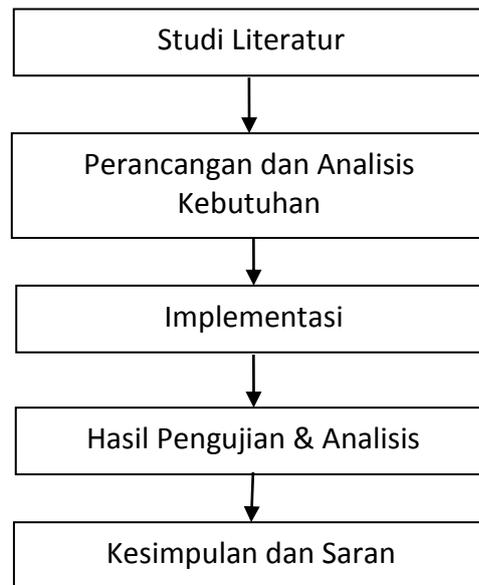


## BAB 3 METODOLOGI DAN PERANCANGAN

Dalam bab ini akan dibahas metodologi yang digunakan dalam penelitian Analisis performansi file sharing menggunakan protokol JXTA dan Gnutella. Tahapan – tahapan yang akan dilakukan dalam penelitian ini adalah sebagai berikut :



**Gambar 3.1 Tahapan Penelitian**

### 3.1 Studi Literatur

Studi literatur dilakukan dengan tujuan mempelajari konsep-konsep yang dibutuhkan dalam membangun sistem ini. Literatur yang dipelajari di antaranya adalah jurnal ilmiah, buku dan berbagai informasi dari internet baik artikel maupun dokumentasi. Literatur yang dipelajari berisikan pemahaman konsep *Peer-to-Peer*, protokol yang digunakan untuk melakukan *file sharing* yaitu JXTA dan Gnutella, serta panduan penggunaan setiap protokol dalam pembuatan perangkat lunak. Pada tahap ini juga masalah yang dibahas didefinisikan agar pokok bahasan yang diangkat lebih mudah dipahami.

### 3.2 Perancangan

Perancangan merupakan tahap pembuatan rancangan sistem yang akan digunakan untuk melakukan implementasi. Pada penelitian ini, tahap perancangan dibagi menjadi dua tahapan, yaitu analisis kebutuhan dan perancangan arsitektur.

Tahap analisis kebutuhan pada penelitian ini dibagi menjadi dua yaitu kebutuhan fungsional dan non-fungsional. Didalam kebutuhan fungsional akan

dijelaskan bagaimana kemampuan sistem yang akan digunakan sehingga implementasi selanjutnya dapat dijalankan. Sedangkan pada kebutuhan non-fungsional akan menjelaskan kebutuhan apa saja yang diperlukan oleh sistem agar bekerja dengan baik sesuai dengan perancangan yang akan dibuat.

### **3.2.1 Analisis Kebutuhan**

Analisis kebutuhan didapatkan dari pembahasan studi literatur yang telah dilakukan. Analisis kebutuhan dilakukan untuk menentukan semua kebutuhan yang diperlukan dalam melakukan penelitian ini yaitu melakukan analisis performansi terhadap *file sharing* menggunakan JXTA dan Gnutella. Analisis kebutuhan berisikan penjelasan tentang hal-hal yang dibutuhkan dalam penelitian sehingga langkah selanjutnya yaitu implementasi dan pengujian dapat berjalan yang dibagi menjadi dua yaitu kebutuhan fungsional dan non-fungsional. Kebutuhan-kebutuhan yang dihasilkan akan dijadikan acuan dalam melakukan penelitian dimulai dari perancangan hingga pengujian.

#### **3.2.1.1 Kebutuhan Fungsional**

Analisa kebutuhan fungsional dilakukan untuk memberikan gambaran mengenai kemampuan dari suatu sistem sehingga implementasi dan pengujian yang akan dilakukan selanjutnya dapat dilakukan. Berikut adalah kebutuhan fungsional suatu sistem:

- Sistem dapat melakukan mekanisme distribusi konten data dengan metode *peer-to-peer* melalui aplikasi *file sharing* yang digunakan.
- Kedua aplikasi *file sharing* dapat menghasilkan nilai yang dapat diukur sebagai parameter perbandingan.

