

Implementasi *Load Balancing* pada *Learning Management System* di SMK Negeri 2 Yogyakarta

Salma Kusumawardhani, Eko Marpanaji
Universitas Negeri Yogyakarta
E-mail: salmakusumawardhani.2020@student.uny.ac.id

ABSTRAK

SMK Negeri 2 Yogyakarta mengembangkan *Learning Management System* (LMS) untuk mempermudah pengelolaan proses pembelajaran dengan sumber daya server yang sudah terpenuhi sebagai upaya mengoptimalkan pemanfaatan teknologi informasi guna meningkatkan kualitas layanan digital. Akan tetapi, pada beberapa kasus seperti saat ujian berlangsung kerap mengalami overload. Penelitian ini bertujuan untuk mengetahui mengenai proses penerapan *load balancing* pada LMS agar lebih optimal. Metode penelitian yang digunakan diantaranya perumusan masalah, instalasi dan konfigurasi, uji coba, dan analisis hasil. Dalam proses pengembangan ini juga dilakukan pengujian dengan mengacu standar software quality model ISO 25010 pada aspek Usability dan Performance Efficiency. Hasil dari proses penelitian ini yaitu menghasilkan LMS yang telah dilakukan optimalisasi kinerja penerapan High Availability menggunakan metode *load balancing* dengan algoritma Round Robin melalui software HAProxy sebagai penyeimbang beban, serta Galera Cluster pada database untuk meminimalisir kegagalan pada data. Hasil pengujian pada aspek Usability memastikan penerapan *load balancing* dengan memaksimalkan penggunaan sumber daya dan meminimalisir waktu respon serta kegagalan sistem. Aspek Performance Efficiency berdasarkan Quality of Service mendapatkan hasil bahwa seluruh pengguna sudah berhasil mengakses LMS tanpa error atau terjadi kesalahan pada periode tertentu.

Kata kunci: *Learning Management System*, *load balancing*, HAProxy

ABSTRACT

SMK Negeri 2 Yogyakarta has developed a *Learning Management System* (LMS) to facilitate the management of the learning process with existing server resources as an effort to optimize the utilization of information technology to enhance digital service quality. However, in some cases such as during exams, it often experiences overload. This research aims to investigate the process of implementing *load balancing* on the LMS to make it more optimal. The research methods used include problem formulation, installation and configuration, testing, and analysis of results. In this development process, testing was also conducted referring to the ISO 25010 software quality model standards on the aspects of Usability and Performance Efficiency. The results of this research process are the production of an LMS that has been optimized for performance with the implementation of High Availability using *load balancing* methods with Round Robin algorithms through HAProxy software as a load balancer, as well as Galera Cluster on the database to minimize data failures. The test results on the Usability aspect ensure the implementation of *load balancing* by maximizing resource usage and minimizing response time and system failures. The Performance Efficiency aspect based on Quality of Service has resulted in all users successfully accessing the LMS without errors or failures occurring during certain periods.

Keywords: *Learning Management System*, *load balancing*, HAProxy

PENDAHULUAN

Semakin meningkatnya kompleksitas layanan digital dalam sektor pendidikan, salah satu aspek yang menjadi fokus utama adalah peningkatan kualitas layanan tersebut [1]. Salah satu layanan yang menjadi

prioritas dalam peningkatan kualitas tersebut yaitu *Learning Management System* (LMS) yang merupakan perangkat lunak atau software sebagai penunjang proses pembelajaran secara *online* [2]. SMK Negeri 2 Yogyakarta telah mengembangkan

Learning Management System dengan mengadopsi Moodle sebagai bentuk pengoptimalan penggunaan teknologi informasi untuk meningkatkan kualitas layanan digital.

