

**PENERAPAN LOGIKA FUZZY TAHANI PADA PENERIMAAN
BEASISWA PENINGKATAN PRESTASI AKADEMIK
(STUDI KASUS : STMIK BUMIGORA MATARAM)**

SKRIPSI



Oleh :

**M RODI TAUFIK AKBAR
1310530126**

**PROGRAM STUDI S1 TEKNIK INFORMATIKA
SEKOLAH TINGGI MANAJEMEN INFORMATIKA DAN KOMPUTER
(STMIK) BUMIGORA
MATARAM
2018**

**PENERAPAN LOGIKA FUZZY TAHANI PADA PENERIMAAN
BEASISWA PENINGKATAN PRESTASI AKADEMIK
(STUDI KASUS : STMIK BUMIGORA MATARAM)**

SKRIPSI



Diajukan Sebagai Salah Satu Syarat untuk Memenuhi Kebutuhan Studi
Jenjang Strata Satu (S1) Program Studi Teknik Informatika Pada
Sekolah Tinggi Manajemen Informatika dan Komputer
(STMIK) Bumigora Mataram

Oleh :

**M RODI TAUFIK AKBAR
1310530126**

**PROGRAM STUDI TEKNIK INFORMATIKA
SEKOLAH TINGGI MANAJEMEN INFORMATIKA DAN KOMPUTER
(STMIK) BUMIGORA
MATARAM
2018**

**PENERAPAN LOGIKA FUZZY TAHANI PADA PENERIMAAN
BEASISWA PENINGKATAN PRESTASI AKADEMIK
(STUDI KASUS : STMIK BUMIGORA MATARAM)**

SKRIPSI

Diajukan Sebagai Salah Satu Syarat untuk Memenuhi Kebutuhan Studi
Jenjang Strata Satu (S1) Program Studi Teknik Informatika
pada Sekolah Tinggi Manajemen Informatika dan Komputer
(STMIK) Bumigora Mataram

Oleh :

**M RODI TAUFIK AKBAR
1310530126**

Ketua,


Heroe Santoso, M.Kom
NIK. 96.6.64

**SEKOLAH TINGGI MANAJEMEN INFORMATIKA DAN KOMPUTER
(STMIK) BUMIGORA MATARAM
PROGRAM STUDI TEKNIK INFORMATIKA**

SKRIPSI

JUDUL : Penerapan Logika Fuzzy Tahani Pada Penerimaan
Basiswa Peningkatan Prestasi Akademik
(Studi kasus : STMIK Bumigora Mataram)

NAMA : M Rodi Taufik Akbar

NIM : 1310530126

NPM : 13.8.349.74.75.0.5.0089

PROGRAM STUDI : Teknik Informatika

JENJANG : Strata Satu (S1)

DIUJIKAN : 25 Agustus 2018

Menyetujui
Muhammad Yunus, M.Kom
Pembimbing



Tanggal Menyetujui : 31/8/2018

Telah diterima dan disetujui sebagai salah satu syarat untuk memperoleh
Gelar Akademik Sarjana Komputer (S.Kom)

Mengetahui
Ni Gusti Ayu Dasriani, M.Kom
Ketua Program Studi S1 Teknik Informatika



Tanggal Mengetahui : 31 Agustus 2018

**PENERAPAN LOGIKA FUZZY TAHANI PADA PENERIMAAN
BEASISWA PENINGKATAN PRESTASI AKADEMIK
(STUDI KASUS : STMIK BUMIGORA MATARAM)**

LEMBAR PENGESAHAN PENGUJI


Diajukan Sebagai Salah Satu Syarat untuk Memenuhi Kebutuhan Studi
Jenjang Strata Satu (S1) Program Studi Teknik Informatika
pada Sekolah Tinggi Manajemen Informatika dan Komputer
(STMIK) Bumigora Mataram

Oleh:


**M RODI TAUFIK AKBAR
1310530126**

Disetujui oleh Penguji :

1. Ahmad Ashril Rizal, M.Cs.
NIK. 11.6.160


27/6 - 2016

2. Syahroni Hidayat, ST., M.Eng.
NIK. 13.6.177


31/6 - 2016

KATA PENGANTAR

Puji syukur penulis panjatkan Kehadiran Allah S.W.T yang maha pemurah lagi Maha Penyayang. Atas rahmat dan nikmat yang telah dianugerahkannya sehingga penulis dapat menyelesaikan skripsi yang berjudul "**PENERAPAN LOGIKA FUZZY TAHANI PADA PENERIMAAN BEASISWA PENINGKATAN PRESTASI AKADEMIK (STUDI KASUS : STMIK BUMIGORA MATARAM)**".

Dengan selesainya skripsi ini, maka penulis ingin mengucapkan terima kasih kepada pihak-pihak yang telah banyak membantu dalam menyelesaikan skripsi ini. Dalam kesempatan ini penulis menyampaikan ucapan terimakasih kepada:

1. Bapak Heroe Santoso, M.Kom., selaku Ketua Sekolah Tinggi Manajemen Informatika dan Komputer (STMIK) Bumigora Mataram.
2. Ibu Ni Gusti Ayu Dasriani, M.Kom., selaku Ketua Program Studi S1 Teknik Informatika.
3. Bapak Muhammad Yunus, M.Kom selaku pembimbing yang selalu membantu saya dalam menyelesaikan skripsi ini
4. Bapak/Ibu Dosen yang telah memberikan ilmu selama dalam perkuliahan.
5. Untuk orang tua saya Bapak dan Ibu tercinta yang telah memberikan dukungan moril dan dukungan materi serta

mendoakan, memberikan semangat dalam penyusunan skripsi ini.

6. Teman-teman Mahasiswa Angkatan 2013 STMIK Bumigora Mataram, dan semua teman yang ada di kos-kosan maupun teman sedesa yang tidak bisa disebutkan satu persatu, atas semua bantuannya baik secara Penulis menyadari dengan sepenuhnya bahwa dalam penulisan skripsi ini masih jauh dari kesempurnaan, maka penulis senantiasa mengharapkan teguran, kritik serta saran yang sifatnya membangun untuk dapat lebih sempurnanya tugas akhir atau sejenisnya pada masa-masa yang akan datang.

Akhirnya penulis berharap semoga tulisan ini bermanfaat bagi semua pihak yang membacanya di kemudian hari.

Mataram, Agustus 2018

Penulis



SEKOLAH TINGGI MANAJEMEN INFORMATIKA DAN KOMPUTER
(STMIK) BUMIGORA MATARAM

LEMBAR PERNYATAAN KEASLIAN

Saya yang bertandatangan di bawah ini:

Nama : M Rodi Taufik Akbar
NIM : 1310530126
Program studi : S1 Teknik Informatika
Kompetensi : Rekayasa Perangkat Lunak

Menyatakan bahwa skripsi yang berjudul:

**PENERAPAN LOGIKA FUZZY TAHANI PADA PENERIMAAN BEASISWA
PENINGKATAN PRESTASI AKADEMIK (STUDI KASUS : STMIK
BUMIGORA MATARAM)**

Benar-benar merupakan hasil karya pribadi dan seluruh sumber yang dikutip maupun dirujuk telah saya nyatakan dengan benar dan apabila di kemudian hari terdapat penyimpangan dan ketidakbenaran dalam pernyataan ini, maka saya bersedia menerima sanksi akademik sesuai dengan aturan yang berlaku.

Mataram, 30 Agustus 2018

M Rodi Taufik Akbar

NIM.1310530126

IZIN PENGGUNAAN

Skripsi ini merupakan syarat kelulusan pada Program Studi S1 Teknik Informatika STMIK Bumigora Mataram, dengan ini penulis setuju jika skripsi ini digandakan (diduplikasi) baik sebagian maupun seluruhnya, ataupun dikembangkan untuk kepentingan akademis yang disetujui oleh pembimbing penulis, Pembantu Ketua I atau Ketua STMIK Bumigora.

Untuk dimaklumi, bahwa menduplikasi, mempublikasikan atau menggunakan skripsi ini, maupun bagian-bagiannya dengan tujuan komersial / keuntungan finansial, tidak diizinkan tanpa adanya izin tertulis dari STMIK Bumigora. Jika hal ini dilanggar maka STMIK Bumigora akan memberikan sanksi sesuai dengan hukum yang berlaku

Penghargaan akademis terkait isi dari skripsi ini adalah pada penulis dan STMIK Bumigora.

Permintaan izin untuk menduplikasi atau menggunakan materi dari skripsi ini baik sebagian maupun seluruhnya harus ditujukan pada:

Pembantu Ketua I

Ketua Program Studi S1 Teknik Informatika

STMIK Bumigora Mataram

ABSTRAK

Sistem database relasional yang ada sampai sekarang ini hanya mampu menangani data yang bersifat pasti (crisp), deterministik dan presisi. Padahal dalam kondisi nyata, sering sekali dibutuhkan data yang samar untuk proses pengambilan keputusan. Untuk pengambilan keputusan yang melibatkan variabel samar berdasarkan data crisp yang ada di dalam database, dapat direalisasikan dengan mengembangkan suatu aplikasi untuk menangani query yang diberikan kepada sistem database serta dengan melakukan fuzzifikasi pada data.

Di setiap lembaga pendidikan khususnya Perguruan Tinggi banyak sekali beasiswa yang ditawarkan kepada mahasiswa. Untuk mendapatkan beasiswa tersebut maka harus sesuai dengan aturan yang telah ditetapkan. Oleh sebab itu tidak semua mahasiswa yang mengajukan permohonan untuk menerima beasiswa dapat diberikan. Karena jumlah mahasiswa yang mengajukan permohonan yang banyak dan kriteria penilaian yang banyak pula, maka perlu dibangun suatu sistem yang dapat membantu memberikan rekomendasi penerima beasiswa.

Selama ini sistem yang digunakan untuk proses seleksi beasiswa PPA di Sekolah Tinggi Manajemen Informatika dan Komputer (STMIK) Bumigora Mataram masih dikerjakan secara semi manual menggunakan Microsoft Excel. Oleh karena itu perlu dibuatkan sistem yang khusus untuk merekomendasikan beasiswa PPA dengan menggunakan metode penerapan logika Fuzzy Tahani.

Hasil dari penelitian berupa sistem mampu melakukan seleksi beasiswa PPA dengan tingkat akurasi 73,3.

Kata kunci : *Fuzzy, Tahani, Beasiswa, PPA, STMIK Bumigora*

DAFTAR ISI

Halaman:

HALAMAN JUDUL	
HALAMAN PENGESAHAN	
KATA PENGANTAR.....	i
LEMBAR PERNYATAAN KEASLIAN.....	iii
IZIN PENGGUNA	iv
ABSTRAK	v
DAFTAR ISI	vi
DAFTAR GAMBAR	ix
DAFTAR TABEL	x
DAFTAR LAMPIRAN	xi
 BAB I PENDAHULUAN.....	 1
1.1 latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Batasan Masalah.....	2
1.4 Tujuan dan Manfaat Penulisan	3
1.4.1 Tujuan Penulisan	3
1.4.2 Manfaat Penulisan	3
1.5 Metodologi Penelitian	3
1.5.1 Metode Pengumpulan Data.....	3
1.6 Metode Pengembangan.....	4
1.7 Sistematika Penulisan.....	5
 BAB II LANDASAN TEORI	 7
2.1 Logika Fuzzy	7
2.1.1 Pengertian Fuzzy	7
2.1.2 Konsep Kekaburan (<i>Fuzziness</i>) dan Sejarah Logika Fuzzy	9
2.1.3 Himpunan Keanggotaan.....	10
2.1.4 Fungsi Keanggotaan.....	13
2.1.5 Operator Dasar Zadeh untuk Operasi Himpunan Fuzzy.....	14
2.1.6 Perbedaan Himpunan Fuzzy dengan Probabilistik.....	15
2.1.7 Keunggulan Logika Fuzzy.....	17
2.2 Pangkalan Data.....	18
2.2.1 Pengertian Pangkalan Data.....	18
2.2.2 Sejarah Pangkalan Data	19
2.3 Fuzzy Tahani.....	20
2.4 Basis Data.....	25
2.4.1 Konsep DBMS(Database Management System).....	26
2.5 <i>Fuzzy Database</i>	27
2.5 <i>Beasiswa</i>	29
2.5.1 Pengertian Beasiswa	29
2.5.2 Jenis-jenis Beasiswa.....	31
2.7 <i>Waterfall</i>	33
2.8 Perangkat Analisa Sistem.....	35
2.8.1 Data Flow Diagram (DFD).....	35
2.8.2 Entity Relational Diagram (ERD).....	36
2.8.3 Normalisasi.....	39

2.8.4	Flowchart.....	40
2.8.5	Kamus Data.....	44
2.8	PHP.....	44
2.8.1	Sejarah PHP.....	44
2.8.2	Sintaksis Dasar.....	46
2.8.3	Membuat Program.....	48
2.8.4	Kelebihan PHP dari Pemrograman Lain.....	49
2.9	CodeIgniter.....	50
2.9.1	Framework.....	51
2.9.2	Desain Patern MVC (Model, View, Controller).....	52
2.10	XAMPP.....	57
2.10.1	Sejarah dan Pengembang.....	54
2.10.2	Asal kata dari XAMPP.....	58
2.10.3	Bagian-Bagian penting xampp.....	59
2.10.4	Komponen.....	60
2.11	MySQL.....	61
2.11.1	Sistem manajemen basis data relasional.....	62
2.11.2	Sejarah MySQL.....	63
2.11.3	Keistimewaan MySQL.....	65
2.11.4	Kelebihan dan Kekurangan MySQL dibanding RDMS Lain.....	67
2.11.5	Kemampuan MySQL.....	69
2.11.6	Bahasa Pemrograman.....	70
2.11.7	Penggunaan.....	70
2.11.8	Administrasi.....	71
2.11.9	Akuisisi oleh Sun dan Oracle.....	71
2.11.10	Komunitas MySQL Indonesia.....	72
BAB III ANALISA DAN PERANCANGAN SISTEM		73
3.1	Metodelogi.....	73
3.2	Tinjauan Beasiswa PPA	74
3.2.1	Alur Penerimaan Beasiswa PPA	74
3.2.2	Tujuan Beasiswa PPA.....	75
3.2.3	Manfaat Beasiswa PPA.....	75
3.3	Pembentukan Variabel dan Himpunan Fuzzy	75
3.3.1	Data Masukan	75
3.3.2	Data keluaran.....	76
3.3.3	Kebutuhan Perangkat Lunak	76
3.3.4	Perangkat Keras	77
3.3.5	Sumber Daya Manusia.....	78
3.4	Desain Sistem.....	78
3.4.1	Struktur Database.....	78
3.4.2	Relasi Database	82
3.4.3	Desain Diagram Sistem	83
3.5	Rules Fuzzy	84
3.6	Desain Interface	84
3.7	Arsitektur Sistem Fuzzy Tahani	88
3.8	Perhitungan Fuzzy Tahani.....	89
3.8.1	Menghitung Fungsi Keanggotaan	89
BAB IV HASIL DAN PEMBAHASAN		94

4.1	Persiapan Implentasi.....	94
4.1.1	Instalasi Web Server dan Database Server.....	94
4.1.2	Instalasi Teks Editor.....	94
4.1.3	Instalasi Perangkat Pendukung.....	94
4.2	Tahapan Pembangunan.....	95
4.2.1	Pembuatan Database.....	95
4.2.2	Pembangunan Program.....	95
4.3	Penjelasan Flowchart Program dan Program.....	95
4.3.1	Halaman Login.....	96
4.3.2	Halaman Beranda.....	97
4.3.3	Halaman Tampil Data Mahasiswa.....	98
4.3.4	Halaman Input Data Mahasiswa.....	99
4.3.5	Halaman Pengaturan Kriteria.....	100
4.3.6	Halaman Perhitungan Proses Fuzzyfikasi.....	101
4.3.7	Halaman Tampil Penerima Beasiswa.....	102
4.4	Tahap Pengujian.....	103
BAB V	KESIMPULAN DAN SARAN.....	108
5.1	Kesimpulan.....	108
5.2	Saran.....	108

DAFTAR REFERENSI
LAMPIRAN

DAFTAR GAMBAR

	Halaman
Gambar 2.1 Himpunan fuzzy pada variabel mahasiswa	12
Gambar 2.2 Representasi linear naik.....	13
Gambar 2.3 Representasi linear turun.....	14
Gambar 2.4 Pengolahan data menjadi informasi.....	18
Gambar 2.5 Fungsi keanggotaan untuk variabel usia	23
Gambar 3.1 Alur penerimaan beasiswa	74
Gambar 3.2 Relasi database	83
Gambar 3.3 Desain sistem	78
Gambar 3.4 Desain login	85
Gambar 3.5 Desain beranda.....	85
Gambar 3.6 Desain input data mahasiswa	86
Gambar 3.7 Desain tampil data mahasiswa.....	86
Gambar 3.8 Desain pengaturan kriteria	87
Gambar 3.9 Desain proses hitung fuzzifikasi	87
Gambar 3.10 Desain penerima beasiswa	88
Gambar 3.11 Arsitektur sistem fuzzy tahani.....	88
Gambar 3.12 Grafik IPK	89
Gambar 3.13 Grafik penghasilan orang tua	91
Gambar 3.14 Grafik jumlah tanggungan orang tua	92
Gambar 3.15 Grafik Jumlah sertifikat.....	93
Gambar 4.1 Flowchart login.....	96
Gambar 4.2 Tampilan halaman login	96
Gambar 4.3 Flowchart halaman beranda.....	97
Gambar 4.4 Tampilan halaman beranda.....	97
Gambar 4.5 Flowchart tampil data mahasiswa	98
Gambar 4.6 Tampilan halaman data mahasiswa	98
Gambar 4.7 Flowchart input data mahasiswa	99
Gambar 4.8 Halaman input data mahasiswa	99
Gambar 4.9 Flowchart pengaturan kriteria.....	100
Gambar 4.10 Halaman pengaturan kriteria	100
Gambar 4.11 Flowchart fuzzifikasi	101
Gambar 4.12 Proses hitung fuzzifikasi.....	101
Gambar 4.13 Flowchart tampil penerima beasiswa	102
Gambar 4.14 Halaman tampil perhitungan penerima beasiswa	102

DAFTAR TABEL

	Halaman
Tabel 2.1 Contoh data karyawan	23
Tabel 2.2 Tabel karyawan berdasarkan umur	24
Tabel 2.3 Data karyawan yang masih muda dan memiliki gaji tinggi	25
Tabel 2.4 Simbol data flow diagram	36
Tabel 2.5 Simbol flowchart	41
Tabel 2.6 Versi CodeIgniter	54
Tabel 3.1 Tabel inputan	75
Tabel 3.2 Tabel mahasiswa	79
Tabel 3.3 Tabel PPA	79
Tabel 3.4 Tabel bobotipk	80
Tabel 3.5 Tabel bobotPekerjaan	80
Tabel 3.6 Tabel bobotpenghasilan	80
Tabel 3.7 Tabel bobottanggungan	81
Tabel 3.8 Tabel bobotsertifikat	81
Tabel 3.9 Tabel user	81
Tabel 3.10 Tabel konfigurasi	82
Tabel 3.11 Rule Fuzzy	84
Tabel 4.1 Data pendaftar beasiswa	103
Tabel 4.2 Data Pendaftar penerima beasiswa	106

DAFTAR LAMPIRAN

	Halaman
Lampiran A. Source Code	
Source code login.....	A.1
Source code admin.....	A.1
Source code fuzzifikasi.....	A.4
Source code models Konfigurasi.....	A.6
Source code models mahasiswa.....	A.6

BAB I

PENDAHULUAN

1.1 Latar Belakang

Pemerintah melalui direktorat Jenderal Pendidikan Tinggi Kementerian Pendidikan Nasional berupaya mengalokasikan dana untuk memberikan bantuan biaya pendidikan kepada mahasiswa yang orang tuanya tidak mampu untuk membiayai pendidikannya, dan memberikan beasiswa kepada mahasiswa yang mempunyai prestasi tinggi, baik di bidang akademik dan atau non akademik.

Di setiap lembaga pendidikan khususnya Perguruan Tinggi banyak sekali beasiswa yang ditawarkan kepada mahasiswa. Untuk mendapatkan beasiswa tersebut maka harus sesuai dengan aturan yang telah ditetapkan. Oleh sebab itu tidak semua mahasiswa yang mengajukan permohonan untuk menerima beasiswa dapat diberikan. Karena jumlah mahasiswa yang mengajukan permohonan yang banyak dan kriteria penilaian yang banyak pula, maka perlu dibangun suatu sistem yang dapat membantu memberikan rekomendasi penerima beasiswa.

Sekolah Tinggi Manajemen Informatika dan Komputer (STMIK) Bumigora Mataram merupakan salah satu sekolah komputer di Nusa Tenggara Barat yang berusaha untuk menghasilkan mahasiswa yang berpotensi dan ahli pada bidang informatika (Buku pedoman STMIK Bumigora Mataram, 2013).

Selama ini sistem yang digunakan untuk proses seleksi beasiswa PPA di Sekolah Tinggi Manajemen Informatika dan Komputer(STMIK) Bumigora Mataram masih dikerjakan secara semi manual menggunakan Microsoft excel. Oleh karena itu perlu dibuatkan sistem yang khusus untuk perekomendasi beasiswa PPA dengan menggunakan metode penerapan logika Fuzzy Tahani.

1.2 Rumusan Masalah

Adapun rumusan masalah dari penelitian ini adalah menentukan penerima beasiswa Peningkatan Prestasi Akademik(PPA) dilingkungan kampus STMIK Bumigora Mataram

1.3 Batasan Masalah

Berdasarkan rumusan masalah di atas penulis memberikan batasan masalah sebagai berikut:

- a. Penelitian dilakukan di STMIK Bumigora Mataram pada bidang kemahasiswaan
- b. Kasus yang diteliti adalah untuk penerimaan Beasiswa Peningkatan Prestasi Akademik(PPA)
- c. Input atau Variabel yang digunakan adalah IPK, Penghasilan Orang Tua, Jumlah Tanggungan, Pekerjaan dan Jumlah Sertifikat. Dimana setiap variable terdiri dari masing-masing tiga himpunan fuzzy yaitu rendah, sedang dan tinggi.
- d. Output sistem berupa hasil seleksi beasiswa yang disertai derajat atau presentase ketegasan.

1.4 Tujuan dan Manfaat Penelitian

1.4.1 Tujuan

Adapun tujuan membangun sistem Fuzzy Tahani ini adalah untuk menyeleksi penerimaan beasiswa PPA.

1.4.2 Manfaat

Adapun manfaat yang dapat diperoleh dari sistem ini adalah:

- a. Memudahkan bagi kemahasiswaan dalam seleksi beasiswa
- b. Membantu hasil yang lebih objektif karena disertai dengan bobot atau derajat ketegasan.

1.5 Metode Penelitian

1.5.1 Metode Pengumpulan Data

Metode Pengumpulan Data adalah suatu metode dimana peneliti mengumpulkan data-data dan menganalisis fakta-fakta mengenai suatu masalah pada penelitian. Adapun beberapa metode yang digunakan dalam penelitian pengumpulan data yang digunakan dalam penyusunan tugas akhir ini yaitu :

- a. Studi Literatur

Mencari literatur berupa paper, text book dan tutorial – tutorial yang berkaitan dengan permasalahan pada penulisan skripsi sehingga mudah dalam penyelesaiannya.

b. Wawancara

Dalam hal ini penulis melakukan Tanya jawab langsung dengan bagian kemahasiswaan yang menangani mengenai beasiswa.

1.6 Metode Pengembangan

Dalam penyusunan skripsi ini penulis menggunakan metode waterfall (pressman 2002) dengan tahapan-tahapan sebagai berikut :

- a. Analisis kebutuhan perangkat lunak. Proses pengumpulan kebutuhan diintensifkan dan difokuskan, khususnya pada perangkat lunak, untuk memahami sifat program yang dibangun, pereayasa perangkat lunak (analisis) harus memahami domain informasi, tingkah laku, unjuk karya, dan antar muka (*interface*) yang diperlukan.
- b. Design. Desain perangkat lunak sebenarnya adalah proses multi langkah yang berfokus pada empat atribut sebuah program yang berbeda; struktur data, arsitektur perangkat lunak, representasi perangkat lunak yang dapat diperkirakan demi kualitas sebelum dimulai pemunculan kode.
- c. Generasi kode. Desain harus diterjemahkan kedalam bentuk mesin yang bisa dibaca, langkah pembuatan kode melakukan tugas ini jika desain dilakukan dengan cara yang lengkap, pembuatan kode dapat dilakukan secara mekanis.
- d. Test atau Pengujian. Sekali kode dibuat pengujian program dimulai. Proses pengujian berfokus pada logika internal perangkat lunak, memastikan bahwa semua pernyataan sudah diuji, dan pada eksternal

fungsional yaitu mengarahkan pengujian untuk menemukan kesalahan-kesalahan dan memastikan bahwa input dibatasi akan memberikan hasil aktual yang sesuai dengan hasil dibutuhkan.

1.7 Sistematika Penulisan

Adapun sistematika penulisan skripsi ini, penulis akan melakukan pembahasan dalam beberapa bab antara lain:

BAB I PENDAHULUAN

Bab ini diuraikan tentang latar belakang masalah, perumusan masalah, batasan masalah, tujuan dan manfaat penelitian, metodologi penelitian, dan sistematika penulisan.

BAB II LANDASAN TEORI

Bab ini berisi tentang teori-teori yang digunakan dalam penelitian ini. Pengertian algoritma logika fuzzy tahani dan aplikasi pendukung web serta alat perancangan yang akan digunakan.

BAB III METODOLOGI PENELITIAN

Bab ini berisi tentang metode yang digunakan selama penelitian seperti, metode pengumpulan data, dan metode pengembangan sistem

BAB IV HASIL DAN PEMBAHASAN

Bab ini berisi tentang gambaran umum penerapan algoritma logika fuzzy tahani pada beasiswa dan analisa sistem yang berjalan, literatur sejenis, analisa sistem usulan, perancangan sistem, perancangan database, coding dan testing.

BAB V PENUTUP

Bab ini berisi kesimpulan dari hasil penelitian yang telah dilakukan dan saran-saran dari penulis untuk mengembangkan penelitian berikutnya agar lebih baik lagi.

DAFTAR PUSTAKA

BAB II

LANDASAN TEORI

2.1 Logika Fuzzy

2.1.1 Pengertian Fuzzy

Logika *fuzzy* adalah suatu cara yang tepat untuk memetakan suatu ruang *input* kedalam suatu ruang *output*. Titik awal dari konsep modern mengenai ketidakpastian adalah paper yang dibuat oleh Lofti A Zadeh (1965), dimana Zadeh memperkenalkan teori yang memiliki obyek-obyek dari *himpunan fuzzy* yang memiliki batasan yang tidak presisi dan keanggotaan dalam himpunan *fuzzy*, dan bukan dalam bentuk logika benar (*true*) atau salah (*false*), tapi dinyatakan dalam derajat (*degree*). Konsep seperti ini disebut dengan *Fuzziness* dan teorinya dinamakan *Fuzzy Set Theory*. *Fuzziness* dapat didefinisikan sebagai logika kabur berkenaan dengan semantik dari suatu kejadian, fenomena atau pernyataan itu sendiri. Seringkali ditemui dalam pernyataan yang dibuat oleh seseorang, evaluasi dan suatu pengambilan keputusan. Sebagai contoh:

1. Manajer pergudangan mengatakan pada manajer produksi seberapa banyak persediaan barang pada akhir minggu ini, kemudian manajer produksi akan menetapkan jumlah barang yang harus diproduksi esok hari.
2. Pelayan restoran memberikan pelayanan terhadap tamu, kemudian tamu akan memberikan tip yang sesuai atas baik tidaknya pelayanan yang diberikan.

Fuzzy system (sistem kabur) didasari atas konsep himpunan kabur yang memetakan domain *input* kedalam domain *output*. Perbedaan mendasar himpunan tegas dengan himpunan kabur adalah nilai keluarannya. Himpunan tegas hanya memiliki dua nilai *output* yaitu nol atau satu, sedangkan himpunan kabur memiliki banyak nilai keluaran yang dikenal dengan nilai derajat keanggotaannya.

