

IMPLEMENTASI OPENSTACK UNTUK INFRASTRUKTUR PRIVATE CLOUD COMPUTING (STUDI KASUS UNTUK FASILITAS MAHASISWA UTDI)

Rega Panji Anugrah ¹⁾, Indra Yatini ²⁾, Muhammad Agung Nugroho ³⁾

^{1),2),3)} Prodi Informatika Universitas Teknologi Digital Indonesia
email : regaanugrah11@gmail.com¹⁾, indrayatini@utdi.ac.id²⁾, m.agung.n@utdi.ac.id³⁾

Abstraksi

Perkembangan Teknologi cloud membawa perubahan dalam berbagai model layanan. Saat ini cloud bukan hanya dapat digunakan sebagai infrastruktur (IaaS), Platform (Paas), dan software (SaaS). Namun mulai dapat digunakan hanya pada fungsi tertentu seperti cloud function as a service (FaaS). Hadirnya teknologi cloud ini semakin memudahkan institusi untuk mengembangkan layanan sendiri yang mereplikasi fungsi dari layanan cloud. Dengan ketersediaan engine opensource seperti openstack, opennebula, dan openshift, proses membangun infrastruktur mandiri untuk layanan cloud semakin terjangkau dari sisi biaya. Openstack merupakan open sources software yang dapat dikembangkan secara mandiri dengan berfokus pada infrastruktur (IaaS) karena menyediakan beragam fitur yang memadai. Penelitian ini menggunakan metode eksperimen dalam membangun layanan IaaS dengan openstack, pengujian yang dilakukan berupa pembuatan mesin virtual, monitoring mesin VM, serta uji performa mesin VM. Dari hasil penelitian, penggunaan mesin peneliti dapat memadai untuk menjalankan 10 mesin VM dengan stabil, dan rata-rata waktu untuk booting time 30 detik.

Kata Kunci:

Cloud Computing, Infrastructure as a Service, Openstack, Private cloud computing

Abstract

The development of cloud technology brings changes in various service models. Currently the cloud can not only be used as infrastructure (IaaS), Platform (Paas), and software (SaaS). But starting to be used only on certain functions such as cloud function as a service (FaaS). The presence of cloud technology makes it easier for institutions to develop their own services that replicate the functions of cloud services. With the availability of open source engines such as openstack, opennebula, and openshift, the process of building a standalone infrastructure for cloud services is increasingly affordable in terms of cost. Openstack is an open source software that can be developed independently by focusing on infrastructure (IaaS) because it provides a variety of adequate features. This study uses an experimental method in building IaaS services with openstack, the tests carried out in the form of creating virtual machines, monitoring VM machines, and testing the performance of VM machines. From the research results, the use of the research machine can be sufficient to run 10 VM machines stably, and the average boot time is 30 seconds.

Keywords:

Cloud Computing, Infrastructure as a Service, Openstack, Private cloud computing

Pendahuluan

Perkembangan Teknologi cloud membawa perubahan dalam berbagai model layanan. Saat ini cloud bukan hanya dapat digunakan sebagai infrastruktur (IaaS), Platform (Paas), dan software (SaaS). Namun mulai dapat digunakan hanya pada fungsi tertentu seperti cloud function as a service (FaaS). Hadirnya teknologi cloud ini semakin memudahkan institusi untuk mengembangkan layanan sendiri yang mereplikasi fungsi dari layanan cloud. Dengan ketersediaan engine opensource seperti openstack, opennebula, dan openshift, proses membangun infrastruktur mandiri untuk layanan cloud semakin terjangkau dari sisi biaya.

