

## RANCANG BANGUN APLIKASI UNTUK MERANCANG JARINGAN KOMPUTER

Agung Prihandono, Dwi Budi Santoso

Fakultas Teknologi Informasi, Universitas Stikubank Semarang

### Abstrak:

Perkembangan teknologi *jaringan komputer* melaju dengan sangat pesat. Berbagai bidang aktivitas mau tidak mau harus dipaksa untuk menggunakan teknologi ini karena kebutuhan dan persaingan. Disamping itu biaya pengembangan jaringan harus ditekan seminimal mungkin.

Aplikasi ini dibuat guna memenuhi kebutuhan akan sarana dalam merancang jaringan sebelum diterapkan pada jaringan sesungguhnya. Dalam perancangan dan pengembangan aplikasi ini, dipaparkan metode pengembangan sistem dengan menggunakan *uses case* sebagai teknik pemodelan sistem yang dilengkapi dengan dokumen skenario, yang kemudian dikembangkan menjadi sebuah aplikasi jadi dengan menggunakan Borland C++ Builder sebagai aplikasi pemrogramannya.

Aplikasi ini hanya digunakan untuk merancang jaringan komputer lokal atau LAN (*local area network*) dengan hub sebagai alat penghubung jaringan. Setelah rancangan jaringan dibuat, maka rancangan tersebut dapat diperiksa hubungan antar masing-masing objek komputer dengan komputer lain apakah telah saling terhubung atau belum dan dengan alamat IP dan nilai subnet yang sesuai.

**Kata Kunci:** jaringan komputer, use-case, local area network

### PENDAHULUAN

Sebuah komputer yang berdiri sendiri (*stand a lone*) mempunyai keterbatasan diantaranya, penggunaan bermacam-macam perangkat tambahan, baik perangkat keras maupun perangkat lunak, maka perangkat tambahan tersebut harus terhubung atau *ter-install* dengan komputer yang bersangkutan agar dapat memanfaatkan fasilitasnya. Dengan demikian, jumlah perangkat tambahan yang digunakan harus sebanding dengan jumlah komputer yang ada. Padahal, dalam sebuah ruangan dengan banyak komputer mungkin hanya dibutuhkan satu printer yang dapat dipakai bersama-sama secara bergantian.

Jaringan komputer merupakan model komputer tunggal yang melayani seluruh tugas-tugas komputasi suatu organisasi, telah diganti oleh sekumpulan komputer berjumlah banyak yang terpisah-pisah akan tetapi saling berhubungan dalam melaksanakan tugasnya atau suatu himpunan interkoneksi (*interconnected*) sejumlah komputer autonomus (Tanenbaum;1997:1).

Dalam perkembangannya, teknologi jaringan komputer melaju dengan sangat cepat. Berbagai bidang aktivitas tersebut mau tidak mau “*dipaksa*” menggunakan teknologi ini karena didesak oleh kebutuhan dan menyesuaikan dengan tuntutan masyarakat yang semakin mementingkan kecepatan dalam berproduksi maupun berkomunikasi. Keberadaannya harus mutlak ada dan dijadikan ujung tombak serta tulang punggung dalam melahirkan sumber daya manusia unggulan untuk menghadapi kompetisi global.

Ketertinggalan informasi dapat terjadi jika tidak mau memanfaatkan peluang yang terbuka dari perkembangan teknologi informasi dan telekomunikasi, baik sebagai pengelola maupun sebagai pengguna.

Penghubung antar komputer pun mengalami perubahan serupa. Mulai dari teknologi telegraf yang memanfaatkan gelombang radio hingga teknologi serat optik dan laser menjadi tumpuan perkembangan jaringan komputer.

Minimnya pengetahuan tentang aturan dan tata cara merancang sebuah jaringan komputer dapat menghambat dalam pengembangan jaringan komputer. Hal ini secara tidak langsung mengharuskan kita untuk mempelajari lebih jauh tentang bagaimana merancang sebuah jaringan komputer. Untuk mempelajarinya pun masih membutuhkan biaya yang tidak sedikit, terutama dalam pengadaan perangkat keras yang dibutuhkan dalam merancang jaringan komputer. Semakin besar fasilitas untuk pembelajaran jaringan yang akan dibangun, semakin besar pula biaya yang dikeluarkan.

Karena alasan tersebut kami akan merancang sebuah aplikasi untuk mendesain jaringan komputer. Dengan aplikasi ini diharapkan dapat membantu untuk memahami bagaimana aturan dan tata urutan dalam membangun sebuah jaringan komputer.

### PERUMUSAN MASALAH

Prinsip dasar dalam sistem jaringan adalah proses pengiriman data atau informasi

dari pengirim ke penerima melalui suatu media komunikasi tertentu. Tujuan dibangunnya suatu jaringan komputer adalah untuk membawa data-informasi dari sisi pengirim menuju penerima secara cepat & tepat tanpa adanya kesalahan melalui media transmisi atau media komunikasi tertentu.

