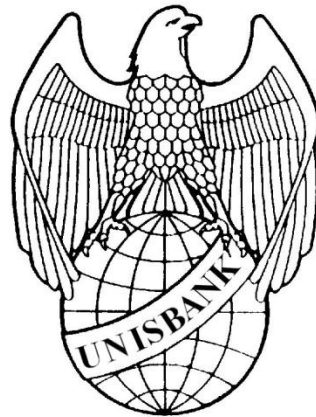


# **SISTEM PENDUKUNG KEPUTUSAN UNTUK MEMILIH BANK DENGAN LOGIKA FUZZY**

Tugas Akhir disusun untuk memenuhi syarat  
Mencapai gelar Kesarjanaan Komputer pada  
Program Studi Teknik Informatika  
Jenjang Program Strata-1



oleh:

**RIFKI WARTA PRADITYA**  
08.01.53.0095

10432

**FAKULTAS TEKNOLOGI INFORMASI  
UNIVERSITAS STIKUBANK (UNISBANK)**

**SEMARANG**

**2013**

## PERNYATAAN

Dengan ini saya selaku mahasiswa Fakultas Teknologi Informasi Universitas Stikubank:

**Nama : RIFKI WARTA PRADITYA**

**NIM : 08.01.53.0095**

**Jenjang Program Studi : Strata 1**

setuju / tidak setuju \*) naskah ringkasan penelitian (calon naskah berkala penelitian Sarjana) yang disusun oleh yang bersangkutan setelah mendapat arahan dari pembimbing, dipublikasikan **dengan / tanpa \***) mencantumkan nama tim pembimbing sebagai *co-author*.

Kemudian harap maklum.

Semarang, 15 Maret 2013

Yang bersangkutan,



Rifki Warta P  
08.01.53.0095

Mengetahui,

Pembimbing I

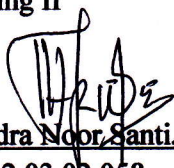


Dwi Agus Diartono, S.Kom, M.Kom

NIY: Y.2.90.03.054

Tanggal: 15 Maret 2013

Pembimbing II



Rina Candra Noor Santi, S.Pd, M.Kom

NIY: YU.2.03.02.058

Tanggal: 15 Maret 2013

\*) coret yang tidak perlu

## HALAMAN PENGESAHAN

Telah dipertahankan di depan tim dosen penguji Tugas Akhir Fakultas Teknologi Informasi, Universitas STIKUBANK (UNISBANK) Semarang dan diterima sebagai salah satu syarat guna menyelesaikan jenjang Program Strata 1, Program Studi Teknik Informatika.

Semarang, 8 Maret 2013

Ketua



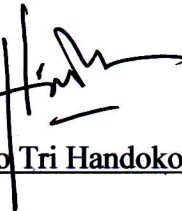
(Dwi Agus Diartono, S.Kom, M.Kom)

Sekretaris



(Rina Candra Noor Santi, S.Pd, M.Kom)

Anggota



(Drs. Widiyanto Tri Handoko, M.Kom)

MENGETAHUI :  
UNIVERSITAS STIKUBANK (UNISBANK) SEMARANG

Fakultas Teknologi Informasi  
Dekan

Dwi Agus Diartono, S.Kom, M.Kom

## **MOTTO DAN PERSEMBAHAN**

### **MOTTO**

- ✎ Hidup adalah perjuangan
- ✎ Sabar menghadapi masalah dan bersyukur merupakan salah satu pedoman hidup manusia
- ✎ Orang yang bahagia adalah orang yang dijauhkan dari fitnah dan bila dtimpa ujian serta cobaan ia selalu bersabar
- ✎ Lupakan kesempurnaan, dan cobalah mengejar kesempurnaan
- ✎ Sesungguhnya setelah mengalami kesulitan selalu ada jalan kemudahan
- ✎ Tiada hari tanpa instropeksi diri
- ✎ buku merupakan jendela informasi dunia

### **PERSEMBAHAN**

1. Allah S.W.T
2. Teman-teman Teknik Informatika.
3. Sahabat–sahabat dan saudara –saudara yang memberi semangat.

**FAKULTAS TEKNOLOGI INFORMASI  
UNIVERSITAS STIKUBANK (UNISBANK) SEMARANG**

Program Studi : Teknik Informatika  
Tugas Akhir Sarjana Komputer  
Semester Ganjil Tahun 2013

**SISTEM PENDUKUNG KEPUTUSAN UNTUK MEMILIH BANK DENGAN LOGIKA  
FUZZY**

**Rifki Warta Praditya  
NIM : 08.01.53.0095**

**Abstrak**

Bisnis perbankan merupakan bisnis jasa yang berdasarkan pada azas kepercayaan perlu diiringi oleh perkembangan teknologi komputer untuk dapat memenuhi kebutuhan para nasabah sehingga masalah kualitas layanan menjadi faktor yang sangat menentukan dalam keberhasilan usaha dimana nasabah selalu memiliki pertimbangan atau faktor-faktor sebelum mengambil suatu keputusan. Untuk dapat membantu hal tersebut, perlu didukung ke dalam suatu sistem pendukung keputusan berbasis komputerisasi. Sistem pendukung keputusan (*decision support system*) selain dapat memberikan informasi juga dapat membantu menyediakan berbagai alternatif yang dapat ditempuh dalam proses pengambilan keputusan.

Adapun tujuan yang akan dicapai adalah untuk membuat aplikasi berbasis sistem pendukung keputusan untuk rekomendasi untuk memilih Bank menggunakan metode *Fuzzy* database model Tahani.

Hasil dalam penelitian ini adalah membuat suatu sistem yang dapat membantu para pembuat keputusan untuk menentukan solusi pemilihan Bank yang optimal berupa grafik rekomendasi dalam pemilihan Bank dengan menggunakan Logika *Fuzzy* Tahani.

**Kata Kunci**

Sistem Pendukung Keputusan, Logika *Fuzzy* Tahani, Bank

Semarang : 7 Pebruari 2013

Pembimbing I

Pembimbing II

(Dwi Agus Diartono, S.Kom, M.Kom)

(Rina Candra Noor Santi, S.Pd, M.Kom)

## KATA PENGANTAR

Dengan memanjatkan puji syukur kehadirat Allah SWT. yang telah melimpahkan segala rahmat, hidayah dan inayah-Nya kepada penulis sehingga laporan tugas akhir dengan judul **“SISTEM PENDUKUNG KEPUTUSAN UNTUK MEMILIH BANK DENGAN LOGIKA FUZZY”** dapat penulis selesaikan sesuai dengan rencana karena dukungan dari berbagai pihak yang tidak ternilai besarnya. Oleh karena itu penulis menyampaikan terimakasih kepada:

1. Dr. Bambang Suko Priyono, MM selaku Rektor Universitas Stikubank Semarang.
2. Dwi Agus Diartono, S.Kom, M.Kom, selaku Dekan Fakultas Teknologi Informasi.
3. Dewi Handayani UN, S.Kom, M.Kom selaku Ka. ProgdI Teknik Informatika.
4. Dwi Agus Diartono, S.Kom, M.Kom selaku Pembimbing I dan Rina Candra Noor Santi, S.Pd, M.Kom selaku Pembimbing II yang telah membantu dan memberikan bimbingan serta pengarahan dalam penulisan Laporan Tugas Akhir ini.
5. Dosen-dosen pengampu di Program Studi Teknik Informatika Fakultas Teknologi Informasi Universitas Stikubank Semarang yang telah memberikan ilmu dan pengalamannya masing-masing, sehingga penulis dapat mengimplementasikan ilmu yang telah disampaikan.

Semoga Allah SWT memberikan balasan yang lebih besar kepada beliau-beliau, dan pada akhirnya penulis berharap bahwa penulisan laporan tugas akhir ini dapat bermanfaat dan berguna sebagaimana fungsinya.

Semarang, Pebruari 2013

Penulis

## DAFTAR ISI

HALAMAN JUDUL.....	i
HALAMAN PERSETUJUAN.....	ii
HALAMAN PENGESAHAN.....	iii
MOTTO DAN PERSEMBAHAN .....	iv
ABSTRAKSI .....	v
KATA PENGANTAR .....	vi
DAFTAR ISI.....	vii
DAFTAR TABEL.....	xi
DAFTAR GAMBAR .....	xii
<b>BAB I     PENDAHULUAN</b>	
1.1 Latar Belakang Masalah.....	1
1.2 Perumusan Masalah .....	4
1.3 Pembatasan Masalah .....	4
1.4 Tujuan Penelitian .....	5
1.5 Manfaat Penelitian .....	5
1.5.1. Bagi Masyarakat.....	5
1.5.2. Bagi Akademik.....	5
1.5.3. Bagi Penulis .....	6
1.6 Metodologi Penelitian .....	6
1.6.1. Objek Penelitian .....	6
1.6.2. Jenis Data .....	6
1.6.3. Metode Pengumpulan Data .....	7
1.6.4. Metode Pengembangan Sistem .....	7
1.7 Sistematika Penulisan .....	8
<b>BAB II    LANDASAN TEORI</b>	
2.1 Sistem Pendukung Keputusan.....	10
2.1.1. Pengertian Sistem.....	10

2.1.2.	Pengertian Keputusan.....	10
2.1.3.	Pengertian Pengambilan Keputusan.....	11
2.1.4.	Pengertian Sistem Pendukung Keputusan.....	12
2.1.5.	Karakteristik dan Kemampuan SPK .....	14
2.1.6.	Komponen-Komponen Sistem Pendukung Keputusan .....	15
2.2.	Bank .....	18
2.3.	Logika Fuzzy.....	20
2.3.1.	Perbedaan Himunan Fuzzy dengan Himpunan Pasti (Crisp).....	21
2.3.2.	Himpunan Fuzzy .....	24
2.3.3.	Fungsi Keanggotaan.....	26
2.3.4.	Basis Data Fuzzy Model Tahani .....	30
2.4.	Komponen Pemrograman Berorientasi Objek .....	34
2.4.1.	Menentukan Objek dan Kelas .....	34
2.4.2.	Menentukan Struktur Objek dan Hirarki Kelas ..	35
2.4.3.	Menentukan Subjek.....	36
2.4.4.	Menentukan Atribut .. ..	36
2.4.5.	Menentukan Metoda.....	37
2.4.6.	Menentukan Message.....	38
2.4.7.	Analisa Abbot .....	39
2.5.	Analisa dan Perancangan Berorientasi Objek .....	39
2.5.1.	OOP (Object Oriented Programing) .....	39
2.5.2.	Use Case.....	40
2.5.3.	Diagram Kelas.....	41
2.5.4.	Diagram Sekuensial .....	42
2.6.	Delphi.....	43
2.6.1.	Keunggulan Delphi .....	43
2.6.2.	Kelemahan Delphi.....	44
2.6.3.	Konversi Tipe Data .....	44



2.7.	MySQL.....	45
2.7.1.	Sejarah Singkat MySQL .....	46
<b>BAB III</b>	<b>ANALISA SISTEM</b>	
3.1	Analisa Permasalahan .....	47
3.2	Analisa Kriteria Sistem .....	49
3.2.1.	Kriteria Bunga.....	49
3.2.2.	Kriteria ATM .....	51
3.2.3.	Kriteria Administrasi.....	52
3.2.4.	Kriteria Pelayanan.....	54
<b>BAB IV</b>	<b>PERANCANGAN SISTEM</b>	
4.1	Analisa Abbot .....	56
4.2	Kata Benda .....	57
4.2.1.	Calon Kata Benda .....	57
4.2.2.	Eliminasi .....	57
4.3	Kata Kerja .....	57
4.3.1.	Calon Kata Kerja.....	57
4.3.2.	Eliminasi .....	58
4.4	Kelas Atribut .....	58
4.5	Use Case Diagram.....	59
4.6	Class Diagram .....	60
4.7	Activity Diagram.....	61
4.8	Sequence Diagram .....	62
4.9	Struktur Database .....	62
4.9.1.	Tabel Kriteria .....	62
4.9.2.	Tabel Bank .....	63
4.9.3.	Tabel Nilai.....	63
4.10	Model SPK.....	64
4.10.1.	Model Bunga.....	64
4.10.2.	Model ATM .....	66

4.10.3. Model Administrasi.....	67
4.10.4. Model Pelayanan.....	69
4.10.5. Domain.....	71
4.11 Perancangan Interface .....	72
4.11.1. Perancangan Form Utama.....	72
4.11.2. Perancangan Form Login.....	73
4.11.3. Perancangan Form Kriteria.....	73
4.11.4. Perancangan Form Bank.....	74
4.11.5. Perancangan Form Nilai.....	75
4.11.6. Perancangan Form SPK.....	76
4.11.7. Perancangan Hasil SPK.....	77

## BAB V IMPLEMENTASI SISTEM

5.1 Form Utama .....	79
5.2. Login.....	80
5.3. Menu Admin .....	81
5.3.1. Form Kriteria.....	81
5.3.2. Form Bank.....	83
5.3.3. Form Nilai .....	84
5.4. Form SPK.....	85
5.5. Kebutuhan Sistem .....	86
5.5.1. Kebutuhan Perangkat Keras.....	86
5.5.2. Kebutuhan Perangkat Lunak.....	87

## BAB VI KESIMPULAN DAN SARAN

6.1 Kesimpulan .....	88
6.2. Saran.....	89

## DAFTAR PUSTAKA

## LAMPIRAN

## DAFTAR TABEL

Tabel 2.1.	Data Mentah Karyawan.....	30
Tabel 2.2.	Tabel Temporer Karyawan .....	30
Tabel 2.3.	Karyawan Berdasarkan Umur .....	33
Tabel 2.4.	Simbol Use Case .....	41
Tabel 3.1.	Derajat Keanggotaan Kriteria Bunga .....	50
Tabel 3.2.	Derajat Keanggotaan Kriteria ATM.....	52
Tabel 3.3.	Derajat Keanggotaan Kriteria Administrasi.....	54
Tabel 3.4.	Derajat Keanggotaan Kriteria Pelayanan .....	55
Tabel 4.1.	Kelas Atribut .....	58
Tabel 4.2.	Tabel Kriteria .....	63
Tabel 4.3.	Tabel Bank .....	63
Tabel 4.4.	Tabel Nilai.....	64
Tabel 4.5.	Derajat Keanggotaan Kriteria Bunga .....	65
Tabel 4.6.	Derajat Keanggotaan Kriteria ATM.....	67
Tabel 4.7.	Derajat Keanggotaan Kriteria Administrasi.....	69
Tabel 4.8.	Derajat Keanggotaan Kriteria Pelayanan .....	70

## DAFTAR GAMBAR

Gambar 2.1.	Komponen SPK.....	18
Gambar 2.2.	Himpunan Muda, Parobaya, dan Tua.....	21
Gambar 2.3.	Himpunan Fuzzy Untuk Variabel Umur .....	23
Gambar 2.4.	Himpunan Fuzzy Temperatur (°C) .....	25
Gambar 2.5.	Representasi Linier Naik.....	27
Gambar 2.6.	Representasi Linier Turun.....	27
Gambar 2.7.	Representasi Kurva Segitiga .....	28
Gambar 2.8.	Representasi Kurva Trapesium .....	28
Gambar 2.9.	Representasi Kurva Bentuk Bahu .....	29
Gambar 2.10.	Fungsi Keanggotaan Untuk Variabel Usia.....	32
Gambar 2.11.	Kelas.....	41
Gambar 2.11.	Diagram Sekuensial .....	42
Gambar 3.1.	Fungsi Keanggotaan pada Kriteria Bunga .....	49
Gambar 3.2.	Fungsi Keanggotaan pada Kriteria ATM.....	51
Gambar 3.3.	Fungsi Keanggotaan pada Kriteria Administrasi .....	53
Gambar 3.4.	Fungsi Keanggotaan pada Kriteria Pelayanan .....	55
Gambar 4.1.	Use Case Diagram.....	59
Gambar 4.2.	Class Diagram .....	60
Gambar 4.3.	Activity Diagram.....	61
Gambar 4.4.	Sequence Diagram .....	62
Gambar 4.5.	Fungsi Keanggotaan pada Model Bunga .....	64
Gambar 4.6.	Fungsi Keanggotaan pada Model ATM.....	66
Gambar 4.7.	Fungsi Keanggotaan pada Model Administrasi .....	68
Gambar 4.8.	Fungsi Keanggotaan pada Model Pelayanan .....	69
Gambar 4.9.	Perancangan Form Utama .....	72
Gambar 4.10.	Perancangan Form Login .....	73
Gambar 4.11.	Perancangan Form Kriteria .....	74
Gambar 4.12.	Perancangan Form Bank .....	75
Gambar 4.13.	Perancangan Form Nilai.....	76

Gambar 4.14.	Perancangan Form SPK .....	77
Gambar 4.15.	Perancangan Hasil SPK.....	78
Gambar 5.1.	Form Utama .....	79
Gambar 5.2.	Form Login .....	80
Gambar 5.3.	Pesan Salah .....	81
Gambar 5.4.	Form Kriteria.....	82
Gambar 5.5.	Form Bank.....	83
Gambar 5.6.	Form Nilai .....	84
Gambar 5.7.	Form SPK.....	85
Gambar 5.8.	Hasil SPK.....	86

# BAB I

## PENDAHULUAN

### 1.1. Latar Belakang Masalah

Bisnis perbankan merupakan bisnis jasa yang berdasarkan pada azas kepercayaan perlu diiringi oleh perkembangan teknologi komputer untuk dapat memenuhi kebutuhan para nasabah sehingga masalah kualitas layanan menjadi faktor yang sangat menentukan dalam keberhasilan usaha dimana nasabah selalu memiliki pertimbangan atau faktor-faktor sebelum mengambil suatu keputusan. Untuk dapat membantu hal tersebut, perlu didukung ke dalam suatu sistem pendukung keputusan berbasis komputerisasi. Sistem pendukung keputusan (*decision support system*) selain dapat memberikan informasi juga dapat membantu menyediakan berbagai alternatif yang dapat ditempuh dalam proses pengambilan keputusan.

Pada tahun 1965, Lofti Zadeh mengembangkan suatu teori Logika *Fuzzy*. Logika ini merupakan konsep dasar dari sistem *Fuzzy* yang dapat digunakan untuk melakukan perhitungan terhadap suatu variabel masukan berdasarkan dari kriteria-kriteria yang ada. Dalam teori Logika *Fuzzy*, kriteria dinyatakan dalam derajat keanggotaan dan derajat dari kebenaran, sehingga sesuatu dapat dikatakan sebagian benar dan sebagian salah dalam waktu yang bersamaan. Kriteria dari pertimbangan faktor-faktor nasabah tersebut dapat

dituangkan ke dalam suatu konsep Logika *Fuzzy*. Sebagai contoh, untuk setiap bunga Bank yang ada, dapat dipetakan ke dalam fungsi keanggotaan sehingga didapatkan suatu nilai derajat keanggotaan dari setiap bunga Bank yang ada tersebut.

Tabungan adalah simpanan pihak ketiga pada bank yang penarikannya hanya dapat dilakukan menurut syarat-syarat tertentu yang disepakati, tetapi tidak dapat ditarik dengan cek, bilyet giro dan atau alat lainnya yang dipersamakan dengan itu. Walaupun bila dibandingkan dengan giro atau deposito, peranan tabungan dalam komposisi sumber dana perbankan relatif lebih kecil dan tingkat fluktuasi dana tabungan ini dianggap sangat kecil dan tidak selabil dana yang bersumber dari giro, namun tabungan merupakan jenis produk bank yang paling banyak diminati oleh nasabah sehingga merupakan komoditi yang paling penting.

Untuk pihak nasabah sendiri dari sekian banyak jumlah Bank yang ada, masing-masing menawarkan kriteria-kriteria dan keunggulan yang berbeda-beda, dan dari setiap kriteria dan keunggulan tersebut merupakan sebagai bahan pertimbangan nasabah untuk melakukan pemilihan Bank untuk membuka tabungan. Akan tetapi tanpa adanya data dan informasi yang kuantitatif sebagai penunjang proses pemilihan tempat untuk membuka tabungan di bank, maka pemilihan menjadi tidak obyektif dan seringkali terjadi kekeliruan. Hal ini justru akan mengakibatkan kerugian materiil dalam jangka waktu yang lama baik dari pihak Bank maupun nasabah itu sendiri.

Dari berdasar atas betapa pentingnya untuk menentukan peningkatan mutu dan pelayanan Bank secara efektif, sesuai keadaan perekonomian dan kebutuhan para nasabah. Maka sekiranya akan diperlukan sebuah analisis mengenai pemilihan Bank oleh Nasabah sebagai tempat menabung diharapkan para nasabah akan dapat mengambil suatu keputusan yang logis dan efektif mengenai pemilihan tempat menabung di bank yang sesuai dengan prioritas dan kebutuhan. Penelitian ini menitikberatkan pada analisis pemilihan Bank sebagai tempat menabung (yang merupakan salah satu sumber dana Bank dari pihak ketiga) dengan melakukan perbandingan antar preferensi dari faktor – faktor yang mempengaruhi pemilihan tempat menabung.

Metode analisis data untuk teknik pembobotan dan pengambilan keputusan yang dipergunakan dalam penelitian ini adalah menggunakan *Fuzzy database* yang dapat digunakan untuk menyampaikan informasi dari data yang bersifat *ambiguous*. Terdapat beberapa model *database* dalam logika *Fuzzy*, antara lain model Tahani. *Fuzzy database* model Tahani masih tetap menggunakan relasi standar, hanya saja model ini menggunakan teori himpunan *fuzzy* untuk mendapatkan informasi pada *query*-nya. Sistem yang akan dibangun telah ditentukan terlebih dahulu kriteria-kriteria yang umumnya nasabah akan menabung. Dari hasil kriteria pengguna, maka sistem melakukan perhitungan *fire strength* yang merupakan hasil operasi dalam himpunan *Fuzzy*.



## 1.2. Perumusan Masalah

Berdasarkan latar belakang masalah di atas maka rumusan masalah penelitian ini adalah "Bagaimana dapat membangun suatu aplikasi berbasis sistem pendukung keputusan untuk dapat memberikan rekomendasi bagi nasabah untuk memilih Bank sebagai tempat untuk menabung dengan metode *Fuzzy database* model Tahani.

## 1.3. Pembatasan Masalah

Untuk mengatasi masalah yang ada, maka penulis membatasi permasalahannya sebagai berikut :

1. Bank yang dibahas hanya daerah Semarang.
2. Variabel *Fuzzy* yang digunakan terdiri dari bunga, fasilitas pelayanan, fasilitas ATM dan biaya administrasi termasuk biaya ATM.
3. Fungsi keanggotaan yang digunakan adalah fungsi bahu yang terdiri dari bahu kiri dan bahu kanan dan fungsi keanggotaan segitiga.
4. Pembentukan *query* menggunakan operator AND atau OR untuk menghubungkan antar variabel.
5. Hasil dari *query* pengguna merupakan nilai rekomendasi yang diberikan, dan besarnya nilai rekomendasi yang diberikan merupakan nilai-nilai yang berada di dalam derajat keanggotaan antara 0 sampai 1.
6. Pembuatan program menggunakan bahasa pemrograman Delphi dan MySQL.

