Gestión de Citas

El módulo de **Gestión de Citas** en el sistema PetCare permite registrar, consultar, modificar y eliminar citas asociadas a veterinarios y mascotas. Esta funcionalidad está implementada en el formulario FrmCita de la capa de interfaz de usuario (UI) y utiliza las capas de lógica de negocio (BLL) y acceso a datos (DAL) para gestionar las citas y mantener los datos consistentes en la base de datos.

The screenshot shows the 'Citas' (Appointments) module of the PetCare application. At the top, there's a navigation bar with links for 'Administración', 'Servicio', and 'Salir'. The main area is divided into several sections:

- Alta Cita (New Appointment):** A form with dropdowns for 'Mascota' (Monchi), 'Veterinario' (Marcos), 'Hora' (19:58:12 p. m.), and 'Fecha' (sábado , 23 de noviembre). It includes buttons for 'Agregar Borrador' (Add Draft), 'Confirmar' (Confirm), and 'Guardar' (Save).
- Citas (Appointments):** A DataGridView displaying a list of appointments. The columns are: ID, Fecha, Hora, Veterinarios, and Mascotas. The data is as follows:

ID	Fecha	Hora	Veterinarios	Mascotas
13	28/11/2024	11:47:21.9140000	Marcos	Monchi
21	30/11/2024	12:50:55.6550000	Juan	Firulaís
22	23/11/2024	14:52:05.9709782	Marcos	Monchi
23	7/12/2024	10:52:05.9710000	Romina	Coqui

- Modificar Cita (Edit Appointment):** A form for editing an appointment. It shows the selected appointment (ID 13) and fields for 'Mascota' (Monchi), 'Veterinario' (Marcos), 'Hora' (11:47:21 a. m.), and 'Fecha' (jueves , 28 de noviembre). A 'Modificar' (Modify) button is present.
- Eliminar Cita (Delete Appointment):** A form for deleting an appointment. It shows the selected appointment (ID 13) and buttons for 'Eliminar' (Delete) and 'Cerrar' (Close).

Funciones del Módulo de Gestión de Citas

1. Registro de Citas

- Permite registrar una cita nueva asociada a un veterinario y una mascota.
- **Campos Obligatorios:**
 - **Fecha:** Debe ser una fecha futura.
 - **Hora:** Debe estar dentro del rango permitido de atención (07:00 a 19:00).
 - **Veterinario:** Seleccionado desde un combo box que lista los veterinarios disponibles.
 - **Mascota:** Seleccionada desde un combo box que lista las mascotas registradas.
- **Proceso Interno:**
 - El usuario ingresa los datos requeridos en el formulario y presiona "Guardar".
 - Se crea una instancia de la clase Cita con los datos ingresados.
 - El método GuardarCita de la clase CitaBusiness valida los datos:
 - La fecha debe ser posterior al día actual.
 - La hora debe estar dentro del rango permitido.
 - La cita debe estar asociada a un veterinario y una mascota existentes.
 - Si las validaciones son exitosas, la cita se almacena en la base de datos utilizando el método GuardarCita de la clase CitaData.

The screenshot shows a user interface for creating a new appointment. The title 'Alta Cita' is at the top. There are four input fields: 'Mascota' (dropdown, selected 'Monchi'), 'Veterinario' (dropdown, selected 'Marcos'), 'Hora' (time picker, set to '19:58:12 p.m.'), and 'Fecha' (date picker, set to 'sábado , 23 de noviemb'). At the bottom are three buttons: 'Agregar Borrador', 'Confirmar' (highlighted in orange), and 'Guardar'.

2. Modificación de Citas

- Permite actualizar los datos de una cita existente.
- **Proceso:**
 - El usuario selecciona una cita de la tabla para cargar sus datos en los campos del formulario.
 - Los datos son editados y el usuario presiona "Modificar".
 - El método ModificarCita de CitaBusiness valida los nuevos datos y, si son correctos, actualiza la cita en la base de datos mediante CitaData.

