BAB 2
TINJAUAN PUSTAKA

2.1 Profil Tempat Kerja Praktek

PT Eforel Citra Utama merupakan Premier Channel Partner Advantech Indonesia. Berdiri sejak tahun 1994, Eforel telah mengembangkan bisnis dalam bidang *Industrial Computer*, *Automation* & *Control System*. Kompetensi utama PT Eforel Citra Utama adalah melakukan kegiatan bisnis dengan produk-produk Advantech & hal tersebut diperkuat dengan strategi total solusi kami.

2.1.1 Sejarah Instansi

PT Eforel Citra Utama berdiri pada tanggal 2 Februari 1994, dengan bidang usaha yang ditangani adalah meliputi elektrikal, instrumentasi, suplier dan kontraktor. Adanya transformasi kemajuan teknologi yang pesat membuat PT Eforel Citra Utama secara khusus mengembangkan diri pada bidang teknologi informasi dan otomasi. Untuk mendukung riset dan pengembangan, PT Eforel Citra Utama bekerja sama dengan Advantech yang telah bersertifikasi ISO 9001 untuk penyediaan perangkat keras yang memenuhi standar industri.

2.1.2 Logo Instansi

Deskripsi dari logo instansi PT Eforel Citra Utama yaitu, warna biru muda pada logo melambangkan bahwa PT Eforel Citra Utama merupakan salah satu mitra Advantech, karena Advantech didominasi oleh warna biru muda, dan gambar yang berwarna silver seperti huruf E pada logo melambangkan huruf pertama dari kata "EFOREL".



Gambar 2.1 Logo PT Eforel Citra Utama

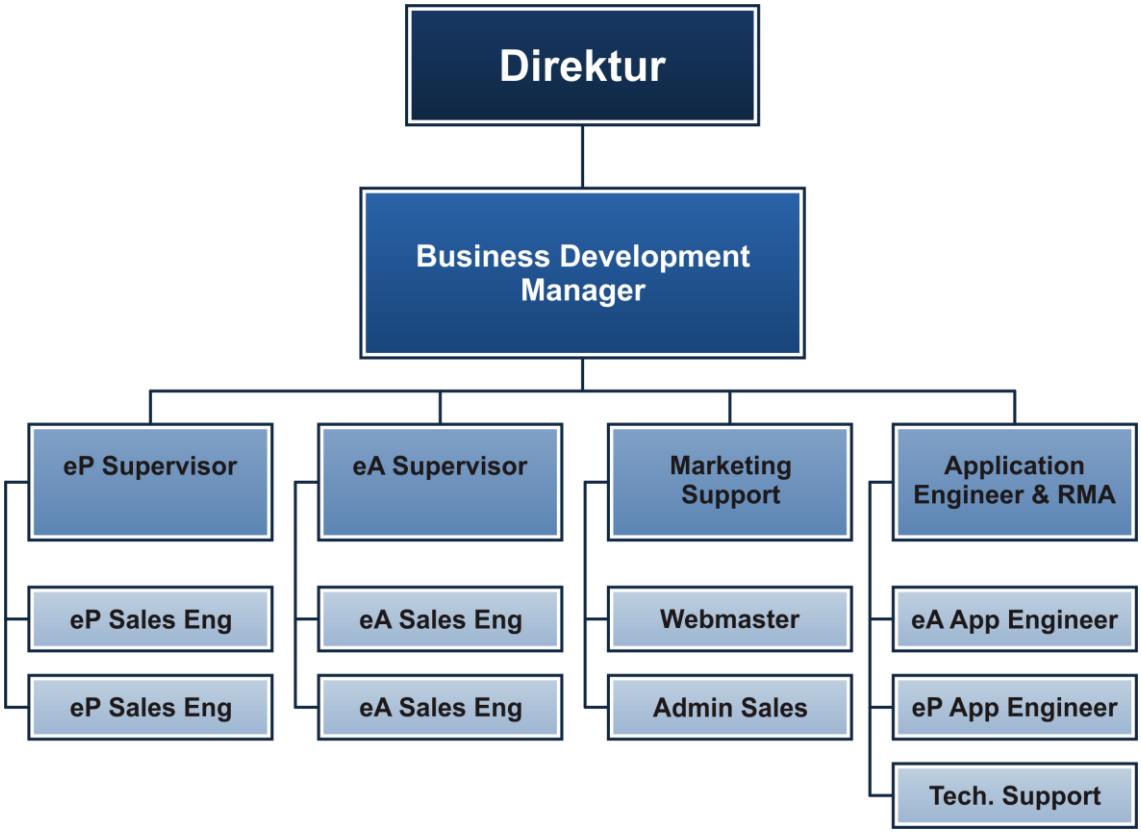
2.1.3 Visi PT EFOREL CITRA UTAMA

PT Eforel Citra Utama adalah menjadi distributor komputer industri dan peralatan otomasi yang terpercaya di Indonesia dengan pertumbuhan pangsa pasar & nilai penjualan yang tumbuh secara berkelanjutan.