#### **1.4. Tujuan Penelitian**

Tujuan yang ingin dicapai dari penelitian ini adalah sebagai berikut:

1. Mempelajari dan mengetahui bagaimana metode *Fuzzy* database model Tahani dapat digunakan untuk memberikan rekomendasi untuk memilih Bank.
2. Membuat aplikasi berbasis sistem pendukung keputusan untuk rekomendasi untuk memilih Bank menggunakan metode *Fuzzy* database model Tahani.

#### **1.5. Manfaat Penelitian**

Adapun manfaat yang diharapkan dalam penulisan Skripsi ini adalah :

##### **1.5.1. Bagi Masyarakat**

Manfaat yang diperoleh dalam penelitian ini adalah diharapkan dapat membantu masyarakat dalam mengambil keputusan untuk memilih Bank sebagai tempat menabung yang sesuai dengan prioritas dan kebutuhan yang optimal dengan menggunakan Logika *Fuzzy*

##### **1.5.2. Bagi Akademik**

Sebagai bahan referensi yang dapat dipergunakan untuk perbandingan dan kerangka acuan untuk persoalan yang sejenis, sehingga dapat meningkatkan kualitas pendidikan. Serta dapat menjadi bahan acuan dan dorongan bagi akademik serta menjadi tolak ukur keberhasilan dalam memberikan bekal ilmu kepada mahasiswa sebelum terjun dalam persaingan tenaga kerja yang nyata.

### **1.5.3. Bagi Penulis**

Sebagai sarana untuk menerapkan dan mengembangkan ilmu yang didapat selama perkuliahan

## **1.6. Metodologi Penelitian**

Metodologi penelitian merupakan suatu cara atau prosedur yang digunakan untuk mengumpulkan data, dengan perantara teknik tertentu. Dalam penulisan skripsi ini, akan menggunakan metode penelitian yaitu :

### **1.6.1. Objek Penelitian**

Objek dari penelitian ini yaitu data-data tentang Bank di Semarang yang kemudian dimasukkan ke dalam metode Logika *Fuzzy* Tahani untuk dijadikan sebagai sistem pendukung keputusan untuk memilih bank terbaik.

### **1.6.2. Jenis Data**

Jenis data yang digunakan dalam penulisan skripsi ini antara lain data sekunder yaitu data yang diperoleh dari data penulis dalam bentuk yang sudah jadi yang bersifat informasi dan kutipan, baik dari internet maupun literatur, dokumen-dokumen yang berkaitan dengan sistem pendukung keputusan dan jurnal yang berhubungan dengan skripsi yang dibuat.

### **1.6.3. Metode Pengumpulan Data**

Metode yang digunakan dalam proses pengumpulan data sebagai bahan pembuatan sistem adalah studi pustaka yang dilakukan dengan mempelajari konsep logika *Fuzzy*, himpunan *Fuzzy*, operasi dalam himpunan *Fuzzy*, *Fuzzy* database, sistem pendukung keputusan dan OOD.

### **1.6.4. Metode Pengembangan Sistem**

Metode yang digunakan dalam pengembangan sistem adalah tahapan dalam membangun suatu SPK (Kusumadewi, 2002:153) diantaranya :

#### **1. Perencanaan**

Dalam tahap ini penulis mendefinisikan perencanaan dari sistem pendukung pengambilan keputusan untuk memilih Bank dengan Logika *Fuzzy* serta keputusan dengan cara menganalisa model pembuatan keputusan untuk memilih Bank yang optimal.

#### **2. Analisis**

Pada tahap ini melakukan analisis terhadap data yang telah diperoleh berdasarkan kriteria sumber yang menunjang dalam menyelesaikan permasalahan.

#### **3. Perancangan**

Pada tahap ini melakukan perancangan sistem dengan menggunakan *use case* diagram, *class* diagram, *sequence* diagram dan *state*

diagram. Pada tahap ini juga merancang pembuatan database yang dibuat berdasarkan *class* diagram juga membuat dialog dari sistem pendukung keputusan yang akan dibuat.

#### 4. Implementasi

Pada tahap ini melakukan perancangan terhadap sistem dan rancangan dalam tahap desain ke dalam bahasa pemrograman Delphi dan MySQL.

### 1.7. Sistematika Penulisan

Untuk memudahkan dalam penulisan skripsi ini, penulis akan menyajikan sistematika penulisan sebagai berikut :

#### Bab I            PENDAHULUAN

Pada bab ini akan dijelaskan tentang latar belakang masalah, perumusan masalah, pembatasan masalah, tujuan dan manfaat penelitian, metodologi penelitian, dan sistematika penulisan.

#### Bab II            LANDASAN TEORI

Berisikan hal-hal teoritis yang ada hubungannya dengan permasalahan yang sedang dibahas dan digunakan dalam penelitian.

**Bab III                    ANALISA SISTEM**

Pada bab ini akan dijelaskan tentang analisa sistem pendukung pengambilan keputusan untuk memilih Bank dengan menggunakan Logika *Fuzzy*.

**Bab IV                    PERANCANGAN SISTEM**

Pada bab ini akan dijelaskan tentang perancangan sistem pendukung pengambilan keputusan untuk memilih Bank dengan menggunakan Logika *Fuzzy*

**Bab V                    IMPLEMENTASI SISTEM**

Pada bab ini akan dijelaskan mengenai implementasi sistem program sistem pendukung pengambilan keputusan untuk memilih Bank dengan menggunakan Logika *Fuzzy*.

**Bab VI                    KESIMPULAN DAN SARAN**

Pada bab ini berisi tentang kesimpulan dan saran hasil penelitian.

## **BAB II**

### **LANDASAN TEORI**

#### **2.1. Sistem Pendukung Keputusan**

##### **2.1.1. Pengertian Sistem**

Sistem adalah suatu kesatuan yang terdiri atas komponen-komponen yang saling berhubungan erat menurut suatu rencana tertentu, dalam upaya untuk mencapai suatu tujuan atau beberapa tujuan tertentu.(Jogiyanto, 2002:123). Maka dapat disimpulkan bahwa sistem terdiri dari beberapa sub sistem. Sub sistem adalah bagian dari sistem dimana dengan bersatunya sub sistem tersebut, sistem akan dapat berjalan dengan sempurna.

##### **2.1.2. Pengertian Keputusan**

Adapun pengertian keputusan yaitu dapat dijabarkan sebagai berikut (Hasan,2002:9):

1. Keputusan adalah hasil pemecahan masalah yang dihadapi dengan tegas. Suatu keputusan merupakan jawaban yang pasti terhadap suatu pertanyaan.
2. Keputusan adalah suatu atau sebagai hukum situasi. Apabila semua fakta dari suatu itu dapat diperolehnya dan semua terlibat,

baik pengawas maupun pelaksana mau menaati ketentuannya maka tidak sama dengan menaati perintah.

3. Keputusan adalah pemilihan diantara suatu alternatif . Definisi ini mengandung tiga pengertian yaitu ada pilihan atas dasar logika, ada beberapa alternatif yang harus dan dipilih satu yang terbaik dan ada tujuan yang ingin dicapai dan keputusan itu makin mendekatkan pada tujuan tersebut.
4. Keputusan adalah suatu pengakhiran daripada proses pemikiran tentang suatu masalah untuk menjawab pertanyaan apa yang harus diperbuat guna mengatasi masalah tersebut dengan menjatuhkan pilihan pada satu alternatif.

Dari pengertian keputusan diatas dapat disimpulkan bahwa keputusan adalah suatu pemecahan masalah sebagai suatu hukum situasi yang dilakukan melalui pemilihan satu alternatif dari beberapa alternatif.

### **2.1.3. Pengertian Pengambilan Keputusan**

Adapun pengertian pengambilan keputusan dapat dijabarkan sebagai berikut (Hasan,2002:10) :

1. Pengambilan keputusan yaitu pemilihan alternatif perilaku tertentu dari dua atau lebih alternatif yang ada.



2. Pengambilan keputusan yaitu suatu pendekatan yang sistematis terhadap hakikat alternatif yang dihadapi dan mengambil tindakan yang menurut perhitungan merupakan tindakan yang paling tepat.
3. Pengambilan keputusan adalah proses yang digunakan untuk memilih suatu tindakan sebagai cara pemecahan masalah.

Dari pengertian pengambilan keputusan diatas dapat disimpulkan bahwa pengambilan keputusan adalah suatu proses pemilihan alternatif terbaik dari beberapa alternatif secara sistematis untuk ditindak lanjuti sebagai suatu cara pemecahan masalah.

#### **2.1.4. Pengertian Sistem Pendukung Keputusan**

Sistem pendukung keputusan didefinisikan sebagai sebuah sistem yang dimaksudkan untuk mendukung para pengambil keputusan manajerial dalam situasi situasi tertentu. Sistem pendukung keputusan dimaksudkan untuk menjadi alat bantu bagi para pengambil keputusan untuk memperluas kapabilitas mereka, namun tidak untuk menggantikan penilaian mereka (Turban, 2005:1).

Sistem pendukung keputusan merupakan sistem berbasis komputer interaktif yang membantu pengambil keputusan memanfaatkan data dan model untuk menyelesaikan masalah-masalah yang tak terstruktur. Ada beberapa hal yang menjadi alasan digunakannya Sistem Pendukung Keputusan, yaitu keadaan ekonomi yang tidak stabil, peningkatan persaingan yang terjadi dalam dunia

bisnis, kebutuhan akan informasi baru yang akurat, penyediaan informasi yang tepat waktu dan usaha untuk mengurangi biaya operasi. Selain itu, alasan lain dalam pengembangan Sistem pendukung Keputusan adalah perubahan perilaku komputasi *end-user*, *end-user* bukanlah *programmer*, sehingga mereka membutuhkan alat dan prosedur yang mudah untuk digunakan.

Proses pengambilan keputusan terdiri dari tiga fase proses, yaitu (Turban, 2005:2) :

1. Fase *intelligence* adalah fase dimana dilakukan pencarian kondisi-kondisi yang dapat menghasilkan keputusan.
2. Fase *design* adalah fase untuk menemukan, mengembangkan dan menganalisis materi-materi yang mungkin untuk dikerjakan.
3. Fase *choice* adalah fase dimana terjadi pemilihan dari materi-materi yang tersedia untuk menjadi keputusan akhir

Proses-proses yang terjadi pada kerangka kerja Sistem Pendukung Keputusan dibedakan atas :

1. Terstruktur

Mengacu pada permasalahan rutin dan berulang untuk solusi standar yang ada.

## 2. Tak Terstruktur

Keadaan yang kabur, permasalahan kompleks dimana tidak ada solusi yang tepat. Masalah yang tidak terstruktur terjadi akibat tidak adanya tiga fase proses yang terstruktur.

## 3. Semi Terstruktur

Terdapat beberapa keputusan terstruktur, tetapi tak semuanya dari fase-fase yang ada.

### **2.1.5. Karakteristik dan Kemampuan SPK**

Sistem Pendukung Keputusan memiliki beberapa karakteristik dan kemampuan, antara lain (Turban, 2005:5):

1. SPK menyediakan dukungan bagi pengambil keputusan utamanya pada situasi semi terstruktur dan tak terstruktur dengan memadukan pertimbangan manusia dan informasi terkomputerisasi.
2. Dukungan disediakan untuk berbagai level manajerial yang berbeda, mulai dari pimpinan puncak sampai manager lapangan.
3. Dukungan juga disediakan bagi individu dan juga bagi grup
4. SPK menyediakan dukungan ke berbagai keputusan yang berurutan atau saling berkaitan.
5. Mampu memberikan dukungan keputusan melalui beberapa fase : *intelligence, design, choice* dan *implementation*.
6. SPK selalu bisa beradaptasi sepanjang masa. Pengambil keputusan harus reaktif, mampu mengatasi perubahan kondisi secepat mungkin

dan beradaptasi untuk membuat SPK selalu bisa menangani perubahan.

7. SPK mudah digunakan. User harus merasa nyaman dengan sistem, karena itu sistem harus mempunyai dukungan grafis yang baik, antar muka yang *user-friendly*, sehingga menjadi sistem yang interaktif.
8. Pengambil keputusan memiliki kontrol menyeluruh terhadap proses pengambilan keputusan dalam menyelesaikan masalah. SPK secara khusus ditujukan untuk mendukung dan tak menggantikan pengambil keputusan. Pengambil keputusan dapat menindak lanjuti rekomendasi dari SPK untuk digunakan atau tidak digunakan.

#### **2.1.6. Komponen-Komponen Sistem Pendukung Keputusan**

Sistem pendukung keputusan terdiri dari 3 komponen utama atau subsistem yaitu (Turban, 2005:30):

##### **1. Subsistem data (*database*)**

Subsistem data merupakan komponen sistem pendukung keputusan penyedia data bagi sistem. Data yang dimaksud disimpan dalam suatu pangkalan data (*database*) yang diorganisasikan oleh suatu sistem yang disebut dengan sistem manajemen pangkalan data (*Data Base Management System* atau *DBMS*). Melalui *DBMS* inilah data dapat diambil dan dievakuasi dengan cepat. Pangkalan data dalam sistem pendukung keputusan berasal dari dua sumber

yaitu sumber internal (dari dalam perusahaan) dan sumber eksternal (dari luar perusahaan). Data eksternal ini sangat berguna bagi manajemen dalam mengambil keputusan.

## 2. Subsistem model (*model base*)

Keunikan sistem pendukung keputusan adalah kemampuannya dalam mengintegrasikan data dengan model-model keputusan. Model adalah suatu peniruan dari alam nyata. Kendala yang sering dihadapi dalam merancang suatu model adalah bahwa model yang disusun ternyata tidak mampu mencerminkan seluruh variabel alam nyata. Sehingga keputusan yang diambil yang didasarkan pada model tersebut menjadi tidak akurat dan tidak sesuai dengan kebutuhan. Oleh karena itu dalam menyimpan berbagai model pada sistem pangkalan model harus tetap dijaga fleksibilitasnya, artinya harus ada fasilitas yang mampu membantu pengguna untuk memodifikasi atau menyempurnakan model seiring dengan perkembangan pengetahuan. Hal lain yang perlu diperhatikan adalah pada setiap model yang disimpan hendaknya ditambahkan rincian keterangan dan penjelasan yang kompeherensif mengenai model yang dibuat, sehingga pengguna atau perancang :

- a. Mampu membuat model yang baru dengan mudah dan cepat.
- b. Mampu mengakses dan mengintegrasikan sub rutin model.

- c. Mampu menghubungkan model dengan model yang lain melalui pangkalan data.
- d. Mampu mengelola model base dengan fungsi manajemen yang analog dengan manajemen database.

### 3. Subsistem dialog (*user system interface*)

Keunikan lain dari Sistem Pendukung Keputusan adalah adanya fasilitas yang mampu mengintegrasikan sistem yang terpasang dengan pengguna secara interaktif. Fasilitas atau subsistem ini dikenal sebagai subsistem dialog, inilah sistem diartikulasikan dan diimplementasikan sehingga pengguna atau pemakai dapat berkomunikasi dengan sistem yang dirancang.

Fasilitas yang dimiliki oleh subsistem ini adalah :

#### a. Bahasa aksi (*action language*)

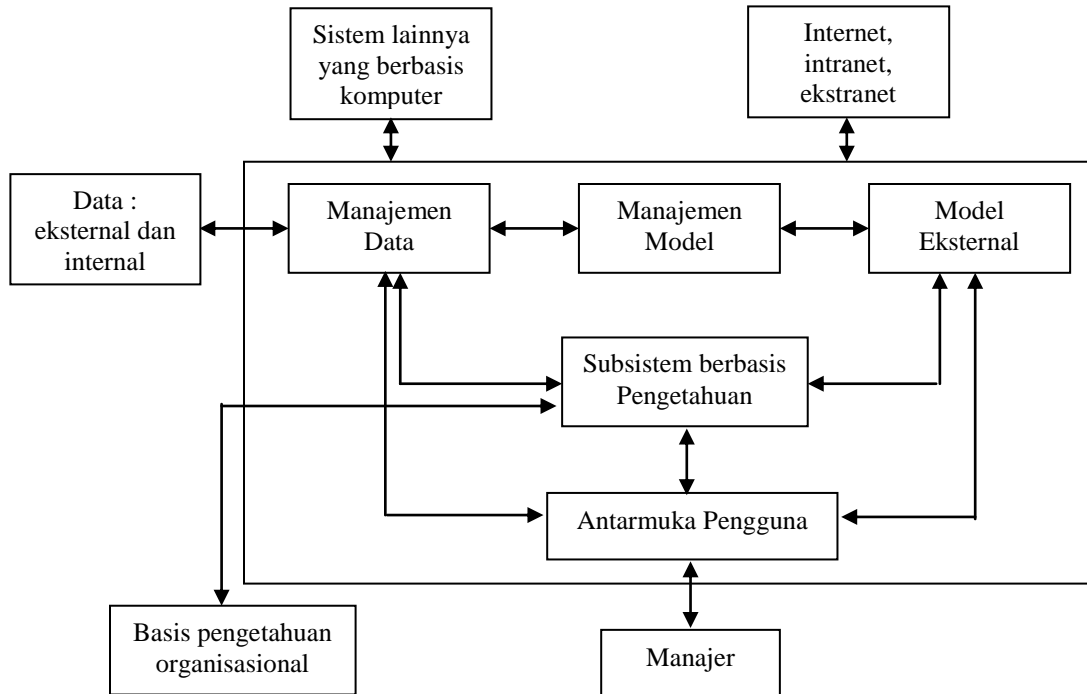
Merupakan suatu perangkat lunak yang dapat digunakan pengguna untuk berkomunikasi dengan sistem. Komunikasi ini dilakukan melalui berbagai media seperti *keybord*.

#### b. Bahasa tampilan

Merupakan suatu perangkat lunak yang berfungsi sebagai sarana untuk menampilkan sesuatu. Peralatan yang digunakan merealisasi tampilan diantaranya adalah printer, grafik monitor dan lain-lain.

c. Bahasa pengetahuan

Merupakan bagian yang mutlak diketahui oleh pengguna sehingga sistem ini dirancang dapat berfungsi secara efektif.



Gambar 2.1. Komponen SPK  
(Turban, 2005)

## 2.2. Bank

Sesuai dengan pengertian bank menurut UU-RI No. 10/1998 tentang Perbankan, bahwa Bank adalah badan usaha yang menghimpun dana dari masyarakat dalam bentuk simpanan dan menyalurkannya kepada masyarakat dalam bentuk kredit dan/atau bentuk-bentuk lainnya dalam rangka meningkatkan taraf hidup rakyat banyak (Kasmir, 2002), maka aktivitas

perbankan yang pertama adalah menghimpun dana dari masyarakat luas yang dikenal dengan istilah di dunia perbankan adalah kegiatan *funding*. Pengertian menghimpun dana maksudnya adalah mengumpulkan atau mencari dana dengan cara membeli dari masyarakat luas.

Kunci dari keberhasilan manajemen bank adalah bagaimana bank tersebut bisa merebut hati masyarakat sehingga peranannya sebagai financial intermediary dapat berjalan dengan baik, karena kegiatan manajemen dana bank meliputi perencanaan, pelaksanaan dan pengendalian terhadap penghimpunan pengalokasian dana dari masyarakat.

Pengertian tabungan menurut Undang-Undang Perbankan No.10 Tahun 1998, adalah simpanan pihak ketiga pada bank yang penarikannya hanya dapat dilakukan menurut syarat-syarat tertentu yang disepakati, tetapi tidak dapat ditarik dengan cek, bilyet giro dan atau alat lainnya yang dipersamakan dengan itu.

Menurut penelitian Robiah Rokhatul Ayun, Moses L. Singgih (2003) mengenai pengaruh kualitas pelayanan terhadap peningkatan keuntungan bank dengan pendekatan *Return On Quality*, dimana perkembangan dalam kualitas layanan dan tingginya tuntutan nasabah terhadap *service quality* suatu bank, menuntut banyak bank mengubah organisasi, produk dan pelayanannya. Menurut penelitian Winasta Ayu Duri (2002), mengenai analisa strategi pemasaran tabungan untuk meningkatkan pangsa pasar, yang dilatarbelakangi



oleh masalah kompetisi yang sengit menyebabkan semua bank berlomba-lomba menarik nasabah dengan mengeluarkan produk-produk baru yang menarik nasabah. Salah satu produk yang bersaing ketat di hampir semua bank umum adalah tabungan. Di mana tujuan penelitian tersebut untuk menganalisa faktor-faktor yang mempengaruhi keputusan konsumen (nasabah bank) dalam memilih tabungan, dengan kesimpulan, sebagai berikut :

1. Faktor I terdiri dari multifungsi ATM, kartu ATM, jaringan yang luas, saldo minimum dan lokasi yang strategis.
2. Faktor II terdiri dari suku bunga dan jaminan keamanan.

### **2.3. Logika Fuzzy**

*Logika fuzzy* pertama kali ditemukan oleh profesor Lotfi A. Zadeh, dari Universitas California, pada bulan Juni 1965. *Logika fuzzy* merupakan generalisasi dari logika klasik yang hanya memiliki dua nilai keanggotaan, yaitu 0 dan 1. Dalam *logika fuzzy*, nilai kebenaran suatu pernyataan berkisar dari sepenuhnya benar, sampai dengan sepenuhnya salah. Dengan teori himpunan *fuzzy*, suatu objek dapat menjadi anggota dari banyak himpunan dengan derajat keanggotaan yang berbeda dalam masing-masing himpunan. Konsep ini berbeda dengan himpunan klasik (*crisp*). Teori himpunan klasik tergantung pada logika dua nilai (*two valued logic*) untuk menentukan apakah

sebuah objek merupakan suatu anggota himpunan atau bukan. (Kusumadewi, 2010:1)

### 2.3.1. Perbedaan Himpunan *Fuzzy* dengan Himpunan Pasti (*Crisp*)

Pada himpunan tegas (*crisp*), nilai keanggotaan suatu item  $x$  dalam suatu himpunan  $A$ , yang sering ditulis dengan  $\mu_A[x]$ , memiliki dua kemungkinan (Kusumadewi, 2010:3), 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.

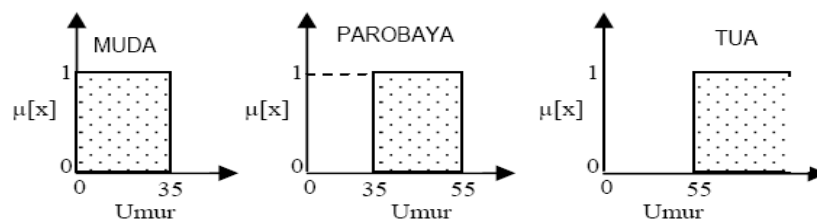
Contoh : misalkan variabel umur dibagi menjadi 3 kategori, yaitu:

MUDA                    umur < 35 tahun

PAROBAYA             $35 \leq \text{umur} \leq 55$  tahun

TUA                      umur  $\geq 55$  tahun

Nilai keanggotaan secara grafis, himpunan MUDA, PAROBAYA, dan TUA ini dapat dilihat pada gambar 2.2 berikut :



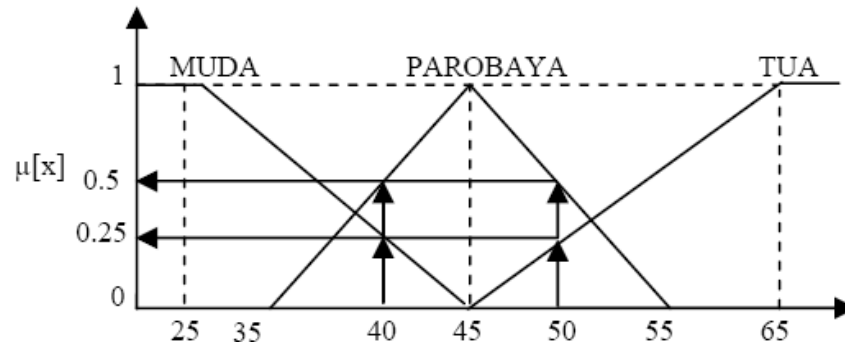
Gambar 2.2. Himpunan Muda, Parobaya, dan Tua (Kusumadewi, 2010)

Pada gambar di atas, dapat dijelaskan bahwa :

1. Apabila seseorang berusia 34 tahun, maka ia dikatakan MUDA ( $\mu_{MUDA}[34] = 1$ );
2. Apabila seseorang berusia 35 tahun, maka ia dikatakan TIDAK MUDA ( $\mu_{MUDA}[35] = 0$ );
3. Apabila seseorang berusia 35 tahun kurang 1 hari, maka ia dikatakan TIDAK PAROBAYA ( $\mu_{PAROBAYA}[35^{th} - 1 \text{ hari}] = 0$ );

Berdasarkan contoh di atas, bisa dikatakan pemakaian himpunan *crisp* untuk menyatakan umur sangatlah tidak adil, adanya perubahan sedikit saja pada suatu nilai mengakibatkan perbedaan kategori yang cukup signifikan.

