



**UNIVERSIDAD CENTROAMERICANA  
JOSÉ SIMEÓN CAÑAS**

## **ADMINISTRACIÓN DE BASES DE DATOS**

**Sección 02**

### **PROYECTO DE CÁTEDRA: ADMINISTRACIÓN DE BASES DE DATOS CON SQL SERVER**

Estudiantes:

Márquez Monterrosa, Andrea Lissette, 00033924

Martínez Ruiz, Katherine Nicole, 00045724

Menjivar Rosales, Johana Raquel, 00226324

Guillén González, Walter Samuel, 00097524

Vásquez Maravilla, Stephanie Alessandra, 00056024

Docente de la asignatura:

Ing. James Edward Humberstone Morales

Fecha de entrega del informe: 26/11/2025

## PLAN DE ADMINISTRACIÓN DE LA BASE DE DATOS

### Descripción del sistema elegido.

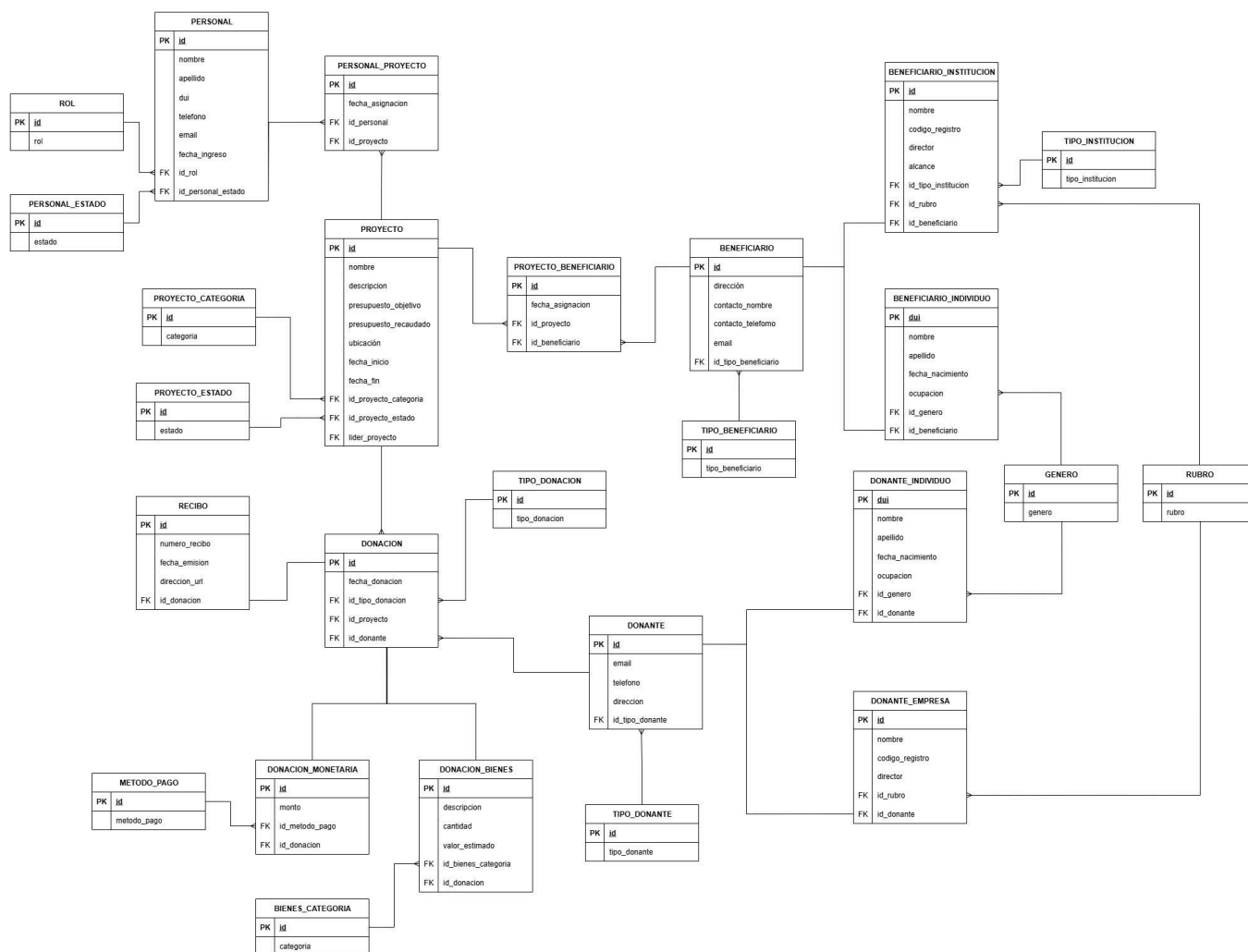
El sistema que se desarrolla en este proyecto es de una Organización No Gubernamental (ONG), estas organizaciones enfrentan desafíos administrativos y de gestión de datos debido a su dependencia de la confianza pública y la necesidad de maximizar el impacto de los proyectos que llevan a cabo con limitados recursos.

Para ello se ha creado el siguiente diseño de una Base de Datos autocontenida que tiene el propósito de proveer una estructura organizada y eficiente que no dependa de la instancia principal del servidor para brindarle mayor portabilidad y que las configuraciones se mantengan consistentes en todos los entornos en los que se utilice. Igualmente se ha hecho uso de los principios de normalización para asegurar la integridad de la Base de Datos.

Otro punto importante son los mecanismos de seguridad que aseguran la calidad de los registros, como el uso correcto de claves primarias y foráneas, al igual que restricciones que obligan el ingreso de toda información necesaria para el correcto funcionamiento del sistema. Adicionalmente, se han implementado índices relevantes que permiten la optimización de velocidad de las consultas agregadas, posibilitando la eficiencia en la recuperación de datos.

Por último, abordando el tema de Inteligencia de Negocios, se hace uso de la herramienta de Power BI para la generación de reportes intuitivos que permiten transformar los datos de la ONG en información estratégica, esto se logra mediante consultas avanzadas que involucran el uso de funciones ventana.

