**PROPOSAL**

# APLIKASI PERKIRAAN CURAH HUJAN KOTA KENDARI

## MENGGUNAKAN METODE *INTERVAL* *TYPE-2* *FUZZY* *LOGIC* *SYSTEM*



**REZA SANJAYA**

**E1E1 16 051**

# JURUSAN TEKNIK INFORMATIKA

# FAKULTAS TEKNIK

# UNIVERSITAS HALU OLEO

# KENDARI

**2020**

**DAFTAR ISI**

**HALAMAN JUDUL** .............................................................................................. i

**HALAMAN PENGESAHAN** ............................................................................... ii

**HALAMAN PERNYATAAN** .............................................................................. iv

**INTI SARI**............................................................................................................... v

**ABSTRACT** .......................................................................................................... vi

**HALAMAN PERSEMBAHAN**.......................................................................... vii

**KATA PENGANTAR** .......................................................................................... ix

**DAFTAR ISI** ........................................................................................................... x

**DAFTAR GAMBAR** ........................................................................................... xii

**DAFTAR TABEL**............................................................................................... xiv

**BAB I PENDAHULUAN** ....................................................................................... 1

1.1 Latar Belakang ....................................................................................... 1

1.2 Rumusan Masalah .................................................................................. 3

1.3 Batasan Masalah..................................................................................... 3

1.4 Tujuan Penelitian ................................................................................... 3

1.5 Manfaat Penelitian ................................................................................. 3

1.6 Sistematika Penulisan............................................................................. 4

1.7 Tinjauan Pustaka .................................................................................... 5

**BAB II LANDASAN TEORI** ................................................................................ 7

2.1 Metereologi dan Klimatologi ................................................................. 7

2.1.1 Cuaca .......................................................................................... 7

2.1.2 Iklim ........................................................................................... 8

2.1.3 Curah Hujan ............................................................................... 8

2.1.4 Penyinaran Matahari .................................................................. 8

2.1.5 Angin .......................................................................................... 8

2.2 Logika Fuzzy ........................................................................................ 9

2.2.1 Fungsi Keanggotaan *Fuzzy* ......................................................... 9

2.3 Sistem Logika *Fuzzy* Tipe 2 ................................................................. 11

2.3.1 Fungsi Keanggotaan *Upper* dan *Lower*.................................... 11

2.3.2 *Rules* ......................................................................................... 13

2.3.3 *Inference* *System*....................................................................... 13

2.3.4 *Type Reducer Karnik Mendel Method* ..................................... 13

2.3.5 Deffuzifikasi ............................................................................. 15

2.3.6 Contoh Perhitungan Manual .................................................... 16

2.4 Metode Pengujian Sistem ..................................................................... 19

2.5 *Flowchart* ............................................................................................. 20

2.6 *Unifield Modeling Language* (UML) ................................................... 21

2.6.1 Pengertian UML ....................................................................... 21

2.6.2 Diagram – Diagram UML ........................................................ 22

**BAB III METODOLOGI PENELITIAN** .......................................................... 28

3.1 Metode Pengumpulan Data .................................................................. 28

3.2 Metode Pengembangan Sistem ............................................................ 30

3.3 Waktu dan Tempat Penelitian .............................................................. 30

3.3.1 Waktu Penelitian ...................................................................... 30

3.3.2 Tempat Penelitian..................................................................... 31

## BAB IV ANALISIS DAN PERANCANGAN SISTEM .................................... 32

4.1 Analisis Sistem ..................................................................................... 32

4.1.1 Analisis Kebutuhan Fungsional ............................................... 32

4.1.2 Analisis Kebutuhan Nonfungsional ......................................... 32

4.2 Analisis Perancangan Sistem ............................................................... 33

4.2.1 Flowchart Sistem ...................................................................... 34

4.2.2 *Unified* *Modeling* *Language* (UML) ........................................ 38

4.2.3 *User Interface*........................................................................... 43

**DAFTAR PUSTAKA**

# DAFTAR GAMBAR

**Gambar 2.1** Kurva Segitiga .................................................................................. 10

**Gambar 2.2** Kurva Trapesium .............................................................................. 10

**Gambar 2.3** Sistem Logika *Fuzzy Type-2* ............................................................. 11

**Gambar 2.4** Fungsi Keanggotaan *Interval* *Fuzzy* Tipe 2 ...................................... 12

**Gambar 2.5** *Membership Function* Temperatur Udara ........................................ 16

**Gambar 2.6** *Membership* *Function* Kelembapan .................................................. 17

**Gambar 4.1** *Flowchart* Sistem .............................................................................. 34

**Gambar 4.2** *Flowchart* Fuzzifikasi ....................................................................... 35

**Gambar 4.3** *Flowchart* *Inference*.......................................................................... 36

**Gambar 4.4** *Flowchart* *Type* *Reducer*................................................................... 37

**Gambar 4.5** *Flowchart* Defuzzifikasi ................................................................... 38

**Gambar 4.6** *Use Case Diagram* ............................................................................ 39

**Gambar 4.7** *Activity Diagram Login* .................................................................... 39

**Gambar 4.8** *Activity Diagram Input* Data ............................................................. 40

**Gambar 4.9** *Activity* *Diagram* Prediksi................................................................. 40

**Gambar 4.10** *Sequence Diagram Login* ............................................................... 41

**Gambar 4.11** *Sequence* *Diagram* *Input* Data ........................................................ 42

**Gambar 4.12** *Sequence* *Diagram* Prediksi ............................................................ 42

**Gambar 4.13** *Class Diagram* ................................................................................ 43

**Gambar 4.14** Halaman *Login*................................................................................ 44

**Gambar 4.15** Halaman Beranda............................................................................ 44

**Gambar 4.16** Halaman *Input* ................................................................................ 45

**Gambar 4.17** Halaman Prediksi ............................................................................ 46

# DAFTAR TABEL

**Tabel 2.1** *Inference* ............................................................................................... 18

**Tabel 2.2** *Hasil Perhitungan* 𝑦𝑙 *dan* 𝑦𝑟 *................................................................* 19

**Tabel 2.3** Defuzzifikasi ........................................................................................ 19

**Tabel 2.4** Simbol *Flowchart* ................................................................................. 20

**Tabel 2.5** Simbol *Use* *Case* *Diagram* ................................................................... 23

**Tabel 2.6** Simbol *Activity* *Diagram* ...................................................................... 24

**Tabel 2.7** Simbol *Class* *Diagram* ......................................................................... 25

**Tabel 2.8** Simbol *Sequence* *Diagram* ................................................................... 26

**Tabel 3.1** Sampel Data Meteorologi ..................................................................... 28

**Tabel 3.2.** Fase Pengembangan *Software* ............................................................. 30

**Tabel 3.3** Tabel Waktu Penelitian ........................................................................ 31

**Tabel 4.1** Spesifikasi Minimum Perangkat Keras ................................................ 33

**Tabel 4.2** Spesifikasi Perangkat Lunak ................................................................ 33

**Tabel 5.1** Sampel Data Meteorologi yang dinormalisasi ..................................... 48

**Tabel 5.2** Sampel Data Meteorologi Harian ......................................................... 49

**Tabel 5.3** Kategori Intensitas Curah Hujan .......................................................... 58

**Tabel 5.4** *Inference* ............................................................................................... 59

**Tabel 5.5** *Type* *Reducer* ........................................................................................ 59

**Tabel 5.6** Deffuzifikasi ......................................................................................... 59

**Tabel 5.7** Perbandingan Data Aktual dan Prediksi Curah Hujan ......................... 60

**Tabel 5.8** Hasil Uji Coba RMSE dan *Accuracy* ................................................... 61

# BAB I

# PENDAHULUAN

## 1.1 Latar Belakang

Cuaca adalah keadaan atmosfer pada setiap waktu kesatuan, menurut Gibbs dan hasil Konferensi Iklim Dunia (Soerjadi & Swarinoto, 2010). Kondisi cuaca merupakan hal penting yang perlu dipelajari karena cuaca di suatu daerah menentukan rangkaian aktifitas manusia. Sebagai contoh, informasi iklim dan klasifikasinya banyak menjadi acuan untuk bidang pertanian, transportasi, dan pariwisata seperti: pelayaran, penerbangan, dan masa pola tanam. Beberapa penelitian juga menyebutkan bahwa iklim mempengaruhi kondisi keadaan ekonomi di suatu daerah (Rizky dkk., 2012). Sehingga informasi tentang cuaca sangat dibutuhkan karena dapat mempengaruhi aktivitas masyarakat.

Hujan berperan penting bagi kehidupan. Curah hujan yang turun pada suatu wilayah dapat diprediksi namun tidak dapat ditentukan secara pasti, menggunakan data historis masa lampau untuk memprediksi besarnya curah hujan yang akan datang (Muflih dkk., 2019), namun dengan adanya sistem atau aplikasi yang memprediksi curah hujan dapat dijadikan sebuah acuan atau pendukung keputusan sebelum melakukan aktivitas, sehingga masyarakat bisa memperkirakan kegiatan – kegiatan yang akan dilakukan untuk kedepannya.

Curah hujan dapat dipengaruhi oleh beberapa faktor diantaranya adalah temperatur udara, kelembapan, kecepatan angin, tekanan udara, total lapisan awan, dan lama penyinaran matahari.

Logika *fuzzy* merupakan salah satu cabang dari *Artificial Intelligence* dan juga merupakan modifikasi dari teori himpunan dimana setiap anggota dari himpunan ini memiliki derajat keanggotaan yang nilainya kontinu yaitu 0 sampai 1. Logika *fuzzy* merupakan metodologi sistem kontrol untuk memecahkan masalah yang bisa diimplementasikan pada sistem, baik sistem yang sederhana, sistem kecil, *embedded system*, jaringan PC, *multichannel* atau *workstation* berbasis akuisisi data

dan sistem kontrol. Metodologi ini dapat diterapkan pada perangkat keras, perangkat lunak ataupun kombinasi dari keduanya (Mandey dkk., 2017).

Teori *Fuzzy* pertama kali dikenalkan oleh Lotfi Zadeh pada tahun 1965 di Califomia University (Nurohmah, 2015). *Type-1 fuzzy logic system* terdiri dari beberapa komponen yaitu *fuzzification*, *rule base*, *inference system* dan *defuzzification*. Walaupun *Type-1 fuzzy set* dimaksudkan untuk merepresentasikan ketidakpastian, penelitian telah menunjukkan *Type-1 fuzzy set* memiliki keterbatasan dalam merepresentasikan dan memodelkan ketidakpastian dan meminimalisir pengaruhnya. Hal ini karena nilai keanggotaan dalam *Type-1 fuzzy set* adalah nilai *crisp* (Wu, 2014)*.*

Namun seiring dengan perkembangan teknologi saat ini, logika *fuzzy* mengalami perkembangan juga, baik dari segi *inference system* yang memiliki beberapa macam metode seperti tsukamoto, mamdani, tsugeno dan masih banyak lagi. Hingga pada saat ini *type-1 fuzzy logic system* berkembang menjadi *type-2 fuzzy logic system* yang dimana komponen yaitu *fuzzification*, *rule base*, *inference system*, *type reduction* dan *defuzzification.* Penggunaan *Type-2 fuzzy set* dilakukan untuk memodelkan ketidakpastian secara lebih jauh. Berbagai penelitian menunjukkan bahwa hasil *type-2 fuzzy logic system* lebih baik dari *type-1 fuzzy logic system*. Tutorial dan artikel yang semakin banyak mengenai *type-2 fuzzy logic system* juga mengakibatkan jumlah penelitian mengenai *type-2 fuzzy logic system* semakin banyak. Walaupun biaya komputasi *type-2 fuzzy logic system* lebih besar dari *type-1 fuzzy logic system*, kemajuan kemampuan komputasi perangkat keras telah mengakomodasi kebutuhan komputasi tersebut (Sajiah & Setiawan, 2016).

*Type-2 fuzzy logic system* memiliki ketidakpastian yang lebih jauh dibandingkan *type-1 fuzzy logic system*. *Type-2 fuzzy logic system* juga memiliki fungsi keanggotaan dan *type reducer* yang perlu diperhatikan karena memiliki potensi yang lebih baik.