Himpunan *fuzzy* digunakan untuk mengantisipasi hal tersebut. Seseorang dapat masuk dalam 2 himpunan yang berbeda, MUDA dan PAROBAYA, PAROBAYA dan TUA, dan sebagainya. Seberapa besar eksistensinya dalam himpunan tersebut dapat dilihat pada nilai keanggotaannya. Dalam himpunan *fuzzy*, himpunan umur muda, parobaya, dan tua dapat digambarkan sebagai berikut :



Gambar 2.3. Himpunan *Fuzzy* Untuk Variabel Umur  
(Kusumadewi, 2010)

Pada gambar di atas, dapat dilihat bahwa :

1. Seseorang yang berumur 40 tahun, termasuk dalam himpunan MUDA dengan  $\mu_{MUDA}[40] = 0,25$ ; namun dia juga termasuk dalam himpunan PAROBAYA dengan  $\mu_{PAROBAYA}[40] = 0,5$ .
2. Seseorang yang berumur 50 tahun, termasuk dalam himpunan TUA dengan  $\mu_{TUA}[50] = 0,25$ ; namun juga termasuk dalam himpunan PAROBAYA dengan  $\mu_{PAROBAYA}[50] = 0,5$ .

Jadi, jika pada himpunan *crisp*, nilai keanggotaan hanya ada dua kemungkinan, yakni 0 dan 1, pada himpunan *fuzzy* nilai keanggotaan terletak pada rentang 0 atau 1. Apabila  $x$  memiliki nilai keanggotaan *fuzzy*  $\mu_A[x]=0$  berarti  $x$  tidak menjadi anggota himpunan A, demikian pula apabila  $x$  memiliki nilai keanggotaan *fuzzy*  $\mu_A[x]=1$ , berarti  $x$  menjadi anggota penuh himpunan A.

### 2.3.2. Himpunan Fuzzy

Himpunan *fuzzy* memiliki 2 atribut, yaitu (Kusumadewi, 2010:15) :

1. *Linguistik*, yaitu penamaan suatu grup yang mewakili suatu keadaan atau kondisi tertentu dengan menggunakan bahasa alami, seperti : Muda, Tua, Parobaya.
2. *Numeris*, yaitu suatu nilai (angka) yang menunjukkan ukuran dari suatu variabel seperti : 40, 25, 50.

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

#### 1. Variabel *Fuzzy*

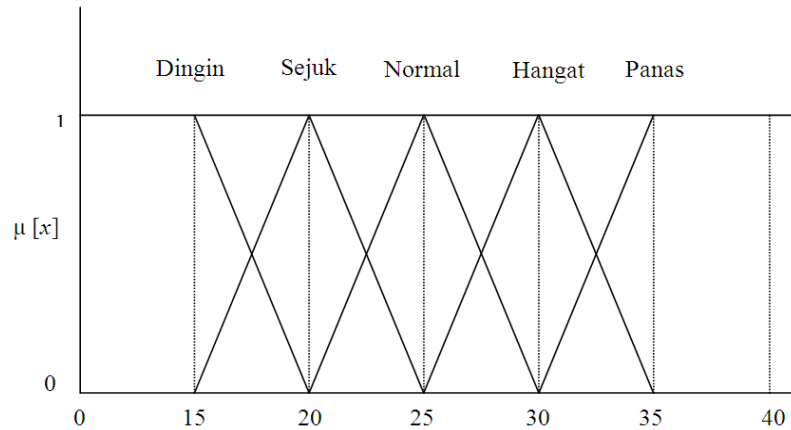
Variabel *fuzzy* merupakan variabel yang akan dibahas dalam suatu sistem *fuzzy*. Contoh : umur, temperatur, permintaan.

#### 2. Himpunan *Fuzzy*

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

Contoh :

- a. Variabel umur, terbagi menjadi tiga buah himpunan *fuzzy*, yaitu: muda, parobaya, dan tua.
- b. Variabel temperatur, terbagi menjadi lima buah himpunan *fuzzy*, yaitu: dingin, sejuk, normal, hangat, dan panas.



Gambar 2.4. Himpunan *Fuzzy* Temperatur ( $^{\circ}\text{C}$ )  
(Kusumadewi, 2010)

### 3. Semesta Pembicaraan

Semesta pembicaraan adalah keseluruhan nilai yang diperbolehkan untuk dioperasikan dalam suatu variabel *fuzzy*. Semesta pembicaraan merupakan himpunan bilangan *real* yang senantiasa naik (bertambah) secara monoton dari kiri ke kanan. Nilai semesta pembicaraan dapat berupa bilangan positif maupun negatif. Seharusnya nilai semesta pembicaraan ini tidak dibatasi batas atasnya.

Contoh :

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

### 4. Domain

Domain himpunan *fuzzy* adalah keseluruhan nilai yang diijinkan dalam semesta pembicaraan dan boleh dioperasikan dalam

suatu himpunan *fuzzy*. Seperti halnya semesta pembicaraan, domain merupakan himpunan bilangan *real* yang senantiasa naik (bertambah) secara monoton dari kiri ke kanan. Nilai domain dapat berupa bilangan positif maupun negatif.

Contoh domain himpunan *fuzzy* :

- a. MUDA =  $[0, 45]$
- b. PAROBAYA =  $[35, 55]$
- c. TUA =  $[45, +\infty]$

### 2.3.3. Fungsi Keanggotaan

Fungsi keanggotaan (*membership function*) adalah suatu kurva yang menunjukkan pemetaan titik-titik input data ke dalam nilai keanggotaannya (sering juga disebut dengan derajat keanggotaan) yang memiliki interval antara 0 sampai 1. Salah satu cara yang dapat digunakan untuk mendapatkan nilai keanggotaan adalah dengan melalui pendekatan fungsi. Ada beberapa fungsi yang dapat digunakan (Kusumadewi, 2010:34):

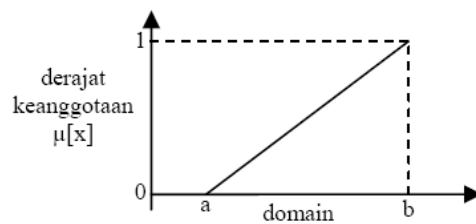
#### 1. Representasi Linier

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 linear, yaitu:

a. Representasi Linear Naik

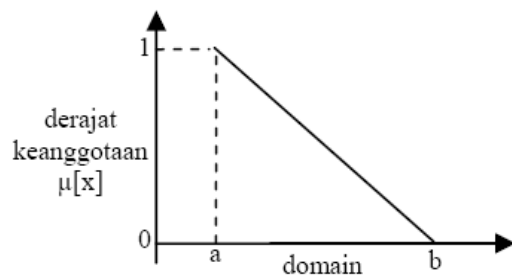
Kenaikan himpunan dimulai pada nilai domain yang memiliki derajat keanggotaan nol (0) bergerak ke kanan menuju ke nilai domain yang memiliki derajat keanggotaan lebih tinggi



Gambar 2.5. Representasi Linier Naik (Kusumadewi, 2010)

b. Representasi Linier Turun

Representasi linear turun merupakan kebalikan dari linear naik. 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.

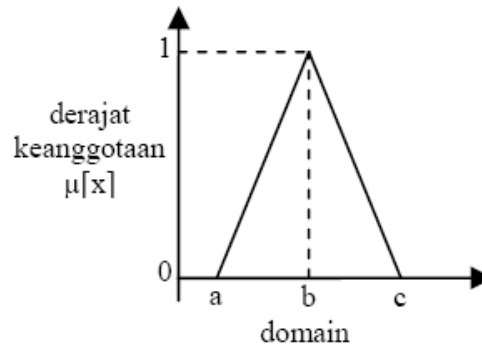


Gambar 2.6. Representasi Linier Turun (Kusumadewi, 2010)



## 2. Representasi Kurva Segitiga

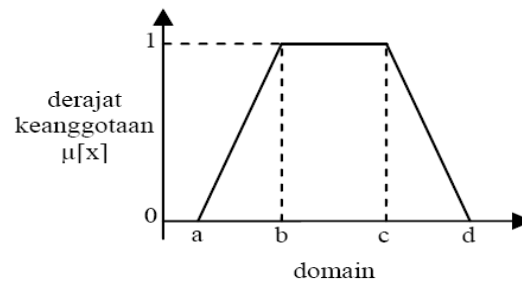
Kurva segitiga pada dasarnya merupakan gabungan antara 2 garis (linier).



Gambar 2.7. Representasi Kurva Segitiga (Kusumadewi, 2010)

## 3. Representasi Kurva Trapesium

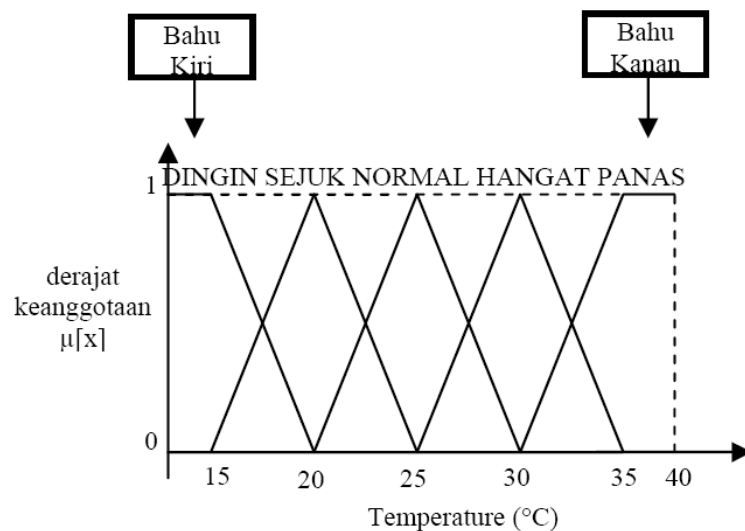
Kurva trapesium pada dasarnya seperti bentuk segitiga, hanya saja ada beberapa titik yang memiliki nilai keanggotaan 1.



Gambar 2.8. Representasi Kurva Trapesium (Kusumadewi, 2010)

#### 4. Representasi Kurva Bentuk Bahu

Daerah yang terletak di tengah-tengah suatu variabel yang direpresentasikan dalam bentuk segitiga, pada sisi kanan dan kirinya akan naik dan turun (misalkan : DINGIN bergerak ke SEJUK bergerak ke HANGAT dan bergerak ke PANAS). Tetapi terkadang salah satu sisi dari variabel tersebut tidak mengalami perubahan. Sebagai contoh, apabila telah mencapai kondisi PANAS, kenaikan temperatur akan tetap berada pada kondisi PANAS. Himpunan *fuzzy* 'bahu', bukan segitiga, digunakan untuk mengakhiri variabel suatu daerah *fuzzy*. Bahu kiri bergerak dari benar ke salah, sebaliknya bahu kanan bergerak dari salah ke benar. Gambar berikut menunjukkan variable TEMPERATUR dengan daerah bahunya.



Gambar 2.9. Representasi Kurva Bentuk Bahu  
(Kusumadewi, 2010)

#### 2.3.4. Basis Data *Fuzzy Model Tahani*

##### 1. Konsep Dasar

Sebagian besar basis data standar diklasifikasikan berdasarkan bagaimana data tersebut dipandang oleh user. Misalkan data karyawan yang tersimpan pada tabel dt\_karyawan dengan field NIP, Nama, Tgl Lahir dan Tahun Masuk seperti pada Tabel berikut ini (Kusumadewi, 2010:54) :

Tabel 2.1. Data Mentah Karyawan  
(Kusumadewi, 2010)

<b>NIP</b>	<b>Nama</b>	<b>Tgl Lahir</b>	<b>Th Masuk</b>
01	Lita	02-02-1981	1996
02	Irwan	23-09-1963	1985
03	Sari	23-09-1975	1988
04	Andri	02-02-1974	1998
05	Budi	26-04-1969	1990

Kemudian dari tabel tersebut, akan diolah menjadi suatu tabel temporer untuk menghitung umur karyawan dan masa kerjanya. Tabel tersebut diberi nama dengan tabel Karyawan.

Tabel 2.2. Tabel Temporer Karyawan  
(Kusumadewi, 2010)

<b>NIP</b>	<b>Nama</b>	<b>Umur (th)</b>	<b>Masa Kerja</b>
01	Lita	30	6
02	Irwan	48	17
03	Sari	36	14
04	Andri	37	4
05	Budi	42	12

\*) Dimisalkan saat ini adalah tahun 2011

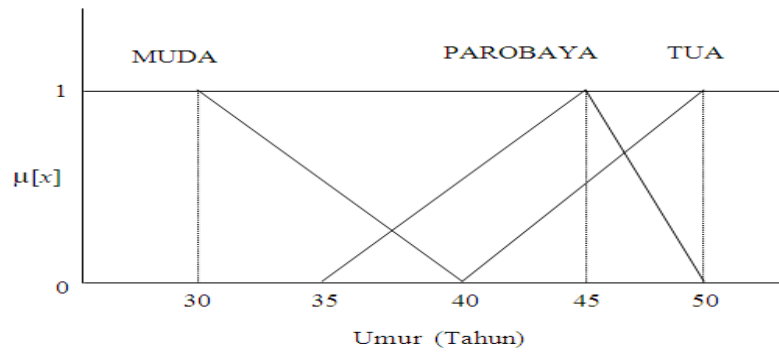
Dengan menggunakan basis data standar, dapat dicari data-data karyawan dengan spesifikasi tertentu dengan menggunakan query. Misalkan kita ingin mendapatkan informasi tentang nama-nama karyawan yang usianya kurang dari 35 tahun, maka kita bisa ciptakan suatu *query* sebagai berikut :

```
SELECT NAMA  
FROM KARYAWAN  
WHERE (Umur < 35)
```

Sehingga akan muncul nama Lita.

Namun pada kenyataannya, informasi yang dibutuhkan adalah dari data-data yang bersifat ambiguous. Apabila hal ini terjadi, maka dapat diatasi dengan basis data *fuzzy*. Salah satu diantaranya adalah model Tahani. Basisdata *fuzzy* model Tahani masih tetap menggunakan relasi standar, hanya saja model ini menggunakan teori himpunan *fuzzy* untuk mendapatkan informasi pada query-nya.

Misalkan mengkategorikan usia karyawan di atas ke dalam himpunan : MUDA, PAROBAYA, dan TUA. Seperti pada bab berikut :



Gambar 2.10. Fungsi Keanggotaan Untuk Variabel Usia (Kusumadewi, 2010)

a. Fungsi keanggotaan Muda (MD):

$$\mu_{MD}[x] \begin{cases} 1; & x \leq 30 \\ \frac{40-x}{10}; & 30 \leq x \leq 40 \\ 0; & x \geq 40 \end{cases}$$

b. Fungsi keanggotaan Parobaya (PB):

$$\mu_{PB}[x] \begin{cases} 0; & x \leq 35 \text{ atau } x \geq 50 \\ \frac{x-35}{10}; & 35 \leq x \leq 45 \\ \frac{50-x}{5}; & 45 \leq x \leq 50 \end{cases}$$

c. Fungsi keanggotaan Tua (TA):

$$\mu_{TA}[x] \begin{cases} 0; & x \leq 40 \\ \frac{x - 40}{10}; & 40 \leq x \leq 50 \\ 1; & x \geq 50 \end{cases}$$

Pada tabel berikut ini menunjukkan tabel karyawan berdasarkan pada umur dengan derajat keanggotaan pada setiap himpunan.

Tabel 2.3. Karyawan Berdasarkan Umur  
(Kusumadewi, 2010)

NIP	Nama	Umur	Derajat Keanggotaan ([x])		
			Muda	Parobaya	Tua
01	Lita	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

Besarnya nilai rekomendasi berkisar antara [0 1], dengan rekomendasi tertinggi adalah 1 dan berangsur tidak direkomendasikan apabila nilainya semakin mendekati 0.

Dari tabel karyawan di atas dapat diciptakan suatu query sebagai berikut :

*SELECT NAMA FROM KARYAWAN WHERE Umur=MUDA)*

sehingga muncul nama Lita, Sari, Andi.

## **2.4. Komponen Pemrograman Berorientasi Objek**

### **2.4.1. Menentukan Objek dan Kelas**

Tujuan utama untuk mengidentifikasi kelas dan objek adalah menyesuaikan penggambaran secara teknis dari sistem sedekat mungkin dengan gambaran konseptual dengan dunia nyata. Tahap-tahap dalam menentukan kelas dan objek adalah :

- a. Memahami pokok permasalahan
- b. Menentukan kelas dan objek sebagai suatu abstraksi dunia nyata.

Hal ini dapat membantu untuk mendapatkan pemahaman dari masalah yang signifikan dengan tujuan dapat dijalankan dalam sistem.

Tujuan lain untuk mengidentifikasi kelas dan objek adalah membuat kerangka kerja yang stabil untuk analisis dan membuat spesifikasi.

Tujuan terakhir adalah untuk menghindari penyimpangan pada transisi dari tahap analisis sistem ke perancangan. Inti dari perancangan adalah mengambil kebutuhan dan menambahkan rinciannya pada implementasi sesuai dengan pokok masalah yang ada.

## 2.4.2. Menentukan Struktur Objek dan Hirarki Kelas

### 1. Definisi Struktur

*Gen-Spec structure* diperoleh dengan melihat dari bagian yang membedakan dengan kelas dan mempunyai kekhususan yang dimiliki oleh kelas.

*Whole-Part structure* adalah bagian dari metode dasar bagaimana cara orang berpikir, yaitu dengan melihat suatu objek adalah bagian dari suatu kelas.

### 2. Cara Menentukan Struktur

Pada setiap kelas dan objek diperhatikan setiap kelas untuk *Gen-Spec structure* dan setiap objek untuk *Whole-Part structure*.

Cara menentukan *Gen-Spec structure* adalah :

- a. Apakah terdapat dalam masalah domain ?
- b. Apakah hubungan dengan sistem ?
- c. Apakah ada yang dapat diwariskan ?
- d. Apakah spesialisasi dapat sesuai dengan kriteria dari kelas dan objek ?

Cara menentukan *Whole-part structure* adalah :

- a. *Assembly* – bagian
- b. *Contour* – isi
- c. *Collection* – anggota



### **2.4.3. Menentukan Subjek**

#### **1. Definisi Subjek**

Subjek merefleksikan masalah domain dan keterkaitannya dengan sistem. dasar utama untuk mengidentifikasi subjek adalah kompleksitas masalah domain seperti yang ditentukan dalam *Gen-Spec structure* dan *Whole-Part structure*. Subjek adalah bagian dari masalah domain dan sistem secara keseluruhan.

Subjek digunakan sebagai pedoman bagi analis, manajer, klien untuk menyelesaikan model yang besar dan kompleks. Subjek sangat berguna untuk mengorganisasi paket pekerjaan pada proyek yang besar, berdasarkan investigasi dengan analisis berorientasi objek.

#### **2. Cara Menentukan Subjek**

- a. Memilih kelas yang tertinggi dalam setiap struktur menjadi suatu subjek.
- b. Memilih setiap kelas dan objek menjadi subjek.

### **2.4.4. Menentukan Atribut**

#### **1. Definisi Atribut**

Atribut adalah properti, kualitas dan karakteristik yang dapat ditentukan pada orang atau barang.

## 2. Cara Menentukan Atribut

Cara menentukan atribut dengan menentukan apa yang harus diketahui dari objek yaitu :

- a. Bagaimana menggambarkan secara umum ?
- b. Bagaimana menggambarkan sesuai dengan masalah domain ?
- c. Bagaimana menggambarkan konteks sistem ?
- d. Apakah yang perlu diketahui ?
- e. Informasi apakah yang perlu diingat ?

### 2.4.5. Menentukan Metoda

#### 1. Definisi Metoda

Metoda adalah perilaku spesifik dimana suatu objek mempunyai tanggung jawab untuk menampilkannya.

#### 2. Cara Menentukan Metoda

Cara menentukan metoda dapat dilakukan dengan beberapa kegiatan yaitu :

- a. Identifikasi *Object State*
- b. Identifikasi metoda yang diperlukan
- c. Identifikasi hubungan *message*
- d. Spesifikasi metoda

## 2.4.6. Menentukan Message

### 1. Definisi Message

Message adalah komunikasi antara objek. Hubungan message adalah model hubungan yang menggambarkan suatu objek mengirimkan message kepada objek lain untuk melakukan pemrosesan.

### 2. Cara Menentukan Message

Cara menentukan message yang diperlukan, maka perhatian pada objek meliputi :

- a. Apakah objek lain memerlukan metode dari objek tersebut ?
- b. Apakah objek lain memerlukan salah satu metoda ?
- c. Telusuri setiap hubungan message dan ulangi kedua pertanyaan itu

Message dikirimkan melalui hubungan message untuk meminta suatu metoda dijalankan. Pada hubungan message adalah sebagai berikut :

- a. Pengirim mengirimkan message
- b. Setelah penerima menerima message, maka satu atau lebih metoda dijalankan.
- c. Penerima mengirimkan kembali jawaban kepada pengirim.

