

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Melakukan konfigurasi terhadap *service* yang ada pada sistem operasi Linux tidaklah mudah bagi orang yang baru pertama kali mengenal sistem operasi Linux. Hal ini terjadi bagi mahasiswa yang berasal dari SMA atau SMK yang masuk ke kampus TI dan di dalam kurikulumnya mengajarkan mengenai sistem operasi Linux dan juga jaringan.

Berdasarkan survei yang dihimpun dari 57 Mahasiswa didapatkan 56,14% atau 32 mahasiswa yang pernah menggunakan sistem operasi Linux dengan berbagai macam distro. Dari jumlah mahasiswa yang pernah menggunakan sistem operasi Linux tersebut diperoleh data sebesar 75% mengalami kendala dalam melakukan konfigurasi pada sistem operasi tersebut. Hal tersebut terjadi karena mengalami kesulitan ketika menggunakan terminal Linux untuk melakukan konfigurasi. Kebanyakan mahasiswa mengalami kesulitan karena hampir semua pengoperasian sistem operasi linux menggunakan *command line interface* dalam mengkonfigurasi suatu sistem atau ketika melakukan pemasangan *software*. Tidak seperti halnya dengan sistem operasi *Windows* dalam melakukan konfigurasi atau pemasangan *software* hanya dengan melakukan klik ganda dan pengisian beberapa data sesuai dengan kebutuhan. Hal ini yang menyebabkan mahasiswa kurang berminat dalam sistem operasi Linux, selain itu karena banyaknya *command line* dan juga atribut tambahan yang digunakan dalam konfigurasi menyebabkan terhambatnya proses pembelajaran mengenai konfigurasi *service* paket.

Shell programming bisa dijadikan menjadi salah satu solusi yang bisa digunakan untuk mengembangkan sebuah aplikasi yang bertujuan untuk membantu pengguna untuk melakukan konfigurasi dengan lebih mudah. Salah satu tujuan dari penggunaan *shell programming* otomatisasi dalam penggunaan beberapa *command* yang nantinya akan digunakan untuk melakukan konfigurasi. Selain untuk otomatisasi juga bertujuan untuk melakukan efisiensi kumpulan beberapa

command line yang digunakan untuk melakukan banyak konfigurasi sehingga menjadi satu buah *script* yang meringkas seluruh *command line* yang akan digunakan. Dengan meringkas *command line* yang akan digunakan untuk tujuan tertentu maka akan mempermudah seorang pengguna atau *administrator* Linux dalam melakukan konfigurasi terhadap *service* yang ada pada sistem operasi Linux. Penggunaan *shell programming* untuk membuat sebuah program yang ditujukan untuk melakukan konfigurasi terhadap layanan yang ada sistem operasi Linux akan jauh lebih membantu seseorang dalam melakukan konfigurasi di sistem operasi Linux karena lebih efisien baik dalam hal waktu dan juga dalam proses konfigurasi.

Berdasarkan permasalahan yang ada dan solusi yang mungkin dapat digunakan, muncul sebuah ide untuk membuat sebuah aplikasi *open source* berbasis *shell programming* yang dapat digunakan untuk membantu pengguna maupun administrator dalam menggunakan sistem operasi Linux guna melakukan konfigurasi *service*. Aplikasi yang dikembangkan dengan memanfaatkan *shell programming* sebagai bahasa atau *script* yang digunakan untuk dasar dari aplikasi tersebut. Selain dengan *shell programming* aplikasi tersebut akan memanfaatkan *package* yang tersedia pada sistem operasi Linux yakni Zenity. Zenity merupakan *package* yang sudah tersedia pada sistem operasi Linux yang menggunakan *General User Interface(GUI)* GNOME yang berfungsi untuk membentuk sebuah tampilan atau *user interface* yang terintegrasi dengan *Shell* sehingga nanti aplikasi tersebut akan membantu pengguna dalam melakukan konfigurasi. Dengan pengembangan aplikasi ini diharapkan akan membantu pengguna dalam menggunakan sistem operasi Linux baik sebagai pemula maupun ditujukan untuk administrator dalam mengelola *server*.

1.2 Rumusan Masalah

Rumusan masalah yang akan dibahas lebih lanjut dalam proyek akhir ini adalah bagaimana membuat aplikasi berbasis shell untuk memudahkan pengguna dalam memanfaatkan sistem operasi Linux dalam melakukan konfigurasi *service* jaringan.

1.3 Tujuan

Adapun tujuan dari penyusunan proyek akhir ini adalah membuat pengguna yang baru saja mengenal dunia Linux terbantu dalam mengoperasikan sistem operasi Linux serta membantu pengguna atau *administrator* dalam melakukan konfigurasi *service* jaringan dengan memanfaatkan *shell programming*.

