BAB 2
TINJAUAN PUSTAKA

2.1 Tinjauan Perusahaan

2.1.1. Sejarah Instansi

Mengacu Kepada Peraturan Gubernur Jawa Barat Nomor 40 Tahun 2009 tentang Tugas Pokok. Fungsi, Rincian Tugas dan Tata Kerja Dinas Perhubungan Provinsi Jawa Barat disebutkan dalam Pasal 2 Bab II bahwa tugas pokok dinas dalam hal ini Dinas Perhubungan adalah mempunyai tugas pokok melaksanakan urusan pemerintahan daerah bidang perhubungan berdasarkan asas otonomi, asas dekonsentrasi, dan tugas pembantuan.

Dasar hukum organisasi Dinas Perhubungan Propinsi Jawa Barat dibentuk dengan Peraturan Daerah Provinsi Jawa Barat Nomor 21 Tahun 2008 tentang Organisasi dan Tata Kerja Dinas Daerah Provinsi Jawa Barat (Lembaran Daerah Tahun 2008 Nomor 20 Seri D, Tambahan Lembaran Daerah Nomor 55) sebagai perubahan tentang Dinas Daerah. Berdasarkan Peraturan Gubernur Provinsi Jawa Barat Nomor : 40 Tahun 2008 tentang Tugas Pokok, Fungsi , Rincian Tugas Unit dan Tata Kerja Dinas Perhubungan Provinsi Jawa Barat adalah melaksanakan urusan pemerintahan daerah berdasarkan azas otonomi dan tugas pembantuan urusan bidang perhubungan.

Dalam rancangan Peraturan Gubernur Jawa Barat tentang Tugas Pokok, Fungsi Rincian Tugas Unit dan Tata Kerja Dinas Perhubungan Provinsi Jawa Barat, Dinas Perhubungan mempunyai fungsi:

1. perumusan dan penetapan kebijakan teknis urusan bidang transportasi darat, tranportasi laut dan ASDP, transportasi udara, bina sistem operasional transportasi;

2. penyelenggaraan bidang urusan perhubungan meliputi transportasi darat, transportasi laut dan ASDP, transportasi udara, bina sistem operasional transportasi;
3. pembinaan dan pelaksanaan tugas-tugas perhubungan meliputi transportasi darat, tranportasi laut dan ASDP, transportasi udara, bina sistem operasional transportasi;
4. pengkoordinasian dan pembinaan UPTD;
5. pelaksanaan tugas lain dari Gubernur sesuai dengan tugas dan fungsinya.

Sebagai landasan dalam pelaksanaan Tugas Pokok dan Fungsi, Dinas juga telah dilengkapi dengan peraturan daerah yang memudahkan operasionalisasi pelaksanaan tugas dan pelayanan masyarakat melalui :

1. Peraturan Daerah Provinsi Jawa Barat Nomor 21 Tahun 2001 tentang Penyelenggaraan Perhubungan, dan
2. Peraturan Daerah Provinsi Jawa Barat Nomor 22 Tahun 2002 tentang Retribusi Penyelenggaraan Perhubungan.

Kedua peraturan daerah tersebut saat ini sedang dalam proses revisi, yang pada tahun anggaran 2008 telah dilaksanakan Kajian Akademis dan pada tahun anggaran 2009 telah dianggarkan untuk penyusunan Raperda revisi Peraturan Daerah Provinsi Jawa Barat nomor 21 Tahun 2001 dan Peraturan Daerah Provinsi Jawa Barat Nomor 22 Tahun 2002.

2.1.2. Logo Instansi

Logo Dinas Perhubungan



Gambar 2.1 Logo Dinas Perhubungan

Unsur logo tersebut terdiri dari :

1. Sayap tujuh helai disebelah kiri dan tujuh helai di sebelah kanan
2. Jangkar yang menyatu dengan sayap dan ekor
3. Bola dunia warna biru dengan garis-garis warna emas yang menyatu dengan roda gigi sebanyak 12 buah warna emas dan 12 buah warna biru
4. Ekor warna emas lima helai
5. Padi 45 butir dan kapas 17 buah seluruhnya berwarna emas yang tangkainya diikat dengan pita warna emas berbentuk angka delapan
6. Seloka "Wahana Manghayu Warga Pertiwi" diletakkan di dalam jangkar warna biru
7. Pita warna emas dan biru diletakkan dibawah lingkaran warna mas dengan tulisan Departemen Perhubungan Republik Indonesia

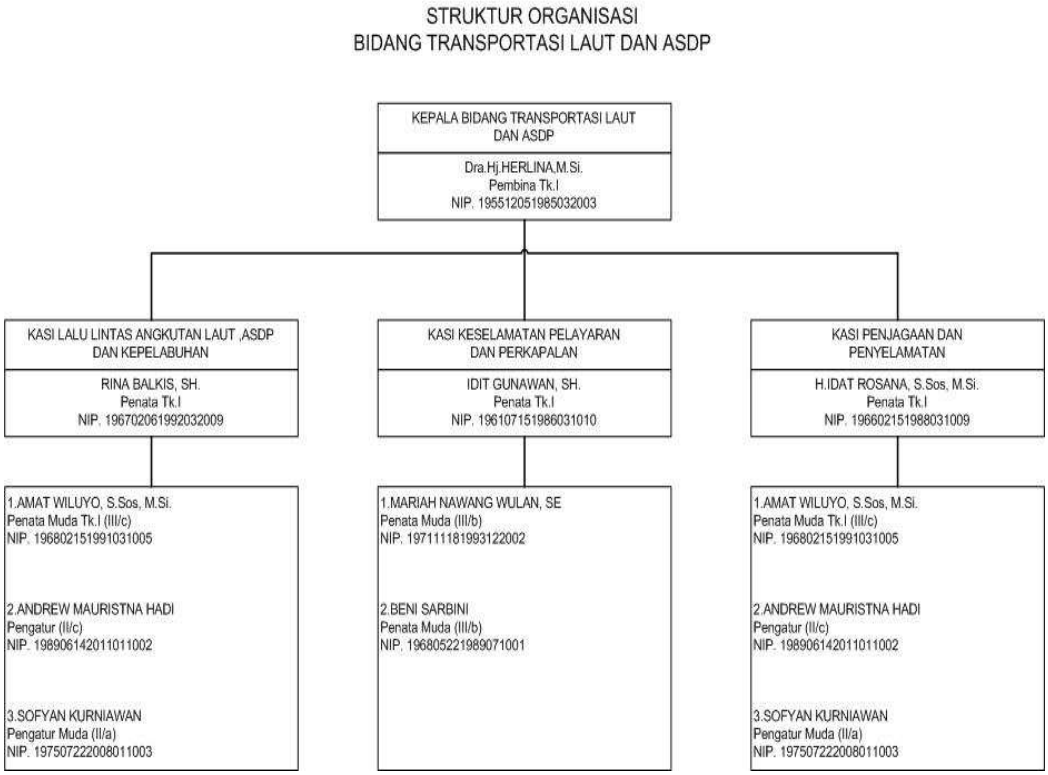
Arti unsur-unsur pada lambang ialah :