#### **2.4.7. Analisa Abbot**

Analisa Abbot merupakan langkah-langkah dalam pembuatan class diagram. Analisa ini dilakukan dengan langkah-langkah yaitu:

- a. Buat daftar kata benda yang muncul dalam paparan sistem.
- b. Tinggalkan kata benda yang perlu disimpan oleh perangkat lunak.
- c. Buang kata benda yang terlalu umum, tidak relevan, mendua arti, berulang atau sinonim dengan kata lain.
- d. Tentukan kata benda yang memiliki kata benda lain.
- e. Kata benda yang memiliki benda lain menjadi calon kelas dan kata benda yang dimiliki benda lain menjadi atribut.
- f. Periksa kembali kelas dan atribut yang ditemukan, tambahkan atribut lain apabila diperlukan.
- g. Buat daftar kata kerja yang muncul dalam paparan sistem.
- h. Temukan relasi memiliki antara kelas dan kata kerja yang anda di daftar. Kata kerja menjadi operasi dari kelas.
- i. Periksa kembali kelas, atribut dan operasi yang ditemukan

### **2.5. Analisa dan Perancangan Berorientasi Objek**

#### **2.5.1. OOP (*Object Oriented Programing*)**



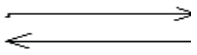
OOP atau pemrograman berorientasi objek adalah suatu cara baru dalam berpikir serta berlogika dalam menghadapi masalah-masalah yang akan dicoba atasi dengan bantuan komputer. OOP tidak

seperti pendahulunya (pemrograman terstruktur), mencoba melihat permasalahan lewat pengamatan dunia nyata dimana setiap objek adalah entitas tunggal yang memiliki kombinasi struktur data dan fungsi tertentu. Ini kontras dengan pemrograman terstruktur dimana struktur data dan fungsi didefinisikan secara terpisah dan tidak berhubungan secara erat. Filosofi OOP menciptakan sinergi yang luar biasa sepanjang siklus pengembangan perangkat lunak (perencanaan, analisis, perancangan, serta implementasi) sehingga dapat diterapkan pada perancangan sistem secara umum menyangkut perangkat lunak, perangkat keras, serta sistem informasi secara keseluruhan. (Gunadi, 2002:17).

### **2.5.2. Use Case**

Use case adalah bagian dari tingkat tinggi fungsionalitas yang disebut oleh sistem. Dengan kata lain use case menggambarkan bagaimana seseorang menggunakan sistem. (Gunadi, 2002:60).  
Dibawah ini merupakan simbol-simbol yang digunakan dalam pembuatan use case.

Tabel 2.4. Simbol Use Case  
(Gunadi, 2002)

No.	Komponen	Arti	Keterangan
1.		Actor	Manusia, user, pengguna sistem yang berhubungan langsung dengan sistem
2.		Proses	Perilaku yang ditunjukkan atau dilakukan oleh actor.
3.		Relasi	Penghubung antara actor dengan proses atau proses dengan proses.

Dalam relasi ini terdapat dua komponen yaitu :

1. *Include*

Menunjukkan bagian dari elemen (yang ada di garis tanpa panah) memicu eksekusi bagian dari elemen lain (yang ada di garis dengan panah).

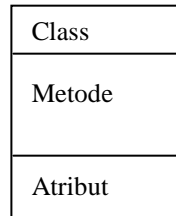
2. *Extend*

Menunjukkan suatu bagian dari elemen di garis tanpa panah bisa disisipkan ke dalam elemen yang ada di garis dengan pan.

**2.5.3. Diagram Kelas**

Diagram kelas adalah alat perancangan terbaik untuk tim pengembangan perangkat lunak. Diagram tersebut membantu pengembang mendapatkan struktur sistem sebelum menuliskan kode

program, membantu untuk memastikan bahwa sistem adalah rancangan terbaik. (Gunadi, 2002:102).



Gambar 2.11. Kelas  
(Gunadi o, 2002)

#### 1. *Class*

*Class* adalah definisi umum (pola, template atau cetak biru) untuk himpunan objek sejenis, kelas menetapkan spesifikasi perilaku dan objek - objek tersebut.

#### 2. *Attributes* (Atribut)

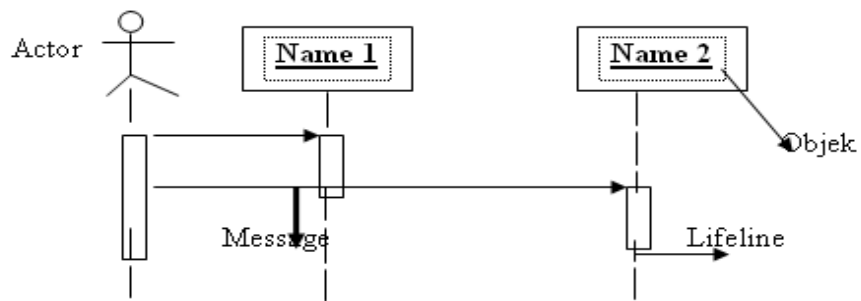
*Attributes* merupakan karakteristik suatu objek dan biasanya ditandai dengan kata sifat dan “*frase*” milik.

#### 3. *Method* (Metode)

*Method* merupakan subprogram yang tergantung yang bersama-sama dengan atribut.

### 2.5.4. Diagram Sekuensial

Diagram sequensial atau sequence diagram digunakan untuk menunjukkan aliran fungsionalitas dalam use case. (Gunadi, 2006:10).



Gambar 2.12. Diagram Sekuensial  
(Gunadi, 2002)

## 2.6. Delphi

Borland delphi merupakan perangkat lunak untuk menyusun program aplikasi yang berbasis bahasa pemrograman pascal yang bekerja dalam lingkungan sistem operasi windows dan sepenuhnya menggunakan metode OOP (*Object Oriented Programming*). Delphi menyediakan fasilitas IDE sebagai area kerja program aplikasi, dimana seluruh pengaksesan fungsi-fungsi yang diperlukan dan disediakan dalam satu tampilan. Lingkungan pemrograman visual seperti delphi memungkinkan aplikasi dengan mudah dan cepat. (Pranata, 2002:1)

Pada suatu program aplikasi, dirancang suatu tampilan yang terdiri dari form beserta komponen-komponen yang dibutuhkan, kode program untuk menentukan rutin yang harus dikerjakan dan file project untuk mengkoordinasi semua file yang ada. Hasil rencana tampilan disimpan pada file dengan ekstensi



\*.pas dan semua file yang dibuat diorganisasi oleh sebuah file project dengan ekstensi \*.dpr.

### **2.6.1. Keunggulan Delphi**

Keunggulan dari pemrograman delphi yaitu (Pranata, 2002:3) :

1. Bahasa pemrograman yang terstruktur dan *syntax* yang jelas, sehingga mudah dibaca, dipelajari dan dipahami.
2. Kecepatan kompilasi-nya.
3. Dukungan komponen tambahan yang sangat banyak di Internet, bahkan untuk kategori freeware dengan kualitas yang sama atau lebih dari komponen komersial.
4. Dukungan komunitas di Internet yang sangat banyak.

### **2.6.2. Kelemahan Delphi**

Kelemahan dari pemrograman delphi yaitu (Pranata, 2002:4) :

1. Hasil kompilasi (file exe) memiliki ukuran yang cukup besar jika dibandingkan dengan Kompiler lain.
2. Waktu proses aplikasi sebagian besar sedikit lebih lambat jika dibandingkan dengan Visual C.
3. Tidak banyak perkembangan untuk platform win32 sejak Delphi 7.
4. Pindahanya beberapa founder delphi ke Microsoft sejak Delphi 5.

### 2.6.3. Konversi Tipe Data

Dalam pengolahan data memerlukan suatu konversi tipe data. Contoh konversi tipe data yang dapat dilakukan oleh delphi antara lain (Pranata, 2002:7) :

1. Strtoint

Mengubah tipe data string ke integer.

2. Inttostr

Mengubah tipe data integer ke string.

3. Timetostr

Mengubah tipe data time ke dalam bentuk string.

4. Strtfloat

Mengubah tipe data string ke dalam bentuk real.

5. Floattostr

Mengubah tipe data real ke dalam bentuk string.

## 2.7. MySQL

MySQL adalah relational database management system (RDBMS) yang didistribusikan secara gratis dibawah licensi GPL (*General Public License*). MySQL sebenarnya merupakan turunan salah satu konsep utama dalam database sejak lama yaitu SQL (*Structured Query Language*). SQL adalah sebuah konsep pengoperasian database terutama untuk pemilihan/seleksi dan

pemasukan data yang memungkinkan pengoperasian data dikerjakan dengan mudah dan secara otomatis. (Prasetyo, 2003:1)

Keandalan suatu sistem database dapat diketahui dari cara kerja optimizer nya dalam melakukan proses perintah-perintah SQL, yang dibuat oleh user maupun program-program aplikasinya. Sebagai database server, MySQL dapat dikatakan lebih unggul dibandingkan database server lainnya dalam query data. Hal ini terbukti untuk query yang dilakukan oleh single user, kecepatan query My SQL dapat sepuluh kali lebih cepat dari PostgreSQL dan lima kali lebih cepat dibandingkan Interbase. MySQL dikembangkan sekitar tahun 1994 oleh sebuah perusahaan pengembang software dan konsultan database bernama MySQL AB yang bertempat di Swedia. Pada saat itu perusahaan tersebut masih bernama TcX DataKonsult AB dan tujuan awal dikembangkannya MySQL untuk mengembangkan aplikasi berbasis web pada client. Awalnya Michael Widenius pengembang satu-satunya di TcX memiliki sebuah aplikasi UNIREG dan rutin ISAM buatannya sendiri dan sedang mencari antarmuka SQL yang cocok untuk diimplementasikan ke dalamnya. Mula-mula Monty memakai miniSQL(mSQL) pada eksperimennya, namun mSQL dirasa kurang sesuai karena terlalu lambat dalam pemrosesan query.

Akhirnya Monty menghubungi David Hughes, pembuat mSQL yang sedang merilis versi kedua dari mSQL. Kemudian Monty mencoba membuat sendiri mesin SQL yang memiliki antarmuka mirip dengan SQL tetapi dengan kemampuan yang lebih dan lahirlah MySQL.

## **BAB III**

### **ANALISA SISTEM**

#### **3.1. Analisa Permasalahan**

Bisnis perbankan merupakan bisnis jasa yang berdasarkan pada asas kepercayaan sehingga masalah kualitas layanan menjadi faktor yang sangat menentukan dalam keberhasilan usaha. Kualitas layanan merupakan suatu bentuk penilaian konsumen terhadap tingkat layanan yang diterima (*perceived service*) dengan tingkat layanan yang diharapkan.

Tabungan adalah simpanan pihak ketiga pada bank yang penarikannya hanya dapat dilakukan menurut syarat-syarat tertentu yang disepakati, tetapi tidak dapat ditarik dengan cek, bilyet giro dan atau alat lainnya yang dipersamakan dengan itu. Walaupun bila dibandingkan dengan giro atau deposito, peranan tabungan dalam komposisi sumber dana perbankan relatif lebih kecil dan tingkat fluktuasi dana tabungan ini dianggap sangat kecil dan tidak selabil dana yang bersumber dari giro, namun tabungan merupakan jenis produk bank yang paling banyak diminati oleh nasabah sehingga merupakan komoditi yang paling penting.

ATM Bersama adalah jaringan ATM pertama yang beroperasi di Indonesia, yang awalnya menghubungkan 21 bank di Indonesia dan saat ini jaringan ATM Bersama memiliki 75 anggota. ATM Bersama menyediakan

banyak fasilitas termasuk cek saldo, penarikan tunai dan pemindahan dana secara online seketika ke rekening lain dari sesama anggota jaringan ATM Bersama.

Untuk pihak nasabah sendiri dari sekian banyak jumlah Bank yang ada, masing-masing menawarkan kriteria-kriteria dan keunggulan yang berbeda-beda, dan dari setiap kriteria dan keunggulan tersebut merupakan sebagai bahan pertimbangan nasabah untuk melakukan pemilihan Bank untuk membuka tabungan. Akan tetapi tanpa adanya data dan informasi yang kuantitatif sebagai penunjang proses pemilihan tempat untuk membuka tabungan di bank, maka pemilihan menjadi tidak obyektif dan seringkali terjadi kekeliruan. Hal ini justru akan mengakibatkan kerugian materiil dalam jangka waktu yang lama baik dari pihak Bank maupun nasabah itu sendiri.

Dari berdasar atas betapa pentingnya untuk menentukan peningkatan mutu dan pelayanan Bank secara efektif, sesuai keadaan perekonomian dan kebutuhan para nasabah. Maka sekiranya akan diperlukan sebuah analisis mengenai pemilihan Bank oleh Nasabah sebagai tempat menabung diharapkan para nasabah akan dapat mengambil suatu keputusan yang logis dan efektif mengenai pemilihan tempat menabung di bank yang sesuai dengan prioritas dan kebutuhan. Penelitian ini menitikberatkan pada analisis pemilihan Bank sebagai tempat menabung (yang merupakan salah satu sumber dana Bank dari pihak ke tiga) dengan melakukan perbandingan antar preferensi dari faktor – faktor yang mempengaruhi pemilihan tempat menabung. Data-data penelitian berupa data

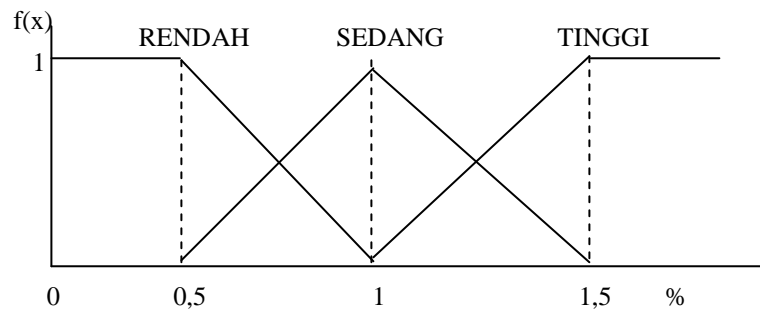
kualitatif yang diperoleh dari Bank-Bank yang dinilai memiliki keunggulan seperti Bank Mandiri, Bank BNI, Bank BCA, Bank BPD Jateng, Bank Mega, Bank Danamon dan Bank BRI.

### 3.2. Analisa Kriteria Sistem

Tahap pembuatan aplikasi ini, terlebih dahulu adalah menentukan dan merencanakan kriteria-kriteria dalam pemilihan Bank sebagai tempat untuk menabung yaitu

#### 3.2.1. Kriteria Bunga

Bunga merupakan suku bunga Bank yang diberikan kepada nasabah setiap bulan. Kriteria bunga dibagi menjadi 3 himpunan *fuzzy*, yaitu : RENDAH, SEDANG dan TINGGI. Himpunan RENDAH dan TINGGI menggunakan pendekatan fungsi keanggotaan yang berbentuk bahu, sedangkan himpunan SEDANG menggunakan pendekatan berbentuk segitiga seperti pada gambar 3.1.



Gambar 3.1. Fungsi Keanggotaan pada Kriteria Bunga

Fungsi keanggotaan pada kriteria bunga dirumuskan sebagai berikut:

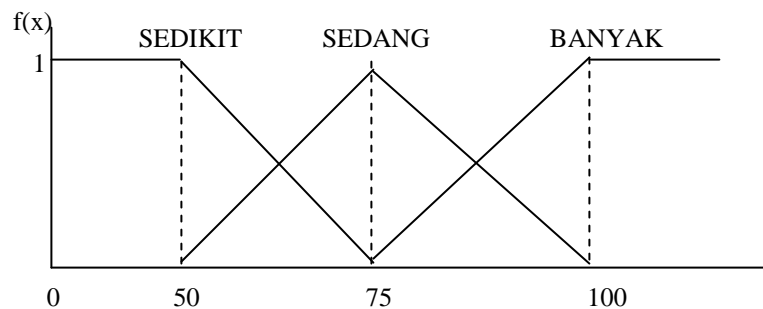
$$\begin{aligned} \mu_{\text{ Bunga RENDAH } [x_1]} & \begin{cases} 1 & x_1 \leq 0,5 \\ \frac{1 - x_1}{0,5} & 0,5 \leq x_1 \leq 1 \\ 0 & x_1 \geq 1 \end{cases} \\ \mu_{\text{ Bunga SEDANG } [x_1]} & \begin{cases} 0 & x_1 \leq 0,5 \text{ atau } x_1 \geq 1,5 \\ \frac{x_1 - 0,5}{0,5} & 0,5 \leq x_1 \leq 1 \\ \frac{1,5 - x_1}{0,5} & 1 \leq x_1 \leq 1,5 \end{cases} \\ \mu_{\text{ Bunga TINGGI } [x_1]} & \begin{cases} 0 & x_1 \leq 1 \\ \frac{x_1 - 1}{0,5} & 1 \leq x_1 \leq 1,5 \\ 1 & x_1 \geq 1,5 \end{cases} \end{aligned}$$

Tabel 3.1. Derajat Keanggotaan Kriteria Bunga

No.	Bank	Bunga	Derajat Keanggotaan ( $[x_1]$ )		
			RENDAH	SEDANG	TINGGI
1.	BRI	2,5%	0	0	1
2.	BNI	2,25%	0	0	1
3.	Mandiri	2,25%	0	0	1
4.	BCA	2,1%	0	0	1
5.	BTN	3%	0	0	1

### 3.2.2. Kriteria ATM

ATM merupakan tempat untuk mengambil uang secara tunai dari mesin ATM. ATM yang dimaksud disini adalah ATM yang dimiliki oleh pihak bank dan ATM bersama. Kriteria ATM dibagi menjadi 3 himpunan *fuzzy*, yaitu : SEDIKIT, SEDANG dan BANYAK. Himpunan SEDIKIT dan BANYAK menggunakan pendekatan fungsi keanggotaan yang berbentuk bahu, sedangkan himpunan SEDANG menggunakan pendekatan berbentuk segitiga seperti pada gambar 3.2.



Gambar 3.2. Fungsi Keanggotaan pada Kriteria ATM

Fungsi keanggotaan pada kriteria ATM dirumuskan sebagai berikut:

$$\mu_{\text{ATM SEDIKIT}} [x_2] = \begin{cases} 1 & x_2 \leq 50 \\ \frac{75 - x_2}{25} & 50 \leq x_2 \leq 75 \\ 0 & x_2 \geq 75 \end{cases}$$



$$\mu_{\text{ATM SEDANG}} [x_2] = \begin{cases} 0 & x_2 \leq 50 \text{ atau } x_2 \geq 100 \\ \frac{x_2 - 50}{25} & 50 \leq x_2 \leq 75 \\ \frac{100 - x_2}{25} & 75 \leq x_2 \leq 100 \end{cases}$$

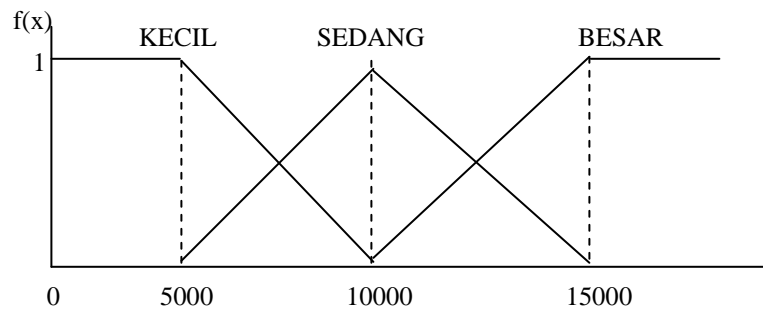
$$\mu_{\text{ATM BANYAK}} [x_2] = \begin{cases} 0 & x_2 \leq 75 \\ \frac{x_2 - 75}{25} & 75 \leq x_2 \leq 100 \\ 1 & x_2 \geq 100 \end{cases}$$

Tabel 3.2. Derajat Keanggotaan Kriteria ATM

No.	Bank	ATM	Derajat Keanggotaan ( $[x_2]$ )		
			SEDIKIT	SEDANG	BANYAK
1.	BRI	30	1	0	0
2.	BNI	31	1	0	0
3.	Mandiri	100	0	0	1
4.	BCA	158	0	0	1
5.	BTN	23	1	0	0

### 3.2.3. Kriteria Administrasi

Administrasi merupakan biaya potongan administrasi yang dibebankan kepada nasabah setiap bulan. Kriteria administrasi dibagi menjadi 3 himpunan *fuzzy*, yaitu : KECIL, SEDANG dan BESAR. Himpunan KECIL dan BESAR menggunakan pendekatan fungsi keanggotaan yang berbentuk bahu, sedangkan himpunan SEDANG menggunakan pendekatan berbentuk segitiga seperti pada gambar 3.3.



Gambar 3.3. Fungsi Keanggotaan pada Kriteria Administrasi

Fungsi keanggotaan pada kriteria administrasi dirumuskan sebagai berikut:

$$\mu_{\text{Administrasi KECIL}} [x_3] \begin{cases} 1 & x_3 \leq 5000 \\ \frac{10000 - x_3}{5000} & 5000 \leq x_3 \leq 10000 \\ 0 & x_3 \geq 10000 \end{cases}$$

$$\mu_{\text{Administrasi SEDANG}} [x_3] \begin{cases} 0 & x_3 \leq 5000 \text{ atau } x_3 \geq 15000 \\ \frac{x_3 - 5000}{5000} & 5000 \leq x_3 \leq 10000 \\ \frac{15000 - x_3}{5000} & 10000 \leq x_3 \leq 15000 \end{cases}$$

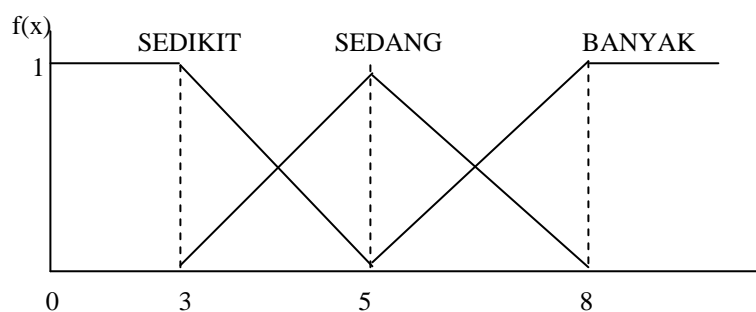
$$\mu_{\text{Administrasi BESAR}} [x_3] \begin{cases} 0 & x_3 \leq 10000 \\ \frac{x_3 - 10000}{5000} & 10000 \leq x_3 \leq 15000 \\ 1 & x_3 \geq 15000 \end{cases}$$

Tabel 3.3. Derajat Keanggotaan Kriteria Administrasi

No.	Bank	ADM	Derajat Keanggotaan ( $[x_3]$ )		
			KECIL	SEDANG	BESAR
1.	BRI	10000	0	1	0
2.	BNI	10000	0	1	0
3.	Mandiri	10000	0	1	0
4.	BCA	12000	0	0,6	0,4
5.	BTN	9000	0,2	0,8	0

### 3.2.4. Kriteria Pelayanan

Pelayanan merupakan jumlah *customer service* dan *teller* yang dapat melayani nasabah pada Bank tersebut. Kriteria pelayanan dibagi menjadi 3 himpunan *fuzzy*, yaitu : SEDIKIT, SEDANG dan BANYAK. Himpunan SEDIKIT dan BANYAK menggunakan pendekatan fungsi keanggotaan yang berbentuk bahu, sedangkan himpunan SEDANG menggunakan pendekatan berbentuk segitiga seperti pada gambar 3.4.



