

**SANTIAGO CENTENO CASTAÑO**

**P3**

**PEDRO FELIPE GÓMEZ BONILLA**

**CAMPUSLANDS  
APOLO  
RUTA BACKEND  
FLORIDABLANCA  
2024**

## Tabla de Contenidos

<b>Introducción</b>	<b>4</b>
<b>Caso de Estudio</b>	<b>5</b>
<b>Planificación</b>	<b>5</b>
Construcción del Modelo Conceptual	5
Descripción	5
Gráfica	5
Descripción Técnica	6
Construcción del Modelo Lógico	6
Descripción	6
Gráfica	6
Descripción Técnica	6
Normalización del Modelo Lógico	6
Primera Forma Normal (1FN)	7
Descripción	7
Gráfica	7
Descripción Técnica	7
Segunda Forma Normal (2FN)	7
Descripción	8
Gráfica	8
Descripción Técnica	8
Tercera Forma Normal (3FN)	8
Descripción	8
Gráfica	8
Descripción Técnica	8
Construcción del Modelo Físico	9
Descripción	9
Código	9
Descripción Técnica	10
Diagrama E-R	10
Descripción	10
Gráfica	10
Descripción Técnica	10
Tablas	11
Descripción	11
Gráfica	11
Descripción Técnica	11
Relaciones entre Tablas	11
Descripción	11
Gráfica	12
Descripción Técnica	12
Inserción de Datos	12
Descripción	12

	Gráfica	12
	Descripción Técnica	12
<b>Referencias</b>		<b>13</b>

# Introducción

En este documento voy a explorar y realizar un análisis profundo de la base de datos, basada en las muertes accidentales en Colombia de los años 2015-2024. Crearé dos diagramas, uno para el modelo conceptual de la base de datos y el otro para el modelo lógico. Indagaré en cada entidad y sus atributos para saber qué es lo que puedo mejorar de esta base de datos y así lograr una base de datos más concisa, ya que es posible encontrar redundancia o desorden en esta base de datos y considero que puede ser mucho mejor si manipulo los datos de una manera más estructurada.

## Caso de Estudio

La base de datos a analizar está centrada en una estadística de muertes accidentales ocurridas en Colombia desde el año 2015 hasta el año 2024. En esta base de datos se encuentran secciones como: edad, género, lugar y fecha; también se especifican más datos con respecto a la causa y motivo de muerte.

La base de datos está estructurada de manera poco conveniente para consultas de datos específicos, así que la ordenaré según los estándares necesarios. Se aplicarán normas formales a la base de datos para evitar redundancias.

## Planificación

### Construcción del Modelo Conceptual

Para crear el modelo conceptual empecé con una versión beta del mismo la cual mostraba los datos de manera muy simplificada, este proceso me facilita mucho la creación del modelo conceptual real.