Penerapan *Learning Management System* di SMK Negeri 2 Yogyakarta yang diharapkan mampu memudahkan pengelolaan pembelajaran berbasis *online* nyatanya masih belum bisa menjawab tujuan penerapannya secara optimal. SMK Negeri 2 Yogyakarta memiliki lebih dari 2000 siswa, yang apabila melaksanakan kegiatan pembelajaran yang membutuhkan sistem untuk dapat diakses dalam satu waktu secara bersamaan, seperti pada saat ujian berlangsung, dapat mengakibatkan server menjadi *overload*, melambat, dan akhirnya menjadi *down*. Berdasarkan hasil wawancara kepada kepala ICT SMK Negeri 2 Yogyakarta, yaitu Bapak Haryanto, ST., M.Kom. telah dilakukan *Penetration Testing* berupa audit jaringan melalui aspek banyaknya siswa yang ada, server *Learning Management System* hanya dapat menampung 800 siswa.

Sebenarnya pengembangan *Learning Management System* di SMK Negeri 2 Yogyakarta telah mencapai tahap yang cukup matang dalam beberapa aspek. Salah satunya adalah ketersediaan infrastruktur teknologi yang memadai, yaitu memiliki server dengan jumlah 12 *core* CPU dengan penyimpanan sebesar 2 TB dan 16 GB kapasitas memori (RAM). Sementara itu, rekomendasi minimum yang disarankan pada Moodle *Community Forums* dalam pengoperasian Moodle diperlukan setidaknya 6 *core* CPU dengan pembagian *server web*, *database server*, dan server aplikasi dengan penyesuaian penggunaannya [3].

Implementasi *Learning Management System* belum berjalan dengan optimal meskipun telah ditempatkan sumber daya yang cukup kuat pada server dan adanya *bandwidth* sebesar 300 hingga 900 Mbps yang telah diintegrasikan dalam perencanaan infrastruktur. Server dengan spesifikasi yang ada seharusnya mampu menangani beban kerja yang cukup besar serta *bandwidth* yang cukup tinggi seharusnya mendukung kelancaran akses dan distribusi *Learning Management System*. Meskipun demikian, terdapat salah satu aspek yang masih perlu menjadi perhatian serius, yaitu pengoptimalan dalam menerapkan *High Availability* pada *Learning Management System*.

Oleh karena itu, untuk memastikan aktivitas dan kelancaran layanan yang dapat diakses dari *Learning Management System* dapat berjalan optimal, diperlukannya standar *availability* dengan menerapkan metode *load balancing*. Melalui metode *load balancing* dapat memaksimalkan penggunaan sumber daya yang sudah tersedia yang memungkinkan distribusi beban kerja secara merata di antara server yang sudah ada [4].

Load balancing merupakan suatu teknologi untuk mendistribusikan permintaan layanan yang datang pada waktu yang bersamaan ke beberapa server sekaligus, sehingga beban kerja diantara beberapa server tersebut menjadi lebih sedikit, serta dapat memastikan bahwa tidak ada kelebihan beban pada salah satu server, maka dari itu kinerja dan ketersediaan server dapat meningkat [5]. Apabila terjadi kegagalan pada salah satu server, tidak akan mempengaruhi layanan yang sedang berjalan karena akan ada server lain yang melakukan *backup* dengan mendistribusikan

beban ke server lain, sehingga sistem yang ada menjadi terdistribusi [6]. Layaknya berbagai jenis pengembangan *Learning Management System*, penelitian terhadap penerapan teknologi *load balancing* pada *Learning Management System* perlu dilakukan guna mengetahui perbedaan kinerja *Learning Management System* yang ditinjau dari banyaknya pengguna yang mengakses pada satu waktu yang bersamaan melalui pengujian berstandar ISO/IEC 25010 atau dikenal sebagai SQuaRE (*Software Quality Requirements and Evaluation*) dengan aspek *Usability* dan *Performance Efficiency*. Oleh karena itu, penelitian ini dilakukan dengan tujuan meningkatkan kinerja server pada penggunaan *Learning Management System* yang ada di SMK Negeri 2 Yogyakarta, dengan harapan dapat mengatasi permasalahan yang ada pada sisi pengguna agar proses pembelajaran *berbasis online* dapat berjalan dengan lancar bersamaan dengan media pembelajaran yang efisien.

