

## **BAB III**

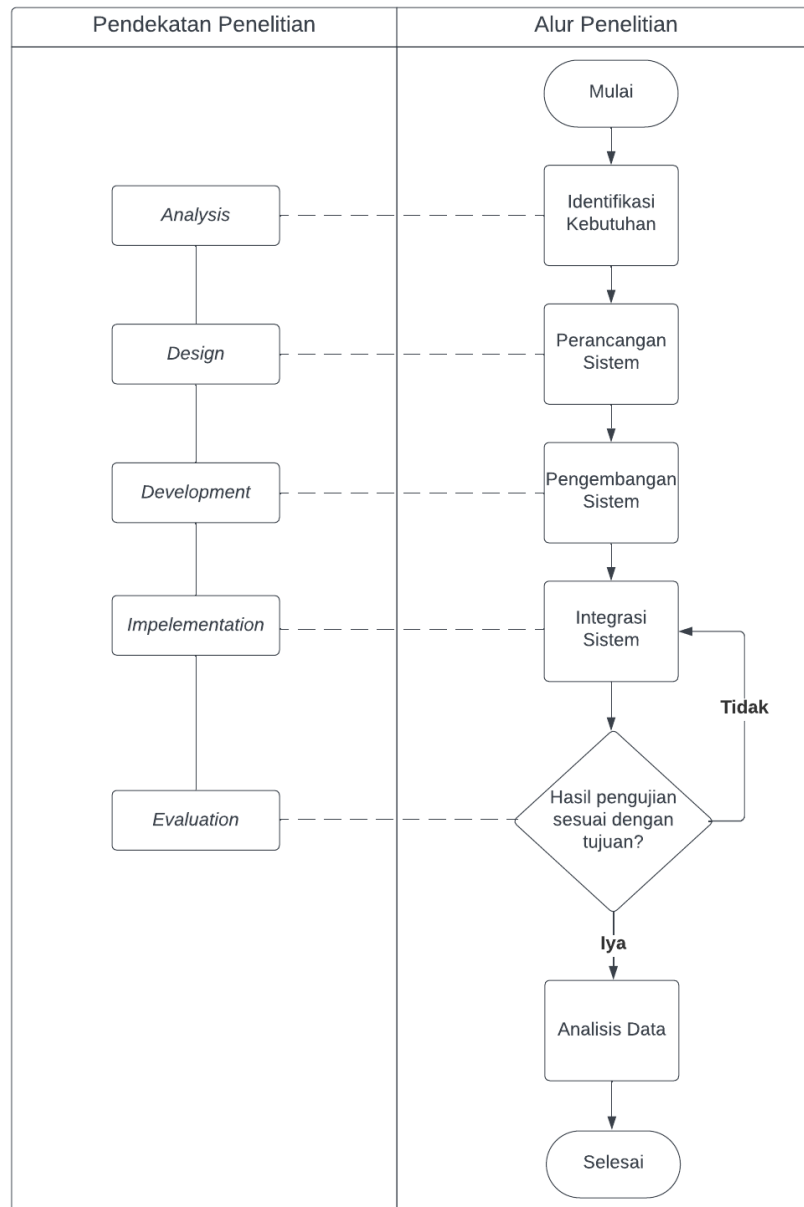
### **METODE PENELITIAN**

#### **3.1 Jenis Penelitian**

Dalam penelitian ini, jenis penelitian yang digunakan dengan model pendekatan *Research and Development* (R&D) yaitu metode penelitian yang difokuskan pada pengembangan produk dengan tujuan menghasilkan suatu produk tertentu dan menguji keefektifannya, serta akan mengevaluasi hasil penelitian yang didapatkan (A. Saputra dkk., 2020). Dengan pendekatan model ADDIE yang merupakan pendekatan sistematis dalam pengembangan media pembelajaran. Tahap pertama ialah menentukan kebutuhan layanan yang akan digunakan lalu dilanjutkan ke tahap desain agar menghasilkan sebuah *output* berupa struktur gambar dan bagan dari pembuatan infrastruktur. Dilanjutkan dengan dengan tahap pengembangan yang menghasilkan *output* berupa pengkonfigurasi setiap layanan yang digunakan dari gambar infrastruktur, kemudian di implementasikan dengan melakukan pengujian terhadap fungsionalitas layanan yang akan saling terintegrasi serta di evaluasi terhadap hasil dari performansi dan rancangan infrastruktur.