1. Burung merupakan simbolis sarana tercepat untuk mencapai sasaran dan jangkauan perhubungan juga melambangkan Perhubungan Udara
2. Jangkar merupakan sarana kokoh dan kuat menggambarkan missi perhubungan dapat menjangkau Kepulauan Nusantara maupun seluruh dunia dengan tabah dan tenang sekaligus melambangkan Perhubungan Laut

2.1.3. Struktur Organisasi dan *Job Description*

Struktur Organisasi dan *Job Description* menjelaskan struktur organisasi yang ada di dalam Dinas Perhubungan Jawa Barat khusus nya dan penjelasan pekerjaan yang dilakukan dari setiap bagiannya.

2.1.3.1. Struktur Organisasi



Gambar 2.2 Stuktur Organisasi Dinas Perhubungan Jawa barat bidang Transportasi Laut dan ASDP

2.1.3.2. Job Description

Adapun Job Description di bidang Transportasi Laut dan ASDP di dinas perhubungan jawa barat adalah sebagai berikut :

A. Kepala Bidang Transportasi Laut dan ASDP

Bidang Transportasi Laut dan ASDP mempunyai tugas pokok menyelenggarakan pengkajian bahan kebijakan operasional dan fasilitasi transportasi laut dan ASDP(Angkutan sungai, danau, dan penyebrangan).

Dalam menyelenggarakan tugas pokok sebagaimana dimaksud di atas, Bidang Transportasi Laut dan ASDP mempunyai fungsi :

1) pengkajian bahan kebijakan operasional transportasi laut dan ASDP;
2) pengkajian bahan fasilitasi transportasi laut dan ASDP;
3) penyelenggaraan fasilitasi bidang transportasi laut dan ASDP.

B. Analis Sarpras Angkutan Laut
   1) Menyusun Bahan Kebijakan Penetapan Lokasi Pelabuhan dan Terminal Khusus ;
   2) Menyusun Bahan Kebijakan Penetapan Rencana Induk Pelabuhan dan Terminal Khusus ;
   3) Menyusun Bahan Kebijakan Penetapan Daerah Lingkungan Kerja (DLKr) dan Daerah Lingkungan Pengawasan (DLKP) Pelabuhan dan Terminal Khusus ;

C. Analis Perijinan Usaha Angkutan Laut dan Analis Perijinan Pelabuhan Laut
   1) Menyusun Bahan Saran Pertimbangan Teknis PerijinanPembukaan Kantor Cabang Perusahaan Angkutan / Pelayaran Nasional yang berdomisili dan beroperasi pada Lintas Pelabuhan ;
   2) Menyusun bahan kebijakan teknis lalu lintas dan angkutan laut, ASDP dan Kepelabuhanan lainya

D. Penyusun Laporan
   1) Menyusun Program Kerja Seksi
   2) Menyusun bahan telaahan staf sebagai bahan pertimbangan pengambilan kebijakan
   3) Menyusun Laporan dan Evaluasi Keiatan Seksi Lalu Lintas dan Angkutan Laut, ASDP dan Kepelabuhanan
   4) Melaksanakan koordinasi dengan unit kerja terkait
   5) Melaksanakan tugas lain sesuai dengan tugas pokok dan fungsinya

E. Operator Komputer
   1) Mengetik konsep-konsep naskah dinas
   2) Mengerjakan tugas-tugas kedinasan lain
   3) Memelihara kinerja komputer agar tetap optimal

F. Penyusun Laporan Seksi dan Bendahara Pengeluaran Pembantu
   1) Menyusun Program Kerja Seksi
   2) Menyusun bahan kebijakan teknis pencatatan kapal dalam buku register pas kapal yang berlayar di perairan daratan
   3) Menyusun bahan telaahan staf sebagai bahan pertimbangan pengambilan kebijakan
   4) Menyusun laporan dan evaluasi kegiatan Seksi Keselamatan Pelayaran dan Perkapalan
   5) Melaksanakan koordinasi dengan unit kerja terkait
   6) Menyusun laporan keuangan

G. Analis Jaringan Angkutan Laut / ASDP dan Analis Keselamatan Pelayaran
   1) Menyusun bahan kebijakan penetapan jaringan angkutan SDP
   2) Menyusun bahan kebijakan penerbitan pas perairan daratan
   3) Menyusun bahan kebijakan penetapan tarif angkutan SDP kelas Ekonomi
   4) Menyusun bahan kebijakan pelaksanaan pengukuran kapal yang berlayar di perairan daratan
   5) Menyusun bahan kebijakan teknis pemeriksaan konstruksi, permesinan, perlengkapan kapal yang berlayar di perairan daratan

H. Pengawas Keselamatan

1) Menyusun bahan kebijakan teknis penegakan peraturan perundang-undangan bidang pelayaran ;
2) Menyusun bahan kebijakan prosedur pengawasan dan penjagaan serta penyelamatan di DLKr dan DLKp;
3) Menyusun bahan kebijakan prosedur tetap dan melaksanakan pertolongan musibah kecelakaan kapal di DLKr dan DLKp ;
4) Menyusun bahan kebijakan koordinasi pelaksanaan kegiatan penyelidikan kecelakaan kapal, bantuan pencarian dan pertolongan (SAR), penanggulangan pencemaran laut dan perairan daratan serta pengamanan wilayah kerja pelabuhan
5) Menyusun bahan kebijakan prosedur penyidikan tindak pidana di pelabuhan dan penegakan peraturan perundang-undangan pelayaran dan kepelabuhanan
6) Menyusun bahan kebijakan penyusunan pedoman pelaksanaan kegiatan pengawasan keselamatan dan perlengkapan kapal

I. Pengagenda

1) Mengagenda surat masuk/keluar Bidang dan mendistribusikannya ke Kabid/Seksi sesuai disposisi ;
2) Menggandakan naskah Dinas ;
3) Menatakelola naskah dinas yang masuk dan ke luar dari Bidang :
   a. Menyusun klasifikasi naskah dinas ;
   b. Menyusun Arsip naskah dinas secara cepat, tepat dan mudah ditemukan kembali

2.2. Landasan Teori

Landasan teori menjelaskan beberapa teori yang berkaitan dengan permasalahan yang dibahas sebagai dasar pemahaman dalam sebuah sistem serta metode yang dipakai untuk kegiatan pengembangan terhadap sistem sendiri.

