

Infrastructure as Code (IaC) Menggunakan OpenStack untuk Kemudahan Pengoperasian Jaringan Cloud Computing (Studi Kasus: Smart City di Provinsi Bali)

Infrastructure as Code (IaC) Using OpenStack for Ease of Operation of Cloud Computing Network (Case Study of Smart City in Bali Province)

I Putu Agus Eka Pratama

Program Studi Teknologi Informasi, Fakultas Teknik, Universitas Udayana, Jl. Raya Kampus Unud, Bukit Jimbaran, Badung, Bali, Indonesia

eka.pratama@unud.ac.id

Naskah diterima: 11 November 2020, direvisi: 14 April 2021, disetujui: 7 Juni 2021

Abstract

The implementation of smart city requires cloud computing technology, as found in the implementation of Bali Smart Island in Bali Province. Despite the convenience and benefits provided, there are several challenges faced by users related to cloud computing implementation. One of these challenges is the runtime in cloud computing, which can affect the quality of public services and the level of public trust toward the government. For this reason, this study proposes a solution for using OpenStack-based IaC using Neo-CLI to shorten operational time on cloud computing. IaC is the form of OpenStack orchestration, which is carried out on twelve nodes (stacks) in the network, including nodes for provinces, districts, and cities in Bali Province. Testing was performed by doing development until the stack is formed, then recording the time of each process. The test results indicated an average time of 9.1 seconds for the development node process to form. Based on the results, it is concluded that the proposed OpenStack-based IaC implementation using Neo-CLI can shorten the operating time of Cloud Computing. The test results also confirmed the hypothesis proposed in this study.

Keywords: *Cloud Computing, Infrastructure as Code (IaC), Neo-CLI, OpenStack, operating time*

Abstrak

Implementasi smart city memerlukan adanya teknologi cloud computing, sebagaimana dijumpai pada implementasi Bali Smart Island di Provinsi Bali. Di balik kemudahan dan manfaat yang diberikan, terdapat sejumlah tantangan yang dihadapi oleh pengguna terkait dengan implementasi cloud computing. Salah satunya adalah lamanya waktu operasional cloud computing, yang dapat memengaruhi kualitas layanan publik dan tingkat kepercayaan masyarakat terhadap pemerintah. Untuk itu, pada penelitian ini diusulkan solusi penggunaan IaC berbasis OpenStack menggunakan Neo-CLI untuk mempersingkat waktu operasional pada cloud computing. IaC dalam bentuk orkestrasi OpenStack, dilakukan pada dua belas node (stack) di dalam jaringan, yang meliputi node untuk provinsi, kabupaten, dan kota di Provinsi Bali. Pengujian dilakukan dengan melakukan development

hingga stack terbentuk, kemudian mencatat waktu masing-masing proses. Dari pengujian yang dilakukan, diperoleh hasil waktu rata-rata sebesar 9,1 detik untuk terbentuknya proses development node. Berdasarkan hasil tersebut, diperoleh kesimpulan bahwa usulan implementasi IaC berbasis OpenStack menggunakan Neo-CLI mampu mempersingkat waktu pengoperasian cloud computing. Hasil pengujian juga turut membuktikan hipotesis yang diajukan di dalam penelitian ini.

Kata kunci: Cloud Computing, Infrastructure as Code (IaC), Neo-CLI, OpenStack, waktu pengoperasian

PENDAHULUAN

Smart city merupakan konsep kota pintar yang digagas dengan tujuan untuk mengelola potensi yang terdapat di suatu kota atau daerah, maupun menyelesaikan permasalahan yang terjadi pada kota atau daerah tersebut, dengan memanfaatkan peran dari teknologi informasi (Abdurrozzaq, 2019). *Smart city* telah banyak diterapkan di berbagai kota dan daerah di Indonesia, yang dirintis melalui Gerakan Menuju 100 *Smart City* oleh Kementerian Komunikasi dan Informatika (Kominfo) bersama dengan Kementerian Dalam Negeri, Kementerian PUPR, Bappenas dan Kantor Staf Kepresidenan. Selain mencakup wilayah perkotaan, *smart city* juga mencakup bentuk hirarki daerah lainnya, yaitu dapat berupa *smart province* (mencakup wilayah provinsi), *smart regency* (mencakup wilayah kabupaten), *smart village* (mencakup wilayah desa), dan *smart island* (mencakup wilayah pulau).

Provinsi Bali menerapkan *smart city* dalam bentuk cakupan pulau (*smart island*). Implementasi *smart city* di Provinsi Bali untuk cakupan keseluruhan Pulau Bali, diberi nama *Bali Smart Island*, dengan mengusung visi *Nangu Sad Kerthi Loka Bali* (membangun alam, masyarakat, dan budaya Bali berbasis kearifan lokal *Tri Hita Karana*) dan jargon *One Island One Management* (satu pulau satu manajemen terintegrasi). Visi dan jargon ini menjadi dasar dari konsep *Bali Smart Island* serta implementasinya yang berbasiskan teknologi informasi.

Bali Smart Island merupakan satu kesatuan pemerintahan berbasis teknologi informasi yang terintegrasi, mencakup Bali daratan dan pulau-pulau kecil di sekitarnya (Nusa Penida, Nusa Ceningan, Nusa Lembongan, Pulau Menjangan). Jargon *One Island One Management*, jika diartikan secara harfiah, adalah integrasi dan satu kesatuan komando pusat. Integrasi pemerintah Provinsi Bali dengan kabupaten dan kota, mencakup tiga level. Ketiga level integrasi tersebut meliputi integrasi di level jaringan, *database*, dan aplikasi.

Guna mewujudkan integrasi, digunakanlah *cloud computing* sebagai salah satu bentuk terapan teknologi informasi. Untuk mendukung hal ini, maka Pemerintah Provinsi Bali menyediakan akses internet gratis di tempat-tempat publik, baik di perkotaan, kabupaten, kecamatan, hingga hirarki terbawah yang mencakup kelurahan, desa, dan banjar di seluruh wilayah Provinsi Bali (Nusa Bali, 2019). Hal ini bertujuan untuk menyediakan infrastruktur pendukung utama bagi jaringan komputer, internet, dan *cloud computing*.

Cloud computing merupakan teknologi yang paling banyak digunakan pada implementasi *smart city* (Pratama, 2014). National Institute of Standard and Technology (NIST) mendefinisikan *cloud computing* sebagai sebuah model dan mekanisme pada jaringan komputer, dengan komputer, perangkat, dan sumber daya komputasi yang saling terhubung satu sama lain, dapat diakses kapan pun dan di mana pun, lintas *platform*, dapat menyediakan sumber daya komputasi secara cepat, serta dapat disesuaikan dengan kebutuhan pengguna (Mell & Grance, 2011). Di

Indonesia, implementasi *smart city* menggunakan *cloud computing* pada sektor pemerintahan telah dilakukan oleh Lembaga Kebijakan Pengadaan Barang/Jasa Pemerintah (LKPP), Badan Informasi Geospasial (BIG), Balai IPTEK-net BPPT, serta Kementerian Komunikasi dan Informatika.