Fig 1. Diagrama relacional diseñado.



## Diccionario de datos.

A partir del diagrama relacional propuesto, a continuación se presenta el desglose de los datos asociados a cada tabla diseñada, detallando su descripción y la lógica de su construcción.

**Tabla 1. Personal.**

Nombre de dato	Tipo de dato	Restricción	Descripción
id	INT	PK, NOT NULL	Identificador único, autonumérico derivado de una secuencia y Clave Primaria de la tabla..
nombre	VARCHAR(50)	NOT NULL	Nombre de la persona.
apellido	VARCHAR(50)	NOT NULL	Apellido de la persona.
dui	VARCHAR(15)	NOT NULL	Documento Único de Identidad de la persona.
telefono	VARCHAR(20)	NOT NULL	Número de contacto de la persona.
email	VARCHAR(100)	NOT NULL	Correo electrónico de la persona
fecha_ingreso	DATE	NOT NULL	Fecha en la que la persona ingresó a la ONG, ya sea como empleado o voluntario.
id_rol	INT	FK, NOT NULL	Clave foránea que define si la persona es empleado o voluntario.
id_personal_estado	INT	FK, NOT NULL	Clave foránea que indica el estado actual del personal.
codigo_persona1	Columna calculada (resultado VARCHAR)	PERSISTED	Código de identificación único del personal, derivado del id e id_rol.

**Tabla 2. Proyecto.**

Nombre de dato	Tipo de dato	Restricción	Descripción
id	INT	PK, NOT NULL	Identificador único, autonumérico derivado de una secuencia y Clave Primaria de la tabla.
nombre	VARCHAR(75)	NOT NULL	Nombre oficial o título del proyecto de la ONG.
descripcion	TEXT	NOT NULL	Descripción breve de las metas y metodologías del proyecto.
presupuesto_objetivo	DECIMAL(10,2)	NOT NULL	Monto total de fondos requeridos para la ejecución del proyecto.
presupuesto_recaudado	DECIMAL(10,2)	NOT NULL	Monto total de fondos que se han logrado recaudar hasta el momento.
ubicacion	TEXT	NOT NULL	Lugar donde se desarrolla el proyecto.
fecha_inicio	DATE	NOT NULL	Fecha en la que el proyecto fue propuesto a la ONG.
fecha_fin	DATE	NOT NULL	Fecha prevista o real de la finalización del proyecto.

id_proyecto_categoria	INT	FK, NOT NULL	Clave foránea que indica la categoría a la que pertenece el proyecto.
id_proyecto_estado	INT	FK, NOT NULL	Clave foránea que define el estado actual del proyecto.
lider_proyecto	INT	FK, NOT NULL	Clave foránea que define el personal responsable del proyecto.
codigo_proyecto	Columna calculada (resultado VARCHAR)	PERSISTED	Código de identificación único de los proyectos, derivado del id.

**Tabla 3. Beneficiario.**

Nombre de dato	Tipo de dato	Restricción	Descripción
id	INT	PK, NOT NULL	Identificador único, autonumérico derivado de una secuencia y Clave Primaria de la tabla.
dirección	TEXT	NOT NULL	Dirección física del beneficiario o la comunidad a la que pertenece.
contacto_nombre	VARCHAR(50)	NOT NULL	Nombre de la persona de contacto, puede ser el propio beneficiario o un representante.
contacto_telefono	VARCHAR(20)	NOT NULL	Número de teléfono de contacto del beneficiario o representante.
email	VARCHAR(100)	NOT NULL	Correo electrónico de contacto del beneficiario o representante.
id_tipo_beneficiario	INT	FK, NOT NULL	Clave foránea que define si el beneficiario es individuo o institución.
codigo_beneficiario	Columna calculada (resultado VARCHAR)	PERSISTED	Código de identificación único de los beneficiarios, derivado del id e id_tipo_beneficiario.

**Tabla 4. Donante.**

Nombre de dato	Tipo de dato	Restricción	Descripción
id	INT	PK, NOT NULL	Identificador único, autonumérico derivado de una secuencia y Clave Primaria de la tabla.
email	VARCHAR(100)	NOT NULL	Correo electrónico de contacto del donante.
telefono	VARCHAR(20)	NOT NULL	Número de teléfono de contacto del donante.
direccion	TEXT	NOT NULL	Dirección física del donante.
id_tipo_donante	INT	FK, NOT NULL	Clave foránea que define si el donante es individuo o empresa.

codigo_donante	Columna calculada (resultado VARCHAR)	PERSISTED	Código de identificación único de los donantes, derivado del id e id_tipo_donante.
----------------	---------------------------------------	-----------	--

**Tabla 5. Donación.**

Nombre de dato	Tipo de dato	Restricción	Descripción
id	INT	PK, NOT NULL	Identificador unico, autonumerico derivado de una secuencia y Clave Primaria de la tabla
fecha_donacion	DATE	NOT NULL	Fecha en que se realizo la donacion
id_tipo_donacion	INT	FK, NOT NULL	Clave foranea que define si la donacion es monetaria o bienes
id_proyecto	INT	FK, NOT NULL	Clave foranea que indica a cual proyecto esta asociada la donacion
id_donante	INT	FK, NOT NULL	Clave foranea que indica el donante que realizo la donacion
codigo_donacion	Columna calculada (resultado VARCHAR)	PERSISTED	Código de identificación único de las donacion, derivado del id.e id_tipo_donacion

**Tabla 6. Donacion\_Monetaria.**

Nombre de dato	Tipo de dato	Restricción	Descripción
id	INT	PK, INT	Identificador único y que se genera automáticamente mediante un valor incremental
monto	DECIMAL	NOT NULL	El valor del monto de la donación monetaria
id_metodo_pago	INT	NOT NULL	Clave foránea que define qué método de pago se utilizo en la donación
id_donacion	INT	NOT NULL	Clave foránea que indica la donación a la que está asociada la donación monetaria

