

THIS SECTION IS FILLED BY THE AUTHOR

PLEASE FILL THE FORM BELOW AND DO NOT REMOVE THIS FROM MANUSCRIPT

Authors Submission Checklist

All manuscripts must meet the following criteria to be processed into the peer-review stage. Failure to meet the following criteria may increase the processing time of the manuscript or even rejection.

Write *OK* if the authors have confirmed that the manuscript meets the criteria. If the author cannot fulfill these criteria, give a reason, and write in the checklist with highlighted text.

1	Template and Layout	Authors Claim & Comment	Editors Comment
1.1	Have used the latest Sinkron template (latest update June, 2021)	√	
1.2	Introduction, Literature Review Methods, Result, Discussion, and Conclusion are written in a separate section.	√	
1.3	Not using Bullet / Numbering in the manuscript text.	√	
1.4	The length of the text is between 6 and 12 pages.	√	
2	Title and Authors Information		
2.1	The title is written briefly, describing the contents and contributions of the manuscript. The title also does not use abbreviations or acronyms except the standard ones. The title consists of a maximum of 14 words.	√	
2.2	All authors have been written in the manuscript and also in the OJS system. One of the authors who corresponded through the OJS system was determined as the correspondence authors.	√	
2.3	All authors must include e-mails in the manuscript and the OJS system.	√	
2.4	Affiliation information is written in 1 line in the text, at least showing the name of the institution and country. Full Address information should be written in 1 line bellow affiliation information.	√	
2.5	The title of the manuscript is written using 'Capitalize Each Word', except conjunctions using lowercase.	√	
2.6	The place and time of research do not need to be included in the title unless the research is specific to a particular time and place.	√	
3	Abstracts and Keywords		
3.1	Abstract consists of 200 to a maximum of 250 words.	√	
3.2	Abstracts are arranged in a structured manner. Consisting of Background, Objective, Methods, Results, and Conclusion	√	
3.3	The number of keywords minimal 5 words	√	
3.4	All keywords must be written in the abstract.	√	
3.5	Keywords are arranged in alphabetical order.	√	
3.6	A combination of all keywords represents the content, contribution, or purpose of the text.	√	
3.7	Keywords can be in the form of research subjects' category. Avoid using adjectives as keywords in the text.	√	
4	Introduction		
4.1	Has explained the context/object of research and must focus following the scope of the title.	√	
4.2	Has explained why this research is important.	√	

*name of corresponding author



This is anCreative Commons License This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

4.3	It has included the existing research gap, explaining the unique differences between this study compared to similar previous studies.	√	
4.4	Has written research objectives/contributions explicitly.	√	
4.5	The length of this section is between 300 – 700 words.	√	
5	Literature Review		
5.1	This section is optional. This section may only be available if the explanation related to previous research is not sufficient if it is only shown in the introduction. If this section exists, you must cite at least 6 recent papers that are closely related to the title of the paper.		
6	Methods		
6.1	Has cited a source who helped the authors in the selection of methods.	√	
6.2	Has explained how data is obtained.	√	
6.3	The procedure has been written chronologically and clearly.	√	
6.4	The authors have written how the results will be measured, tested, and evaluated.	√	
7	Results		
7.1	The results of the study are written in logical sequences, according to the research flow.	√	
7.2	The results of the study are presented in the form of narrative/textual, tables, or images in the form of graphs or diagrams. Avoid displaying raw data.	√	
8	Discussion		
8.1	An explanation of the results and in-depth analysis has been written.	√	
8.2	A comparison of results with other studies has been written.	√	
8.3	Limitation of study or threat to validity has been written.	√	
9	Conclusion		
9.1	It has been explicitly written down the contribution of the manuscript and the answers to the research questions.	√	
10	Acknowledgments		
10.1	If it is funded research, information on the funders agency must be provided along with the contract number(if any).		
11	References		
11.1	Always use automatic citation tools for managing your references. make sure that you already double-check the citation metadata, automatic citation tool usually has metadata errors. Use Mendeley to make Citation & References/Bibliography. Use APA ^{6th} Edition citation style.	√	
11.2	Make sure all information related to the reference is written completely (eg Who, Where, and When). Check from the primary source for this information. write the name of the journal without abbreviations. please include DOI if it's available.	√	
11.3	Have cite a minimum of 15 references. a minimum of 80% comes from journal articles/conferences. a minimum of 80% must be published in the last 5 years.	√	