Cloud computing menyediakan pilihan tiga jenis layanan dan empat model *deployment* untuk pengguna. Tiga jenis layanan yang disediakan meliputi: 1) *Infrastructure as a Service (IaaS) Cloud* sebagai jenis layanan *cloud computing* yang menyediakan layanan berupa sumber daya komputasi dan infrastruktur jaringan berbasis *cloud computing*, 2) *Platform as a Service (PaaS) Cloud* sebagai jenis layanan *cloud computing* yang menyediakan layanan berupa *platform* pengembangan dan *library* pengembangan perangkat lunak komputer (*software*) berbasis *cloud computing*, 3) *Software as a Service (SaaS) Cloud* sebagai jenis layanan *cloud computing* yang menyediakan layanan perangkat lunak komputer (*software*) siap pakai bagi pengguna akhir tanpa perlu melakukan instalasi dan konfigurasi, berbasis teknologi *cloud computing*.

Adapun empat model *deployment* yang disediakan oleh *cloud computing* meliputi: 1) *Private Cloud* sebagai model *deployment cloud computing* yang dikembangkan dan dijalankan khusus pada jaringan lokal (intranet) saja, 2) *Public Cloud* sebagai model *deployment cloud computing* yang dikembangkan dan dijalankan khusus pada jaringan publik (internet), 3) *Hybrid Cloud* sebagai model *deployment cloud computing* yang dikembangkan dan dijalankan pada jaringan lokal (intranet) dan jaringan publik (internet), serta 4) *Community Cloud* sebagai model *deployment cloud computing* yang dikembangkan dan dijalankan khusus pada suborganisasi/instansi dengan kebutuhan spesifik, baik pada jaringan publik (internet) maupun jaringan lokal (intranet).

Meski demikian, terdapat enam tantangan pada implementasi jaringan *cloud computing* yang dihadapi oleh pengguna pribadi, organisasi, dan *Cloud Service Provider (CSP)*. Keenam tantangan tersebut meliputi: 1) Tidak meratanya infrastruktur, 2) Lamanya waktu pengoperasian, 3) Kurangnya sumber daya manusia (SDM) yang handal, 4) Isu keamanan data, 5) Isu privasi pengguna, 6) Adanya penyesuaian dengan proses bisnis organisasi (Teguh, 2018). Dari keenam tantangan ini, waktu pengoperasian merupakan salah satu tantangan yang berat, karena berkaitan dengan performa layanan dan kualitas layanan yang diberikan.

Pengoperasian *cloud computing* membutuhkan waktu cukup lama karena adanya penyesuaian terhadap ketiga jenis layanan dan keempat model *deployment* yang dipilih, melalui sejumlah perintah yang dijalankan di sisi *server* pada sistem operasi *Linux* (Sunaryo, Teddyana & Kasmawi, 2017). Perintah yang digunakan di dalam pengoperasian *cloud computing* cukup banyak, karena pengoperasian *cloud computing* memerlukan adanya virtualisasi, manajemen jaringan, manajemen sumber daya komputasi, keamanan data dan privasi pengguna. Selain itu, pengoperasian dan pengelolaan *cloud computing* juga memerlukan adanya pemahaman mengenai penggunaan sejumlah perintah dasar oleh administrator untuk dapat memenuhi lima karakteristik utama di dalamnya (Arsa, 2019). Kelima karakteristik pada *cloud computing* meliputi: penyediaan layanan sesuai kebutuhan pengguna (*on-demand self service*), dapat diakses secara luas (*broad network access*), layanan tersedia secara terpusat dan pembagian beban (*resource pooling*), kapasitas dapat disesuaikan dengan kebutuhan (*rapid elasticity*), serta layanan dapat diukur sesuai kebutuhan (*measured service*).

Di dalam implementasi *smart city* pada sektor pemerintahan, waktu pengoperasian *cloud computing* yang cukup lama dapat memengaruhi kualitas layanan publik yang disediakan pemerintah untuk masyarakat. Penurunan kualitas layanan publik dapat menurunkan tingkat kepercayaan masyarakat terhadap pemerintah. Hal yang sama juga berlaku untuk implementasi *cloud computing* pada *Bali Smart Island*, yang memiliki masalah pada waktu pengoperasian yang

cukup lama.

Lamanya waktu pengoperasian dalam pengimplementasian *cloud computing* pada *Bali Smart Island* dipengaruhi oleh adanya dua belas *node* (sepuluh di antaranya adalah *node* yang merepresentasikan provinsi, kabupaten, dan kota) yang perlu dikelola dengan baik. Untuk itu, diperlukan adanya sebuah solusi untuk melakukan otomatisasi pada pengoperasian *cloud computing* serta minimalisasi jumlah perintah yang digunakan untuk mempersingkat waktu pengoperasian pada *cloud computing*.

Berangkat dari permasalahan tersebut, penelitian ini mengusulkan solusi berupa *Infrastructure as Code* (IaC) berbasis *OpenStack* ke dalam jaringan *cloud computing*, dengan mengadopsi teknik *DevOps* dalam bentuk orkestrasi. *DevOps* (*Development and Operational*) merupakan sebuah model pengembangan *software* dan operasional sistem yang menggabungkan tahap pengembangan *software* di dalam jaringan komputer dengan operasional jaringan komputer dengan menggunakan perangkat lunak berlisensi *open source* sesuai dengan fungsinya masing-masing (Liming, 2015). Di dalam implementasinya, terdapat sejumlah teknik yang digunakan pada *DevOps*, salah satunya IaC.

Infrastructure as Code (IaC) adalah salah satu teknik pada *DevOps* yang mendefinisikan infrastruktur jaringan komputer ke dalam bentuk otomatisasi berbasis sumber kode, dengan tujuan untuk menyediakan lingkungan pengembangan *software*, konfigurasi, manajemen, serta infrastruktur jaringan, dengan memperhatikan aspek-aspek keamanan (Siebra, 2018). IaC menekankan kepada pengelolaan dan penyediaan infrastruktur berbasis *file* yang dapat dibaca oleh komputer di dalam penanganan jaringan dan otomatisasi, dengan meminimalkan konfigurasi fisik pada komputer dan jaringan (Nguyen, 2017).