#### **3.2 Alur Penelitian**

Penelitian yang digunakan untuk penelitian ini mengacu pada pendekatan ADDIE yang hasilnya akan di uji keefektifannya beserta fungsionalitasnya dalam membangun sistem yang dirancang. Alur dan pendekatan penelitian terdiri dari identifikasi kebutuhan spesifikasi perangkat lunak dan perangkat keras (*Analysis*) seperti menentukan layanan GCP dan *device* yang ingin digunakan untuk memulai penelitian ini dilakukan, kemudian perancangan sistem infrastruktur komputasi awan (*Design*), pengembangan sistem *web server* (*Development*), integrasi layanan komputasi awan pada GCP (*Implementation*), hasil pengujian yang disesuaikan dengan tujuan (*Evaluation*), dan analisis data. Untuk lebih jelasnya dapat dilihat pada *flowchart* yang disajikan pada gambar 3.1 berikut.



Gambar 3.1 Alur dan Pendekatan Penelitian

Gambar 3.1 menjelaskan tahapan dalam melakukan analisis dan implementasi komputasi awan berbasis *web* dengan pemanfaatan *load balancing* dan *auto-scaling* menggunakan GCP yang di bentuk dalam diagram.

Sahat Parulian, 2024

**ANALISIS DAN IMPLEMENTASI INFRASTRUKTUR KOMPUTASI AWAN BERBASIS WEB DENGAN PEMANFAATAN LOAD BALANCING DAN AUTO-SCALING PADA GOOGLE CLOUD PLATFORM**

Universitas Pendidikan Indonesia | repository.upi.edu | Perpustakaan.upi.edu

### 3.2.1 Identifikasi Kebutuhan

Tahapan ini dimulai dengan menentukan spesifikasi dari perangkat keras dan perangkat lunak yang akan digunakan agar penelitian ini dapat berjalan dengan baik. Informasi mengenai perangkat keras yang digunakan dapat dilihat dari Tabel 3.1 berikut.

Tabel 3.1 Spesifikasi perangkat keras

No	Perangkat keras	Spesifikasi
1	<i>Processor</i>	Intel(R) Core (TM) i5-1035G1 CPU @ 1.00GHz, 1190 Mhz, 4 Core(s), 8 Logical <i>Processor</i> (s)
2	RAM	8 GB
3	VGA / GPU	NVIDIA GeForce GTX 330
4	Storage	512 GB SSD
5	Operasi Sistem	Windows 11 Pro 64-bit

Sedangkan untuk perangkat lunak akan menggunakan beberapa *tools* yang diperlukan untuk merancang infrastruktur komputasi awan, diantaranya akun GCP *free trial*, layanan Compute Engine, Cloud DNS, Cloud CDN, Load Balancing, Apache Jmeter dan file repositori *web* statis. Selain itu, adanya penetapan kategori jumlah *traffic* berdasarkan besaran jumlah *users* yang akan di simulasikan menggunakan Apache Jmeter dalam rentang waktu beberapa menit. Besaran jumlah *users* dapat dikategorikan sebagai *traffic* rendah, sedang, dan tinggi. Kategori besaran jumlah tersebut dapat dilihat pada Tabel 3.2

No	Kategori <i>Traffic</i>	Jumlah <i>User</i> (per 5 menit)
1	Rendah	1 - 5000
2	Sedang	6000 – 10000

Sahat Parulian, 2024

**ANALISIS DAN IMPLEMENTASI INFRASTRUKTUR KOMPUTASI AWAN BERBASIS WEB DENGAN PEMANFAATAN LOAD BALANCING DAN AUTO-SCALING PADA GOOGLE CLOUD PLATFORM**

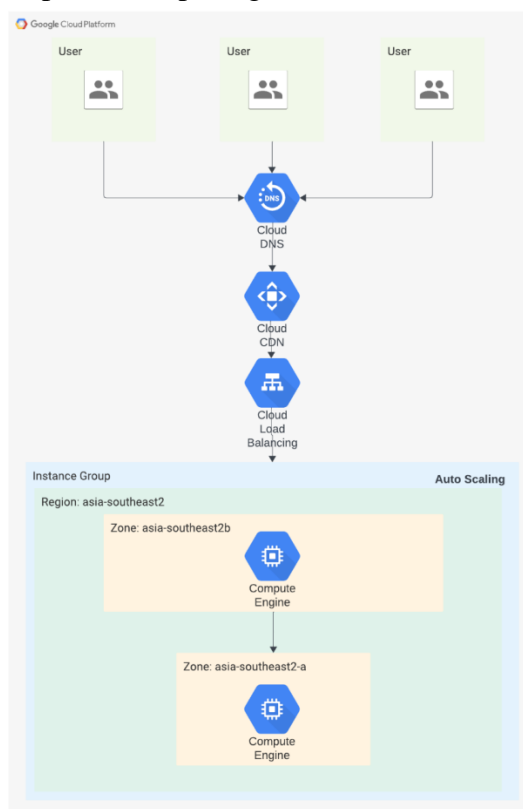
Universitas Pendidikan Indonesia | repository.upi.edu | Perpustakaan.upi.edu

