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.

# 2. Diagrama Entidad-Relación (ERD)

## 2.1. Entidades y sus Atributos

**Usuarios**

* id\_usuario (PK)
* nombre\_usuario
* email (único)
* password (encriptada)
* rol
* created\_at
* updated\_at

**Turistas**

* id\_turista (PK)
* id\_usuario (FK, referencia a Usuarios)
* nombre
* apellido
* dni
* pasaporte
* nacionalidad
* fecha\_nacimiento
* genero

**Grupos**

* id\_grupo (PK)
* nombre
* descripcion

**Itinerarios**

* id\_itinerario (PK)
* id\_grupo (FK, referencia a Grupos)
* fecha\_inicio
* fecha\_fin
* estado\_presupuesto\_id (FK, referencia a Estados\_Presupuesto)
* created\_at
* updated\_at

**Itinerario\_Turistas** (relación N:M entre Itinerarios y Turistas)

* id\_itinerario (FK, referencia a Itinerarios)
* id\_turista (FK, referencia a Turistas)

**Estados\_Presupuesto**

* id\_estado (PK)
* nombre\_estado

**Programas**

* id\_programa (PK)
* nombre
* descripcion
* tipo
* duracion
* costo

**Itinerario\_Programas** (relación N:M entre Itinerarios y Programas)

* id\_itinerario\_programa (PK)
* id\_itinerario (FK, referencia a Itinerarios)
* id\_programa (FK, referencia a Programas)
* fecha

**Detalle\_Machu\_Itinerario**

* id\_itinerario\_programa (FK, referencia a Itinerario\_Programas)
* empresa\_tren
* horario\_tren\_ida
* horario\_tren\_retorno
* nombre\_guia

**Detalle\_Machupicchu**

* id\_programa (FK, referencia a Programas)
* ruta
* tiempo\_visita

## 2.2. Relaciones

* Un **usuario** puede estar asociado a un **turista** (uno a uno), ya que cada turista registrado pertenece a un usuario del sistema.
* Un **grupo** puede tener muchos **itinerarios**, pero cada itinerario pertenece a un único grupo (uno a muchos).
* Un **itinerario** puede tener múltiples **turistas**, y un turista puede participar en varios itinerarios. Esto es una relación muchos a muchos, reflejada en la tabla **Itinerario\_Turistas**.
* Un **estado de presupuesto** puede estar asociado a muchos **itinerarios**, pero cada itinerario tiene un único estado (uno a muchos).
* Un **itinerario** puede incluir múltiples **programas**, y un **programa** puede estar en varios itinerarios. Esta es una relación muchos a muchos, reflejada en la tabla **Itinerario\_Programas**.
* Un **itinerario\_programa** puede tener detalles específicos de Machu Picchu, como **empresa de tren, horarios y guía asignado** (uno a uno), reflejado en la tabla **Detalle\_Machu\_Itinerario**.
* Un **programa** de Machu Picchu puede tener información adicional como **ruta y tiempo de visita** (uno a uno), reflejado en la tabla **Detalle\_Machupicchu**.