2.1.4 Misi PT EFOREL CITRA UTAMA

PT EFOREL CITRA UTAMA adalah pengembangan pangsa pasar dan komunitas pelanggan terhadap produk berteknologi tinggi dengan harga terjangkau melalui dukungan teknis dan pelayanan pelanggan yang lebih baik.

2.1.5 Struktur Organisasi



Gambar 2.2 Struktur Organisasi

Deskripsi Jabatan :

1. *Direktur*

   Seseorang yang ditunjuk untuk memimpin Perseroan Terbatas (PT). Direktur dapat seseorang yang memiliki perusahaan tersebut atau orang profesional yang ditunjuk oleh pemilik usaha untuk menjalankan dan memimpin perseroan terbatas.

2. *Business Development Manager*

   Bertanggungjawab untuk mempertahankan dan mengembangkan kinerja divisi-divisi produk yang sekarang ada dan memberi saran misalnya untuk mendivestasi produk tertentu yang telah minim hasilnya dan mendorong produk perkembangan yang marketnya akan berkembang, disamping itu mencari bidang-bidang baru untuk diversifikasi, dengan proyek-proyek baru yang dituangkan dalam suatu *Feasiblity Study*.

3. *eP Supervisor*

   Seseorang yang menangani orang-orang yang memproduksi dan atau melakukan kinerja pelayanan.

4. *eP Sales Eng*

   Seorang *Sales Engineer* biasanya masuk setelah bagian lain dari tim sales mem-*follow up* sebuah *project*. *Sales Engineer* biasanya bertanggungjawab terhadap hal berikut :

   a. Bertugas untuk memberikan solusi terhadap masalah atau kebutuhan pelanggan secara teknikal.

   b. Mempresentasikan produk atau solusi sesuai kebutuhan pelanggan.

   c. Menjawab RFP (*Request For Proposal*) apakah *comply* antara kebutuhan pelanggan dengan produk yang dimiliki.

   d. Mendorong pelanggan untuk menggunakan produk yang dimiliki, bahkan bila kebutuhan belum terpikirkan oleh pelanggan (JUALAN).

5. *eA Supervisor*

   Bertanggungjawab untuk hasil atas orang-orang yang diawasi terutama mutu dan jumlah dari produk dan pelayanan.

6. *eA Sales Eng*

   Seseorang yang mempunyai kemampuan menjual suatu produk dan mempunyai keahlian sebagai teknisi.

7. *Marketing Support*

   Divisi yang menunjang dan memberikan pelayanan kepada Divisi Sales & Marketing

8. *Webmaster*

   Orang yang bertanggung jawab untuk memelihara website, termasuk memastikan *server web*, perangkat keras dan perangkat lunak agar beroperasi secara akurat, merancang website, menghasilkan dan merevisi halaman web, membalas komentar pengguna, dan memeriksa lalu lintas melalui situs. Webmaster juga harus berpengalaman dalam transaksi perangkat lunak Web dan keamanan.

9. *Admin Sales*

   a. Membuat Surat Jalan, dan memeriksa jadwal serta tagihan sales.

   b. Mengontrol batas kredit langganan.

   c. Membantu tugas sales didalam kantor dengan menyiapkan segala keperluan yang dibutuhkan Sales, dan lain-lain.

10. *Application Engineer & RMA*

    Engineer yang bertugas memilih/menghitung produk mana yang sesuai dengan permintaan pembeli.

11. *eA Engineer*

    Seseorang yang bekerja dibidang *engineering* dan mampu menggunakan ilmunya untuk menghasilkan suatu benda yang dapat dimanfaatkan dalam kehidupan sehari-hari.

12. *eP Engineer*

    Suatu ilmu keteknikan yang dipraktekkan kedalam kehidupan kita untuk mempermudah kita dalam melakukan sesuatu.

13. *Technical Support*

    Seseorang yang mampu untuk memperbaiki suatu sistem yang bermasalah dalam jaringan. *Technical Support* juga merupakan staf pendukung teknis komputer yang dapat mengoperasikan PC *stand alone*, instalasi OS, instalasi dan konfigurasi jaringan, instalasi modem dan setting konfigurasi komunikasi *wireless*, instalasi *software* anti virus, *restore* dan *backup system*, penanganan awal atas masalah PC.

