

BAB 3 METODOLOGI

Pada bab ini dijelaskan langkah-langkah yang akan dilakukan dalam pengerjaan tugas akhir yaitu, studi literatur, perancangan, implementasi, analisis dan pengujian dari simulasi yang akan dibuat. Kesimpulan dan saran disertakan sebagai catatan atas aplikasi dan kemungkinan arah pengembangan aplikasi selanjutnya. Berikut diagram alir runtutan pengerjaan tugas akhir ini :



Gambar 3.1 Diagram Blok Metodologi

3.1 Studi Literatur

Studi literatur menjelaskan dasar teori yang digunakan untuk menunjang penulisan skripsi. Studi literature Teori-teori pendukung tersebut meliputi:

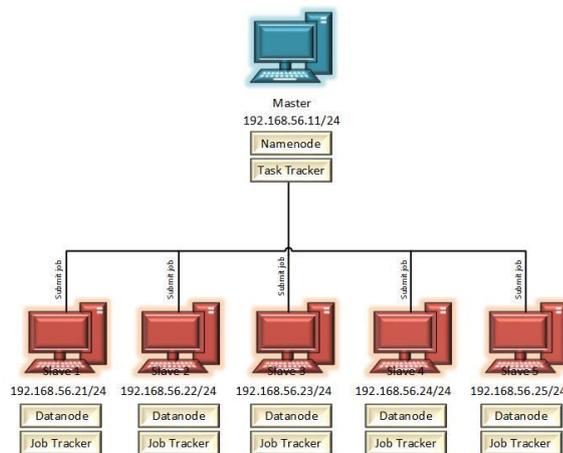
1. Konsep *Big Data*
2. Hadoop
 - a. Hadoop Distributed *File System* (HDFS)
 - b. Map Reduce
 - c. YARN
3. Hadoop *Multi Node*
4. Algoritme *Scheduling*
 - a. Algoritme *Fair Share Scheduling*
 - b. Algoritme *Capacity Scheduling*
5. Parameter Pengujian
 - a. *Job Fail Rate*
 - b. *Latency*
 - c. *Throughput*

1.2 Lingkungan Penelitian

Pada tahap lingkungan penelitian ini mencakup spesifikasi pembangunan jaringan pada sistem Hadoop dan konfigurasi-konfigurasi yang dibutuhkan untuk melakukan implementasi terhadap kinerja dari algoritme *Fair Share Scheduling* dan *Capacity Scheduling*. Pada Hadoop terdapat resource manager yang digunakan sebagai wadah untuk melakukan monitoring algoritme *scheduling* terhadap pengiriman *job*. Pada lingkungan penelitian ini dibuat sedemikian rupa sesuai dengan topologi yang direncanakan sehingga hasil pengujian pada algoritme *Fair Share Scheduling* dan *Capacity Scheduling*.

1.2.1 Topologi Jaringan

Pada pembangunan sistem Hadoop diperlukan infrastruktur jaringan yang dibutuhkan untuk membangun komunikasi antar *node* yang bersifat *multinode*. Pada Hadoop *multinode* ini diperlukan 6 *node* yang terdiri 1 komputer *head master* dan 5 komputer *virtual* yang digunakan sebagai komputer *slave*. Kemudian akan dilakukan konfigurasi Hadoop *multinode cluster*. Perancangan topologi jaringan dapat dilihat pada gambar 3.2.



Gambar 3.2 Arsitektur Topologi Jaringan

Pada gambar 3.2 arsitektur perancangan topologi jaringan Dimana *head master* memiliki IP 192.168.56.11/24 dan kelima *slave* memiliki IP yang berbeda yaitu, pada *slave* 1 memiliki IP 192.168.56.21/24, *slave* 2 memiliki IP 192.168.56.22/24, *slave* 3 memiliki IP 192.168.56.23/24, *slave* 4 memiliki IP 192.168.56.24/24, dan *slave* 5 memiliki IP 192.168.56.25/24. Kelima komputer *slave* tersebut digunakan untuk mengirim *job* ke komputer *master*. Pada komputer *slave* bisa mengirimkan lebih dari satu *job* ke komputer *master*. Jenis *job* yang digunakan untuk distribusi data berupa *job wordcount* dan *job wordmean*

