Diseño de Base de Datos - TaskManager

**Fecha:**

**Autor:**

**Versión:** 1.0

# 1. Introducción

# Este documento describe el modelo de base de datos relacional para la aplicación TaskManager, enfocada en la gestión de tareas, listas, usuarios, etiquetas y colaboración. Se detallan las entidades principales, relaciones (incluyendo N:M), restricciones y el diccionario de datos.

# Objetivos del diseño: garantizar integridad referencial, evitar duplicidades (emails, asociaciones N:M), facilitar consultas con índices adecuados y permitir escalabilidad.

# 2. Diagrama Entidad-Relación (ERD)

## 2.1. Entidades y sus Atributos

1. **Usuarios**

* id\_usuario (PK)
* nombre
* email (único)
* fecha\_registro

1. **Listas**

* id\_lista (PK)
* nombre\_lista
* descripcion
* id\_usuario (FK, referencia a Usuarios)

1. **Tareas**

* id\_tarea (PK)
* titulo
* descripcion
* fecha\_creacion
* fecha\_limite
* id\_lista (FK, referencia a Listas)

1. **Etiquetas**

* id\_etiqueta (PK)
* nombre\_etiqueta

1. **Colaboradores (relación N:M entre Usuarios y Listas)**

* id\_usuario (FK, referencia a Usuarios)
* id\_lista (FK, referencia a Listas)

1. **Tareas\_Etiquetas (relación N:M entre Tareas y Etiquetas)**

* id\_tarea (FK, referencia a Tareas)
* id\_etiqueta (FK, referencia a Etiquetas)

## 2.2. Relaciones

* Un **usuario** puede tener muchas **listas**, pero cada **lista** tiene un único **usuario propietario**. Esto se refleja en la relación entre Usuarios y Listas.
* Una **lista** puede tener múltiples **colaboradores**, y un **colaborador** es un **usuario** que tiene acceso a esa lista. La relación entre Usuarios y Listas es muchos a muchos, lo que justifica la tabla **Colaboradores**.
* Una **tarea** puede tener múltiples **etiquetas**, y una **etiqueta** puede estar asociada a varias **tareas**. La relación entre **Tareas** y **Etiquetas** es muchos a muchos, lo que se refleja en la tabla **Tareas\_Etiquetas**.

## 2.3. Reglas y Restricciones

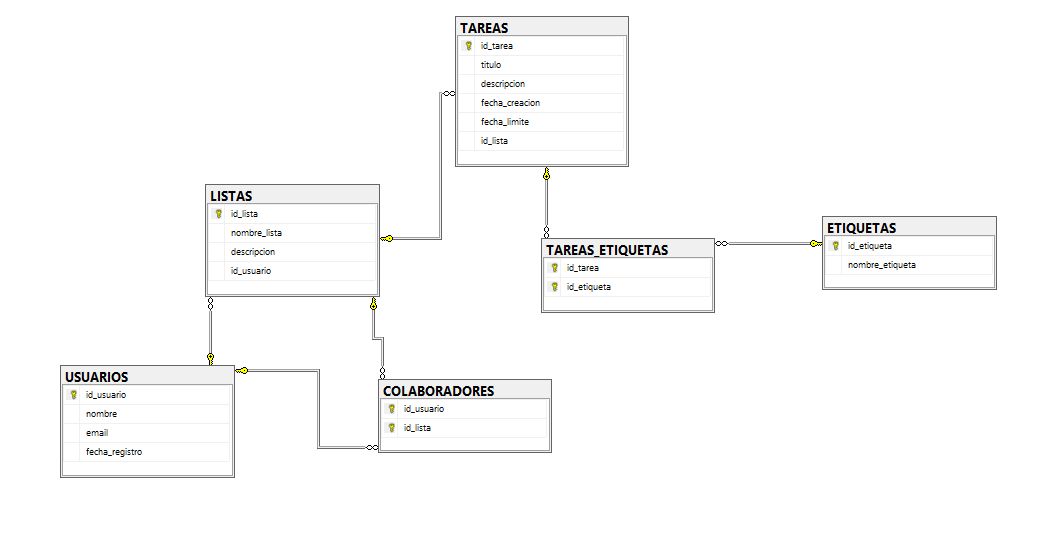
* El email de un **usuario** debe ser único.
* Un **usuario** puede ser colaborador en muchas listas, pero no puede ser un colaborador duplicado en la misma lista.

