

# **Trabajo Final Integrador:**

## **Arquitectura y Desarrollo SQL - Clínica "VetSalud"**

Presentado por:

Julián Javier Gómez Reyes

Bryan Camilo Chiquiza Malagon

Brian Steven Lerma Maceto

Curso:

Bases de datos

Maestría en ciencia de datos.

Universidad Santo tomas  
diciembre de 2025

# Arquitectura y Desarrollo SQL - Clínica "VetSalud"

## Contexto del Negocio:

"La clínica necesita un sistema para gestionar su operación. Se requiere almacenar información de los **dueños** de las mascotas (cédula, nombre, dirección, varios teléfonos). Cada **mascota** (nombre, especie, raza, fecha de nacimiento) pertenece a un único dueño. Los **Veterinarios** (cédula, nombre, especialidad) atienden a las mascotas en **citas** (fecha, hora, motivo). En cada cita, se genera un diagnóstico y se pueden recetar varios medicamentos. La clínica también quiere llevar un registro del **historial de vacunación** de cada mascota, considerando que una mascota tiene muchas vacunas a lo largo de su vida."

## Entidades

ENTIDAD: dueños - FUERTE						
PK	CAMPO	TIPO	NOT NULL	UNIQUE	FK	OBSERVACIONES
X	id	bigint	X	X		AUTOINCREMENTAL
	cedula	String (15)	X			
	nombre	String (50)	X			
	direccion	String (100)	X			
	created_at	timestamp(0)				
	updated_at	timestamp(0)				

ENTIDAD: mascotas - FUERTE						
PK	CAMPO	TIPO	NOT NULL	UNIQUE	FK	OBSERVACIONES
X	id	bigint	X	X		AUTOINCREMENTAL
	nombre	String (50)	X			
	especie	text (255)	X			
	raza	text (255)	X			
	fecha_nacimiento	date	X			
	dueno_id	bigint	X		X	
	created_at	timestamp(0)				
	updated_at	timestamp(0)				

ENTIDAD: medicamentos						
PK	CAMPO	TIPO	NOT NULL	UNIQUE	FK	OBSERVACIONES
X	id	bigint	X	X		AUTOINCREMENTAL
	nombre	string(50)				
	descripcion	text (255)	X			
	created_at	timestamp(0)				
	updated_at	timestamp(0)				

ENTIDAD: veterinarios - FUERTE						
PK	CAMPO	TIPO	NOT NULL	UNIQUE	FK	OBSERVACIONES
X	id	bigint	X	X		AUTOINCREMENTAL
	cedula	string(15)	X			
	nombre	string(50)	X			
	especialidad	string(50)	X			
	created_at	timestamp(0)				
	updated_at	timestamp(0)				

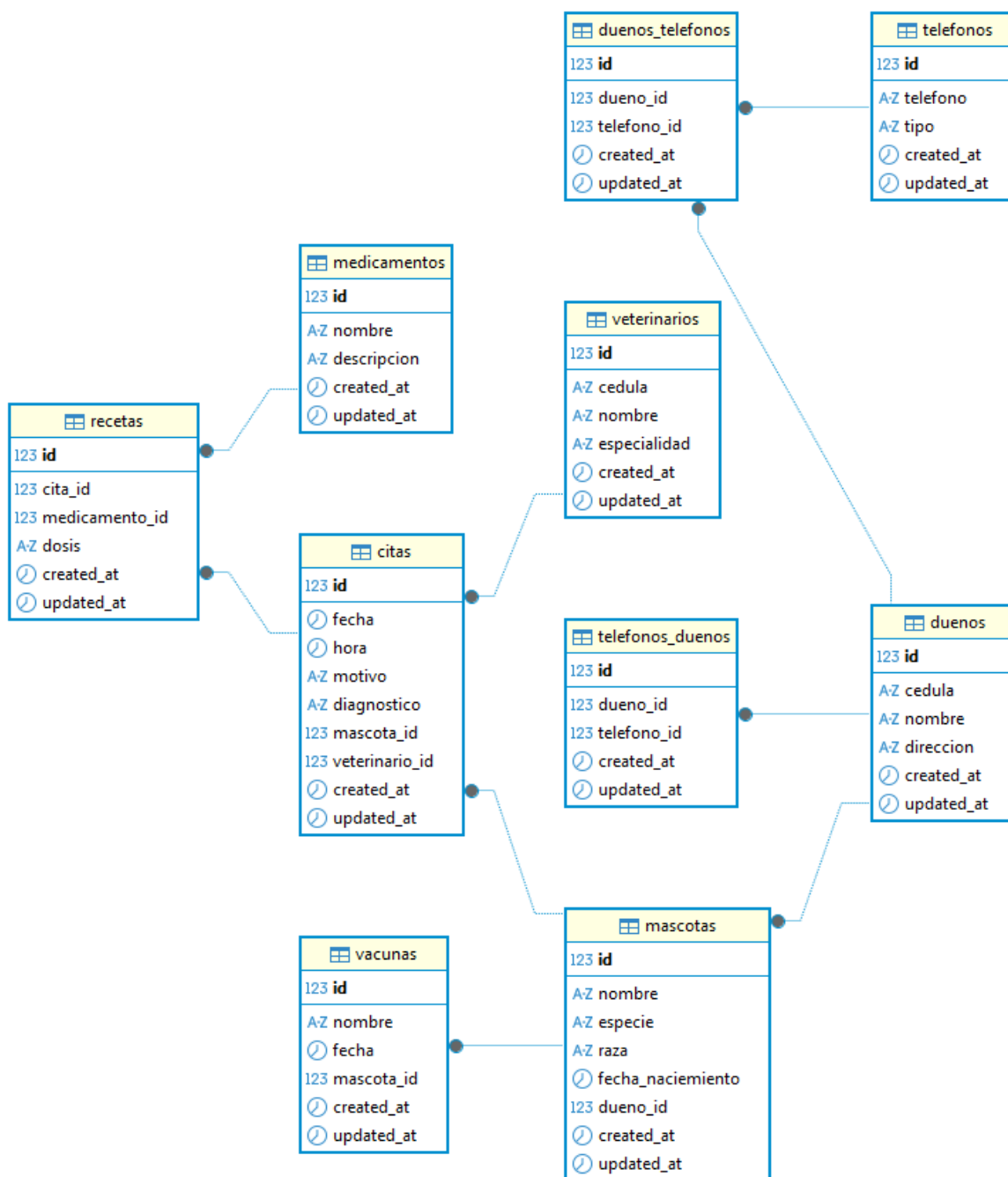
ENTIDAD: recetas - DÉBIL						
PK	CAMPO	TIPO	NOT NULL	UNIQUE	FK	OBSERVACIONES
X	id	bigint	X	X		AUTOINCREMENTAL
	cita_id	bigint	X		X	
	medicamento_id	bigint	X		X	
	dosis	string(50)	X			
	created_at	timestamp(0)				
	updated_at	timestamp(0)				