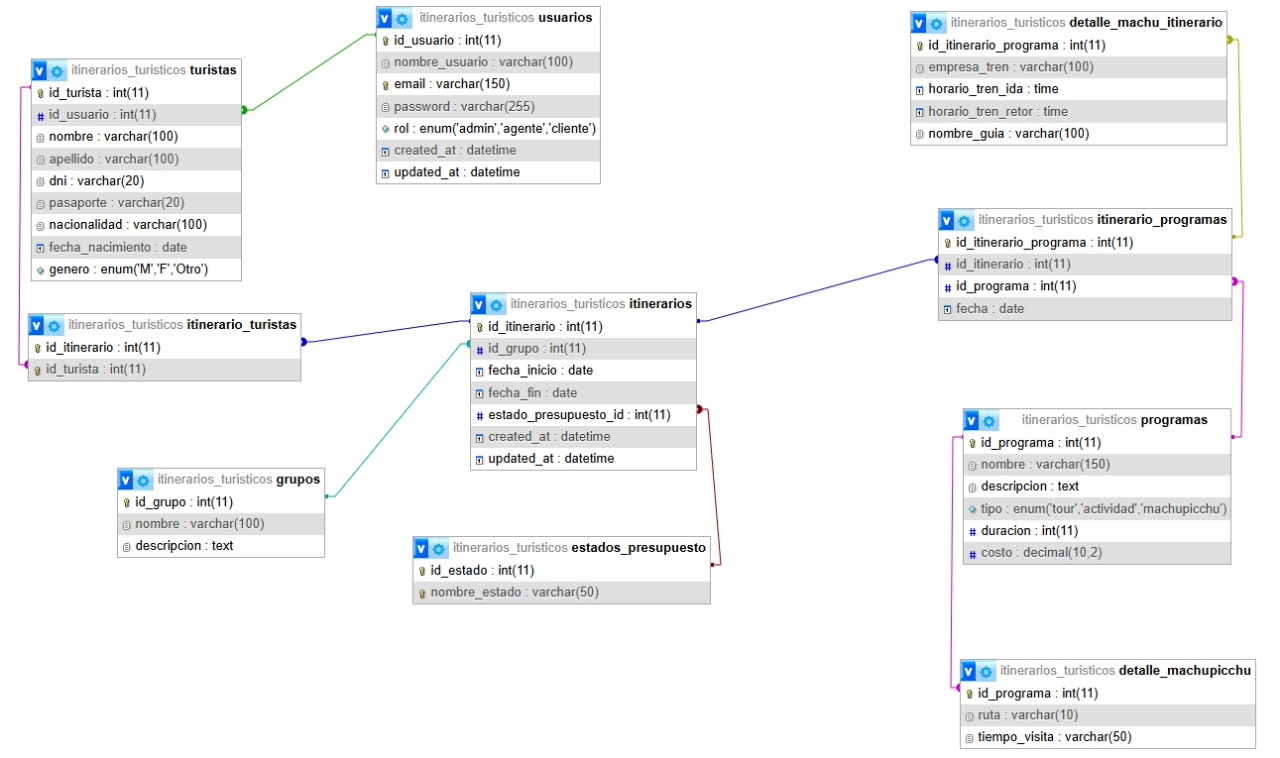
## 2.3. Reglas y Restricciones

* El **email** de un usuario debe ser único en la tabla **Usuarios**.
* Cada **turista** debe estar asociado a un único **usuario**; un usuario no puede tener dos registros de turista duplicados.
* Cada **itinerario** pertenece a un único **grupo**, pero un grupo puede tener muchos itinerarios.
* La relación **Itinerario\_Turistas** no permite duplicados; un turista no puede estar asignado al mismo itinerario más de una vez.
* La relación **Itinerario\_Programas** no permite duplicados; un programa no puede asignarse al mismo itinerario más de una vez.
* Cada **detalle de Machu Picchu** (Detalle\_Machu\_Itinerario) debe estar asociado a un **itinerario\_programa** válido; no se permiten registros huérfanos.
* Cada **detalle de Machu Picchu** (Detalle\_Machupicchu) debe estar asociado a un **programa** válido de Machu Picchu; no se permiten registros huérfanos.
* Todos los campos obligatorios, como nombres, fechas y costos, deben estar completos al crear o actualizar registros.

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

* **Usuarios (1) -> (1) Turistas** (cada turista pertenece a un usuario).
* **Grupos (1) -> (N) Itinerarios** (cada itinerario pertenece a un grupo).
* **Itinerarios (N) -> (N) Turistas** (relación a través de la tabla **Itinerario\_Turistas**).
* **Estados\_Presupuesto (1) -> (N) Itinerarios** (cada itinerario tiene un único estado).
* **Itinerarios (N) -> (N) Programas** (relación a través de la tabla **Itinerario\_Programas**).
* **Itinerario\_Programas (1) -> (1) Detalle\_Machu\_Itinerario** (detalle específico de Machu Picchu para cada itinerario-programa).
* **Programas (1) -> (1) Detalle\_Machupicchu** (información adicional de cada programa en Machu Picchu)

# 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\_usuario | VARCHAR(255) | Nombre de usuario. | NOT NULL |
| email | VARCHAR(255) | Dirección de correo electrónico del usuario. | UNIQUE, NOT NULL |
| password | VARCHAR(255) | Contraseña encriptada del usuario. | NOT NULL |
| rol | VARCHAR(50) | Rol del usuario (admin, turista, etc.). | NOT NULL |
| created\_at | DATETIME | Fecha de creación del registro. | NOT NULL |
| updated\_at | DATETIME | Fecha de última actualización del registro. | NOT NULL |