Logika *fuzzy* adalah peningkatan dari logika *Boolean* yang berhadapan dengan konsep kebenaran sebagian. Dimana logika klasik (*crisp*) menyatakan bahwa segala hal dapat diekspresikan dalam istilah *binary* (0 atau 1, hitam atau putih, ya atau tidak). Logika *fuzzy* menggantikan kebenaran *Boolean* dengan tingkat kebenaran. Logika *fuzzy* memungkinkan nilai keanggotaan antara 0 dan 1, tingkat keabuan dan juga hitam dan putih, dan dalam bentuk *linguistic*, konsep tidak pasti seperti “sedikit”, “lumayan”, dan “sangat”. Logika ini diperkenalkan oleh Dr. Lotfi Zadeh dari Universitas California, Barkeley pada tahun 1965. Logika *fuzzy* telah digunakan pada bidang-bidang seperti taksonomi, topologi, linguistik, teori automata, teori pengendalian, psikologi, *pattern recognition*, pengobatan, hukum, *decision analysis*, *system theory and information retrieval*. Pendekatan *fuzzy* memiliki kelebihan pada hasil yang terkait dengan sifat kognitif manusia, khususnya pada situasi yang melibatkan pembentukan konsep, pengenalan pola, dan pengambilan keputusan dalam lingkungan yang tidak pasti atau tidak jelas.

Ada beberapa alasan mengapa orang memakai logika fuzzy

(Kusumadewi S, Purnomo H, 2010) antara lain:

1. Konsep logika *fuzzy* mudah dimengerti. Konsep matematis yang

mendasari penalaran *fuzzy* sangat sederhana dan mudahdimengerti.

2. Logika *fuzzy* sangat fleksibel.
3. Logika *fuzzy* memiliki toleransi terhadap data-data yang tidak tepat.
4. Logika *fuzzy* mampu memodelkan fungsi-fungsi *nonlinear* yang sangat kompleks.
5. Logika *fuzzy* dapat membangun dan mengaplikasikan pengalaman-pengalaman para pakar secara langsung tanpa harus melalui proses pelatihan.
6. Logika *fuzzy* dapat bekerjasama dengan teknik-teknik kendali secara konvensional.
7. Logika *fuzzy* didasarkan pada bahasa alami.

2.1.2 Konsep Kekaburan (*Fuzziness*) dan Sejarah Logika Fuzzy

Ketidaktegasan atau kekaburan merupakan salah satu ciri dari bahasa sehari-hari manusia untuk mengungkapkan konsep atau gagasan dalam berkomunikasi dengan orang lain. Pada taraf tertentu banyak kata atau istilah yang memuat salah satu bentuk kekaburan. Bentuk-bentuk kekaburan atau ketidakjelasan lainnya adalah:

1. Keambiguan (*ambiguity*), yang terjadi karena suatu kata atau istilah mempunyai makna ganda.
2. Keacakan (*randomness*), yaitu ketidakpastian mengenai sesuatu hal karena hal itu belum terjadi (akan terjadi).
3. Ketidakjelasan akibat tidak lengkapnya informasi yang ada (*incompleteness*).
4. Ketidaktepatan (*imprecision*) yang disebabkan oleh keterbatasan

alat dan metode untuk mengumpulkan informasi.

5. Kekaburan semantik, yaitu kekaburan yang disebabkan karena makna dari suatu kata atau istilah tidak dapat didefinisikan secara tegas, misalnya cantik, tinggi, kaya, pintar dan sebagainya.

Istilah Fuzzy pada tulisan ini lebih menekankan pada bentuk kekaburan semantik. Suatu kata atau istilah dikatakan kabur (*fuzzy*, *vague*) secara semantik apabila kata atau istilah tersebut tidak dapat didefinisikan secara tegas (benar atau salah) apakah suatu objek tertentu memiliki ciri atau sifat yang diungkapkan oleh kata atau istilah itu atau tidak. Meskipun secara umum manusia dapat berkomunikasi secara cukup memadai mengenai makna dari suatu istilah tetapi pasti terdapat perbedaan pemaknaan terhadap istilah tersebut oleh masing-masing individu, yang diakibatkan misalnya oleh persepsi pribadi, lingkungan kebudayaan, latar belakang pengalaman dan pendidikan dan lain-lain. (Susilo, 2006).

Ketidaktegasan semantik ini dari segi keilmuan seringkali menimbulkan masalah karena penelitian ilmiah pada umumnya memerlukan ketepatan dan kepastian berkenaan dengan makna istilah-istilah yang dipakai. Untuk mengatasi masalah tersebut biasanya diciptakan suatu bahasa sendiri sesuai dengan bidang ilmu yang bersangkutan yang mampu menangkap dan mengungkap ketidakjelasan atau kekaburan istilah-istilah dari bahasa sehari-hari secara memadai.

Menurut Hagiwara dalam Nugroho A.S, 2007 bahasa yang dimaksud harus dapat memecahkan permasalahan tidak hanya

dengan menggunakan angka-angka saja. Tetapi juga dapat melakukan perhitungan dengan menggunakan kata-kata (*linguistik*) atau variabel-variabel yang mengandung ketidakpastian atau ketidakjelasan.

2.1.3 Himpunan Keanggotaan

Pada himpunan tegas (*crisp*), nilai keanggotaan suatu item x dalam suatu himpunan A , yang sering ditulis dengan $\mu_A[x]$, memiliki 2 kemungkinan (Kusumadewi S, Purnomo H, 2010) yaitu:

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

Terkadang kemiripan antara keanggotaan *fuzzy* dengan probabilitas menimbulkan kerancuan. Keduanya memiliki nilai pada interval $[0,1]$, namun interpretasi nilainya sangat berbeda antara kedua kasus tersebut. Keanggotaan *fuzzy* memberikan suatu ukuran terhadap pendapat atau keputusan, sedangkan probabilitas mengindikasikan proporsi terhadap keseringan suatu hasil bernilai benar dalam jangka panjang. Misalnya, jika nilai keanggotaan bernilai suatu himpunan *fuzzy* USIA adalah 0.9; maka tidak perlu dipermasalahkan berapa seringnya nilai itu diulang secara individual untuk mengharapkan suatu hasil yang hampir pasti muda. Di lain pihak, nilai probabilitas 0.9 usia berarti 10% dari himpunan tersebut diharapkan tidak muda.

Himpunan *fuzzy* memiliki 2 atribut, yaitu:

1. Linguistik, yaitu penamaan suatu grup yang mewakili suatu

keadaan atau kondisi tertentu dengan menggunakan bahasa alami, seperti: MUDA, PAROBAYA, TUA

2. Numeris, yaitu suatu nilai (angka) yang menunjukkan ukuran dari suatu variable seperti: 40, 25, 50, dsb.

Ada beberapa hal yang perlu diketahui dalam memahami sistem *fuzzy* (Aplikasi logika *fuzzy* untuk pendukung keputusan, Sri Kusumadewi, Hari Purnomo, Edisi kedua, Graha Ilmu, 2010), yaitu:

a. Variabel *fuzzy*

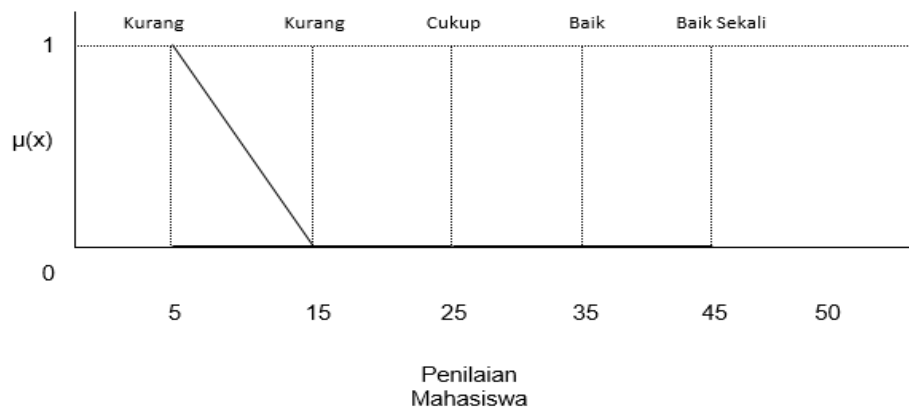
Variable *fuzzy* merupakan variabel yang hendak dibahas dalam suatu sistem *fuzzy*. Contoh: umur, temperature, permintaan, dsb.

b. Himpunan *Fuzzy*

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

Contoh:

- a. Variabel mahasiswa, terbagi menjadi 5 himpunan fuzzy, yaitu: kurang sekali, kurang, cukup, baik dan baik sekali.
- b. Variabel dosen, terbagi menjadi 3 himpunan fuzzy, yaitu: cukup, baik, dan baik sekali. Seperti terlihat pada gambar 2.1.



Gambar 2.3 Himpunan Fuzzy pada Variabel Mahasiswa

a. 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 berupa bilangan positif maupun negatif. Ada kalanya nilai semesta pembicaraan ini tidak dibatasi batas atasnya.

Contoh:

- Semesta pembicaraan untuk variabel mahasiswa: $[0 \ 50]$
- Semesta pembicaraan untuk variabel dosen: $[0 \ 50]$

b. 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 dan bilangan negatif. Contoh domain himpunan *fuzzy*:

a. Kurang Sekali	= [0	15]
b. Kurang	= [5	25]
c. Cukup	= [15	35]
d. Baik	= [25	45]
e. Baik Sekali	= [35	50]

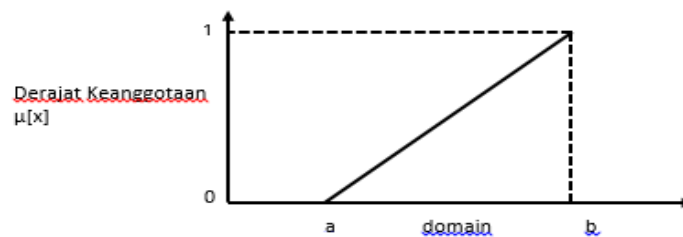
2.1.4 Fungsi Keanggotaan

Fungsi keanggotaan adalah suatu kurva yang menunjukkan pemetaan titik-titik *input* data kedalam nilai keanggotaannya (sering juga disebut dengan derajat keanggotaan) yang memiliki interval antara 0 sampai 1. Salah satu cara yang dapat digunakan untuk mendapatkan nilai keanggotaan adalah dengan melalui pendekatan fungsi. Apabila U menyatakan himpunan universal dan A adalah himpunan fungsi *fuzzy* dalam U , maka A dapat dinyatakan sebagai pasangan terurut (Wang, 1997 dari Wulandari, F., 2005). Ada beberapa fungsi yang bisa digunakan.

a. Representasi Linear

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

Ada 2 keadaan himpunan *fuzzy* yang *linear*. Pertama, kenaikan himpunan dimulai pada nilai domain yang memiliki derajat keanggotaan nol (0) bergerak ke kanan menuju ke nilai domain yang memiliki derajat keanggotaan lebih tinggi (Kusumadewi S, Purnomo H, 2010). Seperti terlihat pada gambar 2.2.

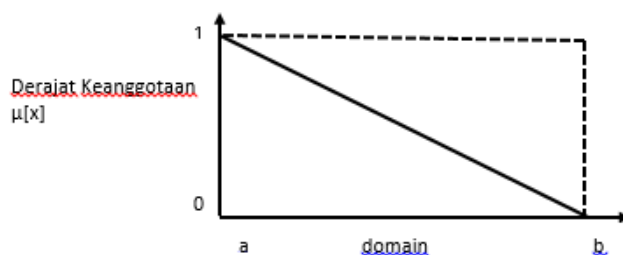


Gambar 2.2 Representasi linier naik

Fungsi keanggotaan:

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

Kedua, merupakan kebalikan dari yang pertama. Garis lurus dimulai dari nilai domain dengan derajat keanggotaan tertinggi pada sisi kiri, kemudian bergerak menurun ke nilai domain yang memiliki derajat keanggotaan lebih rendah. Seperti terlihat pada gambar 2.3.



Gambar 2.3 Representasi linier turun

Fungsi keanggotaan:

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

2.1.5 Operator Dasar Zadeh untuk Operasi Himpunan *Fuzzy*

Seperti halnya himpunan konvensional, ada beberapa operasi yang didefinisikan secara khusus untuk mengkombinasi dan memodifikasi himpunan *fuzzy*. Nilai keanggotaan sebagai hasil dari operasi 2 himpunan sering dikenal dengan nama *fire strength* atau α -predikat. Ada 3 operator dasar yang diciptakan oleh Zadeh (Kusumadewi, 2003), yaitu:

1) Operasi AND

Operator ini berhubungan dengan operasi interseksi pada himpunan. α -predikat sebagai hasil operasi dengan operator 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) Operasi OR

Operator ini berhubungan dengan operasi union pada himpunan. α -predikat sebagai hasil operasi dengan operator 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) Operasi NOT

Operator ini berhubungan dengan operasi komplemen pada himpunan. α -predikat sebagai hasil operasi dengan operator NOT diperoleh dengan mengurangi nilai keanggotaan elemen pada himpunan yang bersangkutan dari 1.

$$\mu_{\bar{A}} = 1 - \mu_A[x]$$

1.1.6 Perbedaan Himpunan Fuzzy dengan Probabilistik

Teori probabilitas adalah suatu cabang matematika yang sudah mulai dikembangkan sejak abad 17 untuk menyelidiki gejala ketidakpastian, dan telah berhasil diterapkan dalam banyak bidang. Pada waktu teori kabur mulai dipublikasikan pada tahun 1965 para ahli teori probabilitas menganggap bahwa teori baru itu sama saja dengan teori probabilitas yang sudah lama dikenal dalam dunia matematika, karena kedua teori tersebut menangani gejala ketidakpastian dan keduanya mempunyai nilai dalam selang tertutup $[0, 1]$.

1]. Maka mereka menganggap bahwa teori kabur itu bukanlah barang baru dan tidak menambah apa-apa dalam khasanah ilmu pengetahuan. Pandangan tersebut sama sekali tidak benar dan untuk meyakinkan hal itu kiranya perlu dijelaskan mengapa kedua teori tersebut tidaklah sama.

Baik teori probabilitas maupun teori kabur keduanya memang menangani menangani gejala ketidakjelasan, tetapi ketidakpastian yang berbeda jenisnya. Ketidakpastian yang digarap dalam teori probabilitas adalah keacakan (*randomness*), yaitu ketidakpastian mengenai sesuatu hal yang disebabkan karena hal itu belum terjadi (akan terjadi). Ketidakpastian semacam itu akan hilang, dan akan berubah menjadi kepastian, pada waktu hal itu telah terjadi. Misalnya seseorang mengalami ketidakpastian mengenai apakah lamaran pekerjaannya ke sebuah perusahaan diterima atau tidak. Pada waktu ia menerima surat dari perusahaan itu, yang memberitahukan bahwa ia diterima, ketidakpastian itu langsung hilang dan berubah menjadi kepastian. Sedangkan ketidakpastian yang digarap dalam teori himpunan kabur adalah kekaburan semantik mengenai suatu kata atau istilah yang tidak dapat didefinisikan secara tegas. Kekaburan semantik itu tetap ada (tidak berubah) meskipun halnya telah terjadi. Misalnya seseorang merasa tidak pasti apakah besok pagi cuacanya akan dingin atau tidak. Ketidakpastiannya mengenai keadaan cuaca besok pagi itu adalah keacakan (dengan peluang tertentu akan terjadi) yang besok pagi akan berubah menjadi kepastian, sedangkan dinginnya cuaca adalah suatu kekaburan semantik (dengan fungsi keanggotaan tertentu) yang besok pagi atau kapanpun tetap merupakan kekaburan. Lagi pula dalam teori probabilitas jumlah peluang dari semua kemungkinan yang dapat terjadi harus sama dengan satu. Hal semacam itu tidak disyaratkan dalam teori himpunan kabur. Kedua teori itu memang berbeda, tetapi tidak perlu

dipertentangkan, justru sebaliknya kedua teori itu dapat bekerjasama dan saling melengkapi. Misalnya dalam istilah sangat mungkin terjadi terkandung konsep keacakan maupun kekaburan semantik yang dapat digarap oleh kedua teori tersebut.

Berkaitan dengan kerjasama itu, pada awal dekade yang lalu Zadeh memperkenalkan suatu konsep baru yang disebutnya *komputasi lunak (soft computing)*, yang merupakan sinergi dari beberapa teori dan metodologi untuk menghasilkan suatu sistem cerdas yang semakin mendekati kecerdasan manusia dalam bernalar dan belajar dengan data-data yang tidak pasti dan tidak tepat. Unsur- unsur pokok dari komputasi lunak itu adalah teori kabur (*fuzzy*), jaringan saraf (*neural network*), algoritma genetik (*genetic algorithms*), teori probabilitas, dan teori Chaos

1.1.7 Keunggulan Logika Fuzzy

Ada beberapa alasan mengapa orang menggunakan logika fuzzy, antara lain:

1. Konsep logika fuzzy mudah dimengerti. Konsep matematis yang mendasari penalaran fuzzy dan mudah dimengerti.
2. Logika fuzzy sangat fleksibel.
3. Logika fuzzy memiliki toleransi terhadap data-data yang tidak tepat.
4. Logika fuzzy mampu memodelkan fungsi-fungsi nonlinear yang sangat kompleks.
5. Logika fuzzy dapat membangun dan mengaplikasikan pengalaman-pengalaman para pakar secara langsung tanpa harus melalui pelatihan.
6. Logika fuzzy didasarkan pada bahasa alami.
7. Sangat mudah dihibridkan dengan teknologi lain, misalnya algoritma genetika, jaringan saraf, dan lain-lain.

2.2 Pangkalan Data

2.2.1 Pengertian Pangkalan Data

Pangkalan data (*database*) merupakan kumpulan dari data yang saling berhubungan satu dengan yang lainnya, tersimpan di perangkat keras komputer dan digunakan perangkat lunak untuk memanipulasinya. Sistem pangkalan data (*database system*) adalah suatu sistem informasi yang mengintegrasikan kumpulan data yang saling berhubungan satu dengan yang lainnya dan membuatnya tersedia untuk beberapa aplikasi dalam suatu organisasi.

Sebagaimana diketahui manajemen modern mengikutsertakan informasi sebagai sumber daya yang setara dengan sumber daya manusia, uang mesin, dan material. Informasi adalah suatu bentuk penyajian data yang melalui mekanisme pemrosesan, berguna bagi pihak tertentu misalnya manajer. Bagi pihak manajer informasi sebagai pengambil keputusan.



Gambar 2.4 Pengolahan data menjadi informasi

2.2.2 Sejarah Pangkalan Data

Dari awal era komputer, penyimpanan dan penggunaan data menjadi fokus aplikasi yang utama. Sistem manajemen pangkalan data (*database management system* disingkat DBMS) *general*

purpose pertama, yang didesain oleh Charles Bachman di General Electric pada awal tahun 1960-an, disebut Integrated Data Store (IDS). DBMS ini membentuk dasar untuk model data jaringan, yang distandarisasi oleh *Conference on Data System Language* (CODASYL) dan sangat mempengaruhi sistem pangkalan data sepanjang tahun 1960-an. Bachman adalah orang pertama yang menerima ACM's Turing Award (semacam hadiah Nobel untuk ilmu komputer) untuk karyanya dalam bidang pangkalan data beliau menerima penghargaan itu pada tahun 1973.

Pada akhir 1960-an, IBM mengembangkan *Information Management System* (IMS) DBMS, yang digunakan dalam banyak instalasi besar hingga saat ini. IMS membentuk dasar bagi kerangka kerja representasi data alternatif yang disebut model data hierarki. Pada saat yang sama, dikembangkan pula sistem SABRE untuk membuat reservasi penerbangan penerbangan yang dikembangkan oleh *American Airlines* dan IBM, dan memungkinkan beberapa orang mengakses data yang sama melalui jaringan komputer.

Pada tahun 1970, Edgar Codd, di IBM *San Jose Research Laboratory*, mengusulkan sebuah kerangka kerja baru untuk representasi data, disebut model data relasional. Hal ini terbukti menjadi faktor yang sangat penting dalam pengembangan sistem pangkalan data, memicu pesatnya perkembangan beberapa DBMS yang didasarkan pada model relasional, bersamaan dengan banyaknya hasil teoritis yang menempatkan field pada satu fondasi perusahaan. Pada tahun 1981, Codd meraih Turing Award untuk

karyanya yang luar biasa ini. Sistem pangkalan data ditetapkan sebagai suatu disiplin akademik, dan popularitas DBMS relasional kemudian mengubah susunan komersial. Manfaatnya diakui secara luas, dan penggunaan DBMS untuk mengelola data perusahaan menjadi praktik standar (Ramakhrisnan, 2000)

2.2.3 Pangkalan Data Relasional

Model relasional lebih populer dari pada tipe model pangkalan data lain seperti *hierarki* dan *network*, karena kesederhanaannya. Relasional Database Management System (RDBMS) menjadi sangat populer karena mudah digunakan dan strukturnya lebih fleksibel. Komponen dari model relasional adalah:

1. Kumpulan objek atau relasi yang menyimpan data.
2. Kumpulan operator yang dapat digunakan pada relasi untuk menghasilkan relasi lain.
3. Integritas data untuk akurasi dan konsistensi.

Database relasional menggunakan relasi atau tabel dua dimensi untuk menyimpan informasi, yang masing-masing relasi tersusun atas tupel atau baris dan atribut. Relasi dirancang sedemikian rupa sehingga dapat menghilangkan kemubaziran data dan menggunakan kunci tamu untuk berhubungan dengan relasi lain.

Ada beberapa sifat yang melekat pada suatu relasi, yaitu;

1. Tidak ada tupel (baris) yang kembar.
2. Urutan tupel tidaklah penting (tupel-tupel dapat dipandang dalam

sebarang urutan)

3. Setiap atribut memiliki nama yang unik.
4. Letak atribut bebas (urutan atribut tidak penting).
5. Setiap atribut memiliki nilai tunggal dan jenisnya sama untuk semua tupel.

Pada model relasional, jumlah tupel suatu relasi disebut **kardinalitas** dan jumlah atribut suatu relasi disebut **derajat** (*degree*) atau terkadang disebut *arity*. Relasi yang berderajat satu (hanya memiliki satu atribut) disebut **unary**. Relasi yang berderajat dua disebut **binary** dan relasi yang berderajat tiga disebut **ternary**. Relasi yang berderajat n disebut **n -ary**. (Kadir, 1999)

2.3 Fuzzy Tahani

Fuzzy Tahani adalah salah satu cabang dari logika *fuzzy*, yang merupakan salah satu metode *fuzzy* yang menggunakan basis data standar. Tahani mendeskripsikan suatu metode pemrosesan *query fuzzy*, dengan didasarkan atas manipulasi bahasa yang dikenal dengan nama *SQL (Structured Query Language)*, sehingga model *fuzzy Tahani* sangat tepat digunakan dalam proses pencarian data yang tepat dan akurat (Anggraeni, Indarto, Kusumadewi, 2004 dalam Amalia, L., Fananie, Z. B., Utama, D. N., 2010).

Model Tahani tersusun atas tahapan yaitu:

- a. Menggambarkan Fungsi Keanggotaan

Fungsi keanggotaan (membership function) adalah suatu kurva yang menunjukkan pemetaan titik-titik input data ke dalam nilai

keanggotaannya (sering juga disebut dengan derajat keanggotaan) yang memiliki internal antara 0 sampai 1, salah satu cara yang dapat digunakan untuk mendapatkan nilai keanggotaan melalui pendekatan fungsi. Beberapa fungsi yang dapat digunakan yaitu Representasi kurva Linier, Representasi Kurva Segitiga, Representasi Kurva Trapesium, Representasi Kurva Bentuk Bahu, Representasi Kurva-S, Representasi Kurva Bentuk Lonceng (Bell Curve). Masing-masing fungsi tersebut, akan menghasilkan nilai antara “0” dan “1” dengan cara yang berbeda, sesuai dengan jenis representasi yang digunakan.

b. Fuzzyfikasi

Fuzzyfikasi adalah fase pertama dari perhitungan fuzzy yaitu pengubahan nilai tegas ke nilai fuzzy. Prosesnya adalah sebagai berikut: Suatu besaran analog dimasukkan sebagai masukan (crisp input), lalu input tersebut dimasukkan pada batas scope dari membership function. Membership function ini biasanya dinamakan membership function input. Keluaran dari proses fuzzifikasi ini adalah sebuah nilai input fuzzy atau yang biasanya dinamakan fuzzy input.

c. Fuzzyfikasi Query

Fuzzyfikasi Query diasumsikan sebuah query konvensional (nonfuzzy) DBMS yang akan mencoba membuat dan menerapkan sebuah sistem dasar logika fuzzy query.

d. Operator Dasar Zadeh untuk Operasi Himpunan Fuzzy.

Nilai keanggotaan sebagai dari 2 himpunan fuzzy dikenal dengan nama Fire Strength atau α -predikat. Sangat mungkin digunakan operator dasar dalam proses query berupa operator AND dan OR.

α -predikat sebagai hasil operasi dengan operator AND diperoleh dengan mengambil nilai keanggotaan terkecil antar elemen pada himpunan-himpunan yang bersangkutan, dinotasikan : $\mu_{A \cap B} = \min(\mu_A[x], \mu_B[y])$. Sedangkan untuk hasil operasi dengan operator OR diperoleh dengan mengambil nilai keanggotaan terbesar antar elemen pada himpunan-himpunan yang bersangkutan, dinotasikan : $\mu_{A \cup B} = \max(\mu_A[x], \mu_B[y])$.

Alternatif yang direkomendasikan adalah alternatif yang memiliki nilai Fire Strength atau tingkat kesesuaian dengan kriteria pilihan di atas angka 0 (nol) sampai dengan angka 1 (satu).

Dengan menggunakan database standar, kita dapat mencari data-data dengan spesifikasi tertentu dengan menggunakan query. Misal kita ingin mendapatkan informasi tentang nama-nama karyawan yang usianya kurang dari 35 tahun. Contoh data yang digunakan tersaji pada tabel 2.1, maka kita bisa ciptakan suatu query :

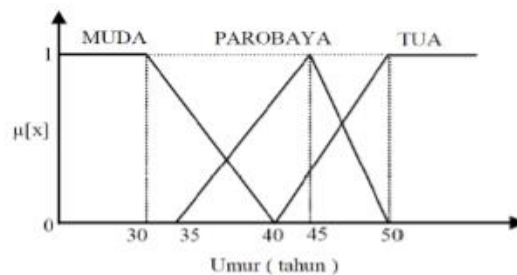
Select Nama From Karyawan Where (Umur <35)

Tabel 2.1 Contoh data karyawan

Nip	Nama	Umur	Thn Masuk	Gaji
01	Lia	30	1996	750.000
02	Iwan	48	1985	1.500.000
03	Sari	36	1988	1.255.000
04	Andi	37	1998	1.040.000
05	Budi	42	1990	950.000
06	Amir	39	1989	1.600.000
07	Rian	37	1997	1.250.000
08	Kiki	32	2001	550.000
09	Alda	35	1999	735.000
10	Yoga	25	2000	860.000

Sehingga muncul nama-nama Lia, Kiki, dan Yoga. Namun seseorang terkadang membutuhkan informasi dari data-data yang bersifat ambiguous.

Apabila hal ini terjadi, maka bisa digunakan basis data Fuzzy. Fungsi keanggotaan untuk variabel usia diilustrasikan pada gambar 2.5 sebagai berikut:



Gambar 2.5 Fungsi Keanggotaan untuk Variabel Usia

Penentuan nilai derajat keanggotaan muda, bisa menggunakan persamaan 1 atau 2 atau 3. Untuk penentuan nilai derajat keanggotaan parobaya menggunakan persamaan 4 atau 5 atau 6, sedangkan untuk menentukan derajat keanggotaan tua menggunakan persamaan 7 atau 8 atau 9 sebagai berikut:

$$\begin{aligned} \mu_{\text{Muda}}[x] & \begin{cases} 1 & x < 30 \\ (40 - x)/10 & 30 \leq x \leq 40 \\ 0 & x > 40 \end{cases} & \begin{matrix} (1) \\ (2) \\ (3) \end{matrix} \\ \mu_{\text{Parobaya}}[x] & \begin{cases} 0 & x < 30 \text{ atau } x > 50 \\ (x - 35)/10 & 35 \leq x \leq 45 \\ (50 - x)/5 & 45 \leq x \leq 50 \end{cases} & \begin{matrix} (4) \\ (5) \\ (6) \end{matrix} \\ \mu_{\text{Tua}}[x] & \begin{cases} 0 & x < 40 \\ (x - 40)/10 & 40 \leq x \leq 50 \\ 1 & x > 50 \end{cases} & \begin{matrix} (7) \\ (8) \\ (9) \end{matrix} \end{aligned}$$

Berdasarkan pengolahan fungsi keanggotaan untuk variabel usia dengan derajat keanggotaan muda, parobaya dan tua didapat hasil seperti disajikan pada tabel 2.2 berikut:

Tabel 2.2 Tabel karyawan berdasarkan umur

Nip	Nama	Umur	Muda	Derajat Keanggotaan	Tua
-----	------	------	------	---------------------	-----

				[x] Parobaya	
01	Lia	30	1	0	0
02	Iwan	48	0	0,4	0,8
03	Sari	36	0,4	0,1	0
04	Andi	37	0,3	0,2	0
05	Budi	42	0	0,7	0,2
06	Amir	39	0,1	0,4	0
07	Rian	37	0,3	0,2	0
08	Kiki	32	0,8	0	0
09	Alda	35	1,5	0	0
10	Yoga	25	1	0	0

Ada beberapa query yang dapat diberikan untuk menggali informasi lainnya, misalnya: Siapa sajakah karyawan yang masih muda tapi memiliki gaji tinggi ?. Kita bisa menggunakan query di bawah ini dan dari hasil query tersebut didapatkan hasil seperti tersaji pada tabel 2.3.

