

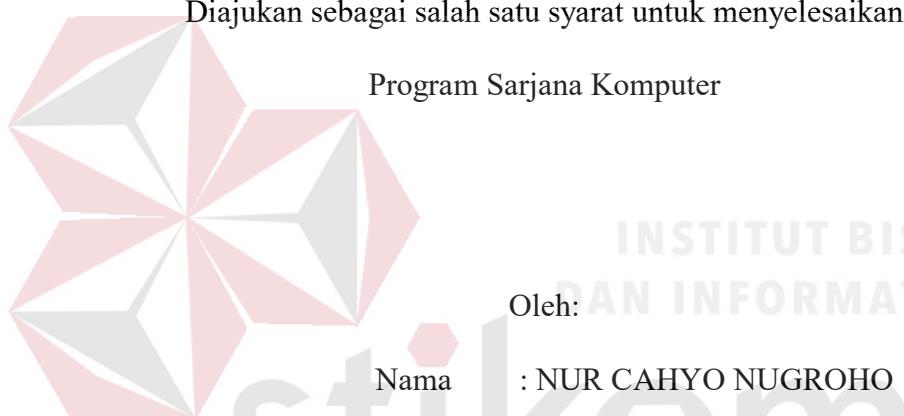
**SISTEM PERAMALAN KEWASPADAAN PENYAKIT MUSIMAN
DENGAN METODE EXPONENTIAL SMOOTHING OLEH WINTER
(STUDI KASUS PUSKESMAS KUPANG KABUPATEN MOJOKERTO)**



**SEKOLAH TINGGI
MANAJEMEN INFORMATIKA & TEKNIK KOMPUTER
SURABAYA
2007**

**SISTEM PERAMALAN KEWASPADAAN PENYAKIT MUSIMAN
DENGAN METODE EXPONENTIAL SMOOTHING OLEH WINTER
(STUDI KASUS PUSKESMAS KUPANG KABUPATEN MOJOKERTO)**

SKRIPSI



INSTITUT BISNIS
DAN INFORMATIKA

Oleh:

Nama : NUR CAHYO NUGROHO

NIM : 03.41010.0073

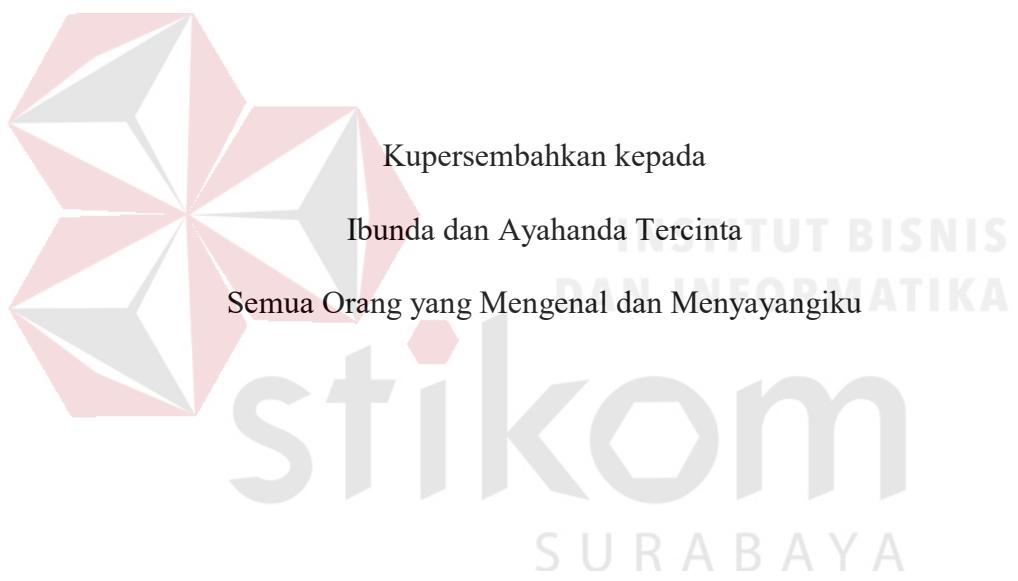
Program : S1 (Strata Satu)

Jurusan : Sistem Informasi

**SEKOLAH TINGGI
MANAJEMEN INFORMATIKA & TEKNIK KOMPUTER
SURABAYA
2007**



*Kalau bisa dikerjakan sendiri,
kenapa harus dikerjakan orang lain?*



ABSTRAKSI

Penyakit adalah salah satu hal yang tidak bisa dihindari oleh manusia dan kapan pun bisa menyerang kesehatan kita. Beberapa jenis penyakit hanya mewabah pada musim-musim tertentu. Apabila diketahui jenis penyakit yang akan berpotensi mewabah di periode depan, maka penanggulangan terhadap penyakit tadi bisa dipersiapkan lebih dini. Namun kebanyakan orang kurang sadar dan tidak mengetahui pola jenis penyakit tertentu yang akan mewabah di masa datang. Hal-hal tersebut merupakan alasan mengapa kita membutuhkan proses peramalan.

Berdasarkan data rekap penyakit bulanan periode sebelumnya, dapat diketahui pola penyakit yang mempunyai kecenderungan musiman. Dari data hasil seleksi ini, beberapa jenis penyakit yang mempunyai potensi besar terjadi di periode depan bisa diprediksi. Bagi pihak pelayanan kesehatan seperti puskesmas, hal ini bisa dimanfaatkan untuk memberikan penyuluhan lebih dini kepada masyarakat tentang cara-cara menanggulangi penyakit, selain itu kebutuhan obat dasar untuk periode depan bisa juga diprediksi. Hal ini bisa mengurangi potensi kekurangan stok obat.

Dalam Tugas Akhir ini, implementasi *Neural Network* metode *Backpropagation* digunakan untuk menghasilkan konstanta inputan bagi metode peramalan *Exponential Smoothing*. Dengan pemanfaatan metode ini, hasil ramalan bisa menghasilkan keakuratan yang lebih tinggi karena memiliki *Mean Square Error* (MSE) yang lebih kecil. Dengan demikian, hasil ramalan bisa digunakan sebagai pembantu dalam penanggulangan penyakit.

KATA PENGANTAR

Puji syukur atas kasih karunia dan anugerah-Nya, sehingga penulis dapat menyelesaikan tugas akhir ini, untuk memenuhi persyaratan dalam menyelesaikan Program Studi Strata Satu di Sekolah Tinggi Manajemen Informatika & Teknik Komputer Surabaya. Tugas Akhir ini membahas tentang pembuatan Sistem Peramalan Kewaspadaan Terhadap Penyakit Musiman Di Puskesmas Kupang Kabupaten Mojokerto.

Pada kesempatan ini, penulis menyampaikan rasa penghargaan dan terima kasih kepada yang terhormat :

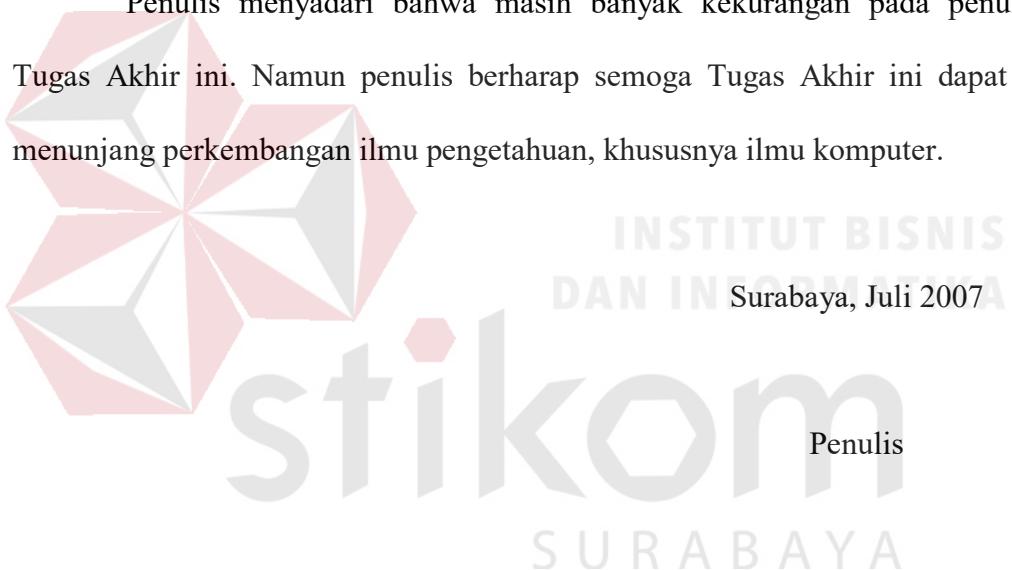
1. Bapak Romeo, ST. sebagai dosen pembimbing II atas segala arahan dan bimbingannya.
2. Ibu A B. Tjandrarini, S.Si., M.Kom. sebagai dosen pembimbing I atas segala bimbingan dan dukungannya.
3. Ayah dan ibu serta segenap keluarga di rumah atas segala dukungan moral, doa dan materi selama penggerjaan Tugas Akhir ini.
4. Ririn, Samvara, para penghuni pondok Tukimun Indah, arek-arek DNA dan MUGI, para angkatan 2001 yang super gokil (moolan_luv, bakol_jamu, yulisiswanto01) ketika di *chat room*, Made alias Ca_dek, serta penghuni kelas elite W1 angkatan 2003 atas segala bantuan ide-ide dan dukungannya selama penggerjaan Tugas Akhir ini.
5. Para dagelan di ruang 701 (WID, TOM, TYA, NOV, ITO, DEV), para sesepuh di 702 (PAC, YOP) beserta para punggawanya (TEA, SAR, LYD, ANN, CAH), serta semua rekan Labkom yang sering memberi penguatan

negatif dan tidak segan-segan menanyakan kapan Tugas Akhir ini selesai. Juga bapak Soetam Rizky Wicaksono, S.Kom. atas segala kesempatan dan bimbingan selama ini.

6. ROO yang selalu sabar dan bisa mengerti kondisiku ketika proses penggerjaan Tugas Akhir berlangsung. Serta koleksi anime dan *console* PSP-ku yang selalu setia menemaniku ketika lagi suntuk mengerjakan Tugas Akhir ini.

Semoga Allah SWT memberikan pahala yang setimpal kepada semua pihak yang telah memberikan bantuan, bimbingan ataupun nasehat-nasehat.

Penulis menyadari bahwa masih banyak kekurangan pada penulisan Tugas Akhir ini. Namun penulis berharap semoga Tugas Akhir ini dapat ikut menunjang perkembangan ilmu pengetahuan, khususnya ilmu komputer.



DAFTAR ISI

	Halaman
ABSTRAKSI	vi
KATA PENGANTAR	vii
DAFTAR ISI	ix
DAFTAR TABEL.....	xii
DAFTAR GAMBAR	xv
DAFTAR LAMPIRAN.....	xx
BAB I PENDAHULUAN	1
1.1 Latar Belakang Masalah	1
1.2 Perumusan Masalah	2
1.3 Pembatasan Masalah	2
1.4 Tujuan	3
1.5 Sistematika Penulisan	4
BAB II LANDASAN TEORI	6
2.1 Teknik Peramalan	6
2.1.1 Teknik Peramalan Kuantitatif	6
2.1.2 Data Musiman	7
2.1.3 Metode Exponential Smoothing oleh Winter	8
2.1.4 Pengukuran Kesalahan Peramalan	9
2.2 Jaringan Syaraf Tiruan	11
2.2.1 Algoritma Backpropagation	12
2.3 Pusat Kesehatan Masyarakat (Puskesmas)	13

	Halaman
BAB III PERANCANGAN SISTEM	14
3.1 Analisa Permasalahan	14
3.2 Desain Arsitektur	14
3.2.1 Arsitektur Neural Network	15
3.2.2 Arsitektur Proses Peramalan	16
3.3 Perancangan Proses	17
3.4 Alur Sistem (System Flow)	20
3.5 Entity Relationship Diagram (ERD)	30
3.5.1 Conceptual Data Model (CDM)	31
3.5.2 Physical Data Model (PDM)	32
3.6 Struktur Database	32
3.7 Desain Input-Output	37
3.8 Desain Uji Coba	47
3.8.1 Desain Uji Coba Fitur Dasar Sistem	47
3.8.2 Desain Uji Coba Proses Pelatihan NN dan Pengujian NN	59
3.8.3 Desain Uji Coba Proses Perbandingan Peramalan Dengan Bantuan NN	61
BAB IV IMPLEMENTASI DAN EVALUASI	62
4.1 Kebutuhan Sistem	62
4.2 Implementasi Sistem	62
4.3 Uji Coba dan Evaluasi	80
4.3.1 Uji Coba Fitur Dasar Sistem	80

Halaman

4.3.2	Uji Coba Pelatihan NN Dan Pengujian NN	108
4.3.3	Uji Coba Proses Perbandingan Peramalan Dengan Bantuan NN	113
4.3.4	Analisa Hasil Uji Coba	116
BAB V	PENUTUP	122
5.1	Kesimpulan	122
5.2	Saran	122
DAFTAR PUSTAKA		124
LAMPIRAN	125



DAFTAR TABEL

		Halaman
Tabel 3.1	Tabel Penyakit	32
Tabel 3.2	Tabel Login	33
Tabel 3.3	Tabel Obat	33
Tabel 3.4	Tabel Penyakit_Obat	33
Tabel 3.5	Tabel Hasil_Klasifikasi	34
Tabel 3.6	Tabel Rekap_Penyakit_Bulanan	34
Tabel 3.7	Tabel Panjang_Ramalan_Terbaik	34
Tabel 3.8	Tabel Pola_NN_Terbaik	35
Tabel 3.9	Tabel Data_Latih	35
Tabel 3.10	Tabel Ramalan_Exp_Terbaik	36
Tabel 3.11	Tabel Ramalan_NN_Terbaik	36
Tabel 3.12	Tabel Login	48
Tabel 3.13	Tabel Test Case Login	48
Tabel 3.14	Tabel Penyakit	48
Tabel 3.15	Tabel Test Case Manipulasi Data Penyakit	49
Tabel 3.16	Tabel Obat	50
Tabel 3.17	Tabel Test Case Manipulasi Data Obat	50
Tabel 3.18	Tabel Login	52
Tabel 3.19	Tabel Test Case Manipulasi Data Login	52
Tabel 3.20	Tabel Hasil_Klasifikasi	53
Tabel 3.21	Tabel Test Case Pengambilan Data	54
Tabel 3.22	Tabel Test Case Peramalan	54

	Halaman	
Tabel 3.23	Tabel Test Case Pelatihan	55
Tabel 3.24	Tabel Test Case Pencarian Pola NN Terbaik	56
Tabel 3.25	Tabel Test Case Pencarian Panjang Ramalan Terbaik	57
Tabel 3.26	Tabel Test Case Perbandingan Hasil Ramalan	58
Tabel 3.27	Tabel Test Case Laporan	59
Tabel 3.28	Tabel Test Case Pelatihan NN	60
Tabel 3.29	Tabel Test Case Pengujian NN	60
Tabel 3.30	Tabel Test Case Pencarian Pola NN Terbaik	60
Tabel 3.31	Tabel Test Case Pencarian Panjang Periode Ramal Terbaik	60
Tabel 3.32	Tabel Test Case Perbandingan Peramalan	61
Tabel 4.1	Tabel Login	81
Tabel 4.2	Tabel Hasil Test Case Login	81
Tabel 4.3	Tabel Penyakit	82
Tabel 4.4	Tabel Hasil Test Case Manipulasi Data Penyakit	82
Tabel 4.5	Tabel Obat	84
Tabel 4.6	Tabel Test Case Manipulasi Data Obat	85
Tabel 4.7	Tabel Login	86
Tabel 4.8	Tabel Hasil Test Case Manipulasi Data Login	87
Tabel 4.9	Tabel Hasil_Klasifikasi	89
Tabel 4.10	Tabel Hasil Test Pengambilan Data	89
Tabel 4.11	Tabel Hasil Test Case Peramalan	91
Tabel 4.12	Tabel Hasil Test Case Pelatihan	92
Tabel 4.13	Tabel Hasil Test Case Pencarian Pola NN Terbaik	95

Tabel 4.14	Tabel Hasil Test Case Pencarian Panjang Ramalan Terbaik	98
Tabel 4.15	Tabel Hasil Test Case Perbandingan Hasil Ramalan	102
Tabel 4.16	Tabel Test Case Laporan	105
Tabel 4.17	Tabel Data Penyakit Diuji	108
Tabel 4.18	Tabel Data Latih	109
Tabel 4.19	Tabel Target Output Data Latih	109
Tabel 4.20	Tabel Hasil Test Case Pelatihan NN	110
Tabel 4.21	Tabel Iterasi dan MSE Proses Pelatihan NN	110
Tabel 4.22	Tabel Hasil Test Case Proses Pengujian NN	111
Tabel 4.23	Tabel Output Pelatihan NN	111
Tabel 4.24	Tabel Hasil Test Case Proses Pencarian Pola NN Terbaik	112
Tabel 4.25	Tabel Hasil Test Case Proses Pencarian Ramalan Terbaik.....	112
Tabel 4.26	Tabel Data Penyakit Diuji	113
Tabel 4.27	Tabel Data Time Series Diuji	114
Tabel 4.28	Tabel Output Konstanta Diharapkan	114
Tabel 4.29	Tabel Hasil Test Case Perbandingan Peramalan	114
Tabel 4.30	Tabel Output Konstanta Proses Perbandingan	115
Tabel 4.31	Tabel Pengaruh Panjang Periode Ramalan Terhadap MSE	117
Tabel 4.32	Tabel Data Skenario Diuji	118
Tabel 4.33	Tabel Pengaruh Banyak Iterasi Pelatihan Ramalan Terhadap MSE dan MAD	118
Tabel 4.34	Tabel Pengaruh Pola NN Ramalan Pada MSE dan MAD (1) ...	119
Tabel 4.35	Tabel Pengaruh Pola NN Ramalan Pada MSE dan MAD (2) ...	120

DAFTAR LAMPIRAN

Halaman

Lampiran 1	Lampiran List Program	125
Lampiran 2	Biodata Penulis	167



BAB I

PENDAHULUAN

1.1 Latar Belakang Masalah

Puskesmas Kupang adalah puskesmas yang melayani wilayah kecamatan Jetis di kabupaten Mojokerto. Setiap pasien datang dicatat oleh bagian loket untuk mengetahui jenis keluhan penyakit. Berdasarkan data ini, rekap bulanan penyakit yang diderita di wilayah puskesmas tadi bisa diketahui.

Selama ini pihak puskesmas hanya sebatas membuat rekap dan grafik penyakit bulanan tanpa ada pengolahan data sehingga pihak puskesmas kesulitan dalam meramalkan penyakit musiman yang akan muncul pada periode mendatang. Hal ini terbukti dengan kurangnya stok obat ataupun peralatan medis yang muncul pada rekap stok obat bulanan. Berdasarkan permasalahan yang ada akan dibuat sistem peramalan untuk menghasilkan laporan penyakit yang harus diwaspadai pada periode tertentu. Dengan adanya peramalan ini, pihak puskesmas diharapkan bisa lebih siap dan waspada dalam menyediakan layanan medis serta obat yang digunakan.

Setelah melakukan analisis terhadap rekap penyakit bulanan, didapatkan beberapa data penyakit yang memiliki pola musiman. Berdasarkan beberapa penelitian dan studi literatur yang ada, salah satu metode peramalan yang tepat digunakan adalah pemulusan eksponensial (*exponential smoothing*). Pada metode pemulusan eksponensial dibutuhkan suatu konstanta pemulusan yang baik untuk mengurangi nilai *Mean Square Error* (MSE) dari hasil peramalan. Oleh karena itu, pada sistem ini digunakan *neural network* algoritma *backpropagation* untuk

membantu menghasilkan nilai konstanta yang paling tepat bagi konstanta pemulusan. Dari sistem peramalan ini akan didapatkan laporan berupa jenis penyakit musiman yang perlu diwaspadai pada periode mendatang beserta penanganan yang bisa diambil.

1.2 Perumusan Masalah

Dari uraian pada latar belakang masalah, maka perumusan masalahnya adalah sebagai berikut:

1. Bagaimana menganalisis penyakit-penyakit dari rekap penyakit bulanan yang mengandung pola data musiman untuk digunakan dalam proses peramalan di masa mendatang.
2. Bagaimana membangun dan merancang sistem dengan *neural network* algoritma *backpropagation* untuk menghasilkan konstanta pemulusan yang tepat.
3. Bagaimana membangun dan merancang sistem peramalan kewaspadaan terhadap penyakit musiman di cakupan wilayah puskesmas Kupang kabupaten Mojokerto dengan menggunakan metode *exponential smoothing* yang menggunakan nilai konstanta pemulusan yang dihasilkan alogoritma *backpropagation*.

1.3 Pembatasan Masalah

Dalam sistem ini, agar tidak menyimpang dari tujuan yang akan dicapai maka pembahasan masalah dibatasi pada hal-hal sebagai berikut:

1. Aplikasi ini berbasis *desktop* yang dibuat dengan menggunakan bahasa pemrograman VB.Net dan database SQL Server 2000.

2. Sistem dapat digunakan oleh user berikut :
 - a. Petugas medis (dokter) serta bagian penyuluhan masyarakat untuk mengetahui jenis penyakit apa saja yang akan mewabah pada periode bulan berikutnya.
 - b. Petugas apotek untuk mengetahui jenis dan jumlah obat yang akan dipesan ke Dinas Kesehatan Kabupaten Mojokerto.
3. Aplikasi ini tidak menangani masalah sekuritas jaringan.
4. Data aktual yang dipakai untuk meramalkan penyakit musiman adalah data rekap penyakit bulanan pada puskesmas Kupang kabupaten Mojokerto mulai Januari 2004 sampai Desember 2006 yang mempunyai pola data musiman.
5. Jumlah jenis penyakit yang digunakan untuk sekali proses peramalan periode selanjutnya adalah 10 penyakit yang mempunyai kecenderungan mewabah.
6. Aplikasi peramalan ini tidak menyangkut faktor-faktor penyebab suatu penyakit bisa mewabah pada periode mendatang.
7. Metode pengolahan data yang dipakai untuk memperoleh informasi tentang jenis penyakit musiman yang akan mewabah pada periode mendatang adalah *exponential smoothing* oleh Winter dan *neural network* algoritma *backpropagation*.

1.4 Tujuan

Tujuan dari tugas akhir ini adalah :

1. Memanfaatkan data rekap penyakit bulanan yang memiliki pola data musiman sebagai data dasar ramalan untuk proses peramalan.

2. Membuat aplikasi yang bisa menghasilkan konstanta inputan terbaik bagi proses peramalan *exponential smoothing* dengan memanfaatkan *neural network* metode *backpropagation*.
3. Membuat aplikasi peramalan dengan metode *exponential smoothing* untuk menghasilkan ramalan jumlah penyakit tertentu yang akan mewabah di periode mendatang.

1.5 Sistematika Penulisan

Penulisan tugas akhir yang berjudul “Sistem Peramalan Kewaspadaan terhadap Penyakit Musiman di Puskesmas Kupang Kabupaten Mojokerto” akan dibahas pada bagian ini. Terdapat lima bab yang masing-masing telah dikelompokkan sesuai pembahasan. Setiap bab akan dijelaskan secara detail di bawah ini.

Bab pertama berisi tentang pendahuluan yang merupakan gambaran umum penulisan. Pada bagian ini dijelaskan tentang latar belakang masalah, perumusan masalah, pembatasan masalah, tujuan, dan keterangan mengenai sistematika penulisan.

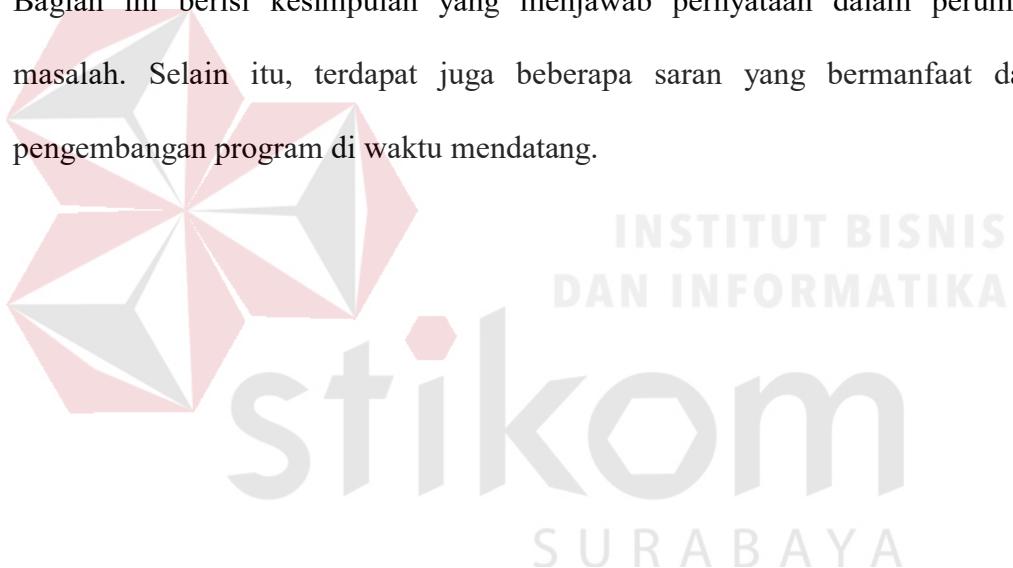
Bab kedua berisi tentang landasan teori yang digunakan dalam tugas akhir. Bagian ini menjelaskan teori yang mendukung pokok pembahasan tugas akhir yang meliputi definisi dan konsep peramalan. Selain itu, terdapat juga konsep *Exponential Smoothing* oleh Winter, *Neural Network* metode *backpropagation*, dan konsep dari puskesmas.

Bab ketiga berisi tentang perancangan sistem aplikasi tugas akhir yang dibuat. Bagian ini menjelaskan mengenai analisis permasalahan yang terjadi pada

puskesmas. Terdapat juga desain arsitektur, perancangan proses, alur sistem, struktur database, dan desain input output.

Bab keempat berisi tentang implementasi dan evaluasi dari aplikasi tugas akhir. Bagian ini menjelaskan tentang implementasi program yang berisi langkah-langkah implementasi perancangan program dan hasil implementasi program. Analisis terhadap kinerja program juga dibahas setelah melakukan proses implementasi.

Bab kelima adalah bagian penutup laporan tugas akhir yang dibuat. Bagian ini berisi kesimpulan yang menjawab pernyataan dalam perumusan masalah. Selain itu, terdapat juga beberapa saran yang bermanfaat dalam pengembangan program di waktu mendatang.



BAB II

LANDASAN TEORI

2.1 Teknik Peramalan

Teknik peramalan diperlukan untuk menetapkan kapan suatu peristiwa akan terjadi atau timbul sehingga tindakan yang tepat dapat dilakukan. Peramalan merupakan bagian integral dari kegiatan pengambilan keputusan manajemen. Organisasi selalu menentukan saran dan tujuan, berusaha menduga faktor-faktor lingkungan, lalu memilih tindakan yang diharapkan akan menghasilkan pencapaian sasaran dan tujuan tersebut. Kebutuhan akan peramalan meningkat sejalan dengan usaha manajemen untuk mengurangi ketergantungan pada hal-hal yang belum pasti (Makridakis, Wheelwright, McGee, 1999: 14-15).

2.1.1 Teknik Peramalan Kuantitatif

Teknik peramalan kuantitatif digunakan jika data historis tersedia cukup memadai dan jika data tersebut dianggap cukup representatif untuk meramalkan masa depan. Teknik peramalan kuantitatif terdapat 2 jenis teknik yaitu:

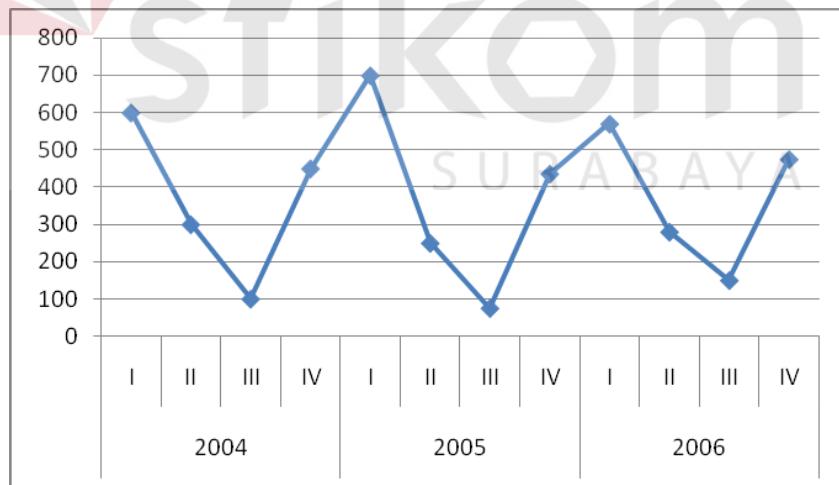
- a. Teknik Statistik, menitikberatkan pada pola, perubahan pola, dan faktor gangguan yang disebabkan oleh pengaruh acak. Contoh dari teknik statistik adalah: *trend linier, single moving average, double moving averages, exponential smoothing*, dan lain-lain.
- b. Teknik Deterministik, mencakup identifikasi dan penentuan hubungan antara variabel yang akan diprakirakan dengan variabel-variabel lain yang mempengaruhinya. Teknik ini mencakup regresi sederhana dan regresi

berganda, indikator-indikator utama, model ekonometrik, survey antisipasi, dan model input-output (Arsyad, 2001: 36-37).

2.1.2 Data Musiman

Data musiman adalah data berpola deret yang dipengaruhi oleh faktor musiman seperti kuartal tahun tertentu, bulan tertentu, atau hari tertentu. Pengaruh dari pola data ini bisa berupa iklim cuaca ataupun pengaruh masa-masa yang berulang tiap tahunnya seperti masa liburan dan hari raya.

Beberapa metode peramalan yang bisa dipergunakan untuk data musiman antara lain: metode dekomposisi klasik, metode Census II, pemulusan eksponensial (*exponential smoothing*) dari Winter, dan metode Box-Jenkins (Arsyad, 2001: 53). Berikut contoh grafik pola data musiman yang berulang setiap tahun dalam bentuk kuartal bulan.



Gambar 2.1 Grafik Pola Data Musiman

2.1.3 Metode *Exponential Smoothing* oleh Winter

Metode yang mengulang perhitungan secara terus menerus dengan menggunakan data terbaru. Metode ini berdasarkan pada perhitungan rata-rata pemulusan data-data masa lalu secara eksponensial. Setiap data diberi bobot dengan data lebih baru memiliki bobot lebih besar. Bobot yang digunakan adalah α untuk data paling baru, $\alpha(1-\alpha)$ digunakan untuk data yang agak lama, $\alpha(1-\alpha)^2$ untuk data yang lebih lama lagi, dan seterusnya

Secara matematis, persamaan pemulusan eksponensial dapat dituliskan sebagai berikut:

$$\hat{Y}_{t+1} = \alpha Y_t + (1-\alpha) \hat{Y}_t$$

Keterangan:

\hat{Y}_{t+1} = nilai ramalan untuk periode berikutnya

α = konstanta pemulusan ($0 < \alpha < 1$)

Y_t = data baru atau nilai aktual pada periode t

\hat{Y}_t = nilai pemulusan yang lama atau rata-rata yang dimuluskan hingga periode t-1

Metode *exponential smoothing* yang digunakan untuk pola data musiman adalah yang sesuai dikemukakan oleh Winter. Ada beberapa proses persamaan yang digunakan oleh metode ini.

a. Pemulusan Eksponensial

$$A_t = \alpha \frac{Y_t}{S_{t-L}} + (1-\alpha)(A_{t-1} + T_{t-1}) \quad (2.1)$$

b. Estimasi Trend

$$T_t = \beta(A_t - A_{t-1}) + (1 - \beta)T_{t-1} \quad (2.2)$$

c. Estimasi Musiman

$$S_t = \theta \frac{Y_t}{A_t} + (1 - \theta)S_{t-L} \quad (2.3)$$

d. Ramalan untuk Periode p di Masa Datang

$$\hat{Y}_{t+p} = (A_t + p T_t)S_{t-L+p} \quad (2.4)$$

Keterangan:

A_t = nilai pemulusan yang baru

α = konstanta pemulusan untuk data ($0 \leq \alpha \leq 1$)

Y_t = data baru atau nilai aktual pada periode t

T_t = estimasi trend

β = konstanta pemulusan untuk estimasi trend ($0 \leq \beta \leq 1$)

S_t = estimasi musiman

θ = konstanta pemulusan untuk estimasi musiman ($0 \leq \theta \leq 1$)

p = periode yang diramalkan

L = panjang musim

\hat{Y}_{t+p} = peramalan pada periode p

2.1.4 Pengukuran Kesalahan Peramalan

Dalam pemodelan deret berkala, sebagian data yang diketahui dapat digunakan untuk meramalkan sisa data berikutnya sehingga dapat dilakukan perhitungan ketepatan peramalan secara lebih baik. Ketepatan peramalan pada masa yang akan datang adalah yang sangat penting.

Jika X_i merupakan data aktual untuk periode i dan F_i merupakan ramalan untuk periode yang sama, maka kesalahannya dapat dituliskan sebagai berikut (Spyros, 1991: 40):

$$e_i = X_i - F_i$$

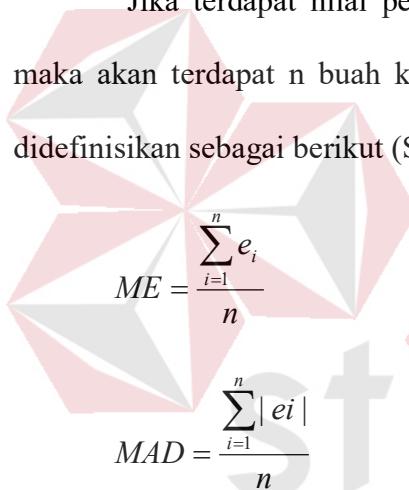
Keterangan:

e_i = kesalahan pada periode i

X_i = data aktual pada periode i

F_i = peramalan pada periode i

Jika terdapat nilai pengamatan dan peramalan untuk n periode waktu, maka akan terdapat n buah kesalahan dan ukuran statistik standart yang dapat didefinisikan sebagai berikut (Spyros, 1991: 41):



$$ME = \frac{\sum_{i=1}^n e_i}{n}$$

$$MAD = \frac{\sum_{i=1}^n |e_i|}{n}$$

$$MSE = \frac{\sum_{i=1}^n e_i^2}{n}$$

(2.5)

$$MSE = \frac{\sum_{i=1}^n e_i^2}{n}$$

(2.6)

Keterangan:

ME = *Mean Error* atau nilai tengah kesalahan

MAD = *Mean Absolute Deviation* atau nilai tengah kesalahan absolute deviasi

MSE = *Mean Squared Error* atau nilai kesalahan kuadrat

Selain ukuran ketepatan yang ada di atas, ada juga ukuran-ukuran ketepatan relatif yang di antaranya menyangkut kesalahan prosentase. Tiga ukuran relatif yang digunakan adalah:

$$PE_i = \left(\frac{X_i - F_i}{X_i} \right) \times 100$$

$$MPE = \frac{\sum_{i=1}^n PE_i}{n}$$

$$MAPE = \frac{\sum_{i=1}^n |PE_i|}{n}$$

Keterangan:

PE_i = Percentage Error atau kesalahan prosentase

MPE = Mean Percentage Error atau nilai tengah kesalahan prosentase

$MAPE$ = Mean Absolute Percentage Error atau nilai tengah kesalahan prosentase absolut

2.2 Jaringan Syaraf Tiruan

Jaringan syaraf adalah salah satu representasi buatan dari otak manusia yang selalu mencoba untuk mensimulasikan proses pembelajaran pada otak manusia tersebut. Istilah buatan disini digunakan karena jaringan syaraf ini diimplementasikan dengan menggunakan program komputer yang mampu menyelesaikan sejumlah proses perhitungan selama proses pembelajaran.

