

# Server Worker Power Optimization with Virtual Machine Live Migration Technique Using Fuzzy Mamdani

Muhammad Azzahari<sup>1</sup>, Ibnu Khaldun<sup>2</sup>, Mahmud<sup>3</sup>, Rozaliana<sup>4\*</sup>

<sup>1</sup>Jurusan Teknologi Informasi dan Komputer Politeknik Negeri Lhokseumawe

<sup>2</sup>Jurusan Teknik Elektro Politeknik Negeri Lhokseumawe

<sup>3</sup>Jurusan Teknik Mesin Politeknik Negeri Lhokseumawe

<sup>4</sup>Fakultas Ekonomi dan Bisnis Universitas Bumi Persada

## Informasi Artikel

Diterima : 2 Mei 2025  
Revisi : 9 Juni 2025  
Publikasi : 20 Juni 2025

## Kata Kunci:

Optimasi Daya  
Live Migration  
Virtual Machine  
CPU Usage  
Fuzzy Mamdani

## ABSTRAK

Penelitian ini membahas optimasi daya pada server worker dengan menerapkan teknik live migration virtual machine (VM) menggunakan metode logika fuzzy Mamdani. Tujuannya adalah memindahkan VM secara otomatis ke server dengan penggunaan CPU yang lebih rendah, sehingga konsumsi daya dapat ditekan. Sistem dirancang menggunakan platform Proxmox VE dan dilakukan simulasi selama 30 menit dengan dua kondisi: tanpa dan dengan metode Mamdani. Hasil menunjukkan bahwa penggunaan fuzzy Mamdani dapat mengurangi konsumsi daya sebesar 0,26 watt dibandingkan dengan sistem tanpa migrasi. Walau penghematan terlihat kecil, metode ini efektif diterapkan pada skala pusat data yang lebih besar. Teknik ini memungkinkan perpindahan beban kerja VM secara efisien, sehingga server yang tidak aktif bisa berada dalam kondisi idle dengan konsumsi daya minimal. Kesimpulannya, metode fuzzy Mamdani dapat digunakan sebagai solusi penghematan energi di lingkungan komputasi awan tanpa mengorbankan performa layanan.

## ABSTRACT

This study focuses on power optimization of worker servers by applying live migration techniques for virtual machines (VMs) using the Mamdani fuzzy logic method. The main goal is to automatically move VMs to servers with lower CPU usage, thereby reducing overall power consumption. The system was built using the Proxmox VE platform and simulated for 30 minutes under two conditions: without and with Mamdani fuzzy logic. The results show that the use of fuzzy Mamdani reduces power consumption by 0.26 watts compared to the non-migration system. Although the savings may seem minor, the method is effective when applied at a larger data center scale. This technique enables efficient workload distribution by migrating VMs, allowing idle servers to reduce energy usage. In conclusion, Mamdani fuzzy logic offers a practical solution for energy-saving in cloud computing environments without compromising system performance.

This is an open-access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license



## \*Penulis Koresponden

Email: [azzahari@pnl.ac.id](mailto:azzahari@pnl.ac.id)

Cara sitasi IEEE::

M. Azzahari, I. Khaldun, M. Mahmud, dan R. Rozaliana, "Server Worker Power Optimization with Virtual Machine Live Migration Technique Using Fuzzy Mamdani," *Journal of Artificial Intelligence and*

## 1. PENDAHULUAN

Dewasa ini data center merupakan pusat informasi yang menjadi tulang punggung data digital yang menggunakan teknologi internet. Pesatnya pertumbuhan ekonomi dan peningkatan penggunaan data digital sejalan dengan aktivitas kehidupan modern, maka kualitas layanan Data Center yang dapat diakses secara cepat dari layanan cloud menjadi tantangan selain tidak mengorbankan keamanan data. Setiap organisasi seperti Facebook, Equinix, Amazon, dan Google memiliki pusat data sendiri sehingga mampu mengembangkan bisnis yang dijalankan menggunakan layanan cloud. Data center memainkan peran penting utama pada kesuksesan bisnis digital. Penting untuk menemukan kemungkinan pilihan untuk mengoptimalkan infrastruktur dan meningkatkan efisiensi dan produktivitas data center [1]. Dewasa ini data center merupakan pusat informasi yang menjadi tulang punggung data digital yang menggunakan teknologi internet. Pesatnya pertumbuhan ekonomi dan peningkatan penggunaan data digital sejalan dengan aktivitas kehidupan modern, maka kualitas layanan Data Center yang dapat diakses secara cepat dari layanan cloud menjadi tantangan selain tidak mengorbankan keamanan data. Setiap organisasi seperti Facebook, Equinix, Amazon, dan Google memiliki pusat data sendiri sehingga mampu mengembangkan bisnis yang dijalankan menggunakan layanan cloud. Data center memainkan peran penting utama pada kesuksesan bisnis digital. Penting untuk menemukan kemungkinan pilihan untuk mengoptimalkan infrastruktur dan meningkatkan efisiensi dan produktivitas data center [2].

Penelitian optimasi daya data center yang dilakukan telah difokuskan pada penjadwalan prediktif secara realtime [3]. Teknik pengoptimalan daya tersebut dilakukan berdasarkan pertimbangan pola data awal berupa nilai CPU usage dengan menggunakan Moving Average. Di dalam penelitian tersebut disampaikan bahwa dalam pertimbangan kinerja layanan worker server selalu berada pada keadaan yang optimal, agar efisiensi daya pada server worker tersebut dapat dilakukan proses live migration terhadap Virtual Machine (VM) yang dilakukan dengan bantuan prediksi CPU usage. Eksekusi live migration pada VM bekerja sesuai nilai threshold yang telah ditentukan. maka sistem akan menjalankan migrasi VM ke server worker yang lain terlebih dahulu [4].

Pada penelitian ini, simulasi pengaruh optimalisasi daya server worker pada kualitas Data Center dianalisis dengan menggunakan software Proxmox V.E. Pemilihan Proxmox Virtual Environment (Proxmox VE) sebagai Platform data center didasarkan pada hasil penelitian yang menyatakan Proxmox VE merupakan Operating System (OS) mesin virtual yang saat ini mulai banyak dipakai oleh para pengguna teknologi virtualisasi [5]. Hal tersebut disebabkan teknologi virtualisasi OS tersebut bersifat Open source (gratis) sehingga lebih mudah untuk dimodifikasi dan dikembangkan kedepannya. Sebagai hasil akhir dari penelitian ini, dapat disampaikan bahwa penerapan metode Fuzzy Mamdani menggunakan metode Fuzzy Mamdani dengan teknik live migration VM bisa megefisienkan daya pada server worker.

