



S.I.G.T.O.

Base De Datos

Software Code Builders

Rol	Apellido	Nombre	C.I	Email	Tel/Cel.
Coordinador	Bebekian	Joaquin	5.584.659-6	joacobebekian115@gmail.com	092500221
Sub-Coordinador	Ces	Nicolas	5.585.997-7	nicolasces181@gmail.com	097099331
Integrante 1	Serpa	Ignacio	5.480.675-3	nachoserpa@gmail.com	099760992

Docente: Carambula, Leonardo

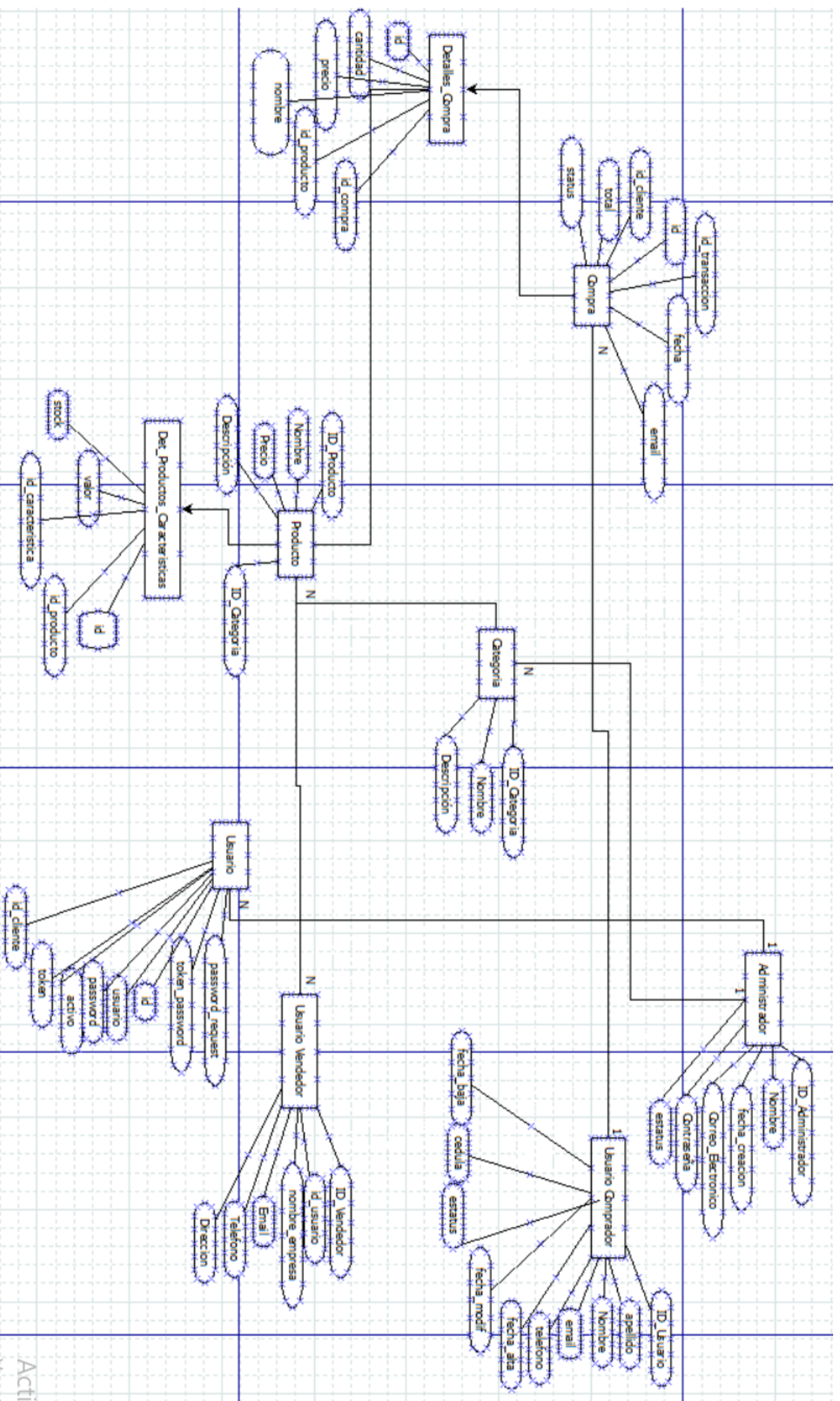
Fecha de culminación

5/11/2024

TERCERA ENTREGA

I.S.B.O.

