

## KATA PENGANTAR

Dalam kesempatan kali ini, peneliti dapat mengucapkan Alhamdulillah dan juga puji syukur kepada Tuhan Yang Maha Esa. Pembuatan tugas akhir ini adalah perjalanan peneliti di tahun akhir menempuh pendidikan S1 Sistem Informasi, Fakultas Rekayasa Industri, Telkom University.

Ucapan banyak terimakasih peneliti ucapkan kepada:

1. Bapak Umar Yunan Kurnia Septo Hedyanto, S.T., M.T. Selaku pembimbing 1 yang sudah memberikan banyak masukan dan arahan pada pembuatan tugas akhir ini hingga dapat menyelesaikan tugas akhir ini.
2. Bapak Muhammad Fathinuddin S.Si., M.T. Selaku pembimbing 2 yang sudah memberikan banyak masukan dan juga arahan pada pembuatan tugas akhir ini hingga dapat menyelesaikan tugas akhir ini.
3. Bapak Nofrian dan juga Ibu Yenny S.H. selaku kedua orang tua penulis yang selalu memberikan doa dan juga semangat selama menjalankan Pendidikan S1 Sistem Informasi khususnya dalam pengerjaan tugas akhir ini.
4. Kak Deta, Bang Rian, Mbak Onny, Bang Tito, Mbak Cia, Bang Halim, dan juga Bang Sena yang sudah memberikan semangat, doa, dan juga arahan dalam pengerjaan tugas akhir ini hingga akhirnya tugas akhir ini dapat peneliti selesaikan.
5. Seseorang dengan NIM 1202204322 yang sudah menemani peneliti hingga akhir.
6. Rekan kelompok tugas akhir peneliti, Jizdan Rizkiardi, Daniel Yusril Jasli, dan Akbar Febriyanto yang telah memberikan masukan dan arahan ketika peneliti merasa kesulitan dalam pengerjaan tugas akhir ini.

Terimakasih juga peneliti berikan untuk semua pihak yang secara tidak langsung berkontribusi dalam pembuatan hingga dapat menyelesaikan tugas akhir ini. Pada tugas akhir ini masih memiliki banyak kekurangan dalam penulisan, maka dari itu peneliti masih membutuhkan kritik dan saran.

# Daftar Isi

ABSTRAK .....	i
ABSTRACT .....	ii
LEMBAR PENGESAHAN .....	iii
LEMBAR PERNYATAAN ORISINALITAS .....	iv
KATA PENGANTAR .....	v
Daftar Isi.....	vi
Daftar Gambar.....	ix
Daftar Tabel .....	x
Daftar Lampiran .....	xi
Daftar Singkatan.....	xii
BAB I PENDAHULUAN.....	1
I.1 Latar Belakang.....	1
I.2 Perumusan Masalah .....	2
I.3 Tujuan Penelitian .....	3
I.4 Batasan Penelitian.....	3
I.5 Manfaat Penelitian .....	4
I.6 Sistematika Penulisan .....	4
BAB II TINJAUAN PUSTAKA .....	7
II.1 Penelitian Terdahulu.....	7
II.2 Alasan Pemilihan Teori, Kerangka Kerja, Atau Mekanisme .....	9
II.3 Cloud Computing .....	10
II.4 Server.....	13
II.5 Virtual Machine.....	14
II.6 Amazon Web Services .....	15

II.7 Amazon Elastic Compute Cloud .....	17
II.7.1 Instance Tujuan Umum .....	17
II.7.2 Instance Komputasi yang Dioptimalkan .....	18
II.7.3 Instance Memori yang Dioptimalkan .....	18
II.7.4 Instance Komputasi Terakselerasi .....	18
II.8 Software Development Life Cycles .....	19
<b>BAB III    METODOLOGI PENELITIAN .....</b>	<b>21</b>
III.1 Model Konseptual .....	21
III.2 Sistematika Penyelesaian Masalah .....	22
III.2.1    Tahap Awal .....	22
III.2.2 Tahap Analisis .....	23
III.2.3 Tahap Desain .....	23
III.2.4 Tahap Akhir .....	23
III.3 Pengumpulan Data .....	24
III.4 Pengolahan Data atau Pengembangan Produk .....	24
III.5 Metode Evaluasi .....	24
<b>BAB IV    ANALISIS DAN PERANCANGAN .....</b>	<b>25</b>
IV.1 Spesifikasi Kebutuhan Teknologi .....	25
IV.1.1 Perangkat Lunak .....	25
IV.1.2 Perangkat Keras .....	25
IV.2 Spesifikasi Amazon EC2 .....	26
IV.2.1 Spesifikasi Minimum .....	27
IV.2.2 Spesifikasi Rekomendasi .....	28
IV.2.3 Spesifikasi Tertinggi .....	30
IV.2.4 Security Group Default Cloud Server Amazon EC2 .....	32
IV.2.5 Security Group Custom Cloud Server Amazon EC2 .....	33

IV.3 Desain Sistem .....	35
IV.3.1 Topologi Rancangan Sistem Cloud .....	36
IV.4 Perancangan Cloud Server Menggunakan Amazon EC2 .....	37
IV.4.1 Menyiapkan Spesifikasi .....	37
IV.4.2 Launch Instance .....	37
IV.4.3 Memastikan Cloud Server Aktif .....	38
IV.4.4 Membuka Website Odoo 16 Menggunakan IP Cloud Server.....	39
IV.5 Skenario Pengujian .....	40
IV.5.1 Skenario Pengujian Load Amazon Web Services .....	40
IV.5.2 Skenario Pengujian Load Web Server .....	42
IV.5.3 Skenario Pengujian Load Aplikasi Berbasis Web .....	43
<b>BAB V HASIL DAN PEMBAHASAN .....</b>	<b>45</b>
V.1 Spesifikasi Rekomendasi Untuk Implementasi Cloud Server Odoo16....	45
V.2 Hasil Testing Menggunakan Instance Type Seri t2.micro .....	46
V.3 Evaluasi Akhir.....	47
<b>BAB VI KESIMPULAN DAN SARAN .....</b>	<b>49</b>
VI.1 Kesimpulan .....	49
VI.2 Saran .....	50
Daftar Pustaka .....	51
Lampiran .....	52

## Daftar Gambar

Gambar II. 1 Seri yang Ada Pada Instance Tujuan Umum.....	18
Gambar II. 2 Seri yang Ada Pada Instance Komputasi yang Dioptimalkan .....	18
Gambar II. 3 Seri yang Ada Pada Instance Memori yang Dioptimalkan.....	18
Gambar II. 4 Seri yang Ada Pada Instance Komputasi Terakselerasi .....	19
Gambar II. 5 Software Development Life Cycles (SDLC).....	20
Gambar III. 1 Model Konseptual Hevner .....	21
Gambar III. 2 Proses (SDLC).....	22
Gambar IV. 1 Spesifikasi Laptop.....	26
Gambar IV. 2 Security Group Default Amazon EC2 .....	33
Gambar IV. 3 Topologi Rancangan Sistem Cloud.....	36
Gambar IV. 4 Launch Instance .....	38
Gambar IV. 5 Command Untuk Memastikan Cloud Server Aktif.....	38
Gambar IV. 6 Konfigurasi Menggunakan PuTTY.....	39
Gambar IV. 7 Username dan Password Odoo 16 .....	40
Gambar IV. 8 Skenario Pengujian Load AWS .....	41
Gambar IV. 9 Skenario Pengujian Load Web Server .....	42
Gambar IV. 10 Skenario Pengujian Load Aplikasi Berbasis Web .....	43
Gambar V. 1 IP Address Cloud Server .....	46
Gambar V. 2 Halaman Utama Website Odoo16.....	47

## Daftar Tabel

Tabel II. 1 Penelitian Terdahulu.....	7
Tabel II. 2 Alasan Pemilihan Teori .....	9
Tabel IV. 1 Perangkat Lunak .....	25
Tabel IV. 2 Spesifikasi Laptop.....	26
Tabel IV. 3 Spesifikasi Minimum Cloud Server.....	27
Tabel IV. 4 Spesifikasi Rekomendasi Cloud Server Seri M.....	28
Tabel IV. 5 Spesifikasi Rekomendasi Cloud Server Seri C.....	29
Tabel IV. 6 Spesifikasi Rekomendasi Cloud Server Seri R.....	29
Tabel IV. 7 Spesifikasi Tertinggi Cloud Server Seri M.....	30
Tabel IV. 8 Spesifikasi Tertinggi Cloud Server Seri C.....	30
Tabel IV. 9 Spesifikasi Tertinggi Cloud Server Seri R.....	31
Tabel IV. 10 Security Group Inbound Rules .....	34
Tabel IV. 11 Security Group Outbound Rules.....	35
Tabel IV. 12 Spesifikasi Cloud Server yang Digunakan Untuk Testing .....	37
Tabel V. 1 Spesifikasi Pilihan.....	45

## Daftar Lampiran

Lampiran 1 Fitur Pada Instance Type Seri M7i .....	52
Lampiran 2 Rekomendasi Perangkat Keras Server Odoo.....	53
Lampiran 3 Spesifikasi Minimum, Rekomendasi, dan Tertinggi Cloud Server .	54
Lampiran 4 Security Group Amazon EC2 .....	56
Lampiran 5 Perancangan Server Berbasis Cloud Amazon EC2 Hingga Cloud Server Aktif.....	58

## Daftar Singkatan

Singkatan	Deskripsi	Halaman pertama kali digunakan
AWS	: Amazon Web Services	1
ERP	: Enterprise Resource Planning	1
SDLC	: Software Development Life Cycle	3
Amazon EC2	: Amazon Elastic Compute Cloud	3
Amazon EBS	: Amazon Elastic Block Storage	15
vCPU	: Virtual Central Processing Unit	26
RAM	: Random Access Memory	26
TCP	: Transmission Control Protocol	33
HTTP	: Hypertext Transfer Protocol	33
HTTPS	: Hypertext Transfer Protocol Secure	33
SSH	: Secure Shell	33
ICMP	: Internet Control Message Protocol	33

## BAB I PENDAHULUAN

### I.1 Latar Belakang

*Cloud Computing* atau biasa yang kita kenal dengan sebutan komputasi awan merupakan sebuah metode untuk dapat menyampaikan layanan melalui internet. Layanan yang dimaksud bisa berupa aplikasi seperti penyimpanan data, *server*, *database*, jaringan, dan juga perangkat lunak. Seiring dengan perkembangan zaman, maka teknologi yang kita gunakan saat ini pun akan ikut berkembang. Dengan adanya perkembangan teknologi tersebut, maka dibutuhkan yang namanya pengoptimalan atau memaksimalkan layanan yang ada saat ini.

Enterprise Resource Planning System (ERP System) atau yang berperan penting dalam sebuah perusahaan sebagai alat untuk meningkatkan efisiensi operasional, meminimalkan biaya infrastruktur, dan juga mempercepat pengambilan keputusan strategis (Anggi, 2020). Namun, untuk dapat menjalankan *ERP System* dengan lebih efisien, sebuah Perusahaan atau instansi akan sangat memerlukan infrastruktur yang kuat. Teknologi infrastruktur server tradisional sering kali akan memakan biaya yang cukup besar dalam hal pengadaan, pemeliharaan, serta rentan juga terhadap gangguan dan kendala. Dalam hal ini, teknologi komputasi awan atau *cloud computing* telah hadir untuk memberikan solusi yang cukup menjanjikan.

Telkom University khususnya Fakultas Rekayasa Industri nantinya akan mengimplementasikan *ERP System*. Dengan adanya pengimplementasian ini Fakultas Rekayasa Industri akan sangat terbantu untuk meningkatkan efisiensi dan juga mempercepat pengambilan keputusan strategis. Untuk dapat mengimplementasikan *ERP system* pada Fakultas Rekayasa Industri dibutuhkan infrastruktur *server* yang kuat. Dengan mengangkat media *cloud computing* untuk pengimplementasian *server* berbasis *cloud* akan membuat infrastruktur *server* yang akan dibangun menjadi kuat dengan segala kelebihan dari media *cloud computing*.

Pada permasalahan di atas, Amazon Web Services (AWS) merupakan salah satu jawaban untuk dapat mengoptimalkan atau memaksimalkan *ERP System* yang akan di jalankan. AWS merupakan salah satu penyedia layanan *cloud computing* terkemuka di dunia saat ini. Layanan yang diberikan pada AWS akan sangat berguna untuk dapat mendukung Telkom University khususnya Fakultas Rekayasa Industri dalam menjalankan *ERP System* mereka secara lebih efisien. Layanan *cloud* seperti komputasi elastis, penyimpanan yang dapat diperluas, dan basis data terkelola dapat memungkinkan Fakultas Rekayasa Industri untuk mengimplementasikan dan mengelola *ERP system* mereka tanpa harus menghadapi batasan infrastruktur fisik tradisional.

Mengangkat AWS untuk merancang infrastruktur *server* berbasis *cloud* dalam konteks *ERP System* dapat membawa manfaat yang cukup signifikan. Hal ini dapat membantu Telkom University khususnya Fakultas Rekayasa Industri menghemat biaya, meningkatkan skalabilitas, mengurangi kompleksitas pemeliharaan, dan akan memungkinkan akses lebih cepat dan lebih fleksibel ke data dan aplikasi bisnis. Oleh karena itu, penelitian ini akan membahas tentang perancangan infrastruktur *server* berbasis *cloud* menggunakan AWS untuk dapat mendukung implementasi *ERP System* pada Telkom University khususnya Fakultas Rekayasa Industri, serta manfaat dan tantangan yang terkait dengan pendekatan ini.

Dengan sebuah pemahaman yang mendalam tentang potensi perpindahan infrastruktur *server* ke *cloud* dan akan berfokus kepada AWS sebagai penyedia layanan, diharapkan penelitian ini dapat memberikan hal yang positif untuk *ERP System* yang akan dijalankan di Telkom University bandung khususnya pada Fakultas Rekayasa Industri.

## **I.2 Perumusan Masalah**

Rumusan masalah yang akan mendasari penelitian adalah:

1. Apa saja manfaat yang akan didapat oleh Telkom University khususnya Fakultas Rekayasa Industri dalam mengangkat AWS sebagai

infrastruktur *server* berbasis *cloud* untuk dapat menjalankan *ERP System*?

2. Bagaimana cara untuk dapat mengoptimalkan implementasi infrastruktur *server* berbasis *cloud* menggunakan *AWS* agar dapat memastikan efisiensi biaya yang diperlukan untuk *ERP System* Telkom University khususnya Fakultas Rekayasa Industri?

### **I.3 Tujuan Penelitian**

Tujuan dari penelitian ini adalah :

1. Dengan adanya infrastruktur *server* berbasis *cloud* ini diharapkan *ERP System* yang ada di Telkom university khususnya Fakultas Rekayasa Industri bisa dijalankan dengan lebih efisien dan lebih baik.
2. Memberikan spesifikasi dan biaya yang diperlukan untuk *server* berbasis *cloud* mulai dari spesifikasi minimum, spesifikasi rekomendasi, hingga spesifikasi tertinggi untuk dapat menjalankan implementasi *ERP System* pada Telkom University khususnya Fakultas Rekayasa Industri .

### **I.4 Batasan Penelitian**

Adapun batasan masalah dari penelitian ini yaitu sebagai berikut :

1. Penelitian ini hanya akan memberikan spesifikasi minimum, spesifikasi rekomendasi, hingga spesifikasi tertinggi untuk perancangan *server* berbasis *cloud* menggunakan Amazon EC2.
2. Penelitian ini akan memberikan biaya yang dibutuhkan untuk perancangan *server* berbasis *cloud* menggunakan Amazon EC2 mulai dari spesifikasi minimum, spesifikasi rekomendasi, hingga spesifikasi tertinggi.
3. Penelitian ini menggunakan metode penelitian Software Development Life Cycles (SDLC) yang dimana hanya menggunakan tahap analisis dan desain.
4. Penelitian ini hanya akan melakukan perancangan *server* berbasis *cloud* menggunakan Amazon EC2 mulai dari pemilihan instance hingga dapat

membuka website Odoo 16 menggunakan ip *cloud server* yang sudah dibuat.

