

BAB II

LANDASAN TEORI

2.1 Definisi Sistem Informasi

Menurut **Satzinger, Jackson, dan Burd (2012:6)**, Sistem informasi merupakan kumpulan dari komponen-komponen yang mengumpulkan, memproses, menyimpan, dan menyediakan output dari setiap informasi yang dibutuhkan dalam proses bisnis serta aplikasi yang digunakan melalui perangkat lunak, database dan bahkan proses manual yang terkait.

Secara garis besar, sistem informasi bisa diartikan sebagai sistem yang saling terintegrasi satu sama lain secara penuh atau optimal sehingga pengolahan, penyimpanan, pengelolaan, pemrosesan dan penyajian informasi suatu perusahaan atau organisasi dapat tersaji dalam berbagai jenis informasi yang akurat sehingga nantinya dapat dijadikan sebagai acuan penentu keputusan guna berhasil mencapai tujuan yang telah disepakati bersama.

2.1.1 Ciri-Ciri Sistem Informasi

Ciri-ciri sistem informasi adalah sebagai berikut:

1. Satu kesatuan : Satu-Kesatuan organisasi.
2. Bagian-bagian : Ada manajemen, karyawan, pemangku kepentingan (stakeholder) lainnya, gedung kantor, sub-sistem komputer (perangkat keras, perangkat lunak, perangkat jaringan, sumber daya manusia, basis data dan informasi).
3. Terjalin erat : Tercermin dalam bentuk hubungan, interaksi, prosedur kerja antar manajemen.

4. Mencapai tujuan : Menghasilkan informasi yang berkualitas dari manajemen dan pemicu kepentingan lainnya.

2.2 Karakteristik Sistem Informasi

Suatu sistem mempunyai karakteristik atau sifat-sifat tertentu, yaitu mempunyai komponen sistem (*component*) batas sistem (*boundary*) lingkungan luar sistem (*environment*) penghubung (*interface*) masukan (*input*) pengolahan (*process*) keluaran (*output*) dan sasaran (*objective*) atau tujuan (*goals*).

Berikut adalah penjelasan dari karakteristik sistem :

a. Komponen Sistem

Komponen sistem (*Component*) yaitu dari sejumlah komponen yang saling berinteraksi yang artinya saling bekerja sama membentuk satu kesatuan. Elemen-elemen sistem dapat berupa suatu subsistem atau bagian-bagian dari sistem. Setiap subsistem mempunyai sistem untuk menjalankan suatu fungsi tertentu dan mempengaruhi proses sistem secara keseluruhan, setiap sistem tidak peduli betapa kecilnya pun yang mengandung sub-sub sistem.

b. Batas Sistem

Batas Sistem (*Boundary*) yaitu daerah yang membatasi suatu sistem dengan sistem lainnya atau dengan lingkungan luarnya. Batas sistem ini memungkinkan suatu sistem dipandang.

c. Lingkungan Luar Sistem

Lingkungan luar sistem (*Environment*) dari suatu sistem yaitu apapun di luar batas dari sistem yang mempengaruhi operasi sistem. Lingkungan luar sistem dapat bersifat menguntungkan dan dapat juga merugikan sistem tersebut.

d. Penghubung

Penghubung sistem (*Interface*) merupakan media penghubung antara satu sub sistem dengan subsistem lainnya, sumber daya-sumber daya mengalir dari subsistem ke subsistem yang lainnya. Dengan penghubung satu subsistem dapat berintegrasi dengan subsistem yang lainnya membentuk satu kesatuan.

e. Masukan

Masukan (*Input*) merupakan energi yang dimasukkan ke dalam sistem. Masukan dapat berupa masukan perawatan (*maintenance input*) dan masukan sinyal (*signal input*).

f. Keluaran

Keluaran (*Output*) Merupakan hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna dan sisa pembuangan. Keluaran dapat merupakan masukan untuk subsistem yang lain atau kepada supra sistem.

g. Pengolahan

Pengolahan (*Process*) yaitu suatu sistem dapat mempunyai suatu bagian pengolah yang akan mengubah masukan menjadi keluaran.

h. Sasaran (*Objectives*) dan Tujuan (*Goal*)

Suatu sistem dikatakan berhasil apabila mengenai sasaran atau tujuannya. Jika suatu sistem tidak mempunyai sasaran, maka operasi sistem tidak akan ada gunanya. Penulis menyimpulkan bahwa sistem adalah kumpulan bagian-bagian atau subsistem-subsistem yang disatukan dan didesain untuk mencapai suatu tujuan.

