

Desarrollador: Miguel Eduardo Rodríguez Garcés

Fecha de inicio del proyecto: 06/04/2024

Fecha de finalización del proyecto: 12/04/2024

Teléfono: +57 3122176158

INTRODUCCIÓN

El presente proyecto tiene como finalidad mejorar la estructura de la base de datos **ProyectoUsuarios**, debido a que es un sistema de base de datos simple y no cuenta con algunos registros relevantes como el **tipo de contrato**, debido a que un trabajador puede tener uno o más trabajos. La **categoría de la empresa, modalidad de trabajo, cargo empleado, compañía o empresa** y **ciudad** se separan como tablas independientes, ya que al separar como una tabla independiente y relacionarla con la tabla de empresa puede reducir errores al momento de ingresar la información correspondiente. Para llevar a cabo esta solución se van a implementar técnicas SQL como **INSERT SELECT** para insertar registros a una tabla mediante una consulta **SELECT**, **UPDATE** para corregir algunas inconsistencias cuando sea requerido y creaciones de nuevas tablas.

METODOLOGÍA

La base de datos denominada como **ProyectoUsuarios** consiste en dos tablas que está conformada por el siguiente esquema (ver figura 1):

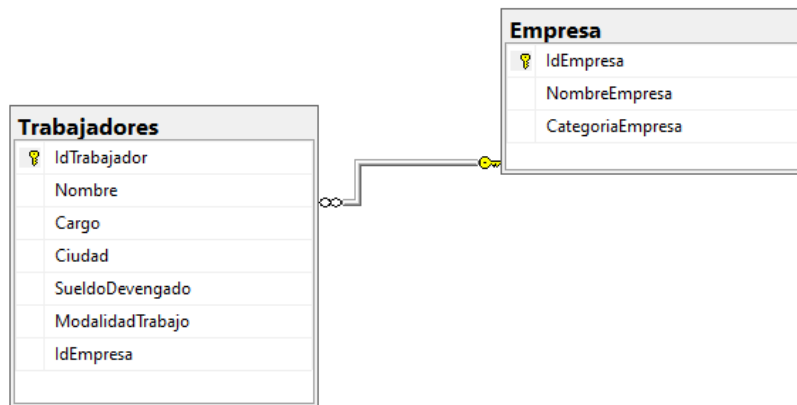


Figura 1. Modelo de Entidad Relación.

Realizando un análisis al modelo de entidad de relación se observa que es posible mejorar el sistema. El modelo propuesto es el siguiente (Ver figura 2):

Desarrollador: Miguel Eduardo Rodríguez Garcés

Fecha de inicio del proyecto: 06/04/2024

Fecha de finalización del proyecto: 12/04/2024

Teléfono: +57 3122176158

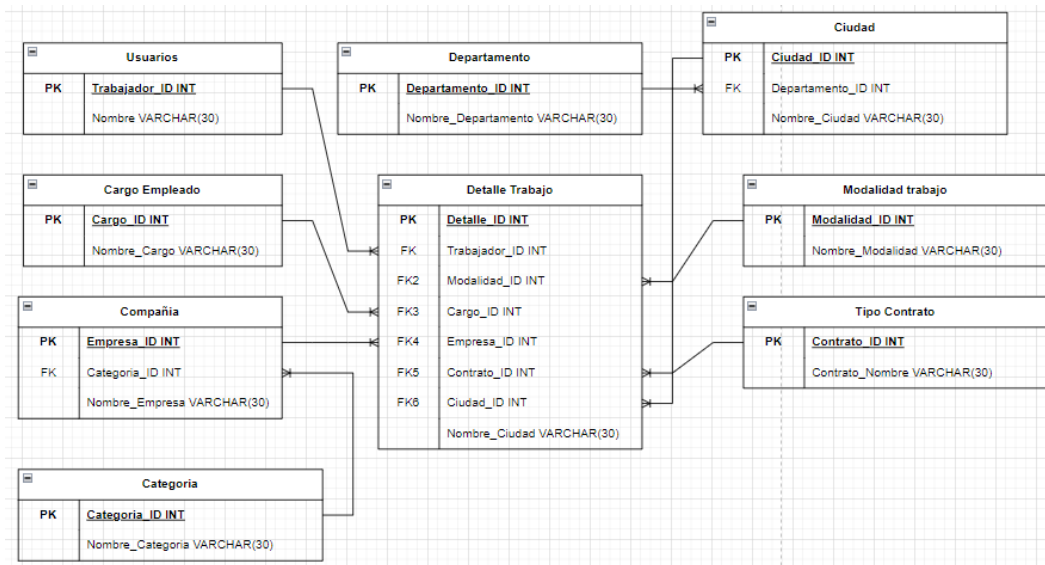


Figura 2. Nuevo Modelo de Entidad Relación.

