

BAB I

PENDAHULUAN

1.1 Latar Belakang

Era digitalisasi saat ini perkembangan teknologi tidak hanya di bidang industri. Perkembangan di era 4.0 sangat mengandalkan kecepatan informasi. Aplikasi *monolithic* yang telah berkembang selama bertahun-tahun dapat menjadi besar, kompleks, dan pada tahap selanjutnya bahkan memfossil, yang berarti akumulasi teknis menghasilkan struktur yang tidak jelas yang membuat produk tidak dapat dipertahankan dengan upaya yang wajar. Struktur informasi dari setiap generasi akan terus mengalami modifikasi dan perkembangan. Perbaikan struktur arsip dimulai dari luasnya keinginan jaringan yang berdampak pada semakin cepatnya perangkat arsip mengalami perbaikan. *Microservices* adalah ide di mana paket dibagi menjadi elemen praktis kecil yang dapat tepat dan tidak bias dari aditif yang berbeda, dengan penerapan API agar jika satu perangkat gagal, sekarang tidak lagi berpengaruh pada aditif perangkat yang berbeda dan terus bekerja.

Ketahanan *Microservices* adalah isolasi jika salah satu perangkat gagal, kegagalan tersebut sekarang tidak akan lagi berpengaruh pada kinerja keseluruhan dari aditif alternatif. Dengan demikian, kerumitan dapat diredam dan relaksasi dari aditif perangkat alternatif dapat tetap berfungsi. Ketahanan cara tahan atau mampu beradaptasi dengan modifikasi selain kesalahan atau membahayakan infrastruktur. Dalam studi struktur arsip, ketahanan sering terlihat sebagai kebutuhan non-praktis yang dapat dilihat dari faktor *skalabilitas*, keandalan, rawatan, dan ketersediaan. Kelanjutan dari penelitian adalah dalam bentuk pemodelan dan pembuatan bukti ide modifikasi struktur perangkat lunak yang dialokasikan, terutama didasarkan sepenuhnya pada *microservices* dan *Docker*, dengan arsitektur *Microservices - Docker* yang dikembangkan dengan perangkat *framework Spring Boot*. Penggunaan *microservice* dibicarakan bahwa lebih cepat dan beban kerja sistem lebih ringan.

Pada tahun 2020, dilakukan penelitian oleh Nabor C. Mendonca, Carlos M. Aderaldo, Javier C'amara, dan David Garlan dengan judul "*Model-Based Analysis of Microservice Resiliency Patterns*". Hasilnya, pendekatan ini memungkinkan untuk mengukur dampak penerapan setiap pola ketahanan pada berbagai atribut kualitas dan menentukan cara terbaik untuk menyesuaikan parameternya dalam konteks skenario interaksi layanan klien yang sederhana.

Pada tahun 2021, Sebastian Frank, Alireza Hakamian, Lion Wagner, Dominik Kesim, Jóakim von Kistowski, dan André van Hoorn melakukan penelitian dengan judul "*Scenario-based Resilience Evaluation and Improvement of Microservice Architectures: An Experience Report*" yang mengidentifikasi 12 skenario ketahanan. Peneliti secara kuantitatif mengevaluasi persyaratan ketahanan dan menyarankan peningkatan ketahanan dalam ruang lingkup kedua skenario menggunakan ChaosToolkit. Pada tahun yang sama, Sanjana G B dan Prof. Girish Rao Salanke N S. melakukan penelitian dengan judul "*High Resilient Messaging Service for Microservice Architecture*" yang berhasil mengurangi kode yang diperlukan untuk integrasi di sisi produsen dan konsumen.

Pada tahun 2019, David Gesvindr, Jaroslav Davidek, dan Barbora Buhnova melakukan penelitian dengan judul "*Design of Scalable and Resilient Applications using Microservice Architecture in PaaS Cloud*" dan mengukur dampaknya pada aplikasi sampel.

Pada tahun 2022, Matteo Camilli dan Barbara Russo melakukan penelitian dengan judul "*Modeling Performance of Microservices Systems with Growth Theory*" yang menunjukkan bahwa model pertumbuhan berbentuk S non-linear menangkap terjadinya pelanggaran kinerja layanan mikro dengan akurasi tinggi. Namun, kekurangan dari beberapa penelitian sebelumnya adalah tidak membahas tentang performa *throughput*, *response time*, dan *CPU usage* di dalam *resiliensi* atau ketahanan.

Pengembangan penelitian ini penulis melakukan penerapan *web service* menggunakan arsitektur *microservices* dengan *framework Spring Boot* di dalam *docker* kontainer untuk *resiliensi* sistem *e-commerce*. Penerapan dengan

framework ini di harapkan pengembangan terhadap performa *throughput*, *response time*, dan *CPU usage* dapat meningkatkan kualitas arsitektur *microservices* menggunakan *framework Spring Boot* yang menggunakan bahasa pemrograman *java*. penelitian ini berjudul "ARSITEKTUR MICROSERVICE UNTUK RESILIENSI SISTEM WEB SERVICE E-COMMERCE".

1.2 Rumusan Masalah

Rumusan dari permasalahan yang telah dikemukakan, maka penulis menyimpulkan rumusan masalah dalam penelitian ini sebagai berikut :

1. Bagaimana membuat model desain arsitektur *microservices* pada *e-commerce* ?
2. Bagaimana kinerja sistem *e-commerce* berbasis *microservice* berdasarkan indikator *throughput*, *response time*, dan *CPU usage* untuk menangani jumlah *trafik* yang tinggi ?

1.3 Tujuan Penelitian

Tujuan penelitian berdasarkan rumusan masalah diatas adalah:

1. Menghasilkan model desain arsitektur *microservices* pada *e-commerce*.
2. Untuk mengetahui kinerja sistem *e-commerce* berbasis *microservice* berdasarkan indikator *throughput*, *response time*, dan *CPU usage* untuk menangani jumlah *trafik* yang tinggi.

1.4 Manfaat Penelitian

Adapun manfaat penelitian ini adalah :

- 1 Hasil akhir dari penelitian ini diharapkan dapat membantu mahasiswa dan tenaga kerja dalam menerapkan Untuk pengembangan arsitektur *microservices* di dalam *e-commerce*.
- 2 Hasil dari pengamatan ini diharapkan dapat memberikan kontribusi untuk pembanding penelitian selanjutnya terhadap hasil pengujian performa pada arsitektur *microservices* dengan *framework Spring Boot*.

1.5 Batasan Masalah

Batasan masalah dalam penelitian ini antara lain:

- 1) Penelitian Bahasa *pemograman* yang digunakan dalam penelitian ini yaitu JAVA.
- 2) Parameter yang di ujikan dalam penelitian ini adalah *throughput*, *response time*, dan *CPU usage*.
- 3) Penelitian ini akan membatasi pengujian pada parameter "*thread*" dengan nilai 500, 2500, 4500, 6500, 8500, dan 10.000 (pada *localhost*). Parameter-parameter lainnya tidak akan diuji dalam penelitian ini.

