



**SISTEM PENDUKUNG KEPUTUSAN STATUS GIZI BALITA  
MENGUNAKAN K-NEAREST NEIGHBOR**

**SKRIPSI**

oleh

**Yusa Dwi Iriani  
NIM 102410101121**

**PROGRAM STUDI SISTEM INFORMASI  
UNIVERSITAS JEMBER  
2015**



**SISTEM PENDUKUNG KEPUTUSAN STATUS GIZI BALITA  
MENGUNAKAN K-NEAREST NEIGHBOR**

**SKRIPSI**

diajukan guna melengkapi tugas akhir dan memenuhi salah satu syarat untuk menyelesaikan Program Studi Sistem Informasi (S1) dan mencapai gelar Sarjana Sistem Informasi

oleh

**Yusa Dwi Iriani  
NIM 102410101121**

**PROGRAM STUDI SISTEM INFORMASI  
UNIVERSITAS JEMBER  
2015**

## PERSEMBAHAN

Skripsi ini saya persembahkan untuk:

1. Allah SWT, Tuhan Yang Maha Pengasih yang telah senantiasa memberikan kelancaran dan kemudahan dalam menyelesaikan skripsi ini;
2. Bapak Kuspani dan Ibu Muldjayatiningsih yang tiada hentinya memberikan dukungan serta doanya;
3. Keluarga tersayang, terimakasih untuk segala nasehat, semangat dan perhatian yang selalu diberikan;
4. Bapak Dr. Saiful Bukhori, ST.,M.Kom dan Ibu Windi Eka Yulia Retnani S.Kom.,MT selaku dosen pembimbing yang selalu memberikan motivasi dan arahan untuk dapat menyelesaikan skripsi;
5. Ibu Leersia Yusi Ratnawati, S.KM., M.Kes., mbak Dicka Indo Putri, S.KM., serta praktisi kesehatan pada posyandu ..... yang membantu dalam pemahaman dan arahan tentang gizi.
6. Teman-teman sekaligus keluarga jember saya yang selalu menemani saya dari awal bersama dan selalu ada menyemangati dan membantu saya selama ini. Nay, Ain, Erick, Syafiq, Indra, Donny, Arbi, Hamdan, Hani, Bombom, Pipit, Awang, Rasya, Anggi, Dudud, Mala, dan teman-teman lainnya yang tak bisa saya sebutkan satu persatu;
7. Keluarga besar Program Studi Sistem Informasi Universitas Jember.

## MOTTO

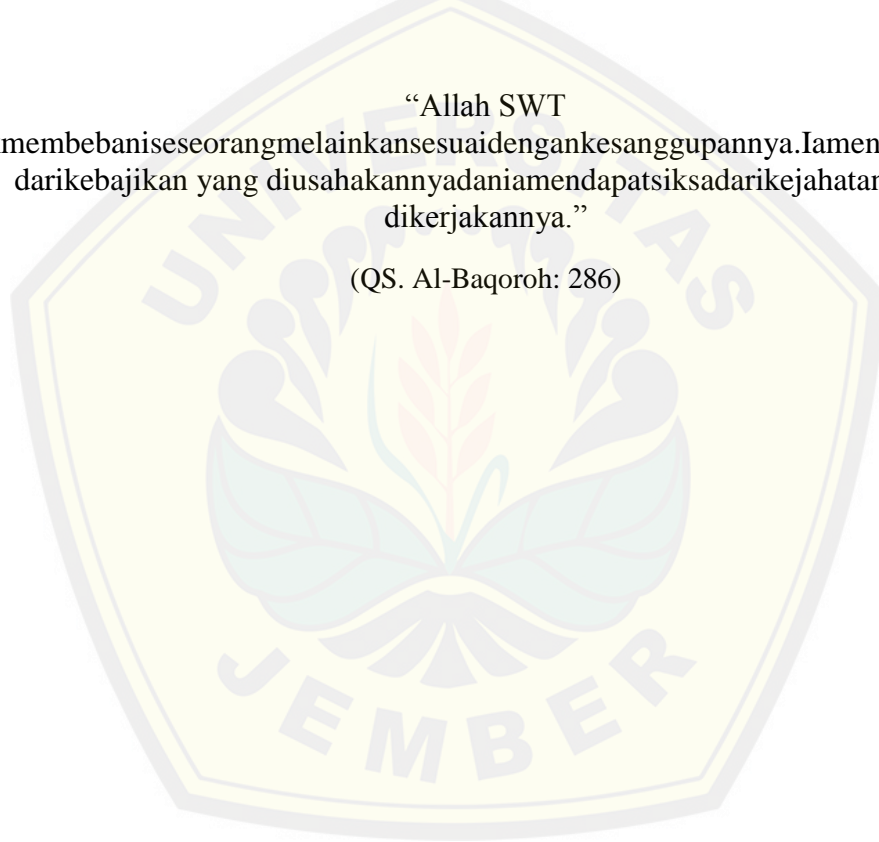
Where there is a will, there is a way

“Sesungguhnya dalam kesulitan ada kemudahan”

(QS. Al-Insyirah: 6)

“Allah SWT tidak membebani seseorang melainkan sesuai dengan kesanggupannya. Ia mendapat pahala dari kebajikan yang diusahakannya dan ia mendapat siksa dari kejahatan yang dikerjakannya.”

(QS. Al-Baqoroh: 286)



## PERNYATAAN

Saya yang bertanda tangan di bawah ini:

Nama : YUSA DWI IRIANI

NIM : 102410101121

menyatakan dengan sesungguhnya bahwa karya tulis ilmiah dengan judul “Sistem Pendukung Keputusan Status Gizi Balita Menggunakan K-Nearest Neighbor” adalah benar - benar hasil karya sendiri, kecuali kutipan yang sudah saya sebutkan sumbernya, belum pernah diajukan pada institusi manapun, dan bukan karya jiplakan. Saya bertanggung jawab atas keabsahan dan kebenaran isinya sesuai dengan sikap ilmiah yang harus dijunjung tinggi.

Demikian pernyataan ini saya buat dengan sebenar - benarnya, tanpa ada tekanan dan paksaan dari pihak manapun, serta bersedia mendapat sanksi akademik jika ternyata di kemudian hari pernyataan ini tidak benar.

Jember, 29 Juni 2015

Yang menyatakan,

Yusa Dwi Iriani

NIM.102410101121

## PENGESAHAN PEMBIMBING

Skripsi berjudul “Sistem Pendukung Keputusan Status Gizi Balita Menggunakan K-*Nearest Neighbor*”, telah diuji dan disahkan pada:

Hari, tanggal : Senin, 29 Juni 2015

Tempat : Program Studi Sistem Informasi Universitas Jember

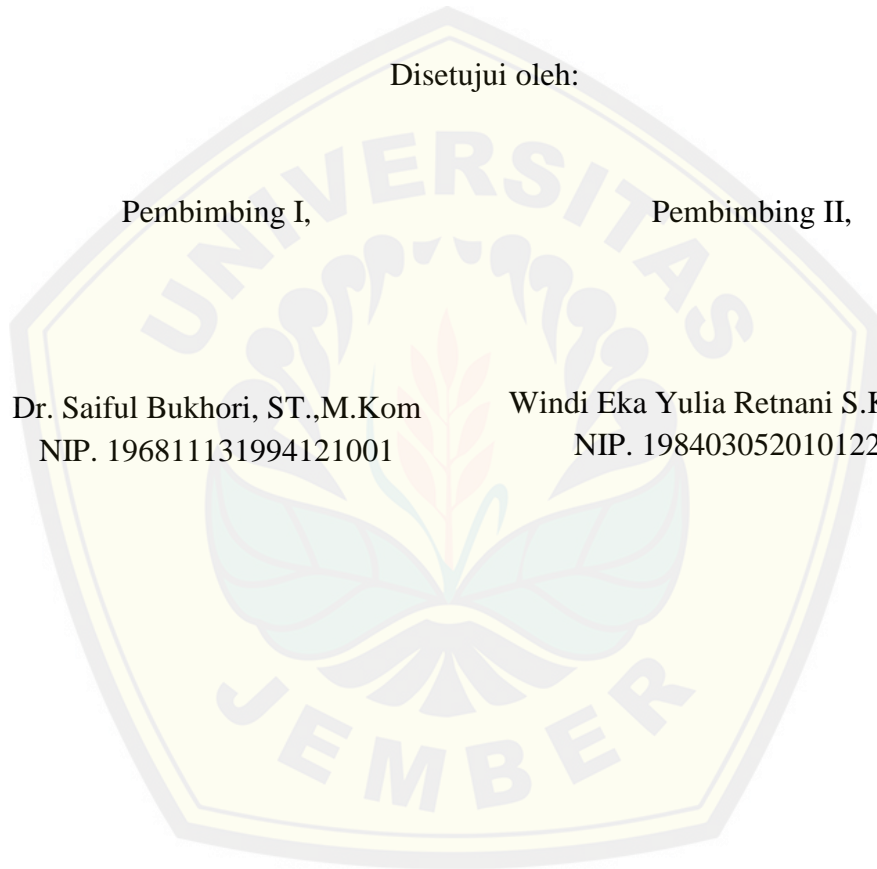
Disetujui oleh:

Pembimbing I,

Pembimbing II,

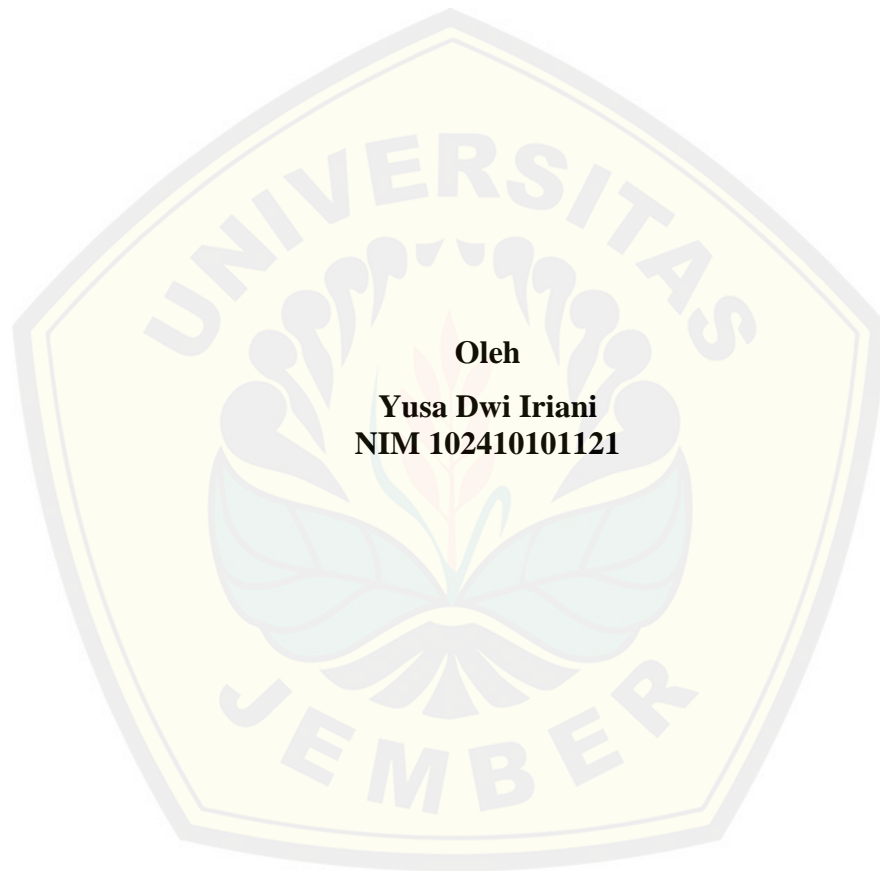
Dr. Saiful Bukhori, ST.,M.Kom  
NIP. 196811131994121001

Windi Eka Yulia Retnani S.Kom.,MT  
NIP. 198403052010122002



**SKRIPSI**

**SISTEM PENDUKUNG KEPUTUSAN STATUS GIZI BALITA  
MENGUNAKAN K-NEAREST NEIGHBOR**



Oleh  
**Yusa Dwi Iriani**  
**NIM 102410101121**

Pembimbing

Pembimbing Utama : Dr. Saiful Bukhori, ST.,M.Kom  
Pembimbing Pendamping : Windi Eka Yulia Retnani S.Kom.,MT

## PENGESAHAN

Skripsi yang berjudul “Sistem Pendukung Keputusanstatus Gizi Balita Menggunakan K-Nearest Neighbor”, telah diuji dan disahkan pada:

Hari : Senin

Tanggal : 29 Juni 2015

Tempat : Program Studi Sistem Informasi Universitas Jember.

Tim Penguji:

Penguji I,

Penguji II,

Drs. Antonius Cahya P, M. App., Sc., Ph.D  
NIP. 196909281993021001

M. Arief Hidayat, S.Kom., M.Kom  
NIP.198101232010121003

Mengesahkan,  
Ketua Program Studi Sistem Informasi  
Universitas Jember,

Prof Drs. Slamir M. Comp Sc, Ph.D  
NIP. 196704201992011001



## RINGKASAN

Sistem Penunjang Keputusan Status Gizi Balita, Yusa Dwi Iriani, 102410101121; 2015: 139 halaman; Program Studi Sistem Informasi Universitas Jember.

Gizi merupakan hal yang penting untuk keseimbangan tubuh manusia. Agar dapat menjaga kelangsungan hidup, manusia membutuhkan asupan gizi yang tepat. Kadar gizi yang dibutuhkan tiap orang berbeda beda sehingga perlu mengetahui status gizi pada dirinya sendiri. Penentuan status gizi seseorang umumnya dilakukan oleh ahli gizi atau praktisi kesehatan yang dapat mengukur dan mengklasifikasikan ukuran tersebut tergolong status gizi tertentu. Mayoritas masyarakat masih kurang peduli dengan penentuan status gizi pada balita. Bahkan untuk datang ke posyandu atau ahli gizi setempat, orang tua hanya menilai gizi anaknya secara kasat mata apakah balita gemuk tergolong bayi sehat atau normal. Karena masih kurangnya pengetahuan masyarakat, maka penilaian status gizi masih dengan cara manual. Seorang ibu terkadang hanya mengukur berat badan bayinya bertambah atau berkurang tiap bulannya. Sehingga dibutuhkan perhitungan yang valid agar status gizi bayi dapat diketahui secara akurat.

Salah satu alternatif teknologi yang dapat membantu pengukuran tersebut adalah dengan sistem penunjang keputusan status gizi menggunakan metode klasifikasi *K-Nearest Neighbor*. Maka, tingkat kesalahan dalam pengukuran dapat dikurangi menggunakan sistem berbasis web. Sistem ini telah diimplementasikan dan diuji menggunakan *fulltrain fulltest*. *Fulltrain fulltest* merupakan data *training* yang disembunyikan kelasnya untuk diuji dari data *set* secara keseluruhan. Hasil yang didapatkan dengan menggunakan  $K=3$  memiliki tingkat akurasi sebesar 78.43% pada data anak laki-laki dan 87.76% pada data anak perempuan.

## PRAKATA

Puji syukur kepada Allah SWT, atas segala limpahan rahmat, hidayat dan karunia-Nya maka penulis dapat menyelesaikan skripsi yang berjudul “Sistem Pendukung Keputusan Status Gizi Balita Menggunakan *K-Nearest Neighbor*”. Skripsi ini disusun guna melengkapi tugas akhir dan sebagai salah satu syarat menyelesaikan pendidikan Strata Satu (S1) pada Program Studi Sistem Informasi Universitas Jember.

Dalam proses penyusunan skripsi ini, penulis tidak lepas dari bantuan berbagai pihak. Maka dari itu, dalam kesempatan ini penulis ingin mengucapkan banyak terima kasih kepada:

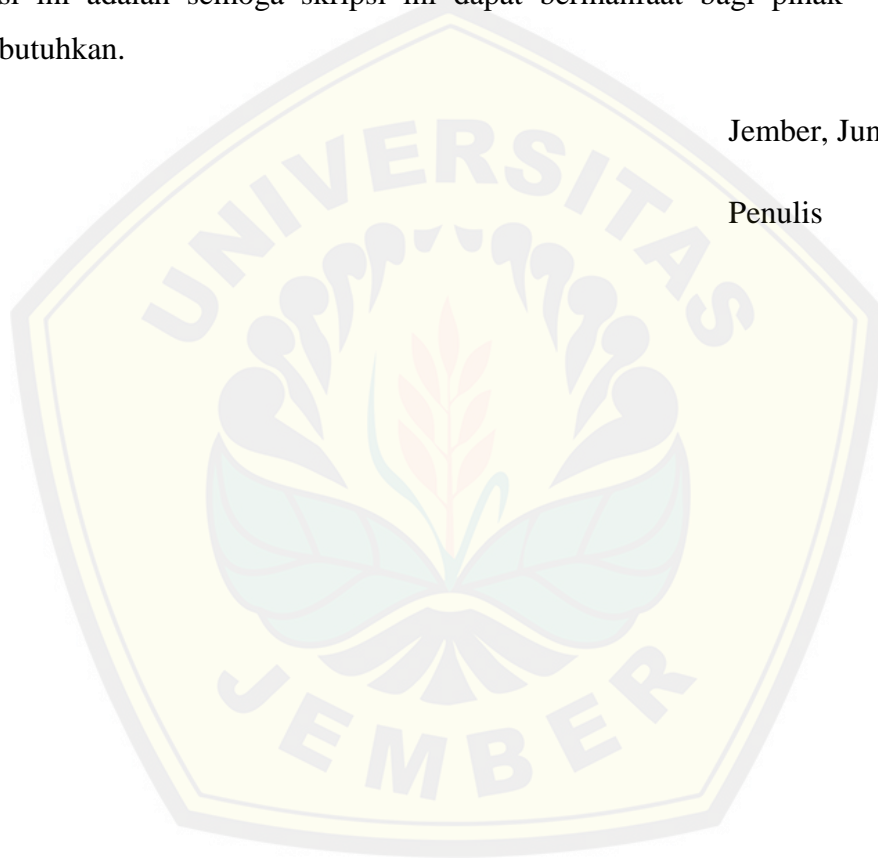
1. Bapak Dr. Saiful Bukhori, ST.,M.Kom dan Ibu Windi Eka Yulia Retnani S.Kom.,MT sebagai Dosen Pembimbing utama dan Pendamping;
2. Bapak Prof. Drs. Slamini, M.Comp.Sc.,Ph.D selaku ketua Program Studi SistemInformasi yang menjadi tempat mahasiswa menuntut ilmu;
3. Seluruh Bapak dan Ibu dosen beserta staf karyawan di Program Studi Sistem Informasi Universitas Jember;
4. Bapak, Ibuserita keluarga tercinta yang memberikan doa dan semangat demi terselesainya skripsi ini;
5. Ibu Leersia Yusi Ratnawati, S.KM., M.Kes. yang juga telah membantu banyak dalam kuliah singkat untuk memperdalam ilmu gizi;
6. Bapak Kuspandi dan Ibu Muldjayatiningsih yang meberikan banyak motivasi dan kasih sayang.
7. Kakak perempuan saya beserta keluarga dan saudara-saudara lainnya.
8. Keluarga besar Uklam Foundation Nay, Indra,Doni, Arbi, Anggi, Ain, Pipit, Rasya, Syafiq, Hamdan, Brian, Hani, Awang.
9. Teman-teman Program Studi Sistem Informasi Universitas Jember terutama Zerone dan adikku Bryan, kadek, dan Gedhe;

10. Teman-teman paguyuban Jabodetabek IMADA JAYA;
11. Penghuni kosan Jawa VII no.73, Jember beserta pak Badrun sekeluarga;
12. Serta semua pihak yang tidak dapat disebutkan satu persatu.

Demi kesempurnaan skripsi ini, penulis akan menerima segala masukan baik berupa kritik atau saran dari semua pihak. Harapan penulis dengan terselesaikannya skripsi ini adalah semoga skripsi ini dapat bermanfaat bagi pihak - pihak yang membutuhkan.

Jember, Juni 2015

Penulis



**DAFTAR ISI**

	<b>Halaman</b>
<b>PERSEMBAHAN .....</b>	<b>ii</b>
<b>MOTTO .....</b>	<b>iii</b>
<b>PERNYATAAN.....</b>	<b>iv</b>
<b>PENGESAHAN PEMBIMBING.....</b>	<b>v</b>
<b>HALAMAN PEMBIMBINGAN.....</b>	<b>vi</b>
<b>PENGESAHAN.....</b>	<b>vii</b>
<b>RINGKASAN .....</b>	<b>viii</b>
<b>PRAKATA.....</b>	<b>ix</b>
<b>DAFTAR ISI.....</b>	<b>xi</b>
<b>DAFTAR TABEL .....</b>	<b>xiv</b>
<b>DAFTAR GAMBAR.....</b>	<b>xvi</b>
<b>DAFTAR LAMPIRAN.....</b>	<b>xviii</b>
<b>BAB 1. PENDAHULUAN .....</b>	<b>1</b>
1.1. Latar Belakang .....	1
1.2. Rumusan Masalah.....	2
1.3. Tujuan dan Manfaat .....	3
1.3.1. Tujuan .....	3
1.3.2. Manfaat .....	3
1.4. Batasan Masalah .....	3
1.5. Sistematika Penulisan Skripsi .....	4
<b>BAB 2. TINJAUAN PUSTAKA.....</b>	<b>5</b>
2.1. Sistem Penunjang Keputusan.....	5
2.2. Gizi.....	5
2.3. Status Gizi.....	6
2.3.1. Macam- Macam Status Gizi.....	7
2.3.2. Penilaian Status Gizi .....	8
2.3.3. Jenis Parameter .....	10

2.4.	K – Nearest Neighbor .....	11
2.5.	Metode Waterfall .....	13
<b>BAB 3. METODOLOGI PENELITIAN.....</b>		<b>15</b>
3.1.	Pendekatan Penelitian .....	15
3.2.	Tempat dan Waktu Penelitian .....	15
3.3.	Alur Penelitian .....	15
3.3.1.	Studi Literatur .....	16
3.3.2.	Tahap Pengumpulan Data .....	17
3.3.3.	Teknik Pengumpulan data.....	17
3.4.	<i>K-Nearest Neighbor</i> untuk penentuan status gizi balita .....	18
3.5.	Metode Pembuatan Sistem.....	19
3.5.1.	Analisis Sistem.....	19
3.5.2.	Desain Sistem.....	20
3.5.3.	Implementasi sistem.....	21
3.5.4.	Pengujian Sistem.....	22
3.5.5.	Pemeliharaan dan Perbaikan .....	23
<b>BAB 4. ANALISIS DAN PERANCANGAN .....</b>		<b>24</b>
4.1.	Pengumpulan data.....	24
4.1.1.	Data Set .....	24
4.1.2.	Data Normalisasi.....	24
4.2.	Analisis kebutuhan.....	27
4.3.	Desain Sistem.....	28
4.3.1.	Business Proses .....	28
4.3.2.	<i>Usecase</i> Diagram .....	29
4.3.3.	Skenario Diagram .....	30
4.3.4.	Sequence Diagram .....	37
4.3.5.	Activity Diagram.....	40
4.3.6.	Class Diagram.....	44
4.3.7.	Entity Relation Diagram .....	44

4.3.8. Koding.....	45
<b>BAB 5. HASIL DAN PEMBAHASAN.....</b>	<b>53</b>
5.1. Implementasi <i>K-Nearest Neighbor</i> untuk Penentuan Status Gizi Balita .....	53
5.2. Implementasi sistem.....	58
5.3. Pengujian sistem .....	68
5.3.1. Pengujian <i>White Box</i> .....	68
5.3.2. Pengujian <i>Black Box</i> .....	78
5.4. Pengujian data untuk menentukan akurasi data yang optimal dengan nilai K yang berbeda.....	79
<b>BAB 6. PENUTUP .....</b>	<b>80</b>
6.1. Kesimpulan .....	80
6.2. Saran .....	80
<b>DAFTAR PUSTAKA .....</b>	<b>81</b>
<b>LAMPIRAN.....</b>	<b>83</b>

## DAFTAR TABEL

	<b>Halaman</b>
Tabel 4.1 Data Tinggi/Panjang Badan Normalisasi AnakLaki-laki.....	26
Tabel 4.2 Definisi Aktor .....	30
Tabel 4.3 Deskripsi <i>Usecase</i> .....	30
Tabel 4.4 Skenario Login.....	31
Tabel 4.5 Skenario Data <i>Training</i> .....	32
Tabel 4.6 Skenario Cek Gizi Balita.....	35
Tabel 4.7 Potongan kode program pada fitur data <i>training</i> (controller/bos) .....	46
Tabel 4.8 potongan kode program pada fitur cek gizi(controller/user) .....	46
Tabel 4.9 Potongan kode program pada Model .....	47
Tabel 5.1 hasil hitung jarak data uji dengan data training .....	54
Tabel 5.2 Hasil pengurutan sesuai jarak terkecil hingga terjauh .....	56
Tabel 5.3 <i>Test case</i> Pengujian <i>Function</i> hitung_dt() jalur 1 .....	69
Tabel 5.4 <i>Test case</i> Pengujian <i>Function</i> hitung_dt() jalur 2 .....	70
Tabel 5.5 <i>Test case</i> Pengujian <i>Function</i> hitung_du().....	71
Tabel 5.6 <i>Test case</i> Pengujian <i>Function</i> normalisasi() jalur 1 .....	74
Tabel 5.7 <i>Test case</i> Pengujian <i>Function</i> normalisasi() jalur 2 .....	74
Tabel 5.8 <i>Test case</i> Pengujian <i>Function</i> normalisasi() jalur 3 .....	74
Tabel 5.9 <i>Test case</i> Pengujian <i>Function</i> hitung() jalur 1 .....	77
Tabel 5.10 <i>Test case</i> Pengujian <i>Function</i> hitung() jalur 2 .....	77
Tabel 5.11 <i>Test case</i> Pengujian <i>Function</i> hitung() jalur 3 .....	77
Tabel 5.12 <i>Test case</i> Pengujian <i>Function</i> hitung() jalur 4 .....	77
Tabel 5.13 <i>Test case</i> Pengujian <i>Function</i> hitung() jalur 5 .....	77
Tabel 5.14 <i>Test case</i> Pengujian <i>Function</i> hitung() jalur 6 .....	78
Tabel 5.15 <i>Test case</i> Pengujian <i>Function</i> hitung() jalur 7 .....	78
Tabel 5.16 <i>Test case</i> Pengujian <i>Function</i> hitung() jalur 8 .....	78
Tabel 5.17 <i>Test case</i> Pengujian <i>Function</i> hitung() jalur 9 .....	78
Tabel 5.18 <i>Test case</i> Pengujian <i>Function</i> hitung() jalur 10 .....	78

Tabel 5.19 hasil pengujian pada data anak laki-laki ..... 79

Tabel 5.20 Hasil pengujian pada data anak perempuan..... 79





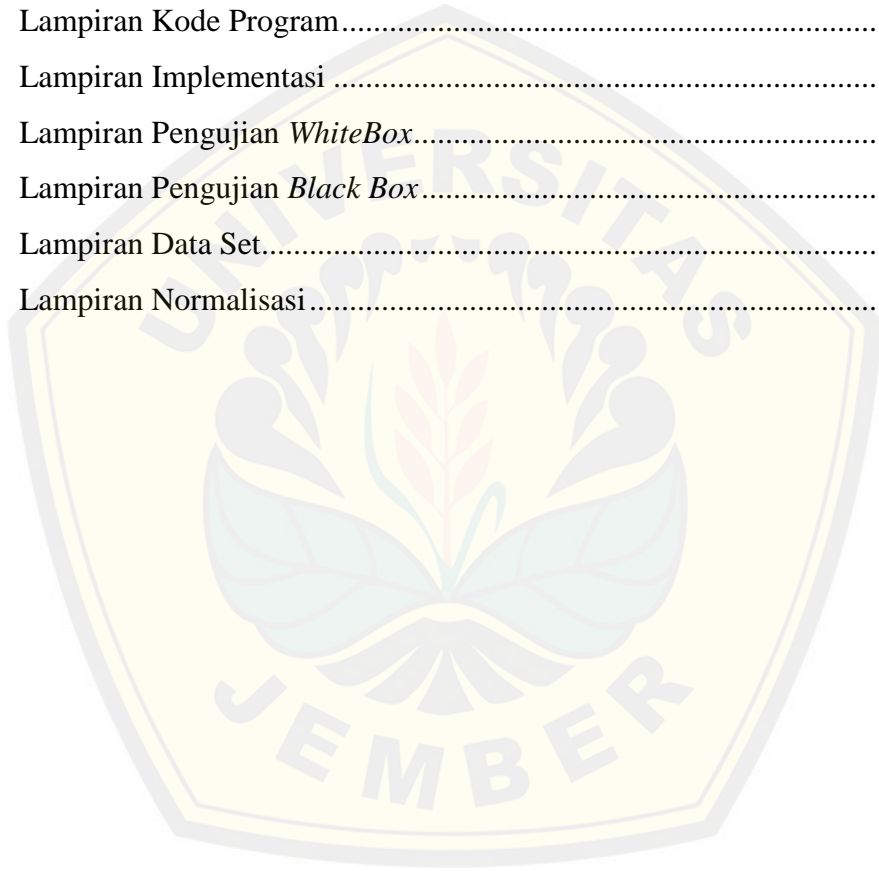
DAFTAR GAMBAR

	<b>Halaman</b>
Gambar 2.1 Jenis Ukuran Antropometri Gizi .....	9
Gambar 2.2 Model Waterfall .....	14
Gambar 3.1 alur penelitian .....	16
Gambar 3.2 Flowchart Penereapan K-Nearest Neighbor.....	19
Gambar 4.1 <i>Bussines Process</i> Sistem Pendukung Keputusan Status Gizi Balita .....	28
Gambar 4.2 <i>Usecase Diagram</i> Sistem Pendukung Keputusan Status gizi balita .....	29
Gambar 4.3 <i>Sequence Diagram Login</i> .....	38
Gambar 4.4 <i>Sequence Diagram Data Training</i> .....	39
Gambar 4.5 <i>Sequence Diagram Cek Gizi</i> .....	40
Gambar 4.6 <i>Activity Diagram Login</i> .....	41
Gambar 4.7 <i>Activity Diagram Data Training</i> .....	42
Gambar 4.8 <i>Activity Diagram Cek Gizi</i> .....	43
Gambar 4.9 <i>Class Diagram</i> Sistem Pendukung Keputusan Status Gizi balita.....	44
Gambar 4.10 <i>Entity Relation Diagram</i> Sistem Pendukung Keputusan Gizi Balita ....	45
Gambar 5.1 <i>Form login</i> .....	58
Gambar 5.2 halaman menu utama <i>admin</i> .....	59
Gambar 5.3 halaman menu utama <i>user</i> .....	60
Gambar 5.4 halaman menu data <i>training admin</i> .....	61
Gambar 5.5 form tambah data <i>training</i> .....	62
Gambar 5.6 Form <i>Data Training</i> yang telah terisi .....	62
Gambar 5.7 <i>Alert</i> Data berhasil diinputkan .....	62
Gambar 5.8 Input Data <i>Training</i> gagal .....	63
Gambar 5.9 <i>Edit Data Training</i> .....	63
Gambar 5.10 Hapus Data <i>Training</i> .....	64
Gambar 5.11 halaman menu cek gizi <i>user</i> .....	64
Gambar 5.12 Form Cek Gizi .....	65
Gambar 5.13 Warning field belum terisi .....	65

Gambar 5.14 Hasil perhitungan normalisasi data uji dan data <i>training</i> .....	66
Gambar 5.15 Hasil perhitungan jarak antara data uji dan data <i>training</i> menggunakan K-NN.....	66
Gambar 5.16 Hasil hitung K-NN yang telah di- <i>sorting</i> dan ditandai sesuai K.....	67
Gambar 5.17 Hasil status gizi .....	67
Gambar 5.18 Potongan <i>Listing Program</i> pada <i>function</i> hitung_dt .....	68
Gambar 5.19 Diagram alir <i>function</i> hitung_dt .....	69
Gambar 5.20 Grafik alir <i>function</i> hitung_dt.....	69
Gambar 5.21 Potongan <i>Listing program</i> pada <i>function</i> hitung_du.....	70
Gambar 5.22 Diagram alir <i>function</i> hitung_du .....	71
Gambar 5.23 Grafik alir <i>function</i> hitung_du.....	71
Gambar 5.24 Potongan <i>listing program</i> pada <i>function</i> normalisasi.....	72
Gambar 5.25 Diagram alir pada <i>function</i> normalisasi .....	73
Gambar 5.26 Grafik alir pada <i>function</i> normalisasi.....	73
Gambar 5.27 Potongan <i>listing program</i> pada <i>function</i> hitung.....	75
Gambar 5.28 Diagram alir <i>function</i> hitung .....	76
Gambar 5.29 Grafik alir pada <i>function</i> hitung .....	76

**DAFTAR LAMPIRAN**

	<b>Halaman</b>
A. Lampiran Perancangan.....	83
A. 1. Skenario diagram .....	83
A. 2. <i>Sequence Diagram</i> .....	92
A. 3. <i>Activity Diagram</i> .....	95
B. Lampiran Kode Program.....	98
C. Lampiran Implementasi .....	103
D. Lampiran Pengujian <i>WhiteBox</i> .....	107
E. Lampiran Pengujian <i>Black Box</i> .....	119
F. Lampiran Data Set.....	134
G. Lampiran Normalisasi.....	137



## BAB 1. PENDAHULUAN

### 1.1. Latar Belakang

Laporan PBB dalam kurun waktu 5 tahun terakhir menggarisbawahi bahwa 870 juta orang di dunia masih kekurangan gizi, sementara 1,4 juta orang lain mengalami obesitas dan penyakit terkait. Lebih dari 90 persen anak-anak yang terancam itu berasal dari Afrika dan Asia. "Lebih dari sepertiga kematian dalam kelompok umur itu mengidap malnutrisi," demikian laporan UNICEF pada VIVAnews 12 November 2009.