Setiap sel syaraf (*neuron*) akan memiliki satu inti sel, inti sel ini nanti yang akan bertugas untuk melakukan pemrosesan informasi. Informasi yang akan diterima oleh dendrit. Selain menerima informasi, dendrit juga menyertai axon sebagai keluaran dari suatu pemrosesan informasi. Informasi hasil olahan ini akan menjadi masukan bagi neuron lain dan antar dendrit kedua sel tersebut dipertemukan dengan synapsis. Informasi yang dikirimkan antar neuron ini berupa

rangsangan yang dilewatkan melalui dendrit. Informasi yang datang dan diterima oleh dendrit akan dijumlahkan dan dikirim melalui axon ke dendrit akhir yang bersentuhan dengan dendrit dari neuron yang lain. Informasi ini akan diterima oleh neuron lain jika memenuhi batasan tertentu, yang sering dikenal dengan nama nilai ambang (*threshold*) yang dikatakan teraktivasi (Siang, 2005: 2-5).

2.2.1 Algoritma *Backpropagation*

Algoritma jaringan syaraf tiruan yang digunakan dalam sistem ini adalah algoritma backpropagation. Backpropagation adalah pelatihan jenis terkontrol (*supervised*) yang menggunakan pola penyesuaian bobot untuk mencapai nilai kesalahan minimum antara keluaran hasil prediksi dengan keluaran nyata. Persamaan untuk mencari nilai kesalahan adalah sebagai berikut:

$$e = \frac{1}{2} \| d - y \|^2 \quad (2.7)$$

Dengan d adalah keluaran yang terjadi, y adalah keluaran hasil prediksi, dan e adalah kesalahan yang akan diminimumkan. Pelatihan backpropagation merupakan suatu penyesuaian bobot dengan memperhatikan kesalahan keluaran yang nyata dengan hasil prediksi. Pada prinsipnya bobot awal diambil secara acak kemudian bobot tersebut berubah ke arah bawah yang disebut *gradient descent*. Hal tersebut dilakukan secara berulang hingga harga kesalahan akan sama dengan nol.

Dengan menggunakan deferensial pada persamaan fungsi aktivasi sigmoid, fungsi keluaran Y dan fungsi keluaran lapisan tersembunyi serta persamaan kesalahan dan gradient. Maka deferensial kesalahan terhadap bobot w diberikan pada persamaan di bawah ini:

$$\frac{\delta e}{\delta w_{o_{jk}}} = (d - y(w)).y(w).(1 - y(w))h_j(w) \quad (2.8),$$

sedangkan deferensial kesalahan terhadap bobot w_i diberikan pada persamaan di bawah ini:

$$\frac{\delta e}{\delta w_{i_{ij}}} = (d - y(w)).y(w).(1 - y(w)) \left\{ \sum_{j=1}^h w_{o_{jk}} \frac{\delta h_j}{\delta w_{i_{ij}}} \right\} \quad (2.9),$$

sehingga harga bobot yang baru akan sama dengan persamaan berikut ini:

$$w_{baru} = w_{lama} + \lambda(1 - \alpha)\Delta w + \alpha \cdot \Delta w_{lama} \quad (2.10),$$

dengan α adalah suatu konstanta belajar yang diberikan, Δw adalah gradient kesalahan persamaan deferensial di atas (Kusumadewi dan Hartati, 2006: 93-98).

2.3 Pusat Kesehatan Masyarakat (Puskesmas)

Pusat Kesehatan Masyarakat, disingkat Puskesmas adalah tempat pelayanan kesehatan bagi masyarakat yang terletak di tingkat kecamatan di Indonesia. Pengelolaan Puskesmas umumnya berada di bawah Dinas Kesehatan Kabupaten/Kota. (Wikipedia, 2006: <http://id.wikipedia.org/wiki/Puskesmas>)



BAB III

PERANCANGAN SISTEM

3.1 Analisa Permasalahan

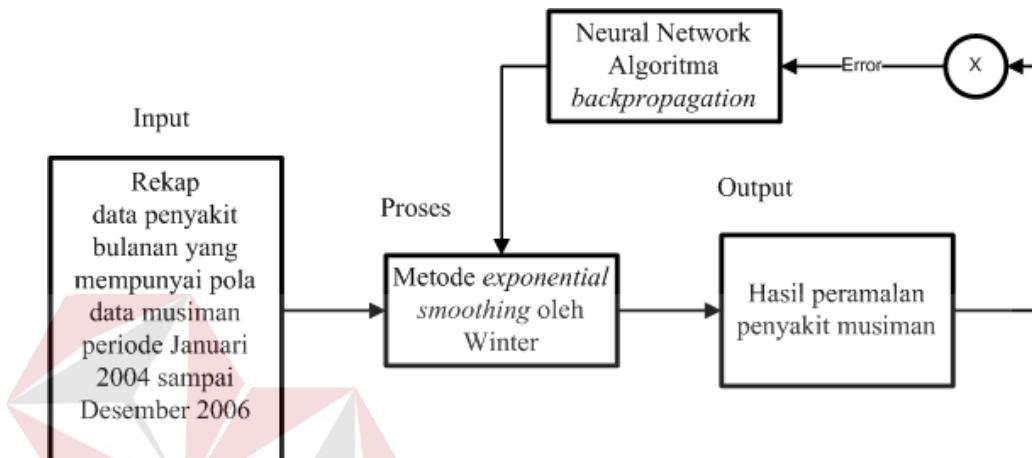
Aplikasi peramalan dengan metode *Exponential Smoothing* oleh Winter yang digunakan dalam tugas akhir ini menggunakan tiga konstanta untuk melakukan proses peramalan. Permasalahan yang muncul adalah kita tidak mengetahui berapa nilai konstanta yang terbaik, oleh karena itu terkadang digunakan nilai acak untuk mengisi konstanta tadi atau dengan cara melakukan *trial and error* yang jumlahnya bisa sangat banyak. Oleh karena itu, tugas akhir ini berusaha untuk menghasilkan nilai konstanta dari proses ramalan dengan bantuan *Neural Network* algoritma *Backpropagation*.

Proses dari *Neural Network* dilakukan sebelum melakukan proses peramalan yang berfungsi untuk menentukan nilai konstanta dengan kemungkinan menghasilkan *Mean Square Error* (MSE) yang paling kecil. Dengan MSE yang lebih kecil, berarti nilai peramalan semakin mendekati kebenaran. Di akhir proses, terdapat perbandingan antara hasil peramalan dengan cara konvensional penentuan konstanta dan dengan peramalan yang menggunakan konstanta hasil perhitungan *Neural Network*.

3.2 Desain Arsitektur

Aplikasi ini dibangun dalam bentuk aplikasi desktop (*desktop application*) dan menggunakan bahasa pemrograman Visual Basic.Net. Desain arsitektur seperti terlihat pada gambar 3.1 menjelaskan bahwa rekap data penyakit bulanan yang mempunyai pola data musiman akan diproses untuk menghasilkan

peramalan dengan menggunakan *exponential smoothing*. Algoritma *backpropagation* membantu dalam hal penentuan nilai konstanta pemulusan yang tepat agar menghasilkan nilai peramalan yang mempunyai nilai kesalahan yang terendah.

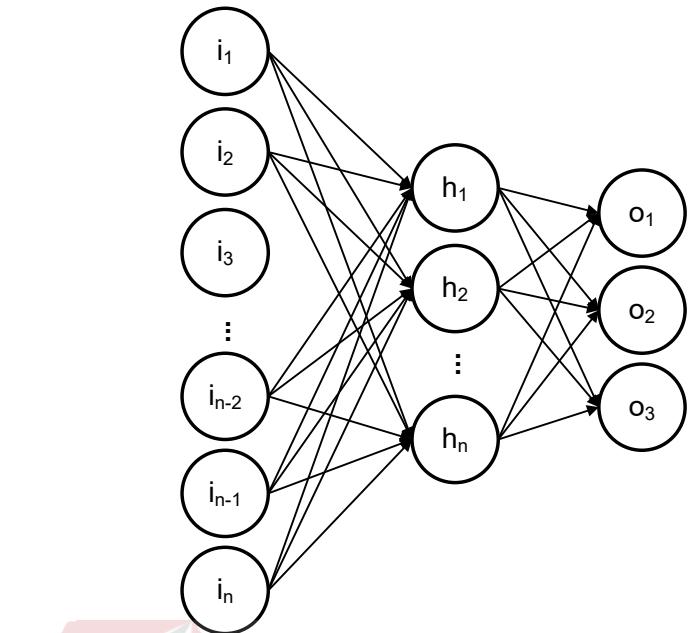


Gambar 3.1 Proses Sistem Secara Global

3.2.1 Arsitektur Neural Network

Pola *neural network* terdiri dari tiga bagian yaitu layer *input*, layer *hidden*, dan layer *output*. Layer *input* terdiri minimal dari tujuh neuron yang terdiri dari 5 data inputan *time series* dan 2 neuron yang berupa panjang ramalan dan panjang periode.

Layer *hidden* terdiri dari pilihan kelipatan 10 yang dimulai dari 1, 10, 20, 30, 40, 50, 60, 70, 80, 90 sampai 100 neuron. Sedangkan layer *output* terdiri dari 3 neuron yang akan menghasilkan nilai untuk tiga konstanta peramalan *exponential smoothing*. Pada proses selanjutnya, terdapat proses untuk memilih pola jaringan yang terbaik untuk tiap penyakit. Sehingga tiap penyakit memiliki pola jaringan yang berbeda. Gambar 3.2 menunjukkan arsitektur pola *neural network*.



Gambar 3.2 Arsitektur Pola Neural Network

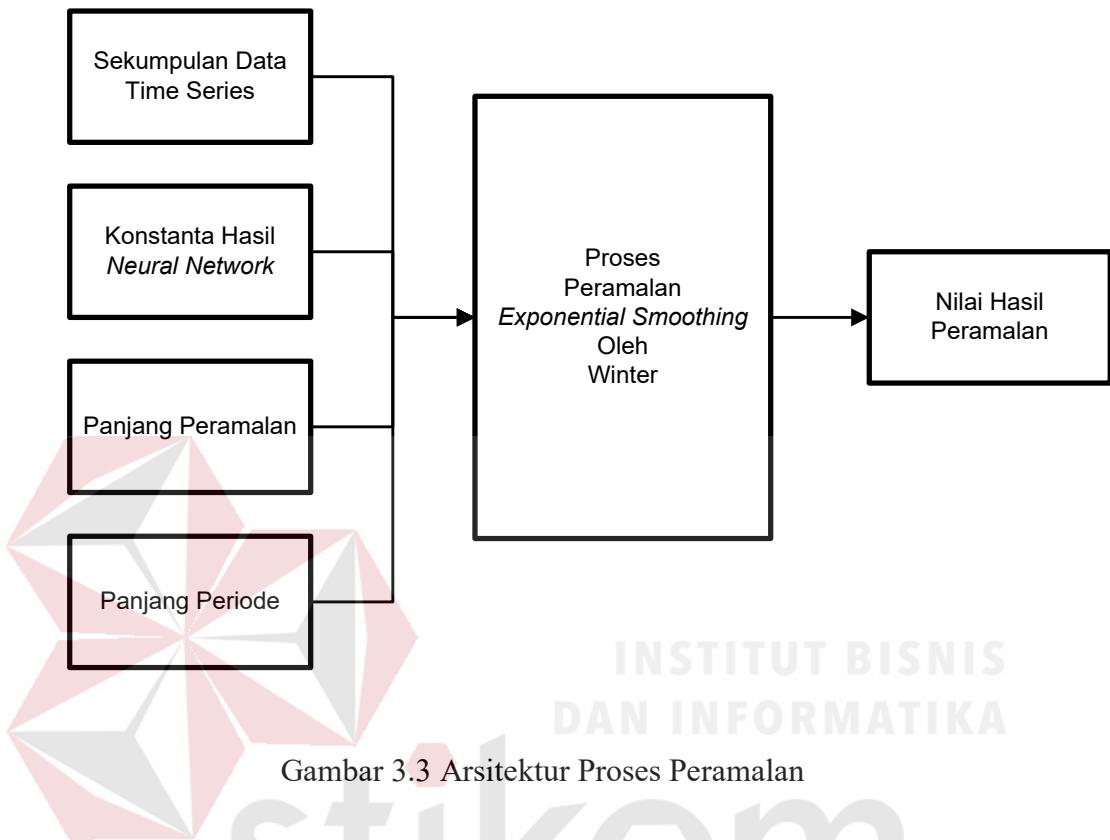
Lingkaran dengan nilai i adalah *neuron input* yang merupakan data *time series* berupa data actual rekap penyakit. Sedangkan i_{n-2} dan i_{n-1} merupakan panjang ramalan dan panjang periode.

Lingkaran dengan nilai h merupakan *neuron hidden* yang berupa kelipatan 10 seperti dijelaskan di atas. Sedangkan lingkaran dengan nilai o merupakan neuron output yang berjumlah 3 mewakili masing-masing konstanta untuk peramalan *exponential smoothing* oleh Winter.

3.2.2 Arsitektur Proses Peramalan

Setelah mendapatkan konstanta dari proses *neural network*, proses peramalan bisa dilakukan. Setelah terdapat sekumpulan data *time series*, konstanta peramalan, panjang peramalan, dan panjang periode, maka proses peramalan akan dapat dilakukan untuk menghasilkan nilai peramalan penyakit berpola musiman

yang akan mewabah pada periode selanjutnya. Berikut desain arsitektur proses peramalan pada tugas akhir ini.



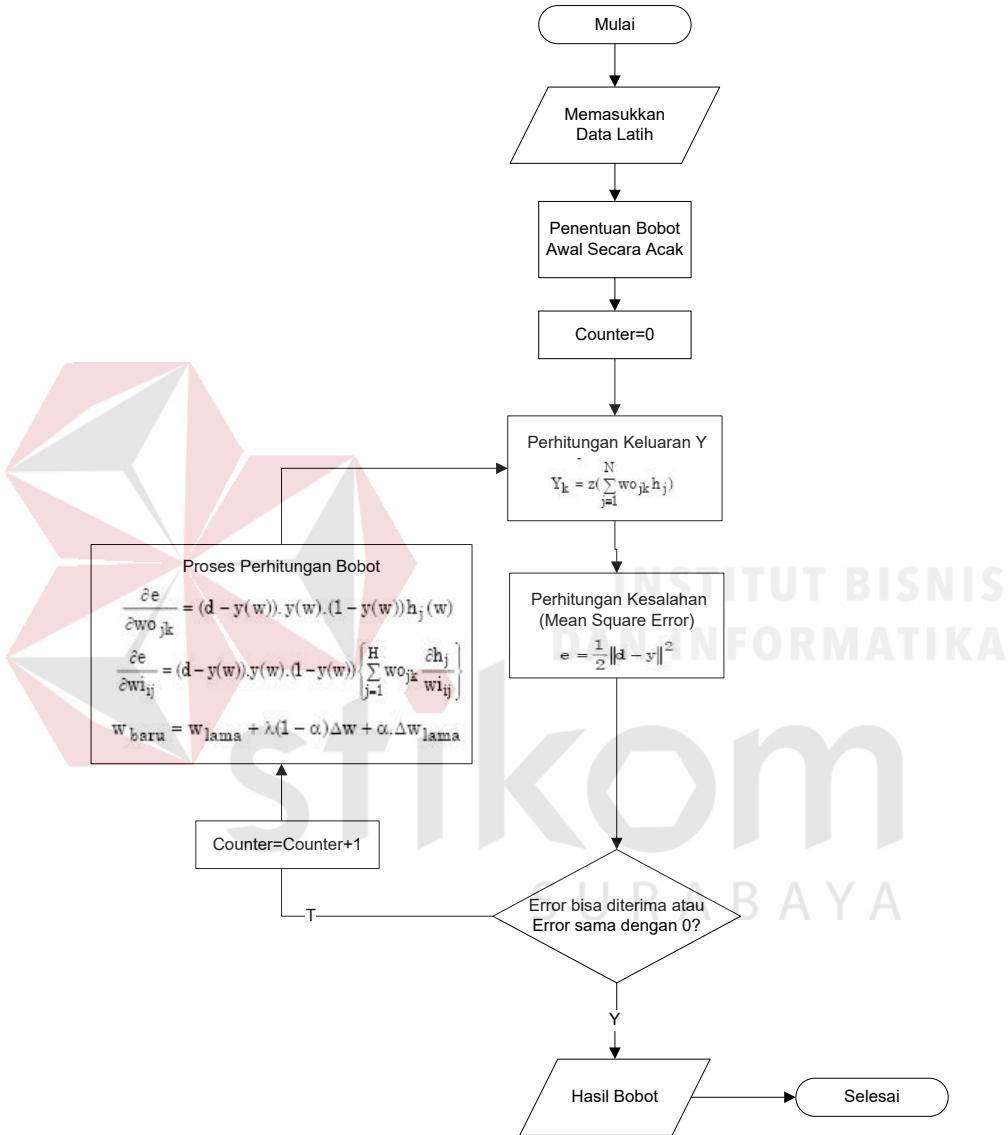
3.3 Perancangan Proses

Perancangan proses dalam sistem ini adalah sebagai berikut:

1. Diagram alir proses pelatihan dengan *backpropagation*
2. Diagram alir proses peramalan dengan *exponential smoothing*

Pada gambar 3.4 menjelaskan proses pelatihan data dengan metode *backpropagation*. Proses ini dimulai dengan memasukkan data latih berupa data rekap penyakit bulanan. Proses perhitungan bobot dilakukan sampai *error* yang terjadi bisa diterima atau sama dengan nol.

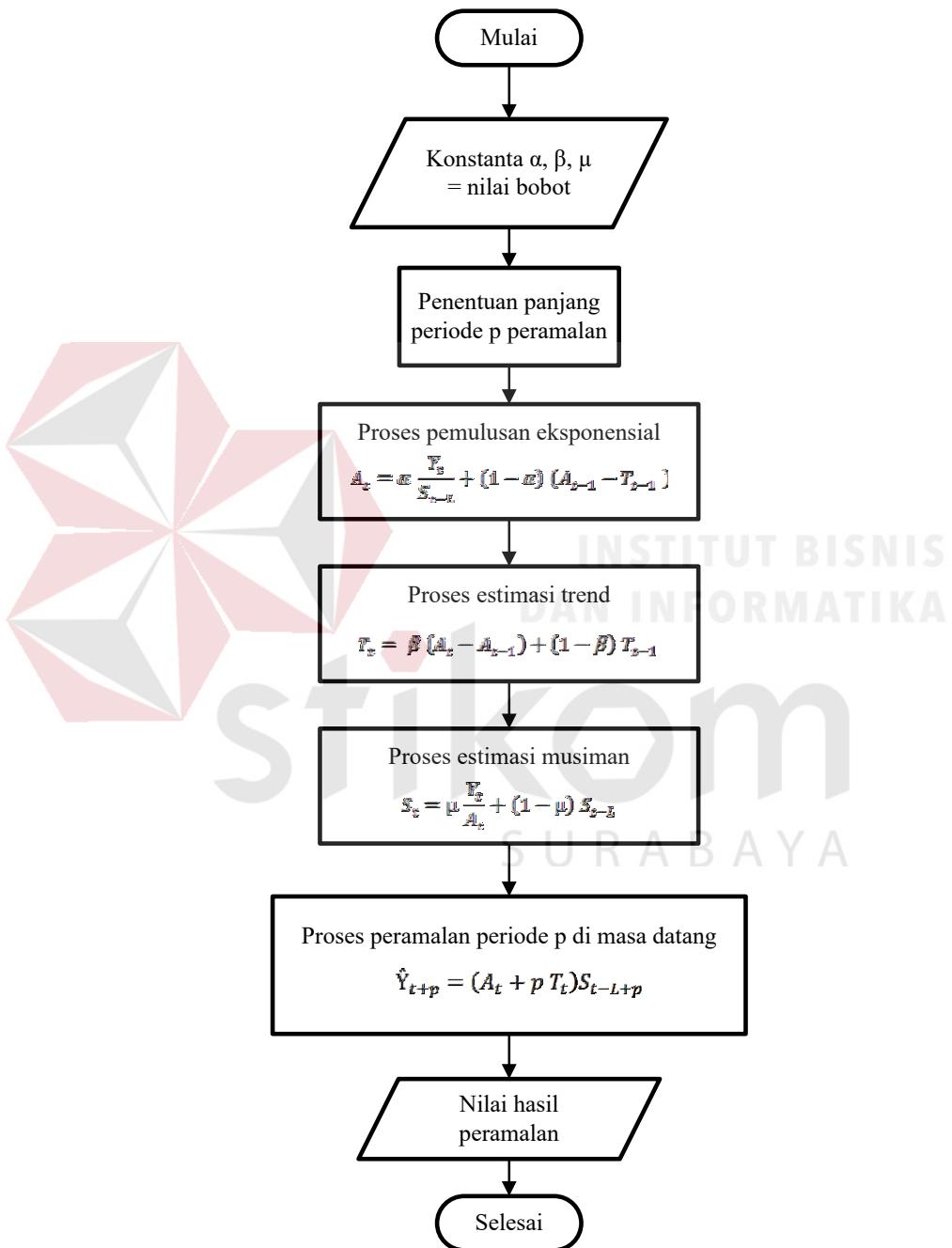
Setelah proses pelatihan berhenti, hasil bobot akan disimpan untuk dijadikan sebagai konstanta pada proses peramalan dengan metode exponential smoothing.



Gambar 3.4 Diagram Alir Proses Pelatihan dengan Backpropagation

Setelah diperoleh nilai bobot dari proses pelatihan *backpropagation*, maka nilai ini dipakai sebagai konstanta pada proses peramalan *exponential smoothing*. Terdapat proses memasukkan periode p di masa datang yang dijadikan

sebagai acuan untuk menghasilkan nilai peramalan penyakit musiman pada periode selanjutnya. Gambar 3.5 menjelaskan proses peramalan ini dalam bentuk diagram alir.



Gambar 3.5 Diagram Alir Proses Peramalan

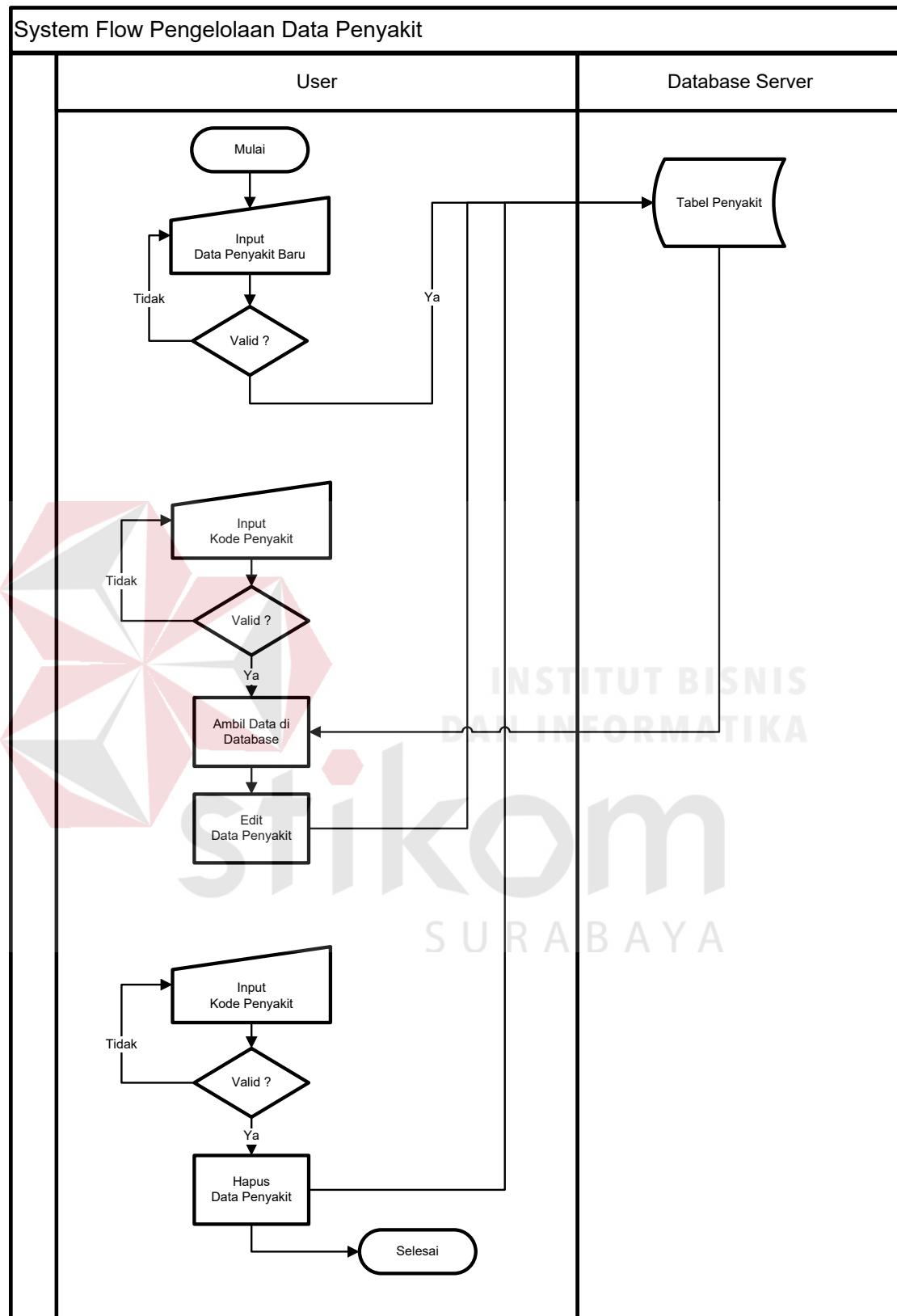
3.4 Alur Sistem (*System Flow*)

Alur sistem pada sistem peramalan kewaspadaan terhadap penyakit musiman adalah sebagai berikut:

1. System flow pengelolaan data penyakit
2. System flow pengelolaan data obat
3. System flow pengelolaan data login
4. System flow pengelolaan data rekap penyakit bulanan
5. System flow login
6. System flow pencarian panjang ramalan terbaik
7. System flow pencarian pola NN terbaik
8. System flow peramalan *Exponential Smoothing* oleh Winter
9. System flow perbandingan hasil ramalan

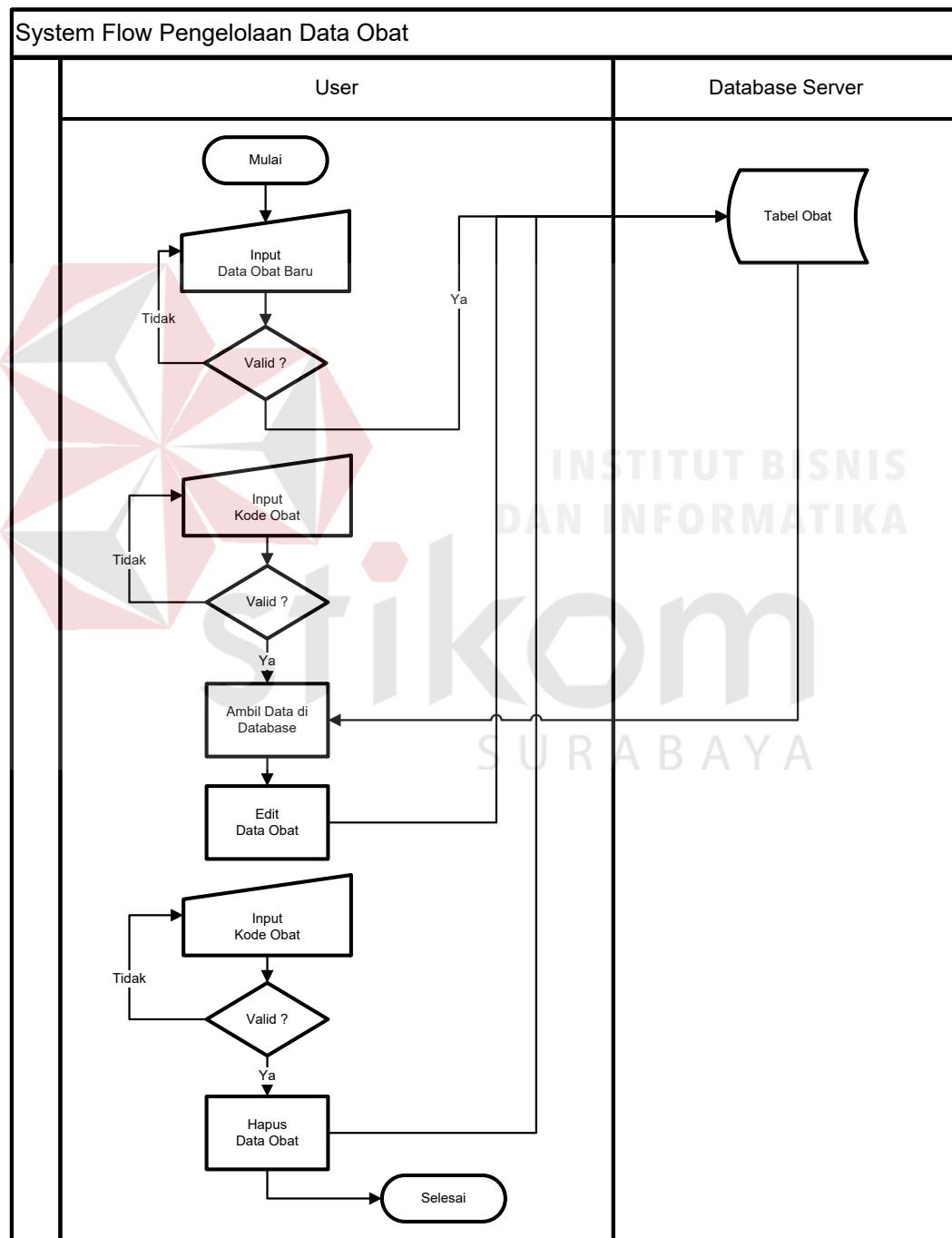
Masing-masing system flow akan digambarkan pada sub bab ini. Setiap system flow menggambarkan aliran sistem dari aplikasi, mulai dari pengelolaan data sampai proses utama yang berupa proses neural network dan peramalan exponential smoothing. Secara keseluruhan masing-masing system flow akan membentuk sistem global dari aplikasi peramalan yang dibuat.

Pada gambar 3.6 akan digambarkan proses pengelolaan data penyakit yaitu tambah data baru, ubah data, dan hapus data. Proses ini hanya bisa dilakukan oleh user yang memiliki status sebagai admin. Setiap proses pengelolaan data terdapat proses pengecekan data, apabila tidak valid, maka akan diulang prosesnya dan tidak disimpan pada database. Proses ini berpengaruh langsung terhadap tabel Penyakit.



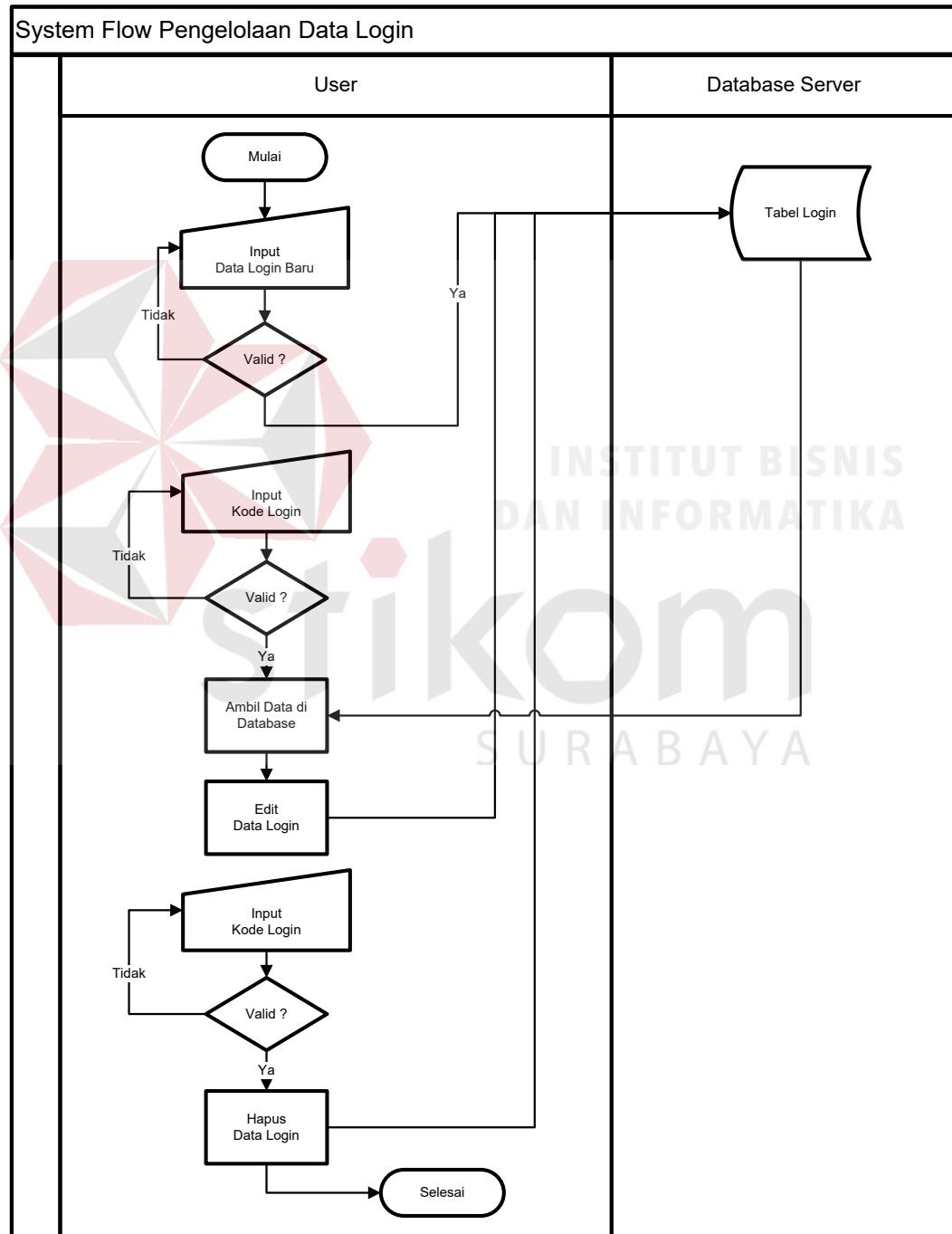
Gambar 3.6 System Flow Pengelolaan Data Penyakit

Proses pengelolaan data obat terlihat pada gambar 3.7 yaitu tambah data baru, ubah data, dan hapus data. Pengelolaan data ini hanya bisa dilakukan oleh user yang memiliki user dengan status admin. Proses ini berpengaruh langsung terhadap tabel Obat.



