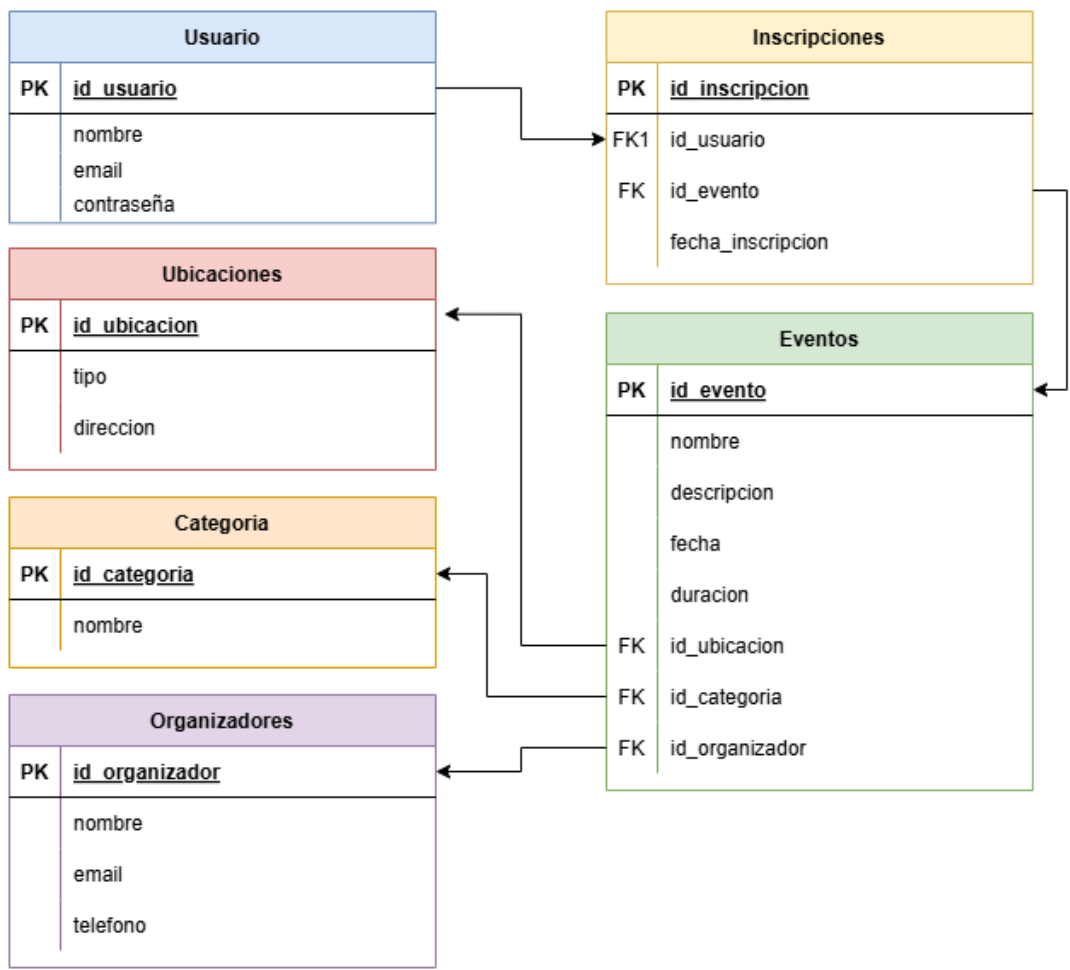
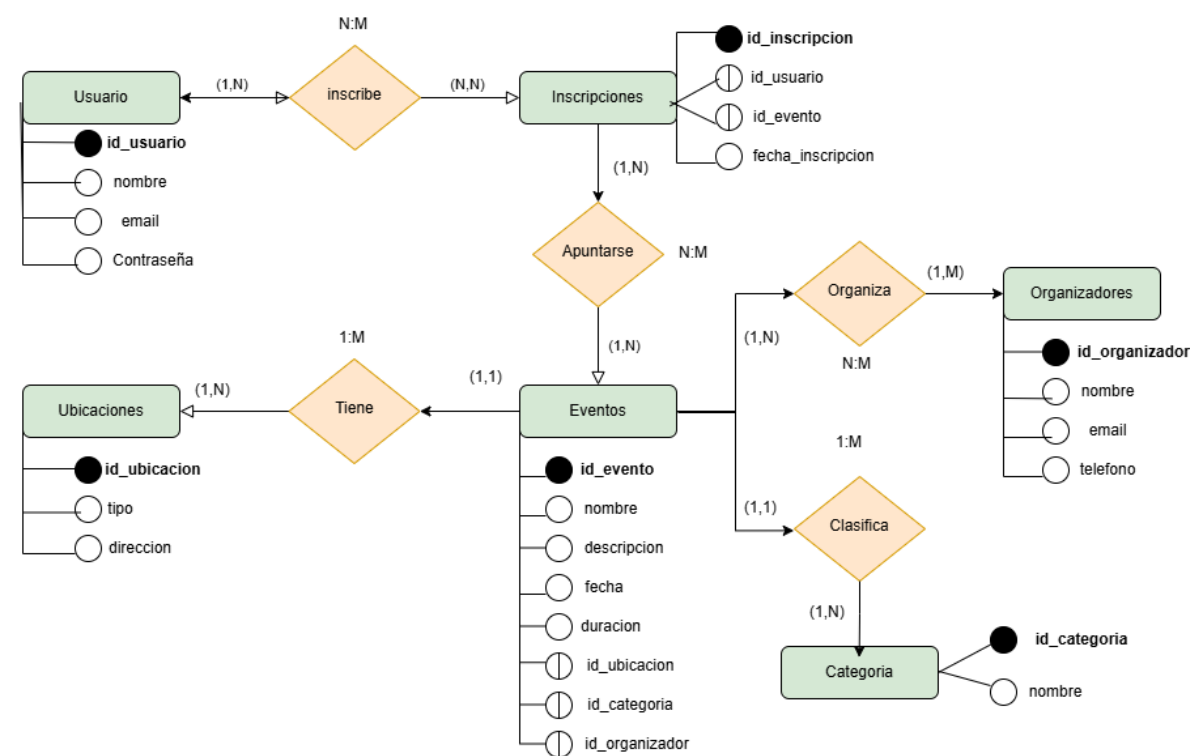


