



REPÚBLICA BOLIVARIANA DE VENEZUELA

MINISTERIO DEL PODER POPULAR PARA LA EDUCACION UNIVERSITARIA

UNIVERSIDAD NACIONAL EXPERIMENTAL DE TELECOMUNICACIONES E
INFORMATICA

CURSO: **Base de Datos I**

SECCION: 7A

Evaluación 3: Diseño Lógico y Físico de la Base de Datos (Caso 37)

Profesora: Yuly Delgado

Autor:

Darwin Colmenares C.I: 19.064.945

Caracas, febrero de 2025

1. Entidades Principales y sus Atributos

1.1. Donante

- **Atributos:**
 - **numero_donante** (PK): Identificador único del donante.
 - **nombre**: Nombre del donante.
 - **apellidos_razon_social**: Apellidos o razón social del donante.
 - **direccion**: Dirección del donante.
 - **telefono**: Número de teléfono del donante.
 - **correo_electronico**: Correo electrónico del donante (único).
 - **tipo_donante**: Tipo de donante (**individual, empresa, fundación**).
 - **historial_donaciones**: Historial de donaciones realizadas por el donante.
 - **preferencias_comunicacion**: Preferencias de comunicación del donante.

1.2. Proyecto

- **Atributos :**
 - **nombre_proyecto** (PK): Nombre único del proyecto.
 - **descripcion**: Descripción del proyecto.
 - **ubicacion**: Ubicación geográfica del proyecto.
 - **fecha_inicio**: Fecha de inicio del proyecto.
 - **fecha_fin**: Fecha de finalización del proyecto.
 - **presupuesto**: Presupuesto asignado al proyecto.
 - **objetivos**: Objetivos del proyecto.
 - **beneficiarios**: Beneficiarios del proyecto.

1.3. Donación Económica

- **Atributos:**
 - **numero_donacion** (PK): Identificador único de la donación económica.
 - **fecha_donacion** : Fecha en la que se realizó la donación.
 - **importe** : Cantidad donada.

- **metodo_pago** : Método de pago utilizado (tarjeta_credito , tarjeta_debito , transferencia_bancaria , pago_movil , efectivo).
- **estado_donacion** : Estado de la donación (recibida , confirmada , procesada).
- **proyecto_destino** (FK): Proyecto al que se destina la donación.
- **numero_donante** (FK): Donante que realizó la donación.

1.4. Donación en Especie

- **Atributos :**

- **numero_donacion** (PK): Identificador único de la donación en especie.
- **fecha_donacion**: Fecha en la que se realizó la donación.
- **descripcion_articulos**: Descripción de los artículos donados.
- **cantidad**: Cantidad de artículos donados.
- **unidad_medida**: Unidad de medida de los artículos donados.
- **valor_estimado**: Valor estimado de los artículos donados.
- **estado_donacion**: Estado de la donación (recibida, clasificada, distribuida).
- **proyecto_destino** (FK): Proyecto al que se destina la donación.
- **numero_donante** (FK): Donante que realizó la donación.

1.5. Uso de Fondos

- **Atributos :**

- **id_uso_fondos** (PK): Identificador único del registro de uso de fondos.
- **proyecto** (FK): Proyecto al que pertenece el uso de fondos.
- **partida_presupuestaria**: Partida presupuestaria asociada.
- **importe_asignado**: Importe asignado a la partida.
- **importe_gastado**: Importe gastado en la partida.
- **fecha_gasto**: Fecha en la que se realizó el gasto.
- **descripcion_gasto**: Descripción del gasto realizado.

1.6. Comunicación

- **Atributos :**

- **id_comunicacion** (PK): Identificador único de la comunicación.
- **numero_donante** (FK): Donante al que se dirige la comunicación.
- **fecha_envio**: Fecha de envío de la comunicación.
- **tipo_comunicacion**: Tipo de comunicación (**correo_electronico**, **carta**).
- **contenido**: Contenido de la comunicación.
- **estado**: Estado de la comunicación (**enviada**, **pendiente**).

1.7. Voluntario (opcional)

- **Atributos :**
 - **id_voluntario** (PK): Identificador único del voluntario.
 - **nombre**: Nombre del voluntario.
 - **apellidos**: Apellidos del voluntario.
 - **telefono**: Número de teléfono del voluntario.
 - **correo_electronico**: Correo electrónico del voluntario (único).
 - **proyecto_asignado** (FK): Proyecto al que está asignado el voluntario.

1.8. Informe

- **Atributos :**
 - **id_informe** (PK): Identificador único del informe.
 - **tipo_informe**: Tipo de informe generado (**donaciones_por_donante**, **ingresos_y_gastos**, **impacto_proyectos**, **auditoria_cumplimiento**).
 - **fecha_generacion**: Fecha de generación del informe.
 - **descripcion**: Breve descripción del contenido del informe.
 - **donante_relacionado** (FK): Donante relacionado con el informe (opcional).
 - **proyecto_relacionado** (FK): Proyecto relacionado con el informe (opcional).
 - **total_ingresos**: Total de ingresos registrados en el informe (para informes de ingresos y gastos).
 - **total_gastos**: Total de gastos registrados en el informe (para informes de ingresos y gastos).

- **impacto_proyecto**: Descripción del impacto del proyecto (para informes de impacto de proyectos).
 - **observaciones_auditoria**: Observaciones relacionadas con auditorías y cumplimiento normativo.
-

2. Relaciones entre Entidades

1. Donante - Donación Económica

- Relación: **Realiza**
- Cardinalidad: 1:N (Un donante puede realizar varias donaciones económicas, pero cada donación económica pertenece a un solo donante).

2. Donante - Donación en Especie

- Relación: **Realiza**
- Cardinalidad: 1:N (Un donante puede realizar varias donaciones en especie, pero cada donación en especie pertenece a un solo donante).

3. Donación Económica - Proyecto

- Relación: **Se destina a**
- Cardinalidad: N:1 (Varias donaciones económicas pueden destinarse a un proyecto, pero cada donación económica está asociada a un solo proyecto).

4. Donación en Especie - Proyecto

- Relación: **Se destina a**
- Cardinalidad: N:1 (Varias donaciones en especie pueden destinarse a un proyecto, pero cada donación en especie está asociada a un solo proyecto).

5. Proyecto - Uso de Fondos

- Relación: **Gestiona**
- Cardinalidad: 1:N (Un proyecto puede tener varios registros de uso de fondos, pero cada registro de uso de fondos pertenece a un solo proyecto).

6. Donante - Comunicación

- Relación: **Recibe**

- Cardinalidad: 1:N (Un donante puede recibir varias comunicaciones, pero cada comunicación está dirigida a un solo donante).

7. **Voluntario - Proyecto** (opcional)

- Relación: **Participa en**
- Cardinalidad: N:1 (Un voluntario puede participar en un solo proyecto, pero un proyecto puede tener varios voluntarios).

8. **Donante - Informe**

- Relación: **Está relacionado con**
- Cardinalidad: 1:N (Un donante puede estar relacionado con varios informes, pero cada informe está asociado a un solo donante).

9. **Proyecto - Informe**

- Relación: **Está relacionado con**
- Cardinalidad: 1:N (Un proyecto puede estar relacionado con varios informes, pero cada informe está asociado a un solo proyecto).

3. **Diagrama de Entidad-Relación (DER)**

[Donante] --(Realiza)--> [DonacionEconomica]

[Donante] --(Realiza)--> [DonacionEspecie]

[DonacionEconomica] --(Se destina a)--> [Proyecto]

[DonacionEspecie] --(Se destina a)--> [Proyecto]

[Proyecto] --(Gestiona)--> [UsoFondos]

[Donante] --(Recibe)--> [Comunicacion]

[Voluntario] --(Participa en)--> [Proyecto] (opcional)

[Donante] --(Está relacionado con)--> [Informe]

[Proyecto] --(Está relacionado con)--> [Informe]

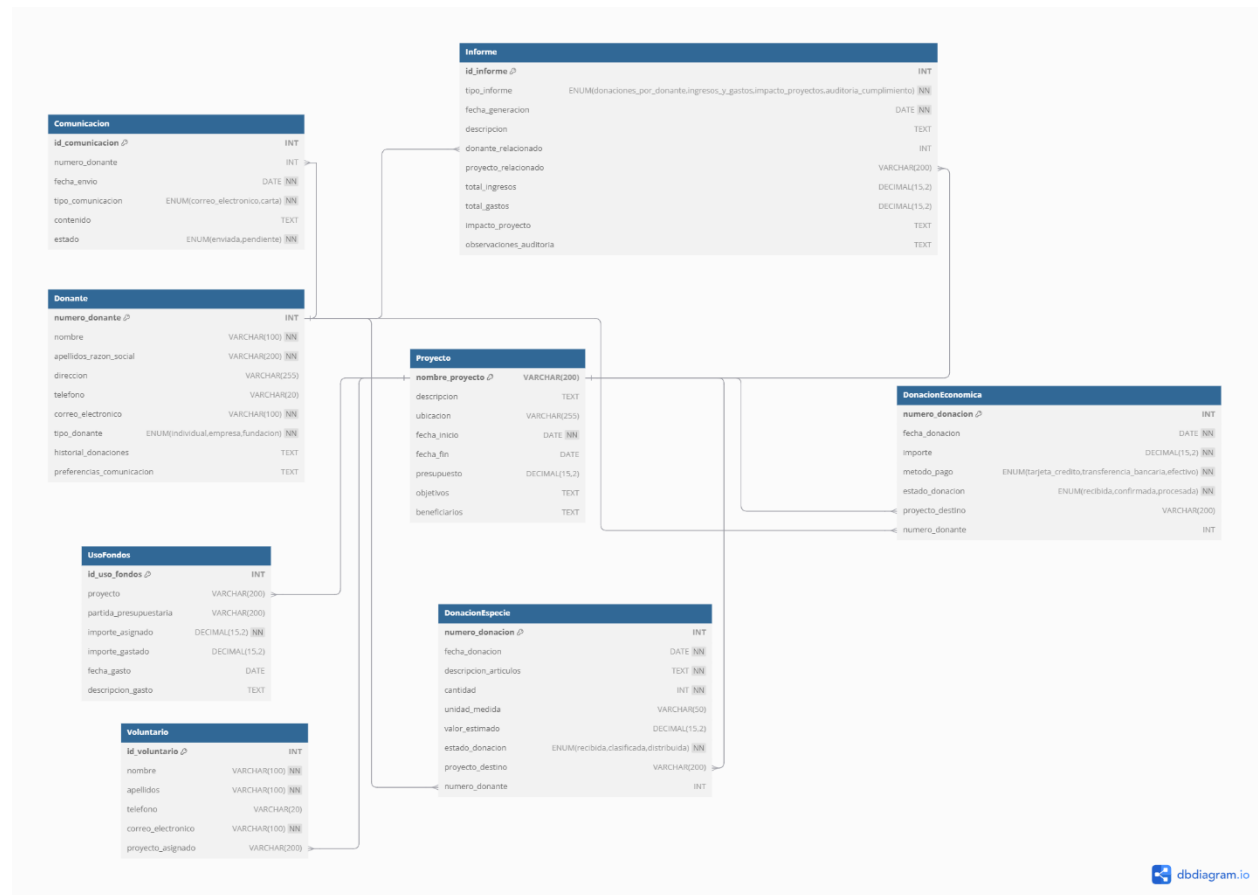
El diagrama de entidad relación fue realizado en el sitio web