**Tabla 7. Donacion\_Bienes**

Nombre de dato	Tipo de dato	Restricción	Descripción
id	INT	PK, INT	Identificador único y que se genera automáticamente mediante un valor incremental
descripcion	TEXT	NOT NULL	Descripción física del bien que se está donando
cantidad	INT	NOT NULL	La cantidad de ítems que se están donando
valor_estimado	DECIMAL	NOT NULL	El valor que se estima que tienen los bienes

id_bienes_categoria	INT	NOT NULL	Clave foránea que define a qué categoría pertenece el bien donado
id_donacion	INT	NOT NULL	Clave foránea que indica la donación a la que está asociada la donacion_bienes

**Tabla 8. Donante\_Individuo**

Nombre de dato	Tipo de dato	Restricción	Descripción
dui	INT	PK, INT	Documento Único de Identidad (DUI) que se utiliza como identificador único de cada registro
nombre	TEXT	NOT NULL	Nombres que tiene la persona
apellido	INT	NOT NULL	Apellidos que tiene la persona
fecha_nacimiento	DECIMAL	NOT NULL	Fecha en la que nació la persona
ocupación	INT	NOT NULL	Ocupación que tiene en ese momento la persona que dona
id_genero	INT	NOT NULL	Clave foránea que define a qué género pertenece la persona
id_donante	INT	NOT NULL	Clave foránea que indica el donante al que está relacionado donante_individuo

**Tabla 9. Donante\_Empresa**

Nombre de dato	Tipo de dato	Restricción	Descripción
id	INT	PK, INT	Identificador único y que se genera automáticamente mediante un valor incremental
nombre	TEXT	NOT NULL	Nombre por el que se identifica a la empresa
codigo_registro	INT	NOT NULL	Código de registro que posee la empresa
director	DECIMAL	NOT NULL	Director actual de la empresa
id_rubro	INT	NOT NULL	Clave foránea que define a qué rubro pertenece la empresa
id_donante	INT	NOT NULL	Clave foránea que indica el donante al que está relacionado donante_empresa

**Tabla 10. Recibo.**

Nombre de dato	Tipo de dato	Restricción	Descripción
id	INT	PK, NOT NULL	Identificador único, autonumérico derivado de una secuencia y Clave Primaria de la tabla
numero_recibo	VARCHAR(100)	NOT NULL	Código de identificación único de los recibos, derivado del id.
fecha_emision	DATE	NOT NULL	Fecha en la que se emite el recibo
direccion_url	TEXT	NOT NULL	Dirección url donde se encuentre el recibo

id_donacion	INT	FK, NOT NULL	Clave foránea que indica a cuál donación está asociado el recibo
-------------	-----	--------------	--

**Tabla 11. Proyecto\_Beneficiario.**

Nombre de dato	Tipo de dato	Restricción	Descripción
id	INT	PK, NOT NULL	Identificador único y que se genera automáticamente mediante un valor incremental
fecha_asignacion	DATE	NOT NULL	Fecha en la que se asignó el beneficiario al proyecto
id_proyecto	INT	NOT NULL	Clave foránea que indica el proyecto involucrado
id_beneficiario	INT	NOT NULL	Clave foránea que indica el beneficiario involucrado

**Tabla 12. Personal\_Proyecto.**

Nombre de dato	Tipo de dato	Restricción	Descripción
id	INT	PK, NOT NULL	Identificador único, autonumérico derivado de una secuencia y Clave Primaria de la tabla
fecha_asignacion	VARCHAR(100)	NOT NULL	Fecha en la que se asignó el personal al proyecto.
id_personal	INT	FK, NOT NULL	Clave foránea que indica el personal involucrado
id_proyecto	INT	FK, NOT NULL	Clave foránea que indica el proyecto involucrado

**Tabla 13. Beneficiario\_Individuo**

Nombre de dato	Tipo de dato	Restricción	Descripción
dui	INT	PK, NOT NULL	Documento Único de Identidad (DUI) que se utiliza como identificador único de cada registro
nombre	VARCHAR(50)	NOT NULL	Nombres que tiene la persona
apellido	VARCHAR(50)	NOT NULL	Apellidos que tiene la persona
fecha_nacimiento	DATE	NOT NULL	Fecha en la que nació la persona
ocupacion	VARCHAR(30)	NOT NULL	Ocupación que tiene en ese momento la persona Beneficiaria
id_genero	INT	NOT NULL	Clave foránea que define a qué género pertenece la persona
id_beneficiario	INT	NOT NULL	Clave foránea que indica el beneficiario al que está relacionado beneficiario_individuo

**Tabla 13. Beneficiario\_Institucion**

Nombre de dato	Tipo de dato	Restricción	Descripción
id	INT	PK, NOT NULL	Identificador único y que se genera automáticamente mediante un valor incremental
nombre	VARCHAR(75)	NOT NULL	Nombre por el que se identifica a la empresa
codigo_registro	VARCHAR(15)	NOT NULL	Código de registro que posee la empresa
director	DATE	NOT NULL	Director actual de la empresa
Alcance	VARCHAR(30)	NOT NULL	Alcance de personas que pueden ser ayudadas
id_tipo_institucion	INT	FK, NOT NULL	Clave foránea que define que tipo de institución es el beneficiario
id_rubro	INT	FK, NOT NULL	Clave foránea que define a qué rubro pertenece la empresa
id_beneficiario	INT	FK, NOT NULL	Clave foránea que indica el beneficiario al que está relacionado beneficiario_institucion

#### Políticas de seguridad implementadas.

Para garantizar la confidencialidad y orden de la información, así como la seguridad de la Base de Datos, se han definido e implementado una serie de políticas que cumplen con los estándares de seguridad y el control de mínimo privilegio.

- **Esquemas.**

La Base de Datos se ha dividido en cinco esquemas, los cuales se desglosan a continuación:

**Negocio.** Contiene las entidades principales de la organización que poseen la información más detallada y crítica, además contiene las relaciones de muchos a muchos que vincula a las entidades principales.

