APLIKASI SECURITY UNTUK MICROSOFT WINDOWS

LAPORAN TUGAS AKHIR

Diajukan Untuk Memenuhi Syarat Kelulusan
Pada Jurusan Teknik Informatika Fakultas Teknik
Universitas Widyatama Bandung
Program Pendidikan Strata Satu (S-1)

Oleh:
WITTJE
0600034

JURUSAN TEKNIK INFORMATIKA
FAKULTAS TEKNIK UNIVERSITAS WIDYATAMA
BANDUNG
2006

ABSTRAKSI

Sistem keamanan sistem operasi *windows* sangat diperlukan untuk menghindari terjadinya hal-hal yang tidak diharapkan, seperti perubahan konfigurasi dan *setting* pada sistem operasi *windows,* yang secara otomatis dapat mengubah nilai-nilai *registry windows* yang dapat berakibat fatal pada sistem operasi *windows.*

Aplikasi *Security* untuk *Microsoft windows* dirancang untuk memudahkan *administrator* untuk mengunci beberapa fungsi windows dan *file executable.* Aplikasi ini dapat menyembunyikan *drives* yang dianggap sebagai tempat penyimpanan data penting.

Proses pembangunan aplikasi ini dikembangkan dengan menggunakan metoda *prototype,* tahap analisis dan perancangannya menggunakan perancangan terstruktur.

Aplikasi *Security* untuk *Microsoft windows* ini, dirancang dengan menggunakan *programming tool* Borland Delphi 7.0.

Kata kunci: *Security,* fungsi windows, file executable, drives, registry

ABSTRACT

Security system for windows operating system is very recommended for avoid trouble occured, like modify configuration and setting windows operating system, can be change the windows registry value automatically, may be result fatal error occurred in windows operating system.

The Security application for Microsoft windows designed to help administaror to lock some windows functions and file*.exe. The application can hidden drives for important data storage.

This application development process is developed by using prototyping methodology, analysis and designed using structured methodology.

The Security application for Microsoft windows, designed with using programming tool Borland Delphi 7.0.

Key words: Security, windows functions, f ile executable, drives, registry

BAB I

PENDAHULUAN

1.1 Latar Belakang Masalah

Banyak hal yang tidak menyenangkan yang bisa terjadi pada komputer kita. Misalnya ada orang yang baik sengaja atau pun tidak sengaja menghapus *file-file* penting kita. Atau merubah berbagai konfigurasi pada *windows.* Contohnya, mengakses komponen-komponen yang tersedia di dalam *control panel* untuk merubah beberapa konfigurasi *windows.* Selain itu bisa saja orang lain mengakses aplikasi yang digunakan untuk mengolah data penting Anda untuk melihat data-data yang bersifat rahasia maupun memodifikasinya. Kejadian seperti itu sangat mungkin terjadi bila komputer Anda merupakan komputer "umum". Maksudnya banyak orang lain yang turut memakai, misalnya teman, adik atau saudara. Biasanya, seorang pemula (dalam hal komputer) biasanya sangat senang mencoba-coba segala sesuatu pada komputer.

Jadi, untuk menghindari hal-hal yang tidak kita inginkan yang terjadi pada komputer kita, maka akan dibangun aplikasi *security* untuk sistem operasi yang khususnya berbasis *microsoft windows 98* yang masih banyak kelemahan dalam hal keamanan khususnya pembatasan wewenang *user*. aplikasi ini memudahkan *administrator* untuk membatasi wewenang *user* dalam hal pemakaian sistem operasi *windows* yang terinstalasi di komputer tersebut. Dan aplikasi ini dilengkapi dengan perlindungan *password* untuk mengaksesnya dan aplikasi ini beroperasi secara tersembunyi, sehingga user tidak dapat mengaksesnya secara bebas untuk mengaktifkan fungsi-fungsi *windows* yang telah dinonaktifkan, menampilkan kembali *drives* yang disembunyikan, mengaktifkan kembali program aplikasi yang telah dikunci oleh *administrator*.

1.2 Rumusan Masalah

Perumusan masalah pada pembangunan aplikasi *security* untuk *microsoft windows*, sebagai berikut:

1. Bagaimana membangun aplikasi yang dapat digunakan *administrator* dalam membatasi wewenang *user* dalam hal pemakaian komputer yang berbasis sistem operasi *windows.*
2. Bagaimana aplikasi tersebut tidak dapat diakses *user* secara bebas.

1.3 Batasan masalah

Batasan masalah yang ada pada penelitian ini ialah:

a. Hanya beberapa fungsi windows yang dapat dinonaktifkan.
   Fungsi-fungsi *windows* yang dapat dinonaktifkan yaitu :
   1. *Task Manager*
   2. *Taskbar Properties*
   3. *Registry editor*
   4. *Run Command*
   5. Fungsi pencarian pada *start menu*
   6. *Control Panel*
   7. Menghilangkan *Recent Documents* pada *start menu*
   8. Membatasi wewenang untuk *Log Off Option*
   9. *Display setting*
   10. *CD-ROM Autorun Function*
b. Menyembunyikan *drives* yang dianggap sebagai tempat penyimpan data penting
c. Hanya aplikasi yang berupa *file* *.exe yang dapat diproteksi.
d. Tidak mencakup masalah keamanan jaringan komputer.

1.4 Tujuan Penelitian

Berdasarkan latar belakang masalah tersebut, maka tujuan penelitian ini, adalah:

1. Membangun aplikasi *security* untuk *microsoft windows* yang dapat digunakan untuk membatasi wewenang *user.*

2. Membangun aplikasi *security* untuk *microsoft windows* yang dapat menyembunyikan *drives* yang dianggap sebagai tempat penyimpan data penting.
3. Membangun aplikasi *security* untuk *microsoft windows* yang dapat menonaktifkan *program* yang berupa *file*.exe* yang dianggap sebagai program pengolahan data penting.

1.5 Metodologi Penelitian

Paradigma pembangunan perangkat lunak yang digunakan dalam penelitian ini, yaitu *prototyping model*.

Sedangkan metode pengumpulan data yang digunakan antara lain:

1. Observasi yaitu pengamatan langsung terhadap sistem yang berjalan.
2. Studi literatur yaitu mempelajari teori yang berhubungan dengan permasalahan yang diteliti untuk membangun suatu program aplikasi.

1.6 Sistematika Penulisan

Sistematika penulisan yang akan dibahas pada penelitian ini adalah :

BAB I Pendahuluan, menjelaskan mengenai latar belakang masalah, rumusan masalah, batasan masalah, tujuan tugas akhir, metodologi penelitian, dan sistematika penulisan.

BAB II Landasan Teori, membahas tentang teori keamanan, otentifikasi pemakai, mekanisme proteksi sistem komputer, metode rekayasa perangkat lunak dengan model prototipe, analisis menggunakan metode data flow oriented, perangkat lunak dan teori software aplikasi yang digunakan untuk membangun sistem dan membahas secara singkat mengenai bahasa pemrograman yang digunakan dalam mengimplementasikan sistem yang dibangun.

BAB III Analisis, menguraikan tentang analisis dalam membangun aplikasi *security* untuk *microsoft windows*. Analisis sistem yang dilakukan meliputi identifikasi masalah, deskripsi pemakai, deskripsi sistem dan membahas *Data Context Diagram* (DCD), *Data Flow Diagram* (DFD), *Process Specification* (PSPEC), *Control Flow Diagram* (CFD), Control Specification (CSPEC) dan kamus data.

BAB IV Perancangan, menjelaskan mengenai rancangan antarmuka aplikasi *security* untuk *microsoft windows* dan perancangan *arsitektural*.

BAB V Implementasi sistem, menjelaskan mengenai lingkungan implementasi, implementasi antar muka beserta petunjuk penggunaan modul-modul yang terdapat dalam aplikasi yang dibangun.

BAB VI Kesimpulan dan saran, berisi kesimpulan dan saran-saran yang dinilai perlu untuk memperbaiki dan mengembangkan aplikasi *security* untuk *microsoft windows.*

BAB II
LANDASAN TEORI

2.1 Keamanan

Keamanan sistem operasi merupakan bagian masalah keamanan sistem komputer secara total. Keamanan sistem komputer adalah untuk menjamin sumber daya tidak digunakan atau dimodifikasi oleh orang yang tidak diotorisasi. Pengamanan termasuk masalah teknis, manajerial, legalitas dan politis.

Keamanan sistem terbagi menjadi 3 yaitu :

1. **Keamanan eksternal**

   Keamanan eksternal berkaitan dengan pengamanan fasilitas komputer dari penyusup dan bencana.

2. **Keamanan *interface* pemakai**

   Keamanan *interface* pemakai berkaitan dengan identifikasi pemakai sebelum pemakai diijinkan mengakses program dan data yang disimpan.

3. **Keamanan internal**

   Keamanan internal berkaitan dengan pengamanan beragam kendali yang dibangun pada perangkat keras dan sistem operasi yang menjamin operasi yang handal.

2.1.1 Masalah-masalah keamanan

Terdapat dua masalah keamanan yang penting, yaitu:

1. Kehilangan data
2. Penyusup

Kehilangan data dapat disebabkan antara lain:

1. Bencana, contohnya:
   - Kebakaran
   - Banjir
   - Gempa bumi
   - Kerusuhan
   - Ketidak stabilan listrik

2. Kesalahan perangkat keras dan perangkat lunak, contohnya:
   - *Disk drive* tidak terbaca
   - Kesalahan program (*bugs*)
   - Beberapa aksesoris komputer tidak dapat dikenali sistem operasi.
3. Kesalahan/kelalaian manusia
   - Kesalahan pemasukan data
   - Eksekusi proses program yang salah
   - Kehilangan *data storage*
   - Salah *format Harddisk*

Kehilangan data dapat diatasi dengan mengelola beberapa *backup* dan *backup* dapat ditempatkan pada tempat yang khusus menyimpan data cadangan.

Penyusup, terdiri dari:

1. Penyusup pasif, yaitu yang membaca data yang tak diotorisasi.
2. Penyusup aktif, yaitu yang mengubah data yang tak diotorisasi.

Kategori penyusupan, terdiri dari:

1. Lirikan mata pemakai non-*user,* contohnya lirikan non-*user* pada saat *user* mengetikkan *password* untuk dapat mengakses fasilitas yang bukan haknya.
2. *Spyware* yang dirancang untuk mendapatkan ataupun mengubah data yang diinginkan penyusup.

2.1.2 Ancaman-ancaman keamanan

Kebutuhan keamanan sistem komputer dikategorikan menjadi 3 aspek, yaitu:

1. Kerahasiaan

Kerahasiaan adalah keterjaminan bahwa informasi di sistem komputer hanya dapat diakses oleh pihak-pihak yang diotorisasi dan modifikasi tetap menjaga konsistensi dan keutuhan data di sistem.

2. Integritas

Integritas adalah keterjaminan bahwa informasi di sistem komputer hanya dapat dimodifikasi oleh pihak-pihak yang diotorisasi.

3. **Ketersediaan**

   Ketersediaan adalah keterjaminan bahwa sumber daya sistem komputer tersedia bagi pihak-pihak yang diotorisasi saat diperlukan.

Tipe-tipe ancaman terhadap keamanan sistem komputer dapat dimodelkan dengan memandang fungsi sistem komputer sebagai penyedia informasi. Berdasarkan fungsi ini, ancaman terhadap sistem komputer dikategorikan menjadi empat ancaman, yaitu:

1. **Interupsi**

   Sumber daya sistem komputer dihancurkan atau menjadi tak berfungsi.

   Contoh:

   - Penghancuran bagian perangkat keras, seperti *harddisk*
   - Pemutusan kabel komunikasi jaringan

2. **Intersepsi**

   Pihak tak diotorisasi dapat mengakses sumber daya. Intersepsi merupakan ancaman terhadap kerahasiaan. Pihak tak diotorisasi dapat berupa orang atau program komputer.

   Contoh:

   - Penyadapan untuk mengambil data rahasia
   - Mengkopi file tanpa diotorisasi

3. **Modifikasi**

   Pihak tak diotorisasi tidak hanya mengakses tapi juga merusak sumber daya.

   Contoh:

   - Mengubah nilai-nilai file data

4. **Fabrikasi**

   Pihak tak diotorisasi menyisipkan data-data palsu ke program.

   Contoh:

   - Penambahan record ke *database* program.

2.2 Otentifikasi Pemakai

Kebanyakan proteksi didasarkan asumsi sistem mengetahui identitas *user*. Masalah identifikasi pemakai ketika *log in* disebut otentifikasi pemakai.

2.2.1 *Password*

Password merupakan kata kunci *user* yang harus diingat, dan diketikkan pada *text editor* yang telah disediakan, untuk mengakses sistem komputer.

Teknik *password* ini mempunyai kelemahan yang cukup banyak. *User* cenderung memilih *password* yang mudah diingat. Penyusup yang kenal dengan *user* dapat mencoba *log in* dengan sesuatu yang diketahuinya mengenai *user*, contohnya: tanggal lahir, nomor plat kendaraan, angka *favorite*.

Percobaan Morris dan Thompson menyatakan proteksi *password* dapat ditembus dengan mudah.

Percobaan yang dilakukan adalah:

- Memasukkan *string-string* pendek karakter acak, identitas *user*.
- Isian di *file* dicocokkan dengan *file* password.

Hasil percobaan menunjukkan lebih dari 86% cocok dengan *password* yang digunakan *user* di *file password*.

Upaya untuk lebih mengamankan proteksi *password,* antara lain:

1. *Salting*
2. *One-time password*
3. Satu daftar panjang pertanyaan dan jawaban
4. Tanggapan-tanggapan

Salting

Merupakan penambahan *string* pendek ke *string password* yang diberikan *user* sehingga mencapai panjang *password* tertentu.

One-Time Password

Pemakai harus mengganti *password* secara teratur. Upaya ini untuk membatasi peluang *password* telah diketahui atau dicoba-coba *user* lain. Bentuk ekstrim pendekatan ini adalah *one-time password,* yaitu *user* membuat satu buku berisi

daftar *password.* Setiap kali *user log in, user* menggunakan *password* berikutnya yang terdapat di daftar *password.*

Dengan *one-time password, user* direpotkan keharusan menjaga agar buku *passwordnya* jangan sampai dicuri.

