

BAB II TINJAUAN PUSTAKA

Tinjauan pustaka merupakan kumpulan dasar teori yang digunakan untuk menunjang kelancaran pelaksanaan penelitian. Disamping itu, tinjauan pustaka juga dapat berguna sebagai pedoman dalam penelitian sehingga pelaksanaan penelitian dapat terfokus pada tujuan yang akan dicapai.

2.1 Penelitian Terdahulu

Terdapat beberapa penelitian yang telah dilakukan mengenai material requirement planning, inventori maupun sistem basis data yang menjadi referensi dalam melakukan penelitian ini. Berikut adalah beberapa hasil penelitian terdahulu:

1. Rusman. (2014), melakukan inventory control yang dapat mengatur ketersediaan bahan baku produksi pada PT Bambang Djaja Surabaya. Pada penelitian ini menggunakan Metode fuzzy mamdani. Hasil yang sudah didapat adalah model fuzzy yang digunakan untuk inventory control dalam menentukan jumlah order bahan baku (order quantity) dan titik pemesanan kembali (reorder point) berdasarkan *input* persediaan dan permintaan di setiap periode.
2. Sukandy dkk. (2013), dalam penelitiannya ini mengambil studi kasus di PT Perkebunan Mitra Ogan Baturaja. Perusahaan ini bergerak di bidang pengolahan minyak sawit. Tujuan dari penelitian tersebut untuk membuat aplikasi yang menerapkan metode *fuzzy* mamdani untuk prediksi jumlah produksi berdasarkan data permintaan dan persediaan. Hasil pengujian aplikasi, diketahui bahwa prediksi dari metode mamdani sebagian besar dapat memenuhi permintaan yang ada. Dengan ini, perusahaan dapat memprediksi lebih cepat dari pada dengan manual.
3. Atta dkk. (2015), objek penelitian di PT. UD Lumba-lumba. Perusahaan ini memproduksi keripik singkong yang terletak di malang. Penelitian ini merancang sistem informasi manajemen produksi dengan pemanfaatan pendekatan *fuzzy* logic untuk penentuan jumlah produksi. Hasilnya, dengan menggunakan *fuzzy* mamdani dan sistem informasi PHP dan MySQL dapat mengelola informasi dengan lebih baik dan dapat menggantikan keputusan produksi dari seorang pakar.

Berikut ini merupakan perbandingan penelitian yang dapat dilihat pada Tabel 2.1.

Tabel 2.1 Perbandingan Penelitian

	Produk	Metode	Hasil Penelitian
Rusman (2014)	Trafo	<i>Fuzzy</i> mamdani, MATLAB	Memberikan hasil <i>output</i> berupa <i>order quantity</i> dan ROP berdasarkan <i>input</i> permintaan dan persediaan. Perbandingan perhitungan fuzzy menggunakan MATLAB dan manual mendekati kebenaran.
Sukandy (2013)	Minyak sawit	<i>Fuzzy</i> mamdani, SQL dan Visual Basic	Diketahui bahwa prediksi jumlah produksi berdasarkan data permintaan dan persediaan dari metode mamdani sebagian besar dapat memenuhi permintaan yang ada.
Atta (2015)	Keripik Singkong	<i>Fuzzy Mamdani</i> , PHP dan MySQL	Dapat mengelola informasi lebih baik dan dapat menggantikan keputusan jumlah produksi dari seorang pakar.
Penelitian ini (2016)	Aspal	<i>Fuzzy</i> mamdani, MATLAB, Ms. Access dan VBA	Pengendalian persediaan berupa pendukung keputusan <i>order quantity</i> dan ROP dan menghasilkan sistem Informasi Manajemen Basis Data yang lebih baik daripada sistem sebelumnya.

2.2 Persediaan

Persediaan dapat diartikan sebagai barang-barang yang disimpan untuk digunakan atau dijual pada masa atau periode yang akan datang (Ristono, 2009). Perusahaan yang melakukan kegiatan produksi akan memiliki tiga jenis persediaan, yaitu:

1. Persediaan bahan baku
2. Persediaan bahan setengah jadi
3. Persediaan barang jadi

Adanya berbagai macam persediaan menuntut perusahaan untuk melakukan tindakan yang berbeda dan juga terkait dengan permasalahan permintaan konsumen. Sehingga pengendalian persediaan perlu diperhatikan karena berkaitan langsung dengan biaya yang harus ditanggung perusahaan sebagai akibat adanya persediaan. Oleh sebab itu, persediaan yang ada harus seimbang dengan kebutuhan.