**Catálogo.** Almacena las tablas de referencia que contienen valores predefinidos que sirven para clasificar datos en otros esquemas.

**Transacciones.** Almacena los registros de eventos y movimientos diarios de donaciones de la organización.

**Análisis.** Almacena vistas relevantes para ser utilizadas en reportes físicos o en la generación de reportes dinámicos utilizando Power BI.

**Administrativo.** Tiene el fin de almacenar vistas que incluyen procesos relevantes para el DBA por el mantenimiento y funcionamiento correcto de la Base de Datos.

- **Roles.**



Para la implementación de Mínimo Privilegio se han generado una serie de roles que permiten una estructura ordenada de trabajo, asegurando que cada usuario tiene únicamente los permisos indispensables para ejecutar sus tareas.

Tabla . Descripción de los roles creados.

ROL	DESCRIPCIÓN GENERAL	PERMISOS	RESTRICCIONES
db_admin	Representa el nivel más alto de control dentro de la Base de Datos diseñado para el encargado de DBA de la ONG.  Sus responsabilidades abarcan el control total del sistema, gestión de seguridad, mantenimiento y recuperación.	Se le otorga el control completo de la Base de Datos, único rol que no está limitado por Principio de Mínimo Privilegio.	-
db_rrhh	Creado para el sector de Recursos Humanos, su principal responsabilidad es garantizar la integridad y actualización de los empleados y voluntarios, cumpliendo con la confidencialidad de la información personal.	Se le permite la lectura de las tablas Rol y Personal_Estado pertenecientes al esquema Catálogo.  Tiene permisos de lectura, inserción y modificación sobre la tabla Personal perteneciente al esquema Negocio.	Tiene prohibición total en el esquema Transacciones.  Se le niega el acceso al resto del esquema Catálogo.  Se le niega el acceso al resto del esquema Negocio.
db_coordinador_proyectos	Diseñado para los encargados de ejecutar y supervisar las iniciativas de la ONG, sus permisos están optimizados para la gestión de proyectos, beneficiarios y el personal involucrado.	Se le permite la lectura de todo el esquema Catálogo.  Se le otorga la lectura, inserción y modificación sobre todo el esquema Negocio.	Tiene prohibición total en el esquema Transacciones.  Se le restringe el acceso a la clasificación de pagos, bienes y donaciones del esquema Catálogo.  Se le niega la capacidad de modificar y crear registros en Personal, igualmente no puede leer, insertar, modificar o eliminar ningún dato relacionado a Donación, todo ello perteneciente al esquema Negocio.
db_gestor_finanzas	Es el responsable de la integridad y trazabilidad de todos los registros financieros de la ONG, es decir las donaciones monetarias, donantes y recibos.  Está diseñado para darle la autonomía necesaria para manejar únicamente las transacciones monetarias.	Tiene permiso de lectura sobre las tablas de referencia financieras del esquema Catálogo.  Se le permite la lectura, inserción y modificación de las tablas relacionadas a las donaciones del esquema Negocio.  Se le permite la lectura, inserción y modificación de las tablas relacionadas a las transacciones monetarias del esquema Transacciones.	Se le prohíbe el acceso al resto del esquema Catálogo.  Se le niega la capacidad de alterar y eliminar en el resto de tablas dentro del esquema Negocio.  Se le restringe el acceso a Donacion_Bienes y alterar o eliminar en el resto de tablas del esquema Transacciones.
db_gestor_inventario	Es el encargado de la entrada y control físico de los bienes donados a la ONG.  Está diseñado para que de manera autónoma pueda garantizar la trazabilidad de las existencias y correcta clasificación de los bienes.	Tiene permiso de lectura sobre las tablas de referencia de bienes del esquema Catálogo.  Se le permite leer, insertar y modificar las tablas relacionadas a Donación en el esquema Negocio.	Se le prohíbe el acceso al resto del esquema Catálogo.  Se le niega la capacidad de alterar y eliminar en el resto de tablas dentro del esquema Negocio.

		Se le permite la lectura, inserción y modificación de las tablas relacionadas a las transacciones de bienes del esquema Transacciones.	Se le restringe el acceso a Donacion_Monetaria y alterar o eliminar en el resto de tablas del esquema Transacciones.
db_gestor_reportes	Se ha diseñado para ser un consumidor de datos, su única función es la generación de informes, gráficos y análisis a partir de los datos preprocesados que obtiene.	Se le permite la lectura de todo el esquema Análisis.	Se le prohíbe cualquier otra acción dentro de la Base de Datos.
db_docente	Está diseñado específicamente para el docente de la asignatura.	Tiene permiso de lectura sobre toda la Base de Datos.	Se le prohíbe cualquier otra acción dentro de la Base de Datos.

- Usuarios.

Una vez definidos los roles operativos de la Base de Datos, estos se asignan a usuarios que representan a las personas que interactúan con la información de la ONG recibiendo los permisos definidos a través de su rol asignado.

Debido a que se trabaja con una Base de Datos autocontenida se ha configurado para que los usuarios, a pesar de que utilicen autenticación de SQL Server, obtengan las políticas de seguridad del sistema operativo Windows.

Usuarios creados y los roles a los que pertenecen:

**usuario\_dba:** db\_admin.

**coordinador\_proyectos:** db\_coordinador\_proyectos.

**gestor\_finanzas:** db\_gestor\_finanzas.

**gestor\_inventario:** db\_gestor\_inventario.

**gestor\_reportes:** db\_gestor\_reportes.

**lector\_maestro:** db\_docente.

- Auditoría de la Base de Datos.

Se ha definido la configuración de Auditoría de Servidor ubicando el destino de los registros en archivos físicos en el sistema operativo, cada archivo de log tiene un tamaño máximo de 50 MB y se mantienen un máximo de veinte archivos, asegurando que no se consuma todo el espacio del disco.