METODE

Penelitian ini dikembangkan dengan menggunakan merujuk pada penelitian yang dilakukan oleh Rachmawan et al, 2016 yaitu Penerapan Teknik Load Balancing Pada Web Server Lokal Dengan Metode NTH Menggunakan Mikrotik, melalui 4 tahap, diantaranya perumusan masalah, instalasi dan konfigurasi, uji coba, dan analisis hasil [7].

1. Perumusan masalah

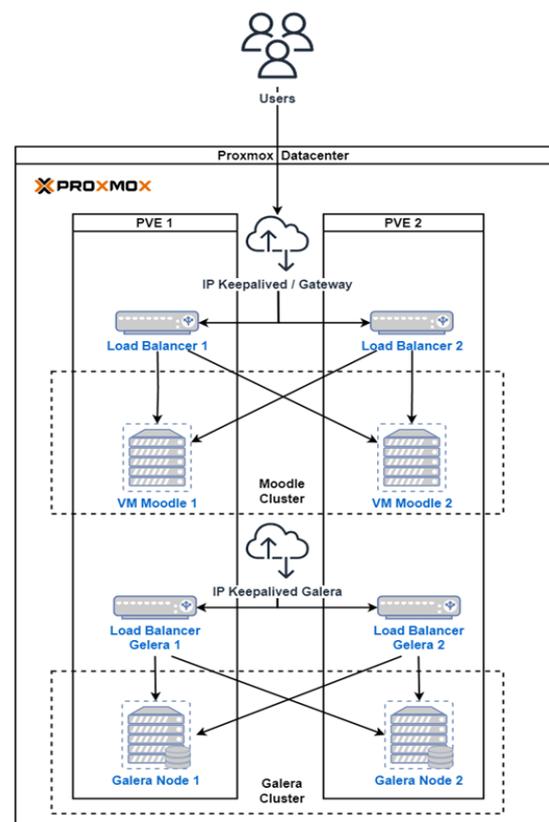
Perumusan masalah dimulai dari analisis kebutuhan pengguna serta penggunaan informasi dan data lainnya untuk merancang dan mengembangkan produk atau sistem yang sesuai. Perumusan

masalah dibagi menjadi dua kategori, yaitu persyaratan fungsional dan persyaratan non-fungsional. Persyaratan fungsional mengacu pada fungsionalitas atau fitur-fitur yang harus ada dalam produk atau sistem untuk memenuhi kebutuhan pengguna, sementara persyaratan non-fungsional mengacu pada kinerja, keamanan, keandalan, dan faktor-faktor lain yang harus dipenuhi oleh produk atau sistem [8].

2. Instalasi dan Konfigurasi

Implementasi dalam penelitian ini merujuk pada tahapan dimana desain sistem yang telah dibuat sebelumnya akan diwujudkan ke dalam bentuk nyata [9].

Desain pada penelitian ini mencakup rancangan arsitektur sistem dan teknologi yang digunakan untuk implementasi *load balancing* pada *Learning Management System*. Berikut merupakan design arsitektur *Learning Management System* yang saat ini digunakan.

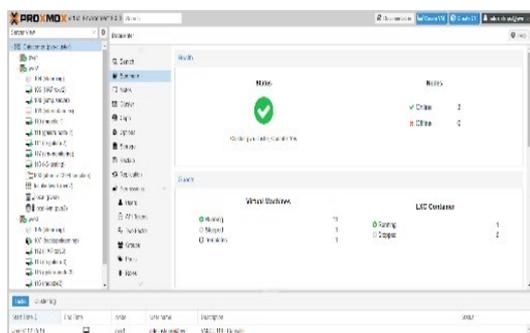


Gambar 1. Desain Arsitektur LMS dengan Load Balancing

Berikut penjelasan dari desain arsitektur yang dikembangkan dalam penelitian ini.

