



**Universidad Nacional  
Autónoma de México.  
Facultad de Ingeniería.**



## ***BASES DE DATOS.***

***Profesor:***

***Fernando Arreola Franco.***

***Grupo: 1.***

***Tarea 1.***

***Alumna:***

***Segura Garduño Karen Alin.***



***Fecha de entrega: 09-febrero -2022.***

## Modelo Orientado a Objetos.

Con el modelo de datos orientado a objetos, tanto los datos como sus relaciones están contenidos en una única estructura conocida como objeto.

El desarrollo de cualquier sistema consiste en realizar tres etapas: análisis, diseño y programación. Durante el proceso de análisis, en el modelo orientado a objetos, se realiza el modelado y la declaración de objetos. El manejo de los objetos está determinado por las implementaciones de los métodos detectados en el análisis.

En el análisis de la estructura de objetos se realiza lo siguiente:

- Se identifican los tipos de objetos y sus asociaciones representados a través de un esquema de objetos.
- Se organizan los tipos de objetos en subtipos y supertipos, indicando la jerarquía por medio de herencia.
- Estudiar la composición de objetos complejos, definiendo el mecanismo que controla la naturaleza de los objetos que se encuentran dentro de otros objetos.

### Características del Modelo Orientado a Objetos:

- El Modelado Orientado a Objetos está basado en el paradigma orientado a objetos.
- Trata el almacenamiento de objetos (persistencia de los objetos).
- Define un lenguaje para la definición y manipulación de objetos. Incluye mecanismos para optimizar el acceso (Indexación y Clustering), el control de la concurrencia, seguridad y gestión de usuarios, facilidad de consulta y recuperación ante fallos.
- Debido a que es un esquema orientado a objetos incluye: Encapsulamiento, herencia, polimorfismo, etc.

### Conceptos clave:

- La encapsulación oculta información al resto de objetos, de manera que pueden impedir los conflictos o los accesos incorrectos. Consiste en unir en la clase las variables (características) y los métodos (comportamientos), de manera que solo se tiene una unidad, de la que se conoce su comportamiento, pero no los detalles internos.
- La herencia en base de datos orientada a objetos hace referencia a que los objetos heredan comportamientos dentro de una jerarquía de clases, es decir, una clase se deriva de otra de manera que extiende su funcionalidad. La clase de la que se hereda puede llamarse clase base, clase padre,

superclase, clase ancestro, etc. (dependiendo del lenguaje de programación que se esté usando)

- El polimorfismo es la propiedad que permite que una operación pueda aplicarse a objetos de distinta tipología.

## Modelos NoSQL.

Las bases de datos NoSQL son reconocidas porque son fáciles de desarrollar, por su funcionalidad y el rendimiento a escala. Utilizan una variedad de modelos de datos para acceder y administrar datos. Estos tipos de bases de datos están optimizados específicamente para aplicaciones que requieren grandes volúmenes de datos, baja latencia y modelos de datos flexibles, lo que se logra mediante la flexibilización de algunas de las restricciones de coherencia de datos en otras bases de datos.

Los datos almacenados no requieren estructuras fijas como tablas, normalmente no soportan operaciones JOIN, ni garantizan completamente ACID (atomicidad, consistencia, aislamiento y durabilidad), y habitualmente escalan bien horizontalmente. Las Bases de Datos pertenecen al modelo no relacional (los datos no tienen por qué estar relacionados entre sí y, por lo tanto, no tienen que almacenarse en estructuras fijas como las tablas del modelo de base de datos relacional.)

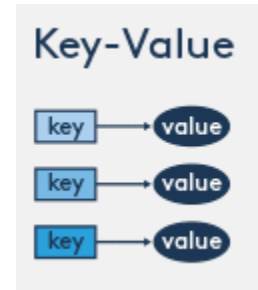
### Características y ventajas de NoSQL.