Meskipun tujuannya sederhana, tetapi bukan berarti tidak ada masalah. Dalam hal ini, terdapat beberapa hal yang masih dirasa sebagai kendala, antara lain :

1. Masih mahalnya fasilitas komunikasi yang tersedia dan bagaimana cara memanfaatkan jaringan komunikasi yang ada secara efisien.
2. Jalur transmisi yang digunakan tidak benar-benar bebas dari masalah gangguan transmisi (*noise*).
3. Keterbatasan baik dari jumlah maupun kualitas SDM yang menguasai teknologi jaringan ini.
4. Keterbatasan dalam penyediaan modal investasi, karena untuk melakukan investasi teknologi dibutuhkan biaya yang tidak sedikit.

## PAPARAN SISTEM

Aplikasi untuk merancang jaringan komputer ini dapat digunakan untuk membantu dan memahami tentang cara merancang jaringan komputer sebelum diterapkan secara langsung pada perancangan yang sebenarnya.

Spesifikasi kebutuhan yang diharapkan pada aplikasi ini dibentuk dalam satu dokumen spesifikasi kebutuhan perangkat lunak (*software requirement specification*), terdiri dari :

1. User dapat memanipulasi file rancangan (SRS - ApliJarkom – 001)
2. User dapat menggambar rancangan jaringan komputer (SRS - ApliJarkom – 002)
3. User dapat melakukan konfigurasi IP Address pada rancangan (SRS - ApliJarkom – 003)
4. User dapat melakukan pengecekan rancangan apakah rancangan sudah benar atau belum (SRS - ApliJarkom – 004)
5. User dapat menggunakan fasilitas bantuan dalam melakukan perancangan (SRS - ApliJarkom – 005)
6. User dapat menggunakan fasilitas bantuan untuk mempelajari konsep jaringan komputer (SRS - ApliJarkom – 006)

Dari spesifikasi di atas, interaksi hanya terjadi antara pemakai dengan komputer (aplikasi ini) saja. Ketika pemakai membuka atau memulai menjalankan aplikasi, maka pemakai akan dihadapkan dengan tampilan form utama. Pada keadaan ini aplikasi siap dijalankan dan pemakai dapat memulai untuk memanipulasi dengan membuat file rancangan baru atau membuka file rancangan yang telah dibuat sebelumnya. Ketika pemakai memilih menu untuk membuat file rancangan baru maka aplikasi segera menampilkan satu form rancangan yang digunakan sebagai area untuk merancang jaringan komputer. Dan ketika pemakai memilih untuk membuka file rancangan yang sudah dibuat sebelumnya, maka aplikasi menampilkan form dialog untuk membuka file dan membuka rancangan file yang dibuka oleh pemakai.

Untuk memulai perancangan, pemakai memilih komponen berupa *icon* komputer dan hub pada toolbar yang terdapat dalam form rancangan. Setelah dipilih dan diklik pada salah satu *icon* tersebut, maka komponen tersebut akan ditampilkan dalam area rancangan berupa obyek yang dapat digerakkan menggunakan piranti *mouse*. Jumlah maksimal obyek komputer yang dapat digambar adalah 160 buah komputer. Dan obyek hub dibatasi hanya untuk 20 buah hub.

Obyek komputer dapat diberi konfigurasi alamat IP (*IP Address*) dan tujuan koneksi dengan obyek lain. Obyek hub hanya dapat dikonfigurasi tujuan koneksi dari masing-masing *port* yang disediakan. Agar obyek komputer dapat diberi konfigurasi *IP address*-nya, maka pada rancangan harus ada minimal 2 buah obyek komputer, dan harus sudah saling terhubung. Jika hanya terdapat satu buah obyek komputer dan pemakai ingin memberi konfigurasi *IP address*, maka aplikasi akan menampilkan pesan untuk menambah banyaknya obyek komputer.

Setelah masing-masing obyek saling terhubung, pemakai dapat melakukan pengecekan apakah hubungan masing-masing obyek tersebut telah berhasil melalui form konsol. Pemakai memilih salah satu obyek komputer kemudian memilih menu untuk membuka form konsol. Pada form konsol ini pemakai dapat menjalankan perintah *ping* yang berfungsi untuk mendeteksi apakah obyek komputer yang dituju telah terhubung sesuai rancangan dan konfigurasi alamat *IP Address*. Pemakai menuliskan perintah Ping diikuti dengan alamat IP atau nama komputer

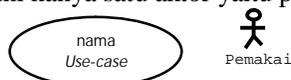
yang akan dituju sesuai dengan konfigurasi sebelumnya. Jika semua obyek komputer telah diberi alamat IP dan sudah saling terhubung, maka ketika pemakai menjalankan perintah “Ping IP\_Address” maka program ini akan menampilkan pesan bahwa koneksi telah berhasil.

Dalam proses perancangan tersebut pemakai dapat menggunakan fasilitas bantuan yang disertakan dalam aplikasi. Bantuan ini tidak bersifat interaktif dimana pemakai dapat mencari topik tertentu, tetapi hanya berbentuk dokumen biasa yang berisi petunjuk perancangan dan dasar teori jaringan komputer.