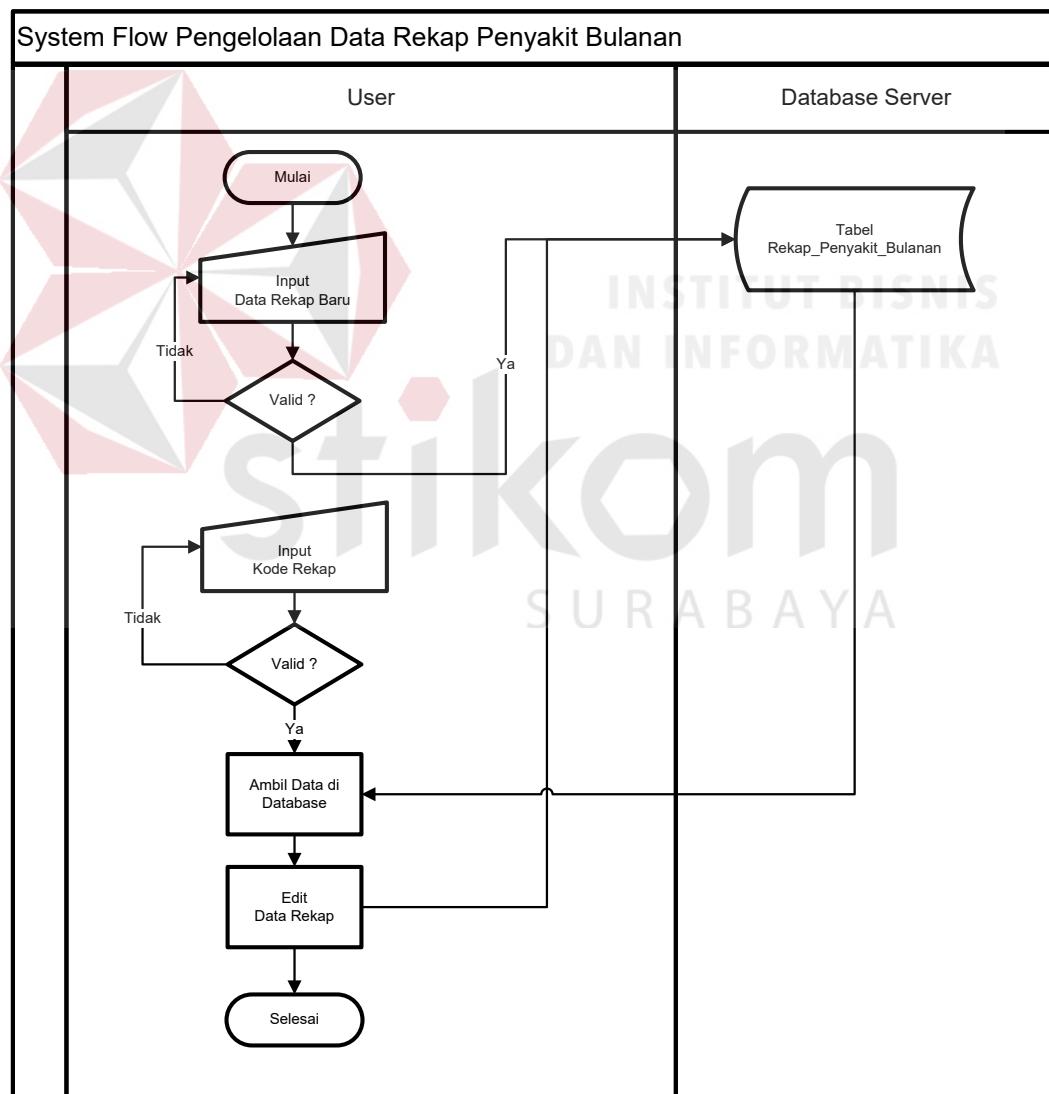
Gambar 3.7 System Flow Pengelolaan Data Obat

Pada gambar 3.8 terlihat proses pengelolaan data login yaitu tambah data baru, ubah data, dan hapus data. Pengelolaan data ini hanya bisa dilakukan oleh user yang memiliki status admin. Proses ini berpengaruh langsung terhadap tabel Login.



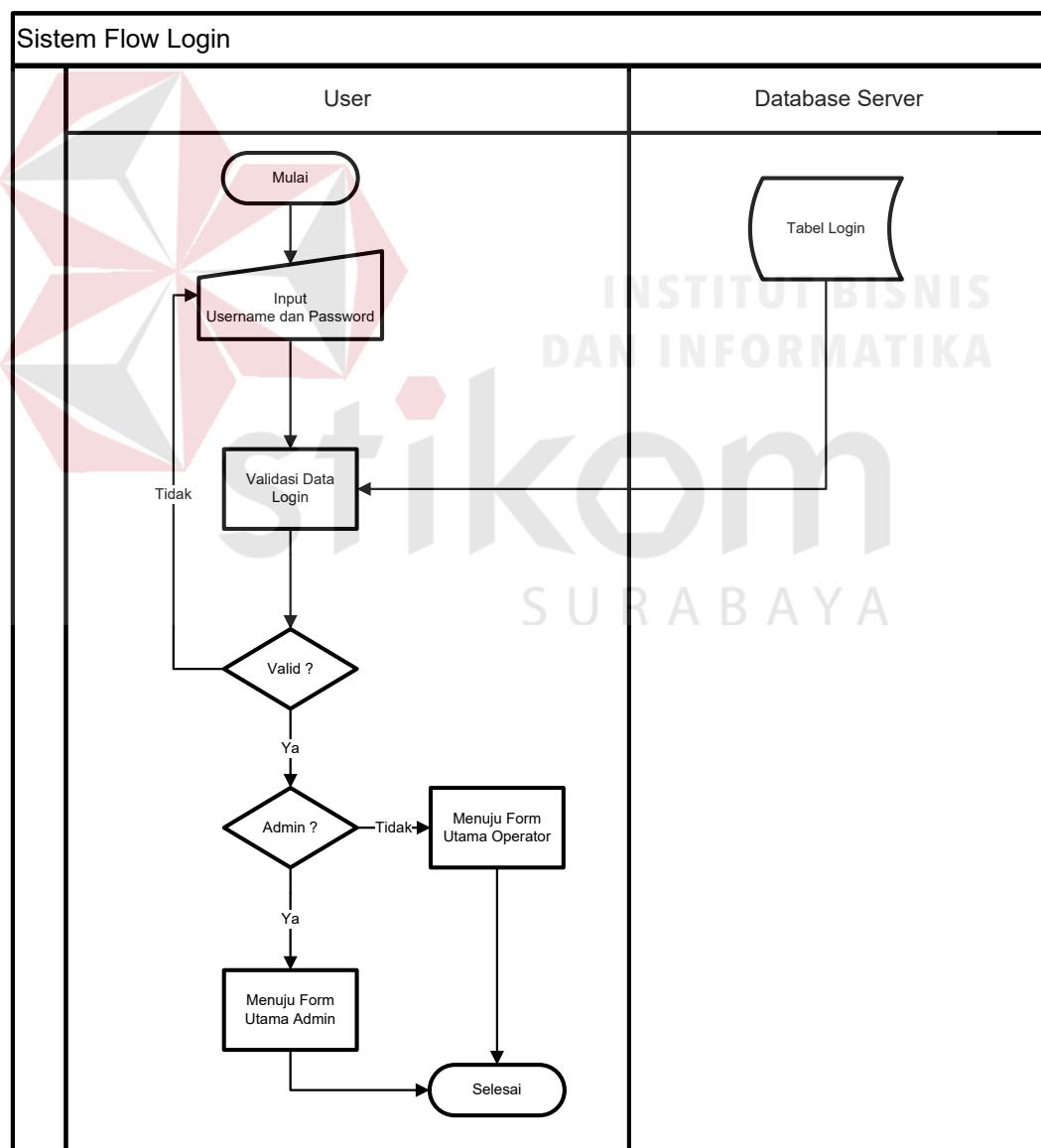
Gambar 3.8 System Flow Pengelolaan Data Login

Proses pengelolaan data rekap penyakit bulanan terlihat pada gambar 3.9 yaitu tambah data baru, ubah data, dan hapus data. Tidak hanya user admin yang bisa melakukan pengelolaan data rekap penyakit bulanan, user yang memiliki status sebagai operator juga bisa melakukan proses ini. Terdapat proses pengecekan data setiap kali melakukan pengelolaan data, apabila data telah valid maka akan disimpan pada database. Proses ini berpengaruh langsung terhadap tabel Rekap_Penyakit_Bulanan.



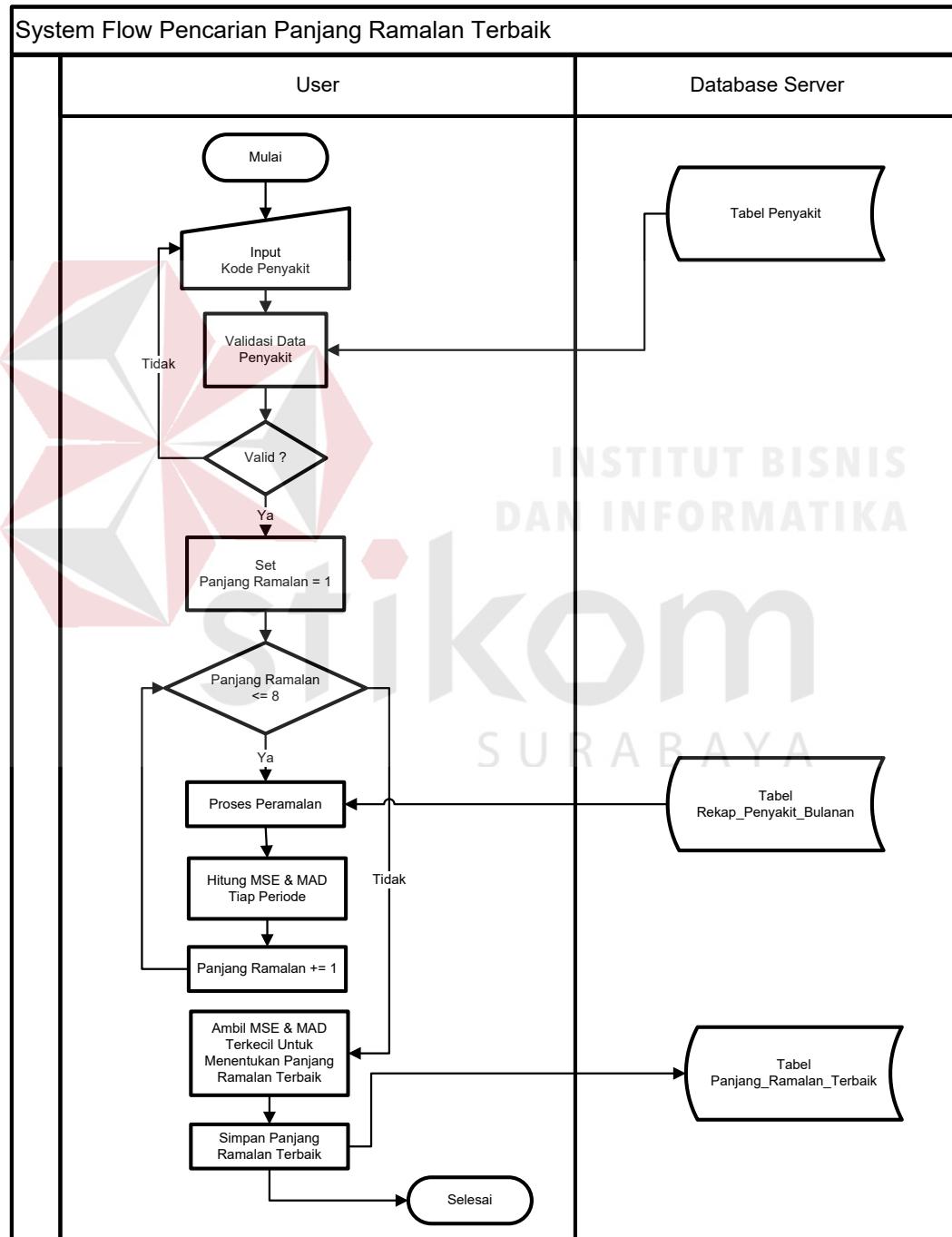
Gambar 3.9 System Flow Pengelolaan Data Rekap Penyakit Bulanan

Pada gambar 3.10 dijelaskan proses login yang dimulai dari user menginputkan data login berupa username dan password. Setelah divalidasi di tabel Login, kemudian dicek status dari user tadi, apakah termasuk admin atau operator biasa. Status ini akan mempengaruhi ke form mana user tadi akan diarahkan. Untuk user admin, akan diarahkan ke form utama admin dan untuk user operator akan diarahkan ke form utama operator. Selain itu, terdapat juga perbedaan hak akses yang bisa dilakukan oleh user dengan status berbeda.



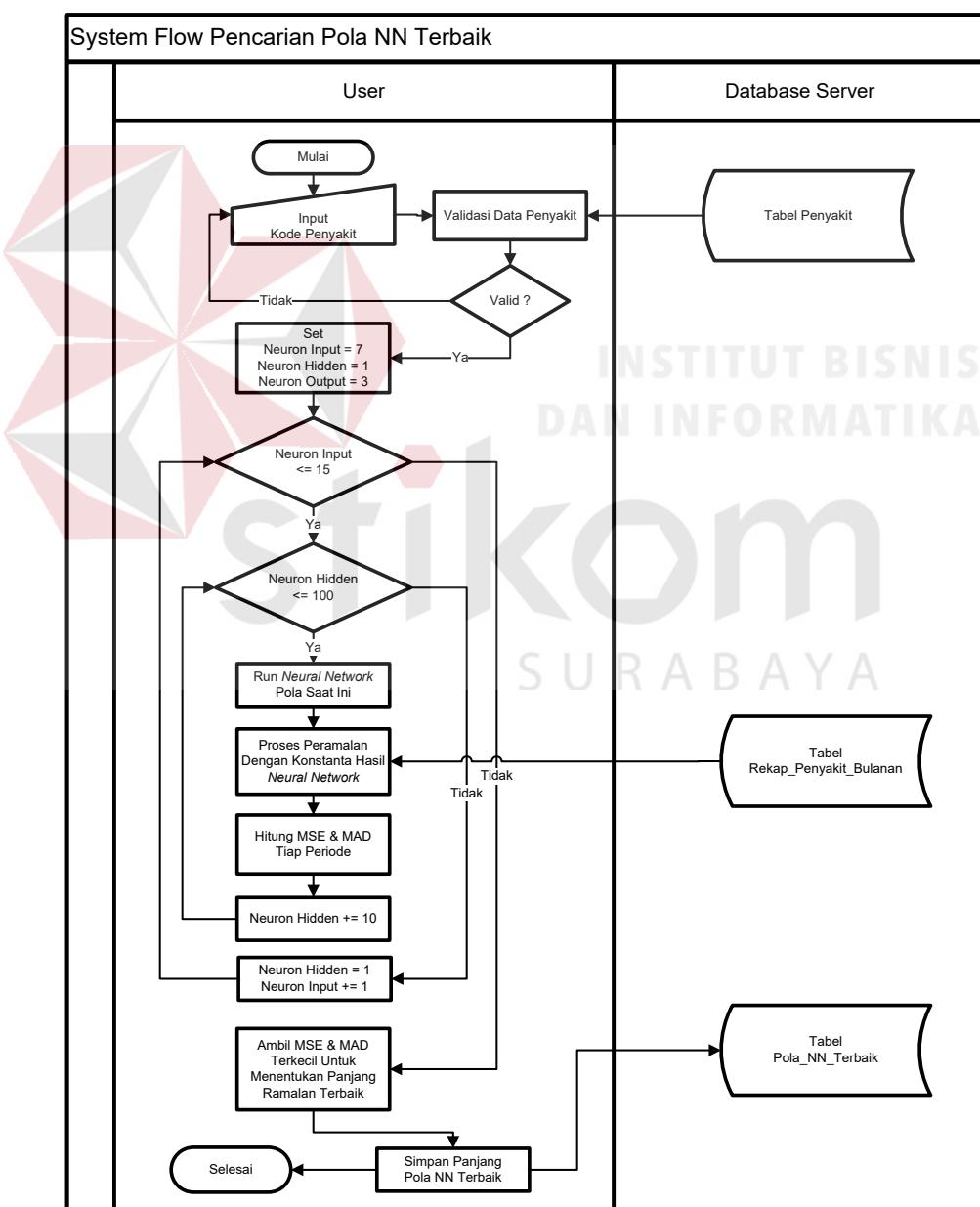
Gambar 3.10 System Flow Proses Login

Pada gambar 3.11 menjelaskan proses pencarian panjang ramalan terbaik untuk tiap penyakit. Setelah dilakukan proses *looping* sampai 8 kali untuk melakukan proses peramalan dan hitung *error*, maka didapatkan panjang ramalan terbaik untuk penyakit itu yang ditandai dengan MSE dan MAD yang terkecil.



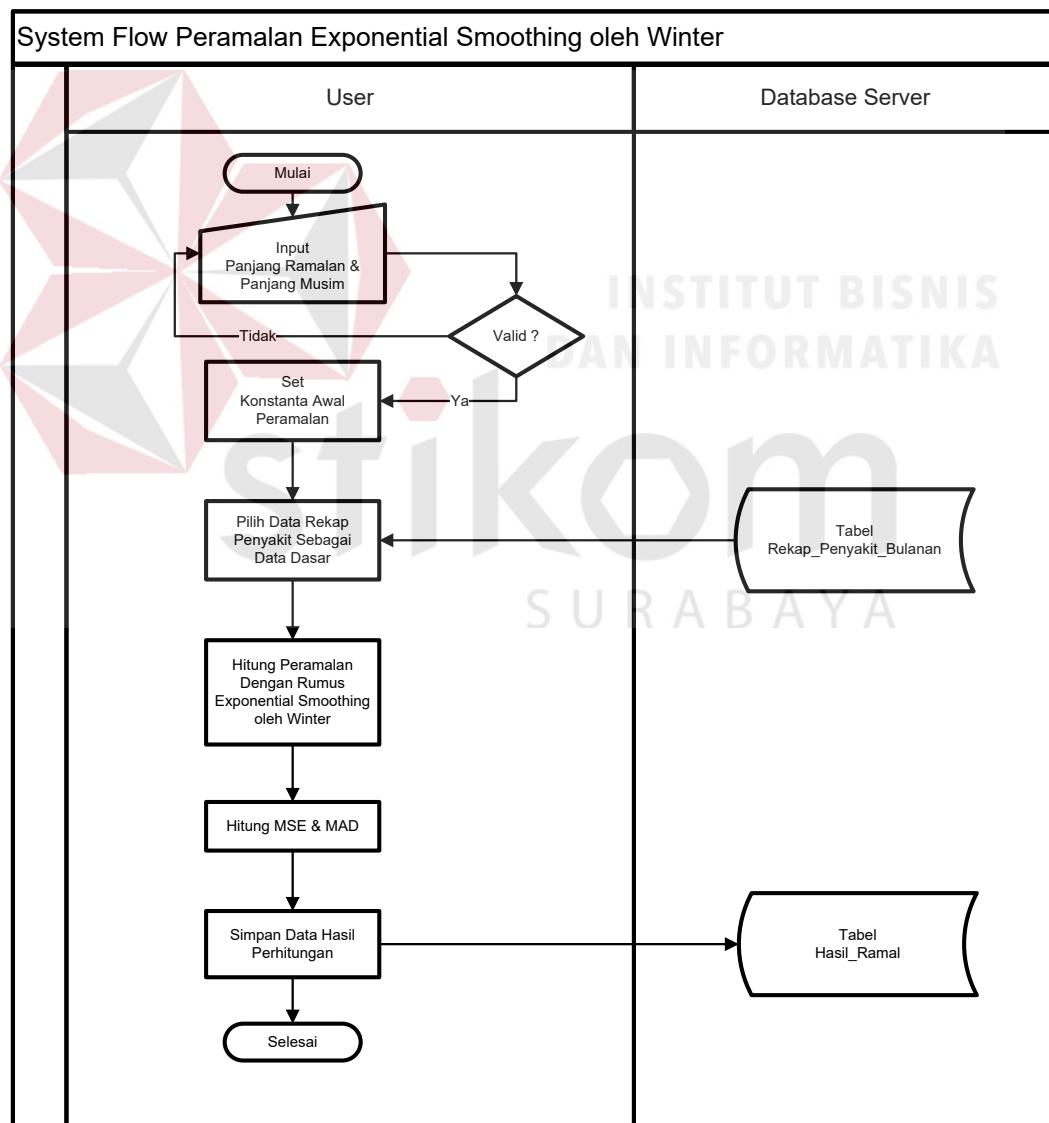
Gambar 3.11 System Flow Pencarian Panjang Ramalan Terbaik

Pada gambar 3.12 menjelaskan proses pencarian pola *neural network* terbaik untuk tiap penyakit. *Looping* dilakukan maksimal 15 *neuron input*, tiap *neuron input* ada *neuron hidden* yang juga looping untuk kelipatan 10. Setiap pola terdapat proses peramalan dan hitung *error*. Setelah semua looping, maka didapatkan pola NN terbaik untuk penyakit itu yang ditandai dengan MSE dan MAD yang terkecil.



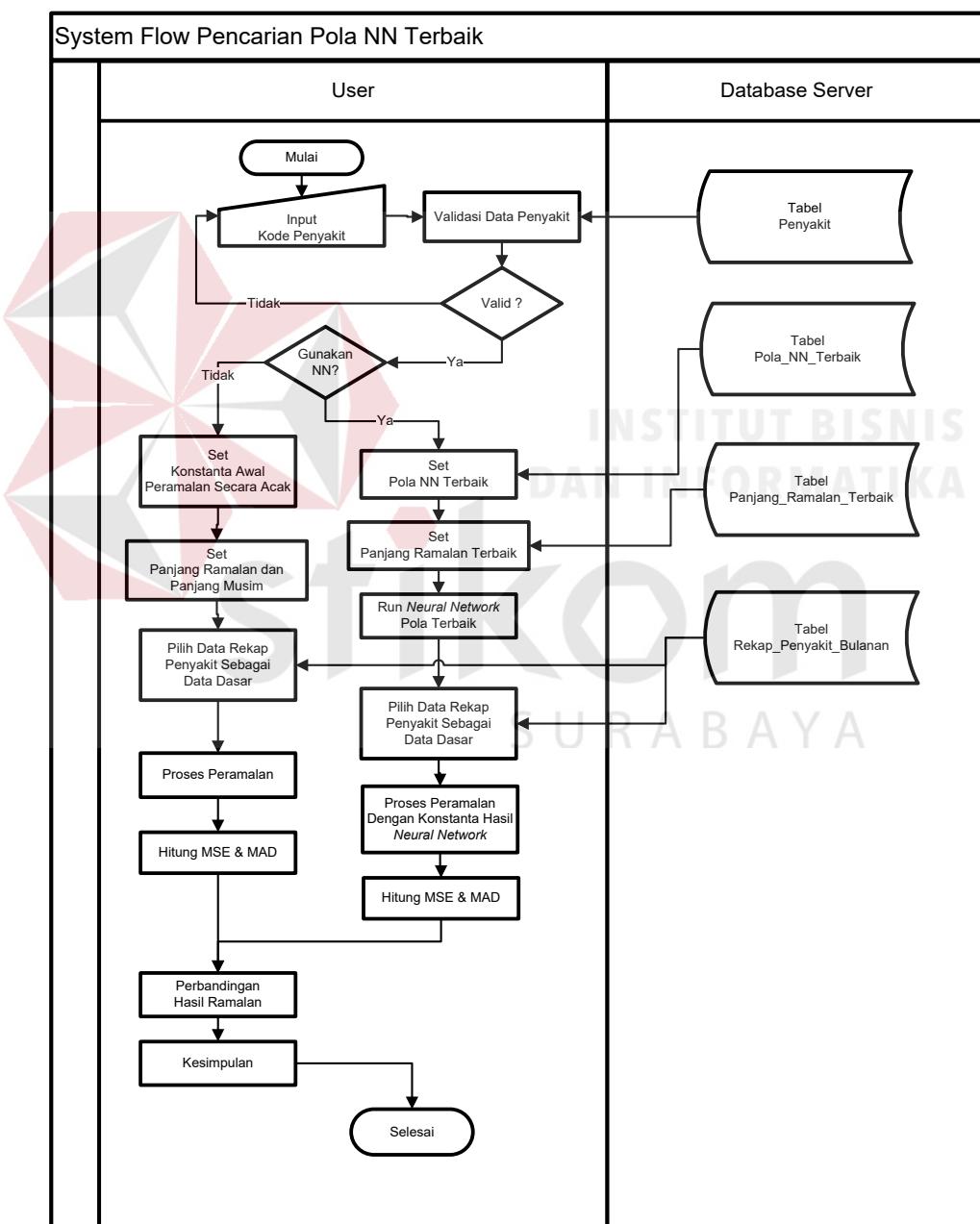
Gambar 3.12 System Flow Pencarian Pola NN Terbaik

Pada gambar 3.13 menjelaskan proses peramalan dengan *Exponential Smoothing* oleh Winter. Beberapa hal yang harus ada sebelum proses peramalan dilakukan adalah, data rekap penyakit sebagai data dasar, tiga kontanta peramalan, panjang periode, dan panjang musim. Setelah dilakukan proses peramalan, bisa didapatkan error dengan membandingkan data aktual dengan nilai hasil peramalan. Untuk *flow* rumus peramalan bisa dilihat pada gambar 3.5 diagram alir proses peramalan.



Gambar 3.13 System Flow Peramalan Exponential Smoothing oleh Winter

Pada gambar 3.14 menjelaskan proses perbandingan hasil peramalan antara peramalan konvensional dengan peramalan dengan bantuan *Neural Network* untuk menentukan konstanta awal. Dari situ hasil error bisa dibandingkan untuk tiap hasil peramalan yang bisa menghasilkan kesimpulan peramalan mana yang terbaik.



Gambar 3.14 System Flow Perbandingan Hasil Ramalan

3.5 Entity Relationship Diagram (ERD)

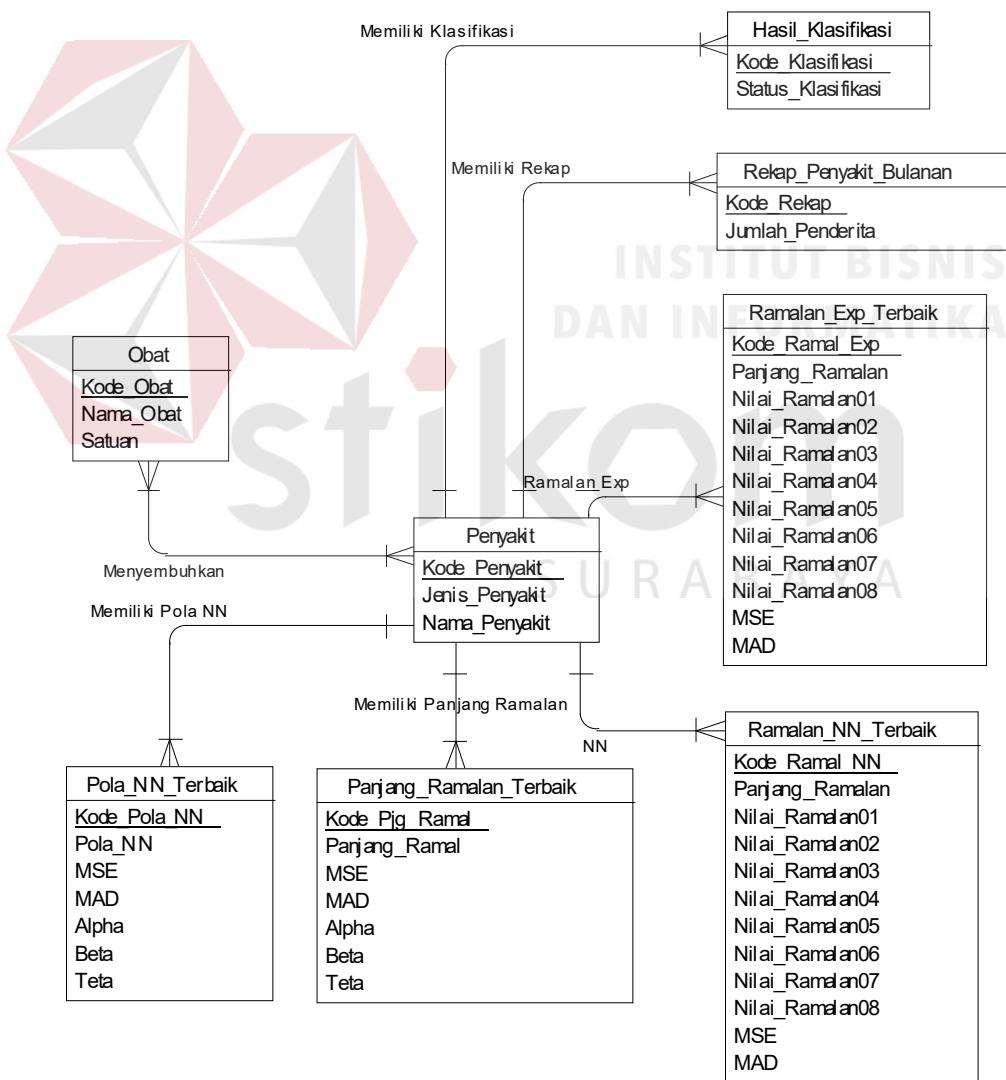
ERD merupakan suatu desain sistem yang digunakan untuk merepresentasikan, menentukan dan mendokumentasikan kebutuhan-kebutuhan untuk sistem pemrosesan *database*. ERD juga menyediakan bentuk untuk menunjukkan struktur keseluruhan dari data user. Dalam ERD data-data tersebut digambarkan dengan menggambarkan simbol *entity*. Dalam perancangan sistem ini terdapat beberapa *entity* yang saling terkait untuk menyediakan data-data yang dibutuhkan oleh sistem yaitu :

1. Entity Login untuk menyimpan data login ke aplikasi.
2. Entity Penyakit untuk menyimpan data penyakit.
3. Entity Obat untuk menyimpan data obat.
4. Entity Penyakit_Obat untuk menyimpan data obat dari masing-masing penyakit.
5. Entity Hasil_Klasifikasi untuk menyimpan penyakit hasil klasifikasi yang mempunyai kecenderungan musiman.
6. Entity Rekap_Penyakit_Bulanan untuk menyimpan data jumlah penyakit bulanan.
7. Entity Data_Latih untuk menyimpan data latih yang akan digunakan untuk pelatihan *neural network*.
8. Entity Panjang_Ramalan_Terbaik untuk menyimpan data panjang ramalan terbaik untuk tiap penyakit.
9. Entity Pola_NN_Terbaik untuk menyimpan data pola *neural network* terbaik untuk tiap penyakit.

10. Entity Ramalan_Exp_Terbaik untuk menyimpan data peramalan *exponential smoothing* oleh Winter yang terbaik dalam satu kali proses peramalan.
11. Entity Ramalan_NN_Terbaik untuk menyimpan data peramalan dengan *neural network* yang terbaik dalam satu kali proses peramalan.

3.5.1 Conceptual Data Model (CDM)

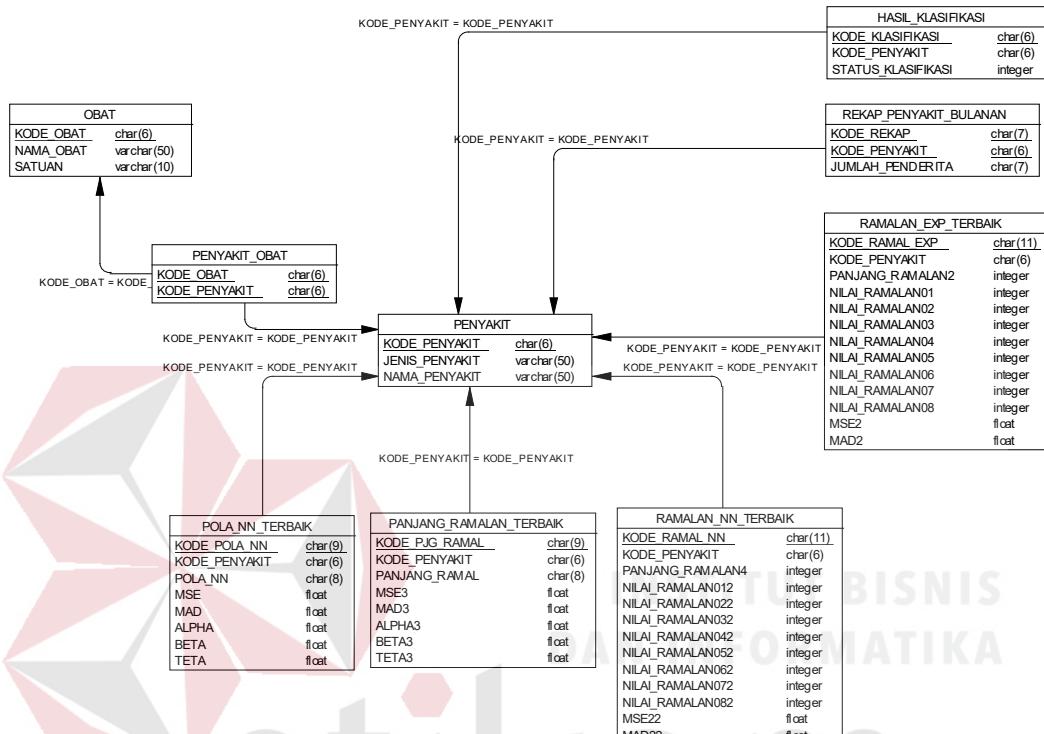
Model konseptual dari database yang telah dirancang dan digunakan pada aplikasi peramalan ini adalah sebagai berikut:



Gambar 3.15 Conceptual Data Model

3.5.2 Physical Data Model (PDM)

Model fisikal dari database yang telah dirancang dan digunakan dalam aplikasi peramalan ini adalah sebagai berikut:



Gambar 3.16 Physical Data Model

3.6 Struktur Database

Berikut ini adalah rancangan seluruh tabel yang digunakan dalam aplikasi peramalan:

1. Nama : Penyakit

Fungsi : untuk menyimpan data penyakit

Tabel 3.2 Tabel Penyakit

Nama	Tipe Data	Panjang	Keterangan
Kode_Penyakit	Char	6	Primary Key
Jenis_Penyakit	VarChar	50	
Nama_Penyakit	VarChar	50	

2. Nama : Login

Fungsi : untuk menyimpan data login

Tabel 3.1 Tabel Login

Nama	Tipe Data	Panjang	Keterangan
ID	Char	7	Primary Key
Username	VarChar	25	
Password	VarChar	25	
Status	VarChar	10	

3. Nama : Obat

Fungsi : untuk menyimpan data obat

Tabel 3.3 Tabel Obat

Nama	Tipe Data	Panjang	Keterangan
Kode_Obat	Char	6	Primary Key
Nama_Obat	VarChar	50	
Satuan	VarChar	10	

4. Nama : Penyakit_Obat

Fungsi : untuk menyimpan data obat dari masing-masing penyakit, dari sini bisa diketahui jenis-jenis obat untuk tiap penyakit dan juga bisa diketahui tiap obat bisa menyembuhkan penyakit apa saja

Tabel 3.4 Tabel Penyakit_Obat

Nama	Tipe Data	Panjang	Keterangan
Kode_Penyakit	Char	6	Primary Key Foreign Key - Penyakit(Kode_Penyakit)
Kode_Obat	Char	6	Primary Key Foreign Key - Obat(Kode_Obat)

5. Nama : Hasil_Klasifikasi

Fungsi : menyimpan penyakit yang memiliki kecenderungan musiman

Tabel 3.5 Tabel Hasil_Klasifikasi

Nama	Tipe Data	Panjang	Keterangan
Kode_Klasifikasi	Char	6	Primary Key
Kode_Penyakit	Char	6	Foreign Key - Penyakit(Kode_Penyakit)
Status_Klasifikasi	Integer	1	

6. Nama : Rekap_Penyakit_Bulanan

Fungsi : menyimpan data jumlah penyakit bulanan

Tabel 3.6 Tabel Rekap_Penyakit_Bulanan

Nama	Tipe Data	Panjang	Keterangan
Kode_Rekap	Char	7	Primary Key
Kode_Penyakit	Char	6	Primary Key
Jumlah_Penderita	Integer	4	Foreign Key - Penyakit(Kode_Penyakit)

7. Nama : Panjang_Ramalan_Terbaik

Fungsi : menyimpan data panjang ramalan terbaik untuk tiap penyakit

Tabel 3.7 Tabel Panjang_Ramalan_Terbaik

Nama	Tipe Data	Panjang	Keterangan
Kode_Pjg_Ramal	Char	9	Primary Key
Kode_Penyakit	Char	6	Foreign Key - Penyakit(Kode_Penyakit)
Panjang_Ramal	Integer	2	
MSE	Float	4	
MAD	Float	4	
Alpha	Float	4	
Beta	Float	4	
Teta	Float	4	

8. Nama : Pola_NN_Terbaik

Fungsi : menyimpan data pola *neural network* terbaik untuk tiap penyakit

Tabel 3.8 Tabel Pola_NN_Terbaik

Nama	Tipe Data	Panjang	Keterangan
Kode_Pola_NN	Char	9	Primary Key
Kode_Penyakit	Char	6	Foreign Key - Penyakit(Kode_Penyakit)
Pola_NN	Integer	2	
MSE	Float	4	
MAD	Float	4	
Alpha	Float	4	
Beta	Float	4	
Teta	Float	4	

9. Nama : Ramalan_Exp_Terbaik

Fungsi : menyimpan data peramalan *exponential smoothing* oleh Winter yang terbaik dalam satu kali proses peramalan

Tabel 3.9 Tabel Ramalan_Exp_Terbaik

Nama	Tipe Data	Panjang	Keterangan
Kode_Ramal_Exp	Char	11	Primary Key
Kode_Penyakit	Char	6	Foreign Key - Penyakit(Kode_Penyakit)
Panjang_Ramalan	Integer	2	
Nilai_Ramalan01	Integer	5	
Nilai_Ramalan02	Integer	5	
Nilai_Ramalan03	Integer	5	
Nilai_Ramalan04	Integer	5	
Nilai_Ramalan05	Integer	5	
Nilai_Ramalan06	Integer	5	
Nilai_Ramalan07	Integer	5	
Nilai_Ramalan08	Integer	5	
MSE	Float	4	
MAD	Float	4	

10. Nama : Ramalan_NN_Terbaik

Fungsi : menyimpan data peramalan *NN* terbaik dalam proses peramalan

Tabel 3.10 Tabel Ramalan_NN_Terbaik

Nama	Tipe Data	Panjang	Keterangan
Kode_Ramal_NN	Char	11	Primary Key
Kode_Penyakit	Char	6	Foreign Key - Penyakit(Kode_Penyakit)
Panjang_Ramalan	Integer	2	
Nilai_Ramalan01	Integer	5	
Nilai_Ramalan02	Integer	5	
Nilai_Ramalan03	Integer	5	
Nilai_Ramalan04	Integer	5	
Nilai_Ramalan05	Integer	5	
Nilai_Ramalan06	Integer	5	
Nilai_Ramalan07	Integer	5	
Nilai_Ramalan08	Integer	5	
MSE	Float	4	
MAD	Float	4	

11. Nama : Data_Latih

Fungsi : menyimpan data latih yang akan digunakan untuk pelatihan *NN*

Tabel 3.11 Tabel Data_Latih

Nama	Tipe Data	Panjang	Keterangan
Kode_Latih	Char	9	Primary Key
Jumlah_Data	Integer	2	
Data_Aktual01	Integer	5	
Data_Aktual02	Integer	5	
Data_Aktual03	Integer	5	
Data_Aktual04	Integer	5	
Data_Aktual05	Integer	5	
Data_Aktual06	Integer	5	
Data_Aktual07	Integer	5	
Data_Aktual08	Integer	5	
Data_Aktual09	Integer	5	
Data_Aktual10	Integer	5	
Data_Aktual11	Integer	5	
Data_Aktual12	Integer	5	

Tabel 3.11 Tabel Data_Latih (lanjutan)

Nama	Tipe Data	Panjang	Keterangan
Panjang_Ramalan	Integer	2	
Panjang_Musim	Integer	2	
Output_Alpha	Float	4	
Output_Beta	Float	4	
Output_Teta	Float	4	

3.7 Desain Input Output

Bagian ini menjelaskan tentang desain input output dari aplikasi peramalan, penggambaran desain tiap form adalah sebagai berikut:

1. Form Login

Terdapat *textbox* username dan password untuk login. Data login menentukan apakah user sebagai admin atau sebagai operator biasa saja.