3BG



Esquema Relacional Normalizado (3FN)

Entidades y Atributos de Base de Datos

1. Administradores

- id (Clave primaria)
- nombre
- email (único)
- password
- fecha_creacion
- estatus

2. Caracteristicas

- id (Clave primaria)
- caracteristica
- activo

3. Clientes

- id (Clave primaria)
- nombre
- apellido
- email (único)
- telefono
- cedula
- estatus
- fecha_alta
- fecha_modif
- fecha_baja

4. Compra

- id (Clave primaria)
- id_transaccion
- fecha
- status
- email
- id_cliente (Clave foránea referenciando Clientes)
- total

5. Detalles_Compra

- id (Clave primaria)
- id_compra (Clave foránea referenciando Compra)
- id_producto (Clave foránea referenciando Productos)
- nombre
- precio
- cantidad

6. Det_Productos_Caracteristicas

- id (Clave primaria)
- id_producto (Clave foránea referenciando Productos)
- id_caracteristica (Clave foránea referenciando Caracteristicas)
- valor

- **stock**

7. Productos

- **id (Clave primaria)**
- **nombre**
- **descripcion**
- **precio**
- **descuento**
- **id_categoria (Clave foránea referenciando Categoria)**
- **activo**
- **fecha_subida**

8. Usuarios

- **id (Clave primaria)**
- **usuario (único)**
- **password**
- **activo**
- **token**
- **token_password**
- **password_request**
- **id_cliente (Clave foránea referenciando Clientes)**

9. Categoria

- **id (Clave primaria)**
- **nombre**
- **descripcion**

Relaciones del DER

1. **Administradores:**
 - **Relación:** Un **Administrador** puede gestionar (crear, editar, eliminar) las **Categorías**.
 - **Tipo:** 1 a N (uno a muchos)
 - **Relación:** Un **Administrador** puede gestionar (crear, editar, eliminar) los **Usuarios**.
 - **Tipo:** 1 a N (uno a muchos)
2. **Clientes:**
 - **Relación:** Un **Cliente** puede realizar muchas **Compras**.
 - **Tipo:** 1 a N (uno a muchos)
3. **Usuario:**
 - **Relación:** Cada **Usuario** está asociado a un **Cliente**.
 - **Tipo:** 1 a 1 (uno a uno)
4. **Empresa (Usuario Vendedor):**
 - **Relación:** Un **Usuario Vendedor** puede publicar muchos **Productos**.
 - **Tipo:** 1 a N (uno a muchos)
5. **Productos:**
 - **Relación:** Cada **Producto** pertenece a una **Categoría**.
 - **Tipo:** N a 1 (muchos a uno)
 - **Relación:** Cada **Producto** puede tener muchas **Det_Productos_Características**.
 - **Tipo:** 1 a N (uno a muchos)
 - **Relación:** Cada **Producto** puede estar presente en muchas **Detalles_Compra**.
 - **Tipo:** 1 a N (uno a muchos)
 - **Relación:** Cada **Producto** es publicado por un **Usuario Vendedor**.
 - **Tipo:** N a 1 (muchos a uno)
6. **Categoría:**
 - **Relación:** Una **Categoría** puede incluir muchos **Productos**.
 - **Tipo:** 1 a N (uno a muchos)
7. **Característica:**
 - **Relación:** Cada **Característica** puede asociarse a muchos **Det_Productos_Características**.
 - **Tipo:** 1 a N (uno a muchos)
8. **Det_Productos_Características:**
 - **Relación:** Cada **Det_Productos_Características** está asociado a un **Producto** y una **Característica**.
 - **Tipo:** N a 1 (muchos a uno)
9. **Compra:**
 - **Relación:** Cada **Compra** es realizada por un **Cliente**.
 - **Tipo:** N a 1 (muchos a uno)
 - **Relación:** Cada **Compra** puede tener muchos **Detalles_Compra**.
 - **Tipo:** 1 a N (uno a muchos)
10. **Detalles_Compra:**
 - **Relación:** Cada **Detalles_Compra** está asociado a una **Compra** y a un **Producto**.
 - **Tipo:** N a 1 (muchos a uno)

