JURNAL PASTI
Jurnal Sistem Teknologi Informasi

Vol. 1 No. 1 (2025): Mei 2025

e-ISSN: 0000-0000

RANCANG BANGUN CLOUD STORAGE BERBASIS DOCKER CONTAINER

Penulis	**Zaenal Mutaqin Subekti[1], Domo Pranowo Kuswandono[1], Haryono1, Widiyawati[1], Verawati[2], Aliy Hafiz[2], Hidayatul Akmal[1]**
Institusi	[1]Universitas Bani Saleh [2]Institut Teknologi Bisnis dan Bahasa Dian Cipta Cendikia
Email Korespondensi	zaenalms@ubs.ac.id

Riwayat Naskah	
Diterima	1 Mei 2025
Revisi	10 Mei 2025
Rilis	25 Mei 2025

Abstrak

Penelitian ini bertujuan untuk merancang, mengimplementasikan serta menguji kelayakan sistem cloud storage berbasis Docker container menggunakan layanan ownCloud sebagai solusi penyimpanan data di sekolah. Sistem ini dirancang untuk memudahkan seluruh elemen seklah, seperti guru dan staf untuk mengakses, menyimpan, dan membagikan data secara efisien. Pada implementasinya, para pengguna dapat melakukan berbagai fungsi seperti mengunggah, mengunduh, dan berbagi file melalui aplikasi ownCloud. Setiap pengguna dapat mengunggah file secara mandiri, dan file tersebut hanya dapat diakses oleh pengunggah kecuali admin memberikan hak akses kepada pengguna lain. fitur keamanan ini memastikan bahwa data pengguna tetap private kecuali ada izin khusus dari admin. Selain itu, pengguna tidak dapat melihat folder milik pengguna lain tanpa adanya hak akses yang diberikan. Jika beberapa pengguna memiliki hak akses yang sama dalam grup tertentu, mereka dapat berbagi file dalam folder bersama yang telah diatur hak aksesnya, memungkinkan kolaborasi yang lebih efisien antar anggota. Pengujian yang dilakukan terhadap sistem mencakup beberapa aspek penting, seperti kecepatan akses, kemudahan penggunaan, serta skalabilitas sistem. Hasil pengujian menunjukkan bahwa sistem cloud storage yang dikembangkan mampu beroperasi dengan baik dalam berbagai skenario penggunaan. Hasil ini menunjukkan bahwa sistem cloud storage yang dikembangkan layak untuk digunakan disekolah dan berhasil. Selain itu, sistem ini memberikan dampak positif terhadap proses manajemen administrasi disekolah, dengan kemudahan akses dan kolaborasi yang lebih baik antara seluruh pihak elemen disekolah.

Kata Kunci ***Cloud Storage, ownCloud, Docker Container***

DOI

http://doi.org/10.37932/10.9001/jurnalpasti.v1i1.968

Abstract

This study aims to design, implement and test the feasibility of a Docker container-based cloud storage system using the ownCloud service as a data storage solution in schools. This system is designed to facilitate all school elements, such as teachers and staff, to access, store and share data efficiently. In its implementation, users can perform various functions such as uploading, downloading and sharing files through the ownCloud application. Each user can upload files independently, and the files can only be accessed by the

Karya ini dilisensikan di bawah Creative Commons Attribution-ShareAlike 4.0 International License

JURNAL PASTI
Jurnal Sistem Teknologi Informasi

Vol. 1 No. 1 (2025): Mei 2025

e-ISSN: 0000-0000

uploader unless the admin grants access rights to other users. This security feature ensures that user data remains private unless there is special permission from the admin. In addition, users cannot view folders belonging to other users without the granted access rights. If multiple users have the same access rights in a particular group, they can share files in a shared folder that has been set for access rights, allowing for more efficient collaboration between members. Tests conducted on the system cover several important aspects, such as access speed, ease of use, and system scalability. The test results show that the developed cloud storage system is able to operate well in various usage scenarios. These results indicate that the developed cloud storage system is feasible for use in schools and is successful. In addition, this system has a positive impact on the administrative management process in schools, with easier access and better collaboration between all elements in the school.

