**IMPLEMENTASI JARINGAN SARAF TIRUAN UNTUK MENDETEKSI KECENDERUNGAN GANGGUAN MAKAN PADA REMAJA MENGGUNAKAN METODE BACKPROPAGATION**

Diajukan untuk memenuhi salah satu syarat mencapai derajat Sarjana S-1 Program Studi Informatika



Disusun oleh:

**BUDIARTI**

**5130411242**

**PROGRAM STUDI INFORMATIKA**

**FAKULTAS TEKNOLOGI INFORMASI DAN ELEKTRO**

**UNIVERSITAS TEKNOLOGI YOGYAKARTA**

**2018**

**HALAMAN PENGESAHAN**

**IMPLEMENTASI JARINGAN SARAF TIRUAN UNTUK MENDETEKSI KECENDERUNGAN GANGGUAN MAKAN PADA REMAJA MENGGUNAKAN METODE BACKPROPAGATION**

**PROYEK TUGAS AKHIR**

Disusun oleh

BUDIARTI

5130411242

Telah dipertanggungjawabkan di dalam Sidang Proyek Tugas Akhir  
pada tanggal 22 Februari 2018

Tim Penguji:

Donny Avianto, S.T., M.T.

Ketua

Muhammad Fachrie, S.T., M.Cs

Anggota

Dr.Arief Hermawan, S.T., M.T.  
Anggota

Tugas akhir ini telah diterima sebagai salah satu syarat untuk mencapai derajat Sarjana S-1 Program Studi Informatika

Yogyakarta,

Ketua Program Studi Informatika

Fakultas Teknologi Informasi dan Elektro,

Universitas Teknologi Yogyakarta

Dr. Enny Itje Sela, S.Si., M.Kom

**LEMBAR PERNYATAAN**

Saya yang bertanda tangan di bawah ini:

Nama : Budiarti

NIM : 5130411242

Program Studi : Informatika

Menyatakan bahwa Proyek Tugas Akhir yang berjudul:

**IMPLEMENTASI JARINGAN SARAF TIRUAN UNTUK MENDETEKSI KECENDERUNGAN GANGGUAN MAKAN PADA REMAJA MENGGUNAKAN METODE BACKPROPAGATION**

merupakan karya ilmiah asli saya dan belum pernah dipublikasikan oleh orang lain, kecuali yang tertulis sebagai acuan dalam naskah ini dan disebutkan dalam daftar pustaka. Apabila di kemudian hari, karya saya disinyalir bukan merupakan karya asli saya, maka saya bersedia menerima konsekuensi apa yang diberikan Program Studi Informatika Fakultas Teknologi Informasi dan Elektro Universitas Teknologi Yogyakarta kepada saya.

Demikian surat pernyataan ini saya buat dengan sebenarnya.

Yogyakarta, 22 Februari 2018  
Yang menyatakan,

Budiarti

**MOTTO**

*“Jika kamu telah memulai maka selesaikanlah sampai tuntas dan tanpa meninggalkan beban untuk orang lain”*

*“Selama Tuhan masih memberimu kesempatan untuk bernafas dan membuka mata setelah kamu tertidur, maka yakinlah bahwa kamu masih diberi kesempatan untuk terus berjuang”*

*“Bahkan jika semua orang meragukanmu dan meninggalkanmu ingatlah kamu Tuhan yang memberi hidup”*

**PERSEMBAHAN**

Dengan menyebut nama Allah SWT dan Nabi besar Muhammad SAW, Alhamdulillah penulis ucapkan atas segala kuasa dan rahmat-Nya pada akhirnya laporan ini dapat terlesaikan dengan baik. Laporan Tugas Akhir ini penulis persembahkan sebagai wujud terimakasih yang mendalam kepada :

1. Allah SWT atas rahmat dan hidayah-Nya.
2. Baginda Rasulullah Muhammad SAW sebagai pembimbing dan suri tauladan dalam hidupuku, yang syaf’at-Nya dinantikan pada hari akhir nanti.
3. Ibuku dan Bapaku tercinta yang telah mendukung, mendoakan dan memberikan motivasi yang besar, serta kasih sayang yang teramat besar yang tak mungkin bisa saya balas dengan apapun.
4. Bapak Dr. Arief Hermawan, S.T., M.T yang sudah berkenan untuk membantu dan menjadi dosen pembimbing hingga terselesaikannya laporan ini.
5. Ibu Respati Andamari yang berkenan menjadi narasumber dalam penelitian ini
6. Teman-teman seperjuanganku Dewi, Yessi, Fitria, Mas Zak dan lainnya, terimakasih untuk bantuannya selama ini.
7. Keluarga Pusat Studi Robotika UTY, Bu Ro dan teman-teman terima kasih telah menjadi keluarga kedua dan selalu memberi semangat untuk menyelesaikan TA ini. Terimakasih untuk suka duka yang terlewati selama ini.
8. Teman-teman Teknik Informatika angkatan 2013. Khususnya TI-D 2013 terimakasih untuk 4 tahun ini.
9. Pihak yang tidak bisa saya sebut satu persatu, terimakasih atas segala kontriusinya dalam membatu saya menyelesaikan TA ini.

**INTISARI**

**IMPLEMENTASI JARINGAN SARAF TIRUAN UNTUK MENDETEKSI KECENDERUNGAN GANGGUAN MAKAN PADA REMAJA MENGGUNAKAN METODE BACKPROPAGATION**

Gangguan makan adalah orang-orang yang memiliki tingkat kepercayaan diri yang rendah, dimana mereka merasa tidak sebanding dengan orang lain. Gangguan makan atau eating disorders tidak sepenuhnya berhubungan dengan makanan, pola makan, dan berat badan saja, tetapi mengenai perasaan dan ekspresi diri atau merupakan sebuah penyakit mental, yang mana penyakit ini merupakan ancaman serius karena penderita akan makan dalam jumlah banyak atau justru makan dalam jumlah sedikit. Sebagian besar gangguan makan disebabkan oleh iteraksi kompleks dari faktor genetik, biologis, perilaku, psikologis, dan sosial. Dengan sistem identifikasi ganggua makan ini diharapkan dapat membantu dan menambah pengetahuan masyarakat umum dalam menentukan jenis gangguan makan yang diderita. Data yang diambil adalah nilai dari jawaban kuisioner yang telah dijawab oleh responden.

Penelitian ini dilakukan melalui beberapa tahap diantaranya pengumpulan data melalui observasi, wawancara, dan studi pustaka. Dari data tersebut, maka data akan di analisis untuk mendapatkan gambaran dalam menentukan desain yang akan dikembangkan pada sistem. Selanjutnya dilakukan pengujian terhadap sistem untuk mengetahui fungsi dari setiap komponen sistem.

Hasil dari pengembangan sistem berupa prediksi ganggguan makan terpilih menggunakan bahasa pemrograman Netbeans IDE 8.0.2 dan MySQL sebagai databasenya, yang mengacu pada data pelatihan menggunakan metode *backpropagation*. Dari hasil pengujian didapatkan nilai keakuratan 100% dari data terlatih dan 60% dari data tak terlatih.

**Kata kunci : *Gangguna Makan, Backpropagation, Netbeans IDE***

**ABSTRACT**

**IMLEMATION OF NERURAL NETWORK FOR DETECTING TREN EATING DISORDER IN ADLOSCENT USING BACKPROPAGATION METHOD**

Eating disorders are people who have low self-esteem, which they feel is not comparable with others. Eating disorders or eating disorders are not entirely related to diet, diet, and weight alone, but about feelings and self-expression or a mental illness, which is a serious threat because the patient will eat in large quantities or eat in quantities a little. Most eating disorders are caused by complex interactions of genetic, biological, behavioral, psychological, and social factors. With the identification system of ganggua makan is expected to help and increase the knowledge of the general public in determining the type of eating disorder suffered. The data taken is the value of the answer questionnaire that has been answered by the respondent.

This research is done through several stages including data collection through observation, interview, and literature study. From these data, then the data will be analyzed to get a picture in determining the design to be developed on the system. Then testing the system to determine the function of each component of the system.

The result of the development of the system in the form of prediction of eating gangguan was chosen using the programming language Netbeans IDE 8.0.2 and MySQL as its database, which refers to the training data using backpropagation method. From the test results obtained 100% accuracy value of trained data and 60% of untrained data.

**Keywords: *Eating Disorder, Backpropagation, Netbeans IDE***

**KATA PENGANTAR**

Puji syukur Alhamdulillah penulis ucapkan kepada Allah SWT, atas rahmat dan hidayah–Nya, sehingga penulis dapat menyusun Laporan Proyek Tugas Akhir dengan judul “**IMPLEMENTASI JARINGAN SARAF TIRUAN UNTUK MENDETEKSI KECENDERUNGAN GANGGUAN MAKAN PADA REMAJA MENGGUNAKAN METODE BACKPROPAGATION**”.

Penyusunan Laporan Tugas Akhir diajukan sebagai salah satu syarat untuk memperoleh gelar sarjana pada Program Studi Teknik Informatika Fakultas Teknologi Informasi dan Elektro Universitas Teknologi Yogyakarta.

Dalam penulisan Laporan Proyek Tugas Akhir ini dapat terlaksana dengan baik tentunya tidak terlepas dari beberapa pihak yang terlibat. Untuk itu penulis ucapkan terima kasih yang sedalam-dalamnya kepada:

1. Dr. Bambang Moertono Setiawan, MM., Akt., CA, selaku Rektor Universitas Teknologi Yogyakarta yang selalu memberikan motivasi.
2. Dr. Erik Iman Heri Ujianto, M.Kom, selaku Dekan Fakultas Teknologi Informasi dan Elektro yang telah memberikan dukungan.
3. Dr. Enny Itje Sela, S.Si., M.Kom, Selaku Kaprodi Teknik Informatika yang selalu memberi nasihat.
4. Rodhiyah Mardhiyah, S.Kom., M.Kom, selaku Dosen Wali kelas Teknik Informatika D yang selalu memberikan pegarahan.
5. Dr. Arief Hermawan, S.T., M.T, selaku Dosen pembimbing Proyek Tugas Akhir yang selalu bersabar dalam membimbing.
6. Ibu Dra. Sri Respati Adamari, M.Si., Psi selaku narasumber dalam penelitian ini
7. Kedua Orang Tua, Keluarga dan Sahabat yang telah memberi dukungan secara materiil, moral dan spiritual.
8. Rekan-rekan Teknik Informatika Universitas Teknologi Yogyakarta kelas TI D angkatan 2013.
9. Semua pihak yang tak dapat disebutkan satu persatu, baik langsung maupun tidak langsung membantu dalam penulisan Tugas Akhir ini.

Akhir kata, penulis menyadari bahwa sepenuhnya akan terbatasnya pengetahuan penyusun, sehingga tidak menutup kemungkinan jika ada kesalahan serta kekurangan dalam penyusunan Laporan Tugas Akhir, untuk itu sumbang saran dan masukan dari pembaca sangat diharapkan sebagai bahan pelajaran berharga dimasa yang akan datang. Penulis berharap semoga Tugas Akhir ini bermanfaat bagi pembaca dan semua pihak yang berkepentingan dengan Tugas Akhir ini.

Yogyakarta, Februari 2018

Penulis

Budiarti

**DAFTAR ISI**

**Halaman Judul** i

**Halaman Pengesahan** ii

**Halaman Pernyataan** iii

**Intisari** iv

***Abstract*** *v*

**Persembahan** vi

**Motto** vii

**Kata Pengantar** viii

**Daftar Isi**  i

**Daftar Gambar** iii

**Daftara Tabel v**

**BAB I PENDAHULUAN**

I.1 Latar Belakang 1

I.2 Rumusan Masalah 2

I.3 Batasan Masalah 2

I.4 Tujuan Penelitian 3

I.5 Manfaat Penelitian 3

I.6 Sistematika Penulisan 3

**BAB II LANDASAN TEORI**

II.1 Kajian Penelitian 5

II.2 Kajian Teori 7

II.2.1 Jaringan Saraf Tiruan 7

II.2.2 Perambatan Galat Mundur 8

II.2.2.1 Algoritma *Backpropagation* 8

II.2.3 Deteksi 13

II.2.4 Remaja 13

II.2.5 Gangguan Makan 13

II.2.5.1 Definisi Gangguan Makan 13

II.2.5.2 Anoreksia Nervosa 14

II.2.5.3 Bulimia Nervosa 15

II.2.5.4 Binge Eating Disorder 16

II.2.6 Data Flow Diagram (DFD) 17

II.2.7 Entity Relationship Diagram (ERD) 18

II.2.8 Flowchart 20

**BAB III METODE PENELITIAN**

III.1 Object Penelitian 23

III.2 Analisis Sistem 23

III.2.1 Observasi 23

III.2.2 Wawancara 24

III.2.3 Angket 24

III.2.3 Studi Kepustakaan 26

III.3 Perancangan Sistem 26

III.3 Implementasi Sistem 27

III.3 Perangkat Pendukung Penelitian 28

**BAB IV ANALISIS RANCANGAN SISTEM**

IV.1 Analisis Sistem 29

IV.2 Perancangan Sistem Implementasi JST untuk DGM 30

IV.2.1 Arsitektur Jaringan 30

IV.2.2 Diagram Alir Data (DAD) 31

IV.2.2.1 Diagram Konteks 31

IV.2.2.2 Diagram Jenjang (level 0) 32

IV.2.2.3 Diagram Rinci Level 1 33

IV.2.2.4 Diagram Rinci level 2 proses 2 33

IV.2.2.5 Diagram Rinci level 2 proses 3 34

IV.2.2 Flowchart Sistem 35

IV.2.2.1 Flowchart Pelatihan Data 35

IV.2.2.2 Flowchart Pengujian Data 37

IV.2.2.3 Flowchart Prediksi Diagnosa Penyakit 38

IV.2.3 Entity Relationship Diagram (ERD) 49

IV.2.4 Struktur Database 40

IV.2.5 Perancangan Antar Muka 45

IV.2.5.1 Perancangan Form Login 45

IV.2.5.2 Perancangan Form Menu Utama 45

IV.2.5.3 Perancangan Form User 47

IV.2.5.4 Perancangan Form Kategori 48

IV.2.5.5 Perancangan Form Soal 49

IV.2.5.6 Perancangan Form Data Gangguan 49

IV.2.5.7 Perancangan Input Data Latih 50

IV.2.5.8 Perancangan Form Pelatihan 51

IV.2.5.9 Perancangan Form Pengujian 51

IV.2.5.10 Icon Logout 52

**BAB V IMPLEMENTASI DAN PEMBAHASAN SISTEM**

V.1 Implementasi Sistem 53

V.2 Pembahasan Sistem 53

V.2.1 Halaman Login 54

V.2.2 Halaman Utama 55

V.2.3 Halaman Data User 57

V.2.4 Halaman Data Kategori 59

V.2.5 Halaman Data Pertanyaan 60

V.2.6 Halaman Data Gangguan 62

V.2.7 Halaman Data Input Data 63

V.2.8 Halaman Data Input Gangguan 65

V.2.9 Halaman Pelatihan 67

V.2.10 Halaman Pengujian 69

V.2.11 Halaman Prediksi 82 V.2.12 Halaman Simulasi Hitugan Manual 84

V.2.13 Halaman Simulasi Pelatihan 89

**BAB VI PENUTUP**

VI.1 Kesimpulan 91

VI.2 Saran 91

**Daftar Pustaka**

**Lampiran**

**DAFTAR GAMBAR**

**Gambar 2.1** Tiga Lapis Jaringan Galat Mundur 9

**Gambar 2.2** One to One 19

**Gambar 2.3** One to Many 20

**Gambar 2.4**  Many to Many 20

**Gambar 4.1**  Arsiterktur JST 31

**Gambar 4.2**  Diagram Konteks 32

**Gambar 4.3**  Diagram Jenjang 32

**Gambar 4.4**  Diagram Rinci Level 1 33

**Gambar 4.5**  Diagram Rinci Level 2 Proses 2 34

**Gambar 4.6**  Diagram Rinci Level 2 Proses 3 35

**Gambar 4.7**  Flowchart Pelatihan Data 36

**Gambar 4.8**  Flowchart Pengujian Data 37

**Gambar 4.9** Flowchart Prediksi Diagnosa Gangguan Makan 38

**Gambar 4.10** Entity Relationship Diagram (ERD) 40

**Gambar 4.11**  Relasi Tabel 44

**Gambar 4.12** Perancangan Form Login 45

**Gambar 4.13** Perancangan Utama Pada Menu Master 46

**Gambar 4.14** Perancangan Utama Pada Proses JST 46

**Gambar 4.15** Perancangan Halaman Prediksi 47

**Gambar 4.16** Perancangan Halaman User 48

**Gambar 4.17** Perancangan Halaman Kategori 48

**Gambar 4.18** Perancangan Halaman Pertanyaan 49

**Gambar 4.19** Perancangan Halaman Gangguan 50

**Gambar 4.20** Perancangan Halaman Input Data Latih 50

**Gambar 4.21** Perancangan Halaman Pelatihan 51

**Gambar 4.22** Perancangan Halaman Pengujian 52

**Gambar 5.1** Halaman Login 54

**Gambar 5.2** *Source Code* Perintah *Login* 54

**Gambar 5.3** Informasi *Login* Berhasil 55

**Gambar 5.4** Halaman Utama 56

**Gambar 5.5** *Source Code* Menu Utama 57

**Gambar 5.6**  Halaman Data User 57

**Gambar 5.7** *Source Code* Halaman Data User 58

**Gambar 5.8** Halaman Halaman data Kategori 59

**Gambar 5.9**  *Source Code* Data Kategori 59

**Gambar 5.10** Halaman Data Pertanyaan 60

**Gambar 5.11** *Source Code* Pertanyaan 61

**Gambar 5.12** Halaman Data Gangguan 62

**Gambar 5.13** *Source Code* Gangguan 62

**Gambar 5.14** Halaman Input Data 63

**Gambar 5.15** *Source Code* Input Data 64

**Gambar 5.16** Halaman Input Gangguan 65

**Gambar 5.17** *Source Code* Input Gangguan 66

**Gambar 5.18** Halaman Pelatihan 67

**Gambar 5.19** *Source Code* Pelatihan 68

**Gambar 5.20** Pengujian Data Yang Telah Dilatih 69

**Gambar 5.21** *Source Code* Pengujian 70

**Gambar 5.22** Pengujian Data Yang Belum Dilatih 75

**Gambar 5.23** *Source Code* Pengujian 76

**Gambar 5.24** Halaman Data Prediksi 82

**Gambar 5.25** *Source Code* Perintah Identifikasi 83

**Gambar 5.26** Halaman Data Pelatihan Manual 88

**DAFTAR TABEL**

**Tabel 2.1** Tabel Perbandingan Tinjauan Pustaka 6

**Tabel 2.2** Tabel Simbol DAD (Diagram Alur Data) 18

**Tabel 2.3** Tabel Simbol-simbol ERD 19

**Tabel 2.4** Tabel Simbol-simbol Flowchart 22

**Tabel 3.1** Tabel Data Objek Penelitian 23

**Tabel 3.2** Tabel Angket Pertanyaan Kategori Pengaruh Citra Tubuh 24

**Tabel 3.3** Tabel Angket Pertanyaan Kategori Kebiasaan Makan 25

**Tabel 3.4** Tabel Angket Pertanyaan Kategori Riwayat Diet 25

**Tabel 3.5** Tabel Nilai Jawaban 26

**Tabel 4.1** Tabel User 41

**Tabel 4.2** Tabel Kategori 41

**Tabel 4.3** Tabel Pertanyaan 42

**Tabel 4.4** Tabel Perolehan 42

**Tabel 4.5** Tabel Gangguan 43

**Tabel 4.6** Tabel Bobot Hidden 43

**Tabel 4.7** Tabel Bobot Output 44

**Tabel 5.1** Hasil Pelatihan Data dalam bentuk Kuisioner 71

**Tabel 5.2** Data Pelatihan 72

**Tabel 5.3** Contoh Hitung Nilai Bobot 1 73

**Tabel 5.4** Hasil Pengujian Data Terlatih 74

**Tabel 5.5** Pengujian Data Tak Terlatih dalam Bentuk Kuisioner 78

**Tabel 5.6** Data Pengujian 79

**Tabel 5.7** Contoh Hitung Nilai Bobot 2 80

**Tabel 5.8** Hasil Pengujian Data Tak Terlatih 81

**Tabel 5.9** Data Perhitungan Manual 84

**Tabel 5.10** Diagnosa 84

**Tabel 5.11** Bobot Input ke Hidden 84

**Tabel 5.12** Bobot Hidden ke Output 85

**Tabel 5.13** Normalisasi Data 86

**Tabel 5.14** Data Pelatihan 89

**Tabel 5.15** Data Uji Tidak Terlatih 89

**Tabel 5.16** Hasil Pelatihan 89

**Tabel 5.17** Hasil Pengujian 90

**BAB I**

**PENDAHULUAN**

* 1. **Latar Belakang**

Perubahan kebiasaan makan sangat tidak baik jika dibiarkan, hal ini akan menyebabkan gangguan makan. Gangguan makan digambarkan sebagai gangguan berat dalam perilaku makan dan perhatian yang berlebihan tentang berat dan bentuk badan. Biasanya remaja tidak memperhatikan aspek gizi dan kesehatan ketika sedang makan. Sebenarnya gangguan makan atau eating disorders tidak sepenuhnya berhubungan dengan makanan, pola makan, dan berat badan, tetapi mengenai perasaan dan ekspresi diri atau merupakan sebuah penyakit mental, yang mana penyakit ini merupakan ancaman serius karena penderita akan makan dalam jumlah banyak atau justru makan dalam jumlah sedikit.

Penderita gangguan makan biasanya adalah orang-orang yang memiliki tingkat kepercayaan diri yang rendah, dimana mereka merasa tidak sebanding dengan orang lain. Mungkin untuk mengidentifikasi penyakit “gangguan makan” terhihat sangat sederhana, namun kenyataannya keadaan ini tidak hanya mencakup satu perilaku, namun ada berbagai perilaku abnormal yang dapat hadir dalam sejumlah gambaran klinis.

Berdasarkan keterangan *The National Eating Disorder Association* sebanyak 20 juta perempuan dan 10 juta laki-laki megalami gangguan makan serius. Kondisi ini mencakup *anorexia nervosa*, *bulimia nervosa*, dan *binge eating disorder* (BED). Sebagian besar gangguan makan disebabkan oleh interaksi kompleks dari faktor genetik, biologis, perilaku, psikologis, dan sosial.

Hasil penelitian ini diharapkan dapat membantu remaja yang merasa bahwa dirinya mengalami kecenderungan gangguan makan, dan termasuk dalam kategori *anorexsia nervosa*, *bulmia nervosa* atau *binge eating disorder* (BED). Sehingga setelah remaja atau individu mengidentifikasi kategori kecenderungan gangguan makan yang dideritanya, individu tersebut dapat mencari pertolongan untuk mengobatinya kepada pihak psikiatri.

Dari latar belakang tersebut maka diperlukan sistem yang dapat mengidentifikasi kecenderungan gangguan makan pada remaja sehingga dapat membantu remaja mencegah terjadinya gangguan makan di kemudian hari.

* 1. **Rumusan Masalah**

Berdasarkan latar belakang masalah yang telah diuraikan, maka permasalahan yang ada dapat dirumuskan sebagai berikut:

1. Bagaimana membangun sistem yang dapat mendeteksi kecenderungan gangguan makan pada remaja menggunakan metode *Backpropagation*?
2. Berapa tingkat ketepatan hasil deteksi kencenderungan gangguan makan pada remaja menggunakan metode *Backpropagation*?
   1. **Batasan Masalah**

Berdasarkan rumusan masalah diatas maka penulis membatasi masalah yang akan diteliti adalah :

1. Objek penelitian ini adalah remaja yang masih dalam masa pertumbuhan yaitu usia 15-19 tahun.
2. Pertanyaan yang digunakan sebagai inputan merupakan skala psikologi buatan Laila (2009) dengan beberapa modifikasi dan berfokus pada perilaku remaja terhadap presepsi bentuk tubuh ideal. Modifikasi yang dilakukan berupa penambahan skala jawaban dari 2 pilihan jawaban (ya atau tidak) menjadi 5 pilihan jawaban (tidak pernah, hampir tidak pernah, kadang-kadang, sering, atau sangat sering), serta pembagian kategori antara lain pengaruh citra tubuh, kebiasaan makan, dan riwayat diet.
3. Jawaban individu akan diproses oleh sistem dengan menerapakan metode Jaringan Saraf Tiruan *Backpropagation*.
4. Output yang dihasilkan yaitu berupa informasi apakah remaja tersebut mengalami kecenderuangan gangguan makan, dan termasuk *anorexsia nervosa*, *bulmia nervosa,* *binge eating disorder* atau normal.
   1. **Tujuan Penelitian**

Tujuan yang ingin dicapai dalam penelitian ini antara lain:

1. Membangun sebuah sistem yang dapat membantu untuk mendeteksi terjadinya kecenderungan gangguan makan pada remaja mengggunakan metode *Backpropagation* yang bertujuan memudahkan remaja yang bertindak sebagai user dalam mendeteksi resiko gangguan makan yang mungkin dideritanya.
2. Mengetahui tingkat ketepatan deteksi gangguan makan pada remaja menggunakan jaringan saraf tiruan metode *Backpropagation*.
   1. **Manfaat Penelitian**

Dengan adanya sistem yang dapat mengidentifikasi jenis gangguan makan mana yang diderita agar dapat ditanggulangi atau diatasi lebih awal sehingga tidak menyebabkan berbagai masalah kesehatan lain, seperti gangguan pencernaan, malnutrisi dan gangguan pertumbuhan.

* 1. **Sistematika Penulisan**

Untuk memudahkan pemahaman dalam penyusunan laporan Tugas Akhir ini, maka sistematika penulisan laporan adalah sebagai berikut:

BAB I. PENDAHULUAN

Bab ini akan menjelaskan mengenai latar belakang masalah, rumusan masalah, batasan masalah, tujuan penelitian, dan sistematika penuliisan.

BAB II. LANDASAN TEORI

Bab ini berisi sekumpulan teori dan dan pembahasan yang diteliti serta memabahas penelitian-penelitian terdahulu yang memiliki keterkaitan dengan penelitian yang sedang dilakukan penulis.

BAB III. METODE PENELITIAN

Bab ini akan memaparkan bagaimana sistem dibangun dari awal, mulai dari analisis data hingga pengujian pada sistem telah selesai dibuat.

BAB IV. ANALISIS DAN PERANCANGAN SISTEM

Menjelaskan tentang perancangan dari system pendeteksi kecenderungan gangguan makan.

BAB V. IMPLEMENTASI DAN PEMBAHASAN

Menguraikan tentang cara kerja dari sistem, bagaimana cara menggunakan system serta melakukan pengujan sistem.

BAB VI. PENUTUP

Berisi kesimpulan dan saran berdasarkan penelitian yang telah dilakukan.

DAFTAR PUSTAKA

Berisi tentang kumpulan sumber yang digunakan sebagai acuan penulisan laporan.

LAMPIRAN

Berisi lampiran *sourcode* program yang dibuat.

**BAB II**

**LANDASAN TEORI**

* 1. **Kajian Hasil Penelitian**

Beberapa hasil penelitian yang pernah dilakukan oleh peneliti sebelumnya dan menjadi acuan dan sumber referensi dalam penelitian ini adalah :

1. Penelitian oleh Yudhawati (2014), dengan judul Implementasi Jaringan Saraf Tiruan untuk Mendeteksi Daya Tahan Terhadap Stres pada Mahasiswa Tingkat Akhir menggunakan Metode *Backpropagation.* Penelitian tersebut membahas tentang bagaimana cara mendeteksi daya tahan terhadap stres pada mahasiswa tingkat menggunkan metode *backpropagation*. Seorang mahasiswa tingkat akhir haruslah memiliki daya tahan yang tinggi berkaitan dengan kemampuan menghadapi stres, selain itu harus memiliki motivasi yang tinggi serta mengendalikan tingkah laku. Setelah sukses mendeteksi tingkat stres maka sistem akan memberikan penanganan atau solusi kepada mahasiswa.
2. Penelitian oleh Andrian dan Wahyudi (2014), dengan judul Anaisis Kinerja Jaringan Saraf Tiruan Metode *Backpropagation* dalam memprediksi cuaca di Kota Medan. Penelitian tersebut membahas tentang bagaimana cara memprediksi perkiraan cuaca mendatang menggunakan metode *backpropagation*. Parameter data yang digunakan dalam memprediksi cuaca yaitu data curah hujan, data suhu, serta data kelembaban pada tahun 1997-2013. Jaringan saraf tiruan metode *backpropagation* dapat mengenali pola data yang telah diberikan dengan baik.
3. Penelitian oleh Siswoyo dkk. (2014), dengan judul Klasifikasi Sinyal Otak Menggunkan Metode Logika *Fuzzy* dengan *Neurosky Mindset*. Penelitian ini membahas tentang bagaimana mengetahui informasi sinyal apa yang terdapat pada otak manusia. Tujuan aplikasi ini dikembangkan yaitu untuk menerjemahkan sinyal EEG sehingga dapat melakukan pengontrolan suatu perangkat. Metode yang digunkan adalah *fuzzy* mamdani. klasifikasi sinyal otak ini berupa sinyal yang berbeda antara lain *Alfa, Beta, Theta, Gama, Attention, Meditasi.*

**Tabel 2.1** Perbandingan Tinjauan Pustaka

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No. | Judul | Penulis | Metode | Hasil Kesimpulan |
| 11. | Implementasi Jaringan Saraf Tiruan untuk Mendeteksi Daya Tahan Terhadap Stress pada Mahasiswa Tingkat Akhir menggunakan Metode *Backpropagation* | Yudhawati  (2014) | Algoritma *Backpropagation* | Setelah dilakukan pelatihan dengan diberikan beberapa parameter berupa jumlah hiden layer, jumlah iterasi maksimal, konstanta belajar, selang tampilan dan SSE yang diijinkan maka tingkat keakuratan sistem ini terletak pada iterasi 10,334 atau hanya sekitar 75% dari tahap pengujian. |
| 22. | Analisis Kinerja Jaringan Saraf Tiruan Metode Backpropagation dalam mempresiksi cuaca di Kota Medan | Andrian, Wayuhdi  (2014) | Algoritma *Backpropagation* | Jaringan saraf tiruan metode *backpropagation* dapat mengenali pola data yang telah diberikan dengan baik. Tingkat keaurasian terbesar pada proses pengujian prediksi cuaca di Kota Medan degan jaringan saraf tiruan metode *backpropagation* adalah pada data kelembaban 86,28% pada kuadrat *error* 0,01. |
| 33. | Klasifikasi Sinyal Otak Menggunkan Metode Logika *Fuzzy* dengan *Neurosky Mindset* | Siswoyo dkk.  (2014) | Logika *Fuzzy* | Hasil penelitian menunjukan bahwa hasil pengklasifikasian adalah pilihan untuk klasifikasi data BCL Logika Fuzzy dalam mengontrol nyala lampu LED berhasil dipakai dengan sebuah klasifikasi sinyal otak. Dengan kondisi konsentrasi pemakai dapat mengatur nyala lampu. Ketika kondisi konsentrsi pemakai rendah makan LED redup dan jika konsentrasi pemakai tinggi maka nyala LED akan terang. |

Seperti terlihat pada table 2.1 perbedaan dari ketiga referensi dengan judul yang diangkat oleh penulis terletak pada tema dan metode yang digunakan, yaitu *Backpropagation* untuk mendeteksi kecenderungan makan pada remaja..

* 1. **Kajian Teori**
     1. **Jaringan Saraf Tiruan**

Menurut Hermawan (2006), Jaringan saraf tiruan didefinisikan sebagai suatu sistem pemrosesan informasi yang mempunyai karakteristik menyerupai jaringan saraf manusia. Jaringan saraf tiruan tercipta sebagai suatu geeralisasi model matematis dari pemahaman manusia (*human cognition*) yang didasarkan data asumsi sebagai berikut:

1. Pemrosesan informasi terjadi pada elemen sederhana yang disebut *neuron*.
2. Isyarat mengalir diantara sel saraf/*neuron* melalui suatu sambungan penghubung.
3. Setiap sambungan penghubung memiliki bobot yang bersesuaian. Bobot ini akan digunakan untuk menggandakan atau mengalikan isyarat yang dikirim melaluinya.
4. Setiap sel saraf akan menerapkan fungsi aktivasi terhadap isyarat hasil penjumlahan berbobot yang masuk kepadanya untuk menentukan isyarat keluarannya.

