

BAB II

TINJAUAN PUSTAKA

II.1. Penelitian Terkait

Berdasarkan penelitian dari Fahmi Setiawan (2015) dengan judul “Implementasi Metode Electre Pada Sistem Pendukung Keputusan SNMPTN Jalur Undangan” bahwa Siswa yang berprestasi tinggi dan secara konsisten menunjukkan prestasinya tersebut layak mendapatkan kesempatan untuk menjadi calon mahasiswa melalui SNMPTN. MultiCriteria Decision Making (MCDM) merupakan salah satu metode pengambilan keputusan untuk menyeleksi alternatif terbaik dari sejumlah alternatif berdasarkan kriteria tertentu. Salahsatu metode dalam MCDM adalah Elimination Et Choix Traduisant la Realite (ELECTRE). Metode Electre merupakan metode pengambilan keputusan multikriteria berdasarkan pada konsep Outranking dengan menggunakan perbandingan berpasangandari alternatif-alternatif berdasarkan setiapkriteria yang sesuai. Tujuan dari penelitian ini adalah menerapkan metode Electredan Algoritma Greedy dalam pengambilan keputusan SNMPTN jalur undangan.

Berdasarkan penelitian dari Benni Satria (2019) dengan judul “ Penerapan Metode Electre Sebagai Sistem Pendukung Keputusan Dalam Penerimaan Beasiswa” Sistem Pendukung Keputusan (SPK) dapat menghasilkan keputusan siswa mana saja yang layak untuk menerima beasiswa tersebut berdasarkan kriteria yang sudah ditetapkan sebelumnya. Electre merupakan salah satu algoritma sistem pendukung keputusan yang cocok digunakan, karena algoritma

ini bekerja dengan cara memproses bobot nilai masing-masing kriteria dan calon penerima beasiswa dan menghasilkan nilai mutlak yang mudah digunakan untuk mengambil keputusan. Penelitian ini memanfaatkan algoritma Electre dalam bentuk sebuah sistem pendukung keputusan penerimaan beasiswa, yang menghasilkan sistem komputerisasi yang dapat secara otomatis menentukan siswa yang layak menerima beasiswa, berdasarkan nilai-nilai kriteria yang diinputkan. Sistem pendukung keputusan ini dirancang menggunakan bahasa pemrograman Microsoft Visual Basic .Net 2010.

Berdasarkan penelitian dari Sutirno (2017) dengan judul “Sistem Pendukung Keputusan Metode Electre Pada Bauran Pemasaran (7p) Dalam Memulai Usaha Jasa Center” bahwa untuk menjalankan usaha, modal yang paling penting adalah modal non-materi seperti ilmu manajemen pemasaran dengan teori bauran pemasaran (*marketing mix*). Alat pemasaran tersebut dikenal dengan istilah “4P” dan dikembangkan menjadi “7P” , yaitu : Produk (*product*) , Harga (*price*) , Tempat/distribusi (*place*), Promosi (*promotion*), Orang (*people*), Bukti fisik (*physical evidence*) dan Proses (*process*). Berdasarkan permasalahan tersebut dilakukan penelitian untuk merancang sistem pendukung keputusan yang bertujuan untuk membantu calon pengusaha dalam menentukan jenis usaha berdasarkan kriteria 7P. Metode yang digunakan dalam penelitian ini adalah metode Electre (*Elimination and Choice Translation Reality*). Hasil dari penelitian ini berupa sistem pendukung keputusan penentuan jenis usaha berdasarkan tujuh kriteria (7P) dalam menentukan jenis usaha.

Berdasarkan penelitian dari Suparmaji (2017) dengan judul “Sistem Rental Mobil Di Kota Bengkulu Dengan Metode *Electre Iv* Dalam Membuat Keputusan Pemilihan Mobil Rental Berbasis *Website*” Hasil dari penelitian ini adalah sebuah sistem rental mobil yang mampu menerapkan metode *Electre IV* dalam membuat keputusan pemilihan mobil rental dengan disertai fitur pengolahan data. Pengujian *white box* dilakukan dengan metode *basis path testing* pada algoritma *Electre IV* dan menghasilkan 5 jalur/*path* berbeda, dengan hasil setiap kondisi berhasil dijalankan tanpa ada kesalahan. Pengujian *black box* dilakukan terhadap 12 kelas uji dan 60 skenario pengujian yang diujikan dengan hasil 60 skenario uji berhasil 100%, dan berjalan sebagaimana mestinya. Pengujian akurasi dilakukan untuk mengukur tingkat akurasi metode *Electre IV* dalam membuat keputusan pemilihan mobil rental, dengan tingkat akurasi 85% metode ini dianggap dapat memberikan rekomendasi mobil rental yang sesuai dengan kebutuhan *user*.

