Guía Paso a Paso del Proyecto de Microservicios con Node.js, MSSQL y Docker

# 1. Requisitos Previos

- Docker Desktop con soporte para contenedores Linux  
- Editor de código como Visual Studio Code (opcional)  
- Conexión a Internet para descargar imágenes de Docker

# 2. Estructura del Proyecto

Este proyecto incluye:  
- usuarios-service: autenticación con JWT  
- tickets-service: conexión a SQL Server para registrar tickets  
- frontend-service: formulario HTML para mostrar y agregar tickets  
- mssql: contenedor con SQL Server y script automático

# 3. Script de Inicialización Automática

El archivo `mssql/init.sql`:  
- Crea la base de datos `SoporteDB`  
- Crea la tabla `Tickets`  
- Agrega el usuario `marsolca` con contraseña `Sl5wret6\*`  
- Le da permisos `db\_owner` sobre la base

# 4. Ejecutar el Proyecto

1. Abrir terminal en la raíz del proyecto  
2. Ejecutar: `docker-compose up --build`  
3. Esperar a que aparezca "SQL Server is now ready for client connections" en los logs

# 5. Probar el Microservicio de Tickets

Usa Postman para probar los endpoints:  
- GET http://localhost:3002/tickets  
- POST http://localhost:3002/tickets  
  
Body JSON para POST:  
{  
 "titulo": "No funciona el botón",  
 "descripcion": "El botón enviar no hace nada",  
 "estado": "Pendiente"  
}

# 6. Acceder a SQL Server Manualmente

1. Conéctate al contenedor con:  
`docker exec -it mssql /opt/mssql-tools/bin/sqlcmd -S localhost -U sa -P YourStrongPassword1`  
2. Ejecuta:  
`SELECT name FROM sys.databases;`  
`USE SoporteDB; SELECT \* FROM Tickets;`

# 7. Reinicio y Solución de Problemas

- Si el servicio de tickets da error al iniciar, puede deberse a que la base aún no está lista.  
- Reinicia con: `docker restart microservicios-ticketes-tickets-service-1`  
- Consulta logs con: `docker logs mssql`

# 8. Finalizar el Proyecto

Presiona Ctrl+C para detener los servicios o ejecuta `docker-compose down -v` para eliminarlos completamente.