ENTIDAD: teléfonos - FUERTE						
PK	CAMPO	TIPO	NOT NULL	UNIQUE	FK	OBSERVACIONES
X	id	bigint	X	X		AUTOINCREMENTAL
	telefono	bigint	X			
	tipo	check('Personal', 'Acudiente', 'Auxiliar', 'Familiar')	X			
	created_at	timestamp(0)				
	updated_at	timestamp(0)				

ENTIDAD: duenos_telefonos - DÉBIL						
PK	CAMPO	TIPO	NOT NULL	UNIQUE	FK	OBSERVACIONES
X	id	bigint	X	X		AUTOINCREMENTAL
	dueno_id	bigint	X		X	
	telefono_id	bigint	X		X	
	created_at	timestamp(0)				
	updated_at	timestamp(0)				

ENTIDAD: vacunas – DÉBIL						
PK	CAMPO	TIPO	NOT NULL	UNIQUE	FK	OBSERVACIONES
X	id	bigint	X	X		AUTOINCREMENTAL
	nombre	String(50)	X			
	fecha	date	X			
	masкота_id	bigint	X			
	created_at	timestamp(0)				
	updated_at	timestamp(0)				

# 1. Modelo Entidad - Relación



## 2. Diseño Lógico y Normalización:

Entregue la lista de tablas resultante del mapeo.

### 1. Tablas de Entidades Fuertes

Estas tablas tienen independencia y claves primarias simples.

- **DUENOS**
  - Columnas: id (PK), cedula, nombre, direccion.
- **VETERINARIOS**
  - Columnas: id (PK), cedula, nombre, especialidad.
- **MEDICAMENTOS**
  - Columnas: id (PK), nombre, descripcion.

### 2. Tablas de Entidades Débiles / Multivaluadas

Estas tablas dependen de una entidad fuerte para existir.

- **DUENOS\_TELEFONOS (Resultado del atributo multivaluado "Teléfonos")**
  - Columnas: dueno\_id (PK, FK), telefono\_id (PK, FK).  
*Nota: La clave primaria es compuesta (ambas columnas).*
- **MASCOTAS (Aunque conceptualmente fuerte, depende de un dueño en el modelo relacional)**
  - Columnas: id (PK), nombre, especie, raza, fecha\_nacimiento, dueno\_id (FK).
- **VACUNACION (Entidad débil de Mascotas)**
  - Columnas: id (PK), nombre, fecha, mascota\_id (FK).

### 3. Tablas Transaccionales y de Relación

Estas tablas conectan las entidades anteriores para registrar la operación del negocio.

- **CITAS**
  - Columnas: id (PK), fecha, hora, motivo, diagnostico, mascota\_id (FK), veterinario\_id (FK).

- **RECETAS (Tabla puente para la relación muchos a muchos entre Cita y Medicamento)**
  - **Columnas:** cita\_id (PK, FK), medicamento\_id (PK, FK), dosis.

**Justificación:** Escriba un párrafo explicando por qué su tabla de "Citas" y "Dueños" cumple con la Tercera Forma Normal (3NF), asegurando que no existan dependencias transitivas

La tabla "duenos": La tabla cumple con la 3NF porque todos sus atributos no clave (nombre, dirección) dependen única y exclusivamente de la clave primaria. Se eliminaron las dependencias transitivas al separar los teléfonos en una tabla independiente (DUENO\_TELEFONO); de haberlos dejado en la tabla principal, habríamos violado la 1NF (por ser multivaluados) o generada redundancia. Al no existir columnas como "Código Postal" que a su vez determinen la "Ciudad" dentro de esta misma tabla, garantizamos que no hay atributos que dependan de otros atributos no clave.

Justificación para la tabla "citas": Esta tabla cumple con la 3NF porque cada columna descriptiva (fecha, hora, motivo, diagnóstico) se refiere directamente al evento de la cita identificado por su clave primaria (id) y no a los objetos relacionados. Se evitan las dependencias transitivas al utilizar claves foráneas (imascota\_id, veterinario\_id) únicamente como referencias; si incluyéramos, por ejemplo, el *nombre de la mascota* o la *especialidad del veterinario* en esta tabla, tendríamos una dependencia transitiva (donde el Nombre dependería del ID de la mascota y no del ID de la cita), violando la 3NF. Al mantener esos datos en sus propias tablas, la integridad se conserva.