Keywords ***Cloud Storage, ownCloud, Docker Container***

I. PENDAHULUAN

Perkembangan teknologi informasi yang pesat telah mendorong berbagai institusi, termasuk lembaga pendidikan, untuk mengadopsi sistem digital dalam mendukung aktivitas sehari-hari. Salah satu tantangan utama yang dihadapi sekolah saat ini adalah bagaimana mengelola dan menyimpan dokumen digital secara terpusat, aman, dan mudah diakses oleh seluruh civitas akademika seperti guru dan staf administrasi. Penggunaan media penyimpanan konvensional seperti flashdisk atau hard drive eksternal dinilai tidak efisien, rentan terhadap kehilangan data, dan tidak mendukung akses kolaboratif. Cloud storage menjadi solusi ideal untuk permasalahan tersebut, karena memungkinkan pengguna menyimpan, berbagi, dan mengakses data secara daring dari berbagai perangkat. Namun, layanan cloud komersial seperti Google Drive atau Dropbox seringkali membutuhkan biaya berlangganan dan menimbulkan kekhawatiran atas privasi serta kontrol terhadap data. Oleh karena itu, dibutuhkan solusi cloud storage mandiri yang dapat dioperasikan secara lokal di lingkungan sekolah (Kholil dan Mukmin, 2023)

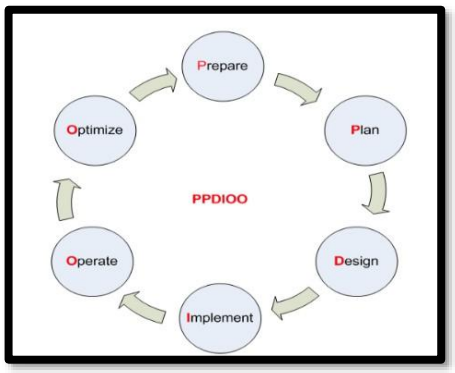
Docker merupakan teknologi containerisasi (Mubarok dan Saptono, 2019; Dwiyatno *et al*., 2020) yang memungkinkan penyebaran aplikasi secara cepat, ringan, dan konsisten di berbagai lingkungan. Dengan memanfaatkan Docker, sistem cloud storage dapat dibangun secara modular dan mudah dikelola, tanpa memerlukan konfigurasi kompleks pada sistem operasi utama. Hal ini sangat bermanfaat bagi sekolah yang memiliki keterbatasan sumber daya teknologi informasi (Dwiyatno dan Rakhmat, 2020)

Karya ini dilisensikan di bawah Creative Commons Attribution-ShareAlike 4.0 International License

Penelitian ini bertujuan untuk merancang dan membangun sistem cloud storage berbasis Docker container (Subekti, 2024). Sistem ini akan diakses oleh guru dan staf melalui jaringan internet, memungkinkan pengelolaan file yang lebih efektif dan efisien. Dengan pendekatan ini, sekolah dapat memiliki kendali penuh atas data, menghemat biaya, serta meningkatkan kolaborasi antar staf dan guru.

II. METODE PENELITIAN

Metode penelitian yang digunakan adalah Metode PPDIOO (Prepare, Plan, Design, Implement, Operate, Optimize). Berikut perancangan penelitian melalui diagram alir dengan metode PPDIOO dalam penelitian sebagai berikut.



Gambar 1. Metode Penelitian

a. Prepare

Prepare adalah tahap untuk melaksanakan rencana kerja baik dari segi teknis maupun strategis yang digunakan untuk membangun Server Cloud. Pada tahap ini penulis melakukan proses pengembangan sistem, dimana sistem akan di implementasikan. Dalam konteks penelitian ini, tahap prepare mencakup analisis kebutuhan pengelolaan penyimpanan berkas serta evaluasi teknologi yang akan digunakan. Studi pustaka dilakukan untuk memahami konsep dasar cloud storage, ownCloud, dan docker container

b. Plan

Tahap Plan melibatkan perencanaan detail dari implementasi sistem berdasarkan pemahaman yang diperoleh dari tahap Prepare. Merencanakan kebutuhan sistem baik itu secara hardware maupun Software yang akan di buat dan diharapkan dapat memberikan gambaran seutuhnya terhadap kebutuhan yang ada.

