



**SISTEM PENDUKUNG KEPUTUSAN PENENTUAN
JUMLAH KELUARGA KURANG MAMPU UNTUK
MENDAPAT BANTUAN BEASISWA PENDIDIKAN
MENGUNAKAN METODE SAW**

Tugas Akhir

disusun sebagai salah satu syarat
untuk memperoleh gelar Ahli Madya
Program Studi Statistika Terapan dan Komputasi

oleh

Diwa Saputra

4112313037

**JURUSAN MATEMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS NEGERI SEMARANG
2018**

PERNYATAAN

Saya menyatakan bahwa tugas akhir ini bebas plagiat, dan apabila di kemudian hari terbukti plagiat dalam tugas akhir ini, maka saya bersedia menerima sanksi berdasarkan ketentuan peraturan perundang-undangan.

Semarang, 21 September 2018



Diwa Saputra

4112313037

HALAMAN PENGESAHAN

Tugas Akhir yang berjudul

Sistem Pendukung Keputusan Penentuan Jumlah Keluarga Kurang Mampu
Untuk Mendapat Bantuan Beasiswa Pendidikan Menggunakan Metode
SAW.

disusun oleh:

Diwa Saputra

4112313037

telah dipertahankan di hadapan sidang Panitia Ujian Tugas Akhir FMIPA UNNES
pada tanggal 25 September 2018

Panitia



~~Prof.~~ Dr. Zaenuri, S.E, M.Si, Akt
NIP 196412231988031001

Sekretaris

Drs. Arief Agoestanto, M.Si.
NIP. 196807221993031005

Penguji 1

Dr. Dr. Wardono, M.Si.
NIP 196202071986011001

Penguji Pembimbing/
Pembimbing Utama

Dr. Scolastika Mariani, M.Si.
NIP. 19650210199102200

MOTTO DAN PERSEMBAHAN

MOTTO

Selalu berpikir positif agar apa yang ada di depan kamu selalu bersifat positif.

PERSEMBAHAN

Alhamdulillah, dengan mengucap rasa syukur kepada Allah Subhanahu Wa Ta'ala atas segala nikmat dan karunia-Nya tugas akhir ini kupersembahkan kepada:

1. Ibuk dan adik tersayang yang senantiasa mendukung, memberikan nasihat dan doa saat sedang sedih maupun senang untuk segala kelancaran dan kemudahan dalam kehidupan saya termasuk dalam menyelesaikan tugas akhir ini.
2. Bapak dan ibu dosen pembimbing yang senantiasa membimbing saya untuk menyelesaikan tugas akhir ini.
3. Keluarga kedua saya dari *dershane* atau rumah belajar yang selalu mendukung, memberikan arahan, serta menemani saya saat saya baik sedang jauh dari orang tua maupun sedang mendapat kesulitan.
4. Sahabat-sahabat saya yang juga senantiasa selalu mendukung saya dalam pengerjaan hingga terselesaikannya tugas akhir ini.
5. Keluarga jauh yang berada di Banjar.

6. Teman-teman masa kecil saya yang juga selalu mendukung untuk segera menyelesaikan tugas akhir ini.
7. Teman-teman alumni SD, SMP, dan SMA.

PRAKATA

Alhamdulillah puji syukur senantiasa penulis panjatkan kehadirat Allah Subhanahu Wa Ta'ala yang telah memberikan rahmat, hidayah serta karunia-Nya sehingga penulis dapat menyelesaikan Tugas Akhir ini dengan judul **“Sistem Pendukung Keputusan Penentuan Jumlah Keluarga Kurang Mampu Untuk Mendapat Bantuan Beasiswa Pendidikan Menggunakan Metode SAW”**.

Penulis dalam mengerjakan dan menyusun tugas akhir ini menyadari telah banyak mendapatkan bantuan, bimbingan, serta dorongan yang sangat bermanfaat dan berarti bagi penulis dari berbagai pihak. Oleh karena itu, penulis menyampaikan terimakasih kepada:

1. Prof. Dr. Fathur Rokhman, Rektor Universitas Negeri Semarang.
2. Prof. Dr. Zaenuri, S.E, M.Si, Akt., Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Negeri Semarang.
3. Drs. Arief Agoestanto, M.Si, Ketua Jurusan Matematika Universitas Negeri Semarang.
4. Dr. Dr. Wardono, M.Si., Ketua Program Studi Statistika Terapan dan Komputasi.
5. Dr. Scolastika Mariani, M.Si., selaku dosen pembimbing yang telah memberikan arahan, bimbingan, dan memberikan kemudahan dalam penyusunan tugas akhir.
6. Segenap civitas Jurusan Matematika Universitas Negeri Semarang, terutama Bapak dan Ibu dosen yang telah memberikan ilmunya dengan tulus.
7. Keluarga kecilku terkhusus ibu dan adik yang senantiasa mendoakan, memberikan nasihat dan motivasi dalam menyelesaikan tugas akhir ini.
8. Keluarga dari bapak Sumari selaku keluarga yang juga senantiasa mendukung dalam menyelesaikan tugas akhir ini.
9. Teman-teman Program Studi Statistika Terapan dan Komputasi angkatan 2013 yang telah membantu dan memberikan saran dalam penyelesaian tugas akhir ini.

10. Semua pihak yang telah membantu baik langsung dan tidak langsung sehingga tugas akhir ini dapat terselesaikan.

Penulis menyadari bahwa dalam penyusunan tugas akhir ini masih banyak terdapat kekurangan dan keterbatasan yang dimiliki penulis sehingga tugas akhir ini masih dikatakan jauh dari kata sempurna. Oleh karena itu, penulis mengharapkan kritik dan saran yang dapat membangun untuk dapat digunakan selanjutnya pada penelitian-penelitian yang lain. Semoga tugas akhir ini dapat bermanfaat dan berguna bagi pembaca maupun untuk penelitian yang lain.

Semarang, 21 September 2018

Penulis,

Diwa Saputra

4112313037

ABSTRAK

Diwa Saputra. 2018. *Sistem Pendukung Keputusan Penentuan Jumlah Keluarga Kurang Mampu Untuk Mendapat Bantuan Beasiswa Pendidikan Menggunakan Metode SAW*. Tugas Akhir. Program Studi Statistika Terapan dan Komputasi Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Negeri Semarang. Pembimbing Utama Dr. Scolastika Mariani, M.Si.

Kemiskinan merupakan masalah multikompleks yang setiap tahun menjadi pusat perhatian pemerintah di negara manapun dan tidak terkecuali Negara Indonesia dimana upaya pemerintah dalam menanggulangi kemiskinan telah dimulai sejak tahun 1970-an diantaranya melalui program RASKIN, BSM, KIP dan lain sebagainya. BSM dan KIP merupakan salah satu program pemerintah dalam mengentaskan kemiskinan melalui jalur pendidikan sebab pendidikan dapat meningkatkan sumber daya manusia. Sumber daya manusia yang tinggi dapat meningkatkan pula taraf hidup akan kesejahteraan masyarakat sehingga dapat meningkatkan pertumbuhan ekonomi. Akan tetapi, dalam kaitannya dengan penyaluran program bantuan untuk keluarga kurang mampu yang akan mendapat bantuan beasiswa pendidikan, dibutuhkan data-data serta variabel-variabel yang cukup memadai sehingga ketepatan dan keakuratan bantuan yang tersalur dapat merata dan tepat sasaran. Dengan demikian dibutuhkan sistem untuk menentukan keluarga kurang mampu secara cepat, efektif dan efisien, yakni sistem pendukung keputusan.

Sistem pendukung keputusan dalam hal ini merupakan cara yang digunakan oleh penulis untuk menentukan keluarga kurang mampu yang akan mendapat bantuan beasiswa pendidikan dengan menggunakan metode *simple additive weighting* dimana konsep dasar metode tersebut ialah penjumlahan terbobot sehingga diperoleh hasil akhir berupa nilai yang telah diolah menggunakan metode tersebut. Adapun hasil akhir dari nilai yang dimaksud akan digunakan untuk menentukan keluarga yang dikatakan kurang mampu untuk mendapat bantuan beasiswa pendidikan. Selain itu, sistem pendukung keputusan yang telah dirancang akan diimplementasikan ke dalam bahasa pemrograman *Visual Basic 2010* dengan menggunakan basis data *MariaDB* sebagai penyimpanan data dari variabel-variabel penentu keluarga kurang mampu.

Hasil akhir dari perancangan dan pembuatan program aplikasi menggunakan bahasa pemrograman *Visual Basic 2010* ialah berupa program aplikasi berbasis desktop yang berjalan di sistem operasi *Windows*. Pada program aplikasi sistem pendukung keputusan tersebut berfungsi untuk mengolah data dan variabel penentu keluarga kurang mampu yang dijalankan menggunakan perintah-perintah pada bahasa pemrograman *Visual Basic 2010* sehingga dihasilkan nilai perhitungan secara matematis menggunakan program aplikasi.

Jadi, dengan menggunakan program aplikasi sistem pendukung keputusan menggunakan metode *simple additive weighting* yang diimplementasikan ke dalam bahasa pemrograman *Visual Basic 2010* diharapkan dapat membantu pengguna

untuk menentukan keluarga kurang mampu yang akan mendapat bantuan beasiswa pendidikan secara efektif serta efisien.

Kata Kunci : Beasiswa Pendidikan, SPK, SAW, *Visual Basic*, Basis Data.

DAFTAR ISI

HALAMAN JUDUL	i
PERNYATAAN	ii
HALAMAN PENGESAHAN	iii
MOTO DAN PERSEMBAHAN	iv
PRAKATA	vi
ABSTRAK	vii
DAFTAR ISI	x
DAFTAR TABEL	xvi
DAFTAR GAMBAR.....	xviii
DAFTAR LAMPIRAN	xvix
BAB I.....	1
PENDAHULUAN.....	1
1.1 Latar Belakang Masalah.....	1
1.2 Rumusan Masalah	9
1.3 Batasan Masalah.....	9
1.4 Tujuan Penelitian.....	10
1.5 Manfaat Penelitian.....	10
BAB 2.....	12
LANDASAN TEORI	12
2.1 Kemiskinan.....	12

2.1.1 Definisi Kemiskinan.....	12
2.1.2 Klasifikasi Kemiskinan	16
2.1.3 Karakteristik Kemiskinan.....	19
2.1.4 Penyebab Kemiskinan	20
2.1.5 Indikator Kemiskinan.....	22
2.1.6 Strategi Penanggulangan Kemiskinan.....	24
2.2 Pendidikan.....	27
2.2.1 Definisi Pendidikan	27
2.2.2 Fungsi dan Tujuan Pendidikan.....	28
2.2.3 Jalur Pendidikan	28
2.2.3.1 Pendidikan Formal	28
2.2.3.2 Pendidikan Nonformal	29
2.2.3.3 Pendidikan Informal.....	29
2.3 Sistem Pendukung Keputusan.....	29
2.3.1 Definisi Sistem Pendukung Keputusan.....	29
2.3.2 Tujuan Sistem Pendukung Keputusan.....	31
2.3.3 Kerangka Kerja Sistem Pendukung Keputusan	32
2.3.4 Karakteristik dan Kemampuan Sistem Pendukung Keputusan.....	34
2.3.5 Tahapan Pengambilan Keputusan	37
2.3.6 Manfaat Sistem Pendukung Keputusan.....	40

2.4 <i>Fuzzy Multi-Attribute Decision Making</i>	41
2.5 <i>Simple Additive Weighting</i>	46
2.6 <i>Microsoft Visual Basic 2010</i>	48
2.6.1 <i>Bahasa Pemrograman Microsoft Visual Basic</i>	49
2.6.2 <i>Bahasa Pemrograman Microsoft Visual Basic 2010</i>	50
2.6.3 <i>Integrated Development Environment (IDE) Microsoft Visual Basic 2010</i>	51
2.6.4 <i>Kelebihan Bahasa Pemrograman Microsoft Visual Basic 2010</i>	59
2.7 <i>XAMPP</i>	60
2.7.1 <i>PHP (PHP Hypertext Preprocessor)</i>	61
2.7.2 <i>phpMyAdmin</i>	63
2.7.3 <i>MySQL</i>	63
2.7.4 <i>MariaDB</i>	64
2.8 <i>Database</i>	66
2.8.1 <i>Konsep Basis Data (Database)</i>	66
2.8.2 <i>Kriteria Basis Data (Database)</i>	68
2.8.3 <i>Kegunaan Basis Data (Database)</i>	69
2.8.4 <i>Tujuan Basis Data (Database)</i>	69
2.8.5 <i>Keuntungan Basis Data (Database)</i>	69
2.8.6 <i>Database Management System (DBMS)</i>	70

2.8.6.1 Definisi <i>Database Management System</i> (DBMS).....	70
2.8.6.2 Fungsi-Fungsi <i>Database Management System</i>	71
2.8.6.3 Tujuan <i>Database Management System</i>	72
2.8.6.4 Bahasa-Bahasa dalam <i>Database Management System</i>	72
2.8.6.5 Keunggulan <i>Database Management System</i>	74
2.8.7 Alat / Perangkat Pemodelan Sistem Informasi.....	75
2.8.7.1 Bagan Alir (<i>Flow Chart</i>).....	75
2.8.7.2 <i>Entity-Relationship Model (E-R Model)</i>	77
BAB 3	105
METODE PENELITIAN	105
3.1 Ruang Lingkup Penelitian.....	105
3.2 Data, Variabel dan Indikator	105
3.3 Jenis dan Sumber Data	106
3.4 Metode Pengumpulan Data	107
3.5 Metode Pengembangan Sistem	108
2.5.1 Analisis (<i>Analysis</i>).....	109
2.5.2 Perancangan (<i>Design</i>)	110
2.5.2.1 Diagram <i>Entity-Relationship (E-R)</i>	110
2.5.2.2 Rancangan Program	111
2.5.3 Pengkodean (<i>Code</i>)	116

2.5.4 Pengujian (<i>Test</i>).....	116
3.6 Metode Analisis Data	117
BAB 4	118
HASIL DAN PEMBAHASAN PENELITIAN	118
4.1 Hasil Pembuatan Program Aplikasi	118
4.1.1 Form <i>Splash Screen</i>	118
4.1.2 Form Login.....	119
4.1.3 Form Menu Awal	120
4.1.4 Form Menu Profil.....	120
4.1.5 Form Ubah Password	121
4.1.6 Form Menu Pegawai	121
4.1.7 Form Input Data Pegawai.....	122
4.1.8 Form Data Penduduk.....	123
4.1.9 Form Data Wilayah	124
4.1.10 Form Data Rumah	125
4.1.11 Form Data Entri.....	125
4.1.12 Form Entri Data.....	126
4.1.13 Form Keputusan	127
4.2 Pembahasan	130
BAB 5	133

KESIMPULAN	133
5.1 Kesimpulan.....	133
5.2 Saran.....	133
DAFTAR PUSTAKA.....	135
LAMPIRAN	147

DAFTAR TABEL

Tabel 2.1 Variabel dan Pemutakhiran Basis Data Terpadu Tahun 2015.....	23
--	----

DAFTAR GAMBAR

Gambar 2.1 DSS Berfokus pada Masalah Semiterstruktur	33
Gambar 2.2 Karakteristik dan Kemampuan Kunci dari DSS	35
Gambar 2.3 Pengambilan Keputusan/Proses Pemodelan	38
Gambar 2.4 Tampilan Jendela <i>Splash Visual Basic 2010 Express</i>	52
Gambar 2.5 Tampilan <i>Start Page Visual Basic 2010 Express</i>	53
Gambar 2.6 Tampilan Kotak Dialog <i>New Project</i>	54
Gambar 2.7 Tampilan <i>Integrated Development Environment Visual Basic 2010 Express</i>	55
Gambar 2.8 Tampilan <i>Title Bar Visual Basic 2010 Express</i>	55
Gambar 2.9 Tampilan <i>Menu Bar Visual Basic 2010 Express</i>	56
Gambar 2.10 Tampilan <i>Toolbar Visual Basic 2010 Express</i>	56
Gambar 2.11 Tampilan <i>Toolbox Visual Basic 2010 Express</i>	56
Gambar 2.12 Tampilan <i>Form Designer Visual Basic 2010 Express</i>	57
Gambar 2.13 Tampilan <i>Solution Explorer Visual Basic 2010 Express</i>	57
Gambar 2.14 Tampilan <i>Properties Visual Basic 2010 Express</i>	58
Gambar 2.15 Tampilan <i>Code Editor Visual Basic 2010 Express</i>	58
Gambar 2.16 Tampilan <i>Error List Visual Basic 2010 Express</i>	59
Gambar 2.17 Diagram Ketergantungan Fungsional	89
Gambar 2.18 Diagram Hubungan antar Bentuk Normal	94
Gambar 2.19 Diagram Proses Normalisasi	95
Gambar 3.1 Diagram <i>E-R</i>	110
Gambar 3.2 Tampilan Awal	111

Gambar 3.3 Form Profil	111
Gambar 3.4 Form Pegawai	112
Gambar 3.5 Form Penduduk	112
Gambar 3.6 Form Wilayah	113
Gambar 3.7 Form Rumah	114
Gambar 3.8 Form Data Entri	115
Gambar 3.9 Form Keputusan	116
Gambar 4.1 Form <i>Splash Screen</i>	119
Gambar 4.2 Form Login	119
Gambar 4.3 Form Menu Awal	120
Gambar 4.4 Form Menu Profil	121
Gambar 4.5 Form Ubah Password	121
Gambar 4.6 Form Menu Pegawai	122
Gambar 4.7 Form Input Data Pegawai	123
Gambar 4.8 Form Data Penduduk	124
Gambar 4.9 Form Data Wilayah	124
Gambar 4.10 Form Data Rumah	125
Gambar 4.11 Form Data Entri	126
Gambar 4.12 Form Entri Data	127
Gambar 4.13 Form Keputusan (Muat Data)	128
Gambar 4.14 Form Keputusan (Penilaian Pertama)	128
Gambar 4.15 Form Keputusan (Penilaian Kedua)	128
Gambar 4.16 Form Keputusan (Penilaian)	129

Gambar 4.17 Form Keputusan (Normalisasi)	129
Gambar 4.18 Form Keputusan (Hasil Akhir)	130

DAFTAR LAMPIRAN

1. Kriteria dan Bobot Penilaian Setiap Variabel.....	147
--	-----

BAB 1

PENDAHULUAN

1.1. Latar Belakang Masalah

Kemiskinan merupakan salah satu indikator yang sangat penting untuk mencapai keberhasilan pembangunan pada suatu negara. Menurut Kunajo (dalam Munir, 2002: 10), suatu negara dikatakan miskin biasanya ditandai dengan tingkat pendapatan perkapita rendah, mempunyai tingkat pertumbuhan penduduk yang tinggi (lebih dari 2 persen per tahun), sebagian besar tenaga kerja bergerak di sektor pertanian, dan terbelenggu dalam kemiskinan. Oleh karena itu, setiap negara akan berusaha keras untuk mencapai pembangunan dengan cara mengoptimalkan pertumbuhan ekonomi dan menurunkan tingkat kemiskinan.

Kemiskinan merupakan salah satu persoalan mendasar yang menjadi pusat perhatian pemerintah di negara manapun, tidak terkecuali Negara Indonesia. Indonesia merupakan salah satu negara berkembang di Asia Tenggara dimana kemiskinan bukan merupakan hal yang baru sebab setiap tahun masalah kemiskinan selalu menjadi prioritas utama dan sangat penting bagi pemerintah Indonesia dalam upaya pembangunan nasional. Kemiskinan merupakan suatu persoalan yang kompleks, karena tidak hanya berkaitan dengan masalah rendahnya tingkat pendapatan serta konsumsi masyarakat, tetapi berkaitan pula dengan rendahnya tingkat pendidikan, kesehatan, serta ketidakberdayaannya untuk berpartisipasi dalam pembangunan serta berbagai masalah yang berkenaan dengan pembangunan manusia. Dimensi-dimensi kemiskinan tersebut termanifestasikan dalam bentuk kekurangan gizi, air, perumahan yang sehat, perawatan kesehatan yang kurang baik, dan tingkat pendidikan yang rendah (Wijayanti dan Wahono, 2005: 215).

Menurut Badan Pusat Statistik (BPS) (2015: 119) Indonesia, kemiskinan dikonseptualisasikan sebagai ketidakmampuan dalam memenuhi kebutuhan dasar dan diukur dari sisi pengeluaran perkapita atau dengan kata lain kemiskinan

dipandang dari sisi ketidakmampuan ekonomi. Berdasarkan data yang diperoleh dari Badan Pusat Statistik (2015: 175) pada tahun 1996, jumlah penduduk miskin di Indonesia mencapai 34,01 juta penduduk. Sedangkan pada tahun 1998, jumlah penduduk miskin berjumlah 49,50 juta penduduk yang artinya, pada tahun tersebut tingkat kemiskinan di Indonesia semakin tinggi sebab pada tahun tersebut telah terjadi krisis moneter di Indonesia (Tim Nasional Percepatan Penanggulangan Kemiskinan, 2010: 1). Namun, setelah tahun 1998 berakhir, jumlah penduduk miskin di Indonesia sempat mengalami penurunan yang pada tahun 2000 berjumlah 38,70 juta penduduk menjadi 37,90 juta penduduk pada tahun 2001 dan meningkat kembali pada tahun 2002 sebesar 38,40 juta penduduk. Namun, pada tahun berikutnya, yakni dari tahun 2003 sampai tahun 2014, jumlah penduduk miskin di Indonesia terus mengalami penurunan secara signifikan hingga data terakhir pada September 2015 jumlah penduduk miskin di Indonesia menjadi 27,73 juta penduduk atau jika diubah dalam persentase maka sebanyak 10,96 persen dari total penduduk di Indonesia, ini artinya upaya pemerintah dalam meminimalisir angka kemiskinan mengalami keberhasilan.

Remi dan Tjiptoherihabto (2002: 1) mengungkapkan bahwa upaya pemerintah Indonesia dalam menanggulangi masalah kemiskinan telah dimulai sejak awal tahun 1970-an diantaranya melalui program penyaluran Beras untuk Rumah Tangga Miskin (RASKIN), Bimbingan Masyarakat (Bimas), dan Bantuan Desa (Bandes). Ada pula program-program pemerintah yang lain untuk menurunkan tingkat kemiskinan, yaitu Bantuan Langsung Tunai (BLT) sekarang bernama Bantuan Langsung Sementara Masyarakat (BLSM), Kredit Usaha Rakyat (KUR) yang digunakan untuk pengembangan Usaha Mikro, Kecil, dan Menengah (UMKM), Program Keluarga Harapan (PKH), Bantuan Siswa Miskin (BSM), Jaminan Kesehatan Masyarakat (Jamkesmas), dan Program Nasional Pemberdayaan Masyarakat (PNPM) Mandiri yang dicetuskan disaat masa kepemimpinan Susilo Bambang Yudhoyono (<https://news.detik.com/berita/2720289/jurus-jitu-sby-mengentaskan-kemiskinan> diakses pada 29 September 2016). Pada tahun 2014, terdapat program-program baru yang diluncurkan pada masa kepemimpinan baru oleh Presiden Joko Widodo, yaitu Jaminan Kesehatan Nasional serta Program

Indonesia Pintar (<http://bisnis.liputan6.com/read/2256860/strategi-pemerintah-jokowi-kurangi-tingkat-kemiskinan> diakses pada 20 September 2016).

Bantuan Siswa Miskin dan Kartu Indonesia Pintar merupakan salah satu dari sekian program pemerintah untuk mengentaskan kemiskinan melalui jalur pendidikan. Pendidikan merupakan suatu hal yang penting dalam rangka untuk mengentaskan masalah kemiskinan karena pendidikan menjadi sarana untuk dapat meningkatkan kualitas sumber daya manusia sehingga dapat menghasilkan tenaga kerja yang lebih produktif. Tenaga kerja yang lebih produktif akan meningkatkan penghasilan atau pendapatan yang lebih banyak atau tinggi sehingga secara agregat akan meningkatkan pertumbuhan ekonomi dalam suatu negara. Dengan demikian maka dibutuhkan upaya pemerintah untuk dapat meningkatkan mutu dan kualitas pendidikan yang layak dan memadai bagi warga yang tinggal dalam negara bersangkutan sehingga kesejahteraan masyarakat akan terjamin. Adapun upaya pemerintah dalam meningkatkan mutu dan kualitas pendidikan ialah melalui kebijakan pembangunan pendidikan. Kebijakan pembangunan pendidikan yang tertuang dalam Rencana Pembangunan Jangka Menengah Nasional (RPJMN) 2015 – 2019 menyebutkan bahwa terdapat tujuh arah arah kebijakan pembangunan pendidikan tahun 2017, yakni memenuhi pembiayaan kegiatan prioritas nasional dalam Rencana Kerja Pemerintah (RKP) tahun 2017 untuk pencapaian Nawacita, memberikan perhatian lebih besar pada daerah tertinggal, terluar, dan terdepan (3T), memastikan masyarakat miskin dan kelompok marjinal lebih mudah mengakses layanan pendidikan dengan memperhatikan keadilan dan kesetaraan gender, memanfaatkan anggaran pembangunan pendidikan semaksimal mungkin dirasakan oleh masyarakat, memastikan keterlibatan publik secara maksimal, dan memperkuat tata kelola pembangunan pendidikan dan kebudayaan, termasuk pelaksanaan anggaran secara transparan dan akuntabel (<https://www.kemdikbud.go.id/main/blog/2016/09/target-kemendikbud-dalam-pengembangan-pendidikan-dan-kebudayaan-tahun-2017> diakses pada 26 September 2018).

Akan tetapi masalah yang menjadi salah satu target dan sasaran pemerintah ialah meningkatkan akses pendidikan. Untuk akses peningkatan pendidikan ini pemerintah memberikan kebijakan dengan memberikan Kartu Indonesia Pintar kepada keluarga kurang mampu dengan anak-anak yang putus sekolah maupun yang tidak dapat melanjutkan pendidikan ke jenjang berikutnya. Dengan kebijakan tersebut maka bantuan beasiswa pendidikan diharapkan mampu meringankan beban masyarakat terkait dengan pendidikan sehingga dengan pendidikan yang terjamin diharapkan pula mampu mengentaskan kemiskinan yang juga akan berdampak pada pertumbuhan ekonomi dalam suatu negara (<https://www.kemdikbud.go.id/main/blog/2016/09/target-kemendikbud-dalam-pengembangan-pendidikan-dan-kebudayaan-tahun-2017> diakses pada 26 September 2018).

Upaya pemerintah dalam mengentaskan kemiskinan telah mengalami keberhasilan yang ditunjukkan dengan menurunnya jumlah penduduk miskin di Indonesia sebanyak 11,57 juta penduduk pada periode 2006 – 2014 (Badan Pusat Statistik, 2015: 175). Keberhasilan dalam upaya pengentasan kemiskinan tersebut tidak lepas kaitannya dengan tercapai tujuan pemerintah yang sukses dilaksanakan melalui program-program pemerintah salah satu program tersebut ialah program beasiswa pendidikan bagi keluarga kurang mampu, seperti Bantuan Siswa Miskin dan Kartu Indonesia Pintar. Program-program tersebut merupakan salah satu kebijakan pemerintah bagi keluarga kurang mampu di Indonesia dengan sumber data yang digunakan ialah data penduduk Indonesia yang sudah melewati tahap penentuan seleksi penduduk miskin. Pemerintah melalui suatu badan penanggulangan kemiskinan yang bernama Tim Nasional Percepatan Penanggulangan Kemiskinan (TN2PK) mengeluarkan sebuah kebijakan untuk menentukan keluarga kurang mampu. Kebijakan tersebut diambil dengan tujuan untuk mempermudah dalam menentukan jumlah keluarga miskin di Indonesia yang berhak menerima program bantuan dari pemerintah, seperti beasiswa pendidikan serta untuk mengurangi adanya kesalahan atau *human error* dalam penentuan keluarga miskin, sebab menurut data dari Badan Pusat Statistik, ada beberapa

rumah tangga yang dikatakan mampu tetapi masuk dalam kategori keluarga tidak mampu.

Menurut Badan Pusat Statistik (2015: 51) Indonesia, ada beberapa variabel yang dapat digunakan untuk menentukan status kesejahteraan rumah tangga di Indonesia. Beberapa variabel penentuan status kesejahteraan rumah tangga yang dimaksud dari Badan Pusat Statistik Indonesia, yaitu: nama, umur, jenis kelamin, status perkawinan, hubungan dengan kepala rumah tangga dan keluarga, kepemilikan kartu identitas, ketercantuman pada kartu keluarga, kepemilikan akta nikah, kecacatan, penyakit kronis, status kehamilan, partisipasi sekolah, pendidikan tertinggi, lama sekolah, mempunyai pekerjaan, lapangan usaha, status pekerjaan, kepemilikan bangunan dan lahan tempat tinggal, luas lantai terluas, jenis lantai terluas, dinding terluas, atap terluas, jumlah kamar tidur, sumber dan cara memperoleh air minum, sumber penerangan utama, bahan bakar utama memasak, kepemilikan jamban, jenis kloset, tempat pembuangan akhir, kepemilikan aset, serta keikutsertaan program keluarga berencana, klaster 1, klaster 2, dan klaster 3.

Berkaitan dengan penentuan keluarga kurang mampu di Indonesia, pemerintah selalu berusaha agar bantuan yang diberikan tersebut dapat tersalur secara merata dan tepat sasaran. Namun, dalam realita di Indonesia banyak terjadi komplain dari pihak masyarakat yang disebabkan karena bantuan yang diberikan tersebut belum sesuai dengan harapan yang ada (<http://www.republika.co.id/berita/koran/halaman-1/16/04/19/o5va079-...1> diakses pada 02 Oktober 2016) mengakibatkan kurang optimalnya program penanggulangan kemiskinan. Selain itu, pertumbuhan penduduk Indonesia yang selalu mengalami kenaikan setiap tahunnya (Badan Pusat Statistik, 2011: 7) menyebabkan sulitnya penentuan keluarga kurang mampu bagi badan penanggulangan kemiskinan Indonesia. Akan tetapi, hal tersebut menjadi pelajaran bagi pemerintah sekaligus pihak-pihak yang terkait akan hal penanggulangan kemiskinan di Indonesia untuk lebih meningkatkan ketelitian serta ketepatan dalam proses penentuan masyarakat kurang mampu. Belajar dari kesalahan serta kesulitan dalam menentukan keluarga miskin di Indonesia, maka dibutuhkan sebuah metode

yang lebih efektif dan efisien. Untuk itu, diperlukan model untuk menyelesaikan permasalahan dalam pengambilan keputusan, yaitu dengan menggunakan suatu sistem yang dikenal dengan istilah Sistem Pendukung Keputusan (SPK).

Sistem Pendukung Keputusan merupakan suatu sistem informasi interaktif yang menyediakan informasi, pemodelan, dan pemanipulasian data dengan tujuan untuk membantu pengambilan keputusan dalam situasi yang semi terstruktur dan situasi yang tidak terstruktur, dimana tak seorang pun tahu pasti bagaimana keputusan seharusnya dibuat (Alter dalam Kusri, 2007: 15-16). Secara umum, sistem pendukung keputusan dirancang untuk mendukung solusi atas suatu masalah atau untuk mengevaluasi suatu peluang. Biasanya sistem pendukung keputusan lebih ditujukan untuk mendukung manajemen dalam melakukan pekerjaan yang bersifat analitis dalam situasi yang kurang terstruktur dan dengan kriteria yang belum pasti. Dengan kata lain sistem pendukung keputusan bertujuan untuk memberikan keputusan yang lebih baik dan efisien, artinya sistem pendukung keputusan dapat memberikan keuntungan bagi para penggunanya, seperti biaya yang relatif rendah, lebih efisien dalam mendukung suatu keputusan, meningkatkan mutu kualitas dan produktivitas sehingga dapat meningkatkan pendapatan, mengurangi risiko kesalahan dalam pengambilan suatu keputusan, dan lain sebagainya. Adapun dengan adanya kemampuan dan kelebihan dari sistem pendukung keputusan tersebut maka sistem pendukung keputusan dapat diimplementasikan dalam berbagai bidang kehidupan, seperti pada industri pembuatan peralatan pencucian, pada perusahaan untuk perencanaan ataupun manajemen karyawan, pada bidang transportasi dalam manajemen jalur perjalanan ataupun penjadwalan, pada bidang pendidikan maupun kesehatan (Pick, 2008: 721-722). Oleh sebab itu, peneliti menggunakan sistem pendukung keputusan sebagai solusi dalam menyelesaikan masalah pengambilan keputusan dengan tujuan bahwa sistem tersebut nantinya akan digunakan untuk menentukan jumlah keluarga kurang mampu yang berhak mendapat bantuan beasiswa pendidikan dari pemerintah Indonesia.

Dalam pengambilan suatu keputusan, terdapat beberapa model yang dapat digunakan dalam sistem pendukung keputusan, salah satunya ialah *Fuzzy Multiple Attribute Decision Making (FMADM)*. *Multi-Attribute Decision Making (MADM)* adalah suatu model yang digunakan untuk mencari alternatif optimal dari sejumlah alternatif dengan kriteria tertentu dimana inti dari MADM ialah menentukan nilai bobot untuk setiap atribut (Zimmermann dalam Kusumadewi, 2006: 72-73). Kemudian, masalah MADM dilanjutkan dengan proses perankingan yang akan menyeleksi alternatif yang sudah diberikan (Yeh dalam Kusumadewi, 2006: 73). Selain itu, terdapat pula metode yang dapat digunakan sebagai solusi untuk menyelesaikan masalah *Multiple Attribute Decision Making (MADM)*, salah satu diantaranya ialah menggunakan metode *Simple Additive Weighting (SAW)*. *Simple Additive Weighting (SAW)* sering pula dikenal dengan istilah metode penjumlahan terbobot. Konsep dasar metode ini ialah mencari penjumlahan terbobot dari rating kinerja pada setiap alternatif pada semua atribut / indikator (Fishburn dan MacCrimmon dalam Kusumadewi, 2006: 74). Metode skoring dan penjumlahan terbobot merupakan istilah lain yang digunakan secara alternatif untuk merepresentasikan pendekatan SAW dalam ilmu keputusan. Metode ini banyak digunakan dalam berbagai pemecahan masalah terkait pengambilan keputusan dengan alasan bahwa metode ini sangat sederhana dan mudah untuk digunakan atau diterapkan dalam pengambilan keputusan sehingga menjadikan metode ini sangat populer dalam berbagai penelitian-penelitian pendukung keputusan (Afshari dkk., 2010: 512). Adapun penelitian tersebut ialah penentuan beasiswa bagi siswa ataupun mahasiswa berprestasi ataupun tidak mampu, mendukung keputusan untuk rekomendasi penerimaan pekerja, penentuan karyawan teladan, mendukung keputusan dalam pemberian pinjaman, pemilihan bibit sapi unggul, sistem pemetaan lahan pertanian berkualitas, dan lain sebagainya.