2.3 Mekanisme Proteksi Sistem Komputer

Pada sistem komputer banyak objek yang perlu diproteksi, yaitu:

1. Objek perangkat keras
2. Objek perangkat lunak

Objek Perangkat Keras

Objek perangkat keras yang perlu diproteksi, antara lain:

- Pemroses
- Segmen memori
- *Disk Drive*
- *Printer*

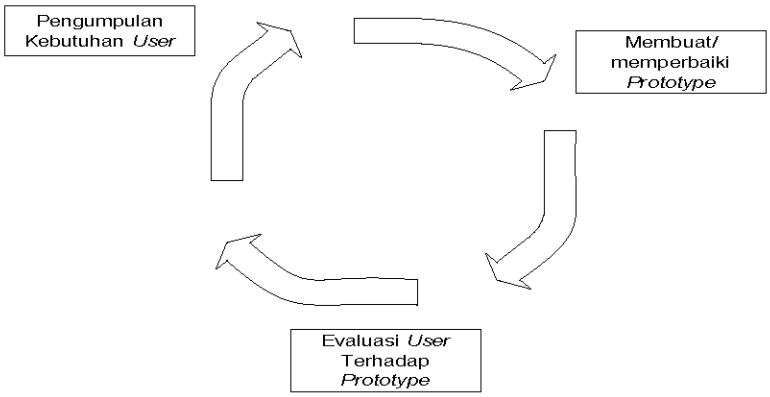
Objek Perangkat Lunak

Objek perangkat lunak yang perlu diproteksi, antara lain:

- Proses
- File
- Basisdata

2.4 Metode Rekayasa Perangkat Lunak dengan Model Prototipe (*Prototyping Model*)

Model ini bertujuan untuk membuat *prototype* dari perangkat lunak yang akan dikembangkan. Sering seorang pelanggan mendefinisikan serangkaian sasaran umum bagi perangkat lunak, tetapi tidak melakukan identifikasi kebutuhan *output*, pemrosesan, ataupun *input* secara detail. Pada kasus yang lain, pengembang mungkin tidak memiliki kepastian terhadap efisiensi algoritma, kemampuan penyesuaian diri dari sebuah sistem operasi atau bentuk-bentuk yang harus dilakukan oleh interaksi manusia dengan mesin. Dalam hal ini, serta pada banyak situasi yang lain, *prototyping paradigma* mungkin menawarkan pendekatan yang terbaik.



Gambar 2.1 Metode RPL dengan Model *Prototipe*

a. Pengumpulan kebutuhan *user*: Mencari informasi yang dibutuhkan untuk pengembangan *prototype*.
b. Membuat/Memperbaiki *prototype*: Mengembangkan *prototype* berdasarkan informasi yang telah didapatkan sebelumnya.
c. Evaluasi terhadap *prototype*: Melakukan pengujian terhadap *prototype*, bertujuan untuk mencari kekurangan yang ada pada *prototype* dan untuk kemudian kembali ke proses pengumpulan kebutuhan.
d. Proses ini terus berulang hingga mendapatkan hasil yang terbaik dari perangkat lunak. Intinya dalam *prototyping model*, yaitu terus menerus mengembangkan *prototype* hingga mendapatkan hasil yang paling baik.

2.5 Analisis Menggunakan *Metode Data Flow Oriented*

Untuk menganalisa sistem yang dibuat dilakukan pemodelan dengan menggunakan *Data Flow Oriented* dengan *tool Data Flow Diagram* (DFD). DFD adalah suatu teknik pemodelan secara grafis yang menggambarkan aliran data dalam sistem serta fungsi-fungsi (proses) yang terlibat dalam transformasi aliran data tersebut. Selain dapat memberikan informasi tambahan yang digunakan selama tahap analisis, DFD juga digunakan untuk merepresentasikan sistem atau perangkat lunak pada berbagai tingkatan abstraksi. Artinya, DFD dapat dibagi menjadi beberapa level yang menggambarkan penambahan aliran informasi dan fungsionalitas yang lebih rinci. DFD level 0 (*Data Context Diagram* (DCD))

merepresentasikan hubungan perangkat lunak/sistem dengan lingkungan yang berkaitan dengan sistem tersebut sebagai satu proses dengan data masukan dan keluaran digambarkan sebagai panah yang masuk dan keluar proses. Selanjutnya pada level yang lebih tinggi (1,2,..dst), proses tersebut dipecah-pecah untuk memperoleh aliran data dan proses yang lebih rinci.

Simbol-simbol yang digunakan dalam Diagram Alir Data (*Data Flow Diagram*) dapat dilihat pada daftar simbol.

2.5.1 CSPEC *(Control Specification)*

Digunakan untuk mengindikasikan bagaimana perlakuan *software* ketika suatu kejadian atau sinyal kontrol mulai terjadi dan proses apakah yang diaktifkan sebagai konsekuensi terjadinya suatu kejadian. CSPEC selalu berhubungan dengan *control bar*.

2.5.2 PSPEC *(Process Specification)*

Deskripsi tentang apa yang terjadi pada proses *level* paling bawah, pada suatu diagram aliran data. Maksud dari spesifikasi ini adalah untuk mendefinisikan apa yang harus dilakukan untuk mengubah aliran masuk *(Input)* menjadi keluaran *(Output)*.

2.5.3 Kamus Data

Kamus data adalah daftar terorganisir dari semua elemen data yang ada pada suatu sistem dengan definisi yang jelas/tepat, sehingga *user* dan analisis sistem bisa mendapat kesepahaman dari *input, output* dan komponen dari penyimpanan dan kalkulasi yang ada.

2.6 Perangkat Lunak (*Software*)

Perangkat lunak (*software*) dapat dikategorikan ke dalam 3 bagian, yaitu:

1. Perangkat lunak sistem operasi (*operating system*), yaitu program yang ditulis untuk mengendalikan dan mengkoordinasi kegiatan dari sistem komputer.

2. Perangkat lunak bahasa (*language software*), yaitu program yang digunakan untuk menerjemahkan instruksi-instruksi yang ditulis dalam bahasa pemrograman ke dalam bahasa mesin.
3. Perangkat lunak aplikasi (*application software*), yaitu program yang ditulis dan diterjemahkan oleh *language software* untuk menyelesaikan suatu aplikasi tertentu.

2.6.1 Software Sistem

Software sistem merupakan *background* program yang memungkinkan *software* aplikasi dapat berfungsi pada peralatan *hardware* sistem computer.

Software sistem dapat dibedakan menjadi 3 bagian, yaitu:

a. *Operating System*, merupakan kumpulan utama dari program yang mengatur aktivitas sistem computer.
b. *Language Translator*, berfungsi untuk mongkonversikan program aplikasi ke dalam bahasa mesin.
c. *Utility Programs*, suatu program yang menjelaskan atau memperluas kegunaan dari sistem operasi.

2.6.2 Software Aplikasi

Software aplikasi merupakan suatu perangkat yang memungkinkan pemakai memahami sistem komputer.

2.6.2.1 Borland Delphi

Delphi merupakan perangkat pengembangan aplikasi yang sudah terkenal dilingkungan windows. Dengan menggunakan perangkat lunak ini kita dapat membangun berbagai aplikasi yang berbasis windows dengan mudah. Dengan pendekatan visual, sehingga lebih memudahkan *programer* untuk membangun aplikasi yang berbasis windows. Delphi menggunakan bahasa objek Pascal sebagai bahasa dasar pemrograman.

2.6.2.2 Sistem Operasi *Microsoft Windows*

Windows merupakan sistem operasi untuk PC yang paling populer saat ini, mulai dari *Windows* 95, 98, NT dan yang baru saja diluncurkan yaitu *Windows* 2000, *Windows* Millenium dan *Windows* XP. Salah satu keunggulan *Windows* adalah kemudahan dalam penggunaannya. Misalnya kemudahan mulai dari install, konfigurasi sampai dengan adanya *feature plug and play* untuk hardware.

2.6.2.3 Hirarki Regristry Windows

Tentunya semua konfigurasi dan *setting* tersebut disimpan dalam sistem operasi, dan untuk menyimpan informasi berbagai *setting* dan konfigurasi, *Windows* menggunakan *registry*. *Registry* merupakan *database* yang digunakan untuk menyimpan semua *setting* dan informasi *hardware dan software*. Salah satu contohnya adalah misalnya seseorang mengganti assosiasi *file* atau meng*install program*, maka perubahan *setting* tersebut akan dituliskan pada *registry*. Contoh lainnya adalah menyembunyikan berbagai menu pada Menu *Start*.

Selain sebagai tempat untuk menyimpan informasi sistem operasi *Windows* sendiri, *registry* juga digunakan sebagai tempat untuk menyimpan berbagai informasi *setting* dan konfigurasi pada aplikasi atau program. Misalnya *WinZip* menggunakan *registry* untuk menyimpan informasi *toolbar*, aplikasi untuk membuka *file (viewer)*, *user name, serial number*, dan lain-lain.

Registry diletakkan pada dua buah *hidden file* yaitu *user.dat* dan *system.dat* yang terletak pada *directory Windows* untuk Win 95/98/Me dan pada *directory* Windows/System32/Config untuk *Windows* NT. Selain menggunakan *registry* (system.dat dan user.dat), *Windows* juga menyimpan informasi *setting* tertentu pada *file* msdos.sys, system.ini dan win.ini.

Registry terdiri dari beberapa bagian yang disebut *key* atau kunci. Terdapat 5 jenis *key* utama pada *registry*, yaitu :

1. ***HKEY_CLASSES_ROOT***

   Berisi semua tipe *file* beserta assosiasinya yang masing-masing tipe *file* tersebut akan mempunyai *subkey* sendiri-sendiri.

2. ***HKEY_CURRENT_USER***

   Berisi informasi tentang *user* yang sedang *log in* pada saat itu.

3. ***HKEY_LOCAL_MACHINE***

   Berisi informasi tentang *hardware* dan *setting software* yang berlaku untuk semua *user*.

4. ***HKEY_USERS***

   Berisi informasi tentang *desktop* dan *user setting* untuk tiap *user* yang berhak *log in* ke komputer tersebut. Tiap *user* mempunyai sebuah *subkey*. Jika hanya terdapat satu *user* maka nama *subkey* tersebut adalah ".default"

5. ***HKEY_CURRENT_CONFIG***

   Berisi informasi tentang konfigurasi *hardware*, berhubungan dengan *HKEY_LOCAL_MACHINE*.

BAB III

ANALISIS

Penganalisaan sistem ini bertujuan untuk mengetahui sistem keamanan sistem operasi *microsoft windows,* sebelum aplikasi *security* untuk *microsoft windows* diterapkan.

Tahap analisis sistem dalam rangka pembangunan perangkat lunak ini, menggunakan metodologi *prototyping model.* Metodologi *prototyping model* ini dimulai dengan *requirements gathering*. *Developers* dan konsumen/*client* bertemu dan menentukan tujuan *software* secara umum, mengidentifikasi kebutuhan-kebutuhan sistem yang telah diketahui sehingga didapatkan suatu *"quick design"*. *Quick design* akan digunakan untuk menyusun suatu *prototype*. Kemudian *prototype* ini akan dianalisis oleh konsumen/*client* dan akan digunakan untuk menentukan kembali kebutuhan-kebutuhan baru dari *software* yang akan dibuat. *Prototype* disediakan sebagai suatu mekanisme untuk mengidentifikasi kebutuhan *software* yang diinginkan oleh konsumen kemudian menjadi kesepakatan dalam kontrak. Analisis sistem dalam rangka pembangunan perangkat lunak ini, menggunakan metodologi *prototyping model,* sehingga analisis sistem akan dilakukan minimal 2 kali tahapan analisis.

3.1 Identifikasi Masalah

Masalah pengubahan konfigurasi *windows,* pencurian, kehilangan, dan modifikasi data oleh orang yang tidak berwewenang sangat sering terjadi pada komputer "umum". Maksudnya banyak orang lain yang turut memakai. Untuk itu, masalah-masalah tersebut perlu diidentifikasi agar dapat diketahui penyebabnya, sehingga masalah tersebut dapat dihindari.

Permasalahan yang ada dalam penelitian ini adalah:

1. Bagaimana cara menyembunyikan dan mengnonaktifkan beberapa fungsi *windows,* sehingga tidak dapat diakses oleh orang yang tidak berwewenang.
2. Bagaimana cara menyembunyikan *Drives* yang digunakan sebagai tempat penyimpanan data penting.

3. Bagaimana cara menghindari orang lain untuk mengakses program aplikasi yang berfungsi untuk mengolah data penting.

Berikut merupakan fungsi windows yang disembunyikan dan dinonaktifkan beserta alasannya:

1. *Task Manager*

   Adapun tujuan pemilihan fungsi windows ini adalah untuk menghindari orang yang tidak berwewenang untuk mematikan aplikasi-aplikasi yang beroperasi secara tersembunyi untuk mengawasi dan mengendalikan fungsi sistem operasi *windows*. Contohnya, aplikasi *driver,* aplikasi *system monitoring*, aplikasi anti virus, aplikasi *security,* dan lain sebagainya.

2. *Taskbar Properties*

   Tujuan utama pemilihan fungsi windows ini adalah untuk menghindari orang yang tidak berwewenang untuk menghapus *recently opened documents* untuk menghilangkan jejak akses *files*. Dan menghilangkan beberapa *short cut* penting pada *start menu,* seperti, *Control Panel, Search, My computer, My Network Places,* dan *Run command.*

4. *Registry Editor*

   Dengan mengubah beberapa nilai *registry windows* melalui *registry editor,* oleh orang yang tidak mengerti tentang *registry windows,* dapat berakibat fatal pada sistem operasi *windows*. Untuk menghindari terjadinya *fatal error* pada sistem operasi *windows*. Maka *registry editor* sangat perlu dinonaktifkan.

5. *Run command*

   Untuk menghindari orang yang tidak berwewenang untuk mengakses *run command* yang dapat memudahkan orang tersebut untuk mengakses beberapa fungsi *windows* dengan cepat. Contohnya, *msconfig, command prompt, regedit.*

6. *Search*

   Seorang penyusup dapat dengan mudah mencari data penting berupa *file* untuk dibuka, dihapus, maupun dimodifikasi, jika adanya fungsi *search,* jadi untuk menghindari terjadinya hal tersebut, maka fungsi *search* merupakan salah satu fungsi *windows* yang dapat menjadi pilihan untuk dinonaktifkan.