3	Tinggi	> 10000
---	--------	---------

Tabel 3.2 Kategori traffic

### 3.2.2 Perancangan Sistem

Tahap kedua ialah perancangan sistem. Pada tahap ini, proses dimulai dengan pembuatan alur infrastruktur komputasi awan menggunakan sistem teknikal. Hal ini melibatkan penyusunan bagan gambar untuk memvisualisasikan integrasi layanan-layanan yang diperlukan. Pendekatan ini memungkinkan pemahaman yang lebih baik terkait bagaimana setiap komponen berinteraksi dalam infrastruktur komputasi awan pada Google Cloud Platform. Visualisasi infrastruktur tersebut dapat dilihat pada gambar 3.2 berikut



Gambar 3.2 Infrastruktur load balancing dan auto-scaling komputasi awan

Sahat Parulian, 2024

**ANALISIS DAN IMPLEMENTASI INFRASTRUKTUR KOMPUTASI AWAN BERBASIS WEB DENGAN PEMANFAATAN LOAD BALANCING DAN AUTO-SCALING PADA GOOGLE CLOUD PLATFORM**

Universitas Pendidikan Indonesia | repository.upi.edu | Perpustakaan.upi.edu

Dari gambar diatas ketika ada *traffic* dengan jumlah *users* yang begitu banyak maka hal pertama kali yang akan di tuju ialah DNS atau biasa disebut dengan domain *website*, dilanjutkan dengan melihat konten *web* yang disajikan oleh CDN, lalu banyaknya *traffic users* akan masuk ke dalam *instance group* melalui *load balancing*. *Load balancing* ini akan menampung besaran *traffic users* sesuai dengan minimal kapasitas *instance group* yang sudah di konfigurasi, pada penelitian ini *instance group* ditetapkan dengan kapasitas 1 buah mesin virtual. Selanjutnya jika *instance group* telah melewati *threshold* dari *autoscaler utilization (load balancing)* maka secara otomatis mesin virtual baru akan bertambah satu dan *Load balancing* dapat mendistribusikan banyaknya *traffic users* ke mesin virtual tersebut secara merata. Apabila jumlah besaran *traffic users* sudah berkurang dari *threshold* yang sudah ditentukan maka secara otomatis akan mengurangi mesin virtual menjadi jumlah semula. Selain itu, dalam tahap ini akan menggunakan jenis mesin virtual berdasarkan spesifikasi yang disajikan dalam tabel 3.3 berikut.

Tabel 3.3 Spesifikasi untuk sistem

Tipe mesin	RAM	Sistem Operasi	Virtual CPU	Wilayah
N2 <i>custom</i>	2 GB	Ubuntu Server 22.04 LTS	2	asia-southeast-2
N2 <i>custom</i>	4 GB	Ubuntu Server 22.04 LTS	2	asia-southeast-2
N2 <i>custom</i>	8 GB	Ubuntu Server 22.04 LTS	4	asia-southeast-2

### 3.2.3 Pengembangan Sistem

Tahap ketiga ialah pengembangan sistem. Di tahap ini, proses pengembangan dimulai dengan menginstal salah satu jenis *web server* yang akan digunakan, yaitu *nginx*, salah satu perangkat lunak *web server* yang populer dan banyak

Sahat Parulian, 2024

ANALISIS DAN IMPLEMENTASI INFRASTRUKTUR KOMPUTASI AWAN BERBASIS WEB DENGAN PEMANFAATAN LOAD BALANCING DAN AUTO-SCALING PADA GOOGLE CLOUD PLATFORM