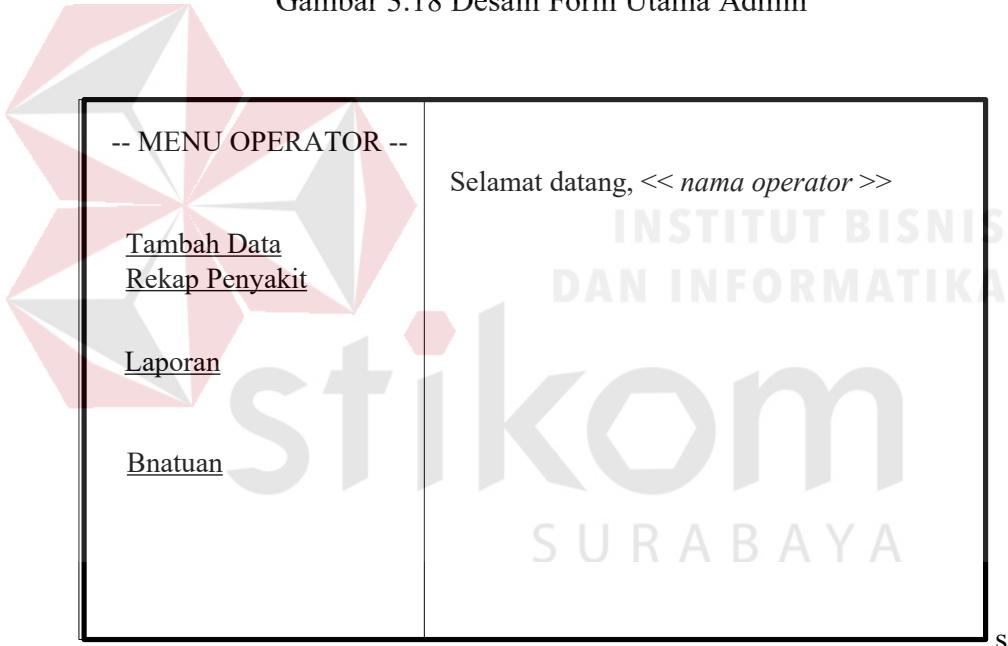
Gambar 3.17 Desain Form Login

2. Form Utama

Form yang akan muncul setelah user berhasil login. Ada perbedaan tampilan menu antara user admin dan user operator biasa. Perbedaan ini berpengaruh terhadap hak akses untuk tiap status user, hal ini ditunjukkan pada menu apa saja yang bisa diakses.

<p>-- MENU ADMIN --</p> <p><u>Pengelolaan Data</u></p> <p><u>Proses Utama</u></p> <p><u>Laporan</u></p> <p><u>Bantuan</u></p>	<p>Selamat datang, << <i>nama admin</i> >></p>
---	--

Gambar 3.18 Desain Form Utama Admin



<p>-- MENU OPERATOR --</p> <p><u>Tambah Data</u></p> <p><u>Rekap Penyakit</u></p> <p><u>Laporan</u></p> <p><u>Bantuan</u></p>	<p>Selamat datang, << <i>nama operator</i> >></p> <p>INSTITUT BISNIS DAN INFORMATIKA stikom SURABAYA</p>
---	---

Gambar 3.19 Desain Form Utama Operator

3. Form Pengelolaan Data Penyakit

Form untuk melakukan pengelolaan data penyakit seperti tambah data baru, mengubah data yang ada, dan menghapus data. Form ini berhubungan langsung dengan tabel Penyakit.

Kode Penyakit	Char(6)																								
Jenis Penyakit	Varchar(50)																								
Nama Penyakit	Varchar(50)																								
<input type="button" value="Simpan"/> <input type="button" value="Edit"/> <input type="button" value="Hapus"/>																									
<table border="1"> <thead> <tr> <th>Kode Penyakit</th> <th>Jenis Penyakit</th> <th>Nama Penyakit</th> </tr> <tr> <th>Char(6)</th> <th>Varchar(50)</th> <th>Varchar(50)</th> </tr> </thead> <tbody> <tr><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td></tr> </tbody> </table>		Kode Penyakit	Jenis Penyakit	Nama Penyakit	Char(6)	Varchar(50)	Varchar(50)																		
Kode Penyakit	Jenis Penyakit	Nama Penyakit																							
Char(6)	Varchar(50)	Varchar(50)																							

Gambar 3.20 Desain Form Pengelolaan Data Penyakit

4. Halaman Pengelolaan Data Rekap Penyakit

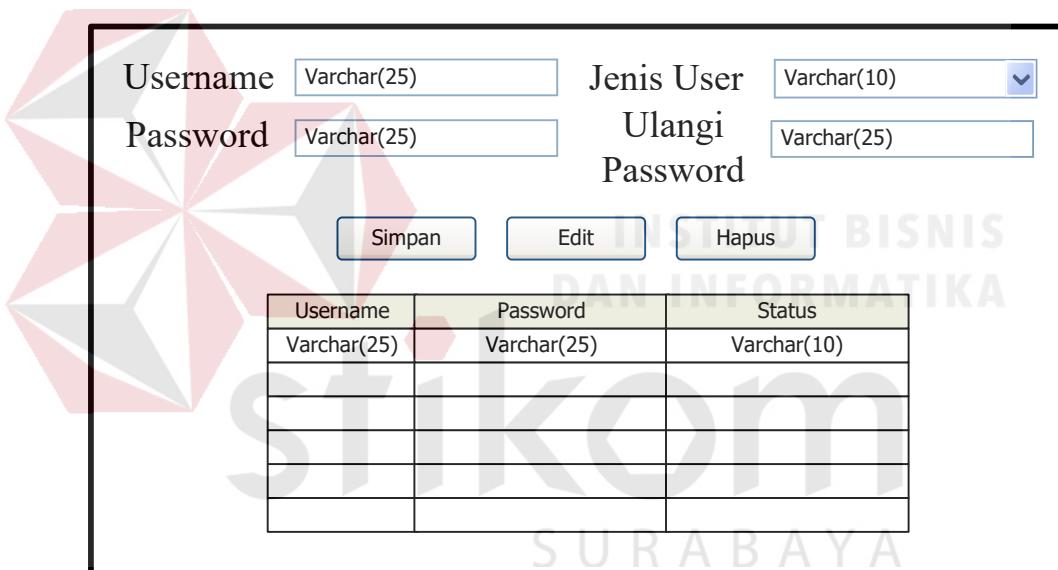
Form untuk melakukan pengelolaan data rekap penyakit bulanan seperti tambah data baru, mengubah data yang ada, dan menghapus data. User dengan status admin atau operator bisa melakukan proses ini. Form ini berhubungan langsung dengan tabel Rekap_Penyakit_Bulanan.

Kode Rekap	Char(7)	<input type="button" value="Data Baru"/> <input type="button" value="Edit Data"/>														
<table border="1"> <thead> <tr> <th>Kode Penyakit</th> <th>Jumlah Penderita</th> </tr> <tr> <th>Char(6)</th> <th>Integer</th> </tr> </thead> <tbody> <tr><td></td><td></td></tr> <tr><td></td><td></td></tr> <tr><td></td><td></td></tr> <tr><td></td><td></td></tr> <tr><td></td><td></td></tr> </tbody> </table>			Kode Penyakit	Jumlah Penderita	Char(6)	Integer										
Kode Penyakit	Jumlah Penderita															
Char(6)	Integer															
<input type="button" value="Simpan"/>																

Gambar 3.21 Desain Form Pengelolaan Rekap Penyakit Bulanan

5. Halaman Pengelolaan Data Login

Form untuk melakukan pengelolaan data login user seperti tambah data baru, mengubah data yang ada, dan menghapus data. TextBox Username untuk memasukkan username, ComboBox Jenis User untuk menentukan status user, dan TextBox Password untuk memasukkan password. Terdapat pengecekan apakah password telah benar dengan memasukkan password yang sama pada TextBox Ulangi Password. Form ini berhubungan langsung dengan tabel Login.



The image shows a user interface for managing login data. It features a large red and grey geometric watermark in the background with the text 'STIKOM SURABAYA' and 'DAN INFORMATIKA'. The main form is enclosed in a black border. It contains the following elements:

- Text input fields for "Username" (Varchar(25)) and "Password" (Varchar(25)).
- A "Jenis User" dropdown menu (Varchar(10)).
- A "Ulangi Password" input field (Varchar(25)).
- Buttons for "Simpan", "Edit", and "Hapus".
- A table with columns "Username" (Varchar(25)), "Password" (Varchar(25)), and "Status" (Varchar(10)). The table has 7 rows, with the first row being the header.

Gambar 3.22 Desain Form Pengelolaan Data Login

6. Form Pengelolaan Data Obat

Form untuk melakukan pengelolaan data obat seperti tambah data baru, mengubah data yang ada, dan menghapus data. Hanya user dengan status admin yang bisa mengakses form pengelolaan obat ini. Form ini berhubungan langsung dengan tabel Obat.

Kode Obat	Char(6)																								
Nama Obat	Varchar(50)																								
Satuan	Varchar(20)																								
<input type="button" value="Simpan"/> <input type="button" value="Edit"/> <input type="button" value="Hapus"/>																									
<table border="1"> <thead> <tr> <th>Kode Obat</th> <th>Nama Obat</th> <th>Satuan</th> </tr> <tr> <th>Char(6)</th> <th>Varchar(50)</th> <th>Varchar(20)</th> </tr> </thead> <tbody> <tr><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td></tr> </tbody> </table>		Kode Obat	Nama Obat	Satuan	Char(6)	Varchar(50)	Varchar(20)																		
Kode Obat	Nama Obat	Satuan																							
Char(6)	Varchar(50)	Varchar(20)																							

Gambar 3.23 Desain Form Pengelolaan Data Obat

7. Form Klasifikasi

Form untuk melakukan klasifikasi data penyakit yang memiliki kecenderungan pola musiman dalam periode waktu tertentu. Di sini bisa didapatkan beberapa jenis penyakit pada periode tertentu yang memiliki kecenderungan musiman. Setelah ditekan tombol Proses, data penyakit musiman akan keluar pada DataGrid di bawahnya.

Periode	<input checked="" type="radio"/> Default <input checked="" type="radio"/> Manual <input type="text" value="Date"/> <input type="button" value="Proses"/>																								
<table border="1"> <thead> <tr> <th>Kode Penyakit</th> <th>Nama Penyakit</th> <th>Total Pasien</th> </tr> <tr> <th>Char(6)</th> <th>Varchar(50)</th> <th>Integer</th> </tr> </thead> <tbody> <tr><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td></tr> </tbody> </table>		Kode Penyakit	Nama Penyakit	Total Pasien	Char(6)	Varchar(50)	Integer																		
Kode Penyakit	Nama Penyakit	Total Pasien																							
Char(6)	Varchar(50)	Integer																							

Gambar 3.24 Desain Form Klasifikasi

8. Form Peramalan *Exponential Smoothing*

Form untuk melakukan peramalan exponential smoothing dari data penyakit hasil klasifikasi. Ketika tombol Simpan ditekan, maka data akan tersimpan di tabel Data_Latih yang bisa dimanfaatkan untuk proses *neural network*.

Panjang Ramalan	Integer	Proses																																																																
Panjang Musim	Integer																																																																	
<table border="1"> <thead> <tr> <th>Kode Penyakit</th> <th>Nama Penyakit</th> <th>Hasil Ramalan</th> <th>Alpha</th> <th>Beta</th> <th>Teta</th> <th>MSE</th> <th>MAD</th> </tr> <tr> <th>Char(6)</th> <th>Varchar(50)</th> <th>Integer</th> <th>Float(4)</th> <th>Float(4)</th> <th>Float(4)</th> <th>Float(4)</th> <th>Float(4)</th> </tr> </thead> <tbody> <tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr> </tbody> </table>			Kode Penyakit	Nama Penyakit	Hasil Ramalan	Alpha	Beta	Teta	MSE	MAD	Char(6)	Varchar(50)	Integer	Float(4)	Float(4)	Float(4)	Float(4)	Float(4)																																																
Kode Penyakit	Nama Penyakit	Hasil Ramalan	Alpha	Beta	Teta	MSE	MAD																																																											
Char(6)	Varchar(50)	Integer	Float(4)	Float(4)	Float(4)	Float(4)	Float(4)																																																											
Simpan																																																																		

Gambar 3.25 Desain Form Peramalan Exponential Smoothing

9. Form Data Latih

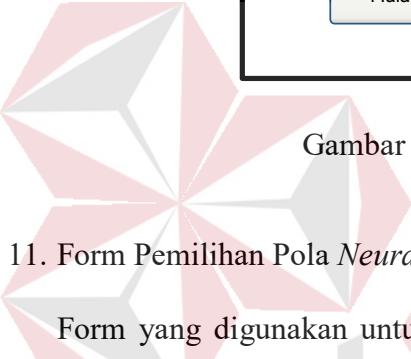
Form untuk melihat data-data latih yang ada dan menentukan berapa panjang neuron input yang akan digunakan dalam proses pelatihan.

Panjang Neuron Input	Integer	Proses																																																																
<table border="1"> <thead> <tr> <th>Kode Latih</th> <th>Jumlah Data Latih</th> <th>Data Aktual</th> <th>Panjang Ramalan</th> <th>Panjang Musim</th> <th>Alpha</th> <th>Beta</th> <th>Teta</th> </tr> <tr> <th>Char(9)</th> <th>Integer</th> <th>Integer</th> <th>Integer</th> <th>Integer</th> <th>Float(4)</th> <th>Float(4)</th> <th>Float(4)</th> </tr> </thead> <tbody> <tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr> </tbody> </table>			Kode Latih	Jumlah Data Latih	Data Aktual	Panjang Ramalan	Panjang Musim	Alpha	Beta	Teta	Char(9)	Integer	Integer	Integer	Integer	Float(4)	Float(4)	Float(4)																																																
Kode Latih	Jumlah Data Latih	Data Aktual	Panjang Ramalan	Panjang Musim	Alpha	Beta	Teta																																																											
Char(9)	Integer	Integer	Integer	Integer	Float(4)	Float(4)	Float(4)																																																											
Simpan																																																																		

Gambar 3.26 Desain Form Data Latih

10. Form Pelatihan

Form untuk melakukan proses pelatihan pada pola jaringan network. Proses bisa dihentikan dengan dua cara yaitu, apabila error sudah sama dengan nol atau secara manual dengan menekan tombol Berhenti apabila error sudah dirasa cukup.



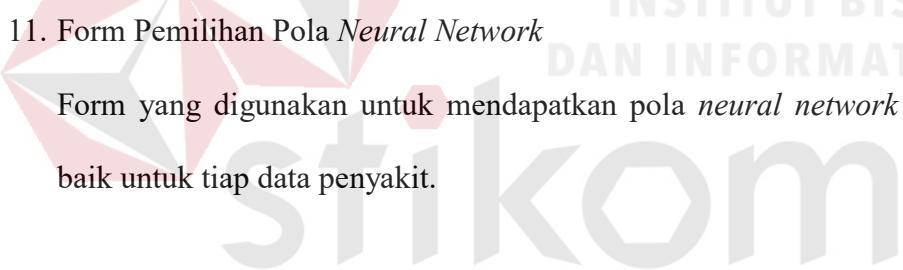
Form Pelatihan (Training Form) dengan tampilan sebagai berikut:

- Iterasi ke-:
- MSE:
- Tambah Satu Iterasi:
- Mulai:
- Simpan:
- Lihat Grafik:

Gambar 3.27 Desain Form Pelatihan

11. Form Pemilihan Pola *Neural Network*

Form yang digunakan untuk mendapatkan pola *neural network* yang paling baik untuk tiap data penyakit.



Form Pemilihan Pola *Neural Network* (Selection Form of NN Pattern) dengan tampilan sebagai berikut:

- Jumlah Neuron Input:
- Kode Penyakit:
- Jumlah Neuron Hidden:
- Nama Penyakit:
- Jumlah Neuron Output:
- Mulai:
- Pola Terbaik:
- Simpan:

Tabel hasil pemilihan pola:

Kode Pola	Pola NN	MSE	MAD	Alpha	Beta	Teta
Char(5)	Varchar(8)	Float(4)	Float(4)	Float(4)	Float(4)	Float(4)

Gambar 3.28 Desain Form Pemilihan Pola NN

12. Form Pemilihan Panjang Ramalan

Form yang digunakan untuk mendapatkan panjang ramalan yang paling baik untuk tiap data penyakit.

Kode	Panjang Ramalan	Data Aktual	Nilai Ramalan	MSE	MAD
Char(5)	Integer	Integer	Integer	Float(4)	Float(4)

Gambar 3.29 Desain Form Pemilihan Panjang Ramalan

13. Form Perbandingan Hasil Ramalan

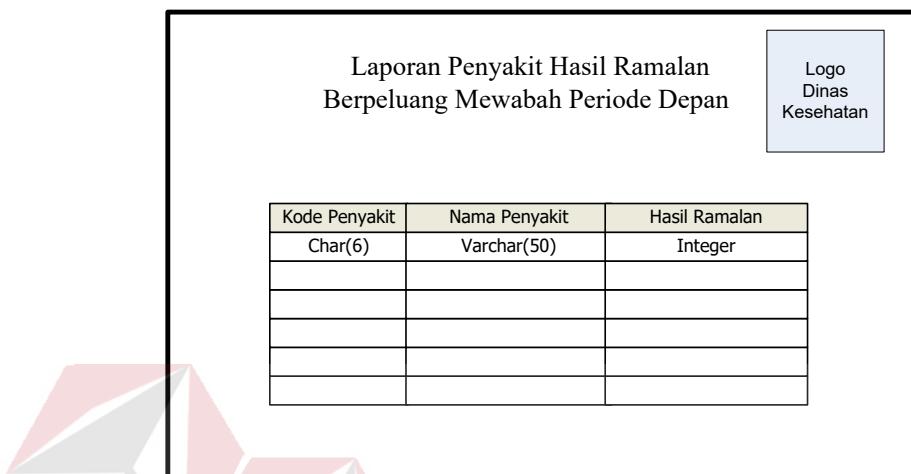
Form untuk melihat hasil perbandingan antara hasil ramalan konvensional dengan *exponential smoothing* biasa dan dengan ramalan hasil bantuan *neural network*.

-- Exponential Smoothing oleh Winter --						-- Neural Network --					
Kode	Pjg Ramalan	Data Aktual	Nilai Ramal	MSE	MAD	Kode	Pjg Ramalan	Data Aktual	Nilai Ramal	MSE	MAD
Char(5)	Integer	Integer	Integer	Float(4)	Float(4)	Char(5)	Integer	Integer	Integer	Float(4)	Float(4)

Gambar 3.30 Desain Form Perbandingan Hasil Ramalan

14. Form Laporan Hasil Ramalan

Form yang berisi laporan hasil ramalan 10 besar penyakit yang berpotensi mewabah untuk periode mendatang.



Laporan Penyakit Hasil Ramalan
Berpeluang Mewabah Periode Depan

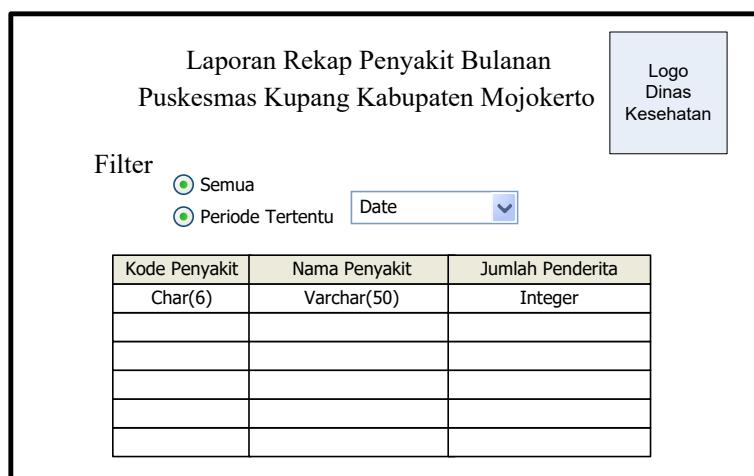
Kode Penyakit	Nama Penyakit	Hasil Ramalan
Char(6)	Varchar(50)	Integer

Logo
Dinas
Kesehatan

Gambar 3.31 Desain Form Laporan Hasil Ramalan

15. Form Laporan Rekap Penyakit

Form yang berisi laporan hasil rekap penyakit bulanan. Selain bisa melihat semua data rekap, terdapat juga pilihan untuk melihat laporan berdasarkan bulan yang dimasukkan.



Laporan Rekap Penyakit Bulanan
Puskesmas Kupang Kabupaten Mojokerto

Filter

Semua
 Periode Tertentu Date

Kode Penyakit	Nama Penyakit	Jumlah Penderita
Char(6)	Varchar(50)	Integer

Logo
Dinas
Kesehatan

Gambar 3.32 Desain Form Laporan Rekap Penyakit

16. Form Laporan Hasil Perbandingan

Form yang berisi laporan hasil perbandingan antara hasil ramalan *exponential smoothing* dengan hasil ramalan dengan bantuan *neural network*.

Laporan Perbandingan Hasil Ramalan Berpeluang Mewabah Periode Depan					Logo Dinas Kesehatan
-- Exponential Smoothing oleh Winter --					-- Neural Network --
Kode Penyakit	Nama Penyakit	Ramalan	MSE	MAD	Kode Penyakit
Char(6)	Varchar(50)	Integer	Float(4)	Float(4)	Char(6)

Gambar 3.33 Desain Form Laporan Hasil Perbandingan

17. Form Laporan Perkiraan Penggunaan Obat

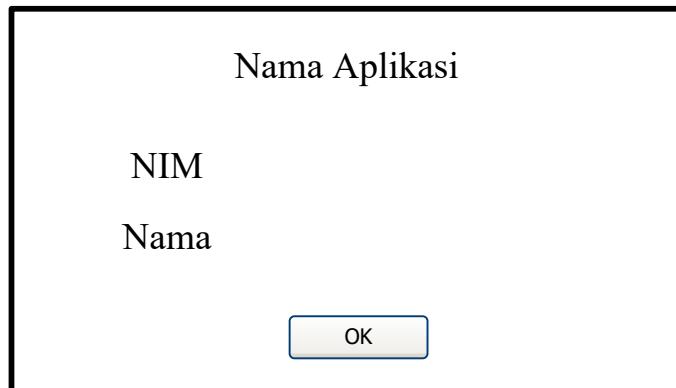
Form yang berisi laporan hasil perbandingan antara hasil ramalan *exponential smoothing* dengan hasil ramalan dengan bantuan *neural network*.

Laporan Rekap Perkiraan Penggunaan Obat Minimal			Logo Dinas Kesehatan
Puskesmas Kupang Kabupaten Mojokerto			
Kode Obat	Nama Obat	Jumlah Stok Minimal	
Char(6)	Varchar(50)	Integer	

Gambar 3.34 Desain Form Laporan Perkiraan Penggunaan Obat

18. Form Tentang Pembuat

Form berisi tentang info aplikasi dan info pembuatnya.



Nama Aplikasi

NIM

Nama

OK

Gambar 3.35 Desain Halaman Tentang Pembuat

3.8 Desain Uji Coba

Desain uji coba bertujuan untuk memastikan bahwa aplikasi telah dibuat dengan benar sesuai dengan kebutuhan atau tujuan yang diharapkan. Kekurangan atau kelemahan aplikasi pada tahap ini akan dievaluasi sebelum diimplementasikan secara nyata. Proses pengujian menggunakan *Black Box Testing* dimana aplikasi akan diuji dengan melakukan berbagai percobaan untuk membuktikan bahwa aplikasi yang telah dibuat sudah sesuai dengan tujuan.

3.8.1 Desain Uji Coba Fitur Dasar Sistem

Pada bagian ini akan dibahas desain uji coba fitur dasar sistem untuk menentukan apakah fitur-fitur dari aplikasi peramalan ini telah berjalan sesuai dengan fungsinya. Tiap fitur yang disediakan akan diuji hasilnya sesuai dengan tabel test case. Desain uji coba fitur dasar sistem dari tiap form aplikasi peramalan adalah sebagai berikut:

1. Desain Uji Coba Fitur Login

Proses login dilakukan dengan cara menginputkan username dan password.

Dari username dan password ini akan diketahui status login, apakah sebagai admin atau operator.

Data login yang digunakan terlihat pada tabel 3.12 berisi username dan password untuk masing-masing user admin dan operator.

Tabel 3.12 Tabel Login

ID	Username	Password	Status
ADM-001	admin	admin	Admin
OPR-001	operator	operator	Operator

Tabel 3.13 Tabel Test Case Login

Test Case ID	Tujuan	Input	Output Diharapkan
1	Deskripsi username dan password yang valid	Memasukkan data login username= <i>admin</i> dan password= <i>admin</i>	Form Login tertutup dan muncul form utama untuk admin
2	Deskripsi username dan password non valid	Memasukkan data login username= <i>operator</i> dan password= <i>coba</i>	Muncul pesan "Data Login Salah"

2. Desain Uji Coba Fitur Manipulasi Data Penyakit

Proses manipulasi data penyakit adalah proses tambah, ubah, dan hapus data.

Proses ini bertujuan untuk mengetahui apakah proses manipulasi data bisa dilakukan melalui aplikasi.

Tabel 3.14 Tabel Penyakit

Kode_Penyakit	Jenis_Penyakit	Nama_Penyakit
P-0103	PENYAKIT INFENSI PADA USUS	DISENTRI
P-0306	PENYAKIT BAKTERI	PES

Tabel 3.15 Tabel Test Case Manipulasi Data Penyakit

Test Case ID	Tujuan	Input	Output Diharapkan
3	Tambah data baru ke tabel penyakit	Memasukkan data penyakit: kode_penyakit= <i>P-0103</i> , jenis_penyakit= <i>PENYAKIT INFEKSI PADA USUS</i> , nama_penyakit= <i>DISENTRI</i> kemudian menekan tombol Simpan	Muncul pesan "Data Berhasil Masuk" dan data baru muncul pada data grid
4	Ubah data dari tabel penyakit	Memasukkan kode_penyakit= <i>P-0306</i> , dari data yang ada, diubah menjadi: jenis_penyakit= <i>PENYAKIT INFEKSI PADA USUS BESAR</i> , nama_penyakit= <i>DISENTRI KELAS BERAT</i> kemudian menekan tombol Edit	Muncul pesan "Data Berhasil Diubah" dan data hasil edit muncul pada data grid
5	Hapus data dari tabel penyakit	Memasukkan kode_penyakit= <i>P-0306</i> , dari data yang ada, kemudian menekan tombol Hapus	Muncul pesan "Data Berhasil Dihapus" dan data yang dihapus, tidak muncul pada data grid
6	Menghindari data kode penyakit kosong pada tabel penyakit	Memasukkan data penyakit: kode_penyakit=(<i>kosong</i>), jenis_penyakit= <i>PENYAKIT INFEKSI PADA USUS</i> , nama_penyakit= <i>DISENTRI</i> kemudian menekan tombol Simpan	Muncul pesan "Kode Penyakit Tidak Boleh Kosong" dan data tidak tersimpan pada tabel penyakit
7	Menghindari data jenis penyakit kosong pada tabel penyakit	Memasukkan data penyakit: kode_penyakit= <i>P-0103</i> , jenis_penyakit=(<i>kosong</i>), nama_penyakit= <i>DISENTRI</i> kemudian menekan tombol Simpan	Muncul pesan "Jenis Penyakit Tidak Boleh Kosong" dan data tidak tersimpan pada tabel penyakit

Tabel 3.15 Tabel Test Case Manipulasi Data Penyakit (lanjutan)

Test Case ID	Tujuan	Input	Output Diharapkan
8	Menghindari data nama penyakit kosong pada tabel penyakit	Memasukkan data penyakit: kode_penyakit=P-0103, jenis_penyakit= PENYAKIT INFEKSI PADA USUS, nama_penyakit=(kosong) kemudian menekan tombol Simpan	Muncul pesan "Nama Penyakit Tidak Boleh Kosong" dan data tidak tersimpan pada tabel penyakit

3. Desain Uji Coba Fitur Manipulasi Data Obat

Proses manipulasi data obat adalah proses tambah, ubah, dan hapus data. Proses ini bertujuan untuk mengetahui apakah proses manipulasi data bisa dilakukan melalui aplikasi. Setiap proses manipulasi data obat akan berpengaruh langsung dengan tabel obat. Tabel 3.16 adalah data contoh yang akan digunakan dan tabel 3.17 adalah test case manipulasi data.

Tabel 3.16 Tabel Obat

Kode_Obat	Nama_Obat	Satuan
O-0111	AMOKSILIN KAPSUL 250 MG	KAP
O-0124	AQUADEST STERIL	BTL

Tabel 3.17 Tabel Test Case Manipulasi Data Obat

Test Case ID	Tujuan	Input	Output Diharapkan
9	Tambah data baru ke tabel obat	Memasukkan data obat: kode_obat=O-0111, nama_obat=AMOKSILIN KAPSUL 250 MG, satuan=KAP kemudian menekan tombol Simpan	Muncul pesan "Data Berhasil Masuk" dan data baru muncul pada data grid

Tabel 3.17 Tabel Test Case Manipulasi Data Obat (lanjutan)

Test Case ID	Tujuan	Input	Output Diharapkan
10	Ubah data dari tabel obat	Memasukkan kode_obat= <i>O-0124</i> , dari data yang ada, diubah menjadi: nama_obat= <i>AQUA</i> , satuan= <i>GALON</i> kemudian menekan tombol Edit	Muncul pesan "Data Berhasil Diubah" dan data hasil edit muncul pada data grid
11	Hapus data dari tabel obat	Memasukkan kode_obat= <i>O-0124</i> , dari data yang ada, kemudian menekan tombol Hapus	Muncul pesan "Data Berhasil Dihapus" dan data yang dihapus, tidak muncul pada data grid
12	Menghindari data kode obat kosong pada tabel obat	Memasukkan data obat: kode_obat=(<i>kosong</i>), nama_obat= <i>AQUADEST STERIL</i> , satuan= <i>BTL</i> kemudian menekan tombol Simpan	Muncul pesan "Kode Obat Tidak Boleh Kosong" dan data tidak tersimpan pada tabel obat
13	Menghindari data nama obat kosong pada tabel obat	Memasukkan data obat: kode_obat= <i>O-0124</i> , nama_obat=(<i>kosong</i>), satuan= <i>TAB</i> kemudian menekan tombol Simpan	Muncul pesan "Nama Obat Tidak Boleh Kosong" dan data tidak tersimpan pada tabel obat
14	Menghindari data satuan kosong pada tabel obat	Memasukkan data obat: kode_obat= <i>O-0124</i> nama_obat= <i>AQUADEST STERIL</i> , satuan=(<i>kosong</i>) kemudian menekan tombol Simpan	Muncul pesan "Satuan Tidak Boleh Kosong" dan data tidak tersimpan pada tabel obat

4. Desain Uji Coba Fitur Manipulasi Data Login

Proses manipulasi data login adalah proses tambah, ubah, dan hapus data.

Proses ini bertujuan untuk mengetahui apakah proses manipulasi data bisa dilakukan melalui aplikasi. Tabel 3.18 adalah data contoh yang digunakan dan tabel 3.19 adalah test case manipulasi data login. Dari test case manipulasi

data login akan dijabarkan desain uji coba yang akan dilakukan. Dari sini akan diketahui apakah fitur dasar ini sudah sesuai dengan yang diharapkan.

Tabel 3.18 Tabel Login

ID	Username	Password	Status
ADM-001	Admin	admin	Admin
OPR-001	operator	operator	Operator

Tabel 3.19 Tabel Test Case Manipulasi Data Login

Test Case ID	Tujuan	Input	Output Diharapkan
15	Tambah data baru ke tabel login	Memasukkan data login: username= <i>admin</i> , password= <i>admin</i> , ulangi password= <i>admin</i> , status= <i>Admin</i> kemudian menekan tombol Simpan	Muncul pesan "Data Berhasil Masuk" dan data baru muncul pada data grid
16	Ubah data dari tabel login	Memilih data dari gridview dengan ID= <i>OPR-001</i> , dari data yang ada, diubah menjadi: username= <i>opera</i> , password= <i>opera</i> kemudian menekan tombol Edit	Muncul pesan "Data Berhasil Diubah" dan data hasil edit muncul pada data grid
17	Hapus data dari tabel login	Memilih data dari gridview dengan ID= <i>OPR-001</i> , dari data yang ada, kemudian menekan tombol Hapus	Muncul pesan "Data Berhasil Dihapus" dan data yang dihapus, tidak muncul pada data grid
18	Menghindari data username kosong pada tabel login	Memasukkan data login: Username= <i>(kosong)</i> , password= <i>operator</i> , ulangi password= <i>operator</i> , status= <i>Operator</i> kemudian menekan tombol Simpan	Muncul pesan "Username Tidak Boleh Kosong" dan data tidak tersimpan pada tabel login

Tabel 3.19 Tabel Test Case Manipulasi Data Login (lanjutan)

Test Case ID	Tujuan	Input	Output Diharapkan
19	Menghindari data password kosong pada tabel login	Memasukkan data login: Username= <i>operator</i> , password= <i>(kosong)</i> , ulangi password= <i>operator</i> , status= <i>Operator</i> kemudian menekan tombol Simpan	Muncul pesan "Password Tidak Boleh Kosong" dan data tidak tersimpan pada tabel login
20	Menghindari kesalahan pengisian password	Memasukkan data login: Username= <i>operator</i> , password= <i>oprtr</i> , ulangi password= <i>operator</i> , status= <i>Operator</i> kemudian menekan tombol Simpan	Muncul pesan "Password Tidak Sama" dan data tidak tersimpan pada tabel login

5. Desain Uji Coba Fitur Pengambilan Data Berpola Musiman

Proses ini mengambil beberapa data penyakit yang mempunyai kecenderungan berpola musiman dari database. Penyakit yang mempunyai status klasifikasi 1 yang diambil dalam proses ini karena tidak semua penyakit memiliki pola musiman. Setelah itu hanya 10 jenis penyakit yang diambil dalam proses ini untuk digunakan dalam aplikasi peramalan. Tabel 3.20 adalah data contoh, jika mempunyai status 1 berarti data tersebut berpola musiman, jika status 0 berarti tidak mempunyai pola musiman. Tabel 3.21 menunjukkan test case pengambilan data.