2.2 Landasan Teori

2.2.1 Pengertian Sistem Informasi

Deskripsi mengenai pengertian sistem informasi menurut beberapa ahli :

a. Sistem Informasi merupakan sistem yang mempunyai kemampuan untuk mengumpulkan informasi dari semua sumber dan menggunakan berbagai media untuk menampilkan informasi [2].

b. Sistem informasi adalah suatu sistem didalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang mendukung fungsi organisasi yang bersifat manajerial dalam kegiatan strategi dari suatu organisasi untuk dapat menyediakan kepada pihak luar tertentu dengan laporan-laporan yang diperlukan [3].

c. Sistem informasi adalah sistem didalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, membantu dan mendukung kegiatan operasi, bersifat manajerial dari suatu organisasi dan membantu mempermudah penyediaan laporan yang diperlukan [4].

d. Sistem informasi adalah data yang dikumpulkan, dikelompokkan dan diolah sedemikian rupa sehingga menjadi sebuah satu kesatuan informasi yang saling terkait dan saling mendukung sehingga menjadi suatu informasi yang berharga bagi yang menerimanya [5].

2.2.1.1 Karakteristik Sistem

Suatu sistem mempunyai karakteristik tertentu, yaitu mempunyai komponen-komponen (*components*), batas sistem (*boundary*), lingkaran luar sistem (*environment*), penghubung (*interface*), masukkan (*input*), keluaran (*output*), pengolah (*process*), dan sasaran (*objective*) atau tujuan (*goal*). Dibawah ini merupakan penjelasan dari masing-masing karakteristik tersebut :

1. Komponen Sistem (*components*)

   Sistem terdiri dari sejumlah komponen yang saling berinteraksi, bekerja sama membentuk kesatuan. Komponen-komponen atau elemen-elemen sistem dapat berupa suatu subsistem atau bagian-bagian dari sistem. Setiap subsistem mempunyai sifat-sifat dari sistem untuk menjalankan suatu fungsi tertentudan mempengaruhi proses sistem secara keseluruhan.

2. Batas Sistem (*boundary*)

   Batas sistem merupakan daerah yang membatasi antara suatu sistem dengan sistem yang lainnya atau dengan lingkungan luarnya. Batas sistem ini memungkinkan sistem dipandang sebagai suatu kesatuan. Batas suatu sistem menunjukkan ruang lingkup (*scoope*) dari sistem tersebut.

3. Lingkungan Luar Sistem (*environment*)

   Lingkungan luar dari suatu sistem adalah apapun diluar batas dari sistem yang mempengaruhi operasi sistem. Lingkungan yang menguntungkan harus tetap dijaga dan dipelihara karena merupakan *energy* dari sistem. Sedangkan lingkungan luar yang merugikan harus ditahan dan dikendalikan, karena jika tidak akan mengganggu kelangsungan sistem.

4. *Interface*

   Interface merupakan media penghubung antara suatu subsistem dengan subsistem yang lainnya. *Interface* ini memungkinkan satu subsistem untuk mengalirkan sumber daya ke subsistem lainnya.

5. *Input*

   Input merupakan energi yang dimasukkan ke dalam sistem. *Input* dapat berupa *maintenance input* dan *signal input. Maintenance input* adalah energi yang dimasukkan supaya sistem tersebut dapat beroperasi. *Signal Input* adalah energi yang diproses untuk menghasilkan *output*.

6. *Output*

   Output merupakan hasil dari energi yang diolah dan diklasifikasikan menjadi *output* yang berguna dan sisa pembuangan. *Output* dapat menjadi *input* untuk subsistem yang lain.

7. Pengolah Sistem (*Process*)

   Suatu sistem mempunyai bagian pengolah yang akan merubah masukan menjadi keluaran.

8. Sasaran Sistem (*objective*)

   Suatu sistem mempunyai tujuan (*goal*) atau sasaran (*objective*). Sasaran dari sistem menentukan *Input* yang dibutuhkan dan *Output* yang akan dihasilkan.

2.2.1.2 Klasifikasi Sistem

Sistem dapat diklasifikasikan melalui beberapa sudut pandang, diantaranya :

1. Sistem diklasifikasikan sebagai sistem abstrak (*abstract system*) dan sistem fisik (*physical system*).

   Sistem abstrak adalah sistem yang berupa pemikiran atau ide-ide yang tidak tampak secara fisik.

2. Sistem diklasifikasikan sebagai sistem alamiah (*natural system*) dan sistem buatan manusia (*human made system*).

   Sistem alamiah adalah sistem yang terjadi melalui sistem alam, tidak dibuat oleh manusia.

3. Sistem diklasifikasikan sebagai sistem tertentu (*deterministic system*) dan sistem tak tentu (*probabilistic system*).

Sistem tertentu beroperasi dengan tingkah laku yang sudah diprediksi. Interaksi diantara bagian-bagiannya dapat diprediksi dengan pasti, sehingga *output* dari sistem dapat diramalkan. Sistem tak tentu adalah sistem yang kondisi masa depannya tidak dapat diprediksi karena mengandung unsur probabilitas.