### 1.1 Data Center

Tabel 1. Struktur Pembiayaan Data Center

Tier	Requirement
1	<ul style="list-style-type: none"> <li>Komponen semuanya nonredundan (<i>Server, link</i> dan lain-lain).</li> <li>Garansi ketersediaan (<i>Availability</i>) sebesar 99,671%.</li> <li>Mencakup semua aspek Tier-1.</li> </ul>
2	<ul style="list-style-type: none"> <li>Komponen sudah redundan.</li> <li>Garansi ketersediaan (<i>availability</i>) sebesar 99,74%.</li> <li>Mencakup semua aspek Tier-1 dan Tier-2.</li> </ul>
3	<ul style="list-style-type: none"> <li>Sumber listrik ada dua (<i>dual powered</i>) dan link banyak (<i>multiple</i>).</li> <li>Semua perangkat harus memiliki dual power supply dan kompatibel.</li> <li>Garansi ketersediaan (<i>availability</i>) sebesar 99,982%.</li> <li>Mencakup semua aspek <i>Tier-1, Tier-2</i> dan <i>Tier-3</i>.</li> </ul>
4	<ul style="list-style-type: none"> <li>Semua perangkat pendinginan memiliki <i>dual power supply</i>.</li> <li>Garansi ketersediaan (<i>availability</i>) sebesar 99,995%.</li> </ul>

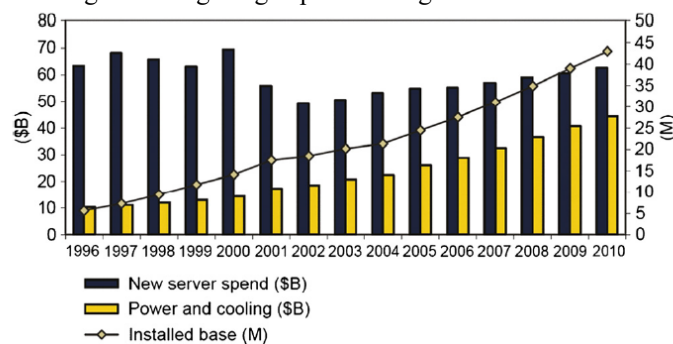
Data center adalah suatu fasilitas yang digunakan untuk menempatkan sistem komputer dan komponen-komponen terkaitnya, seperti sistem telekomunikasi dan penyimpanan data. Fasilitas ini biasanya mencakup juga catu daya redundan (cadangan), koneksi komunikasi data redundan, pengontrol lingkungan (AC, ventilasi), pencegah bahaya kebakaran, serta piranti keamanan fisik. Data center adalah kunci utama yang

mampu menjadi tolak ukur penilaian sebuah provider cloud computing, semakin baik data center maka semakin baik pula layanan cloud computing yang dihasilkan. Data center memiliki tingkatan atau level tersendiri yang disebut tier [6]. Terlihat di dalam Tabel 2.1 mengenai ketersediaan (*availability*) dalam persen, seperti Tier-4 memiliki garansi ketersediaan sebesar 99,995% ini berarti data center dan semua perangkat yang ada harus dapat beroperasi sebanyak 99,995% per tahun. Artinya data center hanya boleh mengalami down time sebesar  $(100\% - 99,995\%) = 0,005\%$  [7].

Kesalahan dalam optimasi daya pada data center dapat berpengaruh terhadap performa data center, sekaligus akan menyebabkan down time data center semakin meningkat. Jika melihat persyaratan sebuah data center maka jelas sekali performa server atau data center harus tetap dijaga walaupun konsumsi daya pada data center tersebut telah dikurangi. Sehingga perlu adanya mekanisme optimasi konsumsi daya yang seiring dengan kebutuhan persyaratan data center tersebut.

## 1.2 Optimasi Daya Data Center

Masalah utama dari pesatnya perkembangan komputasi, baik itu komputasi individual maupun komputasi sekelas data center provider cloud computing adalah meningkatnya konsumsi energi. Seperti terlihat dalam Gambar 1. di bawah ini terlihat bahwa daya dan pendinginan pada data center cenderung memiliki biaya yang terus meningkat seiring dengan perkembangan waktu.



Gambar 1. Struktur Pembiayaan Data Center

Untuk mengatasi hal ini, pengelola data center memiliki beberapa pilihan di antaranya sebagai berikut [1].

1. Menambah kapasitas daya dan pendingin
2. Membangun data center baru

3. Melakukan pengelolaan energi yang memaksimalkan kapasitas yang ada. Dua pilihan awal sangat mahal karena melibatkan belanja modal dan instalasi baru. Maka pilihan ketigalah yang paling memungkinkan untuk mengatasi dua hal tersebut di atas. Dalam pengelolaan energi diperlukan suatu teknik dalam manajemen energi tersebut, beberapa penelitian penulis menemukan teknik manajemen energi tersebut yaitu Dynamic Voltage and Frequency Scaling (DVFS) [4].

## 1.3 Virtual Machine

Dalam sebuah penelitian menjelaskan [8] Virtualisasi telah menjadi aset utama di banyak bidang teknologi informasi. Komputasi awan umumnya didasarkan pada konsep yang disebut virtualisasi. Dalam komputasi, virtualisasi mengacu pada pembuatan versi virtual ke perangkat atau sumber daya apa pun, seperti: server, sistem operasi, perangkat penyimpanan, atau jaringan, yang berarti kerangka membagi sumber daya ke dalam beberapa lingkungan eksekusi. Secara tipikal VM mengemulasikan lingkungan komputasi tetapi meminta sumber daya CPU, Memori, Hard Disk dan jaringan yang akan diatur oleh lapisan virtualisasi yang akan menerjemahkan permintaan ini ke fisik hardware. VM diciptakan dengan lapisan virtualisasi seperti hypervisor atau platform virtualisasi yang berjalan diatas server sistem operasi. Sistem operasi ini dikenal sebagai Host. Lapisan virtualisasi dapat digunakan untuk menciptakan banyak lingkungan VM yang terisolasi. Secara tipikal, *Guest Operating System*, *Virtual Machine* dan Programnya sadar bahwa mereka berjalan pada *platform virtual*, sepanjang *platform virtual* mendukung.