Tabel 3.20 Tabel Hasil_Klasifikasi

Kode_Penyakit	Status_Klasifikasi
P-0103	1
P-0306	0

Tabel 3.21 Tabel Test Case Pengambilan Data

Test Case ID	Tujuan	Input	Output Diharapkan
21	Mengambil data penyakit	Memilih radio button Default atau Manual, lalu menekan tombol Proses	Progress Bar bergerak sampai 100%, lalu data keluar pada datagrid
22	Mengecek tanggal awal tidak boleh lebih dari tanggal akhir	Memilih radio button Manual, lalu memilih periode awal=JAN 2006 dan periode akhir=DES 2005 kemudian menekan tombol Proses	Muncul pesan "Periode Akhir Tidak Boleh Kurang Dari Periode Awal" dan data tidak muncul pada datagrid
23	Memunculkan 10 jenis penyakit hasil klasifikasi	Menekan tombol Klasifikasi	Form Klasifikasi tertutup dan muncul Form Hasil Klasifikasi beserta 10 jenis penyakit pada datagrid

6. Desain Uji Coba Fitur Peramalan *Exponential Smoothing* oleh Winter
- Proses ini melakukan peramalan dengan metode *exponential smoothing* oleh Winter. Ada inputan yang dimasukkan yaitu panjang ramalan dan panjang musim.

Tabel 3.22 Tabel Test Case Peramalan

Test Case ID	Tujuan	Input	Output Diharapkan
24	Menghasilkan hasil ramalan untuk periode depan	Memasukkan panjang ramalan=1 dan panjang musim=4 lalu menekan tombol Ramal	Progress Bar bergerak sampai 100%, lalu data hasil ramalan keluar pada datagrid Hasil Ramalan, sedangkan list error keluar pada datagrid List Error
25	Mengecek data tersimpan sebagai data latih atau tidak	Setelah proses peramalan, menekan tombol Simpan	Muncul pesan "Semua Data Telah Tersimpan"

7. Desain Uji Coba Fitur Pelatihan Data *Neural Network*

Proses ini melakukan proses pelatihan pola yang akan digunakan pada proses *neural network*.

Tabel 3.23 Tabel Test Case Pelatihan

Test Case ID	Tujuan	Input	Output Diharapkan
26	Memilih data latih dengan panjang inputan data time series	Memilih nilai panjang neuron input=5 lalu menekan tombol OK	Semua data latih yang mempunyai panjang data time series sepanjang 5 periode muncul pada datagrid
27	Menyimpan data latih pada XML	Setelah memilih data time series, menekan tombol Simpan	Muncul pesan "Data Latih Telah Tersimpan"
28	Melakukan proses pelatihan	Menekan tombol Mulai	Tombol Mulai berubah menjadi tombol Berhenti dan proses pelatihan berjalan ditunjukkan dengan textbox Iterasi yang melakukan counter
29	Memberhentikan proses pelatihan	Menekan tombol Berhenti	Counter pada textbox iterasi dan proses pelatihan berhenti lalu tombol Berhenti berubah menjadi tombol Mulai
30	Menyimpan data hasil pelatihan	Setelah proses pelatihan, menekan tombol Simpan	Keluar grafik hasil pelatihan dan data tersimpan

8. Desain Uji Coba Fitur Pencarian Pola NN

Proses ini melakukan proses pencarian pola NN terbaik untuk tiap penyakit lalu menyimpan hasilnya pada database.

Tabel 3.24 Tabel Test Case Pencarian Pola NN Terbaik

Test Case ID	Tujuan	Input	Output Diharapkan
31	Menampilkan nama penyakit	Memilih kode penyakit=P-0103 pada combo box kode penyakit	Nama penyakit=Disentri keluar pada textbox nama penyakit
32	Menjalankan looping mencari pola NN	Memilih kode penyakit pada combo box kode penyakit lalu menekan tombol Mulai	Tombol Mulai berubah menjadi tombol Berhenti dan proses looping berjalan ditunjukkan dengan bertambahnya data pada datagrid
33	Memberhentikan proses	Menekan tombol Berhenti	Data pada datagrid tidak bertambah lagi, counter pada textbox jumlah neuron input dan hidden juga berhenti lalu tombol Berhenti berubah menjadi tombol Mulai
34	Mengambil pola terbaik	Setelah proses looping selesai, menekan link button Ambil Pola Terbaik	Pola terbaik muncul pada bagian bawah ditandai dengan MSE dan MAD terkecil, tombol Simpan menjadi aktif
35	Menyimpan pola terbaik	Setelah mengambil data pola terbaik, menekan tombol Simpan	Keluar pesan "Data Telah Tersimpan"

9. Desain Uji Coba Fitur Pencarian Panjang Ramalan

Proses ini melakukan proses pencarian panjang ramalan terbaik untuk tiap penyakit lalu menyimpan hasilnya pada database. Tiap penyakit akan diuji untuk mendapatkan panjang ramalan terbaik. Tabel 3.25 menunjukkan test case pencarian panjang ramalan terbaik untuk tiap penyakit.

Tabel 3.25 Tabel Test Case Pencarian Panjang Ramalan Terbaik

Test Case ID	Tujuan	Input	Output Diharapkan
36	Menampilkan nama penyakit	Memilih kode penyakit=P-0103 pada combo box kode penyakit	Nama penyakit=Disentri keluar pada textbox nama penyakit
37	Menjalankan looping mencari panjang ramalan	Memilih kode penyakit pada combo box kode penyakit lalu menekan tombol Mulai	Tombol Mulai berubah menjadi tombol Berhenti dan proses looping berjalan ditunjukkan dengan bertambahnya data pada datagrid
38	Memberhentikan proses	Menekan tombol Berhenti	Data pada datagrid tidak bertambah lagi, counter angka juga berhenti lalu tombol Berhenti berubah menjadi tombol Mulai
39	Mengambil panjang ramalan terbaik	Setelah proses looping selesai, menekan link button Ambil Nilai Terbaik	Nilai terbaik muncul pada bagian bawah ditandai dengan MSE dan MAD terkecil, tombol Simpan menjadi aktif
40	Menyimpan panjang ramalan terbaik	Setelah mengambil data panjang ramalan terbaik, menekan tombol Simpan	Keluar pesan "Data Telah Tersimpan"

10. Desain Uji Coba Fitur Perbandingan Hasil Ramalan

Proses ini melakukan proses perbandingan hasil ramalan exponential smooting oleh Winter konvensional tanpa memanfaatkan neural network dan peramalan dengan exponential smooting oleh Winter yang menggunakan bantuan neural network untuk penentuan konstanta.

Tabel 3.26 Tabel Test Case Perbandingan Hasil Ramalan

Test Case ID	Tujuan	Input	Output Diharapkan
41	Menampilkan nama penyakit	Memilih kode penyakit=P-0103 pada combo box kode penyakit	Nama penyakit=Disentri keluar pada textbox nama penyakit
42	Mencegah inputan kode penyakit kosong	Menekan tombol Mulai dengan inputan lengkap kecuali kode penyakit yang sengaja tidak dipilih	Keluar pesan "Pilih Kode Penyakit Dulu"
43	Mencegah inputan kosong pada panjang ramalan	Menekan tombol Mulai dengan inputan lengkap kecuali panjang ramalan yang sengaja tidak diisi	Keluar pesan "Masukkan Panjang Ramalan Dulu"
44	Mencegah inputan kosong pada neuron hidden	Menekan tombol Mulai dengan inputan lengkap kecuali neuron hiddeng yang sengaja tidak diisi	Keluar pesan "Masukkan Panjang Neuron Hidden Dulu"
45	Menggunakan anjuran panjang ramalan	Menekan link button Gunakan di sebelah anjuran panjang ramalan terbaik	Nilai panjang ramalan terbaik muncul pada textbox Panjang Ramalan
46	Menggunakan anjuran pola NN	Menekan link button Gunakan di sebelah anjuran pola NN terbaik	Nilai pola NN terbaik muncul pada textbox Jumlah Neuron Input, Jumlah Neuron Hidden, dan Jumlah Neuron Output
47	Menjalankan looping untuk melakukan proses peramalan	Setelah semua inputan lengkap, menekan tombol Mulai	Tombol Mulai berubah menjadi tombol Berhenti dan proses looping berjalan ditunjukkan dengan bertambahnya data pada datagrid
48	Menyimpan panjang ramalan terbaik	Setelah mengambil data panjang ramalan terbaik, menekan tombol Simpan	Keluar pesan "Data Telah Tersimpan"

11. Desain Uji Coba Fitur Laporan

Proses ini untuk menghasilkan laporan yang diambil dari database dan ditampilkan dalam form lewat *crystal report*. Tabel 3.27 menunjukkan test case untuk menguji apakah fitur laporan berhasil sesuai dengan yang diharapkan.

Tabel 3.27 Tabel Test Case Laporan

Test Case ID	Tujuan	Input	Output Diharapkan
49	Menampilkan laporan hasil peramalan	Menekan menu Laporan, lalu memilih sub menu Laporan Hasil Ramalan	Form Laporan Hasil Peramalan muncul dan data laporan tampil pada crystal report
50	Menampilkan laporan rekap penyakit	Menekan menu Laporan, lalu memilih sub menu Laporan Rekap Penyakit	Form Laporan Rekap Penyakit muncul dan data laporan tampil pada crystal report
51	Menampilkan laporan hasil perbandingan	Menekan menu Laporan, lalu memilih sub menu Laporan Perbandingan Hasil Ramalan	Form Laporan Hasil Perbandingan muncul dan data laporan tampil pada crystal report
52	Menampilkan laporan perkiraan penggunaan obat	Menekan menu Laporan, lalu memilih sub menu Laporan Perkiraan Penggunaan Obat	Form Laporan Perkiraan Obat muncul dan data laporan tampil pada crystal report

3.8.2 Desain Uji Coba Proses Pelatihan NN dan Pengujian NN

Uji coba proses pelatihan NN dilakukan dengan memilih beberapa data penyakit dengan panjang *time series* tertentu. Selain itu juga dimasukkan data panjang ramalan dan data panjang musim. Untuk proses keluaran data latih, berupa tiga konstanta untuk proses peramalan.

Tabel 3.28 Tabel Test Case Pelatihan NN

Test Case ID	Tujuan	Input	Output Diharapkan
53	Mengetahui proses pelatihan NN bisa menghasilkan output	Data skenario jenis penyakit sebanyak 10 baris masing-masing 12 data <i>time series</i> dengan menggunakan salah satu pola NN	<i>Mean Square Error</i> (MSE)

Tabel 3.29 Tabel Test Case Pengujian NN

Test Case ID	Tujuan	Input	Output Diharapkan
54	Menguji tingkat kebenaran NN	Data skenario dari data latih, dicoba satu persatu, dengan pola NN yang sama dengan proses pelatihan	Tiga konstanta untuk proses peramalan yang nilainya sama dengan target data latih

Tabel 3.30 Tabel Test Case Pencarian Pola NN Terbaik

Test Case ID	Tujuan	Input	Output Diharapkan
55	Mengetahui pola NN terbaik untuk tiap penyakit	Data skenario berupa data penyakit dan pola NN hasil looping mulai 5-1-3 sampai 15-100-3 yang berjalan dengan kelipatan 10	Pola NN terbaik, <i>Mean Square Error</i> (MSE), <i>Mean Absolute Deviation</i> (MAD), tiga konstanta peramalan

Tabel 3.31 Tabel Test Case Pencarian Panjang Periode Ramal Terbaik

Test Case ID	Tujuan	Input	Output Diharapkan
56	Mengetahui panjang ramalan terbaik untuk tiap penyakit	Data skenario berupa data penyakit dan pola NN hasil looping mulai periode 1 sampai 8	Panjang periode terbaik, <i>Mean Square Error</i> (MSE), <i>Mean Absolute Deviation</i> (MAD), tiga konstanta peramalan

3.8.3 Desain Uji Coba Proses Perbandingan Peramalan Dengan Bantuan NN

Uji coba proses peramalan exponential smoothing oleh Winter dengan bantuan NN adalah menggunakan tiga konstanta hasil pelatihan pola NN. Konstanta ini akan dipakai dalam rumus proses peramalan. Proses ini bertujuan untuk mengetahui seberapa tepat bantuan dari NN dalam proses peramalan yang ditandai dengan nilai MSE dan MAD yang dihasilkan.

Tabel 3.32 Tabel Test Case Perbandingan Peramalan

Test Case ID	Tujuan	Input	Output Diharapkan
57	Membandingkan MSE dan MAD peramalan tanpa bantuan NN dan dengan bantuan NN	Data time series penyakit, pola NN	MSE dan MAD untuk tiap peramalan tanpa bantuan NN dan dengan bantuan NN



BAB IV

IMPLEMENTASI DAN EVALUASI

4.1 Kebutuhan Sistem

Sebelum dapat mengimplementasikan dan menjalankan aplikasi peramalany ini, dibutuhkan perangkat keras dan perangkat lunak dengan kondisi tertentu agar dapat berjalan dengan baik. Berikut ini adalah kebutuhan minimal perangkat keras untuk menjalankan aplikasi:

- a. Komputer dengan prosesor Pentium 733 MHz
- b. Memori 256 MB,

sedangkan kebutuhan perangkat lunak digunakan untuk pengembangan aplikasi ini adalah sebagai berikut :

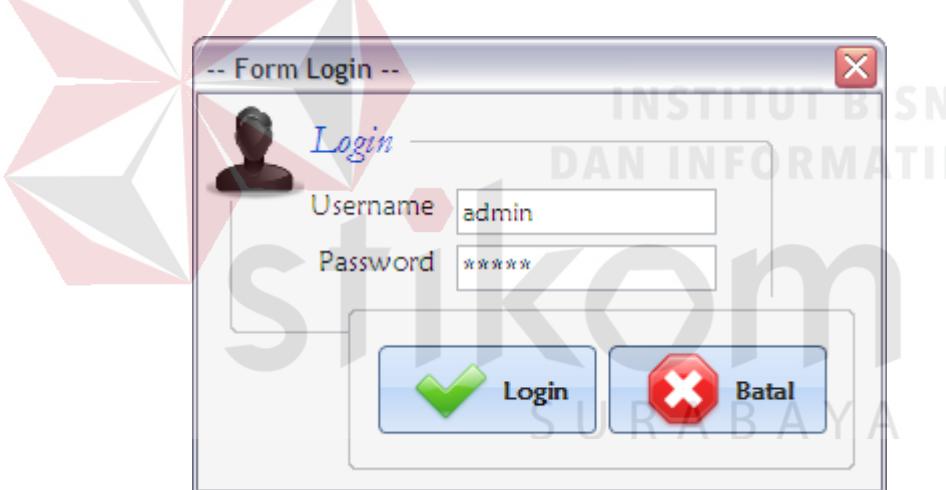
- a. .NET Framework 2.0
- b. Sistem Operasi Windows 2000 atau Windows XP Professional atau Windows Server 2003
- c. Microsoft SQL Server 2000

4.2 Implementasi Sistem

Pada sub bab ini akan dijelaskan tentang penggunaan aplikasi yang telah dibuat, yaitu aplikasi untuk administrator maupun aplikasi untuk pengguna, mulai dari tampilan aplikasi, fungsi dan cara penggunaannya. Pertama kali menjalankan aplikasi yang muncul adalah form login, yaitu form untuk memasukkan username dan password yang akan menentukan status login dan hak yang diperoleh. Ada dua status login pada aplikasi peramalan ini yaitu admin dan operator biasa.

Pada gambar 4.1 terlihat form yang pertama kali muncul ketika aplikasi berjalan. Terdapat 2 textbox isian yaitu *username* dan *password*, pada isian textbox *password* secara otomatis inputan akan diubah menjadi karakter bintang atau *asterix* (*).

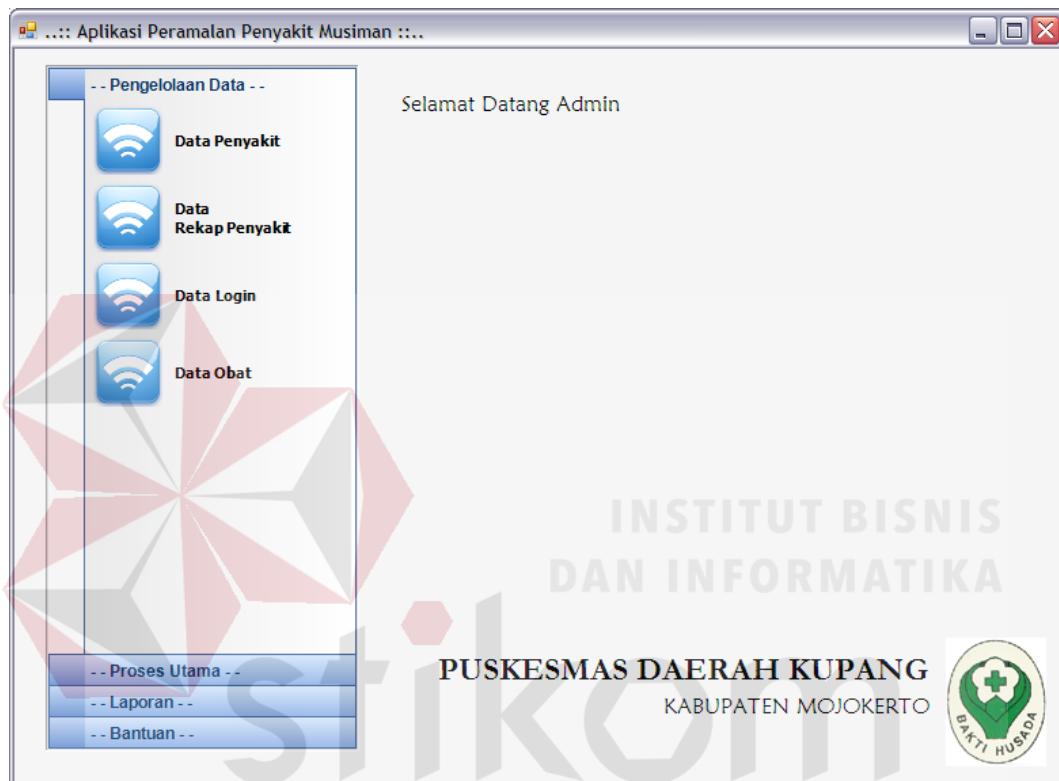
Pada bagian bawah terdapat 2 tombol yaitu tombol Login dan tombol Batal. Tombol Login digunakan untuk mengecek apakah data login yang dimasukkan benar apa tidak. Apabila data login benar, status data login akan diketahui dan akan mempengaruhi hak dari user tadi. Tombol Batal digunakan untuk membersihkan inputan pada *textbox username* dan *password* serta membatalkan proses login.



Gambar 4.1 Form Login

Apabila user login sebagai admin, maka secara otomatis user akan diarahkan ke form utama admin. Sedangkan user dengan status operator akan otomatis diarahkan ke form utama admin. Perbedaan dari status login adalah hak yang dimiliki oleh user dalam aplikasi peramalan ini, hal ini ditunjukkan dengan perbedaan masing-masing menu yang muncul untuk tiap user.

Pada gambar 4.2 terlihat form utama yang muncul setelah user login dengan status sebagai admin. Pada sebelah kiri form terdapat menu yang menunjukkan apa saja yang bisa dilakukan oleh admin. Selain itu terdapat label Selamat Datang Admin pada bagian atas form.



Gambar 4.2 Form Utama Admin

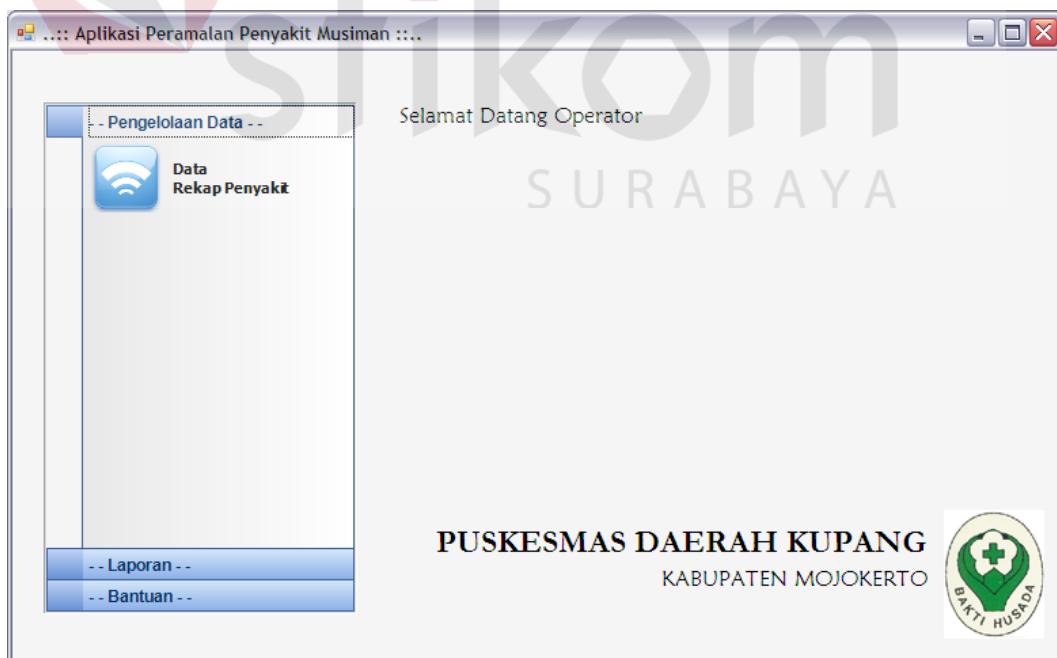
Gambar 4.3 menunjukkan menu yang muncul apabila user login sebagai admin. Ada 4 menu utama yaitu:

1. Pengelolaan Data : Untuk melakukan pengelolaan data
2. Proses Utama : Untuk melakukan proses utama seperti peramalan dan proses *neural network*
3. Laporan : Untuk menampilkan beberapa laporan
4. Bantuan : Untuk menampilkan file *help.chm* dan form *about*



Gambar 4.3 Tampilan Menu Admin

Pada gambar 4.4 terlihat form utama yang muncul setelah user login dengan status sebagai operator. Terdapat menu yang menunjukkan apa saja yang bisa dilakukan oleh operator.



Gambar 4.4 Form Utama Operator

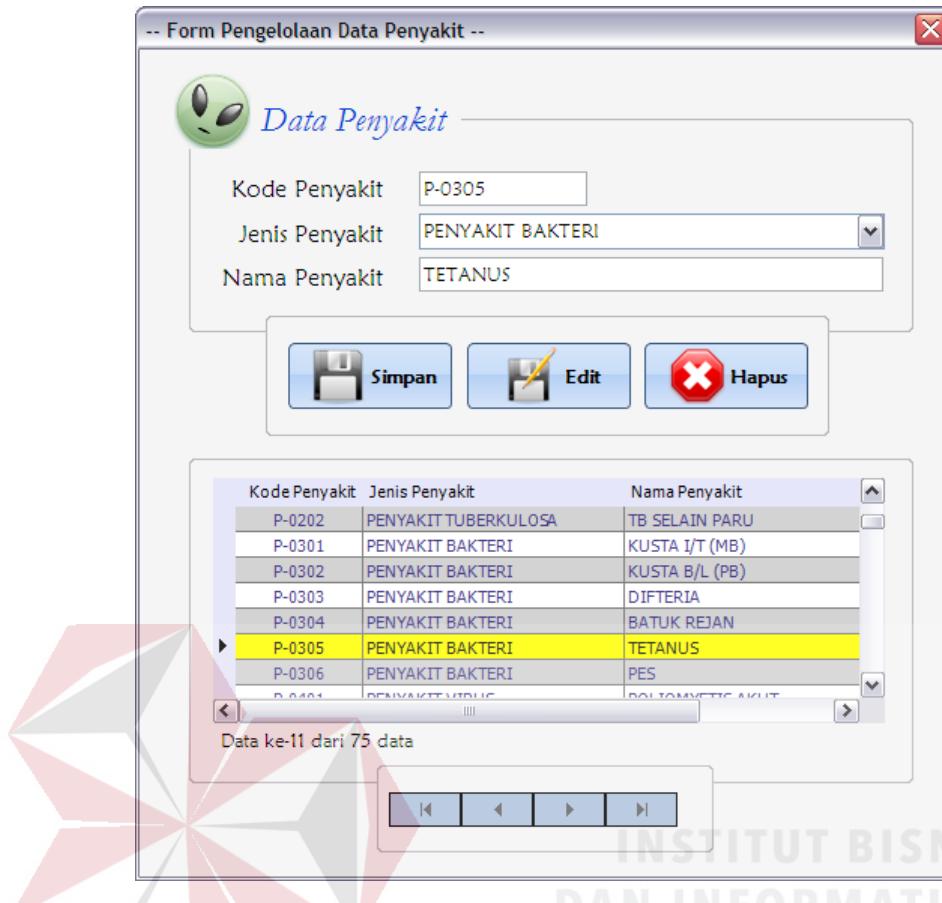
Gambar 4.5 menunjukkan menu yang muncul apabila user login sebagai operator. Ada 3 menu utama yaitu:

1. Pengelolaan Data : Untuk melakukan pengelolaan data rekap penyakit bulanan
2. Laporan : Untuk menampilkan beberapa laporan
3. Bantuan : Untuk menampilkan file *help.chm* dan form *about*



Gambar 4.5 Tampilan Menu Operator

Proses pengelolaan data penyakit ditunjukkan pada gambar 4.6 yang merupakan *capture* tampilan form pengelolaan data penyakit. Terdapat 3 isian yang perlu dimasukkan yaitu kode penyakit, jenis penyakit, dan nama penyakit. Selain itu terdapat 3 tombol yaitu tombol Simpan, tombol Edit, dan tombol Hapus untuk melakukan simpan data, edit data, dan hapus data. Ketika data yang ada pada datagrid dipilih, maka detail data tersebut akan ditampilkan pada control yang ada di atasnya sesuai dengan data pada datagrid. Terdapat juga keterangan posisi data yang dipilih pada datagrid yang berupa label di bawah datagrid.



Gambar 4.6 Form Pengelolaan Data Penyakit



Gambar 4.7 Pesan Berhasil Pengelolaan Data

Gambar 4.7 menunjukkan beberapa pesan yang akan muncul ketika proses pengelolaan data pernyakit berhasil dilakukan. Gambar 4.8 menunjukkan beberapa pesan yang akan muncul ketika proses pengelolaan data pernyakit gagal dilakukan. Masing-masing pesan menunjukkan jenis kesalahan yang terjadi ketika proses pengelolaan data.



Gambar 4.8 Pesan Kesalahan Pengelolaan Data

Proses pengelolaan data obat ditunjukkan pada gambar 4.9 yang merupakan *capture* tampilan form pengelolaan data obat. Terdapat 3 isian yang perlu dimasukkan yaitu kode obat, nama obat, dan satuan. Selain itu terdapat 3 tombol yaitu tombol Simpan, tombol Edit, dan tombol Hapus untuk melakukan simpan data, edit data, dan hapus data. Semua data akan ditampilkan pada datagrid di bagian bawah, apabila data dipilih pada datagrid maka data tadi juga ditampilkan sesuai detail data pada datagrid.

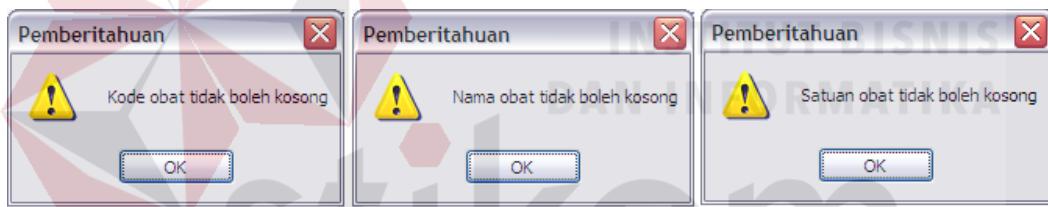
Kode Obat	Nama Obat	Satuan
O-0101	AIR RAKSA DENTAL USE	BTL
O-0102	ALAT SUNTIK SEKALI PAKAI 1 ML	SET
O-0103	ALAT SUNTIK SEKALI PAKAI 2.5 ML	SET
O-0104	ALAT SUNTIK SEKALI PAKAI 5 ML	SET
O-0105	ALBENDAZOL TABLET 400 MG	TAB
O-0106	ALOPURINOL TABLET 100 MG	TAB
O-0107	AMINOFILIN TABLET 200 MG	TAB

Gambar 4.9 Form Pengelolaan Data Obat



Gambar 4.10 Pesan Berhasil Pengelolaan Data

Gambar 4.10 menunjukkan beberapa pesan yang akan muncul ketika proses pengelolaan data obat berhasil dilakukan. Gambar 4.11 menunjukkan beberapa pesan yang akan muncul ketika proses pengelolaan data obat gagal dilakukan karena ada isian yang masih kosong. Setiap pesan error yang muncul menunjukkan kolom dari tabel obat mana yang belum terisi dengan benar.



Gambar 4.11 Pesan Kesalahan Pengelolaan Data

Proses pengelolaan data login ditunjukkan pada gambar 4.12 yang merupakan *capture* tampilan form pengelolaan data login. Terdapat 4 isian yang perlu dimasukkan yaitu username, password, ulangi password dan jenis user. Selain itu terdapat 3 tombol yaitu tombol Simpan, tombol Edit, dan tombol Hapus untuk melakukan simpan data, edit data, dan hapus data. Proses ini akan berpengaruh langsung dengan tabel login pada database. Textbox username untuk memasukkan username, combobox jenis user untuk menentukan status user, dan textbox password untuk memasukkan password. Terdapat juga textbox ulangi password untuk mengecek kebenaran password yang telah dimasukkan.

-- Form Pengelolaan Data Login --

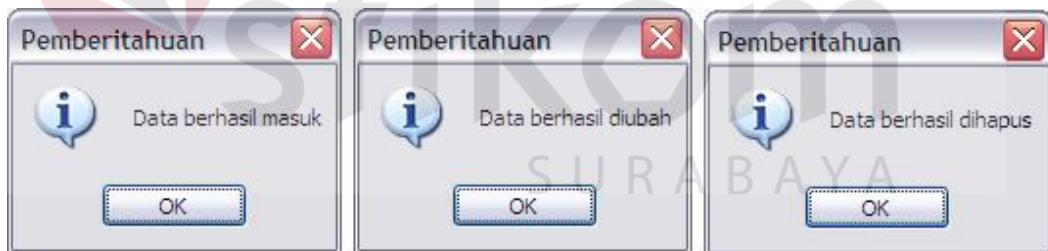
Data Login

Username	operator	Jenis User	Operator
Password	*****	Ulangi Password	*****

ID	Username	Password	Status
ADM-001	admin	*****	Admin
OPR-001	operator	*****	Operator
OPR-002	sasa	***	Operator
*			

Data ke-2 dari 3 data

Gambar 4.12 Form Pengelolaan Data Login



Gambar 4.13 Pesan Berhasil Pengelolaan Data

Gambar 4.13 menunjukkan beberapa pesan yang akan muncul ketika proses pengelolaan data login berhasil dilakukan. Gambar 4.14 menunjukkan beberapa pesan yang akan muncul ketika proses pengelolaan data login gagal dilakukan karena ada isian yang masih kosong. Masing-masing pesan menunjukkan kolom dari tabel login apa yang masih belum terisi dengan benar.



Gambar 4.14 Pesan Kesalahan Pengelolaan Data

Proses pengelolaan data rekap penyakit bulanan ditunjukkan pada gambar 4.15 di bawah ini. Terdapat 1 *textbox* di bagian paling atas untuk memasukkan kode rekap. Jika *link button* Data Baru ditekan, maka hanya tombol Simpan yang aktif dan isian dianggap sebagai data baru. Sedangkan jika link button Edit Data ditekan, maka hanya tombol Edit yang aktif dan isian dianggap data lama yang diubah. Isian data dilakukan langsung pada gridview, setiap pengelolaan data pada form ini akan berpengaruh langsung pada tabel Rekap_Penyakit_Bulanan di database.

Kode Penyakit	Jumlah Penderita
P-0101	52
P-0102	0
P-0103	23
P-0104	11
P-0201	39
P-0202	4
P-0301	68
P-0302	27
P-0303	19
P-0304	n

Gambar 4.15 Form Pengelolaan Rekap Penyakit Bulanan

Proses klasifikasi penyakit yang memiliki pola musiman ditunjukkan pada gambar 4.16 di bawah ini. Inputan hanya berupa periode yang akan digunakan untuk menentukan panjang periode data rekap penyakit sebagai data dasar. Apabila combobox default yang dipilih, maka data dasar yang digunakan adalah data selama tiga tahun pada periode sebelumnya. Sedangkan jika combobox manual yang dipilih, maka periode data dasar yang digunakan adalah sesuai dengan periode inputan. Setelah ditekan tombol Proses, maka hasilnya akan keluar pada gridview di bawahnya.

Kode Penyakit	Nama Penyakit	Total Pasien
P-0102	DIARE	3370
P-0103	DISENTRI	338
P-0104	INFEKSI PADA USUS LAINNYA	1069
P-0201	TB PARU	711
P-0406	CACAR AIR	140
P-0703	PENYAKIT KECACINGAN	46
P-0801	GANGGUAN PSIKOTIK	454
P-1002	KATARAK	40
P-1003	KELAINAN RETRAKSI	1137
P-1005	PENYAKIT MATA LAINNYA	1069
P-1101	INFEKSI TELINGA TENGAH	876
P-1204	DIAULITIS	2210

Gambar 4.16 Form Klasifikasi

Setelah ditekan tombol Klasifikasi dari proses di atas, akan keluar klasifikasi hasil sebanyak 10 jenis penyakit saja yang nantinya akan dilakukan proses peramalan. Tampilan ini ditunjukkan pada gambar 4.17 yang menunjukkan kode penyakit, nama penyakit, dan total pasien selama periode tersebut.

-- Form Hasil Klasifikasi --

Hasil Klasifikasi

Kode Penyakit	Nama Penyakit	Total Pasien
P-0703	PENYAKIT KECACINGAN	46
P-1706	HYPEREMESIS	144
P-1002	KATARAK	40
P-1601	PENYAKIT SALURAN KENCING	203
P-0406	CACAR AIR	140
P-1901	KECELAKAAN & RUDA PAKSA	813
P-1701	KEGUGURAN	53
P-0103	DISENTRI	338
P-2001	PENYAKIT KULIT INFENSI	2308
P-2002	PENYAKIT KULIT ALERGI	2673
*		

 Simpan  Batal

Gambar 4.17 Form Hasil Klasifikasi

Pada gambar 4.18 dapat dilihat proses peramalan dengan exponential smoothing oleh Winter. Data yang digunakan sebagai data dasar peramalan adalah 10 jenis penyakit hasil proses klasifikasi sebelumnya. Selain itu ada inputan panjang ramalan untuk menentukan panjang periode peramalan dan panjang musiman untuk menentukan banyak periode dalam satu tahun. Setelah ditekan tombol Ramal, hasil peramalan untuk tiap penyakit akan keluar pada *datagrid* di bawahnya berupa nilai peramalan tiap periode sesuai panjang ramalan yang dimasukkan.

Pada bagian *datagrid* Daftar Error terdapat daftar *error* untuk tiap kali peramalan yang merupakan hasil pengurangan antara nilai hasil ramalan dengan nilai data aktual. Terdapat juga nilai konstanta peramalan yang digunakan berupa alpha, beta, dan teta. Setelah proses ini dilakukan, tombol Simpan bisa ditekan untuk menyimpan data hasil proses ini ke database.

