

**SISTEM PENDUKUNG KEPUTUSAN PEMBERIAN
PINJAMAN PADA KOPERASI WANITA AL-WARDAH
MENGUNAKAN METODE SIMPLE ADDITIVE WIGHTING**

SKRIPSI



**Disusun Oleh :
M.MAKHSHUSHUN BIL IZZI
12.18.260**

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI NASIONAL MALANG
2016**

LEMBAR PERSETUJUAN

**SISTEM PENDUKUNG KEPUTUSAN PEMBERIAN PINJAMAN PADA
KOPERASI WANITA AL-WARDAH MENGGUNAKAN METODE
SIMPLE ADDITIVE WEIGHTING**

SKRIPSI

*Disusun dan Diajukan untuk melengkapi dan memenuhi persyaratan guna
mencapai Gelar Sarjana Komputer Strata Satu (S-1)*

Disusun Oleh :

M.MAKHSHUSHUN BIL IZZI

NIM : 12.18.260

Diperiksa dan Disetujui

Dosen Pembimbing I

Dosen Pembimbing II

Karina Auliasari, ST, M.Eng
NIP.P.1031000426

Moh. Miftakhur Rokhman, S.Kom, M.Kom
NIP.P. 1031500479

Mengetahui

Ketua Program Studi Teknik Informatika S-1



Joseph Dedy Irawan, ST, MT
NIP. 197404162005011002

**PROGRAM STUDI TEKNIK INFORMATIKA S-1
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI NASIONAL MALANG
2016**

LEMBAR KEASLIAN
PERNYATAAN KEASLIAN SKRIPSI

Saya yang bertanda tangan dibawah ini :

Nama : M.Makhshushun Bil Izzi

NIM : 12.8.260

Program Studi : Teknik Informatika S-1

Fakultas : Fakultas Teknologi Industri

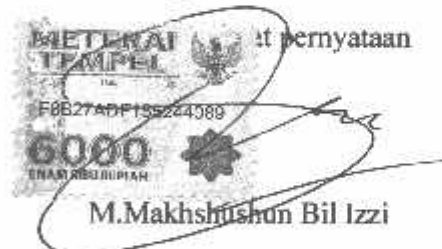
Menyatakan dengan sesungguhnya bahwa Skripsi saya yang berjudul :

“Sistem Pendukung Keputusan Pemberian Pinjaman Pada Koperasi Wanita AL-Wardah Menggunakan Metode Simple Additive Weighting (SAW)”

Adalah skripsi sendiri bukan duplikasi serta mengutip atau menyadur seluruhnya karya orang lain kecuali dari sumber aslinya.

Malang, 10 Januari 2015

Biaya pernyataan
METERAI
TEMPEL
F8B27ADP155224588
6000
ENAM RIBU RUPIAH
M.Makhshushun Bil Izzi



Abstrak

Disetiap organisasi bisnis tidak rentang dengan adanya suatu masalah dalam pengolahan data-data. Di antaranya adalah organisasi koperasi, Menurut ILO atau Organisasi buruh Internasional bahwa pengertian koperasi adalah badan usaha yang beranggotakan orang seorang atau badan hukum. Setiap koperasi yang ada harus melandaskan seluruh kegiatannya pada prinsip koperasi serta asas kekeluargaan untuk meningkatkan gerakan ekonomi rakyat. Koperasi juga merupakan koperasi yang menyelenggarakan pelayanan jasa yang di butuhkan oleh anggota, antara lain yaitu simpan pinjam. Koperasi wanita simpan pinjam AL-Wardah merupakan koperasi wanita yang ada di desa bangkok glagah lamongan adalah salah satu Koperasi yang dipercaya oleh masyarakat. Semakin tingginya minat masyarakat untuk mendapatkan pinjaman, membuat pihak koperasi kesulitan dalam menentukan siapa yang layak menerima pinjaman atau tidak. Selain itu, proses penentuan siapa yang layak menerima pinjaman masih dilakukan secara manual, sehingga kurang efisien dalam pelaksanaannya. Oleh karena itu, penulis berinisiatif untuk merancang suatu sistem yang dapat membantu pihak koperasi dalam menentukan siapa yang layak menerima pinjaman, sehingga dapat lebih efisien dalam pelaksanaannya.

Ada beberapa model yang dapat digunakan untuk membangun sebuah SPK. Berdasarkan hal-hal ini, maka metode Simple Additive Weighting (SAW) digunakan dalam penelitian ini yaitu untuk menentukan calon peminjam yang layak menerima pinjaman dari koperasi dengan mempertimbangkan kriteria-kriteria yang telah ditentukan oleh pihak koperasi. Adapun kriteria-kriteria yang menjadi dasar pengambilan keputusan oleh pihak koperasi dalam menentukan calon peminjam adalah status kredit, produktivitas usaha, kondisi usaha, jaminan, dan kolektibilitas. Walaupun pemilihan calon peminjam yang akan menerima pinjaman tetap ditentukan sepenuhnya oleh pihak koperasi, namun Sistem Pendukung Keputusan ini akan menampilkan nilai prioritas global dari yang tertinggi hingga terendah dari calon nasabah tersebut, sehingga akan memudahkan dan membantu pihak koperasi dalam mengambil keputusan.

Dengan adanya sistem pendukung keputusan ini, diharapkan bisa membantu dapat membantu mengurangi waktu pengerjaannya serta mempercepat proses seleksi pemberian pinjaman karena proses seleksi dilakukan secara otomatis dan mengurangi biaya yang di keluarkan koperasi untuk menentukan siapa yang layak mendapat pinjaman dan tidak.

Kata Kunci: Sistem pendukung keputusan, Metode SAW, MADM, Koperasi.

KATA PENGANTAR

Puji syukur kehadiran Allah yang maha kuasa, karena telah memberikan rahmat dan hidayah-Nya sehingga penyusun dapat menyelesaikan skripsi dengan judul SISTEM PENDUKUNG KEPUTUSAN PEMBERIAN PINJAMAN PADA KOPERASI WANITA AL-WARDAH MENGGUNAKAN METODE SIMPLE ADDITIVE WIGHTING sesuai dengan waktu yang ditentukan.

Penyusunan skripsi ini merupakan salah satu persyaratan untuk menyelesaikan program pendidikan Strata Satu (S-1) Teknik Informatika, Fakultas Teknologi Industri di Institut Teknologi Nasional Malang.

Pada penyusunan skripsi ini kami mengucapkan banyak terima kasih sebesar-besarnya kepada :

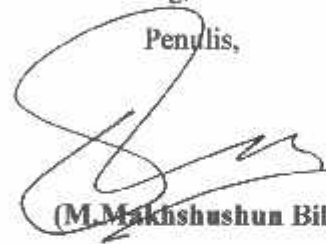
1. Bapak Mustaqol Ghomam dan Ibu Mutamakkinah, yang merupakan kedua orang tua dan pendukung utama dari segi moril maupun materil.
2. Bapak Dr. Ir. Lalu Mulyadi, MT, selaku Rektor Institut Teknologi Nasional Malang.
3. Bapak Ir. Anang subardi, MT, selaku Dekan Fakultas Teknologi Industri Institut Teknologi Nasional Malang.
4. Joseph Dedy Irawan, ST, MT, selaku Ketua Program Studi Teknik Informatika, Institut Teknologi Nasional Malang.
5. Bapak Sonny Prasetio, ST, MT, selaku Sekertaris Program Studi Teknik Informatika, Institut Teknologi Nasional Malang.
6. Ibu Karunia Auliasari, ST. M.Eng, selaku Dosen Pembimbing I, yang selalu memberikan masukan.
7. Bapak Moh. Miftakhur Rokhman, S.Kom.M.Kom, selaku Dosen Pembimbing II, yang selalu memberikan masukan.

8. Bapak dan Ibu dosen Program Studi Teknik Informatika yang telah membantu dalam penulisan dan masukan.

Penyusun menyadari bahwa skripsi masih jauh dari sempurna, oleh karena itu penyusun mengharapkan kritik dan saran dari pembaca. Semoga skripsi ini bisa bermanfaat bagi pembaca.

Malang, Januari 2016

Penulis,



(M. Makhshushun Bil Izzi)

DAFTAR ISI

ABSTRAK	iv
KATA PENGANTAR.....	v
DAFTAR ISI.....	vii
DAFTAR GAMBAR.....	ix
DAFTAR TABEL	x
BAB I PENDAHULUAN	
1.1. Latar Belakang	1
1.2. Rumusan Masalah	2
1.3. Batasan Masalah.....	2
1.4. Tujuan	2
1.5. Metode Penulisan	3
1.6. Sistematika Penulisan	4
BAB II LANDASAN TEORI	
2.1. Koperasi	5
2.2. Sistem Pendukung Keputusan (SPK)	5
2.2.1 Komponen Sistem Pendukung Keputusan	6
2.3. <i>Microsoft Visual Studio</i>	7
2.4. <i>Microsoft SQL Server</i>	7
2.5. <i>Multiple Attribute Decision Making (MADM)</i>	8
2.5.1 Metode-Metode Penyelesaian Masalah MADM (<i>Multi Attribute Decision Making</i>)	10
2.5.2 Metode <i>Simple Additive Weighting (SAW)</i>	20
2.6. Teori <i>Flowchart</i>	21
BAB III ANALISIS DAN PERANCANGAN SISTEM	
3.1. Identifikasi Masalah	22
3.2. Perancangan Proses	22
3.3. Hak Akses Pengguna Aplikasi	26
3.4. Pemodelan Keputusan	27
3.4.1 Proses Pengambilan Keputusan	27
3.4.2 Pohon Keputusan/Table Keputusan	28
3.4.3 Pengumpulan Sampel Data	29
3.5. Perancangan Sistem	32
3.5.1 <i>Flowchart SAW (Simple Additive Weighting)</i>	32
3.5.2 Perancangan Antar Muka (<i>Interface</i>)	33
3.5.2.1 Perancangan Tampilan	33

BAB IV IMPLEMENTASI DAN PENGUJIAN

4.1	Implementasi Sistem Pendukung Keputusan	38
4.1.1	Halaman <i>Login</i>	38
4.1.2	Halaman Menu.....	38
4.1.3	Halaman Data Anggota	39
4.1.4	Halaman Dana Tersedia.....	40
4.1.5	Halaman Data Peminjam	40
4.1.6	Halaman Proses.....	41
4.1.7	Halaman Cari anggota	41
4.1.8	Halaman Bobot Kriteria.....	42
4.1.9	Halaman Data User	42
4.1.10	Halaman Laporan	43
4.2	Hasil Pengujian Sistem	44
4.2.1	Pengujian Fungsional Sistem	44
4.2.2	Pengujian <i>Sampel</i> Data.....	44
4.2.3	Pengujian Sistem Operasi.....	45
4.2.4	Pengujian Sistem Perhitungan	46
BAB V PENUTUP		
5.1.	Kesimpulan	47
5.2.	Saran	47
DAFTAR PUSTAKA		48

DAFTAR GAMBAR

Gambar 2.1 Komponen Sistem Pendukung Keputusan (SPK)	7
Gambar 3.1 Blok Diagram Sistem	23
Gambar 3.2 Struktur Menu Kepala Koperasi	24
Gambar 3.3 ERD (<i>Entity Relationship Diagram</i>)	25
Gambar 3.4 DFD (<i>Data Flow Diagram</i>) Level 0	25
Gambar 3.5 DFD (<i>Data Flow Diagram</i>) Level 1	26
Gambar 3.6 Flowchart SAW(<i>Simple Additive Weighting</i>)	32
Gambar 3.7 Halaman <i>Login</i>	33
Gambar 3.8 Halaman Menu Utama	34
Gambar 3.9 Halaman Data <i>User</i>	34
Gambar 3.10 Halaman Data Anggota	35
Gambar 3.11 Halaman Pemohon Pinjaman	35
Gambar 3.12 Halaman Bobot Pinjaman	36
Gambar 3.13 Halama Dana	36
Gambar 3.14 Halaman Hasil Seleksi Pinjaman	37
Gambar 4.1 <i>Form Login</i>	38
Gambar 4.2 Halaman Menu	39
Gambar 4.3 Halaman Data Anggota	39
Gambar 4.4 Halaman Dana Tersedia	40
Gambar 4.5 Halaman Data Peminjam	40
Gambar 4.6 Halaman Proses	41
Gambar 4.7 Halaman Cari Anggota	41
Gambar 4.8 Halaman Bobot Kriteria	42
Gambar 4.9 Halaman Data User	43
Gambar 4.10 Halaman Laporan Report	43

DAFTAR TABEL

Tabel 3.1 Detail Kriteria Pinjaman.....	28
Tabel 3.2 Nilai Bobot Pinjaman.....	28
Tabel 3.3 Nilai Variabel Logika.....	28
Tabel 3.4 Data Pemohon	29
Tabel 3.5 Rating Kecocokan Dari Setiap Alternatif Pada Setiap Kriteria.....	29
Tabel 4.1 Hasil Pengujian Fungsional Sistem.....	44
Tabel 4.2 Hasil Pengujian <i>Sampel</i> Data	45
Tabel 4.3 Hasil Pengujian Pada Beberapa Sistem Operasi	45
Tabel 4.4 Hasil Perhitungan Manual Dengan Perhitungan Sistem	46

BAB I

PENDAHULUAN

1.1. Latar Belakang

Disetiap organisasi bisnis tidak rentang dengan adanya suatu masalah dalam pengolahan data-data. Di antaranya adalah organisasi koperasi, Menurut ILO atau Organisasi buruh Internasional bahwa pengertian koperasi adalah badan usaha yang beranggotakan orang seorang atau badan hukum. Setiap koperasi yang ada harus melandaskan seluruh kegiatannya pada prinsip koperasi serta asas kekeluargaan untuk meningkatkan gerakan ekonomi rakyat. Koperasi juga merupakan koperasi yang menyelenggarakan pelayanan jasa yang di butuhkan oleh anggota, antara lain yaitu simpan pinjam.

Demikian halnya dengan koperasi wanita simpan pinjam AL-Wardah merupakan koperasi wanita yang ada di desa bangkok glagah lamongan adalah salah satu Koperasi yang dipercaya oleh masyarakat. Semakin tingginya minat masyarakat untuk mendapatkan pinjaman, membuat pihak koperasi kesulitan dalam menentukan siapa yang layak menerima pinjaman atau tidak. Selain itu, proses penentuan siapa yang layak menerima pinjaman masih dilakukan secara manual, sehingga kurang efisien dalam pelaksanaannya. Oleh karena itu, penulis berinisiatif untuk merancang suatu sistem yang dapat membantu pihak koperasi dalam menentukan siapa yang layak menerima pinjaman, sehingga dapat lebih efisien dalam pelaksanaannya.

Dengan permasalahan tersebut maka, perlu adanya solusi pemecahan masalah yang ada dengan membuat suatu sistem pendukung keputusan, tujuan utama dari sistem pendukung keputusan ini adalah membantu dalam proses pengambilan keputusan adalah untuk meningkatkan kemampuannya dalam memutuskan masalah. Keputusan yang dihasilkan nantinya dapat memenuhi batasan yang ditentukan. Sistem pendukung keputusan pemberian pinjaman pada Koperasi Wanita AL-Wardah ini menggunakan metode Simple Additive Weighting (SAW) dengan konsep dasar mencari penjumlahan tebobot dari rating kinerja pada setiap alternatif pada semua atribut atau kriteria peminjam. Sistem ini

berusaha membantu mengatasi kesulitan atau masalah-masalah yang terjadi diatas dan sistem ini lebih bersifat memberikan dukungan atau pertimbangan bagi pihak koperasi dan membantu pihak koperasi dalam pengambilan keputusan sehingga dapat meningkatkan efektivitas dan efisiensi dari proses pengambilan keputusan itu sendiri.

1.2. Rumusan Masalah

Berdasarkan latar belakang masalah di atas maka akan dirumuskan masalah sebagai berikut :

1. Bagaimana cara menerapkan Simple Additive Weighting (SAW) pada sistem pendukung keputusan siapa yang layak menerima pinjaman dari koperasi AL-Wardah ?
2. Bagaimana cara membangun aplikasi yang mampu berjalan pada dekstop ?

1.3. Batasan Masalah

Batasan-batasan masalah yang terdapat dalam pembangunan aplikasi ini adalah:

1. Sistem berbasis pengetahuan yang akan dirancang ini berlaku untuk sistem operasi berbasis Dekstop.
2. Proses yang diteliti adalah proses siapa yang layak mendapat pinjaman dari koperasi wanita AL-WARDAH.
3. Metode yang digunakan dalam penyelesaian masalah ini adalah metode *Simple Additive Weighting* (SAW).
4. Sistem dikembangkan dengan menggunakan software *Visual Basic*.

1.4. Tujuan

Adapun yang menjadi tujuan penulisan dalam penyusunan Skripsi adalah sebagai berikut :

1. Menerapkan metode Simple Additive Weighting (SAW) untuk menentukan siapa yang layak mendapat pinjaman dari koperasi wanita AL-WARDAH.
2. Untuk membantu pengambil keputusan, dalam menentukan siapa saja yang berhak mendapatkan pinjaman sesuai dengan kriteria-kriteria yang telah ditentukan.

1.5. Metode Penulisan

Metodologi yang digunakan dalam penulisan adalah sebagai berikut :

1. Studi Literatur

Studi literatur dilakukan dengan mengumpulkan bahan-bahan referensi baik dari buku, artikel, paper, jurnal, makalah, maupun situs-situs internet. Studi literatur yang dilakukan terkait dengan Sistem Pendukung Keputusan, serta data-data koperasi yang menentukan kelayakan anggota koperasi yang akan meminjam pada koperasi.

2. Analisis Permasalahan

Pada tahap ini dilakukan analisis terhadap *business process* proses menentukan pemberian pinjaman sebelum dikembangkan system pendukung keputusan. Pada tahap ini dilakukan pengamatan terhadap pola pengambilan keputusan yang dilakukan oleh satu orang dalam menentukan anggota yang layak mendapatkan pinjaman.

3. Perancangan Sistem

Pada tahap ini dilakukan pemodelan keputusan, perancangan database untuk pengelolaan data kriteria dan alternatif yang dipakai, perancangan arsitektur sistem dan perancangan algoritma sistem yang sesuai dengan metode *Simple Additive Weighting* (SAW)..

4. Implementasi Sistem

Pada tahap ini dilakukan proses implementasi dengan metode *Simple Additive Weighting* (SAW). pada sistem pendukung keputusan untuk

menyeleksi siapa anggota yang layak mendapatkan pinjaman berdasarkan kriteria kriteria serta bobot yang sudah ditentukan.

5. Pengujian

Pada tahap ini dilakukan proses pengujian, meliputi pengujian penerapan algoritma metode *Simple Additive Weighting* (SAW), berdasarkan jumlah kriteria dan sample alternative yang berbeda, pengujian fungsi-fungsi pada sistem dan pengujian penggunaan system oleh pihak pengambil keputusan di Koperasi Wanita AL-Wardah.

1.6. Sistematika Penulisan

Sistematika penulisan proposal skripsi ditujukan untuk memberikan gambaran dan uraian dari proposal skripsi secara garis besar yang meliputi bab-bab sebagai berikut:

BAB I : PENDAHULUAN

Menguraikan tentang latar belakang, rumusan masalah, batasan masalah, tujuan, dan metode penelitian pada proposal skripsi yang dibuat.

BAB II: TINJAUAN PUSATAKA

Menguraikan tentang pengertian koperasi, penjelasan metode dan metode yang di gunakan adalah *Simple Additive Weighting* (SAW), penjelasan software, dan model system yang yang digunakan, penjelasan software yang digunakan meliputi penggunaan VB.Net, Microsoft SQL Server, pada proposal judul.

BAB III : METODE PELAKSANAAN

Menjelaskan tentang analisa dari sistem yang akan berjalan meliputi alur flowchat pada program, penjelsana konsep DFD yang akan digunakan, menjelaskan tentang struktur menu yang akan dibuat dan hasil gambaran *layout* yang akan diterapkan pada sistem alur sistem.

BAB IV : PENUTUP

Bab ini berisikan uraian tentang kesimpulan dan saran yang dapat digunakan untuk penelitian selanjutnya.