*name of corresponding author



This is anCreative Commons License This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

11.4	More than 50% of the references must come from credible sources. for example Sinkron, ScienceDirect, IEEE Explorer, ACM Digital Library, Springer Link, Scopus, Clarivate Analytics, and Indonesian national accredited journals (SINTA 1/2).	√	
12	Tables and Figures		
12.1	Already using the appropriate format in displaying images and tables.	√	
12.2	The image or graph must be self-explanatory, so it does not depend on the text.	√	
12.3	The minimum image resolution is not specified, but must be visible when published online (colored) or printed (black and white).	√	
12.4	Images do not have copyright issues.	√	
12.5	All images and tables have been referenced in the manuscript.	√	
12.6	The maximum number of images and tables is respectively 6. giving the reason to the editor if you need to display more than the recommended amount.	√	
12	Languages		
12.1	There are no spelling mistakes, typos, and grammatical errors. You can use tools such as Grammarly or recommendations provided by MS Word for language checking.	√	
12.2	Use direct and short sentences. Long sentences do not make writing more professional; they only confuse readers.	√	

*name of corresponding author



This is an Creative Commons License This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

This Section is filled by The Author

PLEASE FILL THE FORM BELOW AND DO NOT REMOVE THIS FROM MANUSCRIPT

OUTLINING HOW THE ISSUES ARE ADDRESSED

Title of paper: IMPLEMENTASI SCALING DOCKER DAN KUBERNETES MENGGUNAKAN METODE HORIZONTAL SCALER UNTUK LAYANAN WORDPRESS

1. Address all the concerns/recommendations of the reviewers.
2. All amendments made are to be **highlighted in yellow color** in the revised paper.

Reviewer # 1

Final Recommendation (*choose one according to the reviewer's comments and cross unnecessary*) :

- Accepted without modification
- Accepted with minor corrections
- Accepted with major modification
- Rejected

Comments	Addressed (Y/N)	Reply/Action taken

(Please add more rows if needed)

Reviewer # 2

Final Recommendation (*choose one according to the reviewer's comments and cross unnecessary*) :

- Accepted without modification
- Accepted with minor corrections
- Accepted with major modification
- Rejected