IaC pada dasarnya bukanlah bagian dari *cloud computing*, baik pada ketiga jenis layanan maupun keempat model *deployment*. Jika ketiga jenis layanan pada *cloud computing* dan keempat model *deployment* pada *cloud computing* menyediakan *server*, *storage*, *networking*, dan proses virtualisasi, maka IaC berfokus pada transformasi proses-proses dari jenis layanan dan model *deployment cloud computing* ke dalam sebuah *file* dengan kode terstruktur di dalamnya, yang menganut konsep *bare-metal DevOps* (Shivam, 2020). *Bare-metal* merupakan *instances cloud computing* yang memudahkan pengguna menjalankan satu atau lebih baris perintah melalui *file* dengan kode terstruktur di dalamnya, agar dapat mewujudkan *infrastructure service* melalui pendekatan *DevOps*. IaC muncul seiring dengan makin pesatnya *DevOps* di dalam pengembangan dan implementasi *software*.

Untuk dapat mendukung jalannya IaC pada *DevOps*, digunakanlah *container*. *Container* merupakan *software opensource* yang digunakan untuk isolasi virtual. Di dalamnya tersedia sejumlah pustaka dan *tool* pengujian *software* untuk menjalankan *software* dan *kernel* secara simultan pada sebuah sistem operasi, tanpa perlu menggunakan *virtual machine* (Docker, 2020). *Software container* yang paling banyak digunakan adalah *Docker*.

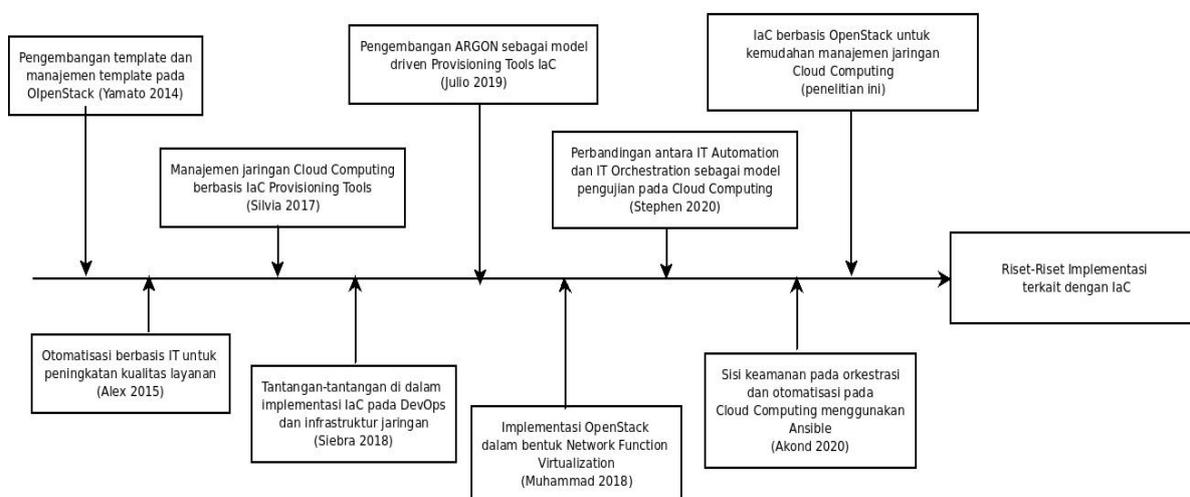
Selain *Docker*, *container* yang umum digunakan di dalam *DevOps* dan IaC untuk orkestrasi adalah *Kubernetes*. *Kubernetes* (*k8s*) merupakan *software open source* untuk orkestrasi *container*, otomatisasi, *deployment*, dan *scaling*, yang bekerja dengan cara mengelompokkan sejumlah *container* untuk membentuk unit logis aplikasi (Kubernetes, 2020). *Openstack* merupakan *software open source* berbasis *IaaS Cloud*, yang menyediakan kontrol sumber daya komputasi, orkestrasi, otomatisasi, serta pengelolaan jaringan (OpenStack, 2020). Orkestrasi menggunakan *OpenStack* (Red Hat, 2020), memberikan efektivitas sumber daya komputasi pada jaringan *cloud computing* dari sisi analisis struktural dan fungsional, berdasarkan kebutuhan sumber daya komputasi berskala tinggi (Daniel, 2015).

Pertanyaan penelitian yang diajukan di dalam penelitian ini adalah: 1) Bagaimana teknis

manajemen *node* di dalam jaringan *cloud computing* melalui orkestrasi *OpenStack* pada *IaC* yang diimplementasikan untuk meminimalisasi waktu pengoperasian? 2) Bagaimana cara mengukur waktu operasional pada pengoperasian jaringan *cloud computing* untuk setiap *node (stack)*? Hipotesis yang diajukan di dalam penelitian ini adalah implementasi *IaC* berbasis *OpenStack* ke dalam jaringan *cloud computing* pada implementasi *smart city* di Provinsi Bali dalam bentuk *Bali Smart Island*, dapat mempermudah dan mempersingkat waktu pengoperasian pada setiap *node (stack)* yang terbentuk. Tujuan dari penelitian ini adalah untuk menyelesaikan permasalahan waktu pengoperasian *cloud computing* pada implementasi *Bali Smart Island* dan sekaligus membuktikan hipotesis yang diajukan.

Terdapat delapan buah penelitian sebelumnya yang menjadi *state of the art* dari penelitian ini. Penelitian pertama mengkaji pengembangan *template* dan manajemen *template* pada *OpenStack* untuk memudahkan proses manajemen sumber daya komputasi virtual di dalam *cloud computing* (Yamato, 2014). Penelitian kedua menganalisis sejumlah tantangan yang dihadapi oleh pengguna, individu dan organisasi, di dalam implementasi *IaC* pada *DevOps* dan infrastruktur (Siebra, 2018). Penelitian ketiga membahas tentang bentuk dan contoh lain otomatisasi berbasis teknologi informasi untuk meningkatkan kualitas layanan (Alex, 2015). Penelitian keempat menguraikan bentuk lain dari implementasi *OpenStack* berupa *Network Function Virtualization* berbasis *cloud computing* (Muhammad, 2018). Penelitian kelima melakukan perbandingan antara *IT Automation* dan *IT orchestration* yang menjadi acuan di dalam pemilihan model untuk pengujian kemudahan dan kecepatan pengoperasian pada *cloud computing* (Stephen, 2020). Penelitian keenam menelaah sisi keamanan dari proses orkestrasi dan otomatisasi pada *cloud computing* menggunakan *Ansible*, terlepas dari berbagai kemudahan yang diberikannya (Akond, 2020). Penelitian ketujuh membahas mengenai pengembangan ARGON sebagai sebuah model *driven Provisioning Tools IaC* yang dikembangkan dengan model *Domain Specific Language (DSL)* untuk kebutuhan infrastruktur pada *cloud computing*, dengan fokus pengujian pada layanan *cloud computing* dari *Microsoft Azure* dan *Amazon Web Service* (Julio, 2019). Penelitian kedelapan menguraikan upaya manajemen jaringan *cloud computing* secara lebih baik berbasiskan *IaC* dengan tipe *Provisioning Tools IaC* menggunakan *provisioning script* pada layanan *Amazon Web Service* (Silvia, 2017).