7. *Remove recent documents from start menu*

   Untuk dapat mengetahui *history user* mengakses *files*. Dengan cara menampilkan kembali *recent documents* pada *start menu* yang telah disembunyikan.

8. *Start menu log off option*

   Untuk menghindari terjadinya penggantian *user* lain untuk *log in* pada saat *user* sedang *log in*.

9. *Display setting*

   Untuk menghindari orang lain untuk mengubah tampilan layar *windows*, mengunci layar windows dengan *screen saver password protection*, mengubah *screen resolution* yang terlalu tinggi, yang dapat mempercepat rusaknya *monitor*.

10. CD-ROM *autorun function*

    Untuk menghindari terjadinya *program not responding*, pada saat CD-ROM mengalami kesulitan membaca CD *storage* (cacat) untuk menjalankan program autorun yang banyak memakan kapasitas *memory*.

3.2 Deskripsi Pemakai

Aplikasi *security* untuk *microsoft windows* ini, hanya dapat diakses oleh *administrator*, *administrator* yaitu orang yang memiliki wewenang untuk mengoperasikan aplikasi *security* untuk *microsoft windows* yang berfungsi untuk membatasi hak akses *user*, dalam hal pemakaian sistem operasi *windows* yang terinstalasi di komputer tersebut. Sedangkan *user* yaitu orang yang dibatasi hak aksesnya, dalam hal pemakaian sistem operasi *windows* dan tidak berhak untuk mengakses aplikasi tersebut.

3.3 Deskripsi Sistem

System security untuk *microsoft windows* merupakan aplikasi yang berfungsi sebagai media untuk memudahkan *administrator* dalam rangka membatasi hak akses *user*, dalam hal pemakaian sistem operasi *windows*. *Security* untuk *microsoft windows* ini berfungsi untuk meningkatkan keamanan data dan

mencegah terjadinya perubahan konfigurasi sistem operasi *microsoft windows* dari orang yang tidak diotorisasi.

3.4 Data Flow Diagram

Pemodelan *system security* untuk *microsoft windows* disini menggunakan metode *Data Flow Oriented* dengan tool *Data Flow Diagram* (DFD). DFD merupakan suatu teknik pemodelan menggunakan notasi-notasi grafis yang menunjukan aliran informasi dan perubahannya yang diterapkan sebagai perubahan data dari *input* dan diproses menjadi *output.*Berdasarkan model *prototyping* yang digunakan untuk menganalisa sistem yang akan dibangun maka pembuatan DFD akan dilakukan dua kali. Pembuatan DFD pertama dilakukan oleh pihak pengembang sistem sebagai awal dari pembangunan sistem yang akan dikonsultasikan dengan pihak pemakai sistem.

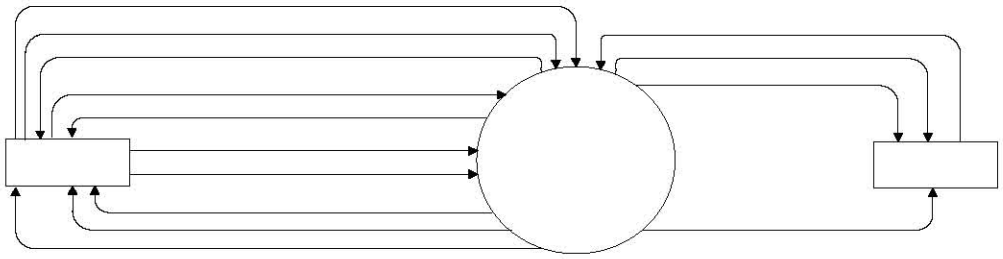
Pembuatan DFD kedua dilakukan setelah mendapatkan masukan atau penambahan sebuah proses baru dari pemakai sistem.

3.4.1 Analisis tahap pertama sebelum konsultasi

Analisis sistem yang dilakukan oleh pengembang sebelum dilakukan konsultasi kepada pemakai sistem analisis ini didasarkan pada kebutuhan sistem.

3.4.1.1 DFD Level 0 Tahap Pertama Analisis

DFD level 0 yang digunakan pada tahap pertama analisis ialah:

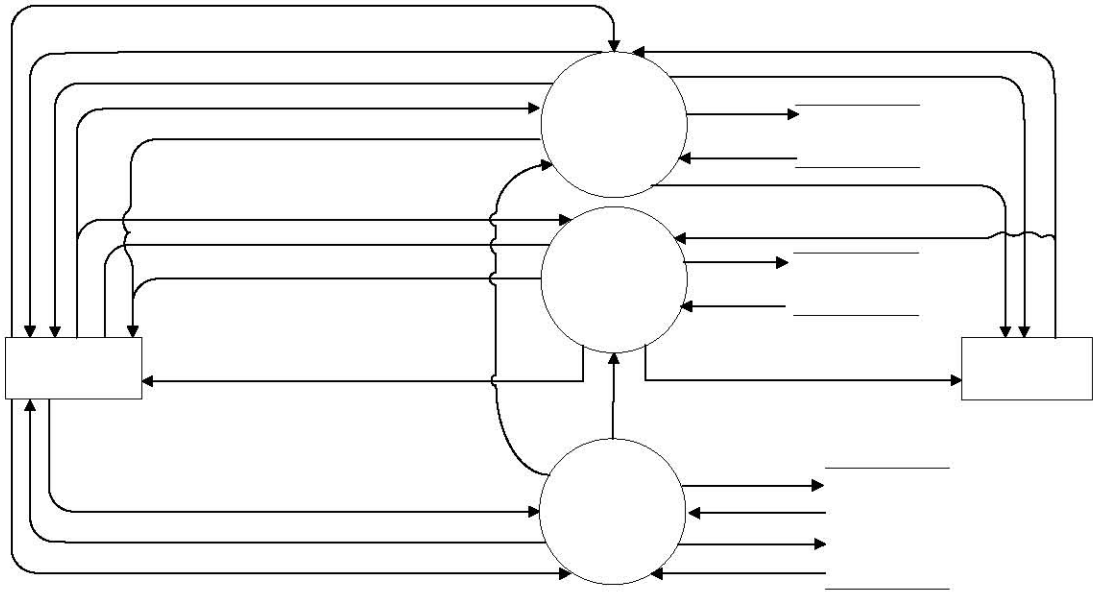


Gambar 3.1 DFD Level 0 Tahap Pertama Analisis

DFD level 0 tahap pertama analisis ini mempresentasikan hubungan sistem aplikasi *security windows* dengan *administrasi dan user* sebagai satu proses dengan data masukan dan keluaran yang digambarkan sebagai panah *input* dan *output.*

3.4.1.2 DFD Level 1 Tahap Pertama Analisis

DFD level 1 yang digunakan pada tahap pertama analisis adalah:

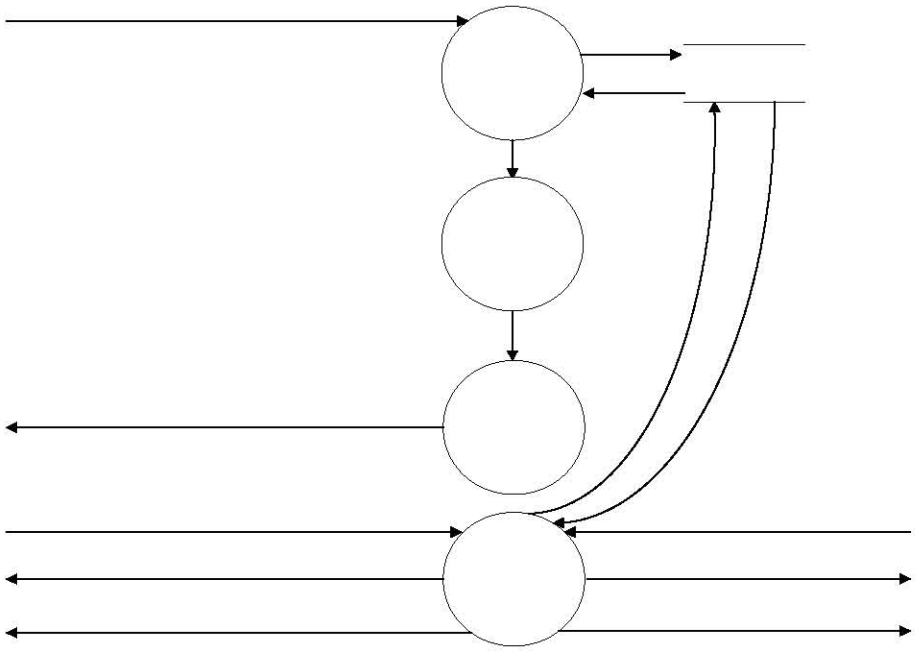


Gambar 3.2 DFD Level 1 Tahap Pertama Analisis: APLIKASI *SECURITY WINDOWS*

DFD level 1 tahap pertama analisis ini merupakan turunan dari DFD level 0 tahap pertama analisis, yang tediri dari 3 proses yaitu proses *disable / enable* fungsi *windows*, *hidden / show drives,* dan validasi. Dan setiap proses memiliki data *input/output* yang ada pada *data store.* Adapun *data store* tersebut yaitu fungsi *windows*, *drives*, dan *password* yang merupakan tempat penyimpanan data.

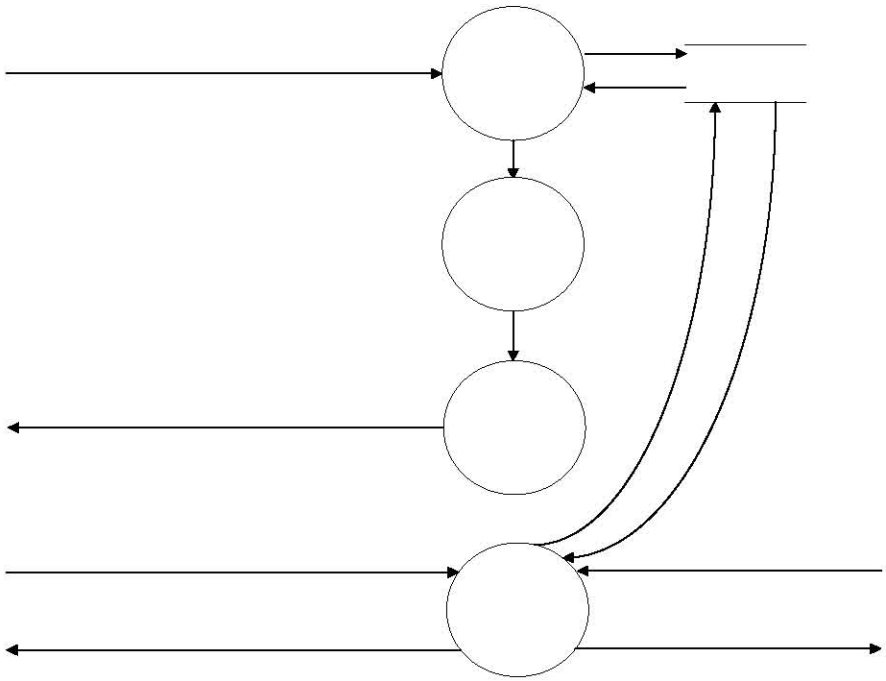
3.4.1.3 DFD Level 2 Tahap Pertama Analisis

DFD level 2 yang digunakan pada tahap pertama analisis adalah:



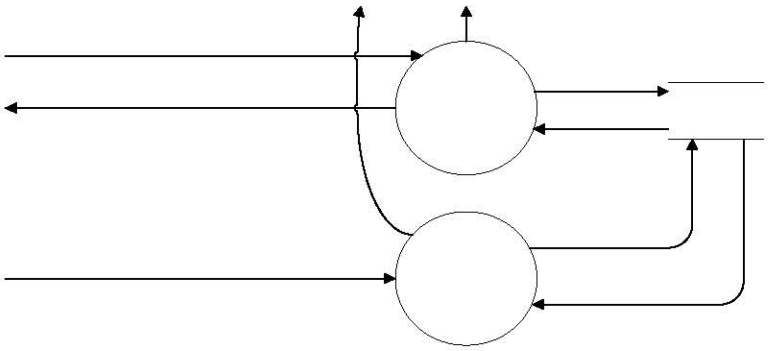
Gambar 3.3 DFD Level 2 Proses 1 Tahap Pertama Analisis: Disable/Enable Fungsi Windows

DFD level 2 proses 1 tahap pertama analisis ini merupakan turunan dari DFD level 1 proses 1 tahap pertama analisis, yang tediri dari 4 proses yaitu proses pilih fungsi *windows*, ubah nilai *registry windows,* proses *restart,* dan akses fungsi *windows.*



Gambar 3.4 DFD Level 2 Proses 2 Tahap Pertama Analisis: Hidden/Show Drives

DFD level 2 proses 2 tahap pertama analisis ini merupakan turunan dari DFD level 1 proses 2 tahap pertama analisis, yang tediri dari 4 proses yaitu proses pilih nama *drive*, ubah nilai *registry windows,* proses *restart,* dan akses sistem operasi.



Gambar 3.5 DFD Level 2 Proses 3 Tahap Pertama Analisis : Validasi

DFD level 2 proses 3 tahap pertama analisis ini merupakan turunan dari DFD level 1 proses 3 tahap pertama analisis, yang tediri dari 2 proses yaitu proses validasi *password*, dan *clear password.*

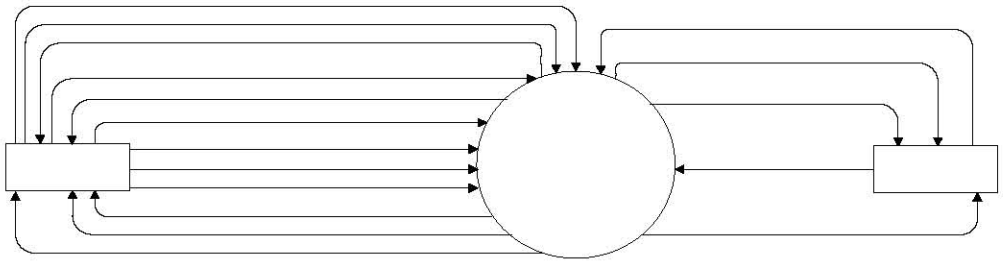
3.4.2 Analisis tahap kedua setelah konsultasi

Analisis sistem yang dilakukan setelah konsultasi dengan pemakai sistem, sehingga mendapat masukan atau tambahan proses yang diinginkan oleh pemakai sistem.