2.3 Perencanaan dan Pengendalian Produksi

Menurut Teguh Baroto (2002) perencanaan dan pengendalian produksi (PPC) adalah aktivitas bagaimana mengelola proses produksi tersebut. PPC merupakan tindakan manajemen yang sifatnya abstrak (tidak dapat dilihat secara nyata). Sistem komputer barangkali merupakan analogi yang tepat untuk sistem produksi. Proses produksi adalah perangkat kerasnya (*hardware*) dan PPC adalah perangkat lunaknya (*software*). PPC pada industri manufaktur apapun akan memiliki fungsi atau aktivitas yang sama diantaranya:

1. Mengelola persediaan. Tindakan pengelolaan persediaan diantaranya berupa melakukan transaksi persediaan.
2. Menyusun rencana agregat (penyusunan permintaan dengan kapasitas).
3. Membuat Jadwal Induk Produksi (JIP). JIP adalah rancangan terperinci mengenai apa dan berapa unit yang harus diproduksi pada suatu periode tertentu untuk setiap item produksi.
4. Merencanakan kebutuhan. JIP yang telah berisi apa dan berapa yang harus dibuat selanjutnya harus diterjemahkan ke dalam kebutuhan komponen, sub-assembly dan bahan penunjang untuk penyelesaian produk.
5. Monitoring dan pelaporan pembebanan kerja disbanding kapasitas produksi.

2.4 Logika Fuzzy

Fuzzy logic atau *system fuzzy* merupakan suatu cara yang tepat untuk memetakan suatu ruang input ke dalam suatu ruang output. Dasar logika *fuzzy* adalah teori himpunan *fuzzy*. Menurut Kusumadewi (2010) Pada teori himpunan *fuzzy*, peranan derajat keanggotaan sebagai penentu keberadaan elemen dalam suatu himpunan sangatlah penting. Nilai keanggotaan atau derajat keanggotaan atau *membership function* menjadi ciri utama dalam penalaran dengan logika *fuzzy* tersebut.

2.4.1 Himpunan Fuzzy

Pada himpunan tegas (*crisp*), nilai keanggotaan suatu item x dalam suatu himpunan A yang sering ditulis dengan $\mu_A[x]$, memiliki 2 kemungkinan (Suyanto, 2011) yaitu:

- Satu (1), yang berarti bahwa suatu item menjadi anggota dalam suatu himpunan, atau
- Nol (0), yang berarti bahwa suatu item tidak menjadi anggota dalam suatu himpunan

Jika pada himpunan *crisp*, nilai keanggotaan hanya ada 2 kemungkinan yaitu 0 dan 1, maka pada himpunan *fuzzy* nilai keanggotaan terletak pada rentang 0 sampai 1. Himpunan *fuzzy* memiliki 2 atribut:

- a. Linguistik, yaitu penamaan suatu grup yang mewakili suatu keadaan atau kondisi tertentu dengan menggunakan bahasa alami, seperti MUDA, PAROBAYA, dan TUA.
- b. Numeris, yaitu suatu nilai (angka) yang menunjukkan ukuran dari suatu variabel seperti: 40, 25, 35, dan lain-lain.

Ada beberapa hal yang perlu diketahui dalam memahami sistem *fuzzy*, yaitu:

1. Variabel *fuzzy* merupakan variable yang hendak dibahas dalam suatu sistem *fuzzy*. Contoh: umur, temperatur, dan permintaan.
2. Himpunan *fuzzy* merupakan suatu grup yang mewakili suatu kondisi atau keadaan tertentu dalam suatu variabel *fuzzy*. Contoh: variabel umur terbagi menjadi 3 himpunan *fuzzy*, yaitu MUDA, PAROBAYA, dan TUA.
3. Semesta pembicaraan adalah keseluruhan nilai yang diperbolehkan untuk dioperasikan dalam suatu variabel *fuzzy*. Semesta pembicaraan merupakan himpunan bilangan real yang senantiasa naik (bertambah) secara monoton dari kiri ke kanan. Nilai semesta pembicaraan dapat berupa bilangan positif maupun negatif. Adakalanya nilai semesta pembicaraan ini tidak dibatasi batas atasnya. Contoh: semesta pembicaraan untuk variabel umur = $(0 - \infty)$
4. Domain himpunan *fuzzy* adalah keseluruhan nilai yang diizinkan dalam semesta pembicaraan dan boleh dioperasikan dalam suatu himpunan *fuzzy*. Seperti halnya semesta pembicaraan, domain merupakan himpunan bilangan real yang senantiasa naik (bertambah) secara monoton dari kiri ke kanan. Nilai domain dapat berupa bilangan positif maupun negatif. Contoh:
 - a. MUDA = $(0 \ 45)$
 - b. PAROBAYA = $(35 \ 55)$
 - c. TUA = $(45 \ +\infty)$

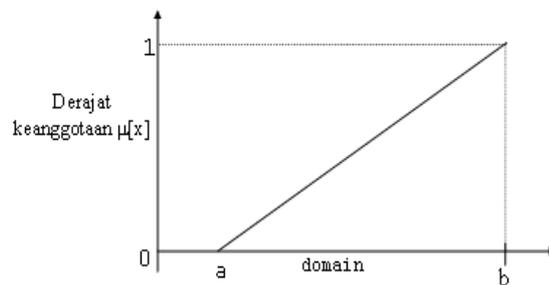
2.4.2 Fungsi Keanggotaan

Fungsi keanggotaan (*membership function*) adalah suatu kurva yang menunjukkan pemetaan titik-titik input data ke dalam nilai keanggotaannya (sering juga disebut dengan derajat keanggotaan) yang memiliki interval antara 0 sampai 1. Salah satu cara yang dapat digunakan untuk mendapatkan nilai keanggotaan adalah dengan melalui pendekatan fungsi. Ada beberapa fungsi yang bisa digunakan.

a. Representasi Linier

Pada representasi linier, pemetaan *input* ke derajat keanggotaannya digambarkan sebagai suatu garis lurus. Bentuk ini paling sederhana dan menjadi pilihan yang baik untuk mendekati suatu konsep yang kurang jelas.

