

Elaboración de un modelo relacional de base de datos, para la Administración y Seguimiento de Proyectos del Departamento Nacional de Planeación

Paula M. Bohorquéz M.¹, Diana C. Gómez B.²

Facultad de Ingeniería y Ciencias Básicas

Universidad Central

Maestría en Analítica de Datos

Curso de Bases de Datos

Bogotá, Colombia

¹pbohorquezm@ucentral.edu.co

²dgomez10@ucentral.edu.co

Octubre 18, 2024

Contents

1	Introducción	3
2	Características del proyecto de investigación que hace uso de Bases de Datos	3
2.1	Título del proyecto de investigación	3
2.2	Objetivo general	3
2.2.1	Objetivos específicos	3
2.3	Alcance	3
2.4	Pregunta de investigación	3
2.5	Hipótesis	3
3	Reflexiones sobre el origen de datos e información	4
3.1	¿Cuál es el origen de los datos e información ? (Max 100 Palabras) - (<i>Primera entrega</i>) . . .	4
3.2	¿Cuáles son las consideraciones legales o éticas del uso de la información? (Max 100 Palabras) - (<i>Primera entrega</i>)	4
3.3	¿Cuáles son los retos de la información y los datos que utilizara en la base de datos en términos de la calidad y la consolidación? (Max 100 Palabras) - (<i>Primera entrega</i>)	4
3.4	¿Qué espera de la utilización de un sistema de Bases de Datos para su proyecto? (Max 100 Palabras) - (<i>Primera entrega</i>)	4
4	Diseño del Modelo de Datos del SMBD (Sistema Manejador de Bases de Datos)(<i>Primera entrega</i>)	5
4.1	Características del SMBD (Sistema Manejador de Bases de Datos) para el proyecto (<i>Primera entrega</i>)	5
4.2	Diagrama modelo de datos (<i>Primera entrega</i>)	5
4.3	Imágenes de la Base de Datos (<i>Primera entrega</i>)	5
4.4	Código SQL - lenguaje de definición de datos (DDL) (<i>Primera entrega</i>)	7
4.5	Código SQL - Manipulación de datos (DML) (<i>Primera entrega</i>)	9
4.6	Código SQL + Resultados: Vistas (<i>Primera entrega</i>)	10
4.7	Código SQL + Resultados: Triggers (<i>Primera entrega</i>)	10
4.8	Código SQL + Resultados: Funciones (<i>Primera entrega</i>)	11
4.9	Código SQL + Resultados: procedimientos almacenados (<i>Primera entrega</i>)	11

5	Bases de Datos No-SQL (<i>Segunda entrega</i>)	13
5.1	Diagrama Bases de Datos No-SQL (<i>Segunda entrega</i>)	13
5.2	SMBD utilizado para la Base de Datos No-SQL (<i>Segunda entrega</i>)	13
6	Aplicación de ETL (Extract, Transform, Load) y Bodega de Datos (<i>Tercera entrega</i>)	14
6.1	Ejemplo de aplicación de ETL y Bodega de Datos (<i>Tercera entrega</i>)	14
6.2	Automatización de Datos (<i>Tercera entrega</i>)	14
6.3	Integración de Datos (<i>Tercera entrega</i>)	14
7	Proximos pasos (<i>Tercera entrega</i>)	15
8	Lecciones aprendidas (<i>Tercera entrega</i>)	16
9	Bibliografía	17

1 Introducción

El Departamento Nacional de Planeación (DNP) es la entidad técnica encargada de diseñar, coordinar y apoyar la planificación del presupuesto de los recursos de inversión del país. Actualmente, existen diversas fuentes de información que permiten conocer el manejo de los proyectos, incluyendo la asignación de presupuestos y el avance de cada uno, contratos, entre otros. Aunque al día de hoy no se dispone de una base de datos consolidada, es fundamental contar con una que optimice administrar y hacer seguimiento de estos proyectos para una gestión mas eficiente y precisa.

