

## **BAB 4**

### **REKAYASA KEBUTUHAN**

#### **4.1 Gambaran Umum Sistem**

Sistem yang dibangun dalam penelitian ini digunakan untuk melakukan *routing* pada jaringan *software defined network* berdasarkan *cost* setiap *link* dalam suatu *path*. Jalur yang akan digunakan sebagai media pengiriman data adalah *path* dengan *cost* paling kecil. Sistem ini membutuhkan beberapa spesifikasi kebutuhan agar sistem dapat berjalan sesuai dengan yang diharapkan.

##### **4.1.1 Tujuan**

Tujuan dari sistem ini adalah untuk menerapkan *Yen algorithm routing* pada jaringan *software defined network*. algoritme ini digunakan untuk menemukan beberapa jalur dalam suatu topologi jaringan. Sistem ini akan dibangun pada lingkup jaringan *emulator mininet* dengan menggunakan *ryu controller*.

##### **4.1.2 Perspektif Sistem**

Sistem ini dikatakan bekerja sesuai dengan kebutuhan jika sistem mampu mendapatkan sebanyak *K-Path* yang memiliki *cost* paling kecil sesuai dengan yang dibutuhkan dari topologi jaringan SDN, dan sistem mampu menggunakan semua jalur yang telah ditemukan sebagai media pengiriman data.

##### **4.1.3 Persyaratan Hardware**

Untuk membangun sistem ini agar dapat berjalan dengan baik sesuai tujuan awal, maka dibutuhkan *hardware* pendukung. *Hardware* pendukung yang dibutuhkan sistem ini adalah Laptop. Laptop digunakan sebagai tempat *instalasi* semua *software* yang butuhkan dalam pembangunan sistem. *Hardware* yang digunakan dalam penelitian ini harus menggunakan sistem operasi *ubuntu* dan mampu untuk *instalasi emulator mininet*.

##### **4.1.4 Persyaratan Software**

Untuk membangun sistem ini dibutuhkan beberapa *software* pendukung agar sistem yang dibangun dapat bekerja sesuai dengan tujuan. *software* pendukung tersebut antara lain :

1. *Ubuntu Linux 16.04 64-bit*

2. *Mininet*
3. *Controller Ryu*
4. *Iperf, ping*
5. *Openflow 1.3*
6. *OpenvSwitch 2.5.2*

#### **4.1.5 Lingkungan Operasi Sistem**

Sistem ini cocok digunakan di lingkungan jaringan dengan aktifitas *traffic* yang besar setiap saat. Dengan menggunakan sistem ini maka dapat dikatakan setiap komunikasi data dalam jaringan akan mempunyai jalur komunikasi yang berbeda-beda dan kemacetan dalam jaringan akan dapat diminimalisir.

#### **4.1.6 Batasan Sistem**

Batasan-batasan pada sistem ini adalah :

1. Sistem hanya diterapkan didalam lingkup jaringan *emulasi mininet*.
2. Sistem hanya menggunakan *Controller Ryu*

#### **4.2 Kebutuhan Sistem**

Pada bagian ini dijelaskan semua kebutuhan sistem yang akan digunakan dalam penelitian yang akan dikerjakan agar sistem dapat berjalan sesuai dengan yang diinginkan. Kebutuhan sistem diklasifikasikan menjadi dua bagian yaitu kebutuhan fungsional dan kebutuhan non-fungsional. Kebutuhan sistem dapat mempermudah melalui proses perancangan dan implementasi.

##### **4.2.1 Kebutuhan Fungsional**

Kebutuhan fungsional merupakan kebutuhan apa saja yang harus dikerjakan sistem yang dibuat, informasi semua hal yang harus ada dan bisa dihasilkan oleh sistem. Kebutuhan fungsional dari sistem ini adalah sebagai berikut:

1. Sistem dapat menemukan beberapa jalur untuk mencapai tujuan pada suatu topologi jaringan.

Sistem akan melakukan *routing* untuk mencari setiap jalur yang dapat dilalui oleh data dari *source* ke *destination*. Semua jalur yang ditemukan tersebut akan dilakukan *sorting* untuk menemukan *K-Path* yang paling pendek.