**Tabla turitas**

| Campo | Tipo de Dato | Descripción | Restricciones |
| --- | --- | --- | --- |
| id\_turista | INT | Identificador único del turista. | PRIMARY KEY, AUTO\_INCREMENT |
| id\_usuario | INT | Referencia al usuario asociado. | FOREIGN KEY REFERENCES Usuarios(id\_usuario), NOT NULL |
| nombre | VARCHAR(255) | Nombre del turista. | NOT NULL |
| apellido | VARCHAR(255) | Apellido del turista. | NOT NULL |
| dni | VARCHAR(50) | Documento Nacional de Identidad. |  |
| pasaporte | VARCHAR(50) | Número de pasaporte. |  |
| nacionalidad | VARCHAR(100) | Nacionalidad del turista. |  |
| fecha\_nacimiento | DATE | Fecha de nacimiento del turista. |  |
| genero | VARCHAR(20) | Género del turista. |  |

**Tabla Grupos**

| Campo | Tipo de Dato | Descripción | Restricciones |
| --- | --- | --- | --- |
| id\_grupo | INT | Identificador único del grupo | PRIMARY KEY, AUTO\_INCREMENT |
| nombre | VARCHAR(255) | Nombre del grupo turístico | NOT NULL |
| descripcion | TEXT | Descripción del grupo |  |

### Tabla: Itinerarios

| Campo | Tipo de Dato | Descripción | Restricciones |
| --- | --- | --- | --- |
| id\_itinerario | INT | Identificador único del itinerario | PRIMARY KEY, AUTO\_INCREMENT |
| id\_grupo | INT | Referencia al grupo del itinerario | FOREIGN KEY REFERENCES Grupos(id\_grupo), NOT NULL |
| fecha\_inicio | DATE | Fecha de inicio del itinerario | NOT NULL |
| fecha\_fin | DATE | Fecha de fin del itinerario | NOT NULL |
| estado\_presupuesto\_id | INT | Referencia al estado del presupuesto | FOREIGN KEY REFERENCES Estados\_Presupuesto(id\_estado) |
| created\_at | DATETIME | Fecha de creación del registro | NOT NULL |
| updated\_at | DATETIME | Fecha de actualización del registro | NOT NULL |

### Tabla: Estados\_Presupuesto

| Campo | Tipo de Dato | Descripción | Restricciones |
| --- | --- | --- | --- |
| id\_estado | INT | Identificador único del estado | PRIMARY KEY, AUTO\_INCREMENT |
| nombre\_estado | VARCHAR(100) | Nombre del estado | NOT NULL |

### Tabla: Itinerario\_Turistas

| Campo | Tipo de Dato | Descripción | Restricciones |
| --- | --- | --- | --- |
| id\_itinerario | INT | Referencia al itinerario | FOREIGN KEY REFERENCES Itinerarios(id\_itinerario), NOT NULL |
| id\_turista | INT | Referencia al turista | FOREIGN KEY REFERENCES Turistas(id\_turista), NOT NULL |
| PRIMARY KEY (id\_itinerario, id\_turista) |  | Llave compuesta para evitar duplicados |  |

### Tabla: Programas

| Campo | Tipo de Dato | Descripción | Restricciones |
| --- | --- | --- | --- |
| id\_programa | INT | Identificador del programa | PRIMARY KEY, AUTO\_INCREMENT |
| nombre | VARCHAR(255) | Nombre del programa | NOT NULL |
| descripcion | TEXT | Descripción del programa |  |
| tipo | VARCHAR(100) | Tipo de programa (tour, visita) | NOT NULL |
| duracion | VARCHAR(50) | Duración estimada |  |
| costo | DECIMAL(10,2) | Costo del programa |  |

### Tabla: Itinerario\_Programas

| Campo | Tipo de Dato | Descripción | Restricciones |
| --- | --- | --- | --- |
| id\_itinerario\_programa | INT | Identificador único | PRIMARY KEY, AUTO\_INCREMENT |
| id\_itinerario | INT | Referencia al itinerario | FOREIGN KEY REFERENCES Itinerarios(id\_itinerario), NOT NULL |
| id\_programa | INT | Referencia al programa | FOREIGN KEY REFERENCES Programas(id\_programa), NOT NULL |
| fecha | DATE | Fecha asignada al programa | NOT NULL |