**ANALISA**

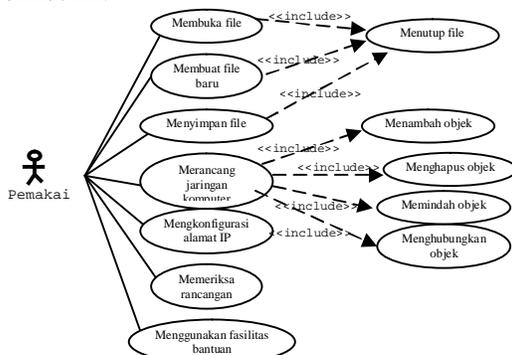
**Pemodelan Use Case**

Diagram use case digunakan untuk mendeskripsikan apa yang seharusnya dilakukan terhadap aplikasi/sistem. Diagram use-case menyediakan cara mendeskripsikan pandangan eksternal terhadap sistem dan interaksi-interaksinya dengan dunia luar. Pada pemodelan use-case, sistem dipandang sebagai kotak hitam yang menyediakan use-case. Bagaimana sistem melaksanakan sesuatu, bagaimana use-case diimplementasikan dan bagaimana kerja internalnya belum diperhatikan. Masing-masing diagram use-case menunjukkan sekumpulan use-case, aktor dan hubungannya. Aktor yang terlibat dengan aplikasi ini hanya satu aktor yaitu pemakai.



Gambar 1 use case dan aktor

Fungsionalitas aplikasi digambarkan dalam bentuk diagram use-case. Diagram use-case dari aplikasi ini ditunjukkan pada gambar berikut ini.



Gambar 2 Diagram Use Case

**Dokumen Skenario**

Skenario merupakan serangkaian langkah-langkah yang menggambarkan interaksi yang terjadi antara pemakai dengan sistem. Skenario bersifat satu kesatuan dengan use-case, dimana setiap use-case memiliki skenario yang berbeda. Karena skenario merupakan bentuk lain dari use-case yang dibuat dalam bentuk teks/narasi.

Dokumen skenario dimulai dengan membuat skenario utama sebagai main success scenario dalam bentuk urutan sejumlah tahapan/langkah, yang digunakan untuk menggambarkan sebuah langkah-langkah yang lengkap dari suatu kejadian dalam sebuah use-case. Kemudian membuat skenario tambahan (extensions) sebagai alternatif skenario yang mungkin dapat terjadi di dalam skenario utama. Skenario tambahan ini menjelaskan kemungkinan untuk berhasil dan kemungkinan gagal ketika pemakai menjalankan aksi tertentu dalam skenario utama. Ketika skenario menguraikan perilaku, skenario tetap terstruktur secara alami. Karena dalam dokumen skenario ini menyediakan struktur dalam serangkaian langkah-langkah yang dikombinasikan menjadi satu kesatuan.

Dari diagram use-case di atas dapat dibuat sebuah dokumen skenario sebagai berikut :

1. Use Case : *Membuka file*

Skenario :

- a. Pemakai membuka file yang telah dirancang sebelumnya
- b. Aplikasi menampilkan file rancangan
- c. Aplikasi mengaktifkan toolbar komponen

Kemungkinan 1 : File rancangan tidak ditemukan

- a. Pemakai menutup form dialog untuk membuka file
- b. Aplikasi kembali ke tampilan menu utama

Kemungkinan 2 : File rancangan ditemukan

- a. Kembali ke langkah ke b pada skenario utama

2. Use Case : *Membuat file baru*

Skenario :

- a. Pemakai membuat file baru
- b. Aplikasi menampilkan form area perancangan
- c. Aplikasi mengaktifkan toolbar komponen

Kemungkinan 1 : nama form area perancangan sudah ada

- a. Sistem meminta konfirmasi untuk menutup file sebelumnya
3. Use Case : *Menyimpan file*  
Skenario :
- Pemakai menyimpan file rancangan
  - Aplikasi menyimpan file rancangan
- Kemungkinan 1 : nama file sudah ada
- Aplikasi menampilkan pesan bahwa file dengan nama tersebut sudah ada
  - Aplikasi menampilkan pilihan untuk menumpuk file yang sudah ada atau menyimpan dengan nama baru
  - Aplikasi kembali ke langkah a pada skenario utama
- Kemungkinan 2 : nama file belum ada
- Aplikasi kembali ke langkah a pada skenario utama
4. Use Case : *Merancang jaringan komputer*  
Skenario :
- Pemakai memilih komponen pada area perancangan
  - Pemakai menggeser objek pada area perancangan
  - Aplikasi merekam data tentang letak komponen
5. Use Case : *Menambah objek*  
Skenario :
- Pemakai memilih icon objek yang diinginkan pada toolbar
  - Aplikasi merekam data tentang objek
- Kemungkinan 1: Penambahan objek melebihi batas maksimal
- Aplikasi menampilkan pesan bahwa banyaknya objek telah mencapai batas maksimal
  - Aplikasi mematikan fungsi dari icon untuk menambah objek yang ada pada masing-masing icon di toolbar
  - Aplikasi kembali ke langkah a pada skenario utama
- Kemungkinan 2 : Banyaknya objek belum melebihi batas maksimal
- Aplikasi kembali ke langkah ke a pada skenario utama
6. Use Case : *Menghapus objek*  
Skenario :
- Pemakai menghapus objek
  - Aplikasi merekam data tentang objek
7. Use Case : *Menggeser objek*  
Skenario :
- Pemakai memilih komponen dan memindahkannya dalam area perancangan
  - Aplikasi merekam data tentang objek
8. Use Case : *Menghubungkan objek*  
Skenario :
- Pemakai memilih objek pertama dan memilih menu untuk menghubungkan objek
  - Aplikasi menampilkan form konfigurasi port
  - Pemakai memilih/mengklik port kosong kemudian menutup form konfigurasi port
  - Aplikasi menampilkan/menggambar garis yang mengikuti gerakan mouse
  - Pemakai memilih objek lain yang ingin dihubungkan
  - Aplikasi menampilkan form konfigurasi port
  - Pemakai memilih/mengklik port kosong kemudian menutup form konfigurasi port
  - Aplikasi menggambar garis yang menghubungkan objek
  - Aplikasi merekam data tentang objek
- Kemungkinan 1 : Pemakai menghubungkan dengan objek yang sudah ada koneksi
- Aplikasi menolak untuk menghubungkan objek yang sudah ada koneksinya
  - Aplikasi mematikan port pada objek lain yang sudah ada koneksinya
  - Aplikasi kembali ke langkah a pada skenario utama
- Kemungkinan 2 : Objek lain belum ada koneksi
- Aplikasi kembali ke langkah e pada skenario utama
9. Use Case : *Mengkonfigurasi alamat IP*  
Skenario :
- Pemakai memilih objek komputer dan memilih menu konfigurasi
  - Aplikasi menampilkan form konfigurasi
  - Pemakai mengisi data konfigurasi
  - Aplikasi merekam data konfigurasi tentang objek
- Kemungkinan 1 : Alamat IP ada yang sama
- Aplikasi menampilkan pesan bahwa sudah ada alamat yang bersangkutan
  - Kembali ke langkah c pada skenario utama
- Kemungkinan 2 : Alamat IP belum digunakan