Diagramas Entidad-Relación



## DESCRIPCIÓN DE ENTIDADES

ENTIDAD: Usuario									
Atributo	Descripción	Dominio	Tipo/longitud			Opcional?		Univaluado?	
			Num	Alf	Otro	S	N	S	N
id_usuario	Código único asignado secuencialmente al usuario	Números	Numérico				X		X
nombre	Nombre del usuario	Cadena de texto	Alfanumérico				X		X
email	Correo electrónico del usuario	Cadena de texto	Alfanumérico				X		X
contraseña	Contraseña del usuario (cifrada)	Cadena de texto	Alfanumérico				X		X
<b>Observaciones y restricciones de la entidad:</b> El email debe ser único para cada usuario. La contraseña debe ser cifrada para asegurar la seguridad.									

ENTIDAD: Evento									
Atributo	Descripción	Dominio	Tipo/longitud			Opcional?		Univaluado?	
			Num	Alf	Otro	S	N	S	N
id_evento	Código único asignado	Números	Numérico				X	X	
nombre	Nombre del evento	Cadena de texto	Alfanumérico				X	X	
descripcion	Descripción del evento	Texto	Alfanumérico				X	X	
fecha	Fecha del evento	Fecha válida	Otro				X	X	
duracion	Duración del evento en minutos	Números	Numérico				X	X	
id_ubicacion	Código de la ubicación asociada	Números	Numérico				X	X	
id_categoria	Código de la categoría asociada	Números	Numérico				X	X	
id_organizador	Código del organizador asociado	Números	Numérico				X	X	
<b>Observaciones y restricciones de la entidad:</b> La fecha debe ser válida. La duración del evento debe ser en minutos.									