Kemajuan teknologi informasi yang semakin berkembang dengan pesatnya ternyata telah membawa dampak yang multikompleks dalam berbagai segi kehidupan manusia, salah satu diantaranya ialah pembuatan program aplikasi menggunakan bahasa pemrograman *Visual Basic*. *Visual Basic* merupakan salah satu bahasa pemrograman yang dirancang untuk memudahkan pengguna (*user*)

dalam membuat maupun mengembangkan suatu program aplikasi. Di sisi lain, terdapat kelebihan yang dimiliki dari bahasa pemrograman *Visual Basic* daripada dengan bahasa pemrograman lain, seperti *Borland Delphi*, *Turbo Pascal*, dan *IDE Net*. Beberapa kelebihan tersebut diantaranya ialah mudah digunakan (*user friendly*), mudah dipahami, serta tampilan visual yang menarik. Selain itu, jika ditinjau dari segi matematis, bahasa pemrograman *Visual Basic* memiliki fitur khusus untuk perhitungan matematika yang akurat, yang belum peneliti temukan dalam bahasa pemrograman lain. Fitur yang dimaksud dalam bahasa pemrograman *Visual Basic* ialah fitur *class* “.math”, dengan fitur ini, keakuratan hasil perhitungan bisa diandalkan serta angka kesalahan dalam perhitungan dapat diminimalisir sekecil mungkin (Winarno, 2010: 116).

Berdasarkan beberapa pemikiran yang telah diuraikan di atas, maka peneliti akan mengangkat sebuah penelitian dengan judul “Sistem Pendukung Keputusan Penentuan Jumlah Keluarga Kurang Mampu Untuk Mendapat Bantuan Beasiswa Pendidikan Menggunakan Metode SAW”. Adapun selanjutnya, peneliti akan mencoba mengimplementasikan sistem pendukung keputusan menggunakan metode *saw* tersebut menggunakan bahasa pemrograman *Visual Basic*. Diharapkan dengan adanya pembuatan program aplikasi ini akan mampu mengurangi kesalahan (*error*) dalam menentukan keluarga kurang mampu. Selain itu, bagi pihak yang berwenang dalam hal penanggulangan kemiskinan di Indonesia, dengan adanya program aplikasi sistem pendukung keputusan ini, nantinya dapat mempermudah dalam penentuan jumlah keluarga kurang mampu yang berhak mendapat bantuan beasiswa pendidikan dari pemerintah baik dari segi efisiensi waktu, ketelitian atau keakuratan, maupun pemerataan.

1.2. Rumusan Masalah

Berdasarkan latar belakang di atas, maka permasalahan penelitian ini dapat dirumuskan sebagai berikut:

1. Bagaimana rancang bangun serta cara kerja dari sistem pendukung keputusan penentuan jumlah keluarga kurang mampu untuk mendapat bantuan beasiswa pendidikan menggunakan metode *Simple Additive Wighting* (SAW)?
2. Bagaimana implementasi sistem pendukung keputusan menggunakan bahasa pemrograman *Visual Basic 2010*?

1.3. Batasan Masalah

Berdasarkan latar belakang serta rumusan masalah yang telah diuraikan, peneliti melakukan pembatasan masalah dengan tujuan agar pembahasan terfokus dan mendalam serta tidak meluas dan menyimpang dari penelitian yang dimaksudkan. Adapun batasan masalah dari penelitian ini adalah pembuatan sistem pendukung keputusan penentuan jumlah keluarga kurang mampu yang berhak mendapat bantuan beasiswa pendidikan dari pemerintah Indonesia menggunakan model pengambilan keputusan *Fuzzy Multiple Atribute Decision Making* atau FMADM. Untuk menyelesaikan masalah FMADM, peneliti menggunakan metode *Simple Additive Weighting* (SAW) dengan konsep dasar adalah mencari penjumlahan terbobot dari ranting kinerja pada setiap alternatif terbaik dengan kriteria-kriteria tertentu (Fishbun dan MacCrimmon dalam Kusumadewi, 2006: 74). Selain itu, peneliti akan menggunakan variabel-variabel penentuan status kesejahteraan rumah tangga yang bersumber dari Badan Pusat Statistik (BPS). Pada setiap variabel-variabel tersebut akan diberikan bobot penilaian yang ditentukan berdasarkan pada instansi atau perusahaan terkait dengan hasil akhir berupa penjumlahan dari semua bobot penilaian tersebut diperoleh nilai terbesar yang dipilih sebagai alternatif terbaik sebagai solusi.

Produk yang akan peneliti hasilkan nantinya berupa program aplikasi berbasis desktop dimana merupakan implementasi dari sistem pendukung

pendukung keputusan ke dalam bahasa pemrograman *Visual Basic 2010* serta basis data yang digunakan adalah *MariaDB* yang berjalan menggunakan perangkat lunak berbasis *server offline* bernama *XAMPP*. Diharapkan, keluaran dari program aplikasi ini mampu meringankan beban pegawai atau pihak yang terkait untuk menentukan keluarga kurang mampu secara akurat, tepat, merata, serta efisien yang diakibatkan karena begitu tingginya angka penduduk Indonesia dan tidak ada lagi pemalsuan data penduduk antara yang mampu dan tidak mampu.

1.4. Tujuan Penelitian

Adapun tujuan penelitian ini sesuai dengan rumusan masalah yang telah dikemukakan di atas, yaitu:

1. Mendeskripsikan rancang bangun serta cara kerja dari sistem pendukung keputusan penentuan jumlah keluarga kurang mampu untuk mendapat bantuan beasiswa pendidikan menggunakan metode *Simple Additive Weighting* (SAW).
2. Mengimplementasikan sistem pendukung keputusan menggunakan bahasa pemrograman *Visual Basic 2010*.

1.5. Manfaat Penelitian

Adapun manfaat yang diharapkan dari penelitian serta penyusunan tugas akhir ini, yaitu sebagai berikut:

1. Bagi Mahasiswa

Manfaat dari hasil penelitian dan penyusunan tugas akhir ini diharapkan mampu digunakan sebagai sarana untuk menerapkan ilmu yang telah diperoleh selama proses perkuliahan sehingga dapat menjadi bekal untuk terjun dalam dunia kerja. Selain itu, diharapkan pula dapat membantu mahasiswa dalam memahami teori serta cara kerja mengenai sistem pendukung keputusan dan implementasi

penggunaan sistem pendukung keputusan menggunakan bahasa pemrograman *Visual Basic 2010*.

2. Bagi Jurusan Matematika

Hasil penelitian dan penyusunan tugas akhir ini diharapkan dapat memberikan kontribusi pemikiran dan teori mengenai pembuatan sistem pendukung keputusan dengan metode *Simple Additive Weighting* menggunakan bahasa pemrograman *Visual Basic 2010*. Gagasan ini pula diharapkan mampu menjadi bahan rujukan ataupun referensi yang baik serta bermanfaat baik bagi mahasiswa maupun pembaca. Selain itu, tidak menutup kemungkinan rujukan yang digunakan tersebut akan menjadi kajian untuk studi kasus ataupun pengembangan penelitian lebih lanjut.

3. Bagi Badan Penanggulangan Kemiskinan

Adapun manfaat yang diharapkan bagi pihak atau badan yang menanggulangi kemiskinan di Indonesia, yaitu:

- a. Memberikan kemudahan dalam penyeleksian untuk menentukan jumlah keluarga kurang mampu yang lebih berhak mendapat bantuan beasiswa pendidikan dari pemerintah.
- b. Meningkatkan efisiensi waktu agar penentuan dan penyeleksian tidak terlalu lama.
- c. Memperkecil kemungkinan kesalahan akan penentuan jumlah keluarga yang berhak mendapat bantuan beasiswa pendidikan dari pemerintah.
- d. Meningkatkan kualitas kinerja dalam badan yang menanggulangi masalah kemiskinan di Indonesia dengan konsep sistem yang baik dan efisien.

BAB 2

LANDASAN TEORI

Landasan teori ini dibuat dengan tujuan untuk membahas teori terkait kemiskinan, penyebab kemiskinan, serta cara penanggulangan kemiskinan. Adapun pada bagian cara penanggulangan kemiskinan akan dibahas pula cara menentukan jumlah keluarga kurang mampu menggunakan sistem pendukung keputusan dengan metode *Simple Additive Weighting* dan implementasinya menggunakan bahasa pemrograman *Visual Basic 2010*. Berikut pemaparan secara rinci mengenai teori yang menjadi landasan penelitian serta pembuatan tugas akhir ini.

2.1. Kemiskinan

Sub bab pada bahasan ini, akan menjelaskan teori mengenai kemiskinan yang meliputi definisi kemiskinan, bentuk-bentuk / klasifikasi kemiskinan, karakteristik kemiskinan, penyebab terjadinya kemiskinan, kriteria / indikator kemiskinan, serta strategi pemerintah dalam menanggulangi masalah kemiskinan.

2.1.1. Definisi Kemiskinan

Secara konseptual, istilah kemiskinan dipandang sebagai ketidakmampuan dari sisi ekonomi untuk memenuhi kebutuhan dasar baik berupa makanan (pangan) maupun bukan makanan (sandang dan papan) yang diukur dari sisi pengeluaran (Badan Pusat Statistik, 2015: 2). Sedangkan, menurut Badan Perencanaan Pembangunan Nasional (2004: 9), kemiskinan tidak lagi dipahami hanya sebatas ketidakmampuan ekonomi, tetapi juga dipandang sebagai kondisi dimana seseorang atau sekelompok orang baik laki-laki maupun perempuan yang mana hak-hak dasarnya tidak dapat terpenuhi secara layak untuk menempuh dan mengembangkan kehidupan yang bermartabat. Hal tersebut dapat ditinjau dari aspek pendekatan berbasis hak yang memiliki artian bahwa masyarakat miskin mempunyai hak-hak dasar yang sama dengan kelompok atau anggota masyarakat lainnya. Ketidaksamaan kesempatan untuk mengakumulasi basis kekuatan sosial yang meliputi: aset, sumber-sumber keuangan, organisasi dan jaringan sosial,

pengetahuan dan informasi dengan tujuan untuk memperoleh pekerjaan menjadikan seseorang menjadi miskin (Friedman dalam Ridlo, 2001: 8).

Menurut Chambers (1988: 145-148), seseorang dapat dikatakan miskin diakibatkan karena terperangkap dalam situasi ketidakberuntungan (*deprivation trap*) yang memiliki lima dimensi, yaitu:

1) Kemiskinan (*Proper*)

Permasalahan kemiskinan merupakan suatu konsep yang dipandang sebagai ketidakmampuan pendapatan dalam mencukupi kebutuhan-kebutuhan pokok baik sandang, pangan, maupun papan. Selain itu, konsep kemiskinan juga berlaku tidak hanya kepada kelompok yang tidak memiliki pendapatan, tetapi pula berlaku bagi kelompok yang telah memiliki pekerjaan. Namun, pendapatan yang diperoleh dari hasil pekerjaan tersebut tidak cukup untuk memenuhi kebutuhan dasar.

2) Ketidakberdayaan (*Powerless*)

Rendahnya kemampuan dalam aspek pendapatan bagi seseorang atau sekelompok orang dengan tujuan untuk memenuhi kebutuhan dasar akan berdampak pada kekuatan sosial (*social power*) dalam memperoleh keadilan ataupun persamaan hak-hak untuk mendapatkan kehidupan yang layak atau bermartabat bagi kemanusiaan.

3) Kerentanan menghadapi Situasi Darurat (*State of Emergency*)

Pendapatan seseorang atau sekelompok orang yang rendah akan berdampak pada kemampuan untuk menghadapi situasi yang tidak terduga atau situasi darurat yang sewaktu-waktu membutuhkan alokasi pendapatan untuk menyelesaikannya. Situasi tidak terduga yang dimaksud dapat berupa bencana alam, kondisi kesehatan yang membutuhkan biaya pengobatan yang relatif mahal ataupun situasi-situasi darurat lain yang mana cara penyelesaiannya ialah dengan membutuhkan kemampuan pendapatan yang mencukupi.

4) Ketergantungan (*Dependency*)

Keterbatasan akan hal pendapatan ataupun kekuatan sosial dalam diri seseorang akan mengakibatkan tingkat ketergantungan terhadap pihak lain menjadi lebih tinggi. Mereka yang tidak memiliki kemampuan atau kekuatan untuk

menciptakan solusi atau penyelesaian masalah terutama berkaitan cara untuk mencari pendapatan baru akan sangat terbantu dengan adanya bantuan dari pihak lain. Bantuan dari pihak lain tersebut sangat diperlukan guna untuk mengatasi persoalan-persoalan yang berkaitan dengan kebutuhan akan sumber pendapatan.

5) Keterasingan (*Isolation*)

Dimensi keterasingan yang dimaksud oleh Chambers ialah faktor tempat atau lokasi yang menyebabkan seseorang atau sekelompok orang menjadi miskin. Secara umum, masyarakat yang dikatakan miskin berada jauh dari daerah dengan pusat-pusat pertumbuhan ekonomi. Hal ini disebabkan karena sebagian besar fasilitas kesejahteraan lebih banyak terkonsentrasi di daerah dengan pusat-pusat pertumbuhan ekonomi, seperti daerah perkotaan atau kota-kota besar. Akibatnya, masyarakat yang tinggal jauh dari pusat pertumbuhan ekonomi atau dapat dikatakan berada di daerah terpencil akan sulit dijangkau oleh fasilitas-fasilitas kesejahteraan relatif sehingga kondisi tersebut menjadi penyebab taraf hidup masyarakat menjadi rendah yang selanjutnya menjadi penyebab pula adanya kemiskinan.

Dari kelima dimensi yang dikemukakan oleh Chambers tersebut, terdapat dua situasi ketidakberuntungan yang perlu mendapat perhatian khusus, diantaranya ialah ketidakberdayaan (*powerless*) dan kerentanan menghadapi situasi darurat (*state of emergency*). Hal tersebut menjadi perhatian khusus dikarenakan kedua situasi ketidakberuntungan ini sering menjadi penyebab seseorang atau sekelompok orang yang dikatakan miskin menjadi bertambah miskin. Untuk ketidakberdayaan dimana terdapat kasus-kasus mereka yang dikatakan kurang mampu tidak dapat melakukan perlawanan terhadap situasi yang tidak menguntungkan oleh pihak lain, seperti bantuan yang seharusnya diperuntukan kepada masyarakat miskin tapi tidak tersalurkan sebagaimana semestinya atau ketika mereka dipermainkan oleh pihak yang memiliki kekuasaan baik dari aspek politik, ekonomi, maupun sosial. Sedangkan, untuk kerentanan menghadapi situasi darurat ialah mereka yang dikatakan kurang mampu tidak dapat memenuhi kebutuhan akan hal yang tidak terduga, seperti bencana ataupun penyakit yang menyebabkan orang atau

sekelompok orang harus menjual harta benda miliknya sehingga orang atau sekelompok orang tersebut menjadi lebih miskin.

Sementara itu, Tjokrowinoto (dalam Sulistiyani, 2004: 27-28) juga mengemukakan pendapat yang sejalan dengan Chambers dimana kemiskinan jika dilihat dari sisi profil kemiskinan (*poverty profile*) masyarakat tidak hanya menyangkut persoalan kesejahteraan (*welfare*) semata, tetapi menyangkut pula persoalan akan kerentanan (*vulnerability*), ketidakberdayaan (*powerless*), peluang kerja yang sulit dicari, sebagian besar penghasilan yang diperoleh digunakan untuk memenuhi kebutuhan dasar/pokok, angka ketergantungan yang tinggi, rendahnya akses terhadap pasar, dan kemiskinan terefleksi dalam budaya kemiskinan yang diwarisi dari satu generasi ke generasi berikutnya. Berdasarkan pembahasan di atas mengenai pernyataan-pernyataan berkaitan dengan definisi tentang kemiskinan maka dapat disimpulkan bahwa kemiskinan merupakan suatu ketidakberdayaan menghadapi situasi yang berkaitan langsung dengan masalah ketidakmampuan dari aspek ekonomi untuk mencukupi kebutuhan dasar/pokok, seperti pangan, sandang, dan papan. Masalah tersebut terjadi dikarenakan seseorang ataupun sekelompok orang terjebak ke dalam kondisi ketidakmampuan pendapatan untuk mencukupi kebutuhan dasar/pokok sehingga kurang mampu untuk mencapai standar hidup atau kehidupan yang layak (Suryawati, 2005: 122). Selain itu, kemiskinan pula terjadi bukan hanya pada seseorang yang belum memiliki pekerjaan, tetapi juga terjadi pada mereka yang telah memiliki pekerjaan. Namun, pendapatan tersebut masih kurang untuk mencukupi kebutuhan dasar/pokok.

Kemiskinan merupakan masalah yang bersifat multidimensional, artinya konsep kemiskinan tidak hanya dipandang sebagai ketidakmampuan pendapatan dalam memenuhi kebutuhan dasar/pokok, tetapi juga dipandang sebagai ketidakberdayaan sebagai akibat rendahnya kualitas akan hal kesehatan dan pendidikan, rendahnya kebutuhan akan hal teknologi terutama sistem informasi dan komunikasi, serta kurangnya partisipasi organisasi sosial politik sehingga beresiko mendapatkan perlakuan negatif baik itu secara sosial maupun politik (Ala, 1981: 51). Akibatnya pula, negara-negara di dunia terus bergerak serta berusaha untuk

mengentaskan masalah kemiskinan. Melalui *Millennium Summit 2000* yang disetujui oleh 189 negara sepakat untuk membuat sebuah deklarasi *Millennium* yang mencakup kebebasan, keamanan, dan pembangunan termasuk penanggulangan kemiskinan dan kelaparan, lingkungan hidup, hak asasi manusia, serta *governance* (*United Nations Development Programme*, 2003: 27). Deklarasi yang dimaksudkan ialah deklarasi *Millennium Development Goals* (MDG) yang merupakan suatu indikator untuk dijadikan kerangka dalam mengukur kemajuan pembangunan antar negara di dunia dengan prioritas utama untuk memberantasan kemiskinan (*United Nations Development Programme*, 2003: 1).

2.1.2. Klasifikasi Kemiskinan

Berdasarkan kondisi kemiskinan yang dipandang sebagai bentuk masalah yang bersifat multidimensional, kemiskinan dapat dibagi dalam 4 (empat) macam bentuk (Nasikun dalam Suryawati, 2005: 122), yaitu:

1) Kemiskinan Relatif

Kemiskinan secara relatif merupakan kondisi kemiskinan yang terjadi karena akibat dari adanya pengaruh kebijakan pembangunan yang belum mampu menjangkau seluruh lapisan masyarakat sehingga menyebabkan ketimpangan terhadap distribusi pendapatan atau dengan kata lain terjadi perbedaan distribusi pendapatan antar kelompok masyarakat atau antar lapisan sosial dalam suatu daerah (Badan Pusat Statistik, 2008: 5). Di lain sisi, kemiskinan relatif dapat pula diartikan sebagai suatu kondisi masyarakat yang mana sudah dikatakan mampu dari segi pendapatan untuk memenuhi kebutuhan dasar/pokok minimum, tetapi masih jauh lebih rendah jika dibandingkan dengan keadaan masyarakat yang lain (Esmara dalam Ridlo, 2001: 10). Selain itu, ada pula istilah standar minimum yang disusun menurut Badan Pusat Statistik berdasarkan keadaan masyarakat pada waktu tertentu sesuai dengan kondisi hidup penduduk dalam suatu negara dan perhatian utama terfokus pada golongan penduduk miskin. Berkaitan dengan cara untuk mengidentifikasi dan menentukan sasaran penduduk miskin, garis kemiskinan relatif sudah cukup untuk dapat digunakan dengan pengecualian, yaitu perlu adanya penyesuaian

terhadap tingkat pembangunan dalam suatu negara secara menyeluruh. Namun, penggunaan garis kemiskinan relatif tidak dapat digunakan untuk membandingkan tingkat kemiskinan antar negara dan waktu dengan alasan karena tidak mencerminkan tingkat kesejateraan yang sama antar satu sama lain (Badan Pusat Statistik, 2008: 6).

2) Kemiskinan Absolut

Kemiskinan secara absolut merupakan kemiskinan dengan kondisi dimana seseorang atau sekelompok orang memiliki pendapatan kurang atau berada di bawah garis kemiskinan sehingga berakibat pada ketidakmampuan dalam mencukupi kebutuhan dasar/pokok minimum, seperti pangan, sandang, kesehatan, perumahan dan pendidikan yang diperlukan untuk memperoleh kehidupan yang lebih layak dan bermartabat (Badan Pusat Statistik, 2008: 6). Menurut Badan Pusat Statistik (2008: 6), garis kemiskinan merupakan suatu istilah yang digunakan untuk menyatakan seseorang atau sekelompok orang yang mana konsumsi rata-rata atau pengeluaran rata-rata digunakan untuk kebutuhan dasar/pokok minimum. Lebih lanjut Badan Pusat Statistik menerangkan bahwa garis kemiskinan absolut digunakan pula untuk menilai efek dari kebijakan anti kemiskinan antar waktu atau dengan kata lain untuk memperkirakan dampak dari suatu kejadian/proyek terhadap tingkat kemiskinan sehingga berguna dalam menganalisis kemajuan dalam memerangi kemiskinan. Selain itu, garis kemiskinan absolut juga digunakan oleh Bank Dunia untuk membandingkan angka kemiskinan antar satu negara dengan negara yang lain hanya jika garis kemiskinan absolut juga digunakan antar negara-negara yang menjadi perbandingan tersebut. Bagi Bank Dunia, penggunaan garis kemiskinan absolut memiliki manfaat dalam menentukan kemana arah dari aliran sumber dana finansial yang dapat tersalurkan sebagaimana semestinya.

3) Kemiskinan Struktural

Kemiskinan struktural merupakan kemiskinan yang diartikan sebagai suatu kondisi dimana tidak terpenuhinya kebutuhan asasi atau esensial sebagai manusia seperti kebutuhan subsistensi, afeksi, keamanan, identitas, proteksi,

kebebasan, partisipasi, dan waktu luang (Purwandari, 2011: 27). Selain itu, menurut Purwandari, konsep kemiskinan struktural dapat pula diartikan sebagai kondisi kemiskinan yang timbul sebagai akibat dari struktur sosial yang rumit menjadi masyarakat termarginalisasi dan sulit untuk memperoleh akses terhadap berbagai peluang. Sejalan dengan definisi yang dikemukakan oleh Puwandari, Suyanto (dalam Badan Pusat Statistik, 2008: 7-8) juga menjelaskan bahwa kemiskinan struktural merupakan kemiskinan yang ditengarai atau diduga bersebab dari kondisi struktur atau tatanan kehidupan yang tidak menguntungkan. Dikatakan tidak menguntungkan karena tatanan itu bukan hanya melahirkan kemiskinan, tetapi juga melanggengkan dan membiarkan kemiskinan tersebut terus terjadi di tengah masyarakat. Dengan kata lain, kemiskinan dengan kondisi tersebut tidak disebabkan oleh faktor-faktor alami ataupun faktor-faktor pribadi yang berasal dari diri seseorang itu sendiri, melainkan disebabkan oleh faktor yang berasal dari struktur atau tatanan sosial yang tidak adil. Tatanan sosial yang tidak adil tersebut menyebabkan masyarakat sulit untuk mengakses sumber daya yang dibutuhkan untuk mengembangkan serta meningkatkan kualitas hidup masyarakat tersebut.

4) Kemiskinan Kultural

Kemiskinan kultural atau dapat disebut pula sebagai kemiskinan budaya merupakan suatu bentuk kemiskinan yang mengacu pada sikap dan perilaku hidup seseorang atau sekelompok orang yang disebabkan oleh gaya hidup, kebiasaan hidup, dan budaya di mana mereka merasa hidup berkecukupan dan tidak merasa kekurangan (Rustanti, 2015: 146). Selain itu, Badan Pusat Statistik (2008: 8) juga mengungkapkan bahwa kemiskinan kultural merupakan kemiskinan yang diakibatkan oleh faktor-faktor adat dan budaya suatu daerah tertentu yang relatif tidak mau untuk memperbaiki taraf hidup dengan tata cara modern sehingga menjebak seseorang atau masyarakat ke dalam kemiskinan. Namun, kemiskinan dengan bentuk tersebut menurut Badan Pusat Statistik seyogyanya bisa dikurangi atau bahkan secara bertahap dapat dihilangkan dengan cara mengabaikan faktor-faktor yang menjadi penyebab kemiskinan,

seperti adat dan budaya pada daerah tertentu yang menghalangi seseorang untuk melakukan perubahan-perubahan ke arah tingkat kehidupan yang lebih baik.

Di antara keempat bentuk kemiskinan yang telah dibahas tersebut, terdapat salah satu bentuk kemiskinan yang paling banyak mendapatkan perhatian di bidang ilmu sosial, yaitu kemiskinan struktural. Hal tersebut dikarenakan kemiskinan struktural merupakan salah satu bentuk kemiskinan yang dapat menimbulkan atau yang menjadi penyebab tumbuh dan berkembangnya ketiga bentuk kemiskinan yang lain, yaitu kemiskinan absolut, kemiskinan relatif, dan kemiskinan kultural (Jarnasy, 2004: 8-9).

2.1.3. Karakteristik Kemiskinan

Pembahasan yang akan dipaparkan pada bagian ini adalah teori mengenai karakteristik kemiskinan. Secara umum, terdapat lima (5) karakteristik atau ciri-ciri kelompok (penduduk) miskin menurut Salim (dalam Suryawati, 2005: 123). Kelima karakteristik penduduk miskin yang dimaksud tersebut, yaitu:

- 1) Rata-rata tidak memiliki faktor produksi sendiri, seperti tanah, modal, peralatan kerja, maupun keterampilan yang memadai.
- 2) Pada umumnya, tingkat pendidikan yang dimiliki oleh masyarakat tersebut relatif rendah.
- 3) Status pekerjaan lebih banyak atau didominasi oleh wirausaha yang mana modal yang dimiliki serta lingkup kerja yang ada relatif lebih kecil. Keadaan yang demikian tersebut dapat pula disebut sebagai usaha kecil-kecilan atau masyarakat bekerja dalam sektor informal sehingga terkadang mereka diartikan pula sebagai pekerja yang berada dalam posisi setengah menganggur.
- 4) Letak tempat tinggal yang sangat jauh dari pusat-pusat pertumbuhan regional atau lebih banyak berada di dalam kawasan pedesaan atau kawasan tertentu di daerah perkotaan (*slum area*).
- 5) Memiliki kesempatan yang relatif rendah untuk memperoleh (dalam jumlah yang cukup) bahan kebutuhan pokok, pakaian, perumahan, fasilitas kesehatan, air minum, pendidikan, angkutan, fasilitas komunikasi, dan kesejahteraan sosial lainnya.

2.1.4. Penyebab Kemiskinan

Kemiskinan merupakan masalah yang bersifat multidimensional, artinya kemiskinan bukan hanya dipandang sebagai ketidakmampuan dalam aspek pendapatan yang digunakan untuk memenuhi kebutuhan dasar/pokok, tetapi juga dipandang sebagai ketidakberdayaan sebagai akibat rendahnya kualitas akan hal kesehatan dan pendidikan, rendahnya kebutuhan akan hal teknologi terutama sistem informasi dan komunikasi, serta kurangnya partisipasi organisasi sosial politik sehingga beresiko mendapatkan perlakuan negatif baik itu secara sosial maupun politik (Ala, 1981: 51). Dengan kata lain, kemiskinan bukan hanya dipandang dalam aspek pendapatan saja, tetapi juga dipandang dalam aspek yang cukup luas, yakni meliputi aspek sosial, budaya, politik, lingkungan (alam dan geografis), kesehatan, pendidikan, agama, dan budi pekerti.

Di negara-negara yang dikatakan sudah maju, kemiskinan lebih bersifat individual, artinya kemiskinan tersebut disebabkan karena seseorang mengalami kecacatan baik itu secara fisik maupun mental, ketuaan, sakit yang sedemikian parah, dan lain sebagainya. Sedangkan di negara-negara berkembang seperti di Indonesia, kemiskinan lebih disebabkan karena adanya faktor-faktor yang kurang menguntungkan masyarakat atau penduduk dalam suatu negara yang bersangkutan seperti kebijakan ekonomi yang hanya dikuasai mereka yang memiliki kedudukan lebih tinggi serta sistem atau struktur politik yang tidak adil (Suharto, 2009: 17).

Dalam Paket Informasi Dasar yang disusun oleh Lembaga Penelitian SMERU (2001: 4) menyebutkan bahwa faktor-faktor yang menjadi dasar penyebab terjadinya kemiskinan itu, adalah sebagai berikut:

- 1) Kegagalan kepemilikan, terutama tanah dan modal.
- 2) Terbatasnya ketersediaan bahan kebutuhan dasar, sarana, dan prasarana.
- 3) Kebijakan pembangunan yang bias perkotaan dan bias sektor.
- 4) Adanya perbedaan kesempatan di antara anggota masyarakat dari sistem yang kurang mendukung.
- 5) Adanya perbedaan sumber daya manusia dan perbedaan antar sektor ekonomi (ekonomi tradisional versus ekonomi modern).

- 6) Rendahnya produktivitas dan tingkat pembentukan modal dalam masyarakat.
- 7) Budaya hidup yang dikaitkan dengan kemampuan seseorang mengelola sumber daya alam dan lingkungannya.
- 8) Tidak adanya tata pemerintahan yang bersih dan baik (*good governance*).
- 9) Pengelolaan sumber daya alam yang berlebihan dan tidak berwawasan lingkungan.

Selain beberapa faktor tersebut yang telah disebutkan oleh Lembaga Penelitian SMERU, penyebab timbulnya masalah kemiskinan juga diungkapkan oleh Bradshaw (2006: 6-14) yang mana dipaparkan sebagai berikut:

- 1) Kemiskinan yang disebabkan oleh Kelemahan Individu

Teori pertama tentang kemiskinan ini menjelaskan bahwa kemiskinan memiliki makna yang sangat luas dan beragam dimana berfokus pada individu yang bertanggung jawab atas kemiskinan mereka sendiri. Selain itu, para ahli lainnya menganggap bahwa suatu individu tersebut kurang dalam hal bekerja atau kemampuan individu secara genetis relatif rendah seperti intelegensi yang kurang mengakibatkan timbulnya masalah kemiskinan.

- 2) Kemiskinan yang disebabkan oleh Sistem Kebudayaan yang mendukung Sub-Budaya Miskin

Teori kedua ini menunjukkan bahwa kemiskinan tersebut timbul akibat dari transmisi kepercayaan, nilai-nilai, sikap-sikap atau karakter individual, serta kemampuan sosial dari generasi ke generasi selanjutnya. Budaya kemiskinan ini pula sangat sulit untuk dihilangkan/dihapuskan jika telah masuk atau menjadi bagian dari kehidupan masyarakat.

- 3) Kemiskinan yang disebabkan oleh Ekonomi, Politik, dan Diskriminasi

Dalam teori ketiga ini menjelaskan bahwa kemiskinan terjadi dikarenakan oleh kondisi ekonomi suatu individu atau masyarakat yang lemah sehingga masyarakat kurang terlibat dalam urusan politik. Kondisi tersebut juga menimbulkan adanya perbedaan sosial atau diskriminasi antar masyarakat sehingga mereka memiliki kesempatan dan sumber daya yang terbatas dalam upaya meningkatkan kesejahteraan dan taraf hidup yang lebih layak.

4) Kemiskinan yang disebabkan oleh Perbedaan Kondisi Geografis

Penyebab kemiskinan pada teori keempat ini diakibatkan oleh kondisi geografis yang kurang menguntungkan dimana seseorang ataupun sekelompok orang yang tinggal di suatu daerah tertentu memiliki sumber daya yang relatif sedikit/kurang yang dibutuhkan untuk mengembangkan serta meningkatkan kualitas hidup seseorang atau sekelompok orang tersebut.

5) Kemiskinan yang disebabkan oleh Siklus Ketergantungan

Teori kelima ini menjelaskan mengenai siklus saling ketergantungan antara individu dengan sumber daya masyarakat dimana individu yang tidak dapat memiliki atau menguasai sumber daya masyarakat tidak akan mampu untuk berpartisipasi dalam kegiatan ekonomi. Kondisi yang demikian menyebabkan kemampuan untuk membayar pajak menjadi menurun dan akibatnya timbul kemiskinan.

Berdasarkan kedua pernyataan yang telah dibahas sebelumnya baik yang dikemukakan oleh Lembaga Penelitian SMERU maupun Bradshaw maka dapat disimpulkan bahwa penyebab terjadinya kemiskinan adalah adanya kegagalan dalam hal kepemilikan aset, kekurangan sumber daya baik secara kualitas maupun kuantitas, kondisi geografis yang kurang mendukung, kerentanan dan ketidakmampuan dalam menghadapi situasi sulit, rendahnya partisipasi dalam institusi negara dan masyarakat, serta budaya dan siklus yang menyebabkan masalah kemiskinan terus berulang dari generasi ke generasi berikutnya.

2.1.5. Indikator Kemiskinan

Pemerintah dalam upaya menekan laju pertumbuhan kemiskinan telah dimulai sejak tahun 1970-an, tetapi pada tahun tersebut belum ada kebijakan perlindungan sosial secara spesifik di Indonesia (Tim Nasional Percepatan Penanggulangan Kemiskinan, 2015: 4). Namun, setelah terjadi krisis moneter Asia pada tahun 1997 – 1998, pemerintah merancang dan melaksanakan program-program yang khusus ditujukan untuk penduduk miskin seperti Program Jaring Pengaman Sosial (JPS). Tujuan dilaksanakan Program Jaring Pengaman Sosial (JPS) ialah untuk memberikan perlindungan dalam bidang pangan, kesehatan,

pendidikan, dan akses kepada sumber pendapatan untuk penduduk yang sudah miskin sebelum terjadi krisis dan penduduk yang menjadi miskin akibat dampak krisis. Sasaran penerima program tersebut ialah mereka yang dikatakan sejahtera berdasarkan variabel-variabel yang ditetapkan oleh Badan Pusat Statistik (BPS) pada tahun 2015 sebagai indikator penentu tingkat kesejahteraan rumah tangga (Tim Nasional Percepatan Penanggulangan Kemiskinan, 2010: 62-66; 2013: 10). Adapun variabel yang digunakan oleh Badan Pusat Statistik tersebut yang digunakan sebagai indikator untuk mengukur tingkat kesejahteraan rumah tangga dapat diberikan sebagaimana yang tercantum pada tabel 2.2 berikut ini.

Tabel 2.1 Variabel dan Pemutakhiran Basis Data Terpadu Tahun 2015
(PBDT 2015)

Variabel Individu (Anggota Rumah Tangga)	Variabel Rumah Tangga
Nama	Kepemilikan Bangunan Tempat Tinggal
Umur	Kepemilikan Lahan Tempat Tinggal
Jenis Kelamin	Luas Lantai
Status Perkawinan	Jenis Lantai Terluas
Hubungan dengan Kepala Rumah Tangga dan Keluarga	Jenis Dinding Terluas
Kepemilikan Kartu Identitas	Jenis Atap Terluas
Ketercantuman pada Kartu Keluarga	Jumlah Kamar Tidur
Kepemilikan Akta Nikah	Sumber Air Minum
Kecacatan	Cara Memperoleh Air Minum
Penyakit Kronis	Sumber Penerangan Utama
Status Kehamilan	Bahan Bakar/Energi Utama Memasak
Partisipasi Sekolah	Kepemilikan Jamban
Pendidikan Tertinggi	Jenis Kloset
Lama Sekolah	Tempat Pembuangan Akhir
Mempunyai Pekerjaan	Kepemilikan Aset

Lapangan Usaha	Keikutsertaan Program Keluarga Berencana, Klaster 1, Klaster 2, dan Klaster 3
Status Pekerjaan	

Sumber: PBDT 2015

Data hasil dari Pemutakhiran Basis Data Terpadu (PBDT) 2015 akan digunakan untuk menyalurkan bantuan dari pemerintah melalui program-program perlindungan sosial yang telah ditentukan dan ditetapkan oleh Badan Pusat Statistika melalui indikator penentu tingkat kesejahteraan rumah tangga atau dengan kata lain indikator untuk menentukan masyarakat kurang mampu yang berhak menerima bantuan dari pemerintah.

