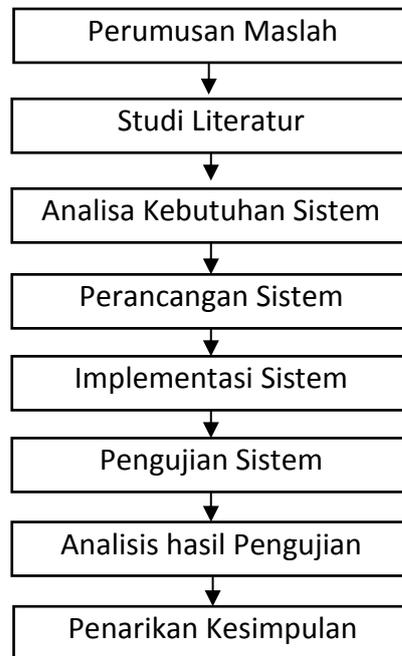


BAB 3 METODOLOGI

Pada bab ini dibahas mengenai metodologi penelitian yang digunakan beserta tahapan-tahapan yang akan diimplementasikan, sebagai berikut :



Gambar 3.1 Diagram pelaksanaan penelitian

3.1 Perumusan Masalah

Permasalahan yang dihadapi dalam penelitian ini adalah bagaimana mengimplementasi sistem laboratorium virtual mandiri yang dapat mengakomodir kebutuhan mahasiswa saat melakukan praktikum seperti ketika melakukan praktikum pada laboratorium secara konvensional. Dan bagaimana mengatasi ketidakseragaman konfigurasi pada saat melakukan praktikum. Seperti yang telah dijelaskan sebelumnya praktikum yang dilaksanakan di laboratorium konvensional masih memiliki beberapa kekurangan seperti banyak mahasiswa yang masih kebingungan untuk melakukan konfigurasi beberapa aplikasi untuk praktikum. Salah satu solusi dari permasalahan tersebut adalah menggunakan laboratorium virtual mandiri dimana setiap pengguna (mahasiswa) memiliki komputer virtual mandiri sesuai dengan kebutuhan praktikum sehingga mahasiswa dapat melakukan praktikum dengan efektif dan efisien.

3.2 Studi Literatur

Tahapan ini dilakukan untuk mempelajari tentang macam-macam literatur yang berhubungan dengan penelitian yang akan penulis lakukan yaitu perancangan laboratorium virtual menggunakan *docker*. Literatur ini nantinya akan digunakan untuk penguat serta landasan dasar pada penelitian yang

dilakukan penulis. Literatur diperoleh dari berbagai sumber, diantaranya adalah jurnal, buku serta internet. Beberapa literature yang menjadi referensi penulis sebagai berikut:

1. Studi berkaitan dengan *virtualization* dari penelitian Maholtra dkk (2014) dengan judul *Virtualization In Cloud Computing*. Dalam penelitian tersebut membahas tentang *cloud computing*, bagaimana virtualisasi dapat berjalan di *cloud computing*, dan arsitektur dasar virtualisasi.
2. Studi berkaitan dengan *container* dari penelitian yang dilakukan oleh Dwiyanto Gunawan (2016) dengan judul Rancang Bangun Arsitektur Laboratorium Virtual Jaringan Komputer Dengan Virtualisasi Berbasis *Container*. Penelitian tersebut membuat sistem *back end* laboratorium virtual yang mampu menyediakan komputer virtual.
3. Studi berkaitan tentang laboratorium virtual dari penelitian yang dilakukan oleh Sumarto (2014) dengan judul virtual lab berbasis *cloud computing* menggunakan *platform ovirt* pada jaringan terpusat. Dalam penelitiannya penulis melakukan pengujian terhadap lab komputer virtual dengan studi kasus matakuliah pemrograman berbasis objek menunjukkan bahwa *server* yang digunakan selama pengujian mampu menjalankan 10 komputer virtual dengan baik.
4. Studi berkaitan dengan *docker* yang ditulis dengan Sebastien Goasguen (2016) dengan judul *Docker Cook Book*. Buku tersebut membahas tentang *manage container*, membuat dan membagikan *container* hingga tahap lanjutan seperti konfigurasi *docker* dan pengembangan.