2.3. Pengertian Sistem Pendukung Keputusan

Sistem Pendukung Keputusan (SPK) pertama kali diungkapkan pada awal tahun 1970-an oleh Michael S. Scott Morton dengan istilah Management Decision System. Sistem tersebut adalah suatu sistem yang berbasis komputer yang ditunjukan untuk membantu pengambil keputusan dengan memanfaatkan data dan model tertentu untuk memecahkan berbagai persoalan yang tidak terstruktur (Dewanto, 2015). Istilah SPK mengacu pada suatu sistem yang memanfaatkan dukungan komputer dalam proses pengambilan keputusan. Untuk memberikan pengertian yang lebih maka ada beberapa definisi mengenai SPK oleh beberapa ahli.

Menurut Turban, Sistem Pendukung Keputusan (SPK) merupakan system informasi yang berbasis komputer yang fleksibel, interaktif dan dapat diadaptasi, yang dikembangkan untuk mendukung solusi untuk masalah manajemen spesifik yang tidak terstruktur. Sistem Pendukung Keputusan menggunakan data, memberikan antarmuka pengguna yang mudah dan dapat menggabungkan pemikiran pengambilan keputusan (Turban, Sharda & Delen, 2011).

2.3.1 Tujuan Sistem Pendukung Keputusan

Tujuan sistem pendukung keputusan adalah sebagai berikut :

1. Merancang, implementasi dan pengujian terhadap Sistem Pendukung Keputusan Penentuan Penerima Bantuan Program Pemerintah.
2. Studi kepustakaan yaitu penelusuran informasi kepustakaan baik mengenai Sistem Pendukung Pengambilan Keputusan maupun peraturan-peraturan pemerintah yang terkait.
3. Wawancara dan observasi, mencari dan mengumpulkan data-data yang ada relevansinya dengan penelitian ini pada instansi yang terkait.
4. Pemodelan matematis atau pemodelan kuantitatif, yaitu pemodelan dengan memberi bobot angka terhadap kriteria-kriteria penilaian yang spesifik untuk penentuan penerima bantuan Program Pemerintah.

2.4.2 Kelebihan Sistem Pendukung Keputusan (SPK)

Kelebihan Sistem Pendukung Keputusan (SPK) yaitu Dapat memperluas kemampuan pengambil keputusan, menghemat waktu yang dibutuhkan untuk memecahkan masalah, menghasilkan solusi dengan lebih cepat, mampu memberikan berbagai alternatif dalam pengambilan keputusan, memperkuat keyakinan pengambil keputusan dan memberikan keuntungan kompetitif bagi organisasi secara keseluruhan.

2.4.3 Kekurangan Sistem Pendukung keputusan (SPK)

Walaupun dirancang dengan sangat teliti dan mempertimbangkan seluruh faktor yang ada, Sistem pendukung Keputusan (SPK) mempunyai kekurangan atau keterbatasan diantaranya yaitu ada beberapa kemampuan manajemen dan bakat manusia yang tidak dapat dimodelkan, SPK terbatas untuk memberikan

alternatif dari pengetahuan yang diberikan, proses-proses yang dapat dilakukan oleh SPK tergantung pada kemampuan perangkat lunak yang digunakan, harus selalu diadakan perubahan secara kontinyu dan SPK dirancang untuk membantu / mendukung pengambilan keputusan dengan mengolah informasi dan data yang diperlukan dan bukan untuk mengambil alih pengambilan keputusan.

2.5 Definisi Metode AHP

Analytical Hierarchy Process merupakan suatu metode pendukung keputusan yang dikembangkan oleh Thomas L. Saaty. Model pendukung keputusan ini akan menguraikan masalah multi faktor atau multi kriteria yang kompleks menjadi suatu hierarki (susunan). Menurut Saaty hierarki didefinisikan sebagai suatu representasi dari sebuah permasalahan yang kompleks dalam suatu struktur multilevel dimana level pertama adalah tujuan, yang diikuti level faktor, kriteria, sub kriteria, dan seterusnya hingga level terakhir dari alternatif.