- a. Menggunakan 2 buah server fisik yang dimanajemen pada Proxmox *Virtual Environment*.
- b. Dalam satu server fisik terdiri dari 2 buah VM *load balancer*, VM *web server*, dan VM *database*.
- c. *Load balancing* akan diterapkan dengan HAProxy pada kedua VM *load balancernya*, untuk membagi permintaan pengguna pada *web server* maupun *databasenya*.
- d. Manajemen *database* menggunakan Galera *Cluster* yang akan dapat sinkronisasi antar *database* dengan penyeimbang beban HAProxy.

Tahap selanjutnya adalah mempersiapkan *virtual machine* (VM) yang akan digunakan untuk implementasi arsitektur *load balancing* pada Proxmox *Virtual Environment* yang telah diterapkan 2 buah server fisik menjadi sebuah *cluster* Datacenter. Melakukan instalasi HAProxy pada masing-masing VM *load balancer*, yaitu HAProxy1 dan HAProxy2 sebagai pendistribusian beban kerja pada aplikasi Moodle.



Gambar 2. Infrastruktur LMS pada Proxmox

Setelah itu, dilakukan konfigurasi dengan menambahkan HAProxy *Listener* berupa *fe_moodle*, menentukan server *web backend* dengan menambahkan IP VM

moodle1 dan *moodle2*, dan mengaktifkan statistik HAProxy untuk menyediakan informasi pemantauan.

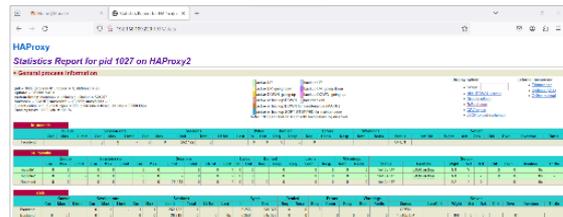
```
frontend fe_moodle
  bind *:180
  default_backend be_moodle

backend be_moodle
  server moodle1 192.168.109.205:80 check
  server moodle2 192.168.109.206:80 check

listen stats
  bind *:1936
  stats enable
  stats hide-version
  stats refresh 5s
  stats show_node
  stats auth user:pass:password
```

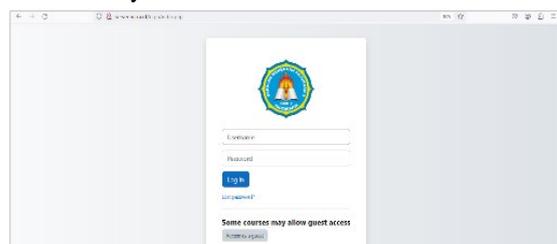
Gambar 3. Konfigurasi HAProxy

Selain melakukan instalasi HAProxy untuk aplikasi Moodle, selanjutnya melakukan instalasi HAProxy untuk masing-masing VM *load balancer* Galera, yaitu *lb-galera-1* dan *lb-galera-2* sebagai pendistribusian beban kerja pada *database* yang berada pada Galera *cluster*. Hasil dari konfigurasi tersebut dapat diketahui pada statistik HAProxy dengan mengakses IP Keepalive dengan port 1936 dan */stats*.



Gambar 4. Hasil Statistik HAProxy

Pada instalasi moodle di kedua VM, yaitu *moodle1* dan *moodle2* dilakukan dengan instalasi Nginx *web server*, MariaDB server, dan PHP *Extension* yang selanjutnya menginstall Moodle. Setelah proses instalasi telah selesai, selanjutnya dilakukan pemindahan *repository* ke */var/www/moodledata* dan melakukan setup databasenya.



Gambar 5. Tampilan LMS dengan Load Balancing

3. Uji Coba

Uji coba dilakukan dengan menggunakan metode pengujian berbasis ISO/IEC 25010 dengan penerapan karakteristik *Usability* (Kemudahan Penggunaan) dan *Performance Efficiency* (Efisiensi Kinerja).