Dari delapan penelitian yang telah dipelajari dengan seksama tersebut, diperoleh sebuah *gap* penelitian. *Research gap* dimaksud adalah belum adanya penelitian dan pembahasan



Gambar 1. Roadmap Penelitian IaC Berbasis OpenStack pada Cloud Computing

mengenai *Infrastructure as Code (IaC)* dengan tipe *Provisioning Tools* pada *cloud computing*, yang menggunakan orkestrasi berbasis *OpenStack*, untuk mempermudah dan mempersingkat waktu pengoperasian *cloud computing*. Roadmap penelitian disajikan pada Gambar 1.

METODE PENELITIAN

Di dalam penelitian ini, digunakan sejumlah perangkat lunak komputer (*software*) dan perangkat keras komputer (*hardware*) pendukung. *Software* di sisi *client* untuk pengembangan menggunakan sistem operasi Linux Ubuntu 20.04 LTS 64 bit, *Python3*, *python3-ven*, *python3-distutils*, *python3-testresources*, *pip*, dan *NEO-CLI*. Adapun *software* di sisi *server* dan jaringan *cloud computing* menggunakan *OpenStack*, *Kubernetes*, dan sistem operasi Linux Ubuntu 20.04 LTS 64 bit.

Hardware di sisi pengembang menggunakan *notebook* Toshiba (*processor Intel core i3*, memori RAM 4 GB, penyimpanan internal 500GB). Adapun *hardware* di sisi *server* menggunakan *processor Intel Quad core*, memori RAM 16GB, serta penyimpanan internal 1 TB SSD *Hard Drive*. Selain itu, digunakan juga *hardware* pendukung untuk kebutuhan koneksi internet selama penelitian dilakukan, berupa sebuah modem *internet* dan *router* jaringan.

Penelitian dilakukan pada masa pandemi Covid-19 secara *remote* dari lokasi rumah peneliti ke dalam jaringan *cloud computing*. Waktu penelitian berlangsung dari bulan Mei 2020 hingga Juni 2020. Metode penelitian yang digunakan adalah metode penelitian kualitatif dengan pendekatan studi kasus. Studi kasus yang diangkat adalah permasalahan waktu di dalam pengoperasian *cloud computing* pada implementasi *Bali Smart Island*, yang melibatkan manajemen terhadap dua belas *node* di dalamnya.

Adapun kedua belas *node* tersebut meliputi satu buah *node* provinsi, delapan buah *node* kabupaten, satu buah *node* kotamadya, satu buah *node virtual machine*, dan satu buah *node key-pairs*. Kedua belas *node* ini saling terhubung satu sama lain. Jaringan *cloud computing* dengan kedua belas *node* di dalamnya merupakan *environment prototype* yang ditujukan untuk penelitian dan pengujian.

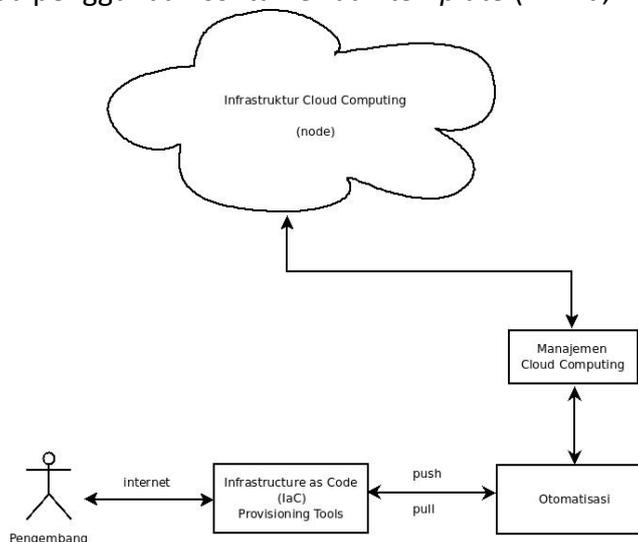
Metode yang digunakan di dalam penelitian ini adalah DSRM atau *Design Science Research Methodology* (Peppers, Tuunanen, Rothenberger, & Chatterjee, 2007). DSRM meliputi delapan buah langkah berikut: 1) Studi literatur, 2) Motivasi penelitian, 3) Identifikasi masalah, 4) Usulan solusi dan desain usulan solusi, 5) Pembuatan prototipe atau implementasi solusi, 6) Demo atau pengujian terhadap prototipe atau implementasi, 7) Dokumentasi hasil penelitian dan pengujian, 8) Publikasi hasil penelitian (jurnal/proceeding/buku).

Identifikasi masalah dilakukan melalui rumusan masalah dalam bentuk pertanyaan penelitian serta hipotesis penelitian. Dua buah pertanyaan penelitian yang diajukan di dalam penelitian ini adalah: 1) Bagaimana teknis dari manajemen *node* di dalam jaringan *cloud computing* melalui orkestrasi *OpenStack* pada IaC yang diimplementasikan untuk meminimalisasi waktu pengoperasian? 2) Bagaimana cara mengukur waktu operasional pada pengoperasian jaringan *cloud computing* untuk setiap *node (stack)*?

Hipotesis yang diajukan di dalam penelitian ini adalah bahwa implementasi IaC berbasis *OpenStack* ke dalam jaringan *cloud computing* pada implementasi *smart city* di Provinsi Bali dalam bentuk *Bali Smart Island* dapat mempermudah dan mempersingkat waktu pengoperasian pada setiap *node (stack)* yang terbentuk.

Usulan solusi yang diajukan di dalam penelitian ini adalah penggunaan *Infrastructure as Code (IaC)* berbasis *OpenStack* dalam bentuk orkestrasi yang mengadopsi sistem di dalam *DevOps*. IaC terdiri dari empat pilihan tipe, yaitu: 1) *Scripting* IaC yang diimplementasikan secara langsung

menggunakan sumber kode perangkat lunak (*script*), 2) *Configuration Management Tool* IaC yang menggunakan *file* konfigurasi dalam bentuk minimal satu baris kode, 3) *Provisioning Tools* IaC yang berfokus pada pembuatan infrastruktur berbasis kode, dan 4) *Container and Templating Tools* IaC yang memfokuskan pada penggunaan *container* dan *template* (Emilia, 2020).