Gambar 3.4. Fungsi Keanggotaan pada Kriteria Pelayanan

Fungsi keanggotaan pada kriteria pelayanan dirumuskan sebagai berikut:

$$\begin{aligned} \mu_{\text{Pelayanan SEDIKIT}}^{[x_4]} & \begin{cases} 1 & x_4 \leq 3 \\ \frac{5 - x_4}{2} & 3 \leq x_4 \leq 5 \\ 0 & x_4 \geq 5 \end{cases} \\ \mu_{\text{Pelayanan SEDANG}}^{[x_4]} & \begin{cases} 0 & x_4 \leq 3 \text{ atau } x_4 \geq 8 \\ \frac{x_4 - 3}{2} & 3 \leq x_4 \leq 5 \\ \frac{8 - x_4}{3} & 5 \leq x_4 \leq 8 \end{cases} \\ \mu_{\text{Pelayanan BANYAK}}^{[x_4]} & \begin{cases} 0 & x_4 \leq 5 \\ \frac{x_4 - 5}{3} & 5 \leq x_4 \leq 8 \\ 1 & x_4 \geq 8 \end{cases} \end{aligned}$$

Tabel 3.4. Derajat Keanggotaan Kriteria Pelayanan

No.	Bank	Pelayanan	Derajat Keanggotaan ( $[x_4]$ )		
			SEDIKIT	SEDANG	BANYAK
1.	BRI	3	1	0	0
2.	BNI	4	0,5	0,5	0
3.	Mandiri	4	0,5	0,5	0
4.	BCA	3	1	0	0
5.	BTN	3	1	0	0

## BAB IV

### PERANCANGAN SISTEM

Perancangan sistem pendukung keputusan pemilihan bank dibuat dengan menggunakan use case diagram, class diagram dan activity diagram.

#### 4.1. Analisa Abbot

Sistem yang akan **dibangun** terdiri dari dua **pengguna** yaitu **admin** dan **user** dimana **admin** harus **login** dengan **memasukkan** **user name** dan **password** ke sistem terlebih dahulu untuk dapat **menggunakan** **menu admin** sedangkan **user** dapat **masuk** tanpa **login**. Jika **login admin** benar maka **admin** dapat **memasukkan** **data admin** yang terdiri dari **data kriteria** yang terdiri dari **kdkriteria, nmkriteria, jnskriteria, fungsi, nilfungsi1, nilfungsi2** dan **nilfungsi3** dengan proses **tambah, ubah, hapus, simpan, batal** pada **data kriteria** tersebut. **Data Bank** yang terdiri dari **idbank, nmbank, alamat, telp, bunga, jmlatm, administrasi, pelayanan** dengan proses **tambah, ubah, hapus, simpan, batal** pada **data Bank** tersebut. **Data nilai** yang terdiri dari **kdkriteria, idbank, nilfuzzy** dengan proses **tambah, ubah, hapus, simpan, batal** pada **data nilai** tersebut.

User dapat **menggunakan** **menu SPK** dengan cara **memilih** **kriteria** yang **diinginkan** kemudian sistem akan **memberikan** **rekomendasi pemilihan Bank** yang terbaik **menurut** **kriteria** yang **dipilih**.

## **4.2. Kata Benda**

### **4.2.1. Calon Kata Benda**

Dari Analisa Abbot diatas di dapat beberapa kata benda yaitu: pengguna, admin, user, login, username, password, menu, login, data, kriteria, kdkriteria, nmkriteria, jnskriteria, fungsi, nilfungsi1, nilfungsi2, nilfungsi3, bank, idbank, nmbank, bunga, jmlatm, administrasi, pelayanan, nilai, nilfuzzy, SPK, rekomendasi.

### **4.2.2. Eliminasi**

Eliminasi terhadap calon kata benda didapatkan hasil yaitu :

#### **1. Redundant Class**

Menghilangkan calon kelas yang berlebihan diantaranya surat pengguna, admin, user, login, username, password, menu, login, data.

#### **2. Kelas**

Berdasarkan eliminasi diatas maka diperoleh kelas-kelas sebagai berikut : kriteria, bank dan nilai.

## **4.3. Kata Kerja**

### **4.3.1. Calon Kata Kerja**

Dari Analisa Abbot diatas di dapat beberapa kata kerja yaitu: dibangun, memasukkan, menggunakan, memasukkan, tambah, ubah,

hapus, simpan, batal, memilih, diinginkan, memberikan, menurut, dipilih.

#### 4.3.2. Eliminasi

Eliminasi terhadap calon kata kerja didapatkan hasil yaitu :

1. Redundant Methode

dibangun, memasukkan, menggunakan, memasukkan, memilih, diinginkan, memberikan, menurut, dipilih.

2. Metode

Berdasarkan eliminasi diatas maka diperoleh metode yaitu tambah, ubah, hapus, simpan, batal.

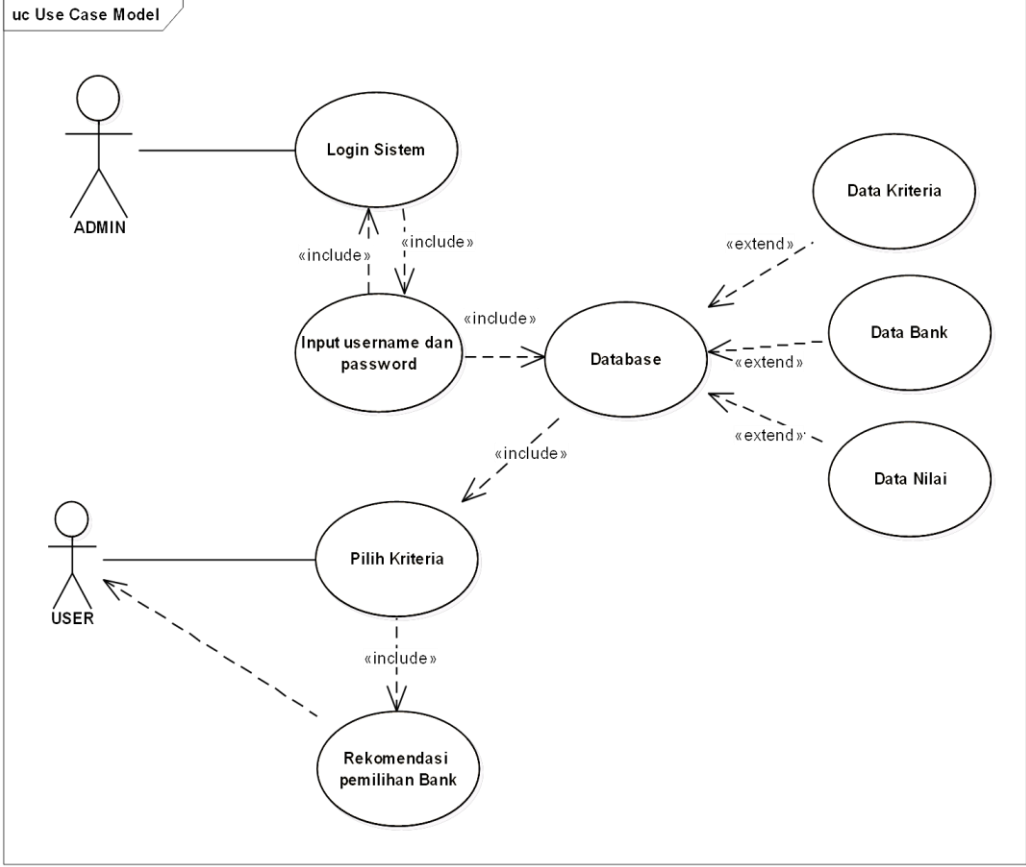
#### 4.4. Kelas Atribut

Dari Analisa Abbot diatas di dapat kelas atribut seperti pada tabel 4.1

Tabel 4.1. Kelas Atribut

No.	Kelas	Atribut
1.	Kriteria	kdkriteria, nmkriteria, jnskriteria, fungsi, nilfungsi1, nilfungsi2, nilfungsi3
2.	Bank	idbank, nmbank, bunga, jmlatm, administrasi, pelayanan
3.	Nilai	Kdkriteria, idbank, nilfuzzy

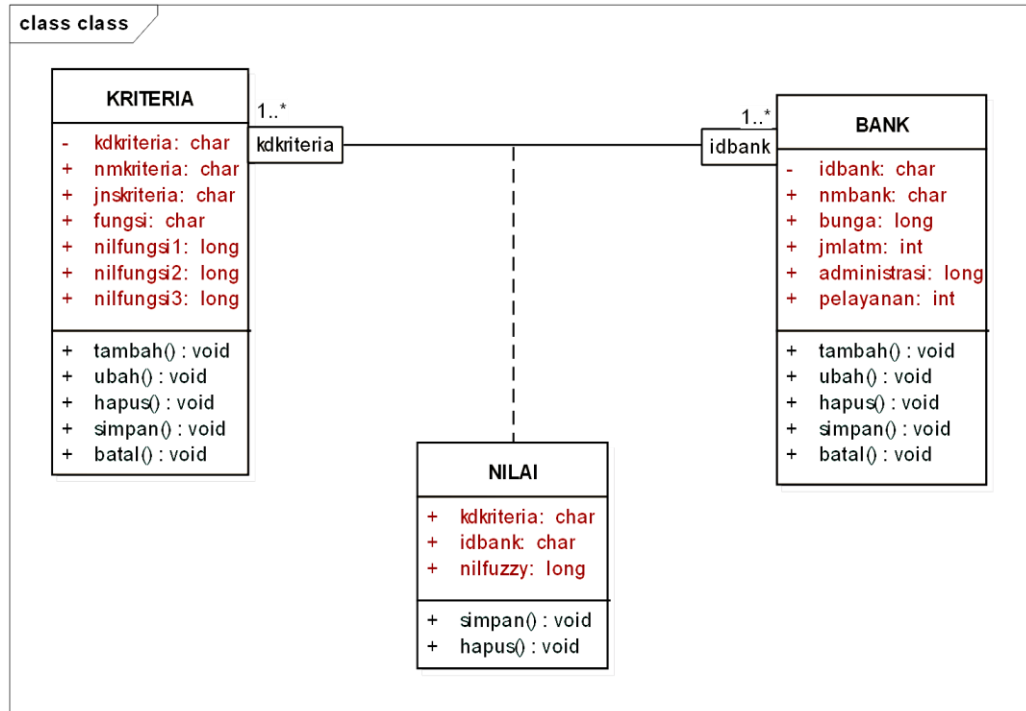
### 4.5. Use Case Diagram



Gambar 4.1. Use Case Diagram

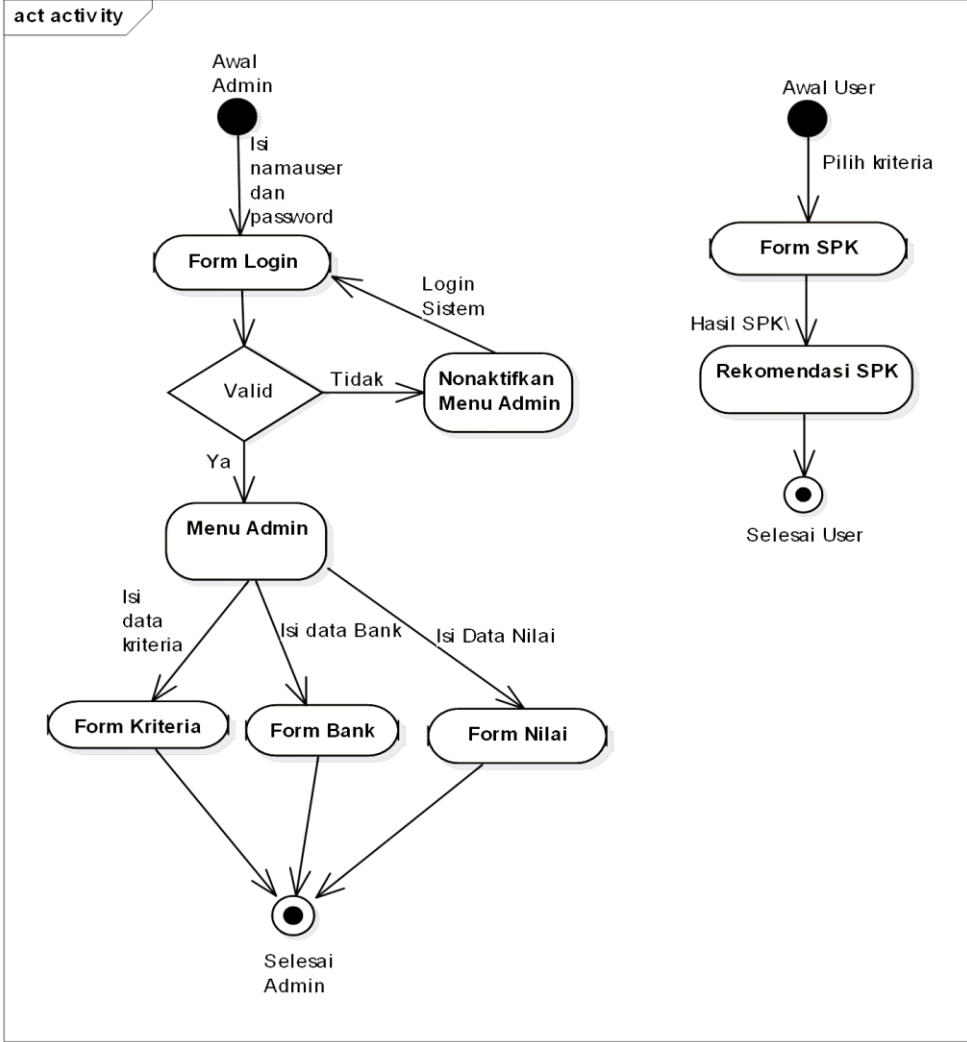


## 4.6. Class Diagram



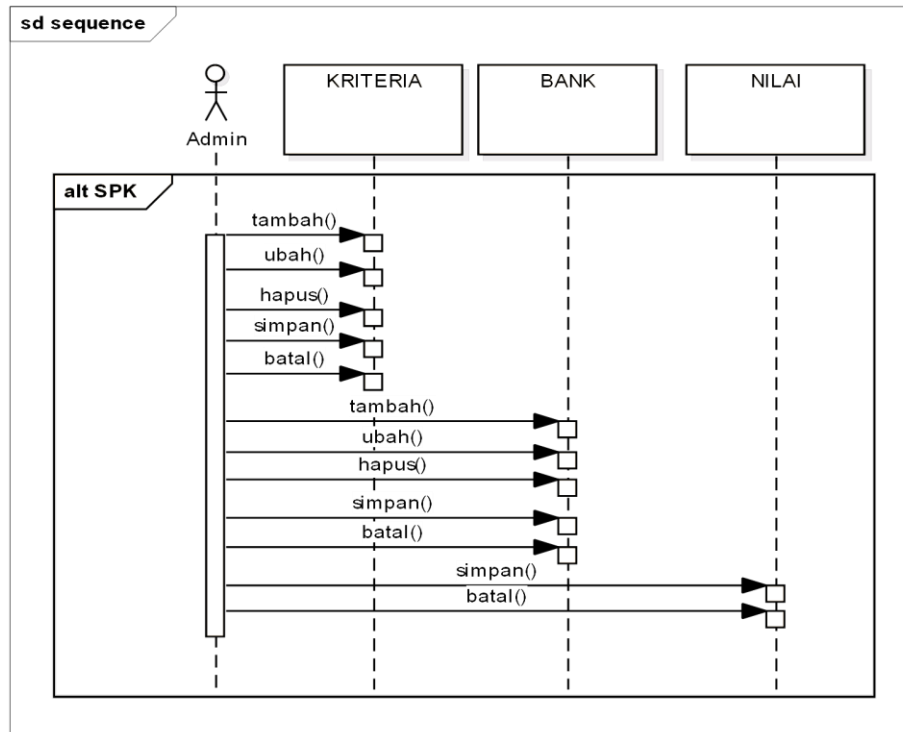
Gambar 4.2. Class Diagram

4.7. Activity Diagram



Gambar 4.3. Activity Diagram

## 4.8. Sequence Diagram



Gambar 4.4. Sequence Diagram

## 4.9. Struktur Database

Struktur database sistem pendukung keputusan pemilihan Bank dibuat berdasarkan class diagram seperti pada gambar 4.4.

### 4.9.1. Tabel Kriteria

Tabel kriteria digunakan untuk menyimpan data kriteria dengan kode kunci utama kdkriteria. Tabel kriteria ditunjukkan seperti pada tabel 4.2.

Tabel 4.2. Tabel Kriteria

<b>Field Name</b>	<b>Type</b>	<b>Size</b>	<b>Key</b>	<b>Keterangan</b>
Kdkriteria	Varchar	3	*	Kode Kriteria
Nmkriteria	Varchar	30		Nama Kriteria
Jnskriteria	Varchar	20		Jenis Kriteria
Fungsi	Varchar	15		Fungsi
Nilfungsi1	Double			Nilai Fungsi 1
Nilfungsi2	Double			Nilai Fungsi 2
Nilfungsi3	Double			Nilai Fungsi 3

#### 4.9.2. Tabel Bank

Tabel bank digunakan untuk menyimpan data bank dengan kode kunci utama idbank. Tabel kriteria ditunjukkan seperti pada tabel 4.3

Tabel 4.3. Tabel Bank

<b>Field Name</b>	<b>Type</b>	<b>Size</b>	<b>Key</b>	<b>Keterangan</b>
Idbank	Varchar	3	*	ID Bank
Nmbank	Varchar	30		Nama Bank
Bunga	Double			Bunga
Jmlatm	Int			Jumlah ATM
Administrasi	Double			Administrasi
Pelayanan	Int			Pelayanan

#### 4.9.3. Tabel Nilai

Tabel nilai digunakan untuk menyimpan data nilai dengan kode kunci tamu idbank dan kdkriteria. Tabel kriteria ditunjukkan seperti pada tabel 4.4.

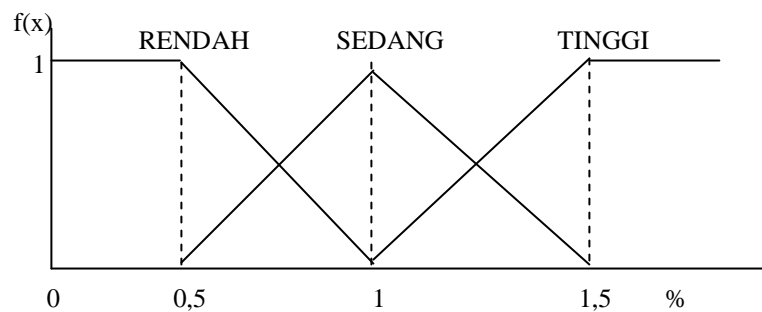
Tabel 4.4. Tabel Nilai

Field Name	Type	Size	Key	Keterangan
Kdkriteria	Varchar	3		Kode Kriteria
Idbank	Varchar	3		ID Bank
Nilfuzzy	Double			Nilai Fuzzy

#### 4.10. Model SPK

##### 4.10.1. Model Bunga

Bunga merupakan suku bunga Bank yang diberikan kepada nasabah setiap bulan. Model bunga dibagi menjadi 3 himpunan *fuzzy*, yaitu : RENDAH, SEDANG dan TINGGI. Himpunan RENDAH dan TINGGI menggunakan pendekatan fungsi keanggotaan yang berbentuk bahu, sedangkan himpunan SEDANG menggunakan pendekatan berbentuk segitiga seperti pada gambar 4.5.



Gambar 4.5. Fungsi Keanggotaan pada Model Bunga

Fungsi keanggotaan pada model bunga dirumuskan sebagai berikut:

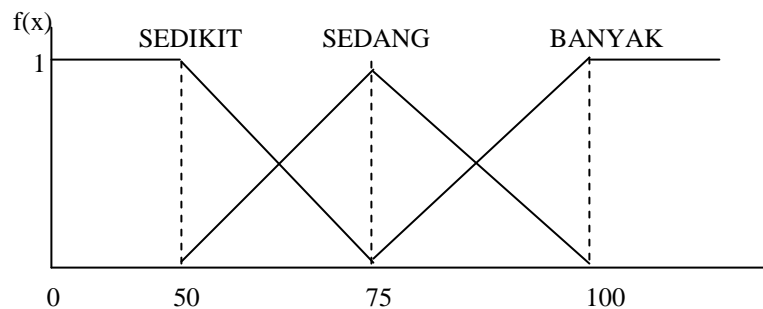
$$\begin{aligned}
 \mu_{\text{ Bunga RENDAH } [x_1]} & \begin{cases} 1 & x_1 \leq 0,5 \\ \frac{1 - x_1}{0,5} & 0,5 \leq x_1 \leq 1 \\ 0 & x_1 \geq 1 \end{cases} \\
 \mu_{\text{ Bunga SEDANG } [x_1]} & \begin{cases} 0 & x_1 \leq 0,5 \text{ atau } x_1 \geq 1,5 \\ \frac{x_1 - 0,5}{0,5} & 0,5 \leq x_1 \leq 1 \\ \frac{1,5 - x_1}{0,5} & 1 \leq x_1 \leq 1,5 \end{cases} \\
 \mu_{\text{ Bunga TINGGI } [x_1]} & \begin{cases} 0 & x_1 \leq 1 \\ \frac{x_1 - 1}{0,5} & 1 \leq x_1 \leq 1,5 \\ 1 & x_1 \geq 1,5 \end{cases}
 \end{aligned}$$

Tabel 4.5. Derajat Keanggotaan Kriteria Bunga

No.	Bank	Bunga	Derajat Keanggotaan ( $[x_1]$ )		
			RENDAH	SEDANG	TINGGI
1.	BRI	2,5%	0	0	1
2.	BNI	2,25%	0	0	1
3.	Mandiri	2,25%	0	0	1
4.	BCA	2,1%	0	0	1
5.	BTN	3%	0	0	1

#### 4.10.2. Model ATM

ATM merupakan tempat untuk mengambil uang secara tunai dari mesin ATM. ATM yang dimaksud disini adalah ATM yang dimiliki oleh pihak bank dan ATM bersama. Model ATM dibagi menjadi 3 himpunan *fuzzy*, yaitu : SEDIKIT, SEDANG dan BANYAK. Himpunan SEDIKIT dan BANYAK menggunakan pendekatan fungsi keanggotaan yang berbentuk bahu, sedangkan himpunan SEDANG menggunakan pendekatan berbentuk segitiga seperti pada gambar 4.6.



Gambar 4.6. Fungsi Keanggotaan pada Model ATM

Fungsi keanggotaan pada model ATM dirumuskan sebagai berikut:

$$\mu_{\text{ATM SEDIKIT}} [x_2] = \begin{cases} 1 & x_2 \leq 50 \\ \frac{75 - x_2}{25} & 50 \leq x_2 \leq 75 \\ 0 & x_2 \geq 75 \end{cases}$$

$$\mu_{\text{ATM SEDANG}} [x_2] = \begin{cases} 0 & x_2 \leq 50 \text{ atau } x_2 \geq 100 \\ \frac{x_2 - 50}{25} & 50 \leq x_2 \leq 75 \\ \frac{100 - x_2}{25} & 75 \leq x_2 \leq 100 \end{cases}$$

$$\mu_{\text{ATM BANYAK}} [x_2] = \begin{cases} 0 & x_2 \leq 75 \\ \frac{x_2 - 75}{25} & 75 \leq x_2 \leq 100 \\ 1 & x_2 \geq 100 \end{cases}$$

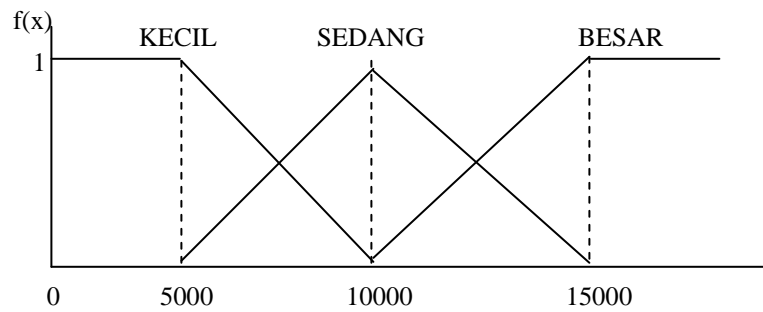
Tabel 4.6. Derajat Keanggotaan Kriteria ATM

No.	Bank	ATM	Derajat Keanggotaan ( $[x_2]$ )		
			SEDIKIT	SEDANG	BANYAK
1.	BRI	30	1	0	0
2.	BNI	31	1	0	0
3.	Mandiri	100	0	0	1
4.	BCA	158	0	0	1
5.	BTN	23	1	0	0

#### 4.10.3. Model Administrasi

Administrasi merupakan biaya potongan administrasi yang dibebankan kepada nasabah setiap bulan. Model administrasi dibagi menjadi 3 himpunan *fuzzy*, yaitu : KECIL, SEDANG dan BESAR. Himpunan KECIL dan BESAR menggunakan pendekatan fungsi keanggotaan yang berbentuk bahu, sedangkan himpunan SEDANG menggunakan pendekatan berbentuk segitiga seperti pada gambar 4.7.





