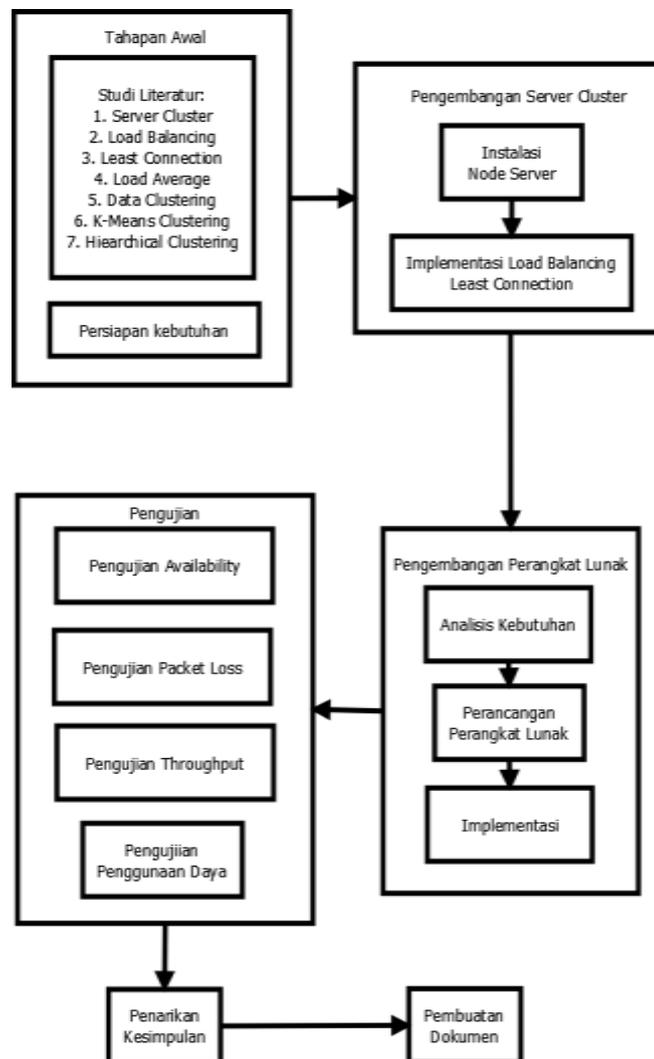


BAB III

METODOLOGI PENELITIAN

3.1 Desain Penelitian

Dalam melaksanakan penelitian dibutuhkan desain penelitian agar memudahkan pembaca untuk melihat alur penelitian yang dilakukan. Desain penelitian adalah tahapan yang dilakukan dalam penelitian untuk memudahkan penyusunan dalam melakukan penelitian. Berikut adalah tahapan desain penelitian yang dilakukan:



Gambar 3.1 Desain Penelitian

Berikut pembahasan gambar 3.1:

