



UNIVERSIDAD
CATÓLICA
BOLIVIANA
COCHABAMBA

Departamento de Ingeniería y Ciencias Exactas
Carrera de Ingeniería de Sistemas

GESTIÓN DE MASCOTAS EN UNA VETERINARIA

PROYECTO - TECNOLOGIAS WEB I

Estudiante:

Alison Alejandra Montaña Arias

Cochabamba – Bolivia

Octubre, 2025

SISTEMA VETERINARIA (API)

El principal objetivo del **Sistema de Gestión Veterinaria** es optimizar la administración de dueños, mascotas, citas médicas y servicios ofrecidos por la clínica.

El enfoque principal está en aplicar correctamente los conceptos aprendidos: arquitectura en capas, uso de Entity Framework Core, DTOs, mapeo con AutoMapper, validaciones con FluentValidation, manejo de excepciones y respuesta estandarizada con ApiResponse.

1. Casos de uso Identificados

Los **casos de uso principales** del sistema, ordenados por prioridad, son:

- Caso de uso 1: Registrar dueño y mascota asociada

Descripción: Permite registrar un nuevo dueño en el sistema y asociar una o más mascotas.

Validaciones:

- El número de teléfono del dueño debe ser único.
- Todos los campos (nombre, dirección, teléfono) son obligatorios.
- No se pueden registrar mascotas sin un dueño válido.

Acciones:

- Verificar si el dueño existe.
- Crear el registro del dueño y sus mascotas asociadas.
- Retornar el nuevo registro con estado HTTP **201 Created**.

- Caso de uso 2: Agendar cita veterinaria

Descripción: Registra una nueva cita para una mascota con un veterinario y servicio específico.

Validaciones:

- La mascota debe existir.
- La fecha y hora deben ser posteriores a la actual.
- El estado inicial debe ser siempre "Pendiente".

Acciones:

- Crear la cita con estado inicial “Pendiente”.
- Registrar la relación entre mascota, veterinario y servicio.
- Retornar mensaje de confirmación con código **201 Created**.

- **Caso de uso 3: Generar reporte de citas pendientes**

Descripción: Permite obtener un listado de todas las citas con estado “Pendiente”, ordenadas por fecha.

Validaciones:

- Mostrar solo citas pendientes.
- Incluir información del dueño, mascota y veterinario.

Acciones:

- Consultar las citas filtradas por estado y fecha.
- Devolver el listado en formato JSON.
- Retornar código **200 OK**.

- **Caso de uso 4: Actualizar estado de cita**

Descripción: Permite cambiar el estado de una cita (Pendiente a Atendida / Cancelada).

Validaciones:

- La cita debe existir.
- Solo se permiten los estados: “Pendiente”, “Atendida” y “Cancelada”.

Acciones:

- Buscar la cita.
- Actualizar su estado.
- Retornar mensaje de confirmación con código **200 OK**.

- **Caso de uso 5: Listar mascotas con su dueño**

Descripción: Muestra todas las mascotas registradas junto con la información básica de sus dueños.

Validaciones:

- Mostrar solo mascotas activas (si aplica baja lógica).
- Permitir filtrar por nombre o especie.

Acciones:

- Realizar consulta relacional entre Mascotas y Dueños.
- Retornar lista de resultados en formato JSON.

- **Caso de uso 6: Registrar servicio veterinario**

Descripción: Registra los distintos servicios que ofrece la clínica (consulta, vacunación, cirugía, etc.).

Validaciones:

- El nombre del servicio no puede repetirse.
- Todos los campos son obligatorios.

Acciones:

- Crear el registro del servicio.
- Devolver resultado HTTP **201 Created**.

- **Caso de uso 7: Registrar veterinario**

Descripción: Permite dar de alta a un veterinario con sus datos personales y especialidad.

Validaciones:

- El correo debe ser único.
- Todos los campos son requeridos.

Acciones:

- Registrar veterinario con su especialidad.
- Retornar respuesta HTTP **201 Created**.

2. Entidad-Relación de la base de datos

La base de datos se modeló utilizando **Entity Framework Core** bajo el enfoque *Code First*.