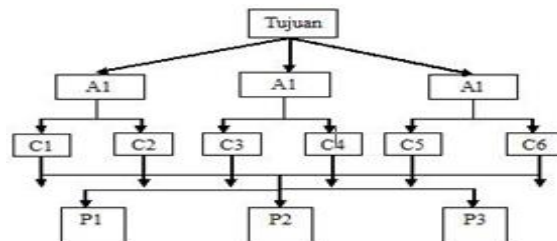
Analytical Hierarchy Process digunakan sebagai metode pemecahan masalah dibanding dengan metode yang lain karena alasan-alasan berikut:

1. Struktur yang berhierarki, sebagai konsekuensi dari kriteria yang dipilih, sampai pada sub kriteria yang paling dalam.
2. Memperhitungkan validitas sampai dengan batas toleransi inkonsistensi sebagai kriteria dan alternatif yang dipilih oleh pengambil keputusan.

2.5.1. Langkah Penyelesaian Metode Analisis Hierarki Proses (AHP)

Menurut Sri Eniyati (2010 : 20) pada Analisis Hierarki Proses terjadi penyusunan permasalahan kedalam suatu struktur hirarki sehingga pengambilan

keputusan semaksimal mungkin dapat melibatkan semua faktor yang perlu dipertimbangkan dan akan terlihat jelas kaitan antara faktor yang satu dengan yang lain. Susunan model AHP ditunjukkan pada gambar dibawah ini:



Gambar 2.1 Susunan Hierarki

Untuk menilai perbandingan tingkat kepentingan elemen, Saaty (1994) menetapkan skala kuantitatif 1 sampai 9. Nilai dan definisi pendapat kualitatif dari skala perbandingan Saaty bisa diukur menggunakan table analisis pada tabel berikut:

Tabel 2.1 Skala Perbandingan Hierarki

Tingkat Kepentingan	Definisi	Keterangan
1	Sama pentingnya	Kedua elemen mempunyai pengaruh yang sama.
3	Agak lebih penting yang satu atas lainnya.	Pengalaman dan penilaian sangat memihak satu elemen dibandingkan dengan pasangannya.
5	Cukup penting	Pengalaman dan keputusan menunjukkan kesukaan atas satu aktifitas lebih dari yang lain.
7	Sangat penting	Pengalaman dan keputusan menunjukkan kesukaan yang kuat atas satu aktifitas lebih dari yang lain.
9	Mutlak lebih penting	Satu elemen mutlak lebih disukai dibandingkan dengan pasangannya, pada tingkat

		keyakinan tertinggi.
2,4,6,8	Nilai tengah diantara dua nilai berdekatan	Bila kompromi dibutuhkan

Langkah-langkah dalam metode AHP meliputi :

1. Mendefinisikan masalah dan menentukan solusi yang diinginkan, lalu menyusun hierarki dari permasalahan yang dihadapi.
2. Menentukan prioritas elemen
 - a. Langkah pertama dalam menentukan prioritas elemen adalah membuat perbandingan pasangan, yaitu membandingkan elemen secara berpasangan sesuai kriteria yang diberikan.
 - b. Matriks perbandingan berpasangan diisi menggunakan bilangan untuk merepresentasikan kepentingan relatif dari suatu elemen terhadap elemen yang lainnya.
3. Sintesis

Pertimbangan-pertimbangan terhadap perbandingan berpasangan disintesis untuk memperoleh keseluruhan prioritas. Hal-hal yang dilakukan dalam langkah ini adalah:

 - a. Menjumlahkan nilai-nilai dari setiap kolom pada matriks
 - b. Membagi setiap nilai dari kolom dengan total kolom yang bersangkutan untuk memperoleh normalisasi matriks.
 - c. Menjumlahkan nilai-nilai dari setiap baris dan membaginya dengan jumlah elemen untuk mendapatkan nilai rata-rata.
4. Mengukur Konsistensi

Dalam pembuatan keputusan, penting untuk mengetahui seberapa baik konsistensi yang ada. Hal-hal yang dilakukan dalam langkah ini adalah

sebagai berikut:

- a. Kalikan setiap nilai pada kolom pertama dengan prioritas relatif elemen pertama, nilai pada kolom kedua dengan prioritas relatif elemen kedua dan seterusnya.
- b. Jumlahkan setiap baris
- c. Hasil dari penjumlahan baris dibagi dengan elemen prioritas relatif yang bersangkutan
- d. Jumlahkan hasil bagi di atas dengan banyaknya elemen yang ada, hasilnya disebut λ maks

5. Melakukan penghitungan *Consistency Index* (CI) dengan rumus:

$$CI = (\lambda_{\max} - n) / n \quad (1)$$

Dimana n = banyaknya elemen.