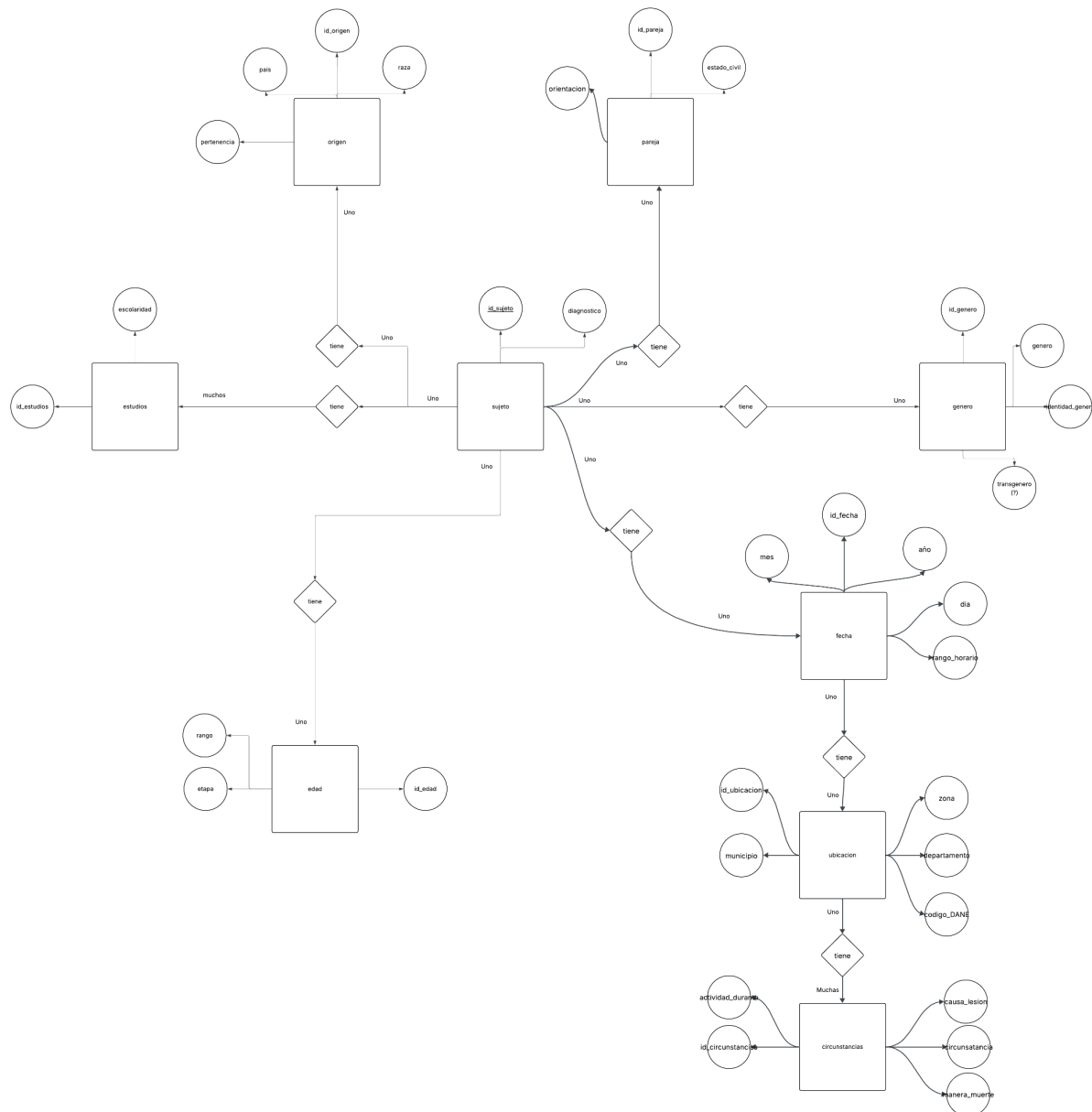
A la hora de crear este modelo tuve que hacer un análisis, en el análisis note que los datos originales estaban muy esparcidos y podrían ser amontonados de manera más inteligente para así evitar redundancia en la base de datos real y hacer un sistema fácil de acceder y entender.

### Descripción

Al crear el modelo conceptual yo considere, según la base de datos necesitada, que este gráfico debía tener 9 entidades y la mayoría de estas entidades deben estar conectadas a un único sujeto ya que mucha de la información viene de una misma persona, por eso le di un núcleo al

gráfico. El núcleo funciona como la identidad de la persona debido a que la persona no cuenta con un nombre, pero sí con una ID y información adicional de quién era.

## Gráfica



## Descripción Técnica

Todas las entidades conectan con el “sujeto” excepto la “ubicación” y “circunstancias” que van acopladas junto con la fecha, ya que no tiene relación con la información personal del sujeto. Las demás que si están conectadas directamente con “sujeto” cumplen la función de dar información sobre la identidad de la persona, desde su edad hasta su nivel de estudios,

todo esto es importante para saber quién era el individuo en cuestión y porque murió, la información de como y donde murió ya es aparte de quien era.