En este contexto, el objetivo de este trabajo es diseñar un modelo de base de datos que permita gestionar y analizar eficientemente los datos relacionados con los proyectos aprobados por el DNP. Este modelo, relacional, mejorará el seguimiento de los proyectos permitiendo la generación de consultas específicas y garantizando la trazabilidad de los mismos según la entidad financiera. Así facilitará un monitoreo más efectivo a lo largo del tiempo .

2 Características del proyecto de investigación que hace uso de Bases de Datos

2.1 Título del proyecto de investigación

Elaboración de un modelo relacional de base de datos, para la Administración y Seguimiento de Proyectos del Departamento Nacional de Planeación (DNP).

2.2 Objetivo general

Diseñar un modelo de base de datos que permita gestionar y analizar de manera eficiente los datos relacionados con los proyectos aprobados por el DNP.

2.2.1 Objetivos específicos

- Evaluar la calidad de los datos para garantizar su integridad y coherencia, asegurando que la información utilizada en el modelo de base de datos sea precisa y confiable..
- Elaborar un diagrama entidad-relación (ER) para representar de manera estructurada las entidades y relaciones de los proyectos del Departamento Nacional de Planeación, facilitando el diseño y comprensión del modelo de base de datos.
- Desarrollar un conjunto de consultas que validen el correcto funcionamiento del modelo de base de datos implementado, permitiendo verificar la precisión en la gestión y recuperación de la información.

2.3 Alcance

El alcance de este proyecto es diseñar un modelo de base de datos que permita gestionar y analizar de manera eficiente los datos relacionados con los proyectos aprobados por el Departamento Nacional de Planeación (DNP) durante el período 2023-2024 para Colombia.

2.4 Pregunta de investigación

¿Cómo diseñar un modelo de base de datos que permita, de manera eficiente, visualizar y analizar la gestión de los proyectos aprobados por el Departamento Nacional de Planeación (DNP)?

2.5 Hipotesis

El diseño de un modelo de base de datos relacional eficiente permitirá una gestión y análisis más efectivo de los proyectos aprobados por el Departamento Nacional de Planeación (DNP), mejorando la trazabilidad de los proyectos y la visibilidad de su avance, lo que contribuirá a un mayor control y aprovechamiento de los recursos destinados a dichos proyectos.

3 Reflexiones sobre el origen de datos e información

El DNP al ser una entidad pública debe garantizar el acceso a la información según la Ley 1712 de 2014; además, el decreto 1081 de 2015 especifica la responsabilidad de las entidades públicas y la disposición de los datos a través de la plataforma digital de Datos Abiertos de Colombia (<https://www.datos.gov.co/>) de donde se tomaron 6 fuentes de datos para ser analizados y procesados.

3.1 ¿Cual es el origen de los datos e información ? (Max 100 Palabras) - (*Primera entrega*)

El DNP obtiene la información a través de plataformas como el Sistema Electrónico de Contratación Pública (SECOP), el Sistema de Monitoreo, Seguimiento, Control y Evaluación (SMSCE), y otros repositorios de datos abiertos proporcionados por el Gobierno de Colombia. Estos sistemas permiten a las entidades públicas reportar el estado de avance, los recursos asignados y la ejecución de los proyectos, asegurando que la información esté disponible para el público. Además, el DNP puede complementar esta información con datos de otras entidades y ministerios que también gestionan o participan en proyectos financiados con recursos públicos. Todo esto bajo la premisa de promover la transparencia, la eficiencia en el uso de los recursos y el control ciudadano sobre las inversiones del Estado colombiano.

3.2 ¿Cuales son las consideraciones legales o eticas del uso de la información? (Max 100 Palabras) - (*Primera entrega*)

El uso de la información sobre los proyectos que sigue el DNP en Colombia se rige por consideraciones legales y éticas clave. La Ley de Transparencia (1712 de 2014) garantiza el acceso público a estos datos, mientras que la Ley de Protección de Datos Personales (1581 de 2012) protege la privacidad en caso de información sensible. Éticamente, se promueve la transparencia, la rendición de cuentas y el uso responsable de los datos. Además, se debe respetar la confidencialidad y los derechos de autor, asegurando que la información no sea manipulada ni utilizada de forma malintencionada o para fines indebidos.