SELECT nama FROM Karyawan Where Umur="Muda" dan Gaji="Tinggi"

Tabel 2.3 Data Karyawan yang masih muda dan memiliki gaji tinggi

Nip	Nama	Umur	Gaji/Bln (Rp)	Muda	Derajat Keanggotaan [x] Parobaya	Tua
03	Sari	36	1.255.000	0,4	0,1	0
07	Rian	37	1.250.000	0,3	0,2	0
06	Amir	39	1.600.000	0,1	0,4	0
04	Andi	37	1.040.000	0,3	0,2	0
01	Lia	30	750.000	1	0	0
02	Iwan	48	1.500.000	0	0,4	0,8
05	Budi	42	950.000	0	0,7	0,2
08	Kiki	32	550.000	0,8	0	0
09	Alda	35	735.000	0,5	0	0
10	Yoga	25	860.000	1	0	0

2.4 Basis Data

Data merupakan representasi dari fakta atau gambaran mengenai suatu obyek atau kejadian. Data dinyatakan dengan nilai dalam bentuk angka, deretan karakter atau simbol. Misalnya fakta atau kenyataan

tentang biodata siswa seperti Nama lengkap, Alamat, Nama orang tua dan lain-lain. Contoh lain dari fakta mengenai kejadian atau peristiwa misalnya adalah transaksi penjualan online yang meliputi data waktu, penjual, pembeli, nilai transaksi dan lain-lain. Basis data (database) adalah Cara mendokumentasikan berbagai macam data yang kemudian dimanajemen dengan sebuah sistem untuk kemudian disimpan dalam sebuah media penyimpanan. Dalam basis data, data yang ada tidak hanya sekedar diletakkan dan disimpan begitu saja dalam sebuah media penyimpanan, Akan tetapi dikelola dengan sistem pengaturan tertentu. Dengan demikian, data dengan jumlah besar dan kompleks dapat tersusun sangat baik sehingga memungkinkan pengaksesan data dengan mudah dan cepat oleh pengguna. Basis data juga bisa diartikan sekumpulan informasi yang sangat komplek.

2.4.1 Konsep DBMS (*Database Management System*)

Kumpulan atau gabungan *database* dengan perangkat lunak dinamakan *Database Management System* atau disingkat menjadi DBMS. DBMS merupakan koleksi terpadu dari *database* dan program-program komputer yang digunakan untuk mengakses dan memelihara *database*. Program-program tersebut menyediakan berbagai fasilitas operasi untuk memasukkan, melacak, dan memodifikasi data kedalam *database*, mendefinisikan data baru, serta mengolah data menjadi informasi yang dibutuhkan. (Ladjamudin, 2005).

Tujuan Utama dari DBMS adalah untuk menyediakan suatu lingkungan yang mudah dan efisien penggunaan, penarikan dan penyimpanan data dan informasi. (Ladjamudin, 2005).

Menurut Lajamudin (2005), dibandingkan dengan sistem tradisional yang berbasis kertas, DBMS memiliki empat keunggulan sebagai berikut:

1. Kepraktisan; sistem yang berbasis kertas akan menggunakan kertas yang sangat banyak untuk menyimpan informasi, Sedangkan DBMS menggunakan media penyimpanan sekunder yang berukuran kecil tetapi padat informasi.
2. Kecepatan, mesin dapat mengambil atau mengubah data jauh lebih cepat dari padamanusia
3. Mengurangi kejemuian, orang cenderung menjadi bosan kalau melakukan tindakan-tindakan yang berulang yang menggunakan tangan (misalnya harus mengganti suatu informasi).
4. Kekinian, informasi yang tersedia pada DBMS akan bersifat mutakhir dan akurat setiap saat.

2.5 Fuzzy Database

Sistem basis data (*database system*) adalah suatu sistem informasi yang mengintegrasikan kumpulan data yang saling berhubungan dan membuatnya tersedia untuk beberapa aplikasi (Kusumadewi S, Purnomo H, 2004).

Database adalah kumpulan dari data yang saling berhubungan satu dengan yang lainnya, tersimpan di perangkat keras komputer dan digunakan perangkat lunak untuk memanipulasinya. Sedangkan sistem basis data adalah suatu sistem informasi yang mengintegrasikan kumpulan data yang saling berhubungan satu dengan yang lainnya dan membuatnya tersedia

untuk beberapa aplikasi dalam suatu organisasi (Eliyani, Pujiyanto, U., Rosyadi, D., 2009, dari Kusumadewi, S., 2004).

Fuzzifikasi query diasumsikan sebuah *query* konvensional (*nonfuzzy*), DBMS yang akan mencoba membuat dan menerapkan sebuah sistem dasar logika *fuzzy query* (*fuzzy logic based querying system*). Kelebihan *query* fuzzifikasi yaitu dapat mencapai kelenturan (*flexibility*) dari DBMS, penanganan *error* otomatis, pencarian yang fleksibel, dan kesanggupan merespon kosong.

Awal penanganan ketidakpastian dengan manajemen basis data dikembangkan di dalam kerangka manajemen sistem basis data yang bukan *fuzzy*. Biasanya, sistem ini berhadapan dengan evaluasi dan konstruksi tentang *fuzzy query* dengan *database* yang bersifat tegas, dan mengabaikan permasalahan dalam penyajian langsung dari data *fuzzy* di DBMS (Mashkuri Hj Yaacob, 1997:43 dalam Setiyowati, M.I, Seta, B.A, 2007).

Sebagian besar basis data *fuzzy* merupakan perluasan dari model basis data relasional, namun dikemas dalam formulasi yang berbeda tergantung pada tipe ambiguitas yang akan diekspresikan dan dimanipulasi. Tahani mendeskripsikan suatu metode untuk melakukan pengolahan *query fuzzy* didasarkan pada manipulasi data. Disini konsep teori *fuzzy* lebih banyak digunakan untuk melakukan pengolahan *query*. Basis data yang diusulkan oleh Zadeh, mengekspresikan ambiguitas data dengan cara memperluas model data. Perluasan dilakukan dengan cara menggunakan relasi *fuzzy* berupa *grade* yang ditambahkan pada relasi standar (Kusumadewi S, 2007).

Sebagian besar basis data standar diklasifikasikan berdasarkan bagaimana data tersebut dipandang oleh pengguna. Pada kenyataannya, terkadang pengguna membutuhkan informasi dari data-data yang bersifat *ambiguous*, contoh “mencari data karyawan yang masih muda dan memiliki gaji yang tinggi”. Apabila ini terjadi, dapat digunakan basis data *fuzzy*. Selama ini sudah ada penelitian tentang basis data *fuzzy*. Salah satu diantaranya adalah model Tahani. Basis data *fuzzy* model Tahani masih tetap menggunakan relasi standar, hanya saja model ini menggunakan teori himpunan *fuzzy* untuk mendapatkan informasi pada *query*-nya. Tahani mendeskripsikan suatu metode pemrosesan *query fuzzy* dengan didasarkan atas manipulasi bahasa yang dikenal dengan nama SQL (Anggraeni, R., Indarto, W., Kusumadewi, S., 2004).

2.6 Beasiswa

2.6.1 Pengertian Beasiswa

Beasiswa adalah pemberian berupa bantuan keuangan yang diberikan kepada perorangan, mahasiswa atau pelajar yang digunakan demi keberlangsungan pendidikan yang ditempuh. Menurut Murniasih (2009) beasiswa diartikan sebagai bentuk penghargaan yang diberikan kepada individu agar dapat melanjutkan pendidikan ke jenjang yang lebih tinggi. Penghargaan itu dapat berupa akses tertentu pada suatu institusi atau penghargaan berupa bantuan keuangan. Pada dasarnya, beasiswa adalah penghasilan bagi yang menerimanya. Hal ini sesuai dengan ketentuan pasal 4 ayat (1) Undang-undang PPh/2000. Disebutkan pengertian penghasilan adalah tambahan kemampuan

ekonomis dengan nama dan dalam bentuk apa pun yang diterima atau diperoleh dari sumber Indonesia atau luar Indonesia yang dapat digunakan untuk konsumsi atau menambah kekayaan Wajib Pajak. Karena beasiswa bisa diartikan menambah kemampuan ekonomis bagi penerimanya, berarti beasiswa merupakan penghasilan.

Beasiswa dapat diberikan oleh lembaga pemerintah, perusahaan ataupun yayasan. Pemberian beasiswa dapat dikategorikan pada pemberian cuma-cuma ataupun pemberian dengan ikatan kerja (biasa disebut ikatan dinas) setelah selesainya pendidikan. Lama ikatan dinas ini berbeda-beda, tergantung pada lembaga yang memberikan beasiswa tersebut. beasiswa juga banyak diberikan kepada perkelompok (group) misalnya ketika ada event perlombaan yang diadakan oleh lembaga pendidikan, dan salah satu hadiahnya adalah beasiswa.

2.6.2 Jenis-jenis Beasiswa

Menurut Murniasih (2009), ada beberapa jenis beasiswa yaitu:

a. Beasiswa Penghargaan

Beasiswa ini biasanya diberikan kepada kandidat yang memiliki keunggulan akademik. Beasiswa ini diberikan berdasarkan prestasi akademik mereka secara keseluruhan. Misalnya, dalam bentuk Indeks Prestasi Kumulatif (IPK). Meski sangat kompetitif, beasiswa ini ada dalam berbagai bentuk.

b. Beasiswa Bantuan

Jenis beasiswa ini adalah untuk mendanai kegiatan akademik para mahasiswa yang kurang beruntung, tetapi

memiliki prestasi. Komite beasiswa biasanya memberikan beberapa penilaian pada kesulitan ini, misalnya, seperti pendapatan orangtua, jumlah saudara kandung yang sama-sama tengah menempuh studi, pengeluaran, biaya hidup, dan lain-lain.

c. Beasiswa Atletik

Universitas biasanya merekrut atlet populer untuk diberikan beasiswa dan dijadikan tim atletik perguruan tinggi mereka. Banyak atlet menyelesaikan pendidikan mereka secara gratis, tetapi membayarnya dengan prestasi olahraga. Beasiswa seperti ini biasanya tidak perlu dikejar, karena akan diberikan kepada mereka yang memiliki prestasi.

d. Beasiswa Penuh

Banyak orang menilai bahwa beasiswa diberikan kepada penerimanya untuk menutupi keperluan akademik secara keseluruhan. Jika Anda benar-benar beruntung, tentunya Anda akan mendapatkan beasiswa seperti ini. Beasiswa akan diberikan untuk menutupi kebutuhan hidup, buku, dan biaya pendidikan. Namun, banyak beasiswa lainnya meng-cover biaya hidup, buku, atau sebagian dari uang sekolah.

Salah satu beasiswa yang terdapat di STMIK Bumigora Mataram adalah Beasiswa Peningkatan Prestasi Akademik (PPA).

Beasiswa Peningkatan Prestasi Akademik adalah beasiswa yang diberikan untuk peningkatan pemerataan dan kesempatan belajar bagi mahasiswa yang mengalami kesulitan membayar

biaya pendidikannya sebagai akibat krisis ekonomi, terutama bagi mahasiswa yang berprestasi akademik.

Adapaun tujuan beasiswa Peningkatan Prestasi Akademik secara umum yaitu :

- 1) Meningkatkan pemerataan dan kesempatan belajar bagi mahasiswa yang mengalami kesulitan membayar pendidikan.
- 2) Mendorong dan mempertahankan semangat belajar mahasiswa agar mereka dapat menyelesaikan studi/pendidikan tepat waktunya.
- 3) Mendorong untuk meningkatkan prestasi akademik sehingga memacu peningkatan kualitas pendidikan.

2.7 Waterfall

Model *Waterfall* sering disebut model sekuensial linier (*Sequential Linear*) atau alur hidup klasik (*classic life cycle*). Model *waterfall* menyediakan pendekatan alur hidup perangkat lunak secara sekuensial atau terurut dimulai dari analisis, desain, pengodean, pengujian, dan tahap pendukung (pressman 2002).

- a. Analisis kebutuhan perangkat lunak. Proses pengumpulan kebutuhan diintensifkan dan difokuskan, khususnya pada perangkat lunak, untuk memahami sifat program yang dibangun, perekayasa perangkat lunak (analisis) harus memahami domain informasi, tingkah laku, unjuk karya, dan antar muka (interface) yang diperlukan.

- b. Design. Desain perangkat lunak sebenarnya adalah proses multi langkah yang berfokus pada empat atribut sebuah program yang berbeda; struktur data, arsitektur perangkat lunak, representasi perangkat lunak yang dapat diperkirakan demi kualitas sebelum dimulai pemunculan kode.
- c. Generasi kode. Desain harus diterjemahkan kedalam bentuk mesin yang bisa dibaca, langkah pembuatan kode melakukan tugas ini jika desain dilakukan dengan cara yang lengkap, pembuatan kode dapat dilakukan secara mekanis.
- d. Test atau Pengujian. Sekali kode dibuat pengujian program dimulai. Proses pengujian berfokus pada logika internal perangkat lunak, memastikan bahwa semua pernyataan sudah diuji, dan pada eksternal fungsional yaitu mengarahkan pengujian untuk menemukan kesalahan-kesalahan dan memastikan bahwa input dibatasi akan memberikan hasil aktual yang sesuai dengan hasil dibutuhkan.
- e. Pendukung atau pemeliharaan

Tidak menutup kemungkinan sebuah perangkat lunak mengalami perubahan ketika sudah dikirimkan ke *user*. Perubahan bisa terjadi Karena adanya kesalahan yang muncul dan tidak terdeteksi saat pengujian atau perangkat lunak harus beradaptasi dengan lingkungan baru. Tahap pendukung atau perawatan dapat mengulangi proses pengembangan mulai dari analisis spesifikasi untuk perubahan perangkat lunak yang sudah ada, tetapi tidak untuk membuat perangkat lunak baru.

2.8 Perangkat Analisa Sistem

2.8.1. *Data Flow Diagram (DFD)*

Data Flow Diagram merupakan model dari sistem untuk menggambarkan pembagian sistem ke modul yang lebih kecil. Salah satu keuntungan menggunakan DFD adalah memudahkan pemakai atau user yang kurang menguasai komputer untuk mengerti sistem yang akan dikerjakan. DFD terdiri dari 3 bagian yaitu, (Ladjamudin, 2005) :

1. Diagram Konteks

Diagram konteks adalah diagram yang terdiri dari suatu proses dan menggambarkan ruang lingkup suatu sistem. Diagram konteks merupakan level tertinggi dari DFD yang menggambarkan seluruh input ke sistem ataupun output ke sistem. Ia akan memberi gambaran tentang keseluruhan sistem. Sistem dibatasi oleh boundary (dapat digambarkan dengan garis putus). Dalam diagram konteks harus ada satu proses, tidak boleh ada store dalam diagram konteks.

2. Diagram Nol/Zero (*Overview Diagram*)

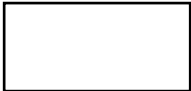

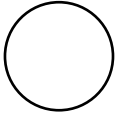

Diagram nol adalah yang menggambarkan proses dan data flow diagram. Diagram nol memberikan pandangan secara menyeluruh mengenai sistem yang ditangani, mengenai tentang fungsi-fungsi utama atau proses yang ada, aliran data, dan eksternal entity.

3. Diagram Rinci (Level Diagram)

Diagram rinci adalah diagram yang menguraikan proses apa yang ada dalam diagram zero atau diagram level di atasnya.

Tabel Elemen Dasar *Data Flow Diagram* versi Yourdon, De Marco

Tabel 2.4 Simbol Data Flow Diagram

Simbol	Keterangan
	Entitas Luar menggambarkan sesuatu yang adadi luar sistem, tetapi ia memberikan data ke dalam sistem atau mendapat data dari sistem.
	Arus Data menggambarkan arus data yang berupa masukan untuk sistem atau hasil dari sistem.
	Proses menggambarkan apa yang dikerjakan oleh sistem.
	Penyimpanan Data menggambarkan tempat penyimpanan data yang ada dalam sistem.

(Sumber: Ladjamudin, 2005)

2.8.2. Entity Relational Diagram (ERD)

Entity Relationship Diagram (ERD) adalah notasi yang digunakan untuk melakukan aktifitas pemodelan data. Atribut dari masing-masing objek data yang ditulis pada ERD dapat digambarkan dengan menggunakan deskripsi objek data. (Pressman, 2002)

Model E-R terdiri dari beberapa komponen dasar yaitu sebagai berikut:

1. Entitas

Entitas adalah sesuatu atau objek di dunia nyata yang dapat dibedakan dari sesuatu atau objek yang lainnya. Sebagai contoh,

setiap mahasiswa dalam suatu universitas adalah suatu entitas. Setiap fakultas dalam suatu universitas adalah juga suatu entitas. Dapat dikatakan bahwa entitas bersifat konseptual/abstrak atau nyata hadir di dunia nyata.

2. Atribut

Atribut adalah properti deskriptif yang dimiliki oleh setiap anggota dari himpunan entitas. Sebagai contoh entitas mahasiswa, atribut-atribut yang dimiliki adalah nim, nama mahasiswa, alamat dan lain-lain.

3. Hubungan antar relasi (Relationship)

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

4. Kardinalitas/Derajat Relasi

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

Kardinalitas relasi yang terjadi di antara dua himpunan entitas dapat berupa :

a. Satu ke Satu (One to One)

Setiap entitas pada himpunan entitas A berhubungan dengan banyak dengan satu entitas pada himpunan entitas B, begitupun sebaliknya.

b. Satu ke Banyak (One to Many)

Setiap entitas pada himpunan entitas A dapat berhubungan dengan banyak entitas pada himpunan entitas B, tetapi tidak sebaliknya, dimana setiap entitas pada himpunan entitas B berhubungan dengan paling banyak dengan satu entitas pada himpunan entitas A.

c. Banyak ke Satu (Many to One)

Setiap entitas pada himpunan entitas A berhubungan dengan paling banyak dengan satu entitas pada himpunan entitas B, tetapi tidak sebaliknya dengan entitas B.

d. Banyak ke Banyak (Many to Many)

Setiap entitas pada himpunan entitas A dapat berhubungan dengan banyak entitas pada himpunan entitas B, dan demikian sebaliknya, dimana setiap entitas pada himpunan entitas B dapat berhubungan dengan banyak entitas pada himpunan entitas A.

2.8.3 Normalisasi

Normalisasi adalah proses pengelompokan data kedalam bentuk tabel atau relasi atau file untuk menyatakan entitas dan hubungan mereka sehingga terwujud satu bentuk database yang mudah untuk dimodifikasi, Ada beberapa bentuk normal yaitu (Ladjamudin, 2005) :

1. Bentuk Normal I (*First Normal Form / 1-NF*).

Pada tahap ini dilakukan penghilangan beberapa group elemen yang berulang agar menjadi satu harga tunggal yang berinteraksi di antara setiap baris pada suatu tabel, dan setiap atribut harus mempunyai nilai data yang atomic (bersifat *atomic value*).

Syarat normal kesatu:

- a. Setiap data dibentuk dalam flat file, data dibentuk dalam satu record demi satu *record* nilai dari *field* berupa "*atomic value*".
- b. Tidak ada set attribute yang berulang atau bernilai ganda.
- c. Telah dibentuknya *primary key* untuk tabel/relasi tersebut.
- d. Tiap attribute hanya memiliki satu pengertian.

2. Bentuk Normal II (*Second Normal Form/ 2-NF*)

Suatu relasi memenuhi 2-NF jika dan hanya jika :

- a. Bentuk data telah memenuhi kriteria bentuk normal kesatu.
- b. Attribute bukan kunci haruslah memiliki ketergantungan fungsi utama pada kunci sepenuhnya.

3. Bentuk Normal III (*Third Normal Form / 3-NF*).

Suatu relasi memenuhi bentuk III (3-NF) jika dan hanya jika :

- a. Bentuk data telah memenuhi kriteria bentuk normal kedua.
- b. Attribute bukan kunci haruslah tidak memiliki ketergantungan transitif, seluruh atribut bukan kunci pada suatu relasi hanya memiliki ketergantungan fungsional pada primary key di relasi itu saja.

2.7.4 Flowchart

Flowchart adalah bagan-bagan yang mempunyai arus menggambarkan langkah-langkah suatu masalah. Flowchart merupakan cara penyajian dari suatu algoritma. (Ladjamudin, 2005).

Menurut (Ladjamudin, 2005) ada dua macam *flowchart* yang menggambarkan proses dengan komputer, yaitu:

1. *System Flowchart*

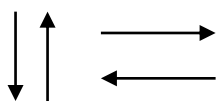

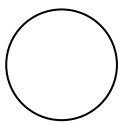
Bagan yang memperlihatkan urutan proses dalam sistem dengan menunjukkan alat media input, output, serta jenis media penyimpanan dalam proses pengolahan data.

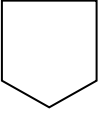


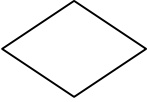


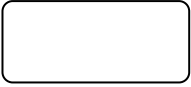
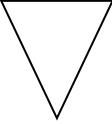
2. *Program Flowchart*


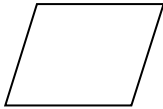
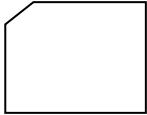
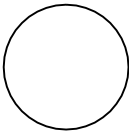



Bagan yang memperlihatkan instruksi yang digambarkan dengan simbol tertentu untuk memecahkan masalah dalam suatu program.

Tabel Elemen Dasar *Flowchart*

Tabel 2.5 Simbol Flowchart

Simbol	Keterangan
	Arus/flow Menyatakan jalannya arus suatu proses
	Communication link Menyatakan bahwa adanya transisi suatu data/informasi dari suatu lokasi ke lokasi lainnya
	Connector Menyatakan sambungan dari satu proses ke proses lainnya dalam halaman/ lembar yang sama

Simbol	Keterangan
	Offline Connector, Menyatakan sambungan dari satu proses ke proses lainnya dalam halaman/ lembar yang berbeda
	Offline Connector, Menyatakan sambungan dari satu proses ke proses lainnya dalam halaman/ lembar yang berbeda
	Manual Menyatakan suatu tindakan (proses) yang tidak dilakukan oleh computer
	Decision/Logika Untuk menunjukkan suatu kondisi tertentu yang akan menghasilkan dua kemungkinan jawaban, ya/tidak
	Predifined Proses Untuk menyatakan penyediaan tempat penyimpanan suatu pengolahan untuk memberi harga awal
	Terminal Untuk menyatakan permulaan atau akhir program
	Keying Operation Untuk menyatakan segala jenis operasi yang diproses dengan menggunakan suatu mesin yang mempunyai keyboard
	Off-Line Storage Untuk menunjukkan bahwa data dalam simbol ini akan disimpan ke suatu media tertentu.

	<p><i>Manual input</i></p> <p>Untuk memasukkan data secara manual dengan menggunakan online keyboard.</p>
	<p><i>Input-Output</i></p> <p>Untuk menyatakan proses <i>input</i> dan outputnya tanpa tergantung dengan jenis peralatannya.</p>
	<p><i>Punched Card</i></p> <p>Untuk menyatakan <i>input</i> berasal dari kartu atau output ditulis ke kartu.</p>
	<p><i>Magnetic-tape unit</i></p> <p>Untuk menyatakan <i>input</i> berasal dari pita magnetic atau output disimpan ke pita magnetic</p>
	<p><i>Disk Storage</i></p> <p>Untuk menyatakan <i>input</i> berasal dari disk atau output disimpan ke disk</p>
	<p><i>Document</i></p> <p>Untuk mencetak laporan ke pr inter</p>
	<p><i>Display</i></p> <p>Untuk menyatakan peralatan output yang digunakan berupa layar (video, komputer)</p>

(Sumber : Ladjamudin, 2005)

2.7.5 Kamus Data

Kamus data berfungsi membantu pelaku sistem untuk mengartikan aplikasi secara detail dan mengorganisasi semua elemen data yang digunakan dalam sistem secara persis sehingga pemakai dan penganalisa sistem mempunyai dasar pengertian yang

sama tentang masukan, keluaran, penyimpanan dan proses (Ladjamudin, 2005).

2.8 PHP

PHP adalah singkatan dari *Hypertext Preprocessor*. *PHP* mengizinkan pengembang untuk menempelkan kode di dalam *Hypertext Markup Language (HTML)* dengan menggunakan Bahasa yang sama seperti *Perl* dan *UNIX shells*. Objek tersusun sebagai halaman *HTML*, tetapi dengan generasi konten dinamis yang *programmatic* (Simarmata, 2009).

2.8.1 Sejarah PHP

Pada awalnya PHP merupakan kependekan dari *Personal Home Page* (Situs personal). PHP pertama kali dibuat oleh Rasmus Lerdorf pada tahun 1995. Pada waktu itu PHP masih bernama *Form Interpreted (FI)*, yang wujudnya berupa sekumpulan skrip yang digunakan untuk mengolah data formulir dari web.

Selanjutnya Rasmus merilis kode sumber tersebut untuk umum dan menamakannya PHP/FI. Dengan perilisan kode sumber ini menjadi sumber terbuka, maka banyak pemrogram yang tertarik untuk ikut mengembangkan PHP.

Pada November 1997, dirilis PHP/FI 2.0. Pada rilis ini, *interpreter* PHP sudah diimplementasikan dalam program C. Dalam rilis ini disertakan juga modul-modul ekstensi yang meningkatkan kemampuan PHP/FI secara signifikan.

Pada tahun 1997, sebuah perusahaan bernama Zend menulis ulang *interpreter* PHP menjadi lebih bersih, lebih baik, dan lebih

cepat. Kemudian pada Juni 1998, perusahaan tersebut merilis interpreter baru untuk PHP dan meresmikan rilis tersebut sebagai PHP 3.0 dan singkatan PHP diubah menjadi akronim berulang *PHP: Hypertext Preprocessing*.

Pada pertengahan tahun 1999, Zend merilis interpreter PHP baru dan rilis tersebut dikenal dengan PHP 4.0. PHP 4.0 adalah versi PHP yang paling banyak dipakai pada awal abad ke-21. Versi ini banyak dipakai disebabkan kemampuannya untuk membangun aplikasi web kompleks tetapi tetap memiliki kecepatan dan stabilitas yang tinggi.

Pada Juni 2004, Zend merilis PHP 5.0. Dalam versi ini, inti dari interpreter PHP mengalami perubahan besar. Versi ini juga memasukkan model pemrograman berorientasi objek ke dalam PHP untuk menjawab perkembangan bahasa pemrograman ke arah paradigma berorientasi objek. Server web bawaan ditambahkan pada versi 5.4 untuk mempermudah pengembang menjalankan kode PHP tanpa menginstall software server.

Versi terbaru dan stabil dari bahasa pemrograman PHP saat ini adalah versi 7.0.16 dan 7.1.2 yang resmi dirilis pada tanggal 17 Februari 2017.

2.8.2 Sintaksis Dasar

Pembatas

PHP hanya mengeksekusi kode yang ditulis dalam pembatas sebagaimana ditentukan oleh dasar sintaks PHP. Apapun di luar pembatas tidak diproses oleh PHP (meskipun teks PHP ini masih

mengendalikan struktur yang dijelaskan dalam kode PHP. Pembatas yang paling umum adalah "<?php" untuk membuka dan ">" Untuk menutup kode PHP. Tujuan dari pembatas ini adalah untuk memisahkan kode PHP dari kode di luar PHP, seperti HTML, Javascript.

