

Ejercicio de Aplicación #3 - Bases de Datos No Relacionales

1. ¿Qué es una base de datos NoSQL?

Una base de datos NoSQL es un sistema de almacenamiento de datos que no sigue el modelo relacional tradicional. Está diseñada para manejar grandes volúmenes de datos, escalabilidad horizontal y esquemas flexibles.

2. Diferencias con base de datos relacional

Característica	NoSQL	Relacional (RDBMS)
Modelo de datos	Flexible (documentos, clave-valor)	Tablas con filas y columnas
Esquema	Dinámico o inexistente	Rígido y predefinido
Escalabilidad	Horizontal	Vertical
Transacciones ACID	Limitadas o parciales	Completas (ACID)
Uso común	Big Data, IoT, Apps en tiempo real	ERP, sistemas bancarios

3. Ventajas y Desventajas

Ventajas:

- Alta escalabilidad.
- Alto rendimiento en lectura/escritura.
- Esquema flexible.
- Ideal para datos semi-estructurados o no estructurados.

Desventajas:

- Menor soporte para transacciones complejas.
- Consistencia eventual (según CAP).
- Mayor curva de aprendizaje si se viene de SQL tradicional.

4. Tipos principales de bases de datos NoSQL

- Key-Value: Redis, DynamoDB → ideales para caché, almacenamiento simple.
- Document: MongoDB, CouchDB → flexibles, trabajan con JSON/BSON.
- Columnar: Cassandra, HBase → eficaces para consultas masivas.
- Grafos: Neo4j, ArangoDB → representan relaciones complejas entre nodos.

5. Comparación técnica

Criterio	MongoDB (Document)	Cassandra (Columnar)	Redis (Key-Value)	Neo4j (Graph)
Escalabilidad	Alta horizontal	Muy alta horizontal	Alta	Media
Rendimiento	Muy bueno	Excelente	Extremadamente rápido	Bueno
Consistencia	Eventual (BASE)	Eventual	Alta si se configura	Alta en lecturas
Modelado de datos	Flexible (JSON)	Columnas anchas	Simple clave-valor	Nodos y aristas
Tolerancia a fallos	Alta (replicas)	Alta (peer-to-peer)	Alta	Media-alta

6. Caso real: Netflix con Cassandra

- Tipo de NoSQL utilizado: Column-oriented (Apache Cassandra).
- Razón: Necesitaban escalabilidad y replicación entre regiones.
- Beneficios: tolerancia a fallos, alto rendimiento, replicación multi-región.

7. Comparación NoSQL vs RDBMS

Escenarios ideales para NoSQL:

- Gran volumen de datos.
- Requerimientos de escalabilidad horizontal.
- Esquema flexible.

Escenarios ideales para RDBMS:

- Transacciones complejas.
- Alta consistencia.
- Estructura de datos fija y tabular.

8. Empresas líderes que usan NoSQL

- Netflix → Cassandra
- Amazon → DynamoDB
- Facebook → RocksDB, TAO
- Uber → MongoDB