2.1.6. Strategi Penanggulangan Kemiskinan

Strategi untuk mengatasi kemiskinan tidak lepas dari strategi pembangunan yang diterapkan oleh masing-masing negara, tidak terkecuali Negara Indonesia. Di Indonesia, strategi penanggulangan kemiskinan didasarkan pada 4 (empat) prinsip (Tim Nasional Percepatan Penanggulangan Kemiskinan, 2010: 51-61), yaitu:

1) Pembangunan yang Inklusif

Konsep ini mengandung pengertian bahwa pertumbuhan harus mampu menciptakan lapangan kerja produktif dalam jumlah yang besar sehingga akan ada peningkatan pendapatan pada mayoritas penduduk, peningkatan taraf hidup, serta pengurangan terhadap angka kemiskinan. Selain itu, untuk mencapai pertumbuhan ekonomi yang tinggi dan berkelanjutan perlu diciptakan iklim usaha yang kondusif di dalam negeri, yakni dengan cara mendorong atau mendukung usaha mikro, kecil, dan menengah (UMKM) untuk terus menciptakan nilai tambah, termasuk melalui pasar ekspor. Paradigma pembangunan inklusif mengandung arti lebih dari sekedar melihat hasil pembangunan secara agregat, artinya indikator-indikator pembangunan seperti peningkatan pendapatan per kapita tidak hanya dievaluasi secara agregat, tetapi juga berdasarkan kelompok dalam penduduk seperti pendapatan, daerah, sektor,

gender dan lainnya. Pembangunan inklusif juga perlu dipahami dalam konteks kewilayahan serta prioritas lebih besar pada sektor pedesaan dan pertanian. Dalam konteks kewilayahan, setiap daerah di Indonesia dapat berfungsi sebagai pusat pertumbuhan dengan sumber daya dan komoditi unggulan yang berbeda sehingga perekonomian daerah tersebut akan membentuk karakteristik perekonomian secara nasional dimana secara tidak langsung akan memperkuat ekonomi domestik. Prioritas utama pada sektor pedesaan dan pertanian ialah bertujuan agar penduduk di desa dan yang bekerja di sektor pertanian bisa lebih produktif dengan cara meningkatkan investasi publik di bidang sarana dan prasarana penunjang produktifitas seperti pengairan, teknologi baik teknologi pertanian maupun teknologi non-pertanian seperti teknologi informasi dan komunikasi. Hal tersebut diperlukan agar para petani dapat memantau pergerakan harga, kondisi cuaca, maupun jalur-jalur pemasaran produk. Selain itu, perlu juga adanya peningkatan di bidang industri dan jasa, pembangunan jalan di desa serta sarana dan prasarana transportasi dengan tujuan untuk mempermudah, memperlancar, dan memperluas akses melalui pedesaan.

2) Akses Masyarakat Miskin terhadap Pelayanan Dasar

Memperbaiki akses masyarakat miskin terhadap pelayanan pendidikan, kesehatan air bersih dan sanitasi, serta pangan dan gizi akan membantu mengurangi biaya yang harus dikeluarkan. Selain itu, akses terhadap pelayanan dasar akan berperan dalam meningkatkan kualitas modal manusia (*human capital*) terutama di kalangan masyarakat miskin.

3) Pemberdayaan Kelompok Masyarakat Miskin

Pemberdayaan kelompok miskin pada hakikatnya adalah berusaha untuk meningkatkan taraf hidup penduduk miskin tanpa membuat penduduk tersebut bergantung pada program bantuan. Program penanggulangan kemiskinan yang bersifat karitatif mungkin akan menyelesaikan kemiskinan secara sesaat, tapi pendekatan seperti ini tidak bertahan di jangka panjang, sebab penduduk miskin cenderung jadi tergantung pada program penanggulangan kemiskinan yang diterapkan oleh pemerintah. Di sisi lain, pemberdayaan kelompok masyarakat miskin juga bertujuan agar kelompok tersebut bisa lebih produktif dan dapat

mengembangkan potensi diri maupun bersama, misalnya dengan mengadakan program kewirausahaan di kalangan penduduk miskin.

4) Memerbaiki dan Mengembangkan Sistem Perlindungan Sosial bagi Penduduk Miskin dan Rentan

Sistem perlindungan sosial yang dimaksud ialah program-program yang diadakan pemerintah dengan tujuan untuk membantu individu dan masyarakat dalam menghadapi guncangan-guncangan (*shocks*) dalam hidup, seperti jatuh sakit, kematian anggota keluarga, kehilangan pekerjaan, ditimpa bencana atau bencana alam, dan sebagainya. Sistem perlindungan sosial yang efektif akan mengantisipasi agar seseorang atau masyarakat yang mengalami guncangan tidak sampai jatuh miskin.

Selain strategi penanggulangan kemiskinan yang ditetapkan oleh Tim Nasional Percepatan Penanggulangan Kemiskinan, adapula strategi atau upaya lain dalam menanggulangi kemiskinan, yaitu partisipasi aktif seluruh masyarakat melalui sebuah gerakan massif. Gerakan ini dilakukan dengan alasan bahwa upaya penanggulangan kemiskinan bukan hanya menjadi tanggung jawab pemerintah, tetapi menjadi tanggung jawab bersama yang mana masyarakat ikut andil dalam menanggulangi kemiskinan. Hal tersebut ditinjau melalui program Gerakan Terpadu Penanggulangan Kemiskinan (Gerdu Taskin) yang dicanangkan pemerintah sejak tahun 1998. Menurut Kadji (2013: 5), Gerakan Terpadu Penanggulangan Kemiskinan (Gerdu Taskin) merupakan upaya penanggulangan kemiskinan yang terpadu dan menyeluruh yang dilakukan oleh pemerintah, kalangan swasta, lembaga swadaya masyarakat (NGO), dan organisasi masyarakat, masyarakat luas, serta keluarga miskin itu sendiri. Lalu, sejak tahun 1998/1999 diimplementasikan kebijakan Program Pengembangan Kecamatan (PPK), dan Program Penanggulangan Kemiskinan Perkotaan (P2KP), serta muncul Program Nasional Pemberdayaan Masyarakat baik itu di tingkat kecamatan (PNPM-PPK) maupun di tingkat kota (PNPM-P2KP) yang mana secara substantif menggerakkan masyarakat untuk berpartisipasi aktif dan ikut serta dalam gerakan penanggulangan kemiskinan. Sehubungan dengan peran pemerintah dalam setiap program penanggulangan kemiskinan tersebut, Sumodiningrat (1999: 202) mengatakan

bahwa program pemberdayaan masyarakat tersebut dirancang oleh pemerintah dalam rangka untuk memecahkan 3 (tiga) masalah utama dalam pembangunan, yaitu: pengangguran, ketimpangan, serta penanggulangan kemiskinan. Dalam penanggulangan kemiskinan, upaya penanggulangan kemiskinan melalui beberapa kebijaksanaan pemberdayaan masyarakat tersebut bertujuan untuk memberikan ruang gerak, fasilitas publik, serta kesempatan-kesempatan yang kondusif bagi masyarakat yang memiliki kemampuan ataupun kemungkinan kelompok masyarakat miskin untuk mengatasi masalah mereka sendiri dan bukan menekan dan mendesak mereka ke pinggir-pinggir atau ke posisi-posisi ketergantungan. Jadi, dengan kata lain program-program pemerintah yang bertujuan untuk menanggulangi kemiskinan akan berjalan dengan optimal jika semua pihak ataupun kalangan baik itu pemerintah maupun masyarakat dapat bekerja sama antara satu sama lainnya sehingga manfaat dari program-program penanggulangan kemiskinan tersebut dapat tersalurkan secara merata dan tepat sasaran dapat bermanfaat bagi pihak-pihak yang menjadi penerima program bantuan dari pemerintah.

2.2. Pendidikan

Pada sub bab ini, akan membahas teori mengenai definisi pendidikan, fungsi pendidikan, tujuan pendidikan, serta jenis pendidikan. Adapun pemaparan lebih lanjut mengenai pendidikan dapat diberikan sebagai berikut.

2.2.1. Definisi Pendidikan

Pendidikan merupakan suatu usaha secara sadar dan terencana untuk mewujudkan suasana belajar dan proses pembelajaran agar peserta didik secara aktif mengembangkan potensi dirinya untuk memiliki kekuatan spiritual keagamaan, pengendalian diri, kepribadian, kecerdasan, akhlak mulia, serta keterampilan yang diperlukan dirinya, masyarakat, bangsa, maupun negara (UU Sistem Pendidikan No. 20 Tahun 2003 Bab 1 Pasal 1 Ayat 1). Sedangkan menurut Hasibuan (dalam Flippo, 2002:69) yang menyebutkan bahwa definisi dari pendidikan berhubungan dengan peningkatan pengetahuan umum dan pemahaman

atas lingkungan kita secara menyeluruh. Selain itu, ada pula pengertian pendidikan menurut Pusat Bahasa Departemen Pendidikan Nasional (2002: 263), yakni suatu proses pengubahan sikap dan tatalaku seseorang atau sekelompok orang dalam usaha untuk mendewasakan manusia melalui upaya pengajaran dan pelatihan, proses, cara, dan perbuatan mendidik.

2.2.2. Fungsi dan Tujuan Pendidikan

Berdasarkan UU Nomor 20 Tahun 2003 tentang Sistem Pendidikan Nasional Bab 2 bahwa fungsi dari pendidikan ialah untuk mengembangkan kemampuan dan membentuk watak serta peradaban bangsa yang bermartabat dalam rangka mencerdaskan kehidupan bangsa.

Sedangkan tujuan dari pendidikan ialah untuk berkembangnya potensi peserta didik agar menjadi manusia yang beriman dan bertakwa kepada Tuhan Yang Maha Esa, berakhlak mulia, sehat, berilmu, cakap, kreatif, mandiri, dan menjadi warga negara yang demokratis serta bertanggung jawab.

2.2.3. Jalur Pendidikan

Jalur pendidikan menurut UU Nomor 20 Tahun 2003 Bab 4 mengenai jalur, jenjang, dan jenis pendidikan ialah terdiri atas pendidikan formal, pendidikan nonformal, serta pendidikan informal. Adapun pemaparan lebih lanjut mengenai jalur pendidikan dapat dijelaskan sebagai berikut ini.

2.2.3.1. Pendidikan Formal

Pendidikan formal merupakan pendidikan yang memiliki susunan yang terstruktur, memiliki jenjang atau tingkatan, dalam periode waktu-waktu tertentu, serta berlangsung dari jenjang sekolah dasar hingga jenjang perguruan tinggi dan tercakup di samping studi akademis umum, juga berbagai program khusus dan lembaga guna untuk latihan tenis dan professional (UU Nomor 20 Tahun 2003).

2.2.3.2. Pendidikan Nonformal

Pendidikan informal merupakan suatu proses jalur pendidikan di luar pendidikan formal yang dapat dilaksanakan secara terstruktur dan berjenjang, baik tersendiri maupun merupakan bagian dari suatu kegiatan yang luas, yang dimaksudkan untuk memberikan layanan kepada sasaran didik tertentu dalam mencapai tujuan-tujuan belajar (UU Nomor 20 Tahun 2003).

2.2.3.3. Pendidikan Informal

Pendidikan informal merupakan pendidikan yang terjadi seumur hidup karena pendidikan ini dapat diperoleh dari pengalaman sehari-hari dan jalur pendidikan keluarga ataupun pengaruh lingkungan sekitar, baik tetangga, pekerjaan, perpustakaan ataupun media massa (UU Nomor 20 Tahun 2003).

2.3. Sistem Pendukung Keputusan

Dalam sub bab ini, akan dibahas teori mengenai sistem pendukung keputusan yang mencakup definisi dari sistem pendukung keputusan, tujuan dari sistem pendukung keputusan, kerangka kerja sistem pendukung keputusan, karakteristik dan kemampuan sistem pendukung keputusan, tahapan pengambilan keputusan, serta manfaat atau keuntungan dari penggunaan sistem pendukung keputusan.

2.3.1. Definisi Sistem Pendukung Keputusan

Konsep sistem pendukung keputusan (decision support system) pertama kali dikemukakan oleh G. Anthony Gorry dan Michael S. Scott-Morton pada tahun 1971. Dalam artikel yang berjudul "A Framework for Management Information Systems" yang diterbitkan pada tahun 1971 di Sloan Management Review, mengungkapkan bahwa sistem pendukung keputusan merupakan sistem penghasil informasi berbasis komputer yang ditujukan untuk membantu manager dalam mengambil suatu keputusan dan memecahkan berbagai masalah baik dalam situasi terstruktur, semi terstruktur, maupun tidak terstruktur (McLeod, 2008: 331). Sejalan dengan konsep McLeod, Bonczek, dkk., dalam buku yang berjudul Decision

Support Systems And Intelligent Systems (Turban, 2005: 137) juga mendefinisikan istilah dari sistem pendukung keputusan yang dinyatakan sebagai sebuah sistem informasi berbasis komputer yang terdiri dari 3 (tiga) komponen yang saling berinteraksi, yakni: 1) sistem bahasa (mekanisme untuk memberikan komunikasi antara pengguna dan komponen sistem pendukung keputusan), 2) sistem pengetahuan (respositori pengetahuan domain masalah yang ada pada sistem pendukung keputusan atau sebagai data atau sebagai prosedur), dan 3) sistem pemrosesan masalah (hubungan antara dua komponen lainnya yang terdiri dari satu atau lebih kapabilitas manipulasi masalah umum yang diperlukan untuk pengambilan keputusan).

Sistem pendukung keputusan sebagai suatu sistem berbasis komputer juga digunakan untuk membantu dalam mengambil suatu keputusan di antara beberapa alternatif yang ada (Black, 2009: 2). Sejalan dengan pendapat tersebut, Daihani (2001: 55) juga mengungkapkan bahwa sistem pendukung keputusan merupakan suatu sistem berbasis komputer yang menghasilkan berbagai alternatif keputusan untuk membantu manajemen dalam menangani berbagai permasalahan yang terstruktur ataupun tidak terstruktur dengan menggunakan data dan model. Selain itu, menurut pendapat Alter (dalam Kusriani, 2007: 15-16) yang mengungkapkan definisi sistem pendukung keputusan merupakan suatu sistem informasi interaktif yang menyediakan informasi, pemodelan, serta pemanipulasian data dengan tujuan untuk membantu pengambilan keputusan dalam situasi yang semi terstruktur dan situasi yang terstruktur dimana tidak seorang pun tahu pasti bagaimana keputusan seharusnya dibuat.

Dengan demikian, berdasarkan dari beberapa pendapat ahli mengenai definisi dari sistem pendukung keputusan, maka dapat disimpulkan bahwa sistem pendukung keputusan merupakan suatu sistem penghasil informasi dan beberapa alternatif keputusan berbasis komputer / computer based information system (CBIS) yang terdiri dari tiga komponen, yakni sistem bahasa, sistem pengetahuan, serta sistem pemrosesan dimana dirancang untuk membantu seorang atau para pengambil keputusan dalam memecahkan berbagai masalah baik dalam situasi

terstruktur, semi terstruktur, maupun tidak terstruktur. Dengan kata lain, sistem pendukung keputusan dibangun dengan tujuan untuk mendukung solusi atas suatu masalah atau untuk mengevaluasi suatu peluang. Namun, biasanya sistem pendukung keputusan lebih ditujukan untuk mendukung manajemen dalam melakukan pekerjaan yang bersifat analitis dalam situasi yang kurang terstruktur dan dengan kriteria yang belum pasti.

2.3.2. Tujuan Sistem Pendukung Keputusan

Sistem pendukung keputusan dirancang bukan untuk mengotomatisasikan pengambilan suatu keputusan dari berbagai alternatif yang ada, tetapi memberikan suatu perangkat interaktif yang memungkinkan pengambil keputusan untuk melakukan berbagai analisis dan solusi terhadap masalah yang sedang dihadapi menggunakan metode-metode yang tersedia. Menurut Nurdin (dalam Nofriansyah, 2014: 4-5) yang menyebutkan bahwa ada beberapa tujuan yang menjadi dasar dirancangnya sistem pendukung keputusan, yakni sebagai berikut:

- 1) Membantu manajer dalam pengambilan keputusan atas masalah semi terstruktur.
- 2) Memberikan dukungan atas pertimbangan manajer dan bukan dimaksudkan untuk menggantikan fungsi manajer.
- 3) Meningkatkan efektivitas keputusan yang diambil oleh manajer lebih daripada perbaikan efisiensinya.
- 4) Kecepatan komputasi yang memungkinkan para pengambil keputusan untuk melakukan banyak hal dalam kegiatan komputasi secara cepat dengan biaya yang relatif rendah.
- 5) Peningkatan mutu produktivitas.
- 6) Dukungan terhadap mutu kualitas pengambilan keputusan.
- 7) Teknologi pengambilan keputusan dapat menciptakan suatu daya saing antar sesama pengambil keputusan untuk lebih meningkatkan mutu dan kualitas keputusan yang dibuat.
- 8) Mengatasi keterbatasan kognitif dalam pemrosesan serta penyimpanan.

Jadi, dengan adanya sistem pendukung keputusan diharapkan akan mampu mengatasi masalah-masalah terutama masalah dalam hal membantu manajer untuk mengambil suatu keputusan, mendukung keputusan yang diambil oleh manajer, serta meningkatkan efektivitas dari pengambilan keputusan yang dibuat tersebut.

2.3.3. Kerangka Kerja Sistem Pendukung Keputusan

Menurut Gorry dan Scott Morton yang menggabungkan kerja Simon dan Anthony (dalam Turban, 2005: 15-18), mengusulkan kerangka kerja klasik untuk pendukung keputusan dimana dalam kerangka kerja tersebut juga menggambarkan beberapa isu tambahan, seperti hubungan antara teknologi dan evolusi sistem terkomputerisasi. Berikut proses-proses yang terjadi pada kerangka kerja sistem pendukung keputusan:

1) Terstruktur

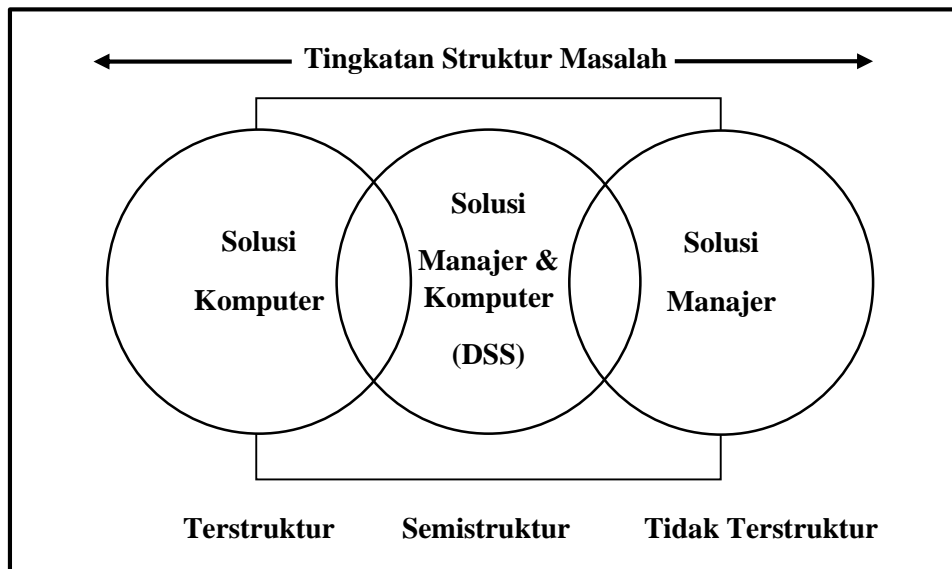
Pada umumnya proses terstruktur merupakan suatu proses yang terjadi secara rutin, berulang-ulang, serta dengan prosedur yang sudah pasti/jelas sehingga mudah dalam menyelesaikan berbagai masalah pengambilan suatu keputusan menggunakan metode dengan solusi standar. Selain itu, pengambil keputusan dapat menggunakan dukungan klerikal/administrasi, pemrosesan data/transaksi, atau model-model ilmu yang berkaitan dengan manajemen untuk menyelesaikan masalah terstruktur.

2) Tidak Terstruktur

Pada proses tidak terstruktur, masalah yang terjadi tidak dapat diprediksi serta bersifat sangat kompleks sehingga tidak ada metode penyelesaian solusi secara pasti. Jadi, instuisi manusia sering menjadi dasar dalam mengambil suatu keputusan. Selain itu, masalah-masalah tidak terstruktur umumnya mencakup perencanaan layanan baru, rekrutmen eksekutif, serta memilih satu set riset dan proyek pengembangan untuk tahun mendatang. Dengan kata lain, dukungan yang dapat digunakan pada masalah tidak terstruktur ialah alat pendukung keputusan lanjutan, seperti sistem pakar, sistem pendukung kelompok (GGS), dan sistem manajemen pengetahuan (KMS).

3) Semiterstruktur

Proses semiterstruktur merupakan proses yang berada antara proses terstruktur dan tidak terstruktur serta memiliki beberapa elemen/unsur dari proses terstruktur dan tidak terstruktur. Pemecahan masalah semiterstruktur tersebut meliputi prosedur solusi standar dan penilaian manusia dimana keduanya merupakan kombinasi dari masalah terstruktur dan tidak terstruktur. Sebagai contoh untuk permasalahan semiterstruktur ini ialah perdagangan saham, manajemen anggaran pemasaran untuk produk konsumen, dan melakukan analisis akuisisi modal.



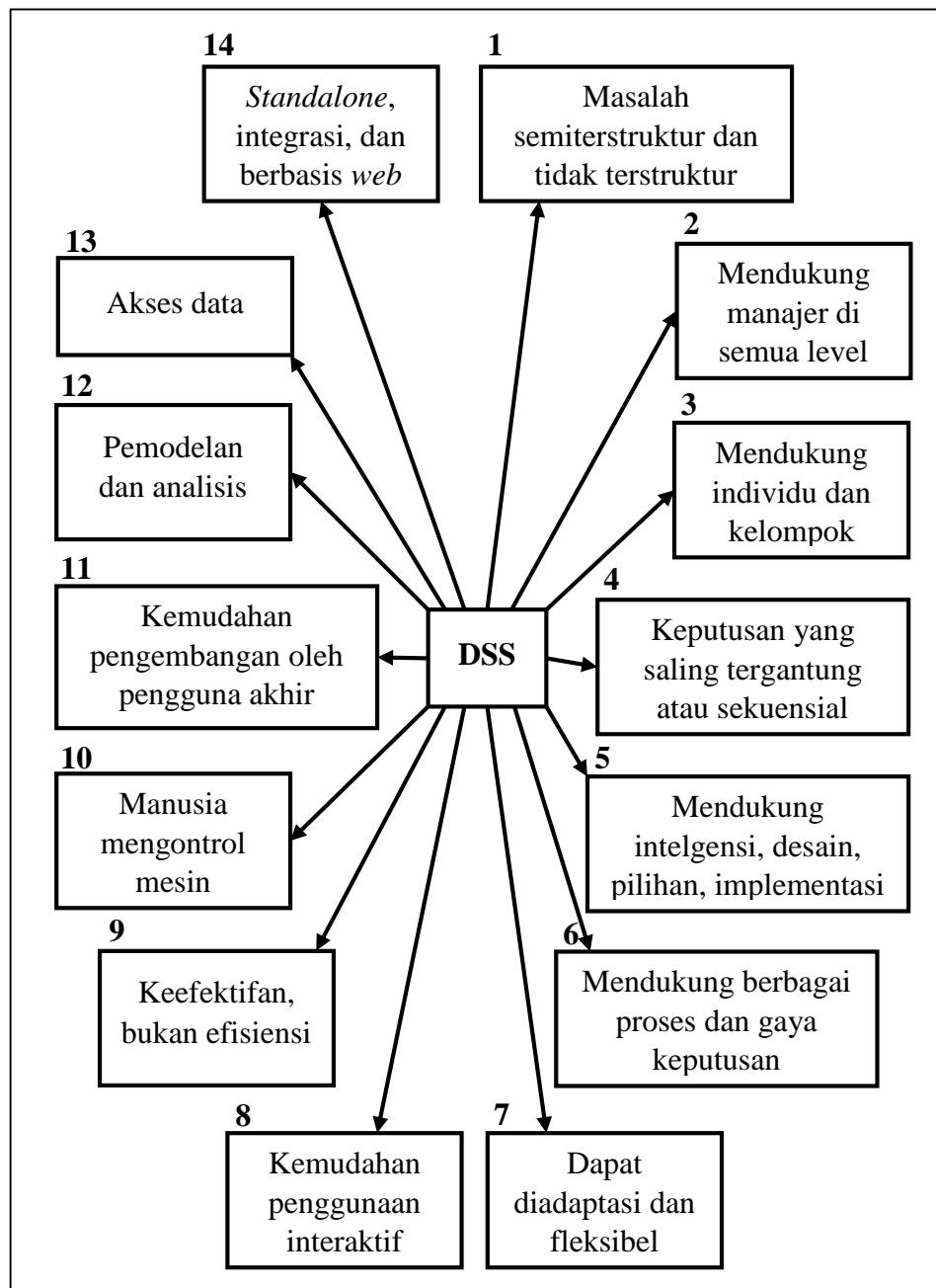
Gambar 2.1 DSS Berfokus pada Masalah Semiterstruktur (McLeod, 2001: 61)

Sistem pendukung keputusan (*decision support system*) menyediakan berbagai model dalam memecahkan masalah yang berkaitan dengan pengambilan keputusan terutama masalah semistruktur. Menurut McLeod (2001: 60), DSS tidak dimaksudkan untuk menggantikan tugas manajer. Pada gambar 2.1 yang mana menggambarkan hubungan antara struktur masalah dengan tingkat dukungan yang dapat disediakan oleh komputer. Komputer dapat diterapkan pada bagian masalah yang terstruktur, tetapi manajer bertanggung jawab atas bagian yang tidak terstruktur (menerapkan penilaian atau intuisi dan melakukan analisis). Manajer dan komputer bekerja sama sebagai tim pemecahan masalah dalam memecahkan masalah yang berada di area semi-terstruktur yang luas. Jadi, untuk masalah

tersebut, sistem pendukung keputusan dapat meningkatkan kualitas informasi yang menjadi basis pengambilan keputusan dengan cara memberikan solusi bukan hanya satu solusi tunggal, tetapi juga serangkaian solusi alternatif (Turban, 2005: 16). Hal tersebut dapat membantu manajer/pengambil keputusan untuk memahami dengan lebih baik masalah yang sedang terjadi sehingga mampu membuat keputusan menjadi lebih baik dan berkualitas.

2.3.4. Karakteristik dan Kemampuan Sistem Pendukung Keputusan

Berdasarkan definisi dari sistem pendukung keputusan itu sendiri, maka dapat diketahui karakteristik dan kemampuan dari sistem pendukung keputusan. Berikut karakteristik dan kemampuan dari sistem pendukung keputusan yang dijabarkan melalui gambar di bawah ini (Turban, 2005: 141-143):



Gambar 2.2 Karakteristik dan Kemampuan Kunci dari DSS

- 1) Sistem pendukung keputusan menyediakan dukungan bagi para pengambil keputusan, terutama pada situasi semi terstruktur dan tidak terstruktur dengan mengombinasikan pertimbangan manusia dan informasi terkomputerisasi.
- 2) Dukungan disediakan untuk berbagai level manajerial yang berbeda yang dimulai dari level pimpinan puncak/tertinggi hingga ke level manajer lapangan.

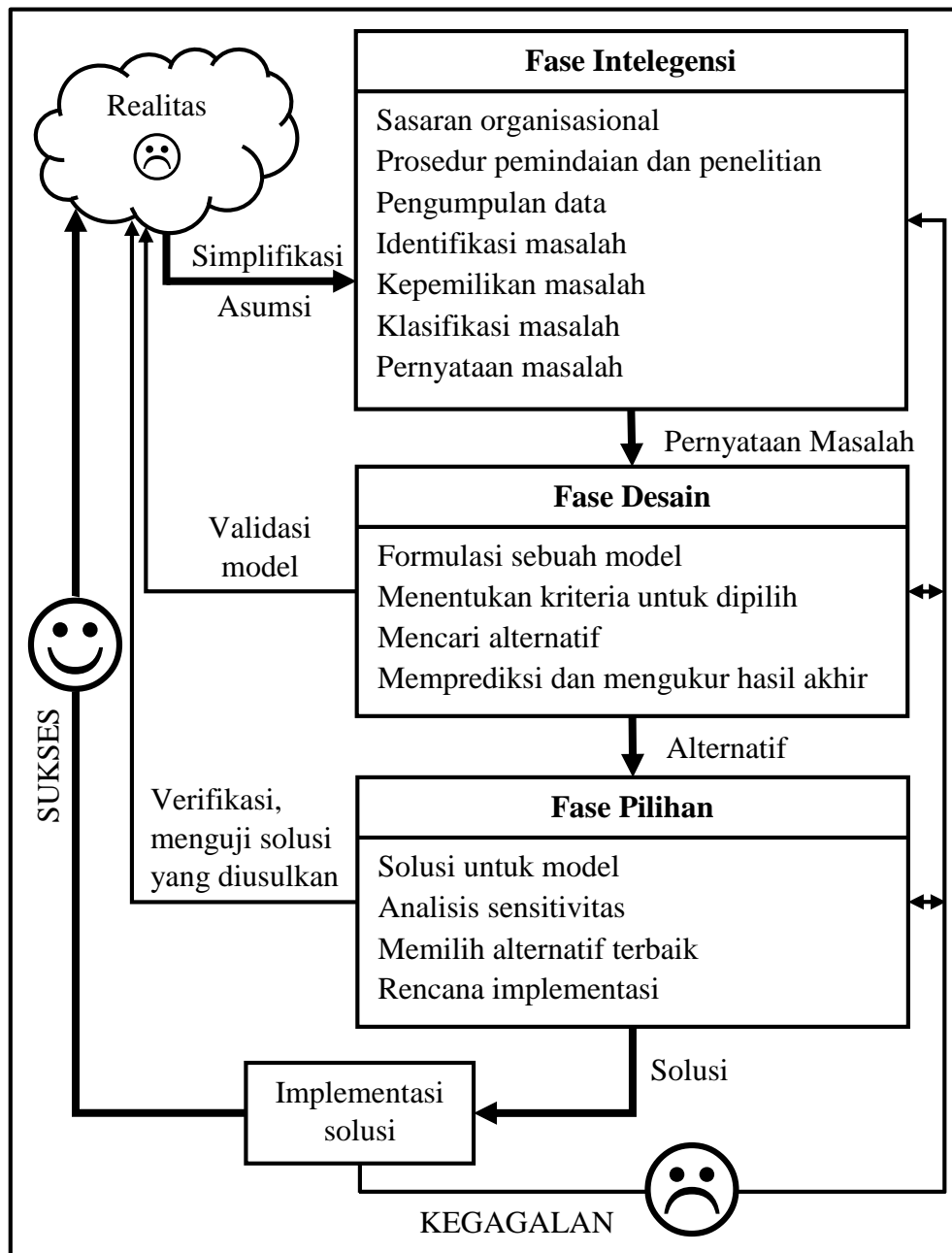
- 3) Dukungan untuk individu dan kelompok dimana dalam masalah terstruktur sering memerlukan keterlibatan individu dari suatu departemen dan tingkat organisasional yang berbeda atau bahkan dari organisasi yang lain.
- 4) Sistem pendukung keputusan menyediakan dukungan untuk berbagai keputusan yang bersifat independen atau saling terkait antar satu sama lain (sekuensial) dimana keputusan tersebut dapat dibuat satu kali, beberapa kali, dan bahkan berulang kali (dalam interval yang sama).
- 5) Mampu memberikan dukungan terhadap beberapa fase proses pengambilan keputusan, seperti intelegensi, desain, pilihan, dan implementasi.
- 6) Mendukung berbagai proses dan gaya dalam pengambilan keputusan.
- 7) Sistem pendukung keputusan mampu beradaptasi dengan waktu / perkembangan jaman dimana pembuat keputusan harus lebih reaktif dalam menghadapi perubahan-perubahan kondisi secara cepat dan tepat. Selain itu, sistem pendukung keputusan harus bersifat fleksibel sehingga pengguna dapat menambah, menghapus, mengombinasikan, merubah, maupun mengatur unsur-unsur dasar yang terdapat pada sistem pendukung keputusan.
- 8) Mudah digunakan atau bersifat *user friendly*, kemampuan grafis yang kuat, serta antarmuka sistem yang interaktif sehingga dapat meningkatkan keefektifan dari penggunaan DSS.
- 9) Peningkatan terhadap keefektifan pengambilan keputusan, seperti akurasi, *timeliness*, serta kualitas lebih diutamakan daripada tingkat efisiensinya, seperti biaya pengambilan keputusan walaupun dalam penerapannya, DSS membutuhkan waktu yang lebih lama, tetapi hasil akhir dari keputusan yang dibuat sangat baik.
- 10) Pengguna memiliki kendali penuh terhadap semua langkah dan proses dalam sistem pendukung keputusan untuk memecahkan suatu masalah. Hal tersebut berkaitan dengan tujuan dari sistem pendukung keputusan yang mana ditekankan untuk mendukung pengguna dalam mengambil suatu keputusan dan bukan untuk menggantikan posisi dari pengguna dalam mengambil suatu keputusan.

- 11) Pengguna akhir dapat mengembangkan dan memodifikasi sendiri sistem yang sederhana.
- 12) Model-model digunakan dalam sistem pendukung keputusan dengan tujuan untuk menganalisis situasi atau keadaan pengambilan keputusan. Jadi, dengan adanya kemampuan pemodelan maka pengguna dapat melakukan uji coba terhadap sistem pendukung keputusan dengan strategi yang berbeda-beda di bawah konfigurasi yang berbeda-beda pula.
- 13) Disediakan dukungan akses terhadap berbagai macam sumber data, format, dan tipe yang dimulai dari sistem informasi geografis hingga sistem berorientasi objek.
- 14) Sistem pendukung keputusan dapat digunakan sebagai alat standalone baik pada satu lokasi, didistribusikan dalam organisasi secara menyeluruh, maupun dalam beberapa organisasi yang memiliki interval yang sama. Selain itu, sistem pendukung keputusan dapat diintegrasikan dengan sistem pendukung keputusan lain dan atau aplikasi yang lain, serta dapat terhubung baik secara internal maupun eksternal dengan menggunakan jaringan dan teknologi *web*.

Karakteristik dan kemampuan kunci dari sistem pendukung keputusan tersebut berguna bagi para pengambil keputusan untuk membuat suatu keputusan yang lebih baik dan lebih konsisten pada suatu cara pembuatan keputusan sesuai dengan kriteria dan prosedur yang sudah pasti/jelas.

2.3.5. Tahapan Pengambilan Keputusan

Berdasarkan pemodelan dari pengambilan keputusan yang telah dipaparkan sebelumnya bahwa pengambilan keputusan memiliki beberapa proses atau alur dalam mengambil suatu keputusan. Menurut Simon (dalam Turban, 2005: 64), proses pengambilan keputusan tersebut meliputi 4 (empat) fase utama dalam memecahkan suatu masalah seperti yang ditunjukkan pada gambar 2.3 berikut ini.