3.3 Implementasi

Implementasi merupakan penjelasan secara umum dari proses instalasi dan konfigurasi Hadoop sesuai dengan lingkungan perancangan yang diinginkan. Implementasi ini digunakan untuk perancangan analisis dari *algoritme scheduling* dengan menggunakan parameter *Job Fail Rate*, *Latency*, dan *Throughput*. Pada analisis ini dilakukan dari mulai perancangan kebutuhan sampai hasil akhir analisis dari data yang diperoleh dari hasil simulasi. Implementasi lebih difokuskan dalam perancangan sistem yang meliputi:

1. Spesifikasi sistem.
2. Instalasi Hadoop *multinode cluster*.
3. Konfigurasi jaringan.
4. Konfigurasi Hadoop *multinode cluster*.
5. Konfigurasi *Capacity Scheduling*.
6. Konfigurasi *Fair Share Scheduling*.
7. Konfigurasi *Job WordCount*.

3.4 Pengujian

Tahap pengujian pada penelitian ini dilakukan pada Hadoop *multinode* yang terdiri dari 3 skenario pengujian yang berbeda pada masing-masing algoritme *Fair Share Scheduling* dan *Capacity Scheduling*. Pada masing-masing algoritme *scheduling* akan dilakukan pengiriman *job* dari *slave* ke *master* Hadoop berdasarkan skenario pengujian yang ditentukan. Pengujian ini dilakukan dengan 3 skenario pengujian yang berbeda berdasarkan pada variabel bebas dan variabel tetap. Untuk skenario pengujian pertama menggunakan variabel bebas berupa variasi ukuran data dan variabel tetap yaitu, jumlah *job* dan jenis *job*. Skenario pengujian kedua menggunakan variabel bebas berupa variasi jumlah *job* dan variabel tetap yaitu, ukuran data dan jenis *job*. Skenario pengujian ketiga menggunakan variabel bebas berupa variasi jenis *job* dan variabel tetap yaitu, ukuran data dan jumlah *job*.

Pada tabel 3.1 akan menjelaskan ke 3 skenario yang akan dilakukan pada proses pengujian pada masing-masing *scheduling* ini sebagai berikut.

Tabel 3.1 Skenario Pengujian

Skenario Pengujian	Variabel Bebas	Variabel Tetap
Variasi ukuran data	Variasi Ukuran data = 1,5 GB, 386,81 MB, dan 20 MB	Jumlah <i>Job</i> = 5 <i>job</i>
		Jenis <i>Job</i> = wordcount
Variasi jumlah <i>job</i>	Variasi Jumlah <i>Job</i> = 5 <i>job</i> , 10 <i>job</i> , dan 15 <i>job</i>	Ukuran data = 1,5 GB
		Jenis <i>Job</i> = wordcount
Variasi jenis <i>job</i>	Variasi Jenis <i>Job</i> = wordcount dan wordmean	Ukuran data = 1,5 GB
		Jumlah <i>Job</i> = 5

Pada pengujian ini akan menggunakan parameter pembandingan yaitu, *Job Fail Rate*, *Latency*, dan *Throughput* sebagai acuan perhitungan kinerja sistem. Akumulasi dari semua hasil pengujian yang didapatkan, akan diperoleh berdasarkan grafik nilai untuk mengetahui perbandingan nilai dari masing-masing algoritme *Fair Share Scheduling* dan *Capacity Scheduling* sehingga dapat diketahui kinerja terbaik antara kedua algoritme *scheduling* untuk diimplementasikan pada pengiriman *job* pada sistem Hadoop.

Pada pengujian ini menggunakan 2 jenis *job* yaitu, *job wordcount*, dan *job wordmean*. Pada kedua jenis *job* ini memiliki perbedaan karakteristik yang dapat dilihat pada tabel 3.2.

Tabel 3.1 Jenis Job Hadoop

No.	Jenis Job	Karakteristik Job
1.	Word Count	<i>Job</i> ini digunakan untuk menghitung kemiripan jumlah kata yang terdapat pada sebuah <i>file</i> yang berukuran besar. <i>File</i> tersebut berupa <i>file plain text</i> yang dipakai sebagai data dan terletak pada direktori HDFS pada Hadoop.
2.	Word Mean	<i>Job</i> ini digunakan untuk menghitung rata-rata dari jumlah keseluruhan kata yang terdapat pada suatu <i>file input</i> yang terletak pada direktori HDFS pada sistem Hadoop.