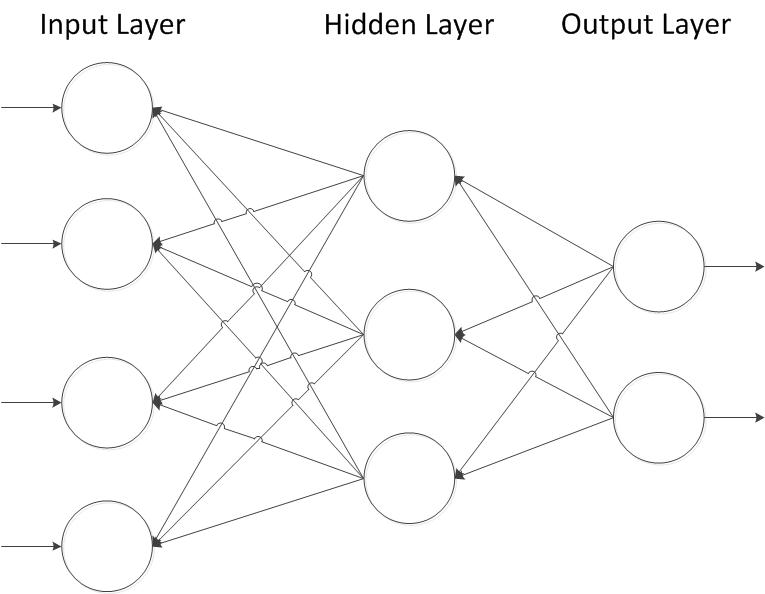
Jaringan saraf tiruan merupakan sistem komputasi dimana arsitektur dan operasi diilhami dari pengetahuan tentang sel saraf biologis di dalam otak, yang merupakan salah satu representasi buatan dari otak manusia yang selalu mencoba menstimulasi proses pembelajaran pada otak manusia tersebut. Jaringan saraf tiruan dapat digambarkan sebagai model matematis dan komputasi untuk fungsi aproksimasi non-linear, klasifikasi data cluster dan regresi non-parametrik atau sebuah simulasi dari koleksi model saraf biologi (Hermawan, 2006).

* + 1. **Perambatan Galat Mundur (*Backpropagation*)**

Jaringan perambatan galat mundur (*backpropagation*) memiliki algoritma pelatihan dengan menggunakan metode belajar terbimbing pada jaringan diberikan sepasang pola yang terdiri atas pola masukan dan pola yang diinginkan ketika suatu pola diberikan kepada jaringan, bobot-bobot diubah untuk memperkecil perbedaan pola keluaran dan pola yang diinginkan latihan ini dilakukan secara berulang-ulang agar semua pola yang dikeluarkan jaringan dapat memenuhi pola yang diinginkan (Hermawan, 2006).

* + - 1. **Algoritma *Backpropagation***

Algoritma pelatihan jaringan saraf *backpropagation* (perambatan galat mundur) terdiri atas 2 langkah, yaitu perambatan maju dan perambatan mundur. Langkah perambatan maju dan perambatan mundur ini dilakukan pada jaringan untuk setiap pola yang diberikan selama jaringan mengalami pelatihan. Gambar 2.1 telah menunjukkan Jaringan *backpropagation* dengan 3 lapisan pengolah, yaitu lapisan masukan, lapisan tersembunyi, dan lapisan keluaran. Ketiga lapisan tersebut terhubung secara penuh (Hermawan, 2006).



**Gambar 2.1** Tiga Lapisan Perambatan Galat Mundur

Selama perambatan maju, tiap unit masukan xi menerima sebuah masukan sinyal ke masing-masing lapisan tersembunyi z1,…….,zp. Tiap unit tersembunyi ini kemudian menghitung aktivasinya dan mengirimkan sinyalnya (zj) ke tiap unit keluaran. Tiap unit keluaran (yk) menghitung aktivasiya (yk) untuk membentuk respon pada jaringan untuk memberikan pola masukan,

Selama pelatihan, tiap unit keluaran membandingkan perhitungan aktivasinya yk dengan nilai targetnya untuk menentukan kesalahan pola tersebut dengan unit itu. Berdasarkan kesalahan ini, faktor (k = 1,…….,m) dihitung. digunakan untuk menyebarkan kesalahan pada unit keluaran yk kembali kesemua unit pada lapisan sebelunya(unit-unit tersembunyi yang dihubungkan ke yk). Selain itu digunakan untuk memperbaharui bobot-bobot antara keluaran dan lapisan tersembunyi. Dengan cara yang sama, faktor (j = 1,…….p) dihitung untuk tiap unit tersembunyi zj. Tidak perlu untuk menyebarkan kesalahan kembali ke lapisan masukan, tetapi digunakan untuk memperbaharui bobot-bobot antaralapisan tersembunyi dan lapisan masukan.

Setelah seluruh faktor δ ditentukan, bobot untuk semua laisan diatur secara serentak. Pengaturan bobot (dari unit tersembunyi zj ke unit keluaran yk) didasarkan pada faktor dan aktivasi zj dari unit masukan. Untuk langkah selengkapnya adalah sebagai berikut:

**Prosedur Pelatihan**

Algoritma selengkapnya pelatihan *backpropagation* adalah sebagai berikut:

Langkah 0 : Inisialisasi bobot-bobot (tetapkan nilai acak kecil)

Langkah 1 : Bila syarat berhenti adalah salah, kerjakan langkah 2

sampai 9.

Langkah 2 : Untuk setiap pasangan pelatihan, kerjakan langkah 3 sampai 8.

**Perambatan Maju**

Langkah 3 : Tiap unit masukan (xi, *i* = 1,…., n) menerima isyarat

masukan xi dan diteruskan ke unit-unit tersembunyi.

Langkah 4 : Setiap unit tersembunyi (zi, *i*=1, … p)

jumlahkan bobot sinyal masukannya,

= bias pada unit j aplikasi fungsi aktifasi menghitung sinyal keluaran dengan menerapkan fungsi aktivasi hitung:

Dan kirimkan sinyal ini keseluruh unit pada lapisan diatasnya (unit-unit keluaran).

Langkah 5 : Tiap unit keluaran (yk, k=1, … m)

jumlahkan bobot sinyal masukannya.

= Bias pada unit keluaran k dan aplikasi fungsi aktivasinya untuk menghitung sinyalkeluaran dengan menerapkan fungsi aktivasi hitung,

**Perambatan Mundur:**

Langkah 6 : Tiap unit keluaran (yk, k=1, …… m) menerima pola target berkaitan dengan masukan pola pelatihan. Hitung galat atau kesalahan informasinya:

Hitung koreksi bobotnya(digunakan untuk memperbaharui nantinya)

Hitung koreksi biasnya(digunakan untuk memperbaharui nantinya dikirimkan ke unit-unit pada lapisan dibawahnya).

Langkah 7 : Setiap unit lapisan tersembunyi (zj, j=1, …... p) jumlahkan hasil perubahan masukannya (dari unit-unit dilapisan atasnya).

Hitung galat informasinya:

Hitung koreksi bobot biasnya:

Langkah 8 : Tiap unit keluaran (yk, k=1, …, m) perbaharui

bobot-bobot dan biasnya (j=0,1, …, p)

Tiap unit tersembunyi (zj, j=1, …, p) memperbaharui

Bobot dan prasikapnya (i=0,1, … n);

Langkah 9 : Uji syarat berhenti.

Perhitungan kesalahan merupakan pengukuran bagaimana jaringan dapat belajar dengan baik sehingga jika dibandingkan dengan pola yang baru akan dengan mudah dikenali. Kesalahan pada keluaran jaringan merupakan selisih antara keluaran sebenarnya (*current output*) dan keluaran yang diinginkan (*desired output*). Selisih yang dihasilkan antara keduanya biasanya ditentukan dengan cara dihitung menggunakan suatu persamaan. (Hermawan, 2006).

Langkah menghitung *Sum Square Error* (SSE) adalah sebagai berikut:

* + - 1. Hitung keluaran jaringan saraf untuk masukan pertama.
      2. Hitung selisih antara nilai keluaran jaringan saraf dan nilai target / yang diinginkan untuk setiap keluaran.
      3. Kuadratkan setiap keluaran kemudian hitung seluruhnya. Ini merupakan kuadrat kesalahan untuk contoh latihan.

Adapun rumusnya adalah:

Dengan:

T*jp* : nilai keluaran jaringan saraf

X*jp* : nilai target / yang diinginkan untuk setiap keluaran

Selain SSE dibutuhkan juga *Mean Square Error* (MSE) dalam proses pengujian:

Hitung SSE.

Hasilnya dibagi dengan perkalian antara banyaknya data pada pelatihan dan banyaknya keluaran.

Rumus:

Dengan:

T*jp* : nilai keluaran jaringan saraf

X*jp* : nilai target / yang diinginkan untuk setiap keluaran

n : jumlah data pembelajaran

* + 1. **Deteksi**

Poerwadarminta (2007) Dalam Kamus Besar Bahasa Indonesia ketiga. Deteksi adalah usaha untuk menemukan keberadaan, anggapan atau kenyataan. Deteksi umumnya suatu proses untuk memeriksa atau melakukan pemeriksaan terhadap sesuatu dengan menggunakan cara dan teknik tertentu.

* + 1. **Remaja**

Menurut WHO, remaja adalah penduduk dalam jangka usia 10-19 tahun, menurut Peraturan Menteri Kesehatan RI Nomer 25 Tahun 2014, remaja adalah penduduk dalam rentang usia 10-18 tahun dan menurut Badan Kependudukan dan Keluarga Berencana (BKKBN) rentang usia remaja adalah 10-24 tahun dan belum menikah. Jumlah kelompok usia 10-19 tahun di Indonesia menurut Sensus Penduduk 2010 sebanyak 43,5 juta atau sekitar 18% dari jumlah penduduk. Di dunia diperkirakan remaja berjumlah 1,2 milyar atau 18% dari jumlah penduduk dunia(WHO, 2014).

Masa remaja merupakan masa peralihan dari masa kanak-kanak ke dewasa. Selain itu masa ini merupakan periode terjadinya pertumbuhan dan perkembangan baik secara fisik, psikologis maupun intelektual. Sifat khas remaja mempunyai rasa keingintahuan yang besar, menyukai petualangan dan tantangan.

* + 1. **Gangguan Makan**
       1. **Definisi Gangguan Makan**

Gangguan makan adalah suatu penyakit mental yang dapat menjadikan ancaman serius bagi seseorang biasanya ditandai dari pola makan orang sehari-hari, seperti makan dalam jumlah yang sangat sedikit atau makan secara berlebihan (National Institute of Mental Health, 2016). Kondisi ini dapat dimulai hanya dari makan terlalu sedikit atau terlalu banyak tetapi memiliki obsesi pada makanan selama kehidupan seseorang yang mengarah pada perubahan yang parah. Selain pola makan abnormal yang berbahaya dan adanya kekhawatiran tentang berat badan atau bentuk tubuh, gangguan ini seringkali terjadi bersama dengan penyakit mental lainnya seperti depresi, penyalahgunaan zat, atau gangguan kecemasan. (National Institute of Mental Health, 2016).

* + - 1. **Anorexia Nervosa**

Davison dkk. (2010) menjelaskan *anorexia nervosa* berasal dari istilah anorexia yang berarti hilangnya selera makan dan nervosa yang berarti dipengaruhi oleh emosional, sehingga anorexia nervosa sering disebut dengan hilangnya selera makan dikarenakan pengaruh emosional. Akan tetapi pengertian tersebut tidak tepat karena sebagian besar penderita anorexia nervosa secara aktual tidak kehilangan selera makan. Anorexia nervosa adalah sindrom dimana seseorang dengan sengaja melaparkan dirinya untuk menjadi kurus, dan mengalami penurunan berat badan yang sangat drastis (Davison dkk. 2010). Selain itu Anorexia nervosa adalah sindrom dimana seseorang mempertahankan berat badanya agar tetap rendah dan biasanya mereka takut akan mengalami kegemukan dan cenderung mempertahankan berat badan agar tetap kurus.

Orang dengan *anorexia nervosa* mungkin menganggap diri mereka kelebihan berat badan, bahkan saat mereka mengalami kekurangan berat badan. Orang dengan *anorexia nervosa* biasanya menimbang diri berulang kali, sangat membatasi jumlah makanan yang mereka makan, dan makan dalam jumlah sangat kecil untuk makanan tertentu. *Anorexia nervosa* memiliki tingkat kematian tertinggi pada setiap gangguan jiwa. Sementara banyak wanita muda dan pria dengan gangguan ini meninggal karena komplikasi yang terkait dengan kelaparan, yang lainnya meninggal karena bunuh diri. Pada wanita, bunuh diri jauh lebih umum pada orang dengan *anorexia* dibandingkan kebanyakan gangguan jiwa lainnya (National Institute of Mental Health, 2016). Gejalanya meliputi:

1. Sangat terbatas makan.
2. Enggan untuk mempertahankan berat badan normal atau sehat.
3. Ketakutan yang intens akan pertambahan berat badan.
4. Citra tubuh yang menyimpang, harga diri yang sangat dipengaruhi oleh persepsi berat badan dan bentuk tubuh, atau penolakan serius terhadap berat badan yang rendah.

Terdapat 4 ciri yang diperlukan untuk menegakkan diagnosis Anorexia nervosa sebagai beikut (Davison dkk. 2010):

1. Penderita menolak untuk mempertahankan berat badan normal, hal ini biasanya berarti bahwa berat badan orang tersebut kurang dari 85% dari berat badan yang dianggap normal bagi usia dan tinggi badannya.
2. Penderita sangat takut bila berat badannnya bertambah dan rasa takut tersebut tidak berkurang dengan turunnya berar badan serta tidak pernah merasa cukup kurus.
3. Penderita memiliki pandangan yang menyimpang tentangbentuk tubuh mereka. Bahkan dalam kondisi kurus kering mereka tetap merasa bahwa mereka kelebihan berat badan atau beberapa bagian tubuh tertentu, khususnya perut, pantat, dan paha terlalu gemuk.
4. Pada penderita perempuan, kondisi tubuh yang sangat kurus menyebabkan amenorea, yaitu berhentinya periode menstruasi.

Sedangkan menurut Diagnostic and Statistical Manual of Mental Disorder IV (DSM-IV) terdapat 4 kriteria untuk menegakkan diagnosis *anorexia nervosa* yaitu:

1. Menolak mempertahankan berat badan normal.
2. Meskipun berat badannya sangat kurang, namun mengalami ketakutan yang amat sangat menjadi gemuk.
3. Gangguan citra tubuh.
4. Pada perempuan yang telah mengalami menstruasi, terjadi amenorea.
   * + 1. **Bulimia Nervosa**

Orang dengan *bulimia nervosa* memiliki episode berulang dan sering makan dalam jumlah besar dan secara cepat dan tidak terkendali. Tindakan tersebut diikuti dengan perilaku berlebihan, seperti muntah paksa, penggunaan obat pencahar atau diuretik yang berlebihan, puasa atau olahraga berlebihan dengan tujuan mencegah bertambahnya berat badan atau kombinasi dari perilaku ini. Tidak seperti anorexia nervosa, orang dengan *bulimia nervosa* biasanya mempertahankan berat badan yang sehat atau relatif normal (National Institute of Mental Health, 2016).

Menurut Diagnostic and Statistical Manual of Mental Disorder IV (DSM-IV) terdapat beberapa kriteria untuk menegakkan diagnosis *bulimia nervosa* yaitu:

1. Mengalami periode binge eating berulang kali yang ditandai dengan dua kreiteria berikut:
   1. Memakan dalam jumlah yang besar (jauh lebih besar dari porsi normal) dalam satu periode waktu tertentu dengan jarak yang cukup dekat, missal setiap 2 jam sekali.
   2. Memiliki rasa tidak dapat mengontrol perilaku makan berlebihan saat episode berikut berlangsung.
2. Melakukan tindakan berlebihan untuk mencegah peningkatan berat badan, seperti muntah dengan sengaja, penyalahgunaan obat pencahar atau diuretik yang berlebihan, puasa atau olahraga berlebihan.
3. Terjadinya binge eating dan tindakan berlebihan yang tidak baik setiap 2 kali seminggu dalam 3 bulan.
4. Terlalu mengutamakan berat badan dan bentuk tubuh dalam mengevaluasi diri.
5. Gangguan ini tidak muncul secara ekslusif pada episode anorexia nervosa.
   * + 1. **Binge Eating Disorder**

Orang dengan binge eating disorder (BED) kehilangan kendali atas makanannya. Tidak seperti bulimia nervosa, periode binge eating tidak diikuti dengan pembersihan, olah raga berlebihan, atau puasa. Akibatnya, orang dengan gangguan binge eating sering kelebihan berat badan atau obesitas. Binge eating disorder adalah kelainan makan yang paling umum terjadi di Amerika Serikat (National Institute of Mental Health, 2016).

Gejala binge eating disorder (National Institute of Mental Health, 2016) meliputi:

1. Mengonsumsi makanan dalam jumlah besar dalam jumlah waktu tertentu.
2. Makan bahkan saat sedang kenyang atau tidak lapar.
3. Makan dengan sangat cepat di depan umum.
4. Makan sampai merasa tidak nyaman karena kekenyangan.
5. Makan sendirian atau diam-diam untuk menghindari rasa malu.
6. Merasa tertekan, malu, atau bersalah karena makan berlebihan.
7. Sering berdiet, mungkin tanpa penurunan berat badan.
   * 1. ***Data Flow Diagram* (DFD)**

*Data Flow Diagram* merupakan suatu diagram yang menggambarkan alir data dalam suatu entitas ke sistem atau sistem ke entitas. DFD juga dapat diartikan sebagai teknik grafis yang menggambarkan alir data dan transormasi yng digunakan sebagai perjalanan data dari input atau masukan menuju keluaran atau output (Saputra, 2016).

(Kadir, 2012) DFD adalah suatu model logika data atau proses yang dibuat untuk menggambarkan dari mana asal data dan kemana tujuan data yang keluar dari sistem, dimana data disimpan, proses apa yang menghasilkan data tersebut interaski antara data yang tersimpan dan proses yang dikenakan pada data tersebut.

Simbol-simbol DFD berdasarkan simbol DeMarco dan Yourdon dalam penggambaran DFD ditunjukkan dalam Tabel 2.2.

**Tabel 2.2** Simbol DAD

|  |  |
| --- | --- |
| SIMBOL | MAKNA |
|  | Notasi Kesatuan Luar DAD |
|  | Arus data yang mengalir dari kesatuan luar ke proses |
|  | Notasi proses di DAD |
|  | Simbol dari simpanan data di DAD |

* + 1. ***Entity Relational Diagram* (ERD)**

(Kadir, 2012) Entity Relationship Diagram (ERD) merupakan teknik yang digunakan untuk memodelkan kebutuhan data dari suatu organisasi, biasanya oleh Sistem Analys dalam tahap analisis persyaratan proyek pengembangan sistem. Sementara seolah-olah teknik diagram atau alat peraga memberikan dasar untuk desain database relasional yang mendasari sistem informasi yang dikembangkan. ERD bersama-sama dengan detail pendukung merupakan model data yang pada gilirannya digunakan sebagai spesifikasi untuk database.

Fathansyah (2015) Model *Entity Relationship Diagram* berisi komponen-komponen himpunan entitas dan himpunan relasi yang masing-masing dilengkapi dengan atribut-atribut yang merepresentasikan seluruh fakta dari ‘dunia nyata’ yang kita tinjau, dapat digambarkan dengan sistematis dengan Diagram *Entity-Relationship* (Diagram E-R).

Terdapat beberapa simbol yang digunakan dalam ERD, sebagaimana terlihat pada Tabel 2.3.

**Tabel 2.3** Simbol ERD

|  |  |
| --- | --- |
| **SIMBOL** | **DESKRIPSI** |
|  | Persegi Panjang, menyatakan himpunan entitas. |
|  | Elips, menyatakan atribut (atribut yang berfungsi sebagai key) |
|  | Belah Ketupat, menyatakan himpunan relasi. |
|  | Garis, sebagai penghubung antara entitas, relasi dan atribut |

Menurut Nova (2012), *Cardinality* (kardinalitas) menyatakan jumlah anggota entitas yang terlibat di dalam relasi yang terjadi. Dalam hal ini relasi yang terjadi akan membentuk relasi hubungan. Kardinalitas biasanya diekspresikan sebagai secara sederhana “satu” atau “banyak”. Beberapa bentuk kardinalitas diantaranya:

1. One To One (1:1)

Sebuah objek hanya dapat memiliki satu kejadian ke objek yang direlasikannya. Contohnya, seorang suami hanya dapat memiliki seorang istri, begitupula seorang istri hanya dapat memiliki seorang suami. Seperti pada Gambar 2.2



**Gambar 2.2** One To One

1. One To Many (1:M)

Sebuah objek hanya dapat memiliki satu kejadian ke objek yang direlasikannya, tetapi objek yang direlasikan dapat memiliki banyak kejadian untuk objek perelasi. Contohnya Seorang dosen dapat membimbing beberapa mahasiswa dan beberapa mahasiswa dapat dibimbing oleh seorang dosen. Seperti pada Gambar 2.3



**Gambar 2.3** One To Many

1. Many To Many (M:M)

Sebuah kejadian dari sebuah objek dapat berhubungan dengan satu atau lebih kejadian dari objek yang berhubungan, begitupula sebaliknya. Contohnya Seorang pelanggan dapat membeli beberapa barang dan sebuah barang dapat dibeli oleh beberapa pelanggan. Seperti pada Gambar 2.4



**Gambar 2.4** Many To Many

* + 1. **Flowchart**

Menurut Yakub (2012), Bagan alir (*flowchart*) adalah bagan yang menggambarkan urutan instruksi proses dan hubungan satu proses dengan proses lainnya menggunakan simbol-simbol tertentu. Bagan alir digunakan sebagai alat bantu komunikasi dan dokumentasi. Pada saat akan menggambarkan suatu bagan alir analisis sistem atau programmer dapat mengikuti pedoman-pedoman sebagai berikut:

1. Bagan alir sebaiknya digambarkan dari atas ke bawah dan mulai dari bagian kiri suatu halaman.
2. Kegiatan dalam bagan alir harus ditunjukan dengan jelas.
3. Harus ditunjukan dari mana kegiatan akan dimulai dan di mana akan berakhirnya.
4. Masing-masing kegiatan dalam alir sebaiknya digunakan suatu kata yang mengawali suatu pekerjaan.
5. Masing-masing kegiatan dalam bagan alir harus dalam urutan yang semestinya.
6. Kegiatan yang terpotong dan akan disambung di tempat lain harus ditunjukan dengan jelas menggunakan simbol penghubung.
7. Gunakan bagan alir yang standar.

Dalam penulisan bagan alir (*flowchart*) dikenal dengan tiga model yaitu:

1. Bagan Alir Sistem

Bagan alir sistem (*system flowchart*) merupakan bagan yang menunjukan pekerjaan secara keseluruhan dari sistem. Bagan ini menjelaskan urut-urutan dari prosedur-prosedur yang ada dalam sistem dan menunjukan apa yang dikerjakan sistem.

1. Bagan Alir Dokumen

Bagan alir dokumen (*document flowchart*) merupakan bagan yang menunjukan bagan alir yang menunjukan arus dari laporan dan formulir termasuk tembusan-tembusannya. Bagan alir dokumen menggunakan simbol-simbol yang sama dengan bagan alir sistem.

1. Bagan Alir Program

Bagan alir program (*program flowchart*) adalah suatu bagan yang menggambarkan ururtan proses secara mendetail dan hubungan anatara proses yang satu dengan proses lainnya dalam suatu program. Bagan alir program juga yang menjelaskan secara rinci langkah-langkah dari proses program. Bagan alir program dibuat dari derivikasi bagan alir sistem. Bagan alir programdapat terdiri dari bagan alir logika program (*program logic flowchart*) dan bagan alir program computer terinci (*detailed computer program flowchart*).

1. Bagan alir logika program, digunakan untuk menggambarkan tiap-tiap langkah dari dalam program secara logika.
2. Bagan alir program terinci, digunakan untuk menggambarkan instruksi-instruksi program komputer secara terinci.

Dalam menggambarkan bagan alir program (*program flowchart*) dapat menggunakan simbol-simbol standar yang biasa digunakan, sebagaimana terlihat pada Tabel 2.4.

**Tabel 2.4** Simbol-simbol Bagan Alir Program

|  |  |
| --- | --- |
| **Simbol** | **Keterangan** |
|  | Proses, digunakan untuk pengolahan aritmatiak dan pemindahan data. |
|  | Terminal, digunakan untuk menunjukan awal dan akhir dari program. |
|  | *Preparation,* digunakan untuk memberikan nilai awal pada suatu variabel. |
|  | Keputusan, digunakan untuk mewakili operasi perbandingan logika. |
|  | Proses terdefinisi, digunakan untuk prose yang detailnya dijelaskan terpisah. |
|  | Penghubung, digunakan untuk menunjukan hubungan arus proses yang terputus masih dalam halaman yang sama. |
|  | Penghubung halaman lain, digunakan untuk menunjukan hubungan arus proses yang terputus dalam halaman yang berbeda. |

**BAB III**

**METODE PENELITIAN**

* 1. **Objek Penelitian**

Adapun yang menjadi objek penelitian yaitu tentang kecendrungan gangguan makan yang dialami remaja. Dimana kecenderungan gangguan makan ini merupakan salah satu penyakit psikologi atau penyakit metal yang sering diderita oleh remaja. Kecenderungan gangguan makan dibagi menjadi beberapa jenis diantaranya *anorexia nervosa*, *bulimia nervosa*, *binge eating disorder* (BED) dan normal*.* Karena penyakit ini lebih banyak diderita oleh remaja maka objek penelitian yang digunakan adalah remaja dimana mereka masih dalam masa pertumbuhan baik fisik maupun mental.

**Tabel 3.1** Data Objek Penelitian

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| NNo | Jenis Gangguan | Jumlah Penderita | Laki-laki | Perempuan | Rentang usia |
| 1. | Anoreksia Nervosa | 11 | 4 | 7 | 15-19 |
| 2. | Bulimia Nervosa | 30 | 8 | 22 | 15-19 |
| 3. | Binge Eating Disorder | 12 | 3 | 9 | 16-19 |
| 4. | Normal | 13 | 2 | 11 | 16-19 |

* 1. **Analisis Sistem**

Pengumpulan data adalah suatu metode yang digunakan untuk mendapatkan suatu informasi yang harus dikerjakan pada saat pembuatan sistem. Untuk mempermudah penelitian ini peneliti menggunkan beberapa metode pengumpulan data, diantaranya adalah:

* + 1. **Observasi**

Observasi yaitu suatu kegiatan dengan melakukan pengamatan tentang kecenderungan gangguan makan pada remaja, pengamatan ini dilakukan dengan cara mengamati aktivitas-aktivitas yang sedang berjalan dan data-data yang diperlukan untuk memenuhi kebutuhan sistem yang akan dibuat.

* + 1. **Wawancara**

Wawancara merupakan suatu teknik pengumpulan data secara langsung, saling bertukar pikiran dan informasi mengenai permasalahan yang ditentukan. Materi dari wawancara meliputi masalah kecenderungan gangguan makan pada remaja, dimana gangguan makan merupakan salah satu penyakit mental atau penyakit psikologi klinis. Dalam penelitian ini wawancara dilakukan dengan sasaran kepada narasumber yang ahli dibidang psikologi oleh karena itu penulis memilih Dra. Sri Respati Adamari, M.Si., Psi sebagai narasumber. Dalam wawancara ini dilakukan pembagian kategori pertanyaan antara lain pengaruh citra tubuh, kebiasaan makan, dan riwayat diet.

* + 1. **Angket**

Di dalam penelitian ini angket merupakan alat ukur bagi responden yang nantinya dijadikan inputan untuk sistem yang akan dibuat. Didalam angket berisi sekumpulan pertanyaan yang terkait gangguan makan dengan diberikan beberapa dengan keadaan responden. Di dunia psikologi angket sering disebut dengan skala psikologi. Skala yang dipakai merupakan skala yang dibuat oleh Laila (2009) dengan beberapa modifikasi. Setelah melakukan wawancara dengan narasumber pertanyaan dari angket tersebut dibagi menjadi 3 ketegori antara lain pengaruh citra tubuh, kebiasaan makan, dan riwayat diet. Jawaban yang diberikan *user* akan disimbolkan dalam nilai kisaran 0 sampai 1 setiap pertanyaan.

**Tabel 3.2** Angket pertanyaan rasa percaya diri terhadap citra tubuh no 1-4

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No** | **Pertanyaan** | **TP** | **HTP** | **KD** | **CS** | **SS** |
| 1 | Seberapa sering kamu merasa gemuk? |  |  |  |  |  |
| 2 | Seberapa sering kamu merasa takut menjadi gemuk atau bertambah berat badan? |  |  |  |  |  |
| 3 | Seberapa sering berat badan kamu mempengaruhi tingkat kepercayaan diri kamu? |  |  |  |  |  |
| 4 | Seberapa sering bentuk tubuh kamu mempengaruhi tingkat kepercayaan diri kamu? |  |  |  |  |  |

**Tabel 3.3** Angket pertanyaan kategori kebiasaan makan no 5-13

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No** | **Pertanyaan** | **TP** | **HTP** | **KD** | **CS** | **SS** |
| 5 | Seberapa sering kamu makan dalam jumlah amat banyak menurut orang-orang sekitar kamu? |  |  |  |  |  |
| 6 | Seberapa sering kamu merasa tidak dapat berhenti makan atau sulit mengendalikan banyaknya makanan yang kamu makan? |  |  |  |  |  |
| 7 | Seberapa sering kamu makan dalam jumlah besar pada malam hari ketika dalam keadaan tidak bisa tidur? |  |  |  |  |  |
| 8 | Seberapa sering kamu makan lebih cepat dari biasanya? |  |  |  |  |  |
| 9 | Seberapa sering kamu makan sampai kamu merasa kekenyangan? |  |  |  |  |  |
| 10 | Seberapa sering kamu makan dalam jumlah besar ketika kamu tidak merasa lapar? |  |  |  |  |  |
| 11 | Seberapa sering kamu makan sendirian karena malu jika terlihat orang lain makan dalam jumlah banyak? |  |  |  |  |  |
| 12 | Seberapa sering kamu merasa malu atau kecewa ketika setelah makan berlebihan? |  |  |  |  |  |
| 13 | Seberapa sering kamu merasa marah setelah makan karena kamu tidak dapat mengendalikan perilaku makan sehingga berat badan kamu naik lagi? |  |  |  |  |  |

**Tabel 3.4** Angket pertanyaan riwayat diet 14-17

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No** | **Pertanyaan** | **TP** | **HTP** | **KD** | **CS** | **SS** |
| 14 | Seberapa sering kamu memuntahkan makanan untuk mencegah terjadinya kenaikan berat badan? |  |  |  |  |  |
| 15 | Seberapa sering kamu menggunakan obat pencahar, jamu, atau pelangsing untuk mencegah terjadinya kenaikan berat badan? |  |  |  |  |  |
| 16 | Seberapa sering kamu berpuasa atau makan kurang dari 2 kali sehari untuk mencegah terjadinya kenaikan berat badan? |  |  |  |  |  |
| 17 | Seberapa sering kamu berolahraga berlebihan untuk mencegah kenaikan berat badan? |  |  |  |  |  |

Sedangkan untuk nilai dari tiap bobot adalah sebagai berikut:

TP = Tidak Pernah

HTP = Hampir Tidak Pernah

KD = Kadang-Kadang

CS = Cukup Sering

SS = Sangat Sering

**Tabel 3.5** Nilai Jawaban

|  |  |
| --- | --- |
| Jawaban | Nilai |
| Tidak Pernah | 0 |
| Hampir Tidak Pernah | 0,25 |
| Kadang-Kadang | 0,50 |
| Cukup Sering | 0,75 |
| Sangat Sering | 1 |

* + 1. **Studi Kepustakaan**

Studi kepustakaan dilakukan untuk mencari landasan teori dari berbagai literature yang berkaitan masalah penelitian. Dalam tahap ini peeliti melakukan pencarian referensi yang berkaitan dengan kecenderungan gangguan makan pada remaja, jaringan saraf tiruan serta literature lain yang dibutuhkan. Bentuk dari pencarian referensi meliputi membaca buku, mengumpulkan konsep-konsep teoritis dan mempeljari karya-karya ilmiah terdahulu. Dari proses ini di dapatkan keterangan-keterangan serta teori-teori yang di jadikan dasar dalam penelitian ini.