Comments	Addressed (Y/N)	Reply/Action taken

*name of corresponding author



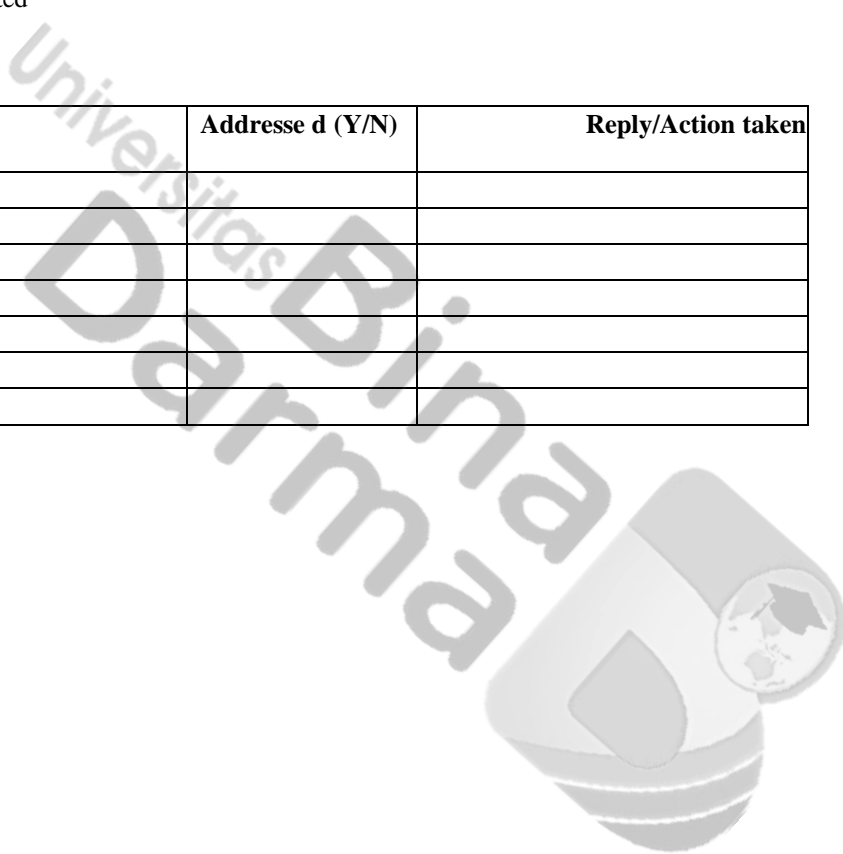
This is an Creative Commons License This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

Reviewer # 3

Final Recommendation (*choose one according to the reviewer's comments and cross unnecessary*) :

- Accepted without modification
- Accepted with minor corrections
- Accepted with major modification
- Rejected

Comments	Addressed (Y/N)	Reply/Action taken



*name of corresponding author



This is anCreative Commons License This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.



*name of corresponding author



This is anCreative Commons License This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

Implementation Docker and Kubernetes Scaling Using Horizontal Scaler Method for Wordpress Services

Suryayusra¹⁾, Nova Destarina^{2)*}, Edi Surya Negara³⁾, Edi Supratman⁴⁾, Maria Ulfa⁵⁾

¹⁾²⁾³⁾⁴⁾⁵⁾ informatics engineering, Universitas Bina Darma Palembang, Indonesia

¹⁾suryayusra@binadarma.ac.id , ²⁾novadestarina092@gmail.com , ³⁾e.s.negara@binadarma.ac.id ,

⁴⁾edi_supratman@binadarma.ac.id , ⁵⁾mu@binadarma.ac.id

Submitted :xxxxx | **Accepted** : xxxx | **Published** : xxxxx

Abstract: Container is a technology that has recently been widely used because of the additional features that are very easy and convenient to use, especially for web hosting service developers, with Container making it easier for system admins to manage applications including building, processing and running applications on Container. With Container the process of creating and using the system will be easier but along with too many user requests so that the service does not run optimally. Therefore, the Container must have good scalability and performance. Scalability is needed for systems that can adjust to the needs of user demand and performance is needed to maintain the quality of services provided. This research aims to implement scaling using Docker and Kubernetes in terms of scalability and performance. The parameters of comparison between Docker and Kubernetes are for scalability, scaling up and scaling down time and for performance. The method in this research uses the Action Research methodology, which is a research model that is simultaneously practiced and theorized. With the initial steps of problem identification, action planning, action implementation, observation and evaluation. Based on the results that have been obtained, Docker consumes more CPU & Memory Usage Resources, namely at 500 Users Kubernetes consumes Resources with an average of 94.47%-4.70% while in Kubernetes 89.11%-4.50 because in Kubernetes itself has a complex system, especially special component components such as APIs, Metrics Server, Kubernetes manager to run the Container. While in Docker only has Docker Manager and Docker Compose components.

Keywords: Container, Docker, Kubernetes, Load Testing, Scaling.

INTRODUCTION

Docker and Kubernetes have emerged as solutions to address infrastructure scalability. Docker enables the packaging of applications into containers. Docker emerged as a response to problems arising in the software development and deployment process. While Docker addressed the problem of packaging and running applications, Kubernetes was born to address the challenges of Container management at scale. By using Containers, the need arises to efficiently manage 2 concurrently running Containers (Putri et al., 2021). Containers are isolated environments that run on top of the same operating system but do not share address space with each other (Silva et al., 2019).