2. Sistem dapat memilih *path* yang memiliki *cost* paling kecil pada jaringan.

Sistem ini akan melakukan *sorting* terhadap semua *path* yang ditemukan berdasarkan *cost* yang paling kecil. Jalur yang dipilih untuk pengiriman data adalah jalur yang *cost* yang paling sedikit.

3. Sistem dapat menggunakan semua jalur yang telah dipilih untuk mengirimkan data.

Sistem ini akan menggunakan semua jalur yang telah dipilih berdasarkan *cost* paling kecil untuk mengirim data dari sumber ke tujuan. Jadi sistem ini tidak hanya menggunakan satu jalur saja untuk mengirimkan data melainkan menggunakan sebanyak *K-Path* yang telah dipilih untuk masing-masing *host* yang akan mengirimkan data.

#### **4.2.2 Kebutuhan Non-Fungsional**

Kebutuhan non-fungsional merupakan kebutuhan perangkat keras maupun perangkat lunak untuk mendukung pembangunan sistem pada penelitian ini. Berikut ini adalah kebutuhan non-fungsional dari sistem ini.

##### **A. Kebutuhan Perangkat Keras**

Kebutuhan Perangkat keras yaitu spesifikasi komputer yang digunakan untuk implementasi sistem.

Komputer/laptop dengan spesifikasi:

- CPU: Intel Core i5.

Minimal Prosesor yang digunakan agar dapat menjalankan sistem ini dengan baik adalah Intel core i5.

- RAM: 4096 MB.

Minimal RAM yang digunakan agar dapat menjalankan sistem ini dengan baik adalah 4 GB. RAM digunakan sebagai memori dari proses yang sedang berjalan semakin besar RAM dari komputer maka performa sistem akan semakin baik.

- Hard disk: 500 GB.

Hard disk diperlukan sebagai memory yang digunakan oleh Sistem Operasi sebagai media penyimpanan.

- Laptop Asus-A455L
- Monitor 14 Inchi

## **B. Kebutuhan Perangkat Lunak**

Kebutuhan Perangkat lunak yaitu seluruh perangkat lunak yang digunakan untuk implementasi sistem.

- *Ubuntu Linux 16.04 64-bit*

*Ubuntu Linux 16.04 64 bit* digunakan sebagai sistem operasi untuk menjalankan sistem. Sistem Operasi ini menyediakan terminal yang mendukung pengembangan sistem yang akan dibangun menjadi lebih mudah.

- *Mininet*

*Mininet* merupakan *emulator* jaringan *open source* yang mendukung protokol *openflow*. *Mininet* ini diperlukan untuk mendukung pembangunan sistem yang berbasis *openflow*. *Mininet* juga diperlukan untuk menjalankan program *controller*.

- *Miniedit*

*Miniedit* merupakan Gui dari *mininet* yang digunakan sebagai media untuk membangun topologi jaringan. *Miniedit* ini sangat membantu dalam pengembangan sistem karena dengan menggunakan ini dapat membangun topologi dengan mudah.

- *Mininam*

*Mininam* digunakan sebagai media untuk melihat animasi gerak jaringan.

- *Ryu Controller*

*Ryu Cotroller* merupakan sebuah *controller* yang berbasis bahasa Python. Sistem ini menggunakan *Ryu controller* karena *ryu controller* mendukung *openflow 1.3*.

- *Iperf, Ping*

*Iperf* dan Ping merupakan sebuah aplikasi yang digunakan untuk mengetahui kinerja dari sebuah sistem jaringan. *Iperf* dan ping didalam sistem ini berfungsi untuk menunjukkan performa dari sistem apakah sudah baik atau tidak.

- *OpenvSwitch 2.5.2*: sebagai virtual *OpenFlow switch*.
- Openflow 1.3