* 1. **Perancangan Sistem**

Pada tahap ini akan menspesifikasikan bagaimana sistem dapat digunakan untuk mendeteksi kecenderungan gangguan makan pada remaja sejak dini. Untuk dapat memenuhi kebutuhan pengguna, sistem ini memerlukan tahap perancangan seperti *input*, *output*, *database*, output dan *interface*.

1. Desain Masukan(*input*)

Perancangan ini bertujuan untuk menentukan data masukan yang akan digunakan dalam menjalankan atau mengoperasikan sistem sesuai dengan kebutuhan.

1. Desain Proses

Perancangan prose ini dilakukan dengan menggunakan *flowchart* dan DFD (*Data Flow Diagram*). *Flowchart* berfungsi untuk menggambarkan intruksi proses dan hubungan satu proses dengan proses lainnya menggunakan simbol-simbol tertentu sedangkan untuk DFD untuk penggambaran sistem yang dilakukan dengan terstruktur yang dimulai dari diagram konteks, diagram jenjang yang merupakan level 0 dan digram rinci yang dimulai dari level 1 hingga level (n) yang menguraikan kegiatan dari diagram konteks dan diagram jenjang dengan jelas dan detail.

1. Desain *Database*

Desain *database* adalah pembuatan wadah penyimpanan data hasil kuisionersebagai data latih.

1. Desain Keluaran (*output*)

Perancangan keluaran bertujuan menentukan keluaran (output) yang akan digunakan oleh sistem. Keluaran (*output*) tersebut berupa tampilan di layar hasil deteksi kecenderungan gangguan makan dari kuisioner atau data yang telah dimasukkan.

1. Desain Interface

Desain interface adalah perancangan antarmuka dilakukan sesederhana mungkin (*user friendly*) agar mudah dipahami dan dimengerti oleh remaja atau pengguna, tetapi tidak menghilangkan unsur-unsur penting dalam menyampaikan informasi dari sistem.

* 1. **Implementasi Sistem**

Tahap implementasi ini bertujuan untuk melakukan kegiatan spesifikasi gambar rancangan sistem ke dalam kegiatan yang sebenranya dari sistem informasi yang akan dibangun atau dikembangkan, kemudian mengimplementasikan sistem tersebut ke dalam bahasa pemrograman java dengan aplikasi Netbeans IDE 8.0.2 dan MySQL yang berfungsi untuk pengelolaan data. Pada tahap implementasi ini sistem harus dapat berjalan secara optimal. Tahap implementasi ini mempunyai beberapa kegiatan yang harus dilakukan yaitu program dan testing data, pelatihan dan pengujian sistem yang telah dibuat atau dikembangkan.

1. Program dan Testing

Pada tahap ini dilakukan perancangan algoritma yang dilakukan dengan menggunakan pendekatan top-down (pemrograman modular). Selanjutnya adalah pembuatan program aplikasi yang menggunakan bahasa pemrograman yang sudah dipilih. Setelah program selesai dibuat maka akan dilakukan testing untuk mengetahui bagaimana kinerja sistem tersebut.

1. Pelatihan dan Pengujian

Tahapan ini dilakukan setelah sistem yang dibangun sudah jadi mengetahui kinerja sistem yang dibangun menggunakan dengan bahasa pemrograman yang digunakan dalam aplikasi tersebut. Setelah itu sistem akan dan memerlukan pelatihan secara menyeluruh. Dengan pelatihan maka akan diuji dengan beberapa data uji yang berfungsi untuk mengetahui berapa persen tingkat kebenannya.

* 1. **Perangkat Pendukung Penelitian**

Perangkat keras dan perangkat lunak yang digunakan sebagai pendukung penelitian sebagai berikut :

1. *Hardware*

*Hardware* yang dibutuhkan untuk menjalankan sistem informasi inventaris ini adalah:

1. Processor minimal Intel Core i3.
2. *Random Acces Memory* (RAM) minimal 1GB.
3. Printer digunakan untuk mencetak laporan hasil analisis.
4. *Software*

*Software* yang dibutuhkan dalam sistem informasi ini adalah:

1. *Operating System* (OS) atau sistem operasi yang digunakan minimal Windows 7.
2. *Database* (basis data) yang digunakan adalah Xampp.

*Tools* yang digunakan untuk membangun sistem adalah Netbeans IDE 8.0.2

**BAB IV**

**ANALISIS DAN PERANCANGAN SISTEM**

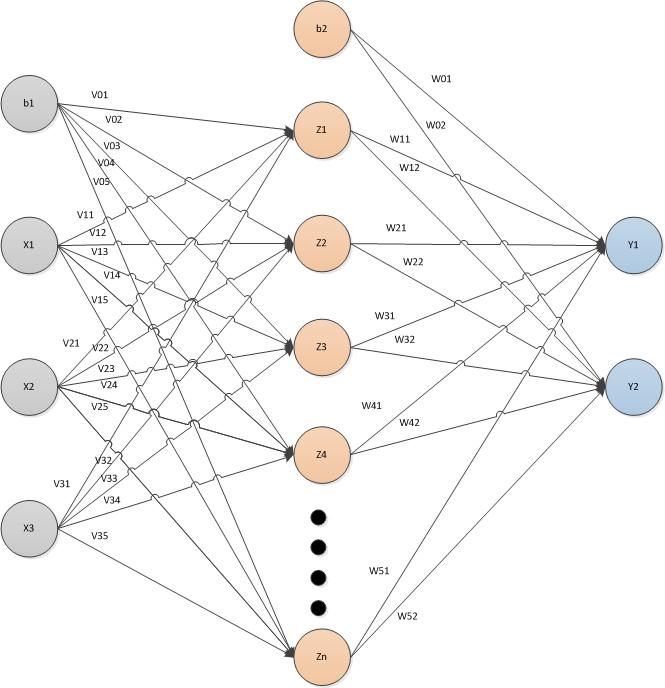
* 1. **Analisis Sistem**

Sistem deteksi kecenderungan gangguan makan merupakan sistem yang digunakan untuk menentukan apakah user mengidap ganguan makan atau tidak. Kecenderungan gangguan makan ini merupakan salah satu penyakit mental atau penyakit psikologi klinis dimana sering dialamai banyak orang akan tetapi keadaan tersebut sering tidak disadari terutama oleh para remaja. Gangguan makan yang diabaikan atau tidak segera ditangani akan berdampak pada kesehatan tubuh, gangguan makan ini akan memicu penyakit lain yang lebih berbahaya bahkan akan berdampak pada kematian. Kecenderungan gangguan makan pada penelitian ini dipusatkan pada remaja. Sistem ini dibuat menggunakan salah satu algoritma jaringan saraf tiruan yaitu *Backpropagation*. Jaringan saraf tiruan melakukan proses pengenalan gangguan makan berdasarkan dari pengalaman, pengalaman tersebut dilatihkan dan menghasilkan bobot yang dapat digunakan untuk mengenali gangguan makan seseorang. Input berupa kuisioner yang kemudian dibagi menjadi tiga kategori yaitu pengaruh citra tubuh, kebiasaan makan dan riwayat diet. Output yang dihasilkan dari sistem ini yaitu berupa deteksi gagguan makan yang dialami seseorang apakah mengalami kecenderungan *anorexia nervosa, bulimia nervosa, binge eating* disorder atau normal.

Pada penelitian ini digunakan data pengalaman atau data sample berupa sekelompok responden yang telah diwawancarai oleh seorang pakar psikologi dan juga mengisi formulir angket. Pakar psikologi tersebut yaitu Dra. Sri Respati Andamari, M.Si., Psi. Beliau merupakan dosen psikologi Universitas Teknologi Yogyakarta. Peneliti memilih beliau dengan alasan beliau seorang merupakan kosoler di student service UTY, selain itu beliau juga melayani konsultasi pada remaja di biro konsultasi miliknya sehingga lebih tahu mana yang memiliki memiliki kecenderungan gangguan makan dan tidak. Dari hasil wawancara yangdilakukan oleh narasumber dan responden tersebut akan dijadikan target bagi sistem.

Dari permasalahan yang terlalu sering akan berdampak pada penentuan deteksi awal kecenderungan gangguan makan, sehingga diperlukan sistem untuk mendeteksi kecenderungan gangguan makan dengan begitu remaja akan tau lebih awal kecenderungan gangguan makan yang dideritanya dan cepat dilakukan penanganan.

* 1. **Perancangan Sistem Implementasi JST untuk Mendeteksi Kecenderungan Gangguan Makan**
     1. **Perancangan Arsitektur Jaringan Syaraf Tiruan**



**Gambar 4.1** Arsitektur Jaringan

Pada gambar arsitektur jaringan saraf diatas, pada bagian lapisan input terdapat 3 inputan meliputi X1, X2, X3 dan 1 bias yang selalu bernilai 1. X1 mewakili kategori pengaruh citra tubuh, X2 mewakili kategori kabiasaan makan, dan X3 mewakili kategori riwayat diet. Pada bagian lapisan tersembunyi digambarkan dengan nilai Z1, Z2, Z3,Z4,…Zn. Zn merupakan lapisan tersembunyi keberapa, mengapa dibuatsampai n karena pada sistem jaringan tersebunyi menjadi parameter inputan sehingga jaringan tersembunyi dapat diberikan berbeda-beda pada setiap kali pelatihan, untuk menentukan tingkat konvergensi tercepat. Pada lapisan output tersebut hanya terdapat dua sel yaitu Y1 dan Y2 karena memiliki 4 nilai antara lain 00, 01, 10 dan 11 dimana 00 untuk *anoreksia nervosa*, 01 untuk *bulimia nervosa*, 10 untuk *binge eating disorder*, dan 11 untuk normal.

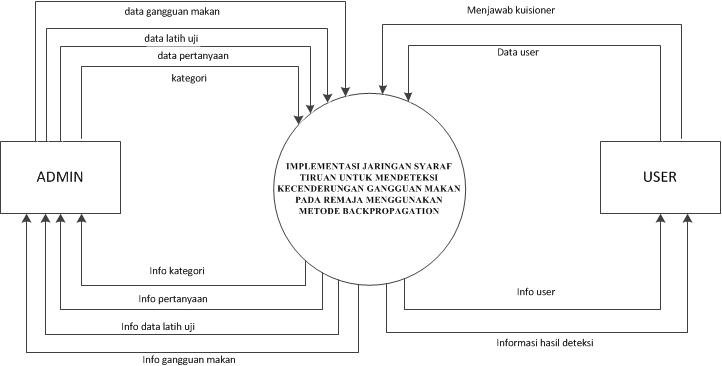
Bobot pada lapisan input menuju lapisan tersembunyi diwakilkan dengan variable Vij, i bernilai 0, 1, 2, 3 dan j sejumlah hidden layer dimana j bernilai 1, 2, 3, 4….n. sedangkang bobot dari lapisan tersembunyi menujulapisan output diberi variable Wj, dimana j merupakan jumlah hidden layer, jadi j=1,2,3…..n, sedangkan W0 merupakan bobot dari bias menuju lapisan keluaran.

* + 1. **Diagram Alir Data (DAD)**

Diagram Alir Data (DAD) merupakan alat untuk membuat diagram yang sebaguna dalam perancangan sistem untuk menggambarkan alur kerja sistem dalam pengolahan data dari input sampai output data.

* + - 1. **Diagram Konteks**

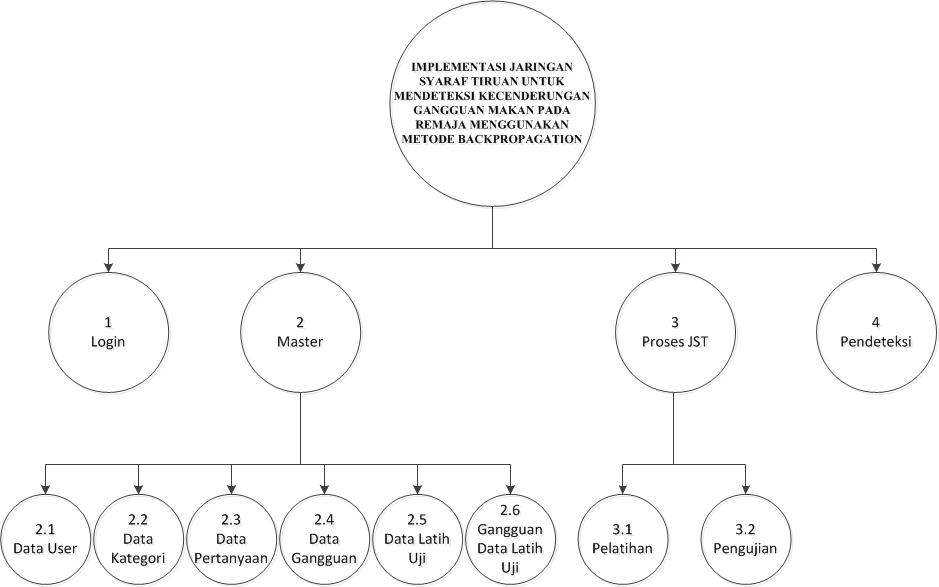
Diagram Konteks merupakan gambaran sistem secara garis besar, dimana gambaran tersebut berupa hubungan antara entitas yang saling terkait dengan sistem. Entitas yang saling berhubungan adalah admin dan user, dimana admin dapat mengakses layanan pada sistem sedangakan user hanya dapat mengakses menu layanan user yang disediakan. Diagram konteks dapat dilihat pada Gambar 4.2.



**Gambar 4.3** Diagram Konteks

* + - 1. **Diagram Jenjang (Level 0)**

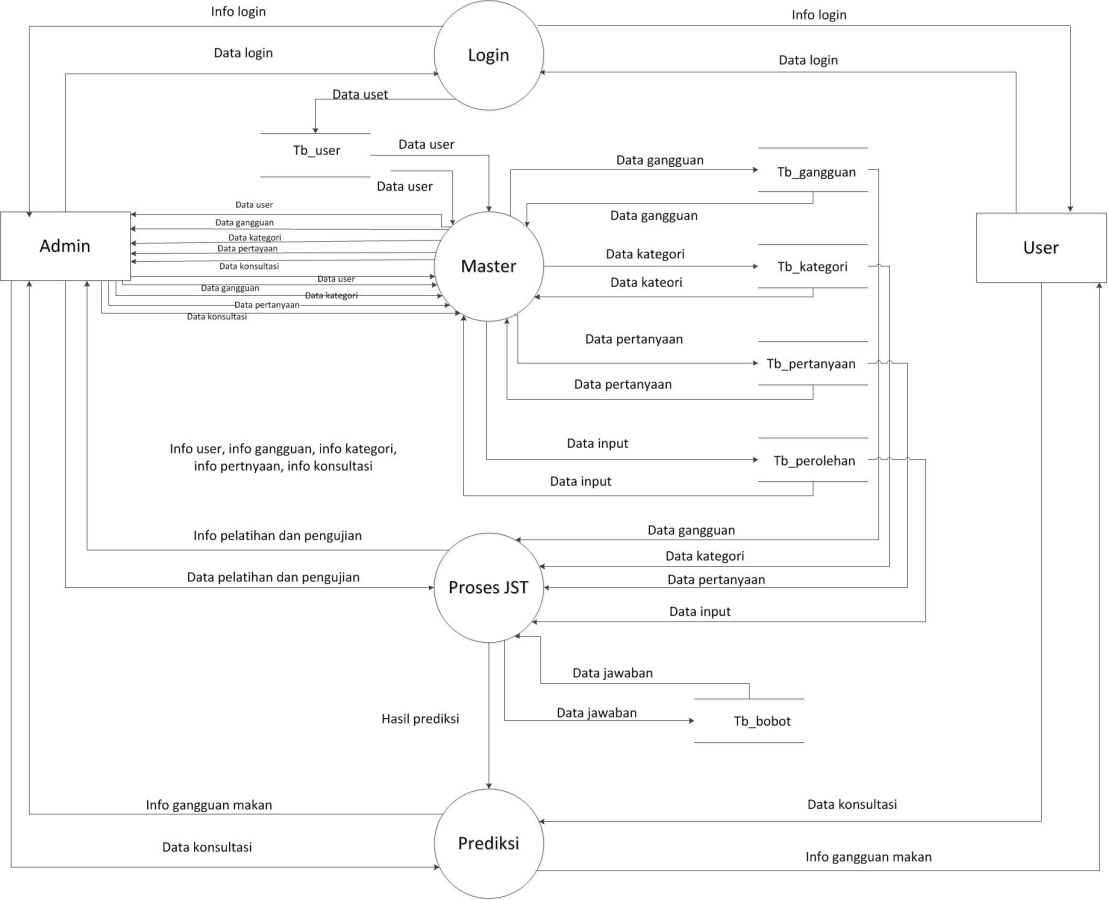
Diagram Jenjang adalah penjabaran menjelaskan beberapa proses yang terjadi pada sistem pendeteksi kecenderugan gangguan makan. Proses yang ada dalam sistem meliputi proses login, proses mastes data, proses pelatihan, proses pengujian, proses deteksi. Diagram jenjang dapat dilihat pada Gambar 4.3.



**Gambar 4.3** Diagram Jenjang

* + - 1. **Diagram Alir Data Level 1**

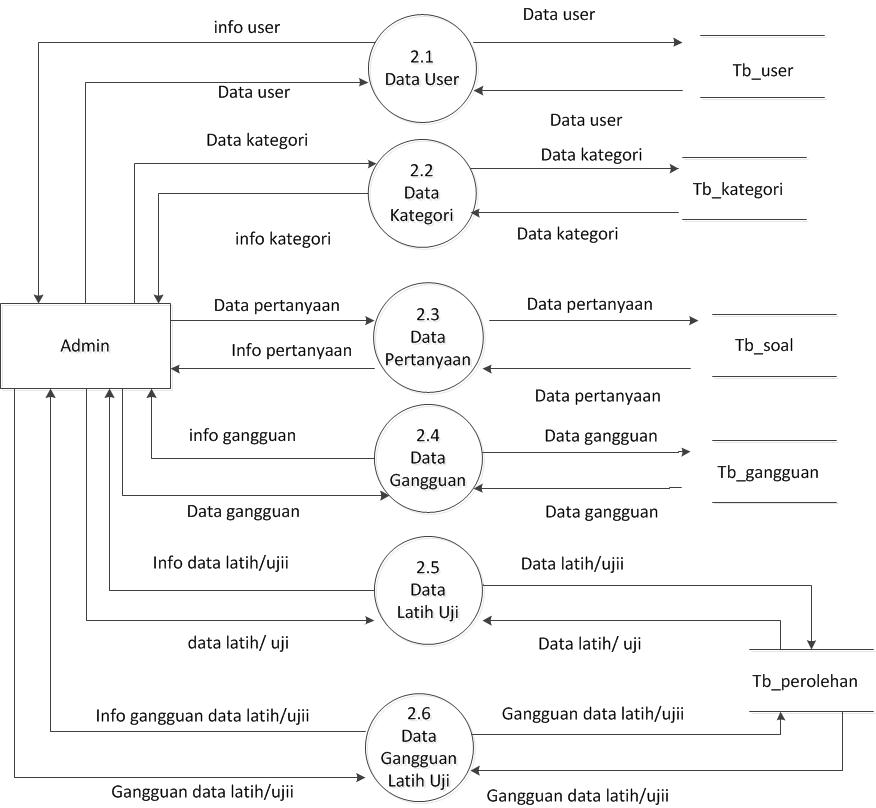
Diagram arus data level 1 merupakan gambaran detail sistem implementasi JST untuk mendeteksi kecenderungan gangguan makan pada remaja menggunakan metode *backpropagation* yang akan dibangun. Pada DAD Level1 juga meruppakan penjabaran dari Diagram Jenjang, entitas yang terkait antara lain entitas admin, entitas user, selain itu proses yang terjadi adalah proses login, master data, pelatihan, pengujian, dan presiksi. DAD Level 1 dapat dilihat pada Gambar 4.4



**Gambar 4.4** Diagram Alir Data Level1

* + - 1. **Diagram Alir Data Level 2 Proses 2**

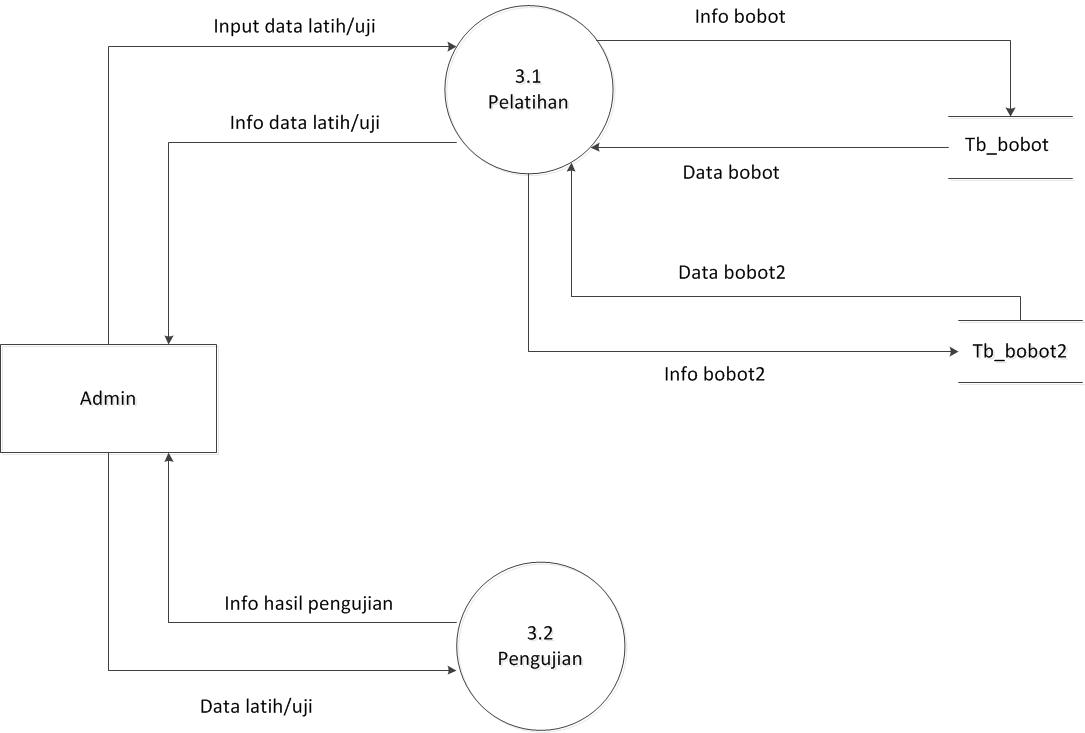
Dalam diagram rinci level 2 proses 2 ini merupakan uraian dari proses master seperti proses input data user, data kategori, data pertanyaan data gangguan serta data latih uji. Rancangan DFD pada diagram rinci level 2 proses 2 sistem implementasi JST untuk mendeteksi kecenderungan gangguan makan pada remaja menggunakan metode *backpropagation* tersebut dapat dilihat pada Gambar 4.5



**Gambar 4.5** Diagram Alir Data Level 2 Proses 2

* + - 1. **Diagram Alir Data Level 2 Proses 3**

Dalam diagram rinci level 2 proses 2 ini merupakan uraian dari proses JST seperti berupa proses pelatihan dan pengujian data. Rancangan DFD pada diagram rinci level 2 proses 3 sistem implementasi JST untuk mendeteksi kecenderungan gangguan makan pada remaja menggunakan metode *backpropagation* tersebut dapat dilihat pada Gambar 4.6



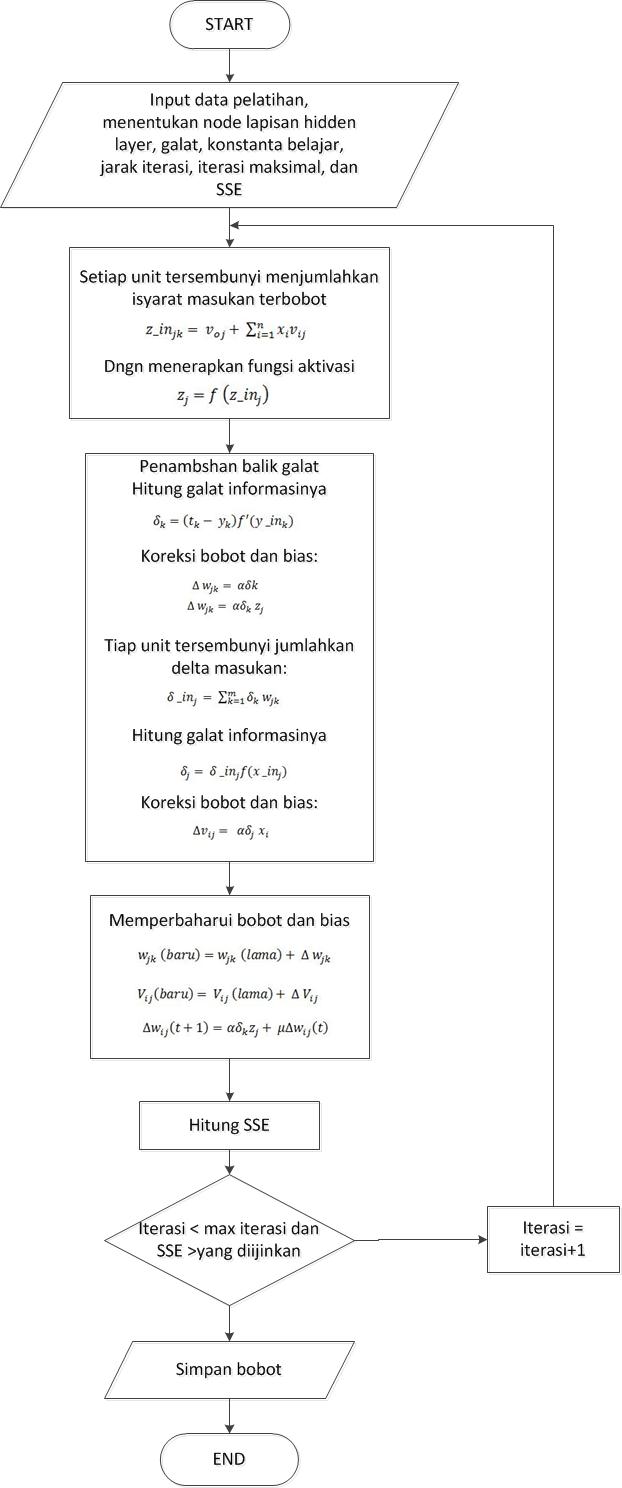
**Gambar 4.6** Diagram Alir Data Level 2 Proses 3

* + 1. **Flowchart Sistem**

*Flowchart* pada perancangan sistem yang dibangun yaitu saat proses pelatihan data, pengujian data, dan deteksi gangguan makan, dimulai dari proses pengisian kuisioner hingga mendapatkan parameter untuk dijadikan data pelatihan. Proses pelatihan jaringan saraf tiruan dan pengujian jaringan saraf tiruan tersebut menggunakan metode *backpropagation*. Proses pelatihan jaringan saraf tiruan digunakan untuk mendapatkan bobot yang terbaik untuk digunakan pada proses pengujian. Selanjutnya pada proses pengujian digunakan untuk akurasu dari bobot yang di dapatkan pada proses pelatihan jaringan saraf tiruan.

* + - 1. **Flowchart Pelatihan Data**

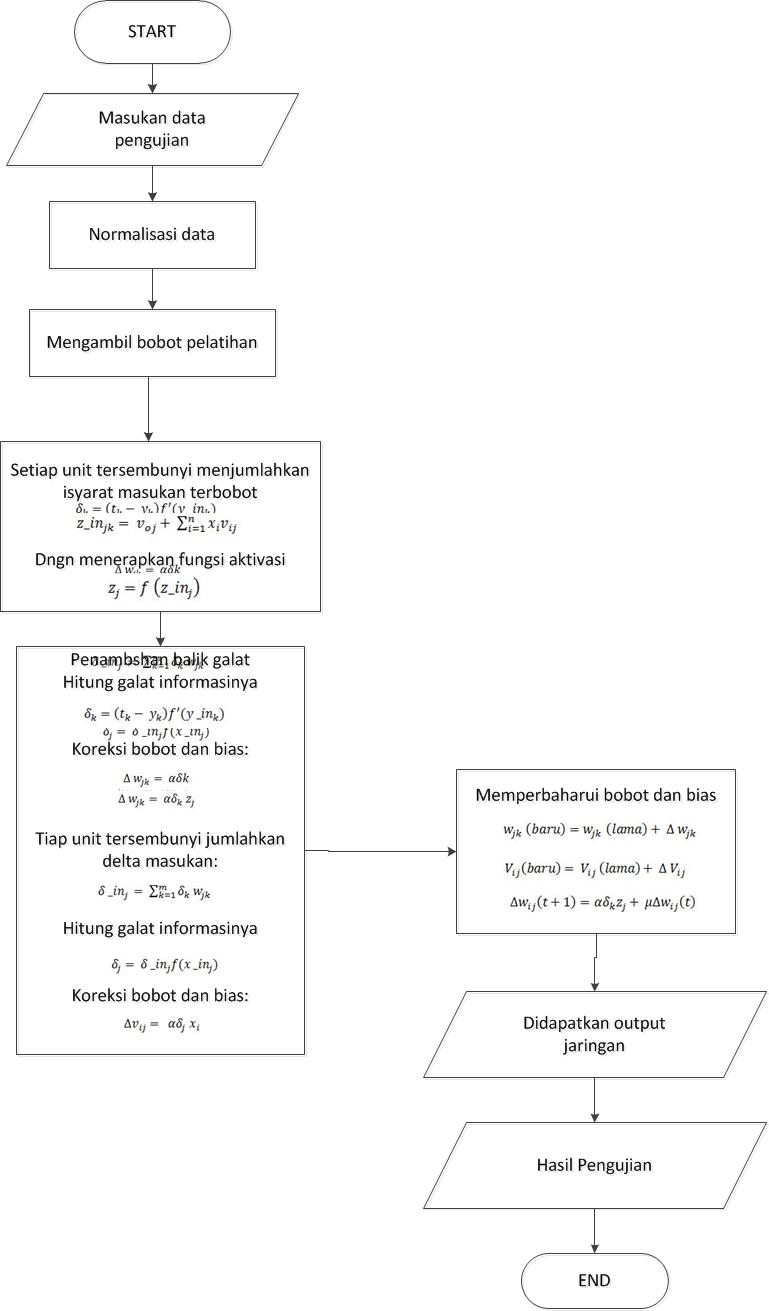
Pada *flowchart* pelatihan data ini merupakan proses untuk mendapatkan bobot yang sesuai. Bobot tersebut akan digunakan sebagai proses pengujian data. Rancangan *Flowchart* pelatihan data pada sistem implementasi JST untuk mendeeteksi kecenderungan gangguan makan pada remaja menggunakan metode *backpropagation* tersebut dapat dilihat pada Gambar 4.7.



**Gambar 4.7** *Flowchart* Pelatihan Data

* + - 1. **Flowchart Pengujian Data**

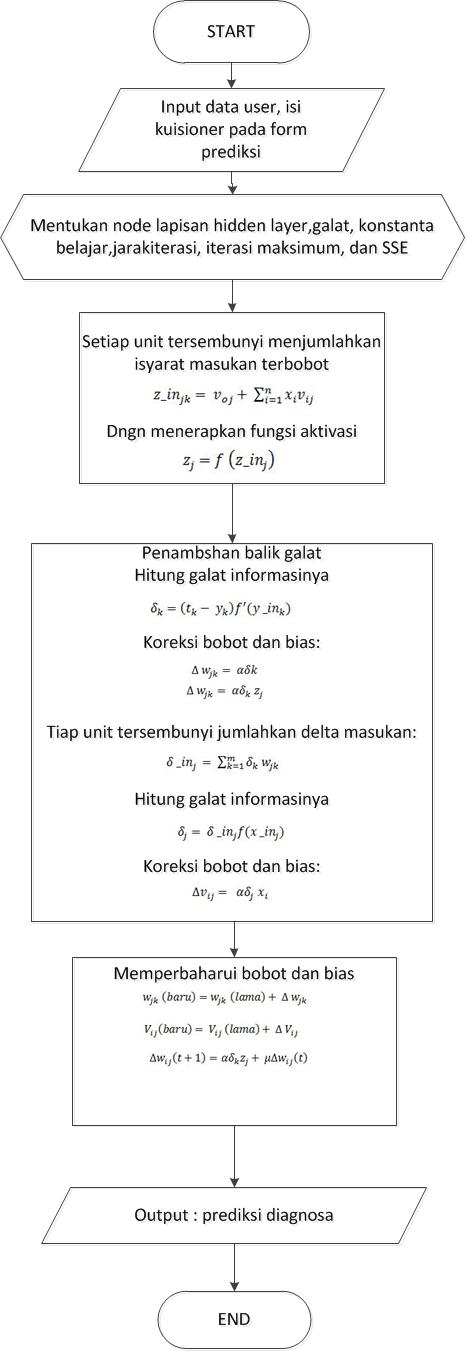
Pada *flowchart* pengujian data ini merupakan proses menguji data baru yang mana belum pernah dilatih sebelumnya. Rancangan *Flowchart* Pengujian data pada sistem implementasi JST untuk mendeeteksi kecenderungan gangguan makan pada remaja menggunakan metode *backpropagation* tersebut dapat dilihat pada Gambar 4.8.



