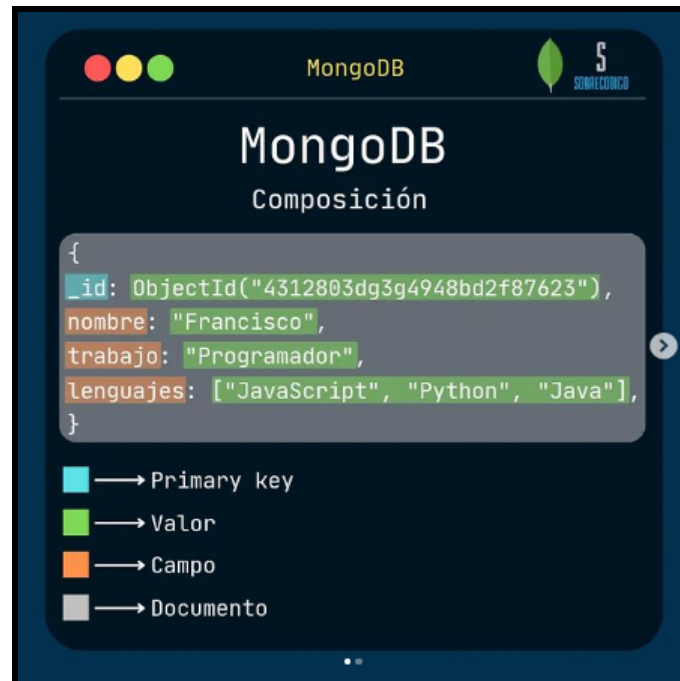


Objetivos:

- ❖ Valorar las ventajas e inconvenientes de utilizar una base de datos NoSQL.
- ❖ Instalar el gestor de base de datos.
- ❖ Establecer la conexión con la base de datos.
- ❖ Desarrollar aplicaciones que gestionan la información almacenada en bases de datos NoSQL: efectuar consultas, añadir y eliminar colecciones a la base de datos desde diferentes entornos de desarrollo.



"MongoDB es la base de datos NoSQL líder. Nos permite trabajar con documentos JSON binarios (BSON) en lugar de utilizar un sistema clásico de tablas y relaciones lo cual hace que Mongo sea muy veloz y cada vez más utilizado en el mundo del desarrollo."



Es importante entender que no siempre que almacenamos datos, vamos a usar SQL. MongoDB, entre otros, nos permite a diferencia de SQL, almacenar información de manera no tan estructurada.

Tarea:

Efectuar desde diferentes entornos consultas, adiciones y borrado de colecciones a una base de datos MongoDB propuesta por el alumno.

- **Requerimiento 1:** desarrollo de un CRUD básico y presentación de la documentación requerida de al menos una de las siguientes aplicaciones de consola:
 - node.js;
 - C#;
 - C# con MongoDBEntities
- **Requerimiento 2:** aplicación con interfaz gráfica que realice el CRUD de la base de datos y alguna **funcionalidad adicional** (ej. ampliación de posibles consultas)
 - Aplicación WPF, o
 - Aplicación ASP Net o
 - Aplicación Windows Form
- **Requerimiento 3** - además de los requerimientos 1 y 2, el alumno desarrolla:
 - Un segundo tipo de aplicación Windows Form, WPF o ASP Net diferente al realizado en el requerimiento 2 con otra nueva **funcionalidad adicional**, o
 - Una API Rest con Net (**AspNet WebAPI**). En este caso la interacción con la aplicación se realizará con Postman

Recursos

Lo primero que tienes que hacer es: [Instalar MongoDB y MongoDB Compass, su interfaz gráfica.](#)

Ejemplos en consola:

- [Conectar C# a MongoDB | Get, insert, update y delete](#) (ap. de consola) (13mn.)
- [Tutorial: MongoDB con C# ¡La Manera Fácil!](#) (ap. de consola y librería [MongoDB.Entities](#))
- [Node.js MongoDB en W3Schools](#) (ap. de consola con node.js; JavaScript)

Ejemplos ap con interfaz gráfica:

- [MongoDB en aplicación WPF](#) (aplicación WPF .NET Core)
- [Crear CRUD Maestro/Detalle en MVC .NET Core y MongoDB](#) (asp Net MVC)
- [Conexión y mantenimiento de datos en mongodb en aplicación Windows form](#) (windows forms)

Ejemplo WebApi

- [ASP.NET Core WebApi + MongoDB](#)

Herramientas para el backup:

- El enlace oficial de documentos está [aquí](#) .
- [En w3schools también tienes indicaciones para copia de seguridad y recuperación](#)

Otros recursos:

Todo MongoDB en:

- [Introducción a MongoDB](#)
- [Bases de datos y colecciones](#)
- [Vistas](#)
- [Operaciones de MongoDB CRUD](#)

Nota Aclaratoria:

La actividad es de **carácter individual**

No podrá coincidir la base de datos de trabajo con ninguna de las que se emplean en los ejemplos propuestos de los recursos proporcionados, ni tampoco se admiten coincidencias con trabajos presentados por otros alumnos.

Entregables:

- Backup (*mongodump* y *mongorestore*) o Json/bson de la base de datos creada.
- Aplicación / aplicaciones desarrolladas
- Documentación. Pdf explicativo en el que se indique claramente:
 - **Requerimientos completados.** Incluye al comienzo de la documentación la siguiente tabla resumen completada según tu desarrollo:

	tipo de aplicación (<i>consola/ Windows Form, WPF, Asp Net MVC /Web API</i>)
Req. 1	
Req .2	
Req. 3	

- **Muestra de la BD:** incluye en la documentación una captura o muestra de la estructura de los documentos Bson/Json de la BD
- **Para la prueba de la aplicación:** indicaciones para restaurar la base de datos empleada en el servidor. Indica claramente el tipo de backup que proporcionas (cómo has hecho el backup) y los pasos a seguir para restaurar la base de datos y probar la aplicación en cualquier equipo de desarrollo.
- **Funcionalidad.** Breve descripción de:
 - cada una de las opciones del CRUD. Código y capturas de pantalla demo de la ejecución exitosa del mismo.
 - la funcionalidad añadida (para requerimientos 2 y 3). Incluir también capturas de pantalla demostrativas de la ejecución exitosa y código relevante para su logro.
- **Otros.** Investigación realizada y/o fuentes consultadas, dificultades encontradas y otros aspectos de interés a reseñar

Fecha límite de entrega: 17 de marzo a las 14:10h