Diccionario de Datos

Administradores

Campo	Tipo de Dato	Descripción
id	INT	Identificador único del administrador
nombre	VARCHAR(100)	Nombre del administrador
email	VARCHAR(100)	Correo electrónico del administrador
password	VARCHAR(255)	Contraseña del administrador
fecha_creacion	TIMESTAMP	Fecha de creación del registro
estatus	ENUM('activo', 'inactivo')	Estado del administrador

Clientes

Campo	Tipo de Dato	Descripción
id	INT	Identificador único del cliente
nombre	VARCHAR(50)	Nombre del cliente
apellido	VARCHAR(50)	Apellido del cliente
email	VARCHAR(50)	Correo electrónico del cliente
telefono	VARCHAR(50)	Número de teléfono del cliente
cedula	VARCHAR(20)	Cédula de identidad del cliente
estatus	TINYINT	Estado del cliente (activo/inactivo)
fecha_alta	DATETIME	Fecha de alta del cliente
fecha_modif	DATETIME	Fecha de modificación del registro
fecha_baja	DATETIME	Fecha de baja del cliente

Compra

Campo	Tipo de Dato	Descripción
id	INT	Identificador único de la compra
id_transaccion	VARCHAR(20)	Identificador de la transacción
fecha	DATETIME	Fecha de la compra
status	VARCHAR(20)	Estado de la compra
email	VARCHAR(50)	Correo electrónico del cliente
id_cliente	INT	Identificador del cliente (FK)
total	DECIMAL(10,2)	Total de la compra

Detalles_Compra

Campo	Tipo de Dato	Descripción
id	INT	Identificador único del detalle de compra
id_compra	INT	Identificador de la compra (FK)
id_producto	INT	Identificador del producto (FK)
nombre	VARCHAR(200)	Nombre del producto
precio	DECIMAL(10,2)	Precio del producto
cantidad	INT	Cantidad de productos en la compra

Det_Productos_Caracteristicas

Campo	Tipo de Dato	Descripción
id	INT	Identificador único de la relación
id_producto	INT	Identificador del producto (FK)
id_caracteristica	INT	Identificador de la característica (FK)
valor	VARCHAR(30)	Valor de la característica
stock	INT	Stock disponible del producto

Productos

Campo	Tipo de Dato	Descripción
id	INT	Identificador único del producto
nombre	VARCHAR(200)	Nombre del producto
descripcion	TEXT	Descripción del producto
precio	DECIMAL(10,2)	Precio del producto
descuento	TINYINT	Descuento aplicable
id_categoria	INT	Identificador de la categoría (FK)
activo	INT	Estado del producto (activo/inactivo)
fecha_subida	DATE	Fecha de subida del producto

Usuarios

Campo	Tipo de Dato	Descripción
id	INT	Identificador único del usuario
usuario	VARCHAR(50)	Nombre de usuario
password	VARCHAR(120)	Contraseña del usuario
activo	INT	Estado de actividad del usuario
token	VARCHAR(40)	Token de sesión
token_password	VARCHAR(40)	Token para recuperación de contraseña
password_request	INT	Indicador de solicitud de cambio de contraseña
id_cliente	INT	Identificador del cliente (FK)

Categoria

Campo	Tipo de Dato	Descripción
id	INT	Identificador único de la categoría
nombre	VARCHAR(50)	Nombre de la categoría
descripcion	TEXT	Descripción de la categoría