Gizi buruk, terutama pertumbuhan yang terhambat, merupakan sebuah masalah kesehatan masyarakat yang utama di Indonesia. Gizi buruk bagi seseorang akan mengakibatkan timbulnya suatu penyakit. Seperti misalnya pada gizi lebih bisa mengakibatkan gagal jantung, hipertensi, diabetes. Sedangkan pada gizi kurang dapat mengakibatkan buta senja, gondok, anemia, dan keratinasi sel.

Umumnya untuk mengecek kurang atau lebihnya gizi pada diri seseorang perlu mendatangi ahli kesehatan lalu melakukan beberapa tahap tes laboratorium. Akan tetapi, dengan kesibukan yang dimiliki serta tidak terlalu memanfaatkan teknologi komputer atau *smartphone* yang mereka gunakan membuat mereka mengabaikan status gizi pada dirinya. Perhitungan standar untuk perhitungan gizi biasanya menggunakan parameter tinggi badan dan berat badan. Dua parameter itu masih tergolong kurang, karena dalam buku Penilaian Status gizi dipaparkan bahwa diperlukan berbagai jenis parameter. Parameter tersebut antara lain: umur berat badan, tinggi badan, lingkaran lengan atas, lingkaran kepala, lingkaran dada, dan jaringan lunak.

Para ahli komputer merancang sebuah Sistem Pendukung Keputusan (SPK) untuk membantu para ahli suatu bidang untuk mengambil suatu keputusan sehingga mengurangi resiko kesalahan yang ada. Penelitian menggunakan metode *K-Nearest Neighbor* (K-NN) merupakan penelitian yang menggunakan data latih untuk mencari

jarak terdekat sebanyak  $k$  yang ditentukan. Data latih tersebut berkaitan dengan pengukuran status gizi lampau.

Telah dilakukan penelitian pada penentuan status gizi menggunakan *Pocket-PC* oleh Tedy Rismawan (2008). Dimana penentuan status gizi menggunakan berat badan, tinggi badan, lingkar lengan atas, lingkar kepala, lingkar dada dan lapisan lemak bawah kulit. Penentuan nilai  $k$  tidak ditetapkan dari awal menggunakan cross validation sehingga penarikan jarak terdekat masih belum valid. Aplikasi yang dibangun hanya sekedar mengecek tidak ada tampilan data simpan dari hasil pengujian sebelumnya. Kemudian penelitian berikutnya pada pemilihan menu makanan sehat dan bergizi yang diulas oleh Moch. Noor Afandie (2014) penentuan nilai  $k$  terlalu sedikit yaitu  $k=2$ . Pada dasarnya penetapan nilai  $k$  tidak bisa terlalu sedikit karena kurangnya kevalidan data dan nilai  $k$  terlalu besar akan mengurangi efek noise pada klasifikasi.

Berdasarkan uraian di atas diperlukan sebuah sistem yang dapat melakukan perhitungan status gizi dengan berbagai parameter. Untuk membuat sistem tersebut mendapatkan hasil yang lebih akurat maka dibuatlah sistem berbasis web berdasarkan perhitungan jarak terdekat dengan penarikan kesimpulan menggunakan nilai  $K$  untuk menentukan status gizi.

## 1.2. Rumusan Masalah

Berdasarkan permasalahan yang telah diuraikan di atas, maka rumusan masalah dalam penelitian ini adalah sebagai berikut:

1. Bagaimana menerapkan metode *K-Nearest Neighbor* pada sistem penunjang keputusan status gizi?
2. Bagaimana merancang dan membangun sistem penunjang keputusan untuk menentukan status gizi?
3. Bagaimana hasil pengujian terhadap data untuk menentukan akurasi data yang optimal dengan nilai  $K$  yang berbeda?

### **1.3. Tujuan dan Manfaat**

#### 1.3.1. Tujuan

Tujuan dari penelitian ini adalah :

1. Mengimplementasikan metode K-NN yang dapat menghitung serta menentukan status gizi balita
2. Membandingkan hasil status gizi menggunakan dengan nilai K yang berbeda dalam pengujian akurasi metode KNN
3. Merancang dan membangun sebuah Sistem Penunjang Keputusan Status Gizi

#### 1.3.2. Manfaat

Manfaat dari penelitian ini :

1. Bagi penulis  
Menambah pengetahuan dan pengalaman dalam mengaplikasikan teori yang telah diperoleh di bangku kuliah dan sebagai wahana untuk memperoleh pengetahuan baru dalam bidang penelitian dan penulisan karya ilmiah.
2. Bagi Objek Penelitian  
Diharapkan dapat membantu instansi kesehatan dalam penghitungan dan penentuan status gizi yang optimal untuk menghindari balita kegemukan dan/atau kekurangan gizi
3. Bagi peneliti lain  
Hasil penelitian ini dapat menambah referensi bagi peneliti dalam melakukan penelitian yang sama.

### **1.4. Batasan Masalah**

Batasan masalah yang ada dalam penelitian ini adalah sebagai berikut :

1. Sistem yang dibangun merupakan Sistem Penunjang Keputusan Status Gizi yang menggunakan parameter jenis kelamin, umur, berat badan, tinggi badan, lingkaran lengan atas, dan lingkaran kepala.

2. Sampel dari penelitian ini adalah bayi yang berumur di bawah lima tahun yang berada di Jalan Jawa VII RT. 01 dan 02 RW. 024 lingkungan Tegal Boto Lor.
3. Sistem informasi berbasis web.

### 1.5. Sistematika Penulisan Skripsi

Sistematika penulisan dalam penyusunan tugas akhir ini adalah:

1. **Pendahuluan**  
Bab ini terdiri dari latar belakang, rumusan masalah, tujuan dan manfaat, batasan masalah dan sistematika penulisan.
2. **Tinjauan Pustaka**  
Bab ini berisi tentang teori-teori, materi dan informasi yang digunakan dalam penelitian. Dimulai dari kajian pustaka mengenai sistem penunjang keputusan, gizi, status gizi, *datamining*, *K-Nearest Neighbor*, model *waterfall*.
3. **Metodologi Penelitian**  
Bab ini menguraikan tentang metode yang akan digunakan selama penelitian. Meliputi pendekatan penelitian, studi literatur, sumber data penelitian, dan tahapan analisis hingga model perancangan sistem.
4. **Analisis dan Perancangan Sistem**  
Bab ini menguraikan tentang analisis dan perancangan sistem informasi yang akan dibangun.
5. **Hasil dan Pembahasan**  
Bab ini menjelaskan tentang hasil dan pembahasan dari penelitian yang telah dilakukan.
6. **Penutup**  
Bab ini berisi kesimpulan dari penelitian yang telah dilakukan dan saran untuk penelitian selanjutnya

## BAB 2. TINJAUAN PUSTAKA

### 2.1. Sistem Penunjang Keputusan

Sistem penunjang keputusan menurut Turban & Aronson (1998) didefinisikan sebagai sistem yang digunakan untuk mendukung dan membantu pihak manajemen melakukan pengambilan keputusan pada kondisi semi terstruktur dan tidak terstruktur. Sistem penunjang keputusan merupakan suatu sistem yang membantu mengambil keputusan berbasis komputer untuk memecahkan masalah dalam pengambilan keputusan yang memanfaatkan data dan model tertentu. Sistem Penunjang Keputusan atau yang dikenal dengan SPK, dirancang untuk membantu pengambil keputusan dalam memecahkan masalah. SPK dirancang sedemikian rupa sehingga dapat digunakan atau dioperasikan dengan mudah oleh orang yang tidak memiliki dasar kemampuan pengoperasian komputer yang tinggi dan bersifat alternatif, serta SPK dirancang dengan menekankan pada aspek kemampuan adaptasi yang tinggi.

SPK bertujuan untuk menyediakan informasi, membimbing, memberikan prediksi serta mengarahkan kepada pengguna informasi agar dapat melakukan pengambilan keputusan dengan lebih baik. Peter G.W. Keen dengan Scott Morton (1978) telah mendefinisikan tiga tujuan yang harus dicapai oleh sistem pendukung keputusan, yaitu:

1. Sistem harus dapat membantu manajer dalam membuat keputusan guna memecahkan masalah semi terstruktur.
  2. Sistem harus dapat mendukung manajer, bukan mencoba menggantikannya.
  3. Sistem harus dapat meningkatkan efektivitas pengambilan keputusan manajer.
- Pada dasarnya konsep *Decision Support System* hanyalah sebatas pada kegiatan membantu para manajer melakukan penilaian serta menggantikan posisi dan peran manajer.

### 2.2. Gizi

“Gizi merupakan suatu proses organisme menggunakan makanan yang dikonsumsi secara normal melalui proses digesti, absorpsi, transportasi,



penyimpanan, metabolisme, dan pengeluaran zat-zat yang tidak digunakan untuk mempertahankan kehidupan, pertumbuhan dan fungsi normal dari organ-organ, serta menghasilkan energi” (Deswani Idrus dan Gatot Kunanto,1990).Selain gizi dari makanan air mineral sebanyak delapan gelas per hari diperlukan bagi kehidupan untuk menjaga kelangsungan hidup yang sehat dan aktif.

Gizi tak hanya mampu untuk dapat menjaga kelangsungan hidup, melainkan menjaga kondisi tubuh agar tetap sehat. Sesuai paparan Djiteng Roedjito (1989) dalam buku Kajian Penelitian Gizi, gizi itu kita perlukan untuk menjamin kelangsungan hidup, menjalankan alat-alat tubuh, pertumbuhan (pada bayi dan anak) dan pergantian bagian-bagian badan yang aus (kulit, darah, dan lain-lain). Gizi itu kita terima secara teratur dalam ragam mutu dan jumlah yang cukup agar kita tetap sehat, bergairah dan kuat bekerja.

Manusia sangat membutuhkan gizi untuk hidupnya. Tapi pada kenyataannya banyak yang belum mengerti seberapa besar nilai gizi yang dibutuhkan oleh tubuh. Padahal setiap kegiatan dalam tubuh membutuhkan asupan gizi yang tepat. Dengan asupan gizi yang tepat maka segala proses yang ada dalam tubuh kita tidak akan kekurangan bahan bakar. Selain itu tidak hanya gizi saja yang akan membuat kita sehat tapi juga dibutuhkan olah raga yang teratur agar asupan gizi dalam tubuh kita bisa diproses secara maksimal.

### **2.3. Status Gizi**

Gizi merupakan salah satu faktor penting yang menentukan tingkat kesehatan dan keserasian antara perkembangan fisik dan perkembangan mental. Tingkat gizi seseorang dalam suatu masa bukan saja ditentukan oleh konsumsi zat gizi pada masa lampau, bahkan jauh sebelum masa itu (Budiyanto, 2002).Tingkat keadaan gizi normal tercapai bila kebutuhan zat gizi optimal terpenuhi dan seimbang. Optimal dan seimbang tidak terbatas hanya terkait dengan kesehatan tubuh melainkan ketersediaan

energi, fungsi membangun dan memelihara jaringan tubuh, perkembangan otak, kemampuan belajar dan produktivitas kerja.

Pentingnya sebuah gizi menjadikan suatu status gizi yang dibutuhkan oleh pakar sehingga dapat diketahui status gizi seseorang seperti apa yang diperolehnya. Status gizi adalah keadaan tubuh yang merupakan hasil akhir dari keseimbangan antara zat gizi yang masuk ke dalam tubuh dan utilisasinya (Gibson, 1990). Keadaan keseimbangan tubuh tersebut digambarkan menjadi suatu refleksi. Menurut Euis Sunarti(2004) gambaran tersebut sebagai refleksi dari konsumsi pangan dan penggunaannya oleh tubuh Terdapat gambaran ekspresi dari keadaan keseimbangan tubuh yang dibedakan atas status gizi buruk, gizi kurang, gizi baik, dan gizi lebih.

Saat ini gizi bisa dinilai dari statusnya, status gizi bisa menjelaskan ukuran mengenai kondisi tubuh seseorang. Dengan diketahuinya status gizi maka kita bisa meningkatkan ataupun menjaga status gizi dengan tepat. Kebanyakan orang kurang peduli dengan status gizi pada diri mereka. Padahal status gizi bisa menjadi acuan untuk menjadikan tubuh agar lebih sehat. Status gizi yang sudah diketahui akan memudahkan kita mengatur pola makan maupun pola olahraga yang tepat. Sehingga akan menjadikan tubuh kita sehat secara efektif dan efisien.

### 2.3.1. Macam- Macam Status Gizi

#### 1. Gizi Buruk

Kondisi kurang gizi yang disebabkan rendahnya konsumsi energi dan protein (KEP) dalam makanan sehari-hari seperti *marasmus*, *kwashiorkor*, dan *marasmic kwashiorkor*.

#### 2. Gizi Kurang

Suatu kondisi kekurangan gizi akibat jumlah kandungan mikro atau makronutrien yang tidak tercukupi

#### 3. Gizi Baik (Normal)

Keadaan gizi seseorang dalam keadaan seimbang untuk asupan makanan yang dibutuhkan oleh tubuh.

#### 4. Gizi Lebih (Kegemukan)

Kegemukan merupakan gangguan nutrisi yang sudah umum. Kegemukan disini termasuk juga *overweight* dan obesitas. *Overweight* merupakan penimbunan lemak berlebihan pada jaringan subkutan atau jaringan lainnya. Sedangkan obesitas merupakan kondisi ini terjadi bila asupan secara merata pada seluruh jaringan dan melebihi energi yang dibutuhkan.

#### 2.3.2.

##### Penilaian Status Gizi

Penilaian status gizi terbagi dalam pengukuran langsung dan tidak langsung. Dalam penelitian ini menggunakan salah satu penilaian pengukuran tidak langsung yaitu antropometri. “Salah satu cara pemantauan status gizi adalah menggunakan indeks massa tubuh” (Supriasa, 2001). Indeks massa tubuh (IMT) merupakan pengukuran dari hasil banding antara berat badan dalam satuan kilogram dengan kuadrat tinggi badan dalam satuan meter yang sesuai rumus yang tertera pada persamaan 2.1 (Depkes RI):

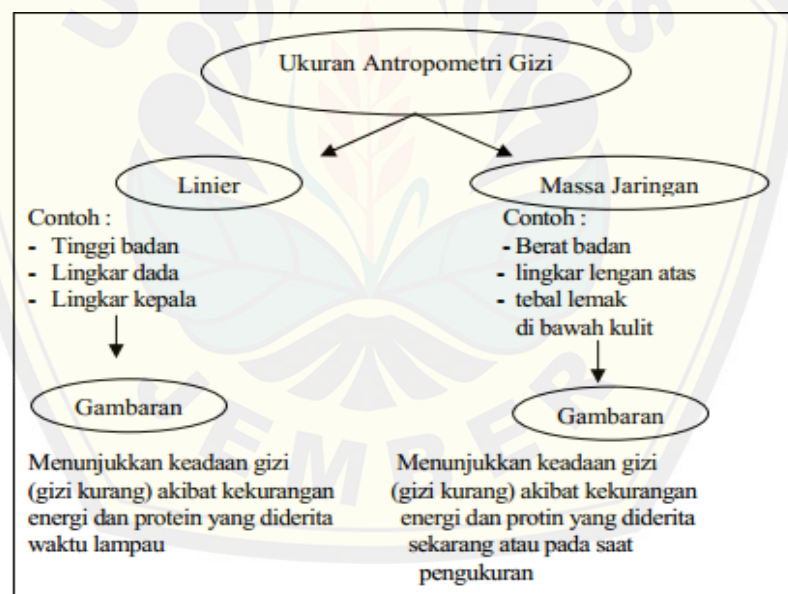
$$IMT = \frac{\text{Berat Badan (kg)}}{(\text{Tinggi Badan (m)})^2} \quad (2.3)$$

Dimana tinggi badan dan berat badan adalah dua parameter yang biasa digunakan untuk pengukuran status gizi. Akan tetapi penggunaan IMT hanya berlaku terhadap pengukuran orang dewasa, 6655 untuk kasus bayi, anak, remaja, ibu hamil, dan olahragawan menurut Supriasa dalam bukunya menjelaskan bahwa IMT tidak dapat diterapkan sehingga untuk pengukuran pada bayi dan anak masih memerlukan berbagai jenis parameter. Parameter tersebut antara lain: umur berat badan, tinggi badan, lingkaran lengan atas, lingkaran kepala, lingkaran dada, dan jaringan lunak.

Penelitian ini tidak memakai jaringan lunak karena secara umum belum tentu memiliki alat ukur untuk jaringan lunak. Membutuhkan alat ukur khusus yang biasanya ada di bidang kesehatan. Sedangkan lingkaran dada juga tidak digunakan karena

biasanya pada dilakukan hanya pada anak 2 sampai 3 tahun. Sehingga parameter yang digunakan dalam penelitian ini adalah jenis kelamin, usia, tinggi badan, berat badan, lingkaran lengan atas, serta lingkaran kepala karena masih tergolong mudah dalam pengukurannya.

Antropometri merupakan salah satu penilaian status gizi secara langsung. Secara umum antropometri artinya ukuran tubuh manusia. Antropometri gizi berhubungan dengan berbagai pengukuran dimensi tubuh dari berbagai tingkat umur dan tingkat gizi diukur dari berbagai jenis ukuran tubuh diantaranya, berat badan, tinggi badan, lingkaran lengan atas, dan tebal lemak dibawah kulit (Supriasa, 2002). Gambar 2.1 menunjukkan jenis ukuran antropometri gizi dan gambaran yang diberikan dari jenis ukuran tersebut.



Gambar 2.1 Jenis Ukuran Antropometri Gizi  
Sumber : I dewa Nyoman Supriasa, dkk., (2002:35)

Pengukuran antropometri memiliki beberapa keuntungan dan kelebihan, yaitu mampu menyediakan informasi mengenai riwayat gizi masa lalu sesuai gambaran ukuran antropometri gizi, yang tidak dapat diperoleh dengan bukti yang sama melalui metode pengukuran lainnya. Pengukuran ini dapat dilakukan dengan relatif cepat,

mudah, dan reliable menggunakan peralatan-peralatan yang *portable*, tersedianya metode-metode yang terstandardisasi, dan digunakannya peralatan yang terkaliberasi. Untuk membantu dalam menginterpretasi data antropometrik, pengukuran umumnya dinyatakan sebagai suatu indeks, seperti tinggi badan menurut umur (Gibson, 2005).

### 2.3.3. Jenis Parameter

#### 1. Umur

Umur sangat memegang peranan dalam penentuan status gizi, kesalahan dalam penentuan akan menyebabkan interpretasi status gizi yang salah. Pada perhitungan yang menggunakan indeks antropometri hasil penimbangan berat badan maupun tinggi badan yang akurat akan menjadi tidak berarti bila tidak disertai dengan penentuan umur yang tepat. Biasanya kesalahan muncul adanya kecenderungan untuk memilih angka yang mudah seperti 1 tahun; 1,5 tahun; 2 tahun. Oleh sebab itu penentuan umur anak perlu dihitung dengan cermat dengan pembulatan. Ketentuannya adalah 1 tahun adalah 12 bulan, 1 bulan adalah 30 hari. Jadi perhitungan umur adalah dalam bulan penuh, artinya sisa umur dalam hari tidak diperhitungkan (Depkes, 2004).

#### 2. Jenis Kelamin

Jenis kelamin merupakan faktor internal yang menentukan kebutuhan gizi sehingga terdapat hubungan antara jenis kelamin dengan status gizi (Apriadi 1986). Kebutuhan gizi yang berbeda ini disebabkan aktivitas anak laki-laki lebih tinggi dibandingkan dengan anak perempuan sehingga membutuhkan gizi yang tinggi. Menurut Almatsier (2005), tingkat kebutuhan pada anak laki-laki lebih banyak jika dibandingkan dengan perempuan. Begitu juga dengan kebutuhan energi, sehingga laki-laki mempunyai peluang untuk menderita KEP yang lebih tinggi daripada perempuan apabila kebutuhan akan protein dan energinya tidak terpenuhi dengan baik.

#### 3. Berat Badan

Berat badan merupakan salah satu ukuran yang memberikan gambaran massa jaringan, termasuk cairan tubuh. Berat badan sangat peka terhadap perubahan yang mendadak baik karena penyakit infeksi maupun konsumsi makanan yang menurun.

#### 4. Tinggi Badan

Tinggi badan memberikan gambaran fungsi pertumbuhan yang dilihat dari keadaan kurus kering dan kecil pendek. Tinggi badan sangat baik untuk melihat keadaan gizi masa lalu terutama yang berkaitan dengan keadaan berat badan lahir rendah dan kurang gizi pada masa balita.

#### 5. Lingkar Lengan Atas

Lingkar lengan atas memberikan gambaran tentang keadaan jaringan otot dan lapisan lemak bawah kulit. Lingkar lengan atas mencerminkan cadangan energi yang mencerminkan pada Kekurangan Energi Protein (KEP) pada balita.

#### 6. Lingkar Kepala

Lingkar kepala adalah standar prosedur dalam ilmu kedokteran anak secara praktis, biasanya untuk memeriksa keadaan patologi dari besarnya kepala atau peningkatan ukuran kepala. Contoh: hidrosefalus dan mikrosefalus.

### 2.4. K – Nearest Neighbor

Klasifikasi merupakan proses untuk menyatakan suatu objek ke dalam salah satu kategori data mining. Klasifikasi data terdiri dari 2 langkah proses. Pertama adalah *learning* (fase *training*), dimana algoritma klasifikasi dibuat untuk menganalisa data *training* lalu direpresentasikan dalam bentuk *rule* klasifikasi. Proses kedua adalah klasifikasi, dimana data tes digunakan untuk memperkirakan akurasi dari *rule* klasifikasi (Han, 2006). Komponen klasifikasi terdiri dari kelas, prediksi, data latih (*training dataset*), data uji. Salah satu algoritma yang sering digunakan untuk pengklasifikasian satu prediksi data baru adalah *K-Nearest Neighbor* (K-NN).

K-NN termasuk kelompok *instance-based learning*. Algoritma ini juga merupakan salah satu teknik *lazy learning*. K-NN dilakukan dengan mencari

kelompok  $k$  objek dalam data *training* yang paling dekat (mirip) dengan objek pada data baru atau data testing (Wu, 2009). Algoritma sederhana ini bekerja sesuai jarak terpendek dari data uji ke data latih untuk menentukan K-NN. Data latih terdiri dari  $n$  atribut dan nilai  $k$  untuk menentukan jarak terdekatnya. Nilai  $k$  yang tinggi akan mengurangi efek *noise* pada klasifikasi, tetapi membuat batasan antara setiap klasifikasi menjadi semakin kabur. Ada banyak cara untuk mengukur jarak kedekatan antara data baru dengan data lama (data *training*), diantaranya *Euclidean Distance* dan *Manhattan Distance (city block distance)*, yang paling sering digunakan adalah euclidean distance (Bramer, 2007). *Euclidean distance* merupakan rumus pencarian jarak  $d$  dari akar selisih antara data pada *record* ke- $i$  dan  $j$  sesuai rumus yang tertera pada persamaan 2.2 (Han J dan Kamber M) :

$$d(x_i, x_j) = \sqrt{\sum_r^n (\alpha_r(x_i) - \alpha_r(x_j))^2} \quad (2.2)$$

Keterangan:

$d(x_i, x_j)$  : Jarak *Euclidean (Euclidean Distance)*.

$(x_i)$  : *record* ke-  $i$

$(x_j)$  : *record* ke-  $j$

$(\alpha_r)$  : data ke- $r$

$i, j$  : 1, 2, 3, ...  $n$

Untuk atribut dengan nilai kategori, pengukuran dengan *euclidean distance* tidak cocok. Sebagai penggantinya, digunakan fungsi sebagai berikut (Larose, 2006):

$$\begin{aligned} \text{different}(a_i, b_i) &= \{ 0 \text{ jika } a_i = b_i \\ &= 1 \text{ selainnya} \end{aligned}$$

Dimana  $a_i$  dan  $b_i$  adalah nilai kategori. Jika nilai atribut antara dua *record* yang dibandingkan sama maka nilai jaraknya 0, artinya mirip, sebaliknya, jika berbeda maka nilai kedekatannya 1, artinya tidak mirip sama sekali. Misalkan atribut warna

dengan nilai merah dan merah, maka nilai kedekatannya 0, jika merah dan biru maka nilai kedekatannya 1.

Untuk mengukur jarak dari atribut yang mempunyai nilai besar, seperti atribut pendapatan, maka dilakukan normalisasi. Normalisasi bisa dilakukan dengan *min-max normalization* atau *Z-score standardization* (Larose, 2006). Jika data *training* terdiri dari atribut campuran antara numerik dan kategori, lebih baik gunakan *min-max normalization* (Larose, 2006). Untuk menormalisasi dapat menggunakan rumus *Z-score* pada 2.3 berikut :

$$V' = \frac{(v - \bar{A})}{\sigma_A} \quad (2.3)$$

Keterangan:

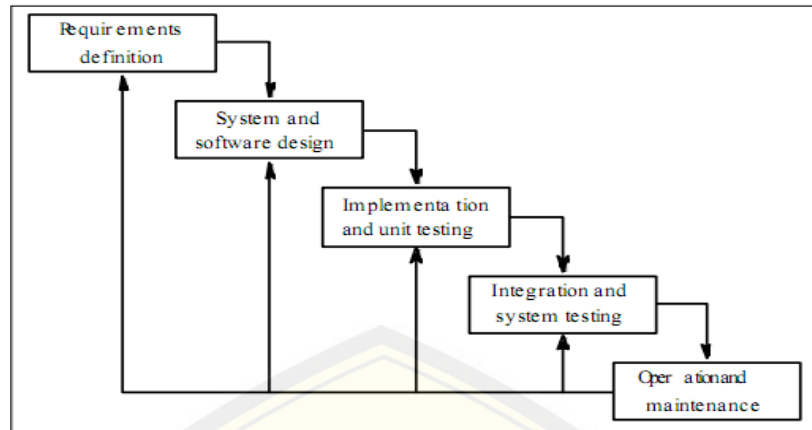
- $V'$  : hasil normalisasi *Z-score*.  
 $v$  : nilai yang akan dinormalisasikan  
 $\bar{A}$  : nilai rata-rata atribut A  
 $\sigma_A$  : standar deviasi atribut A

## 2.5. Metode Waterfall

Model waterfall merupakan model klasik yang bersifat sistematis, berurutan dalam pembangunan *software*. Menggunakan model ini dikarenakan proses-prosesnya mudah dipahami dan jelas struktur serta tahapan pengerjaan *project* yang mudah dikontrol. Terdapat lima fase dalam model *waterfall* menurut Royce (1970) pada buku *Software Engineering* Jld. 1 pada Gambar 2. 2.

1. Analisa dan definisi persyaratan, tujuan dari pembuatan sistem didefinisikan secara rinci serta difungsikan sebagai spesifikasi sistem.





Gambar 2.2 Model Waterfall

Sumber: Royce, 1970 dalam Software Engineering Jl. 1

2. Perancangan sistem dan perangkat lunak, proses perancangan sistem secara keseluruhan diidentifikasi dan deskripsi abstraksi sistem yang mendasar dan hubungannya.
3. Implementasi dan pengujian unit, tahap ini perancangan dirangkai dalam program lalu diuji pada setiap unitnya bahwa telah sesuai dengan spesifikasinya
4. Integrasi dan pengujian sistem, program diintegrasikan dan diuji sebagai sistem yang lengkap dan persyaratan sistem terpenuhi
5. Operasi dan pemeliharaan, pengecekan sistem setelah digunakan *user* dari berbagai error serta perbaikan atas implementasi dan pengembangan pelayanan sistem, sementara persyaratan-persyaratan baru ditambahkan.

### **BAB 3. METODOLOGI PENELITIAN**

Bab ini menguraikan tentang metode penelitian perancangan dan pembuatan sistem pendukung keputusan status gizi balita. Pada penelitian ini menggunakan metode pendekatan penelitian, tempat dan waktu penelitian, alur penelitian, studi literatur, data dan sumber data penelitian hingga metode yang akan digunakan dalam pembuatan sistem.

#### **3.1. Pendekatan Penelitian**

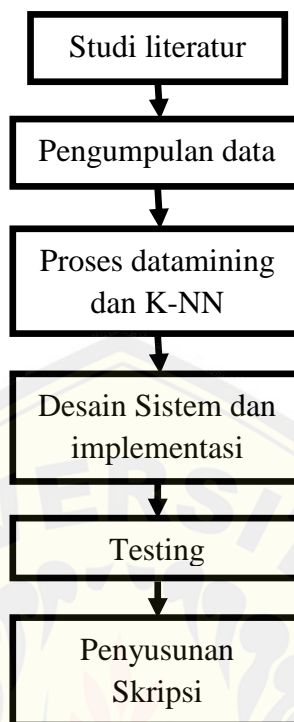
Tujuan dari penelitian ini adalah mengkaji secara mendalam tentang penerapan K- *Nearest Neighbor* pada pengecekan status gizi balita. Pendekatan yang dilakukan dalam penelitian ini yaitu peneliti mengukur dan mewawancarai informan untuk dimintai data yang akan diuji. Selanjutnya akan diformulasikan untuk mendapatkan data yang akan dijelaskan pada pembahasan penelitian.

#### **3.2. Tempat dan Waktu Penelitian**

Tempat penelitian dilakukan di beberapa desa yang ada di Jember, tepatnya di posyandu jalan jawa VII RT. 01 dan 02 RW. 024 lingkungan Tegal Boto Lor. Waktu penelitian akan dilaksanakan selama enam bulan, dimulai pada Maret hingga April 2015

#### **3.3. Alur Penelitian**

Alur Penelitian dalam penelitian ini merupakan gambaran alur penelitian yang akan digunakan sebagai acuan penelitian. Alur penelitian dalam penelitian ini terdapat pada Gambar 3.1.



Gambar 3.1 alur penelitian  
Sumber: Hasil Analisis 2015

### 3.3.1. Studi Literatur

Berdasarkan diagram alur penelitian yang terdapat pada Gambar 3.1 tahap yang pertama adalah studi literatur dibutuhkan untuk menunjang pemahaman dan pengetahuan penulis tentang materi, konsep, teori, dan metode apa yang diperlukan dalam proses pengerjaan tugas akhir ini. Studi literatur yang dilakukan meliputi penelitian - penelitian terdahulu di berbagai jurnal, buku, skripsi, dan *e-book*. Adapun hal-hal yang harus dipahami melalui studi literatur adalah beberapa penelitian terdahulu mengenai implementasi metode *K-Nearest Neighbor* pada penentuan status gizi untuk balita, parameter yang dibutuhkan dalam penentuan status gizi menggunakan *K-Nearest Neighbor*.

### 3.3.2. Tahap Pengumpulan Data

Pengumpulan data dilakukan untuk memperoleh informasi yang dibutuhkan serta tujuan penelitian terpenuhi. Data yang digunakan dalam penelitian ini berdasarkan sumber datanya meliputi data primer dan data sekunder:

#### 1. Sumber Data Primer

Sumber data primer diperoleh langsung dari balita yang melakukan pengecekan gizi, pada tempat penelitian. Adapun data yang diperoleh dari sumber data meliputi data balita dan data rekaprata-rata gizi balita yang akan dijadikan parameter, data *set*, data *training*, dan data uji untuk menentukan status gizi, serta alur kerja pada pengecekan gizi untuk menentukan fitur-fitur yang akan dibangun membangun Sistem Pendukung Keputusan Status Gizi Balita.

#### 2. Sumber Data Sekunder

Dalam penelitian ini penulis memperoleh sumber data sekunder untuk menunjang penelitian seperti literatur yang diperoleh dari buku teks, jurnal yang diperoleh dari internet, dan penelitian sebelumnya.