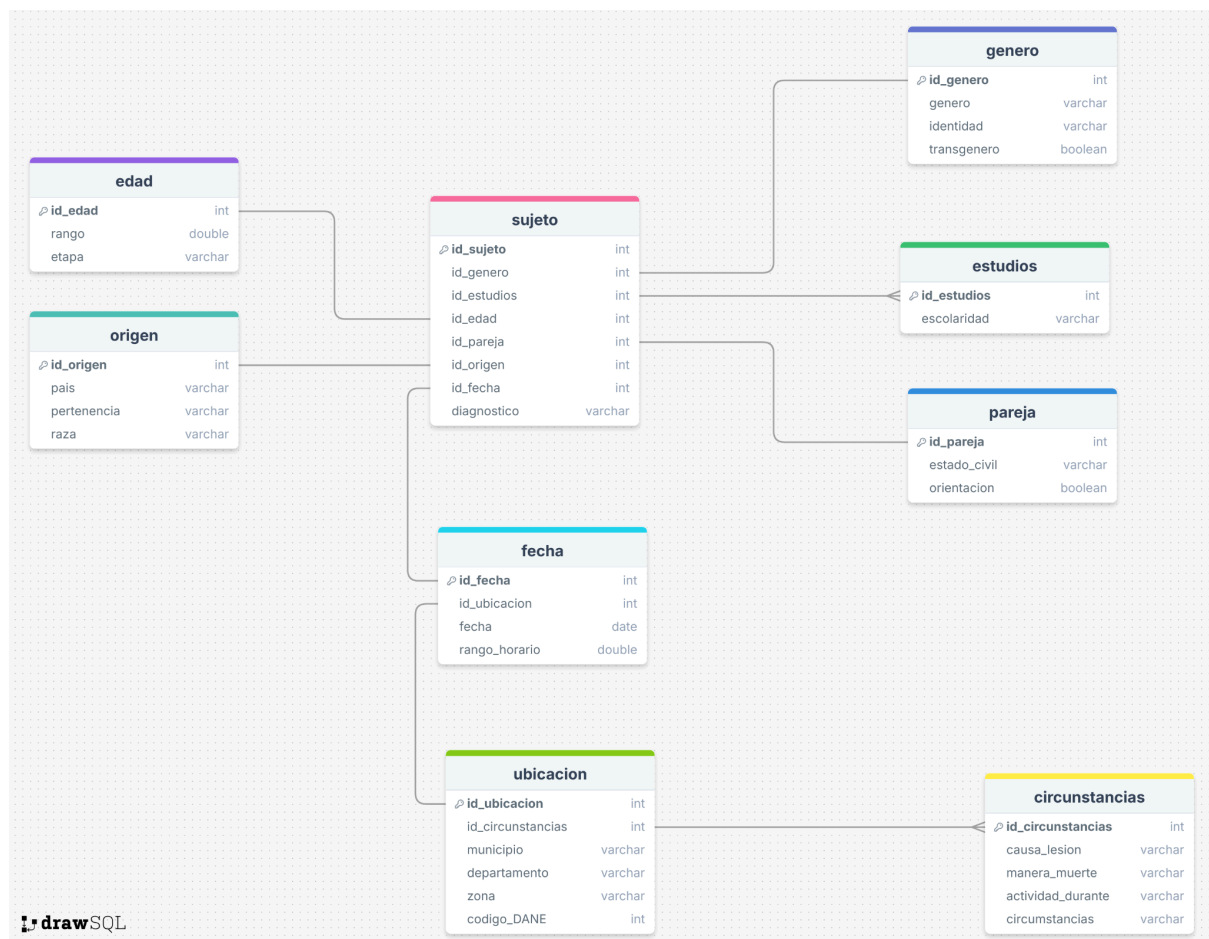
## Construcción del Modelo Lógico

Para el modelo lógico tuve en cuenta el modelo conceptual. Todo lo que en el modelo lógico se añadió, en el modelo conceptual ya estaba, ya que no hubo diferencia alguna, excepto las ID de sujeto que fueron necesarias para hacer la conexión de relaciones entre las tablas.