Terdapat 2 keadaan himpunan fuzzy yang linier. Pertama, kenaikan himpunan dimulai pada nilai domain yang memiliki derajat keanggotaan nol [0] bergerak ke kanan menuju ke nilai domain yang memiliki derajat keanggotaan lebih tinggi yang dapat dilihat pada Gambar 2.1.

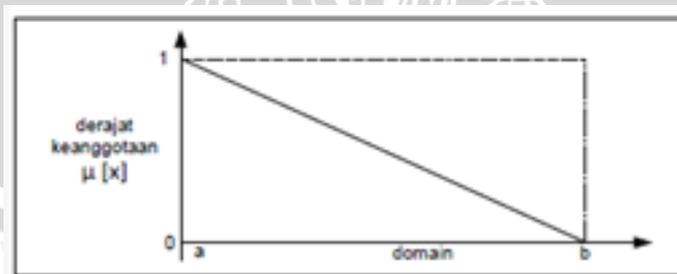


Gambar 2.1 Representasi Linier Naik
Sumber: Kusumadewi, 2010

Rumus fungsi keanggotaan representasi linier naik.

$$\mu[x] = \begin{cases} 0; & x \leq a \\ \frac{x-a}{b-a}; & a \leq x \leq b \\ 1; & x \geq b \end{cases} \quad (2-1)$$

Kedua, merupakan kebalikan yang pertama. Garis lurus dimulai dari nilai domain dengan derajat keanggotaan tertinggi pada sisi kiri, kemudian bergerak menurun ke nilai domain yang memiliki derajat keanggotaan lebih rendah yang dapat dilihat pada Gambar 2.2.



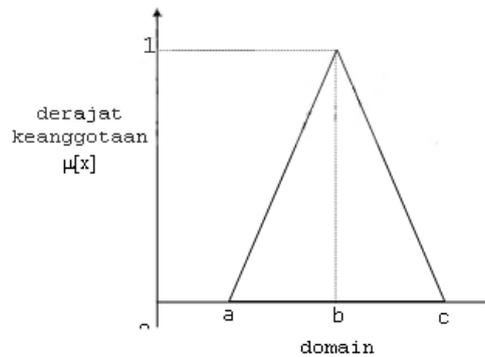
Gambar 2.2 Representasi Linier turun
Sumber: Kusumadewi, 2010

Rumus fungsi keanggotaan representasi linier turun.

$$\mu[x] = \begin{cases} \frac{b-x}{b-a} & a < x < b \\ 0 & x \geq b \end{cases} \quad (2-2)$$

b. Representasi Kurva Segitiga

Kurva segitiga pada dasarnya merupakan gabungan antara 2 garis (linier) seperti terlihat pada Gambar 2.3



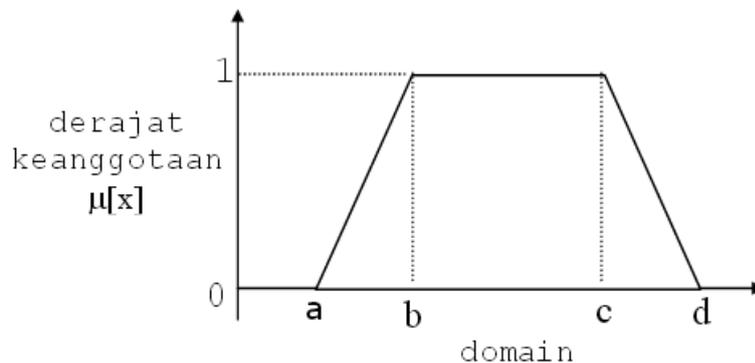
Gambar 2.3 Kurva Segitiga
Sumber: Kusumadewi, 2010

Rumus fungsi keanggotaan kurva segitiga :

$$\mu[x; a, b, c] = \begin{cases} 0; & x \leq a \text{ atau } x \geq c \\ \frac{(x-a)}{(b-a)}; & a \leq x \leq b \\ \frac{(c-x)}{(c-b)}; & b \leq x \leq c \end{cases} \quad (2-3)$$

c. Representasi Kurva Trapesium

Kurva trapesium pada dasarnya seperti bentuk segitiga, hanya saja ada beberapa titik yang memiliki nilai keanggotaan 1 seperti terlihat pada Gambar 2.4.



