

APLIKASI FILE TRANSFER PROTOCOL BERBASIS TCP/IP

LAPORAN TUGAS AKHIR

Diajukan untuk Memenuhi Salah Satu Syarat
dalam Menempuh Ujian Sidang Sarjana
di Jurusan Teknik Informatika

Oleh :

Andreas Sinaga

06.01.058



**FAKULTAS TEKNIK
JURUSAN TEKNIK INFORMATIKA
UNIVERSITAS WIDYATAMA BANDUNG
2005 / 2006**

SURAT PERNYATAAN

Saya yang bertanda tangan dibawah ini :

Nama : Andreas Sinaga
Tempat dan Tanggal Lahir : Medan / 25 Januari 1980
Alamat Orang Tua : Jl. Jermal I Gg. Setiabudi No. 5 Medan

Menyatakan bahwa skripsi ini adalah benar dan hasil karya sendiri. Bila terbukti tidak demikian, saya bersedia menerima segala akibatnya, termasuk pencabutan kembali gelar Sarjana Teknik yang telah saya peroleh.

Bandung,

Andreas Sinaga

Abstraksi

Aplikasi Client Server berbasis TCP/IP ini adalah suatu sistem aplikasi transfer *file* yang menggunakan protokol TCP/IP sebagai penghubung komunikasi antar *client* dan *server*.

Pembangunan Aplikasi Client Server berbasis TCP/IP ini bertujuan untuk memudahkan pekerjaan manajemen file pada suatu server. Manajemen file pada suatu server umumnya dilakukan pada sebuah *website* atau *remote computer*.

Aplikasi ini dibangun dengan metode waterfall dan menggunakan bahasa pemrograman Borland Delphi 7 dan Cerberus FTP Server 2.32 sebagai aplikasi server.

Metode yang digunakan untuk analisis dan perancangan sistem adalah metode Data Flow Oriented dengan memakai *tool* Data Flow Diagram (DFD).

Kata Kunci :

FTP, download, upload.

Abstraction

The Application Of Client Server base on this TCP/IP is a system application of transfer file using protocol TCP/IP as communications link between client and server.

Development Of The Application Of Client Server base on this TCP/IP aim to facilitate work of file management at one particular server. File management at one particular server generally done at a website or remote computer.

The application built up with method waterfall and using programming language Borland Delphi 7 and Cerberus FTP Server 2.32 as the application of server.

Method which applied for analysis and scheme of system is method Data Flow Oriented by using Data Flow Diagram (DFD) tool.

Keywords :

FTP, download, upload.

KATA PENGANTAR

Segala puji syukur bagi Tuhan Yang Maha Esa atas berkat dan karuniaNya sehingga penulis dapat menyelesaikan laporan Tugas Akhir ini. Pertolongan dan penjagaanNya selalu menyertai bagi orang yang berserah senantiasa kepadaNya.

BAB I

PENDAHULUAN

1.1 Latar Belakang Masalah

Penggabungan kedua teknologi antara teknologi komputer dan teknologi komunikasi akan berpengaruh terhadap bentuk dari organisasi sistem komputer. Konsep “pusat komputer” dimana dalam sebuah ruangan terdapat sebuah komputer besar, yang dulu dipakai seluruh pemakai secara bersamaan untuk mengolah perkerjaannya, sudah tidak relevan lagi dengan perkembangan teknologi komputer di zaman sekarang ini. Model komputer tunggal yang bertugas melayani proses komputasi suatu organisasi sudah digantikan dengan sebuah kumpulan komputer, printer dan peralatan lainnya tetapi masih saling berhubungan dalam melaksanakan tugasnya. Sistem ini disebut dengan jaringan komputer.

Komunikasi data merupakan proses pengiriman informasi diantara dua titik menggunakan kode biner melalui suatu aliran transmisi dan peralatan *switching*, bisa antar komputer dan komputer, komputer dengan terminal, komputer dengan peralatan, peralatan dengan peralatan. Informasi bergerak melalui kabel ataupun tanpa kabel sehingga memungkinkan pengguna jaringan komputer dapat saling bertukar data dan dokumen, mencetak pada *printer* yang sama, dan bersama-sama dalam menggunakan *hardware/software* yang terhubung dalam sebuah jaringan.

Dewasa ini, komputer dan perangkat pendukungnya telah menjadi alat komunikasi yang cukup efisien dengan segala kemampuan pengolahan data yang paling luas dalam penggunaannya, maka tidak ada bentuk informasi yang tidak dapat dikerjakan dan ditangani oleh komputer dan perangkat pendukungnya.

Oleh karena itu sarana komunikasi sekarang sudah mengarah kepada komputer sebagai alat pendukung dan memegang peranan penting dalam hal komunikasi data.

Manusia dalam berkomunikasi antar sesamanya, sering terjadi kedua pihak baik pengirim maupun penerima berita tidak mengerti informasi yang disampaikan. Salah satu alasan utamanya adalah ketidaksamaan bahasa yang digunakan diantara mereka. Agar keduanya dapat memahami informasi yang disampaikan, maka diperlukan bahasa yang dapat dipahami oleh kedua belah pihak, atau dengan kata lain harus ada aturan yang jelas dan disepakati untuk dapat berkomunikasi. Komunikasi antar mesin/komputer pun demikian pula, apabila komputer/mesin tersebut merupakan produk dari berbagai pabrik, oleh karena itu diperlukan suatu aturan agar pengirim dan penerima mengerti informasi yang dikirim, jadi dalam komunikasi data juga memerlukan sebuah peraturan atau prosedur yang saling menterjemahkan bahasa yang dipakai pengirim dan penerima. Aturan itu adalah protokol, yaitu suatu kumpulan dari aturan-aturan yang berhubungan dengan komunikasi data agar komunikasi data dapat dilakukan dengan benar. Protokol pada dasarnya, adalah sebuah persetujuan semua pihak yang berkomunikasi tentang bagaimana komunikasi tersebut harus dilakukan.

Salah satu model protokol adalah TCP/IP, kumpulan dari protokol yang memungkinkan komunikasi antar jaringan. TCP/IP menetapkan bagaimana proses komunikasi jaringan dan bagaimana seharusnya bentuk suatu data, agar komputer yang menerima dapat menerima informasi dengan benar. Pada model protokol TCP/IP terdapat lapisan-lapisan yang mempunyai fungsi masing-masing. Salah satunya adalah lapisan *Application*. Fungsi dari lapisan *Application* adalah melayani permintaan pemakai untuk mengirim dan menerima data. Data ini kemudian disampaikan ke layer Transport untuk proses lebih lanjut. Salah satu protokol yang terdapat dalam lapisan *Application* adalah FTP (*File Transfer Protocol*).

FTP (*File Transfer Protocol*) merupakan layanan yang paling banyak digunakan oleh pemakai Internet untuk mengambil atau meletakkan file dari atau ke mesin lain di

Internet. FTP sama seperti *login* ke suatu mesin, tapi disini user dibatasi pada sejumlah *command* tertentu, dan untuk anonymous FTP user, dibatasi hanya pada direktori tertentu. Arsip yang di-set agar dapat diakses oleh publik disebut arsip anonymous FTP, karena sembarang orang dapat *login* (hanya alamat e-mail orang tersebut diminta sebagai identifikasi). FTP client terdapat pada hampir semua jenis komputer, dari *palmtop* sampai *supercomputer*. Setelah e-mail, FTP adalah layanan Internet yang paling umum digunakan.

1.2 Identifikasi Masalah

Identifikasi masalah yang diambil dalam penulisan skripsi ini mengenai pembuatan aplikasi *client-server* untuk pengiriman data antar dua komputer (*Client-Server*) dengan berbasis protokol TCP/IP.

1.3 Batasan Masalah

Dalam tugas akhir ini penelitian dibatasi pada pembuatan aplikasi *client-server* dengan melakukan pendekatan pada *layer transport* pada protokol TCP/IP. Karena aplikasi *client-server* ini cukup luas, agar tidak terjadi penyimpangan tujuan, diperlukan pembatasan masalah, yaitu :

- a. Pembuatan aplikasi *client-server* yang dibuat adalah FTP (*File Transfer Protocol*).
- b. Dalam melakukan koneksi FTP (*File Transfer Protocol*), *user* dapat *login* ke *account* dan dapat melakukan *anonymous login* atau sebagai *guest* (tamu) di *server* FTP (*File Transport Protocol*) untuk mendapatkan hak akses.
- c. Tahapan-tahapan pengembangan perangkat lunak secara umum adalah analisis kebutuhan sistem, perancangan, pengkodean, pengujian serta pemeliharaan.

1.4 Tujuan Penelitian

Kegiatan penelitian ini bertujuan untuk :

1. Membuat aplikasi *client-server* pengiriman data berbasis protokol TCP/IP.
2. Agar dalam pengiriman data, data yang dikirim sama dengan data yang diterima.

1.5 Metodologi Penelitian

Penelitian dilaksanakan di Laboratorium Sistem Jaringan Komputer, Jurusan Teknik Informatika, Fakultas Teknik, Universitas Widyatama Bandung.

Dalam melaksanakan penelitian, digunakan metoda deskriptif yaitu metoda penelitian yang bertujuan untuk membuat sebuah gambaran yang sistematis dan objektif dengan cara pengumpulan data yang bertujuan untuk menarik suatu kesimpulan yang benar.

Dalam melakukan pengumpulan data digunakan metode antara lain :

1. Studi Literatur

Yaitu teknik pengumpulan data dengan cara mengumpulkan literatur, jurnal, *white paper*, *text book* dan bacaan yang ada kaitannya dengan judul penelitian.

2. Observasi

Melakukan teknik pembuatan *Local Area Network (LAN)* secara langsung.

-
3. Alat yang diperlukan dalam pembuatan aplikasi client-sever pengiriman data melalui protokol TCP/IP adalah sebagai berikut :
 - a. Perangkat Keras
 1. Kabel UTP
 2. 2 (dua) buah LAN *Card*
 3. 2 (dua) buah *Personal Computer*
 - b. Perangkat Lunak
 1. Sistem Operasi *Windows XP*
 2. Protokol TCP/IP
 3. Borland Delphi 7.0 untuk pembuatan aplikasi *Client-Server*

1.6 Sistematika Penulisan

Dalam penyusunan laporan tugas akhir ini, dibuat suatu kerangka pembahasan dalam bentuk bab per bab sebagai berikut :

Bab I Pendahuluan, berisi latar belakang masalah, indentifikasi masalah, batasan masalah, tujuan penelitian, metodologi penelitian dan sistematika penulisan dalam pembangunan aplikasi *client-server*.

Bab II Landasan Teori, yang menjelaskan teori-teori yang berhubungan dengan perangkat lunak yang akan dibangun dan hal-hal yang terkait dalam pembangunan aplikasi *client-server*.

Bab III Analisis Sistem, melihat prinsip kerja dari sistem, melakukan analisis aplikasi *client-server* yang dibuat.

Bab IV Perancangan Sistem, berisi perancangan proses yang akan menjadi modul-modul sebagai dasar dari kode program aplikasi *client-server* yang akan dibuat.

Bab V Implementasi Sistem, akan memperlihatkan bagaimana rancangan yang telah ada akan dibuat menjadi tabel-tabel, kode program maupun antarmuka yang dibutuhkan.

Bab VI Penutup, berisi kesimpulan dan saran dari hasil pembangunan aplikasi *client-server* agar dapat diperbaiki dikemudian hari.

BAB II

LANDASAN TEORI

2.1 Jaringan Komputer

Jaringan komputer bukanlah suatu hal yang baru lagi dalam dunia teknologi informasi. Hampir di semua perusahaan dan instansi memanfaatkan jaringan komputer untuk mempercepat dan melancarkan laju arus informasi yang sangat diperlukan dalam perusahaan atau instansi tersebut. *Internet* yang sudah sangat populer pada masa sekarang ini adalah suatu jaringan komputer raksasa yang merupakan jaringan komputer yang saling terhubung dan saling berinteraksi satu sama lainnya. Hal ini terjadi karena pesatnya perkembangan teknologi informasi dan jaringan, sehingga dalam beberapa tahun saja jumlah pengguna jaringan komputer yang tergabung dalam *internet* meningkat dengan angka yang cukup tinggi.

Jaringan komputer adalah kumpulan komputer dan peralatan lainnya seperti *printer*, *scanner* dan yang lainnya yang saling terhubung. Jaringan komputer melakukan hubungan untuk melakukan pemakaian bersama atas sumber daya yang ada pada jaringan komputer tersebut. Pemakaian bersama dalam sumber daya termasuk di dalamnya adalah *file*, *printer*, *scanner* dan perangkat lainnya yang terhubung dalam jaringan. Pemakai dapat mencetak pada satu atau lebih *printer* yang sama, mengakses suatu *file* yang terdapat pada komputer lain dan memakai *software* atau *hardware* yang dapat dipakai bersama-sama, tanpa harus mengakses secara fisik ke komputer yang memiliki sumber daya. Tiap-tiap komputer, *printer*, *scanner* dan peralatan lainnya yang terhubung dalam jaringan disebut *node*. Dan setiap jaringan komputer memiliki dua, ratusan, ribuan bahkan jutaan *node*.

Dalam sebuah jaringan komputer biasanya terdiri dari dua atau lebih komputer yang saling berhubungan satu sama lain dan saling berbagi sumber daya, misalnya *CD-ROM*, *printer*, *scanner*, pertukaran file bahkan berkomunikasi secara elektronik. Komputer

yang terhubung, dimungkinkan berhubungan dengan media kabel, saluran telepon, gelombang radio, satelit, sinar infra merah, atau tanpa kabel.

Jaringan komputer memiliki beberapa manfaat dan keuntungan, antara lain :

- a. Berbagi *file* : Dengan memanfaatkan jaringan komputer, berbagai informasi dapat dibagikan dan di akses secara bersama-sama.
- b. Berbagi sumber daya : Dengan memanfaatkan jaringan komputer, kita dapat berbagi peralatan yang berhubungan dengan komputer misalnya dalam sebuah kantor yang kecil hanya memerlukan satu *printer* untuk dipakai secara bersama-sama dalam jaringan.
- c. Berbagi aplikasi : Dengan memanfaatkan jaringan komputer, kita dapat berbagi aplikasi dengan pemakai lainnya tanpa harus memasang aplikasi tersebut pada tiap komputer.

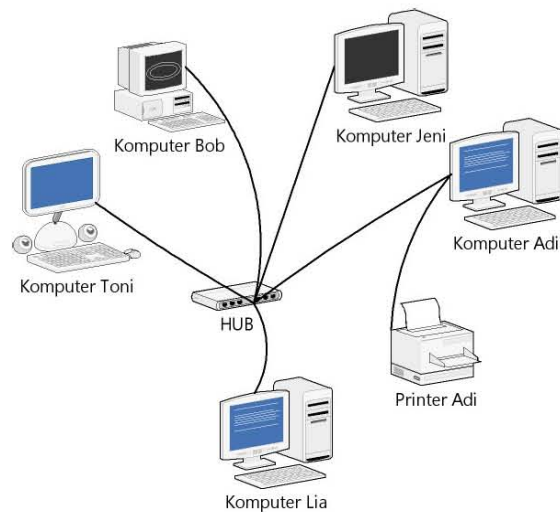
Keuntungan lainnya dari jaringan komputer adalah kita dapat melakukan komunikasi dengan menggunakan berbagai aplikasi yang dibuat khusus untuk komunikasi, bahkan dengan menggunakan jaringan komputer kita dapat melakukan pertemuan atau konferensi melalui jaringan komputer.

2.1.1 Tipe-Tipe Jaringan Komputer

Secara garis besar terdapat dua tipe jaringan, yaitu jaringan *Peer to Peer* dan jaringan *Client-Server*. Pada jaringan *peer to peer*, setiap komputer yang terhubung ke jaringan dapat bertindak baik sebagai *workstation* maupun *server*. Sedangkan pada jaringan *Client-Server*, hanya satu komputer yang bertugas sebagai *server* dan komputer lain berperan sebagai *workstation*. Kedua tipe jaringan tersebut memiliki keunggulan dan kelemahannya masing-masing.

2.1.1.1 Jaringan *Peer to Peer*

Jaringan *peer to peer* adalah tipe paling sederhana dari jaringan komputer. Dalam jaringan *peer to peer* setiap komputer dapat berkomunikasi langsung dengan setiap komputer. Pada dasarnya, tidak ada komputer yang memiliki hak akses lebih dari yang lainnya. Tetapi setiap komputer dapat dikonfigurasi untuk dapat memberikan akses terhadap beberapa sumber dan beberapa sumber lainnya tetap tidak dapat diakses di jaringan. Jaringan komputer *peer to peer* biasanya terdiri atas dua atau lebih komputer pribadi (*personal computer*). Setiap komputer dapat saling mengirim dan menerima informasi dari dan ke komputer lainnya.



Gambar 2.1 Jaringan *Peer to Peer*

Keuntungan dari tipe jaringan *peer to peer* adalah :

- ✦ Konfigurasi sederhana. Untuk hal ini, jaringan *peer to peer* dapat digunakan pada lingkungan yang tidak membutuhkan seorang ahli jaringan.
- ✦ Murah dalam pemasangan dan pemeliharaan jaringan. Cocok untuk lingkungan yang membutuhkan sarana jaringan komputer yang murah.

Kelemahan dari jaringan *peer to peer* adalah :

- ✦ Sulit dalam penyesuaian, apabila jaringan bertambah besar penambahan beberapa elemen yang diperlukan menjadi lebih sulit.
- ✦ Kurang aman, kemudahan dalam pemasangan menyebabkan data dan sumber daya yang lainnya menjadi mudah ditemukan dan digunakan oleh pemakai yang tidak diizinkan.

Sebagai contoh, jika sebuah komputer adalah bagian dari jaringan *peer to peer* dan ada komputer yang lain menjadi bagian jaringan tersebut, apabila komputer yang lain ingin menggunakan data yang ada di komputer lainnya maka komputer tersebut harus mengakses data melalui jaringan. Jika seseorang mengubah data, maka akan sulit melihat data yang akan digunakan pada saat ini. Dapat dibayangkan, makin banyak komputer yang ditambah pada jaringan *peer to peer* maka akan sulit pula dalam hal penanganan dan perawatan.

