

# Misión MongoDB NoSQL - El Cronista de Datos NoSQL

---

**Alumna:** Alexandra Tana

**Asignatura:** Modelado Avanzado de Bases de Datos

**Fecha:** Octubre 2025

**Institución:** ESPE

---

## Descripción del Proyecto

Este proyecto implementa un **Bestiario Digital de Criaturas Fantásticas** utilizando MongoDB como base de datos NoSQL. El objetivo es explorar las ventajas del modelo orientado a documentos frente al modelo relacional tradicional, aprovechando la flexibilidad del esquema para almacenar criaturas con atributos heterogéneos y variables.

El bestiario incluye criaturas como dragones, fénix, leviatanes, unicornios y gólems, cada una con características únicas que demuestran la potencia de MongoDB para manejar datos semi-estructurados.

---

## Estructura del Repositorio

```
mision-mongodb-alexandra-tana/
|
├── README.md                # Este archivo - Documentación principal
├── ANALISIS_NOSQL.md        # Análisis comparativo NoSQL vs SQL
└── misiones_mongodb.js      # Script con todas las operaciones CRUD
```

## Requisitos Previos

Para ejecutar este proyecto necesitas:

1. **MongoDB Atlas** (cuenta gratuita) o **MongoDB Community Server** instalado localmente
  2. **MongoDB Shell (mongosh)** instalado
  3. **Visual Studio Code** con la extensión **MongoDB for VS Code** (opcional pero recomendado)
  4. **Git** para clonar el repositorio
- 

## Instalación y Configuración

Paso 1: Clonar el repositorio


```
git clone https://github.com/alexadritana/mision-mongodb-alexandra-tana.git
cd mision-mongodb-alexandra-tana
```

## Paso 2: Configurar MongoDB Atlas

1. Crea una cuenta gratuita en [MongoDB Atlas](#)
2. Crea un cluster gratuito (M0)
3. Configura Database Access (crea un usuario con contraseña)
4. Configura Network Access (permite tu IP o 0.0.0.0/0 para desarrollo)
5. Obtén tu cadena de conexión (Connection String)

## Cómo Ejecutar el Script

### Opción 1: Usando VS Code (Recomendado)

1. Abre el proyecto en Visual Studio Code
2. Instala la extensión "MongoDB for VS Code"
3. Conecta a tu cluster usando la cadena de conexión
4. Abre el archivo `misiones_mongodb.js`
5. Renómbalo temporalmente a `misiones_mongodb.mongodb.js` (para habilitar el playground)
6. Presiona el botón **Play** () que aparece arriba a la derecha
7. Los resultados aparecerán en el panel inferior

### Opción 2: Usando MongoDB Shell (mongosh)

```
# Conéctate a tu cluster
mongosh "tu_cadena_de_conexion_aqui"

# Ejecuta el script
load('misiones_mongodb.js')
```

**Nota:** Reemplaza `tu_cadena_de_conexion_aqui` con tu Connection String de MongoDB Atlas.

## Operaciones Implementadas

El script `misiones_mongodb.js` realiza las siguientes operaciones:

### ☒ CREATE (Inserción)

- `insertOne()` - Inserta el Dragón de Fuego
- `insertMany()` - Inserta 4 criaturas adicionales (Fénix, Leviatán, Unicornio, Gólem)

### ☒ READ (Lectura)

- Consulta todas las criaturas del bestiario
- Filtra criaturas por habitat específico (Bosque Encantado)
- Encuentra criaturas con nivel de peligro mayor a 8

## ☑ UPDATE (Actualización)

- `updateOne()` - Añade nueva habilidad al Dragón de Fuego
- `updateMany()` - Incrementa el nivel de peligro de criaturas del Bosque Encantado

---

## 🐉 Ejemplo de Criatura

```
{
  nombre: "Dragón de Fuego",
  habitat: "Montañas Volcánicas",
  nivel_peligro: 10,
  dieta: ["rocas", "caballeros"],
  habilidades: ["vuelo", "aliento de fuego", "garras afiladas"],
  estadisticas: {
    ataque: 95,
    defensa: 80,
    velocidad: 70
  },
  descubierto_por: "Eldrin el Valiente",
  fecha_descubrimiento: new Date("2024-01-15")
}
```

---

## 📖 Documentación Adicional

Para entender las decisiones de diseño y el análisis comparativo entre MongoDB y bases de datos relacionales, consulta:

- [ANALISIS\\_NOSQL.md](#) - Análisis detallado NoSQL vs SQL, otros tipos de bases de datos NoSQL, y casos de estudio reales

---

## 🔧 Tecnologías Utilizadas

- **MongoDB Atlas** - Base de datos NoSQL en la nube
- **MongoDB Shell (mongosh)** - Cliente de línea de comandos
- **Visual Studio Code** - Editor de código
- **MongoDB for VS Code** - Extensión para desarrollo
- **Git & GitHub** - Control de versiones

---

## 📝 Commits Semánticos

Este proyecto sigue la convención de commits semánticos:

- `feat:` - Nueva funcionalidad
- `docs:` - Cambios en documentación
- `refactor:` - Refactorización de código

- **chore:** - Tareas de mantenimiento
- 



## Autora

### Alexandra Tana

Estudiante de Ingeniería en Tecnologías de la Información

ESPE - Escuela Politécnica del Ejército

---



## Licencia

Este proyecto es parte de una actividad académica para la asignatura de Modelado Avanzado de Bases de Datos.

---



## Enlaces Útiles

- [Documentación oficial de MongoDB](#)
  - [MongoDB University \(cursos gratuitos\)](#)
  - [MongoDB Atlas](#)
- 

★ Si este proyecto te fue útil, no olvides darle una estrella en GitHub