Universitas Pendidikan Indonesia | repository.upi.edu | Perpustakaan.upi.edu

digunakan di dunia. Nginx berfungsi menyajikan halaman *web* kepada pengguna yang mengaksesnya melalui protokol HTTP atau HTTPS. *Web server* juga berperan sebagai penyimpan data, termasuk dokumen HTML, gambar, file CSS, dan file JavaScript. Dari segi perangkat lunak, *web server* berfungsi sebagai pusat kontrol untuk memproses permintaan yang diterima dari *browser*. *Web server* memiliki peran penting dalam mengatur semua komunikasi antara *browser* dan *server* untuk memproses suatu *web*. Selanjutnya *web server* akan menampilkan aplikasi statis mengenai informasi seputar *travel* wisata. Selain itu, pada tahap ini, *website* juga akan dihubungkan dengan *domain* bernama *travisionate.my.id*. Penggunaan *domain* bertujuan untuk mempermudah akses ke *web* tanpa perlu memasukkan IP dari situs yang akan diakses. Kemudian akan melakukan perancangan *load balancing* yang dimulai dengan konfigurasi pada bagian *front-end*, seperti pemilihan protokol yang digunakan untuk mengakses *web* dan konfigurasi penggunaan alamat IP. Pada bagian *front-end*, pemilihan protokol mencakup *http* dan *https*. Selanjutnya, konfigurasi bagian *back-end* melibatkan penentuan tipe *back-end* yang digunakan, penetapan *threshold Autoscale utilization (load balancing)*, konfigurasi CDN, dan pemilihan jenis metode *load balancing* yang digunakan.

### 3.2.4 Integrasi Sistem

Tahap keempat ialah implementasi. Pada tahap ini, proses implementasi dilaksanakan dengan mengintegrasikan setiap layanan GCP yang telah dikonfigurasi, seperti *web server*, *load balancing*, *auto-scaling*, dan *database*. Selain itu, pada tahap ini, akan dilakukan pengujian terhadap *load balancing* dan *auto-scaling* dengan memastikan fungsionalitas infrastruktur komputasi awan di GCP dapat berjalan dengan baik.

