BAB II

LANDASAN TEORI

Bab ini membahas mengenai teori-teori, definisi para ahli dan konsep yang digunakan untuk penelitian yang dilakukan. Dengan kata lain pada bab ini berisi tentang tinjauan pustaka yang berisi hal-hal yang menunjang dan membantu proses penelitian khususnya pada aspek literasinya. Dan berikut adalah pembahasan lebih jelasnya.

2.1 Konsep Sistem Informasi

Menurut Laudon yang diterjemahkan oleh Chriswan Sungkono dan Mahmudin Eka,

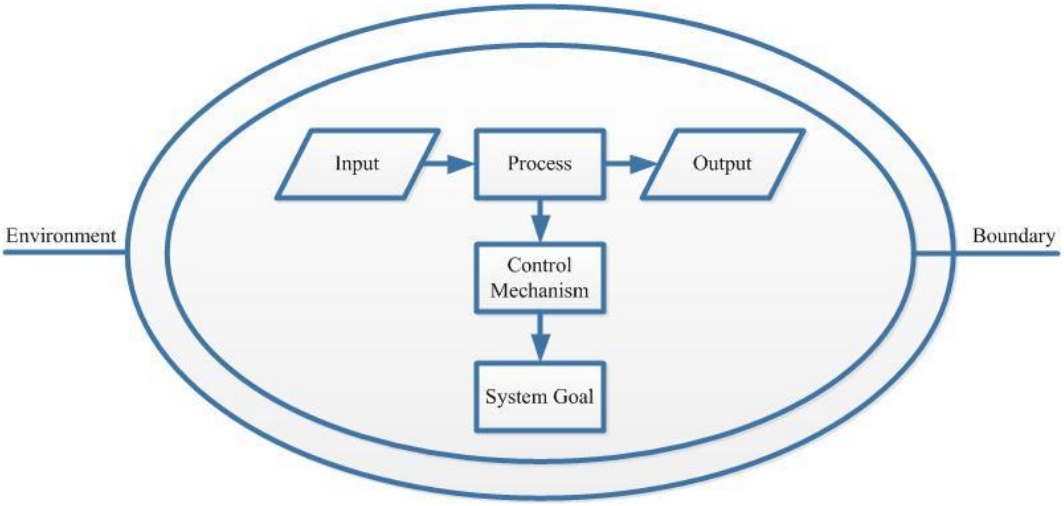
> Sistem Informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi, bersifat manajerial dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan [1,p.42].

Sama halnya dengan teori Sistem informasi yang dikemukakan Edhy Sutanta, dimana beliau menyatakan bahwa sistem informasi merupakan sekumpulan elemen yang terdapat dalam sebuah sistem yang disatukan oleh satu pencapaian dan tujuan yang sama [2].

Dari teori di atas dapat disimpulkan bahwa Sistem Informasi merupakan suatu sistem yang merupakan dari proses kumpulan fakta yang di lihat, yang kemudian di catat dalam suatu data, kemudian data tersebut diolah menjadi suatu informasi yang bermanfaat bagi yang membutuhkan.

2.2 Elemen Sistem Informasi

Sistem Informasi memiliki elemen sistem tersendiri. Dimana elemen-elemen inilah yang menjadikan sistem tersebut menjadi satu kesatuan yang saling behubungan dan elemen-elemen tersebutlah yang menentukan sebuah sistem sebagai suatu sistem informasi. Adapun elemen-elemen sistem informasi tersebut diantaranya *system goal, input, process, output, control mechanism and feed back, boundary, environment* [3]. Jika dilihat gambar dibawah ini akan diketahui ketergantungan elemen sistem satu sama lain.



Gambar 2.1 Elemen Sistem Informasi
(Sumber: Pengenalan Sistem Informasi [3, p.55])

1. System Goal

System goal merupakan target atau sasaran sistem yang mengarahkan suatu sistem sebagaimana arah sistem tersebut ditentukan. Sasaran sistem dalam mengolah input yang masuk. Tujuan dari sistem bersifat mengerucut dimana berbagai tujuan dari banyak pihak dikerucutkan menjadi satu tujuan yang sama yang dikemas dalam sistem itu sendiri. Namun secara umum sistem ditujukan

sebagai pendukung fungsi kepengurusan, pendukung pengambilan keputusan dan pendukung kegiatan operasi.