BAB II

LANDASAN TEORI

2.1 Koperasi

Pengertian koperasi menurut undang – undang tahun 1967 adalah system organisasi ekonomi pada rakyat yang memiliki sifat sosial, memiliki beberapa anggota dan berbadan hukum. Koperasi adalah suatu susunan pada ekonomi sebagai salah satu bentuk usaha bersama berdasarkan pada asas kekeluargaan. Koperasi bukan sebuah perkumpulan modal akan tetapi perkumpulan dari orang – orang yang akan menjadi anggota koperasi. System kerjasama yang ada dalam koperasi berdasarkan pada sebuah rasa persamaan suatu derajat, tidak membedakan antara anggota yang satu dengan anggota yang lainnya. Kerja koperasi juga didasari atas adanya rasa kesadaran yang dimiliki oleh seluruh anggotanya. Koperasi dijadikan sebagai salah satu wadah sosial dan juga wadah demokrasi ekonomi. System kerja yang terjadi didalam sebuah koperasi disesuaikan dengan kemauan anggotanya yang dihasilkan melalui proses mufakat yang telah disetujui oleh seluruh anggota koperasi [1].

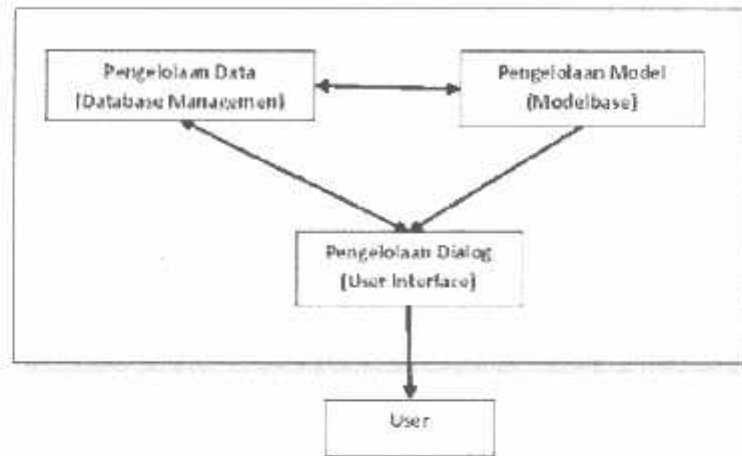
2.2 Sistem Pendukung Keputusan (SPK)

Sistem Pendukung Keputusan (SPK) atau Decision Support System (DSS) adalah sebuah sistem yang mampu memberikan kemampuan pemecahan masalah maupun kemampuan pengkomunikasian untuk masalah dengan kondisi semi terstruktur dan tak terstruktur. Sistem ini digunakan untuk membantu pengambilan keputusan dalam situasi semi terstruktur dan situasi yang tidak terstruktur, dimana tak seorangpun tahu secara pasti bagaimana keputusan seharusnya dibuat (Turban, 2001).

SPK bertujuan untuk menyediakan informasi, membimbing, memberikan prediksi serta mengarahkan kepada pengguna informasi agar dapat melakukan pengambilan keputusan dengan lebih baik .

2.2.1 Komponen Sistem Pendukung Keputusan

Secara umum Sistem Pendukung Keputusan dibangun oleh tiga komponen besar yaitu *database management*, *Model Base* dan *Software System/User Interface*. Komponen SPK tersebut dapat digambarkan seperti gambar 2.1 komponen sistem pendukung keputusan.



Gambar 2.1 Komponen Sistem Pendukung Keputusan (SPK) [3]

a. *Database Management*

Merupakan subsistem data yang terorganisasi dalam suatu basis data. Data yang merupakan suatu sistem pendukung keputusan dapat berasal dari luar maupun dalam lingkungan. Untuk keperluan SPK, diperlukan data yang relevan dengan permasalahan yang hendak dipecahkan melalui simulasi.

b. *Model Base*

Merupakan suatu model yang merepresentasikan permasalahan kedalam format kuantitatif (model matematika sebagai contohnya) sebagai dasar simulasi atau pengambilan keputusan, termasuk didalamnya tujuan dari permasalahan (objektif), komponen-komponen terkait, batasan-batasan yang ada (*constraints*), dan hal-hal terkait lainnya. Model Base memungkinkan pengambil keputusan menganalisa secara utuh dengan mengembangkan dan membandingkan solusi alternatif.

c. *User Interface*

Terkadang disebut sebagai subsistem dialog, merupakan penggabungan antara dua komponen sebelumnya yaitu *Database Management* dan *Model Base* yang disatukan dalam komponen ketiga (*user interface*), setelah sebelumnya dipresentasikan dalam bentuk model yang dimengerti *computer*. *User Interface* menampilkan keluaran sistem bagi pemakai dan menerima masukan dari pemakai kedalam Sistem Pendukung Keputusan.

2.3 *Microsoft Visual Studio*

Microsoft Visual studio .NET versi terbaru dengan *.NET Framework 4.5* merupakan pengembangan kelanjutan dari Microsoft Visual Studio *.NET Framework 4.0* sebelumnya, yaitu Visual Studio 2012 yang diproduksi oleh Microsoft.

Pada bulan Februari tahun 2002 Microsoft memproduksi teknologi *.NET Framework* versi 1.0, teknologi *.NET* ini didasarkan atas susunan berupa *.NET Framework*, sehingga setiap produk baru yang terkait dengan teknologi *.NET* akan selalu berkembang mengikuti perkembangannya *.NET Framework* dengan berbasis *Graphical User Interface (GUI)*. Pada perkembangan nantinya, mungkin untuk membuat program dengan teknologi *.NET*, memungkinkan para pengembang perangkat lunak akan dapat menggunakan lintas sistem operasi, yaitu dapat dikembangkan di sistem operasi Windows juga dapat dijalankan pada sistem operasi lain, misalkan pada sistem operasi Linux, seperti pada pemrograman java oleh Sun Microsystem. Menggunakan teknologi ADO *.NET* kelebihan-kelebihan yang ditawarkan, terutama memungkinkan pengembangan perangkat lunak secara cepat mampu membuat program yang *robust*, serta berbasiskan *integrasi* ke internet yang dikenal dengan *XML Web Service*. (Darmayuda, Ketut. 2014)

2.4 *Microsoft SQL Server*

Microsoft SQL Server adalah sebuah sistem manajemen basis data relasional (RDBMS) produk Microsoft. Bahasa query utamanya adalah Transact-

SQL yang merupakan implementasi dari SQL standar ANSI/ISO yang digunakan oleh Microsoft dan Sybase. Umumnya SQL Server digunakan di dunia bisnis yang memiliki basis data berskala kecil sampai dengan menengah, tetapi kemudian berkembang dengan digunakannya SQL Server pada basis data besar. Microsoft SQL Server dan Sybase/ASE dapat berkomunikasi lewat jaringan dengan menggunakan protokol TDS (Tabular Data Stream). Selain dari itu, Microsoft SQL Server juga mendukung ODBC (*Open Database Connectivity*), dan mempunyai driver JDBC untuk bahasa pemrograman Java. Fitur yang lain dari SQL Server ini adalah kemampuannya untuk membuat basis data mirroring dan clustering. Pada versi sebelumnya, MS SQL Server 2000 terserang oleh cacing komputer SQL Slammer yang mengakibatkan kelambatan akses Internet pada tanggal 25 Januari 2003. (Darmayuda, Ketut. 2014).

2.5 *Multiple Attribute Decision Making (MADM)*

Pada dasarnya, proses MADM dilakukan melalui 3 tahap, yaitu: penyusunan komponen-komponen situasi, analisis, dan sintesis informasi pada tahap penyusunan komponen-komponen situasi, akan dibentuk tabel taksiran yang berisi identifikasi alternatif dan spesifikasi tujuan, kriteria dan atribut. Salah satu cara untuk menspesifikasikan tujuan situasi $|O_i, i=1, \dots, t|$ adalah dengan cara mendaftar konsekuensi-konsekuensi yang mungkin dari alternatif yang telah teridentifikasi $|A_i, i=1, \dots, n|$. Selain itu juga disusun atribut-atribut yang akan digunakan $|a_k, k=1, \dots, m|^{[2]}$.

Tahap analisis dilakukan melalui 2 langkah. Pertama, mendatangkan taksiran dari besaran yang potensial, kemungkinan dan ketidakpastian yang berhubungan dengan dampak-dampak yang mungkin pada setiap alternatif. Kedua, meliputi pemilihan dari preferensi pengambil keputusan untuk setiap nilai, dan ketidakpedulian terhadap resiko yang timbul. Pada langkah pertama, beberapa metode menggunakan fungsi distribusi $|p_j(x)|$ yang menyatakan probabilitas kumpulan atribut $|a_k|$ terhadap setiap alternatif $|A_i|$. Konsekuensi juga dapat ditentukan secara langsung dari agregasi sederhana yang dilakukan pada informasi terbaik yang tersedia. Demikian pula, ada beberapa cara untuk menentukan preferensi pengambil keputusan pada setiap konsekuensi yang dapat dilakukan pada

langkah kedua. Metode yang paling sederhana adalah untuk menurunkan bobot atribut dan kriteria adalah dengan fungsi utilitas atau penjumlahan terbobot^[2].

Secara umum, *model multi attribute decision making* dapat didefinisikan sebagai berikut^[2]:

Misalkan $A = \{a_i \mid i=1, \dots, n\}$ adalah himpunan alternatif-alternatif keputusan dan $C = \{c_j \mid j=1, \dots, m\}$ adalah himpunan tujuan yang diharapkan, maka akan ditentukan alternatif x^0 yang memiliki derajat harapan tertinggi terhadap tujuan-tujuan yang relevan c_j ^[2].

Sebagian besar pendekatan MADM dilakukan melalui 2 langkah yaitu: pertama, melakukan agregasi terhadap keputusan-keputusan yang tanggap terhadap semua tujuan pada setiap alternatif. Kedua, melakukan perankingan alternatif-alternatif keputusan tersebut berdasarkan hasil agregasi keputusan^[2].

Dengan demikian, bisa dikatakan bahwa, masalah *multi attribute decision making* (MADM) adalah mengevaluasi m alternatif A_i ($i=1, 2, \dots, m$) terhadap sekumpulan atribut atau kriteria C_j ($j=1, 2, \dots, n$), dimana setiap atribut saling tidak bergantung satu dengan yang lainnya. Matrik keputusan setiap alternatif terhadap setiap atribut X , diberikan sebagai^[2]:

$$X = \begin{bmatrix} x_{11} & x_{12} & \dots & x_{1n} \\ x_{21} & x_{22} & \dots & x_{2n} \\ \vdots & \vdots & \dots & \vdots \\ x_{m1} & x_{m2} & \dots & x_{mn} \end{bmatrix} \dots\dots\dots(1)$$

Dimana x_{ij} merupakan rating kinerja alternatif ke- i terhadap atribut ke- j . nilai bobot yang menunjukkan tingkat kepentingan relatif setiap atribut, diberikan sebagai, W ^[2]:

$$W = \{w_1, w_2, \dots, w_n\} \dots\dots\dots(2)$$

Rating kinerja (X), dan nilai bobot (W) merupakan nilai utama yang merepresentasikan preferensi absolut dari pengambil keputusan. Masalah MADM diakhiri dengan proses perankingan untuk mendapatkan alternatif terbaik yang diperoleh berdasarkan nilai keseluruhan preferensi yang diberikan^[2].

2.5.1 Metode-Metode Penyelesaian Masalah MADM (*Multi Attribute Decision Making*)

Ada beberapa metode yang dapat digunakan untuk menyelesaikan masalah MADM, antara lain ^[2]:

1. *Simple Additive Weighting Method* (SAW)

Metode *Simple Additive Weighting* (SAW) sering juga dikenal istilah metode penjumlahan terbobot. Konsep dasar metode SAW adalah mencari penjumlahan terbobot dari rating kinerja pada setiap alternatif pada semua atribut (Fishburn, 1967) (MacCrimmon, 1968). Metode SAW membutuhkan proses normalisasi matriks keputusan (X) ke suatu skala yang dapat diperbandingkan dengan semua rating alternatif yang ada ^[2].

$$r_{ij} = \begin{cases} \frac{x_{ij}}{\max x_{ij}} & \text{jika } j \text{ adalah atribut kebermanfaatan (benefit)} \\ \frac{\min x_{ij}}{x_{ij}} & \text{jika } j \text{ adalah atribut biaya (cost)} \end{cases} \dots\dots\dots (1)$$

Dimana r_{ij} adalah nilai rating kinerja ternormalisasi dari alternative A_i pada atribut C_j ; $i = 1,2,\dots,m$ dan $j=1,2,\dots,n$. Nilai preferensi untuk setiap alternatif (V_i) diberikan sebagai ^[2]:

$$V_i = \sum_{j=1}^n w_j r_{ij} \dots\dots\dots (2)$$

Nilai V_i yang lebih besar mengindikasikan bahwa alternatif A_i lebih terpilih ^[2].

2. *Weighted Product* (WP)

Metode *Weighted Product* (WP) menggunakan perkalian untuk menghubungkan rating atribut, dimana rating setiap atribut harus dipangkatkan dulu dengan bobot atribut yang bersangkutan. Proses ini sama halnya dengan proses normalisasi. Preferensi untuk alternatif A_i diberikan sebagai berikut ^[2]:

$$S_i = \prod_{j=1}^n x_{ij}^{w_j}$$

dengan $i=1,2,\dots,m$; dimana $\sum w_j = 1$.

w_j adalah pangkat bernilai positif untuk atribut keuntungan, dan bernilai negatif untuk atribut biaya.

3. *Elimination and Choise Expressing Reality (ELECTRE)*

Menurut Janko dan Bernoider (2005:11), ELECTRE merupakan salah satu metode pengambilan keputusan multikriteria berdasarkan pada konsep outranking dengan menggunakan perbandingan berpasangan dari alternatif-alternatif berdasarkan setiap kriteria yang sesuai [2].

Metode ELECTRE digunakan pada kondisi di mana alternatif yang sesuai dapat dihasilkan. Dengan kata lain, ELECTRE digunakan untuk kasus-kasus dengan banyak alternatif namun hanya sedikit kriteria yang dilibatkan [2].

Suatu alternatif dikatakan mendominasi alternatif yang lainnya jika satu atau lebih kriterianya melebihi (dibandingkan dengan kriteria dari alternatif yang lain) dan sama dengan kriteria lain yang tersisa [2].

Langkah-langkah yang dilakukan dalam penyelesaian masalah menggunakan metode electre adalah sebagai berikut [2].

Langkah 1 : Normalisasi matrik keputusan.

Dalam prosedur ini, setiap atribut diubah menjadi nilai yang comparable. Setiap normalisasi dari nilai r_{ij} dapat dilakukan dengan Rumus (1) :

$$r'_{ij} = \frac{x_{ij}}{\sqrt{\sum_{i=1}^m x_{ij}^2}}, \text{ untuk } i=1,2,3,\dots,m \text{ dan } j=1,2,3,\dots,n. \dots\dots\dots(1)$$

Sehingga didapat matriks R hasil normalisasi,

$$R = \begin{bmatrix} r_{11} & r_{12} & \dots & r_{1n} \\ r_{21} & r_{22} & \dots & r_{2n} \\ \dots & \dots & \dots & \dots \\ r_{mj} & r_{m2} & \dots & r_{mn} \end{bmatrix}$$

R adalah matriks yang telah dinormalisasi, dimana m menyatakan alternatif, n menyatakan kriteria dan r_{ij} adalah normalisasi pengukuran pilihan dari alternatif ke- i dalam hubungannya dengan kriteria ke- j .

Langkah 2 : Pembobotan pada matrik yang telah dinormalisasi.

Setelah di normalisasi, setiap kolom dari matrik R dikalikan dengan bobot-bobot (w_j) yang ditentukan oleh pembuat keputusan. Sehingga, *weighted normalized matrix* adalah $V=RW$ yang ditulis dalam Rumus (2) ini :

$$V = \begin{bmatrix} v_{11} & v_{12} & \dots & v_{1n} \\ v_{21} & v_{22} & \dots & v_{2n} \\ \dots & \dots & \dots & \dots \\ v_{mj} & v_{m2} & \dots & v_{mn} \end{bmatrix}$$

$$RW = \begin{bmatrix} w_1 r_{11} & w_2 r_{12} & \dots & w_n r_{1n} \\ w_1 r_{21} & w_2 r_{22} & \dots & w_n r_{2n} \\ \dots & \dots & \dots & \dots \\ w_1 r_{mj} & w_2 r_{m2} & \dots & w_n r_{mn} \end{bmatrix} \dots\dots\dots(2)$$

dimana W adalah

$$W = \begin{bmatrix} w_1 & 0 & 0 & \dots & 0 \\ 0 & w_2 & 0 & \dots & \\ \dots & \dots & \dots & \dots & \\ 0 & 0 & 0 & \dots & w_n \end{bmatrix}, \text{ dan } \sum_{i=1}^n w_i = 1$$

Langkah 3 : Menentukan *concordance* dan *discordance set*.

Untuk setiap pasang dari alternatif k dan l ($k, l = 1, 2, 3, \dots, m$ dan $k \neq l$) kumpulan kriteria J dibagi menjadi dua *subsets*, yaitu *concordance* dan *discordance*. Bilamana sebuah kriteria dalam suatu alternatif termasuk *concordance* adalah :

$$C_{kl} = \{j, y_{kj} \geq y_{lj}\}, \text{ untuk } j = 1, 2, 3, \dots, n \quad \dots\dots\dots(3)$$

Sebaliknya, komplementer dari subset

ini adalah *discordance*, yaitu bila :

$$D_{kl} = \{j, y_{kj} < y_{lj}\}, \text{ untuk } j = 1, 2, 3, \dots, n \quad \dots\dots\dots(4)$$

Langkah 4 : Hitung matriks *concordance* dan *discordance*.

a. *Concordance*

Untuk menentukan nilai dari elemenelemen pada matriks *concordance* adalah dengan menjumlahkan bobot – bobot yang termasuk dalam subset *concordance*, secara matematisnya adalah pada Rumus (5):

$$c_{kl} = \sum_{j \in C_{kl}} w_j \quad \dots\dots\dots(5)$$

Sehingga matrik *concordance* yang dihasilkan adalah :

$$C = \begin{bmatrix} - & c_{12} & c_{13} & \dots & c_{1n} \\ c_{21} & - & c_{23} & \dots & c_{2n} \\ \dots & & & & \\ c_{m1} & c_{m2} & c_{m3} & \dots & - \end{bmatrix}$$

b. *Discordance*

Untuk menentukan nilai dari elemenelemen pada matriks *discordance* adalah dengan membagi maksimum selisih nilai kriteria yang termasuk dalam subset *discordance* dengan maksimum selisih nilai seluruh kriteria yang ada, secara matematisnya adalah :

$$d_{kl} = \frac{\left\{ \max (v_{mn} - v_{mn-ln}) \right\}; m, n \in D_{kl}}{\left\{ \max (v_{mn} - v_{mn-ln}) \right\}; m, n = 1, 2, 3, \dots} \dots\dots(6)$$

Sehingga diperoleh matrik *discordance* :

$$D = \begin{bmatrix} - & d_{12} & d_{13} & \dots & d_{1m} \\ d_{21} & - & d_{23} & \dots & d_{2m} \\ \dots & & & & \\ d_{m1} & d_{m2} & d_{m3} & \dots & - \end{bmatrix}$$

Langkah 5 : Menentukan matrik dominan *concordance* dan *discordance*.

a. *Concordance*

Matrik dominan *concordance* dapat dibangun dengan bantuan nilai *threshold*, yaitu dengan membandingkan setiap nilai elemen matriks *concordance* dengan nilai *threshold*.

$$C_{kl} \geq c$$

dengan nilai *threshold* (c), adalah :

$$c = \frac{\sum_{k=1}^n \sum_{l=1}^n c_{kl}}{m * (m - 1)} \dots\dots\dots(7)$$

dan nilai setiap elemen matriks F sebagai matriks dominan *concordance* ditentukan sbb :

$$f_{kl} = 1, \text{ jika } c_{kl} \geq c \text{ dan } f_{kl} = 0, \text{ jika } c_{kl} < c$$

b. *Discordance*

Untuk membangun matriks dominan *discordance* juga menggunakan bantuan nilai *threshold*, yaitu :

$$d = \frac{\sum_{k=1}^n \sum_{l=1}^n d_{kl}}{m * (m - 1)} \dots\dots\dots(8)$$

dan nilai setiap elemen untuk matriks G sebagai matriks dominan *discordance* ditentukan sebagai berikut :

$$g_{kl} = 0, \text{ jika } c_{kl} \geq d \text{ dan } g_{kl} = 1, \text{ jika } c_{kl} < d$$

Langkah 6 : Menentukan *aggregate dominance matrix*.

