-- Tablas maestras

CREATE TABLE categorias (

id\_categoria SERIAL PRIMARY KEY,

nombre VARCHAR(100) NOT NULL UNIQUE

);

CREATE TABLE departamentos (

id\_departamento SERIAL PRIMARY KEY,

nombre VARCHAR(100) NOT NULL UNIQUE

);

CREATE TABLE ubicaciones (

id\_ubicacion SERIAL PRIMARY KEY,

nombre VARCHAR(100) NOT NULL UNIQUE

);

CREATE TABLE marcas (

id\_marca SERIAL PRIMARY KEY,

nombre VARCHAR(100) NOT NULL UNIQUE

);

-- Tabla de responsables

CREATE TABLE solicitantes (

id\_solicitante SERIAL PRIMARY KEY,

cedula VARCHAR(20) NOT NULL UNIQUE,

nombre VARCHAR(100) NOT NULL,

id\_departamento INTEGER REFERENCES departamentos(id\_departamento)

);

-- Tabla de productos

CREATE TABLE productos (

id\_producto SERIAL PRIMARY KEY,

codigo VARCHAR(50) NOT NULL UNIQUE,

nombre VARCHAR(100) NOT NULL,

id\_marca INTEGER REFERENCES marcas(id\_marca),

id\_categoria INTEGER REFERENCES categorias(id\_categoria)

activo BOOLEAN DEFAULT TRUE

);

-- Tabla de solicitudes

CREATE TABLE solicitudes (

id\_solicitud SERIAL PRIMARY KEY,

id\_departamento INTEGER REFERENCES departamentos(id\_departamento),

id\_solicitante INTEGER REFERENCES solicitantes(id\_solicitante),

numero\_memorandum VARCHAR(50) UNIQUE,

fecha\_solicitud TIMESTAMP WITH TIME ZONE DEFAULT CURRENT\_TIMESTAMP,

comentario TEXT

id\_responsable\_entrega INTEGER REFERENCES usuarios(id);

);

-- Tabla de detalle de solicitudes

CREATE TABLE detalle\_solicitud (

id\_detalle\_solicitud SERIAL PRIMARY KEY,

id\_solicitud INTEGER NOT NULL REFERENCES solicitudes(id\_solicitud),

id\_producto INTEGER NOT NULL REFERENCES productos(id\_producto),

cantidad INTEGER NOT NULL CHECK (cantidad > 0)

);

-- Tabla de inventario

CREATE TABLE inventario (

id\_inventario SERIAL PRIMARY KEY,

id\_producto INTEGER NOT NULL REFERENCES productos(id\_producto),

id\_ubicacion INTEGER REFERENCES ubicaciones(id\_ubicacion),

stock INTEGER NOT NULL DEFAULT 0,

estado\_stock VARCHAR(20) NOT NULL CHECK (estado\_stock IN ('disponible', 'reservado', 'agotado', “stock bajo”))

);

CREATE TABLE solicitudes\_compra (

id SERIAL PRIMARY KEY,

producto VARCHAR(100) NOT NULL,

cantidad INTEGER NOT NULL,

motivo VARCHAR(50) NOT NULL,

prioridad VARCHAR(20) NOT NULL,

proveedor VARCHAR(100),

fecha TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

estado VARCHAR(20) DEFAULT 'Pendiente'

);

CREATE TABLE proveedores (

id\_proveedor SERIAL PRIMARY KEY,

nombre VARCHAR(100) NOT NULL UNIQUE,

contacto VARCHAR(100),

telefono VARCHAR(20),

email VARCHAR(100)

);

CREATE TABLE movimientos (

id\_movimiento SERIAL PRIMARY KEY,

id\_producto INTEGER NOT NULL REFERENCES productos(id\_producto),

tipo VARCHAR(10) NOT NULL CHECK (tipo IN ('Entrada', 'Salida')),

cantidad INTEGER NOT NULL,

id\_ubicacion INTEGER REFERENCES ubicaciones(id\_ubicacion),

id\_responsable INTEGER REFERENCES responsables(id\_responsable),

referencia VARCHAR(100), fecha TIMESTAMP DEFAULT CURRENT\_TIMESTAMP

);