Berdasarkan latar belakang yang telah diuraikan diatas, maka penulis mengambil topik penelitian dengan judul “**APLIKASI PERKIRAAN CURAH HUJAN KOTA KENDARI MENGGUNAKAN METODE *INTERVAL TYPE2 FUZZY LOGIC SYSTEM***”.

## 1.2 Rumusan Masalah

Rumusan masalah dalam penelitian ini adalah bagaimana efektifitas metode *Interval Type-2 Fuzzy Logic System* dalam memprediksi curah hujan di Kota Kendari.

## 1.3 Batasan Masalah

Dalam menyelesaikan penelitian ini batasan masalah yang menjadi fokus utama antara lain:

1. Sistem atau aplikasi perkiraan curah hujan ini hanya dapat dijadikan acuan saja.
2. Metode yang digunakan adalah *Interval Type-2 Fuzzy Logic System*
3. Data yang digunakan data meteorologi pada tahun 2019 Badan Meteorologi, Klimatologi dan Geofisika (BMKG) Stasiun Meteorologi Maritim Kendari
4. Data yang diolah pada sistem ini hanya data yang memiliki curah hujan saja.
5. *Type reducer* yang digunakan adalah *Karnik Mendel-Method*
6. Variabel yang digunakan hanya 5, yaitu temperatur, kelembapan, lama penyinaran matahari, kecepatan angin dan curah hujan.

## 1.4 Tujuan Penelitian

Adapun tujuan dari penelitian ini adalah bagaimana efektifitas metode *Interval Type-2 Fuzzy Logic System* dalam memprediksi curah hujan di Kota Kendari.

## 1.5 Manfaat Penelitian

Adapun manfaat dari penelitian ini adalah sebagai berikut:

1. Dapat memprediksi curah hujan di Kota Kendari dengan lebih akurat.
2. Dapat memberitahu informasi tentang curah hujan dengan lebih mudah

## 1.6 Sistematika Penulisan

Laporan Penelitian ini nantinya disusun dengan sistematika penulisan sebagai berikut:

## BAB I PENDAHULUAN

Bab ini berisi tentang latar belakang penelitian, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, sistematika penulisan laporan dan tinjauan pustaka.

## BAB II LANDASAN TEORI

Bab ini memuat pengertian-pengertian dan teori-teori yang berkaitan dalam pembuatan sistem atau aplikasi prediksi curah hujan di Kota Kendari, yaitu penjelasan tentang *Interval Type-2 Fuzzy Logic System*, Meteorologi dan Klimatologi*.*

## BAB III METODOLOGI PENELITIAN

Bagian ini memuat prosedur dan pengumpulan data, prosedur pengembangan perangkat lunak serta waktu dan tempat penelitian.

## BAB IV ANALISIS DAN PERANCANGAN

Bab ini akan membahas tentang analisis dan perancangan dari aplikasi yang akan dibuat, dengan menggunakan desain Unified Modelling Language (UML) serta desain user interface.

## BAB V IMPLEMENTASI DAN PEMBAHASAN

Dalam bab ini akan dikaji mengenai implementasi hasil perancangan aplikasi yang dibuat serta melakukan pengujian terhadap sistem.

## BAB VI PENUTUP

Bab ini berisi kesimpulan dari program yang telah dibuat serta saran yang diperlukan untuk pengembangan program berikutnya.

### 1.7 Tinjauan Pustaka

Hasil penelitian sebelumnya yang bertema *Fuzzy Logic System* yang menjadi acuan dan referensi penyusunan dalam menyusun tugas akhir ini. Salah satu penelitian sebelumnya yang mendasari penelitian ini adalah penelitian yang dilakukan oleh Mahasiswa Institut Teknologi Sepuluh Nopember (ITS) yaitu Dynes Rizky Navianti (2012) dengan judul penelitian “Penerapan *Fuzzy Inference System* pada Prediksi Curah Hujan di Surabaya Utara”, hasil penelitian ini menunjukkan model yang diperoleh memiliki rata-rata akurasi sebesar 77,68% terhadap data yang dianalisis. Hasil akurasi tertinggi pada model musim penghujan = 70,96% pada bulan Januari, sedangkan akurasi pada model musim kemarau = 100% pada bulan Agustus.

Selanjutnya ada penelitian yang dilakukan oleh Mahasiswa S2 Universitas

Gadjah Mada (UGM) yaitu Adha Mashur Sajiah (2016)dengan judul penelitian “*Interval Type-2 Fuzzy Logic System* untuk *Diagnosis Coronary Artery Disease*” hasil penelitian ini struktur *Interval Type-2 Fuzzy Logic System* untuk diagnosis *Coronary Artery Disease* telah dijabarkan. Sistem telah dikembangkan untuk deteksi dini penyakit CAD. Ketidakpastian dalam diagnosis CAD direpresentasikan dengan IT2 FLS. *Fuzzy* *rule* dihasilkan secara otomatis dengan bantuan JRip Weka. Terakhir sistem diujikan terhadap *dataset heart statlog*. Hasil sistem masih memiliki potensi untuk dikembangkan lebih jauh, hal ini tidak terlepas dari banyaknya opsi pemilihan metode-metode pengembangan komponen IT2 FLS.

Penelitian lainnya adalah Penelitian yang dilakukan oleh Mahasiswa Universitas Sam Ratulangi yaitu Fleydi Novisnky Mandey (2017) dengan judul penelitian “Pemodelan Sistem Prediksi Intensitas Curah Hujan di Kota Manado Dengan Menggunakan Kontrol Logika *Fuzzy*” hasil penelitian menunjukkan bahwa pada suhu 24,2° kelembapan 93% tekanan 1009 millibar dan kecepatan angin 2 knot maka prediksi intensitas curah hujan yaitu 70,1 mm atau termasuk dalam kategori hujan lebat sedangkan data curah hujan dari BMKG yang digunakan sebagai perdandingan prediksi ini yaitu hanya menunjukkan nilai 84 untuk kondisi tersebut atau berarti bahwa masih memiliki selisih sebesar 13,9. Kemudian untuk pengolahan data pada hari lainnya setelah dilakukan pengujian maka didapatkan hasil berikut pada suhu 25,4° kelembapan 89% tekanan 1014,8 milibar dan kecepatan angin 2 knot, prediksi intensitas curah hujan yaitu 69,8 mm atau termasuk dalam kategori hujan lebat sedangkan data curah hujan dari BMKG hanya menunjuk angka 63 mm atau memiliki selisih sebesar 6,8. Berdasarkan uji logika *fuzzy* untuk menghitung keakuratan penelitian maka dihitung nilai error rata-rata dalam RMSE yaitu 11,411%. Hasil ini menunjukkan bahwa nilai *error* yang terdapat dalam prediksi ini cukup besar yang berarti bahwa prediksi intensitas curah hujan dengan pemodelan kontrol logika *fuzzy* ini belum dapat digunakan. Namun perbedaan hasil prediksi tersebut yang mempunyai perbedaan yang masih sangat jauh dengan data pembandingnya tidak berarti bahwa pemodelan menggunakan kontrol logika *fuzzy* ini tidak bisa digunakan hanya saja jika digunakan untuk memprediksi intensitas curah hujan seperti yang telah dilakukan perlu dilakukan peninjauan kembali untuk data-data yang digunakan dalam prediksi ini yaitu datadata dari BMKG yang sepertinya masih diragukan keakuratannya.

Kesimpulannya penelitian untuk memprediksi curah hujan sudah banyak dilakukan namun kebanyakan masih menggunakan metode *type-1 fuzzy logic system*, sehingga peneliti mengusulkan menggunakan metode *Interval Type-2 Fuzzy Logic System* yang memiliki potensi lebih baik dan akurat.

# BAB II

# LANDASAN TEORI

## 2.1 Meteorologi dan Klimatologi

Klimatologi, seperti halnya meteorologi, yaitu ilmu tentang atmosfer. Meteorologi menitikberatkan pada atmosfer, sedangkan klimatologi terutama berkaitan dengan hasil proses tersebut (Sudiarta, 2013)**.**

Meteorologi berasal dari kata Yunani, yaitu meteoros, yang artinya benda yang ada di dalam udara dan logos artinya ilmu atau kajian. Jadi *meteorology*

didefinisikan sebagai ilmu yang mempelajari proses fisis dan gejala cuaca yang terjadi di dalam atmosfer pada lapisan bawah, yaitu troposfer. Kajian *meteorology* diperlukan dalam pembangunan irigasi, objek pariwisata, tempat peristirahatan, perkebunan, perikanan, lapangan terbang, pelayaran, proyek industri, dan sebagainya (Sudiarta, 2013).

Klimatologi berasal dari kombinasi dua kata Yunani, yaitu klima yang diartikan sebagai kemiringan (*slope*) bumi yang mengarah pada pengertian lintang tempat, dan logos yang diartikan sebagai ilmu. Jadi klomatologi didefinisikan sebagai yang mencari gambaran dan penjelasan sifat iklim, mengapa iklim di berbagai tempat di bumi berbeda, dan bagaimana kaitan antara iklim dan aktivitas manusia. Klimatologi dapat juga didefinisikan sebagai ilmu yang mempelajari jenis iklim dimuka bumi dan faktor penyebabnya. Karena klimatologi mencakup interpretasi dan koleksi data pengamatan maka ilmu ini memerlukan teknik statistik (Sudiarta, 2013).

### 2.1.1 Cuaca

Dari apa yang dikemukakan Gibbs dan hasil Konferensi Iklim Dunia dapat diambil sarinya bahwa "cuaca" adalah keadaan atmosfer pada setiap waktu kesatuan. Dengan demikian kita dapat mengatakan tentang cuaca saat ini, cuaca pukul 12, cuaca nanti sore, cuaca tanggal 17 Agustus 1945, cuaca besok pagi, cuaca minggu ini, cuaca minggu depan, cuaca bulan depan, dan seterusnya apabila waktuwaktu tersebut diartikan sebagai satu satuan waktu. Jadi cuaca menyatakan keadaan

yang berlangsung pada saat atau selama waktu kesatuan (Soerjadi & Swarinoto, 2010).

### 2.1.2 Iklim

Iklim mengandung pengertian kebiasaan cuaca yang terjadi di suatu tempat atau daerah dan juga memberi pengertian bahwa iklim adalah ciri kecuacaan suatu tempat atau daerah, dan bukan cuaca rata-rata. Oleh karena itu, tidak mungkin kita mengatakan iklim hari ini, iklim besok pagi, iklim minggu depan dan seterusnya. Tetapi kita dapat mengatakan iklim zaman dahulu, iklim selama ini, iklim di waktu mendatang. Jadi, iklim berkaitan dengan periode waktu panjang tidak tentu (Soerjadi & Swarinoto, 2010).

### 2.1.3 Curah Hujan

Jumlah curah hujan dicatat dalam inci atau millimeter (1 inci = 25,4 mm). Di daerah tropis hujannya lebih lebat daripada di daerah lintang tinggi. Garis yang menghubungkan titik-titik dengan curah hujan sama selama periode tertentu disebut isohyets (Soerjadi & Swarinoto, 2010).

### 2.1.4 Penyinaran Matahari

Penyinaran matahari merupakan aktivitas matahari yang memancarkan sinarnya ke Bumi. Aktivitas matahari yang memancarkan sinarnya di Bumi ini merupakan aktivitas alamiah karena matahari adalah sebuah bintang besar yang memiliki cahayanya sendiri, sementara Bumi merupakan planet yang berada di sekitarnya dan mengorbit kepadanya (Fatma, 2018).

### 2.1.5 Angin

Angin ialah gerak udara yang sejajar dengan permukaan bumi. Udara bergerak dari daerah bertekanan tinggi ke daerah bertekanan rendah. Angin diberi nama sesuai dengan dari arah mana angin datang, misalnya angin timur adalah angin yang dari arah timur, angin laut adalah angin yang bertiup dari laut ke darat, dan angin lembah adalah angin yang datang dari lembah menaiki pegunungan (Sudiarta, 2013).