### 3.3.3. Teknik Pengumpulan data

Pengumpulan data diperlukan untuk mendapatkan data yang dibutuhkan dalam membangun Sistem Penunjang Keputusan Status Gizi Balita. Teknik yang digunakan pada tahap pengumpulan data dalam penelitian ini sebagai berikut:

#### a. Observasi

Pengumpulan data melalui observasi berupa melakukan pengamatan objek secara langsung. Tujuan melakukan observasi ini yaitu dapat mengetahui langsung bagaimana alur kerja yang terjadi pada objek yang diteliti. Setelah melakukan pengamatan, dilakukan pencatatan secara sistematis dari hasil pengamatan tersebut. Observasi dilakukan oleh peneliti dengan mendatangi ibu Leersia Yusi Ratnawati, S.KM., M.Kes. sebagai ahli gizi serta ibu bidan dan para praktisi yang menangani Posyandudi Jalan Jawa VII RT. 01 dan 02 RW. 024 untuk mengamati serta mencatat

data balita dan hasil rekap gizi balita yang akan digunakan sebagai data set dalam SPKstatus gizi balita menggunakan K-NN.

b. Wawancara

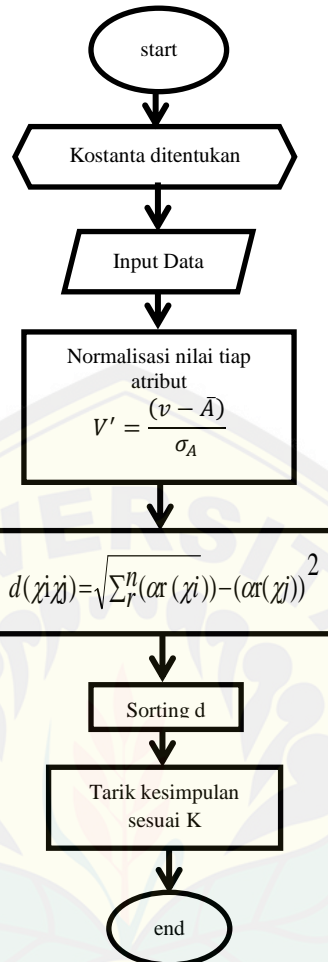
Dalam melakukan observasi, tahap wawancara langsung pun dilakukan terhadap pihak yang terkait. Pada tahap ini, pengumpulan data dengan melakukan wawancara langsung terhadap pihak yang terkait dengan objek yang diteliti untuk memperoleh data yang diinginkan. Wawancara dilakukan guna mendapatkan alur kerja pada objek yang diteliti yang akan digunakan dalam menentukan fitur-fitur yang akan dibangun.

### 3.4. **K-Nearest Neighbor untuk penentuan status gizi balita**

Sistem Pendukung Keputusan Status Gizi Balita dengan K-NN berfungsi untuk mengetahui status gizi balita secara terperinci agar tidak terjadi ketidakseimbangan antara berat badan yang selama ini hanya dipantau dengan beberapa parameter yang lainnya yang telah ditetapkan peraturan setempat. Sistem ini memiliki fitur - fitur yang diharapkan dapat membantu tenaga medis atau para ahli gizi dalam mengetahui status gizi balita. Salah satu fitur yaitu fitur pengecekan gizi balita yang dapat mengkalkulasi data untuk memeberikan status gizi pada balita.

Berdasarkan algoritma K-NN, perhitungan dapat dilakukan sebagai berikut sesuai flowchart pada Gambar3.2 :

1. Tentukan nilai K.
2. Input data yang akan diuji.
3. Normalisasikan nilai yang telah diinputkan dan data *training* sesuai rumus normalisasi Z-score.
4. Hitung jarak tiap atribut yang diinputkan dengan data *training* berdasarkan rumus *Euclidian distance*.
5. Selanjutnya data diurutkan berdasarkan jarak yang terpendek hingga terbesar
6. Ambil jarak terpendek sesuai nilai K yang ditentukan.



Gambar 3.2 Flowchart Penereapan K-Nearest Neighbor  
Sumber: Hasil Analisis 2015

### 3.5. Metode Pembuatan Sistem

Metode yang digunakan dalam pembuatan sistem ini menggunakan model *waterfall*. Model *waterfall* memiliki beberapa tahapan untuk membangun sistem informasi. Tahapan yaitu analisis kebutuhan sistem, desain sistem, implementasi sistem, pengujian sistem, dan perawatan. Tahap-tahap kegiatan pada model *waterfall* terdapat pada Gambar 2. 3.

#### 3.5.1. Analisis Sistem

Pelayanan, batasan dan tujuan sistem ditentukan melalui konsultasi dengan pakar gizi. Persyaratan ini kemudian didefinisikan secara rinci tersebut berguna untuk

membangun fitur-fitur yang akan dibuat pada sistem serta data yang dibutuhkan dalam sistem dan berfungsi sebagai spesifikasi sistem.

### 3.5.2. Desain Sistem

Proses perancangan sistem membagi persyaratan dalam sistem perangkat keras atau perangkat lunak. Sistem ini dirancang menggunakan beberapa diagram *Unified Modeling Language* (UML) yang dirancang menggunakan konsep *Object-Oriented Programming* (OOP). Diagram tersebut berfungsi untuk membantu merancang fitur-fitur dan kebutuhan fungsional dan non-fungsional sistem yang akan dikembangkan. Diagram tersebut antara lain:

#### 1. *Usecase Diagram*

Sistem pendukung keputusan status gizi balita ini akan memiliki tujuh *usecase* utama. Terdapat dua aktor yang memiliki hak akses berbeda-beda. Terdapat satu aktor yang berperan sebagai *admin* dalam sistem pendukung keputusan status gizi balita. Sedangkan aktor lainnya sebagai *user* yaitu bidan atau petugas kesehatan yang melakukan pengecekan gizi terhadap balita.

#### 2. *Activity Diagram*

Setelah membuat *usecase* diagram maka selanjutnya akan membuat skenario *usecase* dan *activity diagram*. *Activity diagram* menggambarkan skenario *usecase* dalam bentuk bagan *activity*. Terdapat tujuh *usecase* utama maka akan ada tujuh skenario *usecase* dan akan ada tujuh *activity diagram*.

#### 3. *Sequence Diagram*

*Sequence diagram* juga terdapat tujuh *sequence* dari masing-masing *usecase*. Biasanya lifeline yang digunakan adalah halaman dari *view*, *controller* dan juga *database*. Dari *sequence* dapat terlihat *action* apa saja yang ada di setiap *usecase*.

#### 4. *Class Diagram*

*Class diagram* nantinya akan digambarkan pada masing-masing *usecase* dan juga digambarkan menjadi satu dari rangkaian seluruh *class diagram* pada *usecase*. Pada *class diagram* akan tampak terlihat perancangan apa yang digunakan. Ketika menggunakan perancangan OOP maka *class diagram* akan dibagi menjadi tiga kelas diantaranya *model*, *view*, dan *controller*.

#### 5. *Entity Relationship Diagram (ERD)*

ERD nantinya akan sama seperti *class diagram* setiap *usecase* akan dibuat ERD dan setelah itu semua ERD dari setiap *usecase* akan dirangkai menjadi satu rangkaian ERD sistem pendukung keputusan status gizi balita. Dari ERD dapat dilihat relasi dari masing-masing entitas, apa hubungan antara entitas satu dengan entitas yang lainnya.

#### 3.5.3. Implementasi sistem

Pada tahap ini, akan disesuaikan dengan *desain* sistem yang telah dikerjakan pada tahap perancangan sistem. Perancangan perangkat lunak akan direalisasikan sebagai serangkaian program atau unit program. Pengujian unit melibatkan verifikasi bahwa setiap unit telah memenuhi spesifikasinya. Pada tahap ini sistem yang sudah dirancang sesuai dengan kebutuhan *user* akan dicoba implementasikan, sehingga bisa diketahui unit program mana yang sudah sesuai atau belum dengan kebutuhan dan spesifikasi dari sistem. Kegiatan yang dilakukan selama tahap implementasi antara lain:

- a.) Penulisan kode program (coding) menggunakan bahasa pemrograman PHP (Page Hyper Text Pre-Processor), CSS (Cascading Style Sheet), Javascript dan framework CodeIgniter.
- b.) Manajemen data menggunakan DBMS MySQL



#### 3.5.4. Pengujian Sistem

Pengujian sistem akan menilai apakah aplikasi yang dirancang telah sesuai dengan apa yang diharapkan, serta untuk mengevaluasi keunggulan dan kelemahan terhadap kualitas sistem. Pada unit program atau program individual yang telah diintegrasikan akan diuji sebagai sistem yang lengkap untuk menjamin bahwa persyaratan sistem telah dipenuhi. Setelah pengujian sistem, perangkat lunak dikirim kepada pengguna. Tahapan ini digunakan untuk mengetahui kinerja sistem secara keseluruhan. Sistem akan langsung dikirim pada pengguna setelah diuji. Sehingga apabila menemukan beberapa kesalahan dapat dilakukan perbaikan dalam penyempurnaan kemampuan sistem. Pengujian sistem dilakukan oleh peneliti bersama pakar gizi serta praktisi kesehatan yang ada di tempat penelitian.

*White box testing* (Hanif, 2007) adalah cara pengujian dengan melihat ke dalam modul untuk meneliti kode-kode program yang ada, dan menganalisis apakah ada kesalahan atau tidak. Pengujian *White Box* dilakukan oleh tim penguji dari *developer*, dimana tidak hanya memperhatikan masukan / keluaran (I / O) tetapi juga algoritma yang digunakan apakah sesuai dengan rancangan yang dibuat atau tidak. Pada pengujian *white-box* dilakukan pengukuran program untuk melihat tingkat kompleksitas program dengan menggunakan metode *cyclomatic complexity* dan mengevaluasi tingkat kemudahan untuk mengerti kode oleh *developer* dan *mainteners* dengan menggunakan matrik *size* yang terdiri dari *Line of Code* (LOC), *Non Comment Non Blank* (NCNB), dan *Executable Statement* (EXEC).

“*Black Box testing* adalah strategi pengujian hanya berdasarkan persyaratan dan spesifikasi. yang tidak memerlukan pengetahuan tentang jalur intern, struktur, atau implementasi dari perangkat lunak yang teruji” (S. Koirala dan S. Sheikh, 2008). Pengujian *black box* teras dilakukan oleh pengguna / *user*, dimana hanya memperhatikan masukan / keluaran (I / O) apakah sesuai dengan sistem yang dijalankan. Pengujian dilakukan oleh beberapa personalia dan *admin* Pengujian sistem

pengecekan status giziakan diambil secara acak. Uji coba *black box* berusaha untuk menemukan kesalahan dalam beberapa kategori ,diantaranya :

1. Fungsi-fungsi yang salah atau hilang
2. Kesalahan interface
3. Kesalahan dalam struktur data atau akses database eksternal
4. Kesalahan performa
5. Kesalahan inisialisasi dan terminasi.

#### 3.5.5. Pemeliharaan dan Perbaikan

Ini merupakan suatu fase siklus hidup yang paling lama. Sistem diinstal dan dipakai. Pemeliharaan mencakup koreksi dari berbagai *error* yang tidak ditemukan pada tahap-tahap terdahulu, perbaikan atas implementasi unit sistem dan pengembangan pelayanan sistem, sementara persyaratan-persyaratan baru ditambahkan. Tahap ini ditujukan sebagai *maintenance* sistem, sehingga apabila terdapat *error* akan diperbaiki dan sampai sistem bisa bekerja secara maksimal dengan kemungkinan terjadi *error* minimal.

## BAB 4. ANALISIS DAN PERANCANGAN

Pada bab ini akan dijelaskan mengenai proses perancangan untuk membuat Sistem Pendukung Keputusan Status Gizi balita. Proses perancangan sistem dimulai dari analisis kebutuhan fungsional dan non-fungsional sistem, dilanjutkan dengan pembuatan *usecase diagram*, skenario, *activity diagram*, *sequence diagram*, *class diagram* dan *entity relation diagram* (ERD) sesuai dengan tahapan model *waterfall* yang telah diuraikan pada bab sebelumnya.

### 4.1. Pengumpulan data

Pengumpulan data dilakukan agar Sistem Pendukung Keputusan Status Gizi Balita dapat dibuat sesuai dengan kebutuhan pengguna. Pengumpulan data dilakukan dengan teknik analisis dokumen berdasarkan hasil observasi dan wawancara yang didapatkan dari pakar gizi dan praktisi posyandu sebagai pengguna utama sistem.

#### 4.1.1. Data Set

Data set merupakan kumpulan atribut atau variabel dan kelas, yang nantinya akan dibagi menjadi dua macam data yaitu data *training* dan data *testing*. Data yang sudah dibagi menjadi data *training* akan digunakan untuk proses *learning* dalam K-NN. Sedangkan data yang termasuk dalam data *testing* nantinya akan digunakan dalam proses pengujian. Dalam pembuatan sistem pendukung keputusan status gizi balita pada fitur cek gizi akan menggunakan seluruh data set sebagai data *training* dan data *testing*, sedangkan atribut yang digunakan meliputi *id\_jk*, umur, BB, PB, LLA, dan LK. Semua atribut tersebut termasuk ke dalam kelas yang bernama kelas status. Untuk lebih jelasnya data set dapat dilihat pada lampiran F.

#### 4.1.2. Data Normalisasi

Untuk mengukur jarak dari atribut yang mempunyai nilai besar, seperti atribut nilai gizi, maka dilakukan normalisasi. Pada penelitian ini, data dari tiap atribut akan

dinormalisasikan. Untuk tiap nilai yang ada pada atribut tinggi badan akan dicari nilai normalisasi. Menormalisasikannya dapat menggunakan rumus Z-score 4.1 berikut :

$$V' = \frac{(v-\bar{A})}{\sigma_A} \quad (4.1)$$

Mula-mula nilai rata-rata dari jumlah data pada atribut dibagi banyaknya data pada atribut tersebut dicari dengan menggunakan rumus *Mean* 4.2 berikut :

$$\bar{A} = \frac{\sum_{i=1}^n A_i}{n} \quad (4.2)$$

Keterangan :

$\bar{A}$  = rata-rata hitung

$A_i$  = nilai sample ke-i

n = jumlah sample

$$\begin{aligned} \bar{x} = & (92.5 + 88 + 81 + 87.8 + 82 + 94.2 + 86 + 88.5 + 77.7 + 78 + 86 + 73 \\ & + 88 + 73 + 82 + 75 + 72 + 80.5 + 78 + 71 + 75 + 80 + 81 + 88 \\ & + 92 + 76.5 + 77 + 83 + 86 + 71.7 + 82.5 + 83 + 82 + 76 + 73 \\ & + 75 + 90 + 74 + 79 + 74 + 76 + 76.5 + 72 + 85.2 + 70 + 76 \\ & + 94.6 + 76 + 91.2 + 82 + 82) / 51 = 80.6745098 \end{aligned}$$

Mencari nilai simpangan baku pada persamaan 4.3 berikut:

$$\sigma_A = \sqrt{\frac{n \sum_{i=1}^n A_i^2 - (\sum_{i=1}^n A_i)^2}{n(n-1)}} \quad (4.3)$$

Keterangan:

n = jumlah data = 51

$\sum_{i=1}^n A_i$  = jumlah dari tiap nilai = 4114.4

$\sum_{i=1}^n A_i^2$  = jumlah dari tiap nilai yang dikuadratkan = 334134.8

$$\sigma_A = \sqrt{\frac{51 \times 334134.8 - 4114.4^2}{51(51 - 1)}} = 332,23$$

Lalu memasukan nilai yang telah dihitung diatas dengan nilai yang akan dinormalisasikan dengan rumus 4.1. Berikut contoh data kesatu yang akan dinormalisasikan dan contoh hasil perhitungan atribut tinggi badan pada anak laki-laki yang telah dinormalisasikan dari rumus-rumus diatas yang tersaji pada Tabel 4.1. Untuk lebih jelasnya data set dapat dilihat pada lampiran E.

$$V' = \frac{(v - \bar{A})}{\sigma_A} = \frac{(92,5 - 80,675)}{332,23} = 0.03559379$$

Tabel 4.1 Data Tinggi/Panjang Badan Normalisasi Anak Laki-laki

No.	TB	Normalisasi
1	92.5	0.03559379
2	88	0.02204915
3	81	0.0009797
4	87.8	0.02144716
5	82	0.00398962
6	94.2	0.04071066
7	86	0.01602931
8	88.5	0.02355411
9	77.7	-0.00895304
10	78	-0.00805006
11	86	0.01602931
12	73	-0.02309967
13	88	0.02204915
14	73	-0.02309967
15	82	0.00398962
16	75	-0.01707983
17	72	-0.02610959
18	80.5	-0.00052526
19	78	-0.00805006
20	71	-0.02911951

21	75	-0.01707983
22	80	-0.00203022
23	81	0.0009797
24	88	0.02204915
25	92	0.03408883
26	76.5	-0.01256495
27	77	-0.01105998
28	83	0.00699954
29	86	0.01602931
30	71.7	-0.02701257
31	82.5	0.00549458
32	83	0.00699954
33	82	0.00398962
34	76	-0.01406991
35	73	-0.02309967
36	75	-0.01707983
37	90	0.02806899
38	74	-0.02008975
39	79	-0.00504014
40	74	-0.02008975
41	76	-0.01406991
42	76.5	-0.01256495
43	72	-0.02610959
44	85.2	0.01362137
45	70	-0.03212943
46	76	-0.01406991
47	94.6	0.04191463
48	76	-0.01406991
49	91.2	0.0316809
50	82	0.00398962
51	82	0.00398962

#### 4.2. Analisis kebutuhan

Analisis kebutuhan perangkat lunak merupakan cara mengidentifikasi permasalahan yang ada untuk dicatat dan dijadikan bahan untuk mulai membangun aplikasi Sistem Pendukung Keputusan Status Gizi Balita. Analisis kebutuhan yang

dilakukan meliputi proses pengumpulan data kebutuhan fungsional, kebutuhan non-fungsional.

Kebutuhan fungsional pada Sistem Pendukung Keputusan Status Gizi balita ini antara lain:

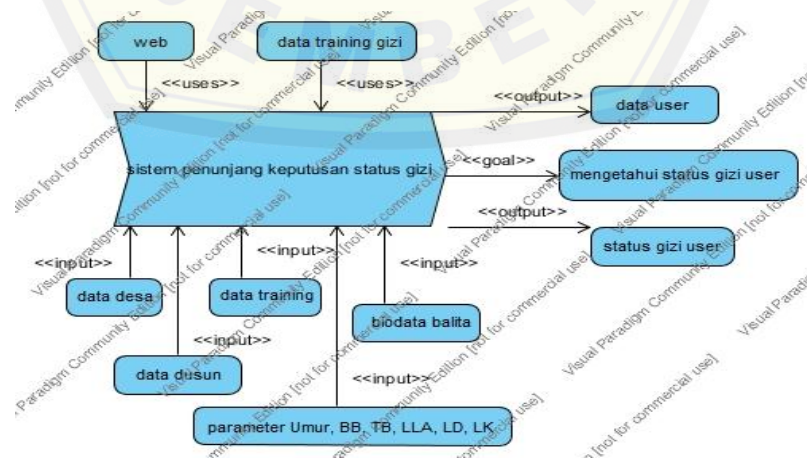
1. Sistem mampu menampilkan hasil status gizi,
2. Sistem mampu mencatat gizi tiap tiap balita,
3. Sistem mampu menampilkan rata-rata gizi perbulan, perdesa, perdesun, maupun perorangan,
4. Sistem mampu mencatat data *training*.

Kebutuhan non-fungsional pada penelitian ini adalah sistem menggunakan tampilan yang *user friendly*, sehingga memudahkan pengguna dalam mengoperasikannya

### 4.3. Desain Sistem

#### 4.3.1. Business Proses

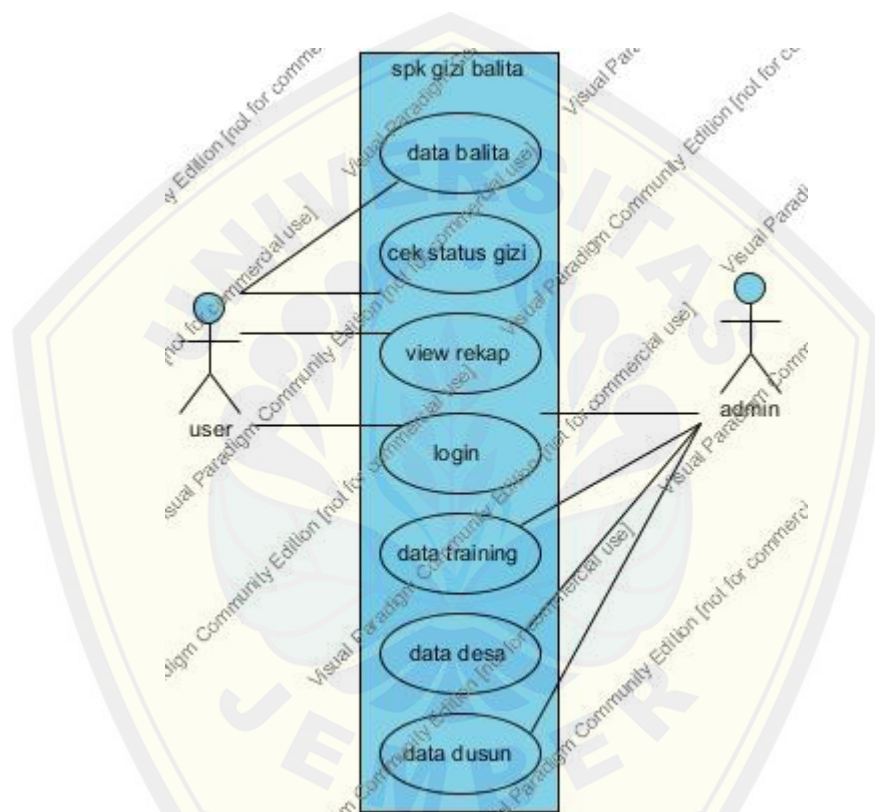
Gambar 4.1 merupakan *business process* dari sistem pendukung keputusan status gizi balita yang didalamnya terdapat *input*, *output*, dan tujuan(*goal*) dari sistem informasi tersebut.



Gambar 4.1 *Bussines Process* Sistem Pendukung Keputusan Status Gizi Balita  
Sumber: Hasil Analisis 2015

#### 4.3.2. Usecase Diagram

*Usecase diagram* menggambarkan sebuah interaksi antara satu atau lebih aktor dengan fitur-fitur sistem informasi yang akan dibuat. Gunanya untuk mengetahui fitur apa saja yang ada didalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fitur – fitur tersebut. Pada sub bab ini berisi *usecase diagram*, definisi aktor, dan definisi *usecase*.



Gambar 4.2 *Usecase Diagram* Sistem Pendukung Keputusan Status gizi balita  
Sumber: Hasil Analisis 2015

Pada *usecase diagram* pada Gambar 4.2 tersebut terdapat dua klasifikasi aktor yang dapat menggunakan sistem pendukung keputusan status gizi balita. Dua aktor tersebut diantaranya adalah *admin* dan *user* (praktisi kesehatan). Deskripsi aktor serta *usecase* dijelaskan pada tabel 4.2 dan tabel 4.3 berikut:



Tabel 4.2 Definisi Aktor

No.	Aktor	Deskripsi
1.	<i>Admin</i>	<i>Admin</i> merupakan aktor yang mengelola sistem dan manajemen data <i>user</i> dengan memiliki hak akses yaitu menginputkan, mengedit, menghapus data <i>training</i> , tempat desa dan dusun
2.	<i>User</i>	<i>User</i> merupakan aktor yang mengakses sistem dalam menginputkan, mengedit, dan menghapus data balita, mengecek status gizi, melihat <i>record</i> status gizi balita tiap bulan pada desa, dusun maupun balita itu sendiri

Tabel 4.3 Deskripsi *Usecase*

No.	<i>Usecase</i>	Deskripsi
1.	Login	<i>Usecase</i> ini digunakan untuk proses untuk melakukan <i>login</i> pengguna SPK status gizi sebelum memasuki halaman sesuai level pengguna
2.	Data Balita	<i>Usecase</i> ini digunakan <i>user</i> untuk menginputkan, mengubah dan menghapus data balita yang menggunakan SPK status gizi
3.	Cek Status Gizi	<i>Usecase</i> ini mengkalkulasi nilai-nilai dari tiap parameter untuk memberikan informasi status gizi balita
4.	View History	<i>Usecase</i> ini memberikan informasi hasil data rekaman cek status gizi balita dari total keseluruhan yang <i>user</i> kendali, seperti desa, dusun, maupun balita itu sendiri
5.	Data <i>Training</i>	<i>Usecase</i> ini berfungsi untuk menambahkan, mengubah, dan menghapus data <i>training</i> yang akan digunakan dalam pengkalkulasian cek status gizi
6.	Data Desa	<i>Usecase</i> ini berfungsi untuk menambahkan, memperbarui, dan menghapus desa yang akan menggunakan sistem ini
7.	Data Dusun	<i>Usecase</i> ini berfungsi untuk menambahkan, memperbarui, dan menghapus dusun dari desa yang akan menggunakan sistem ini

#### 4.3.3. Skenario Diagram

Menjelaskan alur cerita (skenario) pada setiap *usecase* yang telah digambarkan dalam *usecase diagram* secara lebih detail. Diagram skenario berisi nama *usecase*, deskripsi *usecase*, aktor, *Pre Condition*, *Post Condition*, skenario normal, dan skenario alternatif.

4.3.4.1. Skenario *Login*

Skenariologin merupakan penjelasan urutan reaksi antara aktor dengan sistem secara normal maupun kondisi alternatif pada fitur login. Penjelasan skenario login dapat dilihat pada tabel 4.4.

Tabel 4.4 Skenario Login

Nama	Login
Aktor	<i>User/admin</i>
Prekondisi	<i>User/ admin</i> belum bisa menggunakan fitur yang ada pada sistem
Postkondisi	<i>User/ admin</i> sudah bisa menggunakan fitur yang ada pada sistem
Event Flow	<ol style="list-style-type: none"> <li>1. <i>User/admin</i> membuka aplikasi.</li> <li>2. <i>User/admin</i> memasukkan <i>username</i> dan <i>password</i> masing-masing.</li> <li>3. Setelah memasukkan <i>username</i> dan <i>password</i> <i>user/admin</i> dapat memilih menu yang telah disediakan oleh sistem untuk masing-masing.</li> </ol>
Skenario Login	
Aktor	Sistem
1. Membuka sistem cek gizi	
	<ol style="list-style-type: none"> <li>2. Menampilkan form log in dengan field yang tersedia antara lain : <ol style="list-style-type: none"> <li>a. <i>Username</i> (input text: harus diisi)</li> <li>b. <i>Password</i> (input password: harus diisi)</li> </ol> </li> </ol>
3. Mengisi form login yang tersedia ( <i>username</i> , <i>password</i> )	
4. Klik tombol login	
	<ol style="list-style-type: none"> <li>5. Verifikasi <i>username</i> dan <i>password</i></li> <li>6. Menampilkan menu yang tersedia sesuai dengan jenis <i>user</i></li> </ol>
Skenario Alternatif Login	
<i>Username/password salah</i>	
3a. Salah memasukkan <i>Username</i> yang tidak tersedia atau <i>password</i> yang salah	
4a. Klik tombol login	

	5a. Verifikasi <i>username</i> dan <i>password</i>
	6a. Menampilkan <i>warning message</i> “ <i>username/password salah</i> ”
	7a. Menampilkan kembali halaman login
<b>Hanya mengisi salah satu field/tidak mengisi</b>	
3b. Tidak mengisi /hanya mengisi salah satu field	
4b. Klik tombol login	
	5b. Menampilkan form login dan <i>warning</i> “harap isi bidang ini”

#### 4.3.4.2. Skenario Data Training

Skenario data *training* merupakan penjelasan urutan reaksi antara *admin* dengan sistem secara normal maupun kondisi alternatif pada fitur data *training*. Penjelasan skenario data *training* dapat dilihat pada tabel 4.5.

Tabel 4.5 Skenario Data Training

Nama	Data Training
Deskripsi	<i>Admin</i> dapat menambah, meng- <i>edit</i> , dan menghapus Data Training
Aktor	<i>Admin</i>
Prekondisi	<i>Admin</i> memilih menu data <i>training</i>
Postkondisi	Data Training berhasil ditambahkan di- <i>edit</i> , maupun dihapus
<b>Skenario Data Training</b>	
Aktor	Sistem
1. Melakukan login	
	2. Menampilkan halaman <i>admin</i> dengan menu Home, Data <i>training</i> , Desa, Dusun
3. Memilih menu data <i>training</i>	
	4. Menampilkan tabel data <i>training</i> , submenu filter tampilan sesuai jenis kelamin, dan tombol tambah data <i>training</i>
<b>Skenario Input Data Training</b>	
5. Klik Tambah Data Training	
	6. Menampilkan form tambah data <i>training</i> yang berisi <i>field</i> :

	<ul style="list-style-type: none"> <li>a. Id data <i>training</i></li> <li>b. Jenis kelamin (jquery: harus dipilih)</li> <li>c. Umur (<i>input</i> angka: harus diisi)</li> <li>d. Tinggi Badan (<i>input</i> angka: harus diisi)</li> <li>e. Berat Badan (<i>input</i> angka: harus diisi)</li> <li>f. Lingkar Lengan Atas (<i>input</i> angka: harus diisi)</li> <li>g. Lingkar Kepala (<i>input</i> nomor: harus diisi)</li> <li>h. Status (jquery: harus dipilih)</li> </ul>
7. Menginputkan data <i>training</i> dengan prosedur sebagai berikut:	
<ul style="list-style-type: none"> <li>a. Pilih Jenis kelamin</li> <li>b. Isi Umur</li> <li>c. Isi Tinggi Badan</li> <li>d. Isi Berat Badan</li> <li>e. Isi Lingkar Lengan Atas</li> <li>f. Isi Lingkar Kepala</li> <li>g. Pilih Status Gizi</li> </ul>	
8. Klik <i>Submit</i>	
	9. Data disimpan ke database
	10. Sistem secara otomatis menampilkan menu data <i>training</i> yang telah diperbarui
<b>Skenario Alternatif Input Data Training</b>	
<b>Input data dengan format salah</b>	
7a. Menginputkan dengan format yang tidak sesuai	
	8a. Sistem menampilkan peringatan data yang diinputkan tidak sesuai format pada sisi field yang salah format
<b>Data tidak lengkap</b>	
7b. Tidak mengisi data dengan lengkap	
8b. Klik <i>Submit</i>	
	9b. Menampilkan kembali form input data <i>training</i> dan <i>alert</i> harap isi bidang ini
<b>Batal saat sedang menginputkan data <i>training</i></b>	
7c. Klik <i>back</i>	

	8c. Menampilkan kembali halaman menu data <i>training</i>
--	---

<b>Skenario Edit Data Training</b>	
Aktor	Sistem
5. Klik tombol <i>edit</i> pada baris data <i>training</i> yang datanya akan <i>diedit</i>	
	6. Menampilkan form <i>edit</i> data balita dengan data sebelumnya
7. Menginputkan data pada bagaian yang akan di- <i>edit</i>	
8. Klik <i>Submit</i>	
	9. Data disimpan ke database
	10. Sistem secara otomatis menampilkan menu data <i>training</i> yang telah diperbarui
<b>Skenario Alternatif Edit Data Training</b>	
<b>Edit data dengan format salah</b>	
7a. Mengeditkan dengan format yang tidak sesuai	
	8a. Sistem menampilkan peringatan data yang <i>diedit</i> kan tidak sesuai format pada sisi field yang salah format
<b>Data tidak lengkap</b>	
7b. Tidak mengisi data dengan lengkap	
8b. Klik <i>Submit</i>	
	9b. Menampilkan kembali form <i>edit</i> data <i>training</i> dan <i>alert</i> harap isi bidang ini
<b>Batal saat sedang mengeditkan data user</b>	
7c. Klik <i>back</i>	
	8c. Menampilkan kembali halaman menu bayi

<b>Skenario Hapus Data Training</b>	
Aktor	Sistem
5. Klik tombol hapus pada baris data	

<i>training</i> yang datanya akan dihapus	
	6. Menampilkan kotak <i>warning</i> “yakin?”
7. Klik ok	
	8. Data terhapus
	9. Menampilkan dan mererefresh halaman menu data <i>training</i>
<b>Skenario Alternatif Hapus data User</b>	
<b>Batal hapus</b>	
7a. Klik batal	
	8a. Menampilkan kembali menu data <i>training</i>

#### 4.3.4.3.Skenario Cek Gizi Balita

Skenario cek gizi pada balita merupakan penjelasan urutan reaksi antara *user* dengan sistem secara normal maupun kondisi alternatif pada fitur cek gizi. Penjelasan skenario login dapat dilihat pada tabel 4.6.