Kubernetes provides a solution to this problem by providing an open-source platform for Container application scalability. With Kubernetes, users can manage their applications in a secure, scalable way. Kubernetes provides a robust platform for scaling Containers (Rina & Ridha, 2021). One of the most widely used Container-based virtualization today is Docker. Docker is one of the useful tools for running containers. However, managing containers to create many services and handle many users is a big task for Docker (Sugiyatno & M, 2023). In Kubernetes, a Container is a Pod the smallest deployment unit that can contain one or more Containers. The Pod provides an execution environment that includes a shared network namespace, File system, and storage volumes (Zulfikar, 2022).

The purpose of this research is to compare the performance of the scaling process in Docker and Kubernetes. In Kubernetes using Horizontal Scaler method while in Docker using the same Horizontal Scaler method. The

*name of corresponding author



This is anCreative Commons License This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

comparison parameters between Docker and Kubernetes are Load Testing for scalability, Scaling up and Scaling down time for performance.

LITERATURE REVIEW

Scaling is the main problem in this case. Inadequate resources that are unable to cope with strong changes in workload over time, in which case the application experiences low performance or too much user demand, in which case the utilization of resources allocated to the Container is low. Therefore, scaling over time is required (Subhi et al., 2021). Load Testing is a performance testing technique where system response is measured under various conditions and loads. This test helps determine how software behaves when multiple users access the software simultaneously. Scalability is the ability of a Container to handle an increasing number of users. Scalability can be achieved through existing configurations or by adding Containers, thus allowing new users access to the application without significant delays (Yedutun et al., 2019). Performance is the result of work that can be achieved. In the context of information technology, it refers to how well a system or application operates in terms of responsiveness, speed, efficient use of resources, and ability to handle a given workload (Aruan & Rahayu, 2023).

Load Testing is a critical approach in measuring and evaluating the performance of an application. In the context of Load Testing performance analysis, especially when faced with a high workload (Andrianto & Suyatno, 2024). The Central Processing Unit (CPU), often referred to as the processor or the brain of the computer, has the main function of processing and managing all the calculations and commands that allow the computer to operate. Since the CPU generates heat during its work process, it is usually equipped with fans and heat sinks to keep the temperature down (Rivki et al., 2019). Web hosting is a service provided by a web hosting provider that allows users or others to store and publish websites or web applications on the internet (Yosli, 2021).

The previous research related to comparing the performance of the Scaling process on Kubernetes and Docker Swarm. In Kubernetes using Horizontal Pod auto scaler method while in Docker Swarm using Scaling method. The comparison parameters between Kubernetes and Docker Swarm are Load Testing for scalability, Scaling up and Scaling down time for performance (Firdaus et al., 2020). Further previous research on Comparative Analysis of Docker Swarm Web Server Performance with Kubernetes Cluster, Before testing was carried out on two Cluster servers with different orchestration tools. Both Cluster servers are built on Google Cloud Platform with the same server specifications. Then docker swarm manager is connected to two swarm nodes, Kubernetes Cluster control plane is connected to two nodes. After that, install Nginx as a web server on the swarm manager and control plane, and replicated to other nodes. Both Cluster servers are then load tested using the Apache Jmeter application on a laptop that has been prepared. The measurement data results are organized based on the parameters used in this study (Prasetyo & Salimin, 2021). The next previous research with the title Implementation of Optimistic Concurrency Control on E-Commerce Application Systems Based on MicroServices Architecture Using Kubernetes, Testing is implemented with the Load Testing method. In Load Testing, several parameters need to be considered. The results obtained are in the form of a comparison of the amount of data. Experiments were carried out with several trials with different parameters (Ammar Dwi Anwari et al., 2021).