#### **3.2.1.2 Kebutuhan non-fungsional**

Kebutuhan non-fungsional merupakan kebutuhan yang menunjang sistem agar dapat bekerja dengan baik dan dapat berjalan sesuai dengan perancangan yang akan dibuat. Perangkat yang dibutuhkan adalah sebagai berikut :

Perangkat Keras:

- 1 unit Laptop dan 3 unit komputer dengan spesifikasi : *Processor Intel® Core™ i3 M350 (2.3GHz), Memory RAM 4 GB, dan Broadcom NetLink™ Gigabit Ethernet Network Adapter.*

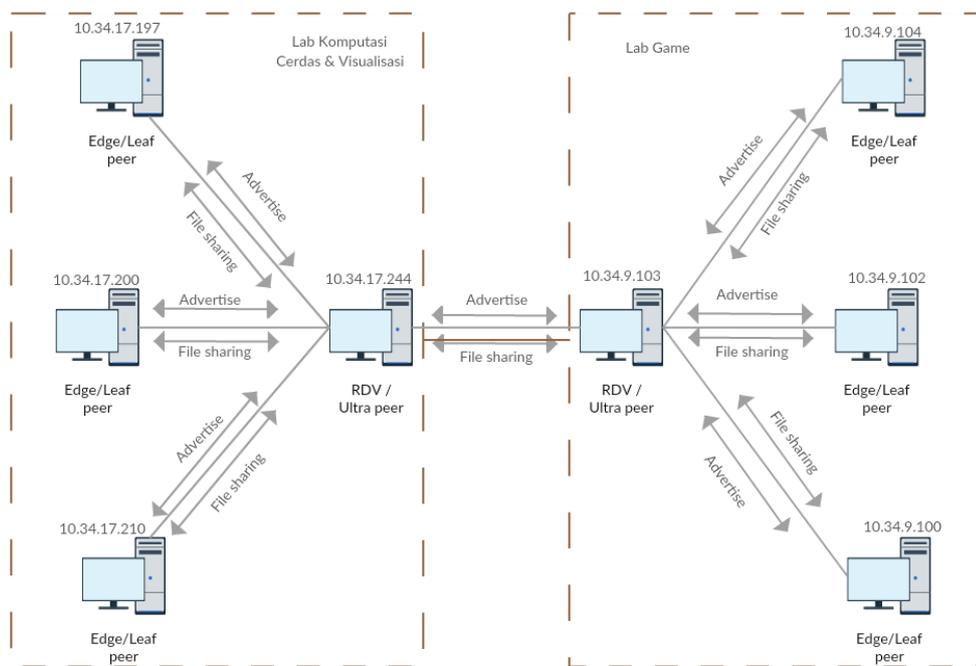
Perangkat Lunak:

- Sistem operasi Windows 7 Ultimate 64bit dan Ubuntu 14.04 LTS.
- VMware Workstation versi 11.0.0.
- JRE versi 6.0.17.
- Gnucleus versi 2.2.0.0.

- Wireshark versi 2.0.4.
- NetBeans IDE versi 8.0
- Wine 2.0

### 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 fungsional dan non-fungsional yang didapat dari tahap analisis kebutuhan. Pada tahap ini, rancangan sistem digambarkan dengan rancangan lingkungan pengujian sistem yang didasarkan pada 2 fungsi utama, yaitu: fungsi *advertise* yang dilakukan oleh setiap *peer* dan fungsi *file share* oleh setiap *peer* dan kepada *peer* lain.



**Gambar 3.2 Desain Arsitektur Sistem**

Gambar 3.2 menggambarkan arsitektur sistem yang akan dibuat. Dimulai dari *RDV/Ultra peer* (10.34.17.244) yang saling terhubung dengan *RDV/Ultra peer* (10.34.9.103) lalu kedua *peer* tersebut melakukan *advertisement* yang berarti melakukan pertukaran informasi satu sama lain. Informasi yang dibagikan saat melakukan *advertisement* adalah informasi yang dibutuhkan oleh masing-masing *peer* untuk melakukan *file sharing* yang didalamnya terdapat alamat IP *peer* dan nama *file* yang dibagikan oleh *peer*. Selanjutnya *Edge/Leaf peer* terhubung dengan *RDV/Ultra peer* masing-masing dan melakukan *advertisement*. Setelah itu antar *RDV/Ultra peer* melakukan *advertisement* kembali untuk memperbarui informasi yang didapatkan dari *Edge/Leaf peer*. Setelah informasi diperbarui fungsi *file sharing* dapat dilakukan antar *peer*.

