



República de Colombia

SENA - Centro De Servicios Financieros - Distrito Capital

Tecnología en Análisis y Desarrollo de Software

Ficha: 2879688

## Creación de la estructura de la BD y aplicación de restricciones

**GA6-220501096-AA2-EV02.**

Brayan Josue Yanez Gonzalez  
PPT: 1246628

## **Introducción**

Este informe presenta la creación de la estructura de una base de datos basada en las características del software en desarrollo. Se detallan las tablas, atributos, tipos de datos y restricciones establecidas, incluyendo llaves primarias y foráneas. Además, se han diseñado los esquemas utilizando MySQL Workbench, garantizando una adecuada normalización y funcionalidad.

## **Objetivo**

El objetivo de este documento es describir la estructura de la base de datos creada para el proyecto formativo, especificando los atributos de cada tabla y las restricciones aplicadas para garantizar la integridad de los datos.

## **Estructura de la Base de Datos**

La base de datos está compuesta por las siguientes tablas:

### **1. Tabla usuarios**

```
CREATE TABLE `tbl_usuarios` (  
  `PKUSU_NCODIGO` int NOT NULL AUTO_INCREMENT,  
  `USU_CNOMBRE` varchar(45) DEFAULT NULL,  
  `USU_CDOCUMENTO` varchar(45) DEFAULT NULL,  
  `USU_CUSUARIO` varchar(45) DEFAULT NULL,  
  `USU_CUSUARIO_AD` varchar(250) DEFAULT NULL,  
  `USU_CROL` varchar(45) DEFAULT NULL,  
  `USU_CPASSWORD` varchar(45) DEFAULT NULL,  
  `USU_TOKEN` varchar(45) DEFAULT NULL,  
  `USU_AUTH_TOKEN` varchar(250) DEFAULT NULL,  
  `USU_CESTADO` varchar(45) DEFAULT NULL,  
  `USU_TIPO_LOGUEO` varchar(45) DEFAULT NULL,  
  `USU_CAUXILIAR` varchar(45) DEFAULT NULL,  
  `USU_LAST_ACTIVITY` timestamp NULL DEFAULT NULL ON UPDATE  
  CURRENT_TIMESTAMP,  
  `USU_CIDIOMA` varchar(3) DEFAULT 'es',  
  `USU_CORREO` varchar(45) DEFAULT NULL,  
  `USU_NCHATS` int DEFAULT '60',
```

```

`USU_DETALLE` varchar(45) DEFAULT NULL,
PRIMARY KEY (`PKUSU_NCODIGO`),
UNIQUE KEY `USU_CUSUARIO_UNIQUE` (`USU_CUSUARIO`)
) ENGINE=InnoDB AUTO_INCREMENT=2 DEFAULT CHARSET=utf8mb4
COLLATE=utf8mb4_0900_ai_ci;

```

#### Atributos:

- id\_usuario: Identificador único del usuario.
- nombre: Nombre del usuario.
- correo: Dirección de correo electrónico única.
- telefono: Número de teléfono.
- fecha\_registro: Fecha de registro del usuario.

#### Restricciones:

- id\_usuario es la llave primaria.
- correo debe ser único.

## 2. Tabla grupos

```

CREATE TABLE `tbl_grupo` (
  `grp_id` int NOT NULL AUTO_INCREMENT,
  `grp_fecha` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP,
  `grp_numero` varchar(255) NOT NULL DEFAULT '-',
  `grp_nombre` varchar(255) NOT NULL DEFAULT '-',
  `grp_detalle` varchar(255) NOT NULL DEFAULT '-',
  `grp_tipo_estado` varchar(45) NOT NULL DEFAULT 'Activo',
  `grp_actualizacion` timestamp NOT NULL DEFAULT
CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
  `grp_fk_usuarios` varchar(45) NOT NULL DEFAULT '-',
  PRIMARY KEY (`grp_id`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4
COLLATE=utf8mb4_0900_ai_ci;

```

#### Atributos:

- id\_grupo: Identificador único del grupo.
- nombre\_grupo: Nombre del grupo.