-- Form Peramalan Exp Smoothing --

Periode

Panjang Ramalan 2 Panjang Musim 4 Ramal Simpan

Proses = 100 %

Hasil Peramalan

Kode Penyak	Nama Penyak	Periode-1	Periode-2
P-0103	DISENTRI	33	32
P-0406	CACAR AIR	15	17
P-0703	PENYAKIT KE	4	4
P-1002	KATARAK	8	33
P-1401	PNEUMONIA	24	31
P-1601	PENYAKIT SAL	37	33
P-1701	KEGUGURAN	5	4
P-1901	KECELAKAAN	79	81
P-2001	PENYAKIT KUL	227	225

10

Daftar Error

Kode Penyak	Alpha	Beta
P-0103	0.1	0.1
P-0406	0.2	0.1
P-0703	0.1	0.1
P-1002	0.2	0.9
P-1401	0.1	0.8
P-1601	0.1	0.6
P-1701	0.2	0.1
P-1901	0.1	0.4
P-2001	0.1	0.0

10

Gambar 4.18 Form Peramalan

Untuk mendukung proses neural network, dibutuhkan data latih dari hasil peramalan sebelumnya. Pada gambar 4.19 ditunjukkan pemilihan panjang neuron input yang akan digunakan sebagai data latih. Setelah ditekan tombol OK, maka akan muncul data latih sesuai jumlah data aktual yang dimasukkan.

-- Form Data Latih --

Filter Neuron Input

Panjang Neuron Input 5 OK

Data Latih

Kode_Latih	Jumlah_DtA	Dt_Aktual01	Dt_Aktual02	Dt_Aktual03	Dt_Aktual04	Dt_Aktual05
DL05-0001	5	0.004	0.006	0.004	0.017	0.005
DL05-0002	5	0.007	0.008	0.003	0.002	0.005
DL05-0003	5	0.021	0.034	0.03	0.017	0.025
DL05-0004	5	0.036	0.067	0.034	0.028	0.021
DL05-0005	5	0.173	0.155	0.18	0.16	0.188
DL05-0006	5	0.023	0.007	0.01	0.012	0.015
DL05-0007	5	0.002	0.013	0.005	0.009	0.002
DL05-0008	5	0.009	0.012	0.018	0.016	0
DL05-0009	5	0.051	0.051	0.057	0.00	0.051

Simpan Batal

Gambar 4.19 Form Data Latih

Proses pelatihan ditunjukkan pada gambar 4.20 yang berfungsi untuk melakukan pelatihan terhadap pola *NN*, proses ini bisa dilakukan secara satu persatu dengan menekan tombol Tambah Satu Iterasi atau dengan menekan tombol Mulai yang akan berhenti setelah MSE mencapai 0. Ketika tombol Mulai ditekan, maka tulisan menjadi Berhenti untuk menghentikan proses *looping*. Textbox iterasi menunjukkan jumlah iterasi yang telah dilakukan pada proses pelatihan. Terdapat juga link button untuk melihat hasil grafik dari proses pelatihan neural network berupa nilai MSE dan panjang iterasi.



Gambar 4.20 Form Pelatihan

Pada gambar 4.21 bisa dilihat proses pencarian pola *NN* terbaik untuk tiap penyakit dengan cara *looping*. Setelah memilih kode penyakit dan menekan tombol Mulai, maka proses *looping* akan berjalan. Setelah semua proses selesai, lakukan pada *link button* Ambil Pola Terbaik untuk mendapatkan pola yang paling baik ditandai dengan MSE dan MAD yang paling kecil dari semua. Tombol Simpan digunakan untuk menyimpan data pola *NN* terbaik dari jenis penyakit yang sedang diuji.

-- Form Cari Pola NN --

Data Pendukung

Jumlah Neuron Input	15	Kode Penyakit	P-0406
Jumlah Neuron Hidden	100	Nama Penyakit	CACAR AIR
Jumlah Neuron Output	3		

Hasil Pola NN

ID	PolaNN	MSE	MAD	Alpha	Beta	Teta
P-001	7-1-3	101.82	7.58	0.8155101	0.5680959	0.7256774
P-002	7-10-3	76.13	6.62	0.7425193	0.2874114	0.1501662
P-003	7-20-3	54.21	5.41	0.3761352	0.4495228	0.5926788
P-004	7-30-3	123.81	8.24	0.7789363	0.9414694	0.3598334
P-005	7-40-3	43.35	3.9	0.02467852	0.03112332	0.9515021
P-006	7-50-3	127.74	8.35	0.9403043	0.6171395	0.9331846
P-007	7-60-3	45.11	4.43	0.1479926	0.6439975	0.9567043
P-008	7-70-3	92.92	7.27	0.6091856	0.9505332	0.7119362
P-009	7-80-3	82.28	6.87	0.995628	0.04291296	0.7732476
P-010	7-90-3	43.40	4	0.0000102020	0.00005045	0.7510125

[Ambil Pola Terbaik](#)

Pola Terbaik = **7-40-3**

Nilai MSE	43.35	Nilai Alpha	0.02
Nilai MAD	3.90	Nilai Beta	0.03
		Nilai Teta	0.95

Gambar 4.21 Form Pencarian Pola NN Terbaik

Pada gambar 4.22 bisa dilihat proses pencarian panjang periode ramalan terbaik untuk tiap penyakit dengan cara *looping*. Setelah memilih kode penyakit dan menekan tombol Mulai, maka proses looping akan berjalan. Proses looping dilakukan sebanyak 8 periode peramalan atau setara dengan periode ramalan 2 tahun ke depan. Setelah proses selesai, lakukan penekanan pada *link button* “Ambil Nilai Terbaik” untuk mendapatkan panjang periode ramalan yang paling baik ditandai dengan MSE dan MAD yang paling kecil dari semua. Tombol Simpan digunakan untuk menyimpan data panjang periode ramalan terbaik dari jenis penyakit yang sedang diuji. Masing-masing penyakit memiliki panjang periode peramalan yang berbeda sehingga harus dilakukan proses yang sama beberapa kali untuk tiap penyakit.

-- Form Cari Pjg Ramalan Terbaik --

Data Penyakit

Kode Penyakit: P-0406

Nama Penyakit: CACAR AIR

Hasil Ramalan

08

Mulai **Simpan**

ID	PjgRamalan	DataAktual	NilaiRamala	MSE	MAD
P-001	1	5	7	3.46	1.86
P-002	2	20	7	170.78	14.79
P-003	3	19	9	133.45	12.3
P-004	4	9	8	89.28	8.52
P-005	5	11	10	67.06	6.55
P-006	6	16	10	60.79	6.43
P-007	7	6	11	55.47	6.26
P-008	8	24	10	75.44	7.36

Panjang Ramalan Terbaik = 1

Ambil Nilai Terbaik

Nilai MSE	3.46	Nilai Alpha	0.10
Nilai MAD	1.86	Nilai Beta	0.40
		Nilai Teta	0.10

Gambar 4.22 Form Pencarian Panjang Periode Ramalan Terbaik

Pada gambar 4.23 bisa dilihat proses perbandingan antara hasil peramalan konvensional dan proses peramalan dengan menggunakan konstanta hasil *neural network*. Setelah dipilih kode penyakit, maka akan diambil data pola NN terbaik dan data panjang periode ramalan terbaik yang bisa digunakan dalam proses peramalan. Pola data NN terbaik dan panjang periode ramalan terbaik didapatkan dari proses sebelumnya. Selain menggunakan hasil yang ada, terdapat juga pilihan jika ingin memasukkan pola data NN dan panjang periode ramalan secara manual. Hasil ramalan akan ditampilkan pada datagrid di bawah, di situ bisa dilihat perbandingan hasil dari ramalan tanpa bantuan NN dan dengan bantuan NN. Di bawah tiap datagrid terdapat nilai konstanta yang digunakan dalam proses ramalan. Setelah semua proses looping dilakukan, maka bisa ditekan tombol Simpan untuk menyimpan data perbandingan hasil ramal.

-- Form Perbandingan Hasil Ramalan --

Data Pendukung

Kode Penyakit	P-0406		
Nama Penyakit	CACAR AIR		
Panjang Ramalan	1	Panjang Musim	4
Jumlah Neuron Input	7	Jumlah Neuron Output	3
Jumlah Neuron Hidden	40		

Periode Awal	Dec 2005
Periode Akhir	Apr 2007

[OK](#)

-- Anjuran --

Panjang Ramalan Terbaik = **1** [Gunakan](#)
 Pola Terbaik = **7-40-3** [Gunakan](#)

----- Exponential Smoothing ----- Neural Network -----

ID_Exp	PjgRamalan	DataAktual	NilaiRama
ES-001	1	10	12

ID_NN	PjgRamalan	DataAktual	NilaiRama
NN-001	1	10	13

Nilai Alpha: 0.10
 Nilai Beta: 0.10
 Nilai Teta: 0.10

Nilai Alpha: 0.02
 Nilai Beta: 0.03
 Nilai Teta: 0.95

Gambar 4.23 Form Perbandingan Hasil Ramalan

Pada gambar 4.24 bisa dilihat form laporan hasil ramalan penyakit yang berpeluang mewabah pada periode selanjutnya. Selain nama penyakit, nilai hasil ramalan juga muncul yang menandakan ramalan jumlah penderita pada periode selanjutnya. Laporan ini dibatasi sebanyak 10 jumlah penyakit yang berpeluang besar mewabah pada periode mendatang. Dari laporan ini bisa diambil antisipasi untuk pencegahan terhadap masing-masing penyakit oleh pihak medis di puskesmas. Karena menggunakan *crystal report*, maka dari form ini bisa langsung dilakukan *print* hasil ramalan ini untuk pelaporan dengan cara menekan tombol Print Report pada bagian atas *toolbar*.

-- Form Report Hasil Ramalan --

Main Report

Penyakit Hasil Ramalan
Berpeluang Mewabah Periode Depan



Kode Penyakit	Nama Penyakit	Hasil Ramal
P-0103	DISENTRI	31
P-0406	CACAR AIR	16
P-0703	PENYAKIT KECACINGAN	3
P-1002	KATARAK	7
P-1401	PNEUMONIA	15
P-1601	PENYAKIT SALURAN KENCING	32
P-1701	KEGUGURAN	4
P-1901	KECELAKAAN & RUDA PAKSA	79
P-2001	PENYAKIT KULIT INFENSI	228
P-2002	PENYAKIT KULIT ALERGI	231

Current Page No.: 1 Total Page No.: 1 Zoom Factor: 100%

Gambar 4.24 Form Laporan Hasil Ramalan

Pada gambar 4.25 bisa dilihat form laporan rekап penyakit bulanan. Pada form ini laporan bisa dilihat secara seluruh atau berdasarkan periode bulan tertentu saja. Untuk menampilkan seluruh laporan tanpa filter bulan, bisa memilih *radio button* Semua. Sedangkan untuk filter berdasarkan bulan tertentu, pilih dulu *radio button* Periode Tertentu, lalu pilih bulan dan tahunnya. Selain itu, ada *link button* Kecuali 0 yang akan menghilangkan laporan jenis penyakit yang tidak ada penderitanya. Pada bagian kiri form terdapat juga kode rekап penyakit yang mewakili rekап periode bulan dan tahun tertentu. Karena menggunakan *crystal report*, maka dari form ini bisa langsung dilakukan *print* hasil ramalan ini untuk pelaporan dengan cara menekan tombol Print Report pada bagian atas *toolbar*.

Kode Penyakit	Nama Penyakit	Jumlah P.
P-0101	KOLERE	
P-0102	DIARE	
P-0103	DISENTRI	
P-0104	INFEKSI PADA USUS LAINNYA	
P-0201	TB PARU	
P-0202	TB SELAIN PARU	
P-0301	KUSTA I/T (MB)	
P-0302	KUSTA B/L (PB)	
P-0303	DIFTERIA	
P-0304	BATUK REJAN	
P-0305	TETANUS	
...		

Gambar 4.25 Form Laporan Rekap Penyakit Bulanan

4.3 Uji Coba dan evaluasi

Uji coba dan evaluasi bertujuan untuk memastikan bahwa aplikasi telah dibuat dengan benar sesuai dengan kebutuhan atau tujuan yang diharapkan. Kekurangan atau kelemahan aplikasi pada tahap ini akan dievaluasi sebelum diimplementasikan secara nyata. Proses pengujian menggunakan *Black Box Testing* dimana aplikasi akan diuji dengan melakukan berbagai percobaan untuk membuktikan bahwa aplikasi yang telah dibuat sudah sesuai dengan tujuan.

4.3.1 Uji Coba Fitur Dasar Sistem

Uji coba fitur dasar sistem ini bertujuan untuk mengecek apakah semua fitur yang ada dalam aplikasi telah berjalan sesuai dengan fungsinya. Berikut ini adalah uji coba fitur dasar sistem yang telah dilakukan:

1. Uji Coba Fitur Login

Proses login dilakukan dengan cara menginputkan username dan password.

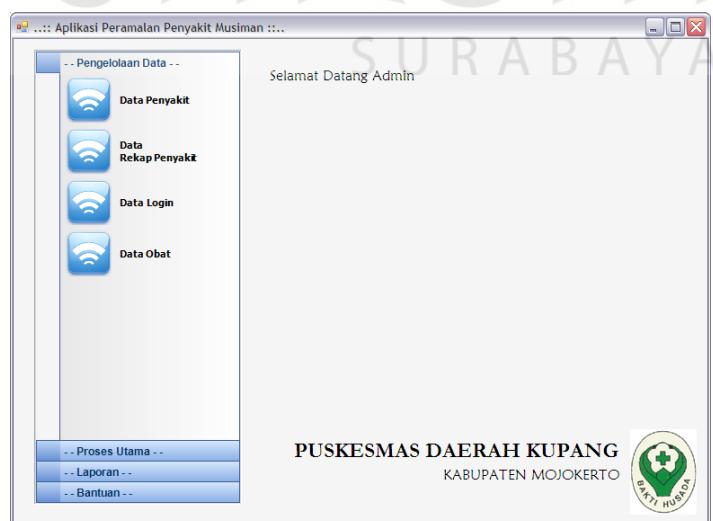
Dari username dan password ini akan diketahui status login, apakah sebagai admin atau operator.

Tabel 4.1 Tabel Login

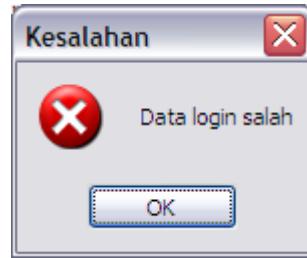
ID	Username	Password	Status
ADM-001	admin	Admin	Admin
OPR-001	operator	Operator	Operator

Tabel 4.2 Tabel Hasil Test Case Login

Test Case ID	Tujuan	Input	Output Diharapkan	Status
1	Deskripsi username dan password yang valid	Memasukkan data login username= <i>admin</i> dan password= <i>admin</i>	Form Login tertutup dan muncul form utama untuk admin	Sukses (Gambar 4.26)
2	Deskripsi username dan password non valid	Memasukkan data login username= <i>operator</i> dan password= <i>coba</i>	Muncul pesan "Data Login Salah"	Sukses (Gambar 4.27)



Gambar 4.26 Form Utama Admin Setelah Login



Gambar 4.27 Pesan Kesalahan Pengisian Data Login

2. Uji Coba Fitur Manipulasi Data Penyakit

Proses manipulasi data penyakit adalah proses tambah, ubah, dan hapus data.

Proses ini bertujuan untuk mengetahui apakah proses manipulasi data bisa dilakukan melalui aplikasi. Tabel 4.3 menunjukkan data contoh yang digunakan dan tabel 4.4 menunjukkan hasil uji coba manipulasi data penyakit yang telah dilakukan.

Tabel 4.3 Tabel Penyakit

Kode_Penyakit	Jenis_Penyakit	Nama_Penyakit
P-0103	PENYAKIT INFEKSI PADA USUS	DISENTRI
P-0306	PENYAKIT BAKTERI	PES

Tabel 4.4 Tabel Hasil Test Case Manipulasi Data Penyakit

Test Case ID	Tujuan	Input	Output Diharapkan	Status
3	Tambah data baru ke tabel penyakit	Memasukkan data penyakit: kode_penyakit=P-0103, jenis_penyakit=PENYAKIT INFEKSI PADA USUS, nama_penyakit=DISENTRI kemudian menekan tombol Simpan	Muncul pesan "Data Berhasil Masuk" dan data baru muncul pada data grid	Sukses (Gambar 4.28 sebelah kiri)

Tabel 4.4 Tabel Hasil Test Case Manipulasi Data Penyakit (lanjutan)

Test Case ID	Tujuan	Input	Output Diharapkan	Status
4	Ubah data dari tabel penyakit	Memasukkan kode_penyakit= <i>P-0306</i> , dari data yang ada, diubah menjadi: jenis_penyakit= <i>PENYAKIT INFEKSI PADA USUS BESAR</i> , nama_penyakit= <i>DISENTRI KELAS BERAT</i> kemudian menekan tombol Edit	Muncul pesan "Data Berhasil Diubah" dan data hasil edit muncul pada data grid	Sukses (Gambar 4.28 sebelah tengah)
5	Hapus data dari tabel penyakit	Memasukkan kode_penyakit= <i>P-0306</i> , dari data yang ada, kemudian menekan tombol Hapus	Muncul pesan "Data Berhasil Dihapus" dan data yang dihapus, tidak muncul pada data grid	Sukses (Gambar 4.28 sebelah kanan)
6	Menghindari data kode penyakit kosong pada tabel penyakit	Memasukkan data penyakit: kode_penyakit=(<i>kosong</i>), jenis_penyakit= <i>PENYAKIT INFEKSI PADA USUS</i> , nama_penyakit= <i>DISENTRI</i> kemudian menekan tombol Simpan	Muncul pesan "Kode Penyakit Tidak Boleh Kosong" dan data tidak tersimpan pada tabel penyakit	Sukses (Gambar 4.29 sebelah kiri)
7	Menghindari data jenis penyakit kosong pada tabel penyakit	Memasukkan data penyakit: kode_penyakit= <i>P-0103</i> , jenis_penyakit=(<i>kosong</i>), nama_penyakit= <i>DISENTRI</i> kemudian menekan tombol Simpan	Muncul pesan "Jenis Penyakit Tidak Boleh Kosong" dan data tidak tersimpan pada tabel penyakit	Sukses (Gambar 4.29 sebelah tengah)
8	Menghindari data nama penyakit kosong pada tabel penyakit	Memasukkan data penyakit: kode_penyakit= <i>P-0103</i> , jenis_penyakit= <i>PENYAKIT INFEKSI PADA USUS</i> , nama_penyakit=(<i>kosong</i>) kemudian menekan tombol Simpan	Muncul pesan "Nama Penyakit Tidak Boleh Kosong" dan data tidak tersimpan pada tabel penyakit	Sukses (Gambar 4.29 sebelah kanan)



Gambar 4.28 Pesan Berhasil Pengelolaan Data Penyakit



Gambar 4.29 Pesan Kesalahan Pengelolaan Data Penyakit

3. Uji Coba Fitur Manipulasi Data Obat

Proses manipulasi data obat adalah proses tambah, ubah, dan hapus data.

Proses ini bertujuan untuk mengetahui apakah proses manipulasi data bisa dilakukan melalui aplikasi. Setiap proses yang dilakukan di sini berpengaruh langsung terhadap tabel Obat di database. Pada tabel 4.5 bisa dilihat contoh data nyata yang digunakan sebagai contoh untuk proses manipulasi data. Sedangkan tabel 4.6 menunjukkan beberapa test case untuk menguji apakah proses manipulasi data obat telah sesuai dengan yang diharapkan. Hasil dari uji coba berupa pesan berhasil ditunjukkan pada gambar 4.30 dan pesan berupa kesalahan muncul pada gambar 4.31.

Tabel 4.5 Tabel Obat

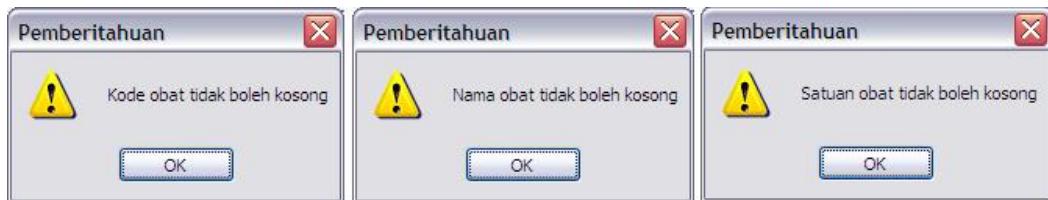
Kode_Obat	Nama_Obat	Satuan
O-0111	AMOKSILIN KAPSUL 250 MG	KAP
O-0124	AQUADEST STERIL	BTL

Tabel 4.6 Tabel Test Case Manipulasi Data Obat

Test Case ID	Tujuan	Input	Output Diharapkan	Status
9	Tambah data baru ke tabel obat	Memasukkan data obat: kode_obat=0-0111, nama_obat=AMOKSILIN KAPSUL 250 MG, satuan=KAP kemudian menekan tombol Simpan	Muncul pesan "Data Berhasil Masuk" dan data baru muncul pada data grid	Sukses (Gambar 4.30 sebelah kiri)
10	Ubah data dari tabel obat	Memasukkan kode_obat=0-0124, dari data yang ada, diubah menjadi: nama_obat=AQUA, satuan=GALON kemudian menekan tombol Edit	Muncul pesan "Data Berhasil Diubah" dan data hasil edit muncul pada data grid	Sukses (Gambar 4.30 sebelah tengah)
11	Hapus data dari tabel obat	Memasukkan kode_obat=0-0124, dari data yang ada, kemudian menekan tombol Hapus	Muncul pesan "Data Berhasil Dihapus" dan data yang dihapus, tidak muncul pada data grid	Sukses (Gambar 4.30 sebelah kanan)
12	Menghindari data kode obat kosong pada tabel obat	Memasukkan data obat: kode_obat=(kosong), nama_obat=AQUADEST STERIL, satuan=BTL kemudian menekan tombol Simpan	Muncul pesan "Kode Obat Tidak Boleh Kosong" dan data tidak tersimpan pada tabel obat	Sukses (Gambar 4.31 sebelah kiri)
13	Menghindari data nama obat kosong pada tabel obat	Memasukkan data obat: kode_obat=0-0124, nama_obat=(kosong), satuan=TAB kemudian menekan tombol Simpan	Muncul pesan "Nama Obat Tidak Boleh Kosong" dan data tidak tersimpan pada tabel obat	Sukses (Gambar 4.31 sebelah tengah)
14	Menghindari data satuan kosong pada tabel obat	Memasukkan data obat: kode_obat=0-0124 nama_obat=AQUADEST STERIL, satuan=(kosong) kemudian menekan tombol Simpan	Muncul pesan "Satuan Tidak Boleh Kosong" dan data tidak tersimpan pada tabel obat	Sukses (Gambar 4.31 sebelah kanan)



Gambar 4.30 Pesan Berhasil Pengelolaan Data Obat



Gambar 4.31 Pesan Kesalahan Pengelolaan Data Obat

4. Uji Coba Fitur Manipulasi Data Login

Proses manipulasi data login adalah proses tambah, ubah, dan hapus data. Proses ini bertujuan untuk mengetahui apakah proses manipulasi data bisa dilakukan melalui aplikasi. Setiap proses yang dilakukan di sini berpengaruh langsung terhadap tabel Login di database. Pada tabel 4.7 bisa dilihat contoh data nyata yang digunakan sebagai contoh untuk proses manipulasi data. Sedangkan tabel 4.8 menunjukkan beberapa test case untuk menguji apakah proses manipulasi data login telah sesuai dengan yang diharapkan. Hasil dari uji coba berupa pesan berhasil ditunjukkan pada gambar 4.32 dan pesan berupa kesalahan muncul pada gambar 4.33.

Tabel 4.7 Tabel Login

ID	Username	Password	Status
ADM-001	admin	admin	Admin
OPR-001	operator	operator	Operator

Tabel 4.8 Tabel Hasil Test Case Manipulasi Data Login

Test Case ID	Tujuan	Input	Output Diharapkan	Status
15	Tambah data baru ke tabel login	Memasukkan data login: username= <i>admin</i> , password= <i>admin</i> , ulangi password= <i>admin</i> , status= <i>Admin</i> kemudian menekan tombol Simpan	Muncul pesan "Data Berhasil Masuk" dan data baru muncul pada data grid	Sukses (Gambar 4.32 sebelah kiri)
16	Ubah data dari tabel login	Memilih data dari gridview dengan ID= <i>OPR-001</i> , dari data yang ada, diubah menjadi: username= <i>opera</i> , password= <i>opera</i> kemudian menekan tombol Edit	Muncul pesan "Data Berhasil Diubah" dan data hasil edit muncul pada data grid	Sukses (Gambar 4.32 sebelah tengah)
17	Hapus data dari tabel login	Memilih data dari gridview dengan ID= <i>OPR-001</i> , dari data yang ada, kemudian menekan tombol Hapus	Muncul pesan "Data Berhasil Dihapus" dan data yang dihapus, tidak muncul pada data grid	Sukses (Gambar 4.32 sebelah kanan)
18	Menghindari data username kosong pada tabel login	Memasukkan data login: Username= <i>(kosong)</i> , password= <i>operator</i> , ulangi password= <i>operator</i> , status= <i>Operator</i> kemudian menekan tombol Simpan	Muncul pesan "Username Tidak Boleh Kosong" dan data tidak tersimpan pada tabel login	Sukses (Gambar 4.33 sebelah kiri)
19	Menghindari data password kosong pada tabel login	Memasukkan data login: Username= <i>operator</i> , password= <i>(kosong)</i> , ulangi password= <i>operator</i> , status= <i>Operator</i> kemudian menekan tombol Simpan	Muncul pesan "Password Tidak Boleh Kosong" dan data tidak tersimpan pada tabel login	Sukses (Gambar 4.33 sebelah tengah)

Tabel 4.8 Tabel Hasil Test Case Manipulasi Data Login (lanjutan)

Test Case ID	Tujuan	Input	Output Diharapkan	Status
20	Menghindari kesalahan pengisian password	Memasukkan data login: Username= <i>operator</i> , password= <i>oprtr</i> , ulangi password= <i>operator</i> , status= <i>Operator</i> kemudian menekan tombol Simpan	Muncul pesan "Password Tidak Sama" dan data tidak tersimpan pada tabel login	Sukses (Gambar 4.33 sebelah kanan)



Gambar 4.32 Pesan Berhasil Pengelolaan Data Login



Gambar 4.33 Pesan Kesalahan Pengelolaan Data Login

5. Uji Coba Fitur Pengambilan Data Berpola Musiman

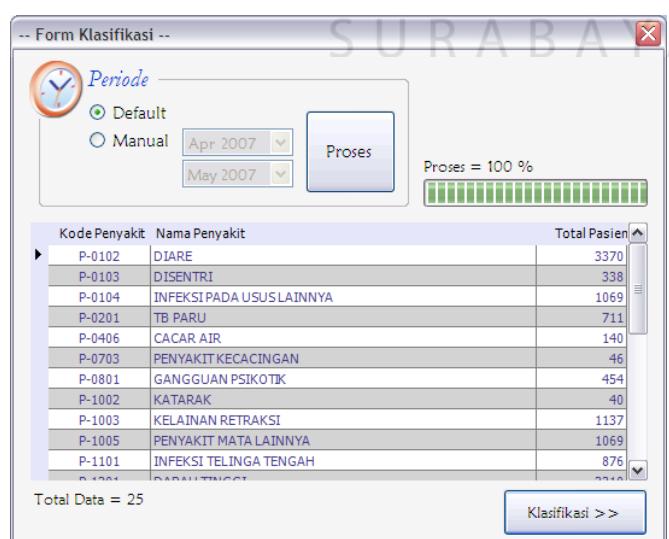
Proses ini mengambil beberapa data penyakit yang mempunyai kecenderungan berpola musiman dari database. Penyakit yang mempunyai status klasifikasi 1 yang diambil dalam proses ini karena tidak semua penyakit memiliki pola musiman. Setelah itu hanya 10 jenis penyakit yang diambil dalam proses ini untuk digunakan dalam aplikasi peramalan. Pada tabel 4.9 bisa dilihat contoh data penyakit yang memiliki pola musiman.

Tabel 4.9 Tabel Hasil_Klasifikasi

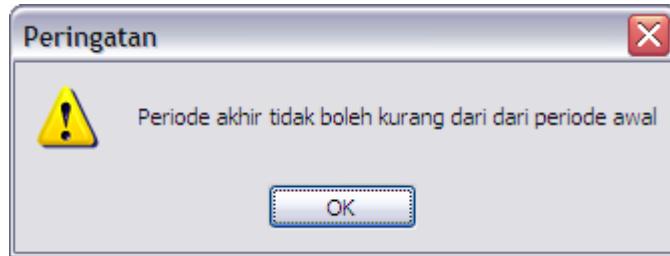
Kode_Penyakit	Status_Klasifikasi
P-0103	1
P-0306	0

Tabel 4.10 Tabel Hasil Test Case Pengambilan Data

Test Case ID	Tujuan	Input	Output Diharapkan	Status
21	Mengambil data penyakit	Memilih radio button Default atau Manual, lalu menekan tombol Proses	Progress Bar bergerak sampai 100%, lalu data keluar pada datagrid	Sukses (Gambar 4.34)
22	Mengecek tanggal awal tidak boleh lebih dari tanggal akhir	Memilih radio button Manual, lalu memilih periode awal=JAN 2006 dan periode akhir=DES 2005 kemudian menekan tombol Proses	Muncul pesan "Periode Akhir Tidak Boleh Kurang Dari Periode Awal" dan data tidak muncul pada datagrid	Sukses (Gambar 4.35)
23	Memunculkan 10 jenis penyakit hasil klasifikasi	Menekan tombol Klasifikasi	Form Klasifikasi tertutup dan muncul Form Hasil Klasifikasi beserta 10 jenis penyakit pada datagrid	Sukses (Gambar 4.36)



Gambar 4.34 Uji Coba Fitur Form Klasifikasi



Gambar 4.35 Pesan Kesalahan Pengisian Periode



Hasil Klasifikasi		
Kode Penyakit	Nama Penyakit	Total Pasien
P-0703	PENYAKIT KECACINGAN	46
P-1706	HYPEREMESIS	144
P-1002	KATARAK	40
P-1601	PENYAKIT SALURAN KENCING	203
P-0406	CACAR AIR	140
P-1901	KECELAKAAN & RUDA PAKSA	813
P-1701	KEGUGURAN	53
P-0103	DISENTRI	338
P-2001	PENYAKIT KULIT INFENSI	2308
P-2002	PENYAKIT KULIT ALERGI	2673
*		

At the bottom are two buttons: a blue 'Simpan' (Save) button with a floppy disk icon and a blue 'Batal' (Cancel) button with a red 'X' icon.

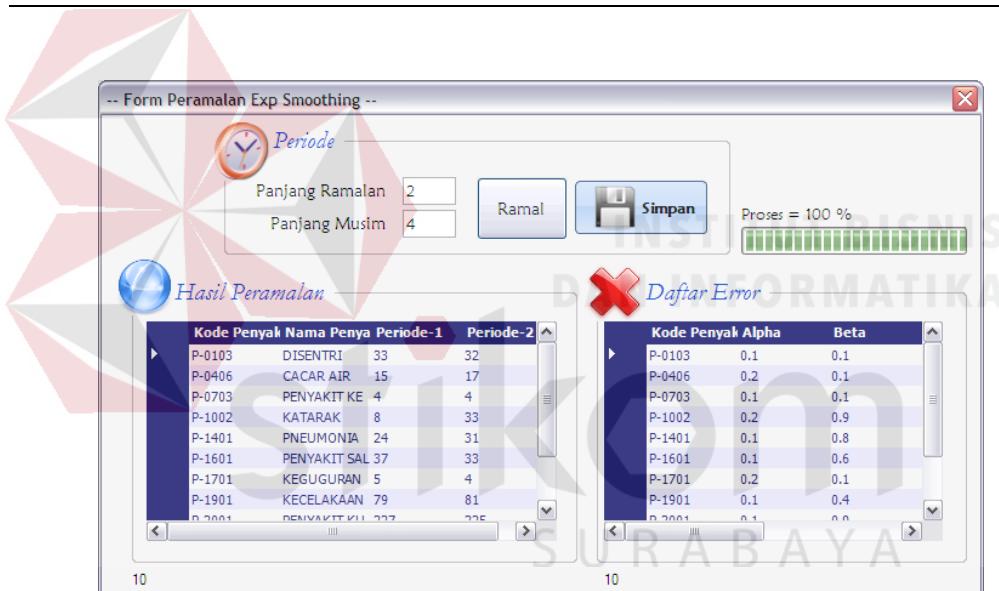
Gambar 4.36 Uji Coba Fitur Form Hasil Klasifikasi

6. Uji Coba Fitur Peramalan *Exponential Smoothing* oleh Winter

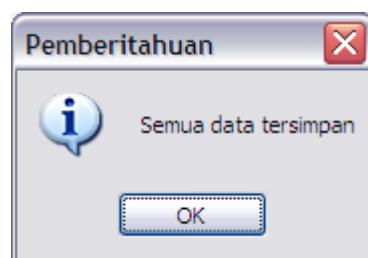
Proses ini melakukan peramalan dengan metode *exponential smoothing* oleh Winter. Setelah memasukkan data panjang ramalan dan panjang musim serta memanfaatkan data penyakit hasil klasifikasi sebelumnya, maka proses peramalan bisa dilakukan. Tabel 4.11 menunjukkan hasil test case proses peramalan ini, sedangkan gambar 4.37 dan gambar 4.38 adalah tampilan dari aplikasi yang menunjukkan bahwa test case telah berhasil dilakukan.

Tabel 4.11 Tabel Hasil Test Case Peramalan

Test Case ID	Tujuan	Input	Output Diharapkan	Status
24	Menghasilkan hasil ramalan untuk periode depan	Memasukkan panjang ramalan=1 dan panjang musim=4 lalu menekan tombol Ramal	Progress Bar bergerak sampai 100%, lalu data hasil ramalan keluar pada datagrid Hasil Ramalan, sedangkan list error keluar pada datagrid List Error	Sukses (Gambar 4.37)
25	Mengecek data tersimpan sebagai data latih atau tidak	Setelah proses peramalan, menekan tombol Simpan	Muncul pesan "Semua Data Telah Tersimpan"	Sukses (Gambar 4.38)



Gambar 4.37 Uji Coba Fitur Form Peramalan



Gambar 4.38 Pesan Pemberitahuan Berhasil Menyimpan Data

7. Uji Coba Fitur Pelatihan Data *Neural Network*

Proses ini melakukan proses pelatihan pola yang akan digunakan pada proses *neural network*. Pelatihan akan menjadi maksimal jika MSE telah mencapai 0 atau mendekati 0. Selain itu, proses looping pelatihan data bisa berhenti jika user menganggap bahwa error yang muncul sudah dapat diterima. Tabel 4.12 menunjukkan hasil test case pelatihan pola neural network ini.