Tabel 4.6 Skenario Cek Gizi Balita

Nama	Cek gizi
Aktor	<i>User</i>
Prekondisi	<i>User</i> mengisi form kondisi balita
Postkondisi	<i>User</i> berhasil mengecek gizi balita dan data disimpan ke sistem
<b>Skenario Cek Status Gizi</b>	
Aktor	Sistem
1. Melakukan Login	
	2. Menampilkan halaman <i>user</i> dengan menu Baby, Cek gizi, Rekap gizi, Logout
3. Memilih menu Cek Gizi	
	4. Menampilkan tabel nama bayi, submenu filter desa dan dusun, dan tambah data bayi bila data bayi belum ada.
5. Klik tombol cek gizi pada baris nama bayi yang akan dicek	
	6. Menampilkan form data <i>user</i> yang berisi field: a. Nama balita

	<ul style="list-style-type: none"> <li>b. Tanggal cek</li> <li>c. Kondisi balita saat dicek, <i>user</i> menginputkan data balita berupa BB, TB, LLA, LK, dan K untuk penarikan kesimpulan setelah sorting</li> </ul>
7. Menginputkan kondisi balita yang dicek	
8. Klik cek	
	<ul style="list-style-type: none"> <li>9. Menampilkan proses perhitungan sebagai berikut: <ul style="list-style-type: none"> <li>a. Menampilkan nilai kondisi yang telah dinormalisasi</li> <li>b. Menampilkan data <i>training</i> yang telah dinormalisasi</li> <li>c. Menghitung</li> </ul> </li> </ul>
10. Klik K-NN	
	11. Menampilkan hasil hitung jarak terdekat dari data uji dengan tiap data <i>training</i>
12. Klik Sorting	
	<ul style="list-style-type: none"> <li>13. Menampilkan proses perhitungan sebagai berikut: <ul style="list-style-type: none"> <li>a. Mengurutkan data yang terdekat</li> <li>b. Mengambil kesimpulan sesuai k yang ditentukan</li> </ul> </li> </ul>
	14. Menampilkan data kondisi dan status gizi balita yang dicek, serta tombol cek balita lainnya atau keluar
15. Klik cek	
	16. Data disimpan ke database
	17. Sistem secara otomatis menampilkan menu cek gizi bayi
<b>Skenario Alternatif Cek Gizi</b>	
<b>Batal mengecek karena kesalahan data yang akan dicek</b>	
7. Klik <i>back</i>	
	8. Menampilkan kembali halaman cek gizi bayi
<b>Input data dengan format salah</b>	
7b. Menginputkan dengan format yang tidak sesuai	

	8b. Sistem menampilkan peringatan data yang diinputkan tidak sesuai format pada sisi field yang salah format
<b>Data tidak lengkap</b>	
7c. Tidak mengisi data dengan lengkap	
8c. Klik cek	
	9c. Menampilkan kembali form cek gizi
<b>Batal cek karena salah penginputan nilai kondisi</b>	
7a. Klik <i>back</i>	
	8a. Menampilkan form data <i>user</i> yang berisi field: a. Nama balita b. Tanggal cek c. Kondisi balita saat dicek, <i>user</i> menginputkan data balita berupa BB, TB, LLA, LK
<b>Data direkap</b>	
15a. Klik rekap	
	16a. Data disimpan ke database
	17a. Menampilkan rekap data bayi yang dipilih

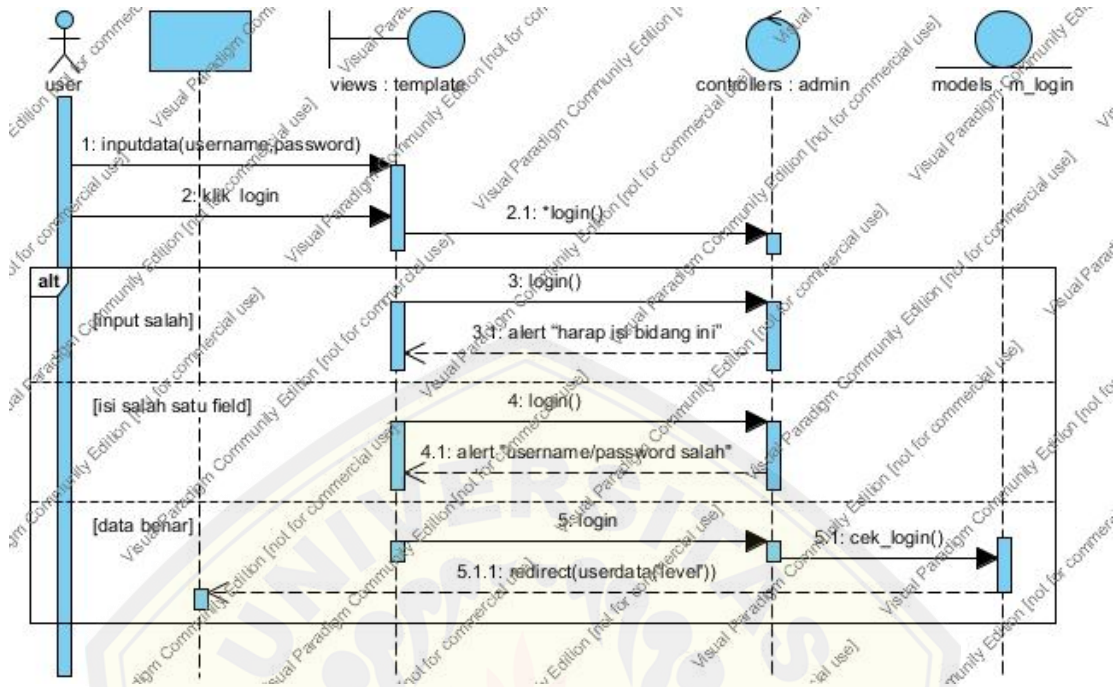
#### 4.3.4. Sequence Diagram

*Sequence diagram* menggambarkan skenario yang dimodelkan dalam aliran logika sistem dengan secara berurutan. Sub bab ini akan menggambarkan aliran logika pada sistem pendukung keputusan status gizi balita.

##### 4.3.4.1. Sequence Diagram Login

*Sequence diagram login* menggambarkan proses yang terjadi ketika aktor melakukan login. Proses yang dilakukan oleh tiap aktor hingga proses yang dilakukan dalam *database* akan digambarkan pada diagram berikut.

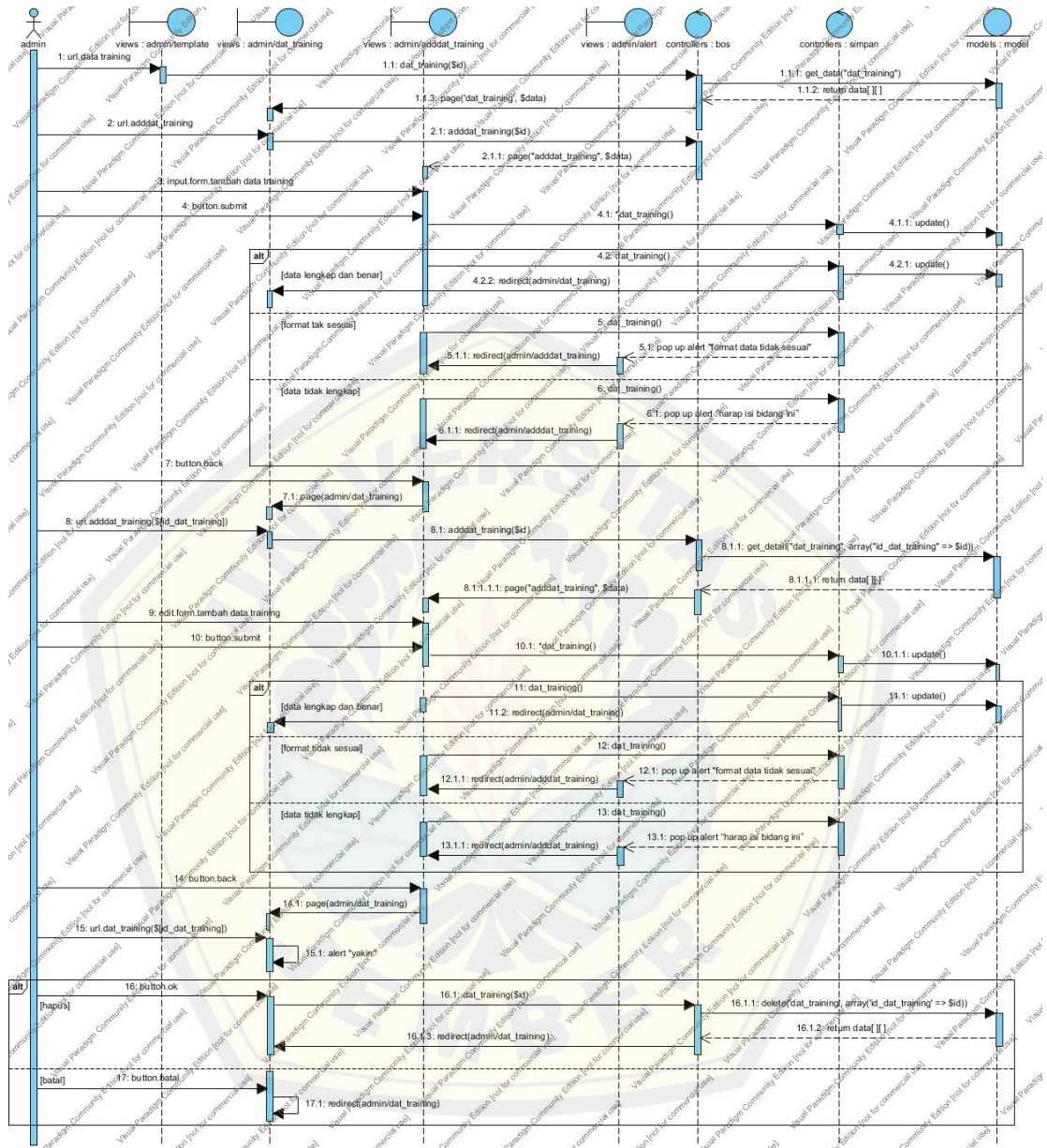




Gambar 4.3 Sequence Diagram Login  
Sumber: Hasil Analisis 2015

4.3.4.2. Sequence Diagram Data Training

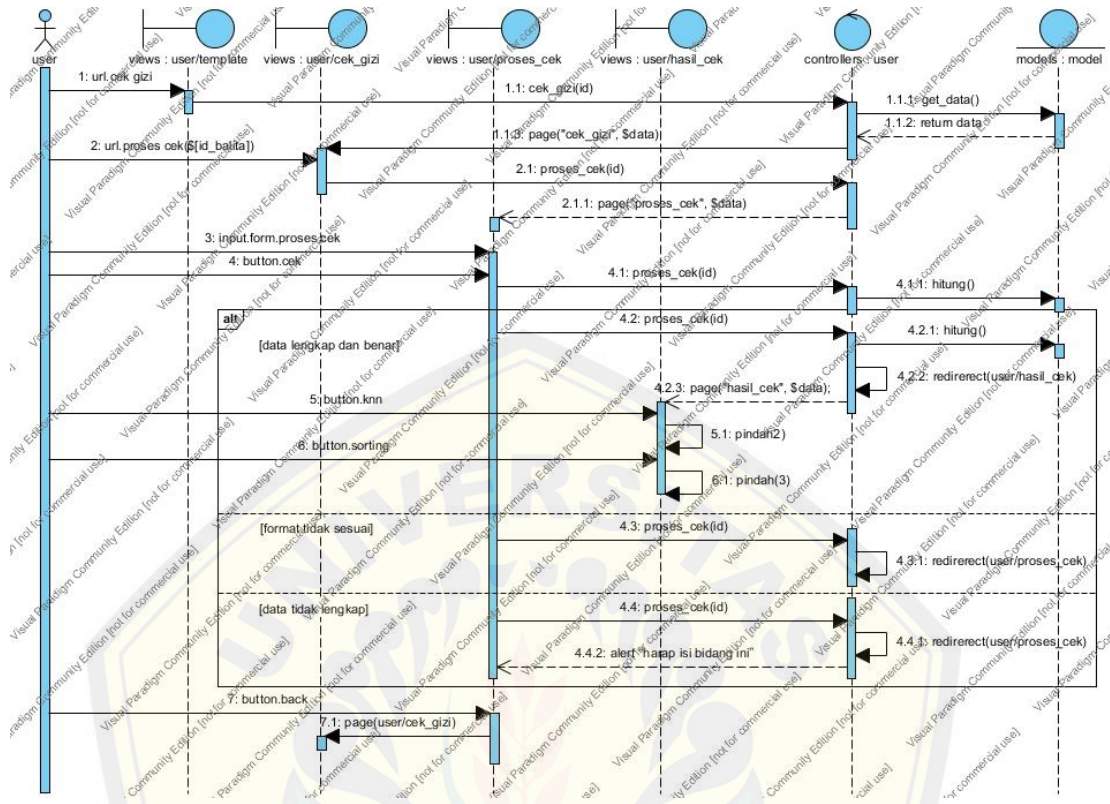
Sequence diagram data training menggambarkan alur proses yang terjadi ketika admin mulai melakukan proses tambah, edit dan hapus data training hingga proses yang dilakukan oleh sistem ke dalam database berikut.



Gambar 4.4 Sequence Diagram Data Training  
 Sumber: Hasil Analisis 2015

4.3.4.3. Sequence Diagram Cek Gizi

Sequence diagram cek gizi menggambarkan alur proses yang terjadi ketika user melakukan pengecekan gizi pada balitahingga proses yang dilakukan oleh sistem ke dalam database berikut.



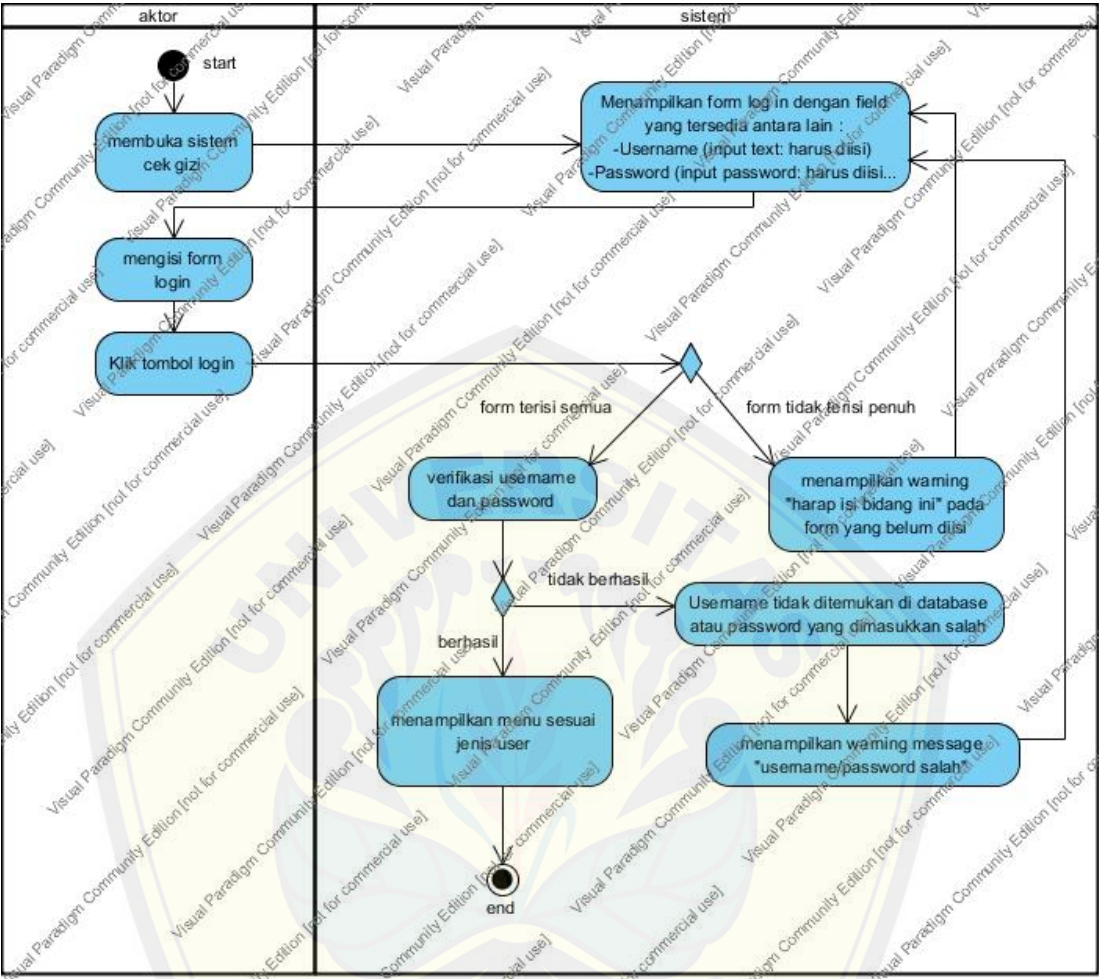
Gambar 4.5 Sequence Diagram Cek Gizi  
 Sumber: Hasil Analisis 201

4.3.5. Activity Diagram

Activity diagram menggambarkan skenario dalam bentuk bagan. Bagan tersebut menjelaskan aktivitas antara aktor dengan sistem pada Sistem Pendukung Keputusan Status Gizi Balita.

4.3.5.1. Activity Login

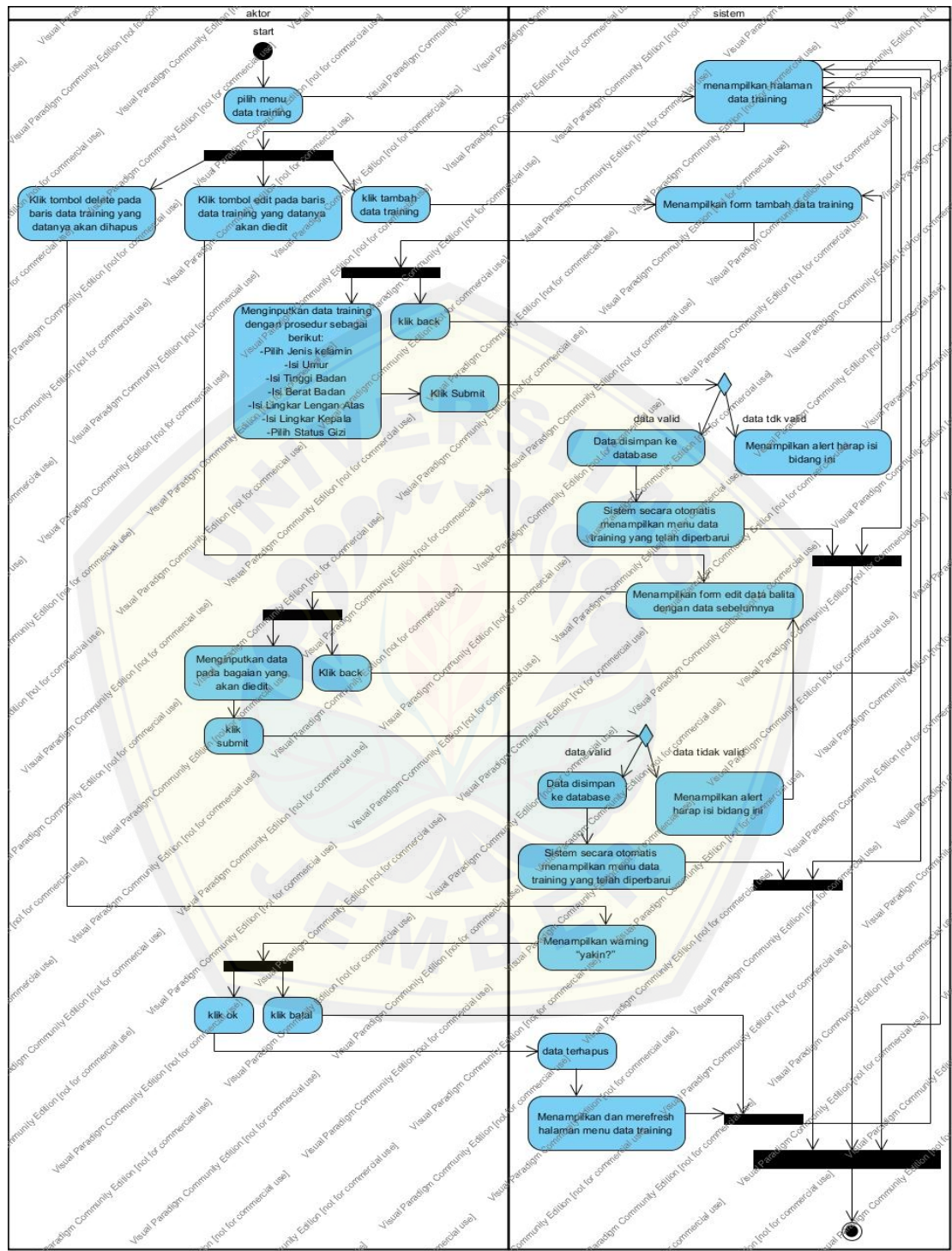
Activity login menggambarkan aliran aktivitas aktor dan sistem pada fitur login. Dalam proses ini terdapat dua aktor yang menggunakan fitur login. Dua aktor tersebut adalah admin dan user. Setelah proses login berhasil, sistem akan menampilkan menu sesuai level aktor. Activity login digambarkan dapat dilihat pada gambar 4.6.



Gambar 4.6 Activity Diagram Login  
Sumber: Hasil Analisis 2015

4.3.5.2. Activity Data Training

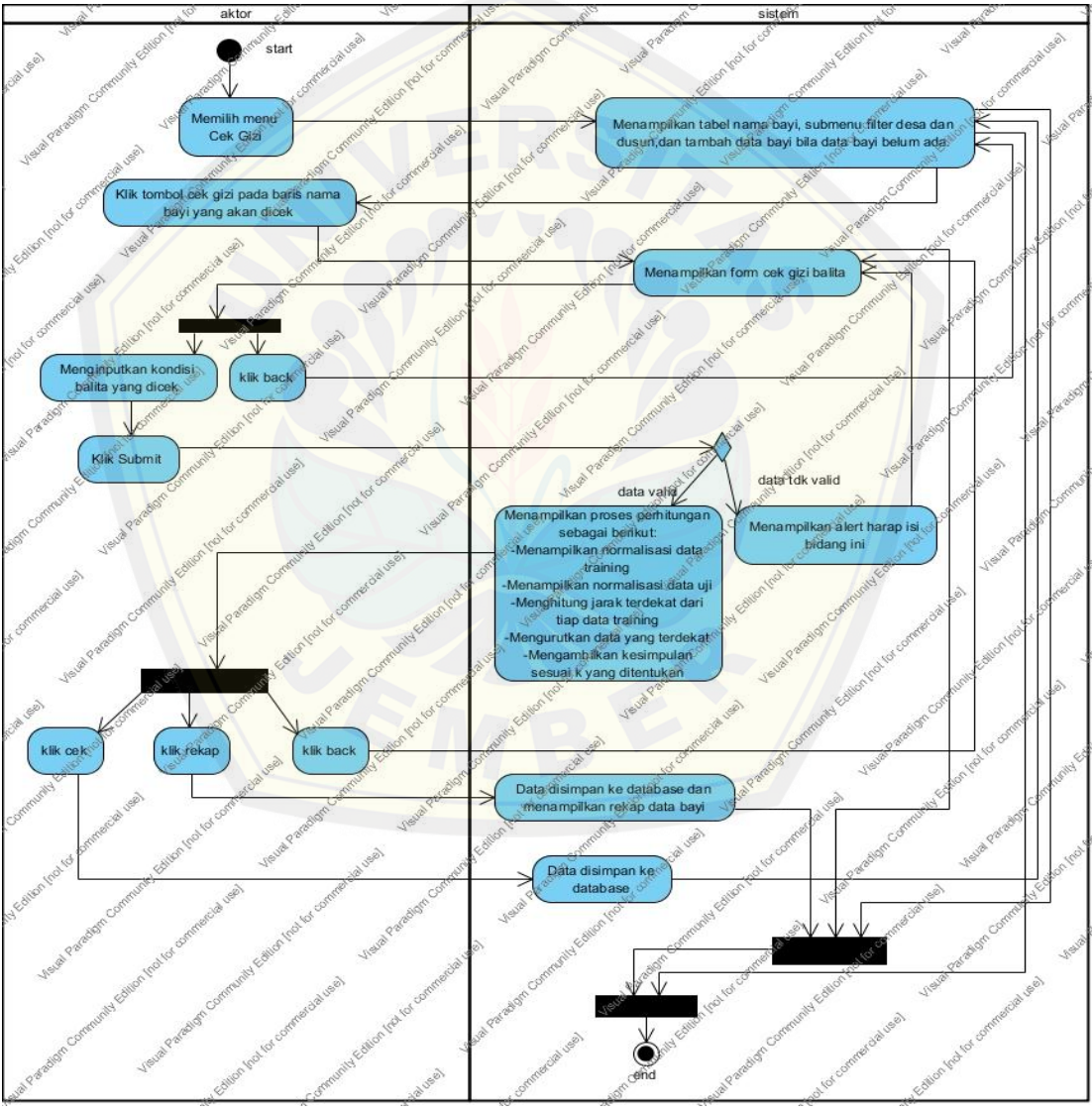
Activity data training menggambarkan aliran aktivitas admin dan sistem pada fitur data training. Activity ini bertujuan untuk menambahkan, mengedit, dan menghapus data training guna membantu dalam perhitungan K-NN dalam proses penentuan status gizi balita. Activity data training digambarkan dapat dilihat pada Gambar 4.7.



Gambar 4.7 Activity Diagram Data Training  
Sumber: Hasil Analisis 2015

4.3.5.3. Activity Cek Gizi Balita

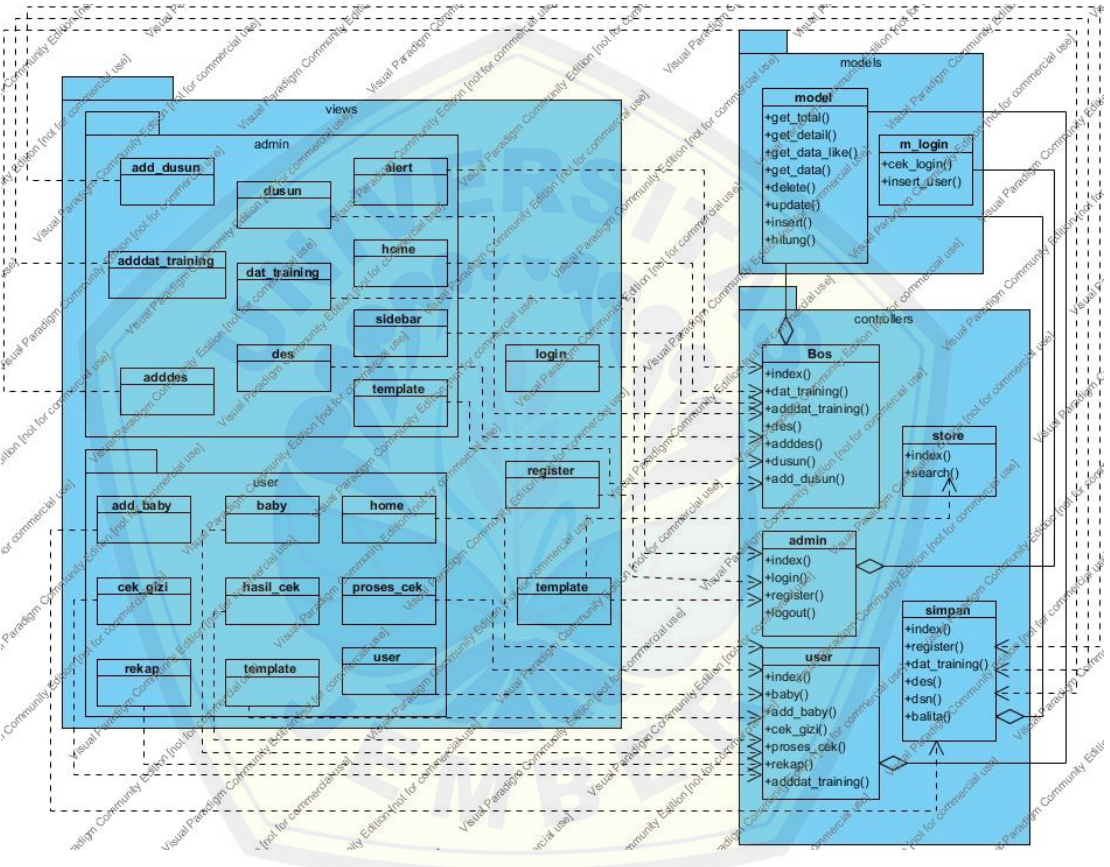
Activity cekgizi menggambarkan aliran aktivitas user dan sistem pada fitur cek gizi. Activity ini menggunakan data training yang telah ditambahkan pada fitur data training yang kemudian diolah dalam sistem dengan menggunakan metode K-Nearest Neighbor. Activity cek gizi balita digambarkan dapat dilihat pada Gambar 4.8.



Gambar 4.8 Activity Diagram Cek Gizi  
Sumber: Hasil Analisis 2015

4.3.6. Class Diagram

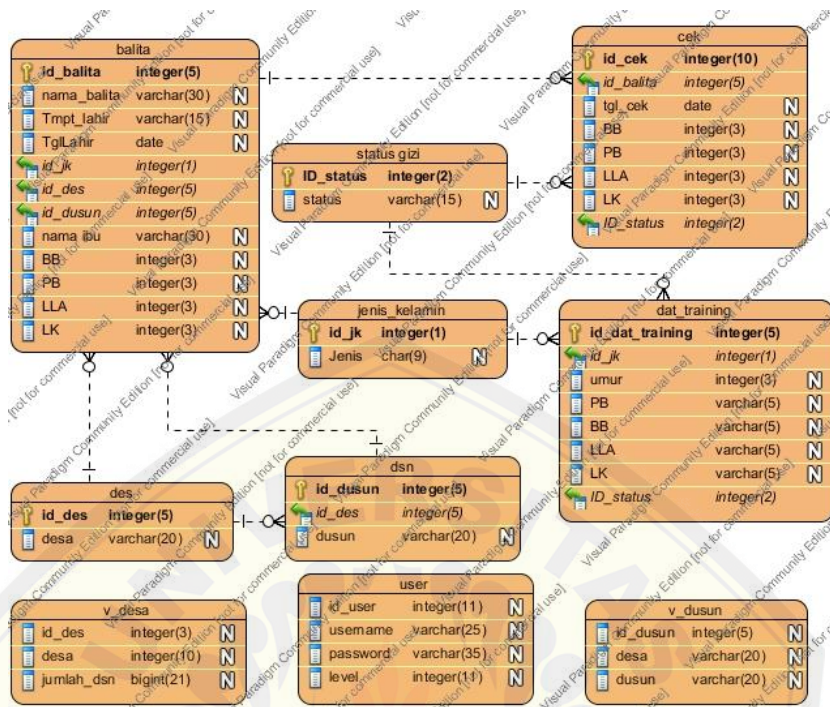
Class diagram menggambarkan kelas-kelas pada sistem yang dibangun dan hubungan antara kelas satu dan lainnya serta berisi atribut dan method apa saja yang ada didalamnya. Class diagram sistem penentuan keputusan status gizi balita dapat dilihat pada Gambar 4.9 .



Gambar 4.9 Class Diagram Sistem Pendukung Keputusan Status Gizi balita  
Sumber: Hasil Analisis 2015

4.3.7. Entity Relation Diagram

Entity Relation Diagram pada sistem penentuan gizi balita menggambarkan hubungan antar entitas dalam sebuah basis data. Entity relation diagram sistem pendukung keputusan status gizi balita dapat dilihat pada gambar 4.10 .



Gambar 4.10 Entity Relation Diagram Sistem Pendukung Keputusan Gizi Balita  
Sumber: Hasil Analisis 2015

#### 4.3.8. Koding

Pada tahap ini, desain selanjutnya akan diterjemahkan kedalam bentuk *coding* atau pengkodean menggunakan bahasa pemrograman yaitu *Page Hyper Text Pre-Process*(PHP), *Hyper Text Markup Language* (HTML), *Cascading Style Sheet* (CSS), dan *Javascript*. Manajemen data menggunakan *database Mysql*. Dalam perancangan Sistem Penunjang Keputusan Status Gizi Balita ini menggunakan *framework Code Igniter* untuk memudahkan di dalam pengembangan dan penulisan *coding* di dalam sebuah pemrograman. Proses pengkodean dalam membangun Sistem Penunjang Keputusan Status Gizi Balita ini lebih lanjut akan dijelaskan pada lampiran B. Kode program *controller* untuk fitur data *training* dapat dilihat pada Tabel 4.7, cek gizi pada Tabel 4.8 dan model pada Tabel 4.9.