## 1.4 Live Migration Virtual Machine

Live migration adalah sebuah teknologi dimana seluruh VM yang sedang berjalan dipindahkan dari mesin fisik yang satu ke mesin fisik yang lain [9][10]. Secara lebih jelas, hal tersebut berarti seluruh VM termasuk memori yang sedang aktif dan status terakhir dari lingkungan tersebut dipindahkan dari mesin fisik asal ke mesin fisik tujuan (antar host). Hal seperti ini dimaksudkan agar pada saat terjadi

perpindahan VM yang berisi servis online pengguna tidak harus terputus dalam melakukan kegiatannya. Pada tahap akhir perpindahannya virtual I/O devices akan terputus dari sumber dan akan dihubungkan kembali ke mesin fisik tujuan.

### 1.5 Logika Fuzzy

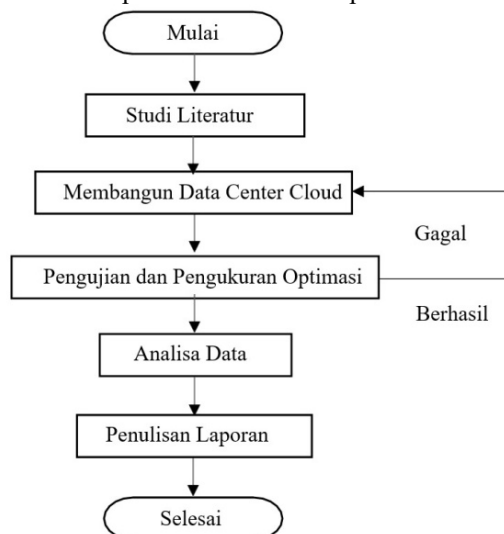
Logika fuzzy adalah metodologi sistem *control* pemecahan masalah, yang cocok untuk diimplementasikan pada sistem, mulai dari sistem yang sederhana, sistem kecil, *embedded system*, jaringan PC, multi channel atau *workstation* berbasis akuisisi data, dan sistem *control*. Dalam logika klasik dinyatakan bahwa segala sesuatu bersifat biner, yang artinya adalah hanya mempunyai dua kemungkinan, “Ya atau Tidak”, “Benar atau Salah”, “Baik atau Buruk” dan lain-lain. Oleh karena itu, sistem ini dapat mempunyai nilai keanggotaan 0 atau 1. Akan tetapi, dalam logika *fuzzy* memungkinkan nilai keanggotaan berada di antara 0 dan 1. Artinya, bisa saja suatu keadaan mempunyai dua nilai “Ya dan Tidak”, “Benar dan Salah”, “Baik dan Buruk” secara bersamaan, namun besar nilainya tergantung pada bobot keanggotaan yang dimilikinya [12]. Bila dibandingkan dengan logika bersifat konvensional, kelebihan logika fuzzy adalah kemampuannya dalam proses penalaran secara bahasa sehingga dalam perancangannya tidak memerlukan persamaan matematik yang rumit. Himpunan fuzzy memiliki dua atribut yaitu: (1) Linguistik, yaitu nama suatu kelompok yang mewakili suatu keadaan tertentu dengan menggunakan bahasa alami, misalnya RENDAH, MENENGAH, TINGGI mewakili variabel temperature, (2) Numeris, yaitu suatu nilai yang menunjukkan ukuran dari suatu variabel, misalnya 10, 35, 40, dan sebagainya.

### 1.6 Metode Mamdani

Metode Mamdani adalah suatu metode Max-Min. Metode ini diperkenalkan oleh Ebrahim Mamdani pada tahun 1975. Untuk mendapatkan hasil dengan menggunakan metode ini, maka diperlukan 4 tahapan. Pembentukan himpunan fuzzy pada metode mamdani, baik variabel input maupun variabel output dibagi menjadi satu atau lebih himpunan fuzzy. Aplikasi fungsi implikasi yang digunakan adalah min. Komposisi aturan yaitu inferensi diperoleh dari kumpulan dan korelasi antar aturan. Ada 3 macam aturan Max, Additive dan Probabilistik OR (Probor). Penegasan (Defuzzyfikasi) dimana input disini adalah suatu himpunan fuzzy yang diperoleh dari komposisi aturan-aturan fuzzy. Output defuzzyfikasi: *Centroid (Center of Mass)*, dan *Mean of Maximum (MOM)*. Logika fuzzy mamdani adalah metode yang sangat fleksibel dan memiliki toleransi pada data yang ada. Mamdani memiliki kelebihan yakni, lebih intuitif, diterima oleh banyak pihak. Penggunaan fuzzy mamdani ini sama halnya dengan penggunaan metode peramalan pada bidang statistik. Penentuan analisis berdasarkan pendekatan fuzzy lebih efisien dalam pendekatan menggunakan angka dibanding dengan metode peramalan. Peramalan dalam statistik dapat menghasilkan galat (error) lebih besar dari pendekatan fuzzy. Dengan metode fuzzy maka akan menghasilkan output yang lebih dekat dengan keadaan sebenarnya [13].

## 2. METODE

Untuk melaksanakan penelitian ini, dibutuhkan beberapa alat dan bahan penunjang penelitian. Objek dari penelitian ini menggunakan Usage CPU sebagai indikator yang akan dianalisis dan digunakan sebagai variabel untuk menentukan trafik pada data center komputasi awan.



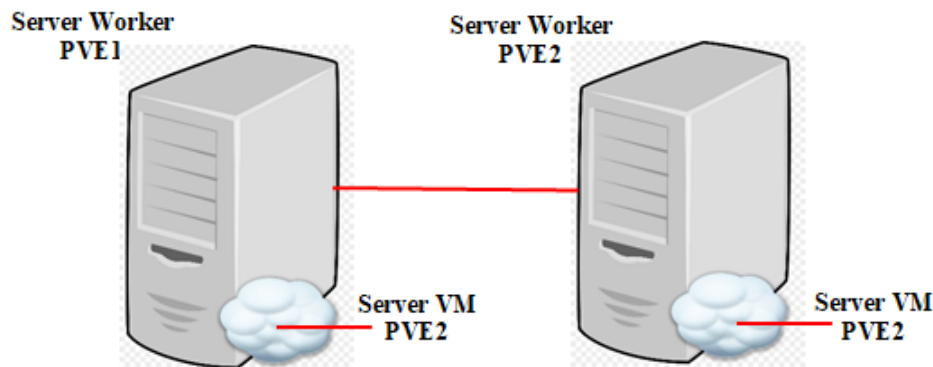
Gambar 2. Alur Penelitian

## 2.1 Studi Literatur

Penelitian dimulai dari tahapan studi literatur. Kegiatan yang dilakukan adalah mencari dan memahami semua yang berhubungan langsung dengan penelitian ini. Diantara materi-materi yang harus dipelajari dan dipahami adalah pengertian *Software Proxmox VE 6.0*, *Server Worker*, *Server Virtual Machine (VM)*, *Clustering Server Worker*, *CPU Usage*, *Powerstat Tool*, *Hitperft Tools*, *Auto-Migration*, *penggunaan Pemrograman Bash Shell*, *Secure Shell (SSH)*, *Ubuntu Server16.04*, *VMware Workstation*, *Matlab*.