3.3 ¿Cuales son los retos de la información y los datos que utilizara en la base de datos en terminos de la calidad y la consolidación? (Max 100 Palabras) - (*Primera entrega*)

Los retos en la gestión de la información en las bases de datos del DNP incluyen garantizar la calidad de los datos, asegurando precisión, completitud. La normalización es clave para consolidar datos de distintas fuentes. Muchas veces no todas las entidades tienen la información actualizada lo que incrementa la dificultad de unificar en una sola base toda la información.

3.4 ¿Que espera de la utilización de un sistema de Bases de Datos para su proyecto? (Max 100 Palabras) - (*Primera entrega*)

Se espera una gestión eficiente y organizada de la información, que facilite la recolección, almacenamiento y consulta de datos de manera rápida y precisa, para poder generar una visualización ordenada y única de todos los datos, reduciendo duplicidad que es lo más común en estos datos al venir de tantas fuentes de información.

4 Diseño del Modelo de Datos del SMBD (Sistema Manejador de Bases de Datos) (Primera entrega)

4.1 Características del SMBD (Sistema Manejador de Bases de Datos) para el proyecto (Primera entrega)

4.2 Diagrama modelo de datos (Primera entrega)

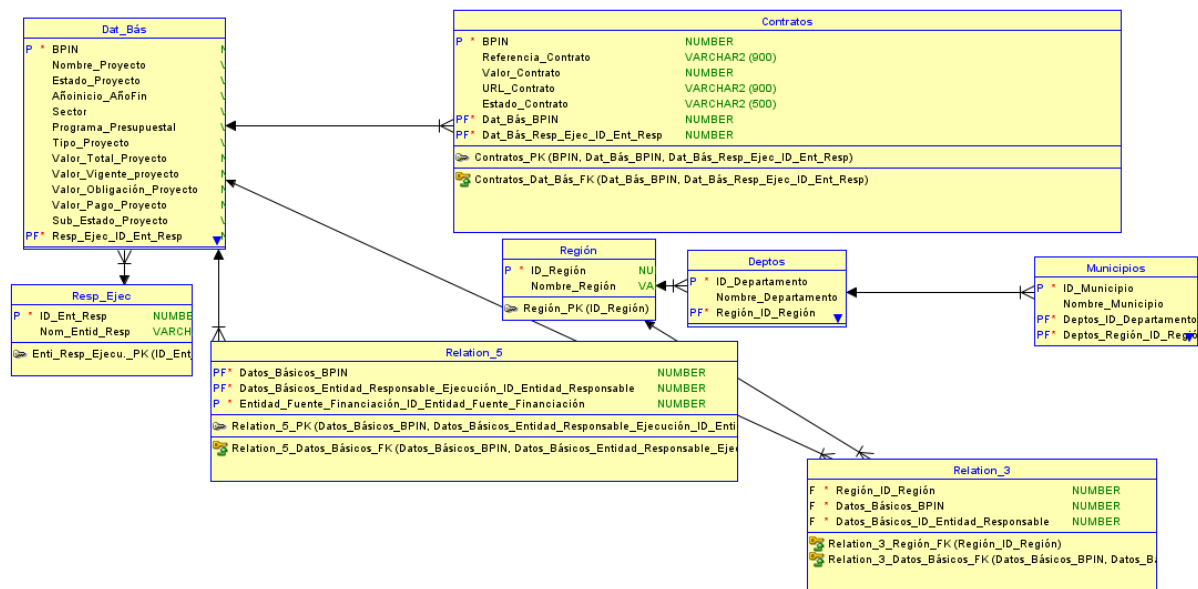


Figure 1: Diagrama Entidad-Relación Proyecto DNP

4.3 Imágenes de la Base de Datos (Primera entrega)

COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1 BPIN	NUMBER	No	(null)	1 (null)	
2 NOMBRE_PROYECTO	VARCHAR2(1000 ...	Yes	(null)	2 (null)	
3 ESTADO_PROYECTO	VARCHAR2(500 B...	Yes	(null)	3 (null)	
4 AÑOINICIO_AÑOFIN	VARCHAR2(200 B...	Yes	(null)	4 (null)	
5 SECTOR	VARCHAR2(500 B...	Yes	(null)	5 (null)	
6 PROGRAMAPresup...	VARCHAR2(900 B...	Yes	(null)	6 (null)	
7 TIPO_PROYECTO	VARCHAR2(900 B...	Yes	(null)	7 (null)	
8 VALOR_TOTAL_PRO...	NUMBER	Yes	(null)	8 (null)	
9 VALOR_VIGENTE_P...	NUMBER	Yes	(null)	9 (null)	
10 VALOR_OBLIGACIÓ...	NUMBER	Yes	(null)	10 (null)	
11 VALOR_PAGO_PROY...	NUMBER	Yes	(null)	11 (null)	
12 SUB_ESTADO_PROY...	VARCHAR2(1000 ...	Yes	(null)	12 (null)	