Langkah selanjutnya adalah menentukan *aggregate dominance matrix* sebagai matriks E , yang setiap elemennya merupakan perkalian antara elemen matriks F dengan elemen matriks G , sebagai berikut :

$$e_{kl} = f_{kl} \times g_{kl} \dots\dots\dots(9)$$

Langkah 7 : Eliminasi alternatif yang *less favourable*.

Matriks E memberikan urutan pilihan dari setiap alternatif, yaitu bila $e_{kl} = 1$ maka alternatif A_k merupakan pilihan yang lebih baik daripada A_l . Sehingga baris dalam matriks E yang memiliki jumlah $e_{kl} = 1$ paling sedikit dapat dieliminasi. Dengan demikian alternatif terbaik adalah yang mendominasi alternatif lainnya.

4. *Technique For Order Preference by Similarity to Ideal Solution* (TOPSIS)

Technique for Order Preference by Similarity to Ideal Solution (TOPSIS) didasarkan pada konsep dimana alternatif terpilih yang terbaik tidak hanya memiliki jarak terpendek dari solusi ideal positif, namun juga memiliki jarak terpanjang dari solusi ideal negatif. TOPSIS banyak digunakan dengan alasan konsepnya sederhana dan mudah dipahami, komputasinya efisien dan memiliki kemampuan untuk mengukur kinerja relatif dari alternatif-alternatif keputusan dalam bentuk matematis yang sederhana [2].

TOPSIS membutuhkan rating kinerja setiap alternatif A_i pada setiap kriteria C_j yang ternormalisasi, yaitu ^[2]:

$$r_{ij} = \frac{x_{ij}}{\sqrt{\sum_{i=1}^m x_{ij}^2}}$$

Solusi ideal positif A^+ dan solusi ideal negatif A^- dapat ditentukan berdasarkan rating bobot ternormalisasi (y_j^0) sebagai ^[2]:

$$y_j = w_j r_{ij}$$

$$A^+ = (y_1^+, y_2^+, \dots, y_n^+)$$

$$A^- = (y_1^-, y_2^-, \dots, y_n^-)$$

dengan

$$y_j^+ = \begin{cases} \max_i y_{ij}; & \text{jika } j \text{ adalah atribut keuntungan} \\ \min_i y_{ij}; & \text{jika } j \text{ adalah atribut biaya} \end{cases}$$

$$y_j^- = \begin{cases} \min_i y_{ij}; & \text{jika } j \text{ adalah atribut keuntungan} \\ \max_i y_{ij}; & \text{jika } j \text{ adalah atribut biaya} \end{cases}$$

Jarak antara alternatif A_i dengan *solusi ideal positif* dirumuskan sebagai:

$$D_i^+ = \sqrt{\sum_{j=1}^n (y_{ij}^+ - y_{ij})^2};$$

Jarak antara alternatif A_i dengan *solusi ideal negatif* dirumuskan sebagai:

$$D_i^- = \sqrt{\sum_{j=1}^n (y_{ij} - y_{ij}^-)^2};$$

Nilai *preferensi* untuk setiap alternatif (V_i) diberikan sebagai:

$$V_i = \frac{D_i^-}{D_i^- + D_i^+};$$

Nilai V_i yang lebih besar menunjukkan bahwa alternatif A_i lebih dipilih

5. Analytic Hierarchy Process (AHP)

Analytical Hierarchy Process(AHP) adalah salah satu bentuk metode pengambilan keputusan yang pada dasarnya berusaha menutupi semua kekurangan dari metode sebelumnya. Peralatan utama dari metode

AHP adalah sebuah hirarki fungsional dengan input utamanya adalah persepsi manusia. Dengan hirarki, suatu yang kompleks dan tidak terstruktur dipecahkan ke dalam kelompok dan kemudian kelompok tersebut diatur menjadi suatu bentuk hirarki [2].

Langkah yang harus dilakukan dalam menyelesaikan persoalan dengan AHP (Mulyono, 1996:108) yaitu [2]:

a. Decomposition

Decomposition adalah proses menganalisa permasalahan riil dalam struktur hirarki atas unsur – unsur pendukungnya. Struktur hirarki secara umum dalam metode AHP yaitu: Jenjang 1 : Goal atau Tujuan, Jenjang 2 : Kriteria, Jenjang 3 : Subkriteria (optional), Jenjang 4 : Alternatif.

b. Comperative judgment

Comperative judgment adalah berarti membuat suatu penilaian tentang kepentingan relatif antara dua elemen pada suatu tingkat tertentu yang disajikan dalam bentuk matriks dengan menggunakan skala prioritas. Jika terdapat n elemen, maka akan diperoleh matriks pairwise comparison (matriks perbandingan) berukuran $n \times n$ dan banyaknya penilaian yang diperlukan adalah $n(n-1)/2$. Ciri utama dari matriks perbandingan yang dipakai dalam metode AHP adalah elemen diagonalnya dari kiri atas ke kanan bawah adalah satu karena elemen yang dibandingkan adalah dua elemen yang sama. Selain itu, sesuai dengan sistematika berpikir otak manusia, matriks perbandingan yang terbentuk akan bersifat matriks resiprokal dimana apabila elemen A lebih disukai dengan skala 3 dibandingkan elemen B, maka dengan sendirinya elemen B lebih disukai dengan skala $1/3$ dibanding elemen A.

Dengan dasar kondisi – kondisi di atas dan skala standar input AHP dari 1 sampai 9, maka dalam matriks perbandingan tersebut angka terendah yang mungkin terjadi adalah $1/9$, sedangkan angka tertinggi yang mungkin terjadi adalah $9/1$. Angka 0 tidak dimungkinkan dalam matriks ini, sedangkan pemakaian skala dalam bentuk desimal dimungkinkan sejauh si

expert memang menginginkan bentuk tersebut untuk persepsi yang lebih akurat.

c. Synthesis of priority

Setelah matriks perbandingan untuk sekelompok elemen selesai dibentuk maka langkah berikutnya adalah mengukur bobot prioritas setiap elemen tersebut. Hasil akhir dari penghitungan bobot prioritas tersebut adalah suatu bilangan desimal di bawah satu (misalnya 0.01 sampai 0.99) dengan total prioritas untuk elemen – elemen dalam satu kelompok sama dengan satu. Bobot prioritas dari masing – masing matriks dapat menentukan prioritas lokal dan dengan melakukan sintesa di antara prioritas lokal, maka akan didapat prioritas global.

Usaha untuk memasukkan kaitan antara elemen yang satu dengan elemen yang lain dalam menghitung bobot prioritas secara sederhana dapat dilakukan dengan cara berikut:

1. Jumlahkan elemen pada kolom yang sama pada matriks perbandingan yang terbentuk. Lakukan hal yang sama untuk setiap kolom.
2. Bagilah setiap elemen pada setiap kolom dengan jumlah elemen kolom tersebut (hasil dari langkah 1). Lakukan hal yang sama untuk setiap kolom sehingga akan terbentuk matrik yang baru yang elemen – elemennya berasal dari hasil pembagian tersebut.
3. Jumlahkan elemen matrik yang baru tersebut menurut barisnya.
4. Bagilah hasil penjumlahan baris (hasil dari langkah 3) dengan total alternatif agar didapatkan prioritas terakhir setiap elemen dengan total bobot prioritas sama dengan satu.

Proses yang dilakukan untuk membuat total bobot prioritas sama dengan satu biasa disebut proses normalisasi.

d. Logical consistency

Salah satu asumsi utama metode AHP yang membedakannya dengan metode yang lainnya adalah tidak adanya syarat konsistensi mutlak. Dengan metode AHP yang memakai persepsi manusia sebagai inputannya maka ketidakkonsistenan itu mungkin terjadi karena manusia mempunyai keterbatasan dalam menyatakan persepsinya secara konsisten terutama kalau membandingkan banyak elemen. Berdasarkan konsisi ini maka manusia dapat menyatakan persepsinya dengan bebas tanpa harus berpikir apakah persepsinya tersebut akan konsisten nantinya atau tidak. Persepsi yang 100 % konsisten belum tentu memberikan hasil yang optimal atau benar dan sebaliknya persepsi yang tidak konsisten penuh mungkin memberikan gambaran keadaan yang sebenarnya atau yang terbaik.

Penentuan nilai preferansi antar elemen harus secara konsisten logis, yang dapat diukur dengan menghitung Consistency Index (CI) dan Consistency Ratio (CR)

$$CI = \frac{\lambda - n}{n - 1}$$

$$CR = \frac{CI}{RI}$$

dimana : λ = eigen value, n = ukuran matriks, RI = Random Index

Untuk mendapatkan nilai λ digunakan rumus berikut:

$$\lambda = \frac{\sum_{k=1}^n a_{ki} p_i}{p_i}$$

dimana a_{ki} = nilai perbandingan dari elemen ke-1 dengan elemen ke-i,

p_i = nilai prioritas dari elemen ke-i.

Untuk metode AHP, tingkat inkonsistensi yang masih bisa diterima adalah sebesar 10% ke bawah. Jadi apabila nilai $CR \leq 0.1$ maka hasil preferensi cukup baik dan sebaliknya jika $CR > 0.1$ hasil proses AHP tidak valid sehingga harus diadakan revisi penilaian karena tingkat inkonsistensi yang terlalu besar dapat menjurus pada suatu kesalahan.

2.5.2 Metode *Simple Additive Weighting* (SAW)

Metode *Simple Additive Weighting* (SAW) sering juga dikenal istilah metode penjumlahan terbobot. Konsep dasar metode SAW adalah mencari penjumlahan terbobot dari rating kinerja pada setiap alternatif pada semua atribut (Fishburn, 1967) (MacCrimmon, 1968). Metode SAW membutuhkan proses normalisasi matriks keputusan (X) ke suatu skala yang dapat diperbandingkan dengan semua rating alternatif yang ada [2].

$$r_{ij} = \begin{cases} \frac{x_{ij}}{\text{Max } x_{ij}} & \text{jika } j \text{ adalah atribut keberuntungan (benefit)} \\ \frac{\text{Min } x_{ij}}{x_{ij}} & \text{jika } j \text{ adalah atribut biaya (cost)} \end{cases} \dots\dots\dots (1)$$

Dimana r_{ij} adalah nilai rating kinerja ternormalisasi dari alternative A_i pada atribut C_j ; $i = 1, 2, \dots, m$ dan $j = 1, 2, \dots, n$. Nilai preferensi untuk setiap alternatif (V_i) diberikan sebagai [2].

$$V_i = \sum_{j=1}^n w_j r_{ij} \dots\dots\dots (2)$$

Nilai V_i yang lebih besar mengindikasikan bahwa alternatif A_i lebih terpilih [2].

2.5.3 Langkah Penyelesaian MADM Metode SAW

Dalam penelitian ini menggunakan MADM dengan metode SAW. Adapun langkah-langkahnya adalah [2].

1. Memberikan nilai setiap alternatif (A_i) pada setiap kriteria (C_j) yang sudah ditentukan, dimana nilai $i = 1, 2, \dots, m$ dan $j = 1, 2, \dots, n$.
2. Memberikan nilai bobot (W) yang juga didapatkan berdasarkan nilai *crisp*.
3. Melakukan normalisasi matriks dengan cara menghitung nilai rating kinerja ternormalisasi (r_{ij}) dari alternatif A_i pada atribut C_j berdasarkan persamaan yang disesuaikan dengan jenis atribut (atribut

keuntungan/*benefit* = MAKSIMUM atau atribut biaya/*cost* = MINIMUM). Apabila berupa atribut keuntungan maka nilai crisp (X_{ij}) dari setiap kolom atribut dibagi dengan nilai crisp MAX (MAX X_{ij}) dari tiap kolom, sedangkan untuk atribut biaya, nilai crisp MIN (MIN X_{ij}) dari tiap kolom atribut dibagi dengan nilai crisp (X_{ij}) setiap kolom.

4. Melakukan proses perankingan untuk setiap alternatif (V_i) dengan cara mengalikan nilai bobor (w_i) dengan nilai rating kinerja ternormalisasi (r_{ij}).

2.6 Teori *Flowchart*

Flowchart adalah penggambaran secara grafik dari langkah-langkah dan urutan-prosedur dari suatu program. *Flowchart* menolong analis dan programmer untuk memecahkan masalah kedalam segmen-segmen yang lebih kecil dan menolong dalam menganalisis alternatif-alternatif lain dalam pengoperasian.

Flowchart biasanya mempermudah penyelesaian suatu masalah khususnya masalah yang perlu dipelajari dan dievaluasi lebih lanjut.

BAB III

ANALISIS DAN PERANCANGAN

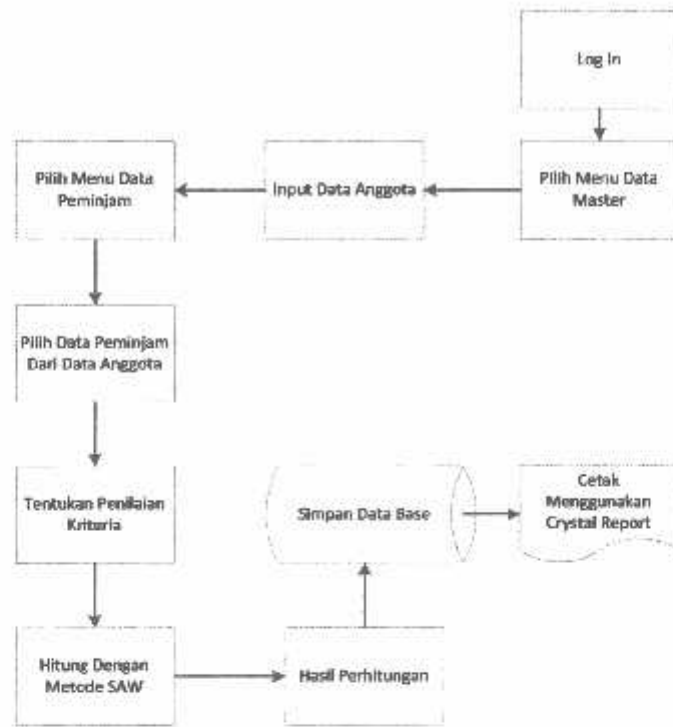
3.1 Identifikasi Masalah

Permasalahan pada penentuan anggota yang berhak menerima pinjaman pada Koperasi Wanita AL-Wardah ternyata belum mendekati tepat. Seperti dari segi penentuan siapa anggota yang berhak mendapatkan pinjaman misalnya anggota yang sebenarnya tidak layak mendapatkan pinjaman namun mendapatkan pinjaman sebaliknya anggota yang berhak mendapatkan pinjaman tetapi tidak mendapatkan pinjaman. Dari permasalahan tersebut dapat diidentifikasi bahwa dalam penentuan anggota yang berhak menerima pinjaman belum mempergunakan alat bantu, metode atau aplikasi pendukung keputusan secara khusus.

3.2 Perancangan Proses

1. Blok Diagram

Blok diagram adalah diagram dari sistem di mana bagian utama atau fungsi yang diwakili oleh blok dihubungkan dengan garis yang menunjukkan hubungan dari blok. Pada Gambar 3.1



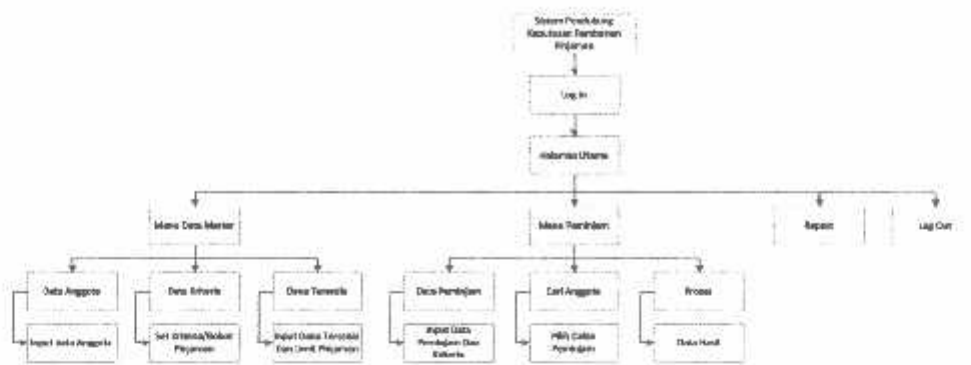
Gambar 3.1. Blok Diagram Sistem

Dari keterangan gambar di atas yaitu :

- a. Untuk menjalankan program harus login terlebih dahulu, setelah login berhasil pilih menu data master, kemudian input data anggota.
- b. Pilih menu data peminjam, setelah itu pilih data peminjam dari data anggota, selanjutnya tentukan penilain kriteria. Lalu hitung nilai tersebut dengan metode SAW.
- c. Simpan data hasil perhitungan kemudian cetak hasil perhitungan.

2. Struktur Menu

Struktur menu merupakan struktur yang menampilkan menu apa saja yang terdapat di dalam aplikasi. Struktur menu yang akan dirancang di dalam aplikasi ini seperti pada Gambar 3.2 berikut.



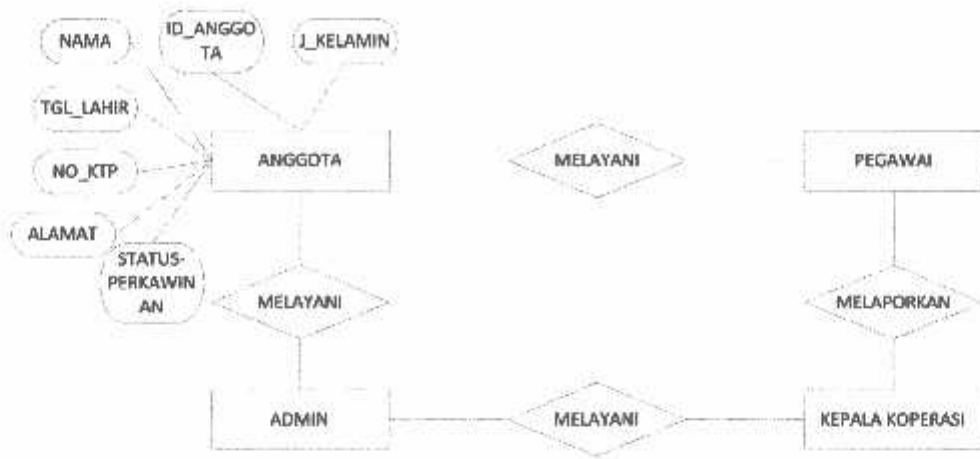
Gambar 3.2. Struktur Menu Kepala Koperasi

Dari keterangan gambar di atas yaitu :

- Halaman pertama yang muncul dari program ini adalah halaman login.
- Setelah itu masuk ke halaman utama, dan pada halaman utama ada 4 menu yaitu Menu Data Master, Menu Pinjaman, Report dan Logout.
- Pada menu data master berisi pilihan data anggota, data kriteria dan dana tersedia.
- Pada menu peminjam berisi pilihan data peminjam, cari anggota dan proses.
- Menu report berisi hasil perhitungan dan menu logout untuk keluar dari program.

3. ERD (Entity Relationship Diagram)

ERD merupakan model data berupa notasi grafis dalam pemodelan data konseptual yang menggambarkan hubungan antara penyimpanan. ERD dari aplikasi ini ditunjukkan seperti pada Gambar 3.3 di bawah ini.