Karya ini dilisensikan di bawah Creative Commons Attribution-ShareAlike 4.0 International License

1. Kebutuhan Hardware

Dalam implementasi yang dilakukan pada penelitian ini, penulis menggunakan laptop dan VPS (Virtual Private Server). Sebelum penulis melakukan pembelian VPS pastinya ada beberapa acuan yang dijadikan dalam menentukan spesifikasi VPS seperti apa yang penulis akan gunakan. Tentu dalam menggunakan ownCloud yang paling penting adalah Storage (Penyimpanan) harus sesuai dengan yang dibutuhkan. Dikarenakan jumlah user dari seluruh elemen madrasah = 10 user. Berikut pembagian storage untuk masing-masing user untuk selama satu tahun ajaran baru :

No	Spesifikasi
1	Processor 1 Core
2	RAM 1 GB
3	HDD 40 GB

Tabel 1. Spesifikasi VPS

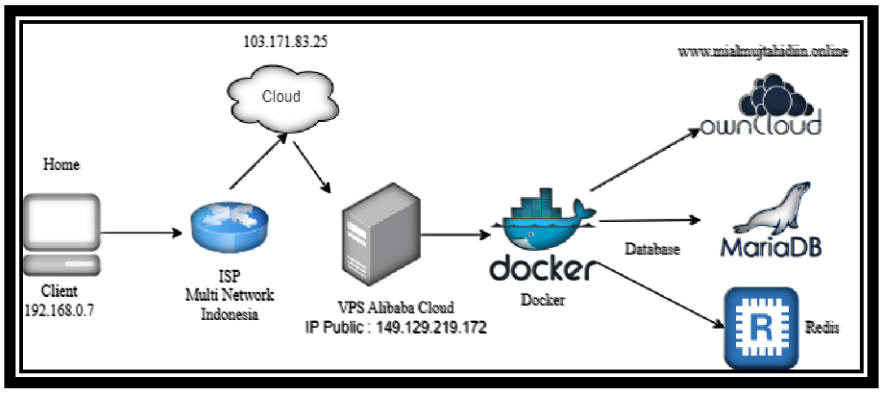
2. Kebutuhan Software

Perangkat lunak atau software berfungsi untuk mengatur dan membantu penggunanya untuk melakukan segala aktivitas yang berkaitan dengan perangkat keras, atau menjembatani antara pengguna dengan perangkat keras. Berikut software yang penulis gunakan dalam penelitian ini antara lain:

a. Windows 11 Profesional: sistem operasi yang digunakan penulis untuk melakukan konfigurasi server sekaligus menjadi client
b. Ubuntu 20.04: sistem operasi yang digunakan server pada virtual private server dan yang digunakan sebagai server slims pada penelitian ini.
c. ownCloud: Cloud storage atau penyimpanan online yang digunakan pada penelitian ini.
d. Alibaba Cloud Client: Software dari Alibaba yang digunakan penulis sebagai client untuk menghubungkan serta mengkonfigurasi server pada virtual private server.
e. Docker: Perangkat lunak yang berfungsi sebagai virtualisasi untuk menjalankan image yang akan digunakan pada server sehingga dapat diakses secara publik.
f. Wireshark: Perangkat lunak yang berfungsi untuk mengetahui hasil dari pengujian kinerja performance sistem ownCloud

Karya ini dilisensikan di bawah Creative Commons Attribution-ShareAlike 4.0 International License

Pada tahap ini membuat desain untuk melakukan pemantauan virtual server yang akan diimplementasikan pada jaringan cloud computing. Dengan dibuatnya desain ini, diharapkan akan memberikan gambaran keseluruhan dari topologi yang akan digunakan