Tabel 4.7 Potongan kode program pada fitur data *training*(controller/bos)

```

Public function dat_training($id) {
    if (isset($id)) {
        $data['status'] = $this->model->delete('dat_training', array('id_dat_training' =>
$Id));
    }
    if($this->input->get_post('filter')){
        $data['filter']=$this->input->post('filter');
        $filter=array("id_jk"=>$data['filter']);
    }else{
        $data['filter']="";
        $filter=array();
    }
    $data['title'] = "Data Training";
    $data['halaman'] = $this->paging("bos/dat_training", $total, $perpage);
    $data['data'] = $this->model->get_data("dat_training", $filter, array(
        "order_by" => array("id_dat_training", "asc"),
        "limit" => array($id, $perpage)
    ));
    $this->page('dat_training', $data);
}

public function adddat_training($id) {
    if (isset($id)) {
        $data = $this->model->get_detail("dat_training", array("id_dat_training" =>
$Id));
    }
    $data['title'] = "Tambah Data Training";
    $this->page("adddat_training", $data);
}

```

Tabel 4.8 potongan kode program pada fitur cek gizi(controller/user)

```

public function cek_gizi($id) {
    if (isset($id)) {
        $data['status'] = $this->model->delete('cek', array('id_balita' => $id));
    }
    if($this->input->get_post('filter')){
        $data['filter']=$this->input->post('filter');
        $filter=array("id_jk"=>$data['filter']);
    }else{
        $data['filter']="";
        $filter=array();
    }
}

```

```

$data['title'] = "Data Bayi";
$data['data'] = $this->model->get_data("balita", $filter, array(
    "order_by" => array("id_balita", "asc")
));
$data['desa'] = $this->model->get_data("des");
$this->page("cek_gizi", $data);
}
public function proses_cek($id) {
    if($this->input->get_post('BB')){
        $data['dt']=$this->model->hitung_dt($this->input->post('id_balita'));
        $data['du']=$this->model->hitung_du(array(
            'id_balita'=>$this->input->post('id_balita'),
            'BB'=>$this->input->post('BB'),
            'PB'=>$this->input->post('PB'),
            'LLA'=>$this->input->post('LLA'),
            'LK'=>$this->input->post('LK'));
        $data['hasil']=$this->model->hitung($data['dt'],$data['du'],$this->input-
>post('k'));
        $this->page("hasil_cek", $data);
//        redirect(base_url("user/rekap")); //buat muculin proses kalo d comment
// SIMPAN KE TABEL CEK
        if ($this->input->get_post('nama_balita')) {
            $input = array('id_balita' => $this->input->post('id_balita'),
                'tgl_cek' => $this->input->post('tgl_cek'),
                'BB' => $this->input->post('BB'),
                'PB' => $this->input->post('PB'),
                'LLA' => $this->input->post('LLA'),
                'LK' => $this->input->post('LK'),
                'LD' => $this->input->post('LD'),
                'status' => $data['hasil']['statusnya']);
            $status = $this->model->update('cek', $input);
        }
        }else{
            $data = $this->model->get_detail("balita", array("id_balita"=>$id));
            $this->page("proses_cek", $data);
        }
    }
}
}

```

Tabel 4.9 Potongan kode program pada Model

```

class model extends CI_Model {
    private $sb_bb;private $sb_pb;private $sb_lla;private $sb_lk;
    private $r_bb;private $r_pb;private $r_lla;private $r_lk;
    //menghitung total jumlah baris
    public function get_total($table, $cond, $groupby) {
        if (isset($cond)) {
            $this->db->where($cond);

```

```
}
if (isset($groupby)) {
    $this->db->group_by($groupby);
}
$query = $this->db->get($table);
return $query->num_rows();
}

//ambil data
public function get_detail($table, $cond) {
    if (isset($cond)) {
        $this->db->where($cond);
    }
    $query = $this->db->get($table);
    foreach ($query->result_array() as $r)
        ;
    return $r;
}

public function get_data_like($table, $cond, $opt) {
    for ($x = 0; $x < count($cond['column']); $x++) {
        for ($y = 0; $y < count($cond['key']); $y++) {
            $this->db->or_like($cond['column'][$x], $cond['key'][$y]);
        }
    }
    if (is_array($opt)) {
        if (isset($opt['order_by'])) {
            $this->db->order_by($opt['order_by'][0], $opt['order_by'][1]);
        }
    }
    $query = $this->db->get($table, $opt['limit'][1], $opt['limit'][0]);
    return $query;
}

public function get_data($table, $cond, $no, $perpage) {
    if (isset($cond)) {
        $this->db->where($cond);
    }
    if (is_array($no)) {
        if (isset($no['order_by'])) {
            $this->db->order_by($no['order_by'][0], $no['order_by'][1]);
        }
        if (isset($no['group_by'])) {
            $this->db->group_by($no['group_by']);
        }
        if (isset($no['limit'])) {
```

```
$query = $this->db->get($table, $no['limit'][1], $no['limit'][0]);
} else {
    $query = $this->db->get($table);
}
} else {
    if (isset($no) && isset($perpage)) {
        $query = $this->db->get($table, $perpage, $no);
    } else {
        $query = $this->db->get($table);
    }
}
return $query;
}

public function delete($table, $cond) {
    if (isset($cond)) {
        $this->db->where($cond);
    }
    $query = $this->db->delete($table);
    return $query;
}

public function update($table, $insert) {
    if (trim($insert["id_{$table}"]) == "") {
        $query = $this->db->insert($table, $insert);
    } else {
        $this->db->where(array("id_{$table}" => $insert["id_{$table}"]));
        $query = $this->db->update($table, $insert);
    }
    return $query;
}

public function insert($table, $data) {
    return $this->db->insert($table, $data);
}

public function hitung_dt($id) {
    $data = $this->db->get_where("balita", array('id_balita' => $id));
    $balita = $data->row_array();
    $jk = $balita['id_jk'];
    $data = $this->db->get_where("dat_training", array('id_jk' => $jk));
    $hasil = array();
    $idx = 0;
    foreach ($data->result_array() as $v) {
        $hasil['bb'][$idx] = $v['BB'];
        $hasil['bb2'][$idx] = pow($v['BB'], 2);
        $hasil['pb'][$idx] = $v['PB'];
    }
}
```

```

$hasil['pb2'][$idx] = pow($v['PB'], 2);
$hasil['lla'][$idx] = $v['LLA'];
$hasil['lla2'][$idx] = pow($v['LLA'], 2);
$hasil['lk'][$idx] = $v['LK'];
$hasil['lk2'][$idx] = pow($v['LK'], 2);
$hasil['status'][$idx] = $v['ID_status'];
$hasil['jk'][$idx] = $v['id_jk'];
$idx++;
}
return $this->normalisasi($hasil,TRUE);
}

public function hitung_du($data) {
    $q = $this->db->get_where("balita", array('id_balita' => $data['id_balita']));
    $balita = $q->row_array();
    $jk = $balita['id_jk'];
    $hasil = array();
    $idx = 0;
    $hasil['bb'][$idx] = $data['BB'];
    $hasil['bb2'][$idx] = pow($data['BB'], 2);
    $hasil['pb'][$idx] = $data['PB'];
    $hasil['pb2'][$idx] = pow($data['PB'], 2);
    $hasil['lla'][$idx] = $data['LLA'];
    $hasil['lla2'][$idx] = pow($data['LLA'], 2);
    $hasil['lk'][$idx] = $data['LK'];
    $hasil['lk2'][$idx] = pow($data['LK'], 2);
    $hasil['status'][$idx] = "";
    $hasil['jk'][$idx] = $jk;
    return $this->normalisasi($hasil,FALSE);
}

function normalisasi($hasil,$status) {
    $output = array();
    $idx = 0;
    $jml_data = count($hasil['status']);
    if($status==TRUE){
        $this->sb_bb=sqrt($jml_data * array_sum($hasil['bb2']) -
pow(array_sum($hasil['bb']), 2)
        / $jml_data * ($jml_data - 1));
        $this->sb_pb=sqrt($jml_data * array_sum($hasil['pb2']) -
pow(array_sum($hasil['pb']), 2)
        / $jml_data * ($jml_data - 1));
        $this->sb_lla=sqrt($jml_data * array_sum($hasil['lla2']) -

```

```

pow(array_sum($hasil['lla'], 2)
    / $jml_data * ($jml_data - 1));
    $this->sb_lk=sqrt($jml_data * array_sum($hasil['lk2']) -
pow(array_sum($hasil['lk'], 2) /
    $jml_data * ($jml_data - 1));
    $this->r_bb=(array_sum($hasil['bb']) / $jml_data);
    $this->r_pb=(array_sum($hasil['pb']) / $jml_data);
    $this->r_lla=(array_sum($hasil['lla']) / $jml_data);
    $this->r_lk=(array_sum($hasil['lk']) / $jml_data);
}
for ($i = 0; $i < $jml_data; $i++) {
    $output[$idx]['jk'] = $hasil['jk'][$i] == 2 ? "Wanita" : "Pria";
    $output[$idx]['bb'] = $hasil['bb'][$i];
    $output[$idx]['n_bb'] = ($hasil['bb'][$i] - $this->r_bb) / $this->sb_bb;
    $output[$idx]['pb'] = $hasil['pb'][$i];
    $output[$idx]['n_pb'] = ($hasil['pb'][$i] - $this->r_pb) / $this->sb_pb;
    $output[$idx]['lla'] = $hasil['lla'][$i];
    $output[$idx]['n_lla'] = ($hasil['lla'][$i] - $this->r_lla) / $this->sb_lla;
    $output[$idx]['lk'] = $hasil['lk'][$i];
    $output[$idx]['n_lk'] = ($hasil['lk'][$i] - $this->r_lk) / $this->sb_lk;
    $output[$idx]['status'] = $hasil['status'][$i];
    $idx++;
}
return $output;
}

public function hitung($strain,$data,$k) {
    $ket = array("", "Gizi Buruk", "Kurang", "Normal", "Over Weight",
"Obesitas");
    $hasil = array();
    foreach ($strain as $v) { //rumus knn
        $hasil[] = array(
            "jarak" =>
            sqrt(pow(($v['n_bb'] - $data[0]['n_bb']), 2) +
                pow(($v['n_pb'] - $data[0]['n_pb']), 2) +
                pow(($v['n_lla'] - $data[0]['n_lla']), 2) +
                pow(($v['n_lk'] - $data[0]['n_lk']), 2)),
            "status" => $ket[$v['status']],
            "id_status" => $v['status']
        );
    }
    $sort = $hasil; //sortingan dari jarak
    for ($i = 0; $i < count($sort) - 1; $i++) {
        for ($j = $i + 1; $j < count($sort); $j++) {
            if ($sort[$j]['jarak'] < $sort[$i]['jarak']) {
                $tmp = $sort[$i];

```

```
        $sort[$i] = $sort[$j];
        $sort[$j] = $tmp;
    }
}
$status = array(); //pengambilan kesimpulan sesuai k
for ($i = 0; $i < $k; $i++) {
    $sort[$i]["style"] = "color:red";
    $status[$sort[$i]['id_status']]['jumlah'] ++;
}
$sterbesar = $sort[0]['id_status'];
if ($sort[0]['jarak'] == 0) {
    $statusnya = $sort[0]['status'];
} else {
    for ($i = 1; $i < 5; $i++) {
        for ($j = $i + 1; $j <= 5; $j++) {
            if ($status[$j]['jumlah'] > $status[$sterbesar]['jumlah']) {
                $sterbesar = $j;
            }
        }
    }
    $statusnya = $ket[$sterbesar];
} //status
$data['status'] = $sterbesar;
// $this->db->insert("cek", $data);
return array("hasil" => $hasil, "sort" => $sort, "statusnya" => $statusnya);
}
```

## BAB 5. HASIL DAN PEMBAHASAN

### 5.1. Implementasi K-Nearest Neighbor untuk Penentuan Status Gizi Balita

Implementasi K-Nearest Neighbor (K-NN) pada kasus penentuan status gizi balita berdasarkan jarak terpendek dari data yang diuji ke data latih. Pencarian jarak terpendek pada K-NN biasanya menggunakan *Euclidian distance*. Akan tetapi, bila saat pengukuran jarak dari atribut memiliki nilai yang besar, seperti atribut pendapatan, maka perlu dilakukan normalisasi terlebih dahulu dengan menggunakan *Z-score standardization*. K-NN akan mencari k buah dalam urutan data terpendek untuk pengklasifikasian data uji tergolong kelas yang mana.

Berdasarkan algoritma K-NN, perhitungan dapat dilakukan seperti yang telah dijelaskan pada subbab yang membahas K-NN sebelumnya sebagai berikut:

Langkah 1.

Tentukan nilai K. misal K=3,

Langkah 2.

Input data yang akan diuji. Misal seorang anak perempuan memiliki berat badan 7,5kg, tinggi badan 70cm, lingkaran lengan atas 10cm, dan lingkaran kepala 30cm.

Langkah 3.

Normalisasikan nilai yang telah diinputkan dan data *training* sesuai rumus normalisasi Z-score.  $V' = \frac{(v - \bar{A})}{\sigma_A}$  Diketahui nilai rata-rata berat badan, tinggi badan, lingkaran lengan atas dan lingkaran kepala 49 anak perempuan pada data *training* secara berurutan adalah 10.02244898; 77.85306122; 19.69387755; 39.87755102.

Serta simpangan baku dari tiap nilai berat badan, lingkaran badan, lingkaran lengan atas, dan lingkaran kepala adalah 80.28508388; 380.950519; 165.7576298; 215.6959311. Setelah mendapatkan nilai rata-rata dan simpangan baku, data uji dinormalisasikan. Nilai normalisasi berat badan anak perempuan adalah

$$V' = \frac{(v - \bar{A})}{\sigma_A} = \frac{(7,5 - 10,02245)}{80,285} = -0.03141865$$



Nilai normalisasi tinggi badan anak perempuan adalah

$$V' = \frac{(v - \bar{A})}{\sigma_A} = \frac{(70 - 77,853)}{380,9505} = -0.020614386$$

Nilai normalisasi lingkaran lengan atas anak perempuan adalah

$$V' = \frac{(v - \bar{A})}{\sigma_A} = \frac{(10 - 19,693)}{165,7576} = -0.178115156$$

Nilai normalisasi lingkaran kepala anak perempuan adalah

$$V' = \frac{(v - \bar{A})}{\sigma_A} = \frac{(30 - 39,877)}{215,6959} = -0.1256149$$

Untuk nilai normalisasi data uji, dilakukan cara yang sama seperti perhitungan pada data uji dan hasil normalisasi data training anak perempuan dapat dilihat pada lampiran G.

Langkah 4.

Hitung jarak tiap atribut yang diinputkan dengan data *training* berdasarkan

rumus *Euclidian distance*  $d(\chi\chi) = \sqrt{\sum_r^n (\alpha(\chi_i)) - (\alpha(\chi_i))^2}$ . Misal penghitungan

jarak pada data training ke-1 dengan data anak perempuan yang diuji  $d =$

$$\sqrt{(0.015912682 - (-0.03141865))^2 + (0.018760806 - (-0.020614386))^2 + (0.001846808 - (-0.178115156))^2 + (0.000567693 - (-0.1256149))^2}$$

$$d = \sqrt{0.052099016} = 0.228252088$$

Jarak pada antara data uji dengan data training ke1 adalah 0,228252088. Pada data berikutnya hingga data ke 49 dicari jaraknya seperti pada Tabel 5.1 berikut.

Tabel 5.1 hasil hitung jarak data uji dengan data training

Berat Badan	Tinggi Badan	Lingkar Lengan Atas	Lingkar Kepala	<i>Euclidian distance</i> d
11.3	85	20	40	0.228252088
9.3	82	21	39	0.225527267

8.2	74	16	34	0.204135062
12.5	94	24	45	0.885687185
9.6	75.2	15	32	0.176769338
7	67	15	33	0.177973167
9.6	86	25	45	0.262508577
11.3	79.7	16	45	0.222443507
10.2	80	20	43	0.232016961
9.6	82	20	43	0.231707253
10	79	20	42	0.22860911
10	82	17	44	0.221602297
11.5	88	22	46	0.255549177
9.8	81	17	40	0.20922771
9.4	78.5	19	39	0.819307846
10.8	84	18	46	0.234403603
10.5	83	25	48	0.270872004
10	83	22	40	0.234373375
10	75	15	36	0.187528963
12	86.3	25	47	0.272563463
8	83	20	42	0.228662902
8	71	15	35	0.182875349
11	71	16	38	0.199645922
8	65	15	32	0.175889227
11	83	23	39	0.238886142
11	84	16	41	0.211300048
13	82	25	45	0.26860356
11.3	82	22	38	0.231894791
9	65.7	15	35	0.183098677
7.5	62	19	35	0.203228185
8.6	67.6	20	38	0.215131902
12.5	89	24	43	0.26008805
10	78	25	40	0.247963367
8	65	17	32	0.186272539
8.5	66.4	20	39	0.217725262
9	74.5	20	35	0.208528867
7.5	69	18	34	0.194606813
9	72	18	39	0.208180902
7.5	70	15	40	0.195860459
8	76	23	37	0.22906207

8.6	71.7	20	46	0.237296859
12.5	92.2	25	48	0.279437652
10	74	25	38	0.242695375
12.3	91	25	40	0.258241092
10	82	20	40	0.22421029
14	65	15	33	0.194811504
12	82	20	40	0.229002722
10.7	74	17	45	0.2240715
12	82	20	40	0.229002722

Langkah 5.

Selanjutnya data diurutkan berdasarkan jarak yang terpendek hingga terbesar seperti pada Tabel 5.2.

Tabel 5.2 Hasil pengurutan sesuai jarak terkecil hingga terjauh

Berat Badan	Tinggi Badan	Lingkar Lengan Atas	Status	<i>Euclidian distance d</i>
65	15	32	BAIK	0.175889227
75.2	15	32	BAIK	0.176769338
67	15	33	BAIK	0.177973167
71	15	35	LEBIH	0.182875349
65.7	15	35	BAIK	0.183098677
65	17	32	BAIK	0.186272539
75	15	36	BAIK	0.187528963
69	18	34	BAIK	0.194606813
65	15	33	BAIK	0.194811504
70	15	40	LEBIH	0.195860459
71	16	38	KURANG	0.199645922
62	19	35	BAIK	0.203228185
74	16	34	LEBIH	0.204135062
72	18	39	BAIK	0.208180902
74.5	20	35	BAIK	0.208528867
81	17	40	LEBIH	0.20922771
84	16	41	BURUK	0.211300048
67.6	20	38	BAIK	0.215131902
66.4	20	39	BAIK	0.217725262
82	17	44	BAIK	0.221602297
79.7	16	45	BAIK	0.222443507

74	17	45	BURUK	0.2240715
82	20	40	BAIK	0.22421029
82	21	39	BAIK	0.225527267
85	20	40	LEBIH	0.228252088
79	20	42	BAIK	0.22860911
83	20	42	BAIK	0.228662902
82	20	40	BAIK	0.229002722
82	20	40	BAIK	0.229002722
76	23	37	BAIK	0.22906207
82	20	43	BAIK	0.231707253
82	22	38	BAIK	0.231894791
80	20	43	BAIK	0.232016961
83	22	40	LEBIH	0.234373375
84	18	46	BAIK	0.234403603
71.7	20	46	KURANG	0.237296859
83	23	39	LEBIH	0.238886142
74	25	38	BAIK	0.242695375
78	25	40	BAIK	0.247963367
88	22	46	BAIK	0.255549177
91	25	40	BAIK	0.258241092
89	24	43	BAIK	0.26008805
86	25	45	BAIK	0.262508577
82	25	45	LEBIH	0.26860356
83	25	48	LEBIH	0.270872004
86.3	25	47	KURANG	0.272563463
92.2	25	48	BAIK	0.279437652
78.5	19	39	BAIK	0.819307846
94	24	45	BAIK	0.885687185

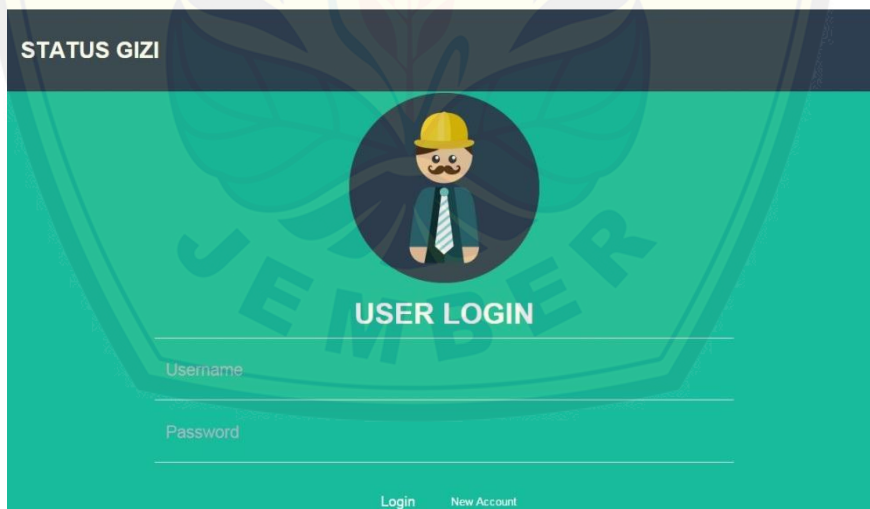
Langkah 6. Ambil jarak terpendek sesuai nilai K yang ditentukan. Misal k yang ditentukan pada langkah pertama  $k=3$ , maka 3 jarak terpendek pertama lah yang akan diambil kesimpulan untuk penentuan kondisi balita tersebut tergolong gizi mana. Hasil menunjukkan pada tabel 5.2 adalah 3 data urutan teratas adalah 3 baik. Sehingga dengan kondisi anak perempuan yang memiliki berat badan 7,5kg, tinggi badan 70cm, lingkaran lengan atas 10cm, dan lingkaran kepala 30cm, anak tergolong gizi baik.

## 5.2. Implementasi sistem

Tahap implementasi ini merupakan tahap pengkodean dari perancangan yang telah dibuat ke dalam bahasa pemrograman. Dalam penelitian ini, bahasa pemrograman yang digunakan yaitu bahasa pemrograman PHP. Tahap pengkodean akan menghasilkan beberapa *interface* atau tampilan dari Sistem Pendukung Keputusan Status Gizi Balita sesuai dengan hak akses *user*. Tampilan yang akan disajikan meliputi tampilan menu *login*, tampilan halaman utama menu *admin*, tampilan halaman utama menu *data training*, tampilan halaman utama menu *user*, dan tampilan halaman menu cek gizi. Untuk implementasi lainnya dari Sistem Pendukung Keputusan Status Gizi Balita dapat dilihat pada lampiran C.

### 1. Tampilan *Login*

Tampilan menu *login* untuk setiap *user* yaitu *useradmin* dan *userpraktisikesehatan* untuk mengakses fitur yang disediakan oleh Sistem Pendukung Keputusan Status Gizi Balita yang ditunjukkan pada Gambar 5.1.

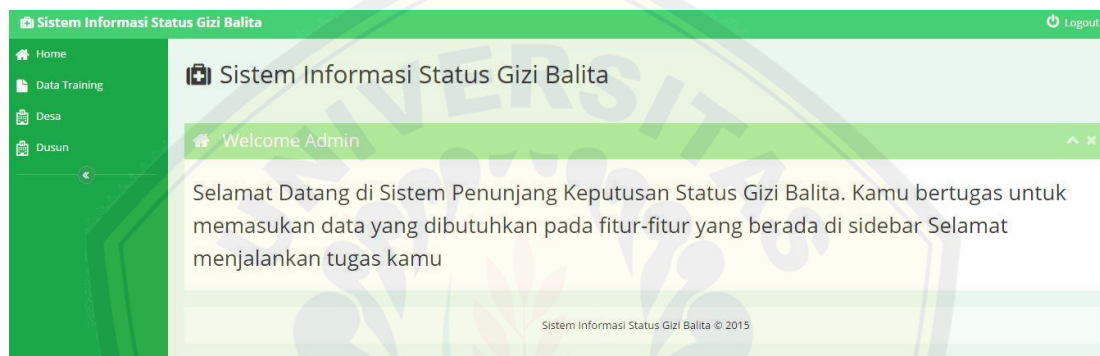


Gambar 5.1 *Form login*

Gambar 5.1 merupakan *interface* dari *form login*. Pada *form* ini *user* harus memasukkan *username* dan *password* yang telah mereka miliki dan sesuai dengan level *user*. Setiap *user* dapat masuk ke dalam sistem apabila *username* dan *password* yang mereka masukkan telah sesuai. Setelah *user* masuk ke dalam sistem dengan

*username* dan *password* yang benar, maka sistem akan memberikan hak akses menu sesuai dengan level *user*.

Berikut tampilan halaman utama *admin* pada Sistem Pendukung Keputusan Status Gizi Balita terdapat pada Gambar 5.2. Tampilan berikut merupakan tampilan sistem yang menampilkan pilihan menu untuk *useradmin*. Ketika *user* sebagai *admin* maka *admin* dapat mengakses beberapa menu yang terdiri menu *Data Training*, *Desa*, dan *Dusun*.



Gambar 5.2 halaman menu utama *admin*

Sedangkan untuk halaman utama *user* praktisi kesehatan pada Sistem Pendukung Keputusan Status Gizi Balita terdapat pada Gambar 5.3. Tampilan berikut merupakan tampilan sistem yang menampilkan pilihan menu untuk *userpraktisi*. Ketika *user* sebagai praktisi kesehatan maka *user* dapat mengakses beberapa menu yang terdiri menu *Baby*, *Cek Gizi*, dan *Rekap Gizi*.

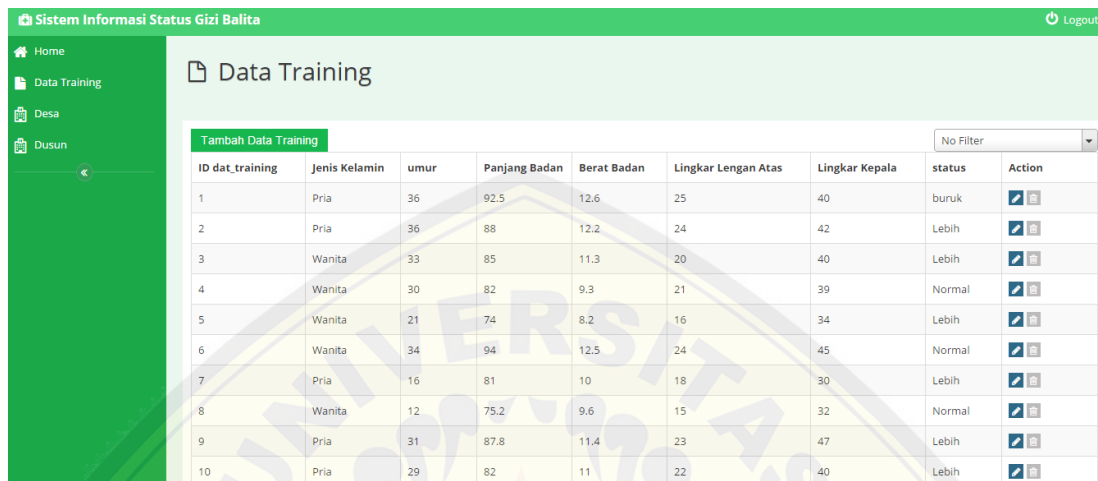


Gambar 5.3 halaman menu utama *user*

2. Tampilan halaman menu *Data Trainingadmin*

Menu *data trainingadmin* pada SPKStatus Gizi Balita terdiri dari sub menu tambah, *edit* dan hapus *data training*. Adminsaat mengakses menu ini memiliki hak

akses penuh pada data *training* guna perhitungan dalam menentukan status gizi balita yang ada pada menu *user* sebagai praktisi kesehatan. Tampilan halaman utama menu *data training* terdapat pada Gambar 5.4.



ID dat_training	Jenis Kelamin	umur	Panjang Badan	Berat Badan	Lingkar Lengan Atas	Lingkar Kepala	status	Action
1	Pria	36	92.5	12.6	25	40	buruk	<input type="checkbox"/> <input type="checkbox"/>
2	Pria	36	88	12.2	24	42	Lebih	<input type="checkbox"/> <input type="checkbox"/>
3	Wanita	33	85	11.3	20	40	Lebih	<input type="checkbox"/> <input type="checkbox"/>
4	Wanita	30	82	9.3	21	39	Normal	<input type="checkbox"/> <input type="checkbox"/>
5	Wanita	21	74	8.2	16	34	Lebih	<input type="checkbox"/> <input type="checkbox"/>
6	Wanita	34	94	12.5	24	45	Normal	<input type="checkbox"/> <input type="checkbox"/>
7	Pria	16	81	10	18	30	Lebih	<input type="checkbox"/> <input type="checkbox"/>
8	Wanita	12	75.2	9.6	15	32	Normal	<input type="checkbox"/> <input type="checkbox"/>
9	Pria	31	87.8	11.4	23	47	Lebih	<input type="checkbox"/> <input type="checkbox"/>
10	Pria	29	82	11	22	40	Lebih	<input type="checkbox"/> <input type="checkbox"/>

Gambar 5.4 halaman menu data *training* admin

Untuk menambah data *training*, pada fitur ini menyediakan form tambah data setelah mengklik *button* “Tambah Data *Training*”. Atribut yang digunakan diantaranya jenis kelamin, umur, panjang badan, berat badan, lingkar lengan atas, lingkar kepala, serta status. Tampilan form tambah data *training* dapat dilihat pada Gambar 5.5.

Form data *training* tersebut diisi seperti Gambar 5.6. Setelah form terisi, selanjutnya admin pilih tombol “Submit” untuk menyimpan data *training*. Apabila data telah sesuai maka akan muncul alert “Operation execution do exelent” seperti pada Gambar 5.7.



Sistem Informasi Status Gizi Balita

Logout

Home  
Data Training  
Desa  
Dusun

### Tambah Data Training

ID Data Training: ID Data Training

Jenis Kelamin: Pria (1)

umur: umur

Panjang Badan: Panjang Badan

Berat Badan: Berat Badan

Lingkar Lengan Atas: Lingkar lengan Atas

Lingkar Kepala: Lingkar Kepala

status: Buruk (1)

Submit Back

Gambar 5.5 form tambah data *training*

Sistem Informasi Status Gizi Balita

Logout

Home  
Data Training  
Desa  
Dusun

### Tambah Data Training

ID Data Training: ID Data Training

Jenis Kelamin: Wanita(2)

umur: 12

Panjang Badan: 100

Berat Badan: 10

Lingkar Lengan Atas: 10

Lingkar Kepala: 40

status: Kurang(2)

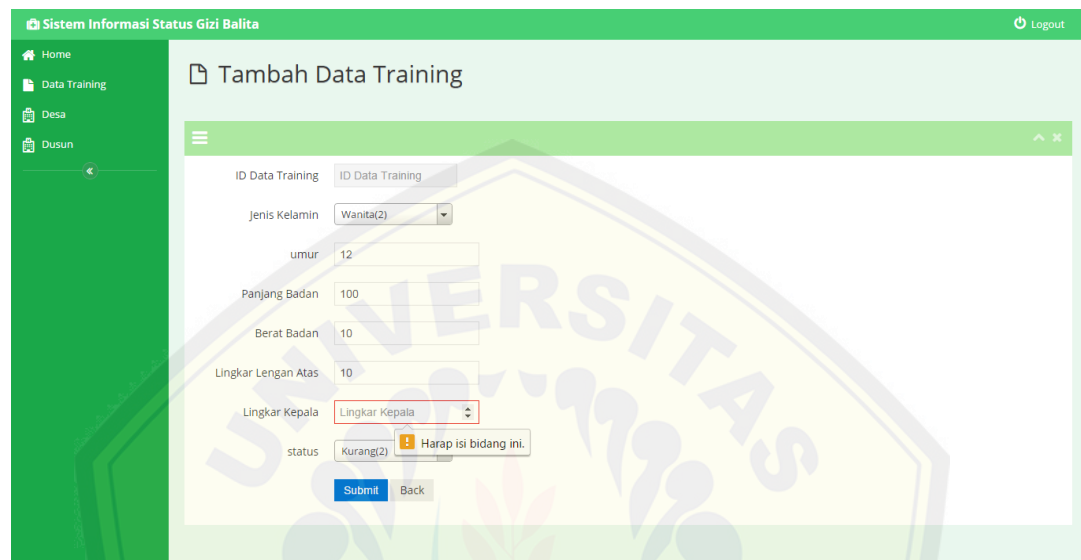
Submit Back

Gambar 5.6 Form *Data Training* yang telah terisi



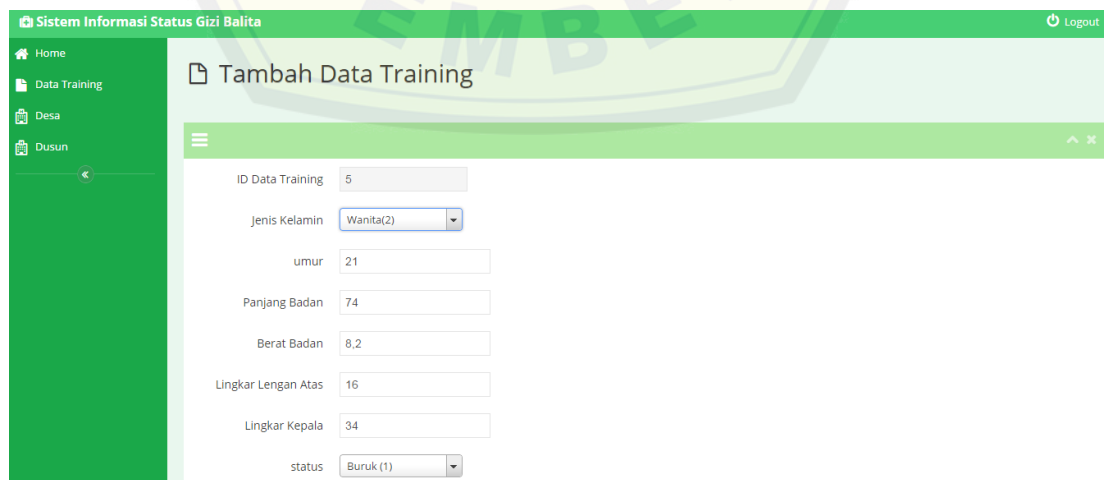
Gambar 5.7 *Alert* Data berhasil diinputkan

Ketika *admin* belum menginputkan data *training* dengan lengkap dan memilih tombol “Submit” maka sistem akan menampilkan *alert* “harap isi bidang ini” seperti pada Gambar 5.8.