Por otro lado, la Especificación de Auditoría se enfoca en acciones críticas que comprometen la integridad de la información de las personas o la seguridad del sistema, identificando dichas acciones mediante los roles previamente definidos. Si se desea visualizar el registro de Logs, se puede acceder a una vista diseñada para eso en el script.

Tabla . Especificación de auditoría.

ROL	TABLAS SOMETIDAS	PROPÓSITO
db_admin	SCHEMA::Negocio, SCHEMA::Transacciones	Registra cuando el administrador de la Base de Datos accede a datos sensibles y financieros, y monitorea cualquier eliminación de registros en entidades principales.
db_coordinador_proyectos	Proyecto, Beneficiario	Registra la manipulación de los datos personales y de contacto de los beneficiarios, además monitorea el ciclo de vida de los proyectos registrados.
	Beneficiario_Individuo, Beneficiario_Institución	
db_gestor_finanzas	Recibo, Donacion_Monetaria, Donación, Donante, Donante_Empresa, Donante_Individuo	Debido a la alta sensibilidad de los datos que se manejan, monitorea el flujo financiero de las donaciones recibidas por la ONG, también asegura la correcta manipulación de los datos monetarios y controla las correcciones de las transacciones de este tipo.
db_gestor_inventario	Recibo, Donacion_Bienes, Donacion, Donante, Donante_Empresa,	Al también ser datos con alta sensibilidad, se monitorea el ingreso de los bienes donados a la ONG y registra las correcciones de inventario, asegurando la correcta manipulación de la existencia de bienes y el

	Donante_Individuo	control de las correcciones de las transacciones de este tipo.
db_rrhh	Personal	Registra el acceso a la información del personal asegurando la privacidad de los datos. Monitorea y modifica información sensible de todo el personal de la ONG.
Cualquier rol	-	Registra la creación y modificación de roles y usuarios, también la manipulación de los privilegios que poseen y la estructura de la Base de Datos.

### Consultas optimizadas e índices aplicados.

#### Consultas.

Se crearon diferentes consultas avanzadas diseñadas para obtener información detallada de los datos almacenados. Estas consultas hacen uso de funciones ventana que sirven para realizar cálculos sin perder el detalle por fila. Las consultas presentadas muestran el uso de las funciones en casos reales.

A continuación se presentan las consultas creadas junto a su descripción:

- Ranking de donadores. Se utiliza la función ventana RANK(), esta consulta está diseñada para calcular el monto total donado por cada donante, sumando tanto las donaciones monetarias y de bienes, y luego ordenando a dichos donantes para identificar a los de mayor contribución.
- Porcentaje de proyectos según estado. Se utiliza la función ventana COUNT(\*) OVER(PARTITION BY), esta consulta relaciona los proyectos que se encuentran en cada estado con el número total de proyectos registrados en la ONG.
- Personal con más participación en proyectos. Se utiliza la función ventana ROW\_NUMBER() OVER (ORDER BY COUNT ( ) DESC), identifica a los miembros del personal que han participado en el mayor número de proyectos, asignándoles un número de rango único.

#### Índices.

Con el objetivo de optimizar la velocidad y rendimiento de las consultas realizadas en las tablas que conforman la base de datos se crearon diferentes índices tanto del tipo NONCLUSTERED como UNIQUE.

Los índices de tipo NONCLUSTERED se crearon con propósito de acelerar consultas en tablas muy solicitadas con carga frecuente en columnas por solicitudes tipo WHERE, JOIN y LIKE. Estos tipos de índices se crearon sobre todo en las tablas principales que se encuentran dentro del esquema Negocio.

Los índices UNIQUE se crearon con el propósito de preservar la integridad de datos y conservar las reglas del modelo normalizado de la Base de Datos, se utilizan en tablas catálogo, tablas cruzadas por relaciones muchos a muchos y también en las tablas donde se generan códigos utilizando secuencias que no deben repetirse.

Lista de índices algunos índices relevantes que fueron definidos para una optimización enfocada en el acceso por nombre y IDs:

- CREATE INDEX IX\_Proyecto\_lider\_proyecto ON Negocio.Proyecto(lider\_proyecto);
- CREATE INDEX IX\_Donacion\_id\_proyecto ON Negocio.Donacion(id\_proyecto);
- CREATE INDEX IX\_Donacion\_id\_donante ON Negocio.Donacion(id\_donante);
- CREATE INDEX IX\_Personal\_nombrexapellido ON Negocio.Personal(nombre,apellido);
- CREATE INDEX IX\_Proyecto\_nombre ON Negocio.Proyecto(nombre);
- CREATE INDEX IX\_Recibo\_numero\_recibo ON Transacciones.Recibo(numero\_recibo);
- CREATE INDEX IX\_Beneficiario\_Institucion\_nombre ON Negocio.Beneficiario\_Institucion(nombre);
- CREATE INDEX IX\_Donante\_Empresa\_nombre ON Negocio.Donante\_Empresa(nombre);
- CREATE INDEX IX\_Beneficiario\_Individuo\_nombrexapellido ON Negocio.Beneficiario\_Individuo(nombre,apellido);
- CREATE INDEX IX\_Donante\_Individuo\_nombrexapellido ON Negocio.Donante\_Individuo(nombre,apellido);

A continuación se demuestra la velocidad de algunas consultas antes y después de crear índices:

- Ranking de donadores:

Imagen 1. Antes de la creación del índice.

```
WITH Donaciones AS (
SELECT
    d.id AS id_donante,
    COALESCE(di.nombre + ' ' + di.apellido, de.nombre) AS donante,
    COALESCE(dm.monto, 0) AS monto_monetario,
    COALESCE(db.valor_estimado, 0) AS monto_bienes
FROM Negocio.Donante d
LEFT JOIN Negocio.Donante_Individuo di ON d.id = di.id_donante
LEFT JOIN Negocio.Donante_Empresa de ON d.id = de.id
LEFT JOIN Negocio.Donacion don ON don.id_donante = d.id
LEFT JOIN Transacciones.Donacion_Monetaria dm ON don.id = dm.id_donacion
LEFT JOIN Transacciones.Donacion_Bienes db ON don.id = db.id_donacion
)
```

17% ▾

Results Messages

SQL Server parse and compile time:  
CPU time = 0 ms, elapsed time = 38 ms.