Gambar 3.3 ERD

4. DFD (Data Flow Diagram)

DFD adalah suatu diagram yang menggunakan notasi-notasi untuk menggambarkan arus dari data pada suatu sistem, yang penggunaannya sangat membantu untuk memahami sistem secara logika, terstruktur dan jelas.

a. DFD Level 0

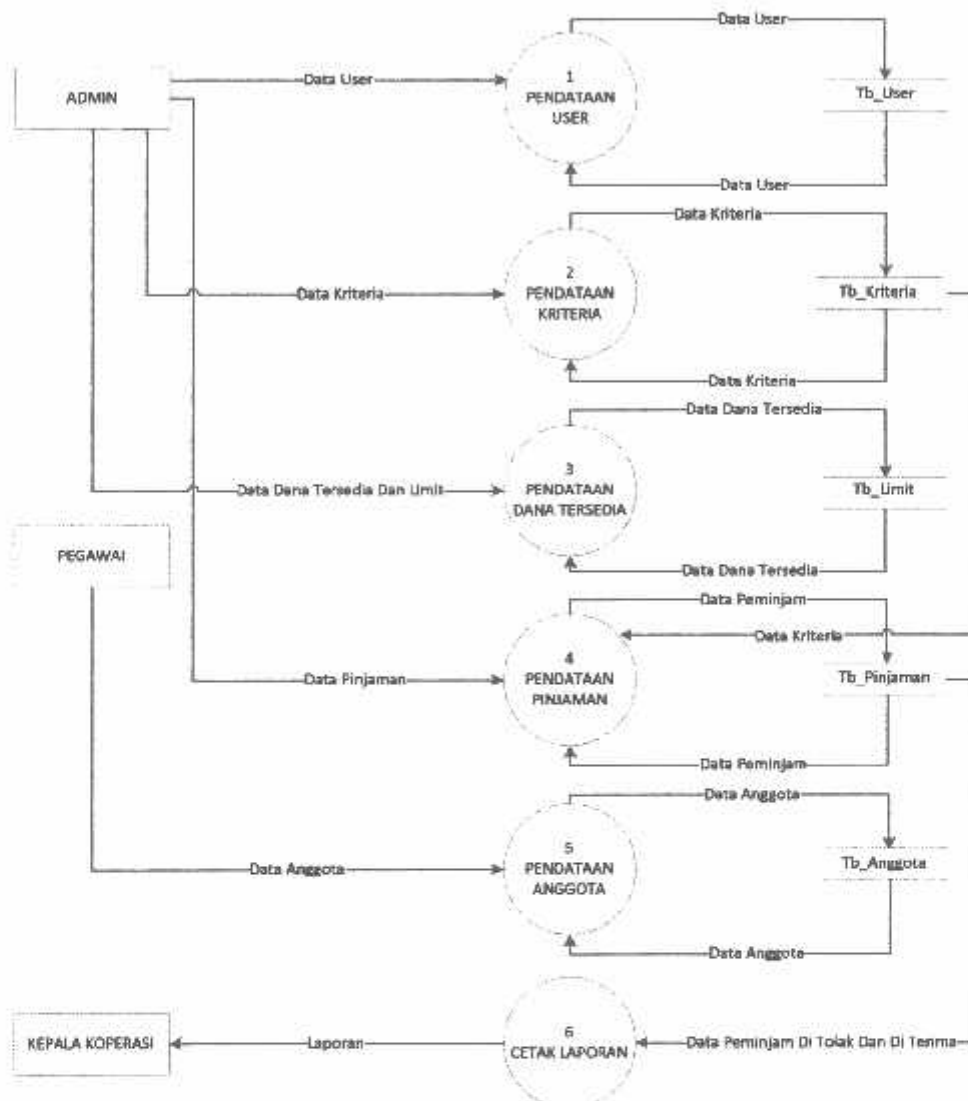
DFD Level 0 menggambarkan satu lingkaran besar yang mewakili seluruh proses yang terdapat di dalam suatu sistem. Berikut DFD Level 0 dari aplikasi spk ini ditunjukkan pada gambar 3.4.



Gambar 3.4. DFD Level 0

b. DFD Level 1

DFD Level 1 merupakan satu lingkaran besar yang mewakili lingkaran-lingkaran kecil yang ada di dalamnya dan merupakan pemecahan dari DFD Level 0. DFD Level 1 aplikasi ini ditunjukkan pada Gambar 3.5 di bawah ini.



Gambar 3.5. DFD Level 1

3.3 Hak Akses Pengguna Aplikasi

Hak akses dalam program ini ada 3 yaitu admin, bagian pendataan, dan kepala dinas. Yang penjelasannya sebagai berikut :

1. Admin bertugas untuk mendaftarkan pengguna-pengguna yang akan menggunakan aplikasi pada menu daftar pengguna dan mengedit bagian menu tentang system, pendataan kriteria, pendataan dana tersedia dan pendataan pinjaman.
2. Kepala Kantor bisa mencetak laporan
3. Pegawai bisa melihat rekap laporan.

3.4 Pemodelan Keputusan

3.4.1 Proses Pengambilan Keputusan

Untuk menghasilkan keputusan siapa anggota penerima pinjaman tertentu harus berdasarkan beberapa kriteria yang disesuaikan dengan *standar operating procedure* seleksi penerima pinjaman di Koperasi Wanita AL-Wardah. Beberapa kriteria tersebut memiliki bobot penilaian yang berbeda, penentuan bobot kriteria juga disesuaikan dengan *standar operating procedure* seleksi penerima Koperasi Wanita AL-Wardah. Rincian kriteria dan pembobotannya digambarkan dalam bentuk tabel keputusan.

- Tabel Keputusan Pinjaman

Tabel keputusan pemberian pinjaman membutuhkan beberapa kriteria, detail kriteria dan bobot nilai yang dibutuhkan ditunjukkan dalam Tabel 3.1 dan dan Tabel 3.2.

Tabel 3.1 Detail Kriteria Pinjaman

Kriteria	Keterangan	Nilai
Jaminan	>100.000	0.25
	>2000.000	0.5
	>3000.000	0.75
	>4000.000	1
Pinjaman	>100.000	0.25
	>2000.000	0.5
	>3000.000	0.75

	>4000.000	1
Gaji	>100.000	0.25
	>1000.000	0.5
	>2000.000	0.75
	>3000.000	1
Jangka Waktu	>6 bulan	1
	>12 bulan	0.75
	>18 bulan	0.5
	>24 bulan	0.25

Tabel 3.2 Nilai Bobot Pinjaman

Kriteria	Bobot
Jaminan	45%
Pinjaman	5%
Gaji	35 %
Jangka Waktu	15%

3.4.2 Pohon Keputusan / Table Keputusan

Suatu koperasi akan memberikan penilaian terhadap keputusan siapa peminjam yang berhak mendapatkan pinjaman, ditunjukkan dalam Tabel 3.3.

Tabel 3.3 Nilai Variabel Logika

Kriteria	Variabel Logika
Jaminan	C1
Pinjaman	C2
Gaji	C3
Jangka Waktu	C4

3.4.3 Pengumpulan *Sampel Data*

Contoh penerapan

Contoh studi kasus penerima pinjaman ada 4 data anggota pemohon ditunjukkan dalam Tabel 3.4.

Tabel 3.4 Data Pemohon

Data Pemohon	Kriteria			
	Jaminan	Pinjaman	Gaji	Jangka Waktu
Mutamakinah	1.200.000	2.400.000	2.500.000	8 bulan
Hj Alimah	2.200.000	1.700.000	2.000.000	15 bulan
Atik	800.000	2.800.000	4.000.000	22 bulan
Hj Fatimah	6.000.000	1.300.000	3.100.000	7 bulan

Berdasarkan data pemohon diatas dapat dibentuk matriks keputusan X yang telah dikonversikan ke dalam nilai yang sudah ditentukan, sebagai berikut:

Tabel 3.5 Rating kecocokan dari setiap alternatif pada setiap kriteria.

Alternatif	Kriteria			
	C ₁	C ₂	C ₃	C ₄
Mutamakinah	0.25	0.5	0.75	1
Hj Alimah	0.5	0.25	0.5	0.75
Atik	0.25	0.5	1	0.5
Hj Fatimah	1	0.25	1	1

Keterangan : C₁ = Jaminan
 C₂ = Pinjaman
 C₃ = Gaji
 C₄ = Jangka Waktu

Kemudian Rating kecocokan dari setiap alternatif pada setiap kriteria dapat dihitung sebagai berikut:

$$V_{11} = 0.25/\text{Max} \{0.25;0.5;0.25;1\} \quad 0.25/1 = 0.25$$

$$V_{12} = 0.5/\text{Max} \{0.25;0.5;0.25;1\} \quad 0.5/1 = 0.5$$

$$V_{13} = 0.25/\text{Max} \{0.25;0.5;0.25;1\} \quad 0.25/1 = 0.25$$

$$V_{14} = 1/\text{Max} \{0.25;0.5;0.25;1\} \quad 1/1 = 1$$

$$V_{21} = 0.5/\text{Max} \{0.5;0.25;0.5;0.25\} \quad 0.5/0.5 = 1$$

$$V_{22} = 0.25/\text{Max} \{0.5;0.25;0.5;0.25\} \quad 0.25/0.5 = 0.5$$

$$V_{23} = 0.5/\text{Max} \{0.5;0.25;0.5;0.25\} \quad 0.5/0.5 = 1$$

$$V_{24} = 0.25/\text{Max} \{0.5;0.25;0.5;0.25\} \quad 0.25/0.5 = 0.5$$

$$V_{31} = 0.75/\text{Max} \{0.75;0.5;1;1\} \quad 0.75/1 = 0.75$$

$$V_{32} = 0.5/\text{Max} \{0.75;0.5;1;1\} \quad 0.5/1 = 0.5$$

$$V_{33} = 1/\text{Max} \{0.75;0.5;1;1\} \quad 1/1 = 1$$

$$V_{34} = 1/\text{Max} \{0.75;0.5;1;1\} \quad 1/1 = 1$$

$$V_{41} = 1/\text{Max} \{1;0.25;1;1\} \quad 1/1 = 1$$

$$V_{42} = 0.75/\text{Max} \{1;0.25;1;1\} \quad 0.75/1 = 0.75$$

$$V_{43} = 0.5/\text{Max} \{1;0.25;1;1\} \quad 0.5/1 = 0.5$$

$$V_{44} = 1/\text{Max} \{1;0.25;1;1\} \quad 1/1 = 1$$

Dari perhitungan di atas dapat di bentuk matriks R

$$R = \begin{bmatrix} 0,25 & 1 & 0,75 & 1 \\ 0,5 & 0,5 & 0,5 & 0,75 \\ 0,25 & 1 & 1 & 0,5 \\ 1 & 0,5 & 1 & 1 \end{bmatrix} \quad W = [0,45 \quad 0,05 \quad 0,35 \quad 0,15]$$

Melakukan proses perangkingan

$$\begin{aligned} V_1 &= (0.45 * 0.25) + (0.05 * 1) + (0.35 * 0.75) + (0.15 * 1) \\ &= 0.1125 + 0.05 + 0.2625 + 0.15 \\ &= 0.575 \end{aligned}$$

$$V_2 = (0.45 * 0.5) + (0.05 * 0.5) + (0.35 * 0.5) + (0.15 * 0.75)$$

$$=0.225+0.025+0.175+0.1125$$

$$=0.5375$$

$$V3 = (0.45 * 0.25) + (0.05 * 1) + (0.35 * 1) + (0.15 * 0.5)$$

$$=0.1125+0.05+0.35+0.075$$

$$=0.5875$$

$$V4 = (0.45 * 1) + (0.05 * 0.5) + (0.35 * 1) + (0.15 * 1)$$

$$=0.45+0.025+0.35+0.15$$

$$=0.975$$

Perhitungan Nilai preferensi

Hasil Perangkingan diperoleh:

$V1 = 0.575$, $V2 = 0.5375$, $V3 = 0.5875$, $V4 = 0.975$. Nilai terbesar ada pada Juminten, dengan demikian alternatif Juminten adalah alternatif yang terpilih sebagai alternatif terbaik. Untuk bobot akhir penerima pinjaman yaitu tergantung pada dana koperasi yang tersedia, sehingga dapat mengetahui berapa jumlah anggota peminjam yang berhak menerima pinjaman tersebut.

3.5 Perancangan Sistem

3.5.1 Flowchart SAW (Simple Additive Weighting)

Flowchart adalah serangkaian bagan-bagan yang menggambarkan alir untuk seleksi penerima pinjaman pada Koperasi Wanita AL-Wardah, ditunjukkan dalam Gambar 3.6.



Gambar 3.6 Flowchart SAW (*Simple Additive Weighting*)

Dari Gambar 3.6, dapat dilihat bahwa dalam penghitungan metode SAW (*Simple Additive Weighting*) membutuhkan inputan berupa nilai setiap alternative pada setiap kriteria dan memberikan inputan berupa nilai bobot pada setiap kriteria.

Membuat matrik keputusan kemudian melakukan normalisasi matriks dengan cara menghitung nilai rating kinerja ternormalisasi (r_{ij}) dari alternatif A_i pada atribut C_j berdasarkan persamaan yang disesuaikan dengan jenis atribut (atribut keuntungan/*benefit* = MAKSIMUM atau atribut biaya/*cost* = MINIMUM). Apabila berupa atribut keuntungan maka nilai crisp (X_{ij}) dari setiap kolom atribut dibagi dengan nilai crisp MAX ($\text{MAX } X_{ij}$) dari tiap kolom, sedangkan untuk atribut biaya, nilai crisp MIN ($\text{MIN } X_{ij}$) dari tiap kolom atribut dibagi dengan nilai crisp (X_{ij}) setiap kolom.

Melakukan proses perankingan untuk setiap alternatif (V_i) dengan cara membagi hasil jumlah perkalian nilai kriteria dengan bobot di bagi dengan jumlah semua hasil dari perkalian untuk hasil preferensi atau perankingan setiap alternatif yang ada.

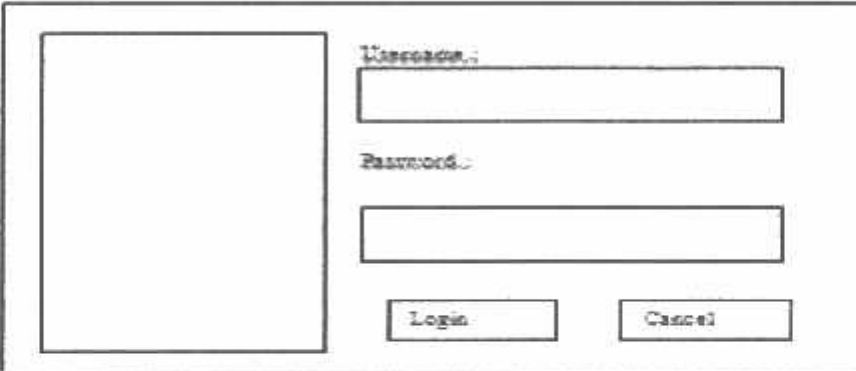
3.5.2 Perancangan antarmuka (*interface*)

Perancangan antarmuka (*interface*) menggambarkan bagaimana perangkat lunak berkomunikasi dengan dirinya sendiri, dengan sistem yang berinteroperasi dengannya dan dengan manusia yang menggunakannya.

3.5.2.1 Perancangan Tampilan

1. Halaman *Login*

Sebelum masuk ke halaman utama aplikasi ini, terlebih dahulu *user* dalam kasus ini adalah administrator, kepala sekolah, staf TU, Bimbingan Konseling harus *login*, Ditunjukkan dalam Gambar 3.7.



The image shows a login form with a large empty rectangular box on the left side. To the right of this box, there are two input fields. The first is labeled "Username:" and the second is labeled "Password:". Below these fields are two buttons: "Login" and "Cancel".

Gambar 3.7. Halaman *Login*

2. Halaman Menu Utama

Halaman menu utama pada aplikasi ini paling penting karena diletakkannya kontrol dalam pengaksesan kehalaman yang lainnya, ditunjukkan dalam Gambar 3.8.

FILE MASTER ▲	Peminjam ▲	Laporan		Log Out
Data Anggota	Data Peminjam			
Data Kriteria	Cari Anggota			
Dana Tersedia	Prose			
User				
M.Makhshushun 811 tzi 2016				

Gambar 3.8. Halaman Menu Utama

3. Halaman Data *User*

Halaman data user adalah halaman yang fungsi untuk mensetting atau membuat *user* akses ke dalam aplikasi, ditunjukkan dalam Gambar 3.9.

FILE MASTER ▲	Peminjam ▲	Laporan		Log Out
Data Anggota	Data Peminjam			
Data Kriteria	Cari Anggota			
Dana Tersedia	Prose			
User				
M.Makhshushun 811 tzi 2016				

Gambar 3.9. Halaman Data *User*

4. Halaman Data Anggota

Halaman data Anggota adalah halaman yang fungsi untuk memproses data anggota, ditunjukkan dalam Gambar 3.10.

Id Anggota	: <input type="text"/>	Pekerjaan	: <input type="text"/>
Nama	: <input type="text"/>	Status Kawin	: <input type="text"/>
Alamat	: <input type="text"/>	Tgl Lahir	: <input type="text"/>
No Telp	: <input type="text"/>	No KTP	: <input type="text"/>
<input type="button" value="Simpan"/> <input type="button" value="Ubah"/> <input type="button" value="Hapus"/> <input type="button" value="Reset"/>			
<div style="border: 1px solid black; padding: 10px; width: fit-content; margin: 0 auto;">List Data Anggota</div>			

Gambar 3.10 Halaman Data Anggota

5. Halaman Pemohon Pinjaman/Kriteria Pinjaman

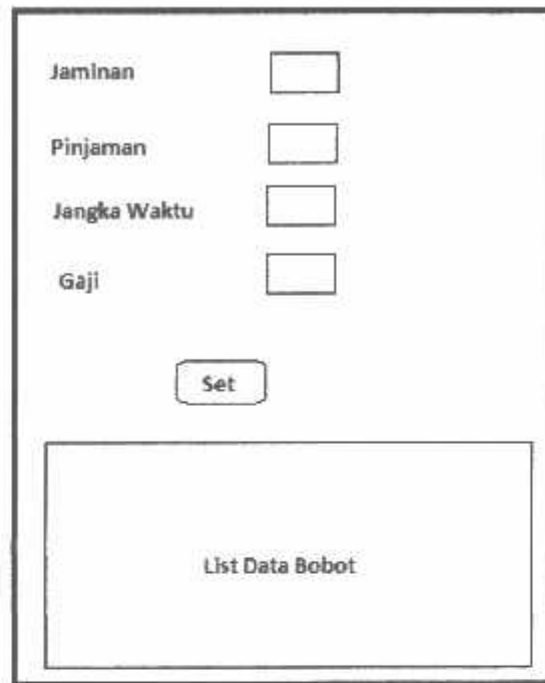
Halaman pemohon pinjaman adalah halaman yang fungsi untuk memproses data kriteria yang pada anggota, ditunjukkan dalam Gambar 3.11.

No Pendaftar	<input type="text"/>	Jaminan	<input type="text"/>
Id Anggota	<input type="text"/>	Pinjaman	<input type="text"/>
Tgl Sekarang	<input type="text"/>	Jangka Waktu	<input type="text"/>
Bungah	<input type="text"/>	Gaji	<input type="text"/>
<input type="button" value="Simpan"/> <input type="button" value="Proses"/>			
<div style="border: 1px solid black; padding: 10px; width: fit-content; margin: 0 auto;">List Data Peminjam</div>			

Gambar 3.11. Halaman Pemohon Pinjaman

6. Bobot Pinjaman

Halaman bobot pinjaman adalah halaman yang berfungsi untuk nilai pada setiap kriteria yang ada pada pinjaman, ditunjukkan dalam Gambar 3.12.



Jaminan

Pinjaman

Jangka Waktu

Gaji

Set

List Data Bobot

Gambar 3.12. Halaman Bobot Pinjaman

7. Halaman Dana

Halaman Dana adalah halaman yang berfungsi untuk nilai pada setiap dana tersedia yang di miliki oleh koperasi untuk peminjam dan limit pinjaman untuk limit berapa besar peminjam mengajukan pinjaman, ditunjukkan dalam Gambar 3.13.



Dana

Limit Pinjaman

SET

Gambar 3.13. Halaman Dana

8. Halaman Hasil Seleksi Pinjaman

Halaman hasil seleksi Pinjaman adalah halaman yang berfungsi untuk mengetahui anggota yang berhak menerima pinjaman dan tidak berhak menerima pinjaman, ditunjukkan dalam Gambar 3.14.



Gambar 3.14 Halaman Hasil Seleksi Pinjaman

BAB IV IMPLEMENTASI DAN PENGUJIAN

4.1 Implementasi Sistem Pendukung Keputusan

4.1.1 Halaman Login

Halaman *login* merupakan halaman yang pertama kali muncul pada sistem bagian *desktop*, dimana pada halaman ini terdapat sebuah *form* yang harus diisi dengan *username* dan *password* tiap *user* bagian agar dapat masuk kedalam sistem. Jika *username* dan *password* benar maka akan muncul tampilan akun yang berbeda untuk tiap-tiap level akses sesuai dengan *username* dan *password* yang telah diisi, ditunjukkan dalam Gambar 4.1.



Gambar 4.1 *Form Login*

4.1.2 Halaman Menu

Halaman menu merupakan tampilan yang digunakan mengakses halaman-halaman lain untuk memudahkan *user*, ditunjukkan dalam Gambar 4.2.



Gambar 4.2 Halaman Menu

4.1.3 Halaman Data Anggota

Halaman data anggota merupakan halaman yang menampilkan dan menginputkan semua data anggota baru, dimana pada halaman ini seorang *user* dapat menambah, merubah, menghapus dan mencari data anggota yang ada, halaman ini berada pada menu file master, ditunjukkan dalam Gambar 4.3.