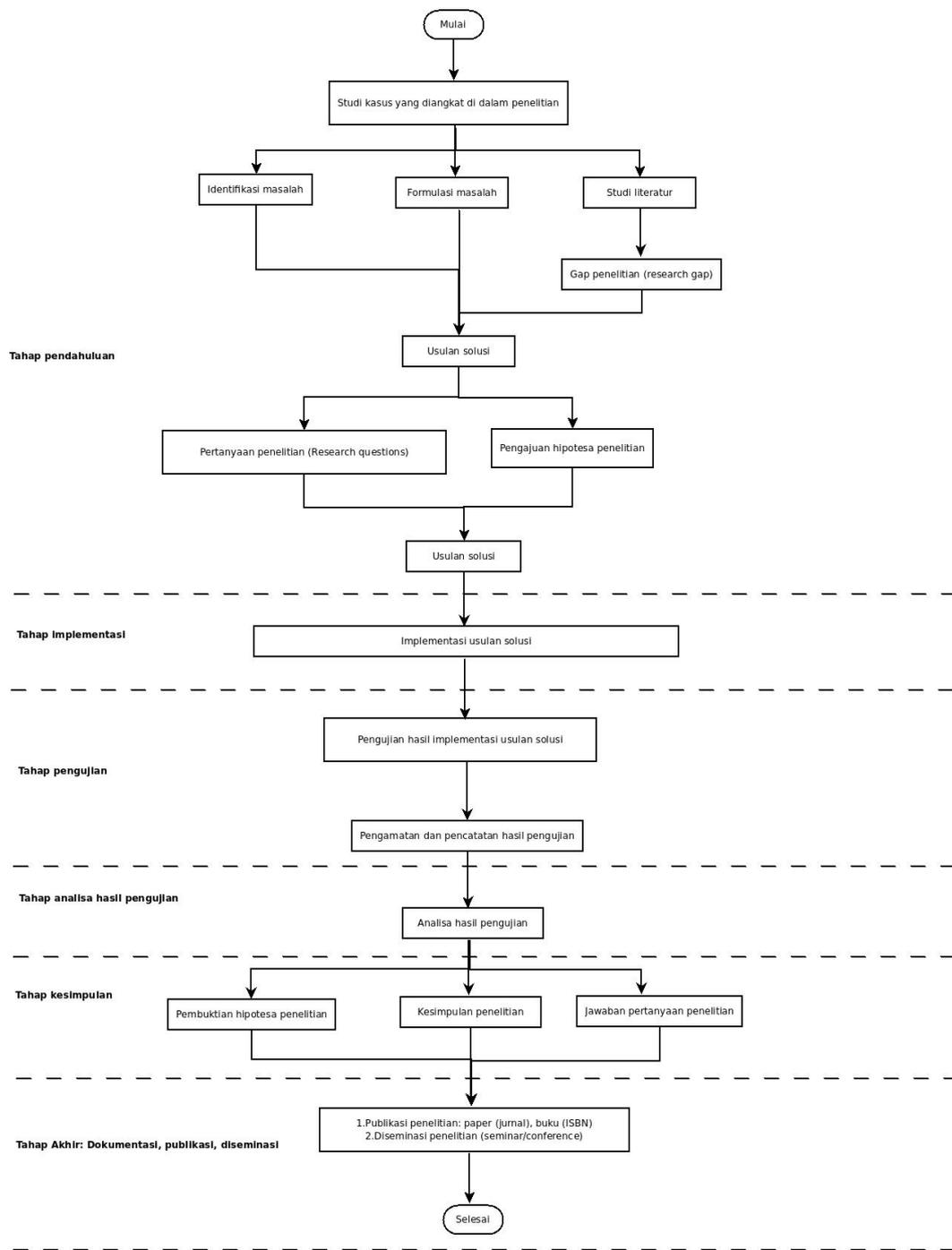
Gambar 2. Bagan Usulan Solusi

Pada usulan penelitian ini, tipe IaC yang diusulkan adalah *Provisioning Tools* IaC. Fokus utama dari *Provisioning Tools* IaC adalah pada *deployment node* untuk infrastruktur pada jaringan *cloud computing* berbasis kode, sehingga mampu mempercepat dan menghemat waktu untuk manajemen dan *deployment* melalui peringkasan perintah yang tersimpan pada *file* konfigurasi untuk otomatisasi. Gambar 2 menunjukkan bagan dari usulan solusi yang diajukan.

Gambar 2 memperlihatkan usulan solusi berupa IaC berbasis *OpenStack* menggunakan orkestrasi yang mengadopsi sistem pada *DevOps* untuk mempersingkat waktu manajemen dan pengoperasian *node (stack)* pada *cloud computing*. Pengembang (dalam hal ini peneliti) melakukan koneksi ke IaC *Provisioning Tools* melalui internet, untuk dapat melakukan otomatisasi (*pull* dan *push*) ke dalam sebuah *file* konfigurasi berbasis kode di dalamnya. Otomatisasi ini akan membantu manajemen *node-node cloud computing* pada proses *deployment* dari semua *node* yang dibentuk (inisialisasi hingga akhir). Antara pengguna dan sistem, diperantarai oleh sebuah *tool* yang menjadi media untuk penyediaan *file* konfigurasi otomatisasi.

Di dalam penelitian ini, terdapat enam tahapan penelitian. Tahapan pada penelitian ini meliputi: 1) Tahap pendahuluan, 2) Tahap implementasi, 3) Tahap pengujian, 4) Tahap analisis hasil pengujian, 5) Tahap kesimpulan, dan 6) Tahap akhir (dokumentasi, publikasi, diseminasi). Tahapan yang dilakukan pada penelitian ini beserta langkah-langkahnya, digambarkan ke dalam sebuah bagan alir (*flowchart*) penelitian. *Flowchart* penelitian disajikan pada Gambar 3.

Prototipe sistem menggunakan Neo-CLI pada sistem operasi Linux Ubuntu 20.04 64bit, yang terhubung ke dalam jaringan *cloud computing* secara *remote*. Demo, dokumentasi, dan publikasi dalam bentuk pengujian dan pembahasan, pencatatan hasil pengujian dan analisis hasil pengujian, serta penulisan *paper* publikasi. Dilakukan juga pencatatan data waktu pengoperasian jaringan *cloud computing* pada proses *deployment* setiap *node (stack)*, untuk kemudian dihitung waktu rata-ratanya. Terakhir, dilakukan pembahasan hasil pengujian serta penarikan kesimpulan penelitian, yang menjawab pertanyaan penelitian dan membuktikan hipotesis penelitian.