5. Penelitian ini hanya melakukan *testing* pembuatan *cloud server* menggunakan instance yang *free* yang disediakan oleh AWS.
6. Penelitian ini tidak membahas hal lain terkait dengan ERP System, seperti halnya pelatihan untuk pengguna, atau pengembangan aplikasi.

### I.5 Manfaat Penelitian

Manfaat dari penelitian ini adalah :

1. Bagi Telkom University khususnya Fakultas Rekayasa Industri yang akan menjalankan *ERP System*, penelitian ini bermanfaat untuk dapat mengefisienkan kerja *dari ERP System* yang akan dijalankan.
2. Dari penelitian ini Telkom University khususnya Fakultas Rekayasa Industri akan bisa memilih mana yang lebih efisien dalam hal biaya antara *server* berbasis *cloud* menggunakan AWS dan juga *server* fisik.

### I.6 Sistematika Penulisan

Pada penelitian kali ini akan diuraikan dengan sistematika penulisan sebagai berikut:

#### Bab I           Pendahuluan

Pada bagian bab ini akan berisi uraian mengenai permasalahan yang akan di bahas mulai dari latar belakang, perumusan masalah, tujuan penelitian, batasan penelitian, manfaat penelitian, dan yang terakhir yaitu sistematika penulisan.

#### Bab II          Tinjauan Pustaka

Pada bab ini akan berisi literatur yang relevan dengan permasalahan yang akan di teliti dan akan dibahas pula hasil hasil dari penelitian terdahulu. Semaksimalnya akan terdapat lebih dari satu metodologi kerja yang akan disertakan pada bab ini untuk dapat menyelesaikan permasalahan atau meminimalisir gap

antara kondisi eksisting dengan target yang akan di capai. Pada akhir bab, analisis mengenai pemilihan metodologi kerja yang harus dijelaskan untuk dapat menentukan metodologi yang akan digunakan pada penelitian kali ini.

### Bab III Metodologi Penelitian

Metodologi penelitian adalah strategi dan juga langkah-langkah yang akan dilakukan dalam penelitian ini untuk dapat menjawab rumusan masalah yang sudah di bahas sebelumnya. Penyusunan metodologi penelitian harus dilakukan secara tepat agar dapat mengetahui apakah metode yang digunakan pada penelitian ini sudah sesuai atau belum. Pada bab ini juga akan dijelaskan secara rinci mulai dari tahap merumuskan masalah penelitian hingga mendapatkan hasil dari penelitian yang sudah di jalankan.

### Bab IV Analisis dan Perancangan

Pada bab ini analisis dan perancangan akan dijelaskan mulai dari desain topologi sistem cloud yang akan dibuat, spesifikasi software dan juga hardware, memberikan spesifikasi minimum dan rekomendasi untuk dapat membuat cloud server di Telkom University khususnya Fakultas Rekayasa Industri

### Bab V Hasil dan Pembahasan

Pada bab ini akan di lampirkan hasil dari penelitian yang sudah di lakukan mulai dari analisis hingga memberikan spesifikasi minimum dan rekomendasi untuk membuat cloud server. Dari hasil yang sudah di lampirkan, akan diberikan pembahasan mengenai mana yang lebih memungkinkan untuk di pilih dari spesifikasi yang sudah diberikan.

### Bab VI Kesimpulan dan Saran

Pada bab ini akan dijelaskan kesimpulan dari penelitian yang dilakukan dan juga jawaban dari pertanyaan penelitian yang

sudah disajikan di pendahuluan. Saran penelitian dikemukakan pada bab ini untuk penelitian selanjutnya.

## BAB II TINJAUAN PUSTAKA

### II.1 Penelitian Terdahulu

Penelitian terdahulu atau yang biasa kita sebut dengan kajian Pustaka merupakan studi atau penelitian yang telah dilakukan sebelumnya oleh peneliti-peneliti lain dalam sebuah bidang atau topik tertentu. Penelitian terdahulu sangatlah penting dalam sebuah proses penelitian yang akan kita jalani. Dengan adanya penelitian terdahulu akan memberikan landasan teoritis, metodologis, dan juga temuan-temuan yang dapat mendukung atau melengkapi penelitian yang sedang atau akan dilakukan. Berikut merupakan penelitian-penelitian terdahulu mengenai topik penelitian yang akan diteliti yaitu Perancangan Infrastruktur *Server* Berbasis *Cloud* untuk *ERP System* menggunakan AWS (*Amazon Web Services*):

Tabel II. 1 Penelitian Terdahulu

No	Nama	Judul Penelitian	Metode Penelitian	Hasil Penelitian	Perbedaan (Penelitian Sekarang)
1.	M. Ramadhan Julianti, Syaipul Ramadhan, Ade Mulyana (2019)(M. Ramadhan Julianti, 2019)	Perancangan <i>Server Cloud Computing Model Infrastruktur As A Service</i> Berbasis <i>Proxmox</i> pada PT Fortuna Mediatama	Metode yang digunakan dalam penelitian ini adalah metode kualitatif	Server <i>cloud computing</i> dapat mempermudah sistem administrator untuk melakukan <i>maintenance hardware</i> , dan pilihan <i>container (CT)</i> sebagai server dapat menjadikan solusi yang cepat, hemat memori, dan handal untuk membangun server website.	Perbedaan pada penelitian terdahulu dengan penelitian kali ini adalah peneliti menggunakan AWS ( <i>Amazon Web Services</i> ) sebagai penyedia layanan <i>cloud</i> yang akan digunakan untuk <i>ERP system</i> .
2.	Azharudin, Moch Fajar Hikmatulloh, Risdianto	Perancangan Sistem <i>Cloud Storage</i> Menggunakan	Metode penelitian yang digunakan	Hasil dari perancangan sistem data <i>storage</i> ini	Perbedaan pada penelitian terdahulu dengan

No	Nama	Judul Penelitian	Metode Penelitian	Hasil Penelitian	Perbedaan (Penelitian Sekarang)
	Irwan, Uus Firdaus (2023) (Azharudin, 2023)	<i>TreuNas Melalui VirtualBox</i>	adalah observasi, wawancara, dan konsep teori. Selain itu, juga dilakukan pengamatan langsung untuk dapat mengetahui kebutuhan dan peran sistem penyimpanan data.	menggunakan FreeNas melalui VirtualBox. Dimana sebelumnya penyimpanan masih menggunakan <i>Google Drive</i> yang semua berkasnya disatukan antara arsip satu dan yang lainnya. Tujuan dari perancangan sistem ini adalah untuk dapat memisahkan penyimpanan antara arsip yang dibutuhkan dengan arsip lainnya.	penelitian saat ini adalah untuk penelitian saat ini peneliti menggunakan <i>AWS (Amazon Web Services)</i> untuk penyedia layanan <i>cloud</i> dan metode yang peneliti gunakan adalah <i>SDLC (Software Development Life Cycle)</i> .
3.	Andy Rafsandy (2023) (Andy Rafsandy, 2023)	Perancangan Storage Server Menggunakan <i>Nextcloud</i> Pada PT Nusa Ina	Metode penelitian yang digunakan adalah observasi, wawancara, dan studi pustaka.	Menggunakan private cloud memungkinkan PT Nusa Ina untuk memiliki kendali penuh atas data yang disimpan dan dipertukarkan. Data tetap berada di dalam jaringan internal perusahaan, mengurangi risiko pelanggaran keamanan data dan mematuhi peraturan	Perbedaan pada penelitian terdahulu dengan penelitian saat ini adalah untuk penelitian saat ini peneliti menggunakan <i>AWS (Amazon Web Services)</i> untuk penyedia layanan <i>cloud</i> dan metode yang peneliti gunakan adalah <i>SDLC (Software Development</i>

No	Nama	Judul Penelitian	Metode Penelitian	Hasil Penelitian	Perbedaan (Penelitian Sekarang)
				privasi yang berlaku Administrator dan pengguna mempunyai akun masing-masing yang digunakan untuk mengakses layanan <i>nextcloud</i> .	<i>Life Cycle</i> .

## II.2 Alasan Pemilihan Teori, Kerangka Kerja, Atau Mekanisme

Pada sub ini bab ini akan berisi tentang analisis teori, kerangka kerja, atau mekanisme yang cocok dan tempat untuk bisa menjawab rumusan masalah yang sudah ditentukan. Berikut table perbandingan kerangka kerja dan justifikasi pemilihan dari penelitian ini:

Tabel II. 2 Alasan Pemilihan Teori

Karakteristik kerangka kerja	PPDIOO	NDLC	SDLC	Penelitian ini
Pendefinisian tahapan dalam proses	<i>Fleksible</i>	<i>Fleksible</i>	<i>Fleksible</i>	Penelitian ini membutuhkan kerangka kerja yang fleksibel.
Cakupan seluruh domain permasalahan	Mencakup : Persiapan dalam implementasi dan manajemen jaringan, strategi jaringan dan perancangan arsitektur, desain dan juga implementasi infrastruktur jaringan.	Mencakup : Perancangan, pengembangan, dan juga pengelolaan jaringan komunikasi, proteksi terhadap ancaman dan keamanan jaringan, skalabilitas jaringan untuk dapat menangani	Mencakup : Analisis kebutuhan <i>software</i> dan juga <i>hardware</i> , Implementasi dan juga integrasi perangkat lunak dalam sebuah sistem server, Pembaruan dan pemeliharaan perangkat lunak.	Penelitian ini hanya berfokus pada analisis dan juga desain dari sistem server berbasis <i>cloud</i> menggunakan penyedia layanan <i>Amazon Web Services (AWS)</i> khususnya

Karakteristik kerangka kerja	PPDIOO	NDLC	SDLC	Penelitian ini
		pertumbuhan trafik, pembaruan dan pemeliharaan infrastruktur jaringan.		Amazon EC2 untuk implementasi <i>ERP System</i> Odoo 16.
Pemilihan kerangka kerja	Kerangka kerja yang dipilih pada penelitian kali ini adalah SDLC dikarenakan kerangka kerja SDLC mencakup seluruh domain permasalahan yang ada di penelitian ini seperti: <ol style="list-style-type: none"> <li>1. Analisis kebutuhan spesifikasi beserta sistem keamanan <i>cloud server</i> yang akan digunakan.</li> <li>2. Desain sistem <i>cloud server</i> yang akan di implementasikan untuk <i>ERP System</i> Odoo 16.</li> </ol>			

### II.3 Cloud Computing

*Cloud computing* atau yang biasa kita sebut dengan komputasi awan merupakan suatu model yang akan memungkinkan akses jaringan yang mudah, nyaman, dan *on-demand* ke suatu sumber daya komputasi yang dapat dikonfigurasi bersama (misal, jaringan, server, penyimpanan, aplikasi, dan layanan) yang akan dapat segera diatur dan dilepaskan dengan usaha manajemen minimal atau interaksi dengan penyedia layanan (Mell & Grance, 2011). Jadi dengan adanya *cloud computing* ini akan memudahkan akses kedalam jaringan, server, penyimpanan, dll. Menurut (Mell & Grance, 2011) *cloud computing* memiliki lima karakteristik pokok, tiga model layanan, dan empat model penyebaran.

#### 1. Karakteristik pokok :

- *On-demand self-services* : Pengguna dapat menentukan sendiri kapabilitas komputasi, seperti waktu *server* dan penyimpanan jaringan, secara otomatis sesuai kebutuhan tanpa memerlukan keterlibatan manusia dengan setiap penyedia layanan.
- *Broad network access* : Kapabilitas yang bisa diakses melalui jaringan dan ditempuh melalui mekanisme standar yang akan memungkinkan pengguna dapat menggunakan berbagai *platform* (seperti ponsel, tablet, laptop, dan *workstation*).

- *Resource pooling* : Penggabungan sumber daya komputasi yang dimiliki penyedia untuk melayani beberapa konsumen virtual yang berbeda, dialokasikan secara dinamis sesuai dengan kebutuhan konsumen. Terdapat unsur kemandirian lokasi, dimana pelanggan pada umumnya tidak memiliki kendali atau pengetahuan tentang lokasi dari sumber daya yang disediakan, meskipun ada kemungkinan dapat menentukan lokasi pada tingkat abstraksi yang lebih tinggi, seperti negara, negara bagian, atau pusat data. Contoh dari sumber daya tersebut termasuk penyimpanan pemrosesan, *memory*, *bandwidth* jaringan, dan juga mesin virtual.
- *Rapid elasticity* : Kemampuan yang dapat diperbesar dan dikurangi secara elastis, terkadang dilakukan secara otomatis untuk dapat menyesuaikan dengan cepat sesuai dengan permintaan yang berkembang. Bagi pengguna, kemampuan yang tersedia seringkali terlihat tidak terbatas dan dapat disesuaikan dalam jumlah apapun kapan saja.
- *Measured Service* : Sistem *cloud computing* dapat secara otomatis memantau dan juga mengoptimalkan penggunaan sumber daya dengan menggunakan kemampuan pengukuran pada tingkat abstraksi tertentu yang sesuai dengan jenis layanan (seperti penyimpanan, pemrosesan, *bandwidth*, dan akun pengguna aktif). Penggunaan sumber daya akan dapat terus dipantau, dikontrol, dan dilaporkan, bertujuan untuk memberikan keterbukaan baik bagi penyedia layanan maupun konsumen yang menggunakan layanan tersebut.

## 2. Model layanan :

- *Cloud Software as a Service (SaaS)* : Kemampuan yang diberikan kepada pengguna untuk dapat menggunakan aplikasi yang disediakan oleh penyedia dan beroperasi diatas infrastruktur *cloud*. Aplikasi ini dapat diakses dari berbagai perangkat klien melalui antarmuka *web browser*, seperti email berbasis web. Pengguna tidak perlu mengelola

atau mengendalikan infrastruktur *cloud* yang mendasari, termasuk jaringan, *server*, sistem operasi, penyimpanan, atau bahkan konfigurasi aplikasi individual, kecuali dalam beberapa kasus terbatas pada pengaturan khusus aplikasi pengguna. Contoh dari layanan ini antara lain *Google Apps*, *Salesforce.com*, dan aplikasi jejaring sosial seperti Facebook.

- *Cloud Platform as a Service (PaaS)* : Kemampuan yang diberikan kepada pengguna untuk menyalurkan aplikasi yang telah mereka buat atau peroleh ke dalam infrastruktur komputasi *cloud* menggunakan bahasa pemrograman dan alat yang didukung oleh penyedia layanan. Pengguna tidak perlu mengurus atau mengontrol infrastruktur *cloud* yang mendasari, seperti jaringan, *server*, sistem operasi, atau penyimpanan. Meskipun begitu, pengguna tetap memiliki kendali atas konfigurasi hosting aplikasi yang mereka salurkan. Beberapa contoh yang telah menerapkan PaaS ini adalah Force.com dan investasi Microsoft Azure.
- *Cloud Infrastructure as a Service (IaaS)* : Kemampuan yang diberikan kepada pengguna untuk memproses, menyimpan, menjalin jaringan, dan sumber daya komputasi esensial lainnya, di mana pengguna dapat menyebarkan dan menjalankan perangkat lunak secara fleksibel, termasuk sistem operasi dan aplikasi. Meskipun pengguna tidak mengelola atau mengontrol infrastruktur awan yang mendasar, mereka tetap memiliki kendali atas sistem operasi, penyimpanan, aplikasi yang disebarkan, dan mungkin sebagian kendali atas komponen jaringan tertentu (seperti *firewall host*). Contoh layanan ini mencakup Amazon Elastic Compute Cloud dan Simple Storage Service.