## Descripción

El modelo lógico, así como el modelo conceptual, contiene 9 tablas, cada tabla tiene conexión con sujeto de alguna u otra forma, por lo que es necesario poner las ID correspondientes en la tabla principal, cada ID corresponde a una parte de la identidad de la persona.

## Gráfica



## Descripción Técnica

La tabla "sujeto" tiene como relación a 6 tablas secundarias, la sexta es "fecha", que encadena a "ubicación" y "circunstancia", estas 2 están aparte porque no aportan a la identidad de la persona. La cardinalidad en esta tabla va desde uno a uno hasta uno a mucho, pero no hay muchos a muchos porque ninguna tabla necesito una conexión de muchos a muchos.

## Normalización del Modelo Lógico

Para que la base de datos cumpla con los estándares se debe hacer caso a la normalización. La normalización manda que los datos se organicen a través de diferentes niveles, forma normal 1, 2 y 3. Estas formas normales se tuvieron en cuenta a lo largo de la creación del modelo conceptual y por consiguiente, también del modelo lógico.

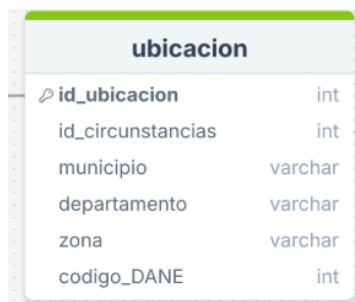
### Primera Forma Normal (1FN)

La primera forma normal indica que un dato solo puede contener un valor en la misma línea. Si un dato va a tener más de un valor se necesitan asignar varias líneas para este proceso.

#### Descripción

Me aseguré de que cada atributo tuviera su único valor. Para implementar la lógica de añadir nuevos atributos se necesita usar código, por lo tanto en el modelo lógico no tengo conocimiento para hacerlo.

#### Gráfica



ubicacion	
id_ubicacion	int
id_circunstancias	int
municipio	varchar
departamento	varchar
zona	varchar
codigo_DANE	int

#### Descripción Técnica

### Segunda Forma Normal (2FN)

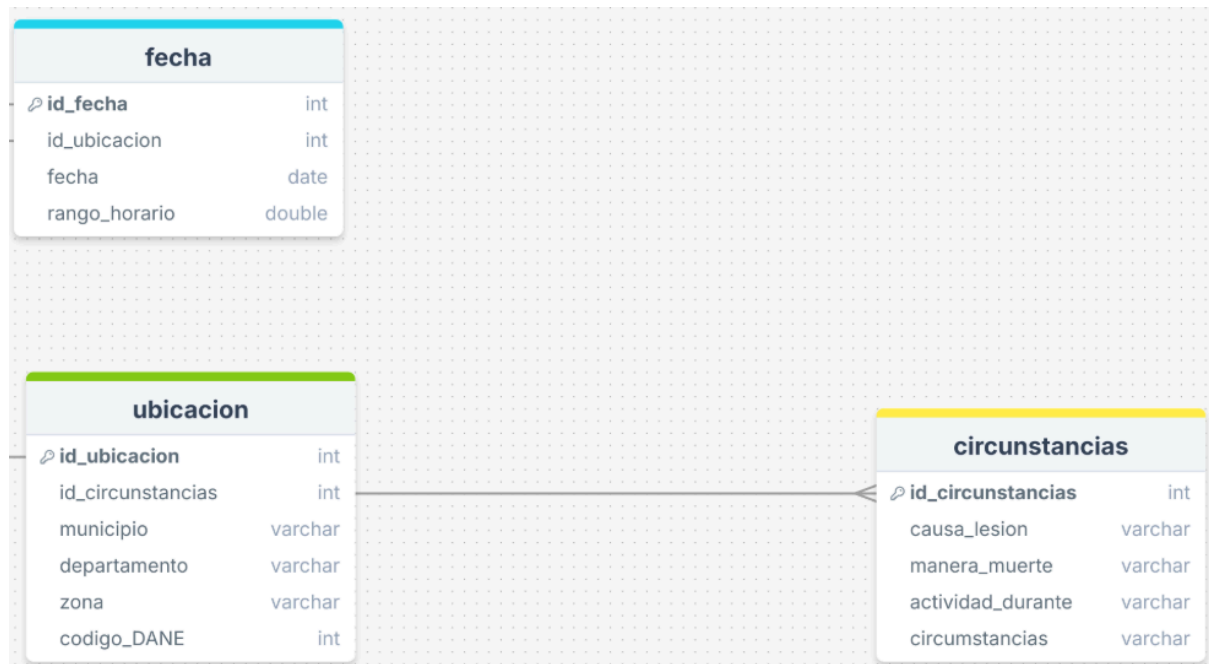
La segunda forma normal se encarga de ordenar los datos en diferentes tablas. Una persona normal podría pensar que cliente y ciudad deberían ir juntos pero a veces en base de datos la separación de estos es esencial para mantener un orden y lógica correctos.