1. Tahapan Awal

Ginancar Aji Sudarsono, 2018

PENJADWALAN PEMADAMAN NODE SERVER PADA SERVER CLUSTER BERDASARKAN KLASTERISASI DATA UPTIME

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

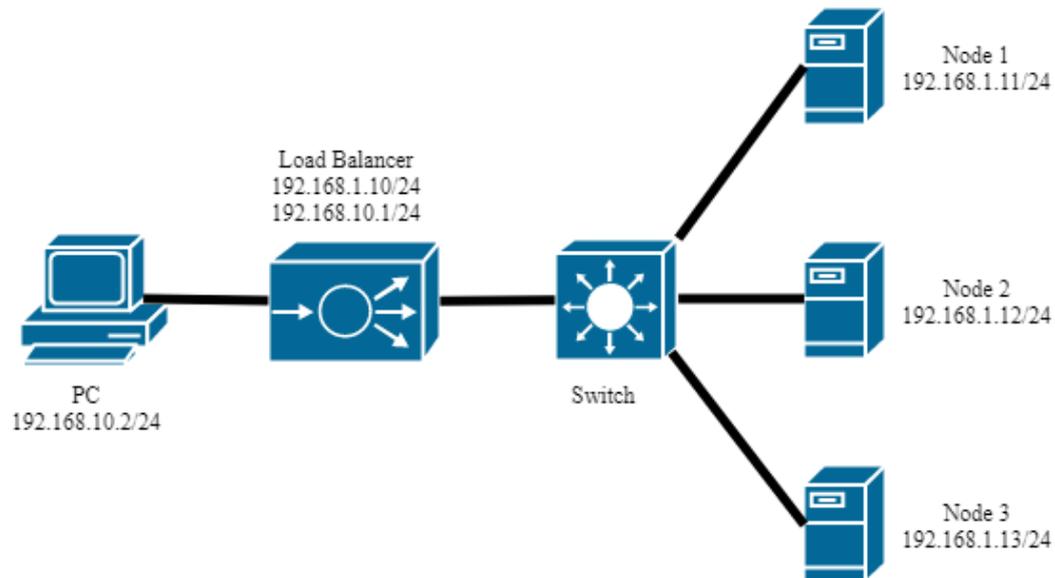
- a. Studi literatur, tahap ini dilakukan kajian mengenai berbagai teori yang mendukung penelitian ini khususnya *server cluster*, *load balancing*, *least connection*, *data clustering*, *load average*, *k-means clustering* dan *hierarchical clustering*.
 - b. Persiapan kebutuhan, tahap ini bertujuan untuk menyiapkan alat dan bahan yang akan digunakan pada penelitian ini
2. Pengembangan *server cluster*
 - a. Instalasi *node server*, tahap ini untuk instalasi perangkat keras dan perangkat lunak *node-node server* yang digunakan pada penelitian ini.
 - b. Implementasi *load balancing least connection*, tahap ini untuk mengimplementasikan algoritma *least connection* pada *load balancer*.
 3. Pengembangan perangkat lunak dilakukan berdasarkan metode sekuensial linear yang terdiri dari tahapan-tahapan *analysis*, *design*, *code* dan *testing*.
 4. Pengujian, pada tahap ini adalah pengambilan data daya listrik, *availability*, *throughput* dan *packet loss* sebelum dan sesudah penggunaan penjadwalan pemadaman
 5. Penarikan kesimpulan
 6. Pembuatan dokumen meliputi pembuatan dokumen skripsi, jurnal dan dokumen teknis.

3.2 Metode Penelitian

3.2.1 Pengembangan *Server Cluster*

Dalam penelitian ini, sistem merupakan empat buah *server* yang terhubung dalam sebuah LAN dimana satu *server* berperan sebagai *load balancer* yang berfungsi untuk menerima *request* dan meneruskannya ke tiga *node server* lainnya di dalam *cluster* untuk diproses. Infrastruktur jaringan yang digunakan diadaptasi dari topologi *star*. Topologi jaringan tersebut dapat dilihat pada gambar 3.2.

Load balancer server memiliki dua *Network Interface Card* (NIC) yang memiliki peran yang berbeda. NIC yang pertama (*eth0*) berfungsi untuk menghubungkan *load balancer* dengan PC. NIC yang kedua (*eth1*) berfungsi untuk menghubungkan *load balancer* dengan *node* satu, dua dan tiga. *Load balancer* menggunakan mekanisme *least connection*. Mekanisme memberikan beban kepada *node server* yang tidak sibuk sehingga meminimalisir *overload*.



Gambar 3.2 Topologi Server Cluster

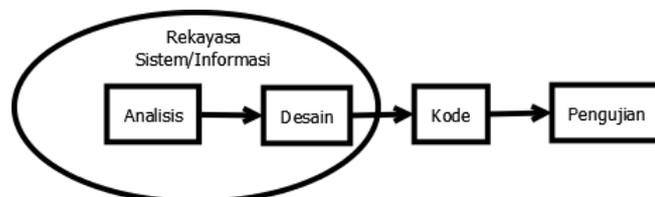
Tabel 3.1 Konfigurasi Server

Nama	IP address	Netmask	Gateway
<i>Load Balancer (Eth0)</i>	192.168.10.1	255.255.255.0	-
<i>Load Balancer (Eth1)</i>	192.168.1.10	255.255.255.0	192.168.10.1
<i>Node 1</i>	192.168.1.11	255.255.255.0	192.168.1.10
<i>Node 2</i>	192.168.1.12	255.255.255.0	192.168.1.10
<i>Node 3</i>	192.168.1.13	255.255.255.0	192.168.1.10