Gambar 2.3 Pengambilan Keputusan/Proses Pemodelan
(Turban, 2005: 65)

1) Fase Intelegensi

Pada fase ini terjadi serangkaian tahapan yang dilakukan terhadap realitas atau masalah yang ada. Tahapan-tahapan tersebut meliputi penentuan sasaran yakni sasaran terhadap realitas pada sistem pendukung keputusan dan dilakukan pencarian prosedur hingga pengumpulan data terhadap masalah yang dihadapi.

Setelah data terkumpul akan diuji data tersebut melalui identifikasi masalah dan ditentukan kepemilikan masalah tersebut hingga pada akhirnya masalah akan diklasifikasi serta pada tahap terakhir diperoleh keluaran berupa pernyataan masalah. Jadi, dengan kata lain pada fase ini terjadi pencarian terhadap keputusan-keputusan yang akan dibuat terhadap masalah yang sedang dihadapi. Kemudian, dilakukan identifikasi dan penentuan masalah serta menggambarkan masalah tersebut menjadi suatu bentuk yang mudah dipahami, yakni dalam bentuk pernyataan masalah.

2) Fase Desain

Pada fase ini terjadi pembuatan, pengembangan, serta menganalisis semua masalah yang mungkin dapat diselesaikan melalui formulasi/pembuatan model. Formulasi model atau pemodelan ini dilakukan dapat memrepresentasikan/mewakili realitas yang ada dengan cara membuat asumsi-asumsi yang dapat menyederhanakan realitas sehingga mudah untuk memahami kondisi masalah yang sedang dihadapi. Selanjutnya model yang telah dibuat akan divalidasi dan ditentukan kriteria-kriteria pemilihan untuk menentukan alternatif model yang dapat menyelesaikan masalah dan pada tahap akhir diperoleh alternatif solusi untuk memilih model keputusan yang lebih tepat dan dapat memrepresentasikan realitas atau kenyataan yang sesungguhnya.

3) Fase Pilihan

Pada fase pilihan, masalah yang ada sudah tidak digunakan/diperlukan lagi seperti pada fase intelegensi dan fase desain dan pada fase ini terjadi proses pencarian, evaluasi, dan rekomendasi atau pemilihan terhadap alternatif solusi untuk model yang telah dirancang pada fase desain. Selanjutnya, solusi untuk model yang telah dipilih akan dianalisis serta diuji guna menentukan viabilitasnya dan pada tahap akhir diperoleh alternatif terbaik dari alternatif-alternatif solusi untuk memilih model keputusan serta rencana untuk implementasinya.

4) Fase Implementasi

Pada fase implementasi terjadi proses dimana membuat suatu solusi yang telah direkomendasikan/diusulkan dari fase pilihan dapat bekerja dan pada tahap ini

tidak memerlukan implementasi dari sebuah sistem komputer. Selanjutnya, implementasi yang sukses diterapkan akan dapat menyelesaikan atau memecahkan suatu masalah secara realitas/nyata. Sedangkan untuk implementasi yang gagal akan mengarahkan para pengambil keputusan melalui umpan balik (*feedback*) yang ditandai dengan *loop* untuk kembali ke tahap awal dari proses fase pengambilan keputusan.

Tahapan atau fase dari sistem pendukung keputusan tersebut dirancang dengan tujuan untuk menghasilkan keputusan yang lebih baik dan lebih efisien. Selain itu, tahapan pengambilan keputusan juga merupakan bagian yang sangat penting dari sistem pendukung keputusan sebab dalam proses tersebut digambarkan fase-fase dari penentuan masalah hingga sampai kepada solusi untuk membantu para pengambil keputusan (*user*) dalam mengambil suatu keputusan secara tepat serta akurat.

2.3.6. Manfaat Sistem Pendukung Keputusan

Menurut Julianti (2011: 20-21), ada beberapa manfaat atau keuntungan yang dapat diperoleh dari penggunaan sistem pendukung keputusan bagi pengguna/para pengambil keputusan. Beberapa manfaat atau keuntungan yang dimaksud ialah sebagai berikut:

- 1) Memiliki kemampuan untuk mendukung keputusan dalam memecahkan suatu masalah yang sangat kompleks.
- 2) Bertindak secara cepat karena sistem pendukung keputusan melakukan analisis kuantitatif terhadap situasi yang tidak diharapkan seperti kondisi yang selalu berubah-ubah dengan sangat cepat sehingga dapat mengurangi pemakaian waktu yang berlebihan (efisiensi waktu).
- 3) Memiliki kemampuan untuk mencoba berbagai strategi dengan cepat dan tepat.
- 4) Dapat mempelajari dan mengembangkan program baru dengan menggunakan pola analisis "*what if*" (apabila), merupakan salah satu sarana dalam pelatihan manajer/pengguna.

- 5) Dapat membangun hubungan komunikasi, sehingga pengumpulan data dan pemecahan masalah merupakan alat yang dapat digunakan untuk meningkatkan kerjasama kelompok/tim.
- 6) Meningkatkan pengendalian pengukuran dan meningkatkan kinerja organisasi.
- 7) Dapat terjadi penghematan biaya karena menghindari kesalahan dalam mengambil suatu keputusan.
- 8) Keputusan lebih objektif dan konsisten dibandingkan dengan mengambil keputusan hanya dengan menggunakan intuisi saja.
- 9) Meningkatkan efektivitas manajerial dengan mengefisiensikan waktu dalam analisis, perencanaan, dan pelaksanaan.
- 10) Meningkatkan produktivitas dari analisis.
- 11) Sistem pendukung keputusan mampu menyajikan berbagai alternatif keputusan.
- 12) Sistem pendukung keputusan dapat menyediakan bukti tambahan untuk memberikan pembenaran sehingga memperkuat posisi dalam pengambilan suatu keputusan.

2.4. *Fuzzy Multi-Attribute Decision Making*

Pada hakikatnya, dasar teori dari *Fuzzy Multi-Attribute Decision Making* diawali dengan konsep mengenai *Multi-Attribute Decision Making* (MADM). *Multi-Attribute Decision Making* (MADM) merupakan salah satu model dari *Multiple Criteria Decision Making* (MCDM) yang juga merupakan suatu metode pengambilan keputusan untuk menetapkan alternatif terbaik dari sejumlah alternatif berdasarkan beberapa kriteria tertentu (Kusumadewi, 2006: 69). Selain itu, tujuan dari adanya *Multi-Attribute Decision Making* (MADM) ialah untuk dapat menyelesaikan masalah-masalah dalam ruang diskrit. Oleh sebab itu pula, pada *Multi-Attribute Decision Making* (MADM) biasanya digunakan untuk melakukan penilaian atau seleksi terhadap beberapa alternatif dalam jumlah yang terbatas. Jadi, secara umum dapat dikatakan pula bahwa *Multi-Attribute Decision Making* (MADM) dapat menyeleksi alternatif terbaik dari sejumlah alternatif yang ada.

Di samping itu, menurut Rudolphi (dalam Kusumadewi, 2006: 72-73) yang menyatakan bahwa proses *Multi-Attribute Decision Making* (MADM) dapat dilakukan melalui 3 (tiga) tahap, yakni penyusunan komponen-komponen situasi, analisis, serta sintesis informasi. Pada tahap penyusunan komponen-komponen situasi, akan dibentuk tabel taksiran yang berisi identifikasi alternatif dan spesifikasi tujuan, kriteria, dan atribut. Untuk menspesifikasikan tujuan situasi yang dinotasikan $|O_i, i = 1, \dots, t|$, dapat dilakukan dengan cara menghimpun kumpulan/daftar konsekuensi-konsekuensi yang mungkin dari alternatif yang telah teridentifikasi yang dapat dinotasikan dengan $|A_i, i = 1, \dots, n|$. Selain itu juga disusun atribut-atribut yang akan digunakan pada tahap penyusunan komponen-komponen situasi ini dimana dinotasikan sebagai $|a_k, k = 1, \dots, m|$.

Pada tahap kedua, yakni analisis yang mana dapat dilakukan melalui 2 (dua) langkah. Pertama, membuat taksiran atau perkiraan terhadap besaran potensial, peluang, dan ketidakpastian yang berhubungan dengan dampak-dampak yang mungkin terjadi pada setiap alternatif yang ada. Kedua, meliputi pemilihan dari preferensi pengambilan keputusan untuk setiap nilai guna menghindari resiko yang mungkin akan terjadi. Selain itu, pada langkah pertama tersebut dimana beberapa metode menggunakan fungsi distribusi dengan notasi $|p_j(x)|$ yang menyatakan probabilitas dari kumpulan atribut $|a_k|$ terhadap setiap alternatif $|A_i|$. Konsekuensi atau dampak yang mungkin terjadi pada setiap alternatif dapat pula ditentukan secara langsung dari agregasi sederhana yang dilakukan pada informasi terbaik yang tersedia. Demikian pula, pada langkah kedua dimana ada beberapa cara yang dapat dilakukan untuk menentukan preferensi pengambilan keputusan pada setiap konsekuensi/resiko yang mungkin akan terjadi. Adapun cara atau metode yang dimaksud ialah dengan menurunkan bobot atribut dan untuk kriteria ialah dengan cara fungsi utilitas atau penjumlahan terbobot.

Multi-Attribute Decision Making (MADM) merupakan salah satu model dari *Multiple Criteria Decision Making* (MCDM) yang secara umum dapat pula didefinisikan sebagai berikut (Zimmermann dalam Kusumadewi, 2006: 72-73):

Misalkan $A = \{a_i | i = 1, \dots, n\}$ merupakan himpunan alternatif-alternatif keputusan dan $C = \{c_j | j = 1, \dots, m\}$ merupakan himpunan tujuan yang diharapkan, maka akan ditentukan alternatif x^0 yang memiliki derajat harapan tertinggi terhadap tujuan-tujuan yang relevan c_j . Di sisi lain, sebagian besar pendekatan *Multi-Attribute Decision Making* (MADM) dilakukan melalui 2 (dua) langkah, yakni: pertama, melakukan agregasi terhadap keputusan-keputusan yang memiliki responsi terhadap semua tujuan pada setiap alternatif. Sedangkan yang kedua, yakni melakukan perankingan terhadap alternatif-alternatif keputusan tersebut berdasarkan hasil agregasi keputusan.

Berdasarkan hal tersebut maka dapat dikatakan pula bahwa masalah pada *Multi-Attribute Decision Making* (MADM) dapat diselesaikan dengan cara mengevaluasi m pada setiap alternatif $A_i (i = 1, 2, \dots, m)$ terhadap sekumpulan atribut atau kriteria $C_j (j = 1, 2, \dots, n)$ dimana setiap atribut tidak saling bergantung antara satu sama lainnya (Kusumadewi, 2006: 73). Adapun matriks keputusan setiap alternatif terhadap setiap atribut tersebut dapat dinotasikan sebagai berikut:

$$X = \begin{bmatrix} x_{11} & x_{12} & \dots & x_{1n} \\ x_{21} & x_{22} & \dots & x_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ x_{m1} & x_{m2} & \dots & x_{mn} \end{bmatrix}$$

dimana x_{ij} merupakan rating kinerja alternatif ke- i terhadap atribut ke- j . Nilai bobot yang dapat menunjukkan tingkat kepentingan relatif pada setiap atribut, dapat dinotasikan sebagai berikut:

$$W = \{w_1, w_2, \dots, w_n\}$$

Rating kerja (X) dan nilai bobot (W) merupakan nilai utama yang dapat merepresentasikan preferensi absolut dari pengambil keputusan. Dengan demikian, masalah *Multi-Attribute Decision Making* (MADM) diakhiri dengan proses perankingan untuk mendapatkan alternatif terbaik yang diperoleh berdasarkan nilai keseluruhan dari preferensi yang diberikan (Yeh dalam Kusumadewi, 2006: 73).

Adapun beberapa metode yang dapat digunakan untuk menyelesaikan masalah *Multi-Attribute Decision Making* (MADM), yakni:

- 1) *Simple Additive Weigthing Method (SAW)*,
- 2) *Weigthed Product (WP)*,
- 3) ELECTRE,
- 4) *Technique for Order Preference by Similarity to Ideal Solution (TOPSIS)*,
- 5) *Analytic Hierarchy Process (AHP)*.

Namun, menurut Kusumadewi (2006: 136) yang berpendapat bahwa pada metode-metode *Multi-Attribute Decision Making (MADM)* klasik tersebut memiliki beberapa kelemahan. Kelemahan-kelemahan yang dimaksud, antara lain:

- 1) Tidak cukup efisien untuk dapat menyelesaikan masalah-masalah pengambilan keputusan yang melibatkan data-data yang tidak tepat, tidak pasti, dan tidak jelas (Zhang dalam Kusumadewi, 2006: 136).
- 2) Biasanya diasumsikan bahwa keputusan akhir terhadap alternatif-alternatifnya dinyatakan dengan bilangan riil sehingga membuat tahap perankingan tersebut kurang merepresentasikan beberapa permasalahan tertentu serta penyelesaian masalah hanya terpusat pada tahap agregasinya saja (Zimmermann dalam Kusumadewi, 2006: 137).

Salah satu cara yang dapat digunakan untuk menyelesaikan permasalahan tersebut ialah dengan menggunakan *Fuzzy Multi-Attribute Decision Making (FMDAM)* (Zhang dalam Kusumadewi, 2006: 137).

Fuzzy Multi-Attribute Decsion Making (FMADM) merupakan salah satu model dari *Fuzzy Multiple Criteria Decision Making (FMCDM)* dan juga suatu metode yang digunakan untuk menentukan serta mencari alternatif optimal dari sejumlah alternatif yang ada dengan kriteria tertentu (Kusumadewi, 2006: 136). Pada dasarnya, inti dari *Fuzzy Multi-Attribute Decision Making (FMDAM)* ialah menentukan nilai bobot untuk setiap atribut (kriteria) yang kemudian dilanjutkan dengan proses perankingan yang akan menyeleksi alternatif yang sudah diberikan sebelumnya. Selain itu, *Fuzzy Multi-Attribute Decision Making (FMDAM)* juga memiliki suatu tujuan tertentu yang mana diklasifikasi dalam 2 (dua) tipe, yaitu (Simoes-Marques dalam Kusumadewi, 2006: 136) menyeleksi alternatif berdasarkan atribut (kriteria) optimal atau memiliki ciri-ciri terbaik dan mengklasifikasikan alternatif berdasarkan peran tertentu.

Di samping itu, terdapat pula 2 (dua) tahap yang dilakukan untuk dapat menyelesaikan masalah *Fuzzy Multi-Attribute Decision Making* (FMADM) (Kusumadewi, 2006: 136), yaitu sebagai berikut:

- 1) Membuat *rating* pada setiap alternatif berdasarkan agregasi derajat ketepatan/kecocokan pada semua kriteria.
- 2) Melakukan perankingan terhadap semua alternatif dengan tujuan untuk mendapatkan alternatif terbaik. Adapun 2 (dua) cara yang dapat digunakan dalam proses perankingan tersebut, yakni melalui *defuzzy* atau melalui preferensi *fuzzy*. Pada metode *defuzzy*, hal pertama yang dilakukan ialah mengkonversi data pada bilangan *fuzzy* menjadi bentuk bilangan *crisp* sebab proses perankingan tersebut didasarkan pada data yang mana berbentuk bilangan *crisp*. Namun, pada metode *defuzzy* tersebut terdapat kemungkinan untuk kehilangan beberapa informasi terutama yang menyangkut dengan masalah ketidakpastian. Oleh karena itu, penggunaan metode preferensi *fuzzy* lebih tepat untuk digunakan dalam mengatasi masalah yang berkaitan dengan ketidakpastian dimana proses dalam metode ini dimulai dari penentuan bilangan *fuzzy* hingga perankingan (Lee dalam Kusumadewi, 2006: 136).

Selain itu, jika ditinjau dari segi tipe data yang digunakan pada setiap alternatif-alternatif dalam pengambilan keputusan, *Fuzzy Multi-Attribute Decision Making* (FMADM) dapat dibagi menjadi 3 (tiga) kelompok data, yaitu: 1) semua data yang digunakan adalah data *fuzzy*, 2) semua data yang digunakan adalah data *crisp*, atau yang terakhir 3) data yang digunakan merupakan campuran antara data *fuzzy* dan data *crisp*.

Menurut Kusumadewi (2006: 145) yang mengungkapkan bahwa terdapat salah satu mekanisme yang dapat digunakan untuk mengatasi atau menyelesaikan masalah *Fuzzy Multi-Attribute Decision Making* (FMADM), yakni dengan mengaplikasikan metode *Multi-Attribute Decision Making* (MADM) klasik seperti *Simple Additive Weighting* (SAW), *Weighted Product* (WP), *ELECTRE*, *Technique for Order Preference by Similarity to Ideal Solution* (TOPSIS), ataupun *Analytic Hierarchy Process* (AHP) dimana metode-metode tersebut digunakan untuk melakukan proses perankingan. Namun, sebelum melakukan proses perankingan

yang mana untuk terlebih dahulu dilakukan konversi data *fuzzy* ke dalam bentuk data *crisp* (Chen dalam Kusumadewi, 2006: 145). Apabila data *fuzzy* tersebut tersedia dalam bentuk linguistik, maka data harus dikonversi terlebih dahulu dalam bentuk bilangan *fuzzy* yang selanjutnya akan dikonversi lagi ke dalam bilangan *crisp*.

2.5. Simple Additive Weighting

Menurut Fishburn dan MacCrimmon (dalam Kusumadewi, 2006: 74), metode *simple additive weighting* (SAW) sering pula dikenal dengan istilah metode penjumlahan terbobot. Konsep dasar dari metode *simple additive weighting* (SAW) ini ialah mencari penjumlahan terbobot dari rating kinerja pada setiap alternatif yang ada pada semua atribut.

Pada metode *simple additive weighting* (SAW) diperlukan langkah/proses perhitungan normalisasi matriks keputusan (X) ke suatu skala yang dapat dibandingkan dengan semua *rating* alternatif yang ada (Kusumadewi, 2006: 74; Putra dan Dinna, 2011: 17-18; Firdausa dkk., 2016: 2). Adapun rumus untuk rating kinerja ternormalisasi dapat diberikan sebagai berikut.

$$r_{ij} = \begin{cases} \frac{x_{ij}}{\text{Max}_i x_{ij}}; & \text{jika } j \text{ atribut keuntungan (benefit)} \\ \frac{\text{Min}_i x_{ij}}{x_{ij}}; & \text{jika } j \text{ atribut biaya (cost)} \end{cases}$$

dimana:

r_{ij} = nilai *rating* kinerja ternormalisasi.

x_{ij} = nilai atribut yang dimiliki dari setiap kriteria.

$\text{Max}_i x_{ij}$ = nilai maksimum atau nilai terbesar dari setiap kriteria.

$\text{Min}_i x_{ij}$ = nilai minimum atau nilai terkecil dari setiap kriteria.

dengan ketentuan normalisasi, ialah sebagai berikut:

- 1) *Rating* kinerja ternormalisasi dapat dikatakan memiliki atribut keuntungan (*benefit*) apabila dalam nilai atribut tersebut dapat memberikan keuntungan bagi

pengambil keputusan. Sedangkan untuk atribut biaya (*cost*) apabila pada atribut tersebut dapat memberikan pengeluaran (biaya) bagi pengambil keputusan jika nilainya semakin besar.

- 2) Apabila berupa atribut keuntungan maka nilai (x_{ij}) dari setiap kolom atribut harus dibagi dengan nilai $(\text{Max}_i x_{ij})$ dari setiap kolom. Sedangkan untuk atribut biaya, nilai $(\text{Min}_i x_{ij})$ dari setiap kolom atribut harus dibagi dengan nilai (x_{ij}) dari setiap kolom.

Selanjutnya, nilai preferensi untuk setiap alternatif (V_i) dapat pula diberikan sebagai berikut.

$$V_i = \sum_{j=1}^n w_j r_{ij}$$

dimana:

V_i = nilai preferensi untuk setiap alternatif.

w_j = nilai bobot dari setiap kriteria.

r_{ij} = nilai *rating* kinerja ternormalisasi.

Dengan demikian, pada nilai preferensi ini dimana jika terdapat nilai V_i yang jauh lebih besar atau tinggi dari nilai lainnya maka dapat mengindikasikan bahwa alternatif A_i merupakan alternatif yang lebih terpilih dari alternatif lainnya.

Sementara itu, menurut Rusnadi (2014: 7) yang mengatakan bahwa adapula beberapa kelebihan yang dimiliki oleh metode *simple additive weighting* (SAW), yakni: 1) menentukan nilai bobot untuk setiap atribut yang kemudian dilanjutkan dengan proses perankingan dimana dalam proses tersebut akan menyeleksi alternatif terbaik dari sejumlah alternatif yang ada, 2) penilaian akan lebih efektif dan lebih tepat karena didasarkan pada nilai kriteria dan bobot preferensi yang sudah ditentukan.

2.6. Microsoft Visual Basic 2010

Sejarah singkat bahasa pemrograman *Visual Basic* diawali dengan adanya sebuah bahasa pemrograman yang diberi nama *BASIC (Beginner's All-Purpose Symbolic Instruction Code)* (Hidayatullah, 2012: 3). Bahasa pemrograman *BASIC* pertama kali mulai dikembangkan pada tahun 1964 oleh John G. Kemeney dan Thomas E. Kurtz dari Perguruan Tinggi Dartmouth, New Hampshire, Amerika Serikat. Sesuai dengan nama dari bahasa pemrograman tersebut, *BASIC* ditunjukkan sebagai bahasa yang paling sederhana atau bahasa dasar bagi mereka yang tidak terlalu familiar dengan dunia pemrograman. Akan tetapi, tidak semua pengguna *BASIC* merupakan pemrogram pemula karena bahasa pemrograman *BASIC* dapat pula digunakan oleh para ahli pemrograman lainnya.

Pada tahun 1991 *Microsoft* mengeluarkan bahasa pemrograman *Visual Basic* yang merupakan pengembangan dari bahasa pemrograman *BASIC* terutama pada sisi antarmuka dengan versi awal pengembangan tersebut ialah *1.0* atau dapat pula disebut *Visual Basic 1.0 (VB 1.0)* hingga sampai versi *6.0 (VB 6.0)* bahasa pemrograman *Visual Basic* menjadi salah satu bahasa pemrograman terpopuler di dunia. Lalu pada akhir tahun 1999 perkembangan teknologi *.Net* diluncurkan dimana *Microsoft* memosisikan teknologi *.Net* tersebut sebagai *platform* yang dapat merancang dan membangun *XML Web Services*. *XML Web Services* tersebut memungkinkan program aplikasi dengan tipe manapun dapat berjalan pada sistem komputer dengan tipe manapun pula serta dapat pula mengambil data dengan tipe apapun yang tersimpan dalam suatu *server* melalui sebuah jaringan internet. Sehingga pada tahun 2002 perkembangan *Visual Basic* mengalami penyempurnaan dan menjadi awal perkembangan dari *Visual Basic .Net* dengan versi awal yang juga sekaligus generasi terbaru dari *Visual Basic*, yakni *Visual Basic .Net 2002 (VB. Net 2002)* atau dapat disebut pula dengan *Visual Basic 7.0 (VB 7.0)* hingga versi terbaru yang dirilis oleh *Microsoft* adalah *Visual Basic 15.0 (Visual Basic 2015)*. Akan tetapi, dalam hal ini peneliti menggunakan bahasa pemrograman *Visual Basic 2010* sebagai salah satu bahasa pemrograman yang digunakan untuk merancang dan

membuat program aplikasi berbasis *desktop* dimana program aplikasi tersebut merupakan implementasi sistem pendukung keputusan menggunakan logika *fuzzy*.

2.6.1. Bahasa Pemrograman *Microsoft Visual Basic*

Bahasa pemrograman *Visual Basic* dibangun berdasarkan dua pendekatan, yang pertama ialah pendekatan *Visual* dan yang kedua ialah pendekatan *Basic* (Tim Penelitian dan Pengembangan Wahana Komputer, 2001: 1-2). Kata “*Visual*” dalam *Visual Basic* menunjukkan cara bahwa bahasa pemrograman tersebut dapat digunakan untuk merancang dan membuat program aplikasi *Windows* berbasis *Graphical User Interface* (GUI). Dengan cara tersebut maka pemrogram tidak perlu lagi menuliskan instruksi pemrograman dalam bentuk kode baris, tetapi pemrogram dapat dengan mudah melakukan “*drag and drop*” terhadap objek-objek yang akan digunakan menggunakan beberapa *tool* yang terdapat dalam bahasa pemrograman *Visual Basic*. Sedangkan kata “*Basic*” merupakan bagian dari bahasa pemrograman *BASIC* yang mana dalam perkembangannya sudah banyak digunakan oleh para pemrogram untuk merancang suatu program aplikasi.

Sejalan dengan pendapat yang dipaparkan oleh Tim Penelitian dan Pengembangan Wahana Komputer, Sudarjo (dalam Elistri, dkk., 2014: 106) mengatakan bahwa *Visual Basic* merupakan salah satu *development tools* yang digunakan untuk dapat membangun program aplikasi dalam lingkungan *Windows* (aplikasi berbasis *Windows*). Dalam membangun suatu program aplikasi, *Visual Basic* menggunakan pendekatan *visual* untuk merancang antar muka pengguna atau *user interface* dalam bentuk *form*, sedangkan untuk koding menggunakan dialek atau gaya bahasa dari bahasa pemrograman *BASIC* yang cenderung mudah dipelajari. Lebih lanjut Sudarjo menjelaskan bahwa pada pemrograman *visual*, pengembangan aplikasi dimulai dengan pembentukan *user interface*, kemudian mengatur properti dari objek-objek yang digunakan dalam *user interface*, dan baru dilakukan penulisan kode program untuk mengangani kejadian-kejadian (*event*).

2.6.2. Bahasa Pemrograman *Microsoft Visual Basic 2010*

Bahasa pemrograman *Microsoft Visual Basic 2010* merupakan salah satu komponen dari paket bahasa pemrograman *Microsoft Visual Studio 2010* (Tim Penelitian dan Pengembangan Wahana Komputer, 2010: iii). Paket yang diluncurkan oleh *Microsoft* pada 12 April 2010 dengan nama kode *Dev10* dan telah menggunakan fitur *.Net Framework 4.0* tersebut memuat beberapa jenis bahasa pemrograman seperti *Visual Basic*, *Visual C++*, *Visual Web Developer*, *Visual C#*, serta *Visual F#* (Shelly dan Hoisington, 2010: 19). *Microsoft Visual Studio 2010* hadir dalam beberapa paket edisi untuk pengguna bahasa pemrograman tersebut, yakni *Professional*, *Premium*, *Ultimate*, serta *Express* dan dalam hal ini peneliti menggunakan *Microsoft Visual Studio 2010* edisi *Express*. *Microsoft Visual Studio 2010 Express* merupakan kumpulan *freeware* dari *integrated development environment* (IDE) yang dikembangkan oleh *Microsoft* dan juga merupakan versi ringan dari *Microsoft Visual Studio*. Gagasan dikembangkan edisi *Express* ini ialah untuk menyediakan efisiensi, mudah digunakan, dan mudah dipelajari bagi pengguna *integrated development environment* (IDE) selain pengembang program, seperti penggemar maupun mahasiswa. Karena *Microsoft Visual Studio 2010* hadir dalam edisi *Express* maka produk yang dimuatpun juga hadir dalam edisi *Express*, seperti *Visual Basic 2010 Express*, *Visual Web Developer 2010 Express*, *Visual C++ 2010 Express*, *Visual C# 2010 Express*, serta *SQL Server 2010 Express* (Sheldon dkk., 2010: 4-7).

Sementara itu, *Integrated Development Environment* (IDE) pada *Visual Studio 2010* telah didesain ulang dengan tampilan visual yang lebih menarik dan interaktif serta lebih mudah digunakan (*user friendly*) baik oleh para pengembang aplikasi maupun pengguna aplikasi. Sedangkan untuk *code editor* pada *Visual Basic 2010* telah menambah fitur *highlights reference* yang berguna ketika satu simbol/kode dalam bahasa pemrograman tersebut dipilih/diseleksi maka simbol/kode yang sama meskipun pada keadaan ataupun penggunaan yang berbeda tetapi akan sama-sama terlihat berwarna dengan warna yang sama pula. Misalkan apabila kode *math.* dipilih/diseleksi maka seluruh kode dengan nama “*math.*” akan

terlihat berwarna dengan warna yang sama (Tim Penelitian dan Pengembangan Wahana Komputer, 2010: iii). Selain itu, terdapat pula fitur tambahan lain yang menyatakan pengembangan terhadap bahasa pemrograman *Visual Basic .Net*, yakni dukungan terhadap pengembangan program aplikasi menggunakan *Microsoft SilverLight*, dukungan terhadap program aplikasi berbasis *cloud computing*, serta perluasan dukungan terhadap basis data, baik *standalone* maupun *database server* (Shelly dan Hoisington, 2010: 21).

2.6.3. *Integrated Development Environment (IDE) Microsoft Visual Basic 2010*

Pada bahasa pemrograman *Microsoft Visual Basic 2010* terdapat sebuah lingkungan kerja atau biasa disebut sebagai *integrated development environment* (IDE) yang hadir dengan tampilan atau antar muka (*user interface*) lebih menarik dan interaktif (Tim Penelitian dan Pengembangan Wahana Komputer, 2010: iii). Dengan tampilan antar muka (*user interface*) yang hadir lebih menarik dan interaktif tersebut maka dapat memudahkan *user* dalam memahami maupun menggunakan *integrated development environment* (IDE) itu sendiri dalam bahasa pemrograman *Microsoft Visual Basic 2010*.

Integrated development environment (IDE) menyediakan beberapa sarana yang diperlukan untuk dapat merancang, membangun, memodifikasi, menerjemahkan, maupun mengoreksi kesalahan sebuah program aplikasi, seperti *form designer*, *source code editor*, *compiler*, *debugger*, dan fasilitas pendukung lainnya (Bradley dan Millspaugh, 2010: 8). Sarana-sarana tersebut memungkinkan seorang pengembang dalam membuat suatu program aplikasi menjadi lebih mudah dan dapat mendukung proses pengembangan program aplikasi menjadi lebih cepat dan efisien.

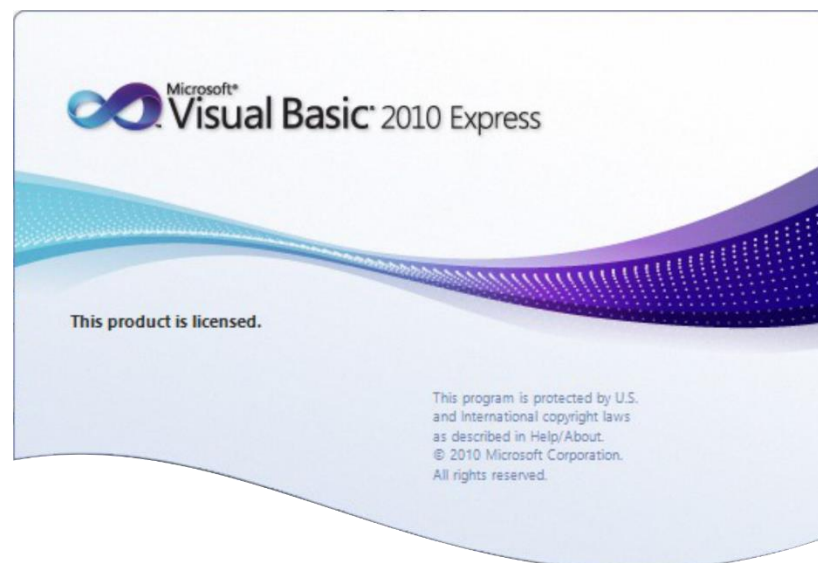
Namun, sebelum melakukan proses pengembangan suatu program aplikasi maka seorang pengembang harus mengetahui terlebih dahulu konsep dasar dari bahasa pemrograman *Visual Basic 2010*. Konsep dasar yang dimaksud berupa dasar

pengetahuan, pemahaman, dan penguasaan mengenai lingkungan kerja atau *integrated development environmet* (IDE) dari *Visual Basic 2010*.

Dengan demikian, maka seorang pengembang dapat dengan mudah menggunakan fasilitas-fasilitas yang telah disediakan oleh *integrated development environment* (IDE) untuk merancang dan membangun sebuah program aplikasi *Windows* berbasis *graphical user interface*. Adapun pemaparan secara rinci mengenai lingkungan kerja dalam bahasa pemrograman *Visual Basic 2010* beserta dengan tampilannya yang dijabarkan lebih lanjut melalui gambar di bawah ini dimana edisi yang digunakan peneliti dalam hal ini ialah versi *Express*:

1) *Splash Screen Microsoft Visual Basic 2010 Express*

Pada saat menjalankan bahasa pemrograman *Microsoft Visual Basic 2010* maka bahasa pemrograman tersebut pertama kali akan menampilkan sebuah jendela *splash Microsoft Visual Basic 2010* seperti yang ditunjukkan oleh gambar 2.4 di bawah ini.

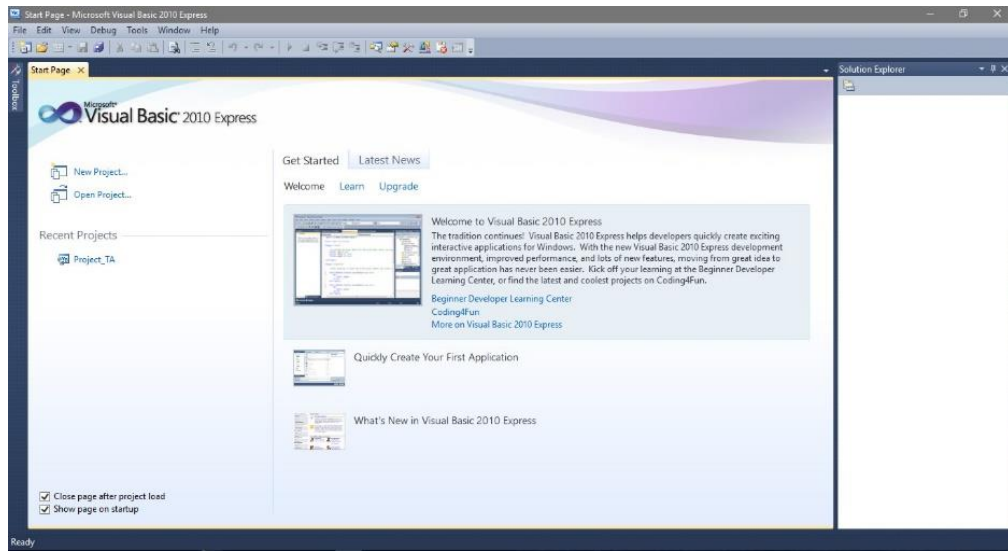


Gambar 2.4 Tampilan Jendela *Splash Visual Basic 2010 Express*

Jendela *splash* tersebut berfungsi sebagai layar pembuka dalam bahasa pemrograman *Visual Basic 2010* sebelum masuk pada lingkungan kerja yang menjadi tempat perancangan dan pembuatan sebuah program aplikasi.

2) *Start Page Microsoft Visual Basic 2010 Express*

Setelah sebelumnya jendela *splash* muncul saat pertama kali menjalankan bahasa pemrograman *Visual Basic 2010 Express* maka selanjutnya akan keluar jendela *start page Microsoft Visual Basic 2010 Express* seperti yang ditunjukkan oleh gambar 2.5 di bawah ini.