Modificar Cita

Cita	13
Mascota	Monchi
Veterinario	Marcos
Hora	11:47:21 a. m.
Fecha	jueves , 28 de noviemb

Modificar

3.Eliminación de Citas

- Permite eliminar una cita del sistema.
- **Proceso:**
 - El usuario selecciona una cita de la tabla o ingresa su ID en el campo correspondiente.
 - Al presionar "Eliminar", se verifica que la cita existe mediante el método GetById de CitaData.
 - Si la cita existe, se elimina utilizando el método EliminarCita.

Eliminar Cita

Cita	13
------	----

Eliminar

Validaciones del Módulo

El sistema realiza varias validaciones para garantizar la consistencia y corrección de los datos de las citas:

- **Fecha:** La fecha de la cita debe ser posterior al día actual.
- **Hora:** La hora de la cita debe estar entre las 07:00 y las 19:00.
- **Veterinario:** Se debe seleccionar un veterinario válido.
- **Mascota:** Se debe seleccionar una mascota válida.
- **Existencia:** No se puede modificar o eliminar una cita que no exista.

Flujo del Usuario en la Gestión de Citas

1. Registro:

- El usuario ingresa los datos de la cita, selecciona un veterinario y una mascota desde los combo boxes, y presiona "Guardar".
- Si los datos son válidos, la cita se registra en la base de datos y aparece en la tabla.

2. Consulta:

- El usuario presiona "Actualizar" para cargar todas las citas registradas.
- Los datos se muestran en un DataGridView donde pueden seleccionarse para modificaciones o eliminaciones.

3. Modificación:

- El usuario selecciona una cita de la tabla para editar sus datos en el formulario.
- Presiona "Modificar" y los cambios se validan y actualizan en la base de datos.

4. Eliminación:

- El usuario selecciona una cita de la tabla o ingresa su ID y presiona "Eliminar".
- La cita se elimina de la base de datos si existe.

Gestión de Productos

El módulo de **Gestión de Productos** en el sistema PetCare permite registrar, consultar, modificar y eliminar productos relacionados con la veterinaria, como medicamentos, alimentos y accesorios. Este módulo está implementado en el formulario FrmProducto de la capa de interfaz de usuario (UI) y utiliza las capas de lógica de negocio (BLL) y acceso a datos (DAL) para manejar la información de los productos y mantener los datos actualizados en la base de datos.

The screenshot displays four panels of the PetCare application interface:

- Alta Producto:** A form for adding new products. It contains fields for Nombre (Name), Descripción (Description), Precio (Price), and Stock. Below the fields are buttons for Agregar (Add), Borrador (Draft), Confirmar (Confirm), and Guardar (Save).
- Modificar Producto:** A form for modifying existing products. It shows a list of products with ID 3 selected. Fields include Nombre (Name), Descripción (Description), Precio (Price), and Stock. A button labeled Modificar (Modify) is present.
- Productos:** A DataGridView displaying a list of products. The columns are ID, Nombre, Descripción, Precio, and Stock. The data is as follows:

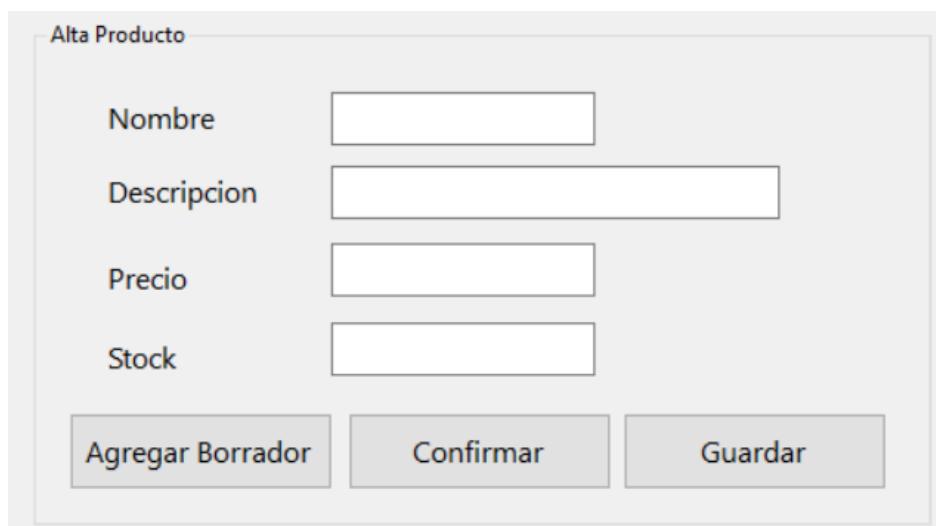
ID	Nombre	Descripción	Precio	Stock
3	Juguete	Juguete en forma de h...	2000,00	100
4	Comida	Comida de perro	10000,00	50
6	Collar	Collar con nombre	5000,00	20
7	Hueso plastico	Hueso de juguete	10000,00	50
9	Bolso	Bolso para gatos	25000,00	10
10	Pipeta	Pipeta anti pulgas	2000,00	100