II.2. Uraian Teoritis

II.2.1. Sistem

Sistem merupakan sekumpulan unsur atau komponen dan prosedur yang harus berhubungan erat (*interrelated*) satu sama lain dan berfungsi secara bersama-sama agar tujuan yang sama (*common purpose*) dapat dicapai. Setiap sistem dibuat untuk mencapai suatu tujuan (*goal*) atau sasaran (*objectives*). Tujuan (*goal*) meliputi ruang lingkup yang luas, sedangkan sasaran (*objectives*) lebih dikenal pada sub sistemnya karena meliputi ruang lingkup yang sempit dibanding tujuan. Tujuan maupun sasaran pada prinsipnya adalah sesuatu yang hendak diraih untuk

memenuhi kebutuhan (*need*) dan keinginan (*wish*) serta sifatnya senantiasa positif (lebih mengarah pada kebaikan), sehingga alat ukur tercapainya suatu tujuan/sasaran adalah kepuasan (*satisfaction*) (Mei Hotma Mariati : 2016).

II.2.2. Informasi

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

1. Akurat (*Accurate*)

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

2. Tepat waktu (*Timelines*)

Berarti informasi yang datang pada si penerima tidak boleh terlambat. Informasi yang sudah usang tidak akan mempunyai nilai logika karena informasi merupakan landasan dalam pengambilan keputusan.

3. Relevan (*Relevance*)

Berarti informasi tersebut bermanfaat bagi pemakainya (Deppi Linda, 2016 : 62-63).

II.2.3. Sistem Informasi

Sistem informasi adalah sejumlah komponen (manusia, komputer, teknologi informasi, dan prosedur kerja), ada sesuatu yang diproses (data menjadi informasi), dan dimaksudkan untuk mencapai suatu sasaran atau

tujuan. Penggunaan sistem informasi telah banyak diterapkan diberbagai bidang termasuk dalam bisnis. Salah satu tujuan penerapan sistem informasi dalam bidang bisnis agar dapat meningkatkan keuntungan bisnis dengan menggunakan kemampuan yang didapatkan dari sistem informasi. Ada beberapa kemampuan dari sistem informasi yang dapat mendukung dalam bidang bisnis. Kemampuan tersebut seperti pengurangan biaya, mempercepat pekerjaan, dapat meningkatkan kemudahan dalam pengambilan keputusan, dan peningkatan pelayanan terhadap pelanggan. (Alfian Nurlifa : 2017)

II.2.4. Konsep Sistem Informasi

Faktor-faktor yang menentukan kehandalan dari suatu sistem informasi atau informasi dapat dikatakan baik jika memenuhi kriteria-kriteria sebagai berikut :

a) Keunggulan (*Usefulness*)

Yaitu suatu sistem yang harus dapat menghasilkan informasi yang tepat dan relevan untuk mengambil keputusan manajemen dan personil operasi dalam organisasi.

b) Ekonomis

Kemampuan sistem yang mempengaruhi sistem harus bernilai manfaat minimal, sebesar biayanya.

c) Kehandalan (*Reliability*)

Keluaran dari sistem harus mempunyai tingkat ketelitian tinggi dan sistem tersebut harus beroperasi secara efektif.

d) Pelayanan (*Customer Service*)

Yakni suatu sistem memberikan pelayanan yang baik dan efisien kepada para pengguna sistem pada saat berhubungan dengan organisasi.

e) Kapasitas (*Capacity*)

Setiap sistem harus mempunyai kapasitas yang memadai untuk menangani setiap periode sesuai yang dibutuhkan.

f) Sederhana dalam kemudahan (*Simplicity*)

Sistem tersebut lebih sederhana (umum) sehingga struktur dan operasinya dapat dengan mudah dimengerti dan prosedur mudah diikuti.

g) Fleksibel (*Flexibility*)

Sistem informasi ini harus dapat digunakan dalam kondisi yang bagaimana yang diinginkan oleh organisasi tersebut atau pengguna tertentu.

h) Komponen Sistem Informasi

Istilah dalam komponen sistem informasi adalah blok bangunan (*building block*) yang dapat dibagi menjadi enam blok yaitu :

a. Blok masukan (*Input block*)

Blok *input* merupakan data–data yang masuk ke dalam sistem informasi, yang dapat berupa *document-document* dasar yang dapat diolah menjadi suatu informasi tertentu.

b. Blok model (*Model block*)

Blok ini terdiri dari kombinasi prosedur, logika dan model matematik yang akan mengolah data *input* untuk menghasilkan suatu informasi yang dibutuhkan.

c. Blok keluaran (*Output block*)