ENTIDAD: Organizadores									
Atributo	Descripción	Dominio	Tipo/longitud			Opcional?		Univaluado?	
			Num	Alf	Otro	S	N	S	N
id_organizador	Código único asignado	Números	Numérico				X	X	
nombre	Nombre del organizador	Cadena de texto	Alfanumérico				X	X	
email	Correo electrónico	Cadena de texto	Alfanumérico				X	X	
telefono	Número de teléfono	Cadena de números	Numérico				X	X	
<b>Observaciones y restricciones de la entidad:</b> El email debe ser único para cada organizador.									

ENTIDAD: Inscripciones									
Atributo	Descripción	Dominio	Tipo/longitud			Opcional?		Univaluado?	
			Num	Alf	Otro	S	N	S	N
id_inscripcion	Código único asignado	Números	Numérico				X	X	
id_usuario	Código del usuario asociado	Números	Numérico				X	X	
id_evento	Código del evento asociado	Números	Numérico				X	X	
fecha_inscripcion	Fecha de inscripción	Fecha válida	Otro				X	X	
<b>Observaciones y restricciones de la entidad:</b> La fecha de inscripción debe ser válida.									

ENTIDAD: Categorías									
Atributo	Descripción	Dominio	Tipo/longitud			Opcional?		Univaluado?	
			Num	Alf	Otro	S	N	S	N
id_categoria	Código único asignado	Números	Numérico				X	X	
nombre	Nombre de la categoría	Cadena de texto	Alfanumérico				X	X	
<b>Observaciones y restricciones de la entidad:</b> El nombre de la categoría debe ser único.									

ENTIDAD: Ubicaciones									
Atributo	Descripción	Dominio	Tipo/longitud			Opcional?		Univaluado?	
			Num	Alf	Otro	S	N	S	N
id_ubicacion	Código único asignado	Números	Numérico				X	X	
tipo	Tipo de ubicación (Online o Presencial)	Enumeración	Otro				X	X	
direccion	Dirección completa	Cadena de texto y números	Alfanumérico			X		X	
<b>Observaciones y restricciones de la entidad:</b> La dirección es opcional solo si el tipo es "Online".									

### Código Usado para Oracle SQL Developer:

-- Crear secuencias para auto-incremento

CREATE SEQUENCE usuarios\_seq START WITH 1 INCREMENT BY 1;

CREATE SEQUENCE organizadores\_seq START WITH 1 INCREMENT BY 1;

CREATE SEQUENCE categorias\_seq START WITH 1 INCREMENT BY 1;

CREATE SEQUENCE ubicaciones\_seq START WITH 1 INCREMENT BY 1;

CREATE SEQUENCE eventos\_seq START WITH 1 INCREMENT BY 1;

CREATE SEQUENCE inscripciones\_seq START WITH 1 INCREMENT BY 1;

-- Tabla de Usuarios

```
CREATE TABLE Usuarios (
    id_usuario NUMBER PRIMARY KEY,
    nombre VARCHAR2(100) NOT NULL,
    email VARCHAR2(100) UNIQUE NOT NULL,
    contrasena VARCHAR2(255) NOT NULL
);
```