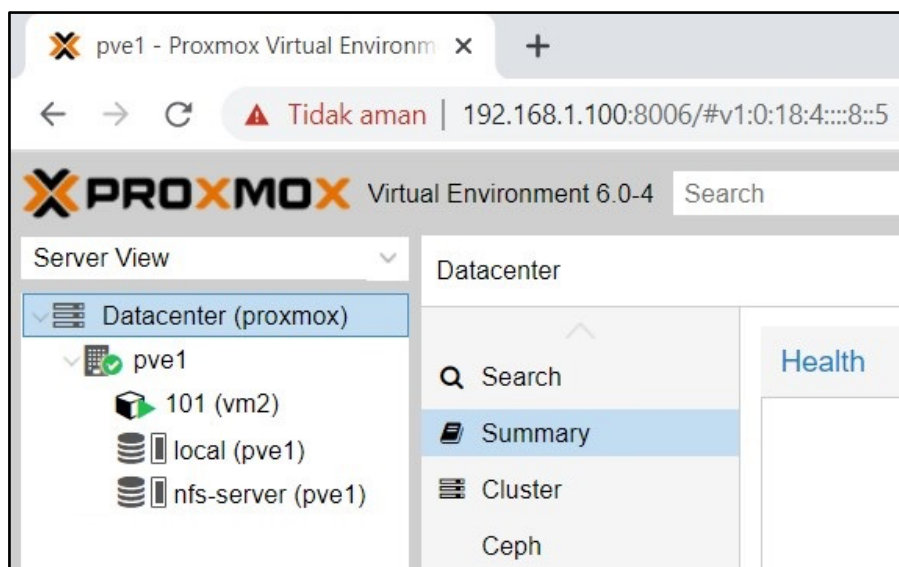
## 2.2 Membangun Server Worker

Berikutnya adalah membangun data center komputasi awan menggunakan *platform opensource Virtual Proxmox VE. 60*, *VMware Workstation Pro*, *Ubuntu Server 16.04* dan *Windows 10*. Kemudian membuat 2 *server worker* dan 2 *server virtual machine (VM)* yang dihubungkan menjadi 1 *cluster*.



Gambar 3. Topologi *Server Worker*

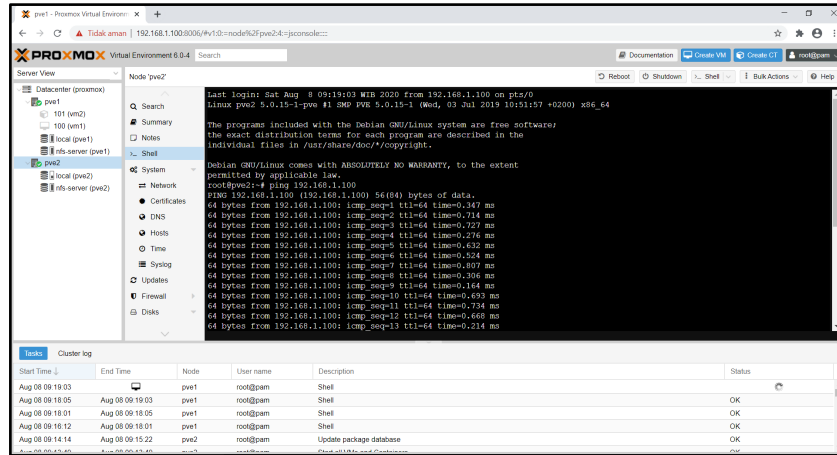
*Server Worker* dibangun secara virtual memanfaatkan platform Proxmox VE 6.0 dengan spesifikasi lengkap mesin virtual *Server Worker*. *Server VM* dibangun secara virtual juga dengan memanfaatkan sistem operasi linux ubuntu server 16.04 dengan spesifikasi lengkap mesin *virtual Server VM*. Pada gambar 4. adalah salah satu tampilan antarmuka *server worker* yang ditampilkan melalui web browser dengan menggunakan Ip address 192.168.1.100:8006. *Server Worker* terdiri dari 2 (PVE1 dan PVE2).



Gambar 4. Antarmuka *Server Worker PVE1*

## 2.3 Pengujian Koneksi Antar Server Worker

Pengujian yang dimaksud pada tahapan ini adalah pengujian menghubungkan antar *server worker* yaitu PVE1 dan PVE 2 melalui ping koneksi antar keduanya. Hasil ping antar kedua server worker bisa diperhatikan pada Gambar 5.



Gambar 5. Antarmuka Koneksi dari PVE1 ke PVE2

**2.4 Monitoring Usage CPU**

CPU atau Central Processing Unit merupakan komponen yang mengkonsumsi lebih banyak daya di server, penting untuk memahami lebih bagaimana konsumsi daya oleh host berkembang mengenai beban pada CPU tersebut [9]. tingkat penggunaan resource yang dipakai oleh aplikasi dapat dibedakan menjadi beberapa komponen seperti CPU-intensive, memory-intensive, network io-intensive dan disk storage IO-intensive. Aplikasi CPU-intensive dinyatakan relatif dalam mengkonsumsi CPU lebih besar dibandingkan dengan resource lainnya [14]. CPU usage atau Time CPU merupakan suatu method untuk mengukur kuantitas usage CPU, khususnya metode untuk mengukur kuantitas penggunaan CPU yang mampu mendapatkan jumlah kredibel penggunaan CPU tanpa mengubah algoritma dalam rangka untuk beradaptasi dengan sistem operasi, misalnya Windows atau Linux Operating System [14].