Cloud computing merupakan layanan komputasi teknologi informasi yang didalamnya terdapat layanan seperti hardware, software dan aplikasi *third party* yang dapat diperoleh melalui internet. Layanan

ini dapat disesuaikan mengikuti kebutuhan resource pengguna dan biaya yang dimiliki untuk investasi penggunaan dan resources dengan basis pembayaran secara bulanan atau sesuai penggunaan per menit/jam [1]. Beberapa organisasi memigrasikan beban kerja di lokasi internal ke cloud karena ketersediaan infrastruktur, perangkat lunak, dan layanan platform yang hemat biaya dan skalabel. Untuk memudahkan proses migrasi, banyak vendor cloud menyediakan layanan, kerangka kerja, dan alat yang dapat digunakan untuk penyebaran aplikasi di infrastruktur cloud. Menemukan layanan dan infrastruktur yang paling tepat untuk aplikasi tertentu yang menghasilkan kinerja yang diinginkan dengan biaya minimal, merupakan suatu tantangan[2]. Salah satu model deployment dalam layanan cloud adalah private cloud, dimana perusahaan atau organisasi dapat menggunakan

cloud untuk kepentingan internal perusahaan atau organisasi yang bertujuan untuk membatasi akses dari pihak luar, dan hanya bisa diakses oleh internal perusahaan agar mempermudah komunikasi sesama pegawai. Openstack [3] adalah sebuah software open source dalam cloud computing yang berorientasi di bidang Infrastructure as Service atau IaaS, openstack mengendalikan proses komputasi dan sumber daya jaringan dalam sebuah data center melalui dashboard yang memberikan control administrasi sekaligus memberikan hak akses pada pengguna melalui web interface.

Pada penelitian ini akan melakukan perancangan infrastruktur cloud untuk fasilitas layanan mahasiswa berupa virtual mesin. Penggunaan mesin utama sebagai server terbatas pada mesin yang dimiliki peneliti, dan diujicoba untuk mengetahui batas maksimum virtual mesin yang dapat berjalan stabil.

Tinjauan Pustaka

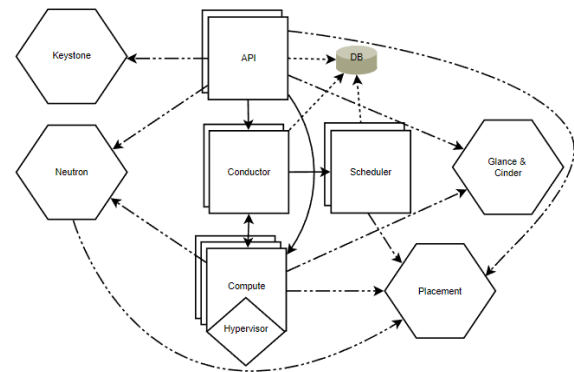
A. Cloud Computing

Cloud computing atau komputasi awan adalah model untuk memungkinkan akses ke sumber daya komputasi yang berkembang dalam teknologi informasi dan telah menjadi model bisnis yang dominan untuk memberikan infrastruktur, komponen, dan aplikasi TI. Komputasi awan memungkinkan individu dan organisasi untuk mengakses sumber daya TI sesuai permintaan, dari perangkat apa pun, dan kapan pun, sebagai layanan terukur. Ini menurunkan biaya investasi ke komputasi bekinerja tinggi, memungkinkan organisasi untuk memanfaatkan komputasi dimana mereka memiliki keterbatasan modal, atau keahlian operasional. Sementara pasar saat ini mendesak ketersediaan sumber daya TI sesuai permintaan, penyedia cloud menawarkan jumlah dan variasi layanan yang terus meningkat yang dibangun di atas kumpulan sumber daya komputasi sharing dan secara elastis dapat meningkatkan permintaan komputasi yang terus berkembang[4].

B. Openstack Nova

Nova[5] terdiri dari beberapa proses server, masing-masing melakukan fungsi yang berbeda. Antarmuka yang digunakan pengguna adalah REST API, sementara komponen Nova secara internal berkomunikasi melalui mekanisme pengiriman pesan RPC. Server API memproses permintaan REST, yang biasanya melibatkan pembacaan/penulisan basis data, secara opsional mengirim pesan RPC ke layanan Nova lainnya, dan menghasilkan respons terhadap panggilan REST. Pesan RPC dilakukan melalui library oslo.messaging, sebuah abstraksi di atas antrian pesan. Sebagian besar komponen nova utama dapat dijalankan di beberapa server, dan memiliki manajer yang mendengarkan pesan RPC. Pengecualian utama adalah nova-compute, di mana satu proses berjalan pada hypervisor yang dikelolanya (kecuali saat

menggunakan driver VMware atau Ironic). Arsitektur openstack nova dapat terlihat pada gambar 1.