Sentencias DDL para la Creación de Tablas y Estructuras Necesarias

-- Creación de la tabla Administradores

CREATE TABLE administradores (

id INT PRIMARY KEY AUTO_INCREMENT,

nombre VARCHAR(100) NOT NULL,

email VARCHAR(100) UNIQUE NOT NULL,

password VARCHAR(255) NOT NULL,

fecha_creacion TIMESTAMP DEFAULT CURRENT_TIMESTAMP,


```
    estatus ENUM('activo', 'inactivo') DEFAULT 'activo'
);
```

-- Creación de la tabla Clientes

```
CREATE TABLE clientes (
    id INT PRIMARY KEY AUTO_INCREMENT,
    nombre VARCHAR(50) NOT NULL,
    apellido VARCHAR(50) NOT NULL,
    email VARCHAR(50) UNIQUE NOT NULL,
    telefono VARCHAR(50) NOT NULL,
    cedula VARCHAR(20) NOT NULL,
    estatus TINYINT NOT NULL,
    fecha_alta DATETIME NOT NULL,
    fecha_modif DATETIME DEFAULT NULL,
    fecha_baja DATETIME DEFAULT NULL
);
```

-- Creación de la tabla Usuarios

```
CREATE TABLE usuarios (
    id INT PRIMARY KEY AUTO_INCREMENT,
    usuario VARCHAR(50) UNIQUE NOT NULL,
    password VARCHAR(120) NOT NULL,
    activo INT DEFAULT 0,
    token VARCHAR(40) NOT NULL,
    token_password VARCHAR(40) DEFAULT NULL,
    password_request INT DEFAULT 0,
```

```
id_cliente INT NOT NULL,  
  
FOREIGN KEY (id_cliente) REFERENCES clientes(id)  
  
);
```

-- Creación de la tabla Productos

```
CREATE TABLE productos (  
  
id INT PRIMARY KEY AUTO_INCREMENT,  
  
nombre VARCHAR(200) NOT NULL,  
  
descripcion TEXT NOT NULL,  
  
precio DECIMAL(10,2) NOT NULL,  
  
descuento TINYINT DEFAULT 0,  
  
id_categoria INT NOT NULL,  
  
activo INT NOT NULL,  
  
fecha_subida DATE DEFAULT CURDATE()  
  
);
```

-- Creación de la tabla Caracteristicas

```
CREATE TABLE caracteristicas (  
  
id INT PRIMARY KEY AUTO_INCREMENT,  
  
caracteristica VARCHAR(50) NOT NULL,  
  
activo INT NOT NULL  
  
);
```

-- Creación de la tabla Det_Productos_Caracteristicas

```
CREATE TABLE det_productos_caracteristicas (  
  
id INT PRIMARY KEY AUTO_INCREMENT,
```

```
id_producto INT NOT NULL,  
id_carateristicas INT NOT NULL,  
valor VARCHAR(30) NOT NULL,  
stock INT NOT NULL,  
FOREIGN KEY (id_producto) REFERENCES productos(id),  
FOREIGN KEY (id_carateristicas) REFERENCES caracteristicas(id)  
);
```

-- Creación de la tabla Compra

```
CREATE TABLE compra (  
    id INT PRIMARY KEY AUTO_INCREMENT,  
    id_transaccion VARCHAR(20) NOT NULL,  
    fecha DATETIME NOT NULL,  
    status VARCHAR(20) NOT NULL,  
    email VARCHAR(50) NOT NULL,  
    id_cliente INT NOT NULL,  
    total DECIMAL(10,2) NOT NULL,  
    FOREIGN KEY (id_cliente) REFERENCES clientes(id)  
);
```