Dengan penambahan proses dari pemakai sistem akan mempengaruhi analisis sistem yang sudah dilakukan sebelumnya. Pada kesempatan ini mencoba untuk menangani penambahan proses dengan memasukan proses tersebut ke DFD tahap pertama, sehingga DFD tahap pertama dikembangkan lebih lanjut. Pengembangan DFD tahap pertama hasil konsultasi dengan pemakai sistem akan dibuat pada DFD tahap kedua. Pada dasarnya DFD tahap kedua merupakan DFD tahap pertama yang mendapatkan masukan proses baru, adapun proses baru tersebut yaitu proses nonaktif / aktif *file**.exe.

3.4.2.1 DFD Level 0 Tahap Kedua Analisis

DFD level 0 yang digunakan pada tahap kedua analisis ialah:

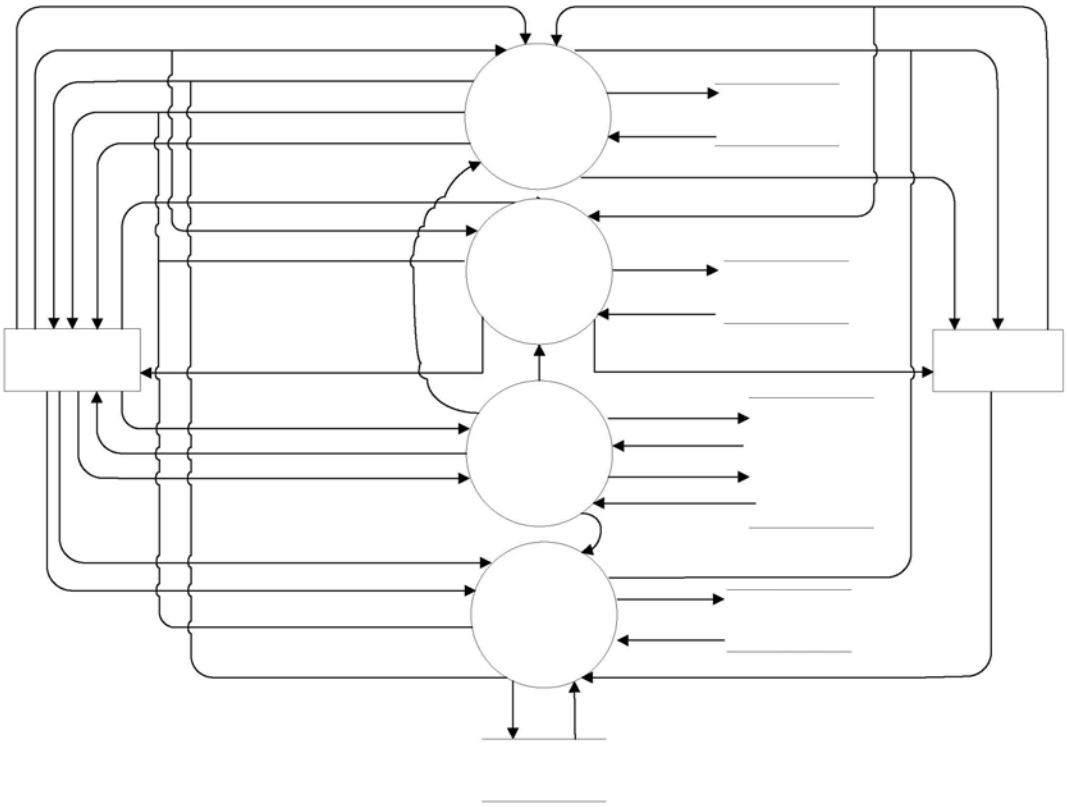


Gambar 3.6 DFD Level 0 Tahap Kedua Analisis

DFD level 0 tahap kedua analisis ini mempresentasikan hubungan sistem aplikasi *security windows* ini dengan *administrasi dan user* sebagai satu proses dengan data masukan dan keluaran digambarkan sebagai panah *input* dan *output*.

3.4.2.2 DFD Level 1 Tahap Kedua Analisis

DFD level 1 yang digunakan pada tahap kedua analisis adalah:

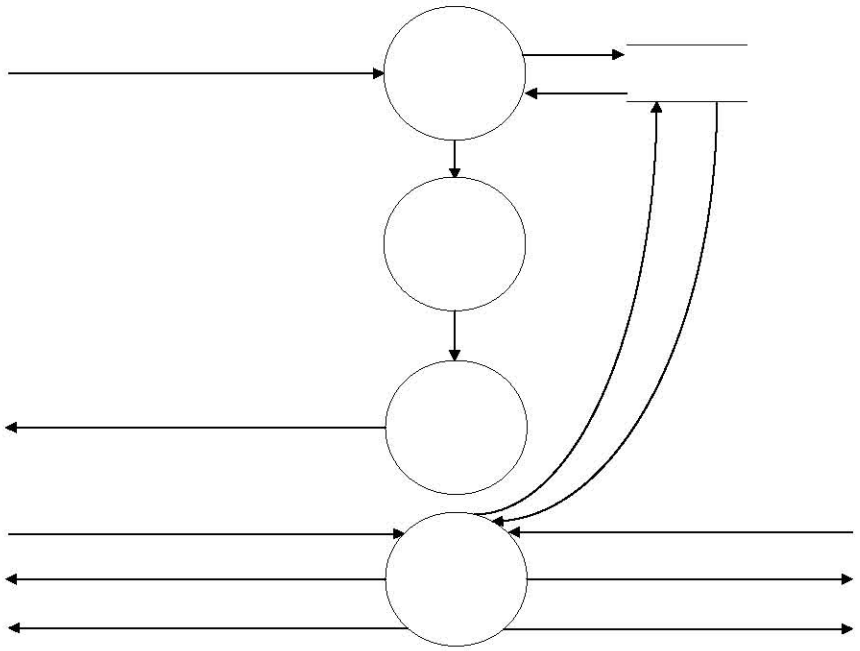


Gambar 3.7 DFD Level 1 Tahap Kedua Analisis: APLIKASI *SECURITY WINDOWS*

DFD level 1 tahap kedua analisis ini merupakan turunan dari DFD level 0 tahap kedua analisis, yang tediri dari 4 proses yaitu proses *disable / enable* fungsi *windows*, *hidden / show drives,* validasi dan nonaktif / aktif *file**.exe. Dan proses 1 sampai proses 3 memiliki data *input/output* yang ada pada *data store.* Adapun *data store* tersebut yaitu fungsi *windows*, *drives*, dan *password,* yang merupakan tempat penyimpanan data, kecuali proses 4 terdiri dari 2 *data store* yaitu *data store registry.*

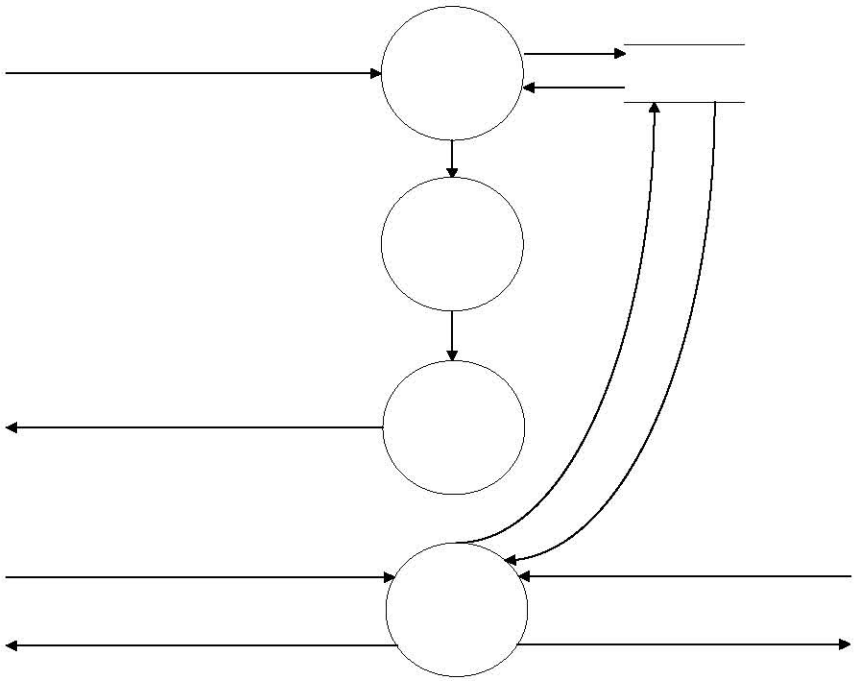
3.4.2.3 DFD Level 2 Tahap Kedua Analisis

DFD level 2 yang digunakan pada tahap kedua analisis adalah:



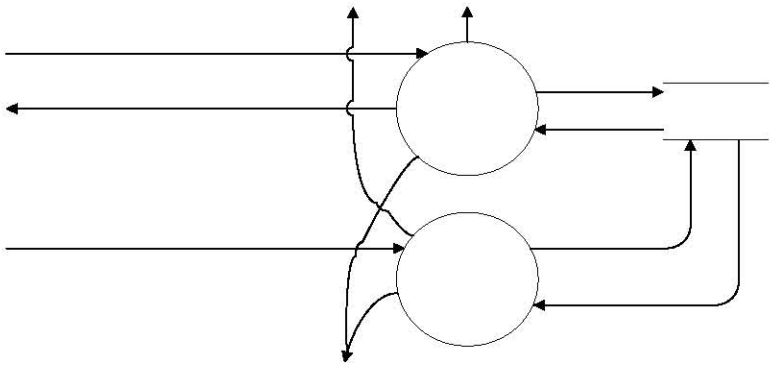
Gambar 3.8 DFD Level 2 Proses 1 Tahap Kedua Analisis: Disable/Enable Fungsi Windows

DFD level 2 proses 1 tahap kedua analisis ini merupakan turunan dari DFD level 1 proses 1 tahap kedua analisis, yang tediri dari 4 proses yaitu proses pilih fungsi *windows*, ubah nilai *registry windows,* proses *restart,* dan akses fungsi *windows.*



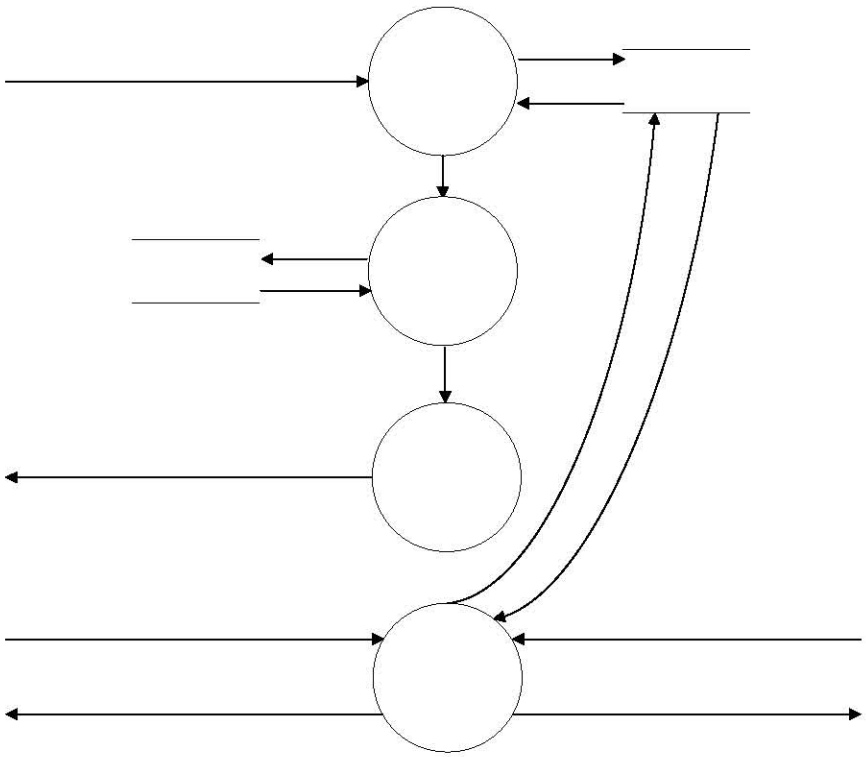
Gambar 3.9 DFD Level 2 Proses 2 Tahap Kedua Analisis: Hidden/Show Drives

DFD level 2 proses 2 tahap kedua analisis ini merupakan turunan dari DFD level 1 proses 2 tahap kedua analisis, yang tediri dari 4 proses yaitu proses pilih nama *drive*, ubah nilai *registry windows,* proses *restart,* dan akses sistem operasi.



Gambar 3.10 DFD Level 2 Proses 3 Tahap Kedua Analisis: Validasi

DFD level 2 proses 3 tahap kedua analisis ini merupakan turunan dari DFD level 1 proses 3 tahap kedua analisis, yang tediri dari 2 proses yaitu proses validasi *password*, dan *clear password.*



Gambar 3.11 DFD Level 2 Proses 4 Tahap Kedua Analisis: Nonaktif/Aktif File*.exe

DFD level 2 proses 4 tahap kedua analisis ini merupakan turunan dari DFD level 1 proses 4 tahap kedua analisis dan merupakan pengembangan dari DFD tahap pertama analisis, yang tediri dari 4 proses yaitu proses pilih nama *file**.exe, ubah nilai *registry windows,* proses *restart,* dan akses *file*.exe.*

3.5 PSPEC

3.5.1 PSPEC Tahap Pertama Analisis

Spesifikasi proses dilihat dari DFD tahap pertama analisis, adalah:

1. Proses 1.1 Pilih Fungsi *Windows*

Narasi :
1. Pilih fungsi *windows* yang telah disediakan oleh *store* fungsi *windows* untuk dinonaktifkan/diaktifkan kembali.
2. Proses menerima fungsi *windows* mana yang akan dinonaktifkan/diaktifkan kembali dari *store* fungsi *windows.*
3. Proses akan memberikan laporan hanya berupa fungsi *windows* mana yang akan diproses ke proses berikutnya.

2. Proses 1.2 Ubah Nilai Registry Windows

Narasi :
1. Proses menerima laporan berupa fungsi *windows* mana yang akan dinonaktifkan/diaktifkan.
2. Proses akan merubah nilai *registry windows* yang berhubungan dengan fungsi *windows* mana yang akan dinonaktifkan/diaktifkan kembali.
3. Perubahan nilai *registry* akan dikirim ke proses berikutnya.