Gambar 1. Arsitektur Openstack Nova[1]

Nova juga menggunakan database pusat yang (secara logis) dibagi di antara semua komponen. Namun, untuk membantu peningkatan, DB diakses melalui lapisan objek yang memastikan control plane yang ditingkatkan masih dapat berkomunikasi dengan nova-compute yang menjalankan rilis sebelumnya. Untuk memungkinkan nova-compute proxy, permintaan DB melalui RPC ke manajer pusat yang disebut nova-conductor.

C. Penelitian Sebelumnya

Penelitian ini merujuk pada beberapa penelitian sebelumnya, penelitian Analisa dan perancangan sistem cloud berbasis openstack menggunakan devstack[6] membahas mengenai openstack yang mengatur sistem penyimpanan perangkat, jaringan, hingga akses user yang dapat menjalankan proyek virtual itu sendiri. Dengan demikian, openstack dapat digunakan sebagai layanan cloud yang memfasilitasi IaaS (Infrastruktur as a Service) yang open source. Openstack juga memiliki kelebihan dimana memiliki layanan image yang dapat menyimpan image sistem operasi. Sehingga pengguna tidak perlu harus memiliki sistem operasi terlebih dahulu untuk menginstall sistem operasi di dalam openstack. Melainkan sistem operasi juga dapat dibagikan oleh user lain ke user lainnya. Layanan ini membuat sistem openstack dapat memanfaatkan sistem cloud yang lebih lengkap dan menyeluruh.

Dalam penelitian lain yang berkaitan dengan perancangan private cloud dengan menggunakan openstack dan menurut agama islam[7] memaparkan tentang penerapan private cloud pada sebuah organisasi untuk kebutuhan internal. Hasil rancangan model private cloud menggunakan OpenStack diujikan pada mesin virtual menggunakan: Processor, Intel Core i5-5200U CPU@2.40 GHzx4, Ram: 12 GB, Hardisk: 100 Gb. Hasil pengujian menunjukkan private cloud telah terbangun dengan baik.

Berdasarkan penelitian yang telah dilakukan Saleh Dwiyatno[8] dalam implementasi sistem cloud

computing pada kantor CV. Kota Baru dapat menggunakan tampilan web service. Kemudian untuk penyimpanan data pada perangkat dapat dilakukan melalui satu akses pada server owncloud yang dapat diakses dari ip local dan ip public. Penelitian penerapan teknologi cloud dengan menggunakan object storage swift [9] terbukti berhasil dalam pengujian komunikasi object storage swift. Swift merupakan salah satu fitur cloud untuk penyimpanan data di cloud dengan model object storage. Model penerapan cloud computing dapat diterapkan pada lingkungan universitas seperti penelitian yang dilakukan oleh M. Ridwan Efendi [10] yang menerapkan implementasi layanan cloud IaaS, PaaS, dan SaaS pada lingkungan Universitas Bayangkara Jakarta.

Implementasi pemanfaatan private cloud storage dapat berjalan dengan baik pada lembaga pendidikan[11], dari hasil penelitian dapat disimpulkan bahwa sistem informasi dibangun menggunakan model *incremental* yang terdiri dari tahapan *requirement*, *specification*, *architecture design*, *code* dan *test*. Model ini cocok untuk diterapkan pada pengembangan sistem website. Dalam tahapan ini dilakukan penerapan cloud storing yang memiliki 4 tahapan yaitu ketersediaan sumber daya, image server, instance yang running dan terminate. Dalam tahapan uji fungsionalitas sistem dibangun telah sesuai dengan kebutuhan pengguna; untuk uji kinerja sistem, efisiensi parameter setup time yaitu 23.33 menit lebih cepat dibandingkan sistem non cloud. Untuk ability dapat menyediakan 4 server virtual sementara sistem non cloud hanya 2 server dan access area yang lebih beragam. Namun kekurangan terdapat pada respon time dengan waktu 2.33 detik lebih lambat dari sistem yang dibangun tanpa menggunakan infrastruktur cloud.