- Eliminar Producto:** A form for deleting products. It shows a list of products with ID 3 selected. Buttons for Eliminar (Delete) and Cerrar (Close) are present.

Funciones del Módulo de Gestión de Productos

1. Registro de Productos

- Permite registrar un nuevo producto en el sistema con información completa sobre su descripción, precio y stock.
- **Campos Obligatorios:**
 - **Nombre:** Nombre del producto.
 - **Descripción:** Detalles del producto.
 - **Precio:** Precio unitario (debe ser un valor positivo).
 - **Stock:** Cantidad disponible (debe ser un número entero positivo).
- **Proceso Interno:**
 - El usuario llena los campos requeridos en el formulario y presiona "Guardar".
 - Se crea una instancia de la clase Producto con los datos ingresados.
 - El método GuardarProductos de la clase ProductoBusiness valida la información y, si es correcta, guarda los datos en la base de datos utilizando el método GuardarProducto de la clase ProductoData.



El formulario se titula "Alta Producto". Contiene cuatro campos de texto para "Nombre", "Descripción", "Precio" y "Stock". Abajo de los campos hay tres botones: "Agregar Borrador", "Confirmar" y "Guardar".

2. Modificación de Productos

- Permite actualizar la información de un producto existente.
- **Proceso:**
 - El usuario selecciona un producto de la tabla para cargar sus datos en los campos del formulario.
 - Los datos son editados y se presiona el botón "Modificar".
 - El método ModificarProducto de ProductoBusiness valida los nuevos datos y, si son correctos, actualiza el producto en la base de datos mediante ProductoData.

Modificar Producto

Producto	<input type="text" value="3"/>
Nombre	<input type="text" value="Juguete"/>
Descripción	<input type="text" value="Juguete en forma de hueso"/>
Precio	<input type="text" value="2000,00"/>
Stock	<input type="text" value="100"/>
<input type="button" value="Modificar"/>	

3. Eliminación de Productos

- Permite eliminar un producto del sistema.
- **Proceso:**
 - El usuario selecciona un producto de la tabla o ingresa su ID en el campo correspondiente.
 - Al presionar "Eliminar", se verifica que el producto existe mediante el método GetById de ProductoData.
 - Si el producto existe, se elimina utilizando el método EliminarProducto.

Eliminar Producto

Producto	<input type="text" value="3"/>	<input type="button" value="Eliminar"/>
----------	--------------------------------	---

Validaciones del Módulo

El sistema asegura la consistencia de los datos a través de varias validaciones:

- El nombre y la descripción no pueden estar vacíos.
- El precio debe ser un valor numérico positivo.
- El stock debe ser un número entero positivo.
- No se puede modificar o eliminar un producto que no exista en la base de datos.

Flujo del Usuario en la Gestión de Productos

1. **Registro:**
 - El usuario ingresa los datos del producto en el formulario y presiona "Guardar".
 - Si los datos son válidos, el producto se registra en la base de datos y aparece en la tabla.
2. **Consulta:**
 - Al abrir el formulario o presionar "Actualizar", se carga la lista de productos registrados.
 - Los datos se muestran en un DataGridView donde pueden seleccionarse para modificaciones o eliminaciones.

1. Modificación:

- El usuario selecciona un producto de la tabla y edita sus datos en el formulario.
- Presiona "Modificar" y los cambios se validan y actualizan en la base de datos.

2. Eliminación:

- El usuario selecciona un producto de la tabla o ingresa su ID y presiona "Eliminar".
- El producto se elimina de la base de datos si existe.