(109 rows affected)

Table 'Worktable'.	Scan count 0,	logical reads 0,	physical reads 0,	page server reads 0,	read-ahead reads 0,
Table 'Workfile'.	Scan count 0,	logical reads 0,	physical reads 0,	page server reads 0,	read-ahead reads 0,
Table 'Donacion'.	Scan count 100,	logical reads 200,	physical reads 1,	page server reads 0,	read-ahead reads 0,
Table 'Donante Empresa'.	Scan count 0,	logical reads 209,	physical reads 1,	page server reads 0,	read-ahead reads 0,
Table 'Donante Individuo'.	Scan count 1,	logical reads 2,	physical reads 1,	page server reads 0,	read-ahead reads 0,
Table 'Donacion'.	Scan count 1,	logical reads 4,	physical reads 1,	page server reads 0,	read-ahead reads 0,
Table 'Donacion Monetaria'.	Scan count 1,	logical reads 4,	physical reads 1,	page server reads 0,	read-ahead reads 0,
Table 'Donacion Bienes'.	Scan count 1,	logical reads 2,	physical reads 1,	page server reads 0,	read-ahead reads 0,

SQL Server Execution Time:  
CPU time = 0 ms, elapsed time = 116 ms.

Completion time: 2025-11-25T20:27:32.7289793-06:00

Imagen 2. Después de la creación del índice.

```

WITH Donaciones AS (
    SELECT
        d.id AS id_donante,
        COALESCE(di.nombre + ' ' + di.apellido, de.nombre) AS donante,
        COALESCE(dm.monto, 0) AS monto_monetario,
        COALESCE(db.valor_estimado, 0) AS monto_bienes
    FROM Negocio_Donante d
    LEFT JOIN Negocio_Donante_Individuo di ON d.id = di.id_donante
    LEFT JOIN Negocio_Empresa de ON d.id = de.id
    LEFT JOIN Negocio_Donacion don ON don.id_donante = d.id
    LEFT JOIN Transacciones.Donacion_Monetaria dm ON don.id = dm.id_donaci
    LEFT JOIN Transacciones.Donacion_Bienes db ON don.id = db.id_donacion
),
Totales AS (
    SELECT
        id_donante,
        donante,

```

- Porcentaje de proyecto según su estado:

Imagen 3. Antes de la creación del índice.

```

SELECT DISTINCT
    pe.estado AS estado_proyecto,
    COUNT(*) OVER (PARTITION BY pe.estado) AS total_por_estado,
    CAST(
        COUNT(*) OVER (PARTITION BY pe.estado) * 100.0 /
        COUNT(*) OVER (
            AS DECIMAL(10,2)
        ) AS porcentaje
FROM Negocio.Proyecto p
INNER JOIN Catalogo.Proyecto_Estado pe
ON p.id_proyecto_estado = pe.id;

```

117 %

Results Messages

SQL Server parse and compile time:  
CPU time = 15 ms, elapsed time = 19 ms.

(5 rows affected)

Table 'Worktable'. Scan count 6, logical reads 826, physical reads 0, page server reads 0  
Table 'Proyecto'. Scan count 1, logical reads 30, physical reads 1, page server reads 0  
Table 'Proyecto\_Estado'. Scan count 1, logical reads 2, physical reads 1, page server reads 0

SQL Server Execution Times:  
CPU time = 0 ms, elapsed time = 7 ms.

Completion time: 2025-11-25T20:29:18.90183929-06:00

Imagen 4. Después de la creación del índice.

```
SELECT DISTINCT
    pe.estado AS estado_proyecto,
    COUNT(*) OVER (PARTITION BY pe.estado) AS total_por_estado
FROM Proyecto p
INNER JOIN Catalogo.Proyecto_Estado pe
    ON p.id_proyecto_estado = pe.id;
```

100 %

Results Messages

Tiempo de análisis y compilación de SQL Server:  
 Tiempo de CPU = 0 ms, tiempo transcurrido = 6 ms.

(5 rows affected)

Tabla "Worktable". Número de examen 6, lecturas lógicas 826, lecturas físicas 0.  
 Tabla "Proyecto". Número de examen 1, lecturas lógicas 30, lecturas físicas 1, 1  
 Tabla "Catalogo.Proyecto\_Estado". Número de examen 1, lecturas lógicas 2, lecturas físicas 1

Tiempos de ejecución de SQL Server:  
 Tiempo de CPU = 0 ms, tiempo transcurrido = 16 ms.

Completion time: 2025-11-25T20:27:36.0333899-06:00

- Personal con más participación en proyectos:

Imagen 5. Antes de la creación del índice.

```

SELECT
    per.nombre + ' ' + per.apellido AS lider,
    COUNT(pp.id_proyecto) AS cantidad_proyectos,

    ROW_NUMBER() OVER (
        ORDER BY COUNT(pp.id_proyecto) DESC
    ) AS ranking
FROM Negocio.Personal per
INNER JOIN Negocio.Personal_Proyecto pp
    ON per.id = pp.id_personal
GROUP BY per.nombre, per.apellido;

```

Imagen 6. Después de la creación del índice.

```
SELECT
    per.nombre + ' ' + per.apellido AS lider,
    COUNT(pp.id_proyecto) AS cantidad_proyectos,
    ROW_NUMBER() OVER (
        ORDER BY COUNT(pp.id_proyecto) DESC
    ) AS ranking
FROM Negocio.Personal per
INNER JOIN Negocio.Personal_Proyecto pp
    ON per.id = pp.id_personal
GROUP BY per.nombre, per.apellido;
```

16

Results Messages

Tempo de análisis y compilación de SQL Server:  
 Tiempo de CPU = 0 ms, tiempo transcurrido = 7 ms.