**Gambar 4.8** *Flowchart* Pengujian Data

* + - 1. **Flowchart Deteksi Penyakit**

Pada *flowchart* prediksi ini merupakan proses untuk memberikan informasi berupa prediksi dimana *user* mendapatkan hasil gangguan makan yang diderita menurut hasil diagnosa awal sesuai dengan kuisione yang sudah diinputkan. Rancangan *Flowchart* Pengujian data pada sistem implementasi JST untuk mendeeteksi kecenderungan gangguan makan pada remaja menggunakan metode *backpropagation* tersebut dapat dilihat pada Gambar 4.9.



**Gambar 4.9** *Flowchart* Prediksi Kecenderungan Gangguan Makan

* + 1. ***Entity Relational Diagram* (ERD)**

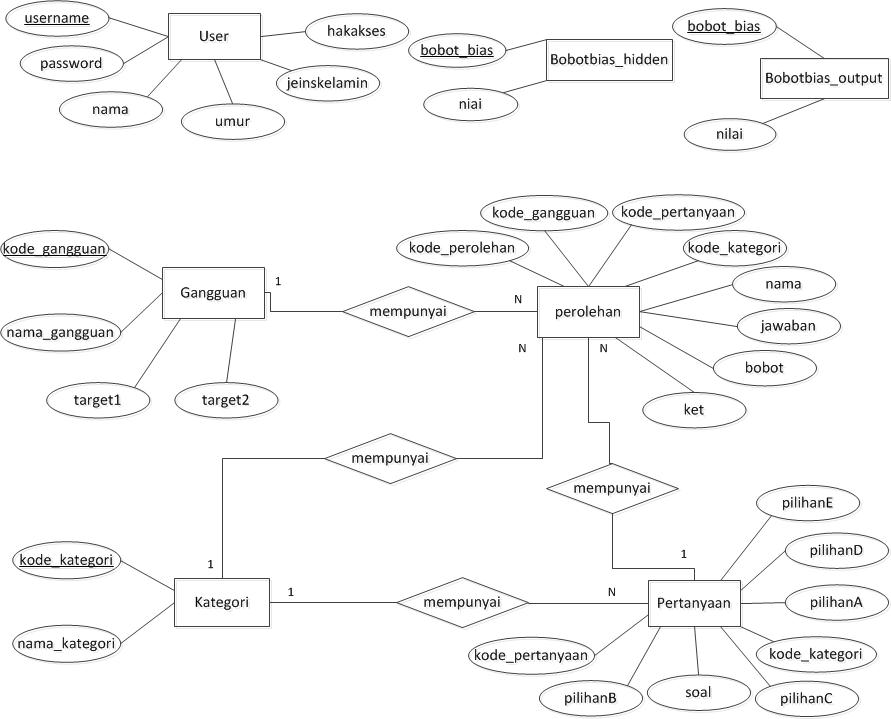
Aplikasi yang dibuat akan menerima *input* yang terdiri data kategori, data penyakit, data soal, dan data user. Kemudian, data akan dilatih dan diuji dengan metode *backpropagation* untuk mendapatkan hasil yang maksimal. Sistem tersebut akan menyimpan data-data tersebut dalam sebuah *database* dan akan diproses sesuai dengan aturan pada metode *backpropagation*, proses *output* yang dihasilkan pada aplikasi tersebut berupa deteksi awal gangguan makan.

1. Identifikasi Entitas

Entitas-entitas yang terlibat dalam pengolahan data diagnosa awal penyakit kepala yaitu sebaga berikut:

1. Entitas User
2. Entitas Kategori
3. Entitas Pertanyaan
4. Entitas Perolehan
5. Entitas Gangguan
6. Entitas Bobot Hidden
7. Entitas Bobot Output
8. Relasi Antar Entitas

Relasi antar entitas merupakan hubungan antara sejumlah entitas yang berasal dari himpunan yang berbeda. *Entity Relationalship Diagram* (ERD) pada sistem implementasi JST untuk mendeteksi kecenderungan gangguan makan pada remaja mempunyai alur seperti pada Gambar 4.10.



**Gambar 4.10** *Entity Relational Diagram* (ERD)

* + 1. **Struktur Database**

Rancangan struktur table yang digunakan dalam sistem implementasi JST untuk mendeeteksi kecenderungan gangguan makan pada remaja adalah sebagai berikut:

1. Tabel User
2. Tebel Kategori
3. Tebel Pertanyaan
4. Tabel Gangguan
5. Tebel Perolehan
6. Table Bobot hidden
7. Table Bobot output

Detail informasi dari table-tabel tersebut antara lain sebagai berikut:

1. Tabel User

Nama tabel : user

Primary key : username

Foreign key :

Deskripsi tabel : tabel ini digunakan untuk menyimpan data user serta hak aksesya.

Rancangan struktur tabel user dapat dilihat pada Tabel 4.1

**Tabel 4.1** Tabel User

|  |  |  |  |
| --- | --- | --- | --- |
| Kolom | Tipedata | Field Size | Keterangan |
| Username | varchar | 25 | Primary Key |
| Password | varchar | 10 |  |
| nama\_user | varchar | 25 |  |
| Umur | int | 11 |  |
| Jeniskelamin | varchar | 10 |  |
| hak\_akses | varchar | 10 |  |

1. Tabel Kategori

Nama tabel : kategori

Primary key : kode\_kategori

Foreign key :

Deskripsi tabel : tabel ini digunakan untuk menyimpan kategori-kategori dari soal-soal yang di jadikan kuisioner.

Rancangan struktur tabel kategori dapat dilihat pada Tabel 4.2

**Tabel 4.2** Tabel kategori

|  |  |  |  |
| --- | --- | --- | --- |
| Kolom | Tipe Data | Field Size | Keterangan |
| kode\_kategori | varchar | 10 | Primary Key |
| nama\_kategori | varchar | 30 |  |

1. Tabel pertanyaan

Nama tabel : pertanyaan

Primary key : kode\_pertanyaan

Foreign key : kode\_kategori

Deskripsi tabel : tabel ini digunakan untuk menyimpan pertanyaan serta pilihan jawabannya

Rancangan struktur tabel pertanyaan dapat dilihat pada Tabel 4.3

**Tabel 4.3** Tabel Pertanyaan

|  |  |  |  |
| --- | --- | --- | --- |
| Kolom | Tipe Data | Field Size | Keterangan |
| kode\_pertanyaan | int | 10 | Primary Key |
| kode\_kategori | varchar | 10 | Foreign Key |
| Pertanyaan | varchar | 255 |  |
| pilihanA | varchar | 20 |  |
| pilihanB | varchar | 20 |  |
| pilihanC | varchar | 20 |  |
| pilihanD | varchar | 20 |  |
| pilihanE | varchar | 20 |  |

1. Tabel Perolehan

Nama tabel : perolehan

Primary key : kode\_perolehan, kode\_gangguan

Foreign key : username

Deskripsi tabel : tabel ini digunakan untuk menyimpan skor perolehan yang sudah dijawab oleh user.

Rancangan struktur tabel perolehan dapat dilihat pada Tabel 4.4.

**Tabel 4.4** Tabel Perolehan

|  |  |  |  |
| --- | --- | --- | --- |
| Kolom | Tipe Data | Field Size | Keterangan |
| kode\_perolehan | int | 10 | Primary Key |
| Nama | varchar | 10 |  |
| kode\_pertanyaan | int | 30 | Foreign Key |
| kode\_gangguan | varchar | 10 | Foreign Key |
| kode\_kategori | varchar | 30 | Foreign Key |
| Jawaban | varchar | 50 |  |
| Bobot | double | - |  |
| Ket | varchar | 30 |  |

1. Table gangguan

Nama tabel : gangguan

Primary key : kode\_gagguan

Foreign key : -

Deskripsi tabel : tabel ini digunakan untuk menyimpan data jenis gangguan makan.

Rancangan struktur tabel gangguan dapat dilihat pada Tabel 4.6

**Tabel 4.5** Tabel gangguan

|  |  |  |  |
| --- | --- | --- | --- |
| Kolom | Tipe Data | Field Size | Keterangan |
| kode\_gangguan | varchar | 10 | Primary Key |
| nama\_gangguan | varchar | 30 |  |
| target1 | Int | 1 |  |
| target2 | Int | 1 |  |

1. Table Bobot hidden

Nama tabel : bobotbias\_hidden

Primary key : bobot\_bias

Foreign key : -

Deskripsi tabel : tabel ini digunakan untuk menyimpan bobot hasil perhitungan dari input ke hidden.

Rancangan struktur tabel bobot hidden dapat dilihat pada Tabel 4.6

**Tabel 4.6** Tabel bobot hidden

|  |  |  |  |
| --- | --- | --- | --- |
| Kolom | Tipe Data | Field Size | Keterangan |
| bobot\_bias | varchar | 30 | Primary Key |
| Nilai | double |  |  |

1. Table Bobot Output

Nama tabel : bobotbias\_output

Primary key : bobot\_bias

Foreign key : -

Deskripsi tabel : tabel ini digunakan untuk menyimpan bobot hasil perhitungan dari hidden ke output.

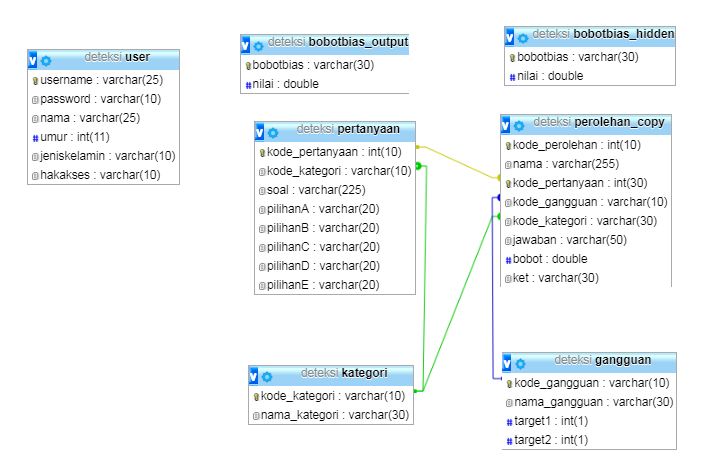
Rancangan struktur tabel bobot output dapat dilihat pada Tabel 4.6

**Tabel 4.7** Tabel bobot output

|  |  |  |  |
| --- | --- | --- | --- |
| Kolom | Tipe Data | Field Size | Kategori |
| bobot\_bias | varchar | 30 | Primary Key |
| Nilai | double |  |  |

1. Diagram Relasi Tabel

Diagram relasi tabel menggambarkan hubungan antar tabel dengan melalui perantara *primary key* pada tabel dengan *foreign key* yang berada pada tabel lain dalam 1 basis data dijelaskan melalui Gambar 4.11 Diagram Relasi Tabel.



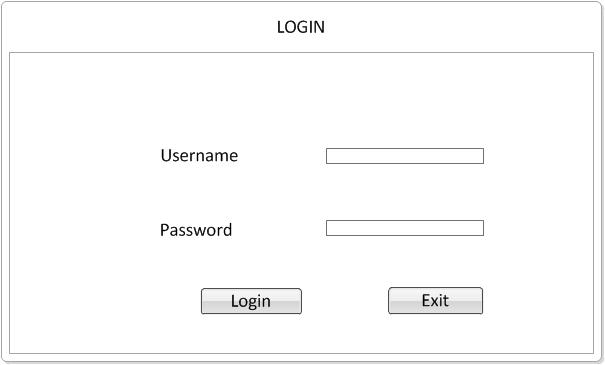
**Gambar 4.11** Diagram Relasi Tabel

* + 1. **Perancangan Antar Muka**

Perancangan interface digunakan untuk memasukkan data kedalam *database*, memproses data, serta menghasilkan *output* prediksi diagnosa awal.

* + - 1. **Perancangan Form Login**

Form login ini berfungsi untuk melindungi sistem. Antarmuka untuk form login ini terdapat inputan *username,* inputan *password,* tombol untuk melakukan *login* yang berguna untuk mengaktifkan menu yang tersedia dalam menu utama dan tombol cancel yang berguna untuk melakukan pembatalan *login* dan keluar dari aplikasi. Desain antarmuka *login* dapat dilihat pada gambar 4.12.

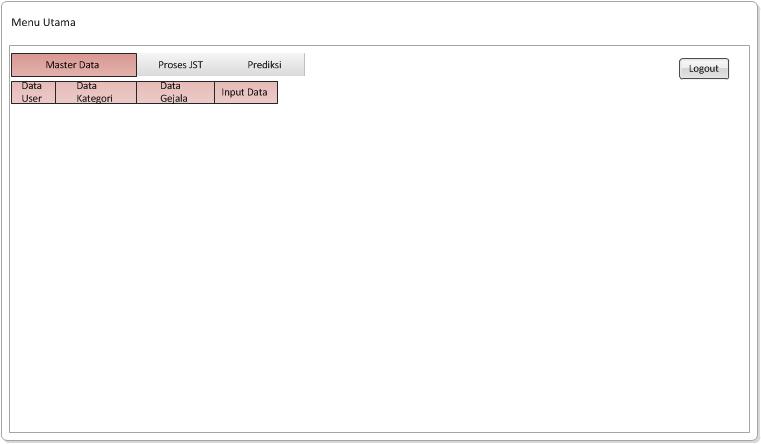


**Gambar 4.12** Perancangan Form Login

* + - 1. **Perancangan Form Menu Utama**

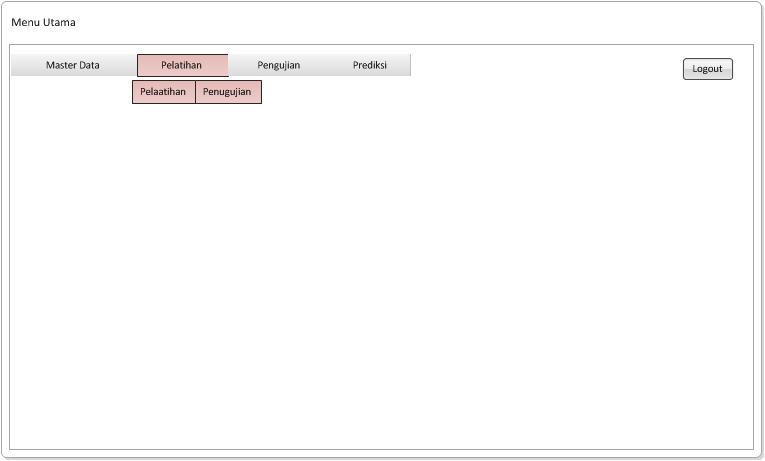
Antarmuka pada form ini berisi menu-menu yang terletak dalam menu utama yang dikelompokkan menggunakan komponen atau icon. Apabila di klik akan membuka sebmenu yang tersedia. Menu yang ditampilkan sesuai hak aksesyang diberikan.

Pada menu master data terdapat 3 submenu yang tersedia, yakni data user, data kategori, dan data soal. Apabila di klik akan membuka form yang dipilih. Antarmuka Menu Master Data terlihat pada Gambar 4.13.



**Gambar 4.13** Perancangan Menu Utama pada Menu Master

Pada proses JST berisi sub menu pelatihan dan pengujian data. Antarmuka menu proses JST dapat dilihat pada gambar 4.14.



**Gambar 4.14** Antarmuka Proses JST

Pada menu prediksi, apabila menu ini di klik akan menyediakan form untuk menjawab kuisioner berupa tes potensi akademik yang nantinya akan menyediakan informasi prediksi program studi yang terpilih. Antarmuka menu prediksi dapat dilihat pada gambar 4.15.

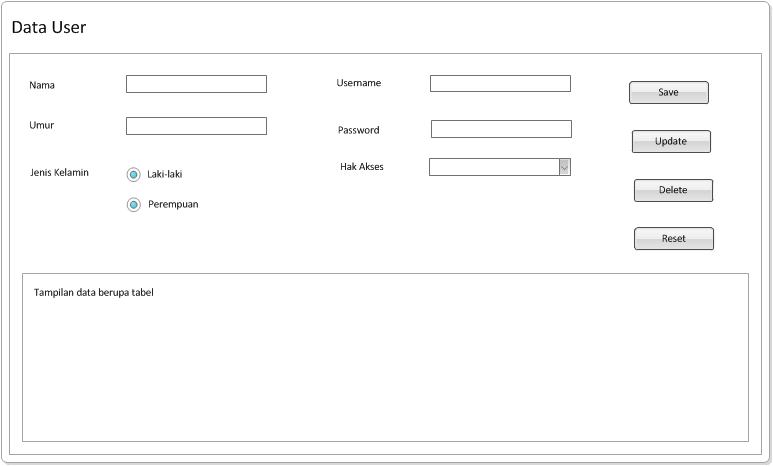


**Gambar 4.15** Antarmuka Menu Prediksi

Pada menu logout, apabila menu ini di klik akan keluar dari menu utama dan kembali ke antarmuka login. Untuk antarmuka menu logout dapat dilihat pada gambar 4.15.

* + - 1. **Perancangan Form User**

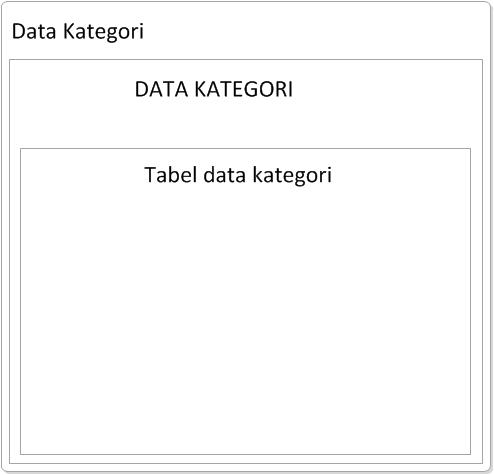
Perancangan form user berfungsi untuk menambahkan data user agar dapat menggunakan sistem ini dengan *username* dan *password* dapat diisi bebas dan hak akses dapat memilih antara *admin* dan *user*. Halaman data *user* dapat dilihat pada gambar 4.16.



**Gambar 4.16** Perancangan Halaman User

* + - 1. **Perancangan Form Kategori**

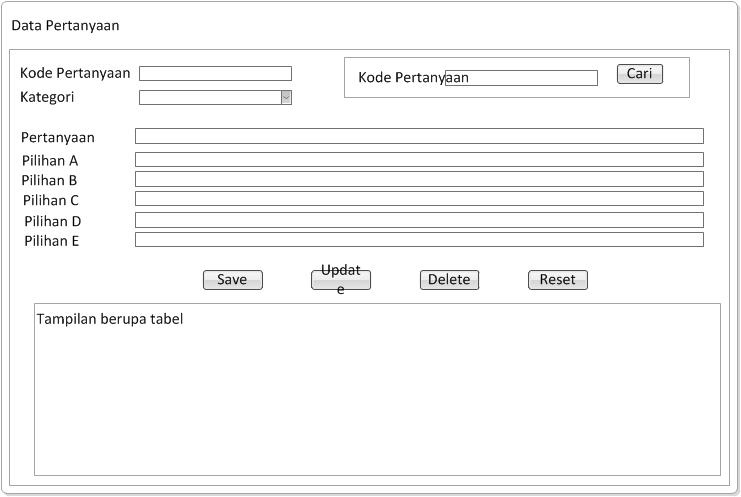
Perancangan form kategori berfungsi untuk menampilkan data kategori ke dalam *database* Halaman data kategoridapat dilihat pada gambar 4.17.



**Gambar 4.17** Perancangan Halaman Kategori

* + - 1. **Perancangan Form Soal**

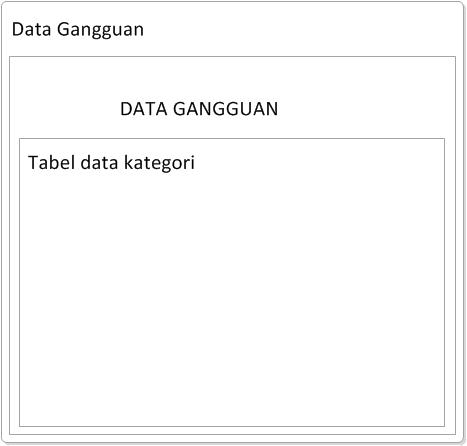
Perancangan form soal berfungsi untuk menambahkan data soal atau kuisioner ke dalam *database*. Pada halaman ini menyediakan pengelolaan data soal serta pilihan jawabannya baik simpan, hapus maupun ubah serta cari. Halaman data soalseperti pada gambar 4.18.



**Gambar 4.18** Perancangan Halaman Soal

* + - 1. **Perancangan Form Data Gangguan**

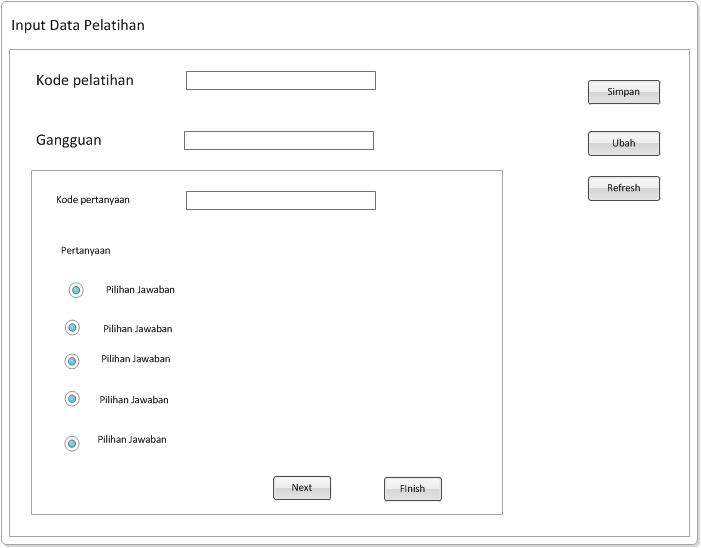
Perancangan form data gangguan berfungsi untuk menampilkan data gangguan ke dalam *database*. Halaman data kategoridapat dilihat pada gambar 4.19.



**Gambar 4.19** Perancangan Halaman Gangguan

* + - 1. **Perancangan Form Input Data Latih**

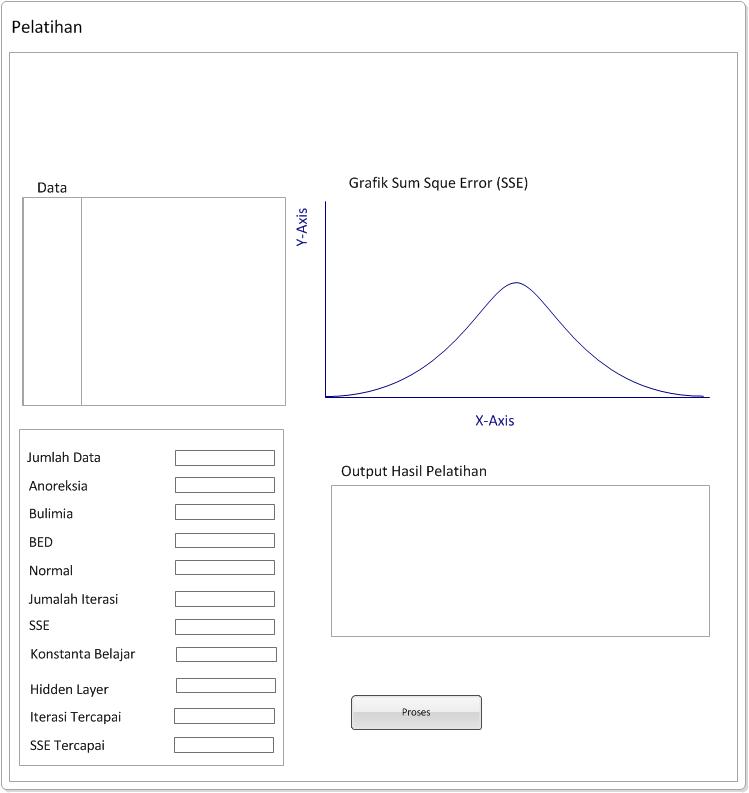
Perancangan form input data latih berfungsi untuk mengisikan datata pelatihan ke dalam *database* kemudian akan diproses untuk acuan *diagnosa* awal gangguan makan yang telah diderita. Halaman inputdata latihdapat dilihat pada gambar 4.20.



**Gambar 4.20** Perancangan Halaman input data latih

* + - 1. **Perancangan Form Pelatihan**

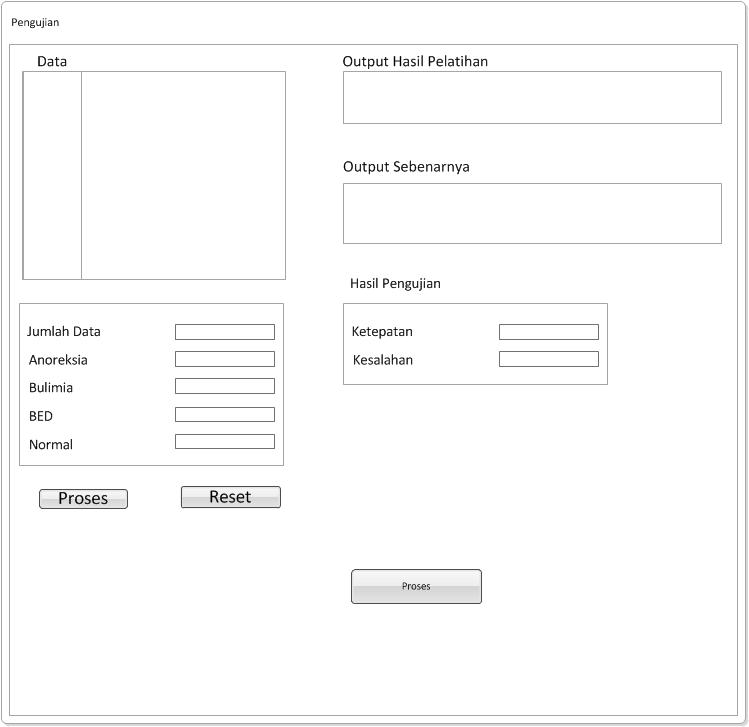
Pada halaman ini menyediakan tampilan kepada admin untuk memproses data latih, yang akan digambarkan di dalam grafik. Perancangan Halaman Pelatihan terlihat pada Gambar 4.21



**Gambar 4.21** Perancangan Halaman Pelatihan

* + - 1. **Perancangan Form Pengujian**

Pada halaman ini menyediakan tampilan kepada admin untuk memproses data uji. Perancangan Halaman Pengujian terlihat pada Gambar 4.22.



**Gambar 4.22** Antarmuka Pengujian tab Proses Pengujian

* + - 1. **Icon Logout**

Dalam tampilan *logout*, *user* disediakan *icon* untuk dapat keluar dari antarmuka utama. Setelah *logout*, *user* akan diarahkan pada antarmuka menu login kembali.

**BAB V**

**IMPLEMENTASI DAN PEMBAHASAN**

* 1. **Implementasi Sistem**

Sistem informasi ini berbentuk aplikasi berbasis *desktop* yang diakses oleh admin dan konsultan. Sistem informasi ini digunakan untuk memanajemen data user dan data konsultasi. Hak akses antara admin dan konsultan berbeda. Hak akses yang dimiliki konsultan adalah hak yang bisa mengakses data user dan data konsultasi, hak akses yang dimiliki admin adalah hak penuh untuk mengakses sistem tersebut.

Sistem informasi ini adalah sebuah aplikasi yang dibangun menggunakan bahasa pemrograman Netbeans IDE 8.0.2. Selain itu, aplikasi ini dalam membangun *database* menggunakan MySQL. Dalam instalasi sistem tersebut, *file* *database* yang sudah dibuat akan disertakan ketika dalam membuat *file exe*, sehingga dalam penerapan pada instansi akan tetap berjalan meskipun tidak memiliki (*Database Management Systems*) DBMS yang digunakan dalam membuat aplikasi tersebut*.* Jika akan melakukan instalasi langkah pertama yang harus dilakukan adalah *mengcopy file exe* yang sudah dibuat dengan satu folder dengan *database* kedalam komputer. Selanjutnya adalah proses instalasi, dengan mengikuti petunjuk–petunjuk yang sudah terdapat pada proses instalasi tersebut. Instalasi dapat dilakukan apabila komputer memiliki ruang 100Mb minimal untuk hasil yang maksimal.

* 1. **Pembahasan Sistem**

Pembahasan sistem merupakan pembahasan mengenai alur program dari sistem informasi yang dibuat dimana alur program dari penginputan data, proses jaringan saraf tiruan (JST) yang dilakukan, dan hasil *output* yang berupa deteksi kecenderungan gangguan makan pada remaja dari program.

* + 1. **Halaman Login**

Halaman *login* ini digunakan penguna untuk masuk ke dalam sistem dengan memasukkan *username* dan *password* pengguna yang telah terdaftar dalam *database* sistem. Tampilan login bisa dilihat di gambar 5.1.



**Gambar 5.1** Form Login

Pada form ini terdapat dua tombol, yaitu *login* dan *exit*. Tombol login digunakan untuk *login* sistem setelah mengisikan *username* dan *password*. Tombol *exit* digunakan untuk keluar dari sistem. Jika belum memiliki *username* dan *password* maka pilih menu daftar maka akan dilanjutkan ke halaman selanjutnya.

|  |
| --- |
| try {Statement Stm = (Statement)TugasAkhir.getConnection().createStatement();  String Str = "Select \* from user where " + "username='"+ user.getText() +"' and password= '" + pass.getText()+"'";  ResultSet Res = Stm.executeQuery(Str); if (Res.next()) {  JOptionPane.showMessageDialog(this, "Id dan Password Cocok");}else { JOptionPane.showMessageDialog(this, "Id dan Password Tidak Cocok");  clear();user.requestFocus();  }} catch(Exception exc){  System.err.println(exc.getMessage());  JOptionPane.showMessageDialog(null, exc); } |

**Gambar 5.2** Sourcode Perintah Login

Pada gambar 5.2 merupakan sebuah gambaran dari halaman *login* yang terdapat kolom *username* dan *password* untuk memasukkan data login. Tombol login ini digunakan untuk proses *login* agar dapat masuk ke dalam sistem.

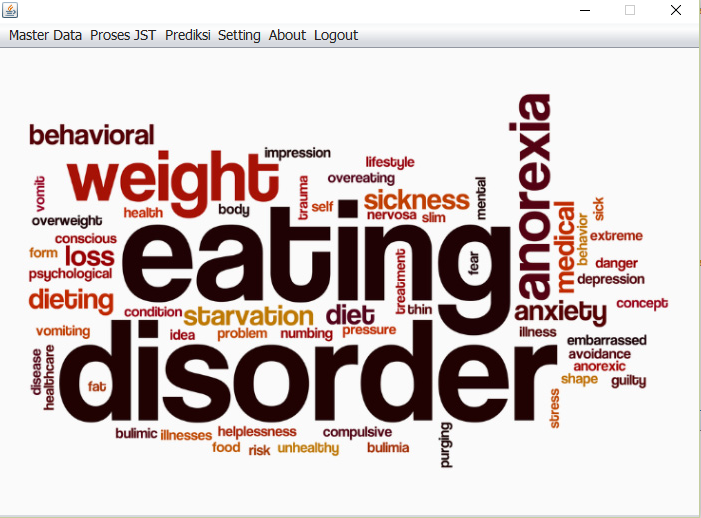
Pada gambar 5.2 merupakan potongan *source code* yang terdapat di halaman login tersebut jika proses login berhasil maka akan muncul kotak dialog yang berisi informasi “Id dan password cocok” dan terdapat *username* pada informasi. Karena, *username* yang terdapat di dalam informasi tersebut diambil dari kombinasi *username* dan *password* yang sesuai dengan data yang ada pada *database*. Kotak dialog tersebut dapat dilihat pada gambar 5.3.