Gambar 2. Design Topologi

c. Implement

Implement adalah proses membangun Server Cloud menggunakan ownCloud, dimulai dari proses instalasi dan konfigurasi Server Cloud hingga berhasil. Adapun tahapan yang dilakukan yaitu sebagai berikut:

1. Membuat Instance Virtual Machine
2. Akses SSH VPS Alibaba Cloud
3. Konfigurasi Domain pada Cloudflare
4. Instalasi Docker pada Ubuntu 20.04 LTS
5. Instalasi dan Konfigurasi ownCloud pada Server

d. Operate

Operate yaitu proses pengoperasian ownCloud seperti manajemen user, upload dan download dokumen, serta sharing dokumen ke user lainnya.

e. Optimize

Proses peningkatan kapasitas (storage, resource) sesuai dengan kebutuhan user agar kinerja ownCloud optimal. Pada tahap terakhir dari metode PPDIOO adalah Optimize, yang meliputi perawatan maupun perbaikan terhadap sistem yang telah dibangun. Tahap pemeliharaan dan

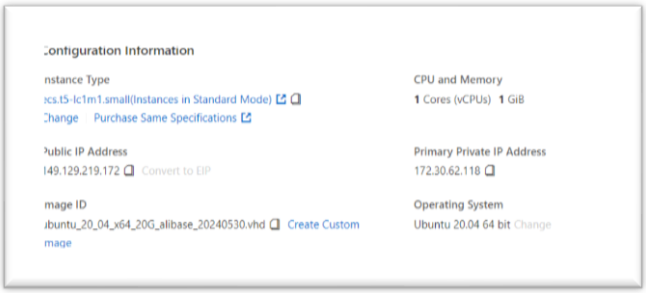
Karya ini dilisensikan di bawah Creative Commons Attribution-ShareAlike 4.0 International License

perbaikan ini terdapat beberapa proses yang harus di lakukan agar sistem dapat beradaptasi dengan perkembangan kebutuhan sistem jaringan..

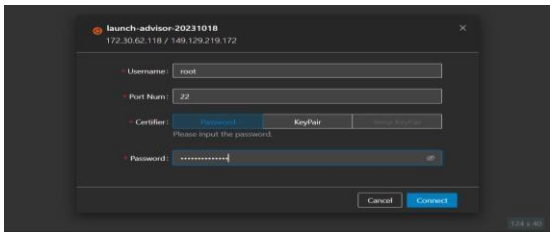
III. HASIL DAN PEMBAHASAN

Pembahasan dimulai dengan melakukan instalasi ownCloud pada VPS sesuai dengan topologi.

a. Instalasi dan Konfigurasi ownCloud server
Membuat instance VPS



Gambar 3. Create Instance



Gambar 4. Akses SSH

Melakukan Konfigurasi DNS di Cloudflare mengatur bagaimana domain namadomaian.online, subdomain www.namdomain.online mengarah ke IP VPS. A Record yaitu (Address Record) disini DNS mengarahkan nama domain ke alamat IP VPS

Instalasi Docker pada Ubuntu 20.04 LTS:

- Langkah pertama perbarui paket terlebih dahulu
  #apt-get update
- Selanjutnya, instal beberapa paket prasyarat yang memungkinkan apt menggunakan paket lewat HTTPS:
  #sudo apt install apt-transport-https ca-certificates curl software-properties-common

Karya ini dilisensikan di bawah Creative Commons Attribution-ShareAlike 4.0 International License

- Lalu tambahkan kunci GPG untuk repositori Docker resmi ke sistem Pengguna: #curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key add –
- Tambahkan repositori Docker ke sumber APT:
  #sudo add-apt-repository "deb [arch=amd64

  https://download.docker.com/linux/ubuntu focal stable"
- Akhirnya, setelah tahapan diatas sudah dilakukan. Selanjutnya instal Docker dan juga Docker Compose:
  #apt install docker
  #apt install docker compose