(496 rows affected)

Tabla "Worktable". Número de examen 0, lecturas lógicas 0, lecturas físicas (

Tabla "Personal\_Proyecto". Número de examen 1, lecturas lógicas 4, lecturas (

Tabla "Personal". Número de examen 1, lecturas lógicas 15, lecturas físicas (

Tempos de ejecución de SQL Server:  
 Tiempo de CPU = 0 ms, tiempo transcurrido = 124 ms.

Completion time: 2025-11-25T20:29:17.8692508-06:00

Se evidencia que la implementación de los índices ayuda a que las consultas sean más rápidas y eficientes. Asimismo, en el esquema Administrativo se encuentra una vista llamada v\_Fragmentacion\_Indices que debe utilizar el DBA para consultar el estado de fragmentación de los índices para así darles su respectivo mantenimiento.

### Estrategia de dimensionamiento, respaldo y recuperación.

Una Base de Datos con arquitectura robusta y seguridad lógica también debe garantizar que su información siempre esté disponible, accesible y protegida, por lo tanto se presentan las siguientes estrategias que abordan la tolerancia a errores y el crecimiento de la ONG.

#### Dimensionamiento.

Se refiere al espacio de almacenamiento que ocupan los objetos de la Base de Datos, como parte del Plan de Capacidad, se listan el dimensionamiento total de cada tabla del sistema, tomando en cuenta los índices, logs y crecimiento a futuro, usando la fórmula:  $(\text{Fila prom.} + 7) \times \text{Filas estimadas} \times \text{Factor crecimiento} \times \text{Índices} \times \text{Paginación} / (1024 * 1024)$

- Personal:  $(268 + 7) * 1000 * 1.3 * 1.9 * 1.1 / (1024 * 1024) = 0.713 \text{ MB}$
- Proyecto.:  $(774 + 7) * 200 * 1.3 * 1.9 * 1.1 / (1024 * 1024) = 0.405 \text{ MB}$
- Beneficiario:  $(344 + 7) * 1500 * 1.3 * 1.3 * 1.1 / (1024 * 1024) = 0.933 \text{ MB}$
- Donante:  $(292 + 7) * 100 * 1.3 * 1.3 * 1.1 / (1024 * 1024) = 0.053 \text{ MB}$
- Donación:  $(27 + 7) * 400 * 1.3 * 1.9 * 1.1 / (1024 * 1024) = 0.035 \text{ MB}$
- Personal\_Proyecto:  $(15 + 7) * 970 * 1.3 * 1.3 * 1.1 / (1024 * 1024) = 0.038 \text{ MB}$
- Proyecto\_Beneficiario:  $(15 + 7) * 717 * 1.3 * 1.3 * 1.1 / (1024 * 1024) = 0.028 \text{ MB}$
- Beneficiario\_Institucion:  $(241 + 7) * 59 * 1.3 * 1.9 * 1.1 / (1024 * 1024) = 0.038 \text{ MB}$
- Beneficiario\_Individuo:  $(164 + 7) * 1441 * 1.3 * 1.6 * 1.1 / (1024 * 1024) = 0.538 \text{ MB}$
- Donante\_Individuo:  $(164 + 7) * 50 * 1.3 * 1.6 * 1.1 / (1024 * 1024) = 0.019 \text{ MB}$
- Donante\_Empresa :  $(233 + 7) * 50 * 1.3 * 1.9 * 1.1 / (1024 * 1024) = 0.031 \text{ MB}$
- Recibo:  $(222 + 7) * 400 * 1.3 * 1.6 * 1.1 / (1024 * 1024) = 0.2 \text{ MB}$
- Donacion\_Monetaria:  $(21 + 7) * 270 * 1.3 * 1.0 * 1.1 / (1024 * 1024) = 0.01 \text{ MB}$
- Donación\_Bienes:  $(325 + 7) * 130 * 1.3 * 1.0 * 1.1 / (1024 * 1024) = 0.059 \text{ MB}$

Tamaño total de la bd: 3.1MB

Tamaño del log diario:  $3.1 * 0.20 = 0.62 \text{ MB}$

Tamaño del diferencial:  $0.62 * 7 = 4.34 \text{ MB}$

#### Plan de Respaldo y Recuperación.

Debido a que la Base de Datos maneja transacciones tanto financieras, de bienes y de proyectos, se requiere una frecuencia de respaldo considerable y una capacidad de recuperación rápida.

- Estrategia de Respaldo.  
La cantidad máxima aceptable de datos que se pueden perder es de 4 horas debido al tipo de registros que se manejan, por ello se propone una combinación de respaldos completos, diferenciales y de registro de transacciones de la siguiente manera:

Tipo de Respaldo	Frecuencia	Hora programada
Completo	Semanal	Domingos a las 03:00 AM
Diferencial	Diario	Lunes a Sábado a las 10:00 PM
Registro de Transacciones	Cada 4 horas	Lunes a Sábado, 08:00 AM, 12:00 PM, 04:00 PM, 08:00 PM

- Estrategia de Recuperación.

El tiempo máximo que la Base de Datos puede estar fuera de servicio después de una falla es de 2 horas para no paralizar las operaciones diarias de la ONG, por ello se propone el siguiente procedimiento de recuperación:

1. Diagnóstico inicial de la falla. Identificación del tipo y alcance de la falla.
2. Secuencia de restauración:
  - Respaldo Completo más reciente usando NORECOVERY.
  - Respaldo Diferencial más reciente usando NORECOVERY.
  - Aplicación de los Registros de Transacciones siguientes al Respaldo Diferencial, hasta el archivo antes de la falla.
  - Utilizar la restauración de punto en el tiempo de la Base de Datos para restaurar exactamente al momento anterior de la falla.
  - Finalizar restaurando con RECOVERY para que la Base de Datos sea nuevamente accesible.

