SRI SIVASUBRAMANIYA NADAR COLLEGE OF ENGINEERING, KALAVAKKAM
(An Autonomous Institution, Affiliated to Anna University, Chennai)
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
III Year CSE (V Semester)
Academic Year 2026-27
UCS3513 SYSTEM DESIGN LABORATORY

Lab Exercise 5 - Design and simulate Distributed Data Routing using Consistent Hashing

Objective

To deploy multiple MongoDB instances using Docker, implement the Consistent Hashing algorithm in a Spring Boot application, distribute student records across multiple storage nodes, and analyze data migration and scalability in a distributed storage system.

Problem Statement

A university's Student Information System stores a large number of student records. As the volume of data grows, storing all records in a single MongoDB server leads to increased storage requirements and reduced system availability.

To improve scalability and minimize data migration during system expansion, deploy multiple MongoDB instances using Docker. Implement the Consistent Hashing algorithm in a Spring Boot application to distribute the Student collection across multiple storage nodes. Add and remove storage nodes dynamically, redistribute only the affected records, and analyze the effectiveness of Consistent Hashing in a distributed database system.

Tasks

- Deploy three independent MongoDB instances using Docker.
- Develop a Spring Boot application implementing the Consistent Hashing algorithm.
- Register all MongoDB instances on the hash ring using virtual nodes.
- Route Student records to the appropriate MongoDB instance using the generated hash value.
- Retrieve Student records through the Spring Boot application.
- Add a new MongoDB storage node and update the hash ring.
- Migrate only the affected student records to the newly added storage node.
- Remove an existing MongoDB storage node from the hash ring.
- Redistribute only the affected student records to the remaining storage nodes.
- Compare the data distribution before and after adding or removing storage nodes.

Analysis

1. Explain the role of:
    - Hash Ring
    - Virtual Nodes
    - MongoDB Storage Node

2. Explain how Consistent Hashing determines the storage location of a student record.
3. Explain what happens when a new MongoDB storage node is added.
4. Explain what happens when an existing MongoDB storage node is removed.
5. Compare modulo-based hashing with Consistent Hashing.
6. Compare Application-Level Consistent Hashing with MongoDB Horizontal Sharding.

Expected Learning Outcomes

After completing this experiment, students will be able to:

- Deploy multiple MongoDB instances using Docker.
- Develop distributed applications using Spring Boot.
- Implement the Consistent Hashing algorithm using a hash ring.
- Route data dynamically across multiple storage nodes.
- Analyze data migration during node addition and removal.
- Understand the architecture of application-level distributed storage.
- Explain the scalability benefits of Consistent Hashing in distributed systems.