ID Anggota	Nama Anggota	Alamat	No Tlp/Hp	Pekerjaan	Status Perkawinan	Tanggal Lahir	No Ktp
AWD001	Putri Mekarika	lerongan	08573836538	Pegawa	Menikah	12/17/1977	122432453345
AWD002	Iq almah	lerongan	08573853428	Pegawa	Menikah	12/17/1977	13242424234
AWD003	ask almahidi	lerongan	08121736363	Pegawa	Menikah	12/17/1988	232323
AWD004	Ij Iqmah	lerongan	08123433334	Wiran	Menikah	12/17/1967	12232455
AWD005	usu	lerongan	08583534232	Wiran	Menikah	1/14/1977	5435454645
AWD007	hidayun	lerongan	08917473434	Wiran	Menikah	1/14/1967	11121123
AWD017	ma rief	lerongan	08532323	Wiran	Menikah	11/1/1977	1496754343200
AWD019	Tasah	lerongan	180936342	Pegawa	Menikah	11/20/1977	159000043434
AWD011	usu feah	lerongan	0899328323	Pegawa	Menikah	11/1/1973	122223457
AWD006	julianday	lerongan	087856856	Wiran	Menikah	1/14/1977	543232323
AWD008	khoyah	lerongan	0812344343	Pegawa	Menikah	1/1/1991	1232323
AWD012	nur sponesah	lerongan	087162135	Wiran	Menikah	11/1/1962	2121222222

Gambar 4.3 Halaman Data Anggota

4.1.4 Halaman Dana Tersedia

Halaman dana tersedia merupakan halaman yang berisi kuota penerima pinjaman, dimana pada halaman ini seorang *user* dapat data data dana tersedia dan limit pinjaman, ditunjukkan dalam Gambar 4.4.

Gambar 4.4 Halaman Dana Tersedia

4.1.5 Halaman Data Peminjam

Halaman data peminjam merupakan halaman yang berisi tentang beberapa kriteria, dimana pada halaman ini seorang *user* dapat menyimpan data pemohon pinjaman dan memproses hasilnya, ditunjukkan dalam Gambar 4.5.

No Pendaftaran	Tanggal Sekarang	Id Anggota	Jumlah	Pinjaman	Bunga	Jangka Waktu	Angsuran	Dap
1001	1-13-2016	AW0001	1000000	2000000	0.3	8	500000	2500000
1002	1-13-2016	AW0002	2200000	1700000	0.3	15	130000	2000000
1003	1-13-2016	AW0003	3000000	2800000	0.3	20	157777	4000000
1004	1-13-2016	AW0004	4000000	1300000	0.3	7	22957	3100000
1005	1-16-2016	AW0005	3000000	3000000	0.2	12	118000	4000000
1006	1-26-2016	AW0007	5000000	5000000	0.3	12	500000	12000000
1007	1-26-2016	AW0019	5000000	3500000	0.3	12	350000	12000000

Gambar 4.5 Halaman Data Peminjam

4.1.6 Halaman Proses

Halaman proses merupakan halaman yang berisi hasil perhitungan dari kriteria pada halaman peminjam, dimana pada halaman ini seorang *user* dapat melihat hasil dari proses tersebut, ditunjukkan dalam Gambar 4.6

The screenshot shows a web application window titled 'proses'. It contains five data tables arranged in a grid-like layout:

- PERKALIAN BORDO:** A table with 5 columns: No Perbaikan, Jumlah, Pemas, Jangka Waktu, and Dg. It lists four rows of data.
- INDUKSI:** A table with 5 columns: No Perbaikan, Jumlah, Pemas, Jangka Waktu, and Dg. It lists four rows of data.
- PERBAIKAN:** A table with 5 columns: No Perbaikan, Jumlah, Pemas, Jangka Waktu, and Dg. It lists four rows of data.
- PERKALIAN BORDO (top right):** A table with 5 columns: No Perbaikan, Jumlah, Pemas, Jangka Waktu, and Dg. It lists four rows of data.
- PERBAIKAN BORDO (middle right):** A table with 2 columns: No Perbaikan and Jumlah. It lists four rows of data.
- HASIL (bottom right):** A table with 5 columns: No Perbaikan, Jumlah, Pemas, Jangka Waktu, and Dg. It lists four rows of data.

Gambar 4.6 Halaman Proses

4.1.7 Halaman Cari Anggota

Halaman cari anggota merupakan halaman yang berfungsi untuk mencari data anggota pada halaman data peminjam untuk memilih calon peminjam, dimana pada halaman ini seorang *user* dapat memilih calon peminjam, ditunjukkan dalam Gambar 4.7

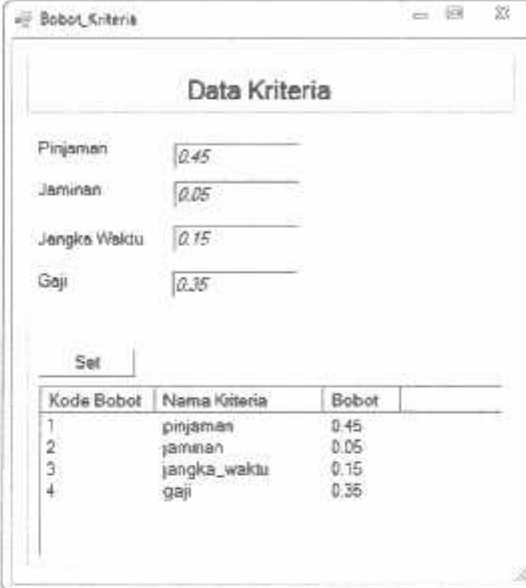
The screenshot shows a web application window titled 'cari_anggota'. It features a table titled 'Data Anggota' with the following columns: Id Anggota, Nama Anggota, Alamat, No TLP/No, Pekerjaan, Status Perkawinan, Tanggal Lahir, and No Rp. Below the table are two search input fields labeled 'Cari' and 'Pilih'.

Id Anggota	Nama Anggota	Alamat	No TLP/No	Pekerjaan	Status Perkawinan	Tanggal Lahir	No Rp
AW0007	mas jah	lemongan	0892233	Wirausaha	Menikah	11/1/1977	1456734343200
AW0011	ulu labun	lemongan	089330623	Pegawai Swasta	Menikah	11/1/1975	3123232457
AW0006	watanobay	lemongan	087656566	Wirausaha	Menikah	1/14/1977	543232323
AW0008	khoyah	lemongan	08123448343	Pegawai Swasta	Menikah	1/1/1991	12323270
AW0012	nur syanah	lemongan	0857162195	Wirausaha	Menikah	11/1/1962	212322222
AW0005	amath	lemongan	0812987943	Pegawai Swasta	Menikah	11/1/1992	32357676332
AW0018	dakiah	lemongan	082137322	Wirausaha	Menikah	11/20/1962	1456734343200
AW0009	latulun radhiah	lemongan	087112313	Pegawai Swasta	Menikah	11/1/1988	3232323
AW0015	rahmah	lemongan	0812343473	Wirausaha	Menikah	11/1/1982	145678900
AW0014	ulah	lemongan	0894565	Wirausaha	Menikah	11/1/1969	3242087634
AW0004	mas jah	lemongan	0892233	Wirausaha	Menikah	11/1/1977	1456734343200

Gambar 4.7 Halaman Cari Anggota

4.1.8 Halaman Bobot Kriteria

Halaman kriteria/bobot merupakan halaman yang fungsi untuk memberi nilai bobot pada setiap kriteria yang ada, dimana pada halaman ini seorang *user* dapat mengubah nilai bobot dari setiap kriteria, ditunjukkan dalam Gambar 4.8



The screenshot shows a window titled "Bobot_Kriteria" with a "Data Kriteria" section. It contains four input fields with the following values:

- Pinjaman: 0.45
- Jaminan: 0.05
- Jangka Waktu: 0.15
- Gaji: 0.35

Below the form is a "Set" button and a table with the following data:

Kode Bobot	Nama Kriteria	Bobot
1	pinjaman	0.45
2	jaminan	0.05
3	jangka_waktu	0.15
4	gaji	0.35

Gambar 4.8. Halaman Bobot Kriteria

4.1.9 Halaman Data User

Halaman Data User merupakan halaman yang berfungsi untuk, dimana pada halaman ini admin membuat username dan password untuk user, ditunjukkan dalam Gambar 4.9.

Kode User	Nama	Username	Password	Hak Akses
7	pegawai	pegawai	pegawai	pegawai
8	kepala koperasi	kepala	kepala	kepala koper...
9	admin	admin	admin	administrator
17	izz	izz	izz	adminstrator

Gambar 4.9. Halaman Data User

4.1.10 Halaman Laporan/Report

Halaman Laporan/Report merupakan halaman yang berfungsi untuk, dimana pada halaman ini melihat laporan peminjam yang di terima dan tidak di terima oleh anggota yang mau meminjam, ditunjukkan dalam Gambar 4.10.

Status	Dibayar	Jumlah	Bulan	Tgl	Anggota	Jml
Melaya	1.000.000	3.000.000	1.20	12	300.000	12.000.000
Tidak	1.000.000	3.000.000	1.20	12	300.000	12.000.000
% Laporan	1.000.000	3.000.000	1.20	1	300.000	1.000.000
aktif	1.000.000	3.000.000	1.20	12	300.000	12.000.000
dit. admin	1.000.000	3.000.000	1.20	22	152.121	4.000.000

Gambar 4.10. Halaman Laporan/Report

4.2 Hasil Pengujian Sistem

Pengujian merupakan tahap uji coba dari sistem untuk menguji setiap fungsi vital agar nantinya dapat diambil kesimpulan apakah sistem berjalan dengan baik sesuai tujuan awal pembuatan.

4.2.1 Pengujian fungsional sistem

Pengujian fungsional sistem dilakukan untuk menguji fitur-fitur yang ada pada sistem pendukung keputusan pemberian pinjaman pada Koperasi Wanita AL-Wardah. Hasil pengujian fungsional sistem ditunjukkan dalam Tabel 4.1.

Tabel 4.1. Hasil Pengujian Fungsional Sistem

No	Fungsi yang diuji	Hasil	
		Berjalan	Tidak berjalan
1	Input data anggota	✓	
2	Input data kriteria	✓	
3	Input dana tersedia	✓	
4	Input data calon peminjam	✓	
5	Cari data anggota calon peminjam	✓	
6	Hasil proses	✓	
7	Report/Laporan	✓	

Dari hasil pengujian fungsional sistem menunjukkan bahwa fitur-fitur yang ada pada sistem dapat berjalan dengan sempurna.

4.2.2 Pengujian *sampel* data

Pada *sampel* data dilakukan dengan menggunakan 4 *sampel* data. Dimana pengujian ini dilakukan untuk mengetahui tingkat keberhasilan dalam proses seleksi pemberian pinjaman. Dan berikut adalah tabel hasil pengujian dari *sampel* data, ditunjukkan pada Tabel 4.2 dibawah ini.

Tabel 4.2. Hasil Pengujian *sampel* data

Jumlah Data	Hasil
4 Data	Berjalan

4.2.3 Pengujian Sistem Operasi

Pada tahapan pengujian aplikasi bagian *desktop* ini dilakukan dengan menggunakan beberapa sistem operasi. Dimana pengujian ini dilakukan untuk mengetahui tingkat keberhasilan dalam bagian *desktop*. Dan berikut adalah tabel hasil pengujian dari pembuatan aplikasi menggunakan beberapa sistem operasi yang ditunjukkan pada Tabel 4.3 dibawah ini.

Tabel 4.3 Hasil pengujian pada beberapa sistem operasi

Proses	Sistem Operasi			
	Windows Xp	Windows 7	Windows 8	Ubuntu
Tampilan aplikasi	y	y	y	X
Proses login	y	y	y	X
Proses pinjaman	Y	y	y	X
Proses Kriteria/Bobot pinjaman	Y	y	y	X
Proses Kuota	Y	y	y	X
Proses Metode SAW	Y	y	y	X
Proses Report	Y	y	y	X
Proses Hasil Seleksi	Y	y	y	X

Keterangan

y : sesuai

x : tidak sesuai

Pada hasil pengujian aplikasi dengan menggunakan 4 sistem operasi yang berbeda, semua tampilan dan fungsi aplikasi dapat berjalan 100% pada 3 sistem operasi yaitu Windows XP, Windows 7, Windows 8.

4.2.4 Pengujian Sistem Perhitungan

Pengujian sistem perhitungan ini dilakukan membandingkan hasil akhir perhitungan manual dengan perhitungan sistem dimana hasil perbandingan tersebut akan mengetahui seberapa besar prosentasi erornya di tunjukkan pada table 4.4.

Tabel 4.4. Hasil Perhitungan Manual dengan Perhitungan Sistem

Anggota Pemohon	Hasil Perhitungan Manual	Hasil Perhitungan Sistem	Error Sistem
Mutamakkinah	0,575	0,575	0%
Hj. Alimah	0,5375	0,5375	0%
Atik Rahmawati	0,5875	0,5875	0%
Hj. Fatimah	0,975	0,975	0%
Jumlah Rata-rata Error			0%

Pada hasil perhitungan diatas diambil dari studi kasus penerima pinjaman ada dengan 4 data calon pemohon pinjaman yang dihitung menggunakan cara manual dan perhitungan sistem yang menghasilkan nilai tertinggi 0%, terendah 0% dan nilai rata-rata error 0% seperti pada tabel 4.4.

BAB V PENUTUP

5.1. Kesimpulan

Berdasarkan hasil pengujian sistem pendukung keputusan seleksi pemberian pinjaman yang dikembangkan pada Koperasi Wanita AL-Wardah, didapatkan beberapa kesimpulan yaitu sebagai berikut :

1. Berdasarkan hasil pengujian fungsional sistem, menunjukkan bahwa fitur-fitur yang ada pada sistem dapat berjalan dengan sempurna dan menggunakan sistem operasi menunjukkan bahwa aplikasi dapat berjalan 100% dalam hal tampilan dan fungsi pada 3 sistem operasi yaitu Windows XP, Windows 7, Windows 8.
2. Berdasarkan hasil pengujian data *sampel* menunjukkan bahwa dengan menggunakan 4 *sampel* data dapat berjalan dengan baik.

5.2 Saran

Berdasarkan kesimpulan di atas, maka terdapat beberapa saran antara lain :

1. Untuk pengembangan sistem pendukung keputusan seleksi penerima pinjaman lebih lanjut ini perlu diterapkan beberapa dengan metode *Multiple Attribute Decision Making* (MADM) dengan operasi yang berbeda seperti *Analytical Hierarchy Process* (AHP), *Technique for Order Preference by Similarity to Ideal Solution* (TOPSIS), dan *Weighted Product* (WP).
2. Untuk pengembangan sistem pendukung keputusan seleksi penerima pinjaman selanjutnya hendaknya dilakukan pengujian efisiensi algoritma metode, pengujian dapat dilakukan dengan membandingkan hasil seleksi pinjaman dengan metode *Multiple Attribute Decision Making* (MADM) yang berbeda seperti metode SAW dengan metode *Weighted Product* (WP).

Daftar Pustaka

- [1] Turban. (2001). *Sistem Pendukung Keputusan*.
 - [2] Kusumadewi, Sri; Hartati, Sri; Harjoko, Agus; Wardoyo, Retantyo. 2006. *Fuzzy Multi-Attribute Decision Making (Fuzzy MADM)*. Yogyakarta. Graha Ilmu.
 - [3] Turban Aronson, 2005 *Fuzzy Multiple Attribute Decision Making (FMADM)*. Yogyakarta : Andi.
 - [4] Yuswanto. 2006. *Pemrograman Dasar Visual Basic.Net 2008*. Prestasi Pustaka Publisher
 - [5] AW, Imam Amirullah. 2005. *SQL Server 2000*. Yogyakarta: Graha Ilmu.
-

LAMPIRAN

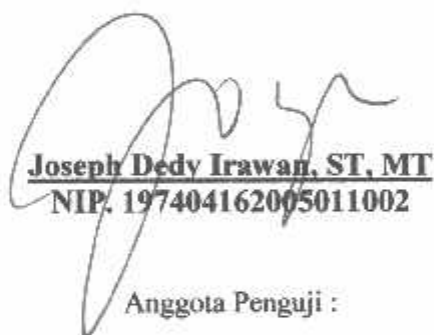
**BERITA ACARA UJIAN SKRIPSI
FAKULTAS TEKNOLOGI INDUSTRI**

NAMA : M.Makhshushun Bil Izzi
NIM : 1218260
JURUSAN : Teknik Informatika S-1
JUDUL : Sistem Pendukung Keputusan Pemberian Pinjaman Pada Koperasi Wanita AL-Wardah Menggunakan Metode Simple Additive Weighting (SAW).

Dipertahankan dihadapan Majelis Penguji Skripsi Jenjang Strata Satu (S-1) pada :
Hari : Sabtu
Tanggal : 16 Januari 2016
Nilai : 74,3 (B+)

Panitia Ujian Skripsi :

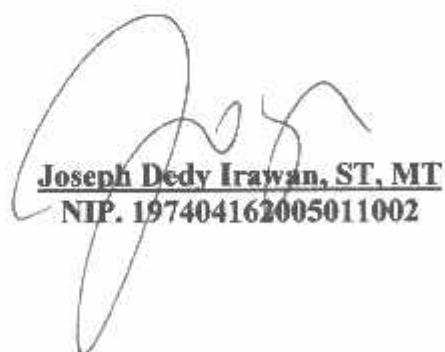
Ketua Majelis Penguji

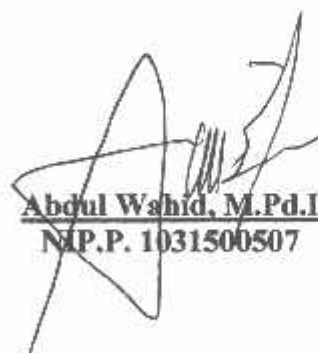

Joseph Dedy Irawan, ST, MT
NIP. 197404162005011002

Anggota Penguji :

Dosen Penguji I

Dosen Penguji II




Joseph Dedy Irawan, ST, MT
NIP. 197404162005011002


Abdul Wahid, M.Pd.I
NIP.P. 1031500507

FORMULIR PERBAIKAN SKRIPSI

Dalam pelaksanaan ujian skripsi jenjang Strata 1 Program Studi Teknik Informatika, maka perlu adanya perbaikan skripsi untuk mahasiswa :


NAMA : M.Makhshushun Bil izzi
NIM : 1218260
JURUSAN : Teknik Informatika S-1
JUDUL : Sistem Pendukung Keputusan Pemberian Pinjaman Pada Koperasi Wanita AL-Wardah Menggunakan Metode Simple Additive Weighting (SAW).

No	Penguji	Tanggal	Uraian	Paraf
1.	Penguji I	16 Januari 2016	1. Rumusan Masalah 2. Cek penulisan laporan 3. BAB III jelaskan: <ul style="list-style-type: none">• Blok diagram system• ERD, DFD 4. Demo program lagi	
2.	Penguji II	16 Januari 2016	1. Di perjelas antara yang di terima dan di tolak	

Dosen Penguji I


Joseph Dedy Irawan, ST., MT.
NIP. 197404162005011002

Dosen Penguji II


Abdul Wahid, M.Pd.I
NIP.P. 1031500507

Dosen Pembimbing I


Karunia Auliasari, ST. M.Eng
NIP.P.1031000426

Dosen Pembimbing II


Moh. Miftakhur Rokhman, S.Kom.M.Kom
NIP.P. 1031500479



PERKUMPULAN PENGELOLA PENDIDIKAN UMUM DAN TEKNOLOGI NASIONAL MALANG
INSTITUT TEKNOLOGI NASIONAL MALANG

FAKULTAS TEKNOLOGI INDUSTRI
FAKULTAS TEKNIK SIPIL DAN PERENCANAAN
PROGRAM PASCASARJANA MAGISTER TEKNIK

PT. BNI (PERSERO) MALANG
BANK NIAGA MALANG

Kampus I : Jl. Bendungan Sigura-gura No. 2 Telp. (0341) 551431 (Hunting), Fax. (0341) 553015 Malang 65145
Kampus II : Jl. Raya Karanglo, Km. 2 Telp. (0341) 417636 Fax. (0341) 417634 Malang

Malang, 23 Oktober 2015

Nomor : ITN-593/LINF/TA/2015

Lampiran : --

Perihal : Bimbingan Skripsi

Kepada : Yth. Bpk/Tbu Karina Auliasari, ST. M.Eng
Dosen Pembina Program Studi Teknik Informatika S-1
Institut Teknologi Nasional
Malang

Dengan Hormat,

Sesuai dengan permohonan dan persetujuan dalam proposal skripsi untuk mahasiswa :

Nama : M. MAKHSHUSHUN BIL IZZI
Nim : 1218260
Prodi : Teknik Informatika S-1
Fakultas : Teknologi Industri

Maka dengan ini pembimbingan kami serahkan sepenuhnya kepada Saudara/i selama waktu 6 (enam) bulan, terhitung mulai tanggal :

23 Oktober 2015 S/D 23 Maret 2016

Sebagai satu syarat untuk menempuh Ujian Akhir Sarjana Teknik, Program Studi Teknik Informatika S-1.