Gambar 2.4 Kurva Trapesium
Sumber: Kusumadewi, 2010

Rumus fungsi keanggotaan kurva trapesium :

$$\mu[x; a, b, c] = \begin{cases} 0; & x \leq a \text{ atau } x \geq d \\ \frac{(x-a)}{(b-a)}; & a \leq x \leq b \\ 1; & b \leq x \leq c \\ \frac{(d-x)}{(d-c)}; & c \leq x \leq d \end{cases} \quad (2-4)$$

2.5 Sistem Inferensi Fuzzy

Sistema inferensi *fuzzy* adalah cara memetakan ruang input menuju ruang output menggunakan logika *fuzzy* (Sutojo, 2011). Sistema inferensi *fuzzy* mencoba untuk memformulasikan proses-proses penalaran bahasa manusia melalui logika guzzy dan biasa digunakan untuk membuat keputusan dan bekerja sesuai dengan keputusan yang dibuat tersebut.

2.5.1 Fuzzy Metode Mamdani

Menurut Sutojo (2011) metode mamdani paling sering digunakan dalam aplikasi-aplikasi karena strukturnya yang sederhana, yaitu menggunakan operasi MIN-MAX atau MAX-PRODUCT. Untuk mendapatkan output, diperlukan empat tahapan berikut:

1. *Fuzzyfikasi* yaitu suatu proses untuk mengubah suatu masukan dari bentuk tegas (*crisp*) menjadi *fuzzy* (variabel linguistik) yang biasanya disajikan dalam bentuk himpunan-himpunan *fuzzy* dengan suatu fungsi kenggotaannya masing-masing.
2. Pembentukan basis pengetahuan *fuzzy/ rule based* (rule dalam bentuk IF....THEN) merupakan suatu bentuk aturan relasi/implikasi “jika-maka” atau “if-then”.
3. Mesin penalaran yaitu proses implikasi dalam menalar nilai masukan guna penentuan nilai keluaran sebagai bentuk pengambilan keputusan. Aplikasi fungsi implikasi menggunakan fungsi MIN dan Komposisi antar-rule menggunakan fungsi MAX (menghasilkan himpunan *fuzzy* baru).
4. *Defuzzyfikasi* merupakan proses pemetaan himpunan *fuzzy* ke himpunan tegas (*crisp*). Terdapat 4 metode dalam melakukan defuzzifikasi yaitu:

- a. Metode centroid (*Composite Moment*)

Pada metode ini, solusi *crisp* diperoleh dengan cara mengambil titik pusat (z^*) daerah *fuzzy*.

- b. Metode Bisektor

Pada metode ini, solusi *crisp* diperoleh dengan cara mengambil nilai pada domain *fuzzy* yang memiliki nilai keanggotaan setengah dari jumlah total nilai keanggotaan pada daerah *fuzzy*.

- c. Metode *Mean of Maximum* (MOM)

Pada metode ini, solusi *crisp* diperoleh dengan cara mengambil nilai rata-rata domain yang memiliki nilai keanggotaan maksimum.

d. Metode *Largest of Maximum* (LOM)

Pada metode ini, solusi *crisp* diperoleh dengan cara mengambil nilai terbesar dari domain yang memiliki nilai keanggotaan maksimum.

e. Metode *Smallest of Maximum* (SOM)

Pada metode ini, solusi *crisp* diperoleh dengan cara mengambil nilai terkecil dari domain yang memiliki nilai keanggotaan maksimum.

2.5.2 Fuzzy Metode Tsukamoto dan fuzzy Metode Sugeno

Menurut Kusumadewi (2010) pada metode tsukamoto, setiap konsekuen pada aturan yang berbentuk *if-then* harus direpresentasikan dengan suatu himpunan *fuzzy* dengan fungsi keanggotaan yang monoton. Sebagai hasilnya, *output* hasil inferensi dari tiap aturan diberikan secara tegas (*crisp*) berdasarkan α -predikat (*fire strength*). Hasil akhirnya diperoleh dengan menggunakan rata-rata terbobot.

Penalaran dengan metode sugeno hampir sama dengan penalaran metode mamdani, hanya saja *output* (konsekuen) sistem tidak berupa himpunan fuzzy, melainkan berupa konstanta atau persamaan linier.

Dari tiga metode yaitu tsukamoto, mamdani dan sugeno, dipilihnya metode mamdani dibandingkan dengan yang lain diantaranya adalah karena bersifat intuitif, mencakup bidang yang luas, dan sesuai dengan proses input informasi manusia.

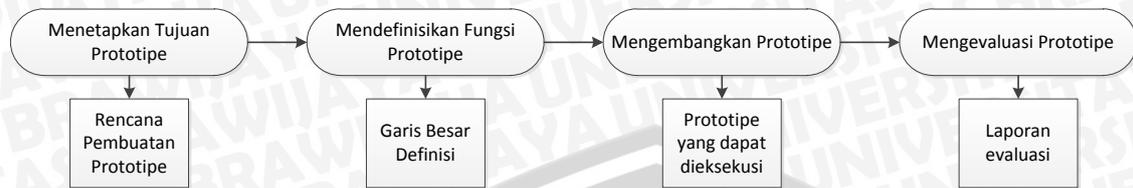
2.6 Analisis dan Perancangan Sistem

Menurut Shelly dan Rosenblatt (2012) adalah tahapan-tahapan proses untuk mengembangkan sistem informasi berkualitas tinggi. Sehingga analisis dan perancangan sistem didefinisikan sebagai proses organisasional kompleks dimana sistem informasi berbasis computer diimplementasikan. Sebuah sistem informasi mengkombinasikan teknologi, manusia dan data untuk mendukung kebutuhan bisnis.