-- Creación de la tabla Detalles_Compra

```
CREATE TABLE detalles_compra (  
    id INT PRIMARY KEY AUTO_INCREMENT,  
    id_compra INT NOT NULL,  
    id_producto INT NOT NULL,  
    nombre VARCHAR(200) NOT NULL,
```

```
precio DECIMAL(10,2) NOT NULL,  
cantidad INT NOT NULL,  
FOREIGN KEY (id_compra) REFERENCES compra(id),  
FOREIGN KEY (id_producto) REFERENCES productos(id)  
);
```

-- Creación de la tabla Configuración

```
CREATE TABLE configuracion (  
    id INT PRIMARY KEY AUTO_INCREMENT,  
    nombre VARCHAR(50) NOT NULL,  
    valor TINYTEXT NOT NULL  
);
```

```
);
```

Estudio de los Permisos sobre BD, Tablas y Columnas

1. Administrador:

Permisos:

- **Acceso completo** (**SELECT**, **INSERT**, **UPDATE**, **DELETE**) a todas las tablas de la base de datos.
- Permisos para **crear y eliminar tablas**, modificar la estructura de las tablas y gestionar usuarios (**CREATE**, **DROP**, **ALTER**).
- Justificación: El rol de Administrador necesita control total para gestionar la base de datos y realizar operaciones avanzadas, como cambios en la estructura de la base de datos y la administración de usuarios.

2. Vendedor:

Permisos:

- **SELECT** en las tablas: **productos**, **clientes**, **detalles_compra** y **compra**.
- **INSERT** y **UPDATE** en la tabla **productos** y **detalles_compra** para gestionar el inventario y detalles de ventas.
- Justificación: El rol de Vendedor necesita poder consultar productos, clientes y pedidos, así como actualizar los detalles de los productos y las compras relacionadas.

3. Cliente:

Permisos:

- **SELECT** en las tablas: **productos** y **caracteristicas**.
- **INSERT** en las tablas **compra** y **detalles_compra** para poder realizar compras y agregar detalles de pedidos.
- Justificación: El rol de Cliente debe poder consultar productos y realizar compras, pero no puede modificar datos existentes.

4. Usuario Invitado:

Permisos:

- **SELECT** en las tablas: **productos** y **caracteristicas**.
- Justificación: Los usuarios invitados deben tener acceso de solo lectura para explorar productos y sus características, pero sin realizar modificaciones ni compras.

Gestión de Usuarios y Permisos en la Base de Datos

Para garantizar la seguridad y la correcta gestión de accesos a la base de datos del proyecto, se han definido roles específicos con diferentes niveles de permisos, y se han creado usuarios asociados a estos roles. A continuación, se detallan las sentencias SQL para la creación de usuarios y la asignación de permisos.

1. Creación de Usuarios

-- Creación de usuarios con contraseñas seguras

```
CREATE USER 'admin_user'@'localhost' IDENTIFIED BY 'securepassword';  
CREATE USER 'vendedor_user'@'localhost' IDENTIFIED BY 'securepassword';  
CREATE USER 'cliente_user'@'localhost' IDENTIFIED BY 'securepassword';
```

2. Asignación de Permisos por Rol

Permisos para el Administrador: El administrador tiene permisos completos para gestionar la base de datos y todas sus tablas.

```
GRANT ALL PRIVILEGES ON tienda_online.* TO 'admin_user'@'localhost';
```

Permisos para el Vendedor: El vendedor puede ver y modificar los productos y gestionar las compras.

-- Permisos de lectura y modificación en productos

```
GRANT SELECT, INSERT, UPDATE ON tienda_online.productos TO  
'vendedor_user'@'localhost';
```

-- Permisos para gestionar detalles de compras

```
GRANT SELECT, INSERT, UPDATE ON tienda_online.detalles_compra TO  
'vendedor_user'@'localhost';
```

Permisos para el Cliente: El cliente puede consultar productos y realizar compras.