Gambar 4.7. Fungsi Keanggotaan pada Model Administrasi

Fungsi keanggotaan pada model administrasi dirumuskan sebagai berikut:

$$\mu_{\text{Administrasi KECIL}} [x_3] \begin{cases} 1 & x_3 \leq 5000 \\ \frac{10000 - x_3}{5000} & 5000 \leq x_3 \leq 10000 \\ 0 & x_3 \geq 10000 \end{cases}$$

$$\mu_{\text{Administrasi SEDANG}} [x_3] \begin{cases} 0 & x_3 \leq 5000 \text{ atau } x_3 \geq 15000 \\ \frac{x_3 - 5000}{5000} & 5000 \leq x_3 \leq 10000 \\ \frac{15000 - x_3}{5000} & 10000 \leq x_3 \leq 15000 \end{cases}$$

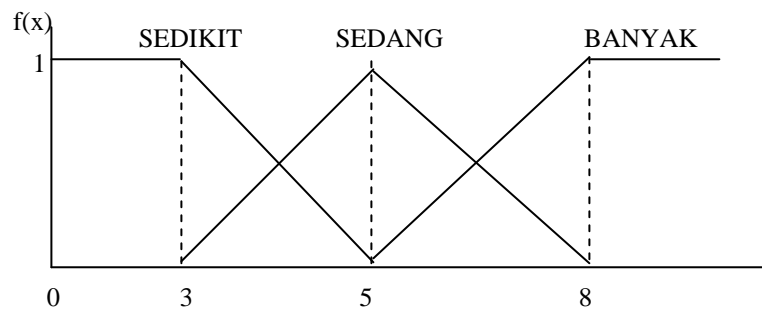
$$\mu_{\text{Administrasi BESAR}} [x_3] \begin{cases} 0 & x_3 \leq 10000 \\ \frac{x_3 - 10000}{5000} & 10000 \leq x_3 \leq 15000 \\ 1 & x_3 \geq 15000 \end{cases}$$

Tabel 4.7. Derajat Keanggotaan Kriteria Administrasi

No.	Bank	ADM	Derajat Keanggotaan ( $[x_3]$ )		
			KECIL	SEDANG	BESAR
1.	BRI	10000	0	1	0
2.	BNI	10000	0	1	0
3.	Mandiri	10000	0	1	0
4.	BCA	12000	0	0,6	0,4
5.	BTN	9000	0,2	0,8	0

#### 4.10.4. Model Pelayanan

Pelayanan merupakan jumlah *customer service* dan *teller* yang dapat melayani nasabah pada Bank tersebut. Model pelayanan dibagi menjadi 3 himpunan *fuzzy*, yaitu : SEDIKIT, SEDANG dan BANYAK. Himpunan SEDIKIT dan BANYAK menggunakan pendekatan fungsi keanggotaan yang berbentuk bahu, sedangkan himpunan SEDANG menggunakan pendekatan berbentuk segitiga seperti pada gambar 4.8.



Gambar 4.8. Fungsi Keanggotaan pada Model Pelayanan

Fungsi keanggotaan pada model pelayanan dirumuskan sebagai berikut:

$$\begin{aligned} \mu_{\text{Pelayanan SEDIKIT}} [x_4] & \begin{cases} 1 & x_4 \leq 3 \\ \frac{5 - x_4}{2} & 3 \leq x_4 \leq 5 \\ 0 & x_4 \geq 5 \end{cases} \\ \mu_{\text{Pelayanan SEDANG}} [x_4] & \begin{cases} 0 & x_4 \leq 3 \text{ atau } x_4 \geq 8 \\ \frac{x_4 - 3}{2} & 3 \leq x_4 \leq 5 \\ \frac{8 - x_4}{3} & 5 \leq x_4 \leq 8 \end{cases} \\ \mu_{\text{Pelayanan BANYAK}} [x_4] & \begin{cases} 0 & x_4 \leq 5 \\ \frac{x_4 - 5}{3} & 5 \leq x_4 \leq 8 \\ 1 & x_4 \geq 8 \end{cases} \end{aligned}$$

Tabel 4.8. Derajat Keanggotaan Kriteria Pelayanan

No.	Bank	Pelayanan	Derajat Keanggotaan ( $[x_4]$ )		
			SEDIKIT	SEDANG	BANYAK
1.	BRI	3	1	0	0
2.	BNI	4	0,5	0,5	0
3.	Mandiri	4	0,5	0,5	0
4.	BCA	3	1	0	0
5.	BTN	3	1	0	0

#### 4.10.5. Domain

Domain himpunan fuzzy adalah keseluruhan nilai yang diizinkan dalam semesta pembicaraan dan boleh dioperasikan dalam himpunan fuzzy. Domain himpunan fuzzy yaitu :

1. Bunga

Bunga ( $X_1$ )	Domain
RENDAH	[0 1]
SEDANG	[0.5 1.5]
TINGGI	[1 $+\infty$ ]

2. ATM

ATM ( $X_2$ )	Domain
SEDIKIT	[0 75]
SEDANG	[50 100]
BANYAK	[75 $+\infty$ ]

3. Administrasi

Administrasi ( $X_3$ )	Domain
KECIL	[0 10000]
SEDANG	[5000 15000]
BESAR	[10000 $+\infty$ ]

4. Pelayanan

Pelayanan ( $X_4$ )	Domain
SEDIKIT	[0 5]
SEDANG	[3 8]
BANYAK	[5 $+\infty$ ]

## 4.11. Perancangan Interface

### 4.11.1. Perancangan Form Utama

Form utama merupakan halaman utama dalam menjalankan program aplikasi yang mengacu pada use case diagram gambar 4.1. Pada halaman aplikasi ini terdapat beberapa menu, yaitu menu file, admin dan SPK. Perancangan form utama seperti pada gambar 4.9.

File	Admin	SPK
Login	Kriteria	SPK
Logout	Bank	
Keluar	Nilai	

**Sistem Pendukung Keputusan  
Pemilihan Bank**

Gambar 4.9. Perancangan Form Utama

#### 4.11.2. Perancangan Form Login

Form login digunakan admin untuk masuk ke menu admin yang mengacu pada use case diagram gambar 4.1. Perancangan form login seperti pada gambar 4.10.

File	Admin	SPK
<b>Login</b>		
Logout		
Keluar		

User Name	<input type="text"/>
Password	<input type="password"/>
	<input type="button" value="Login"/>
	<input type="button" value="Batal"/>
	<input type="button" value="Keluar"/>

Gambar 4.10. Perancangan Form Login

#### 4.11.3. Perancangan Form Kriteria

Form kriteria digunakan admin untuk mengisi data kriteria yang mengacu pada use case diagram gambar 4.1. Perancangan form kriteria seperti pada gambar 4.11.

<b>File</b>	<b>Admin</b>	<b>SPK</b>			
<table border="1" style="margin-left: auto; margin-right: auto;"> <tr><td><b>Kriteria</b></td></tr> <tr><td>Bank</td></tr> <tr><td>Nilai</td></tr> </table> <p style="text-align: center;"><b>Sistem Pendukung Keputusan Pemilihan Bank</b></p>			<b>Kriteria</b>	Bank	Nilai
<b>Kriteria</b>					
Bank					
Nilai					

Kode Kriteria	<input type="text"/>						
Nama Kriteria	<input type="text"/>						
Jenis Kriteria	<input type="text"/>						
Fungsi	<input type="text"/>						
Nilai Fungsi	<input type="text"/> <input type="text"/> <input type="text"/>						
<table border="1" style="width: 100%; height: 100px;"> <tr><td style="width: 10%;"></td><td style="width: 10%;"></td><td></td></tr> <tr><td></td><td></td><td></td></tr> </table>							
<table border="1" style="width: 100%;"> <tr> <td>Tambah</td> <td>Ubah</td> <td>Hapus</td> <td>Simpan</td> <td>Batal</td> </tr> </table>		Tambah	Ubah	Hapus	Simpan	Batal	
Tambah	Ubah	Hapus	Simpan	Batal			

Gambar 4.11. Perancangan Form Kriteria

#### 4.11.4. Perancangan Form Bank

Form bank digunakan admin untuk mengisi data bank yang mengacu pada use case diagram gambar 4.1. Perancangan form bank seperti pada gambar 4.12.

File	Admin	SPK			
<table border="1"> <tr><td>Kriteria</td></tr> <tr><td><b>Bank</b></td></tr> <tr><td>Nilai</td></tr> </table>			Kriteria	<b>Bank</b>	Nilai
Kriteria					
<b>Bank</b>					
Nilai					
<h2>Sistem Pendukung Keputusan Pemilihan Bank</h2>					

ID Bank	<input type="text"/>					
Nama Bank	<input type="text"/>					
Bunga (%)	<input type="text"/>					
Jumlah ATM	<input type="text"/>					
Administrasi	<input type="text"/>					
Pelayanan	<input type="text"/>					
<input type="text"/>	<input type="text"/>					
<input type="text"/>	<input type="text"/>					
<table border="1"> <tr> <td>Tambah</td> <td>Ubah</td> <td>Hapus</td> <td>Simpan</td> <td>Batal</td> </tr> </table>		Tambah	Ubah	Hapus	Simpan	Batal
Tambah	Ubah	Hapus	Simpan	Batal		

Gambar 4.12. Perancangan Form Bank

#### 4.11.5. Perancangan Form Nilai

Form nilai digunakan admin untuk melihat data nilai *Fuzzy* yang mengacu pada use case diagram gambar 4.1. Perancangan form nilai seperti pada gambar 4.13.



File	Admin	SPK			
<table border="1"> <tr><td>Kriteria</td></tr> <tr><td>Bank</td></tr> <tr><td><b>Nilai</b></td></tr> </table>			Kriteria	Bank	<b>Nilai</b>
Kriteria					
Bank					
<b>Nilai</b>					
<p><b>Sistem Pendukung Keputusan Pemilihan Bank</b></p>					

<<	<	>	>>

Gambar 4.13. Perancangan Form Nilai

#### 4.11.6. Perancangan Form SPK

Form SPK ini digunakan untuk memasukkan data pemilihan bank yang terbaik yang mengacu pada use case diagram gambar 4.1. Perancangan form SPK seperti pada gambar 4.14.

File	Admin	SPK								
<table border="1" style="margin: auto;"> <tr><td><b>SPK</b></td></tr> </table>			<b>SPK</b>							
<b>SPK</b>										
<table border="1" style="margin: auto;"> <tr> <td style="width: 100px;">Nama</td> <td style="width: 200px; height: 20px;"></td> </tr> <tr> <td>Alamat</td> <td style="height: 20px;"></td> </tr> <tr> <td style="text-align: center;">Login</td> <td style="text-align: center;">Batal</td> </tr> <tr> <td style="text-align: center;">Keluar</td> <td></td> </tr> </table>			Nama		Alamat		Login	Batal	Keluar	
Nama										
Alamat										
Login	Batal									
Keluar										

Bunga	<input type="text"/>
ATM	<input type="text"/>
Administrasi	<input type="text"/>
Pelayanan	<input type="text"/>
<input type="button" value="Cetak"/> <input type="button" value="Batal"/>	

Gambar 4.14. Perancangan Form SPK

#### 4.11.7. Perancangan Hasil SPK

Form hasil SPK ini digunakan untuk mencetak data pemilihan bank yang terbaik yang mengacu pada use case diagram gambar 4.1. Perancangan form hasil SPK seperti pada gambar 4.15.

Bunga	<input type="text"/>
ATM	<input type="text"/>
Administrasi	<input type="text"/>
Pelayanan	<input type="text"/>
<input type="button" value="Cetak"/> <input type="button" value="Batal"/>	

**Rekomendasi Pemilihan Bank**

---

Kriteria Yang Dimasukkan

Grafik

Gambar 4.15. Perancangan Hasil SPK



## 1. File

Menu file digunakan untuk melakukan login ke menu admin dan logout dari menu admin.

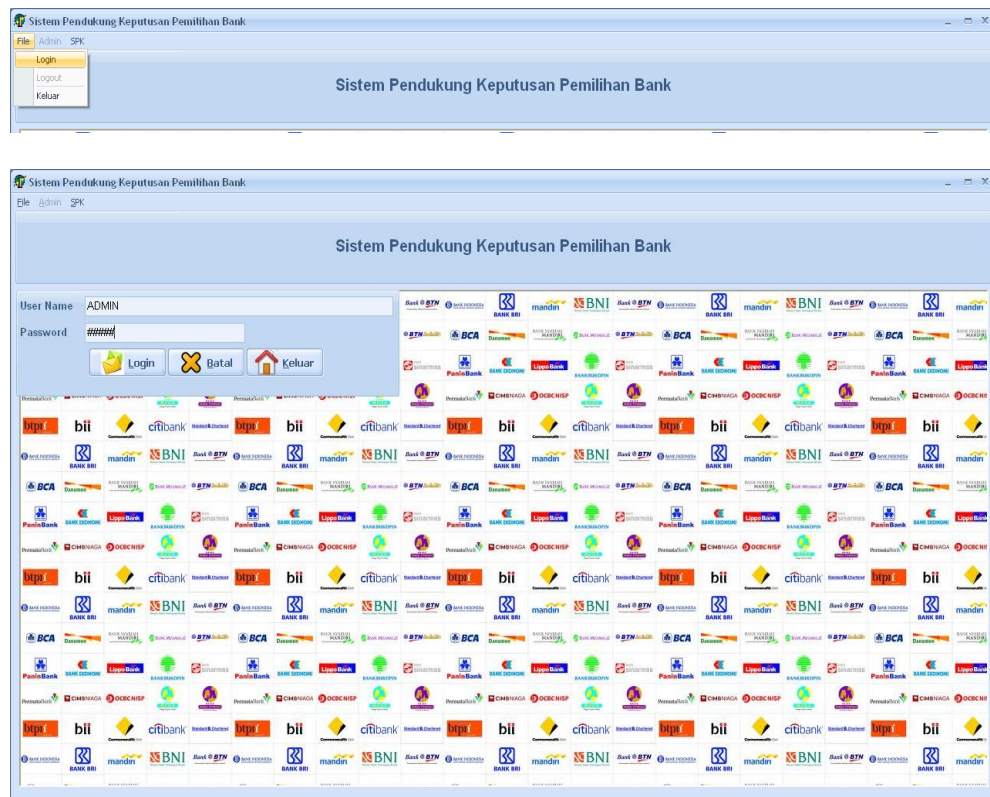
## 2. Admin

Menu admin digunakan admin untuk mengisi data kriteria, data bank dan melihat data nilai *Fuzzy*.

## 3. SPK

Menu SPK digunakan untuk melakukan proses pemilihan Bank dengan Logika *Fuzzy*.

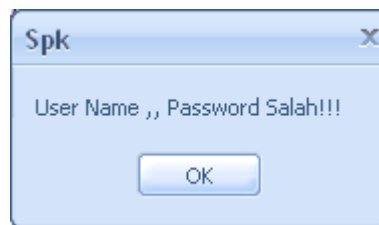
## 5.2. Login



Gambar 5.2. Form Login

Form login digunakan untuk masuk ke menu admin yang berisi data kriteria, data bank dan data nilai. Form login dapat dilihat pada gambar 5.2 yang merupakan implementasi dari perancangan program pada gambar 4.10.

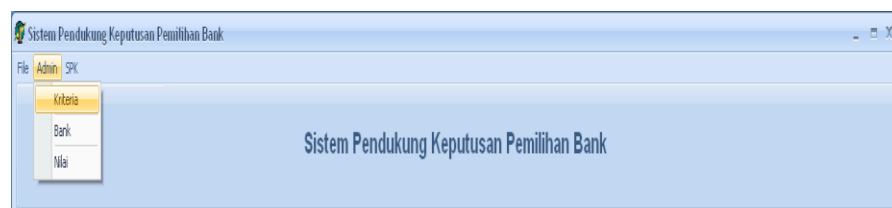
Untuk masuk ke menu admin isi user name dan password yang benar kemudian klik tombol login, jika user name dan password benar maka menu admin akan aktif dan dapat digunakan, jika user name dan password salah maka akan ditampilkan pesan seperti gambar 5.3.



Gambar 5.3. Pesan Salah

### 5.3. Menu Admin

#### 5.3.1. Form Kriteria



The screenshot shows a software window titled "Kriteria". It contains several input fields and a table. The input fields are: "Kode Kriteria" (K01), "Nama Kriteria" (BUNGA), "Jenis Kriteria" (RENDAH), "Fungsi" (BAHU KIRI), and "Nilai Fungsi" (0.5, 1, 0). Below these is a table with 7 columns: Kode Kriteria, Kriteria, Jenis, Fungsi, Nilai 1, Nilai 2, and Nilai 3. The table lists 11 criteria (K01 to K11). At the bottom of the window are six buttons: "Tambah", "Ubah", "Hapus", "Keluar", "Simpan", and "Batal".

Kode Kriteria	Kriteria	Jenis	Fungsi	Nilai 1	Nilai 2	Nilai 3
K01	BUNGA	RENDAH	BAHU KIRI	0.5	1	0
K02	BUNGA	SEDANG	SEGITIGA	0.5	1	1.5
K03	BUNGA	TINGGI	BAHU KANAN	1	1.5	0
K04	ATM	SEDIKIT	BAHU KIRI	50	75	0
K05	ATM	SEDANG	SEGITIGA	50	75	100
K06	ATM	BANYAK	BAHU KANAN	75	100	0
K07	ADMINISTRASI	KECIL	BAHU KIRI	5000	10000	0
K08	ADMINISTRASI	SEDANG	SEGITIGA	5000	10000	15000
K09	ADMINISTRASI	BESAR	BAHU KANAN	10000	15000	0
K10	PELAYANAN	SEDIKIT	BAHU KIRI	3	5	0
K11	PELAYANAN	SEDANG	SEGITIGA	3	5	8

Gambar 5.4. Form Kriteria

Form kriteria digunakan untuk memasukkan nama kriteria *Fuzzy*. Form kriteria dapat dilihat pada gambar 5.4 yang merupakan implementasi dari perancangan program pada gambar 4.11. Pada form kriteria terdapat 6 tombol yang dapat digunakan yaitu tombol tambah untuk menambah data kriteria *Fuzzy*, tombol ubah untuk mengubah data kriteria *Fuzzy*, tombol hapus untuk menghapus data kriteria *Fuzzy*, tombol keluar untuk keluar dari form kriteria, tombol simpan untuk menyimpan data kriteria *Fuzzy*, tombol batal untuk membatalkan pengisian data kriteria *Fuzzy*.

### 5.3.2. Form Bank

The screenshot shows the 'Sistem Pendukung Keputusan Pemilihan Bank' application. The main window displays the title 'Sistem Pendukung Keputusan Pemilihan Bank' and a menu with options: File, Admin, SPK, Kriteria, Bank, and Nilai. The 'Bank' form is open, showing the following data:

ID Bank	Nama Bank	Bunga	ATM	Administrasi	Pelayanan
001	BNI	1	50	10000	5

Buttons at the bottom: Tambah, Ubah, Hapus, Keluar, Simpan, Batal.

Gambar 5.5. Form Bank

Form bank digunakan untuk memasukkan nama Bank. Form Bank dapat dilihat pada gambar 5.5 yang merupakan implementasi dari perancangan program pada gambar 4.12. Pada form Bank terdapat 6 tombol yang dapat digunakan yaitu tombol tambah untuk menambah data Bank, tombol ubah untuk mengubah data Bank, tombol hapus untuk menghapus data Bank, tombol keluar untuk keluar dari form Bank, tombol simpan untuk menyimpan data Bank, tombol batal untuk membatalkan pengisian data Bank.

### 5.3.3. Form Nilai

ID Bank	Nama Bank	Kode Kriteria	Nama Kriteria	Nilai Fuzzy
001	BNI	K01	BUNGA	0
001	BNI	K02	BUNGA	1
001	BNI	K03	BUNGA	0
001	BNI	K04	ATM	1
001	BNI	K05	ATM	0
001	BNI	K06	ATM	0
001	BNI	K07	ADMINISTRASI	0
001	BNI	K08	ADMINISTRASI	1
001	BNI	K09	ADMINISTRASI	0
001	BNI	K10	PELAYANAN	0
001	BNI	K11	PELAYANAN	1
001	BNI	K12	PELAYANAN	0

Gambar 5.6. Form Nilai

Form nilai digunakan untuk melihat daftar nilai *Fuzzy*. Form nilai dapat dilihat pada gambar 5.6 yang merupakan implementasi dari perancangan program pada gambar 4.13. Pada form nilai terdapat 5 tombol yang dapat digunakan yaitu tombol first untuk melihat data awal nilai *Fuzzy*, tombol prev untuk melihat data nilai *Fuzzy* sebelumnya, tombol next untuk melihat data nilai *Fuzzy* selanjutnya, tombol last untuk melihat data terakhir nilai *Fuzzy* dan tombol keluar untuk keluar dari form nilai.



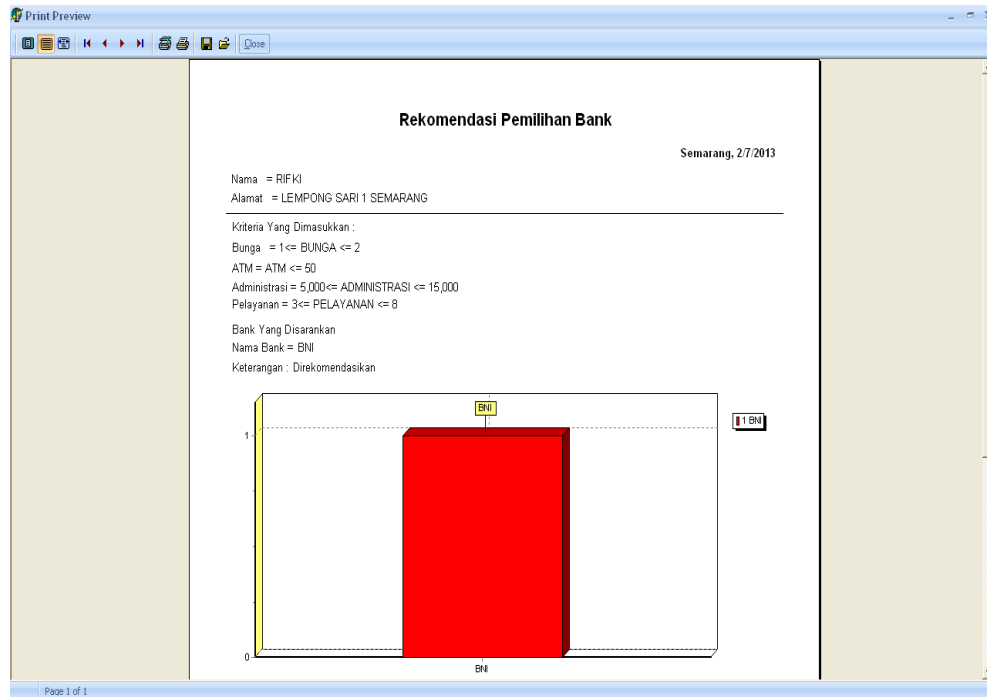
## 5.4. Form SPK

The image displays three screenshots of the SPK (Sistem Pendukung Keputusan Pemilihan Bank) application. The top screenshot shows the main menu with a 'SPK' button. The middle screenshot shows the login screen with fields for 'Nama' (RIFKI) and 'Alamat' (LEMPONG SARI 1 SEMARANG), and buttons for 'Login', 'Batal', and 'Keluar'. The bottom screenshot shows the decision criteria selection screen with dropdown menus for 'Bunga' (1 <= BUNGA <= 2), 'ATM' (ATM <= 50), 'Administrasi' (5,000 <= ADMINISTRASI <= 15,000), and 'Pelayanan' (3 <= PELAYANAN <= 8), along with 'Cetak' and 'Batal' buttons.