-- Trigger para auto-incremento en Usuarios

```
CREATE OR REPLACE TRIGGER trg_usuarios_ai
BEFORE INSERT ON Usuarios
```

```
FOR EACH ROW
```

```
BEGIN
```

```
    SELECT usuarios_seq.NEXTVAL INTO :NEW.id_usuario FROM dual;
```

```
END;
```

```
/
```

```
-- Tabla de Organizadores
```

```
CREATE TABLE Organizadores (
```

```
    id_organizador NUMBER PRIMARY KEY,
```

```
    nombre VARCHAR2(100) NOT NULL,
```

```
    email VARCHAR2(100) UNIQUE NOT NULL,
```

```
    telefono VARCHAR2(20)
```

```
);
```

```
CREATE OR REPLACE TRIGGER trg_organizadores_ai
```

```
BEFORE INSERT ON Organizadores
```

```
FOR EACH ROW
```

```
BEGIN
```

```
    SELECT organizadores_seq.NEXTVAL INTO :NEW.id_organizador FROM dual;
```

```
END;
```

```
/
```

```
-- Tabla de Categorías
```

```
CREATE TABLE Categorías (
```

```
    id_categoria NUMBER PRIMARY KEY,
```

```
    nombre VARCHAR2(50) NOT NULL
```

```
);
```

```
CREATE OR REPLACE TRIGGER trg_categorias_ai
```

```

BEFORE INSERT ON Categorías
FOR EACH ROW
BEGIN
    SELECT categorias_seq.NEXTVAL INTO :NEW.id_categoria FROM dual;
END;
/

-- Tabla de Ubicaciones
CREATE TABLE Ubicaciones (
    id_ubicacion NUMBER PRIMARY KEY,
    tipo CHAR(10) CHECK (tipo IN ('Online', 'Presencial')) NOT NULL,
    direccion VARCHAR2(255)
);

CREATE OR REPLACE TRIGGER trg_ubicaciones_ai
BEFORE INSERT ON Ubicaciones
FOR EACH ROW
BEGIN
    SELECT ubicaciones_seq.NEXTVAL INTO :NEW.id_ubicacion FROM dual;
END;
/

-- Tabla de Eventos
CREATE TABLE Eventos (
    id_evento NUMBER PRIMARY KEY,
    nombre VARCHAR2(100) NOT NULL,
    descripcion CLOB,
    fecha DATE NOT NULL,
    duracion NUMBER NOT NULL,

```

```
id_ubicacion NUMBER,  
id_categoria NUMBER,  
id_organizador NUMBER,  
FOREIGN KEY (id_ubicacion) REFERENCES Ubicaciones(id_ubicacion),  
FOREIGN KEY (id_categoria) REFERENCES Categorias(id_categoria),  
FOREIGN KEY (id_organizador) REFERENCES Organizadores(id_organizador)  
);
```

```
CREATE OR REPLACE TRIGGER trg_eventos_ai  
BEFORE INSERT ON Eventos  
FOR EACH ROW  
BEGIN  
    SELECT eventos_seq.NEXTVAL INTO :NEW.id_evento FROM dual;  
END;  
/
```

-- Tabla de Inscripciones

```
CREATE TABLE Inscripciones (  
    id_inscripcion NUMBER PRIMARY KEY,  
    id_usuario NUMBER,  
    id_evento NUMBER,  
    fecha_inscripcion DATE DEFAULT SYSDATE,  
    FOREIGN KEY (id_usuario) REFERENCES Usuarios(id_usuario),  
    FOREIGN KEY (id_evento) REFERENCES Eventos(id_evento)  
);
```

```
CREATE OR REPLACE TRIGGER trg_inscripciones_ai  
BEFORE INSERT ON Inscripciones  
FOR EACH ROW
```

BEGIN

SELECT inscripciones\_seq.NEXTVAL INTO :NEW.id\_inscripcion FROM dual;

END;

/

--prueba insertando datos aleatorios

-- Insertar Usuarios

INSERT INTO Usuarios (id\_usuario, nombre, email, contrasena) VALUES  
(usuarios\_seq.NEXTVAL, 'Juan Pérez', 'juan@example.com', 'pass123');

INSERT INTO Usuarios (id\_usuario, nombre, email, contrasena) VALUES  
(usuarios\_seq.NEXTVAL, 'María López', 'maria@example.com', 'secure456');