3.3 Analisa Kebutuhan Sistem

Analisa kebutuhan dilakukan untuk mendapatkan semua yang dibutuhkan untuk pembuatan laboratorium virtual jaringan komputer. Oleh karena itu diperlukan definisi dari proses praktikum tersebut serta apa yang membuat sesi pembelajaran dapat disebut sebagai suatu praktikum.

Menurut kamus besar Bahasa Indonesia praktikum merupakan bagian dari proses belajar yang memberikan kesempatan siswa untuk menguji dan mengimplementasikan apa yang mereka dapat dalam hal teori. Pembelajaran dalam hal ini yang dimaksud adalah matakuliah jaringan komputer.

3.3.1 Identifikasi Aktor

Identifikasi aktor digunakan untuk mengidentifikasi aktor-aktor yang terlibat dalam sistem yang dirancang. Identifikasi aktor ini dapat dilihat pada table berikut ini:

Tabel 3.1 Tabel Pengujian *Container* Berdasarkan Memory

Aktor	Deskripsi
Dosen / pengajar	Dosen atau pengajar dapat membuat <i>image</i> sesuai bab praktikum jaringan komputer melalui <i>dockerfile</i> kemudian <i>push image</i> tersebut pada <i>docker registry</i> .
Mahasiswa	Mahasiswa dapat <i>pull image</i> dari <i>docker registry</i> yang disediakan dosen dan digunakan untuk praktikum.

3.3.2 Kebutuhan Fungsional

Dari penjelasan diatas maka penulis menganalisis kebutuhan fungsional dari sistem yang ada. Beberapa kebutuhan tersebut adalah:

1. Dosen atau pengajar dapat membuat testbed laboratorium virtual jaringan komputer.
2. Dosen atau pengajar dapat membuat *image* setiap bab praktikum jaringan komputer.
3. Dosen atau pengajar melakukan *upload image* pada *docker registry* yang tersedia.
4. Mahasiswa dapat melakukan *pull image* sesuai bab praktikum.
5. Mahasiswa dapat membuat *container* baru.
6. Mahasiswa dapat menghapus *container*.

3.3.3 Kebutuhan Non-Fungsional

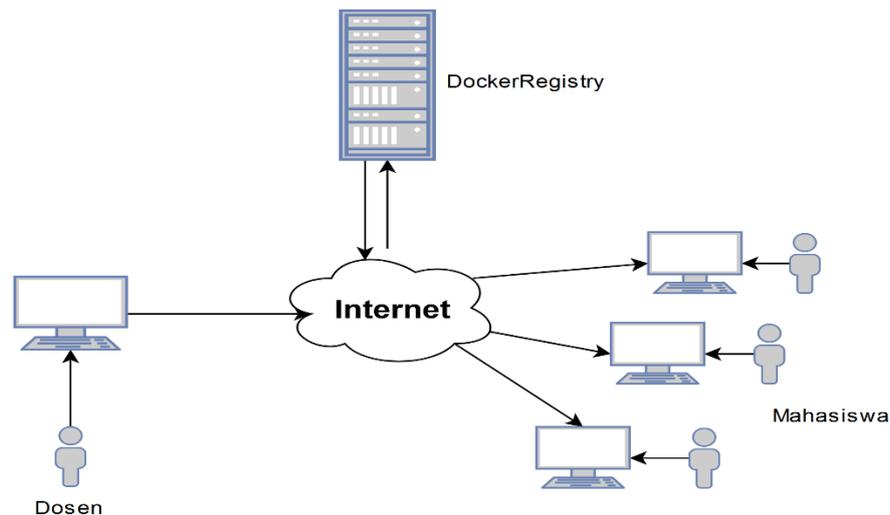
Dari variabel-variabel yang telah disebutkan sebelumnya merupakan kebutuhan fungsional sistem, maka dari itu juga dibutuhkan beberapa komponen yang bersifat non-fungsional untuk mendukung penelitian ini, dimana komponen akan dibagi menjadi dua bagian yaitu dari segi perangkat keras dan perangkat lunak. Berikut adalah kebutuhan non-fungsional:

- Perangkat Keras (*Hardware*) :
 - Processor intel Core i3-2370M CPU2.40GHz
 - RAM 8 GB
 - HDD 500 GB
 - Monitor 14 inc
- Perangkat Lunak (*Software*) :

- Sistem Operasi Linux Ubuntu 14.04
- *Docker Engine*

3.4 Perancangan Sistem

Pada tahap ini sistem akan dirancang dan dibangun dari penelitian yang telah dilakukan yaitu studi literatur dan analisis kebutuhan sistem. Perancangan sistem dapat dilihat pada Gambar 3.2.



Gambar 3.2 Perancangan Sistem

Seperti yang terlihat pada Gambar 3.2 merupakan perancangan yang akan dibuat untuk laboratorium virtual jaringan komputer berbasis *docker*, dalam pembuatan laboratorium virtual ini dibutuhkan sebuah *docker registry* yang merupakan tempat penyimpanan *image* yang telah dibuat oleh dosen.

Pertama-tama dosen atau pengajar membuat (menyediakan) komponen yang dibutuhkan pada bab praktikum jaringan komputer yang di konfigurasi dan terisolasi dalam sebuah *container*. *Container* tersebut dijadikan sebuah *image* dan di *upload (push)* pada *dockerhub*.

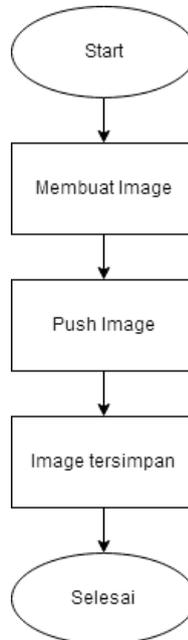
Dockerhub merupakan tempat penyimpanan atau *repository* berbasis *cloud* untuk menyimpan *image* yang telah dibuat. Untuk menyimpan *image* pada *dockerhub* diperlukan akun yang teregistrasi.

Mahasiswa hanya melakukan *download (pull) image* pada *dockerhub* yang telah dikonfigurasi dan menjalankan *container* tersebut pada masing-masing perangkatnya sesuai bab praktikum yang sedang dilakukan melalui Terminal. Terminal merupakan *command line* yang disediakan linux secara *default* yang digunakan untuk *pull image* untuk menjalankan sebuah *image*.

Dalam setiap *push* dan *pull image* diperlukan jaringan yang menghubungkan ke internet. Pada system yang dibuat ini jaringan internet menggunakan bridge untuk menghubungkan antara *host* dan *docker*.

3.4.1 Perancangan Sistem Penyediaan Materi Praktikum

Sistem penyediaan *image* praktikum ini digunakan untuk memberikan materi praktikum dalam bentuk *image* yang nantinya akan dipakai mahasiswa sebagai praktikum, Dimana sistem ini menyediakan fitur pembuatan *image*. Berikut alur kerja sistem dalam bentuk *flowchart*.



Gambar 3.3 Flowchart Penyediaan Materi Praktikum

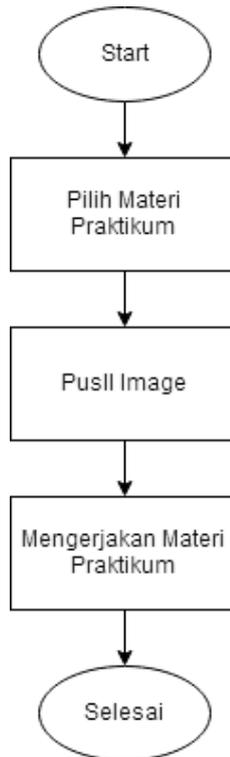
Gambar 3.3 merupakan alur kerja sistem untuk menyediakan *image* praktikum. Pada awalnya *image digenerated* melalui *dockerfile* yang sudah dibuat dan berisi baris-baris perintah untuk melakukan instalasi keperluan materi praktikum.

