

SKRIPSI

**ANALISIS SCALING KONTAINER APLIKASI MOODLE
MENGUNAKAN KUBEADM DAN KUBERNETES SEBAGAI
MANAJEMEN WORKLOADS**



BAGUS FAHRUDIN JAMAL

NIM: 195410183

**PROGRAM STUDI INFORMATIKA
PROGRAM SARJANA
FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS TEKNOLOGI DIGITAL INDONESIA
YOGYAKARTA**

2023

SKRIPSI

**ANALISIS SCALING KONTAINER APLIKASI MOODLE
MENGUNAKAN KUBEADM DAN KUBERNETES SEBAGAI
MANAJEMEN WORKLOADS**

Diajukan sebagai salah satu syarat untuk menyelesaikan studi jenjang strata satu (S1)

Program Studi Informatika

Universitas Teknologi Digital Indonesia

Yogyakarta



Disusun oleh

BAGUS FAHRUDIN JAMAL

NIM: 195410183

**PROGRAM STUDI INFORMATIKA
PROGRAM SARJANA
FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS TEKNOLOGI DIGITAL INDONESIA
YOGYAKARTA**

2023

PERNYATAAN KEASLIAN SKRIPSI

Dengan ini saya menyatakan bahwa naskah skripsi ini belum pernah diajukan untuk memperoleh gelar Sarjana di suatu Perguruan Tinggi, dan sepanjang pengetahuan saya tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara sah diacu dalam naskah ini dan disebutkan dalam daftar pustaka.

Yogyakarta, 2 Januari 2023



Bagus Fahrudin Jamal

NIM: 195410183

KATA PENGANTAR

Puja dan puji syukur tetap penulis ucapkan kepada kehadiran Allah SWT, karena telah memberikan Rahmat dan Hidayah-Nya penulis dapat menyelesaikan Skripsi ini. Sholawat serta salam juga selalu tercurahkan kepada Nabi Muhammad SAW, keluarga dan sahabatnya.

Selanjutnya penulis juga ingin menyampaikan banyak terima kasih atas dukungan, bantuan, bimbingan, dan masukan dari berbagai belah pihak selama penyusunan skripsi ini sampai bisa selesai. Oleh karena itu pada kesempatan ini penulis ingin menyampaikan rasa terima kasih yang sebesar-besarnya terutama kepada:

1. Allah SWT, yang telah memberikan nikmat sehat sehingga penulis dapat menyelesaikan skripsi ini.
2. Bapak Ir. Totok Suprawoto, M.M., M.T. selaku Rektor Universitas Teknologi Digital Indonesia.
3. Ibu Dini Fakta Sari, S.T., M.T. selaku Ketua Program Studi Informatika.
4. Bapak M. Agung Nugroho, S.Kom., M.Kom. selaku Dosen Pembimbing yang telah membantu dan membimbing selama pengerjaan skripsi ini.
5. Bapak Wagito, S.T., M.T., dan Ibu Dr. L.N. Harnaningrum, S.Si, M.T. selaku dosen penguji skripsi penulis.
6. Ayah, Ibu, Saudara-i, yang masih semangat berjuang memberikan dukungan, mengingatkan, dan selalu memberikan do'a yang terbaik.

7. Imas Intan permatasari, Amd.Kom. yang masih tetap memberikan dukungan di berbagai macam keadaan hidup.
8. Teman kuliah seperjuangan, Rekan kerja, dan yang lain yang tidak dapat disebutkan satu-persatu semuanya, terimakasih atas semua ide, do'a, dan semua dukungan.
9. Semua pihak yang telah memberikan masukan, arahan, dan dukungan, yang tidak dapat penulis sebutkan satu per satu.

Penulis Menyadari bahwa tulisan ini tidak luput dari kesalahan dan kekurangan. Oleh karena itu, semua kritik dan saran pembaca akan penulis terima dengan senang hati untuk perbaikan naskah penelitian yang lebih baik.