Merupakan informasi yang menghasilkan sekumpulan data yang nantinya akan disimpan berupa data cetak laporan.

d. Blok teknologi (*Technologi block*)

Blok teknologi merupakan penunjang utama dalam berlangsungnya sistem informasi. Yang memiliki beberapa komponen yaitu diantaranya alat memasukkan data (*input device*), alat untuk menyimpan dan mengakses data (*storege device*), alat untuk menghasilkan dan mengirimkan keluaran (*output divice*) dan alat untuk membentuk pengendalian sistem secara keseluruhan (*control device*). Teknologi informasi terdiri dari 3 (tiga) bagian utama, yaitu teknisi (*humanware* atau *braiware*), perangkat lunak (*software*), dan perangkat keras (*hardware*).

e. Blok basis data (*Database block*)

Basis data merupakan kumpulan dari data yang saling berhubungan satu dengan yang lainnya, tersimpan di perangkat keras komputer dan digunakan perangkat lunak untuk memanipulasinya. Data perlu di simpan dan perlu di organisasi sedemikian rupa, supaya informasi yang dihasilkan berkualitas.

f. Blok kendali (*Control block*)

Beberapa pengendalian perlu dirancang dan diterapkan untuk meyakinkan bahwa hal-hal yang dapat merusak sistem dapat di cegah bila terlanjur terjadi. (I Made Budi Adnyana, 2016)

II.2.5. Metode *Elimination Et Choix Traduisant la Realite (ELECTRE)*

Electre merupakan salah satu metode pengambilan keputusan multikriteria berdasarkan pada konsep *Outranking* dengan menggunakan perbandingan berpasangan dari alternatif-alternatif berdasarkan setiap kriteria yang sesuai. Metode *Electre* digunakan pada kondisi dimana alternatif yang kurang sesuai dengan kriteria dieliminasi, dan alternatif yang sesuai dapat dihasilkan.

Dengan kata lain, *Electre* digunakan untuk kasus-kasus dengan banyak alternatif namun hanya sedikit kriteria yang dilibatkan. Suatu alternatif dikatakan mendominasi alternatif yang lainnya jika satu atau lebih kriterianya melebihi (dibandingkan dengan kriteria dari alternatif yang lain) dan sama dengan kriteria lain yang tersisa. Langkah-langkah yang dilakukan dalam penyelesaian masalah menggunakan metode *Electre* adalah sebagai berikut (Fahmi Setiawan, dkk ; 2015 : 89).

Langkah 1 : Normalisasi matrik keputusan.

Dalam prosedur ini, setiap atribut diubah menjadi nilai yang *comparable*.

Setiap normalisasi dari nilai r_{ij} dapat dilakukan dengan Rumus

$$r_{ij} = \frac{x_{ij}}{\sqrt{\sum_{i=1}^m x_{ij}^2}}, \text{ untuk } i=1,2,3,\dots,m \text{ dan } j=1,2,3,\dots,n.$$

Sehingga didapat matriks R hasil normalisasi

$$R = \begin{bmatrix} r_{11} & r_{12} & \dots & r_{1n} \\ r_{21} & r_{22} & \dots & r_{2n} \\ \dots & & & \\ r_{m1} & r_{m2} & \dots & r_{mn} \end{bmatrix}$$

R adalah matriks yang telah dinormalisasi, dimana m menyatakan alternatif, n menyatakan kriteria dan r_{ij} adalah normalisasi pengukuran pilihan dari alternatif ke- i dalam hubungannya dengan kriteria ke- j .

Langkah 2 : Pembobotan pada matrik yang telah dinormalisasi.

Setelah di normalisasi, setiap kolom dari matrik R dikalikan dengan bobot-bobot (w_j) yang ditentukan oleh pembuat keputusan. Sehingga, *weighted normalized matrix* adalah $V=RW$ yang ditulis dalam Rumus ini (Fahmi Setiawan, dkk ; 2015 : 90).

$$V = \begin{bmatrix} v_{11} & v_{12} & \dots & v_{1n} \\ v_{21} & v_{22} & \dots & v_{2n} \\ \dots & \dots & \dots & \dots \\ v_{m1} & v_{m2} & \dots & v_{mn} \end{bmatrix} = RW = \begin{bmatrix} w_1 r_{11} & w_2 r_{12} & \dots & w_n r_{1n} \\ w_1 r_{21} & w_2 r_{22} & \dots & w_n r_{2n} \\ \dots & \dots & \dots & \dots \\ w_1 r_{m1} & w_2 r_{m2} & \dots & w_n r_{mn} \end{bmatrix}$$

dimana W adalah

$$W = \begin{bmatrix} w_1 & 0 & 0 & \dots & 0 \\ 0 & w_2 & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & w_n \end{bmatrix}, \text{ dan } \sum_{i=1}^n w_i = 1$$

Langkah 3 : Menentukan *concordance* dan *discordance set*.