Gambar 5.7. Form SPK

Form SPK digunakan untuk melakukan pemilihan bank dimana hasilnya akan ditampilkan dalam bentuk laporan dan grafik. Form SPK dapat dilihat pada gambar 5.7 yang merupakan implementasi dari perancangan program pada gambar 4.14. Pada form spk terdapat 2 tombol yang dapat digunakan yaitu tombol cetak digunakan untuk memproses pemilihan Bank berdasarkan kriteria yang dipilih dan hasilnya akan diurutkan berdasarkan *firestrength* dari masing-masing Bank dari nilai terbesar sampai terkecil dimana nilai terbesar adalah 1 dan nilai terkecil adalah 0 seperti pada gambar 5.8 yang merupakan implementasi dari

perancangan program pada gambar 4.15 dan tombol batal digunakan untuk membatalkan pemilihan Bank atau mengulang proses pemilihan Bank.



Gambar 5.8. Hasil SPK

## 5.5. Kebutuhan Sistem

Dalam merancang dan membangun sistem pendukung keputusan pemilihan Bank ada beberapa spesifikasi perangkat keras dan perangkat lunak yang dibutuhkan yaitu :

### 5.5.1. Kebutuhan Perangkat Keras

Perangkat keras adalah komponen fisik peralatan yang membentuk sistem komputer serta peralatan lain yang mendukung komputer dalam menjalankan tugasnya. Sifat umum dari perangkat

keras adalah dapat dilihat dan dipegang bentuk fisiknya. Perangkat

keras yang dibutuhkan untuk menjalankan aplikasi ini yaitu :

1. Processor Intel (R) Pentium IV Dual Core E2180 2,0 GHz
2. Memori 1 GB DDR2 PC 5300
3. HDD 250 Gb
4. DVD RW 24 X
5. LCD 15”
6. Keyboard dan Mouse PS/2
7. Printer

#### **5.5.2. Kebutuhan Perangkat Lunak**

Perangkat lunak merupakan kebalikan dari perangkat keras dimana fisiknya adalah mempunyai bentuk fisik yang tidak dapat dipegang. Perangkat lunak yang dibutuhkan yaitu :

1. Sistem operasi menggunakan Microsoft Windows 7.
2. Database untuk pengolahan data menggunakan MySQL Server 5.0 dan pembuatan program menggunakan Delphi.

## BAB VI

### KESIMPULAN DAN SARAN

Setelah melakukan pembahasan pada bab sebelumnya, maka pada bab ini penulis akan mengemukakan hasil akhir yang berisi kesimpulan dan saran yang dapat ditarik dari uraian-uraian pada bab-bab sebelumnya mengenai aplikasi sistem pendukung pengambilan keputusan untuk pemilihan Bank dengan Logika *Fuzzy*.

#### 6.1. Kesimpulan

1. Penelitian ini telah menghasilkan aplikasi sistem pendukung pengambilan keputusan untuk pemilihan Bank dengan Logika *Fuzzy* yang dapat digunakan untuk memberikan keputusan atau merekomendasikan pemilihan Bank berdasarkan *firestrength* dari kriteria yang dipilih.
2. Aplikasi sistem pendukung pengambilan keputusan untuk pemilihan Bank dengan Logika *Fuzzy* dibuat dengan menggunakan perangkat lunak (*software*) Borland Delphi dengan menggunakan database MySQL Server 5.
3. Hasil perhitungan Logika *Fuzzy* yang diterapkan ini akan menghasilkan keluaran nilai intensitas antara 0-1 sehingga Bank yang nilainya 0 merupakan Bank yang tidak direkomendasikan untuk dipilih dan nilai 1 merupakan Bank yang direkomendasikan untuk dipilih.

4. Hasil akhir dari penelitian ini adalah membuat suatu sistem yang dapat membantu para pembuat keputusan untuk menentukan solusi pemilihan Bank yang optimal berupa grafik rekomendasi dalam pemilihan Bank dengan menggunakan Logika *Fuzzy* Tahani.

## **6.2. Saran**

Saran yang dapat diberikan terhadap penelitian ini sebagai berikut :

1. Sistem ini dapat ditambahkan lagi kriteria yang lebih banyak agar pemilihan Bank lebih kompleks.
2. Seiring perkembangan teknologi informasi, maka akan lebih baik jika sistem yang ada sekarang untuk kedepannya dapat dikembangkan lagi dengan memanfaatkan teknologi internet sehingga masyarakat dapat menggunakan sendiri secara langsung.

## DAFTAR PUSTAKA

- Gunadi, Suhendar Hariman, 2002, **Visual Modelling Menggunakan UML dan Rational Rose**, Penerbit Informatika, Bandung
- Hasan, 2002, **Pokok-Pokok Materi Teori Pengambilan Keputusan**, Ghalia Indonesia, Jakarta
- Jogiyanto.HM, 2002, **Analisis dan Desain Sistem Informasi**, Andi Offset, Yogyakarta
- Kasmir, 2002, **Bank & Lembaga Keuangan Lainnya, Edisi keenam**, PT. RajaGarfindoPersada, Jakarta
- Kusumadewi, Sri. 2010. **Aplikasi Logika Fuzzy untuk Pendukung Keputusan**, Graha Ilmu, Yogyakarta.
- Pranata, Antony, 2002, **Dasar Pemrograman Delphi 6.0**, Andi Offset, Yogyakarta.
- Prasetyo, Didik Dwi, 2003, **Administrasi Database Server MySQL**, Elex Media Komputindo, Jakarta.
- Turban, Efraim, 2005, **Sistem Pendukung Keputusan dan Sistem Cerdas**, Andi, Yogyakarta.

# LAMPIRAN I

## LISTING PROGRAM

```
unit Utama;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics,
  Controls, Forms,
  Dialogs, sSkinManager, Menus, AdvMenus, DB, ADODB, StdCtrls,
  sEdit,
  Buttons, sBitBtn, ExtCtrls, sPanel, jpeg, sSkinProvider, TeeProcs,
  TeEngine, Chart;

type
  Tfmutama = class(TForm)
    sPanel1: TsPanel;
    ADOConnection1: TADOConnection;
    sSkinManager1: TsSkinManager;
    sSkinProvider1: TsSkinProvider;
    MainMenu1: TMainMenu;
    File1: TMenuItem;
    Login1: TMenuItem;
    MenuItem1: TMenuItem;
    Logout1: TMenuItem;
    MenuItem2: TMenuItem;
    Keluar1: TMenuItem;
    Master1: TMenuItem;
    DaftarUniversitas1: TMenuItem;
    MenuItem3: TMenuItem;
    Jurusan1: TMenuItem;
    N9: TMenuItem;
    Universitas1: TMenuItem;
    SPK1: TMenuItem;
    SPK2: TMenuItem;
    sPanel2: TsPanel;
    Label2: TLabel;
    Label1: TLabel;
    edtnama: TsEdit;
    edtpass: TsEdit;
    btnbatal: TsBitBtn;
    btnlogin: TsBitBtn;
    btnkeluar: TsBitBtn;
    Chart1: TChart;
    sPanel3: TsPanel;
```

```

Label3: TLabel;
Label4: TLabel;
sEdit1: TsEdit;
sEdit2: TsEdit;
sBitBtn1: TsBitBtn;
sBitBtn2: TsBitBtn;
sBitBtn3: TsBitBtn;
procedure FormActivate(Sender: TObject);
procedure btnkeluarClick(Sender: TObject);
procedure btnloginClick(Sender: TObject);
procedure Login1Click(Sender: TObject);
procedure Logout1Click(Sender: TObject);
procedure FormShow(Sender: TObject);
procedure Keluar1Click(Sender: TObject);
procedure DaftarUniversitas1Click(Sender: TObject);
procedure Jurusan1Click(Sender: TObject);
procedure Universitas1Click(Sender: TObject);
procedure SPK2Click(Sender: TObject);
procedure sBitBtn2Click(Sender: TObject);
procedure sBitBtn1Click(Sender: TObject);
procedure sBitBtn3Click(Sender: TObject);
procedure btnbatalClick(Sender: TObject);
private
  { Private declarations }
public
  { Public declarations }
end;

var
  fmutama: Tfmutama;

implementation

uses Kriteria, Bank, Nilai, Uspk;

{$R *.dfm}

procedure Tfmutama.FormActivate(Sender: TObject);
begin
  sPanel2.Visible:=false;
  sPanel3.Visible:=false;
end;

procedure Tfmutama.btnkeluarClick(Sender: TObject);
begin
  FormActivate(sender);
end;

procedure Tfmutama.btnloginClick(Sender: TObject);
begin
  if ((edtnama.Text='ADMIN') and (edtpass.Text='ADMIN')) then
    begin

```



```

        sPanel2.Visible:=false;
        Login1.Enabled:=false;
        Logout1.Enabled:=true;
        MainMenu1.Items[1].Enabled:=true;
    end
else
    begin
        ShowMessage('User Name ,, Password Salah!!!');
        btnbatalClick(sender);
    end;

end;

procedure Tfmutama.Login1Click(Sender: TObject);
begin
    sPanel2.Visible:=true;
    edtnama.SetFocus;
end;

procedure Tfmutama.Logout1Click(Sender: TObject);
begin
    MainMenu1.Items[1].Enabled:=false;
    Logout1.Enabled:=false;
    Login1.Enabled:=true;
end;

procedure Tfmutama.FormShow(Sender: TObject);
begin
    MainMenu1.Items[1].Enabled:=false;
    Logout1.Enabled:=false;
    Login1.Enabled:=true;
end;

procedure Tfmutama.Keluar1Click(Sender: TObject);
begin
    close;
end;

procedure Tfmutama.DaftarUniversitas1Click(Sender: TObject);
begin
    fmkriteria.ShowModal
end;

procedure Tfmutama.Jurusan1Click(Sender: TObject);
begin
    fmbank.ShowModal;
end;

procedure Tfmutama.Universitas1Click(Sender: TObject);
begin
    fmnilai.ShowModal;
end;

```

```

procedure Tfmutama.SPK2Click(Sender: TObject);
begin
sEdit1.Text:='';
sEdit2.Text:='';
sPanel3.Visible:=true;
end;

procedure Tfmutama.sBitBtn2Click(Sender: TObject);
begin
if ((sEdit1.Text<>'') and (sEdit2.Text<>'')) then
begin
sPanel3.Visible:=false;
fmspk.ShowModal;
end
else
begin
ShowMessage('Data Tidak Boleh Kosong');
sEdit1.SetFocus;
end;
end;

procedure Tfmutama.sBitBtn1Click(Sender: TObject);
begin
sEdit1.Text:='';
sEdit2.Text:='';
sEdit1.SetFocus;

end;

procedure Tfmutama.sBitBtn3Click(Sender: TObject);
begin
FormActivate(sender);

end;

procedure Tfmutama.btnbatalClick(Sender: TObject);
begin
edtnama.Text:='';
edtpass.Text:='';
edtnama.SetFocus;

end;

end.

```

```

unit Bank;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics,
  Controls, Forms,
  Dialogs, DB, StdCtrls, sComboBox, ADODB, sSkinManager,
  sSkinProvider,
  Buttons, sBitBtn, sEdit, Grids, BaseGrid, AdvGrid, DBAdvGrid;

type
  Tfmbank = class(TForm)
    Label1: TLabel;
    Label2: TLabel;
    Label7: TLabel;
    grdbank: TDBAdvGrid;
    edtkode: TsEdit;
    edtnama: TsEdit;
    btntambah: TsBitBtn;
    btnubah: TsBitBtn;
    btnhapus: TsBitBtn;
    btnkeluar: TsBitBtn;
    btnsimpan: TsBitBtn;
    btnbatal: TsBitBtn;
    edtatm: TsEdit;
    dsbank: TDataSource;
    tbbank: TADOTable;
    qycari: TADOQuery;
    Label3: TLabel;
    edtbunga: TsEdit;
    Label6: TLabel;
    edtadministrasi: TsEdit;
    edtpelayanan: TsEdit;
    Label9: TLabel;
    qysimpan: TADOQuery;
    tbbankidbank: TStringField;
    tbbanknmbank: TStringField;
    tbbankbunga: TFloatField;
    tbbankjmlatm: TIntegerField;
    tbbankadministrasi: TFloatField;
    tbbankpelayanan: TIntegerField;
    procedure FormActivate(Sender: TObject);
    procedure btntambahClick(Sender: TObject);
    procedure btnsimpanClick(Sender: TObject);
    procedure btnubahClick(Sender: TObject);
    procedure grdbankClickCell(Sender: TObject; ARow, ACol:
Integer);
    procedure grdbankKeyDown(Sender: TObject; var Key: Word;
      Shift: TShiftState);
    procedure btnhapusClick(Sender: TObject);
    procedure btnbatalClick(Sender: TObject);
  end;

```

```

    procedure FormCreate(Sender: TObject);
    procedure edtkodeChange(Sender: TObject);
    procedure btnkeluarClick(Sender: TObject);
    procedure edtnamaKeyPress(Sender: TObject; var Key: Char);
    procedure edtbungaKeyPress(Sender: TObject; var Key: Char);
    procedure edtatmKeyPress(Sender: TObject; var Key: Char);
    procedure edtadministrasiKeyPress(Sender: TObject; var Key:
Char);
    private
        { Private declarations }
    public
        { Public declarations }
    end;

var
    fmbank: Tfmbank;
    ket:string;
implementation

{$R *.dfm}

procedure Tfmbank.FormActivate(Sender: TObject);
begin
    edtkode.Enabled := False;
    edtnama.Enabled := False;
    edtbunga.Enabled := False;
    edtatm.Enabled := False;
    edtadministrasi.Enabled := False;
    edtpelayanan.Enabled := False;

    formcreate(sender);
    btntambah.Enabled:=true;
    btnkeluar.Enabled:=true;
    btnsimpan.Enabled:=false;
    btnbatal.Enabled:=false;
    ket := '';
    tbbank.Active:=false;
    tbbank.Active:=true;
    grdbank.Refresh;
    if tbbank.IsEmpty=true then
        begin
            btnhapus.Enabled:=false;
            btnubah.Enabled := false;
            grdbank.Enabled:=false;
        end
    else
        begin
            grdbank.Enabled:=true;
            btnhapus.Enabled:=true;
            btnubah.Enabled := true;
        end;
end;
end;

```

```

procedure TfmBank.btntambahClick(Sender: TObject);
begin
    ket := 'ISI';
    formcreate(sender);
    btnsimpan.Enabled:=true;
    btnbatal.Enabled:=true;
    btntambah.Enabled:=false;
    btnhapus.Enabled:=false;
    btnkeluar.Enabled:=false;
    btnubah.Enabled:=false;
    grdbank.Enabled:=false;
    edtkode.Enabled := true;
    edtnama.Enabled := False;
    edtbunga.Enabled := False;
    edtatm.Enabled := False;
    edtadministrasi.Enabled := False;
    edtpelayanan.Enabled := False;
    edtkode.SetFocus;
end;

procedure TfmBank.btnsimpanClick(Sender: TObject);
var
    nilai:real;
begin
    qycari.SQL.Clear;
    qycari.SQL.Add('DELETE FROM nilai WHERE idbank=:a');
    qycari.Parameters[0].Value:=edtkode.Text;
    qycari.ExecSQL;

    //hitung kriteria BUNGA
    qycari.SQL.Clear;
    qycari.SQL.Add('SELECT * FROM kriteria WHERE nmkriteria="BUNGA"');
    qycari.Open;
    qycari.First;

    repeat
        if Trim(qycari['fungsi'])='BAHU KIRI' then
            begin
                if StrToFloat(edtbunga.Text)<=qycari['nilfungsi1'] then
                    nilai:=1
                else if ((StrToFloat(edtbunga.Text)>=qycari['nilfungsi1']) AND
                    (StrToFloat(edtbunga.Text)<=qycari['nilfungsi2'])) then
                    nilai:=(qycari['nilfungsi2']-
                    StrToFloat(edtbunga.Text))/(qycari['nilfungsi2']-
                    qycari['nilfungsi1'])
                else
                    nilai:=0;
            end
        else if Trim(qycari['fungsi'])='SEGITIGA' then
            begin

```

```

        if ((StrToFloat(edtbunga.Text)<=qycari['nilfungsi1']) OR
(StrToFloat(edtbunga.Text)>=qycari['nilfungsi3'])) then
            nilai:=0
        else if ((StrToFloat(edtbunga.Text)>=qycari['nilfungsi1']) AND
(StrToFloat(edtbunga.Text)<=qycari['nilfungsi2'])) then
            nilai:=(StrToFloat(edtbunga.Text)-
qycari['nilfungsi1'])/(qycari['nilfungsi2']-qycari['nilfungsi1'])
        else if ((StrToFloat(edtbunga.Text)>=qycari['nilfungsi2']) AND
(StrToFloat(edtbunga.Text)<=qycari['nilfungsi3'])) then
            nilai:=(qycari['nilfungsi3']-
StrToFloat(edtbunga.Text))/(qycari['nilfungsi3']-
qycari['nilfungsi2']);
        end
        else if Trim(qycari['fungsi'])='BAHU KANAN' then
            begin
                if StrToFloat(edtbunga.Text)<=qycari['nilfungsi1'] then
                    nilai:=0
                else if ((StrToFloat(edtbunga.Text)>=qycari['nilfungsi1']) AND
(StrToFloat(edtbunga.Text)<=qycari['nilfungsi2'])) then
                    nilai:=(StrToFloat(edtbunga.Text)-
qycari['nilfungsi1'])/(qycari['nilfungsi2']-qycari['nilfungsi1'])
                else
                    nilai:=1;
            end;
            qysimpan.SQL.Clear;
            qysimpan.SQL.Add('INSERT INTO nilai VALUES (:a,:b,:c)');
            qysimpan.Parameters[0].Value:=qycari['kdkriteria'];
            qysimpan.Parameters[1].Value:=edtkode.Text;
            qysimpan.Parameters[2].Value:=nilai;
            qysimpan.ExecSQL;

            qycari.Next;
        until qycari.Eof;

//hitung kriteria ATM
qycari.SQL.Clear;
qycari.SQL.Add('SELECT * FROM kriteria WHERE nmkriteria="ATM"');
qycari.Open;
qycari.First;

repeat
    if Trim(qycari['fungsi'])='BAHU KIRI' then
        begin
            if StrToFloat(edtatm.Text)<=qycari['nilfungsi1'] then
                nilai:=1
            else if ((StrToFloat(edtatm.Text)>=qycari['nilfungsi1']) AND
(StrToFloat(edtatm.Text)<=qycari['nilfungsi2'])) then
                nilai:=(qycari['nilfungsi2']-
StrToFloat(edtatm.Text))/(qycari['nilfungsi2']-qycari['nilfungsi1'])
            else
                nilai:=0;
        end
    end
until qycari.Eof;

```

```

end
else if Trim(qycari['fungsi'])='SEGITIGA' then
begin
    if ((StrToFloat(edtatm.Text)<=qycari['nilfungsi1']) OR
(StrToFloat(edtatm.Text)>=qycari['nilfungsi3'])) then
        nilai:=0
    else if ((StrToFloat(edtatm.Text)>=qycari['nilfungsi1']) AND
(StrToFloat(edtatm.Text)<=qycari['nilfungsi2'])) then
        nilai:=(StrToFloat(edtatm.Text)-
qycari['nilfungsi1'])/(qycari['nilfungsi2']-qycari['nilfungsi1'])
    else if ((StrToFloat(edtatm.Text)>=qycari['nilfungsi2']) AND
(StrToFloat(edtatm.Text)<=qycari['nilfungsi3'])) then
        nilai:=(qycari['nilfungsi3']-
StrToFloat(edtatm.Text))/(qycari['nilfungsi3']-
qycari['nilfungsi2']);
    end
else if Trim(qycari['fungsi'])='BAHU KANAN' then
begin
    if StrToFloat(edtatm.Text)<=qycari['nilfungsi1'] then
        nilai:=0
    else if ((StrToFloat(edtatm.Text)>=qycari['nilfungsi1']) AND
(StrToFloat(edtatm.Text)<=qycari['nilfungsi2'])) then
        nilai:=(StrToFloat(edtatm.Text)-
qycari['nilfungsi1'])/(qycari['nilfungsi2']-qycari['nilfungsi1'])
    else
        nilai:=1;
    end;
qysimpan.SQL.Clear;
qysimpan.SQL.Add('INSERT INTO nilai VALUES (:a,:b,:c)');
qysimpan.Parameters[0].Value:=qycari['kdkriteria'];
qysimpan.Parameters[1].Value:=edtkode.Text;
qysimpan.Parameters[2].Value:=nilai;
qysimpan.ExecSQL;

qycari.Next;
until qycari.Eof;

//hitung kriteria ADMINISTRASI
qycari.SQL.Clear;
qycari.SQL.Add('SELECT * FROM kriteria WHERE
nmkriteria="ADMINISTRASI"');
qycari.Open;
qycari.First;

repeat
    if Trim(qycari['fungsi'])='BAHU KIRI' then
        begin
            if StrToFloat(edtadministrasi.Text)<=qycari['nilfungsi1'] then
                nilai:=1
            else if ((StrToFloat(edtadministrasi.Text)>=qycari['nilfungsi1'])
AND (StrToFloat(edtadministrasi.Text)<=qycari['nilfungsi2'])) then