METHOD

The method in this study uses the Action Research methodology, which is a research model that is at the same time practicing and theorizing, or combining theory and implementing it in practice (Hasan, 2019).

*name of corresponding author



This is anCreative Commons License This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

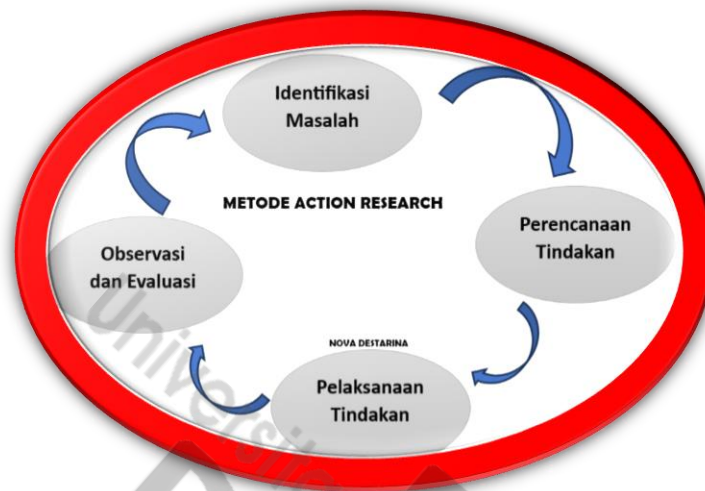


Figure 1. Action Research Methodology

In summary, the stages in Action Research consist of a cycle of problem identification is to identify problems or challenges related to the scalability of wordpress web hosting services using Docker and Kubernete, action planning is to plan steps to implement Scaling on Docker and Kubernetes, action implementation is The planned steps are then implemented in a production environment using Docker and Kubernetes, and observation and evaluation of actions Observing system performance after the implementation of Scaling steps. collecting data related to performance and Scaling on the system for evaluation. This entire cycle is conducted based on the context and objectives of the action research being conducted.

Some tools and hardware that will be used in the research are the specifications of the hardware used, among others :

Table 1. Hardware Specifications

Komponen	Spesifikasi
Nama Perangkat	Laptop Acer Aspire Lite 14
Processor	12th Gen Intel(R) Core(TM) i3-1215U 1.20 GHz
RAM	8 GB memory
HDD	512 GB HDD

The software to be used includes :

1. Docker Desktop as Software.
2. Docker as Container controller.
3. Kubernetes as Container controller.
4. Metric Server as monitoring in kubernetes.
5. Apache Jmeter as stressing tool for Wordpress web hosting.

RESULT

This section displays the results of the analysis. As explained earlier, the parameters chosen to be measured are Load Testing for scalability, Scaling up and Scaling down time for performance. The purpose of Load Testing is to see the scalability of each Container. By looking at the load on CPU & Memory usage with the specified User. CPU & Memory usage shows what percentage is needed in the process carried out in the experiment.