2. Input

Terdapat dua jenis inputan yaitu *maintenance input* dan *signal input*. Jika inputan dianalogikan sebagai sekempulan energi yang dimasukkan ke dalam sistem. Maka *maintenance input* merupakan energi yang dimasukan dengan tujuan memberikan kesempatan suatu sistem untuk beroperasi, contohnya adalah *operating system*, digunakan untuk mengoperasikan komputernya. Sedangkan *signal input* merupakan energi yang nantinya akan dilakukan pemrosesan yang kemudian akan menghasilkan output. Hal tersebut menyatakan bahwa *signal input* merupakan data-data yang akan di proses dan kemudian dijadikan informasi.

3. Process

Process merupakan tahap pengolahan inputan dimana inputan yang hendak diproses atau diolah akan masuk pada tahap perubahan atau transformasi yang nantinya akan menghasilkan suatu yang memilki nilai yang lebih yaitu *output*.

4. Output

Output merupakan keluaran hasil dari pemrosesan inputan. Adapun jenis inputan yang diproses adalah *signal input*. *Output* yang dihasilkan dari proses pengolahan *input* dapat berupa apa saja, tergantung dari inputan dan proses pengolahannya.

5. Control Mechanism dan *Feed Back*

Control Mechanism diwujudkan dengan memanfaatkan umpan balik (*feed back*) yang ada. Umpan balik sendiri difungsikan sebagai titik acu yang dapat memantau apakah sistem yang digunakan dapat berjalan sesuai dengan tujuan atau sebaliknya.

6. Boundary

Boundary merupakan batasan sistem dianalogikan sebagai daerah-daerah yang membatasi satu sistem dengan system yang lainnya maupun dengan lingkungan luarnya. Dimana Batasan ini memungkinkan suatu system dipandang subagai suatu kesatuan yang utuh, dikarenakan batasan tersebut menunjukan ruang lingkup dari system tersebut.

7. Environment

Environment merupakan berbagai hal yang terdapat diluar *Boundary* atau batasan system. Dimana walaupun posisinya berada diluar namun tetap memiliki peranan penting terhadap sistem yang bersangkutan.

2.3 Pemahaman Mengenai Organisasi

Organisasi merupakan sebuah tim yang terdiri dari beberapa individu yang cukup matang dengan tingkat ketergantungan antar anggotanya yang disatukan dalam suatu wadah yang diwarnai dengan motivasi untuk mencapai tujuan bersama [4]. Dalam ilmu-ilmu sosial, organisasi dipelajari dari berbagai bidang

ilmu, terutama sosiologi, ekonomi, ilmu politik, psikologi, dan manajemen. Organisasi pada dasarnya digunakan sebagai tempat atau wadah dimana orang-orang berkumpul, bekerjasama secara rasional dan sistematis, terencana, terorganisasi, terpimpin dan terkendali, dalam memanfaatkan sumber daya, sarana-parasarana, data, dan lain sebagainya yang digunakan secara efisien dan efektif untuk mencapai tujuan organisasi.

Sebuah organisasi dapat terbentuk karena dipengaruhi oleh beberapa aspek seperti penyatuan visi dan misi serta tujuan yang sama dengan perwujudan eksistensi sekelompok orang tersebut terhadap masyarakat. Organisasi yang dianggap baik adalah organisasi yang dapat diakui keberadaannya oleh masyarakat disekitarnya, karena memberikan kontribusi seperti; pengambilan sumber daya manusia dalam masyarakat sebagai anggota-anggotanya sehingga menekan angka pengangguran.

