

# UNIVERSIDAD NACIONAL AUTONOMA DE MÉXICO - FACULTAD DE INGENIERIA

## BASE DE DATOS - TAREA 1

Lilian De la peña Osorio - Grupo 01

14 de agosto 2025

### 1 Modelo orientado a objetos

La orientación a objetos representa el mundo real y resuelve problemas a través de objetos, ya sean tangibles o digitales. Este paradigma tecnológico considera un sistema como una entidad dinámica formada de componentes. Un sistema sólo se define por sus componentes y la manera en que éstos interactúan. En el caso de los sistemas de gestión de base de datos orientada a objetos (OODBMS por sus siglas en inglés), la persistencia implica almacenar los valores de atributos de un objeto con la transparencia necesaria para que el desarrollador de aplicaciones no tenga que implementar ningún mecanismo distinto al mismo lenguaje de programación orientado a objetos.

### 2 Modelos NoSQL (clave-valor, documentales, grafos y columnar) Descripción, ventajas, desventajas, casos de uso

Las bases de datos NoSQL, también conocidas como bases de datos personalizadas, están diseñadas para modelos de datos específicos y almacenan los datos en esquemas flexibles que se escalan con facilidad para aplicaciones modernas. Las bases de datos NoSQL, también conocidas como bases de datos personalizadas, están diseñadas para modelos de datos específicos y almacenan los datos en esquemas flexibles que se escalan con facilidad para aplicaciones modernas.

**-VENTAJAS:** Las bases de datos NoSQL están diseñadas específicamente para modelos de datos no relacionales y tienen esquemas flexibles para crear aplicaciones modernas.

1. Flexibilidad: Las bases de datos NoSQL generalmente ofrecen esquemas flexibles que permiten un desarrollo más rápido y más iterativo. El modelo de datos flexible hace que las bases de datos NoSQL sean ideales para datos semiestructurados y no estructurados.
2. Escalabilidad: Las bases de datos NoSQL generalmente están diseñadas para escalar horizontalmente usando clústeres distribuidos de hardware, en lugar de escalar añadiendo servidores caros y sólidos. Algunos proveedores de la nube manejan estas operaciones en segundo plano, como un servicio completamente administrado.
3. Alto rendimiento: Las bases de datos NoSQL están optimizadas para modelos de datos y patrones de acceso específicos. Esto permite un mayor rendimiento que si intentara lograr una funcionalidad similar con bases de datos relacionales.
4. Altamente funcional: Las bases de datos NoSQL proporcionan API altamente funcionales y tipos de datos que están diseñados específicamente para cada uno de sus respectivos modelos de datos.

**-DESVENTAJAS:** La falta de estructura puede hacer que los datos sean poco fiables y difíciles de organizar. A continuación, se enumeran algunas quejas comunes:

1. La falta de SQL: la queja constante sobre NoSQL es que no viene con SQL. SQL es una tecnología madura, diseñada para organizar datos, mientras que NoSQL no lo es. Los analistas están completamente familiarizados con SQL y úsalo regularmente. NoSQL se diseñó inicialmente para investigar cantidades masivas de datos almacenados y desorganizados, recopilados de múltiples fuentes.
2. La falta de estandarización: Cada marca de base de datos NoSQL utiliza su propio esquema único. En algunos casos (como MongoDB), no existe esquema. En otras situaciones (como Elasticsearch), es dinámico. En otros diseños, se asemeja a las bases de datos relacionales (por ejemplo, Cassandra). El problema aquí es que cada sistema único tiene sus propias fortalezas y debilidades, que deben conocerse antes de elegir la base de datos NoSQL adecuada para la situación en cuestión.
3. La ausencia de uniones: SQL utiliza uniones para establecer conexiones entre tablas de bases de datos mediante la coincidencia de columnas, lo que a su vez crea una relación entre ellas. La mayoría de las consultas complejas utilizadas en una base de datos SQL implican comandos de unión. En esencia, una condición de unión ayuda a combinar los datos de dos o más tablas de bases de datos.

**-CASOS DE USO:** La gama de bases de datos NoSQL y sus respectivos casos de uso son muy variados. Si bien es difícil presentar un conjunto representativo de casos de uso, a continuación proporcionamos algunos ejemplos ilustrativos

para iniciar la reflexión y lo alentamos a obtener más información sobre cada base de datos NoSQL y sus respectivos casos de uso.

1.Administración de datos en tiempo en tiempo real: Puede ofrecer recomendaciones en tiempo real, personalización y experiencias de usuario mejoradas con bases de datos NoSQL. Por ejemplo, Disney+ ofrece su amplia biblioteca de contenido digital a más de 150 millones de suscriptores mediante la tecnología de base de datos NoSQL.

2.Aplicaciones de alta disponibilidad: Las bases de datos NoSQL distribuidas son excelentes para crear aplicaciones de alta disponibilidad y baja latencia para mensajería, redes sociales, uso compartido de archivos y más. Por ejemplo, Snapchat tiene más de 290 millones de usuarios que envían miles de millones de imágenes y mensajes de video a diario.

### 3 Referencias

- “Modelo Orientado a Objetos”. 403 Forbidden. Accedido el 15 de agosto de 2025. Disponible: [https://repositorio/moodle/pluginfile.php/2731/mod\\_resource/content/1/UPA-Modelo-Orientado-Objetos/index.html](https://repositorio/moodle/pluginfile.php/2731/mod_resource/content/1/UPA-Modelo-Orientado-Objetos/index.html)

-“Bases de datos no relacionales — Bases de datos de gráficos — AWS”. Amazon Web Services, Inc. Accedido el 15 de agosto de 2025. [En línea]. Disponible: <https://aws.amazon.com/es/nosql/>