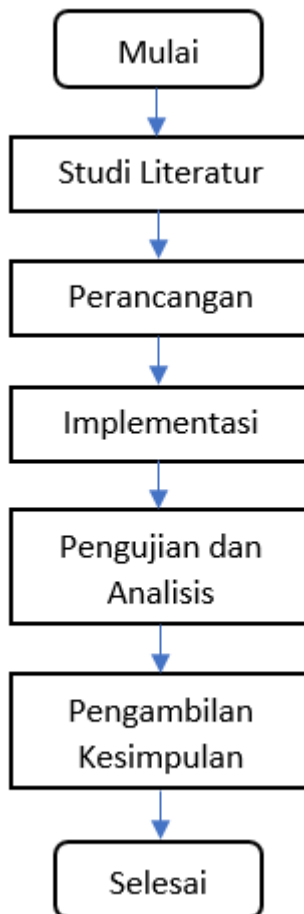


## BAB 3 METODOLOGI

Dalam bab ini akan dibahas mengenai metodologi yang digunakan dalam penelitian analisis perbandingan performansi kontroler Floodlight, kontroler Maestro, kontroler RYU, kontroler POX, kontroler ONOS dalam arsitektur *Software Defined Network* (SDN). Langkah-langkah yang akan dilakukan pada penelitian ini adalah sebagai berikut :



**Gambar 3.1 Flowchart Metodologi**

**Sumber:** penulis

### 3.1 Studi Literatur

Pada bagian studi literatur dibahas mengenai dasar teori dan penelitian-penelitian yang terkait serta relevan terhadap penelitian yang dilakukan oleh penulis. Penelitian yang digunakan sebagai rujukan oleh penulis terkait dengan perbandingan kontroler *Software Defined Network* (SDN). Dasar teori yang digunakan dalam penelitian ini terdiri dari *Software Defined Network* (SDN), *Openflow*, kontroler Floodlight, kontroler Maestro, kontroler RYU, kontroler POX, kontroler ONOX, Mininet, dan Cbench.

## 3.2 Perancangan

Perancangan merupakan tahap pembuatan rancangan sistem yang akan digunakan sebagai dasar pembuatan sistem. Pada skripsi ini, tahap perancangan dibagi menjadi dua bagian, yaitu analisis kebutuhan dan perancangan sistem.

### 3.2.1 Analisis Kebutuhan

Pada tahap ini dijelaskan mengenai kebutuhan yang digunakan untuk menunjang penelitian terkait dengan perbandingan performansi kontroler *Software Defined Network* (SDN).

Sedangkan kebutuhan lain terkait dengan kontroler *Software Defined Network* (SDN) yang dibandingkan yaitu Floodlight, Maestro, RYU, POX, dan ONOX. Setiap kontroler yang dibandingkan tersebut diimplementasikan pada lingkungan pengujian sistem. Untuk dapat diketahui performansi yang dihasilkan dari masing-masing kontroler.

### 3.2.2 Perancangan Lingkungan Pengujian Sistem

Perancangan lingkungan pengujian sistem merupakan tahap yang bertujuan untuk membuat rancangan sistem yang akan digunakan sebagai acuan dalam melakukan implementasi sehingga dapat digunakan untuk pengujian. Tahap perancangan lingkungan pengujian didasarkan pada daftar kebutuhan yang didapat dari tahap analisis kebutuhan.

## 3.3 Implementasi

Pada tahap ini dijelaskan mengenai proses implementasi dari rancangan lingkungan pengujian yang digunakan. Selain itu juga dilakukan implementasi terhadap masing-masing kontroler yang dibandingkan ke dalam lingkungan pengujian. Langkah implementasi yang dilakukan meliputi sebagai berikut:

1. Penerapan topologi sesuai perancangan
2. Instalasi kontroler yang akan diuji
3. Instalasi simulator dan peralatan pendukung
4. Penentuan jumlah *switch* dan *host* untuk pengujian
5. Menjalankan simulasi sesuai dengan skenario

## 3.4 Pengujian dan Analisis

Pada tahap ini akan dilakukan pengujian dan analisis terhadap performansi parameter *throughput* dan parameter *latency* dari masing-masing kontroler yang diuji. Pada penelitian ini terdapat beberapa skenario untuk menguji performansi kontroler dan menganalisis hasil yang diperoleh dari simulasi dengan parameter yang meliputi rata-rata *throughput* dan *latency*. Pengujian yang akan dilakukan terdiri dari beberapa skenario pengujian.

### 3.4.1 Pengujian Throughput Dengan Jumlah Switch dan Host bervariasi

Pengujian *throughput* dilakukan untuk mengetahui banyaknya *flow* yang mampu direspon oleh kontroler. Pengujian dilakukan sebanyak enam kali pengujian dengan jumlah *switch* dan *host* bervariasi pada setiap percobaan.

