



---

# INFORME FINAL

---

CARLOS ANDRÉS ACEVEDO PACHECO  
LUIS EDUARDO BARRAGÁN HERRERA



16 DE NOVIEMBRE DE 2025  
PROGRAMACIÓN ORIENTADA A OBJETOS  
BUCARAMANGA  
UTS

## Contenido

CARACTERISTICAS GENERALES DEL PROYECTO .....	2
Introducción.....	2
Objetivo General.....	2
Alcance del Proyecto .....	3
•     Módulo de Gestión de Dueños.....	3
•     Módulo de Gestión de Mascotas.....	3
•     Módulo de Veterinarios.....	3
•     Módulo de Gestión de Citas .....	3
•     Historial Clínico .....	3
Entornos, Tecnologías y Herramientas Utilizadas.....	3
•     Lenguaje y Framework.....	3
•     IDE.....	3
•     Sistema Gestor de Base de Datos .....	4
•     Driver de conexión.....	4
•     Librerías internas usadas .....	4
•     Sistema Operativo.....	4
Diseño de Arquitectura – MVC .....	4
•     MODELO .....	4
•     DAO (Data Access Object).....	5
•     VISTAS – UI (Swing) .....	5
Pruebas y Validación del Sistema .....	6
Pruebas funcionales: .....	6

# CARACTERISTICAS GENERALES DEL PROYECTO

## Introducción

El presente documento describe en detalle el desarrollo de un sistema de gestión para una veterinaria, creado en Java SE utilizando Swing, con conexión real a una base de datos PostgreSQL.

El sistema fue construido como parte del taller académico enfocado en:

- Construcción de interfaces gráficas
- Manejo de eventos
- Conexión a bases de datos
- Aplicación del patrón de diseño MVC
- Implementación completa de CRUD
- Diseño de un panel administrativo moderno con íconos

Este sistema reproduce una aplicación real que podría ser utilizada en una veterinaria pequeña o mediana para gestionar sus operaciones básicas.

## Objetivo General

Desarrollar una aplicación de escritorio estable, funcional y modular que permita la gestión centralizada de:

- Dueños de mascotas
- Mascotas
- Veterinarios
- Citas veterinarias
- Historial clínico básico

## Alcance del Proyecto

El sistema incluye las siguientes funcionalidades principales:

- **Módulo de Gestión de Dueños**

Permite administrar los datos de los dueños de mascotas, almacenando su información personal y de contacto.

- **Módulo de Gestión de Mascotas**

Permite registrar mascotas y asociarlas a un dueño existente.

Incluye atributos como especie, raza, color, edad y peso.

- **Módulo de Veterinarios**

Controla el registro de veterinarios, su especialidad y datos de contacto.

- **Módulo de Gestión de Citas**

Un módulo central que permite:

- Agendar citas veterinarias
- Relacionar Mascota – Veterinario
- Registrar fecha y hora
- Especificar motivo
- Consultar, modificar y eliminar citas

- **Historial Clínico**

Permite revisar todas las citas realizadas por mascota, simulando un historial clínico básico.

## Entornos, Tecnologías y Herramientas Utilizadas

- **Lenguaje y Framework**

- Java SE 8+
- Swing Para la construcción de las interfaces gráficas

- **IDE**

- NetBeans

- Sistema Gestor de Base de Datos
  - PostgreSQL 15+
- Driver de conexión
  - PostgreSQL JDBC Driver, este permite establecer la conexión desde Java a la base de datos.
- Librerías internas usadas
  - JDBC (java.sql) → para consultas SQL
  - LocalDate / LocalTime → para manejo de fechas y horas
- Sistema Operativo
  - Windows 10 (entorno del estudiante)

## Diseño de Arquitectura – MVC

El proyecto utiliza una estructura lógica basada en MVC (Modelo – Vista – Controlador):

### • MODELO

Ubicado en “com.veterinaria.model”

Representa las entidades de la base de datos:

- Dueño
- Mascota
- Veterinario
- Cita

Cada clase contiene:

- Atributos privados
- Constructores
- Métodos getters/setters

- DAO (Data Access Object)

Ubicado en “com.veterinaria.dao”

Aquí se encuentran:

- Interfaces como DueñoDAO, MascotaDAO...
- Implementaciones: DueñoDAOImpl, MascotaDAOImpl...

Realizan operaciones del tipo:

- INSERT
- UPDATE
- DELETE
- SELECT

Utilizan la clase de conexión:

```
Connection conn = Conexion.getConnection();
```

- VISTAS – UI (Swing)

Ubicado en “com.veterinaria.ui”

Incluye todas las interfaces gráficas:

- Principal (Panel administrativo con botones e iconos)
- FrmDueno
- FrmMascotas
- FrmVeterinarios
- FrmCitas
- FrmHistorial

Cada formulario tiene:

- ✓ Botones para CRUD
- ✓ Validaciones de entrada
- ✓ Interacción con DAOs

## Pruebas y Validación del Sistema

Las pruebas realizadas incluyen:

### Pruebas funcionales:

- Registro de nuevos dueños
- Agregar mascotas referenciadas a su dueño
- Crear, editar y eliminar veterinarios
- Agendar citas con fecha, hora y motivo
- Ver historial filtrado por mascota
- Validación de restricciones:
  - Campos obligatorios
  - Relaciones FK
  - Tipos correctos de fecha y hora