*name of corresponding author



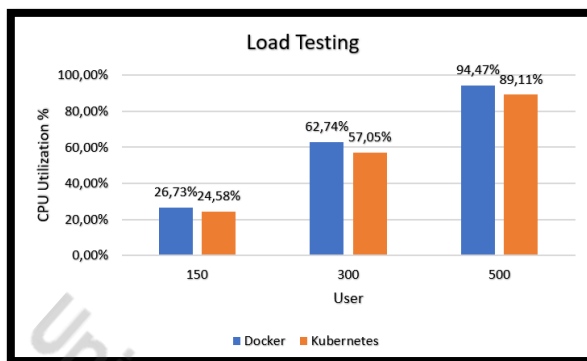


Figure 2. CPU Usage Load Testing Diagram

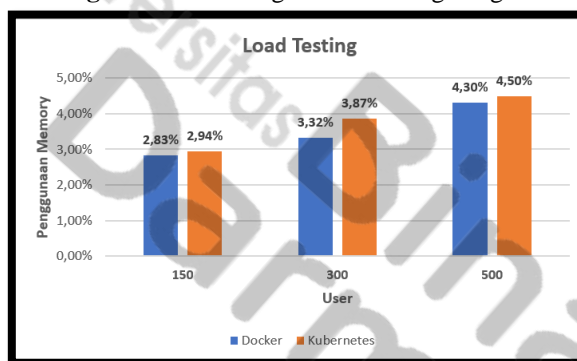


Figure 3. Memory Usage Load Testing Diagram

Seen from the picture the average results of Load Testing when generating 150 users consumes CPU & Memory usage with an average of 26.73%-2.83% while on Kubernetes 24.58%-2.94%. Furthermore, when generating 300 Users consumes CPU usage with an average of 62.74%-3.32% while in Kubernetes 57.05%-3.87%. Furthermore, when 500 users take up CPU usage with an average of 94.47%-4.30% while in Kubernetes 89.11%-4.50. Because in Kubernetes itself has a complex system, especially special components such as Metric Server, Kubernetes manager, and kubectl to run the Container. While in Docker only has Docker Manager and Docker Compose components.

The purpose of scaling up is to see the performance of each Container. By looking at the Container Creating time and the Container Load Testing time. The purpose of Scaling down is to see the performance of each Container. By measuring CPU usage when the service is not in use.

```
PS D:\SCALING DOCKER> docker-compose up -d
time="2024-08-30T20:51:17+07:00" level=warning msg="D:\\SCALING DOCKER\\docker-compose.yml: 'version' is obsolete"
[+] Running 5/5
 ✓ Network scalingdocker_default      Created          0.0s
 ✓ Volume "scalingdocker_wordpress_data" Created          0.0s
 ✓ Volume "scalingdocker_db_data"     Created          0.0s
 ✓ Container scalingdocker-db-1       Started          0.4s
 ✓ Container scalingdocker-wordpress-1 Started          0.7s

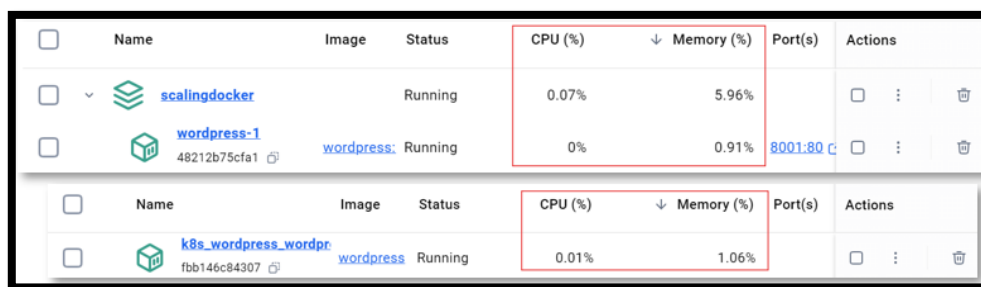
PS D:\SCALING KUBERNETES> kubectl get pods
NAME                                READY   STATUS    RESTARTS   AGE
mysql-678cf5cdf5-9mzjw             1/1    Running   16 (3d23h ago)  10d
wordpress-78cd998686-snw8z        1/1    Running   0           6s
```

Figure 4. Docker built in 0.7s & Kubernetes built in 6s

Creating a Docker Container takes 0.7s (0.7 seconds) while creating a new Kubernetes Pod (Container) on a node takes 6s (6 seconds).

*name of corresponding author





	Name	Image	Status	CPU (%)	Memory (%)	Port(s)	Actions
	scalingdocker		Running	0.07%	5.96%		
	wordpress-1 48212b75cfa1	wordpress	Running	0%	0.91%	8001:80	

	Name	Image	Status	CPU (%)	Memory (%)	Port(s)	Actions
	k8s_wordpress_wordpr fbb146c84307	wordpress	Running	0.01%	1.06%		

Figure 5. CPU and Memory utilization in Docker and Kubernetes when services are no longer in use

When the Request Generator finishes generating the User. Pod (Container) or service will return to 0%. and CPU & Memory Usage of each node will return to normal.

