Proyecto

Consruir el Backend

Constuir el proecto con Node.js

Instalar las dependencias necesarias

Configurar la conexioc MySQL

Definir el modelo de tareas.

Crear turas API CRUD Create Read Update Delete

Probar con postman

Instalar mysql

<https://dev.mysql.com/downloads/file/?id=536788>

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

El contenido generado por IA puede ser incorrecto.

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

El contenido generado por IA puede ser incorrecto.

Mysql -u root -p

Contrasena para mysql: GRaudifo2736!

Crear una carpeta para el backend

Se instala express mysql2 cors y dotenv

Texto

El contenido generado por IA puede ser incorrecto.



“npm install express mysql2 cors dotenv”

Express Es un framework que sirve para crear el servidor web.

Mysql2 Es para conectar Node.js con MySQL.

Cors Es para permitir particiones desde el frontend

Dotenv Es para manejar variables de entorno credenciales para la base de datos.

Nodemon para reiniciar automáticamente el servidor cada vez que hagamos cambios.

Interfaz de usuario gráfica, Texto

El contenido generado por IA puede ser incorrecto.

“npm install – g nodemon”

Se crea un servidor a un archivo que llamaremos server.js

/////

const express = require("express");

const cors = require("cors");

const mysql = require("mysql2");

require("dotenv").config();

const app = express();

app.use(cors());

app.use(express.json()); // Para leer JSON en las peticiones

// Configuración de la base de datos

const db = mysql.createConnection({

host: process.env.DB\_HOST || "localhost",

user: process.env.DB\_USER || "root",

password: process.env.DB\_PASS || "",

database: process.env.DB\_NAME || "tareas\_db",

});

// Conectar a la base de datos

db.connect((err) => {

if (err) {

console.error("Error al conectar a la base de datos:", err);

} else {

console.log("Conectado a la base de datos MySQL ");

}

});

// Ruta de prueba

app.get("/", (req, res) => {

res.send("Servidor funcionando correctamente");

});

// Iniciar el servidor

const PORT = process.env.PORT || 5000;

app.listen(PORT, () => {

console.log(`Servidor corriendo en http://localhost:${PORT}`);

});

////

Empezar a correr el servidor

cd C:\Users\joabr\OneDrive\Escritorio\Proyecto GRAVEN\backend

 npx nodemon server.js

Texto

El contenido generado por IA puede ser incorrecto.

SQL

Texto

El contenido generado por IA puede ser incorrecto.

SHOW DATABASES;

USE tareas\_db;

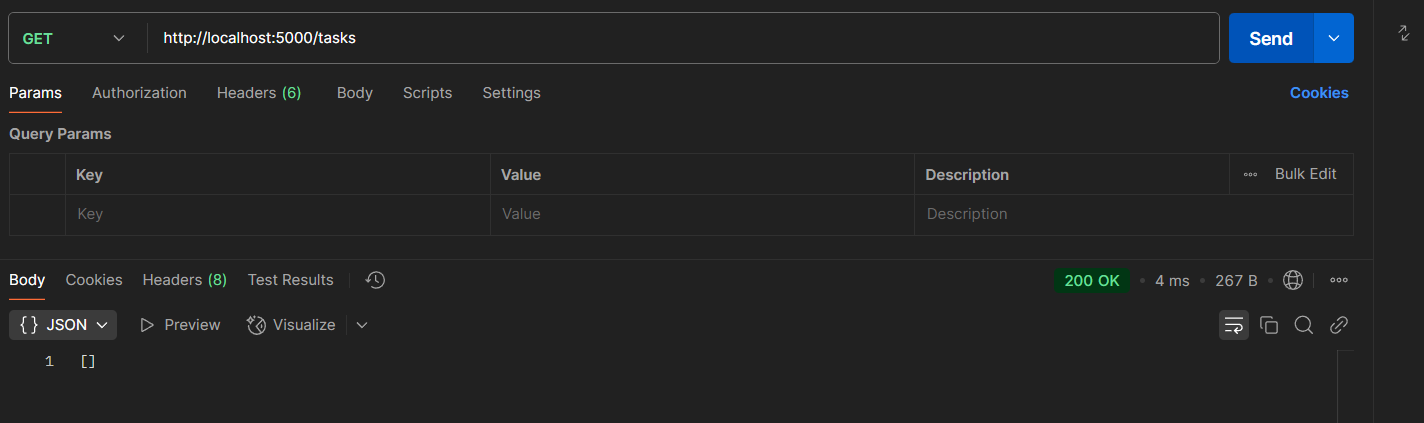
SHOW TABLES;

Select \* from tasks;

Texto

El contenido generado por IA puede ser incorrecto.

GET http://localhost:5000/tasks



Captura de pantalla de un teléfono celular

El contenido generado por IA puede ser incorrecto.

Captura de pantalla de un teléfono celular

El contenido generado por IA puede ser incorrecto.

Captura de pantalla de un teléfono celular

El contenido generado por IA puede ser incorrecto.

Captura de pantalla de un teléfono celular

El contenido generado por IA puede ser incorrecto.

Postman Backend

Obtener todas las tareas usando el método GET

URL: <http://localhost:5000/tasks>

Recupera todas las tareas de la base de datos.

Captura de pantalla de un celular

El contenido generado por IA puede ser incorrecto.

Agrega una nueva tarea POST

URL: <http://localhost:5000/tasks>

Encabezados:

Content-Type: application/json

Cuerbo Body:

{

“name”: “Ir al cine”

}

Captura de pantalla de un teléfono celular

El contenido generado por IA puede ser incorrecto.

Actualizar el estado de una tarea.

Método PUT

URL: <http://localhost:5000/tasks/:id>

Encabezados: 0 no completada, 1 completada.

Content-Type: application/json

{

“status”: 1

}

Captura de pantalla de un teléfono celular

El contenido generado por IA puede ser incorrecto.

Eliminar una tarea DELETE

URL: <http://localhost:5000/tasks/:id>

Reemplaza: id por el ID de la tarea que deseas eliminar.

Captura de pantalla de un teléfono celular

El contenido generado por IA puede ser incorrecto.