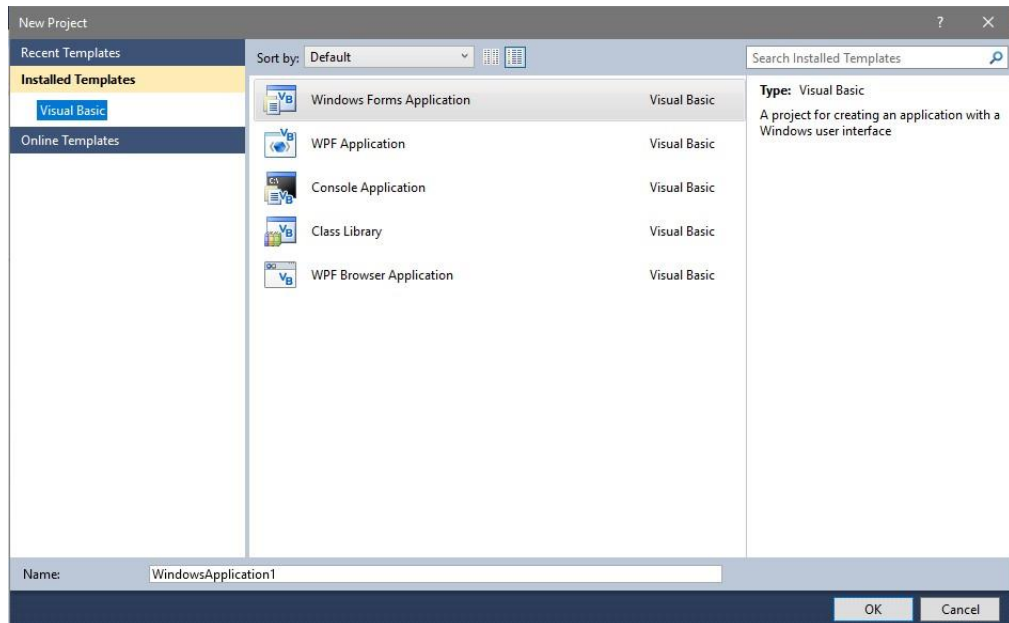
Gambar 2.5 Tampilan *Start Page Visual Basic 2010 Express*

Jendela *start page* tersebut berfungsi sebagai layar yang menampilkan informasi serta menu-menu yang berguna bagi seorang pengembang untuk mempersiapkan lingkungan kerja dari *Visual Basic 2010 Express* sebelum memulai pembuatan sebuah program aplikasi maupun membuka lembar kerja yang sebelumnya sudah pernah dibuat.

3) *Integrated Development Environment Microsoft Visual Basic 2010 Express*

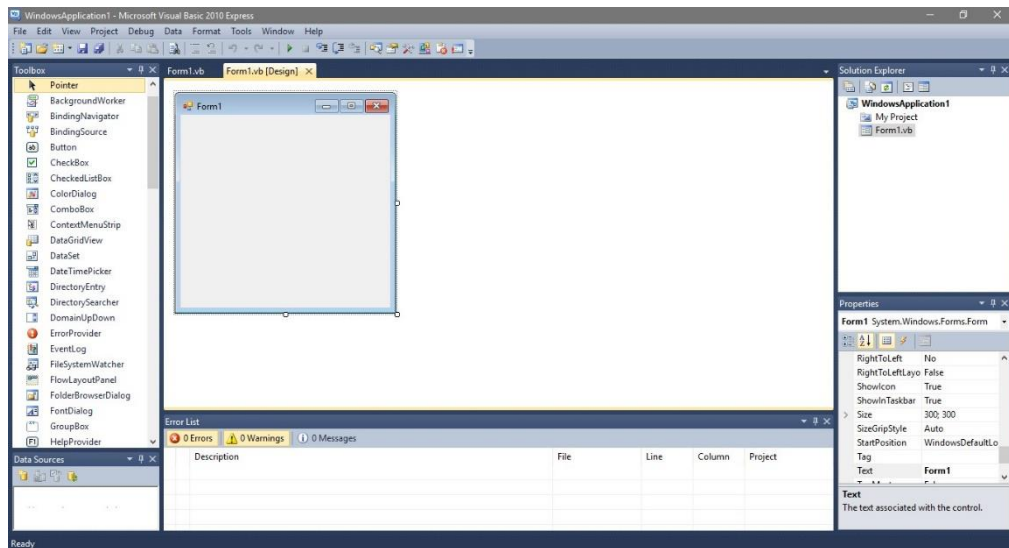
Setelah sebelumnya dipaparkan mengenai tampilan awal dari bahasa pemrograman *Visual Basic 2010 Express* maka selanjutnya akan dipaparkan pula mengenai lingkungan kerja atau biasa disebut dengan *integrated development environment* atau IDE dari *Visual Basic 2010 Express*. Namun, sebelum masuk pada *integrated development environment* dari *Visual Basic 2010 Express* maka pada jendela *start page* sebelumnya akan tersedia dua menu untuk masuk ke dalam lingkungan kerja, yakni *New Project* dan *Open Project*. Menu *new project* berguna untuk dapat membuat sebuah proyek baru sedangkan menu *open project* berguna untuk dapat membuka proyek yang

sudah ada dan dibuat sebelumnya, dalam hal ini akan dipilih menu *New Project*. Setelah memilih menu *New Project* maka akan muncul kotak dialog *New Project* yang menyediakan pilihan basis dari *integrated development environment* yang akan ditampilkan dalam perancangan dan pembuatan program aplikasi seperti yang ditunjukkan dalam gambar 2.6 di bawah ini.



Gambar 2.6 Tampilan Kotak Dialog *New Project*

Karena dalam hal ini *integrated development environment* yang ditampilkan berbasis *user interface* maka pilih pilihan *Windows Forms Application* lalu klik *OK*. Selanjutnya akan muncul jendela *integrated development environment* dari *Visual Basic 2010 Express* seperti yang ditunjukkan oleh gambar 2.7 di bawah ini.



Gambar 2.7 Tampilan *Integrated Development Environment Visual Basic 2010 Express*

Pada *integrated development environment* atau IDE dalam bahasa pemrograman *Visual Basic 2010 Express* yang ditunjukkan oleh gambar 6.4 di atas, terdapat komponen-komponen yang menjadi sarana atau fasilitas untuk merancang maupun membangun sebuah program aplikasi berbasis *Windows*. Adapun komponen-komponen yang dimaksud dapat dipaparkan lebih lanjut sebagai berikut:

1) *Title Bar*

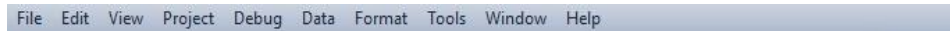
Title bar merupakan salah satu komponen dari *integrated development environment* yang berguna untuk menampilkan nama dari proyek yang sedang dijalankan seperti ditunjukkan pada gambar 2.8 di bawah ini.



Gambar 2.8 Tampilan *Title Bar Visual Basic 2010 Express*

2) *Menu Bar*

Menu bar merupakan daftar menu yang berisikan daftar mengenai perintah-perintah yang dapat digunakan saat bekerja pada lingkungan *Visual Basic*. Adapun daftar menu yang dimaksud dapat terdiri dari menu *File, Edit, View, Project, Build, Debug, Data, Tools, Window*, dan *Help* seperti yang terlihat pada gambar 2.9 di bawah ini.

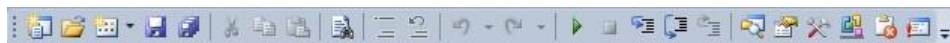


Gambar 2.9 Tampilan *Menu Bar Visual Basic 2010 Express*

Setiap menu dalam *menu bar* di atas terdapat *submenu* yang masing-masing memiliki fungsi dan kegunaan tertentu yang lebih lanjut dapat dipaparkan sebagai berikut:

3) *Toolbar*

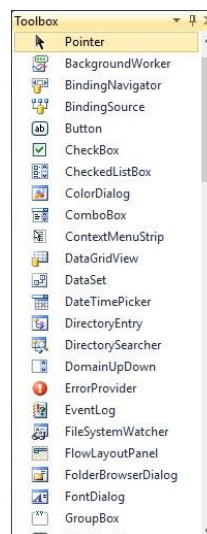
Toolbar merupakan sekumpulan tombol yang direpresentasikan menggunakan model ikonik dengan fungsi untuk dapat mengakses perintah-perintah tertentu yang terdapat dalam *Menu Bar* secara cepat dimana menu tersebut merupakan menu yang sering dipakai oleh pengguna dari *Visual Basic*. Adapun bentuk *Toolbar* dalam *Visual Basic 2010 Express* dapat ditunjukkan pada gambar 2.10 di bawah ini.



Gambar 2.10 Tampilan *Toolbar Visual Basic 2010 Express*

4) *Toolbox*

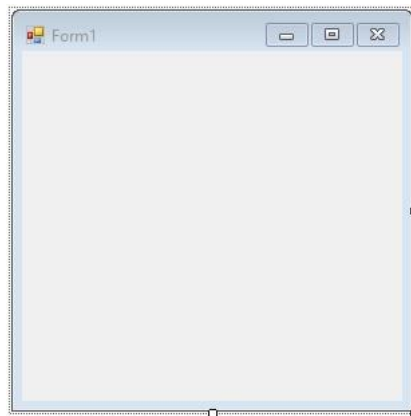
Toolbox merupakan sebuah kotak piranti yang menyediakan berbagai macam kontrol atau objek item dimana item tersebut dapat ditempatkan pada *Form Designer* guna untuk dapat mendukung perancangan maupun pembuatan sebuah program aplikasi. Bentuk dari *Toolbox* dapat ditunjukkan pada gambar 2.11 di bawah ini.



Gambar 2.11 Tampilan *Toolbox Visual Basic 2010 Express*

5) *Form Designer*

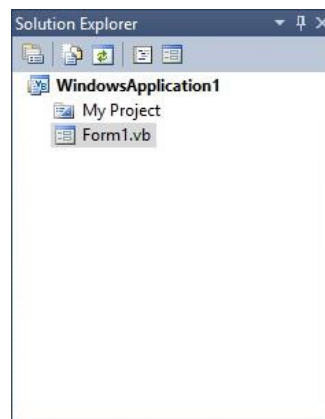
Form designer merupakan suatu media yang digunakan sebagai area kerja untuk dapat menempatkan/meletakkan kontrol-kontrol atau objek-objek item sehingga dengan dukungan dari objek atau kontrol item tersebut maka seorang pengembang dapat merancang, mendesain, ataupun membangun sebuah program aplikasi. Adapun gambaran dari *Form Designer* ini dapat diperlihatkan dalam gambar 2.12 di bawah ini.



Gambar 2.12 Tampilan *Form Designer*
Visual Basic 2010 Express

6) *Solution Explorer*

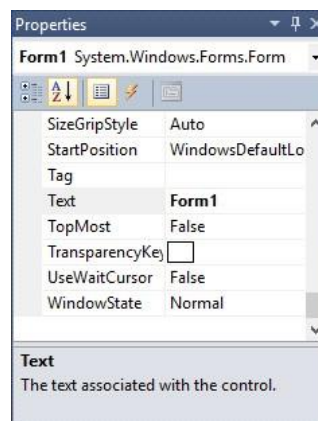
Solution explorer merupakan sebuah jendela yang dapat menampilkan secara hirarki daftar nama-nama file dari proyek yang sedang dibuat. yang ditunjukkan oleh gambar 2.13 di bawah ini.



Gambar 2.13 Tampilan *Solution Explorer*
Visual Basic 2010 Express

7) *Properties*

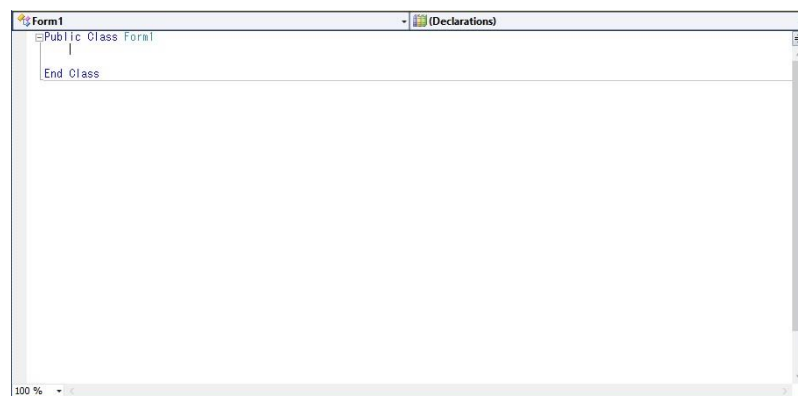
Properties merupakan sebuah jendela yang digunakan untuk dapat menampilkan daftar terstruktur berisi informasi mengenai pengaturan (*setting*) dari suatu objek yang sedang dipilih/diseleksi. Adapun informasi yang dimaksud dapat berupa *Name*, *Size*, *Colour*, *Position*, ataupun *Text* seperti yang diperlihatkan dalam gambar 2.14 di bawah ini.



Gambar 2.14 Tampilan *Properties Visual Basic 2010 Express*

8) *Code Editor*

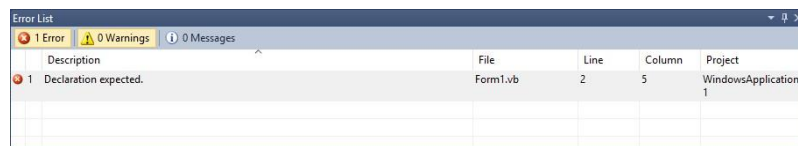
Code editor merupakan sebuah jendela yang digunakan untuk dapat menampilkan dan menuliskan kode-kode dari suatu program aplikasi yang sedang dibuat. Kode-kode program yang dimaksud merupakan sekumpulan perintah yang dituliskan untuk dapat menjalankan objek-objek sehingga dapat melakukan tugas-tugas tertentu sesuai dengan kode yang telah dibuat sebelumnya. Adapun gambaran dari *code editor* dapat diperlihatkan dalam gambar 2.15 di bawah ini.



Gambar 2.15 Tampilan *Code Editor Visual Basic 2010 Express*

9) *Error List*

Error list merupakan jendela yang digunakan untuk menampilkan kesalahan pada suatu daftar koding yang dianggap memiliki masalah dalam pembuatan sebuah program aplikasi sehingga pengembang dapat mengetahui letak permasalahan tersebut secara cepat dan efisien. Berikut gambaran mengenai *error list* seperti yang ditunjukkan oleh gambar 2.16 di bawah ini.



Gambar 2.16 Tampilan *Error List Visual Basic 2010 Express*

2.6.4. Kelebihan Bahasa Pemrograman *Micorosft Visual Basic 2010*

Berdasarkan pembahasan mengenai bahasa pemrograman *Microsoft Visual Basic 2010* di atas maka dapat diketahui kelebihan dari bahasa pemrograman *Microsoft Visual Basic 2010*. Adapun kelebihan yang dimiliki oleh bahasa pemrograman *Microsoft Visual Basic 2010* seperti yang dimaksud tersebut dapat diberikan sebagai berikut:

- 1) Memiliki tampilan *Integrated Development Environment (IDE)* atau antarmuka yang menarik sehingga lebih mudah untuk digunakan (*user friendly*).
- 2) Program aplikasi *Windows* berbasis *Graphical User Interface* sehingga pengguna tidak perlu lagi menuliskan instruksi pemrograman dalam bentuk kode baris (*line*) tetapi pengguna dapat dengan mudah melakukan “*drag and drop*” terhadap objek-objek yang akan digunakan menggunakan beberapa *tools* yang telah tersedia.
- 3) Gaya bahasa atau dialek untuk koding yang mudah dipelajari.
- 4) Memiliki fitur *highlights reference* yang berguna ketika satu simbol/kode diseleksi maka simbol/kode yang sama meskipun pada keadaan ataupun penggunaan yang berbeda maka akan sama-sama terlihat berwarna dengan warna yang sama pula.

- 5) Memiliki fitur *class .math* yang merupakan fitur khusus untuk perhitungan matematika secara akurat.
- 6) Mendukung teknologi *.Net* yang digunakan sebagai *platform* untuk dapat merancang dan membangun *XML Web Service* yang memungkinkan program aplikasi dengan tipe manapun dapat berjalan pada sistem komputer dengan tipe manapun pula.

2.7. XAMPP

XAMPP merupakan suatu *tool* yang menyediakan paket program aplikasi berbasis web yang digunakan untuk keperluan pengembangan suatu program berbasis web (pemrograman web), khususnya *PHP* dan *MySQL* (Nugroho, 2013: 1; Riyanto, 2014: 1). Lebih lanjut Buana (2014: 4) menjelaskan bahwa *XAMPP* yang dikembangkan pada tahun 2002 oleh *Apache Friends* yang didirikan oleh Kai ‘Oswalad’ Sedler dan Kay Vogelgesang tersebut (<http://www.apachefriends.org/about.html> diakses pada 28 Agustus 2017) merupakan perangkat lunak yang bersifat *open source* dimana pengguna dapat mengunduh program tersebut secara gratis (legal) serta dapat mendukung di semua *platform* sistem operasi seperti *Windows*, *Linux*, *Solaris*, maupun *Mac OS*.

Sementara itu, nama *XAMPP* merupakan sebuah singkatan yang diambil dari *X* (*X for Cross Platform*/dapat mendukung banyak sistem operasi), *A* (*Apache HTTP Server*/aplikasi web server), *M* (*MySQL*/aplikasi database server), *P* (*PHP*/bahasa pemrograman), dan *P* (*Perl*) (Kiran, 2014: 18). Dengan kata lain, setiap kata dalam singkatan di atas dapat mewakili nama dari setiap program aplikasi dimana program-program aplikasi tersebut merupakan paket yang sudah terdapat dalam sebuah aplikasi bernama *XAMPP* sehingga pengguna tidak perlu lagi melakukan instalasi dan konfigurasi program-program aplikasi satu per satu secara manual sebab *XAMPP* akan melakukan instalasi dan konfigurasi secara otomatis. Adapun versi *XAMPP* yang digunakan dalam hal ini ialah versi 7.0.0

dimana di dalam *XAMPP* tersebut menyediakan paket perangkat lunak yang terdiri dari:

- 1) *Apache 2.4.18*
- 2) *MariaDB 10.1.9*
- 3) *PHP 7.0.0*
- 4) *phpMyAdmin 4.5.1*
- 5) *OpenSSL 1.0.2e*
- 6) *ADODB 5.18*
- 7) *Mercury Mail Transport System v4.62*
- 8) *FileZilla FTP Server 0.9.41*
- 9) *Webalizer 2.23-04*
- 10) *Strawberry Perl 5.16.1.1*
- 11) *Tomcat 7.0.41*
- 12) *XAMPP Security*
- 13) *XAMPP mailToDisk 1.0*
- 14) *XAMPP Control Version 3.2.2*

2.7.1. PHP (PHP Hypertext Preprocessor)

PHP atau *PHP:Personal Home Page Hypertext Preprocessor* merupakan salah satu jenis bahasa pemrograman berbasis web yang terdiri dari kode-kode (*script*) dimana *script* tersebut bersifat *server-side scripting* yang dapat ditambahkan atau diintegrasikan ke dalam *script HTML* sehingga dapat membuat suatu halaman web menjadi interaktif dan dinamis atau tidak lagi bersifat statis (Anhar, 2010: 23; Prasetyo, 2008: 19). *Server-side scripting* dalam *PHP* memiliki arti bahwa *script PHP* hanya dapat berjalan pada sisi *server* saja dan tanpa ada *server* maka *script PHP* tersebut tidak dapat berjalan sebagaimana mestinya (Nugroho, 2004: 369). Lebih lanjut dapat dikatakan pula bahwa *script PHP* tersebut diletakkan dan sepenuhnya dijalankan/diproses pada sebuah *server* tetapi disertakan dalam bentuk dokumen *HTML* biasa (Sidik, 2012: 4). Adapun dokumen *HTML* biasa yang dihasilkan oleh suatu aplikasi dalam perancangan suatu web bukan merupakan dokumen *HTML* yang dibuat menggunakan editor teks atau

editor *HTML* melainkan merupakan hasil dari pemrosesan *script PHP* menjadi dokumen *HTML*.

Di sisi lain, menurut Kadir (2008: 2) yang mengatakan bahwa konsep kerja dari bahasa pemrograman *PHP* diawali dengan permintaan (*request*) suatu halaman web oleh *browser client*. Berdasarkan *URL (Uniform Resource Locator)* atau alamat dari suatu web pada jaringan internet, *browser* mendapatkan alamat dari *web server*, mengidentifikasi halaman yang dikehendaki, dan menyampaikan segala informasi yang dibutuhkan oleh *web server*. Selanjutnya, *web server* akan mencarikan berkas/*file* yang diminta dan memberikan data tersebut ke *web browser*. *Browser* yang telah mendapatkan data yang dibutuhkan maka akan segera melakukan proses penerjemahan kode dan menampilkan ke layar pengguna (*user*). Akan tetapi, apabila suatu halaman yang dipanggil oleh pengguna (*user*) mengandung *script PHP* maka prinsip kerja sama dengan pemanggilan pada *script HTML* yang telah dibahas sebelumnya. Namun, pada saat permintaan (*request*) oleh *browser client* yang selanjutnya akan dikirim ke *web server* maka *web server* akan memeriksa terlebih dahulu tipe berkas/*file* yang diminta oleh pengguna (*user*). Jika tipe berkas yang diminta adalah *PHP* maka akan diidentifikasi isi *script* dari halaman *PHP* tersebut. Apabila di dalam berkas tidak mengandung *script PHP* maka permintaan dari pengguna (*user*) akan langsung ditampilkan ke layar pengguna (*browser client*), tetapi jika di dalam berkas mengandung *script PHP* maka proses akan diteruskan ke modul *PHP* yang bertindak sebagai mesin yang dapat menerjemahkan *script PHP* dan mengolah *script PHP* ke dalam dokumen *HTML*. Selanjutnya hasil terjemahan ke dalam dokumen *HTML* akan ditampilkan pada layar pengguna (*browser client*).

Berdasarkan konsep kerja tersebut, dapat dikatakan bahwa bahasa pemrograman *PHP* bukan merupakan bahasa pemrograman yang lengkap dikarenakan pada bahasa pemrograman *PHP* tidak menyertakan sebuah *compiler* yang dapat menjadikan bahasa pemrograman *PHP* tersebut dapat memiliki sistem kerja sebagai *compiler* (Nugroho, 2004: 369). Sistem kerja sebagai *compiler* pada bahasa pemrograman *PHP* dapat membantu bahasa pemrograman tersebut untuk

merancang dan membuat sebuah program dengan ekstensi tertentu yang dapat dijalankan sendiri tanpa bantuan dari program lain (pihak ketiga). Akan tetapi, bahasa pemrograman *PHP* tersebut mempunyai konsep kerja sebagai *interpreter* yang selalu membutuhkan *server* pendukung yang disebut dengan *web server* untuk dapat menerjemahkan *script PHP* ke dalam bentuk dokumen *HTML* (pembangun halaman *web*) dan bahasa pemrograman *PHP* itu sendiri sebagai bahasa pemrograman yang dapat menjalankan semua *script* program dalam *PHP* (Sunarfrihantono, 2002: 4).

2.7.2. *phpMyAdmin*

Menurut Nugroho (2013: 71) yang menjelaskan bahwa definisi dari *phpMyAdmin* ialah suatu aplikasi atau *tools* yang dapat digunakan untuk mempermudah pemakai (*user*) maupun pemilik *server* yang merupakan tingkatan tertinggi dari admin (*root*) dalam mengelola/memanajemen *database MySQL* dan *server MySQL* secara visual. Sejalan dengan pendapat yang diutarakan oleh Nugroho tersebut, Buana (2014: 2) dan Firdaus (2007: 15) melanjutkan bahwa *phpMyAdmin* merupakan suatu program *open source* yang berbasis web dirancang dan dibangun menggunakan bahasa pemrograman *PHP* sehingga dapat diperoleh secara gratis tanpa harus membeli lisensi asli serta dapat memudahkan *administrator web server* dalam mengelola *database* tanpa terlebih dahulu harus menguasai perintah atau sintaks-sintaks berbasis teks (*command line*) dari *SQL* dalam pembuatan *database*.

2.7.3. *MySQL*

MySQL yang merupakan *data base management system* pertama kali mulai dikembangkan pada tahun 1994 oleh sebuah perusahaan perangkat lunak bernama *TcX Data Consult AB* yang kemudian berganti label menjadi *MySQL-AB*. “*My*” pada kata *MySQL* bukan berarti pengertian *my* dalam bahasa Inggris, tetapi konon merupakan nama putri dari Michael Widenius yang merupakan seorang pemrogram *data base management system* (DBMS). Versi lain menyebutkan bahwa “*My*”

adalah kependekan dari “*Monty*” yang merupakan julukan untuk Michael Widenius (Fathansyah, 2012: 485).

MySQL atau *My Structured Query Language* merupakan sebuah program *database server* yang mampu menerima dan mengirimkan data dengan sangat cepat, *multi user*, serta menggunakan bahasa standar *SQL (Structured Query Language)* sebagai bahasa dasar *query* (Nugroho, 2005: 1). Sifat dari program *MySQL* tersebut ialah *open source* sehingga dapat diperoleh secara gratis tanpa perlu membeli lisensinya. Selain itu, program *MySQL* yang bersifat *open source* dapat menyatakan pula bahwa program tersebut dilengkapi oleh *source code* (kode yang digunakan untuk membuat *MySQL*) dimana hasil dari program yang berbentuk kode tersebut dapat dijalankan secara langsung di dalam sistem operasi (Kadir, 2009: 15).

Akan tetapi, sejak *MySQL* diakuisisi pada tahun 2009 oleh Perusahaan *Oracle* maka perusahaan yang mengembangkan *MySQL* sebelumnya mulai merancang dan membangun sistem manajemen basis data baru dengan nama *MariaDB* (Bartholomew, 2012: 2).

2.7.4. *MariaDB*

MariaDB merupakan salah satu implementasi sistem manajemen basis data relasional yang disempurnakan dan dikembangkan dari sistem manajemen basis data relasional *MySQL* dengan tujuan untuk menggantikan *MySQL* itu sendiri (Solihin dkk., 2016: 654; Bartholomew, 20012: 2). Selain itu, *MariaDB* juga dikembangkan sebagai perangkat lunak *open source* yang menyediakan antarmuka *SQL* untuk dapat mengakses data yang didistribusikan secara gratis di bawah lisensi *GNU General Public License (GPL)* (<https://mariadb.org/about-us> diakses pada 30 Agustus 2017; Solihin dkk., 2016: 654). Jadi, setiap pengguna (*user*) dapat secara bebas menggunakan *MariaDB*, tetapi dengan batasan perangkat lunak tersebut tidak boleh dijadikan produk turunan yang bersifat komersil.

MariaDB dirancang dan dikembangkan oleh beberapa pengembang utama (inti) yang sama dari *MySQL* termasuk Michael Widenius yang menjadi pelopor pengembangan dari *MariaDB* (Bartholomew, 2012: 2). Penamaan *Maria* dalam

kata *MariaDB* diambil berdasarkan nama dari salah satu putri Michael Widenius yang lebih muda dimana penamaan tersebut tidak jauh berbeda dengan pengambilan nama dari *MySQL* yang diambil berdasarkan nama *My* yang juga merupakan salah satu putri dari Michael Widenius (<https://mariadb.com/kb/en/the-mariadb-library/why-is-the-software-called-mariadb/> diakses pada 09 September 2017).

MariaDB merupakan salah satu *database server* yang sangat populer digunakan di dunia dimana pertumbuhan *database* yang bersifat *open source* ini telah mencapai 12 (dua belas) juta pengguna di seluruh dunia (<https://mariadb.com/products/why-mariadb> diakses pada 09 September 2017). Hal tersebut dapat terjadi karena *MariaDB* memiliki kemampuan yang sangat cepat, terukur, dan kuat, dengan ekosistem yang melimpah akan mesin penyimpanan, *plugin*, dan banyak alat lainnya sehingga membuat *MariaDB* sangat fleksibel untuk digunakan dalam berbagai kasus manajemen basis data (Bartholomew, 2012: 3). Walaupun *MariaDB* didasarkan pada kode pengembangan *MySQL* sehingga memiliki sistem kerja yang sama dengan *MySQL*, tetapi *MariaDB* memiliki kemampuan yang lebih dengan banyak optimalisasi performa yang dilakukan oleh *MariaDB* dibandingkan dengan *MySQL* (Kuntani dkk., 2015: 131). Adapun optimalisasi performa yang dilakukan oleh *MariaDB* ialah dalam hal eksekusi *query* sehingga mampu menangani lebih banyak perintah *SQL* dalam waktu yang bersamaan, memiliki fitur baru seperti peningkatan kinerja, pengujian yang lebih baik, serta perbaikan *bug* yang tidak ditemukan pada *MySQL* (Bartholomew, 2012: 3). Perbaikan-perbaikan tersebut dapat dilakukan oleh beberapa pengembang utama dari *MariaDB* yang dibantu oleh pihak yang berasal dari kontributor dan perusahaan lain yang bekerja sama dengan para pengembang *MariaDB*. Selain itu, *MariaDB* dapat digunakan oleh banyak pengguna dikarenakan memiliki kemampuan kinerja dengan kelas *enterprise*, keamanan dan kredibilitas yang tinggi sehingga dapat digunakan oleh perusahaan/organisasi mulai pada tingkat bank maupun pemerintahan, kompatibilitas yang tinggi, serta arsitektur yang unik dan modern sehingga dapat memudahkan para pengembang maupun pengguna dalam memahami dan menggunakan program aplikasi tersebut untuk memenuhi

kebutuhan atau tujuan tertentu (<https://mariadb.com/products/why-mariadb> diakses pada 09 September 2017).

2.8. Database

Pada sub bab bahasan ini akan menguraikan teori mengenai basis data (*database*) yang mencakup definisi atau konsep dasar dari basis data, tujuan dari rancangan basis data, serta keuntungan dari sistem basis data. Selain itu, akan diuraikan pula teori mengenai sistem manajemen basis data yang juga mencakup definisi, bahasa-bahasa yang terdapat di dalam sistem manajemen basis data, fungsi, komponen, serta arsitektur dari sistem manajemen basis data. Kemudian di akhir dari sub bab ini akan dipaparkan alat-alat pemodelan pada suatu sistem informasi yang mencakup diagram alir dan *entity-relationship diagram* (ERD). Adapun pembahasan mengenai teori-teori tersebut lebih lanjut dipaparkan seperti berikut.

2.8.1. Konsep Basis Data (*Database*)

Basis data atau dalam bahasa inggris dapat pula disebut sebagai *database* merupakan sekumpulan data yang dirancang secara sistematis/terorganisir dengan tujuan agar dapat digunakan/mendukung oleh banyak aplikasi secara efisien dengan cara memusatkan data dan mengontrol *data redundant* (Laudon dan Laudon, 2012: 212). Sedangkan menurut Connolly dan Begg (2010: 65), *database* adalah sekumpulan data yang terhubung secara logis dan data tersebut dirancang untuk dapat memenuhi kebutuhan akan hal informasi dari suatu organisasi. Lebih lanjut Date (2004: 11) menjelaskan bahwa *database* merupakan suatu kumpulan data persisten yang digunakan oleh sistem aplikasi dengan tujuan untuk diberikan kepada suatu perusahaan ataupun organisasi lain yang membutuhkan.

Sejalan dengan ketiga pendapat tersebut, Marlinda (2004: 1) juga mengungkapkan bahwa basis data merupakan suatu susunan/kumpulan data operasional lengkap dari suatu organisasi/perusahaan yang diorganisir/dikelola dan disimpan secara terintegrasi dengan menggunakan metode tertentu pada suatu

komputer sehingga mampu menyediakan informasi secara optimal yang digunakan untuk keperluan pemakainya (*user*). Selain itu, basis data juga dapat diartikan pula sebagai sekumpulan data yang saling terhubung satu dengan lainnya, disimpan di perangkat keras komputer, serta dapat dimanipulasi atau dikelola dengan menggunakan perangkat lunak pada suatu komputer (Jogiyanto, 2005: 217).

Sementara itu, adapula definisi dari sistem basis data yang merupakan suatu sistem yang dapat menyusun dan mengelola *record-record* menggunakan bantuan komputer dimana hal tersebut bertujuan untuk menyimpan atau merekam serta memelihara data operasional lengkap sebuah organisasi/perusahaan sehingga mampu menyediakan informasi yang optimal yang diperlukan pemakai (*user*) untuk proses mengambil suatu keputusan (Marlinda, 2004: 1). Pendapat yang sama juga diungkapkan oleh Date (2004: 6) dimana sistem basis data pada dasarnya merupakan sistem penyimpanan *record* terkomputerisasi atau dengan kata lain sistem terkomputerisasi yang secara umum bertujuan untuk menyimpan informasi serta memungkinkan pengguna (*user*) untuk dapat mengambil ataupun memperbarui informasi sesuai dengan keinginan atau keperluan masing-masing pengguna tersebut. Selanjutnya, James F. Courtney Jr. dan David B. Paradise dalam buku yang berjudul “*Database System for Management*” menjelaskan bahwa definisi dari sistem *database* merupakan sekumpulan *database* yang dapat digunakan/dipakai secara bersama-sama, terdapat personal-personal serta teknik-teknik yang merancang dan mengelola *database* dengan komputer sebagai alat untuk mendukung sistem tersebut (Sutabri, 2005: 161).

Berdasarkan definisi dari beberapa pendapat yang telah dibahas sebelumnya maka dapat disimpulkan bahwa definisi dari *database* atau dapat pula disebut sebagai basis data merupakan sekumpulan data yang terhubung secara logis, dirancang secara sistematis/terorganisir, dan disimpan dengan cara terintegrasi dimana bertujuan agar mudah untuk dimanipulasi ataupun dikelola serta agar dapat digunakan/mendukung oleh banyak aplikasi secara efisien dengan menggunakan metode tertentu pada suatu komputer sehingga mampu menyediakan informasi secara optimal yang digunakan untuk keperluan pemakainya (*user*) baik itu

penggunaan dalam perorangan maupun perusahaan. Sedangkan sistem basis data yang mana dapat disimpulkan berdasarkan beberapa pendapat yang telah dibahas sebelumnya, yakni merupakan suatu sistem yang tersusun atas sekumpulan *record-record* yang saling terhubung satu sama lain (*database*) yang dikelola menggunakan bantuan komputer dengan tujuan untuk menyimpan data, memelihara data operasional lengkap suatu organisasi, maupun *shared* data sehingga mampu menyediakan informasi yang optimal yang diperlukan oleh pemakai (*user*) untuk mengambil suatu keputusan serta memungkinkan pula pengguna tersebut untuk dapat mengambil ataupun memperbarui informasi sesuai dengan keinginan atau keperluan masing-masing pemakainya.

2.8.2. Kriteria Basis Data (*Database*)

Berdasarkan konsep mengenai basis data (*database*) yang telah dipaparkan sebelumnya, maka dapat diketahui kriteria dari basis data (*database*) itu sendiri ialah sebagai berikut (Sutabri, 2005: 161):

- 1) Bersifat *data oriented* dan bukan *program oriented*.
- 2) Dapat digunakan oleh beberapa program aplikasi tanpa perlu mengubah basis datanya.
- 3) Dapat berkembang dengan mudah baik dari segi volume maupun dari segi strukturnya.
- 4) Dapat memenuhi kebutuhan sistem-sistem baru secara mudah.
- 5) Kerangkapan data (*data redundancy*) minimal.

