

# **BAB I**

## **PENDAHULUAN**

### **1.1. Latar Belakang**

Pontianak, ibukota Provinsi Kalimantan Barat, dikenal sebagai kota khatulistiwa karena tepat dilalui oleh garis lintang nol derajat atau garis khatulistiwa (*equator*). Sebagai kota yang terletak pada garis lintang nol derajat, Pontianak termasuk dalam negara yang berada dalam wilayah iklim tropis dan memiliki dua musim yaitu musim kemarau dan musim penghujan.

Musim penghujan adalah musim dengan ciri meningkatnya curah hujan di suatu wilayah dibandingkan biasanya dalam jangka waktu tertentu secara tetap. Sedangkan curah hujan adalah banyaknya hujan yang tercurah (turun) di suatu daerah dalam jangka waktu tertentu. Musim kemarau dan musim penghujan berganti selama enam bulan sekali. Namun menurut WWF (*World Wildlife Fund for Nature*) Indonesia, beberapa tahun terakhir ini, perubahan iklim global terasa ditandai dengan tidak menentunya perputaran musim kemarau maupun musim penghujan. Untuk mengetahui perubahan pola curah hujan tersebut, maka dirancanglah prediksi besaran curah hujan untuk melihat dan menganalisa pola hujan yang akan terbentuk ke depannya.

Prediksi atau *forecasting* adalah meramalkan, memproyeksikan, atau mengadakan perkiraan / taksiran terhadap berbagai kemungkinan yang akan terjadi pada masa mendatang. Prediksi atau *forecasting* dapat dilakukan dengan melibatkan pengambilan data historis dan memproyeksikannya ke masa mendatang dengan suatu bentuk model matematis. Metode peramalan (*forecasting*) terdiri dari metode kualitatif dan kuantitatif. Metode kualitatif adalah metode yang menganalisis kondisi objektif dengan apa adanya atau peramalan yang didasarkan atas data kualitatif pada masa lalu. Peramalan kualitatif memanfaatkan faktor-faktor penting seperti intuisi, pendapat, pengalaman pribadi sehingga hasil peramalan metode ini sangat bergantung kepada orang yang menyusunnya. Metode kuantitatif adalah peramalan yang didasarkan atas data

kuantitatif atau model matematis yang beragam dengan data masa lalu. Salah satu metode yang termasuk metode kuantitatif adalah metode *time series* atau runtun waktu. *Time series* atau runtun waktu adalah himpunan observasi data terurut dalam waktu, sehingga metode *time series* menganalisis dan menentukan pola data pada masa lampau yang dikumpulkan berdasarkan urutan waktu dan memproyeksikannya untuk mendapatkan prediksi di masa mendatang.

Sistem informasi prediksi besaran curah hujan yang akan dibangun menggunakan *forecasting* atau peramalan dengan metode *Fuzzy Time Series*. Logika *fuzzy* digunakan karena dapat memetakan suatu input ke dalam suatu output dan memiliki toleransi terhadap data-data yang tersedia. Dasar logika *fuzzy* adalah teori himpunan *fuzzy*. Teori himpunan *fuzzy* merupakan kerangka matematis untuk merepresentasikan ketidakpastian, ketidakjelasan, ketidaktepatan, kekurangan informasi dan kebenaran parsial (Kusumadewi, 2002).

Berdasarkan uraian tersebut, penelitian ini berfokus pada penerapan metode *Fuzzy Time Series* dalam meramalkan besaran curah hujan. Adapun penelitian ini diangkat dengan judul “*Prediksi Besaran Curah Hujan menggunakan Metode Fuzzy Time Series*”.

## **1.2. Rumusan Masalah**

Berdasarkan latar belakang, rumusan masalah dalam penelitian ini adalah bagaimana mengimplementasikan metode *Fuzzy Time Series* untuk membangun aplikasi yang dapat mengolah dan menentukan pola data curah hujan serta memprediksi besaran curah hujan.

## **1.3. Tujuan Penelitian**

Adapun tujuan dari penelitian yang dilakukan adalah mengimplementasikan metode *Fuzzy Time Series* untuk membangun aplikasi yang dapat mengolah dan menentukan pola data curah hujan serta memprediksi besaran curah hujan.

## **1.4. Batasan Masalah**

Pembatasan masalah dari penelitian yang akan dilakukan adalah :

1. Penelitian ini difokuskan pada implementasi metode *Fuzzy Time Series* dalam melakukan prediksi besaran curah hujan.
2. Data yang digunakan adalah data historis curah hujan pada Stasiun Supadio periode 1 Januari 2003 hingga 31 Desember 2017 yang diperoleh dari Badan Meteorologi, Klimatologi, dan Geofisika Kalimantan Barat.
3. Data yang digunakan untuk membangkitkan *missing value* dari data BMKG adalah data curah hujan dari 3 buah pos hujan yakni Pontianak (PTK-11), Padang Tikar (PTK-16), dan Desa Ambawang Kuala (PTK-22) yang diperoleh dari Balai Wilayah Sungai Kalimantan I (BWSK I).

#### **1.5. Sistematika Penulisan Skripsi**

Sistematika dari penulisan tugas akhir ini disusun dalam 5 (lima) bab yang terdiri dari Bab I Pendahuluan, Bab II Tinjauan Pustaka, Bab III Metodologi Penelitian, Bab IV Implementasi dan Hasil Pengujian serta Bab V Penutup.

Bab I Pendahuluan adalah bab yang berisi latar belakang, perumusan masalah, tujuan penelitian, pembatasan masalah, dan sistematika penulisan.

Bab II Tinjauan Pustaka adalah bab yang berisi uraian sistematis tentang hasil-hasil penelitian yang didapat oleh peneliti terdahulu dan landasan teori yang ada hubungannya dengan penelitian yang akan dilakukan.

Bab III Metodologi Penelitian adalah bab yang berisi tentang data dan perangkat penelitian, metode penelitian, dan diagram alir penelitian.

Bab IV Hasil dan Analisis adalah bab yang berisi tahapan pengujian, hasil pengujian dan analisis pengujian yang mengarah kepada suatu kesimpulan.

Bab V Penutup adalah bab yang berisi kesimpulan dari penelitian telah dilakukan dan saran/rekomendasi untuk perbaikan, pengembangan atau kesempurnaan / kelengkapan penelitian yang telah dilakukan.

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **2.1. Landasan Teori**

##### **2.1.1. Aplikasi**

Aplikasi adalah penggunaan dalam suatu computer, instruksi (*instruction*) atau pernyataan (*statement*) yang disusun sedemikian rupa sehingga computer dapat memproses input menjadi output. Aplikasi merupakan program siap pakai yang dapat digunakan untuk menjalankan perintah-perintah dari pengguna aplikasi tersebut dengan tujuan mendapatkan hasil yang lebih akurat sesuai dengan tujuan pembuatan aplikasi tersebut, aplikasi mempunyai arti yaitu pemecahan masalah menggunakan salah satu teknik pemrosesan data aplikasi yang biasanya berpacu pada sebuah komputasi yang diinginkan atau diharapkan maupun pemrosesan data yang diharapkan (Jogiyanto, 2005).

##### **2.1.2. Teori Hujan**

###### **2.1.2.1. Pengertian Curah Hujan**

Curah hujan merupakan ketinggian air hujan yang terkumpul dalam tempat yang datar, tidak menguap, tidak meresap, dan tidak mengalir. Satuan curah hujan selalu dinyatakan dalam satuan milimeter atau inchi namun untuk di Indonesia satuan curah hujan yang digunakan adalah dalam satuan milimeter (mm). Curah hujan dalam 1 (satu) milimeter memiliki arti dalam luasan satu meter persegi pada tempat yang datar tertampung air setinggi satu milimeter atau tertampung air sebanyak satu liter.

Intensitas curah hujan adalah jumlah curah hujan dalam suatu satuan waktu tertentu, yang biasanya dinyatakan dalam mm/jam, mm/hari, mm/tahun, dan sebagainya ; yang berturut-turut sering disebut hujan jam-jaman, harian, tahunan, dan sebagainya. Biasanya data yang sering digunakan untuk analisis adalah nilai maksimum, minimum dan nilai rata-ratanya.

#### **2.1.2.2. Proses Terjadinya Hujan**

Siklus hidrologi merupakan proses yang berlangsung secara terus menerus dimana air bergerak dari bumi ke atmosfer dan kemudian kembali ke bumi lagi. Proses ini diawali dengan menguapnya air di permukaan tanah dan laut ke udara. Uap air tersebut bergerak dan naik ke atmosfer, yang kemudian mengalami kondensasi dan berubah menjadi titik-titik air yang berbentuk awan. Selanjutnya titik-titik air tersebut jatuh sebagai hujan ke permukaan lautan dan daratan. Hujan yang jatuh sebagian tertahan oleh tumbuh-tumbuhan (intersepsi) dan selebihnya sampai ke permukaan tanah. Sebagian air hujan yang sampai ke permukaan tanah akan meresap ke dalam tanah (infiltrasi) dan sebagian lainnya mengalir di atas permukaan tanah (aliran permukaan atau surface runoff mengisi cekungan tanah, danau, dan masuk ke sungai dan akhirnya mengalir ke laut. Air yang meresap ke dalam tanah sebagian mengalir secara vertikal di dalam tanah (perkolasi) mengisi air tanah (ground water) yang kemudian keluar sebagai mata air atau mengalir ke sungai. Akhirnya aliran air di sungai akan sampai ke laut (Triatmodjo, 2008).

Pada dasarnya hujan dapat terjadi di sembarang tempat, asalkan terdapat dua faktor, yaitu faktor massa udara yang lembab dan faktor sarana meteorologi yang dapat mengangkat massa udara tersebut untuk berkondensasi. Hujan terjadi akibat massa udara yang mengalami penurunan suhu di bawah titik embun yang dapat mengalami perubahan pembentukan molekul air. Apabila massa udara terangkat ke atas dan mengalami perubahan suhu sampai mencapai ketinggian yang memungkinkan terjadinya kondensasi, maka akan dapat membentuk awan. Hujan hanya dapat terjadi apabila molekul-molekul air hujan sudah mencapai ukuran lebih dari 1 mm. Agar hujan dapat terjadi diperlukan titik-titik kondensasi, amoniak, debu dan asam belerang. Titik-titik kondensasi ini mempunyai sifat yang dapat mengambil uap air dari udara.

#### **2.1.2.3. Alat Pengukur Curah Hujan**

Dari beberapa jenis presipitasi, hujan adalah yang paling bisa diukur. Pengukuran dapat dilakukan secara langsung dengan menampung air hujan yang jatuh, namun tidak dapat dilakukan di seluruh wilayah tangkapan air, akan tetapi

hanya dapat dilakukan pada titik-titik yang ditetapkan dengan menggunakan alat pengukur hujan (Triatmodjo, 2008).

Alat Pengukur Curah Hujan terbagi menjadi 3 jenis, yaitu pengukur curah hujan biasa (observarium), pengukur curah hujan otomatis, dan pengukuran curah hujan digital. Prinsip kerja alat pengukur curah hujan antara lain :

1. Pengukur curah hujan biasa (observarium) curah hujan yang jatuh diukur tiap hari dalam kurun waktu 24 jam.
2. Pengukur curah hujan otomatis melakukan pengukuran curah hujan selama 24 jam dengan merekam jejak hujan menggunakan pias yang terpasang dalam jam alat otomatis tersebut dan dilakukan penggantian pias setiap harinya pada pukul 00.00 GMT.
3. Pengukuran curah hujan digital dimana curah hujan langsung terkirim ke monitor komputer berupa data sinyal yang telah diubah ke dalam bentuk satuan curah hujan.

### 2.1.3. Teori Peramalan

Secara umum pengertian peramalan adalah tafsiran. Namun dengan menggunakan teknik-teknik tertentu maka peramalan bukan hanya sekedar tafsiran. Ada beberapa definisi tentang peramalan, diantaranya:

- a. Peramalan atau *forecasting* diartikan sebagai penggunaan teknik-teknik statistik dalam bentuk gambaran masa depan berdasarkan pengolahan angka-angka historis (Buffa , 2007).
- b. Peramalan merupakan bagian integral dari kegiatan pengambilan keputusan manajemen (Makridakis et al, 1999).
- c. Peramalan adalah sebuah teknik yang menggunakan data historis untuk memperkirakan proyek yang akan datang (Chapman and Stephen, 2006).

Metode peramalan merupakan cara memperkirakan apa yang akan terjadi pada masa depan secara sistematis dan pragmatis atas dasar data yang relevan pada masa yang lalu, sehingga dengan demikian metode peramalan diharapkan dapat memberikan objektivitas yang lebih besar. Selain itu metode peramalan dapat memberikan cara pengerjaan yang teratur dan terarah, dengan demikian dapat dimungkinkannya penggunaan teknik penganalisaan yang lebih maju.

Dengan penggunaan teknik-teknik tersebut maka diharapkan dapat memberikan tingkat kepercayaan dan keyakinan yang lebih besar, karena dapat diuji penyimpangan atau deviasi yang terjadi secara ilmiah.

#### **2.1.3.1. Jenis-jenis Peramalan**

Berdasarkan sifatnya, peramalan dibedakan atas dua macam yaitu:

a. Peramalan Kualitatif

Peramalan kualitatif adalah peramalan yang didasarkan atas pendapat suatu pihak, dan datanya tidak bisa direpresentasikan secara tegas menjadi suatu angka atau nilai. Hasil peramalan yang dibuat sangat bergantung pada orang yang menyusunnya. Hal ini penting karena hasil peramalan tersebut ditentukan berdasarkan pemikiran yang instuisi, pendapat dan pengetahuan serta pengalaman penyusunnya.

b. Peramalan Kuantitatif

Peramalan kuantitatif adalah peramalan yang didasarkan atas data kuantitatif masa lalu (data historis). Hasil peramalan yang didapat sangat bergantung pada metode yang dipergunakan dalam peramalan tersebut. Baik tidaknya metode yang dipergunakan ditentukan oleh perbedaan atau penyimpangan antara hasil ramalan dengan kenyataan yang terjadi. Semakin besar penyimpangan antara hasil ramalan dengan kenyataan yang akan terjadi maka semakin baik pula metode yang digunakan.

#### **2.1.3.2. Jangka Waktu Peramalan**

Jangka waktu peramalan dapat dikelompokkan menjadi tiga kategori, yaitu (Heizer and Render, 2005):

1. Peramalan jangka pendek, peramalan untuk jangka waktu kurang dari tiga bulan.
2. Peramalan jangka menengah, peramalan untuk jangka waktu antara tiga bulan sampai tiga tahun.
3. Peramalan jangka panjang, peramalan untuk jangka waktu lebih dari tiga tahun.

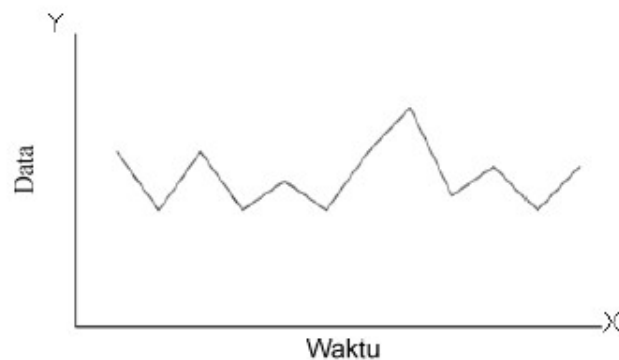
#### **2.1.3.3. Jenis-jenis Pola Data**

Langkah penting dalam memilih suatu metode deret berkala (*time series*) yang tepat adalah dengan mempertimbangkan jenis pola data, sehingga metode yang paling tepat dengan pola tersebut dapat diuji.

Pola data dapat dibedakan menjadi empat jenis, yaitu (Makridakis et al, 1999):

1. Pola Horizontal (H) atau Horizontal Data Pattern

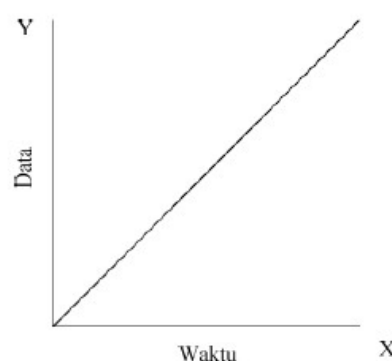
Pola data ini terjadi bilamana data berfluktuasi di sekitar nilai rata-rata yang konstan. Suatu produk yang penjualannya tidak meningkat atau menurun selama waktu tertentu termasuk jenis ini. Bentuk pola horizontal ditunjukkan seperti gambar 2.1.



Gambar 2.1 Pola Data Horizontal

2. Pola Trend (T) atau Trend Data Pattern

Pola data ini terjadi bilamana terdapat kenaikan atau penurunan sekuler jangka panjang dalam data. Contohnya penjualan perusahaan, produk bruto nasional (GNP) dan berbagai indikator bisnis atau ekonomi lainnya, selama perubahan sepanjang waktu. Bentuk pola trend ditunjukkan seperti gambar 2.2.

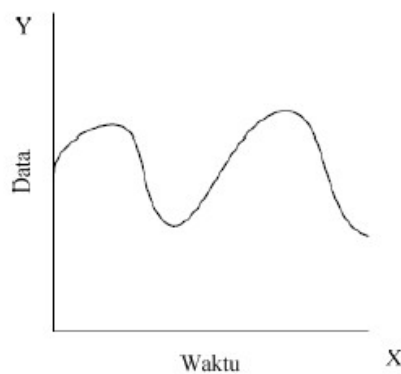




Gambar 2.2 Pola Data Trend

### 3. Pola Musiman (S) atau Seasonal Data Pattern

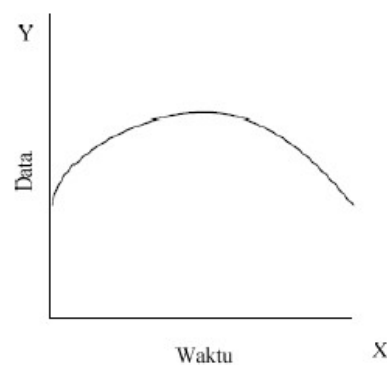
Pola data ini terjadi bilamana suatu deret dipengaruhi oleh faktor musiman (misalnya kuartal tahun tertentu, bulan atau hari-hari pada minggu tertentu). Penjualan dari produk seperti minuman ringan, es krim dan bahan bakar pemanas ruang semuanya menunjukkan jenis pola ini. Bentuk pola musiman ditunjukkan seperti gambar 2.3.



Gambar 2.3 Pola Data Musiman

### 4. Pola Siklis (S) atau Cycled Data Pattern

Pola data ini terjadi bilamana datanya dipengaruhi oleh fluktuasi ekonomi jangka panjang seperti yang berhubungan dengan siklus bisnis. Contohnya penjualan produk seperti mobil, baja. Bentuk pola siklis ditunjukkan seperti gambar 2.4.



