BAB II

TINJAUAN PUSTAKA

2.1 Dinas Sosial Provinsi Jawa Barat

Dinas Sosial Provinsi Jawa Barat merupakan Dinas Teknis Daerah yang berada dan bertanggung jawab kepada Gubernur melalui Sekretaris Daerah Provinsi Jawa Barat yang mempunyai tugas pokok merumuskan kebijakan operasional di bidang kesejahteraan sosial dan melaksanakan sebagian kewenangan dekonsentrasi yang dilimpahkan kepada Gubernur serta Tugas Pembantuan.

Pembangunan bidang Kesejahteraan Sosial di lingkungan Pemerintah Provinsi Jawa Barat secara instansional dilaksanakan oleh Dinas Sosial Provinsi Jawa Barat yang keberadaan kelembagaannya dikukuhkan dengan Peraturan Daerah nomor Nomor 5 tahun 2002 yo. Nomor 15 tahun 2000 tentang Dinas Daerah Provinsi Jawa Barat (Lembaran Daerah tahun 2000 Nomor 20 Seri D) dan Keputusan Gubernur Jawa Barat nomor 49 tahun 2003, tentang Pembentukan Instalasi Unit Pelaksana Teknis Dinas pada Dinas Sosial Provinsi Jawa Barat.

Dalam penyelenggaraan tugas pokok tersebut Dinas Sosial Provinsi Jawa Barat mempunyai fungsi :

1. Perumusan kebijakan operasional bidang kesejahteraan sosial.
2. Penyelenggaraan pelayanan umum bidang kesejahteraan sosial.
3. Pembinaan dan fasilitas pelaksanaan tugas – tugas bidang kesejahteraan sosial meliputi program, pemberdayaan partisipasi sosial masyarakat, pengembangan sosial, pemulihan sosial, bantuan, dan perlindungan sosial serta UPTD.
4. Penyelenggaraan ketatausahaan Dinas Sosial.

Dinas Sosial Provinsi Jawa Barat mempunyai Visi dan Misi sebagai berikut :

A. Visi

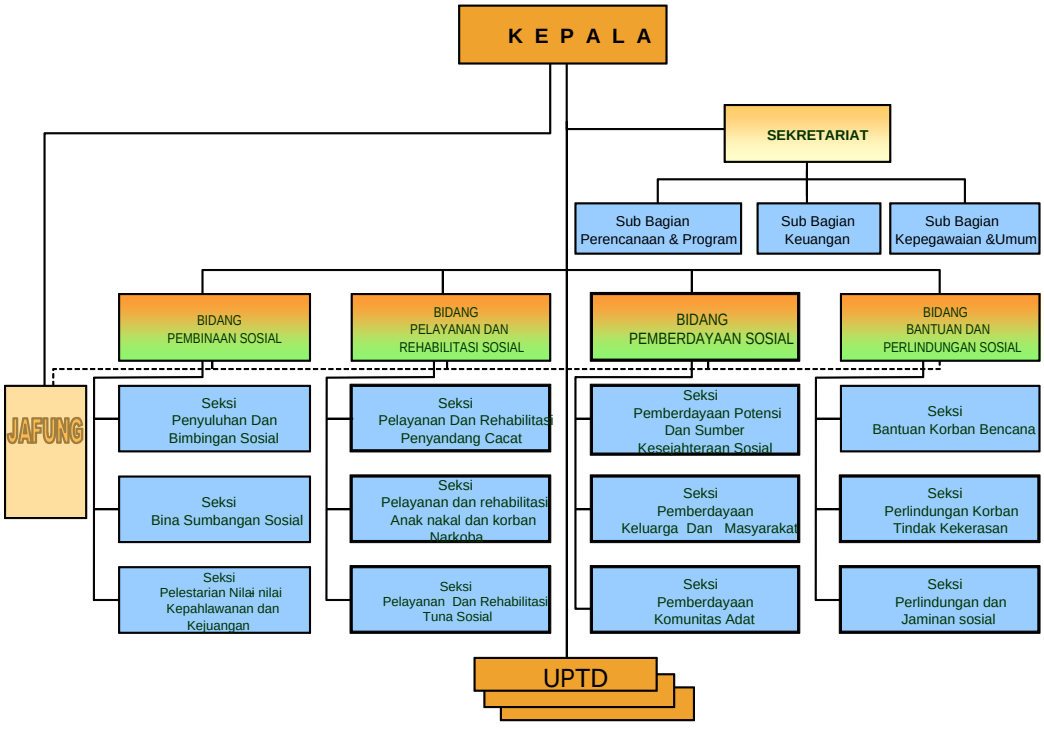
“TERPENUHINYA KESEJAHTERAAN SOSIAL MASYARAKAT JAWA BARAT TAHUN 2013”.

B. Misi

1. Meningkatkan pelayanan sosial bagi Penyandang Masalah Kesejahteraan Sosial (PMKS).
2. Meningkatkan peran serta dan kepedulian masyarakat terhadap penyelenggaraan pelayanan sosial dasar.
3. Meningkatkan sumber daya manusia yang profesional dalam bidang kesejahteraan sosial.
4. Meningkatkan fasilitasi dan koordinasi pembangunan kesejahteraan sosial.
5. Meningkatkan keterjangkauan dan mutu pelayanan sosial.
6. Melestarikan Nilai-nilai Keperintisan, Kepahlawanan dan Kejuangan.
7. Meningkatkan upaya pengurangan resiko bencana.