Yogyakarta, 10 Januari 2023

Penulis

Bagus Fahrudin Jamal

NIM: 195410183

DAFTAR ISI

PERNYATAAN KEASLIAN SKRIPSI.....	iii
KATA PENGANTAR	iv
DAFTAR ISI.....	vi
DAFTAR GAMBAR	viii
DAFTAR TABEL.....	x
INTISARI.....	xi
<i>ABSTRACT</i>	xii
BAB I PENDAHULUAN	1
1.1 Latar Belakang Masalah.....	1
1.2 Rumusan Masalah	3
1.3 Ruang Lingkup	3
1.4 Tujuan Penelitian.....	3
1.5 Manfaat Penelitian.....	4
BAB II TINJAUAN PUSTAKA DAN DASAR TEORI.....	5
2.1 Tinjauan Pustaka	5
2.2 Dasar Teori	7
2.2.1 Scalability.....	7
2.2.2 Container	7

2.2.3. Kubernetes.....	9
2.2.4. Kubeadm	11
2.2.5. Load dan testing performa	11
2.2.6. Apache Jmeter.....	12
BAB III METODE PENELITIAN.....	15
3.1 Bahan dan Data	15
3.2 Peralatan	15
3.3 Prosedur dan Pengumpulan Data	16
3.4 Analisis dan Rancangan Sistem.....	17
BAB IV IMPLEMENTASI DAN PEMBAHASAN	22
4.1 Implementasi dan Uji Coba Sistem	22
4.1.1. Pengujian dengan Docker Compose	28
4.1.2. Pengujian dengan Kubernetes (Tanpa <i>Scaling</i>)	29
4.1.3. Pengujian dengan Kubernetes (<i>Scaling</i> minimal pod 1).....	30
4.1.4. Pengujian dengan Kubernetes (<i>Scaling</i> minimal pod 5).....	30
4.2 Pembahasan	35
BAB V PENUTUP.....	42
5.1 Kesimpulan.....	42
5.2 Saran.....	42
DAFTAR PUSTAKA	44

DAFTAR GAMBAR

Gambar 2.1 Arsitektur Docker	8
Gambar 2.2 Arsitektur Kubernetes (Hausenblas, 2018)	10
Gambar 2.3 Multithreading Pada Jmeter (Rodrigues et al., 2019).....	13
Gambar 3.1 Skema Prosedur Pengujian.....	17
Gambar 3.2 Skema Rancangan Sistem Docker Compose	19
Gambar 3.3 Skema Rancangan Sistem VM Kubernetes.....	20
Gambar 4.1 Folder konfigurasi Docker Compose	23
Gambar 4.2 Menjalankan Perintah Kubernetes dengan kubectl	23
Gambar 4. 3 Konfigurasi Skenario Dalam Jmeter	24
Gambar 4.4 Kondisi CPU Host Tanpa Ada VM Berjalan	25
Gambar 4.5 Kondisi CPU Host dengan VM Docker Compose	26
Gambar 4.6 Penggunaan Sumberdaya Di Dalam VM Docker Compose .	27
Gambar 4.7 Kondisi CPU Host Dengan VM Kubernetes.....	27
Gambar 4.8 Penggunaan Sumberdaya Di Dalam VM Kubernetes	28
Gambar 4.9 Grafik Jumlah Sampel Pengujian.....	31
Gambar 4.10 Grafik Tingkat Sampel Sukses Saat Pengujian	32
Gambar 4.11 Grafik Tingkat Error Pada Sampel Pengujian.....	33
Gambar 4.12 Grafik Rata-rata Tingkat Error Saat Pengujian	34
Gambar 4.13 Sampel Pengujian HTTP Error 503 Pada Jmeter	36
Gambar 4.14 HTTP Error 503 Di Halaman Browser	36

Gambar 4.15 HTTP Status 200 OK Di Halaman Browser	37
Gambar 4.16 Perubahan Konfigurasi Host Database Pada Config.php....	38
Gambar 4.17 Error 503 Setelah Merubah Konfigurasi	39
Gambar 4.18 Potongan Kode Exception Aplikasi Moodle.....	39
Gambar 4.19 Perubahan Potongan Kode Exception Aplikasi Moodle	40
Gambar 4.20 HTTP Error 404 Dengan Pesan Service Unavailable	40
Gambar 4.21 Pod Mysql Running Setelah Restart 3 Kali.....	41