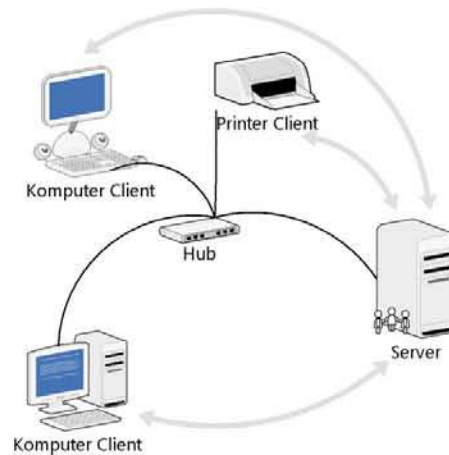
Walaupun jaringan *peer to peer* biasanya kecil, sehingga penggunaannya terbatas pada rumah ataupun kantor yang kecil, dalam lima tahun terakhir, jaringan *peer to peer* sudah tersambung melalui internet. Tipe jaringan *peer to peer* ini disebut P2P (*person to person*), menghubungkan komputer dari seluruh dunia untuk berbagi data antar *hard disk* pengguna. Tidak seperti jaringan *peer to peer* yang biasa, jaringan ini membutuhkan *software* khusus untuk berbagi data.

2.1.1.2 Jaringan Client Server

Tipe lain dalam pembuatan jaringan adalah dengan menggunakan komputer pusat, sebagai sentral komunikasi dan pembagian sumber daya antar komputer dalam jaringan. Bentuk dari komputer *client* adalah komputer pribadi, atau dalam kata lain disebut juga *workstation*. Jaringan yang menggunakan komputer *server* untuk memberikan akses untuk berbagi data, penyimpanan data, dan pembagian penggunaan perangkat dikenal sebagai jaringan *client/server*. Istilah jaringan *client/server* biasanya digunakan untuk merancang

sebuah jaringan dimana *client* tergantung pada *server* dalam komunikasi dan pembagian sumber daya yang ada.

Setiap komputer yang terdapat dalam jaringan *client/server* bertindak sebagai *client* atau sebagai *server*. *Client* di dalam jaringan masih dapat menjalankan aplikasi pada komputer *client* itu sendiri. Tetapi dengan terhubung pada *server*, *client* menjadi mempunyai pilihan dalam berbagi data, aplikasi, dan perangkat lainnya. *Client* dalam jaringan *client/server* tidak memberikan hak akses langsung pada pengguna lainnya, tetapi menggunakan *server* sebagai perantara hubungan. Gambar dibawah akan menjelaskan bagaimana pembagian data, dan perangkat lainnya.



Gambar 2.2 Pembagian sumber daya dalam jaringan *client/server*

Agar dapat berfungsi sebagai *server*, sebuah komputer harus dijalankan pada sistem operasi jaringan atau *Network System Operation* (NOS), aplikasi yang dirancang untuk :

- ✦ Mengatur pembagian data dan perangkat lainnya
- ✦ Mengatur hak akses pengguna
- ✦ Mengatur data mana saja yang dapat dibuka dan dilihat
- ✦ Membatasi pengguna yang dapat mengakses jaringan
- ✦ Mengatur komunikasi
- ✦ Menyediakan aplikasi bagi pengguna

Umumnya, sebuah komputer yang dipakai untuk *server* memiliki memori yang besar, *processor* yang berkecepatan tinggi dan media penyimpanan yang berkapasitas besar. Walaupun jaringan *client/server* lebih rumit dalam perancangan dan pemeliharaan dibandingkan jaringan *peer to peer*, jaringan *client/server* memberikan banyak keuntungan, antara lain :

- ✦ *Login* pengguna pada jaringan komputer berbasis *server* dapat disimpan pada satu *server*.
- ✦ Akses pada suatu kelompok sumber daya (seperti data dan printer) dapat diberikan secara terpusat pada perorangan atau grup pengguna.
- ✦ Masalah pada jaringan dapat ditelusuri, didiagnosis, dan dapat diperbaiki dari satu tempat.
- ✦ *Server* dioptimalkan untuk menangani proses yang berat dan bekerja untuk menangani permintaan *client*.

Kesemua keuntungan diatas membuat jaringan *client/server* lebih mudah dalam perawatan, lebih aman dibandingkan dengan jaringan *peer to peer*. Lebih mudah apabila ada penambahan komputer dan perluasan jaringan.

2.1.2 Local Area Network

LAN dapat didefinisikan sebagai *network* atau jaringan sejumlah sistem komputer yang lokasinya terbatas didalam satu gedung, satu kompleks gedung atau suatu kampus dan tidak menggunakan media fasilitas komunikasi umum seperti telepon, melainkan pemilik dan pengelola media komunikasinya adalah pemilik LAN itu sendiri. Dari definisi diatas dapat kita ketahui bahwa sebuah LAN dibatasi oleh lokasi secara fisik. Adapun penggunaan LAN itu sendiri mengakibatkan semua komputer yang terhubung dalam jaringan dapat bertukar data atau dengan kata lain berhubungan. Kerjasama ini semakin berkembang dari hanya pertukaran data hingga penggunaan peralatan secara bersama. LAN yang umumnya menggunakan *hub*, akan mengikuti prinsip

kerja *hub* itu sendiri. Dalam hal ini adalah bahwa *hub* tidak memiliki pengetahuan tentang alamat tujuan sehingga penyampaian data secara *broadcast*, dan juga karena *hub* hanya memiliki satu *domain collision* sehingga bila salah satu *port* sibuk maka *port-port* yang lain harus menunggu.

2.1.2.1 Komponen Dasar LAN

Beberapa komponen dasar yang biasanya membentuk suatu LAN adalah sebagai berikut:

1. *Workstation*

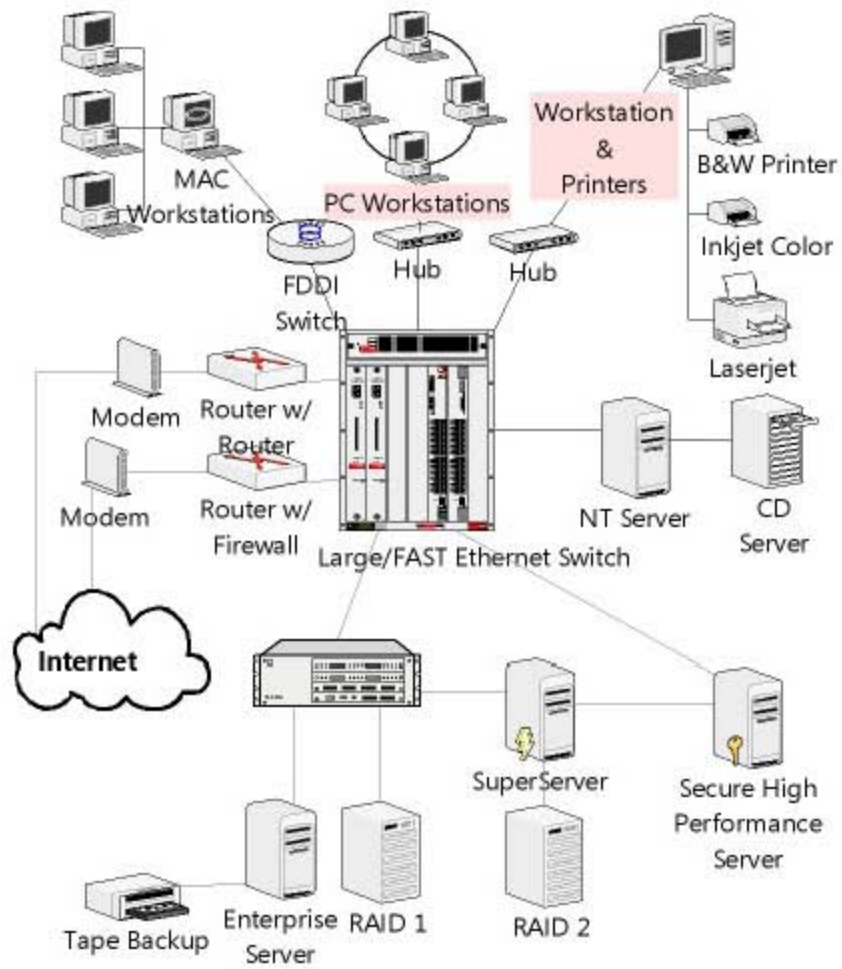
Workstation merupakan node atau host yang berupa suatu sistem komputer. Sistem komputer ini dapat berupa PC atau dapat pula berupa suatu komputer yang besar seperti sistem *minicomputer*, bahkan suatu *mainframe*. *Workstation* dapat bekerja sendiri (*stand-alone*) dapat pula menggunakan jaringan untuk bertukar data dengan *workstation* atau *user* yang lain.

2. *Server*

Perangkat keras (*hardware*) yang berfungsi untuk melayani jaringan dan *workstation* yang terhubung pada jaringan tersebut. Berdasarkan jenis pelayanannya dikenal *disk server*, *file server*, *print server*, dan suatu *server* juga dapat mempunyai beberapa fungsi pelayanan sekaligus.

3. *Link* (hubungan)

Workstation dan *server* tidak dapat berfungsi apabila peralatan tersebut secara fisik tidak terhubung. Hubungan tersebut dalam LAN dikenal sebagai media transmisi yang umumnya berupa kabel.



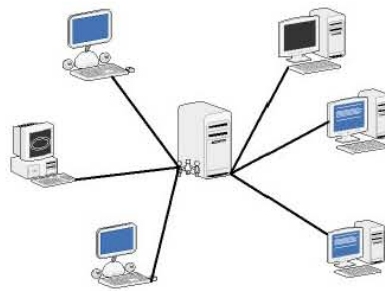
Gambar 2.3 Jaringan Client/Server yang Kompleks

2.1.2.2 Topologi LAN

Pengertian topologi Jaringan adalah susunan lintasan aliran data di dalam jaringan yang secara fisik menghubungkan simpul yang satu dengan simpul lainnya. Berikut ini adalah beberapa topologi jaringan yang ada dan dipakai hingga saat ini, yaitu :

1. Topologi Star

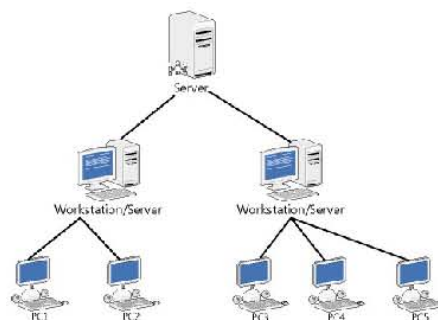
Beberapa simpul atau node dihubungkan dengan simpul pusat atau host, yang membentuk jaringan fisik seperti bintang.



Gambar 2.4 Topologi Star

2. Topologi Hierarkis

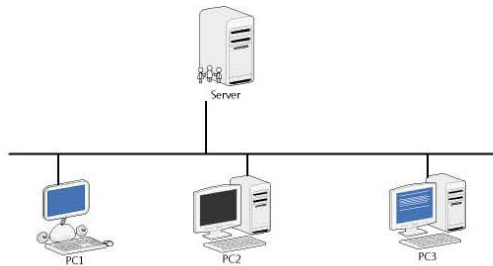
Berbentuk seperti pohon bercabang yang terdiri dari komputer induk(host) dihubungkan dengan simpul atau node lain secara berjenjang.



Gambar 2.5 Topologi Hirarkis

3. Topologi *Bus*

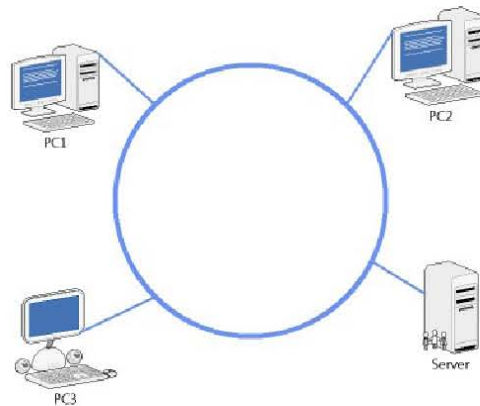
Beberapa simpul atau node dihubungkan dengan jalur data (bus). Masing-masing node dapat melakukan tugas-tugas dan operasi yang berbeda namun semua mempunyai hierarki yang sama.



Gambar 2.6 Topologi Bus

4. Topologi *Ring*

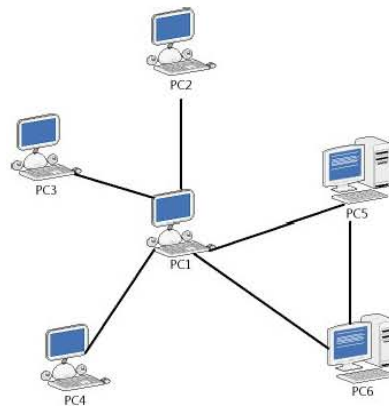
Bentuk ini merupakan gabungan bentuk topologi loop dan bus, jika salah satu simpul/node rusak, maka tidak akan mempengaruhi komunikasi node yang lain karena terpisah dari jalur data.



Gambar 2.7 Topologi Ring

5. Topologi *Loop*

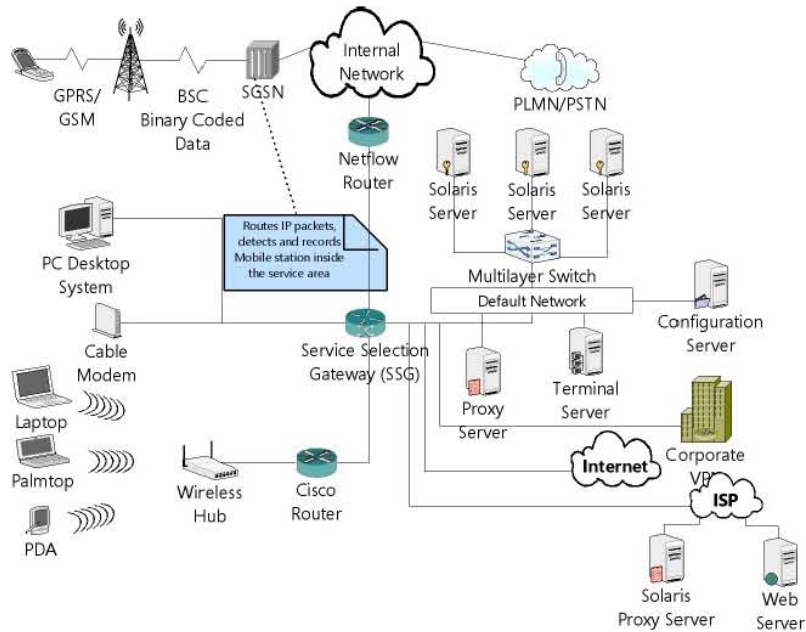
Merupakan hubungan antar simpul/*node* secara serial dalam bentuk suatu lingkaran tertutup. Dalam bentuk ini tak ada central *node/host*, semua mempunyai hierarki yang sama.



Gambar 2.8 Topologi Loop

2.1.3 *Metropolitan Area Network*

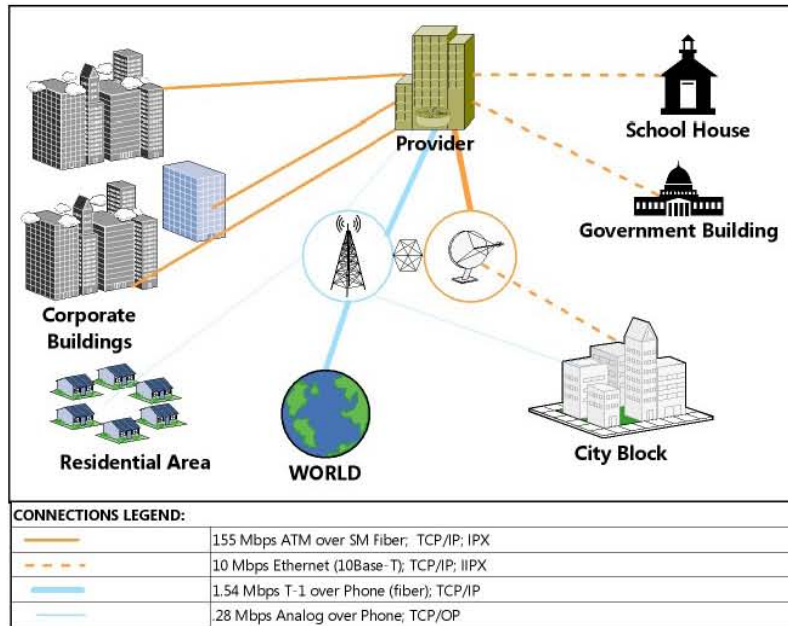
Metropolitan Area Network (MAN), pada dasarnya merupakan versi LAN yang berukuran lebih besar dan biasanya menggunakan teknologi yang sama dengan LAN. MAN dapat mencakup kantor-kantor perusahaan yang letaknya berdekatan atau juga sebuah kota dan dapat dimanfaatkan untuk keperluan pribadi (swasta) atau umum. MAN mampu menunjang data dan suara, bahkan dapat berhubungan dengan jaringan televisi kabel.



Gambar 2.9 Metropolitan Area Network

2.1.4 Wide Area Network

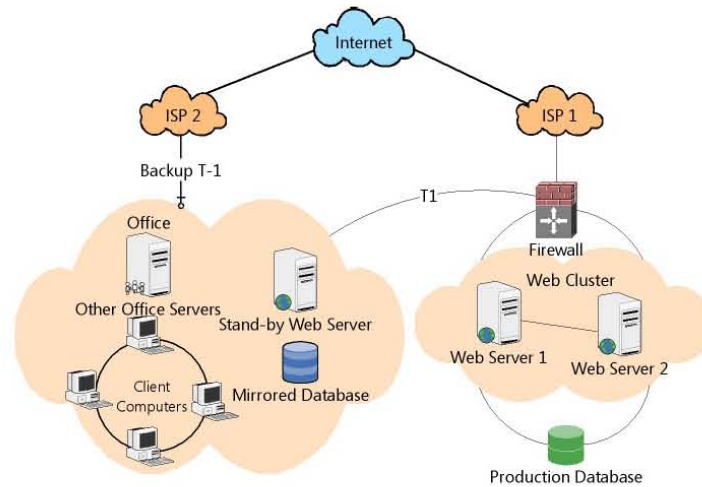
Wide Area Network (WAN), jangkauannya mencakup daerah geografis yang luas, seringkali mencakup sebuah negara bahkan benua. WAN terdiri dari kumpulan mesin-mesin yang bertujuan untuk menjalankan program-program (aplikasi) pemakai.