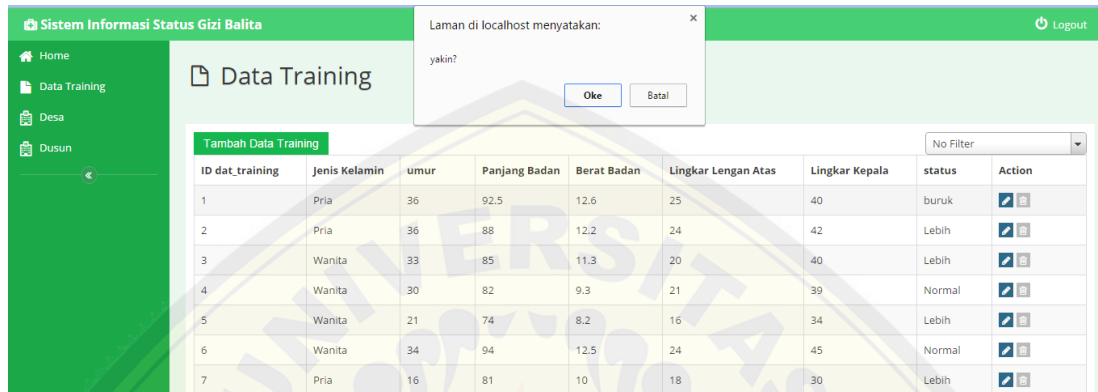
Gambar 5.8 Input Data *Training* gagal

Ketika *admin* akan merubah data *training*, setelah memilih salah satu data yang akan *edit*, tampilan *edit data training* akan seperti pada Gambar 5.9. *admin* mengisi pada atribut yang akan diubah dan klik “Submit “ setelah mengisi atribut yang tersedia.



Gambar 5.9 Edit Data *Training*

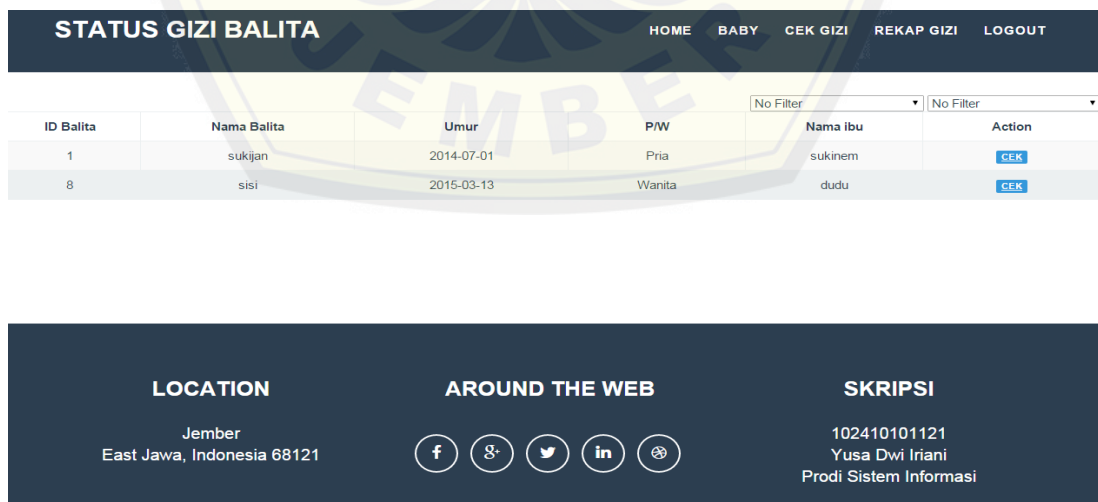
Untuk menghapus salah satu data *training*, setelah *admin* klik button hapus, sistem akan menampilkan kotak *alert* hapus seperti pada Gambar 5.10. Data akan terhapus bila *admin* klik “Oke” pada kotak *alert*, sedangkan klik “Batal” maka data batal terhapus.



Gambar 5.10 Hapus Data *Training*

3. Tampilan menu Cek Gizi *user*

Menu cek gizi *user* pada SPK Status Gizi Balita terdiri menu untuk pengecekan gizi. *User* saat mengakses menu ini memiliki hak akses menginputkan data balita saat pengecekan gizi guna perhitungan dalam menentukan status gizi balita. Tampilan halaman utama menu cek gizi terdapat pada Gambar 5.11.



Gambar 5.11 halaman menu cek gizi *user*

Mula-mula *user* melakukan pengecekan dengan memilih salah satu nama yang akan dicek dan klik “Cek”. Sistem akan menampilkan form Cek Gizi seperti pada Gambar 5.12. *User* menginputkan data selesai ukuran pada balita yang akan dicek. Akan tetapi bila *user* mengosongkan field yang tersedia maka sistem akan menampilkan *alert* “harap isi bidang ini” seperti pada Gambar 5.13 .



The screenshot shows a web interface titled "STATUS GIZI BALITA" with a navigation menu (HOME, BABY, CEK GIZI, REKAP GIZI, LOGOUT). The form contains the following fields:

Nama Balita	sukijan
Tanggal Cek	2015-06-18 09:18:19
BB	10
PB	110
LLA	16
LK	50
K	5

Buttons: Cek, Back

Gambar 5.12 Form Cek Gizi



The screenshot shows the same web interface as Gambar 5.12, but with a warning message on the 'K' field. The data is the same as in Gambar 5.12.

Nama Balita	sukijan
Tanggal Cek	2015-06-18 09:18:19
BB	10
PB	110
LLA	16
LK	LK
K	5

Buttons: Cek, Back

Warning: Harap isi bidang ini.

Gambar 5.13 Warning field belum terisi

Setelah klik Button cek setelah mengisi form cek gizi, maka akan menampilkan hasil perhitungan seperti pada Gambar 5.14 sampai 5.16.

The screenshot shows a web application interface for 'STATUS GIZI BALITA'. At the top, there are navigation links: HOME, BABY, CEK GIZI, REKAP GIZI, and LOGOUT. Below the navigation, there are three buttons: 'Normalisasi' (blue), 'KNN' (green), and 'Sorting' (orange). A 'Back' button is located in the top right corner. The main content area displays two tables. The first table is titled 'NORMALISASI DATA UJI' and has columns: No, JK, BB, PB, LLA, LK, and Status. The second table is titled 'NORMALISASI DATA TRAINING' and has the same columns. A large watermark of the 'UNIVERSITAS SUMBER MANDIRI' logo is overlaid on the tables.

No	JK	BB	PB	LLA	LK	Status
1	Pria	-0.0068	0.0440	-0.0144	0.0232	

No	JK	BB	PB	LLA	LK	Status
2	Pria	0.0153	0.0177	0.0254	-0.0026	Buruk
3	Pria	0.0119	0.0110	0.0209	0.0025	Lebih
4	Pria	-0.0068	0.0005	-0.0055	-0.0285	Lebih
5	Pria	0.0051	0.0107	0.0165	0.0155	Lebih
6	Pria	0.0017	0.0020	0.0121	-0.0026	Lebih
7	Pria	0.0144	0.0203	0.0209	0.0232	Lebih
8	Pria	0.0102	0.0080	0.0077	-0.0026	Buruk
9	Pria	0.0102	0.0117	0.0077	-0.0026	Normal
10	Pria	-0.0017	-0.0045	-0.0055	-0.0078	Normal

Gambar 5.14 Hasil perhitungan normalisasi data uji dan data *training*

The screenshot shows the same web application interface as before, but with the 'KNN' button highlighted. The main content area displays a table titled 'KNN' with columns: No, Jarak, and Status. A large watermark of the 'UNIVERSITAS SUMBER MANDIRI' logo is overlaid on the table.

No	Jarak	Status
53	0.058509000833516	Gizi Buruk
54	0.055791409715257	Over Weight
55	0.068183021106953	Over Weight
56	0.04758877938942	Over Weight
57	0.056628781102623	Over Weight
58	0.047521409662504	Over Weight
59	0.052349726668195	Gizi Buruk
60	0.049845730011969	Normal
61	0.058440791285111	Normal
62	0.05788936496787	Normal
63	0.051120246423061	Normal
64	0.069984538615968	Normal
65	0.059542919642267	Normal

Gambar 5.15 Hasil perhitungan jarak antara data uji dan data *training* menggunakan K-NN

No	Jarak	Status
1	0.038797986528082	Normal
2	0.042143108518292	Over Weight
3	0.044400458772206	Normal
4	0.046125605379419	Over Weight
5	0.046925256536744	Normal
6	0.047521409662504	Over Weight

Gambar 5.16 Hasil hitung K-NN yang telah di-*sorting* dan ditandai sesuai K

Setelah sistem melakukan perhitungan seperti gambar-gambar yang ada di atas, status gizi didapatkan dari hasil *sorting* data dari nilai terkecil hingga terbesar. Data diberi tanda warna merah sesuai dengan K yang diinputkan pada form cek. Bila  $K=3$  maka tiga data teratas akan diberi tanda merah dan ditarik kesimpulan dari hasil status yang terbanyak. Sesuai gambar 5.16 3 status terdekat adalah 1 overweight dan 2 normal. Sehingga status yang diambil adalah status normal. Status dapat dilihat di bagian bawah tabel sorting seperti pada Gambar 5.17. Data dan status akan tersimpan bila *admin* klik “Cek” sistem akan menampilkan kembali halaman cek gizi atau klik “Rekap” untuk melihat hasil rekap

**STATUS GIZI BALITA** HOME BABY CEK GIZI REKAP GIZI LOGOUT

50 0.23048469954699 Normal

51 0.23758664724675 Normal

**NORMAL**

Cek Rekap

**LOCATION**  
Jember  
East Jawa, Indonesia 68121

**AROUND THE WEB**

**SKRIPSI**  
102410101121  
Yusa Dwi Iriani  
Prodi Sistem Informasi

Copyright © Your Website 2015

Gambar 5.17 Hasil status gizi

### 5.3. Pengujian sistem

#### 5.3.1. Pengujian *White Box*

Pengujian *white box* ini mengukur kinerja sistem untuk melihat tingkat kompleksitas program. Teknik pengukuran yang digunakan yaitu pengujian jalur yang memiliki diagram dan grafik alir, perhitungan kompleksitas siklomatik, dan *test case*. Untuk lebih jelasnya pengujian Sistem Penunjang Keputusan Status Gizi dapat dilihat pada Lampiran D.

##### 5.3.1.1. Pemanggilan Data *Training*

Pada pembahasan sub bab berikut ini adalah tahap –tahap pengujian pada data *training*. Gambar 5.18 merupakan potongan *listing program* pada *function* *hitung\_dt* dalam model. *Hitung dt* merupakan kode program untuk pemanggilan data *training* sebelum dinormalisasi. Untuk diagram dan grafik alir dapat dilihat pada Gambar 5.19 dan 5.20, sedangkan *test case* data *training* dapat dilihat pada Tabel 5.3 dan 5.4.

#### a. *Listing Program*

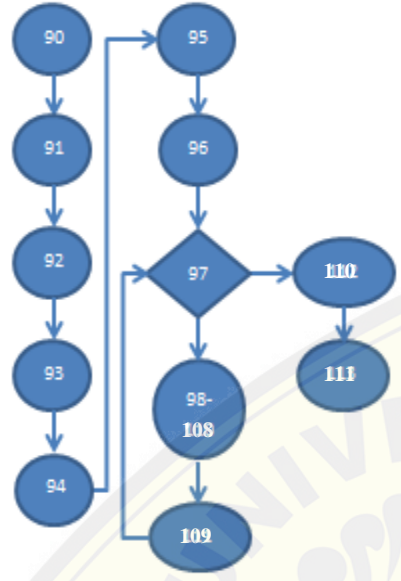
```

92 public function hitung_dt($id) { 1
93     $data = $this->db->get_where("balita", array('id_balita' => $id)); 2
94     $balita = $data->row_array(); 3
95     $jk = $balita['id_jk']; 4
96     $data = $this->db->get_where("dat_training", array('id_jk' => $jk)); 5
97     $hasil = array(); 6
98     $idx = 0; 7
99     foreach ($data->result_array() as $v) { 8
100         $hasil['bb'][$idx] = $v['BB'];
101         $hasil['bb2'][$idx] = pow($v['BB'], 2);
102         $hasil['pb'][$idx] = $v['PB'];
103         $hasil['pb2'][$idx] = pow($v['PB'], 2);
104         $hasil['lla'][$idx] = $v['LLA'];
105         $hasil['lla2'][$idx] = pow($v['LLA'], 2);
106         $hasil['lk'][$idx] = $v['LK'];
107         $hasil['lk2'][$idx] = pow($v['LK'], 2);
108         $hasil['status'][$idx] = $v['ID_status'];
109         $hasil['jk'][$idx] = $v['id_jk'];
110         $idx++;
111     } 10
112     return $this->normalisasi($hasil, TRUE); 11
113 } 12

```

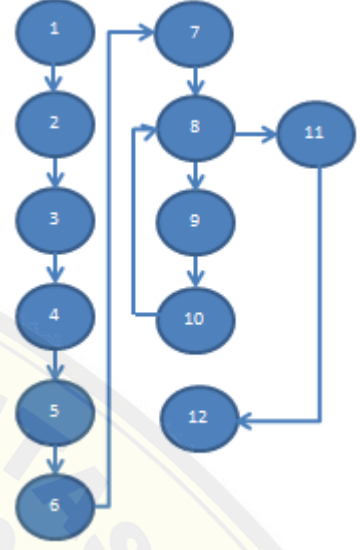
Gambar 5.18 Potongan *Listing Program* pada *function* *hitung\_dt*  
Sumber : Hasil Analisis 2015

b. Diagram Alir



Gambar 5.19 Diagram alir *function* hitung\_dt  
Sumber : Hasil Analisi 2015

c. Grafik Alir



Gambar 5.20 Grafik alir *function* hitung\_dt  
Sumber : Hasil Analisi 2015

d. Perhitungan Kompleksitas Siklomatik

Perhitungan Kompleksitas Siklomatik  $V(G)$  dengan grafik alir  $G$  dimana rumus  $V(G) = E - N + 2$ . Untuk Grafik alir pada *function* hitung\_dt() memiliki 12 edge dan 12 node.

$$V(G) = 12 - 12 + 2 = 2$$

e. Test Case

Tabel 5.3 Test case Pengujian *Function* hitung\_dt() jalur 1

Test Case	Jika data terpenuhi
Target yang diharapkan	Data terpanggil dan diproses
Hasil pengujian	Benar
Path/Jalur	1-2-3-4-5-6-7-8-9-10-8-11-12



Tabel 5.4 Test case Pengujian Function hitung\_dt() jalur 2

Test Case	Jika data tidak ada
Target yang diharapkan	Data tidak akan diproses
Hasil pengujian	Benar
Path/Jalur	1-2-3-4-5-6-7-8-11-12

### 5.3.1.2. Pemanggilan Data Uji

Pada pembahasan sub bab ini adalah tahap –tahap pengujian pada data uji. Gambar 5.21 merupakan potongan *listing program* pada *function* hitung\_du dalam model. Hitung\_du merupakan kode program untuk pemanggilan data uji sebelum dinormalisasi. Untuk diagram dan grafik alir dapat dilihat pada Gambar 5.22 dan 5.23, sedangkan *test case* data uji dapat dilihat pada Tabel 5.5.

#### a. Listing Program

```

116 public function hitung_du($data) { 1
117     $q = $this->db->get_where("balita", array('id_balita' => $data['id_balita2']));
118     $balita = $q->row_array(); 3
119     $jk = $balita['id_jk']; 4
120     $hasil = array(); 5
121     $idx = 0; 6
122     $hasil['bb'][$idx] = $data['BB']; 7
123     $hasil['bb2'][$idx] = pow($data['BB'], 2); 8
124     $hasil['pb'][$idx] = $data['PB']; 9
125     $hasil['pb2'][$idx] = pow($data['PB'], 2); 10
126     $hasil['lla'][$idx] = $data['LLA']; 11
127     $hasil['lla2'][$idx] = pow($data['LLA'], 2); 12
128     $hasil['lk'][$idx] = $data['LK']; 13
129     $hasil['lk2'][$idx] = pow($data['LK'], 2); 14
130     $hasil['status'][$idx] = ""; 15
131     $hasil['jk'][$idx] = $jk; 16
132     return $this->normalisasi($hasil, FALSE); 17
} 18

```

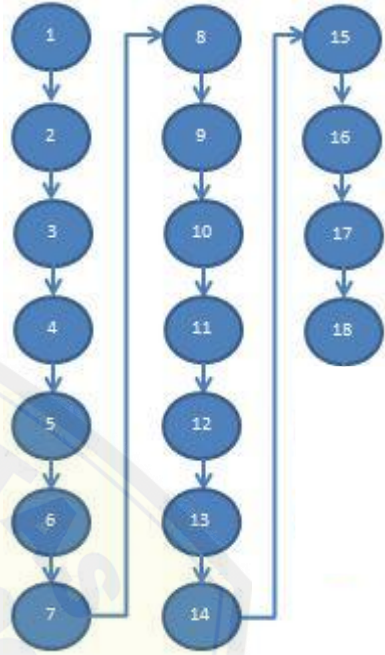
Gambar 5.21 Potongan Listing program pada *function* hitung\_du  
Sumber : hasil Analisis 2015

b. Diagram Alir



Gambar 5.22 Diagram alir *function*hitung\_du  
Sumber : Hasil Analisis 2015

c. Grafik Alir



Gambar 5.23 Grafik alir function  
hitung\_du  
Sumber : Hasil Analisis 2015

d. Perhitungan Kompleksitas Siklomatik  
 Perhitungan Kompleksitas Siklomatik  $V(G)$  dengan grafik alir  $G$  dimana rumus  $V(G) = E - N + 2$ . Untuk Grafik alir pada *function* hitung\_du() memiliki 19 *edge* dan 20 *node*.

$$V(G) = 17 - 18 + 2 = 1$$

e. Test Case

Tabel 5.5 Test case Pengujian *Function* hitung\_du()

Test Case	Jika data terisi
Target yang diharapkan	Data terproses
Hasil pengujian	Benar
Path/Jalur	1-2-3-4-5-6-7-8-9-10-11-12-13-14-15-16-17-18

### 5.3.1.3. Pemanggilan Normalisasi

Pada pembahasan sub bab ini adalah tahap –tahap pengujian pada normalisasi data uji dan data *training*. Gambar 5.24 merupakan potongan *listing program* pada *function* normalisasi dalam model. Normalisasi merupakan kode program untuk normalisasi data uji dan data *training* sebelum masuk ke perhitungan jarak menggunakan K-NN. Untuk diagram dan grafik alir dapat dilihat pada Gambar 5.25 dan 5.26, sedangkan *test case* data *training* dapat dilihat pada Tabel 5.6 hingga Tabel 5.8.

#### a. Listing Program

```

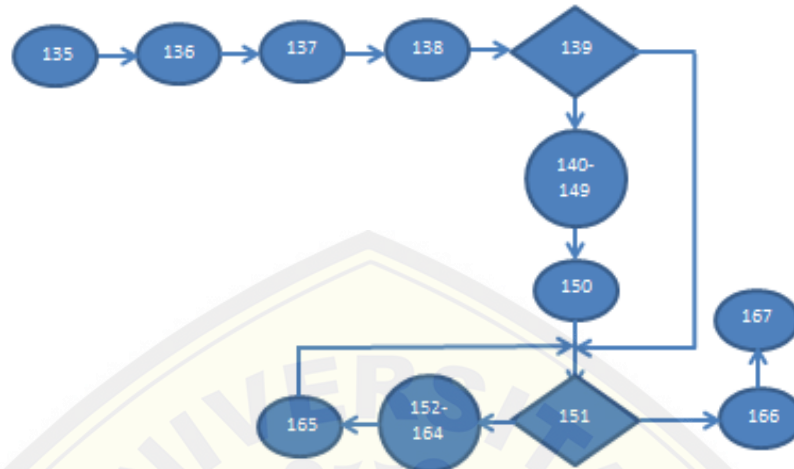
136 function normalisasi($hasil,$status) { 1
137     $output = array(); 2
138     $idx = 0; 3
139     $jml_data = count($hasil['status']); 4
140     if($status==TRUE){ 5
141         $this->sb_bb=sqrt($jml_data * array_sum($hasil['bb2']) - pow(array_sum($hasil['bb']), 2)
142         / $jml_data * ($jml_data - 1));
143         $this->sb_pb=sqrt($jml_data * array_sum($hasil['pb2']) - pow(array_sum($hasil['pb']), 2)
144         / $jml_data * ($jml_data - 1));
145         $this->sb_lla=sqrt($jml_data * array_sum($hasil['lla2']) - pow(array_sum($hasil['lla']), 2)
146         / $jml_data * ($jml_data - 1));
147         $this->sb_lk=sqrt($jml_data * array_sum($hasil['lk2']) - pow(array_sum($hasil['lk']), 2)
148         / $jml_data * ($jml_data - 1));
149         $this->r_bb=(array_sum($hasil['bb']) / $jml_data);
150         $this->r_pb=(array_sum($hasil['pb']) / $jml_data);
151         $this->r_lla=(array_sum($hasil['lla']) / $jml_data);
152         $this->r_lk=(array_sum($hasil['lk']) / $jml_data);
153     } 7
154     for ($i = 0; $i < $jml_data; $i++) { 8
155         $output[$idx]['jk'] = $hasil['jk'][$i] == 2 ? "Wanita" : "Pria";
156         $output[$idx]['bb'] = $hasil['bb'][$i];
157         $output[$idx]['n_bb'] = ($hasil['bb'][$i] - $this->r_bb) / $this->sb_bb;
158         $output[$idx]['pb'] = $hasil['pb'][$i];
159         $output[$idx]['n_pb'] = ($hasil['pb'][$i] - $this->r_pb) / $this->sb_pb;
160         $output[$idx]['lla'] = $hasil['lla'][$i];
161         $output[$idx]['n_lla'] = ($hasil['lla'][$i] - $this->r_lla) / $this->sb_lla;
162         $output[$idx]['lk'] = $hasil['lk'][$i];
163         $output[$idx]['n_lk'] = ($hasil['lk'][$i] - $this->r_lk) / $this->sb_lk;
164         $output[$idx]['status'] = $hasil['status'][$i];
165         $idx++;
166     } 10
167     return $output; 11
168 } 12

```

Gambar 5.24 Potongan *listing program* pada *function* normalisasi

Sumber : Hasil Analisis 2015

b. Diagram Alir



Gambar 5.25 Diagram alir pada function normalisasi  
 Sumber : Hasil Analisis 2015

c. Grafik Alir



Gambar 5.26 Grafik alir pada function normalisasi  
 Sumber : Hasil Analisis 2015

d. Perhitungan Kompleksitas Siklomatik

Perhitungan Kompleksitas Siklomatik  $V(G)$  dengan grafik alir  $G$  dimana rumus  $V(G) = E - N + 2$ . Untuk Grafik alir pada *function* normalisasi() memiliki 13 *edge* dan 12 *node*.

$$V(G) = 13 - 12 + 2 = 3$$

e. *Test Case*Tabel 5.6 *Test case* Pengujian *Function* normalisasi() jalur 1

Test Case	Hasil proses data training/data uji terproses
Target yang diharapkan	Menjalankan proses normalisasi
Hasil pengujian	Benar
Path/Jalur	1-2-3-4-5-6-7-8-9-10-8-11-12

Tabel 5.7 *Test case* Pengujian *Function* normalisasi() jalur 2

Test Case	Data gagal terproses pada proses sebelumnya
Target yang diharapkan	Data tidak diproses pada bagian pencarian rata-rata dan simpangan baku
Hasil pengujian	Benar
Path/Jalur	1-2-3-4-5-8-9-10-8-11-12

Tabel 5.8 *Test case* Pengujian *Function* normalisasi() jalur 3

Test Case	Tidak ada data yang terpanggil
Target yang diharapkan	Data tidak terproses
Hasil pengujian	Benar
Path/Jalur	1-2-3-4-5-8-11-12

## 5.3.1.4. Penentuan Status Gizi

Pada pembahasan sub bab ini adalah tahap –tahap pengujian pada normalisasi data uji dan data *training*. Gambar 5.27 merupakan potongan *listing program* pada *function* normalisasi dalam model. Normalisasi merupakan kode program untuk normalisasi data uji dan data *training* sebelum masuk ke perhitungan jarak menggunakan K-NN. Untuk diagram dan grafik alir dapat dilihat pada Gambar 5.28 dan 5.29, sedangkan test case data *training* dapat dilihat pada Tabel 5.9 hingga Tabel 5.18.

## a. Listing Program

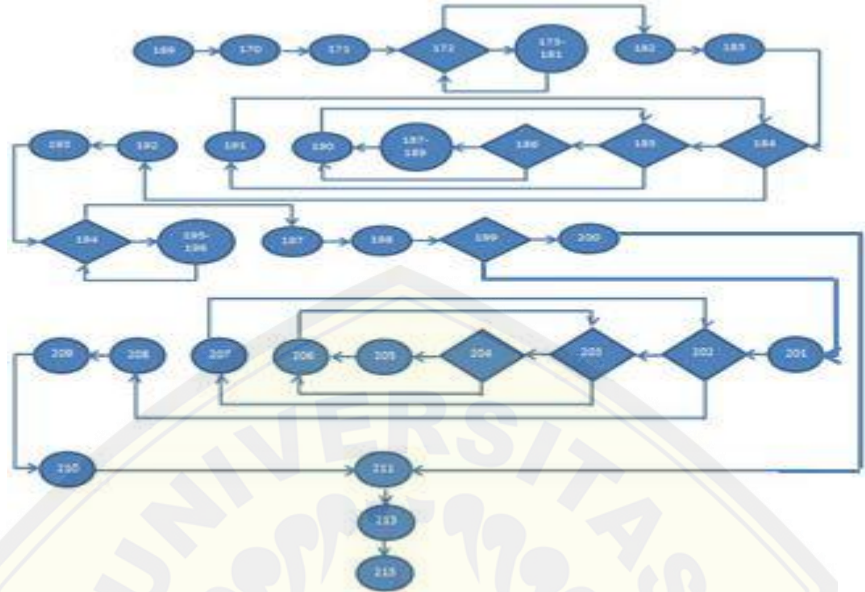
```

170 public function hitung($strain,$data,$k) {1
171     $ket = array("", "Gizi Buruk", "Kurang", "Normal", "Over Weight", "Obesitas");2
172     $hasil = array(); 3
173     foreach ($strain as $v) { //rumus knn 4
174         $hasil[] = array(
175             "jarak" =>
176                 sqrt(pow(($v['n_bb'] - $data[0]['n_bb']), 2) +
177                     pow(($v['n_pb'] - $data[0]['n_pb']), 2) +
178                     pow(($v['n_lla'] - $data[0]['n_lla']), 2) +
179                     pow(($v['n_lk'] - $data[0]['n_lk']), 2)),
180             "status" => $ket[$v['status']],
181             "id_status" => $v['status']
182         );
183     } 6
184     $sort = $hasil; //sortingan dari jarak 7
185     for ($i = 0; $i < count($sort) - 1; $i++) { 8
186         for ($j = $i + 1; $j < count($sort); $j++) { 9
187             if ($sort[$j]['jarak'] < $sort[$i]['jarak']) {10
188                 $tmp = $sort[$i];
189                 $sort[$i] = $sort[$j];
190                 $sort[$j] = $tmp;
191             } 12
192         } 13
193     } 14
194     $status = array(); //pengambilan kesimpulan sesuai k
195     for ($i = 0; $i < $k; $i++) { 15
196         $sort[$i]["style"] = "color:red";
197         $status[$sort[$i]['id_status']]['jumlah'] ++; } 16
198     $terbesar = $sort[0]['id_status'];18
199     if ($sort[0]['jarak'] == 0) { 19
200         $statusnya = $sort[0]['status']; 20
201     } else { 21
202         for ($i = 1; $i < 5; $i++) { 22
203             for ($j = $i + 1; $j <= 5; $j++) { 23
204                 if ($status[$j]['jumlah'] > $status[$terbesar]['jumlah']) { 24
205                     $terbesar = $j; 25
206                 } 26
207             } 27
208         } 28
209         $statusnya = $ket[$terbesar]; 29
210     } //status 30
211     $data['status'] = $terbesar; 31
212     // $this->db->insert("cek", $data);
213     return array("hasil" => $hasil, "sort" => $sort, "statusnya" => $statusnya); 32
214 } 33
215

```

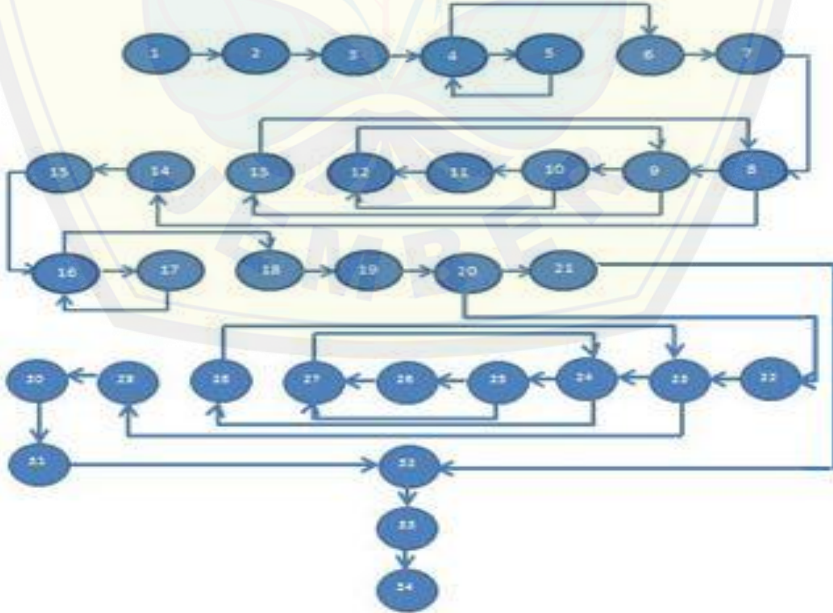
Gambar 5.27 Potongan listing program pada function hitung  
 Sumber : Hasil Analisis 2015

b. Diagram Alir



Gambar 5.28 Diagram alir *function* hitung  
Sumber : Hasil Analisis 2015

c. Grafik Alir



Gambar 5.29 Grafik alir pada *function* hitung  
Sumber: Hasil Analisis 2015

## d. Perhitungan Kompleksitas Siklomatik (jml edge)

Perhitungan Kompleksitas Siklomatik  $V(G)$  dengan grafik alir  $G$  dimana rumus  $V(G) = E - N + 2$ . Untuk Grafik alir pada *function* hitung() memiliki 42 edge dan 34 node.

$$V(G) = 42 - 34 + 2 = 10$$

## e. Test Case

Tabel 5.9 Test case Pengujian *Function* hitung() jalur 1

Test Case	Data berhasil terproses pada fungsi sebelumnya
Target yang diharapkan	Hasil jarak dan status ditemukan
Hasil pengujian	Benar
Path/Jalur	1-2-3-4-5-4-6-7-8-9-10-11-12-9-13-8-14-15-16-17-16-18-19-20-21-32-33-34

Tabel 5.10 Test case Pengujian *Function* hitung() jalur 2

Test Case	Tidak ada data yang terpanggil pada proses sebelumnya
Target yang diharapkan	Pencarian nilai jarak tidak diproses
Hasil pengujian	Benar
Path/Jalur	1-2-3-4-6-7-8-9-10-11-12-9-13-8-14-15-16-17-16-18-19-20-21-32-33-34

Tabel 5.11 Test case Pengujian *Function* hitung() jalur 3

Test Case	Data $i$ untuk proses sorting tidak ada
Target yang diharapkan	Melewatkan proses sorting
Hasil pengujian	Benar
Path/Jalur	1-2-3-4-5-4-6-7-8-14-15-16-17-16-18-19-20-21-32-33-34

Tabel 5.12 Test case Pengujian *Function* hitung() jalur 4

Test Case	Data juntuk proses sorting tidak ada
Target yang diharapkan	Melewatkan proses sorting
Hasil pengujian	Benar
Path/Jalur	1-2-3-4-5-4-6-7-8-9-13-8-14-15-16-17-16-18-19-20-21-32-33-34

Tabel 5.13 Test case Pengujian *Function* hitung() jalur 5

Test Case	Tidak ada data yang disorting
Target yang diharapkan	Proses sorting dilewati
Hasil pengujian	Benar
Path/Jalur	1-2-3-4-5-4-6-7-8-9-10-12-9-13-8-14-15-16-17-16-18-19-20-21-32-33-34



Tabel 5.14 *Test case* Pengujian *Function* hitung() jalur 6

Test Case	Nilai K tidak diinputkan
Target yang diharapkan	Pewarnaan merah pada nilai tidak ditampilkan
Hasil pengujian	Benar
Path/Jalur	1-2-3-4-5-4-6-7-8-9-10-12-9-13-8-14-15-16-18-19-20-21-32-33-34

Tabel 5.15 *Test case* Pengujian *Function* hitung() jalur 7

Test Case	Hasil perhitungan jarak $\neq 0$
Target yang diharapkan	Proses pencarian sesuai nilai K
Hasil pengujian	Benar
Path/Jalur	1-2-3-4-5-4-6-7-8-9-10-12-9-13-8-14-15-16-17-16-18-19-20-22-23-24-25-26-27-24-28-23-29-30-31-32-33-34

Tabel 5.16 *Test case* Pengujian *Function* hitung() jalur 8

Test Case	Data i untuk pencarian jarak tidak ada
Target yang diharapkan	Proses penentuan status dari status terbanyak dilewati
Hasil pengujian	Benar
Path/Jalur	1-2-3-4-5-4-6-7-8-9-10-12-9-13-8-14-15-16-17-18-19-20-22-23-29-30-31-32-33-34

Tabel 5.17 *Test case* Pengujian *Function* hitung() jalur 9

Test Case	Data j untuk pencarian jarak tidak ada
Target yang diharapkan	Proses penentuan status dari status terbanyak dilewati
Hasil pengujian	Benar
Path/Jalur	1-2-3-4-5-4-6-7-8-9-10-12-9-13-8-14-15-16-17-18-19-20-22-23-24-28-23-29-30-31-32-33-34

Tabel 5.18 *Test case* Pengujian *Function* hitung() jalur 10

Test Case	Tidak dapat menentukan status terbesar
Target yang diharapkan	Status tidak ditemukan
Hasil pengujian	Benar
Path/Jalur	1-2-3-4-5-4-6-7-8-9-10-12-9-13-8-14-15-16-17-18-19-20-22-23-24-25-27-24-28-23-29-30-31-32-33-34

### 5.3.2. Pengujian *Black Box*

Blackbox testing merupakan pengujian program yang melihat dari segi fungsional tanpa melihat desain dan kode program. Pengujian ini dilakukan oleh ahli gizi maupun praktisi kesehatan yang bertugas di tempat penelitian. *Black box testing* dapat dilihat pada lampiran E.