DISCUSSIONS

The results of the analysis of all tests carried out in this final project research research can draw the following conclusions :

1. Based on the results that have been obtained, Docker consumes more CPU & Memory Usage Resources, namely at 500 Users Kubernetes consumes Resources with an average of 94.47%-4.70% while in Kubernetes 89.11%-4.50 because in Kubernetes itself has a complex system, especially special component components such as API, Metrics Server, Kubernetes manager to run the Container. While in Docker only has Docker Manager and Docker Compose components.
2. Based on the results that have been obtained Scaling up in Kubernetes is superior because of scaling but in terms of Restart Docker is faster, namely with an average time of 2.1 seconds while Kubernetes 27 seconds. For Scaling down Docker is favored in terms of removing the Container. Because the removal is done with an average time of 1.4 seconds. Although Kubernetes looks longer in deleting but in Kubernetes there is a Container deletion with an average time of 11 seconds. Based on the data above, Kubernetes is more suitable for companies that deal with unpredictable users such as e-commerce, namely Lazada, Shopee, Tokopedia, Olx, Amazon. While Docker is more suitable for companies that deal with predictable users such as internal companies with a certain number of employees.

CONCLUSION

This research certainly still has shortcomings and can be explored further. Therefore, the author provides several suggestions, namely :

1. The scaling process can be improved by using other methods to utilize CPU and memory usage metrics, so that the scaling process can be more effective.
2. Comparing Containers by combining the Scaling process with other processes so as to get which Container is the best

REFERENCES

- Ammar Dwi Anwari, Rizky Januar Akbar, & Royyana Muslim Ijtihadie. (2021). Implementasi Optimistic Concurrency Control Pada Sistem Aplikasi E-Commerce Berdasarkan Arsitektur Microservices Menggunakan Kubernetes. *Jurnal Teknik Its* , 10(2).
- Andrianto, L. D., & Suyatno, D. F. (2024). Analisis Performa Load Testing Antara Mysql Dan Nosql Mongodb Pada RestAPI Nodejs Menggunakan Postman. *Journal of Emerging Information System and Business Intelligence (JEISBI)*, 5(1), 18–26. <https://ejournal.unesa.ac.id/index.php/JEISBI/article/view/58157%0Ahttps://ejournal.unesa.ac.id>
- Aruan, M. C., & Rahayu, W. (2023). Analisis Performa Algoritma Kompresi Data dalam Penyimpanan dan Transfer Data. *LANCAH: Jurnal Inovasi Dan Tren*, 1(2), 228–232.
- Firdaus, B. A., Suryani, V., & Karimah, S. A. (2020). Analisis Performansi Proses Scaling pada Kubernetes dan Docker Swarm Menggunakan Metode Horizontal Scaler. *E-Proceeding of Engineering*, 7(2), 7793–7808.
- Hasan. (2019). Action Research : Desain Penelitian Integratif untuk Mengatasi Permasalahan Masyarakat. *AKSES: Jurnal Ekonomi Dan Bisnis*, 4(8), 12.

*name of corresponding author



This is anCreative Commons License This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