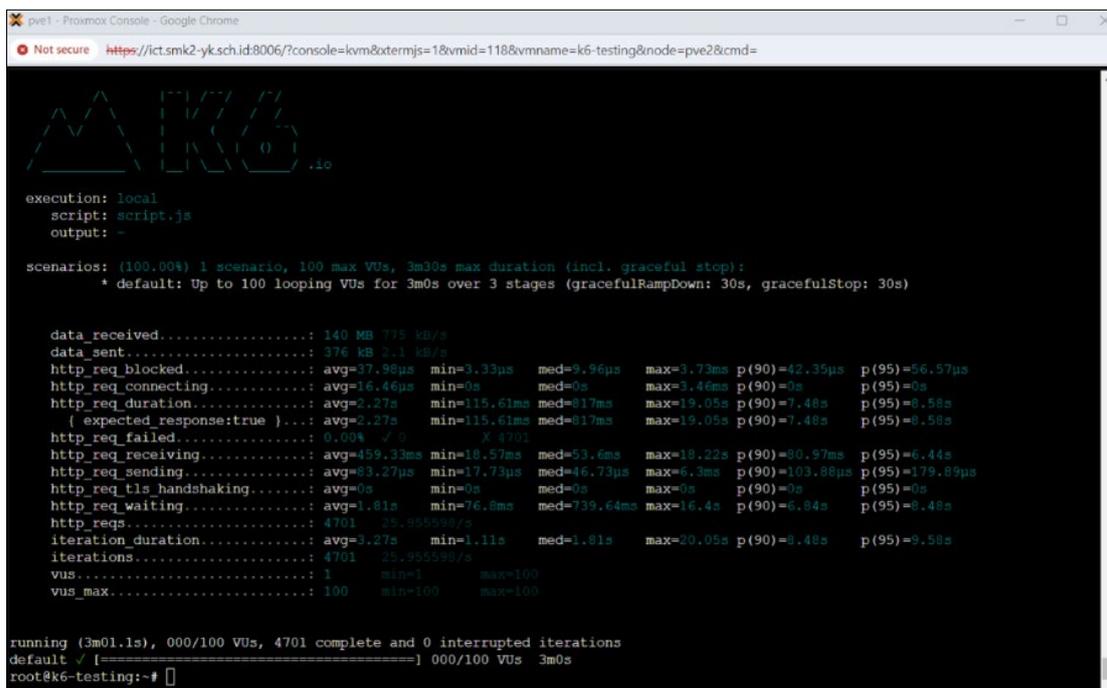
a. Usability

Pengujian *Usability* dilakukan dengan wawancara kepada staf ICT SMK Negeri 2 Yogyakarta untuk melakukan validasi fungsi *load balancing* yang telah diimplementasikan pada *Learning Management System* dapat memenuhi kebutuhan pengguna yang telah direncanakan. Hasil dari pengujian *Usability* disajikan dengan *Conclusion Drawing / Verification* atau Penarikan Kesimpulan bahwa *Learning Management System* yang sebelumnya telah diberikan sumber daya yang maksimal nyatanya masih belum memiliki kinerja yang optimal. Terlihat dari hasil *Load Testing* yang menunjukkan perbedaan signifikan, pengguna tidak mengalami penurunan performa ketika

jumlah pengguna meningkat. Penerapan metode *High Availability* dengan teknologi *load balancing* tidak menemui hambatan atau kendala, dan sebaliknya, terdapat peningkatan dalam durasi *load*. Hal ini mengindikasikan efisiensi sistem yang lebih baik.