INSERT INTO Usuarios (id\_usuario, nombre, email, contrasena) VALUES  
(usuarios\_seq.NEXTVAL, 'Carlos Sánchez', 'carlos@example.com', 'clave789');

-- Insertar Organizadores

INSERT INTO Organizadores (id\_organizador, nombre, email, telefono) VALUES  
(organizadores\_seq.NEXTVAL, 'TechCorp', 'contacto@techcorp.com', '600123456');

INSERT INTO Organizadores (id\_organizador, nombre, email, telefono) VALUES  
(organizadores\_seq.NEXTVAL, 'GreenLife', 'info@greenlife.com', '650987654');

-- Insertar Categorías

INSERT INTO Categorías (id\_categoria, nombre) VALUES (categorias\_seq.NEXTVAL,  
'Tecnología');

INSERT INTO Categorías (id\_categoria, nombre) VALUES (categorias\_seq.NEXTVAL,  
'Sostenibilidad');

INSERT INTO Categorías (id\_categoria, nombre) VALUES (categorias\_seq.NEXTVAL,  
'Innovación');

-- Insertar Ubicaciones

INSERT INTO Ubicaciones (id\_ubicacion, tipo, direccion) VALUES  
(ubicaciones\_seq.NEXTVAL, 'Presencial', 'Centro de Convenciones, Madrid');

INSERT INTO Ubicaciones (id\_ubicacion, tipo, direccion) VALUES  
(ubicaciones\_seq.NEXTVAL, 'Online', NULL);



```
INSERT INTO Ubicaciones (id_ubicacion, tipo, direccion) VALUES  
(ubicaciones_seq.NEXTVAL, 'Presencial', 'Parque Ecológico, Barcelona');
```

-- Insertar Eventos

```
INSERT INTO Eventos (id_evento, nombre, descripcion, fecha, duracion, id_ubicacion,  
id_categoria, id_organizador)
```

```
VALUES (eventos_seq.NEXTVAL, 'Conferencia IA 2025', 'Evento sobre inteligencia  
artificial y aprendizaje automático', TO_DATE('2025-05-20', 'YYYY-MM-DD'), 120, 1, 1,  
1);
```

```
INSERT INTO Eventos (id_evento, nombre, descripcion, fecha, duracion, id_ubicacion,  
id_categoria, id_organizador)
```

```
VALUES (eventos_seq.NEXTVAL, 'Taller de Energía Renovable', 'Aprende sobre energías  
limpias y su aplicación práctica', TO_DATE('2025-06-15', 'YYYY-MM-DD'), 90, 2, 2, 2);
```

```
INSERT INTO Eventos (id_evento, nombre, descripcion, fecha, duracion, id_ubicacion,  
id_categoria, id_organizador)
```

```
VALUES (eventos_seq.NEXTVAL, 'Feria de Startups', 'Exhibición de nuevas startups en  
tecnología', TO_DATE('2025-07-10', 'YYYY-MM-DD'), 180, 3, 3, 1);
```

-- Insertar Inscripciones

```
INSERT INTO Inscripciones (id_inscripcion, id_usuario, id_evento, fecha_inscripcion)
```

```
VALUES (inscripciones_seq.NEXTVAL, 1, 1, SYSDATE);
```

```
INSERT INTO Inscripciones (id_inscripcion, id_usuario, id_evento, fecha_inscripcion)
```

```
VALUES (inscripciones_seq.NEXTVAL, 2, 2, SYSDATE);
```

```
INSERT INTO Inscripciones (id_inscripcion, id_usuario, id_evento, fecha_inscripcion)
```

```
VALUES (inscripciones_seq.NEXTVAL, 3, 3, SYSDATE);
```

--Ver datos de las tablas

```
SELECT * FROM Usuarios;
```

```
SELECT * FROM Organizadores;
```

```
SELECT * FROM Categorías;
```

```
SELECT * FROM Ubicaciones;
```

```
SELECT * FROM Eventos;
```

```
SELECT * FROM Inscripciones;
```