**BASE DE DATOS ASISTENCIA DOCENTE**

-- crear la base de datos

create database controlasistenciadocente;

go

-- usar la base de datos

use controlasistenciadocente;

go

-- Tabla de usuarios (docentes, administradores, empleados)

CREATE TABLE usuarios (

usuarioid INT IDENTITY(1,1) PRIMARY KEY,

nombre NVARCHAR(50) NOT NULL,

contraseña NVARCHAR(50) NOT NULL, -- cambiado a NVARCHAR para cifrado de contraseña

apellido NVARCHAR(100) NOT NULL,

carnetidentidad INT NOT NULL UNIQUE,

rol NVARCHAR(50) NOT NULL, -- 'docente', 'administrador', 'empleado'

email NVARCHAR(100) NOT NULL, -- email para envío de QR y notificaciones

intentosfallidos INT DEFAULT 0, -- para los intentos fallidos de login

tiempobloqueo DATETIME NULL -- tiempo de bloqueo en caso de tres intentos fallidos

foto VARBINARY(MAX) NULL;

numerocelular INT NOT NULL UNIQUE,

);

GO

-- Tabla de asistencias

CREATE TABLE asistencias (

asistenciaid INT IDENTITY(1,1) PRIMARY KEY,

usuarioid INT FOREIGN KEY REFERENCES usuarios(usuarioid),

fecha DATE NOT NULL,

horaentrada TIME NOT NULL,

horasalida TIME NULL,

estado NVARCHAR(50) NOT NULL, -- 'atrasado', 'presente', 'ausente', 'justificado'

codigoregistro NVARCHAR(50), -- código QR usado para la asistencia

);

GO

-- Tabla de sanciones (solo para administradores)

CREATE TABLE sanciones (

sancionid INT IDENTITY(1,1) PRIMARY KEY,

usuarioid INT FOREIGN KEY REFERENCES usuarios(usuarioid),

fecha DATE NOT NULL,

motivo NVARCHAR(200) NOT NULL, -- atraso, falta, justificación

monto DECIMAL(10, 2) NOT NULL

);

GO

-- Tabla de justificaciones de faltas (enviadas por los docentes)

CREATE TABLE justificaciones (

justificacionid INT IDENTITY(1,1) PRIMARY KEY,

usuarioid INT FOREIGN KEY REFERENCES usuarios(usuarioid),

fecha DATE NOT NULL,

motivo NVARCHAR(200) NOT NULL, -- motivo de la falta

estado NVARCHAR(50) DEFAULT 'pendiente' -- pendiente, aprobada, rechazada

);

GO

-- Tabla de horarios para los docentes

CREATE TABLE horarios\_docentes (

horarioid INT IDENTITY(1,1) PRIMARY KEY,

usuarioid INT FOREIGN KEY REFERENCES usuarios(usuarioid),

horaentrada TIME NOT NULL,

horasalida TIME NOT NULL,

materia NVARCHAR(100) NOT NULL,

curso NVARCHAR(100) NOT NULL,

diasemana NVARCHAR(100) NOT NULL -- Ejemplo: 'Lunes, Miércoles, Viernes'

);

GO

-- Tabla para almacenar tokens QR con expiración

CREATE TABLE tokens\_qr (

tokenid INT IDENTITY(1,1) PRIMARY KEY,

usuarioid INT FOREIGN KEY REFERENCES usuarios(usuarioid),

token NVARCHAR(100) NOT NULL, -- el token del QR

fecha\_generacion DATETIME NOT NULL,

expiraen DATETIME NOT NULL -- tiempo de expiración

);

GO