6. Melakukan penghitungan Rasio Konsistensi/*Consistency*

Ratio (CR) dengan rumus:

$$CR = CI/IR \quad (2)$$

Dimana CR = *Consistency Ratio*

CI = *Consistency Index*

IR = *Indeks Random Consistency*

2. Memeriksa konsistensi hierarki. Jika nilainya lebih dari 10%, maka penilaian data *judgment* harus diperbaiki. Namun jika Rasio Konsistensi (CI/CR) kurang atau sama dengan 0,1, maka hasil perhitungan bisa dinyatakan benar, dimana nilai RI atau *random index*.

2.5.2 Kelebihan metode *Analitycal Hierarchy Process* (AHP)

Kelebihan dari model AHP dibandingkan dengan model pengambilan keputusan yang lain terletak pada kemampuannya untuk memecahkan masalah *multiobjectives* dengan multikriteria. Kebanyakan model yang sudah ada memakai *single objectives* dengan multikriteria. Model *Linear Programming* misalnya, memakai suatu tujuan dengan banyak kendala (kriteria). Kelebihan model AHP ini lebih disebabkan oleh fleksibilitasnya yang tinggi terutama dalam pembuatan hirarki. Sifat fleksibel tersebut membuat model AHP dapat menangkap beberapa tujuan dan beberapa kriteria sekaligus dalam sebuah model atau sebuah hirarki.

2.5.3 Kelemahan metode *Analitycal Hierarchy Process* (AHP)

Disamping kelebihan-kelebihan yang dimilikinya, model AHP juga mempunyai beberapa kelemahan. Ketergantungan model ini terhadap input berupa persepsi seorang ahli akan membuat hasil akhir dari model ini menjadi tidak ada artinya apabila si ahli memberikan penilaian yang keliru. Kebanyakan orang bertanya apakah persepsi dari seorang ahli tersebut dapat mewakili kepentingan orang banyak atau tidak.

2.6. Pengertian Penerimaan Keluarga Harapan (PKH)

2.6.1 Studi kasus

Program keluarga harapan dilakukan di Kelurahan Helvetia Tengah Medan dengan tujuan menyelesaikan masalah yang timbul akibat bantuan-bantuan yang akan diterima oleh masyarakat di wilayah tersebut yang kurang tepat dan akurat. Aplikasi akan ditempatkan di Kelurahan dan aplikasi yang akan digunakan akan dikelola oleh hanya 1 orang yaitu Admin. Penerapan metode

AHP sangat dibutuhkan dalam aplikasi yang akan digunakan serta mendapatkan keputusan yang akurat dan tepat.

2.6.2 Warga

Warga mengandung arti peserta anggota warga dalam suatu organisasi atau perkumpulan. Ada istilah rakyat penduduk dan warga negara. Rakyat lebih merupakan konsep politis. Rakyat menunjukkan kepada orang-orang yang berada dibawah satu pemerintahan dan tunduk pada pemerintahan itu. Istilah warga atau rakyat umumnya dilawankan dengan pengusaha.

2.6.3 Keluarga Harapan

Program Keluarga Harapan yang selanjutnya disebut PKH adalah program pemberian bantuan sosial bersyarat kepada Keluarga Penerima Manfaat (KPM) yang ditetapkan sebagai keluarga penerima manfaat PKH. Sebagai upaya percepatan penanggulangan kemiskinan, sejak tahun 2007 Pemerintah Indonesia telah melaksanakan PKH. Program Perlindungan Sosial yang juga dikenal di dunia internasional dengan istilah *Conditional Cash Transfers (CCT)* ini terbukti cukup berhasil dalam menanggulangi kemiskinan yang dihadapi negara.

2.7 Teori Unified Modelling Language (UML)

Unified Modelling Language (UML) adalah sebuah "bahasa" yg telah menjadi standar dalam industri untuk visualisasi, merancang dan mendokumentasikan sistem piranti lunak. UML menawarkan sebuah standar untuk merancang model sebuah sistem.