2.2.1. Aplikasi

Perangkat lunak aplikasi adalah suatu subkelas perangkat lunak komputer yang memanfaatkan kemampuan komputer langsung untuk melakukan suatu tugas yang diinginkan pengguna. Biasanya dibandingkan dengan perangkat lunak sistem yang mengintegrasikan berbagai kemampuan komputer, tapi tidak secara langsung menerapkan kemampuan tersebut untuk mengerjakan suatu tugas yang menguntungkan pengguna. Contoh utama perangkat lunak aplikasi adalah pengolah kata, lembar kerja, dan pemutar media.

Beberapa aplikasi yang digabung bersama menjadi suatu paket kadang disebut sebagai suatu paket atau suite aplikasi (*application suite*). Contohnya adalah Microsoft Office dan Open Office.org, yang menggabungkan suatu aplikasi pengolah kata, lembar kerja, serta beberapa aplikasi lainnya. Aplikasi-aplikasi dalam suatu paket biasanya memiliki antarmuka pengguna yang memiliki kesamaan sehingga memudahkan pengguna untuk mempelajari dan menggunakan tiap aplikasi. Sering kali, mereka memiliki kemampuan untuk saling berinteraksi satu sama lain sehingga menguntungkan pengguna. Contohnya, suatu lembar kerja dapat dibenamkan dalam suatu dokumen pengolah kata walaupun dibuat pada aplikasi lembar kerja yang terpisah

2.2.2. Sistem Informasi Kepelabuhanan

2.2.2.1 Sistem

Sistem adalah sekumpulan unsur / elemen yang saling berkaitan dan saling mempengaruhi dalam melakukan kegiatan bersama untuk mencapai suatu tujuan.

Contoh :

Sistem Komputer terdiri dari Software, Hardware, dan Brainware

Menurut LUDWIG VON BARTALANFY Sistem merupakan seperangkat unsur yang saling terikat dalam suatu antar relasi diantara unsur-unsur tersebut dengan lingkungan.

Menurut ANATOL RAPOROT Sistem adalah suatu kumpulan kesatuan dan perangkat hubungan satu sama lain.

Menurut L. ACKOF Sistem adalah setiap kesatuan secara konseptual atau fisik yang terdiri dari bagian-bagian dalam keadaan saling tergantung satu sama lainnya.

Syarat-syarat sistem :

1. Sistem harus dibentuk untuk menyelesaikan masalah.

2. Elemen sistem harus mempunyai rencana yang ditetapkan.

3. Adanya hubungan diantara elemen sistem.

4. Unsur dasar dari proses (arus informasi, energi dan material) lebih penting dari pada elemen sistem.

5. Tujuan organisasi lebih penting dari pada tujuan elemen.

Analisis dan Perancangan Sistem

Secara garis besar, sistem dapat dibagi 2 :

A. SISTEM FISIK (PHYSICAL SYSTEM) :

Kumpulan elemen-elemen/ unsur-unsur yang saling berinteraksi satu sama lain secara fisik serta dapat diidentifikasikan secara nyata tujuan-tujuannya.

Contoh :

1. Sistem transportasi, elemen : petugas,mesin, organisasi yang menjalankan transportasi

2. Sistem Komputer, elemen : peralatan yang berfungsi bersama-sama untuk menjalankan pengolahan data.

B. SISTEM ABSTRAK (ABSTRACT SYSTEM) :

Sistem yang dibentuk akibat terselenggaranya ketergantungan ide, dan tidak dapat diidentifikasikan secara nyata, tetapi dapat diuraikan elemen-elemennya.

Contoh : Sistem Teologi, hubungan antara manusia dengan Tuhan

KARAKTERISTIK SISTEM

a. Organisasi

b. Interaksi

c. Interdependensi

d. Integrasi

e. Tujuan pokok

KLASIFIKASI SISTEM

A. DETERMINISTIK SISTEM

Sistem dimana operasi-operasi (input/output) yang terjadi didalamnya dapat ditentukan/ diketahui dengan pasti

Contoh :

1. Program komputer, melaksanakan secara tepat sesuai dengan rangkaian instruksinya.
2. Sistem penggajian.

B. PROBABILISTIK SISTEM

Sistem yang input dan prosesnya dapat didefinisikan, tetapi output yang dihasilkan tidak dapat ditentukan dengan pasti; (Selalu ada sedikit kesalahan/penyimpangan terhadap ramalan jalannya sistem).

Contoh :

1. Sistem penilaian ujian
2. Sistem pemasaran.

C. OPEN SISTEM

Sistem yang mengalami pertukaran energi, materi atau informasi dengan lingkungannya. Sistem ini cenderung memiliki sifat adaptasi, dapat menyesuaikan diri dengan lingkungannya sehingga dapat meneruskan eksistensinya.Contoh

Sistem keorganisasian memiliki kemampuan adaptasi.(Bisnis dalam menghadapi persaingan dari pasar yang berubah. Perusahaan yang tidak dapat menyesuaikan diri akan tersingkir).

2.2.2.2 Informasi

Informasi didapat kan dari sebuah sistem informasi atau disebut juga dengan *processing system*. sistem informasi dapat didefinisikan sebagai suatu sistem di dalam suatu organisasi yang merupakan kombinasi dari orang-orang, fasilitas, teknologi, media, prosedur-prosedur dan pengendalian yang ditunjukan untuk mendapatkan jalur komunikasi yang penting, memproses tipe transaksi rutin tertentu, memberi sinyal kepada manajemen dan yang lainnya terhadap kejadian-kejadian internal dan eksternal yang penting dan menyediakan suatu dasar informasi untuk pengambilan keputusan yang cerdik.

2.2.2.3 Sistem Informasi

Sistem informasi merupakan suatu kumpulan dari komponen-komponen dalam perusahaan atau organisasi yang berhubungan dengan proses penciptaan dan pengaliran informasi.

Sistem Informasi tidak harus melibatkan komputer. Sistem Informasi yang menggunakan komputer biasa disebut sistem informasi berbasis komputer (Computer-Based Information Systems atau CBIS).

Dalam prakteknya, istilah sistem informasi lebih sering dipakai tanpa berbasis komputer walaupun dalam kenyataannya komputer merupakan bagian yang penting. Yang dimaksudkan dengan sistem informasi disini adalah sistem informasi yang berbasis komputer.

Sistem Informasi (SI) atau Information System (IS) yang menunjukkan sistem yang dapat menghasilkan informasi yang berguna.