## 2.2 Logika *Fuzzy*

Teori *Fuzzy* pertama kali dikenalkan oleh Lotfi Zadeh pada tahun 1965 di Califomia *University*. Sekarang ini, *fuzzy* banyak digunakan untuk berbagai macam penelitian untuk mengatasi kekurangan yang dipunyai oleh teori *Boolean*. Pada teori *Boolean* hanya dikenal dua keadaan yaitu 0 dan 1, besar dan kecil, panjang atau pendek Sehingga hanya dapat diketahui nilai ekstrim dari suatu keadaan. Padahal terkadang dibutuhkan suatu keadaan yang betul – betul mempresentasikan keadaan nyata. Teori *fuzzy* memberikan solusi dari masalah ini. Teori *fuzzy* mendefinisikan keadaan dalam himpunan yang kontinyu dari nol sampai satu (Nurohmah, 2015).

### 2.2.1 Fungsi Keanggotaan *Fuzzy*

Pemetaan atau *mapping* data *input* kedalam derajat keanggotaan yaitu berupa sebuah himpunan *fuzzy*, disebut Fuzzifikasi atau fungsi keanggotaan (*membership function*).

## 1. Fungsi Keanggotaan Segitiga (*Triangular Membership Function*)

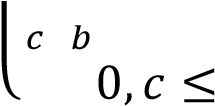
*Triangular Membership Function* memiliki tiga parameter sebagai acuan {*a*, *b*, *c*} seperti pada persamaan 2.1.

𝑎

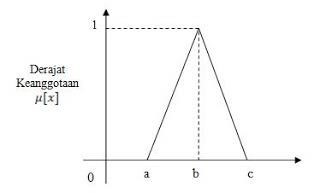
𝑏

𝑦=𝑡𝑟𝑖𝑎𝑛𝑔𝑙𝑒(𝑥;𝑎,𝑏,𝑐) 𝑏 𝑎  (2.1)

,𝑏 𝑥 𝑐

𝑥

Parameter {*a*, *b*, *c*} dengan *a* < *b* < *c* menentukan koordinat *x* dari tiga sudut fungsi keanggotaan segitiga.



## Gambar 2.1 Kurva Segitiga (Nurohmah, 2015)

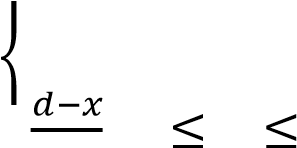
### 2. Fungsi Keanggotaan *Trapezium* (*Trapezoidal Membership Function*)

*Trapezoidal Membership Function* didefinisikan dengan empat parameter {*a*, *b*, *c*, *d*} seperti terlihat dipersamaan 2.2.

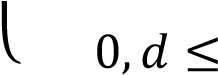
𝑎



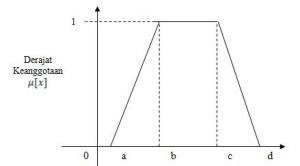
𝑏𝑎 𝑏

𝑦=𝑡𝑟𝑎𝑝𝑒𝑧𝑜𝑖𝑑(𝑥;𝑎,𝑏,𝑐,𝑑)𝑐 (2.2) ,𝑐 𝑥 𝑑

𝑐

𝑥

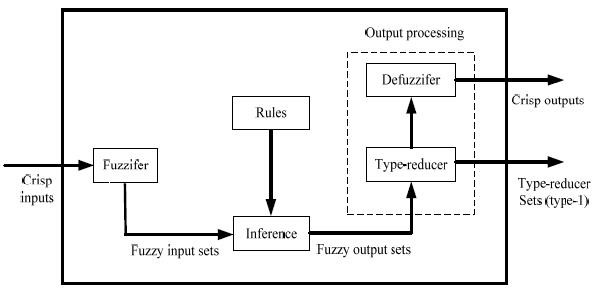
Parameter {*a*, *b*, *c*, *d*} dengan *a* < *b* ≤ *c* < d menentukan koordinat *x* dari empat sudut fungsi keanggotaan trapesium.



## Gambar 2.2 Kurva Trapesium (Nurohmah, 2015)

### 2.3 Sistem Logika *Fuzzy* Tipe 2

*Fuzzy type-2* merupakan bentuk pengembangan dari *fuzzy type-1* dikarenakan *fuzzy type-1* menggunakan *i nput variable* dengan nilai yang pasti dimana hal itu bertentangan dengan kaidah *fuzzy* itu sendiri, maka muncullah konsep pengembangan *fuzzy* yaitu *fuzzy type-2* yang mulai di perkenalkan pada tahun 1960 oleh Zadeh (Saiful, 2018).  Perbedaan struktur utama adalah adanya blok *type reducer* sebelum blok *defuzzifier*. Seperti namanya, *type-reducer* bertugas melakukan transformasi himpunan *fuzzy type-2* menjadi himpunan *fuzzy type-1* sebelum dilakukan defuzzifikasi. Metode *type-reduction* yang umum adalah: *centroid*, *center of sums*, *height*, *modified height* dan *center of sets*. Penjelasan tentang algoritma *fuzzy type2* dapat di lihat pada gambar di bawah (Sajiah & Setiawan, 2016).



## Gambar 2.3 Sistem Logika *Fuzzy Type-2* (Xu dkk., 2016)

### 2.3.1 Fungsi Keanggotaan *Upper* dan *Lower*

*Footprint of uncertainty* (*FOU*) adalah daerah terbatas yang memuat ketidakpastian derajat keanggotaan primer dari fungsi keanggotaan tipe 2. *Upper* dan *lower membership function* adalah dua buah fungsi keanggotaan tipe 1 yang dibatasi *footprint of uncertainty* pada fungsi keanggotaan interval tipe 2. *Upper* *membership function* adalah himpunan *fuzzy* dari fungsi keanggotaan bagian atas atau yang memiliki derajat keanggotaan tertinggi dalam *FOU*. Sedangkan lower *membership function* adalah sebaliknya.

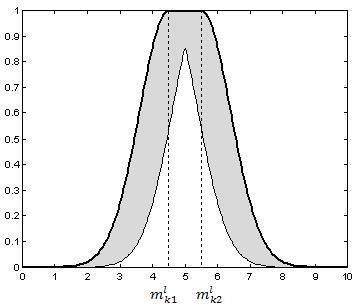
Gambar 2.8 adalah salah satu contoh bentuk *membership function* himpunan *fuzzy* bertipe 2 *interval* (fungsi *Gauss* dengan pergeseran titik tengah), daerah arsiran adalah merupakan *FOU*-nya. Fungsi keanggotaan dalam Gambar 2.8 didefinisikan sebagai berikut:

𝑙 2

𝜇𝑘𝑙 (𝑥𝑘)=𝑒𝑥𝑝(−12(𝑥𝑘−𝜎𝑚𝑘𝑙 𝑘1)), 𝑚𝑙𝑘 ∈[𝑚𝑘𝑙1,𝑚𝑘𝑙2] (2.3)

dengan k =(1,…,p) adalah jumlah *antecedent*, l=(1,…,M) adalah jumlah kaidah

(*rule*).



## Gambar 2.4 Fungsi Keanggotaan *Interval Fuzzy* Tipe 2

Berdasarkan diagram diatas, garis tebal adalah *upper Membership Function*, sedangkan garis tipis adalah *lower Membership Function* (Sari & Wahyunggoro, 2016)*.*

### 2.3.2 *Rules*

Aturan dalam *fuzzy* tidak ada perbedaan secara mendasar antara tipe 1 dan tipe 2 (begitu juga untuk derajat yang lebih tinggi tipe *n*). Seperti disebutkan sebelumnya perbedaan tipe 1 dan tipe 2 adalah sifat fungsi keanggotaan. Hal ini tidak mempengaruhi ketika membentuk basis aturan. Sehingga sebuah *Fuzzy Logic System* tipe 2 dengan *p input* *x1* ∈ *X1*, …, *xp* ∈ *Xp* ≡ *X* dengan satu *output* *y* ∈ *Y*.

Aturan dalam *Fuzzy Logic System* tipe 2 untuk aturan ke l dari *M* aturan adalah

𝑅𝑙 ∶𝐼𝐹 𝑥1 𝑖𝑠 𝐴̃1𝑙 𝑎𝑛𝑑… 𝑥𝑝 𝑖𝑠 𝐴̃𝑙𝑝 𝑇𝐻𝐸𝑁 𝑦 𝑖𝑠 𝑌̃𝑙 𝑙=1,…,𝑀 (2.4)

### 2.3.3 *Inference System*

Mesin inferensi menggabungkan aturan dan melakukan pemetaan *input* himpunan *fuzzy* tipe 2 ke *output* himpunan *fuzzy* tipe 2. Dependensi ini adalah relasi *input*/*output* yang ditunjukkan pada Gambar 2.4, yang menyimpan antara himpunan *fuzzy* tipe 2 yang mengaktifkan aturan dalam mesin inferensi dan himpunan *fuzzy* tipe 2 pada keluaran mesin tersebut.

Pada *Fuzzy Logic System*, digunakan himpunan *fuzzy interval* tipe 2 dan irisan dengan *product t-norm*, sehingga hasil dari operasi *input* dan *antecedent*, yang berada di *firing set* ∏𝑝𝑖=1𝜇𝐹̃𝑖(𝑥𝑖 ≡𝐹𝑙(𝑥)), adalah himpunan *interval* tipe 2.

𝑙𝑙

𝐹𝑙[𝑓𝑙,𝑓] (2.5) dengan

(

𝑥

)

=

[

𝑓

𝑙

(

𝑥

)

,

𝑓

(

𝑥

)

]

≡

𝑓𝑙×𝜇𝐹̃𝑝𝑙(𝑥𝑝) (2.6)

𝑙

𝑓(𝑥)=𝜇𝐹̃1𝑙(𝑥1)×⋯×𝜇𝐹̃𝑝𝑙(𝑥𝑝) (2.7)

### 2.3.4 *Type Reducer Karnik Mendel Method*

*Type reducer* menghasilkan *output* himpunan *fuzzy* tipe 1 yang selanjutnya menjadi *input* *defuzzifier* untuk memberikan output nilai *crisp*. Banyak metode *type* *reducer* yang dapat digunakan, salah satunya *Karnik* *Mendel* *Method.*

̅ ̅ 𝑁𝑛=𝐿+1𝑓𝑛𝑦𝑛

𝑦𝑙 = 𝑘∈[1,𝑁−1] ∑𝑘𝑛=1𝑓̅𝑛+∑𝑁𝑛=𝑘+1𝑓𝑛 ≡ ∑𝐿𝑛=1𝑓̅𝑛+∑𝑁𝑛=𝐿+1𝑓𝑛 (2.8)

min

∑

𝑓

𝑛

𝑦

𝑛

𝑘

𝑛

=

1

+

∑

𝑓

𝑛

𝑁

𝑛

=

𝑘

+

1

𝑦

𝑛

∑

𝑓

𝑛

𝑦

𝑛

𝐿

𝑛

=

1

+

∑

|  |  |
| --- | --- |
| 𝑦𝑟 = 𝑘∈[1,𝑁−1] 𝑛∑=1𝑘𝑛=1𝑓𝑛+∑𝑛𝑛𝑁==𝑘𝑘++11𝑓̅𝑛 ≡ 𝑛∑=1𝑅𝑛=1𝑓𝑛+∑𝑛𝑁𝑛==𝑅𝑅++11𝑓̅𝑛  dimana *switch* *point* *L* dan *R* ditentukan dengan | (2.9) |
| 𝑦𝐿 ≤𝑦𝑙 ≤𝑦𝐿+1 | (2.10) |

max ∑𝑘 𝑓𝑛𝑦̅𝑛+∑𝑁 𝑓̅𝑛𝑦̅𝑛 ∑𝑅 𝑓𝑛𝑦̅𝑛+∑𝑁 𝑓̅𝑛𝑦̅𝑛

𝑦̅𝑅 ≤𝑦𝑟 ≤𝑦̅𝑅+1 (2.11) serta {𝑦̅𝑛} dan {𝑦𝑛} telah diurutkan dengan urutan meningkat.

