

## Práctica 1: Instalación de Docker y del SMBD, PostgreSQL

<b>Integrantes:</b>	Arcos Morales Ramón :	319541478
	Casarrubias Casarrubias Victor Manuel :	421003581
	Castillo Hernández Eduardo :	420003557
	López Asano Miguel Akira :	320219089
	Rivera Lara Sandra Valeria :	320039823

**Fecha:** 26/08/2024

### Ejercicio 1. *¿Qué otros SMBD existen actualmente en el mercado?*

Existen diferentes Sistemas Manejadores de Bases de Datos; por decir algunos: [AltexSoft, 2023]

- MySQL : su dueño es Oracle Corporation.
- MariaDB : con licencia GNU, es de open-source.
- Oracle : como su nombre lo indica, es perteneciente a Oracle Corporation.
- MSSQL : esta herramienta comercial es proporcionada por Microsoft.
- SQLite : proporcionada por SQLite Consortium.
- MongoDB : sistema no relacional.
- Redis : sistema no relacional open-source.
- Cassandra : sistema creado por Apache, basado en Java.
- Elasticsearch : sistema no relacional, almacena los datos como archivos JSON.
- Firebase : sistema no relacional, el cual es propiedad de Google.
- Amazon DynamoDB : sistema no relacional propiedad de Amazon.

### Ejercicio 2. *¿Cuáles son las principales diferencias con PostgreSQL?*

Algunas características que diferencian a este SMBD son:

- Licencia: A diferencia de sistemas como Oracle, MSSQL y MongoDB, vemos que PostgreSQL es completamente open-source. Esto significa que se tiene que pagar licencia o hay restricciones comerciales con los otros sistemas. Mientras que PostgreSQL al ser de código abierto, se pueden crear y conseguir bastantes herramientas y funcionalidades extras gratis [AltexSoft, 2023].
- Escalabilidad: Este sistema posee una escalabilidad vertical, es decir, que los datos se guardan en un servidor al que se le va agregando más poder computacional para crecer [Team, 2024]. Igualmente PostgreSQL permite clusters de servidores, aunque con unas limitaciones. Hay SMBD que sólo poseen escalabilidad horizontal, como Cassandra o Elasticsearch.
- Soporte de datos: PostgreSQL es un SMBD para bases de datos relacionales de objetos, el cual admite una gran variedad de datos como JSON, el cual es NoSQL. Además, permite a los usuarios crear sus propios tipos de datos, así como guardar datos como objetos con propiedades. Por otra parte, hay sistemas puramente relacionales, como MySQL, no relacionales o que no son tan completos [AltexSoft, 2023].

- Seguridad: PostgreSQL tiene funcionalidades más sólidas que las de sistemas como MySQL, las cuales ayudan a mantener la seguridad e integridad de las bases de datos. PostgreSQL brinda ACID en todas las configuraciones, MVCC, control de acceso basado en roles, el cifrado SSL y la seguridad a nivel de fila. Pero no tiene instrumentos de auditoría, los cuales sí poseen sistemas como Oracle [Team, 2023].
- Indexación: PostgreSQL brinda varios tipos de índices, tales como Árbol-B, árbol de búsqueda generalizada (GiST), SP-GiST, hash, invertidos (GIN) y BRIN. Además soporta índices basados en expresiones e índices parciales. Pero los demás sistemas no ofrecen aquella variedad de opciones. MSSQL ofrece índices clúster y no clúster [Team, 2024], MySQL admite sólo árboles-B y árboles-R [Services and Services, ], etc.

**Ejercicio 3.** *¿Por qué una empresa debería escoger una base de datos open source?*

Existen varias razones por las cuales una empresa debería escoger una base de datos de código abierto u open source, entre ellas se encuentra el costo, que como es mencionado en [Logic, 2024] suele ser gratis, por lo que si la empresa no es grande, puede invertir en su lugar en establecer la infraestructura con herramientas de código abierto y aunque puede que el costo inicial sea alto por esto, a la larga dará frutos pues no estará obligado a pagar por la licencia. El hecho de que sea de código abierto también permite adaptar o modificar dicho software para las necesidades específicas de la empresa como es mencionado en [Logic, 2024]. Una razón muy importante para elegir una base de datos de código abierto es la seguridad ya que como se menciona en [Logic, 2024] y [Jyothi, 2022], al ser de código abierto, es posible detectar fallos o bugs en el código que puedan ser vulnerabilidades más rápido, además de poder realizar parches para estos bugs más rápidos que en un software licenciado y de código cerrado. También como es mencionado en [Logic, 2024], las bases de datos de código abierto están diseñadas para poder escalar eficientemente. Por último, cabe mencionar que si se decidiera optar por una opción de código cerrado y con licencia, se cae en el riesgo de que dicha empresa cambie sus políticas mientras se hace uso de su software, afectando a la empresa pues quizá tenga que hacer cambios en el presupuesto previsto para manejar este tipo de cosas, incluso hasta tener que cambiar el sistema porque ya no se adapta a las necesidades.