### Tabla: Detalle\_Machu\_Itinerario

| Campo | Tipo de Dato | Descripción | Restricciones |
| --- | --- | --- | --- |
| id\_itinerario\_programa | INT | Referencia al itinerario-programa | FOREIGN KEY REFERENCES Itinerario\_Programas(id\_itinerario\_programa), PRIMARY KEY |
| empresa\_tren | VARCHAR(255) | Empresa de tren asignada | NOT NULL |
| horario\_tren\_ida | TIME | Hora de salida del tren | NOT NULL |
| horario\_tren\_retorno | TIME | Hora de regreso del tren | NOT NULL |
| nombre\_guia | VARCHAR(255) | Nombre del guía de Machu Picchu | NOT NULL |

### Tabla: Detalle\_Machupicchu

| Campo | Tipo de Dato | Descripción | Restricciones |
| --- | --- | --- | --- |
| id\_programa | INT | Referencia al programa | FOREIGN KEY REFERENCES Programas(id\_programa), PRIMARY KEY |
| ruta | VARCHAR(255) | Ruta del recorrido | NOT NULL |
| tiempo\_visita | TIME | Tiempo estimado de la visita | NOT NULL |

## 4.1. Relaciones y Restricciones:

* **Usuarios ↔ Turistas:** Cada usuario puede estar asociado a un único turista, y cada turista pertenece a un único usuario. La relación está representada por el campo **id\_usuario** en la tabla **Turistas** (uno a uno).
* **Grupos ↔ Itinerarios:** Un grupo puede tener muchos itinerarios, pero cada itinerario pertenece a un único grupo. La relación está representada por el campo **id\_grupo** en la tabla **Itinerarios** (uno a muchos).
* **Itinerarios ↔ Turistas:** Un itinerario puede incluir múltiples turistas, y un turista puede participar en varios itinerarios. Esta relación muchos a muchos está gestionada por la tabla **Itinerario\_Turistas**.
* **Estados\_Presupuesto ↔ Itinerarios:** Cada itinerario tiene un único estado de presupuesto, pero un estado puede estar asociado a muchos itinerarios. La relación está representada por **estado\_presupuesto\_id** en la tabla **Itinerarios** (uno a muchos).
* **Itinerarios ↔ Programas:** Un itinerario puede incluir múltiples programas, y un programa puede pertenecer a varios itinerarios. Esta relación muchos a muchos está gestionada por la tabla **Itinerario\_Programas**.
* **Itinerario\_Programas ↔ Detalle\_Machu\_Itinerario:** Cada registro de itinerario-programa puede tener un detalle específico de Machu Picchu. La relación es uno a uno, representada por **id\_itinerario\_programa** en la tabla **Detalle\_Machu\_Itinerario**.
* **Programas ↔ Detalle\_Machupicchu:** Cada programa de Machu Picchu puede tener un detalle adicional, como ruta y tiempo de visita. La relación es uno a uno, representada por **id\_programa** en la tabla **Detalle\_Machupicchu**.
* **Restricciones principales:**
  + Ningún registro puede estar duplicado en las tablas de relación muchos a muchos (**Itinerario\_Turistas** e **Itinerario\_Programas**).
  + Todos los registros de detalle deben referenciar registros válidos en sus tablas padre (**Detalle\_Machu\_Itinerario** y **Detalle\_Machupicchu**).

# 5. Consideraciones de Escalabilidad

* Crear **índices** en campos de búsqueda frecuente, como **email**, **id\_usuario**, **id\_turista**, **id\_itinerario**, para mejorar el rendimiento de consultas.
* Uso de **claves foráneas con ON DELETE CASCADE** para mantener la **integridad referencial** al eliminar registros relacionados (por ejemplo, al borrar un itinerario, se eliminan automáticamente sus programas asociados).
* Posibilidad de **particionar tablas grandes**, como **Itinerario\_Turistas** o **Itinerario\_Programas**, en sistemas distribuidos para soportar gran volumen de datos.
* Separación de servicios y base de datos por módulos (usuarios, itinerarios, programas) para facilitar escalabilidad horizontal en el backend.

# 6. Conclusiones

* Este modelo relacional permite **gestionar turistas, grupos, itinerarios y programas** de forma organizada y segura.
* Soporta **múltiples turistas por itinerario, múltiples programas por itinerario** y detalles específicos de Machu Picchu sin comprometer la integridad de los datos.
* Permite **escalar el sistema** con el crecimiento de usuarios, turistas y programas, manteniendo eficiencia en consultas y operaciones.
* La arquitectura propuesta ofrece **flexibilidad para futuras integraciones**, como reservas, pagos y aplicación móvil.