CPU usage hal yang diperhitungkan dalam penelitian ini. Karena dalam penelitian ini CPU usage akan digunakan untuk sebagai indikator apakah sever VM dapat di migrasi maupun tidak. Pengukuran usage CPU dapat dilakukan dengan berbagai cara. Pada penelitian ini penulis akan menggunakan perintah ps aux untuk memonitoring CPU usage pada setiap VM. Tampilan bila menjalankan perintah ps aux akan menampilkan beberapa informasi, seperti CPU usage, waktu mulai aplikasi, penggunaan memory, user dll. Untuk mengambil nilai dari CPU usage yang diperlukan saja (dalam bentuk %), maka dapat menggunakan perintah seperti berikut.

```
root@pve2:~# cat /proc/loadavg
```

Hasil CPU usage akan dilakukan monitoring secara otomatis dengan memanfaatkan pemrograman bash shell. Jika CPU usage server worker 1 mengalami kenaikan, maka akan server VM akan dimigrasikan ke server worker 2, jika CPU usage server worker 1 mengalami penurunan, maka server VM pada server worker 2 akan kembali lagi secara otomatis ke server worker 1. Hal ini bertujuan untuk mengoptimalkan pengalokasian server VM dalam penggunaan daya antar kedua server worker.

**2.5 Penentuan Variabel Input dan Output**

Tabel 2. Keterangan Variabel Input dan Output

Variabel	Nama Fungsi	Semesta Pembicara
CPU Usage	Input	[0-100]

Dalam menentukan migrasi suatu server VM ke server worker maka dalam penelitian ini yang menjadi nilai input hanya berdasarkan usage CPU. Dalam pembentukan himpunan fuzzy nantinya akan dilakukan berdasarkan variabel test dari tingkat CPU usage. Adapun range himpunan fuzzy seperti dideskripsikan pada Tabel 3.

Tabel 3. Parameter Penelitian Himpunan Fuzzy

Variabel	Himpunan Fuzzy	Range
CPU Usage	Rendah	[0-45]
	Sedang	[46-90]
	Tinggi	[91-100]

**2.6 Penentuan Nilai Threshold**

Pada Tabel 4, ditentukan nilai threshold sebagai dasar penilaian dari CPU usage dan menjadi indikator dalam menentukan migrasi VM. Indikator tersebut dibagi menjadi 3 nilai yaitu nilai rendah, nilai sedang dan nilai tinggi.

Tabel 4. Nilai *Threshold* serta Kecenderungan

Kriteria	CPU Usage	
	Server Worker 1	Server Worker 2
Rendah	0,0-45%	0,0-45%
Menengah	46-90%	46-90%
Tinggi	91-100%	91-100%

Dalam menentukan nilai threshold maka CPU usage menjadi indikatornya dalam menentukan migrasi VM. Penulis membagi menjadi tiga nilai yang harus ditentukan, yaitu rendah, sedang dan tinggi. Nilai rendah adalah rentang nilai terkecil yang dianggap sebagai nilai kondisi CPU usage terendah. Nilai sedang adalah nilai rentang menengah yang dianggap sebagai nilai kondisi CPU usage dalam kondisi menengah, sedangkan nilai tinggi itu merupakan nilai rentang tertinggi. Kemudian ketiga nilai tersebut akan menunjukkan proses migrasi pada VM, dimana jika kecenderungan nilainya kecil maka menunjukkan VM akan migrasi dari server worker 2 menuju server worker 1, jika sebaliknya kecenderungan nilainya tinggi maka VM akan migrasi server worker 1 menuju server worker 2. Dan jika kecenderungan nilainya menengah, maka VM tidak akan migrasi. Nilai-nilai ini adalah faktor-faktor yang dapat mempengaruhi proses dari penelitian optimasi daya data center komputasi awan melalui migrasi VM pada layanan tersebut.

**2.7 Distribusi Beban dengan *Httpperf Tool***

Pengujian didalam penelitian ini dilakukan dengan memberikan beban pada table 5. Beban berupa request dari server VM1 ke Server VM2 sehingga akan berpengaruh terhadap kondisi trafik CPU usage dari masing-masing server VM. Penelitian ini sebenarnya direncanakan selama 12 jam dengan data CPU usage yang akan diambil adalah rata-rata dari setiap 1 jam, sehingga sistem mampu dapat lebih mudah serta lebih efektif dalam kalkulasi nilai perjam nantinya. Dikarenakan keterbatasan waktu dan perangkat pengujian yang dimiliki oleh penulis, maka pola pengujian dikecilkan menjadi hanya satu jam saja dari 12 jam. Data CPU usage yang diambil adalah rata-rata dalam 5 menit dan nanti akan dihitung rata-rata dari 5 menit tersebut. Pengujian ini dimulai dari menit ke-0 sampai menit ke-30, sehingga total nilai CPU usage berjumlah 6 sampel, dan ini sama halnya seperti pengujian yang dilakukan dengan durasi 12 jam.

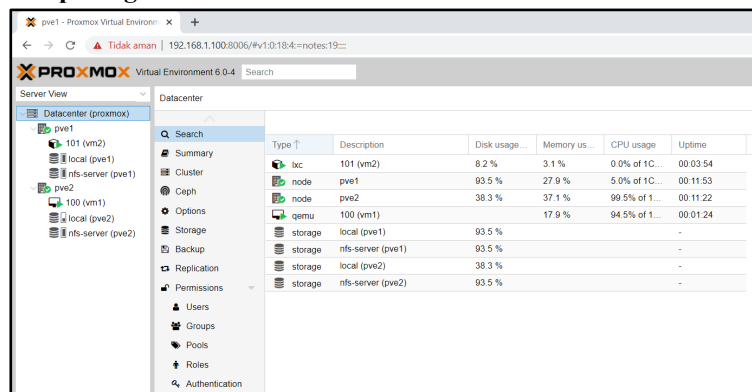
Tabel 5. Nilai Distribusi Beban

Menit/Time	Trafik/ Request (Per detik)	Jumlah Trafik (Request) per 5 Menit
00	1	0
05	6	1000
10	21	5000
15	50	10000
20	87	15000
25	121	20000
30	142	25000

**3. HASIL DAN PEMBAHASAN**

Pengujian dilakukan pada dua skenario, skenario pertama yaitu dengan menerapkan teknik migrasi otomatis dengan mamdani dan pada skenario kedua yaitu dengan teknik migrasi tanpa mamdani. Teknik migrasi adalah suatu sistem yang dibangun dengan memindahkan server VM dikarenakan beban CPU usage yang tinggi, hal ini akan dilakukan migrasi VM secara otomatis. Sedangkan sistem yang dibangun tanpa teknik migrasi, maka tidak melakukan migrasi server secara otomatis.