- a. Aplikasi kembali ke langkah a pada skenario utama
10. Use Case : *Memeriksa rancangan*  
Skenario :
- Pemakai memilih satu objek komputer dan memilih menu konsol untuk menguji hasil rancangan
  - Aplikasi menampilkan form konsol yang sesuai dengan objek yang dipilih
  - Pemakai menjalankan perintah “Ping xxx.xxx.xxx.xxx”
  - Aplikasi menampilkan pesan tentang keberadaan objek komputer yang memiliki alamat IP sesuai dengan alamat IP yang dimasukkan oleh pemakai
  - Pemakai menutup form konsol
- Kemungkinan 1 : Alamat IP tidak ditemukan
- Aplikasi menampilkan pesan bahwa alamat IP yang dituju belum terhubung dengan objek komputer
  - Kembali ke langkah c pada skenario utama
- Kemungkinan 2 : Alamat IP ditemukan
- Aplikasi menampilkan pesan bahwa objek komputer yang memiliki alamat IP tersebut telah terhubung dengan objek komputer pemanggil
  - Aplikasi kembali ke langkah a pada skenario utama
11. Use Case : *Menggunakan fasilitas bantuan*  
Skenario :
- Pemakai memilih menu bantuan dan membukanya
  - Aplikasi menampilkan form yang berisi dokumen bantuan
- Kemungkinan 1 : Pemakai memilih fasilitas bantuan untuk pengoperasian aplikasi
- Kembali ke langkah a pada skenario utama
- Kemungkinan 2 : Pemakai memilih fasilitas bantuan tentang teori dasar jaringan komputer
- Aplikasi kembali ke langkah a pada skenario utama
12. Use Case : *Menutup File*  
Skenario :
- Pemakai memilih menu untuk menutup file
  - Aplikasi menutup semua form aktif
- Kemungkinan 1 : File belum disimpan

- Aplikasi menampilkan konfirmasi untuk menyimpan file
  - Kembali ke langkah b pada skenario utama
- Kemungkinan 2 : File sudah disimpan
- Kembali ke langkah b pada skenario utama

### **Pemodelan Kelas**

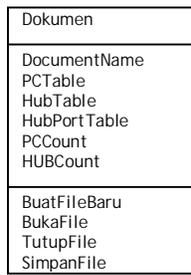
Karena interaksi yang terjadi hanya sistem dengan pemakai, di mana data pemakai tidak direkam oleh sistem, maka sistem hanya merekam data tentang rancangan saja yaitu objek yang digunakan untuk merepresentasikan komponen komputer dan hub, serta merekam dokumen rancangan.

Dokumen rancangan berisi objek-objek yang membentuk rancangan jaringan komputer. Dokumen dapat dilakukan beberapa operasi yaitu simpan, buka dan buat baru. Objek yang terdapat dalam rancangan terdiri dari objek komputer dan objek hub. Objek komputer dapat berisi data tentang posisi, ukuran, *picture*. Objek hub juga berisi data tentang posisi, ukuran dan *picture*. Operasi yang dapat dikenai pada objek komputer antara lain set alamat IP, set koneksi, dan konsol. Sedangkan operasi yang dapat dikenai pada objek hub hanya terdiri dari set koneksi saja.