2.1.1 Struktur Organisasi Dinas Sosial Provinsi Jawa Barat

Struktur organisasi Dinas Sosial Provinsi Jawa Barat dapat dilihat pada gambar 2.1 berikut.



Gambar 2.1 Struktur Organisasi Dinas Sosial Provinsi Jawa Barat

2.2 Bencana

Bencana sering diidentikan dengan sesuatu yang buruk. Paralel dengan istilah *disaster* dalam bahasa Inggris. Bencana adalah sesuatu yang tak terpisahkan dalam sejarah manusia. Manusia bergumul dan terus bergumul agar bebas dari bencana (*free from disaster*). Dalam pergumulan itu, lahirlah praktek mitigasi, seperti mitigasi banjir, mitigasi kekeringan (*drought mitigation*), dan lain-lain. Di Mesir, praktek mitigasi kekeringan sudah berusia lebih dari 4000 tahun. Konsep tentang sistem peringatan dini untuk kelaparan (*famine*) dan kesiap-siagaan (*preparedness*) dengan lumbung raksasa yang disiapkan selama tujuh tahun pertama kelimpahan dan digunakan selama tujuh tahun kekeringan sudah lahir pada tahun 2000 BC, sesuai keterangan kitab Kejadian, dan tulisan-tulisan Yahudi Kuno (Nakmofa, 2007).

2.2.1 Jenis Bencana

Begitu luas wilayah Indonesia dan beraneka ragam penduduk serta letak geografis yang cukup rawan, dengan keadaan demikian di Indonesia tidak hanya terjadi bencana alam saja tapi juga ada bencana-bencana seperti bencana social, teknologi. Untuk lebih jelasnya kita bahas satu per satu bencana yang sudah dan mungkin akan terjadi di Indonesia.

2.2.1.1 Bencana Alam

Bencana alam adalah sebuah bencana yang diakibatkan oleh peristiwa atau serangkaian peristiwa yang disebabkan oleh alam antara lain berupa gempa bumi, tsunami, gunung meletus, banjir, kekeringan, angin topan, dan tanah langsor. Bencana ala mini terjadi secara tiba-tiba dan cukup sulit untuk diprediksi.

2.2.1.2 Bencana Non Alam

Bencana non-alam adalah bencana yang diakibatkan oleh peristiwa atau rangkaian peristiwa non-alam yang antara lain berupa gagal teknologi, gagal modernisasi, epidemi, dan wabah penyakit. Peristiwa ini sering timbul yang diakibatkan oleh manusia. Bencana Teknologi adalah semua kejadian bencana yang diakibatkan oleh kesalahan desain, pengoperasian, kelalaian dan kesengajaan

manusia dalam penggunaan teknologi dan atau industri. Bencana Sosial adalah suatu kondisi dimana terjadi huru-hara (kerusuhan) atau perang atau keadaan yang tidak aman di suatu daerah tertentu yang melibatkan lapisan masyarakat, golongan, suku, ataupun organisasi tertentu.

2.3 Konsep Dasar Sistem

Pengertian dari sistem menurut **Jogiyanto HM** adalah sebagai berikut :

"Sistem adalah kumpulan/group dari sub sistem/bagian/komponen apapun baik phisik maupun non phisik yang saling berhubungan satu sama lain dan bekerja sama secara harmonis untuk mencapai tujuan tertentu".

Sedangkan pandangan sistem yang lebih menekankan pada elemen atau komponen menurut **Jogiyanto HM** didefinisikan sebagai berikut :

"Sistem adalah kumpulan dari sub-sub sistem yang saling berinteraksi, saling bekerja sama dalam satu kesatuan kerja untuk mencapai suatu tujuan tertentu".

2.3.1 Karakteristik Sistem

Suatu sistem mempunyai karakterisitik atau atau sifat-sifat tertentu, yaitu:

1. Komponen-komponen (*Components*)

   Suatu sistem terdiri dari sejumlah komponen yang sering disebut dengan subsistem yang saling berinteraksi, artinya saling bekerja sama membentuk satu kesatuan. Subsistem itu sendiri yang mempunyai sifat-sifat dari sistem itu sendiri dalam menjalankan suatu fungsi dan mempengaruhi proses sistem secara keseluruhan.

2. Batas sistem (*Boundary*)

   Batas sistem merupakan daerah yang membatasi antara suatu sistem dengan sistem yang lainnya atau yang membatasi sistem tersebut dengan lingkungan luarnya sehingga sistem itu dapat dipandang sebagai satu kesatuan. Batasan sistem dapat merupakan ruang lingkup sistem itu sendiri.

3. Lingkungan luar sistem (*Environments*)

   Lingkungan luar yaitu segala sesuatu yang ada diluar sistem dari batasan sistem yang bisa mempengaruhi operasi sistem. Lingkungan luar sistem dapat bersifat menguntungkan juga dapat bersifat merugikan sistem.

4. Penghubung sistem (*Interface*)

   Penghubung merupakan media penghubung antara komponen sistem sehingga sumber-sumber data mengalir diantara komponen sistem dan membentuk satu kesatuan.