Adapun rincian konfigurasi *interface* dari masing-masing *server* dijelaskan pada tabel 3.1. Semua *request* dari *internet* atau *user* akan diterima oleh *load balancer* melalui *eth0*, kemudian oleh *eth0* akan diteruskan ke *node-node* di dalam *cluster* melalui *eth1*. *Request* tersebut diproses oleh *node*, kemudian setelah proses selesai, *node* memberikan respon kepada *load balancer* melalui *eth1*. *Eth1* meneruskan respon ke *eth0* dan *eth0* meneruskan respon ke *user*.

3.2.2 Pengembangan Perangkat Lunak

Pengembangan perangkat lunak menggunakan *The Sequential Linear Model* (Model Sekuensial Linier). Gambar 3.3 menggambarkan model sekuensial linier untuk rekayasa perangkat lunak. Model sekuensial linier meliputi tahapan-tahapan sebagai berikut (Pressman, 2001):



Gambar 3.3 Model *The Sequential Linear* (Pressman, 2001)

1. Analisis Kebutuhan Perangkat Lunak

Untuk memahami sifat program yang dibangun, maka perlu dipahami domain informasi untuk perangkat lunak, serta diperlukan fungsi, perilaku, kinerja dan antarmuka (*interface*) yang diperlukan

2. Desain

Proses ini berfokus pada struktur data, arsitektur perangkat lunak, representasi antarmuka dan algoritma atau perincian prosedur

3. Penerjemahan Kode (*Code Generation*)

Desain harus diterjemahkan ke dalam bentuk yang dapat dibaca oleh mesin.

4. Pengujian (*Testing*)

Proses ini bertujuan untuk memastikan bahwa semua pernyataan sudah diuji, untuk mengungkapkan kesalahan dan untuk memastikan bahwa *input* yang didefinisikan akan menghasilkan hasil yang sesuai dengan yang dibutuhkan.