**3.1 Uji coba dengan tanpa migrasi otomatis**



Gambar 7. Simulasi Sistem *Server Worker*

Uji coba optimasi daya data center ini akan dilakukan selama 30 menit. 30 Menit tersebut dipilih dan dijadikan sebagai pola untuk pengujian. Simulasi sistem tersebut terdiri dari 2 server worker yaitu PVE1 dan PVE2 yang masing-masing memiliki server VM nya (Server VM1 dan Server VM2). Nantinya CPU usage pada server worker yang akan dijadikan sebagai indikator pengukuran konsumsi dayanya. Untuk mengukur konsumsi daya pada *server worker*, juga menggunakan perintah.

```
root@pve:~# Ps aux/awk '{sum=S3; print sum}'tall -n 1
```

Gambar 8 dan gambar 9 menampilkan hasil penggunaan konsumsi daya kedua server worker per 5 menit selama 30 menit.

```
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Mon Aug 17 12:38:30 2020 from 192.168.1.30
root@pve1:~# powerstat -z 300 6
Number of readings low, results may be inaccurate.
Running for 1800.0 seconds (6 samples at 300.0 second intervals).
Power measurements will start in 0 seconds time.
  Time      User      Nice      Sys  Idle      IO  Run  Ctxt/s  IRQ/s  Fork  Exec  Exit  Watts
14:31:45   3.5      0.0      5.1  90.9      0.6   1    614     306   961   805   963   10.41
^X^Z
[1]+  Stopped                  powerstat -z 300 6
root@pve1:~# powerstat -z 300 6
Number of readings low, results may be inaccurate.
Running for 1800.0 seconds (6 samples at 300.0 second intervals).
Power measurements will start in 0 seconds time.
  Time      User      Nice      Sys  Idle      IO  Run  Ctxt/s  IRQ/s  Fork  Exec  Exit  Watts
14:39:25   3.6      0.0      4.0  92.3      0.0   1    520     242   968   809   967   10.38
14:44:25   3.7      0.0      4.1  92.2      0.0   1    525     248   960   806   961   10.40
14:49:25   3.6      0.0      4.0  92.4      0.0   1    508     237   963   807   964   10.62
14:54:25   3.9      0.0      3.8  92.2      0.0   2    517     243   962   807   961   10.59
14:59:25   3.9      0.0      3.8  92.3      0.0   1    512     242   958   806   958   10.85
15:04:25   3.9      0.0      3.8  92.3      0.0   1    513     245   960   806   960   11.14
-----
Average    3.8      0.0      3.9  92.3      0.0  1.2  515.8   242.8  961.8  806.8  961.8  10.67
GeoMean   3.8      0.0      3.9  92.3      0.0  1.1  515.8   242.8  961.8  806.8  961.8  10.66
StdDev    0.1      0.0      0.1  0.1      0.0  0.4   5.4     3.3    3.2    1.1    2.9    0.27
-----
Minimum   3.6      0.0      3.8  92.2      0.0  1.0  508.2   237.1  958.0  806.0  958.0  10.38
Maximum   3.9      0.0      4.1  92.4      0.0  2.0  524.8   247.9  968.0  809.0  967.0  11.14
-----
Summary:
System:   10.67 Watts on average with standard deviation 0.27
root@pve1:~#
```

Gambar 8. Konsumsi Daya Server Worker PVE 1

```
individual files in /usr/share/doc/*/copyright.
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Mon Aug 17 12:40:28 2020 from 192.168.1.30
root@pve2:~# powerstat -z 600 6
Number of readings low, results may be inaccurate.
Running for 3600.0 seconds (6 samples at 600.0 second intervals).
Power measurements will start in 0 seconds time.
  Time      User      Nice      Sys  Idle      IO  Run  Ctxt/s  IRQ/s  Fork  Exec  Exit  Watts
14:39:21   5.3      0.0      4.0  90.6      0.0   3    1740     822   620   460   619   10.37
14:44:21   5.6      0.0      4.1  90.2      0.0   3    1741     831   592   433   592   10.59
14:49:21   5.0      0.0      3.9  91.1      0.0   3    1602     756   608   450   607   10.82
14:54:21   5.5      0.0      4.0  90.5      0.0   3    1795     855   598   442   599   10.65
14:59:21   5.5      0.0      4.0  90.5      0.0   3    1765     840   603   445   602   10.78
15:04:21   5.5      0.0      4.0  90.6      0.0   2    1811     865   601   446   603   11.48
-----
Average    5.4      0.0      4.0  90.6      0.0  2.8 1742.5   828.2  603.7  446.0  603.7  10.78
GeoMean   5.4      0.0      4.0  90.6      0.0  2.8 1741.1   827.4  603.6  445.9  603.6  10.78
StdDev    0.2      0.0      0.1  0.3      0.0  0.4   67.9    35.6   8.8   8.1   8.2   0.34
-----
Minimum   5.0      0.0      3.9  90.2      0.0  2.0 1602.4   755.5  592.0  433.0  592.0  10.37
Maximum   5.6      0.0      4.1  91.1      0.0  3.0 1811.5   865.3  620.0  460.0  619.0  11.48
-----
Summary:
System:   10.78 Watts on average with standard deviation 0.34
root@pve2:~#
```