Variabel

Variabel diawali dengan simbol dolar \$. Pada versi php PHP 5 diperkenalkan jenis isyarat yang memungkinkan fungsi untuk memaksa mereka menjadi parameter objek dari class tertentu, array, atau fungsi. Namun, jenis petunjuk tidak dapat digunakan dengan jenis skalar seperti angka atau string. Contoh variabel dapat ditulis sebagai \$nama_variabel.

Penulisan fungsi, penamaan kelas, nama variabel adalah peka akan huruf besar (Kapital) dan huruf kecil. Kedua kutip ganda "" dari string memberikan kemampuan untuk interpolasi nilai variabel ke dalam string PHP. PHP menerjemahkan baris sebagai spasi, dan pernyataan harus diakhiri dengan titik koma ;.

Komentar

PHP memiliki 3 jenis sintaks sebagai komentar pada kode yaitu tanda blok /* */, komentar 2 baris // Serta tanda pagar # digunakan untuk komentar satu baris. Komentar bertujuan untuk meninggalkan catatan pada kode PHP dan tidak akan diterjemahkan ke program.

Fungsi

Ratusan fungsi yang disediakan oleh PHP serta ribuan lainnya yang tersedia melalui berbagai ekstensi tambahan. fungsi-fungsi ini didokumentasikan dalam dokumentasi PHP. Namun, dalam berbagai tingkat pengembangan, kini memiliki berbagai konvensi penamaan. Sintaks fungsi adalah seperti di bawah ini:

```
function    tampilkan($data="")    // Mendefenisikan fungsi,
        "tampilkan" adalah nama sebuah fungsi

//Diapit oleh tanda kurung kurawal
        if($data) return $data;    else return 'Tidak ada data';
// Melakukan proses pengolahan data, contohnya melalui
kondisi
echo tampilkan("isi halaman") // Menjalankan fungsi
```

2.8.3 Membuat Program

Membuat sebuah halaman web

Sebuah Halaman web yang ditulis menggunakan Bahasa Pemrograman PHP adalah sebagai berikut:

```
<?php

        echo "Halo dunia";

?>
```

Program bilangan Fibonacci

Berikut ini adalah contoh program yang relatif lebih kompleks yang ditulis dengan menggunakan PHP. Contoh program ini adalah program untuk menampilkan 20 bilangan pertama dari deret bilangan Fibonacci. terdapat beberapa variable atau sintax. seperti function. itu merupakan bagian dari javascript.

```
<?php

function fibonacci_seq( $panjang ) {
    for( $l = array(0,1), $i = 2, $x = 0; $i <
$panjang; $i++ ) {
        $l[] = $l[$x++] + $l[$x];
    }
    return $l;
}

fibonacci_seq(20);

// Angka "20" dapat diganti sesuai keinginan

?>
```

2.8.4 Kelebihan PHP dari pemrograman lain

Beberapa kelebihan PHP dari bahasa pemrograman web, antara lain:

1. Bahasa pemrograman PHP adalah sebuah bahasa script yang tidak melakukan sebuah kompilasi dalam penggunaannya.

2. Web Server yang mendukung PHP dapat ditemukan di mana - mana dari mulai apache, IIS, Lighttpd, hingga Xitami dengan konfigurasi yang relatif mudah.
3. Dalam sisi pengembangan lebih mudah, karena banyaknya milis - milis dan developer yang siap membantu dalam pengembangan.
4. Dalam sisi pemahaman, PHP adalah bahasa scripting yang paling mudah karena memiliki referensi yang banyak.
5. PHP adalah bahasa open source yang dapat digunakan di berbagai mesin (Linux, Unix, Macintosh, Windows) dan dapat dijalankan secara runtime melalui console serta juga dapat menjalankan perintah-perintah system.

Tipe data

PHP memiliki 8 tipe data, yaitu:

1. *Boolean*
2. *Integer*
3. *Float/ Double*
4. *String*
5. *Array*
6. *Object*
7. *Resource*
8. *NULL*

2.9 CodeIgniter

CodeIgniter merupakan aplikasi sumber terbuka yang berupa framework PHP dengan model MVC (Model, View, Controller) untuk membangun website dinamis dengan menggunakan PHP. CodeIgniter memudahkan developer untuk membuat aplikasi web dengan cepat mudah dibandingkan dengan membuatnya dari awal. CodeIgniter dirilis pertama kali pada 28 Februari 2006. Versi stabil terakhir adalah versi 3.0.4

2.9.1 Framework

Framework secara sederhana dapat diartikan kumpulan dari fungsi-fungsi/prosedur-prosedur dan class-class untuk tujuan tertentu yang sudah siap digunakan sehingga bisa lebih mempermudah dan mempercepat pekerjaan seorang programmer, tanpa harus membuat fungsi atau class dari awal.

Ada beberapa alasan mengapa menggunakan Framework:

- Mempercepat dan mempermudah pembangunan sebuah aplikasi web.
- Relatif memudahkan dalam proses maintenance karena sudah ada pola tertentu dalam sebuah framework (dengan syarat programmer mengikuti pola standar yang ada)
- Umumnya framework menyediakan fasilitas-fasilitas yang umum dipakai sehingga kita tidak perlu membangun dari awal (misalnya validasi, ORM, pagination, multiple database, scaffolding, pengaturan session, error handling, dll)
- Lebih bebas dalam pengembangan jika dibandingkan CMS

2.9.2 Desain Patern MVC (Model, View, Controller)

Model View Controller merupakan suatu konsep yang cukup populer dalam pembangunan aplikasi web, berawal pada bahasa pemrograman Small Talk, MVC memisahkan pengembangan aplikasi berdasarkan komponen utama yang membangun sebuah aplikasi seperti manipulasi data, user interface, dan bagian yang menjadi kontrol aplikasi. Terdapat 3 jenis komponen yang membangun suatu MVC pattern dalam suatu aplikasi yaitu :

1. View, merupakan bagian yang menangani presentation logic. Pada suatu aplikasi web bagian ini biasanya berupa file template HTML, yang diatur oleh controller. View berfungsi untuk menerima dan merepresentasikan data kepada user. Bagian ini tidak memiliki akses langsung terhadap bagian model.
2. Model, biasanya berhubungan langsung dengan database untuk memanipulasi data (insert, update, delete, search), menangani validasi dari bagian controller, namun tidak dapat berhubungan langsung dengan bagian view.
3. Controller, merupakan bagian yang mengatur hubungan antara bagian model dan bagian view, controller berfungsi untuk menerima request dan data dari user kemudian menentukan apa yang akan diproses oleh aplikasi.

Dengan menggunakan prinsip MVC suatu aplikasi dapat dikembangkan sesuai dengan kemampuan developernya, yaitu programmer yang menangani bagian model dan controller,

sedangkan designer yang menangani bagian view, sehingga penggunaan arsitektur MVC dapat meningkatkan maintainability dan organisasi kode. Walaupun demikian dibutuhkan komunikasi yang baik antara programmer dan designer dalam menangani variabel-variabel yang akan ditampilkan.

Ada beberapa kelebihan CodeIgniter (CI) dibandingkan dengan Framework PHP lain,

- Performa sangat cepat: salah satu alasan tidak menggunakan framework adalah karena eksekusinya yang lebih lambat daripada PHP from the scratch, tapi CodeIgniter sangat cepat bahkan mungkin bisa dibilang codeigniter merupakan framework yang paling cepat dibanding framework yang lain.
- Konfigurasi yang sangat minim (*nearly zero configuration*): tentu saja untuk menyesuaikan dengan database dan keleluasaan routing tetap diizinkan melakukan konfigurasi dengan mengubah beberapa file konfigurasi seperti database.php atau autoload.php, namun untuk menggunakan codeigniter dengan setting standard, anda hanya perlu mengubah sedikit saja file pada folder config.
- Banyak komunitas: dengan banyaknya komunitas CI ini, memudahkan kita untuk berinteraksi dengan yang lain, baik itu bertanya atau teknologi terbaru.
- Dokumentasi yang sangat lengkap: Setiap paket instalasi codeigniter sudah disertai user guide yang sangat bagus dan

lengkap untuk dijadikan permulaan, bahasanya pun mudah dipahami.

- Dan banyak lagi yang lainnya.

Versi

Tabel 2.6 Versi CodeIgniter

Version	Release date	PHP version	Notes
Beta 1.0	28 Februari 2006		
Beta 1.1	10 Maret 2006		
1.2	21 Maret 2006		
1.3	03 April 2006		
1.3.1	11 April 2006		
1.3.2	17 April 2006		
1.3.3	01 Juni 2006		
1.4.0	17 September 2006	>=4.4.1	
1.4.1	21 September 2006		
1.5.0	30 Oktober 2006		
1.5.0.1	31 Oktober 2006		
1.5.1	23 November 2006		
1.5.2	13 Februari 2007		
1.5.3	15 April 2007		
1.5.4	12 Juli 2007		

Version	Release date	PHP version	Notes
1.6.0	30 Januari 2008		
1.6.1	12 Februari 2008		
1.6.2	13 Mei 2008		
1.6.3	26 Juni 2008		
1.7	23 Oktober 2008		
1.7.1	10 Februari 2009		
1.7.2	11 September 2009		
2.0	28 Januari 2011	>=5.1.6	
2.0.1	15 Maret 2011		
2.0.2	07 April 2011		
2.0.3	20 Agustus 2011		
2.1.0	14 November 2011		
2.1.1	12 Juni 2012		
2.1.2	29 Juni 2012		
2.1.3	08 Oktober 2012		
2.1.4	08 Juli 2013		
2.2.0	2 Juni 2014		
2.2.1	22 Januari 2015		CodeIgniter 2.2.1 Released ^[2]
3.0.0	30 Maret 2015	>=5.2.4	CodeIgniter 3.0 ^[3]
2.2.2	15 April 2015		CodeIgniter 2.2.2 Released ^[4]

Version	Release date	PHP version	Notes
2.2.3	14 Juli 2015		CodeIgniter 2.2.3 Released ^[5]
3.0.1	7 Agustus 2015		
2.2.4	20 Agustus 2015		CodeIgniter 2.2.4 Released ^[6]
3.0.2	08 Oktober 2015		CodeIgniter 3.0.2 Released ^[7]
2.2.5	08 Oktober 2015		CodeIgniter 2.2.5 Released ^[8]
3.0.3	31 Oktober 2015		CodeIgniter 3.0.3 Released ^[9]
3.0.4	13 Januari 2016		CodeIgniter 3.0.4 Released ^[11]
Legenda: Versi lama Versi lebih lama, tetapi masih didukung Versi terkini Rilis selanjutnya			

2.10 XAMPP

XAMPP ([\[unsupportedinput\]'zæmp/](#) atau [/'ɛks.æmp/^{\[1\]}](#))

adalah perangkat lunak bebas, yang mendukung banyak sistem operasi, merupakan kompilasi dari beberapa program.

Program ini tersedia dalam GNU General Public License dan bebas, merupakan web server yang mudah digunakan yang dapat melayani tampilan halaman web yang dinamis. Untuk mendapatkannya dapat mendownload langsung dari web resminya.

2.10.1 Sejarah dan Pengembang

XAMPP dikembangkan dari sebuah tim proyek bernama Apache Friends, yang terdiri dari Tim Inti (*Core Team*), Tim Pengembang (*Development Team*) & Tim Dukungan (*Support Team*)

2.10.2 Asal Kata dari XAMPP

XAMPP adalah singkatan yang masing-masing hurufnya adalah:

X

X: Program ini dapat dijalankan di banyak sistem operasi, seperti Windows, Linux, Mac OS, dan Solaris.

A

A: Apache, merupakan aplikasi web server. Tugas utama Apache adalah menghasilkan halaman web yang benar kepada user berdasarkan kode PHP yang dituliskan oleh pembuat halaman web. Jika diperlukan juga berdasarkan kode PHP yang dituliskan, maka dapat saja suatu database diakses terlebih dahulu (misalnya dalam MySQL) untuk mendukung halaman web yang dihasilkan.

M

M: MySQL, merupakan aplikasi database server. Perkembangannya disebut SQL yang merupakan kepanjangan dari Structured Query Language. SQL merupakan bahasa terstruktur yang digunakan untuk mengolah database. MySQL dapat digunakan untuk membuat dan mengelola database beserta isinya.

Kita dapat memanfaatkan MySQL untuk menambahkan, mengubah, dan menghapus data yang berada dalam database.

P

P: PHP, bahasa pemrograman web. Bahasa pemrograman PHP merupakan bahasa pemrograman untuk membuat web yang bersifat *server-side scripting*. PHP memungkinkan kita untuk membuat halaman web yang bersifat dinamis. Sistem manajemen basis data yang sering digunakan bersama PHP dan MySQL. Namun PHP juga mendukung sistem manajemen database Oracle, Microsoft Access, Interbase, d-base, PostgreSQL, dan sebagainya.

P

P: Perl, bahasa pemrograman untuk segala keperluan, dikembangkan pertama kali oleh Larry Wall di mesin Unix. Perl dirilis pertama kali pada tanggal 18 Desember 1987 ditandai dengan keluarnya Perl 1. Pada versi-versi selanjutnya, Perl tersedia pula untuk berbagai sistem operasi varian Unix (SunOS, Linux, BSD, HP-UX), juga tersedia untuk sistem operasi seperti DOS, Windows, Power PC, BeOS, VMS, EBCDIC, dan PocketPC.

2.10.3 Bagian-Bagian penting xampp

Mengenal bagian **XAMPP** yang biasa digunakan pada umumnya:

- **htdocs** adalah folder tempat meletakkan berkas-berkas yang akan dijalankan, seperti berkas PHP, HTML dan skrip lain.

- **phpMyAdmin** merupakan bagian untuk mengelola basis data MySQL yang ada dikomputer. Untuk membukanya, buka browser lalu ketikkan alamat `http://localhost/phpMyAdmin`, maka akan muncul halaman phpMyAdmin.
- **Kontrol Panel** yang berfungsi untuk mengelola layanan (*service*) XAMPP. Seperti menghentikan (*stop*) layanan, ataupun memulai (*start*).

2.10.4 Komponen

XAMPP 1.8.3 untuk Windows :

- Apache 2.4.4
- MySQL 5.6.11
- PHP 5.5.0
- phpMyAdmin 4.0.4
- FileZilla FTP Server 0.9.41
- Tomcat 7.0.41 (with mod_proxy_ajp as connector)
- Strawberry Perl 5.16.3.1 Portable
- XAMPP Control Panel 3.2.1 (from hackattack142)

XAMPP 1.8.3 untuk Linux, including:

- Apache 2.4.4
- MySQL 5.6.11
- PHP 5.5.0
- phpMyAdmin 4.0.4
- OpenSSL 1.0.1e

XAMPP untuk Solaris

XAMPP untuk Mac OS X

2.11 MySQL

MySQL adalah sebuah perangkat lunak sistem manajemen basis data SQL (bahasa Inggris: *database management system*) atau DBMS yang multialur, multipengguna, dengan sekitar 6 juta instalasi di seluruh dunia. MySQL AB membuat MySQL tersedia sebagai perangkat lunak gratis di bawah lisensi GNU General Public License (GPL), tetapi mereka juga menjual di bawah lisensi komersial untuk kasus-kasus di mana penggunaannya tidak cocok dengan penggunaan GPL.

Tidak sama dengan proyek-proyek seperti Apache, di mana perangkat lunak dikembangkan oleh komunitas umum, dan hak cipta untuk kode sumber dimiliki oleh penulisnya masing-masing, MySQL dimiliki dan disponsori oleh sebuah perusahaan komersial Swedia MySQL AB, di mana memegang hak cipta hampir atas semua kode sumbernya. Kedua orang Swedia dan satu orang Finlandia yang mendirikan MySQL AB adalah: David Axmark, Allan Larsson, dan Michael "Monty" Widenius.

2.11.1 Sistem manajemen basis data relasional

MySQL adalah sebuah implementasi dari sistem manajemen basis data relasional (RDBMS) yang didistribusikan secara gratis di bawah lisensi GPL (General Public License). Setiap pengguna dapat secara bebas menggunakan MySQL, namun dengan batasan perangkat lunak tersebut tidak boleh dijadikan produk turunan yang bersifat komersial. MySQL sebenarnya merupakan

turunan salah satu konsep utama dalam basisdata yang telah ada sebelumnya; SQL (Structured Query Language). SQL adalah sebuah konsep pengoperasian basisdata, terutama untuk pemilihan atau seleksi dan pemasukan data, yang memungkinkan pengoperasian data dikerjakan dengan mudah secara otomatis.

Kehandalan suatu sistem basisdata (DBMS) dapat diketahui dari cara kerja pengoptimasi-nya dalam melakukan proses perintah-perintah SQL yang dibuat oleh pengguna maupun program-program aplikasi yang memanfaatkannya. Sebagai peladen basis data, MySQL mendukung operasi basisdata transaksional maupun operasi basisdata non-transaksional. Pada modus operasi non-transaksional, MySQL dapat dikatakan unggul dalam hal unjuk kerja dibandingkan perangkat lunak peladen basisdata kompetitor lainnya. Namun pada modus non-transaksional tidak ada jaminan atas reliabilitas terhadap data yang tersimpan, karenanya modus non-transaksional hanya cocok untuk jenis aplikasi yang tidak membutuhkan reliabilitas data seperti aplikasi blogging berbasis web (wordpress), CMS, dan sejenisnya. Untuk kebutuhan sistem yang ditujukan untuk bisnis sangat disarankan untuk menggunakan modus basisdata transaksional, hanya saja sebagai konsekuensinya unjuk kerja MySQL pada modus transaksional tidak secepat unjuk kerja pada modus.

2.11.2 Sejarah MySQL

MySQL pada awalnya diciptakan pada tahun 1979, oleh Michael "Monty" Widenius, seorang programmer komputer asal Swedia. Monty mengembangkan sebuah sistem database sederhana yang dinamakan UNIREG yang menggunakan koneksi low-level ISAM database engine dengan indexing. Pada saat itu Monty bekerja pada perusahaan bernama TcX di Swedia.

TcX pada tahun 1994 mulai mengembangkan aplikasi berbasis web, dan berencana menggunakan UNIREG sebagai sistem database. Namun sayangnya, UNIREG dianggap tidak cocok untuk database yang dinamis seperti web.

TcX kemudian mencoba mencari alternatif sistem database lainnya, salah satunya adalah mSQL (miniSQL). Namun mSQL versi 1 ini juga memiliki kekurangan, yaitu tidak mendukung indexing, sehingga performanya tidak terlalu bagus.

Dengan tujuan memperbaiki performa mSQL, Monty mencoba menghubungi David Hughes (programmer yang mengembangkan mSQL) untuk menanyakan apakah ia tertarik mengembangkan sebuah konektor di mSQL yang dapat dihubungkan dengan UNIREG ISAM sehingga mendukung indexing. Namun saat itu Hughes menolak, dengan alasan sedang mengembangkan teknologi indexing yang independen untuk mSQL versi 2.

Dikarenakan penolakan tersebut, David Hughes, TcX (dan juga Monty) akhirnya memutuskan untuk merancang dan mengembangkan sendiri konsep sistem database baru. Sistem ini

merupakan gabungan dari UNIREG dan mSQL (yang source codenya dapat bebas digunakan). Sehingga pada May 1995, sebuah RDBMS baru, yang dinamakan MySQL dirilis.

David Axmark dari Detron HB, rekanan TcX mengusulkan agar MySQL di 'jual' dengan model bisnis baru. Ia mengusulkan agar MySQL dikembangkan dan dirilis dengan gratis. Pendapatan perusahaan selanjutnya di dapat dari menjual jasa "support" untuk perusahaan yang ingin mengimplementasikan MySQL. Konsep bisnis ini sekarang dikenal dengan istilah Open Source.

Pada tahun 1995 itu juga, TcX berubah nama menjadi MySQL AB, dengan Michael Widenius, David Axmark dan Allan Larsson sebagai pendirinya. Titel "AB" di belakang MySQL, adalah singkatan dari "Aktiebolag", istilah PT (Perseroan Terbatas) bagi perusahaan Swedia.

2.11.3 Keistimewaan MySQL

MySQL memiliki beberapa keistimewaan, antara lain :

1. **Portabilitas.** MySQL dapat berjalan stabil pada berbagai sistem operasi seperti Windows, Linux, FreeBSD, Mac Os X Server, Solaris, Amiga, dan masih banyak lagi.
2. **Perangkat lunak sumber terbuka.** MySQL didistribusikan sebagai perangkat lunak sumber terbuka, di bawah lisensi GPL sehingga dapat digunakan secara gratis.

3. **Multi-user.** MySQL dapat digunakan oleh beberapa pengguna dalam waktu yang bersamaan tanpa mengalami masalah atau konflik.
4. **'Performance tuning',** MySQL memiliki kecepatan yang menakjubkan dalam menangani query sederhana, dengan kata lain dapat memproses lebih banyak SQL per satuan waktu.
5. **Ragam tipe data.** MySQL memiliki ragam tipe data yang sangat kaya, seperti signed / unsigned integer, float, double, char, text, date, timestamp, dan lain-lain.
6. **Perintah dan Fungsi.** MySQL memiliki operator dan fungsi secara penuh yang mendukung perintah Select dan Where dalam perintah (*query*).
7. **Keamanan.** MySQL memiliki beberapa lapisan keamanan seperti level subnetmask, nama host, dan izin akses *user* dengan sistem perizinan yang mendetail serta sandi terenkripsi.
8. **Skalabilitas dan Pembatasan.** MySQL mampu menangani basis data dalam skala besar, dengan jumlah rekaman (records) lebih dari 50 juta dan 60 ribu tabel serta 5 miliar baris. Selain itu batas indeks yang dapat ditampung mencapai 32 indeks pada tiap tabelnya.
9. **Konektivitas.** MySQL dapat melakukan koneksi dengan klien menggunakan protokol TCP/IP, Unix soket (UNIX), atau Named Pipes (NT).
10. **Lokalisasi.** MySQL dapat mendeteksi pesan kesalahan pada klien dengan menggunakan lebih dari dua puluh bahasa. Meski pun demikian, bahasa Indonesia belum termasuk di dalamnya.

11. **Antar Muka.** MySQL memiliki antar muka (interface) terhadap berbagai aplikasi dan bahasa pemrograman dengan menggunakan fungsi API (Application Programming Interface).
12. **Klien dan Peralatan.** MySQL dilengkapi dengan berbagai peralatan (tool) yang dapat digunakan untuk administrasi basis data, dan pada setiap peralatan yang ada disertakan petunjuk online.
13. **Struktur tabel.** MySQL memiliki struktur tabel yang lebih fleksibel dalam menangani ALTER TABLE, dibandingkan basis data lainnya semacam PostgreSQL ataupun Oracle.

2.11.4 Kelebihan dan Kekurangan MySQL dibanding RDBMS lain

Hal paling mendasar yang menjadikan MySQL pilihan utama sebagai database yang digunakan adalah karena MySQL menggunakan Lisensi GPL dan multiplatform, sehingga lebih disukai para mahasiswa karena tidak membutuhkan biaya besar dalam membuat aplikasi serta tidak harus tergantung pada OS Windows ataupun Linux karena dapat dijalankan pada kedua OS tersebut dan beberapa OS lainnya. Tapi alasan tersebut tidaklah cukup untuk menjadikan MySQL sebagai RDBMS yang akan digunakan. Berikut keunggulan lain yang diberikan

Kelebihan

1. Berlisensi GPL dan Multi Platform.
2. Dapat diintegrasikan dengan beberapa bahasa Pemrograman seperti .Net, Java, Python, Perl yang merupakan bahasa pemrograman yang paling dominan di kalangan programmer.

3. Mendukung ODBC untuk sistem operasi Windows sehingga bisa digunakan aplikasi yang berjalan di windows.
4. Bisa dijalankan pada spesifikasi hardware yang rendah karena lebih hemat resource memory (dibandingkan database lain) sehingga mudah digunakan untuk bahan pembelajaran.
5. MySQL dapat mendeteksi pesan kesalahan pada klien dengan menggunakan lebih dari 20 bahasa meskipun bahasa indonesia belum termasuk didalamnya.
6. MySQL dapat diintegrasikan dengan Hosting.

Kekurangan

1. Banyak mengklaim kurang support terhadap pemrograman Visual/Desktop, sehingga sedikit yang menggunakan untuk aplikasi visual.
2. Karena berlisensi GPL sehingga sulit mendapatkan update untuk *problem* yang *urgent*, sehingga perusahaan skala menengah keatas lebih memilih RDBMS berlisensi dan disupport seperti Oracle dan MS SQL Server
3. Sangat diragukan dalam menangani data skala besar, karena ada beberapa opini yang pro dan kontra terhadap kemampuan MySQL terhadap pengolahan data yang besar.

2.11.5 Kemampuan MySQL

Berikut Fitur serta kapabilitas yang dimiliki oleh MySQL:

1. Unjuk kerja yang tinggi dalam memproses query sederhana, dalam arti dapat memproses lebih banyak SQL per satuan waktu.
2. Memiliki lebih banyak tipe data seperti : signed/unsigned integer yang memiliki panjang data sebesar 1,2,3,4 dan 8 byte, FLOAT, DOUBLE, CHAR, VARCHAR, TEXT, BLOB, DATE, TIME, DATETIME, TIMESTAMP, YEAR, SET dan tipe ENUM.
3. Mendukung field yang dijadikan Index, dengan maksimal **32 index** dalam satu tabel. *
4. MYSQL memiliki beberapa **lapisan keamanan**, seperti *subnetmask*, nama *host*, dan izin akses user dengan sistem perijinan yang mendetail serta sandi/password terenkripsi.
5. **Konektivitas** , MySQL dapat melakukan koneksi dengan klien menggunakan protokol TCP/IP ,Unix soket (UNIX),atau Named Pipes(NT).
6. **Multi-user**. MySQL dapat digunakan oleh beberapa pengguna dalam waktu yang bersamaan tanpa mengalami masalah atau konflik
7. *Command and function*, MySQL memiliki fungsi dan operator secara penuh yang mendukung perintah *select* dan *where* dalam *query*.
8. *Structure Table*, MySQL memiliki struktur tabel yang lebih fleksibel dalam menangani *ALTER TABLE* dibandingkan DBMS lainnya.
9. Mendukung penuh terhadap kalimat SQL GROUP BY dan ORDER BY. Mendukung terhadap fungsi penuh (COUNT(),COUNT(), DISTINCT() AVG(), STD(), SUM(), MAX() dan MIN())

2.11.6 Bahasa Pemrograman

Terdapat beberapa API (*Application Programming Interface*) tersedia yang memungkinkan aplikasi-aplikasi komputer yang ditulis dalam berbagai bahasa pemrograman untuk dapat mengakses basis data MySQL antara lain: bahasa pemrograman C, C++, C#, bahasa pemrograman Eiffel, bahasa pemrograman Smalltalk, bahasa pemrograman Java, bahasa pemrograman Lisp, Perl, PHP, .. Sebuah antarmuka ODBC memanggil MyODBC yang memungkinkan setiap bahasa pemrograman yang mendukung ODBC untuk berkomunikasi dengan basis data MySQL. Kebanyakan kode sumber MySQL dalam ANSI C.

Penggunaan

Mysql sangat populer di aplikasi web MediaWiki (perangkat lunak yang dipakai Wikipedia dan proyek-proyek sejenis) dan PHP-Nuke dan berfungsi sebagai komponen basis data dalam LAMP. Popularitas sebagai aplikasi web dikarenakan kedekatannya dengan popularitas PHP, sehingga seringkali disebut sebagai *Dynamic Duo*.

2.11.7 Administrasi

Untuk melakukan administrasi dalam basis data MySQL, dapat menggunakan modul yang sudah termasuk yaitu *command-line* (perintah: `mysql` dan `mysqladmin`). Juga dapat diunduh dari situs MySQL yaitu sebuah modul berbasis grafik (*GUI*): *MySQL Administrator* dan *MySQL Query Browser*. Selain itu terdapat juga sebuah perangkat lunak gratis untuk administrasi basis data MySQL

berbasis web yang sangat populer yaitu phpMyAdmin. Untuk perangkat lunak untuk administrasi basis data MySQL yang dijual secara komersial antara lain: MySQL front, Navicat dan EMS SQL Manager for MySQL. Untuk pengelolaan website di hosting, database MySQL dapat dikelola melalui phpMyAdmin.

2.11.8 Akusisi oleh Sun dan Oracle

Pada tanggal 16 Januari 2008 Sun Microsystems, Inc mengumumkan aksi korporasi - akuisisi terhadap MySQL AB sehingga menjadikan Sun sebagai salah satu perusahaan dengan produk platform open source terbesar seperti Java, Open Solaris dan akhirnya MySQL.