DAFTAR TABEL

Tabel 2.1 Rangkuman Keterkaitan Dengan Penelitian Sebelumnya	6
Tabel 3.1 Spesifikasi Peralatan	15
Tabel 4.1 Hasil Pengujian Dengan Docker Compose	28
Tabel 4.2 Hasil Pengujian Dengan Kubernetes (Tanpa Scaling).....	29
Tabel 4.3 Hasil Pengujian Dengan Kubernetes (Scaling min pod 1).....	30
Tabel 4.4 Hasil Pengujian Dengan Kubernetes (Scaling min pod 5).....	30

INTISARI

Teknologi Informasi berkembang sangat pesat salah satunya adalah teknologi Kubernetes yang telah diperkenalkan oleh Google pada tahun 2014. Terdapat banyak fitur pada Kubernetes yang dapat di gunakan dan di analisis, akan tetapi pada penelitian ini akan di lakukan uji coba terhadap fitur *Scaling* atau *Horizontal Pod Autoscaling* (HPA) dan penerapannya terhadap aplikasi *Elearning* yang sering di gunakan pada kampus dan sekolah yaitu Moodle.

Proses uji coba yang dilakukan menggunakan metode eksperimental yaitu dengan melakukan percobaan pada suatu variabel, mengamati proses dan mendata hasil percobaan, dan melakukan kesimpulan. Pengujian kemudian dilakukan pada mesin virtual yaitu pada VM dengan OS ubuntu server 22.04 dan komponen kubernetes di dalamnya. Setelah berhasil terpasang dan berstatus *Ready* maka aplikasi *Elearning* Moodle yang telah berada dalam container akan di jalankan. Selain itu docker compose pada VM yang berbeda untuk hasil perbandingan data.

Setelah melakukan pengujian diperoleh hasil dimana pengaktifan *Autoscaling* justru memberikan hasil yang buruk terhadap aplikasi, hal ini dikarenakan *Horizontal Pod Scaling* (HPA) lebih di peruntukkan pada aplikasi yang bersifat *stateless* bukan aplikasi *Elearning* Moodle yang *stateful*. Pada proses Analisa juga mendapati bahwa *Autoscaling* yang di aktifkan pada pod aplikasi memberikan beban berlebih pada pod basisdata menjadikan mysql server tidak mampu memberikan respon. Karena *Autoscaling* tidak dapat dilakukan pada pod mysql maka diharapkan melakukan metode lain seperti database sharding atau ReplicaSet.

Kata Kunci: *Kubernetes, Horizontal Pod Autoscaling, Kubeadm*

ABSTRACT

Information technology is developing very rapidly, one of which is the Kubernetes technology which was introduced by Google in 2014. There are many features in Kubernetes that can be used and analyzed, but in this research a test will be carried out on the Scaling or Horizontal Pod Autoscaling feature (HPA) and its application to the Elearning application that is often used on campuses and schools, namely Moodle.

The trial process was carried out using the experimental method, namely by conducting an experiment on a variable, observing the process and recording the results of the experiment, and making conclusions. Testing was then carried out on a virtual machine, namely on a VM with ubuntu server 22.04 OS and the Kubernetes component in it. After it is successfully installed and the status is Ready, the Elearning Moodle application that is already in the container will be run. Additionally docker compose on a different VM for data comparison results.

After conducting the test, the results obtained where the activation of Autoscaling actually gave bad results to the application, this is because Horizontal Pod Scaling (HPA) is more intended for stateless applications, not stateful like Elearning Moodle applications. The Analysis process also found that Autoscaling which was activated on the application pod gave an excessive load on the database pod making mysql server unable to respond. Since Autoscaling cannot be performed on mysql pods it is desirable to perform other methods such as database sharding or ReplicaSet.

Keywords: Kubernetes, Horizontal Pod Autoscaling, Kubeadm