3. Proses 1.3 Proses Restart

Narasi :
1. Proses menerima perubahan nilai *registry windows.*
2. Dengan adanya perubahan nilai *registry windows,* proses akan mengkonfirmasikan pesan *restart.*

4. Proses 1.4 Akses Fungsi Windows

Narasi :
1. Proses memberikan *output* berupa tampilan fungsi windows dalam kondisi nonaktif/aktif kembali.
2. Jika *administrator / user* mengakses fungsi *windows* yang telah dinonaktifkan, maka proses menampilkan pesan kesalahan.

5. Proses 2.1 Pilih Nama Drive

Narasi :
1. Pilih nama *drive* yang telah disediakan oleh *store drives* untuk dinonaktifkan/diaktifkan kembali.
2. Proses menerima nama *drive* mana yang akan dinonaktifkan/diaktifkan kembali dari *store drives.*
3. Proses akan memberikan laporan hanya berupa nama *drive* mana yang akan diproses ke proses berikutnya.

6. Proses 2.2 Ubah Nilai Registry *Windows*

Narasi :
1. Proses menerima laporan berupa nama *drive* mana yang akan dinonaktifkan/diaktifkan.
2. Proses akan merubah nilai *registry windows* yang berhubungan dengan nama *drive* mana yang akan dinonaktifkan/diaktifkan kembali.
3. Perubahan nilai *registry* akan dikirim ke proses berikutnya.

7. Proses 2.3 Proses Restart

Narasi :
1. Proses menerima perubahan nilai *registry windows.*
2. Dengan adanya perubahan nilai *registry windows,* proses akan mengkonfirmasikan pesan *restart.*

8. Proses 2.4 Akses Sistem Operasi

Narasi :
1. Sewaktu *Administrator / User* mengakses *windows,* proses memberikan *output* hanya berupa tampilan *drive* yang aktif saja sedangkan *drive* yang dinonaktifkan, tidak ditampilkan.

9. Proses 3.1 Validasi Password

Narasi :
1. Proses menerima *password* dari *administrator,* dan membandingkannya dengan *store password.*
2. Jika *password* yang diterima tidak *valid,* maka pesan_invalid_pw ditampilkan.

3. Jika *password* yang diterima *valid,* maka dilanjutkan ke proses-proses yang lain.

10. Proses 3.2 Clear Password

Narasi : 1. Proses mengosongkan data *store password.*

2. Proses mengijinkan *administrator* mengakses proses yang lain tanpa validasi *password* terdahulu.

3.5.2 PSPEC Tahap Kedua Analisis

Spesifikasi proses dilihat dari DFD tahap kedua analisis, adalah:

1. Proses 1.1 Pilih Fungsi *Windows*

Narasi : 1. Pilih fungsi *windows* yang telah disediakan oleh data *store* fungsi *windows* untuk dinonaktifkan/diaktifkan kembali.

2. Proses menerima fungsi *windows* mana yang akan dinonaktifkan/diaktifkan kembali dari *store* fungsi *windows.*

3. Proses akan memberikan laporan hanya berupa fungsi *windows* mana yang akan diproses ke proses berikutnya.

2. Proses 1.2 Ubah Nilai Registry *Windows*

Narasi : 1. Proses menerima laporan berupa fungsi *windows* mana yang akan dinonaktifkan/diaktifkan.

2. Proses akan merubah nilai *registry windows* yang berhubungan dengan fungsi *windows* mana yang akan dinonaktifkan/diaktifkan kembali.

3. Perubahan nilai *registry* akan dikirim ke proses berikutnya.

3. Proses 1.3 Proses Restart

Narasi : 1. Proses menerima perubahan nilai *registry windows.*

2. Dengan adanya perubahan nilai *registry windows,* proses akan mengkonfirmasikan pesan *restart.*

4. Proses 1.4 Akses Fungsi Windows

Narasi : 1. Proses memberikan *output* berupa tampilan fungsi windows dalam kondisi nonaktif/aktif kembali.
2. Jika *administrator / user* mengakses fungsi *windows* yang telah dinonaktifkan, maka proses menampilkan pesan kesalahan.

5. Proses 2.1 Pilih Nama Drive

Narasi : 1. Pilih nama *drive* yang telah disediakan oleh *store drive*s untuk dinonaktifkan/diaktifkan kembali.
2. Proses menerima nama *drive* mana yang akan dinonaktifkan/diaktifkan kembali dari *store drive*s.
3. Proses akan memberikan laporan hanya berupa nama *drive* mana yang akan diproses ke proses berikutnya.

6. Proses 2.2 Ubah Nilai Registry *Windows*

Narasi : 1. Proses menerima laporan berupa nama *drive* mana yang akan dinonaktifkan/diaktifkan.
2. Proses akan merubah nilai *registry windows* yang berhubungan dengan nama *drive* mana yang akan dinonaktifkan/diaktifkan kembali.
3. Perubahan nilai *registry* akan dikirim ke proses berikutnya.

7. Proses 2.3 Proses Restart

Narasi : 1. Proses menerima perubahan nilai *registry windows.*
2. Dengan adanya perubahan nilai *registry windows,* proses akan mengkonfirmasikan pesan *restart.*

8. Proses 2.4 Akses Sistem Operasi

Narasi : 1. Sewaktu *Administrator / User* mengakses *windows,* proses memberikan *output* hanya berupa tampilan *drive* yang aktif saja sedangkan *drive* yang dinonaktifkan, tidak ditampilkan.

9. Proses 3.1 Validasi Password

Narasi :
1. Proses menerima *password* dari *administrator,* dan membandingkannya dengan *store password.*
2. Jika *password* yang diterima tidak *valid,* maka pesan_invalid_pw ditampilkan.
3. Jika *password* yang diterima *valid,* maka dilanjutkan ke proses-proses yang lain.

10. Proses 3.2 Clear Password

Narasi :
1. Proses mengosongkan data *store password.*
2. Proses mengijinkan *administrator* mengakses proses yang lain tanpa validasi *password* terdahulu.

11. Proses 4.1 Pilih Nama File*.exe

Narasi :
1. Pilih nama file*.exe yang telah disediakan oleh data *store* files*.exe untuk dinonaktifkan/diaktifkan kembali.
2. Proses menerima nama file*.exe mana yang akan dinonaktifkan/diaktifkan kembali dari *store* files*.exe.
3. Proses akan memberikan laporan berupa hanya nama file*.exe mana yang akan diproses ke proses berikutnya.

12. Proses 4.2 Ubah Nilai Registry *Windows*

Narasi :
1. Proses menerima laporan berupa nama file*.exe mana yang akan dinonaktifkan/diaktifkan.
2. Proses akan merubah nilai *registry windows* yang berhubungan dengan nama file*.exe mana yang akan dinonaktifkan/diaktifkan kembali.
3. Perubahan nilai *registry* akan dikirim ke proses berikutnya.

13. Proses 4.3 Proses Restart

Narasi : 1. Proses menerima perubahan nilai *registry windows.*

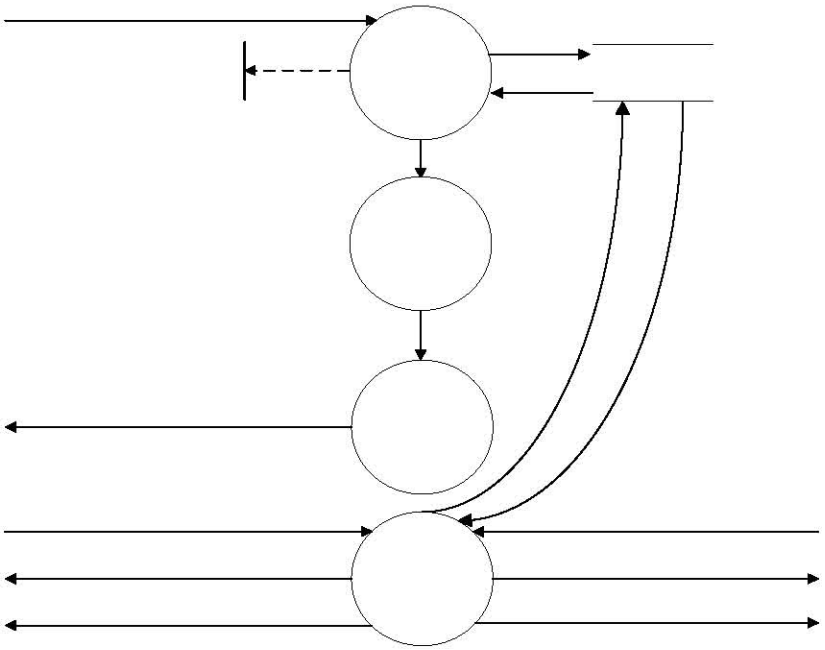
2. Dengan adanya perubahan nilai *registry windows,* proses akan mengkonfirmasikan pesan *restart.*

14. Proses 4.4 Akses File*.exe

Narasi : 1. Jika *administrator / user* mengakses file*.exe yang telah dinonaktifkan, maka proses menampilkan pesan kesalahan.

3.6 Control Flow Diagram

CFD yang digunakan berdasarkan DFD level 2, yaitu :

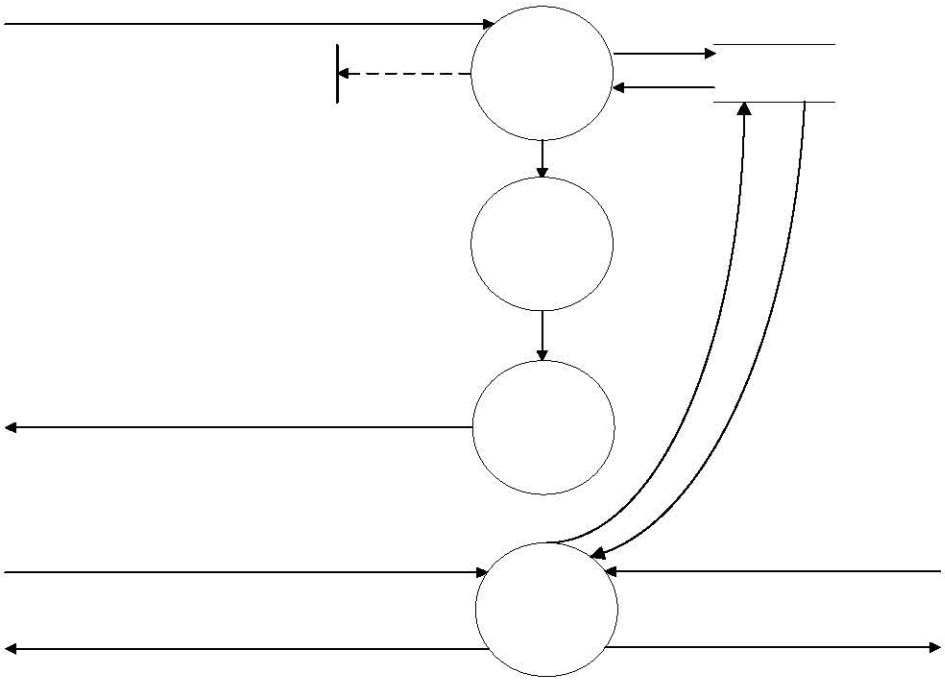


Gambar 3.12 CFD level 2 proses 1: Disable/Enable Fungsi Windows

Tabel dibawah ini menjelaskan *Process Activation Table* level 2 : Proses 1 (Disable/Enable Fungsi Windows)

Tabel 3.1 CSPEC Disable/Enable Fungsi Windows

Control	**Proses 1.1**	**Proses 1.2**	**Proses 1.3**	**Proses 1.4**
Command	DC	1	0	0

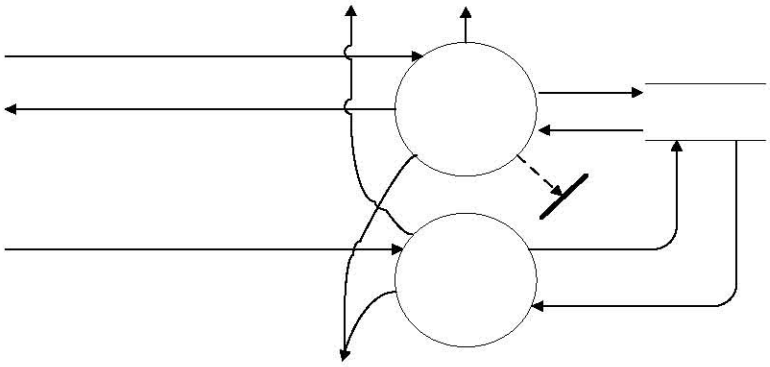


Gambar 3.13 CFD level 2 proses 2: Hidden/Show Drives

Tabel dibawah ini menjelaskan *Process Activation Table* level 2 : Proses 2 (Hidden/Show Drives)

Tabel 3.2 CSPEC Hidden/Show Drives

Control	**Proses 2.1**	**Proses 2.2**	**Proses 2.3**	**Proses 2.4**
Command1	DC	1	0	0

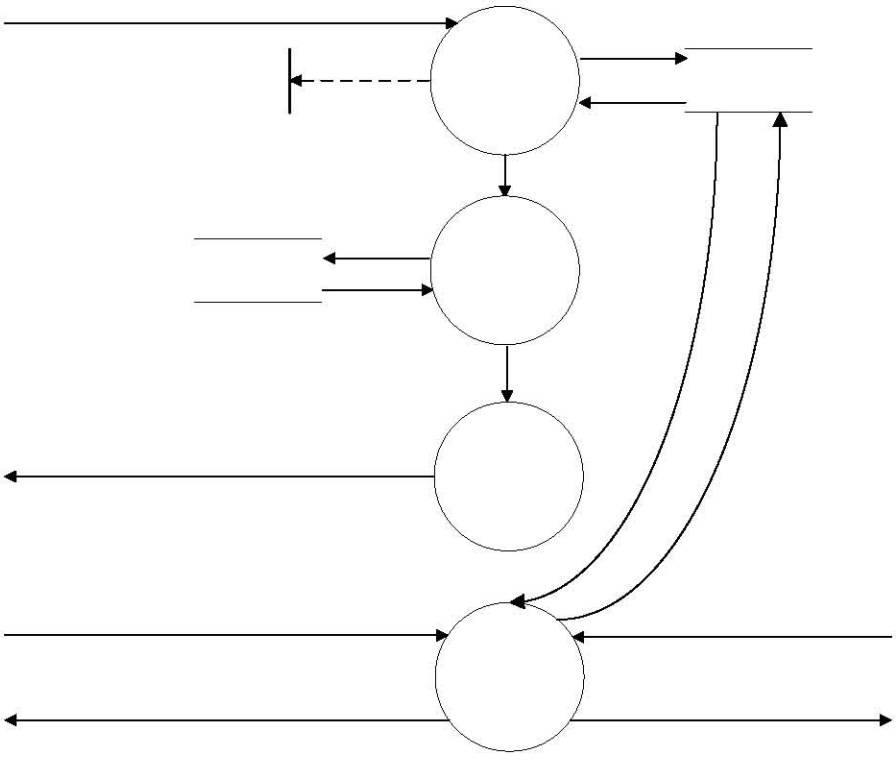


Gambar 3.14 CFD level 2 proses 3: Validasi

Tabel dibawah ini menjelaskan *Process Activation Table* level 2 : Proses 3 (Validasi)