4. Sistem diklasifikasikan sebagai sistem tertutup (*closed system*) dan sistem terbuka (*open system*).

   Sistem tertutup merupakan sistem yang tidak berhubungan dan tidak terpengaruh dengan lingkungan luarnya. Secara teoritis sistem tertutup ini ada, tetapi yang ada hanyalah *relatively closed system* (secara relative tertutup, tetapi tidak benar-benar tertutup). Sistem terbuka adalah sistem yang berhubungan dan terpengaruh dengan lingkungan luarnya. Sistem ini menerima *input* dan menghasilkan *output* untuk lingkungan luar atau subsistem yang lain.

2.2.1.3 Konsep Dasar Informasi

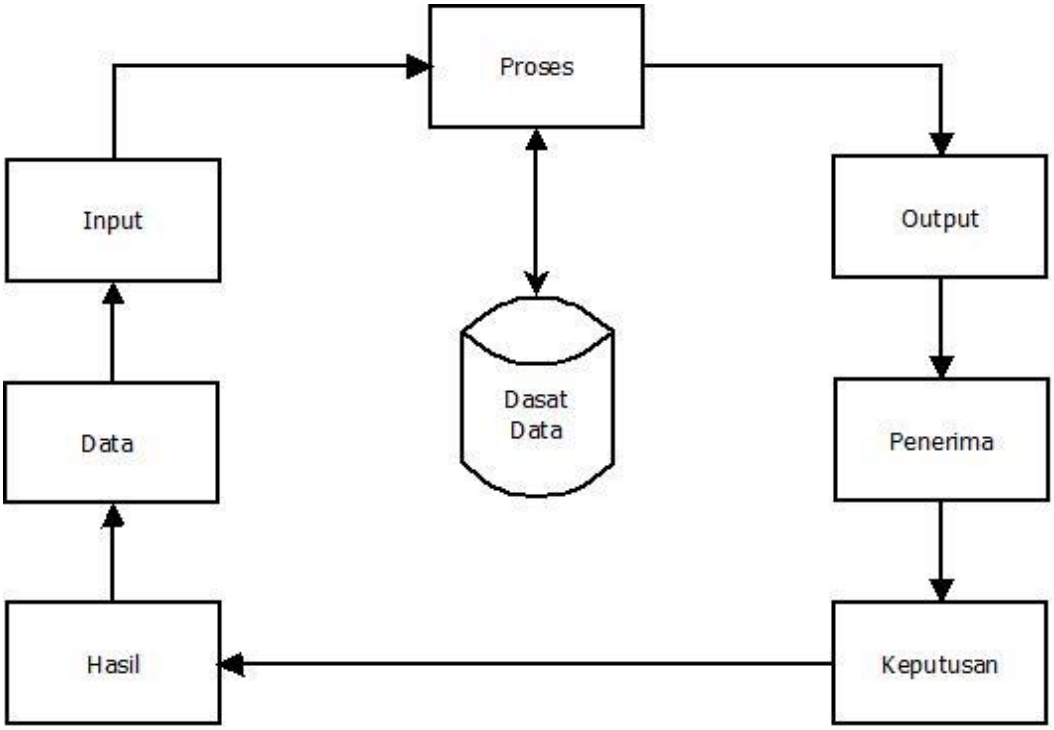
Sumber dari informasi adalah data. Data adalah kenyataan yang menggambarkan suatu kejadian-kejadian dan kesatuan nyata. Kejadian-kejadian adalah sesuatu yang terjadi pada saat tertentu. Informasi dapat didefinisikan sebagai :

"*Data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi yang menerimanya*".

1. Siklus Informasi

   Data merupakan bentuk yang masih mentah yang belum dapat berceritera banyak, sehingga perlu diolah lebih lanjut. Data diolah melalui suatu model untuk dihasilkan informasi.

   Data yang diolah melalui suatu model menjadi suatu informasi, penerima kemudian menerima informasi tersebut, membuat sutu keputusan dan melakukan tindakan yang berarti menghasilkan sutu tindakana yang lain yang akan membuat sejumlah data kembali. Data tersebut akan ditangkap sebagai *input*, diproses kembali lewat suatu model membentuk suatu siklus.



Gambar 2.3 Siklus Informasi

2. Teknologi Informasi

Informasi teknologi adalah sebuah *terminology* kontemporer yang mendeskripsikan kombinasi antara teknologi komputer (*hardware* dan *software*) dengan teknologi komunikasi (data, gambar, dan jaringan suara).

Peranan Teknologi Informasi :

a. Teknologi informasi menggantikan peran manusia. Dalam hal ini, teknologi informasi melakukan otomasi terhadap suatu tugas atau proses.

b. Teknologi memperkuat peran manusia, yakni dengan menyajikan informasi terhadap suatu tugas atau proses.

c. Teknologi informasi berpertan dalam restruksisasi terhadap peran manusia. Dalam hal ini, teknologi berperan dalam melakukan perubahan-perubahan terhadap sekumpulan tugas atau proses.