**Tabel 3.1** List Skenario Pengujian Throughput

Skenario	Deskripsi
1	Diberikan 20 <i>Switch</i> dan setiap <i>Switch</i> terkoneksi dengan 400 <i>Host</i>
2	Diberikan 40 <i>Switch</i> dan setiap <i>Switch</i> terkoneksi dengan 800 <i>Host</i>
3	Diberikan 60 <i>Switch</i> dan setiap <i>Switch</i> terkoneksi dengan 1600 <i>Host</i>
4	Diberikan 80 <i>Switch</i> dan setiap <i>Switch</i> terkoneksi dengan 3200 <i>Host</i>
5	Diberikan 100 <i>Switch</i> dan setiap <i>Switch</i> terkoneksi dengan 6400 <i>Host</i>
6	Diberikan 120 <i>Switch</i> dan setiap <i>Switch</i> terkoneksi dengan 12800 <i>Host</i>

### 3.4.2 Pengujian Throughput Dengan Jumlah Switch bervariasi

Pengujian *throughput* dilakukan untuk mengetahui banyaknya *flow* yang mampu direspon oleh kontroler. Pengujian dilakukan sebanyak enam kali pengujian dengan jumlah *switch* bervariasi pada setiap percobaan.

**Tabel 3.2** List Skenario Pengujian Throughput

Skenario	Deskripsi
1	Diberikan 20 <i>Switch</i> dan setiap <i>Switch</i> terkoneksi dengan 12800 <i>Host</i>
2	Diberikan 40 <i>Switch</i> dan setiap <i>Switch</i> terkoneksi dengan 12800 <i>Host</i>
3	Diberikan 60 <i>Switch</i> dan setiap <i>Switch</i> terkoneksi dengan 12800 <i>Host</i>
4	Diberikan 80 <i>Switch</i> dan setiap <i>Switch</i> terkoneksi dengan 12800 <i>Host</i>
5	Diberikan 100 <i>Switch</i> dan setiap <i>Switch</i> terkoneksi dengan 12800 <i>Host</i>
6	Diberikan 120 <i>Switch</i> dan setiap <i>Switch</i> terkoneksi dengan 12800 <i>Host</i>

### 3.4.3 Pengujian Latency Dengan Jumlah Switch dan Host bervariasi

Pengujian *latency* dilakukan untuk mengetahui banyak waktu yang dibutuhkan kontroler untuk memberikan respon dalam setiap detiknya.

**Tabel 3.3** List Skenario Pengujian Throughput

Skenario	Deskripsi
1	Diberikan 20 <i>Switch</i> dan setiap <i>Switch</i> terkoneksi dengan 400 <i>Host</i>
2	Diberikan 40 <i>Switch</i> dan setiap <i>Switch</i> terkoneksi dengan 800 <i>Host</i>
3	Diberikan 60 <i>Switch</i> dan setiap <i>Switch</i> terkoneksi dengan 1600 <i>Host</i>
4	Diberikan 80 <i>Switch</i> dan setiap <i>Switch</i> terkoneksi dengan 3200 <i>Host</i>
5	Diberikan 100 <i>Switch</i> dan setiap <i>Switch</i> terkoneksi dengan 6400 <i>Host</i>
6	Diberikan 120 <i>Switch</i> dan setiap <i>Switch</i> terkoneksi dengan 12800 <i>Host</i>

### 3.4.4 Pengujian Latency dengan jumlah Switch bervariasi

Pengujian *latency* dilakukan untuk mengetahui banyak waktu yang dibutuhkan kontroler untuk memberikan respon.

**Tabel 3.4** List Skenario Pengujian *Throughput*

Skenario	Deskripsi
1	Diberikan 20 <i>Switch</i> dan setiap <i>Switch</i> terkoneksi dengan 12800 <i>Host</i>
2	Diberikan 40 <i>Switch</i> dan setiap <i>Switch</i> terkoneksi dengan 12800 <i>Host</i>
3	Diberikan 60 <i>Switch</i> dan setiap <i>Switch</i> terkoneksi dengan 12800 <i>Host</i>
4	Diberikan 80 <i>Switch</i> dan setiap <i>Switch</i> terkoneksi dengan 12800 <i>Host</i>
5	Diberikan 100 <i>Switch</i> dan setiap <i>Switch</i> terkoneksi dengan 12800 <i>Host</i>
6	Diberikan 120 <i>Switch</i> dan setiap <i>Switch</i> terkoneksi dengan 12800 <i>Host</i>

## 3.5 Pengambilan Kesimpulan

Tahap kesimpulan dilakukan setelah seluruh tahap-tahap penelitian selesai dilakukan. Pada tahap ini dijelaskan mengenai kesimpulan yang diperoleh dari hasil perbandingan performansi pada kontroler Floodlight, kontroler Maestro, kontroler RYU, kontroler POX, dan kontroler ONOX, berdasarkan skenario yang digunakan. Selain itu juga akan dijelaskan mengenai saran tentan penelitian lebih lanjut.