

TALLER SEMANAL — Plataforma Multi-Motor de Bases de Datos

Ruta Avanzada .NET + React

Duración: 5 días

Modalidad: Trabajo por células

Entregable final: Proyecto desplegado en un VPS real (servidor de Andrés)

1. Descripción General del Taller

En este taller desarrollarás una plataforma web profesional que permita **gestionar múltiples motores de bases de datos** (relacionales y no relacionales) desde una única interfaz.

El objetivo es construir un sistema estilo “**Mini Azure / AWS RDS / Mongo Atlas**”, donde un administrador puede crear instancias de bases de datos para los estudiantes, y cada estudiante puede administrar su propio motor desde el front.

El proyecto combinará conceptos de:

- .NET 9
 - Arquitectura DDD
 - APIs modulares y extensibles
 - React con componentes dinámicos
 - Docker como infraestructura
 - Motores de BD SQL y NoSQL
 - Despliegue real en VPS
 - JWT + Roles
 - Patrones de integración
 - Investigación autónoma
-

2. Objetivo General

Construir una plataforma web donde:

ADMINISTRADOR

- Puede crear instancias de diferentes motores de BD.
- Ver estadísticas, estado, puertos, y acceso.

- Encender y apagar instancias.
- Eliminar o reiniciar contenedores.
- Asignar instancias a estudiantes.

ESTUDIANTE

- Puede acceder a sus instancias creadas.
 - Ejecutar queries SQL/Mongo/Redis según el motor.
 - Administrar tablas, colecciones, claves, etc.
 - Ver registros, editar, eliminar.
 - Explorar datos de manera gráfica.
 - Exportar datos.
-

3. Tecnologías requeridas

Backend (.NET)

- .NET 8 o .NET 9
- Arquitectura DDD (Domain, Application, Infrastructure, API)
- JWT Authentication
- Docker.DotNet o ejecución de procesos Docker
- Npgsql / MySqlConnection / SqlClient / MongoDB.Driver / StackExchange.Redis
- AutoMapper (opcional)

Frontend (React)

- React 18
- React Router
- Axios
- JWT handling
- Monaco Editor (para escribir queries)
- MUI o Tailwind (opcional)
- Tablas dinámicas

Infraestructura

- VPS (Contabo/Vultr/Oracle Free/etc.)
- Docker

- Docker Compose
 - Puertos expuestos
 - Acceso SSH
 - Despliegue manual o CI/CD (opcional)
-

4. Motores de Base de Datos a manejar

Mínimo 5 motores (obligatorio):

1. **SQL Server** (relacional)
2. **PostgreSQL** (relacional)
3. **MySQL/MariaDB** (relacional)
4. **MongoDB** (documental)
5. **Redis** (clave-valor)

Opcional avanzado:

- Neo4j (grafos)
- Elasticsearch (búsqueda)

5. Diagramas

6. Historias de Usuario

HU-001. Como usuario, quiero iniciar sesión mediante JWT para acceder al sistema según mi rol.

HU-002. Como administrador, quiero crear una instancia seleccionando el motor, puerto y credenciales.

HU-003. Como administrador, quiero iniciar y detener motores desde la plataforma.

HU-004. Como administrador, quiero asignar una instancia a un estudiante.

HU-005. Como estudiante, quiero visualizar la información principal de mi motor asignado.

HU-006. Como estudiante, quiero ejecutar consultas desde un editor integrado.

HU-007. Como estudiante, quiero realizar operaciones CRUD visuales sobre mis datos.

HU-008. Como administrador, quiero visualizar logs de contenedores.

HU-009. Como administrador, quiero reiniciar motores en caso de falla.

7. Criterios de Aceptación

- El sistema debe autenticar mediante JWT con roles.
- El administrador puede crear motores y las instancias deben levantarse correctamente.

- Cada estudiante solo puede acceder a su instancia asignada.
- El editor ejecuta consultas y muestra errores o resultados.
- El frontend debe actualizar datos sin recargar la página.
- Las operaciones deben registrarse en logs del servidor.