3. Kualitas Informasi

   Kualitas dari suatu informasi tergantung dari tiga hal, yaitu informasi harus akurat(*accurate*), tepat waktunya (*timeliness*) dan relevan (*relevance*) :

   a. Akurat, berartti informasi harus bebas dari kesalahan-kesalahan dan tidak bias atau menyesatkan. Akurat juga berarti informasi harus jelas mencerminkan maksudnya.

   b. Tepat pada waktunya, berarti informasi yang datang pada penerima tidak boleh terlambat. Informasi yang sudah using tidak akan mempunyai nilai lagi. Bila pengambilan keputusan terlambat, maka dapat berakibat fatal untuk organisasi.

   c. Relevan, berarrti informasi tersebut mempunyai manfaat untuk pemakainya, dengan cara mengurangi ketidakpastian, menaikan kemampuan untuk memprediksi atau menegaskan ekspetasi semula.

4. Nilai Informasi

   Nilai dari informasi (*Value of Information*) dimasukan dari 2 hal yaitu manfaat dan mendapatkannya. Suatu informasi dikatakan bernilai manfaaatnya lebih efektif dibandingkan dengan biaya mendapatkannya. Akan tetapi perlu diperhatikan bahwa informasi informasi yang digunakan didalam suatu sistem informasi umumnya digunkan untuk beberapa kegunaan. Sebagian informasi tidak dapat ditaksir keuntungannya dengan suatu nilai tetapi dapat ditaksir nilai keefektifannya.

2.2.1.4 Konsep Dasar Sistem Informasi

Informasi merupakan hala yang sangat penting didalam mengambil keputusan. Informasi tersbut didapatkan dari sitem informasi (*information system*) atau disebut juga dengan *processing systems* atau *information processing systems* atau *information generating systems*. Sistem informasi adalah :

"Suatu sistem didalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi, bersifat manejerial dan kegiatan

strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan."

2.2.2 Basis Data

Basis Data adalah kumpulan informasi yang disimpan di dalam komputer secara sistematik sehingga dapat diperiksa menggunakan suatu program komputer untuk memperoleh informasi dari basis data tersebut.

2.2.3 ERD

ERD merupakan suatu model untuk menjelaskan hubungan antar data dalam basis data berdasarkan objek-objek dasar data yang mempunyai hubungan antar relasi.

a. *Entiti*

Entiti merupakan objek yang mewakili sesuatu yang nyata dan dapat dibedakan dari sesuatu yang lain. Simbol dari entiti ini biasanya digambarkan dengan persegi panjang.

b. *Atribut*

Setiap entitas pasti mempunyai elemen yang disebut atribut yang berfungsi untuk mendeskripsikan karakteristik dari entitas tersebut. Isi dari atribut mempunyai sesuatu yang dapat mengidentifikasikan isi elemen satu dengan yang lain. Gambar atribut diwakili oleh simbol elips.

c. *Hubungan/Relasi*

Hubungan antara sejumlah entitas yang berasal dari himpunan entitas yang berbeda. Relasi dapat digambarkan sebagai berikut :

Relasi yang terjadi diantara dua himpunan entitas (misalnya A dan B) dalam satu basis data yaitu :

1. Satu ke satu (*One to one*)

   Hubungan relasi satu ke satu yaitu setiap entitas pada himpunan entitas A berhubungan paling banyak dengan satu entitas pada himpunan entitas B.

2. Satu ke banyak (*One to many*)

   Setiap entitas pada himpunan entitas A dapat berhubungan dengan banyak entitas pada himpunan entitas B, tetapi setiap entitas pada entitas B dapat berhubungan dengan satu entitas pada himpunan entitas A.

3. Banyak ke banyak (*Many to many*)

   Setiap entitas pada himpunan entitas A dapat berhubungan dengan banyak entitas pada himpunan entitas B.

2.2.3.1 Varian Entitas dan Relasi

Idealnya himpunan entitas yang terlibat dalam sebuah ERD adalah himpunan entitas kuat dan bebas. Himpunan entitas demikian tidak memiliki ketergantungan dengan himpunan entitas lainnya. Namun demikian, dalam pembentukan ERD tidak selalu dapat melibatkan himpunan entitas seperti itu. Ada kalanya juga melibatkan himpunan entitas yang lemah atau merupakan bagian dari himpunan entitas lainnya.

2.2.3.1.1 Himpunan Entitas Lemah (*Weak Entity Sets*)

Berisi entitas-entitas yang kemunculannya tergantung pada eksistensinya dalam sebuah relasi terhadap entitas lain. Himpunan entitas demikian biasanya tidak memiliki atribut yang dapat berfungsi sebagai *key*, yang dapat menjamin keunikan entitas di dalamnya.