Tabel 4.12 Tabel Hasil Test Case Pelatihan

Test Case ID	Tujuan	Input	Output Diharapkan	Status
26	Memilih data latih dengan panjang inputan data time series	Memilih nilai panjang neuron input=5 lalu menekan tombol OK	Semua data latih yang mempunyai panjang data time series sepanjang 5 periode muncul pada datagrid	Sukses (Gambar 4.39)
27	Menyimpan data latih pada XML	Setelah memilih data time series, menekan tombol Simpan	Muncul pesan "Data Latih Telah Tersimpan"	Sukses (Gambar 4.40)
28	Melakukan proses pelatihan	Menekan tombol Mulai	Tombol Mulai berubah menjadi tombol Berhenti dan proses pelatihan berjalan ditunjukkan dengan textbox Iterasi yang melakukan counter	Sukses (Gambar 4.41)
29	Memberhentikan proses pelatihan	Menekan tombol Berhenti	Counter pada textbox iterasi dan proses pelatihan berhenti lalu tombol Berhenti berubah menjadi tombol Mulai	Sukses (Gambar 4.42)
30	Menyimpan data hasil pelatihan	Setelah proses pelatihan, menekan tombol Simpan	Keluar grafik hasil pelatihan dan data tersimpan	Sukses (Gambar 4.43)

-- Form Data Latih --

Filter Neuron Input

Panjang Neuron Input: 5

OK

Data Latih

Kode_Latih	Jumlah_DtA	Dt_Aktual01	Dt_Aktual02	Dt_Aktual03	Dt_Aktual04	Dt_Aktual05
DL05-0001	5	0.004	0.006	0.004	0.017	0.005
DL05-0002	5	0.007	0.008	0.003	0.002	0.005
DL05-0003	5	0.021	0.034	0.03	0.017	0.025
DL05-0004	5	0.036	0.067	0.034	0.028	0.021
DL05-0005	5	0.173	0.155	0.18	0.16	0.188
DL05-0006	5	0.023	0.007	0.01	0.012	0.015
DL05-0007	5	0.002	0.013	0.005	0.009	0.002
DL05-0008	5	0.009	0.012	0.018	0.016	0
DL05-0009	5	0.001	0.001	0.007	0.00	0.001

Simpan Batal

Gambar 4.39 Uji Coba Fitur Form Data Latih



Gambar 4.40 Pesan Data Latih Telah Tersimpan

-- Form Latih NN --

Pelatihan

Iterasi ke-: 527

Tambah Satu Iterasi

MSE: 0.17768

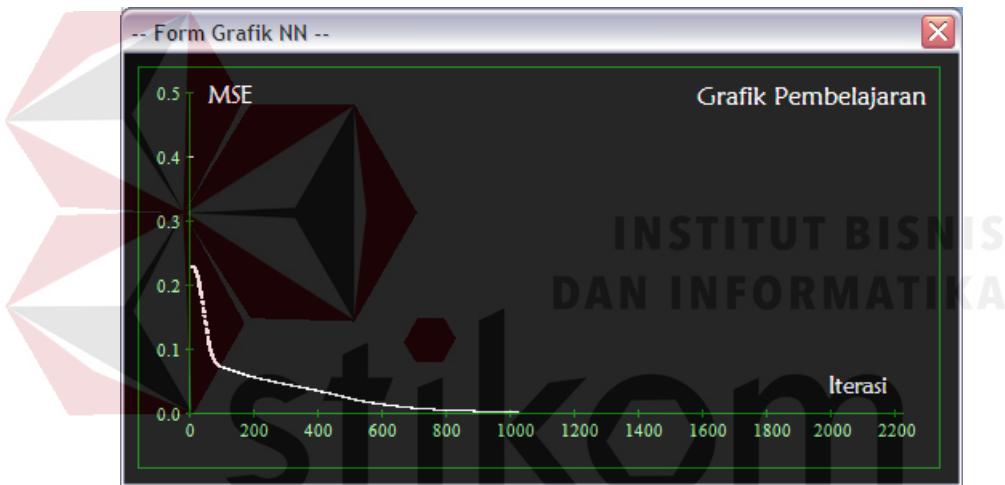
Berhenti Simpan

Lihat Grafik >>

Gambar 4.41 Uji Coba Fitur Form Pelatihan



Gambar 4.42 Uji Coba Fitur Memberhentikan Pelatihan



Gambar 4.43 Uji Coba Fitur Grafik Pelatihan

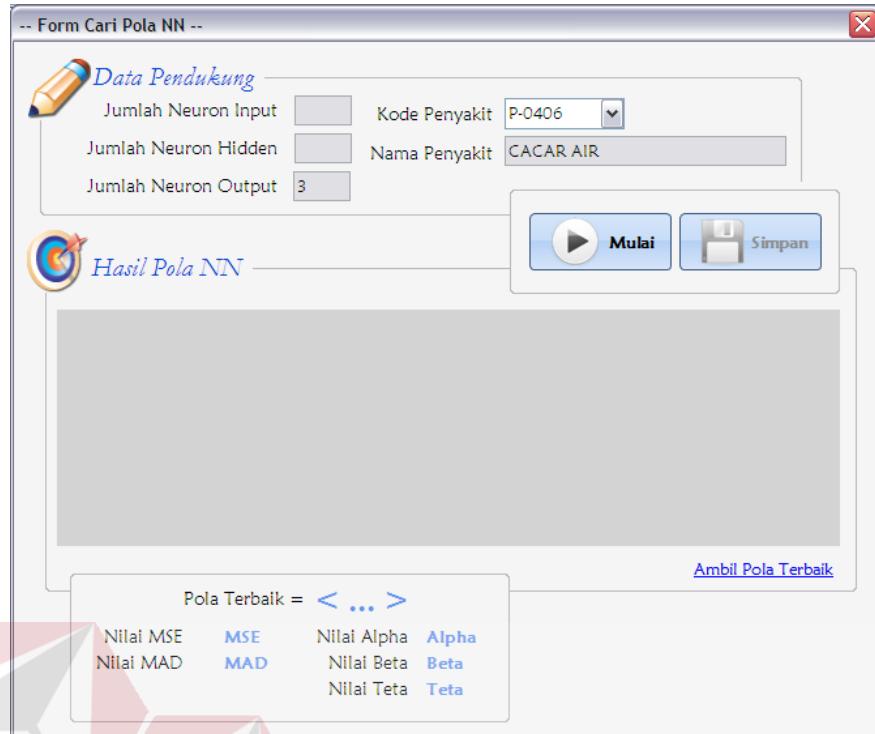
8. Uji Coba Fitur Pencarian Pola NN

Proses ini melakukan proses pencarian pola NN terbaik untuk tiap penyakit lalu menyimpan hasilnya pada database. Karena tiap penyakit mempunyai pola musiman yang berbeda, jadi proses ini harus dilakukan untuk masing-masing penyakit. Tabel 4.13 menunjukkan hasil test case proses pencarian pola neural network terbaik.

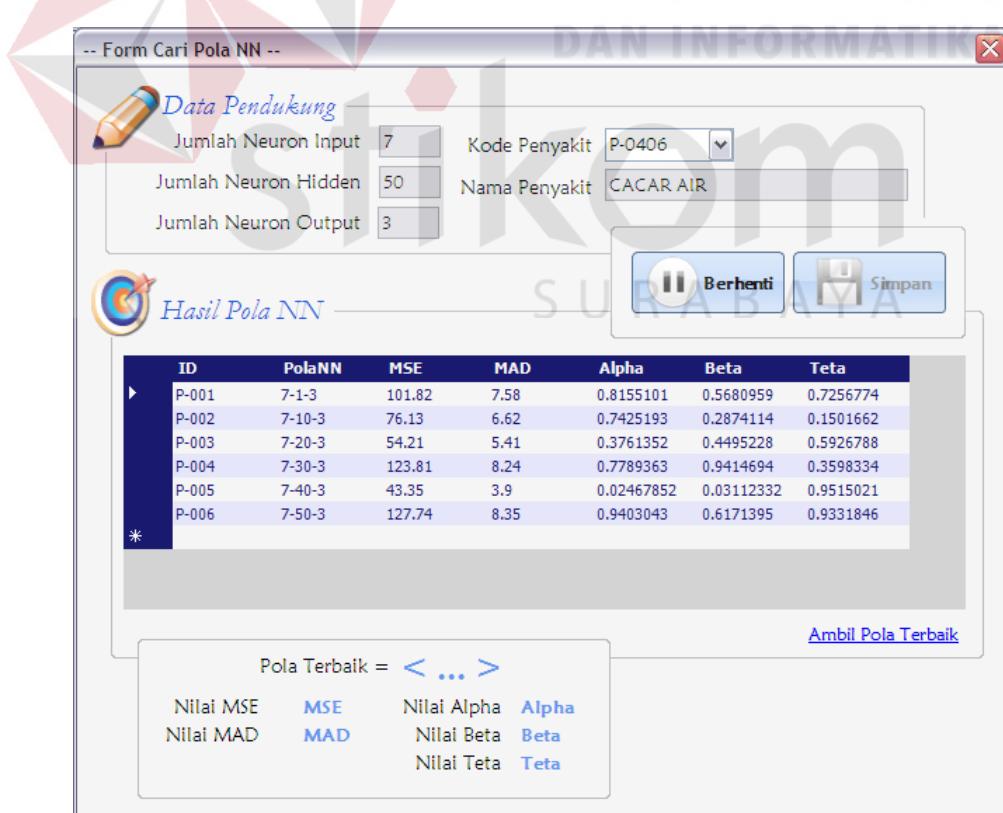
Tabel 4.13 Tabel Hasil Test Case Pencarian Pola NN Terbaik

Test Case ID	Tujuan	Input	Output Diharapkan	Status
31	Menampilkan nama penyakit	Memilih kode penyakit=P-0406 pada combo box kode penyakit	Nama penyakit=Cacar Air keluar pada textbox nama penyakit	Sukses (Gambar 4.44)
32	Menjalankan looping mencari pola NN	Memilih kode penyakit pada combo box kode penyakit lalu menekan tombol Mulai	Tombol Mulai berubah menjadi tombol Berhenti dan proses looping berjalan ditunjukkan dengan bertambahnya data pada datagrid	Sukses (Gambar 4.45)
33	Memberhentikan proses	Menekan tombol Berhenti	Data pada datagrid tidak bertambah lagi, counter pada textbox jumlah neuron input dan hidden juga berhenti lalu tombol Berhenti berubah menjadi tombol Mulai	Sukses (Gambar 4.46)
34	Mengambil pola terbaik	Setelah proses looping selesai, menekan link button Ambil Pola Terbaik	Pola terbaik muncul pada bagian bawah ditandai dengan MSE dan MAD terkecil, tombol Simpan menjadi aktif	Sukses (Gambar 4.47)
35	Menyimpan pola terbaik	Setelah mengambil data pola terbaik, menekan tombol Simpan	Keluar pesan "Data Telah Tersimpan"	Sukses (Gambar 4.48)

Gambar 4.44 sampai dengan gambar 4.48 menunjukkan tampilan form hasil test case dari proses pencarian pola neural network yang terbaik untuk tiap penyakit.



Gambar 4.44 Uji Coba Fitur Pola NN Terbaik (1)



Gambar 4.45 Uji Coba Fitur Pola NN Terbaik (2)

-- Form Cari Pola NN --

Data Pendukung

Jumlah Neuron Input	9	Kode Penyakit	P-0406
Jumlah Neuron Hidden	60	Nama Penyakit	CACAR AIR
Jumlah Neuron Output	3		

Hasil Pola NN

ID	PolaNN	MSE	MAD	Alpha	Beta	Teta
P-001	7-1-3	101.82	7.58	0.8155101	0.5680959	0.7256774
P-002	7-10-3	76.13	6.62	0.7425193	0.2874114	0.1501662
P-003	7-20-3	54.21	5.41	0.3761352	0.4495228	0.5926788
P-004	7-30-3	123.81	8.24	0.7789363	0.9414694	0.3598334
P-005	7-40-3	43.35	3.9	0.02467852	0.03112332	0.9515021
P-006	7-50-3	127.74	8.35	0.9403043	0.6171395	0.9331846
P-007	7-60-3	45.11	4.43	0.1479926	0.6439975	0.9567043
P-008	7-70-3	92.92	7.27	0.6091856	0.9505332	0.7119362
P-009	7-80-3	82.28	6.87	0.995628	0.04291296	0.7732476
P-010	7-80-3	42.40	4	0.000010000	0.000050000	0.001000000

Ambil Pola Terbaik

Pola Terbaik = < ... >

Nilai MSE	MSE	Nilai Alpha	Alpha
Nilai MAD	MAD	Nilai Beta	Beta
		Nilai Teta	Teta

Gambar 4.46 Uji Coba Fitur Pola NN Terbaik (3)

-- Form Cari Pola NN --

Data Pendukung

Jumlah Neuron Input	9	Kode Penyakit	P-0406
Jumlah Neuron Hidden	60	Nama Penyakit	CACAR AIR
Jumlah Neuron Output	3		

Hasil Pola NN

ID	PolaNN	MSE	MAD	Alpha	Beta	Teta
P-001	7-1-3	101.82	7.58	0.8155101	0.5680959	0.7256774
P-002	7-10-3	76.13	6.62	0.7425193	0.2874114	0.1501662
P-003	7-20-3	54.21	5.41	0.3761352	0.4495228	0.5926788
P-004	7-30-3	123.81	8.24	0.7789363	0.9414694	0.3598334
P-005	7-40-3	43.35	3.9	0.02467852	0.03112332	0.9515021
P-006	7-50-3	127.74	8.35	0.9403043	0.6171395	0.9331846
P-007	7-60-3	45.11	4.43	0.1479926	0.6439975	0.9567043
P-008	7-70-3	92.92	7.27	0.6091856	0.9505332	0.7119362
P-009	7-80-3	82.28	6.87	0.995628	0.04291296	0.7732476
P-010	7-80-3	42.40	4	0.000010000	0.000050000	0.001000000

Ambil Pola Terbaik

Pola Terbaik = **7-40-3**

Nilai MSE	43.35	Nilai Alpha	0.02
Nilai MAD	3.90	Nilai Beta	0.03
		Nilai Teta	0.95

Gambar 4.47 Uji Coba Fitur Pola NN Terbaik (4)



Gambar 4.48 Pesan Data Pola NN Tersimpan

9. Uji Coba Fitur Pencarian Panjang Ramalan

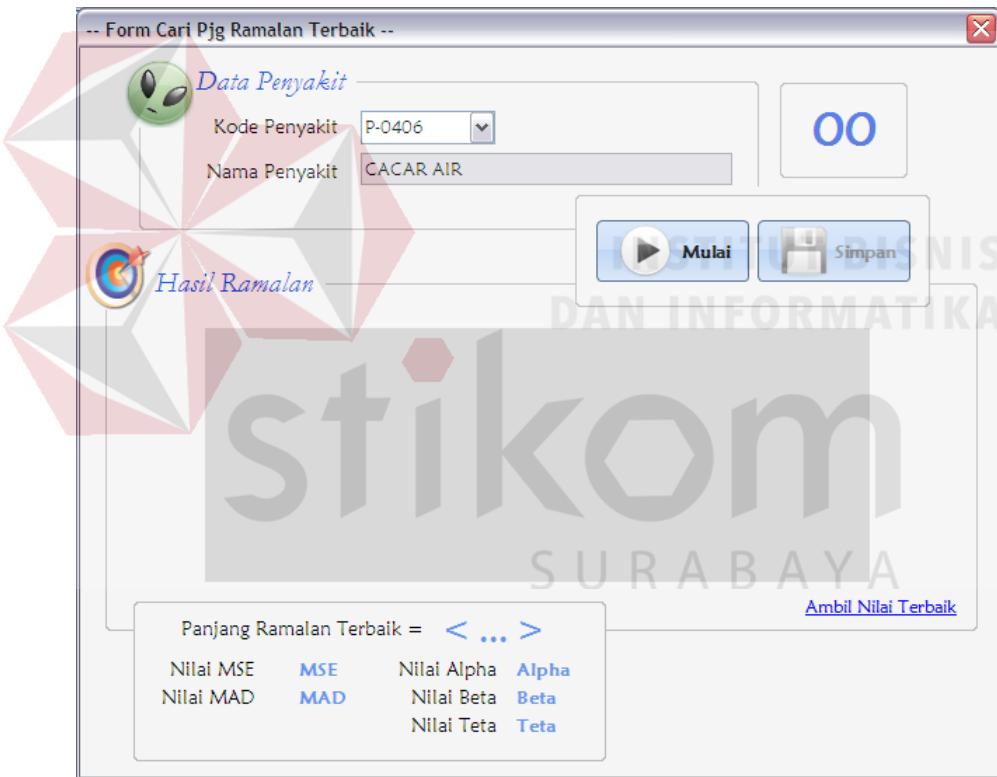
Proses ini melakukan proses pencarian panjang ramalan terbaik untuk tiap penyakit lalu menyimpan hasilnya pada database. Karena setiap penyakit mempunyai pola data yang berbeda, jadi proses ini harus dilakukan berulang kali sesuai jumlah jenis penyakit yang digunakan. Tabel 4.14 menunjukkan hasil test case pencarian panjang ramalan terbaik untuk tiap penyakit.

Tabel 4.14 Tabel Hasil Test Case Pencarian Panjang Ramalan Terbaik

Test Case ID	Tujuan	Input	Output Diharapkan	Status
36	Menampilkan nama penyakit	Memilih kode penyakit=P-0406 pada combo box kode penyakit	Nama penyakit=Cacar Air keluar pada textbox nama penyakit	Sukses (Gambar 4.49)
37	Menjalankan looping mencari panjang ramalan	Memilih kode penyakit pada combo box kode penyakit lalu menekan tombol Mulai	Tombol Mulai berubah menjadi tombol Berhenti dan proses looping berjalan ditunjukkan dengan bertambahnya data pada datagrid	Sukses (Gambar 4.50)
38	Memberhentikan proses	Menekan tombol Berhenti	Data pada datagrid tidak bertambah lagi, counter angka juga berhenti lalu tombol Berhenti berubah menjadi tombol Mulai	Sukses (Gambar 4.51)

Tabel 4.14 Tabel Hasil Test Case Pencarian Panjang Ramalan Terbaik (lanjutan)

Test Case ID	Tujuan	Input	Output Diharapkan	Status
39	Mengambil panjang ramalan terbaik	Setelah proses looping selesai, menekan link button Ambil Nilai Terbaik	Nilai terbaik muncul pada bagian bawah ditandai dengan MSE dan MAD terkecil, tombol Simpan menjadi aktif	Sukses (Gambar 4.52)
40	Menyimpan panjang ramalan terbaik	Setelah mengambil data panjang ramalan terbaik, menekan tombol Simpan	Keluar pesan "Data Telah Tersimpan"	Sukses (Gambar 4.53)



Gambar 4.49 Uji Coba Fitur Pencarian Panjang Ramalan Terbaik (1)

Gambar 4.49 sampai dengan gambar 4.53 menunjukkan tampilan form hasil test case dari proses pencarian panjang periode ramalan yang terbaik untuk tiap penyakit.

-- Form Cari Pjg Ramalan Terbaik --

Data Penyakit

Kode Penyakit: P-0406
Nama Penyakit: CACAR AIR

Hasil Ramalan

05

ID	PjgRamalan	DataAktual	NilaiRamala	MSE	MAD
P-001	1	5	7	3.46	1.86
P-002	2	20	7	170.78	14.79
P-003	3	19	9	133.45	12.3
P-004	4	9	8	89.28	8.52
P-005	5	11	10	67.06	6.55
*					

Panjang Ramalan Terbaik = < ... >

[Ambil Nilai Terbaik](#)

Nilai MSE	MSE	Nilai Alpha	Alpha
Nilai MAD	MAD	Nilai Beta	Beta
		Nilai Teta	Teta

Berhenti **Simpan**

Gambar 4.50 Uji Coba Fitur Pencarian Panjang Ramalan Terbaik (2)

-- Form Cari Pjg Ramalan Terbaik --

Data Penyakit

Kode Penyakit: P-0406
Nama Penyakit: CACAR AIR

Hasil Ramalan

06

ID	PjgRamalan	DataAktual	NilaiRamala	MSE	MAD
P-001	1	5	7	3.46	1.86
P-002	2	20	7	170.78	14.79
P-003	3	19	9	133.45	12.3
P-004	4	9	8	89.28	8.52
P-005	5	11	10	67.06	6.55
P-006	6	16	10	60.79	6.43
*					

Panjang Ramalan Terbaik = < ... >

[Ambil Nilai Terbaik](#)

Nilai MSE	MSE	Nilai Alpha	Alpha
Nilai MAD	MAD	Nilai Beta	Beta
		Nilai Teta	Teta

Mulai **Simpan**

Gambar 4.51 Uji Coba Fitur Pencarian Panjang Ramalan Terbaik (3)

-- Form Cari Pjg Ramalan Terbaik --

Data Penyakit

Kode Penyakit: P-0406
Nama Penyakit: CACAR AIR

Hasil Ramalan

08

Mulai **Simpan**

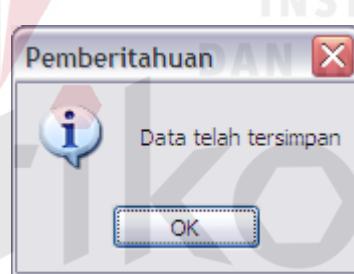
ID	PjgRamalan	DataAktual	NilaiRamala	MSE	MAD
P-001	1	5	7	3.46	1.86
P-002	2	20	7	170.78	14.79
P-003	3	19	9	133.45	12.3
P-004	4	9	8	89.28	8.52
P-005	5	11	10	67.06	6.55
P-006	6	16	10	60.79	6.43
P-007	7	6	11	55.47	6.26
P-008	8	24	10	75.44	7.36

Panjang Ramalan Terbaik = 1

Ambil Nilai Terbaik

Nilai MSE: 3.46
Nilai MAD: 1.86
Nilai Alpha: 0.10
Nilai Beta: 0.40
Nilai Teta: 0.10

Gambar 4.52 Uji Coba Fitur Pencarian Panjang Ramalan Terbaik (4)



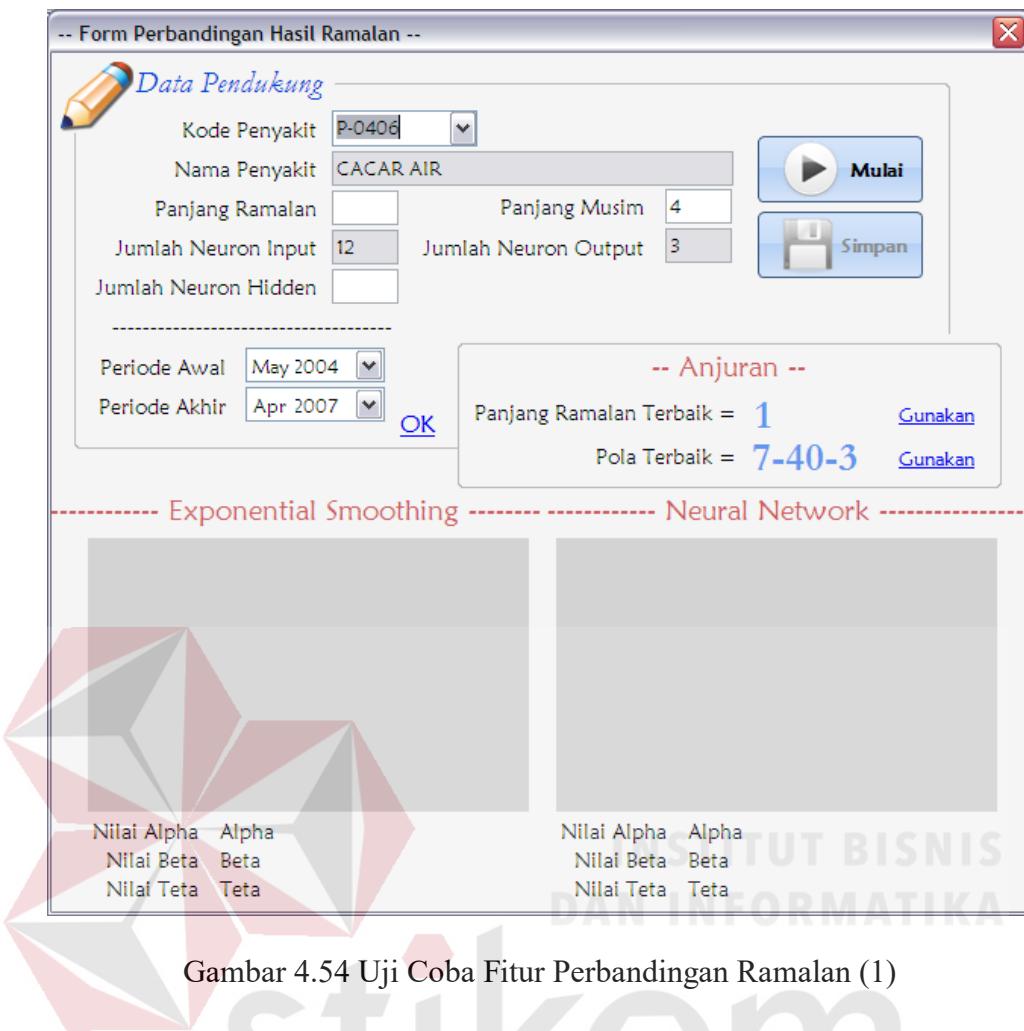
Gambar 4.53 Pesan Data Panjang Ramalan Telah Tersimpan

10. Uji Coba Fitur Perbandingan Hasil Ramalan

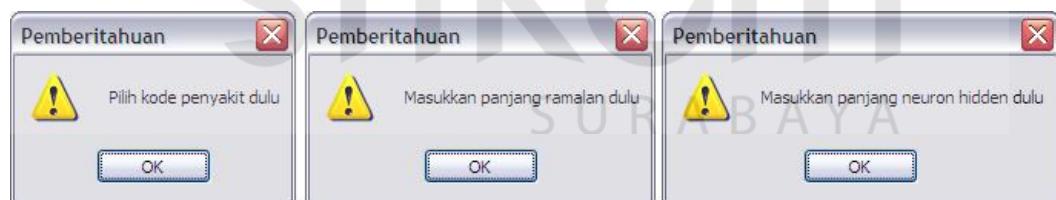
Proses ini melakukan proses perbandingan hasil ramalan exponential smooting oleh Winter konvensional tanpa memanfaatkan neural network dan peramalan dengan exponential smooting oleh Winter yang menggunakan bantuan neural network untuk penentuan konstanta.

Tabel 4.15 Tabel Hasil Test Case Perbandingan Hasil Ramalan

Test Case ID	Tujuan	Input	Output Diharapkan	Status
41	Menampilkan nama penyakit	Memilih kode penyakit=P-0406 pada combo box kode penyakit	Nama penyakit=Cacar Air keluar pada textbox nama penyakit	Sukses (Gambar 4.54)
42	Mencegah inputan kode penyakit kosong	Menekan tombol Mulai dengan inputan lengkap kecuali kode penyakit yang sengaja tidak dipilih	Keluar pesan "Pilih Kode Penyakit Dulu"	Sukses (Gambar 4.55 bagian kiri)
43	Mencegah inputan kosong pada panjang ramalan	Menekan tombol Mulai dengan inputan lengkap kecuali panjang ramalan yang sengaja tidak diisi	Keluar pesan "Masukkan Panjang Ramalan Dulu"	Sukses (Gambar 4.55 bagian tengah)
44	Mencegah inputan kosong pada neuron hidden	Menekan tombol Mulai dengan inputan lengkap kecuali neuron hiddeng yang sengaja tidak diisi	Keluar pesan "Masukkan Panjang Neuron Hidden Dulu"	Sukses (Gambar 4.55 bagian kanan)
45	Menggunakan anjuran panjang ramalan	Menekan link button Gunakan di sebelah anjuran panjang ramalan terbaik	Nilai panjang ramalan terbaik muncul pada textbox Panjang Ramalan	Sukses (Gambar 4.56)
46	Menggunakan anjuran pola NN	Menekan link button Gunakan di sebelah anjuran pola NN terbaik	Nilai pola NN terbaik muncul pada textbox Jumlah Neuron Input, Jumlah Neuron Hidden, dan Jumlah Neuron Output	Sukses (Gambar 4.57)
47	Menjalankan looping untuk melakukan proses peramalan	Setelah semua inputan lengkap, menekan tombol Mulai	Tombol Mulai berubah menjadi tombol Berhenti dan proses looping berjalan ditunjukkan dengan bertambahnya data pada datagrid	Sukses (Gambar 4.58)



Gambar 4.54 Uji Coba Fitur Perbandingan Ramalan (1)



Gambar 4.55 Pesan Kesalahan Proses Perbandingan

Gambar 4.54 sampai dengan gambar 4.58 menunjukkan tampilan form hasil test case dari proses perbandingan hasil ramalan antara ramalan tanpa bantuan neural network dan ramalan dengan bantuan neural network untuk tiap penyakit.

-- Form Perbandingan Hasil Ramalan --

Data Pendukung

Kode Penyakit: P-0406
 Nama Penyakit: CACAR AIR
 Panjang Ramalan: 1
 Panjang Musim: 4
 Jumlah Neuron Input: 12
 Jumlah Neuron Output: 3
 Jumlah Neuron Hidden:
 Periode Awal: May 2004
 Periode Akhir: Apr 2007

-- Anjuran --
 Panjang Ramalan Terbaik = **1** [Gunakan](#)
 Pola Terbaik = **7-40-3** [Gunakan](#)

Exponential Smoothing ----- **Neural Network**

Nilai Alpha Alpha
 Nilai Beta Beta
 Nilai Teta Teta

Nilai Alpha Alpha
 Nilai Beta Beta
 Nilai Teta Teta

Gambar 4.56 Uji Coba Fitur Perbandingan Ramalan (2)

-- Form Perbandingan Hasil Ramalan --

Data Pendukung

Kode Penyakit: P-0406
 Nama Penyakit: CACAR AIR
 Panjang Ramalan: 1
 Panjang Musim: 4
 Jumlah Neuron Input: 7
 Jumlah Neuron Output: 3
 Jumlah Neuron Hidden: 40
 Periode Awal: Dec 2005
 Periode Akhir: Apr 2007

-- Anjuran --
 Panjang Ramalan Terbaik = **1** [Gunakan](#)
 Pola Terbaik = **7-40-3** [Gunakan](#)

Exponential Smoothing ----- **Neural Network**

Nilai Alpha Alpha
 Nilai Beta Beta
 Nilai Teta Teta

Nilai Alpha Alpha
 Nilai Beta Beta
 Nilai Teta Teta

Gambar 4.57 Uji Coba Fitur Perbandingan Ramalan (3)

-- Form Perbandingan Hasil Ramalan --

Data Pendukung

Kode Penyakit: P-0406
 Nama Penyakit: CACAR AIR
 Panjang Ramalan: 1
 Panjang Musim: 4
 Jumlah Neuron Input: 7
 Jumlah Neuron Output: 3
 Jumlah Neuron Hidden: 40

Mulai
 Simpan

Periode Awal: Dec 2005
 Periode Akhir: Apr 2007
 OK

-- Anjuran --
 Panjang Ramalan Terbaik = 1
 Gunakan
 Pola Terbaik = 7-40-3
 Gunakan

----- Exponential Smoothing ----- Neural Network -----

DataAktual	NilaiRamalan	MSE	MAD
10	12	4.51	2.12
*			

DataAktual	NilaiRamalan	MSE	MAD
10	13	11.02	3.32
*			

Nilai Alpha: 0.10
 Nilai Beta: 0.10
 Nilai Teta: 0.10

Nilai Alpha: 0.02
 Nilai Beta: 0.03
 Nilai Teta: 0.95

Gambar 4.58 Uji Coba Fitur Perbandingan Ramalan (4)

11. Uji Coba Fitur Laporan

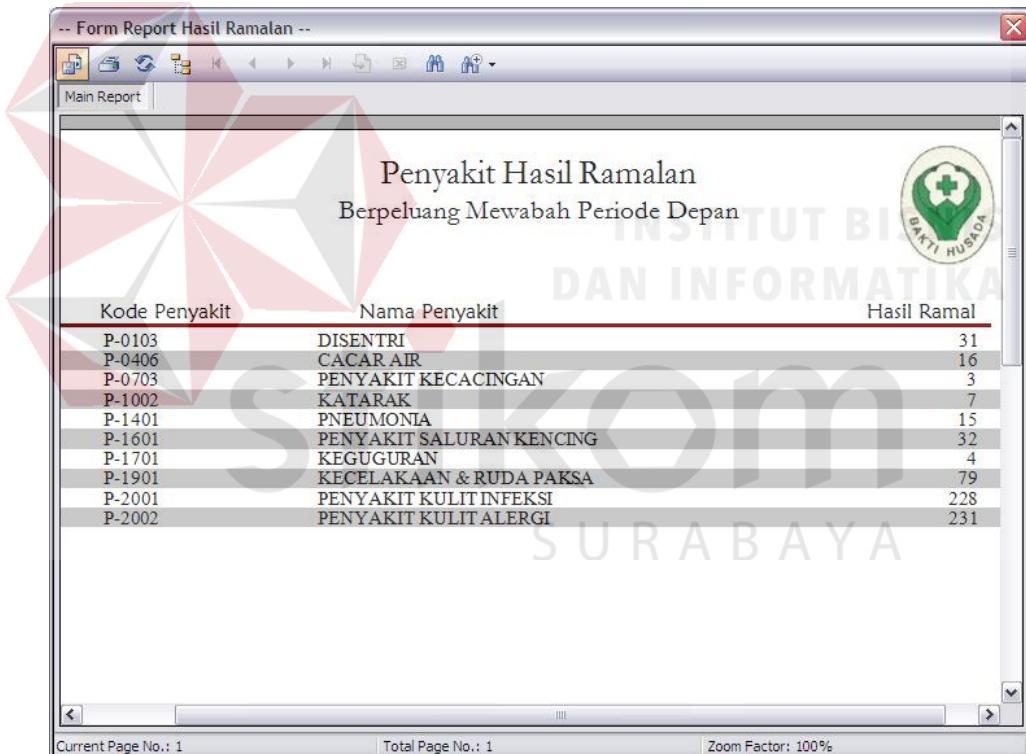
Proses ini untuk menghasilkan laporan yang diambil dari database dan ditampilkan dalam form lewat *crystal report*.

