

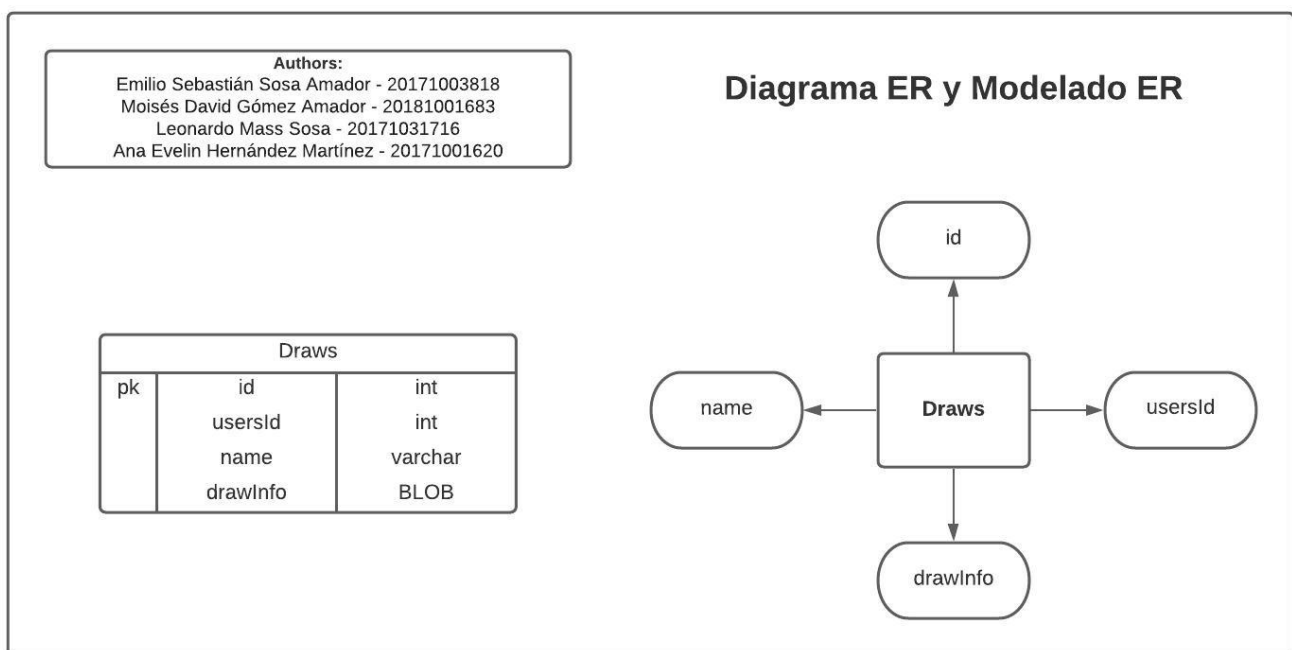
Base de Datos B - Análisis

Al realizar un pequeño y minucioso estudio de los elementos que se deben representar en esta base de datos, se logró distinguir una serie de atributos necesarios para lograr un mejor desempeño en el programa.

A continuación se presenta una explicación detallada de todos los elementos que se encuentran en la Base de Datos B del archivo *DDS_B.sql*.

Modelado ER y Diagrama ER

La base de datos B es requerida para guardar archivos JSON de respaldo con compresión. Estos archivos JSON deben ser los mismos que se almacenan en la base de datos A con tipo de dato JSON y que funcionan como componentes de almacenamiento de datos del sistema de dibujo.



Este diagrama y modelado fueron creados bajo la percepción general de la entidad *Draws* con sus atributos:

- id
- userId
- name
- drawInfo

Ya que esta base de datos solo debe de funcionar como un respaldo de la información de los dibujos en el programa no se establece relación con ninguna otra entidad.