Untuk setiap pasang dari alternatif k dan l ($k, l = 1, 2, 3, \dots, m$ dan $k \neq l$) kumpulan kriteria J dibagi menjadi dua *subsets*, yaitu *concordance* dan *discordance*.

Bilamana sebuah kriteria dalam suatu alternatif termasuk *concordance* adalah :

$$C_{kl} = \{ j, y_{kj} \geq y_{lj} \}, \text{ untuk } j = 1, 2, 3, \dots, n$$

Sebaliknya, komplementer dari subset ini adalah *discordance*, yaitu bila :

$$D_{kl} = \{ j, y_{kj} < y_{lj} \}, \text{ untuk } j = 1, 2, 3, \dots, n$$

Langkah 4 : Hitung matriks *concordance* dan *discordance*.

a. *Concordance*

Untuk menentukan nilai dari elemen-elemen pada matriks *concordance* adalah dengan menjumlahkan bobot-bobot yang termasuk dalam subset *concordance*, secara matematisnya adalah pada Rumus :

$$c_{kl} = \sum_{j \in C_{kl}} w_j$$

Sehingga matrik *concordance* yang dihasilkan adalah :

$$C = \begin{bmatrix} - & c_{12} & c_{13} & \dots & c_{1n} \\ c_{21} & - & c_{23} & \dots & c_{2n} \\ \dots & & & & \\ c_{m1} & c_{m2} & c_{m3} & \dots & - \end{bmatrix}$$

b. *Discordance*

Untuk menentukan nilai dari elemen-elemen pada matriks *discordance* adalah dengan membagi maksimum selisih nilai kriteria yang termasuk dalam subset *discordance* dengan maksimum selisih nilai seluruh kriteria yang ada, secara matematisnya adalah :

$$d_{kl} = \frac{\{\max(v_{mn} - v_{mn-l_n})\}; m, n \in D_{kl}}{\{\max(v_{mn} - v_{mn-l_n})\}; m, n = 1, 2, 3, \dots}$$

Sehingga diperoleh matrik *discordance* :

$$D = \begin{bmatrix} - & d_{12} & d_{13} & \dots & d_{1m} \\ d_{21} & - & d_{23} & \dots & d_{2m} \\ \dots & & & & \\ d_{m1} & d_{m2} & d_{m3} & \dots & - \end{bmatrix}$$

Langkah 5 : Menentukan matrik dominan *concordance* dan *discordance*.

a. Concordance

Matrik dominan *concordance* dapat dibangun dengan bantuan nilai *threshold*, yaitu dengan membandingkan setiap nilai elemen matriks *concordance* dengan nilai *threshold*.

$$C_{kl} \geq c$$

dengan nilai *threshold* (c), adalah :

$$c = \frac{\sum_{k=1}^n \sum_{l=1}^n c_{kl}}{m * (m - 1)}$$

dan nilai setiap elemen matriks F sebagai matriks dominan *concordance* ditentukan sbb :

$$f_{kl} = 1, \text{ jika } c_{kl} \geq c \text{ dan } f_{kl} = 0, \text{ jika } c_{kl} < c$$

b. Discordance

Untuk membangun matriks dominan *discordance* juga menggunakan bantuan nilai *threshold*, yaitu :

$$d = \frac{\sum_{k=1}^n \sum_{l=1}^n d_{kl}}{m * (m - 1)}$$

dan nilai setiap elemen untuk matriks G sebagai matriks dominan *discordance* ditentukan sebagai berikut :

$$g_{kl} = 0, \text{ jika } c_{kl} \geq d \text{ dan } g_{kl} = 1, \text{ jika } c_{kl} < d$$

Langkah 6 : Menentukan *aggregate dominance matrix*.

Langkah selanjutnya adalah menentukan *aggregate dominance matrix* sebagai matriks E , yang setiap elemennya merupakan perkalian antara elemen matriks F dengan elemen matriks G , sebagai berikut :

$$ekl = fkl \times gkl$$

Langkah 7 : Eliminasi alternatif yang less favourable.

Matriks E memberikan urutan pilihan dari setiap alternatif, yaitu bila $ekl = 1$ maka alternatif A_k merupakan pilihan yang lebih baik daripada A_l . Sehingga baris dalam matriks E yang memiliki jumlah $ekl = 1$ paling sedikit dapat dieliminasi. Dengan demikian alternatif terbaik adalah yang mendominasi alternatif lainnya (Fahmi Setiawan, dkk ; 2015 : 92).