## 2.4. Diagrama de Base de Datos (E/R)

* **Usuarios** (1) -> (N) **Listas** (relación de propiedad).
* **Listas** (N) -> (N) **Usuarios** (relación de colaboración a través de la tabla **Colaboradores**).
* **Listas** (1) -> (N) **Tareas** (relación de pertenencia).
* **Tareas** (N) -> (N) **Etiquetas** (relación a través de la tabla **Tareas\_Etiquetas**).

erDiagram  
 USUARIOS ||--o{ LISTAS : "propietario"  
 USUARIOS ||--o{ COLABORADORES : "colabora"  
 LISTAS ||--o{ COLABORADORES : "tiene"  
 LISTAS ||--o{ TAREAS : "contiene"  
 TAREAS ||--o{ TAREAS\_ETIQUETAS : "usa"  
 ETIQUETAS ||--o{ TAREAS\_ETIQUETAS : "clasifica"  
  
 USUARIOS {  
 INT id\_usuario PK  
 VARCHAR nombre  
 VARCHAR email UNIQUE  
 DATETIME fecha\_registro  
 }  
 LISTAS {  
 INT id\_lista PK  
 VARCHAR nombre\_lista  
 TEXT descripcion  
 INT id\_usuario FK  
 }  
 TAREAS {  
 INT id\_tarea PK  
 VARCHAR titulo  
 TEXT descripcion  
 DATETIME fecha\_creacion  
 DATETIME fecha\_limite  
 INT id\_lista FK  
 }  
 ETIQUETAS {  
 INT id\_etiqueta PK  
 VARCHAR nombre\_etiqueta  
 }  
 COLABORADORES {  
 INT id\_usuario FK  
 INT id\_lista FK  
 PK "id\_usuario,id\_lista"  
 }  
 TAREAS\_ETIQUETAS {  
 INT id\_tarea FK  
 INT id\_etiqueta FK  
 PK "id\_tarea,id\_etiqueta"  
 }

# 3. Diagrama Relacional



# 4. Diccionario de Datos

**Tabla Usuarios**

|  |  |  |  |
| --- | --- | --- | --- |
| Campo | Tipo de Dato | Descripción | Restricciones |
| id\_usuario | INT | Identificador único del usuario. | PRIMARY KEY, AUTO\_INCREMENT |
| nombre | VARCHAR(255) | Nombre completo del usuario. | NOT NULL |
| email | VARCHAR(255) | Dirección de correo electrónico del usuario. | UNIQUE, NOT NULL |
| fecha\_registro | DATETIME | Fecha y hora en que el usuario se registró. | NOT NULL |

**Tabla Listas**

|  |  |  |  |
| --- | --- | --- | --- |
| Campo | Tipo de Dato | Descripción | Restricciones |
| id\_lista | INT | Identificador único de la lista. | PRIMARY KEY, AUTO\_INCREMENT |
| nombre\_lista | VARCHAR(255) | Nombre de la lista. | NOT NULL |
| descripcion | TEXT | Descripción de la lista. | NULLABLE |
| id\_usuario | INT | Identificador del usuario propietario. | FOREIGN KEY (Usuarios) |

**Tabla Tareas**

|  |  |  |  |
| --- | --- | --- | --- |
| Campo | Tipo de Dato | Descripción | Restricciones |
| id\_tarea | INT | Identificador único de la tarea. | PRIMARY KEY, AUTO\_INCREMENT |
| titulo | VARCHAR(255) | Título de la tarea. | NOT NULL |
| descripcion | TEXT | Descripción detallada de la tarea. | NULLABLE |
| fecha\_creacion | DATETIME | Fecha y hora en que se creó la tarea. | NOT NULL |
| fecha\_limite | DATETIME | Fecha límite para completar la tarea. | NULLABLE |
| id\_lista | INT | Identificador de la lista a la que pertenece. | FOREIGN KEY (Listas) |

**TablaEtiquetas**

|  |  |  |  |
| --- | --- | --- | --- |
| Campo | Tipo de Dato | Descripción | Restricciones |
| id\_etiqueta | INT | Identificador único de la etiqueta. | PRIMARY KEY, AUTO\_INCREMENT |
| nombre\_etiqueta | VARCHAR(255) | Nombre de la etiqueta. | NOT NULL |

**Tabla Colaboradores**

|  |  |  |  |
| --- | --- | --- | --- |
| Campo | Tipo de Dato | Descripción | Restricciones |
| id\_usuario | INT | Identificador del usuario colaborador. | FOREIGN KEY (Usuarios) |
| id\_lista | INT | Identificador de la lista en la que colabora. | FOREIGN KEY (Listas) |
| Clave Primaria Compuesta |  |  | (id\_usuario, id\_lista) (evita duplicados) |

**Tabla Tareas\_Etiquetas**

|  |  |  |  |
| --- | --- | --- | --- |
| Campo | Tipo de Dato | Descripción | Restricciones |
| id\_tarea | INT | Identificador de la tarea. | FOREIGN KEY (Tareas) |
| id\_etiqueta | INT | Identificador de la etiqueta. | FOREIGN KEY (Etiquetas) |
| Clave Primaria Compuesta |  |  | (id\_tarea, id\_etiqueta) (relación N:M) |

## 4.1. Relaciones y Restricciones:

* **Usuarios ↔ Listas**: Un usuario puede tener muchas listas, pero cada lista pertenece a un único usuario. La relación está representada por el campo id\_usuario en la tabla Listas.
* **Listas ↔ Colaboradores**: La relación entre Usuarios y Listas es muchos a muchos. La tabla Colaboradores maneja esta relación.
* **Listas ↔ Tareas**: Una lista puede tener múltiples tareas, y una tarea pertenece a una única lista. La relación está representada por el campo id\_lista en la tabla Tareas.
* **Tareas ↔ Etiquetas**: La relación entre tareas y etiquetas es muchos a muchos, gestionada por la tabla Tareas\_Etiquetas.

