Manual Tecnico

```
Mapeo fisico de la base de datos:
CREATE SCHEMA REVISTAS DIGITALES;
USE REVISTAS DIGITALES;
CREATE TABLE usuario (
  nombre VARCHAR(100) NOT NULL,
  clave VARCHAR(100) NOT NULL,
  foto LONGBLOB,
  hobbies TEXT,
  intereses TEXT,
  descripcion TEXT,
  rol VARCHAR(15) NOT NULL,
  CONSTRAINT pk_usuario PRIMARY KEY (nombre)
);
CREATE TABLE anuncio (
  id INT AUTO_INCREMENT PRIMARY KEY,
  anunciante VARCHAR(100) NOT NULL,
  fecha DATE NOT NULL,
  tipo VARCHAR(10) NOT NULL,
  vigencia VARCHAR(10) NOT NULL,
  texto TEXT,
  imagen LONGBLOB,
  video LONGBLOB,
  estado VARCHAR(10) NOT NULL,
  precio DECIMAL(10, 2),
  CONSTRAINT fk_anunciante FOREIGN KEY (anunciante) REFERENCES usuario(nombre)
);
CREATE TABLE carteraDigital (
  usuario VARCHAR(100) NOT NULL,
  saldo DECIMAL(10, 2) NOT NULL DEFAULT 0.0,
  CONSTRAINT pk_usuario PRIMARY KEY (usuario),
  CONSTRAINT fk_usuario FOREIGN KEY (usuario) REFERENCES usuario(nombre) ON
DELETE CASCADE
);
CREATE TABLE tipoAnuncio (
  nombre VARCHAR(50) PRIMARY KEY,
  precio DECIMAL(10, 2) DEFAULT 50.00
);
INSERT INTO tipoAnuncio (nombre)
VALUES
```

```
('Txt'),
  ('Txt-Img'),
  ('Video');
CREATE TABLE tag (
  nombre VARCHAR(100) PRIMARY KEY
);
INSERT INTO tag (nombre) VALUES
('Tecnologia'),
('Deportes'),
('Ciencia'),
('Arte'),
('Historia'),
('Economia'),
('Politica'),
('Salud'),
('Literatura'),
('Entretenimiento'),
('Musica'),
('Cine'),
('Viajes'),
('Gastronomia'),
('Educacion'),
('Moda'),
('Cultura'),
('Medio Ambiente'),
('Sociologia'),
('Psicologia');
CREATE TABLE revista (
  nombre VARCHAR(100) NOT NULL,
  editor VARCHAR(100) NOT NULL,
  bloqueo VARCHAR(5),
  fecha DATE,
  costo DECIMAL(10,2),
  descripcion TEXT,
  categoria VARCHAR(50),
  CONSTRAINT pk_revista PRIMARY KEY (nombre),
  CONSTRAINT fk revista editor FOREIGN KEY (editor) REFERENCES usuario(nombre)
);
CREATE TABLE revista_tag (
  revista_nombre VARCHAR(100) NOT NULL,
  tag_nombre VARCHAR(100) NOT NULL,
  CONSTRAINT pk_revista_tag PRIMARY KEY (revista_nombre, tag_nombre),
  CONSTRAINT fk revista tag revista FOREIGN KEY (revista nombre) REFERENCES
revista(nombre).
  CONSTRAINT fk_revista_tag_tag FOREIGN KEY (tag_nombre) REFERENCES tag(nombre)
```

```
);
CREATE TABLE version (
  numero INT NOT NULL,
  PRIMARY KEY (numero)
);
CREATE TABLE publicacion (
  revista VARCHAR(100) NOT NULL,
  version INT NOT NULL,
  pdf LONGBLOB,
  PRIMARY KEY (revista, version),
  CONSTRAINT fk_revista FOREIGN KEY (revista) REFERENCES revista(nombre),
  CONSTRAINT fk_version FOREIGN KEY (version) REFERENCES version(numero)
);
CREATE TABLE suscripcion (
  revista_nombre VARCHAR(100) NOT NULL,
  usuario_nombre VARCHAR(100) NOT NULL,
  fecha DATE.
  CONSTRAINT pk_suscripcion PRIMARY KEY (revista_nombre, usuario_nombre),
  CONSTRAINT fk revista suscripcion FOREIGN KEY (revista nombre) REFERENCES
revista(nombre),
  CONSTRAINT fk_usuario_suscripcion FOREIGN KEY (usuario_nombre) REFERENCES
usuario(nombre)
);
CREATE TABLE megusta (
  revista_nombre VARCHAR(100) NOT NULL,
  usuario nombre VARCHAR(100) NOT NULL,
  PRIMARY KEY (revista_nombre, usuario_nombre),
  CONSTRAINT fk megusta usuario FOREIGN KEY (usuario nombre) REFERENCES
usuario(nombre),
  CONSTRAINT fk_megusta_revista FOREIGN KEY (revista_nombre) REFERENCES
revista(nombre)
);
CREATE TABLE pago (
  id INT AUTO_INCREMENT PRIMARY KEY,
  usuario VARCHAR(100) NOT NULL,
  fecha DATE NOT NULL,
  monto DECIMAL(10, 2) NOT NULL,
  FOREIGN KEY (usuario) REFERENCES usuario(nombre)
);
```