Gambar 2.4 Pola Data Siklis

#### 2.1.3.4. Data Berkala (*Time Series*)

Data berkala adalah data yang dikumpulkan dari waktu ke waktu, untuk menggambarkan perkembangan suatu kegiatan. Analisis data berkala memungkinkan kita untuk mengetahui perkembangan suatu atau beberapa kejadian serta hubungan/pengaruhnya terhadap kejadian lainnya. Pola gerakan data atau nilai-nilai variabel dapat diikuti atau diketahui dengan adanya data berkala, sehingga data berkala dapat dijadikan sebagai dasar untuk:

1. Pembuatan keputusan pada saat ini.
2. Peramalan keadaan perdagangan dan ekonomi pada masa yang akan datang.
3. Perencanaan kegiatan dimasa yang akan datang.

Gerakan-gerakan khas dari data *time series* dapat digolongkan ke dalam empat kelompok utama, yang sering disebut komponen-komponen *time series*:

1. Gerakan jangka panjang atau sekuler merujuk kepada arah umum dari grafik *time series* yang meliputi jangka waktu yang panjang.
2. Gerakan siklis (*cyclical movements*) atau variasi siklis merujuk kepada gerakan naik-turun dalam jangka panjang dari suatu garis atau kurva trend. Siklis yang demikian dapat terjadi secara periodik ataupun tidak, yaitu dapat ataupun tidak dapat mengikuti pola yang tepat sama setelah interval-interval waktu yang sama. Dalam kegiatan bisnis dan ekonomi, gerakan-gerakan hanya dianggap siklis apabila timbul kembali setelah interval waktu lebih dari satu tahun.
3. Gerakan musiman (*seasonal movements*) atau variasi musim merujuk kepada pola-pola yang identik, atau hampir identik, yang cenderung diikuti suatu *time series* selama bulan-bulan yang bersangkutan dari tahun ke tahun. Gerakan-gerakan demikian disebabkan oleh peristiwa-peristiwa yang berulang-ulang terjadi setiap tahun.
4. Gerakan tidak teratur atau acak (*irregular or random movements*) merujuk kepada gerakan-gerakan sporadis dari *time series* yang disebabkan karena peristiwa-peristiwa kebetulan seperti banjir, pemogokan, pemilihan umum, dan sebagainya. Meskipun umumnya dianggap bahwa peristiwa-peristiwa demikian menyebabkan variasi-variasi yang hanya berlangsung untuk jangka pendek, namun dapat

saja terjadi bahwa peristiwa-peristiwa ini demikian hebatnya sehingga menyebabkan gerakan-gerakan siklis atau hal lain yang baru.

#### 2.1.4. Logika Fuzzy

Secara umum logika *fuzzy* adalah suatu logika yang memiliki nilai kekaburan atau kesamaran (*fuzzyness*) antara benar atau salah. Logika *Fuzzy* memungkinkan nilai keanggotaannya antara 0 dan 1. Ada beberapa definisi tentang logika *fuzzy*, diantaranya:

- a. Logika *fuzzy* adalah suatu cara yang tepat untuk memetakan suatu ruang *input* ke dalam suatu ruang *output*, mempunyai nilai kontinyu dan logika *fuzzy* dinyatakan dalam derajat dari suatu keanggotaan dan derajat dari kebenaran (Kusumadewi, 2002).
- b. Logika *fuzzy* adalah logika yang digunakan untuk menjelaskan keambiguan, dimana logika *fuzzy* adalah cabang teori dari himpunan *fuzzy*, himpunan yang menyesuaikan keambiguan.
- c. Logika *fuzzy* memungkinkan nilai keanggotaan antara 0 dan 1, tingkat keabuan dan juga hitam dan putih, dan dalam bentuk linguistik, konsep tidak pasti seperti “sedikit”, “lumayan” dan “sangat” (Zadeh, 1965).
- d. Logika *fuzzy* menyediakan suatu cara untuk merubah pernyataan linguistik menjadi suatu numerik (Synaptic, 2006).

Beberapa alasan digunakannya logika *fuzzy*, antara lain (Kusumadewi dan Purnomo, 2004):

1. Konsep logika *fuzzy* mudah dimengerti, karena di dalam logika *fuzzy* terdapat konsep matematis sederhana dan mudah dimengerti yang mendasari penalaran *fuzzy*.
2. Logika *fuzzy* sangat fleksibel, artinya mampu beradaptasi dengan perubahan-perubahan, dan ketidakpastian yang menyertai permasalahan.
3. Logika *fuzzy* memiliki toleransi terhadap data yang tidak tepat.
4. Logika *fuzzy* mampu memodelkan fungsi–fungsi nonlinier yang sangat kompleks.

5. Logika *fuzzy* dapat bekerjasama dengan teknik–teknik kendali secara konvensional.
6. Logika *fuzzy* didasarkan pada bahasa alami. Logika *fuzzy* menggunakan bahasa sehari-hari sehingga mudah dimengerti.
7. Logika *fuzzy* dapat membangun dan mengaplikasikan pengalaman pengalaman para pakar secara langsung tanpa harus melalui proses pelatihan.

Ada beberapa hal yang menjadi lingkup dari sistem *fuzzy*, yaitu (Kusumadewi dan Purnomo, 2004):

1. Variabel *fuzzy*

Variabel *fuzzy* merupakan variabel yang hendak dibahas dalam suatu sistem *fuzzy*. Contoh: umur, temperatur, permintaan, dsb.

2. Himpunan *fuzzy*

Himpunan *fuzzy* merupakan suatu grup yang mewakili suatu kondisi atau keadaan tertentu dalam suatu variabel *fuzzy*.

Contoh:

- Variabel jarak, terbagi menjadi 3 himpunan *fuzzy*, yaitu: DEKAT, SEDANG dan JAUH.

3. Semesta Pembicaraan

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, tiga objek berupa bilangan positif maupun negatif. Adakalanya nilai semesta pembicaraan ini tidak dibatasi batas atasnya.

Contoh:

- Semesta pembicaraan untuk variabel umur:  $[0 + \infty)$
- Semesta pembicaraan untuk variabel temperatur:  $[0 40]$

4. Domain

Domain himpunan *fuzzy* adalah keseluruhan nilai yang diijinkan 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 :

- MUDA =  $[0, 45]$
- PAROBAYA =  $[35, 55]$
- TUA =  $[45, +\infty)$

#### 2.1.4.1. Himpunan *Fuzzy*

Himpunan *fuzzy* adalah himpunan-himpunan yang akan dibicarakan pada suatu variabel dalam sistem *fuzzy* (Kusumadewi, 2002). Himpunan *fuzzy* digunakan untuk mengantisipasi nilai – nilai yang bersifat tidak pasti. Pada himpunan tegas (*crisp*), nilai keanggotaan suatu item dalam suatu himpunan dapat memiliki dua kemungkinan, yaitu satu (1), yang berarti bahwa suatu item menjadi anggota dalam suatu himpunan, atau nol (0), yang berarti suatu item tidak menjadi anggota dalam suatu himpunan. Sedangkan pada himpunan *fuzzy* nilai keanggotaan terletak pada rentang 0 sampai 1, yang berarti himpunan *fuzzy* dapat mewakili interpretasi tiap nilai berdasarkan pendapat atau keputusan dan probabilitasnya.

Himpunan *fuzzy* memiliki dua atribut, yaitu:

1. Linguistik, yaitu penamaan suatu grup yang mewakili suatu keadaan atau kondisi tertentu dengan menggunakan bahasa alami, seperti: DEKAT, SEDANG, JAUH.
2. Numeris, yaitu suatu nilai (angka) yang menunjukkan ukuran dari suatu variabel seperti: 40, 25, 50 dan sebagainya.

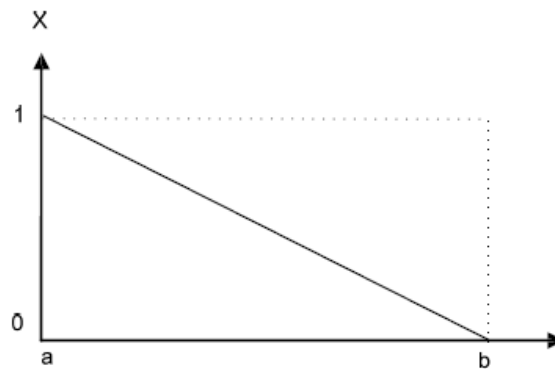
*Fuzzifikasi* merupakan suatu proses untuk mengubah suatu variabel *input* bentuk *crisp* menjadi variabel linguistik dalam bentuk himpunan-himpunan *fuzzy* dengan fungsi keanggotaannya masing-masing.

#### 2.1.4.2. Fungsi Derajat Keanggotaan *Fuzzy*

Fungsi derajat keanggotaan (*membershi function*) adalah suatu kurva yang menunjukkan pemetaan titik-titik *input* data ke dalam derajat keanggotaan yang memiliki *interval* antara 0 sampai 1.

Untuk mendapatkan derajat keanggotaan *fuzzy* digunakan pendekatan fungsi. Ada beberapa fungsi keanggotaan yang dapat digunakan, seperti fungsi linier turun, fungsi linier naik, fungsi segitiga, fungsi trapesium, fungsi-S, fungsi-Z.

Suatu fungsi derajat keanggotaan *fuzzy* disebut fungsi linier turun jika mempunyai 2 parameter, yaitu  $a, b \in \mathbb{R}$ . Pada linier turun, 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 dengan fungsi keanggotaan. Kurva fungsi linier turun diperlihatkan oleh gambar 2.5 (Kusumadewi dan Purnomo, 2004).

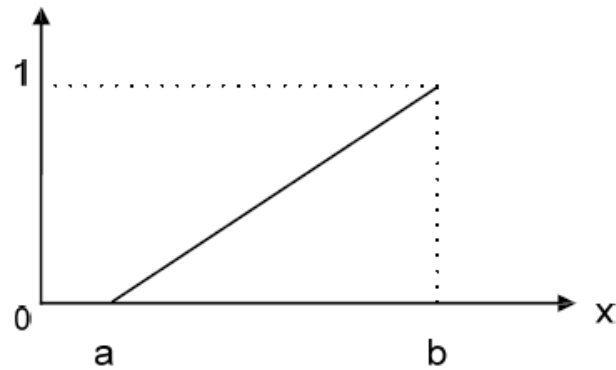


Gambar 2.5 Kurva Fungsi Linear Turun

Fungsi Keanggotaan dirumuskan dengan persamaan 2.1 berikut:

$$\mu[x] = \begin{cases} \frac{(b-x)}{(b-a)}; & a \leq x \leq b \\ 0; & x \geq b \end{cases} \quad (2.1)$$

Sedangkan suatu fungsi derajat keanggotaan *fuzzy* disebut fungsi *linier* naik jika mempunyai 2 parameter, yaitu  $a, b \in \mathbb{R}$ , dan kenaikan himpunan dimulai pada nilai domain yang memiliki derajat keanggotaan nol [0] bergerak menuju nilai domain yang memiliki derajat keanggotaan lebih tinggi. Kurva fungsi linier naik diperlihatkan oleh gambar 2.6 (Kusumadewi dan Purnomo, 2004).

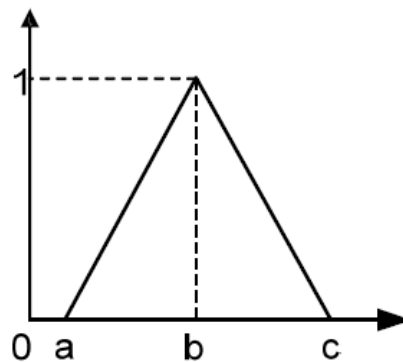


Gambar 2.6 Kurva Fungsi Linear Naik

Fungsi keanggotaan naik dirumuskan dengan persamaan 2.2 berikut:

$$\mu[x] = \begin{cases} 0; & x \leq a \\ \frac{(x-a)}{(b-a)}; & a \leq x \leq b \\ 1; & x \geq b \end{cases} \quad (2.2)$$

Suatu fungsi derajat keanggotaan *fuzzy* disebut fungsi segitiga jika mempunyai tiga buah parameter, yaitu  $a, b, c \in \mathbb{R}$  yang menentukan koordinat  $x$  dari tiga sudut. Kurva ini pada dasarnya merupakan gabungan antara dua garis (*linier*). Kurva fungsi segitiga diperlihatkan oleh gambar 2.7 (Kusumadewi dan Purnomo, 2004).



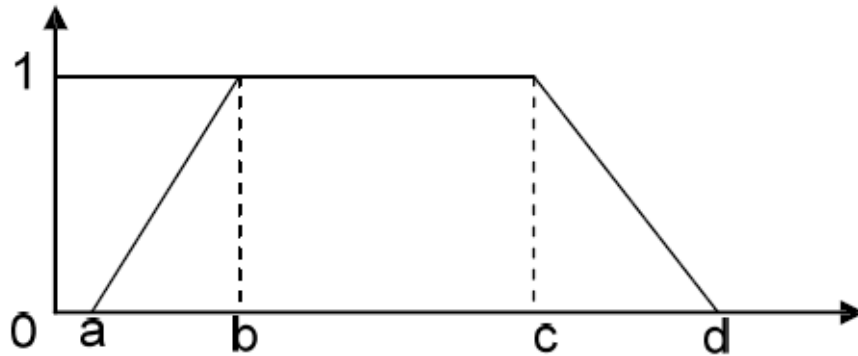
Gambar 2.7 Kurva Segitiga

Fungsi keanggotaan dirumuskan dengan persamaan 2.3 berikut:

$$\mu[x] = \begin{cases} 0; & x \leq a \text{ atau } x \geq c \\ \frac{(x-a)}{(b-a)}; & a \leq x \leq b \\ \frac{(c-x)}{(c-b)}; & b \leq x \leq c \end{cases} \quad (2.3)$$

Suatu fungsi derajat keanggotaan *fuzzy* disebut fungsi trapesium jika mempunyai 4 buah parameter ( $a, b, c, d \in \mathbb{R}$ ). Kurva trapesium pada dasarnya seperti bentuk

segitiga, hanya saja ada beberapa titik yang memiliki nilai keanggotaan 1. Dan Kurva fungsi trapesium diperlihatkan oleh gambar 2.8 (Kusumadewi dan Purnomo, 2004).

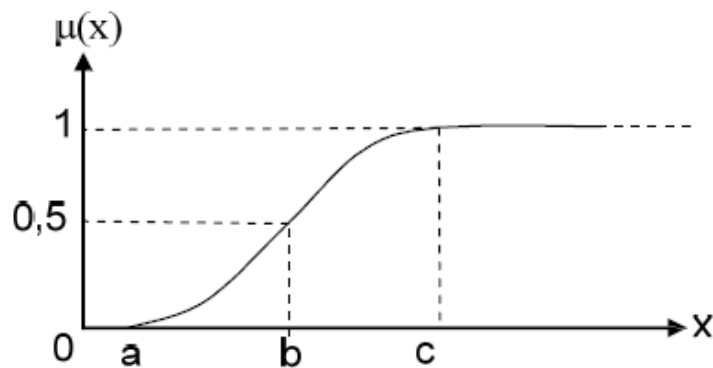


Gambar 2.8 Kurva Trapesium

Fungsi keanggotaan dirumuskan dengan persamaan 2.4 berikut:

$$\mu[x] = \begin{cases} 0; & x \leq a \text{ atau } x \geq d \\ \frac{(x-a)}{(b-a)}; & a \leq x \leq b \\ 1; & b \leq x \leq c \\ \frac{(d-x)}{(d-c)}; & x \geq d \end{cases} \quad (2.4)$$

Suatu derajat keanggotaan *fuzzy* disebut derajat keanggotaan fungsi-S Pertumbuhan jika mempunyai 3 buah parameter yaitu  $a, b, c \in \mathbb{R}$  dengan  $a$  adalah nilai keanggotaan nol,  $b$  adalah titik tengah antara  $a$  dan  $c$  dengan  $\mu(b) = 0.5$  (titik infleksi) dan  $c$  adalah nilai keanggotaan lengkap, Bentuk kurva fungsi-S Pertumbuhan diperlihatkan oleh gambar 2.9.



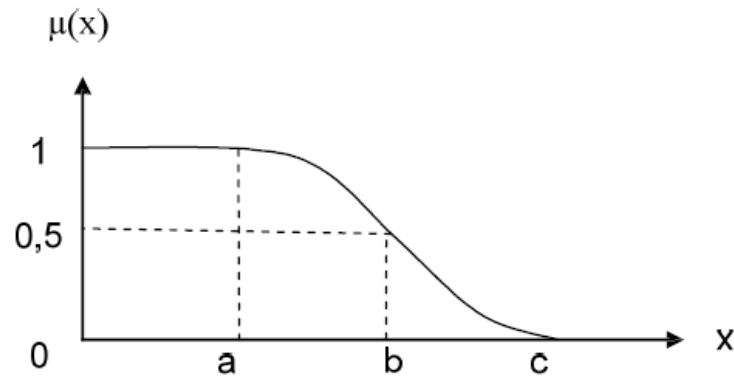
Gambar 2.9 Kurva Fungsi-S Pertumbuhan

Fungsi keanggotaan pada kurva-S Pertumbuhan dirumuskan dengan persamaan 2.5 berikut:



$$S(x; \alpha, \beta, \gamma) = \begin{cases} 0 & x \leq \alpha \\ 2((x - \alpha)/(\gamma - \alpha))^2 & \alpha \leq x \leq \beta \\ 1 - 2((\gamma - x)/(\gamma - \alpha))^2 & \beta \leq x \leq \gamma \\ 1 & x \geq \gamma \end{cases} \quad (2.5)$$

Suatu keanggotaan *fuzzy* disebut fungsi keanggotaan fungsi-S penyusutan jika mempunyai 3 buah parameter yaitu  $a, b, c \in \mathbb{R}$  dengan  $a$  adalah nilai keanggotaan nol,  $b$  adalah titik tengah antara  $a$  dan  $c$  dengan  $\mu(b) = 0.5$  (titik infleksi) dan  $c$  adalah nilai keanggotaan lengkap (Kusumadewi, 2002), Kurva fungsi S Penyusutan diperlihatkan oleh gambar 2.10.