Teori Unified Modelling Language (UML) adalah bahasa untuk menspesifikasi, memvisualisasi, membangun dan mendokumentasikan artifacts (bagian dari informasi yang digunakan atau dihasilkan oleh proses pembuatan

perangkat lunak, artifacts tersebut dapat berupa model, deskripsi atau perangkat lunak) dari sistem perangkat lunak, seperti pada pemodelan bisnis dan sistem non perangkat lunak lainnya.

untuk mendapatkan banyak pandangan terhadap sistem informasi yang akan dibangun, UML menyediakan beberapa diagram visual yang menunjukkan berbagai aspek dalam sistem. Ada beberapa diagram yang disediakan dalam UML antara lain:

1. Diagram use case (use case diagram)
2. Diagram aktivitas (activity diagram)
3. Flowchart
4. Entity Relationship Diagram (ERD)

2.7.1 Diagram Use Case (Use Case diagram)

Use Case adalah kegiatan atau urutan interaksi yang saling berkaitan antara sistem dan aktor. *Use Case* bekerja dengan cara mendeskripsikan tipe interaksi antara *user* sebuah sistem dengan sistemnya sendiri melalui sebuah cerita bagaimana sebuah sistem dipakai. *User Case* juga digunakan untuk membentuk perilaku (behaviour) sistem yang akan dibuat. Sebuah *Use Case* menggambarkan sebuah interaksi antara pengguna (aktor) dengan sistem yang akan dibuat (Al-Fatih,M, 2012).

2.7.1.1 Tujuan *Use Case*

Tujuan *Use Case* yang dibutuhkan antara lain sebagai berikut :

- a. Memetakan kebutuhan system.
- b. Merepresentasikan interaksi pengguna terhadap sistem.
- c. Untuk mengetahui kebutuhan diluar system.

Ada beberapa komponen yang menyusun *Use-Case* diagram yaitu:

1. Actor

Actor adalah seseorang atau sesuatu (misalnya perangkat atau sistem lain) yang berinteraksi secara langsung dengan sistem. Singkatnya: segala sesuatu yang berada di luar sistem namun berinteraksi secara langsung dengan sistem, disebut sebagai Actor. Actor bisa hanya memberikan informasi kepada sistem atau menerima informasi dari sistem atau bisa juga keduanya. Actor biasa digambarkan dengan Stick Man.

2. *Use Case*

Use Case merupakan komponen dalam use-case diagram yang menggambarkan fungsionalitas sebuah sistem atau bahasa mudahnya: apa yang bisa dilakukan sistem. Ada beberapa cara untuk menentukan *Use Case* dalam suatu sistem:

1. Perilaku sistem.
2. Hal yang bisa dilakukan oleh aktor dalam sistem itu.
3. Apa yang dikerjakan oleh perangkat lunak.
4. *Use Case* digambarkan sebagai oval dengan berisi keterangan.

3. Relasi

Relasi adalah komponen yang menyatakan hubungan yang terjadi antara satu komponen dan komponen lain, seperti antara actor dengan *Use Case* dan sebaliknya atau antara *Use Case* dengan *Use Case* lainnya.

Ada beberapa relasi yang terdapat pada *Use Case* diagram:





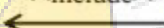
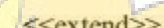
1. *Association*, menghubungkan link antar elemen.
2. *Generalization*, disebut juga dengan *inheritance* (pewarisan), dimana sebuah elemen dapat merupakan spesialisasi dari elemen lainnya.
3. *Dependency*, dimana sebuah elemen bergantung pada elemen lainnya.
4. *Aggregation*, sebuah bentuk asosiasi dimana sebuah elemen berisi elemen lainnya.

Tipe relasi yang mungkin terjadi pada *Use Case* diagram:

- a. <<include>> relasi ke sebuah *Use Case* dimana *Use Case* yang ditambahkan memerlukan *Use Case* ini untuk menjalankan fungsinya.
- b. <<extend>> relasi ke sebuah *Use Case* dimana *Use Case* yang ditambahkan dapat berdiri sendiri walau tanpa *Use Case* tambahan itu.