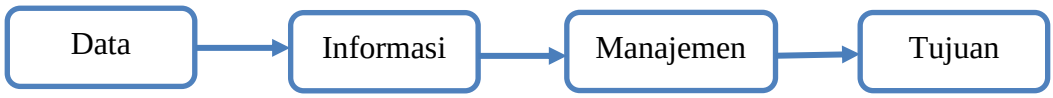
Orang-orang yang ada di dalam suatu organisasi mempunyai suatu keterkaitan yang terus menerus. Rasa keterkaitan ini, bukan berarti keanggotaan seumur hidup. Akan tetapi sebaliknya, organisasi menghadapi perubahan yang konstan di dalam keanggotaan mereka, meskipun pada saat mereka menjadi anggota, orang-orang dalam organisasi berpartisipasi secara relatif teratur.

2.4 Organisasi dan Informasi

Perkembangan teknologi yang semakin pesat memberikan dampak secara langsung maupun tidak langsung terhadap kemajuan organisasi. Dimana perubahan kearah yang positif ini menimbulkan suatu persaingan dimana setiap

organisasi yang memiliki jiwa kompetitif yang tinggi masing-masing ingin menunjukan perkembangannya menjadi lebih maju, lebih cepat, lebih banyak keunggulannya dari organisasi-organisasi yang lainnya.

Dalam pencapaian tujuan organisasi tentunya suatu organisasi dituntut untuk memiliki manajemen yang tepat yang dapat menyokong semua kegiatan yang ada di organisasi tersebut, sehingga semua aktifitas yang dilakukan dapat berjalan sesuai dengan kebutuhan. Untuk menciptakan sistem manajemen yang baik maka diperlukan informasi yang baik pula. Tanpa dukungan informasi, suatu manajemen dalam sebuah organisasi tidak akan dapat mencapai tujuan yang direncanakan. Karena informasi, manajemen dan tujuan organisasi merupakan suatu kesatuan yang saling membutuhkan.



Gambar 2.2 Hubungan data dan tujuan organisasi
(Sumber: Manajemen Sistem Informasi [5,p.5])

2.5 Pemahaman Mengenai Administrasi Kependudukan Desa

Menurut arti kata administrasi berasal dari kata berbahasa latin yaitu *Ad Ministrare* yang maknanya adalah melayani, membantu atau memenuhi. Sedangkan menurut Prof. Drs. H. AW. Widjaja menjelaskan bahwa administrasi dapat diartikan sebagai proses pembuatan penyelenggaraan untuk setiap aktifitas kerjasama antara sekelompok orang yang memiliki tujuan bersama [6].

Dari penjelasan mengenai administrasi di atas dapat disimpulkan bahwa Administrasi Kependudukan Desa merupakan serangkaian proses yang dilakukan oleh sekelompok orang yang memiliki kewenangan atau biasa disebut organisasi yang menjadi penyelenggara pemerintahan tingkat desa (rukun warga). Dimana mereka melakukan pendataan kependudukan sebagai salah satu usaha untuk mewujudkan Demokrasi Pancasila guna peningkatan taraf hidup masyarakatnya.

2.6 Pelaksanaan Administrasi Kependuduk Desa

Dalam pelaksanaan administrasi kependudukan desa pihak organisasi sebagai pemerintahan tingkat desa mewajibkan warganya untuk memberikan data kependudukan secara lengkap. Dimana seluruh data tersebut akan diasripkan dalam bentuk buku-buku. Adapun buku-buku yang wajib dibuat diantaranya adalah sebagai berikut [6]:

1. **Buku Kependudukan**

   Berisi data-data kependudukan yang berasal dari keterangan dan bukti-bukti yang disampaikan oleh penduduk yang bersangkutan dan dapat dipertanggungjawabkan kebenarannya.

2. **Buku Perkembangan Kependudukan**

   Berisi data-data kependudukan yang berasal dari perkembangan kependudukan yang mana perkembangan penduduk disebabkan oleh beberapa faktor diantaranya adalah kematian, kelahiran dan kepindahan.

3. Buku Keluarga

Berisi data-data kepala keluarga yang terdaftar sebagai keluarga di lingkungan yang dimaksud. Dimana tujuan utama dari buku keluarga ini adalah untuk mengetahui jumlah keseluruhan kepala keluarga terdaftar dan juga anggota keluarga untuk setiap keluarganya.

4. Buku Jumlah Penduduk

Berisi data hasil pengolahan buku kependudukan dimana untuk setiap bulannya jumlah penduduk akan dihitung dan hasilnya akan disimpan di buku ini.