Kelima kriteria tersebut membedakan secara nyata/jelas antara *file* basis data dengan *file* tradisional yang bersifat *program oriented*, yakni hanya dapat digunakan oleh satu program aplikasi saja. Selain itu, pada *file* tradisional juga memiliki sifat yang kaku, yakni jika ada perubahan terhadap struktur data maka harus merubah program aplikasi sesuai dengan struktur data tersebut serta terjadi kerangkapan data dan tidak terjaminnya keselarasan data pada *file* tradisional (*data inkonsistensi*) (Marlinda, 2004: 8; Sutabri, 2005: 161).

2.8.3. Kegunaan Basis Data (*Database*)

Penyusunan suatu basis data (*database*) digunakan untuk mengatasi berbagai masalah pada penyusunan data (Kristianto, 1994: 5), yakni:

- 1) Redundansi dan inkonsistensi data.
- 2) Kesulitan dalam hal pengaksesan data.
- 3) Isolasi data untuk standarisasi.
- 4) *Multiple user* atau basis data dapat digunakan oleh banyak pengguna.
- 5) Masalah keamanan data (*security*).
- 6) Masalah dalam integritas data (kesatuan).
- 7) Masalah data independen (kebebasan data).

2.8.4. Tujuan Sistem Basis Data

Setelah dipaparkan mengenai konsep dari basis data, kriteria-kriteria yang dimiliki dari basis data serta kegunaan dari basis data, maka dapat diketahui tujuan dari perancangan sistem basis data. Adapun tujuan dari basis data tersebut (Sutabri, 2005: 160), yakni meliputi:

- 1) Penyediaan sarana akses yang bersifat fleksibel.
- 2) Pemeliharaan integritas data.
- 3) Proteksi data dari kerusakan.
- 4) Penggunaan yang tidak legal serta penyediaan sarana untuk secara bersama (*shared*), seperti keterhubungan data, pengurangan atau meminimalkan kerangkapan data (*redundancy*), menghilangkan ketergantungan data pada program-program aplikasi, menstandarkan definisi elemen data, dan meningkatkan produktivitas personil sistem informasi.

2.8.5. Keuntungan Sistem Basis Data

Menurut Marlinda (2005: 4), terdapat beberapa keuntungan dari perancangan sistem basis data. Beberapa keuntungan yang dimaksud tersebut ialah:

- 1) Mengurangi kerangkapan data (*data redundancy*), yaitu data yang sama disimpan dalam berkas data yang berbeda-beda sehingga *update* dilakukan berulang-ulang.

- 2) Mencegah ketidakkonsistenan data.
- 3) Keamanan data dapat terjaga, yaitu data dapat dilindungi dari pemakai yang tidak berwenang.
- 4) Integritas dapat dipertahankan.
- 5) Data dapat dipergunakan bersama-sama.
- 6) Menyediakan *recovery data*.
- 7) Memudahkan penerapan standarisasi.
- 8) Data bersifat mandiri (*data independence*).
- 9) Keterpaduan data terjaga dimana memelihara keterpaduan data berarti data harus akurat. Hal ini sangat erat hubungannya dengan pengendalian/pengontrolan kerangkapan data dan pemeliharaan keselarasan data.

2.8.6. Database Management System (DBMS)

Pada bahasan ini akan diterangkan teori mengenai *database management system* yang mencakup definisi, fungsi, serta tujuan dari *database management system*. Selain itu, adapula bahasa-bahasa yang terdapat pada *database management system* serta keuntungan dan kelemahan dari penggunaan *database management system*. Berikut pemaparan lebih lanjut mengenai teori tentang *database management system*.

2.8.6.1. Definisi Database Management System (DBMS)

Database management system atau sistem manajemen basis data merupakan perangkat lunak yang dirancang dengan tujuan untuk dapat membantu dalam memakai dan memelihara sekumpulan data dalam jumlah yang besar (Connolly dan Begg, 2015: 64). Pendapat yang sama juga diungkapkan oleh Ramakrishnan dan Gehrke (2003: 4) yang menjelaskan bahwa definisi dari sistem manajemen basis data adalah sistem berbasis perangkat lunak yang memungkinkan pemakai untuk dapat mendefinisikan, membuat, memelihara, dan mengendalikan pengaksesan terhadap basis data. Selain kedua pendapat tersebut, Marlinda (2004: 6) juga mengungkapkan definisi dari sistem manajemen basis data yang merupakan

kumpulan dari perangkat lunak (*software*) yang dapat mengkoordinasikan semua kegiatan yang berhubungan dengan basis data agar data dapat diakses/dipakai oleh pemakai (*user*).

2.8.6.2. Fungsi-Fungsi *Database Management System*

Menurut Marlinda (2004: 7-8), terdapat beberapa fungsi di dalam sistem manajemen basis data. Adapun fungsi-fungsi yang dimaksud ialah sebagai berikut:

1) *Data Definition*

Data definition merupakan salah fungsi dari sistem manajemen basis data dimana sistem manajemen basis data harus dapat mengolah pendefinisian data.

2) *Data Manipulation*

Data manipulation merupakan salah satu fungsi sistem manajemen basis data dimana sistem manajemen basis data harus dapat menangani permintaan-permintaan dari pemakai untuk mengakses data

3) *Data Security dan Integrity*

Data security dan integrity merupakan salah satu fungsi dari sistem manajemen basis data dimana sistem manajemen basis data tersebut dapat memeriksa keamanan dan integritas data yang didefinisikan oleh DBA.

4) *Data Recovery dan Concurrency*

Terdapat dua tugas dari fungsi dari sistem manajemen basis data ini, yakni:

a) Sistem manajemen basis data dapat menangani kegagalan-kegagalan pengaksesan basis data yang dapat disebabkan oleh kesalahan sistem, kerusakan *disk*, dan lain sebagainya.

b) Sistem manajemen basis data harus dapat mengendalikan pengaksesan data yang konkuren, yaitu bila satu data diakses secara bersama-sama oleh lebih dari satu pemakai pada saat bersamaan.

5) *Data Dictionary*

Data dictionary merupakan salah satu fungsi dari sistem manajemen basis data dimana sistem manajemen basis data harus dapat menyediakan *data dictionary*.

6) *Performance*

Performance merupakan salah satu fungsi dari sistem manajemen basis data dimana sistem manajemen basis data harus dapat menangani unjuk kerja dari semua fungsi seefisien mungkin.

2.8.6.3. Tujuan *Database Management System*

Di sisi lain, menurut Hariyanto (2004: 4) yang menyebutkan bahwa sistem manajemen basis data juga mempunyai beberapa tujuan utama dari perancangannya tersebut, yakni:

- 1) Menyediakan lingkungan yang nyaman dan efisien untuk penyimpanan dan pengambilan data dari basis data.
- 2) Sistem manajemen basis data berperan memberi abstraksi data tingkat tinggi ke pemakai.
- 3) Sistem manajemen basis data adalah perangkat lunak yang dirancang untuk membantu pengelolaan dan pemakaian koleksi data yang besar.
- 4) Pemakai berurusan dengan abstraksi tingkat tinggi yang lebih akrab, nyaman, dan fleksibel bukan rincian fisik penyimpanan data yang lebih rumit.

Selain itu, terdapat pula tujuan lain dari perancangan sistem manajemen basis data, yakni:

- 1) Menghindari redundansi dan inkonsistensi data.
- 2) Menghindari kesulitan pengaksesan data.
- 3) Menghindari isolasi data.
- 4) Menghindari terjadinya anomali pengaksesan konkuren.
- 5) Menghindari masalah-masalah keamanan.
- 6) Menghindari masalah-masalah integritas.

2.8.6.4. Bahasa-Bahasa dalam *Database Management System*

Sistem manajemen basis data merupakan perantara antara pemakai dengan basis data yang terdapat di dalam suatu *disk* dan cara untuk berinteraksi/berkomunikasi tersebut dapat dilakukan menggunakan suatu bahasa

khusus yang disebut dengan bahasa basis data atau *database language* yang mana terdiri dari sejumlah perintah/*statement* (Hutahaean, 2014: 69).

Di lain sisi, definisi yang sama dari bahasa basis data juga diungkapkan oleh Hariyanto (2004: 109), yakni merupakan bahasa pada sistem basis data yang digunakan untuk melakukan pendefinisian dan manipulasi basis data. Lebih lanjut Bambang menjelaskan bahwa implementasi dari bahasa sistem manajemen basis data dapat bervariasi sesuai dengan keperluan dari masing-masing perusahaan yang merancang sistem tersebut. Namun pada prinsipnya, bahasa basis data tersebut dapat dikategorisasikan ke dalam tiga komponen bahasa (Hariyanto, 2004: 110-113), yaitu:

1) *Data Definition Language* (DDL)

Data definition language merupakan salah satu dari komponen bahasa basis data yang digunakan untuk mendefinisikan struktur data. Hasil dari perintah *data definition language* tersebut berupa tabel-tabel yang selanjutnya disimpan dalam suatu file khusus yang disebut sebagai *data dictionary/directory*. Komponen bahasa ini banyak digunakan oleh para pemakai dengan tingkatan administrator basis data dimana hal tersebut dapat berguna dalam hal perencanaan atau perancangan file-file basis data.

2) *Data Manipulation Language* (DML)

Data manipulation language merupakan salah satu komponen bahasa basis data yang digunakan untuk mengakses serta memanipulasi data atau dapat dikatakan pula komponen ini digunakan oleh para pemakai untuk keperluan memanipulasi data. Selain itu, menurut Marlinda (2004: 6) yang mengatakan bahwa pada dasarnya terdapat 2 (dua) macam tipe dari *data manipulation language*, yaitu:

a) Prosedural

Pada tipe ini pemakai harus menentukan terlebih dahulu data apa yang dibutuhkan dan bagaimana cara mendapatkannya. Sebagai contoh aplikasi untuk tipe prosedural ialah *dBase III* dan *foxbase*.

b) Non-prosedural

Pada tipe non-prosedural, pemakai harus menentukan data apa yang dibutuhkan tanpa menentukan bagaimana cara mendapatkannya. Sebagai contoh aplikasi pada tipe ini adalah SQL dan QBE.

3) *Data Control Language* (DCL)

Data control language merupakan komponen bahasa basis data yang digunakan untuk mengatur perekaman atau penyimpanan data secara fisik. Komponen bahasa basis data ini digunakan oleh para pemakai sistem basis data untuk keperluan dalam mengatur file-file data secara fisik. Selain itu, ada pula perintah-perintah yang termuat dalam komponen ini, antara lain ialah perintah-perintah merekam yang terdiri dari *Write Record*, *Create Table* serta perintah-perintah menghapus yang terdiri dari *Drop*, *Delete Table*.

2.8.6.5. Keunggulan dan Kelemahan *Database Management System*

Menurut Hariyanto (2004: 5), ada beberapa keunggulan yang dapat diperoleh dari perancangan serta penggunaan sistem manajemen basis data. Beberapa keunggulan tersebut, ialah:

- 1) Pengendalian terhadap redundansi data.
- 2) Konsistensi data.
- 3) Cakupan informasi lebih luas/banyak karena data tersimpan secara bersama-sama.
- 4) Pemakaian data secara bersama-sama.
- 5) Peningkatan integritas data.
- 6) Standarisasi yang optimal.
- 7) Skala ekonomi.
- 8) Penyeimbang kebutuhan-kebutuhan akan sumber daya yang terbatas.
- 9) Peningkatan terhadap pengaksesan data.
- 10) Peningkatan produktivitas.
- 11) Peningkatan konsumsi.
- 12) Peningkatan terhadap layanan *back up* dan pemulihan data (*recovery*).

Selain memiliki beberapa keunggulan, sistem manajemen basis data juga memiliki beberapa kelemahan. Kelemahan-kelemahan tersebut antara lain, ialah:

- 1) Kompleksitas yang tinggi.
- 2) Kapasitas perangkat lunak yang besar.
- 3) Biaya sistem manajemen basis data untuk pengadaan, operasi, dan perawatan.
- 4) Penambahan terhadap biaya perangkat keras untuk mengoperasikan sistem manajemen basis data.
- 5) Biaya konversi dari sistem lama ke sistem yang baru.
- 6) Kinerja yang rendah bila tidak mampu menggunakan dengan baik.
- 7) Resiko yang cukup tinggi bila terjadi suatu kegagalan.

2.8.7. Alat / Perangkat Pemodelan Sistem Informasi

Pada bahasan ini akan dipaparkan mengenai macam-macam alat atau perangkat pemodelan yang digunakan dalam perancangan sebuah sistem informasi. Adapun alat-alat atau perangkat pemodelan yang dimaksud dipaparkan lebih lanjut sebagai berikut:

2.8.7.1. Bagan Alir (*Flowchart*)

Flowchart atau bagan alir merupakan bagan - bagan yang mempunyai arus yang menggambarkan langkah-langkah penyelesaian suatu masalah (Ladjamudin, 2005: 263). Sedangkan, menurut Jogiyanto (2005: 795) yang juga mengungkapkan definisi dari bagan alir, yakni merupakan bagan (*chart*) yang menunjukkan alir (*flow*) di dalam suatu program atau suatu prosedur sistem secara logika. Sejalan dengan pendapat Jogiyanto, Krismiaji (2010: 71) juga menjelaskan bahwa bagan alir merupakan teknik analitis yang digunakan untuk menjelaskan aspek-aspek sistem informasi secara jelas, tepat, serta logis. Lebih lanjut Krismiaji menerangkan bahwa pada bagan alir tersebut terdapat serangkaian simbol standar yang digunakan untuk menguraikan prosedur pengolahan transaksi yang digunakan oleh sebuah perusahaan dan sekaligus menguraikan aliran data dalam sebuah sistem.

Jadi, berdasarkan definisi dari beberapa pendapat yang telah dibahas tersebut maka dapat disimpulkan bahwa definisi dari *flowchart* atau bagan alir

merupakan suatu teknik analisis yang digunakan untuk menjelaskan aspek-aspek mengenai sistem informasi yang dirangkum secara jelas, tepat, dan logis melalui bagan yang memiliki serangkaian simbol standar yang berguna untuk menguraikan prosedur pengolahan transaksi dalam suatu program / sistem secara logika, menguraikan aliran data pada sebuah sistem, serta dapat menggambarkan langkah-langkah penyelesaian suatu masalah.

Di lain sisi, menurut Jogiyanto (2005: 796) yang menjelaskan bahwa bagan alir dapat dibagi menjadi 5 (lima) macam jenis, yakni:

1) Bagan Alir Sistem (*System Flowchart*)

Bagan alir sistem merupakan bagan yang menunjukkan arus pekerjaan secara keseluruhan dari suatu sistem. Bagan tersebut berguna untuk menjelaskan urutan-urutan dari prosedur-prosedur yang ada di dalam sistem (Jogiyanto, 2005: 796). Jadi dengan kata lain bagan alir sistem menunjukkan apa yang dikerjakan di dalam suatu sistem dengan cara menggunakan bagan alir yang dapat menggambarkan urutan-urutan dari prosedur-prosedur yang sedang berjalan di dalam sistem.

2) Bagan Alir Dokumen (*Document Flowchart*)

Bagan alir dokumen atau dapat disebut pula bagan alir formulir (*form flowchart*) atau *paperwork flowchart* merupakan bagan alir yang menunjukkan arus dari laporan serta formulir dan termasuk pula tembusan-tembusannya (Jogiyanto, 2005: 800). Selain itu, ada pula pendapat lain yang mengatakan bahwa bagan alir dokumen merupakan bagan yang dapat menggambarkan aliran dokumen dan informasi antar area pertanggungjawaban di dalam sebuah organisasi (Krismiaji, 2010: 75). Jadi dapat disimpulkan dari kedua pendapat tersebut bahwa bagan alir dokumen merupakan bagan yang dapat menunjukkan / menggambarkan aliran dokumen serta informasi pada suatu sistem.

3) Bagan Alir Skematik (*Schematic Flowchart*)

Bagan alir skematik merupakan bagan alir yang mirip dengan bagan alir sistem, yakni dapat menggambarkan prosedur-prosedur dalam suatu sistem. Hanya saja perbedaannya terdapat pada penggunaan simbol-simbol, yaitu pada bagan alir skematik selain menggunakan simbol-simbol bagan alir sistem, juga

menggunakan gambar-gambar komputer serta peralatan-peralatan lain yang digunakan. Tujuan dari penggunaan gambar-gambar tersebut ialah untuk mempermudah seseorang dalam memahami simbol-simbol bagan alir (Jogiyanto, 2005: 802).

4) Bagan Alir Program (*Program Flowchart*)

Bagan alir program merupakan bagan yang dapat menjelaskan secara rinci langkah-langkah dari proses program. Menurut Jogiyanto (2005: 802), bagan alir program dapat dibagi ke dalam dua macam bagan alir, yaitu:

a) Bagan Alir Logika Program (*Program Logic Flowchart*)

Bagan alir ini digunakan untuk menggambarkan setiap langkah di dalam suatu program komputer secara logika yang disiapkan oleh seorang analis sistem.

b) Bagan Alir Komputer Terinci (*Detailed Computer Program Flowchart*)

Bagan alir komputer terinci digunakan untuk menggambarkan intruksi-intruksi program komputer secara terinci yang disiapkan oleh seorang pemrogram.

5) Bagan Alir Proses (*Process Flowchart*)

Bagan alir proses merupakan bagan alir yang banyak digunakan dalam teknik perindustrian. Bagan alir ini pula sangat berguna bagi seorang analis sistem untuk dapat menggambarkan proses dalam suatu prosedur (Jogiyanto, 2005: 805).

2.8.7.2. Entity-Relationship Model (E-R Model)

Entity-relationship model merupakan suatu pendekatan *top-down* dalam mendesain suatu basis data yang dimulai dengan mengidentifikasi data-data penting yang disebut sebagai *entities* dan *relationship* antara data-data tersebut direpresentasikan dalam bentuk sebuah model (Connolly dan Begg, 2015: 405). Sejalan dengan pendapat yang dikemukakan oleh Connolly dan Begg tersebut, Marlinda (2004: 17) juga menjelaskan bahwa model dari *entity-relationship* merupakan suatu model yang digunakan untuk menjelaskan hubungan antara data dalam suatu basis data yang berdasarkan pada suatu persepsi bahwa fakta dari dunia

nyata tersebut terdiri dari sekumpulan objek dasar yang saling terhubung antar satu sama lain. Marlinda melanjutkan bahwa hubungan antar objek tersebut direpresentasikan menggunakan simbol-simbol grafis tertentu yang disajikan dengan menggunakan sesuatu hal yang kemudian disebut sebagai *entity* dan *relationship*. Selain kedua pendapat tersebut, ada pula Fathansyah (2004: 79) dalam buku yang berjudul *Basis Data* mengungkapkan definisi dari model *entity-relationship* yang merupakan suatu model yang terdiri atas komponen-komponen himpunan entitas dan himpunan relasi yang mana masing-masing dari himpunan tersebut memiliki atribut-atribut yang merepresentasikan seluruh fakta dari dunia nyata yang kita tinjau yang digambarkan secara sistematis menggunakan *entity-relationship diagram* (ERD). Jadi, berdasarkan ketiga pendapat yang telah dipaparkan sebelumnya maka dapat disimpulkan bahwa definisi dari *entity-relationship model* merupakan suatu model hasil dari perancangan basis data menggunakan pendekatan *top-down* yang digunakan untuk menjelaskan hubungan antar data-data dalam sebuah basis data menggunakan komponen-komponen himpunan entitas dan himpunan relasi yang masing-masing memiliki atribut yang berguna untuk dapat merepresentasikan seluruh fakta dari dunia nyata yang digambarkan secara sistematis menggunakan *entity-relationship diagram* (ERD).

Di lain sisi, menurut Marlinda (2004: 17-24) dan Connolly dan Begg (2015: 406-429) yang lebih lanjut menjelaskan bahwa terdapat komponen-komponen pembentuk dari *entity-relationship model*. Komponen-komponen pembentuk yang dimaksud tersebut dipaparkan sebagai berikut:

1) Entitas (*Entity*) dan Himpunan Entitas (*Entity Set*)

Entitas merupakan suatu objek dasar atau individu yang dapat mewakili sesuatu yang nyata dimana setiap objek-objek tersebut mempunyai sekumpulan sifat yang sama dan unik yang dapat mengidentifikasi entitas tersebut sehingga dapat membedakan antara entitas satu dengan entitas yang lain. Sebagai contoh untuk entitas tersebut ialah setiap mahasiswa pada suatu perguruan tinggi. Sedangkan himpunan entitas merupakan kumpulan dari entitas yang memiliki sifat yang sama atau sejenis. Sebagai contoh untuk himpunan entitas ialah kumpulan dari

mahasiswa pada suatu perguruan tinggi tersebut. Himpunan entitas yang dilihat dari segi sifatnya dapat dibagi menjadi dua jenis entitas, yakni:

- a) Entitas yang Bersifat Fisik, yaitu entitas yang dapat dilihat, seperti: rumah, kendaraan, mahasiswa, dosen, dan lain sebagainya.
- b) Entitas yang Bersifat Logik atau Konseptual, yaitu entitas yang tidak dapat dilihat, seperti: pekerjaan, rencana, mata kuliah, dan lain sebagainya.

Pada *entity-relationship diagram* (ERD), entitas dapat digambarkan dengan simbol persegi panjang beserta nama entitas berhuruf kapital yang terdapat di dalam persegi panjang tersebut.

2) Atribut (*Attribute*)

Atribut merupakan karakteristik atau sifat yang dimiliki oleh setiap entitas ataupun *relationship* yang mampu menyediakan penjelasan secara rinci mengenai informasi yang terdapat di dalam suatu entitas ataupun *relationship* tersebut (*attribute value*). Setiap atribut terdapat himpunan nilai-nilai yang mungkin dimiliki oleh satu atau lebih atribut yang mana kemudian dapat disebut sebagai *domain*. Pada *entity-relationship diagram* (ERD), atribut digambarkan dengan bentuk oval. Selain itu, menurut Connolly dan Begg (2015: 413-417) yang menerangkan bahwa atribut dapat dibagi menjadi beberapa tipe atribut. Tipe-tipe atribut yang dimaksud tersebut dapat dipaparkan sebagai berikut:

- a) Atribut Sederhana dan Atribut Komposit (*Simple and Composite Attribute*)
Atribut sederhana merupakan atribut yang hanya terdiri atas komponen tunggal atau dapat dikatakan pula tidak dapat diuraikan lagi menjadi sub atribut lain. Sedangkan atribut komposit merupakan atribut yang terdiri dari beberapa komponen atau dengan kata lain dapat diuraikan lagi menjadi beberapa sub atribut lain.
- b) Atribut Bernilai Tunggal dan Atribut Bernilai Banyak (*Single-Valued and Multi-Valued Attribute*)

Atribut bernilai tunggal merupakan atribut yang hanya dapat menyimpan atau hanya memiliki paling banyak satu nilai untuk setiap entitas yang ada. Sedangkan atribut bernilai banyak merupakan atribut yang dapat

menyimpan atau memiliki lebih dari satu nilai untuk setiap entitas yang ada yang disimbolkan dalam bentuk oval dengan dua garis tepi di luarnya.

c) Atribut Turunan (*Derived Attribute*)

Atribut turunan merupakan atribut yang nilainya dapat diturunkan dari atribut yang terkait atau atribut lain dan tidak harus dalam entitas yang sama. Pada *entity-relationship diagram* (ERD), atribut turunan digambarkan dalam bentuk oval dengan garis tepi putus-putus.

d) Atribut Null (*Non-Mandatory Attribute*)

Atribut null merupakan suatu atribut yang tidak memiliki ataupun tidak diketahui nilai/harganya pada suatu entitas.

e) Atribut Identitas (*Key Attribute*)

Atribut identitas merupakan atribut yang berguna sebagai kunci pada suatu tabel atau entitas dimana atribut identitas tersebut memiliki sifat yang unik dan tidak ada yang menyamai sehingga dapat membedakan suatu entitas dengan entitas yang lain (Yanto, 2016: 36). Di sisi lain, Yanto menyebutkan bahwa atribut identitas tersebut dapat dibagi menjadi beberapa jenis *keys*. Jenis-jenis *keys* yang dimaksud tersebut, yaitu:

a. *Super Key*

Super key adalah suatu atribut ataupun himpunan atribut yang secara unik dapat digunakan untuk mengidentifikasi entitas/himpunan entitas atau baris data yang ada dalam suatu tabel (Yanto, 2016: 37).

b. *Candidate Key*

Candidate key adalah himpunan atribut minimal/*super key* minimum yang secara unik dapat mengidentifikasi setiap entitas yang ada dalam suatu tabel (Connolly dan Begg, 2015: 415).

c. *Primary Key*

Primary key adalah atribut identitas yang dapat dipilih dari *candidate key* dengan banyak pertimbangan sehingga menjadikan identitas tersebut sering digunakan, lebih ringkas, serta paling unik dari atribut yang lain (Connolly dan Begg, 2015: 415).

d. *Composite Key*

Composite key adalah identitas atau kunci yang terdiri dari dua atribut atau lebih. Atribut-atribut tersebut merupakan atribut dengan nilai yang sama unik jika dirangkai menjadi satu kesatuan dan tidak terpisah antar sama lain atau masih dalam entitas yang sama (Connolly dan Begg, 2015: 416; Yanto, 2016: 38).

e. *Alternate Key*

Alternate key adalah *candidate key* yang tidak terpilih menjadi *primary key* atau suatu atribut yang dapat menggantikan *primary key* (Marlinda, 2004: 30).

f. *Foreign Key*

Foreign key adalah atribut dengan *domain* yang sama pada suatu *relationship* dimana merupakan atribut *primary key* dalam sebuah relasi, tetapi pada relasi yang lain atribut tersebut hanya sebagai atribut biasa (Marlinda, 2004: 30).

3) *Relationship*

Relationship merupakan suatu himpunan yang menunjukkan hubungan antara satu atau beberapa entitas yang sama ataupun berbeda (Connolly dan Begg, 2015: 408). Pada *entity-relationship diagram* (ERD), *relationship* disimbolkan dalam bentuk belah ketupat. Di sisi lain, pada *relationship* terdapat derajat *relationship* atau *relationship degree* yang merupakan jumlah entitas yang dapat berpartisipasi atau terhubung dalam suatu *relationship* (Ladjamudin, 2005: 145). Lebih lanjut Ladjamudin menerangkan bahwa pada *relationship*, derajat *relationship* yang sering digunakan di dalam *entity-relationship diagram* (ERD) dapat dipaparkan sebagai berikut:

a) *Unary Relationship*

Unary relationship merupakan suatu model *relationship* yang terjadi di antara entitas yang berasal dari himpunan entitas yang sama. Model *relationship* ini sering disebut pula sebagai *Recursive Relationship* atau *Reflective Relationship*.

b) *Binary Relationship*

Binary relationship merupakan model *relationship* yang dapat menghubungkan dua entitas yang tidak sejenis menjadi satu himpunan entitas yang sejenis. *Relationship* model ini paling umum digunakan dalam pembuatan model data.

c) *Ternary Relationship*

Ternary relationship merupakan model *relationship* yang dapat menghubungkan ketiga entitas yang tidak sejenis.

4) Entitas Kuat dan Entitas Lemah (*Strong and Weak Entity*)

Pada entitas yang telah diuraikan sebelumnya, terdapat dua jenis entitas yang diklasifikasi menjadi entitas kuat dan entitas lemah (Connolly dan Begg, 2015: 417). Kedua entitas tersebut lebih lanjut dapat dipaparkan sebagai berikut:

a) Entitas Kuat (*Strong Entity*)

Entitas kuat merupakan salah satu jenis entitas yang mana keberadaannya tidak bergantung pada suatu entitas ataupun beberapa entitas yang lain (Connolly dan Begg, 2015: 417). Sebagai contoh untuk entitas kuat ialah pada entitas MAHASISWA, entitas DOSEN, entitas MATA KULIAH dimana entitas-entitas tersebut keberadaannya tidak bergantung pada entitas-entitas yang lain dan pula bukan merupakan bagian (sub) dari entitas yang lain. Pada entitas kuat terdapat pula atribut identifikasi (*keys*) yang unik baik itu tersusun atas satu atribut identifikasi maupun beberapa atribut identifikasi. Pada *entity-relationship diagram* (ERD), entitas kuat disimbolkan dalam bentuk persegi panjang dengan satu garis tepi.

b) Entitas Lemah (*Weak Entity*)

Entitas lemah merupakan salah satu jenis entitas yang keberadaannya bergantung pada suatu entitas ataupun beberapa entitas yang lain (Connolly dan Begg, 2015: 417). Sebagai contoh untuk entitas lemah ialah pada entitas HOBI, entitas ORANG TUA dimana entitas-entitas tersebut keberadaannya bergantung pada entitas yang lain, yakni entitas MAHASISWA. Selain itu, karakteristik dari entitas lemah tersebut, yakni tidak memiliki atribut identifikasi yang unik untuk dapat berhubungan dengan entitas yang lain

kecuali dengan entitas yang memiliki kunci utama atau atribut identifikasinya (*primary key*). Pada *entity-relationship diagram* (ERD), entitas lemah dapat disimbolkan dalam bentuk persegi panjang dengan dua garis tepi.

5) *Relationship* Kuat dan *Relationship* Lemah

Pada *relationship* yang telah diuraikan sebelumnya, terdapat dua jenis *relationship* yang diklasifikasi menjadi *relationship* kuat dan *relationship* lemah (Yanto, 2016: 38). Kedua *relationship* tersebut lebih lanjut dapat dipaparkan sebagai berikut:

a) *Relationship* Kuat

Relationship kuat merupakan jenis *relationship* yang digunakan untuk menghubungkan antar entitas kuat (Yanto, 2016: 38). Pada *entity-relationship diagram* (ERD), *relationship* kuat disimbolkan dalam bentuk belah ketupat dengan satu garis tepi.

b) *Relationship* Lemah

Relationship lemah atau dapat pula disebut sebagai *identifying relationship* merupakan jenis *relationship* yang digunakan untuk menghubungkan entitas kuat dengan entitas lemah (Yanto, 2016: 38). Pada *entity-relationship diagram* (ERD), *relationship* lemah disimbolkan dalam bentuk belah ketupat dengan dua garis tepi.

6) Kardinalitas (*Cardinality*)

Kardinalitas merupakan suatu cara yang dipakai untuk menjelaskan jumlah hubungan maksimum yang mungkin terjadi pada suatu *relationship* antara suatu entitas dengan entitas pada suatu himpunan entitas yang lain (Connolly dan Begg, 2015: 424). Connolly dan Begg lebih lanjut menjelaskan bahwa terdapat tiga jenis kardinalitas yang dapat terjadi pada suatu *relationship*. Tiga jenis kardinalitas yang dimaksud tersebut lebih lanjut dapat dipaparkan sebagai berikut:

a) *One to One* (1 : 1)

One to One merupakan jenis kardinalitas yang terjadi untuk setiap entitas pada suatu himpunan entitas “A” yang hanya dapat berhubungan tepat satu

atau paling banyak satu entitas pada suatu himpunan entitas “B” dan berlaku pula sebaliknya setiap entitas pada himpunan entitas “B” hanya dapat berhubungan paling banyak satu entitas pada himpunan entitas “A” (Fathansyah, 2004: 77). Jadi, dapat pula diartikan sebagai hubungan yang terjadi antara satu entitas pada suatu himpunan entitas dengan satu entitas pada suatu himpunan entitas yang lain (Marlinda, 2004: 20).

b) *One to Many* (1 : M) atau *Many to One* (M : 1)

One to Many atau *Many to One* merupakan jenis kardinalitas yang terjadi untuk setiap entitas pada suatu himpunan entitas “A” dapat berhubungan dengan banyak entitas pada suatu himpunan “B” dan tidak berlaku pula sebaliknya atau banyak entitas pada suatu himpunan entitas “A” hanya dapat berhubungan paling banyak satu entitas pada suatu himpunan entitas “B” dan tidak berlaku pula sebaliknya (Fathansyah, 2004: 78). Jadi, dengan kata lain dapat pula diartikan sebagai hubungan yang terjadi antara satu entitas dengan banyak entitas yang lain atau banyak entitas dengan satu entitas yang lain (Marlinda, 2004: 20).

c) *Many to Many* (M : M)

Many to Many (M : M) merupakan jenis kardinalitas yang terjadi untuk setiap entitas pada suatu himpunan entitas “A” dapat berhubungan dengan banyak entitas pada suatu himpunan entitas “B” dan berlaku pula sebaliknya setiap entitas pada himpunan entitas “B” dapat berhubungan dengan banyak entitas pada himpunan entitas “A” (Fathansyah, 2004: 79). Jadi, dapat pula diartikan sebagai hubungan yang terjadi antara banyak entitas pada suatu himpunan entitas dengan banyak entitas pada suatu himpunan entitas yang lain (Marlinda, 2004: 20).

7) Partisipasi (*Participation*)

Partisipasi merupakan suatu cara yang dipakai untuk menjelaskan apakah keberadaan suatu entitas tergantung pada hubungan suatu entitas yang lain (Marlinda, 2004: 21). Sejalan dengan pendapat tersebut, Connolly dan Begg (2015: 425-426) melanjutkan bahwa keberadaan entitas tersebut dapat berupa semua entitas yang dapat terlibat pada suatu hubungan tertentu yang kemudian

disebut sebagai *mandatory participation* atau *total participation* atau hanya beberapa entitas saja yang kemudian disebut sebagai *optional participation* atau *partial participation* dimana dalam sebuah *relationship* partisipasi tersebut dapat ditunjukkan dengan nilai minimum pada kedua sisi yang berlawanan dari *relationship* atau di antara kedua entitas yang saling terhubung. Selain itu, kedua bentuk partisipasi yang telah diutarakan oleh Connolly dan Begg sebelumnya tersebut dapat dijelaskan lebih lanjut sebagai berikut:

a) *Total Participation* atau *Mandatory Participation*

Total participation merupakan salah satu bentuk partisipasi yang dapat menunjukkan ketergantungan penuh untuk setiap entitas terhadap entitas yang lain dalam suatu *relationship* atau dapat dikatakan pula setiap entitas harus memiliki hubungan terhadap entitas yang lain pada suatu *relationship* (Singh, 2011: 248; 285). Jadi, dalam *total participation* atau lebih dikenal dengan sebutan *Mandatory Participation* nilai yang dimiliki tidak akan bernilai null (kosong) atau bahkan harus memiliki nilai dalam atribut pada suatu hubungan entitas dengan nilai minimum ialah 1 yang disimbolkan dengan dua garis lurus (Bagui dan Earp, 2012: 128).

b) *Partial Participation* atau *Optional Participation*

Partial participation merupakan salah satu bentuk partisipasi yang dapat menunjukkan ketergantungan tidak penuh untuk setiap entitas terhadap entitas yang lain dalam suatu *relationship* atau dapat dikatakan pula setiap entitas hanya beberapa atau tidak harus semua memiliki hubungan terhadap entitas yang lain pada suatu *relationship* (Singh, 2011: 248; 285). Jadi, dalam *partial participation* atau lebih dikenal dengan sebutan *optional participation* nilai yang dimiliki dapat bernilai null (kosong) atau dapat dikatakan pula mungkin ada yang tidak memiliki nilai (kosong) dalam atribut pada suatu hubungan entitas tersebut dimana nilai minimum pada *partial participation* ini ialah 0 yang disimbolkan dengan satu garis lurus (Bagui dan Earp, 2012: 129).