5. Masukan sistem (*Input*)

   Masukan yaitu energi yang dimasukan kedalam sistem.

6. Keluaran sistem (*Output*)

   Keluaran yaitu hasil dari energi yang diolah.

7. Pengolah sistem (*Process*)

   Proses yaitu yang akan mengubah masukan menjadi keluaran.

8. Sasaran sistem (*Objective*) atau tujuan (*Goal*)

   Sistem berguna jika mempunyai sasaran atau tujuan, dan sistem dikatakan berhasil jika keluaran – keluaran yang dihasilkan sistem dapat mencapai atau mengenai sasaran atau tujuan dari sistem itu sendiri.

2.4 Konsep Dasar Informasi

Pengertian informasi menurut **Jogiyanto HM** menyebutkan bahwa :

"Informasi adalah data yang diolah menjadi bentuk yang lebih berguna dan berarti bagi yang menerimanya".

2.4.1 Kualitas Informasi

Informasi yang baik adalah informasi yang berkualitas. Informasi yang berkualitas ditentukan oleh hal-hal sebagai berikut :

a. Relevan

   Informasi harus memberikan manfaat yang baik untuk pemakai informasi tersebut.

b. Dapat dipercaya (*realibility*)

Informasi yang akan diberikan dapat dipercaya kebenarannya dan mempunyai data-data yang lengkap dan jelas sumber-sumber datanya.

c. Tepat waktu (*timelines*)

Informasi yang dihasilkan atau dibutuhkan tidak boleh terlambat, karena nantinya tidak mempunyai nilai yang baik, sehingga apabila dijadikan dasar dalam pengambilan keputusan akan berakibat fatal atau kesalahan pengambilan keputusan dan tindakan.

d. Akurat

Informasi harus bebas dari kesalahan-kesalahan dan tidak menyesatkan. Informasi harus jelas mencerminkan maksudnya. Komponen-komponen data yang akurat adalah sebagai berikut :

a. *Completeness*, yaitu informasi yang dihasilkan atau yang dibutuhkan memiliki kelengkapan yang baik, karena bila informasi yang dihasilkan sebagian-sebagian tentunya akan mempengaruhi dalam pengambilan keputusan atau menentukan tindakan secara keseluruhan.
b. *Correctness*, yaitu kebenaran informasi dapat dipertanggungjawabkan dan mempunyai bukti-bukti dan fakta yang kuat.
c. *Security* atau Keamanan, dalam hal ini informasi yang dikirimkan ke setiap orang yang membutuhkannya perlu pengawasan karena struktur pengecekan dapat memutuskan jika informasi yang sensitif ditujukan kepada pemakai yang tidak sah kepada pihak yang salah.

e. Ekonomis

Biaya pembuatan informasi murah dan memberikan manfaat yang besar bagi pemakai.

2.5 Konsep Dasar Sistem Informasi

Sistem informasi menurut **Jogiyanto HM** mendefinisikan bahwa :

"Sistem informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolah transaksi harian, mendukung operasi, bersifat manajerial dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan".

Komponen-komponen sistem informasi adalah sebagai berikut :

1. Perangkat keras (*Hardware*)

   Perangkat keras (*Hardware*) adalah komputer itu sendiri.

2. Perangkat lunak (*Software*)

   Perangkat lunak berupa program-program aplikasi yang akan digunakan, yaitu merupakan kumpulan dari printah atau fungsi yang ditulis dengan aturan tertentu untuk memerintahkan komputer melaksanakan tugas tertentu.

3. Data

   Data merupakan komponen dasar dari informasi yang akan diproses lebih lanjut untuk menghasilkan informasi.

4. Prosedur

   Prosedur merupakan dokumentasi prosedur atau proses sistem, tata cara atau penuntun operasional (aplikasi) dan teknis.

5. Manusia

   Manusia adalah pengguna dari sistem informasi.

2.6 Basis Data

Basis data (*database*), atau sering pula dieja 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. Perangkat lunak yang digunakan untuk mengelola dan memanggil kueri (*query*) basis data disebut sistem manajemen basis data (*database management system*, DBMS). Sistem basis data dipelajari dalam ilmu informasi. Istilah "basis data" berawal dari ilmu komputer. Meskipun kemudian artinya semakin luas, memasukkan hal-hal di luar bidang elektronika, artikel ini mengenai basis data komputer. Catatan yang mirip dengan basis data sebenarnya sudah ada sebelum revolusi industri yaitu dalam bentuk buku besar, kuitansi dan kumpulan data yang berhubungan dengan bisnis.