𝑦𝑙 dan 𝑦𝑟 dihitung dengan menggunakan algoritma Karnik-Mendel seperti berikut.

1. Algoritma Karnik Mendel untuk menghitung 𝑦𝑙

1. Urutkan 𝑦𝑛 (𝑛=1,2,…,𝑁) dalam urutan meningkat dan 𝑦𝑛 yang telah

diurut tetap mempunyai nama yang sama, tapi sekarang 𝑦1 ≤𝑦2 ≤⋯≤𝑦𝑛.

Cocokkan bobot 𝐹𝑛(𝑥′) dengan 𝑦𝑛 yang bersesuaian dan nomori ulang sehingga indeks bersesuaian dengan 𝑦𝑛 yang telah dinomori ulang.

1. Inisialisasi 𝑓𝑛 dengan mengatur

𝑓𝑛+𝑓̅𝑛

𝑓𝑛 = 𝑛=1,2,…,𝑁 (2.12)

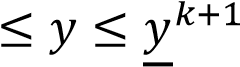
2 kemudian hitung

∑𝑁𝑛=1𝑦𝑛𝑓𝑛

𝑦= ∑𝑁 𝑓𝑛 (2.13)

𝑛=1

1. Mencari *switch* *point k* (1≤𝑘≤𝑁−1) sehingga

𝑦𝑘 (2.14)

1. Atur

# 𝑓̅𝑛, 𝑛≤𝑘

𝑓𝑛 ={𝑓𝑛, 𝑛 >𝑘 (2.15)

dan hitung

′ = ∑∑𝑁𝑛=𝑁1𝑦𝑓𝑛𝑛𝑓𝑛 (2.16)

𝑦

𝑛=1

1. Periksa jika 𝑦′ =𝑦. Jika ya, berhenti dan set 𝑦𝑙 =𝑦 dan 𝐿=𝑘. Jika tidak, menuju ke langkah ke (f)
2. Set 𝑦=𝑦′ dan menuju langkan (c).

2. Algoritma Karnik Mendel untuk menghitung 𝑦𝑟

1. Urutkan 𝑦̅𝑛 (𝑛=1,2,…,𝑁) dalam urutan meningkat dan 𝑦̅𝑛 yang telah diurut tetap mempunyai nama yang sama, tapi sekarang 𝑦̅1 ≤𝑦̅2 ≤⋯≤𝑦̅𝑛. Cocokkan bobot 𝐹𝑛(𝑥′) dengan 𝑦̅𝑛 yang bersesuaian dan nomori ulang sehingga indeks bersesuaian dengan 𝑦̅𝑛 yang telah dinomori ulang.
2. Inisialisasi 𝑓𝑛 dengan mengatur

𝑓𝑛+𝑓̅𝑛

𝑓𝑛 = 𝑛=1,2,…,𝑁 (2.17)

2 kemudian hitung

𝑦= ∑∑𝑁𝑛=𝑁1𝑦̅𝑓𝑛𝑛𝑓𝑛 (2.18)

𝑛=1

1. Mencari *switch* *point k* (1≤𝑘≤𝑁−1) sehingga

𝑦̅𝑘 ≤𝑦≤𝑦̅𝑘+1 (2.19)

1. Atur

𝑓𝑛, 𝑛≤𝑘

𝑓𝑛 ={ (2.20)

𝑓̅𝑛, 𝑛 >𝑘

dan hitung

𝑦′ = ∑∑𝑁𝑛=𝑁1𝑦̅𝑓𝑛𝑛𝑓𝑛 (2.21)

𝑛=1

1. Periksa jika 𝑦′ =𝑦. Jika ya, berhenti dan set 𝑦𝑟 =𝑦 dan 𝑅=𝑟. Jika tidak, menuju ke langkah ke (f)
2. Set 𝑦=𝑦′ dan menuju langkan (c).

## 2.3.5 Defuzzifikasi

Setelah mendapatkan himpunan interval *output* dari *type reducer*, untuk mendapatkan nilai *crisp output* keluaran akhir sistem digunakan rata-rata dari nilai *yl* dan *yr*. Nilai tersebut dapat dihitung dengan persamaan 2.36 berikut.

𝑦(𝑥)= 𝑦𝑙+𝑦𝑟 (2.22)

2

**2.3.6 Contoh Perhitungan Manual**

## 1. Fuzzifikasi

Pada proses fuzzifikasi akan dilakukan proses pemetaan nilai crisp dari variabel yang telah ditentukan.

## a. Variabel Temperatur Udara

𝜇𝑠𝑒𝑗𝑢𝑘(𝑥)=𝑡𝑟𝑎𝑝𝑚𝑓(𝑥;25.7,25.7,27.5,28 1)

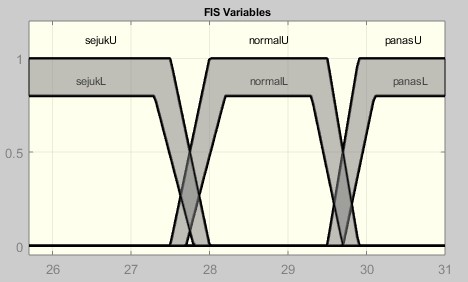
𝜇𝑠𝑒𝑗𝑢𝑘(𝑥)=𝑡𝑟𝑎𝑝𝑚𝑓(𝑥;25.7,25.7,27.3,27.8 0.8)

𝜇𝑛𝑜𝑟𝑚𝑎𝑙(𝑥)=𝑡𝑟𝑎𝑝𝑚𝑓(𝑥;27.5,28,29.5,29.9 1)

𝜇𝑛𝑜𝑟𝑚𝑎𝑙(𝑥)=𝑡𝑟𝑎𝑝𝑚𝑓(𝑥;27.7,28.2,29.3,29.7 0.8)

𝜇𝑝𝑎𝑛𝑎𝑠(𝑥)=𝑡𝑟𝑎𝑝𝑚𝑓(𝑥;29.5,29.9,31,31 1)

𝜇𝑝𝑎𝑛𝑎𝑠(𝑥)=𝑡𝑟𝑎𝑝𝑚𝑓(𝑥;29.7,30.1,31,31 0.8)



## Gambar 2.5 *Membership Function* Temperatur Udara

𝜇𝑠𝑒𝑗𝑢𝑘(27,3)=1

𝜇𝑠𝑒𝑗𝑢𝑘(27,3)=0,8

## b. Variabel Kelembapan

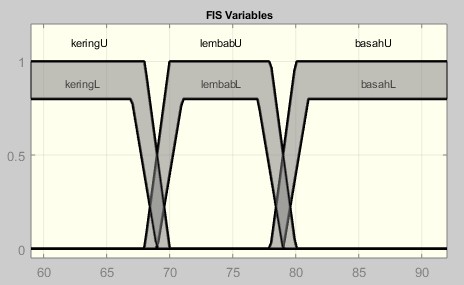
𝜇𝑘𝑒𝑟𝑖𝑛𝑔(𝑥)=𝑡𝑟𝑎𝑝𝑚𝑓(𝑥;59,59,68,70 1) 𝜇𝑘𝑒𝑟𝑖𝑛𝑔(𝑥)=𝑡𝑟𝑎𝑝𝑚𝑓(𝑥;59,59,67,69 0.8)

𝜇𝑙𝑒𝑚𝑏𝑎𝑏(𝑥)=𝑡𝑟𝑎𝑝𝑚𝑓(𝑥;68,70,78,80 1)

𝜇𝑙𝑒𝑚𝑏𝑎𝑏(𝑥)=𝑡𝑟𝑎𝑝𝑚𝑓(𝑥;69,71,77,79 0.8)

𝜇𝑏𝑎𝑠𝑎ℎ(𝑥)=𝑡𝑟𝑎𝑝𝑚𝑓(𝑥;78,80,92,92 1)

𝜇𝑏𝑎𝑠𝑎ℎ(𝑥)=𝑡𝑟𝑎𝑝𝑚𝑓(𝑥;79,81,92,92 0.8)



## Gambar 2.6 *Membership* *Function* Kelembapan

𝜇𝑏𝑎𝑠𝑎ℎ(84)=1

𝜇𝑏𝑎𝑠𝑎ℎ(84)=0,8

### 2. Rules

Adapun rules yang akan digunakan adalah sebagai berikut:

* IF Temperatur is **Sejuk** AND Kelembapan is **Kering** THEN Curah Hujan is **Ringan**
* IF Temperatur is **Sejuk** AND Kelembapan is **Lembab** THEN Curah Hujan is **Sedang**
* IF Temperatur is **Sejuk** AND Kelembapan is **Basah** THEN Curah Hujan is

## Sedang

* IF Temperatur is **Normal** AND Kelembapan is **Kering** THEN Curah Hujan is **Ringan**
* IF Temperatur is **Normal** AND Kelembapan is **Lembab** THEN Curah Hujan is **Ringan**
* IF Temperatur is **Normal** AND Kelembapan is **Basah** THEN Curah Hujan is **Sedang**
* IF Temperatur is **Panas** AND Kelembapan is **Kering** THEN Curah Hujan is **Ringan**
* IF Temperatur is **Panas** AND Kelembapan is **Lembab** THEN Curah Hujan is **Ringan**
* IF Temperatur is **Panas** AND Kelembapan is **Basah** THEN Curah Hujan is **Sedang**

### 3. *Inference*, *Type Reducer* dan Deffuzifikasi

Adapun pada tahap ini terdapat beberapa proses seperti *inference*, *type reducer* dan deffuzifikasi yang dapat dilihat pada Tabel 2.1;2.2;2.3.

**Tabel 2.1 *Inference***

|  |  |  |
| --- | --- | --- |
| Aturan  Ke | *Firing* *Interval* | Konsekuen |
| R3 | 3  [𝑓 ;𝑓3]= [𝜇𝑠𝑒𝑗𝑢𝑘(𝑇𝑈)×𝜇𝑏𝑎𝑠𝑎ℎ(𝐾);𝜇𝑠𝑒𝑗𝑢𝑘(𝑇𝑈)  ×𝜇𝑏𝑎𝑠𝑎ℎ(𝐾)]  =1 ×1 ;0,8 ×0,8  3  [𝑓 ;𝑓3]=1 ;0,64 | [𝑦̅3;𝑦3]  =[20;5] |

Berdasarkan *type* *reducer* yang digunakan, didapatkan *L* = 1 dan *R* = 1.

Sehingga nilai 𝑦𝑙 dan 𝑦𝑟 dapat dilihat pada Tabel 2.2.

## Tabel 2.2 Hasil Perhitungan 𝒚𝒍 dan 𝒚𝒓

|  |
| --- |
| 𝑓̅  𝑦𝑙 ̅ |
| 𝑓1𝑦̅1 0,64∗20  𝑦𝑟 |

## Tabel 2.3 Deffuzifikasi

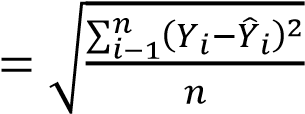
|  |
| --- |
| 𝑦𝑙 +𝑦𝑟 5+20  𝑦= = =12,5  2 2 |

Berdasarkan perhitungan yang telah dijelaskan pada Tabel 2.1;2.2;2.3 diperoleh nilai prediksi curah hujan sebesar 12.5 mm.

### 2.4 Metode Pengujian Sistem

Metode pengujian sistem yang digunakan yaitu dengan menghitung nilai *error* dari setiap proses prediksi. Ada beberapa parameter pengukuran nilai *error*, yaitu *Mean Square Error, Root Mean Square Error*, *Mean Absolute Error* dan masih banyak lagi.