### 3. Model penyebaran *cloud computing* :

- *Private cloud* : Infrastruktur *cloud* yang hanya semata-mata dioperasikan bagi suatu organisasi. Ini mungkin dimiliki, dikelola dan juga dijalankan oleh suatu organisasi, pihak ketiga atau

kombinasi dari beberapa pihak dan mungkin ada pada *on premis* atau *off premis*.

- *Community cloud* : Infrastruktur *cloud* akan digunakan secara bersama oleh beberapa organisasi dan akan mendukung komunitas tertentu yang telah berbagi concerns (misal ; misi, persyaratan keamanan, kebijakan, dan juga pertimbangan kepatuhan). Ini mungkin akan dikelola oleh organisasi atau pihak ketiga dan mungkin ada pada *on premis* atau *off premis*.
- *Public cloud* : Infrastruktur *cloud* yang akan disediakan untuk umum atau kelompok industri besar dan juga dimiliki oleh sebuah organisasi yang akan menjual layanan *cloud*.
- *Hybrid cloud* : Infrastruktur *cloud* yang merupakan komposisi dari dua atau lebih *cloud* (swasta, komunitas, atau publik) yang masih entitas unik namun terikat bersama oleh standar atau kepemilikan teknologi yang menggunakan data dan portabilitas aplikasi (misal, *cloud bursting for load-balancing between clouds*).

#### **II.4 Server**

*Server* merupakan seperangkat komputer yang berisi program-program yang mampu menghasilkan sebuah informasi dan informasi tersebut didistribusikan kepada komputer *client* yang akan mengaksesnya. *Server* sendiri secara sederhana dapat berupa satu buah komputer untuk beberapa layanan aplikasi, atau jika jaringannya lebih kompleks dan rumit.

Dengan demikian, *server* dapat diatur sedemikian rupa sehingga hanya memberikan satu atau beberapa layanan, sementara layanan lainnya ditangani oleh *server* terpisah. Ini menciptakan kerjasama dan kolaborasi antara beberapa *server* untuk menyediakan layanan dan informasi kepada sejumlah klien. Biasanya, konfigurasi *server* yang kompleks dan rumit ini diterapkan di organisasi besar seperti perusahaan kelas atas. Sementara itu, *server* yang terdiri dari satu komputer yang menyediakan beberapa layanan biasanya digunakan dalam lingkungan yang lebih kecil, seperti sekolah, perkantoran, atau usaha kecil

dan menengah (UKM) (Suryana, 2013). Menurut (Suryana, 2013) *server* sendiri memiliki berbagai macam jenis, berdasarkan fungsinya *server* dapat dibedakan menjadi :

1. *Web Server* : *Server* yang berfungsi untuk dapat memberikan layanan protocol http, contoh aplikasi *web server* yaitu *Apache, Microsoft IIS, Tomcat, Nginx*, dll.
2. *Database Server* : *Server* ini yang berfungsi untuk menyimpan data secara terpusat dan mendistribusikan ke klien melalui jaringan *wireless* ataupun kabel, contoh *database server* *MySQL, Postgres, MS SQL Server, Oracle, Interbase*, dll.
3. *FTP Server* : *Filezilla, FTPd, pro-FTPd, Wu-FTPd, ftpX, Troll-FTPd*.
4. *Mail Server* : *Mercury, Merak, sendmail, postfix*, dll.
5. *Print / File Server* : *Samba Server*
6. *DNS Server* : *Server* ini yang berfungsi untuk menerjemahkan alamat *host* menjadi *IP address*, contoh *Bind*.
7. *DHCP Server* : *Server* ini yang berfungsi memberikan *IP Address* secara otomatis ke komputer klien.
8. *Proxy Server* : Aplikasi ini akan diterapkan untuk dapat membatasi hak akses ke internet ataupun ke suatu *server*.

## **II.5 Virtual Machine**

*Virtual machine* atau mesin virtual merupakan aplikasi perangkat lunak yang dapat menjalankan Sebagian besar fungsi komputer fisik. Mesin virtual biasanya dikenal sebagai tamu yang dibuat dalam lingkungan komputasi lain yang di sebut “*host*”. Beberapa mesin virtual dapat berada dalam satu host pada satu waktu yang sama. (Margaret Rouse, 2020)

Menurut (Sofyan Mufti Prasetyo, 2024) mesin virtual memiliki berbagai macam fungsi yang sebagai berikut:

1. Berganti ke versi sistem operasi yang lebih baru  
Dengan menggunakan *virtual machine*, pengguna akan dapat melakukan adopsi metode yang aman untuk dapat menjalankan versi pengembangan.

Sebagai contoh pengguna dapat menjalankan sistem operasi Windows 10 pada komputer pengguna yang dimana pengguna juga sedang menjalankan sistem operasi Windows 8.

2. Virtualisasi dari *desktop* pengguna

Pengguna dapat menyimpan aplikasi *desktop* dan sistem operasi di dalam sebuah mesin virtual yang terpisah dari server dan menggunakan PC sebagai “*thin client*” ke server.

3. Melakukan pengujian dengan sistem operasi yang berbeda

Dalam hal ini pengguna juga dapat dengan mudah menginstal distribusi Windows yang berbeda dalam mesin virtual untuk melakukan suatu pengujian. Beberapa sistem operasi dapat dijalankan didalam server yang sama, hal ini dapat menghilangkan kebutuhan untuk mendedikasikan satu fisik server ke satu aplikasi.

*Virtual machine* atau mesin virtual juga memiliki berbagai macam jenis yang dapat digunakan diantaranya Virtual Box, Parallers, VMWare, QEMU, Microsoft Virtual PC, dan masih banyak lagi.

## II.6 Amazon Web Services

Amazon Web Services atau yang biasa disebut dengan AWS merupakan salah satu *platform* penyedia layanan *cloud* terkemuka saat ini. AWS akan menawarkan beberapa rangkaian produk berbasis *cloud global* yang luas termasuk juga komputasi, penyimpanan, *database*, analitik, jaringan, seluler, alat pengembangan, alat manajemen, *IoT*, keamanan, dan aplikasi perusahaan. Layanan-layanan yang ada di AWS akan dapat dimanfaatkan berdasarkan waktu, *volume*, perhitungan, waktu dan ruang penyimpanan (*Aws-Overview*, n.d.).

Menurut (Sasongko & Afrianto, n.d.) AWS menyediakan berbagai macam layanan komputasi awan untuk dapat membangun, mengelola, dan memperluas infrastruktur TI. Beberapa layanan yang diberikan diantaranya :

1. Komputasi

Layanan yang terkait dengan sumber daya komputasi diantaranya adalah Amazon Elastic Compute Cloud (Amazon EC2) untuk penyedia *server* virtual, Amazon EC2 untuk pengelola konteiner, dan AWS Lambda untuk dapat menjalankan kode tanpa *server*.

## 2. Penyimpanan

Layanan yang terkait dengan penyimpanan data diantaranya adalah Amazon Simple Storage Services (Amazon S3) untuk penyimpanan objek, Amazon Elastic Block Store (Amazon EBS) untuk dapat menyimpan blok, dan juga Amazon Glacier untuk menyimpan arsip.

## 3. Basis Data

Layanan yang terkait dengan pengelolaan basis data diantaranya adalah Amazon Relational Database Services (Amazon RDS) untuk pengelolaan basis data rasional, Amazon DynamoDB untuk dapat mengelola basis data NoSQL, dan Amazon Redshift untuk data *warehouse*.

## 4. Jaringan

Layanan yang terkait dengan jaringan diantaranya adalah Amazon Virtual Private cloud (Amazon VPC) untuk membuat jaringan yang aman pada sebuah komputasi awan, Amazon Route 53 untuk manajemen DNS, dan juga AWS Elastic Load Balancing untuk dapat membagi beban lalu lintas.

## 5. Analitik

Layanan yang terkait dengan analisis data diantaranya adalah Amazon Kinesis untuk *streaming* data, Amazon Elastic Map Reduce (Amazon EMR) untuk pemrosesan data besar, dan juga Amazon Athena untuk *query* data di S3.

## 6. Caching

Layanan untuk menyimpan salinan data dengan lokasi yang lebih dekat di seluruh dunia adalah sebuah konsep yang digunakan oleh jaringan pengiriman konten atau *content delivery network* (CDN) dinamakan Amazon Cloud Front.

## **II.7 Amazon Elastic Compute Cloud**

Amazon elastic compute cloud (Amazon EC2) Merupakan sebuah layanan web yang menyediakan atau memberikan komputasi yang aman dan juga dapat diubah ukurannya di *cloud*. Ini dirancang untuk dapat membuat komputasi skala web lebih mudah bagi pengembang (*Aws-Overview*, n.d.).

Antarmuka web yang sederhana pada Amazon EC2 memungkinkan pengguna untuk memperoleh dan mengonfigurasi kapasitas dengan gesekan minimal. Dengan antarmuka ini, pengguna memiliki kendali penuh atas sumber daya komputasi mereka dan dapat beroperasi di lingkungan komputasi Amazon yang telah terbukti kinerjanya. Amazon EC2 mengurangi waktu yang dibutuhkan untuk mendapatkan dan mem-*boot* instans *server* baru, menjadi hanya beberapa menit. Hal ini memungkinkan pengguna untuk dengan cepat menyesuaikan kapasitas, baik menambah maupun mengurangi, sesuai dengan perubahan kebutuhan komputasi mereka. Selain itu, Amazon EC2 mengubah dinamika ekonomi komputasi dengan memberikan kemampuan untuk membayar hanya untuk kapasitas yang benar-benar digunakan. Dengan menyediakan alat bagi pengembang dan administrator sistem, Amazon EC2 juga memfasilitasi pembangunan aplikasi yang tangguh terhadap kegagalan dan isolasi diri dari skenario kegagalan umum (*Aws-Overview*, n.d.).

Dalam Amazon EC2 memiliki berbagai macam tipe instance untuk dapat menjalankan *cloud server* yang akan di rancang. Berikut merupakan berbagai macam tipe instance beserta kegunaannya.

### **II.7.1 Instance Tujuan Umum**

Berdasarkan website AWS instance tujuan umum akan dapat memberikan keseimbangan dalam komputasi, sumber daya jaringan, memori, dan juga dapat digunakan untuk berbagai macam beban kerja. Instance ini sangat cocok untuk aplikasi yang menggunakan sumber daya ini dengan proporsi yang setara seperti server web.

M7g	M7i	M7i-flex	M7a	Mac	M6g	M6i	M6in	M6a	M5	M5n	M5zn	M5a
M4	T4g	T3	T3a	T2								

Gambar II. 1 Seri yang Ada Pada Instance Tujuan Umum

### II.7.2 Instance Komputasi yang Dioptimalkan

Berdasarkan website AWS instance komputasi yang dioptimalkan akan cocok untuk aplikasi terikat komputasi yang akan memanfaatkan prosesor dengan performa yang tinggi. Instans ini sangat cocok untuk beban kerja pemrosesan *batch*, transkode media, *server* web performa tinggi, komputasi performa tinggi (HPC), pemodelan ilmiah, *server* khusus *game* dan mesin *server* iklan, inferensi *machine learning*, serta aplikasi intensif komputasi lainnya.

C7g	C7gn	C7i	C7i-Flex	C7a	C6g	C6gn	C6i	C6in	C6a	C5	C5n	C5a
C4												

Gambar II. 2 Seri yang Ada Pada Instance Komputasi yang Dioptimalkan

### II.7.3 Instance Memori yang Dioptimalkan

Berdasarkan website AWS Instance ini dirancang agar dapat menghasilkan performa cepat untuk beban kerja yang akan memproses set data besar dalam sebuah memori.

R8g	R7g	R7i	R7iz	R7a	R6g	R6i	R6in	R6a	R5	R5n	R5b	R5a	R4
X2gd	X2idn	X2iedn	X2iezn	X1	X1e	Memori Tinggi	z1d						

Gambar II. 3 Seri yang Ada Pada Instance Memori yang Dioptimalkan

### II.7.4 Instance Komputasi Terakselerasi

Berdasarkan website AWS instance ini memanfaatkan akselerator perangkat keras atau *co-processor* untuk melaksanakan tugas-tugas seperti penghitungan *floating point*, pemrosesan grafis, atau pencocokan pola data dengan efisiensi lebih tinggi dibandingkan perangkat lunak yang berjalan pada CPU.

P5	P4	P3	P2	G6	G5g	G5	G4dn	G4ad	G3	Trn1	Inf2	Inf1	DL1
DL2q	F1	VT1											

Gambar II. 4 Seri yang Ada Pada Instance Komputasi Terakselerasi

## II.8 Software Development Life Cycles

Software development life cycles (SDLC) merupakan sebuah tahapan-tahapan pekerjaan yang akan dilakukan oleh analisis sistem dan programmer dalam membangun sebuah sistem informasi (Dwanoko, n.d.). Ada 6 tahapan secara umum yang ada di dalam *Software development life cycles* (SDLC) menurut (Dwanoko, n.d.) yaitu :

1. Analisis sistem, yaitu membuat sebuah analisis aliran kerja manajemen yang sedang berjalan.
2. Desain spesifikasi kebutuhan sistem, yaitu akan melakukan perincian mengenai apa saja dibutuhkan dalam melakukan pengembangan sistem dan membuat perancangan yang berkaitan dengan proyek sistem.
3. Implementasi sistem, yaitu sebuah tahapan untuk menjalankan sistem yang sesuai dengan fungsinya masing-masing.
4. Pengujian sistem, yaitu akan melakukan pengujian terhadap sistem yang sudah berhasil dibuat.
5. Penyebaran, yaitu akan melakukan penyebaran dari sistem yang sudah dibuat.
6. Pemeliharaan sistem, yaitu akan menerapkan dan memelihara sistem yang sudah berhasil dibuat.

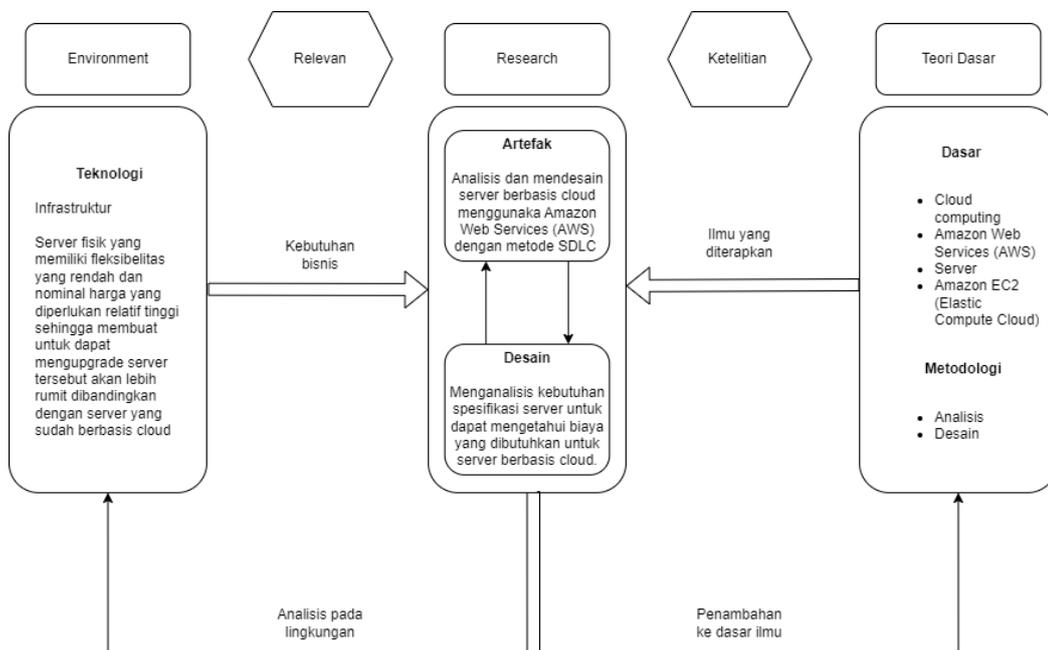


Gambar II. 5 *Software Development Life Cycles (SDLC)*

## BAB III METODOLOGI PENELITIAN

### III.1 Model Konseptual

Model konseptual merupakan sebuah gambaran abstrak yang akan menunjukkan hubungan antara factor-faktor tertentu yang akan memberikan dampak terhadap penelitian ini. Model konseptual juga akan membantu peneliti dalam menyusun peecahan masalah dan memberikan solusi dari permasalahan yang ada. Untuk model konseptual yang digunakan dalam penelitian ini sebagai berikut.