Konsep dasar dari basis data adalah kumpulan dari catatan-catatan, atau potongan dari pengetahuan. Sebuah basis data memiliki penjelasan terstruktur dari

jenis fakta yang tersimpan di dalamnya: penjelasan ini disebut skema. Skema menggambarkan obyek yang diwakili suatu basis data, dan hubungan di antara obyek tersebut. Ada banyak cara untuk mengorganisasi skema, atau memodelkan struktur basis data: ini dikenal sebagai model basis data atau model data. Model yang umum digunakan sekarang adalah model relasional, yang menurut istilah layman mewakili semua informasi dalam bentuk tabel-tabel yang saling berhubungan dimana setiap tabel terdiri dari baris dan kolom (definisi yang sebenarnya menggunakan terminologi matematika). Dalam model ini, hubungan antar tabel diwakili denga menggunakan nilai yang sama antar tabel. Model yang lain seperti model hierarkis dan model jaringan menggunakan cara yang lebih eksplisit untuk mewakili hubungan antar tabel. Istilah *basis data* mengacu pada koleksi dari data-data yang saling berhubungan, dan perangkat lunaknya seharusnya mengacu sebagai *sistem manajemen basis data* (*database management system/DBMS*). Jika konteksnya sudah jelas, banyak *administrator* dan *programmer* menggunakan istilah basis data untuk kedua arti tersebut.

DBMS (*Database Management System*) adalah *software* yang menangani semua akses ke basis data. Secara konsep apa yang terjadi adalah sebagai berikut :

1. *User* melakukan pengaksesan basis data untuk informasi yang diperlukannya menggunakan suatu bahasa manipulasi data, biasanya disebut SQL.
2. DBMS menerima *request* dari user dan menganalisa request tersebut
3. DBMS memeriksa skema *eksternal user*, pemetaan eksternal/konseptual, skema konseptual, pemetaan konseptual/internal, dan struktur penyimpanan. DBMS mengeksekusi operasi-operasi yang diperlukan untuk memenuhi permintaan *user*.

2.6.1 Bahasa Basis Data

DBMS merupakan perantara bagi pemakai dengan basis data dalam disk. Cara berkomunikasi/berinteraksi antara pemakai dengan basis data tersebut diatur dalam suatu bahasa khusus yang ditetapkan oleh perusahaan pembuat *DBMS*. Bahasa tersebut dapat disebut sebagai bahasa basis data yang terdiri atas sejumlah

perintah yang diformulasikan oleh *user* dan diproses oleh *DBMS* untuk melakukan suatu aksi atau pekerjaan tertentu.

Ada 3 bahasa yang digunakan dalam basis data yaitu :

1. DDL (*Data Definition Language*)

   Merupakan bahasa definisi data yang digunakan untuk membuat dan *memanage* objek *database* seperti *database*, tabel dan *view*.

2. DML (*Data Manipulation Language*)

   Merupakan bahasa manipulasi data yang digunakan untuk memanipulasi data pada objek *database* seperti tabel.

3. DCL (*Data Control Language*)

   Merupakan bahasa yang digunakan untuk mengendalikan pengaksesan data.

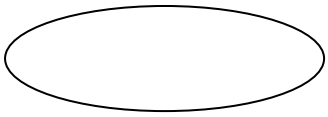
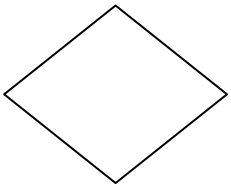
2.6.2 Kamus Data

Kamus data merupakan alat yang digunakan untuk mendeskripsikan isi dan semantik data dan seluruh aliran informasi dalam sistem. Kamus data terdiri dari nama data, sumber atau tujuan data, deskripsi, bentuk data, periode, dan struktur data. Tujuannya adalah untuk memberikan informasi mengenai definisi, struktur, pemakai dari masing-masing elemen. Dan elemen merupakan unit terkecil.

2.7 *Entity Relationship Diagram* (ERD)

ERD merupakan suatu model perancangan basis data yang terdiri dari sekumpulan objek dan relasi antar objek tersebut, serta dapat digunakan untuk menggambarkan relasi antara dua entitas atau lebih. Komponen – komponen yang digunakan dalam membangun ERD dapat dilihat pada table 2.1 berikut.

Tabel 2.1 Simbol – Simbol ERD

No	Simbol	Keterangan
1		*Entitas* Merupakan kumpulan suatu objek sejenis yang keberadaanya dapat dibedakan secara unik dari objek.
2		*Atribut* *Atribut* adalah unsur atau elemen yang ada dalam satu entity atau biasa disebut sebagai *data element, data field, data item*.
3		*Relasi* Relasi adalah suatu hubungan yang terjadi antara entitas yang satu dengan entitas yang lain.
4		*Link* *Link* adalah suatu penghubung antara himpunan relasi dengan himpunan entitas dan himpunan entitas dengan atributnya

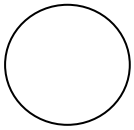
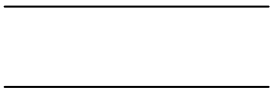
2.8 Diagram Konteks

Diagram konteks adalah pemodelan yang menggambarkan sistem yang berbasis komputer secara umum yang kemudian akan dirancang dan akan dibuat menjadi program. Tujuan pembuatan diagram konteks untuk memperlihatkan sebuah proses yang berinteraksi dengan lingkungannya. Pada Diagram Konteks akan terlihat bagaimana arus data yang masuk dan bagaimana arus data keluar dari sistem yang selalu berhubungan dengan entitas luar yang mempengaruhi sistem.