1.4 Batasan Masalah

Adapun batasan masalah dalam proyek akhir ini adalah sebagai berikut.

1. Aplikasi ini terbatas dalam beberapa sistem operasi seperti Ubuntu, Cent OS dan FreeBSD.
2. Menggunakan Zenity sebagai *package* pembentuk *user interface*-nya yang tersedia pada GNOME, KDE dan LXDE.
3. Pengguna dapat melakukan *install, update, remove* paket yang diinginkan sesuai daftar yang disediakan dalam menu.
4. Pengguna dapat melakukan *install* dan *remove* paket yang berada di luar list paket dengan memasukan nama paket.
5. Pengguna dapat memilih *repository* yang diinginkan secara dari list yang disediakan.
6. Aplikasi harus dijalankan dalam mode *user root*.
7. Aplikasi dapat digunakan untuk proses *mounting* dari *external harddrive*.
8. *Service* yang disediakan pada aplikasi ini meliputi DNS, SNMP, Email, DHCP, FTP, *Virtualization* dan juga *web server*.
9. Aplikasi tidak berjalan pada sistem operasi *server mode* atau tanpa GUI GNOME, KDE, LXDE dan sejenisnya.

1.5 Definisi Operasional

Aplikasi adalah penerapan dari rancangan sistem untuk mengolah data yang menggunakan aturan atau ketentuan bahasa pemrograman tertentu dan merupakan suatu program komputer yang dibuat untuk mengerjakan dan melaksanakan tugas khusus dari pengguna.

Shell merupakan aplikasi yang memungkinkan pengguna dapat berkomunikasi dengan komputer. Tugas *shell* adalah membaca perintah yang diberikan pemakai dan menerjemahkan perintah (*command interpreter*) tersebut sebagai suatu permintaan dan meneruskan ke *kernel* [1].

Linux adalah sistem operasi turunan dari sistem operasi Unix. Unix sendiri adalah sistem operasi jaringan yang digunakan untuk melayani jaringan-jaringan besar dan termasuk sistem operasi yang cukup tua umurnya [2].

1.6 Metode Pengerjaan

Adapun metode yang saya gunakan dalam proses pengerjaan proyek akhir ini adalah dengan menggunakan *Software Development Life Cycle* dengan metode *prototyping*. Adapun tahapan yang dilakukan adalah sebagai berikut :

1. Pengumpulan kebutuhan

Mengumpulkan teori dan literatur, dokumentasi dan petunjuk-petunjuk yang berkaitan dengan proyek akhir yang akan dilaksanakan, melakukan diskusi dengan pembimbing akademik mengenai proyek akhir yang akan dibangun.

2. Membangun *Prototyping*

Membangun *prototype* aplikasi yang berfokus pada penyajian kepada pengguna yang dituju.

3. Evaluasi *Prototyping*

Pada tahap ini proses evaluasi yang dilakukan oleh pengguna apakah *prototype* yang sudah disediakan sebelumnya sudah sesuai dengan tujuan pembuatan

aplikasi. Apabila pada proses evaluasi ini pengguna sudah berfikir aplikasi yang dibangun sesuai dengan tujuan dari pembuatan aplikasi tersebut maka proses bisa berlanjut ke proses selanjutnya.

4. Coding

Pada tahap ini adalah proses menerjemahkan dari hasil *prototype* kedalam sebuah kode program.

5. Pengujian aplikasi

Ketika aplikasi yang sudah dibangun telah siap digunakan maka terlebih dahulu harus dilaksanakan pengujian aplikasi sebelum berikan kepada pengguna. Metode yang digunakan untuk pengujian aplikasi ini adalah dengan menggunakan metode *black box*.

6. Evaluasi aplikasi

Pada tahap evaluasi ini, pengguna akan melakukan evaluasi aplikasi yang sudah dibangun sudah sesuai dengan yang diharapkan. Jika hasil dari evaluasi ini sudah sesuai maka aplikasi sudah dapat digunakan oleh pengguna.

1.7 Jadwal Pengerjaan

Tabel 1-1
Jadwal Pengerjaan

No	Kegiatan	Waktu Pelaksanaan																
		April 2014				Mei 2014				Juni 2014				Juli 2014				
		1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	
1	Pengumpulan Kebutuhan	■	■	■	■													
2	Membangun <i>Prototyping</i>					■	■											
3	Evaluasi <i>Prototype</i>						■	■										
4	<i>Coding</i>									■	■	■	■	■	■			
5	Pengujian Aplikasi													■	■			
6	Evaluasi Aplikasi															■	■	