[\[dbdiagram.io\]](https://dbdiagram.io/)(<https://dbdiagram.io/>)

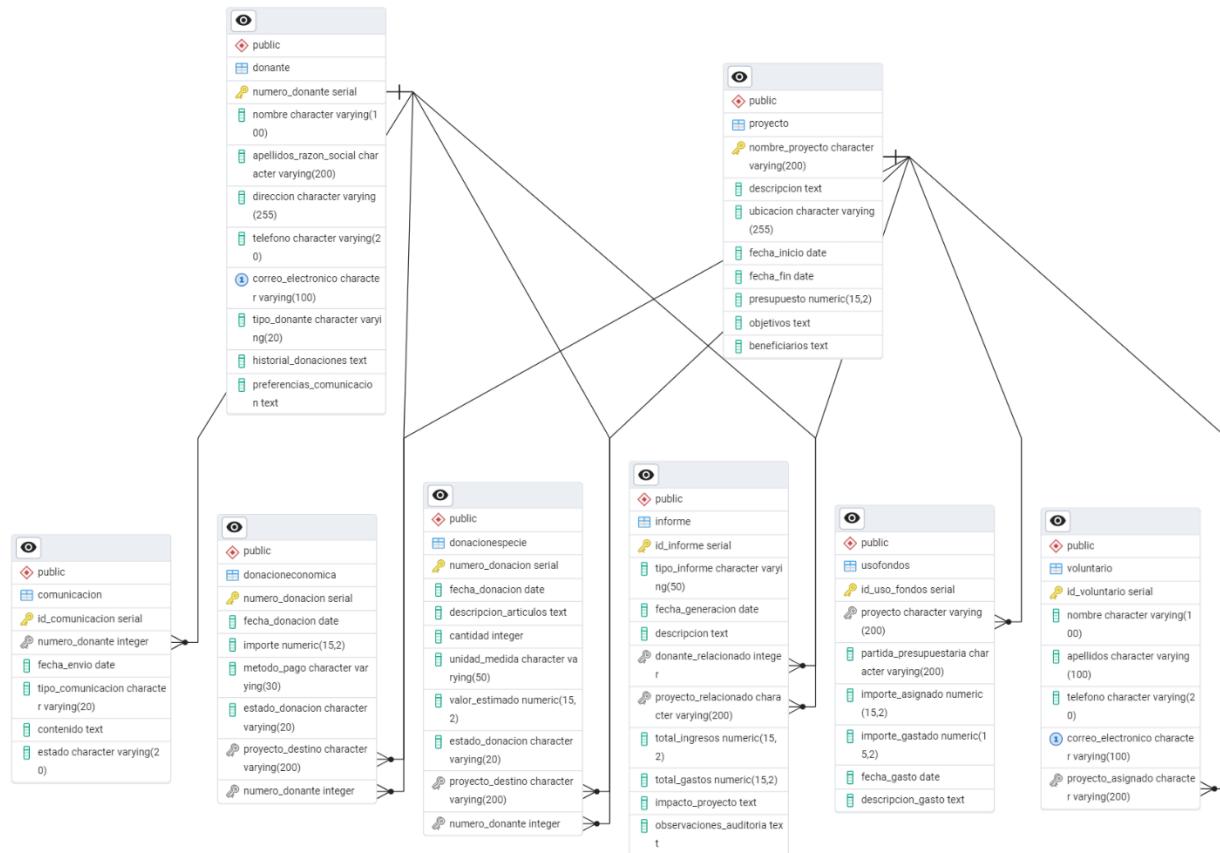
A continuación, se muestra la imagen y el link de la ruta del proyecto en el sitio web mencionado:

Link: [\[DEFINITIVO-ERD-ONG-Ayuda-Accion-DARWIN-COLMENARES-19064945 - dbdiagram.io\]](https://dbdiagram.io/d/DEFINITIVO-ERD-ONG-Ayuda-Accion-DARWIN-COLMENARES-19064945-67a0d253263d6cf9a0ddc044)(<https://dbdiagram.io/d/DEFINITIVO-ERD-ONG-Ayuda-Accion-DARWIN-COLMENARES-19064945-67a0d253263d6cf9a0ddc044>)

Muestra del diagrama en Imagen:



3.1 Diagrama Generado en PostgreSQL



4. Script SQL para crear la BBDD en PostgreSQL

-- Crear la base de datos

CREATE DATABASE ong_ayuda_accion

WITH

OWNER = postgres

TEMPLATE = postgres

ENCODING = 'UTF8'

LC_COLLATE = 'Spanish_Venezuela.1252'

LC_CTYPE = 'Spanish_Venezuela.1252'

LOCALE_PROVIDER = 'libc'

TABLESPACE = pg_default

CONNECTION LIMIT = -1

IS_TEMPLATE = False;

-- Conectar a la base de datos

\c ong_ayuda_accion;

5. Script SQL

-- Creación de la tabla Donante

```
CREATE TABLE Donante (  
    numero_donante SERIAL PRIMARY KEY,  
    nombre VARCHAR(100) NOT NULL,  
    apellidos_razon_social VARCHAR(200) NOT NULL,  
    direccion VARCHAR(255),  
    telefono VARCHAR(20),  
    correo_electronico VARCHAR(100) UNIQUE NOT NULL,  
    tipo_donante VARCHAR(20) CHECK (tipo_donante IN ('individual', 'empresa',  
    'fundacion')) NOT NULL,  
    historial_donaciones TEXT,  
    preferencias_comunicacion TEXT  
);
```

-- Creación de la tabla Proyecto

```
CREATE TABLE Proyecto (  
    nombre_proyecto VARCHAR(200) PRIMARY KEY,  
    descripcion TEXT,  
    ubicacion VARCHAR(255),  
    fecha_inicio DATE NOT NULL,  
    fecha_fin DATE,  
    presupuesto DECIMAL(15, 2),  
    objetivos TEXT,  
    beneficiarios TEXT  
);
```

-- Creación de la tabla DonacionEspecie

```
CREATE TABLE DonacionEspecie (  

```

```

numero_donacion SERIAL PRIMARY KEY,
fecha_donacion DATE NOT NULL,
descripcion_articulos TEXT NOT NULL,
cantidad INT NOT NULL,
unidad_medida VARCHAR(50),
valor_estimado DECIMAL(15, 2),
estado_donacion VARCHAR(20) CHECK (estado_donacion IN ('recibida',
'clasificada', 'distribuida')) NOT NULL,
proyecto_destino VARCHAR(200) REFERENCES Proyecto(nombre_proyecto),
numero_donante INT REFERENCES Donante(numero_donante)
);

```

Se modifica la tabla “donacioneconomica” para agregar nuevos métodos de pago, a continuación, el script:

- Modificar la columna metodo_pago para incluir los nuevos valores

```
ALTER TABLE DonacionEconomica
```

```
DROP CONSTRAINT IF EXISTS donacioneconomica_metodo_pago_check;
```

```
ALTER TABLE DonacionEconomica
```

```
ADD CONSTRAINT donacioneconomica_metodo_pago_check
```

```
CHECK (metodo_pago IN ('tarjeta_credito', 'tarjeta_debito', 'transferencia_bancaria',
'pago_movil', 'efectivo'));
```

La Estructura final queda de la siguiente forma:

- Creación de la tabla DonacionEconomica con los nuevos métodos de pago

```
CREATE TABLE DonacionEconomica (
```

```
numero_donacion SERIAL PRIMARY KEY,
```

```
fecha_donacion DATE NOT NULL,
```

```
importe DECIMAL(15, 2) NOT NULL,
```

```
metodo_pago VARCHAR(30) CHECK (metodo_pago IN ('tarjeta_credito',
'tarjeta_debito', 'transferencia_bancaria', 'pago_movil', 'efectivo')) NOT NULL,
```

```
estado_donacion VARCHAR(20) CHECK (estado_donacion IN ('recibida', 'confirmada',
'procesada')) NOT NULL,
```

```

proyecto_destino VARCHAR(200) REFERENCES Proyecto(nombre_proyecto),
numero_donante INT REFERENCES Donante(numero_donante)
);

-- Creación de la tabla DonacionEspecie
CREATE TABLE DonacionEspecie (
numero_donacion SERIAL PRIMARY KEY,
fecha_donacion DATE NOT NULL,
descripcion_articulos TEXT NOT NULL,
cantidad INT NOT NULL,
unidad_medida VARCHAR(50),
valor_estimado DECIMAL(15, 2),
estado_donacion VARCHAR(20) CHECK (estado_donacion IN ('recibida', 'clasificada',
'distribuida')) NOT NULL,
proyecto_destino VARCHAR(200) REFERENCES Proyecto(nombre_proyecto),
numero_donante INT REFERENCES Donante(numero_donante)
);

-- Creación de la tabla UsoFondos
CREATE TABLE UsoFondos (
id_uso_fondos SERIAL PRIMARY KEY,
proyecto VARCHAR(200) REFERENCES Proyecto(nombre_proyecto),
partida_presupuestaria VARCHAR(200),
importe_asignado DECIMAL(15, 2) NOT NULL,
importe_gastado DECIMAL(15, 2),
fecha_gasto DATE,
descripcion_gasto TEXT
);

-- Creación de la tabla Comunicacion
CREATE TABLE Comunicacion (
id_comunicacion SERIAL PRIMARY KEY,