Berselang setahun kemudian, tepatnya pada tanggal 20 April 2009 giliran Oracle melakukan akuisisi terhadap Sun Microsystems. Sejak saat itu berkembang isu Oracle - yang memiliki produk database yang berkompetisi dengan MySQL - akan mematikan MySQL. Namun sampai sejauh ini hal tersebut belum terbukti.

2.11.9 Komunitas MySQL Indonesia

MySQL termasuk salah satu database utama yang digunakan di Indonesia terutama di kalangan developer PHP. Ini disebabkan native integration antara PHP dan MySQL serta banyaknya hosting yang menyertakan kedua produk tersebut karena bebas lisensi dan sudah terbukti tangguh dan efisien.

Website komunitas MySQL Indonesia yang terpisah dari komunitas PHP dapat Anda kunjungi di <http://groups.google.com/group/mysql-indonesia>.

BAB III

ANALISIS DAN PERANCANGAN

3.1 Metodologi

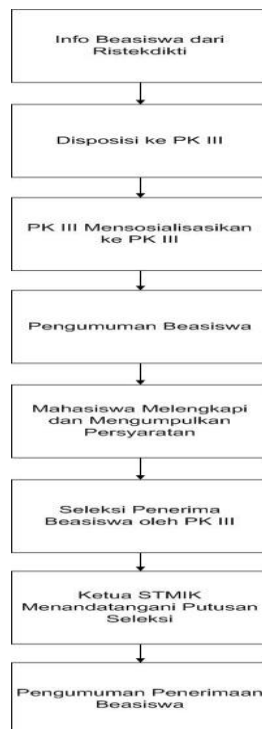
Dalam lima tahap metode waterfall (pressman 2002), penulis hanya menggunakan empat tahapan sesi sebagai berikut :

- a. Analisis kebutuhan perangkat lunak. Proses pengumpulan kebutuhan diintensifkan dan difokuskan, khususnya pada perangkat lunak, untuk memahami sifat program yang dibangun, perekayasa perangkat lunak (analisis) harus memahami domain informasi, tingkah laku, unjuk karya, dan antar muka (interface) yang diperlukan.
- b. Design. Desain perangkat lunak sebenarnya adalah proses multi langkah yang berfokus pada empat atribut sebuah program yang berbeda; struktur data, arsitektur perangkat lunak, representasi perangkat lunak yang dapat diperkirakan demi kualitas sebelum dimulai pemunculan kode.
- c. Generasi kode. Desain harus diterjemahkan kedalam bentuk mesin yang bisa dibaca, langkah pembuatan kode melakukan tugas ini jika desain dilakukan dengan cara yang lengkap, pembuatan kode dapat dilakukan secara mekanis.
- d. Test atau Pengujian. Sekali kode dibuat pengujian program dimulai. Proses pengujian berfokus pada logika internal perangkat lunak, memastikan bahwa semua pernyataan sudah diuji, dan pada eksternal fungsional yaitu mengarahkan pengujian untuk menemukan kesalahan-

kesalahan dan memastikan bahwa input dibatasi akan memberikan hasil aktual yang sesuai dengan hasil dibutuhkan.

3.2 Tinjauan Beasiswa PPA

3.2.1 Alur Penerimaan Beasiswa PPA



Gambar 3.1 Alur Penerimaan Beasiswa

3.2.2 Tujuan Beasiswa PPA

Adapun tujuan dari beasiswa PPA adalah Meningkatkan motivasi belajar mahasiswa, meningkatkan prestasi mahasiswa baik dalam akademik/kurikuler, ko-kurikuler maupun ekstrakurikuler, menjamin penyelesaian studi mahasiswa tepat waktu.

3.2.3 Manfaat Beasiswa PPA

Adapun Manfaat dari beasiswa PPA antara lain:

- a. Meningkatkan rasa semangat untuk terus belajar dan mengasah keahliannya.
- b. Memacu intensitas untuk belajar
- c. Memberikan rasa bangga tersendiri bagi seseorang
- d. Sebagai pemacu untuk kearah yang lebih baik.
- e. Lebih mengerti arti sebuah nilai materi.

3.3 Pembentukan Variabel dan Himpunan Fuzzy

3.3.1 Data Masukan (Input)

Jumlah variabel input yang digunakan sebanyak 4 variabel dimana setiap variabel mengandung 3 himpunan fuzzy sehingga total himpunan fuzzy yang digunakan adalah 12 himpunan. Dalam perancangan sistem fuzzy ini, menggunakan kurva bahu untuk himpunan rendah dan tinggi, sedangkan untuk himpunan sedang menggunakan kurva berbentuk segitiga.

Tabel 3.1 Tabel Inputan

No	Variabel	Himpunan	Parameter	Domain	
1	IPK	Rendah	0.5 - 2	(0-4)	Kurva Bahu
		Sedang	1.5 – 2.79		Segitiga
		Tinggi	2.5 - 4		Kurva Bahu
2	Penghasilan Orang Tua	Rendah	750000-1500000	(750000 - 4500000)	Kurva Bahu
		Sedang	1000000-3000000		Segitiga
		Tinggi	2500000-4500000		Kurva Bahu
3	Jumlah Tanggungan Orang Tua	Rendah	1-3	(1-7)	Kurva Bahu
		Sedang	2-5		Segitiga
		Tinggi	4-7		Kurva Bahu
4	Jumlah Sertifikat	Rendah	1-3	(1-7)	Kurva Bahu
		Sedang	2-5		Segitiga

		Tinggi	4-7		Kurva Bahu
--	--	--------	-----	--	------------

3.3.1 Data Keluaran (Output)

Output yang dihasilkan oleh sistem fuzzy pemilihan beasiswa ini berupa hasil rekomendasi mahasiswa yang paling cocok untuk mendapatkan beasiswa yang disertai bobot.

3.3.3 kebutuhan Perangkat Lunak (Software)

Analisis kebutuhan perangkat lunak adalah kebutuhan yang digunakan untuk membuat aplikasi penerapan *Logika Fuzzy Tahani* pada penerimaan beasiswa Peningkatan Prestasi Akademik(PPA) yaitu sebagai berikut :

1. Microsoft Windows 10 atau Windows digunakan sebagai sistem operasi laptop atau computer.
2. UC Browser atau google chrome untuk menjalankan sistem.
3. Xampp 5.5.15 (*Mysql*) digunakan sebagai database sistem yang berfungsi untuk menampung field dan record yang diinputkan disistem. Sehingga suatu saat data yang telah diinputkan tersebut dapat dicari dan diambil oleh sistem serta diakses.
4. Sublime Text atau Netbeans IDE 8.1 untuk *editor Source code* program.

3.3.4 Perangkat Keras (Hardware)

Perangkat keras yang dibutuhkan dalam membangun sistem penSeleksian beasiswa PPA ini, diperlukan beberapa *hardware* atau perangkat keras. Adapun perangkat keras yang dibutuhkan yaitu:

1. Minimal intel (R) Celeron (R) CPU 887 @ 1.50 GHz
2. Memory minimal 2 GB DDR2
3. Hardisk minimal 320 GB
4. LCD 10.1 WLED
5. Keyboard
6. Mouse

3.3.5 Sumber Daya Manusia

Pengguna sistem ini adalah bagian kemahasiswaan STMIK Bumigora Mataram dan untuk mengoptimalkan penggunaan aplikasi, maka akan dilaksanakan training penggunaan / SDM.

3.4 Desain Sistem

Desain sistem merupakan pembuatan rancangan sistem fuzzy yang berkaitan dengan fungsionalitas. Berikut adalah beberapa hal yang dilakukan untuk membangun sistem fuzzy.

3.4.1 Struktur Database

Aplikasi beasiswa yang penulis buat ini menggunakan 4 tabel yaitu tbl_mahasiswa, tbl_ppa, tbl_bobot_ppa, tbl_login. Adapun struktur dari masing-masing tabel adalah sebagai berikut

1. Tabel Mahasiswa

Tabel ini digunakan untuk menyimpan data-data mahasiswa yang akan diseleksi untuk mendapatkan beasiswa PPA. Adapun field-field yang terdapat pada tabel tbl_mahasiswa sebagai berikut:

Tabel 3.2 Tabel Mahasiswa

Field	Type	Size	Key	Keterangan
Npm	Varchar	24	Primary	Npm Mahasiswa
nama_mahasiswa	Varchar	50		Nama Mahasiswa
Jk	Varchar	20		Jenis Kelamin
Ttl	Varchar	70		Tempat Tanggal Lahir
Alamat	Varchar	70		Alamat Mahasiswa
No_hp	Varichar	12		Nomor hp Mahasiswa
Jurusan_prodi	Varchar	40		Jurusan Mahasiswa
Agama	Varchar	15		Agama
Semester	Int	3		Semester
Nama_ayah	Varchar	40		Nama Ayah
Nama_ibu	Varchar	40		Nama Ibu
Ipk	Float			Jumlah IPK
Pekerjaan_ortu	Varchar	50		Pekerjaan Ortu
Penghasilan_ortu	Int	11		Penghasilan Ortu
Jumlah_tanggungan	Int	11		Jumlah Tanggungan Ortu
Jumlah_Sertifikat	Float			Jumlah Sertifikat

2. Tabel PPA

Tabel ini digunakan untuk menyimpan hasil query beasiswa PPA dari masing-masing kriteria. Adapun field-field dalam tabel tbl_ppa ini adalah sebagai berikut:

Tabel 3.3 Tabel PPA

Field	Type	Size	Key	Keterangan
Npm	Varchar	24	Primary	Npm Mahasiswa
Nama	Varchar	70		Nama Mahasiswa
Ipk	Float	0		Skor Ipk Kumulatif
Penghasilan	Float	0		Skor Penghasilan Ortu
Tanggungan	Float	0		Skor Tanggungan Ortu
Sertifikat	Float	0		Skor Jumlah Sertifikat

3. Tabel Bobot

Tabel ini adalah untuk menyimpan bobot dari masing-masing kriteria yang akan digunakan untuk proses perhitungan metode fuzzy database tahani beasiswa PPA. Adapun field yang terkandung pada tabel tbl_Bobotipk, tbl_Bobotpekerjaan, tbl_Bobotpenghasilan, tbl_Bobotjumlah tanggungan, tbl_Bobotjumlah sertifikat ini adalah sebagai berikut :

Tabel 3.4 Tabel Bobotipk

Field	Type	Size	Key	Keterangan
Id	Int	11	Primary	Id bobot mahasiswa
Npm	Varchar	24		Npm Mahasiswa
Rendah	Float	0		Rendah
Sedang	Float	0		Sedang
Tinggi	Float	0		Tinggi

Tabel 3.5 Tabel Bobot Pekerjaan

Field	Type	Size	Key	Keterangan
Id	Int	11	Primary	Id bobot mahasiswa
Npm	Varchar	24		Npm Mahasiswa
Rendah	Float	0		Rendah
Sedang	Float	0		Sedang

Tinggi	Float	0		Tinggi
--------	-------	---	--	--------

Tabel 3.6 Tabel Bobot penghasilan

Field	Type	Size	Key	Keterangan
Id	Int	11	Primary	Id bobot mahasiswa
Npm	Varchar	24		Npm Mahasiswa
Rendah	Float	0		Rendah
Sedang	Float	0		Sedang
Tinggi	Float	0		Tinggi

Tabel 3.7 Tabel Bobot tanggungan

Field	Type	Size	Key	Keterangan
Id	Int	11	Primary	Id bobot mahasiswa
Npm	Varchar	24		Npm Mahasiswa
Rendah	Float	0		Rendah
Sedang	Float	0		Sedang
Tinggi	Float	0		Tinggi

Tabel 3.8 Tabel Bobot sertifikat

Field	Type	Size	Key	Keterangan
Id	Int	11	Primary	Id bobot mahasiswa
Npm	Varchar	24		Npm Mahasiswa
Rendah	Float	0		Rendah
Sedang	Float	0		Sedang
Tinggi	Float	0		Tinggi

4. Tabel User

Tabel ini adalah untuk menyimpan nama Admin dan Operator yang akan login kedalam sistem. Adapun field-field dari tabel tbl_user ini adalah sebagai berikut:

Tabel 3.9 Tabel user

Field	Type	Size	Key	Keterangan
Id_user	Int	40	Primary	ID
Nama	Varchar	255		Nama Pengguna
Username	Varchar	255		Username Admin/Operator
Password	Varchar	255		Password
Jenis_akun	Varchar	255		Jenis Akun

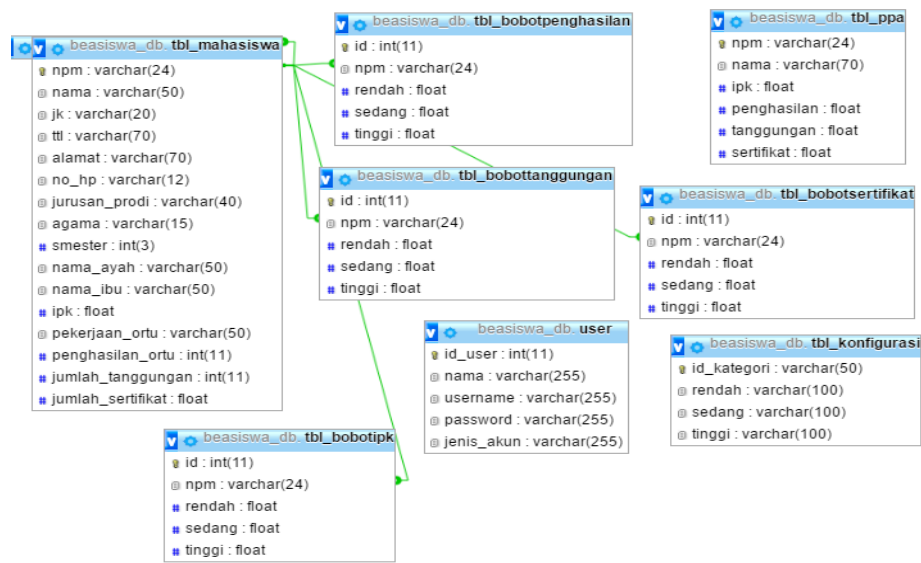
5. Tabel Konfigurasi

Tabel ini adalah untuk menyimpan batas kriteria data yang ada pada semua tabel. Adapun field-field dari tabel tbl_konfigurasi ini adalah sebagai berikut

Tabel 3.10 Tabel Konfigurasi

Field	Type	Size	Key	Keterangan
Id_kategori	Varchar	50	Primary	Id kategori
Rendah	Varchar	100		Rendah
Sedang	Varchar	100		Sedang
Tinggi	Varchar	100		Tinggi

3.4.2 Relasi Database



Gambar 3.2 Relasi Database

3.4.3 Desain Diagram Sistem

Sistem adalah sekumpulan elemen yang saling berkaitan dan saling mempengaruhi dalam hal melakukan kegiatan bersama untuk mencapai suatu tujuan. Salah satu desain sistem yang saya buat ini adalah desain sistem Logika Fuzzy Tahani untuk menentukan penerima beasiswa PPA

yang dimulai dari memasukkan data mahasiswa, menentukan batas-batas kriteria IPK, Pekerjaan Ortu, Penghasilan Ortu, Tanggungan Ortu dan Jumlah Sertifikat, Proses Fuzzifikasi atau perhitungan bobot derajat keanggotaan, Proses query berdasarkan kriteria yang ditetapkan dan terakhir data mahasiswa penerima beasiswa.



Gambar 3.3 Desain sistem

3.5 Rules Fuzzy

Berikut ini rule fuzzy bisa di lihat pada tabel 3.11.

Tabel 3.11 Rule Fuzzy

Ket	Rule
R1	If IPK tinggi and Penghasilan rendah and Tanggungan rendah and Sertifikat tinggi then Ya
R2	If IPK tinggi and Penghasilan rendah and Tanggungan rendah and Sertifikat sedang then Ya
R3	If IPK tinggi and Penghasilan rendah and Tanggungan rendah and Sertifikat rendah then Ya
R4	If IPK tinggi and Penghasilan rendah and Tanggungan tinggi and Sertifikat rendah then Ya
R5	If IPK tinggi and Penghasilan rendah and Tanggungan sedang and Sertifikat rendah then Ya

R6	If IPK tinggi and Penghasilan tinggi and Tanggungan rendah and Sertifikat rendah then Ya
R7	If IPK tinggi and Penghasilan sedang and Tanggungan rendah and Sertifikat rendah then Ya
R8	If IPK tinggi and Penghasilan rendah and Tanggungan rendah and Sertifikat rendah then Ya

3.6 Desain Interface

Desain interface digunakan untuk memudahkan pembangunan sistem penerapan logika fuzzy tahani pada beasiswa PPA, berikut adalah desain antar muka untuk masing-masing proses.

1. Desain Login

Gambar 3.4 Desain Login

2. Desain Beranda

Gambar 3.5 Desain Beranda

3. Desain Input Data Mahasiswa

KLASIFIKASI PENERIMAAN BEASISWA PPA-FUZZY TAHANI

Admin

DATA MAHASISWA CALON PENERIMA BEASISWA

NPM

Semester

Nama

Nama Ayah

Jenis Kelamin

Nama Ibu

Ttl

IPK

Alamat

Penghasilan Ortu

No.Tip

Juml.Tanggungan Ortu

Jurusan/Prodi

Jml.Sertifikat

Agama

Simpan Data

HOME ADMIN

Halo, Admin

DATA DAN KETENTUAN

Data Mahasiswa

Pengaturan Kriteria

FUZZIFIKASI DAN HASIL KLASIFIKASI

Proses Fuzzifikasi

Data Penerima Beasiswa

Kalender

Gambar 3.6 Desain Input Data Mahasiswa

4. Desain Tampil Data Mahasiswa

[illegible]

Gambar 3.7 Desain Tampil Data Mahasiswa

5. Desain Pengaturan Kriteria

KLASIFIKASI PENERIMAAN BEASISWA PPA-FUZZY TAHANI

Admin

IPK

Rendah

Tinggi

Sedang

Penghasilan Orang Tua

Rendah

Tinggi

Sedang

Jumlah Tanggungan

Rendah

Tinggi

Sedang

Jumlah Sertifikat

Rendah

Tinggi

Sedang

HOME ADMIN

☐ Halo, Admin

DATA DAN KETENTUAN

Data Mahasiswa

Pengaturan Kriteria

FUZZIFIKASI DAN HASIL KLASIFIKASI

Proses Fuzzifikasi

Data Penerima Beasiswa

Kalender

Gambar 3.8 Desain Pengaturan kriteria

6. Desain Proses Fuzzifikasi

KLASIFIKASI PENERIMAAN BEASISWA PPA-FUZZY TAHANI

Gambar 3.9 Desain Proses Hitung Fuzzyfikasi

Ortu dan Jumlah Sertifikat dengan Tiga himpunan pada masing-masing Variabel.

1 IPK (Indeks Prestasi Komulatif)

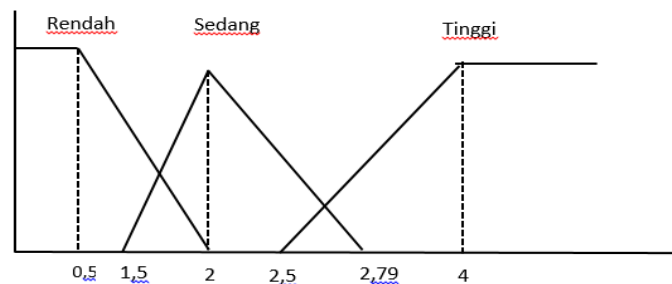
Pada Variabel IPK menggunakan tiga himpunan yaitu Rendah, Sedang dan Tinggi. Berikut tiga himpunan pada variabel IPK :

Rendah = 0.5 - 2

Sedang = 1.5 – 2.79

Tinggi = 2.5 - 4

Dari range IPK diatas dapat dibuat fungsi keanggotaan terlihat pada gambar dibawah ini :



Gambar 3.12 Grafik IPK (Indeks Prestasi Akademik)

Fungsi Keanggotaan

$$Rendah = \begin{cases} 1 & a \leq 0,5 \\ \frac{(2-a)}{0,5} & 0,5 \leq a \leq 2 \\ 0 & a > 2 \end{cases} \quad Tinggi = \begin{cases} 0 & 2,5 \leq a \text{ atau } a \geq 4 \\ \frac{(a-2,5)}{1,25} & 2,5 \leq a \leq 4 \\ 1,5 & a \geq 4 \end{cases}$$

$$Sedang = \begin{cases} 0 & 1,5 \leq a \text{ atau } a > 2,79 \\ \frac{(a - 1,5)}{0,5} & 1,5 \leq a \leq 2 \\ \frac{(2,79 - a)}{0,75} & 2 \leq a \leq 2,79 \end{cases}$$

2 Penghasilan

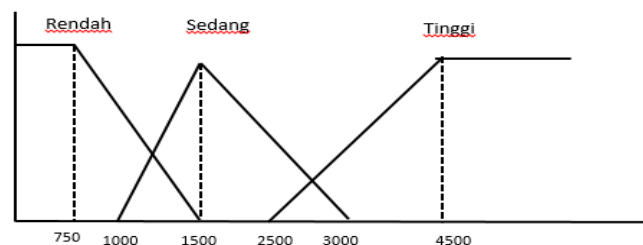
Pada Variabel Penghasilan menggunakan tiga himpunan yaitu Rendah, Sedang dan Tinggi. Berikut tiga himpunan pada variabel Penghasilan :

Rendah = 750000 - 1500000

Sedang = 1000000 - 3000000

Tinggi = 2500000 - 4500000

Dari range Penghasilan diatas dapat dibuat fungsi keanggotaan terlihat pada gambar dibawah ini :



Gambar 3.13 Grafik Penghasilan Orang Tua

Fungsi Keanggotaan

$$Rendah = \begin{cases} 1 & a < 750.000 \\ \frac{(1.500.000 - a)}{750.000} & 750.000 \leq a \leq 1.500.000 \\ 0 & a \geq 1.500.000 \end{cases} \quad Tinggi = \begin{cases} 0 & 2.500.000 < a \text{ atau } a > 4.500.000 \\ \frac{(a - 2.500.000)}{1.000.000} & 2.500.000 \leq a \leq 4.500.000 \\ 1.000.000 & a \geq 4.500.000 \end{cases}$$

$$Sedang = \begin{cases} 0 & 1.000.000 < a \text{ atau } a > 3.000.000 \\ \frac{(a - 1.000.000)}{0,5} & 1.000.000 \leq a \leq 1.500.000 \\ \frac{(3.000.000 - a)}{1.500.000} & 1.500.000 \leq a \leq 3.000.000 \end{cases}$$

3 Tanggungan

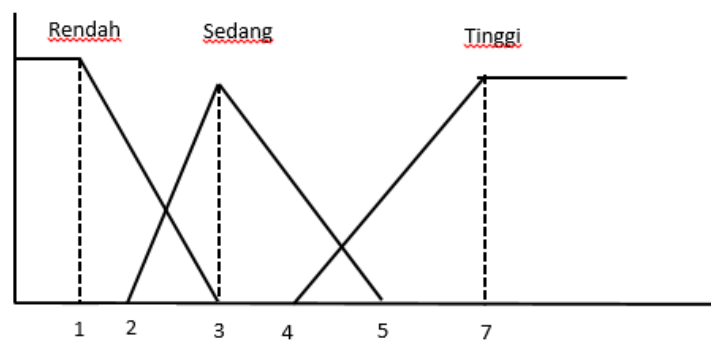
Pada Variabel Tanggungan menggunakan tiga himpunan yaitu Rendah, Sedang dan Tinggi. Berikut tiga himpunan pada variabel tanggungan :

Rendah = 1 - 3

Sedang = 2 - 5

Tinggi = 4 - 7

Dari range Tanggungan diatas dapat dibuat fungsi keanggotaan terlihat pada gambar dibawah ini :



Gambar 3.14 Grafik Jumlah Tanggungan Orang Tua

Fungsi Keanggotaan

$$Rendah = \begin{cases} 1 & a \leq 1 \\ \frac{(a-1)}{2} & 1 \leq a \leq 3 \\ 0 & a \geq 3 \end{cases}$$

$$Tinggi = \begin{cases} 0 & a \leq 4 \\ \frac{(a-4)}{3} & 4 \leq a \leq 7 \\ 1 & a \geq 7 \end{cases}$$

$$Sedang = \begin{cases} 0 & a \leq 2 \text{ atau } a \geq 5 \\ \frac{(2-a)}{2} & 2 \leq a \leq 3 \\ \frac{(5-a)}{2} & 3 \leq a \leq 5 \end{cases}$$

4 Jumlah sertifikat

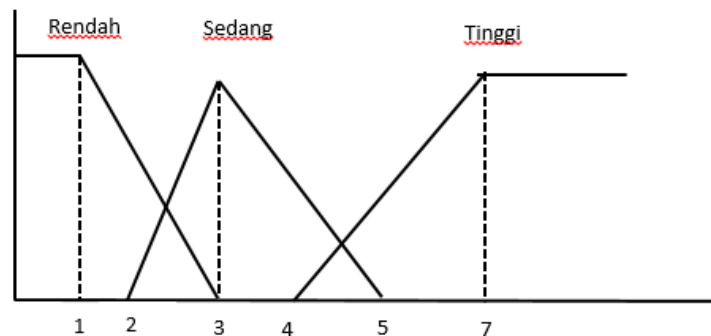
Pada Variabel jumlah sertifikat menggunakan tiga himpunan yaitu Rendah, Sedang dan Tinggi. Berikut tiga himpunan pada variabel jumlah sertifikat :

Rendah = 1 - 3

Sedang = 2 - 5

Tinggi = 4 - 7

Dari range Jumlah sertifikat diatas dapat dibuat fungsi keanggotaan terlihat pada gambar dibawah ini :



Gambar 3.15 Grafik Jumlah Sertifikat

Fungsi Keanggotaan

$$Rendah = \begin{cases} 1 & a \leq 1 \\ \frac{(a-1)}{2} & 1 \leq a \leq 3 \\ 0 & a \geq 3 \end{cases}$$

$$Tinggi = \begin{cases} 0 & a \leq 4 \\ \frac{(a-4)}{3} & 4 \leq a \leq 7 \\ 1 & a \geq 7 \end{cases}$$

$$Sedang = \begin{cases} 0 & a \leq 2 \text{ atau } a \geq 5 \\ \frac{(2-a)}{2} & 2 \leq a \leq 3 \\ \frac{(5-a)}{2} & 3 \leq a \leq 5 \end{cases}$$

BAB IV

HASIL DAN PEMBAHASAN

Bab ini menjelaskan tahapan-tahapan implementasi sistem penerapan logika fuzzy tahani pada beasiswa PPA. Berikut adalah penjelasannya:

4.1. Persiapan Implementasi

4.1.1. Instalasi Web Server dan Database Server

Web server dan database server sudah termasuk dalam satu program yaitu XAMPP. Web server menggunakan Apache server dan database server menggunakan MySQL.

4.1.2. Instalasi Teks Editor

Teks editor yang digunakan untuk memudahkan penulis dalam membangun program adalah Sublime Text 3.

4.1.3. Instalasi Perangkat Lunak Pendukung

Perangkat lunak pendukung yang digunakan penulis adalah MySQL Workbench untuk memudahkan dalam perancangan database.

4.2. Tahap Pembangunan

4.2.1. Pembuatan Database

Pembuatan database menggunakan database server MySQL dan dibantu dengan aplikasi MySQL Workbench untuk memudahkan penulis. Database digunakan untuk menyimpan data-data yang diperlukan untuk

menjalankan sistem penerapan logika fuzzy tahani pada penerima beasiswa PPA.