Gestión de Facturas

El módulo de **Gestión de Facturas** en el sistema PetCare permite registrar, consultar, modificar y eliminar facturas asociadas a los clientes y productos de la veterinaria. Este módulo está implementado en el formulario FrmFactura de la capa de interfaz de usuario (UI) y utiliza las capas de lógica de negocio (BLL) y acceso a datos (DAL) para manejar las operaciones de facturación y mantener los datos consistentes en la base de datos.

The screenshot shows the PetCare application interface for managing invoices. At the top, there is a navigation bar with icons for Home, Admin, Service, and Exit. The main area is divided into two main sections: 'Cargar factura' (Load Invoice) on the left and a 'Facturas' (Invoices) grid on the right.

Cargar factura: This section contains fields for 'Fecha Emisión' (Emission Date), 'Producto' (Product), 'Cliente' (Client), 'Cantidad' (Quantity), and a 'Total:' field showing '----'. It also features a 'Crear Factura' (Create Invoice) button.

Facturas: This section displays a grid of invoice data with columns: ID, Fecha (Date), Total, Clientes (Clients), Productos (Products), and Cantidad (Quantity). The data is as follows:

ID	Fecha	Total	Clientes	Productos	Cantidad
1	21/11/2024	50000,00	Martinez	Comida	5
2	22/11/2024	30000,00	Rosales	Hueso plastico	3
4	22/11/2024	50000,00	Perez	Collar	10
5	22/11/2024	4000,00	Rosales	Juguete	2

Editar factura: This section contains fields for 'Id Fac. a Modificar' (Invoice ID to Modify), 'Producto' (Product), 'Cliente' (Client), and 'Cantidad' (Quantity). It features a 'Modificar Factura' (Modify Invoice) button.

Buttons at the bottom: 'Eliminar factura' (Delete Invoice).

Funciones del Módulo de Gestión de Facturas

1. Registro de Facturas

- Permite registrar una nueva factura asociando un cliente y uno o más productos con sus cantidades.

○ Campos Obligatorios:

- **Cliente:** Seleccionado desde un combo box que lista los clientes registrados.
- **Producto:** Seleccionado desde un combo box que lista los productos disponibles.
- **Cantidad:** La cantidad del producto que se desea incluir en la factura.

- **Fecha:** Fecha en la que se genera la factura.
- **Proceso Interno:**
 - El usuario llena los datos requeridos en el formulario y presiona "Guardar".
 - Se crea una instancia de la clase Factura con los datos ingresados.
 - El método agregarFactura de la clase FacturaBusiness valida la información:
 - El cliente y el producto deben ser válidos y existir en la base de datos.
 - La cantidad debe ser mayor a cero.
 - La fecha debe ser válida (generalmente la fecha actual).
 - El stock del producto debe ser suficiente para la cantidad solicitada.
 - Si las validaciones son exitosas, la factura se almacena en la base de datos utilizando el método GuardarFactura de FacturaData.

Cargar factura

Fecha Emision	sábado , 23 de noviembre
Producto	Juguete
Cliente	Rosales
Cantidad	<input type="text"/>
Total:	-----
Crear Factura	

2.Modificación de Facturas

- Permite actualizar los datos de una factura existente.
- **Proceso:**
 - El usuario selecciona una factura de la tabla para cargar sus datos en los campos del formulario.
 - Edita los campos deseados y presiona el botón "Modificar".
 - El método modificarFactura de FacturaBusiness valida los nuevos datos y, si son correctos, actualiza la factura en la base de datos mediante FacturaData.

Editar factura

Id Fac. a Modificar	<input type="text" value="1"/>
Producto	<input type="text" value="Comida"/>
Cliente	<input type="text" value="Martinez"/>
Cantidad	<input type="text" value="5"/>
Modificar Factura	

3. Eliminación de Facturas

- Permite eliminar una factura del sistema.
- **Proceso:**
 - El usuario selecciona una factura de la tabla o ingresa su ID en el campo correspondiente.
 - Al presionar "Eliminar", se verifica que la factura existe mediante el método ObtenerPorId de FacturaData.
 - Si la factura existe, se elimina utilizando el método EliminarFactura.