```

```

numero_donante INT REFERENCES Donante(numero_donante),
fecha_envio DATE NOT NULL,
tipo_comunicacion VARCHAR(20) CHECK (tipo_comunicacion IN ('correo_electronico',
'carta')) NOT NULL,
contenido TEXT,
estado VARCHAR(20) CHECK (estado IN ('enviada', 'pendiente')) NOT NULL
);

```

-- Creación de la tabla Voluntario (opcional)

```

CREATE TABLE Voluntario (
id_voluntario SERIAL PRIMARY KEY,
nombre VARCHAR(100) NOT NULL,
apellidos VARCHAR(100) NOT NULL,
telefono VARCHAR(20),
correo_electronico VARCHAR(100) UNIQUE NOT NULL,
proyecto_asignado VARCHAR(200) REFERENCES Proyecto(nombre_proyecto)
);

```

-- Creación de la tabla Informe

```

CREATE TABLE Informe (
id_informe SERIAL PRIMARY KEY,
tipo_informe VARCHAR(50) CHECK (tipo_informe IN ('donaciones_por_donante',
'ingresos_y_gastos', 'impacto_proyectos', 'auditoria_cumplimiento')) NOT NULL,
fecha_generacion DATE NOT NULL,
descripcion TEXT,
donante_relacionado INT REFERENCES Donante(numero_donante),
proyecto_relacionado VARCHAR(200) REFERENCES Proyecto(nombre_proyecto),
total_ingresos DECIMAL(15, 2),
total_gastos DECIMAL(15, 2),
impacto_proyecto TEXT,
observaciones_auditoria TEXT
);

```

);

6. Explicación del Script

6.1. Tabla Donante

Descripción:

Almacena la información de los donantes, incluyendo datos personales y preferencias de comunicación.

Estructura :

- **numero_donante SERIAL PRIMARY KEY :**
 - Identificador único del donante.
 - **SERIAL** es un tipo de dato en PostgreSQL que genera automáticamente valores autoincrementales.
- **nombre VARCHAR(100) NOT NULL :**
 - Nombre del donante (obligatorio).
- **apellidos_razon_social VARCHAR(200) NOT NULL :**
 - Apellidos o razón social del donante (obligatorio).
- **direccion VARCHAR(255) :**
 - Dirección del donante (opcional).
- **telefono VARCHAR(20) :**
 - Número de teléfono del donante (opcional).
- **correo_electronico VARCHAR(100) UNIQUE NOT NULL :**
 - Correo electrónico del donante (único y obligatorio).
- **tipo_donante VARCHAR(20) CHECK (tipo_donante IN ('individual', 'empresa', 'fundacion')) NOT NULL :**
 - Tipo de donante (**individual**, **empresa** o **fundación**), validado mediante una restricción **CHECK**.
- **historial_donaciones TEXT :**
 - Historial de donaciones realizadas por el donante (opcional).
- **preferencias_comunicacion TEXT :**
 - Preferencias de comunicación del donante (opcional).

Relaciones :

- Relacionada con **DonacionEconomica** y **DonacionEspecie** mediante la clave foránea **numero_donante**.
 - Relacionada con **Comunicacion** mediante la clave foránea **numero_donante**.
 - Relacionada con **Informe** mediante la clave foránea **donante_relacionado**.
-

6.2. Tabla Proyecto

Descripción :

Almacena la información de los proyectos gestionados por la ONG.

Estructura :

- **nombre_proyecto VARCHAR(200) PRIMARY KEY :**
 - Nombre único del proyecto (clave primaria).
- **descripcion TEXT :**
 - Descripción del proyecto (opcional).
- **ubicacion VARCHAR(255) :**
 - Ubicación geográfica del proyecto (opcional).
- **fecha_inicio DATE NOT NULL :**
 - Fecha de inicio del proyecto (obligatoria).
- **fecha_fin DATE :**
 - Fecha de finalización del proyecto (opcional).
- **presupuesto DECIMAL(15, 2) :**
 - Presupuesto asignado al proyecto (opcional).
- **objetivos TEXT :**
 - Objetivos del proyecto (opcional).
- **beneficiarios TEXT :**
 - Beneficiarios del proyecto (opcional).

Relaciones :

- Relacionada con **DonacionEconomica** y **DonacionEspecie** mediante la clave foránea **proyecto_destino**.
- Relacionada con **UsoFondos** mediante la clave foránea **proyecto**.

- Relacionada con **Voluntario** mediante la clave foránea **proyecto_asignado**.
- Relacionada con **Informe** mediante la clave foránea **proyecto_relacionado**.

6.3. Tabla DonacionEconomica

Descripción :

Registra las donaciones en dinero realizadas por los donantes.

Estructura :

- **numero_donacion SERIAL PRIMARY KEY :**
 - Identificador único de la donación económica.
- **fecha_donacion DATE NOT NULL :**
 - Fecha en la que se realizó la donación (obligatoria).
- **importe DECIMAL(15, 2) NOT NULL :**
 - Cantidad donada (obligatoria).
- **metodo_pago VARCHAR(30) CHECK (metodo_pago IN ('tarjeta_credito', 'tarjeta_debito', 'transferencia_bancaria', 'pago_movil', 'efectivo')) NOT NULL :**
 - Método de pago utilizado (tarjeta_credito, tarjeta_debito, transferencia_bancaria, pago_movil, efectivo), validado mediante una restricción **CHECK**.
- **estado_donacion VARCHAR(20) CHECK (estado_donacion IN ('recibida', 'confirmada', 'procesada')) NOT NULL :**
 - Estado de la donación (recibida, confirmada, procesada), validado mediante una restricción **CHECK**.
- **proyecto_destino VARCHAR(200) REFERENCES Proyecto(nombre_proyecto) :**
 - Proyecto al que se destina la donación (clave foránea).
- **numero_donante INT REFERENCES Donante(numero_donante) :**
 - Donante que realizó la donación (clave foránea).

Relaciones :

- Relacionada con **Donante** mediante la clave foránea **numero_donante**.
- Relacionada con **Proyecto** mediante la clave foránea **proyecto_destino**.

6.4. Tabla DonacionEspecie

Descripción :

Registra las donaciones en especie realizadas por los donantes.

Estructura :

- **numero_donacion SERIAL PRIMARY KEY :**
 - Identificador único de la donación en especie.
- **fecha_donacion DATE NOT NULL :**
 - Fecha en la que se realizó la donación (obligatoria).
- **descripcion_articulos TEXT NOT NULL :**
 - Descripción de los artículos donados (obligatoria).
- **cantidad INT NOT NULL :**
 - Cantidad de artículos donados (obligatoria).
- **unidad_medida VARCHAR(50) :**
 - Unidad de medida de los artículos donados (opcional).
- **valor_estimado DECIMAL(15, 2) :**
 - Valor estimado de los artículos donados (opcional).
- **estado_donacion VARCHAR(20) CHECK (estado_donacion IN ('recibida', 'clasificada', 'distribuida')) NOT NULL :**
 - Estado de la donación (**recibida**, **clasificada**, **distribuida**), validado mediante una restricción **CHECK**.
- **proyecto_destino VARCHAR(200) REFERENCES Proyecto(nombre_proyecto) :**
 - Proyecto al que se destina la donación (clave foránea).
- **numero_donante INT REFERENCES Donante(numero_donante) :**
 - Donante que realizó la donación (clave foránea).

Relaciones :

- Relacionada con **Donante** mediante la clave foránea **numero_donante**.
- Relacionada con **Proyecto** mediante la clave foránea **proyecto_destino**.

6.5. Tabla UsoFondos

Descripción :

Permite rastrear cómo se utilizan los fondos asignados a cada proyecto.

Estructura :

- **id_uso_fondos SERIAL PRIMARY KEY :**
 - Identificador único del registro de uso de fondos.
- **proyecto VARCHAR(200) REFERENCES Proyecto(nombre_proyecto) :**
 - Proyecto al que pertenece el uso de fondos (clave foránea).
- **partida_presupuestaria VARCHAR(200) :**
 - Partida presupuestaria asociada (opcional).
- **importe_asignado DECIMAL(15, 2) NOT NULL :**
 - Importe asignado a la partida (obligatorio).
- **importe_gastado DECIMAL(15, 2) :**
 - Importe gastado en la partida (opcional).
- **fecha_gasto DATE :**
 - Fecha en la que se realizó el gasto (opcional).
- **descripcion_gasto TEXT :**
 - Descripción del gasto realizado (opcional).

Relaciones :

- Relacionada con **Proyecto** mediante la clave foránea **proyecto**.

6.6. Tabla Comunicacion

Descripción :

Registra las comunicaciones enviadas a los donantes.

Estructura :

- **id_comunicacion SERIAL PRIMARY KEY :**
 - Identificador único de la comunicación.

- **numero_donante INT REFERENCES Donante(numero_donante) :**
 - Donante al que se dirige la comunicación (clave foránea).
- **fecha_envio DATE NOT NULL :**
 - Fecha de envío de la comunicación (obligatoria).
- **tipo_comunicacion VARCHAR(20) CHECK (tipo_comunicacion IN ('correo_electronico', 'carta')) NOT NULL :**
 - Tipo de comunicación (**correo_electronico**, **carta**), validado mediante una restricción **CHECK**.
- **contenido TEXT :**
 - Contenido de la comunicación (opcional).
- **estado VARCHAR(20) CHECK (estado IN ('enviada', 'pendiente')) NOT NULL :**
 - Estado de la comunicación (**enviada**, **pendiente**), validado mediante una restricción **CHECK**.