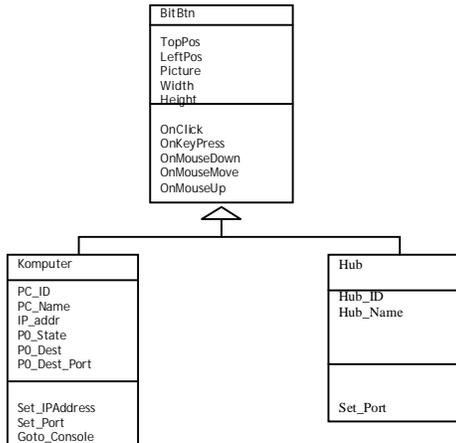
Dari deskripsi di atas dapat dibuat kelas dari objek-objek yang ada. Kelas tersebut terdiri dari kelas dokumen, kelas komputer, kelas hub, dan kelas hub\_port. Kelas dokumen berisi data tentang nama dari kelas-kelas yang lain.

Kelas komputer dan kelas hub merupakan penurunan sifat dari kelas *BitBtn*. *BitBtn* merupakan salah satu komponen yang disediakan dalam aplikasi pemrograman C++ Builder. Pada kelas *BitBtn* memiliki operasi *onclick*, *onkeypress*, *onmousedown*, *onmousemove*, dan *onmouseup*. Semua operasi pada kelas *bitbtn* merupakan interaksi secara langsung pemakai dengan aplikasi menggunakan peralatan input *mouse* dan *keyboard*.

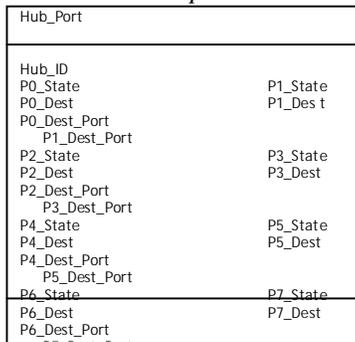
Kelas *hub\_port* digunakan untuk menampung informasi mengenai port-port yang ada pada sebuah hub. Informasi tersebut berupa status dari masing-masing port apakah sudah terpakai atau belum, jika telah terpakai maka akan direkam tujuan dari pemakai port tersebut.



Gambar 3 Kelas dokumen

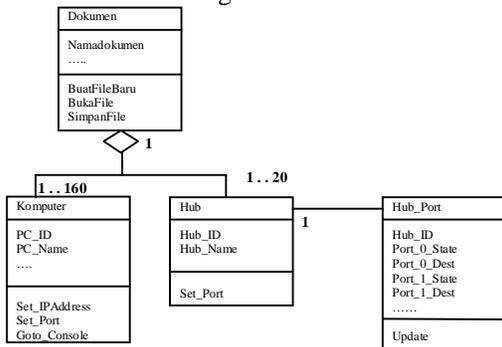


Gambar 4 Kelas komputer dan kelas hub



Gambar 3.5 Kelas Hub\_port

Kelas-kelas di atas dapat direlasikan sebagai berikut:



Gambar 6 Diagram kelas

Tabel yang akan direkam oleh aplikasi terdiri dari empat buah tabel. Yaitu tabel dokumen, komputer, hub, dan hub\_port.

Struktur dari tabel tersebut adalah sebagai berikut :

1. Tabel Dokumen

No	Fields	Type	Size	Specs	Keterangan
1.	DocumentName	String	50	['a'..'z', 'A'..'Z']	Primary key
2.	PCTable	String	20	['0'..'9']	
3.	HubTable	String	20	['a'..'z', 'A'..'Z']	
4.	HubPortTable	String	20	['0'..'9']	

Tabel 1 Tabel dokumen

2. Tabel Komputer

No	Fields	Type	Size	Specs	Keterangan
1.	PC_ID	String	5	['P', 'C', '0'..'9']	Primary key
2.	PC_Name	String	20	['a'..'z', 'A'..'Z']	
3.	IP_addr	String	15	['0'..'9']	
4.	Subnet_addr	String	15	['0'..'9', '.', ':']	
5.	Gateway	String	15	['0'..'9', '.', ':']	
6.	P0_State	Boolean	8	['0'..'9', ':']	
7.	P0_Dest	String	8	['0'..'9', ':']	
8.	P0_Dest_Port	String	2	[ True   False ]	
9.	TopPos	String		['a'..'z', 'A'..'Z']	
10.	LeftPos	Integer		['0'..'9']	
11.	VertPos	Integer		['0'..'9']	
12.	HorPos	Integer		['P', '0'..'7']	
13.	Width	Integer		[0..9]	
14.	Height	Integer		[0..9]	
15.	Glyph	String		[0..9]	

Tabel 2 Tabel Komputer

3. Tabel Hub

No	Fields	Type	Size	Specs	Keterangan
1.	Hub_ID	String	5	['H', 'U', 'B', '0'..'9']	Primary key
2.	Hub_Name	String	15	['a'..'z', 'A'..'Z']	
3.	TopPos	String		['a'..'z', 'A'..'Z']	
4.	LeftPos	String		['0'..'9']	
5.	Width	Integer		[0..9]	
6.	Height	Integer		[0..9]	
7.	Glyph	Integer		[0..9]	