2.9 *Data Flow Diagram* (DFD)

Data Flow Diagram adalah representasi grafik yang menggambarkan arus data dari suatu sistem. *Data Flow Diagram* menggambarkan komponen – komponen tersebut, asal dan tujuan dan penyimpanan data. Dan merupakan salah satu alat dalam perancangan sistem yang menggunakan simbol-simbol untuk menggambarkan aliran data melalui serangkaian proses yang saling berhubungan. DFD merupakan penurunan atau penjabaran dari diagram konteks. Simbol – simbol yang digunakan pada pembuatan DFD dapat dilihat pada tabel 2.2

Tabel 2.2 Simbol - Simbol DFD

No	Simbol	Keterangan
1		*Entity External* Merupakan kesatuan di lingkungan luar sistem yang dapat berupa orang, organisasi, atau sistem lain. Lingkungan luar dan batas sistem dinotasikan dalam simbol yang sama.
2		Arus data (*Data Flow)* Arus data (*Data Flow*) di *DFD* diberi simbol suatu panah. Arus data ini mengalir diantara proses (*process*).
3		*Process* Proses adalah kegiatan yang dilakukan oleh manusia, mesin, atau komputer yang bertujuan mengolah data yang dimasukkan menjadi keluaran yang diinginkan.
4		Simpanan data Simpanan data dapat berupa file, database, arsip, table atau buku.

2.10 Internet

Internet merupakan jaringan global komputer dunia, besar dan sangat luas sekali dimana setiap komputer saling terhubung satu sama lainnya dari negara ke negara lainnya di seluruh dunia dan berisi berbagai macam informasi, mulai dari text, gambar, audio, video, dan lainnya yang terhubung melalui protokol TCP/IP. Dalam mengatur integrasi dan komunikasi jaringan komputer ini menggunakan protokol yaitu TCP/IP**.** TCP (Transmission Control Protocol) bertugas untuk memastikan bahwa semua hubungan bekerja dengan benar, sedangkan IP (Internet Protocol) yang mentransmisikan data dari satu komputer ke komputer lain. TPC/IP secara umum berfungsi memilih rute terbaik transmisi data, memilih rute alternatif jika suatu rute tidak dapat di gunakan, mengatur dan mengirimkan paket-paket pengiriman data.

2.11 World Wide Web (WWW)

Website atau situs juga dapat diartikan sebagai kumpulan halaman yang menampilkan informasi data teks, data gambar diam atau gerak, data animasi, suara, video dan atau gabungan dari semuanya, baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling terkait dimana masing-masing dihubungkan dengan jaringan-jaringan halaman (*hyperlink*) dokumen web yang ditampilkan dalam *browser* web.

2.11.1 Sejarah Web

Sejarah Web bermula Maret 1989 di European Laboratory for Particle Physics (lebih dikenal dengan nama CERN), di kota Geneva dekat perbatasan Perancis dan Swiss. CERN (*Consei European pour la Receherche Nuclarie*) merupakan suatu organisasi yang didirikan oleh 18 negara di Eropa, peneliti lainnya dari CERN mengusulkan suatu protokol sistem distribusi informasi di Internet yang memungkinkan para anggotanya yang tersebar di seluruh dunia saling membagi informasi dan bahkan untuk menampilkan informasi tersebut dalam bentuk grafik. Pada tahun yang sama CERN dan MIT kemudian mendirikan suatu konsorsium yang dinamakan World Wide Web Consortium (W3C) yang bertugas untuk membangun standar bagi Web.

2.11.2 Aplikasi Web

Aplikasi web pada awalnya dibangun hanya dengan menggunakan bahasa yang disebut dengan HTML dan protokol yang dinamakan *Hypertext Transfer Protocol* (HTTP). Pada perkembangan berikutnya sejumlah *script* dan objek dikembangkan untuk memperluas kemampuan HTML, antara lain *script* PHP dan *Active Server Pages* (ASP), sedangkan contoh berupa objek adalah Applet (Java). Aplikasi web sendiri dapat dibagi menjadi dua, yaitu :

1. Web statis, dibentuk dengan HTML saja. Kekurangan aplikasi ini hanya terletak pada keharusan untuk memelihara program secara terus menerus untuk megikuti perubahan yang terjadi.
2. Web dinamis, yang terdiri dari *server-side scripting* seperti *script* ASP dan PHP dan *client-side scripting* seperti Vbscript ataupun Javascript. Web dinamis mengatasi kekurangan web statis karena aplikasi web yang dinamis memanfaatkan kemampuan *server* web untuk mengakses sistem database.

2.12 *Geographic Information System*

Sistem informasi geografis (*Geographic Information System*, GIS) adalah sistem yang dapat digunakan untuk menangkap, menyimpan, menganalisis, serta mengelola data dan karakteristik yang berhubungan yang secara spasial mengambil referensi ke bumi. Lebih jauh, sistem ini dapat didefinisikan sebagai sistem komputer untuk memadukan, menyimpan, membagi, serta menampilkan informasi yang mengambil acuan geografis.

Teknologi GIS menggunakan informasi digital yang didapatkan dari metode pembuatan data *digital*. Metode pembuatan yang umum digunakan adalah *digitization*, yaitu peta cetak atau rencana survey yang ditransfer ke dalam bentuk media digital menggunakan program komputer (*Computer aided drafting*, CAD) serta kapabilitas *georeferencing*.