#### Descripción

Antes de hacer cualquier modelo esta forma normal fue utilizada para separar datos que personalmente creí que deben ir separados para no tener una base de datos saturada sino que

funcional. La separación de tablas permite que se implemente y modifique lógica futura a la base de datos, como por ejemplo si en un futuro se necesita agregar una dependencia entre tablas. En este caso la segunda forma normal haría el acceso a cada tabla fácil para un programador de base de datos.

## Gráfica



## Descripción Técnica

### Tercera Forma Normal (3FN)

La tercera forma normal establece que si un atributo dentro de una entidad no depende directamente de su llave principal entonces se debe crear un atributo alternativo con dicho atributo y darle una llave foránea que relaciona ambos atributos.

## Descripción

Al crear tablas independientes que no dependen de la llave principal se da libertad de funcionamiento a la base de datos, ya que le da la posibilidad de añadirle relación a una tabla futura junto con la tabla secundaria recién creada. A veces parece que las tablas ya están estructuradas correctamente pero vale la pena revisar si la tercera forma normal se aplica del todo, porque puede que se esté omitiendo la creación de más tablas útiles en un futuro.



Gráfica



Descripción Técnica

# Construcción del Modelo Físico

## Descripción

Para la creación del modelo físico, se copió el código que la página draw sql brindo, aunque es posible crear el código manualmente y es algo que muchas personas optan por hacer, es preferible copiar el código directo de la página para hacerlo todo más rápido.