### Automatización de Respaldos.

Las estrategias mencionadas anteriormente se pueden automatizar usando el servicio SQL SERVER AGENT, útil para la gestión proactiva de la Base de Datos. Por ello, se muestra la creación de Jobs, los cuales son una secuencia de pasos que SQL Server ejecuta según el orden indicado, en este caso se han programado para realización de los Respaldos Completos y Registros de Transacciones.

Imagen 7. Evidencia de job para Respaldo Completo.

```
USE msdb;

EXEC msdb.dbo.sp_add_job
    @job_name = N'Job_Full_Backup_DonacionesDB',
    @enabled = 1,
    @description = N'Respaldo completo semanal';

EXEC msdb.dbo.sp_add_jobstep
    @job_name = N'Job_Full_Backup_DonacionesDB',
    @step_name = N'Backup_Full_Step',
    @subsystem = N'TSQL',
    @command = N'BACKUP DATABASE DonacionesDB
    TO DISK = ''C:\Backup\DonacionesDB_FULL_${ESCAPE_QUOTES}(DATE)}.bak''
    WITH INIT, STATS = 10, COMPRESSION;',
    @database_name = N'DonacionesDB';

EXEC msdb.dbo.sp_add_schedule
    @schedule_name = N'Schedule_Semanal_Dom_03h_FULL',
    @freq_type = 8,
    @freq_interval = 1,
    @freq_recurrence_factor = 1,
    @active_start_time = 030000;

EXEC msdb.dbo.sp_attach_schedule
    @job_name = N'Job_Full_Backup_DonacionesDB',
    @schedule_name = N'Schedule_Semanal_Dom_03h_FULL';
```

Messages

Table 'sysmainplan\_subplans'. Scan count 1, logical reads 0, physical reads 0, page server reads 0.

SQL Server Execution Times:  
CPU time = 0 ms, elapsed time = 7 ms.

SQL Server Execution Times:  
CPU time = 0 ms, elapsed time = 0 ms.

SQL Server Execution Times:  
CPU time = 79 ms, elapsed time = 124 ms.

Completion time: 2025-11-26T10:59:25.0058422-06:00

Imagen 8. Evidencia de job para Registro de Transacciones.

```
USE msdb;

EXEC msdb.dbo.sp_add_job
    @job_name = N'Job_Log_Backup_DonacionesDB',
    @enabled = 1,
    @description = N'Respaldo de Registro de Transacciones.';

EXEC msdb.dbo.sp_add_jobstep
    @job_name = N'Job_Log_Backup_DonacionesDB',
    @step_name = N'Backup_Log_Step',
    @subsystem = N'TSQL',
    @command = N'BACKUP LOG DonacionesDB
    TO DISK = ''C:\Backup\DonacionesDB_LOG_${ESCAPE_QUOTES}(DATE)}_${ESCAPE_QUOTES}(TIME)}.trn''
    WITH NOINIT, STATS = 10, COMPRESSION;',
    @database_name = N'DonacionesDB';

EXEC msdb.dbo.sp_add_schedule
    @schedule_name = N'Schedule_4_Horas_LOG',
    @freq_type = 4,
    @freq_interval = 1,
    @freq_subday_type = 8,
    @freq_subday_interval = 4,
    @active_start_time = 080000;

EXEC msdb.dbo.sp_attach_schedule
    @job_name = N'Job_Log_Backup_DonacionesDB',
    @schedule_name = N'Schedule_4_Horas_LOG';
```

Messages

Table 'sysmainplan\_subplans'. Scan count 1, logical reads 0, physical reads 0, page server reads 0, read-ahead reads 0, page se

SQL Server Execution Times:  
CPU time = 0 ms, elapsed time = 9 ms.

SQL Server Execution Times:  
CPU time = 0 ms, elapsed time = 0 ms.

SQL Server Execution Times:  
CPU time = 63 ms, elapsed time = 117 ms.

Completion time: 2025-11-26T11:12:16.6077160-06:00

### Power BI.

Se hizo uso de Power BI para generar un reporte que se centra en la parte de Donaciones, permite poder visualizar de forma interactiva variables que influyen en las donaciones y cómo cambian dependiendo de lo que se seleccione.

El reporte está compuesto por los siguientes componentes:

- Total ingreso en Donaciones. Utiliza una medida llamada TotalDonacion que devuelve el valor total en dólares de todas las donaciones, tanto monetarias como de bienes.
- Estado del proyecto. Son una serie de botones que contienen los cinco estados en los que pueden estar los proyectos, si se presiona uno de ellos cambia el valor dentro del total ingreso filtrando por ese estado.

- Total de Donaciones por Mes. Gráfico de columnas agrupadas que utiliza TotalDonacion y fecha\_donacion, devuelve un gráfico donde se pueden observar los meses y la cantidad de donaciones que se realizan en cada uno.
- Donación por Categoría. Treemap que utiliza el tipo de categoría a la que puede pertenecer un proyecto y los valores de TotalDonacion, permitiendo visualizar las categorías con las donaciones que recibieron.
- Tipo Donación. Gráfico de anillos, que utiliza los dos tipos de donación posible, siendo Monetaria o Bienes, mostrando el porcentaje que representa cada uno del total de donaciones.
- Tipo Donante: Gráfico de anillos, que utiliza dos tipos de donantes posibles, siendo Empresa o Individuo, mostrando el porcentaje que representan cada uno del total de donaciones.
- Método pago donación. Gráfico de barras agrupadas, utiliza el metodo\_pago de donación monetaria y la medida Total\_Donacion\_Monetaria, devolviendo un gráfico donde se puede ver la comparación entre cada uno de los métodos de pago y lo que representan en el total
- Grupo de tarjetas. Se crearon cuatro tarjetas que contienen datos relevantes, como la cantidad de donantes actuales, cantidad de beneficiarios actuales, cantidad de proyectos y los proyectos activos en ese momento.

Imagen 9. Muestra del reporte creado con Power BI.