Eliminar factura

Id Factura a Eliminar.	<input type="text"/>	Eliminar Factura
------------------------	----------------------	-------------------------

Validaciones del Módulo

El sistema realiza varias validaciones para garantizar la consistencia de los datos:

- **Cliente y Producto:** Deben ser seleccionados de los combo boxes y existir en la base de datos.
- **Cantidad:** Debe ser mayor a cero.
- **Stock de Productos:** La cantidad solicitada debe estar disponible en el inventario.
- **Fecha:** Generalmente la fecha debe ser la actual para reflejar operaciones del día.
- **Existencia:** No se puede modificar o eliminar una factura que no exista en la base de datos.

Cálculo de Totales

El total de una factura se calcula automáticamente multiplicando el precio del producto por la cantidad solicitada. Este cálculo se realiza en la lógica de negocio (BLL) antes de guardar o actualizar la factura, asegurando que los valores almacenados en la base de datos sean correctos.

Flujo del Usuario en la Gestión de Facturas

1. Registro:

- El usuario selecciona un cliente y un producto, ingresa la cantidad deseada y la fecha.
- Presiona "Guardar" y, si los datos son válidos, la factura se registra en la base de datos y aparece en la tabla.

2. Consulta:

- Al abrir el formulario o presionar "Actualizar", se carga la lista de facturas registradas.
- Los datos se muestran en un DataGridView donde pueden seleccionarse para modificaciones o eliminaciones.

3. Modificación:

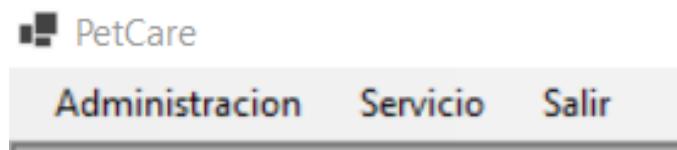
- El usuario selecciona una factura de la tabla para editar sus datos en el formulario.
- Presiona "Modificar" y los cambios se validan y actualizan en la base de datos.

4. Eliminación:

- El usuario selecciona una factura de la tabla o ingresa su ID y presiona "Eliminar".
- La factura se elimina de la base de datos si existe.

Cierre de Sesión y Salida del Sistema

El sistema PetCare incluye una funcionalidad básica para cerrar la sesión actual y salir del sistema de manera segura. Estas opciones están diseñadas para garantizar la protección de los datos y permitir una gestión adecuada de la sesión de los usuarios.



Cierre de Sesión

El cierre de sesión está implementado de forma manual a través de la interacción con el formulario principal del sistema. Una vez que un usuario termina de utilizar el sistema, tiene la opción de salir cerrando el formulario correspondiente.

Flujo del Cierre de Sesión:

1. **Desde el Formulario Principal:**
 - Después de completar las tareas, el usuario puede seleccionar la opción de cerrar o salir del formulario en la interfaz principal del sistema.
 - Esto implica cerrar cualquier ventana abierta y devolver el sistema al estado inicial.
2. **Implementación Técnica:**
 - En el código, el cierre del formulario implica la finalización de la ejecución del programa si se encuentra en el formulario principal.
 - No hay un método explícito de "cierre de sesión", pero al salir del sistema, cualquier información temporal sobre la sesión activa (e.g., empleado autenticado) deja de estar disponible.

Consideraciones:

- Para mayor seguridad, se recomienda que los usuarios siempre cierren sesión (cerrando la aplicación) al finalizar su trabajo.
- Actualmente, no se gestiona un cierre de sesión que permita cambiar de usuario sin cerrar el programa completo.

Salida del Sistema

La salida del sistema está diseñada para garantizar que los recursos utilizados se liberan correctamente y que no queden procesos en ejecución que puedan causar problemas.

Flujo para Salir del Sistema:

1. **Desde el Formulario de Login:**
 - Si el usuario está en la pantalla de inicio de sesión (FrmLogin), puede presionar el botón "Salir".
 - Al hacerlo, el método Application.Exit() se ejecuta, finalizando el programa.
2. **Desde el Formulario Principal o Secundario:**
 - Los formularios secundarios (FrmCita, FrmCliente, etc.) incluyen botones para cerrar cada ventana, devolviendo el control al formulario principal.
 - Al cerrar el formulario principal, el sistema finaliza automáticamente.