3.2.2.1 Analisis Perangkat Lunak

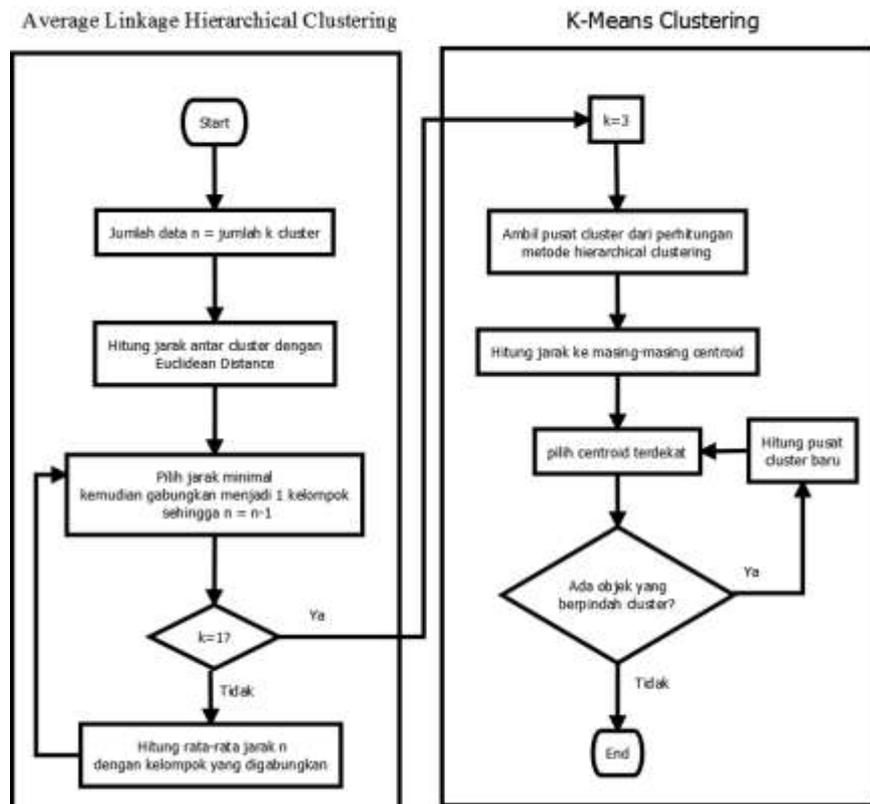
1. Analisis Kebutuhan

Perangkat lunak yang dibuat merupakan sebuah perangkat lunak yang memberikan fungsi pemadaman *node server* dengan tujuan bisa menghemat penggunaan daya listrik. Pengembangan perangkat lunak ini membutuhkan beberapa dependensi berupa perangkat keras, perangkat lunak maupun *library*. Pemadaman *server* dapat terwujud dengan menerapkan gabungan metode *average linkage hierarchical clustering* dan *K-means clustering* pada *uptime log* dari setiap

node server cluster. Data *uptime* tersebut terdiri atas durasi *uptime*, jumlah *user*, *load average* pada satu menit, lima menit dan 15 menit terakhir. Dari hasil *clustering*, dapat ditentukan *load* tertinggi dan terendah beserta waktu dan durasi pemadaman dari masing-masing *node*. Masing-masing *node* akan dipadamkan sesuai waktu dan durasi pemadaman pada keesokan harinya.

2. Analisis Input

Input yang dibutuhkan dalam perangkat lunak ini berupa data *uptime* dari masing-masing anggota *server cluster* dan rentang tanggal. Data *uptime* tersebut ditangkap secara *realtime* dari *server* yang digunakan. Data tersebut berupa sebuah *string* yang harus diolah terlebih dahulu sebelum dimasukkan ke perangkat lunak. *Uptime* tersebut terdiri dari “<current tume> <uptime duration>, <currently logged in user(s)>, <load average for the past 1,5,15 minutes>”. Data tersebut kemudian dipecah, sehingga yang diambil hanya *load average* dalam satu, lima dan 15 menit terakhir serta *current time*. *User* memasukan rentang tanggal yang akan diolah oleh perangkat lunak sehingga dapat memberikan *output*.



Gambar 3.4 Gabungan *Average Linkage Hierarchical Clustering* dan *K-Means Clustering*

3. Analisis Output

Keluaran dari perangkat lunak ini adalah berupa jam pemadaman dan durasi terlama dari masing-masing *node server* yang diolah menggunakan gabungan *average linkage hierarchical clustering* dan *K-Means clustering* seperti yang ditunjukkan gambar 3.4. Data dari hasil *clustering* tersebut digunakan untuk menjadwalkan pemadaman *node server*.

3.2.2.2 Rancangan Perangkat Lunak

Rancangan perangkat lunak ini berisi *input* berupa data *uptime* dari *node*, proses yang ada pada sistem yaitu *getlog*, *powefficiency* dan *shutdown*. Berikut penjelasan dari masing-masing proses:

a. Getlog

Pada tahap ini dilakukan pengambilan data *uptime* setiap menit dan pengolahan data *uptime* agar menjadi format *input* yang dibutuhkan oleh perangkat lunak. Hasil pengolahan tersebut kemudian disimpan didalam *log*.

b. Powefficiency

Pada tahap ini, data *uptime* yang telah melalui proses *getlog* akan diambil berdasarkan *node* dibagi ke dalam *cluster* dengan menggunakan *average linkage hierarchical clustering* dan *K-Means clustering*. Data *cluster* yang digunakan adalah *uptime* yang berisi *date*, *load average* satu, lima dan 15 menit. Proses *clustering* dikelompokkan ke dalam tiga buah *cluster* yaitu :

1. Cluster rendah

Cluster ini diisi oleh data dengan nilai ketiga *load average* yang rendah

2. Cluster sedang

Cluster ini diisi oleh data dengan nilai ketiga *load average* yang sedang

3. Cluster tinggi

Cluster ini diisi oleh data dengan nilai ketiga *load average* yang tinggi

Setelah proses *clustering*, data akan diurutkan berdasarkan *cluster*, tanggal dan jam untuk diolah sehingga mendapatkan waktu pemadaman dan penyalaan.

c. Shutdown

Pada tahap ini dilakukan proses pemadaman. *Node* akan dipadamkan dan dinyalakan pada jam yang sama dengan hasil dari proses *powefficiency*.