Gambar 9. Konsumsi daya Server Worker PVE 2

### 3.2 Uji coba dengan migrasi otomatis

Jika server worker PVE1 mengalami tingginya CPU usage maka salah satu server VM akan dimigrasikan ke server worker PVE2, dan kemudian jika server worker PVE1 mengalami penurunan CPU usage, maka server VM akan dikembalikan ke server worker PVE 1 dan seterusnya. Berikut tampilan monitoring CPU usage menggunakan Bash Shell Script pada gambar 10. Berdasarkan monitoring diatas maka penggunaan konsumsi daya masing-masing *server worker* akan berbeda, dikarena jika trafik penggunaan yang di tandakan melalui tingginya penggunaan *usage CPU*, maka *server VM* akan dimigrasikan kepada masing-masing *server worker*. Dan jika *usage CPU* kedua server rendah dikarena beban penggunaan *server* yang menurun, maka kedua *server VM* pada *server worker* PVE2 akan dimigrasikan ke server worker PVE1.

```

192.168.1.100 - PuTTY
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Mon Aug 17 15:36:35 2020 from 192.168.1.30
root@pve1:~# ./cpuserver.sh
CPU load
Monitoring Ubuntu

CPU load is high
CPU load is critical!
Machine is now high requirement

Starting Migration
Configuration file 'nodes/pve1/gemu-server/100.conf' does not exist

Machine Moved to Proxmox2
^Z
[1]+  Stopped                  ./cpuserver.sh
root@pve1:~# ./cpuserver.sh
CPU load
Monitoring Ubuntu

CPU load is high
CPU load is critical!
Machine is now high requirement

Starting Migration

Machine Moved to Proxmox2
CPU load has dropped
Machine is now low requirement

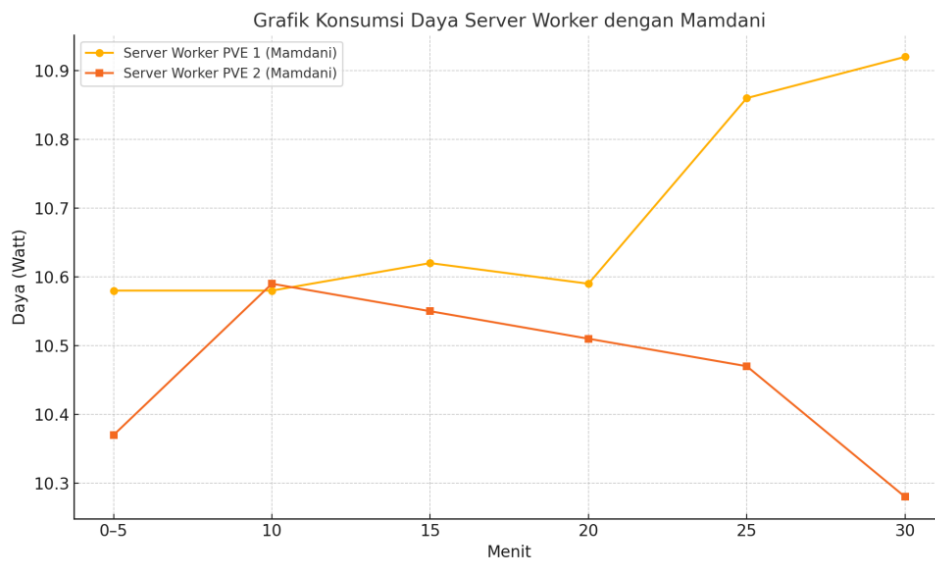
Starting Migration

Machine Moved to Proxmox1
    
```

Gambar 10. Tampilan *Monitoring Server Worker* PVE1

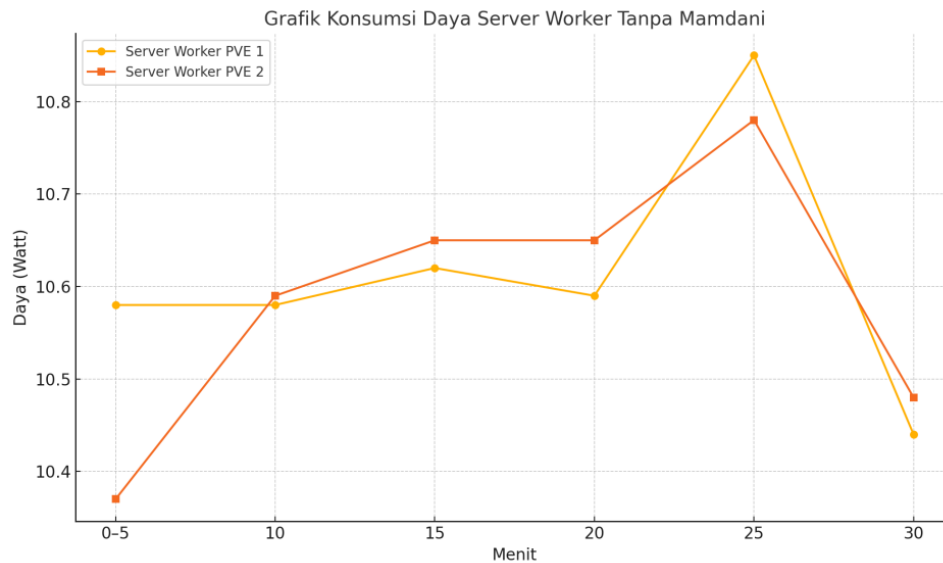
### 3.3 Analisa Data

Berdasarkan hasil pengujian, penulis ingin memaparkan perbandingan yang terjadi setelah hasil uji coba dengan teknik migrasi dengan mamdani dan tanpa mamdani. Pemaparan sangat berfokus pada nilai konsumsi daya pada masing-masing pengujian. Maka dari hasil masing-masing pengujian yang dibandingkan berdasarkan dari dua skenario uji coba yang telah ditetapkan pada tahap sebelumnya.



Gambar 11. Konsumsi Daya *Server Worker* dengan Mamdani

Berdasarkan gambar 11 dan gambar 12 adalah nilai konsumsi daya kedua pengujian yang menggunakan teknik migrasi vm secara otomatis dan sebaliknya. Maka berdasarkan nilai dari kedua tabel tersebut menunjukkan penghematan dapat dilakukan dengan teknik migrasi secara otomatis. Ini adalah hasil yang sangat diharapkan dan sesuai dengan hipotesis sebelum penulis melakukan pengujian. Jadi penulis menyatakan bahwa total konsumsi daya dengan menggunakan metode mamdani adalah sebesar 126.92 (Watt) dan adapun pada layanan data center tanpa menggunakan migrasi otomatis adalah 127.18 Watt. Maka dengan migrasi vm menggunakan metode mamdani mampu memberikan efisiensi konsumsi daya sebesar 0.26 Watt.