Untuk implementasi pada layanan cloud seperti google cloud platform dengan Kubernetes dan Nextcloud dapat menjadi solusi dalam membangun sebuah cloud storage untuk collaboratin research pada lingkungan kampus [12]. Pengujian pada penelitian ini membuktikan bahwa platform nextcloud yang berjalan pada Kubernetes mendapatkan nilai yang baik dengan skala A pada pengujian Performance access speed menggunakan GTmetrix. Kemudian penggunaan pada 20 pengguna performance user request, dan dapat berjalan pada peak dengan connect time 0-5ms dan level bit/s error menunjukkan nilai rata-rata yang tidak jauh berbeda. Penurunan performa yang dapat terjadi dapat disolusikan lebih optimal dengan menggunakan model scaling pada container dengan kubernetes, dapat pula scaling ini digabungkan dengan metode scaling dari layanan cloud google dengan kombinasi penggunaan proxy, load balancing, dan scaling fitur dari kubernetes dan google cloud platform.

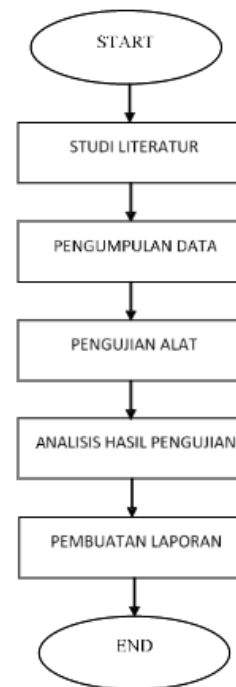
Metode Penelitian

Penelitian ini melakukan implementasi langsung dalam bentuk eksperimen pembuatan infrastruktur

IaaS berbasis private cloud. Adapun tahapan penelitian yang dilakukan adalah

- 1) Studi literatur, dengan mengumpulkan dan mengidentifikasi masalah pada penelitian sebelumnya dan mencari kemungkinan implementasi di lingkungan kampus.
- 2) Pengumpulan data, dilakukan untuk mengumpulkan data yang diperlukan dalam melakukan ujicoba, data ini diperoleh dari beberapa sumber literasi dan menggunakan metode observasi langsung pada server yang diujikan.
- 3) Pengujian, pada tahapan ini dilakukan pengujian dari hasil implementasi openstack untuk pengelolaan virtual mesin yang dibuat didalam infrastruktur openstack
- 4) Analisis pengujian, untuk mengetahui apakah semua permasalahan yang terjadi sudah dapat terselesaikan, infrastruktur yang dibuat dapat berjalan dengan baik dan menghitung kemampuan optimal dari ketersediaan perangkat dengan infrastruktur openstack
- 5) Pembuatan laporan, Penyusunan hasil dari penelitian yang telah dilakukan.

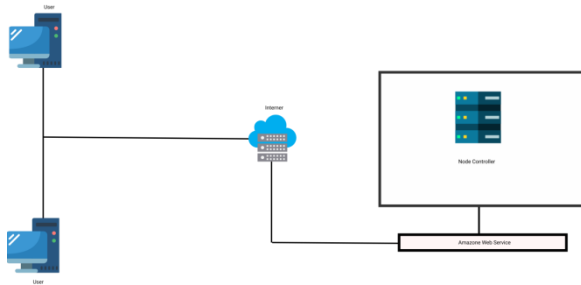
Secara keseluruhan proses tahapan penelitian ini disajikan pada gambar 2.



Gambar 2. Tahapan penelitian

Model perancangan pada penelitian ini menggunakan perangkat dengan spesifikasi 4 core CPU, 16 GB RAM, Hardisk 80 GB, dan Ubuntu versi 20.04.3 LTS, yang divisualisasikan melalui gambar 3. Dari Arsitektur tersebut, terdiri dari 1 node yang berfungsi sebagai controller node dan berjalan dalam virtual machine yang terkoneksi dengan computer host. Perangkat ini terhubung ke

internet dan pengujian hanya untuk jaringan local tidak melalui koneksi yang dapat diakses publik.