Adapun metode pengujian untuk mencari nilai *error* pada penelitian ini adalah metode *Root Mean Square Error* (RMSE). *Root Mean Square Error* merupakan parameter yang digunakan untuk mengevaluasi nilai hasil pengujian terhadap nilai estimasi. Nilai RMSE adalah akar kuadrat dari nilai *Mean Square Error*. Nilai perbedaan disebut residual atau disebut juga nilai *error* hasil prediksi. RMSE termasuk teknik pengukuran yang mampu mengukur akurasi variabel tertentu dengan model yang berbeda dan dengan jumlah data yang sama. RMSE dapat dituliskan dengan persamaan 2.23. (Sari & Wahyunggoro, 2016).

𝑅𝑀𝑆𝐸 (2.23)

Keterangan:

𝑌=𝐷𝑎𝑡𝑎 𝑎𝑘𝑡𝑢𝑎𝑙

𝑌̂=𝐷𝑎𝑡𝑎 𝑝𝑟𝑒𝑑𝑖𝑘𝑠𝑖

𝑛=𝐽𝑢𝑚𝑙𝑎ℎ 𝑑𝑎𝑡𝑎

Kemudian setelah mendapat nilai RSME, akan dicari tingkat akurasi sesuai dengan persamaan 2.24.

𝐴𝑐𝑐𝑢𝑟𝑎𝑐𝑦=100−𝑅𝑀𝑆𝐸 (2.24)

### 2.5 *Flowchart*

*Flowchart* merupakan gambaran secara grafik dari langkah-langkah dan urutan prosedur suatu program. *Flowchart* menolong analis dan *programmer* untuk memecahkan masalah kedalam segmen-segmen yang lebih kecil dan menolong dalam menganalisis alternatif – alternatif lain dalam pengoperasian, untuk mendesain program dan merepresentasikan program. *Flowchart* biasanya mempermudah penyelesaian suatu masalah khususnya masalah yang perlu dipelajari dan dievaluasi lebih lanjut.

## Tabel 2.4 Simbol *Flowchart* (Kiswanto, 2016)

|  |  |
| --- | --- |
| **Simbol** | **Deskripsi** |
| *Terminator/ Terminal Points* | Awal atau akhir program |
| *Flow* | Arah aliran program, merepresentasikan alur kerja |
| *Preparation* | Inisialisasi / Pemberian Nilai Awal |
| *Process* | Proses / Pengolahan Data, Mempresentasikan Operasi |

## Tabel 2.4 Simbol *Flowchart* (Lanjutan)

|  |  |
| --- | --- |
| **Simbol** | **Deskripsi** |
| *Input/Output* | *Input/Output* Data Yang Diproses Atau Informasi |
| *Predefined Process* | Rincian Operasi Berada Di Tempat Lain |
| *Decision* (Keputusan) | Seleksi Atau Kondisi, Keputusan Dalam Program |
| *On Page Connector*  (Penghubung) | Penghubung Bagian-Bagian *Flowchart* Pada Halaman  Yang Sama |

### 2.6 Unifield Modeling Language (UML)

#### 2.6.1 Pengertian UML

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

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

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

Seperti bahasa-bahasa lainnya, UML mendefinisikan notasi dan syntax/semantik. Notasi UML merupakan sekumpulan bentuk khusus untuk menggambarkan berbagai diagram piranti lunak. Setiap bentuk memiliki makna tertentu, dan UML *syntax* mendefinisikan bagaimana bentuk-bentuk tersebut dapat dikombinasikan. Notasi UML terutama diturunkan dari 3 notasi yang telah ada sebelumnya: Grady Booch OOD (*Object-Oriented Design*), Jim Rumbaugh OMT (*Object Modeling Technique*), dan Ivar Jacobson OOSE (*Object-Oriented Software Engineering*) (Dharwiyanti & Wahono, 2003).

#### 2.6.2 Diagram – Diagram UML

Seperti yang telah dikemukakan sebelumnya, UML merupakan tool yang tepat untuk memodelkan sebuah rancangan perangkat lunak. Seperti hanya tool untuk memodelkan perangkat lunak terstruktur, UML pun memiliki diagramdiagram tertentu dan terstandar. Adapun jenis diagram yang digunakan dalam pemodelan UML yaitu:

### 1. *Use* *Case* *Diagram*

*Use case diagram* menampilkan sekumpulan *use case* dan *actor* (pelaku) dan hubungan diantara *use case* dan *actor* tersebut. *Use case diagram* digunakan untuk penggambaran *use case* statik dari suatu *system*. *Use case* menjelaskan apa yang dilakukan sistem (atau subsistem) tetapi tidak menspesifikasikan cara kerjanya. *Flow of event* digunakan untuk menspesifikasikan kelakuan dari *use case*. *Flow of event* menjelaskan *use case* dalam bentuk tulisan dengan sejelas-jelasnya, diantaranya bagaimana, kapan *use case* dimulai dan berakhir, ketika *use case* berinteraksi dengan aktor, obyek apa yang digunakan, alur dasar dan alur *alternative*.

## Tabel 2.5 Simbol *Use Case Diagram* (Kiswanto, 2016)

|  |  |  |
| --- | --- | --- |
| **Simbol** | **Nama** | **Deskripsi** |
|  | *Use Case* | Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor, biasanya dinyatakan dengan menggunakan kata kerja di awal frase nama *use case.* |
|  | Aktor | Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat. |
|  | Asosiasi | Komunikasi antara aktor dan *use case* yang berpartisipasi pada *use* *case* atau *use* *case* memiliki interaksi dengan aktor. |
|  | Ekstensi (*extend*) | Relasi *use* *case* tambahan ke sebuah *use* *case* dimana *use* *case* yang ditambahkan dapat berdiri sendiri walau tanpa *use* *case* tambahan itu. |
|  | Generalisasi | Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah *use case* dimana fungsi yang satu adalah fungsi yang lebih umum dari yang lainnya. |
|  | Menggunakan  (*include*) | Relasi *use case* tambahan ke sebuah *use* *case* dimana *use* *case* yang ditambahkan memerlukan *use* *case* ini untuk menjalankan fungsinya atau sebagai syarat dijalankan *use* *case* ini. |

### 2. *Activity* *Diagram*

Diagram aktivitas atau *activity* diagram menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses pada perangkat lunak. Yang perlu di perhatikan adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan yang di lakukan aktor.

## Tabel 2.6 Simbol *Activity Diagram* (Kiswanto, 2016)

|  |  |  |
| --- | --- | --- |
| **Simbol** | **Nama** | **Deskripsi** |
|  | Status Awal | Status awal aktivitas sistem. |
|  | Aktivitas | Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja. |
|  | Percabangan  (*decision*) | Asosiasi percabangan digunakan untuk mendefinisikan jika ada pilihan aktivitas lebih dari satu. |
|  | Penggabungan  (*join*) | Asosiasi penggabungan digunakan untuk mendefinisikan beberapa aktivitas yang digabungkan menjadi satu. |
|  | Status Akhir | Status akhir yang dilakukan sistem. |
|  | *Swimlane* | Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi. |

### 3. Class Diagram

*Class Diagram* adalah diagram yang digunakan untuk menampilkan beberapa kelas yang ada dalam sistem perangkat lunak yang akan dikembangkan. *Class Diagram* menunjukkan antar kelas dalam sistem yang sedang dibangun dan bagaimana mereka saling berkolaborasi untuk mencapai suatu tujuan.

Susunan struktur kelas yang baik pada diagram kelas sebaiknya memiliki jenis-jenis kelas berikut:

1. *Main* *Class*

*Class* yang memiliki fungsi awal dieksekusi ketika sistem dijalankan.

1. *Class* yang menangani tampilan sistem (*view*)

*Class* yang mendefinisikan dan mengatur tampilan ke pemakai.

1. *Class* yang diambil dari pendefinisian *use* *case* (*controller*)

*Class* yang menangani fungsi – fungsi yang harus ada diambil dari pendefinisian use case, kelas ini biasanya disebut dengan kelas proses yang menangani proses bisnis pada perangkat lunak.

1. *Class* yang diambil dari pendefinisian data (model)

*Class* yang digunakan untuk memegang atau membungkus data menjadi sebuah kesatuan yang diambil maupun akan disimpan ke basis data. Adapun simbol – simbol yang digunakan dalam *Class* diagram adalah sebagai berikut: **Tabel 2.7 Simbol *Class Diagram* (Kiswanto, 2016)**

|  |  |  |
| --- | --- | --- |
| **Simbol** | **Nama** | **Deskripsi** |
|  | *Class* | Kelas pada stuktur sistem. |
|  | Antarmuka  (*interface*) | Sama dengan konsep antarmuka dalam pemrograman berorientasi objek. |
|  | Asosiasi | Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan *multipicity*. |
|  | Asosiasi Berarah  (*directed* *association*) | Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan *multiplicity*. |

## Tabel 2.7 Simbol *Class Diagram*

|  |  |  |
| --- | --- | --- |
|  | Generalisasi | Relasi antar kelas dengan makna generalisasi – spesialisasi (umum khusus). |
|  | Kebergantungan  (*dependency*) | Relasi antar kelas dengan makna kebergantungan antar kelas. |
|  | Agregasi  (*aggreggation*) | Relasi antar kelas dengan makna semua bagian (*whole* *part*). |

### 4. *Sequence* *Diagram*

*Sequence* *diagram* adalah suatu diagram interaksi yang menekankan pada pengaturan waktu dari pesan-pesan. Diagram ini menampilkan sekumpulan peran dan pesan-pesan yang dikirim dan diterima oleh instansi yang memegang peranan tersebut. *Sequence* diagram menangkap objek dan *class* yang terlibat dalam skenario dan urut-urutan pesan yang ditukar antara objek diperlukan untuk melaksanakan fungsionalitas skenario. *Sequence* *diagram* berasosiasi dengan *use case* selama proses pengembangan. Dalam *Unified Model Language* (UML), objek dalam *sequence diagram* digambar dengan segi empat yang berisi nama objek yang diberi garis bawah. Objek dapat diberi nama dengan tiga cara: (nama objek), (nama objek dan *class*) atau (hanya nama *class* (*anonymous* *object*). Berikut notasi *sequence* *diagram* seperti terlihat pada gambar dibawah ini.

## Tabel 2.8 Simbol *Sequence Diagram* (Kiswanto, 2016)

|  |  |  |
| --- | --- | --- |
| **Simbol** | **Nama** | **Deskripsi** |
|  | Aktor | Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat. |

**Tabel 2.8 Simbol *Sequence Diagram* (Lanjutan)**

|  |  |  |
| --- | --- | --- |
|  | Garis Hidup  (*lifeline*) | Menyatakan kehidupan suatu objek. |
|  | Objek | Menyatakan objek yang berinteraksi. |
|  | Waktu Aktif | Menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan di dalamnya. |
|  | Pesan Tipe  *Call* | Menyatakan suatu objek memanggil operasi atau metode yang ada pada objek lain atau dirinya sendiri. |
|  | Pesan Tipe  *Send* | Menyatakan bahwa suatu objek mengirimkan data/ masukan/ informasi ke objek lainnya, arah panah mengarah pada objek yang dikirimi. |

## BAB III

## METODOLOGI PENELITIAN

### 3.1 Metode Pengumpulan Data

Metode pengumpulan data yang digunakan yaitu kajian pustaka yang digunakan untuk mencari literatur atau sumber pustaka yang berkaitan dengan perangkat lunak yang dibuat dan membantu memperjelas teori-teori yang ada. Sumber literatur berupa buku, *e-book*, *paper*, jurnal, karya ilmiah, dan beberapa situs penunjang.

Sedangkan data yang akan diolah oleh sistem adalah data yang diambil dari Badan Meteorologi, Klimatologi dan Geofisika (BMKG) Stasiun Meteorologi Maritim Kendari.