Mekanisme *file sharing* dimulai dengan salah satu *peer* melakukan pencarian terhadap *file* yang diinginkan dalam bentuk *query* atas nama *file* tersebut. Selanjutnya *query* tersebut dikirimkan kepada *RDV/Ultrapeer* untuk dicocokkan apakah nama *file* yang diinginkan ada atau tidak pada *index file* yang terdapat pada *RDV/Ultrapeer*. Jika *file* ditemukan, maka *RDV/Ultrapeer* akan menghubungi *peer-peer* yang memiliki *file* tersebut dan meminta untuk mengirimkan *file* tersebut pada *peer* yang melakukan *request* pada *file* tersebut.

Pada saat proses *file sharing* berlangsung, *file* dikirimkan langsung kepada *peer* yang melakukan permintaan jika kedua *peer* yaitu *peer* yang berperan sebagai penerima *file* dan *peer-peer* yang memiliki *file* tersebut berada pada satu jaringan yang sama. Proses pengiriman *file* berbeda jika kedua *peer* tersebut berada pada jaringan yang berbeda, sebagai contoh saat *peer* yang meminta *file* berada dibawah *RDV/Ultrapeer* yang memiliki alamat IP 10.34.17.244, sedangkan *peer* yang memiliki *file* yang diminta berada dibawah *RDV/Ultrapeer* yang memiliki alamat IP 10.34.9.103. Pengiriman *file* akan dilakukan dengan meneruskan *file* yang diminta kepada *RDV/Ultrapeer* yang telah terhubung. Selanjutnya *file* akan di kirimkan kepada *RDV/Ultrapeer* yang berada pada jaringan lain yang terhubung sehingga *RDV/Ultrapeer* yang lain dapat meneruskan *file* yang diinginkan kepada *peer* yang melakukan permintaan terhadap *file* tersebut.

### 3.3 Implementasi

Implementasi sistem dilakukan berdasarkan rancangan sistem yang dibuat sebelumnya. Implementasi hasil rancangan diharapkan memenuhi kebutuhan yang dianalisa sebelumnya. Adapun implementasi sistem sebagai berikut.

1. Melakukan instalasi *VMware* pada komputer dengan sistem operasi *Windows*.
2. Melakukan instalasi *Ubuntu* pada *VMware*.
3. Melakukan instalasi *Java Runtime Environment* versi 6.0.17 pada setiap komputer.
4. Melakukan instalasi *Wine* pada setiap komputer yang menggunakan sistem operasi *Ubuntu*.
5. Melakukan instalasi *Gnucleus* sebagai perangkat lunak yang menggunakan protokol *Gnutella*.
6. Melakukan instalasi *myjxta* sebagai perangkat lunak yang menggunakan protokol *JXTA*.
7. Melakukan instalasi *Wireshark* untuk melakukan analisis terhadap trafik jaringan.
8. Membuat skema *peer* sebagai *seeder*.
9. Menjalankan program yang menggunakan protokol *myjxta* dan *Gnucleus* pada setiap komputer dan *Virtual Machine* yang dibuat.

10. Melakukan konfigurasi pada setiap program yang dijalankan (menentukan peran sebagai *RDV/Ultra peer* atau *Edge/Leaf peer*, menentukan alamat *ip*, menentukan *peer* yang akan dihubungkan, dan menentukan *file* yang akan dibagikan).
11. Menjalankan fungsi *file share* dan melakukan mekanisme pengiriman *file* yang telah ditentukan.
12. Mengetahui hasil analisis performa menggunakan *Wireshark*.
13. Melakukan perbandingan dari hasil analisis masing-masing protokol.

### 3.4 Pengujian & Analisis

Pengujian merupakan tahap yang dilakukan untuk menguji program yang telah diimplementasikan. Pengujian dilakukan dengan mengunduh *file-file* yang telah ditentukan dan dilakukan secara bergantian dari *peer* yang berbeda-beda jumlahnya berdasarkan skema-skema berikut ini:

		Labgame			
		-	1	2	3
Labkcv	-	-	A	N	O
	1	K	B	C	F
	2	L	D	E	G
	3	M	H	I	-

**Tabel 3.1 Skema Pengujian**

Tabel 3.1 menunjukkan *peer* sebagai *seeder* saat pengujian berlangsung dengan jumlah seperti berikut ini:

- Satu *seeder* : skema A dan skema K
- Dua *seeder* : skema B, skema N, dan skema L
- Tiga *seeder* : skema C, skema D, skema M dan skema O
- Empat *seeder* : skema E, skema F, dan skema H
- Lima *seeder* : skema G dan skema I

Skema – skema yang dibuat untuk pengujian pada penelitian ini berfungsi menentukan jumlah dan letak *seeder* yang akan digunakan. Letak *seeder* yang digunakan pada setiap skema yang dibuat juga berbeda – beda. Berikut adalah penjelasan penggunaan *seeder* pada setiap skema:

- Pada skema A dan skema K menggunakan satu *seeder* yang masing – masing terletak pada Labkcv dan Labgame.
- Skema B, skema N, dan skema L menggunakan dua *seeder* yang masing – masing berada di dalam Labkcv dan Labgame, dua *seeder* di dalam Labgame, dan dua *seeder* di dalam Labkcv.