Demikian agar maklum dan atas perhatian serta bantuannya kami sampaikan terima kasih.

Mengetahui
Program Studi Teknik Informatika S-1
Ketua,

Joseph Dedy Irawan, ST., MT.
NIP : 197404162005031602

Form S.4a



PERKUMPULAN PENGELOLA PENDIDIKAN UMUM DAN TEKNOLOGI NASIONAL MALANG
INSTITUT TEKNOLOGI NASIONAL MALANG

FAKULTAS TEKNOLOGI INDUSTRI
FAKULTAS TEKNIK SIPIL DAN PERENCANAAN
PROGRAM PASCASARJANA MAGISTER TEKNIK

PT. BNI (PERSERO) MALANG
BANK NIAGA MALANG

Kampus I : Jl. Bendungan Sigura-gura No. 2 Telp. (0341) 551431 (Hunting), Fax. (0341) 558016 Malang 65145
Kampus II : Jl. Raya Karanglo, Km 2 Telp. (0341) 417636 Fax. (0341) 417634 Malang

Malang, 23 Oktober 2015

Nomor : ITN-593/LINF/TA/2015
Lampiran : ---
Perihal : Bimbingan Skripsi

Kepada : Yth. Bpk/Ibu Moh. M Rokhman, S.Kom, M.Kom
Dosen Pembina Program Studi Teknik Informatika S-1
Institut Teknologi Nasional
Malang

Dengan Hormat,
Sesuai dengan permohonan dan persetujuan dalam proposal skripsi untuk mahasiswa :

Nama : M. MAKHSHUSHUN BIL IZZI
Nim : 1218260
Prodi : Teknik Informatika S-1
Fakultas : Teknologi Industri

Maka dengan ini pembimbingan kami serahkan sepenuhnya kepada Saudara/i selama waktu 6 (enam) bulan, terhitung mulai tanggal :

23 Oktober 2015 S/D 23 Maret 2016

Sebagai satu syarat untuk menempui Ujian Akhir Sarjana Teknik, Program Studi Teknik Informatika S-1.
Demikian agar maklum dan atas perhatian serta bantuannya kami sampaikan terima kasih.

Mengetahui
Program Studi Teknik Informatika S-1
Ketua,

Joseph Dedy Irawan S.P., M.T.
NIP. 19740416200500 002

Form 5-4a

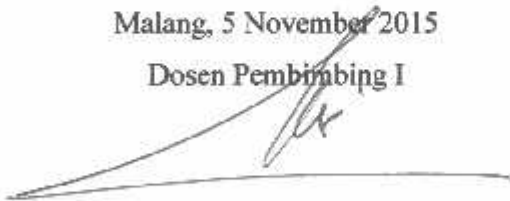
LEMBAR ASISTENSI DOSEN PEMBIMBING I

Nama : M.Makhshushun Bil Izzi
NIM : 12.18.260
Masa Bimbingan : 23 Oktober 2015 s/d 23Maret 2016
Judul Skripsi : Sistem Pendukung Keputusan Pemberian Pinjaman Pada Koperasi Wanita AL-Wardah Menggunakan Metode Simple Additive Weighting (SAW).

No	Tanggal	Uraian	Paraf Pembimbing
1	5-11-2015	Pembahasan Kriteria & Alternatif	Li
2.	19-11-2015	Pembahasan Bab I & II	Li
3.	26-11-2015	Pembahasan Bab III & Coding	Li
4.	27-11-2015	Pembahasan Algoritma Program SAW	Li
5.	3-12-2015	Perubahan Hasil Coding	Li
6.	7-1-2016	Hasil Implementasi	Li
7.	4-12-2015	Pengujian Sistem .	Li
8.	5-1-2016	BAB IV Revisi	Li
9.	8-1-2016	BAB V Revisi	Li
10.	11-1-2016	Laporan Dan Coding Fix	Li





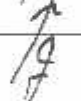



Malang, 5 November 2015

Dosen Pembimbing I


Karina Auliasari, ST. M.Eng
NIP. P. 1031000426

LEMBAR ASISTENSI DOSEN PEMBIMBING 2

Nama : M.Makhshushun Bil Izzi
NIM : 12.18.260
Masa Bimbingan : 23 Oktober 2015 s/d 23Maret 2016
Judul Skripsi : Sistem Pendukung Keputusan Pemberian Pinjaman Pada Koperasi Wanita AL-Wardah Menggunakan Metode Simple Additive Weighting (SAW).

No	Tanggal	Uraian	Paraf Pembimbing
1.	24-11-2015	Perbaikan Latar Belakang, Tinjau Pustaka SAW, Blok Diagram dan Flowchart SAW	
2.	25-11-2015	Perbaikan Laporan	
3.	4-12-2015	BAB IV Revisi	
4.	6-12-2015	BAB IV Fix	
5.	5-1-2016	BAB V Revisi	
6.	8-1-2016	Pengujian Sistem	
7.	9-1-2016	Program Revisi	
8.	12-1-2016	Program Fix	
9.			
10			

Malang, 23 Oktober 2015

Dosen Pembimbing II



Moh. Miftakur/Rokhman, S.Kom. M.Kom
NIP. P. 1031000438



PERKUMPULAN PENGELOLA PENDIDIKAN UMUM DAN TEKNOLOGI NASIONAL MALANG
INSTITUT TEKNOLOGI NASIONAL MALANG

FAKULTAS TEKNOLOGI INDUSTRI
FAKULTAS TEKNIK SIPIL DAN PERENCANAAN
PROGRAM PASCASARJANA MAGISTER TEKNIK

PT BNI (PERSERO) MALANG
BANK NIAGA MALANG

Kampus I : Jl. Bendungan Sigura-gura No. 2 Telp. (0341) 551431 (Hunting), Fax. (0341) 553015 Malang 65145
Kampus II : Jl. Raya Karanglo, Km 2 Telp. (0341) 417636 Fax. (0341) 417634 Malang

Malang, 11 Nopember 2015

Nomor : ITN-11-596/IX.T.INF/2015
Lampiran : -
Perihal : **PENELITIAN SKRIPSI/SURVEI**

Kepada : Yth. Ketua Koperasi Wanita Al-Wardah
Jl. Pemuda RT 03/RW 01 Bangkok Glagah
Lamongan

Dengan hormat,


Bersama dengan surat ini kami mohon kebijaksanaan Bapak/Ibu agar Mahasiswa kami dari **Fakultas Teknologi Industri, Program Studi Teknik Informatika S-1** mohon dapat di ijinakan untuk melakukan pengambilan data untuk pengujian penelitian skripsi di **Koperasi Wanita Al-Wardah**.

Survey akan dilakukan pada : 10 Oktober 2015 s/d 5 November 2015
Adapun mahasiswa tersebut adalah :

M. Makhshushun Bil Izzi NIM : 12.18.260

Setelah melaksanakan survey, hasil dari survey akan digunakan untuk penulisan laporan penelitian/skripsi.
Demikian agar maklum dan atas perhatian serta bantuannya kami ucapkan banyak terima kasih.

Program Studi
Teknik Informatika S-1
Ketua


Joseph Dede Irawan, ST, MT.
NIP. 19740416 200501 1 002

Tembusan Kepada :
1. Arsip



KOPERASI WANITA "AL WARDAH"

Badan Hukum No.518/BH/XVI.10/557/413.111/2010

BANGKOK GLAGAH LAMONGAN

Bangkok , 13 November 2015

Nomor : 027/KOP.WARDAH/XI/2015
Lampiran : -
Hal : IZIN PENELITIAN SKRIPSI/SURVEI
Kepada : Yth. Ketua Program Studi Teknik Informatika ITN Malang

Dengan hormat,

Menindaklanjuti surat bapak nomor : ITN-11-596/IX.T.INF/2015 Tertanggal 11 November 2015 tentang permohonan izin penelitian skripsi/survei di koperasi wanita "Al. Wardah" Desa Bangkok Glagah Lamongan.

Yang dilakukan mahasiswa yang bernama:

Nama : M.MakhshushunBillizzi

Nim : 12.18.260

Kami izinkan, agar dapat digunakan untuk penulisan laporan penelitian skripsi/survei.



MUHAMMAD BILIZZI, S.Pd

Source Code Metode Simple Additive Weighting (SAW)

```
Public Class p1
    Dim max_pinjaman As Integer = 0
    Dim max_jaminan As Integer = 0
    Dim max_gaji As Integer = 0
    Dim max_jangka_waktu As Integer = 0
    Dim bobot_pinjaman As Double = 0
    Dim bobot_jaminan As Double = 0
    Dim bobot_jangka_waktu As Double = 0
    Dim bobot_gaji As Double = 0
    Dim dana_tersedia As Integer

    Private Sub lihatdanatersedia()
        sambung_database()

        Dim query As New SqlClient.SqlCommand("select * from
t_dana where id=1 ", koneksi)
        Dim re As SqlClient.SqlDataReader
        re = query.ExecuteReader
        re.Read()
        dana_tersedia = re!dana_tersedia

    End Sub

    Private Sub lv_pinjaman_SelectedIndexChanged(sender As Object,
e As EventArgs) Handles lv_pinjaman.SelectedIndexChanged

    End Sub
    Private Sub tampilrangking()
        lv_hasil_rangking.Items.Clear()
        sambung_database()

        Dim query As New SqlClient.SqlCommand("select * from
t_hasil inner join t_pinjaman on t_pinjaman.no_pendaftaran =
t_hasil.no_pendaftaran order by jumlah desc ", koneksi)
        Dim re As SqlClient.SqlDataReader
        re = query.ExecuteReader
        Dim query_update As New SqlCommand(String)
        Dim status As String = ""
        Dim i As Integer = 0
        Do While re.Read
            dana_tersedia = dana_tersedia - re!pinjaman

            lv_hasil_rangking.Items.Add(re!no_pendaftaran)
            lv_hasil_rangking.Items(i).SubItems.Add(re!jaminan)
            lv_hasil_rangking.Items(i).SubItems.Add(re!pinjaman)

lv_hasil_rangking.Items(i).SubItems.Add(re!jangka_waktu)
            lv_hasil_rangking.Items(i).SubItems.Add(re!gaji)
            If dana_tersedia > 0 Then
                lv_hasil_rangking.Items(i).BackColor =
Color.AliceBlue
                status = "diterima"
            Else
                status = "ditolak"
            End If
            query_update.Add("UPDATE t_hasil SET status='" &
status & "' WHERE no_pendaftaran ='" & re!no_pendaftaran & "';")
        Loop
    End Sub
End Class
```

```

        i = i + 1
    Loop
    lv_hasil_rangking.Refresh()
    re.Close()
    Dim x As Integer

    Dim query_update_eek As SqlCommand
    For x = 0 To query_update.Count - 1
        Try
            query_update_eek = New
SqlCommand.SqlCommand(query_update(x), koneksi)
            query_update_eek.ExecuteNonQuery()

            Catch ex As Exception
                MsgBox(ex.Message)
            End Try

        Next

    End Sub
    Private Sub tampillistview()
        lv_pinjaman.Items.Clear()
        sambung_database()

        Dim query As New SqlCommand("Select * from
t:pinjaman ", koneksi)
        Dim re As SqlCommand.ExecuteReader
        re = query.ExecuteReader
        Dim i As Integer = 0
        Do While re.Read
            If i = 0 Then
                max_jangka_waktu = re!jangka_waktu
            End If
            lv_pinjaman.Items.Add(re!no_pendaftaran)
            lv_pinjaman.Items(i).SubItems.Add(re!jaminan)
            lv_pinjaman.Items(i).SubItems.Add(re!pinjaman)
            lv_pinjaman.Items(i).SubItems.Add(re!gaji)
            lv_pinjaman.Items(i).SubItems.Add(re!jangka_waktu)

            If max_jaminan < re!jaminan Then
                max_jaminan = re!jaminan
            End If

            If max_pinjaman < re!pinjaman Then
                max_pinjaman = re!pinjaman
            End If

            If max_jangka_waktu > re!jangka_waktu Then
                max_jangka_waktu = re!jangka_waktu
            End If

            If max_gaji < re!gaji Then
                max_gaji = re!gaji
            End If

            i = i + 1
        Loop

```

```

lv_pinjaman.Refresh()
End Sub
Private Sub tampilistviewbobot()
    ListView_bobot.Items.Clear()
    sambung_database()

    Dim query As New SqlClient.SqlCommand("Select * from
t_pinjaman ", koneksi)
    Dim re As SqlClient.SqlDataReader
    re = query.ExecuteReader
    Dim i As Integer = 0
    Do While re.Read
        ListView_bobot.Items.Add(re!no_pendaftaran)

        ListView_bobot.Items(i).SubItems.Add(hitung_bobot_jaminan(re!jamin
an))

        ListView_bobot.Items(i).SubItems.Add(hitung_bobot_pinjaman(re!pinj
aman))

        ListView_bobot.Items(i).SubItems.Add(hitung_bobot_gaji(re!gaji))

        ListView_bobot.Items(i).SubItems.Add(hitung_bobot_jangka_waktu(re!
jangka_waktu))
        i = i + 1
    Loop
    ListView_bobot.Refresh()
End Sub
Private Sub tampilistviewnormalisasi()
    lv_normalisasi.Items.Clear()
    sambung_database()

    Dim query As New SqlClient.SqlCommand ("Select * from
t_pinjaman ", koneksi)
    Dim re As SqlClient.SqlDataReader
    re = query.ExecuteReader
    Dim i As Integer = 0
    Do While re.Read
        lv_normalisasi.Items.Add(re!no_pendaftaran)

        lv_normalisasi.Items(i).SubItems.Add(hitung_bobot_jaminan(re!jamin
an) / hitung_bobot_jaminan(max_jaminan))

        lv_normalisasi.Items(i).SubItems.Add(hitung_bobot_pinjaman(re!pinj
aman) / hitung_bobot_pinjaman(max_pinjaman))

        lv_normalisasi.Items(i).SubItems.Add(hitung_bobot_gaji(re!gaji) /
hitung_bobot_gaji(max_gaji))

        lv_normalisasi.Items(i).SubItems.Add(hitung_bobot_jangka_waktu(re!
jangka_waktu) / hitung_bobot_jangka_waktu(max_jangka_waktu))

        i = i + 1
    Loop
    lv_normalisasi.Refresh()
End Sub
Private Sub tampilistviewnormalisasiikalibobot()
    lv_panggil_bobot.Items.Clear()

```

```

sambung_database()

Dim query As New SqlClient.SqlCommand("Select * from
t_pinjaman ", koneksi)
Dim re As SqlClient.SqlDataReader
re = query.ExecuteReader
Dim i As Integer = 0

Do While re.Read
    lv_panggil_bobot.Items.Add(re!no_pendaftaran)

lv_panggil_bobot.Items(i).SubItems.Add(hitung_bobot_jaminan(re!jaminan) / hitung_bobot_jaminan(max_jaminan) * bobot_jaminan)

lv_panggil_bobot.Items(i).SubItems.Add(hitung_bobot_pinjaman(re!pinjaman) / hitung_bobot_pinjaman(max_pinjaman) * bobot_pinjaman)

lv_panggil_bobot.Items(i).SubItems.Add(hitung_bobot_gaji(re!gaji) / hitung_bobot_gaji(max_gaji) * bobot_gaji)

lv_panggil_bobot.Items(i).SubItems.Add(hitung_bobot_jangka_waktu(re!jangka_waktu) / hitung_bobot_jangka_waktu(max_jangka_waktu) * bobot_jangka_waktu)

    i = i + 1
Loop
lv_panggil_bobot.Refresh()
End Sub

Private Sub tampil ListViewNormalisasiJumlah()
lv_jumlah.Items.Clear()
sambung_database()
Dim query_sql As String = ""
Dim query_reset As New SqlClient.SqlCommand("truncate
table t_hasil ", koneksi)
query_reset.ExecuteNonQuery()
Dim query As New SqlClient.SqlCommand("Select * from
t_pinjaman ", koneksi)
Dim re As SqlClient.SqlDataReader
re = query.ExecuteReader
Dim i As Integer = 0
Dim status As String = ""
Do While re.Read
    lv_jumlah.Items.Add(re!no_pendaftaran)

lv_jumlah.Items(i).SubItems.Add((hitung_bobot_jaminan(re!jaminan) / hitung_bobot_jaminan(max_jaminan) * bobot_jaminan) +
(hitung_bobot_pinjaman(re!pinjaman) / hitung_bobot_pinjaman(max_pinjaman) * bobot_pinjaman) +
(hitung_bobot_jangka_waktu(re!jangka_waktu) / hitung_bobot_jangka_waktu(max_jangka_waktu) * bobot_jangka_waktu) +
(hitung_bobot_gaji(re!gaji) / hitung_bobot_gaji(max_gaji) * bobot_gaji))

    query_sql = query_sql + "insert into t_hasil values("
    & re!no_pendaftaran & ", " &
Replace((hitung_bobot_jaminan(re!jaminan) / hitung_bobot_jaminan(max_jaminan) * bobot_jaminan) +
(hitung_bobot_pinjaman(re!pinjaman) / hitung_bobot_pinjaman(max_pinjaman) * bobot_pinjaman) +

```

```

(hitung_bobot_jangka_waktu(re!jangka_waktu) /
hitung_bobot_jangka_waktu(max_jangka_waktu) * bobot_jangka_waktu)
+ (hitung_bobot_gaji(re!gaji) / hitung_bobot_gaji(max_gaji) *
bobot_gaji), ",", "-") & "%", "%": "%
        i = i + 1
    Loop
    re.Close()

    Dim queer_input As New SqlConnection(query_sql,
koneksi)
    queer_input.ExecuteNonQuery()

    lv_jumlah.Refresh()
End Sub

Private Sub GroupBox1_Enter(sender As Object, e As EventArgs)
End Sub

Private Sub proses_Load(sender As Object, e As EventArgs)
Handles MyBase.Load
    lihatdanatersedia()
    tampillistview()
    tampillistviewbobot()
    tampillistviewnormalisasi()
    panggilbobot()
    tampillistviewnormalisasikalibobot()
    tampillistviewnormalisasijumlah()
    tampilrangking()

End Sub
Function hitung_bobot_pinjaman(input As Integer) As Double
    Dim bobot_pinjaman As Double = 0.1
    Dim pinjaman As Integer = input
    If pinjaman > 4000000 Then
        bobot_pinjaman = 1
    ElseIf pinjaman > 3000000 Then
        bobot_pinjaman = 0.75
    ElseIf pinjaman > 2000000 Then
        bobot_pinjaman = 0.5
    ElseIf pinjaman > 100000 Then
        bobot_pinjaman = 0.25
    End If
    Return bobot_pinjaman
End Function

Function hitung_bobot_jaminan(input As Integer) As Double
    Dim bobot_jaminan As Double = 0.1
    Dim jaminan As Integer = input
    If jaminan > 4000000 Then
        bobot_jaminan = 1
    ElseIf jaminan > 3000000 Then
        bobot_jaminan = 0.75
    ElseIf jaminan > 2000000 Then
        bobot_jaminan = 0.5
    ElseIf jaminan > 100000 Then
        bobot_jaminan = 0.25
    End If