## Código

```
CREATE TABLE `sujeto`(  
  `id_sujeto` INT UNSIGNED NOT NULL AUTO_INCREMENT PRIMARY KEY,  
  `id_genero` INT NOT NULL,  
  `id_estudios` INT NOT NULL,  
  `id_edad` INT NOT NULL,  
  `id_pareja` INT NOT NULL,  
  `id_origen` INT NOT NULL,  
  `id_fecha` INT NOT NULL,  
  `diagnostico` VARCHAR(255) NOT NULL  
);  
  
CREATE TABLE `edad`(  
  `id_edad` INT UNSIGNED NOT NULL AUTO_INCREMENT PRIMARY KEY,  
  `rango` DOUBLE NOT NULL,  
  `etapa` VARCHAR(255) NOT NULL  
);  
  
CREATE TABLE `genero`(  
  `id_genero` INT UNSIGNED NOT NULL AUTO_INCREMENT PRIMARY KEY,  
  `genero` VARCHAR(255) NOT NULL,  
  `identidad` VARCHAR(255) NOT NULL,  
  `transgenero` BOOLEAN NOT NULL  
);  
  
CREATE TABLE `pareja`(  
  `id_pareja` INT UNSIGNED NOT NULL AUTO_INCREMENT PRIMARY KEY,  
  `estado_civil` VARCHAR(255) NOT NULL,  
  `orientacion` BOOLEAN NOT NULL  
);  
  
CREATE TABLE `origen`(  
  `id_origen` INT UNSIGNED NOT NULL AUTO_INCREMENT PRIMARY KEY,  
  `pais` VARCHAR(255) NOT NULL,  
  `pertenencia` VARCHAR(255) NOT NULL,  
  `raza` VARCHAR(255) NOT NULL  
);  
  
CREATE TABLE `fecha`(  
  `id_fecha` INT UNSIGNED NOT NULL AUTO_INCREMENT PRIMARY KEY,  
  `id_ubicacion` INT NOT NULL,  
  `fecha` DATE NOT NULL,
```

```

        `rango_horario` DOUBLE NOT NULL
    );
CREATE TABLE `ubicacion` (
    `id_ubicacion` INT UNSIGNED NOT NULL AUTO_INCREMENT PRIMARY KEY,
    `id_circunstancias` INT NOT NULL,
    `municipio` VARCHAR(255) NOT NULL,
    `departamento` VARCHAR(255) NOT NULL,
    `zona` VARCHAR(255) NOT NULL,
    `codigo_DANE` INT NOT NULL
);
CREATE TABLE `estudios` (
    `id_estudios` INT UNSIGNED NOT NULL AUTO_INCREMENT PRIMARY KEY,
    `escolaridad` VARCHAR(255) NOT NULL
);
CREATE TABLE `circunstancias` (
    `id_circunstancias` INT UNSIGNED NOT NULL AUTO_INCREMENT PRIMARY KEY,
    `causa_lesion` VARCHAR(255) NOT NULL,
    `manera_muerte` VARCHAR(255) NOT NULL,
    `actividad_durante` VARCHAR(255) NOT NULL,
    `circumstancias` VARCHAR(255) NOT NULL
);
ALTER TABLE
    `ubicacion` ADD CONSTRAINT `ubicacion_id_circunstancias_foreign` FOREIGN
KEY(`id_circunstancias`) REFERENCES `circunstancias`(`id_circunstancias`);
ALTER TABLE
    `sujeito` ADD CONSTRAINT `sujeito_id_pareja_foreign` FOREIGN KEY(`id_pareja`)
REFERENCES `pareja`(`id_pareja`);
ALTER TABLE
    `sujeito` ADD CONSTRAINT `sujeito_id_genero_foreign` FOREIGN KEY(`id_genero`)
REFERENCES `genero`(`id_genero`);
ALTER TABLE
    `fecha` ADD CONSTRAINT `fecha_id_ubicacion_foreign` FOREIGN KEY(`id_ubicacion`)
REFERENCES `ubicacion`(`id_ubicacion`);
ALTER TABLE
    `sujeito` ADD CONSTRAINT `sujeito_id_edad_foreign` FOREIGN KEY(`id_edad`)
REFERENCES `edad`(`id_edad`);
ALTER TABLE
    `sujeito` ADD CONSTRAINT `sujeito_id_fecha_foreign` FOREIGN KEY(`id_fecha`)
REFERENCES `fecha`(`id_fecha`);
ALTER TABLE
    `sujeito` ADD CONSTRAINT `sujeito_id_estudios_foreign` FOREIGN KEY(`id_estudios`)
REFERENCES `estudios`(`id_estudios`);
ALTER TABLE
    `sujeito` ADD CONSTRAINT `sujeito_id_origen_foreign` FOREIGN KEY(`id_origen`)
REFERENCES `origen`(`id_origen`);

```



## Descripción Técnica

El código presentado crea cada tabla y al final hace las conexiones con sus llaves foráneas, estas conexiones de llave foráneas a primaria son en realidad las relaciones presentadas en los anteriores modelos. A cada columna de cada tabla creada en el modelo físico se le puede asignar un valor con el comando: "insert into", pero debido a la naturaleza de este análisis no es necesario introducir ningún valor a ningún atributo.