Relaciones :

- Relacionada con **Donante** mediante la clave foránea **numero_donante**.

6.7. Tabla Voluntario (opcional)

Descripción :

Almacena la información de los voluntarios y el proyecto al que están asignados.

Estructura :

- **id_voluntario SERIAL PRIMARY KEY :**
 - Identificador único del voluntario.
- **nombre VARCHAR(100) NOT NULL :**
 - Nombre del voluntario (obligatorio).
- **apellidos VARCHAR(100) NOT NULL :**
 - Apellidos del voluntario (obligatorio).
- **telefono VARCHAR(20) :**
 - Número de teléfono del voluntario (opcional).
- **correo_electronico VARCHAR(100) UNIQUE NOT NULL :**

- Correo electrónico del voluntario (único y obligatorio).
- **proyecto_asignado VARCHAR(200) REFERENCES Proyecto(nombre_proyecto) :**
 - Proyecto al que está asignado el voluntario (clave foránea).

Relaciones :

- Relacionada con **Proyecto** mediante la clave foránea **proyecto_asignado**.

6.8. Tabla Informe

Descripción :

Almacena los detalles de los informes generados por la ONG.

Estructura :

- **id_informe SERIAL PRIMARY KEY :**
 - Identificador único del informe.
- **tipo_informe VARCHAR(50) CHECK (tipo_informe IN ('donaciones_por_donante', 'ingresos_y_gastos', 'impacto_proyectos', 'auditoria_cumplimiento')) NOT NULL :**
 - Tipo de informe generado (donaciones_por_donante, ingresos_y_gastos, impacto_proyectos, a auditoria_cumplimiento), validado mediante una restricción **CHECK**.
- **fecha_generacion DATE NOT NULL :**
 - Fecha de generación del informe (obligatoria).
- **descripcion TEXT :**
 - Breve descripción del contenido del informe (opcional).
- **donante_relacionado INT REFERENCES Donante(numero_donante) :**
 - Donante relacionado con el informe (clave foránea, opcional).
- **proyecto_relacionado VARCHAR(200) REFERENCES Proyecto(nombre_proyecto) :**
 - Proyecto relacionado con el informe (clave foránea, opcional).
- **total_ingresos DECIMAL(15, 2) :**
 - Total de ingresos registrados en el informe (opcional).

- **total_gastos DECIMAL(15, 2) :**
 - Total de gastos registrados en el informe (opcional).
- **impacto_proyecto TEXT :**
 - Descripción del impacto del proyecto (opcional).
- **observaciones_auditoria TEXT :**
 - Observaciones relacionadas con auditorías y cumplimiento normativo (opcional).

Relaciones :

- Relacionada con **Donante** mediante la clave foránea **donante_relacionado** (opcional).
- Relacionada con **Proyecto** mediante la clave foránea **proyecto_relacionado** (opcional).

7. Resumen de la Explicación del Script

1. Tipos de Datos y Autoincremento :

- Se utiliza **SERIAL** en lugar de **AUTO_INCREMENT** para generar valores autoincrementales.
- Los tipos **VARCHAR**, **TEXT**, **DATE** y **DECIMAL** se usan para definir atributos como nombres, descripciones, fechas e importes.

2. Restricciones de Validación :

- Se emplean restricciones **CHECK** para limitar los valores permitidos en columnas como **tipo_donante**, **metodo_pago**, **estado_donacion** y **tipo_comunicacion**.
- Ejemplo: **tipo_donante** solo puede ser 'individual', 'empresa' o 'fundación'.

3. Claves Primarias y Foráneas :

- Cada tabla tiene una clave primaria (**PRIMARY KEY**) para identificar registros únicos.
- Las relaciones entre tablas se establecen mediante claves foráneas (**REFERENCES**), asegurando la integridad referencial.

4. Tablas Principales :

- **Donante** : Almacena datos personales y preferencias de comunicación de los donantes.
- **Proyecto** : Registra información sobre los proyectos gestionados por la ONG.
- **DonacionEconomica y DonacionEspecie** : Registran las donaciones en dinero y en especie, respectivamente, vinculadas a donantes y proyectos.
- **UsoFondos** : Rastrea cómo se utilizan los fondos asignados a cada proyecto.
- **Comunicacion** : Registra las comunicaciones enviadas a los donantes.
- **Voluntario (opcional)** : Almacena información de los voluntarios y su asignación a proyectos.
- **Informe** : Almacena detalles de los informes generados, relacionados opcionalmente con donantes y proyectos.

5. Relaciones entre Tablas :

- Las relaciones son consistentes y bien definidas:
 - **Donante** está relacionado con **DonacionEconomica**, **DonacionEspecie**, **Comunicacion** e **Informe**.
 - **Proyecto** está relacionado con **DonacionEconomica**, **DonacionEspecie**, **UsoFondos**, **Voluntario** e **Informe**.

6. Consideraciones Adicionales :

- Se incluyen campos opcionales (**NULL**) para datos que no siempre son obligatorios.
- Las restricciones garantizan la validez de los datos, como importes positivos o estados específicos.

8. Funcionalidades Implementadas

8.1. Gestión de Donantes

- **Registro de Donantes** :
 - Se pueden registrar nuevos donantes con información personal (nombre, apellidos/razón social, dirección, teléfono, correo electrónico) y preferencias de comunicación.
- **Segmentación de Donantes** :

- Los donantes se clasifican por tipo (**individual, empresa, fundación**) y se pueden segmentar según su historial de donaciones o frecuencia de contribuciones.
- **Consulta y Actualización :**
 - Permite consultar y modificar los datos de los donantes, incluyendo su historial de donaciones y preferencias.

8.2. Gestión de Donaciones

- **Donaciones Económicas :**
 - Registro de donaciones en dinero con detalles como fecha, importe, método de pago (**tarjeta_credito, transferencia_bancaria, efectivo**) y estado (**recibida, confirmada, procesada**).
 - Asignación de donaciones a proyectos específicos.
- **Donaciones en Especie :**
 - Registro de donaciones en especie con descripción de los artículos, cantidad, unidad de medida, valor estimado y estado (**recibida, clasificada, distribuida**).
 - Asignación de donaciones en especie a proyectos específicos.
- **Seguimiento de Donaciones :**
 - Rastreo del estado de las donaciones desde su recepción hasta su procesamiento o distribución.

8.3. Gestión de Proyectos

- **Registro de Proyectos :**
 - Creación de nuevos proyectos con información detallada (nombre, descripción, ubicación, fechas de inicio y fin, presupuesto, objetivos y beneficiarios).
- **Asignación de Recursos :**
 - Vinculación de donaciones (económicas y en especie) a proyectos específicos.
- **Seguimiento del Progreso :**
 - Monitoreo del uso de fondos y recursos asignados a cada proyecto.

8.4. Seguimiento del Uso de Fondos

- **Registro de Gastos :**

- Registro detallado de los gastos realizados en cada proyecto, incluyendo partida presupuestaria, importe asignado, importe gastado, fecha y descripción.
- **Generación de Informes Financieros :**
 - Facilita la creación de informes sobre ingresos y gastos por proyecto.

8.5. Comunicación con Donantes

- **Envío de Comunicaciones :**
 - Registro de comunicaciones enviadas a los donantes (correos electrónicos, cartas) con detalles como fecha, tipo de comunicación, contenido y estado (**enviada, pendiente**).
- **Personalización :**
 - Las comunicaciones pueden personalizarse según las preferencias de los donantes.
- **Informes de Impacto :**
 - Envío de informes sobre el impacto de las donaciones en los proyectos.

8.6. Generación de Informes

- **Tipos de Informes :**
 - **Donaciones por Donante :** Resumen de las donaciones realizadas por cada donante.
 - **Ingresos y Gastos :** Detalle de los ingresos recibidos y los gastos realizados en cada proyecto.
 - **Impacto de Proyectos :** Evaluación del impacto generado por los proyectos.
 - **Auditorías y Cumplimiento Normativo :** Informes para auditorías internas y externas.
- **Relaciones Opcionales :**
 - Los informes pueden estar relacionados con donantes específicos o proyectos específicos.

8.7. Gestión de Voluntarios (Opcional)

- **Registro de Voluntarios :**
 - Almacenamiento de información personal de los voluntarios (nombre, apellidos, teléfono, correo electrónico).

- **Asignación a Proyectos :**
 - Vinculación de voluntarios a proyectos específicos.
- **Seguimiento de Participación :**
 - Monitoreo de la participación de los voluntarios en los proyectos.

8.8. Plataforma Online (Extensible)

- **Acceso para Donantes :**
 - Los donantes pueden consultar el estado de sus donaciones y recibir actualizaciones sobre los proyectos.
- **Integración con Redes Sociales :**
 - Difusión de campañas de recaudación de fondos mediante redes sociales.

8.9. Seguridad y Cumplimiento

- **Protección de Datos :**
 - Implementación de medidas para proteger la información personal de los donantes y los datos financieros.
- **Cumplimiento Normativo :**
 - Generación de informes para cumplir con requisitos legales y regulatorios.

9. Consideraciones Adicionales

9.1. Índices y Optimización

- **Índices en Columnas Frecuentes :**
 - Crear índices en columnas que se consulten con frecuencia, como **fecha_donacion**, **correo_electronico**, **nombre_proyecto** y **estado_donacion**. Esto mejora el rendimiento de las consultas.

Ejemplo:

```
CREATE INDEX idx_fecha_donacion ON DonacionEconomica(fecha_donacion);
```

- **Optimización de Consultas :**
 - Utilizar consultas eficientes y evitar cargas innecesarias al sistema, especialmente cuando se trabaja con grandes volúmenes de datos.

9.2. Restricciones de Negocio (REVISAR ESTO A VER SI SE DEJA O SE ELIMINA)

- **Validación de Datos :**

- Asegurarse de que los importes sean positivos mediante restricciones **CHECK**.