```

```

        nilai:=(qycari['nilfungsi2']-
StrToFloat(edtadministrasi.Text))/(qycari['nilfungsi2']-
qycari['nilfungsi1'])
        else
            nilai:=0;
        end
    else if Trim(qycari['fungsi'])='SEGITIGA' then
        begin
            if ((StrToFloat(edtadministrasi.Text)<=qycari['nilfungsi1']) OR
(StrToFloat(edtadministrasi.Text)>=qycari['nilfungsi3'])) then
                nilai:=0
            else if ((StrToFloat(edtadministrasi.Text)>=qycari['nilfungsi1'])
AND (StrToFloat(edtadministrasi.Text)<=qycari['nilfungsi2'])) then
                nilai:=(StrToFloat(edtadministrasi.Text)-
qycari['nilfungsi1'])/(qycari['nilfungsi2']-qycari['nilfungsi1'])
            else if ((StrToFloat(edtadministrasi.Text)>=qycari['nilfungsi2'])
AND (StrToFloat(edtadministrasi.Text)<=qycari['nilfungsi3'])) then
                nilai:=(qycari['nilfungsi3']-
StrToFloat(edtadministrasi.Text))/(qycari['nilfungsi3']-
qycari['nilfungsi2']);
            end
        else if Trim(qycari['fungsi'])='BAHU KANAN' then
            begin
                if StrToFloat(edtadministrasi.Text)<=qycari['nilfungsi1'] then
                    nilai:=0
                else if ((StrToFloat(edtadministrasi.Text)>=qycari['nilfungsi1'])
AND (StrToFloat(edtadministrasi.Text)<=qycari['nilfungsi2'])) then
                    nilai:=(StrToFloat(edtadministrasi.Text)-
qycari['nilfungsi1'])/(qycari['nilfungsi2']-qycari['nilfungsi1'])
                else
                    nilai:=1;
            end;
            qysimpan.SQL.Clear;
            qysimpan.SQL.Add('INSERT INTO nilai VALUES (:a,:b,:c)');
            qysimpan.Parameters[0].Value:=qycari['kdkriteria'];
            qysimpan.Parameters[1].Value:=edtkode.Text;
            qysimpan.Parameters[2].Value:=nilai;
            qysimpan.ExecSQL;

            qycari.Next;
        until qycari.Eof;

//hitung kriteria PELAYANAN
qycari.SQL.Clear;
qycari.SQL.Add('SELECT * FROM kriteria WHERE
nmkriteria="PELAYANAN"');
qycari.Open;
qycari.First;

repeat
    if Trim(qycari['fungsi'])='BAHU KIRI' then
        begin

```



```

        if StrToFloat(edtpelayanan.Text)<=qycari['nilfungsi1'] then
            nilai:=1
        else if ((StrToFloat(edtpelayanan.Text)>=qycari['nilfungsi1'])
AND (StrToFloat(edtpelayanan.Text)<=qycari['nilfungsi2'])) then
            nilai:=(qycari['nilfungsi2']-
StrToFloat(edtpelayanan.Text))/(qycari['nilfungsi2']-
qycari['nilfungsi1'])
        else
            nilai:=0;
        end
        else if Trim(qycari['fungsi'])='SEGITIGA' then
            begin
                if ((StrToFloat(edtpelayanan.Text)<=qycari['nilfungsi1']) OR
(StrToFloat(edtpelayanan.Text)>=qycari['nilfungsi3'])) then
                    nilai:=0
                else if ((StrToFloat(edtpelayanan.Text)>=qycari['nilfungsi1'])
AND (StrToFloat(edtpelayanan.Text)<=qycari['nilfungsi2'])) then
                    nilai:=(StrToFloat(edtpelayanan.Text)-
qycari['nilfungsi1'])/(qycari['nilfungsi2']-qycari['nilfungsi1'])
                else if ((StrToFloat(edtpelayanan.Text)>=qycari['nilfungsi2'])
AND (StrToFloat(edtpelayanan.Text)<=qycari['nilfungsi3'])) then
                    nilai:=(qycari['nilfungsi3']-
StrToFloat(edtpelayanan.Text))/(qycari['nilfungsi3']-
qycari['nilfungsi2']);
                end
            else if Trim(qycari['fungsi'])='BAHU KANAN' then
                begin
                    if StrToFloat(edtpelayanan.Text)<=qycari['nilfungsi1'] then
                        nilai:=0
                    else if ((StrToFloat(edtpelayanan.Text)>=qycari['nilfungsi1'])
AND (StrToFloat(edtpelayanan.Text)<=qycari['nilfungsi2'])) then
                        nilai:=(StrToFloat(edtpelayanan.Text)-
qycari['nilfungsi1'])/(qycari['nilfungsi2']-qycari['nilfungsi1'])
                    else
                        nilai:=1;
                end;
            qysimpan.SQL.Clear;
            qysimpan.SQL.Add('INSERT INTO nilai VALUES (:a,:b,:c)');
            qysimpan.Parameters[0].Value:=qycari['kdkriteria'];
            qysimpan.Parameters[1].Value:=edtkode.Text;
            qysimpan.Parameters[2].Value:=nilai;
            qysimpan.ExecSQL;

            qycari.Next;
        until qycari.Eof;

```

```

if ket='ISI' then
    begin
        qycari.SQL.Clear;
        qycari.SQL.Add('INSERT INTO bank values (:a,:b,:c,:d,:e,:f)');
        qycari.Parameters[0].Value:=edtkode.Text;

```

```

qycari.Parameters[1].Value:=edtnama.Text;
qycari.Parameters[2].Value:=StrToFloat(edtbunga.Text);
qycari.Parameters[3].Value:=StrToFloat(edtatm.Text);
qycari.Parameters[4].Value:=StrToFloat(edtadministrasi.Text);
qycari.Parameters[5].Value:=StrToFloat(edtpelayanan.Text);
qycari.ExecSQL;

btntambahClick(sender);
tbbank.Active:=false;
tbbank.Active:=true;
tbbank.Last;
end
else if ket='EDIT' then
begin
qycari.SQL.Clear;
qycari.SQL.Add('UPDATE bank SET
nmbank=:a,bunga=:d,jmlatm=:e,administrasi=:f,pelayanan=:g WHERE
idbank=:j');
qycari.Parameters[7].Value:=edtkode.Text;
qycari.Parameters[0].Value:=edtnama.Text;
qycari.Parameters[1].Value:=StrToFloat(edtbunga.Text);
qycari.Parameters[2].Value:=StrToFloat(edtatm.Text);
qycari.Parameters[3].Value:=StrToFloat(edtadministrasi.Text);
qycari.Parameters[4].Value:=StrToFloat(edtpelayanan.Text);
qycari.ExecSQL();

FormActivate(sender);
end;

end;

procedure Tfmbank.btnubahClick(Sender: TObject);
begin
if tbbank.IsEmpty=true then
btubah.Enabled := false
else
begin
ket:='EDIT';
btnsimpan.Enabled:=true;
btnbatal.Enabled:=true;
btntambah.Enabled:=false;
btnhapus.Enabled:=false;
btnkeluar.Enabled:=false;
btubah.Enabled:=false;
edtkode.Enabled:=false;
edtnama.Enabled := true;
edtbunga.Enabled := true;
edtatm.Enabled := true;
edtadministrasi.Enabled := true;
edtpelayanan.Enabled := true;
edtnama.SetFocus;
end;
end;

```

```

end;

procedure Tfmbank.grdbankClickCell(Sender: TObject; ARow,
  ACol: Integer);
begin
edtkode.Text:= tbbankidbank.Value;
edtnama.Text:= tbbanknmbank.Value;
edtbunga.Text:= FloatToStr(tbbankbunga.Value);
edtadm.Text:= FloatToStr(tbbankjmlatm.Value);
edtadministrasi.Text:= FloatToStr(tbbankadministrasi.Value);
edtpelayanan.Text:= FloatToStr(tbbankpelayanan.Value);
end;

procedure Tfmbank.grdbankKeyDown(Sender: TObject; var Key: Word;
  Shift: TShiftState);
begin
edtkode.Text:= tbbankidbank.Value;
edtnama.Text:= tbbanknmbank.Value;
edtbunga.Text:= FloatToStr(tbbankbunga.Value);
edtadm.Text:= FloatToStr(tbbankjmlatm.Value);
edtadministrasi.Text:= FloatToStr(tbbankadministrasi.Value);
edtpelayanan.Text:= FloatToStr(tbbankpelayanan.Value);
end;

procedure Tfmbank.btnhapusClick(Sender: TObject);
begin
if tbbank.IsEmpty=true then
  btnhapus.Enabled:=false
else
  begin
  if MessageDlg('Data Ingin Dihapus', mtwarning, [mbYes, mbNo], 0)
= mrYes then
    begin
    btnhapus.Enabled:=true;
    qycari.SQL.Clear;
    qycari.SQL.Add('DELETE FROM nilai WHERE idbank=:a');
    qycari.Parameters[0].Value:=edtkode.Text;
    qycari.ExecSQL;

    qycari.SQL.Clear;
    qycari.SQL.Add('DELETE FROM bank WHERE idbank=:a');
    qycari.Parameters[0].Value:=edtkode.Text;
    qycari.ExecSQL;

    FormActivate(sender);
    end;
  end
end;

procedure Tfmbank.btnbatalClick(Sender: TObject);
begin
FormActivate(sender);
end;

```

```

end;

procedure Tfmbank.FormCreate(Sender: TObject);
begin
edtkode.Text:= '';
edtnama.Text:= '';
edtbunga.Text:= '0';
edtatm.Text:= '0';
edtadministrasi.Text:= '0';
edtpelayanan.Text:= '0';
end;

procedure Tfmbank.edtkodeChange(Sender: TObject);
begin
if Length(edtkode.Text)<3 then
  exit
else
  begin
    if ket='ISI' then
      begin
        qycari.SQL.Clear;
        qycari.SQL.Add('SELECT * FROM bank WHERE idbank=:a');
        qycari.Parameters[0].Value:=edtkode.Text;
        qycari.Open;
        if qycari.RecordCount>0 then
          begin
            ShowMessage('ID Bank '+edtkode.Text+' Sudah Ada');
            btntambahClick(sender);
          end
        else
          begin
            edtkode.Enabled:=false;
            edtnama.Enabled := true;
            edtbunga.Enabled := true;
            edtatm.Enabled := true;
            edtadministrasi.Enabled := true;
            edtpelayanan.Enabled := true;
            edtnama.SetFocus;
          end;
        end;
      end;
    end;
  end;
end;

procedure Tfmbank.btnkeluarClick(Sender: TObject);
begin
Close;
end;

procedure Tfmbank.edtnamaKeyPress(Sender: TObject; var Key: Char);
begin
if key=#13 then
  edtbunga.SetFocus;

```

end;

```
procedure Tfmbank.edtbungaKeyPress(Sender: TObject; var Key: Char);  
begin  
  if key=#13 then  
    edtatm.SetFocus;  
  end;
```

```
procedure Tfmbank.edtatmKeyPress(Sender: TObject; var Key: Char);  
begin  
  if key=#13 then  
    edtadministrasi.SetFocus;  
  end;
```

```
procedure Tfmbank.edtadministrasiKeyPress(Sender: TObject; var Key:  
Char);  
begin  
  if key=#13 then  
    edtpelayanan.SetFocus;  
  end;
```

end.

```

unit Nilai;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics,
  Controls, Forms,
  Dialogs, StdCtrls, sEdit, DB, sComboBox, ADODB, sSkinManager,
  sSkinProvider, Buttons, sBitBtn, Grids, BaseGrid, AdvGrid,
  DBAdvGrid,
  ExtCtrls, sPanel;

type
  Tfmnilai = class(TForm)
    grdfuzzy: TDBAdvGrid;
    btnfirst: TsBitBtn;
    btnprev: TsBitBtn;
    btnnext: TsBitBtn;
    btnkeluar: TsBitBtn;
    btnlast: TsBitBtn;
    dsnilai: TDataSource;
    qynilai: TADOQuery;
    qynilaikdkriteria: TStringField;
    qynilaiidbank: TStringField;
    qynilainilfuzzy: TFloatField;
    qynilainmkriteria: TStringField;
    qynilainmbank: TStringField;
    procedure btnkeluarClick(Sender: TObject);
    procedure btnfirstClick(Sender: TObject);
    procedure btnprevClick(Sender: TObject);
    procedure btnnextClick(Sender: TObject);
    procedure btnlastClick(Sender: TObject);
    procedure FormActivate(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  fmnilai: Tfmnilai;

implementation

uses Utama;

{$R *.dfm}

procedure Tfmnilai.btnkeluarClick(Sender: TObject);
begin
  Close;
end;

```

```
procedure Tfmnilai.btnfirstClick(Sender: TObject);
begin
  qynilai.First;
end;

procedure Tfmnilai.btnprevClick(Sender: TObject);
begin
  qynilai.Prior;
end;

procedure Tfmnilai.btnnextClick(Sender: TObject);
begin
  qynilai.Next;
end;

procedure Tfmnilai.btnlastClick(Sender: TObject);
begin
  qynilai.Last;
end;

procedure Tfmnilai.FormActivate(Sender: TObject);
begin
  qynilai.Close;
  qynilai.Open;
end;

end.
```

```

unit Uspk;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics,
  Controls, Forms,
  Dialogs, DB, ADODB, sSkinProvider, sSkinManager, StdCtrls,
  Buttons,
  sBitBtn, sComboBox;

type
  Tfmspk = class(TForm)
    Label7: TLabel;
    label13: TLabel;
    Label3: TLabel;
    cbbunga: TsComboBox;
    cbatm: TsComboBox;
    cbpelayanan: TsComboBox;
    btncetak: TsBitBtn;
    btnubah: TsBitBtn;
    DataSource1: TDataSource;
    qysimpan: TADOQuery;
    qyspk: TADOQuery;
    Label2: TLabel;
    cbadministrasi: TsComboBox;
    qycari: TADOQuery;
    procedure FormActivate(Sender: TObject);
    procedure cbbungaClick(Sender: TObject);
    procedure cbatmClick(Sender: TObject);
    procedure cbadministrasiClick(Sender: TObject);
    procedure cbpelayananClick(Sender: TObject);
    procedure btncetakClick(Sender: TObject);
    procedure btnubahClick(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  fmspk: Tfmspk;
  bunga, bungal, atm, atm1, adm, adm1, pel, pel1: string;
  sql1: string;
implementation

uses Utama, Ulapspk;

{$R *.dfm}

procedure Tfmspk.FormActivate(Sender: TObject);
begin

```



```

cbbunga.Text:='';
cbatm.Text:='';
cbadministrasi.Text:='';
cbpelayanan.Text:='';

cbbunga.Items.Clear;
cbatm.Items.Clear;
cbadministrasi.Items.Clear;
cbpelayanan.Items.Clear;

qysimpan.SQL.Clear;
qysimpan.SQL.Add('SELECT * FROM kriteria WHERE nmkriteria="BUNGA"
ORDER BY kdkriteria');
qysimpan.Open;
qysimpan.First;
cbbunga.Items.Add('ABAIKAN');
repeat
  if qysimpan.RecNo=1 then
    cbbunga.Items.Add('Bunga <=
'+FormatFloat('##,##0',qysimpan['nilfungsi1']))
  else if qysimpan.RecNo=qysimpan.RecordCount then
    cbbunga.Items.Add('Bunga >
'+FormatFloat('##,##0',qysimpan['nilfungsi2']))
  else
    cbbunga.Items.Add(FormatFloat('##,##0',qysimpan['nilfungsi1'])+
'<= Bunga <= '+FormatFloat('##,##0',qysimpan['nilfungsi3']));
qysimpan.Next;
until qysimpan.Eof;

qysimpan.SQL.Clear;
qysimpan.SQL.Add('SELECT * FROM kriteria WHERE nmkriteria="ATM"
ORDER BY kdkriteria');
qysimpan.Open;
qysimpan.First;
cbatm.Items.Add('ABAIKAN');
repeat
  if qysimpan.RecNo=1 then
    cbatm.Items.Add('ATM <=
'+FormatFloat('##,##0',qysimpan['nilfungsi1']))
  else if qysimpan.RecNo=qysimpan.RecordCount then
    cbatm.Items.Add('ATM >
'+FormatFloat('##,##0',qysimpan['nilfungsi2']))
  else
    cbatm.Items.Add(FormatFloat('##,##0',qysimpan['nilfungsi1'])+ '<=
ATM <= '+FormatFloat('##,##0',qysimpan['nilfungsi3']));
qysimpan.Next;
until qysimpan.Eof;

qysimpan.SQL.Clear;
qysimpan.SQL.Add('SELECT * FROM kriteria WHERE
nmkriteria="ADMINISTRASI" ORDER BY kdkriteria');
qysimpan.Open;

```

```

qysimpan.First;
cbadministrasi.Items.Add('ABAIKAN');
repeat
  if qysimpan.RecNo=1 then
    cbadministrasi.Items.Add('Administrasi <=
'+FormatFloat('##,##0',qysimpan['nilfungsil']))
  else if qysimpan.RecNo=qysimpan.RecordCount then
    cbadministrasi.Items.Add('Administrasi >
'+FormatFloat('##,##0',qysimpan['nilfungsi2']))
  else

cbadministrasi.Items.Add(FormatFloat('##,##0',qysimpan['nilfungsil']
)+ '<= Administrasi <=
'+FormatFloat('##,##0',qysimpan['nilfungsi3']));
qysimpan.Next;
until qysimpan.Eof;

qysimpan.SQL.Clear;
qysimpan.SQL.Add('SELECT * FROM kriteria WHERE
nmkriteria="PELAYANAN" ORDER BY kdkriteria');
qysimpan.Open;
qysimpan.First;
cbpelayanan.Items.Add('ABAIKAN');
repeat
  if qysimpan.RecNo=1 then
    cbpelayanan.Items.Add('Pelayanan <=
'+FormatFloat('##,##0',qysimpan['nilfungsil']))
  else if qysimpan.RecNo=qysimpan.RecordCount then
    cbpelayanan.Items.Add('Pelayanan >
'+FormatFloat('##,##0',qysimpan['nilfungsi2']))
  else

cbpelayanan.Items.Add(FormatFloat('##,##0',qysimpan['nilfungsil'])+
'<= Pelayanan <= '+FormatFloat('##,##0',qysimpan['nilfungsi3']));
qysimpan.Next;
until qysimpan.Eof;

end;

procedure Tfmspk.cbbungaClick(Sender: TObject);
begin
qysimpan.SQL.Clear;
qysimpan.SQL.Add('SELECT * FROM kriteria WHERE nmkriteria="BUNGA"
ORDER BY kdkriteria');
qysimpan.Open;
qysimpan.First;
if cbbunga.ItemIndex>0 then
  qysimpan.RecNo:=cbbunga.ItemIndex;

if cbbunga.ItemIndex>0 then
  begin
    bunga:=qysimpan['kdkriteria'];

```

```

    bungal:='kdkritria=' +QuotedStr(bunga) +' OR ';
end
else
begin
    bunga:='';
    bungal:='';
end;
end;

procedure Tfmspk.cbatmClick(Sender: TObject);
begin
qysimpan.SQL.Clear;
qysimpan.SQL.Add('SELECT * FROM kriteria WHERE nmkriteria="ATM"
ORDER BY kdkriteria');
qysimpan.Open;
qysimpan.First;
if cbatm.ItemIndex>0 then
    qysimpan.RecNo:=cbatm.ItemIndex;

if cbatm.ItemIndex>0 then
begin
    atm:=qysimpan['kdkriteria'];
    atm1:='kdkritria=' +QuotedStr(atm) +' OR ';
end
else
begin
    atm:='';
    atm1:='';
end;
end;

procedure Tfmspk.cbadministrasiClick(Sender: TObject);
begin
qysimpan.SQL.Clear;
qysimpan.SQL.Add('SELECT * FROM kriteria WHERE
nmkriteria="ADMINISTRASI" ORDER BY kdkriteria');
qysimpan.Open;
qysimpan.First;
if cbadministrasi.ItemIndex>0 then
    qysimpan.RecNo:=cbadministrasi.ItemIndex;

if cbadministrasi.ItemIndex>0 then
begin
    adm:=qysimpan['kdkriteria'];
    adm1:='kdkritria=' +QuotedStr(adm) +' OR ';
end
else
begin
    adm:='';
    adm1:='';
end;
end;
end;

```

```

procedure Tfmspk.cbelayananClick(Sender: TObject);
begin
qysimpan.SQL.Clear;
qysimpan.SQL.Add('SELECT * FROM kriteria WHERE
nmkriteria="PELAYANAN" ORDER BY kdkriteria');
qysimpan.Open;
qysimpan.First;
if cbelayanan.ItemIndex>0 then
  qysimpan.RecNo:=cbelayanan.ItemIndex;

if cbelayanan.ItemIndex>0 then
  begin
  pel:=qysimpan['kdkriteria'];
  pell:='kdkriteria=' +QuotedStr(pel) +' OR ';
  end
else
  begin
  pel:='';
  pell:='';
  end;
end;

procedure Tfmspk.btncetakClick(Sender: TObject);
begin
sql1:='';

cbbungaClick(sender);
cbatmClick(sender);
cbadministrasiClick(sender);
cbelayananClick(sender);

qysimpan.SQL.Clear;
qysimpan.sql.Add('DELETE FROM firestrenght');
qysimpan.ExecSQL;

qysimpan.SQL.Clear;
qysimpan.sql.Add('DELETE FROM nilai1');
qysimpan.ExecSQL;

sql1:=Trim(bungal+atml+adml+pell);
delete(sql1,length(sql1)-2,3);
//Memol.Text:=sql1;

qyspk.SQL.Clear;
qyspk.SQL.Add('SELECT a.idbank,a.nilfuzzy FROM nilai a INNER JOIN
bank b ON a.idbank=b.idbank WHERE '+sql1);
qyspk.Open;
qyspk.First;

repeat

```

```

qysimpan.SQL.Clear;
qysimpan.SQL.Add('INSERT INTO nilail VALUES (:a,:b)');
qysimpan.Parameters[0].Value:=qyspk['idbank'];
qysimpan.Parameters[1].Value:=qyspk['nilfuzzy'];
qysimpan.ExecSQL;
qyspk.Next;
until qyspk.Eof;

qyspk.SQL.Clear;
qyspk.SQL.Add('SELECT idbank FROM bank');
qyspk.Open;
qyspk.First;

repeat
  qysimpan.SQL.Clear;
  qysimpan.SQL.Add('SELECT min(nilai) AS nilai FROM nilail WHERE
idbank=:z');
  qysimpan.Parameters[0].Value:=qyspk['idbank'];
  qysimpan.Open;

  qycari.SQL.Clear;
  qycari.SQL.Add('INSERT INTO firestrength VALUES (:a,:b)');
  qycari.Parameters[0].Value:=qyspk['idbank'];
  qycari.Parameters[1].Value:=qysimpan['nilai'];
  qycari.ExecSQL;
  qyspk.Next;
until qyspk.Eof;

with fmlaporan do
  begin
    qycari.sql.Clear;
    qycari.SQL.Add('SELECT a.nmbank,b.nilai FROM bank a inner join
firestrength b on a.idbank=b.idbank where b.nilai>0');
    qycari.Open;
    if qycari.RecordCount=0 then
      begin
        ShowMessage('Tidak Ada Bank Yang Sesuai Dengan Kriteria Yang
Dipilih'+#13+'Silahkan Ulangi Lagi');
      end
    else
      begin
        QRLabel6.Caption:= 'Nama    = '+fmutama.sEdit1.Text;
        QRLabel7.Caption:= 'Alamat  = '+fmutama.sEdit2.Text;
        QRLabel2.Caption:= 'Bunga   = '+cbbunga.Text;
        QRLabel3.Caption:= 'ATM    = '+cbatm.Text;
        QRLabel4.Caption:= 'Administrasi = '+cbadministrasi.Text;
        QRLabel5.Caption:= 'Pelayanan = '+cbpelayanan.Text;
        QuickRep1.Preview;
      end;
    end;
  end;
end;

```

```
procedure Tfmspk.btnubahClick(Sender: TObject);  
begin  
  FormActivate(sender);  
end;  
  
end.
```