II.2.6. Sistem Pendukung Keputusan

Sistem Pendukung Keputusan (SPK) didefinisikan sebagai sebuah sistem yang bertujuan untuk mendukung seorang manajer mengambil sebuah keputusan dalam kondisi permasalahan yang semi terstruktur. SPK difungsikan sebagai fasilitas yang dapat memperkuat kapabilitas sang pengambil keputusan, namun tidak sepenuhnya menggantikan peran pengambil keputusan tersebut. SPK digunakan pada pengambilan keputusan yang melibatkan pertimbangan dari manajer, atau pada pengambilan keputusan yang tidak sepenuhnya dapat diselesaikan dengan perhitungan. (Ma'ruf : 2016 : 290)

II.2.6.1. Komponen Sistem Pendukung Keputusan

Komponen Sistem Pendukung Keputusan (SPK) terdiri dari:

1. Subsistem Manajemen Data (*Data-management Subsystem*).
2. Subsistem Manajemen Model (*Model Management Subsystem*).
3. Subsistem Antarmuka Pengguna (*User Interface Subsystem*).
4. Subsistem Manajemen Berbasis Pengetahuan (*Knowledge-based Management Subsystem*).

II.2.6.2. Proses Pengambilan Keputusan

Pengambilan keputusan hampir tidak merupakan pilihan antara benar dan salah, tetapi yang sering terjadi adalah pilihan yang “hampir benar” dan yang “mungkin salah”. Pengambilan keputusan di dalam suatu organisasi merupakan hasil proses komunikasi dan partisipasi yang terus menerus dari keseluruhan organisasi. Hasil keputusan tersebut dapat merupakan pernyataan yang disetujui antar alternatif atau antar prosedur untuk mencapai tujuan tertentu. Proses pengambilan keputusan melibatkan 4 tahapan, yaitu:

1. *Intelligence*, tahap ini merupakan proses penelusuran dan pendeteksian dari lingkup problematika serta proses pengenalan masalah.
2. *Design*, tahap ini merupakan proses menemukan, mengembangkan, dan menganalisis alternatif tindakan yang bisa dilakukan. Tahap ini meliputi proses untuk mengerti masalah, menurunkan solusi, dan menguji kelayakan solusi.

3. *Choice*, pada tahap ini dilakukan proses pemilihan diantara berbagai alternatif tindakan yang mungkin dijalankan. Dari tahap ini didapatkan keluaran berupa dokumen solusi dan rencana implementasinya.
4. *Implementation*, dalam tahap ini pengambilan keputusan menjalankan rangkaian aksi pemecahan yang dipilih di tahap *choice*. Implementasi yang sukses ditandai dengan terjawabnya masalah yang dihadapi, sementara kegagalan ditandai dengan tahap adanya masalah yang sedang dicoba untuk diatasi. (Ma'ruf : 2016 : 291)

II.2.7. Visual Studio 2010

Visual studio merupakan suatu perangkat lunak yang dapat digunakan untuk pengembangan berbagai macam aplikasi yang memiliki berbagai macam tipe antara lain aplikasi *desktop* (*Windows Form*, *CommandLine (Console)*), *Aplikasi Web*, *Windows Mobile (Paket PC)*. Visual Studio 2010 memiliki lebih dari satu kompiler, SDK (*Software Development Kit*), dan Dokumentasi Tutorial (*MSDN Library*). Kompiler yang dimasukkan kedalam Visual Studio 2010 antara lain Visual Basic, Visual C#, Visual C++, Visual InterDev, Visual J++, Visual F#, dan Visual Source Safe, dan banyak yang lainnya. Dan semua itu sudah terpaket dan diperuntukkan kedalam platform .Net Framework 4.0 atau versi yang lebih tinggi. Visual studio ini dapat digunakan untuk membuat aplikasi yang berbasis desktop yang merupakan platform windows, namun juga dapat dijalankan dalam bentuk Microsoft Intermediate Language diatas .Net Framework. Selain itu Visual Studio juga dapat digunakan untuk membuat aplikasi yang dapat dijalankan diatas

windows mobile yang berjalan diatas .Net Compact Framework. (Rolly Yesputra : 2017 : 1).

Visual Studio 2010 terbagi menjadi beberapa tipe diantaranya :

1. Visual Studio 2010 Express Edition yang bisa digunakan secara gratis tanpa memberikan royalti kepada Microsoft Inc.
2. Visual Studio Standard Edition
3. Visual Studio 2010 Professional Edit
4. Visual Studio 2010 Ultimate Edition (Rolly Yesputra : 2017 : 1).