Gambar 2.10 Wide Area Network

2.1.5 Global Area Network

Sebenarnya terdapat banyak jaringan didunia ini, seringkali menggunakan perangkat keras dan perangkat lunak yang berbeda-beda . Orang yang terhubung ke jaringan sering berharap untuk bisa berkomunikasi dengan orang lain yang terhubung ke jaringan lainnya. Keinginan seperti ini memerlukan hubungan antar jaringan yang seringkali tidak kompatibel dan berbeda. Biasanya untuk melakukan hal ini diperlukan sebuah mesin yang disebut *gateway* guna melakukan hubungan dan melaksanakan terjemahan yang diperlukan, baik perangkat keras maupun perangkat lunaknya. Kumpulan jaringan yang ter-interkoneksi inilah yang disebut dengan *Global Area Network* atau lebih dikenal dengan *Internet*.



Gambar 2.11 Global Area Network

2.2 Transmission Control Protocol / Internet Protokol (TCP/IP)

Internet Protocol dikembangkan pertama kali oleh *Defense Advanced Research Projects Agency* (DARPA) pada tahun 1970 sebagai awal dari usaha untuk mengembangkan protokol yang dapat melakukan interkoneksi berbagai jaringan komputer yang terpisah, yang masing-masing jaringan tersebut menggunakan teknologi yang berbeda. Protokol utama yang dihasilkan proyek ini adalah *Internet Protocol* (IP). Riset yang sama dikembangkan pula yaitu beberapa protokol level tinggi yang didesain dapat bekerja dengan IP. Yang paling penting dari proyek tersebut adalah *Transmission Control Protocol* (TCP), dan semua grup protokol diganti dengan *TCP/IP suite*. Pertama kali TCP/IP diterapkan di ARPANET, dan mulai berkembang setelah Universitas California di Berkeley mulai menggunakan TCP/IP dengan sistem operasi UNIX. Selain *Defense Advanced Research Projects Agency* (DARPA) ini yang mengembangkan *Internet Protocol*, yang juga mengembangkan TCP/IP adalah *Department of defense* (DOD).

TCP/IP merupakan protokol jaringan komputer terbuka dan bisa terhubung dengan berbagai jenis perangkat keras dan lunak. TCP terdiri beberapa layer atau lapisan yang memiliki fungsi tertentu dalam komunikasi data. Setiap fungsi dari *layer* selain dapat

bekerjasama dengan *layer* pada tingkat lebih rendah atau lebih tinggi, juga bisa berkomunikasi dengan *layer* sejenis pada *remote host* (peering). IP adalah jantung TCP/IP memiliki peran sebagai pembawa data yang independen. Semua dokumen TCP/IP dalam bentuk *public document* IEN dan RFC. IP dibagi atas kelas network A, B, dan C. Sedangkan kelas D untuk keperluan *reverse* IP yang boleh diabaikan. IP ditulis dalam bilangan desimal dari 0 sampai 255. Data yang mengalir antar *layer* atau antar *host* dienkapsulasi dan diberi *header* agar tiap *layer* bisa memprosesnya. Sebuah *host* tidak tahu alamat IP *gateway* di *network* lain, tetapi data mengalir ke *host* tujuan di *network* lain melalui *gateway network* nya setelah diberi penentuan rutin alamat IP.

TCP/IP adalah salah satu perangkat lunak jaringan komputer (*networking software*) yang terdapat dalam sistem UNIX, dan dipergunakan dalam banyak komunikasi data UNIX dalam *local area network* (LAN) maupun *Internet*.

TCP singkatan dari *transfer control protocol* dan IP singkatan dari *Internet Protocol*. TCP/IP menjadi satu nama karena fungsinya selalu bergandengan satu sama lain dalam komunikasi data. TCP/IP saat ini dipergunakan dalam banyak jaringan komputer lokal (LAN) yang terhubung ke Internet, karena memiliki sifat:

1. Merupakan protokol standar yang terbuka, gratis dan dikembangkan terpisah dari perangkat keras komputer tertentu. Karena itu protokol ini banyak didukung oleh vendor perangkat keras, sehingga TCP/IP merupakan pemersatu perangkat keras komputer yang beragam merk begitu juga sebagai pemersatu berbagai perangkat lunak yang beragam merk.
2. Berdiri sendiri dari perangkat keras jaringan apapun.
Sifat ini memungkinkan TCP/IP bergabung dengan banyak jaringan komputer. TCP/IP bisa beroperasi melalui sebuah Ethernet, sebuah *token ring*, sebuah saluran *dial-up*, sebuah X-25 dan secara *virtual* melalui berbagai media fisik transmisi data.

-
3. Bisa dijadikan alamat umum sehingga tiap perangkat yang memakai TCP/IP akan memiliki sebuah alamat unik dalam sebuah jaringan komputer lokal, atau dalam jaringan komputer global seperti *Internet*.
 4. Protokol ini distandarisasi dengan skala tinggi secara konsisten, dan bisa memberikan servis kepada *user-user* di dunia.

2.2.1 Komunikasi TCP/IP

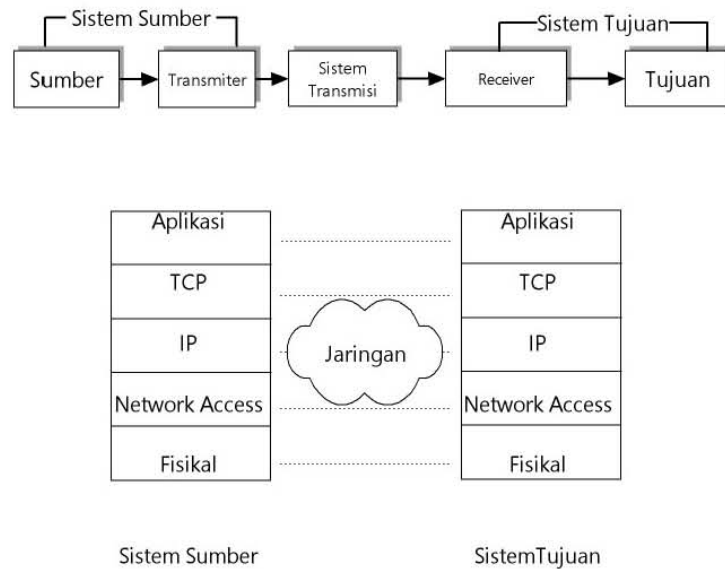
Dua arsitektur protokol telah disediakan sebagai dasar atau basis bagi pengembangan standar komunikasi yaitu :

1. TCP/IP *Protocol suite*
2. OSI reference model

TCP/IP adalah arsitektur standar yang banyak dipergunakan dan OSI telah menjadi model standar untuk mengklasifikasikan fungsi-fungsi komunikasi. TCP/IP merupakan hasil dari pengembangan dan riset protokol yang dilakukan atas jaringan paket-switched eksperimental (*experimental packet-switched network*), ARPANET, yang didanai oleh *Defense Advanced Research Project Agency* (DARPA), dan secara umum dikenal sebagai *TCP/IP protocol suite*. *Protocol suite* ini terdiri dari sekumpulan protokol dalam jumlah besar yang dijadikan sebagai standar Internet.

Tidak ada model protokol TCP/IP resmi sebagaimana yang ada dalam OSI. Bagaimana juga, bila didasarkan atas standar-standar protokol yang telah dikembangkan, kita dapat menyusun task-task komunikasi untuk TCP/IP menjadi lima lapisan independen secara relatif :

- ✦ Lapisan Aplikasi (*Application Layer*)
- ✦ Lapisan *Host-to-Host* ataupun *Transport*
- ✦ Lapisan Internet (*Internet Layer*)
- ✦ Lapisan Akses Jaringan (*Network Access Layer*)
- ✦ Lapisan Fisik (*Physical Layer*)



Gambar 2.12 Model Arsitektur Protokol

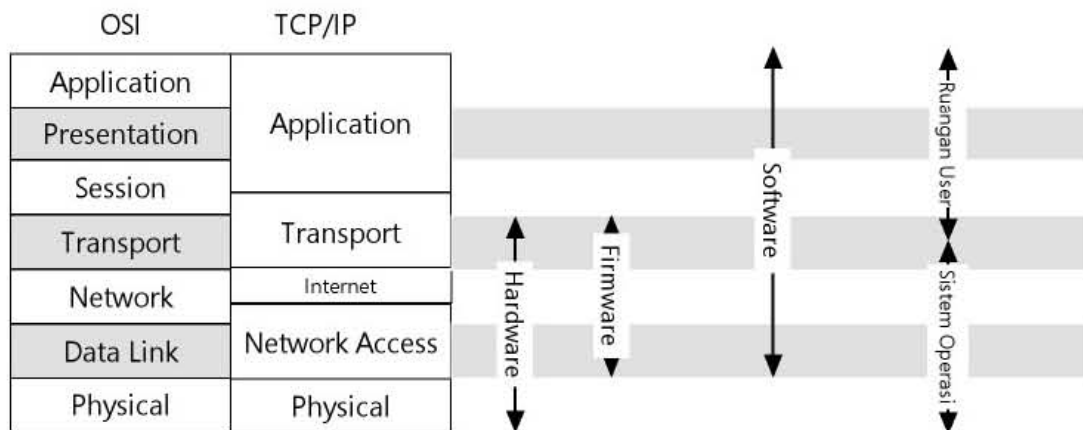
Physical layer meliputi *interface* fisik antara suatu perangkat transmisi data (misalnya *workstation*, komputer) dengan sebuah media transmisi atau jaringan. Lapisan ini berkaitan dengan karakteristik-karakteristik khusus dari media transmisi, sifat sinyal, *rate* data dan lain-lain yang berkaitan dengan hal itu.

Network Access Layer berkaitan dengan pertukaran data antara sebuah ujung sistem dengan jaringan mana dihubungkan. Software khusus yang dipergunakan pada lapisan ini tergantung pada tipe jaringan yang digunakan, standar-standar berbeda telah dikembangkan untuk *circuit switching*, paket *switching* (misalnya X.25), LAN (misalnya Ethernet), dan lain-lain.

Lapisan *Network Access* berkaitan dengan pengaksesan ke mana serta pengiriman data melalui sebuah jaringan untuk dua ujung sistem yang dihubungkan ke jaringan yang sama. Dalam kasus tertentu di mana dua perangkat dihubungkan ke jaringan yang berbeda, diperlukan prosedur-prosedur tertentu agar data dapat melintasi jaringan yang bermacam-macam. Ini merupakan fungsi dari *internet layer*. IP (*Internet Protocol*) digunakan pada

lapisan ini untuk menyediakan fungsi *routing* untuk melintasi jaringan yang bermacam-macam. Protokol ini diterapkan tidak hanya pada ujung sistem namun juga pada jalur-jalurnya. *Router* adalah suatu *prosesor* yang menghubungkan dua jaringan dan fungsi utamanya adalah untuk *me-relay* data dari satu jaringan ke jaringan yang lain pada jalurnya mulai dari sumber ke ujung sistem tujuan.

Perancang OSI mengasumsikan bahwa model dan protokol-protokol yang dikembangkan di dalam model ini akan mendominasi komunikasi komputer, akhirnya menggantikan *proprietary protocol* dan saingan *multivendor* seperti TCP/IP. Hal ini tidak terjadi. Meskipun, banyak protokol yang sangat bermanfaat telah dikembangkan sesuai konteks OSI, seluruh tujuh model lapisan tidak dapat berkembang dengan baik, arsitektur TCP/IP yang mendominasi.



Gambar 2.13 Perbandingan antara Protokol OSI dan TCP/IP

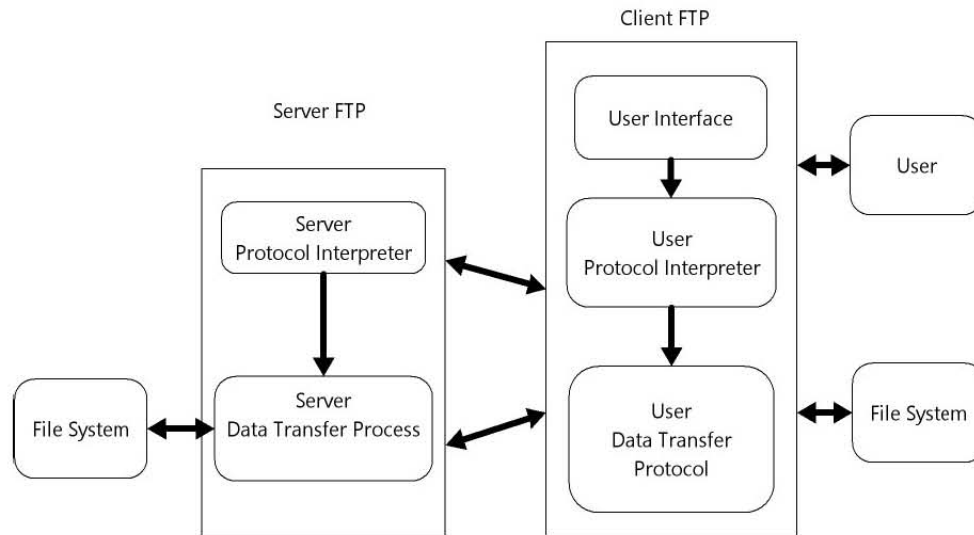
2.2.2 Layanan TCP/IP

Salah satu lapisan dari TCP/IP adalah lapisan *transport*. Pada lapisan ini tersedia layanan pengiriman pesan, pengiriman pesan ini bertugas untuk mendeteksi kesalahan dalam pengiriman data tersebut. Ada dua kategori umum deteksi kesalahan dapat dilakukan oleh lapisan transport :

-
- a. *Reliable delivery*, berarti kesalahan tidak dapat terjadi, tetapi kesalahan akan dideteksi jika terjadi.
 - b. *Unreliable delivery*, bukan berarti kesalahan mungkin terjadi, tetapi menunjukkan bahwa lapisan transport tidak memeriksa kesalahan tersebut.

2.2.3 File Transfer Protocol (FTP)

Layanan lapisan aplikasi salah satunya adalah *File Transfer Protocol* (FTP). *File Transfer Protokol* (FTP) memungkinkan pengguna komputer yang satu untuk dapat mengirim ataupun menerima file ke komputer jaringan. Layanan ini memungkinkan para pengguna internet untuk melakukan pengiriman (*upload*) atau menyalin (*download*) sebuah file antara komputer lokal dengan komputer lain yang terhubung dalam jaringan internet. FTP umumnya dimanfaatkan sebagai sarana pendukung untuk kepentingan pertukaran maupun penyebarluasan sebuah file melalui jaringan internet. FTP juga dimanfaatkan untuk melakukan proses *upload* suatu halaman *web* ke *webserver* agar dapat diakses oleh pengguna *internet* lainnya. Secara teknis, aplikasi FTP disebut sebagai *FTP client*, aplikasi ini umumnya dimanfaatkan untuk transaksi FTP yang bersifat dua arah (*active FTP*). Modus ini memungkinkan pengguna untuk melakukan baik proses *upload* maupun proses *download*. Tidak semua *server* FTP dapat diakses dalam modus *active*. Untuk mencegah penyalahgunaan yang dapat berakibat fatal bagi sebuah *server* FTP maka pengguna FTP untuk modus *active* harus memiliki hak akses untuk mengirimkan file ke sebuah server FTP. Hak akses tersebut berupa sebuah *login name* dan *password* sebagai kunci untuk memasuki sebuah sistem FTP *server*. Untuk modus *passive*, selama memang tidak ada restriksi dari pengelola server, umumnya dapat dilakukan oleh semua pengguna dengan modus *anonymous* login (*login* secara anonim). Kegiatan melakukan *download* software dari *internet* misalnya, juga dapat digolongkan sebagai *passive* FTP.



Gambar 2.14 Diagram File Transfer Protocol

Kedua komputer (host) harus bekerja pada jaringan TCP/IP. *User* melakukan *request* melalui sebuah aplikasi FTP. Aplikasi menyampaikan pesan kepada *User Protocol Interpreter* (UPI) dan meneruskan kepada *Server Protocol Interpreter* (SPI) melalui port 21. *Protocol Interpreter* mengontrol dan melewatkan perintah FTP melalui hubungan. Seperti yang terlihat pada diagram, Protocol Interface tidak hanya mengirimkan pesan pada semua tetapi juga dengan *Data Transfer Protocol* (DTP) pada setiap sisi masing-masing. Pada *File Transfer Protocol* terdapat perintah-perintah dalam melakukan operasi. Perintah dasar FTP adalah sebagai berikut :

Tabel 2.1 Perintah dasar FTP

? atau help	Menampilkan daftar perintah
cd	Penggantian direktori
dir atau ls	Isi dari direktori
user	Identifikasi user
get	Download File
pwd	Melihat direktori yang aktif
quit	Keluar
send	Mengirimkan data

2.3 IP Address

Jaringan Internet merupakan integrasi dari puluhan juta komputer (*host*) yang tersambung melalui ratusan ribu jaringan di seluruh dunia. Komputer di rumah atau di kantor anda dapat berhubungan dengan komputer manapun di Internet karena semuanya memiliki *IP Address* sebagai alat pengenalan. Alokasi *IP Address* untuk setiap komputer yang tersambung pada sebuah jaringan harus dilakukan dengan benar agar *routing* dapat berjalan dengan baik.

Walaupun bagi para pengguna Internet umumnya kita hanya perlu mengenal *hostname* dari mesin yang dituju, seperti: *widyatama.ac.id*, *server.indo.net.id*, *rad.net.id*, *ui.ac.id*, *itb.ac.id*. Bagi komputer untuk bekerja langsung menggunakan informasi tersebut akan relatif lebih sulit karena tidak ada keteraturan yang dapat di programkan dengan mudah. Untuk mengatasi hal tersebut, komputer mengidentifikasi alamat setiap komputer menggunakan sekumpulan angka sebanyak 32 bit yang dikenal sebagai *IP address*.