- descripcion: Descripción del grupo.

**Restricciones:**

- id\_grupo es la llave primaria.

### 3. Tabla mensajes

```
CREATE TABLE mensajes (
    id_mensaje INT AUTO_INCREMENT PRIMARY KEY,
    id_usuario INT,
    id_grupo INT,
    contenido TEXT NOT NULL,
    fecha_envio TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    FOREIGN KEY (id_usuario) REFERENCES usuarios(id_usuario) ON
DELETE CASCADE,
    FOREIGN KEY (id_grupo) REFERENCES grupos(id_grupo) ON DELETE
CASCADE
);
```

**Atributos:**

- id\_mensaje: Identificador único del mensaje.
- id\_usuario: Usuario que envía el mensaje.
- id\_grupo: Grupo al que se envía el mensaje.
- contenido: Contenido del mensaje.
- fecha\_envio: Fecha y hora de envío.

**Restricciones:**

- id\_mensaje es la llave primaria.
- id\_usuario y id\_grupo son llaves foráneas que aseguran la integridad referencial.

#### 4. Tabla chats\_management

```
CREATE TABLE chats_management (  
    id_chat INT AUTO_INCREMENT PRIMARY KEY,  
    id_usuario INT,  
    estado ENUM('activo', 'inactivo') DEFAULT 'activo',  
    FOREIGN KEY (id_usuario) REFERENCES usuarios(id_usuario) ON  
    DELETE CASCADE  
);
```

##### Atributos:

- id\_chat: Identificador único del chat.
- id\_usuario: Usuario que gestiona el chat.
- estado: Estado del chat (activo o inactivo).

##### Restricciones:

- id\_chat es la llave primaria.
- id\_usuario es llave foránea.

#### 5. Tabla line\_profiling

```
CREATE TABLE line_profiling (  
    id_linea INT AUTO_INCREMENT PRIMARY KEY,  
    id_usuario INT,  
    actividad VARCHAR(255) NOT NULL,  
    FOREIGN KEY (id_usuario) REFERENCES usuarios(id_usuario) ON  
    DELETE CASCADE  
);
```

##### Atributos:

- id\_linea: Identificador único de la línea de perfil.
- id\_usuario: Usuario relacionado con la actividad.
- actividad: Descripción de la actividad realizada.

#### **Restricciones:**

- id\_linea es la llave primaria.
- id\_usuario es llave foránea.

#### **6. Tabla bot\_tree**

```
CREATE TABLE bot_tree (  
    id_nodo INT AUTO_INCREMENT PRIMARY KEY,  
    id_padre INT NULL,  
    contenido TEXT NOT NULL,  
    FOREIGN KEY (id_padre) REFERENCES bot_tree(id_nodo) ON DELETE  
    SET NULL  
);
```

#### **Atributos:**

- id\_nodo: Identificador único del nodo del bot.
- id\_padre: Identificador del nodo padre en la jerarquía.
- contenido: Contenido del nodo.

#### **Restricciones:**

- id\_nodo es la llave primaria.
- id\_padre es una llave foránea autorreferencial.

## **2. Creación de la Base de Datos en MySQL Workbench**

Se diseñó la estructura de la base de datos en MySQL Workbench, definiendo las relaciones entre tablas y aplicando las restricciones necesarias para garantizar la integridad de los datos.

### **3. Aplicación de Restricciones**

Se implementaron las siguientes restricciones en la base de datos:

- Llaves primarias en todas las tablas para garantizar la unicidad de los registros.
- Llaves foráneas con opciones ON DELETE CASCADE y ON DELETE SET NULL para mantener la integridad referencial.
- Restricción UNIQUE en correo de la tabla usuarios.
- Restricción ENUM en el campo estado de la tabla chats\_management.

### **Conclusión**

La creación de esta base de datos garantiza una estructura organizada y eficiente para el manejo de información en el sistema en desarrollo. Se han aplicado buenas prácticas de normalización y restricción de datos, asegurando su integridad y coherencia en el almacenamiento. La implementación en MySQL Workbench facilita la visualización y modificación de los esquemas según sea necesario.