**Ejercicio 4.** *¿Cuáles son las ventajas, para un DBA el trabajar con un SMBD, open source?*

- Independencia tecnológica. No se depende del fabricante del software, el DBA, puede modificar a libertad el funcionamiento del SMBD para adaptarlo a necesidades específicas de cada usuario y de la empresa, de igual forma, permite la posibilidad de arreglar fallos, estandarizar las versiones utilizadas y optimizar el rendimiento, garantizando la viabilidad y adaptabilidad a largo plazo.
- Transparencia y seguridad. El software de código abierto, le permite al DBA total transparencia sobre el funcionamiento interno del SMBD con el que está trabajando, evitando así, posibles acciones malintencionadas por parte de la empresa distribuidora del programa. La búsqueda de vulnerabilidades de seguridad puede ser llevada a cabo de una manera más rigurosa y por lo tanto actuar de forma inmediata ante cualquier brecha de seguridad.
- Colaboración. La comunidad de código abierto es bastante diversa pues hay desde entusiastas, desarrolladores, hasta académicos colaborando en un mismo proyecto. Este conocimiento en conjunto puede ser de utilidad para el DBA, pues una visita a algún foro sobre la SMBD que utiliza, le permitirá acceder a nuevas ideas, desarrollar otras o incluso compartirlas, mejorando su entendimiento de la herramienta utilizada y en el caso ideal, poder llevarlas a cabo para un mejor desarrollo del SMBD.

**Ejercicio 5.** *¿Qué son las bases de datos NoSQL? Menciona 3 ventajas y desventajas contra las bases relacionales.*

Las bases de datos NoSQL son aquellas que no son relacionales y guardan datos no estructurados o semi-estructurados sin hacer uso de tablas, campos y columnas. Usualmente guardan la información en parejas de llave-valor o utilizan los documentos JSON. [A.Nayak, 2013] [MongoDB, ]

Tres ventajas de las bases de datos NoSQL sobre las relacionales son:

1. Son fáciles de escalar y son tolerantes a particiones, haciendo que sean la mejor opción para usar en la nube y a través de distintos servidores.

2. Son muy flexibles, ideales para datos dinámicos y cambios frecuentes.
3. Existen diversos modelos de datos (Llave-valor, documentos, gráficas, O.O., entre otros.) para distintas aplicaciones.

Tres desventajas de las bases de datos NoSQL en comparación de las relacionales:

1. Tienen normalización, lo que permite que los datos sean consistentes.
2. Cumplen con los criterios ACID (Atomicity, Consistency, Isolation, and Durability), creando más seguridad e integridad para prevenir que la base de datos posea estados inconsistentes.
3. Poseen lenguajes de búsqueda (SQL), haciendo que sean más simples de utilizar y consultar.

[J.Montemagno, 2022] [MongoDB, ]

## Referencias

- [AltexSoft, 2023] AltexSoft (2023). Comparing Database Management Systems: MySQL, PostgreSQL, MSSQL Server, MongoDB, Elasticsearch, and others.
- [A.Nayak, 2013] A.Nayak, A.Poriya, D. (2013). Type of nosql databases and its comparison with relational databases. *International Journal of Applied Information Systems*.
- [J.Montemagno, 2022] J.Montemagno, H. (2022). Sql vs. nosql data. <https://learn.microsoft.com/en-us/dotnet/architecture/cloud-native/relational-vs-nosql-data>.
- [Jyothi, 2022] Jyothi, J. (2022). Mogodb: A nosql database with amazing advantages and features. *Journal homepage: www. ijprpr. com ISSN, 2582:7421*.
- [Logic, 2024] Logic, S. (2024). Difference between open source database and licensed database — Comprehensive guide.
- [MongoDB, ] MongoDB (-). Relational vs. non-relational databases. <https://www.mongodb.com/resources/compare/relational-vs-non-relational-databases>.
- [Services and Services, ] Services, A. W. and Services, A. W. S. W. ¿Cuál es la diferencia entre MySQL y PostgreSQL?
- [Team, 2023] Team, A. A. (2023). PostgreSQL vs. Oracle: What's the Difference?
- [Team, 2024] Team, E. (2024). A Complete Comparison of PostgreSQL vs Microsoft SQL Server.