```



```

Return bobot_jaminan
End Function

Function hitung_bobot_gaji(input As Integer) As Double
Dim bobot_gaji As Double = 0.1
Dim gaji As Integer = input
If gaji > 3000000 Then
    bobot_gaji = 1
ElseIf gaji > 2000000 Then
    bobot_gaji = 0.75
ElseIf gaji > 1000000 Then
    bobot_gaji = 0.5
ElseIf gaji > 100000 Then
    bobot_gaji = 0.25
End If
Return bobot_gaji
End Function

Function hitung_bobot_jangka_waktu(input As Integer) As Double
Dim bobot_jangka_waktu As Double = 0.1
Dim jangka_waktu As Integer = input
If jangka_waktu > 24 Then
    bobot_jangka_waktu = 0.25
ElseIf jangka_waktu > 18 Then
    bobot_jangka_waktu = 0.5
ElseIf jangka_waktu > 12 Then
    bobot_jangka_waktu = 0.75
Else
    bobot_jangka_waktu = 1
End If
Return bobot_jangka_waktu
End Function

Private Sub panggilbobot()
    sambung_database()
    Dim bobot_array(4) As Double
    Dim cmd As New SqlCommand("Select * from t_kriteria")
    Dim reader As SqlCommand.ExecuteReader
    cmd.Connection = koneksi
    reader = cmd.ExecuteReader
    Dim i As Integer = 0
    Do While reader.Read
        bobot_array(i) = reader!bobot_kriteria
        i = i + 1
    Loop
    bobot_jaminan = bobot_array(0)
    bobot_pinjaman = bobot_array(1)
    bobot_jangka_waktu = bobot_array(2)
    bobot_gaji = bobot_array(3)

End Sub

End Class

```

Source Code Bobot Pinjaman

```
Imports System.Data.SqlClient
Imports System.Windows.Forms
Public Class Form2
    Dim bangsuran As Double
    Dim bjaminan As Double
    Dim btabungan As Double
    Dim blama_anggota As Double
    Dim bgaji As Double
    Private Sub Form2_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
        tampil()
    End Sub
    Private Sub tampil()
        lv_kriteria.Items.Clear()
        sambung_database()
        Dim cmd As New SqlCommand("Select * from t_kriteria")
        Dim reader As SqlDataReader
        cmd.Connection = koneksi
        reader = cmd.ExecuteReader
        Dim i As Integer = 0
        Do While reader.Read
            lv_kriteria.Items.Add(reader!kode_bobot)
            lv_kriteria.Items(i).SubItems.Add(reader!nama_kriteria)
            lv_kriteria.Items(i).SubItems.Add(reader!bobot_kriteria)
            i = i + 1
        Loop
        lv_kriteria.Refresh()
    End Sub
    Private Sub jumlah()
    End Sub
    Private Sub tb_pinjaman_KeyPress(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyPressEventArgs)
        If Not ((e.KeyChar >= "0" And e.KeyChar <= "9") Or e.KeyChar = "." Or e.KeyChar = vbBack) Then
            e.Handled = True
        End If
    End Sub
    Private Sub tb_jaminan_KeyPress(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyPressEventArgs)
        If Not ((e.KeyChar >= "0" And e.KeyChar <= "9") Or e.KeyChar = "." Or e.KeyChar = vbBack) Then
            e.Handled = True
        End If
    End Sub
End Class
```

```

End Sub

Private Sub tb_jangka_waktu_KeyPress(ByVal sender As Object,
ByVal e As System.Windows.Forms.KeyPressEventArgs)
    If Not ((e.KeyChar >= "0" And e.KeyChar <= "9") Or
e.KeyChar = "." Or e.KeyChar = vbBack) Then
        e.Handled = True

    End If
End Sub

Private Sub tb_gaji_KeyPress(ByVal sender As Object, ByVal e
As System.Windows.Forms.KeyPressEventArgs)
    If Not ((e.KeyChar >= "0" And e.KeyChar <= "9") Or
e.KeyChar = "." Or e.KeyChar = vbBack) Then
        e.Handled = True

    End If
End Sub

Private Sub btn_set_Click(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles btn_set.Click

    If tb_pinjaman.Text = "" Or tb_jaminan.Text = "" Or
tb_jangka_waktu.Text = "" Or tb_gaji.Text = "" Then
        MsgBox("Data harus di isi", MsgBoxStyle.Critical,
"Peringatan")
    Exit Sub
    End If

    Dim a, b, c, d As Double
    a = tb_pinjaman.Text
    b = tb_jaminan.Text
    c = tb_jangka_waktu.Text
    d = tb_gaji.Text
    MsgBox(a)

    Dim i As Decimal
    i = a + b + c + d

    If i = 1 Then
        updae()
        tampil()

        Form2_Load(sender, e)

    Exit Sub
    Else

        MsgBox("Total data harus 100", MsgBoxStyle.Critical,
"Peringatan")
    Exit Sub

    End If
End Sub
Private Sub updae()

```

```

sambung_database()
Try
    Dim query1 As New SqlClient.SqlCommand("Update
lv_kriteria set bobot_kriteria = '" & Replace(tb_pinjaman.Text,
",", ".") & "' where kode_bobot = '1'")
    query1.Connection = koneksi
    query1.ExecuteNonQuery()

    Dim query2 As New SqlClient.SqlCommand("Update
lv_kriteria set bobot_kriteria = '" & Replace(tb_jaminan.Text, ",",
".") & "' where kode_bobot = '2'")
    query2.Connection = koneksi
    query2.ExecuteNonQuery()

    Dim query3 As New SqlClient.SqlCommand("Update
lv_kriteria set bobot_kriteria = '" & Replace(tb_jangka_waktu.Text,
",", ".") & "' where kode_bobot = '3'")
    query3.Connection = koneksi
    query3.ExecuteNonQuery()

    Dim query4 As New SqlClient.SqlCommand("Update
lv_kriteria set bobot_kriteria = '" & Replace(tb_gaji.Text, ",",
".") & "' where kode_bobot = '4'")
    query4.Connection = koneksi
    query4.ExecuteNonQuery()

    MsgBox("Berhasil", MsgBoxStyle.Information,
"Berhasil")

    Catch ex As Exception
        MsgBox("Proses Gagal", MsgBoxStyle.Critical,
"Peringatan")
    End Try

End Sub

Private Sub ListView1_Click(DyVal sender As System.Object,
ByVal e As System.EventArgs) Handles lv_kriteria.Click

End Sub

Private Sub ListView1_SelectedIndexChanged(ByVal sender As
System.Object, ByVal e As System.EventArgs) Handles
lv_kriteria.SelectedIndexChanged
    tb_pinjaman.Text = lv_kriteria.Items(0).SubItems(2).Text
    tb_jaminan.Text = lv_kriteria.Items(1).SubItems(2).Text
    tb_jangka_waktu.Text =
lv_kriteria.Items(2).SubItems(2).Text
    tb_gaji.Text = lv_kriteria.Items(3).SubItems(2).Text
End Sub

```

```

Private Sub GroupBox1_Enter(sender As Object, e As EventArgs)
Handles GroupBox1.Enter

    End Sub
End Class

```

Source Code Cari Anggota

```

Public Class Form1
Private Sub tampillistview()
lv_anggota.Items.Clear()
sambung_database()

Dim query As New SqlClient.SqlCommand("select * from
t_anggota where id_anggota not in (select id_anggota from
t_pinjaman)", koneksi)
Dim re As SqlClient.SqlDataReader
re = query.ExecuteReader
Dim i As Integer = 0
Do While re.Read

lv_anggota.Items.Add(re!id_anggota)
lv_anggota.Items(i).SubItems.Add(re!nama_anggota)
lv_anggota.Items(i).SubItems.Add(re!alamat)
lv_anggota.Items(i).SubItems.Add(re!no_tlp)
lv_anggota.Items(i).SubItems.Add(re!pekerjaan)
lv_anggota.Items(i).SubItems.Add(re!status_perkawinan)
lv_anggota.Items(i).SubItems.Add(re!tgl_lahir)
lv_anggota.Items(i).SubItems.Add(re!no_ktp)
i = i + 1

Loop
lv_anggota.Refresh()
End Sub
Private Sub cari_anggota_Load(sender As Object, e As
EventArgs) Handles MyBase.Load
tampillistview()
End Sub

Private Sub lv_anggota_SelectedIndexChanged(sender As Object,
e As EventArgs) Handles lv_anggota.SelectedIndexChanged

End Sub

Private Sub tb_cari_TextChanged(sender As Object, e As
EventArgs) Handles tb_cari.TextChanged
lv_anggota.Items.Clear()
sambung_database()

Dim query As New SqlClient.SqlCommand("select * from
t_anggota where id_anggota not in (select id_anggota from
t_pinjaman) and (id_anggota like '1' & tb_cari.Text & '2' or
nama_anggota like '1' & tb_cari.Text & '2' or pekerjaan like '3' &
tb_cari.Text & '4')", koneksi)
Dim re As SqlClient.SqlDataReader
re = query.ExecuteReader

```

```

Dim i As Integer = 0
Do While re.Read
    lv_anggota.Items.Add(re!id_anggota)
    lv_anggota.Items(i).SubItems.Add(re!nama_anggota)
    lv_anggota.Items(i).SubItems.Add(re!alamat)
    lv_anggota.Items(i).SubItems.Add(re!no_tlp)
    lv_anggota.Items(i).SubItems.Add(re!pekerjaan)
    lv_anggota.Items(i).SubItems.Add(re!status_perkawinan)
    lv_anggota.Items(i).SubItems.Add(re!tgl_lahir)
    lv_anggota.Items(i).SubItems.Add(re!no_ktp)
    'lv_siswa.Items(i).BackColor = Color.GreenYellow
    i = i + 1
Loop
lv_anggota.Refresh()
End Sub

Private Sub lv_anggota_MouseClick(sender As Object, e As
EventArgs) Handles lv_anggota.MouseClick
    lb_id_anggota.Text =
lv_anggota.SelectedItems(0).SubItems(0).Text
    lb_nama_anggota.Text =
lv_anggota.SelectedItems(0).SubItems(1).Text

End Sub

Private Sub pilih_Click(sender As Object, e As EventArgs)
Handles pilih.Click
    Pendaftaran_Pinjaman.id_anggota.Text = lb_id_anggota.Text
    Me.Dispose()
    Pendaftaran_Pinjaman.Show()
End Sub

Private Sub GroupBox1_Enter(sender As Object, e As EventArgs)
Handles GroupBox1.Enter

End Sub

Private Sub lb_id_anggota_Click(sender As Object, e As
EventArgs) Handles lb_id_anggota.Click

End Sub
End Class

```

Source Code Dana Tersedia

```

Public Class DanaTersedia

    Private Sub DanaTersedia_Load(sender As Object, e As
EventArgs) Handles MyBase.Load
        tampil()
        tampil2()

    End Sub

    Private Sub bt_dana_Click(sender As Object, e As EventArgs)
Handles bt_dana.Click

```

```

    sambung_database()

    Dim query As New SqlClient.SqlCommand("update t_dana set
dana tersedia='" & dana_tersedia.Text & "' where id=1 ", koneksi)
    query.ExecuteNonQuery()
    Dim query2 As New SqlClient.SqlCommand("update t_limit
set limit_pinjaman='" & limit_pinjaman.Text & "' where id=1 ",
koneksi)
    query2.ExecuteNonQuery()
    tampil()
    tampil2()
    MsgBox("data telah tersimpan")
End Sub
Sub tampil()
    sambung_database()

    Dim query As New SqlClient.SqlCommand("select * from
t_dana where id=1 ", koneksi)
    Dim re As SqlClient.SqlDataReader
    re = query.ExecuteReader
    re.Read()
    dana_tersedia.Text = re!dana_tersedia
End Sub

Sub tampil2()
    sambung_database()

    Dim query As New SqlClient.SqlCommand("select * from
t_limit where id=1 ", koneksi)
    Dim re As SqlClient.SqlDataReader
    re = query.ExecuteReader
    re.Read()
    limit_pinjaman.Text = re!limit_pinjaman
End Sub

Private Sub limit_pinjaman_TextChanged(sender As Object, e As
EventArgs) Handles limit_pinjaman.TextChanged

    End Sub
End Class

```

Source Code Data User

```

Public Class FormDataUser

    Private Sub simpan_Click(sender As Object, e As EventArgs)
Handles simpan.Click
        tambah()
        tampillistview()
    End Sub
    Private Sub edit()
        If kode_user.Text = "" Then
            MsgBox("Pilih Data Dulu", MsgBoxStyle.Critical, "")
        Else
            Try
                sambung_database()
            Catch ex As Exception
                MsgBox(ex.Message, MsgBoxStyle.Critical, "Error")
            End Try
        End If
    End Sub
End Class

```

```

        Dim ubah As New SqlClient.SqlCommand("update
t_user set nama = '" & nama.Text & "', username = '" &
username.Text & "', pass = '" & pass.Text & "', hak akses = '" &
hak_akses.Text & "' where kode_user = '" & kode_user.Text & "' ")
        ubah.Connection = koneksi
        ubah.ExecuteNonQuery()
        MsgBox("Data Berhasil Di Ubah ",
MsgBoxStyle.Information, "Informasi")
        tampilistview()
    Catch ex As Exception
        MsgBox("Gagal " & ex.Message)
    End Try
End If
End Sub

Private Sub tampilistview()
    lv_anggota.Items.Clear()
    sambung_database()

    Dim query As New SqlClient.SqlCommand("Select * from
t_user ", koneksi)
    Dim re As SqlClient.SqlDataReader
    re = query.ExecuteReader
    Dim i As Integer = 0
    Do While re.Read
        lv_anggota.Items.Add(re!kode_user)
        lv_anggota.Items(i).SubItems.Add(re!nama)
        lv_anggota.Items(i).SubItems.Add(re!username)
        lv_anggota.Items(i).SubItems.Add(re!pass)
        lv_anggota.Items(i).SubItems.Add(re!hak_akses)
        i = i + 1
    Loop
    lv_anggota.Refresh()
End Sub

Private Sub tambah()
    If kode_user.Text = "" Or nama.Text = "" Or e.Text = "" Or
pass.Text = "" Or hak_akses.Text = "" Then
        MsgBox("Data Harus Lengkap", MsgBoxStyle.Critical,
"Peringatan")
    Else
        Try
            sambung_database()
            Dim query As New SqlClient.SqlCommand("insert into
t_user(nama,username,pass,hak_akses) values ('" & nama.Text &
"', '" & username.Text & "', '" & pass.Text & "', '" & hak_akses.Text
& "')")

            query.Connection = koneksi
            query.ExecuteNonQuery()
            MsgBox("Tersimpan Sam", MsgBoxStyle.Information,
"informasi")
        Catch ex As Exception
            MsgBox("error-" & ex.Message)
        End Try
    End If
End If

```



```

End Sub

Private Sub delete()
    If kode_user.Text = "" Then
        MsgBox("Pilih Data Sulu", MessageBoxButtons.Critical, "")
    Else
        sambung_database()
        If MessageBox.Show("Apa Anda Yakin Data Di hapus '" &
kode_user.Text & "'?", " Hapus Data", MessageBoxButtons.YesNo) =
"" & Windows.Forms.DialogResult.Yes Then
            Try
                Dim hapusdata As New
SqlConnection("delete from t_user where kode_user = '" &
kode_user.Text & "'")
                hapusdata.Connection = koneksi
                hapusdata.ExecuteNonQuery()
                MsgBox("Data Berhasil Dihapus",
MessageBoxStyle.Information, "Information")
                tampillistview()
            Catch ex As Exception
                MsgBox("Filed" & MessageBoxButtons.Critical,
"Caagal")
            End Try
        End If
    End If
End Sub

Private Sub ubah_Click(sender As Object, e As EventArgs)
Handles ubah.Click

End Sub

Private Sub hapus_Click(sender As Object, e As EventArgs)
Handles hapus.Click
    delete()

End Sub

Private Sub lv_anggota_MouseClick(sender As Object, e As
EventArgs) Handles lv_anggota.MouseClick
    kode_user.Text =
lv_anggota.SelectedItems(0).SubItems(0).Text
    nama.Text = lv_anggota.SelectedItems(0).SubItems(1).Text
    username.Text =
lv_anggota.SelectedItems(0).SubItems(2).Text
    pass.Text = lv_anggota.SelectedItems(0).SubItems(3).Text
    hak_akses.Text =
lv_anggota.SelectedItems(0).SubItems(4).Text
End Sub

Private Sub Label2_Click(sender As Object, e As EventArgs)
Handles Label2.Click

End Sub

```

```

Private Sub FormDataUser_Load(sender As Object, e As
EventArgs) Handles MyBase.Load
    tampillistview()

End Sub

Private Sub kode_user_Click(sender As Object, e As EventArgs)
Handles kode_user.Click

End Sub
End Class

```

Source Code Form Laporan

```

Public Class FormLaporan

Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
Dim rep As New laporan
Dim dt As DataTable
Dim dtt As DataSetPenerimaanTableAdapters.DataTable1TableAdapter = New
DataSetPenerimaanTableAdapters.DataTable1TableAdapter
dt = dtt.GetData(cb_penerimaan.Text)
rep.SetDataSource(dt)
Me.CrystalReportViewer1.ReportSource = rep

End Sub

Private Sub FormLaporan_Load(sender As Object, e As EventArgs) Handles MyBase.Load

End Sub
End Class

```

Source Code Form Menu Utama

```

Public Class FormMenuUtama

Shared Property Data1User As Object

Private Sub ToolStripSplitButton1_ButtonClick(sender As
Object, e As EventArgs) Handles DataMaster.ButtonClick

End Sub

Private Sub DataAnggota_Click(sender As Object, e As
EventArgs) Handles DataAnggota.Click
Halaman Anggota.ShowDialog()

End Sub

Private Sub DataKriteria_Click(sender As Object, e As
EventArgs) Handles DataKriteria.Click
Halaman Kriteria.ShowDialog()

End Sub

```

```

Private Sub LaporanToolStripMenuItem_Click(sender As Object, e
As EventArgs) Handles data_pinjaman.Click
    LaporanPinjaman.ShowDialog()

End Sub

Private Sub ProsesToolStripMenuItem_Click(sender As Object, e
As EventArgs) Handles ProsesToolStripMenuItem.Click
    Proses.ShowDialog()

End Sub

Private Sub CariDataAnggotaToolStripMenuItem_Click(sender As
Object, e As EventArgs) Handles CariDataAnggota.Click
    CariDataAnggota.ShowDialog()

End Sub

Private Sub LaporanPinjamanToolStripMenuItem_Click(sender As
Object, e As EventArgs)

End Sub

Private Sub ToolStripButton1_Click(sender As Object, e As
EventArgs) Handles keluar.Click
    If (MsgBox("Yakin Keluar?", vbYesNo, "Keluar") = vbYes)
Then
        Close()

    End If
End Sub

Private Sub DanaTersediaToolStripMenuItem_Click(sender As
Object, e As EventArgs) Handles
DanaTersediaToolStripMenuItem.Click
    DanaTersedia.ShowDialog()

End Sub

Private Sub DataLaporan_ButtonClick(sender As Object, e As
EventArgs) Handles DataLaporan.ButtonClick
    DataLaporan.ShowDialog()

End Sub

Private Sub Label1_Click(sender As Object, e As EventArgs)
Handles Label1.Click

End Sub

Private Sub DataUserToolStripMenuItem_Click(sender As Object,
e As EventArgs) Handles DataUserToolStripMenuItem.Click
    Form1.ShowDialog()

End Sub

```

```

Private Sub FormMenuUtama_Load(sender As Object, e As
EventArgs) Handles MyBase.Load

End Sub
End Class

```

Source Code Form Login

```

Public Class FormLogin

Private Sub OK_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles OK.Click
login()

End Sub

Private Sub login()
sambung_database()
Dim pass, hak, nama, kode_user As String

pass = tb_pass.Text

Dim query As New SqlConnection("select * from
tbl_user where username='" & tb_user.Text & "' and pass = '" & pass
& "'")
query.Connection = koneksi
Dim rea As SqlDataReader
rea = query.ExecuteReader

If rea.Read = True Then
hak = rea!hak_akses
nama = rea!nama
kode_user = rea!kode_user

.....

FormMenuUtama.data_pinjaman.Visible = False
FormMenuUtama.DataKriteria.Visible = False
FormMenuUtama.DanaTersediaToolStripMenuItem.Visible =
False
FormMenuUtama.CariDataAnggota.Visible = False
FormMenuUtama.ProsesToolStripMenuItem.Visible = False
FormMenuUtama.DataLaporan.Visible = False
FormMenuUtama.DataAnggota.Visible = False
FormMenuUtama.DataLaporan.Visible = False
FormMenuUtama.DataMaster.Visible = False
FormMenuUtama.DataPemohon.Visible = False
FormMenuUtama.DataUserToolStripMenuItem.Visible =
False

.....

If hak = "administrator" Then
FormLogin.Show()
FormMenuUtama.data_pinjaman.Visible = True
FormMenuUtama.DataKriteria.Visible = True