Tabel 4.16 Tabel Test Case Laporan

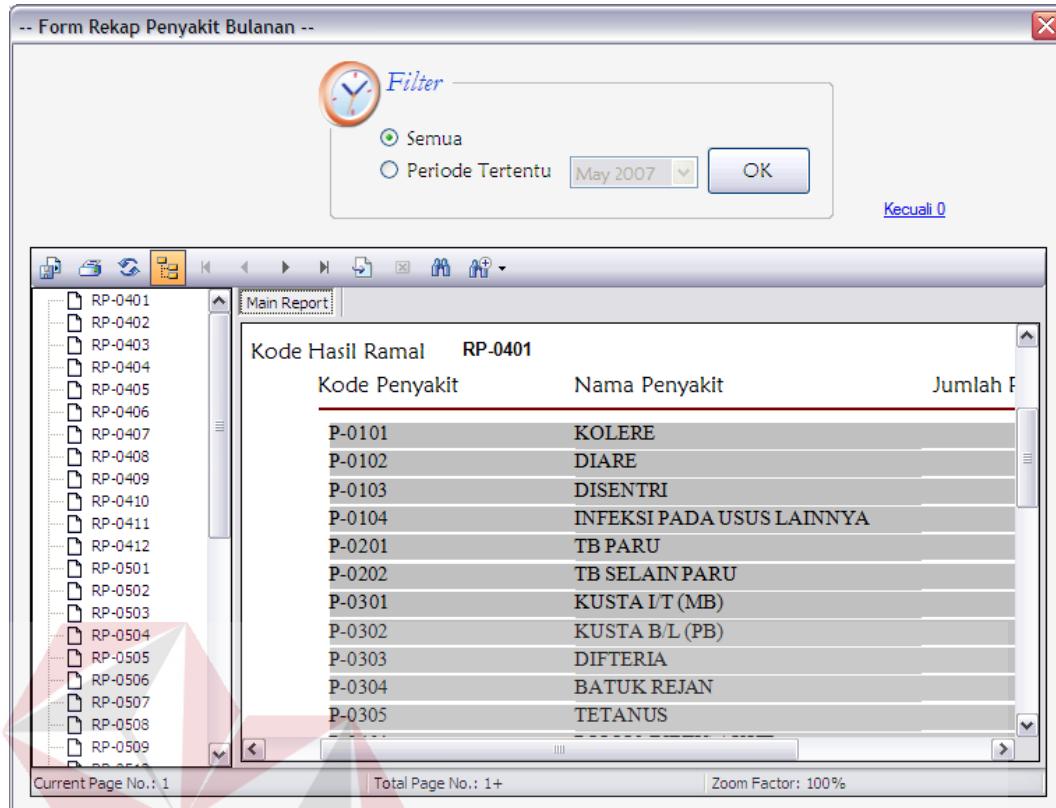
Test Case ID	Tujuan	Input	Output Diharapkan	Status
49	Menampilkan laporan hasil peramalan	Menekan menu Laporan, lalu memilih sub menu Laporan Hasil Ramalan	Form Laporan Hasil Peramalan muncul dan data laporan tampil pada crystal report	Sukses (Gambar 4.59)
50	Menampilkan laporan rekap penyakit	Menekan menu Laporan, lalu memilih sub menu Laporan Rekap Penyakit	Form Laporan Rekap Penyakit muncul dan data laporan tampil pada crystal report	Sukses (Gambar 4.60)

Tabel 4.16 Tabel Test Case Laporan (lanjutan)

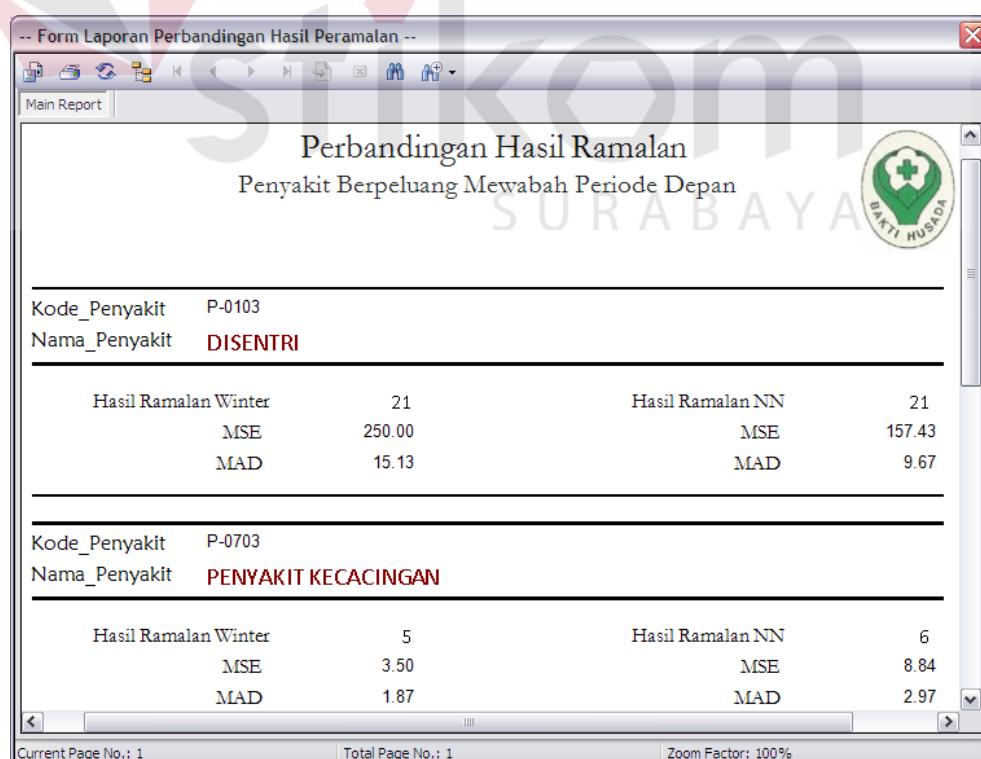
Test Case ID	Tujuan	Input	Output Diharapkan	Status
51	Menampilkan laporan hasil perbandingan	Menekan menu Laporan, lalu memilih sub menu Laporan Perbandingan Hasil Ramalan	Form Laporan Hasil Perbandingan muncul dan data laporan tampil pada crystal report	Sukses (Gambar 4.61)
52	Menampilkan laporan perkiraan penggunaan obat	Menekan menu Laporan, lalu memilih sub menu Laporan Perkiraan Penggunaan Obat	Form Laporan Perkiraan Obat muncul dan data laporan tampil pada crystal report	Sukses (Gambar 4.62)



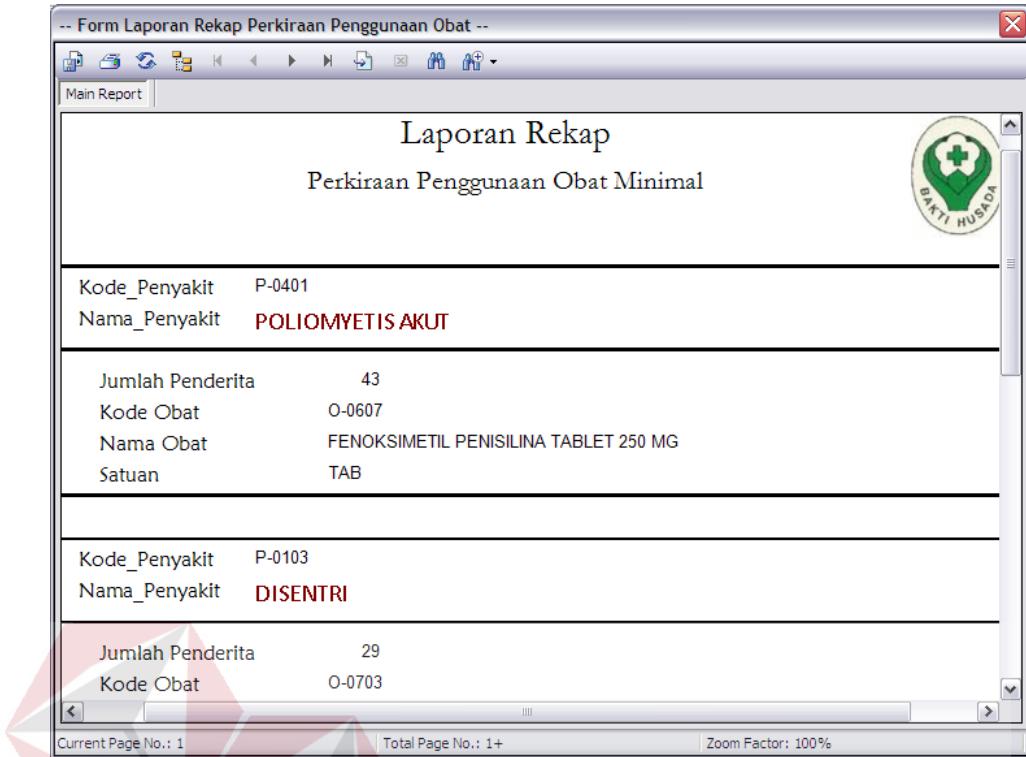
Gambar 4.59 Uji Coba Fitur Laporan Hasil Ramalan



Gambar 4.60 Uji Coba Fitur Laporan Hasil Rekap Penyakit



Gambar 4.61 Uji Coba Fitur Laporan Hasil Perbandingan Ramalan



Gambar 4.62 Uji Coba Fitur Laporan Perkiraan Penggunaan Obat

4.3.2 Uji Coba Proses Pelatihan NN dan Pengujian NN

Uji coba proses pelatihan NN dilakukan dengan memilih beberapa data penyakit dengan panjang *time series* tertentu. Selain itu juga dimasukkan data panjang ramalan dan data panjang musim. Untuk proses keluaran data latih, berupa tiga konstanta untuk proses peramalan. Pada tabel 4.17 bisa dilihat salah satu contoh data penyakit yang akan diuji yaitu penyakit dengan kode penyakit P-0103 dengan pola NN 14-10-3.

Tabel 4.17 Tabel Data Penyakit Diuji

Kode_Penyakit	Pola NN
P-0103	14-10-3

Tabel 4.18 Tabel Data Latih

Data 1	Data 2	Data 3	Data 4	Data 5	Data 6	Data 7	Data 8	Data 9	Data 10	Data 11	Data 12
0.30	0.10	0.90	0.50	0.70	0.30	0.40	0.90	0.80	0.40	0.30	0.10
0.90	0.50	0.10	0.10	0.80	0.40	0.20	0.10	0.10	0.20	0.10	0.20
0.60	0.70	0.20	0.30	0.50	0.80	0.50	0.20	0.40	0.40	0.60	0.50
0.30	0.50	0.30	0.80	0.40	0.10	0.60	0.60	0.80	0.10	0.20	0.10
0.70	0.40	0.70	0.70	0.10	0.50	0.90	0.90	0.40	0.70	0.50	0.40
0.20	0.70	0.90	0.20	0.20	0.10	0.20	0.20	0.20	0.10	0.30	0.30
0.10	0.90	0.20	0.90	0.90	0.90	0.10	0.60	0.10	0.80	0.10	0.10
0.60	0.20	0.70	0.40	0.60	0.70	0.60	0.50	0.90	0.40	0.90	0.80
0.30	0.40	0.50	0.50	0.30	0.40	0.20	0.20	0.60	0.20	0.30	0.30
0.60	0.15	0.30	0.80	0.20	0.30	0.50	0.20	0.50	0.20	0.20	0.20

Penyakit yang diuji ditunjukkan pada tabel 4.17 yang menggunakan pola NN 14-10-3 dengan panjang ramalan 1 dan panjang musim 4. Data latih yang digunakan ditunjukkan pada tabel 4.18 sebanyak 12 rentet data *time series* yang mewakili data penyakit selama 12 periode. Hasil output yang berupa konstanta ditunjukkan pada tabel 4.19. Konstanta-konstanta ini menunjukkan tiga konstanta awal yang akan digunakan sebagai konstanta pada proses peramalan exponential smoothing oleh Winter.

Tabel 4.19 Tabel Target Output Data Latih

Konstanta 1	Konstanta 2	Konstanta 3
0.90	0.10	0.80
0.50	0.40	0.10
0.40	0.90	0.10
0.20	0.90	0.20
0.10	0.90	0.10
0.70	0.10	0.90
0.10	0.10	0.10
0.10	0.50	0.10
0.20	0.90	0.10
0.10	0.20	0.40

Tabel 4.20 Tabel Hasil Test Case Proses Pelatihan NN

Test Case ID	Tujuan	Input	Output Diharapkan	Status
53	Mengetahui proses pelatihan NN bisa menghasilkan output	Data skenario jenis penyakit sebanyak 10 baris masing-masing 12 data <i>time series</i> dengan menggunakan salah satu pola NN	<i>Mean Square Error</i> (MSE)	Sukses (Gambar 4.63)



Gambar 4.63 Uji Coba Proses Pelatihan NN

Tabel 4.21 Tabel Iterasi dan MSE Proses Pelatihan NN

Iterasi (Epoch)	MSE
1	0.18394
100	0.00442
500	0.00087
1000	0.00068
2000	0.00053
3000	0.00037
4000	0.00017
4389	0.00002

Tabel 4.22 Tabel Hasil Test Case Proses Pengujian NN

Test Case ID	Tujuan	Input	Output Diharapkan	Status
54	Menguji tingkat kebenaran NN	Data skenario dari data latih, dicoba satu persatu, dengan pola NN yang sama dengan proses pelatihan	Tiga konstanta untuk proses peramalan yang nilainya sama dengan target data latih	Sukses (Gambar 4.64)

The screenshot shows a software window titled "Form Uji NN". At the top, there are three input fields: "Jumlah Neuron Input" (14), "Jumlah Neuron Hidden" (10), and "Jumlah Neuron Output" (3). Below these, there is a table titled "Data Latih" (Training Data) with 12 rows of data. The table has columns for Data 6 through Data 12 and Target 1, 2, 3. The last row of the table is highlighted in yellow. Below the training data table is a smaller table titled "Data Uji" (Test Data) with 12 rows, each with a "No" column and a "Nilai" column. To the right of the tables is a button labeled "Uji >>>". On the far right, there is a panel with the following values: "Nilai Alpha: 0.7", "Nilai Beta: 0.1", and "Nilai Teta: 0.9".

Gambar 4.64 Proses Pengujian NN

Tabel 4.23 Tabel Output Pelatihan NN

Nilai Alpha	Nilai Beta	Nilai Teta
0.90	0.10	0.80
0.50	0.40	0.10
0.40	0.90	0.10
0.20	0.90	0.20
0.10	0.90	0.10
0.70	0.10	0.90
0.10	0.10	0.10
0.10	0.50	0.10
0.20	0.90	0.10
0.10	0.20	0.40

Tabel 4.24 Tabel Hasil Test Case Proses Pencarian Pola NN Terbaik

Test Case ID	Tujuan	Input	Output Diharapkan	Status
55	Mengetahui pola NN terbaik untuk tiap penyakit	Data skenario berupa data penyakit dan pola NN hasil looping mulai 5-1-3 sampai 15-100-3 yang berjalan dengan kelipatan 10	Pola NN terbaik, <i>Mean Square Error (MSE)</i> , <i>Mean Absolute Deviation (MAD)</i> , tiga konstanta peramalan	Sukses (Gambar 4.65)

Gambar 4.65 Uji Coba Pencarian Pola NN Terbaik

Tabel 4.25 Tabel Hasil Test Case Proses Pencarian Panjang Ramalan Terbaik

Test Case ID	Tujuan	Input	Output Diharapkan	Status
56	Mengetahui panjang ramalan terbaik untuk tiap penyakit	Data skenario berupa data penyakit dan pola NN hasil looping mulai periode 1 sampai 8	Panjang periode terbaik, <i>Mean Square Error (MSE)</i> , <i>Mean Absolute Deviation (MAD)</i> , tiga konstanta peramalan	Sukses (Gambar 4.66)

-- Form Cari Pjg Ramalan Terbaik --

Data Penyakit

Kode Penyakit: P-0406
Nama Penyakit: CACAR AIR

Hasil Ramalan

08

Mulai **Simpan**

ID	PjgRamalan	DataAktual	NilaiRamala	MSE	MAD
P-001	1	5	7	3.46	1.86
P-002	2	20	7	170.78	14.79
P-003	3	19	9	133.45	12.3
P-004	4	9	8	89.28	8.52
P-005	5	11	10	67.06	6.55
P-006	6	16	10	60.79	6.43
P-007	7	6	11	55.47	6.26
P-008	8	24	10	75.44	7.36

Panjang Ramalan Terbaik = 1 [Ambil Nilai Terbaik](#)

Nilai MSE: 3.46 Nilai Alpha: 0.10
Nilai MAD: 1.86 Nilai Beta: 0.40
Nilai Teta: 0.10

Gambar 4.66 Uji Coba Pencarian Panjang Ramalan Terbaik

4.3.3 Uji Coba Proses Perbandingan Peramalan Dengan Bantuan NN

Uji coba proses peramalan exponential smoothing oleh Winter dengan bantuan neural network adalah menggunakan tiga konstanta hasil pelatihan pola neural network. Konstanta ini akan dipakai dalam rumus proses peramalan. Proses ini bertujuan untuk mengetahui seberapa tepat bantuan dari neural network dalam proses peramalan yang ditandai dengan nilai MSE dan MAD yang dihasilkan. Dengan nilai konstanta yang tepat, maka hasil peramalan akan semakin mendekati kebenaran.

Tabel 4.26 Tabel Data Penyakit Diuji

Kode_Penyakit	Pola NN	Panjang Ramalan	Panjang Musim
P-0103	14-10-3	1	4

Tabel 4.27 Tabel Data Time Series Diuji

Data 1	Data 2	Data 3	Data 4	Data 5	Data 6	Data 7	Data 8	Data 9	Data 10	Data 11	Data 12
0.30	0.10	0.90	0.50	0.70	0.30	0.40	0.90	0.80	0.40	0.30	0.10

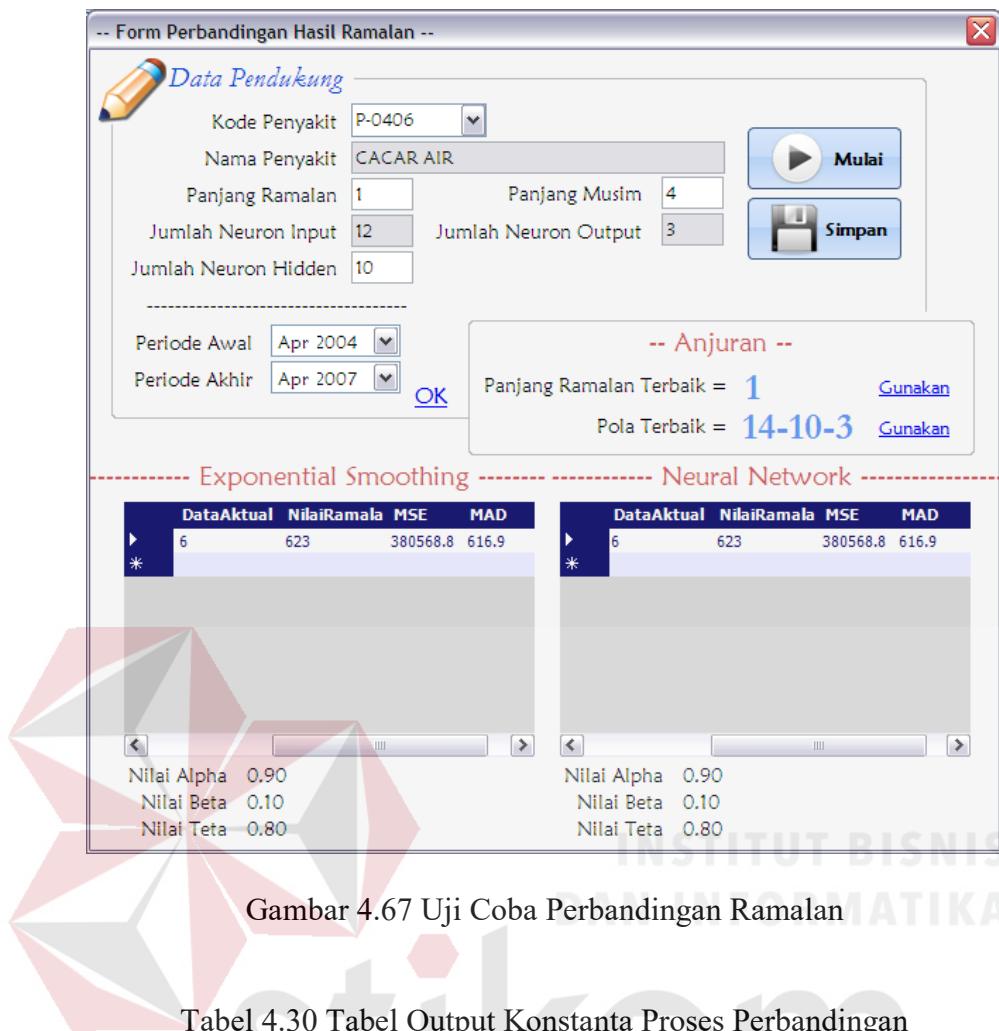
Tabel 4.28 Tabel Output Konstanta Diharapkan

Konstanta 1	Konstanta 2	Konstanta 3
0.90	0.10	0.80

Pada tabel 4.26 ditunjukkan data yang akan diuji yaitu penyakit dengan kode penyakit P-0406 yang menggunakan pola NN 14-10-3 sesuai dengan data latih. Kemudian panjang ramalan ditentukan nilainya sebanyak 1 dan panjang musim nilainya 4. Tabel 4.27 terdapat sekumpulan data time series yang akan diuji dan pada tabel 4.28 terdapat konstanta peramalan yang diharapkan keluar sesuai dengan data latih. Beberapa *test case* dari proses peramalan ini ditunjukkan pada tabel 4.29 untuk proses pengujian. Gambar 4.67 menunjukkan tampilan form hasil pengujian perbandingan peramalan, sedangkan tabel 4.30 adalah output konstanta hasil proses perbandingan.

Tabel 4.29 Tabel Hasil Test Case Perbandingan Peramalan

Test Case ID	Tujuan	Input	Output Diharapkan	Status
57	Membandingkan MSE dan MAD peramalan tanpa bantuan NN dan dengan bantuan NN	Data time series penyakit, pola NN	MSE dan MAD untuk tiap peramalan tanpa bantuan NN dan dengan bantuan NN	Sukses (Gambar 4.67)



Gambar 4.67 Uji Coba Perbandingan Ramalan

Tabel 4.30 Tabel Output Konstanta Proses Perbandingan

Nilai Alpha	Nilai Beta	Nilai Teta
0.90	0.10	0.80
0.50	0.40	0.10
0.40	0.90	0.10
0.20	0.90	0.20
0.10	0.90	0.10
0.70	0.10	0.90
0.10	0.10	0.10
0.10	0.50	0.10
0.20	0.90	0.10
0.10	0.20	0.40

4.3.4 Analisis Hasil Uji Coba

Pada bagian ini akan ditampilkan analisis hasil uji coba yang telah dilakukan pada aplikasi dengan hasil sebagai berikut:

1. Analisis Hasil Uji Coba Fitur Dasar Sistem

Analisis hasil uji coba dari keseluruhan uji yang dilakukan akan menentukan kelayakan fitur dasar sistem berdasarkan desain yang telah ditetapkan. Fitur-fitur dasar sistem dinilai layak bilamana keseluruhan hasil uji coba ini sesuai dengan *output* yang diharapkan. Pada uji coba yang telah dilakukan pada fitur-fitur dasar sistem seperti tampak pada test case 1 sampai test case 52 dapat disimpulkan bahwa fitur-fitur dasar tersebut telah berjalan dengan baik dan tidak terdapat *error*. Fungsi-fungsi tambah, ubah, hapus, simpan maupun tampil dapat berjalan sebagaimana mestinya.

2. Analisis Hasil Uji Coba Perhitungan Sistem

Analisis hasil uji coba dilakukan untuk menentukan kelayakan proses pelatihan dan proses peramalan berdasarkan desain yang telah ditetapkan. Proses pelatihan dan peramalan dinilai layak bilamana keseluruhan hasil uji coba ini sesuai dengan *output* yang diharapkan. Dari uji coba perhitungan sistem yang telah dilakukan didapatkan hasil bahwa proses pelatihan dan peramalan dengan data yang diskenariokan telah berjalan dengan baik. Pelatihan data seperti tampak pada test case 53 berhasil dilakukan dan menghasilkan *Mean Square Error* (MSE). Proses pengujian neural network untuk mendapatkan output yang sesuai dengan target data latih juga telah tercapai, perbandingan antara target data latih dengan output proses pengujian bisa dilihat pada tabel 4.23 dan tabel 4.30.

3. Analisis Hasil Uji Coba Validasi Sistem

Analisis hasil uji coba validasi sistem dilakukan untuk mengetahui dan menganalisa apakah proses-proses utama dalam sistem dengan masukan keseluruhan data yang ada telah berjalan sebagaimana mestinya.

Tabel 4.31 Tabel Pengaruh Panjang Periode Ramalan Terhadap MSE

Kode Penyakit	MSE Periode 1	MSE Periode 2	MSE Periode 3	MSE Periode 4	MSE Periode 5	MSE Periode 6	MSE Periode 7	MSE Periode 8	Periode Terbaik
P-0703	0.22	0.27	0.24	7.45	6.64	5.35	4.88	4.27	1
P-1701	12.39	14.79	15.6	12.37	13.46	12.78	11.04	10.14	8
P-1601	46.34	46.34	44.14	40.9	92.73	76	154.7	174.4	4
P-1002	2.45	12.15	9.75	40.43	144.34	377.33	473.54	458.38	1
P-0406	3.46	170.78	133.45	89.28	67.06	60.79	55.47	75.44	1
P-1401	9.4	33.14	37.87	26.51	20.22	22.72	20.7	250	1
P-1901	51.85	55.98	95.42	79.17	82.52	81.5	95.81	86.75	1
P-2002	341.73	491.75	313.5	349.51	281.14	234.25	196.11	255.47	7
P-0103	539.78	560.02	280.76	214.12	170.85	139.66	120.62	103.42	8
P-2001	39.01	73.63	106.65	82.37	291.55	261.67	294.11	440.26	1

Pada tabel 4.31 dapat dilihat proses pencarian panjang periode peramalan untuk beberapa jenis penyakit. Setiap periode peramalan dimunculkan MSE yang terjadi dan di sebelah paling kanan adalah periode peramalan yang terbaik ditandai dengan MSE yang paling kecil dari 8 kali periode. Dari 10 data yang diuji, terlihat bahwa tiap penyakit mempunyai panjang periode peramalan terbaik yang berbeda-beda hal ini dikarenakan pola *time series* yang berbeda-beda pada tiap data penyakit. Namun, dari proses ini juga diketahui bahwa 6 dari 10 kali percobaan, panjang periode ramalan sebanyak 1 paling banyak muncul. Hal ini berarti, panjang periode ramalan sebanyak 1 paling sedikit menghasilkan *error*.

Tabel 4.32 Tabel Data Skenario Diuji

Data 1	Data 2	Data 3	Data 4	Data 5	Data 6	Data 7	Data 8	Data 9	Data 10	Data 11	Data 12
0.30	0.10	0.90	0.50	0.70	0.30	0.40	0.90	0.80	0.40	0.30	0.10
0.90	0.50	0.10	0.10	0.80	0.40	0.20	0.10	0.10	0.20	0.10	0.20
0.60	0.70	0.20	0.30	0.50	0.80	0.50	0.20	0.40	0.40	0.60	0.50

Data *time series* pada tabel 4.32 adalah data yang akan diuji dalam proses pengujian NN per iterasi pelatihan. Untuk mengetahui hasil pengaruh iterasi terhadap MSE dan MAD peramalan bisa dilihat pada tabel 4.33.

Tabel 4.33 Tabel Pengaruh Banyak Iterasi Pelatihan Terhadap MSE dan MAD hasil peramalan

Banyak Iterasi	Galat Pelatihan NN	Time Series 1		Time Series 2		Time Series 3	
		MSE Hasil Peramalan	MAD Hasil Peramalan	MSE Hasil Peramalan	MAD Hasil Peramalan	MSE Hasil Peramalan	MAD Hasil Peramalan
1	0.5582	188729	337	235452	329	104324	284
10	0.2464	134613	349	215417	327	97432	282
100	0.0492	103457	276	208923	325	97432	282
500	0.0401	103457	276	208923	325	85433	241
1000	0.0023	98183	266	207103	324	79324	221
2000	0.0008	98183	266	207103	324	73806	208
4000	0.0004	98183	266	207103	324	73806	208

Proses pelatihan dilakukan dalam beberapa iterasi, setiap kelipatan iterasi, dihitung proses peramalan dengan bantuan output berupa konstanta dari pengujian *neural network*. Dari 3 data yang diuji, terlihat bahwa semakin galat mendekati 0.0000 maka proses peramalan akan menghasilkan konstanta yang sesuai dengan target yang ada pada data latih. Sehingga MSE dan MAD hasil peramalan juga semakin kecil.

Tabel 4.34 Tabel Pengaruh Pola NN Ramalan Terhadap MSE dan MAD (1)

Pola NN	MSE Hasil Ramal	MAD Hasil Ramal	Pola NN	MSE Hasil Ramal	MAD Hasil Ramal	Pola NN	MSE Hasil Ramal	MAD Hasil Ramal
7-1-3	385.12	15.54	10-1-3	349.21	15.32	13-1-3	225.04	12.18
7-10-3	395.8	15.7	10-10-3	248.47	13.27	13-10-3	424.09	16.2
7-20-3	397.61	15.73	10-20-3	168.39	10.16	13-20-3	360.76	13.8
7-30-3	367.94	15.28	10-30-3	370.1	15.08	13-30-3	258.89	12.67
7-40-3	260.64	13.38	10-40-3	386.34	15.18	13-40-3	282.93	12.07
7-50-3	307.97	14.28	10-50-3	181.85	9.87	13-50-3	189.92	11.08
7-60-3	330.37	14.67	10-60-3	212.04	11.94	13-60-3	175.33	10.41
7-70-3	470.17	16.74	10-70-3	334.39	14.3	13-70-3	121.37	8.11
7-80-3	324.12	14.56	10-80-3	206.4	10.9	13-80-3	191.12	11.23
7-90-3	254.48	13.25	10-90-3	202.3	11.68	13-90-3	436.35	16.72
7-100-3	255.68	13.28	10-100-3	319.82	14.14	13-100-3	295.26	12.69
8-1-3	372.54	15.69	11-1-3	412.93	17.44	14-1-3	259.91	12.62
8-10-3	507.03	18.67	11-10-3	312.86	13.32	14-10-3	292.06	12.8
8-20-3	579.92	19.82	11-20-3	612.72	18.03	14-20-3	276.27	12.07
8-30-3	499.85	18.49	11-30-3	152.13	9.78	14-30-3	774.52	20.87
8-40-3	564.82	19.55	11-40-3	377.42	16.28	14-40-3	233.16	11.93
8-50-3	229.3	12.65	11-50-3	309.85	14.91	14-50-3	155.61	10
8-60-3	211.24	11.82	11-60-3	341.5	13.08	14-60-3	721.35	21.22
8-70-3	529.54	18.85	11-70-3	520.86	17.17	14-70-3	422.33	14.91
8-80-3	353.57	15.64	11-80-3	312.84	14.44	14-80-3	345.22	14.28
8-90-3	435.33	17.58	11-90-3	531.71	16.76	14-90-3	229.39	11.72
8-100-3	557.57	19.43	11-100-3	582.9	19.53	14-100-3	258.57	13.46
9-1-3	306.57	14.27	12-1-3	415.96	17.3	15-1-3	280.08	13.77
9-10-3	222.62	11.96	12-10-3	416.08	16.03	15-10-3	278.38	12.82
9-20-3	246.49	12.9	12-20-3	582.42	19.86	15-20-3	288.96	13.63
9-30-3	315.87	14.97	12-30-3	374.6	16.5	15-30-3	292.64	12.81
9-40-3	303.52	14.41	12-40-3	371.82	15.76	15-40-3	674.56	20.34
9-50-3	813.13	22.2	12-50-3	156.34	9.67	15-50-3	241.38	11.95
9-60-3	470.68	17.45	12-60-3	258.84	11.1	15-60-3	530.33	17.15
9-70-3	231.66	11.16	12-70-3	136.01	9.06	15-70-3	242.86	11.36
9-80-3	847.66	22.55	12-80-3	526.66	16.17	15-80-3	440.97	14.97
9-90-3	255.03	13.62	12-90-3	252.78	11.19	15-90-3	274.7	11.84
9-100-3	246.93	13.67	12-100-3	586.73	16.73	15-100-3	219.57	10.69

Tabel 4.34 adalah hasil perhitungan dengan menggunakan data *time series* dari jenis penyakit disentri dengan kode P-0103. Pola terbaik adalah 13-70-3.

Tabel 4.35 Tabel Pengaruh Pola NN Ramalan Terhadap MSE dan MAD (2)

Pola NN	MSE Hasil Ramal	MAD Hasil Ramal	Pola NN	MSE Hasil Ramal	MAD Hasil Ramal	Pola NN	MSE Hasil Ramal	MAD Hasil Ramal
7-1-3	101.82	7.58	10-1-3	97.91	7.86	13-1-3	76.84	7.11
7-10-3	76.13	6.62	10-10-3	142.1	8.7	13-10-3	165.39	9.5
7-20-3	54.21	5.41	10-20-3	80.09	6.41	13-20-3	117.71	9.09
7-30-3	123.81	8.24	10-30-3	107.67	8.46	13-30-3	71.79	6.52
7-40-3	43.35	3.9	10-40-3	112.57	8.72	13-40-3	96.27	8.08
7-50-3	127.74	8.35	10-50-3	65.33	6.24	13-50-3	55.21	5.77
7-60-3	45.11	4.43	10-60-3	95.33	7.67	13-60-3	55.41	5.76
7-70-3	92.92	7.27	10-70-3	97.39	7.93	13-70-3	66.61	6.07
7-80-3	82.28	6.87	10-80-3	60.96	6.23	13-80-3	74.24	6.53
7-90-3	43.49	4	10-90-3	96.96	7.77	13-90-3	135.57	9.19
7-100-3	43.46	3.98	10-100-3	91.52	7.53	13-100-3	95.09	7.97
8-1-3	62.14	6.3	11-1-3	110.72	7.38	14-1-3	92	7.15
8-10-3	87.13	7.58	11-10-3	86.91	7.09	14-10-3	105.12	7.98
8-20-3	101	8.14	11-20-3	201.92	11.91	14-20-3	116.94	8.84
8-30-3	80.75	7.28	11-30-3	82.82	6.69	14-30-3	197.51	10.39
8-40-3	79.39	7.16	11-40-3	106.06	7.28	14-40-3	138.8	9.47
8-50-3	71.24	5.79	11-50-3	95.35	7.15	14-50-3	63.28	6.36
8-60-3	85.63	6.38	11-60-3	94.72	7.66	14-60-3	191.06	10.88
8-70-3	70.59	6.68	11-70-3	163.1	10.68	14-70-3	194.22	11.67
8-80-3	67.87	6.56	11-80-3	89.73	7.5	14-80-3	171.96	10.26
8-90-3	107.28	8.36	11-90-3	176.07	11.09	14-90-3	92.67	7.64
8-100-3	84.03	7.4	11-100-3	136.4	8.1	14-100-3	90.74	7.68
9-1-3	49.04	5.66	12-1-3	106.53	7.53	15-1-3	92.37	7.36
9-10-3	70.39	6.5	12-10-3	122.09	7.75	15-10-3	92.19	7.63
9-20-3	59.52	6.01	12-20-3	131.7	8.4	15-20-3	100.77	8.08
9-30-3	53.07	5.93	12-30-3	95.87	7.71	15-30-3	100.04	8.13
9-40-3	47.47	5.49	12-40-3	107.29	8.03	15-40-3	175.45	10.26
9-50-3	189.5	10.75	12-50-3	55.89	5.27	15-50-3	70.37	7.03
9-60-3	67.66	6.1	12-60-3	73.51	6.49	15-60-3	245.23	13.1
9-70-3	61	6.1	12-70-3	79.16	6.74	15-70-3	68.38	7.15
9-80-3	238.88	12.31	12-80-3	177.14	10.86	15-80-3	206.93	12.31
9-90-3	72.35	6.6	12-90-3	75.57	6.71	15-90-3	114.76	8.72
9-100-3	98	7.45	12-100-3	211.32	11.66	15-100-3	59.21	6.26

Tabel 4.35 adalah hasil perhitungan dengan menggunakan data *time series* dari jenis penyakit cacar air dengan kode P-0406. Pola terbaik adalah 7-40-3.

Dari proses pencarian pola NN terbaik ini, bisa dilihat jika pola NN untuk menghasilkan MSE dan MAD yang paling kecil adalah tidak sama untuk tiap penyakit. Belum tentu pola terbaik pada penyakit A, baik juga untuk diujikan ke penyakit B. Masing-masing penyakit memiliki pola sendiri-sendiri karena memiliki karakteristik *time series* yang berbeda.



BAB V

PENUTUP

5.1 Kesimpulan

Berdasarkan hasil uji coba dan analisa yang telah dilakukan dalam pembuatan aplikasi Peramalan Kewaspadaan Penyakit Musiman Dengan Metode Exponential Smoothing Oleh Winter dengan Studi Kasus Puskesmas Kupang Kabupaten Mojokerto, dapat diambil kesimpulan sebagai berikut:

1. Setiap penyakit memiliki karakteristik data *time series* yang berbeda sehingga masing-masing penyakit memiliki panjang periode ramalan berbeda dengan penyakit lainnya. Panjang periode ramalan yang paling sedikit menghasilkan *error* adalah sebanyak 1 periode
2. Pola jaringan *neural network* terbaik bagi suatu penyakit, belum tentu terbaik juga untuk penyakit lainnya karena setiap penyakit memiliki karakteristik data *time series* yang berbeda.
3. Berdasarkan hasil uji coba didapatkan bahwa semakin banyak iterasi pelatihan jaringan, maka hasil MSE dan MAD akan semakin kecil dan konstanta peramalan yang dihasilkan akan semakin mendekati target output dari data latih.

5.2 Saran

Adapun saran-saran yang dapat digunakan untuk mengembangkan aplikasi yang telah dibuat adalah sebagai berikut:

1. Pengembangan dengan proses peramalan musiman metode lain seperti metode dekomposisi klasik, regresi berganda runtut waktu, dan Box-Jenkins.

2. Pengembangan hasil peramalan dengan menyertakan faktor-faktor penyebab suatu penyakit bisa bersifat musiman.
3. Pengembangan dengan menggunakan teknologi web sehingga bisa diakses oleh orang lain di luar pihak puskesmas.



DAFTAR PUSTAKA

Arsyad, Lincoln. 2001. Peramalan Bisnis Edisi Pertama. Jogyakarta: Universitas Gajah Mada.

Kusumadewi, Sri., Hartati, Sri., 2006, NEURO FUZZY: Integrasi Sistem Fuzzy & Jaringan Syaraf. Yogyakarta: Graha Ilmu.

Microsoft Corporation. 2005. MSDN Library for Visual Studio .NET 2005. Microsoft Corporation

Purnomo, Mauridhi Hery, Kurniawan, Agus. 2006. Supervised Neural Networks dan Aplikasinya. Yogyakarta: Graha Ilmu.

Romeo, ST. 2003. Testing dan Implementasi Sistem. Surabaya: STIKOM.

Siang, Jong Jek. 2005. Jarangan Syaraf Tiruan & Pemrogramannya Menggunakan Matlab. Yogyakarta: Andi Offset.

Spyros, M., Steven, C., Wheelwright, V. E. Mcgee. 1991. Metode dan Aplikasi Peramalan. Jakarta: Erlangga.

