**Diagrama Entidad-Relación (DER):**

Diagrama

Descripción generada automáticamente

**Viajeros - Documentos:** Relación de uno a muchos. Un viajero puede tener múltiples documentos, pero cada documento está relacionado con un único viajero.

**Viajeros - EntradasViajes / SalidasViajes:** Relación de uno a muchos. Un viajero puede tener múltiples entradas y salidas, pero cada entrada o salida está vinculada a un único viajero.

**Usuarios - EntradasViajes / SalidasViajes:** Relación de uno a muchos. Un usuario puede registrar múltiples entradas y salidas de viajes, pero cada entrada o salida es registrada por un único usuario.

**Codigo de SQL:**

CREATE DATABASE MIGRACION

USE MIGRACION

-- Creación de la tabla Usuarios (Fuerte)

CREATE TABLE Usuarios (

ID\_Usuario INT PRIMARY KEY IDENTITY(1,1),

Username VARCHAR(50) NOT NULL UNIQUE,

Password VARCHAR(255) NOT NULL,

Rol VARCHAR(50) NOT NULL

);

-- Creación de la tabla Viajeros (Fuerte)

CREATE TABLE Viajeros (

ID\_Viajero INT PRIMARY KEY IDENTITY(1,1),

Cedula\_Pasaporte VARCHAR(20) NOT NULL UNIQUE,

Nombre VARCHAR(100) NOT NULL,

Segundo\_Nombre VARCHAR(100) NULL,

Apellido1 VARCHAR(100) NOT NULL,

Apellido2 VARCHAR(100) NOT NULL,

Fecha\_Nacimiento DATE NOT NULL,

Nacionalidad VARCHAR(50) NOT NULL,

Correo VARCHAR(100) NOT NULL,

Sexo CHAR(1) NOT NULL CHECK (Sexo IN ('M', 'F')),

CHECK (Fecha\_Nacimiento <= GETDATE()),

CHECK (Correo LIKE '%\_@\_\_%.\_\_%')

);

-- Creación de la tabla EntradasViajes (Débil)

CREATE TABLE EntradasViajes (

ID\_Entrada INT PRIMARY KEY IDENTITY(1,1),

ID\_Viajero INT NOT NULL,

Fecha\_Entrada DATETIME NOT NULL,

Lugar\_Entrada VARCHAR(100) NOT NULL,

ID\_Usuario INT NOT NULL,

FOREIGN KEY (ID\_Viajero) REFERENCES Viajeros(ID\_Viajero),

FOREIGN KEY (ID\_Usuario) REFERENCES Usuarios(ID\_Usuario),

CHECK (Fecha\_Entrada <= GETDATE())

);

-- Creación de la tabla SalidasViajes (Débil)

CREATE TABLE SalidasViajes (

ID\_Salida INT PRIMARY KEY IDENTITY(1,1),

ID\_Viajero INT NOT NULL,

Fecha\_Salida DATETIME NOT NULL,

Lugar\_Salida VARCHAR(100) NOT NULL,

ID\_Usuario INT NOT NULL,

FOREIGN KEY (ID\_Viajero) REFERENCES Viajeros(ID\_Viajero),

FOREIGN KEY (ID\_Usuario) REFERENCES Usuarios(ID\_Usuario)

);

-- Creación de la tabla Documentos (Débil)

CREATE TABLE Documentos (

ID\_Documento INT PRIMARY KEY IDENTITY(1,1),

Tipo\_Documento VARCHAR(50) NOT NULL,

Numero\_Documento VARCHAR(100) NOT NULL UNIQUE,

Fecha\_Expiracion DATE NOT NULL,

ID\_Viajero INT NOT NULL,

FOREIGN KEY (ID\_Viajero) REFERENCES Viajeros(ID\_Viajero),

CHECK (Fecha\_Expiracion >= GETDATE())

);

-- Creación del trigger para validar la Fecha\_Salida

CREATE TRIGGER trg\_ValidarFechaSalida

ON SalidasViajes

FOR INSERT, UPDATE

AS

BEGIN

IF EXISTS (

SELECT 1

FROM Inserted i

JOIN EntradasViajes e ON i.ID\_Viajero = e.ID\_Viajero

WHERE i.Fecha\_Salida < e.Fecha\_Entrada

)

BEGIN

RAISERROR ('Fecha de salida no puede ser anterior a la fecha de entrada.', 16, 1);

ROLLBACK TRANSACTION;

END

END;

**Análisis del código SQL**

**Tabla Usuarios (Fuerte)**

ID\_Usuario: Clave primaria con autoincremento.

Username: Campo único que no puede ser nulo.

Password: Almacena la contraseña (se asume que está encriptada, dado el tamaño).

Rol: Define el rol del usuario (ej. administrador, operador).

**Tabla Viajeros (Fuerte)**

ID\_Viajero: Clave primaria con autoincremento.

Cedula\_Pasaporte: Identificación única del viajero.

Nombre, Segundo\_Nombre, Apellido1, Apellido2: Información personal del viajero.

Fecha\_Nacimiento: Valida que la fecha de nacimiento sea anterior o igual a la fecha actual.

Nacionalidad: Nacionalidad del viajero.

Correo: Validado para tener un formato adecuado de correo electrónico.

Sexo: Campo con restricción para aceptar solo 'M' o 'F'.

CHECK (Fecha\_Nacimiento <= GETDATE()): Asegura que la fecha de nacimiento no sea futura.

CHECK (Correo LIKE '%\_@\_\_%.\_\_%'): Valida el formato del correo electrónico.

**Tabla EntradasViajes (Débil)**

ID\_Entrada: Clave primaria con autoincremento.

ID\_Viajero: Llave foránea que referencia a la tabla Viajeros.

Fecha\_Entrada: Valida que la fecha de entrada no sea futura.

Lugar\_Entrada: Lugar donde el viajero ingresó.

ID\_Usuario: Llave foránea que referencia a la tabla Usuarios.

Restricciones:

CHECK (Fecha\_Entrada <= GETDATE()): Asegura que la fecha de entrada no sea en el futuro.

**Tabla SalidasViajes (Débil)**

ID\_Salida: Clave primaria con autoincremento.

ID\_Viajero: Llave foránea que referencia a la tabla Viajeros.

Fecha\_Salida: Fecha de salida del viajero.

Lugar\_Salida: Lugar donde el viajero salió.

ID\_Usuario: Llave foránea que referencia a la tabla Usuarios.

Nota: No hay un CHECK explícito, pero se controla a través de un TRIGGER.

**Tabla Documentos (Débil)**

ID\_Documento: Clave primaria con autoincremento.

Tipo\_Documento: Tipo de documento (ej. pasaporte, visa).

Numero\_Documento: Número único del documento.

Fecha\_Expiracion: Valida que la fecha de expiración sea futura o presente.

ID\_Viajero: Llave foránea que referencia a la tabla Viajeros.

Restricciones:

CHECK (Fecha\_Expiracion >= GETDATE()): Asegura que el documento no esté vencido.

**Trigger para Validar la Fecha de Salida**

Verifica que la Fecha\_Salida no sea anterior a la Fecha\_Entrada del mismo viajero.

Si se detecta una inconsistencia, se genera un error y se revierte la transacción.