Tabel 3 Tabel Hub

4. Tabel Hub\_port

No	Fields	Type	Size	Specs	Keterangan
1.	Hub_ID	String	5	['H', 'U', 'B', '0'..'9']	Primary key
2.	P0_State	Boolean	8	[ True   false ]	
3.	P0_Dest	String	8	['a'..'z', 'A'..'Z']	
4.	P0_Dest_Port	String	2	['0'..'9']	
5.	P1_State	Boolean	8	[ True   false ]	
6.	P1_Dest	String	8	['a'..'z', 'A'..'Z']	
7.	P1_Dest_Port	String	2	['0'..'9']	
8.	P2_State	Boolean	8	[ True   false ]	
9.	P2_Dest	String	8	['a'..'z', 'A'..'Z']	
10.	P2_Dest_Port	String	2	['0'..'9']	
11.	P3_State	Boolean	8	[ True   false ]	
12.	P3_Dest	String	8	['a'..'z', 'A'..'Z']	
13.	P3_Dest_Port	String	2	['0'..'9']	
14.	P4_State	Boolean	8	[ True   false ]	
15.	P4_Dest	String	8	['a'..'z', 'A'..'Z']	
16.	P4_Dest_Port	String	2	['0'..'9']	
17.	P5_State	Boolean	8	[ True   false ]	
18.	P5_Dest	String	8	['a'..'z', 'A'..'Z']	
19.	P5_Dest_Port	String	2	['0'..'9']	
20.	P6_State	Boolean	8	[ True   false ]	
21.	P6_Dest	String	8	['a'..'z', 'A'..'Z']	
22.	P6_Dest_Port	String	2	['0'..'9']	

Tabel 4 Tabel Hub\_port

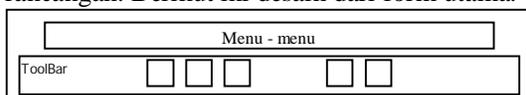
Dari tabel-tabel tersebut terdapat satu tabel yang akan digunakan sebagai file utama, yaitu tabel dokumen. Tabel ini digunakan dalam memanipulasi setiap file rancangan. Dimana hanya akan merekam satu record yang berisi data nama dari ketiga tabel lainnya.

**DESAIN**

Form yang digunakan dalam aplikasi ini terdiri dari form menu utama, form perancangan, form konfigurasi port untuk koneksi dengan objek komputer atau hub lain, form konfigurasi alamat IP, dan form konsol. Tampilan Form

a. Form Utama

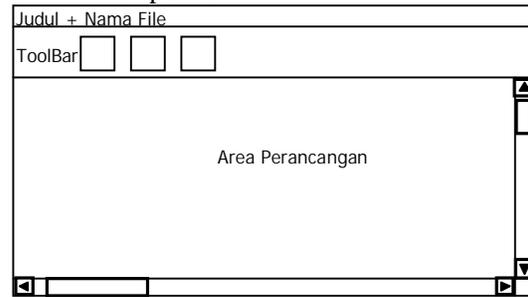
Form utama berisi menu-menu utama yang digunakan untuk menampilkan form-form lainnya seperti form perancangan dan form bantuan. Juga terdapat toolbar yang berisi icon-icon untuk memanipulasi file rancangan. Berikut ini desain dari form utama.



Gambar 7 Form utama

b. Form Perancangan

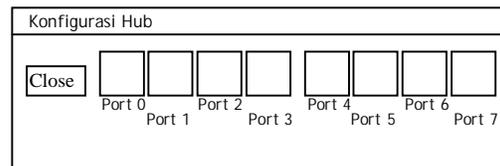
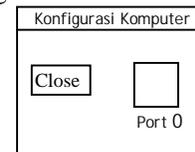
Form perancangan digunakan untuk membuat rancangan jaringan komputer.



Gambar 8 Form perancangan

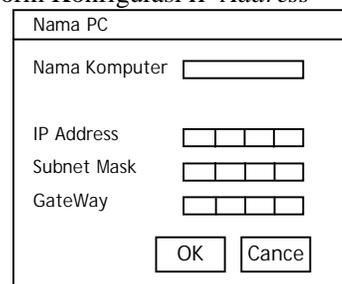
c. Form Konfigurasi

Form konfigurasi ini digunakan untuk mengatur koneksi dari objek komputer maupun objek hub dengan objek lain. Untuk menghubungkan dengan objek lain, pemakai hanya memilih dengan mengklik port yang masih kosong.



Gambar 9 Form konfigurasi koneksi

d. Form Konfigurasi IP Address



Gambar 30 Form konfigurasi alamat IP

e. Form Konsole

Form konsol ini digunakan untuk menguji koneksi masing-masing objek dalam rancangan jaringan. Melalui form ini pemakai dapat menguji rancangan jaringannya dengan menggunakan perintah ping. Perintah ini dalam sistem operasi digunakan untuk mengirim informasi permintaan ke alamat IP yang dituju. Jika alamat yang dituju sudah aktif dan terhubung dengan jaringan, maka

hasil dari perintah ini akan menampilkan pesan bahwa informasi yang dikirim telah berhasil sampai tujuan dan mendapat tanggapan.



Gambar 11 Form konsol

## ALGORITMA

Untuk dapat membangun sebuah aplikasi diperlukan algoritma-algoritma yang jelas agar memudahkan dalam penerjemahan ke dalam kode program. Algoritma dibuat berdasarkan fungsi-fungsi yang ditampilkan pada pemodelan *use case*.