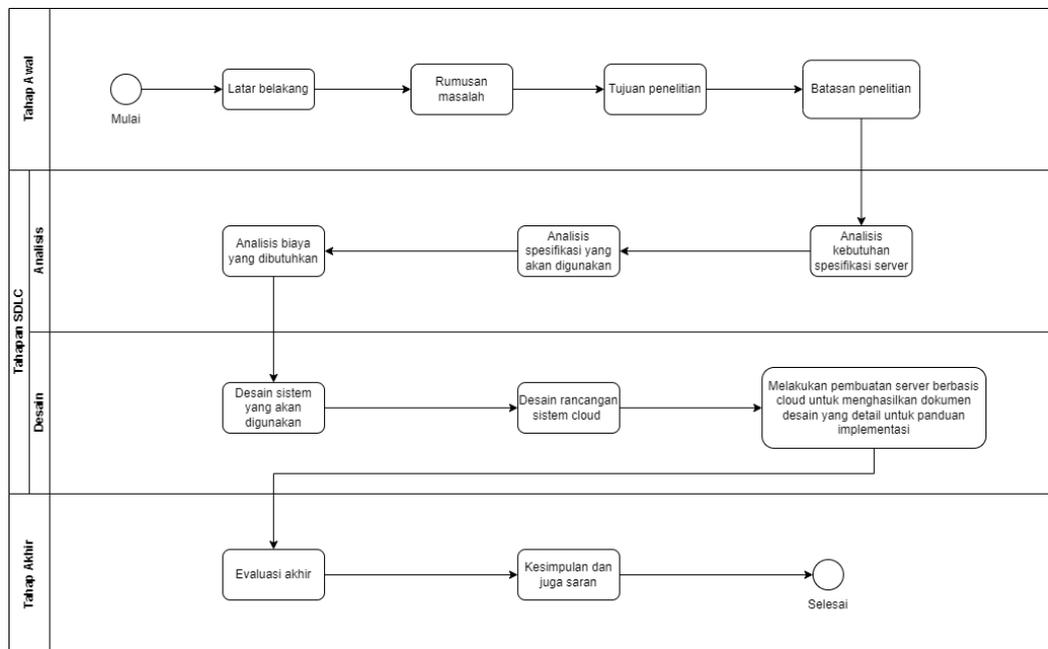
Gambar III. 1 Model Konseptual Hevner

Pada gambar III.1 Model Konseptual Hevner akan menjelaskan mengenai bagaimana model konseptual hevner yang sudah dibangun dalam penelitian ini meliputi dari lingkungan, penelitian dan juga dasar teori yang digunakan, pada aspek yang pertama yaitu lingkungan memiliki aspek teknologi yang menggambarkan tentang permasalahan yang ada pada teknologi saat ini dan batasan permasalahan. Teknologi yang digunakan untuk dapat mengatasi masalah yang terjadi pada lingkungan yaitu AWS khususnya menggunakan layanan Amazon EC2 yang dijadikan sebagai media *web server* untuk

menggantikan *server* fisik yang ada saat ini. Dalam metode konseptual penelitian ini menggunakan SDLC.

### III.2 Sistematika Penyelesaian Masalah

Sistematika penyelesaian masalah yang digunakan pada penelitian ini adalah untuk dapat menjelaskan tahapan-tahapan yang akan dilalui mulai dari awal yaitu mengidentifikasi masalah hingga akhir yaitu penyusunan laporan akhir dengan metode *Software Development Life Cycles* (SDLC) untuk *ERP system* pada Telkom University khususnya Fakultas Rekayasa Industri.



Gambar III. 2 Proses (SDLC)

#### III.2.1 Tahap Awal

Pada tahap penelitian ini akan dimulai dengan membuat latar belakang. Setelah membuat latar belakang, peneliti akan menyusun rumusan masalah berdasarkan latar belakang yang sudah dibuat. Setelah membuat rumusan masalah, peneliti akan menyusun batasan masalah yang dimana bertujuan agar penelitian ini akan berfokus pada pembahasan yang sudah dibuat. Setelah itu, peneliti akan

menyusun tujuan dari penelitian ini berdasarkan rumusan dan juga batasan masalah yang sudah dibuat.

### **III.2.2 Tahap Analisis**

Pada tahap ini peneliti akan menganalisis kebutuhan *server*. Setelah melakukan analisis kebutuhan *server* peneliti akan menganalisis spesifikasi *server* berbasis *cloud* yang ada di Amazon EC2 agar dapat memberikan rekomendasi spesifikasi *server* berbasis *cloud*. Setelah itu peneliti akan menganalisis biaya yang akan dikeluarkan untuk pengimplementasian *server* berbasis *cloud* berdasarkan spesifikasi yang sudah di tentukan.

### **III.2.3 Tahap Desain**

Tahap ini nantinya peneliti akan mendesain sistem apa saja yang akan digunakan mulai dari *server*, *virtual machine*, dan juga internet. Setelah melakukan desain sistem peneliti akan melakukan desain rancangan sistem *cloud*, pada tahap ini peneliti akan membuat topologi rancangan sistem *cloud* yang akan menggambarkan bagaimana *server* berbasis *cloud* ini bekerja. Setelah itu peneliti akan melakukan pembuatan *server* berbasis *cloud* agar dapat menghasilkan dokumen desain yang nantinya akan berguna menjadi panduan saat implementasi.

### **III.2.4 Tahap Akhir**

Pada tahap ini peneliti akan memberikan evaluasi akhir dari tahapan tahapan yang sudah dilalui sebelumnya. Setelah memberikan evaluasi akhir, peneliti akan memberikan kesimpulan dan juga saran dari penelitian yang sudah di lakukan oleh peneliti.

### **III.3 Pengumpulan Data**

Pada penelitian ini, pengumpulan data hanya membutuhkan satu tipe data yaitu sekunder yang dikumpulkan melalui referensi jurnal, artikel, website AWS, dan juga forum Odoo.

### **III.4 Pengolahan Data atau Pengembangan Produk**

Untuk pengolahan data akan dimulai dengan observasi untuk mendapatkan data yang sudah ada pada objek yang sudah ditentukan. Setelah mendapatkan data yang dibutuhkan, data akan dikelola dengan metode SDLC dengan menggunakan AWS sebagai *web service* untuk merancang spesifikasi *server* berbasis *cloud*.

### **III.5 Metode Evaluasi**

Metode evaluasi akan membahas mengenai bagaimana kondisi dari server *ERP system* Telkom University khususnya Fakultas Rekayasa Industri, peneliti akan melakukan pengambilan data dan melakukan analisa dari data yang sudah didapatkan. Jika hasil dari pengambilan data kurang bergitu memuaskan maka peneliti akan memberikan rekomendasi atau sebuah usulan melalui simulasi yang sudah dilakukan pada AWS.

## BAB IV ANALISIS DAN PERANCANGAN

### IV.1 Spesifikasi Kebutuhan Teknologi

Untuk dapat merancang dan menjalankan *cloud server* peneliti harus memiliki perangkat yang sudah memenuhi spesifikasi standar untuk dapat menjalankan *cloud server* tersebut. Perangkat yang dimaksud terbagi menjadi dua yaitu perangkat lunak dan juga perangkat keras.

#### IV.1.1 Perangkat Lunak

Perangkat lunak yang digunakan dalam penelitian ini ada beberapa AWS, Amazon EC2, dan juga PuTTY yang berperan sangat penting dalam penelitian ini. Untuk dapat merancang infrastruktur *cloud server*, 3 perangkat lunak tersebut merupakan kunci utama untuk dapat merancang dan menjalankan *cloud server*.

Tabel IV. 1 Perangkat Lunak

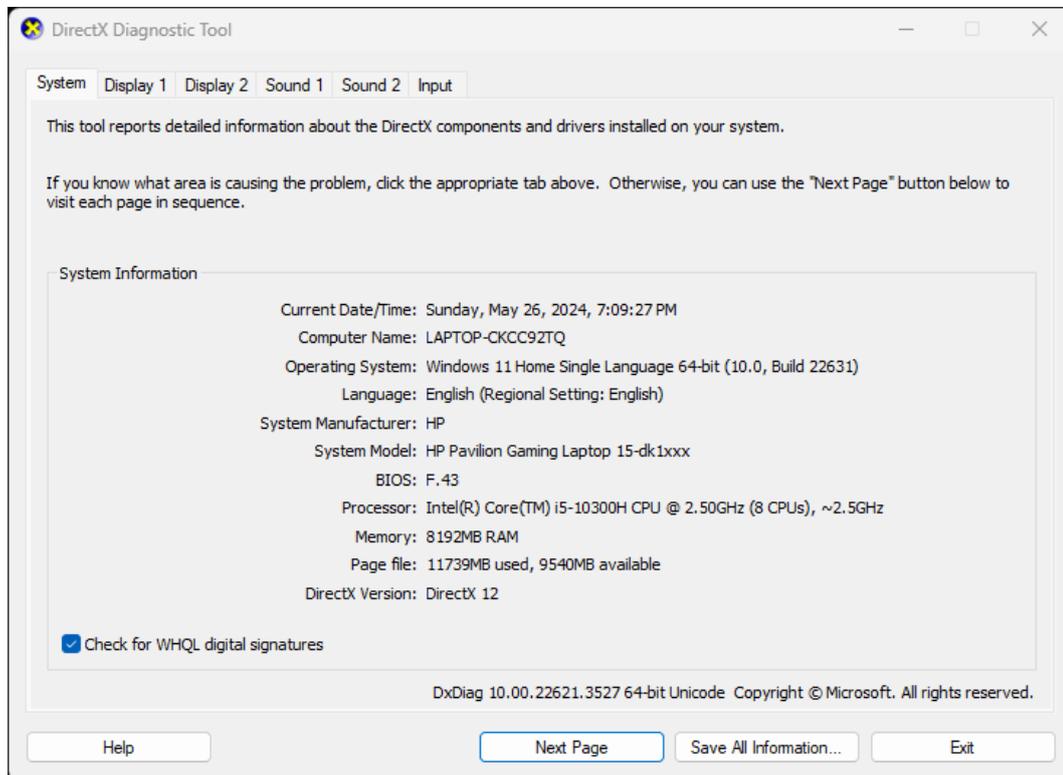
Perangkat Lunak	
Amazon Web Services (AWS)	Amazon Web Services (AWS) merupakan perangkat lunak berbasis web yang menyediakan berbagai macam layanan <i>cloud</i> , mulai dari penyimpanan, basis data, server, dll.
Amazon EC2	Amazon EC2 merupakan salah satu layanan yang ada didalam AWS. Amazon EC2 adalah layanan server berbasis <i>cloud</i> yang akan membuat server bisa di akses dari mana saja dan kapan saja.
PuTTY	PuTTY adalah emulator terminal <i>open-source</i> yang akan mendukung protokol jaringan mulai dari SCP, SSH, Telnet, rlogin, dan koneksi socket mentah.

#### IV.1.2 Perangkat Keras

1. Laptop HP Pavilion Gaming 15

Untuk penelitian kali ini peneliti menggunakan device laptop HP pavilion gaming 15 yang digunakan sebagai perangkat utama selama proses

penelitian ini berlangsung. Berikut spesifikasi perangkat keras yang digunakan :



Gambar IV. 1 Spesifikasi Laptop

Tabel IV. 2 Spesifikasi Laptop

Spesifikasi Laptop	
<i>Operating System</i>	Windows 11 Home 64-bit
<i>Language</i>	English
<i>System Manufacturer</i>	HP
<i>System Model</i>	HP Pavilion Gaming 15-dk1xxx
<i>BIOS</i>	F.43
<i>Processor</i>	Intel Core i5-10300h CPU @2.50GHZ
<i>Memory</i>	8192MB RAM
<i>Page file</i>	11739MB RAM used, 9540MB Available

## IV.2 Spesifikasi Amazon EC2

Untuk dapat menjalankan cloud server peneliti memilih untuk menggunakan Amazon EC2 sebagai layanan *cloud server* yang ada pada AWS. Pada Amazon

EC2 terdapat banyak instance *type* yang dapat digunakan sesuai dengan kebutuhan. Instance *type* juga memiliki 4 jenis instance mulai dari instance tujuan umum, instance komputasi yang dioptimalkan, instance memori yang dioptimalkan, dan juga instance komputasi terakselerasi. Dari setiap jenis instance yang ada memiliki berbagai macam seri yang dimiliki beserta dengan spesifikasi yang ada pada seri tersebut.

Pada penelitian kali ini peneliti hanya berfokus menggunakan *operating system* Linux dan akan memberikan spesifikasi minimum, rekomendasi, dan tertinggi beserta harga untuk berlangganan Amazon EC2 pada AWS.

#### IV.2.1 Spesifikasi Minimum

Spesifikasi minimum *server* berbasis *cloud* menggunakan Amazon EC2 berdasarkan analisis website AWS dan forum aplikasi Odoo 16 yang dapat menampung hingga 10 pengguna sebagai berikut :

Tabel IV. 3 Spesifikasi Minimum *Cloud Server*

Spesifikasi Minimum	
<i>Instance Type</i>	T3.Medium
<i>RAM</i>	4GiB
<i>CPU</i>	<i>Double-core</i> vCPU
<i>Storage</i>	EBS ( <i>Elastic Block Storage</i> )
<i>Operating System</i>	Linux
<i>User</i>	<i>Up to 10 users</i>
Harga	USD 0.0416 / jam atau setara dengan Rp676.99 / jam

Spesifikasi seperti yang tertera pada tabel IV.3 Spesifikasi Minimum *Cloud Server* dapat menampung hingga 10 pengguna. Instance ini sudah dilengkapi dengan RAM 4 GIB dan juga 2 vCPU. Pada instance ini penyimpanan yang digunakan adalah EBS, penyimpanan ini akan memungkinkan fleksibilitas dalam ukuran dan juga kinerja dari penyimpanan itu sendiri. Dengan spesifikasi tersebut, instance ini sangat cocok untuk aplikasi kecil dan tidak memerlukan ruang penyimpanan yang cukup besar.

Dalam perhitungan harga layanan Amazon EC2 untuk spesifikasi minimum *cloud server* yang diperuntukkan untuk aplikasi *ERP System* yaitu Odoo 16 adalah USD 0,0416 / jam atau setara dengan Rp676.99 / jam. Untuk berlangganan selama satu hari penuh itu akan diperkirakan mengeluarkan biaya sebesar USD 0,9984 / hari atau setara dengan Rp16.247 / hari. Sedangkan untuk berlangganan selama 1 tahun Amazon EC2 akan memungut biaya sebesar USD 364,416 / tahun atau setara dengan Rp5.930.505 / tahun.