Adapun sampel data meteorologi yang digunakan dapat dilihat pada Tabel 3.1.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Tanggal | Temperatur Minimum (°C) | Temperatur Maximum (°C) | Temperatur Rata  -    rata (°C) | Kelembapan Rata  -    rata (%) | Curah Hujan (mm) | Lamanya Penyinaran Matahari (Jam) | Kecepatan Angin Maksimum (m/s) | Arah Angin Saat Kecepatan Maksimum  (  )  ° | Kecepatan Angin Rata  -    rata (m/s  ) | Arah Angin Terbanyak (°  ) |
| 01-01-2019 | 25.4 | 31.8 | 28.1 | 84 | 2.5 | 1.3 | 3 | 60 | 1 | C |
| 02-01-2019 | 25.7 | 29.4 | 27 | 90 |  | 3.2 | 3 | 60 | 1 | C |
| 03-01-2019 | 25.4 | 32 | 27.8 | 84 | 14 | 0.3 | 4 | 110 | 1 | C |
| 04-01-2019 | 24.8 | 32.4 | 28.1 | 81 | 4.8 | 5.9 | 3 | 110 | 1 | C |
| 05-01-2019 | 24.2 | 31.8 | 27.5 | 84 | 12 | 6.3 | 2 | 260 | 1 | SW |
| 06-01-2019 | 24 | 32.8 | 27.8 | 82 | 1.2 | 2.6 | 3 | 60 | 1 | C |

### Tabel 3.1 Sampel Data Meteorologi

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Tanggal | Temperatur Minimum (°C) | Temperatur Maximum (°C) | Temperatur Rata  -    rata (°C) | Kelembapan Rata  -    rata (%) | Curah Hujan (mm) | Lamanya  Penyinaran Matahari (Jam) | Kecepatan Angin Maksimum (m/s) | Arah Angin Saat Kecepatan Maksimum  (  )  ° | Kecepatan Angin Rata  -    rata (m/s) | Arah Angin Terbanyak (°  ) |
| 07-01-2019 | 24.8 | 30.8 | 27.5 | 84 | 1.5 | 6.3 | 3 | 50 | 1 | NE |
| 08-01-2019 | 24.8 | 32 | 27.6 | 85 |  | 2.7 | 3 | 60 | 1 | NE |
| 09-01-2019 | 24.8 | 27.8 | 25.7 | 92 | 29.2 | 5.8 | 2 | 40 | 0 | C |
| 10-01-2019 | 23.5 | 32.6 | 26.8 | 87 | 10.2 | 0.1 | 4 | 120 | 1 | C |
| 11-01-2019 | 24 | 32 | 26.8 | 86 | 28.3 | 5.2 | 3 | 60 | 1 | W |
| 12-01-2019 | 23.8 | 32 | 27.6 | 84 | 2 | 7.4 | 3 | 100 | 1 | W |
| 13-01-2019 | 24 | 33.5 | 27.8 | 84 |  | 5.6 | 4 | 120 | 1 | C |
| 14-01-2019 | 25.6 | 31 | 27.5 | 86 | 6.9 | 6.6 | 3 | 90 | 1 | C |
| 15-01-2019 | 23.8 | 30.6 | 27.1 | 86 | 24.1 | 3.9 | 4 | 20 | 1 | NE |
| 16-01-2019 | 24 | 30 | 26.9 | 91 | 36 | 2.8 | 2 | 120 | 1 | C |
| 17-01-2019 | 24.6 | 30.2 | 26.5 | 88 | 24.6 | 2.7 | 2 | 110 | 0 | C |
| 18-01-2019 | 24.2 | 31.8 | 27.4 | 84 | 22.5 | 1.2 | 3 | 60 | 1 | NE |
| 19-01-2019 | 24.8 | 29.6 | 27.2 | 90 |  | 5.8 | 2 | 110 | 0 | C |
| 20-01-2019 | 25.2 | 31.4 | 27.5 | 89 |  | 0 | 2 | 270 | 1 | C |
| 21-01-2019 | 24.8 | 32 | 27.4 | 90 | 3.8 | 0 | 3 | 50 | 1 | C |
| 22-01-2019 | 25.2 | 32.6 | 27.9 | 87 | 2.2 | 4.1 | 3 | 120 | 1 | NE |
| 23-01-2019 | 24.4 | 34.2 | 28.5 | 79 | 15.3 | 4.1 | 3 | 240 | 1 | SW |

**Tabel 3.1 Sampel Data Meteorologi (Lanjutan)**

Keterangan:

C = Central

E = East

W = West

SW = Southwest

N = North

NE = Northeast

Catatan: Jika data kosong berarti tidak dilakukan pengukuran atau data tidak terukur

#### 3.2 Metode Pengembangan Sistem

Metode pengembangan sistem yang saya gunakan yaitu *Rational Unified Process*, yang mendefinisikan tahap pelaksanaan, kegiatan peran pelaksana, hasil kerja dan prinsip yang harus diikuti.

### Tabel 3.2. Fase Pengembangan *Software*

|  |  |
| --- | --- |
| **Fase *Rational Unified***  ***Process*** | **Proses yang dilakukan** |
| *Inception* | Analisis pada aplikasi yang akan dibuat. Seperti manfaat dan tujuan pembuatan aplikasi serta batasan masalah yang akan diterapkan. |
| *Elaboration* | Mendesain *UML* sistem, seperti *usecase diagram,* *activity diagram, class diagram* dan *sequence diagram* dari perangkat lunak yang akan dibuat, serta desain antarmuka aplikasi yang akan dibuat. |
| *Construction* | Membuat *interface* aplikasi dan *coding* perangkat lunak. |
| *Transition* | Pada tahap ini meliputi testing dan pengujian sistem. Memperbaiki masalah-masalah yang muncul saat pembuatan dan setelah pengujian aplikasi. |

#### 3.3 Waktu dan Tempat Penelitian

##### 3.3.1 Waktu Penelitian

Waktu pelaksanaan penelitian tugas akhir dilaksanakan mulai dari bulan Februari 2020. Rincian kegiatan dapat dilihat pada Tabel 3.3.

### Tabel 3.3 Tabel Waktu Penelitian

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| No  . | Uraian |  |  | |  |  | | | | Bulan | | | |  | | | |  | | | |
|  | Juli | |  | Agustus | | | | September | | | | Oktober | | | | November | | | |
| 1 | 2 | 3 | 4 | 1 | 2 | 3 | 4 | 1 | 2 | 3 | 4 | 1 | 2 | 3 | 4 | 1 | 2 | 3 | 4 |
| 1. | *Inception* |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 2. | *Elaboration* |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 3. | *Construction* |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 4. | *Transition* |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

#### 3.3.2 Tempat Penelitian

Penelitian tugas akhir bertempat di Jl. Jendral Sudirman No.158, Kp. Salo, Badan Meteorologi, Klimatologi dan Geofisika (BMKG) Stasiun Meteorologi Maritim Kendari.

### 3.4 Analisis Sistem

Analisis sistem adalah suatu tahapan yang bertujuan untuk mengetahui dan mengamati siapa saja yang terlibat dalam sistem. Pembahasan yang ada pada analisis sistem ini yaitu analisis kebutuhan fungsional meliputi perancangan sistem menggunakan bahasa pemodelan UML (*Unified Modeling Language),* perancangan tampilan *interface* serta analisis kebutuhan nonfungsional yang meliputi kebutuhan perangkat keras dan perangkat lunak yang akan digunakan.

#### 3.4.1 Analisis Kebutuhan Fungsional

Analisis kebutuhan fungsional adalah segala bentuk data yang dibutuhkan oleh sistem agar sistem dapat berjalan sesuai dengan prosedur yang dibangun melalui perancangan sistem. Adapun kebutuhan fungsional dari sistem yang akan dibangun yaitu Perancangan diagram sistem menggunakan bahasa pemodelan UML (*Unified Modeling Language)* yang meliputi *flowchart* sistem, *flowchart* metode, *use case diagram, activity diagram, sequence diagram, class diagram* dan *design interface* sistem.

#### 3.4.2 Analisis Kebutuhan Nonfungsional

Analisis kebutuhan nonfungsional adalah sebuah langkah dimana pembangun aplikasi menganalisis sumber daya kebutuhan untuk membangun aplikasi yang akan dibangun. Analisis kebutuhan nonfungsional yang dilakukan dibagi dalam dua tahap, yaitu anlisis kebutuhan perangkat keras dan analisis kebutuhan perangkat lunak. Kebutuhan perangkat keras yaitu kebutuhan perangkat atau komponen yang dibutuhkan pada sistem dan perangkat lunak yaitu kebutuhan perangkat lunak untuk membantu agar komponen perangkat keras dapat berfungsi dan dapat dijalankan pada sistem.

### 1. Kebutuhan Perangkat Keras

Untuk menerapkan rancangan yang telah dijelaskan sebelumnya, dibutuhkan beberapa perangkat keras sebagai sarana untuk mengimplementasikan aplikasi yang dibangun. Berikut spesifikasi minimum laptop atau komputer yang dibutuhkan.

### Tabel 4.1 Spesifikasi Minimum Perangkat Keras

### 

|  |  |  |
| --- | --- | --- |
| No | Nama Perangkat | Spesifikasi |
| 1. | *Processor* | *Intel Celeron* |
| 2. | *Memori* | *RAM 2 GB DDR3 L Memory* |

#### 2. Kebutuhan Perangkat Lunak

Adapun perangkat lunak atau *software* yang dibutuhkan adalah sebagai berikut:

### Tabel 4.2 Spesifikasi Perangkat Lunak

|  |  |  |
| --- | --- | --- |
| No | Nama Perangkat Lunak | Spesifikasi |
| 1. | *Operating System* | *Windows 10* |
| 2. | *Text Editor* | *Sublime, Visual Studio Code* |
| 3. | *Xampp* | *Versi 3.2.1* |
| 4. | *Browser* | *Google Chrome / Mozila Firefox /*  *Opera GX Browser* |

#### 3.5 Analisis Perancangan Sistem

Perancangan sistem yang akan dibangun terdiri atas perancangan *flowchart* dan perancanagan *Unified Modeling Language* (UML) serta perancangan *user interface.*

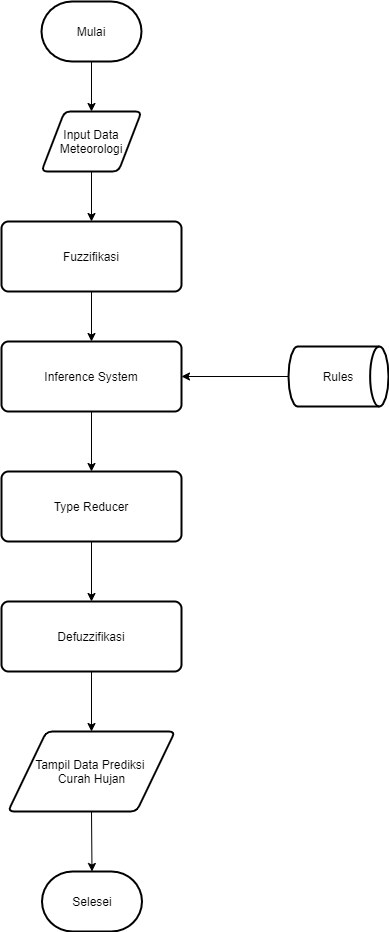
##### 3.5.1 *Flowchart*

Pada sistem ini terdapat beberapa *flowchart* diantaranya adalah sebagai berikut.

#### 1. *Flowchart* Sistem

Adapun alur kerja *flowchart* sistem pada Aplikasi Perkiraan Curah Hujan di Kota Kendari Menggunakan *Interval Fuzzy Type 2* adalah sebagai berikut. a. *User* menginput data klimatologi

1. Data tersebut masuk dalam proses fuzzifikasi
2. Lalu masuk dalam proses *inference system* dan menyesuaikan dengan rules yang ada
3. Kemudian memasuki proses *type reducer*
4. Sebelum di tampilkan data di defuzzifikasi
5. Tampil data prediksi curah hujan

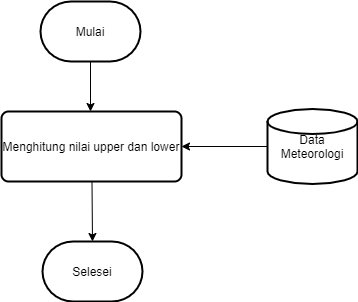


### Gambar 4.1 *Flowchart* Sistem

#### 2. *Flowchart* Fuzzifikasi

Adapun *flowchart* fuzzifikasi adalam sistem ini adalah sebagai berikut.