### Algoritma Manipulasi File

Dalam memanipulasi file, aplikasi akan berjalan dengan dua jenis data. Pertama adalah data tabel penampung, data ini akan merekam setiap detail dari masing-masing objek komputer dan hub ketika melakukan perancangan, antara lain menampung posisi objek, keadaan port, port tujuan, alamat IP, dan lain sebagainya. Kedua adalah data tabel penyimpanan. Data inilah yang digunakan sebagai file rancangan yang dapat dimanipulasi. Proses manipulasi file terbagi dalam tiga bagian/fungsi utama yaitu fungsi membuat file baru, menyimpan file dan membuka file. Sehingga kita memerlukan tiga buah algoritma yaitu :

- a. Membuat file baru  
Algoritma ini diperlukan untuk menyiapkan suatu dokumen baru dengan menginisialisasi variable penampung yang bersifat sementara (*temporary*) sebelum data benar-benar ditulis di media penyimpan secara fisik.
- b. Menyimpan file  
Algoritma ini diperlukan untuk merekam semua data dari variable penampung sementara ke dalam bentuk file dokumen dan file tabel sehingga secara fisik dapat tersimpan di media penyimpanan
- c. Membuka file

### Algoritma Merancang Jaringan

Dalam merancang jaringan terdapat empat bagian/fungsi pokok yaitu :

- a. Menambah objek
- b. Menghapus objek
- c. Memindah objek

Dalam memindah objek aplikasi berinteraksi dengan perangkat mouse dimana terdapat tiga fungsi utama yaitu fungsi ketika tombol mouse ditekan (*onmousedown*), mouse digeser (*mousemove*) dan tombol mouse diangkat (*mouseup*). Algoritma tersebut adalah sebagai berikut:

1. OnMouseDown

Algoritma ini menangani kejadian ketika tombol mouse ditekan pada sebuah objek

2. OnMouseMove

Pada algoritma ini terdapat fungsi untuk *rubberbanding*, yaitu fungsi yang digunakan untuk menggambar garis yang seolah-olah mengikuti gerakan mouse. Serta fungsi *onmousemove*.

3. OnMouseUp

Algoritma ini menangani kejadian ketika mouse tombol mouse diangkat dari sebuah objek

- d. Menghubungkan objek

Dalam fungsi ini terbagi dalam tiga fungsi yaitu fungsi pemanggil, fungsi objek PC, dan fungsi objek HUB.

1. Algoritma fungsi pemanggil
2. Algoritma fungsi objek PC
3. Algoritma fungsi objek HUB

### Algoritma Mengkonfigurasi Alamat IP

Algoritma dalam fungsi ini adalah untuk menangani pemberian kombinasi angka sebagai alamat IP pada objek PC. Dimana akan membatasi jumlah dan banyaknya angka yang dapat diisi sesuai dengan aturan pemberian alamat IP beserta *w masknya*

### Algoritma Memeriksa Rancangan

Dalam memeriksa rancangan, menggunakan konsep perintah *ping* pada *command prompt* di *windows*. Algoritma dilakukan dengan melacak satu persatu port tujuan dari objek PC dan HUB.

### Algoritma pada Form Konsol

Pada form konsol terdapat algoritma yang digunakan untuk memeriksa dan mengeksekusi perintah-perintah internal. Proses untuk mengetahui jenis perintah

tersebut menggunakan cara membaca karakter yang diketik pada sebuah komponen *memo* dan memisahkannya berdasarkan karakter spasi kosong. Setelah itu karakter disimpan ke dalam variabel untuk kemudian diperiksa jenis perintah yang akan dieksekusi.

## PENUTUP

### Simpulan

Berdasarkan analisis dan perancangan sehingga menghasilkan sebuah aplikasi untuk merancang jaringan komputer, maka diperoleh suatu kesimpulan sebagai berikut :

1. Membangun suatu jaringan komputer membutuhkan pengetahuan khusus mengenai jaringan karena harus mempertimbangkan banyak aspek, agar tidak terjadi pemborosan sumber daya.
2. Aplikasi ini dapat membantu dalam merancang sebuah jaringan komputer khususnya jaringan lokal atau LAN yang hanya menggunakan hub sebagai penghubung antar komputer.
3. Setiap objek komputer maupun objek hub yang digunakan pada perancangan dapat ditambahkan sesuai keinginan pengguna karena objek tersebut dibuat secara dinamis dan merupakan turunan dari komponen *BitButton* yang ada pada C++ Builder.
4. Aplikasi tidak dapat membuka file rancangan yang sudah disimpan dan atau membuat file rancangan baru apabila pada aplikasi tersebut form rancangannya sedang aktif dan sudah terdapat objek komputer maupun objek hub di dalamnya. Jadi agar dapat membuka file atau membuat file rancangan baru, aplikasi harus ditutup terlebih dahulu. Hal ini disebabkan karena ketika membersihkan form perancangan dari rancangan yang berisi objek-objek komputer maupun hub, aplikasi hanya membersihkan data yang ada pada tabel penampung yang merekam setiap detail objek-objek tersebut tetapi tidak menghapus komponen yang diciptakan secara dinamis sebagai objek-objek komputer dan hub.
5. Aplikasi ini dilengkapi dengan adanya form konsol, di mana cara kerjanya menyerupai *command prompt* pada sistem operasi *windows* yang dapat digunakan untuk memeriksa hubungan antar komputer dalam suatu jaringan komputer, yaitu dengan perintah *ping*.

