# Conceptos y comandos básicos de la replicación en bases de datos NoSQL.

Actividad 2 – Unidad 2

**Nombre de los estudiantes**

Santiago Herrera Rocha

**Docente**

Jorge Isaac Castañeda Valbuena

CORPORACIÓN UNIVERSITARIA IBEROAMERICANA

INGENIERÍA DE SOFTWARE, FACULTAD DE INGENIERÍA

BASES DE DATOS AVANZADAS

Bogotá, D.C.

01 de diciembre de 2024

# Requerimientos no funcionales y casos de prueba

## Requerimientos no funcionales

## Redundancia:

**Descripción:**

* El sistema debe garantizar la disponibilidad de los datos del torneo en al menos tres nodos replicados en MongoDB.
* Las réplicas deben reflejar los datos del nodo primario en tiempo real, asegurando consistencia en todas las instancias.

**Criterios Técnicos:**

* Configuración de un conjunto de réplicas en MongoDB con una instancia primaria y dos secundarias.
* Todos los nodos deben estar sincronizados. Es decir, los nodos secundarios reflejan los datos primarios en tiempo real.

## Disponibilidad 24x7:

**Descripción:**

* En caso de falla del nodo primario, el sistema debe garantizar continuidad operativa mediante failover automático.
* En caso de falla del primario, un secundario asumirá el rol primario en menos de 10 segundos.

**Criterios Técnicos:**

* Habilitación del mecanismo de failover automático en el conjunto de réplicas.
* Inserciones y lecturas disponibles en el nuevo nodo primario tras la transición de un nodo a otro.

## Casos de prueba para redundancia y disponibilidad

## Caso de prueba 1: Verificar replicación exitosa Objetivo: Comprobar que los datos insertados en el primario son replicados a los secundarios.

**Pasos:**

1. Insertar un documento en el nodo primario.
2. Verificar su existencia en los nodos secundarios utilizando comandos de consulta.

**Resultado Esperado:** El documento debe aparecer en los nodos secundarios.

## Caso de prueba 2: Simular failover automático Objetivo: Verificar que el mecanismo de failover automático funciona correctamente.

**Pasos:**

1. Detener el nodo primario.
2. Observar cuál nodo secundario asume el rol de primario.

**Resultado Esperado:** Un nodo secundario debe convertirse en primario en menos de 10 segundos.

## Caso de prueba 3: Verificar consistencia de datos después del failover.

**Objetivo:** Garantizar que la replicación continúa tras la transición del primario.

**Pasos:**

1. Insertar un nuevo documento en el nodo que asumió como primario tras el failover.
2. Verificar la replicación del documento en los otros nodos secundarios

**Resultado Esperado:** Los nodos secundarios deben reflejar los nuevos datos insertados.