1. Mengambil data meteorologi dari *database*
2. Menentukan nilai keanggotaan upper dan lower dari tiap-tiap variabel yang digunakan.

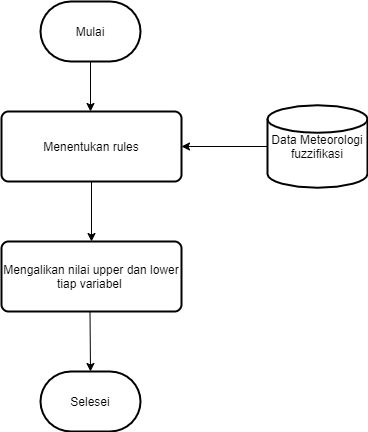


### Gambar 4.2 Flowchart Fuzzifikasi

#### 3. *Flowchart* *Inference*

Adapun *flowchart inference* dalam sistem ini adalah sebagai berikut.

1. Data meteorologi yang telah difuzzifikasi akan ditentukan rulesnya
2. Kemudian nilai keanggotaan upper akan dikalikan denga nilai keanggotaan upper dari variabel yang lain, begitu juga dengan nilai keanggotaan lower



#### Gambar 4.3 *Flowchart* *Inference*

##### 4. *Flowchart* *Type* *Reducer*

Adapun *flowchart type* *reducer* pada sistem ini adalah sebagai berikut.

1. Sorting nilai 𝑦̅𝑛dan 𝑦𝑛dengan nilai meningkat.

𝑓̅𝑛+𝑓𝑛

1. Hitung nilai 𝑓𝑛 = dimana 𝑛=1,2,…,𝑁

2

∑𝑁𝑛=1𝑦𝑛𝑓𝑛

1. Hitung nilai y, apabila mencari yl gunakan rumus 𝑦= ∑𝑁𝑛 𝑓𝑛 namun, jika

=1

mencari yr gunakan rumus 𝑦= ∑∑𝑁𝑛=𝑁𝑛1=𝑦1̅𝑓𝑛𝑛𝑓𝑛

1. Cari nilai k dengan aturan *k* (1≤𝑘≤𝑁−1)

∑𝑘

1. Hitung nilai 𝑦′, apabila mencari yl gunakan rumus 𝑛=∑1𝑘𝑓̅𝑛𝑦𝑓̅𝑛𝑛++∑∑𝑛𝑁𝑛𝑁==𝑘𝑘++11𝑓𝑓𝑛𝑛𝑦𝑛

𝑛=1

∑𝑘𝑛=1𝑓𝑛𝑦̅𝑛+∑𝑁𝑛=𝑘+1𝑓̅𝑛𝑦̅𝑛 namun, jika mencari yr gunakan rumus 𝑁 𝑓̅𝑛

∑

𝑓

𝑛

𝑘

𝑛

=

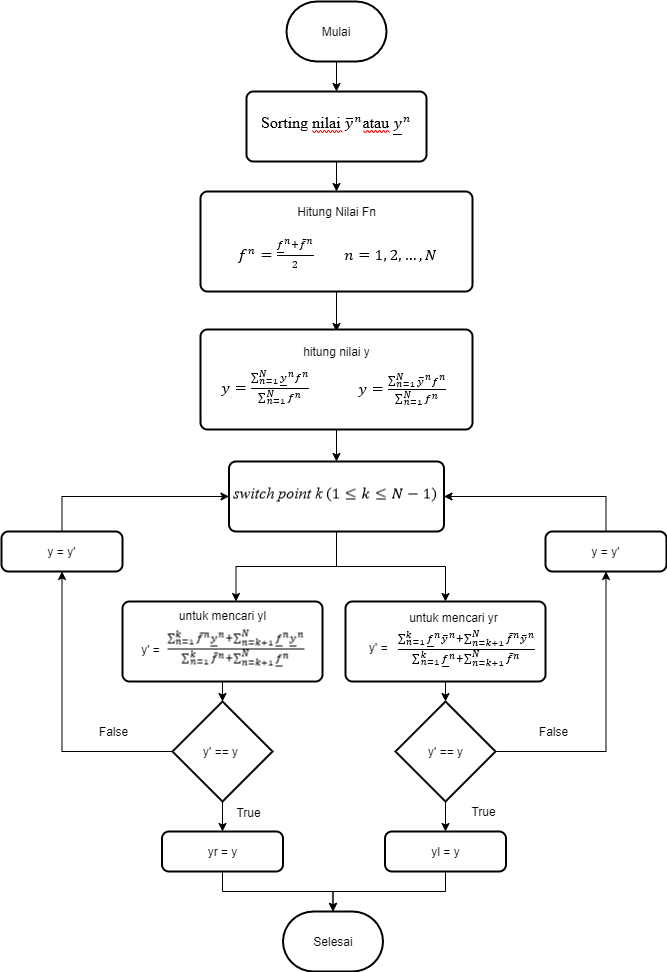
1

+

∑

𝑛=𝑘+1

1. Jika 𝑦′ ==𝑦, set 𝑦𝑙=𝑦
2. Jika tidak, set 𝑦=𝑦′, kemudian kembali mencari nilai k.

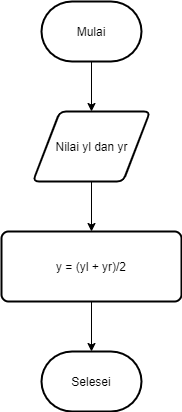


#### Gambar 4.4 *Flowchart* *Type* *Reducer*

##### 5. *Flowchart* Defuzzifikasi

Adapun *flowchart* defuzzifikasi pada sistem ini adalah sebagai berikut.

1. Mengambil nilai yl dan yr
2. Kemudian menghitung nilai y dengan rumus 𝑦(𝑥)= 𝑦𝑙+𝑦𝑟 2



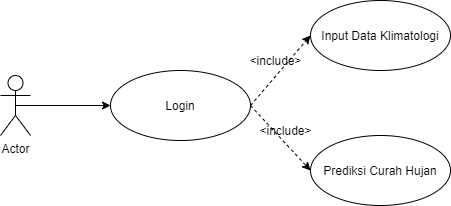
### Gambar 4.5 *Flowchart* Defuzzifikasi

#### 3.5.2 Unified Modeling Language (UML)

Aplikasi dibangun dengan menggunakan *Unified Modeling Language* (UML). UML merupakan bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram yang terdiri dari *Use Case Diagram, Activty Diagram, Sequence Diagram* dan *Class Diagram.*

#### 1. *Use Case Diagram*

*Case diagram* menjelaskan apa yang akan dilakukan oleh sistem yang akan dibangun dan siapa yang berinteraksi dengan sistem. Berikut adalah *use case diagram* untuk sistem yang akan dibangun.



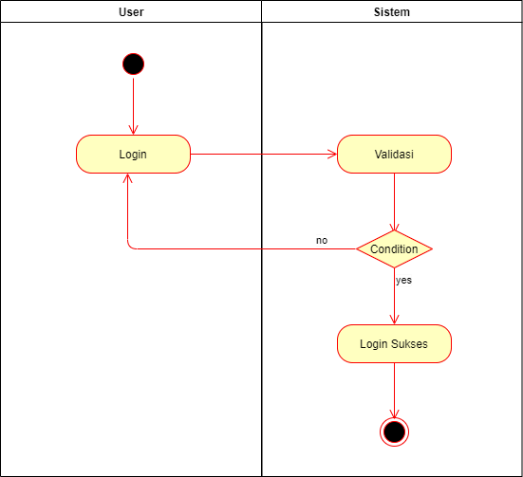
#### Gambar 4.6 *Use Case Diagram*

##### 2. *Activity Diagram*

*Activity diagram* menjelaskan tentang aktivitas apa saja yang terjadi dalam sebuah sistem.

#### a. *Activity Diagram Login*

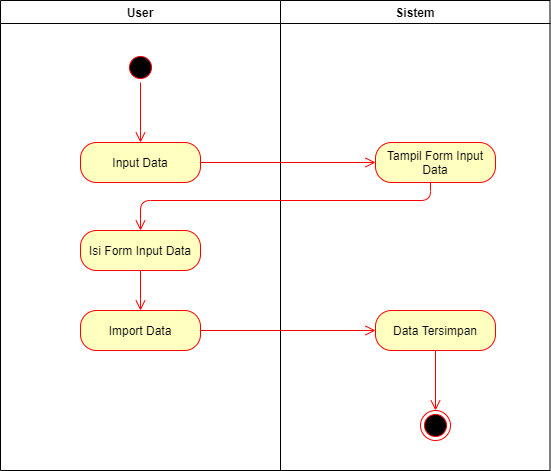
Pada Gambar 4.7 menunjukkan *Activity Diagram Login*. Pada diagram ini dimulai dengan *user* yang mengisi *username* dan *password* kemudian sistem melakukan pengecekan *username* dan *password*, apabila sesuai maka user dapat menjalankan sistem, namun apabila data yang dimasukkan tidak *valid* maka sistem akan menampilkan halaman *login.*



**Gambar 4.7 *Activity Diagram Login***

#### b. *Activity Diagram Input* Data

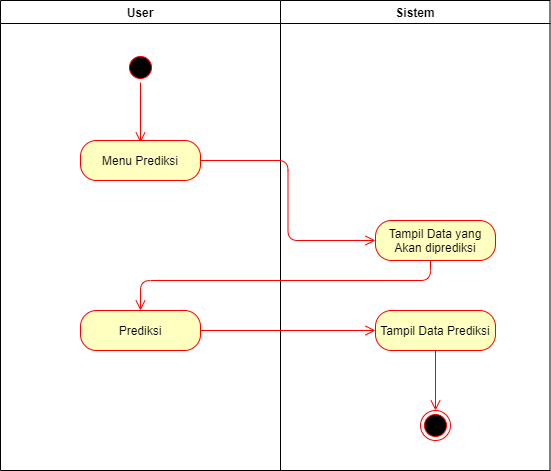
Pada Gambar 4.8 menunjukkan *Activity Diagram Input* Data. Pada *diagram* ini dimulai dari *user* yang memilih menu *input* data, lalu sistem akan menampilkan *form* *input* data. Kemudian user menginputkan data, setelah itu data akan tersimpan.



#### Gambar 4.8 *Activity Diagram Input* Data

##### c. *Activity* *Diagram* *Prediksi*

Pada Gambar 4.9 menunjukkan *Activity* *Diagram* *Prediksi*. Pada *diagram* ini dimulai dari *user* yang memilih menu prediksi, lalu sistem akan menampilkan data yang akan di prediksi. Kemudian *user* menekan *button* prediksi, maka akan muncul data yang telah di prediksi.



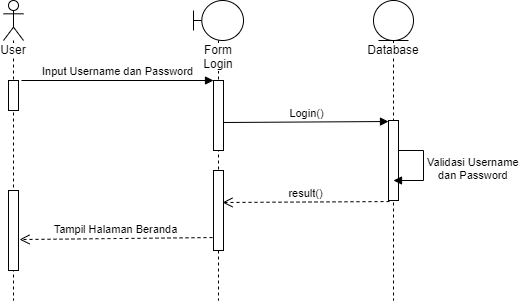
#### Gambar 4.9 *Activity* *Diagram* *Prediksi*

##### 3. *Sequence Diagram*

*Sequence diagram* menjelaskan tentang kolaborasi dinamis antara sejumlah *object* untuk menunjukkan rangkaian pesan yang dikirim antara *object* serta interaksi antara *object.*

#### a. *Sequence* *Diagram* *Login*

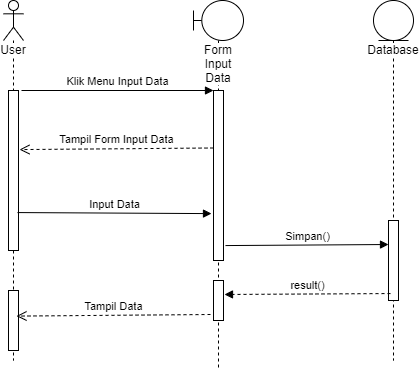
Pada Gambar 4.10 menunjukkan *Sequence* *Diagram* *Login*. Pada *diagram* ini digambarkan *user* memasukkan *username* dan *password* pada *form* *login*. Kemudian sistem melakukan fungsi *login* yang dimana data yang diinputkan melewati proses validasi, setelah divalidasi sistem akan mengirimkan hasilnya, jika berhasil maka akan menampilkan halaman beranda.