Selanjutnya adalah *image* di *push* ke *docker registry* yang merupakan *repository* atau tempat penyimpanan *docker image* secara *cloud*. Pada *docker registry* ini *repository* dapat dibuat sesuai keinginan. diperlukan akun yang teregistrasi untuk menyimpan *image* di *repository dockerhub*.

3.4.2 Perancangan Pengerjaan Materi Praktikum Jaringan Komputer

Perancangan praktikum digunakan sebagai kebutuhan yang diperlukan mahasiswa untuk melakukan praktikum jaringan komputer. Mahasiswa diberikan akses untuk menggunakan *image* yang di-*download* sebagai perangkat melaksanakan praktikum.

Materi praktikum jaringan komputer yang digunakan dalam penelitian ini ada tiga materi. Mahasiswa hanya *pull image* dan mengerjakan praktikum sesuai materi. Berikut merupakan langkah-langkah dalam bentuk *flowchart*



Gambar 3.4 Flowchart Pengerjaan Materi Praktikum

Gambar 3.4 merupakan alur kerja sistem untuk pengerjaan materi praktikum. Seperti yang telah dijelaskan sebelumnya, materi praktikum pada penelitian ini ada tiga materi sehingga mahasiswa dapat memilih materi yang akan dikerjakan untuk praktikum.

kemudian *pull* materi tersebut dari *docker registry*. *Pull* tersebut bertujuan untuk menarik *image* praktikum yang sudah disediakan pada *dockerhub* sehingga dapat dikerjakan pada masing-masing laptopnya.

3.5 Implementasi Sistem

Implementasi sistem membahas langkah-langkah membangun sebuah sistem yang telah diimplementasikan. Tahapan implementasi antara lain:

- Melakukan instalasi *docker engine* pada sistem operasi ubuntu 14.04.
- Melakukan instalasi dan konfigurasi *container* menggunakan *dockerfile* untuk laboratorium virtual mandiri.
- Melakukan *push image* pada *docker registry* yang merupakan tempat penyimpanan image secara *cloud*.
- Melakukan *pull image* untuk mengerjakan materi praktikum yang disediakan pada *docker registry*
- Menguji dan menganalisis sistem laboratorium virtual jaringan komputer

3.6 Pengujian Sistem

Pada tahap ini akan dilakukan pengujian dan analisis pengujian dari parameter yang telah ditentukan antara lain kemampuan komputer dalam menampung banyak *container*, berapa banyak *container* yang dapat terpasang pada komputer mahasiswa.

3.6.1 Pengujian Performa Komputer Berdasarkan CPU untuk membuat *container*

Skenario pada pengujian banyak *container* yang dapat terpasang dilakukan dengan melihat performa komputer. Performa komputer dapat dilihat dari sisi CPU yang terpakai. Terdapat 3 skenario pengujian yang akan dilakukan, dapat dilihat pada table 3.2

Tabel 3.2 Tabel Pengujian *Container* Berdasarkan Beban pada CPU

Skenario	Jumlah <i>Container</i> yang dibuat	Beban Kerja yang dibutuhkan Sistem untuk membuat image (%)
1	5	Hasil Diketahui Pada Saat Pengujian
2	10	
3	15	
4	20	
Rata-rata		

3.6.2 Pengujian Performa Komputer Berdasarkan *Memory* Untuk Membuat *Container*

Skenario pada pengujian ini dilakukan dengan membuat *container* pada masing perangkat mahasiswa. *Container* akan dibuat sesuai dengan skenario yang telah ditetapkan. Pada saat *container* dijalankan untuk praktikum maka akan dilihat berapakah beban *memory* yang didapat computer. Hasil pengujian dapat dilihat pada tabel 3.3