b. Performance Efficiency

Pengujian *Performance Efficiency* dilakukan menggunakan *testing tool* Grafana K6 dengan menentukan skenario pengujian yang mencakup beban tinggi dan kondisi pengguna yang realistis, pada pengujian kali ini menggunakan skala, seperti yang sudah diterapkan pada tahap implementasi, yaitu menguji *Learning Management System* dengan skala maksimal sumber daya yang telah diterapkan, yaitu 100 pengguna atau pada pengujian K6 ini adalah 100 VUs (*Virtual Users*) dengan waktu akses selama 2 menit 30 detik. Berikut merupakan konfigurasi pengujian dari K6. Hasil dari konfigurasi pengujian adalah sebagai berikut.



```
pve1 - Proxmox Console - Google Chrome
Not secure https://ict.smk2-yk.sch.id:8006/?console=lvm&xtermjs=1&vmid=118&vmmname=k6-testing&nnode=pve2&cmd=

M K6 .io

execution: local
script: script.js
output: -

scenarios: (100.00%) 1 scenario, 100 max VUs, 3m30s max duration (incl. graceful stop):
 * default: Up to 100 looping VUs for 3m0s over 3 stages (gracefulRampDown: 30s, gracefulStop: 30s)

data_received.....: 140 MB 775 kB/s
data_sent.....: 376 kB 2.1 kB/s
http_req_blocked.....: avg=37.98µs min=3.33µs med=9.96µs max=3.73ms p(90)=42.35µs p(95)=56.57µs
http_req_connecting.....: avg=16.46µs min=0s med=0s max=3.46ms p(90)=0s p(95)=0s
http_req_duration.....: avg=2.27s min=115.61ms med=817ms max=19.05s p(90)=7.48s p(95)=8.58s
  ( expected_response:true )...: avg=2.27s min=115.61ms med=817ms max=19.05s p(90)=7.48s p(95)=8.58s
http_req_failed.....: 0.00% 0/4701 0/4701
http_req_receiving.....: avg=459.33ms min=18.57ms med=53.6ms max=18.22s p(90)=80.97ms p(95)=6.44s
http_req_sending.....: avg=83.27µs min=17.73µs med=46.73µs max=6.3ms p(90)=103.88µs p(95)=179.89µs
http_req_tls_handshaking.....: avg=0s min=0s med=0s max=0s p(90)=0s p(95)=0s
http_req_waiting.....: avg=1.81s min=76.8ms med=739.64ms max=16.4s p(90)=6.84s p(95)=8.48s
http_reqs.....: 4701 25.955590/s
iteration_duration.....: avg=3.27s min=1.11s med=1.81s max=20.05s p(90)=8.48s p(95)=9.58s
iterations.....: 4701 25.955590/s
vus.....: 1 min=1 max=100
vus_max.....: 100 min=100 max=100

running (3m01.1s), 000/100 VUs, 4701 complete and 0 interrupted iterations
default ✓ [=====] 000/100 VUs 3m0s
root@k6-testing:~#
```

Gambar 6. Hasil Pengujian Performance Efficiency dengan K6

Hasil pengujian *Performance Efficiency* dievaluasi menggunakan parameter *Quality of Service* (QoS) yang merupakan standar dari ETSI-TIPHON melalui faktor berikut.

$$\begin{aligned} &1) \textit{Throughput} \\ &= \frac{\textit{Jumlah data}}{\textit{Waktu pengiriman}} \times 100\% \\ &= \frac{140 \textit{ mb}}{2.27 \textit{ detik}} \times 100\% \\ &= 61.72\% \end{aligned}$$

Hasil perhitungan *Throughput* memperoleh kategori Bagus dengan hasil 61.72% pada indeks 3.

$$\begin{aligned} &2) \textit{Packet Loss} \\ &= \frac{\textit{Data Hilang}}{\textit{Data Terkirim}} \times 100\% \\ &= 0376 \textit{ kB} \times 100\% \end{aligned}$$

Hasil perhitungan *Packet Loss* memperoleh kategori Sangat Bagus dengan hasil 0% pada indeks 4.

$$\begin{aligned} &3) \textit{Delay} \\ &= \frac{\textit{Total Delay}}{\textit{Paket Diterima}} \\ &= 37.98 \textit{ }\mu\text{s} \end{aligned}$$

Hasil perhitungan *Delay* memperoleh kategori Sangat Bagus dengan hasil dibawah 150 ms pada indeks 4, yaitu 37.98 μ s untuk menunggu slot koneksi yang tersedia.

$$\begin{aligned} &4) \textit{Jitter} \\ &= \frac{\textit{Paket Hilang}}{\textit{Paket Dikirim}} \\ &= \frac{83.27 \textit{ }\mu\text{s}}{37.98 \textit{ }\mu\text{s}} \\ &= 2.192 \textit{ }\mu\text{s} \\ &= 0.002192 \textit{ ms} \end{aligned}$$

Hasil perhitungan *Jitter* memperoleh kategori Bagus dengan hasil di antara 0 - 75 ms, yaitu 0.002192 ms pada indeks 3.