Gambar 3. Skema Alur Kerja Sistem

Pada node controller semua service-service, flavor, protocol dan penyediaan image untuk membangun infrastruktur private cloud di proses dan di jalankan. Virtual machine terhubung secara private dengan komputer host yang terhubung ke internet. Sementara user adalah orang yang akan menggunakan layanan private cloud.

Hasil dan Pembahasan

Implementasi openstack untuk infrastructure private cloud. Berdasarkan analisis dan perancangan sistem yang telah dibuat sebelumnya, maka implementasi merupakan tahap yang berisi tentang pengoperasian dan pengujian.

Proses building openstack pada mesin personal ini melalui beberapa tahapan seperti update package manager apt, instalasi dan update package manager snap, instalasi microstack yang digunakan sebagai basis openstack, dan terakhir menjalankan openstack dengan spesifikasi virtual mesin yang sesuai. Secara proses dapat tergambar dalam perintah berikut :

```
sudo apt update
sudo apt upgrade
sudo apt install snap
sudo snap refresh
sudo systemctl disable open-vm-tools.service
sudo snap install microstack --edge --devmode
microstack launch cirros
```

Setelah tahapan ini, openstack berjalan dapat terlihat pada gambar 4, dengan alamat address yang dapat diakses secara publik 192.168.222.164 dan diakses secara private pada address 10.20.20.87. Untuk spesifikasi menggunakan ml.tiny dengan VCPU 1 core, hardisk 1 GB, memori 512 MB [12].

```
ubuntu@ip-172-31-39-162:~$ microstack.openstack server list
+-----+-----+-----+-----+-----+-----+
| ID | Name | Status | Networks |
+-----+-----+-----+-----+
| 26df7405-7337-4f6d-b5ba-d040f6e6a3ca | vm01 | ACTIVE | test=192.168.222.164, 10.20.20.87 |
+-----+-----+-----+-----+
| Image | Flavor |
+-----+-----+
| cirros | ml.tiny |
+-----+-----+
```

Gambar 4. Server openstack aktif

Setelah aktif, administrator dapat melakukan akses ke dashboard dari openstack. Dalam tahapan ini,

administrator dapat melihat detil server VM yang telah berjalan di openstack, di menu Bernama daftar instances. Gambaran dashboard pada openstack tersaji pada gambar 5.

Gambar 5. Daftar instances pada dashboard openstack

Dalam proses pengujian pembuatan virtual machine, hasil dari seluruh virtual machine yang berhasil dibuat dengan melalui command line ataupun launch instance pada dashboard openstack virtual machine yang berhasil di bangun ada tiga dengan nama vm01, vm02, vm03 dan seluruh virtual machine dengan status aktif. Untuk proses pembuatan mesin VM, secara penggunaan openstack nova berjalan dengan baik, karena dalam pengujian pada gambar 6, peneliti dapat membuat beberapa mesin (dalam contoh ini adalah mesin VM03) dengan menggunakan cara melalui sistem pada openstack. Pada pengujian ini mesin dapat dibuat dengan nama VM03.

Gambar 6. Proses pembuatan mesin VM pada Openstack Nova

Selanjutnya, dilakukan pengujian untuk menggunakan image sistem operasi, dalam percobaan ini menggunakan image sistem operasi cirros yang berbasis pada centos. Untuk proses pengujian pembuatan pada sistem openstack disajikan pada gambar 7.

Gambar 7. Ujicoba penggunaan image OS pada openstack nova

Dari hasil pengujian ini, kemudian peneliti melakukan pengujian untuk melihat secara sistem openstack dimana VM yang telah dibuat dapat berjalan normal atau tidak. Proses ini dapat dilihat

pada gambar 8, dimana terlihat list VM yang telah dibuat dalam pengujian sebelumnya.

```

root@ip-172-31-39-162:~# microstack openstack server list
+-----+-----+-----+-----+-----+-----+
| ID      | Name | Status | Networks | Image | Flavor |
+-----+-----+-----+-----+-----+-----+
| cf59d8ed-9351-4340-b061-533a582b2c4a | vm01 | SHUTOFF | test=192.168.222.203, 10.20.20.42 | cirros | m1.tiny |
| c0d8d4b0-803b-4405-bf4c-efc5f81d8ee8 | vm02 | SHUTOFF | test=192.168.222.170, 10.20.20.148 | cirros | m1.tiny |
| 16d74957-7337-4f6d-b0ba-d0d0f8e8a3ca | vm03 | SHUTOFF | test=192.168.222.164, 10.20.20.87 | cirros | m1.tiny |
+-----+-----+-----+-----+-----+-----+