Definisi Sistem Informasi

Ada beragam definisi sistem informasi, sebagaimana tercantum di bawah ini.
Menurut Alter (1992)

Sistem informasi adalah kombinasi antara prosedur kerja, informasi, orang, dan teknologi informasi yang diorganisasikan untuk mencapai tujuan dalam sebuah organisasi.

Menurut Bodnar dan Hopwood (1993)

Sistem informasi adalah kumpulan perangkat keras dan perangkat lunak yang dirancang untuk mentransformasikan data ke dalam bentuk informasi yang berguna.

Menurut Gelinas, Oram, dan Wiggins (1990)

Sistem informasi adalah suatu sistem buatan manusia yang secara umum terdiri atas sekumpulan komponen berbasis komputer dan manual yang dibuat

untuk menghimpun, menyimpan, dan mengelola data serta menyediakan informasi keluaran kepada pemakai.

2.2.2.4 Kepelabuhanan

Kepelabuhanan yaitu segala sesuatu yang berkaitan dengan kegiatan penyelenggaraan pelabuhan dan kegiatan lainnya dalam melaksanakan fungsi pelabuhan untuk menunjang kelancaran, keamanan dan ketertiban arus lalu lintas kapal, penumpang dan/atau barang, keselamatan berlayar, tempat perpindahan intra dan/ atau antar moda serta mendorong perekonomian nasional dan daerah.

Dalam Bahasa Indonesia dikenal 2 istilah yang berhubungan dengan arti pelabuhan yaitu Bandar dan Pelabuhan. Bandar (harbor) adalah daerah pelabuhan yang terlindung terhadap gelombang dan angin untuk berlabuhnya kapal-kapal. Sementara pelabuhan adalah daerah perairan yang terlindung terhadap gelombang/ arus, sehingga kapal dapat berputar (Turning basin), bersandar / membuang sauh sehinnga bongkar muat atas barang dan perpindahan penumpang dapat dilaksanakan, guna mendukung fungsi-fungsi tersebut di babgun dermaga, jalan, gudang, fasilitas penerangan, telekomunikasi dan sebagainya sehingga fungsi pemindahan muatan dari/ ke kapal yang bersandar di pelabuhan menuju tujuan selanjutnya dapat dilakukan.

Selain itu berdasarkan beberapa referensi, pelabuhan dapat diartikan sebagai berikut :

Pelabuhan adalah wilayah yang terdiri atas daratan dan perairan dengan batas tertentu sebagai tempat kegiatan pemerintah dan kegiatan ekonomi yang di pergunakan sebagai tempat bersandar, berlabuh, naik-turunnya penumpang dan atau bongkar muat barang yang dilengkapi dengan fasilitas keselamatan pelayaran dan kegiatan penunjang serta sebagai tempat perpindahan intra dan antar moda (PP Nomor 69 Tahun 2001).

Pelabuhan (port) adalah daerah perairan yang terlindung terhadap gelombang, yang dilengkapi dengan fasilitas terminal laut meliputi dermaga

dimana kapal dapat bertambat untuk bongkar muat barang , gudang laut (transito) dan tempat-tempat penyimpanan dimana barang-barang dapat disimpan dalam waktu lebih lama selama menunggu pengiriman ke daerah tujuan (Triatmojo 1996).

Pelabuhan adalah sebagai tempat yang terlindung dari gerakan gelombang laut, sehingga bongkar muat dapat dilaksanakan demi menjamin keamanan barang (Kramadibrata 1935)

Pelabuhan adalah sebuah areal perairan tertutup untuk melindungi dan memberi keamanan bagi kapal dari angin kencang, tempat mengambil bahan makanan serta bahan bakar reparasi atau tempat transfer muatan.

Pelabuhan adalah merupakan suatu pintu gerbang dan memperlancar hubungan antar daerah pulau atau bahkan antar benuadan bangsa yang dapat memajukan daerah belakangnya (hinterland). Daerah belakang ini adalah yang memiliki kepentingan hubungan ekonomi, sosial, dan lain-lain dengan pelabuhan tersebut.

2.2.2.5 Sistem Informasi Kepelabuhanan

Sistem informasi kepelabuhanan adalah salah satu bentuk sistem dan laporan mengenai informasi produktifitas kepelabuhanan khususnya yang ada di wilayah pemerintahan provinsi jawa barat, di dalamnya terdapat proses input-output data mengenai kepelabuhan. Serta adanya laporan bulanan yang mudah di gunakan.

Adapun dalam sistem informasi kepelabuhanan ini terdapat 3 (tiga) aktifitas. Antara lain :

1. Bongkar dan Muat barang
2. Kedatangan dan Keberangkatan kapal
3. Angkutan penumpang pengguna jasa Transportasi kapal Laut

2.2.3. 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. Perangkat lunak yang digunakan untuk mengelola dan memanggilkueri (*query*) basis data disebut sistem manajemen basis data (*database management system*, DBMS). Sistem basis data dipelajari dalam ilmu informasi.

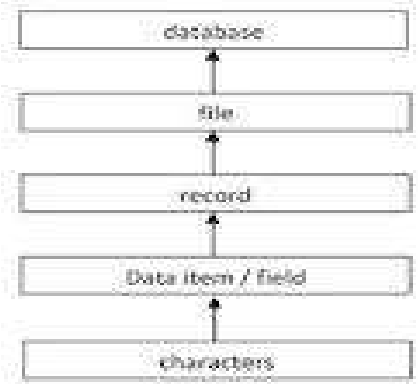
2.2.3.1.Pengertian Basis Data

Secara umum, basis data dapat diartikan sebagai kumpulan data yang tersimpan dalam suatu tabel yang saling berelasi.

Adapun kegunaan basis data adalah :

1. Basis data merupakan komponen penting dalam sistem informasi, karena merupakan dasar dalam menyediakan informasi.
2. Menentukan kualitas informasi yang akurat, tepat pada waktunya, dan relevan, dimana informasi dikatakan bernilai jika manfaatnya lebih efektif dibandingkan dengan biaya untuk mendapatkannya.
3. Mengurangi duplikasi data (*data redudancy*).
4. Meningkatkan relasi antar data (*data relatability*).
5. Mengurangi pemborosan *external storage*.

Di dalam basis data terdapat beberapa istilah berikut yang juga dikenal sebagai urutan jenjang data :



1. *Characters*

   Merupakan bagian data yang terkecil yaitu berupa karakter numerik, huruf, maupun karakter spesial (simbol) yang membentuk item data/ *field*.