Antes de llegar al modelo propuesto se realizarán algunos procedimientos como creación de tablas e incluso técnicas de ETL. Se observa que en la variable de **cargo** hay registros inconsistentes (ver figura 3).

| | Cargo |
|----|-------------------------------|
| 12 | Especialista de pruebas |
| 13 | Especialista en servidores |
| 14 | Gerente de proyectos TI |
| 15 | Ingeniero de ciberseguridad |
| 16 | Ingeniero de datos |
| 17 | Ingeniero de infraestructura |
| 18 | Ingeniero de proyectos TI |
| 19 | Ingeniero Devops |
| 20 | Ingenierode ciberseguridad |
| 21 | Lider Bussinness Intelligence |

Figura 3. Datos inconsistentes.

Esa inconsistencia se corrige con la siguiente línea de código:

```
UPDATE Trabajadores SET Cargo = 'Ingeniero de ciberseguridad' WHERE Cargo = 'Ingenierode ciberseguridad';
```

```
UPDATE Trabajadores SET Cargo = 'Desarrollador Fullstack' WHERE Cargo = 'Desarrollador web fullstack';
```

Desarrollador: Miguel Eduardo Rodríguez Garcés

Fecha de inicio del proyecto: 06/04/2024

Fecha de finalización del proyecto: 12/04/2024

Teléfono: +57 3122176158

Una vez corregido todos esos detalles se procede con la creación de las tablas mapeadas en la figura 2:

```
CREATE TABLE ModalidadTrabajo(  
    Modalidad_Id INT PRIMARY KEY,  
    ModalidadTrabajo VARCHAR(30) NOT NULL  
);  
  
-- Insertando los registros  
  
INSERT INTO ModalidadTrabajo  
SELECT ROW_NUMBER() OVER (ORDER BY ModalidadTrabajo DESC) AS Modalidad_Id,  
ModalidadTrabajo FROM Trabajadores  
GROUP BY ModalidadTrabajo;
```

El código consiste en la creación de la tabla **ModalidadTrabajo**, la cual contiene información de las modalidades de trabajo como lo son **híbrido**, **remoto** y **presencial**. Por último, la función **INSERT INTO ModalidadTrabajo** consiste en insertar algunos registros en dicha tabla y al final esos registros se insertan utilizando un **SELECT** en la tabla **Trabajadores**, la cual en ese momento se encuentra sin realizar cambios como en la figura 1.

Se aplican esas mismas instrucciones para la creación de las otras tablas.

Tabla cargo empleado

```
CREATE TABLE CargoEmpleado(  
    Cargo_Id INT PRIMARY KEY,  
    Nombre_Cargo VARCHAR(30) NOT NULL  
);  
  
INSERT INTO CargoEmpleado  
SELECT ROW_NUMBER() OVER (ORDER BY Cargo ASC) AS Cargo_Id, Cargo FROM Trabajadores  
GROUP BY Cargo;
```

Tabla usuarios

```
CREATE TABLE Usuarios(  
    Trabajador_ID INT PRIMARY KEY,  
    Nombre_Usuario VARCHAR(30)  
);  
  
-- Insertando registros  
  
INSERT INTO Usuarios  
SELECT IdTrabajador, Nombre FROM Trabajadores;
```

Desarrollador: Miguel Eduardo Rodríguez Garcés

Fecha de inicio del proyecto: 06/04/2024

Fecha de finalización del proyecto: 12/04/2024

Teléfono: +57 3122176158

Tabla categoria de empresa

```
CREATE TABLE Categoria(  
    Categoria_ID INT PRIMARY KEY,  
    Categoria_Nombre VARCHAR(30)  
);  
  
INSERT INTO Categoria  
SELECT ROW_NUMBER() OVER (ORDER BY CategoriaEmpresa ASC) AS Categoria_ID,  
    CategoriaEmpresa FROM Empresa  
GROUP BY CategoriaEmpresa;
```

