**PROGRAMA DE INGENIERIA DE SISTEMAS**

**OCTAVO SEMESTRE**

**Proyecto sobre una tienda de peliculas - MongoDB**

**Integrantes:** Oneyda Pérez

Milena Tisoy

**Docente:** Brayan Arcos

**Asignatura:** Base de Datos y Almacenamiento Masivo

Tabla de contenido

[Resumen ejecutivo 3](#_Toc176870872)

[Introducción 3](#_Toc176870873)

[Contexto y motivación: 3](#_Toc176870874)

[Alcance del informe 3](#_Toc176870875)

[Objetivos 3](#_Toc176870876)

[Metodología 3](#_Toc176870877)

[Herramientas utilizadas: 3](#_Toc176870878)

[Procedimientos: 3](#_Toc176870879)

[Desarrollo del informe 4](#_Toc176870880)

[Descripción de la base de datos 4](#_Toc176870881)

[Esquema de la base de datos 4](#_Toc176870882)

[Consultas NoSQL 4](#_Toc176870883)

[Consultas realizadas 4](#_Toc176870884)

[Resultado de consultas 4](#_Toc176870885)

[Explicación de consultas 5](#_Toc176870886)

[Diseño de base de datos 5](#_Toc176870887)

[Consideraciones de diseño 5](#_Toc176870888)

[Análisis y discusión 5](#_Toc176870889)

[Conclusiones 5](#_Toc176870890)

[Recomendaciones 5](#_Toc176870891)

# **Resumen ejecutivo**

Este informe presenta el análisis de una base de datos MongoDB para una tienda de películas, enfocándose en consultas NoSQL, se destaca la estructura de las colecciones, las consultas realizadas, y la interpretación de los resultados obtenidos.

# **Introducción**

## **Contexto y motivación:**

la gestión eficiente de datos sobre clientes, películas, empleados y venta es esencial. MongoDB ofrece flexibilidad para manejar volúmenes de datos grandes no estructurados, lo que lo hace ideal para ser aplicado en este tipo de ejercicios.

## **Alcance del informe**

Aspectos como la estructura de la base de datos, consultas NoSQL, y optimizar.

## **Objetivos**

El objetivo principal es demostrar como es el trabajo que se puede desarrollar en MongoDB para la gestión de información en este caso una tienda de películas, y realizar consultas eficientes.

# **Metodología**

## **Herramientas utilizadas:**

* MongoDB como base de datos NoSQL

## **Procedimientos:**

* Definir las colecciones: *Clientes, Empleados, Películas, Ventas*
* Ejecución de consultas NoSQL, para encontrar información especifica

# **Desarrollo del informe**

## **Descripción de la base de datos**

### **Esquema de la base de datos**

* **Clientes:** almacena datos como nombre, correo, teléfono
* **Empleados:** diseñado para almacenar información sobre los empleados como id, nombre y el cargo que desempeña
* **Películas:** en esta colección contiene una lista de películas como de sus atributos que pueden ser el título, genero y el precio de la película.
* **Ventas**: registra la compra enlazada a clientes y películas.

## **Consultas NoSQL**

### **Consultas realizadas**

* Listar los clientes que han comprado la película con título
* Obtener el nombre de los empleados que han realizado ventas superiores a determinado valor.
* Obtener todas las películas que han sido vendidas
* Consultar todas las películas de un género determinado
* Contar cuántas ventas fueron realizadas por un determinado empleado.

### **Resultado de consultas**

* Se realizo una búsqueda en la colección VENTAS, cruzando con PELICULAS para obtener los clientes que compraron una película en particular, usando **$lookup** para unir las colecciones de ventas y películas, donde se listan los nombres de los clientes que comprar la película.
* En la colección VENTAS, se utiliza el operador **$gte** (mayor o igual que) y se relaciona con la colección EMPLEADOS, para listar los empleados que realizaron ventas superiores a, por ejemplo 9.00.
* En la colección PELICULAS se utiliza **$match** para filtrar el campo de genero y listar las el género de las películas.
* Se utiliza el operador **$group** en la colección VENTAS, agrupando por el id del empleado, para mostrar el número total de ventas realizadas.

### **Explicación de consultas**

Se utilizó la sintaxis de MongoDB basada en documentos JSON, aplicando operadores como **$match y $group** para filtrar y agrupar datos.

## **Diseño de base de datos**

### **Consideraciones de diseño**

* Clave primaria: Usando **\_id** autogenerado para cada documento.
* Colección Ventas: Incluye referencias cruzadas a clientes y películas para optimizar consultas relacionadas con transacciones.
* Uso de índices en campos clave como el nombre de películas y fecha de transacción.

# **Análisis y discusión**

Estas consultas reflejan la flexibilidad de MongoDB para gestionar y consultar datos complejos, como ventas asociadas a múltiples colecciones (películas, clientes, empleados). La implementación de operaciones como **$lookup, $match y $group** facilita la extracción de información útil, demuestra su flexibilidad y eficiencia para manejar bases de datos con relaciones de tipo NoSQL, lo que mejora la toma de decisiones en la gestión de la tienda.

# **Conclusiones**

MongoDB se muestra eficaz para gestionar una base de datos compleja de una tienda de películas, donde la flexibilidad de su esquema permite optimizar tanto el almacenamiento como la ejecución de consultas específicas. El uso de agregaciones ha facilitado la ejecución de operaciones complejas, como la identificación de empleados con mejor rendimiento y el análisis de películas más populares.

# **Recomendaciones**

* La consulta sobre empleados con ventas superiores permite identificar a los mejores vendedores, lo que puede servir para análisis de rendimiento.
* El filtrado por género de películas mejora la gestión de inventario y las recomendaciones de productos.
* El conteo de ventas por empleado apoya la evaluación de productividad.