Nim	Nama
1010xxx	A (character)

2. *Field*

   Merepresentasikan suatu atribut dari *record* yang menunjukan suatu item dari data, misalnya nama.

3. *Record*

   Merupakan kumpulan dari *field*, menggambarkan suatu unit data individu tertentu. Kumpulan *record* membentuk suatu *file*. Misalnya, *file* personalia, tiap-tiap *record* mewakili data tiap-tiap karyawan.

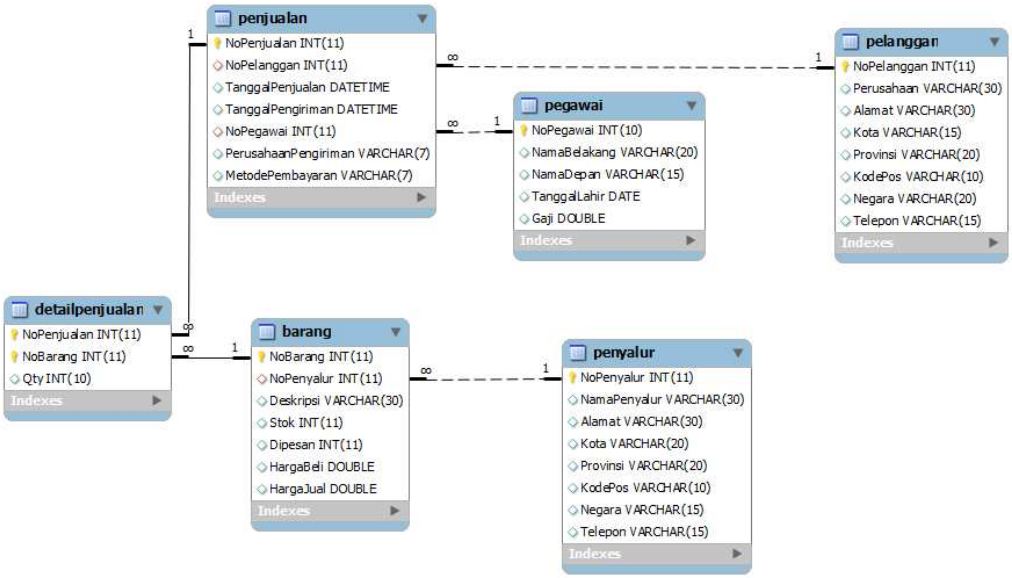
4. *File*

   File terdiri dari *record-record* yang menggambarkan satu kesatuan data yang sejenis. Misalnya, *file* mata kuliah berisi data tentang semua mata kuliah yang ada.



5. *Database*

   Merupakan kumpulan dari *File*/Tabel yang saling berelasi.



2.2.3.2. Normalisasi Basis Data

Normalisasi merupakan sebuah teknik dalam logika desain sebuah basis data, yaitu mengelompokkan atribut dari suatu relasi sehingga membentuk struktur relasi yang baik (tanpa redudansi).

1. Normalisasi ke-1

   Suatu data dikatakan normal pertama jika sudah memiliki karakter berikut:

   a. Mendefinisikan Atribut kunci (*key)*

   b. Tidak ada group yang berulang

   c. Semua atribut bukan *key* bergantung pada atribut kunci

2. Normalisasi ke-2

   Suatu data dikatakan normal kedua jika sudah memiliki karakter berikut:

   a. Sudah memenuhi dalam bentuk normal pertama

   b. Sudah tidak ada ketergantungan parsial, dimana seluruh *field* tergantung pada sebagian *keyfield*

2.2.3.3. Sistem Manajemen Basis Data

Sistem Manajemen Basis Data adalah perangkat lunak yang didesain untuk membantu dalam hal pemeliharaan dan utilitas kumpulan data dalam jumlah besar, serta menjadi alternatif penggunaan secara khusus untuk aplikasi, seperti penyimpanan data dalam *file* dan menulis kode aplikasi yang spesifik untuk pengaturannya. Adapun keuntungan dari DBMS (*Database Management System*) adalah :

1. Data *independence,* DBMS dapat mengurangi kebergantungan antar data.
2. *Efficient data access,* DBMS dapat menimpa data dan *meretrive* data dengan efisien. Apabila suatu saat data tersebut di butuhkan, maka efektif dalam pemanggilannya.
3. Data *integrity* dan *security,* saat data di akses menggunakan DBMS, DBMS mengintegrasi *constraints* dari tiap kata tersebut.
4. *Data administration*, Administrasi data digunakan untuk mengatur hal-hal yang berhubungan dengan distribusi data, hak akses user yang mendistribusikan atau menggunakan data tersebut, mengorganisasikan data untuk mengurangi redudansi data dan *fine-tuning* penyimpanan data sehingga *retrive* data berjalan dengan efisien.
5. *Concurrent access and crash recovery*, DBMS dapat menjadwalkan *concurrent access* (pengaksesan secara bersamaan) terhadap suatu data sehingga menghindari dari *crash*.
6. *Reduced application development time,* DBMS mendukung banyak fungsi penting yang diakses oleh banyak aplikasi yang tersimpan dalam media penyimpanan.

2.2.3.4. Sistem Manajemen Basis Data Relasional

RDBMS (*Relational Database Management System*) merupakan sekumpulan data yang saling beruhubungan yang disimpan sedemikian rupa sehingga mudah diambil informasinya bagi pengguna. Ada tiga prinsip dalam RDBMS :

1. *Data Definition*

Mendefinisikan data yang akan dibuat, seperti nama tabel dan *field* serta keterangan mengenai *field* tersebut.

2. *Data Manipulation*

Memanipulasi data yang sudah dibuat, seperti mengubah nama *field* dan keterangannya, menghapus *record*.

3. *Data Control*

Bagian ini berkenaan pada mengendalikan data kepada siapa saja yang bisa melihat isi data [1].

2.2.4. UML

UML (*Unified Modeling Language*) adalah sebuah bahasa yang berdasarkan grafik/gambar untuk memvisualisasi, menspesifikasikan, membangun, dan pendokumentasian dari sebuah sistem pengembangan software berbasis OO (*Object-Oriented*). UML sendiri juga memberikan standar penulisan sebuah sistem blue print, yang meliputi konsep bisnis proses, penulisan kelas-kelas dalam bahasa program yang spesifik, skema *database*, dan komponen-komponen yang diperlukan dalam sistem *software senagai berikut* :

1. Interface

Interface merupakan kumpulan operasi tanpa implementasi dari suatu *class*. Implementasi operasi dalam *interface* dijabarkan oleh operasi didalam *class*. Oleh karena itu keberadaan *interface* selalu disertai oleh *class* yang mengimplementasikan operasinya. *Interface* ini merupakan salah satu cara mewujudkan prinsip enkapsulasi dalam obyek.