## Diagrama E-R

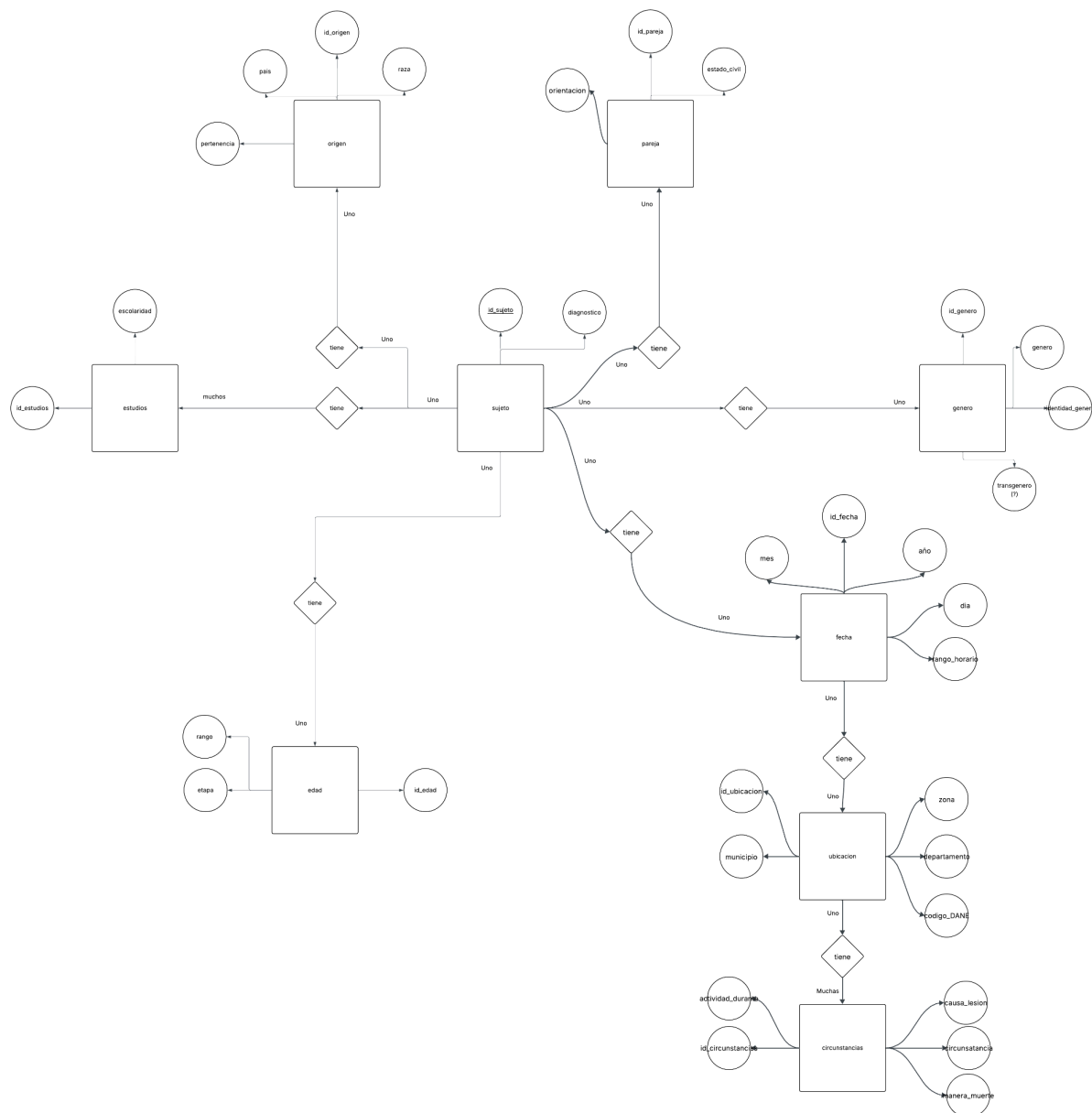
Para crear el modelo E-R empecé con una versión beta del mismo la cual mostraba los datos de manera muy simplificada, este proceso me facilita mucho la creación del modelo conceptual real.