Gambar 12. Konsumsi Daya *Server Worker* tanpa Mamdani

#### 4. KESIMPULAN

Penelitian ini berhasil membuktikan bahwa teknik live migration virtual machine (VM) yang dikendalikan oleh metode logika fuzzy Mamdani dapat mengoptimalkan konsumsi daya pada server worker dalam lingkungan data center berbasis komputasi awan. Melalui proses simulasi yang dilakukan menggunakan platform Proxmox VE selama 30 menit, pengujian dilakukan dalam dua skenario: pertama tanpa penerapan migrasi otomatis, dan kedua dengan penerapan migrasi otomatis menggunakan fuzzy Mamdani. Hasil analisis menunjukkan bahwa skenario dengan fuzzy Mamdani menghasilkan penghematan konsumsi daya sebesar 0,26 watt dibandingkan skenario konvensional. Meski angka tersebut terlihat kecil, namun dalam skala implementasi pusat data yang besar dan operasional terus-menerus, efisiensi ini dapat berkontribusi signifikan terhadap pengurangan biaya energi dan peningkatan keberlanjutan operasional.

Faktor utama yang mempengaruhi proses migrasi adalah nilai CPU usage yang dimonitor secara real-time. Pengambilan keputusan berbasis fuzzy Mamdani memungkinkan sistem melakukan migrasi VM secara otomatis saat nilai CPU usage melewati ambang batas tertentu, serta mengembalikannya saat beban menurun. Hal ini memungkinkan distribusi beban kerja menjadi lebih adaptif dan efisien, serta menjaga server dalam kondisi idle ketika tidak diperlukan, tanpa mengganggu kualitas layanan. Dengan demikian, dapat disimpulkan bahwa penggunaan metode fuzzy Mamdani tidak hanya memberikan solusi cerdas dan fleksibel dalam manajemen beban server, tetapi juga mendukung tercapainya efisiensi energi tanpa mengorbankan kinerja sistem secara keseluruhan. Metode ini layak diterapkan dalam desain dan pengelolaan infrastruktur data center modern yang mengedepankan efisiensi, otomatisasi, dan keberlanjutan.

#### REFERENSI

- [1] P. A. Lima, A. S. B. Neto, and P. Maciel, "Data centers' services restoration based on the decision-making of distributed agents," *Telecommun. Syst.*, vol. 74, no. 3, pp. 367–378, 2020, doi: 10.1007/s11235-020-00660-2.
- [2] D. Filani, "Dynamic Data Center Power Management Trends, Issues, and Solutions," *Intel Technol. J.*, vol. 12, no. 01, pp. 59–67, 2008, doi: 10.1535/itj.1201.06.
- [3] M. M. Than and T. Thein, "Energy-Saving Resource Allocation in Cloud Data Centers," 2020 IEEE Conf. Comput. Appl. ICCA 2020, 2020, doi: 10.1109/ICCA49400.2020.9022819.
- [4] P. Arroba, J. M. Moya, J. L. Ayala, and R. Buyya, "Dynamic Voltage and Frequency Scaling-aware dynamic consolidation of virtual machines for energy efficient cloud data centers," *Concurr. Comput.*, vol. 29, no. 10, 2017, doi: 10.1002/cpe.4067.
- [5] A. Amirullah, R. M. Ijtihadie, and H. Studiawan, "Optimasi Daya Data Center Cloud Computing Pada Workload High Performance Computing (Hpc) Dengan Scheduling Prediktif Secara Realtime," *JUTI J. Ilm. Teknol. Inf.*, vol. 15, no. 1, p. 1, 2017, doi: 10.12962/j24068535.v15i1.a630.
- [6] S. Soni, A. Abdurrahman, and Afdhil Hafid, "Optimalisasi Sumber Daya Komputer Dengan Virtualisasi Server Menggunakan Proxmox Ve," *J. Fasilkom*, vol. 9, no. 2, pp. 369–376, 2019, doi: 10.37859/jf.v9i2.1394.
- [7] Anita Rizky Agustina, Fajar Pradana, and Fitra Abdurrachman Bachtiar, "Pengembangan Aplikasi Pembelajaran dengan Menerapkan Model Pembelajaran Teams-Games-Tournament (TGT)," *J. Nas. Tek. Elektro dan Teknol. Inf.*, vol. 10, no. 2, pp. 111–117, 2021, doi: 10.22146/jnteti.v10i2.1310.
- [8] A. V. Kumar, V. Krishnakumar, and A. N. Kumar, "Efficient performance upsurge in live migration with downturn in the migration time and downtime," *Cluster Comput.*, vol. 22, pp. 12737–12747, 2019, doi: 10.1007/s10586-018-1753-7.
- [9] J. F. Zhao and J. T. Zhou, "Strategies and methods for cloud migration," *Int. J. Autom. Comput.*, vol. 11, no. 2, pp. 143–152, 2014, doi: 10.1007/s11633-014-0776-7.
- [10] P. G. Jeba Leelipushpam and J. Sharmila, "Live VM migration techniques in cloud environment - A survey," 2013 IEEE Conf.

- Inf. Commun. Technol. ICT 2013, no. 1ct, pp. 408–413, 2013, doi: 10.1109/CICT.2013.6558130.
- [11] M. Ahkam, R. Munadi, and T. Y. Arif, “Precopy Live Migration on Core Network Virtualization Internet Protocol Multimedia Subsystem,” *J. Aceh Phys. Soc.*, vol. 8, no. 2, pp. 55–60, 2019, doi: 10.24815/jacps.v8i2.12613.
- [12] J. Nasir, J. Suprianto, P. Studi, T. Informatika, and U. Putera, “Analisis Fuzzy Logic Menentukan,” *J. Edik Inform.*, vol. 2, pp. 177–186, 2017.
- [13] M. Abrori and A. H. Prihamayu, “Aplikasi Metode Fuzzy Mamdani dalam Pengambilan Keputusan Penentuan Jumlah Produksi,” *Kaunia*, vol. XI, no. 2, p. 92306, 2015.
- [14] A. Fadil and W. Wibisono, “Pemilihan Dan Migrasi Vm Menggunakan Medm Untuk Peningkatan Kinerja Layanan Pada Cloud Computing,” *Kinetik*, vol. 1, no. 2, pp. 79–90, 2016, doi: 10.22219/kinetik.v1i2.26.
- [15] A. Andreas, M. H. Purnomo, and M. Hariadi, “Penerapan Logika Fuzzy untuk Pembentukan Sutradara Otonom dalam hal Pencahayaan pada Machinima (Application of Fuzzy Logic to Develop Autonomous Directors for Lighting in Machinima),” *J. Nas. Tek. Elektro dan Teknol. Inf.*, vol. 9, no. 1, pp. 45–54, 2020.