3.2.3 Pengujian

Sebelum pemadaman dilaksanakan, daya yang dipakai diukur untuk dijadikan bahan perbandingan dengan pengukuran setelah pemadaman. Skenario pengujian sebelum dan sesudah pemadaman sebagai berikut:

1. *Server cluster* diberi *load* konkuren sebesar 180 pada jam 00.00-07.59
2. *Server cluster* diberi *load* konkuren sebesar 360 pada jam 08.00-15.59
3. *Server cluster* diberi *load* konkuren sebesar 540 pada jam 16.00-23.59

3.2.3.1 Pengujian Kualitas Server

Pengujian kualitas *server* meliputi *availability*, *throughput* dan *packet loss* menggunakan perangkat lunak Siege dilakukan saat skenario pengujian dijalankan.

3.2.3.2 Pengujian Daya

Pengujian daya dilakukan dengan menggunakan *energy meter* dilakukan saat skenario pengujian dijalankan. *energy meter* adalah alat untuk mengukur Watt, Ampere, Voltase dan biaya yang dikeluarkan oleh alat yang menggunakan energi listrik. Berikut gambar 3.5 adalah *energy meter*:



Gambar 3.5 *Energy Meter*

3.3 Alat dan Bahan Penelitian

3.3.1 Alat Penelitian

Alat penelitian yang digunakan adalah terbagi menjadi dua tahap yang dijelaskan pada tabel 3.2 untuk tahap pengembangan dan tabel 3.3 untuk tahap implementasi sebagai berikut:

Tabel 3.2 Alat Penelitian Tahap Pengembangan

Perangkat Keras		Perangkat Lunak
Item	Spesifikasi	
1. Laptop	Processor Intel® Core™ i3 2310M	1. Ubuntu 16.04 LTS
	RAM 4GB	2. Python
	HDD 500GB	3. Sublime Text 3
	Monitor 14" dan 17"	

Tabel 3.3 Alat Penelitian Tahap Implementasi

Perangkat Keras			Perangkat Lunak
Item	Keterangan	Spesifikasi	
1. Laptop	<i>User</i>	Processor Intel® Core™ i3 2310M	1. Ubuntu 16.04 LTS
		RAM 4GB	2. Linux Mint
		HDD 500GB	3. Python
		Monitor 14"	4. Sublime Text 3
2. PC	<i>Load Balancer</i>	Processor Intel Pentium E2200	5. NGINX
		RAM 2GB	6. Mysql
		HDD 640GB	7. PHP
		Monitor 17"	8. Siege
3. Laptop	<i>Node 1</i>	Processor Intel Pentium E2200	9. OpenSSH
		RAM 4GB	10. Putty
		HDD 320GB	
		Monitor 14"	
4. Laptop	<i>Node 2</i>	Processor Intel® Core™ i3 2310M	
		RAM 4GB	
		HDD 320GB	
		Monitor 14"	
5. Laptop	<i>Node 3</i>	Processor AMD E1-2100	

Perangkat Keras			Perangkat Lunak
Item	Keterangan	Spesifikasi	
		RAM 4GB	
		HDD 500GB	
		Monitor 11.6"	
6. Switch D-Link			
7. Kabel <i>Cross</i>			
8. Kabel <i>Straight</i>			
9. Energy Meter			

3.3.2 Bahan Penelitian

Adapun bahan penelitian yang digunakan diambil dari data *uptime* setiap *node* pada *server cluster* pada 2-3 Mei 2018.