# 5. Consideraciones de Escalabilidad

* Índices en campos de búsqueda frecuente (email, id\_usuario, id\_lista).
* Uso de claves foráneas con ON DELETE CASCADE para mantener integridad referencial.
* Posibilidad de particionar tablas de tareas por lista en sistemas distribuidos.

# 6. Conclusiones

Este modelo relacional soporta la flexibilidad necesaria para gestionar tareas colaborativas, etiquetas personalizadas y múltiples listas por usuario. Permite escalar el sistema con crecimiento de usuarios y tareas sin comprometer la integridad de los datos.

# Anexo A. DDL SQL (MySQL 8.0, InnoDB, utf8mb4)

-- USUARIOS  
CREATE TABLE usuarios (  
 id\_usuario INT UNSIGNED NOT NULL AUTO\_INCREMENT,  
 nombre VARCHAR(255) NOT NULL,  
 email VARCHAR(255) NOT NULL,  
 fecha\_registro DATETIME NOT NULL DEFAULT CURRENT\_TIMESTAMP,  
 CONSTRAINT pk\_usuarios PRIMARY KEY (id\_usuario),  
 CONSTRAINT uq\_usuarios\_email UNIQUE (email)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci;  
  
-- LISTAS  
CREATE TABLE listas (  
 id\_lista INT UNSIGNED NOT NULL AUTO\_INCREMENT,  
 nombre\_lista VARCHAR(255) NOT NULL,  
 descripcion TEXT NULL,  
 id\_usuario INT UNSIGNED NOT NULL,  
 CONSTRAINT pk\_listas PRIMARY KEY (id\_lista),  
 CONSTRAINT fk\_listas\_usuarios  
 FOREIGN KEY (id\_usuario) REFERENCES usuarios(id\_usuario)  
 ON DELETE CASCADE  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci;  
  
-- TAREAS  
CREATE TABLE tareas (  
 id\_tarea INT UNSIGNED NOT NULL AUTO\_INCREMENT,  
 titulo VARCHAR(255) NOT NULL,  
 descripcion TEXT NULL,  
 fecha\_creacion DATETIME NOT NULL DEFAULT CURRENT\_TIMESTAMP,  
 fecha\_limite DATETIME NULL,  
 id\_lista INT UNSIGNED NOT NULL,  
 CONSTRAINT pk\_tareas PRIMARY KEY (id\_tarea),  
 CONSTRAINT fk\_tareas\_listas  
 FOREIGN KEY (id\_lista) REFERENCES listas(id\_lista)  
 ON DELETE CASCADE,  
 CONSTRAINT ck\_tareas\_fechas  
 CHECK (fecha\_limite IS NULL OR fecha\_limite >= fecha\_creacion)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci;  
  
CREATE INDEX ix\_tareas\_lista ON tareas (id\_lista);  
  
-- ETIQUETAS  
CREATE TABLE etiquetas (  
 id\_etiqueta INT UNSIGNED NOT NULL AUTO\_INCREMENT,  
 nombre\_etiqueta VARCHAR(100) NOT NULL,  
 CONSTRAINT pk\_etiquetas PRIMARY KEY (id\_etiqueta)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci;  
  
-- COLABORADORES  
CREATE TABLE colaboradores (  
 id\_usuario INT UNSIGNED NOT NULL,  
 id\_lista INT UNSIGNED NOT NULL,  
 CONSTRAINT pk\_colaboradores PRIMARY KEY (id\_usuario, id\_lista),  
 CONSTRAINT fk\_colab\_usuario  
 FOREIGN KEY (id\_usuario) REFERENCES usuarios(id\_usuario)  
 ON DELETE CASCADE,  
 CONSTRAINT fk\_colab\_lista  
 FOREIGN KEY (id\_lista) REFERENCES listas(id\_lista)  
 ON DELETE CASCADE  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci;  
  
CREATE INDEX ix\_colab\_lista ON colaboradores (id\_lista);  
  
-- TAREAS\_ETIQUETAS  
CREATE TABLE tareas\_etiquetas (  
 id\_tarea INT UNSIGNED NOT NULL,  
 id\_etiqueta INT UNSIGNED NOT NULL,  
 CONSTRAINT pk\_tareas\_etiquetas PRIMARY KEY (id\_tarea, id\_etiqueta),  
 CONSTRAINT fk\_te\_tarea  
 FOREIGN KEY (id\_tarea) REFERENCES tareas(id\_tarea)  
 ON DELETE CASCADE,  
 CONSTRAINT fk\_te\_etiqueta  
 FOREIGN KEY (id\_etiqueta) REFERENCES etiquetas(id\_etiqueta)  
 ON DELETE CASCADE  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci;  
  
CREATE INDEX ix\_te\_etiqueta ON tareas\_etiquetas (id\_etiqueta);