Adapun simbol-simbol *Use Case Diagram* dapat dilihat pada tabel 2.3 sebagai berikut:






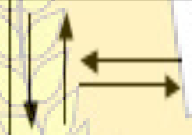
Tabel 2.2 Simbol-simbol *Use Case Diagram*

Simbol	Keterangan
	Aktor : Mewakili peran orang, sistem yang lain, atau alat ketika berkomunikasi dengan <i>use case</i>
	<i>Use case</i> : Abstraksi dan interaksi antara sistem dan aktor
	<i>Association</i> : Abstraksi dari penghubung antara aktor dengan <i>use case</i>
	<i>Generalisasi</i> : Menunjukkan spesialisasi aktor untuk dapat berpartisipasi dengan <i>use case</i>
	Menunjukkan bahwa suatu <i>use case</i> seluruhnya merupakan fungsionalitas dari <i>use case</i> lainnya
	Menunjukkan bahwa suatu <i>use case</i> merupakan tambahan fungsional dari <i>use case</i> lainnya jika suatu kondisi terpenuhi

2.7.2 Activity Diagram

Activity Diagram merupakan alur kerja (*workflow*) atau kegiatan (aktivitas) dari sebuah sistem atau menu yang ada pada perangkat lunak. *Activity Diagram* juga digunakan untuk mendefinisikan urutan atau pengelompokan tampilan dari sistem/*user interface* dimana setiap aktivitas dianggap memiliki sebuah rancangan antar muka tampilan serta rancang menu yang ditampilkan pada perangkat lunak.

Tabel 2.3 simbol–simbol Activity Diagram.

NO	GAMBAR	NAMA	KETERANGAN
1		Activity	Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain
2		Action	State dari sistem yang mencerminkan eksekusi dari suatu aksi
3		Initial Node	Bagaimana objek dibentuk atau diawali.
4		Activity Final Node	Bagaimana objek dibentuk dan diakhiri
5		Decision	Digunakan untuk menggambarkan suatu keputusan / tindakan yang harus diambil pada kondisi tertentu
6		Line Connector	Digunakan untuk menghubungkan satu simbol dengan simbol lainnya

2.7.3 Flowchart

Pengertian Flowchart (Diagram Alur).

Menurut Indrajani (2011:22), *Flowchart* merupakan penggambaran secara grafik dari langkah-langkah dan urutan prosedur suatu program. Biasanya mempengaruhi penyelesaian masalah yang khususnya perlu dipelajari dan dievaluasi lebih lanjut.

Flowchart di bedakan menjadi 5 jenis *flowchart*, antara lain *system flowchart*, *document flowchart*, *schematic flowchart*, *program flowchart*, *process flowchart*.

Masing-masing jenis *flowchart* akan dijelaskan berikut ini :

a. *System Flowchart*

System Flowchart dapat didefinisikan sebagai bagan yang menunjukkan arus pekerjaan secara keseluruhan dari sistem. Bagan ini menjelaskan urutan-urutan dari prosedur-prosedur yang ada di dalam sistem. Bagan alir sistem menunjukkan apa yang dikerjakan di sistem.

b. *Document Flowchart*

Bagan alir dokumen (*document flowchart*) atau disebut juga bagan alir formulir (*form flowchart*) atau *paperwork flowchart* merupakan bagan alir yang menunjukkan arus dari laporan dan formulir termasuk tembusan-tembusannya.

c. *Schematic Flowchart*

Bagan alir skematik (*schematic flowchart*) merupakan bagan alir yang mirip dengan bagan alir sistem, yaitu untuk menggambarkan prosedur didalam sistem. Perbedaanya adalah, bagan alir skematik selain menggunakan symbol-simbol bagan alir sistem, juga menggunakan gambar-gambar komputer dan peralatan lainnya yang digunakan. Maksud penggunaan gambar-gambar ini adalah untuk memudahkan komunikasi kepada orang yang kurang paham dengan simbol-simbol bagan alir. Penggunaan gambar-gambar ini memudahkan untuk dipahami, tetapi sulit dan lama menggambaranya.

d. *Program Flowchart*

Bagan alir program (*program flowchart*) merupakan bagan yang menjelaskan secara rinci langkah-langkah dari proses program. Bagan alir program dibuat dari derivikasi bagan alir sistem.

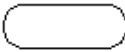


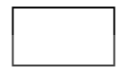





Bagan alir program dapat terdiri dari dua macam, yaitu bagan alir logika program (*program logic flowchart*) dan bagan alir program komputer terinci (*detailed computer program flowchart*). Bagan alir logika program digunakan untuk menggambarkan tiap-tiap langkah di dalam program komputer secara logika. Bagan alat logika program ini dipersiapkan oleh analis sistem. Gambar berikut menunjukkan bagan alir logika program.