Gambar 2.10 Kurva Fungsi-S Penyusutan

Fungsi keanggotaan pada kurva-S Penyusutan dirumuskan dengan persamaan 2.6 berikut:

$$S(x; \alpha, \beta, \gamma) = \begin{cases} 1 & x \leq \alpha \\ 1 - 2((x - \alpha)/(\gamma - \alpha))^2 & \alpha \leq x \leq \beta \\ 2((\gamma - x)/(\gamma - \alpha))^2 & \beta \leq x \leq \gamma \\ 0 & x \geq \gamma \end{cases} \quad (2.6)$$

#### 2.1.4.3. Operator Fuzzy

Nilai keanggotaan sebagai hasil dari operasi dua himpunan disebut dengan *fire strength* atau  $\alpha$ -predikat. Ada tiga operator dasar yang diciptakan oleh Zadeh, yaitu (Kusumadewi, 2002):

##### 1. Operator NOT

Misalnya  $A$  adalah suatu himpunan tegas dalam semesta  $X$ , maka komplemen dari  $A$ , yaitu  $A'$ . Operator ini berhubungan dengan operasi komplemen pada himpunan.  $\alpha$ -predikat sebagai hasil operasi

NOT diperoleh dengan mengurangi nilai keanggotaan elemen pada himpunan yang bersangkutan dari satu.

$$\mu A' = 1 - \mu A[x]$$

## 2. Operator OR

Gabungan dari himpunan-himpunan tegas  $A$  dan  $B$  dalam semesta  $X$ , yaitu  $A \cup B$ . Operator ini berhubungan dengan operasi union pada himpunan.  $\alpha$ -predikat sebagai hasil operasi OR diperoleh dengan mengambil nilai keanggotaan terbesar antar elemen pada himpunan-himpunan yang bersangkutan.

$$\mu A \cup B = \max(\mu A[x], \mu B[y])$$

## 3. Operator AND

Irisan dari himpunan-himpunan tegas  $A$  dan  $B$  dalam semesta  $X$ , yaitu  $A \cap B$ . Operator ini berhubungan dengan operasi interseksi pada himpunan.  $\alpha$ -predikat sebagai hasil operasi AND diperoleh dengan mengambil nilai keanggotaan terkecil antar elemen pada himpunan-himpunan yang bersangkutan.

$$\mu A \cap B = \min(\mu A[x], \mu B[y])$$

### 2.1.4.4. Penegasan (*Defuzzy*)

*Defuzzyfikasi* merupakan langkah terakhir dalam sistem logika *fuzzy* dimana tujuannya adalah mengkonversi setiap hasil dari *inference engine* yang diekspresikan dalam bentuk *fuzzy set* ke suatu bilangan real. *Input* dari proses *defuzzyfikasi* adalah suatu himpunan *fuzzy* yang diperoleh dari komposisi aturan-aturan *fuzzy*, sedangkan *output* yang dihasilkan merupakan suatu bilangan pada domain himpunan *fuzzy* tersebut. Sehingga jika diberikan suatu himpunan *fuzzy* dalam range tertentu, maka harus dapat diambil suatu nilai crisp tertentu sebagai *output*.

### 2.1.5. Peramalan dengan Metode *Fuzzy Time Series* (FTS)

*Fuzzy Time Series* (FTS) adalah metode peramalan data yang menggunakan prinsip-prinsip *fuzzy* sebagai dasarnya. Sistem peramalan dengan *fuzzy time series* menangkap pola dari data yang telah lalu kemudian digunakan untuk memproyeksikan data yang akan datang. Pertama kali dikembangkan oleh

Song and Chissom pada tahun 1993. Metode ini sering digunakan oleh para peneliti untuk menyelesaikan masalah peramalan.

#### 2.1.5.1. Definisi *Fuzzy Time Series* oleh Chen

Perbedaan utama antara *fuzzy time series* dan konvensional *time series* yaitu pada nilai yang digunakan dalam peramalan, yang merupakan himpunan *fuzzy* dari bilangan-bilangan *real* atas himpunan semesta yang ditentukan. Himpunan *fuzzy* dapat diartikan sebagai suatu kelas bilangan dengan batasan yang samar.

Jika  $U$  adalah himpunan semesta,  $U = \{u_1, u_2, \dots, u_n\}$ , maka suatu himpunan

*fuzzy*  $A$  dari  $U$  didefinisikan sebagai  $A = \frac{f_A(u_1)}{(u_1)} + \frac{f_A(u_2)}{(u_2)} + \dots + \frac{f_A(u_n)}{(u_n)}$  dimana  $f_A$

adalah fungsi keanggotaan dari  $A, f_A: U \rightarrow [0,1]$  dan  $1 \leq i \leq n$ .

Sedangkan definisi dari *fuzzy time series* adalah misalkan  $Y(t) (t = \dots, 0, 1, 2, \dots)$ , adalah himpunan *fuzzy*  $f_i(t) (t = \dots, 0, 1, 2, \dots)$ . Maka  $F(t)$  dinyatakan sebagai *fuzzy time series* terhadap  $Y(t) (t = \dots, 0, 1, 2, \dots)$ .

Dari definisi di atas, dapat dilihat bahwa  $F(t)$  bisa dianggap sebagai variabel linguistik dan  $f_i(t) (t = \dots, 0, 1, 2, \dots)$  bisa dianggap sebagai kemungkinan nilai linguistik dari  $F(t)$ , dimana  $f_i(t) (t = \dots, 0, 1, 2, \dots)$  direpresentasikan oleh suatu himpunan *fuzzy*. Bisa dilihat juga bahwa  $F(t)$  adalah suatu fungsi waktu dari  $t$  misalnya, nilai-nilai dari  $F(t)$  bisa berbeda pada waktu yang berbeda tergantung pada kenyataan bahwa himpunan semesta bisa berbeda pada waktu yang berbeda. Dan jika  $F(t)$  hanya disebabkan oleh  $F(t-1)$  maka hubungan ini digambarkan sebagai  $F(t-1) \rightarrow F(t)$ .

#### 2.1.6. Alat Bantu Perancangan Sistem

Dalam merancang suatu sistem terdapat banyak hal yang harus diperhatikan sehingga perlu digunakan alat bantu untuk memodelkan aplikasi yang akan dibuat. Terdapat banyak bentuk model yang dapat digunakan dalam perancangan sebuah sistem antara lain model narasi, *prototype*, model grafis atau

diagram dan lain sebagainya. Dalam hal ini, tidak menjadi masalah model mana yang akan digunakan asalkan pemodelan yang dibuat harus mampu mempresentasikan visualisasi bentuk sistem yang diinginkan pemakai, karena sistem akhir yang dibuat bagi pemakai akan diturunkan dari model. Pada dunia pemodelan sistem terdapat sejumlah cara mempresentasikan sistem melalui diagram misalnya *data flow diagram (DFD)* dan *Entity Relationship Diagram (ERD)*.

#### **2.1.6.1. Data Flow Diagram (DFD)**

Diagram alir data adalah diagram yang digunakan untuk memodelkan sistem secara logik. Seperti halnya bagan alir dokumen, diagram alir data pun dapat digunakan baik pada tahap analisis maupun tahap desain, namun kecenderungan diagram ini lebih cocok digunakan untuk tahap desain karena dengan diagram tersebut batasan ruang lingkup sistem terlihat sangat jelas sehingga pekerjaan pengembangan sistem yang dilakukan dapat lebih fokus.

Pemodelan *DFD* diawali dengan pembuatan *context diagram*. Secara simbol, *DFD* dan *context diagram* menggunakan jenis dan bentuk simbol yang sama, namun secara aturan terdapat perbedaan antara pemodelan *DFD* dan *context diagram*, dimana pada *context diagram* hanya diizinkan sebuah simbol proses saja sedangkan pada *DFD* dapat lebih dari satu simbol proses. Selain itu *context diagram* ditujukan untuk menggambarkan interaksi sistem dengan lingkungan luar, sedangkan pada *DFD* juga digambarkan hubungan antar proses didalam sistem.

Dalam pembuatan *DAD* atau *DFD* terdapat 3 level, yaitu:

##### **1) Diagram Konteks**

Diagram konteks menggambarkan satu lingkaran besar yang dapat mewakili seluruh proses yang terdapat didalam suatu sistem. Diagram konteks sering dikatakan sebagai diagram nomol 0 (nol), karena diagram ini adalah tinggkatan tertinggi dalam *DFD*. Diagram ini sangat sederhana untuk diciptakan karena pada diagram konteks sama sekali tidak memuat penyimpanan. Hal itu dilakukan karena semua entitas eksternal yang ditujukan pada diagram konteks yang berisii aliran-aliran data utama menuju dan dari sistem.

2) Diagram Nol (diagram level-1)

Diagram level nol adalah pemecahan dari diagram konteks, diagram ini memuat penyimpanan data.

3) Diagram Rinci

Adalah diagram yang digunakan untuk menguraikan atau pemecahan proses yang ada dalam diagram nol.

#### 2.1.6.2. *Entity Relationship Diagram (ERD)*

Menurut Sutanta (2011) “*Entity Relationship Diagram (ERD)* adalah suatu model data yang dikembangkan berdasarkan objek.” *Entity Relationship Diagram (ERD)* digunakan untuk menjelaskan hubungan antar data dalam basis data kepada pengguna secara logis. *Entity Relationship Diagram (ERD)* didasarkan pada suatu persepsi bahwa real world terdiri atas obyek-obyek dasar tersebut. Penggunaan *Entity Relationship Diagram (ERD)* relatif mudah dipahami, bahkan oleh para pengguna yang awam. Bagi perancang atau analis sistem, *Entity Relationship Diagram (ERD)* berguna untuk memodelkan sistem yang nantinya, basis data akan dikembangkan. Model ini juga membantu perancang atau analis sistem pada saat melakukan analisis dan perancangan basis data karena model ini dapat menunjukkan macam data yang dibutuhkan dan korelasi antar data di dalamnya.

#### 2.1.6.3. *Basis Data (Database)*

Menurut Raharjo (201), *database* yaitu kumpulan data yang terintegrasi dan diatur sedemikian rupa sehingga data tersebut dapat dimanipulasi, diambil, dan dicari secara cepat.

Menurut Raharjo (2011), selain berisi data, *database* juga berisi metadata yaitu data yang menjelaskan tentang struktur dari data itu sendiri, contoh: informasi yang diperoleh tentang nama-nama kolom dan tipe data yang ada pada sebuah tabel, data nama kolom dan tipe yang ditampilkan tersebut disebut metadata.

Menurut Raharjo (2011), *primary key* yaitu suatu aturan yang berguna untuk memastikan bahwa setiap baris data di dalam suatu tabel bersifat

unik/berbeda antara baris yang satu dengan yang lainnya sedangkan *foreign key* berguna untuk mendefinisikan kolom-kolom pada suatu tabel yang nilainya mengacu ke tabel lain.

#### **2.1.6.4. Structured Query Language (SQL)**

Menurut Raharjo (2011), SQL yaitu kependekan dari *Structured Query Language*, adalah bahasa atau kumpulan perintah standar yang digunakan untuk berkomunikasi dengan *database*.

#### **2.1.6.5. MySQL**

Menurut Raharjo (2011), mysql mengimplementasikan model *database* relasional maka disebut sebagai *Relational Database Management System/RDBMS*.

Menurut Raharjo (2011), mysql adalah software *RDBMS* atau *server database* yang dapat mengelola *database* dengan sangat cepat, dapat menampung data dalam jumlah sangat besar, dapat diakses oleh banyak user/*multi-user*, dan dapat melakukan suatu proses secara sinkron atau berbarengan/*multi-threaded*.

Menurut Raharjo (2011), *mysql* banyak digunakan di berbagai kalangan untuk melakukan penyimpanan dan pengolahan data, mulai dari kalangan akademis sampai ke industri, baik industri kecil, menengah, maupun besar.

#### **2.1.6.6. Embarcadero RAD Studio Delphi XE5**

*Delphi* adalah *IDE (Integrated Development Environment)* yang berbasis *Windows* dan mengolah bahasa pemrograman *Pascal* serta menyediakan fasilitas pembuatan aplikasi visual seperti *Visual Basic*. *Delphi* memberikan kemudahan dalam menggunakan kode program, kompilasi yang cepat, penggunaan file unit ganda untuk pemrograman modular, pengembangan perangkat lunak, pola desain yang menarik serta diperkuat dengan bahasa pemrograman yang terstruktur dalam bahasa pemrograman objek *Pascal*.

*Delphi* memiliki tampilan khusus yang didukung suatu lingkup kerja komponen *Delphi* untuk membangun suatu aplikasi dengan menggunakan *Visual*

*Component Library (VCL)*. Sebagian besar pengembang *Delphi* menuliskan dan mengkompilasi kode program dalam *IDE (Integrated Development Environment)*.

*Delphi* adalah salah satu *IDE* yang sangat populer dikarenakan bahasa yang digunakan adalah turunan dari bahasa *Pascal*. *Delphi* juga adalah *IDE* yang berjalan di *platform windows*, yang paling populer adalah *Delphi 7*. Seiring berjalannya waktu *Delphi* mengalami perubahan yang sangat signifikan, semuanya berawal saat *Borland* menjual produknya ke pihak *embarcadero*. Dipihak *embarcadero*, *Delphi* mulai memuncak dengan rilisnya versi terbaru, bahkan dalam beberapa tahun terakhir ini *Delphi* sudah mengalami beberapa perubahan versi mulai dari *Delphi 2010*, *Delphi 2011 (XE)*, *Delphi XE2*, *Delphi XE3*, *Delphi XE4* dan yang paling teranyar adalah *Delphi XE5* yang mampu berjalan di *multi-platform* seperti *windows*, *android* dan *Ios*.

## 2.2. Pengujian Sistem

### 2.2.1. Metode Mean Average Percentage Error (MAPE)

*Mean Average Percentage Error (MAPE)* dihitung dengan menggunakan kesalahan absolut pada tiap periode dibagi dengan nilai observasi yang nyata untuk periode itu dan kemudian nilai tersebut dirata-ratakan. MAPE mengindikasikan seberapa besar kesalahan dalam meramal yang dibandingkan dengan nilai nyata.

$$MAPE = \frac{100\%}{n} \sum_{t=1}^n \left| \frac{X_t - \hat{X}_t}{X_t} \right| \quad (2.7)$$

dengan

$n$  = banyak data

$X_t$  = data observasi nyata pada waktu  $t$

$\hat{X}_t$  = data hasil peramalan pada waktu  $t$

## 2.3. Studi Literatur

Beberapa penelitian terdahulu yang telah melakukan penelitian tentang curah hujan salah satunya yaitu Fleydi Novisnky Mandey (2017) yang telah meneliti tentang peramalan curah hujan menggunakan logika *Fuzzy* dengan model sistem penalaran metode Mamdani. Penelitian ini menggunakan *software MatLab R2009a* untuk melakukan perhitungan curah hujan dengan menggunakan logika

*Fuzzy* model penalaran Mamdani. Penelitian ini menghitung curah hujan berdasarkan faktor-faktor yang dapat mempengaruhi curah hujan yakni suhu, kelembaban, tekanan, dan kecepatan angin. Pengujian pada hasil peramalan menggunakan perhitungan *Root Mean Square Error* (RMSE) dengan melibatkan sebanyak 23 data dari semua kategori intensitas prediksi yakni cerah, hujan ringan, hujan sedang, hujan lebat, dan hujan sangat lebat.

Selain itu, Aria Bayu Elfajar (2017) juga melakukan penelitian menggunakan peramalan *Fuzzy Time Series* pada wisatawan Kota Batu. Penelitian ini berfokus pada peramalan wisatawan Kota Batu dengan menggunakan pola data dari data yang telah lalu kemudian digunakan untuk memproyeksikan data yang akan datang. Pengujian pada penelitian ini menggunakan perhitungan *After Forecasting Error Rate* (AFER) dengan melibatkan 60 data latih.

Penelitian yang akan dilakukan ini berfokus pada peramalan besaran curah hujan dengan menggunakan metode peramalan *Fuzzy Time Series* dengan menggunakan data historis curah hujan periode 1 Januari 2003 hingga 31 Desember 2017. Pengujian pada penelitian ini akan menggunakan perhitungan *Mean Average Percentage Error* (MAPE) yang menghitung seberapa besar rata-rata kesalahan dalam meramal yang dibandingkan dengan nilai nyata pada periode tertentu. Tujuan dari penelitian yang dilakukan adalah menghasilkan sebuah aplikasi yang dapat mengolah dan menentukan pola menggunakan metode *Fuzzy Time Series* serta menggunakan pola data tersebut untuk memprediksi besaran curah hujan.



## **BAB III**

### **METODOLOGI PENELITIAN DAN PERANCANGAN SISTEM**

#### **3.1. Metodologi Penelitian**

##### **3.1.1. Data Penelitian**

Data penelitian berupa data historis curah hujan periode 1 Januari 2003 hingga 31 Desember 2017. Data historis curah hujan yang digunakan merupakan data Stasiun Supadio yang diperoleh dari Badan Meteorologi, Klimatologi, dan Geofisika (BMKG) Kalimantan Barat serta diperoleh dari 3 buah pos hujan yakni Pontianak (PTK-11), Padang Tikar (PTK-16), dan Desa Ambawang Kuala (PTK-22) yang diperoleh dari Balai Wilayah Sungai Kalimantan I (BWSK I).

##### **3.1.2. Alat Penelitian**

###### **3.1.2.1. Alat Bantu Penelitian**

Alat bantu penelitian yang digunakan dalam penelitian ini adalah :

1. *Flowchart*, untuk menggambarkan jalannya algoritma atau proses pada sistem. *Flowchart* dapat membantu perancangan alur logika program sehingga dapat dengan mudah mengetahui kesalahan dalam suatu program.
2. *Data Flow Diagram (DFD)*, untuk menggambarkan aliran data pada sistem.
3. *Entity Relationship Diagram (ERD)*, untuk menjelaskan hubungan antar data dalam basis data berdasarkan objek-objek dasar data yang mempunyai hubungan antar relasi.
4. Metode *Fuzzy Time Series*, untuk menentukan pola data *time series* dan memproyeksikan pola tersebut untuk proses *forecasting*.

###### **3.1.2.2. Perangkat Lunak**

Perangkat lunak yang digunakan dalam penelitian ini adalah :

1. Sistem operasi Windows 10

2. XAMPP Version 5.6.32
3. Embarcadero RAD Studio Delphi XE5
4. Browser Mozilla Firefox

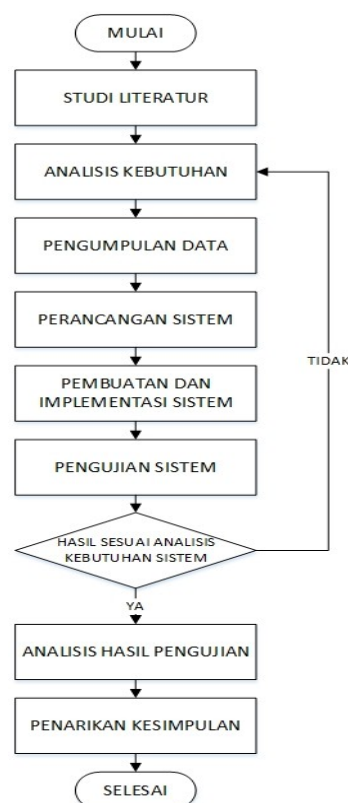
### 3.1.2.3. Perangkat Keras

Perangkat keras yang digunakan dalam penelitian ini adalah laptop ASUS tipe A455L dengan spesifikasi :

1. Intel Core i5-4210U
2. Harddisk 500 GB
3. DDR 4 GB
4. LCD 14" HD LED LCD

## 3.2. Metode Penelitian

Langkah-langkah penelitian yang dilakukan dalam pembuatan aplikasi dapat dilihat pada Gambar 3.1 berikut :



Gambar 3.1. Diagram Alir Penelitian

### 3.2.1. Studi Literatur

Studi literatur dilakukan dengan mencari referensi ilmu pengetahuan yang berhubungan dengan sistem yang akan dibuat seperti referensi tentang kritik dan saran, DFD (*Data Flow Diagram*), teknik peramalan (*forecasting*), *Fuzzy Time Series* maupun melalui penelitian-penelitian sebelumnya.