2.2.3.1.2 Sub Entitas (*Subtype Entities*)

Merupakan himpunan entitas yang beranggotakan entitas-entitas yang merupakan bagian dari himpunan entitas yang lebih superior (utama). Sub Entitas ini merupakan hasil dekomposisi (spesialisasi) himpunan entitas berdasarkan pengelompokan tertentu.

2.2.4 DFD

Data Flow Diagram (DFD) adalah alat pembuatan model yang memungkinkan profesional sistem untuk menggambarkan sistem sebagai suatu jaringan proses

fungsional yang dihubungkan satu sama lain dengan alur data, baik secara manual maupun komputerisasi.

DFD ini merupakan alat perancangan sistem yang berorientasi pada alur data dengan konsep dekomposisi dapat digunakan untuk penggambaran analisa maupun rancangan sistem yang mudah dikomunikasikan oleh profesional sistem kepada pemakai maupun pembuat program.

a. *Komponen Terminator/Entitas Luar*

   Terminator mewakili entitas eksternal yang berkomunikasi dengan sistem yang sedang dikembangkan. Biasanya terminator dikenal dengan nama entitas luar (*external entity*).

   Komponen terminator ini perlu diberi nama sesuai dengan dunia luar yang berkomunikasi dengan sistem yang sedang dibuat modelnya, dan biasanya menggunakan kata benda, misalnya Admin.

b. *Komponen Proses*

   Komponen proses menggambarkan bagian dari sistem yang mentransformasikan *input* menjadi *output*. Proses diberi nama untuk menjelaskan proses/kegiatan apa yang sedang/akan dilaksanakan. Pemberian nama proses dilakukan dengan menggunakan kata kerja transitif (kata kerja yang membutuhkan obyek).

c. *Komponen Data Store*

   Komponen ini digunakan untuk membuat model sekumpulan paket data dan diberi nama dengan kata benda jamak, misalnya Mahasiswa.

   Data store ini biasanya berkaitan dengan penyimpanan-penyimpanan, seperti file atau database yang berkaitan dengan penyimpanan secara komputerisasi, misalnya file disket, file harddisk, file pita magnetik. *Data store* juga berkaitan dengan penyimpanan secara manual seperti buku alamat, file, folder, dan agenda.

   Suatu *data store* dihubungkan dengan alur data hanya pada komponen proses, tidak dengan komponen DFD lainnya. Alur data yang menghubungkan *data store* dengan suatu proses mempunyai pengertian sebagai berikut :

1. Alur data dari *data Store* yang berarti sebagai pembacaan atau pengaksesan satu paket tunggal data, lebih dari satu paket data, sebagian dari satu paket tunggal data, atau sebagian dari lebih dari satu paket data untuk suatu proses
2. Alur data ke *data store* yang berarti sebagai peng-*update*-an data, seperti menambah satu paket data baru atau lebih, menghapus satu paket atau lebih, atau mengubah/memodifikasi satu paket data atau lebih .

d. *Komponen Data Flow/Alur Data*

Suatu data flow/alur data digambarkan dengan anak panah, yang menunjukkan arah menuju ke dan keluar dari suatu proses. Alur data ini digunakan untuk menerangkan perpindahan data atau paket data/informasi dari satu bagian sistem ke bagian lainnya. Alur data perlu diberi nama sesuai dengan data/informasi yang dimaksud, biasanya pemberian nama pada alur data dilakukan dengan menggunakan kata benda.