<https://publikasiilmiah.unwahas.ac.id/index.php/AKSES/article/view/523>

- Prasetyo, S. E., & Salimin, Y. (2021). Analisis Perbandingan Performa Web Server Docker Swarm dengan Kubernetes Cluster. *CoMBInES - Conference on Management, Business, Innovation, Education and Social Sciences*, 1(1), 825–833. <https://journal.uib.ac.id/index.php/combines/article/view/4512>
- Putri, S. N., Arif, M., & Ridha, F. (2021). *Implementasi Clustered Container Dengan Docker Sarm*. 201–208.
- Rina, N. K. S., & Ridha, M. A. F. (2021). The Implementasi Kubernetes Cluster Menggunakan KVM. *ABEC Indonesia*, 209–217. <https://abecindonesia.org/proceeding/index.php/abec/article/view/151%0Ahttps://abecindonesia.org/proceeding/index.php/abec/article/download/151/149>
- Rivki, M., Bachtiar, A. M., Informatika, T., Teknik, F., & Indonesia, U. K. (2019). *No 主観的健康感を中心とした在宅高齢者における健康関連指標に関する共分散構造分析Title. 112.*
- Silva, V. G. da, Kirikova, M., & Alksnis, G. (2019). Containers for Virtualization: An Overview. *Applied Computer Systems*, 23(1), 21–27. <https://doi.org/10.2478/acss-2018-0003>
- Subhi, R., Ruslianto, I., Ristian, U., Reayasa, J., Komputer, S., Mipa, F., Tanjungpura, U., Prof, J., Hadari, H., & Pontianak, N. (2021). Implementasi Teknik Scaling Pada Sistem Manajemen Balancing Server Berbasis Website. *Jurnal Komputer Dan Aplikasi*, 09(02), 316–326.
- Sugiyatno, & M, I. (2023). Analisis Perbandingan Performansi Respon Waktu Web Server dan Failover Antara Kubernetes Dan Docker Swarm pada Container Orchestration. *Jurnal Informatika Komputer, Bisnis Dan Manajemen*, 21(3), 43–53. <https://doi.org/10.61805/fahma.v21i3.9>
- Yedutun, K., Noertjahyana, A., & Novianus Palit, H. (2019). Implementasi Container Kubernetes untuk Mendukung Scalability. *Jurnal Infra*, 7(2), 1–5.
- Yosli, R. (2021). Meningkatkan Kapasitas Hosting, Mengelola Content Management System Untuk Kenyamanan Memakai Website Berbayar. *JAVIT: Jurnal Vokasi Informatika*, 31–37. <https://doi.org/10.24036/javit.v1i2.6>
- Zulfikar, A. (2022). *Penggunaan Docker Dan Kubernetes Pipeline Dalam Pengembangan Aplikasi Prediksi Cacat Perangkat Lunak Melalui Pendekatan Mlops.*

Nova Destarina, Implementation Docker and Kubernetes Scaling Using Horizontal Scaler Method for Wordpress Services

Pemberitahuan

[Sinkron] Letter of Acceptance (LoA)

2024-09-13 06:07 PM

Sinkron
JURNAL & PENELITIAN TEKNIK INFORMATIKA

e-ISSN : 2541-2019, p-ISSN : 2541-044X

Dear Author:

Nova Destarina, Suryayusra, Edi Surya Negara, Edi Supratman, Maria Ulfia:

We have reached a decision regarding your submission to Sinkron : jurnal dan penelitian teknik informatika, "Implementation Docker and Kubernetes Scaling Using Horizontal Scaler Method for Wordpress Services".


Our decision is to: **Accept Submission and publish to Volume 8 Issue 4, October 2024**

Online manuscripts are available at the link <https://jurnal.polgan.ac.id/index.php/sinkron/article/view/14091> in the period described above.

Thus this decision letter is made and is an Official Letter of Acceptance (LoA)

If further discussion is needed, please contact the Editor on Whatsapp +6282251583783

Best regards,



Sinkron
POLITEKNIK GANESHA MEDAN

Muhammad Kholruddin Harahap
Editor in chief

Note: This email is official and we do not provide LoA in PDF form. Please Print as PDF this email if you need it.

Sinkron (Penelitian dan Jurnal Teknik Informatika)
Politeknik Ganesha Medan
Teknik Informatika Department
Jl. Veteran No. 194, Pasar VI, Labuhan Deli, Deli Serdang
North Sumatera, Indonesia

sinkron:jurnal.dan.penelitian.teknik.informatika


Suryayusra M. Lion

[/index.php/sinkron/authorDashboard/submission/14091](https://index.php/sinkron/authorDashboard/submission/14091)

1.

*name of corresponding author



This is anCreative Commons License This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.