CREATE TABLE usuarios (

id SERIAL PRIMARY KEY,

nombre\_completo VARCHAR(100) NOT NULL,

email VARCHAR(100) UNIQUE NOT NULL,

usuario VARCHAR(50) UNIQUE NOT NULL,

password VARCHAR(64) NOT NULL, -- Almacenará el hash SHA-256

rol VARCHAR(20) DEFAULT 'usuario',

fecha\_registro TIMESTAMP DEFAULT CURRENT\_TIMESTAMP

);

Mariast

123

estas son las tablas involucradas:-- Tablas maestras CREATE TABLE categorias ( id\_categoria SERIAL PRIMARY KEY, nombre VARCHAR(100) NOT NULL UNIQUE ); CREATE TABLE departamentos ( id\_departamento SERIAL PRIMARY KEY, nombre

estas son las tablas involucradas:-- Tablas maestras

CREATE TABLE categorias (

id\_categoria SERIAL PRIMARY KEY,

nombre VARCHAR(100) NOT NULL UNIQUE

activo BOOLEAN DEFAULT TRUE

);

CREATE TABLE departamentos (

id\_departamento SERIAL PRIMARY KEY,

nombre VARCHAR(100) NOT NULL UNIQUE

activo BOOLEAN DEFAULT TRUE

);

CREATE TABLE ubicaciones (

id\_ubicacion SERIAL PRIMARY KEY,

nombre VARCHAR(100) NOT NULL UNIQUE

activo BOOLEAN DEFAULT TRUE

);

CREATE TABLE marcas (

id\_marca SERIAL PRIMARY KEY,

nombre VARCHAR(100) NOT NULL UNIQUE

activo BOOLEAN DEFAULT TRUE

);

-- Tabla de solicitantes

CREATE TABLE solicitantes (

id\_solicitante SERIAL PRIMARY KEY,

cedula VARCHAR(20) NOT NULL UNIQUE,

nombre VARCHAR(100) NOT NULL,

id\_departamento INTEGER REFERENCES departamentos(id\_departamento)

activo BOOLEAN DEFAULT TRUE

);

-- Tabla de productos

CREATE TABLE productos (

id\_producto SERIAL PRIMARY KEY,

codigo VARCHAR(50) NOT NULL UNIQUE,

nombre VARCHAR(100) NOT NULL,

id\_marca INTEGER REFERENCES marcas(id\_marca),

id\_categoria INTEGER REFERENCES categorias(id\_categoria),

activo BOOLEAN DEFAULT TRUE

activo BOOLEAN DEFAULT TRUE

);

-- Tabla de solicitudes

CREATE TABLE solicitudes (

id\_solicitud SERIAL PRIMARY KEY,

id\_departamento INTEGER REFERENCES departamentos(id\_departamento),

id\_solicitante INTEGER REFERENCES solicitantes(id\_solicitante),

numero\_memorandum VARCHAR(50) UNIQUE,

fecha\_solicitud TIMESTAMP WITH TIME ZONE DEFAULT CURRENT\_TIMESTAMP,

comentario TEXT,

id\_responsable\_entrega INTEGER REFERENCES usuarios(id)

activo BOOLEAN DEFAULT TRUE

);

-- Tabla de detalle de solicitudes

CREATE TABLE detalle\_solicitud (

id\_detalle\_solicitud SERIAL PRIMARY KEY,

id\_solicitud INTEGER NOT NULL REFERENCES solicitudes(id\_solicitud),

id\_producto INTEGER NOT NULL REFERENCES productos(id\_producto),

cantidad INTEGER NOT NULL CHECK (cantidad > 0)

);

-- Tabla de inventario

CREATE TABLE inventario (

id\_inventario SERIAL PRIMARY KEY,

id\_producto INTEGER NOT NULL REFERENCES productos(id\_producto),

id\_ubicacion INTEGER REFERENCES ubicaciones(id\_ubicacion),

stock INTEGER NOT NULL DEFAULT 0,

estado\_stock VARCHAR(20) NOT NULL CHECK (estado\_stock IN ('disponible', 'reservado', 'agotado', 'stock bajo'))

);

-- Tabla de solicitudes de compra

CREATE TABLE solicitudes\_compra (

id SERIAL PRIMARY KEY,

producto VARCHAR(100) NOT NULL,

cantidad INTEGER NOT NULL,

motivo VARCHAR(50) NOT NULL,

prioridad VARCHAR(20) NOT NULL,

proveedor VARCHAR(100),

fecha TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

estado VARCHAR(20) DEFAULT 'Pendiente'