**5.4. Pengujian data untuk menentukan akurasi data yang optimal dengan nilai K yang berbeda**

Implementasi algoritma *K-Nearest Neighbor*(K-NN) pada aplikasi ini merupakan algoritma yang diterapkan pada sistem penunjang keputusan status gizi balita untuk menentukan status gizi pada balita. Setelah *training set* diinputkan, sistem siap diuji. Pengujian terhadap sistem ini menentukan penggunaan K mana yang optimal dalam menentukan status gizi tergolong klasifikasi mana. Pengujian ini menggunakan pengujian seluruh data training set yang disembunyikan kelas statusnya menjadi data uji yang biasa disebut *fulltrain fulltest*. Nilai K yang digunakan dalam metode ini yaitu K=3, K=4, K=5, dan K=7.

Tabel 5.19 hasil pengujian pada data anak laki-laki

Data anak laki-laki	K3	K4	K5	K7
Benar	40	39	35	34
Salah	11	12	16	17
persentase benar	78.43%	76.47%	68.63%	66.67%
persentase salah	21.57%	23.53%	31.37%	33.33%

Tabel 5.20 Hasil pengujian pada data anak perempuan

Data anak perempuan	K3	K4	K5	K7
Benar	43	41	37	37
Salah	6	8	12	12
persentase benar	87.76%	83.67%	75.51%	75.51%
persentase salah	12.24%	16.33%	24.49%	24.49%

Berdasarkan hasil pengujian pada tabel 5.19 dan 5.20, antara nilai K=3, K=4, K=5, dan K=7, nilai akurasi yang didapatkan yaitu 78.43%, 76.47%, 68.63%, 66.67% pada data anak laki-laki dan 87.76% , 83.67%, 75.51%, 75.51% pada data anak perempuan. Terlihat nilai akurasi terbaik diperoleh dengan nilai K=3 masing masing pada data anak laki-laki dan perempuan sebesar 78.43% dan 87.76%. Dengan jumlah data yang ada, nilai *k* yang semakin tinggi akan mengurangi efek *noise* pada klasifikasi, tetapi membuat batasan antara setiap klasifikasi menjadi semakin kabur.

## BAB 6. PENUTUP

### 6.1. Kesimpulan

Kesimpulan yang dapat diambil dari penelitian ini adalah sebagai berikut:

1. Sistem Penunjang Keputusan status gizi balita menggunakan *K-Nearest Neighbor* ini mampu menentukan status gizi berdasarkan parameter dari balita yang akan dicek menggunakan rumus *Z-score* untuk menormalisasikan nilai pada tiap parameter sebelum dilakukan penghitungan jarak menggunakan rumus *Euclidian distance*. Pengklasifikasian dilakukan bila nilai jarak antar data *training* dengan data yang diuji telah diurutkan berdasarkan jarak terdekat yang kemudian ditarik kesimpulannya berdasarkan nilai *K* yang telah ditentukan.
2. Pembuatan sistem ini dibangun berdasarkan model *Waterfall*. Pada tahap analisa, persyaratan pembangunan sistem didapatkan dari hasil observasi dan wawancara kepada pihak terkait. Pada tahap –tahap berikutnya pembangunan sistem dilakukan berdasarkan hasil analisa dan implementasi sistem.
3. Metode *K-Nearest Neighbor* pada penelitian ini, dari jumlah data *training* yang dipakai, memiliki *accuracy* sebesar 78,43% pada data anak laki-laki dan 87,76% pada data anak perempuan dengan nilai  $K=3$ . Nilai akurasi tersebut akan bergantung dengan semakin besarnya data dan/atau nilai *K*, akurasi data akan berbeda beda hasilnya

### 6.2. Saran

Sistem Pendukung Keputusan Status Gizi Balita ini membutuhkan pengembangan lebih lanjut dengan membangun sistem dengan *interface* yang lebih menarik, menguji kembali dengan jumlah data yang lebih banyak, serta penelitian lebih lanjut untuk atribut atribut yang digunakan agar hasil yang didapatkan lebih akurat. Sistem ini juga bisa dikembangkan tidak hanya mengukur status gizi balita saja, namun diharapkan dapat mengukur status gizi untuk segala usia menggunakan sistem gizi dengan menggunakan metode yang berbeda atau mengkombinasi metode *K-NN* dengan metode lainnya.

**DAFTAR PUSTAKA**

[http://dunia.news.viva.co.id/news/read/104891-ratusan\\_juta\\_anak\\_kekurangan\\_gizi](http://dunia.news.viva.co.id/news/read/104891-ratusan_juta_anak_kekurangan_gizi)(diakses 23 September 2014)

Afandie, Mochamad Noor. Cholissodin, Imam. Supianto, Ahmad Afif. Implementasi Metode K-Nearest Neighbor untuk Pendukung Keputusan Pemilihan Menu Makanan Sehat dan Bergizi. Universitas Brawijaya. Malang.

<http://ptiik.ub.ac.id/doro/archives/detail/>

Al Fatta, Hanif. 2007. Analisis dan Perancangan Sistem Informasi untuk Keunggulan Bersaing Perusahaan dan Organisasi Modern. Yogyakarta. ANDI

Almatsier, S. (2005). Prinsip Dasar Gizi. Jakarta: Gramedia Pustaka Utama

Apriadji, W. H. 1986. Gizi Keluarga. P.T. Penebar Swadaya, Jakarta.

Budiyanto, MAK. 2002. Dasar - Dasar Ilmu Gizi. Malang. UMM Press.

Depkes, RI, 2004, Analisis Situasi Gizi dan Kesehatan Masyarakat, Jakarta

Han, Jiawei and Kamber M. . 2006. "Data Mining: Concept and Techniques". Second *Edition*. Elsevier Inc.

Koirala, Shivprasad dan Sham Sheikh. 2009. Software Testing Interview Questions. Canada. Jones & Bartlett

Larose, D. T. 2005. Discovering Knowledge in Data. New Jersey : John Willey & Sons, Inc.

Peter G.W. Keen dan Scott Morton. 1978. Decision support systems: an organizational perspective Addison-Wesley series on decision support. Universitas Michigan. Addison-Wesley Pub. Co.

- Rismawan, Tedy. Wiedha Irawan, Ardhitya. Prabowo, Wahyu. Kusumadewi, Sri., 2008. **Sistem Pendukung Keputusan Berbasis Pocket PC sebagai Penentu Status Gizi Menggunakan Metode K-NN (K-Nearest Neighbor)**. Universitas Islam Indonesia. Yogyakarta  
<http://journal.uui.ac.id/index.php/jurnal-teknoin>
- Roedjito D, Djiteng. 1989. **Kajian Penelitian Gizi**. Jakarta. PT. Mediyatama Sarana Perkasa, Jakarta.
- Rosa A.S dan M. Shalahuddin. *Modul Pembelajaran Rekayasa Perangkat Lunak (Terstruktur dan Berorientasi Objek)*. Bandung: Modula.
- Supariasa, I Dewa Nyoman. Bakri, Bachyar. Fajar, Ibnu. 2013. **Penilaian Status gizi**. Jakarta. Buku kedokteran EGC.
- Sunarti, Euis. 2004. **Mengasuh Dengan Hati** Tantangan yang Menantang. Jakarta. PT. Elex Media Komputindo
- Turban, Efraim and Jay E. Aronson. 1998 *Decision Support Systems and Intelligent Systems*, Prentice Hall, Upper Saddle River, NJ.
- Wihardandi, Aji. Laporan PBB: 870 Juta Penduduk Dunia Kurang Gizi, 100 Juta Diantaranya Balita  
<http://www.mongabay.co.id/2012/10/19/laporan-pbb-870-juta-penduduk-dunia-kurang-gizi-100-juta-diantaranya-balita/>(diakses 23 September 2014).
- Witten, I. H., Frank, E., & Hall, M. A. 2011. *Data Mining: Practical Machine Learning and Tools*. Burlington : Morgan Kaufmann Publisher.

## LAMPIRAN

## A. Lampiran Perancangan

## A. 1. Skenario diagram

## Skenario Data Balita

Nama	Data Balita
Deskripsi	<i>User</i> dapat menambah, meng- <i>edit</i> , dan menghapus data balita
Aktor	<i>User</i>
Prekondisi	<i>User</i> memilih menu data balita
Postkondisi	Data balita berhasil ditambahkan di- <i>edit</i> , maupun dihapus
<b>Skenario Normal Data Balita</b>	
Aktor	Sistem
1. Melakukan login	
	2. Menampilkan halaman <i>user</i> dengan menu Baby, Cek gizi, Rekap gizi , Logout
3. Memilih menu baby	
	4. Menampilkan tabel nama balita, submenu filter tampilan sesuai jenis kelamin, tombol tambah data balita, serta tombol <i>edit</i> dan hapus pada tiap nama balita
<b>Skenario Normal Input Data Balita</b>	
5. Klik Tambah Data Balita	
	6. Menampilkan form tambah data balita yang berisi <i>field</i> : a. Id balita b. Nama balita ( <i>input</i> nama: harus diisi) c. Tempat lahir ( <i>input</i> nama: harus diisi) d. Tanggal lahir (jquery: harus dipilih) e. Jenis kelamin (jquery: harus dipilih) f. Desa (jquery: harus dipilih) g. Dusun (jquery: harus dipilih) h. Nama ibu ( <i>input</i> nama: harus diisi) i. Kondisi awal balita (berupa BB, TB, LLA, LK) ( <i>input</i> nomor: harus diisi)

7. Menginputkan data balita dengan prosedur sebagai berikut: a. Isi Nama balita tsb b. Isi nama tempat lahir balita tsb c. Pilih Tanggal lahir sesuai kelahiran balita tsb d. Pilih Jenis kelamin balita tsb e. Pilih Nama Desa balita tsb f. Pilih Nama Dusun balita tsb g. Isi Nama ibu dari balita tsb h. Isi Kondisi awal balita (berupa BB, TB, LLA, LK)	
8. Klik <i>Submit</i>	
	9. Data disimpan ke database
	10. Sistem secara otomatis menampilkan menu baby yang telah diperbarui
<b>Skenario Alternatif Input Data Balita</b>	
<b>Input data dengan format salah</b>	
7a. Menginputkan dengan format yang tidak sesuai	
	8a. Sistem menampilkan peringatan data yang diinputkan tidak sesuai format pada sisi field yang salah format
	9a. Menampilkan kembali form tambah data balita
<b>Data tidak lengkap</b>	
7b. Tidak mengisi data dengan lengkap	
8b. Klik <i>Submit</i>	
	9b. Menampilkan kembali form input data <i>user</i> dan <i>alert</i> harap isi bidang ini
<b>Batal saat sedang menginputkan data balita</b>	
7c. Klik <i>back</i>	
	8c. Menampilkan kembali halaman menu bayi

<b>Skenario Edit Data Balita</b>	
Aktor	Sistem
5. Klik tombol <i>edit</i> pada baris nama	

balita yang datanya akan <i>diedit</i>	
	6. Menampilkan form <i>edit</i> data balita dengan data sebelumnya
7. Menginputkan data pada bagaian yang akan <i>diedit</i>	
8. Klik <i>Submit</i>	
	9. Data disimpan ke database
	10. Sistem secara otomatis menampilkan menu baby yang telah diperbarui
<b>Skenario Alternatif <i>Edit</i> Data Balita</b>	
<b><i>Edit</i> data dengan format salah</b>	
7a. Mengeditkan dengan format yang tidak sesuai	
	8a. Sistem menampilkan peringatan data yang <i>diedit</i> kan tidak sesuai format pada sisi field yang salah format
<b>Data tidak lengkap</b>	
7b. Tidak mengisi data dengan lengkap	
8b. Klik <i>Submit</i>	
	9b. Menampilkan kembali form <i>edit</i> data <i>user</i> dan <i>alert</i> harap isi bidang ini
<b>Batal saat sedang mengeditkan data balita</b>	
7c. Klik <i>back</i>	
	8c. Menampilkan kembali halaman menu bayi

<b>Skenario hapus Data Balita</b>	
Aktor	Sistem
5. Klik tombol delete pada baris nama balita yang datanya akan dihapus	
	6. Menampilkan warning “yakin?”
7. Klik ok	
	8. Data terhapus
	9. Menampilkan dan merefresh halaman menu baby
<b>Skenario Alternatif Hapus Data Balita</b>	
<b>Batal hapus</b>	
7a. Klik batal	



	8. Menampilkan kembali menu bayi
--	----------------------------------

## Skenario View Rekap Gizi

Nama	Rekap Gizi
Aktor	<i>User</i>
Prekondisi	<i>User</i> memilih menu rekap gizi
Postkondisi	<i>User</i> melihat rekap data gizi balita
Skenario Rekap Gizi	
Aktor	Sistem
1. Melakukan Login	
	2. Menampilkan halaman <i>user</i> dengan menu Baby, Cek gizi, Rekap gizi, Logout
3. Memilih menu Rekap Gizi	
	4. Menampilkan tabel data gizi balita

## Skenario Data Desa

Nama	Data Desa
Deskripsi	<i>Admin</i> dapat menambah, meng- <i>edit</i> , dan menghapus Data Desa
Aktor	<i>Admin</i>
Prekondisi	<i>Admin</i> memilih menu data desa
Postkondisi	Data desa berhasil ditambahkan di- <i>edit</i> , maupun dihapus
Skenario Data Desa	
Aktor	Sistem
1. Melakukan login	
	2. Menampilkan halaman <i>admin</i> dengan menu Home, Data <i>training</i> , Desa, Dusun
3. Memilih menu desa	
	4. Menampilkan tabel data desa dan tombol tambah data desa
Skenario Input Data Desa	
5. Klik Tambah Data Desa	
	6. Menampilkan form tambah data desa yang berisi <i>field</i> : a. Id desa b. Desa ( <i>input</i> nama: harus diisi)
7. Menginputkan nama desa	

8. Klik <i>Submit</i>	
	9. Data disimpan ke database
	10. Sistem secara otomatis menampilkan menu data desa yang telah diperbarui
<b>Skenario Alternatif Input Data Desa</b>	
<b>Input data dengan format salah</b>	
7a. Menginputkan dengan format yang tidak sesuai	
	8a. Sistem menampilkan peringatan data yang diinputkan tidak sesuai format pada sisi field yang salah format
<b>Data tidak lengkap</b>	
7b. Tidak mengisi data dengan lengkap	
8b. Klik <i>Submit</i>	
	9b. Menampilkan kembali form input data desa dan <i>alert</i> harap isi bidang ini
<b>Batal saat sedang menginputkan data desa</b>	
7c. Klik <i>back</i>	
	8c. Menampilkan kembali halaman menu data desa

<b>Skenario Edit Data Desa</b>	
Aktor	Sistem
5. Klik ikon pensil untuk mengedit nama desa pada nama desa yang akan diubah	
	6. Menampilkan form <i>edit</i> data desa yang berisi <i>field</i> : a. Id desa b. Desa ( <i>input</i> nama: harus diisi)
7. Mengubah nama desa	
8. Klik <i>Submit</i>	
	9. Data disimpan ke database
	10. Sistem secara otomatis menampilkan menu data desa yang telah diperbarui
<b>Skenario Alternatif Edit Data Desa</b>	
<b>Edit data dengan format salah</b>	
7a. Mengubah dengan format yang tidak sesuai	

	8a. Sistem menampilkan peringatan data yang diinputkan tidak sesuai format pada sisi field yang salah format
<b>Data tidak lengkap</b>	
7b. Tidak mengisi data dengan lengkap	
8b. Klik <i>Submit</i>	
	9b. Menampilkan kembali form <i>edit</i> data desa dan <i>alert</i> harap isi bidang ini
<b>Batal saat sedang mengubah data desa</b>	
7c. Klik <i>back</i>	
	8c. Menampilkan kembali halaman menu data desa

<b>Skenario Hapus Data Desa</b>	
Aktor	Sistem
5. Klik tombol delete pada baris data desa yang datanya akan dihapus	
	6. Menampilkan warning “yakin?”
7. Klik ok	
	8. Data terhapus
	9. Menampilkan dan merefresh halaman menu data desa
<b>Skenario Alternatif Hapus Data Desa</b>	
<b>Batal hapus</b>	
7a. Klik batal	
	8a. Menampilkan kembali menu data desa

## Skenario Data Dusun

Nama	Data Dusun
Deskripsi	<i>Admin</i> dapat menambah, meng- <i>edit</i> , dan menghapus Data Dusun
Aktor	<i>Admin</i>
Prekondisi	<i>Admin</i> memilih menu data dusun
Postkondisi	Data dusun berhasil ditambahkan di- <i>edit</i> , maupun dihapus
<b>Skenario Data Dusun</b>	
Aktor	Sistem

1. Melakukan login	
	2. Menampilkan halaman <i>admin</i> dengan menu Home, Data <i>training</i> , Desa, Dusun
3. Memilih menu dusun	
	4. Menampilkan tabel data dusun dan tombol tambah data dusun
<b>Skenario Input Data Dusun</b>	
5. Klik Tambah Data Dusun	
	6. Menampilkan form tambah data dusun yang berisi <i>field</i> : a. Id dusun b. Desa (jquery: harus dipilih) c. Dusun ( <i>input</i> nama: harus diisi)
7. Menginputkan data dusun dengan prosedur sebagai berikut: a. Pilih nama desa b. Isi nama dusun	
8. Klik <i>Submit</i>	
	9. Data disimpan ke database
	10. Sistem secara otomatis menampilkan menu data dusun yang telah diperbarui
<b>Skenario Alternatif Input Data Dusun</b>	
<b>Input data dengan format salah</b>	
7a. Menginputkan dengan format yang tidak sesuai	
	8a. Sistem menampilkan peringatan data yang diinputkan tidak sesuai format pada sisi field yang salah format
<b>Data tidak lengkap</b>	
7b. Tidak mengisi data dengan lengkap	
8b. Klik <i>Submit</i>	
	9b. Menampilkan kembali form input data dusun dan <i>alert</i> harap isi bidang ini
<b>Batal saat sedang menginputkan data dusun</b>	
7c. Klik <i>back</i>	

	8c. Menampilkan kembali halaman menu data dusun
--	---

Skenario <i>Edit Data Dusun</i>	
Aktor	Sistem
5. Klik ikon pensil untuk mengedit nama dusun pada nama dusun yang akan diubah	
	6. Menampilkan form tambah data dusun yang berisi <i>field</i> : a. Id dusun b. Desa (jquery: harus dipilih) c. Dusun ( <i>input</i> nama: harus diisi)
7. Mengubah nama dusun	
8. Klik <i>Submit</i>	
	9. Data disimpan ke database
	10. Sistem secara otomatis menampilkan menu data desa yang telah diperbarui
Skenario Alternatif <i>Edit Data Dusun</i>	
<b><i>Edit data dengan format salah</i></b>	
7a. Mengubah dengan format yang tidak sesuai	
	8a. Sistem menampilkan peringatan data yang diinputkan tidak sesuai format pada sisi field yang salah format
<b>Data tidak lengkap</b>	
7b. Tidak mengisi data dengan lengkap	
8b. Klik <i>Submit</i>	
	9b. Menampilkan kembali form <i>edit</i> data dusun dan <i>alert</i> harap isi bidang ini
<b>Batal saat sedang mengubah data dusun</b>	
7c. Klik <i>back</i>	
	8c. Menampilkan kembali halaman menu data dusun

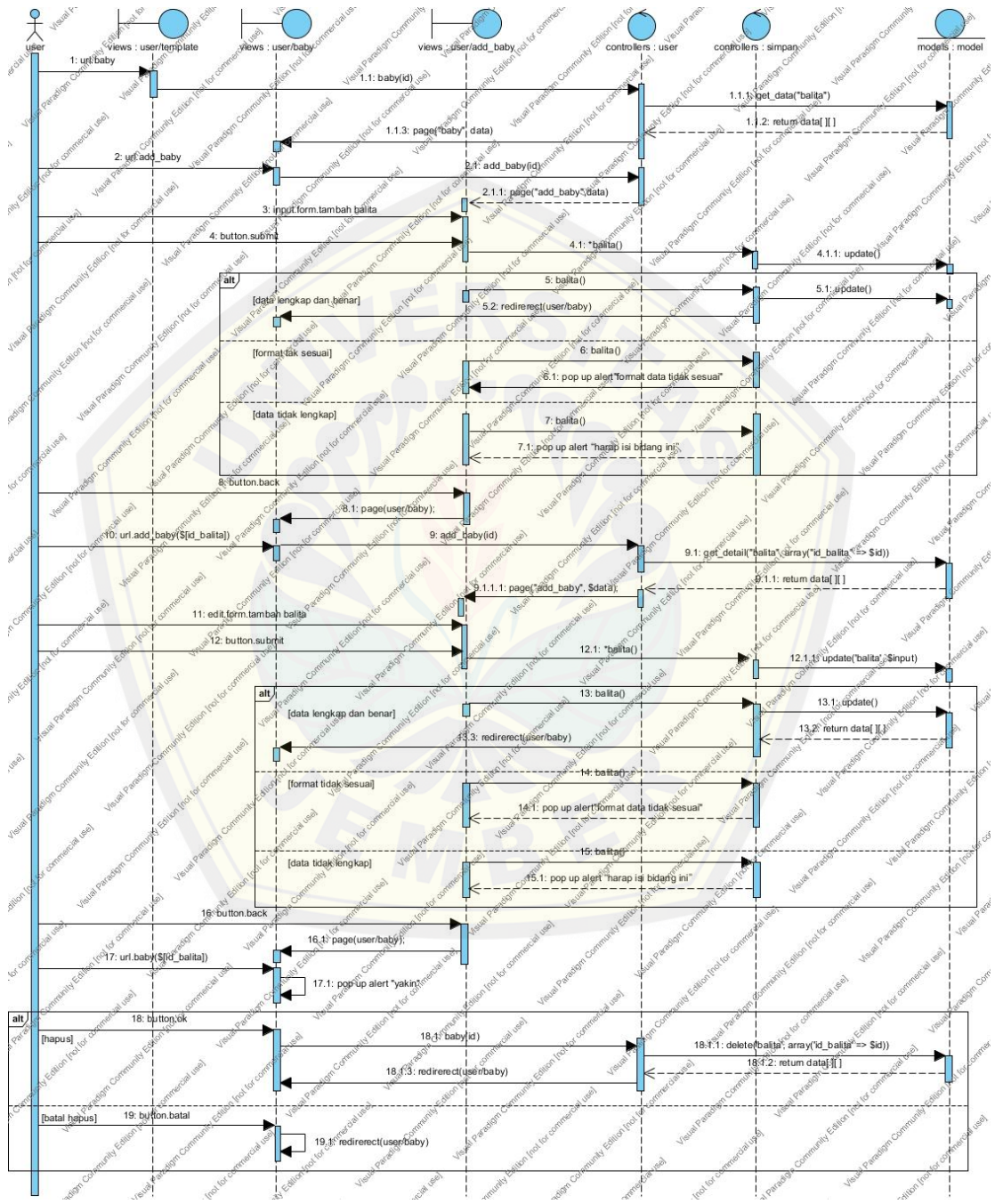
Skenario Hapus Data Dusun	
Aktor	Sistem
5. Klik tombol delete pada baris data dusun yang datanya akan	

dihapus	
	6. Menampilkan warning “yakin?”
7. Klik ok	
	8. Data terhapus
	9. Menampilkan dan merefresh halaman menu data dusun
<b>Skenario Alternatif Hapus Data Dusun</b>	
<b>Batal hapus</b>	
8a. Klik batal	
	9a. Menampilkan kembali menu data dusun