4.2.2. Pembangunan Program

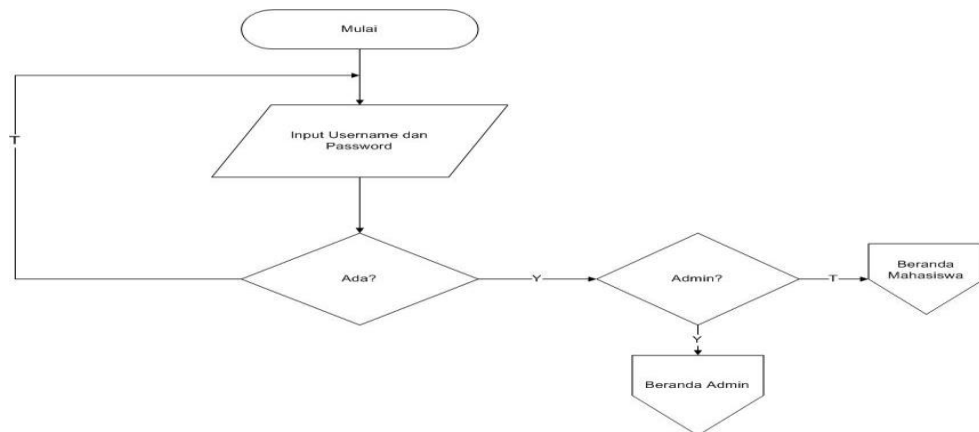
Pembangunan program menggunakan bantuan teks editor Sublime Text 3. Program dibangun menggunakan bahasa pemrograman PHP. Sublime Text 3 dipilih karena merupakan aplikasi yang ringan dan memiliki banyak plugin tambahan yang dapat memudahkan penulis dalam membangun sistem penerapan logika fuzzy tahani pada penerima beasiswa PPA.

4.3. Penjelasan Flowchart Program dan Program

Pada sub bab ini akan dijelaskan alur proses yang terjadi dalam program menggunakan flowchart dan beberapa tampilan program. Sesuai dengan perancangan yang sudah dijelaskan dalam BAB III, sistem penerapan logika fuzzy tahani pada penerima beasiswa PPA. diantaranya: Login, beranda, input data mahasiswa, input pengaturan kriteria, input penerima beasiswa. Berikut adalah penjelasannya:

4.3.1. Halaman Login

Halaman ini adalah tampilan awal dari sistem. Halaman login digunakan oleh pengguna untuk masuk ke sistem penerapan logika fuzzy tahani pada penerima beasiswa PPA. Berikut flowchart halaman login:



Gambar 4.1 Flowchart Login

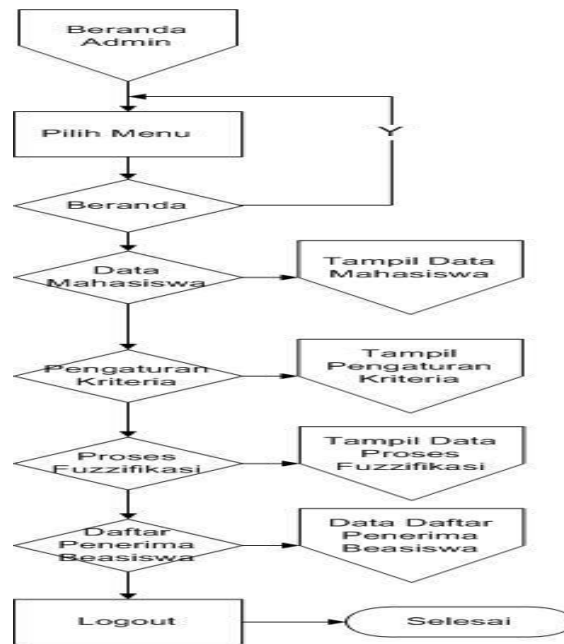
Tampilan Halaman Login

The screenshot shows a web application interface. At the top, there is a blue header bar with the text 'KLASIFIKASI PENERIMA BEASISWA PPA' on the left and 'Home Login' on the right. Below the header, the main content area has a light gray background. On the left side, there is a login form titled 'SILAHKAN LOG-IN DI SINI:'. The form contains three input fields: 'username' with a user icon, 'password' with a key icon, and a dropdown menu with a gear icon. Below these fields is a blue button labeled 'Masuk'. To the right of the form, there is a text block titled 'Ketentuan:' which states: 'Gunakan akun yang valid untuk memperoleh akses ke sistem. Apabila belum memiliki akun atau mengalami masalah, silahkan hubungi admin.' At the bottom right of the page, there are small icons for zooming and a percentage indicator '80%'.

Gambar 4.2 Tampilan Halaman Login

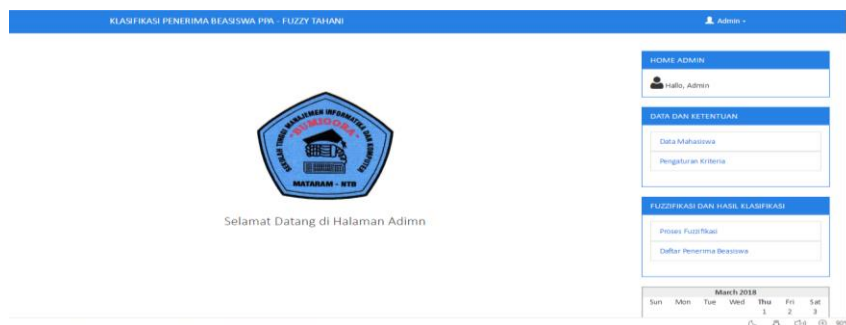
4.3.2. Halaman Beranda

Halaman beranda adalah halaman yang dituju jika pengguna berhasil login. Berikut flowchart halaman Beranda



Gambar 4.3 Flowchart Halaman Beranda

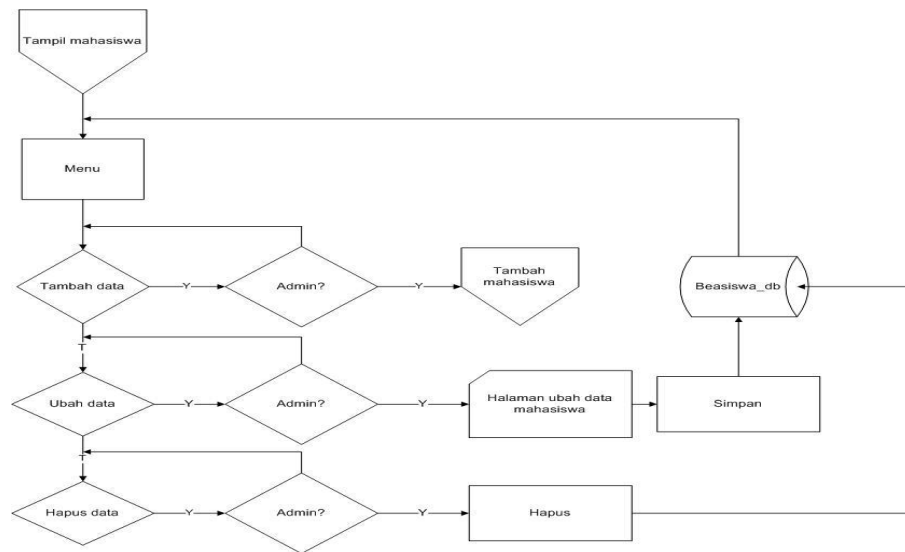
Tampilan Halaman Beranda



Gambar 4.4 Tampilan Halaman Beranda

4.3.3. Halaman Tampil Data Mahasiswa

Halaman ini digunakan untuk menampilkan data mahasiswa yang sudah diinputkan. Berikut flowchart halaman tampil data mahasiswa:



Gambar 4.5 Flowchart Tampil Data Mahasiswa

Halaman Tampil Data Mahasiswa

KLASIFIKASI PENERIMA BEASISWA PPA - FUZZY TAHANI

Admin -

DATA MAHASISWA CALON PENERIMA BEASISWA

Entri Data

No	NPM	Nama	Alamat	Jurusan/Profil	IPK	Nama Ayah	Pekerjaan	Aksi
1	12.8.349.74.75.0.5.0142	Maktub	Praya	S1 Teknik Informatika	1.4	Dra	Petani	<div><div></div><div>+</div><div>-</div></div>
2	13.8.349.74.75.0.5.0103	Sti Asyah	Mataram	S1 Teknik Informatika	3.5	Mariwan	Pedagang	<div><div></div><div>+</div><div>-</div></div>
3	13.8.349.74.75.0.5.0112	Adi Natsalegawa	Rembang	D3 Teknik Informatika	1.5	Sohrun	Nelayan	<div><div></div><div>+</div><div>-</div></div>
4	13.8.349.74.75.0.5.0115	Auliya	Kopang	S1 Teknik Informatika	1.7	Murtaji	Buruh Tani	<div><div></div><div>+</div><div>-</div></div>
5	13.8.349.74.75.0.5.0116	Aini	Jarapra	D3 Teknik Informatika	2	Sukron	Tukang	<div><div></div><div>+</div><div>-</div></div>
6	13.8.349.74.75.0.5.0130	Imam Mar'uf	Mataram	S1 Teknik Informatika	3.4	Misbahudin	PNS	<div><div></div><div>+</div><div>-</div></div>
7	13.8.349.74.75.0.5.0141	Alexis Sances	Sakra	S1 Teknik Informatika	2	Ropak	Nelayan	<div><div></div><div>+</div><div>-</div></div>
8	13.8.349.74.75.0.5.0171	Wayan	Tanjung	S1 Teknik Informatika	2.1	Masrum	Petani	<div><div></div><div>+</div><div>-</div></div>
9	13.8.349.74.75.0.5.0257	Arman	Ampenan	S1 DKV	3.12	mastur	Nelayan	<div><div></div><div>+</div><div>-</div></div>
10	13.8.349.74.75.0.5.0260	Khairun Naji	Gurung Sari	S1 Teknik Informatika	2.78	Sulhan	Pensiunan	<div><div></div><div>+</div><div>-</div></div>
11	13.8.349.74.75.0.5.0262	Sakira	Terera	S1 Teknik Informatika	1.5	Wimar	Tukang	<div><div></div><div>+</div><div>-</div></div>

HOME ADMIN

Halo, Admin

DATA DAN KETENTUAN

Data Mahasiswa

Pengaturan Kriteria

FUZZIFIKASI DAN HASIL KLASIFIKASI

Proses Fuzzifikasi

Daftar Penerima Beasiswa

July 2018

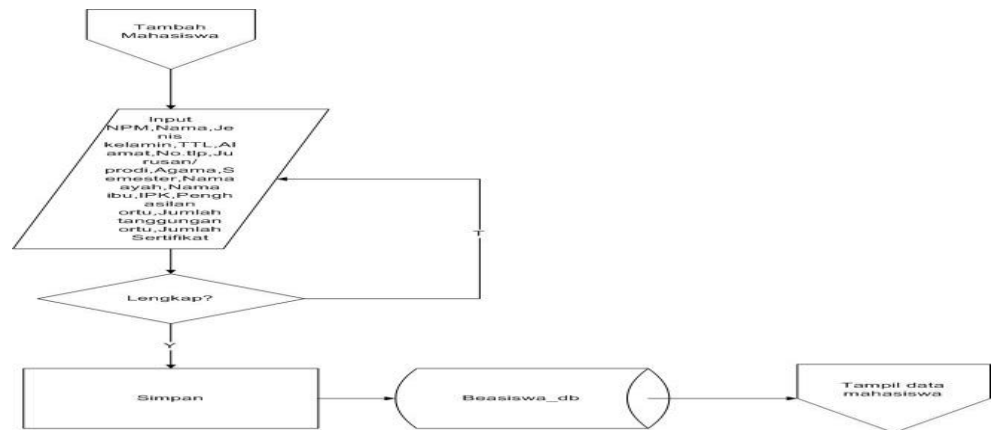
Sun	Mon	Tue	Wed	Thu	Fri	Sat
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31				

Gambar 4.6 Gambar Halaman Tampil Data Mahasiswa

4.3.4. Halaman Input Data Mahasiswa

Halaman ini digunakan untuk melakukan input data mahasiswa.

Berikut flowchart input data mahasiswa:



Gambar 4.7 Flowchart input data mahasiswa

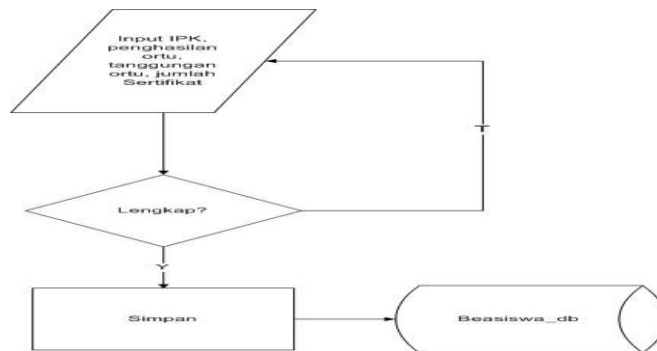
Halaman input data mahasiswa

The screenshot shows a web application interface for "KLASIFIKASI PENERIMA BEASISWA PPA - FUZZY TAHANI". The main section is titled "DATA MAHASISWA CALON PENERIMA BEASISWA" and contains a form with two columns of input fields. The left column includes NPM, Nama, Jenis Kelamin (dropdown), TTL, Alamat, No. Tlp, Jurusan/Prodi, and Agama (dropdown). The right column includes Smerster, Nama Ayah, Nama Ibu, IPK, Pekerjaan Ortu (dropdown), Penghasilan Ortu, Jum. Tanggungan Ortu, and Jum. Sertifikat. A "Simpan Data" button is at the bottom center. The right sidebar contains a "HOME ADMIN" section with a user greeting, a "DATA DAN KETENTUAN" section with links to "Data Mahasiswa" and "Pengaturan Kriteria", and a "FUZZIFIKASI DAN HASIL KLASIFIKASI" section with links to "Proses Fuzzifikasi" and "Daftar Penerima Beasiswa". At the bottom right, there is a calendar for July 2018.

Gambar 4.8 Halaman input data mahasiswa

4.3.5. Halaman Pengaturan Kriteria

Halaman ini menampilkan data pengaturan kriteria. Berikut flowchart halaman tampil data pengaturan kriteria :



Gambar 4.9 Flowchart Pengaturan kriteria

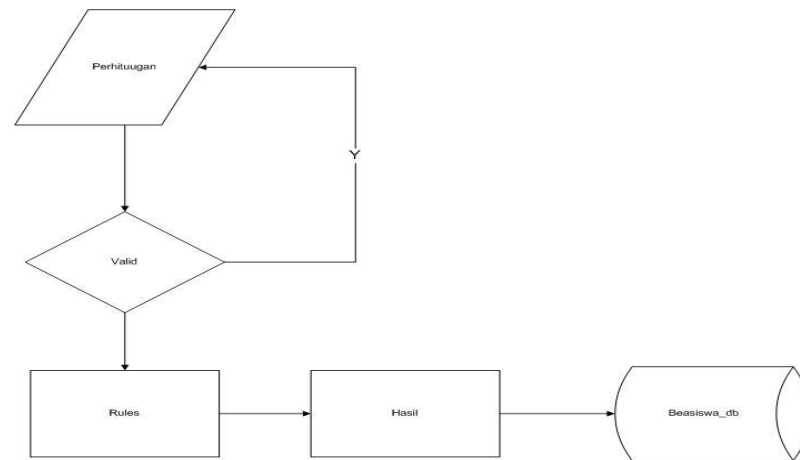
Halaman pengaturan kriteria

Gambar 4.10 Pengaturan kriteria

4.3.6. Halaman Perhitungan Proses Fuzzyfikasi

Halaman ini digunakan untuk menghitung proses fuzzyfikasi.

Berikut flowchart halaman proses fuzzyfikasi :



Gambar 4.11 Flowchart Fuzzyfikasi

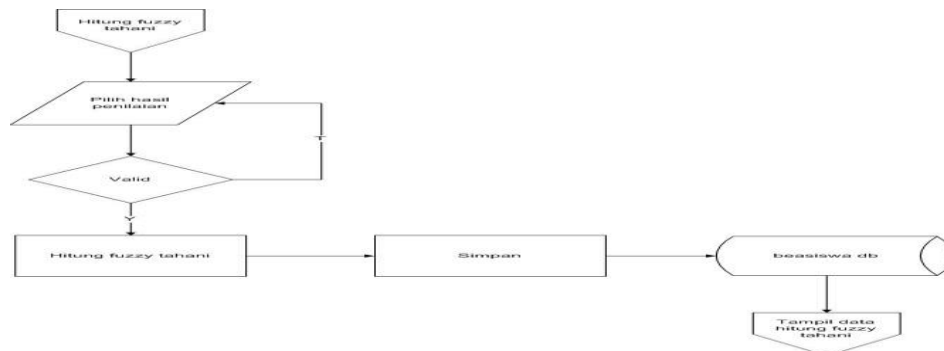
Halaman perhitungan proses fuzzyfikasi

KLASIFIKASI PENERIMA BEASISWA PPA - FUZZY TAHANI													
Admin													
PROSES FUZZIFIKASI (PERHITUNGAN BOBOT/NILAI DERAJAT KEANGGOTAAN)													
HITUNG													
No	NPM	IPK			Penghasilan			Jumlah Tanggungan			Sertifikat		
		Rendah	Sedang	Tinggi	Rendah	Sedang	Tinggi	Rendah	Sedang	Tinggi	Rendah	Sedang	Tinggi
1	12.8.349.74.75.0.5.0142	0.4	0	0	1	0	0	0.5	0	0	0.5	0	0
2	13.8.349.74.75.0.5.0103	0	0	0.666667	0.8	0	0	0.5	0	0	1	0	0
3	13.8.349.74.75.0.5.0112	0.333333	0	0	1	0	0	0.5	0	0	0	0	1
4	13.8.349.74.75.0.5.0115	0.2	0.4	0	0.8	0	0	0	0	0.333333	1	0	0
5	13.8.349.74.75.0.5.0116	0	1	0	0.733333	0	0	0.5	0	0	0	0.5	0.333333
6	13.8.349.74.75.0.5.0130	0	0	0.6	0.666667	0	0	0	0	1	0	0	1
7	13.8.349.74.75.0.5.0141	0	1	0	0	0.8	0	0.5	0	0	1	0	0
8	13.8.349.74.75.0.5.0171	0	0.873418	0	0.933333	0	0	1	0	0	0.5	0	0
9	13.8.349.74.75.0.5.0257	0	0	0.413333	0	0.466667	0	0	0	0.333333	0	0	1
10	13.8.349.74.75.0.5.0260	0	0.0126582	0.186667	0	0.666667	0	0	0	0.666667	0	0.5	0.333333
11	13.8.349.74.75.0.5.0262	0.333333	0	0	0	0	0.5	1	0	0	0.5	0	0
12	13.8.349.74.75.0.5.0266	0.266667	0.2	0	0.866667	0	0	0	0.5	1	0	1	1

Gambar 4.12 Gambar proses hitung fuzzyfikasi

4.3.7. Halaman Tampil Penerima Beasiswa

Halaman ini menampilkan data hasil perhitungan penerima beasiswa. Berikut flowchart halaman tampil perhitungan penerima beasiswa:



Gambar 4.13 Flowchart Tampil penerima beasiswa

Halaman Tampil Perhitungan Penerima Beasiswa :

KLASIFIKASI PENERIMA BEASISWA PPA - FUZZY TAHANI Admin

QUERY KLASIFIKASI PENERIMA BEASISWA

IPK

 Penghasilan

Tanggungan

 Sertifikat

Proses Data

HOME ADMIN

Hallo, Admin

DATA DAN KETENTUAN

[Data Mahasiswa](#)

[Pengaturan Kriteria](#)

FUZZIFIKASI DAN HASIL KLASIFIKASI

[Proses Fuzzifikasi](#)

[Daftar Penerima Beasiswa](#)

TABEL HASIL KLASIFIKASI PENERIMA BEASISWA PPA

No	NPM	Nama	Bobot IPK	Bobot Penghasilan	Bobot Jumlah Tanggungan	Bobot Sertifikat	Total Bobot
1	11.8.349.74.75.0.5.0058	Festabiqul Khairat	0.6	1	0.333333	0.666667	2.6
2	11.8.349.74.75.0.5.0087	Widan	0.693333	1	0.333333	1	3.026666
3	11.8.349.74.75.0.5.0119	Azmi	0.613333	0.833333	1	0.666667	3.113333
4	11.8.349.74.75.0.5.0163	Dhedet Pratama	0.733333	0.4	0.333333	1	2.466666
5	12.8.349.74.75.0.5.0032	SriYunita Anggraeni	0.373333	0.7392	1	1	3.112533
6	13.8.349.74.75.0.5.0174	I Lena Aulia	0.64	0.153566	1	1	2.793566

August 2018

Sun	Mon	Tue	Wed	Thu	Fri	Sat
			1	2	3	4
5	6	7	8	9	10	11

Gambar 4.14 Halaman Tampil Perhitungan Penerima Beasiswa

4.4. Tahap Pengujian

Pengujian dilakukan dengan menggunakan data asli untuk pendaftar beasiswa PPA tahun 2014. Tujuan pengujian adalah bagaimana hasilnya menggunakan metode fuzzy tahani pada penerimaan beasiswa 2014. Jumlah pendaftar beasiswa 2014 adalah 64 mahasiswa. Sistem penerapan fuzzy tahani pada penerimaan beasiswa PPA 2014 dapat dinilai baik menggunakan

$$\text{Rumus : Akurasi} = \frac{\text{Hasil Dengan Sistem}}{\text{Hasil Dengan data}} \times 100$$

Data Pendaftar Beasiswa 2014

Tabel 4.1 Data Pendaftar Beasiswa

N o	NPM	Nama	Seme ster	IPK	Penghasilan	Tanggung an	Jumlah Sertifik at
1	11.8.349.74.75.0.5.0163	DHEDET PRATAMA	VI	3,60	Rp.1.200.000	5	10
2	11.8.349.74.75.0.5.0053	AGUS PARHAN SURIAZ	VI	3,50	Rp.1.012.150	3	4
3	12.8.349.74.75.0.5.0164	AGUSFIAN HADI SAPUTRA	IV	3,33	Rp. 450.000	3	
4	13.8.349.74.75.0.5.0106	ROBIUZZAMA N	II	3,50	Rp.4.620.100	3	1
5	11.8.349.72.74.0.3.0002	BAIQ. WIDYA MAYA FEBRIANA	VI	3,24	Rp.1.019.000	3	2
6	13.8.349.74.75.0.5.0045	LALU MUH. KUMBAYONI	II	3,72	Rp.500.000	2	1
7	13.8.349.74.75.0.5.0040	FRANSDIKA DHARMAWAN	II	3,92	Rp.Rp.2.200.000	3	1
8	12.8.349.32.53.0.2.0005	EKA NURLAILA WARZUKI	IV	3,72	Rp.750.000	2	4
9	12.8.349.74.75.0.5.0174	ANDI ARISKA NOVIANTI	IV	3,29	Rp. 500.000	3	2
10	13.8.349.74.75.0.5.0130	IMAM MA'RUF	II	3,64	Rp.3.206.400	2	1
11	11.8.349.74.75.0.5.0006	BILI NATA NAJWA	VI	3,62	Rp.2.000.000	4	
12	11.8.349.74.75.0.5.0002	MUHAMMAD SYAKIR	VI	3,62	Rp.500.000	3	
13	11.8.349.74.75.0.5.0119	AZMI	VI	3,42	Rp.1.750.000	4	5
14	11.8.349.74.75.0.5.0087	WILDAN	VI	3,54	Rp.1.500.000	5	9
15	11.8.349.74.75.0.5.0011	SITI AGRIPINA ALODIA YUSUF	VI	3,44	Rp.4.011.204	4	3
16	11.8.349.74.75.0.5.0275	NINSI PUTRIANA DEVI	VI	3,00	Rp.600.000	4	4
17	13.8.349.74.75.0.5.0217	NURHANDINI	II	3,47	Rp.3.357.900	2	
18	11.8.349.74.75.0.5.0247	DEVIANA AYU LESTARI	VI	3,29	Rp.1.527.930	4	
19	13.8.349.74.75.0.5.0066	FENI ALFAONITA	II	3,56	Rp.1.000.000	3	
20	13.8.349.74.75.0.5.0127	FIRDAUS TRI ENDAH WAYUNINGSIH	II	3,42	Rp.4.668.975	3	3
21	13.8.349.74.75.0.5.0133	LALU ABD RAHMAN HAKIM	II	3,58	Rp.1.071.500	1	4
22	11.8.349.74.75.0.5.0056	TRIPUSNAWATI	VI	3,09	Rp.900.000	2	2
23	12.8.349.74.75.0.5.0142	KURTUBIAH	IV	3,19	Rp.750.000	4	
24	13.8.349.74.75.0.5.0138	SUNANDA AFRIANTO	II	3,22	Rp.1.200.000	1	
25	12.8.349.74.75.0.5.0196	ZUHRI JULIARDI	IV	3,42	Rp.1.910.900	2	1
26	13.8.349.74.75.0.5.0075	HAIRUL IMAM	II	3,72	Rp.3.327.200	1	1
27	12.8.349.74.75.0.5.0193	AHMAD MEIRADI	IV	3,43	Rp.900.000	5	1
28	11.8.349.74.75.0.5.0238	FERRY AGUSMANSY	VI	3,27	Rp.2.000.000	4	3

		AH MA'AS					
29	11.8.349.74.75.0.5.0071	ABDUL AZIZ ALI	VI	3,21	Rp.1.500.000	2	10
30	11.8.349.74.75.0.5.0120	LALU HENDIAWAN DIPA	VI	3,54	Rp.821.726	2	3
31	12.8.349.32.53.0.3.0018	DWI RAGITA MULYAWATI	IV	3,52	Rp.3.693.531	4	6
32	13.8.349.74.75.0.5.0173	DINDARI BELA QUR'ANI	II	3,47	Rp.700.000	2	2
33	12.8.349.74.75.0.5.0221	ANDY HARDIYANTI HASTUTI	IV	3,75	Rp.2.770.000	4	
34	12.8.349.74.75.0.5.0169	B. DINDA USWATUN HASANAH	IV	3,37	Rp.4.355.200	2	1
35	12.8.349.74.75.0.5.0032	SRI YUNITA ANGGRAENI	IV	3,06	Rp.1.891.200	4	3
36	13.8.349.74.75.0.5.0008	GALUH WAHYUNI FAZRI	IV	3,43	Rp.1.766.600	3	
37	13.8.349.74.75.0.5.0033	AHMAD SYAHBANDI	II	3,86	Rp.700.000	1	
38	13.8.349.74.75.0.5.0112	SATRIADI KAWIRIAN	II	3,56	Rp.500.000	4	
39	11.8.349.74.75.0.5.0022	MOH. ZAINUDDIN	VI	3,11	Rp.300.000	1	3
40	13.8.349.32.53.0.2.0011	MUHAMMAD ZAKI AZHARI	II	3,21	Rp.700.000	3	
41	11.8.349.74.75.0.5.0006	MUH. RIDHA FADLI	VI	3,00	Rp.3.225.400	2	2
42	11.8.349.74.75.0.5.0090	MUH. ERWIN INDRAWAN	VI	3,48	Rp.600.000	2	7
43	13.8.349.74.75.0.5.0174	I LENA AULIA	11	3,36	Rp.2.769.651	4	10
44	13.8.349.32.53.0.2.0022	HALIMATUSS AKDIAH	II	3,24	Rp.500.000	2	
45	13.8.349.74.75.0.5.0220	ANA ROHMIATI	II	3,13	Rp.3.712.373	3	
46	12.8.349.32.53.0.2.0025	MUH. KHOLIS KAWIRYA PUTRA	IV	3,34	Rp.3.331.900	2	2
47	11.8.349.74.75.0.5.0058	FASTABIQUL KHAIRAT	VI	3,40	Rp.1.500.000	5	5
48	11.8.349.74.75.0.5.0133	LALU ZAMRUD ZULBAINI KARNA	VI	3,29	Rp.2.738.500	3	4
49	12.8.349.74.75.0.5.0126	SHOFIAN ANSHORI	IV	3,37	Rp.600.000	4	1
50	12.8.349.74.75.0.5.0098	AHMAD MUHAR DIAN LASMITA	IV	3,34	Rp.500.000	2	
51	12.8.349.74.75.0.5.0106	MUHAMAD AZWAR	IV	3,38	Rp.500.000	4	
52	11.8.349.74.75.0.5.0048	HERLINA APRIANA	VI	3,16	Rp.700.000	2	5
53	13.8.349.0.74.75.5.0205	LALU YOGI PRATAMA	II	3,81	Rp.3.099.200	2	
54	13.8.349.74.75.0.5.0164	RUDI HENDRAWAN	II	3,58	Rp.1.000.000	3	
55	12.8.349.74.75.0.5.0005	RADEN AGUNG BAGUS WIRO KUSUMO	IV	3,69	Rp.2.000.000	3	
56	13.8.349.74.75.0.5.0009	WINDA YUSNITA	II	3,40	Rp.1.917.100	1	
57	13.8.349.74.75.0.5.0197	DANI ARISTIYAWAN	II	3,17	Rp.2.518.200	3	

58	12.8.349.74.75.0.5.0002	SULPAN	IV	3,28	Rp.1.920.000	2	
59	13.8.349.74.75.0.5.0128	PUTRI AISYA GUSTI	II	3,42	Rp.2.280.000	1	4
60	12.8.349.74.75.0.5.0012	BQ. NURASMAH	IV	3,34	Rp.700.000	6	
61	13.8.349.74.75.0.5.0207	HAYATULLOH FIRMAN HADI	II	3,50	Rp.1.500.000	3	
62	12.8.349.74.75.0.5.0070	BAIQ ELA NOPITA LESTARI	IV	3,29	Rp.1.200.000	2	
63	12.8.349.74.75.0.5.0006	BAHRUL HIDAYAT	IV	3,63	Rp.2.769.651	3	
64	12.8.349.74.75.0.5.0076	WISNU MAULANA	IV	3,62	Rp.2.000.000	1	

Data Pendaftar Penerima Beasiswa 2014

Tabel 4.2 Data Pendaftar Penerima Beasiswa dengan manual dan hasil sistem

No	NPM	Nama dengan data manual	Nama Ujicoba dengan Sistem	Semester	IPK	Penghasilan	Tanggung an	Jumlah Sertifikat
1	11.8.349.74.75.0.5.0163	DHEDET PRATAMA	-	6	3.60	1.200.000	5	10
2	13.8.349.74.75.0.5.0040	FRANSDIKA DHARMA WAN	-	2	3.92	2.200.000	3	4
3	12.8.349.32.53.0.2.0005	EKA NURLAILA WARZUKI	EKA NURLAILA WARZUKI	4	3.72	750.000	4	
4	12.8.349.74.75.0.5.0174	ANDI ARISKA NOPIANTI	ANDI ARISKA NOPIANTI	4	3.29	5.00.000	5	1
5	13.8.349.74.75.0.5.0066	FENI ALFAONITA	-	2	3.56	1.000.000	3	2
6	13.8.349.74.75.0.5.0133	LALU ABD RAHMAN HAKIM	LALU ABD RAHMAN HAKIM	2	3.58	1.071.500	2	1
7	13.8.349.74.75.0.5.0075	HAIRUL IMAM	HAIRUL IMAM	6	3.72	3.327.200	1	1
8	11.8.349.74.75.0.5.0120	LALU HENDIAWAN DIPA	LALU HENDIAWAN DIPA	6	3.45	821.726.36	4	4
9	12.8.349.32.53.0.3.0018	DWI RAGITA MULYAWATI	-	4	3.52	3.693.246	4	2
10	12.8.349.74.75.0.5.0169	B. DINDA USWATUN HASANAH	B. DINDA USWATUN HASANAH	4	3.37	4.355.200	2	1
11	13.8.349.74.75.0.5.0033	AHMAD SYAHBANDI	AHMAD SYAHBANDI	2	3.86	700.000	1	
12	13.8.349.74.75.0.5.0112	SATRIADI KAWIRIAN	SATRIADI KAWIRIAN	2	3.56	500.000	4	
13	11.8.349.74.75.0.5.0090	MoH. ERWIN INDRAWAN	MoH. ERWIN INDRAWAN	6	3.48	600.000	2	5

14	13.8.349.74.75.0.5 .0205	LALU YOGI PRATAMA	LALU YOGI PRATAMA	2	3.81	3.099.200	2	9
15	13.8.349.74.75.0.5 .0045	LALU MUH. KUMBAY ONI	LALU MUH. KUMBAYO NI	2	3.72	500.000	2	3

Hasil Perhitungan :

$$Akurasi = \frac{\text{Hasil Dengan Sistem}}{\text{Hasil Dengan data}} \times 100$$

$$= \frac{11}{15} \times 100$$

$$= 73.3$$

Kesimpulan :

Berdasarkan hasil perhitungan akurasi dari data pendaftar penerima beasiswa 2014 adalah 73.3, dengan rincian hasil dengan sistem mendapatkann 11 mahasiswa yang layak mendapatkan beasiswa dengan rule yang ada dan data hanya 15 kuota yang disediakan.

