**Sistema de Gestión para una Clínica Veterinaria (Avance)**

María del Mar Díaz Ruiz, Marshall Gutierrez Arce, Ronny Jafeth Salas, Jairo José Silva Martínez

BA-INFORM, Universidad Nacional

Programacion – EIF20

Gloriana Peña Ramirez

9 de Octubre del 2024

# 

# Índice

[Objetivo General 3](#_Toc179353196)

[Objetivos Específicos 3](#_Toc179353197)

[Introducción 3](#_Toc179353198)

[En qué consiste 4](#_Toc179353199)

[Tecnología usada 4](#_Toc179353200)

[Utilización de lo visto en clases 4](#_Toc179353201)

[Conclusiones 5](#_Toc179353202)

# Objetivo General

Desarrollar un sistema de software orientado a objetos para gestionar una clínica veterinaria, permitiendo el manejo de doctores, pacientes (animales), medicinas, citas, laboratorios y personal administrativ.

# Objetivos Específicos

1. Identificar correctamente la herencia para reutilizar código y estructurar jerarquías.

2. Demostrar polimorfismo para manejar diferentes tipos de objetos que compartan una interfaz común.

3. Analizar sobrecarga y sobrescritura de métodos para personalizar el comportamiento de las clases.

4. Establecer un sistema de software extensible y fácil de mantener

# Introducción

Este proyecto plantea la creación de un sistema de software que ayude a gestionar las diversas actividades de una clínica veterinaria. A través de técnicas avanzadas de programación orientada a objetos como la herencia, polimorfismo, sobrecarga y sobrescritura, el sistema permitirá un manejo eficiente de doctores, animales, citas, medicinas y más. Se busca que el sistema sea modular, reutilizable y fácil de mantener.

# En qué consiste

Este programa consiste en lograr administrar las citas, medicina, el laboratorio, los doctores, Personal administrativo, los dueños y sus respectivas mascotas, para así poder tener un mejor control en la clínica.

# Tecnología usada

El sistema está orientado a objetos, empleando las cosas que vimos en clases, como: herencia, polimorfismo, sobrecarga, y sobreescritura. El lenguaje es C++, usado en la aplicación de Zinjai y el UML es hecho desde draw.io.

# Utilización de lo visto en clases

Polimorfismo: Al tener el manejo de distintos tipos de doctores y animales mediante una interfaz común, por ejemplo, diferentes tratamientos y costos dependiendo de la especialidad del doctor y también los diferentes tipos de sonidos de acuerdo al tipo de animal.

Herencia: Utilizada en clases como Doctor, Dueño y Administrativo que extienden de Persona, y en las subclases de Animal.

Sobrecarga: Métodos sobrecargados en la clase Cita para permitir la creación de citas con o sin pruebas de laboratorio.

Sobrescritura: Sobrescritura de métodos en subclases como Animal (por ejemplo, para mostrar el sonido de cada tipo de animal) y doctores para calcular costos de procedimientos.

Archivos: Implementación y manejo de archivos para registrar y almacenar la información del sistema, como el historial médico de animales, medicamentos y citas.

# Conclusiones

El sistema de gestión para la clínica veterinaria proporciona una solución eficiente y estructurada para manejar múltiples aspectos de la operación diaria de la clínica. Al aplicar conceptos avanzados de programación orientada a objetos, se garantiza la extensibilidad, modularidad y facilidad de mantenimiento del sistema. Además, el uso correcto de herencia, polimorfismo, sobrecarga y sobrescritura facilita la personalización de comportamientos, lo que es crucial en un entorno con múltiples tipos de usuarios y pacientes.