Adanya *IP Address* merupakan konsekuensi dari penerapan *Internet Protocol* untuk mengintegrasikan jaringan komputer Internet di dunia. Seluruh *host* (komputer) yang terhubung ke Internet dan ingin berkomunikasi memakai TCP/IP harus memiliki *IP Address* sebagai alat pengenalan *host* pada *network*. Secara logika, *Internet* merupakan suatu *network* besar yang terdiri dari berbagai *sub network* yang terintegrasi. Oleh karena itu, suatu *IP Address* harus bersifat unik untuk seluruh dunia. Tidak boleh ada satu *IP Address* yang sama dipakai oleh dua *host* yang berbeda.

2.3.1 Struktur IP Address

IP Address terdiri dari bilangan biner sepanjang 32 bit yang dibagi atas 4 segmen. Tiap segmen terdiri atas 8 bit yang berarti memiliki nilai desimal dari 0 - 255. *Range address* yang bisa digunakan adalah dari 00000000.00000000.00000000.00000000 sampai dengan 11111111.11111111.11111111.11111111. Jadi, ada sebanyak 232 kombinasi *address* yang bisa dipakai diseluruh dunia (walaupun pada kenyataannya ada sejumlah IP

Address yang digunakan untuk keperluan khusus). Jadi, jaringan TCP/IP dengan 32 bit *address* ini mampu menampung sebanyak 232 atau lebih dari 4 milyar *host*.

Untuk memudahkan pembacaan dan penulisan, IP *Address* biasanya direpresentasikan dalam bilangan desimal. Jadi, range address di atas dapat diubah menjadi *address* 0.0.0.0 sampai *address* 255.255.255.255. Nilai desimal dari IP *Address* inilah yang dikenal dalam pemakaian sehari-hari. Beberapa contoh IP *Address* adalah :

- ✦ 45.132.1.20
- ✦ 167.205.9.36
- ✦ 202.152.1.251

Ilustrasi IP *Address* dalam desimal dan biner dapat dilihat pada gambar 2.14 berikut ini :

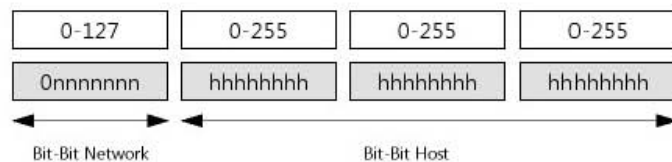
Desimal	167	205	9	36
Biner	10100111	11001101	00001001	00100100

Gambar 2.15 IP Address dalam Bilangan Desimal dan Biner

IP *Address* dapat dipisahkan menjadi 2 bagian, yakni bagian *network* (bit-bit *network/network bit*) dan bagian *host* (bit-bit *host/host bit*). Bit *network* berperan dalam identifikasi suatu *network* dari *network* yang lain, sedangkan bit *host* berperan dalam identifikasi *host* dalam suatu *network*. Jadi, seluruh *host* yang tersambung dalam jaringan yang sama memiliki bit *network* yang sama. Sebagian dari bit-bit bagian awal dari IP *Address* merupakan *network bit/network number*, sedangkan sisanya untuk *host*. Garis pemisah antara bagian *network* dan *host* tidak tetap, bergantung kepada kelas *network*. Ada 3 kelas *address* yang utama dalam TCP/IP, yakni kelas A, kelas B dan kelas C. Perangkat lunak *Internet Protocol* menentukan pembagian jenis kelas ini dengan menguji beberapa bit pertama dari IP *Address*. Penentuan kelas ini dilakukan dengan cara berikut :

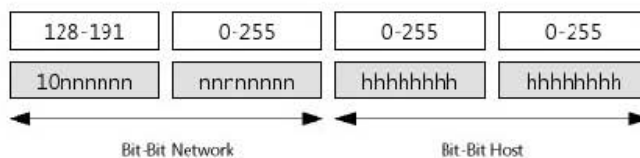
- ✦ Jika bit pertama dari IP *Address* adalah 0, *address* merupakan *network* kelas A. Bit ini dan 7 bit berikutnya (8 bit pertama) merupakan bit *network* sedangkan

24 bit terakhir merupakan bit *host*. Dengan demikian hanya ada 128 *network* kelas A, yakni dari nomor 0.xxx.xxx.xxx sampai 127.xxx.xxx.xxx, tetapi setiap *network* dapat menampung lebih dari 16 juta (256^3) host (xxx adalah variabel, nilainya dari 0 s/d 255). Ilustrasinya dapat dilihat pada gambar 2.15 berikut.



Gambar 2.16 Struktur IP Address Kelas A

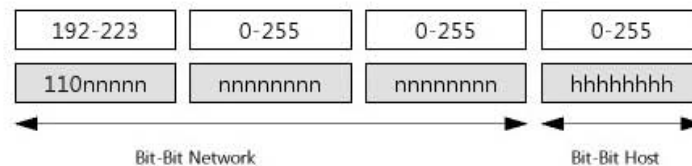
- ✦ Jika 2 bit pertama dari IP Address adalah 10, *address* merupakan *network* kelas B. Dua bit ini dan 14 bit berikutnya (16 bit pertama) merupakan bit *network* sedangkan 16 bit terakhir merupakan bit *host*. Dengan demikian terdapat lebih dari 16 ribu *network* kelas B (64×256), yakni dari *network* 128.0.xxx.xxx - 191.255.xxx.xxx. Setiap *network* kelas B mampu menampung lebih dari 65 ribu *host* (256^2). Ilustrasinya dapat dilihat pada gambar 2.16 berikut.



Gambar 2.17 Struktur IP Address Kelas B

- ✦ Jika 3 bit pertama dari IP Address adalah 110, *address* merupakan *network* kelas C. Tiga bit ini dan 21 bit berikutnya (24 bit pertama) merupakan bit *network* sedangkan 8 bit terakhir merupakan bit *host*. Dengan demikian terdapat lebih dari 2 juta *network* kelas C ($32 \times 256 \times 256$), yakni dari nomor

192.0.0.xxx sampai 223.255.255.xxx. Setiap *network* kelas C hanya mampu menampung sekitar 256 *host*. Ilustrasinya dapat dilihat pada gambar 2.17.



Gambar 2.18 Struktur IP Address Kelas C

Selain ke tiga kelas di atas, ada 2 kelas lagi yang ditujukan untuk pemakaian khusus, yakni kelas D dan kelas E. Jika 4 bit pertama adalah 1110, IP Address merupakan kelas D yang digunakan untuk *multicast address*, yakni sejumlah komputer yang memakai bersama suatu aplikasi (bedakan dengan pengertian *network address* yang mengacu kepada sejumlah komputer yang memakai bersama suatu *network*). Salah satu penggunaan *multicast address* yang sedang berkembang saat ini di *Internet* adalah untuk aplikasi *real-time video conference* yang melibatkan lebih dari dua *host* (*multipoint*), menggunakan *Multicast Backbone (MBone)*. Kelas terakhir adalah kelas E (4 bit pertama adalah 1111 atau sisa dari seluruh kelas). Pemakaiannya dicadangkan untuk kegiatan eksperimental.

Jenis kelas *address* yang diberikan oleh koordinator IP Address bergantung kepada kebutuhan instansi yang meminta, yakni jumlah *host* yang akan diintegrasikan dalam *network* dan rencana pengembangan untuk beberapa tahun mendatang. Untuk perusahaan, kantor pemerintah atau universitas besar yang memiliki puluhan ribu komputer dan sangat berpotensi untuk tumbuh menjadi jutaan komputer, koordinator IP Address akan mempertimbangkan untuk memberikan kelas A. Contoh IP Address kelas A yang dipakai di *Internet* adalah untuk amatir paket radio seluruh dunia, mendapat IP nomor 44.xxx.xxx.xxx.

2.3.2 *Address Khusus*

Selain *address* yang dipergunakan untuk mengenal *host*, ada beberapa jenis *address* yang digunakan untuk keperluan khusus dan tidak boleh digunakan untuk mengenal *host*. *Address* tersebut adalah :

- ✦ *Network Address*. *Address* ini digunakan untuk mengenali suatu *network* pada jaringan *Internet*.
- ✦ *Broadcast Address*. *Address* ini digunakan untuk mengirim/menerima informasi yang harus diketahui oleh seluruh *host* yang ada pada suatu *network*.
- ✦ *Netmask*. adalah *address* yang digunakan untuk melakukan *masking* / *filter* pada proses pembentukan *routing* supaya kita cukup memperhatikan beberapa bit saja dari total 32 bit *IP Address*.

2.4 *Transmission Control Protocol*

TCP merupakan protokol *layer transport* yang dapat menyediakan layanan *connection oriented*, *reliable* dan *byte stream orientation*.

Protokol TCP menyediakan :

- ✦ Spesifikasi format paket data dan ack yang dipertukarkan antara 2 *host* untuk penyediaan layanan yang dapat diandalkan.
- ✦ Spesifikasi metode yang digunakan oleh *software* TCP untuk pengenalan aplikasi tujuan pada suatu mesin.
- ✦ Metode komunikasi antar *host* jika terjadi duplikasi atau *lost packet*.
- ✦ Metode komunikasi antar *host* untuk proses transfer data.

Pada dokumentasi protocol yang ada, hanya mendiskusikan mengenai operasi yang dapat disediakan oleh TCP, tetapi tidak menspesifikasikan prosedur program aplikasi yang terlibat dalam suatu proses operasi. Alasan dari tidak ditentukannya *interface* program aplikasi adalah untuk mendukung fleksibilitas.

2.5 User Datagram Protocol (UDP)

UDP menyediakan mekanisme dasar yang digunakan oleh program aplikasi untuk mengirim *datagram* ke program aplikasi lain. UDP menyediakan *port* protokol yang digunakan untuk membedakan satu program yang sedang dieksekusi dengan yang lain dalam satu mesin. UDP menggunakan protokol dibawahnya (IP) untuk menyampaikan pesan dari satu mesin ke mesin lain dan menyediakan semantik pengiriman datagram yang tidak *reliable*, serta *connectionless* seperti IP. Disini tidak terdapat ACK untuk memastikan sampainya pesan di tujuan, data yang sampai tidak perlu terurut, dan tidak membutuhkan kontrol balik terhadap rate aliran informasi antar mesin. Suatu program aplikasi yang menggunakan UDP menerima tanggung jawab penuh untuk menangani masalah reliabilitas termasuk kehilangan pesan, duplikasi, *delay*, pengiriman yang tidak terurut dan putus koneksi. Karena sifatnya yang *connectionless* dan *unreliable*, UDP digunakan oleh aplikasi-aplikasi yang secara periodic melakukan aktivitas tertentu (misalnya *query routing* tabel pada jaringan lokal), serta hilangnya satu data akan dapat di atasi pada *query* periode berikutnya dan melakukan pengiriman data ke jaringan lokal. Pendeknya jarak tempuh datagram akan mengurangi resiko kerusakan data. Bersifat *broadcasting* atau *multicasting*. Pengiriman *datagram* ke banyak *client* sekaligus akan efisien jika prosesnya menggunakan metode *connectionless*.

2.6 Layanan *Reliable Stream Transport*

2.6.1 Kebutuhan Layanan Pengiriman Stream

Layanan pengiriman data pada level dibawah level *transport* seringkali mengabaikan kehandalan suatu layanan. Kehilangan paket, kegagalan perangkat jaringan, ketidakterurutan data, duplikasi, *delay* yang besar merupakan sebab dibutuhkannya layanan transmisi data yang lebih *reliable*. Karakteristik suatu layanan data dikatakan *reliable* adalah :

- ✦ *Stream Orientation*

Untuk layanan data dalam kapasitas besar, data dianggap sebagai aliran(*stream*) bit yang dibagi-bagi menjadi oktet atau *byte*, aliran *byte* yang masuk ke mesin tujuan memiliki urutan yang sama pada waktu pengiriman.

- ✦ *Virtual Circuit Connection*

Sebelum pengiriman data dilakukan diawali dengan komunikasi antara pengirim dan tujuan mengenai pembentukan sebuah koneksi.

- ✦ *Buffered transfer*

Program aplikasi mengirimkan aliran data ke *virtual circuit* yang terbentuk dengan cara mengirim oktet-oktet data ke *protocol software* secara berulang. Ketika mentransfer data setiap aplikasi dapat menggunakan beragam ukuran oktet. *Protocol software* akan mengirimkan data tersebut sesuai urutan aslinya. *Protocol* ini dapat dengan bebas membagi paket menjadi bagian-bagian independen untuk ditransferkan.

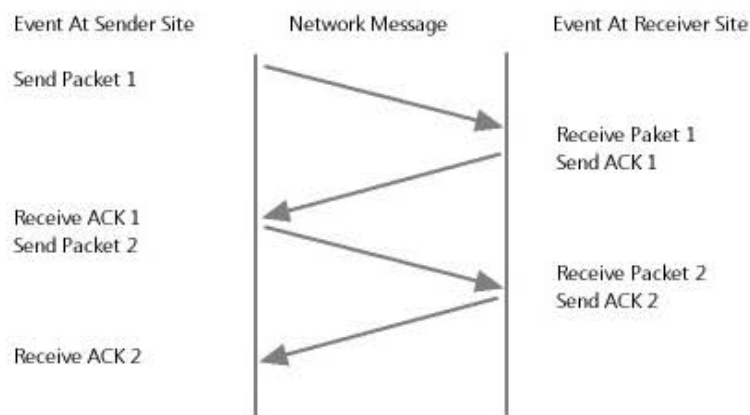
- ✦ *Unstructured Stream*

- ✦ *Full Duplex Connection*

Koneksi yang disediakan oleh layanan *stream* TCP/IP memungkinkan transfer secara bersamaan di dua arah (*full duplex*)

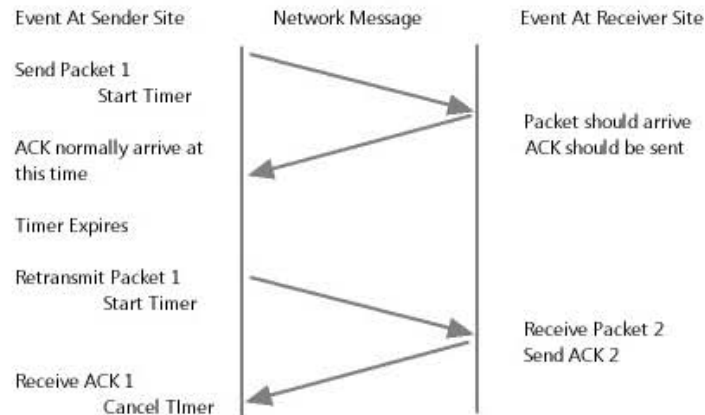
2.6.2 Reliabilitas

Salah satu metode untuk menerapkan layanan *transfer reliable* di atas layanan tidak reliable adalah dengan teknik *Positive Acknowledgement with Retransmission*. Metode yang dijalankan adalah metode pengiriman jawaban untuk pemberitahuan bahwa data yang dikirimkan telah diterima. Pengirim akan menyimpan 1 *record* dari setiap paket yang dikirimkannya dan menunggu datangnya ack sebelum mengirim paket yang berikutnya. Ketika pengiriman dilakukan pengirim juga mengaktifkan suatu timer untuk menandai masa aktif suatu paket. Timer ini juga digunakan untuk batasan bilamana perlu dilakukan retransmisi.



Gambar 2.19 *Positive ACK with retransmission*

Sisi kanan dan kiri menunjukkan jenis-jenis *event* yang terjadi baik di sisi pengirim (*sender*) maupun disisi penerima (*receiver*). Sedangkan garis diagonal menunjukkan *transfer* pesan melalui jaringan. Dalam suatu kasus dimana suatu paket *lost*, maka dilakukan retransmisi pada saat timer *expires* (masih aktif paket habis). Mekanismenya dapat dilihat pada gambar berikut ini :

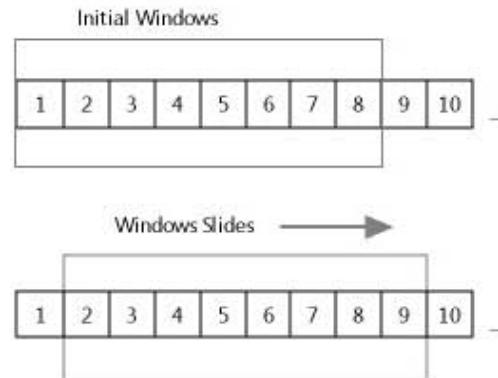


Gambar 2.20 Mekanisme Retransmisi

Masalah yang mungkin muncul akibat dari paket yang terlambat adalah adanya duplikasi data. Untuk menganganinya diperlukan perhatian pada aspek paket dan ack-nya. Duplikasi dapat dideteksi oleh protocol dengan memeriksa *sequence number* yang terdapat pada tiap paket dan meminta penerima untuk mengingat *sequence number* yang telah diterima. Untuk menghindari kesalahan interpretasi antara paket tertunda atau ack yang terduplikasi, *protocol* ack positif akan mengirim kembali *sequence number* dalam paket ack.

2.6.3 *Sliding Windows*

Untuk meningkatkan efisiensi pengiriman sehingga jaringan tidak sering *idle* digunakan *sliding windows*. *Sliding windows* adalah metode pengiriman banyak paket dalam satu ukuran *window* secara kontinyu tanpa harus menunggu kehadiran ack.



Gambar 2.21 Proses *Sliding Window* dengan Delapan Paket

Pada gambar di atas *window* yang dipakai berukuran delapan paket. *Window* akan bergeser sebanyak jumlah ack yang diterimanya yang menandakan boleh dikirimnya paket yang menjadi anggota dalam *window* tersebut. Retransmisi juga akan dilakukan untuk paket-paket yang tidak dijawab dengan ack. Performansi protocol *sliding window* bergantung pada ukuran *window* yang digunakan dan kecepatan jaringan dalam menerima paket.