8) Normalisasi

Normalisasi merupakan suatu teknik perancangan basis data yang digunakan untuk menggabungkan/mengelompokkan suatu atribut-atribut/data-data dalam suatu entitas atau tabel secara optimal sehingga membentuk sebuah *relationship* dengan jumlah kerangkapan data yang sedikit (Marlinda, 2004: 115; Connolly dan Begg, 2015: 451). Tujuan dari normalisasi ialah untuk mengidentifikasi sekumpulan *relationship* sehingga tersedia data bagi suatu organisasi atau perusahaan yang membutuhkan (Connolly dan Begg, 2015: 452). Connolly dan Begg melanjutkan bahwa terdapat karakteristik pada sekumpulan *relationship* tersebut, yakni:

- a) Kapasitas atau atribut dengan jumlah yang seminimal mungkin.
- b) Terdapat ketergantungan fungsional dalam suatu tabel yang sama.
- c) Kerangkapan data minimal.

Selain itu, sejalan dengan pendapat mengenai normalisasi yang telah diuraikan sebelumnya, Kroenke (dalam Kadir, 1999: 65) melanjutkan bahwa normalisasi dapat diartikan pula sebagai suatu proses yang digunakan untuk mengubah suatu tabel yang mana masih memiliki kejanggalan atau masalah tertentu ke dalam dua buah tabel atau lebih yang tidak memiliki kejanggalan atau masalah tersebut lagi. Adapun masalah yang dimaksud oleh Kroenke tersebut dikenal dengan istilah anomali. Anomali atau dapat pula disebut *update anomalies* merupakan suatu masalah atau efek samping yang terjadi karena terdapat kerangkapan data (*data redundancy*) dalam suatu tabel sehingga terjadi ketidakkonsistenan data atau dapat membuat suatu data menjadi hilang ketika data yang lain dihapus ataupun jika ada perubahan pada suatu data maka pada data yang sama secara otomatis tidak ikut berubah dan harus mengubah data secara manual satu per satu (lebih dari satu baris data tersebut) (Kadir, 1999: 65; Connolly dan Begg, 2015: 455; Sarosa, 2009: 6). Menurut Connolly dan Begg yang mengungkapkan bahwa terdapat 3 (tiga) macam anomali yang terjadi dalam suatu basis data. Ketiga macam anomali yang dimaksud tersebut lebih lanjut dipaparkan sebagai berikut:

a) Anomali Penyisipan (*Insertion Anomalies*)

Anomali penyisipan merupakan suatu anomali yang terjadi ketika data yang hendak dilakukan penyisipan *record* pada suatu tabel maka data tersebut tidak mungkin dapat diproses atau dimasukkan (Kadir, 1999: 66). Hal tersebut terjadi karena pada data yang akan disisipkan terdapat atribut yang masih kosong dimana atribut tersebut merupakan atribut yang menjadi *primary key*.

b) Anomali Penghapusan (*Deletion Anomalies*)

Anomali penghapusan merupakan suatu anomali yang terjadi sewaktu dilakukan penghapusan terhadap data pada suatu tabel tidak sesuai dengan yang diharapkan atau dengan kata lain data yang tidak ingin dihapus ikut menghilang pula (Kadir, 1999: 67).

c) Anomali Pengubahan (*Modification Anomalies*)

Anomali pengubahan merupakan suatu anomali yang terjadi ketika terdapat pengubahan terhadap suatu data pada suatu *record* dari tabel maka pada data yang sama/sejenis tersebut secara otomatis tidak ikut berubah dan harus mengubah data secara manual satu per satu (Kadir, 1999: 66).

Di sisi lain, dalam normalisasi dimana sebuah basis data dapat dikatakan baik atau efisien jika pada setiap tabel yang menjadi komponen pembentuk basis data tersebut juga dalam keadaan baik (normal). Selanjutnya, sebuah tabel dapat dikategorikan baik (efisien) atau normal jika memiliki 3 (tiga) kriteria sebagai berikut (Kusrini, 2007: 40):

- a) Jika terdapat dekomposisi/penguraian pada suatu tabel maka dekomposisi tersebut harus terjamin aman (*lossless-join decomposition*).
- b) Terpeliharanya ketergantungan fungsional pada saat terjadi perubahan terhadap suatu data dalam tabel (*dependency preservation*).
- c) Memenuhi ketentuan *Boyce-Codd Normal Form* (BCNF), tetapi jika tidak dapat diminimalkan atau dinormalkan lagi maka telah memenuhi bentuk normalisasi ketiga.

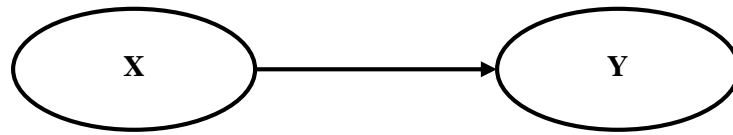
Adapun penjelasan lebih lanjut mengenai ketiga kriteria tersebut, yakni dapat dipaparkan sebagai berikut:

a) *Lossless-Join Decomposition*

Lossless-join decomposition atau dapat pula disebut sebagai *nonloss-join* atau *nonadditive-join* merupakan bagian dari dekomposisi yang merupakan suatu proses yang dilakukan untuk menguraikan suatu tabel (tabel besar) menjadi dua atau lebih tabel baru (tabel kecil) (Kusrini, 2007: 102; Connolly dan Begg, 2015: 496). Lebih lanjut Connolly dan Begg menjelaskan bahwa tabel-tabel baru yang merupakan hasil dari penguraian dari suatu tabel (besar) tersebut jika digabungkan kembali maka akan menghasilkan tabel baru yang serupa atau terbentuk tabel semula sebelum dilakukan penguraian (tabel awal).

b) *Functional Dependencies* (Ketergantungan Fungsional)

Ketergantungan fungsional merupakan suatu konsep dasar dalam normalisasi yang dapat menjelaskan hubungan antara atribut-atribut dalam suatu relasi (Kadir, 1999: 68). Lebih lanjut Kadir menguraikan bahwa secara lebih khusus ketergantungan fungsional dapat menjelaskan nilai pada suatu atribut yang dapat menentukan nilai dari atribut yang lain atau dapat pula diartikan jika “X” dan “Y” merupakan suatu atribut dalam sebuah relasi “R” dimana “Y” pada relasi “R” dapat dikatakan tergantung secara fungsional pada “X” atau nilai dari “X” dapat dikatakan secara fungsional menentukan nilai dari “Y” yang dinotasikan dengan $X \rightarrow Y$, jika dan hanya jika tiap nilai dari “X” di dalam relasi “R” dapat dihubungkan dengan tepat satu nilai “Y” di dalam relasi “R” dengan catatan bahwa “X” dan “Y” dapat terdiri dari satu atau lebih atribut (Connolly dan Begg, 2015: 457). Jadi, berdasarkan penjelasan tersebut dapat dikatakan bahwa “X” merupakan atribut penentu (*determinant*) sedangkan “Y” merupakan atribut yang bergantung (*dependent*). Selain itu, ketergantungan fungsional dapat pula disajikan dalam bentuk diagram seperti yang ditunjukkan pada gambar 2.17 di bawah ini.



Gambar 2.17 Diagram Ketergantungan Fungsional (Connolly dan Begg, 2015: 457)

Jadi, berdasarkan pembahasan yang telah diuraikan tersebut mengenai konsep ketergantungan fungsional maka dapat diketahui bahwa ketergantungan fungsional yang digunakan dalam tahap normalisasi memiliki ciri-ciri atau karakteristik sebagai berikut (Connolly dan Begg, 2015: 460):

1. Terdapat kardinalitas *relationship* 1:1 antara atribut penentu (*determinant*) dengan atribut tergantung (*dependent*) dengan catatan bahwa antar kedua atribut yang berlawanan tersebut kardinalitas *relationship* dapat bernilai *one to one* atau bahkan *one to many*.
 2. Atribut dalam relasi tersebut saling terkait antar satu sama lain.
 3. Atribut penentu memiliki jumlah atribut yang sedikit (minimal) atau tidak mengandung atribut yang tidak diperlukan, dengan kata lain harus ada ketergantungan fungsional secara penuh antara atribut yang bergantung dengan atribut penentu dalam suatu ketergantungan tersebut.
- Selain itu menurut Marlinda (2004: 119-121) yang mengungkapkan bahwa pada ketergantungan fungsional terdapat beberapa macam ketergantungan yang dapat digunakan dalam tahap normalisasi. Beberapa macam ketergantungan yang dimaksud

tersebut lebih lanjut dapat dipaparkan sebagai berikut:

1. Ketergantungan Fungsional Penuh (*Full Functional Dependency / FFD*)
Ketergantungan fungsional penuh merupakan suatu ketergantungan fungsional dimana atribut penentu (*determinant*) memiliki jumlah atribut yang sedikit (minimal) atau dengan kata lain tidak mengandung atribut yang tidak diperlukan guna untuk menghindari anomali yang mungkin akan terjadi antar atribut-atribut pada ketergantungan tersebut (Connolly dan Begg, 2015: 459). Lebih lanjut Connolly dan Begg

menjelaskan bahwa jika dimisalkan “X” dan “Y” merupakan suatu atribut dalam sebuah relasi “R” maka “Y” dapat dikatakan memiliki ketergantungan fungsional penuh terhadap “X”, jika “Y” memiliki ketergantungan fungsional terhadap “X” dan tidak memiliki ketergantungan terhadap bagian dari “X” (bila “X” merupakan atribut kunci gabungan). Selain itu, dapat dikatakan pula $X \rightarrow Y$ merupakan ketergantungan fungsional penuh jika salah satu atau beberapa dari atribut “X” (bila “X” merupakan atribut kunci gabungan) dihapus maka sifat ketergantungan “Y” terhadap “X” akan menghilang dengan kata lain “Y” harus bergantung secara penuh terhadap semua atribut kunci dari “X”.

2. Ketergantungan Fungsional Sebagian (*Partial Functional Dependency / PFD*)

Ketergantungan fungsional sebagian merupakan suatu ketergantungan yang terjadi apabila terdapat atribut bukan kunci dalam suatu bagian dari atribut penentu (*determinant*) dimana dimisalkan “X” dan “Y” merupakan suatu atribut dalam sebuah relasi “R” yang memiliki ketergantungan fungsional $X \rightarrow Y$ dan ketergantungan fungsional tersebut dikatakan memiliki ketergantungan sebagian jika salah satu dari atribut “X” dihapus maka “Y” masih memiliki ketergantungan terhadap “X” (Connolly dan Begg, 2015: 459).

3. Ketergantungan Fungsional Total (*Total Functional Dependency / TFD*)

Ketergantungan fungsional total merupakan suatu ketergantungan yang terjadi apabila antara atribut penentu dengan atribut tergantung saling bergantung satu sama lain atau dapat pula diartikan bahwa nilai dari atribut penentu dapat menentukan nilai dari atribut tergantung dan berlaku pula sebaliknya nilai dari atribut tergantung dapat menentukan nilai dari atribut penentu. Di sisi lain, apabila dimisalkan “X” dan “Y” merupakan suatu atribut dalam sebuah relasi “R” dan “Y” dapat dikatakan memiliki ketergantungan fungsional secara total terhadap “X” jika “Y” mempunyai ketergantungan fungsional terhadap “X” dan “X”

memiliki ketergantungan fungsional terhadap “Y” yang mana hal tersebut dapat dinyatakan dengan notasi:

$$X \leftrightarrow Y$$

4. Ketergantungan Transitif (*Transitive Dependency* / TD)

Ketergantungan fungsional transitif merupakan suatu ketergantungan yang terjadi apabila terdapat atribut bukan kunci yang memiliki ketergantungan fungsional terhadap atribut bukan kunci yang lain dalam suatu relasi. Selain itu, ketergantungan transitif dapat pula didefinisikan, apabila dimisalkan “X”, “Y”, dan “Z” merupakan suatu atribut dalam sebuah relasi “R” dan “Z” dapat dikatakan memiliki ketergantungan transitif terhadap “X” jika “Y” memiliki ketergantungan fungsional terhadap “X” dan “Z” memiliki ketergantungan fungsional terhadap “Y” atau dengan kata lain “Z” memiliki ketergantungan transitif terhadap “X” melalui “Y” dengan catatan bahwa “X” tidak memiliki ketergantungan fungsional terhadap “Y” dan “Z” (ketergantungan total) (Connolly dan Begg, 2015: 460).

5. Ketergantungan Bernilai Banyak (*Multi-Valued Dependency* / MVD)

Ketergantungan bernilai banyak atau dapat pula disebut *multi-valued dependency* pertama kali diperkenalkan oleh R. Fagin pada tahun 1977 yang digunakan pada proses normalisasi dalam bentuk normal keempat (4NF). *Multi-valued dependency* merupakan suatu ketergantungan fungsional dimana ketergantungan tersebut tidak sebatas hanya terdiri dari satu nilai saja tetapi dapat terdiri dari beberapa nilai (Singh, 2011: 351). Selain itu, *multi-valued dependency* dapat pula diartikan apabila “X”, “Y”, dan “Z” merupakan suatu atribut pada suatu relasi “R” dan nilai dari “Y” dapat dikatakan bersifat multidependen terhadap “X” atau nilai “X” dapat menentukan banyak nilai dari “Y” jika pada sekumpulan nilai “Y” yang diberikan pada gabungan atribut (X, Z) hanya tergantung pada nilai “X” dan tidak tergantung pada nilai “Z” atau dapat dikatakan pula nilai “X” dapat menentukan sekumpulan nilai dari “Y” dimana “Y” tidak memiliki ketergantungan terhadap nilai dari “Z” yang mana hal

tersebut dapat dinotasikan sebagai berikut (Kadir, 1999: 85-87; Singh, 2011: 351):

$$X \rightarrow Y$$

Selain kedua pendapat tersebut ada pula Connolly dan Begg (2015: 494) yang juga mengungkapkan bahwa definisi dari *multi-valued dependency* merupakan suatu ketergantungan yang terjadi antar atribut (misalkan “X”, “Y”, dan “Z” merupakan suatu atribut) dalam sebuah relasi “R” dimana setiap nilai dari “X” dapat memiliki hubungan dengan beberapa nilai dari “Y” dan dapat pula memiliki hubungan dengan beberapa nilai dari “Z”, tetapi nilai dari “Y” dan nilai dari “Z” tidak memiliki ketergantungan atau hubungan (*independent*) satu sama lain. Jadi, berdasarkan definisi di atas maka ketergantungan antar atribut dalam relasi “R” dapat dinotasikan sebagai berikut:

$$X \twoheadrightarrow Y$$

$$X \twoheadrightarrow Z$$

sehingga dari kedua ketergantungan tersebut maka *multi-valued dependency* dapat dinotasikan sebagai berikut:

$$X \twoheadrightarrow Y|Z$$

Secara umum, *multi-valued dependency* dapat muncul pada suatu relasi jika atribut yang dimiliki paling tidak berjumlah tiga atribut dengan dua diantara atribut tersebut merupakan atribut bernilai banyak dengan nilai-nilai yang bergantung hanya pada atribut ketiga sehingga *multi-valued dependency* memiliki kardinalitas dengan tipe *one to many* (1:M) atau hubungan satu ke banyak antara atribut penentu dengan atribut tergantung (Kadir, 1999: 85).

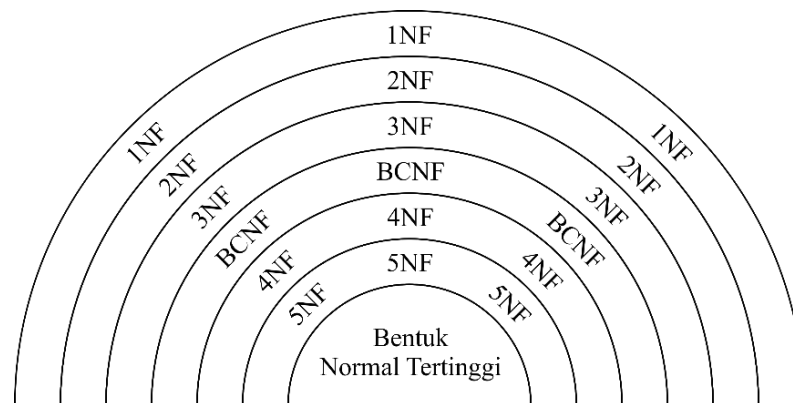
Pada hakikatnya, ketergantungan merupakan suatu konsep dasar yang sangat penting digunakan dalam proses normalisasi. Normalisasi merupakan suatu teknik yang digunakan untuk menjelaskan suatu relasi berdasarkan kunci utama (*candidate key*) serta ketergantungan fungsional yang dimiliki dalam relasi tersebut (Codd dalam Connolly dan Begg, 2015: 464). Lebih lanjut Codd menjelaskan bahwa teknik tersebut melibatkan serangkaian aturan atau

ketentuan yang digunakan untuk menganalisis suatu relasi dalam sebuah basis data sehingga basis data tersebut dapat dinormalisasi sampai tingkat tertentu. Apabila aturan atau ketentuan tersebut belum terpenuhi maka relasi yang belum memenuhi ketentuan harus didekomposisi hingga menjadi suatu relasi yang memenuhi ketentuan normalisasi. Adapun aturan-aturan atau ketentuan-ketentuan normalisasi yang dimaksud tersebut dapat dinyatakan dalam istilah bentuk normal (Kadir, 1999: 73). Pada proses normalisasi, bentuk normal dapat dilakukan secara bertahap atau memiliki tingkatan bentuk normal. Berikut tingkatan bentuk normal yang dimaksud dalam proses (tahapan) normalisasi:

- a) Bentuk normal pertama / *first normal form* (1NF).
- b) Bentuk normal kedua / *second normal form* (2NF).
- c) Bentuk normal ketiga / *third normal form* (3NF).
- d) Bentuk normal Boyce-Codd / *Boyce-Codd normal form* (BCNF).
- e) Bentuk normal keempat / *fourth normal form* (4NF).
- f) Bentuk normal kelima / *fifth normal form* (5NF).

Berdasarkan tingkatan bentuk normal tersebut, tiga bentuk normal paling awal, yakni bentuk normal pertama (1NF), bentuk normal kedua (2NF), dan bentuk normal ketiga (3NF) merupakan bentuk normal yang paling umum digunakan dalam proses normalisasi (Kadir, 1999: 73). Dengan kata lain dalam relasi tersebut sudah tidak terdapat lagi anomali yang menjadi masalah dalam sebuah basis data sehingga proses normalisasi tidak perlu dilanjutkan pada tingkatan yang lebih tinggi lagi. Akan tetapi, dalam kejadian lain apabila proses normalisasi pada bentuk normal ketiga (3NF) masih terdapat suatu anomali maka normalisasi perlu dilakukan pada tingkatan yang lebih tinggi lagi, yakni dalam bentuk normal Boyce-Codd (BCNF). Bentuk normal Boyce-Codd (BCNF) merupakan bentuk normal yang dikemukakan oleh R. Boyce dan E. F. Codd dimana digunakan sebagai perbaikan terhadap bentuk normal ketiga (3NF) (Connolly dan Begg, 2015: 464). Selain itu, adapun bentuk normal yang memiliki tahapan yang lebih tinggi dari bentuk normal Boyce-Codd, yakni bentuk normal keempat (4NF) dan bentuk normal kelima (5NF). Namun, kedua bentuk normal tersebut jarang terjadi dan hanya digunakan pada keadaan-

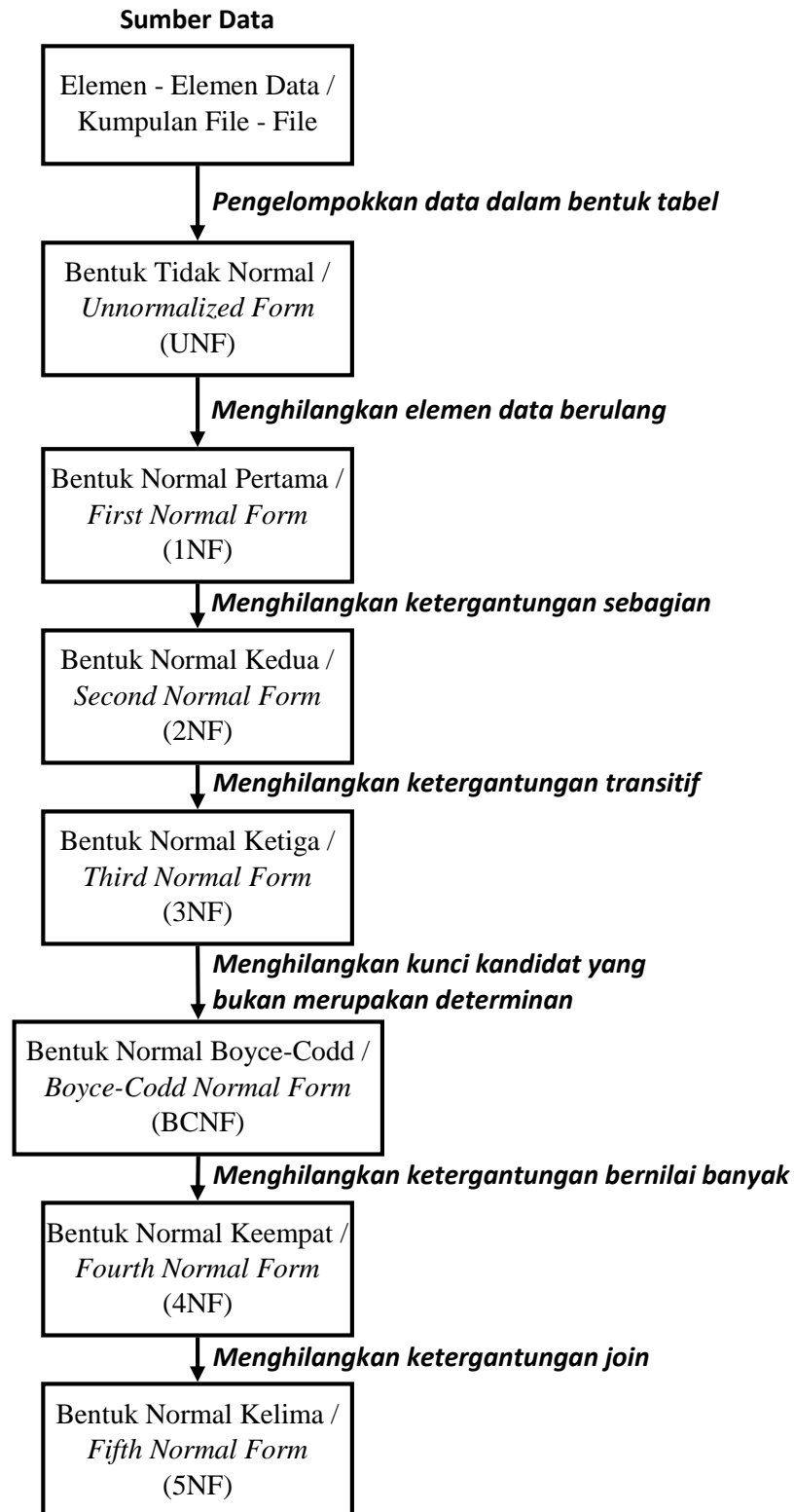
keadaan tertentu, yakni pada relasi yang mengandung *multi-valued dependency* (Kadir, 1999: 73). Sementara itu, hubungan antar bentuk normal yang telah diuraikan tersebut dapat direpresentasikan dalam bentuk diagram seperti yang ditunjukkan oleh gambar 2.18 di bawah, sedangkan untuk tahapan-tahapan yang dilakukan dalam proses normalisasi data tersebut dapat direpresentasikan dalam sebuah diagram seperti yang ditunjukkan pada gambar 2.19 di bawah.



Gambar 2.18 Diagram Hubungan antar Bentuk Normal

(Connolly dan Begg, 2015: 465)

Diagram 2.18 di atas yang merupakan diagram hubungan antar bentuk normal menjelaskan bahwa setiap bentuk normal dalam proses normalisasi tersebut bergantung pada bentuk normal pada tingkat sebelumnya. Sebagai contoh dalam hal tersebut ialah apabila telah mencapai bentuk normal kedua dalam proses normalisasi tersebut maka pasti telah memenuhi ketentuan dalam bentuk normal pertama, begitu pula apabila telah mencapai bentuk normal ketiga maka proses normalisasi pasti telah memenuhi bentuk normal kedua dan seterusnya hingga mencapai bentuk normal yang lebih sederhana dan efisien sesuai dengan kriteria dan ketentuan tertentu. Selain itu, berdasarkan diagram tersebut yang lebih lanjut dapat menjelaskan bahwa semakin tinggi tahapan dalam proses normalisasi maka suatu relasi akan mempunyai kecenderungan yang lebih baik dan efisien atau dalam artian memiliki anomali yang lebih sedikit (minimal anomali) atau bahkan tidak memiliki anomali sama sekali pada suatu relasi tersebut.



Gambar 2.19 Diagram Proses Normalisasi
(Connolly dan Begg, 2015: 465)

Selain itu, terdapat pula diagram proses normalisasi yang ditunjukkan oleh gambar 2.19 di atas dimana diagram tersebut dapat menjelaskan tahapan-tahapan atau tingkatan-tingkatan bentuk normal yang terjadi dalam proses normalisasi. Pada tiap tingkatan tersebut terdapat suatu ketentuan atau aturan yang harus dipenuhi untuk mencapai tingkatan yang lebih tinggi dari tingkatan sebelumnya, seperti bentuk normal pertama yang memiliki ketentuan harus menghilangkan elemen data yang berulang untuk mencapai bentuk normal kedua dan seterusnya hingga diperoleh bentuk normal yang terhindar dari anomali dan ketergantungan fungsional. Sementara itu, ketentuan-ketentuan pada setiap bentuk normal dalam proses normalisasi tersebut lebih lanjut dapat dipaparkan sebagai berikut:

a) Bentuk Tidak Normal / *Unnormalized Form* (UNF)

Suatu relasi dapat dikatakan memiliki bentuk tidak normal atau dapat pula disebut sebagai bentuk yang belum ternormalisasi (*unnormalized form*) apabila dalam relasi tersebut terdapat satu atau lebih sekumpulan data yang berulang (*repeating group*) (Connolly dan Begg, 2015: 466). Pada tahap ini terjadi proses pengambilan informasi data dari sumber data yang dikumpulkan dan diubah atau dikelompokkan menjadi suatu relasi dimana dalam relasi tersebut dapat mengandung atribut komposit, yakni dapat memiliki satu atau lebih nilai (*multiple-value*) pada atribut kunci yang mana nilai tersebut masih dapat dibagi lagi menjadi nilai yang lebih sederhana (*non-atomic*) dalam bentuk atribut baru.

b) Bentuk Normal Pertama / *First Normal Form* (1NF)

Suatu relasi dapat dikatakan memiliki bentuk normal pertama apabila dalam relasi tersebut hanya memiliki paling banyak satu nilai untuk setiap atribut atau perpotongan antara baris dan kolom (atribut bernilai tunggal) (Connolly dan Begg, 2015: 466). Pada tahap ini terjadi perubahan relasi dalam bentuk tidak normal (*unnormalized form*) menjadi relasi dalam bentuk normal pertama (*first normal form*) dengan cara menghilangkan sekelompok data yang berulang. Adapun cara yang dapat dilakukan untuk menghilangkan kelompok data berulang pada suatu relasi yang memiliki

bentuk tidak normal, ialah dengan suatu pendekatan umum. Suatu pendekatan umum yang dimaksud tersebut dapat terdiri atas 2 (dua) pendekatan umum yang mana dapat dipaparkan sebagai berikut:

1. Dengan memasukkan data yang sesuai ke dalam kolom kosong dari baris yang mengandung data berulang. Dengan kata lain, kolom kosong tersebut dapat diisi dengan cara menggandakan data yang tidak berulang jika hal tersebut memang diperlukan. Adapun pendekatan ini sering disebut sebagai perataan suatu relasi (*flattening the tabel*).
2. Dengan menempatkan data berulang bersamaan dengan salinan dari atribut kunci yang asli pada relasi lain. Dengan kata lain, memisahkan atribut yang memiliki data berulang berdasarkan atribut kunci dari relasi utama ke dalam relasi baru/lain. Terkadang relasi yang memiliki bentuk normal tersebut dapat memiliki lebih dari satu data berulang. Dalam kasus seperti tersebut maka proses pada pendekatan ini dapat dilakukan berulang kali hingga tidak terdapat lagi data berulang pada suatu relasi dalam proses tersebut. Apabila tidak terdapat data berulang lagi maka dapat dikatakan relasi sudah dalam bentuk normal pertama.

Berdasarkan dua pendekatan yang telah diuraikan di atas maka dapat diketahui bahwa dari pendekatan pertama tersebut dihasilkan suatu relasi yang tidak mengandung atribut komposit yang di dalam atribut tersebut terdapat nilai yang bersifat bukan atomik (*non-atomic*), yakni suatu nilai yang masih dapat dibagi lagi menjadi komponen nilai yang lebih sederhana serta tidak terdapat pula atribut bernilai banyak yang mana di dalam atribut tersebut mengandung nilai berganda dengan jenis yang sama atau dapat dikatakan pula menghasilkan suatu relasi yang mengandung nilai yang bersifat atomik, yakni suatu nilai yang tidak dapat dibagi lagi menjadi komponen nilai yang lebih sederhana pada tiap perpotongan antara baris dan kolom dan mengandung nilai tunggal, yakni hanya memiliki paling banyak satu nilai untuk setiap perpotongan antara baris dan kolom pada suatu relasi tersebut. Sedangkan pada pendekatan kedua dihasilkan suatu relasi dimana atribut kunci sudah terdefinisikan yang mana digunakan untuk

mengidentifikasi ketergantungan fungsional dalam suatu relasi. Selain itu, setiap kolom pada suatu relasi tersebut hanya memiliki satu pengertian dimana hal tersebut merupakan hasil dari pendekatan pertama yang dikenal dengan istilah *flattening the table* serta atribut kunci dalam relasi sudah dapat didefinisikan yang mana hal tersebut merupakan hasil dari pendekatan kedua. Jadi, dapat disimpulkan berdasarkan uraian di atas bahwa bentuk normal pertama tersebut memiliki ciri-ciri atau ketentuan-ketentuan sebagai berikut:

1. Tidak terdapat sekelompok data yang berulang (*repeating group of data*).
2. Tidak mengandung nilai yang bersifat *non-atomic* (atribut komposit) atau dapat dikatakan terdapat nilai yang bersifat atomik (atribut sederhana) dalam suatu relasi.
3. Tidak mengandung nilai berganda (atribut bernilai banyak) atau dapat dikatakan terdapat nilai tunggal (atribut tunggal) dalam suatu relasi.
4. Atribut kunci telah didefinisikan.

Akan tetapi, proses normalisasi dalam bentuk normal pertama tersebut masih terdapat beberapa ketidakkonsistenan data dikarenakan adanya redundansi serta ketergantungan fungsional yang dapat menghasilkan anomali-anomali pada suatu relasi, baik anomali penyisipan, penghapusan, ataupun perubahan. Oleh sebab itu, untuk menghilangkan beberapa anomali pada relasi tersebut maka perlu dilakukan proses normalisasi bentuk normal kedua.

c) Bentuk Normal Kedua / *Second Normal Form* (2NF)

Suatu relasi dapat dikatakan memiliki bentuk normal kedua apabila telah memenuhi ketentuan bentuk normal pertama dan setiap atribut bukan kunci (*non-key*) harus memiliki ketergantungan fungsional secara penuh terhadap atribut kunci utama (*primary key*) (Connolly dan Begg, 2015: 470). Dalam hal ini terjadi proses normalisasi dari bentuk normal pertama ke bentuk normal kedua dengan ketentuan harus menghilangkan ketergantungan sebagian antar atribut yang saling bergantung dalam suatu relasi, yakni

dengan cara dekomposisi. Adapun proses dekomposisi yang dilakukan untuk menghilangkan ketergantungan sebagian antar atribut dalam suatu relasi tersebut ialah dengan menempatkan suatu atribut bukan kunci (*non-key*) yang memiliki ketergantungan sebagian terhadap atribut penentu (*primary key*) ke dalam relasi baru/lain bersamaan dengan salinan dari atribut penentu (*primary key*) sehingga atribut bukan kunci tersebut dapat bergantung secara penuh terhadap atribut penentunya (*primary key*). Namun, apabila masih terdapat suatu ketergantungan sebagian dalam relasi tersebut maka proses dekomposisi dilakukan kembali sampai tidak ditemukan lagi suatu atribut bukan kunci yang memiliki ketergantungan sebagian terhadap atribut penentu (*primary key*) sehingga proses normalisasi telah memenuhi ketentuan bentuk normal kedua. Akan tetapi, relasi dalam bentuk normal kedua tersebut masih terdapat beberapa masalah seperti yang terjadi pada relasi dalam bentuk normal pertama, yakni adanya ketidakkonsistenan data. Ketidakkonsistenan data pada relasi dalam bentuk normal kedua dapat terjadi karena dalam suatu relasi masih terdapat suatu kerangkapan data (*redundancy data*) dan ketergantungan fungsional terutama ketergantungan fungsional transitif yang dapat menghasilkan anomali-anomali seperti anomali penyisipan, anomali penghapusan, serta anomali perubahan (Hall, 2007: 64). Oleh sebab itu, untuk meminimalkan atau bahkan menghilangkan anomali pada relasi dalam bentuk normal kedua maka diperlukan proses normalisasi dengan tahapan yang lebih tinggi, yakni normalisasi dalam bentuk normal ketiga.

d) Bentuk Normal Ketiga / *Third Normal Form* (3NF)

Suatu relasi dapat dikatakan berada dalam bentuk normal ketiga apabila telah memenuhi ketentuan bentuk normal pertama dan ketentuan bentuk normal kedua serta setiap atribut yang bukan kunci (*primary key*) tidak memiliki ketergantungan transitif terhadap atribut kunci utama (*primary key*) (Connolly dan Begg, 2015: 472). Pada tahap ini terjadi proses normalisasi dalam bentuk normal kedua ke dalam bentuk normal ketiga dengan ketentuan harus menghilangkan ketergantungan fungsional secara

transitif pada suatu relasi. Adapun cara yang dilakukan untuk menghilangkan ketergantungan fungsional secara transitif ialah dengan dekomposisi terhadap suatu relasi. Proses dekomposisi dilakukan dengan menempatkan suatu atribut bukan kunci yang memiliki ketergantungan fungsional terhadap atribut bukan kunci yang lain (*non-primary key*) ke dalam relasi baru/lain bersamaan dengan salinan dari atribut penentu (*primary key*) sehingga relasi pada proses normalisasi tersebut telah memenuhi ketentuan bentuk normal ketiga atau dapat dikatakan telah berada dalam bentuk normal ketiga.