6. Proses pemeriksaan dengan perintah *ping* dilakukan dengan membaca secara langsung pada tabel dari masing-masing objek komputer dan hub yang merekam data-data mengenai keadaan port, tujuan port, alamat IP, dan nilai subnet.
7. Pengujian aplikasi dilakukan dengan dua rancangan. Untuk rancangan pertama pengujian didasarkan pada hubungan antar komputer secara fisik tanpa dipengaruhi oleh nilai subnet pada masing-masing alamat IP, artinya karena tidak diberi nilai subnet maka aplikasi menganggap nilai subnet sama dengan kosong sehingga masing-masing objek komputer memiliki daya tampung alamat host yang sama. Hasil dari pengujian ini telah sesuai dengan uji coba pada jaringan komputer yang sesungguhnya.
8. Pengujian pada rancangan kedua dilakukan berdasarkan hubungan secara fisik dan juga menghitung kombinasi dalam pemberian alamat IP dan nilai subnet. Dengan kombinasi ini memungkinkan sebuah objek komputer memiliki jangkauan alamat host dan daya tampung alamat host yang berbeda dengan objek komputer yang lain. Sehingga hasil dari pengujian ini, seperti yang ditunjukkan pada bab sebelumnya, telah sesuai dengan uji coba pada jaringan komputer yang sesungguhnya.

### Saran

Saran-saran yang diusulkan untuk pengembangan aplikasi ini adalah

1. Untuk pengembangan lebih lanjut, aplikasi dapat dikembangkan agar dapat digunakan untuk jaringan antar LAN dengan media penghubung jaringan tidak hanya menggunakan hub, misalnya menggunakan *switch* dan *router*.
2. Sebaiknya pengembang dapat menggunakan alternatif yang lebih baik dalam membuat objek komputer dan objek hub. Karena objek tersebut merupakan turunan dari komponen yang ada pada C++ Builder, mengakibatkan terciptanya sebuah komponen baru yang belum dapat diakses agar ketika form rancangan dibersihkan dari objek-objek komputer dan hub, komponen dinamis tersebut dapat terhapus secara otomatis, misalnya tidak menggunakan turunan dari sebuah komponen seperti membuat objek sendiri

- dan menampilkannya sebagai *image* yang langsung digambar pada *canvas*.
3. Perintah-perintah yang ada pada form konsol dapat dikembangkan agar dapat menghasilkan keluaran sesuai dengan maksud dari perintah yang bersangkutan.
  4. Untuk pengembangan lebih lanjut, aplikasi dapat mensimulasikan proses pemindahan data pada masing-masing objek komputer sesuai dengan jalur komunikasi yang dirancang dan pemetaan alamat IP yang sesuai, misalnya proses mengkopir file dari satu komputer ke komputer lain.

#### DAFTAR PUSTAKA

- Boggs, Wendy, Michael, 2002. *Mastering UML with Rational Rose 2002*. E-Book, Sybex
- Booch, Rumbaugh, Jacobson, 1998. *The Unified Modeling Language User Guide*. E-Book, Addison-Wesley Longman Inc, Massachusetts
- Bramble, Paul, Adolph, Steve. *Patterns For Effective Use Cases*, E-Book
- Fowler, Martin, 1997. *UML Distilled Third Edition*. E-Book.
- Garland, Jeff, Richard, Anthony, 2003. *Large-Scale Software Architecture*. E-Book. John Wiley & Sons Ltd, England
- Heriyanto, Bambang, 2004. *Rekayasa Sistem Berorientasi Objek*. Informatika, Bandung
- Pressman, Roger S, 2002. *Rekayasa Perangkat Lunak Pendekatan Praktisi*, Buku 1 Terjemahan. Andi, Yogyakarta
- Mansfield, Niall, 2004. *Protokol TCP/IP Mendesain Menggunakan dan Troubleshooting Jaringan TCP/IP di Linux dan Windows*, Jilid 1. Andi, Yogyakarta
- Nugroho, Adi, 2004. *Pemrograman Berorientasi Objek*. Informatika, Bandung
- Oetomo, Budi Soetejo, 2003. *Konsep & Perancangan Jaringan Komputer*. Penerbit Andi, Yogyakarta
- Pender, Thomas A, 2002. *UML Weekend Crash Course*™. E-Book. Wiley Publishing, Inc, Indiana
- Pramono, Joko, 1999. *Mudah Menguasai C++ Builder*, Jilid 2. PT Elex Media Komputindo, Jakarta
- Raharjo, Budi, Heryanto, Imam, 2003. *Pemrograman C++ Builder, dilengkapi dengan studi kasus*. Informatika, Bandung
- Rubbin, David M, 1998, *Uses Of Use Case*. E-Book Softstar Research Inc
- Setiawan, C Yudha, 2004. *Trik & Tip Delphi*. Penerbit Andi, Yogyakarta
- Stallings, William, 2002. *Komunikasi Data dan Komputer Jaringan Komputer*. Salemba Teknika, Jakarta
- Tanenbaum, Andrew S, 1997. *Jaringan Komputer, Edisi 3*. Prenhallindo, Jakarta
- Wagito, 2005. *Jaringan Komputer Teori dan Implementasi Berbasis Linux*. Gava Media, Yogyakarta
- Wijaya, Hendra, 2000. *Cisco Router*. PT Elex Media Komputindo, Jakarta