



Universidad De Granada

E.T.S. DE INGENIERÍAS INFORMÁTICA Y DE TELECOMUNICACIÓN

## TRABAJO TEMA 4

*Diseño y desarrollo de sistemas de información*

**Grupo :**

D1 FlyUGR

**Autores :**

Noura Lachhab Bouhmadi

Pablo Gálvez Ortigosa

Quintín Mesa Romero

Jose Luis Mera Cardoso

# TRABAJO TEMA 4

Hemos elegido MongoDB como Sistema Gestor de Base de Datos NoSQL, que como sabemos, es un sistema orientado a documentos y de código abierto.

Crearemos una base de datos que almacenará, por una parte, las características de los aviones de las distintas aerolíneas, así como las opiniones publicadas por los pasajeros tras cada vuelo.

## 1. Instalación

Hemos instalado **mongodb** en Ubuntu usando el comando:

```
$ sudo apt install -y mongodb
```

Hemos iniciado el servicio **mongodb** y comprobado su funcionamiento usando los comandos:

```
$ sudo systemctl start mongodb
```

```
$ sudo systemctl status mongodb
```

Utilizando la terminal, la orden **mongo** nos permite conectarnos a la terminal de MongoDB, donde podremos crear nuestra base de datos [2] .

## 2. Creación de la base de datos. Operaciones de consulta, creación, inserción, actualización y eliminación de estructuras.

La creación de la base de datos se realiza con el comando:

```
$ use nombreBaseDeDatos
```

Para crear una colección, que sería lo equivalente a crear una tabla en una base de datos relacional, simplemente utilizamos el comando:

```
$ db.createCollection("nombreColeccion")
```

y podemos eliminarla utilizando el comando:

```
$ db.collection.drop()
```

Podemos insertar un documento en la colección mediante el comando:

```
$ db.collection.insertOne({documento})
```

Para consultar los objetos de una colección:

```
$ db.collection.find(options).pretty()
```

El comando para actualizar los documentos de una colección es el siguiente:

```
$ db.collection.updateMany(options)
```

y para eliminar documentos de una colección:

```
$ db.collection.deleteMany(options)
```

En los casos anteriores, *options* refiere a valores específicos de **clave:valor** que determinan a qué objetos afectará la consulta, actualización o borrado [4], [6].

### 3. Ejemplo práctico de los comandos anteriores

En la terminal de MongoDB, el comando

```
$ use FlyUGR
```

nos permite crear una base de datos llamada FlyUGR, así como seleccionarla para las siguientes operaciones.

Para crear una colección, que sería lo equivalente a crear una tabla en una base de datos relacional, simplemente utilizamos el comando:

```
$ db.createCollection("reviews")  
$ db.createCollection("planes")
```

Insertamos ahora dos reseñas:

```
$ db.reviews.insertOne({"flight": "AIR9345", "name":  
"Jules", "stars": 5, "class": "Business", "comment":  
"Perfect!"})
```

```
$ db.reviews.insertOne({"flight": "AIR9349", "name":  
"Julia", "surname": "Watson", "stars": 2, "class": "First  
class", "tags": ["Noisy", "Uncomfortable"]})
```

y dos aviones:

```
$ db.planes.insertOne({"Name": "A330", "Manufacturer":  
"Airbus", "Engines": "RR Trent", "Owned by": ["Turkish  
Airlines", "Iberia", "Air China", "Lufthansa"]})
```

```
$ db.planes.insertOne({"Name": "Boeing 787",  
"Manufacturer": "Boeing", "Alias": "Dreamliner", "Owned  
by": ["Air Europa", "Japan Airlines", "Ryanair"]})
```

Podemos ahora actualizar todas las reseñas con menos de 3 estrellas para que añadir en su campo de etiquetas la opción "Bad review":

```
$ db.reviews.updateMany({stars: {$lt: 3}}, {$push: {tags:
    "Bad review"}})
```