Instalasi dan Konfigurasi ownCloud pada Server

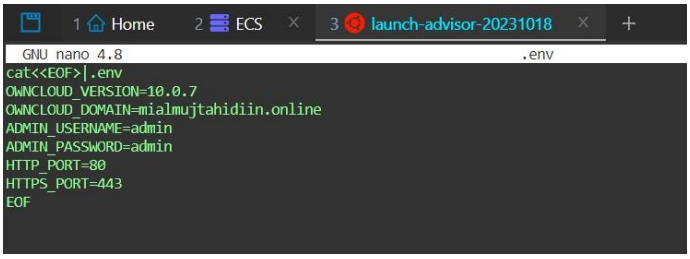
Disini penulis akan menggunakan salah satu fitur dari Docker untuk melakukan konfigurasi ownCloud yaitu menggunakan Docker Compose. Berikut Langkah-langkah konfigurasinya:

- Buat sebuah file konfigurasi untuk file docker-compose.yml #nano docker-compose.yml

Gambar 5. Docker Compose

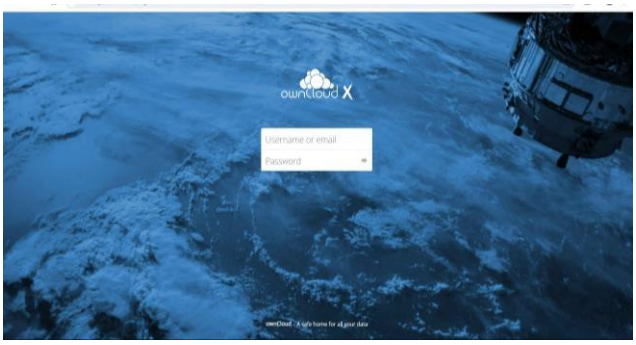
- Buat file konfigurasi .env, yang berisi pengaturan konfigurasi yang diperlukan

Karya ini dilisensikan di bawah Creative Commons Attribution-ShareAlike 4.0 International License



Gambar 6. Konfigurasi .env

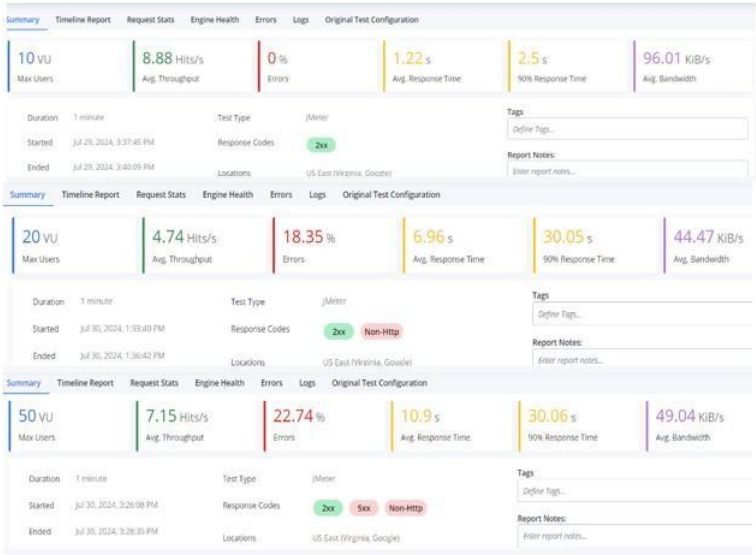
- Kemudian, kita dapat membuild dan memulai container, menggunakan alat baris perintah Docker.
  #docker-compose up -d
- Pada tahap terakhir, lakukan test pada browser dengan menggunakan domain yang sudah dikonfigurasi pada server ownCloud apakah sudah berjalan atau belum.



Gambar 7. Test OwnCloud

Dan selanjutnya adalah menguji kelayakan performance sistem ownCloud. Disini penulis menggunakan metode QoS dimana terdapat tiga parameter yaitu throughput, packet loss dan delay dalam menguji kecepatan akses sistem ownCloud dengan beberapa server lain. Selain itu juga penulis akan menguji kecepatan akses ownCloud dari sisi jumlah pengguna nya menggunakan tools online BlazeMeter.