Oleh karena itu hal yang menjadi alasan dari pemilihan spesifikasi minimum yaitu, instance t3.medium akan menawarkan keseimbangan untuk komputasi, *memory*, dan juga sumber daya jaringan. Instance ini juga dirancang untuk aplikasi dengan penggunaan CPU menengah. Dengan alasan yang sudah dijelaskan sebelumnya dan berdasarkan forum odoo 16 mengenai spesifikasi yang diperlukan untuk jumlah pengguna yang sedikit, spesifikasi pada tabel di atas sudah cukup memenuhi syarat spesifikasi minimum untuk 10 pengguna dan harga yang ditawarkan sangat terjangkau untuk dapat berlangganan layanan ini untuk penggunaan Odoo 16 pada Telkom University khususnya Fakultas Rekayasa Industri.

#### IV.2.2 Spesifikasi Rekomendasi

Spesifikasi rekomendasi *server* berbasis *cloud* menggunakan Amazon EC2 berdasarkan analisis website AWS dan forum aplikasi Odoo 16 sebagai berikut :

Tabel IV. 4 Spesifikasi Rekomendasi *Cloud Server* Seri M

Spesifikasi Rekomendasi			
Seri M			
<i>Instance type</i>	m7i.xlarge	<i>Instance Type</i>	m7i.2xlarge
<i>RAM</i>	16 GiB	<i>RAM</i>	32 GiB
<i>vCPU</i>	Quad-core vCPU	<i>vCPU</i>	8 vCPU
<i>Storage</i>	EBS ( <i>Elastic Block Storage</i> )	<i>Storage</i>	EBS ( <i>Elastic Block Storage</i> )
<i>Operating System</i>	Linux	<i>Operating System</i>	Linux
<i>User</i>	Up to 50 user	<i>User</i>	Up to 50 user
Harga	USD 0.2016 / jam atau setara dengan Rp3.280 / jam	harga	USD 0.4032 / jam atau setara dengan Rp6.561 / jam

Tabel IV. 5 Spesifikasi Rekomendasi *Cloud Server* Seri C

Spesifikasi Rekomendasi			
Seri C			
<i>Instance type</i>	c7i.xlarge	<i>Instance Type</i>	c7i.2xlarge
<i>RAM</i>	8 GiB	<i>RAM</i>	16 GiB
<i>vCPU</i>	Quad-core vCPU	<i>vCPU</i>	8 vCPU
<i>Storage</i>	EBS ( <i>Elastic Block Storage</i> )	<i>Storage</i>	EBS ( <i>Elastic Block Storage</i> )
<i>Operating System</i>	Linux	<i>Operating System</i>	Linux
<i>User</i>	Up to 50 user	<i>User</i>	Up to 50 user
Harga	USD 0.1785 / jam atau setara dengan Rp2.904 / jam	harga	USD 0.357 / jam atau setara dengan Rp5.809 / jam

Tabel IV. 6 Spesifikasi Rekomendasi *Cloud Server* Seri R

Spesifikasi Rekomendasi			
Seri R			
<i>Instance type</i>	r7i.xlarge	<i>Instance Type</i>	r7i.2xlarge
<i>RAM</i>	32 GiB	<i>RAM</i>	64 GiB
<i>vCPU</i>	Quad-core vCPU	<i>vCPU</i>	8 vCPU
<i>Storage</i>	EBS ( <i>Elastic Block Storage</i> )	<i>Storage</i>	EBS ( <i>Elastic Block Storage</i> )
<i>Operating System</i>	Linux	<i>Operating System</i>	Linux
<i>User</i>	Up to 50 user	<i>User</i>	Up to 50 user
Harga	USD 0.2646 / jam atau setara dengan Rp4.306 / jam	harga	USD 0.5292 / jam atau setara dengan Rp8.612 / jam

Spesifikasi seperti yang tertera pada tabel IV.4, IV.5, IV.6 dapat menampung hingga 50 pengguna. Instance ini sudah dilengkapi dengan RAM 8 hingga 64 GIB dan juga 4 hingga 8 vCPU. Pada instance ini penyimpanan yang digunakan adalah EBS, penyimpanan ini akan memungkinkan fleksibilitas dalam ukuran, kinerja, dan membuat pemrosesan akan lebih cepat dan efisien. Dengan spesifikasi tersebut, instance ini sangat cocok untuk aplikasi menengah yang membutuhkan performa yang tinggi dan skalabilitas.

Untuk spesifikasi rekomendasi *server* berbasis *cloud* menggunakan layanan Amazon EC2 AWS memiliki berbagai macam pilihan instance, penelitian kali ini memberikan spesifikasi rekomendasi menggunakan instance *type* seri M, C, dan R. Dalam perhitungan biaya layanan yang diperlukan untuk instance *type* seri M akan memerlukan biaya mulai dari sebesar USD 0.2016 / jam hingga USD

0.4032 / jam atau setara dengan Rp3.280 / jam hingga Rp6.561 / jam. Sedangkan untuk instance *type* seri C akan memerlukan biaya mulai dari sebesar USD 0.1785 / jam hingga USD 0.357 / jam atau setara dengan Rp2.904 / jam hingga Rp5.809 / jam. Untuk instance *type* seri R akan memerlukan biaya mulai dari USD 0.2646 / jam hingga USD 0.5292 / jam atau setara dengan Rp4.306 / jam hingga Rp8.612 / jam.

Oleh karena itu hal yang menjadi alasan dari pemilihan spesifikasi rekomendasi yaitu, instance *m7i*, *c7i*, dan *r7i* memiliki spesifikasi yang relatif sama. Dengan spesifikasi yang terdapat pada instance tersebut, implementasi *ERP system* pada Telkom University khususnya Fakultas Rekayasa Industri akan sangat terbantu untuk menjalankan *cloud server* yang akan di rancang menggunakan Amazon EC2. Tidak hanya itu, dengan spesifikasi tersebut juga dapat menampung 50 pengguna pada penggunaan Odoo 16 dan harga yang ditawarkan juga sangat terjangkau untuk berlangganan layanan tersebut dengan spesifikasi yang sudah tertera pada tabel di atas.

#### IV.2.3 Spesifikasi Tertinggi

Spesifikasi tertinggi *server* berbasis *cloud* menggunakan Amazon EC2 berdasarkan analisis website AWS dan forum aplikasi Odoo 16 sebagai berikut :

Tabel IV. 7 Spesifikasi Tertinggi *Cloud Server* Seri M

Spesifikasi Tertinggi	
Seri M	
<i>Instance Type</i>	m6i.32xlarge
<i>RAM</i>	512 GiB
<i>Storage</i>	EBS ( <i>Elastic Block Storage</i> )
<i>CPU</i>	128 vCPU
<i>Operating System</i>	Linux
<i>User</i>	More than 50 user
Harga	USD 6.144 / jam atau setara dengan Rp99.987 / jam

Tabel IV. 8 Spesifikasi Tertinggi *Cloud Server* Seri C

Spesifikasi Tertinggi	
Seri C	
<i>Instance Type</i>	c6i.32xlarge

Spesifikasi Tertinggi	
RAM	256 GiB
Storage	EBS ( <i>Elastic Block Storage</i> )
CPU	128 vCPU
Operating System	Linux
User	More than 50 user
Harga	USD 5.44 / jam atau setara dengan Rp88.530 / jam

Tabel IV. 9 Spesifikasi Tertinggi *Cloud Server* Seri R

Spesifikasi Tertinggi	
Seri R	
Instance Type	r6i.32xlarge
RAM	1024 GiB
Storage	EBS ( <i>Elastic Block Storage</i> )
CPU	128 vCPU
Operating System	Linux
User	More than 50 user
Harga	USD 8.064 / jam atau setara dengan Rp131.233 / jam

Spesifikasi seperti yang tertera pada tabel IV.7, IV.8, IV.9 dapat menampung lebih dari 50 pengguna. Instance ini sudah dilengkapi dengan RAM 256 hingga 1024 GIB dan juga 128 vCPU. Pada instance ini penyimpanan yang digunakan adalah EBS, penyimpanan ini akan memberikan kapasitas yang besar dan kinerja yang tinggi. Dengan spesifikasi tersebut, instance ini sangat cocok untuk aplikasi yang besar dan membutuhkan sumber daya komputasi yang tinggi dan juga skalabilitas maksimal.

Untuk spesifikasi tertinggi *server* berbasis *cloud* menggunakan layanan Amazon EC2 pada AWS memiliki berbagai macam pilihan instance, penelitian kali ini memberikan spesifikasi tertinggi menggunakan instance *type* seri M, C, dan R. Dalam perhitungan biaya layanan yang diperlukan untuk instance *type* seri M akan memerlukan biaya sebesar USD 6.144 / jam atau setara dengan Rp99.987 / jam. Sedangkan untuk instance *type* seri C akan memerlukan biaya sebesar USD 5.44 / jam atau setara dengan Rp88.530 / jam. Untuk instance *type* seri R akan memerlukan biaya sebesar USD 8.064 / jam atau setara dengan Rp131.233 / jam.

Oleh karena itu hal yang menjadi alasan dari pemilihan spesifikasi tertinggi yaitu, instance m6i, c6i, dan r6i memiliki spesifikasi yang relatif sama. Dengan

spesifikasi yang terdapat pada instance tersebut, implementasi *ERP system* pada Telkom University khususnya Fakultas Rekayasa Industri akan sangat terbantu untuk menjalankan *cloud server* yang akan di rancang menggunakan Amazon EC2. Akan tetapi untuk spesifikasi seperti tabel di atas menjadi tidak berfungsi secara optimal jika kebutuhan dari Odoo 16 yang akan di gunakan tidak mengharuskan spesifikasi sebesar spesifikasi yang sudah tertera. Dengan adanya spesifikasi tersebut, biaya yang dibutuhkan untuk berlangganan pun akan sangat tinggi.

#### **IV.2.4 Security Group Default Cloud Server Amazon EC2**

Pada dasarnya AWS khususnya Amazon EC2 secara otomatis sudah memiliki *security group default*. Jika saat melakukan *launch* instance tidak menentukan *security group*, maka instance tersebut sudah secara otomatis akan memiliki *security group default*. Namun jika tidak ingin menggunakan *security group default*, maka kita dapat membuat *security group custom* yang dapat disesuaikan dengan kebutuhan dari *server* berbasis *cloud* untuk Odoo 16 nantinya.

Berikut merupakan *security group default* yang sudah disiapkan oleh AWS khususnya Amazon EC2.

Jalur masuk			
Sumber	Protokol	Rentang Port	Deskripsi
<i>sg-1234567890abcdef0</i>	Semua	Semua	Mengizinkan lalu lintas jalur masuk dari semua sumber daya yang ditetapkan untuk grup keamanan ini. Sumbernya adalah ID dari grup keamanan ini.

Jalur keluar			
Tujuan	Protokol	Rentang Port	Deskripsi
0.0.0.0/0	Semua	Semua	Mengizinkan semua lalu lintas IPv4 ke luar.
::/0	Semua	Semua	Mengizinkan semua lalu lintas IPv6 ke luar. Aturan ini ditambahkan hanya jika VPC Anda memiliki blok CIDR IPv6 yang dikaitkan.

Gambar IV. 2 *Security Group Default Amazon EC2*

Dapat dilihat pada gambar IV.2 *Security Group Default Amazon EC2* bahwa secara *default*, *security group* dari AWS khususnya Amazon EC2 akan mengizinkan semua lalu lintas port mana saja untuk dapat masuk, dan mengizinkan semua lalu lintas IPv4 dan IPv6 untuk lalu lintas keluar. Untuk *security group default* tidak akan bisa dihapus atau dihilangkan.

#### **IV.2.5 *Security Group Custom Cloud Server Amazon EC2***

Untuk memperkuat keamanan dari *cloud server* yang akan dibuat nantinya, peneliti menyarankan untuk tidak hanya mengandalkan *security group default* melainkan juga untuk dapat membuat *custom security group* sesuai dengan kebutuhan. Pada penelitian kali ini peneliti dapat memberikan rekomendasi untuk *security group* sesuai dengan kebutuhan implementasi *cloud server* untuk Odoo 16 nantinya. Berikut merupakan *security group* yang direkomendasikan oleh peneliti:

Tabel IV. 10 *Security Group Inbound Rules*

Jalur Masuk				
Tipe Protokol	Protokol	Port	Source	Deskripsi
SSH	TCP	22	0.0.0.0/0	Mengizinkan seluruh alamat IPv4 publik komputer atau IP di jaringan lokal
HTTPS	TCP	443	0.0.0.0/0	Mengizinkan akses HTTPS ke dalam melalui alamat IPv4 manapun
HTTP	TCP	80	0.0.0.0/0	Mengizinkan akses HTTP ke dalam melalui alamat IPv4 manapun
<i>Custom</i> ICMP – IPv4	<i>Echo Request</i>	N/A	IPv4	Mengizinkan alamat IPv4 komputer dan alamat IPv4 tertentu

Pada tabel tersebut menunjukkan bahwa aturan *firewall* untuk jalur masuk mencakup beberapa protokol penting dan memiliki fungsi masing-masing. Protokol SSH dengan port 22 dan menggunakan TCP. Protokol ini akan memungkinkan akses dari seluruh alamat IPv4 publik ke komputer dalam jaringan lokal. Protokol ini berguna untuk mengendalikan komputer dari jarak jauh. Selanjutnya protokol HTTPS yang memiliki port 443 dan juga menggunakan TCP. Protokol ini akan mengizinkan akses aman ke situs web melalui alamat IPv4 manapun, dan juga akan menjamin enkripsi data. Selanjutnya protokol HTTP dengan port 80 dan TCP. Protokol ini akan memungkinkan akses ke situs web melalui alamat IPv4 manapun tanpa enkripsi, yang biasanya digunakan untuk lalu lintas web umum. Lalu yang terakhir adalah *custom* ICMP yang memiliki jenis *Echo Request*. Protokol ini tidak memiliki port yang ditentukan. Protokol ini akan mengizinkan alamat IPv4 tertentu untuk mengirim permintaan *echo* “ping”, yang dimana *echo* “ping” biasanya digunakan untuk memastikan konektivitas antara dua perangkat. Protokol ini diatur untuk dapat mengontrol dan juga mengamankan lalu lintas masuk ke jaringan. Protokol ini juga berguna untuk dapat memastikan hanya jenis lalu lintas yang diizinkan yang dapat masuk sesuai dengan kebutuhan jaringan.

Tabel IV. 11 Security Group Outbound Rules

Jalur Keluar				
Type Protokol	Protokol	Port	Source	Deskripsi
<i>All traffic</i>	All	All	0.0.0.0/0	Mengizinkan seluruh lalu lintas keluar

Pada tabel tersebut menunjukkan bahwa aturan *firewall* untuk jalur keluar dalam jaringan komputer adalah mengizinkan semua jenis lalu lintas, menggunakan semua protokol, dan juga semua port. Untuk alamat IP yang diizinkan dapat mengakses adalah seluruh alamat IPv4. Dengan adanya aturan ini akan mengizinkan seluruh lalu lintas keluar tanpa adanya batasan. Pada aturan ini memiliki fungsi untuk dapat memastikan bahwa semua perangkat dalam jaringan dapat mengirimkan data ke luar jaringan tanpa adanya pembatasan protokol atau juga port, hal ini yang akan membuat komunikasi antar perangkat menjadi lancar dan juga tidak terhalang dengan jaringan eksternal.

### IV.3 Desain Sistem

Perancangan *cloud server* untuk keperluan *ERP System* pada Fakultas Rekayasa Industri memiliki fungsi agar *server* yang akan dibangun tidak memakan biaya yang cukup besar, ruang penyimpanan dapat di *upgrade* sesuai kebutuhan dari aplikasi *ERP System*, mudah di akses dari mana saja dengan menggunakan koneksi internet. Komponen yang akan menyusun sistem ini adalah sebagai berikut:

#### 1. *Cloud Server*

Untuk *cloud server* yang akan digunakan yaitu AWS yang nantinya akan disesuaikan untuk spesifikasi keperluan aplikasi *ERP System* yang akan dijalankan.