### **3.2.2. Analisis Kebutuhan**

Analisis kebutuhan merupakan langkah awal untuk menentukan gambaran perangkat yang akan dihasilkan ketika pengembang melaksanakan sebuah proyek pembuatan perangkat lunak. Adapun data yang dibutuhkan adalah data historis curah hujan pada Stasiun Supadio periode 1 Januari 2003 hingga 31 Desember 2017 y. Data historis curah hujan yang digunakan merupakan data Stasiun Supadio yang diperoleh dari Badan Meteorologi, Klimatologi, dan Geofisika (BMKG) Kalimantan Barat serta diperoleh dari 3 buah pos hujan yakni Pontianak (PTK-11), Padang Tikar (PTK-16), dan Desa Ambawang Kuala (PTK-22) yang diperoleh dari Balai Wilayah Sungai Kalimantan I (BWSK I). Data tersebut kemudian akan diolah dan diproses menggunakan metode *Fuzzy Time Series* untuk menganalisa pola data dan menginterpretasikan pola tersebut untuk memprediksi besaran curah hujan mendatang.

### **3.2.3. Pengumpulan Data**

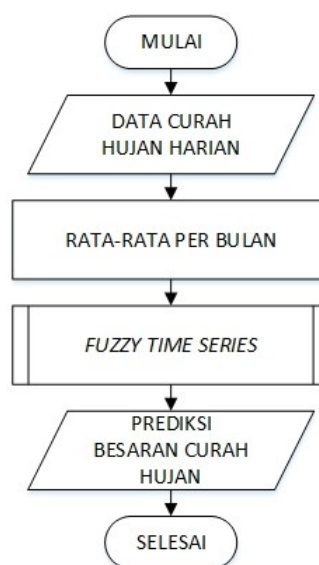
Pengumpulan data dilakukan untuk memperoleh informasi mengenai analisis sistem, analisis kelayakan, dan analisis fungsional. Data historis curah hujan diperoleh dari Badan Meteorologi, Klimatologi, dan Geofisika (BMKG) Kalimantan Barat. Data tersebut berasal dari hasil pengukuran dan pencatatan dari Stasiun Meteorologi Kelas II Supadio Pontianak serta diperoleh dari 3 buah pos hujan yakni Pontianak (PTK-11), Padang Tikar (PTK-16), dan Desa Ambawang Kuala (PTK-22) yang diperoleh dari Balai Wilayah Sungai Kalimantan I (BWSK I).

### 3.2.4. Perancangan Sistem

Metode perancangan sistem terdiri dari perancangan alur algoritma aplikasi / *flowchart*, perancangan arsitektur sistem, perancangan konseptual dari sistem yang dibangun, perancangan basis data dan perancangan antarmuka.

### 3.2.5. Perancangan Aplikasi

Perancangan aplikasi yang akan dibuat dapat dilihat pada Gambar 3.2 berikut :

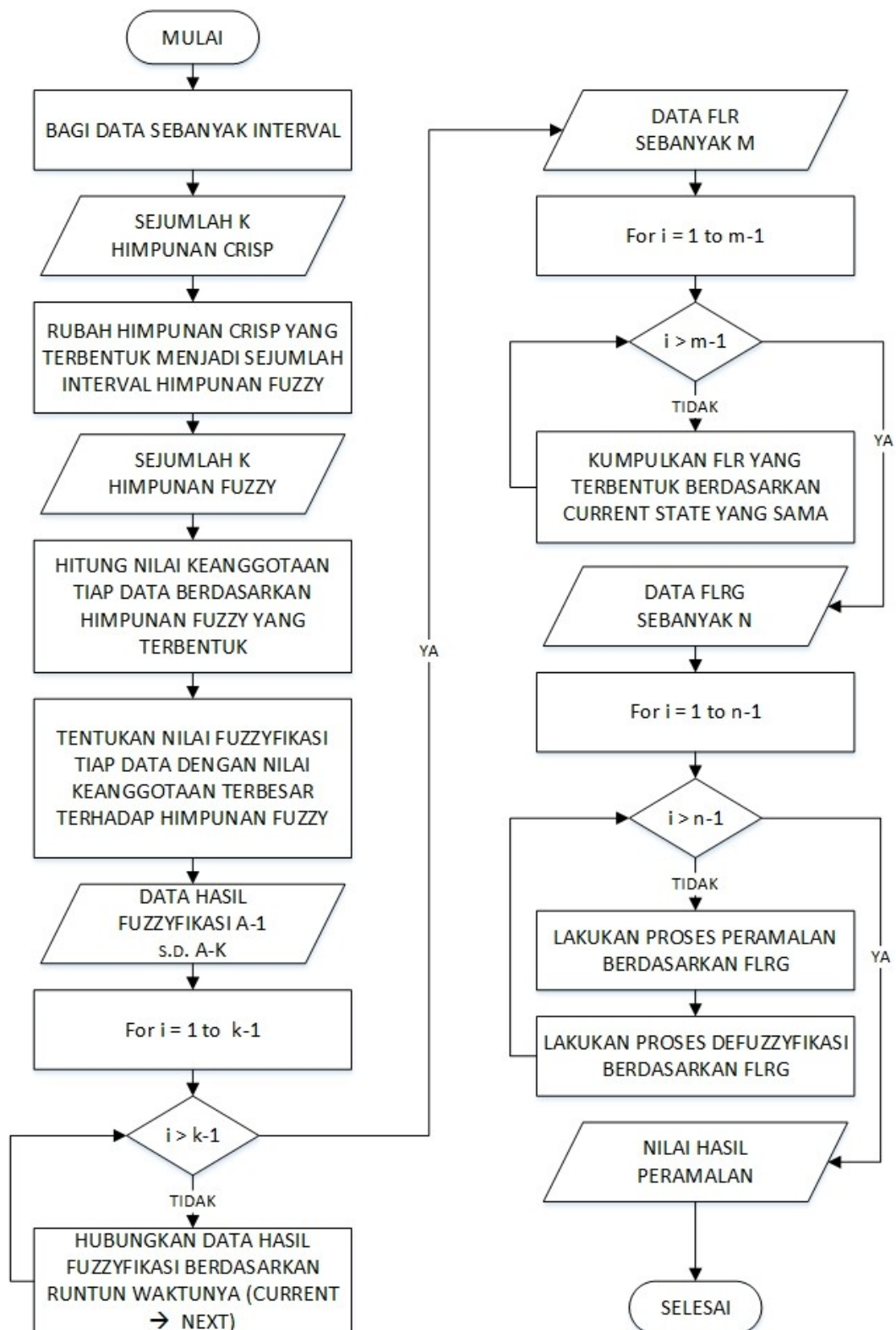


Gambar 3.2. Diagram Alir Aplikasi

Alur jalannya aplikasi pada Gambar 3.2 dijelaskan sebagai berikut :

- User akan memasukkan data curah hujan harian melalui menu *input data*.
- Selanjutnya data akan disimpan di *database*.
- Data curah hujan harian akan dihitung rata-rata per bulan dan data rata-rata per bulan akan disimpan dalam *database*.
- Data curah hujan rata-rata per bulan akan dihitung polanya menggunakan perhitungan *Fuzzy Time Series*.
- Pola data akan diproyeksikan untuk mendapatkan prediksi besaran curah hujan pada periode mendatang.

Adapun diagram alir untuk proses perhitungan *Fuzzy Time Series* dapat dilihat pada Gambar 3.3 berikut :



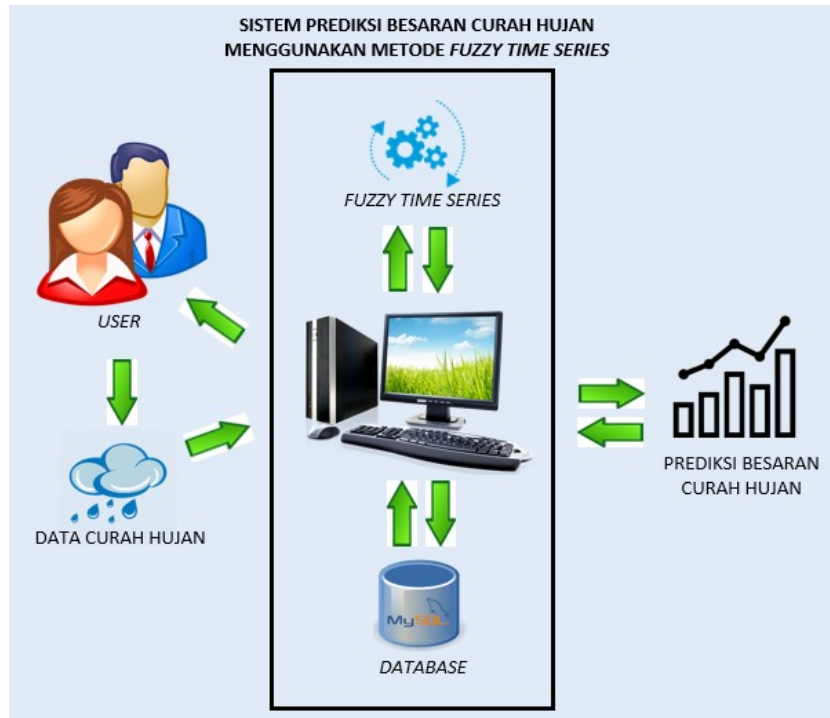
Gambar 3.3. Diagram Alir Proses *Fuzzy Time Series*

Penjelasan untuk diagram alir proses *Fuzzy Time Series* pada Gambar 3.3 adalah sebagai berikut :

- Seluruh data dibagi rata menjadi sejumlah himpunan *crisp* sesuai dengan banyaknya jumlah interval efektif yang digunakan.
- Bentuk sejumlah himpunan *fuzzy* berdasarkan himpunan *crisp* yang terbentuk sesuai dengan jumlah interval.
- Hitung nilai keanggotaan masing-masing data *time series* terhadap masing-masing himpunan *fuzzy* yang terbentuk.
- Tentukan nilai fuzzifikasi masing-masing data *time series* dengan nilai keanggotaan terbesar.
- Hubungkan data hasil fuzzifikasi berdasarkan runtun waktunya dari *current\_state*  $\rightarrow$  *next\_state*.
- Kelompokkan data *fuzzy logic relationship* yang terbentuk berdasarkan nilai *current\_state* yang sama.
- Lakukan proses *forecasting* berdasarkan kelompok *fuzzy logic relationship group* yang terbentuk dan telah didefuzzifikasikan.

### 3.2.6. Arsitektur Sistem

Perancangan arsitektur sistem yang akan dibuat dapat dilihat pada Gambar 3.4 berikut :



Gambar 3.4. Arsitektur Sistem

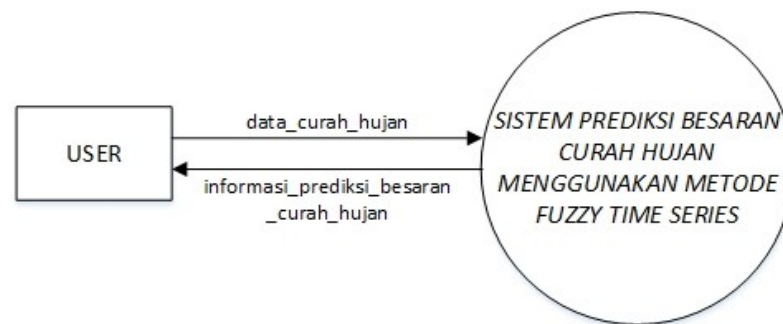
Arsitektur sistem pada Gambar 3.4 dijelaskan sebagai berikut :

- User memasukkan data curah hujan ke dalam sistem. Data tersebut selanjutnya akan disimpan ke dalam *database*.
- Sistem akan menentukan pola data curah hujan menggunakan perhitungan *Fuzzy Time Series*.
- Pola data curah hujan yang diperoleh melalui perhitungan *Fuzzy Time Series* akan diproyeksikan untuk memprediksi besaran curah hujan pada periode mendatang.

### 3.2.7. Perancangan Konseptual dari Sistem yang Dibangun

#### 3.2.7.1. Diagram Konteks

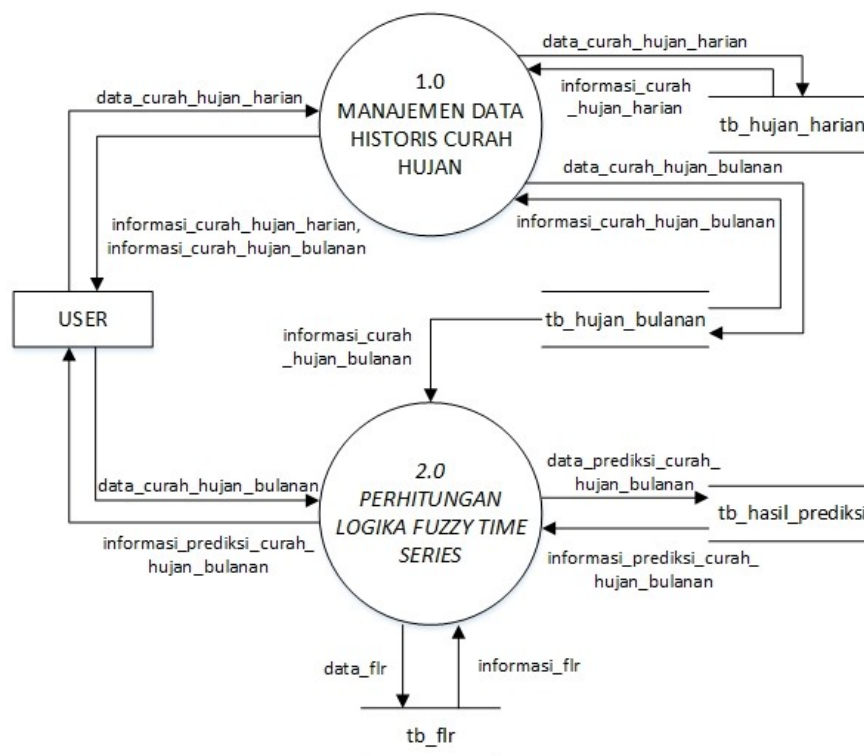
Gambar Diagram Konteks dari sistem prediksi besaran curah hujan menggunakan metode *Fuzzy Time Series* dapat dilihat pada Gambar 3.5.



Gambar 3.5. Diagram Konteks Sistem

#### 3.2.7.2. Diagram Overview

Diagram Overview merupakan gambaran rinci dari proses yang dilakukan pada diagram konteks. Arus data dapat dilihat dengan jelas pada diagram tersebut. Proses-proses yang terjadi digambarkan pada Gambar 3.6.



Gambar 3.6. Diagram Overview Sistem

Penjelasan Diagram Overview sistem pada Gambar 3.6 sebagai berikut :



1. Proses 1.0, yaitu *User* dapat melakukan proses memasukkan, mengolah, atau menghapus data historis curah hujan harian. Pada proses ini juga akan dilakukan perhitungan rata-rata curah hujan per-bulan dan data rata-rata tersebut kemudian akan digunakan untuk perhitungan prediksi *Fuzzy Time Series* pada proses berikutnya.
2. Proses 2.0, yaitu proses perhitungan menggunakan metode *Fuzzy Time Series*. Pada proses ini akan dilakukan perhitungan panjang interval kelas, proses fuzzyfikasi, serta proses perhitungan *forecasting* berdasarkan data curah hujan bulanan.

### 3.2.7.3. Diagram Rinci

Diagram rinci menguraikan lebih lanjut mengenai proses dari diagram overview, yang memperlihatkan arus data masuk dan arus data keluar. Berdasarkan Gambar 3.6, maka diagram rinci pada Gambar 3.7 dijelaskan sebagai berikut :

1. Proses 1.0, yaitu Proses Manajemen Data Historis Curah Hujan. Proses ini terbagi menjadi 2 proses rinci :
  - a. Proses 1.1 :
 

Nama Proses : CRUD Data Historis Curah Hujan Harian

Masukan : data curah hujan harian

Keluaran : informasi curah hujan harian

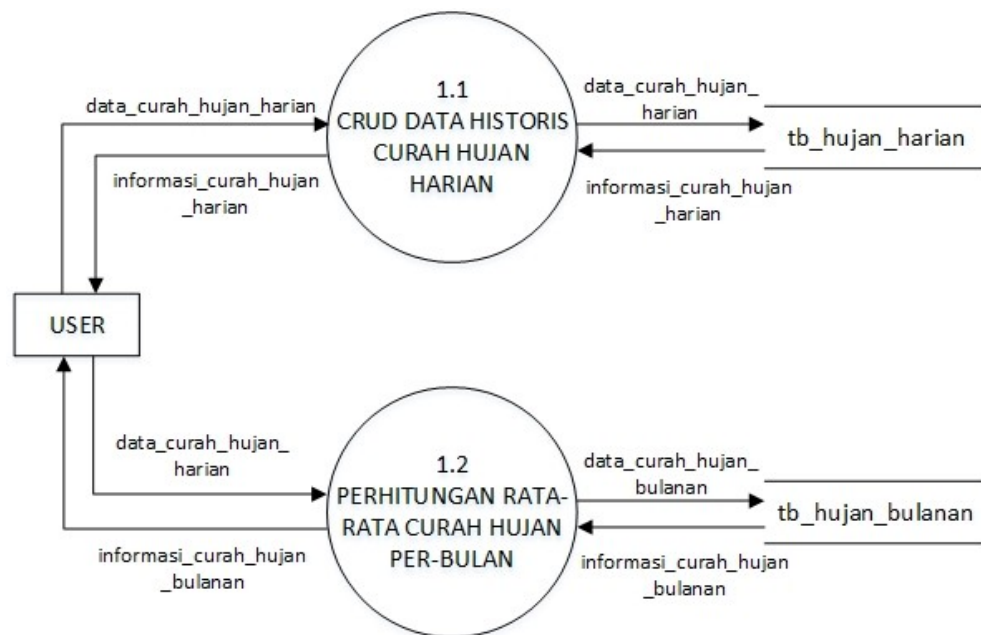
Keterangan : *User* dapat memasukkan, mengubah, dan menghapus data curah hujan harian dan data tersebut disimpan ke dalam tabel hujan harian dalam *database*.
  - b. Proses 1.2 :
 

Nama Proses : Perhitungan Rata-rata Curah Hujan Per-bulan

Masukan : data curah hujan harian

Keluaran : informasi curah hujan bulanan

Keterangan : Proses ini berfungsi untuk menghitung rata-rata curah hujan pada setiap bulan berdasarkan data curah hujan harian.