**Gambar 5.3** Informasi Login Berhasil

* + 1. **Halaman Utama**

Halaman utama merupakan halaman yang muncul setelah melakukan login. Halaman ini digunakan untuk tampilan utama untuk admin atau konsultan. Gambar halaman utama tersebut dapat dilihat pada gambar 5. dan potongan source code yang terdapat di halaman menu utama dapat dilihat pada gambar 5.



**Gambar 5.4** Halaman Utama

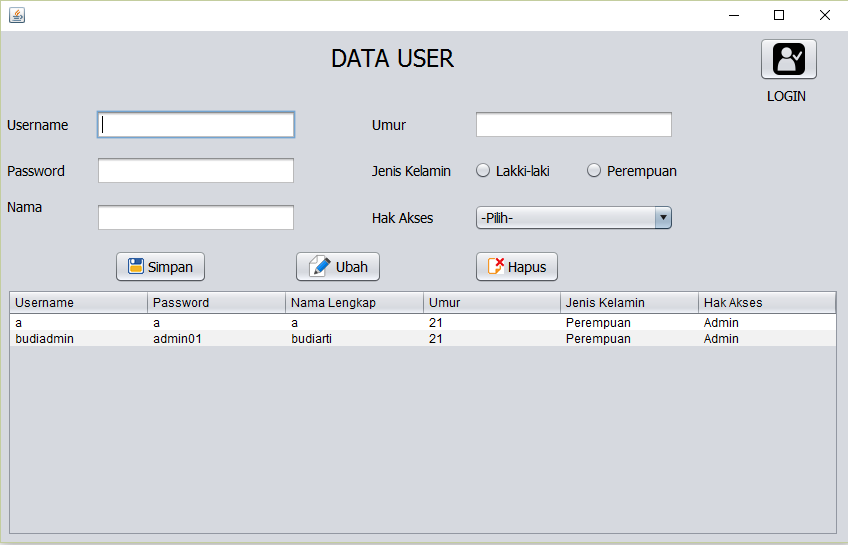
|  |
| --- |
| private void jMenuItem1ActionPerformed(java.awt.event.ActionEvent evt) {  datauser daftar = new datauser();  this.dispose();  daftar.setVisible(true);  }  Private void jMenuItem2ActionPerformed(java.awt.event.ActionEvent evt) {  datakategori kategori = new datakategori();  this.dispose();  kategori.setVisible(true);  }  private void jMenuItem3ActionPerformed(java.awt.event.ActionEvent evt) {  pertanyaan soal = new pertanyaan();  this.dispose();  soal.setVisible(true);  } |

**Gambar 5.5** Potongan Sourcode Halaman Utama

Pada halaman utama ini terdapat menu master yang berisi sub menu data user, gangguan, kategori, pertanyaan, input data pelatihan dan isi gangguan. Pada sub-menu JST terdapat sub-menu pelatihan dan pengujian. Sertta menu prediksi, setting, *about*, dan *logout*.

* + 1. **Halaman Data User**

Pada halaman ini admin dan user dapat dapat melakukan manajemen data user. Halaman ini berfungsi untuk menambah (menyimpan) data, mengubah data, menghapus data user. Gambar menu input data user dapat dilihat pada gambar 5.6 dan potongan source code yang terdapat dihalaman input data user ini dapat dilihat pada gambar 5.6.



**Gambar 5.6** Halaman Data User

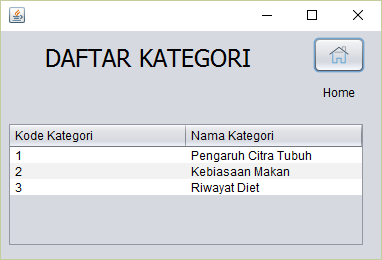
Pada halaman ini tombol simpan untuk menambahkan data, tombol ubah untuk mengubah data, tombol hapus untuk menghpus data serta tombol *login* untuk kembali ke menu *login* dan masuk ke sistem. Data yang harus dimasukan antara lain *username*, *password*, nama lengkap, umur, jenis kelamin, serta setatus pengguna. Berikut merupan potongan *sourcode* pada halaman user.

|  |
| --- |
| String user=username.getText();  String pass=password.getText();  String nam=nama.getText();  String umr=umur.getText();  String hk=(String) hakakses.getSelectedItem();  JenisKelamin();  if(user.equals(""))  {  JOptionPane.showMessageDialog(null, "Pengisian belum lengkap, Isilah data dengan lengkap !");  }else{  Statement stmt=null;  try{  stmt = (Statement)TugasAkhir.getConnection().createStatement();  } catch (SQLException ex) {  Logger.getLogger(datauser.class.getName()).log(Level.SEVERE, null, ex);  }try{  if(stmt.executeUpdate("insert into user values ('"+user+"','"+pass+"','"+nam+"','"+umr+"','"+jk+"','"+hk+"')")!=0) {  JOptionPane.showMessageDialog(null, "Berhasil disimpan");} else {  JOptionPane.showMessageDialog(this, "Gagal simpan");}stmt.close();  } catch (SQLException ex) {  Logger.getLogger(datauser.class.getName()).log(Level.SEVERE, null, ex);  }  clear2();  tampiluser();  }  } |

**Gambar 5.7** Potongan Sourcode Data User

* + 1. **Halaman Daftar Kategori**

Menu daftar data kategori adalah menu yang dibuat untuk memasukkan menampilkan data kategori yang disimpan di database*.* Pada menu button back to home untuk kembali ke menu utama yang digunakan. Menu input data kategori terlihat pada Gambar 5.8.



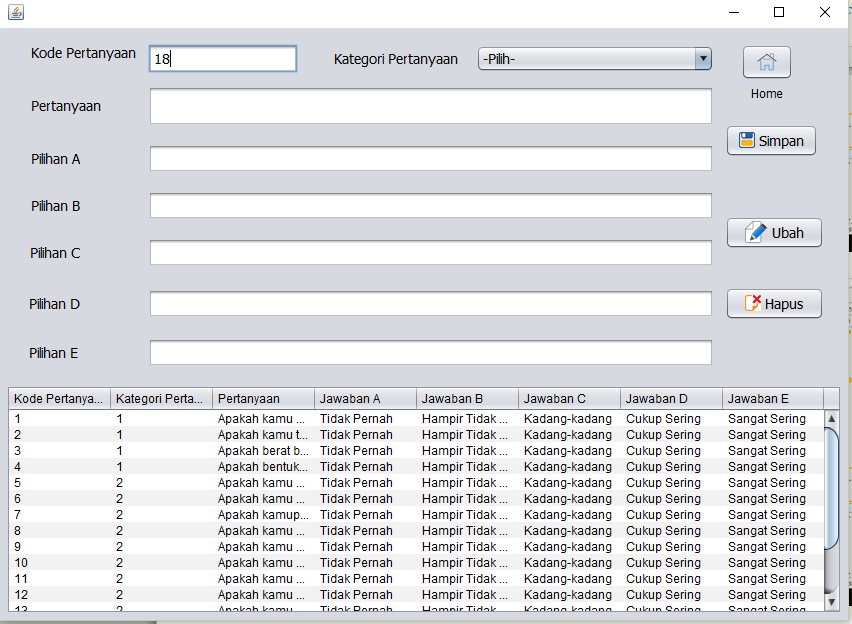
**Gambar 5.8** Halaman Daftar Kategori

|  |
| --- |
| try {  Statement Stm = (Statement)TugasAkhir.getConnection().createStatement();  String sql="select right(kode\_kategori,2) as nama from "  + "kategori order by kode\_kategori desc limit 1";  Rs=Stm.executeQuery(sql);  if(Rs.first()==false){ kode.setText("1");  }else{ Rs.last();  int no=Rs.getInt(1)+1;  String cno=String.valueOf(no);  int pjg\_cno=cno.length();  for (int a=0;a<2-pjg\_cno;a++){cno="0"+cno; }  kode.setText(cno);} }  catch (Exception e) {  System.err.println("Error="+e);  } } |

**Gambar 5.9** Potongan Sourcode Daftar Kategori

* + 1. **Halaman Data Pertanyaan**

Menu input data pertanyaan adalah menu yang dibuat untuk memasukkan data seperti kode soal, kategori, naskah soal, serta pilihan jawaban dari A sampai E yang nanti akan disimpan di dalam *database.* Pada halaman data pertanyaan terdapat 4 button yaitu simpan, ubah, hapus serta back to home untuk kembali ke menu utama yang digunakan untuk manipulasi data pertanyaan baik itu simpan, ubah, dan hapus. Menu input data pertanyaan terlihat pada Gambar 5.10.



**Gambar 5.10** Halaman Data Pertanyaan

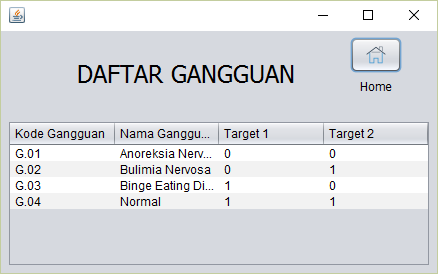
Pada halaman ini tombol simpan untuk menambahkan data, tombol ubah untuk mengubah data, tombol hapus untuk menghpus data serta tombol *home* untuk kembali ke halamanutam. Data yang harus dimasukan antara lain kode pertanyaan, kategori pertanyaan, pertanyaan serta pilihan jawaban yang akan diberikan. Berikut merupan potongan *sourcode* pada halaman pertanyaan.

|  |
| --- |
| String kd1=kodesoal.getText();  String ktg=(String)jComboBox1.getSelectedItem();  String pert=soal.getText();  String ja=jawaba.getText();  String jb=jawabb.getText();  String jc=jawabc.getText();  String jd=jawabd.getText();  String je=jawabe.getText();  if(pert.equals("")||ja.equals("")||jb.equals("")||jc.equals("")||jd.equals("")||je.equals(""))  {  JOptionPane.showMessageDialog(null, "Pengisian belum lengkap, Isilah data dengan lengkap !");  }else{  Statement stmt=null;  try{stmt = (Statement)TugasAkhir.getConnection().createStatement();  } catch (SQLException ex) {  Logger.getLogger(datauser.class.getName()).log(Level.SEVERE, null, ex);  }try{  if(stmt.executeUpdate("insert into pertanyaan values ('"+kd1+"','"+ktg+"','"+pert+"','"+ja+"','"+jb+"','"+jc+"','"+jd+"','"+je+"')")!=0) {  JOptionPane.showMessageDialog(null, "Berhasil disimpan");  } else {  JOptionPane.showMessageDialog(this, "Gagal simpan"); }stmt.close();  } catch (SQLException ex) {  Logger.getLogger(datauser.class.getName()).log(Level.SEVERE, null, ex); }clear();  kodepertanyaan();  tampilpertanyaan();} |

**Gambar 5.11** Potongan Sourcode Data Kategori

* + 1. **Halaman Daftar Gangguan**

Menu daftar gangguan adalah menu yang dibuat untuk menampilkan data gangguan makan dimana terdiri dari kode gangguan, nama gangguan,yang disimpan pada *database*. *.* Pada menu button back to home untuk kembali ke menu utama yang digunakan. Menu input data kategori terlihat pada Gambar 5.12.



**Gambar 5.12** Halaman Daftar Ganggugan

|  |
| --- |
| try {  Statement Stm = (Statement)TugasAkhir.getConnection().createStatement();  String sql="select right(kode\_gangguan,2) as nama from " + "gangguan order by kode\_gangguan desc limit 1";  Rs=Stm.executeQuery(sql);if(Rs.first()==false){  kode1.setText("G."+"01"); } else{  Rs.last();  int no=Rs.getInt(1)+1;  String cno=String.valueOf(no);  int pjg\_cno=cno.length();  for (int a=0;a<2-pjg\_cno;a++){  cno="0"+cno;}kode1.setText("G."+cno);  } } catch (Exception e) {  System.err.println("Error="+e);  } |

**Gambar 5.13** Potongan Sourcode Data Kategori

* + 1. **Halaman Input Data**

Pada halaman ini admin dapat melakukan manajemen data input yang digunakan untuk pelatihan dan pengujian sistem. Halaman ini berfungsi untuk menambah (menyimpan) data. Gambar menu input data pelatihan dan pengujian dapat dilihat pada gambar 5.14.



**Gambar 5.14** Halaman Input Data

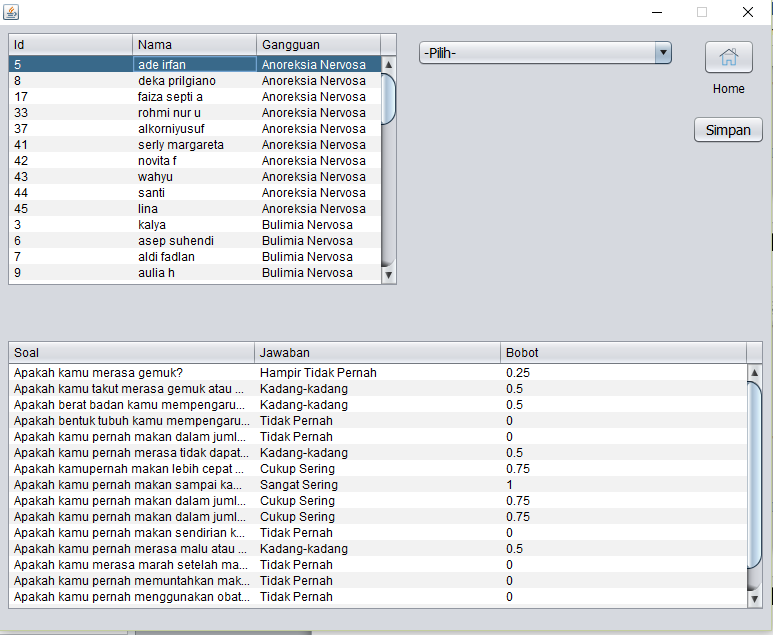
Pada halaman ini data yangharus dimasukan adalah nama, serta jawaban dari masing-masing peserta yang diwawancrai sebagai data pelatihan kemudan tombol simpan digunakan untuk menyimpan data ke dalam *database*. Kemudian tombol home digunakan untuk kembali kehalaman utama. Berikut merupan potongan *sourcode* pada halaman input data.

|  |
| --- |
| public void pilih(){  if(pointer<jumSoal){  //set pilihan ke table  Object[] o = new Object[2];  o[0] = soal[pointer];  o[1] = jawaban[pointer];  tabel.addRow(o);  pointer++;  if(pointer==jumSoal){  jTextArea1.setText("SELESAI");  jLabel1.setText("-");  jTable1.setEnabled(true);  radioDisable();  }else{  jLabel1.setText(""+(pointer+1));  jTextArea1.setText(soal[pointer]);  }  }  if(edit){  jTable1.setValueAt(soal[poin], poin, 0);  jTable1.setValueAt(jawaban[poin], poin, 1);  jTextArea1.setText("SELESAI");  jLabel1.setText("-");  jTable1.setEnabled(true);  radioDisable();  edit = false;  }  buttonGroup1.clearSelection();  } |

**Gambar 5.15** Potongan Sourcode Input Data

* + 1. **Halaman Input Data Gangguan**

Pada halaman ini admin dapat melakukan manajemen data gagguan yang digunakan untuk pelatihan dan pengujian sistem. Halaman ini berfungsi untuk menambah gangguan pada data latih yang telah dimasukan (data remaja yang telah diwawancara). Gambar menu input data gangguan dapat dilihat pada gambar 5.16.

**Gambar 5.16** Halaman Input Gangguan

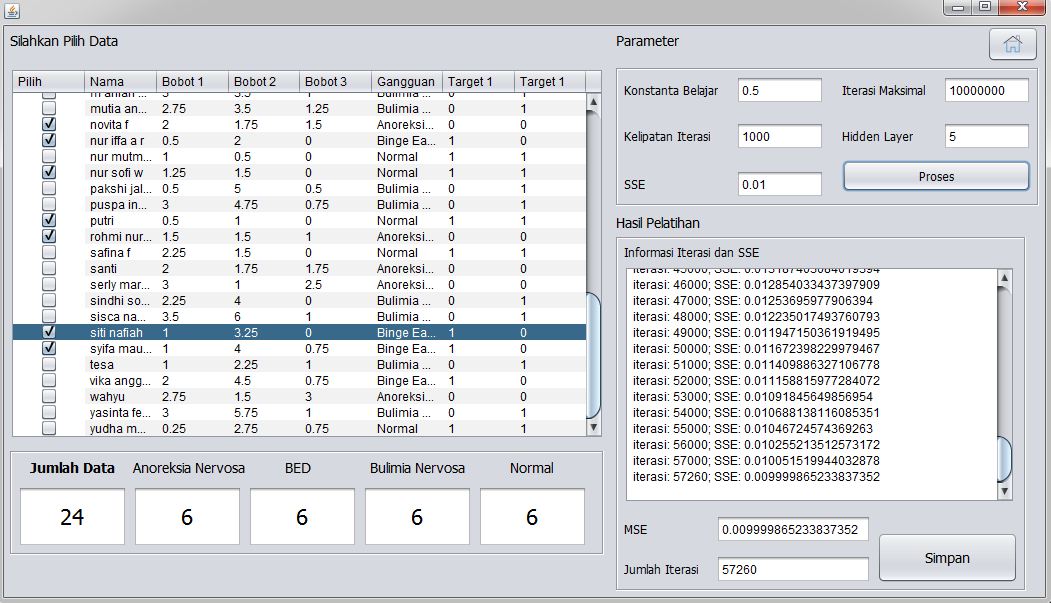
Pada halaman ini terdapat tombol simpan untuk menyimpan data gangguan yang telah diisikan untuk melenkapi data. Hasil pemrosesan tombol tersebut akan berdampak pada *database* tabel perolehan. Berikut merupakan potongan *sourcode* pada halaman input gangguan.

|  |
| --- |
| String kd1=kode1.getText();  String kateg=ggan.getText();  if(kateg.equals("")){  JOptionPane.showMessageDialog(null, "Pengisian belum lengkap, Isilah data dengan lengkap !");  }else{  Statement stmt=null;  try{  stmt = (Statement)TugasAkhir.getConnection().createStatement();  } catch (SQLException ex) { Logger.getLogger(datakategori.class.getName()).log(Level.SEVERE, null, ex);  }try{  if(stmt.executeUpdate("insert into gangguan values ('"  + ""+kode1.getText()+"','"+ggan.getText()+"','"+txtTarget.getText()+")")!=0) {  JOptionPane.showMessageDialog(null, "Berhasil disimpan");  } else {  JOptionPane.showMessageDialog(this, "Gagal simpan");  }  stmt.close();  } catch (SQLException ex) {  Logger.getLogger(datakategori.class.getName()).log(Level.SEVERE, null, ex); }  clear2();  kodegangguan();  tampilgangguan(); } |

**Gambar 5.17** Potongan Sourcode Input Gangguan

* + 1. **Halaman Data Pelatihan**

Pada halaman ini admin dapat melakukan pelatihan data yang ada dalam database pada tabel perolehan dengan beberapa parameter antara lain kontanta belajar, iterasi maksimal, kelipatan iterasi, *hidden layer*, dan SSE. Gambar halaman pelatihan dan pengujian ini dapat dilihat pada gambar 5.18



**Gambar 5.18** Halaman Data Pelatihan

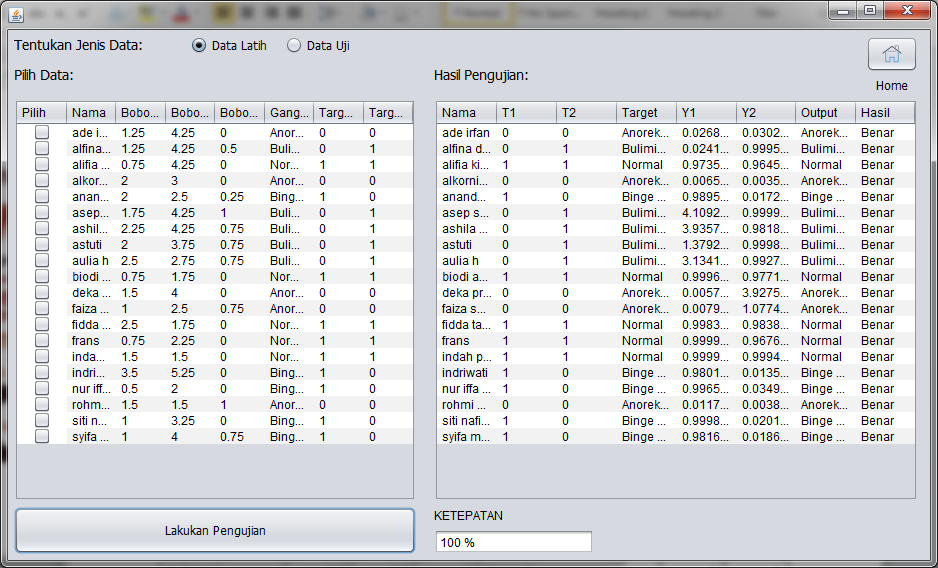
Proses pelatihan data pada gambar 5.18 memasukan konstanta belajar, iterasi maksimal, kelipatan iterasi, *hidden layer* dan SSE . Dipilih data yang akan dilatih, kemudian dilakukan proses pelatihan menggunakan metode *Backpropagation* dengan melakukan klik proses maka didapatkan SSE dan MSE serta terpenuhi pada iterasi ke berapa. Jika pada proses pelatihan belum mencapai iterasi maksimal proses pelatihan data tersebut dinyatakan konvergen. Kemudian simpan bobot kedalam database.

|  |
| --- |
| jTable1.setModel(tabel);  tabel.addColumn("Pilih");tabel.addColumn("Nama"); tabel.addColumn("Bobot 1");tabel.addColumn("Bobot 2");  tabel.addColumn("Bobot 3");tabel.addColumn("Gangguan");  tabel.addColumn("Target 1");tabel.addColumn("Target 1");  tabel.getDataVector().removeAllElements();  tabel.fireTableDataChanged();  try{  Statement s = c.createStatement();  String sql = "SELECT aa.nama AS nama,SUM(aa.bobot1) AS bobot1,SUM(aa.bobot2) AS bobot2,SUM(aa.bobot3) AS bobot3, bb.nama\_gangguan, bb.target1 AS t1, bb.target2 AS t2 " + " FROM (SELECT a.nama,SUM(a.bobot) AS bobot1,0 AS bobot2,0 AS bobot3,a.kode\_gangguan,a.kode\_kategori,a.ket " + " FROM perolehan\_copy AS a"+"WHERE a.kode\_kategori= 1 GROUP BY a.nama " + " UNION ALL " + " SELECT a.nama,0 AS bobot1,SUM(a.bobot) AS bobot2,0 AS bobot3,a.kode\_gangguan,a.kode\_kategori,a.ket " + " FROM perolehan\_copy AS a " + " WHERE a.kode\_kategori=2 GROUP BY a.nama " + " UNION ALL " + " SELECT a.nama,0 AS bobot1,0 AS bobot2,SUM(a.bobot) AS bobot3,a.kode\_gangguan,a.kode\_kategori,a.ket " + " FROM perolehan\_copy AS a " + " WHERE a.kode\_kategori=3 GROUP BY a.nama) AS aa, gangguan AS bb, kategori AS cc " + " WHERE aa.kode\_gangguan=bb.kode\_gangguan AND aa.kode\_kategori=cc.kode\_kategori " + " GROUP BY aa.nama";  ResultSet r = s.executeQuery(sql);  while (r.next()){  Object[] o = new Object[8];  o[0]=false;  o[1]=r.getString("nama");o[2]=r.getString("bobot1");  o[3]=r.getString("bobot2");o[4]=r.getString("bobot3");  o[5]=r.getString("bb.nama\_gangguan");o[6]=r.getString("t1");  o[7]=r.getString("t2");  tabel.addRow(o); } |

**Gambar 5.19** Potongan Sourcode Data Pelatihan

* + 1. **Halaman Data Pengujian**

Pada menu pengujian, admin diminta untuk memilih data sesuai dengan data yang dipilih di menu pelatihan. Pada menu ini menampilkan akurasi pengujian berupa persentase hasil pengujian, yang didapat dari *button* proses. Menu pengujian tab proses pengujian terlihat pada Gambar 5.20.



**Gambar 5.20** Halaman Pengujian Data Terlatih

Pada halaman pengujian bertujuan untuk melakukan pengujian untuk mendapatkan tingkat akurasi jaringan didalam mengenali gangguan. Pilihan data latih digunakan untuk melakukan pengujian data dari hasil pelatihan yang telah dilakukan dan mengetahui tingkat akurasinya. Sedangkan pilihan data uji berfungsi untuk melakukan pengujian data baru yang belum pernah digunakan. Berikut merupakan potongan *sourcode* pada halaman pengujian.

|  |
| --- |
| private void forwardPass(){  final double e = 2.718281828;  double[][] z;  int i,j,k;  double zin,yin;  /\*\* hitung dan cari nilai z \*\*/  zin = 0;  z = new double[jumData][jumHL];  for(i=0;i<jumData;i++){  for(j=0;j<jumHL;j++){  zin = 1\*bv[0][j];  for(k=0;k<jumInput;k++){  zin += ((data[i][k]/max)\*v[k][j]);  }  z[i][j] = 1/(1 + (Math.pow(e, (-(zin)))));  }  }  /\*\* hitung dan cari nilai y \*\*/  yin = 0;  y = new double[jumData][jumOutput];  for(i=0;i<jumData;i++){  for(j=0;j<jumOutput;j++){  yin = 1\*bw[0][j];  for(k=0;k<jumHL;k++){  yin += (z[i][k]\*w[k][j]);  }  y[i][j] = 1/(1 + (Math.pow(e, (-(yin)))));  }  }  } |

**Gambar 5.21** Potongan Sourcode Data Pengujian

Hasil pengujian data terlatih sama dengan target yang diinginkan. Sehingga didapatkan akurasinya yaitu ketepatan 100%. Hasil pengujian data terlatih dapat dilihat pada Tabel 5.1.

**Tabel 5.1** Hasil Pelatihan Data dalam bentuk Jawaban Kuisioner

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Nama Lengkap** | **Umur** | **JK** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** | **12** | **13** | **14** | **15** | **16** | **17** | **Diagnosa**  **Awal** |
| Ade Irfan M | 19 | L | HTP | KD | KD | TP | TP | KD | CS | SS | CS | CS | TP | KD | TP | TP | TP | TP | TP | Anoreksia |
| Alfina Diah | 17 | P | CS | KD | TP | TP | CS | SS | HTP | KD | HTP | KD | HTP | HTP | KD | TP | TP | HTP | HTP | Bullimia |
| Alifia K | 16 | P | TP | HTP | HTP | HTP | SS | CS | KD | CS | KD | CS | TP | TP | TP | TP | TP | TP | TP | Normal |
| Alkorniyusuf | 17 | L | TP | TP | SS | SS | KD | KD | KD | KD | HTP | TP | HTP | KD | TP | TP | TP | TP | TP | Anoreksia |
| Anandari A | 19 | P | KD | HTP | KD | CS | HTP | HTP | HTP | KD | HTP | KD | HTP | TP | HTP | HTP | TP | TP | TP | Binge |
| Asep S | 16 | L | KD | KD | HTP | KD | CS | KD | CS | KD | HTP | KD | KD | HTP | HTP | TP | HTP | KD | HTP | Bullimia |
| Ashila S S | 15 | P | CS | KD | KD | KD | KD | CS | CS | SS | KD | HTP | TP | TP | KD | TP | TP | KD | HTP | Bullimia |
| Astuti | 16 | P | KD | KD | KD | KD | SS | KD | KD | CS | KD | TP | TP | TP | KD | TP | TP | KD | HTP | Bullimia |
| Aulia H | 15 | P | CS | CS | KD | KD | KD | TP | KD | CS | TP | KD | TP | KD | TP | TP | TP | CS | TP | Bullimia |
| Biodi A N | 16 | P | TP | TP | KD | HTP | TP | TP | KD | HTP | HTP | CS | TP | TP | TP | TP | TP | TP | TP | Normal |
| Deka P | 16 | L | TP | KD | TP | SS | SS | SS | TP | SS | TP | SS | TP | TP | TP | TP | TP | TP | TP | Anoreksia |
| Faiza Septi | 19 | P | KD | HTP | TP | HTP | KD | KD | CS | CS | TP | TP | TP | TP | TP | TP | TP | TP | TP | Anoreksia |
| Fidda T D | 19 | P | TP | KD | SS | SS | HTP | TP | KD | KD | KD | TP | TP | TP | TP | TP | TP | TP | TP | Normal |
| Frans B | 19 | L | TP | CS | TP | TP | CS | KD | TP | CS | HTP | TP | TP | TP | TP | TP | TP | TP | TP | Normal |
| Indah P H | 19 | P | KD | KD | TP | KD | TP | KD | TP | KD | TP | KD | TP | TP | TP | TP | TP | TP | TP | Normal |
| Indirawati L | 17 | P | SS | KD | SS | SS | KD | KD | KD | CS | KD | KD | KD | SS | KD | TP | TP | TP | TP | Binge |
| Nur iffa A R | 18 | P | TP | TP | HTP | HTP | KD | KD | HTP | HTP | HTP | HTP | TP | TP | TP | TP | TP | TP | TP | Binge |
| Rohmi Nur | 17 | P | TP | TP | TP | KD | HTP | TP | HTP | HTP | HTP | KD | TP | TP | TP | TP | TP | TP | TP | Anoreksia |
| Siti Nafiah | 17 | P | KD | TP | TP | KD | CS | TP | KD | SS | TP | KD | TP | KD | TP | TP | TP | TP | TP | Binge |
| Syifa M | 16 | P | KD | TP | TP | KD | KD | KD | KD | KD | TP | KD | KD | KD | KD | TP | TP | TP | TP | Binge |

**Tabel 5.2** Data Pelatihan

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Nama** | **Bobot 1** | **Bobot 2** | **Bobot 3** | **Diagnosa** | **Target** | |
| **No.** | **Node 1** | **Node 2** |
| 1. | Ade Irfan | 1.25 | 4.25 | 0 | Anoreksia | 0 | 0 |
| 2. | Alfina Diah A | 1.25 | 4.25 | 0.5 | Bulimia | 0 | 1 |
| 3. | Alifia Kinanti | 0.75 | 4.25 | 0 | Normal | 1 | 1 |
| 4. | Alkorniyusuf | 2 | 3 | 0 | Anoreksia | 0 | 0 |
| 5. | Anandari A P | 2 | 2.5 | 0.25 | Binge | 1 | 0 |
| 6. | Asep Suhendi | 1.75 | 4.25 | 1 | Bulimia | 0 | 1 |
| 7. | Ashila Salma S | 2.25 | 4.25 | 0.75 | Bulimia | 0 | 1 |
| 8. | Astuti | 2 | 3.75 | 0.75 | Bulimia | 0 | 1 |
| 9. | Aulia H | 2.5 | 2.75 | 0.75 | Bulimia | 0 | 1 |
| 10. | Biodi Anggraini N | 0.75 | 1.75 | 0 | Normal | 1 | 1 |
| 11. | Deka Prilgiano | 1.5 | 4 | 0 | Anoreksia | 0 | 0 |
| 12. | Faiza Septi A | 1 | 2.5 | 0.75 | Anoreksia | 0 | 0 |
| 13. | Fidda Tamala | 2.5 | 1.75 | 0 | Normal | 1 | 1 |
| 14. | Frans | 0.75 | 2.25 | 0 | Normal | 1 | 1 |
| 15. | Indah P H | 1.5 | 1.5 | 0 | Normal | 1 | 1 |
| 16. | Indriwati | 3.5 | 5.25 | 0 | Binge | 1 | 0 |
| 17. | Nur Iffa A R | 0.5 | 2 | 0 | Binge | 1 | 0 |
| 18. | Rohmi Nur U | 1.5 | 1.5 | 1 | Anoreksia | 0 | 0 |
|  | **Nama** | **Bobot 1** | **Bobot 2** | **Bobot 3** | **Diagnosa** | **Target** | |
| **No.** | **Node 1** | **Node 2** |
| 19. | Siti Nafiah | 1 | 3.25 | 0 | Binge | 1 | 0 |
| 20. | Syifa Maulidiah | 1 | 4 | 0.75 | Binge | 1 | 0 |