Diagrama Entidad Relacion:

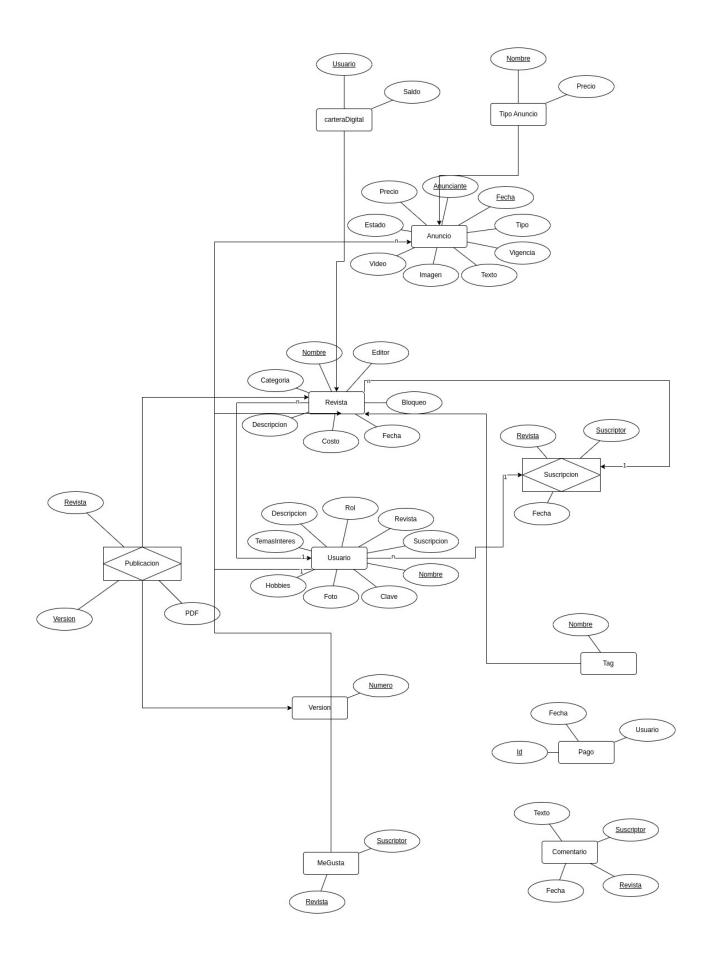


Diagrama de clases:

Usuario

id: int nombre: String clave: String rol: String perfil: ToDo

iniciarSesion() editarPerfil()

Revista

id: int
nombre: String
descripcion: String
fechaCreacion: LocalDate
categoria: CategoriaEnum
etiqueta: EtiquetaEnum
costoDiario: double
autor: Editor
comentarios: Comentario[]
numeroMeGusta: int

Anuncio

id: int tipo: TipoAnuncioEnum contenido: ToDo vigencia: vigenciaEnum propietario: Anunciante

Publicacion

+ field: type

+ method(type): type

Pago

+ field: type

+ method(type): type

Registro

+ field: type

+ method(type): type

Busqueda

+ field: type

+ method(type): type

Comentario

contenido: String autor: Suscriptor fecha: LocalDate revista: Revista

Reporte

generarReporte()

TipoAnuncio

+ field: type

+ method(type): type

Listado

+ field: type

+ method(type): type

Encriptacion

+ field: type

+ method(type): type

CarteraDigital

saldoActual: double

recargarSaldo() verificarSaldo() debitarSaldo()

MeGusta

autor: Suscriptor revista: Revista fecha: LocalDate