```

```

FormMenuUtama.DanaTersediaToolStripMenuItem.Visible = True
FormMenuUtama.CariDataAnggota.Visible = True
FormMenuUtama.ProsesToolStripMenuItem.Visible =
True
FormMenuUtama.DataLaporan.Visible = True
FormMenuUtama.DataUserToolStripMenuItem.Visible =
True
FormMenuUtama.DataMaster.Visible = True
FormMenuUtama.DataPemohon.Visible = True

Elseif hak = "kepala_koperasi" Then
    FormMenuUtama.Show()
    FormMenuUtama.DataLaporan.Visible = True

Elseif hak = "pegawai" Then
    FormMenuUtama.Show()
    FormMenuUtama.DataAnggota.Visible = True
    FormMenuUtama.DataMaster.Visible = True

End If

Class
    MsgBox("Username dan password tidak valid ",
    MessageBoxButtons.Critical, "peringatan")

End If
End Sub
'End Sub
Public Function zeekomp(ByVal textToHash As String) As String
    Dim MD5 As New
System.Security.Cryptography.MD5CryptoServiceProvider()
    Dim Bytes() As Byte =
MD5.ComputeHash(System.Text.Encoding.ASCII.GetBytes(textToHash))
    Dim s As String = Nothing
    For Each by As Byte In Bytes
        s += by.ToString("x2")
    Next
    Return s
End Function

Private Sub Cancel_Click(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles Cancel.Click
    Me.Close()
End Sub

Private Sub HalamanLogIn_Load(sender As Object, e As
EventArgs) Handles MyBase.Load

End Sub
End Class

```

Source Code Form Login

```
Public Class Halaman_Sambutan

    Sub bersih()
        nama_anggota.Clear()
        alamat.Clear()
        no_tlp.Clear()
        no_ktp.Clear()
    End Sub

    Private Sub simpan_Click(sender As Object, e As EventArgs)
        Handles simpan.Click
            anggota()
            tampillistview()

            kodeotomatis()
        End Sub
    Private Sub anggota()
        If id_anggota.Text = "" Or nama_anggota.Text = "" Or
alamat.Text = "" Or no_tlp.Text = "" Or pekerjaan.Text = "" Or
status_kawin.Text = "" Or tgl_lahir.Text = "" Or no_ktp.Text = ""
Then

            MsgBox("Data Harus Lengkap", MsgBoxStyle.Critical,
"Peringatan")
        Else
            Try
                sambung_database()
                Dim query As New SqlClient.SqlCommand("insert into
t_anggota values ('" & id_anggota.Text & "', '" & nama_anggota.Text
& "', '" & alamat.Text & "', '" & no_tlp.Text & "', '" &
pekerjaan.Text & "', '" & status_kawin.Text & "', '" &
tgl_lahir.Text & "', '" & no_ktp.Text & "')")
                query.Connection = koneksi
                query.ExecuteNonQuery()
                MsgBox("Tersimpan Sam", MsgBoxStyle.Information,
"informasi")
            Catch ex As Exception
                MsgBox("error=" + ex.Message)
            End Try
        End If
    End Sub
    Private Sub tampillistview()
        lv_anggota.Items.Clear()
        sambung_database()

        Dim query As New SqlClient.SqlCommand("Select * from
t_anggota ", koneksi)
        Dim re As SqlClient.SqlDataReader
        re = query.ExecuteReader
        Dim i As Integer = 0
        Do While re.Read
            lv_anggota.Items.Add(re!id_anggota)
            lv_anggota.Items(i).SubItems.Add(re!nama_anggota)
        End While
    End Sub
End Class
```

```

lv_anggota.Items(i).SubItems.Add(re!alamat)
lv_anggota.Items(i).SubItems.Add(re!no_tlp)
lv_anggota.Items(i).SubItems.Add(re!pekerjaan)
lv_anggota.Items(i).SubItems.Add(re!status_perkawinan)
lv_anggota.Items(i).SubItems.Add(re!tgl_lahir)
lv_anggota.Items(i).SubItems.Add(re!no_ktp)
i = i + 1
>>>
lv_anggota.Refresh()
End Sub

Private Sub Halaman_Pendataan_Load(sender As Object, e As EventArgs) Handles MyBase.Load
    tampillistview()
    kodeotomatis()

End Sub

Private Sub edit()
    If id_anggota.Text = "" Then
        MsgBox("Fillin Data Dulu", MessageBoxButtons.Critical, "")
    Else
        Try
            sambung_database()
            Dim ubah As New SqlClient.SqlCommand("update t_anggota set nama_anggota = '" & nama_anggota.Text & "', alamat = '" & alamat.Text & "', no_tlp = '" & no_tlp.Text & "', pekerjaan = '" & pekerjaan.Text & "', status_perkawinan = '" & status_kawin.Text & "', tgl_lahir = '" & tgl_lahir.Text & "', no_ktp = '" & no_ktp.Text & "' where id_anggota = '" & id_anggota.Text & "'")

            ubah.Connection = koneksi
            ubah.ExecuteNonQuery()
            MsgBox("Data Berhasil Di Ubah ", MessageBoxButtons.Information, "informasi")
            tampillistview()
        Catch ex As Exception
            MsgBox("Gagal" & ex.Message)
        End Try
    End If
End Sub

Private Sub delete()
    If id_anggota.Text = "" Then
        MsgBox("Fillin Data Dulu", MessageBoxButtons.Critical, "")
    Else
        sambung_database()
        If MessageBox.Show("Apa Anda Yakin Data Di Hapus '" & id_anggota.Text & "'?", "Hapus Data", MessageBoxButtons.YesNo) = "" & Windows.Forms.DialogResult.Yes Then
            Try
                Dim hapusdata As New
                SqlClient.SqlCommand("delete from t_anggota where id_anggota = '" & id_anggota.Text & "'")
                hapusdata.Connection = koneksi
                hapusdata.ExecuteNonQuery()
            End Try
        End If
    End If
End Sub

```

```

        MsgBox("Data Berhasil Dihapus",
MsgBoxStyle.Information, "Information")
        tampilListView()
        Catch ex As Exception
            MsgBox("Filed" & ex.Message, Critical,
"Gagal")
        End Try

    End If
End If
End Sub

Private Sub lv_anggota_MouseDoubleClick(sender As Object, e As
MouseEventArgs) Handles lv_anggota.MouseDoubleClick
    id_anggota.Text =
lv_anggota.SelectedItems(0).SubItems(0).Text
    nama_anggota.Text =
lv_anggota.SelectedItems(0).SubItems(1).Text
    alamat.Text = lv_anggota.SelectedItems(0).SubItems(2).Text
    no_tlp.Text = lv_anggota.SelectedItems(0).SubItems(3).Text
    pekerjaan.Text =
lv_anggota.SelectedItems(0).SubItems(4).Text
    status_kawin.Text =
lv_anggota.SelectedItems(0).SubItems(5).Text
    no_ktp.Text = lv_anggota.SelectedItems(0).SubItems(6).Text

End Sub

Private Sub ubah_Click(sender As Object, e As EventArgs)
Handles ubah.Click
    edit()

End Sub

Private Sub hapus_Click(sender As Object, e As EventArgs)
Handles hapus.Click
    delete()

End Sub

Private Sub tb_cari_TextChanged(ByVal sender As
System.Object, ByVal e As System.EventArgs) Handles
tb_cari.TextChanged
    lv_anggota.Items.Clear()
    sambung_database()

    Dim query As New SqlCommand("Select * from
lv_anggota where id_anggota like '" & tb_cari.Text & "%' or
nama_anggota like '" & tb_cari.Text & "%' or pekerjaan like '" &
tb_cari.Text & "%'", koneksi)
    Dim re As SqlDataReader
    re = query.ExecuteReader
    Dim i As Integer = 0
    Do While re.Read
        lv_anggota.Items.Add(re!id_anggota)
        lv_anggota.Items(i).SubItems.Add(re!nama_anggota)
    
```



```

lv_anggota.Items(i).SubItems.Add(re!alamat)
lv_anggota.Items(i).SubItems.Add(re!no_tlp)
lv_anggota.Items(i).SubItems.Add(re!pekerjaan)
lv_anggota.Items(i).SubItems.Add(re!status_perkawinan)
lv_anggota.Items(i).SubItems.Add(re!tgl_lahir)
lv_anggota.Items(i).SubItems.Add(re!no_ktp)

i = i + 1
Loop
lv_anggota.Refresh()
End Sub
Private Sub kodeotomatis()
sambung_database()
Dim query As New SqlConnection("select * from
t_anggota order by id_anggota desc")
query.Connection = koneksi
Dim r As SqlConnectionReader
r = query.ExecuteReader

Dim s As String = ""
If r.Read() Then
If r.IsDBNull(0) Then
s = 0
Else
s = r.GetString(0)

End If
End If

s = Mid(s, 4, 4)
s = Val(s) + 1
While Len(s) < 4
s = "0" & s
End While
id_anggota.Text = "AW" & s
id_anggota.Refresh()
End Sub
Private Sub GroupBox2_Enter(sender As Object, e As EventArgs)
Handles GroupBox2.Enter

End Sub

Private Sub id_anggota_Click(sender As Object, e As EventArgs)
Handles id_anggota.Click

End Sub

Private Sub reset_Click(sender As Object, e As EventArgs)
Try
Call bersih()
Catch ex As Exception
MsgBox("berhasil")

End Try
End Sub

Private Sub lv_anggota_SelectedIndexChanged(sender As Object,
e As EventArgs) Handles lv_anggota.SelectedIndexChanged

```

```

End Sub

Private Sub Button1_Click(sender As Object, e As EventArgs)
Handles Button1.Click
    nama_anggota.Clear()
    alamat.Clear()
    no_tlp.Clear()
    no_ktp.Clear()

End Sub

Private Sub GroupBox3_Enter(sender As Object, e As EventArgs)
Handles GroupBox3.Enter

End Sub
End Class

```

Halaman Pendataan Pinjaman

```

Public Class Form1
    Dim limit_pinjaman As Integer
    Private Sub lihatlimit()
        sambung_database()

        Dim query As New SqlClient.SqlCommand("select * from
t_limit where id=1", koneksi)
        Dim re As SqlClient.SqlDataReader
        re = query.ExecuteReader
        re.Read()
        limit_pinjaman = re.limit_pinjaman
    End Sub
    Private Sub Label10_Click(sender As Object, e As EventArgs)
Handles Label10.Click

End Sub

Private Sub save()
    If no_daftar.Text = "" Or tgl_sekarang.Text = "" Or
id_anggota.Text = "" Or pinjaman.Text = "" Or jaminan.Text = "" Or
jangka_waktu.Text = "" Or tb_bunga.Text = "" Or angsuran.Text = ""
Or gaji.Text = "" Then

        MsgBox("Data harus lengkap", MsgBoxStyle.Critical,
"Peringatan")
    Else
        If Integer.Parse(gaji.Text) >
Integer.Parse(angsuran.Text) Then
            If Integer.Parse(pinjaman.Text) < limit_pinjaman
Then
                Dim query As New SqlClient.SqlCommand("insert
into t_pinjaman values ('" & no_daftar.Text & "','" &
tgl_sekarang.Text & "','" & id_anggota.Text & "','" & jaminan.Text
& "','" & pinjaman.Text & "','" & Replace(tb_bunga.Text, ",", ".")
& "','" & jangka_waktu.Text & "','" & angsuran.Text & "','" &
gaji.Text & "')")
            End If
        End If
    End If
End Sub

```

```

        Try
            sambung_database()
            query.Connection = koneksi
            query.ExecuteNonQuery()
            MsgBox("Tersimpan Sam",
MsgBoxType.Information, "informasi")
        Catch ex As Exception
            MsgBox("error=" + ex.Message)
        End Try
    Else
        MsgBox("Melebihi Ketentuan")
    End If

Else
    MsgBox("Tidak dapat meminjam gaji terlalu rendah")
End If

End If

End Sub

Private Sub tampilistview()
    lv_pinjaman.Items.Clear()
    sambung_database()

    Dim query As New SqlClient.SqlCommand("Select * from
lv_pinjaman ", koneksi)
    Dim re As SqlClient.SqlDataReader
    re = query.ExecuteReader
    Dim i As Integer = 0
    Do While re.Read
        lv_pinjaman.Items.Add(re!no_pendaftaran)
        lv_pinjaman.Items(i).SubItems.Add(re!tgl_sekarang)
        lv_pinjaman.Items(i).SubItems.Add(re!id_anggota)
        lv_pinjaman.Items(i).SubItems.Add(re!jaminan)
        lv_pinjaman.Items(i).SubItems.Add(re!pinjaman)
        lv_pinjaman.Items(i).SubItems.Add(re!bunga)
        lv_pinjaman.Items(i).SubItems.Add(re!jangka_waktu)
        lv_pinjaman.Items(i).SubItems.Add(re!angsuran)
        lv_pinjaman.Items(i).SubItems.Add(re!gaji)
        i = i + 1
    Loop
    lv_pinjaman.Refresh()
End Sub

Private Sub Pendataan_Pinjaman_Load(sender As Object, e As
EventArgs) Handles MyBase.Load
    lihatlimit()
    tampilistview()
    kodeotomatis()

End Sub

Private Sub edit()
    If no_daftar.Text = "" Then
        MsgBox("Pilih Data Dulu", MsgBoxType.Critical, "")
    Else
        Try
            sambung_database()

```

```

        Dim ubah As New SqlCommand("update
        t_pinjaman set pinjaman = '" & pinjaman.Text & "', jaminan = '" &
        jaminan.Text & "', jangka_waktu = '" & jangka_waktu.Text & "', bunga
        = '" & tb_bunga.Text & "', angsuran = '" & angsuran.Text & "', gaji
        = '" & gaji.Text & "' where no_pendaftaran = '" & no_daftar.Text &
        "' ")

        ubah.Connection = koneksi
        ubah.ExecuteNonQuery()
        MsgBox("Data Berhasil Di Ubah ",
        MsgBoxStyle.Information, "informasi")
        tampilListView()
    Catch ex As Exception
        MsgBox("Gagal" & ex.Message)
    End Try
End If
End Sub

Private Sub delete()
    If no_daftar.Text = "" Then
        MsgBox("Pilih Data Dulu", MsgBoxStyle.Critical, "")
    Else
        sambung_database()
        If MessageBox.Show("Apa Anda Yakin Data Di hapus '" &
        no_daftar.Text & "'?", "hapus Data", MessageBoxButtons.YesNo) =
        "" & Windows.Forms.DialogResult.Yes Then
            Try
                Dim hapusdata As New
                SqlCommand("delete from t_pinjaman where no_pendaftaran
                = '" & no_daftar.Text & "'")
                hapusdata.Connection = koneksi
                hapusdata.ExecuteNonQuery()
                MsgBox("Data Berhasil Dihapus",
                MsgBoxStyle.Information, "informasi")
                tampilListView()
            Catch ex As Exception
                MsgBox("Filed" & ex.Message,
                MsgBoxStyle.Critical,
                "Gagal")
            End Try
        End If
    End If
End Sub

Private Sub TextBox2_TextChanged(sender As Object, e As EventArgs)
End Sub

Private Sub GroupBox1_Enter(sender As Object, e As EventArgs)
Handles GroupBox1.Enter
End Sub

Private Sub lv_pinjaman_MouseClick(sender As Object, e As MouseEventArgs)
End Sub

```

```

Private Sub GroupBox2_Enter(sender As Object, e As EventArgs)
Handles GroupBox2.Enter

End Sub

Private Sub simpan_Click(sender As Object, e As EventArgs)
Handles simpan.Click
save()
tampilistview()

kodeotomatis()
End Sub

Private Sub Button2_Click(sender As Object, e As EventArgs)
edit()

End Sub

Private Sub lv_pinjaman_MouseDoubleClick(sender As Object, e
As MouseEventArgs) Handles lv_pinjaman.MouseDoubleClick
no_daftar.Text =
lv_pinjaman.SelectedItems(0).SubItems(0).Text
tgl_sekarang.Text =
lv_pinjaman.SelectedItems(0).SubItems(1).Text
id_anggota.Text =
lv_pinjaman.SelectedItems(0).SubItems(2).Text
jaminan.Text =
lv_pinjaman.SelectedItems(0).SubItems(3).Text
pinjaman.Text =
lv_pinjaman.SelectedItems(0).SubItems(4).Text
tb_bunga.Text =
lv_pinjaman.SelectedItems(0).SubItems(5).Text
jangka_waktu.Text =
lv_pinjaman.SelectedItems(0).SubItems(6).Text
angsuran.Text =
lv_pinjaman.SelectedItems(0).SubItems(7).Text
gaji.Text = lv_pinjaman.SelectedItems(0).SubItems(8).Text

End Sub

Private Sub jangka_waktu_TextChanged(sender As Object, e As
EventArgs) Handles jangka_waktu.TextChanged
hitung_angsuran()

End Sub

Private Sub Button3_Click(sender As Object, e As EventArgs)
Handles hapus.Click
delete()

End Sub

Private Sub tb_cari_TextChanged(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles tb_cari.TextChanged
lv_pinjaman.Items.Clear()
sambung_database()

Dim query As New SqlConnection("Select * from
t_pinjaman where no_pendaftaran like '%" & tb_cari.Text & "%'" or
no_pendaftaran like '%" & tb_cari.Text & "%'", koneksi)

```

```

Dim re As SqlConnection.SqlDataReader
re = query.ExecuteReader
Dim i As Integer = 0
Do While re.Read
    lv_pinjaman.Items.Add(re!no_pendaftaran)
    lv_pinjaman.Items(i).SubItems.Add(re!tgl_sekarang)
    lv_pinjaman.Items(i).SubItems.Add(re!id_anggota)
    lv_pinjaman.Items(i).SubItems.Add(re!pinjaman)
    lv_pinjaman.Items(i).SubItems.Add(re!jaminan)
    lv_pinjaman.Items(i).SubItems.Add(re!jangka_waktu)
    lv_pinjaman.Items(i).SubItems.Add(re!bunga)
    lv_pinjaman.Items(i).SubItems.Add(re!angsuran)
    lv_pinjaman.Items(i).SubItems.Add(re!gaji)
    'lv_siswa.Items(i).BackColor = Color.GreenYellow
    i = i + 1
Loop
lv_pinjaman.Refresh()
End Sub

Private Sub lv_pinjaman_SelectedIndexChanged(sender As Object,
e As EventArgs) Handles lv_pinjaman.SelectedIndexChanged

End Sub
Private Sub hitung_angsuran()
Try
    Dim hasil_angsuran, pinjaman_, bunga, waktu As Double
    bunga = Double.Parse(tb_bunga.Text)
    waktu = Double.Parse(jangka_waktu.Text)
    pinjaman_ = Double.Parse(pinjaman.Text)
    hasil_angsuran = (pinjaman_ + ((bunga / 100) *
pinjaman_)) / waktu

    angsuran.Text = Math.Round(hasil_angsuran)
Catch
    angsuran.Text = ""
End Try

End Sub
Private Sub pinjaman_TextChanged(sender As Object, e As
EventArgs) Handles pinjaman.TextChanged
    hitung_angsuran()
End Sub

Private Sub angsuran_TextChanged(sender As Object, e As
EventArgs) Handles angsuran.TextChanged

End Sub
Private Sub gaji_TextChanged(sender As Object, e As EventArgs)
Handles gaji.TextChanged

End Sub

Private Sub tb_bunga_TextChanged(sender As Object, e As
EventArgs) Handles tb_bunga.TextChanged
    hitung_angsuran()
End Sub

Private Sub kodeotomatis()

```

```

sambung_database()
Dim query As New SqlClient.SqlCommand("select * from
t_pinjaman order by no_pendaftaran desc")
query.Connection = koneksi
Dim r As SqlClient.SqlDataReader
r = query.ExecuteReader

Dim s As String = ""
If r.Read() Then
    If r.IsDBNull(0) Then
        s = 0
    Else
        s = r.GetString(0)

        End If
End If

s = Mid(s, 4, 4)
s = Val(s) + 1
While Len(s) < 4
    s = "0" & s
End While
no_daftar.Text = "" & s
no_daftar.Refresh()
End Sub

Private Sub Button5_Click(sender As Object, e As EventArgs)
Handles Button5.Click
    pendaftaran.ShowDialog()

End Sub

Private Sub b_proses_Click(sender As Object, e As EventArgs)
Handles b_proses.Click
    proses.ShowDialog()

End Sub

Private Sub no_daftar_Click(sender As Object, e As EventArgs)
Handles no_daftar.Click

End Sub

Private Sub reset_Click(sender As Object, e As EventArgs)
Handles reset.Click
    pinjaman.Clear()
    id_anggota.Clear()
    jaminan.Clear()
    jangka_waktu.Clear()
    angsuran.Clear()
    gaji.Clear()

End Sub

Private Sub Button1_Click(sender As Object, e As EventArgs)
Handles bt_edit.Click
    edit()

```