BAB III

ANALISIS SISTEM

3.1 Deskripsi Sistem

Sistem yang dibangun adalah Aplikasi Transfer File berbasis TCP/IP yang merupakan aplikasi client, dimana sistem hanya melayani permintaan pengguna terhadap server *File Transfer Protocol* (FTP). Berdasarkan fungsi sistem, server *File Transfer Protocol* (FTP) diasumsikan sudah terpasang aplikasi *File Transfer Protocol* (FTP) server yang akan melayani permintaan pengguna, dimana sistem *File Transfer Protocol* (FTP) yang berbasis *client* ini memiliki fasilitas antara lain:

a. Fasilitas *login*

Fasilitas ini untuk *user* yang telah mendaftarkan diri sebagai pengguna sistem Aplikasi Transfer File berbasis TCP/IP. Dengan adanya input *user id* dan *password* dari *user* maka sistem akan mencocokkan dalam *database user* yang terdapat pada server, apakah *user name* dan *password* yang dimasukkan sudah sesuai. Jika sesuai maka sistem akan dapat diakses oleh *user* sesuai dengan hak aksesnya. Tapi jika tidak ada kesesuaian antara *user name* dan *password* maka sistem akan meminta *login* kembali.

b. Identifikasi *user*

Berdasarkan *user name* maka sistem akan mampu untuk mengidentifikasi *user* dan memberikan hak akses yang sesuai.

3.2 Deskripsi Pengguna

3.2.1 Hirarki Pengguna

Pengguna sistem Aplikasi Transfer File berbasis TCP/IP merupakan pengguna yang diberikan masing-masing hak akses untuk keperluan pengaksesan sistem.

Pengguna dari sistem terbagi atas dua pengguna, yaitu :

1. *User*, yang memiliki hak akses untuk melakukan *read* dan *write*.
2. *Anonymous user*, yang memiliki hak akses untuk melakukan proses *read* pada direktori *public*.

3.2.2 Karakteristik Pengguna

Pengguna sistem ini harus seorang yang:

- a. Mampu mengoperasikan komputer dengan baik, dalam artian memiliki pengetahuan yang cukup mengenai komputer.
- b. Memiliki pengetahuan tentang proses-proses yang akan dilakukan pada saat terkoneksi dengan server FTP.

3.3 Analisis Sistem

File Transfer Protocol (FTP) melakukan koneksi antar dua host atau komputer. Dengan hubungan yang terkoneksi maka *File Transfer Protocol* (FTP) dapat melakukan pengiriman (*upload*) atau menyalin (*download*) sebuah atau lebih file antar komputer yang terhubung lokal maupun yang terhubung melalui Internet. *File Transfer Protocol* (FTP) digunakan sebagai suatu sarana pendukung untuk pertukaran dan penyebarluasan sebuah atau lebih file dalam suatu jaringan *internet*. Secara teknis, aplikasi *File Transfer Protocol* (FTP) dibagi dua bagian yaitu :

1. *FTP Client* : aplikasi ini dimanfaatkan untuk transaksi FTP yang bersifat dua arah (*active FTP*).
2. *FTP Server* : aplikasi ini dimanfaatkan untuk proses *login* ke *server* FTP dan melayani semua permintaan *FTP Client*.

3.3.1 Request FTP

Sama halnya dengan protokol HTTP, *FTP client* atau *server* didasarkan atas pola yang biasa. *Client* melakukan koneksi ke *server* (*remote machine*), dimana untuk melakukan koneksi dibutuhkan aplikasi khusus (*FTP Server*). Pada waktu koneksi sudah terbentuk, transaksi antar dua komputer dimulai, *client* melakukan permintaan (*request*) dan komputer *server* melakukan *reply* dengan jawaban atas permintaan *client*.

3.3.1.1 Perintah FTP

Perintah *File Transfer Protocol* (FTP) terdiri dari tiga atau empat karakter dalam huruf besar, dihubungkan dengan spasi dan kemudian parameter yang ditentukan. Perintah *File Transfer Protocol* (FTP) dapat dilihat dalam tabel berikut :

Tabel 3.1 Perintah FTP

Perintah [parameter]	Keterangan
USER [nama user]	Identifikasi user pada Server FTP
PASS [password]	<i>Password</i> masukan dari user (terlebih dahulu memasukkan perintah USER)
ABOR	Membatalkan operasi yang sedang diproses
QUIT	Memutus koneksi dengan <i>server</i>
PORT	Menentukan <i>port</i> yang spesifik dimana koneksi akan dibuat
CWD [direktori baru]	Mengubah direktori
CDUP	Mengubah direktori ke direktori <i>parent</i> (sama seperti "CD" pada MSDOS dan UNIX)
MKD [nama direktori]	Membuat direktori baru
RMD [nama direktori]	Menghapus direktori
DELETION MARK [nama file]	Mengumpulkan file dalam struktur direktori pada <i>server</i>
RNFR [nama file]	Mengubah nama file. Perintah ini menyatakan nama file yang akan diubah dan diikuti dengan perintah RNTO
RNTO [nama file yang baru]	Mengubah nama file yang sudah dinyatakan oleh perintah RNFR dan membuat nama baru yang ada pada parameter
SIZE [nama file]	Melihat ukuran file (dalam <i>bytes</i>) pada file <i>remote</i>
LIST [file atau direktori]	Menampilkan daftar file dalam direktori
NOOP	Perintah ini digunakan untuk melakukan pengecekan apakah koneksi masih aktif atau tidak
RETR [nama file]	Perintah ini untuk mengirimkan sebuah file ke <i>client</i> atau <i>server</i> pada akhir koneksi. Status dan isi dari file pada server tidak akan berubah.
STOR [nama file]	Perintah ini untuk mengetahui apakah file yang dikirim sudah diterima oleh server. Jika file ternyata sudah ada pada server, maka file akan digantikan oleh file yang baru saja dikirimkan
HELP	Perintah ini digunakan untuk menampilkan bantuan

File Transfer Protocol (FTP) mengikuti spesifikasi dari protokol Telnet untuk semua komunikasi yang terjadi selama koneksi berlangsung. Pada saat perintah dieksekusi,

jika terdapat kesalahan, sebuah kode kesalahan akan ditampilkan, diikuti dengan pesan kesalahan.

3.3.2 *Codes Of Return*

Hasil dari operasi pada saat perintah dijalankan terdiri dari tiga angka, setiap angka mempunyai arti, arti dari angka tersebut berikut penjelasannya :

1. Angka pertama : memberikan keterangan pada operasi yang dilakukan. Angka ini menunjukkan sukses atau tidaknya suatu operasi dan menunjukkan siapa yang melakukan kesalahan. Berikut adalah kode angka :
 - ✦ 1xx : *server* telah mengenali perintah yang dimasukkan
 - ✦ 2xx : *server* telah selesai melakukan perintah
 - ✦ 3xx : *server* telah menerima perintah dengan benar
 - ✦ 4xx : *server* telah menerima perintah, tetapi perintah yang dimasukkan salah, melanjutkan ke eksekusi selanjutnya
 - ✦ 5xx : *server* telah menerima perintah, tetapi penuh dengan kesalahan. Tidak memungkinkan untuk melanjutkan eksekusi
2. Angka kedua : Merupakan suatu petunjuk yang menyebabkan suatu kegagalan atau keberhasilan operasi. Kode yang digunakan adalah sebagai berikut :
 - ✦ x0x : sintak
 - ✦ x1x : permintaan informasi, pernyataan dan bantuan
 - ✦ x2x : koneksi kontrol file
 - ✦ x3x : awal dari sesi proses
 - ✦ x4x : *reserved*
 - ✦ x5x : file sistem

Pada tabel berikut dijelaskan kombinasi pesan kesalahan yang dihasilkan pada saat operasi dijalankan.

Tabel 3.2 Pesan Kesalahan

Kode	Keterangan
101	Inisialisasi untuk jawaban perintah
120	Layanan akan segera dimulai dalam beberapa saat
125	Pembukaan koneksi data
200	Perintah Tepat
202	Perintah tidak mendukung
211	Informasi tentang keadaan sistem
212	Informasi pada sistem file
213	Informasi dari file
214	Pesan atau informasi dan bantuan
221	Melakukan pemutusan koneksi
225	Koneksi data terbuka, tapi tidak ada proses pengiriman
226	Memutuskan koneksi data. Permintaan akan file telah selesai dilakukan
227	Masuk ke dalam mode pasif
230	User telah berhasil masuk dan melakukan proses
250	Permintaan akan file telah selesai
257	File atau direktori telah selesai dilakukan
331	Nama user benar, password dibutuhkan
425	Tidak dapat membuka koneksi data
426	Pembatalan pengiriman
450	Membatalkan operasi yang dilakukan pada file
452	Membatalkan operasi karena kapasitas ruang penyimpan tidak mencukupi
500	Kesalahan sintak, perintah tidak dikenali
501	Kesalahan sintak yang terdapat pada parameter
502	Perintah tidak dilaksanakan
503	Kesalahan urutan sintak
504	Perintah tidak dilaksanakan pada parameter yang paling akhir
530	Tidak terhubung ke <i>server</i>
532	Membutuhkan sebuah <i>account</i> untuk menyimpan file
552	Direktori telah melampaui kapasitas
553	Kesalahan penamaan file

3.4 Sistem File Transfer Protocol (FTP)

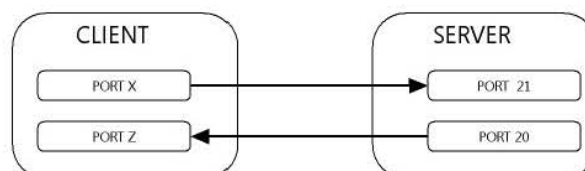
Pada mode koneksi aktif pola kerja sistem *File Transfer Protocol* (FTP) bekerja pada dua cara, yaitu :

1. *Assets* (Cara koneksi)
2. *Liabilities* (lebih dikenal dengan *Passive mode*)

Namun hal yang paling penting yang menjadi dasar pola kerja FTP adalah konsep dari protokol TCP/IP dan konsep dari port.

Port adalah pengenal dari koneksi *remote*. Untuk melakukan koneksi pada *remote* diharuskan menggunakan sebuah *port* yang bebas. Biasanya jumlah *port* berjumlah 1 sampai 65535, dan *port* pertama telah dipakai oleh sistem (tergantung pada sistem operasi). Ketika terhubung dengan sebuah *remote machine*, nama mesin tujuan harus sudah dinyatakan (*IP address* atau nama mesin). Ketika data sampai pada *remote machine*, *remote machine* segera mengetahui koneksi apakah yang ada pada data. Jika *port* tidak ditentukan, maka *remote machine* tidak dapat membedakan koneksi yang akan dibuat. Banyaknya *client* dan *server* harus mempunyai sebuah atau lebih *port* yang aktif ketika melakukan koneksi. *Client* yang melakukan koneksi memilih *port* pada *server*, dan *server* yang menentukan *port* mana yang akan diberikan pada *client*. Berikut adalah urutan cara kerja *File Transfer Protocol* (FTP) yang aktif :

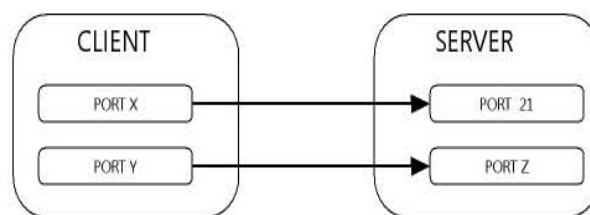
1. *Client* memulai dan mencoba melakukan koneksi pada *server* pada port 21
2. Setelah koneksi terjadi, perintah PORT dikirimkan ke *server* untuk mengetahui *port* mana yang harus digunakan oleh *server* untuk melakukan koneksi data.
3. Untuk operasi tertentu, diperlukan pengiriman file untuk mengetahui mengapa saluran data dibuat. Untuk itu *server* membuat sebuah koneksi dari *port* 20 ke salah satu *port* dari *client*. Kemudian *server* membuat sebuah saluran data yang akan digunakan sebagai media pengiriman informasi.



Gambar 3.1 Pembuatan Saluran Data oleh Server

Selanjutnya adalah koneksi pasif dari *File Transfer Protocol* (FTP) dan perbedaan yang terdapat antara kedua koneksi dijelaskan sebagai berikut :

1. *Client* mencoba melakukan koneksi pada *server* melalui *port X* ke *port 21* pada *server*
2. *Server* mengirimkan perintah PASV untuk mengaktifkan mode pasif. Dan mengirimkan melalui *port* yang aktif port Z.
3. Untuk operasi tertentu, diperlukan pengiriman data, untuk mengetahui mengapa data dibuat. Untuk itu, *client* membuat sebuah koneksi pada sebuah *port* untuk sebuah *port* pada *client*. Ini merupakan jawaban dari *server* pada perintah PASV (*port Z*). Kemudian *client* membuat sebuah saluran data yang akan digunakan sebagai media pengiriman informasi.



Gambar 3.2 Pembuatan Saluran Data oleh Client

3.5 Data Masukan

Data masukan yang paling awal adalah untuk identifikasi user sebagai pengguna sistem. Hanya *user* yang terdaftar atau *anonymous user* saja yang bisa menggunakan sistem. Untuk calon *user* sistem yang belum terdaftar dapat melakukan registrasi melalui *administrator*.

3.6 Pembahasan Sistem Yang Dibangun

Sistem yang akan dibangun adalah aplikasi *client-server* yang bekerja pada jaringan lokal maupun jaringan Internet.

Untuk keamanan dari sistem yang akan dibangun, sistem diproteksi dengan *user name* dan *password*. Untuk koneksi *anonymous*, *user* memasukkan *user name* dan alamat *email* sebagai *password*.

Setelah melakukan proses *login*, *user* dapat melakukan perintah-perintah FTP untuk melakukan proses-proses yang terdapat pada File Transfer Protocol (FTP). Perintah-perintah FTP tidak lagi berupa teks yang akan diketik, tetapi visualisasi dari perintah yang akan dibuat pada aplikasi berupa tombol-tombol, menu, dan input yang visual. Proses yang ada pada File Transfer Protocol (FTP) yang dapat dilakukan oleh *user* adalah mengirim (*upload*), menyalin (*download*), mengubah nama file, mengubah nama direktori, menghapus file dan direktori (hanya pada *user* yang berhak, *anonymous user* tidak diizinkan).

3.6.1 WinInet API

Sistem yang dibangun menggunakan WinInet API dan dibangun untuk aplikasi berbasis sistem operasi Windows. WinInet API adalah sekumpulan fungsi yang memudahkan pembangunan aplikasi *internet* dan lebih mudah untuk dipahami. WinInet API dapat digunakan untuk pembangunan aplikasi *internet* yang akan bekerja dengan sumber daya yang ada pada *internet* melalui protokol HTTP, FTP, Gopher.

Keuntungan memakai WinInet API adalah sintak protokol tidak diperlukan dalam pembuatan aplikasi, begitu pula dengan Windows Socket API. Beberapa fungsi yang terdapat pada WinInet API adalah :

- ⊕ `InternetConnect` : Membuka koneksi FTP, Gopher dan HTTP pada server yang dituju.
- ⊕ `InternetReadFile` : Membaca file pada server yang dituju
- ⊕ `InternetWriteFile` : Menulis file pada server yang dituju
- ⊕ `FTPGetFile` : Menerima file dan menyimpan pada penyimpanan lokal
- ⊕ `FTPRenameFile` : Mengubah nama file pada server FTP

3.6.2 FTP API

WinInet API memberikan 12 (duabelas) fungsi untuk manipulasi dan navigasi file pada server FTP. Ketika bekerja dengan WinInet API, langkah-langkah proses terjadi beberapa kali. Urutan langkah proses yang terjadi adalah :

1. Inisialisasi WinInet API dengan memanggil fungsi InternetOpen
2. Melakukan koneksi pada server dan menerima respon untuk membuka koneksi dengan menggunakan fungsi InternetConnect.
3. Melakukan manipulasi dan navigasi terhadap file yang ada pada server
4. Memutuskan koneksi

Tabel berikut menerangkan masing-masing fungsi FTP pada WinInet API :

Tabel 3.3 Fungsi FTP pada WinInet API

Perintah [parameter]	Keterangan
FtpCreateDirectory	Membuat direktori baru pada server
FtpGetCurrentDirectory	Menampilkan direktori sekarang
FtpRemoveDirectory	Menghapus seluruh isi direktori
FtpSetCurrentDirectory	Mengganti ke direktori sekarang
FtpDeleteFile	Menghapus file pada server
FtpGetFile	Menyalin file dari server
FtpGetFileSize	Menampilkan ukuran file
FtpOpenFile	Membuka sebuah file untuk proses pembacaan dan penulisan
FtpPutFile	Menyalin file ke server
FtpRenameFile	Mengubah nama file di server
FtpCommand	Menyediakan sebuah tampilan untuk mengirimkan perintah ke server FTP

3.7 Fungsi Sistem Yang Dibangun

Fungsi-fungsi yang ada dalam Aplikasi Transfer File berbasis TCP/IP ini adalah :

1. Fasilitas pengiriman (*upload*) dan menyalin (*download*) file dari server FTP yang terhubung jaringan lokal maupun yang terhubung melalui jaringan *internet*.
2. User yang login dengan menggunakan *userid* dan *password*, memiliki hak akses yang penuh terhadap operasi yang akan dilakukan apabila sudah terkoneksi ke server FTP.
3. User yang login dengan menggunakan *anonymous user*, memiliki keterbatasan dalam hak akses terhadap operasi yang akan dilakukan apabila sudah terkoneksi ke server FTP. User hanya dapat melakukan proses *read*.
4. Sistem dapat menampilkan *access progress user* ke server dalam proses *read* ataupun *write* yang ditampilkan dalam *bandwidth speed*.

Dalam proses *read* ataupun *write*, koneksi internet sangat mempengaruhi proses ini. Untuk mengetahui apakah proses masih berjalan ataupun tidak, maka sistem menampilkan performa koneksi komputer dalam bentuk laporan kecepatan *bandwidth*.

5. Sistem dapat menampilkan status proses.

Pada saat user melakukan operasi *read* ataupun *write* pada server FTP, proses yang terjadi pada saat operasi ditampilkan untuk mengetahui apa-apa saja yang terjadi pada saat eksekusi, yaitu FTP.