2. *Interaction*

Interaction digunakan untuk menunjukkan baik aliran pesan atau informasi antar *obyek* maupun hubungan antar *obyek*. Biasanya *interaction* ini dilengkapi

juga dengan teks bernama *operation signature* yang tersusun dari nama operasi, parameter yang dikirim dan tipe parameter yang dikembalikan.

3. *Note*

Note digunakan untuk memberikan keterangan atau komentar tambahan dari suatu elemen sehingga bisa langsung terlampir dalam model. *Note* ini bisa disertakan ke semua elemen notasi yang lain.

4. *Dependency*

Depedency merupakan relasi yang menunjukan bahwa perubahan pada salah satu elemen memberi pengaruh pada elemen lain. Elemen yang ada di bagian tanda panah adalah elemen yang tergantung pada elemen yang ada dibagian tanpa tanda panah.

Terdapat 2 *stereotype* dari *dependency*, yaitu *include* dan *extend*. *Include* menunjukkan bahwa suatu bagian dari elemen (yang ada digaris tanpa panah) memicu eksekusi bagian dari elemen lain (yang ada di garis dengan panah). *Extend* menunjukkan bahwa suatu bagian dari elemen di garis tanpa panah bisa disisipkan kedalam elemen yang ada di garis dengan panah.

5. *Association*

Association menggambarkan navigasi antar *class* (*navigation*), berapa banyak objek lain yang bisa berhubungan dengan satu objek (*multiplicity* antar *class*) dan apakah suatu *class* menjadi bagian dari *class* lainnya (*aggregation*).

Navigation dilambangkan dengan penambahan tanda panah di akhir garis. *Bidirectional navigation* menunjukkan bahwa dengan mengetahui salah satu *class* bisa didapatkan informasi dari *class* lainnya. Sementara *Uni Directional navigation* hanya dengan mengetahui *class* diujung garis association tanpa panah. kita bisa mendapatkan informasi dari *class* di ujung dengan panah, tetapi tidak sebaliknya.

Aggregation mengacu pada hubungan "*has-a*", yaitu bahwa suatu *class* memiliki *class* lain, misalnya Rumah memiliki *class* Kamar.

6. *Generalization*

Generalization menunjukkan hubungan antara elemen yang lebih umum ke elemen yang lebih spesifik. Dengan *generalization*, *class* yang lebih spesifik

(*subclass*) akan menurunkan atribut dan operasi dari *class* yang lebih umum (*superclass*) atau "*subclass is superclass*". Dengan menggunakan notasi *generalization* ini, konsep *inheritance* dari prinsip hirarki dapat dimodelkan.

7. Realization

Realization menunjukkan hubungan bahwa elemen yang ada di bagian tanpa panah akan merealisasikan apa yang dinyatakan oleh elemen yang ada di bagian dengan panah. Misalnya *class* merealisasikan *package*, component merealisasikan *class* atau *interface* [6].

2.2.4.1. Artifact UML

UML menyediakan beberapa notasi dan *artifact* standar yang bisa digunakan sebagai alat komunikasi bagi para pelaku dalam proses analisis dan desain. *Artifact* didalam UML didefinisikan sebagai informasi dalam bentuk yang digunakan atau dihasilkan dalam proses pengembangan perangkat. Contohnya adalah source code yang dihasilkan oleh proses pemrograman.

Yang harus diperhatikan untuk menjaga konsistensi antar artifact selama proses analisis dan desain adalah bahwa setiap perubahan yang terjadi pada satu *artifact* harus juga dilakukan pada *atifact* sebelumnya.

Untuk membuat suatu model, UML memiliki diagram grafis sebagai berikut :

1. *Use Case Diagram*
2. *Class Diagram*
3. *Statechart Diagram*
4. *Activity Diagram*
5. *Sequence Diagram*
6. *Collaboration Diagram*
7. *Component Diagram*
8. *Deployment Digram*

Dibuatnya berbagai jenis diagram diatas karena :

a. Setiap sistem yang kompleks selalu paling baik jika didekati melalui himpunan berbagai sudut pandang yang kecil yang satu sama lain

hampir saling bebas (*independent*). Sudut pandang tunggal senantiasa tidak mencukupi untuk melihat sistem yang besar dan kompleks.

b. Diagram yang berbeda-beda tersebut dapat menyatakan tingkatan yang berbeda-beda dalam proses rekayasa.
c. Diagram-diagram tersebut dibuat agar model yang dibuat semakin mendekati realitas

2.2.4.2.Tujuan UML

Tujuan utama UML diantaranya untuk :

1. Memberikan model yang siap pakai, bahasa pemodelan visual yang ekspresif untuk mengembangkan dan saling menukar model dengan mudah dan dimengerti secara umum.
2. Memberikan bahasa pemodelan yang bebas dari berbagai bahasa pemrograman dan proses rekayasa.
3. Menyatukan praktek-praktek terbaik yang terdapat dalam bahasa pemodelan.

2.2.4.3. OOP

Konsep berorientasi object ini di ciptakan untuk mempermudah dalam pengembangan program dengan cara mengikuti model yang telah ada dalam kehidupan nyata. Konsep - konsep pemrograman berorientasi object dalam java secara umum sama dengan yang digunakan oleh bahasa - bahasa lain, hanya terkadang terdapat perbedaan - perbedaan kecil antara penerapan konsep - konsep tersebut dalam masing - masing bahasa.

Object sebagai konsep kunci dari pemrograman berorientasi objek, baik dalam dunia nyata atau dalam sebuah program, sebuah object memiliki dua karakteristik, yaitu : *State* dan *Behavior*.

State adalah keadaaan dari sebuah objek, seperti MOBIL, memiliki warna, model, tahun pembuatan, kondisi, dll. Sedangkan *Behavior* adalah kelakuan dari objek tersebut, seperti mobil tadi bisa melaju, membelok, membunyikan klakson dan lain sebagainya.Objek menyimpan statenya dalam satu atau lebih variabel, dan mengimplementasikan behaviornya dengan metode. Kesimpulannya Object

adalah bagian software yang dibentuk dengan variabel - variabel dan metode - metode yang berhubungan dengan variabel tersebut.

Sebuah Object yang dibentuk dari sebuah kelas biasa di sebut instans dalam terminologi OOP. artinya object tersebut adalah wujud nyata dari sebuah kelas.

2.2.5. Java

Java adalah bahasa pemrograman yang dapat dijalankan di berbagai komputer termasuk telepon genggam. Dikembangkan oleh Sun Microsystems dan diterbitkan tahun 1995. Java tidak boleh disalah pahami sebagai JavaScript. JavaScript adalah bahasa scripting yang digunakan oleh *web browser*.