5. Buku Pelayanan Administrasi

Buku pelayanan administrasi ini berisi *record* kegiatan pelayanan administrasi yang dilakukan selama masa jabatan.

2.7 Sistem Informasi Administrasi Kependudukan Desa

Dari pemahaman mengenai konsep informasi, organisasi dan informasi, Administrasi Kependudukan Desa serta pelaksanaannya yang diuraikan pada poin sebelumnya. Dapat disimpulkan bahwa Sistem Informasi Administrasi Kependudukan Desa merupakan suatu sistem informasi yang ditujukan khususnya untuk me-*manage* aktifitas-aktifitas administrasi kependudukan desa seperti pendataan penduduk, pendataan perkembangan penduduk, pendataan keluarga, pendataan jumlah penduduk, dan pendataan pelayanan administrasi yang tentunya tidak lepas dari sentuhan teknologi yang sekarang sedang berkembang.

2.8 Pemograman Berbasis *Dekstop*

Pemograman berbasis *dekstop* merupakan salah satu jenis pemograman dari segi kegunaannya. Yang mana biasanya dalam pemograman berbasis *dekstop* akan menghasilkan suatu program aplikasi yang dapat berdiri sendiri pada sebuah PC. Dengan kata lain program yang dibuat memang diperuntukan untuk satu PC dengan melakukan penginstalan program terlebih dahulu, dan biasanya disertai dengan suatu lisensi. Dalam hal ini penulis menggunakan UML dalam pemodelan rancangan sistemnya dan menggunakan JAVA sebagai bahasa pemogramannya serta penggunaan MySQL sebagai *Data Base Management System* (DBMS) dengan menggunakan bahasa SQL. Berikut adalah penjelasan lebih lengkap mengena UML, JAVA dan SQL.

2.8.1 *Unified Modelling Language* (UML)

UML atau *Unified Modelling Language* yang berarti bahasa pemodelan standar merupakan sebuah "bahasa" yang telah menjadi standar dalam dunia industri yang digunakan sebagai *tools* untuk visualisasi, perancangan dan pendokumentasian sistem sebuah perangkat lunak.

Jika UML dikatakan sebagai sebuah bahasa, maka UML memiliki notasi serta sintaks dan sematik. Notasi UML merupakan sekumpulan bentuk khusus untuk menggambarkan berbagai diagram *software*. Pada saat pembuatan model maka diharuskan mengikuti aturan-aturan yang merupakan standarisasi proses pemodelannya. Dalam UML sendiri tidak hanya disajikan dalam bentuk diagram, namun juga menceritakan konteks di dalamnya.

Dengan menggunakan UML kita dapat membuat model untuk semua jenis aplikasi piranti lunak, dimana aplikasi tersebut dapat berjalan pada piranti keras, sistem operasi dan jaringan apapun, serta ditulis dalam bahasa pemrograman apapun. Tetapi karena UML juga menggunakan *class* dan *operation* dalam konsep dasarnya, maka ia lebih cocok untuk penulisan piranti lunak dalam bahasa-bahasa berorientasi objek seperti C++, Java, C# atau VB.NET. Walaupun demikian, UML tetap dapat digunakan untuk *modeling* aplikasi prosedural dalam VB atau C.

2.8.1.1. Penggunaan UML

Menurut Prabowo dan Herlawati UML diaplikasikan untuk maksud tertentu, biasanya antara lain untuk [7,p.6]:

1. Merancang perangkat lunak
2. Sebagai sarana komunikasi antara perangkat lunak dengan proses bisnis
3. Menjabarkan sistem secara rinci untuk analisa dan mencari apa yang diperlukan sistem
4. Mendokumentasikan sistem yang ada, proses-proses dan organisasinya

2.8.1.2. Diagram-diagram UML

Terdapat sembilan jenis diagram yang dikelompokan berdasarkan sifatnya yaitu statis dan dinamis. Diagram-diagram tersebut diantaranya [7,p.10]:

1. ***Class diagram***

   Class diagram merupakan diagram bersifat statis. Diagram ini memperlihatkan himpunan kelas-kelas, antarmuka-antarmuka, kolaborasi-kolaborasi, serta relasi-relasi. Diagram ini umum dijumpai pada pemodelan sistem berorientasi objek. Meskipun bersifat statis, sering pula diagram kelas yang memuat kelas-kelas aktif.