ALTER TABLE DonacionEconomica ADD CONSTRAINT chk_importe_positivo CHECK (importe > 0);

- **Fechas Coherentes :**
 - Validar que las fechas sean lógicas (por ejemplo, **fecha_inicio** debe ser anterior a **fecha_fin**).
- **Estados Válidos :**
 - Garantizar que los estados de las donaciones (**recibida, confirmada, procesada**) y las comunicaciones (**enviada, pendiente**) sigan flujos predefinidos.

9.3. Seguridad

- **Protección de Datos Sensibles :**
 - Cifrar datos sensibles como correos electrónicos y números de teléfono para proteger la privacidad de los donantes.
- **Control de Acceso :**
 - Implementar roles y permisos para restringir el acceso a ciertas funcionalidades según el usuario (administradores, voluntarios, etc.).
- **Cumplimiento de Normativas :**
 - Asegurarse de cumplir con regulaciones como GDPR (en Europa) o CCPA (en California) para el manejo de datos personales.

9.4. Escalabilidad

- **Manejo de Grandes Volúmenes de Datos :**
 - Diseñar la base de datos para manejar un crecimiento significativo en el número de donantes, proyectos y donaciones.
 - Considerar técnicas como particionamiento (**partitioning**) o sharding si es necesario.
- **Replicación y Alta Disponibilidad :**
 - Configurar réplicas de la base de datos para garantizar disponibilidad durante picos de tráfico o fallos del sistema.

9.5. Copias de Seguridad y Recuperación

- **Copias de Seguridad Automáticas :**

- Programar copias de seguridad periódicas para evitar la pérdida de datos en caso de fallos.
- **Plan de Recuperación :**
 - Implementar un plan de recuperación ante desastres para restaurar rápidamente la base de datos en caso de incidentes.

9.6. Integración con Herramientas Externas

- **Plataformas de Pago Online :**
 - Integrar servicios de pago como PayPal, Stripe o MercadoPago para facilitar las donaciones económicas.
- **Redes Sociales :**
 - Conectar el sistema con redes sociales para promover campañas de recaudación de fondos y difundir informes de impacto.
- **Herramientas de Reportes :**
 - Usar herramientas como Power BI, Tableau o Metabase para generar informes visuales avanzados.

9.7. Automatización

- **Generación Automática de Informes :**
 - Crear procedimientos almacenados o scripts para automatizar la generación de informes periódicos.
- **Notificaciones Automáticas :**
 - Implementar notificaciones automáticas (correos electrónicos o SMS) para informar a los donantes sobre el estado de sus donaciones o enviarles informes de impacto.

9.8. Interfaz de Usuario

- **Usabilidad :**
 - Diseñar una interfaz intuitiva y fácil de usar tanto para el personal de la ONG como para los donantes.
- **Accesibilidad :**
 - Asegurarse de que la plataforma sea accesible para personas con discapacidades, cumpliendo con estándares como WCAG.

9.9. Pruebas y Mantenimiento

- **Pruebas de Carga :**

- Realizar pruebas de carga para asegurarse de que el sistema pueda manejar un gran volumen de usuarios y transacciones simultáneas.
- **Mantenimiento Regular :**
 - Realizar mantenimiento preventivo, como la limpieza de datos obsoletos y la actualización de software.

9.10. Cumplimiento Legal y Auditorías

- **Transparencia :**
 - Generar informes detallados para demostrar cómo se utilizan los fondos donados, cumpliendo con los requisitos de transparencia.
- **Auditorías :**
 - Facilitar auditorías internas y externas mediante informes claros y bien estructurados.

9.11. Internacionalización

- **Soporte Multilingüe :**
 - Si la ONG opera en varios países, implementar soporte para múltiples idiomas en la plataforma online y las comunicaciones.
- **Monedas y Tasas de Cambio :**
 - Manejar diferentes monedas y tasas de cambio para donaciones internacionales.

10. Vistas SQL

10.1. Vista: Donaciones Totales por Donante

Esta vista muestra el total de donaciones económicas realizadas por cada donante.

```
CREATE VIEW vw_donaciones_totales_por_donante AS
SELECT
d.numero_donante,
d.nombre || ' ' || d.apellidos_razon_social AS nombre_completo,
SUM(de.importe) AS total_donado
FROM
Donante d
LEFT JOIN
```

DonacionEconomica de ON d.numero_donante = de.numero_donante

GROUP BY

d.numero_donante, d.nombre, d.apellidos_razon_social;

Explicación :

- Combina las tablas **Donante** y **DonacionEconomica**.
- Calcula el total de donaciones económicas (**SUM(de.importe)**) agrupadas por donante.
- Incluye tanto el nombre como los apellidos/razón social del donante.

10.2. Vista: Gastos Totales por Proyecto

Esta vista muestra el total de gastos registrados en cada proyecto.

CREATE VIEW vw_gastos_totales_por_proyecto AS

SELECT

p.nombre_proyecto,

p.descripcion,

SUM(uf.importe_gastado) AS total_gastado

FROM

Proyecto p

LEFT JOIN

UsoFondos uf ON p.nombre_proyecto = uf.proyecto

GROUP BY

p.nombre_proyecto, p.descripcion;

Explicación :

- Combina las tablas **Proyecto** y **UsoFondos**.
- Calcula el total de gastos (**SUM(uf.importe_gastado)**) agrupados por proyecto.
- Incluye la descripción del proyecto para mayor contexto.

10.3. Vista: Impacto de Proyectos

Esta vista muestra el impacto de los proyectos basado en las donaciones recibidas y los gastos realizados.

CREATE VIEW vw_impacto_proyectos AS

```

SELECT
p.nombre_proyecto,
p.objetivos,
COALESCE(SUM(de.importe), 0) AS total_ingresos,
COALESCE(SUM(uf.importe_gastado), 0) AS total_gastos,
COALESCE(SUM(de.importe), 0) - COALESCE(SUM(uf.importe_gastado), 0) AS
saldo_restante
FROM
Proyecto p
LEFT JOIN
DonacionEconomica de ON p.nombre_proyecto = de.proyecto_destino
LEFT JOIN
UsoFondos uf ON p.nombre_proyecto = uf.proyecto
GROUP BY
p.nombre_proyecto, p.objetivos;

```

Explicación :

- Combina las tablas **Proyecto**, **DonacionEconomica** y **UsoFondos**.
- Calcula los ingresos totales (**SUM(de.importe)**), los gastos totales (**SUM(uf.importe_gastado)**) y el saldo restante (**ingresos - gastos**).
- Usa **COALESCE** para manejar valores nulos (por ejemplo, si un proyecto no tiene donaciones o ga

10.4. Vista: Comunicaciones Pendientes

Esta vista muestra las comunicaciones pendientes que aún no han sido enviadas.

```

CREATE VIEW vw_comunicaciones_pendientes AS
SELECT
c.id_comunicacion,
d.nombre || ' ' || d.apellidos_razon_social AS nombre_donante,
c.tipo_comunicacion,
c.fecha_envio,

```

```

c.estado
FROM
Comunicacion c
JOIN
Donante d ON c.numero_donante = d.numero_donante
WHERE
c.estado = 'pendiente';

```

Explicación :

- Combina las tablas **Comunicacion** y **Donante**.
- Filtra las comunicaciones con estado '**pendiente**'.
- Muestra el nombre del donante y detalles de la comunicación.

10.5. Vista: Voluntarios Asignados a Proyectos

Esta vista muestra los voluntarios asignados a cada proyecto.

```

CREATE VIEW vw_voluntarios_por_proyecto AS
SELECT
v.id_voluntario,
v.nombre || ' ' || v.apellidos AS nombre_voluntario,
v.correo_electronico,
p.nombre_proyecto,
p.descripcion
FROM
Voluntario v
JOIN
Proyecto p ON v.proyecto_asignado = p.nombre_proyecto;

```

Explicación:

- Combina las tablas **Voluntario** y **Proyecto**.
- Muestra los voluntarios junto con el proyecto al que están asignados.

10.6. Vista: Informes Generados

Esta vista muestra los informes generados, incluyendo detalles sobre el tipo de informe, el donante relacionado (si existe) y el proyecto relacionado (si existe).

```
CREATE VIEW vw_informes_generados AS
SELECT
i.id_informe,
i.tipo_informe,
i.fecha_generacion,
d.nombre || ' ' || d.apellidos_razon_social AS nombre_donante,
p.nombre_proyecto,
i.total_ingresos,
i.total_gastos,
i.impacto_proyecto
FROM
Informe i
LEFT JOIN
Donante d ON i.donante_relacionado = d.numero_donante
LEFT JOIN
Proyecto p ON i.proyecto_relacionado = p.nombre_proyecto;
```

Explicación:

- Combina las tablas **Informe**, **Donante** y **Proyecto**.
- Incluye información sobre el tipo de informe, el donante relacionado (opcional) y el proyecto relacionado (opcional).
- Proporciona detalles como ingresos, gastos e impacto del proyecto.

10.7. Vista: Donaciones en Especie por Proyecto

Esta vista muestra las donaciones en especie recibidas por cada proyecto.

```
CREATE VIEW vw_donaciones_especie_por_proyecto AS
SELECT
p.nombre_proyecto,
p.descripcion,
```

```

COUNT(ds.numero_donacion) AS cantidad_donaciones,
SUM(ds.cantidad * ds.valor_estimado) AS valor_total_estimado
FROM
Proyecto p
LEFT JOIN
DonacionEspecie ds ON p.nombre_proyecto = ds.proyecto_destino
GROUP BY
p.nombre_proyecto, p.descripcion;

```

Explicación :

- Combina las tablas **Proyecto** y **DonacionEspecie**.
- Calcula la cantidad de donaciones en especie (**COUNT(ds.numero_donacion)**) y el valor total estimado (**SUM(ds.cantidad * ds.valor_estimado)**).
- Agrupa los resultados por proyecto.

11. INDICES SQL

11.1. Índice: Búsqueda por Correo Electrónico (Donante)

Este índice mejora la búsqueda de donantes por su correo electrónico, una operación común al registrar o consultar donantes.

```
CREATE INDEX idx_donante_correo_electronico ON Donante(correo_electronico);
```

Explicación :

- Acelera las consultas que buscan donantes por **correo_electronico**.
- Es útil para garantizar la unicidad del correo y evitar duplicados.