BAB IV

PERANCANGAN SISTEM

Tahap perancangan sistem menggambarkan secara umum mengenai perancangan sistem Aplikasi Transfer File berbasis TCP/IP dan komponen yang termasuk di dalam perancangan. Komponen perancangan terdiri dari *Data Context Diagram*, *Data Flow Diagram*.

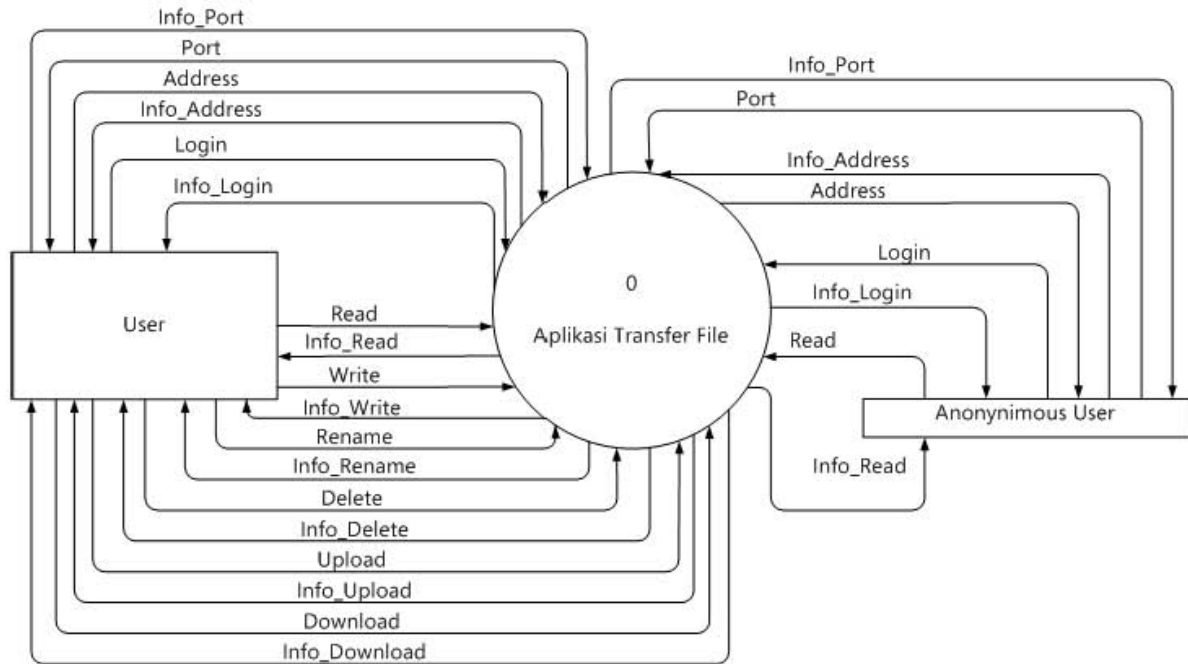
4.1 Gambaran Proses

Gambaran proses setiap aktivitas aplikasi transfer file berupa data context diagram dan data flow diagram.

1. *Data Context Diagram* (DCD), memperlihatkan alur informasi atau aliran data sistem secara global.
2. *Data Flow Diagram* (DFD), merupakan perluasan dari data *Data Context Diagram* dengan kata lain merupakan spesifikasi dari alur sistem yang dibuat.

4.1.1 Data Context Diagram

Dalam diagram ini menggambarkan proses aplikasi yang dibuat secara menyeluruh. Hal ini dapat dilihat pada gambar diagram di bawah (gambar 4.1), yang hanya mempunyai satu atribut proses dan dua entitas yang berhubungan langsung ke sistem. Diagram ini mewakili dan menggambarkan dari sistem yang dibuat secara menyeluruh.



Gambar 4.1 Data Context Diagram (DCD)

Pada DCD diatas *user* dan *anonymous user* memasukkan *user name* dan *password*. Kemudian sistem akan memverifikasi nama dan *password*. Sistem akan memproses lebih lanjut apabila nama dan *password* yang dimasukkan oleh *user* dan *anonymous user*, sistem akan memverifikasi *user* yang mempunyai hak penuh terhadap sistem dan *anonymous user* yang mempunyai hak untuk melakukan aktivitas *read* saja. Setelah nama dan password divalidasi, user dapat melakukan proses aktivitas yang ada pada sistem transfer file yang meliputi : *read file* atau direktori, *write file* atau direktori, *rename file* atau direktori, *download file* atau direktori, *upload file* atau direktori dan *delete file* atau direktori. Sedangkan aktifitas yang dapat dilakukan oleh anonymous user hanya untuk melakukan pembacaan file yang ada direktori yang telah ditetapkan oleh *server* aplikasi.

4.2 Data Flow Diagram

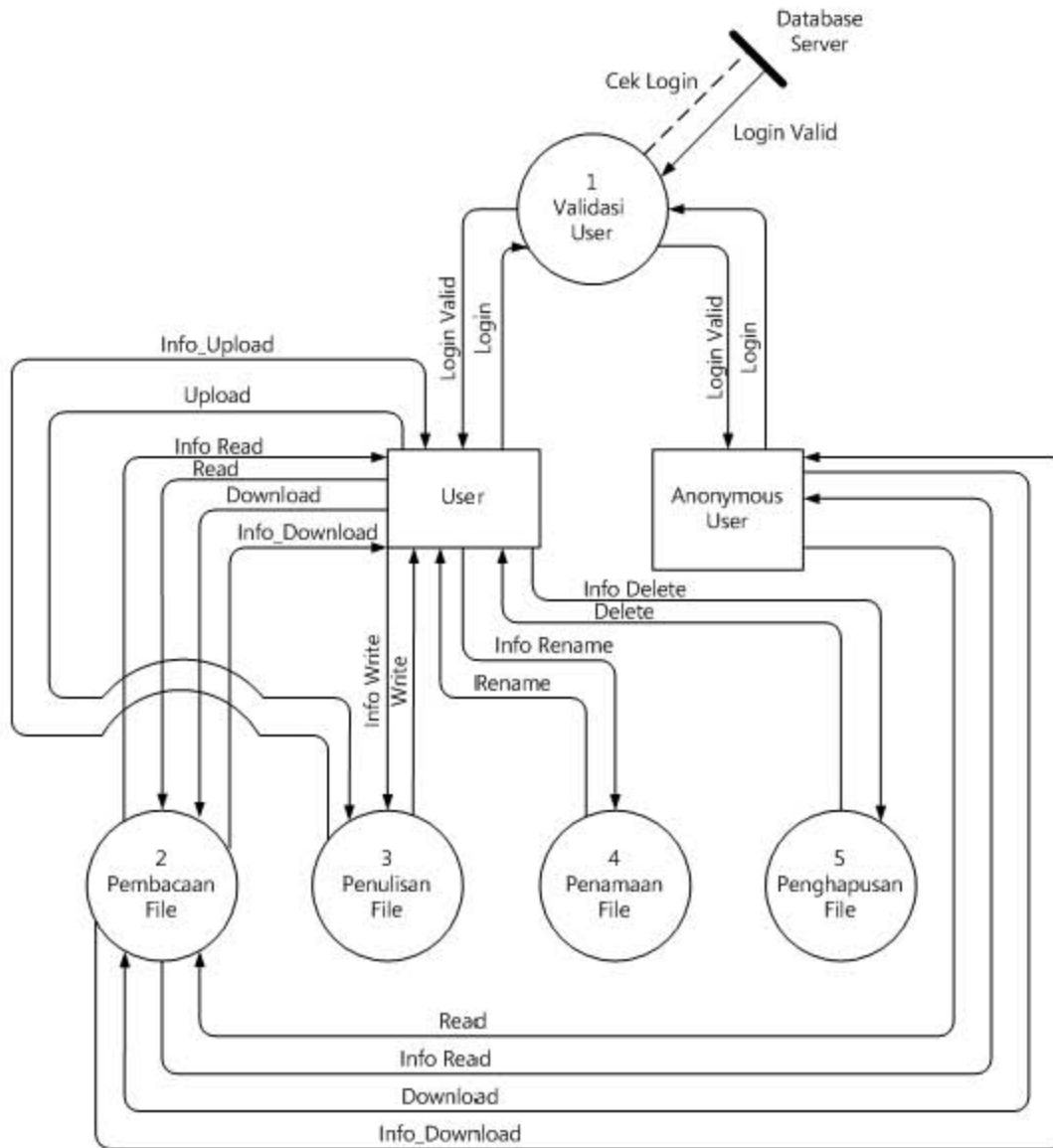
DFD digunakan untuk membuat model sebuah sistem dimana sistem yang dikembangkan adalah Aplikasi Transfer File berbasis TCP/IP yang berbentuk jaringan proses-proses yang saling berhubungan satu dengan yang lainnya.

4.2.1 Data Flow Diagram Level 1

Lanjutan dari *Data Context Diagram* di atas dapat dilihat pada *Data Flow Diagram* di bawah (Gambar 5.2). Pada *Data Context Diagram* akan terlihat entitas yang mana saja yang memasukkan data dan menerima data dari sistem, tetapi entitas tersebut terbatas pada memasukkan dan menerima data saja. Untuk *Data Flow Diagram* akan diketahui proses-proses apa saja yang terdapat dalam sistem.

Adapun yang termasuk dalam *Data Flow Diagram Level 1* ini adalah validasi pengguna, pembacaan data, penulisan data, penamaan data, penghapusan data. Pada proses 1, terdapat proses penyaringan nama dan *password*, apabila proses valid maka akan dilanjutkan pada proses berikutnya, tetapi jika proses penyaringan nama dan password dianggap tidak valid maka proses akan dihentikan sampai disini saja, tidak dilanjutkan ke proses berikutnya. Dimana pada proses 1 ini validasi dilakukan untuk menentukan hak akses dari pengguna sistem, hak akses pengguna terdiri dari : *user* yang terdaftar di server yang mempunyai hak akses penuh terhadap aplikasi transfer file. Proses 2 dilakukan setelah proses penyaringan atau validasi, kemudian user atau anonymous user berhak melakukan pembacaan dan penyalinan data. Proses 3 dilakukan setelah proses penyaringan atau validasi, di proses 3 ini pengguna yang berhak melakukan pembuatan direktori atau pun pengubahan isi file hanya boleh dilakukan oleh *user* yang terdaftar pada *server* aplikasi transfer file. Proses 4 dilakukan setelah proses penyaringan dan validasi, pada proses 4 ini user yang berhak adalah user yang terdaftar di server aplikasi transfer file, dimana pada proses 4 ini, user melakukan perubahan nama terhadap data yang ada. Proses 5 dilakukan setelah proses penyaringan dan validasi, dan user yang berhak adalah *user* yang terdaftar

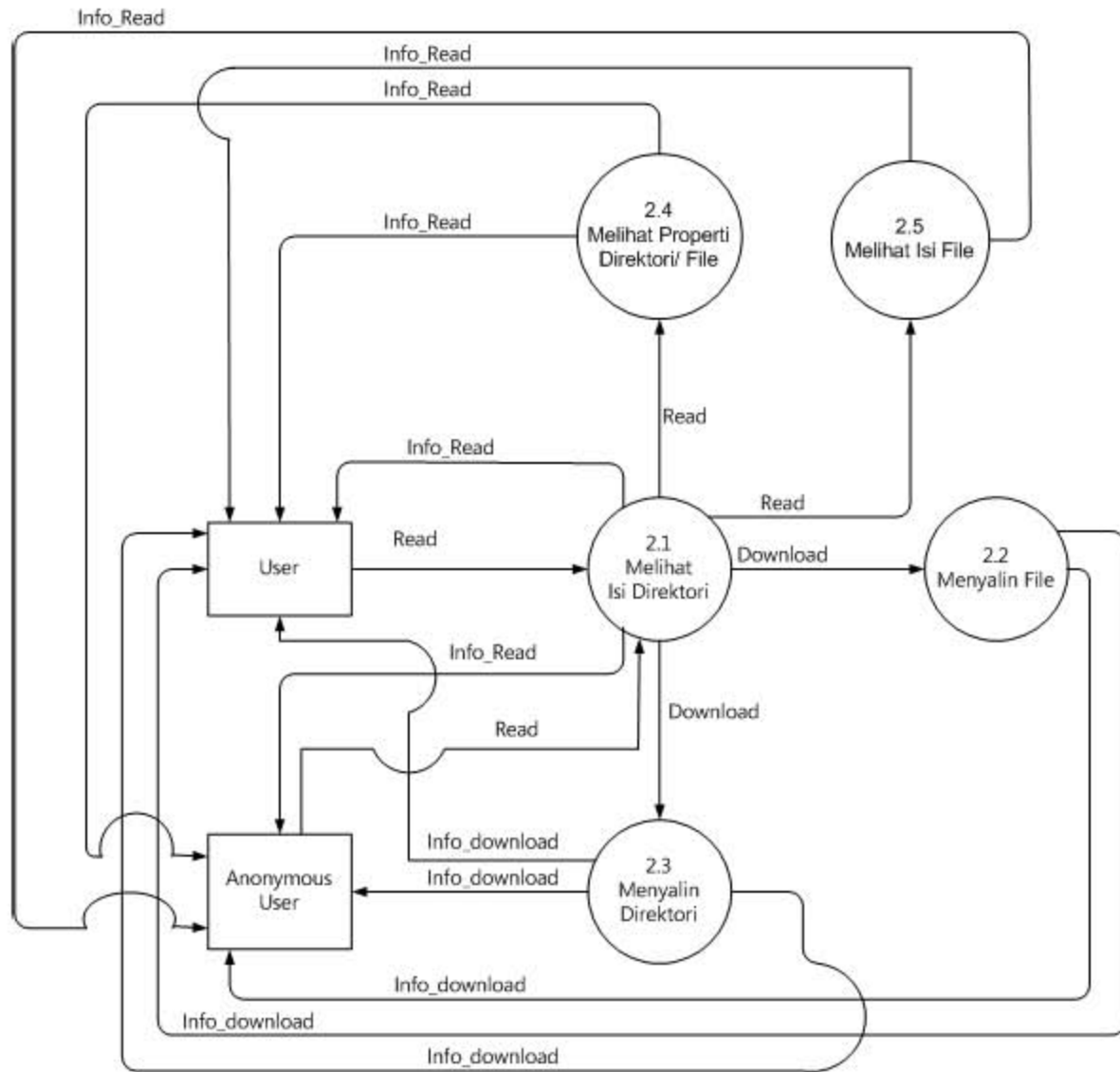
pada *server* aplikasi transfer file, user dapat melakukan penghapusan terhadap file atau direktori yang terdapat di *server*.



Gambar 4.2 Data Flow Diagram Level 1

4.2.2 Data Flow Diagram Proses 2 Level 2

Pada gambar di bawah ini (gambar 4.3) dapat dilihat dengan jelas mengenai pembacaan direktori dan file yang terdapat pada data flow diagram pada level 1 proses 2.

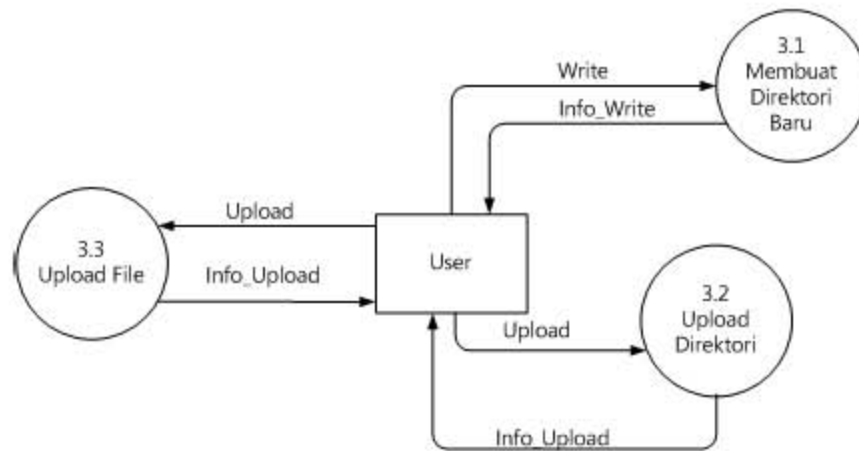


Gambar 4.3 Data Flow Diagram Proses 2 Level 2

4.2.3 Data Flow Diagram Proses 3 level 2

Pada gambar dibawah (gambar 4.4) menggambarkan proses yang lebih jelas dari *data flow diagram* pada level 1 proses 3 mengenai penulisan direktori dan penulisan file. Pada proses 3 level 2 ini terdapat tiga proses yang dilakukan dalam penulisan data, yaitu :

1. Membuat direktori baru : *User* membuat direktori yang baru di server.
2. Upload Direktori : *User* melakukan *upload* direktori lokal ke server.
3. Upload File : *User* melakukan *upload* file lokal ke server.

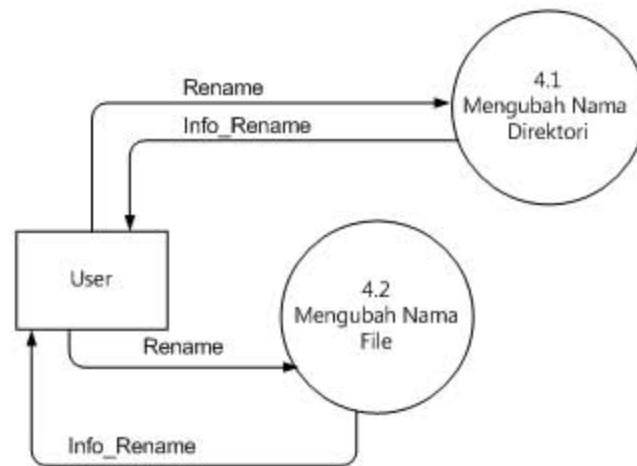


Gambar 4.3 Data Flow Diagram Proses 3 Level 2

4.2.4 Data Flow Diagram Proses 4 Level 2

Pada gambar di bawah ini (gambar 4.4) menggambarkan proses yang lebih jelas dari data flow diagram pada level 1 proses 4. Pada proses 4 level 2 terdapat dua proses yang terjadi dalam proses perubahan nama direktori dan file :

1. Mengubah nama file : *User* mengubah nama file yang terdapat pada *server*.
2. Mengubah nama direktori : *User* mengubah nama direktori yang terdapat di *server*.

