**Recopilacion de informacion**

Kevin Andres Alzate Pino

**ficha**:2671333

Centro Tecnologico Agroindustrial

cartago - valle del cauca 2024

Análisis y Desarrollo de Software

1. diferencias bases de datos relacionales y no relacionales:

Las bases de datos relacionales y no relacionales difieren en su estructura y enfoque de almacenamiento de datos.

**Bases de Datos Relacionales**:

1. Estructura:

- Utilizan tablas para organizar y estructurar los datos.

- Las relaciones entre las tablas se establecen mediante claves primarias y foráneas.

2. Esquema Fijo:

- Tienen un esquema de datos fijo y predefinido.

- Cambiar el esquema puede ser más complicado y requerir alteraciones en varias tablas.

3. Transacciones ACID:

- Adhieren a propiedades ACID (Atomicidad, Consistencia, Aislamiento, Durabilidad) para garantizar la integridad de los datos.

4. Escalabilidad Vertical:

- La escalabilidad suele lograrse mediante la mejora de los recursos de hardware (escalabilidad vertical).

**Bases de Datos No Relacionales (NoSQL)**:

1. Estructura Flexible:

- Pueden utilizar diversos modelos de datos, como documentos, clave-valor, columnares o de grafos.

- La estructura de los datos puede variar según la necesidad.

2. Esquema Dinámico:

- No tienen un esquema fijo, permitiendo la inserción de datos sin una estructura predefinida.

- Adaptados para datos no estructurados o semiestructurados.

3. Escalabilidad Horizontal:

- Suelen ser más adecuadas para la escalabilidad horizontal, agregando nodos a medida que crece la carga.

4. Diversos Modelos:

- Pueden ser de tipo documental (como MongoDB), clave-valor (como Redis), columnares (como Cassandra) o de grafos (como Neo4j).

En resumen, las bases de datos relacionales se centran en la estructura tabular con relaciones definidas, mientras que las no relacionales ofrecen flexibilidad en la estructura de los datos

1. cuadro comparativo ventajas / desventajas:

| **Aspecto** | **Bases de Datos Relacionales** | **Bases de Datos No Relacionales (NoSQL)** |
| --- | --- | --- |
| **Estructura de Dato** | Estructura tabular organizada en tablas. Relaciones definidas entre tablas. | Estructura flexible según el modelo elegido (documento, clave-valor, etc.). Pueden manejar datos no estructurados. |
| **Esquema** | Esquema fijo y predefinido. Los cambios pueden ser complejos. | Esquema dinámico. Permite inserción de datos sin estructura predefinida. |
| **Escalabilidad** | Escalabilidad vertical (mejora de recursos de hardware). | Escalabilidad horizontal (añadiendo nodos). Mejor para entornos distribuidos. |
| **Modelos de Datos** | Principalmente modelos relacionales. SQL es el lenguaje común. | Diversos modelos: documental, clave-valor, columnar, gráfico. Lenguajes específicos para cada modelo. |
| **Transacciones ACID** | Cumplen con propiedades ACID para garantizar consistencia | Algunas bases de datos NoSQL pueden relajar propiedades ACID para mejorar rendimiento en ciertos casos. |
| **Flexibilidad y Agilidad** | Menos flexible en términos de cambios en el esquema. | Mayor flexibilidad en la estructura de datos. Más ágil para cambios en la aplicación. |
| **Aplicaciones Típicas** | - Aplicaciones donde la integridad de los datos es crítica (sistemas transaccionales). | Escenarios con grandes volúmenes de datos y necesidad de escalabilidad horizontal. Datos no estructurados o semiestructurados. |
| **Ejemplos** | MySQL, PostgreSQL, Oracle. | MongoDB, Cassandra, Redis, Neo4j. |

**uso para las base de datos relacionales/no relacionales**

**Bases de Datos Relacionales (RDBMS):**

1. Sistemas de Gestión de Contenido (CMS):

- Ejemplo: WordPress utiliza MySQL para almacenar y gestionar contenido como entradas de blog, páginas y comentarios.

2. Sistemas de Comercio Electrónico:

- Ejemplo: Shopify utiliza bases de datos relacionales para gestionar información de productos, inventario y transacciones.

3. Sistemas Financieros:

- Ejemplo: Aplicaciones bancarias y plataformas de servicios financieros utilizan bases de datos relacionales para garantizar la integridad y consistencia de las transacciones.

4. Sistemas de Reservas y Gestión de Reservas:

- Ejemplo: Un sistema de reservas de hoteles podría utilizar una base de datos relacional para gestionar información sobre habitaciones, disponibilidad y reservas.

**Bases de Datos No Relacionales (NoSQL):**

1. Redes Sociales y Aplicaciones de Redes Sociales:

- Ejemplo: MongoDB se ha utilizado en aplicaciones de redes sociales para manejar grandes volúmenes de datos no estructurados, como perfiles de usuarios y actividades.

2. Análisis de Grandes Volúmenes de Datos (Big Data):

- Ejemplo: Apache Cassandra se utiliza para gestionar grandes cantidades de datos distribuidos en clústeres, siendo útil en entornos de big data.

3. Aplicaciones de Tiempo Real:

- Ejemplo: Firebase Realtime Database (una base de datos en tiempo real NoSQL) se utiliza en aplicaciones que requieren actualizaciones instantáneas, como aplicaciones de mensajería.

4. Gestión de Datos Geoespaciales:

- Ejemplo: Neo4j, una base de datos de grafos, se utiliza para aplicaciones que gestionan datos geoespaciales, como sistemas de navegación y mapas interactivos.