Gambar 3. Flowchart Penelitian

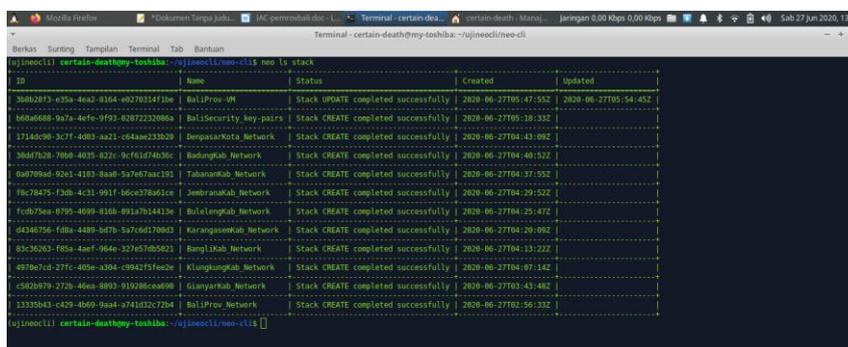
HASIL DAN PEMBAHASAN

Implementasi dilakukan dengan terlebih dahulu membuat koneksi *remote* ke jaringan *cloud computing* menggunakan Neo-CLI, dengan *data center* terletak pada *region* Jakarta (jkt) di URL <https://keystone.jkt-1.neo.id:443/v3>. Setelah terhubung ke jaringan *cloud computing* pada *region* yang telah dipilih tersebut, langkah berikutnya adalah melakukan proses *deployment* terhadap dua belas buah *node (stack)* di dalam jaringan. Kedua belas *node* tersebut terdiri atas sebuah *node* provinsi, delapan *node* kabupaten, satu *node* kota, dan dua *node* penyerta.

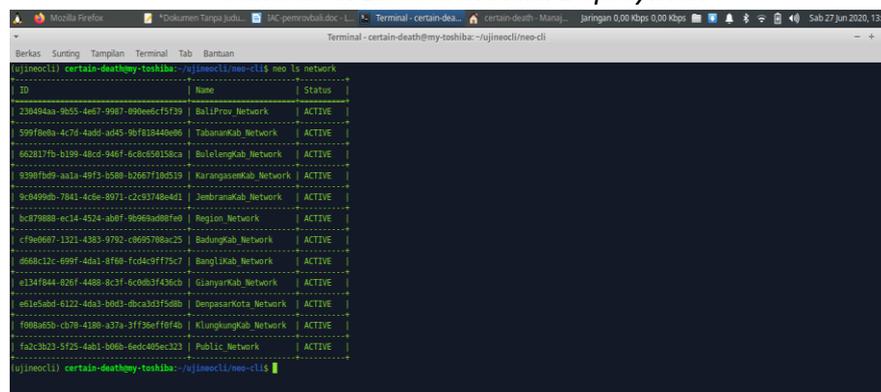
Tabel 1. Node Deployment Bali Smart Island

No	Nama Node	Status
1	BaliProv_Network	node (stack)
2	GianyarKab_Network	node (stack)
3	KlungkungKab_Network	node (stack)
4	BangliKab_Network	node (stack)
5	KarangasemKab_Network	node (stack)
6	BulelengKab_Network	node (stack)
7	JembranaKab_Network	node (stack)
8	TabananKab_Network	node (stack)
9	BadungKab_Network	node (stack)
10	DenpasarKota_Network	node (stack)
11	BaliSecurity_Key-pairs	virtual firewall
12	BaliProv-VM	virtual machine

Proses *deployment* pada setiap *node* di dalam jaringan *cloud computing* dilakukan dengan menggunakan sebaris perintah *neo-create*. Perintah *neo-create* ini memanfaatkan aplikasi *open source* bernama Neo-CLI yang akan menjalankan fungsionalitas orkestrasi pada jaringan *cloud computing* berbasis *engine opensource OpenStack* dan *Kubernetes* (BiznetGIO, 2020). Proses *deployment node* dilakukan satu per satu, dimulai dari *node BaliProv_Network*, dilanjutkan dengan *node GianyarKab_Network*, *node KlungkungKab_Network*, *node BangliKab_Network*, *node KarangasemKab_Network*, *node BulelengKab_Network*, *node JembranaKab_Network*, *node TabananKab_Network*, *node BadungKab_network*, *node DenpasarKota_Network*, dan *node BaliSecurity_Key-pairs*. Adapun *list* dari semua *node* dan *list* semua *network* yang telah berhasil dilakukan *deployment*, ditampilkan pada Gambar 4. *List* dari *network* yang terbentuk pada setiap *node (stack)* yang telah dilakukan *deployment*, ditampilkan pada Gambar 5 beserta status aktif masing-masing.



Gambar 4. List Stack Hasil Deployment



Gambar 5. List Network beserta Status

Pengujian beserta pengamatan hasil pengujian, dilakukan dengan memperhatikan lama waktu dari proses *deployment* terhadap kesepuluh buah *node (stack)*, yang masing-masing mewakili wilayah provinsi, kabupaten, dan kota. Waktu yang dicatat tersebut terbagi atas waktu untuk proses inialisasi dan waktu akhir.

Dengan menggunakan sebaris perintah *neo-create*, proses *deployment* setiap *node* di dalam jaringan *cloud computing* dapat dilakukan dengan lebih mudah dan cepat. Perintah *neo-create* pada dasarnya akan memanggil aplikasi Neo-CLI untuk melakukan proses pembuatan (*create*) dan *deployment node* yang dimaksudkan, dengan menggunakan empat parameter pada jaringan *cloud computing*. Keempat parameter ini digunakan pada semua *node* yang akan di-*deploy* di dalam jaringan. Adapun keempat parameter tersebut meliputi: 1) Pengalamatan ringkas untuk perutean antar-*node* menggunakan *Classless Inter Domain Routing (CIDR)* di 192.168.0.0/24, 2) Pengalamatan *Gateway* di 192.168.0.254, 3) *Package* dalam bentuk infrastruktur virtual untuk penyedia layanan di dalam jaringan *cloud computing (Master)* dan pengguna (*Worker*), 4) *Template private* untuk semua *node* menggunakan *Kubernetes*. Dalam hal ini, *Kubernetes* berfungsi untuk membantu pengelolaan *cluster*.

Tahapan proses pemanggilan perintah *neo-create* dimulai dengan pembuatan jaringan untuk *node (network)*, dilanjutkan dengan pembuatan *cluster* untuk *network* yang telah terbentuk, lalu diakhiri dengan pembuatan mesin virtual (*instance*) di dalam *node* pada jaringan *cloud computing*. Sesuai dengan konsep IaC, konfigurasi dan parameter disimpan secara terstruktur ke dalam sebuah *file* bernama *neo.yml*. *File neo.yml* berada di jaringan lokal tempat pengembangan dilakukan, yaitu pada *virtual environment (venv)* di */home*.

Adapun *node* yang dicatat di dalam pengujian adalah *node* yang mewakili provinsi, kabupaten, dan kota di Provinsi Bali, dengan total sebanyak sepuluh buah *node*. Catatan waktu yang diperlukan dari masing-masing *node* untuk proses inialisasi hingga *node (stack)* terbentuk, ditampilkan pada Tabel 2.