13 RESP_EJEC_ID_EN...	NUMBER	No	(null)	13 (null)	

Figure 2: Base de datos Datos Básicos

	COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1	BPIN	NUMBER	No	(null)	1	(null)
2	REFERENCIA_CONTRATO	VARCHAR2(900 BYTE)	Yes	(null)	2	(null)
3	VALOR_CONTRATO	NUMBER	Yes	(null)	3	(null)
4	URL_CONTRATO	VARCHAR2(900 BYTE)	Yes	(null)	4	(null)
5	ESTADO_CONTRATO	VARCHAR2(500 BYTE)	Yes	(null)	5	(null)
6	DAT_BÁS_BPIN	NUMBER	No	(null)	6	(null)
7	DAT_BÁS_RESP_EJECUTIVO	NUMBER	No	(null)	7	(null)

Figure 3: Base de datos Contratos

	COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1	ID_REGIÓN	NUMBER	No	(null)	1	(null)
2	NOMBRE_REGIÓN	VARCHAR2(60 BYTE)	Yes	(null)	2	(null)

Figure 4: Base de datos Región

	COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1	ID_PROVEEDOR	NUMBER	No	(null)	1	(null)
2	NOMBRE_PROVEEDOR	VARCHAR2(500 BYTE)	No	(null)	2	(null)
3	TIPO_DOCUMENTO	VARCHAR2(50 BYTE)	Yes	(null)	3	(null)

Figure 5: Base de datos Proveedores

	COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1	ID_DEPARTAMENTO	NUMBER	No	(null)	1	(null)
2	NOMBRE_DEPARTAMENTO	VARCHAR2(500 BYTE)	Yes	(null)	2	(null)
3	REGIÓN_ID_REGIÓN	NUMBER	No	(null)	3	(null)

Figure 6: Base de datos Departamentos

PROYECTO V1.sql x DCGB x MUNICIPIOS x					
Columnas Datos Model Restricciones Permisos Estadísticas Disparadores Flashback Dependencias Detalles Particiones Índices SQL					
Acciones...					
	COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID COMMENTS
1	ID_MUNICIPIO	NUMBER	No	(null)	1 (null)
2	NOMBRE_MUNICIPIO	VARCHAR2(500 B...	Yes	(null)	2 (null)
3	DEPTOS_ID_DEPAR...	NUMBER	No	(null)	3 (null)
4	DEPTOS_REGIÓN_I...	NUMBER	No	(null)	4 (null)

Figure 7: Base de datos Municipios

PROYECTO V1.sql x DCGB x RESP_EJEC x					
Columnas Datos Model Restricciones Permisos Estadísticas Disparadores Flashback Dependencias Detalles Particiones Índices SQL					
Acciones...					
	COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID COMMENTS
1	ID_ENT_RESP	NUMBER	No	(null)	1 (null)
2	NOM_ENTID_RESP	VARCHAR2(500 B...	Yes	(null)	2 (null)