Karya ini dilisensikan di bawah Creative Commons Attribution-ShareAlike 4.0 International License



Gambar 8. Uji Performance OwnCloud.

IV. KESIMPULAN

Perancangan dan implementasi sistem cloud storage berbasis Docker Container menggunakan ownCloud di madrasah telah berhasil dilakukan. Sistem ini sudah bisa diakses dimana saja dan kapan saja baik menggunakan Laptop/PC maupun perangkat mobile. Aplikasi ini dapat diakses melalui halaman web atau dengan mengunduh aplikasi ownCloud dari Play Store maupun App Store.

V. DAFTAR PUSTAKA

A. I. Ramdhani, "MENGGUNAKAN EPRINTS DENGAN TEKNOLOGI TUNNELING (STUDI KASUS : UNIVERSITAS BANI SALEH)," vol. 2, no. 1, 2024.

F. Ariadi, C. Iswahyudi, E. Nurnawati, J. Informatika, and I. Akprind, "PENERAPAN DOCKER CONTAINER SEBAGAI TEKNOLOGI RAMAH SKALABILITAS DIBANDING TEKNIK VIRTUALISASI UNTUK MEMBANGUN WEBSITE DI UBUNTU 18.04.4 LTS," vol. 8, no. 2, 2020.

H. Mubarok and H. Saptono, "Jurnal Informatika Terpadu BERBASIS DOCKER PLATFORM," J. Inform. Terpadu, vol. 5, no. 1, pp. 18–23, 2019, [Online]. Available: https://journal.nurulfikri.ac.id/index.php/JIT

M. Kholil and S. Mu'min, "Pengembangan Private Cloud Storage sebagai Sentralisasi Data Universitas Nahdlatul Ulama Sidoarjo Berbasis Open Source Owncloud," J. Ilmu Komput. dan Desain Komun. Vis., vol. 3, no. 1, pp. 34–42, 2018.

M. Ubuntu, S. Pada, and P. T. Xyz, "PERANCANGAN INFRASTRUKTUR DOMAIN NAME SERVER LOKAL," no. 2, pp. 23–29, 2020.

Karya ini dilisensikan di bawah Creative Commons Attribution-ShareAlike 4.0 International License

R. A. Megantara, F. Alzami, R. A. Pramunendar, and D. P. Prabowo, “Pengembangan Dan Implementasi Docker Untuk Memaksimalkan Utilitas Server Universitas Pada Masa Covid-19,” Transmisi, vol. 24, no. 2, pp. 48–54, 2022, doi: 10.14710/transmisi.24.2.48-54.

S. Dwiyatno, E. Rakhmat, and O. Gustiawan, “IMPLEMENTASI VIRTUALISASI SERVER BERBASIS DOCKER CONTAINER,” vol. 7, no. 2, 2020.

Z. M. Subekti et al., “IMPLEMENTASI LOCK MAC ADDRESS STUDI KASUS RT / RW,” vol. 1, no. 1, 2023.

Z. M. Subekti et al., “RANCANG BANGUN INFRASTRUKTUR WEB SERVER,” vol. 2, no. 1, pp. 144–151, 2024.

Z. M. Subekti, A. Fikri, A. Fadil, and H. Nugroho, “Jurnal Teknologi Sistem Informasi dan Sistem Komputer TGD Rancang Bangun Dashboard Iuran Bulanan Lingkup Rukun Tetangga (RT) Jurnal Teknologi Sistem Informasi dan Sistem Komputer TGD,” vol. 5, no. Rw 17, pp. 256–268, 2022.

Z. Zainul and N. H. Romadhan, “Cloud Storage sebagai Pengganti Arsip Manual dalam Penunjang Aktifitas Sehari-hari,” Kohesi J. Multidisiplin Saintek, vol. 1, no. 6, pp. 10–20, 2023.

Karya ini dilisensikan di bawah Creative Commons Attribution-ShareAlike 4.0 International License