Nilai Bobot berasal dari menjumlahkan nilai dari jawaban per kategori, pembagian kategori bisa dilihat pada bab 3. Jawaban dari kuisioner berupa pilihan diubah ke dalam bentuk angka keudian dijumlahkan. Bobot1 merupakan dari jumlah jawaban nomer 1-4, Bobot2 merupakan jumlah jawaban dari nomor 5-13, dan Bobot3 merupakan jumlah jawaban dari nomor 14-17. Contoh :

**Tabel 5.3** Contoh Hitung Nilai Bobot 1

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Nama** | **Pertanyaan** | | | | | | | | | | | | | | | | |
| **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** | **12** | **13** | **14** | **15** | **16** | **17** |
| Ade Irfan | HTP | KD | KD | TP | TP | KD | CS | SS | CS | CS | TP | KD | TP | TP | TP | TP | TP |
| Ade Irfan | 0,25 | 0,5 | 0,5 | 0 | 0 | 0,5 | 0,75 | 1 | 0,75 | 0,75 | 0 | 0,5 | 0 | 0 | 0 | 0 | 0 |

Bobot1 = 0,25 + 0,5 + 0,5 +0

Bobot1 = 1,25

Bobot2 = 0 + 0,5 + 0,75 + 1 ++ 0,75 + 0,75 + 0 + 0,5 + 0

Bobot2 = 4,25

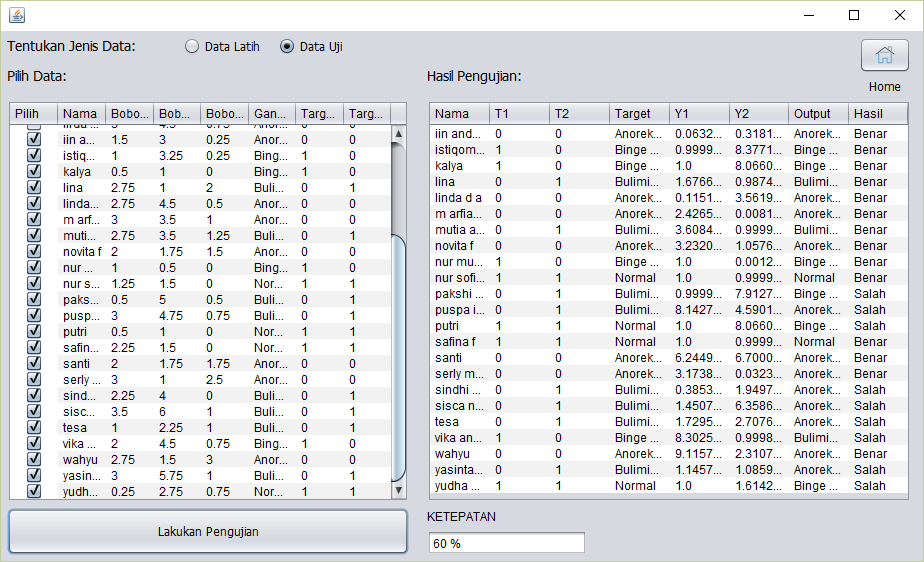
Bobot3 = 0 + 0 + 0 + 0 + 0

Bobot3 = 0

**Tabel 5.4** Hasil Pengujian Data Terlatih

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Nama** | **Y1** | **Y2** | **Output** | **Node 1** | **Node 2** | **Diagnosa awal** | **Keterangan** |
| 1. | Ade Irfan | 0.026819922 | 0.03023115 | Anoreksia | 0 | 0 | Anoreksia | Benar |
| 2. | Alfina Diah A | 0.024133377 | 0.999574546 | Bulimia | 0 | 1 | Bulimia | Benar |
| 3. | Alifia K | 0.9735412 | 0.964529942 | Normal | 1 | 1 | Normal | Benar |
| 4. | Alkorniyusuf | 0.006530778 | 0.003547525 | Anoreksia | 0 | 0 | Anoreksia | Benar |
| 5. | Anandari A P | 0.989550924 | 0.017297422 | Binge | 1 | 0 | Binge | Benar |
| 6. | Asep S | 4.11E-05 | 0.999962429 | Bulimia | 0 | 1 | Bulimia | Benar |
| 7. | Ashila Salma S | 3.94E-05 | 0.981801374 | Bulimia | 0 | 1 | Bulimia | Benar |
| 8. | Astuti | 1.38E-05 | 0.999833956 | Bulimia | 0 | 1 | Bulimia | Benar |
| 9. | Aulia H | 3.13E-04 | 0.992786767 | Bulimia | 0 | 1 | Bulimia | Benar |
| 10. | Biodi Anggraini | 0.999626738 | 0.977157444 | Normal | 1 | 1 | Normal | Benar |
| 11. | Deka Prilgiano | 0.005776145 | 3.93E-04 | Anoreksia | 0 | 0 | Anoreksia | Benar |
| 12. | Faiza Septi A | 0.007949369 | 1.08E-07 | Anoreksia | 0 | 0 | Anoreksia | Benar |
| 13. | Fidda Tamala | 0.998392566 | 0.983889332 | Normal | 1 | 1 | Normal | Benar |
| 14. | Frans | 0.999937349 | 0.96769639 | Normal | 1 | 1 | Normal | Benar |
| 15. | Indah P H | 0.999999507 | 0.999424882 | Normal | 1 | 1 | Normal | Benar |
| 16. | Indriwati | 0.980150538 | 0.013526201 | Binge | 1 | 0 | Binge | Benar |
| 17. | Nur Iffa A R | 0.996584608 | 0.034998738 | Binge | 1 | 0 | Binge | Benar |
| 18. | Rohmi Nur U | 0.01170295 | 0.003820924 | Anoreksia | 0 | 0 | Anoreksia | Benar |
| 19. | Siti Nafiah | 0.999888497 | 0.020113898 | Binge | 1 | 0 | Binge | Benar |
| 20. | Syifa Maulidiah | 0.981611321 | 0.018600929 | Binge | 1 | 0 | Binge | Benar |

Pada menu pengujian proses pengujian data tak terlatih, admin diminta untuk memilih data yang bukan merupakan data yang dilatih pada form pelatihan. Pada menu ini menampilkan akurasi pengujian berupa persentase hasil pengujian, yang didapat dari *button* proses. Menu pengujian proses pengujian data tak terlatih terlihat pada Gambar 5.22



**Gambar 5.22** Halaman Proses Pengujian Data Tak Terlatih

Pada halaman pengujian bertujuan untuk melakukan pengujian untuk mendapatkan tingkat akurasi jaringan didalam mengenali gangguan. Pilihan data latih digunakan untuk melakukan pengujian data dari hasil pelatihan yang telah dilakukan dan mengetahui tingkat akurasinya. Sedangkan pilihan data uji berfungsi untuk melakukan pengujian data baru yang belum pernah digunakan. Berikut merupakan potongan *sourcode* pada halaman pengujian.

|  |
| --- |
| private void dataPengujian(){  int i,j,rowCount; ArrayList list = new ArrayList();  jumData = 0;  //get jumlah data  rowCount = jTable1.getModel().getRowCount();  if(jRadioButton1.isSelected()){  for(i=0;i<rowCount;i++){  jumData = jTable1.getModel().getRowCount();list.add(i);}  }else{ for(i=0;i<rowCount;i++){  if((Boolean)jTable1.getModel().getValueAt(i, 0)==true){  jumData += 1;  list.add(i);  }} }  //get data  data = new double[jumData][jumInput];  target = new int[jumData][jumOutput];  nama = new String[jumData];  gangguan = new String[jumData];  for(i=0;i<jumData;i++){  int a = (Integer)list.get(i);  nama[i] = (String)jTable1.getModel().getValueAt(a,1);  gangguan[i] = (String)jTable1.getModel().getValueAt(a,5);  for(j=0;j<jumInput;j++){  data[i][j] = Double.parseDouble((String)jTable1.getModel().getValueAt(a,j+2))}  target[i][0] = Integer.parseInt((String)jTable1.getModel().getValueAt(a,6));  target[i][1] = Integer.parseInt((String)jTable1.getModel().getValueAt(a,7));  } } |

**Gambar 5.23** Potongan Sourcode Proses Pengujian Data Tak Terlatih

Hasil pengujian data tak terlatih tidak semua data sama dengan target yang diinginkan. Perhitungan prosentase akurasi data didapat dari rumus berikut:

Prosentase = (∑data benar \* 100%)/jumlah data

Prosentase = (13 x 100%)/22

Prosentase = 59%

Sehingga didapatkan akurasinya yaitu ketepatan 59% dan kesalahan 41%. Hasil pengujian data tak terlatih dapat dilihat pada Tabel 5.2.

**Tabel 5.5** Pengujian Data Tak Terlatih dalam bentuk Kuisioner

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Nama Lengkap** | **Umur** | **JK** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** | **12** | **13** | **14** | **15** | **16** | **17** | **Diagnosa Awal** | **Output** | **Ket** |
| Iin Andayani | 19 | P | HTP | KD | KD | HTP | KD | CS | KD | HTP | KD | TP | TP | HTP | HTP | HTP | TP | TP | TP | Anoreksia | Anoreksia | Benar |
| Istiqomah N | 16 | P | KD | KD | TP | TP | KD | KD | CS | KD | HTP | HTP | TP | TP | KD | TP | TP | HTP | TP | Binge | Binge | Benar |
| Kayla nadia | 15 | L | KD | KD | SS | CS | KD | KD | KD | CS | HTP | KD | HTP | HTP | HTP | TP | TP | HTP | CS | Binge | Binge | Benar |
| Lina | 19 | P | CS | CS | KD | CS | HTP | HTP | TP | TP | HTP | TP | TP | HTP | TP | HTP | KD | CS | KD | Bullimia | Bullimia | Benar |
| Linda D A | 19 | P | KD | CS | SS | KD | CS | KD | SS | SS | KD | KD | TP | TP | HTP | KD | TP | TP | TP | Anoreksia | Anoreksia | Benar |
| M. Arfian P | 16 | L | KD | CS | SS | CS | SS | HTP | CS | CS | TP | TP | HTP | HTP | HTP | TP | TP | KD | KD | Anoreksia | Anoreksia | Benar |
| Mutia A | 16 | P | SS | CS | KD | KD | KD | KD | KD | KD | HTP | HTP | TP | KD | KD | TP | KD | KD | HTP | Bullimia | Bullimia | Benar |
| Nur M | 19 | P | KD | KD | TP | TP | TP | TP | TP | KD | TP | TP | TP | TP | TP | TP | TP | TP | TP | Binge | Binge | Benar |
| Nur Sofi | 19 | P | HTP | HTP | HTP | CS | CS | HTP | HTP | KD | TP | TP | TP | TP | TP | RP | TP | TP | TP | Normal | Normal | Benar |
| Pakshi J | 17 | L | KD | TP | TP | TP | SS | SS | SS | SS | KD | HTP | HTP | TP | TP | TP | TP | HTP | TP | Bullimia | Binge | Salah |
| Puspa Indah | 17 | P | SS | SS | KD | KD | KD | CS | CS | CS | KD | HTP | TP | KD | CS | TP | TP | KD | HTP | Bullimia | Anoreksia | Salah |
| Putri A Z. B | 19 | P | TP | TP | TP | KD | TP | TP | KD | KD | TP | TP | TP | TP | TP | TP | TP | TP | TP | Normal | Binge | Salah |
| Safina F | 15 | P | KD | KD | CS | KD | TP | HTP | TP | KD | KD | HTP | TP | TP | TP | TP | TP | TP | TP | Normal | Normal | Benar |
| Santi | 19 | P | HTP | KD | CS | KD | KD | HTP | HTP | HTP | HTP | HTP | TP | TP | TP | HTP | KD | CS | HTP | Aneroksia | Aneroksia | Benar |
| Serly M | 16 | P | CS | SS | CS | KD | HTP | HTP | TP | TP | HTP | TP | TP | TP | HTP | HTP | KD | CS | SS | Aneroksia | Aneroksia | Benar |
| Sindhi S | 19 | P | KD | KD | HTP | SS | CS | KD | KD | CS | KD | KD | KD | HTP | HTP | HTP | TP | TP | HTP | Bullimia | Anoreksia | Salah |
| Sisca Nabila | 17 | P | SS | CS | SS | CS | SS | CS | CS | SS | KD | CS | TP | CS | KD | HTP | TP | HTP | KD | Bulimia | Anoreksia | Salah |
| Tesa | 16 | P | HTP | HTP | HTP | HTP | HTP | HTP | HTP | HTP | HTP | HTP | HTP | HTP | HTP | HTP | HTP | HTP | HTP | Bullimia | Aneroksia | Salah |
| Vika Anggi F | 19 | P | SS | KD | HTP | HTP | KD | KD | CS | KD | CS | CS | CS | TP | TP | CS | TP | TP | TP | Binge | Bullimia | Salah |
| Wahyu | 17 | L | CS | CS | KD | CS | KD | HTP | TP | TP | HTP | HTP | TP | HTP | TP | KD | CS | SS | CS | Anoreksia | Anoreksia | Benar |
| Yasinta F P | 16 | P | SS | CS | KD | CS | KD | CS | CS | CS | KD | KD | KD | KD | SS | TP | KD | HTP | HTP | Bullimia | Anoreksia | Salah |
| Yudha M | 17 | L | TP | TP | TP | HTP | KD | KD | KD | KD | KD | TP | TP | TP | TP | TP | TP | TP | TP | Normal | Binge | Salah |

**Tabel 5.6** Data Pengujian

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **No** | **Nama** | **Bobot 1** | **Bobot 2** | **Bobot 3** | **Diagnosa Awal** | **Target** | |
| **Node 1** | **Node 2** |
| 1. | Iin Andayani | 1.5 | 3 | 0.25 | Anoreksia | 0 | 0 |
| 2. | Istiqomah Mala | 1 | 3.25 | 0.25 | Binge | 1 | 0 |
| 3. | Kalya | 0.5 | 1 | 0 | Binge | 1 | 0 |
| 4. | Lina | 2.75 | 1 | 2 | Bulimia | 0 | 1 |
| 5. | Linda D A | 2.75 | 4.5 | 0.5 | Anoreksia | 0 | 0 |
| 6 | M Arfian Pratama | 3 | 3.5 | 1 | Anoreksia | 0 | 0 |
| 7. | Mutia Angraini | 2.75 | 3.5 | 1.25 | Bulimia | 0 | 1 |
| 8. | Nur Mutmainah | 1 | 0.5 | 0 | Binge | 1 | 0 |
| 9. | Nur Sofi W | 1.25 | 1.5 | 0 | Normal | 1 | 1 |
| 10. | Pakshi Jalasutra | 0.5 | 5 | 0.5 | Bulimia | 0 | 1 |
| 11. | Puspa Indah | 3 | 4.75 | 0.75 | Bulimia | 0 | 1 |
| 12. | Putri | 0.5 | 1 | 0 | Normal | 1 | 1 |
| 13. | Safina F | 2.25 | 1.5 | 0 | Normal | 1 | 1 |
| 14. | Santi | 2 | 1.75 | 1.75 | Anoreksia | 0 | 0 |
| 15. | Serly Margareta | 3 | 1 | 2.5 | Anoreksia | 0 | 0 |
| 16. | Sindhi Soviani | 2.25 | 4 | 0 | Bulimia | 0 | 1 |
| 17. | Sisca Nabila | 3.5 | 6 | 1 | Bulimia | 0 | 1 |
| **No.** | **Nama** | **Bobot 1** | **Bobot 2** | **Bobot 3** | **Diagnosa Awal** | **Target** | |
| **Node 1** | **Node 2** |
| 18. | Tesa | 1 | 2.25 | 1 | Bulimia | 0 | 1 |
| 19. | Vika Anggi F | 2 | 4.5 | 0.75 | Binge | 1 | 0 |
| 20. | Wahyu | 2.75 | 1.5 | 3 | Anoreksia | 0 | 0 |
| 21. | Yasinta Fernanda | 3 | 5.75 | 1 | Bulimia | 0 | 1 |
| 22. | Yudha Mahendra | 0.25 | 2.75 | 0.75 | Normal | 1 | 1 |

Nilai Bobot berasal dari menjumlahkan nilai dari jawaban per kategori, pembagian kategori bisa dilihat pada bab 3. Jawaban dari kuisioner berupa pilihan diubah ke dalam bentuk angka keudian dijumlahkan. Bobot1 merupakan dari jumlah jawaban nomer 1-4, Bobot2 merupakan jumlah jawaban dari nomor 5-13, dan Bobot3 merupakan jumlah jawaban dari nomor 14-17. Contoh :

**Tabel 5.7** Contoh Hitung Nilai Bobot 2

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Nama** | **Pertanyaan** | | | | | | | | | | | | | | | | |
| **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** | **12** | **13** | **14** | **15** | **16** | **17** |
| Iin Andayani | HTP | KD | KD | HTP | KD | CS | KD | HTP | KD | TP | TP | HTP | HTP | HTP | TP | TP | TP |
| Iin Andayani | 0,25 | 0,5 | 0,5 | 0,25 | 0,5 | 0,75 | 0,5 | 0,25 | 0,5 | 0 | 0 | 0,25 | 0,25 | 0,25 | 0 | 0 | 0 |

Bobot1 = 0,25 + 0,5 + 0,5 +0,25

Bobot1 = 1,5

Bobot2 = 0 ,5+ 0,75 + 0,5 + 0,25 + 0,5 + 0 + 0 + 0,25 + 0,25

Bobot2 = 3

Bobot3 = 0,25 + 0 + 0 + 0 + 0

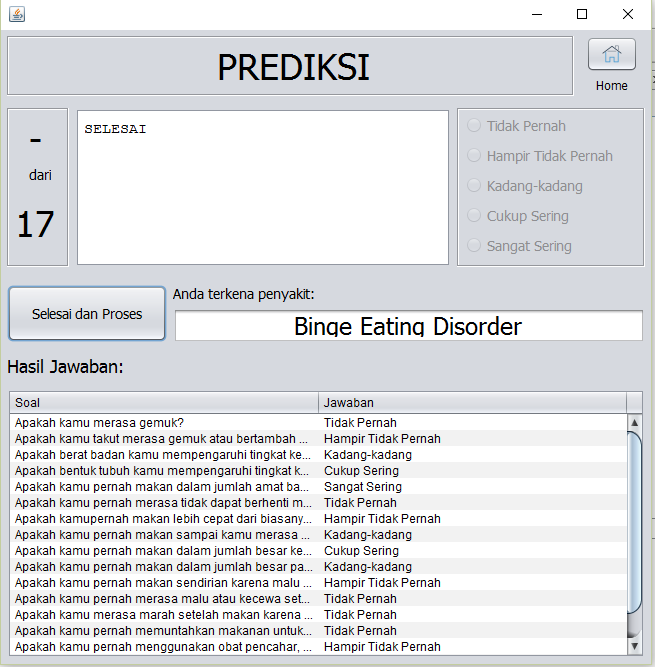
Bobot3 = 0,25

**Tabel 5.8** Hasil Pengujian Data Tak Terlatih

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Nama** | **Y1** | **Y2** | **Output** | **Node 1** | **Node 2** | **Diagnosa Awal** | **Keterangan** |
| 1. | Iin Andayani | 0.99317083 | 0.093218 | Binge | 0 | 0 | Anoreksia | Salah |
| 2. | Istiqomah Mala | 0.99930094 | 0.110594 | Binge | 1 | 0 | Binge | Benar |
| 3. | Kalya | 0.90627044 | 0.02625 | Binge | 1 | 0 | Binge | Benar |
| 4. | Lina | 0.00140202 | 0.006299 | Anoreksia | 0 | 1 | Bulimia | Salah |
| 5. | Linda D A | 0.00985681 | 5.10E-04 | Anoreksia | 0 | 0 | Anoreksia | Benar |
| 6. | M Arfian Pratama | 1.31E-06 | 0.979389 | Bulimia | 0 | 0 | Anoreksia | Salah |
| 7. | Mutia Angraini | 1.01E-07 | 1 | Bulimia | 0 | 1 | Bulimia | Benar |
| 8. | Nur Mutmainah | 0.99516407 | 0.99483 | Normal | 1 | 0 | Binge | Salah |
| 9. | Nur Sofi W | 0.9999976 | 0.999979 | Normal | 1 | 1 | Normal | Benar |
| 10. | Pakshi Jalasutra | 0.98136356 | 0.971977 | Normal | 0 | 1 | Bulimia | Salah |
| 11. | Puspa Indah | 0.01237772 | 9.94E-04 | Anoreksia | 0 | 1 | Bulimia | Salah |
| 12. | Putri | 0.90627044 | 0.02625 | Binge | 1 | 1 | Normal | Salah |
| 13. | Safina F | 0.99999873 | 0.883307 | Normal | 1 | 1 | Normal | Benar |
| 14. | Santi | 1.03E-04 | 2.94E-06 | Anoreksi | 0 | 0 | Anoreksia | Benar |
| 15. | Serly Margareta | 1.13E-04 | 4.02E-05 | Anoreksia | 0 | 0 | Anoreksia | Benar |
| 16. | Sindhi Soviani | 0.00605671 | 2.25E-04 | Anoreksia | 0 | 1 | Bulimia | Salah |
| 17. | Sisca Nabila | 0.84967353 | 2.45E-04 | Binge | 0 | 1 | Bulimia | Salah |
| 18. | Tesa | 1.02E-04 | 9.61E-10 | Anoreksia | 0 | 1 | Bulimia | Salah |
| 20. | Vika Anggi F | 1.31E-04 | 0.999731 | Bulimia | 1 | 0 | Binge | Salah |
| 20. | Wahyu | 1.70E-05 | 5.67E-07 | Anoreksia | 0 | 0 | Anoreksia | Benar |
| 21. | Yasinta Fernanda | 0.46796606 | 0.002128 | Anoreksia | 0 | 1 | Bulimia | Salah |
| 22. | Yudha Mahendra | 9.64E-04 | 6.18E-12 | Anoreksia | 1 | 1 | Normal | Salah |

* + 1. **Halaman Data Prediksi**

Pada tampilan menu prediksi, admin ataupun user dapat memilih jawaban yang dianggap benar. Selain itu, pada menu prediksi menampilkan hasil rekomendasi program studi terpilih yang didapat dari *button finish*, dimana button *finish* akan muncul jika semua kuisioner selesai dijawab. Menu prediksi terlihat pada Gambar 5.24



**Gambar 5.24** Tampilan Form Prediksi

|  |
| --- |
| //assign data  double[] jumBobot = new double[jumInput];  double[][] data = new double[jumData][jumInput];  //menjumlahkan bobot  k=0;  for(i=0;i<jumKategori;i++){j = 0;  jumBobot[i] = 0.0;  while(j<jumPerKategori[i]){  jumBobot[i] += bobot[k]; j++;k++; }  System.out.println("jumBobot["+i+"]: "+jumBobot[i]);  }//konversi data  for(i=0;i<jumKategori;i++){  data[0][i] = jumBobot[i];}  //forward pass  final double e = 2.718281828;  double[][] z; double zin,yin;  System.out.println("max:" +max);  /\*\* hitung dan cari nilai z \*\*/  zin = 0;  z = new double[jumData][jumHL];  for(i=0;i<jumData;i++){  for(j=0;j<jumHL;j++){  zin = 1\*p.bv[0][j];  for(k=0;k<jumInput;k++){  zin += ((data[i][k]/max)\*p.v[k][j]); }  z[i][j] = 1/(1 + (Math.pow(e, (-(zin))))); } }  /\*\* hitung dan cari nilai y \*\*/  yin = 0;  y = new double[jumData][jumOutput];  for(i=0;i<jumData;i++){  for(j=0;j<jumOutput;j++){  yin = 1\*p.bw[0][j];  for(k=0;k<jumHL;k++){  yin += (z[i][k]\*p.w[k][j]);}  y[i][j] = 1/(1 + (Math.pow(e, (-(yin)))));} }} |

**Gambar 5.25** Potongan Sourcode Proses Prediksi

* + 1. **Simulasi Perhitungan Manual**

Simulasi perhitungan manual ini dilakukan dengan tujuan untuk membuktikan perhitungan MSE secara manual dan MSE dari sistem. Data yang digunakan untuk proses ini adalah milih Ade Irfan dengan diagnose awal Anoreksia, datanya adalah sebagai berikut:

**Tabel 5.9** Data Perhitungan Manual

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Pertanyaan** | | | | | | | | | | | | | | | | |
| **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** | **12** | **13** | **14** | **15** | **16** | **17** |
| HTP | KD | KD | TP | TP | KD | CS | SS | CS | CS | TP | KD | TP | TP | TP | TP | TP |
| 0,25 | 0,5 | 0,5 | 0 | 0 | 0,5 | 0,75 | 1 | 0,75 | 0,75 | 0 | 0,5 | 0 | 0 | 0 | 0 | 0 |

**Tabel 5.10** Diagnosa

|  |
| --- |
| **Diagnosa**  **Awal** |
|
| Anoreksia |
| 0 0 |

Konstanta Belajar = 0,5

Iterasi Maksimal = 100

Kelipatan iterasi = 1

Hidden layer = 2

SSE = 0.1

**Tabel 5.11** Bobot *input* ke *hidden*

|  |  |  |
| --- | --- | --- |
|  | Z1 | Z2 |
| X1 | 0.832733798586984 | 0.4694797740121098 |
| X2 | -0.006869906605231 | 0.7489034434760338 |
| X3 | 0.7172998130193451 | 0.4556733161755197 |
| B1 | 0.4195050928952086 | 0.0279705005952891 |

**Tabel 5.12** Bobot *hidden* ke *output*

|  |  |  |
| --- | --- | --- |
|  | Y1 | Y2 |
| Z1 | -0.06444976220578177 | -0.37842522764246456 |
| Z2 | -0.371941138980724 | -0.000353415215472909 |
| B2 | -0.9019764934184369 | -1.0005917545462681 |

Nilai W berasal dari menjumlahkan nilai dari jawaban per kategori, pembagian kategori bisa dilihat pada bab 3. Jawaban dari kuisioner berupa pilihan diubah ke dalam bentuk angka keudian dijumlahkan. W1 merupakan dari jumlah jawaban nomer 1-4, W2 merupakan jumlah jawaban dari nomor 5-13, dan W3 merupakan jumlah jawaban dari nomor 14-17. Kemudian normalisasi data dilakukan dengan cara mencari nilai maksimal dari data kemuadian nilai maksimal tersebut digunakan sebagai pembagi setiap data.

W1 = 0,25 + 0,5 + 0,5 +0

W1 = 1,25

W2 = 0 + 0,5 + 0,75 + 1 + 0,75 + 1 + 0,75 + 0,75 + 0 + 0,5 + 0

W2 = 4,25

W3 = 0 + 0 + 0 + 0 + 0

W3 = 0

Dari ketiga nilai W dapat dilihat bahwa W2 merupakan nilai maksimal atau yangmemiliki nilai paling besar.

W1 normalisasi =

W1 normalisasi = 0,294117647

W2 normalisasi =

W2 normalisasi = 1

W3 normalisasi =

W3 normalisasi = 0

**Tabel 5.13** Normalisasi Data

|  |  |  |
| --- | --- | --- |
|  | W sebelum normalisasi | W sesudah normalisasi |
| X1 | 1,25 | 0,294117647 |
| X2 | 4,25 | 1 |
| X3 | 0 | 0 |

(1 x 0.41950509289520865) + ( 0.294117647 x -0.006869906605231357) + (1 x -0.006869906605231357) + (0 x 0.7172998130193451)

1.548013

= 0.824626569

(1 x 0.02797050059528916) + ( 0.294117647\* x 0.46947977401210983) + (1 x 0.7489034434760338) + (0 x0.4556733161755197)

1.145419

= 0.758673149

(1 x -0.9019764934184369) + (0.824626569 x 0.06444976220578177) + (0.758673149 x -0.371941138980724)

1.23731

𝑛1

= 0.224905

= (0.2249050.37842522764246456) + (0.758673149 x -0.00035341521547290963)

1.31292

= 0.211999

SSE =

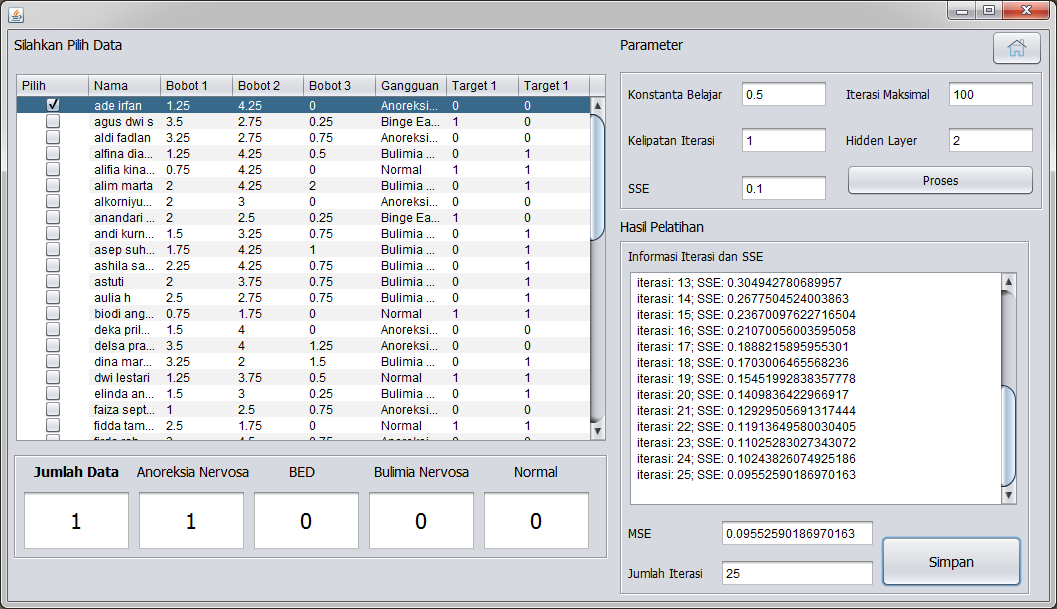
SSE = (0 - )2 + (0 - 0.211999)2

SSE = 0.095526

MSE =

MSE =

MSE = 0.095526



**Gambar 5.26** Halaman Data Pelatihan

Proses pelatihan data pada gambar 5.18 menggunakan konstanta belajar 0,5, iterasi maksimal 100, kelipatan iterasi 1, *hidden layer* 2 dan SSE 0,1. Dipilih 1 data yang memiliki nilai bobot pertama 1,25, bobot kedua 4.25, dan bobot ketiga 0, kemudian dilakukan proses pelatihan menggunakan metode *Backpropagation* maka didapatkan SSE 0.095526 dan MSE 0.095526 pada iterasi ke 25. Sehingga dapat dilihat proses pelatihan data secara manual dan system telah menghasilkan MSE yang sama.