Figure 8: Base de datos Responsable de Ejecución

4.4 Código SQL - lenguaje de definición de datos (DDL) (Primera entrega)

```

CREATE TABLE contratos (
    bpin NUMBER NOT NULL,
    referencia_contrato VARCHAR2(900),
    valor_contrato NUMBER,
    url_contrato VARCHAR2(900),
    estado_contrato VARCHAR2(500)
);

ALTER TABLE contratos
    ADD CONSTRAINT contratos_pk PRIMARY KEY ( bpin);

CREATE TABLE dat_bas (
    bpin NUMBER NOT NULL,
    nombre_proyecto VARCHAR2(1000),
    estado_proyecto VARCHAR2(500),
    añoinicio_añoфин VARCHAR2(200),
    sector VARCHAR2(500),
    programa_presupuestal VARCHAR2(900),
    tipo_proyecto VARCHAR2(900),
    valor_total_proyecto NUMBER,
    valor_vigente_proyecto NUMBER,
    valor_obligación_proyecto NUMBER,
    valor_pago_proyecto NUMBER,
    sub_estado_proyecto VARCHAR2(1000),
    resp_ejec_id_ent_resp NUMBER NOT NULL
);

ALTER TABLE dat_bas ADD CONSTRAINT dat_bas_pk PRIMARY KEY
( bpin,resp_ejec_id_ent_resp );

CREATE TABLE deptos (
    id_departamento NUMBER NOT NULL,

```

```
nombre_departamento VARCHAR2(500)
);
```

```
ALTER TABLE deptos ADD CONSTRAINT deptos_pk PRIMARY KEY
(id_departamento);
```

```
CREATE TABLE fuent_finan (
id_fuente_finan      NUMBER NOT NULL,
"Enti_Fuente_Financ." VARCHAR2(500),
valor_vigente_fuente  NUMBER,
valor_pagado_fuente   NUMBER);
```

```
ALTER TABLE fuent_finan
ADD CONSTRAINT fuent_finan_pk PRIMARY KEY ( id_fuente_finan);
```

```
CREATE TABLE municipios (
id_municipio          NUMBER NOT NULL,
nombre_municipio      VARCHAR2(500)
);
```

```
ALTER TABLE municipios
ADD CONSTRAINT municipios_pk PRIMARY KEY ( id_municipio);
```

```
CREATE TABLE proveedores (
id_proveedor          NUMBER NOT NULL,
nombre_proveedor      VARCHAR2(500) NOT NULL,
tipo_documento        VARCHAR2(50));
```

```
ALTER TABLE proveedores
ADD CONSTRAINT proveedores_pk PRIMARY KEY ( id_proveedor);
```

```
CREATE TABLE region1 (
id_region             NUMBER NOT NULL,
nombre_region         VARCHAR2(60));
```

```
ALTER TABLE region1 ADD CONSTRAINT region1_pk PRIMARY KEY ( id_region );
```

```
CREATE TABLE resp_ejec (
id_ent_resp           NUMBER NOT NULL,
nom_entid_resp        VARCHAR2(500));
```

```
CREATE TABLE relation_70 (
región_id_región      NUMBER NOT NULL,
datos_básicos_bpin    NUMBER NOT NULL,
dat_bas_resp_ejec_id_ent_resp_   NUMBER NOT NULL
);
```

```
CREATE TABLE relation_50 (
    dat_bas_bpin        NUMBER NOT NULL,
    dat_bas_resp_ejec_id_ent_resp_   NUMBER NOT NULL
);
```



```

ALTER TABLE relation_71
  ADD CONSTRAINT relation_71_pk PRIMARY KEY ( dat_bas_bpin,
    dat_bas_resp_ejec_id_ent_resp_);

ALTER TABLE contratos
  ADD CONSTRAINT contratos_dat_bas_fk FOREIGN KEY ( dat_bás_bpin,
    dat_bás_resp_ejec_id_ent_resp)
    REFERENCES dat_bas ( bpin, resp_ejec_id_ent_resp );

ALTER TABLE deptos
  ADD CONSTRAINT deptos_región_fk FOREIGN KEY ( región_id_región )
    REFERENCES región ( id_región );

ALTER TABLE municipios
  ADD CONSTRAINT municipios_deptos_fk FOREIGN KEY
    (deptos_id_departamento,deptos_región_id_región )
  REFERENCES deptos ( id_departamento, región_id_región );

ALTER TABLE relation_70
  ADD CONSTRAINT relation_70_datos_básicos_fk FOREIGN KEY
    (DATOS_BÁSICOS_BPIN,dat_bas_resp_ejec_id_ent_resp_ )
  REFERENCES dat_bas ( bpin,resp_ejec_id_ent_resp );

ALTER TABLE relation_70
  ADD CONSTRAINT relation_70_región_fk FOREIGN KEY ( región_id_región )
    REFERENCES región ( id_región );