);

-- Tabla de proveedores ampliada

CREATE TABLE proveedores (

id\_proveedor SERIAL PRIMARY KEY,

nombre VARCHAR(100) NOT NULL UNIQUE,

contacto VARCHAR(100),

telefono VARCHAR(20),

email VARCHAR(100),

direccion TEXT,

redes\_sociales TEXT,

valoracion INTEGER CHECK (valoracion BETWEEN 1 AND 5),

manejo\_precios VARCHAR(20) CHECK (manejo\_precios IN ('Bajo', 'Medio', 'Alto')),

comentarios TEXT

activo BOOLEAN DEFAULT TRUE

);

-- Tabla de relación proveedor-categoría

CREATE TABLE proveedor\_categoria (

id\_proveedor INTEGER REFERENCES proveedores(id\_proveedor),

id\_categoria INTEGER REFERENCES categorias(id\_categoria),

PRIMARY KEY (id\_proveedor, id\_categoria)

);

-- Tabla de movimientos

CREATE TABLE movimientos (

id\_movimiento SERIAL PRIMARY KEY,

id\_producto INTEGER NOT NULL REFERENCES productos(id\_producto),

tipo VARCHAR(10) NOT NULL CHECK (tipo IN ('Entrada', 'Salida')),

cantidad INTEGER NOT NULL,

id\_ubicacion INTEGER REFERENCES ubicaciones(id\_ubicacion),

id\_responsable INTEGER REFERENCES responsables(id\_responsable),

referencia VARCHAR(100),

fecha TIMESTAMP DEFAULT CURRENT\_TIMESTAMP

);

-- Tabla de usuarios

CREATE TABLE usuarios (

id SERIAL PRIMARY KEY,

nombre\_completo VARCHAR(100) NOT NULL,

email VARCHAR(100) UNIQUE NOT NULL,

usuario VARCHAR(50) UNIQUE NOT NULL,

password VARCHAR(64) NOT NULL, -- Almacenará el hash SHA-256

rol VARCHAR(20) DEFAULT 'usuario',

fecha\_registro TIMESTAMP DEFAULT CURRENT\_TIMESTAMP

activo BOOLEAN DEFAULT TRUE

);

CREATE TABLE proveedor\_producto (

id\_proveedor INTEGER REFERENCES proveedores(id\_proveedor),

id\_producto INTEGER REFERENCES productos(id\_producto),

PRIMARY KEY (id\_proveedor, id\_producto)

);

ALTER TABLE movimientos

DROP CONSTRAINT IF EXISTS movimientos\_id\_responsable\_fkey;

ALTER TABLE movimientos

ADD CONSTRAINT movimientos\_id\_responsable\_fkey

FOREIGN KEY (id\_responsable) REFERENCES usuarios(id);

-- Agregar campo activo a todas las tablas

ALTER TABLE categorias ADD COLUMN activo BOOLEAN DEFAULT TRUE;

ALTER TABLE departamentos ADD COLUMN activo BOOLEAN DEFAULT TRUE;

ALTER TABLE ubicaciones ADD COLUMN activo BOOLEAN DEFAULT TRUE;

ALTER TABLE marcas ADD COLUMN activo BOOLEAN DEFAULT TRUE;

ALTER TABLE solicitantes ADD COLUMN activo BOOLEAN DEFAULT TRUE;

ALTER TABLE proveedores ADD COLUMN activo BOOLEAN DEFAULT TRUE;

ALTER TABLE usuarios ADD COLUMN activo BOOLEAN DEFAULT TRUE;

ALTER TABLE productos ADD COLUMN activo BOOLEAN DEFAULT TRUE;

-- Actualizar registros existentes para que estén activos

UPDATE categorias SET activo = TRUE WHERE activo IS NULL;

UPDATE departamentos SET activo = TRUE WHERE activo IS NULL;

UPDATE ubicaciones SET activo = TRUE WHERE activo IS NULL;

UPDATE marcas SET activo = TRUE WHERE activo IS NULL;

UPDATE solicitantes SET activo = TRUE WHERE activo IS NULL;

UPDATE proveedores SET activo = TRUE WHERE activo IS NULL;

UPDATE usuarios SET activo = TRUE WHERE activo IS NULL;

UPDATE productos SET activo = TRUE WHERE activo IS NULL;