**Alumnos—**

Jesus Adrian Luzanilla Tapia

Gael Alberto Guerra Landavazo

Jorge Cuevas Gastelum

**ID´s—**

00000252699

00000252522

00000252274

**Asignación—**

Revisión proyecto Base de datos

**Fecha—**

17 de febrero de 2025

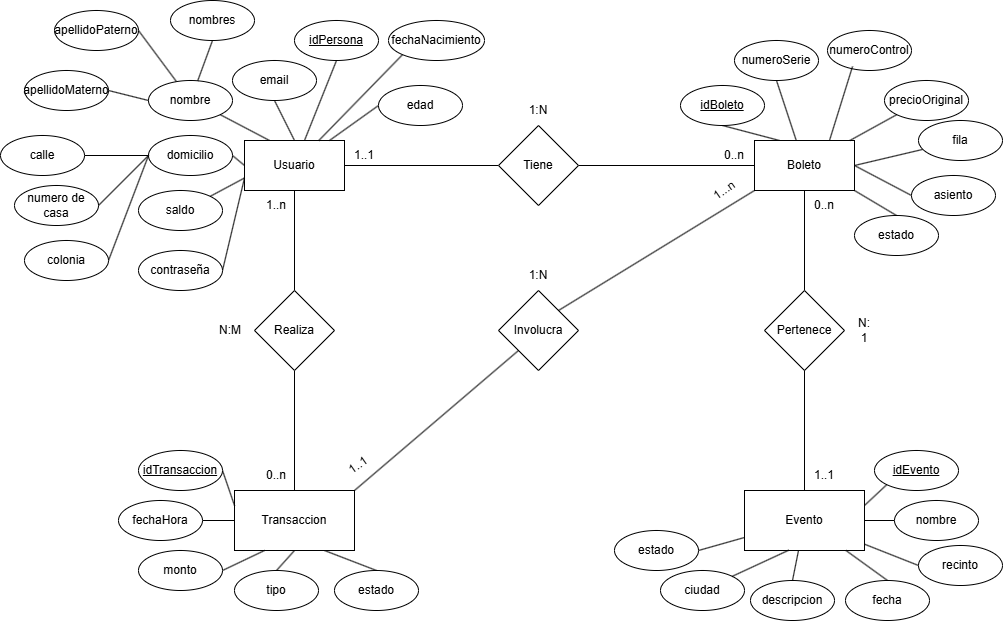
**Materia—**

Bases de datos Avanzadas

**Profesor—**

Christian Gibran Duran Solano

1. Modelo Entidad - Relación de la base de datos



1. Modelo relacional de la base de datos

Usuarios (

idUsuario : INT auto\_incremento requerido,

email : VARCHAR(60) requerido,

nombre : VARCHAR(20) requerido,

contraseña\_hash : VARCHAR(30) requerido,

apellidoPaterno : VARCHAR(20) requerido,

apellidoMaterno : VARCHAR(20) requerido,

fechaNacimiento : DATE requerido,

saldo : DECIMAL requerido default 0,

edad : INT requerido

idDireccion : INT,

)

-se creará una función que incremente la edad cada año

-idDireccion llave foranea que pertenece a DireccionesUsuarios(idDireccion)

DireccionesUsuarios(

idDireccion : INT auto\_incremento requerido,

calle : VARCHAR(30) requerido,

ciudad : VARCHAR(30) requerido,

estado : VARCHAR(30) requerido,

)

Boletos (

idBoleto : INT auto\_incremento requerido,

numeroSerie : VARCHAR(8) requerido,

fila : VARCHAR(8) requerido,

asiento : VARCHAR(8) requerido,

precioOriginal : DECIMAL requerido,

numeroInterno : VARCHAR(10) requerido,

estado : VARCHAR(Disponible, Apartado, Vendido), requerido,

idEvento : INT,

idUsuario: INT

)

- idEvento llave foranea que pertenece a Eventos(idEvento)

- idUsuario llave foranea que pertenece a Usuarios(idUsuario)

Eventos (

idEvento : INT auto\_incremento requerido,

nombre : VARCHAR(50) requerido,

fecha : DATE requerido,

recinto : VARCHAR(40) requerido,

ciudad : VARCHAR(30) requerido,

estado : VARCHAR(30) requerido,

descripcion : VARCHAR(100)

)

Transacciones (

idTransaccion : INT auto\_incremento requerido,

fechaHora : DATE requerido,

monto : DECIMAL requerido,

tipo : VARCHAR(Compra, Reventa) requerido,

estado : VARCHAR(Completada, Cancelada) requerido,

idBoleto : INT,

idComprador : INT,

idVendedor : INT

)

- idBoleto llave foranea que pertenece a Boletos(idBoleto)

- idComprador llave foranea que pertenece a Usuarios(idUsuario)

- idVendedor llave foranea que pertenece a Usuarios(idUsuario)

1. Script SQL de creación de base de datos

CREATE DATABASE bdBoletos;

USE bdBoletos;

CREATE TABLE DireccionesUsuarios (

idDireccion INT AUTO\_INCREMENT PRIMARY KEY,

calle VARCHAR(30) NOT NULL,

ciudad VARCHAR(30) NOT NULL,

estado VARCHAR(30) NOT NULL

);