II.2.8. *SQL Server 2008*

SQL Server 2008 adalah sebuah terobosan baru dari Microsoft dalam bidang *database*. SQL Server adalah DBMS (*Database Management System*) yang dibuat oleh Microsoft untuk ikut berkecimpung dalam persaingan dunia pengolahan data menyusul pendahulunya seperti IBM dan Oracle. SQL Server 2008 dibuat pada saat kemajuan dalam bidang *hardware* sedemikian pesat. Oleh karena itu sudah dapat dipastikan bahwa SQL Server 2008 membawa beberapa terobosan dalam bidang pengolahan dan penyimpanan data. Microsoft merilis SQL Server 2008 dalam beberapa versi yang disesuaikan dengan segment-segment pasar yang dituju. Versi-versi tersebut adalah sebagai berikut. Menurut cara pemrosesan data pada prosesor maka Microsoft mengelompokkan produk ini berdasarkan 2 jenis yaitu :

- a. Versi 32-bit(x86), yang biasanya digunakan untuk komputer dengan singleprosesor (Pentium 4) atau lebih tepatnya prosesor 32 bit dan sistem operasi Windows XP.
- b. Versi 64-bit(x64), yang biasanya digunakan untuk komputer dengan lebih dari satu prosesor (Misalnya Core 2 Duo) dan system operasi 64 bit seperti Windows XP 64, Vista, dan Windows 7. Sedangkan secara keseluruhan terdapat versi-versi seperti berikut ini:
- c. Versi Compact, ini adalah versi “Tipis” dari semua versi yang ada. Versi ini seperti versi desktop pada SQL Server 2000. Versi ini juga digunakan pada handheld device seperti Pocket PC, PDA, SmartPhone, Tablet PC.
- d. Versi Express, ini adalah versi “Ringan” dari semua versi yang ada (tetapi versi ini berbeda dengan versi compact) dan paling cocok untuk latihan. Express Manager standar, integrasi dengan CLR dan XML. (Agus Tinus Setiawan;2016:54).

II.2.9. Basis Data (*Database*)

Pangkalan data atau basis data (*database*) adalah kumpulan informasi yang disimpan di dalam komputer secara sistematis sehingga dapat diperiksa menggunakan suatu program komputer untuk memperoleh informasi dari basis data tersebut. Perangkat lunak yang digunakan untuk mengelola dan memanggil kueri (*query*) basis data disebut sistem manajemen basis data (*database management system*, DBMS). Sistem basis data dipelajari dalam ilmu informasi. Konsep dasar dari basis data adalah kumpulan dari catatan-catatan, atau potongan

dari pengetahuan. Sebuah basis data memiliki penjelasan terstruktur dari jenis fakta yang tersimpan di dalamnya: penjelasan ini disebut skema. Model yang umum digunakan sekarang adalah model relasional, yang mewakili semua informasi dalam bentuk tabel-tabel yang saling berhubungan dimana setiap tabel terdiri dari baris dan kolom. Model yang lain seperti model hierarkis dan model jaringan menggunakan cara yang lebih eksplisit untuk mewakili hubungan antar tabel (Neni Purwati, 2015 : 50).

II.2.10. Normalisasi

Normalisasi merupakan parameter digunakan untuk menghindari duplikasi terhadap tabel dalam basis data dan juga merupakan proses sebuah tabel yang masih memiliki beberapa anomali atau ketidak wajaran sehingga menghasilkan tabel yang lebih sederhana dan struktur yang bagus, yaitu sebuah tabel yang tidak memiliki data *redundancy* dan memungkinkan *user* untuk melakukan *insert*, *delete*, dan *update* pada baris (*recod*) tanpa menyebabkan inkonsistensi data. Tujuannya untuk menghindari beberapa anomali

1. *Insertion Anomaly* adalah proses melakukan penambahan *recod* baru akan tetapi mempengaruhi *user* untuk terjadinya duplikasi data.
2. *Deletion Anomaly* adalah proses melakukan penghapusan *record* akan tetapi akan menyebabkan hilangnya data yang akan dibutuhkan pada *record* lain.
3. *Modification Anomaly* adalah proses merubah data pada sebuah *record* mempengaruhi perubahan pada *record* lain karena adanya duplikasi.

Proses pernomalan tabel ada beberapa tahap yang harus dilakukan yaitu :

1. *First Normal Form (1 NF)*

Sudah tidak ada repeating group yaitu pengulangan yang terjadi pada beberapa atribut atau kolom dalam sebuah tabel, dan juga setiap atribut harus bernilai tunggal. Atribut *multivalued*, *composite*, *derive* tidak tunggal. Setiap nilai dari atribut hanya mempunyai nilai tunggal.