2.7 Prototype

Prototyping adalah pengembangan yang cepat dan pengujian terhadap model kerja (*prototyping*) dari aplikasi baru melalui proses interaksi dan berulang-ulang yang biasa digunakan ahli sistem informasi dan ahli bisnis (Mulyanto, 2009). *Prototyping* disebut juga desain aplikasi cepat atau *Rapid Application Design* (RAD) karena menyederhanakan dan mempercepat desain sistem (O'Brien, 2005). Proses pembuatan *prototype* merupakan proses yang interaktif dan berulang-ulang dengan proses menggabungkan langkah-langkah

dari siklus yang ada dalam pengembangan sistem. *Prototype* dievaluasi beberapa kali sebelum pemakai akhir menyatakan *prototype* diterima. Berikut ini merupakan model pengembangan *prototype* yang dijelaskan pada Gambar 2.5.



Gambar 2.5 Model Pengembangan Prototipe
Sumber: Sommerville, 2006

2.8 Sistem Basis Data

Menurut Jogiyanto (2005) Sistem Basis Data adalah suatu sistem informasi yang mengintegrasikan kumpulan dari data yang berhubungan satu dengan yang lainnya dan membuatnya tersedia untuk beberapa aplikasi yang bermacam-macam di dalam suatu organisasi. Dalam suatu sistem basis data (DBMS) terdapat lima komponen, yaitu:

1. *Hardware* (Perangkat keras)

Hardware dalam hal ini dapat berupa personal komputer, notebook dan jaringan komputer.

2. *Software* (Perangkat Lunak)

Software biasanya terbagi menjadi:

- a. *Software* sistem operasi komputer, contohnya Windows XP, Windows 7, Unix dan Linux.
- b. *Software* untuk *database*, contonya My SQL, Microsoft Access, dan Oracle.
- c. *Software* untuk pemrograman seperti Visual Basic 6.0, Java, C++, Visual Basic .Net, Fortran, dan lain-lain.
- d. *Software* untuk jaringan seperti SISCO.

3. Data

Data merupakan komponen terpenting dalam suatu sistem basis data. Data berfungsi sebagai penghubung antara komputer dan manusia (pengguna).

4. Prosedur

Prosedur adalah kumpulan instruksi dan aturan yang menentukan perancangan dan penggunaan sistem basis data.

5. Manusia

Berdasarkan perannya dalam sebuah basis data, manusia dibedakan menjadi beberapa fungsi antara lain:

a. *Administrator*

Merupakan orang ataupun sekelompok orang yang bertanggung jawab untuk mengatur dan mengendalikan basis data tersebut.

b. *Designer*

Dibedakan menjadi 2, yaitu perancang basis data secara logika dan perancang basis data secara fisik. Perancangan basis data secara logika berhubungan dengan identifikasi data-data antara lain entitas dan atribut. Sedangkan perancangan data secara fisik berhubungan dengan bagaimana desain basis data tersebut dapat direalisasikan.

c. *Programmer*

Merupakan tenaga ahli komputer yang bertugas untuk mengembangkan program-program aplikasi yang diperlukan dalam sistem basis data.

d. *End User*

Yang termasuk dalam pengguna akhir adalah para pemilik sistem, manager, supervisor, operator pelanggan dan lain sebagainya yang terlibat langsung dalam penggunaan sistem basis data.

2.9 *Database Management System (DBMS)*

Database management system adalah suatu sistem atau perangkat lunak yang dirancang untuk mengelola suatu basis data dan menjalankan operasi terhadap data yang diminta banyak pengguna. DBMS merupakan perangkat lunak yang dirancang untuk dapat melakukan utilisasi dan mengelola koleksi data dalam jumlah besar. DBMS juga dirancang untuk dapat melakukan manipulasi data secara mudah.

2.10 *Relational*

Menurut Mark Whitehorn dan Bill Marklyn (2003) *database* relasional adalah Suatu *database* yang menggunakan model relasional, dimana model tersebut menyediakan mekanisme-mekanisme yang kita perlukan untuk menyimpan dan memanipulasi data yang kompleks dengan cara yang sedemikian rupa sehingga kita dapat meng-*query* data itu dan memperoleh jawaban-jawaban yang kita perlukan. Ada dua *field* kunci sejenis yang dapat menghubungkan semua tabel yaitu:

1. *Primary key* merupakan suatu *field* atau kombinasi *field* yang secara unik mengidentifikasi setiap *record* di dalam tabel. Contoh: NIM yang terdapat pada tabel biodata mahasiswa
2. *Foreign key* merupakan suatu *field* atau kombinasi *field* yang lebih baik dibandingkan dengan *field primary key* atau *field* di tabel lain dalam mengidentifikasi suatu *record*. Contoh NIM yang terdapat pada tabel KHS mahasiswa.