```

Los códigos anteriores crean el esquema de la base de datos para recibir información sobre los proyectos del Departamento Nacional de Proyectos, descritos de la siguiente manera:

- * Dat_Bas: Tabla que almacena información de todos los proyectos, incluyendo el valor del proyecto y el estado de su ejecución.
- * Resp_Ejec: Tabla donde se registra la entidad responsable del proyecto junto con su nombre correspondiente.
- * Región: Ubicación del proyecto a nivel nacional.
- * Contratos: Tabla que contiene información consolidada sobre el contrato del proyecto, su estado de ejecución y la URL de seguimiento.
- * Proveedores: Tabla que reúne los datos de los proveedores asociados a los contratos.
- * Deptos: Tabla que lista los departamentos a nivel nacional junto con su código de homologación.
- * Municipios: Tabla que presenta los municipios a nivel nacional, también con su código de homologación.
- * Relaciones: Contiene las relaciones de uno a muchos entre cada una de las tablas.

4.5 Código SQL - Manipulación de datos (DML) *(Primera entrega)*

```

INSERT INTO CONTRATOS (BPIN, REFERENCIA_CONTRATO, VALOR_CONTRATO, URL_CONTRATO, ESTADO_CONTRATO)
VALUES (2023157200002,'MSN OP 06 DE
2023',831849119.7,'','En ejecución');

```

```

INSERT INTO DEPTOS (ID_DEPARTAMENTO, NOMBRE_DEPARTAMENTO, "REGIÓN_ID_REGIÓN")
VALUES (5,'Antioquia',2);

```

```

INSERT INTO REGION1 (ID_REGION,NOMBRE_REGIO) VALUES (3,'Caribe');

INSERT INTO
PROVEEDORES(ID_PROVEEDOR,NOMBRE_PROVEEDOR,TIPO_DOCUMENTO) VALUES
(1040322659,'Eduardo Arango Tamayo','Cedula de Ciudadania');

INSERT INTO DAT_BAS
(BPIN,NOMBRE_PROYECTO,ESTADO_PROYECTO,AÑOINICIO_AÑOFIN,SECTOR,
PROGRAMA_PRESUPUESTAL,TIPO_PROYECTO,VALOR_TOTAL_PROYECTO,
VALOR_VIGENTE_PROYECTO,VALOR_OBLIGACIÓN_PROYECTO,
VALOR_PAGO_PROYECTO,SUB_ESTADO_PROYECTO,RESP_EJEC_ID_ENT_RESP) VALUES
(2023053060033,'Apoyo A La Rendición Y Presentación De Gestión De
La Administración Municipal En El Municipio De Giraldo','En Ejecución',2023-2024,
'Gobierno Territorial',
'4599 Fortalecimiento a la gestión y dirección de la administración pública
territorial','T',54850519,54850519,43300000,43300000,'En ejecución (PGN, Territorio)',5306);

DELETE FROM municipios
WHERE ROWID NOT IN (
    SELECT MIN(ROWID)
    FROM municipios
    GROUP BY id_municipio);

```

4.6 Código SQL + Resultados: Vistas (Primera entrega)

```

CREATE OR REPLACE VIEW vista_contratos_activos AS
SELECT bpin, referencia_contrato, valor_contrato, estado_contrato
FROM contratos
WHERE estado_contrato = 'En ejecución';