A la hora de crear este modelo tuve que hacer un análisis, en el análisis note que los datos originales estaban muy esparcidos y podrían ser amontonados de manera más inteligente para así evitar redundancia en la base de datos real y hacer un sistema fácil de acceder y entender.

## Descripción

Al crear el modelo E-R yo considere, según la base de datos necesitada, que este gráfico debía tener 9 entidades y la mayoría de estas entidades deben estar conectadas a un único sujeto ya que mucha de la información viene de una misma persona, por eso le di un núcleo al gráfico. El núcleo funciona como la identidad de la persona debido a que la persona no cuenta con un nombre, pero sí con una ID y información adicional de quién era.

## Gráfica



## Descripción Técnica

Todas las entidades conectan con el “sujeto” excepto la “ubicación” y “circunstancias” que van acopladas junto con la fecha, ya que no tiene relación con la información personal del sujeto. Las demás que si están conectadas directamente con “sujeto” cumplen la función de dar información sobre la identidad de la persona, desde su edad hasta su nivel de estudios, todo esto es importante para saber quién era el individuo en cuestión y porque murió, la información de como y donde murió ya es aparte de quien era.

## Tablas

Las tablas son las entidades que cada modelo contiene, las entidades contienen columnas a las cuales se les llaman atributos. Pueden haber muchas tablas con muchos atributos dependiendo de la base de datos que se vaya a crear, pero normalmente se mantiene conciso para tener un orden adecuado.

### Descripción

La base de datos analizada contiene al menos 15 tablas o más por eso para hacerla más concisa la reduje a 9. Las 9 tablas contienen todos los atributos necesarios y mencionados en la base de datos original así que ninguna información se omite.

### Gráfica



The image shows a screenshot of a database management interface. At the top, there is a header bar with the word "Tables" on the left, a close button (X) in the center, and a search icon (magnifying glass) on the right. Below the header, there is a list of nine tables, each with a colored vertical bar on the left and a three-dot menu icon on the right. The tables are: circunstancias (yellow bar), estudios (green bar), ubicacion (light green bar), fecha (cyan bar), origen (teal bar), pareja (blue bar), genero (dark blue bar), edad (purple bar), and sujeto (pink bar).

Tables	
circunstancias	⋮
estudios	⋮
ubicacion	⋮
fecha	⋮
origen	⋮
pareja	⋮
genero	⋮
edad	⋮
sujeto	⋮

### Descripción Técnica

La base de datos original contiene información como edad en varias tablas diferentes para separar rangos y etapas, pero yo creí que esto no era necesario, así que junté toda la información respectiva en una sola tabla. Hice el mismo proceso para todas las ta

## Relaciones entre Tablas

Cada tabla se relaciona directa o indirectamente con una tabla principal, por lo menos en esta base de datos, es por eso que es necesario explorar cada conexión con su tabla y el porqué de la relación entre ambas.

### *Descripción*

En todo el diagrama completo hay un total de 8 relaciones, hay relaciones desde uno-uno a uno-muchos, pero no hace relaciones muchos a muchos. Las relaciones ayudan a entender el porqué de las tablas.

### *Gráfica*

### *Descripción Técnica*

Inserción de Datos

### *Descripción*

### *Gráfica*

### *Descripción Técnica*

## Referencias