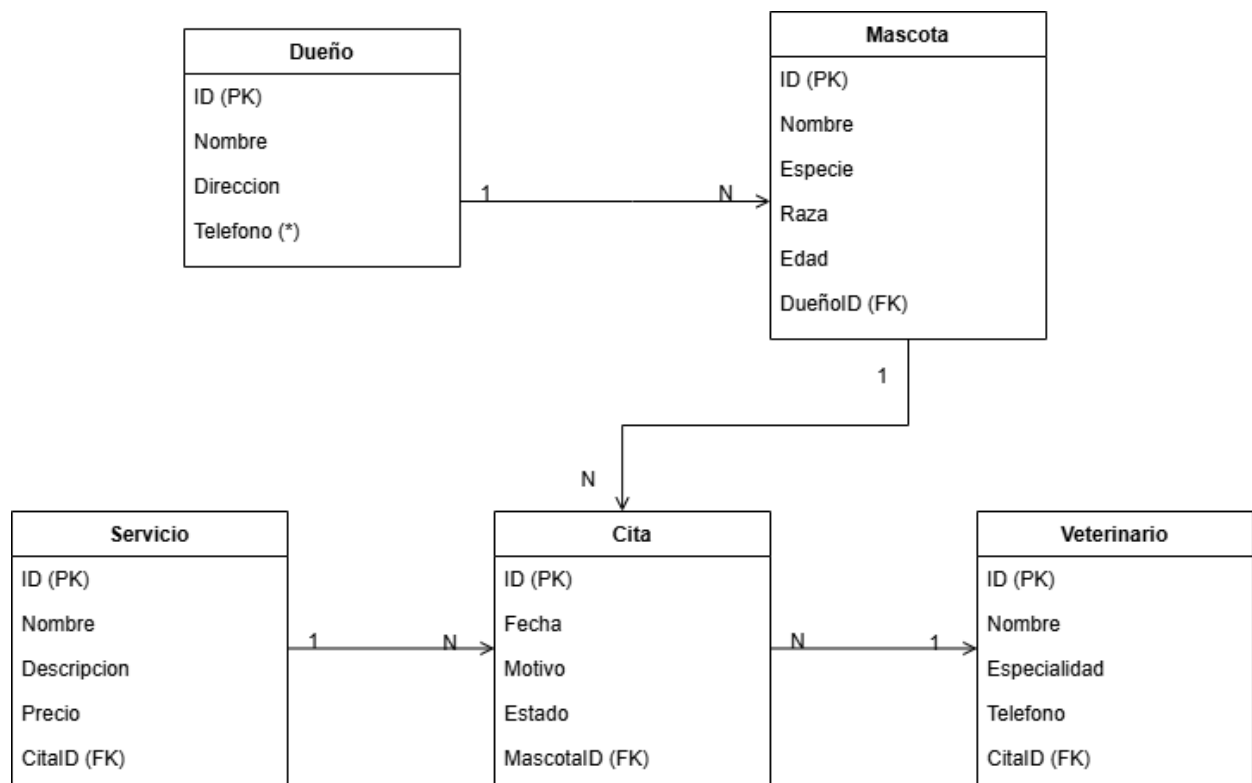
Las entidades principales y sus relaciones son las siguientes:

- **Dueño (Dueno)**
 - Atributos: Id, Nombre, Teléfono, Dirección.
 - Relación: *Uno a muchos* con **Mascota**.
 - Restricciones: Teléfono único (clave de negocio).
- **Mascota**
 - Atributos: Id, Nombre, Especie, Raza, Edad, DuenoId.
 - Relación: *Muchos a uno* con **Dueño**.
 - Relación: *Uno a muchos* con **Cita**.
- **Cita**
 - Atributos: Id, Fecha, Hora, Motivo, Estado, MascotaId, VeterinarioId, ServicioId.
 - Relación: *Muchos a uno* con **Mascota**.
 - Relación: *Muchos a uno* con **Veterinario**.
 - Relación: *Muchos a uno* con **Servicio**.
 - Restricciones: Estado limitado a los valores “Pendiente”, “Atendida”, “Cancelada”.
- **Veterinario**
 - Atributos: Id, Nombre, Especialidad, Correo, Teléfono.
 - Relación: *Uno a muchos* con **Cita**.
 - Uso: Permite identificar qué profesional atiende cada cita.
- **Servicio**
 - Atributos: Id, Nombre, Descripción, Costo.
 - Relación: *Uno a muchos* con **Cita**.
 - Uso: Clasifica las atenciones médicas por tipo (consulta, vacunación, cirugía, etc.).

Resumen de relaciones principales:

- Un **Dueño** puede tener varias **Mascotas**.
- Una **Mascota** puede tener múltiples **Citas**.
- Una **Cita** pertenece a un **Veterinario** y a un **Servicio**.

Esta estructura permite reflejar fielmente la operativa de una clínica veterinaria y facilita la trazabilidad de cada atención.