NOTA: Esta es la versión 1.0 del análisis de la base de datos

Análisis de la Tabla y sus Atributos

Tabla *Draws*

Esta tabla representa los dibujos creados por los usuarios que pasan a la base de datos B como archivos encriptados.

```
1 CREATE TABLE Draws(  
2     id INT AUTO_INCREMENT PRIMARY KEY,  
3     userId INT NOT NULL COMMENT "Id del usuario que creo el dibujo.",  
4     var_name VARCHAR(50) NOT NULL COMMENT "Nombre del dibujo.",  
5     blo_drawInfo BLOB NOT NULL COMMENT  
6     "Contiene archivos .json encriptados con la información del dibujo."  
7 ) COMMENT "Dibujos realizados por los usuarios";
```

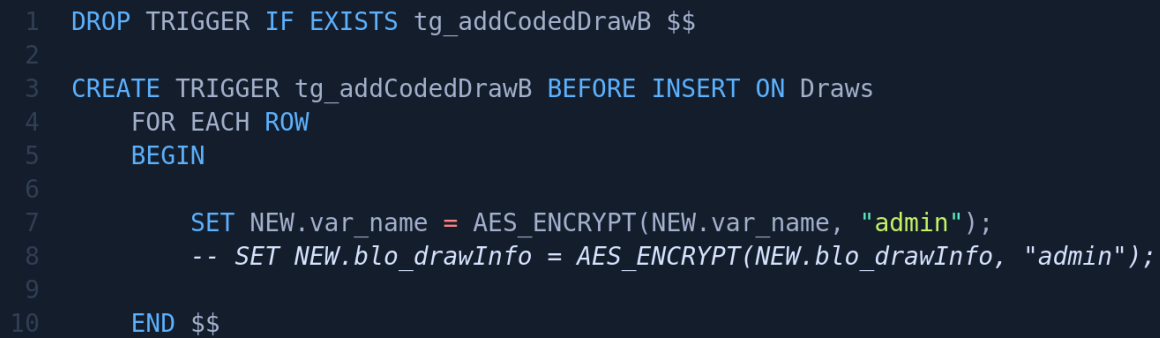
Se definen cuatro atributos para la tabla *Draws* los cuales son:

- **id:** Un atributo de tipo de dato entero con valor autoincremental, definido como la llave primaria de la tabla; este dato es único para cada dibujo generado por los usuarios, que se ingresa en la base de datos y nos permite identificarlo en cualquier otra instancia necesaria.
- **userId:** Atributo de tipo de dato entero no nulo; este dato permite vincular a un usuario con sus dibujos en la tabla *draws* ya que el usuario puede realizar uno o varios dibujos.
- **var_name:** Atributo de tipo de dato varchar no nulo, que corresponde al nombre del dibujo creado por el usuario, este dato es asignado por el usuario, ya sea un usuario normal o el administrador.
- **blo_drawInfo:** Atributo de tipo de dato BLOB no nulo. Este dato contiene el archivo .json encriptado con la información de respaldo de los dibujos en la base de datos A.

Estos atributos son las propiedades de cada dibujo necesarias como respaldo dentro del programa y la base de datos B, los cuales solo pueden ser modificados en caso de que lo haga el administrador del programa.

Análisis de Triggers

Trigger *tg_addCodedDrawB*

A terminal window with a dark background and three colored window control buttons (red, yellow, green) in the top-left corner. It contains SQL code for creating and dropping a trigger. The code is as follows:

```
1 DROP TRIGGER IF EXISTS tg_addCodedDrawB $$
2
3 CREATE TRIGGER tg_addCodedDrawB BEFORE INSERT ON Draws
4   FOR EACH ROW
5   BEGIN
6
7       SET NEW.var_name = AES_ENCRYPT(NEW.var_name, "admin");
8       -- SET NEW.blo_drawInfo = AES_ENCRYPT(NEW.blo_drawInfo, "admin");
9
10  END $$
```

Se elimina el trigger *tg_addCodedDrawB* si es que este existe y posteriormente se crea; este trigger trabaja con la tabla *Draws* de la base de datos B y la acción que realiza es la de encriptar los nuevos datos agregados a la tabla de dibujos cuando detecta una inserción.