* + 1. **Skenario Pelatihan**

**Tabel 5.14** Data pelatihan

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Nama | Bobot1 | Bobot2 | Bobot3 | Diagnosa | Target1 | Target2 |
| Ade Irfan | 1.25 | 4.25 | 0 | Anoreksia | 0 | 0 |
| Agus Dwi S | 3.5 | 2.75 | 0.25 | Binge | 1 | 0 |
| Alfina Diah A | 1.25 | 4.25 | 0.5 | Bulimia | 0 | 1 |
| Alifia Kinanti | 0.75 | 4.25 | 0 | Normal | 1 | 1 |

**Tabel 5.15** Data uji tak terlatih

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Nama | Bobot1 | Bobot2 | Bobot3 | Diagnosa | Target1 | Target2 |
| Aldi Fadlan | 3.25 | 2.75 | 0.75 | Anoreksia | 0 | 0 |
| Alim Marta | 2 | 4.25 | 2 | Bulimia | 0 | 1 |
| Anandari A P | 2 | 2.5 | 0.25 | Binge | 1 | 0 |
| Asep Suhendi | 1.75 | 4.25 | 1 | Bulimia | 0 | 1 |
| Astute | 2 | 3.75 | 0.75 | Bulimia | 0 | 1 |
| Biodi Anggraini | 0.75 | 1.75 | 0 | Normal | 1 | 1 |
| Elinda Anisa | 1.5 | 3 | 0.25 | Bulimia | 0 | 1 |
| Firda R | 3 | 4.5 | 0.75 | Anoreksia | 0 | 0 |
| Indah P H | 1.5 | 1.5 | 0 | Normal | 1 | 1 |
| Indriwati | 3.5 | 5.25 | 0 | Binge | 1 | 0 |

Telah dilakukan beberapa pelatihan dengan kombinasi jumlah hidden layer dan konstanta belajar yang berbeda-beda. Dimana batas iterasi maksimal adalah 50000 iterasi, selang tampilan 2000 dan batas SSE 0.1. Hasil dari pelatihan tersebut ditunjukan pada tabel berikut ini:

**Tabel 5.16** Hasil Pelatihan

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Hidden Layer | Konstanta Belajar | | | | |
| 0.01 | 0.04 | 0.1 | 0.5 | 0.75 |
| 10 | Tidak Konvergen | 30948 | 12956 | 3193 | **1887** |
| 20 | Tidak Konvergen | 40152 | 14023 | 2829 | 2024 |
| 30 | Tidak Konvergen | Tidak Konvergen | 18253 | 6387 | 2691 |
| 40 | Tidak Konvergen | Tidak Konvergen | Tidak Konvergen | 32254 | Tidak Konvergen |
| 50 | Tidak Konvergen | Tidak Konvergen | Tidak Konvergen | Tidak Konvergen | Tidak Konvergen |

Setelah melakukan proses pelatihan maka perlu dilakukan proses pengujian untuk mendapatkan tingkat akurasi jaringan didalam mengenali gangguan. Proses pengujian tersebut meliputi pengujian data terlatih dan pengujian data tak terlatih. Data terlatih merupakan data yang digunakan dalam proses pelatihan, sedangkan data baru merupakan data yang belum pernah dipakai. Tabel 5 menggambarkan tentang data hasil pengujian. Tabel tersebut menunjukan bahwa pelatihan terbaik adalah pada kondisi konstanta belajar 0,75, jumlah hidden layer 10. Pada kondisi tersebut didapatkan tingkat akurasi sebesar 100% pada pengujian data terlatih dan 40% pada pengujian data baru. Tingkat ketepatan tersebut sebenarnya juga didapatkan pada kondisi jumlah hidden layer dan konstanta belajar yang lain, namun alasan memilih pelatihan tersebut adalah pada pelatihan tersebut di dapatkan konvergen lebih cepat yakni pada iterasi ke-1887dan SSE sebesar 0,099893.

**Tabel 5.17** Hasil pengujian

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Konstanta belajar | Hidden Layer | SSE | MSE | Iterasi | Prosentae uji data latih | Prosentase uji data baru |
| 0.04 | 10 | 0.099988 | 0.024997 | 30948 | 100% | 40% |
| 20 | 0.099999 | 0.024999 | 40152 | 100% | 40% |
| 30 | TK | TK | TK | TK | TK |
| 40 | TK | TK | TK | TK | TK |
| 50 | TK | TK | TK | TK | TK |
| 0.1 | 10 | 0.099987 | 0.024996 | 12956 | 100% | 40% |
| 20 | 0.099976 | 0.024994 | 14023 | 100% | 40% |
| 30 | 0.099973 | 0.024993 | 18253 | 100% | 40% |
| 40 | TK | TK | TK | TK | TK |
| 50 | TK | TK | TK | TK | TK |
| 0.5 | 10 | 0.099975 | 0.024993 | 3193 | 100% | 40% |
| 20 | 0.099879 | 0.024969 | 2829 | 100% | 40% |
| 30 | 0.09989 | 0.024972 | 6387 | 100% | 40% |
| 40 | 0.099953 | 0.24988 | 32254 | 100% | 40% |
| 50 | TK | TK | TK | TK | TK |
| 0.75 | **10** | **0.099893** | **0.024973** | **1887** | **100%** | **40%** |
| 20 | 0.099962 | 0.02499 | 2024 | 100% | 40% |
| 30 | 0.099944 | 0.024986 | 2691 | 100% | 40% |
| 40 | TK | TK | TK | TK | TK |
| 50 | TK | TK | TK | TK | TK |

TK= Tidak Konvergen (tidak ditemukannya nilai yang mendekati).

**BAB VI**

**PENUTUP**

**VI.1. Kesimpulan**

Berdasarkan penelitian yang telah dilakukan dan rumusan masalah mengenai bagaimana implementasi dari jaringan saraf tiruan dalam penentuan diagnosa awal jenis sakit kepala dengan metode *backpropagation*, dapat diambil kesimpulan sebagai berikut:

1. Hasil implementasi metode menghasilkan deteksi kecenderungan gangguan makan, akan tetapi hasil ini bukan merupakan acuan bahwa remaja tersebut mengalami gangguan makan atau hanya merupakan deteksi awal dan akan lebih baik jika megalami gangguan makan langsung konsultasikan dengan psikolog agar tidak terjadi hal buruk dikemudian hari
2. Setelah dilakukan pengujian JST dengan 22 data baru maka dihasilkan akurasi pengujian sebesar 59% dengan kesalahan *absolute* sebesar 41% dengan arsitektur pelatihan yang memiliki kontanta belajar bernilai 0,5 kemudian lapisan tersembunyi sejumlah 4 dan SSE bernilai 0,01. Pada kondisi tersebut, untuk mencapai tingkat konvergensi atau ketika SSE (*Sum Square Error*)<=0,01 hanya dibutuhkan 592.169 iterasi. Ketika iterasi ke-592.169 didapatkan SSE sebesar 0,001.

**VI.2. Saran**

Saran yang dapat disampaikan untuk pengembang selanjutnya yaitu implementasi ini dapat dikembangkan dengan melalui sebuah website maupun aplikasi android sehingga pengguna (remaja) yang tidak mempunyai laptop, komputer atau pc dapat menggunakannya untuk mengetahui jenis kecenderuan gangguan makan dideritanya sehingga bisa dilakukan penanganan lebih dini.

**DAFTAR** **PUSTAKA**

Andrian, Yudhi dan M. Rhifky Wahyudi. 2014. “Analisis Kinerja Jaringan Saraf Tiruan Metode Backpropagation dalam memprediksi cuaca di kota Medan”, Seminar Nasional Ilmu Komputer 2014 (SNIKOM).

Davison, Gerald C et al. 2010, *Psikologi Abnormal Edisi ke 9*, Rajawali Pers, Jakarta.

Fathansyah., 2015, *Basis Data,* Informatika Bandung; Bandung.

Hermawan, Arief, 2006, *Jaringan Saraf Tiruan Teori dan Aplikasi*, Andi; Yogyakarta.

National Institute of Mental Health, 2011, *Eating Disorders,* U.S. Department of

Health and Human Services. National Institute of Health.

Siswoyo, dkk. 2014. “Klasifikasi Sinyal Otak Menggunakan Metode Logika Fuzzy dan Nourosky Mindset”, ISSN : 1412-9612.

WHO, 2012, *Growth Reference 5-19 year for* *Adolescent*, diakses tanggal 19 November 2017, <http://www.who.int/growthref/en/>

WHO, 2012, *Adolescent Health*, diakses tanggal 19 November 2017, <http://www.who.int/topics/adolescent_health/en/>

Yakub, 2012, *Pengantar Sistem Informasi*. Graha Ilmu; Jakarta.

Yudhawati, Dian. 2014. “Implementasi Jaringan Saraf Tiruan untuk mendeteksi DayaTahan Terhadap Stress pada Mahasiswa Tingkat Akhir menggunakan metode Backpropagation”, Jurnal Psikologi-ISSN : 1858-3970.

**LAMPIRAN**

**Form Kategori**

public class datakategori extends javax.swing.JFrame {

Statement Stm;

ResultSet Rs;

public datakategori() {

initComponents();

tampilkategori();

}

private void tampilkategori() {

try {

DefaultTableModel dtm = new DefaultTableModel(){

@Override

public boolean isCellEditable(int row, int col) {

return false;

}

};

dtm.addColumn("Kode Kategori");

dtm.addColumn("Nama Kategori");

String sql = "SELECT kode\_kategori, nama\_kategori FROM kategori";

Statement stmt = TugasAkhir.getConnection().createStatement();

ResultSet rs = stmt.executeQuery(sql);

while(rs.next()) {

dtm.addRow(new Object[] {

rs.getString("kode\_kategori"),

rs.getString("nama\_kategori")

});

jTable1.setModel(dtm);

}

stmt.close();

} catch (SQLException ex) {

Logger.getLogger(datakategori.class.getName()).log(Level.SEVERE,null,ex);

}

}

**Form Pertanyaan**

public class pertanyaan extends javax.swing.JFrame {

Statement Stm;

ResultSet Rs;

public pertanyaan() {

Dimension dim = Toolkit.getDefaultToolkit().getScreenSize();

int x = (dim.width-getWidth())/2;

int y = (dim.height-getHeight())/2;

setLocation(y, y);

initComponents();

kodepertanyaan();

clear();

tampilpertanyaan();

namakategori();

}

private void kodepertanyaan(){

try {

Statement Stm = (Statement)TugasAkhir.getConnection().createStatement();

String sql="select right(kode\_pertanyaan,2) as nama from "

+ "pertanyaan order by kode\_pertanyaan desc limit 1";

Rs=Stm.executeQuery(sql);

if(Rs.first()==false){

kodesoal.setText("1");

}

else{

Rs.last();

int no=Rs.getInt(1)+1;

String cno=String.valueOf(no);

int pjg\_cno=cno.length();

for (int a=0;a<2-pjg\_cno;a++){

cno="0"+cno;

}

kodesoal.setText(cno);

}

}

catch (Exception e) {

System.err.println("Error="+e);

}

}

private void namakategori(){

try{

Statement Stm = (Statement)TugasAkhir.getConnection().createStatement();

String sql="SELECT \* FROM kategori";

Rs=Stm.executeQuery(sql);

while (Rs.next()){

jComboBox1.addItem(Rs.getString("kode\_kategori"));

}

Rs.last();

int jmldata = Rs.getRow();

Rs.first();

}catch (Exception e) {

System.err.println("Error="+e);

}

}

private void clear(){

jComboBox1.setSelectedItem("-Pilih-");

soal.setText("");

jawaba.setText("");

jawabb.setText("");

jawabc.setText("");

jawabd.setText("");

jawabe.setText("");

}

private void tampilpertanyaan() {

try {

DefaultTableModel dtm = new DefaultTableModel();

dtm.addColumn("Kode Pertanyaan");

dtm.addColumn("Kategori Pertanyaan");

dtm.addColumn("Pertanyaan");

dtm.addColumn("Jawaban A");

dtm.addColumn("Jawaban B");

dtm.addColumn("Jawaban C");

dtm.addColumn("Jawaban D");

dtm.addColumn("Jawaban E");

String sql = "SELECT kode\_pertanyaan, kode\_kategori, soal, pilihanA, pilihanB, pilihanC, pilihanD, pilihanE FROM pertanyaan";

Statement stmt = TugasAkhir.getConnection().createStatement();

ResultSet rs = stmt.executeQuery(sql);

while(rs.next()) {

dtm.addRow(new Object[] {

rs.getString("kode\_pertanyaan"),

rs.getString("kode\_kategori"),

rs.getString("soal"),

rs.getString("pilihanA"),

rs.getString("pilihanB"),

rs.getString("pilihanC"),

rs.getString("pilihanD"),

rs.getString("pilihanE")

});

jTable1.setModel(dtm);

}

stmt.close();

} catch (SQLException ex) {

Logger.getLogger(datakategori.class.getName()).log(Level.SEVERE,null,ex);

}

}

private void simpanActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

String kd1=kodesoal.getText();

String ktg=(String)jComboBox1.getSelectedItem();

String pert=soal.getText();

String ja=jawaba.getText();

String jb=jawabb.getText();

String jc=jawabc.getText();

String jd=jawabd.getText();

String je=jawabe.getText();

if(pert.equals("")||ja.equals("")||jb.equals("")||jc.equals("")||jd.equals("")||je.equals(""))

{

JOptionPane.showMessageDialog(null, "Pengisian belum lengkap, Isilah data dengan lengkap !");

}

else

{

Statement stmt=null;

try

{

stmt = (Statement)TugasAkhir.getConnection().createStatement();

} catch (SQLException ex) {

Logger.getLogger(datauser.class.getName()).log(Level.SEVERE, null, ex);

}try{

if(stmt.executeUpdate("insert into pertanyaan values ('"+kd1+"','"+ktg+"','"+pert+"','"+ja+"','"+jb+"','"+jc+"','"+jd+"','"+je+"')")!=0) {

JOptionPane.showMessageDialog(null, "Berhasil disimpan");

} else {

JOptionPane.showMessageDialog(this, "Gagal simpan");

}

stmt.close();

} catch (SQLException ex) {

Logger.getLogger(datauser.class.getName()).log(Level.SEVERE, null, ex);

}

clear();

kodepertanyaan();

tampilpertanyaan();

}

}

private void kodesoalActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

}

private void ubahActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

try {

Statement Stm = (Statement)TugasAkhir.getConnection().createStatement();

int konfirm=JOptionPane.showConfirmDialog(this,

"Yakin data di ubah?","Konfirmasi",JOptionPane.YES\_NO\_OPTION);

if(konfirm==0)

{

String kd1=kodesoal.getText();

String ktg=(String)jComboBox1.getSelectedItem();

String pert=soal.getText();

String ja=jawaba.getText();

String jb=jawabb.getText();

String jc=jawabc.getText();

String jd=jawabd.getText();

String je=jawabe.getText();

Stm.executeUpdate("update pertanyaan set kode\_kategori='"+ktg+"',soal='"+pert+

"',pilihanA='"+ja+"', pilihanB='"+jb+"', pilihanC='"+jc+"', pilihanD='"+jd+

"', pilihanE='"+je+"'"+" where kode\_pertanyaan='"+kd1+"'");

JOptionPane.showMessageDialog(this, "Data Berhasil di Simpan");

clear();

kodepertanyaan();

tampilpertanyaan();

}

else

{

}

} catch(Exception e) {

JOptionPane.showMessageDialog(this, "Gagal Ubah");

}

}

private void hapusActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

try {

Statement Stm = (Statement)TugasAkhir.getConnection().createStatement();

int konfirm=JOptionPane.showConfirmDialog(this, "Yakin dihapus?",

"Konfirmasi",JOptionPane.YES\_NO\_OPTION);

if(konfirm==0)

{

String kd1=kodesoal.getText();

Stm.executeUpdate("delete from pertanyaan where kode\_pertanyaan='"+kd1+"'");

JOptionPane.showMessageDialog(this, "Berhasil Hapus");

clear();

Stm.close();

clear();

kodepertanyaan();

tampilpertanyaan();

}

else

{

clear();

Stm.close();

kodepertanyaan();

tampilpertanyaan();

}

}

catch(Exception e)

{

JOptionPane.showMessageDialog(this, "Gagal Hapus"+e);

}

}

private void jTable1MouseClicked(java.awt.event.MouseEvent evt) {

int row = jTable1.getSelectedRow();

kodesoal.setText(jTable1.getValueAt(row, 0).toString());

jComboBox1.setSelectedItem(jTable1.getValueAt(row, 1).toString());

soal.setText(jTable1.getValueAt(row, 2).toString());

jawaba.setText(jTable1.getValueAt(row, 3).toString());

jawabb.setText(jTable1.getValueAt(row, 4).toString());

jawabc.setText(jTable1.getValueAt(row, 5).toString());

jawabd.setText(jTable1.getValueAt(row, 6).toString());

jawabe.setText(jTable1.getValueAt(row, 7).toString());

}

**Form Input Data**

public class input\_data extends javax.swing.JFrame {

Connection cc = TugasAkhir.getConnection();

DefaultTableModel tabel = new DefaultTableModel();

String[] kategori = null;

String[] kdPertanyaan = null;

String[] soal = null;

double[] bobot = null;

String[] a = null;

String[] b = null;

String[] c = null;

String[] d = null;

String[] e = null;

int[] jumPerKategori = null;

int jumSoal = 0;

// int jumKategori = 0;

int pointer = 0;

int poin = 0;

String[] jawaban;

Boolean edit = false;

public input\_data() {

Dimension dim = Toolkit.getDefaultToolkit().getScreenSize();

int x = (dim.width-getWidth())/2;

int y = (dim.height-getHeight())/2;

setLocation(y, y);

initComponents();

jTextField1.setDocument(new validasiinput (validasiinput.huruf));

}

public void tampilSoal(){

int i = 0;

jTable1.setModel(tabel);

//set table column

tabel.addColumn("Soal");

tabel.addColumn("Jawaban");

tabel.getDataVector().removeAllElements();

tabel.fireTableDataChanged();

//get data soal

try{

Statement s = cc.createStatement();

ResultSet r;

//get jumlah soal

String jumSQL = "SELECT COUNT(kode\_pertanyaan) AS jumlah FROM pertanyaan";

r = s.executeQuery(jumSQL);

while (r.next()){

jumSoal = r.getInt("jumlah");

}

r.close();

//assign array length

kategori = new String[jumSoal];

kdPertanyaan = new String[jumSoal];

soal = new String[jumSoal];

bobot = new double[jumSoal];

jawaban = new String[jumSoal];

a = new String[jumSoal];

b = new String[jumSoal];

c = new String[jumSoal];

d = new String[jumSoal];

e = new String[jumSoal];

//get pertanyaan

String sql = "SELECT \* FROM pertanyaan ORDER BY kode\_pertanyaan";

r = s.executeQuery(sql);

i = 0;

while (r.next()){

kategori[i] = r.getString("kode\_kategori");

kdPertanyaan[i] = r.getString("kode\_pertanyaan");

soal[i] = r.getString("soal");

bobot[i] = 0.0;

a[i] = r.getString("pilihanA");

b[i]=r.getString("pilihanB");

c[i]=r.getString("pilihanC");

d[i]=r.getString("pilihanD");

e[i]=r.getString("pilihanE");

i++;

}

r.close();

s.close();

}catch(SQLException ee){

System.out.println(ee);

}

jTable1.setEnabled(false);

jLabel1.setText(""+(pointer+1));

jLabel9.setText(""+jumSoal);

jTextArea1.setText(soal[0]);

jRadioButton2.setText(a[0]);

jRadioButton1.setText(b[0]);

jRadioButton3.setText(c[0]);

jRadioButton4.setText(d[0]);

jRadioButton5.setText(e[0]);

}

public void radioEnable(){

jRadioButton1.setEnabled(true);

jRadioButton2.setEnabled(true);

jRadioButton3.setEnabled(true);

jRadioButton4.setEnabled(true);

jRadioButton5.setEnabled(true);

}

public void radioDisable(){

jRadioButton1.setEnabled(false);

jRadioButton2.setEnabled(false);

jRadioButton3.setEnabled(false);

jRadioButton4.setEnabled(false);

jRadioButton5.setEnabled(false);

}

public void pilih(){

if(pointer<jumSoal){

//set pilihan ke table

Object[] o = new Object[2];

o[0] = soal[pointer];

o[1] = jawaban[pointer];

tabel.addRow(o);

pointer++;

if(pointer==jumSoal){

jTextArea1.setText("SELESAI");

jLabel1.setText("-");

jTable1.setEnabled(true);

radioDisable();

}else{

jLabel1.setText(""+(pointer+1));

jTextArea1.setText(soal[pointer]);

}

}

if(edit){

jTable1.setValueAt(soal[poin], poin, 0);

jTable1.setValueAt(jawaban[poin], poin, 1);

jTextArea1.setText("SELESAI");

jLabel1.setText("-");

jTable1.setEnabled(true);

radioDisable();

edit = false;

}

buttonGroup1.clearSelection();

}

private void jButton4ActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

home utama = new home();

this.dispose();

utama.setVisible(true);

}

private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {

int kode=0;

//cek kode\_perolehan terakhir

try{

Statement s = cc.createStatement();

ResultSet r;

//get jumlah soal

String jumSQL = "SELECT kode\_perolehan FROM perolehan\_copy ORDER BY kode\_perolehan DESC LIMIT 1";

r = s.executeQuery(jumSQL);

while (r.next()){

kode = r.getInt("kode\_perolehan");

kode++;

}

r.close();

}catch (SQLException e){

JOptionPane.showMessageDialog(null, "Kode perolehan gagal dibaca!", "Error Message", 0);

}finally{

//clear();

}

if(jTextField1.getText().equals("")){

JOptionPane.showMessageDialog(this, "Isi nama terlebih dahulu..!!");

}else{

PreparedStatement p;

String sql;

try{

//simpan data baru

for(int i=0;i<jumSoal;i++){

sql = "INSERT INTO perolehan\_copy VALUES(?,?,?,?,?,?,?,?)";

p = cc.prepareStatement(sql);

p.setInt(1, kode);

p.setString(2, jTextField1.getText());

p.setString(3, kdPertanyaan[i]);

p.setString(4, "null");

p.setString(5, kategori[i]);

p.setString(6, jawaban[i]);

p.setDouble(7, bobot[i]);

p.setString(8, "Data Awal");

p.executeUpdate();

p.close();

}

JOptionPane.showMessageDialog(null, "Data Awal Berhasil Disimpan!", "Error Message", 1);

}catch (SQLException e){

JOptionPane.showMessageDialog(null, "Data Awal Gagal Disimpan!", "Error Message", 0);

}finally{

//clear();

}

}

}

private void formWindowOpened(java.awt.event.WindowEvent evt) {

// TODO add your handling code here:

tampilSoal();

}

private void jRadioButton2ActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

if(edit){

bobot[poin] = 0.00;

jawaban[poin] = jRadioButton2.getText();

}else{

bobot[pointer] = 0.00;

jawaban[pointer] = jRadioButton2.getText();

}

pilih();

}

private void jRadioButton1ActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

if(edit){

bobot[poin] = 0.25;

jawaban[poin] = jRadioButton1.getText();

}else{

bobot[pointer] = 0.25;

jawaban[pointer] = jRadioButton1.getText();

}

pilih();

}

private void jRadioButton3ActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

if(edit){

bobot[poin] = 0.5;

jawaban[poin] = jRadioButton3.getText();

}else{

bobot[pointer] = 0.5;

jawaban[pointer] = jRadioButton3.getText();

}

pilih();

}

private void jRadioButton4ActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

if(edit){

bobot[poin] = 0.75;

jawaban[poin] = jRadioButton4.getText();

}else{

bobot[pointer] = 0.75;

jawaban[pointer] = jRadioButton4.getText();

}

pilih();

}

private void jRadioButton5ActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

if(edit){

bobot[poin] = 1.0;

jawaban[poin] = jRadioButton5.getText();

}else{

bobot[pointer] = 1.0;

jawaban[pointer] = jRadioButton5.getText();

}

pilih();

}

private void jTable1MouseClicked(java.awt.event.MouseEvent evt) {

// TODO add your handling code here:

poin = jTable1.getSelectedRow();

jTextArea1.setText(jTable1.getValueAt(poin, 0).toString());

jLabel1.setText(""+(poin+1));

radioEnable();

edit = true;

}

**Form Input Gangguan**

String kolom0,kolom1;

public inputdatagangguan() {

initComponents();

tampiluser();

}

private void TampilCombobox() {

jComboBox1.removeAllItems();

jComboBox1.addItem("-Pilih-");

try {

String query = "select \* from gangguan";

Statement st = TugasAkhir.getConnection().createStatement();

ResultSet rs = st.executeQuery(query);

while (rs.next()) {

jComboBox1.addItem(rs.getString("nama\_gangguan"));

}

st.close();

rs.close();

TugasAkhir.getConnection().close();

} catch(Exception e) {

System.out.println(e);

}

}

private void tampiluser() {

try {

DefaultTableModel dtm = new DefaultTableModel(){

@Override

public boolean isCellEditable(int row, int col) {

return false;

}

};

dtm.addColumn("Id");

dtm.addColumn("Nama");

dtm.addColumn("Gangguan");

String sql = "SELECT DISTINCT kode\_perolehan, nama , nama\_gangguan FROM perolehan\_copy p, gangguan g where p.kode\_gangguan=g.kode\_gangguan ";

Statement stmt = TugasAkhir.getConnection().createStatement();

ResultSet rs = stmt.executeQuery(sql);

while(rs.next()) {

dtm.addRow(new Object[] {

rs.getString("kode\_perolehan"),

rs.getString("nama"),

rs.getString("nama\_gangguan")

});

jTable1.setModel(dtm);

}

stmt.close();

} catch (SQLException ex) {

Logger.getLogger(datauser.class.getName()).log(Level.SEVERE,null,ex);

}

}

private void formWindowOpened(java.awt.event.WindowEvent evt) {

// TODO add your handling code here:

TampilCombobox();

tampiluser();

}

private void jTable1MouseClicked(java.awt.event.MouseEvent evt) {

// TODO add your handling code here:

jComboBox1.setEnabled(true);

jButton1.setEnabled(true);

int baris=jTable1.getSelectedRow();

int kolom=jTable1.getSelectedColumn();

String dataterpilih=jTable1.getValueAt(baris, kolom).toString();

kolom0=jTable1.getValueAt(baris, 0).toString();

kolom1=jTable1.getValueAt(baris, 1).toString();

try {

DefaultTableModel dtm = new DefaultTableModel();

dtm.addColumn("Soal");

dtm.addColumn("Jawaban");

dtm.addColumn("Bobot");

String sql = "SELECT pertanyaan.soal, perolehan\_copy.jawaban,bobot FROM perolehan\_copy, pertanyaan WHERE pertanyaan.kode\_pertanyaan=perolehan\_copy.kode\_pertanyaan and nama='"+kolom1+"'";

Statement stmt = TugasAkhir.getConnection().createStatement();

ResultSet rs = stmt.executeQuery(sql);

while(rs.next()) {

dtm.addRow(new Object[] {

rs.getString("soal"),

rs.getString("jawaban"),

rs.getString("bobot")

});

jTable2.setModel(dtm);

}

stmt.close();

} catch (SQLException ex) {

Logger.getLogger(datauser.class.getName()).log(Level.SEVERE,null,ex);

}

}

private void jTable2MouseClicked(java.awt.event.MouseEvent evt) {

// TODO add your handling code here:

}

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

String kode="";

String trg="";

try {

String ggn=(String) jComboBox1.getSelectedItem();

Statement Stm = (Statement)TugasAkhir.getConnection().createStatement();

String sql = "SELECT kode\_gangguan from gangguan where nama\_gangguan='"+ggn+"'";

ResultSet rs = Stm.executeQuery(sql);

if (rs.next()) {

kode = rs.getString("kode\_gangguan");

//trg = rs.getString("target");

}

//Stm.close();

int konfirm=JOptionPane.showConfirmDialog(this,

"Yakin data disimpan?","Konfirmasi",JOptionPane.YES\_NO\_OPTION);

if(konfirm==0)

{

Stm.executeUpdate("update perolehan\_copy set kode\_gangguan='"+kode+"'"+" where kode\_perolehan='"+kolom0+"'");

JOptionPane.showMessageDialog(this, "Berhasil di ubah");

Stm.close();

tampiluser();

}

else

{

//batalkrActionPerformed(evt);

}

} catch(Exception e) {

JOptionPane.showMessageDialog(this, "Gagal ubah"+e);

}

}

**Form Pelatihan**

public class pelatihan extends javax.swing.JFrame{

private double[][] data;

private int[][] target;

private String[] nama;

private int jumData;

private int jumHL;

private double max;

private static int jumInput = 3;

private static int jumOutput = 2;

private double[][] v,bv,w,bw;

//variables for counting total diseas

int jumlah =0;

int jumAno =0;

int jumBed =0;

int jumNor =0;

int jumBul =0;

private ArrayList<ArrayList<String>> aList;

private XYSeries series;

Connection c = TugasAkhir.getConnection();

public pelatihan() {

initComponents();

txtJumlahHidden.setDocument(new validasiinput(validasiinput.numeric));

txtKelipatanIterasi.setDocument(new validasiinput(validasiinput.numeric));

txtKonstantaBelajar.setDocument(new validasiinput(validasiinput.numeric));

txtMaksimalIterasi.setDocument(new validasiinput(validasiinput.numeric));

txtSse.setDocument(new validasiinput(validasiinput.numeric));

series = new XYSeries("Nilai MSE");

}

private void TampilData() {

DefaultTableModel tabel = new DefaultTableModel(){

@Override

public boolean isCellEditable(int row, int col) {

return col == 0;

}

@Override

public Class<?> getColumnClass(int columnIndex) {

Class<?> clazz = String.class;

switch (columnIndex) {

case 0:

clazz = Boolean.class;

break;

}

return clazz;

}

@Override

public void setValueAt(Object aValue, int row, int column) {

if (aValue instanceof Boolean && column == 0) {

@SuppressWarnings("unchecked")

Vector<Object> rowData = (Vector<Object>)getDataVector().get(row);

rowData.set(0, (boolean)aValue);

fireTableCellUpdated(row, column);

}

}

};

jTable1.setModel(tabel);

tabel.addColumn("Pilih");

tabel.addColumn("Nama");

tabel.addColumn("Bobot 1");

tabel.addColumn("Bobot 2");

tabel.addColumn("Bobot 3");

tabel.addColumn("Gangguan");

tabel.addColumn("Target 1");

tabel.addColumn("Target 1");

tabel.getDataVector().removeAllElements();

tabel.fireTableDataChanged();

try{

Statement s = c.createStatement();

String sql = "SELECT aa.nama AS nama,SUM(aa.bobot1) AS bobot1,SUM(aa.bobot2) AS bobot2,SUM(aa.bobot3) AS bobot3, bb.nama\_gangguan, bb.target1 AS t1, bb.target2 AS t2 " +

" FROM (SELECT a.nama,SUM(a.bobot) AS bobot1,0 AS bobot2,0 AS bobot3,a.kode\_gangguan,a.kode\_kategori,a.ket " +

" FROM perolehan\_copy AS a" +

" WHERE a.kode\_kategori=1 GROUP BY a.nama " +

" UNION ALL " +

" SELECT a.nama,0 AS bobot1,SUM(a.bobot) AS bobot2,0 AS bobot3,a.kode\_gangguan,a.kode\_kategori,a.ket " +

" FROM perolehan\_copy AS a " +

" WHERE a.kode\_kategori=2 GROUP BY a.nama " +

" UNION ALL " +

" SELECT a.nama,0 AS bobot1,0 AS bobot2,SUM(a.bobot) AS bobot3,a.kode\_gangguan,a.kode\_kategori,a.ket " +

" FROM perolehan\_copy AS a " +

" WHERE a.kode\_kategori=3 GROUP BY a.nama) AS aa, gangguan AS bb, kategori AS cc " +

" WHERE aa.kode\_gangguan=bb.kode\_gangguan AND aa.kode\_kategori=cc.kode\_kategori " +

" GROUP BY aa.nama";

ResultSet r = s.executeQuery(sql);

while (r.next()){

Object[] o = new Object[8];

o[0]=false;

o[1]=r.getString("nama");

o[2]=r.getString("bobot1");

o[3]=r.getString("bobot2");

o[4]=r.getString("bobot3");

o[5]=r.getString("bb.nama\_gangguan");

o[6]=r.getString("t1");

o[7]=r.getString("t2");

tabel.addRow(o);

}

r.close();

s.close();

}catch (SQLException e){

JOptionPane.showMessageDialog(this, "Error Load Data","Error Message",0);

}

jumlahdata1.setText(""+jumlah);

ano1.setText(""+jumAno);

bed1.setText(""+jumBed);

bno1.setText(""+jumBul);

nor1.setText(""+jumNor);

}

private void hapusBobotBias(){

PreparedStatement p;

try{

String sql1 = "DELETE from bobotbias\_hidden";

String sql2 = "DELETE from bobotbias\_output";

p = c.prepareStatement(sql1);

p.executeUpdate();

p = c.prepareStatement(sql2);

p.executeUpdate();

p.close();

}catch (SQLException e){

JOptionPane.showMessageDialog(null, "Bobot Bias Gagal Dihapus!", "Error Message", 0);

}finally{

}

}

private void simpanBobotBiasHidden(){

PreparedStatement p;