#### 2. *Virtual Machine*

*Virtual machine* pada Amazon EC2 disebut dengan instance. Pada penelitian kali ini akan dilakukan analisis instance yang akan digunakan

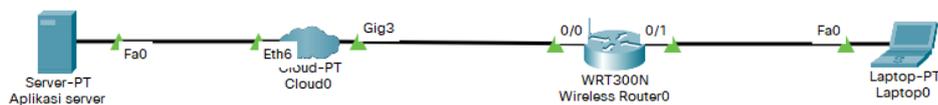
untuk implementasi *ERP system*. Nantinya spesifikasi yang akan direkomendasikan disesuaikan dengan kebutuhan dari Telkom University khususnya Fakultas Rekayasa Industri.

### 3. Internet

*Virtual machine* pada *cloud server* AWS yang sudah dispesifikasikan sesuai dengan kebutuhan dari aplikasi *ERP System* yaitu Odoo 16 ini harus terhubung dengan internet agar dapat menjalankan aplikasi dan juga mengakses *cloud server*.

#### IV.3.1 Topologi Rancangan Sistem *Cloud*

Dalam perancangan infrastruktur *cloud server* untuk *ERP System* akan dibutuhkan yang namanya topologi rancangan sistem *cloud* agar semua perencanaan yang sudah di tetapkan dapat berjalan dengan lancar dan dapat mendukung kebutuhan aplikasi yang akan dijalankan di dalam *cloud* secara maksimal.



Gambar IV. 3 Topologi Rancangan Sistem *Cloud*

Dapat dilihat dari gambar IV.1 bahwa nantinya aplikasi *ERP System* yaitu Odoo 16 akan di jalankan didalam *cloud server* AWS. Agar pengguna dapat mengakses *cloud server* tersebut pengguna diharuskan memiliki perangkat yang sudah terkoneksi dengan internet. *Cloud server* akan bisa di akses dari mana saja dan kapan saja menggunakan koneksi internet.

#### IV.4 Perancangan *Cloud Server* Menggunakan Amazon EC2

Dalam penelitian kali ini, peneliti akan melakukan perancangan *cloud server* menggunakan penyedia layanan AWS khususnya menggunakan layanan Amazon EC2. Untuk dapat merancang *cloud server* yang diinginkan terdapat beberapa langkah yang harus dilakukan mulai dari menyiapkan spesifikasi sesuai kebutuhan perusahaan hingga nantinya *server* yang sudah dibuat berhasil dijalankan. Berikut Langkah-langkah yang sudah peneliti lakukan untuk dapat merancang *cloud server* untuk *ERP System* khususnya Odoo 16 pada Telkom University, Fakultas Rekayasa Industri.

##### IV.4.1 Menyiapkan Spesifikasi

Pada tahap penyiapan spesifikasi *cloud server* yang akan dibuat, peneliti akan mengikuti Langkah-langkah yang sudah ada dan harus diikuti dalam AWS khususnya Amazon EC2. Langkah-langkah yang dimaksud adalah mulai dari pembuatan nama *cloud server*, mencari *Amazon Machine Image* yang ingin digunakan, memilih *Instance type* yang sesuai dengan kebutuhan, membuat *key pair (login)*, dan juga mengatur *network settings* yang sesuai. Dalam penelitian kali ini peneliti menggunakan spesifikasi sebagai berikut untuk melakukan percobaan :

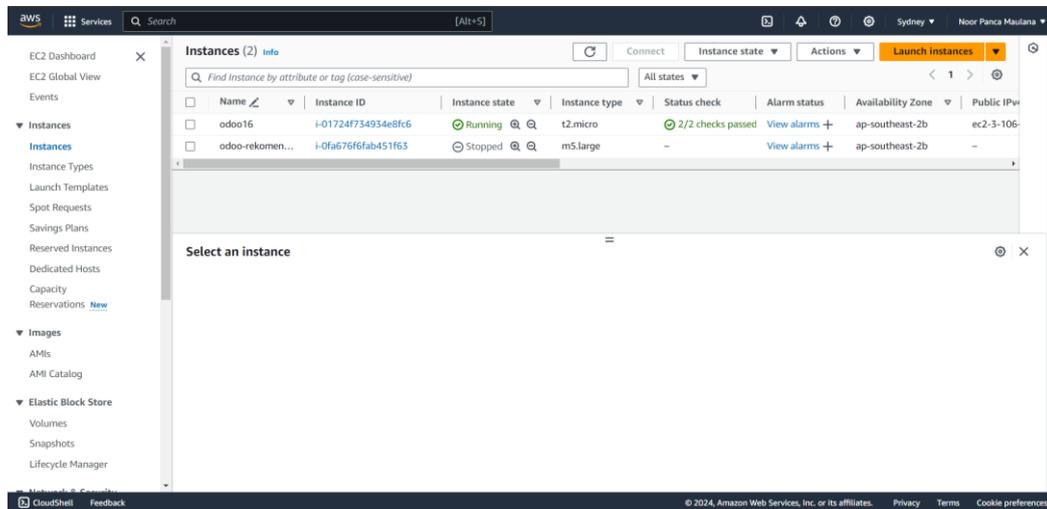
Tabel IV. 12 Spesifikasi *Cloud Server* yang Digunakan Untuk Testing

Spesifikasi Cloud Server	
<i>Name and Tags</i>	Odoo16
<i>Amazon Machine Image</i>	Bitnami-Odoo16
<i>Instance Type</i>	t2.micro
<i>Key Pair</i>	Pk-odoo16.ppk
<i>Security Group</i>	<ul style="list-style-type: none"><li>- Allow SSH traffic from</li><li>- Allom HTTPS traffic from the internet</li><li>- Allow HTTP traffic from the internet</li><li>- Custom ICMP – IPv4</li></ul>

##### IV.4.2 *Launch Instance*

Setelah menyiapkan spesifikasi yang sudah sesuai dengan kebutuhan perusahaan, lakukan *launch instance* untuk menjalankan *cloud server* sesuai

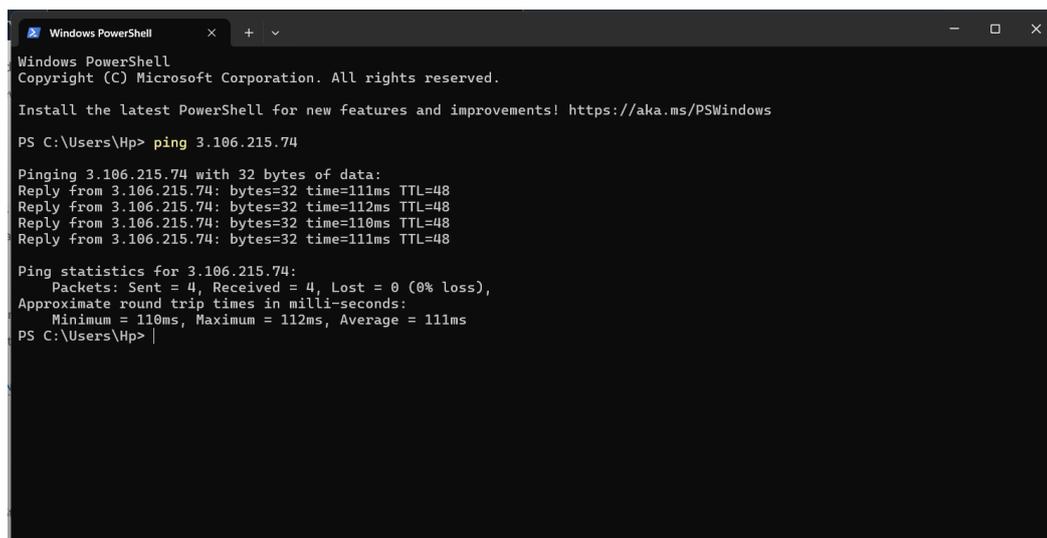
dengan spesifikasi yang sudah disiapkan. Berikut tampilan setelah *cloud server* berhasil dibuat :



Gambar IV. 4 *Launch Instance*

#### IV.4.3 Memastikan *Cloud Server* Aktif

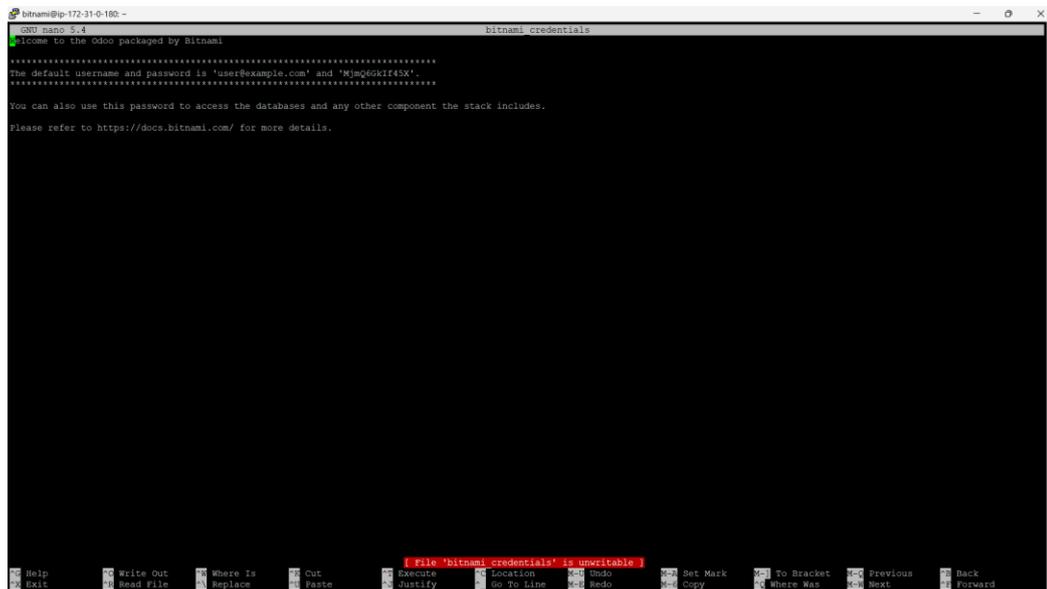
Untuk dapat lebih memastikan *cloud server* yang sudah dibuat itu berjalan, peneliti menjalankan perintah “ping ip” untuk melihat server sudah berjalan atau belum.



Gambar IV. 5 Command Untuk Memastikan *Cloud Server* Aktif



website Odoo 16 menggunakan alamat ip yang sudah tertera pada *cloud server* yang sudah di buat.

A screenshot of a terminal window titled 'bitnami@ip-172-31-0-180 -'. The terminal shows the output of a command, likely 'cat bitnami\_credentials', which displays the following text: 'Welcome to the Odoo packaged by Bitnami', 'The default username and password is 'user@example.com' and 'Mjq66k1f45X'', and a note that the password can also be used to access databases and other stack components, with a reference to 'https://docs.bitnami.com/'. The terminal has a dark background and a light-colored text. At the bottom, there is a menu bar with various icons and labels like 'Help', 'Exit', 'Write Out', 'Read File', 'Where Is', 'Replace', 'Cut', 'Paste', 'Execute', 'Justify', 'Location', 'Go To Line', 'Undo', 'Redo', 'Set Mark', 'Copy', 'To Bracket', 'Where Was', 'Previous', 'Next', 'Back', 'Forward'. A red error message 'File 'bitnami\_credentials' is unreadable' is visible in the terminal's status bar.

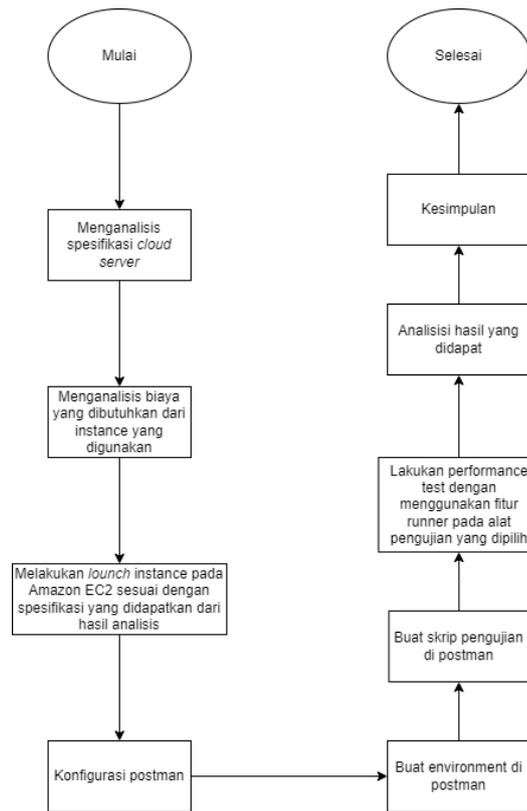
Gambar IV. 7 Username dan Password Odoo 16

## IV.5 Skenario Pengujian

Skenario pengujian ini di lakukan untuk dapat mengetahui performa dari setiap spesifikasi yang sudah di analisis sebelumnya. Dalam konteks ini, skenario pengujian mencakup AWS sebagai infrastuktur, web server, dan aplikasi berbasis web.

### IV.5.1 Skenario Pengujian *Load Amazon Web Services*

Skenario pengujian *load* adalah sebuah proses untuk dapat menguji bagaimana sebuah sistem, aplikasi, atau server berperforma di bawah beban tertentu. Berikut merupakan skenario pengujian *load* AWS.



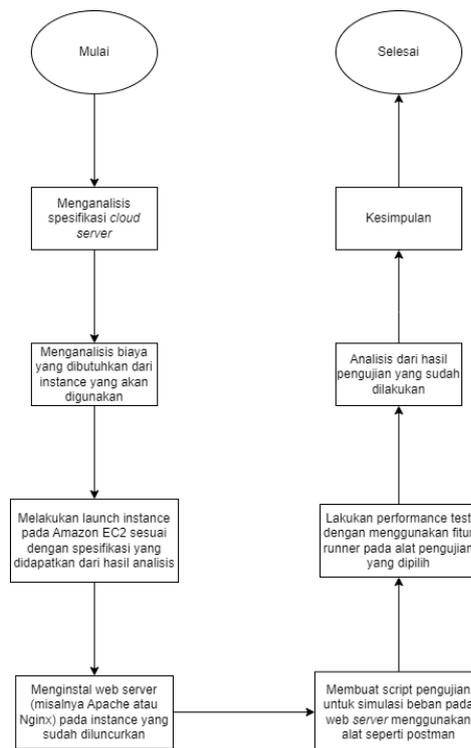
Gambar IV. 8 Skenario Pengujian *Load* AWS

Pada Gambar IV.8 Skenario Pengujian *Load* AWS dapat dilihat bahwa skenario pengujian dapat diuraikan menjadi beberapa langkah sebagai berikut. Pertama yaitu menganalisis spesifikasi yang dibutuhkan untuk dapat membuat cloud server pada Amazon EC2. Lalu setelah menganalisis spesifikasi, analisis biaya yang dibutuhkan dari spesifikasi yang sudah di analisis sebelumnya. Setelah menganalisis spesifikasi dan biaya yang dibutuhkan, lakukan peluncuran instance pada Amazon EC2. Setelah instance diluncurkan dan server sudah aktif, buat collection untuk API yang di host Amazon EC2 untuk melakukan konfigurasi postman. Lalu buat enirontment pada postman seperti base URL, API key, dll. Setelah melakukan pembuatan environment buat skrip pengujian pada postman pada tab “test”. Setelah itu lakukan *performance test* menggunakan fitur runner yang ada pada postman. Setelah itu analisis hasil dari *performance test* yang sudah dilakukan sebelumnya dan berikan kesimpulan.