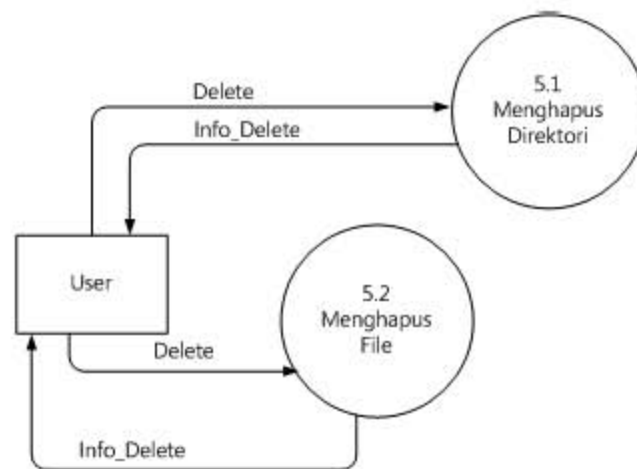


Gambar 4.4 Data Flow Diagram Proses 4 Level 2

4.2.5 Data Flow Diagram proses 5 Level 2

Pada gambar di bawah ini (gambar 4.5) menggambarkan proses yang lebih jelas dari data flow diagram pada level 1 proses 5. Pada proses 5 level 2 terdapat dua proses yang terjadi dalam proses penghapusan file :

1. Menghapus file : *User* menghapus file yang terdapat pada *server*.
2. Menghapus direktori : *User* menghapus direktori yang terdapat di *server*.



Gambar 4.5 Data Flow Diagram Proses 5 Level 2

4.3 Proses Specification

Pada bagian ini akan dijelaskan proses yang ada terdapat pada *data context diagram* dan *data flow diagram* yang telah dibuat diatas. Hal ini bertujuan untuk menjelaskan secara rinci tentang aplikasi yang dibuat.

1. Data Context Diagram

Diagram pada bagian ini menjelaskan aliran-aliran yang akan terjadi pada software secara umum, dimana pada bagian ini terdapat dua buah entitas dan masing-masing memiliki fungsi terhadap sistem. Entitas-entitas yang terdapat pada data context diagram ini adalah *user* dan *anonymous user*, dimana masing-masing entitas memiliki fungsi sebagai berikut :

- *User* memiliki fungsi untuk mengelola direktori dan file yang termasuk didalamnya *read*, *write*, *rename*, *delete*.
- *Anonymous user* memiliki fungsi terbatas yaitu hanya dapat melakukan *read*.

2. Data Flow Diagram Level 1

Diagram ini akan menjelaskan lebih rinci dari aliran-aliran yang terjadi pada data context diagram, dimana proses-proses aliran data yang mengalir dijelaskan disini.

Adapun aliran proses yang mengalir pada data flow diagram level 1 adalah sebagai berikut :

- a. *User* memasukkan nama dan *password* agar sistem mengenali bahwa *user* tersebut memiliki hak akses terhadap *server*.
- b. *Anonymous user* memasukkan nama dan password agar sistem dapat mengenali bahwa user tersebut memiliki hak akses terhadap *server*.
- c. Setelah data yang dimasukkan valid, kemudian masuk ke proses selanjutnya.

3. *Data Flow Diagram Level 2* Proses 2

Pada bagian level ini akan membahas mengenai proses yang terjadi pada level 2 yakni menurunkan data flow diagram level 1 proses 2. Pada bagian ini terdapat proses membaca isi direktori, menyalin file, menyalin isi direktori.

Adapun fungsi dari ketiga proses tersebut adalah :

- a. Melihat isi direktori berfungsi untuk membaca *file-file* yang terdapat di dalam direktori, *user* yang terdaftar memiliki hak akses yang penuh untuk melihat isi semua direktori. *Anonymous user* terbatas hanya pada direktori public atau direktori yang telah ditetapkan oleh *server*.
- b. Menyalin file berfungsi untuk menyalin *file* dari sistem, *user* yang terdaftar memiliki hak akses yang penuh untuk menyalin semua *file* yang terdapat di dalam sistem. *Anonymous user* terbatas hanya pada direktori *public* atau direktori yang telah ditetapkan oleh *server*.
- c. Menyalin isi direktori berfungsi untuk menyalin direktori atau isi keseluruhan dari direktori tersebut. *User* yang terdaftar memiliki hak akses yang penuh untuk menyalin isi semua direktori. *Anonymous user* tidak memiliki hak untuk menyalin direktori atau isi semua *file* yang terdapat ada direktori.
- d. Melihat properti direktori atau file berfungsi untuk melihat besar file atau direktori, dan melihat letak dari direktori atau file tersebut.
- e. Melihat isi file berfungsi untuk melihat isi file, aplikasi transfer file ini terbatas hanya untuk melihat file yang berbentuk atau memiliki format teks.

4. *Data Flow Diagram Level 2* Proses 3

Pada bagian level ini akan membahas mengenai proses yang terjadi pada level 2 yakni menurunkan data flow diagram level 1 proses 3. Pada bagian ini terdapat proses membuat file baru, membuat direktori baru, merubah isi data. Dimana pada proses ini hanya user yang terdaftar pada sistem yang berhak melakukan proses.

Adapun fungsi dari ketiga proses tersebut adalah :

- a. Membuat direktori baru berfungsi untuk membuat atau menyalin direktori ke server.
- b. Upload direktori berfungsi untuk melakukan upload direktori yang terdapat pada komputer lokal ke server.
- c. Upload file berfungsi untuk melakukan upload file yang terdapat pada komputer lokal ke server.

5. *Data Fow Diagram Level 2* Proses 4

Pada bagian level ini akan membahas mengenai proses yang terjadi pada level 2 yakni menurunkan data flow diagram level 1 proses 4. Pada bagian ini terdapat proses mengubah nama *file*, mengubah nama direktori. Dimana pada proses ini hanya *user* yang terdaftar pada sistem yang berhak melakukan proses.

Adapun fungsi dari kedua proses tersebut adalah :

- a. Mengubah *file* data berfungsi utnuak melakukan perubahan nama file yang terdapat pada *server*.
- b. Mengubah nama direktori berfungsi untuk melakukan perubahan nama pada direktori yang ada pada *server*.

6. *Data Flow Diagram Level 2* Proses 5

Pada bagian level ini akan membahas mengenai proses yang terjadi pada level 2 yakni menurunkan data flow diagram level 1 proses 5. Pada bagian ini terdapat proses menghapus file dan menghapus direktori. Dimana pada proses ini hanya *user* yang terdaftar pada sistem yang berhak melakukan proses.

Adapun fungsi dari kedua proses tersebut adalah :

- a. Menghapus *file* berfungsi untuk melakukan penghapusan pada *file* yang terdapat di *server*.
- b. Menghapus direktori berfungsi melakukan penghapusan direktori yang terdapat di *server*.

4.4 Pseudo Code

Proses-proses yang terdapat pada Data Flow Diagram yang dimulai dari proses satu sampai proses lima dituliskan dalam bentuk pseudo code dibawah ini.

4.4.1 PSPEC Level 1 Proses 1 : Validasi

Input	Nama, password
Output	User_valid, anonymoususer_valid
Proses	<pre>If (user.nama <> database server.nama) and (user.password <> database server.password) Then User_valid Call Pembacaanfile Penulisanfile Penamaanfile penghapusanfile else if (anonymoususer.nama <> anonymoususer.nama) and if (anonymoususer.password <> anonymoususer.password) then uservalid call Pembacaanfile</pre>
Keterangan	User berupa nama dan password harus dimasukkan dengan benar, jika tidak maka proses akan dihentikan.

4.4.2 PSPEC Level 2 Proses 2 : Pembacaan File

Input	User_valid, anonymoususer_valid, read
Output	Readvalid, downloadvalid
Proses	Input user_valid, anonymoususer_valid, Open file If file = ada then Readvalid call melihatisidirektori menyalinfile menyalinisidirektori endif downloadvalid
Keterangan	Proses ini dilakukan jika user_valid dan anonymoususer_valid lalu user dan anonymoususer melakukan proses selanjutnya yaitu : melihatisidirektori, menyalinfile dan menyalinisidirektori

4.4.3 PSPEC Level 2 Proses 3 : Penulisan File

Input	User_valid, write
Output	Writevalid, uploadvalid
Proses	Input user_valid, Open file, if file ada = then writevalid uploadvalid update file endif
Keterangan	Proses ini dilakukan jika user_valid lalu melakukan proses membuatfilebaru, membuatdirektoribaru dan merubahisidata

4.4.4 PSPEC Level 2 Proses 4 : Penamaan File

Input User_Valid, Rename

Output RenameValid

Proses Input User_valid,
Open file,
If file = ada then
RenameValid
Update file
endif

Keterangan Proses ini dilakukan jika user_valid lalu melakukan proses mengubahnamafile dan mengubahnamadirektori

4.4.5 PSPEC Level 2 Proses 5 : Penghapusan File

Input User_valid, Delete

Output DeleteValid

Proses Input User_valid,
Open file,
If file = ada then
DeleteValid
Update file
Endif

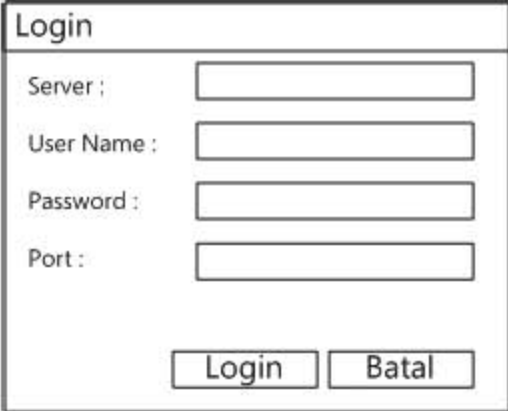
Keterangan Proses ini dilakukan jika user_valid lalu melakukan proses menghapusfile dan menghapusdirektori

4.5 Basis Data

Aplikasi transfer file yang dibangun ini tidak menggunakan basis data. Basis data yang dipakai hanya untuk memeriksa user yang terdaftar pada server sistem dimana basis data untuk *user* ini terdapat pada *server* sistem.

4.6 Perancangan Antar Muka

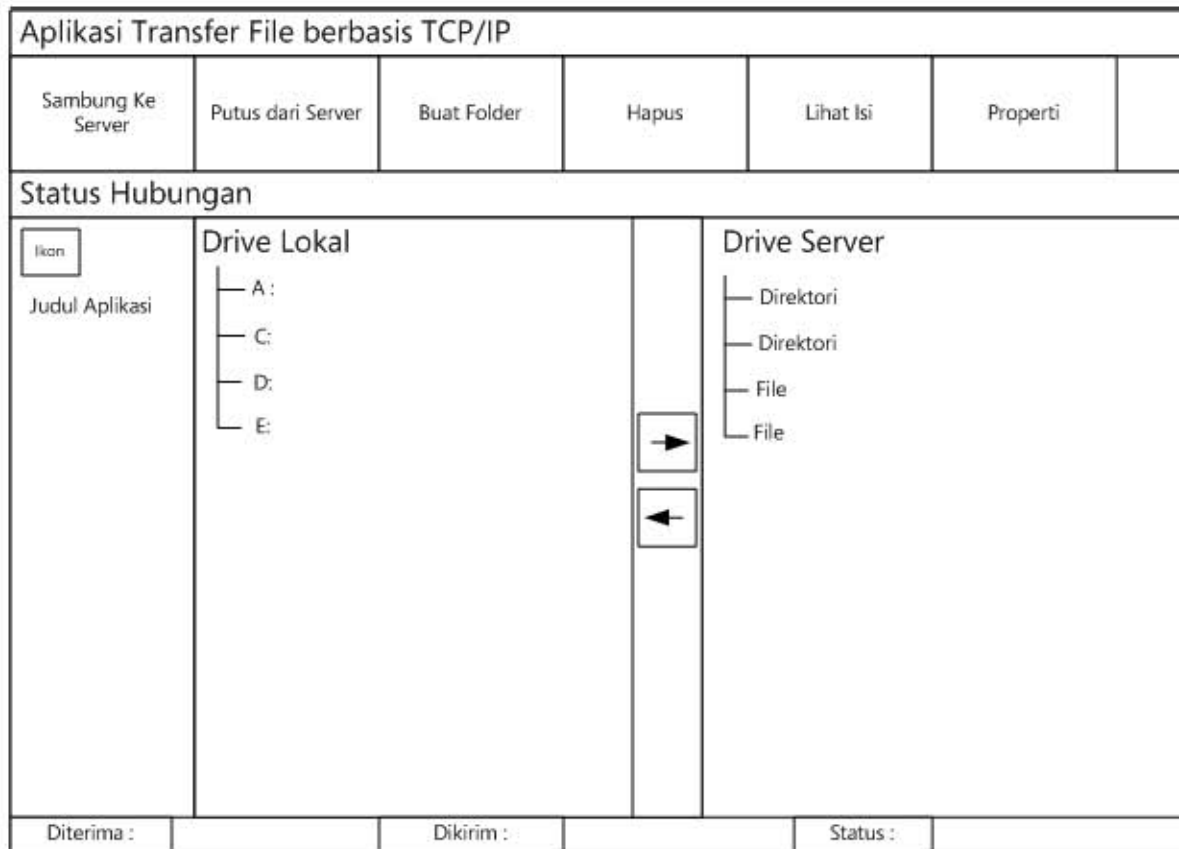
Perancangan antar muka menjelaskan rutinitas program yang akan dijalankan oleh sebuah sistem komputerisasi untuk menjelaskan interaksi antara pemakai dengan program yang dibuat.



The image shows a rectangular dialog box titled "Login". Inside the box, there are four labels with corresponding input fields: "Server :", "User Name :", "Password :", and "Port :". Each label is followed by a rectangular text input field. At the bottom of the dialog box, there are two buttons: "Login" on the left and "Batal" on the right.

Gambar 4.6 Desain Menu Login

Pada menu login, *user* dan *anonymous user* memasukkan *address server* FTP yang dituju, kemudian memasukkan data login berupa *User Name* dan *Password* dan yang terakhir adalah *port* FTP.



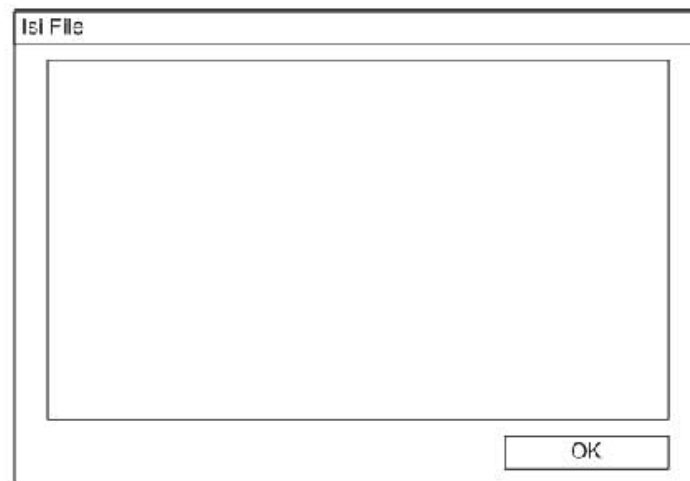
Gambar 4.7 Desain Menu Utama

Gambar di atas adalah tampilan utama dari aplikasi, dimana terdapat tombol navigasi yang terletak pada bagian atas, kemudian *Tree View* untuk *drive server* dan *drive* lokal. Kemudian bagian paling bawah adalah keterangan yang menunjukkan jumlah transfer yang dilakukan selama koneksi dan status dari koneksi yang sedang dilakukan.



Gambar 4.8 Desain Menu Properti

Menu disain properti adalah menu untuk menunjukkan keberadaan suatu file atau direktori dan menampilkan ukuran file atau direktori tersebut.



Gambar 4.9 Desain Menu Isi File

Apabila ada file dengan jenis text maka file akan dengan mudah ditampilkan dengan menggunakan menu isi file, seperti pada gambar diatas.

BAB V

IMPLEMENTASI

5.1 Lingkungan Implementasi

Lingkungan implementasi meliputi lingkungan perangkat keras (*hardware*) dan lingkungan perangkat lunak (*software*).

5.1.1 Lingkungan Perangkat Keras (*Hardware*)

Sistem yang telah dibuat diaplikasikan ke dalam lingkungan operasi yang membutuhkan perangkat pendukung sistem aplikasi. Perangkat ini meliputi perangkat lunak (*software*) dan perangkat keras (*hardware*) untuk sisi server dan sisi client dalam pengoperasian sistem yang baru ini, dengan kebutuhan minimal sebagai berikut :

1. Perangkat Keras untuk *Server*

- a. Processor Pentium III 500 Mhz atau lebih.
- b. Hard Disk 5 Mb untuk aplikasi.
- c. Memori 128 Mb.
- d. Monitor SVGA 1024x768

2. Perangkat Keras Untuk *Client*

Spesifikasi *hardware* yang digunakan untuk membangun sistem ini adalah :

- a. Processor Pentium III 500 Mhz atau lebih.
- c. Hard Disk 5 Mb untuk aplikasi.
- c. Memori 128 Mb.
- d. Monitor SVGA 1024x768

5.1.2 Lingkungan Perangkat Lunak (*Software*)

1. Spesifikasi software yang digunakan untuk membangun sistem ini adalah :
 - a. Sistem Operasi : Microsoft™ Windows XP
 - b. Program Aplikasi : Borland Delphi 7.0
2. Spesifikasi software pada *server* yang digunakan untuk membangun sistem ini adalah :
 - a. Sistem Operasi : Microsoft™ Windows XP
 - b. Program FTP *Server* : Cerberus FTP Server Version 2.32

5.1.3 Implementasi Basis Data

Aplikasi transfer file yang dibangun ini tidak menggunakan basis data. Basis data yang dipakai hanya untuk memeriksa user yang terdaftar pada server sistem dimana basis data untuk user ini terdapat pada server sistem.