CREATE TABLE Usuarios (

idUsuario INT AUTO\_INCREMENT PRIMARY KEY,

email VARCHAR(255) NOT NULL UNIQUE,

nombre VARCHAR(20) NOT NULL,

contraseña\_hash VARCHAR(30) NOT NULL,

apellidoPaterno VARCHAR(20) NOT NULL,

apellidoMaterno VARCHAR(20) NOT NULL,

fechaNacimiento DATE NOT NULL,

saldo DECIMAL(10,2) NOT NULL DEFAULT 0,

edad INT NOT NULL,

idDireccion INT NOT NULL,

FOREIGN KEY (idDireccion) REFERENCES DireccionesUsuarios(idDireccion) ON DELETE CASCADE

);

CREATE TABLE Eventos (

idEvento INT AUTO\_INCREMENT PRIMARY KEY,

nombre VARCHAR(30) NOT NULL,

fecha DATETIME NOT NULL,

recinto VARCHAR(30) NOT NULL,

ciudad VARCHAR(30) NOT NULL,

estado VARCHAR(30) NOT NULL,

descripcion VARCHAR(100) NOT NULL

);

CREATE TABLE Boletos (

idBoleto INT AUTO\_INCREMENT PRIMARY KEY,

numeroSerie CHAR(8) NOT NULL UNIQUE,

fila VARCHAR(10) NOT NULL,

asiento VARCHAR(10) NOT NULL,

precioOriginal DECIMAL(10,2) NOT NULL,

numeroInterno VARCHAR(20) NOT NULL,

estado VARCHAR(20) NOT NULL,

idEvento INT NOT NULL,

idUsuario INT NOT NULL,

FOREIGN KEY (idEvento) REFERENCES Eventos(idEvento) ON DELETE CASCADE,

FOREIGN KEY (idUsuario) REFERENCES Usuarios(idUsuario) ON DELETE CASCADE,

CONSTRAINT revisar\_estadoBoleto CHECK (estado IN ('Disponible', 'Apartado', 'Vendido'))

);

CREATE TABLE Transacciones (

idTransaccion INT AUTO\_INCREMENT PRIMARY KEY,

fechaHora DATETIME NOT NULL DEFAULT CURRENT\_TIMESTAMP,

monto DECIMAL(10,2) NOT NULL,

tipo VARCHAR(15) NOT NULL,

estado VARCHAR(20) NOT NULL,

idBoleto INT NOT NULL,

idComprador INT NOT NULL,

idVendedor INT NOT NULL,

FOREIGN KEY (idBoleto) REFERENCES Boletos(idBoleto) ON DELETE CASCADE,

FOREIGN KEY (idComprador) REFERENCES Usuarios(idUsuario) ON DELETE CASCADE,

FOREIGN KEY (idVendedor) REFERENCES Usuarios(idUsuario) ON DELETE CASCADE,

CONSTRAINT chk\_tipo CHECK (tipo IN ('Compra', 'Reventa')),

CONSTRAINT revisar\_estadoTransaccion CHECK (estado IN ('Completada', 'Cancelada', 'Procesando'))

);

-- FUNCIÓN PARA ACTUALIZAR EDADES CADA AÑO

DELIMITER //

CREATE PROCEDURE ActualizarEdades()

BEGIN

UPDATE Usuarios

SET edad = TIMESTAMPDIFF(YEAR, fechaNacimiento, CURDATE());

END //

DELIMITER ;

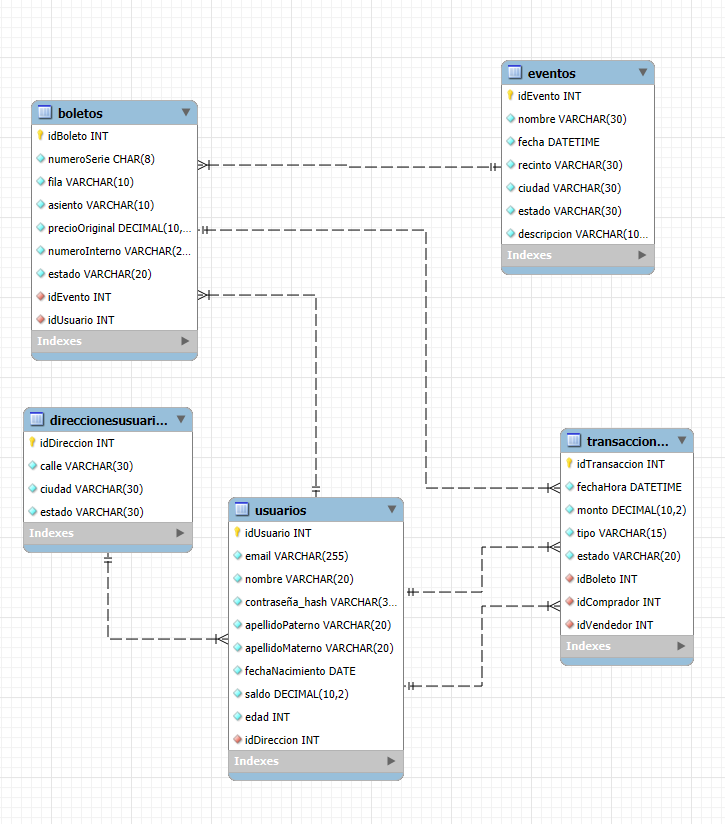
-- EVENTO PARA EJECUTAR LA FUNCIÓN AUTOMÁTICAMENTE CADA AÑO (EL 1 DE ENERO)

CREATE EVENT IF NOT EXISTS EventoActualizarEdades

ON SCHEDULE EVERY 1 YEAR STARTS TIMESTAMP(CURDATE(), '00:00:00')

DO CALL ActualizarEdades();

1. Diagrama de Base de datos generado por el asistente de MySQL Workbench.



1. Storyboard de todo el sistema