Tabla compañía o empresa

```
CREATE TABLE Empresa_Prueba(  
    Empresa_ID INT PRIMARY KEY,  
    Nombre_Empresa VARCHAR(30),  
    Categoria_ID INT,  
    CONSTRAINT FK_CATEGORIA FOREIGN KEY (Categoria_ID) REFERENCES  
    Categoria(Categoria_ID)  
        ON UPDATE CASCADE  
        ON DELETE CASCADE  
);  
  
INSERT INTO Empresa_Prueba  
SELECT e.IdEmpresa, e.NombreEmpresa, c.Categoria_ID FROM Empresa e INNER JOIN  
    Categoria c  
ON c.Categoria_Nombre = e.CategoriaEmpresa;
```

Nota: Como se encuentra una tabla llamada **Empresa** el sistema no permite que existan dos tablas con el mismo nombre. Posteriormente se procede a realizar un cambio de nombre a **Compania**.

```
EXEC sp_rename Empresa_Prueba, Compania;
```

Tabla departamento

```
CREATE TABLE Departamentos(  
    Departamento_ID INT PRIMARY KEY IDENTITY(1, 1),  
    Nombre_Departamento VARCHAR(30)  
);  
  
-- Insertando registros en la tabla de departamentos  
  
INSERT INTO Departamentos VALUES  
    ('Atlantico'),  
    ('Bogota D.C'),  
    ('Santander'),  
    ('Valle del Cauca'),  
    ('Antioquia');
```

Desarrollador: Miguel Eduardo Rodríguez Garcés

Fecha de inicio del proyecto: 06/04/2024

Fecha de finalización del proyecto: 12/04/2024

Teléfono: +57 3122176158

El objetivo principal por el cual se crea la tabla de **Departamentos** antes que la tabla de **Ciudad** es porque la clave primaria de la tabla (**Departamento_ID**) tendrá como función servir como **clave foránea** y relacionar con la tabla de **Ciudad**.

Tabla Ciudad

```
CREATE TABLE Ciudad(  
    Ciudad_ID INT PRIMARY KEY,  
    Nombre_Ciudad VARCHAR(30),  
    Departamento_ID INT,  
    CONSTRAINT FK_DEPARTAMENTO FOREIGN KEY (Departamento_ID) REFERENCES  
Departamentos(Departamento_ID)  
    ON UPDATE CASCADE  
);  
  
-- Insertando registros  
  
INSERT INTO Ciudad  
SELECT ROW_NUMBER() OVER (ORDER BY Ciudad ASC) AS Ciudad_ID, Ciudad, NULL AS  
Departamento_ID FROM Trabajadores  
GROUP BY Ciudad;
```

Tabla DetalleTrabajo

La tabla **DetalleTrabajo** es la tabla de hechos que contiene la información detallada del trabajador como la compañía donde trabaja, el puesto que ejerce, la ciudad donde desempeña sus funciones, la fecha de ingreso a la compañía y el sueldo devengado. Para la creación de dicha tabla se hizo de la siguiente manera:

```
CREATE TABLE DetalleTrabajo(  
    Detalle_ID INT PRIMARY KEY IDENTITY(1, 1),  
    Trabajador_ID INT NOT NULL,  
    Modalidad_ID INT,  
    Cargo_ID INT NOT NULL,  
    Empresa_ID INT NOT NULL,  
    Contrato_ID INT,  
    Ciudad_ID INT,  
    Fecha_Ingreso DATE,  
  
    -- Clave foranea con trabajador  
  
    CONSTRAINT FK_TRABAJADOR FOREIGN KEY (Trabajador_ID) REFERENCES  
Usuarios(Trabajador_ID)  
    ON DELETE CASCADE  
    ON UPDATE CASCADE,  
  
    -- Clave foranea con modalidad de trabajo
```