- Skema yang menggunakan tiga *seeder* yaitu skema C, skema D, skema M dan skema O masing – masing menggunakan dua *seeder* pada Labkcv dan satu *seeder* berada di Labgame dan sebaliknya untuk skema C dan skema D. Sedangkan untuk skema M dan skema O *seeder* berjumlah tiga pada Labgame atau Labkcv.
- Pada skema E, skema F dan skema H menggunakan empat *seeder*. Untuk skema E, empat *seeder* dibagi dua yang masing – masing terletak pada Labkcv dan Labgame dengan jumlah yang sama yaitu dua. Sedangkan pada skema F dan skema H *seeder* terletak pada masing – masing tempat yang jumlahnya yaitu tiga *seeder* berada pada Labkcv/Labgame dan satu *seeder* berada pada Labgame/Labkcv.
- Skema G dan skema I menggunakan lima *seeder* yang jumlah *seeder*-nya dibagi pada masing – masing tempat yaitu tiga *seeder* berada pada Labkcv/Labgame dan dua *seeder* berada pada Labgame/Labkcv.

Pada pengujian dengan menggunakan satu *seeder*(skema A dan skema K), dilakukan persiapan terhadap *file-file* yang telah ditentukan pada *peer* yang telah ditentukan sebagai *seeder*. Pada sisi penerima, dilakukan pencarian *file* tersebut dan jika ditemukan maka mekanisme *file sharing* dapat dilakukan.

Untuk pengujian pada skema yang menggunakan lebih dari satu *seeder* yaitu skema B, skema C, skema D, skema E, skema F, skema G, skema H, skema I dan skema L dimulai dengan mempersiapkan *file – file* yang akan digunakan untuk melakukan pengujian. *File – file* yang dipersiapkan pada *peer* yang berperan sebagai *seeder* kemudian akan dilakukan pencarian terhadap *file* yang telah ditentukan pada sisi penerima atau *receiver*. Selain itu juga dilakukan pengecekan terhadap alamat IP *seeder* apakah sesuai dengan skema. Jika *file* ditemukan dan *seeder* telah sesuai dengan skema yang diujikan, maka mekanisme *file sharing* dapat dilakukan.

Saat pengujian berlangsung, dilakukan *monitoring* terhadap *traffic* yang sedang berjalan yaitu saat mekanisme *file sharing* sedang berlangsung. *Monitoring* dilakukan hanya pada sisi *peer* yang berperan sebagai *receiver* dari mekanisme *file sharing*. Hal itu dilakukan karena pada sisi *receiver* dapat terlihat *peer* mana saja yang berperan sebagai *seeder* dan yang mengirimkan *file* tersebut sehingga hasil yang didapatkan saat pengujian berlangsung sesuai dengan perancangan yang dibuat sebelumnya.

Selanjutnya dilakukan analisis dari hasil pengujian. Pada tahap ini didapatkan hasil berupa *throughput* dan rata-rata *delay* dari masing-masing skema pengujian yang dilakukan. Nilai *throughput* dan rata-rata *delay* diperoleh dari perhitungan terhadap hasil paket data yang telah di-*capture* menggunakan aplikasi yang telah dipersiapkan pada setiap *peer* yang berperan sebagai *receiver*. Selanjutnya dilakukan analisis dan perbandingan terhadap masing-masing protokol yang digunakan. Hasil dari analisis dan perbandingan nantinya akan dijadikan acuan untuk pengambilan kesimpulan.

### **3.5 Kesimpulan dan Saran**

Kesimpulan dan saran merupakan tahap yang dilakukan setelah tahap –tahap sebelumnya terpenuhi yaitu tahap studi literatur, perancangan, implementasi, pengujian dan analisis hasil. Kesimpulan dibuat untuk menjawab permasalahan yang telah dirumuskan sebelumnya yang diperoleh dari analisis hasil pengujian untuk mengetahui performa pada setiap protokol dalam melakukan *file sharing*. Saran dituliskan untuk memberikan informasi yang diharapkan dapat digunakan sebagai acuan untuk penelitian selanjutnya sehingga kekurangan yang dihasilkan pada penelitian ini dapat diperbaiki.