2. ***Package diagram***

   Package diagram merupakan diagram yang bersif statis. Diagram ini memperlihatkan kumpulan kelas-kelas yang merupakan bagian dari diagram komponen

3. ***Use case diagram***

   Use case diagram bersifat statis. Diagram ini memperlihatkan himpunan *use-case* dan aktor-aktor. Diagram ini terutama sangat penting untuk mengorganisasikan dan memodelkan perilaku suatu sistem yang dibutuhkan serta diharapkan pengguna.

4. Sequence dan *Interaction diagram*

Sequence dan *Interaction diagram* bersifat dinamis. Diagram urutan adalah diagram interaksi yang menekankan pada pengiriman pesan dalam waktu tertentu

5. *Communication Diagram*

Communication diagram bersifat dinamis. Diagram sebagai pengganti diagram kolaborasi UML 1.4 yang menekankan organisasi struktural dari objek-objek yang menerima serta mengirim pesan.

6. *Statechart Diagram*

Diagram ini bersifat dinamis. Diagram status memperlihatkan keadaan-keadaan pada sistem, memuat status, transisi, kejadian serta aktifitas. Diagram ini digunakan untuk memperlihatkan sifat dinamis dari antarmuka, kelas, kolaborasi dan terutama penting pada pemodelan sistem-sistem yang reaktif.

7. *Activity Diagram*

Activity diagram bersifat dinamis. Diagram aktifitas adalah tipe khusus dari diagram status yang memperlihatkan aliran dari suatu aktifitas ke aktifitas lainnya dalam suatu sistem. Diagram ini terutama penting dalam pemodelan fungsi-fungsi suatu sistem dan memberi tekanan pada aliran kendali antar objek.

8. Component Diagram

Component diagram bersifat statis. Diagram ini memperlihatkan organisasi serta kebergantungan sistem/perangkat lunak pada komponen-komponen yang telah ada sebelumnya. Diagram ini berhubungan dengan diagram kelas dimana komponen secara tipikal dipetakan ke dalam satu atau lebih kelas, antarmuka dan kolaborasi.

9. Deployment Diagram

Deployment diagram bersifat statis. Diagram ini memperlihatkan konfigurasi saat aplikasi dijalankan. Memuat simpul-simpul beserta komponen-komponen yang ada di dalamnya. Diagram deployment berhubungan erat dengan diagram komponen dimana diagram ini memuat satu atau lebih komponen. Diagram ini sangat berguna saat aplikasi kita berlaku sebagai aplikasi yang dijalankan pada banyak mesin.

2.8.2 JAVA

Java merupakan sebuah *masterpiece* dalam dunia pemograman yang diprakaryai oleh James Gosling pada tahun 1995, ketika beliau masih bergabung dengan Sun Microsystem Inc yang saat ini merupakan bagian dari Oracle. Yang mana JAVA sendiri merupakan bahasa pemograman yang dapat dijalankan di berbagai komputer termasuk telepon genggam. Bahasa ini banyak mengadopsi sintaksis yang terdapat pada C dan C++ namun dengan sintaksis model objek yang lebih sederhana.

Java merupakan bahasa pemograman berorientasi objek yang dapat diaplikasikan untuk pengembangan aplikasi mandiri, berbasi internet, dan lain-lain.

Saat ini java merupakan bahasa pemrograman yang paling populer digunakan, dan secara luas dimanfaatkan dalam pengembangan berbagai jenis perangkat lunak aplikasi ataupun aplikasi berbasis web.

2.8.2.1 Fitur-fitur pada JAVA

Beberapa fitur yang dimiliki oleh JAVA ditujukan untuk membuat beragam aplikasi. Adapun fitur-fitur tersebut adalah sebagai berikut [8]:

1. **Program di lingkungan *web browser***

   Untuk program di lingkungan *web browser* terdapat fitur *applet.* Dimana program ini akan dieksekusi melalui *web browser* kemudian menugaskan *Java Interpreter* untuk mengeksekusinya.