Gambar 3.7. Diagram Rinci 1.0

2. Proses 2.0, yaitu Proses Perhitungan Logika *Fuzzy Time Series*. Proses ini dapat dilihat pada Gambar 3.8 dan terbagi menjadi 2 proses rinci dan :

a. Proses 2.1 :

Nama Proses : Perhitungan Interval Berbasis Rata-rata

Masukan : data curah hujan bulanan

Keluaran : informasi interval kelas

Keterangan : Proses ini menghitung panjang interval kelas yang akan digunakan dalam perhitungan *Fuzzy Time Series* dengan menggunakan rata-rata selisih antardata pada data curah hujan bulanan.

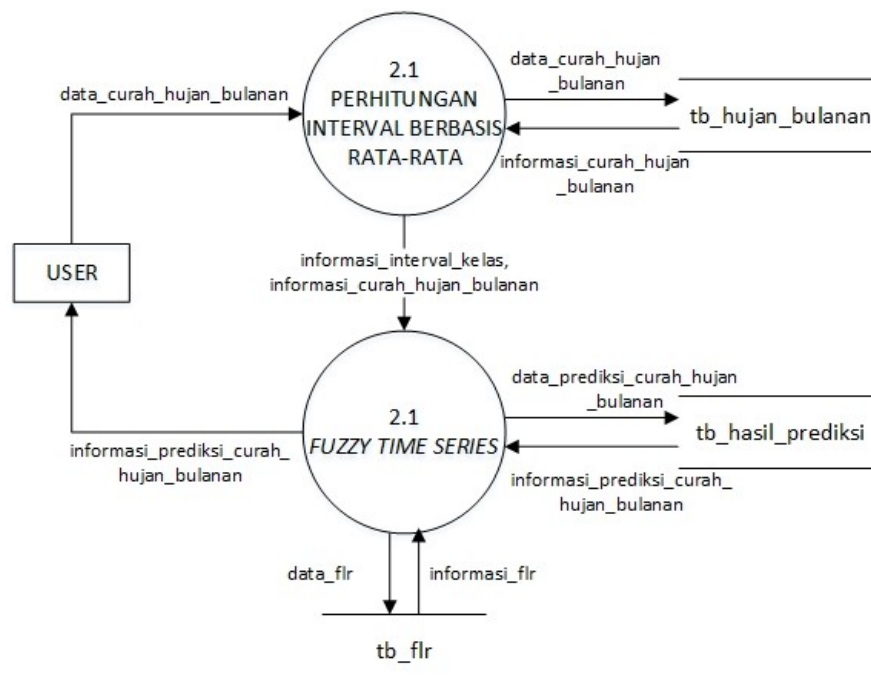
b. Proses 2.2 :

Nama Proses : *Fuzzy Time Series*

Masukan : informasi interval kelas, data curah hujan bulanan

Keluaran : informasi prediksi besaran curah hujan bulanan

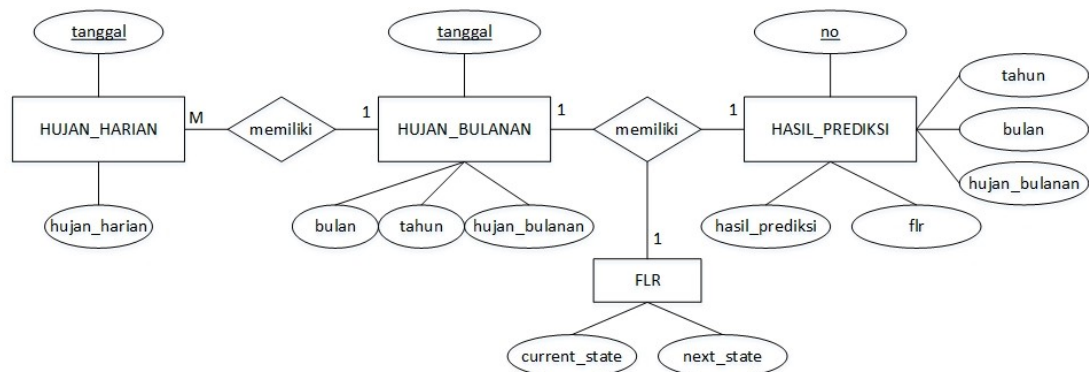
Keterangan : Proses ini menghitung prediksi besaran curah hujan berdasarkan interval kelas serta informasi curah hujan bulanan.



Gambar 3.8. Diagram Rinci 2.0

### 3.2.8. Perancangan Basis Data

#### 3.2.8.1. Perancangan *Entity Relationship Diagram*

Gambar 3.9. *Entity Relationship Diagram*

Penjelasan perancangan ERD pada Gambar 3.9 sebagai berikut :

- Setiap data pada entitas hujan harian memiliki dua atribut yakni tanggal sebagai *primary key* dan curah hujan harian. Entitas hujan harian memiliki hubungan *many-to-one* terhadap entitas hujan bulanan.
- Entitas hujan bulanan memiliki empat atribut yakni tanggal sebagai *foreign key* bawaan dari entitas hujan harian. Atribut lainnya adalah tahun dan bulan dari data *time series*, serta curah hujan bulanan.

Entitas hujan bulanan memiliki hubungan *one-to-one* dengan entitas FLR dan hubungan *one-to-one* dengan entitas hasil prediksi.

- Entitas hasil prediksi memiliki enam atribut yakni no sebagai *primary key*, tahun, bulan, hujan bulanan, flr dan hasil prediksi. Entitas hasil prediksi memiliki hubungan *one-to-one* dengan entitas FLR.

### 3.2.8.2. Spesifikasi Tabel Basis Data

Semua tabel yang ada dalam *database* sistem prediksi besaran curah hujan akan dijelaskan secara spesifik mulai dari nama *field*, *type*, keterangan dan fungsinya. Berikut tabel-tabel yang ada pada tabel prediksi besaran curah hujan.

Nama Tabel : tb\_data\_historis

Keterangan : tabel data historis hujan harian

**Tabel 3.1** Spesifikasi Tabel Data Historis Hujan Harian

Nama Field	Tipe	Fungsi
tanggal (PK)	date	Menyimpan tanggal dalam tabel hujan harian
hujan_harian	double (4,1)	Menyimpan curah hujan per-hari

Nama Tabel : tb\_data\_average

Keterangan : tabel data historis hujan bulanan

**Tabel 3.2** Spesifikasi Tabel Data Historis Hujan Bulanan

Nama Field	Tipe	Fungsi
tanggal (FK)	date	Menyimpan tanggal dalam tabel hujan bulanan
tahun	int(4)	Menyimpan tahun data rata-rata
bulan	int(2)	Menyimpan bulan data rata-rata
hujan_bulanan	double (5,2)	Menyimpan rata-rata curah hujan per-bulan

Nama Tabel : tb\_flr

Keterangan : tabel *fuzzy logic relationship*

**Tabel 3.3** Spesifikasi Tabel *Fuzzy Logic Relationship*

<b>Nama Field</b>	<b>Tipe</b>	<b>Fungsi</b>
current	int(3)	Menyimpan current state dari fuzzy logic relationship data bulanan
next	int(3)	Menyimpan next state dari fuzzy logic relationship data bulanan

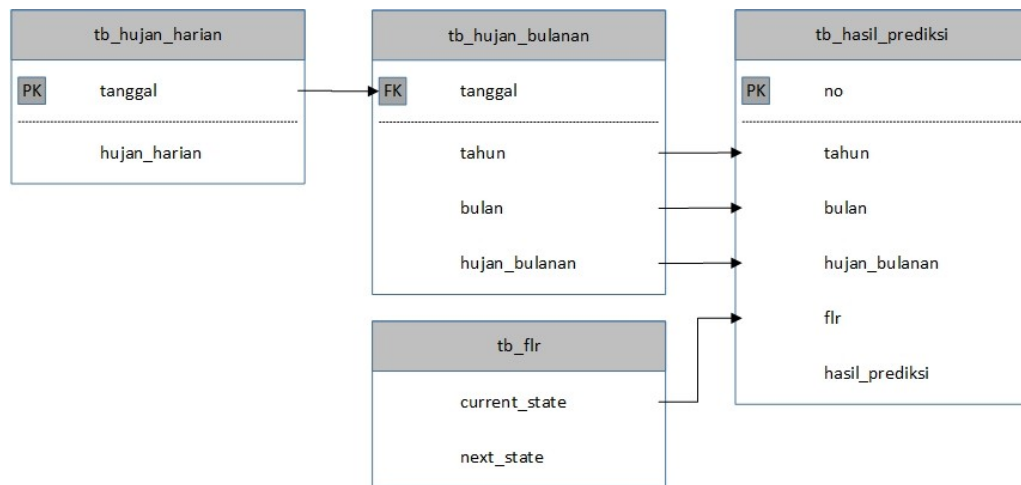
Nama Tabel : tb\_hasil

Keterangan : tabel hasil prediksi besaran curah hujan bulanan

**Tabel 3.4** Spesifikasi Tabel Hasil Prediksi Besaran Curah Hujan Bulanan

<b>Nama Field</b>	<b>Tipe</b>	<b>Fungsi</b>
no (PK)	int(3)	Menyimpan nomor data
tahun	int(4)	Menyimpan tahun data yang bersangkutan
bulan	int(2)	Menyimpan bulan data yang bersangkutan
data	double(5,2)	Menyimpan curah hujan bulanan
flr	varchar(3)	Menyimpan current state dari fuzzy logic relationship data bulanan
hasil	double (5,2)	Menyimpan hasil prediksi curah hujan per-bulan

### 3.2.8.3. Hubungan Antar Tabel



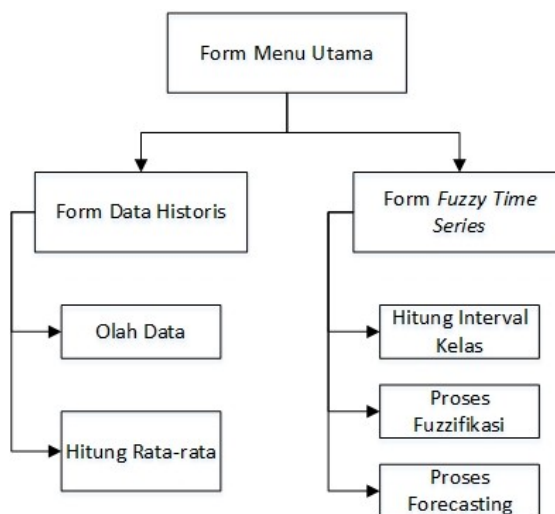
Gambar 3.10. Hubungan Antar Tabel

### 3.2.9. Perancangan Antarmuka

Antarmuka merupakan media interaksi antar *User* dengan sistem. Berikut perancangan *layout* sistem yang akan dibangun.

#### 3.2.9.1. Perancangan Struktur Antarmuka Sistem

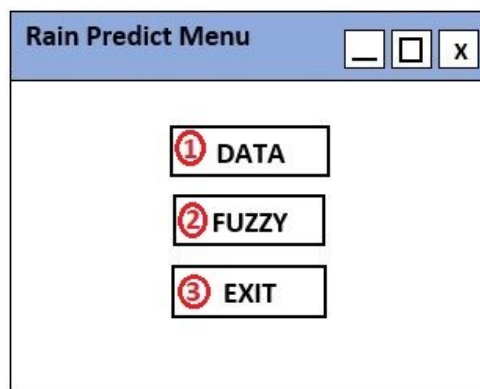
Antarmuka aplikasi dirancang dalam bentuk form-form yang memiliki fungsi tertentu sesuai dengan proses-proses yang ada. Struktur antarmuka aplikasi yang dirancang pada Gambar 3.11 berikut:



Gambar 3.11. Struktur Antarmuka Sistem

#### 3.2.9.2. Layout Antarmuka Halaman Utama

Halaman utama adalah halaman yang ditampilkan paling pertama setelah *user* membuka atau menjalankan aplikasi. Pada halaman utama terdapat tombol menu untuk mengakses proses olah data dan perhitungan *fuzzy time series*. Perancangan tampilan halaman utama dapat dilihat pada Gambar 3.12 berikut :



Gambar 3.12. Antarmuka Halaman Utama

Keterangan :

- (1) Tombol untuk mengakses Form Data Historis yang berfungsi untuk mengolah data historis serta untuk menghitung rata-rata data per-bulan.
- (2) Tombol untuk mengakses Form Fuzzy Time Series yang berfungsi untuk menghitung interval data, melakukan perhitungan *fuzzy time series*, serta melakukan proses *forecasting*.
- (3) Tombol keluar untuk menutup aplikasi.

### 3.2.9.3. Layout Antarmuka Form Data Historis

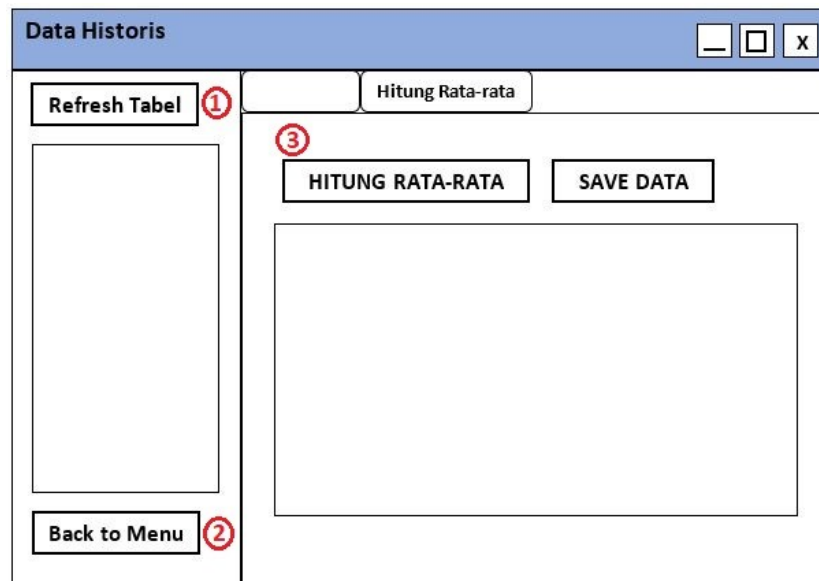
Form Data Historis memiliki 2 panel terpisah yakni, panel Olah Data dan panel Hitung Rata-rata. Panel Olah Data berfungsi untuk menginput, mengubah, dan menghapus data, sedangkan Panel Hitung Rata-rata berfungsi untuk menghitung rata-rata data per-bulan serta menyimpannya ke dalam basis data. Perancangan tampilan Form Data Historis dapat dilihat pada Gambar 3.13 dan Gambar 3.14 berikut :

Gambar 3.13. Antarmuka Panel Olah Data

Keterangan :

- (1) Bagian untuk menampilkan seluruh data historis curah hujan harian. Bagian ini terdiri dari sebuah tombol untuk memperbaharui tampilan tabel dan sebuah *dbgrid* untuk menampilkan data berupa tabel.
- (2) Tombol untuk keluar dari Form Data Historis dan kembali ke tampilan menu utama pada Halaman Utama.
- (3) Bagian untuk menginputkan data baru berupa tanggal dan curah hujan (mm) pada tanggal tersebut.
- (4) Bagian untuk mengubah dan/atau menghapus data dengan cara memilih data pada tabel atau dengan memilih tanggal tertentu.





Gambar 3.14. Antarmuka Panel Hitung Rata-rata

Keterangan :

- (1) Bagian untuk menampilkan seluruh data historis curah hujan harian. Bagian ini terdiri dari sebuah tombol untuk memperbaharui tampilan tabel dan sebuah *dbgrid* untuk menampilkan data berupa tabel.
- (2) Tombol untuk keluar dari Form Data Historis dan kembali ke tampilan menu utama pada Halaman Utama.
- (3) Bagian untuk menjalankan proses perhitungan rata-rata per-bulan serta menampilkan hasil perhitungan berupa tabel pada sebuah *dbgrid*.

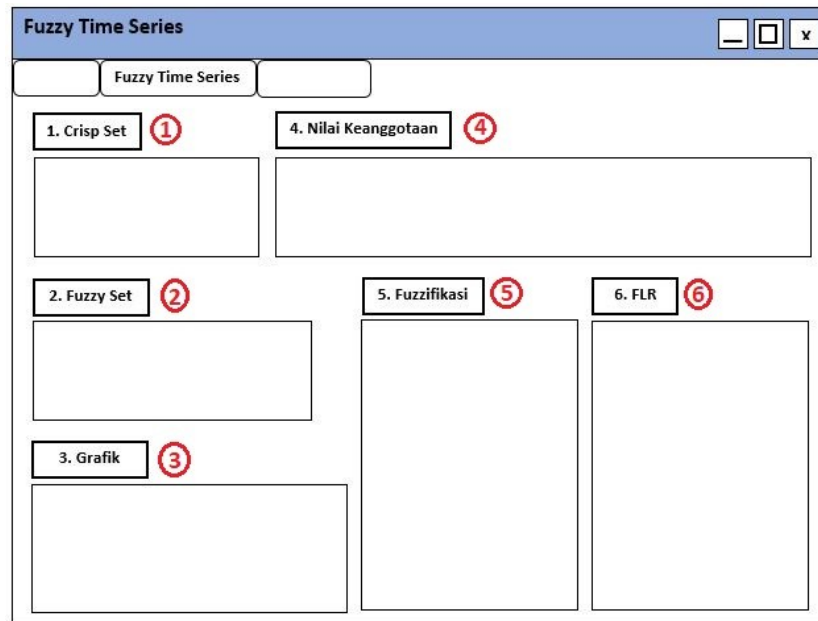
#### 3.2.9.4. Layout Antarmuka Form *Fuzzy Time Series*

Form *Fuzzy Time Series* memiliki 3 panel terpisah yakni, Panel Interval, Panel *Fuzzy Time Series*, serta Panel *Forecasting*. Panel Interval berfungsi untuk menghitung interval rata-rata yang akan menjadi acuan interval ideal yang dapat digunakan pada proses perhitungan *Fuzzy Time Series* pada tahap berikutnya. Panel *Fuzzy Time Series* berfungsi untuk menghitung data-data secara bertahap sesuai dengan metode *Fuzzy Time Series*. Panel *Forecasting* berfungsi untuk melakukan proses *forecasting* atau peramalan berdasarkan hasil perhitungan *Fuzzy Time Series* pada tahap sebelumnya. Perancangan tampilan Form *Fuzzy Time Series* dapat dilihat pada Gambar 3.15, Gambar 3.16, dan Gambar 3.17 berikut :

Gambar 3.15. Antarmuka Panel Interval

Keterangan :

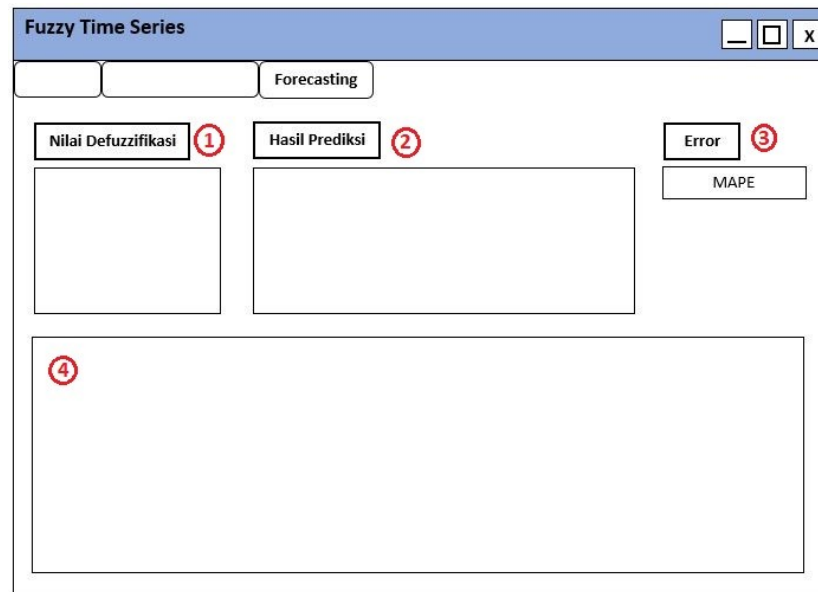
- (1) Bagian untuk memilih periode data *time series* yang akan digunakan untuk seluruh proses perhitungan selanjutnya.
- (2) Bagian untuk menghitung batasan data berupa jumlah data, data terbesar, serta data terkecil dalam periode yang telah ditentukan.
- (3) Bagian untuk menyimpan data dalam *array* serta menampilkannya pada sebuah memo.
- (4) Bagian untuk proses perhitungan interval data yang akan menjadi acuan interval ideal yang dapat digunakan pada proses perhitungan *Fuzzy Time Series* pada tahap selanjutnya.