2. *Second Normal Form (2 NF)*

Untuk menjadikan tabel normal tingkat ke 2 maka sudah 1NF dan setiap atribut yang bukan *primary key* sepenuhnya secara fungsional tergantung pada semua atribut pembentuk *primary key*.

3. *Third Normal Form (3 NF)*

Tabel sudah 2NF dan tidak memiliki *transitive dependencies*, *Transitive Dependencies* adalah ketika atribut yang secara tidak langsung tergantung pada *primary key* dan atribut tersebut juga tergantung pada atribut lain yang bukan *primary key*.

4. *Boyce-codd Normal Form (BCNF)*

Tabel dalam BCNF jika sudah 3NF dan semua *determinants* adalah *candidate keys*. Perbedaan 3NF dan BCNF adalah untuk *functional dependency* $A - B$, 3NF memperbolehkan ketergantungan ada dalam relasi jika B adalah *primary key* dan A bukan merupakan *candidate key*. Sedangkan BCNF menuntut untuk ketergantungan tetap ada dalam relasi, A harus menjadi *candidate key*.

5. *Fourth Normal Form (4 NF)*

Relasi berada pada bentuk normal keempat apabila memenuhi syarat BCNF dan tidak mempunyai *multivalued dependency*.

6. *Fifth Normal Form (5 NF)*

Tabel bentuk normal kelima sering disebut PJNF (*Projection Join Normal Form*), Penyebutan PJNF karena untuk suatu relasi akan berbentuk normal kelima jika tabel tersebut dapat dipecah atau diprograsikan (kemajuan) menjadi beberapa tabel dan dari progersi (kemajuan) dapat disusun kembali (*join*) menjadi tabel yang sama dengan keadaan semula. Jika penyusunan ini tidak mungkin dilakukan dikatakan pada relasi itu terdapat *join dependencies* dan dikatakan bersifat *lossy join*.(Gandung Triyono : 2013 : 19-20)

Normalisasi adalah proses pembentukan struktur basis data sehingga sebagian besar ambiguity bias dihilangkan. Normalisasi merupakan sebuah teknik dalam logika desain sebuah basis data relasional yang mengelompokan atribut dari suatu tabel membentuk struktur tabel yang normal. Adapun kriteria tabel dikatakan normal adalah ketika tidak ada kerangkapandata (redundansi data).(Dwi Puspitasari: 2016 : 341)

Tujuan dari normalisasi adalah untuk :

- a. Untuk menghilangkan kerangkapan data sehingga meminimumkan pemakaian *storage* yang dipakai oleh *base relations (file)*
- b. Untuk mengurangi kompleksitas
- c. Untuk mempermudah pemodifikasian data

II.2.11. *Unified Modelling Language (UML)*

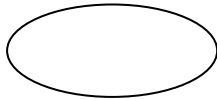
Unified Modeling Language (UML) adalah bahasa spesifikasi standar yang dipergunakan untuk mendokumentasikan, menspesifikasikan dan membangun perangkat lunak. UML merupakan metodologi dalam mengembangkan sistem berorientasi objek dan juga merupakan alat untuk mendukung pengembangan sistem. UML saat ini sangat banyak dipergunakan dalam dunia industri yang merupakan standar bahasa pemodelan umum dalam industri perangkat lunak dan pengembangan sistem. (Gellysa Urva: 2015 : 94)

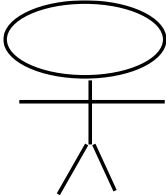


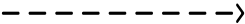
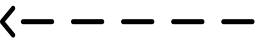
Alat bantu yang digunakan dalam perancangan berorientasi objek berbasis UML adalah sebagai berikut:

1. *Use case diagram*

Use case diagram merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih actor dengan sistem informasi yang akan dibuat. Dapat dikatakan *use case* digunakan untuk mengetahui fungsi apa saja yang berhak menggunakan fungsi-fungsi tersebut. Simbol-simbol yang digunakan dalam *use case diagram*, yaitu:

Tabel II.2. *Use Case Diagram*

Gambar	Keterangan
	<p><i>Use case</i> menggambarkan fungsionalitas yang disediakan sistem sebagai unit-unit yang bentukan pesan antar unit dengan aktor, biasanya dinyatakan dengan menggunakan kata kerja di awal nama <i>use case</i>.</p>



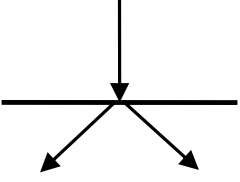
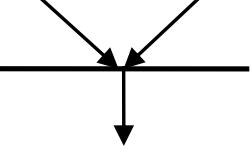
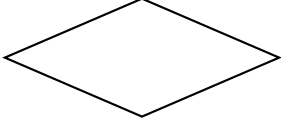