```

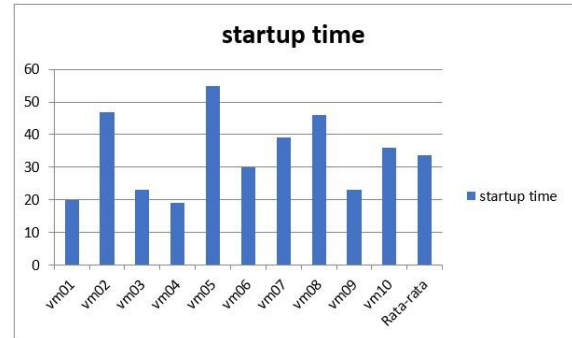
Gambar 8. Mesin VM yang berhasil berjalan aktif

Detail dari server atau virtual machine yang berjalan dan berhasil di install menggunakan command line ataupun dengan melalui dashboard openstack yang berhasil dibangun, server yang dibangun berjumlah 3 server dengan nama server 1 vm01 dengan IP Public 192.168.222.164 dan IP Private 10.20.20.87, server 2 dengan nama vm02 dengan IP Public 192.168.222.170 dan IP Private 10.20.20.148, server 3 dengan nama vm03 dengan IP Public 192.168.222.203 dan IP Private 10.20.20.42. Secara keseluruhan VM yang berjalan selama pengujian dapat terlihat pada tabel 1.

Tabel 1. Pengujian aktivitas VM pada openstack

NO	Instance Name	VCPUs	RAM (MB)	DISK (GB)	State
1	Vm01	1	512	1	Active
2	Vm02	1	512	1	Active
3	Vm03	1	512	1	Active
4	Vm04	1	512	1	Active
5	Vm05	1	512	1	Active
6	Vm06	1	512	1	Active
7	Vm07	1	512	1	Active
8	Vm08	1	512	1	Active
9	Vm09	1	512	1	Active
10	Vm10	1	512	1	Active
TOTAL		10	5120	10	

Sementara untuk pengujian startup dari masing-masing mesin VM dengan spesifikasi m1.tiny melakukan proses startup 1 - 10 mesin virtual secara bersamaan dengan waktu 19 sampai dengan 55 detik. Kemudian agar 10 mesin virtual dapat berjalan secara normal dibutuhkan waktu kurang dari 1 menit. Dalam kondisi normal sistem operasi linux centos beserta aplikasi didalamnya membutuhkan waktu kurang dari 1 menit untuk mesin virtual melakukan proses startup, namun saat lebih dari 10 mesin virtual berjalan dalam kondisi normal membutuhkan waktu lebih dari 60 detik. Dari sini, dapat disimpulkan untuk dapat menjalankan 30 mesin virtual dibutuhkan waktu yang lebih lama, namun perangkat yang digunakan peneliti hanya dapat menjalankan maksimum 10 mesin virtual. Data ini tersaji pada gambar 9.



Gambar 9. Grafik rata-rata waktu startup mesin VM

Hasil ini menunjukkan bahwa dengan spesifikasi server yang digunakan oleh peneliti maksimum mesin VM yang dapat dibuat dan berjalan stabil adalah 10 VM.

Kesimpulan dan Saran

Dalam penelitian ini dapat disimpulkan bahwa secara implementasi server openstack dapat berjalan stabil untuk menangani proses komputasi pada 10 mesin VM dengan waktu rata-rata 20-30 detik untuk proses startup dan mampu berjalan stabil berdasarkan hasil pengujian. Proses deployment pada openstack ini menggunakan single node sehingga masih dapat dimungkinkan untuk melanjutkan penelitian dengan menggunakan multi node atau multivendor cloud.