2.10.1 Jenis-jenis *Relational* dalam Basis Data

Dalam basis data dikenal 3 jenis hubungan, antara lain:

1. *One To One Relationship*

Di *one to one relationship*, setiap *record* di tabel A hanya dapat mempunyai satu *record* yang sesuai di tabel B dan setiap *record* di tabel B hanya dapat mempunyai satu *record* yang sesuai di tabel A. Tipe dari *relationship* ini tidak biasa, karena banyak sekali informasi yang saling berhubungan di dalam cara ini akan menjadi satu tabel. Untuk contoh kamu mungkin ingin membuat sebuah tabel untuk mencari partisipasi pegawai dalam penyumbangan dana permainan sepak bola. Setiap pemain sepak bola di dalam tabel pemain sepak bola mempunyai satu *record* yang sesuai di tabel pegawai. Jadi *one to one relationship* dibuat jika kedua *field* yang saling berhubungan mempunyai *primary key* atau *unique index*.

2. *One To Many Relationship*

Di *one to many relationship* sebuah *record* atau tampilan di table A dapat mempunyai banyak kesesuaian dengan *record* table B, tetapi *record* di table B hanya satu yang sesuai dengan *record* di tabel A. Jadi *one to many relationship* dibuat jika hanya dari *field* yang dihubungkan mempunyai *primary key* atau *unique index*. *Unique index* adalah sebuah *index* diartikan oleh *field* yang diatur petunjuk *property* untuk tidak menyalin atau meniru.

3. *Many To Many Relationship*

Di *many to many relationship*, sebuah *record* di tabel A dapat mempunyai banyak kesesuaian *record* di tabel B dan *record* di tabel B dapat mempunyai banyak kesesuaian *record* di tabel A. Tipe dari *relationship* ini hanya mungkin ditegaskan dengan tiga tabel (disebut *junction table*) yang memiliki *primary key* yang terdiri dari dua *field*, *foreign key* dari tabel A dan tabel B. contoh yaitu satu pengiriman dapat mendapatkan banyak barang dan setiap barang dapat kelihatan dari banyaknya pengiriman. Jadi *many to many relationship* sebenarnya dua kalinya dari *one to many*

relationship dengan *junction table* yang memiliki *primary key* yang terdiri dari dua *field*, *foreign key* dari dua tabel yang berbeda.

4. None

Pada kenyataannya ada beberapa objek yang memang tidak memiliki hubungan sama sekali satu dengan lainnya. Sebagai contoh informasi tentang sewa kantor tidak berhubungan sama sekali dengan informasi tentang pelanggan.

2.11 Normalisasi

Proses normalisasi merupakan proses pengelompokan elemen data menjadi table-table yang optimal. Diharapkan melalui proses normalisasi, satu tabel mewakili satu entity atau satu konsep, bukannya satu tabel yang terdiri atas puluhan kolom untuk menyimpan semua informasi. Pada proses normalisasi selalu diuji pada beberapa kondisi. Apakah ada kesulitan saat menambah (*insert*), menghapus (*delete*), mengubah (*update*), membaca/mencari (*retrieve*) pada satu *database*. Bila ada kesulitan pada pengujian tersebut maka relasi tersebut dipecahkan pada beberapa tabel lagi atau dengan kata lain perancangan belumlah mencapai *database* yang optimal. Tujuan normalisasi menurut Whitehorn dan Marklyn (2004):

1. Untuk mencari dan mengelompokkan seluruh sifat yang berkenaan dengan objek tertentu.
2. Untuk menghilangkan informasi yang berlebihan.
3. Untuk memberikan identifikasi unik bagi *record* individu (*primary key*).

Dalam sistem basis data, terdapat empat bentuk normalisasi yang biasa digunakan antara lain:

1. Bentuk Tidak Normal (*Unnormalized Form*)

Bentuk ini merupakan kumpulan data yang akan direkam, tidak ada keharusan mengikuti suatu format tertentu, data dapat tidak lengkap atau terduplikasi, data dikumpulkan apa adanya sesuai dengan kedatangannya.

2. Bentuk Normal Kesatu (*1NF/First Normal Form*)

Ciri dari bentuk normal kesatu ini yaitu pada setiap data dibentuk dalam flat file, data dibentuk dalam satu demi satu *record* dan nilai dari *field-field*nya berupa *atomic value*. Tidak ada set atribut yang berulang atau atribut yang bernilai ganda. Tiap *field*nya hanya satu pengertian, bukan merupakan kumpulan data yang mempunyai dua arti, hanya satu arti saja dan juga bukan pecahan kata-kata sehingga artinya lain.

3. Bentuk Normal Kedua (2NF/Second Normal Form)

Bentuk normal kedua mempunyai syarat, yaitu bentuk data lebih memenuhi kriteria bentuk normal kesatu. Atribut bukan kunci yang harus bergantung secara fungsi pada kunci utama, sehingga untuk membentuk normal kedua harus sudah ditentukan kunci-kunci *fieldnya*. Kunci *fieldnya* haruslah unik dan dapat mewakili atribut lain yang menjadi anggotanya.

4. Bentuk Normal Ketiga (3NF/Third Normal Form)

Untuk menjadi normal ketiga maka relasi harus dalam bentuk normal kedua dan setiap atribut yang bukan kunci harus bergantung hanya pada *primary key* secara menyeluruh.

