

# Documentación Técnica - TallerFinalWebApi

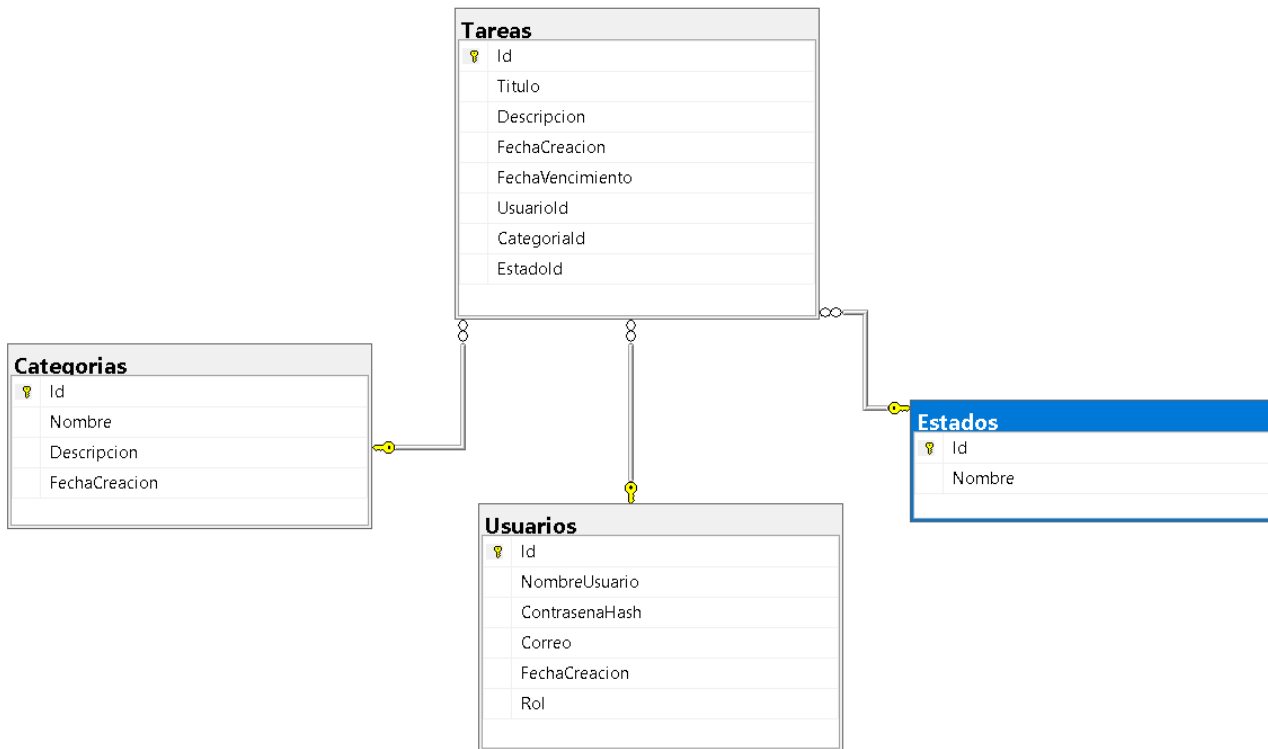
## 1. Estructura del Proyecto

```
TallerFinalWebApi/  
|-- TodoAppApi/  
|   |-- Controllers/  
|       |-- AccountController.cs  
|       |-- AdminController.cs  
|       |-- CategoriaController.cs  
|       |-- EstadoController.cs  
|       |-- TareaController.cs  
|   |-- Models/  
|       |-- Usuario.cs  
|       |-- Categoria.cs  
|       |-- Estado.cs  
|       |-- Tarea.cs  
|   |-- Data/  
|       |-- ApplicationDbContext.cs  
|   |-- Migrations/
```

## 2. Diagrama ER (Modelo Relacional de la Base de Datos)

A continuación se presenta el diagrama relacional de la base de datos utilizado en el sistema:

## Documentación Técnica - TallerFinalWebApi



### 3. Endpoints y Uso

AccountController:

- POST /api/account/regar : Registro de usuario.
- POST /api/account/login : Inicio de sesión.
- POST /api/account/promover : Promueve un usuario a rol admin.

AdminController:

- GET /api/admin/lista-usuarios : Listado de usuarios.
- DELETE /api/admin/eliminar/{id} : Elimina un usuario.

CategoriaController:

- GET /api/categoria : Lista categorías.
- POST /api/categoria : Crea una categoría.
- PUT /api/categoria/{id} : Actualiza categoría.
- DELETE /api/categoria/{id} : Elimina categoría.

EstadoController:

- GET /api/estado : Lista estados.
- POST /api/estado : Crea un estado.
- PUT /api/estado/{id} : Actualiza estado.
- DELETE /api/estado/{id} : Elimina estado.

TareaController:

- GET /api/tarea : Lista tareas.
- GET /api/tarea/{id} : Obtiene tarea por ID.

## Documentación Técnica - TallerFinalWebApi

- POST /api/tarea : Crea una tarea.
- PUT /api/tarea/{id} : Actualiza tarea.
- DELETE /api/tarea/{id} : Elimina tarea.