2. **Program di lingkungan *web server***

   Untuk program di lingkungan *web server*. Terdapat dua fitur yang dapat digunakan. Yang pertama adalah *Java Server Pages* sebagai *web scripting* yang ditempelkan pada html yang nantinya akan diolah oleh *web server*. Yang kedua adalah *Java Servlet* yang merupakan modul di *web server.*

3. **Program mandiri**

   Program mandiri atau biasa disebut *stand alone application*. JAVA merupakan pilihan bagus dalam pembuatan *applet*, sekaligus sebagai bahasa bermaksud umum guna mengembangkan semua jenis program yang dapat dijalankan pada komputer.

2.8.2.2 Konsep dalam JAVA

Terdapat beberapa konsep penting yang dimiliki JAVA dalam pendekatan berorintasi objek yang digunakan dalam penelitian pembangunan sebuah aplikasi, diantaranya adalah sebagai berikut [8]:

1. **Pengapsulan (*Encapsutaion*)**

   Pengapsulan digunakan untuk membagi objek menjadi dua bagian yaitu bagian internal yang berisi rincian-rincian dan bagian eksternal yang biasanya merupakan antarmuka objek. Pembagian objek ini dilakukan dengan tujuan melindungi rincian-rincian yang merupakan bagian internal, sehingga jika terjadi pengaksesan di bagian eksternal mana bagian internalnya akan tetap aman karena tersebunyi.

2. **Pewarisan (*Inheritance*)**

   Pewarisan digunakan untuk mengklasifikasikan kelas-kelas yang ada, dengan cara mendekripsikan kelas-kelas umum yang kemudian kelas-kelas umum tersebut akan dapat diwariskan kepada

kelas lain dengan kemungkinan menambahkan hal-hal yang lebih spesifik lagi dalam setiap sub kelasnya.

3. *Polymorphism*

Polymorphism merupakan salah satu fitur yang ada karena interaksi antara konsep pewarisan dan *dynamic binding*. Dimana *Polymorphism* berarti memiliki banyak bentuk. Yang berarti jika terdapat dua buah atau lebih objek yang secara *interface* memiliki kesamaan namun memiliki perilaku-perilaku yang berbeda maka dapat dikatakan bahwa objek-objek tersebut termasuk kedalam *Polymorphism*.

4. Pesan (*Message*)

Sebagian besar objek-objek yang ada memerlukan objek-objek lain dalam melakukan banyak hal. Pada proses komunikasi objek maka objek yang akan berkomunikasi lewat pesan-pesan. Salah satu contoh pesannya adalah parameter.

2.8.3 *Structured Query Language* (SQL)

SQL merupakan bahasa standar untuk *Relational Database Management System* (RDBMS). Hal tersebut berdasarkan ketetapan yang buat oleh ANSI/ISO yang merupakan badan standarisasi di Amerika. Dimana SQL sendiri dikembangkan dan didefinisikan oleh para peneliti di IBM.

Menurut Bambang Wahyudi SQL merupakan bahasa non-prosedural yang digunakan untuk mengakses *databse* [9, p.243]. Proses kerja dari SQL sendiri dimulai ketika *user* menkonsepkan dengan menggunakan bahasa SQL mengenai apa yang ingin mereka dapatkan, lalu dari perintah yang diberikan tersebut *interpreter* bahasa SQL akan secara otomatis menjalankan sesuai dengan prosedur kerjanya dan kemudian akan mengerjakan yang diperintahkan *user* kepadanya.

Berdasarkan tujuannya, secara garis besar SQL dibagi menjadi dua yaitu *Data Definition Language* (DDL) dan *Data Manipulation Language* (DML) [9]. Dimana DDL diperuntukan untuk penyusuan data, sedangkan DML diperuntukan untuk pengolahan data. Contoh sintak yang digunakan pada DDL seperti CREATE TABLE, ALTER TABLE, DROP TABLE, CREATE INDEX, DROP INDEX, SELECT INTO. Sedangkan untuk DML sintak yang digunakan seperti SELECT, UPDATE, DELETE, INSERT INTO.