Secara umum proses normalisasi yang dilakukan hanya sampai pada tahap bentuk normal ketiga. Hal tersebut dapat terjadi karena pada proses normalisasi sampai tahap bentuk normal ketiga menghasilkan suatu relasi yang memiliki sedikit kerangkapan data (redundansi data yang minimal) serta tidak terdapat lagi anomali pada relasi tersebut. Akan tetapi, tidak menutup kemungkinan pada relasi dalam bentuk normal ketiga masih terdapat suatu anomali yang dapat menyebabkan ketidakkonsistenan data. Oleh karena itu, perlu dilakukan proses normalisasi dengan tingkat yang lebih tinggi dari bentuk normal ketiga, yakni bentuk normal Boyce-Codd (Codd dalam Connolly dan Begg, 2015: 485). telah memenuhi ketentuan relasi dimana tidak terdapat lagi kerangkapan data (*redundancy data*) serta anomali yang terjadi pada suatu relasi tersebut.

e) Bentuk Normal Boyce-Codd / *Boyce-Codd Normal Form* (BCNF)

Bentuk normal Boyce-Codd merupakan salah satu bentuk normal pada proses normalisasi sebagai perbaikan dari bentuk normal ketiga (Kadir, 1999: 83). Lebih lanjut Kadir menjelaskan bahwa suatu relasi yang telah memenuhi ketentuan bentuk normal Boyce-Codd selalu memenuhi ketentuan bentuk normal ketiga, tetapi tidak berlaku sebaliknya. Suatu relasi yang telah memenuhi ketentuan bentuk normal ketiga belum tentu memenuhi ketentuan bentuk normal Boyce-Codd. Adapun perbedaan antara bentuk normal ketiga dengan bentuk normal Boyce-Codd ialah apabila diberikan ketergantungan fungsional $X \rightarrow Y$ maka pada bentuk normal ketiga, "X" dapat berupa atribut bukan kunci sedangkan pada bentuk normal

Boyce-Codd, “X” harus berasal dari atribut kunci kandidat (*candidate key*) (Connolly dan Begg, 2015: 485). Sementara itu, suatu relasi dapat dikatakan berada dalam bentuk normal Boyce-Codd jika dan hanya jika setiap atribut penentu merupakan atribut yang berasal dari atribut kunci kandidat (Connolly dan Begg, 2015: 485). Pada tahap ini terjadi proses normalisasi suatu relasi ke dalam bentuk normal Boyce-Codd dengan ketentuan harus menghilangkan atribut kunci kandidat yang bukan merupakan atribut penentu. Adapun cara yang dapat dilakukan untuk memenuhi ketentuan dalam bentuk normal Boyce-Codd ialah dengan mendekomposisi suatu relasi. Proses dekomposisi dilakukan dengan menempatkan atribut penentu yang bukan merupakan atribut kunci kandidat ke dalam relasi baru bersamaan dengan ketergantungan fungsional atau atribut tergantung sehingga atribut penentu tersebut dapat menjadi atribut kunci kandidat dalam relasi baru dari hasil dekomposisi tersebut sedangkan pada relasi utama atribut tersebut dapat berupa atribut tergantung. Akan tetapi, meskipun bentuk normal Boyce-Codd menghasilkan suatu relasi dengan anomali yang sedikit (*minimal*) yang disebabkan karena terdapat suatu ketergantungan fungsional dalam relasi tersebut dimana penelitian lebih lanjut mengungkapkan bahwa terdapat masalah ketergantungan jenis lain yang dapat menyebabkan terjadi redundansi data dalam suatu relasi, yakni ketergantungan bernilai banyak (*multi-valued dependency*) (Fagin dalam Connolly dan Begg, 2015: 493). Oleh karena itu, perlu dilakukan proses normalisasi pada tahap bentuk normal keempat untuk menghilangkan ketergantungan bernilai banyak sehingga tidak terdapat lagi masalah kerangkapan data dalam suatu relasi.

f) Bentuk Normal Keempat / *Fourth Normal Form* (4NF)

Suatu relasi dapat dikatakan berada dalam bentuk normal keempat jika dan hanya jika telah memenuhi bentuk normal Boyce-Codd dan tidak mengandung dua atau lebih atribut yang bernilai banyak atau dapat pula dikatakan setiap atribut tergantung tidak memiliki ketergantungan bernilai banyak terhadap atribut penentu dalam suatu relasi dimana atribut penentu

tersebut merupakan suatu atribut kunci kandidat (Kadir, 1999: 88; Connolly dan Begg, 2015: 495). Pada tahap ini terjadi proses normalisasi dari bentuk normal Boyce-Codd ke dalam bentuk normal keempat dengan ketentuan harus menghilangkan ketergantungan bernilai banyak (*multi-valued dependency*) dalam suatu relasi. Adapun cara untuk menghilangkan ketergantungan tersebut ialah dengan pendekomposisian suatu relasi. Proses dekomposisi dilakukan dengan menempatkan suatu atribut tergantung yang memiliki ketergantungan bernilai banyak ke dalam relasi baru bersamaan dengan salinan dari atribut penentu.

g) Bentuk Normal Kelima / *Fifth Normal Form* (5NF)

Setiap terjadi suatu penguraian terhadap relasi menjadi dua relasi yang sederhana maka relasi yang dihasilkan tersebut harus memiliki sifat *lossless-join* (Connolly dan Begg, 2015: 496). Sifat ini mengacu pada kemampuan suatu relasi untuk dapat bergabung kembali menjadi relasi asli/semula (dalam bentuk awal tanpa kehilangan data/informasi pada relasi tersebut). Namun, terdapat beberapa kejadian lain dimana dalam suatu relasi pada kasus tertentu dapat diuraikan menjadi lebih dari dua relasi sederhana. Adapun hal tersebut dapat diselesaikan pada normalisasi dalam bentuk normal kelima. Bentuk normal kelima atau *fifth normal form* (5NF) merupakan suatu proses normalisasi yang memiliki bentuk normal tertinggi daripada bentuk normal keempat dimana bentuk normal kelima tersebut memiliki keterkaitan atau berkenaan dengan ketergantungan gabungan (*join dependency*) antar tabel dalam suatu relasi (Yuhfizard, 2003: 44). *Join dependency* atau ketergantungan gabungan merupakan suatu cara yang digunakan untuk dapat menggambarkan tipe dari ketergantungan dimana jika terdapat sebuah relasi $R(X, Y, \dots, Z)$ dengan himpunan bagian dari atribut pada relasi R tersebut dapat dinotasikan dengan A, B, \dots, C , maka relasi R dapat dikatakan memiliki ketergantungan gabungan jika dan hanya jika setiap nilai yang berlaku pada relasi $R(X, Y, \dots, Z)$ sama dengan gabungan proyeksi dari A, B, \dots, C (Kadir, 1999: 88; Connolly dan Begg,

2015: 497). Adapun definisi dari ketergantungan gabungan di atas dapat dinyatakan dengan notasi:

$$*(A, B, \dots, C)$$

dengan

$$A = XY, B = YZ, C = ZX$$

Dengan kata lain, gabungan proyeksi dari (A, B, \dots, C) akan menghasilkan sebuah relasi R dengan nilai dari atribut-atribut yang dimiliki ialah (X, Y, \dots, Z) dimana dalam hal ini dapat dinotasikan pula sebagai berikut:

$$*(A, B, \dots, C) = R(X, Y, \dots, Z)$$

Suatu relasi dapat dikatakan berada dalam bentuk normal kelima (5NF) atau dapat pula disebut sebagai *projection-join normal form* (PJ/NF) jika dan hanya jika setiap ketergantungan gabungan nontrivial dalam R tersirat oleh kunci kandidat dari relasi R (asli) (Date, 2004: 389), dimana:

- a. Ketergantungan gabungan yang dinotasikan $*(A, B, \dots, Z)$ pada relasi R dikatakan trivial jika dan hanya jika setidaknya terdapat satu dari (A, B, \dots, Z) yang merupakan himpunan semua dari atribut R .
- b. Ketergantungan gabungan yang dinotasikan $*(A, B, \dots, Z)$ pada relasi R dapat dikatakan tersirat oleh kunci kandidat R jika dan hanya jika setiap proyeksi (A, B, \dots, Z) mencakup kunci kandidat dari relasi asli.

Pada proses normalisasi, apabila suatu relasi telah mencapai bentuk normal kelima maka secara tidak langsung telah mencapai bentuk normal keempat yang tidak memiliki ketergantungan bernilai banyak (*multi-valued dependency*) (Date, 2004: 390). Jadi, untuk dapat mencapai bentuk normal kelima maka suatu relasi harus berada dalam bentuk normal keempat serta dalam relasi tersebut terhindar dari ketergantungan gabungan nontrivial yang mana proyeksi dalam ketergantungan gabungan tersebut tidak memiliki kunci kandidat dari relasi asli sehingga dapat menyebabkan adanya informasi/data tambahan pada relasi gabungan baru yang berbeda dengan relasi asli (semula) (Connolly dan Begg, 2015: 496). Pada proses normalisasi bentuk normal kelima ini terjadi pendekomposisi terhadap suatu relasi menjadi beberapa relasi yang lebih sederhana sehingga terhindar

dari ketergantungan gabungan nontrivial jika terjadi penggabungan kembali terhadap relasi-relasi hasil dari dekomposisi tersebut atau dapat pula dikatakan tidak terdapat ketergantungan gabungan lagi pada relasi-relasi hasil dekomposisi tersebut (Date, 2004: 390).

BAB 5

KESIMPULAN

5.1. Kesimpulan

Berdasarkan hasil dan pembahasan yang telah dipaparkan sebelumnya maka dapat disimpulkan bahwa:

1. Rancang bangun program aplikasi yang dikembangkan dalam penelitian ini merupakan rancang bangun program aplikasi berbasis desktop yang dibuat menggunakan program aplikasi *Visual Basic 2010* serta basis data yang digunakan adalah *MariaDB* yang berjalan menggunakan perangkat lunak berbasis *server offline* bernama *XAMPP* dimana program aplikasi yang dirancang dan dikembangkan tersebut digunakan untuk menentukan jumlah keluarga kurang mampu yang mendapat bantuan beasiswa pendidikan menggunakan metode *simple additive weighting (SAW)*.
2. Implementasi sistem pendukung keputusan ke dalam bahasa pemrograman *Visual Basic 2010* untuk menentukan keluarga kurang mampu yang mendapat bantuan beasiswa pendidikan dengan metode perhitungan *simple additive weighting (SAW)* yang merupakan metode penjumlahan terbobot dari rating kinerja pada setiap alternatif terbaik dengan kriteria-kriteria yang telah ditentukan sebelumnya sehingga diperoleh data-data jumlah keluarga kurang mampu.

5.2. Saran

Adapun saran-saran yang dapat diberikan dalam penelitian ini dapat dipaparkan sebagai berikut.

1. Program aplikasi yang telah dibuat tersebut perlu ada kajian lebih lanjut dari segi koding yang masih memuat banyak ruang.

2. Tampilan program aplikasi yang masih sederhana untuk dapat dikembangkan lebih lanjut dengan tampilan grafis yang lebih menarik.
3. Program aplikasi yang dibuat perlu adanya pengembangan untuk dapat dibuat menjadi program aplikasi berbasis Android sehingga memudahkan pengguna untuk dapat menginputkan data tanpa harus meluangkan waktu membuka dan menhidupkan komputer.

DAFTAR PUSTAKA

- Ala, Andre Bayo. 1981. *Kemiskinan dan Strategi Memerangi Kemiskinan*. Yogyakarta: Liberty.
- Anhar. 2010. *Panduan Menguasai PHP dan MySQL Secara Otodidak*. Jakarta: Mediakita.
- Apache Friends. 2017. "About". Apache Friends, 2017. <https://www.apachefriends.org/about.html> (28 Agt. 2017).
- Ariyanti, Fiki. 2015. "Strategi Pemerintah Jokowi Kurangi Tingkat Kemiskinan". Liputan6, 22 Juni 2015. <http://bisnis.liputan6.com/read/2256860/strategi-pemerintah-jokowi-kurangi-tingkat-kemiskinan> (20 Sept. 2016).
- Badan Perencanaan Pembangunan Nasional. 2010. *Evaluasi Pelayanan Keluarga Berencana (KB) Bagi Masyarakat Miskin (Keluarga Prasejahtera/KPS dan Keluarga Sejahtera-I/KS-I)*. Jakarta: Direktorat Kependudukan, Pemberdayaan Perempuan, dan Perlindungan Anak, Badan Perencanaan Pembangunan Nasional.
- Badan Pusat Statistik. 2008. *Analisis dan Penghitungan Tingkat Kemiskinan Tahun 2008*. Jakarta: Badan Pusat Statistik.
- Badan Pusat Statistik. 2015. *Data dan Informasi Kemiskinan Kabupaten/Kota 2014*. Jakarta: Nario Sari.
- Badan Pusat Statistik. 2015. *Pemutakhiran Basis Data Terpadu 2015: Pedoman Kepala BPS Provinsi/Kabupaten/Kota*. Jakarta: Badan Pusat Statistik.
- Badan Pusat Statistik. 2015. *Statistik Indonesia 2015*. Jakarta: Badan Pusat Statistik.
- Bagui, Sikha dan Richard Earp. 2012. *Database Design using Entity-Relationship Diagrams*. Edisi ke-2. Florida: CRC Press.

- Bartholomew, Daniel. 2012. *MariaDB vs MySQL*. Dokumen disajikan untuk membandingkan perbedaan antara MariaDB dan MySQL, Birmingham, September 2012.
- Binarto, Suryo. 2012. *Tip dan Trik Membuat Program Penjualan Menggunakan Visual Basic 6.0*. Jakarta: Mediakita.
- Black, Paul dan Tom Stockton. 2009. "Basic Step for the Development of Decision Support Systems". Dalam Marcomini, Antonio (Eds.). *Decision Support Systems for Risk-Based Management of Contaminated Sites*. Hlm. 1-27. New York: Springer Science dan Business Media.
- Bradley, Julia Case dan Anita C. Millspaugh. 2010. *Programming in Visual Basic 2010*. New York: McGraw-Hill Education.
- Bradshaw, Ted K. 2006. "Theories of Poverty and Anti-Poverty Programs in Community Development". *Rural Poverty Research Center Working Paper*. Februari 2006. No. 06-05:1-22. California: University of California.
- Buana, Komang Setia. 2014. *Jago Pemrograman PHP untuk Pemula dan Orang Awam*. Jakarta: Dunia Komputer.
- Bunafit, Nugroho. 2005. *Database Relasional dengan MySQL*. Yogyakarta: ANDI.
- Chambers, Robert. 1988. *Pembangunan Desa Mulai Dari Belakang*. Jakarta: LP3ES.
- Connolly, Thomas dan Carolyn Begg. 2010. *Database Systems: A Practical Approach to Design, Implementation, and Management*. Edisi ke-5. London: Pearson Education.
- Connolly, Thomas dan Carolyn Begg. 2015. *Database Systems: A Practical Approach to Design, Implementation, and Management*. Edisi ke-6. London: Pearson Education.
- Daihani, Dadan Umar. 2001. *Sistem Pendukung Keputusan*. Jakarta: Elex Media Komputindo.

- Date, C. J.. 2004. *An Introduction to Database Systems*. Edisi ke-8. London: Pearson Education.
- Elistri, Melisa, et al. 2014. “Penerapan Metode SAW dalam Sistem Pendukung Keputusan Pemilihan Jurusan pada Sekolah Menengah Atas Negeri 8 Seluma”. *Jurnal Media Infotama*. September 2014. Vol. 10 No. 2:105-109. Bengkulu: Universitas Dehasen.
- Eniyati, Sri. 2011. “Perancangan Sistem Pendukung Pengambilan Keputusan untuk Penerimaan Beasiswa dengan Metode SAW (*Simple Additive Weighting*)”. *Jurnal Teknologi Informasi DINAMIK*. Juli 2011. Vol. 16 No. 2:171-177. Semarang: Universitas Stikubank.
- Faisal, M. Reza. 2008. *Seri Belajar ASP.NET: Membangun Aplikasi Web Mudah dan Cepat*. Jakarta: Indonesia Net Developer Community.
- Fathansyah. 2004. *Buku Teks Komputer: Sistem Basis Data*. Edisi-Lanjutan Buku Basisdata. Yogyakarta: Informatika.
- Fathansyah. 2012. *Basis Data*. Bandung: Informatika.
- Firdaus. 2007. *PHP dan MySQL dengan DREAMWEAVER*. Palembang: Maxikom.
- Firdausa, et al. 2016. “Model Sistem Pendukung Keputusan Pemilihan Sekolah Menggunakan Metode SAW”. Seminar Nasional Teknologi Informasi dan Multimedia 2016. Februari 2016. Hal:1-6. Malang: Universitas Negeri Malang.
- Hall, James A.. 2007. *Sistem Informasi Akuntansi*. Buku ke-2. Edisi ke-4. Terjemahan Dewi Fitriyani dan Deny Kwary Arnos. Jakarta: Salemba Empat.
- Hariyanto, Bambang. 2004. *Sistem Manajemen Basis Data*. Bandung: Informatika.
- Hidayatullah, Priyanto. 2012. *Visual Basic .Net*. Bandung: Informatika.
- Hutahaean, Jeperson. 2014. *Konsep Sistem Informasi*. Yogyakarta: Deepublish.

- IMK. 2014. "Jurus Jitu SBY Mengentaskan Kemiskinan". Detik, 16 Oktober 2014. <https://news.detik.com/berita/2720289/jurus-jitu-sby-mengentaskan-kemiskinan> (29 Sept. 2016).
- Ismail, Zarmawis. 1999. *Masalah Kemiskinan Masyarakat Perkampungan Kumuh di Perkotaan: Kasus Yogyakarta dan Surabaya*. Jakarta: Puslitbang Ekonomi dan Pembangunan, LIPI.
- Jaelani, Dani dan Rachmi Agustiyani. 2011. *Ketenagakerjaan Penduduk Indonesia: Hasil Sensus Penduduk 2010*. Jakarta-Indonesia: Badan Pusat Statistik.
- Jarnasy, Owin. 2004. *Keadilan, Pemberdayaan dan Penanggulangan Kemiskinaan*. Jakarta: Belantika.
- Jogiyanto HM. 2005. *Analisis dan Desain Sistem Informasi: pendekatan terstruktur teori dan praktik aplikasi bisnis*. Edisi ke-3. Yogyakarta: ANDI.
- Julianti, Eka. 2011. *Rancang Bangun Sistem Pendukung Keputusan Dalam Menentukan Peserta Asuransi Rumahkoe Syariah Menggunakan Fuzzy MADM Model Yager*. Skripsi. Universitas Islam Negeri Syarif Hidayatullah, Jakarta.
- K.C., Kiran. 2014. *Developing a dynamic website using WordPress CMS for BPS Finland Oy*. Skripsi. Laurea University of Applied Sciences, Leppävaara.
- Kadir, Abdul. 1999. *Konsep Tuntunan Praktis Basis Data*. Yogyakarta: ANDI.
- Kadir, Abdul. 2008. *Dasar Pemrograman Web Dinamis menggunakan PHP*. Yogyakarta: ANDI.
- Kadir, Abdul. 2009. *Membuat Aplikasi Web dengan PHP + Database MySQL*. Yogyakarta: ANDI.
- Kadji, Yulianto. 2013. "Kemiskinan dan Konsep Teoritisnya". November 2013. Hal:1-7. Gorontalo: Universitas Negeri Gorontalo.

- Kartasasmita, Ginandjar. 1996. *Pembangunan Untuk Rakyat: Memadukan Pertumbuhan dan Pemerataan*. Jakarta: CIDESINDO.
- Katre, Dhananjay. 2003. *Migrating to .NET: A Pragmatic Path to Visual Basic .NET, Visual C++ .NET, and ASP.NET*. New Jersey: Prentice Hall.
- Kemendikbud, Pengelola Web. 2016. "Target Kemendikbud Dalam Pengembangan Pendidikan dan Kebudayaan Tahun 2017". Kemendikbud, 02 September 2016. <https://www.kemdikbud.go.id/main/blog/2016/09/target-kemendikbud-dalam-pengembangan-pendidikan-dan-kebudayaan-tahun-2017> (26 Sept. 2018).
- Kementerian Sosial Republik Indonesia. 2012. *Analisis Kemiskinan: Berdasarkan Data Pendataan Program Perlindungan Sosial (PPLS) 2011*. Jakarta: Kementerian Sosial Republik Indonesia.
- Komite Penanggulangan Kemiskinan. 2004. *Strategi Nasional Penanggulangan Kemiskinan*. Jakarta: Komite Penanggulangan Kemiskinan.
- Krismiaji. 2010. *Sistem Informasi Akuntansi*. Edisi ke-3. Yogyakarta: UPP-STIM YKPN.
- Kristianto, Andri. 2008. *Perancangan Sistem Informasi dan Aplikasinya*. Yogyakarta: Gava Media.
- Kristianto, Harianto. 1994. *Konsep dan Perancangan Database*. Yogyakarta: ANDI.
- Kuncoro, Mudrajad. 2000. *Ekonomi Pembangunan: Teori, Masalah, dan kebijakan*. Yogyakarta: Akademi Manajemen Perusahaan YKPN.
- Kuntani, Daniel Wilhenson, et al. "Implementasi dan Evaluasi Private Cloud di Laboratorium Komputer". *Jurnal INFRA*. April 2015. Vol. 3 No. 2:128-134. Surabaya: Universitas Kristen Petra.
- Kurniawan, Erick. 2010. *Cepat Mahir ASP.NET 3.5 untuk Aplikasi Web Interaktif*. Yogyakarta: ANDI.

- Kusrini. 2007. *Konsep dan Aplikasi Sistem Pendukung Keputusan*. Yogyakarta: ANDI.
- Kusrini. 2007. *Strategi Perancangan dan Pengelolaan Basis Data*. Yogyakarta: ANDI.
- Kusumadewi, Sri dan Hari Purnomo. 2010. *Aplikasi Logika Fuzzy: Untuk Pendukung Keputusan*. Edisi ke-2. Yogyakarta: Graha Ilmu.
- Kusumadewi, Sri, et al. 2006. *Fuzzy Multi-Attribute Decision Making (Fuzzy MADM)*. Yogyakarta: Graha Ilmu.
- Kusumo, Ario Suryo. 2004. *Buku Latihan: Visual Basic .Net versi 2002 dan 2003*. Jakarta: Elex Media Komputindo.
- Ladjamudin, Al-Bahra bin. 2005. *Analisis dan Desain Sistem Informasi*. Yogyakarta: Graha Ilmu.
- Laudon, Kenneth C., dan Jane P. Laudon. 2012. *Management Information Systems: Managing the Digital Firm*. Edisi ke-12. New Jersey: Prentice Hall.
- Lembaga Penelitian SMERU. 2001. *Paket Informasi Dasar: Penanggulangan Kemiskinan*. Jakarta: Lembaga Penelitian SMERU.
- Lembaga Penelitian SMERU. 2016. *Penetapan Kriteria dan Variabel Pendataan Penduduk Miskin yang Komprehensif dalam Rangka Perlindungan Penduduk Miskin di Kabupaten/Kota*. Jakarta: Lembaga Penelitian SMERU.
- Mamone, Mark. 2006. *Practical Mono*. Wisconsin: Kinetic Publishing Services.
- Maniah dan Dini Hamidin. 2017. *Analisis dan Perancangan Sistem Informasi: Pembahasan Secara Praktisi dengan Contoh Kasus*. Yogyakarta: Deepublish.
- MariaDB Foundation. 2017. "About MariaDB". MariaDB Foundation, 2017. <https://mariadb.org/about-us> (30 Agt. 2017).

- MariaDB Foundation. 2017. "Why is the Software Called MariaDB?". MariaDB Foundation, 2017. <https://mariadb.com/kb/en/the-mariadb-library/why-is-the-software-called-mariadb/> (09 Sept. 2017).
- MariaDB Foundation. 2017. "Why MariaDB". MariaDB, 2017. <https://mariadb.com/products/why-mariadb> (09 Sept. 2017).
- Marlinda, Linda. 2004. *Sistem Basis Data*. Yogyakarta: ANDI.
- McLeod, Jr. Raymond dan George P. Schell. 2008. *Sistem Informasi Manajemen*. Edisi ke-10. Terjemahan Ali Akbar Yulianto dan Afia R. Fitriati. Jakarta: Salemba Empat.
- McLeod, Jr. Raymond. 2001. *Sistem Informasi Manajemen: Jilid 2*. Edisi Bahasa Indonesia. Terjemahan Hendra Teguh. Jakarta: Prenhallindo.
- Milroy, Steve, et al. 2002. *.NET Mobile Web Developers Guide*. Rockland: Syngress Publishing.
- Munir, Badrul. 2002. "Perencanaan Pembangunan Daerah". BAPPEDA Provinsi Nusa Tenggara Barat.
- Muslihudin, Muhamad dan Oktafianto. 2016. *Analisis dan Perancangan Sistem Informasi menggunakan Model Terstruktur dan UML*. Yogyakarta: ANDI.
- Nazara, Suahasil dan Sri Kusumastuti Rahayu. 2013. "Program Keluarga harapan (PKH): Program Bantuan Tunai Bersyarat di Indonesia". *TN2PK Research Brief*. Oktober 2013. Nomor 42:1-5. Jakarta: Tim Nasional Percepatan Penanggulangan Kemiskinan.
- Nofriansyah, Dicky. 2014. *Konsep Data Mining vs Sistem Pendukung Keputusan*. Yogyakarta: Deepublish.
- Nugroho, Bunafit. 2004. *PHP dan MySQL dengan Editor DreamWeaver MX*. Yogyakarta: ANDI.
- Nugroho, Bunafit. 2013. *Dasar Pemrograman Web PHP - MySQL dengan Dreamweaver*. Yogyakarta: Gava Media.

- Prasetyo, Eko. 2008. *Pemrograman Web PHP dan MySQL untuk Sistem Informasi Perpustakaan*. Yogyakarta: Graha Ilmu.
- Pressman, Roger S.. 2002. *Rekayasa Perangkat Lunak: Pendekatan Praktisi*. Buku ke-1. Terjemahan Tim Penerjemah Andi. Yogyakarta: ANDI.
- Purwandari, Heru. 2011. "Respon Petani Atas Kemiskinan Struktural: Kasus Desa Perkebunan dan Desa Hutan". *J-SEP*. Juli 2011. Vol. 05 No. 02:24-37. Bogor: Institut Pertanian Bogor.
- Putra, Apriansyah dan Diana Yunika Hardiyanti. 2011. "Penentuan Penerima Beasiswa Dengan Menggunakan *Fuzzy Multiple Attribute Decision Making*". April 2011. Vol. 03 No. 01:286-293. Palembang: Universitas Sriwijaya.
- Qardhawi, Yusuf. 2005. *Spektrum Zakat dalam Membangun Ekonomi Kerakyatan*. Jakarta: Zikrul.
- Rahayu, Trophy Endah. 2011. *Pertumbuhan dan Persebaran Penduduk Indonseia: Hasil Sensus Penduduk 2010*. Jakarta-Indonesia: Badan Pusat Statistik.
- Ramaiah K., Dasaradh. 2014. *Gate and Pgecet for Computer Science and Information Technology*. Delhi: PHI Learning.
- Ramakrishnan, Raghu dan Johannes Gehrke. 2003. *Database Management Systems*. New York: McGraw-Hill Education.
- Remi, Sutycastle Soemitro dan Prijono Tjiptoherijanto. 2002. *Kemiskinan dan Ketidakmerataan di Indonesia*. Jakarta: Penerbit Rineka Cipta.
- Republik Indonesia. 2003. *Undang-Undang No. 20 Tahun 2003 tentang Sistem Pendidikan Nasional*. Lembaran Negara RI Tahun 2003, No. 4301. Sekretariat Negara. Jakarta.
- Ridlo, Mohammad Agung. 2001. *Kemiskinan di Perkotaan*. Semarang: Universitas Sultan Agung Press.
- Riyanto, Slamet. 2014: *Kupas Tuntas Web Responsif*. Jakarta: Elex Media Komputindo.

- Rusnadi, Indra Fantara. 2014. *Implementasi Fuzzy Multiple Attribute Decision Making dengan Metode Simple Additive Weighting dalam Sistem Pendukung Keputusan dengan contoh kasus Penerimaan Beasiswa*. Tugas Akhir. Universitas Widyatama. Bandung.
- Rustanti, Ninik. 2015. *Buku Ajar: Ekonomi Pangan dan Gizi*. Yogyakarta: Deepublish.
- Santoso, Harip. 2004. *VB.NET untuk .NET Programmer*. Jakarta: Elex Media Komputindo.
- Sarosa, Samiaji. 2009. *Sistem Informasi Akuntansi*. Yogyakarta: ANDI.
- Sheldon, Bill, et al. 2010. *Professional Visual Basic 2010 and .NET 4*. Indiana: Wiley Publishing.
- Sheldon, Bill, et al. 2013. *Professional Visual Basic 2012 and .NET 4.5 Programming*. Indiana: John Wiley and Sons.
- Shelly, Gary B. dan Corinne Hoisington. 2010. *Microsoft Visual Basic 2010 for Windows, Web, Office, and Database Applications: Comprehensive*. Boston: Course Technology, Cengage Learning.
- Sidik, Betha. 2012. *Pemrograman Web dengan PHP*. Bandung: Informatika.
- Simanjuntak, Timbul Hamonangan dan Imam Mukhlis. 2012. *Dimensi Ekonomi Perpajakan dalam Pembangunan Ekonomi*. Jakarta: Raih Asa Sukses.
- Singh, Shio Kumar. 2011. *Database Systems: Concepts, Design, and Application*. Edisi ke-2. India: Dorling Kindersley.
- Sitohang, P. 1972. *Pembangunan Ekonomi*. Jakarta: Bratara.
- Sole, Alessandro Del. 2015. *Visual Basic 2015 Unleashed*. Indiana: Sams Publishing.
- Solihin, Muhammad, et al. 2016. "Implementasi Panel Pengunjung Berbasis Layar Sentuh Sebagai Sub Sistem Kunci Otomatis Pada Ruang Dosen Universitas

- Telkom". *E-Proceeding of Engineering*. April 2016. Vol. 3 No. 1:653-660. Bandung: Universitas Telkom.
- Suharto, Edi. 2009. *Membangun Masyarakat, Memberdayakan Rakyat: Kajian Strategis Pembangunan Kesejahteraan Sosial dan Pekerjaan Sosial*. Bandung: Refika Aditama.
- Sukamto, Rosa Ariani dan M Shalahuddin. 2013. *Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek*. Bandung: Informatika.
- Sukirno, Sadono. 1985. *Ekonomi Pembangunan*. Jakarta: LPEF-UI Bima Grafika.
- Sulistiyani, Teguh Ambar. 2004. *Kemitraan dan Model-Model Pemberdayaan*. Yogyakarta: Gava Media.
- Sumodiningrat, Gunawan. 1999. *Pemberdayaan Masyarakat dan Jaring Pengaman Sosial*. Jakarta: Gramedia.
- Sunarfrihantono, Bimo. 2002. *PHP dan MySQL untuk Web*. Yogyakarta: ANDI.
- Suparlan, Parsudi. 2003. "Segi Sosial dan Ekonomi Pemukiman Kumuh". Surakarta: Universitas Muhammadiyah Surakarta. (2008). http://geografi.ums.ac.id/ebook/Social_Education/SOS_NOMI_KUMUH.pdf (11 Jan. 2017).
- Suryawati, Chriswardani. 2005. "Memahami Kemiskinan secara Multidimensional". *JMPK*. September 2005. Vol. 08 No. 03:121-129. Semarang: Universitas Diponegoro.
- Suryawati. 2004. *Teori Ekonomi Mikro*. Yogyakarta: UPP. Akademi Manajemen Perusahaan YKPN.
- Susilowati, Sri Hery. 2010. "Pendekatan Skala Ekuivalensi Untuk Mengukur Kemiskinan". *Forum Penelitian Agro Ekonomi*. Desember 2010. Vol. 28 No. 02:91-105. Bogor: Pusat Sosial Ekonomi dan Kebijakan Pertanian.
- Sutabri, Tata. 2004. *Analisis Sistem Informasi*. Yogyakarta: ANDI.

- Sutabri, Tata. 2005. *Sistem Informasi Manajemen*. Yogyakarta: ANDI.
- Sutrisno, Debbie. 2016. "Proyek Infrastruktur Besar - Besaran Turunkan Kesenjangan". *Republika*, 19 April 2016. <http://www.republika.co.id/berita/koran/halaman-1/16/04/19/o5va079-...1> (02 Okt. 2016).
- Tarmidi, Lepi T. 1992. *Ekonomi Pembangunan*. Jakarta: Pusat Antar Universitas – EK-UI.
- Tickoo, Sham dan Meenu Bhat. 2009. *Learning Visual Basic 2008 with .Net Framework 3.5*. India: Dorling Kindersley.
- Tim Nasional Percepatan Penanggulangan Kemiskinan. 2010. *Penanggulangan Kemiskinan: Situasi Terkini, Target Pemerintah, dan Program Percepatan*. Edisi ke-2. Jakarta: Tim Nasional Percepatan Penanggulangan Kemiskinan.
- Tim Nasional Percepatan Penanggulangan Kemiskinan. 2010. *Penanggulangan Kemiskinan: Situasi Terkini, Target Pemerintah, dan Program Percepatan*. Jakarta: Tim Nasional Percepatan Penanggulangan Kemiskinan.
- Tim Nasional Percepatan Penanggulangan Kemiskinan. 2013. *Pembangunan Basis Data Terpadu: Untuk mendukung Program Perlindungan Sosial*. Jakarta: Tim Nasional Percepatan Penanggulangan Kemiskinan.
- Tim Nasional Percepatan Penanggulangan Kemiskinan. 2015. *Standar Pengelolaan Data Terpadu: Untuk Program Perlindungan Sosial*. Jakarta: Tim Nasional Percepatan Penanggulangan Kemiskinan.
- Tim Penelitian dan Pengembangan Foundstone dan CORE. 2000. *Security in the Microsoft .NET Framework*. Makalah disajikan dalam rangka untuk menganalisis arsitektur keamanan dari *.Net Framework* milik Perusahaan Microsoft, New York, Summer 2000.
- Tim Penelitian dan Pengembangan Wahana Komputer. 2001. *Pemrograman Visual Basic 6.0*. Yogyakarta: ANDI.

- Tim Penelitian dan Pengembangan Wahana Komputer. 2010. *Panduan Praktis Implementasi Konfigurasi Jaringan MS Windows 2008 Server*. Yogyakarta: ANDI.
- Turban, Elfrain, et al. 2005. *Sistem Pendukung Keputusan dan Sistem Cerdas: Jilid 1*. Edisi ke-7. Terjemahan Dwi Prabantini. Yogyakarta: Andi.
- United Nations Development Programme. 2003. *Human Development Report 2003-Millennium Development Goals: A compact among nations to end human poverty*. New York: Oxford University Press.
- Visual Studio. 2017. "Visual Studio 2017 15.3 Release Notes". Microsoft, 14 Agustus 2017. <https://www.visualstudio.com/en-us/news/releasenotes/vs2017-relnotes> (08 Okt. 2017).
- Whitten, Jeffery L. et al. 2006. *Metode Desain dan Analisis Sistem*. Edisi ke-6. Edisi Internasional. Terjemahan Tim Penerjemah Andi. Yogyakarta: ANDI.
- Widarjono, Agus. 1999. "Penduduk dan Pertumbuhan Ekonomi di Indonesia: Analisis Kausalitas". *Jurnal Ekonomi Pembangunan*. Vol. 4 No. 2 Tahun 1999.
- Wijayanti, Diana dan Heri Wahono. 2005. "Analisis Konsentrasi Kemiskinan di Indonesia Periode Tahun 1999-2003". *Jurnal Ekonomi Pembangunan*. Desember 2005. Vol. 10 No. 3:215-225. Yogyakarta: Universitas Islam Indonesia.
- Winarno, Edy, et al. 2010. *Dasar-Dasar Pemrograman Visual Basic 2010*. Jakarta: Elex Media Komputindo.
- Yanto, Robi. 2016. *Manajemen Basis Data menggunakan MySQL*. Yogyakarta: Deepublish.
- Yuhefizard. 2003. *Database Management menggunakan Microsoft Access 2003*. Jakarta: Elex Media Komputindo.