11.2. Índice: Fecha de Donación Económica

Este índice optimiza las consultas que filtran donaciones económicas por fecha, como informes mensuales o anuales.

```
CREATE INDEX idx_donacioneconomica_fecha_donacion ON
DonacionEconomica(fecha_donacion);
```

Explicación :

- Mejora el rendimiento de consultas que agrupan o filtran donaciones económicas por **fecha_donacion**.

11.3. Índice: Estado de Donación Económica

Este índice acelera las consultas que filtran donaciones económicas por su estado (**recibida, confirmada, procesada**).

```
CREATE INDEX idx_donacioneconomica_estado_donacion ON  
DonacionEconomica(estado_donacion);
```

Explicación :

- Facilita la generación de informes sobre el estado actual de las donaciones.

11.4. Índice: Proyecto Destino (Donaciones)

Este índice optimiza las consultas que vinculan donaciones económicas o en especie con proyectos específicos.

```
CREATE INDEX idx_donacion_proyecto_destino ON  
DonacionEconomica(proyecto_destino);
```

```
CREATE INDEX idx_donacionespecie_proyecto_destino ON  
DonacionEspecie(proyecto_destino);
```

Explicación :

- Mejora el rendimiento de consultas que agrupan donaciones por proyecto.
- Es útil para generar informes de ingresos y gastos por proyecto.

11.5. Índice: Fecha de Gasto (Uso de Fondos)

Este índice optimiza las consultas que filtran gastos por fecha, como informes financieros mensuales.

Explicación :

- Acelera las consultas que analizan los gastos realizados en un período específico.

11.6. Índice: Estado de Comunicación

Este índice mejora las consultas que filtran comunicaciones por su estado (**enviada, pendiente**).

```
CREATE INDEX idx_comunicacion_estado ON Comunicacion(estado);
```

Explicación :

- Facilita la identificación de comunicaciones pendientes o enviadas.

11.7. Índice: Fecha de Generación de Informes

Este índice optimiza las consultas que filtran informes por su fecha de generación.

```
CREATE INDEX idx_informe_fecha_generacion ON Informe(fecha_generacion);
```

Explicación :

- Mejora el rendimiento de consultas que generan informes históricos o recientes.

11.8. Índice: Tipo de Informe

Este índice acelera las consultas que filtran informes por su tipo (**donaciones_por_donante**, **ingresos_y_gastos**, etc.).

```
CREATE INDEX idx_informe_tipo_informe ON Informe(tipo_informe);
```

Explicación :

- Facilita la generación de informes específicos según su tipo.

11.9. Índice: Nombre del Proyecto

Este índice mejora las consultas que buscan proyectos por su nombre.

```
CREATE INDEX idx_proyecto_nombre_proyecto ON Proyecto(nombre_proyecto);
```

Explicación :

- Acelera las búsquedas y filtrados de proyectos por su nombre único.

11.10. Índice: Importe Gastado (Uso de Fondos)

Este índice optimiza las consultas que ordenan o filtran gastos por importe gastado.

```
CREATE INDEX idx_usofondos_importe_gastado ON UsoFondos(importe_gastado);
```

Explicación :

- Mejora el rendimiento de consultas que analizan los mayores o menores gastos.

11.11. Índice: Valor Estimado de Donaciones en Especie

Este índice acelera las consultas que filtran o agrupan donaciones en especie por su valor estimado.

```
CREATE INDEX idx_donacionespecie_valor_estimado ON DonacionEspecie(valor_estimado);
```

Explicación:

- Facilita el análisis de donaciones en especie según su valor.

2: Inserción, Consulta y Eliminación de Datos

2.1: Inserción de datos:

```
INSERT INTO Donante (nombre, apellidos_razon_social, direccion, telefono, correo_electronico, tipo_donante, historial_donaciones, preferencias_comunicacion)
```

```
VALUES
```

```
('Juan', 'Pérez López', 'Calle 123, caracas', '0412-1238080', 'juan.perez@gmail.com', 'individual', 'Frecuente', 'Correo electrónico'),
```

```
('María', 'García Fernández', 'Avenida 456, Pueblo', '0416-7990000', 'maria.garcia@gmail.com', 'individual', 'Esporádico', 'Teléfono'),
```

```
('Empresa Solidaria', 'S.A.', 'Oficina 789, caracas', '0212-9001500', 'contacto@solidaria.com.ve', 'empresa', 'Anual', 'Correo electrónico'),
```

```
('Ana', 'Martínez Torres', 'Calle 321, caracas', '0424-1230000', 'ana.martinez@gmail.com', 'individual', 'Primera vez', 'Correo electrónico');
```

```
INSERT INTO Proyecto (nombre_proyecto, descripcion, ubicacion, fecha_inicio, fecha_fin, presupuesto, objetivos, beneficiarios)
```

```
VALUES
```

```
('Educación para Todos', 'Proyecto educativo en zonas rurales', 'Región Norte', '2025-01-01', '2025-12-31', 500000.00, 'Mejorar acceso a educación', 'Niños y jóvenes'),
```

```
('Agua Limpia', 'Proyecto de acceso a agua potable', 'Región Sur', '2025-03-01', '2025-06-30', 7500000.00, 'Instalar sistemas de agua', 'Comunidades rurales'),
```

```
('Vivienda Digna', 'Construcción de viviendas', 'Región Central', '2025-05-01', '2025-11-30', 10000000.00, 'Proporcionar vivienda segura', 'Familias vulnerables');
```

```
INSERT INTO DonacionEconomica (fecha_donacion, importe, metodo_pago, estado_donacion, proyecto_destino, numero_donante)
```

```
VALUES
```

```
('2025-01-15', 5000.00, 'tarjeta_credito', 'confirmada', 'Educación para Todos', 2),
```

```
('2025-02-20', 3000.00, 'transferencia_bancaria', 'procesada', 'Agua Limpia', 3),
```

```
('2025-03-10', 10000.00, 'pago_movil', 'recibida', 'Vivienda Digna', 4),
```

```
('2025-10-05', 2000.00, 'tarjeta_debito', 'confirmada', 'Educación para Todos', 5);
```

```
INSERT INTO DonacionEspecie (fecha_donacion, descripcion_articulos, cantidad, unidad_medida, valor_estimado, estado_donacion, proyecto_destino, numero_donante)
```

```
VALUES
```

('2025-01-20', 'Libros escolares', 50, 'unidades', 250.00, 'clasificada', 'Educación para Todos', 2),

('2025-02-25', 'Ropa de invierno', 100, 'kilogramos', 500.00, 'distribuida', 'Agua Limpia', 3),

('2025-03-15', 'Materiales de construcción', 200, 'metros', 1000.00, 'recibida', 'Vivienda Digna', 4),

('2025-10-10', 'Juguetes educativos', 30, 'unidades', 150.00, 'clasificada', 'Educación para Todos', 5);

INSERT INTO UsoFondos (proyecto, partida_presupuestaria, importe_asignado, importe_gastado, fecha_gasto, descripcion_gasto)

VALUES

('Educación para Todos', 'Material didáctico', 100000.00, 80000.00, '2025-02-01', 'Compra de libros y cuadernos'),

('Agua Limpia', 'Infraestructura', 500000.00, 450000.00, '2025-03-15', 'Instalación de tuberías'),

('Vivienda Digna', 'Construcción', 750000.00, 700000.00, '2025-04-10', 'Compra de materiales'),

('Educación para Todos', 'Capacitación docente', 500000.00, 400000.00, '2025-09-20', 'Talleres para profesores');

INSERT INTO Comunicacion (numero_donante, fecha_envio, tipo_comunicacion, contenido, estado)

VALUES

(2, '2025-01-20', 'correo_electronico', 'Gracias por su donación al proyecto Educación para Todos.', 'enviada'),

(3, '2025-02-25', 'carta', 'Informe sobre el impacto de su donación.', 'pendiente'),

(4, '2025-03-10', 'correo_electronico', 'Actualización del proyecto Agua Limpia.', 'enviada'),

(5, '2025-10-05', 'correo_electronico', 'Agradecimiento por su apoyo continuo.', 'enviada');

INSERT INTO Voluntario (nombre, apellidos, telefono, correo_electronico, proyecto_asignado)

VALUES

('Carlos', 'Ramírez Pérez', '0414-0021578', 'carlos.ramirez@outlook.com', 'Educación para Todos'),

('Laura', 'Torres Gómez', '0424-2345678', 'laura.torres@outlook.com', 'Agua Limpia'),

('Pedro', 'Hernández López', '0426-2587410', 'pedro.hernandez@outlook.com', 'Vivienda Digna'),

('Sofía', 'Martínez Ruiz', '0412-5697015', 'sofia.martinez@outlook.com', 'Educación para Todos');

INSERT INTO Informe (tipo_informe, fecha_generacion, descripcion, donante_relacionado, proyecto_relacionado, total_ingresos, total_gastos, impacto_proyecto, observaciones_auditoria)

VALUES

('donaciones_por_donante', '2025-01-31', 'Resumen de donaciones realizadas por Juan Pérez.', 2, 'Educación para Todos', 50000.00, 45000.00, 'Alcanzamos a 100 niños.', NULL),

('ingresos_y_gastos', '2025-02-28', 'Informe financiero del proyecto Agua Limpia.', 3, 'Agua Limpia', 750000.00, 700000.00, 'Beneficiamos a 500 personas.', NULL),

('impacto_proyectos', '2025-03-31', 'Impacto del proyecto Vivienda Digna.', 4, 'Vivienda Digna', 1000000.00, 950000.00, 'Construidas 20 viviendas.', NULL),

('auditoria_cumplimiento', '2025-10-31', 'Auditoría de cumplimiento normativo.', NULL, NULL, NULL, NULL, NULL, 'Cumple con todas las regulaciones.');

2.2: Consultas básicas:

— Consultar todos los donantes

SELECT

numero_donante,

nombre,

apellidos_razon_social,

correo_electronico,

tipo_donante

FROM

Donante;