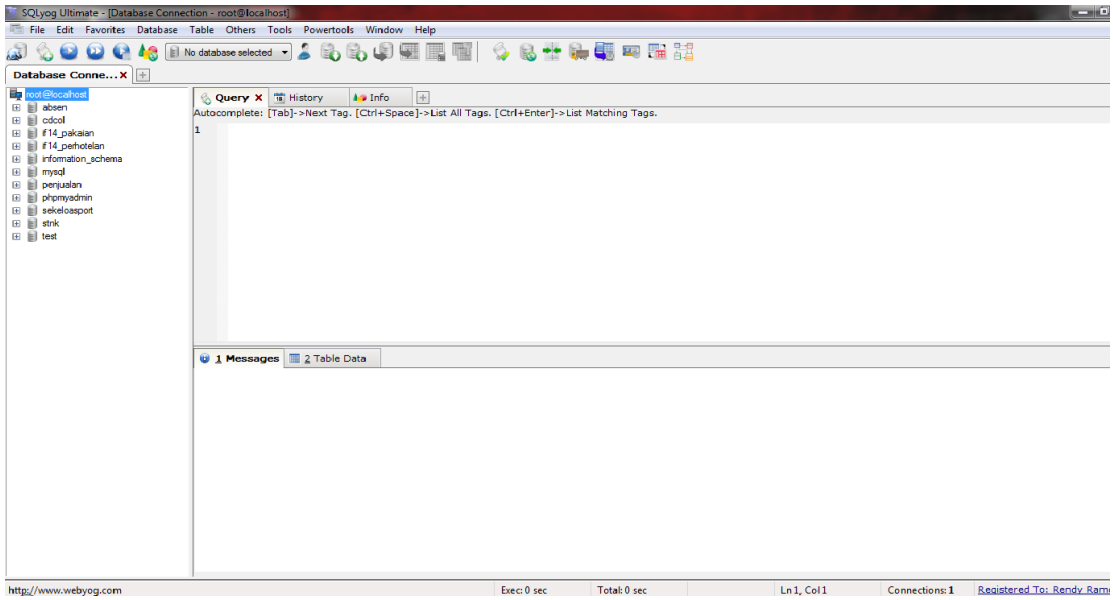
2.2.5 *Query*

Query adalah permintaan yang diberikan oleh *user* untuk mengambil informasi yang tersimpan dalam *database*.

2.2.6 Perangkat Database *SQLyog*

SQLyog adalah salah satu software yang bisa digunakan sebagai GUI *database manager* untuk *database MySQL*. Software ini banyak fungsinya yaitu disediakanya untuk membuat tabel baru, membuat index, merubah struktur tabel yang telah dibuat. Dan bisa juga melakukan *backup database*, *backup* sangat perlu di lakukan untuk menghindari terjadinya kesalahan-kesalahan yang tidak bisa diprediksi akan terjadi.

Berikut tampilan dari Database *SQLyog* :



Gambar 2.4 Tampilan Database SQLyog

2.2.7 *C# (C Sharp)*

C# (C Sharp) merupakan sebuah bahasa pemrograman yang berorientasi objek yang dikembangkan oleh *Microsoft* sebagai bagian dari inisiatif kerangka *.NET Framework*. Bahasa pemrograman ini dibuat berbasiskan bahasa *C++* yang telah dipengaruhi oleh aspek-aspek ataupun fitur bahasa yang terdapat pada bahasa-bahasa pemrograman lainnya seperti *Java*, *Delphi*, *Visual Basic*, dan lain-lain) dengan beberapa penyederhanaan. Menurut standar ECMA-334 *C# Language Specification*, nama *C#* terdiri atas sebuah huruf Latin C (U+0043) yang diikuti oleh tanda pagar yang menandakan angka # (U+0023). Tanda pagar (#) yang digunakan memang bukan tanda kres dalam seni musik (U+266F), dan tanda pagar # (U+0023) tersebut digunakan karena karakter kres dalam seni musik tidak terdapat di dalam keyboard standar. kurang lebih seperti itu. Apa saja yang dimikiki oleh *C#* mari kita lihat deskripsi dibawah ini :

1. Sederhana (*Simple*)

   C# bersifat sederhana, karena bahasa ini didasarkan kepada bahasa *C* dan *C++*. Jika anda familiar dengan *C* dan *C++* atau bahkan *Java*, anda akan menemukan aspek-aspek yang begitu familiar, seperti *statements*, *expression*, *operators*, dan beberapa fungsi yang diadopsi langsung dari *C* dan *C++*, tetapi dengan berbagai perbaikan yang membuat bahasanya menjadi lebih sederhana.

2. OOP (*Object Oriented Languange*)

   C# memenuhi syarat-syarat sebagai sebuah bahasa pemrograman yang bersifat *Object Oriented*, yaitu *encapsulation*, *inheritance* dan *polymorphism*.

3. *Powerfull dan Flexsibel*

   C# bisa digunakan untuk membuat berbagai macam aplikasi, seperti aplikasi pengolah kata, grafik, *spreadsheets*, atau bahkan membuat kompiler untuk sebuah bahasa permrograman.

4. Efesien

   C# tidak memiliki terlalu banyak *keyword*, sehingga dapat mengurangi kerumitan.

5. Modular

   Kode C# ditulis dengan pembagian masing *class-class* (*classes*) yang terdiri dari beberapa routines yang disebut sebagai *member methods*. *Class-class* dan metode-metode ini dapat digunakan kembali oleh program atau aplikasi lain. Hanya dengan memberikan informasi yang dibutuhkan oleh *Class* dan metode yang dimaksud, maka akan dapat membuat suata kode yang dapat digunakan oleh satu atau beberapa aplikasi dan program (*reusable code*).

A. Kelebihan C#

1. *Flexible*

   C# dapat di eksekusi di mesin komputer sendiri atau di transmiskan melalui web dan dieksekusi di komputer lainya.

2. *Powerfull*

   C# memiliki sekumpulan perintah yang sama dengan *C++* yang kaya akan fitur yang lengkap tetapi dengan gaya bahasa yang lebih diperhalus sehingga memudahkan penggunanya.