Pada pengujian load AWS yang akan diukur adalah mulai dari CPU *utilization*, *memory utilization*, *disk*, *network*, dll.

#### IV.5.2 Skenario Pengujian *Load Web Server*

Skenario pengujian *load* adalah sebuah proses untuk dapat menguji bagaimana sebuah sistem, aplikasi, atau *server* berperforma di bawah beban tertentu. Berikut merupakan skenario pengujian *load web server*.



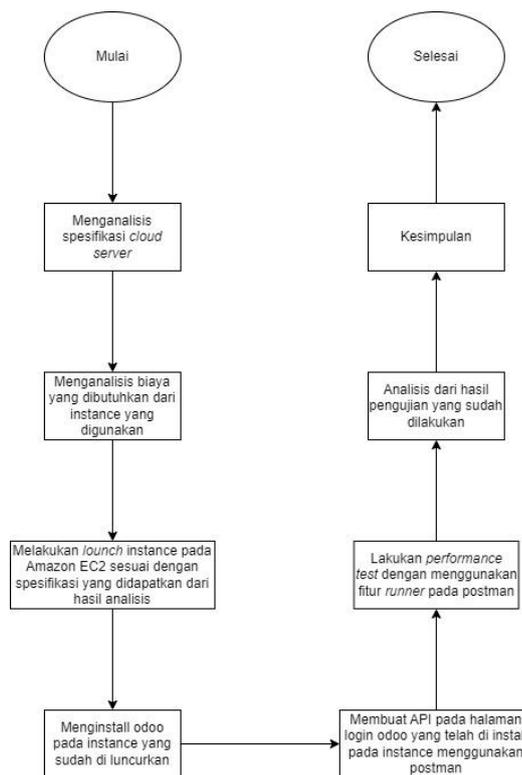
Gambar IV. 9 Skenario Pengujian *Load Web Server*

Pada Gambar IV.9 Skenario Pengujian *Load Web Server* dapat dilihat bahwa skenario pengujian dapat diuraikan menjadi beberapa langkah sebagai berikut. Pertama yaitu menganalisis spesifikasi yang dibutuhkan untuk dapat membuat cloud server pada Amazon EC2. Lalu setelah menganalisis spesifikasi, analisis biaya yang dibutuhkan dari spesifikasi yang sudah di analisis sebelumnya. Setelah menganalisis spesifikasi dan biaya yang dibutuhkan, lakukan peluncuran instance pada Amazon EC2. Setelah instance diluncurkan dan server sudah aktif,

lakukan instalasi web *server* seperti Apache atau Nginx pada instance yang sudah aktif. Setelah melakukan instalasi web server, buat script pengujian untuk simulasi beban pada web *server* sesuai dengan kebutuhan dari Fakultas Rekayasa Industri nantinya menggunakan postman. Setelah itu lakukan *performance test* dengan menggunakan fitur *runner* pada postman. Setelah hasil keluar, lakukan analisis hasil yang sudah di dapat. Setelah itu dari hasil yang sudah di dapat buat kesimpulan. Pada pengujian load web *server* yang akan diukur adalah *response time*, *throughput*, *error rate*, dll.

#### IV.5.3 Skenario Pengujian *Load* Aplikasi Berbasis Web

Skenario pengujian *load* adalah sebuah proses untuk dapat menguji bagaimana sebuah sistem, aplikasi, atau server berperforma di bawah beban tertentu. Berikut merupakan skenario pengujian load aplikasi berbasis web.



Gambar IV. 10 Skenario Pengujian *Load* Aplikasi Berbasis Web

Pada Gambar IV.10 Skenario Pengujian Load AWS dapat dilihat bahwa skenario pengujian dapat diuraikan menjadi beberapa langkah sebagai berikut. Pertama yaitu menganalisis spesifikasi yang dibutuhkan untuk dapat membuat cloud server pada Amazon EC2. Lalu setelah menganalisis spesifikasi, analisis biaya yang dibutuhkan dari spesifikasi yang sudah di analisis sebelumnya. Setelah menganalisis spesifikasi dan biaya yang dibutuhkan, lakukan peluncuran instance pada Amazon EC2. Setelah instance diluncurkan dan server sudah aktif, lakukan instalasi Odoo pada instance yang sudah aktif. Setelah melakukan instalasi dan Odoo dapat berjalan, buat API pada halaman login Odoo menggunakan postman. Setelah API berhasil dibuat, lakukan pengujian performa menggunakan fitur *runner* pada postman dengan cara melakukan simulasi akses API yang telah dibuat sebelumnya menggunakan virtual *user* dengan jumlah yang disesuaikan dengan kebutuhan dari Fakultas Rekaya Industri nantinya. Selanjutnya dari hasil pengujian performa, analisis hasil tersebut dan berikan kesimpulan dari hasil yang sudah didapat. Pada pengujian load aplikasi berbasis web yang akan diukur adalah *response time*, *throughput*, *error rate*, dll.

## BAB V HASIL DAN PEMBAHASAN

### V.1 Spesifikasi Rekomendasi Untuk Implementasi Cloud Server Odoo16

Berdasarkan analisis kebutuhan spesifikasi implementasi *server* berbasis *cloud* menggunakan SAWS untuk *ERP System* pada Telkom University khususnya Fakultas Rekayasa Industri, peneliti dapat memberikan rekomendasi untuk spesifikasi *server* berbasis *cloud* menggunakan AWS khususnya layanan Amazon EC2. Untuk spesifikasi pilihan sebagai berikut:

Tabel V. 1 Spesifikasi Pilihan

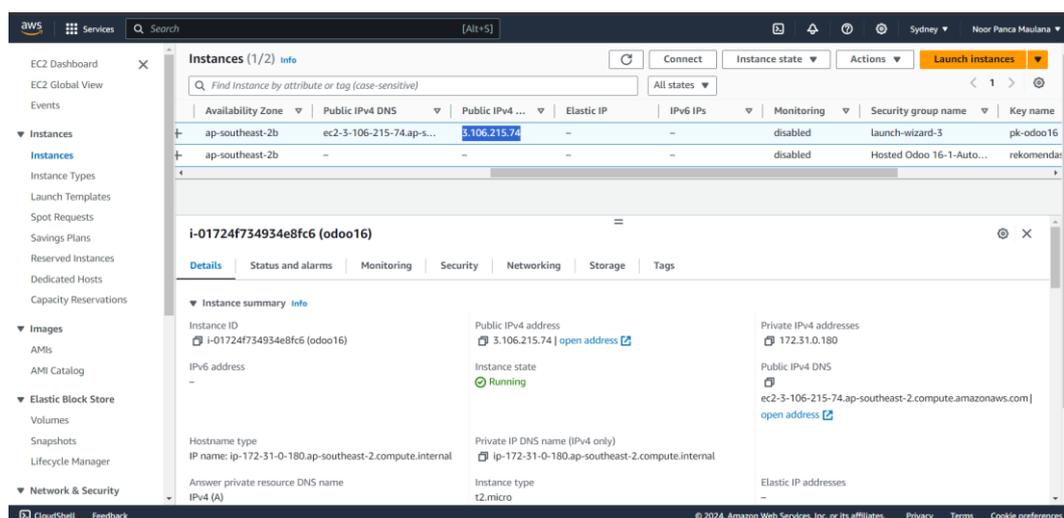
Spesifikasi Pilihan	
Instance	M7i.xlarge
Operating System	Linux
vCPU	4 vCPU
Memory	16GiB
Penyimpanan	Hanya EBS
Performa Jaringan (Gbps)	Hingga 12500 (Mbps)
User	Up to 50 user
Security Group	<i>Inbound</i> Allow SSH traffic from Allow HTTPS traffic from the internet Allow HTTP traffic from the internet Custom ICMP – IPv4 echo request  <i>Outbound</i> Allow all traffic
Harga	M7i.xlarge: USD 0.2016 / jam atau setara dengan Rp3.280 / jam

Instance *type* seri m7i.xlarge merupakan pilihan yang sangat tepat dalam hal spesifikasi maupun biaya yang ditawarkan. Dalam hal spesifikasi, instance ini sangat tepat menjadi pilihan untuk implementasi *cloud server ERP system* pada Telkom University khususnya Fakultas Rekayasa Industri. Dalam hal spesifikasi yang dimiliki, instance ini dapat menampung hingga 50 pengguna. Berdasarkan kebutuhan dari Fakultas Rekayasa Industri, yang nantinya akan menggunakan Odoo 16 ini sebanyak 25 – 30 pengguna. Oleh karena itu peneliti memberikan rekomendasi instance yang paling tepat adalah instance *type* seri m7i.xlarge.

Dalam hal biaya yang ditawarkan instance ini juga menawarkan biaya yang cukup terjangkau untuk dapat mengimplementasikan *cloud server*. Dengan spesifikasi yang dimiliki dan jika dibandingkan dengan instance lainnya dengan spesifikasi yang sama, instance ini memiliki biaya yang relatif terjangkau dibandingkan dengan instance lainnya yang memiliki spesifikasi yang sama. Instance ini dapat memberikan keseimbangan yang baik antara vCPU dan memori yang membuat instance ini sangat cocok untuk beban kerja Odoo 16 itu sendiri.

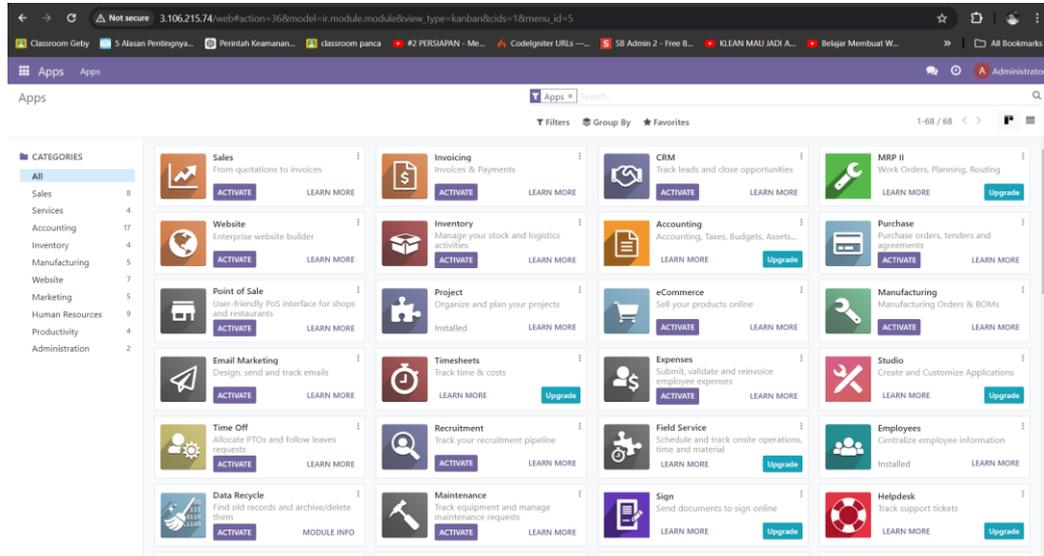
## V.2 Hasil Testing Menggunakan *Instance Type* Seri t2.micro

Pada penelitian kali ini, peneliti sudah melaksanakan pembuatan *cloud server* menggunakan layanan Amazon EC2 khususnya instance *type* seri t2.micro untuk mengetahui langkah-langkah apa saja yang harus dilakukan agar dapat membuat *cloud server* menggunakan AWS khususnya layanan Amazon EC2. Langkah-langkah ini pun akan sangat berguna untuk Telkom University khususnya Fakultas Rekaya Idusrti agar dapat mengimplementasikan *server* berbasis *cloud* untuk *ERP system* Odoo 16. Sesuai Langkah-langkah yang sudah dilakukan pada bab IV.4 Perancangan *Cloud Server* Menggunakan Amazon EC2, berikut merupakan hasil dari testing *cloud server* yang sudah berhasil dibuat dan berhasil masuk ke dalam website Odoo 16.



Gambar V. 1 IP Address Cloud Server

Pada gambar V.1 *IP Address Cloud Server* dapat dilihat *ip address cloud server* yang sudah dibuat dan nantinya dengan ip tersebut peneliti dapat membuka website Odoo 16.



Gambar V. 2 Halaman Utama Website Odoo16

Pada gambar V.2 Halaman Utama Website Odoo 16 dapat dilihat halaman utama dari website Odoo 16 yang sudah dapat dibuka menggunakan alamat ip *cloud server* yang sudah berhasil dibuat.

### V.3 Evaluasi Akhir

Evaluasi akhir dari penelitian ini adalah dapat menunjukkan keberhasilan dari implementasi dan pengujian *cloud server* menggunakan layanan Amazon EC2 khususnya menggunakan instance *type* seri t2.micro dan dapat memberikan rekomendasi instance beserta dengan sistem keamanan *cloud server* yang tepat untuk memenuhi kebutuhan implementasi *cloud server* Odoo 16. Implementasi dari perangkat lunak berhasil dilakukan dengan menggunakan layanan Amazon EC2, dan instance yang direkomendasikan adalah instance *type* seri m7i.xlarge. Seri ini akan menawarkan keseimbangan yang optimal antara vCPU dan juga memori, sehingga cocok untuk beban kerja Odoo 16 yang akan di implementasikan nantinya. Untuk testing pembuatan *cloud server* menggunakan

instance *type* seri t2.micro berhasil menunjukkan bahwa *cloud server* yang dibuat dapat diakses dan digunakan untuk membuka Odoo 16.

## BAB VI KESIMPULAN DAN SARAN

### VI.1 Kesimpulan

Berdasarkan dari hasil analisis dan perancangan *cloud server* yang sudah dilakukan, peneliti dapat menarik kesimpulan sebagai berikut:

1. Penggunaan AWS akan dapat memudahkan Telkom University khususnya Fakultas Rekayasa Industri dalam mengimplementasikan *server* berbasis *cloud* untuk *ERP System* Odoo 16. Dengan AWS sebagai penyedia layanan, Fakultas Rekayasa Industri tidak hanya akan dimudahkan dalam hal merancang dan mengimplementasikan *cloud server* tetapi juga dapat mengurangi biaya pengimplementasian *ERP System* Odoo 16 karna biaya yang ditawarkan oleh AWS sangat terjangkau sesuai kebutuhan dan spesifikasi. *Server* berbasis *cloud* ini nantinya juga akan memudahkan akses dari mana saja dan kapan saja menggunakan koneksi internet yang akan membuat pengelolaan *server* dan juga penyimpanan menjadi lebih efisien dan lebih fleksibel.
2. Penelitian ini juga dapat menunjukkan bahwa implementasi dan pengujian *cloud server* menggunakan Amazon EC2 untuk *ERP System* Odoo16 dapat berjalan dengan baik. Dengan mangangkat instance *type* seri *m7i.xlarge* sebagai instance yang direkomendasikan oleh peneliti, akan memberikan keseimbangan yang sangat optimal antara vCPU dan juga memori yang membuat instance ini sangat cocok untuk beban kerja Odoo16. Instance *type* seri *m7i.xlarge* juga memenuhi kebutuhan spesifikasi yang diharapkan karena adanya kelengkapan fitur dan kemudahan penggunaan AWS. Tidak hanya dalam hal spesifikasi, namun untuk biaya yang akan dikeluarkan juga cukup terjangkau untuk dapat mengimplementasikan *ERP System* menggunakan AWS khususnya Amazon EC2 dengan mangangkat instance *type* seri *m7i.xlarge* sebagai spesifikasi pilihan.