RESULTADO

Data Output

Mensajes

Notificaciones

-- Consultar todos los proyectos

```
SELECT
nombre_proyecto,
descripcion,
ubicacion,
fecha_inicio,
presupuesto
FROM
Proyecto;

RESULTADO
```

Data Output

Mensajes

Notificaciones

SQL

Showing rows: 1 to 3

	nombre_proyecto [PK] character varying (200)	descripcion text	ubicacion character varying (255)	fecha_inicio date	presupuesto numeric (15,2)
1	Educación para Todos	Proyecto educativo en zonas rurales	Región Norte	2025-01-01	500000.00
2	Agua Limpia	Proyecto de acceso a agua potable	Región Sur	2025-03-01	7500000.00
3	Vivienda Digna	Construcción de viviendas	Región Central	2025-05-01	10000000.00

-- Consultar todos los gastos realizados en los proyectos

```
SELECT
uf.proyecto,
uf.partida_presupuestaria,
uf.importe_asignado,
uf.importe_gastado,
uf.fecha_gasto,
```

uf.descripcion_gasto

FROM

UsoFondos uf;

RESULTADO:

	proyecto character varying (200)	partida_presupuestaria character varying (200)	importe_asignado numeric (15,2)	importe_gastado numeric (15,2)	fecha_gasto date	descripcion_gasto text
1	Educación para Todos	Material didáctico	100000.00	80000.00	2025-02-01	Compra de libros y cuadernos
2	Agua Limpia	Infraestructura	500000.00	450000.00	2025-03-15	Instalación de tuberías
3	Vivienda Digna	Construcción	750000.00	700000.00	2025-04-10	Compra de materiales
4	Educación para Todos	Capacitación docente	500000.00	400000.00	2025-09-20	Talleres para profesores

-- Consultar todos los voluntarios asignados a proyectos

SELECT

v.id_voluntario,

v.nombre || ' ' || v.apellidos AS nombre_voluntario,

v.correo_electronico,

v.proyecto_asignado

FROM

Voluntario v;

RESULTADO:

	id_voluntario [PK] integer	nombre_voluntario text	correo_electronico character varying (100)	proyecto_asignado character varying (200)
1	2	Carlos Ramírez Pérez	carlos.ramirez@outlook.com	Educación para Todos
2	3	Laura Torres Gómez	laura.torres@outlook.com	Agua Limpia
3	4	Pedro Hernández López	pedro.hernandez@outlook.com	Vivienda Digna
4	5	Sofía Martínez Ruiz	sofia.martinez@outlook.com	Educación para Todos

2.3: Consultas Avanzadas:

-- Consultar todas las donaciones económicas con detalles del donante

```

SELECT
d.numero_donacion,
d.fecha_donacion,
d.importe,
d.metodo_pago,
dn.nombre || ' ' || dn.apellidos_razon_social AS nombre_donante,
d.proyecto_destino
FROM
DonacionEconomica d
JOIN
Donante dn ON d.numero_donante = dn.numero_donante;

```

RESULTADO

Data Output Mensajes Notificaciones

Showing rows: 1 to 4

	numero_donacion [PK] integer	fecha_donacion date	importe numeric (15,2)	metodo_pago character varying (30)	nombre_donante text	proyecto_destino character varying (200)
1	2	2025-01-15	5000.00	tarjeta_credito	Juan Pérez López	Educación para Todos
2	3	2025-02-20	3000.00	transferencia_bancaria	María García Fernánd...	Agua Limpia
3	4	2025-03-10	10000.00	pago_movil	Empresa Solidaria S.A.	Vivienda Digna
4	5	2025-10-05	2000.00	tarjeta_debito	Ana Martínez Torres	Educación para Todos

-- Consultar todas las donaciones en especie con detalles del donante

```

SELECT
ds.numero_donacion,
ds.fecha_donacion,
ds.descripcion_articulos,
ds.cantidad,
ds.valor_estimado,
dn.nombre || ' ' || dn.apellidos_razon_social AS nombre_donante,
ds.proyecto_destino
FROM
DonacionEspecie ds

```


JOIN

Donante dn ON ds.numero_donante = dn.numero_donante;

RESULTADO:

Data Output

Mensajes

Notificaciones

<

-- Consultar todas las comunicaciones enviadas a los donantes

SELECT

c.id_comunicacion,

dn.nombre || ' ' || dn.apellidos_razon_social AS nombre_donante,

c.fecha_envio,

c.tipo_comunicacion,

c.estado

FROM

Comunicacion c

JOIN

Donante dn ON c.numero_donante = dn.numero_donante;

RESULTADO:

Data Output

Mensajes

Notificaciones

-- Consultar todos los informes generados

SELECT

```

i.id_informe,
i.tipo_informe,
i.fecha_generacion,
i.descripcion,
dn.nombre || ' ' || dn.apellidos_razon_social AS nombre_donante,
i.proyecto_relacionado,
i.total_ingresos,
i.total_gastos
FROM
Informe i
LEFT JOIN
Donante dn ON i.donante_relacionado = dn.numero_donante;

```

RESULTADO:

Data Output Mensajes Notificaciones								
	id_informe [PK] integer	tipo_informe character varying (50)	fecha_generacion date	descripcion text	nombre_donante text	proyecto_relacionado character varying (200)	total_ingresos numeric (15,2)	total_gastos numeric (15,2)
1	2	donaciones_por_dona...	2025-01-31	Resumen de donaciones realizadas por Juan Pérez.	Juan Pérez López	Educación para Todos	50000.00	45000.00
2	3	ingresos_y_gastos	2025-02-28	Informe financiero del proyecto Agua Limpia.	María García Fernánd...	Agua Limpia	750000.00	700000.00
3	4	impacto_proyectos	2025-03-31	Impacto del proyecto Vivienda Digna.	Empresa Solidaria S.A.	Vivienda Digna	1000000.00	950000.00
4	5	auditoria_cumplimiento	2025-10-31	Auditoría de cumplimiento normativo.	[null]	[null]	[null]	[null]

2.4: Eliminación de datos:

-- consulta antes de eliminar al donante

```
SELECT * FROM donacioneconomica;
```

Data Output Mensajes Notificaciones							
	numero_donacion [PK] integer	fecha_donacion date	importe numeric (15,2)	metodo_pago character varying (30)	estado_donacion character varying (20)	proyecto_destino character varying (200)	numero_donante integer
1	2	2025-01-15	5000.00	tarjeta_credito	confirmada	Educación para Todos	2
2	3	2025-02-20	3000.00	transferencia_bancaria	procesada	Agua Limpia	3
3	4	2025-03-10	10000.00	pago_movil	recibida	Vivienda Digna	4
4	5	2025-10-05	2000.00	tarjeta_debito	confirmada	Educación para Todos	5

- Eliminar todas las donaciones económicas de un proyecto

```
DELETE FROM DonacionEconomica
```

```
WHERE proyecto_destino = 'Agua Limpia';
```

RESULTADO:

Data Output Mensajes Notificaciones							
Showing rows: 1 to 3							
	numero_donacion [PK] integer	fecha_donacion date	importe numeric (15,2)	metodo_pago character varying (30)	estado_donacion character varying (20)	proyecto_destino character varying (200)	numero_donante integer
1	2	2025-01-15	5000.00	tarjeta_credito	confirmada	Educación para Todos	2
2	4	2025-03-10	10000.00	pago_movil	recibida	Vivienda Digna	4
3	5	2025-10-05	2000.00	tarjeta_debito	confirmada	Educación para Todos	5

-- Eliminar comunicacion

SELECT * FROM

comunicacion;

Data Output Mensajes Notificaciones						
Showing rows: 1 to 4						
	id_comunicacion [PK] integer	numero_donante integer	fecha_envio date	tipo_comunicacion character varying (20)	contenido text	estado character varying (20)
1	10	2	2025-01-20	correo_electronico	Gracias por su donación al proyecto Educación para Todo...	enviada
2	11	3	2025-02-25	carta	Informe sobre el impacto de su donación.	pendiente
3	12	4	2025-03-10	correo_electronico	Actualización del proyecto Agua Limpia.	enviada
4	13	5	2025-10-05	correo_electronico	Agradecimiento por su apoyo continuo.	enviada

-- Eliminar todas las comunicaciones pendientes

DELETE FROM Comunicacion

WHERE estado = 'pendiente';

RESULTADO:

Data Output Mensajes Notificaciones						
Showing rows: 1 to 3						
	id_comunicacion [PK] integer	numero_donante integer	fecha_envio date	tipo_comunicacion character varying (20)	contenido text	estado character varying (20)
	10	2	2025-01-20	correo_electronico	Gracias por su donación al proyecto Educación para Todo...	enviada
	12	4	2025-03-10	correo_electronico	Actualización del proyecto Agua Limpia.	enviada
	13	5	2025-10-05	correo_electronico	Agradecimiento por su apoyo continuo.	enviada

2.5: Actualización de datos:

-- consulta antes de actualizar correo

SELECT * FROM

donante;

Data Output Mensajes Notificaciones									
	numero_donante [PK] integer	nombre character varying (100)	apellidos_razon_social character varying (200)	direccion character varying (255)	telefono character varying (20)	correo_electronico character varying (100)	tipo_donante character varying (20)	historial_donaciones text	preferencias_comunicacion text
1	2	Juan	Pérez López	Calle 123, caracas	0412-1238080	juan.perez@gmail.com	individual	Frecuente	Correo electrónico
2	3	María	García Fernández	Avenida 456, Pueblo	0416-7990000	maria.garcia@gmail.com	individual	Esporádico	Teléfono
3	4	Empresa Solidaria	S.A.	Oficina 789, caracas	0212-9001500	contacto@solidaria.com.ve	empresa	Anual	Correo electrónico
4	5	Ana	Martínez Torres	Calle 321, caracas	0424-1230000	ana.martinez@gmail.com	individual	Primera vez	Correo electrónico

-- Actualizar el correo electrónico de un donante

UPDATE Donante

SET correo_electronico = 'fernandezmaria@gmail.com'

WHERE numero_donante = 3;