**Gambar 4.10 *Sequence Diagram Login***

#### b. *Sequence* *Diagram* *Input* Data

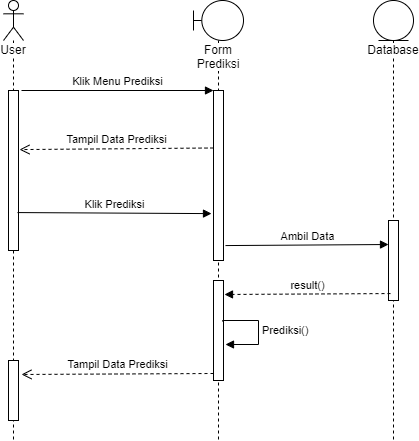
Pada Gambar 4.11 menunjukkan *Sequence* *Diagram* *Input* Data. Pada *diagram* ini digambarkan *user* memilih menu *input* data, maka akan tampil *form input* data, setelah menginput data maka sistem akan melakukan fungsi simpan, kemudian sistem akan mengirim hasilnya, jika berhasil maka akan tampil data yang telah diinput.



#### Gambar 4.11 *Sequence* *Diagram* *Input* Data

##### c. *Sequence* *Diagram* Prediksi

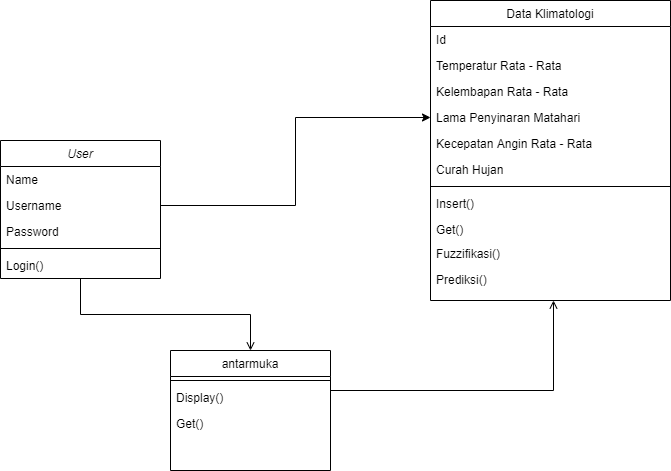
Pada Gambar 4.12 menunjukkan *Sequence* *Diagram* Prediksi. Pada *diagram* ini digambarkan *user* memilih menu prediksi, kemudian sistem akan menampilkan data yang akan diprediksi, setelah *user* menekan *button* prediksi, maka sistem akan mengambil data prediksi dari database lalu akan menjalankan fungsi prediksi, setelah itu akan tampil data yang telah diprediksi.



### Gambar 4.12 *Sequence* *Diagram* Prediksi

#### 4. *Class* *Diagram*

Pada Gambar 4.13 menunjukkan *Class* *Diagram.* Dimana pada diagram ini terdapat 3 tabelyaitu, tabel *user* yang berisi data user pengguna sistem ini, lalu ada tabel data klimatologi yang berisi data yang akan diprediksi, dan yang terakhir tabel antarmuka.



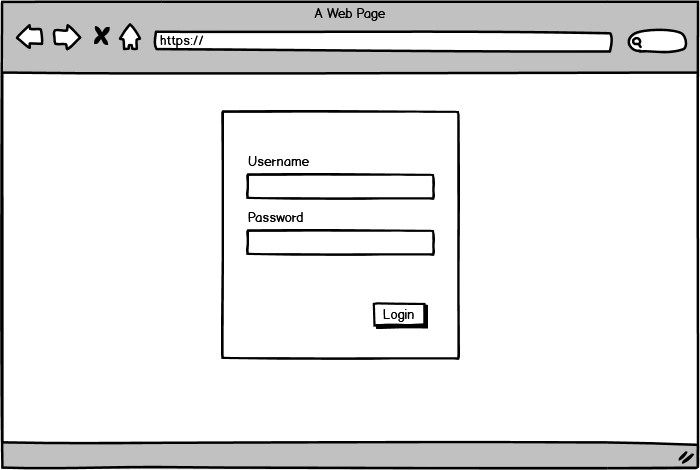
#### Gambar 4.13 *Class Diagram*

##### 3.5.3 *User Interface*

Rancangan antarmuka pengguna atau *design user interface* merupakan penggambaran tampilan yang digunakan secara langsung oleh pengguna terhadap sistem. Rancangan *user interface* ini dibuat sederhana agar mudah dimengerti pengguna dan tidak ada kerumitan dalam menjalankannya sehingga mencapai tujuan perangkat lunak yang *user friendly*.

##### 1. Halaman *Login*

Halaman ini berisi *textfield username*, *textfield password* dan *button login*.



### Gambar 4.14 Halaman *Login*

#### 2. Halaman Beranda

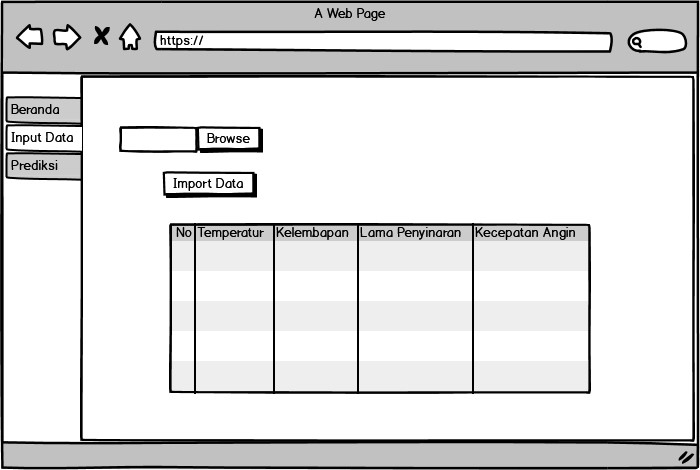
Halaman ini merupakan halaman paling awal terbuka setelah *user* melakukan *login*. Pada halaman ini hanya terdapat teks label.



### Gambar 4.15 Halaman Beranda

#### 3. Halaman *Input*

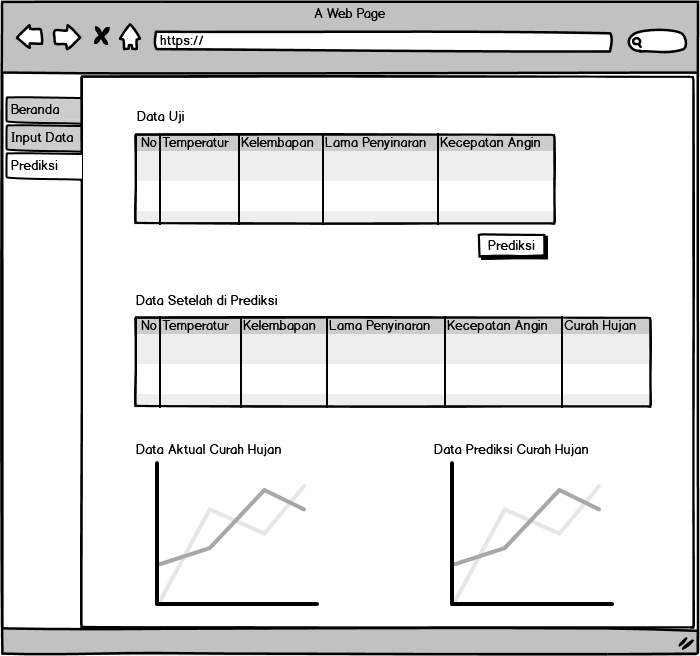
Halaman ini berisi *textfield, button browse, button import* data dan tabel yang berisi data yang telah diinputkan oleh *user.*



### Gambar 4.16 Halaman *Input*

#### 4. Halaman Prediksi

Halaman ini berisi 2 tabel, yang dimana pada tabel pertama berisi data yang akan diprediksi dan tabel kedua berisi data setelah diprediksi. Pada halaman ini juga memiliki *button* prediksi, grafik data curah hujan yang aktual dan grafik data prediksi curah hujan.



**Gambar 4.17 Halaman Prediksi**

DAFTAR PUSTAKA

Dharwiyanti, S., & Wahono, R. S. (2003). Pengantar Unified Modeling LAnguage (UML). *IlmuKomputer.Com*, 1–13. http://www.unej.ac.id/pdf/yanti-uml.pdf

Fatma, D. (2018). *Penyinaran Matahari: Pengertian, Tipe dan Manfaatnya*. Https://Ilmugeografi.Com. https://ilmugeografi.com/astronomi/penyinaranmatahari

Kiswanto, C. F. (2016). SISTEM INFORMASI AKADEMIK SUB-SISTEM : UTILITY DAN EPSBED (Studi Kasus : Fakultas Teknik Universitas Halu Oleo). *Journal of Chemical Information and Modeling*, *53*(9), 1689–1699.

Mandey, F. N., Kolibu, H. S., & Bobanto, M. D. (2017). Pemodelan Sistem Prediksi Intensitas Curah Hujan di Kota Manado Dengan Menggunakan Kontrol

Logika Fuzzy. *Jurnal MIPA*, *6*(2), 19.

https://doi.org/10.35799/jm.6.2.2017.17068

Muflih, G. Z., Sunardi, & Yudhana, A. (2019). *JARINGAN SARAF TIRUAN BACKPROPAGATION UNTUK PREDIKSI CURAH HUJAN DI WILAYAH KABUPATEN WONOSOBO Universitas Ahmad Dahlan , Yogyakarta PENDAHULUAN Hujan berperan penting bagi kehidupan , curah hujan yang turun pada suatu wilayah dapat diprediksi namun tidak*. *4*(1), 45–56.

Nurohmah, H. (2015). Peramalan Beban Jangka Pendek Hari Libur Nasional Dengan Interval Type-2 Fuzzy Inference System Pada Sistem Jawa-Bali. *Prosiding SENTIA 2015 – Politeknik Negeri Malang*, *7*(1), 2085–2347.

Rizky, D. N., Usadha, I. G. N. R., & Farida Agustini Widjajati. (2012). Penerapan Fuzzy Inference System Pada Prediksi Curah Hujan di Surabaya Utara. *Jurnal Sains Dan Seni ITS*, *I*(1), I.

Saiful, M. F. (2018). Implementasi Algoritma Fuzzy Type-2 Untuk Menentukan Perilaku Npc Dalam Game Virtual Reality Survival Shooter. *Universitas Islam Negeri Maulana Malik Ibrahim Malang*. http://etheses.uinmalang.ac.id/11023/1/11650090.pdf

Sajiah, A. M., & Setiawan, N. A. (2016). Interval type-2 fuzzy logic system for diagnosis coronary artery disease. *Communications in Science and Technology*, *1*(2), 51–55. https://doi.org/10.21924/cst.1.2.2016.26

Sari, W. E., & Wahyunggoro, O. (2016). *Interval Type-2 Fuzzy Logic System Pada Estimasi State of Charge ( Soc ) Baterai Lithium Polymer*.

Soerjadi, W., & Swarinoto, Y. (2010). *Iklim Kawasan Indonesia (Dari Aspek Dinamik - Sinoptik)*. https://doi.org/10.1016/j.ejop.2005.06.002

Sudiarta, I. W. (2013). *Pengantar Meteorogi*.

Wu, D. (2014). *A Brief Tutorial on Interval Type-2 Fuzzy Sets and Systems*. 1–10.

Xu, L., Zhu, K., & Yang, X. (2016). Production forecasting of coalbed methane wells based on type-2 fuzzy logic system. *Open Petroleum Engineering*

*Journal*, *9*, 268–278. https://doi.org/10.2174/1874834101609010268