BAB V

PENUTUP

5.1 Kesimpulan

Dari hasil sistem penerimaan beasiswa tahun 2014 di STMIK Bumigora Mataram menggunakan metode fuzzy tahani, penulis dapat menuliskan kesimpulan sebagai berikut:

Sistem Pendukung Keputusan untuk penentuan hasil beasiswa di STMIK Bumigora Mataram yang dibangun dengan metode Fuzzy Tahani memberikan akurasi hingga 73.3%.

5.2 Saran

Dalam penelitian ini masih banyak kekurangan, diantaranya adalah proses input data pendaftar masih dilakukan oleh admin, rentang nilai yang kecil kemungkinan total nilai satu atau dua alternatif menjadi sama, Oleh karena itu bagi yang nantinya ingin melakukan pengembangan dari penelitian ini dapat menambahkan fitur yang lebih lengkap dalam input data , input data dilakukan oleh pendaftar penerima beasiswa sendiri, memberikan rentang nilai yang lebih agar kemungkinan total nilai yang sama menjadi lebih kecil, dan menggunakan metode yang menghasilkan hasil yang lebih akurat.

DAFTAR REFERENSI

- Deddy Irwan : Implementasi Fuzzy Query Pada Database Untuk Rekomendasi Beasiswa. 2008. USU Repository.
- Kadir, Abdul. (2008). Belajar database menggunakan MySQL. Yogyakarta: Penerbit Andi.
- Kadir, Abdul.(2001). Dasar Pemrograman Web Dinamis Menggunakan PHP.Yogyakarta: Penerbit Andi
- Kusumadewi S, Purnomo H. 2004 Fuzzy Database
<http://repository.usu.ac.id/ChapterII> , di akses 25 Januari 2017.
- Kusumadewi S, Purnomo H. 2004 Pengertian Fuzzy Tahani.
<http://repository.usu.ac.id/ChapterII> , di akses 25 Januari 2017.
- Simarmata, Janner. (2009). Rekayasa Web. Medan: Penerbit Andi.
- Thomas Sri Widodo, 2005, "Sistem Neuro Fuzzy", Graha Ilmu, Yogyakarta.
- Ucoe Poe. 2016. Pengertian Codeigniter.
<http://www.ucoepoe.com/2016/05/pengertian-codeigniter-fungsi-kelebihan-dan-kekurangannya.html>. di akses 20 agustus 2017.

LAMPIRAN

Lampiran Jurnal

Penerapan Logika Fuzzy Tahani pada Penerimaan Beasiswa Peningkatan Prestasi Akademik (Studi Kasus: STMIK Bumigora Mataram)

¹M Rodi Taufik Akbar, ²Muhammad Yunus

¹Mahasiswa, ²Dosen, Program Studi Teknik Informatika
STMIK Bumigora Mataram; Jl. Ismail Marzuki Mataram,
rodisarabar@gmail.com, muhyunus.446@gmail.com

Abstrak

Sistem database relasional yang ada sampai sekarang ini hanya mampu menangani data yang bersifat pasti (crisp), deterministik dan presisi. Padahal dalam kondisi nyata, sering sekali dibutuhkan data yang samar untuk proses pengambilan keputusan. Untuk pengambilan keputusan yang melibatkan variabel samar berdasarkan data crisp yang ada di dalam database, dapat direalisasikan dengan mengembangkan suatu aplikasi untuk menangani query yang diberikan kepada sistem database serta dengan melakukan fuzzifikasi pada data. Di setiap lembaga pendidikan khususnya perguruan Tinggi banyak sekali beasiswa yang ditawarkan kepada mahasiswa. Untuk mendapatkan beasiswa tersebut maka harus sesuai dengan aturan yang telah ditetapkan. Oleh sebab itu tidak semua mahasiswa yang mengajukan permohonan untuk menerima beasiswa dapat diberikan. Karena jumlah mahasiswa yang mengajukan permohonan yang banyak dan kriteria penilaian yang banyak pula, maka perlu dibangun suatu sistem yang dapat membantu memberikan rekomendasi penerima beasiswa. Selama ini sistem yang digunakan untuk proses seleksi beasiswa PPA di Sekolah Tinggi Manajemen Informatika dan Komputer(STMIK) Bumigora Mataram masih dikerjakan secara semi manual menggunakan Microsoft excel. Oleh karena itu perlu dibuatkan sistem yang khusus untuk merekomendasikan beasiswa PPA dengan menggunakan metode penerapan logika Fuzzy Tahani. Hasil dari penelitian berupa sistem mampu melakukan seleksi beasiswa PPA dengan tingkat akurasi 73,3%.

Kata kunci : Fuzzy, Tahani, Beasiswa, PPA, STMIK Bumigora

1. PENDAHULUAN

1.1 Latar Belakang

Pemerintah melalui direktorat Jenderal Pendidikan Tinggi Kementerian Pendidikan Nasional berupaya mengalokasikan dana untuk memberikan bantuan biaya pendidikan kepada mahasiswa yang orang tuanya tidak mampu untuk membiayai pendidikannya, dan memberikan beasiswa kepada mahasiswa yang mempunyai prestasi tinggi, baik dibidang akademik dan atau non akademik.

Di setiap lembaga pendidikan khususnya perguruan Tinggi banyak sekali beasiswa yang ditawarkan kepada mahasiswa.. Untuk mendapatkan beasiswa tersebut maka harus sesuai dengan aturan yang telah ditetapkan. Oleh sebab itu tidak semua mahasiswa yang mengajukan permohonan untuk menerima beasiswa dapat diberikan. Karena jumlah mahasiswa yang mengajukan permohonan yang banyak dan kriteria penilaian yang banyak pula, maka perlu dibangun suatu

sistem yang dapat membantu memberikan rekomendasi penerima beasiswa.

Sekolah Tinggi Manajemen Informatika dan Komputer (STMIK) Bumigora Mataram merupakan salah satu sekolah komputer di Nusa Tenggara Barat yang berusaha untuk menghasilkan mahasiswa yang berpotensi dan ahli pada bidang informatika (Buku pedoman STMIK Bumigora Mataram, 2013).

Selama ini sistem yang digunakan untuk proses seleksi beasiswa PPA di Sekolah Tinggi Manajemen Informatika dan Komputer (STMIK) Bumigora Mataram masih dikerjakan secara semi manual menggunakan Microsoft excel. Oleh karena itu perlu dibuatkan sistem yang khusus untuk perekomendasi beasiswa PPA dengan menggunakan metode penerapan logika Fuzzy Tahani.

1.2 Rumusan Masalah

Adapun rumusan masalah dari penelitian ini adalah menentukan penerima beasiswa Peningkatan Prestasi Akademik (PPA) di lingkungan kampus STMIK Bumigora Mataram

1.3 Batasan Masalah

Berdasarkan rumusan masalah di atas penulis memberikan batasan masalah sebagai berikut:

- e. Penelitian dilakukan di STMIK Bumigora Mataram pada bidang kemahasiswaan
- f. Kasus yang diteliti adalah untuk penerimaan Beasiswa Peningkatan Prestasi Akademik (PPA)
- g. Input atau Variabel yang digunakan adalah IPK, Penghasilan Orang Tua, Jumlah Tanggungan, Pekerjaan dan Jumlah Sertifikat. Dimana setiap variabel terdiri dari masing-masing tiga himpunan fuzzy yaitu rendah, sedang dan tinggi.
- h. Output sistem berupa hasil seleksi beasiswa yang disertai derajat atau presentase ketegasan.

1.4 Tujuan dan Manfaat Penelitian

1.4.1 Tujuan

Adapun tujuan membangun sistem Fuzzy Tahani ini adalah untuk menyeleksi penerimaan beasiswa PPA.

1.4.2 Manfaat

Adapun manfaat yang dapat diperoleh dari sistem ini adalah:

- c. Memudahkan bagi kemahasiswaan dalam seleksi beasiswa
- d. Membantu hasil yang lebih objektif karena disertai dengan bobot atau derajat ketegasan.

1.5 Metode Penelitian

1.5.1 Metode Pengumpulan Data

Metode Pengumpulan Data adalah suatu metode dimana peneliti mengumpulkan data-data dan menganalisis fakta-fakta mengenai suatu masalah pada penelitian. Adapun beberapa metode yang digunakan dalam penelitian pengumpulan data yang digunakan dalam penyusunan tugas akhir ini yaitu :

a. Studi Literatur

Mencari literatur berupa paper, text book dan tutorial – tutorial yang berkaitan dengan permasalahan pada penulisan skripsi sehingga mudah dalam penyelesaiannya

b. Wawancara

Dalam hal ini penulis melakukan Tanya jawab langsung dengan bagian kemahasiswaan yang menangani mengenai beasiswa..

1.5.2 Metode Pengembangan

Dalam penyusunan skripsi ini penulis menggunakan metode waterfall (pressman 2002) dengan tahapan-tahapan sebagai berikut :

- e. Analisis kebutuhan perangkat lunak. Proses pengumpulan kebutuhan diintensifkan dan difokuskan, khususnya pada perangkat lunak, untuk memahami sifat program yang dibangun, perekayasa perangkat lunak (analisis) harus memahami domain informasi, tingkah laku, unjuk karya,

dan antar muka (*interface*) yang diperlukan.

- f. Design. Desain perangkat lunak sebenarnya adalah proses multi langkah yang berfokus pada empat atribut sebuah program yang berbeda; struktur data, arsitektur perangkat lunak, representasi perangkat lunak yang dapat diperkirakan demi kualitas sebelum dimulai pemunculan kode.
- g. Generasi kode. Desain harus diterjemahkan kedalam bentuk mesin yang bisa dibaca, langkah pembuatan kode melakukan tugas ini jika desain dilakukan dengan cara yang lengkap, pembuatan kode dapat dilakukan secara mekanis.
- h. Test atau Pengujian. Sekali kode dibuat pengujian program dimulai. Proses pengujian berfokus pada logika internal perangkat lunak, memastikan bahwa semua pernyataan sudah diuji, dan pada eksternal fungsional yaitu mengarahkan pengujian untuk menemukan kesalahan-kesalahan dan memastikan bahwa input dibatasi akan memberikan hasil aktual yang sesuai dengan hasil dibutuhkan.

2. TINJAUAN PUSTAKA

2.1 Logika Fuzzy

2.1.1 Pengertian Fuzzy

Logika *fuzzy* adalah suatu cara yang tepat untuk memetakan suatu ruang *input* kedalam suatu ruang *output*. Titik awal dari konsep modern mengenai ketidakpastian adalah paper yang dibuat oleh Lofti A Zadeh (1965), dimana Zadeh memperkenalkan teori yang memiliki obyek-obyek dari *himpunan fuzzy* yang memiliki batasan yang tidak presisi dan keanggotaan dalam himpunan *fuzzy*, dan bukan dalam bentuk logika benar (*true*) atau salah (*false*), tapi dinyatakan dalam derajat (*degree*). Konsep seperti ini disebut dengan *Fuzziness* dan teorinya dinamakan *Fuzzy Set Theory*. *Fuzziness* dapat didefinisikan sebagai logika kabur berkenaan dengan semantik dari suatu kejadian, fenomena atau pernyataan itu sendiri. Seringkali ditemui dalam pernyataan yang dibuat

oleh seseorang, evaluasi dan suatu pengambilan keputusan

2.1.2 Konsep Kekaburan(*Fuzziness*) dan Sejarah Logika Fuzzy

Ketidaktegasan atau kekaburan merupakan salah satu ciri dari bahasa sehari-hari manusia untuk mengungkapkan konsep atau gagasan dalam berkomunikasi dengan orang lain. Pada taraf tertentu banyak kata atau istilah yang memuat salah satu bentuk kekaburan. Bentuk-bentuk kekaburan atau ketidakjelasan lainnya adalah:

6. Keambiguan (*ambiguity*), yang terjadi karena suatu kata atau istilah mempunyai makna ganda.
7. Keacakan (*randomness*), yaitu ketidakpastian mengenai sesuatu hal karena hal itu belum terjadi (akan terjadi).
8. Ketidakjelasan akibat tidak lengkapnya informasi yang ada (*incompleteness*).
9. Ketidaktepatan (*imprecision*) yang disebabkan oleh keterbatasan alat dan metode untuk mengumpulkan informasi.
10. Kekaburan semantik, yaitu kekaburan yang disebabkan karena makna dari suatu kata atau istilah tidak dapat didefinisikan secara tegas, misalnya cantik, tinggi, kaya, pintar dan sebagainya.

2.1.3 Himpunan Keanggotaan

Pada himpunan tegas (*crisp*), nilai keanggotaan suatu item x dalam suatu himpunan A , yang sering ditulis dengan $\mu_A[x]$, memiliki 2 kemungkinan (Kusumadewi S, Purnomo H, 2010) yaitu:

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

2.1.4 Fungsi Keanggotaan

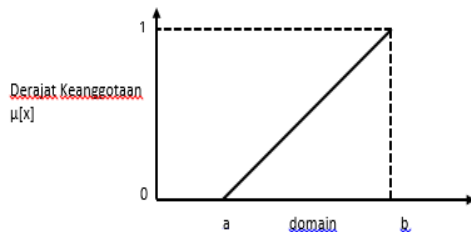
Fungsi keanggotaan adalah suatu kurva yang menunjukkan pemetaan titik-titik *input* data

kedalam nilai keanggotaannya (sering juga disebut dengan derajat keanggotaan) yang memiliki interval antara 0 sampai 1. Salah satu cara yang dapat digunakan untuk mendapatkan nilai keanggotaan adalah dengan melalui pendekatan fungsi. Apabila U menyatakan himpunan universal dan A adalah himpunan fungsi *fuzzy* dalam U , maka A dapat dinyatakan sebagai pasangan terurut[1]. Ada beberapa fungsi yang bisa digunakan.

b. Representasi Linear

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

Ada 2 keadaan himpunan *fuzzy* yang *linear*. Pertama, kenaikan himpunan dimulai pada nilai domain yang memiliki derajat keanggotaan nol (0) bergerak ke kanan menuju ke nilai domain yang memiliki derajat keanggotaan lebih tinggi[]. Seperti terlihat pada gambar 2.2.

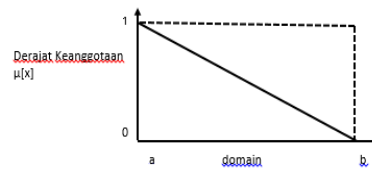


Gambar 2.2 Representasi linier naik

Fungsi keanggotaan:

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

Kedua, merupakan kebalikan dari yang pertama. Garis lurus dimulai dari nilai domain dengan derajat keanggotaan tertinggi pada sisi kiri, kemudian bergerak menurun ke nilai domain yang memiliki derajat keanggotaan lebih rendah. Seperti terlihat pada gambar 2.3.



Gambar 2.3 Representasi linier turun

Fungsi keanggotaan:

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

2.1.5 Operator Dasar Zadeh untuk Operasi Himpunan Fuzzy

Seperti halnya himpunan konvensional, ada beberapa operasi yang didefinisikan secara khusus untuk mengkombinasi dan memodifikasi himpunan *fuzzy*. Nilai keanggotaan sebagai hasil dari operasi 2 himpunan sering dikenal dengan nama *fire strength* atau α -predikat. Ada 3 operator dasar yang diciptakan oleh Zadeh (Kusumadewi, 2003), yaitu:

4) Operasi AND

Operator ini berhubungan dengan operasi interseksi pada himpunan. α -predikat sebagai hasil operasi dengan operator AND diperoleh dengan mengambil nilai keanggotaan terkecil antar elemen pada himpunan-himpunan yang bersangkutan.

$$\mu_{A \cap B} = \min(\mu_A[x], \mu_B[y])$$

5) Operasi OR

Operator ini berhubungan dengan operasi union pada himpunan. α -predikat sebagai hasil operasi dengan operator OR diperoleh dengan mengambil nilai keanggotaan terbesar antar elemen pada himpunan-himpunan yang bersangkutan.

$$\mu_{A \cup B} = \max(\mu_A[x], \mu_B[y])$$

6) Operasi NOT

Operator ini berhubungan dengan operasi komplemen pada

himpunan. α - predikat sebagai hasil operasi dengan operator NOT diperoleh dengan mengurangkan nilai keanggotaan elemen pada himpunan yang bersangkutan dari 1.

$$\mu_A = 1 - \mu_A[x]$$

3. ANALISA DAN PERANCANGAN SISTEM

3.1 Pembentukan Variabel dan Himpunan Fuzzy

3.1.1 Data Masukan (Input)

Jumlah variabel input yang digunakan sebanyak 4 variabel dimana setiap variabel mengandung 3 himpunan fuzzy sehingga total himpunan fuzzy yang digunakan adalah 12 himpunan. Dalam perancangan sistem fuzzy ini, menggunakan kurva bahu untuk himpunan rendah dan tinggi, sedangkan untuk himpunan sedang menggunakan kurva berbentuk segitiga.

Tabel 1 Tabel Inputan

No	Variabel	Himpunan	Parameter	Domain	
1	IPK	Rendah	0.5 - 2	(0-4)	Kurva Bahu
		Sedang	1.5 - 2.79		Segitiga
		Tinggi	2.5 - 4		Kurva Bahu
2	Penghasilan Orang Tua	Rendah	750000-1500000	(750000 - 4500000)	Kurva Bahu
		Sedang	1000000-3000000		Segitiga
		Tinggi	2500000-4500000		Kurva Bahu
3	Jumlah Tanggungan Orang Tua	Rendah	1-3	(1-7)	Kurva Bahu
		Sedang	2-5		Segitiga
		Tinggi	4-7		Kurva Bahu
4	Jumlah Sertifikat	Rendah	1-3	(1-7)	Kurva Bahu
		Sedang	2-5		Segitiga
		Tinggi	4-7		Kurva Bahu

3.1.2 Data Keluaran (Output)

Output yang dihasilkan oleh sistem fuzzy pemilihan beasiswa ini berupa hasil rekomendasi mahasiswa yang paling cocok untuk mendapatkan beasiswa yang disertai bobot.

3.2 Desain Sistem

Desain sistem merupakan pembuatan rancangan sistem fuzzy yang berkaitan dengan fungsionalitas. Berikut adalah beberapa hal yang dilakukan untuk membangun sistem fuzzy.

3.2.1 Struktur Database

Aplikasi beasiswa yang penulis buat ini menggunakan 4 tabel yaitu

tbl_mahasiswa, tbl_ppa, tbl_bobot_ppa, tbl_login. Adapun struktur dari masing-masing tabel adalah sebagai berikut :

a. Tabel mahasiswa

Tabel ini digunakan untuk menyimpan data-data mahasiswa yang akan diseleksi untuk mendapatkan beasiswa PPA. Adapun field-field yang terdapat pada tabel tbl_mahasiswa sebagai berikut

Tabel 2 Tabel Mahasiswa

Field	Type	Size	Key	Keterangan
Npm	Varchar	24	Primary	Npm Mahasiswa
nama_mahasiswa	Varchar	50		Nama Mahasiswa
Jk	Varchar	20		Jenis Kelamin
Ttl	Varchar	70		Tempat Tanggal Lahir
Alamat	Varchar	70		Alamat Mahasiswa
No_hp	Varichar	12		Nomor hp Mahasiswa
Jurusan_prodi	Varchar	40		Jurusan Mahasiswa
Agama	Varchar	15		Agama
Semester	Int	3		Semester
Nama_ayah	Varchar	40		Nama Ayah
Nama_ibu	Varchar	40		Nama Ibu
Ipk	Float			Jumlah IPK
Pekerjaan_ortu	Varchar	50		Pekerjaan Ortu
Penghasilan_ortu	Int	11		Penghasilan Ortu
Jumlah_tanggungan	Int	11		Jumlah Tanggungan Ortu
Jumlah_Sertifikat	Float			Jumlah Sertifikat

b. Tabel tbl_ppa

Tabel ini digunakan untuk menyimpan hasil query beasiswa PPA dari masing-masing kriteria. Adapun field-field dalam tabel tbl_ppa ini adalah sebagai berikut

Tabel 3 Tabel PPA

Field	Type	Size	Key	Keterangan
Npm	Varchar	24	Primary	Npm Mahasiswa
Nama	Varchar	70		Nama Mahasiswa
Ipk	Float	0		Skor Ipk Kumulatif
Penghasilan	Float	0		Skor Penghasilan Ortu

Tanggungan	Float	0		Skor Tanggungan Ortu
Sertifikat	Float	0		Skor Jumlah Sertifikat

c. Tabel Bobot

Tabel ini adalah untuk menyimpan bobot dari masing-masing kriteria yang akan digunakan untuk proses perhitungan metode fuzzy database tahani beasiswa PPA. Adapun field yang terkandung pada tabel tbl_bobot_ppa ini adalah sebagai berikut :

Tabel 4 Tabel Bobotipk

Field	Type	Size	Key	Keterangan
Id	Int	11	Primary	Id bobot mahasiswa
Npm	Varchar	24		Npm Mahasiswa
Rendah	Float	0		Rendah
Sedang	Float	0		Sedang
Tinggi	Float	0		Tinggi

Tabel 5 Tabel BobotPekerjaan

Field	Type	Size	Key	Keterangan
Id	Int	11	Primary	Id bobot mahasiswa
Npm	Varchar	24		Npm Mahasiswa
Rendah	Float	0		Rendah
Sedang	Float	0		Sedang
Tinggi	Float	0		Tinggi

Tabel 6 Tabel Bobotpenghasilan

Field	Type	Size	Key	Keterangan
Id	Int	11	Primary	Id bobot mahasiswa
Npm	Varchar	24		Npm Mahasiswa
Rendah	Float	0		Rendah
Sedang	Float	0		Sedang
Tinggi	Float	0		Tinggi

Tabel 7 Tabel Bobottanggungan

Field	Type	Size	Key	Keterangan
Id	Int	11	Primary	Id bobot mahasiswa
Npm	Varchar	24		Npm Mahasiswa
Rendah	Float	0		Rendah

Sedang	Float	0		Sedang
Tinggi	Float	0		Tinggi

Tabel 8 Tabel Bobotsertifikat

Field	Type	Size	Key	Keterangan
Id	Int	11	Primary	Id bobot mahasiswa
Npm	Varchar	24		Npm Mahasiswa
Rendah	Float	0		Rendah
Sedang	Float	0		Sedang
Tinggi	Float	0		Tinggi

c. Tabel tbl_user

Tabel ini adalah untuk menyimpan nama Admin dan Operator yang akan login kedalam sistem. Adapun field-field dari tabel tbl_login ini adalah sebagai berikut

Tabel 9 Tabel user

Field	Type	Size	Key	Keterangan
Id_user	Int	40	Primary	ID
Nama	Varchar	255		Nama Pengguna
Username	Varchar	255		Username Admin/Operator
Password	Varchar	255		Password
Jenis_akun	Varchar	255		Jenis Akun

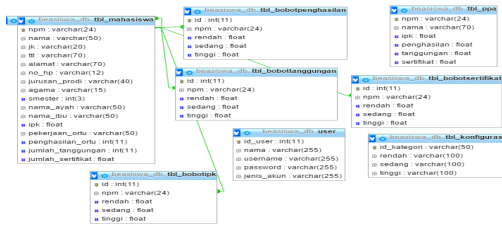
d. Tabel Konfigurasi

Tabel ini adalah untuk menyimpan batas kriteria data yang ada pada semua table. Adapun field-field dari tabel tbl_login ini adalah sebagai berikut

Tabel 10 Tabel Konfigurasi

Field	Type	Size	Key	Keterangan
Id_kategori	Varchar	50	Primary	Id kategori
Rendah	Varchar	100		Rendah
Sedang	Varchar	100		Sedang
Tinggi	Varchar	100		Tinggi

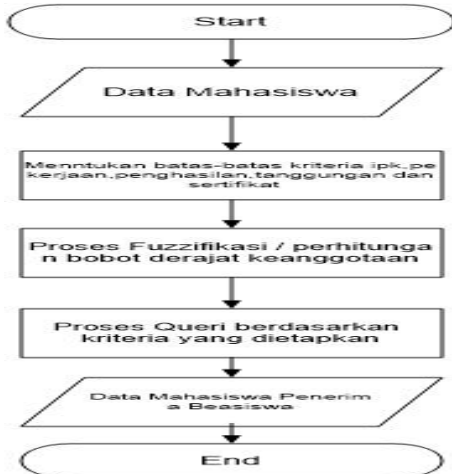
3.3 Relasi Database



Gambar 1 Relasi database

3.4 Desain Diagram Sistem

Sistem adalah sekumpulan elemen yang saling berkaitan dan saling mempengaruhi dalam hal melakukan kegiatan bersama untuk mencapai suatu tujuan. Salah satu desain sistem yang saya buat ini adalah desain sistem Logika Fuzzy Tahani untuk menentukan penerima beasiswa PPA yang dimulai dari memasukkan data mahasiswa, menentukan batas-batas kriteria IPK, Pekerjaan Ortu, Penghasilan Ortu, Tanggungan Ortu dan Jumlah Sertifikat, Proses Fuzzyfikasi atau perhitungan bobot derajat keanggotaan, Proses query berdasarkan kriteria yang ditetapkan dan terakhir data mahasiswa penerima beasiswa.