Bagan alir program komputer terinci (*detailed computer program flowchart*) digunakan untuk menggambarkan instruksi-instruksi program komputer secara terinci. Bagan alir ini dipersiapkan oleh program.

e. *Process Flowchart*

Bagan alir proses (*process flowchart*) merupakan bagan alir yang banyak digunakan di teknik industri. Bagan alir ini juga berguna bagi analis sistem untuk menggambarkan proses dalam suatu prosedur.

Tabel 2.4. Simbol-Simbol Flowchart


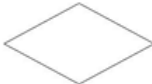


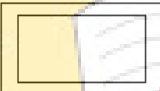
SIMBOL	NAMA	FUNGSI
	TERMINATOR	Permulaan / akhir program
	GARIS ALIR (FLOW LINE)	Arah aliran program
	PREPARATION	Proses inisialisasi/pemberian harga awal
	PROSES	Proses perhitungan/proses pengolahan data
	INPUT/OUTPUT DATA	Proses input/output data, parameter, informasi
	PREDEFINED PROCESS (SUB PROGRAM)	Permulaan sub program/proses menjalankan sub program
	DECISION	Pertandingan pernyataan, penyeleksian data yang memberikan pilihan untuk langkah selanjutnya
	ON PAGE CONNECTOR	Penghubung bagian-bagian flowchart yang berada pada satu halaman
	OFF PAGE CONNECTOR	Penghubung bagian-bagian flowchart yang berada pada halaman berbeda

2.7.4 Entity Relationship Diagram (ERD)

Model *Entity-Relationship* berisi komponen-komponen dari suatu himpunan entitas dan himpunan relasi yang masing-masing dilengkapi dengan atribut-atribut yang merepresentasikan seluruh fakta yang ditinjau sehingga dapat diketahui hubungan antara *entity-entity* yang ada dengan atribut-atributnya.

Simbol-simbol yang digunakan dalam *Entity Relationship Diagram* dijelaskan pada tabel 2.2. berikut ini :

Tabel 2.5. Simbol Entity Relationship Diagram (Diagram E-R-D)

No	Simbol	Nama	Keterangan
1		Entitas	Jenis entitas dapat berupa suatu elemen lingkungan, sumber daya atau transaksi yang <small>FIELD-FIELDnya</small> dipergunakan dalam aplikasi program
2		Hubungan atau Relasi	Menunjukkan nama relasi antar satu entitas dengan entitas lainnya
3		Atribut	Atribut adalah karakteristik dari sebuah entitas
4		Garis Relasi	Menunjukkan hubungan (keterkaitan) antar entitas
5		Entitas Lemah	Entitas yang kemunculannya tergantung dari entitas lain yang lebih kuat

Hubungan/relasi antar atribut yang terdapat pada sistem konseptual secara bebas yang terdiri dari entiti-entiti, dan setiap entiti terdiri dari atribut yang ada, yaitu :

- 1) *Unary*, adalah satu entiti berelasi hanya dengan satu entiti saja.
- 2) *Binary*, adalah satu entiti berhubungan dengan entiti yang lain.
- 3) *Ternary*, adalah satu entiti berhubungan dengan beberapa entiti yang lainnya.

2.8 Perangkat Lunak yang digunakan

2.8.1 Database

Menurut Connolly dan Bagg (2010:5) basis data (database) adalah Sebuah kumpulan data yang secara logis terkait dan dirancang untuk memenuhi suatu kebutuhan informasi dari sebuah organisasi.

Dari definisi di atas, maka dapat disimpulkan database adalah sekumpulan data informasi yang saling berhubungan untuk dapat menyimpan dan menampilkan data, mengakses informasi, menarik kesimpulan dan membuat keputusan informasi dari sebuah organisasi.

Kelebihan Sistem Database adalah sebagai berikut :

1. Kerangkapan dan inkonsistensi data dapat dikontrol sehingga tidak terdapat data rangkap.
2. Memudahkan penerapan batasan-batasan pengamanannya
3. Terpeliharanya integritas data.
4. Data dapat dipakai secara bersama-sama.

2.8.2 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; 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.

2.8.3 VB Studio 2010 Ultimate

Visual Studio 2010 Ultimate adalah sebuah bahasa pemrograman komputer. Dimana pengertian dari bahasa pemrograman itu adalah perintah-perintah atau instruksi yang dimengerti oleh komputer untuk melakukan tugas-tugas tertentu. Visual Studio 2010 selain disebut dengan bahasa pemrograman, juga sering disebut sebagai sarana (tool) untuk menghasilkan program-program aplikasi berbasis windows.