- SQL no es el lenguaje de consulta/modificación de datos principal, aunque sí lo soportan, de ahí el nombre No Solo SQL.
- Los datos no tienen que almacenarse en tablas.
- Generalmente, su arquitectura es distribuida, almacenándose la información en más de una máquina del sistema. Por lo tanto, los sistemas que las soportan tienen una mayor escalabilidad horizontal (a mayor número de nodos mayor rendimiento) y también mayor tolerancia ante fallos en los distintos nodos.
- Son más eficientes en el procesamiento de los datos que las BBDD relacionales, por eso, son la elección para aplicaciones que hacen un uso intensivo de estos (“*streaming*”, etc.).
- Utilizan lo que se conoce como consistencia eventual, que consiste en que los cambios realizados en los datos serán replicados a todos los nodos del sistema, lo cual aumenta el rendimiento de estos sistemas en contraposición a las propiedades ACID de las BBDD relacionales (“*Atomicity, Consistency,*

*Isolation and Durability*” – Atomicidad, Consistencia/Integridad, Aislamiento y Durabilidad).

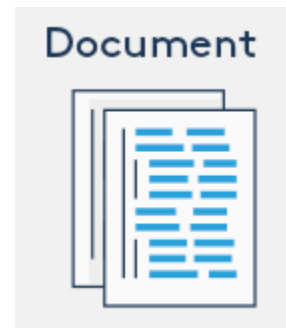
### **Tipos de bases de datos NoSQL.**

- **Bases de datos clave – valor:** Son altamente divisibles y permiten escalado horizontal a escalas que otros tipos de bases de datos no pueden alcanzar. En este tipo de sistema, cada elemento está identificado por una llave única, lo que permite la recuperación de la información de forma muy rápida, información que habitualmente está almacenada como un objeto binario (BLOB). Se caracterizan por ser muy eficientes tanto para las lecturas como para las escrituras.



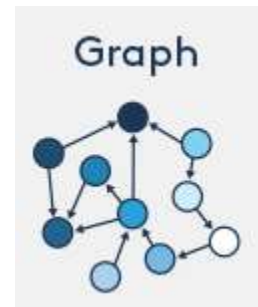
Algunos ejemplos de este tipo son Cassandra, BigTable o HBase.

- **Bases de datos documentales:** Facilitan a los desarrolladores el almacenamiento y la consulta de datos en una base de datos mediante el uso del mismo formato de modelo de documento que emplean en el código de aplicación. La naturaleza flexible, semiestructurada y jerárquica de los documentos y las bases de datos de documentos permite que evolucionen según las necesidades de las aplicaciones. El modelo de documentos funciona bien con catálogos, perfiles de usuario y sistemas de administración de contenido en los que cada documento es único y evoluciona con el tiempo.



Algunos ejemplos de este tipo son MongoDB o CouchDB.

- **Bases de datos de grafos:** Una base de datos de grafos utiliza estructuras de grafos para almacenar, correlacionar y consultar relaciones. Proporcionan una adyacencia libre de índice, de modo que los elementos adyacentes se unen entre sí sin usar un índice. Los casos de uso típicos para una base de datos de gráficos incluyen redes sociales, motores de recomendaciones, detección de fraude y gráficos de conocimiento.



Algunos ejemplos de este tipo son Neo4j, InfoGrid o Virtuoso.

## Bibliografías:

- Acens. (2014, febrero). Bases de datos NoSQL. Qué son y tipos que nos podemos encontrar. [online]. Disponible en: <https://www.acens.com/wp-content/images/2014/02/bbdd-nosql-wp-acens.pdf>
- Amazon. (s.f). ¿Qué es NoSQL?. [online]. Disponible en: <https://aws.amazon.com/es/nosql/>
- UNIR. (2021, septiembre). Bases de Datos NoSQL: qué son y cuáles son sus ventajas. [online]. Disponible en: <https://mexico.unir.net/ingenieria/noticias/bases-de-datos-nosql/#:~:text=Las%20Bases%20de%20Datos%20NoSQL,ventajas%20de%20este%20tipo%20son%3A&text=Los%20datos%20no%20tienen%20que,de%20una%20m%C3%A1quina%20del%20sistema.>
- Julian. (s.f). Encapsulamiento, herencia y polimorfismo en BDOO. [online]. Disponible en: <https://sites.google.com/site/topicosunidad2/encapsulamiento-herencia-y-polimorfismo-en-bdoo>
- Miranda. F.(2013, enero). Modelado Orientado a Objetos. [online]. Disponible en: <https://es.slideshare.net/RafaelMiranda2/modelado-orientado-a-objetos>
- Glosarioit. (s.f). Modelo orientado a objetos - Sección BD/Programación. [online]. Disponible en: [https://www.glosarioit.com/Modelo\\_orientado\\_a\\_objetos](https://www.glosarioit.com/Modelo_orientado_a_objetos)