Untuk melakukan pengiriman *job* kedalam *cluster* Hadoop, pada pengujian ini diperlukan adanya *dataset* dalam bentuk *plaintext* yang digunakan pada ketiga jenis *job* tersebut. *Dataset* ini terletak pada direktori HDFS pada Hadoop sistem yang berfokus ada pencarian huruf dan kata. Data ini harus memenuhi syarat dari Hadoop dengan kategori data yang berukuran besar hingga mencapai ukuran gigabyte per *filenya*. Berikut ini adalah isi data yang akan digunakan pada pengujian *scheduling* dari setiap *resource* dapat dilihat pada tabel 3.3.

Tabel 2.3 Dataset pada HDFS

No.	Nama File	Deskripsi	Ukuran data	Maps
1.	Facebook-names-unique.txt (Ron Bowes, 2010)	Berisi nama pengguna facebook yang terdiri dari nama depan dan nama belakang tanpa adanya duplikasi data.	1,5 GB	12
2.	Facebook-f.last-withcount.txt (Ron Bowes, 2010)	Berisi nama belakang pengguna facebook tanpa adanya duplikasi data.	286,81 MB	3

3.	Adult.csv (UCI adult, 1996)	Data adult ini merupakan <i>dataset</i> pendapatan penduduk di berbagai negara yang digunakan untuk mempresiksi pendapatan tiap penduduk apakah melebihi \$50K atau kurang dari \$50K.	20 MB	1
----	--------------------------------	--	-------	---

Tujuan dilakukan skenario pengujian ini untuk mengukur kinerja dari *Fair Share Scheduling* dan *Capacity Scheduling*. Metode yang digunakan adalah dengan melakukan pengiriman *job* berdasarkan jumlah dan jenis *job* yang berbeda pada setiap skenario. Perbedaan dari jenis *job* dan jumlah *job* yang dilakukan pada skenario pengujian ini dapat memengaruhi kinerja dari masing-masing algoritme *scheduling* tersebut.

3.5 Hasil dan Analisis

Pada pengujian ini dimaksudkan untuk mendapatkan hasil serta dapat melakukan analisis terhadap kinerja dari *Fair Share Scheduling* dengan *Capacity Scheduling*. Dengan menggunakan parameter pengujian yaitu, *Job Fail Rate*, *Latency*, dan *Throughput*. Hasil pengujian ini didapat dari tiga skenario pengujian yang berbeda-beda berdasarkan variasi ukuran data, variasi jumlah *job*, dan variasi jenis *job*. Hasil pengujian ini akan membandingkan kinerja dari algoritme *Fair Share Scheduling* dan *Capacity Scheduling* berdasarkan grafik nilai dari parameter pengujian *Job Fail Rate*, *Latency*, dan *Throughput*.

Analisis perancangan mencakup analisis kebutuhan dan perancangan umum. Hasil analisis ini ditunjukkan dengan grafik yang menggambarkan suatu nilai pembandingan dari parameter *Job Fail Rate*, *Latency*, dan *Throughput* yang telah diuji pada setiap *Job scheduling*. Dari grafik tersebut dapat diketahui kelemahan dan kelebihan pada masing-masing *job scheduling* dari 3 skenario pengujian yang telah dibuat dengan menggunakan jenis *Job wordcount* dan *job wordmean*. Dari hasil nilai tersebut digunakan sebagai dasar melakukan analisis untuk mengetahui kinerja terbaik antara algoritme *Fair Share Scheduling* dengan *Capacity Scheduling*.

3.6 Kesimpulan dan Saran

Pengambilan kesimpulan dilakukan setelah semua tahapan perancangan, implementasi dan analisis pengujian telah selesai dilakukan. Kesimpulan diambil dari hasil pengujian dan analisis ini untuk mengetahui kinerja terbaik dari algoritme *Fair Share Scheduling* dan *Capacity Scheduling* dengan menggunakan *framework* Hadoop. Tahap terakhir dari penulisan ini adalah saran yang dimaksudkan untuk memperbaiki kesalahan-kesalahan yang terjadi selama melakukan penelitian ini dan menyempurnakan penulisan serta untuk memberikan pertimbangan atas pengembangan algoritme penjadwalan lainnya dengan menggunakan parameter berbeda serta jenis *job* yang lebih bervariasi.