2.2.5.1. Versi Awal Java

Versi awal Java ditahun 1996 sudah merupakan versi *release* sehingga dinamakan Java Versi 1.0. Java versi ini menyertakan banyak paket standar awal yang terus dikembangkan pada versi selanjutnya :

1. java.lang: Peruntukan kelas elemen-elemen dasar.
2. java.io: Peruntukan kelas input dan output, termasuk penggunaan berkas.
3. java.util: Peruntukan kelas pelengkap seperti kelas struktur data dan kelas kelas penanggalan.
4. java.net: Peruntukan kelas TCP/IP, yang memungkinkan berkomunikasi dengan komputer lain menggunakan jaringan TCP/IP.
5. java.awt: Kelas dasar untuk aplikasi antarmuka dengan pengguna (GUI)
6. java.applet: Kelas dasar aplikasi antar muka untuk diterapkan pada penjelajah web.

2.2.5.2. Kelebihan Java

1. Multiplatform. Kelebihan utama dari Java ialah dapat dijalankan di beberapa platform / sistem operasi komputer, sesuai dengan prinsip tulis sekali, jalankan di mana saja. Dengan kelebihan ini pemrogram cukup menulis sebuah program Java dan dikompilasi (diubah, dari bahasa yang dimengerti manusia menjadi bahasa mesin / *bytecode*) sekali lalu hasilnya dapat

dijalankan di atas beberapa platform tanpa perubahan. Kelebihan ini memungkinkan sebuah program berbasis java dikerjakan diatas *operating system* Linux tetapi dijalankan dengan baik di atas Microsoft Windows. Platform yang didukung sampai saat ini adalah Microsoft Windows, Linux, Mac OS dan Sun Solaris. Penyebanya adalah setiap sistem operasi menggunakan programnya sendiri-sendiri (yang dapat diunduh dari situs Java) untuk meninterpretasikan *bytecode* tersebut.

2. OOP (*Object Oriented Programming* - Pemrogram Berorientasi Objek) yang artinya semua aspek yang terdapat di Java adalah Objek. Java merupakan salah satu bahasa pemrograman berbasis objek secara murni. Semua tipe data diturunkan dari kelas dasar yang disebut *Object*. Hal ini sangat memudahkan pemrogram untuk mendesain, membuat, mengembangkan dan mengalokasi kesalahan sebuah program dengan basis Java secara cepat, tepat, mudah dan terorganisir. Kelebihan ini menjadikan java sebagai salah satu bahasa pemrograman termudah, bahkan untuk fungsi *advance* seperti komunikasi antar komputer.

3. Perpustakaan Kelas Yang Lengkap, Java terkenal dengan kelengkapan *library*/perpustakaan (kumpulan program program yang disertakan dalam pemrograman java) yang sangat memudahkan dalam penggunaan oleh para pemrogram untuk membangun aplikasinya. Kelengkapan perpustakaan ini ditambah dengan keberadaan komunitas Java yang besar yang terus menerus membuat perpustakaan-perpustakaan baru untuk melingkupi seluruh kebutuhan pembangunan aplikasi.

4. Bergaya C++, memiliki sintaks seperti bahasa pemrograman C++ sehingga menarik banyak pemrogram C++ untuk pindah ke Java. Saat ini pengguna Java sangat banyak, sebagian besar adalah pemrogram C++ yang pindah ke Java. Universitas-universitas di Amerika Serikat juga mulai berpindah dengan mengajarkan Java kepada murid-murid yang baru karena lebih mudah

dipahami oleh murid dan dapat berguna juga bagi mereka yang bukan mengambil jurusan komputer.

5. Pengumpulan sampah otomatis, memiliki fasilitas pengaturan penggunaan memori sehingga para pemrogram tidak perlu melakukan pengaturan memori secara langsung (seperti halnya dalam bahasa C++ yang dipakai secara luas).

2.2.5.3. Kekurangan Java

1. Tulis sekali, perbaiki dimana saja, masih ada beberapa hal yang tidak kompatibel antara flatform satu dengan flatform lainnya. Untuk J2SE misalnya SWT-AWT *bridge* yang sampai sekarang tidak berfungsi pada MAC OS X.

2. Mudah didekompilasi. Dekompilasi adalah proses membalikkan dari kode jadi menjadi kode sumber. Ini dimungkinkan karena kode jadi Java merupakan *bytecode* yang menyimpan banyak atribut bahasa tingkat tinggi, seperti nama-nama kelas, metode, dan tipe data. Hal yang sama juga terjadi pada Microsoft .NET Platform. Dengan demikian, algoritma yang di gunakan program akan lebih sulit di sembunyikan dan mudah di bajak

3. Penggunaan memori yang banyak. Penggunaan memori untuk program berbasis Java jauh lebih besar daripada bahasa tingkat tinggi generasi sebelumnya seperti C/C++ dan Pascal (lebih spesifik lagi, Delphi dan Object Pascal). Biasanya ini bukan merupakan masalah bagi pihak yang menggunakan teknologi terbaru (karena *trend* memori terpasang makin murah), tetapi menjadi masalah bagi mereka yang masih harus berkutat dengan mesin komputer berumur lebih dari 4 tahun [7].

2.2.6. NetBeans

NetBeans mengacu pada dua hal, yakni platform untuk pengembangan aplikasi desktop java, dan sebuah Integrated Development Environment (IDE) yang dibangun menggunakan platform NetBeans.

Platform NetBeans memungkinkan aplikasi dibangun dari sekumpulan komponen perangkat lunak moduler yang disebut ”modul”. Sebuah modul adalah suatu arsip Java (*Java archive*) yang memuat kelas-kelas Java untuk berinetraksi dengan NetBeans Open API dan *file* manifestasi yang mengidentifikasinya sebagai modul. Aplikasi yang dibangun dengan modul-modul dapat dikembangkan dengan menambahkan modul-modul baru. Karena modul dapat dikembangkan secara independen, aplikasi berbasis platform NetBeans dapat dengan mudah dikembangkan oleh pihak ketiga secara mudah dan *powerful*.