Gambar 3.16. Antarmuka Panel *Fuzzy Time Series*

Keterangan :

- (1) Bagian untuk menghitung dan menampilkan himpunan *crisp*.
- (2) Bagian untuk menghitung dan menampilkan himpunan *fuzzy*.
- (3) Bagian untuk menampilkan grafik himpunan *fuzzy* yang terbentuk pada perhitungan sebelumnya.
- (4) Bagian untuk menghitung dan menampilkan nilai keanggotaan dari masing-masing data *time series* pada himpunan *fuzzy* yang terbentuk.
- (5) Bagian untuk menghitung nilai fuzzifikasi setiap data *time series* dengan nilai keanggotaan yang terbesar terhadap himpunan *fuzzy*.
- (6) Bagian untuk menghitung dan menampilkan *Fuzzy Logic Relationship* yang terbentuk pada setiap data *time series* berdasarkan perhitungan nilai keanggotaan data pada himpunan *fuzzy*.



Gambar 3.17. Antarmuka Panel *Forecasting*

Keterangan :

- (1) Bagian untuk menghitung dan menampilkan *Fuzzy Logic Relationship Group* dengan menghubungkan *current state* dengan *next state* pada masing-masing *Fuzzy Logic Relationship* yang telah dihitung pada tahap sebelumnya. Serta bagian untuk menghitung nilai defuzzifikasi dari setiap kelompok *Fuzzy Logic Relationship Group* yang terbentuk dan menampilkan hasilnya pada sebuah memo.
- (2) Bagian untuk menghitung dan menampilkan hasil prediksi nilai berdasarkan perhitungan *fuzzy time series* pada tahapan-tahapan sebelumnya.
- (3) Bagian untuk menghitung dan menampilkan nilai kesalahan (*error*) dengan membandingkan hasil prediksi yang didapat dengan data aktual yang digunakan.
- (4) Bagian untuk menampilkan grafik perbandingan antara data aktual dan data hasil proses prediksi.

### 3.2.9.5.

### 3.2.10. Contoh Perhitungan Manual

#### Langkah 1. Input Data

Misalkan data yang akan digunakan adalah data rata-rata curah hujan per-bulan dalam satuan mm. Tabel 3.5. adalah data aktual rata-rata curah hujan per-bulan Periode 2015-2017.

**Tabel 3.5** Data Aktual Curah Hujan (mm)

Waktu	Aktual	Waktu	Aktual	Waktu	Aktual
Jan-15	19.19	Jan-16	29.41	Jan-17	19.82
Feb-15	24.1	Feb-16	22.76	Feb-17	39.73
Mar-15	22	Mar-16	26.69	Mar-17	23.18
Apr-15	21	Apr-16	21.39	Apr-17	24.38
Mei-15	24.51	Mei-16	21.38	Mei-17	23.45
Jun-15	25.82	Jun-16	20.75	Jun-17	21.84
Jul-15	32.93	Jul-16	28.47	Jul-17	31.48
Agu-15	13.6	Agu-16	17.75	Agu-17	27.97
Sep-15	14.2	Sep-16	18.97	Sep-17	23.07
Okt-15	22.95	Okt-16	22.44	Okt-17	15.31
Nov-15	23.22	Nov-16	24.73	Nov-17	18.38
Des-15	24.34	Des-16	25.27	Des-17	15.89

**Langkah 2.** Definisikan *Universe of Discourse*  $U$  sampai dimana fuzzy set dapat ditetapkan.

Setelah data aktual tersebut dikalkulasi, maka didapatkan nilai minimal dan maximal dari data aktual tersebut sebagai ( $X_{\min} = 13.60$ ,  $X_{\max} = 39.73$ ). Berdasarkan nilai perbedaan tersebut, maka *Universe of Discourse*  $U$  dapat didefinisikan sebagai  $U = [13.60, 39.73]$ .

**Langkah 3.** Hitung interval efektif dengan menggunakan metode berbasis rata-rata.

Dari 60 (enam puluh) data pada tabel 3.6 diperoleh rata-rata selisih sebesar 4.987. Jika nilai 4.987 dibagi dua maka diperoleh nilai 2.494 yang jika

dirujuk pada Tabel 3.6 maka basis interval yang digunakan adalah 1. Kemudian nilai 2.494 dibulatkan berdasarkan basis sehingga diperoleh nilai 2 sebagai panjang interval efektif.

**Tabel 3.6** Tabel Basis Interval

Jangkauan	Basis
0.1 – 1.0	0.1
1.1 – 10	1
11 – 100	10
101 – 1000	100

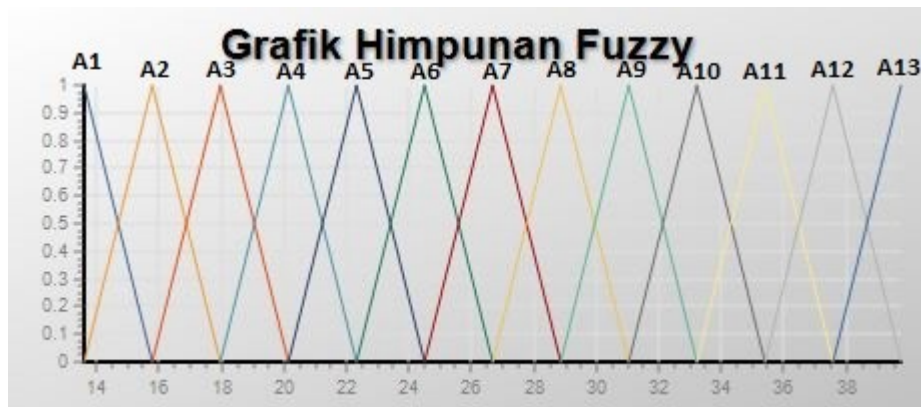
Dan jika 2 digunakan sebagai panjang interval untuk membagi himpunan semesta  $U$ , maka jumlah interval dapat diperoleh dari hasil bagi jangkauan dengan interval. Nilai jangkauan diperoleh dari selisih nilai maksimal dengan nilai minimal, yakni 39.73 (nilai maksimal) dikurangi 13.60 (nilai minimal) sama dengan 26.13. Nilai jangkauan sebesar 26.13 dibagi dengan panjang interval sebesar 2, maka diperoleh nilai 13.067. Dikarenakan jumlah interval haruslah bilangan ganjil, maka dibulatkan ke bilangan ganjil terdekat yaitu 13.

**Langkah 4.** Bagi *Universe of Discourse U* dengan beberapa seri data  $u_1, u_2, \dots, u_n$ , dan tentukan nilai linguistik.

Pertama *Universe of Discourse U* dibagi ke dalam 13 interval yang sama besar dengan cara membagi nilai jangkauan 26.13 dengan 13 dan diperoleh 2.01. Nilai tersebut dijadikan selisih antara batas atas dan batas bawah setiap himpunan.

$$\begin{aligned} u_1 &= [13.60, 15.61], & u_2 &= [15.61, 17.62], & u_3 &= [17.62, 19.63], \\ u_4 &= [19.63, 21.64], & u_5 &= [21.64, 23.65], & u_6 &= [23.65, 25.66], \\ u_7 &= [25.66, 27.67], & u_8 &= [27.67, 29.68], & u_9 &= [29.68, 31.69], \\ u_{10} &= [31.69, 33.70], & u_{11} &= [33.70, 35.71], & u_{12} &= [35.71, 37.72], \\ u_{13} &= [37.72, 39.73]. \end{aligned}$$

Kemudian tentukan 11 nilai linguistik yang membentuk 11 *fuzzy sets*  $A_1, A_2, A_3, A_4, A_5, A_6, A_7, A_8, A_9, A_{10}, A_{11}, A_{12}$ , dan  $A_{13}$  dalam *Universe of Discourse U*, dengan cara membagi nilai jangkauan 26.13 dengan jumlah interval dikurangi satu (13-1), maka diperoleh nilai 2.178. Nilai tersebut dijadikan selisih antara himpunan satu dengan himpunan lainnya. *Fuzzy sets* tersebut dapat digambarkan pada Gambar 3.18 berikut.



Gambar 3.18. Grafik *Fuzzy Sets*

$$\begin{aligned} A_1 &= 13.60; & A_2 &= 15.78; & A_3 &= 17.96; & A_4 &= 20.14; & A_5 &= 22.3; \\ A_6 &= 24.50; & A_7 &= 26.68; & A_8 &= 28.86; & A_9 &= 31.04; & A_{10} &= 33.22; \\ A_{11} &= 35.40, & A_{11} &= 37.58, & A_{11} &= 39.76. \end{aligned}$$

**Langkah 5.** Fuzzifikasi nilai dari data historis.

Dengan menggunakan fungsi nilai keanggotaan dari *fuzzy sets* yang telah terbentuk, maka dapat diperoleh hasil fuzzifikasi dari data historis yang digunakan.

Contoh perhitungan nilai keanggotaan :

Nilai 19.19 terletak diantara  $A_3$  dan  $A_4$ , maka dihitung nilai keanggotaan pada masing-masing  $A_3$  dan  $A_4$ . Berdasarkan Gambar 3.18, nilai 19.19 mengenai  $A_3$  pada kurva turun dan mengenai  $A_4$  pada kurva naik, maka sesuai aturan nilai keanggotaan *fuzzy*, didapatlah perhitungan sebagai berikut :

-  $A_3$  turun :

$$\frac{20.14 - x}{20.14 - 17.96} = \frac{20.14 - 19.19}{20.14 - 17.96} = \frac{0.95}{2.18} = 0.436$$

-  $A_4$  naik :

$$\frac{x - 17.96}{20.14 - 17.96} = \frac{19.19 - 17.96}{17.96 - 20.14} = \frac{1.23}{2.18} = 0.564$$

Dengan melihat kedua nilai tersebut, maka nilai 19.19 masuk ke dalam  $A_4$  karena memiliki nilai keanggotaan terbesar diantara himpunan  $A$  lainnya.

**Tabel 3.7** Nilai Keanggotaan *Fuzzy* Data Historis

Waktu	FTS	$A_1$	$A_2$	$A_3$	$A_4$	$A_5$	$A_6$	$A_7$	$A_8$	$A_9$	$A_{10}$	$A_{11}$
Jan-15	19.19	0	0	0.436	0.564	0	0	0	0	0	0	0
Feb-15	24.1	0	0	0	0	0.183	0.817	0	0	0	0	0
Mar-15	22	0	0	0	0.147	0.853	0	0	0	0	0	0
Apr-15	21	0	0	0	0.606	0.394	0	0	0	0	0	0
Mei-15	24.51	0	0	0	0	0	0.994	0.006	0	0	0	0
Jun-15	25.82	0	0	0	0	0	0.395	0.605	0	0	0	0
...	...	...	...	...	...	...	...	...	...	...	...	...
Jul-17	31.48	0	0	0	0	0	0	0	0	0.798	0.202	0
Agu-17	27.97	0	0	0	0	0	0	0.407	0.593	0	0	0
Sep-17	23.07	0	0	0	0	0.657	0.343	0	0	0	0	0
Okt-17	15.31	0.214	0.786	0	0	0	0	0	0	0	0	0
Nov-17	18.38	0	0	0.808	0.192	0	0	0	0	0	0	0
Des-17	15.89	0	0.950	0.050	0	0	0	0	0	0	0	0

**Tabel 3.8** Hasil Fuzzifikasi Data Historis



<b>Waktu</b>	<b>Aktual</b>	
<b>Jan-15</b>	19.19	A4
<b>Feb-15</b>	24.1	A6
<b>Mar-15</b>	22.00	A5
<b>Apr-15</b>	21.00	A4
<b>Mei-15</b>	24.51	A6
<b>Jun-15</b>	25.82	A7
<b>Jul-15</b>	32.93	A10
<b>Agu-15</b>	13.60	A1
<b>Sep-15</b>	14.20	A1
<b>Okt-15</b>	22.95	A5
<b>Nov-15</b>	23.22	A5
<b>Des-15</b>	24.34	A6
<b>Waktu</b>	<b>Aktual</b>	
<b>Jan-16</b>	29.41	A8
<b>Feb-16</b>	22.76	A5
<b>Mar-16</b>	26.69	A7
<b>Apr-16</b>	21.39	A5
<b>Mei-16</b>	21.38	A5
<b>Jun-16</b>	20.75	A4
<b>Jul-16</b>	28.47	A8
<b>Agu-16</b>	17.75	A3
<b>Sep-16</b>	18.97	A3
<b>Okt-16</b>	22.44	A5
<b>Nov-16</b>	24.73	A6
<b>Des-16</b>	25.27	A6
<b>Waktu</b>	<b>Aktual</b>	
<b>Jan-17</b>	19.82	A4
<b>Feb-17</b>	39.73	A13
<b>Mar-17</b>	23.18	A5
<b>Apr-17</b>	24.38	A6
<b>Mei-17</b>	23.45	A6
<b>Jun-17</b>	21.84	A5



<b>Jul-17</b>	31.48	A9
<b>Agu-17</b>	27.97	A8

<b>Sep-17</b>	23.07	A5
<b>Okt-17</b>	15.31	A2

<b>Nov-17</b>	18.38	A3
<b>Des-17</b>	15.89	A2

**Langkah 6.** Bentuk *Fuzzy Logic Relationship* (FLR) dari tabel fuzzifikasi berdasarkan urutan *time series*-nya sebagaimana ditampilkan pada tabel berikut.

**Tabel 3.9** *Fuzzy Logic Relationship* (FLR)

<i>Time Series</i>	<b>FLR</b>	<i>Time Series</i>	<b>FLR</b>
<b>Jan-15 → Feb-15</b>	A4 → A6	<b>Jul-16 → Agu-16</b>	A8 → A3
<b>Feb-15 → Mar-15</b>	A6 → A5	<b>Agu-16 → Sep-16</b>	A3 → A3
<b>Mar-15 → Apr-15</b>	A5 → A4	<b>Sep-16 → Okt-16</b>	A3 → A5
<b>Apr-15 → Mei-15</b>	A4 → A6	<b>Okt-16 → Nov-16</b>	A5 → A6
<b>Mei-15 → Jun-15</b>	A6 → A7	<b>Nov-16 → Des-16</b>	A6 → A6
<b>Jun-15 → Jul-15</b>	A7 → A10	<b>Des-16 → Jan-17</b>	A6 → A4
<b>Jul-15 → Agu-15</b>	A10 → A1	<b>Jan-17 → Feb-17</b>	A4 → A13
<b>Agu-15 → Sep-15</b>	A1 → A1	<b>Feb-17 → Mar-17</b>	A13 → A5
<b>Sep-15 → Okt-15</b>	A1 → A5	<b>Mar-17 → Apr-17</b>	A5 → A6
<b>Okt-15 → Nov-15</b>	A5 → A5	<b>Apr-17 → Mei-17</b>	A6 → A6
<b>Nov-15 → Des-15</b>	A5 → A6	<b>Mei-17 → Jun-17</b>	A6 → A5
<b>Des-15 → Jan-16</b>	A6 → A8	<b>Jun-17 → Jul-17</b>	A5 → A9
<b>Jan-16 → Feb-16</b>	A8 → A5	<b>Jun-17 → Agu-17</b>	A9 → A8
<b>Feb-16 → Mar-16</b>	A5 → A7	<b>Agu-17 → Sep-17</b>	A8 → A5
<b>Mar-16 → Apr-16</b>	A7 → A5	<b>Sep-17 → Okt-17</b>	A5 → A2
<b>Apr-16 → Mei-16</b>	A5 → A5	<b>Okt-17 → Nov-17</b>	A2 → A3
<b>Mei-16 → Jun-16</b>	A5 → A4	<b>Nov-17 → Des-17</b>	A3 → A2
<b>Jun-16 → Jul-16</b>	A4 → A8	<b>Des-17 → Jan-18</b>	---

**Langkah 7.** Membentuk *Fuzzy Logic Relationship Group* (FLRG) berdasarkan Tabel *Fuzzy Logic Relationship* (FLR).

*Fuzzy Logic Relationship Group* (FLRG) dibentuk dengan cara mengeliminasi FLR yang identik atau sama dan berulang, kemudian FLR dikelompokkan atau digabung menjadi satu grup berdasarkan LHS (*left hand side*) atau *current state* yang sama.