GIS dapat digunakan untuk menggambarkan karakteristik permukaan, *subsurface*, dan atmosfir dari titik – titik informasi secara dua dimensi atau tiga dimensi. Contoh, GIS dapat membuat peta *isopleth* atau garis kontur yang

mengindikasikan perbedaan curah hujan. GIS juga bisa mengenali dan menganalisas hubungan spasial yang ada antara data spasial yang tersimpan secara *digital*. Relasi topologi ini membuat pemodelan spasial dan analisa yang komplek dapat dilakukan. Relasi topologi yang dimodelkan dengan GIS dapat meliputi *adjacency*, *containment*, dan *proximity*. Dengan pemodelan topologi ini kita dapat mendeteksi keberadaan lokasi SPBU, pasar, atau pabrik yang letaknya dekat suatu area seperti persawahan, atau rawa-rawa.

Selain itu, fungsi GIS juga dapat digunakan untuk mensimulasikan rute material sepanjang jaringan linier. Variabel seperti kemiringan, batas kecepatan, dapat dimasukkan kedalam pemodelan jaringan supaya merepresentasikan aliran fenomena secara akurat. Pemodelan jaringan ini umumnya digunakan dalam perencanaan transportasi, pemodelan hidrologi, serta infrastruktur.

2.13 *Tools* Yang Digunakan Untuk Membangun Aplikasi

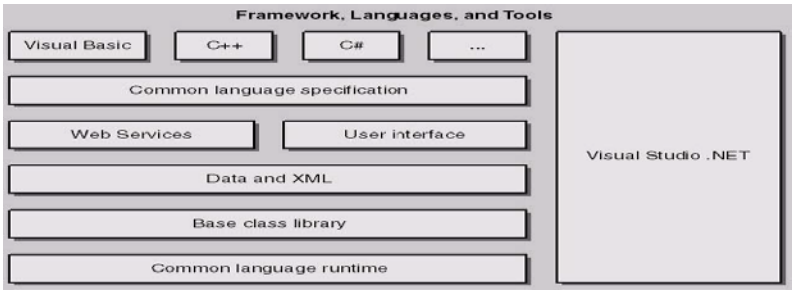
2.13.1 *.NET*

.NET Platform merupakan satu kumpulan teknologi yang dapat mempermudahkan *developer* untuk membuat aplikasi dalam lingkungan dan *device* yang beragam. Secara teknikal, *.NET Platform* menyediakan konsep pemrograman dengan *library* dan modul – modul baru yang konsisten, terlepas dari jenis bahasa pemrograman yang digunakan.

2.13.1.1 *Framework .NET*

Suatu komponen *windows* yang terintegrasi yang dibuat dengan tujuan untuk mendukung pengembangan berbagai macam jenis aplikasi serta untuk dapat menjalankan berbagai macam aplikasi generasi mendatang.

Gambar dari struktur *Framework* .NET dapat dilihat pada gambar 2.2 berikut.



Gambar 2.2 *Framework* .NET

2.13.1.2 Tujuan *Framework .NET*

Framework .NET bertujuan untuk:

1. Untuk menyediakan *environment* kerja didalam mengeksekusi kode yang dapat meminimalisasi proses *software deployment* dan menghindari konflik penggunaan versi software yang dibuat.
2. Untuk menyediakan *environment* kerja yang konsisten bagi bahasa pemrograman yang berorientasi objek (OOP) baik kode objek itu disimpan dan dieksekusi secara lokal, atau dieksekusi secara lokal tapi didistribusikan melalui internet atau dieksekusi secara *remote*.
3. Untuk menyediakan *environment* kerja yang aman dalam hal pengeksekusian kode, termasuk kode yang dibuat oleh pihak ketiga *(third party)*.
4. Membuat para *developer* lebih mudah mengembangkan berbagai macam jenis aplikasi yang lebih bervariasi, seperti aplikasi berbasis windows dan aplikasi berbasis web.
5. Membangun semua komunikasi yang ada didalam standar industri untuk memastikan bahwa semua kode aplikasi yang berbasis Framework .NET dapat berintegrasi dengan berbagai macam kode aplikasi lain.
6. Sebagai salah satu sarana untuk dapat memenuhi tujuan diatas, maka dibuatlah berbagai macam bahasa pemrograman yang dapat digunakan seperti bahasa C#, VB .NET, J#, Perl .NET , Phyton dan lain – lain.

2.13.1.3 Arsitektur *Framework .NET*

Framework .NET terdiri dari dua buah komponen utama, yaitu :

1. CLR *(Common Language Runtime)* merupakan pondasi utama dari *Framework .NET* . CLR merupakan komponen yang bertanggung jawab terhadap berbagai macam hal, seperti :
   a) Bertanggung jawab untuk melakukan manajemen memori.
   b) Melakukan eksekusi kode.
   c) Melakukan verifikasi terhadap keamanan kode.
   d) Menentukan hak akses dari kode.
   e) Melakukan kompilasi kode, dan berbagai layanan sistem lainnya.