Tabel 3.3 CSPEC Validasi

Control	**Proses 3.1**	**Proses 3.2**
Validasi	DC	1



Gambar 3.15 CFD level 2 proses 4: Nonaktif/Aktif File*.exe

Tabel dibawah ini menjelaskan *Process Activation Table* level 2 : Proses 4 (Nonaktif/Aktif File*.exe)

Tabel 3.4 CSPEC Nonaktif/Aktif File*.exe

Control	**Proses 4.1**	**Proses 4.2**	**Proses 4.3**	**Proses 4.4**
Command2	DC	1	0	0

3.7 Kamus Data

Kamus data yang digunakan pada DFD adalah:

Tabel 3.5 Kamus Data

No.	Aliran Data	Type Data	Keterangan
1.	Fungsi_Windows	Boolean	*Pilih fungsi windows*
2.	Akses_W	Boolean	*Akses fungsi windows*
3.	Tampilan_Fungsi_Windows	Boolean	*Fungsi windows dalam kondisi disable/enable*
4.	Nama_Drive	Boolean	*Pilih nama drive*
5.	Tampilan_Drive	Boolean	*Drives dalam kondisi tersembunyi/aktif*
6.	Nama_File*.exe	Boolean	*Pilih nama file*.exe*
7.	Akses_F	Boolean	*Akses file*.exe*
8.	Password	Char	*Data password*
9.	Blank	Char	‘‘
10.	Pesan_Restart	Text	“You must restart this computer for the change setting to take effect. Do you want to restart your computer now?”
11.	Pesan_Kesalahan	Text	[“$ has been disabled by your administrator”|“This operation has been cancelled due to

11.	Pesan_Kesalahan	Text	restrictions in effect on this computer, Please contact your system administrator."]
12.	Pesan_Invalid_PW	Text	"Wrong Password"
13.	Akses	Boolean	*Validasi mengakses proses [1|2|3|4]*
14.	Fungsix	Value	*Data store fungsi*
15.	InfoPassword	Char	*Data store password*
16.	Drivex	Value	*Data store drives*
17.	File*.exe	Value	*Data store file*.exe*
18.	Nilai_Reg	Value	*Data store registry*

BAB IV

PERANCANGAN

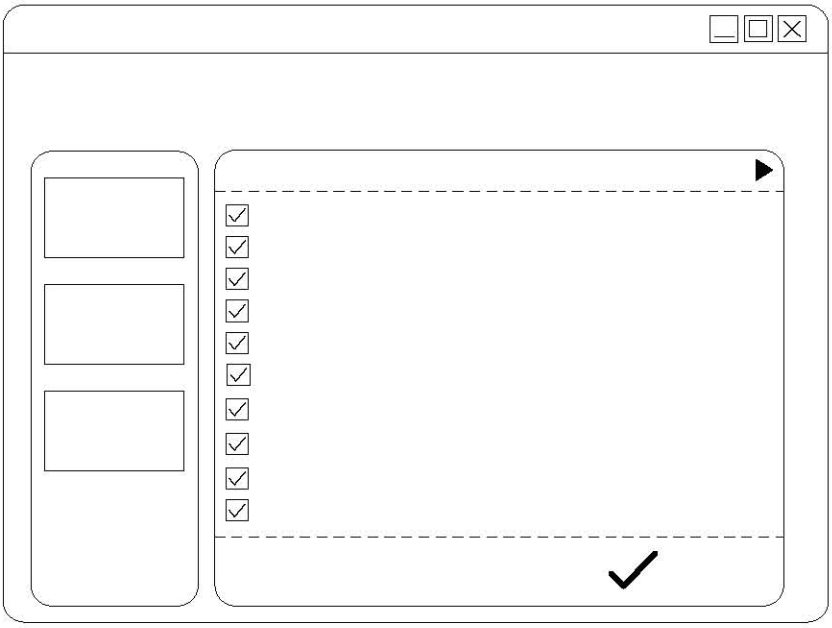
4.1 Perancangan Antarmuka

Perancangan antarmuka menjelaskan rutinitas program yang akan dijalankan oleh sebuah sistem komputerisasi untuk menjelaskan interaksi antara pemakai (*user*) dengan program yang dibuat.

Pada bab ini akan digambarkan rancangan antarmuka yang akan digunakan dalam sistem.

Form yang akan dirancang dalam perancangan antarmuka adalah:

1. Form Disable/Enable Windows Function

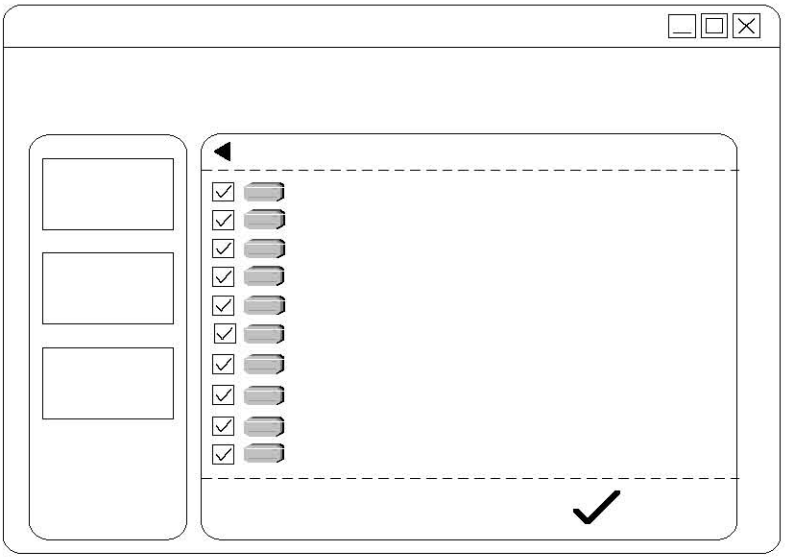


Gambar 4.1 Form Disable/Enable Windows Function

Form disable / enable windows function ini terdiri dari beberapa komponen yaitu *title* program yaitu "Security For Microsoft Windows". *Label* sebagai identitas *form* yang aktif. *Checkpoint* berfungsi sebagai pemilihan fungsi *windows* mana yang perlu di*disable* dari kumpulan fungsi *windows* yang tersedia didalam *checklist*. Tombol *apply* berfungsi sebagai persetujuan untuk pengambilan keputusan untuk penentuan pilihan fungsi

windows mana saja yang *disable / enable* dan mengkonfirmasikan kepada *administrator* untuk pengambilan keputusan untuk melakukan proses *restart. Icon tweak* dan *label disable / enable windows function* berfungsi untuk mengaktifkan *form disable / enable windows function. Icon password* berfungsi untuk mengaktifkan *form password. Icon about* berfungsi untuk mengaktifkan *form about.*

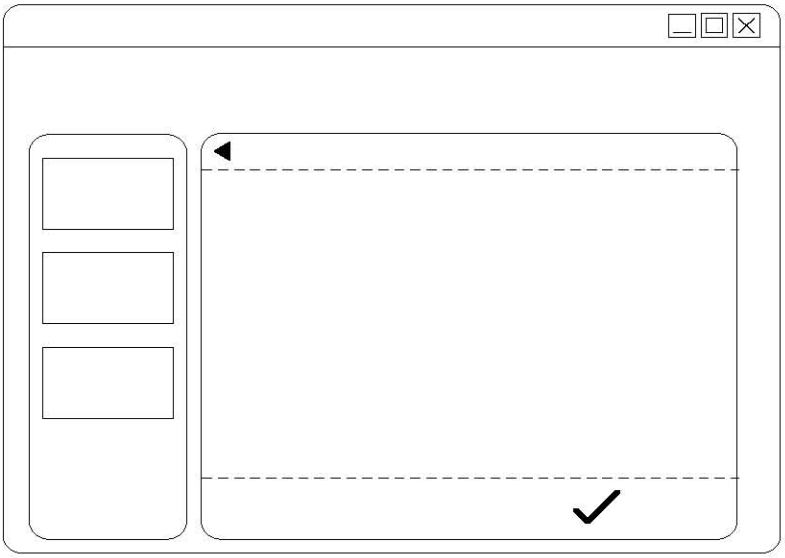
2. Form Hidden/Show Drives



Gambar 4.2 Form Hidden/Show Drives

Form hidden / show drives memiliki beberapa komponen yang sama dengan *form disable / enable windows function* yaitu *title program, label, icon tweak, icon password, icon about.* Sedangkan komponen yang berbeda adalah *label hidden / show drives* berfungsi untuk mengaktifkan *form hidden / show drives. Label restrict running* berfungsi untuk mengaktifkan *form restrict running. Checkpoint* berfungsi sebagai pemilihan *drive* mana yang perlu disembunyikan dari kumpulan *drive* yang tersedia didalam *checklist.* Tombol *apply* berfungsi sebagai persetujuan untuk pengambilan keputusan untuk penentuan pilihan *drive* mana saja yang perlu ditampilkan atau disembunyikan dan mengkonfirmasikan kepada *administrator* untuk pengambilan keputusan untuk melakukan proses *restart.*

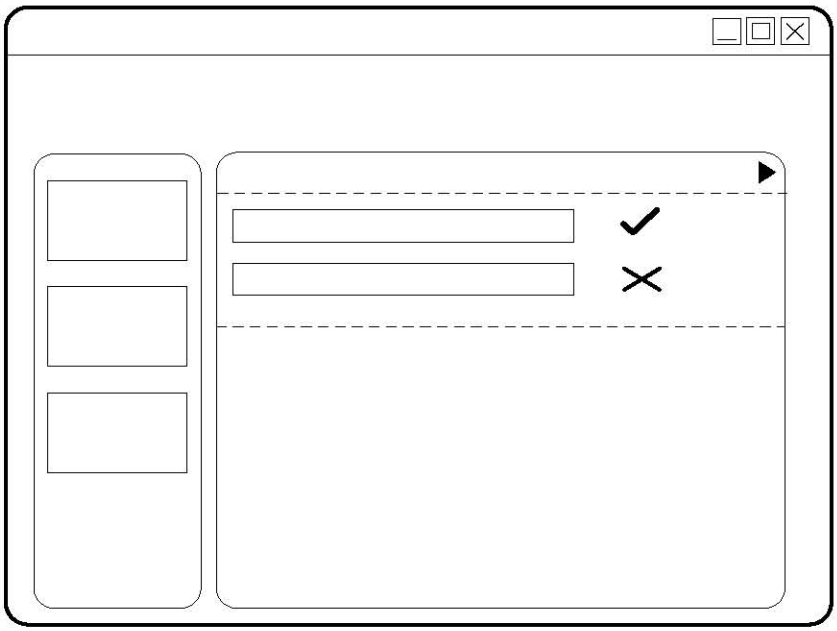
3. Form Restrict Running



Gambar 4.3 Form Restrict Running

Form restrict running memiliki beberapa komponen yang sama dengan *form disable / enable windows function* dan *form hidden / show drives* yaitu *title program, label, icon tweak, icon password, icon about, label hidden / show, label restrict running*. Sedangkan komponen yang berbeda adalah nama *file**.exe berfungsi sebagai daftar kumpulan nama *file**.exe yang telah dinonaktifkan. Tombol *enable* berfungsi untuk mengaktifkan kembali *file**.exe yang telah dinonaktifkan. Tombol *disable* berfungsi untuk mengnonaktifkan *file**.exe. Tombol *apply* berfungsi sebagai persetujuan untuk pengambilan keputusan untuk mengnonaktifkan atau mengaktifkan kembali *file**.exe yang tersedia dan mengkonfirmasikan kepada *administrator* untuk pengambilan keputusan untuk melakukan proses *restart*.

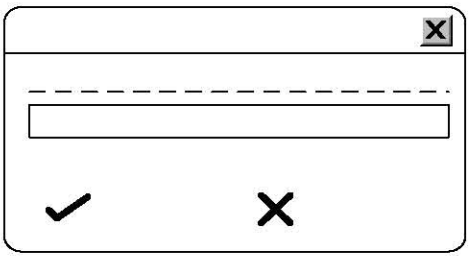
4. Form Password



Gambar 4.4 Form Password

Form password memiliki beberapa komponen yang sama dengan *form disable / enable windows function, form hidden / show drives, form restrict running* yaitu *title program, icon tweak, icon password, icon about.* Sedangkan komponen yang berbeda adalah *label password protection* berfungsi untuk mengaktifkan *form password. Text editor* berfungsi sebagai tempat memasukan karakter *password*, Tombol *set* berfungsi untuk menetapkan *password* yang dimasukan. Tombol *clear* berfungsi untuk mengosongkan *password* yang ada sehingga mengakses program tersebut tanpa validasi *password* terlebih dahulu.