2.2.6.1. Flatform NetBeans

Platform NetBeans adalah *framework* yang dapat digunakan kembali (*reusable*) untuk menyederhanakan pengembangan aplikasi desktop. Ketika aplikasi berbasis platform NetBeans dijalankan, kelas Main dari platform dieksekusi.Modul-modul yang tersedia ditempatkan di sebuah registry di dalam memori, dan tugas startup modul dijalankan. Secara umum, kode modul dimuatkan ke dalam memori hanya ketika ia diperlukan. Aplikasi dapat menginstal modul secara dinamis. Aplikasi dapat memasukkan modul *Update Center* untuk mengijinkan pengguna aplikasi men-*download digitally-signed upgrade* dan fitur-fitur baru secara langsung ke dalam aplikasi yang berjalan. Penginstalan kembali sebuah *upgrade* atau rilis baru tidak memaksa pengguna untuk men-*download* keseluruhan aplikasi lagi.

Platform NetBeans menawarkan layanan-layanan yang umum bagi aplikasi desktop, mengijinkan pengembang untuk fokus ke logika yang spesifik terhadap aplikasi. Fitur-fitur yang disediakan oleh platform NetBeans:

a. Manajemen antarmuka (misal: menu & *toolbar*)
b. Manajemen pengaturan pengguna
c. Manajemen penyimpanan (menyimpan dan membuka berbagai macam data)
d. Manajemen jendela
e. *Wizard framework* (mendukung dialog langkah demi langkah)

2.2.6.2. NetBeans IDE

NetBeans IDE adalah IDE *open source* yang ditulis sepenuhnya dengan bahasa pemrograman Java menggunakan platform NetBeans. NetBeans IDE mendukung pengembangan semua tipe aplikasi Java (J2SE, *web*, EJB, dan aplikasi *mobile*). Fitur lainnya adalah sistem proyek berbasis *Ant*, kontrol versi, dan *refactoring*.Versi terbaru saat ini adalah NetBeans IDE 5.5.1 yang dirilis Mei 2007 mengembangkan fitur-fitur Java EE yang sudah ada (termasuk Java *Persistence support*, EJB-3 dan JAX-WS). Sementara paket tambahannya, NetBeans *Enterprise Pack* mendukung pengembangan aplikasi perusahaan Java EE 5, meliputi alat desain visual SOA, skema XML, web service dan pemodelan UML. NetBeans C/C++ Pack mendukung proyek C/C++.

Modularitas: Semua fungsi IDE disediakan oleh modul-modul. Tiap modul menyediakan fungsi yang didefinisikan dengan baik, seperti dukungan untuk bahasa pemrograman Java, editing, atau dukungan bagi CVS. NetBeans memuat semua modul yang diperlukan dalam pengembangan Java dalam sekali *download*, memungkinkan pengguna untuk mulai bekerja sesegera mungkin. Modul-modul juga mengijinkan NetBeans untuk bisa dikembangkan. Fitur-fitur baru, seperti dukungan untuk bahasa pemrograman lain, dapat ditambahkan dengan menginstal modul tambahan.Sebagai contoh, Sun Studio, Sun Java Studio Enterprise, dan Sun Java Studio Creator dari Sun Microsystem semuanya berbasis NetBeans IDE [7].

2.2.7. MySQL

MySQL adalah sebuah perangkat lunak sistem manajemen basis data SQL (Database Management System) atau DBMS yang multithread, multi-user, dengan sekitar 6 juta instalasi di seluruh dunia. MySQL merupakan perangkat lunak basis data open source yang paling digemari, karena perangkat lunak ini merupakan perangkat lunak basis data yang powerful dan cukup stabil untuk digunakan sebagai media penyimpanan data.

Kehandalan suatu sistem basisdata (DBMS) dapat diketahui dari cara kerja pengoptimasinya dalam melakukan proses perintah-perintah SQL yang dibuat oleh pengguna maupun program-program aplikasi yang memanfaatkannya.

Sebagai peladen basis data, MySQL mendukung operasi basisdata transaksional maupun operasi basisdata nontransaksional. Pada modus operasi nontransaksional, MySQL dapat dikatakan unggul dalam hal unjuk kerja dibandingkan perangkat lunak peladen basisdata kompetitor lainnya. Namun demikian pada modus nontransaksional tidak ada jaminan atas reliabilitas terhadap data yang tersimpan, karenanya modus nontransaksional hanya cocok untuk jenis aplikasi yang tidak membutuhkan reliabilitas data seperti aplikasi blogging berbasis web (*wordpress*), CMS, dan sejenisnya. Untuk kebutuhan sistem yang ditujukan untuk bisnis sangat disarankan untuk menggunakan modus basisdata transaksional, hanya saja sebagai konsekuensinya unjuk kerja MySQL pada modus transaksional tidak secepat unjuk kerja pada modus nontransaksional.

2.2.7.1 Kelebihan MySQL

1. Merupakan DBMS yang gratis / open source berlisensi GPL (generic public license).
2. Cocok untuk perusahaan dengan skala yang kecil.
3. Tidak membutuhkan spesifikasi hardware yang tinggi untuk bisa menjalankan MWSQL ini bahkan dengan spesifikasi hardware yang minimal sekalipun.
4. Bisa berjalan pada lebih dari satu platform system operasi, misalnya windows, linux, FreeBSD, Solaris, dan masih banyak lagi.
5. Cepat dalam menjalankan perintah *SQL / Structured Query Language* misalnya ketika akan menyeleksi suatu data atau memasukkan suatu data karena MYSQL merupakan turunan dari konsep SQL.*Multi user*, artinya database dapat digunakan oleh beberapa user dalam waktu bersamaan tanpa mengalami masalah atau konflik.
6. MYSQL memiliki ragam tipe data yang sangat kaya, seperti signed / unsigned integer, float, double, char, text, date, timestamp, dan lain-lain.
7. MYSQL memiliki beberapa lapisan keamanan, seperti *subnetmask*, nama *host*, dan izin akses user dengan system perijinan yang mendetail serta sandi/password terenkripsi.

8. MYSQL dapat melakukan koneksi dengan computer client menggunakan Protokol TCP/IP, Unix Socket (UNIX), atau Named Pipes (windows NT).
9. MYSQL memiliki antar muka / *interface* terhadap berbagai aplikasi dan bahasa pemrograman dengan menggunakan fungsi API (*Application Programming Interface*).
10. *Command and function*, MYSQL memiliki fungsi dan operator secara penuh yang mendukung perintah *select* dan *where* dalam query.
11. *Structure Table,* MYSQL memiliki struktur tabel yang lebih fleksibel dalam menangani *ALTER TABLE* dibandingkan DBMS lainnya.

2.2.7.2 Kekurangan MySQL

1. Tidak cocok untuk menangani data dengan jumlah yang besar, baik untuk menyimpan data maupun untuk memproses data.
2. Memiliki keterbatasan kemampuan kinerja pada server ketika data yang disimpan telah melebihi batas maksimal kemampuan daya tampung server karena tidak menerapkan konsep *Technology Cluster Server.*