-- nueva consulta despues de actualizar correo

SELECT * FROM

donante;

RESULTADO:

Data Output Mensajes Notificaciones									
	numero_donante [PK] integer	nombre character varying (100)	apellidos_razon_social character varying (200)	direccion character varying (255)	telefono character varying (20)	correo_electronico character varying (100)	tipo_donante character varying (20)	historial_donaciones text	preferencias text
1	2	Juan	Pérez López	Calle 123, caracas	0412-1238080	juan.perez@gmail.com	individual	Frecuente	Correo elec
2	4	Empresa Solidaria	S.A.	Oficina 789, caracas	0212-9001500	contacto@solidaria.com.ve	empresa	Anual	Correo elec
3	5	Ana	Martínez Torres	Calle 321, caracas	0424-1230000	ana.martinez@gmail.com	individual	Primera vez	Correo elec
4	3	María	García Fernández	Avenida 456, Pueblo	0416-7990000	fernandezmaria@gmail.co	individual	Esporádico	Teléfono

-- consulta antes de cambiar estado de una donacion economica

SELECT * FROM DonacionEconomica

Data Output Mensajes Notificaciones							
	numero_donacion [PK] integer	fecha_donacion date	importe numeric (15,2)	metodo_pago character varying (30)	estado_donacion character varying (20)	proyecto_destino character varying (200)	numero_donante integer
1	2	2025-01-15	5000.00	tarjeta_credito	confirmada	Educación para Todos	2
2	4	2025-03-10	10000.00	pago_movil	recibida	Vivienda Digna	4
3	5	2025-10-05	2000.00	tarjeta_debito	confirmada	Educación para Todos	5

-- Cambiar el estado de una donación económica

UPDATE DonacionEconomica

SET estado_donacion = 'procesada'

WHERE numero_donacion = 5;

-- nueva consulta despues de cambiar estado de una donacion economica

```
SELECT * FROM DonacionEconomica;
```

RESULTADO:

	numero_donacion [PK] integer	fecha_donacion date	importe numeric (15,2)	metodo_pago character varying (30)	estado_donacion character varying (20)	proyecto_destino character varying (200)	numero_donante integer
1	2	2025-01-15	5000.00	tarjeta_credito	confirmada	Educación para Todos	2
2	4	2025-03-10	10000.00	pago_movil	recibida	Vivienda Digna	4
3	5	2025-10-05	2000.00	tarjeta_debito	procesada	Educación para Todos	5

3: Diseño Lógico y Físico de la Base de Datos

3.1: Diseño Lógico

Refinamiento del Diseño Lógico

El diseño lógico se basa en los resultados de las fases anteriores (entidades, relaciones y atributos). Se refinan las tablas para garantizar escalabilidad y rendimiento a largo plazo.

1. Normalización :

- Asegurar que las tablas estén en **tercera forma normal (3FN)** para evitar redundancias.
- Ejemplo: La tabla **Donante** ya está normalizada porque cada atributo depende directamente de la clave primaria (**numero_donante**).

2. Escalabilidad :

- Dividir tablas grandes si es necesario. Por ejemplo, si la tabla **DonacionEconomica** crece significativamente, se puede separar en dos tablas:
 - **DonacionEconomica** (datos básicos).
 - **DetallesDonacionEconomica** (detalles adicionales como descripciones o metadatos).

3. Relaciones :

- Mantener claves foráneas para garantizar la integridad referencial.
- Ejemplo: La relación entre **Donante** y **DonacionEconomica** se mantiene mediante la columna **numero_donante**.

4. Tipos de Datos:

- Usar tipos de datos apropiados para optimizar el almacenamiento:
 - **SERIAL** para claves primarias autoincrementales.
 - **DECIMAL(15, 2)** para importes monetarios.
 - **VARCHAR(n)** con longitud adecuada para cadenas de texto.

3.2: Diseño físico:

3.2.1. Optimización del Diseño Físico

El diseño físico se enfoca en mejorar el rendimiento mediante índices, particionamiento y configuración de almacenamiento.

1. Elección de Tipos de Almacenamiento:

- Usar **TOAST (The Oversized-Attribute Storage Technique)** para columnas con datos grandes (por ejemplo, **descripcion** en **Proyecto** o **contenido** en **Comunicacion**).

-- Usar ****TOAST****

```
ALTER TABLE Proyecto ALTER COLUMN descripcion SET STORAGE EXTERNAL;
```

3.2.2. Índices:

- Crear índices en columnas frecuentemente consultadas

-- Indices frecuentemente consultados

```
CREATE INDEX idx_donante_correo_electronico ON Donante(correo_electronico);
```

```
CREATE INDEX idx_donacioneconomica_fecha_donacion ON DonacionEconomica(fecha_donacion);
```

3.2.3. Configuración de PostgreSQL:

- Ajustar parámetros en **postgresql.conf** para mejorar el rendimiento:
 - **Memoria compartida** : Aumentar **shared_buffers** al 25% de la RAM del servidor.

```
shared_buffers = '3GB'
```

Conexiones:

Ajustar **max_connections** según la carga esperada.

```
max_connections = 200
```

Vacuum y Autovacuum:

Configurar para mantener la base de datos limpia.

```
autovacuum = on
```

```
autovacuum_vacuum_scale_factor = 0.1
```

```
autovacuum_analyze_scale_factor = 0.05
```

Trabajo de Escritura (work_mem)

- **Descripción:** Define la cantidad de memoria asignada para operaciones como ordenamientos y agregaciones.
- **Recomendación:** Incrementa este valor si realizas muchas consultas complejas.

```
work_mem = '64MB' # Valor predeterminado es 4MB
```

Mantenimiento de Memoria (maintenance_work_mem)

- **Descripción:** Controla la memoria usada para tareas de mantenimiento como **VACUUM**, **CREATE INDEX** y **ALTER TABLE**.
- **Recomendación:** Aumenta este valor para grandes operaciones de mantenimiento

```
maintenance_work_mem = '1GB'
```

Registro de Consultas Lentas (log_min_duration_statement)

- **Descripción :** Registra consultas que tardan más de un tiempo específico.
- **Recomendación :** Útil para identificar consultas lentas.

```
log_min_duration_statement = 200 # Registra consultas que duran más de 200 ms
```

Almacenamiento Temporal (temp_buffers)

- **Descripción:** Define la cantidad de memoria para operaciones temporales.
- **Recomendación :** Incrementa si realizas muchas operaciones temporales.

```
temp_buffers = '32MB'
```

3.3: Tuneado de Consultas

3.3.1 Análisis de Planes de Ejecución

Usar la herramienta **EXPLAIN** para analizar los planes de ejecución de las consultas y realizar ajustes.

1. Ejemplo:

- Analizar una consulta para obtener donaciones económicas de un proyecto específico

EXPLAIN ANALYZE

SELECT * FROM DonacionEconomica WHERE proyecto_destino = 'Educación para Todos';

RESULTADO:

Data Output		Mensajes	Notificaciones
		Showing rows: 1 to 4 Page No:	
	QUERY PLAN text		
1	Index Scan using idx_donacion_proyecto_destino on donacioneconomica (cost=0.14..8.16 rows=1 width=584) (actual time=0.044..0.048 rows=2 loop...		
2	Index Cond: ((proyecto_destino)::text = 'Educación para Todos')::text)		
3	Planning Time: 2.898 ms		
4	Execution Time: 0.071 ms		

3.3.2. Optimización de Consultas:

- Si la consulta es lenta, agregar un índice en la columna **proyecto_destino**:

```
CREATE INDEX idx_donacioneconomica_proyecto_destino ON  
DonacionEconomica(proyecto_destino);
```

3.4: Backup y Recuperación

3.4.1. Backup Lógico con pg_dump

1. Abrir la Terminal:

Abre una terminal (CMD o PowerShell) con permisos de administrador.

1. Ejecutar el Comando pg_dump :

Usa el siguiente comando para respaldar una base de datos específica:

```
"C:\Program Files\PostgreSQL\17\bin\pg_dump.exe" -U postgres -d ong_ayuda_accion  
-f C:\Users\EQUIPO\Documents\BBDD-ONG-backup.sql
```

3.4.2. Programar Backups Automáticos:

Crear Script .bat para el Backup

El primer paso es crear un archivo de script **.bat** que contenga el comando **pg_dump** para realizar el backup.

1. Abrir el Bloc de Notas:

Abre el Bloc de Notas o cualquier editor de texto.

1. Escribir el Comando pg_dump

```
"C:\Program Files\PostgreSQL\17\bin\pg_dump.exe" -U postgres -d ong_ayuda_accion  
-f C:\Users\EQUIPO\Documents\BBDD-ONG-backup_%date:~10,4%-_%date:~4,2%-  
_%date:~7,2%_%time:~0,2%-_%time:~3,2%.sql
```

Guardar el Archivo:

Guarda el archivo con extensión **.bat**.

Probar el Script:

Haz doble clic en el archivo **.bat**

3.4.3. Restaurar desde Backup

Restauración de Backup Lógico (pg_dump)

3.4.3.1. Preparar la Base de Datos

1. Crear una Nueva Base de Datos :

Si la base de datos no existe, créala

-- crear base de datos para hacer un respaldo

```
CREATE DATABASE ong_backup;
```

3.4.3.2. Ejecutar el Comando de Restauración

1. Abrir la Terminal:

Abre una terminal (CMD o PowerShell) con permisos de administrador.

1. Ejecutar el Comando psql :

Usa el siguiente comando para restaurar el backup:

```
psql "C:\Program Files\PostgreSQL\17\bin\pg_dump.exe" -U postgres -d  
ong_ayuda_accion -f C:\Users\EQUIPO\Documents\BBDD-ONG-backup.sql
```

3.4.3.3. Verificar la Restauración:

Una vez completada la restauración, conecta a la base de datos y verifica que los datos estén intactos y ejecuta estas consultas:

Verificar que las tablas existen en la base de datos restaurada:

```
\dt
```

Contar Registros en una Tabla

```
SELECT COUNT(*) FROM Donante;
```

Consultar Datos Específicos

```
SELECT * FROM Donante LIMIT 10;
```