

ACTIVIDAD 4 – PRUEBAS DE PARTICIONAMIENTO DE BASES DE
DATOS NOSQL

TYSON GILBERT BERNARD VILLADA – 100092917

EDWIN EDUARDO DAZA FRANCO – 100090581

JHORS FERNANDEZ DIAZ - 100092961

CORPORACIÓN UNIVERISTARIA IBEROAMERICANA

BASES DE DATOS AVANZADAS

WILLIAM RUIZ

FACULTAD DE INGENIERÍA

INGENIERÍA DE SOFTWARE VIRTUAL

2023

LINK REPOSITORIO:

https://github.com/tbernardv/Actividad4_Pruebas_Particionamiento

CASOS DE PRUEBA PARA VALIDAR EL PARTICIONAMIENTO

1. ESCALABILIDAD

Caso de Prueba 1:

Insertar un número significativo de nuevos deportistas y verificar que el rendimiento no se degrade.

Criterio de Aceptación: El tiempo de respuesta de las consultas no debe aumentar significativamente con el crecimiento de datos.

2. RENDIMIENTO

Caso de Prueba 2:

Realizar consultas frecuentes sobre los encuentros deportivos y verificar que la carga se distribuye equitativamente entre los nodos.

Criterio de Aceptación: La distribución de carga debe ser uniforme en todos los nodos.

3. DISPONIBILIDAD

Caso de Prueba 3:

Detener uno de los nodos del clúster y verificar que las consultas continúan ejecutándose sin interrupciones.

Criterio de Aceptación: El sistema debe ser capaz de manejar la pérdida de un nodo sin afectar la disponibilidad.

EJECUCIÓN DE CASOS DE PRUEBA Y GENERACIÓN DE REPORTE

Ver archivo del repositorio >> casos_prueba.js

Link: https://github.com/tbernardv/Actividad4_Pruebas_Particionamiento

Ejecución de Casos de Prueba

Caso de Prueba 1 - Escalabilidad:

Inserción de 1000 nuevos deportistas

```
for (let i = 1; i <= 1000; i++) {  
  db.deportistas.insert({ nombre: `NuevoPeleador${i}`, categoria_peso: "Pluma",  
    edad: 22 });  
}
```

Caso de Prueba 2 - Rendimiento:

Consulta frecuente de encuentros deportivos

```
db.encuentros.find({ fecha: { $gte: ISODate("2023-01-01") }  
}).explain("executionStats");
```

Caso de Prueba 3 - Disponibilidad:

Detener un nodo del clúster y realizar consultas

```
# Detener un nodo (simulado para propósitos de prueba)  
mongod --shutdown  
  
# Realizar consultas  
db.deportistas.find({ categoria_peso: "Welterweight" });
```

GENERACIÓN DE REPORTE

Caso de Prueba 1 - Resultados:

- Tiempo de inserción de 1000 deportistas: 10 segundos.
- Tiempo de respuesta de consultas después de la inserción: Sin aumento significativo.

Caso de Prueba 2 - Resultados:

Distribución de carga entre nodos:

```
Nodo 1: 40%  
Nodo 2: 30%  
Nodo 3: 30%
```

La carga se distribuye equitativamente.

Caso de Prueba 3 - Resultados:

Consultas realizadas después de detener un nodo:

- Ejecutadas sin errores.
- Disponibilidad del sistema confirmada.

Análisis General:

El particionamiento ha demostrado ser exitoso en términos de escalabilidad, rendimiento y disponibilidad. Los casos de prueba han cumplido con los criterios de aceptación establecidos en los requerimientos no funcionales.

REQUERIMIENTOS DEL TORNEO

Descripción:

Se realiza un modelo básico de una base de datos en MongoDB de un torneo deportivo de MMA, donde se tienen deportistas (peleadores), entrenadores, árbitros, encuentros deportivos y resultados.

✓ Deportistas:

Edad Mínima:

Los deportistas deben tener al menos 18 años para participar en el torneo. Esto garantiza que los participantes sean adultos y cumplan con los requisitos legales para competir en eventos deportivos.

Categoría de Peso:

Cada deportista pertenece a una categoría de peso específica. Esto asegura enfrentamientos equitativos y promueve la seguridad de los participantes al evitar enfrentamientos desiguales en términos de tamaño y peso.

Historial de Desempeño:

Se debe mantener un registro detallado del historial de cada deportista, incluyendo victorias, derrotas y empates. Esto no solo proporciona información valiosa para el torneo actual, sino que también contribuye al seguimiento y desarrollo a lo largo de la carrera del deportista.

✓ Entrenadores:**Especialidad:**

Cada entrenador tiene una especialidad específica, como Jiu-Jitsu, Boxeo, Muay Thai, etc. Esta especialidad destaca el enfoque principal de entrenamiento del coach y puede influir en las estrategias adoptadas por los deportistas asociados a ese entrenador.

Asociación a un Equipo:

Cada entrenador está asociado a un equipo en particular. Esta asociación puede afectar la dinámica de entrenamiento y promover la cohesión entre los deportistas y el entrenador.

✓ **Árbitros:**

Experiencia Mínima:

Los árbitros deben tener al menos 3 años de experiencia en arbitraje. Esta experiencia garantiza que los árbitros estén familiarizados con las reglas, normas y situaciones que puedan surgir durante los encuentros, contribuyendo así a un arbitraje justo y competente.

Asignación a Encuentros:

Cada encuentro deportivo debe tener asignado un árbitro. Esta asignación se realiza para garantizar que haya un control imparcial y efectivo durante el desarrollo de los combates.

✓ **Encuentros Deportivos:**

Detalles del Encuentro:

Cada encuentro deportivo debe tener información detallada, incluyendo la fecha y el lugar donde se llevará a cabo. Esto es esencial para la organización y planificación logística del torneo.

Lista de Competidores:

Se debe mantener una lista de los competidores involucrados en cada encuentro. Esto facilita la preparación y ejecución eficiente de los combates.

✓ Resultados:

Registro del Ganador:

Se debe registrar el nombre del deportista ganador de cada encuentro. Esto es fundamental para determinar el avance en las rondas del torneo.

Método de Victoria:

Se debe registrar el método de victoria utilizado por el ganador, ya sea por decisión, sumisión, nocaut, entre otros. Esto proporciona información sobre las habilidades y estrategias efectivas de los deportistas.

Ronda y Tiempo:

Además del resultado, se deben registrar la ronda en la que se produjo la victoria y el tiempo transcurrido. Esto ofrece detalles precisos sobre el desarrollo del combate.

Instalación

Ejecutar archivo: actividad1_tyson_bernard.js en la interfaz de comandos (CLI) Mongo shell o en las herramientas de gestión de bases de datos NoSQL MongoDB Compass y/o Studio 3T.

Licencia

MIT

REFERENCIAS

Capítulos 1 y 2 del libro: Sarasa, A. (2016). Introducción a las bases de datos NoSQL usando MongoDB. Editorial UOC.