4. Analisis Hasil

Tahapan analisis hasil menggunakan merupakan kerangka kerja yang digunakan untuk menggambarkan dan menilai kualitas keseluruhan dari suatu sistem atau produk perangkat lunak dengan menyediakan terminologi yang terstruktur untuk membantu

dalam memahami, mengevaluasi, dan menganalisis kualitas perangkat lunak dengan cara yang terukur dan dapat dijelaskan [10].

HASIL DAN PEMBAHASAN

Berdasarkan seluruh proses penelitian, yang dimulai dari pengumpulan persyaratan kebutuhan hingga pemeliharaan, mendapatkan hasil sebagai berikut:

Penelitian ini menghasilkan *Learning Management System* sebagai media penunjang kualitas layanan digital di SMK Negeri 2 Yogyakarta yang telah dilakukan optimalisasi kinerja melalui penerapan *High Availability* menggunakan metode *load balancing* dengan algoritma *Round Robin*. *Learning Management System* pada penelitian ini menggunakan HAProxy untuk mendistribusikan lalu lintas beban *request* secara merata pada dua buah server aplikasi dan server *database*. Sebagai penunjang optimalisasinya, diterapkan *Galera Cluster* pada databasenya untuk meningkatkan ketersediaan layanan dengan meminimalisir kegagalan untuk memastikan konsistensi data. Model pengembangan perangkat lunak menggunakan *Iterative Waterfall Model* dengan melakukan perulangan model *Waterfall* terhadap tahapan standar proses SDLC yaitu *Requirements* (persyaratan kebutuhan), *Design* (desain), *Implementation* (implementasi), *Testing* (pengujian), *Deployment* (distribusi), dan *Maintenance* (pemeliharaan).

Learning Management System di SMK Negeri 2 Yogyakarta telah memenuhi standar ISO/IEC 25010 dalam aspek pengujian *Usability* (Kemudahan Penggunaan) berdasarkan wawancara oleh kepala ICT SMK Negeri 2 Yogyakarta, yaitu Bapak Haryanto, S.T., M.Kom. bahwa dengan

diterapkannya *load balancing*, terjadi perbedaan signifikan dalam responsivitas dan tidak terjadi penurunan performa ketika digunakan oleh pengguna yang banyak dalam satu waktu. Perbandingan antara penerapan infrastruktur server tunggal *Learning Management System* dengan yang sudah diterapkan *load balancing* sudah sangat signifikan dengan memaksimalkan penggunaan sumber daya dan meminimalisir waktu respon serta kegagalan sistem.

Learning Management System di SMK Negeri 2 Yogyakarta telah memenuhi standar ISO/IEC 25010 dalam aspek pengujian *Performance Efficiency* (Efisiensi Kinerja) dilakukan pengujian dengan *load testing* dengan 50 pengguna virtual dalam waktu 2 menit mendapatkan hasil bahwa seluruh pengguna sudah berhasil mengakses *Learning Management System* tanpa *error* atau terjadi kesalahan. Berdasarkan perhitungan *Quality of Service* (QoS) yang merupakan standar dari ETSI-TIPHON, *Learning Management System* memperoleh kategori Bagus dengan hasil 55.57% pada parameter *Throughput* untuk mengukur jumlah data yang dapat ditransmisikan, memperoleh kategori Sangat Bagus dengan hasil 0% pada parameter *Packet Loss* yang merupakan persentase paket data yang hilang selama proses transmisi, memperoleh kategori Sangat Bagus dengan hasil dibawah 150 ms pada parameter *Delay* untuk mengukur waktu yang dibutuhkan untuk mentransmisikan data, dan memperoleh kategori Bagus dengan hasil di antara 0 - 75 ms pada parameter *Jitter* merupakan hasil variasi dalam waktu tiba paket data, yang dapat mengakibatkan ketidakkonsistenan dalam latensi jaringan.