3. *Easier to use*

   C# memodifikasi perintah yang sepenuhnya sama dengan *C++* dan memberitahu dimana letak kesalahan kita bila ada kesalahan dalam aplikasi , hal ini dapat mengurangi waktu kita dalam mencari *error*.

4. *Visually oriented*

   .NET library code yang digunakan oleh *C#* menyediakan bantuan yang dibutuhkan untuk membuat tampilan yang *complicated* dengan *frames, dropdown , tabbed windows, group button , scroll bar , background image* , dan lainnya.

5. *Secure*

   Semua bahasa pemprograman yang digunakan untuk kebutuhan internet mesti memiliki *security* yang benar-benar aman untuk menghindari aksi kejahatan dari pihak lain seperti *hacker* , *C#* memiliki segudang fitur untuk menanganinya.

6. Memory management lebih mudah karena adanya *garbage collector*, yang membebaskan memory secara otomatis sehingga dapat mencegah *memory leak.*

7. *Type safe*, konversi implisit dari tipe data hanya mensupport turunan dan operasi dari tipe data yang melebar (misal dari *int* ke *long*, kalo int ke short tidak bisa) dan ini dideteksi pas *compile*.

8. Banyak fungsi yang tersedia di *Base Class Library .NET Framework.*

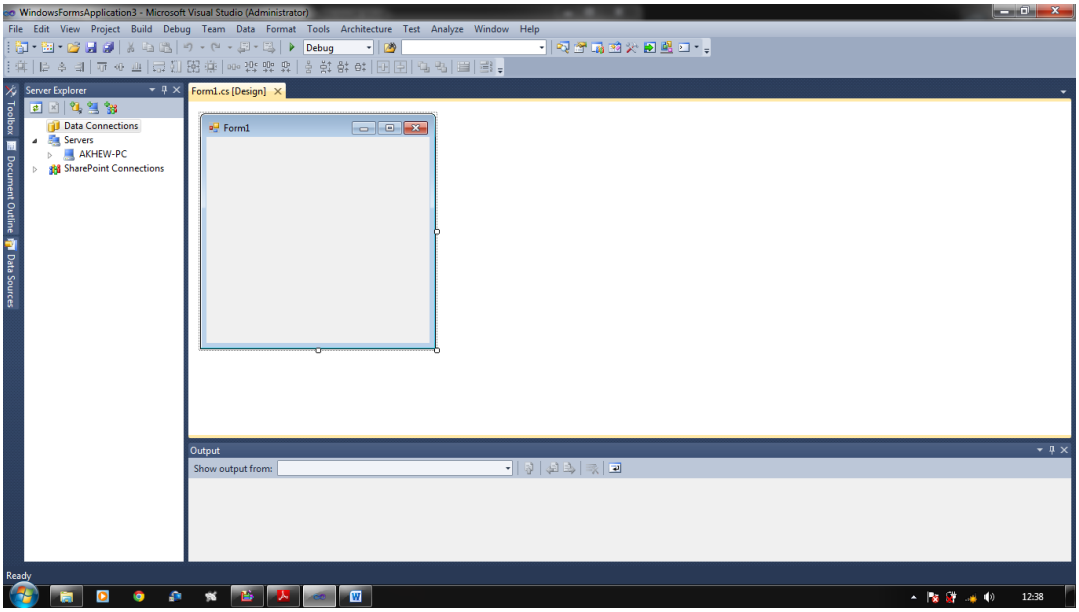
9. *.NET Framework* berkembang cepat dan semakin banyak fitur yang membuat produktivitasnya bertambah, misal: *linq*.

10. Untuk pengembangan aplikasi bisnis/umum atau enterprise, penggunaan *C#* akan lebih produktif daripada bila menggunakan *C++*. Bahasa *C#* masih merupakan turunan dari bahasa C, tetapi seolah-olah dibuat lebih mudah dan produktif seperti *Visaul Basic* dengan tetap mempertahankan fleksibilitas dan "*power*" dari bahasa *C*.

11. Kabarnya *Microsoft* sedang mengembangkan sistem operasi dengan *code name "Singularity"/"Midori"* yang akan menggantikan *Windows*. Sistem operasi ini, API-nya ditulis dengan *C#* dan merupakan *managed code* seperti *.Net Framework* saat ini sehingga tentunya akan mengusung *C#* sebagai bahasa pemprograman standar. Hal ini akan menambah lagi kelebihan *C#*.

B. Kelemahan *C#*

1. Banyaknya operator serta fleksibilitas penulisan program kadang-kadang membingungkan pemakai.
2. Bagi pemula pada umumnya akan kesulitan menggunakan *pointer*.



Gambar 2.5 Tampilan C# (C Sharp)