Desarrollador: Miguel Eduardo Rodríguez Garcés

Fecha de inicio del proyecto: 06/04/2024

Fecha de finalización del proyecto: 12/04/2024

Teléfono: +57 3122176158

```
        CONSTRAINT FK_MODALIDAD_TRABAJO FOREIGN KEY (Modalidad_ID) REFERENCES
ModalidadTrabajo(Modalidad_ID)
        ON DELETE CASCADE
        ON UPDATE CASCADE,

        -- Clave foranea con compañía

        CONSTRAINT FK_EMPRESA_TRABAJO FOREIGN KEY (Empresa_ID) REFERENCES
Compania(Empresa_ID)
        ON DELETE CASCADE
        ON UPDATE CASCADE,

        -- Clave foranea con cargo o puesto de trabajo

        CONSTRAINT FK_PUESTO_TRABAJO FOREIGN KEY (Cargo_ID) REFERENCES
CargoEmpleado(Cargo_ID)
        ON DELETE CASCADE
        ON UPDATE CASCADE,

        -- Clave foranea con tipo de contrato

        CONSTRAINT FK_TIPO_CONTRATO FOREIGN KEY (Contrato_ID) REFERENCES
TipoContrato(Tipo_ID)
        ON DELETE CASCADE
        ON UPDATE CASCADE,

        -- Clave foranea con ciudad

        CONSTRAINT FK_CIUADAD FOREIGN KEY (Ciudad_ID) REFERENCES Ciudad(Ciudad_ID)
        ON DELETE CASCADE
        ON UPDATE CASCADE,

    );
```

Desarrollador: Miguel Eduardo Rodríguez Garcés

Fecha de inicio del proyecto: 06/04/2024

Fecha de finalización del proyecto: 12/04/2024

Teléfono: +57 3122176158

CONCLUSIÓN

Se realizan pruebas de manera exitosa y se logran crear las tablas y modelar los datos de acuerdo con la propuesta (figura 2). El resultado fue el siguiente (ver figura 4):

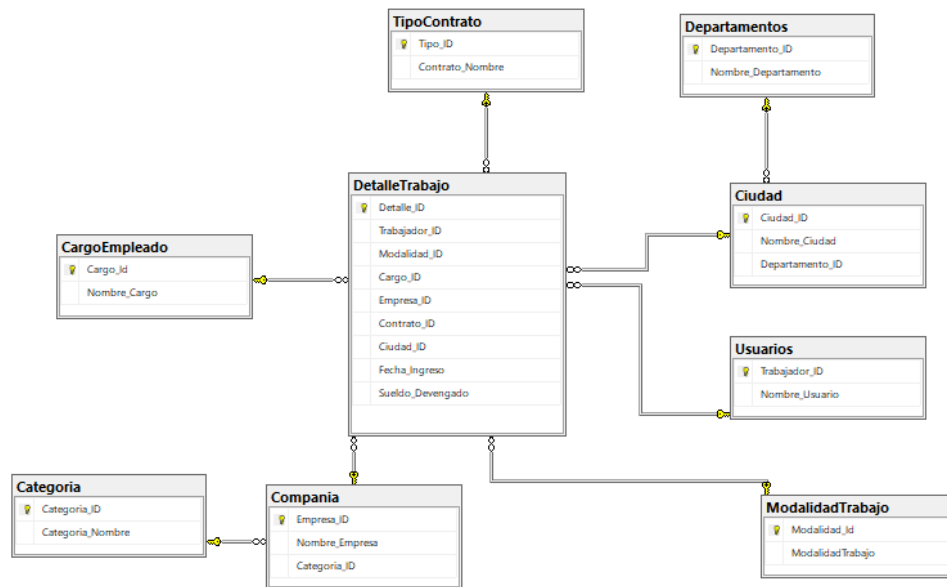


Figura 4. Modelo Entidad de Relación final.

También se reduce de manera significativa el ingreso de datos inconsistentes a la tabla **DetalleTrabajo**, ya que gran parte de las variables corresponde a claves foráneas de las otras tablas y en caso de ingresar un dígito que no se encuentre en ninguna tabla se cancela la operación de insertar información. Para más información acerca de las consultas y códigos revisar en https://github.com/mrodriguez2000/Nueva_DBProyecto_Usuarios.