Queremos ahora guardar el número de asientos del avión con nombre “Boeing 787”. El avión tiene 215 asientos si se divide en tres clases y 242 si tiene configuración única. Por tanto:

```
db.planes.updateOne({Name: "Boeing 787"}, {$set: {seats:
    [{"Classes":1, "Number":242},{ "Classes":3,
    "Number":215}]}})
```

y podemos consultar todas las reseñas o aviones usando:

```
$ db.reviews.find().pretty()
$ db.planes.find().pretty()
```

Finalmente, podemos eliminar las reseñas del vuelo “AIR9349”:

```
$ db.reviews.deleteMany("Flight":"AIR9349")
```

#### **4. Conexión a la base de datos desde una aplicación**

Tal y como se detalla en la documentación de MongoDB [5], la conexión a una base de datos de Mongo se realiza a través de la shell de MongoDB (también conocida como mongosh), o a través de un driver, que puede ser específico de un determinado lenguaje de programación.

Por ejemplo, en Python, se utiliza un driver llamado pymongo, cuyo funcionamiento se detalla en [3].

#### **5. ¿Es adecuado MongoDB para implementar un sistema de venta de billetes de avión?**

En general, es mucho más adecuado realizarlo con un sistema de base de datos relacional.

En el sistema son muy importantes las relaciones entre distintas entidades, como las reservas y los usuarios y los vuelos. Un sistema relacional simplifica mucho el mantenimiento de la integridad referencial en comparación a un sistema orientado a documentos.

Además, todas las transacciones que se realizan en el sistema (pagos, reservas) requieren de ACIDez, lo que descarta una parte importante de los sistemas NoSQL.

Los vuelos, reservas, pagos, etc. son entidades con una estructura bien

definida, por lo que no tendría mucho sentido utilizar MongoDB para modelarlos. Además, las tuplas tienen, en general, pocos atributos que se suelen repetir entre las tuplas.

Quizás se podría plantear implementar la parte de gestión de usuarios utilizando MongoDB, dando mayor libertad a los usuarios para añadir más datos.

También sería interesante utilizar un sistema NewSQL para la implementación, ya que se prevé que el sistema implementado tendrá un gran volumen de transacciones.

## BIBLIOGRAFÍA

- [1] Papiernik, Mateusz. “An Introduction to Document-Oriented Databases.” *DigitalOcean*, 20 de julio de 2021, [digitalocean.com/community/conceptual-articles/an-introduction-to-document-oriented-databases](https://digitalocean.com/community/conceptual-articles/an-introduction-to-document-oriented-databases). Accedido el 25 de diciembre de 2023.
- [2] Jain, Sandeep. “How to Install and Configure MongoDB in Ubuntu?” *GeeksforGeeks*, 5 de octubre de 2021, [geeksforgeeks.org/how-to-install-and-configure-mongodb-in-ubuntu/?ref=lbp](https://www.geeksforgeeks.org/how-to-install-and-configure-mongodb-in-ubuntu/?ref=lbp). Accedido el 25 de diciembre de 2023.
- [3] Real Python. “Python and MongoDB: Connecting to NoSQL Databases – Real Python.” *Real Python*, [realpython.com/introduction-to-mongodb-and-python](https://realpython.com/introduction-to-mongodb-and-python). Accedido el 25 de diciembre de 2023.
- [4] Jain, Sandeep. “MongoDB CRUD operations.” *GeeksforGeeks*, 14 de mayo de 2023, [geeksforgeeks.org/mongodb-crud-operations/?ref=lbp](https://www.geeksforgeeks.org/mongodb-crud-operations/?ref=lbp). Accedido el 25 de diciembre de 2023.
- [5] MongoDB. “Connect to a Database Deployment — MongoDB Atlas.” *MongoDB*, [mongodb.com/docs/atlas/connect-to-database-deployment/](https://mongodb.com/docs/atlas/connect-to-database-deployment/). Accedido el 25 de diciembre de 2023.
- [6] MongoDB. “Collection Methods — MongoDB Manual.” *MongoDB*, [mongodb.com/docs/manual/reference/method/js-collection/](https://mongodb.com/docs/manual/reference/method/js-collection/). Accessed 25 de diciembre de 2023.