**Tabel 3.10** *Fuzzy Logic Relationship Group* (FLRG)

<i>Current State</i>	<i>Next State</i>
A1 →	A1, A5
A2 →	A3
A3 →	A2, A3, A5
A4 →	A6, A8, A13
A5 →	A2, A4, A5, A6, A7, A9
A6 →	A4, A5, A6, A7, A8
A7 →	A5, A10
A8 →	A3, A5
A9 →	A8
A10 →	A1
A11 →	-
A12 →	-
A13 →	A5

Nilai pada Tabel 3.11 didapatkan berdasarkan hasil penggolongan *current\_state* yang sama dari Tabel 3.10. Berdasarkan Tabel 3.10 tidak ditemukan adanya *next\_state* dari A11 dan A12 sehingga tidak terdapat *next\_state* dari A11 dan A12.

**Langkah 8.** Melakukan proses defuzzifikasi dan peramalan berdasarkan FLRG yang telah dibentuk.

Untuk mempermudah proses peramalan maka bisa dihitung terlebih dahulu semua nilai yang mungkin dari hasil fuzzifikasi untuk masing-masing grup.

Untuk grup dengan *current state* A2 hanya memiliki 1 *next\_state* maka hasil defuzzikasi peramalannya adalah  $\frac{A_2}{1}=17.96$  . Untuk grup dengan *current\_state* A1 maka hasil defuzzikasi peramalannya adalah  $\frac{A_1+A_5}{2}=\frac{13.60+22.32}{2}=\frac{35.92}{2}=17.96$  . Dan seterusnya untuk grup yang lain sebagaimana ditampilkan hasilnya pada tabel berikut.

**Tabel 3.11** Hasil Defuzzifikasi FLRG

<i>Current State</i>	<i>Forecasted</i>
A1 →	17.96
A2 →	17.96
A3 →	18.69
A4 →	31.04
A5 →	23.41
A6 →	24.50
A7 →	27.77
A8 →	20.14
A9 →	28.86
A10 →	13.60
A11 →	35.40
A12 →	37.58
A13 →	22.32

### 3.2.11. Pengujian dan Validasi

Setelah hasil defuzzifikasi tiap grup telah dihitung, maka proses peramalan sudah dapat dilakukan untuk tiap data yang ada, sebagaimana ditampilkan pada Tabel 3.12.

**Tabel 3.12** Hasil Perhitungan Manual Peramalan *Fuzzy Time Series*

Waktu	Aktual		Forecasted	Waktu	Aktual		Forecasted
Jan-15	19.19	A4	31.04	Jul-16	28.47	A8	20.14
Feb-15	24.10	A6	24.50	Agu-16	17.75	A3	18.69
Mar-15	22.00	A5	23.41	Sep-16	18.97	A3	18.69
Apr-15	21.00	A4	31.04	Okt-16	22.44	A5	23.41
Mei-15	24.51	A6	24.50	Nov-16	24.73	A6	24.50
Jun-15	25.82	A7	27.77	Des-16	25.27	A6	24.50
Jul-15	32.93	A10	13.60	Jan-17	19.82	A4	31.04
Agu-15	13.60	A1	17.96	Feb-17	39.73	A13	22.32
Sep-15	14.20	A1	17.96	Mar-17	23.18	A5	23.41
Okt-15	22.95	A5	23.41	Apr-17	24.38	A6	24.50
Nov-15	23.22	A5	23.41	Mei-17	23.45	A6	24.50
Des-15	24.34	A6	24.50	Jun-17	21.84	A5	23.41
Jan-16	29.41	A8	20.14	Jul-17	31.48	A9	28.86
Feb-16	22.76	A5	23.41	Agu-17	27.97	A8	20.14
Mar-16	26.69	A7	27.77	Sep-17	23.07	A5	23.41
Apr-16	21.39	A5	23.41	Okt-17	15.31	A2	17.96
Mei-16	21.38	A5	23.41	Nov-17	18.38	A3	18.69
Jun-16	20.75	A4	31.04	Des-17	15.89	A2	17.96

### 3.2.12. Analisis Hasil Pengujian

Analisis dilakukan dalam tahap pengujian dan validasi untuk mengetahui karakteristik sistem dan mengidentifikasi jika terdapat ketidakkonsistenan sistem. Hasil analisis juga digunakan sebagai dasar perbaikan.

### 3.2.13. Penarikan Kesimpulan

Kesimpulan dirumuskan berdasarkan pengujian yang telah dilakukan apakah sistem yang dirancang mampu memberikan informasi prediksi curah hujan yang tepat.

## BAB IV

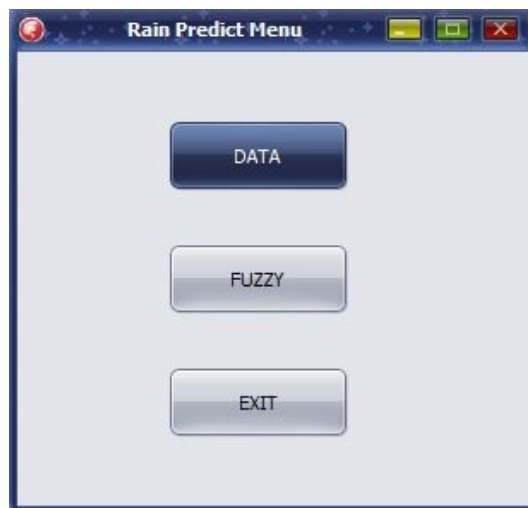
### HASIL DAN PEMBAHASAN

#### 4.1. Hasil Perancangan dan Implementasi Program

Sistem yang dirancang merupakan sebuah aplikasi yang dapat menghitung dan memprediksi besaran curah hujan berdasarkan data historis curah hujan dengan menggunakan aturan perhitungan metode *Fuzzy Time Series*. Aplikasi terdiri dari tiga bagian utama yaitu Form Halaman Utama yang menampilkan menu utama dari aplikasi, Form Data Historis yang berfungsi untuk menjalankan proses *input* dan olah data serta menghitung rata-rata data, dan Form *Fuzzy Time Series* yang berfungsi untuk menentukan pola data menggunakan metode *Fuzzy Time Series* serta menghitung hasil prediksi berdasarkan hasil perhitungan pola data tersebut.

##### 4.1.1. Form Halaman Utama

Form Halaman Utama berfungsi untuk menampilkan pilihan menu pada aplikasi untuk mengakses Form Data Historis atau Form *Fuzzy Time Series*. Antarmuka Form Halaman Utama dapat dilihat pada Gambar 4.1.



Gambar 4.1. Antarmuka Form Halaman Utama



#### 4.1.2. Form Data Historis

Form Data Historis memiliki dua panel utama yakni Panel Olah Data dan Panel Hitung Rata-rata. Panel Olah Data berfungsi sebagai tempat untuk memasukkan data baru, mengubah data yang telah ada, serta menghapus data. Panel Hitung Rata-rata berfungsi untuk menghitung rata-rata curah hujan perbulan dengan cara menjumlahkan curah hujan dalam sebulan dibagi jumlah hari hujan dalam bulan tersebut. Antarmuka Panel Olah Data dan Panel Hitung Rata-rata dapat dilihat pada Gambar 4.2 dan Gambar 4.3.

Gambar 4.2. Antarmuka Panel Olah Data

Gambar 4.3. Antarmuka Panel Hitung Rata-rata

##### 4.1.2.1. Proses Olah Data

Proses Olah Data terbagi menjadi beberapa subproses dalam mengolah data-data historis yang akan digunakan. Keterangan subproses tersebut sebagai berikut :

1. Proses *Refresh* dan *Filter* Tabel Data, berfungsi untuk memperbaharui isi dalam tabel data dan menampilkan kembali tabel data yang sudah diperbaharui serta memfilter data sesuai dengan bulan dan tahun tertentu.
2. Proses *Input* Data, berfungsi untuk memasukkan data baru berupa tanggal dan curah hujan harian (mm) pada tanggal tersebut.
3. Proses Cari Data, berfungsi untuk mencari suatu data pada tanggal tertentu.
4. Proses Ubah Data, berfungsi untuk mengubah data yang telah dicari atau telah dipilih dari tabel kemudian menyimpannya.
5. Proses Hapus Data, berfungsi untuk menghapus data yang telah dicari atau telah dipilih dari tabel.

Semua subproses tersebut dapat dilihat lebih jelas melalui kode program pada Lampiran 1.1, Lampiran 1.2, Lampiran 1.3., Lampiran 1.4 dan Lampiran 1.5.

#### **4.1.2.2. Proses Hitung Rata-rata**

Proses Hitung Rata-rata terbagi menjadi dua subproses, yakni Proses Menghitung Rata-rata dan Proses Simpan Hasil Rata-rata. Keterangan subproses tersebut sebagai berikut :

1. Proses Hitung Rata-rata. Dikarenakan data yang akan digunakan dalam perhitungan *Fuzzy Time Series* adalah data bulanan sedangkan data yang tersimpan sebelumnya adalah data harian, maka diperlukan adanya pengolahan data lebih lanjut, yakni merata-ratakan data harian menjadi data bulanan. Rata-rata tersebut dihitung dengan cara menjumlahkan curah hujan harian dalam sebulan kemudian dibagi dengan jumlah hari hujan pada bulan tersebut.
2. Proses Simpan Hasil Rata-rata. Setelah rata-rata perbulan dihitung, maka data tersebut perlu disimpan ke dalam basis data untuk kemudian digunakan dalam proses perhitungan berikutnya.

Semua subprocess tersebut dapat dilihat lebih jelas melalui kode program pada Lampiran 1.6, Lampiran 1.7.

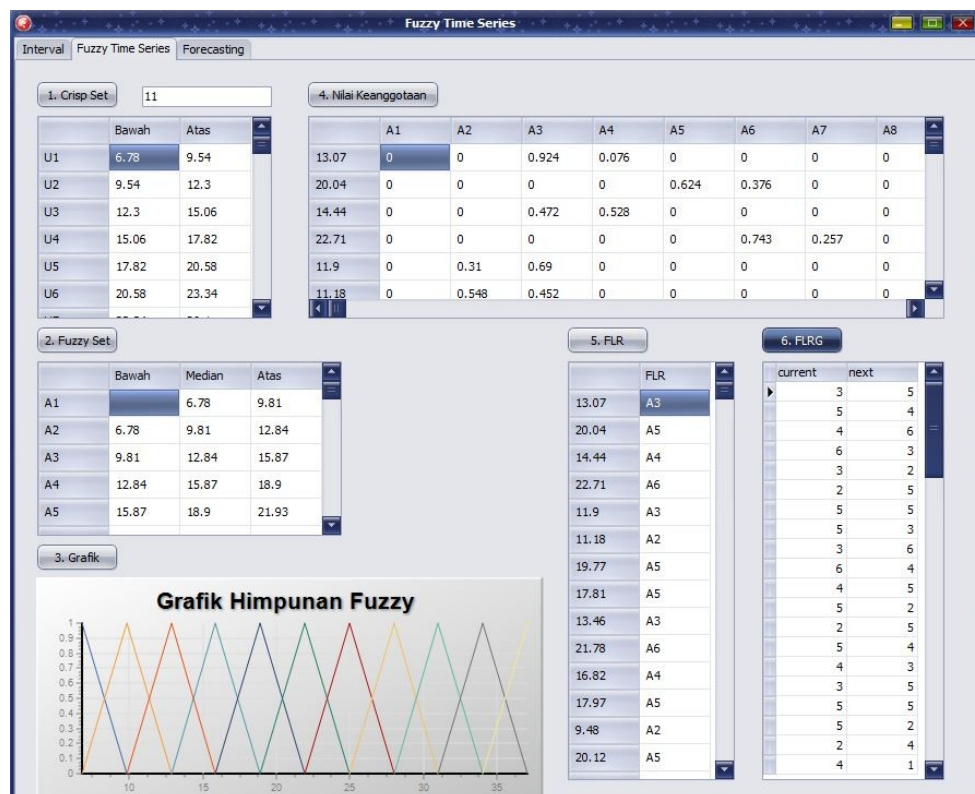
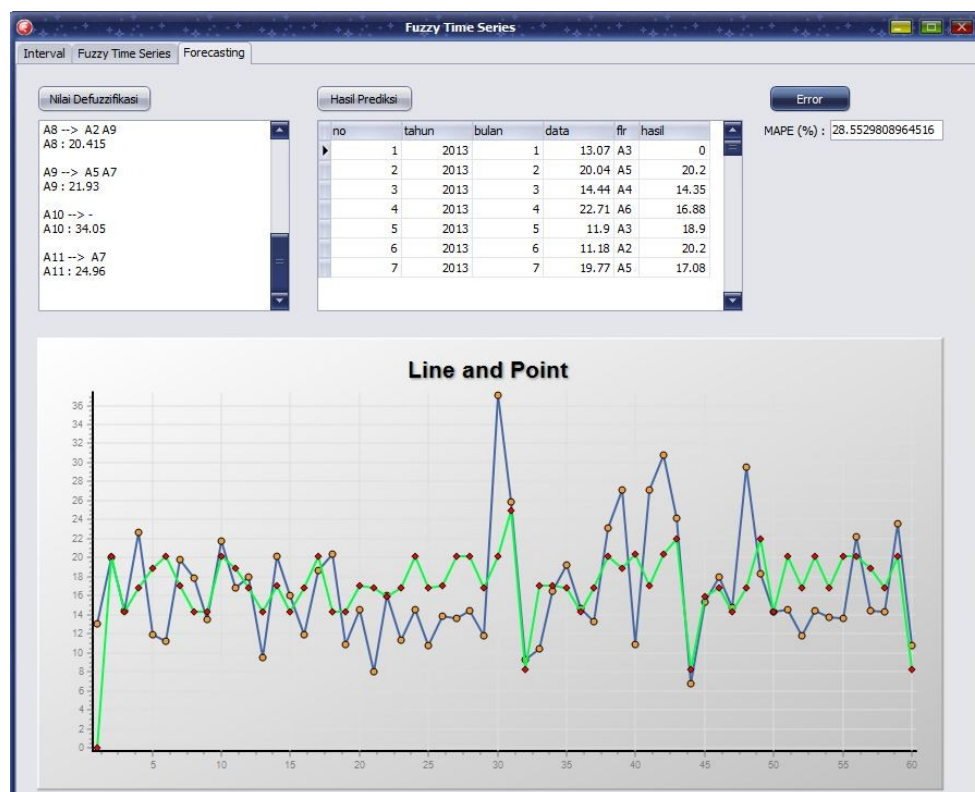
#### 4.1.3. Form *Fuzzy Time Series*

Form *Fuzzy Time Series* memiliki tiga panel utama, yaitu Panel Interval, Panel *Fuzzy Time Series*, dan Panel *Forecasting*. Panel Interval berfungsi untuk menghitung interval efektif untuk pembagian banyaknya himpunan berdasarkan keseluruhan data yang digunakan untuk perhitungan *Fuzzy Time Series* tersebut. Panel *Fuzzy Time Series* berfungsi untuk melakukan perhitungan pola data berdasarkan aturan metode *Fuzzy Time Series*. Panel *Forecasting* berfungsi untuk memproyeksikan pola data untuk melakukan proses peramalan atau *Forecasting*.

The screenshot shows the 'Fuzzy Time Series' application window with the 'Interval' tab selected. The interface is organized into three main panels:

- Periode Time Series:** Contains a 'Pilih Periode Perhitungan' section with dropdown menus for '2013' and 's.d.' followed by '2017'. Below this is a 'Perhitungan Batasan Data' section with a 'n Max Min' button and input fields for 'Jumlah Data' (60), 'Data Terbesar (mm)' (37.09), and 'Data Terkecil (mm)' (6.78).
- Array Data Time Series:** Features an 'Array' button and a list box containing 21 data points, each with a date in parentheses and a numerical value. The data points range from 'Data ke-40 = (2016/4) 10.91' to 'Data ke-60 = (2017/12) 10.8'.
- Perhitungan Interval:** Includes an 'Interval' button and several input fields with calculated values: 'Rata-rata Selsih' (6.37169491525424), 'Setengah Rata-rata' (3.18584745762712), 'Basis' (1), 'Panjang Interval' (3), and 'Jumlah Interval' (10.10333333333333).

Gambar 4.4. Antarmuka Panel Interval

Gambar 4.5. Antarmuka Panel *Fuzzy Time Series*Gambar 4.6. Antarmuka Panel *Forecasting*

Pada Form *Fuzzy Time Series* ini, digunakan beberapa variabel yang ditampilkan pada Kode Program 4.1 dan diterangkan lebih lanjut pada Tabel 4.1.

#### Kode Program 4.1. Variabel Global

```

1  var
2      n, m, pembagi                : integer;
3      slsh_fuzzy, slsh_crisp, max, min : double;
4      FLR, tahunTS, bulanTS        : array of integer;
5      dataTS, median, nilai_defuzzifikasi, dataHasil : array of Double;
6      intv_fuzzy, intv_crisp, keanggotaan : array of array of double;

```

**Tabel 4.1.** Keterangan Variabel dalam Form *Fuzzy Time Series*

Nama Variabel	Keterangan
n	Untuk menyimpan banyaknya data <i>Time Series</i>
m	Untuk menyimpan banyaknya <i>Crisp Set</i> ataupun <i>Fuzzy Set</i> yang akan dibentuk
slsh_fuzzy	Untuk menyimpan besar interval pembentuk <i>Fuzzy Set</i>
slsh_crisp	Untuk menyimpan besar interval pembentuk <i>Crisp Set</i>
max	Untuk menyimpan nilai terbesar dari data <i>Time Series</i>
min	Untuk menyimpan nilai terkecil dari data <i>Time Series</i>
FLR	Untuk menyimpan <i>Fuzzy Logic Relationship</i> yang terbentuk
tahunTS	Untuk menyimpan tahun dari data <i>Time Series</i>
bulanTS	Untuk menyimpan bulan dari data <i>Time Series</i>
dataTS	Untuk menyimpan data <i>Time Series</i>
median	Untuk menyimpan nilai tengah dari setiap <i>Fuzzy Set</i> yang terbentuk
nilai_defuzzifikasi	Untuk menyimpan hasil defuzzifikasi berdasarkan FLRG yang terbentuk
dataHasil	Untuk menyimpan data hasil peramalan
intv_fuzzy	Untuk menyimpan nilai batas bawah, nilai tengah, dan nilai batas atas dari masing-masing <i>Fuzzy Set</i>
intv_crisp	Untuk menyimpan nilai batas bawah dan nilai batas atas dari masing-masing <i>Crisp Set</i>
keanggotaan	Untuk menyimpan nilai keanggotaan masing-masing data <i>Time Series</i>

#### 4.1.3.1. Proses Perhitungan Interval

Proses Perhitungan Interval terbagi menjadi tiga subproses yaitu Proses Perhitungan Batasan Data, Proses Penyimpanan *Array*, dan Proses Perhitungan Interval itu sendiri. Ketiga subproses tersebut akan dijelaskan lebih lanjut dan kode program dari ketiga subproses tersebut dapat dilihat pada Lampiran 1.9, Lampiran 1.10, dan Lampiran 1.11.

