

DOCUMENTO DE REQUERIMIENTOS NO FUNCIONALES

Introducción

El objetivo de este documento es definir los requerimientos no funcionales para la implementación de un sistema de particionamiento (sharding) en la base de datos MongoDB utilizada para gestionar la información de la Premier League. Estos requerimientos se enfocan en garantizar la calidad del sistema en términos de redundancia, disponibilidad, rendimiento y balanceo de carga.

Requerimientos No Funcionales

1. Redundancia y Disponibilidad 24x7

1. Redundancia:

- La base de datos debe estar configurada de manera que cada shard tenga réplicas para asegurar la redundancia de los datos.
- En caso de fallo de un nodo, los datos deben ser accesibles desde las réplicas sin pérdida de información.

2. Disponibilidad 24x7:

- El sistema debe estar disponible en todo momento, permitiendo operaciones de lectura y escritura sin interrupciones.
- Debe existir un mecanismo de failover automático para minimizar el tiempo de inactividad.

2. Estrategia de Particionamiento

1. Distribución de Datos:

- Los datos deben ser distribuidos equitativamente entre al menos 2 shards para asegurar un balance de carga óptimo.
- La clave de particionamiento debe ser seleccionada de manera que permita una distribución uniforme de los datos.

2. Replicación de Shards:

- Cada shard debe tener un conjunto de réplicas configurado, siguiendo un esquema de replicación similar al de un clúster centralizado.
- Las réplicas deben estar ubicadas en diferentes nodos para asegurar la redundancia geográfica.

3. Carga de Datos

1. Tiempo de Inserción de Datos:

- El tiempo de inserción de datos en la colección de equipos debe ser inferior a un segundo por operación.
- Se deben realizar pruebas de carga para validar este requerimiento.

4. Tiempos de Respuesta

1. Tiempo de Respuesta para Consultas:

- Las consultas sobre las colecciones de la base de datos deben tener un tiempo de respuesta inferior a un segundo.
- Este tiempo de respuesta debe ser consistente bajo condiciones de carga normal y moderada.

5. Balanceo de Carga de Datos

1. Funcionamiento del Balanceador de Carga:

- El sistema debe distribuir las inserciones masivas de datos de manera equitativa entre los diferentes nodos.
- Debe ser posible verificar que el balanceador de carga está funcionando correctamente a través de métricas de uso y distribución de datos.

Casos de Pruebas

1. Verificación de Particionamiento:

- Objetivo: Asegurar que los datos se han dividido físicamente en al menos 2 shards y que cada shard tiene sus réplicas.
- Prueba: Realizar consultas que verifiquen la distribución de datos entre shards y la existencia de réplicas.

2. Carga de Datos:

- Objetivo: Validar que el tiempo de inserción de datos en la colección de equipos es inferior a un segundo.
- Prueba: Insertar un documento en la colección de equipos y medir el tiempo de respuesta.

3. Tiempos de Respuesta para Consultas:

- **Objetivo:** Validar que el tiempo de respuesta para consultas sobre las colecciones es inferior a un segundo.
- **Prueba:** Realizar consultas sobre las colecciones y medir los tiempos de respuesta.

4. Validación del Balanceo de Carga de Datos:

- **Objetivo:** Asegurar que el balanceador de carga distribuye los datos equitativamente entre los nodos durante inserciones masivas.
- **Prueba:** Realizar inserciones masivas y verificar la distribución de datos y la carga en cada nodo.