

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA
FACULTAD DE INGENIERÍA
ESCUELA DE CIENCIAS Y SISTEMAS
LABORATORIO DE BASES DE DATOS 2
SEGUNDO SEMESTRE 2023



GRUPO 18

Nombre	Carnet
Javier Alejandro Gutierrez de León	202004765
Enrique Alejandro Pinula Quiñonez	202004707
Gerson Rubén Quiroa del Cid	202000166

CRUD Libros:

1. Ver libros: 1.204 segundos.
2. Crear libro: 1.072 segundos.
3. Actualizar libro: 0.805 segundos.
4. Eliminar libro: 1.120 segundos.

Análisis: Como se puede observar son tiempos muy pequeños, esto se debe a que son pocos datos, eso lo vuelve más rápido. También podemos notar que la consulta que más se tardó en realizar fue listar los libros ya que requiere de traer todos los libros que haya en la colección y mostrar todos.

Por otro lado, la operación que menor tiempo tomó fue actualizar un libro. Ya que se buscó por id y fueron poco datos actualizados. Sin embargo, todas las operaciones fueron realizadas rápidamente.

Cabe mencionar que la velocidad de las operaciones se debe a que se está utilizando una base de datos NoSQL.

CRUD Películas:

1. Ver películas: 698 ms.
2. Crear película: 862 ms.
3. Actualizar película: 1.078 segundos.
4. Eliminar película: 1.196 segundos.

Análisis: Los tiempos obtenidos son bastante pequeños, en especial para visualizar los datos y crearlos. Esto es debido a ser una base de datos NoSQL ya que se enfoca en obtener los datos de manera rápida sin importar otros aspectos como la redundancia de los datos.

Por otro lado, tanto la actualización como la eliminación de datos tardaron un poco más que las demás operaciones, esto debido a que para estas operaciones primero es necesario buscar el objeto como tal, mediante su ID.

Cabe destacar que los tiempos pueden aumentar ligeramente mientras más registros tenga la base de datos, ya que las pruebas fueron realizadas con pocos datos ingresados.

Consultas Libros:

1. Listar todos los libros disponibles: 374 ms.
2. Encontrar libros por Categoría (por ejemplo, Ciencia Ficción): 248 ms.
3. Buscar libros con un autor específico: 167 ms.
4. Mostrar los libros ordenados por calificación promedio (de mayor a menor): 181 ms.
5. Encontrar libros con un precio inferior a 20. 153 ms.
6. Buscar libros con una palabra clave en el título o descripción. 173 ms.
7. Información de los 10 autores más caros (Suma del precio de todos sus libros). 178 ms.
8. Obtener la cantidad de libros en stock para un libro específico: 140 ms.
9. Calcular el precio promedio de todos los libros: 147 ms.
10. Información de todas las categorías. 213 ms.

Análisis: los tiempos de consulta son bastante buenos en general, lo que sugiere que la base de datos de MongoDB está bien diseñada y optimizada para responder eficazmente a la variedad de consultas. Sin embargo, es importante tener en cuenta que estos tiempos pueden variar según el hardware, la cantidad de datos y la carga del servidor de la base de datos.

Consultas Películas:

1. Listar todas las películas disponibles. 814 ms.
2. Buscar películas por género (por ejemplo, Comedia). 815 ms.
3. Mostrar películas con una clasificación R (restringida) o superior. 820 ms.
4. Encontrar películas dirigidas por un director específico. 766 ms.
5. Buscar películas con un precio inferior a 15. 766 ms.
6. Recuperar películas lanzadas en un año específico (por ejemplo, 2022): 745 ms.
7. Información de los 10 directores con mejor calificación (Promedio de todas sus películas): 729 ms.
8. Buscar películas con una palabra clave en el título o descripción: 629 ms.
9. Calcular el precio promedio de todas las películas: 658 ms.
10. Encontrar películas con las mejores calificaciones promedio (ordenadas de mayor a menor): 905 ms.

Análisis: En comparación con el análisis anterior de la colección de datos de libros en MongoDB, podemos notar algunas diferencias significativas en los tiempos de consulta para la colección en DynamoDB. En resumen, los tiempos de consulta en DynamoDB son en general más largos en comparación con MongoDB en el análisis

anterior. Esto puede deberse a diferencias en la estructura y capacidad de indexación de las bases de datos, así como a la complejidad de las consultas realizadas. Optimizar la estructura de datos y las consultas podría ayudar a mejorar la eficiencia de DynamoDB en este contexto.