Proses Perhitungan Batasan Data merupakan proses paling pertama dijalankan saat memulai perhitungan dengan metode *Fuzzy Time Series*. Proses ini memberi batasan periode tahun data *time series* yang akan digunakan untuk keseluruhan proses perhitungan berikutnya. Pada proses ini juga menghasilkan nilai variabel awal yakni banyaknya data *time series* ( $n$ ), nilai terbesar ( $\max$ ) dan nilai terkecil ( $\min$ ) dari data *time series* yang digunakan. Proses perhitungan tersebut dapat dilihat pada Lampiran 1.9.

Proses Penyimpanan *Array Fuzzy Time Series (FTS)* merupakan proses untuk memasukkan dan menyimpan data-data *time series* yang akan digunakan, yakni data curah hujan ( $\text{dataTS}$ ), serta tahun ( $\text{tahunTS}$ ) dan bulan ( $\text{bulanTS}$ ) dari data *time series* yang bersangkutan sebagai acuan tanggal dari data curah hujan tersebut. Proses perhitungan tersebut dapat dilihat pada Lampiran 1.10.

Proses Perhitungan Interval merupakan proses inti dari Panel Perhitungan Interval untuk memperoleh nilai interval efektif yang akan digunakan untuk membagi himpunan semesta ke dalam jumlah yang rata. Proses perhitungan tersebut dapat dilihat pada Lampiran 1.11. Langkah perhitungan dari proses perhitungan interval sebagai berikut :

- Jumlahkan rata-rata selisih antar data *time series*.
- Bagi dua nilai rata-rata selisih tersebut.
- Tentukan basis yang akan digunakan berdasarkan hasil bagi dua nilai rata-rata dengan mengacu pada Tabel 3.7.
- Tentukan panjang interval suatu himpunan dengan membagi nilai jangkauan ( $\max - \min$ ) dengan basis yang telah ditentukan.
- Tentukan jumlah interval himpunan semesta dengan membagi nilai jangkauan ( $\max - \min$ ) dengan panjang interval yang telah ditentukan.

#### 4.1.3.2. Proses Perhitungan *Fuzzy Time Series*

Proses Perhitungan *Fuzzy Time Series* berfungsi untuk menentukan pola data berdasarkan data *time series* yang digunakan. Terdapat enam subproses pada perhitungan *Fuzzy Time Series* ini, yakni Proses Perhitungan *Crisp Set*, Proses Perhitungan *Fuzzy Set*, Proses Menampilkan Grafik, Proses Perhitungan Nilai Keanggotaan, Proses Perhitungan FLR, dan Proses Perhitungan FLRG.

Proses Perhitungan *Crisp Set* atau himpunan *crisp* ini melibatkan nilai interval efektif yang diperoleh melalui perhitungan interval pada proses sebelumnya. Nilai interval pada proses perhitungan sebelumnya haruslah dibulatkan ke nilai ganjil terdekat. Proses perhitungan tersebut dapat dilihat pada Lampiran 1.12. Tahap Proses Perhitungan *Crisp Set* secara terurut sebagai berikut :

- Tentukan nilai interval efektif yang akan digunakan berdasarkan hasil perhitungan interval pada proses sebelumnya, nilai tersebut akan menjadi jumlah himpunan *crisp* yang akan terbentuk.
- Tentukan nilai selisih antar himpunan *crisp* dengan cara membagi nilai jangkauan (max-min) dengan nilai interval efektif (m).
- Tentukan nilai batas bawah dan batas atas (intv\_crisp) dari setiap himpunan *crisp* yang terbentuk.
- Tampilkan hasil pembentukan himpunan *crisp* berupa sebuah tabel dengan keterangan nilai batas atas dan batas bawah pada masing-masing himpunan *crisp*.

Proses Perhitungan *Fuzzy Set* atau himpunan *Fuzzy* berfungsi untuk menentukan dan membentuk himpunan *Fuzzy*. Proses perhitungan tersebut dapat dilihat pada Lampiran 1.13. Tahapan perhitungan proses sebagai berikut :

- Tentukan selisih antar himpunan *fuzzy* dengan membagi nilai jangkauan (max-min) dengan nilai interval efektif yang dikurangi satu (m-1).
- Tentukan nilai batas bawah, nilai tengah dan nilai batas atas (intv\_fuzzy) dari setiap himpunan *fuzzy* yang terbentuk.
- Tampilkan hasil pembentukan himpunan *fuzzy* berupa sebuah tabel dengan keterangan nilai batas bawah, nilai tengah atau median, dan nilai batas atas pada masing-masing himpunan *fuzzy*.

Proses Menampilkan Grafik merupakan sebuah proses untuk menampilkan grafik garis dari himpunan *fuzzy* yang telah terbentuk pada proses sebelumnya. Proses perhitungan tersebut dapat dilihat pada Lampiran 1.14. Adapun tahapan proses ini secara berurut adalah sebagai berikut :

- Bentuk sebuah grafik garis dengan bentuk kurva segitiga untuk masing-masing himpunan *fuzzy* yang terbentuk.
- Adapun nilai pada kaki kiri segitiga menggunakan nilai batas bawah sebuah himpunan, nilai puncak segitiga menggunakan nilai tengah atau median sebuah himpunan, dan nilai kaki kanan segitiga menggunakan nilai batas atas sebuah himpunan.
- Contoh grafik himpunan *fuzzy* yang terbentuk dapat dilihat pada Gambar 3.18.

Proses Perhitungan Nilai Keanggotaan dilakukan pada masing-masing data *time series* terhadap masing-masing himpunan *fuzzy* berdasarkan aturan *fuzzy*. Proses perhitungan tersebut dapat dilihat pada Lampiran 1.15. Tahapan proses tersebut sebagai berikut :

- Tentukan posisi data *time series* tepat diantara dua buah himpunan *fuzzy* sebagai himpunan *fuzzy* bawah dan himpunan *fuzzy* atas.
- Tentukan nilai keanggotaan pada himpunan *fuzzy* bawah dengan cara

$$\left( \frac{\text{data time series} - \text{batas bawah}}{\text{nilai tengah} - \text{batas bawah}} \right) \text{ dan pada himpunan } \textit{fuzzy} \text{ atas}$$

$$\text{dengan cara } \left( \frac{\text{batas atas} - \text{data time series}}{\text{batas atas} - \text{nilai tengah}} \right) \text{ sesuai dengan Persamaan}$$

2.3.

Proses Perhitungan FLR atau *Fuzzy Logic Relationship* merupakan penggolongan masing-masing data *time series* ke dalam sebuah nilai linguistik FLR berdasarkan nilai keanggotaan terbesar dari data *time series* tersebut pada himpunan *fuzzy*. Proses Perhitungan FLR secara lebih jelas dapat dilihat pada Lampiran 1.16.

Proses Perhitungan FLRG atau *Fuzzy Logic Relationship Group* ini adalah untuk menampilkan hubungan antara FLR  $\rightarrow$  FLR+1 pada sebuah tabel, sehingga hasil ini disimpan dalam basis data dan kemudian akan digunakan untuk



proses berikutnya. Proses Perhitungan FLRG secara lebih jelas dapat dilihat pada Lampiran 1.17.

#### **4.1.3.3. Proses *Forecasting***

Proses *Forecasting* adalah proses peramalan berdasarkan hasil perhitungan pola data pada Proses *Fuzzy Time Series*. Proses ini terdiri dari 3 subproses, yaitu Proses Perhitungan Defuzzifikasi, Proses *Forecasting*, dan Proses Perhitungan Tingkat Kesalahan. Selain tiga subproses tersebut, terdapat juga bagian untuk menampilkan grafik garis perbandingan antara data aktual dengan data hasil peramalan dengan *Fuzzy Time Series*.

Proses Perhitungan Defuzzifikasi adalah lanjutan dari Proses Perhitungan FLRG sebelumnya, yakni penggolongan grup berdasarkan *nilai\_current* yang sama dari suatu FLR terhadap nilai FLR+1. Kemudian hasil penggolongan grup tersebut akan dihitung nilai defuzzifikasinya dengan cara menambahkan nilai tengah atau median dari setiap himpunan *fuzzy* pada suatu grup kemudian dibagi dengan jumlah himpunan *fuzzy* pada grup tersebut. Proses perhitungan tersebut dapat dilihat pada Lampiran 1.18.

Proses *Forecasting* adalah proses peramalan berdasarkan hasil perhitungan *Fuzzy Time Series* sebelumnya. Proses peramalan dilakukan dengan cara melihat nilai FLR dari setiap data *time series* dan kemudian mengubahnya menjadi nilai defuzzifikasi sesuai dengan FLR yang berkaitan. Pada proses ini jugalah dilakukan penampilan grafik garis perbandingan antara data aktual dengan data hasil peramalan. Proses *Forecasting* secara lebih jelas dapat dilihat pada Lampiran 1.19.

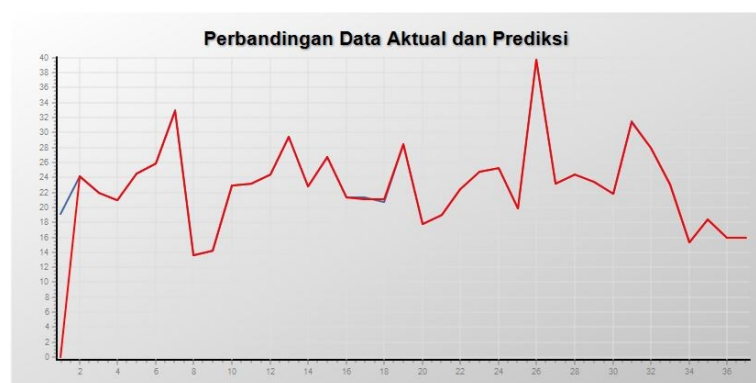
Proses Perhitungan Tingkat Kesalahan merupakan proses terakhir dari serangkaian proses yang berjalan pada aplikasi ini. Proses ini menghitung tingkat kesalahan dengan cara menjumlahkan hasil selisih absolut antara data aktual dengan data hasil prediksi dibagi data aktual, sesuai dengan Persamaan 2.7. Proses *Forecasting* secara lebih jelas dapat dilihat pada Lampiran 1.20.

## **4.2. Hasil Uji**

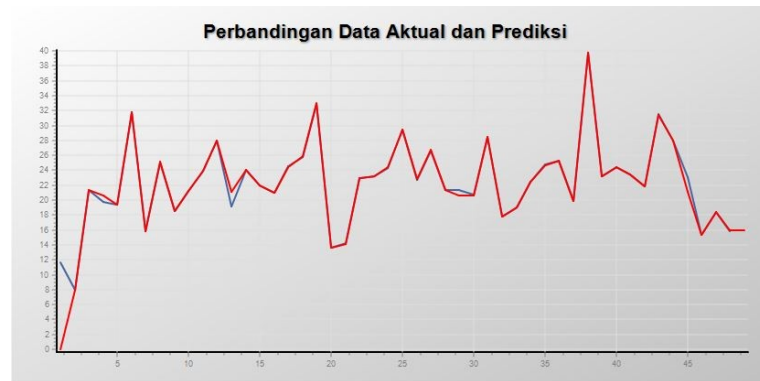
Dalam pengujian oleh program digunakan data *time series* dalam 15 periode waktu yang berbeda, yaitu dari 1 tahun, 2 tahun, 3 tahun, hingga 15 tahun.

Masing-masing data yang digunakan terdiri dari data aktual 12 bulan dari Januari hingga Desember setiap tahun pada setiap periode, sehingga terdapat 12 data pada periode 1 tahun, 24 data pada periode 2 tahun, 36 data pada periode 3 tahun, hingga 180 data pada periode 15 tahun. Pada pengujian masing-masing periode digunakan berbagai interval hingga ditemukan nilai interval optimal yakni interval 401 yang menghasilkan nilai *MAPE* terbaik dari semua hasil pengujian. Pengujian-pengujian dengan menggunakan jumlah interval berbeda akan menghasilkan nilai *MAPE* yang berbeda yang pula, namun untuk mengoptimalkan hasil perbandingan antardata, maka digunakanlah jumlah interval yang sama yakni 401.

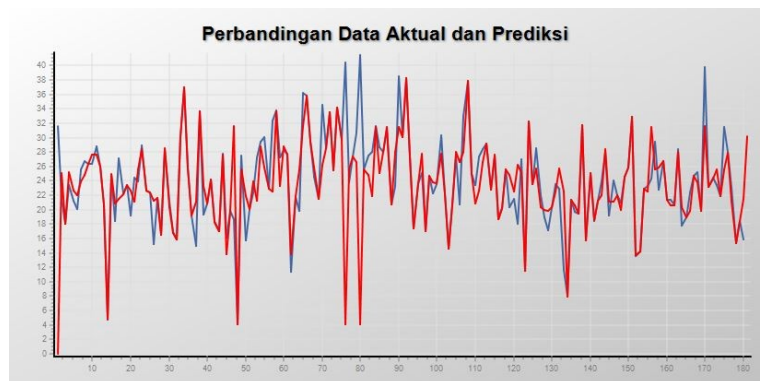
Hasil pengujian terbaik terdapat pada pengujian dengan jumlah tahun sebanyak 3 tahun dengan periode pengujian tahun 2015 – 2017 dengan menggunakan interval optimal 401. Nilai *MAPE* pada pengujian tersebut adalah sebesar 0.1509%, sehingga diperoleh nilai akurasi atau ketepatan sebesar 99.8491%. Sedangkan pengujian dengan hasil *MAPE* terbaik adalah 0.6460% yaitu pada pengujian 4 tahun periode 2014 – 2017 dengan nilai akurasi atau ketepatan sebesar 99.354%. Hasil pengujian dengan nilai *MAPE* terbesar adalah 7.6522% yaitu pada pengujian 15 tahun periode 2003 – 2017 dengan nilai akurasi atau ketepatan sebesar 92.3478%. Beberapa gambar grafik perbandingan nilai data aktual dengan hasil prediksi dapat dilihat pada Gambar 4.7, Gambar 4.8, dan Gambar 4.9.



Gambar 4.7. Grafik Hasil Curah Hujan 2015 – 2017



Gambar 4.8. Grafik Hasil Curah Hujan 2014 – 2017



Gambar 4.9. Grafik Hasil Curah Hujan 2003 – 2017

Dari data hasil prediksi *Fuzzy Time Series* pada berbagai periode berbeda diperoleh nilai *MAPE* yang juga berbeda-beda. Hal ini menunjukkan bahwa penggunaan periode data aktual sebagai bahan perhitungan pola data ternyata sangat berpengaruh terhadap pola data yang dihasilkan dari perhitungan. Tidak hanya data aktual, tetapi jumlah interval yang digunakan dalam membagi data untuk proses perhitungan pola data juga sangat berpengaruh terhadap hasil perhitungan prediksi akhir. Nilai *MAPE* dari pengujian-pengujian yang telah dilakukan dapat dirangkum pada Tabel 4.2.

**Tabel 4.2.** Nilai *Mean Average Percentage Error*

Jumlah Tahun	Periode	Nilai MAPE (%)
1	2017	7.1007
2	2016-2017	3.6946
3	2015-2017	0.1509
4	2014-2017	0.6460
5	2013-2017	3.1105
6	2012-2017	2.7348
7	2011-2017	4.3438
8	2010-2017	3.8931
9	2009-2017	5.5051
10	2008-2017	5.6795
11	2007-2017	5.9691
12	2006-2017	7.3314
13	2005-2017	7.5127
14	2004-2017	7.6237
15	2003-2017	7.6522

#### 4.3. Analisis Hasil

Dari hasil pengujian yang telah dilakukan terhadap program dengan menggunakan berbagai periode waktu data aktual, dapat dilihat bahwa jumlah data mempengaruhi tingkat akurasi dari hasil perhitungan dan peramalan yang dilakukan berdasarkan data tersebut. Penggunaan jumlah interval yang berbeda juga menghasilkan tingkat akurasi yang berbeda. Maka berdasarkan hasil tersebut, analisis terhadap hasil pengujian adalah sebagai berikut :

1. Banyaknya jumlah data yang digunakan memberikan pengaruh besar pada hasil peramalan. Data tersebut digunakan oleh sistem untuk menghitung, menganalisis, dan menentukan pola data yang kemudian akan digunakan untuk memproyeksikan peramalan.
2. Jumlah interval yang digunakan dalam membagi data untuk perhitungan juga memberikan pengaruh pada hasil peramalan. Berdasarkan hasil pengujian pada sistem, ditemukan bahwa nilai interval terbaik yang dapat digunakan untuk membagi data sehingga menghasilkan nilai *MAPE* yang baik adalah interval 401.
3. Pada peramalan dengan menggunakan data curah hujan, nilai *MAPE* terbaik yang dihasilkan adalah 0.151% yaitu pada periode 2015 – 2017 dengan menggunakan jumlah interval 401. Dengan demikian tingkat akurasi pada pengujian peramalan tersebut adalah 99,849%.
4. Metode *Fuzzy Time Series* memiliki nilai akurasi lebih tinggi pada saat digunakan untuk melakukan peramalan pada data yang memiliki pola musiman daripada data yang memiliki pola abstrak.

## **BAB V**

### **PENUTUP**

#### **5.1. Kesimpulan**

Setelah dilakukan analisis dan pengujian terhadap aplikasi Prediksi Besaran Curah Hujan menggunakan *Fuzzy Time Series*, dapat disimpulkan bahwa :

1. Aplikasi yang dibangun dapat melakukan proses olah data curah harian menjadi data curah hujan bulanan serta melakukan proses perhitungan prediksi curah hujan dengan metode *Fuzzy Time Series*.
2. Penggunaan metode *Fuzzy Time Series* dalam aplikasi memungkinkan sistem untuk menentukan pola data *time series* yang digunakan kemudian memproyeksikan pola data tersebut untuk memperoleh hasil prediksi.
3. Hasil perhitungan prediksi *Fuzzy Time Series* sangat dipengaruhi oleh jumlah data aktual yang digunakan sebagai acuan perhitungan pola data serta jumlah interval pada saat perhitungan pola data.

#### **5.2. Saran**

Adapun beberapa hal yang perlu ditambahkan dalam pengembangan aplikasi Prediksi Besaran Curah Hujan menggunakan *Fuzzy Time Series* adalah sebagai berikut :

1. Aplikasi Prediksi Besaran Curah Hujan menggunakan *Fuzzy Time Series* ini dapat ditambahkan dan dikolaborasikan dengan metode perhitungan lainnya agar dapat memperhitungkan faktor-faktor lain yang mempengaruhi curah hujan sehingga dapat memperoleh pola data yang lebih akurat.
2. Aplikasi Prediksi Besaran Curah Hujan menggunakan *Fuzzy Time Series* dapat dikembangkan lagi secara *online* sehingga dapat

melakukan proses integrasi data secara langsung dengan *website* yang menampilkan data curah hujan secara *online*.