Tabel 3.3 Tabel Pengujian *Container* Berdasarkan Beban pada *memory*

Skenario	Jumlah <i>Container</i> yang dibuat	Beban Kerja yang dibutuhkan Sistem untuk membuat image (MB)
1	5	Hasil Diketahui Pada Saat Pengujian
2	10	
3	15	
4	20	
Rata-rata		

3.6.3 Pengujian Fungsional dan Nonfungsional Sistem

Pada pengujian ini akan dilakukan percobaan sistem untuk mengetahui apakah sistem berjalan dengan baik atau sebaliknya. Hasil pengujian tersebut akan dimasukkan kedalam bentuk kuisisioner seperti pada tabel 3.4 dan table 3.5

Tabel 3.4 Tabel Pengujian Fungsional

No	Pengujian	Hasil
1.	Dosen/pengajar dapat membuat <i>tes bed</i> laboratorium virtual jaringan komputer	Hasil Diketahui Pada Saat Pengujian
2.	Dosen dapat melakukan upload (<i>push</i>) <i>image</i>	
3.	Dosen dapat hapus <i>image</i>	
4.	Mahasiswa dapat download (<i>pull</i>) <i>image</i>	
5.	Mahasiswa dapat menghapus laboratorium virtual yang dibuat	
6.	Mahasiswa dapat membuat kembali praktikum berdasarkan <i>image</i> yang sudah ada	

Tabel 3.5 Tabel Pengujian Non-Fungsional

No	Pengujian	Hasil
1.	<i>Docker image</i> dapat di <i>generated</i> dari <i>dockerfile</i> yang sudah dibuat	Hasil Diketahui Pada Saat Pengujian
2.	<i>Image</i> Teori dan Aplikasi Dasar Jaringan Komputer dapat dijalankan untuk praktikum	
3.	<i>Image packet capturing</i> dapat menangkap packet yang lewat pada jaringan wireless	
4.	<i>Image</i> Dasar Pemrograman Jaringan Komputer dapat dijalankan untuk praktikum	
5.	<i>Image</i> pemrograman <i>socket</i> dapat menjalankan <i>python</i> di dalam <i>container</i> sebagai dasar praktikum	
6.	<i>Image</i> pemrograman <i>socket</i> dapat menghubungkan antara <i>client</i> dan <i>server</i> untuk praktikum	
7.	<i>Client</i> pada praktikum pemrograman <i>socket</i> dapat mengirimkan pesan pada <i>server</i> yang dijalankan	
8.	<i>Server</i> pada praktikum pemrograman <i>socket</i> dapat menerima pesan dari <i>client</i> dan merespon	
9.	<i>Image Protokol Layer Transport</i> dapat dijalankan untuk praktikum	

10.	<i>Image protocol layer transport</i> dapat melakukan <i>upload file</i> pada <i>ftp server</i>	
11.	<i>Ftp server</i> dapat diakses melalui <i>browser</i> ataupun aplikasi <i>filezilla</i>	
12.	<i>Ftp server</i> dapat menyimpan <i>file</i> yang di <i>upload</i> pada <i>container</i> yang dibuat	
13.	Septiap <i>image</i> praktikum dapat melakukan <i>ping</i> pada jaringan	

3.7 Kesimpulan

Kesimpulan didapatkan dari hasil pengujian dan hasil analisis metode. Bagian terakhir pada penulisan adalah pemberian saran yang bertujuan untuk bisa memperbaiki kesalahan dan memberi pertimbangan untuk penelitian selanjutnya. Tujuan dari pembuatan laporan ini adalah untuk membuat penelitian ini lebih terperinci dan agar penelitian ini dapat digunakan sebagai referensi pada penelitian selanjutnya.