```

Esta vista muestra todos los contratos que están en ejecución de la tabla contratos.

```

CREATE OR REPLACE VIEW vista_proyectos_contratos AS
SELECT d.bpin, d.nombre_proyecto, c.referencia_contrato, c.valor_contrato, c.estado_contrato
FROM dat_bas d
JOIN contratos c ON d.bpin = c.bpin;

```

Esta vista relaciona los contratos con los proyectos de la tabla dat_bas

4.7 Código SQL + Resultados: Triggers (Primera entrega)

```

CREATE OR REPLACE TRIGGER check_valor_contrato
BEFORE INSERT ON contratos
FOR EACH ROW
BEGIN
    IF :NEW.valor_contrato < 0 THEN
        RAISE_APPLICATION_ERROR(-20001,
        'El valor del contrato no puede ser negativo.');
```

```

END IF;
END;

```

Se crea trigger para asegurar que no se puedan insertar contratos con un valor negativo

en la tabla contratos. Si alguien intenta hacerlo, se genera un error y la inserción se cancela. Esto se hace con el fin de mantener la integridad de los datos en la base de datos.

```
CREATE OR REPLACE TRIGGER update_estado_contrato
BEFORE UPDATE ON contratos
FOR EACH ROW
BEGIN
    -- Verificar si el valor del contrato ha cambiado
    IF :NEW.valor_contrato <> :OLD.valor_contrato THEN
        -- Cambiar el estado a 'Actualizado'
        :NEW.estado_contrato := 'Actualizado';
    END IF;
END;
```

Se crea trigger para asegurar que una vez que se actualice el valor de un contrato en la tabla contratos, su estado se cambie automáticamente a "Actualizado". Puedes modificar la lógica de actualización del estado según tus necesidades.

4.8 Código SQL + Resultados: Funciones (*Primera entrega*)

Esta función devuelve los detalles de los proyectos asociados a un contrato específico.

```
CREATE OR REPLACE FUNCTION obtener_proyecto_por_contrato(
    p_bpin NUMBER
) RETURN SYS_REFCURSOR IS
    v_cursor SYS_REFCURSOR;
BEGIN
    OPEN v_cursor FOR
    SELECT d.bpin, d.nombre_proyecto, c.referencia_contrato, c.valor_contrato, c.estado_contrato
    FROM dat_bas d
    JOIN contratos c ON d.bpin = c.bpin
    WHERE c.bpin = p_bpin;

    RETURN v_cursor;
END obtener_proyecto_por_contrato;
```

La siguiente función devuelve el número de contratos que están en ejecución.

```
CREATE OR REPLACE FUNCTION contar_contratos_activos RETURN NUMBER IS
    v_count NUMBER;
BEGIN
    SELECT COUNT(*)
    INTO v_count
    FROM contratos
    WHERE estado_contrato = 'En ejecución';

    RETURN v_count;
END contar_contratos_activos;
```

4.9 Código SQL + Resultados: procedimientos almacenados (*Primera entrega*)

Este procedimiento cuenta el número de contratos que están celebrados y devuelve el resultado.

```
CREATE OR REPLACE PROCEDURE contar_contratos_celebrados(  
    o_count OUT NUMBER  
) IS  
BEGIN  
    SELECT COUNT(*)  
    INTO o_count  
    FROM contratos  
    WHERE estado_contrato = 'CELEBRADO';  
END contar_contratos_celebrados;
```

5 Bases de Datos No-SQL (*Segunda entrega*)

5.1 Diagrama Bases de Datos No-SQL (*Segunda entrega*)

5.2 SMBD utilizado para la Base de Datos No-SQL (*Segunda entrega*)

- 6 Aplicación de ETL (Extract, Transform, Load) y Bodega de Datos** (*Tercera entrega*)
 - 6.1** Ejemplo de aplicación de ETL y Bodega de Datos (*Tercera entrega*)
 - 6.2** Automatización de Datos (*Tercera entrega*)
 - 6.3** Integración de Datos (*Tercera entrega*)

7 Proximos pasos (*Tercera entrega*)

8 Lecciones aprendidas (*Tercera entrega*)

9 Bibliografía