String sql1,sql2;

try{

//bobot hidden

for(int i=0;i<jumInput;i++){

for(int j=0;j<jumHL;j++){

sql1 = "INSERT INTO bobotbias\_hidden VALUES(?,?)";

p = c.prepareStatement(sql1);

p.setString(1, "v["+i+"]["+j+"]");

p.setDouble(2, v[i][j]);

p.executeUpdate();

p.close();

}

}

//bias hidden

for(int i=0;i<jumHL;i++){

sql1 = "INSERT INTO bobotbias\_hidden VALUES(?,?)";

p = c.prepareStatement(sql1);

p.setString(1, "bv["+i+"]");

p.setDouble(2, bv[0][i]);

p.executeUpdate();

p.close();

}

//jumlah hidden dan nilai inputan max

sql1 = "INSERT INTO bobotbias\_hidden VALUES(?,?)";

p = c.prepareStatement(sql1);

p.setString(1, "jumlah hidden");

p.setDouble(2, jumHL);

p.executeUpdate();

p.close();

sql2 = "INSERT INTO bobotbias\_hidden VALUES(?,?)";

p = c.prepareStatement(sql2);

p.setString(1, "nilai max");

p.setDouble(2, max);

p.executeUpdate();

p.close();

}catch (SQLException e){

JOptionPane.showMessageDialog(null, "Bobot Bias Hidden Gagal Disimpan!", "Error Message", 0);

}finally{

//clear();

}

}

private void simpanBobotBiasOutput(){

PreparedStatement p;

String sql;

try{

//bobot output

for(int i=0;i<jumHL;i++){

for(int j=0;j<jumOutput;j++){

sql = "INSERT INTO bobotbias\_output VALUES(?,?)";

p = c.prepareStatement(sql);

p.setString(1, "w["+i+"]["+j+"]");

p.setDouble(2, w[i][j]);

p.executeUpdate();

p.close();

}

}

//bias output

for(int i=0;i<jumOutput;i++){

sql = "INSERT INTO bobotbias\_output VALUES(?,?)";

p = c.prepareStatement(sql);

p.setString(1, "bw["+i+"]");

p.setDouble(2, bw[0][i]);

p.executeUpdate();

p.close();

}

}catch (SQLException e){

JOptionPane.showMessageDialog(null, "Bobot Bias Output Gagal Disimpan!", "Error Message", 0);

}finally{

//clear();

}

}

private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

home utama = new home();

this.dispose();

utama.setVisible(true);

}

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {

double konsBelajar = Double.parseDouble(txtKonstantaBelajar.getText());

if (konsBelajar > 1.0){

JOptionPane.showMessageDialog(this, "Nilai konstanta belajar harus kurang dari 1");

}else{

jTextArea3.removeAll();

//assign parameter input dan target

int jumAll = jTable1.getRowCount();

int index = 0;

jumData = jumlah;

data = new double[jumData][jumInput];

target = new int[jumData][jumOutput];

nama = new String[jumData];

for(int i=0;i<jumAll;i++){

if((Boolean)jTable1.getValueAt(i, 0)==true){

nama[index] = jTable1.getValueAt(i,1).toString();

for(int j=0;j<jumInput;j++){

data[index][j] = Double.parseDouble(jTable1.getValueAt(i,j+2).toString());

}

target[index][0] = Integer.parseInt(jTable1.getValueAt(i,6).toString());

target[index][1] = Integer.parseInt(jTable1.getValueAt(i,7).toString());

index++;

}

}

//cari nilai maksimal dari inputan

max = 0;

for(int i=0;i<jumData;i++){

for(int j=0;j<jumInput;j++){

if(data[i][j] > max){

max = data[i][j];

}

}

}

//normalisasi data inputan

for(int i=0;i<jumData;i++){

for(int j=0;j<jumInput;j++){

data[i][j] = data[i][j]/max;

}

}

//procedure penghitungan BP

hitungBP(data, target, jumData);

}

}

//procedure hitungBP

private void hitungBP(double[][] x,int[][] t,int jmlData){

final double e = 2.718281828;

double[][] z,y;

double[] hiddenL,sy,q,sz;

int c,i,j,k;

int jmlinpt,jmlHL,jmlOutput,iterasi,maxIter,selang;

double zin,yin,jumlah,sse,galat;

double konsBelajar;

jmlHL = Integer.parseInt(txtJumlahHidden.getText());

maxIter = Integer.parseInt(txtMaksimalIterasi.getText());

galat = Double.parseDouble(txtSse.getText());

konsBelajar = Double.parseDouble(txtKonstantaBelajar.getText());

selang = Integer.parseInt(txtKelipatanIterasi.getText());

if (konsBelajar>=1){

JOptionPane.showMessageDialog(this, "melebihi batas");

}

// //set nilai konfigurasi jaringan

jmlinpt = jumInput;

jmlOutput = jumOutput;

jumHL = jmlHL;

//bobot x-z

v = new double[jmlinpt][jmlHL];

for(i=0;i<jmlinpt;i++){

for(j=0;j<jmlHL;j++){

v[i][j] = Math.random();

}

}

//bias x-z

bv = new double[1][jmlHL];

for(i=0;i<jmlHL;i++){

bv[0][i] = Math.random();

}

//bobot z-y

w = new double[jmlHL][jmlOutput];

for(i=0;i<jmlHL;i++){

for(j=0;j<jmlOutput;j++){

w[i][j] = Math.random();

}

}

//bias z-y

bw = new double[1][jmlHL];

for(i=0;i<jmlOutput;i++){

bw[0][i] = Math.random();

}

/\*\* hitung dan cari nilai z \*\*/

zin = 0;

z = new double[jmlData][jmlHL];

for(i=0;i<jmlData;i++){

for(j=0;j<jmlHL;j++){

zin = 1\*bv[0][j];

for(c=0;c<jmlinpt;c++){

zin += (x[i][c]\*v[c][j]);

}

z[i][j] = 1/(1 + (Math.pow(e, (-(zin)))));

}

}

/\*\* hitung dan cari nilai y \*\*/

yin = 0;

y = new double[jmlData][jmlOutput];

for(i=0;i<jmlData;i++){

for(j=0;j<jmlOutput;j++){

yin = 1\*bw[0][j];

for(k=0;k<jmlHL;k++){

yin += (z[i][k]\*w[k][j]);

}

y[i][j] = 1/(1 + (Math.pow(e, (-(yin)))));

}

}

/\*\* hitung sse \*\*/

sse = 0;

for(i=0;i<jmlData;i++){

for(j=0;j<jmlOutput;j++){

sse += Math.pow((t[i][j]-y[i][j]), 2);

}

}

jTextArea3.append("iterasi: "+0+"; SSE: "+sse+"\n");

//\*\* Masuk iterasi \*\*//

iterasi = 0;

hiddenL = new double[jmlHL];

// System.out.println("iterasi: "+iterasi+"; SSE: "+sse);

do {

iterasi += 1;

for(i=0;i<jmlData;i++){

//hitung nilai zin dan z

for(j=0;j<jmlHL;j++){

jumlah = 1\*bv[0][j];

for(k=0;k<jmlinpt;k++){

jumlah += (x[i][k]\*v[k][j]);

}

hiddenL[j] = 1 / (1 + (Math.pow(e,-(jumlah))));

}

//hitung nilai yin dan y

for(j=0;j<jmlOutput;j++){

jumlah = 1\*bw[0][j];

for(k=0;k<jmlHL;k++){

jumlah += (hiddenL[k]\*w[k][j]);

}

y[i][j] = 1 / (1 + (Math.pow(e,-(jumlah))));

}

//hitung galat informasi sy

sy = new double[jmlOutput];

for(j=0;j<jmlOutput;j++){

sy[j] = (t[i][j]-y[i][j])\*y[i][j]\*(1-y[i][j]);

}

//hitung nilai dinz

q = new double[jmlHL];

for(j=0;j<jmlHL;j++){

q[j]=0;

for(k=0;k<jmlOutput;k++){

q[j] += (sy[k]\*w[j][k]);

}

}

//hitung nilai dz

sz = new double[jmlHL];

for(j=0;j<jmlHL;j++){

sz[j] = q[j]\*hiddenL[j]\*(1-hiddenL[j]);

}

//hitung bobot dan bias baru y

for(j=0;j<jmlOutput;j++){

bw[0][j] += (konsBelajar\*sy[j]\*1);

for(k=0;k<jmlHL;k++){

w[k][j] += (konsBelajar\*sy[j]\*hiddenL[k]);

}

}

//hitung bobot dan bias baru z

for(j=0;j<jmlHL;j++){

bv[0][j] += (konsBelajar\*sz[j]\*1);

for(k=0;k<jmlinpt;k++){

v[k][j] += (konsBelajar\*sz[j]\*x[i][k]);

}

}

}

//hitung dan cari nilai z

for(i=0;i<jmlData;i++){

for(j=0;j<jmlHL;j++){

zin = 1\*bv[0][j];

for(k=0;k<jmlinpt;k++){

zin += (x[i][k]\*v[k][j]);

}

z[i][j] = 1 / (1 + (Math.pow(e,(-(zin)))));

}

}

//hitung dan cari nilai y

for(i=0;i<jmlData;i++){

for(j=0;j<jmlOutput;j++){

yin = 1\*bw[0][j];

for(k=0;k<jmlHL;k++){

yin += (z[i][k]\*w[k][j]);

}

y[i][j] = 1 / (1 + (Math.pow(e,(-(yin)))));

}

}

//hitung sse

sse = 0;

for(i=0;i<jmlData;i++){

for(j=0;j<jmlOutput;j++){

sse += Math.pow((t[i][j]-y[i][j]), 2);

}

}

//selang tampilan

if(((iterasi%selang) == 0) || (sse<galat)){

jTextArea3.append("iterasi: "+iterasi+"; SSE: "+sse+"\n");

}

series.add(sse,iterasi);

} while ((iterasi<maxIter)&&(sse>galat));

//chart

XYSeriesCollection dataset = new XYSeriesCollection(series);

JFreeChart chart = ChartFactory.createXYLineChart("Grafik Mean Squared Error (MSE)",

"SSE", "Iterasi", dataset,PlotOrientation.HORIZONTAL,true,true,true);

ChartPanel chartpanel = new ChartPanel(chart);

JPanel panelChart = new JPanel();

panelChart.setLayout(new BorderLayout());

panelChart.add(chartpanel,BorderLayout.NORTH);

JFrame frame = new JFrame();

frame.add(panelChart);

frame.pack();

setLocationRelativeTo(null);

frame.setVisible(true);

double mseNil = sse/jumData;

mse.setText(""+mseNil);

iterasicapai.setText(""+iterasi);

}

private void formWindowOpened(java.awt.event.WindowEvent evt) {

TampilData();

}

private void jTable1MouseClicked(java.awt.event.MouseEvent evt) {

//count total diseas

int i = jTable1.getSelectedRow();

int getT1 = Integer.parseInt((String)jTable1.getModel().getValueAt(i,6));

int getT2 = Integer.parseInt((String)jTable1.getModel().getValueAt(i,7));

if((Boolean)jTable1.getValueAt(jTable1.getSelectedRow(), 0)==false){

jTable1.setValueAt(true, i, 0);

jumlah++;

if((getT1==0) && (getT2==0)){

jumAno += 1;

}else if((getT1==0) && (getT2==1)){

jumBul += 1;

}else if((getT1==1) && (getT2==0)){

jumBed += 1;

}else {

jumNor += 1;

}

}else{

jTable1.setValueAt(false, i, 0);

jumlah--;

if((getT1==0) && (getT2==0)){

jumAno -= 1;

}else if((getT1==0) && (getT2==1)){

jumBul -= 1;

}else if((getT1==1) && (getT2==0)){

jumBed -= 1;

}else {

jumNor -= 1;

}

}

jumlahdata1.setText(""+jumlah);

ano1.setText(""+jumAno);

bed1.setText(""+jumBed);

bno1.setText(""+jumBul);

nor1.setText(""+jumNor);

}

private void simpanActionPerformed(java.awt.event.ActionEvent evt) {

try{

//reset all data

for(int i=0;i<jumData;i++){

String ket = "Data Awal";

// String sql = "UPDATE perolehan\_copy SET ket=? WHERE nama=?";

String sql = "UPDATE perolehan\_copy SET ket=?";

PreparedStatement p = c.prepareStatement(sql);

p.setString(1, ket);

p.executeUpdate();

p.close();

}

//set data awal to data latih

for(int i=0;i<jumData;i++){

String ket = "Data Latih";

String sql = "UPDATE perolehan\_copy SET ket=? WHERE nama=?";

PreparedStatement p = c.prepareStatement(sql);

p.setString(1, ket);

p.setString(2, nama[i]);

p.executeUpdate();

p.close();

}

//hapus bobot bias yang ada

hapusBobotBias();

//simpan bobot dan bias

simpanBobotBiasHidden();

simpanBobotBiasOutput();

JOptionPane.showMessageDialog(null, "Data Berhasil Disimpan!","Succes Message",1);

}catch (SQLException e){

JOptionPane.showMessageDialog(null, "Data Gagal Disimpan!", "Error Message", 0);

}finally{

//clear();

}

}

Form Pengujian

public class pengujian extends javax.swing.JFrame {

private DefaultTableModel tabel;

private final java.sql.Connection c = TugasAkhir.getConnection();

private double[][] data;

private int[][] target;

private String[] nama;

private String[] gangguan;

private int jumData;

public int jumHL;

public double max;

public static final int jumInput = 3;

public static final int jumOutput = 2;

public double[][] v,bv,w,bw;

private double[][] y;

public pengujian() {

Dimension dim = Toolkit.getDefaultToolkit().getScreenSize();

int x = (dim.width-getWidth())/2;

int y = (dim.height-getHeight())/2;

setLocation(y, y);

initComponents();

}

private void tampilData(){

tabel = new DefaultTableModel(){

@Override

public boolean isCellEditable(int row, int col) {

return col == 0;

}

@Override

public Class<?> getColumnClass(int columnIndex) {

Class<?> clazz = String.class;

switch (columnIndex) {

case 0:

clazz = Boolean.class;

break;

}

return clazz;

}

@Override

public void setValueAt(Object aValue, int row, int column) {

if (aValue instanceof Boolean && column == 0) {

@SuppressWarnings("unchecked")

Vector<Object> rowData = (Vector<Object>)getDataVector().get(row);

rowData.set(0, (boolean)aValue);

fireTableCellUpdated(row, column);

}

}

};

jTable1.setModel(tabel);

tabel.addColumn("Pilih");

tabel.addColumn("Nama");

tabel.addColumn("Bobot 1");

tabel.addColumn("Bobot 2");

tabel.addColumn("Bobot 3");

tabel.addColumn("Gangguan");

tabel.addColumn("Target 1");

tabel.addColumn("Target 1");

tabel.getDataVector().removeAllElements();

tabel.fireTableDataChanged();

}

private void dataPilih(String jenisData){

try{

Statement s = c.createStatement();

String sql = "SELECT aa.nama AS nama,SUM(aa.bobot1) AS bobot1,SUM(aa.bobot2) AS bobot2,SUM(aa.bobot3) AS bobot3, bb.nama\_gangguan, bb.target1 AS t1, bb.target2 AS t2 " +

" FROM (SELECT a.nama,SUM(a.bobot) AS bobot1,0 AS bobot2,0 AS bobot3,a.kode\_gangguan,a.kode\_kategori,a.ket " +

" FROM perolehan\_copy AS a" +

" WHERE a.kode\_kategori=1 GROUP BY a.nama " +

" UNION ALL " +

" SELECT a.nama,0 AS bobot1,SUM(a.bobot) AS bobot2,0 AS bobot3,a.kode\_gangguan,a.kode\_kategori,a.ket " +

" FROM perolehan\_copy AS a " +

" WHERE a.kode\_kategori=2 GROUP BY a.nama " +

" UNION ALL " +

" SELECT a.nama,0 AS bobot1,0 AS bobot2,SUM(a.bobot) AS bobot3,a.kode\_gangguan,a.kode\_kategori,a.ket " +

" FROM perolehan\_copy AS a " +

" WHERE a.kode\_kategori=3 GROUP BY a.nama) AS aa, gangguan AS bb, kategori AS cc " +

" WHERE aa.kode\_gangguan=bb.kode\_gangguan AND aa.kode\_kategori=cc.kode\_kategori AND aa.ket='"+jenisData+"' " +

" GROUP BY aa.nama";

ResultSet r = s.executeQuery(sql);

while (r.next()){

Object[] o = new Object[8];

o[0]=false;

o[1]=r.getString("nama");

o[2]=r.getString("bobot1");

o[3]=r.getString("bobot2");

o[4]=r.getString("bobot3");

o[5]=r.getString("bb.nama\_gangguan");

o[6]=r.getString("t1");

o[7]=r.getString("t2");

tabel.addRow(o);

}

r.close();

s.close();

}catch (SQLException e){

JOptionPane.showMessageDialog(this, "Error Load Data Pilih","Error Message",0);

}finally{

System.out.println("Berhasil load data dipilih");

}

}

public void getBobotBias(){

int i,j;

Statement s;

ResultSet r;

String sql;

try{

s = c.createStatement();

//jumlah hidden

sql = "SELECT \* FROM bobotbias\_hidden WHERE bobotbias='jumlah hidden'";

r = s.executeQuery(sql);

while (r.next()) {

jumHL = Integer.parseInt(r.getString("nilai"));

}

r.close();

//nilai max

sql = "SELECT \* FROM bobotbias\_hidden WHERE bobotbias='nilai max'";

r = s.executeQuery(sql);

while (r.next()) {

max = r.getDouble("nilai");

}

r.close();

//bobot hidden

v = new double[jumInput][jumHL];

for(i=0;i<jumInput;i++){

for(j=0;j<jumHL;j++){

sql = "SELECT \* from bobotbias\_hidden WHERE bobotbias='v["+i+"]["+j+"]'";

r = s.executeQuery(sql);

while (r.next()) {

v[i][j] = r.getDouble("nilai");

}

}

}

r.close();

//bias hidden

bv = new double[1][jumHL];

for(i=0;i<jumHL;i++){

sql = "SELECT \* from bobotbias\_hidden WHERE bobotbias='bv["+i+"]'";

r = s.executeQuery(sql);

while (r.next()) {

bv[0][i] = r.getDouble("nilai");

}

}

r.close();

//bobot output

w = new double[jumHL][jumOutput];

for(i=0;i<jumHL;i++){

for(j=0;j<jumOutput;j++){

sql = "SELECT \* from bobotbias\_output WHERE bobotbias='w["+i+"]["+j+"]'";

r = s.executeQuery(sql);

while (r.next()) {

w[i][j] = r.getDouble("nilai");

}

}

}

r.close();

//bias output

bw = new double[1][jumOutput];

for(i=0;i<jumOutput;i++){

sql = "SELECT \* from bobotbias\_output WHERE bobotbias='bw["+i+"]'";

r = s.executeQuery(sql);

while (r.next()) {

bw[0][i] = r.getDouble("nilai");

}

}

r.close();

s.close();

System.out.println("Berhasil load bobot dan bias");

}catch (SQLException e){

JOptionPane.showMessageDialog(this, "Error Load Bobot Bias","Error Message",0);

}

}

private void dataPengujian(){

int i,j,rowCount;

ArrayList list = new ArrayList();

jumData = 0;

//get jumlah data

rowCount = jTable1.getModel().getRowCount();

if(jRadioButton1.isSelected()){

for(i=0;i<rowCount;i++){

jumData = jTable1.getModel().getRowCount();

list.add(i);

}

}else{

for(i=0;i<rowCount;i++){

if((Boolean)jTable1.getModel().getValueAt(i, 0)==true){

jumData += 1;

list.add(i);

}

}

}

//get data

data = new double[jumData][jumInput];

target = new int[jumData][jumOutput];

nama = new String[jumData];

gangguan = new String[jumData];

for(i=0;i<jumData;i++){

int a = (Integer)list.get(i);

nama[i] = (String)jTable1.getModel().getValueAt(a,1);

gangguan[i] = (String)jTable1.getModel().getValueAt(a,5);

for(j=0;j<jumInput;j++){

data[i][j] = Double.parseDouble((String)jTable1.getModel().getValueAt(a,j+2));

}

target[i][0] = Integer.parseInt((String)jTable1.getModel().getValueAt(a,6));

target[i][1] = Integer.parseInt((String)jTable1.getModel().getValueAt(a,7));

}

}

private void forwardPass(){

final double e = 2.718281828;

double[][] z;

int i,j,k;

double zin,yin;

// System.out.println("max:" +max);

/\*\* hitung dan cari nilai z \*\*/

zin = 0;

z = new double[jumData][jumHL];

for(i=0;i<jumData;i++){

for(j=0;j<jumHL;j++){

zin = 1\*bv[0][j];

for(k=0;k<jumInput;k++){

zin += ((data[i][k]/max)\*v[k][j]);

}

z[i][j] = 1/(1 + (Math.pow(e, (-(zin)))));

}

}

/\*\* hitung dan cari nilai y \*\*/

yin = 0;

y = new double[jumData][jumOutput];

for(i=0;i<jumData;i++){

for(j=0;j<jumOutput;j++){

yin = 1\*bw[0][j];

for(k=0;k<jumHL;k++){

yin += (z[i][k]\*w[k][j]);

}

y[i][j] = 1/(1 + (Math.pow(e, (-(yin)))));

}

}

}

private void hasil(){

DefaultTableModel tabel2 = new DefaultTableModel();

int t1,t2;

int jumBenar=0;

String gangguanOut="";

String cek = "";

jTable2.setModel(tabel2);

tabel2.addColumn("Nama");

tabel2.addColumn("T1");

tabel2.addColumn("T2");

tabel2.addColumn("Target");

tabel2.addColumn("Y1");

tabel2.addColumn("Y2");

tabel2.addColumn("Output");

tabel2.addColumn("Hasil");

tabel2.getDataVector().removeAllElements();

tabel2.fireTableDataChanged();

int jumCol = tabel2.getColumnCount();

for(int i=0;i<jumData;i++){

//target

t1 = (int) Math.round(y[i][0]);

t2 = (int) Math.round(y[i][1]);

try{

Statement s = c.createStatement();

String sql = "SELECT \* FROM gangguan WHERE target1="+t1+" AND target2="+t2;

ResultSet r = s.executeQuery(sql);

while (r.next()) {

gangguanOut = r.getString("nama\_gangguan");

}

r.close();

s.close();

}catch (SQLException e){

System.out.println("Error Load Data Gangguan");

}

if((target[i][0]==t1)&&(target[i][1]==t2)){

cek = "Benar";

jumBenar++;

}else{

cek = "Salah";

}

Object[] obj = new Object[jumCol];

obj[0] = nama[i];

obj[1] = target[i][0];

obj[2] = target[i][1];

obj[3] = gangguan[i];

obj[4] = y[i][0];

obj[5] = y[i][1];

obj[6] = gangguanOut;

obj[7] = cek;

tabel2.addRow(obj);

}

double persen = (jumBenar\*100)/jumData;

jTextField2.setText(new DecimalFormat("##.##").format(persen)+" %");

}

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

home utama = new home();

this.dispose();

utama.setVisible(true);

}

private void jRadioButton1ActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

DefaultTableModel data1 = (DefaultTableModel)jTable1.getModel();

DefaultTableModel data2 = (DefaultTableModel)jTable2.getModel();

jRadioButton2.setSelected(false);

data1.setRowCount(0);

data2.setRowCount(0);

dataPilih("Data Latih");

jTable1.setEnabled(false);

}

private void formWindowOpened(java.awt.event.WindowEvent evt) {

// TODO add your handling code here:

tampilData();

}

private void jRadioButton2ActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

DefaultTableModel data1 = (DefaultTableModel)jTable1.getModel();

DefaultTableModel data2 = (DefaultTableModel)jTable2.getModel();

jRadioButton1.setSelected(false);

data1.setRowCount(0);

data2.setRowCount(0);

dataPilih("Data Awal");

jTable1.setEnabled(true);

}

private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

dataPengujian();

getBobotBias();

forwardPass();

hasil();

}

private void jTable1MouseClicked(java.awt.event.MouseEvent evt) {

// TODO add your handling code here:

int i = jTable1.getSelectedRow();

if((Boolean)jTable1.getValueAt(jTable1.getSelectedRow(), 0)==false){

jTable1.setValueAt(true, i, 0);

}else{

jTable1.setValueAt(false, i, 0);

}

}

From Prediksi

public class prediksi extends javax.swing.JFrame {

Connection cc = TugasAkhir.getConnection();

DefaultTableModel tabel = new DefaultTableModel();

String[] kategori = null;

String[] soal = null;

double[] bobot = null;

String[] a = null;

String[] b = null;

String[] c = null;

String[] d = null;

String[] e = null;

int[] jumPerKategori = null;

int jumSoal = 0;

int jumKategori = 0;

int pointer = 0;

int poin = 0;

String jawaban;

Boolean edit = false;

public prediksi() {

Dimension dim = Toolkit.getDefaultToolkit().getScreenSize();

int x = (dim.width-getWidth())/2;

int y = (dim.height-getHeight())/2;

setLocation(y, y);

initComponents();

}

public void tampilSoal(){

int i = 0;

jTable1.setModel(tabel);

//set table column

tabel.addColumn("Soal");

tabel.addColumn("Jawaban");

tabel.getDataVector().removeAllElements();

tabel.fireTableDataChanged();

//get data soal

try{

Statement s = cc.createStatement();

//get jumlah kategori soal

String jumKategoriSQL = "SELECT COUNT(kode\_kategori) AS jum\_kategori FROM kategori";

ResultSet r = s.executeQuery(jumKategoriSQL);

while (r.next()){

jumKategori = r.getInt("jum\_kategori");

}

r.close();

//get jumlah perKategori

jumPerKategori = new int[jumKategori];

String jumPerKategoriSQL = "SELECT COUNT(kode\_kategori) AS jum\_per\_kategori FROM pertanyaan GROUP BY kode\_kategori;";

r = s.executeQuery(jumPerKategoriSQL);

i = 0;

while (r.next()){

jumPerKategori[i] = r.getInt("jum\_per\_kategori");

i++;

}

r.close();

//get jumlah soal

String jumSQL = "SELECT COUNT(kode\_pertanyaan) AS jumlah FROM pertanyaan";

r = s.executeQuery(jumSQL);

while (r.next()){

jumSoal = r.getInt("jumlah");

}

r.close();

//assign array length

kategori = new String[jumSoal];

soal = new String[jumSoal];

bobot = new double[jumSoal];

a = new String[jumSoal];

b = new String[jumSoal];

c = new String[jumSoal];

d = new String[jumSoal];

e = new String[jumSoal];

//get pertanyaan

String sql = "SELECT \* FROM pertanyaan ORDER BY kode\_pertanyaan";

r = s.executeQuery(sql);

i = 0;

while (r.next()){

kategori[i] = r.getString("kode\_kategori");

soal[i] = r.getString("soal");

bobot[i] = 0.0;

a[i] = r.getString("pilihanA");

b[i]=r.getString("pilihanB");

c[i]=r.getString("pilihanC");

d[i]=r.getString("pilihanD");

e[i]=r.getString("pilihanE");

i++;

}

r.close();

s.close();

}catch(SQLException ee){

System.out.println(ee);

}

jTable1.setEnabled(false);

jLabel1.setText(""+(pointer+1));

jLabel9.setText(""+jumSoal);

jTextArea1.setText(soal[0]);

jRadioButton2.setText(a[0]);

jRadioButton1.setText(b[0]);

jRadioButton3.setText(c[0]);

jRadioButton4.setText(d[0]);

jRadioButton5.setText(e[0]);

}

public void radioEnable(){

jRadioButton1.setEnabled(true);

jRadioButton2.setEnabled(true);

jRadioButton3.setEnabled(true);

jRadioButton4.setEnabled(true);

jRadioButton5.setEnabled(true);

}

public void radioDisable(){

jRadioButton1.setEnabled(false);

jRadioButton2.setEnabled(false);

jRadioButton3.setEnabled(false);

jRadioButton4.setEnabled(false);

jRadioButton5.setEnabled(false);

}

public void pilih(){

if(pointer<jumSoal){

//set pilihan ke table

Object[] o = new Object[2];

o[0] = soal[pointer];

o[1] = jawaban;

tabel.addRow(o);

pointer++;

if(pointer==jumSoal){

jTextArea1.setText("SELESAI");

jLabel1.setText("-");

jTable1.setEnabled(true);

radioDisable();

}else{

jLabel1.setText(""+(pointer+1));

jTextArea1.setText(soal[pointer]);

}

}

if(edit){

jTable1.setValueAt(soal[poin], poin, 0);

jTable1.setValueAt(jawaban, poin, 1);

jTextArea1.setText("SELESAI");

jLabel1.setText("-");

jTable1.setEnabled(true);

radioDisable();

edit = false;

}

buttonGroup1.clearSelection();

}

private void jButton4ActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

home utama = new home();

this.dispose();

utama.setVisible(true);

}

private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {

int i,j,k;

pengujian p = new pengujian();

//get bobot dan bias

p.getBobotBias();

//assign all variable

int jumInput = p.jumInput;

int jumOutput = p.jumOutput;

int jumData = 1;

int jumHL = p.jumHL;

double max = p.max;

double[][] y;

//assign data

double[] jumBobot = new double[jumInput];

double[][] data = new double[jumData][jumInput];

//menjumlahkan bobot

k=0;

for(i=0;i<jumKategori;i++){

j = 0;

jumBobot[i] = 0.0;

while(j<jumPerKategori[i]){

jumBobot[i] += bobot[k];

j++;

k++;

}

System.out.println("jumBobot["+i+"]: "+jumBobot[i]);

}

//konversi data

for(i=0;i<jumKategori;i++){

data[0][i] = jumBobot[i];

}

//forward pass

final double e = 2.718281828;

double[][] z;

double zin,yin;

System.out.println("max:" +max);

/\*\* hitung dan cari nilai z \*\*/

zin = 0;

z = new double[jumData][jumHL];

for(i=0;i<jumData;i++){

for(j=0;j<jumHL;j++){

zin = 1\*p.bv[0][j];

for(k=0;k<jumInput;k++){

zin += ((data[i][k]/max)\*p.v[k][j]);

}

z[i][j] = 1/(1 + (Math.pow(e, (-(zin)))));

}

}

/\*\* hitung dan cari nilai y \*\*/

yin = 0;

y = new double[jumData][jumOutput];

for(i=0;i<jumData;i++){

for(j=0;j<jumOutput;j++){

yin = 1\*p.bw[0][j];

for(k=0;k<jumHL;k++){

yin += (z[i][k]\*p.w[k][j]);

}

y[i][j] = 1/(1 + (Math.pow(e, (-(yin)))));

}

}

//result

int t1,t2;

t1 = (int) Math.round(y[0][0]);

t2 = (int) Math.round(y[0][1]);

String hasil="";

try{

Statement s = cc.createStatement();

String sql = "SELECT \* FROM gangguan WHERE target1="+t1+" AND target2="+t2;

ResultSet r = s.executeQuery(sql);

while (r.next()) {

hasil = r.getString("nama\_gangguan");

}

r.close();

s.close();

}catch (SQLException ee){

System.out.println("Error Load Data Gangguan");

}

jTextField1.setText(hasil);

}

private void formWindowOpened(java.awt.event.WindowEvent evt) {

// TODO add your handling code here:

tampilSoal();

}

private void jRadioButton2ActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

if(edit){

bobot[poin] = 0.00;

}else{

bobot[pointer] = 0.00;

}

jawaban = jRadioButton2.getText();

pilih();

}

private void jRadioButton1ActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

if(edit){

bobot[poin] = 0.25;

}else{

bobot[pointer] = 0.25;

}

jawaban = jRadioButton1.getText();

pilih();

}

private void jRadioButton3ActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

if(edit){

bobot[poin] = 0.5;

}else{

bobot[pointer] = 0.5;

}

jawaban = jRadioButton3.getText();

pilih();

}

private void jRadioButton4ActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

if(edit){

bobot[poin] = 0.75;

}else{

bobot[pointer] = 0.75;

}

jawaban = jRadioButton4.getText();

pilih();

}

private void jRadioButton5ActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

if(edit){

bobot[poin] = 1.0;

}else{

bobot[pointer] = 1.0;

}

jawaban = jRadioButton5.getText();

pilih();

}

private void jTable1MouseClicked(java.awt.event.MouseEvent evt) {

// TODO add your handling code here:

poin = jTable1.getSelectedRow();

jTextArea1.setText(jTable1.getValueAt(poin, 0).toString());

jLabel1.setText(""+(poin+1));

radioEnable();

edit = true;

}