2.12 Data Flow Diagram (DFD)

Data Flow Diagram (DFD) adalah representasi grafik dari sebuah sistem. DFD menggambarkan komponen-komponen sebuah sistem, aliran-aliran data di mana komponen-komponen tersebut, asal, tujuan, dan penyimpanan dari data tersebut. Kita dapat menggunakan DFD untuk dua hal utama, yaitu untuk membuat dokumentasi dari sistem informasi yang ada atau untuk menyusun dokumentasi untuk sistem informasi yang baru. ada beberapa simbol digunakan pada DFD untuk mewakili :

1. Kesatuan Luar (*External Entity*)

Kesatuan luar (*external entity*) merupakan kesatuan (*entity*) di lingkungan luar sistem yang dapat berupa orang, organisasi, atau sistem lain yang berada pada lingkungan luarnya yang memberikan *input* atau menerima *output* dari sistem.

2. Arus Data (*Data Flow*)

Arus Data (*data flow*) di DFD diberi simbol suatu panah. Arus data ini mengalir di antara proses, simpan data dan kesatuan luar. Arus data ini menunjukkan arus dari data yang dapat berupa masukan untuk sistem atau hasil dari proses sistem.

3. Proses (*Process*)

Proses (*process*) menunjukkan pada bagian yang mengubah input menjadi output, yaitu menunjukkan bagaimana satu atau lebih input diubah menjadi beberapa output. Setiap proses mempunyai nama, nama dari proses ini menunjukkan apa yang dikerjakan proses.

4. Simpanan Data (*Data Store*)

Data Store merupakan simpanan dari data yang dapat berupa suatu *file* atau *database* pada sistem komputer.

2.13 Entity Relationship Diagram (ERD)

Entity Relational Diagram merupakan salah satu pemodelan data konseptual yang paling sering digunakan dalam proses pengembangan basis data bertipe relasional. Model E-R adalah rincian yang merupakan representasi logika dari data pada suatu organisasi atau area bisnis tertentu. Model E-R terdiri dari beberapa komponen dasar yaitu sebagai berikut:

1. Entitas

Entitas adalah sesuatu atau objek di dunia nyata yang dapat dibedakan dari sesuatu atau objek yang lainnya. Sebagai contoh, setiap mahasiswa dalam suatu universitas adalah suatu entitas. Setiap fakultas dalam suatu universitas adalah juga suatu entitas. Dapat dikatakan bahwa entitas bisa bersifat konseptual/abstrak atau nyata hadir di dunia nyata.

2. Hubungan antar relasi (*Relationship*)

Hubungan antar relasi adalah hubungan antara suatu himpunan entitas dengan himpunan entitas yang lainnya. Misalnya, entitas mahasiswa memiliki hubungan tertentu dengan entitas matakuliah (mahasiswa mengambil matakuliah). Pada penggambaran model E-R, relasi adalah perekat yang menghubungkan suatu entitas dengan entitas yang lainnya.

3. Kardinalitas/Derajat Relasi

Kardinalitas relasi menunjukkan jumlah maksimum entitas yang dapat berelasi dengan entitas pada himpunan entitas yang lain. Sebagai contoh: entitas-entitas pada himpunan entitas mahasiswa dapat berelasi dengan satu entitas, banyak entitas atau tidak satupun entitas dari himpunan entitas kuliah.

2.14 Visual Basic for Application

Visual Basic for Application (VBA) adalah bahasa pemrograman untuk *Microsoft Office* dan aplikasi yang terkait (MacDonald, 2010). Dalam VBA juga dapat menggunakan *macro* untuk membuat objek dalam aplikasi sistem yang koheren. Dengan menggunakan *macro* dapat membantu menyelesaikan tugas – tugas kecil dalam aplikasi *microsoft*. VBA digunakan kerana beberapa alasan, diantaranya adalah:

1. Membuat aplikasi lebih mudah dipertahankan.
2. Dapat membuat fungsi sesuai keinginan.
3. Membuat atau memanipulasi objek.
4. Melakukan sistem berbasis pada tindakan.

2.15 Microsoft Access

Microsoft Access merupakan program manajemen *database* yang relatif mudah untuk dipelajari karena dalam *Microsoft Access* diberi banyak sekali fasilitas yang membantu pemakai (MacDonald, 2010). Selain itu *Wizard* atau tuntunan tahap demi tahap disediakan oleh *Access* untuk merancang *tabel*, *query*, *form* dan *report* dengan cepat dan mudah. Dan tentu saja keuntungan utama dari penggunaan *Microsoft Access* adalah kemampuan integrasinya yang optimal dengan aplikasi *software* lainnya (terutama *office*). Keunggulan ini disebabkan oleh suatu sarana penunjang yaitu *Visual Basic for Application*. Berikut ini diberikan beberapa istilah yang digunakan dalam *Microsoft Access*, yaitu:

1. *Table*, merupakan komponen utama dari sebuah *database* sekaligus obyek pertama yang harus dibuat. Tabel yang dibuat dapat berjumlah satu atau lebih sesuai dengan kebutuhan.
2. *Queries*, bagian ini digunakan untuk mengatur data mana saja dari suatu tabel yang perlu ditampilkan. Dalam *query* ini juga dapat diatur kriteria atau syarat penampilan suatu data serta bagaimana data tersebut diurutkan.
3. *Form*, bagian ini digunakan untuk memasukan data ke dalam tabel.
4. *Report*, bagian ini digunakan untuk membuat laporan dari suatu data yang telah diolah menjadi informasi. Dari *report* dapat langsung dibuat *print out* informasi yang telah diolah.

