Index: Experimental comparison between GraalVM with Micronaut and Spring Boot in a microservices architecture: performance analysis and load testing

1. **Chapter 1: Overview of Microservices Architecture**
   - 1.1. **Introduction to Microservices Architecture**
     - 1.1.1. Definition and fundamental principles
     - 1.1.2. Modularity and service independence
     - 1.1.3. Scalability, resilience, and maintainability
     - 1.1.4. Use cases in real-world scenarios
   - 1.2. **Differences with Monolithic Architecture**
     - 1.2.1. Structure and characteristics of a monolithic application
     - 1.2.2. Conceptual comparison with microservices: advantages and disadvantages
   - 1.3. **Microservices Architecture in Real-World Context**
     - 1.3.1. Architectural patterns and integration mechanisms in microservices
     - 1.3.2. Orchestration and containerization with Docker and Kubernetes
   - 1.4. **Data Consistency and Communication between Microservices**
     - 1.4.1. Consistency challenges in distributed systems
     - 1.4.2. Communication methods: synchronous and asynchronous
     - 1.4.3. Design considerations on consistency, reliability, and scalability

2. **Chapter 2: Introduction to Micronaut Technology with GraalVM**
   - 2.1. **Introduction and Motivations for the Comparison**
     - 2.1.1. Motivations for comparing Spring Boot and Micronaut
     - 2.1.2. Use cases and philosophical differences between the two frameworks
   - 2.2. **Spring Boot Application**
     - 2.2.1. Internal architecture and convention-over-configuration philosophy
     - 2.2.2. Reflection-based Dependency Injection
     - 2.2.3. Typical structure of a microservices application
     - 2.2.4. Startup efficency and resource management
   - 2.3. **Micronaut and the Emergence of a New Paradigm**
     - 2.3.1. Origins of Micronaut: context and objectives
     - 2.3.2. Compile-time Dependency Injection, AOT, and reduction of reflection
     - 2.3.3. Modularity and native support for GraalVM
   - 2.4. **GraalVM and Ahead-of-Time Compilation**
     - 2.4.1. Overview of GraalVM and differences compared to the traditional JVM
     - 2.4.2. Mechanism of AOT compilation
     - 2.4.3. Benefits in terms of startup time and resource usage in microservices
   - 2.5. **Theoretical Comparative Analysis**
     - 2.5.1. Architectural and performance differences between Spring Boot and Micronaut with GraalVM
     - 2.5.2. Theoretical comparative analysis of performance
     - 2.5.3. Impact on performance in containerized environments

3. **Chapter 3: Middleware Architecture and Design**
   - 3.1. **Objectives of the Developed Middleware**
     - 3.1.1. Purpose and main functionalities of the system
     - 3.1.2. Role of the middleware in communication between two systems
   - 3.2. **System Architecture**
     - 3.2.1. Microservices structure: description of modules and interactions

    3.2.2. Architectural diagram and data flow
    3.2.3. Database management and versioning
3.3. **Microservices Implementation and Comparisons**
    3.3.1. Project structure and dependencies
    3.3.2. Controller development and API analysis
    3.3.3. Exception handling
    3.3.4. Service layer
    3.3.5. Repository layer
3.4. **Deployment and Infrastructure**
    3.4.1. Deployment on Oracle Cloud
    3.4.2. Monitoring and observability tools

4. **Chapter 4: Metrics and Experimental Methodology**
4.1. **Definition of Metrics**
    4.1.1. Analysis of the metrics used
    4.1.2. Rationale for the choice of each metric
4.2. **Testing and Data Collection Tools**
    4.2.1. Apache JMeter: configuration and stress test scenarios
    4.2.2. Prometheus and Grafana for real-time monitoring
4.3. **Testing Methodology**
    4.3.1. Test environment: hardware and software configuration
    4.3.2. Procedures to ensure repeatability and consistency of results

5. **Chapter 5: Experimental Analysis and Results**
5.1. **Test Scenarios**
    5.1.1. Description of simulated workloads and operational conditions
5.2. **Performance Test Results**
    5.2.1. Presentation of collected data
    5.2.2. Quantitative comparison between Spring Boot and Micronaut with GraalVM
5.3. **Discussion of Results**
    5.3.1. Data interpretation
    5.3.2. Analysis of the causes of observed differences
    5.3.3. Considerations on scalability, stability, and resource consumption

6. **Chapter 6: Conclusions and Future Work**
6.1. Summary of experimental results
6.2. Final considerations
6.3. Future developments