## VI.2 Saran

Berdasarkan dari hasil analisis mengenai harga dan spesifikasi dari kebutuhan *cloud server* menggunakan AWS khususnya Amazon EC2, saran yang dapat peneliti berikan kepada peneliti selanjutnya sebagai berikut:

1. Untuk dapat mengimplementasi *server* berbasis *cloud* menggunakan AWS khususnya Amazon EC2, peneliti selanjutnya dapat mencari tau lagi tentang *instance type* seri yang lain untuk dapat membandingkan performa dan biaya yang lebih efisien.
2. Peneliti selanjutnya disarankan untuk dapat melakukan penelitian lebih lanjut mengenai praktik keamanan dari perancangan *server* berbasis *cloud* agar *server* berbasis *cloud* yang akan di implementasi nantinya akan lebih aman.
3. Untuk mengimplementasikan *server* berbasis *cloud* menggunakan AWS khusus nya Amazon EC2, perlu dilakukan penelitian lebih lanjut mengenai biaya lebih mendalam agar dapat mempertimbangkan biaya jangka panjang yang akan dikeluarkan.
4. Peneliti selanjutnya disarankan untuk dapat melakukan pengujian skalabilitas agar dapat mengetahui bagaimana *instance* yang dipilih dapat menangani peningkatan beban kerja.

## Daftar Pustaka

- Andy Rafsandy. (2023). *View of PERANCANGAN STORAGE SERVER MENGGUNAKAN NEXTCLOUD PADA PT NUSA INA.*
- Anggi. (2020). ERP adalah: Pengertian, Manfaat, Fitur, dan Pentingnya Bagi Bisnis Anda. In <https://Accurate.Id/Marketing-Manajemen/Erp-Adalah/>.
- aws-overview.* (n.d.).
- Azharudin, M. F. H. R. I. U. F. (2023). *View of PERANCANGAN SISTEM CLOUD STORAGE MENGGUNAKAN TRUENAS MELALUI VIRTUALBOX.*
- Dwanoko, Y. S. (n.d.). IMPLEMENTASI SOFTWARE DEVELOPMENT LIFE CYCLE (SDLC) DALAM PENERAPAN PEMBANGUNAN APLIKASI PERANGKAT LUNAK. In *Jurnal Teknologi Informasi* (Vol. 7, Issue 2).
- M. Ramadhan Julianti, S. R. A. M. (2019). *perancangan-server-cloud-computing-model-bd3c8dff.*
- Margaret Rouse. (2020, August 14). *Virtual Machine.*
- Mell, P. M., & Grance, T. (2011). *The NIST definition of cloud computing.* <https://doi.org/10.6028/NIST.SP.800-145>
- Sasongko, N., & Afrianto, I. (n.d.). *Tinjauan Literatur: Performa Komputasi Awan Amazon Web Services (AWS).*
- Sofyan Mufti Prasetyo, M. B. A. D. A. M. F. M. A. R. (2024). *View of Mesin Virtual (Virtual Machine)\_ Sekilas Tentang Tujuan, Fungsi, Keuntungan, Dan Pengelolaan Dari Mesin Virtual.*
- Suryana, O. (2013). *Server dan Web Server.* <https://www.researchgate.net/publication/327338081>

# Lampiran

## Lampiran 1 Fitur Pada *Instance Type* Seri M7i

### 1. *Instance type* seri m7i



The screenshot shows the AWS Instance Type selection interface. The 'M7i' instance type is highlighted with a blue border. Below the instance type selection, there is a description and a list of features.

[Instans M7i Amazon EC2](#) didukung oleh prosesor Intel Xeon Scalable Generasi ke-4 dan menghadirkan performa harga lebih baik hingga 15% dibandingkan dengan instans M6i.

**Fitur:**

- Prosesor Intel Xeon Scalable Generasi ke-4 hingga 3,2 GHz (Sapphire Rapids 8488C)
- Advance Matrix Extensions (AMX) yang baru mempercepat operasi perkalian matriks
- 2 ukuran metal: m7i.metal-24xl dan m7i.metal-48xl
- Akselerator bawaan diskret (hanya tersedia pada ukuran bare metal M7i)—Akselerator Streaming Data (DSA), Akselerator Analitik dalam Memori (IAA), dan Teknologi QuickAssist (QAT)—yang memungkinkan pemindahan dan akselerasi operasi data yang efisien, yang membantu mengoptimalkan performa untuk basis data, enkripsi dan kompresi, serta beban kerja manajemen antrean
- Memori DDR5 terbaru, yang menawarkan bandwidth lebih tinggi dibandingkan DDR4
- Mendukung enkripsi memori selalu aktif menggunakan Intel Total Memory Encryption (TME)
- Mendukung hingga 128 lampiran volume EBS per instans
- Hingga 192 vCPU dan memori 768 GiB
- Didukung oleh [AWS Nitro System](#), yaitu kombinasi perangkat keras khusus dan hypervisor ringan

## Lampiran 2 Rekomendasi Perangkat Keras Server Odoo

### 1. Spesifikasi perangkat keras yang direkomendasikan

Criteria	Up to 10 Users	Up to 50 Users	More than 50 Users
<b>CPU</b>	Dual-core CPU	Quad-core CPU	Quad-core or higher CPU
<b>RAM (Memory)</b>	2 GB or more	4-8 GB or more	16 GB or more
<b>Storage</b>	10 GB or more	20 GB or more	50 GB or more (depending on data)
<b>Network</b>	Stable internet connection	Stable internet connection	Stable internet connection
<b>Operating System</b>	Linux (e.g., Ubuntu or CentOS)	Linux (e.g., Ubuntu or CentOS)	Linux (e.g., Ubuntu or CentOS)
<b>Database</b>	PostgreSQL	PostgreSQL	PostgreSQL
<b>Additional Notes</b>	SSDs for improved DB performance	SSDs for improved DB performance	SSDs for improved DB performance
	Regular backups and monitoring	Regular backups and monitoring	Regular backups and monitoring

### 2. Spesifikasi server berbasis *cloud* menggunakan *Amazon Web Services* (AWS)

#### **Amazon Web Services (AWS) Server for Odoo with 150 users**

This is the latest Odoo server requirements we used for project of 150 users:

- compute-optimized instance type that offers a balance of CPU and memory resources
- 6 vCPUs (virtual CPUs)
- 64 GB RAM
- 50GB SSD for root
- 34 media storage
- DB dedicated server:
  - 4vCPU
  - 16GB RAM
  - EBS

## Lampiran 3 Spesifikasi Minimum, Rekomendasi, dan Tertinggi *Cloud Server*

### 1. Spesifikasi minimum

Viewing 714 of 714 available instances

Q t3.medium X 1 match < 1 >

Instance name ▲	On-Demand hourly rate ▼	vCPU ▼	Memory ▼	Storage ▼	Network performance ▼
t3.medium	\$0.0416	2	4 GiB	EBS Only	Up to 5 Gigabit

### 2. Spesifikasi rekomendasi

m7i.xlarge	\$0.2016	4	16 GiB	EBS Only	Up to 12500 Megabit
m7i.2xlarge	\$0.4032	8	32 GiB	EBS Only	Up to 12500 Megabit

c7i.xlarge	\$0.1785	4	8 GiB	EBS Only	Up to 12500 Megabit
c7i.2xlarge	\$0.357	8	16 GiB	EBS Only	Up to 12500 Megabit

r7i.xlarge	\$0.2646	4	32 GiB	EBS Only	Up to 12500 Megabit
r7i.2xlarge	\$0.5292	8	64 GiB	EBS Only	Up to 12500 Megabit

### 3. Spesifikasi Tertinggi

Viewing 714 of 714 available instances

Q m6i.32 X 1 match < 1 >

Instance name ▲	On-Demand hourly rate ▼	vCPU ▼	Memory ▼	Storage ▼	Network performance ▼
m6i.32xlarge	\$6.144	128	512 GiB	EBS Only	50000 Megabit

Viewing 714 of 714 available instances

Q c6i.32 X 1 match < 1 >

Instance name ▲	On-Demand hourly rate ▼	vCPU ▼	Memory ▼	Storage ▼	Network performance ▼
c6i.32xlarge	\$5.44	128	256 GiB	EBS Only	50000 Megabit

Viewing 714 of 714 available instances

Q r6i.32 X 1 match < 1 >

Instance name ▲	On-Demand hourly rate ▼	vCPU ▼	Memory ▼	Storage ▼	Network performance ▼
r6i.32xlarge	\$8.064	128	1024 GiB	EBS Only	50000 Megabit

## Lampiran 4 *Security Group* Amazon EC2

### 1. Protokol SSH

#### Aturan-aturan untuk terhubung ke instans dari komputer Anda

Untuk terhubung ke instans Anda, grup keamanan Anda harus memiliki aturan-aturan ke dalam yang mengizinkan akses SSH (untuk instans Linux) atau akses RDP (untuk instans Windows).

Tipe protokol	Nomor protokol	Port	IP sumber
TCP	6	22 (SSH)	Alamat IPv4 publik dari komputer Anda, atau rentang alamat IP di jaringan lokal Anda. Jika VPC Anda diaktifkan untuk IPv6 dan instans Anda memiliki alamat IPv6, maka Anda dapat memasukkan alamat atau rentang alamat IPv6.
TCP	6	3389 (RDP)	Alamat IPv4 publik dari komputer Anda, atau rentang alamat IP di jaringan lokal Anda. Jika VPC Anda diaktifkan untuk IPv6 dan instans Anda memiliki alamat IPv6, maka Anda dapat memasukkan alamat atau rentang alamat IPv6.

### 2. Protokol HTTP

#### Aturan-aturan server web

Aturan-aturan ke dalam berikut mengizinkan akses HTTP dan HTTPS dari alamat IP mana pun. Jika VPC Anda diaktifkan untuk IPv6, maka Anda dapat menambahkan aturan-aturan untuk mengendalikan lalu lintas HTTP dan HTTPS ke dalam dari alamat IPv6.

Tipe protokol	Nomor protokol	Port	IP sumber	Catatan
TCP	6	80 (HTTP)	0.0.0.0/0	Mengizinkan akses HTTP ke dalam dari alamat IPv4 mana pun

### 3. Protokol HTTPS

#### Aturan-aturan server web

Aturan-aturan ke dalam berikut mengizinkan akses HTTP dan HTTPS dari alamat IP mana pun. Jika VPC Anda diaktifkan untuk IPv6, maka Anda dapat menambahkan aturan-aturan untuk mengendalikan lalu lintas HTTP dan HTTPS ke dalam dari alamat IPv6.

Tipe protokol	Nomor protokol	Port	IP sumber	Catatan
TCP	6	80 (HTTP)	0.0.0.0/0	Mengizinkan akses HTTP ke dalam dari alamat IPv4 mana pun
TCP	6	443 (HTTPS)	0.0.0.0/0	Mengizinkan akses HTTPS ke dalam dari alamat IPv4 mana pun

### 4. Protokol Custom ICMP – IPv4

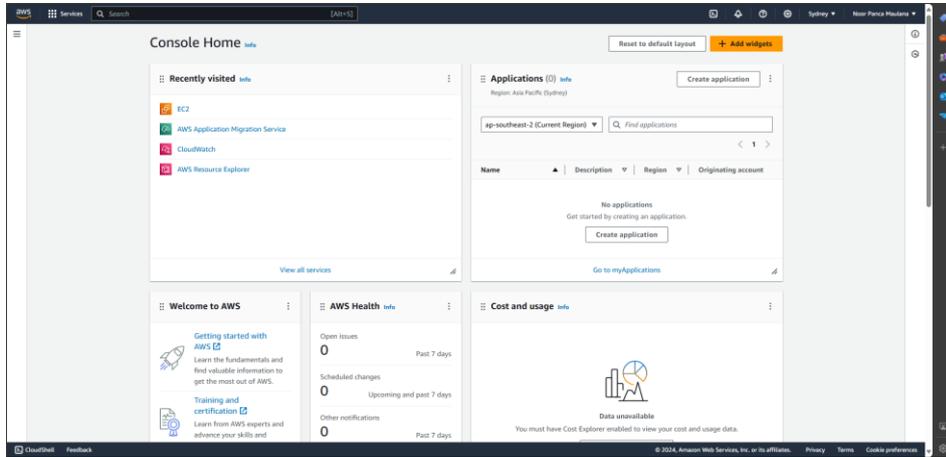
## Aturan-aturan untuk melakukan ping/ICMP

Perintah **ping** merupakan jenis lalu lintas ICMP. Untuk melakukan ping pada instans Anda, Anda harus menambahkan aturan ICMP ke dalam berikut ini.

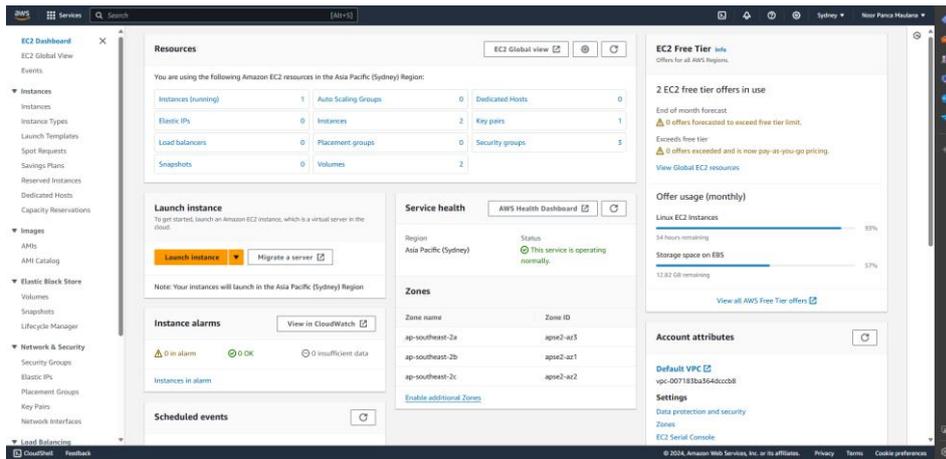
Tipe	Protokol	Sumber
ICMP - IPv4 Kustom	Permintaan Echo	Alamat IPv4 publik dari komputer Anda, alamat IPv4 tertentu, atau alamat IPv4 atau IPv6 dari mana saja.
Semua ICMP - IPv4	IPv4 ICMP (1)	Alamat IPv4 publik dari komputer Anda, alamat IPv4 tertentu, atau alamat IPv4 atau IPv6 dari mana saja.

# Lampiran 5 Perancangan Server Berbasis *Cloud Amazon EC2* Hingga *Cloud Server Aktif*

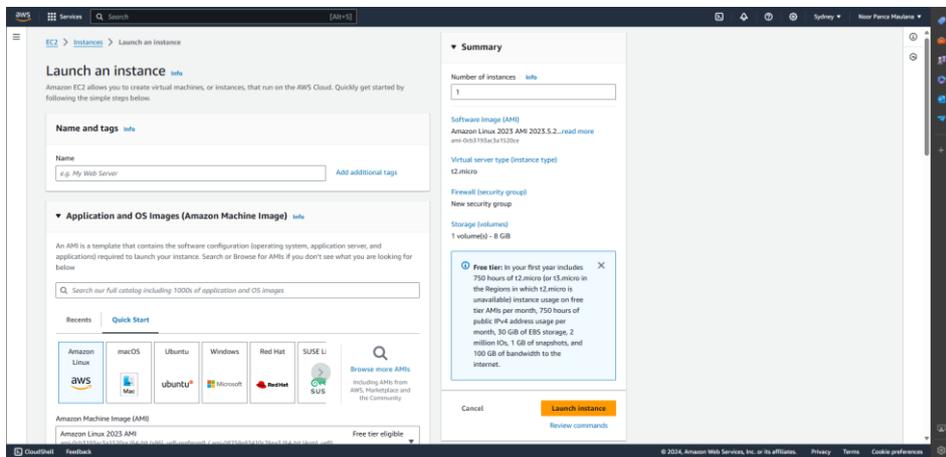
## 1. Halaman *console*



## 2. Dashboard Amazon EC2



## 3. Launch instance



#### 4. Cloud server sudah aktif