   Dengan adanya fungsi CLR ini, maka aplikasi berbasis .NET biasa juga disebut dengan *managed code*. Sedangkan aplikasi di luar itu biasa disebut dengan *unmanaged code* CLR akan melakukan kompilasi kode-kode aplikasi kita menjadi bahasa *assembly* MSIL (*Microsoft Intermediate Language*) Proses kompilasi ini sendiri dilakukan oleh komponen yang bernama JIT *(Just In Time).* JIT hanya akan mengkompilasi metode – metode yang memang digunakan dalam aplikasi. Dan hasil kompilasi ini sendiri disimpan di dalam mesin dan akan dikompilasi kembali jika memang ada perubahan pada kode aplikasi kita.
2. *.NET Framework Class Library* atau kadang juga sering disebut dengan BCL *(Base Class Library).* Koleksi dari *reusable types* yang sangat terintegrasi secara melekat dengan CLR *Class library* bersifat berorientasi terhadap objek yang akan menyediakan tipe dari fungsi-fungsi *managed code*. Hal ini tidak hanya berpengaruh kepada kemudahan dalam hal penggunaan, tetapi juga dapat mengurangi waktu yang diperlukan pada saat eksekusi. Dengan sifat tersebut, maka komponen pihak ketiga akan dengan mudah diaplikasikan ke dalam aplikasi yang dibuat.

   Dengan adanya BCL ini, maka kita bisa menggunakan *Framework .NET* untuk membuat berbagai macam aplikasi, seperti :
   a) Aplikasi *console*.
   b) Aplikasi berbasis *windows (windows form).*
   c) Aplikasi *ASP .NET* (berbasis web).

d) Aplikasi *web services* XML.

e) Aplikasi berbasis *windows services*.

2.13.1.4 Keuntungan *Framework .NET*

Berikut beberapa keuntungan dari *Framework .NET* :

1. Mudah

   Yang dimaksud mudah di sini adalah kemudahan *developer* untuk membuat aplikasi yang dijalankan di *.NET Framework*. Mendukung lebih dari 20 bahasa pemrograman : *VB.NET*, *C#*, *J#*, *C++*, *Pascal*, *Phyton*, *PHP*.

2. Efisien

   Kemudahan pada saat proses pembuatan aplikasi, dan berimplikasi terhadap efisiensi dari suatu proses produktivitas, baik efisien dalam hal waktu pembuatan aplikasi atau juga efisien dalam hal lain, seperti biaya (*cost*).

3. Konsisten

   Kemudahan – kemudahan pada saat proses pembuatan aplikasi, juga bisa berimplikasi terhadap konsistensi pada aplikasi yang kita buat. Misalnya, dengan adanya BCL, maka kita bisa menggunakan objek atau *class* yang dibuat untuk aplikasi berbasis windows pada aplikasi berbasis web. Dengan adanya kode yang bisa diintegrasikan ke dalam berbagai macam aplikasi ini, maka konsistensi kode – kode aplikasi kita dapat terjaga.

4. Produktivitas

   Semua kemudahan di atas, pada akhirnya akan membuat produktivitas menjadi lebih baik. Produktivitas naik, terutama produktivitas para *developer* yang akan berdampak pada meningkatnya produktivitas suatu perusahaan.

2.13.2 Bahasa C#

C# (dibaca C-*Sharp*) merupakan bahasa pemrograman perusahaan *Microsoft* yang didesain dengan target diimplementasikan dalam teknologi *Framework .NET*. Pada tahun 2002, tepatnya bulan Januari, C# maupun

Framework .NET diselesaikan dan dapat diimplementasikan di kalangan industri. Bahasa pemrograman C# merupakan bahasa baru yang sangat handal dan konsisten serta membawa kesan bahasa pemrograman yang modern. C# pada dasarnya bukan bahasa sulit karena intinya mengambil dari berbagai bahasa pemrograman yang telah ada. Bahasa yang paling banyak digunakan dalam sintaksis C# adalah bahasa *Java*. Dimana komposisi bahasa C# adalah sebagai berikut:

1. 70 % *JAVA*
2. 10 % C++
3. 5 % *VISUAL BASIC*
4. 15 % BARU

Visual C# merupakan bagian dari teknologi Visual Studio .*NET* . Pada intinya, Visual C# merupakan bagian dari teknologi .*NET* yang dikembangkan untuk memenuhi kebutuhan industri akan keberadaan teknologi IT modern, yang mampu mengatasi sistem informasi.

2.13.3 ASP .NET

ASP adalah singkatan dari *Active Server Pages* yang merupakan salah satu bahasa pemograman web untuk menciptakan halaman web yang dinamis. ASP merupakan salah satu produk teknologi yang disediakan oleh Microsoft. ASP diproses melalui web server dan hasil proses ini menghasilkan HTML yang akan dikirimkan melalui browser. ASP bekerja pada web server dan merupakan *server side scripting*. ASP .NET memiliki keunggulan jika dibandingkan dengan model pemrograman web sebelumnya, antara lain adalah :

1. Performansi yang lebih baik. ASP .NET adalah suatu program hasil kompilasi *Common Language Runtime* (CLR) yang dijalankan di server, tidak seperti ASP yang lebih dikategorikan sebagai bahasa interpreter, ASP .NET memiliki keunggulan seperti aplikasi lain yang berjalan dalam CLR, yaitu *early binding, just in time compilation*, optimasi program dan proses *caching* yang lebih baik. Hal ini meningkatkan performansi yang jauh lebih baik.