3. Priorización de procesos y etapa trabajada

Durante esta primera entrega del proyecto, se priorizaron los casos de uso fundamentales para el funcionamiento del sistema:

Prioridad	Caso de uso	Estado
Alta	Registrar dueño y mascota	Implementado
Alta	Agendar cita veterinaria	Implementado
Media	Generar reporte de citas pendientes	Implementado
Media	Actualizar estado de cita	Implementado
Baja	Listar mascotas con su dueño	Implementado
Complementaria	Registrar servicios y veterinarios	Implementado en modelo

El sistema actualmente cubre el flujo completo “**Registro → Agendamiento → Control de atención**”, asegurando persistencia de datos y validaciones consistentes.

4. Explicación detallada de cada caso de uso

- Caso de uso 1: Registrar dueño y mascota asociada

Descripción: Permite registrar un nuevo dueño y asociar una o varias mascotas a su perfil.

Flujo resumido:

- Se verifica que el teléfono no esté duplicado.
- Si el dueño no existe, se crea su registro.
- Luego se crea la mascota vinculada mediante su DuenoId.

Resultado: Confirmación del registro (HTTP 201) con los datos del dueño y la mascota creada.

- Caso de uso 2: Agendar cita veterinaria

Descripción: Permite registrar una nueva cita para una mascota existente.

Flujo resumido:

- Se valida que la mascota exista.
- Se verifica que la fecha sea igual o posterior a la actual.
- Se asigna estado inicial "Pendiente".
- Se guarda la cita en la base de datos.

Resultado: Cita registrada correctamente (HTTP 201) o error (400 / 404).

- Caso de uso 3: Generar reporte de citas pendientes

Descripción: Muestra todas las citas con estado "Pendiente" ordenadas por fecha ascendente.

Flujo resumido:

- Se consultan las citas con estado "Pendiente".
- Se incluyen datos del dueño, mascota y veterinario.
- El resultado se devuelve en formato JSON.

Resultado: Listado completo de citas pendientes para seguimiento clínico.

- **Caso de uso 4: Actualizar estado de cita**

Descripción: Permite cambiar el estado de una cita según su atención (Pendiente a Atendida / Cancelada).

Flujo resumido:

- Se busca la cita por ID.
- Se valida su existencia.
- Se actualiza el campo Estado.
- Se guarda el cambio.

Resultado: Mensaje de confirmación (HTTP 200) o error si la cita no existe.

- **Caso de uso 5: Listar mascotas con su dueño**

Descripción: Muestra un listado general de mascotas junto con la información del dueño.

Flujo resumido:

- Se consulta la tabla Mascotas con su relación Dueno.
- Se permite filtrar por especie o nombre.
- Se devuelve la información consolidada.

Resultado: Listado JSON con mascotas y sus dueños activos.

- **Caso de uso 6: Registrar servicio veterinario**

Descripción: Permite definir los distintos tipos de servicios ofrecidos en la clínica (consulta, vacunación, cirugía, etc.).

Flujo resumido:

- El administrador ingresa nombre, descripción y costo.
- El sistema crea el registro en la entidad Servicio.

Resultado: Servicio registrado correctamente (HTTP 201).

- **Caso de uso 7: Registrar veterinario**

Descripción: Permite dar de alta a un veterinario con sus datos personales y especialidad.

Flujo resumido:

- Se ingresan nombre, especialidad, correo y teléfono.
- Se valida que el correo no esté duplicado.
- Se guarda el registro.

Resultado: Veterinario registrado (HTTP 201).

Estas entidades se usan posteriormente al agendar citas y reportes médicos.

5. Criterios de aceptación y reglas de negocio aplicados a cada caso de uso

Caso	Criterio de Aceptación	Reglas de Negocio
1	Se crea correctamente un dueño y su mascota asociada.	El teléfono del dueño es único. No se pueden crear mascotas sin dueño.
2	Se registra una cita válida para una mascota existente.	La fecha debe ser \geq a la actual. Estado inicial: "Pendiente".
3	Se muestra correctamente el listado de citas pendientes.	Solo se muestran citas con estado "Pendiente"
4	Se actualiza el estado de una cita existente	Solo se permiten estados: Pendiente, Atendida, Cancelada.
5	Se listan todas las mascotas con su dueño.	Solo se muestran las mascotas activas.
6	Se registran nuevos servicios veterinarios.	El nombre del servicio debe ser único.
7	Se registran nuevos veterinarios.	El correo electrónico es obligatorio y único.