2.9 Program Aplikasi Pendukung

Berikut adalah program aplikasi pendukung khususnya dalam pembangunan sistem berbentuk program aplikasi. Adapun program aplikasi yang digunakan untuk membangun sistemnya adalah Netbeans dan Jasper Report.

2.9.1 Netbeans IDE

Netbeans merupakan sebuah aplikasi Integrated *Development Environment* (IDE) yang berbasiskan Java dari Sun Microsystems yang berjalan di atas *swing*. *Swing* merupakan sebuah teknologi Java untuk pengembangan aplikasi desktop

yang dapat berjalan pada berbagai macam *platform* seperti Windows, Linux, Mac OS X dan Solaris.

Sebuah IDE merupakan lingkup pemrograman yang diintegrasikan ke dalam suatu aplikasi perangkat lunak yang menyediakan *Graphic User Interface* (GUI), suatu *code editor* atau *text*, suatu *compiler* dan suatu *debugger*. Selain memiliki keunggulan Netbeans pun memiliki beberapa kelemahan. Berikut keunggulan dan kelemahan Netbenas:

Keunggulan Netbeans

1. Memiliki ribuan *free plugin* yang dapat diunggah secara langsung dari *website* resminya.
2. Pengembangan yang dilakukan oleh Sun Microsystem menjadikan Netbeans sangat kompetebel dengan *Swing.*
3. Bisa digunakan untuk berbagai macam pemograman berbasis *web* maupun *desktop*.
4. Tersedianya GlassFish V2 UR2 dan Apache Tomcat 6.0.16

Kelemahan Netbeans

1. Hanya mendukung pengembangan Java GUI Swing
2. Tidak dapat mengedit JAVA GUI karena dipatenkan dalam *generated code.*

2.9.2 MySQL

MySQL adalah sebuah implementasi dari sistem manajemen basisdata relasional (RDBMS) yang didistribusikan secara gratis. Setiap pengguna dapat secara bebas menggunakan MySQL, namun dengan batasan perangkat lunak tersebut tidak boleh dijadikan produk turunan yang bersifat komersial.

MySQL sebenarnya merupakan turunan salah satu konsep utama dalam basisdata yang telah ada sebelumnya yaitu SQL (*Structured Query Language*). SQL adalah sebuah konsep pengoperasian basisdata, terutama untuk pemilihan atau seleksi dan pemasukan data, yang memungkinkan pengoperasian data dikerjakan dengan mudah secara otomatis.

2.9.3 JasperReports

JasperReports adalah aplikasi yang menyediakan fitur pembuatan laporan *open source* yang paling populer di dunia. Jasper Reports seluruhnya ditulis dengan bahasa Java.

JasperReports mampu menggunakan data yang berasal dari segala jenis sumber data dan menghasilkan dokumen dengan *pixel* sempurna yang dapat dilihat, dicetak atau diekspor dalam berbagai format dokumen termasuk HTML, PDF, Excel, RTF, ODT, OpenOffice, XML file dan lain-lain.

2.9.4 Visual Paradigm

Visual Paradigm sebuah *software* model dengan system visualisasi yang memungkinkan model yang telah dibuat dapat digunakan sebagai representasi proyek-proyek lain dilengkapi dengan beberapa fitur yang ada didalmnya sampai

pada menganalisa sebuah proyek yang akan dikerjakan. Diagram dapat disusun sedemikian rupa sehingga dapat dipustakakan menjadi proyek per proyek yang saling berkaitan. Hal ini dapat juga membantu memisahkan terhadap pekerjaan proyek sampai level terkecil.

Visual Paradigm dapat membandingkan perubahan antara diagram yang satu dengan yang lain yang saling berkaitan dan dapat digunakan untuk melacak perbedaan antara diagram-diagram yang ada. Dengan dukungan *design* layaknya *Freehand* dapat menambahkan ke diagram untuk penjelasan lebih detail apa yang menjadi tujuan. Fasilitas lain dapat menggunakan *mouse pen* atau pena untuk *sketching* bentuk jalan, kita juga dapat menambahkan bentuk solid seperti bintang, dan panah *rectangles*.