2. Dukungan alat bantu yang lengkap. *Framework* ASP .NET dilengkapi dengan alat bantu dan *designer* yang lengkap, dalam membuat aplikasi web dengan ASP .NET menggunakan konsep WYSIWYG (*What You See Is What You Get*) yang berarti mendesain halaman web dengan menggunakan *control*, *drag and drop control* dan proses *deployment* aplikasi yang sederhana.
3. Fleksibilitas. Karena ASP .NET berbasiskan *common language runtime*, fleksibilitas dan kemampuan yang dimiliki platform .NET akan tersedia pula bagi *developer* dalam membangun aplikasinya.
4. Konsep *code behind*. ASP .NET memungkinkan kita untuk membuat *user interface* yang terpisah dengan program aplikasi.
5. Kemudahan administrasi. ASP .NET memiliki sistem konfigurasi yang dapat digunakan untuk mengkonfigurasikan aplikasi web dan server. Informasi ini disimpan dalam bentuk teks sehingga memudahkan dalam administrasi.
6. *Security* ASP .NET selain dapat diintegrasikan dengan *windows authentication* juga dapat dikonfigurasi per aplikasi, hal ini menjamin aplikasi yang dibuat, aman.

2.13.4 Visual Studio Web Development 2010 Ultimate

Microsoft Visual Studio merupakan sebuah perangkat lunak lengkap (*suite*) yang dapat digunakan untuk melakukan pengembangan aplikasi, baik itu aplikasi bisnis, aplikasi personal, ataupun komponen aplikasinya, dalam bentuk aplikasi console, aplikasi Windows, ataupun aplikasi Web. Visual Studio mencakup kompiler, SDK, *Integrated Development Environment* (IDE), dan dokumentasi (umumnya berupa MSDN *Library*). Kompiler yang dimasukkan ke dalam paket Visual Studio antara lain Visual C++, Visual C#, Visual Basic, Visual Basic .NET, Visual InterDev, Visual J++, Visual J#, Visual FoxPro, dan Visual SourceSafe.

Microsoft Visual Studio dapat digunakan untuk mengembangkan aplikasi dalam *native code* (dalam bentuk bahasa mesin yang berjalan di atas *Windows*)

ataupun *managed code* (dalam bentuk *Microsoft Intermediate Language* di atas *.NET Framework*). Selain itu, Visual Studio juga dapat digunakan untuk mengembangkan aplikasi *Silverlight*.

Kali ini penulis menggunakan Visual Studio 2010 Ultimate, sebagai perangkat lunak untuk mengembangkan aplikasi ini.

2.13.5 SQL *Server* 2008

SQL *Server* 2008 adalah salah satu produk terkini *Microsoft* yang menyajikan bukti nyata visi *Microsoft* dalam menyediakan *platform* basis data yang mendukung aplikasi dengan kebutuhan kualitas dan integritas data yang sangat kritis sekalipun dengan memperkecil biaya infrastruktur dan memungkinkan semua pengguna mengakses informasi secara lebih efesien. *Platform* ini menyediakan berbagai fitur yang memfokuskan pada:

1. Kepercayaan, memungkinkan organisasi untuk menjalankan aplikasi dengan kebutuhan data yang kritis dengan dukungan tingkat keamanan, kehandalan, dan skalabilitas yang tinggi.
2. Produktivitas, memungkinkan organisasi untuk mengurangi waktu dan biaya yang dibutuhkan untuk mengelola infrastruktur data.
3. Kualitas, memungkinkan penyampaian informasi yang sesuai dengan kondisi dan kebutuhan pengguna.

2.13.6 LINQ

Bahasa LINQ *(Integrated Query)* adalah komponen yang menambahkan kemampuan *query* data pada pemrograman *.NET*. Teknologi ini pertama kali dikenalkan oleh Anders Hejlsberg dalam *Microsoft Professional Developers Conference* (PDC) 2005.

Secara singkat LINQ dapat didefinisikan sebagai sebuah metode yang memudahkan dan menseragamkan cara pengaksesan data. Sehingga dengan menggunakan LINQ *developer* hanya perlu menggunakan sebuah teknik pengaksesan data saja. Karena jika kita lihat dunia pemrogramman saat ini maka umumnya *developer* perlu menguasai kemampuan yang berbeda-beda untuk setiap media penyimpanan data.

2.13.7 *Silverlight*

Microsoft *Silverlight* adalah sebuah kerangka aplikasi untuk menulis dan menjalankan aplikasi bisnis dan media *(rich application internet)* dengan fitur dan tujuan yang sama dengan Adobe Flash. Lingkungan *run-time* untuk *Silverlight* tersedia sebagai plug-in untuk browser web yang paling baru. Sementara versi awal *Silverlight* terfokus pada media *streaming*, versi saat ini mendukung multimedia, grafik dan animasi, dan memberikan dukungan pengembang untuk bahasa CLI dan alat – alat pembangunan.

2.13.8 *Sharpmap*

Sharpmap adalah *mapping library* yang mudah digunakan dalam aplikasi web maupun desktop. *Sharpmap* dapat digunakan untuk mengakses berbagai macam data GIS, memungkinkan *query* data spasial, dan merender peta dengan bagus. *Engine* dibuat dalam bahasa C# dan bebasiskan .NET *Framework* 2.0.