-- Permisos de lectura en productos

```
GRANT SELECT ON tienda_online.productos TO 'cliente_user'@'localhost';
```

-- Permisos para realizar compras

```
GRANT SELECT, INSERT ON tienda_online.compra TO 'cliente_user'@'localhost';  
GRANT SELECT ON tienda_online.detalles_compra TO 'cliente_user'@'localhost';
```

3. Aplicar los Cambios de Permisos

-- Refrescar los privilegios para que se apliquen los cambios

```
FLUSH PRIVILEGES;
```

Justificación de los Permisos

- **Administrador:** Necesita control total para gestionar la base de datos, modificar estructuras, y realizar operaciones avanzadas.
- **Vendedor:** Debe tener acceso para ver y actualizar productos, así como gestionar detalles de compras asociadas.
- **Cliente:** Solo necesita acceso para consultar productos y registrar nuevas compras.

Respaldo de la Base de Datos Implementada para el Proyecto

Política de Respaldo y Logs:

Para garantizar la integridad y disponibilidad de los datos de la base de datos del proyecto, se ha implementado un sistema de respaldos automáticos mediante el uso de **cron jobs** en un entorno Linux. Este enfoque permite realizar copias de seguridad de forma regular y segura, asegurando la protección de los datos y su disponibilidad en caso de emergencia.

Respaldo Automático Diario:

El sistema está configurado para ejecutar un respaldo completo de la base de datos todos los días a las 2:00 AM. Para ello, se ha desarrollado un script en **Bash** que utiliza el comando **mysqldump** de MySQL para exportar la base de datos a un archivo **.sql**. Este respaldo se almacena en una carpeta designada, y se retienen las copias durante un período de 10 días antes de ser eliminadas para optimizar el uso de almacenamiento.

Código del Script de Respaldo en Bash:

```
#!/bin/bash
FECHA=$(date +%Y-%m-%d_%H-%M-%S)
RUTA_RESPALDO="/ruta/a/tu/carpeta_de_respaldo"
ARCHIVO_RESPALDO="$RUTA_RESPALDO/backup_$FECHA.sql"

# Comando mysqldump para el respaldo
mysqldump -u [usuario] -p[contraseña] [nombre_base_datos] > "$ARCHIVO_RESPALDO"

# Verifica si el respaldo fue exitoso
if [ $? -eq 0 ]; then
    echo "Respaldo realizado con éxito: $ARCHIVO_RESPALDO"
else
    echo "Error al realizar el respaldo"
```

fi

Configuración del Cron Job:

El cron job se configura para ejecutar el script de respaldo diariamente a las 2:00 AM con la siguiente línea en el archivo de cron:

```
0 2 * * * /bin/bash /ruta/a/tu/script_respaldo.sh
```

Consultas SQL para la Base de Datos

1. Conteo de Administradores, Clientes y Vendedores

```
SELECT 'Administrador' AS tipo, COUNT(*) AS cantidad FROM
administradores
UNION ALL
SELECT 'Cliente' AS tipo, COUNT(*) AS cantidad FROM clientes
UNION ALL
SELECT 'Vendedor' AS tipo, COUNT(*) AS cantidad FROM usuario
ORDER BY cantidad DESC;
```

2. Listar Productos en un Pedido Específico

```
SELECT d.id_producto, p.nombre, p.precio
FROM detalles_compra d
JOIN productos p ON d.id_producto = p.id
WHERE d.id_compra = [ID de la compra]; -- Reemplazar con el ID real
del pedido
```

3. Clientes con Mayor Número de Compras

```
SELECT CONCAT(c.nombre, ' ', c.apellido) AS Cliente, COUNT(comp.id)
AS TotalCompras
FROM compra comp
JOIN clientes c ON comp.id_cliente = c.id
GROUP BY c.id
ORDER BY TotalCompras DESC
```



```
LIMIT 20;
```

4. Productos No Vendidos en el Último Mes

```
SELECT p.nombre  
FROM productos p  
LEFT JOIN detalles_compra d ON p.id = d.id_producto  
LEFT JOIN compra comp ON d.id_compra = comp.id  
WHERE comp.fecha IS NULL OR comp.fecha < DATE_SUB(CURDATE(),  
INTERVAL 1 MONTH)  
GROUP BY p.id;
```