2.16 Fuzzy Logic Toolbox MATLAB

MATLAB merupakan salah satu bahasa pemrograman dengan kinerja tinggi untuk komputasi teknik yang mengintegrasikan komputasi, visualisasi, dan pemrograman dalam lingkungan yang sederhana (Sivandam, 2007). Pada MATLAB terdapat tool untuk membuat sistem inferensi *fuzzy* (FIS) yang dinamakan *Fuzzy Logic Toolbox*. *Toolbox* ini dibuat dengan menggunakan tool grafis atau dengan fungsi *command line* serta bisa juga dengan cara meng-*generate* secara otomatis menggunakan teknik *neuro-fuzzy* elustering atau adaptif. Sistem ini memiliki akses terhadap *software* Simulink maka dapat dengan mudah menguji sistem *fuzzy* pada derah simulasi diagram blok.

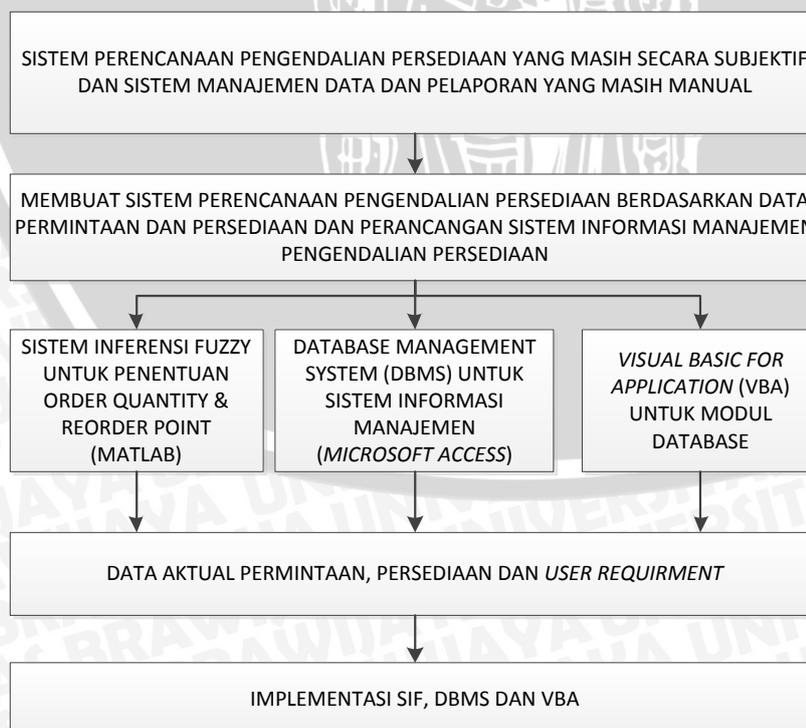
2.17 Kerangka Pikir

Sugiyono (2010) mengemukakan bahwa, kerangka berfikir merupakan model konseptual tentang bagaimana teori berhubungan dengan berbagai faktor yang telah diidentifikasi sebagai masalah yang penting. Dalam permasalahan yang ada pada penelitian

ini yaitu sistem pengendalian persediaan untuk menentukan *order quantity* and *reorder point* bahan baku aspal curah dan sistem manajemen data dan pelaporan yang masih manual sehingga diperlukan ketelitian, kedisiplinan dan konsentrasi yang tinggi dalam pengolahan datanya. Untuk mengatasi permasalahan yang ada, penelitian ini menggunakan beberapa metode yang dipilih yaitu:

1. Sistem inferensi *fuzzy* dengan metode mamdani pada *software* MATLAB digunakan untuk menentukan *order quantity* dan *reorder point* pada bahan baku aspal curah. Dalam proses ini, input dalam sistem inferensi *fuzzy* yaitu data permintaan dan persediaan. Metode ini dipilih dikarenakan dapat mengatasi unsur ketidakpastian dalam permintaan dan persediaan.
2. *Database Management System* (DBMS) dengan *Microsoft Access* untuk pengolahan data berbasis sistem informasi manajemen. Alasan dipilihnya karena dengan DBMS data akan disusun secara terstruktur dalam bentuk tabel-tabel yang saling terintegrasi sehingga untuk pengolahan data menjadi mudah.
3. *Visual Basic for Application* (VBA) berfungsi untuk pembuatan aplikasi pada komputer yang berisi algoritma untuk menjalankan aplikasi. Alasan dipilih adalah VBA merupakan bahasa pemrograman yang fleksibel dan dapat diintegrasikan dengan DBMS seperti Ms. Access.

Berikut ini merupakan kerangka pikir dari penelitian ini seperti terlihat pada Gambar 2.6.



Gambar 2.6 Kerangka Pikir Penelitian