Gambar 2 Desain sistem

3.5 Rules Fuzzy

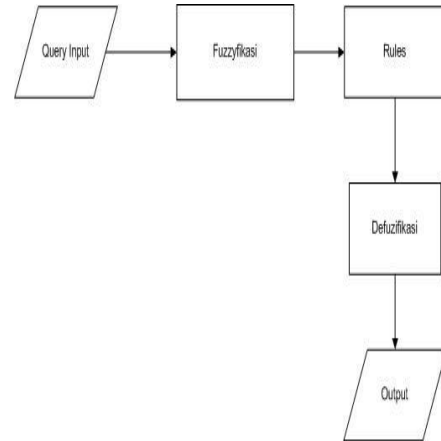
Berikut ini rule fuzzy sebagai berikut :

Tabel 11 Rule Fuzzy

Ket	Rule
R1	If IPK tinggi and Penghasilan rendah and Tanggungan rendah and Sertifikat tinggi then Ya
R2	If IPK tinggi and Penghasilan rendah and Tanggungan rendah and Sertifikat sedang then Ya
R3	If IPK tinggi and Penghasilan rendah and Tanggungan rendah and Sertifikat rendah then Ya
R4	If IPK tinggi and Penghasilan rendah and Tanggungan tinggi and Sertifikat rendah then Ya

R5	If IPK tinggi and Penghasilan rendah and Tanggungan sedang and Sertifikat rendah then Ya
R6	If IPK tinggi and Penghasilan tinggi and Tanggungan rendah and Sertifikat rendah then Ya
R7	If IPK tinggi and Penghasilan sedang and Tanggungan rendah and Sertifikat rendah then Ya
R8	If IPK tinggi and Penghasilan rendah and Tanggungan rendah and Sertifikat rendah then Ya

3.6 Arsitektur Sistem Fuzzy Tahani



Gambar 3 Arsitektur System Fuzzy Tahani

3.7 Perhitungan Fuzzy Tahani

3.7.1 Menghitung Fungsi Keanggotaan

Dalam perhitungan Fuzzy Tahani dengan kasus beasiswa saya membutuhkan lima variabel yaitu IPK, Pekerjaan Ortu, Penghasilan Ortu, Tanggungan Ortu dan Jumlah Sertifikat dengan Empat kategori pada masing-masing Variabel.

1 IPK (Indeks Prestasi Kumulatif)

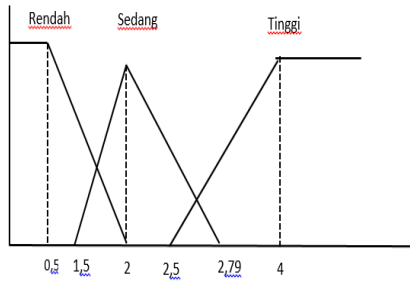
Pada Variabel IPK menggunakan tiga himpunan yaitu Rendah, Sedang dan Tinggi. Berikut tiga himpunan pada variabel IPK :

Rendah = $0.5 - 2$

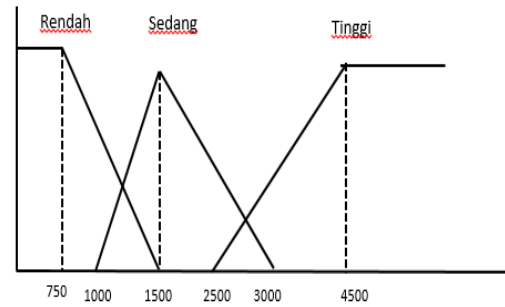
Sedang = $1.5 - 2.79$

Tinggi = $2.5 - 4$

Dari range IPK diatas dapat dibuat fungsi keanggotaan terlihat pada gambar dibawah ini :



Gambar 4 Grafik IPK (Indeks Prestasi Akademik) Fungsi Keanggotaan



Gambar 5 Grafik Penghasilan Orang Tua Fungsi Keanggotaan

$$Rendah = \begin{cases} 1 & a \leq 0,5 \\ \frac{(2-a)}{0,5} & 0,5 \leq a \leq 2 \\ 0 & a > 2 \end{cases} \quad Tinggi = \begin{cases} 0 & 2,5 \leq a \text{ atau } a \geq 4 \\ \frac{(a-2,5)}{1,25} & 2,5 \leq a \leq 4 \\ 1,5 & a \geq 4 \end{cases}$$

$$Sedang = \begin{cases} 0 & 1,5 \leq a \text{ atau } a > 2,79 \\ \frac{(a-1,5)}{0,5} & 1,5 \leq a \leq 2 \\ \frac{(2,79-a)}{0,75} & 2 \leq a \leq 2,79 \end{cases}$$

$$Rendah = \begin{cases} 1 & a < 750.000 \\ \frac{(1.500.000-a)}{750.000} & 750.000 \leq a \leq 1.500.000 \\ 0 & a \geq 1.500.000 \end{cases} \quad Tinggi = \begin{cases} 0 & 1.000.000 < a \text{ atau } a > 3.000.000 \\ \frac{(a-1.000.000)}{0,5} & 1.000.000 \leq a \leq 1.500.000 \\ \frac{(3.000.000-a)}{1.500.000} & 1.500.000 \leq a \leq 3.000.000 \end{cases}$$

2 Penghasilan

Pada Variabel Penghasilan menggunakan tiga himpunan yaitu Rendah, Sedang dan Tinggi. Berikut tiga himpunan pada variabel Penghasilan :

Rendah = 750000 - 1500000

Sedang = 1000000 - 3000000

Tinggi = 2500000 - 4500000

Dari range Penghasilan diatas dapat dibuat fungsi keanggotaan terlihat pada gambar dibawah ini :

3 Tanggungan

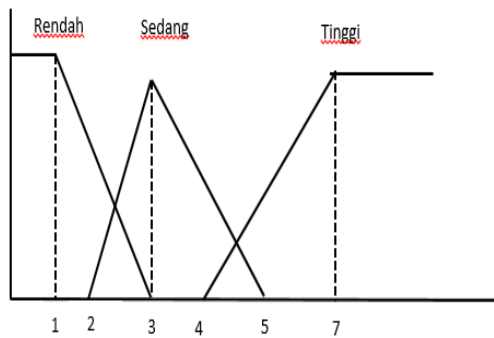
Pada Variabel Tanggungan menggunakan tiga himpunan yaitu Rendah, Sedang dan Tinggi. Berikut tiga himpunan pada variabel tanggungan :

Rendah = 1 - 3

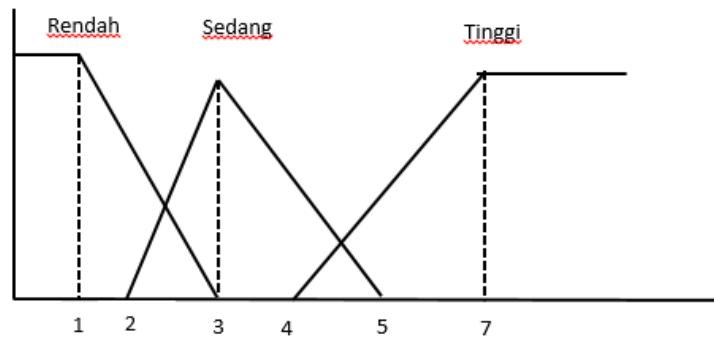
Sedang = 2 - 5

Tinggi = 4 - 7

Dari range Tanggungan diatas dapat dibuat fungsi keanggotaan terlihat pada gambar dibawah ini :



Gambar 6 Grafik Jumlah Tanggungan Orang Tua



Gambar 7 Grafik Jumlah Sertifikat

Fungsi Keanggotaan

$$Rendah = \begin{cases} 1 & 1 \leq a \\ \frac{(a-1)}{2} & 1 \leq a \leq 3 \\ 0 & a \geq 3 \end{cases} \quad Tinggi = \begin{cases} 0 & a \leq 4 \\ \frac{(a-4)}{1} & 4 \leq a \leq 7 \\ 1 & a \geq 7 \end{cases}$$

$$Sedang = \begin{cases} 0 & a \leq 2 \text{ atau } a \geq 5 \\ \frac{(2-2)}{1} & 2 \leq a \leq 3 \\ \frac{(5-a)}{2} & 3 \leq a \leq 5 \end{cases}$$

$$Sedang = \begin{cases} 0 & a \leq 2 \text{ atau } a \geq 5 \\ \frac{(2-2)}{1} & 2 \leq a \leq 3 \\ \frac{(5-a)}{2} & 3 \leq a \leq 5 \end{cases}$$

4 Jumlah sertifikat

Pada Variabel jumlah sertifikat menggunakan tiga himpunan yaitu Rendah, Sedang dan Tinggi. Berikut tiga himpunan pada variabel jumlah sertifikat :

Rendah = 1 - 3

Sedang = 2 - 5

Tinggi = 4 - 7

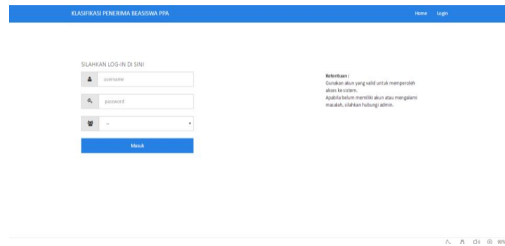
Dari range Jumlah sertifikat diatas dapat dibuat fungsi keanggotaan terlihat pada gambar dibawah ini :

4. HASIL DAN PEMBAHASAN

4.1 Tampilan Program

4.1.1 Halaman login

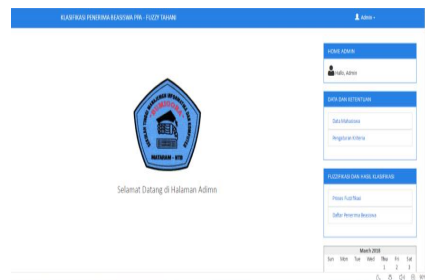
Halaman ini adalah tampilan awal dari sistem. Halaman login digunakan oleh pengguna untuk masuk ke sistem penerapan logika fuzzy tahani pada penerima beasiswa PPA. Berikut tampilan halaman login :



Gambar 1 Halaman login

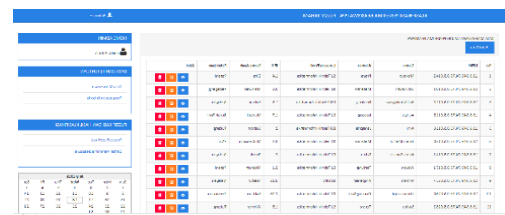
4.1.2 Halaman beranda

Halaman beranda adalah halaman yang dituju jika pengguna berhasil login.



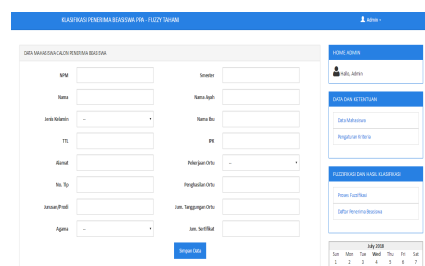
Gambar 9 Halaman beranda

4.1.3 Halaman tampil data mahasiswa



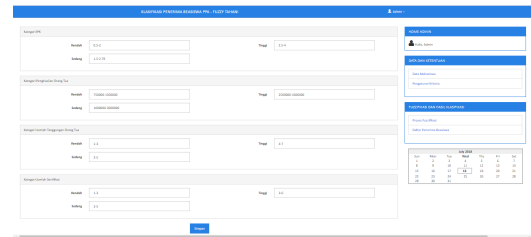
Gambar 2 Halaman tampil data mahasiswa

4.1.4 Halaman input data mahasiswa



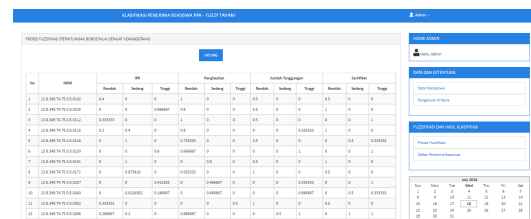
Gambar 11 Tampilan input data mahasiswa

4.1.5 Halaman pengaturan kriteria



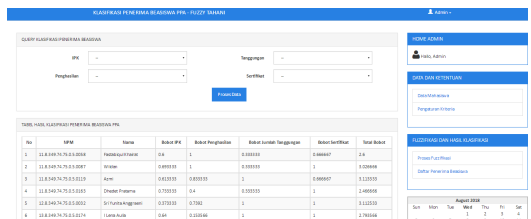
Gambar 12 Pengaturan kriteria

4.1.6 Halaman Fuzzifikasi



Gambar 13 Halaman fuzzifikasi

4.1.7 Halaman tampil penerima beasiswa



Gambar 14 Halaman tampil perhitungan penerima beasiswa

4.2 Tahap Pengujian

Pengujian dilakukan dengan menggunakan data asli untuk pendaftar beasiswa PPA tahun 2014. Tujuan pengujian adalah bagaimana hasilnya menggunakan metode fuzzy tahani pada penerimaan beasiswa 2014. Jumlah pendaftar beasiswa 2014 adalah 64 mahasiswa. Sistem penerapan fuzzy tahani pada penerimaan beasiswa PPA 2014 dapat dinilai baik menggunakan

$$\text{Rumus : } \text{Akurasi} = \frac{\text{Hasil Dengan Sistem}}{\text{Hasil Dengan data}} \times 100$$

Data Pendaftar Beasiswa 2014

Tabel 12 Data Pendaftar Beasiswa

N o	NPM	Nama	Sem ester	IP K	Penghasilan	Tangg ungan	Juml ah Serti fikat
1	11.8.349.74.75.0.5.0163	DHEDET PRATAMA	VI	3,6	Rp.1.200.000	5	10

				0			
2	11.8.349.74. 75.0.5.0053	AGUS PARHAN SURIAZ	VI	3, 5 0	Rp.1.012.15 0	3	4
3	12.8.349.74. 75.0.5.0164	AGUSFIAN HADI SAPUTRA	IV	3, 3 3	Rp. 450.000	3	
4	13.8.349.74. 75.0.5.0106	ROBIUZZA MAN	II	3, 5 0	Rp.4.620.10 0	3	1
5	11.8.349.72. 74.0.3.0002	BAIQ. WIDYA MAYA FEBRIANA	VI	3, 2 4	Rp.1.019.00 0	3	2
6	13.8.349.74. 75.0.5.0045	LALU MUH. KUMBAYO NI	II	3, 7 2	Rp.500.000	2	1
7	13.8.349.74. 75.0.5.0040	FRANSDIK A DHARMAW AN	II	3, 9 2	Rp.Rp.2.200 .000	3	1
8	12.8.349.32. 53.0.2.0005	EKA NURLAILA WARZUKI	IV	3, 7 2	Rp.750.000	2	4
9	12.8.349.74. 75.0.5.0174	ANDI ARISKA NOVIANTI	IV	3, 2 9	Rp. 500.000	3	2
10	13.8.349.74. 75.0.5.0130	IMAM MARUF	II	3, 6 4	*****R p.3.206.400	2	1
11	11.8.349.74. 75.0.5.0006	BILI NATA NAJWA	VI	3, 6 2	Rp.2.000.00 0	4	
12	11.8.349.74. 75.0.5.0002	MUHAMMA D SYAKIR	VI	3, 6 2	Rp.500.000	3	
13	11.8.349.74. 75.0.5.0119	AZMI	VI	3, 4 2	Rp.1.750.00 0	4	5
14	11.8.349.74. 75.0.5.0087	WILDAN	VI	3, 5 4	Rp.1.500.00 0	5	9
15	11.8.349.74. 75.0.5.0011	SITI AGRIPIA ALODIA YUSUF	VI	3, 4 4	Rp.4.011.20 4	4	3
16	11.8.349.74. 75.0.5.0275	NINSI PUTRIANA DEVI	VI	3, 0 0	Rp.600.000	4	4
17	13.8.349.74. 75.0.5.0217	NURHANDI NI	II	3, 4 7	Rp.3.357.90 0	2	
18	11.8.349.74. 75.0.5.0247	DEVIANA AYU LESTARI	VI	3, 2 9	Rp.1.527.93 0	4	
19	13.8.349.74. 75.0.5.0066	FENI ALFAONIT A	II	3, 5 6	Rp.1.000.00 0	3	
20	13.8.349.74. 75.0.5.0127	FIRDAUS TRI ENDAH WAYUNING SIH	II	3, 4 2	Rp.4.668.97 5	3	3
21	13.8.349.74. 75.0.5.0133	LALU ABD RAHMAN HAKIM	II	3, 5 8	Rp.1.071.50 0	1	4
22	11.8.349.74. 75.0.5.0056	TRIPUSNA WATI	VI	3, 0 9	Rp.900.000	2	2
23	12.8.349.74. 75.0.5.0142	KURTUBIA H	IV	3, 1 9	Rp.750.000	4	
24	13.8.349.74. 75.0.5.0138	SUNANDA AFRIANTO	II	3, 2 2	Rp.1.200.00 0	1	

25	12.8.349.74. 75.0.5.0196	ZUHRI JULIARDI	IV	3, 4 2	Rp.1.910.90 0	2	1
26	13.8.349.74. 75.0.5.0075	HAIRUL IMAM	II	3, 7 2	Rp.3.327.20 0	1	1
27	12.8.349.74. 75.0.5.0193	AHMAD MEIRADI	IV	3, 4 3	Rp.900.000	5	1
28	11.8.349.74. 75.0.5.0238	FERRY AGUSMANS YAH MA'AS	VI	3, 2 7	Rp.2.000.00 0	4	3
29	11.8.349.74. 75.0.5.0071	ABDUL AZIZ ALI	VI	3, 2 1	Rp.1.500.00 0	2	10
30	11.8.349.74. 75.0.5.0120	LALU HENDIAWA N DIPA	VI	3, 5 4	Rp.821.726	2	3
31	12.8.349.32. 53.0.3.0018	DWI RAGITA MULYAWA TI	IV	3, 5 2	Rp.3.693.53 1	4	6
32	13.8.349.74. 75.0.5.0173	DINDARI BELA QUR'ANI	II	3, 4 7	Rp.700.000	2	2
33	12.8.349.74. 75.0.5.0221	ANDY HARDIYAN TI HASTUTI	IV	3, 7 5	Rp.2.770.00 0	4	
34	12.8.349.74. 75.0.5.0169	B. DINDA USWATUN HASANAH	IV	3, 3 7	Rp.4.355.20 0	2	1
35	12.8.349.74. 75.0.5.0032	SRI YUNITA ANGGRAEN I	IV	3, 0 6	Rp.1.891.20 0	4	3
36	13.8.349.74. 75.0.5.0008	GALUH WAHYUNI FAZRI	IV	3, 4 3	Rp.1.766.60 0	3	
37	13.8.349.74. 75.0.5.0033	AHMAD SYAHBAND I	II	3, 8 6	Rp.700.000	1	
38	13.8.349.74. 75.0.5.0112	SATRIADI KAWIRIAN	II	3, 5 6	Rp.500.000	4	
39	11.8.349.74. 75.0.5.0022	MOH. ZAINUDDIN	VI	3, 1 1	Rp.300.000	1	3
40	13.8.349.32. 53.0.2.0011	MUHAMMA D ZAKI AZHARI	II	3, 2 1	Rp.700.000	3	
41	11.8.349.74. 75.0.5.0006	MUH. RIDHA FADLI	VI	3, 0 0	Rp.3.225.40 0	2	2
42	11.8.349.74. 75.0.5.0090	MUH. ERWIN INDRAWAN	VI	3, 4 8	Rp.600.000	2	7
43	13.8.349.74. 75.0.5.0174	I LENA AULIA	II	3, 3 6	Rp.2.769.65 1	4	10
44	13.8.349.32. 53.0.2.0022	HALIMATU SSAKDIAH	II	3, 2 4	Rp.500.000	2	
45	13.8.349.74. 75.0.5.0220	ANA ROHMIATI	II	3, 1 3	Rp.3.712.37 3	3	
46	12.8.349.32. 53.0.2.0025	MUH. KHOLIS KAWIRYA PUTRA	IV	3, 3 4	Rp.3.331.90 0	2	2
47	11.8.349.74. 75.0.5.0058	FASTABIQ L KHAIRAT	VI	3, 4 0	Rp.1.500.00 0	5	5
48	11.8.349.74. 75.0.5.0133	LALU ZAMRUD ZULBAINI	VI	3, 2 9	Rp.2.738.50 0	3	4

		KARNA					
49	12.8.349.74. 75.0.5.0126	SHOFIAN ANSHORI	IV	3, 3 7	Rp.600.000	4	1
50	12.8.349.74. 75.0.5.0098	AHMAD MUHAR DIAN LASMITA	IV	3, 3 4	Rp.500.000	2	
51	12.8.349.74. 75.0.5.0106	MUHAMAD AZWAR	IV	3, 3 8	Rp.500.000	4	
52	11.8.349.74. 75.0.5.0048	HERIJNA APRIANA	VI	3, 1 6	Rp.700.000	2	5
53	13.8.349.0.7 4.75.0.5.0205	LALU YOGI PRATAMA	II	3, 8 1	Rp.3.099.20 0	2	
54	13.8.349.74. 75.0.5.0164	RUDI HENDRAW AN	II	3, 5 8	Rp.1.000.00 0	3	
55	12.8.349.74. 75.0.5.0005	RADEN AGUNG BAGUS WIRO KUSUMO	IV	3, 6 9	Rp.2.000.00 0	3	
56	13.8.349.74. 75.0.5.0009	WINDA YUSNITA	II	3, 4 0	Rp.1.917.10 0	1	
57	13.8.349.74. 75.0.5.0197	DANI ARISTYAW AN	II	3, 1 7	Rp.2.518.20 0	3	
58	12.8.349.74. 75.0.5.0002	SULPAN	IV	3, 2 8	Rp.1.920.00 0	2	
59	13.8.349.74. 75.0.5.0128	PUTRI AISYA GUSTI	II	3, 4 2	Rp.2.280.00 0	1	4
60	12.8.349.74. 75.0.5.0012	BQ. NURASMA H	IV	3, 3 4	Rp.700.000	6	
61	13.8.349.74. 75.0.5.0207	HAYATULL OH FIRMAN HADI	II	3, 5 0	Rp.1.500.00 0	3	
62	12.8.349.74. 75.0.5.0070	BAIQ ELA NOPITA LESTARI	IV	3, 2 9	Rp.1.200.00 0	2	
63	12.8.349.74. 75.0.5.0006	BAHRUL HIDAYAT	IV	3, 6 3	Rp.2.769.65 1	3	
64	12.8.349.74. 75.0.5.0076	WISNU MAULANA	IV	3, 6 2	Rp.2.000.00 0	1	

	53.0.2.0005	WARZ UKI	WARZ UKI		2	00		
4	12.8.349.74. 75.0.5.0174	ANDI ARISK A NOPIA NTI	ANDI ARISK A NOPIA NTI	4	3, 2 9	5.00.0 00	5	1
5	13.8.349.74. 75.0.5.0066	FENI ALFAO NITA	-	2	3, 5 6	1.000. 000	3	2
6	13.8.349.74. 75.0.5.0133	LALU ABD RAHM AN HAKIM	LALU ABD RAHM AN HAKI M	2	3, 5 8	1.071. 500	2	1
7	13.8.349.74. 75.0.5.0075	HAIRU L IMAM	HAIRU L IMAM	6	3, 7 2	3.327. 200	1	1
8	11.8.349.74. 75.0.5.0120	LALU HENDI AWAN DIPA	LALU HENDI AWAN DIPA	6	3, 4 5	821.7 26.36	4	4
9	12.8.349.32. 53.0.3.0018	DWI RAGIT A MULY AWATI	-	4	3, 5 2	3.693. 246	4	2
10	12.8.349.74. 75.0.5.0169	B. DINDA USWAT UN HASAN AH	B. DINDA USWA TUN HASA NAH	4	3, 3 7	4.355. 200	2	1
11	13.8.349.74. 75.0.5.0033	AHMA D SYAHB ANDI	AHMA D SYAH BANDI	2	3, 8 6	700.0 00	1	
12	13.8.349.74. 75.0.5.0112	SATRI ADI KAWIR IAN	SATRI ADI KAWI RIAN	2	3, 5 6	500.0 00	4	
13	11.8.349.74. 75.0.5.0090	MoH. ERWIN INDRA WAN	MoH. ERWI N INDRA WAN	6	3, 4 8	600.0 00	2	5
14	13.8.349.74. 75.0.5.0205	LALU YOGI PRATA MA	LALU YOGI PRAT AMA	2	3, 8 1	3.099. 200	2	9
15	13.8.349.74. 75.0.5.0045	LALU MUH. KUMB AYONI	LALU MUH. KUMB AYONI	2	3, 7 2	500.0 00	2	3

Hasil Perhitungan :

$$\begin{aligned}
 \text{Akurasi} &= \frac{\text{Hasil Dengan Sistem}}{\text{Hasil Dengan data}} \times 100 \\
 &= \frac{11}{15} \times 100 \\
 &= 73.3
 \end{aligned}$$

5.PENUTUP

5.1 Kesimpulan

Dari hasil sistem penerimaan beasiswa tahun 2014 di STMIK Bumigora Mataram menggunakan metode fuzzy tahani, penulis dapat menuliskan kesimpulan sebagai berikut:

Data Pendaftar Penerima Beasiswa 2014

Tabel 13 Data Pendaftar Penerima Beasiswa dengan manual dan hasil sistem

N o	NPM	Nama	Nama Ujicoba dengan Sistem	Sem ester	IP K	Pengh asilan	Tangg ungan	Jum lah Serti fikat
1	11.8.349.74. 75.0.5.0163	DHEDE T PRATA MA	-	6	3, 6 0	1.200. 000	5	10
2	13.8.349.74. 75.0.5.0040	FRANS DIKA DHAR MAWA N	-	2	3, 9 2	2.200. 000	3	4
3	12.8.349.32.	EKA NURLA ILA	EKA NURL AILA	4	3, 7	750.0	4	

Sistem Pendukung Keputusan untuk penentuan hasil beasiswa di STMIK Bumigora Mataram yang dibangun dengan metode Fuzzy Tahani memberikan akurasi hingga 73.3%.

5.2 Saran

Dalam penelitian ini masih banyak kekurangan, diantaranya adalah proses input data pendaftar masih dilakukan oleh admin, rentang nilai yang kecil memungkinkan total nilai satu atau dua alternatif menjadi sama, Oleh karena itu bagi yang nantinya ingin melakukan pengembangan dari penelitian ini dapat menambahkan fitur yang lebih lengkap dalam input data, input data dilakukan oleh pendaftar penerima beasiswa sendiri, memberikan rentang nilai yang lebih agar kemungkinan total nilai yang sama menjadi lebih kecil, dan menggunakan metode yang menghasilkan hasil yang lebih akurat.

DAFTAR PUSTAKA

- [1]Deddy Irwan : Implementasi Fuzzy Query Pada Database Untuk Rekomendasi Beasiswa. 2008. USU Repository.
- [2]Kadir, Abdul. (2008). Belajar database menggunakan MySQL. Yogyakarta: Penerbit Andi.
- [3]Kadir, Abdul.(2001). Dasar Pemrograman Web Dinamis Menggunakan PHP.Yogyakarta: Penerbit Andi
- [4]Kusumadewi S, Purnomo H. 2004 Fuzzy Database
<http://repository.usu.ac.id/ChapterII> , di akses 25 Januari 2017.
- [5]Kusumadewi S, Purnomo H. 2004 Pengertian Fuzzy Tahani.
<http://repository.usu.ac.id/ChapterII> , di akses 25 Januari 2017.
- [6]Simarmata, Janner. (2009). Rekayasa Web. Medan: Penerbit Andi.
- [7]Thomas Sri Widodo, 2005, “Sistem Neuro Fuzzy”, Graha Ilmu, Yogyakarta.
- [8]Ucoe Poe. 2016. Pengertian Codeigniter.
<http://www.ucoepoe.com/2016/05/pengertian-codeigniter-fungsi-kelebihan-dan-kekurangannya.html>. di akses 20 agustus 2017.