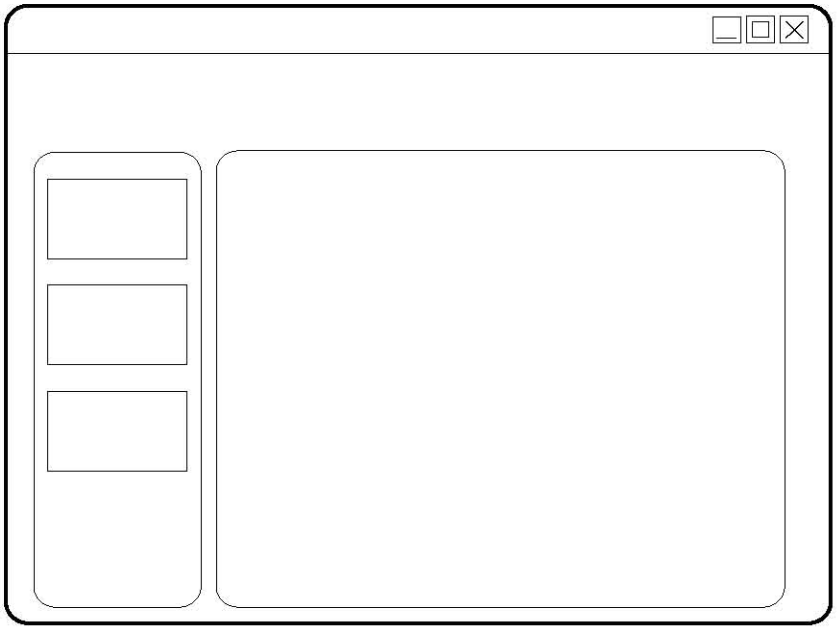
5. Form log in



Gambar 4.5 Form Log in

Form log in memiliki beberapa komponen yaitu *title form* yaitu "Password". T*ext editor* berfungsi untuk menginput *password.* Tombol ok berfungsi untuk validasi *password* yang diinput, jika *valid* maka diijinkan mengakses program "Security For Microsoft Windows" dan jika *invalid* maka muncul pesan "Wrong Password !!!", dan tombol *cancel* berfungsi untuk membatalkan proses validasi *password.*

6. Form About

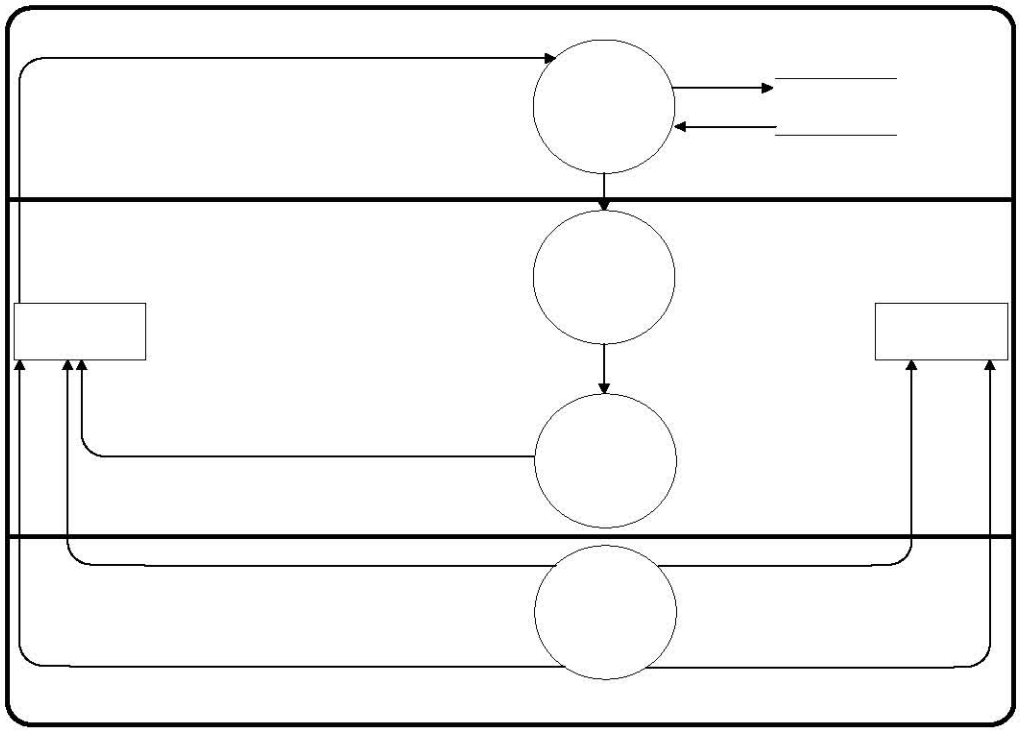


Gambar 4.6 Form About

Form about memiliki beberapa komponen yang sama dengan *form disable / enable windows function, form hidden / show drives, form restrict running* yaitu *title program, icon tweak, icon password, icon about.* Sedangkan komponen yang berbeda adalah *image & text* merupakan tempat keterangan dari *programmer* dan *icon* dari program tersebut.

4.2 Perancangan Arsitektural

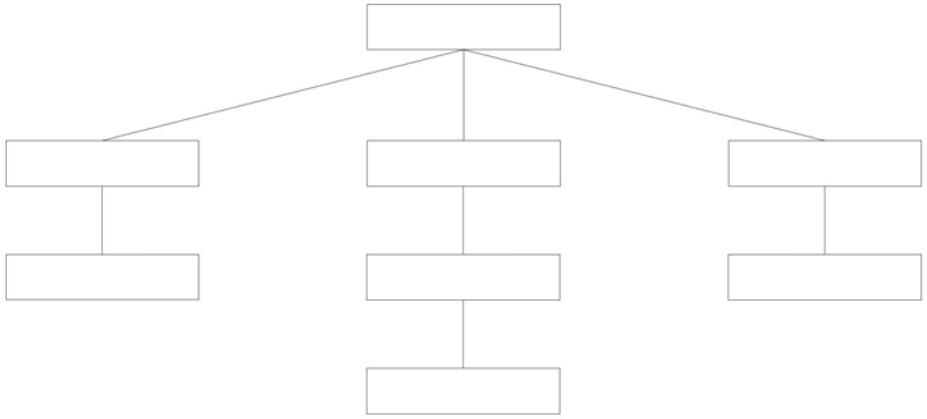
Perancangan pada *security* untuk *microsoft Windows* berdasarkan analisis menghasilkan pemetaan arsitektur (*Transform Mapping)* untuk *Data Flow Diagram Level 2. Transform mapping* yang dilakukan untuk mendapatkan struktur program adalah:



Gambar 4.7 *Transform Mapping* untuk DFD level 2 Proses 1: *Disable/Enable Fungsi Windows*

Transform mapping untuk DFD level 2 proses 1 ini terdiri atas proses pilih fungsi *windows* yang telah dipetakan menjadi modul *input,* proses ubah nilai *registry windows* dan proses *restart* yang telah dipetakan menjadi

modul proses, dan proses akses fungsi *windows* yang telah dipetakan menjadi modul *output.*

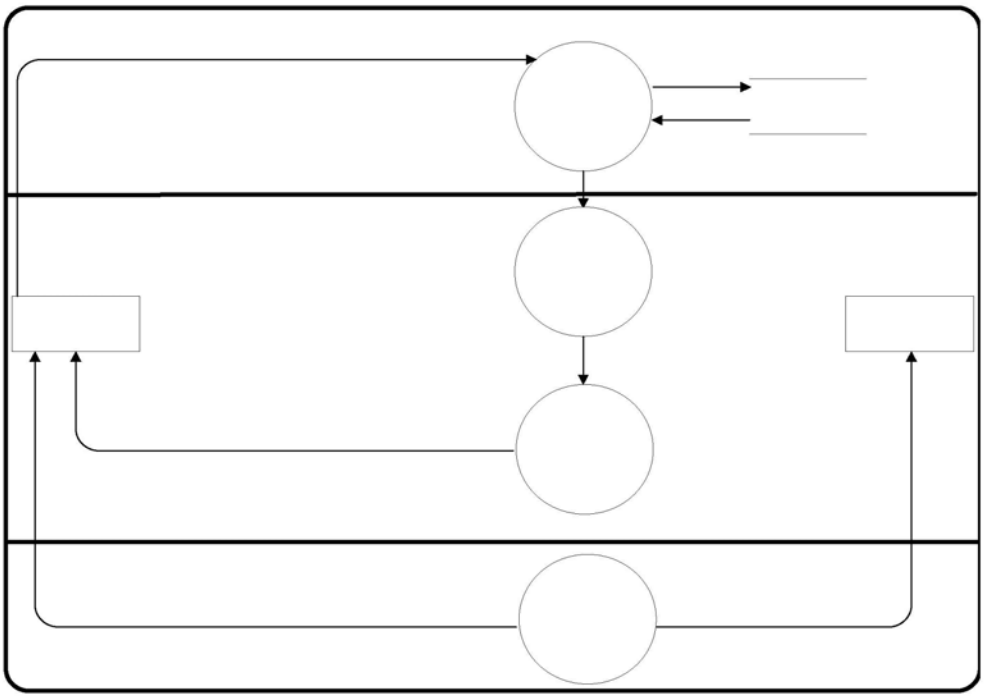


Gambar 4.8 Struktur Program untuk DFD Level 2 Proses 1: *Disable/Enable Fungsi Windows*

Struktur program untuk DFD level 2 proses 1 ini terdiri atas proses pilih fungsi *windows* yang telah dipetakan menjadi modul *input,* proses ubah nilai *registry windows* dan proses *restart* yang telah dipetakan menjadi modul proses, dan proses akses fungsi *windows* yang telah dipetakan menjadi modul *output.*

Input

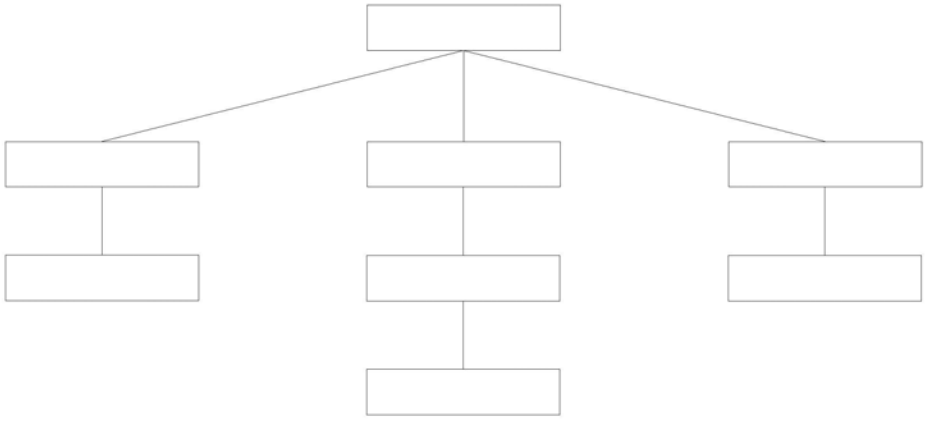
Pilih Fungsi W



Gambar 4.9 *Transform Mapping* untuk DFD level 2 Proses 2: *Hidden/Show Drives*

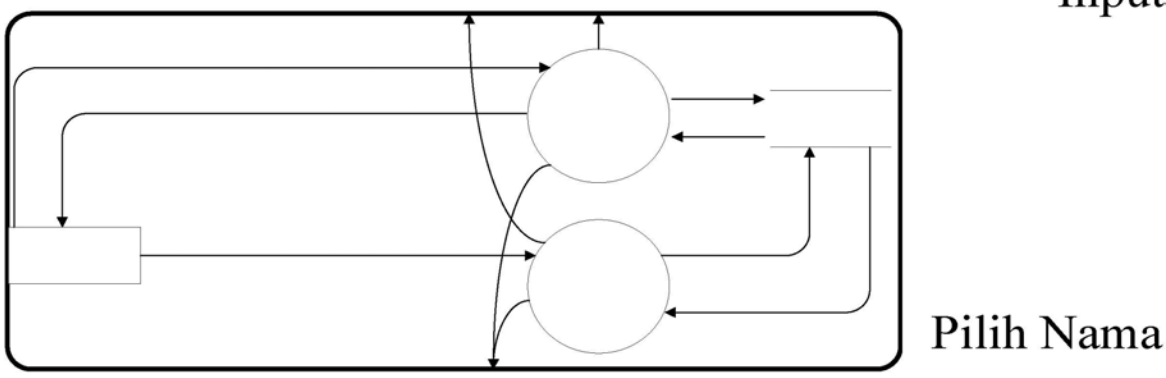
Transform mapping untuk DFD level 2 proses 2 ini terdiri atas proses pilih nama *drive* yang telah dipetakan menjadi modul *input,* proses ubah nilai *registry windows* dan proses *restart* yang telah dipetakan menjadi modul proses, dan proses akses sistem operasi yang telah dipetakan menjadi modul *output.*

Administrator



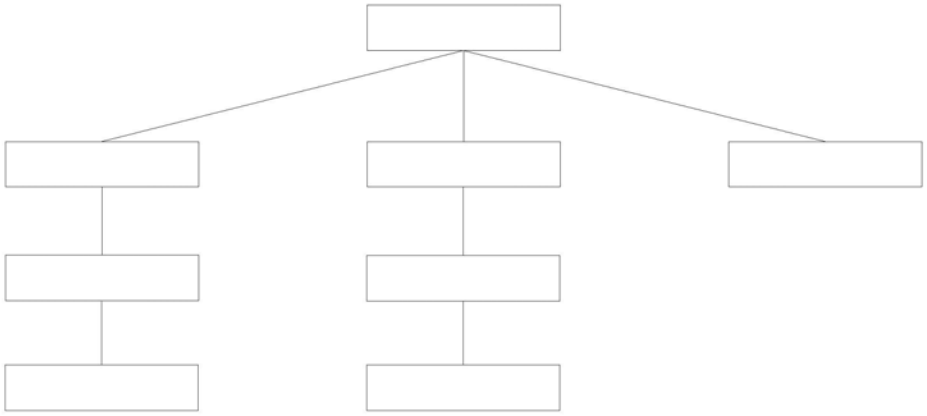
Gambar 4.10 Struktur Program untuk DFD Level 2 Proses 2: *Hidden/Show Drives*

Struktur program untuk DFD level 2 proses 2 ini terdiri atas proses pilih nama *drive* yang telah dipetakan menjadi modul *input,* proses ubah nilai *registry windows* dan proses *restart* yang telah dipetakan menjadi modul proses, dan proses akses sistem operasi yang telah dipetakan menjadi modul *output.*