5.2 Antar Muka Program

Berikut ini akan ditampilkan tampilan Implementasi Antar muka (Interface) untuk Aplikasi Transfer File Berbasis TCP/IP ini adalah sebagai berikut :

5.2.1 Form Login

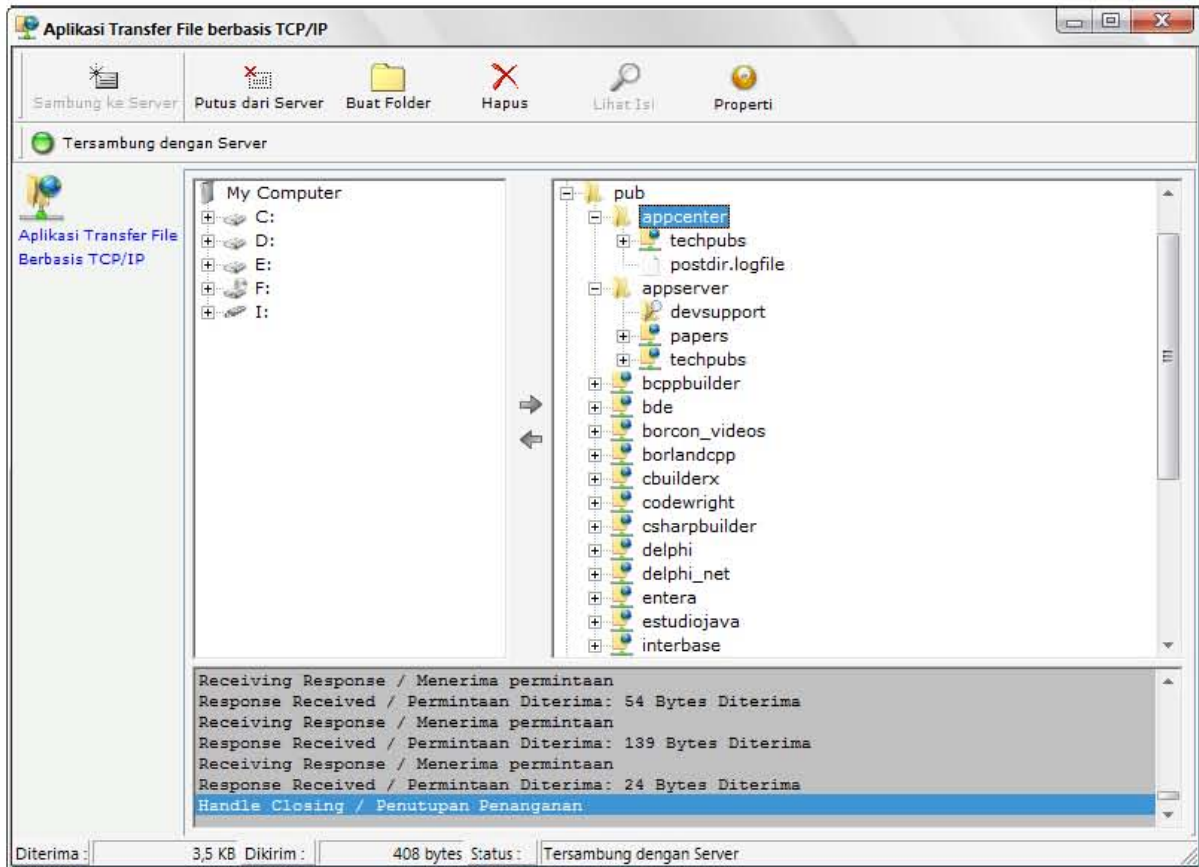


The image shows a Windows-style dialog box titled "Login". It contains four input fields: "Server" with the value "ftp.borland.com", "User Name" with the value "anonymous", "Password" which is masked with asterisks, and "Port" with the value "21". At the bottom of the dialog, there are two buttons: "Login" and "Batal".

Gambar 5.1 Form Login

Pada Gambar 5.1 di atas, user harus memasukkan alamat *server*, *user name*, *password* dan *port* pada kotak yang telah disediakan. Setelah itu tekan tombol *Login*. Jika berhasil, maka akan masuk ke menu utama. Namun jika tidak berhasil, akan ditampilkan pesan kesalahan. Tombol **BATAL** sendiri berfungsi apabila kita tidak jadi memasukkan alamat *server*, *user name*, *password* dan *port*.

5.2.2 Form Utama



Gambar 5.2 Form Utama

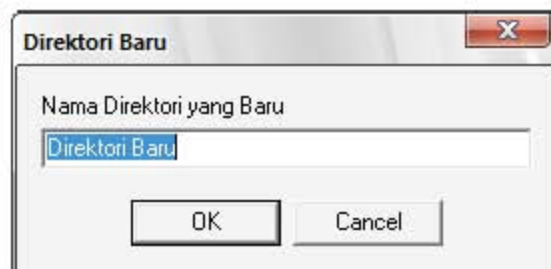
Struktur dari menu utama adalah :

1. Menu Sambung ke Server : menampilkan form login untuk aktifitas login.
2. Menu Putus Dari Server : memutuskan hubungan dari server.
3. Menu Buat Folder : menampilkan form konfirmasi sekaligus penamaan direktori baru yang akan dibuat

-
4. Menu Hapus : menampilkan form konfirmasi dari penghapusan file atau direktori yang ada di server.
 5. Menu Lihat Isi : menampilkan form untuk melihat isi suatu file yang berbentuk teks.
 6. Menu Properti : menampilkan form properti untuk menampilkan properti file atau direktori.

Pada form utama terdapat 2 (dua) buah *Tree View*, yang menampilkan isi masing-masing direktori yang terdapat di *hard disk* lokal maupun yang di *server*. Pada bagian bawah form utama terdapat status koneksi yang berisikan perintah yang sedang dieksekusi pada *server*. Dan pada bagian paling bawah terdapat status yang menampilkan ukuran byte yang dikirim dan yang diterima dan status koneksi.

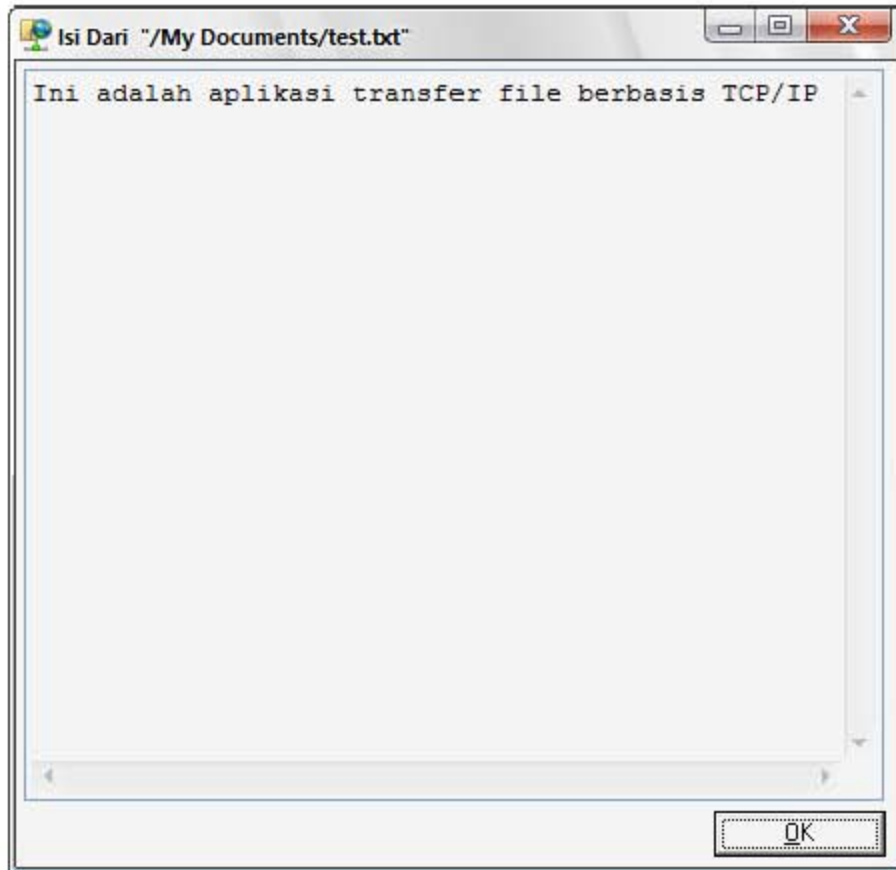
5.2.3 Form Direktori Baru



Gambar 5.3 Form Direktori Baru

Form Direktori Baru ini berfungsi sebagai konfirmasi sekaligus sebagai form untuk masukan nama untuk direktori baru yang akan dibuat.

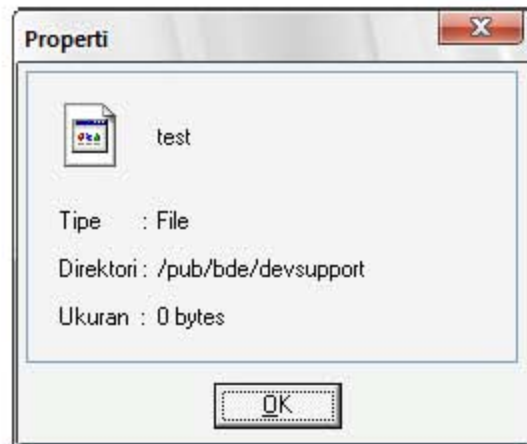
5.2.4 Form Lihat Isi



Gambar 5.4 Form lihat Isi

Form ini berguna untuk melihat isi file. Aplikasi Transfer File Berbasis TCP/IP ini terbatas untuk melihat isi file yang berformat atau berbentuk teks.

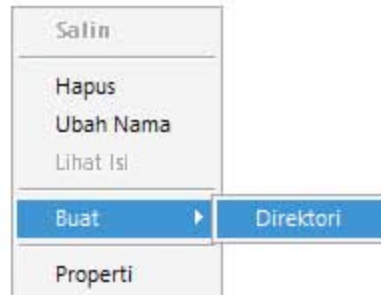
5.2.5 Form Properti



Gambar 5.5 Form Properti

Form Properti ini berfungsi untuk melihat properti suatu file atau direktori. Sama seperti menu properti yang terdapat pada aplikasi Windows Explorer, Microsoft™ Windows XP, tetapi Aplikasi Transfer File Berbasis TCP/IP ini terbatas hanya untuk melihat nama file atau direktori, tipe file atau direktori, letak file atau direktori, ukuran file atau direktori.

5.2.6 Pop Up Menu

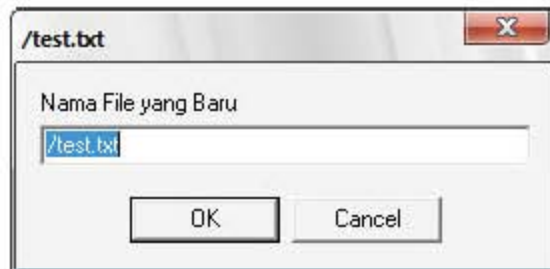


Gambar 5.6 Pop Up Menu

Pop Up Menu Aplikasi Transfer File Berbasis TCP/IP muncul ketika klik kanan dilakukan pada *Tree View server*. Dimana menu dari pop up menu ini terdiri dari

1. Salin berfungsi untuk melakukan penyalinan file yang terdapat pada server ke drive lokal.
2. Hapus berfungsi untuk melakukan penghapusan file atau direktori yang terdapat pada server.
3. Ubah Nama berfungsi untuk rename atau pengubahan nama yang terdapat pada server.
4. Lihat Isi berfungsi untuk menampilkan isi dari suatu file yang berbentuk teks.
5. Buat Direktori berfungsi untuk menampilkan form konfirmasi pembuatan direktori dan masukan nama direktori baru yang akan dibuat.
6. Properti berfungsi untuk menampilkan form properti.

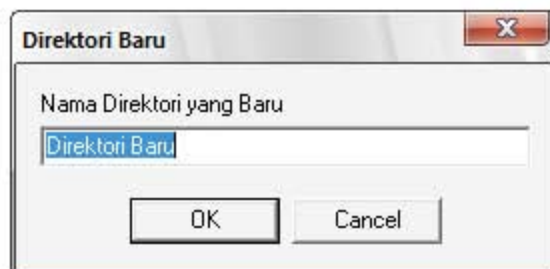
5.2.7 Form Nama *File* Baru



Gambar 5.7 Form Nama *File* Baru

Form Nama *File* Baru ini berfungsi untuk masukan untuk *file* yang akan di *rename* atau *file* yang akan diubah namanya.

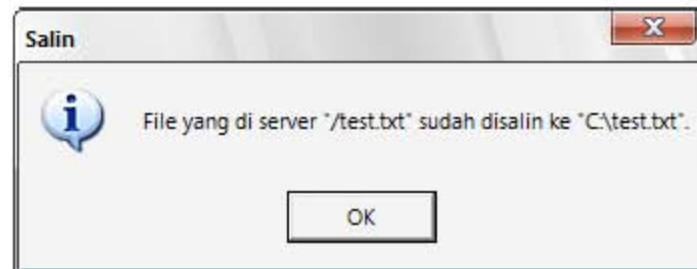
5.2.8 Form Nama Direktori Baru



Gambar 5.8 Form Nama Direktori Baru

Form Nama Direktori Baru ini berfungsi untuk masukan untuk direktori yang akan di *rename* atau direktori yang akan diubah namanya.

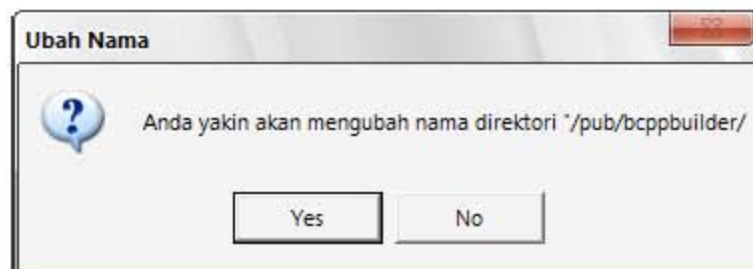
5.2.9 Form Konfirmasi Salin



Gambar 5.9 Form Konfirmasi Salin

Form ini berfungsi untuk menampilkan konfirmasi bahwa file yang disalin dari server telah sukses.

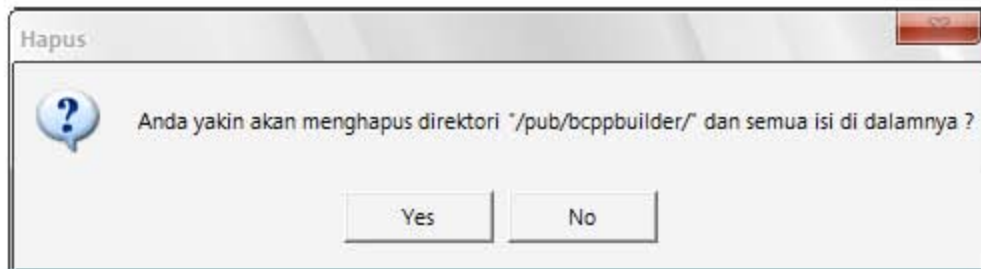
5.2.10 Form Konfirmasi Ubah Nama



Gambar 5.10 Form Konfirmasi Ubah Nama

Form ini berfungsi untuk konfirmasi apakah user setuju untuk melakukan perubahan nama atau rename terhadap file atau direktori.

5.2.11 Form Konfirmasi Hapus



Gambar 5.11 Form Konfirmasi Hapus

Form ini berfungsi untuk konfirmasi apakah user setuju untuk melakukan penghapusan terhadap file atau direktori.

BAB VI

PENUTUP

6.1 Kesimpulan

Aplikasi Client Server berbasis TCP/IP ini merupakan sebuah sistem transfer file yang menggunakan protokol TCP/IP dalam melakukan hubungan antar client dan server. Aplikasi ini dapat menangani perintah *File Transfer Protocol* (FTP) yaitu upload dan download serta perintah FTP lainnya.

Pembangunan aplikasi ini dibuat dengan menggunakan metoda waterfall. Dari awal pembangunan aplikasi, mulai dari tahap analisa, perancangan, sampai implementasi dapat diambil kesimpulan sebagai berikut:

1. Pembangunan Aplikasi Transfer File berbasis TCP/IP ini menyediakan perintah *File Transfer Protocol* (FTP) seperti upload, download, write, read, rename dan delete.
2. Dengan dibuatnya Aplikasi Transfer File berbasis TCP/IP ini diharapkan efisiensi dalam proses manajemen file yang terhubung dengan intranet dan internet dapat menjadi lebih mudah.
3. Aplikasi ini diharapkan dapat mengurangi waktu dalam proses pengetikan perintah *File Transfer Protocol* yang menggunakan command line dan Command Prompt yang terdapat di Microsoft® Windows™ XP.
4. Pada Aplikasi Transfer File ini terdapat perintah Lihat Isi yang masih terbatas pada format file yang berbentuk teks.
5. Pada Aplikasi Transfer File ini terdapat perintah Lihat Properti yang terbatas hanya untuk melihat ukuran file, jenis file, letak file.
6. Pada Aplikasi Transfer File apabila suatu perintah dieksekusi maka terdapat proses menunggu sampai perintah yang sedang dieksekusi selesai.

6.2 Saran

Aplikasi Transfer File ini belum merupakan sistem yang sempurna, sehingga dibutuhkan penyempurnaan sistem secara berkesinambungan sesuai dengan kebutuhan. Untuk pengembangan Aplikasi Transfer File ini mahasiswa disarankan :

1. Pada Aplikasi Transfer File ini terdapat keterbatasan pada saat melihat isi suatu file, terbatas pada format file berbentuk teks. Diharapkan dalam pengembangannya format file yang didukung bukan hanya berbentuk teks saja seperti gambar dan di asosiakan dengan aplikasi eksternal yang dapat menangani format file tersebut.
2. Pada Aplikasi Transfer File ini terdapat keterbatasan pada saat melihat properti suatu file atau direktori, diharapkan dalam pengembangannya dapat ditambahkan informasi yang detil mengenai file atau direktori tersebut.
3. Pada Aplikasi Transfer File ini terdapat keterbatasan pada saat suatu perintah dieksekusi, maka aplikasi menunggu perintah tersebut selesai dilaksanakan atau tidng mendukung *multitasking*. Diharapkan dalam pengembangannya Aplikasi dapat melakukan beberapa perintah sekaligus.