	<p>Aktor adalah <i>abstraction</i> dari orang atau sistem yang mengaktifkan fungsi dari target sistem. Untuk mengidentifikasi aktor harus ditentukan pembagian tenaga kerja dan tugas –tugas yang berkaitan dengan peran pada konteks target sistem. Orang atau sistem bias muncul dalam beberapa peran. Perlu dicatat bahwa aktor berinteraksi dengan <i>use case</i>, tetapi tidak memiliki kontrol terhadap <i>use case</i>.</p>
	<p>Asosiasi antara aktor dan <i>use case</i>, digambarkan dengan garis tanpa panah yang mengindikasikan siapa atau apa yang meminta interaksi secara langsung dan bukannya mengindikasikan aliran data.</p>
	<p>Asosiasi antara aktor dan <i>use case</i> yang menggunakan panah terbuka untuk mengindikasikan bila aktor berinteraksi secara pasif dengan sistem.</p>
	<p><i>Include</i>, merupakan di dalam <i>use case</i> lain (<i>required</i>) atau pemanggilan <i>use case</i> oleh <i>use case</i> lain, contohnya adalah pemanggilan sebuah fungsi program.</p>
	<p><i>Extend</i>, merupakan perluasan dari <i>use case</i> lain jika kondisi atau syarat terpenuhi.</p>

(Sumber : Gellysa Urva: 2015 : 94)

2. Activity diagram

Activity diagram menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis. Simbol-simbol yang digunakan dalam *activity diagram* yaitu:

Tabel II.3. *Activity Diagram*

Gambar	Keterangan
	<i>Start point</i> , diletakan pada pojok kiri atas dan merupakan awal aktifitas.
	<i>End point</i> , akhir aktifitas.
	<i>Activites</i> , menggambarkan suatu proses/kegiatan bisnis.
	<i>Fork</i> (percabangan), digunakan untuk menunjukan kegiatan yang dilakukan secara paralel atau mengabungkan dua kegiatan paralel menjadi satu.
	<i>Join</i> (penggabungan) atau <i>rake</i> , digunakan untuk menunjukan adanya dekomposisi.
	<i>Decision points</i> , menggambarkan pilihan untuk pengambilan keputusan, <i>true, false</i> .
	<i>Swimlane</i> , pembagian <i>activity diagram</i> untuk menunjukan siapa melakukan apa.

(Sumber : Gellysa Urva: 2015 : 94)

3. *Class diagram*

Merupakan hubungan antar kelas dan penjelasan detail tiap-tiap kelas di dalam model desain dari suatu sistem, juga memperlihatkan aturan-aturan dan tanggung jawab entitas yang menentukan perilaku sistem.

Tabel II.4. Multiplicity Class Diagram

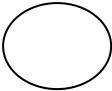
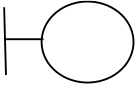
Multiplicity	Penjelasan
1	Satu dan hanya satu
0..*	Boleh tidak ada atau 1 atau lebih
1..*	1 atau lebih
0..1	Boleh tidak ada, maksimal 1
n..n	Batasan antara. Contoh 2..4 mempunyai arti minimal 2 maksimum 4.






(Sumber : Gellysa Urva: 2015 : 95)

4. *Sequence diagram*

Sequencediagram menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan pesan yang dikirimkan dan diterima antar objek. Simbol-simbol yang digunakan dalam *sequence diagram* yaitu:

Tabel II.5. Sequencediagram

Gambar	Keterangan
	<i>EntityClass</i> , merupakan bagian dari sistem yang berisi kumpulan kelas berupa entitas-entitas yang membentuk gambaran awal sistem dan menjadi landasan untuk menyusun basis data.
	<i>Boundary Class</i> , berisi kumpulan kelas yang menjadi <i>interface</i> atau interaksi antara satu atau lebih aktor dengan sistem, seperti tampilan <i>formentary</i> dan form cetak.

	<p><i>Control Class</i>, suatu objek yang berisi logika aplikasi yang tidak memiliki tanggung jawab kepala entitas, contohnya adalah kalkulasi dan aturan bisnis yang melibatkan berbagai objek.</p>
	<p><i>Message</i>, simbol pengirim pesan antar <i>class</i></p>
	<p><i>Recursive</i>, menggambarkan pengiriman pesan yang dikirim untuk dirinya sendiri.</p>
	<p><i>Activation</i> mewakili sebuah eksekusi operasi dari objek, panjang kotak ini berbanding lurus dengan durasi aktivitas sebuah operasi.</p>
	<p><i>Lifeline</i>, garis titik-titik yang terhubung dengan objek, sepanjang <i>lifeline</i> terdapat <i>activation</i></p>

(Sumber : Gellysa Urva: 2015 : 95)