-- Estructura de base de datos para tres tipos de usuarios: Particulares, Empresas y Startups

-- Tabla de Usuarios (tabla base para todos los tipos de usuarios)

CREATE TABLE Usuarios (

id\_usuario SERIAL PRIMARY KEY,

tipo\_usuario VARCHAR(50) NOT NULL CHECK (tipo\_usuario IN ('Particular', 'Empresa', 'Startup')),

nombre VARCHAR(255) NOT NULL,

email VARCHAR(255) UNIQUE NOT NULL,

contrasena VARCHAR(255) NOT NULL,

nivel INT DEFAULT 1,

fecha\_registro TIMESTAMP DEFAULT CURRENT\_TIMESTAMP

);

-- Tabla de Particulares

CREATE TABLE Particulares (

id\_usuario INT PRIMARY KEY REFERENCES Usuarios(id\_usuario),

actividad INT DEFAULT 0 -- Actividad para calcular el nivel

);

-- Tabla de Empresas

CREATE TABLE Empresas (

id\_usuario INT PRIMARY KEY REFERENCES Usuarios(id\_usuario)

);

-- Tabla de Startups

CREATE TABLE Startups (

id\_usuario INT PRIMARY KEY REFERENCES Usuarios(id\_usuario)

);

-- Tabla de Categorías

CREATE TABLE Categorias (

id\_categoria SERIAL PRIMARY KEY,

nombre VARCHAR(255) NOT NULL UNIQUE

);

-- Tabla de Publicaciones (de startups)

CREATE TABLE Publicaciones (

id\_publicacion SERIAL PRIMARY KEY,

id\_startup INT NOT NULL REFERENCES Startups(id\_usuario),

id\_categoria INT REFERENCES Categorias(id\_categoria),

titulo VARCHAR(255) NOT NULL,

contenido TEXT NOT NULL,

fecha\_publicacion TIMESTAMP DEFAULT CURRENT\_TIMESTAMP

);

-- Tabla de Productos (de startups)

CREATE TABLE Productos (

id\_producto SERIAL PRIMARY KEY,

id\_startup INT NOT NULL REFERENCES Startups(id\_usuario),

nombre VARCHAR(255) NOT NULL,

descripcion TEXT NOT NULL,

precio DECIMAL(10, 2) NOT NULL,

fecha\_creacion TIMESTAMP DEFAULT CURRENT\_TIMESTAMP

);

-- Tabla de Likes (de particulares a publicaciones de startups)

CREATE TABLE Likes (

id\_like SERIAL PRIMARY KEY,

id\_publicacion INT NOT NULL REFERENCES Publicaciones(id\_publicacion),

id\_particular INT NOT NULL REFERENCES Particulares(id\_usuario),

fecha\_like TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

UNIQUE(id\_publicacion, id\_particular) -- Evitar likes duplicados

);

-- Tabla de Compras (de particulares a productos de startups)

CREATE TABLE Compras (

id\_compra SERIAL PRIMARY KEY,

id\_producto INT NOT NULL REFERENCES Productos(id\_producto),

id\_particular INT NOT NULL REFERENCES Particulares(id\_usuario),

cantidad INT NOT NULL,

fecha\_compra TIMESTAMP DEFAULT CURRENT\_TIMESTAMP

);

-- Tabla de Retos (de empresas)

CREATE TABLE Retos (

id\_reto SERIAL PRIMARY KEY,

id\_empresa INT NOT NULL REFERENCES Empresas(id\_usuario),

id\_categoria INT REFERENCES Categorias(id\_categoria),

titulo VARCHAR(255) NOT NULL,

descripcion TEXT NOT NULL,

presupuesto DECIMAL(10, 2) NOT NULL,

fecha\_publicacion TIMESTAMP DEFAULT CURRENT\_TIMESTAMP

);

-- Tabla de Propuestas (de startups a retos de empresas)

CREATE TABLE Propuestas (

id\_propuesta SERIAL PRIMARY KEY,

id\_reto INT NOT NULL REFERENCES Retos(id\_reto),

id\_startup INT NOT NULL REFERENCES Startups(id\_usuario),

descripcion TEXT NOT NULL,

fecha\_propuesta TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

estado VARCHAR(50) DEFAULT 'Pendiente' CHECK (estado IN ('Pendiente', 'Aceptada', 'Rechazada'))

);

-- Tabla de Misiones (diarias y semanales para particulares)

CREATE TABLE Misiones (

id\_mision SERIAL PRIMARY KEY,

nombre VARCHAR(255) NOT NULL,

descripcion TEXT NOT NULL,

tipo VARCHAR(50) NOT NULL CHECK (tipo IN ('Diaria', 'Semanal')),

puntos INT NOT NULL

);

-- Tabla de Participacion en Misiones

CREATE TABLE Participacion\_Misiones (

id\_participacion SERIAL PRIMARY KEY,

id\_mision INT NOT NULL REFERENCES Misiones(id\_mision),

id\_particular INT NOT NULL REFERENCES Particulares(id\_usuario),

fecha\_completada TIMESTAMP DEFAULT CURRENT\_TIMESTAMP

);