Tabel 2. Waktu dari Inialisasi hingga Stack Terbentuk

No	Stack	Awal Inialisasi	Akhir
1	BaliProv_Network	2020-06-27 10:56:25	2020-06-27 10:56:29
2	GianyarKab_Network	2020-06-27 11:43:38	2020-06-27 11:43:44
3	KlungkungKab_Network	2020-06-27 12:06:55	2020-06-27 12:07:02
4	BangliKab_Network	2020-06-27 12:12:59	2020-06-27 12:13:12
5	KarangasemKab_Network	2020-06-27 12:19:55	2020-06-27 12:20:17
6	BulelengKab_Network	2020-06-27 12:25:32	2020-06-27 12:25:38
7	TabananKab_Network	2020-06-27 12:37:46	2020-06-27 12:38:03
8	JembranaKab_Network	2020-06-27 12:29:40	2020-06-27 12:29:45
9	BadungKab_Network	2020-06-27 12:40:40	2020-06-27 12:40:46
10	DenpasarKota_Network	2020-06-27 12:43:01	2020-06-27 12:43:06

Dari Tabel 2 tampak bahwa dari masing-masing *node* pada kesepuluh *node* (provinsi, kabupaten, kota) yang terbentuk, setiap *node* memiliki catatan waktu berbeda-beda dari proses inialisasi hingga pasca-*node* terbentuk. Berdasarkan data yang dicatat pada hasil pengujian, *node (stack) BaliProv_Network* memiliki waktu terpendek sebesar 4 detik, sedangkan *KarangasemKab_Network* memiliki waktu terlama sebesar 23 detik. Dari data tersebut, kemudian dilakukan perhitungan waktu rata-rata untuk setiap *node stack* melalui formulasi berikut:

$$\begin{aligned} \text{Waktu rata-rata} &= (\text{selisih awal dan akhir stack/stack}) \text{ detik} \\ &= (91/10) \text{ detik} \\ &= 9,1 \text{ detik.} \end{aligned}$$

Dari perhitungan di atas diperoleh hasil bahwa waktu rata-rata pengoperasian menggunakan *Infrastructure as Code* (IaC) untuk setiap *node* (*stack*) di dalam jaringan *cloud computing* adalah sebesar 9,1 detik. Waktu rata-rata ini terhitung singkat. Demikian pula, waktu yang dibutuhkan setiap *node* lainnya untuk proses inialisasi dan waktu akhir juga terbilang cepat. Hal ini disebabkan oleh penggunaan perintah *neo-create*, yang di dalamnya mewakili sejumlah proses yang diringkas ke dalam sebuah *file* konfigurasi berbasis sumber kode dengan nama *file neo.yml*.

Jika dikaitkan kembali dengan peran dan kontribusinya terhadap implementasi *smart city* berbasis teknologi *cloud computing*, khususnya dalam hal ini *Bali Smart Island* di Provinsi Bali, maka waktu rata-rata sebesar 9,1 detik secara langsung akan meningkatkan kualitas layanan publik berbasis *cloud computing* yang disediakan oleh pemerintah kepada masyarakat. Singkatnya waktu pemrosesan pada *cloud computing* telah mempercepat waktu penyediaan layanan publik dari aplikasi *smart city* berbasis *cloud computing*, tanpa terkendala waktu pengoperasian teknis di sisi *server* dan teknis *cloud computing*.

Selain itu, administrator atau pengelola *cloud computing* akan terbantu di dalam proses manajemen *node* pada jaringan *cloud computing* melalui perintah yang singkat. Jadi, pengelola dapat menghemat waktu dan tenaga di dalam menangani berbagai layanan publik berbasis *cloud computing* yang disediakan oleh pemerintah untuk masyarakat dalam *Bali Smart Island*. Penghematan waktu dan tenaga dalam mengelola *node* dan mengoperasikan *cloud computing* melalui otomatisasi berbasis IaC yang diusulkan dan diujikan di dalam penelitian ini, juga akan mampu meningkatkan kualitas layanan publik pada *smart city* sekaligus meningkatkan kepuasan dan kepercayaan masyarakat terhadap kinerja pemerintah. Dengan demikian, usulan solusi yang disampaikan dalam bentuk penggunaan IaC di dalam penelitian ini, dinilai sudah tepat.

KESIMPULAN

Dari analisis hasil pengujian diperoleh kesimpulan bahwa implementasi IaC berbasis *OpenStack* untuk orkestrasi pada jaringan *cloud computing* dapat menyelesaikan permasalahan lamanya waktu pemrosesan dan operasional pada jaringan *cloud computing*. Waktu rata-rata yang diperoleh dari proses *deployment* hingga *node* (*stack*) terbentuk menggunakan Neo-CLI adalah sebesar 9,1 detik.

Waktu sebesar 9,1 detik ini terhitung cukup cepat untuk penanganan dan operasional *node-node* di jaringan *cloud computing*, mulai dari proses inialisasi hingga terbentuknya *node*. Hal ini disebabkan oleh adanya perintah *neo-cli* yang meringkas sejumlah tahapan di dalam *deployment* (pembuatan *network*, *cluster*, dan *instance*) ke dalam satu perintah, melalui konfigurasi yang tersimpan pada *file neo.yml* di *virtual-environment* (*venv*) /*home*.

Hasil pengujian ini sekaligus membuktikan hipotesis penelitian, yaitu bahwa implementasi IaC berbasis *OpenStack* ke dalam jaringan *cloud computing* pada implementasi *smart city* di Provinsi Bali dalam bentuk *Bali Smart Island*, dapat mempermudah dan mempersingkat waktu pengoperasian setiap *node* (*stack*) yang terbentuk. Hasil pengujian ini juga menjawab dua pertanyaan penelitian yang diajukan. Yang pertama adalah mengenai teknis dari manajemen *node* di dalam jaringan *cloud computing* melalui orkestrasi *OpenStack* pada IaC yang diimplementasikan untuk meminimalisasi waktu pengoperasian. Adapun yang kedua adalah mengenai teknis pengukuran waktu operasional pada pengoperasian jaringan *cloud computing* untuk setiap *node* (*stack*) melalui implementasi dan pengujian.

Dengan memperhatikan hasil pengujian yang diperoleh di dalam penelitian ini,

implementasi *Infrastructure as Code* (IaC) berbasis *OpenStack* dengan memanfaatkan aplikasi *open source* Neo-CLI dinilai merupakan usulan solusi yang tepat. Hal tersebut juga dapat menjadi bahan masukan untuk kemudahan dan kecepatan pengelolaan, operasional, dan manajemen *node* pada jaringan *cloud computing*, agar dapat menunjang berbagai layanan publik yang disediakan oleh pemerintah dalam implementasi *smart city* berbasis teknologi *cloud computing*, dengan studi kasus *Bali Smart Island* di Provinsi Bali.

Ke depannya, penelitian ini dapat dikembangkan lebih lanjut dengan menggunakan tiga tipe IaC lainnya (*Scripting IaC*, *Configuration Management Tools IaC*, *Container and Templating Tools IaC*) pada studi kasus implementasi *smart city* melalui penyediaan layanan publik berbasis *cloud computing*. Jadi, dari hasil penelitian tersebut dapat diperoleh perbandingan waktu yang diperlukan selama proses manajemen dan operasional *node-node* di dalamnya pada *deployment* dari waktu inisialisasi hingga waktu akhir.

UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih yang sebesar-besarnya kepada semua pihak yang turut membantu dan mendukung jalannya penelitian ini secara *online* selama pandemi Covid-19 2020, antara lain: *BiznetGIO*, Pemerintah Provinsi Bali, Universitas Udayana, komunitas Linux Indonesia, tim *developer* Linux dan *open source*, serta keluarga penulis.

DAFTAR PUSTAKA

- Abdurrozzaq, H., Oris K.S., 2019. "Smart City, Konsep Kota Cerdas Sebagai Alternatif Penyelesaian Masalah Perkotaan Kabupaten/Kota, di Kota-Kota Besar Provinsi Sumatera Utara". *Jurnal Buletin Utama Teknik*, Vol.14,No.2, pp.128-135.
- Ahmed E.Y., 2012. *Exploring Cloud Computing Services and Applications*. *Journal of Emerging Trends in Computing and Information Sciences*, Vol. 3, No. 6, pp. 838-847.
- Akond, R., 2020. "Security Smells in Ansible and Chef Scripts: A Replication Study". *Journal of ACM Transactions On Software Engineering and Methodology (TOSEM)*, Vol.01, pp.31-40.
- Alex, B.D., Muhammad K., 2015. "Sistem Otomasi Sebagai Upaya Perbaikan Kualitas Dengan Metode Pada Line Finishing (Studi Kasus : PT X)". *Jurnal Ilmiah Teknik Industri*, Vol.3 No.3, pp.141 – 149.
- Arsa, I.G.N.W., 2019. "Analisis Sistem Cloud Computing IAAS Penyedia Server Cloud Dengan Standar NIST Special Publication 800-145," *Jurnal Sistem dan Informatika*.
- BiznetGIO. 2020. "Dokumentasi neo-cli", BiznetGIO homepage, 24 Juni 2020. <https://github.com/BiznetGIO/neo-cli>
- Daniel, G.,2015. "The Analysis of OpenStack Cloud Computing Platform: Features and Performance," *Journal of Telecommunications and Information Technology*, Vol.3, pp.52-57.
- Docker. 2020. "Get Started with Docker to Help Developer and Development Team Build and Ship Application". Docker Homepage, 24 Juni 2020. <https://www.docker.com/>
- Emilia, M. 2020. "What is container management and why is it important?", Tech Target Homepage. 24 Juni 2020. <https://searchitoperations.techtarget.com/definition/container-management-software>
- Faiq, W., 2017. "Implementasi Cloud Computing di Beberapa Instansi Pemerintahan". *Jurnal Masyarakat Telematika dan Informasi*, Vol.8 No.2, pp.97-108.

- Julio S., 2019. "ARGON: A Model-Driven Infrastructure Provisioning Tool," 2019 ACM/IEEE 22nd International Conference on Model Driven Engineering Languages and Systems Companion (MODELS-C).
- Kubernetes, 2020. "Kubernetes: Orkestrasi Kontainer dengan Skala Produksi, Otomatisasi Kontainer Deployment, Scaling, dan Management". Kubernetes webpage, 24 Juni 2020. <https://kubernetes.io/id/>
- Liming, Z., Ingo, W., Len, B., 2015. DevOps: A Software Architect's Perspective. 3rd edition. United States: Addison-Wesley Publisher.
- Mell, P., Grance, T. 2011. "The NIST Definition of Cloud Computing: Recommendations of the National Institute of Standards and Technology." NIST Special Publication 800-145. U.S.
- Muhammad, F., dkk., 2018. Perancangan dan Implementasi Network Functions Virtualization (NFV) Berbasis Cloud Computing dengan OpenStack. JATIKOM Jurnal Aplikasi dan Teori Ilmu Komputer, Vol.01 No.02, pp.1-6.
- Nguyen,A.T., et al., 2017. DevOps: Introducing Infrastructure-as-Code. 2017 IEEE/ACm International Conference on Software Engineering (ICSE). DOI: 10.1109/ICSE-C.2017.162
- Nusa Bali. 2019. "Pemasangan Fasilitas Wifi Gratis Sasar Lebih dari 4.000 Titik di Bali". Nusa Bali 12 April 2019. <https://www.nusabali.com/berita/50371/pemasangan-fasilitas-wifi-gratis-sasar-lebih-dari-4000-titik-di-bali>
- OpenStack, 2020. "What is OpenStack?", OpenStack homepage, 24 Juni 2020. <https://docs.openstack.org/ussuri>
- Peffer, K., Tuunanen, T., Rothenberger, M.A., Chatterjee, S. 2017. "A Design Science Research Methodology for Information System Research," Vol.24, Issue 3, Winter 2007-8, pp. 45-78.
- Pratama,I.P.A.E., 2014. Smart City Beserta Cloud Computing dan Teknologi - Teknologi Pendukung Lainnya. Bandung: Penerbit Informatika.
- Red Hat, Inc., 2020. "Automation: What is Orchestration?" Red Hat Homepage, 24 Juni 2020. <https://www.redhat.com/en/topics/automation/what-is-orchestration>
- Shivam, Poornima K., 2020. Infrastructure as Code: A Devops First Approach. International Research Journal of Engineering and Technology (IRJET), Volume: 07 Issue: 07, pp.3867-3869.
- Siebra, C., et al., 2018. From Theory to Practice: The Challenges of a DevOps Infrastructure as Code Implementation. In: ICSoft 13th 2018, International Conference on Software Technologies. Porto : Portugal July 26 – 28 2018. Portugal: Porto.
- Silvia, A., 2017. "An Infrastructure Modelling Tool for Cloud Provisioning," 14th IEEE International Conference on Services Computing (IEEE SCC 2017)At: Honolulu, Hawaii, USA.
- Stephen, W., BMC Blog, 2020. "IT Automation vs IT Orchestration: Using Both for Maximum Benefit", 24 Juni 2020. <https://www.bmc.com/blogs/it-orchestration-vs-automation-whats-the-difference>
- Sunaryo, Agus T., Kasmawi, 2017. "Rancang Bangun Server Cloud Computing di Politeknik Negeri Bengkalis". Jurnal INOVTEK POLBENG -SERI INFORMATIKA, Vol.2, No.1, pp. 33-40.
- Teguh, Mulyono. 2018. "E-Government Berbasis Cloud Computing Pada Pemerintah Daerah". Jurnal JIKE, Vol.1, No,1, pp.7-14.
- Yamato, Y., et al., 2014. "Development of template Management Technology for Easy Deployment of Virtual Resources on OpenStack," Journal of Cloud Computing: Advances, Systems and Applications, Vol.3, No.7, pp.1-12.