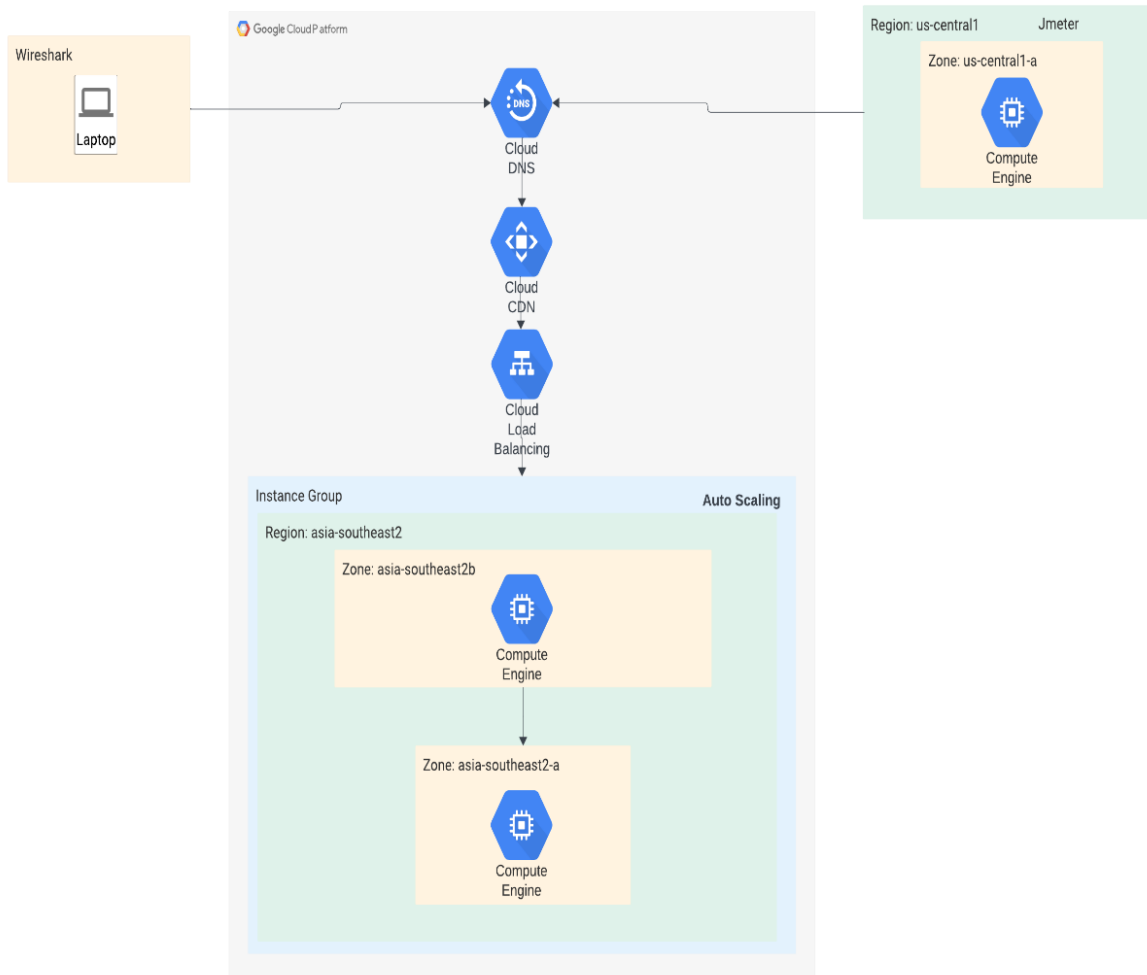
### 3.2.5 Pengujian Sistem

Sahat Parulian, 2024

**ANALISIS DAN IMPLEMENTASI INFRASTRUKTUR KOMPUTASI AWAN BERBASIS WEB DENGAN PEMANFAATAN LOAD BALANCING DAN AUTO-SCALING PADA GOOGLE CLOUD PLATFORM**

Universitas Pendidikan Indonesia | repository.upi.edu | Perpustakaan.upi.edu

Tahap kelima ialah pengujian sistem. Pada tahap ini, dilakukan pengujian terhadap *load balancing* dan *auto-scaling* yang telah terintegrasi dengan *web server*, dengan memberikan *traffic* rendah dan tinggi pada *website* menggunakan Apache Jmeter. Selanjutnya, dilakukan pemantauan terhadap *server* untuk menentukan jumlah mesin virtual yang dihasilkan oleh *auto-scaling*. Selain itu, pada tahap ini, hasil pengujian rancangan infrastruktur komputasi awan dan data yang diperoleh akan dievaluasi untuk memastikan kesesuaian dengan cakupan keseluruhan rancangan dan tujuan penelitian. Data pada rancangan infrastruktur komputasi awan berupa jumlah *traffic* yang masuk kepada *server* melalui pengujian Apache Jmeter dan parameter *delay*, *throughput*, serta *packet loss* yang akan direkam menggunakan Wireshark. Sedangkan data pada performa *load balancing* dan *auto-scaling* berupa jumlah penambahan dan pengurangan mesin virtual baru secara otomatis serta *down-time* saat penambahan dan pengurangan mesin virtual baru. Untuk lebih jelas terkait pengujian sistem dapat dilihat pada Gambar 3.3 berikut.



Gambar 3.3 Pengujian arsitektur jaringan komputasi awan

Dari gambar diatas Apache Jmeter pada *region* us-central1-a akan memberikan *traffic* sesuai dengan kategori yang sudah ditentukan kepada Cloud DNS yang terintegrasi dengan Cloud CDN, lalu di saat yang bersamaan wireshark pada laptop akan merekam parameter *throughput*, *delay* dan *packet loss* dengan mengakses DNS dan CDN yang sudah terintegrasi, kemudian *traffic* akan masuk melewati Cloud *Load Balancing* ke *instance group* yang terdiri dari konfigurasi mesin virtual yang sudah ditetapkan, apabila *traffic* yang diberikan memicu

Sahat Parulian, 2024

**ANALISIS DAN IMPLEMENTASI INFRASTRUKTUR KOMPUTASI AWAN BERBASIS WEB DENGAN PEMANFAATAN LOAD BALANCING DAN AUTO-SCALING PADA GOOGLE CLOUD PLATFORM**

Universitas Pendidikan Indonesia | repository.upi.edu | Perpustakaan.upi.edu



*threshold autoscaler utilization* maka *auto-scaling* akan bertindak untuk membuat mesin virtual baru untuk menangani *traffic* yang masuk.

### **3.2.6 Analisis Data**

Setelah melakukan perancangan dan implementasi sistem lalu peneliti akan melakukan analisis data sesuai dengan parameter yang telah ditentukan untuk mengetahui rancangan infrastruktur komputasi awan, menentukan performansi pada *load balancing* dan *auto-scaling* serta menentukan biaya terhadap mesin virtual yang digunakan selama penelitian.