Daftar Pustaka

- [1] M. A. Nugroho and C. Subiyantoro, "ANALISIS CLUSTER CONTAINER PADA KUBERNETES DENGAN INFRASTRUKTUR GOOGLE CLOUD PLATFORM," *JIPi (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)*, vol. 3, no. 2, Dec. 2018, doi: 10.29100/JIPi.V3I2.651.
- [2] D. Chahal, M. Mishra, S. Palepu, and R. Singhal, "Performance and Cost Comparison of Cloud Services for Deep Learning Workload; Performance and Cost Comparison of Cloud Services for Deep Learning Workload," *Companion of the ACM/SPEC International Conference on Performance Engineering*, doi: 10.1145/3447545.
- [3] A. Shrivastwa, S. Sarat, K. Jackson, C. Bunch, and E. Sigler, "OpenStack: Building a Cloud Environment," 2016, Accessed: Jun. 13, 2022. [Online]. Available: https://books.google.com/books?hl=id&lr=&id=7YJcDgAAQBAJ&oi=fnd&pg=PP1&q=what+is+openstack&ots=J1eCzA_FtG&sig=bcYJnNmV7lXpoiHnVMBkCKVwDTI
- [4] A. Benlian, W. J. Kettinger, A. Sunyaev, and T. J. Winkler, "Special Section: The Transformative Value of Cloud Computing:

- A Decoupling, Platformization, and Recombination Theoretical Framework,” *Journal of Management Information Systems*, vol. 35, no. 3, pp. 719–739, Jul. 2018, doi: 10.1080/07421222.2018.1481634.
- [5] Openstack, “OpenStack Docs: Nova System Architecture,” 2022. <https://docs.openstack.org/nova/rocky/user/architecture.html> (accessed Jun. 27, 2022).
- [6] R. Cannoto, “Analisa dan Perancangan Sistem Cloud Berbasis Openstack Menggunakan Devstack,” 2019, Accessed: Jun. 13, 2022. [Online]. Available: <http://repository.uib.ac.id/id/eprint/1838>
- [7] A. Haris, “Perancangan Private Cloud Dengan Menggunakan Openstack dan Tinjauannya Menurut Agama Islam,” 2018, Accessed: Jun. 13, 2022. [Online]. Available: <http://digilib.yarsi.ac.id/4951/>
- [8] S. Dwiyatno, “PERANCANGAN PRIVATE CLOUD BERBASIS INFRASTRUCTURE AS A SERVICE,” *e-jurnal.lppmunsera.org*, vol. 8, no. 2, 2021, Accessed: Jun. 13, 2022. [Online]. Available: <https://e-jurnal.lppmunsera.org/index.php/PROSISKO/article/view/3705>
- [9] A. Adiyatma, “IMPLEMENTASI OBJECT STORAGE MENGGUNAKAN OPENSTACK SWIFT PADA SISTEM OPERASI LINUX CENTOS 7 - Akakom Repository,” Universitas Teknologi Digital Indonesia, 2016. Accessed: Jun. 13, 2022. [Online]. Available: <https://eprints.utdi.ac.id/800/>
- [10] M. Ridwan *et al.*, “Penerapan Teknologi Cloud Computing Di Universitas (Studi Kasus: Fakultas Teknologi Informasi Universitas Bayangkara Jakarta),” *scholar.archive.org*, Accessed: Jun. 13, 2022. [Online]. Available: <https://scholar.archive.org/work/tzvza6z7vjg6hjvg7tercln4q4/access/wayback/https://journal.ubm.ac.id/index.php/teknologi-informasi/article/download/356/343>
- [11] I. Santiko, “Pemanfaatan private cloud storage sebagai media penyimpanan data e-learning pada lembaga pendidikan,” *journal.uinjkt.ac.id*, vol. 10, no. 2, p. 137, 2017, doi: 10.15408/jti.v10i2.6992.
- [12] M. A. Nugroho and R. Kartadie, “CLOUD STORAGE DENGAN TEKNOLOGI KUBERNETES UNTUK PLATFORM COLLABORATIVE RESEARCH,” *JIPI (Jurnal Ilm. Penelit. dan Pembelajaran Inform.)*, vol. 6, no. 1, pp. 74–81, May 2021, doi: 10.29100/JIPI.V6I1.1908.