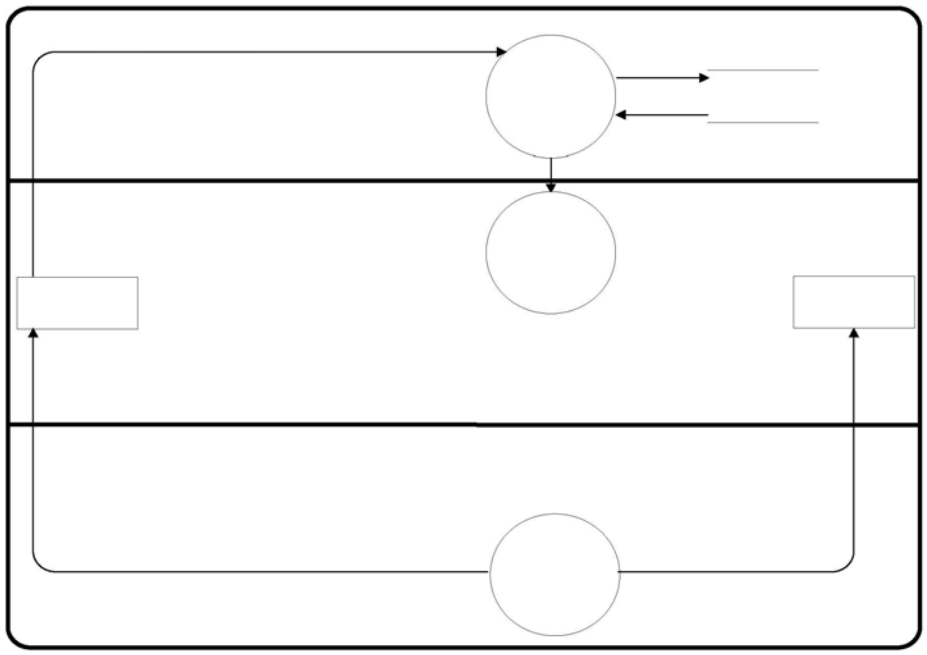
Gambar 4.11 *Transform Mapping* untuk DFD level 2 Proses 3: *Validasi*

Transform mapping untuk DFD level 2 proses 3 ini terdiri atas proses validasi *password* dan proses *clear password* yang sekaligus dipetakan menjadi modul *input* dan *output.*



Gambar 4.12 Struktur Program untuk DFD Level 2 Proses3: *Validasi*

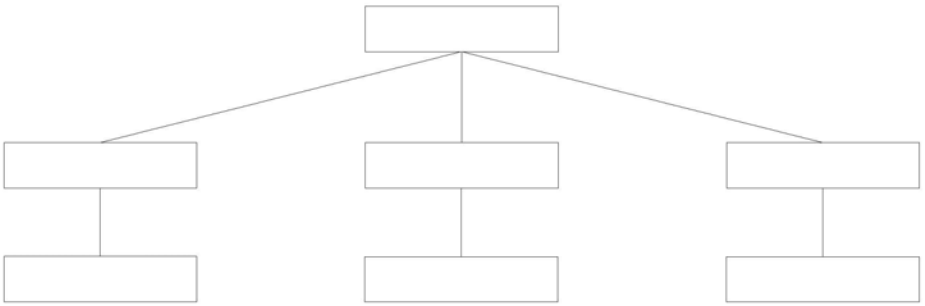
Struktur program untuk DFD level 2 proses 3 ini terdiri atas proses validasi *password* dan proses *clear password* yang sekaligus dipetakan menjadi modul *input* dan *output.*



Gambar 4.13 *Transform Mapping* untuk DFD level 2 Proses 4: Nonaktif/Aktif File*.exe

Transform mapping untuk DFD level 2 proses 4 ini terdiri atas proses pilih nama *file**.exe yang telah dipetakan menjadi modul *input,* proses ubah

nilai *registry windows* yang telah dipetakan menjadi modul proses, dan proses akses *file**.exe yang telah dipetakan menjadi modul *output.*



Gambar 4.14 Struktur Program untuk DFD Level 2 Proses 4:
Nonaktif/Aktif File*.exe

Struktur program untuk DFD level 2 proses 4 ini terdiri atas proses pilih nama *file**.exe yang telah dipetakan menjadi modul *input,* proses ubah nilai *registry windows* yang telah dipetakan menjadi modul proses, dan proses akses *file**.exe yang telah dipetakan menjadi modul *output.*

Untuk penjelasan dari proses-proses yang telah dipetakan dalam pemetaan arsitektur (*Transform Mapping)* untuk *Data Flow Diagram Level 2 di atas,* dapat dilihat pada penjelasan PSPEC tahap kedua analisis (bab3, subbab 3.5.2 halaman 9).

Input

Pilih Nama F

BAB V
IMPLEMENTASI SISTEM

5.1 Lingkungan Implementasi

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

5.1.1 Lingkungan implementasi pada saat pembangunan aplikasi *security* untuk *microsoft windows*

a. Lingkungan Perangkat Keras (*Hardware*)

Spesifikasi *hardware* pada saat pembangunan aplikasi untuk *microsoft windows,* yaitu :

1. *Processor* Pentium IV 1,7 GHz.
2. RAM 256 MB.
3. *Harddisk* 40 GB.
4. *VGA card* 32 Mb

b. Lingkungan Perangkat Lunak (*Software*)

Spesifikasi *software* yang digunakan untuk membangun aplikasi untuk *microsoft windows,* yaitu :

1. Sistem operasi : *Microsoft Windows* XP.
2. Sistem aplikasi : Borland Delphi 7.0.

5.1.2 Lingkungan implementasi pada saat penerapan sistem

a. Lingkungan Perangkat Keras (*Hardware*)

Spesifikasi *hardware* minimal pada saat penerapan aplikasi untuk *microsoft windows,* yaitu :

1. *Processor* Pentium III 1,3 GHz.
2. RAM 128 MB.
3. *Harddisk* 10 GB.
4. *VGA card* 32 Mb

b. Lingkungan Perangkat Lunak (*Software*)

Spesifikasi *software* minimal yang digunakan untuk penerapan aplikasi *security* untuk *microsoft windows,* yaitu : sistem operasi *microsoft windows 98.*

5.2 Implementasi Antarmuka

Implementasi antarmuka ini akan menampilkan tampilan dari rancangan antarmuka, beserta petunjuk penggunaan modul-modul yang terdapat dalam aplikasi yang dibangun.

Aplikasi *security* untuk *microsoft windows* diakses menggunakan file Tweak.exe.

a. Tampilan awal aplikasi *security* untuk *microsoft windows*, sebelum di*password (Form tweaking-disable/enable windows functions).*

Gambar 5.1 *Form tweaking-disable/enable windows functions*

Tampilan ini berupa modul *tweaking*, dengan sub modul berupa *disable/enable windows functions,* yang terdiri dari 10 fungsi *windows*, yang dapat dinonaktifkan sesuai hak akses yang dibatasi *administrator* kepada pengguna sistem operasi *windows.*

- Langkah-langkah mengnonaktifkan fungsi *windows*, adalah:
  1. Klik pada kotak ☐ untuk memilih fungsi *windows* mana yang akan dinonaktifkan. Sampai kotak yang diklik berubah menjadi ☑ .
  2. Klik tombol *Apply* untuk pengambilan keputusan untuk penentuan pilihan fungsi *windows* mana saja yang dinonaktifkan.
  3. Aplikasi akan mengkonfirmasikan *administrator* untuk proses *restart*. Jika *administrator* klik tombol *yes,* maka proses *restart* akan dijalankan. Dan jika *administrator* klik tombol *no*, maka proses *restart* tidak akan dijalankan dan fungsi *windows* yang dipilih untuk dinonaktifkan, belum dalam kondisi *disable*. Fungsi *windows* akan dinonaktifkan setelah proses *restart* dilakukan.

- Langkah-langkah mengaktifkan kembali fungsi *windows*, adalah:
  1. Klik pada kotak ☑ untuk memilih fungsi *windows* mana yang akan diaktifkan kembali. Sampai kotak yang diklik berubah menjadi ☐.
  2. Klik tombol *Apply* untuk pengambilan keputusan untuk penentuan pilihan fungsi *windows* mana saja yang diaktifkan kembali.
  3. Aplikasi akan mengkonfirmasikan *administrator* untuk proses *restart*. Jika *administrator* klik tombol *yes,* maka proses *restart* akan dijalankan. Dan jika *administrator* klik tombol *no*, maka proses *restart* tidak akan dijalankan dan fungsi *windows* yang dipilih untuk diaktifkan kembali, belum dalam kondisi *enable*. Fungsi *windows* akan diaktifkan kembali setelah proses *restart* dilakukan.

b. Tampilan awal / *log-in* aplikasi *security* untuk *microsoft windows*, sesudah di*password*.

Gambar 5.2 Tampilan *log-in*

- Langkah-langkah untuk *log-in*, adalah :
  1. Masukkan *password* yang *valid* pada *edit text* pada *form password.*
  2. Tekan tombol *ok* untuk masuk ke *form tweaking*. Jika *password* yang dimasukkan *invalid* maka akan tampil pesan " *Wrong Password!* ".

c. *Form tweaking-hidden/show drives*

Gambar 5.3 *Form tweaking-hidden/show drives*

Tampilan ini berupa modul *tweaking*, dengan sub modul berupa *hidden/show drives,* yang terdiri dari *drive A-Z* yang dapat disembunyikan sesuai keinginan *administrator*.

- Langkah-langkah menyembunyikan *drives*, adalah:
  1. Klik pada kotak☐ untuk memilih *drives* mana yang akan disembunyikan. Sampai kotak yang diklik berubah menjadi ☑.
  2. Klik tombol *Apply* untuk pengambilan keputusan untuk penentuan pilihan *drives* mana saja yang disembunyikan.
  3. Aplikasi akan mengkonfirmasikan *administrator* untuk proses *restart.* Jika *administrator* klik tombol *yes,* maka proses *restart* akan dijalankan. Dan jika *administrator* klik tombol *no*, maka proses *restart* tidak akan

dijalankan dan *drives* yang dipilih untuk disembunyikan, belum dalam kondisi tersembunyi. *Drives* akan disembunyikan setelah proses *restart* dilakukan.

- Langkah-langkah menampilkan kembali *drives* yang telah disembunyikan, adalah:
  1. Klik pada kotak ☑ untuk memilih *drives* mana yang akan ditampilkan kembali. Sampai kotak yang diklik berubah menjadi☐ .
  2. Klik tombol *Apply* untuk pengambilan keputusan untuk penentuan pilihan *drives* mana saja yang ditampilkan.
  3. Aplikasi akan mengkonfirmasikan *administrator* untuk proses *restart.* Jika *administrator* klik tombol *yes,* maka proses *restart* akan dijalankan. Dan jika *administrator* klik tombol *n*o, maka proses *restart* tidak akan dijalankan dan *drives* yang dipilih untuk ditampilkan, belum dalam kondisi ditampilkan. *Drives* akan ditampilkan setelah proses *restart* dilakukan.

d. Form tweaking-restrict running

Gambar 5.4 *Form tweaking-restrict running*

Tampilan ini berupa modul *tweaking*, dengan sub modul berupa *restrict running*, yang berfungsi untuk menonaktifkan/mengaktifkan kembali *file*.exe*.

- Langkah-langkah untuk menonaktifkan *file*.exe*
  1. Klik tombol *disable,* aplikasi akan membuka *windows explorer* untuk memilih *file*.exe* mana yang akan dinonaktifkan
  2. Pilih *file*.exe* yang akan dinonaktifkan, dan klik tombol *open* untuk mendaftarkan *file*.exe* mana yang akan dinonaktifkan ke dalam list yang tersedia pada *form tweaking-restrict running*.
  3. Klik tombol *apply* untuk melakukan proses *restart*.

- Langkah-langkah untuk mengaktifkan kembali *file*.exe*
  1. Klik nama *file*.exe* yang tersedia pada *list file*.exe* yang telah dinonaktifkan.
  2. Klik tombol *enable* untuk mengaktifkan kembali *file*.exe* yang telah dinonaktifkan.
  3. Klik tombol *apply* untuk melakukan proses *restart*.

e. *Form password*

Gambar 5.5 *Form password*

- Langkah-langkah menset *password*
  1. Masukkan *password* yang sama pada kedua *edit text* yang tersedia.
  2. Klik tombol *set* untuk menset *password*. Jika *password* yang dimasukkan tidak sama antara kedua *edit text*, maka akan tampil pesan "The Password you Typed Did Not Match!"

- Langkah-langkah mengosongkan *password*
  1. Klik ganda pada *edit text1* yang telah di*password*, tekan tombol *delete* pada *keybord* untuk mengosongkan *edit text1*.
  2. Klik ganda pada *edit text2* yang telah di*password*, tekan tombol *delete* pada *keybord* untuk mengosongkan *edit text2*.
  3. Klik tombol *clear* untuk mengosongkan *password*, yang berfungsi untuk mengakses aplikasi *security* untuk *microsoft windows* tanpa melalui proses *log-in* terlebih dahulu.

BAB VI

KESIMPULAN DAN SARAN

Pada bagian ini, akan dijelaskan mengenai kesimpulan yang didapat selama pembangunan aplikasi *security* untuk *microsoft windows* dan saran-saran untuk perbaikan dan pengembangan aplikasi *security* untuk *microsoft windows*.

6.1 Kesimpulan

Kesimpulan yang dapat diambil selama pembangunan aplikasi *security* untuk *microsoft windows*, adalah:

1. Aplikasi *security* untuk *microsoft windows* ini dapat digunakan untuk membatasi wewenang *user* dalam hal pemakaian fungsi-fungsi sistem operasi *microsoft windows*.
2. Aplikasi *security* untuk *microsoft windows* ini dapat menyembunyikan *drives* yang dianggap sebagai tempat penyimpan data penting.
3. Aplikasi *security* untuk *microsoft windows* ini dapat menonaktifkan *program* yang berupa *file*.exe* yang dianggap sebagai program pengolahan data penting.

6.2 Saran

Saran yang dapat diberikan untuk perbaikan dan pengembangan selanjutnya pada aplikasi *security* untuk *microsoft windows*, adalah:

1. *Security* untuk *microsoft windows* diharapkan dapat menangani *security* yang mencakup pada masalah jaringan.
2. *Security* untuk *microsoft windows* diharapkan hanya dapat diinstalasi dan dikontrol oleh *administrator windows*.
3. *Security* untuk *microsoft windows* diharapkan dapat melakukan *recovery* jika terjadi *corrupted file*.