SIMPULAN

Penelitian ini menghasilkan *Learning Management System* (LMS) di SMK Negeri 2 Yogyakarta yang secara signifikan dapat mengoptimalkan kinerja melalui penerapan *High Availability* menggunakan metode *load balancing* dan algoritma Round Robin dengan *software* HAProxy dan Galera Cluster yang diterapkan pada database untuk meningkatkan ketersediaan layanan. Berdasarkan pengujian yang telah dilakukan, *Learning Management System* telah memenuhi standar ISO/IEC 25010 dalam aspek *Usability* dan *Performance Efficiency*. Metode *load balancing* dinilai 100% sesuai dengan rencana, efektif, dan mampu menangani permintaan pengguna yang banyak. Pengelolaan infrastruktur dan distribusi data melalui algoritma Round Robin juga telah efektif dalam menangani *fault tolerance*. Dengan demikian, *Learning Management System* di SMK Negeri 2 Yogyakarta dapat dianggap sebagai solusi yang efektif dalam meningkatkan kualitas layanan digital, memastikan ketersediaan layanan yang tinggi, responsivitas yang baik, dan efisiensi kinerja yang optimal.

DAFTAR RUJUKAN

- [1] Haleema, A., Javaida, M., Qadri, M. A., & Suman, R. (2022). Understanding the role of digital technologies in education: A review. *Sustainable Operations and Computers*, 3, 275-285.
- [2] Turnbull, D., Chugh, R., & Luck, J. (2022). An Overview of the Common Elements of Learning Management System Policies in Higher Education Institutions. *TechTrends*, 66, 855-867. <https://doi.org/10.1007/s11528-022-00752-7>.

- [3] Ramadhani, M. A. (2022). Penerapan dan Evaluasi Kinerja Portabel E Learning Server berbasis MoodleBox. *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, 6(5), 2122-2128.
- [4] Waluyo, M.A., Antony, A., & Setiawan, C. (2023). Implementasi Load Balancing Web Server dengan Haproxy Menggunakan Algoritma Round Robin. *Journal of Intelligent Networks and IoT Global*, 1(1), 46-52.
- [5] Shafiq, D. A., Jhanjhi, N. Z., & Abdullah, A. (2022). Load balancing techniques in cloud computing environment: A review. *Journal of King Saud University – Computer and Information Sciences*, 34, 3910-3933.
- [6] Riskiono, S. D., & Pasha, D. (2020). Analisis Metode Load Balancing dalam Meningkatkan Kinerja Website E-Learning. *Jurnal TEKNOINFO*, 14(1), 22-26.
- [7] Rachmawan, D., Irwan, D., & Argyawati, H. (2016). Penerapan Teknik Load Balancing Pada Web Server Lokal Dengan Metode Nth Menggunakan Mikrotik. *Jurnal Penelitian Ilmu Komputer, System Embedded & Logic*, 4(2), 98-108.
- [8] Allan, G. (2021). Analisa Kebutuhan Kebutuhan Sistem Informasi Manajemen Perusahaan Dagang. *KURAWAL Jurnal Teknologi, Informasi dan Industri*, 4(1), 17-30.
- [9] Lukman. (2019). Implementasi Dan Testing Desain Sistem Berkas SOP Amikom Yogyakarta. *Jurnal Teknologi Informasi*, XIV(2), 20-30.
- [10] Subhiyakto, E. R. & Utomo, D. W. (2017). Analisis Dan Perancangan Aplikasi Pemodelan Kebutuhan Perangkat Lunak Menggunakan Metode Prototyping. *Prosiding Seminar Nasional Multi Disiplin Ilmu & Call For Papers Unisbank ke-3*.