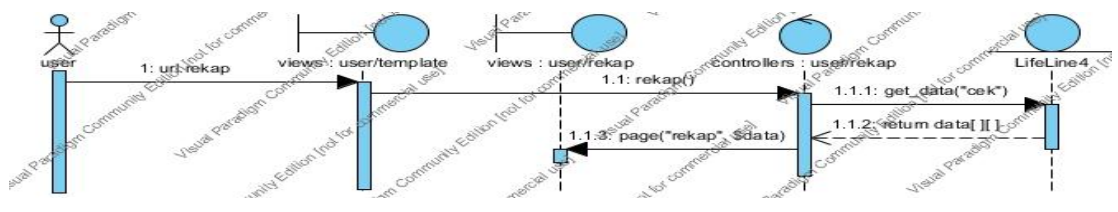


### A. 2. Sequence Diagram

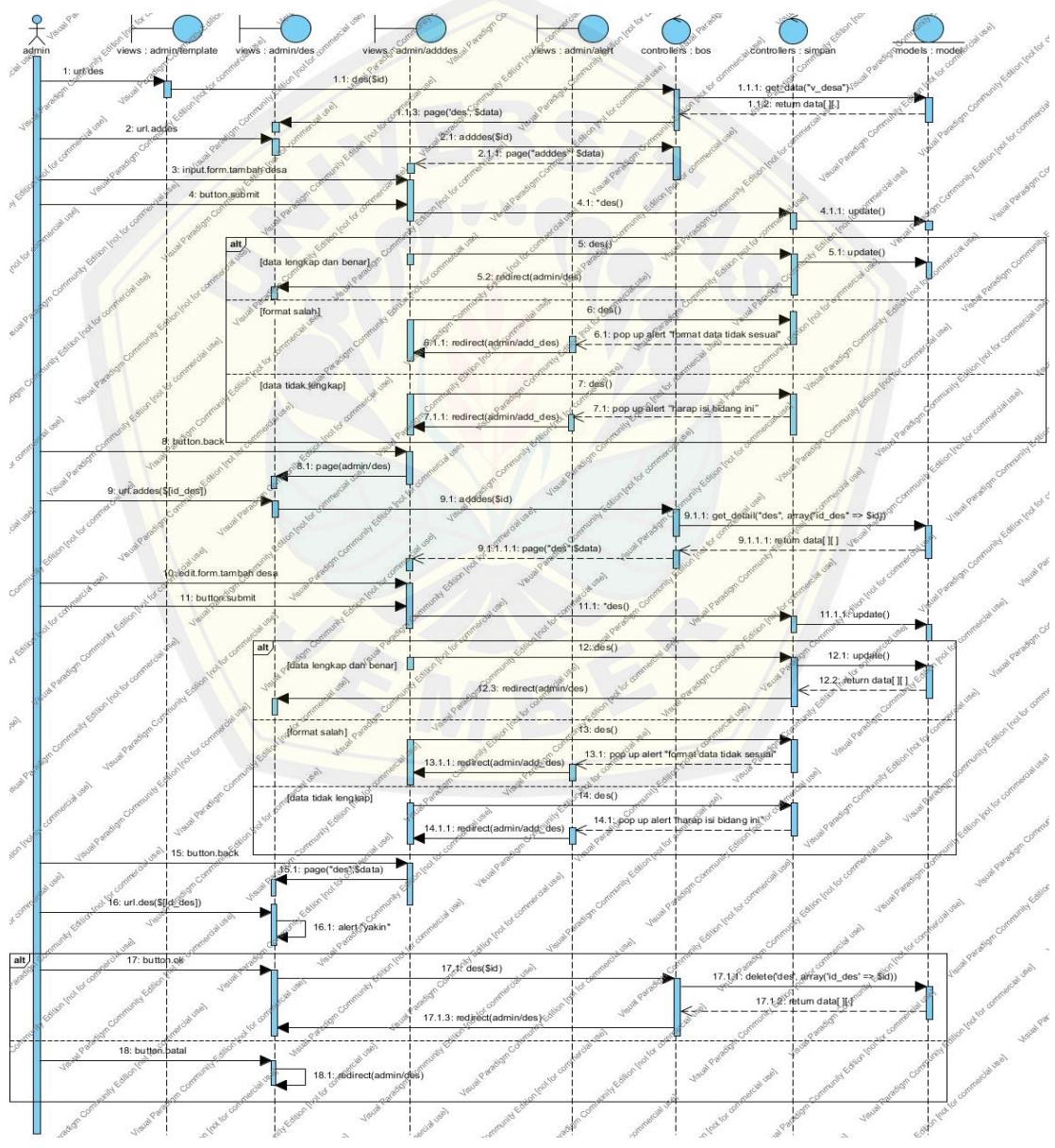
#### Sequence Diagram Data Balita



### Sequence Diagram View Rekap Gizi

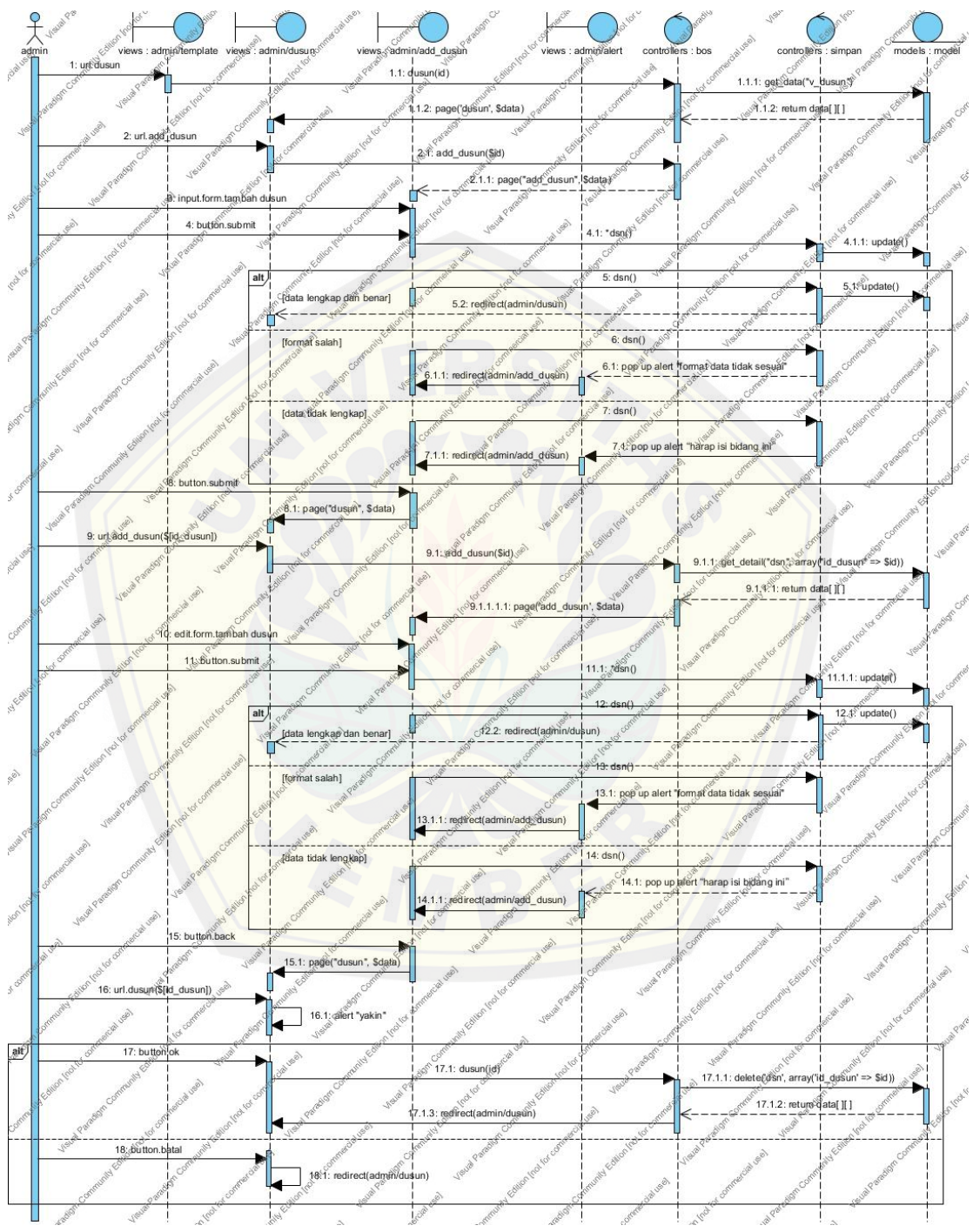


### Sequence Diagram Data Desa



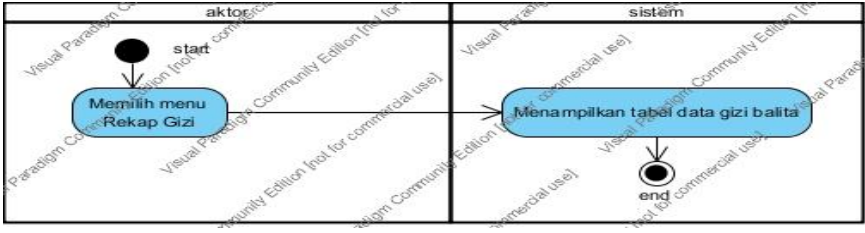


### Sequence Diagram Data Dusun

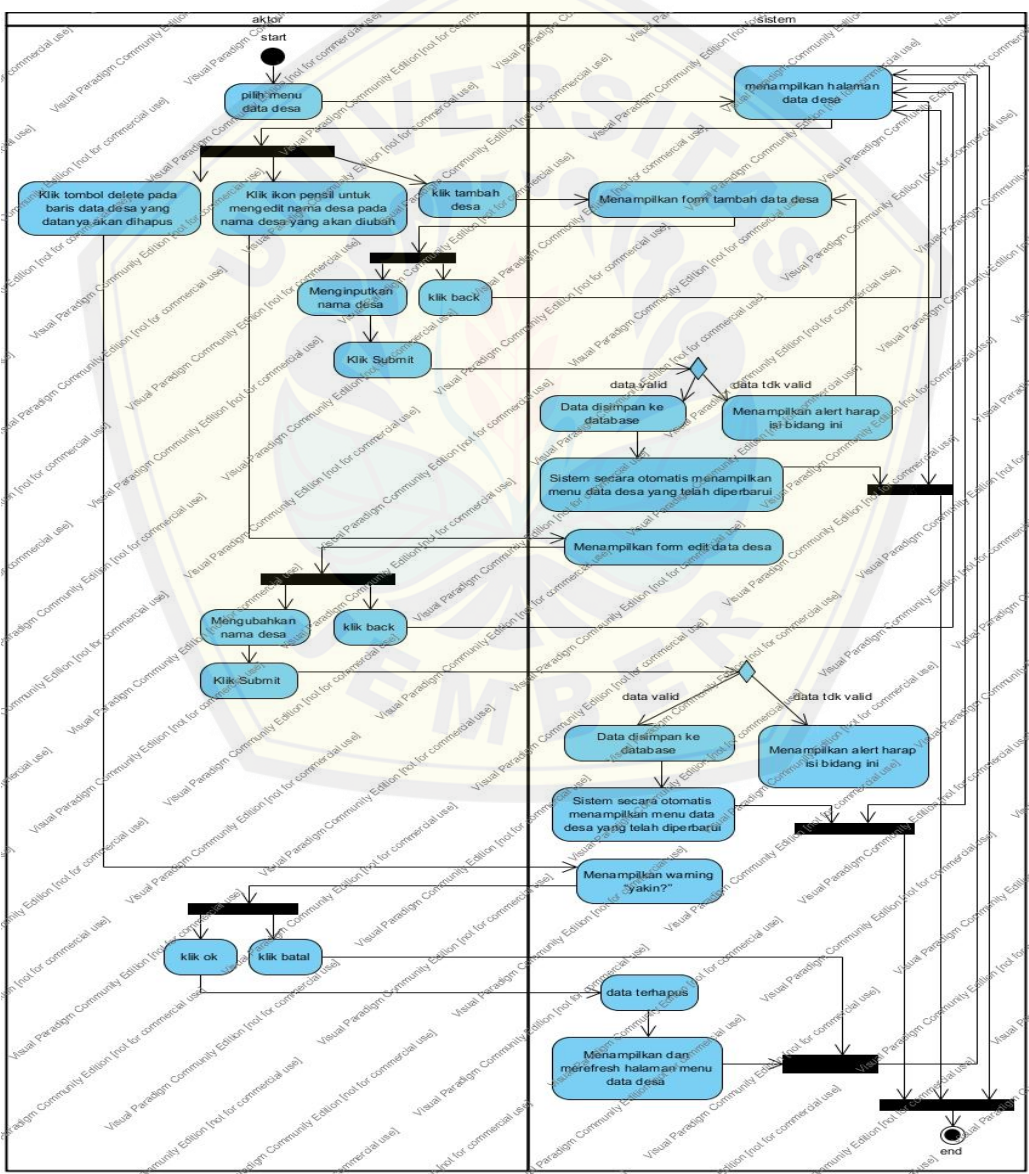




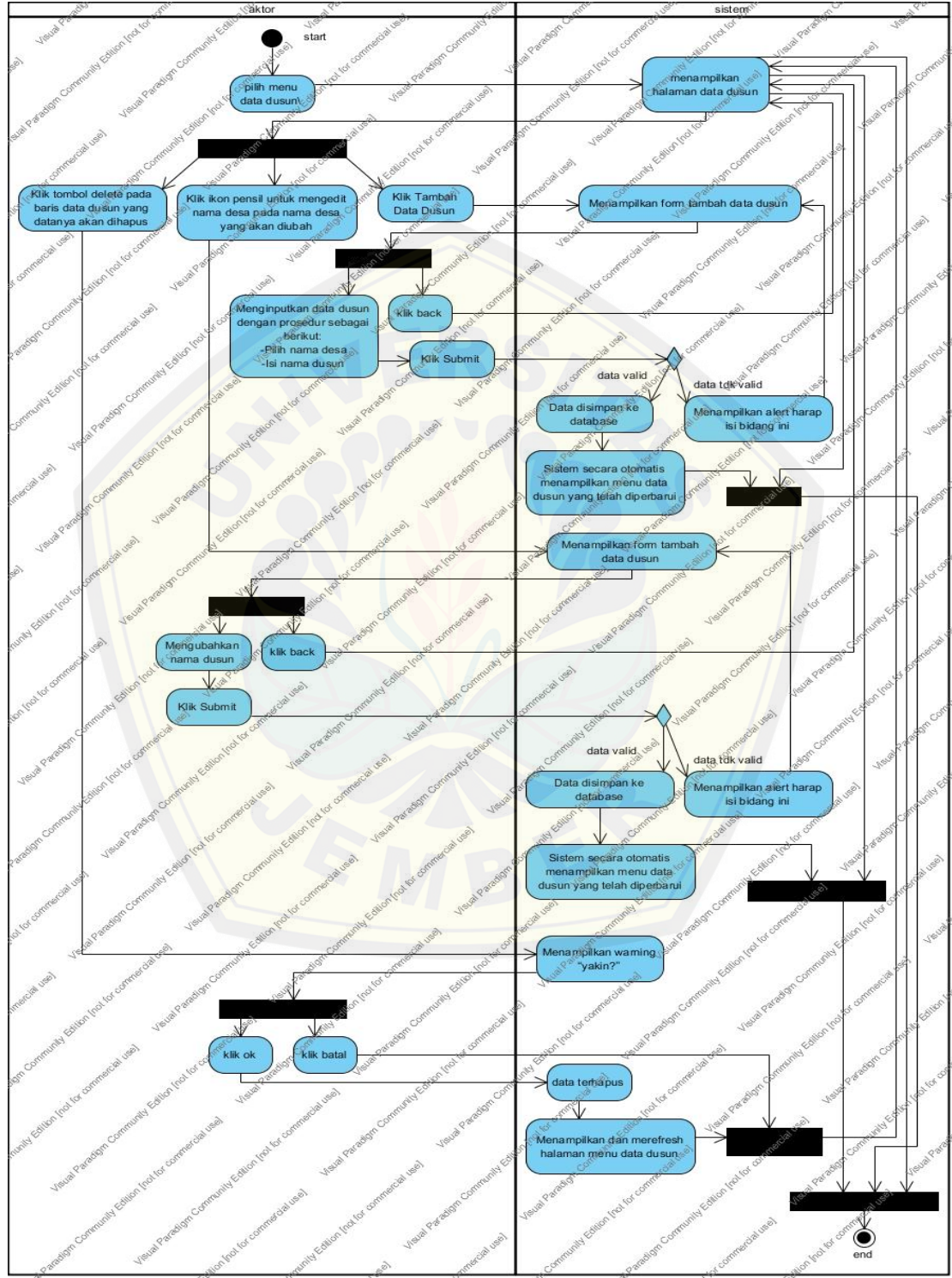
Activity Diagram View Rekap Gizi



Activity Diagram Data Desa



Activity Diagram Data Dusun



**B. Lampiran Kode Program***Function Login* (Controller/admin)

```

public function login() {
    if ($this->input->get_post('name')) {
        $this->load->model('m_login');
        if ($this->m_login->cek_login($this->input->post('name'), $this->input-
>post('pass'))) {
            redirect(base_url() . $this->session->userdata('level'));
        } else {
            redirect(base_url() . "admin/index?status=Username atau Password
salah!");
        }
    }
    redirect(base_url());
}

```

*Function Logout* (Controller/admin)

```

public function logout() {
    $this->session->sess_destroy();
    redirect(base_url());
}

```

*Functiondes*(Controller/bos)

```

public function des($id) {
    if (isset($id)) {
        $data['status'] = $this->model->delete('des', array('id_des' => $id));
    }
    $data['title'] = "Data Desa";
    $data['halaman'] = $this->paging("bos/des", $total, $perpage);
    $data['data'] = $this->model->get_data("v_desa", array(), array(
        "order_by" => array("id_des", "asc"),
        "limit" => array($id, $perpage)
    ));
    $this->page('des', $data);
}

```

*Functionadddes* (Controller/bos)

```

public function adddes($id) {
    if (isset($id)) {

```

```

    $data = $this->model->get_detail("des", array("id_des" => $id));
  }
  $data['title'] = "Tambah Desa";
  $this->page("adddes", $data);
}

```

*Functiondusun* (Controller/bos)

```

public function dusun($id) {
    if (isset($id)) {
        $data['status'] = $this->model->delete('dsn', array('id_dusun' => $id));
    }
    $data['title'] = "Data Dusun";
    $data['halaman'] = $this->paging("bos/dusun", $total, $perpage);
    $data['data'] = $this->model->get_data("v_dusun", array(), array(
        "order_by" => array("id_dusun", "asc")
    ));
    $this->page('dusun', $data);
}

```

*Functionadd\_dusun* (Controller/bos)

```

public function add_dusun($id) {
    if (isset($id)) {
        $data = $this->model->get_detail("dsn", array("id_dusun" => $id));
    }
    $data['title'] = "Tambah Dusun";
    $data['desa'] = $this->model->get_data("des", array());
    $this->page("add_dusun", $data);
}

```

*Functionbaby*(Controller/user)

```

public function baby($id) {
    if (isset($id)) {
        $data['status'] = $this->model->delete('balita', array('id_balita' => $id));
    }
    if($this->input->get_post('filter')){
        $data['filter']=$this->input->post('filter');
        $filter=array("id_jk"=>$data['filter']);
    }else{
        $data['filter']="";
        $filter=array();
    }
}

```

```

}
$data['title'] = "Data Bayi";
$data['data'] = $this->model->get_data("v_balita", $filter, array(
    "order_by" => array("id_balita", "asc")
));

$this->page("baby", $data);
}

```

*Function*add\_baby(Controller/user)

```

public function add_baby($id) {
    if (isset($id)) {
        $data = $this->model->get_detail("balita", array("id_balita" => $id));
    }
    $data['data']=$this->model->get_data("des");
    $data['dat']=$this->model->get_data("dsn");
    $data['title'] = "Tambah bayi";
    $this->page("add_baby", $data);
}

```

*Function*rekap(Controller/user)

```

public function rekap(){

    if($this->input->get_post('filter')){
        $data['filter']=$this->input->post('filter');
        $filter=array("id_jk"=>$data['filter']);
    }else{
        $data['filter']="";
        $filter=array();
    }
    $data['title'] = "rekap gizi";
    $data['data'] = $this->model->get_data("cek", $filter, array(
        "order_by" => array("id_balita", "asc")
    ));
    $this->page("rekap", $data);
}

```

*Function* dat\_training(Controller/simpan)

```

public function dat_training() {
    if ($this->input->get_post('umur')) {

```

```

    $input = array('id_dat_training' => $this->input->post('id_dat_training'),
        'id_jk' => $this->input->post('id_jk'),
        'umur' => $this->input->post('umur'),
        'PB' => $this->input->post('PB'),
        'BB' => $this->input->post('BB'),
        'LLA' => $this->input->post('LLA'),
        'LK' => $this->input->post('LK'),
        'id_status' => $this->input->post('status'));

    $status = $this->model->update('dat_training', $input);
}
redirect(base_url() . "bos/dat_training?status=$status");
}

```

*Functiondes(Controller/simpan)*

```

public function des() {
    if ($this->input->get_post('desa')) {
        $input = array('id_des' => $this->input->post('id_des'),
            'desa' => $this->input->post('desa'),
            'bobot' => $this->input->post('bobot'));
        $status = $this->model->update('des', $input);
    }
    redirect(base_url() . "bos/des?status=$status");
}

```

*Functiondsn(Controller/simpan)*

```

public function dsn() {
    if ($this->input->get_post('dusun')) {
        $input = array('id_dusun' => $this->input->post('id_dusun'),
            'id_des' => $this->input->post('id_des'),
            'dusun' => $this->input->post('dusun'));
        $status = $this->model->update('dsn', $input);
    }
    redirect(base_url() . "bos/dusun?status=$status");
}

```

*Functionbalita(Controller/simpan)*

```

public function balita() {
    if ($this->input->get_post('nama_balita')) {
        $input = array('id_balita' => $this->input->post('id_balita'),

```



```
'nama_balita' => $this->input->post('nama_balita'),
'Tmpt_Lahir' => $this->input->post('Tmpt_Lahir'),
'TglLahir' => $this->input->post('TglLahir'),
'id_jk' => $this->input->post('id_jk'),
'id_jk' => $this->input->post('id_jk'),
'id_jk' => $this->input->post('id_jk'),
'nama_ibu' => $this->input->post('nama_ibu'),
'BB' => $this->input->post('BB'),
'PB' => $this->input->post('PB'),
'LLA' => $this->input->post('LLA'),
'LK' => $this->input->post('LK'));
$status = $this->model->update('balita', $input);
}
redirect(base_url() . "user/baby?status=$status");
}
```

*Function cek\_balita(Controller/simpan)*

```
public function cek_balita() {
    if ($this->input->get_post('nama_balita')) {
        $input = array('id_balita' => $this->input->post('id_balita'),
            'tgl_cek' => $this->input->post('tgl_cek'),
            'BB' => $this->input->post('BB'),
            'PB' => $this->input->post('PB'),
            'LLA' => $this->input->post('LLA'),
            'LK' => $this->input->post('LK'),
            'LD' => $this->input->post('LD'));
        $status = $this->model->update('cek', $input);
    }
    redirect(base_url() . "user/hasil_cek");
}
```

### C. Lampiran Implementasi

#### Halaman Utama Fitur Data Desa

ID Desa	Desa	Jumlah Dusun	Action
1	ajung	0	
2	ambulu	0	
3	arjasa	0	
4	balung	0	
5	bangsalsari	0	
6	gumuk mas	0	
7	jelbuk	0	

#### Form Tambah Data Desa

ID desa:

desa:

#### Form Edit Data Desa

ID desa:

desa:

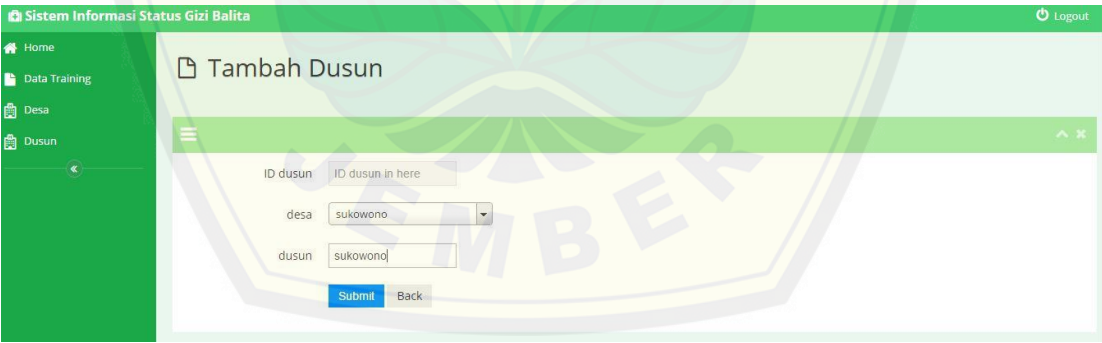
### Tampilan Hapus Data Desa



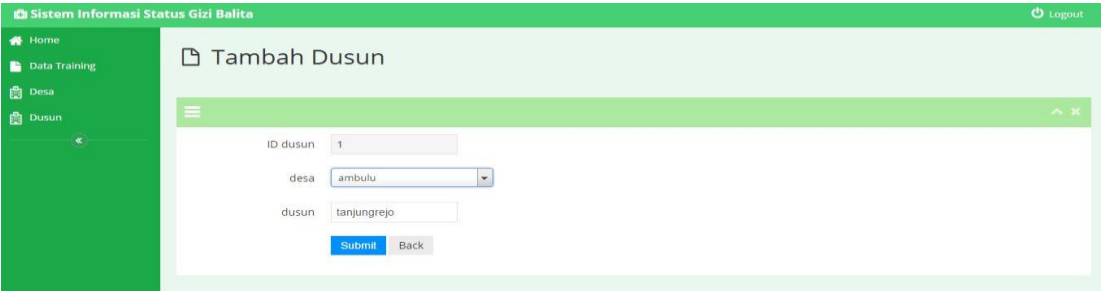
### Halaman Utama Fitur Data Dusun



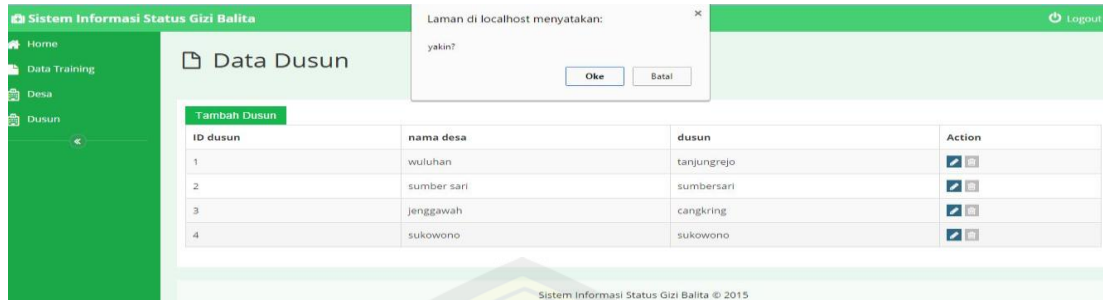
### Form Tambah Data Dusun



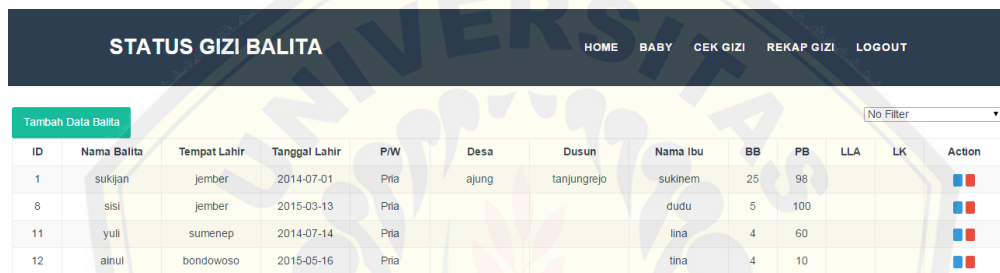
### Form Edit Data Dusun



### Tampilan Hapus Data Dusun



### Halaman Utama Fitur Data Balita



### Form Tambah Data Balita



### Form Edit Data Balita

**STATUS GIZI BALITA**
HOME BABY CEK GIZI REKAP GIZI LOGOUT

---

ID balita  
8

nama balita  
sisi

Tempat Lahir  
jember

Tanggal Lahir  
13/03/2015

Jenis Kelamin Pria (1)

desa ajung dusun tanjungejo nama ibu

dudu

Berat Badan  
5

Panjang Badan  
100

Lingkar Lengan Atas  
10

Lingkar Kepala  
39

[Back](#)

### Tampilan Hapus Data Balita

**STATUS GIZI BALITA**
BABY CEK GIZI REKAP GIZI LOGOUT

Laman di localhost menyatakan:  
yakin?

Tambah Data Balita
No Filter

ID	Nama Balita	Tempat Lahir	Tanggal Lahir	P/W	Desa	Dusun	Nama Ibu	BB	PB	LLA	LK	Action
1	sukijan	jember	2014-07-01	Pria	ajung	tanjungejo	sukinem	25	98			<span style="color: blue;">▶</span> <span style="color: red;">▶</span>
8	sisi	jember	2015-03-13	Wanita			dudu	5	100			<span style="color: blue;">▶</span> <span style="color: red;">▶</span>
11	yuli	sumenep	2014-07-14	Wanita			lina	4	60			<span style="color: blue;">▶</span> <span style="color: red;">▶</span>
12	ainul	bondowoso	2015-05-16	Pria			tina	4	10			<span style="color: blue;">▶</span> <span style="color: red;">▶</span>

### Tampilan Rekap Gizi

**STATUS GIZI BALITA**
HOME BABY CEK GIZI REKAP GIZI LOGOUT

No Filter

No	ID Cek	ID Balita	tanggal cek	BB	PB	LLA	LK	Status
1	4	1	2015-06-18 05:56:27	88	88	18	38	Normal
2	5	1	2015-06-18 06:54:28	90	90	25	40	Gizi Buruk
3	6	1	2015-06-18 06:54:28	88	88	25	42	Over Weigh
4	7	1	2015-06-18 06:54:28	88	88	25	42	Normal
5	8	1	2015-06-18 06:54:28	89	89	25	43	Normal
6	9	1	2015-06-18	100	100	15	50	Over Weigh

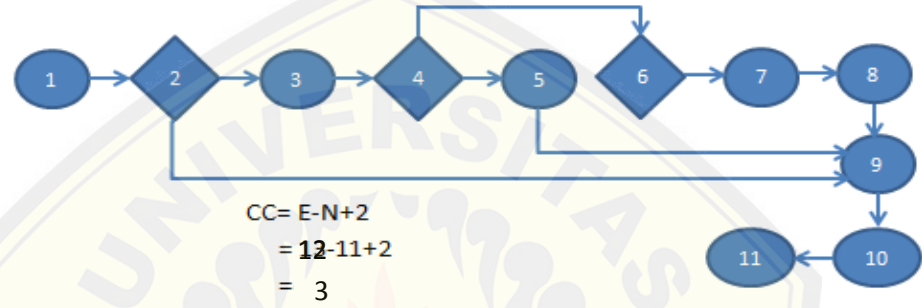
D. Lampiran Pengujian WhiteBox

1. Function Login(Controllers/admin)

```

17 public function login() { 1
18     if ($this->input->get_post('name')) { 2
19         $this->load->model('m_login'); 3
20         if ($this->m_login->cek_login($this->input->post('name'), $this->input->post('pass'))) { 4
21             redirect(base_url() . $this->session->userdata('level')); 5
22         } else { 6
23             redirect(base_url() . "admin/index?status=Username atau Password salah!"); 7
24         } 8
25     } 9
26     redirect(base_url()); 10
27 } 11

```

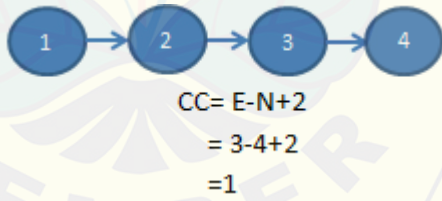


2. Function Logout(Controllers/admin)

```

33 public function logout() { 1
34     $this->session->sess_destroy(); 2
35     redirect(base_url()); 3
36 } 4

```

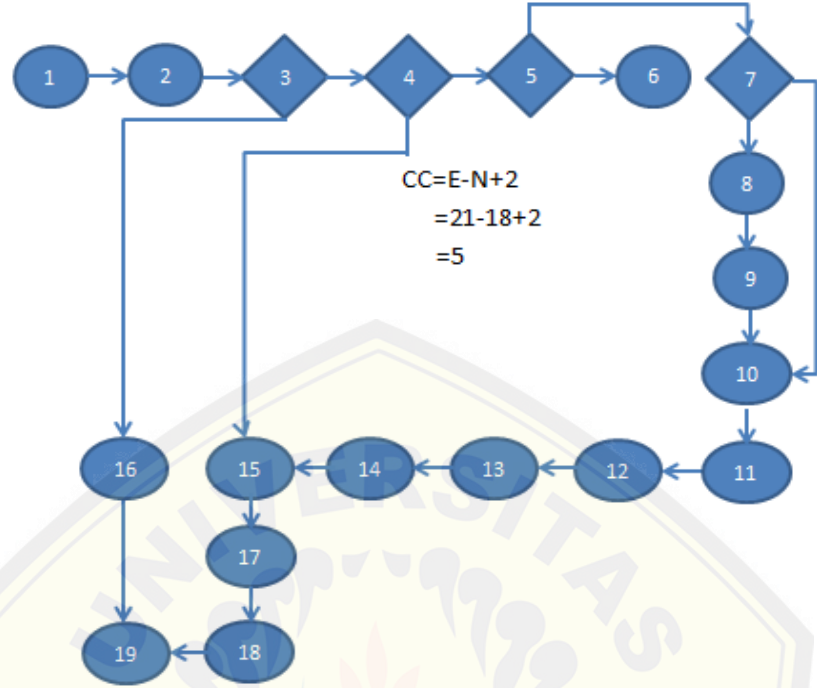


3. Function cek\_login(Models/m\_login)

```

6 public function cek_login($username, $password) { 1
7     $query = $this->db->query("SELECT * FROM `user` WHERE `username`='".$username.'" and `password`='".$md5($password)."'"); 2
8     if ($query->num_rows() > 0) { 3
9         foreach ($query->result_array() as $row) { 4
10            if ($row['level']==1) { 5
11                $this->session->set_userdata('level', 'bos'); 6
12            } elseif ($row['level']==2) { 7
13                $this->session->set_userdata('level', 'user'); 8
14            } 9
15            $this->session->set_userdata('is_admin', TRUE); 10
16            $this->session->set_userdata('user_id', $row['id_user']); 11
17            $this->session->set_userdata('user_name', $row['username']); 12
18            $this->session->set_userdata('user_pass', $row['password']); 13
19        } 14
20        return TRUE; 15
21    } else { 16
22        return FALSE; 17
23    } 18
24 } 19

```

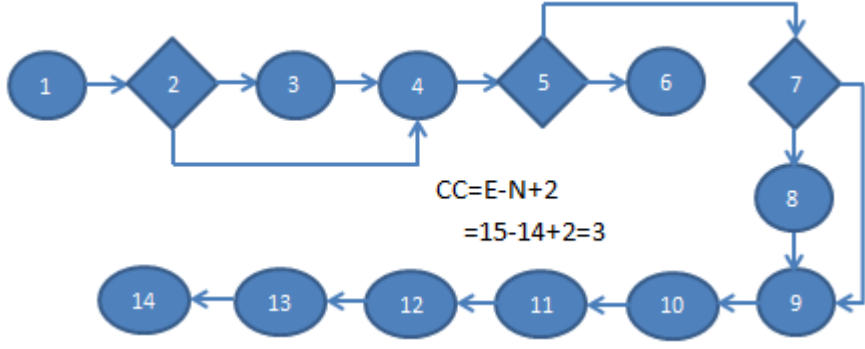


4. Function dat\_training(Controllers/Bos)

```

18 public function dat_training($id) { 1
19     if (isset($id)) { 2
20         $data['status'] = $this->model->delete('dat_training', array('id_dat_training' => $id)); 3
21     } 4
22     if($this->input->get_post('filter')){ 5
23         $data['filter']=$this->input->post('filter');
24         $filter=array("id_jk"=>$data['filter']); 6
25     }else{ 7
26         $data['filter']=""; 8
27         $filter=array();
28     } 9
29     $data['title'] = "Data Training"; 10
30     $data['halaman'] = $this->paging("bos/dat_training", $total, $perpage); 11
31     $data['data'] = $this->model->get_data("dat_training", $filter, array(
32         "order_by" => array("id_dat_training", "asc"),
33         "limit" => array($id, $perpage)
34     )); 12
35     $this->page('dat_training', $data); 13
36 } 14

```



5. Function adddat\_training(Controllers/Bos)

```

37 public function adddat_training($id) { 1
38     if (isset($id)) { 2
39         $data = $this->model->get_detail("dat_training", array("id_dat_training" => $id)); 3
40     } 4
41     $data['title'] = "Tambah Data Training"; 5
42     $this->page("adddat_training", $data); 6
43 } 7

```



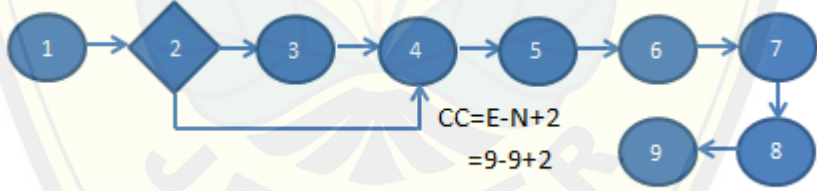
$$\begin{aligned}
 CC &= E - N + 2 \\
 &= 7 - 7 + 2 \\
 &= 2
 \end{aligned}$$

6. Function des(Controllers/Bos)

```

44 public function des($id) { 1
45     if (isset($id)) { 2
46         $data['status'] = $this->model->delete('des', array('id_des' => $id)); 3
47     } 4
48     $data['title'] = "Data Desa"; 5
49     $data['halaman'] = $this->paging("bos/des", $total, $perpage); 6
50     $data['data'] = $this->model->get_data("v_desa", array(), array(
51         "order_by" => array("id_des", "asc"),
52         "limit" => array($id, $perpage)
53     )); 7
54     $this->page('des', $data); 8
55 } 9

```



$$\begin{aligned}
 CC &= E - N + 2 \\
 &= 9 - 9 + 2 \\
 &= 2
 \end{aligned}$$

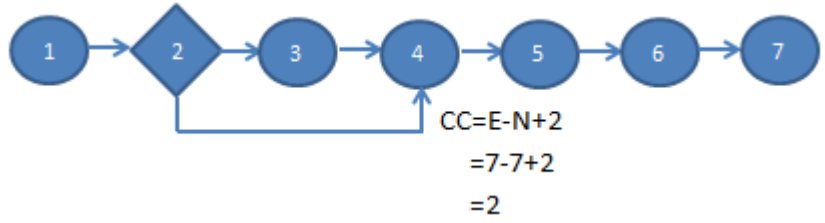
7. Function adddes(Controllers/Bos)

```

57 public function adddes($id) { 1
58     if (isset($id)) { 2
59         $data = $this->model->get_detail("des", array("id_des" => $id)); 3
60     } 4
61     $data['title'] = "Tambah Desa"; 5
62     $this->page("adddes", $data); 6
63 } 7

```

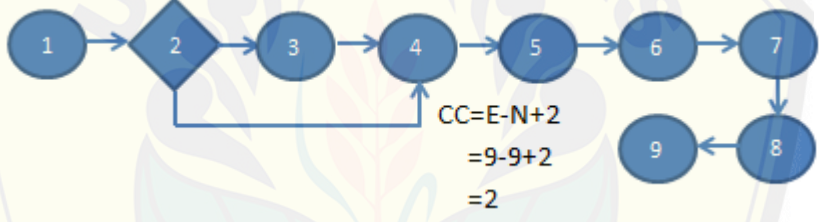




8. Function *delete\_dusun*(Controllers/Bos)

```

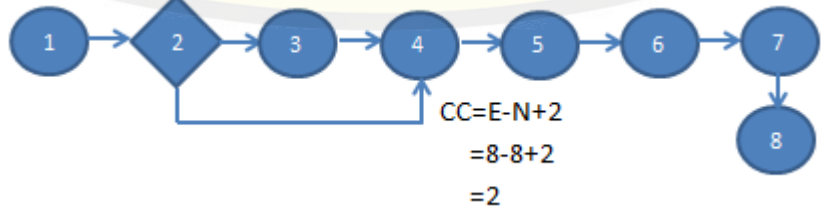
65 public function delete_dusun($id) { 1
66     if (isset($id)) { 2
67         $data['status'] = $this->model->delete('dsn', array('id_dusun' => $id)); 3
68     } 4
69     $data['title'] = "Data Dusun"; 5
70     $data['halaman'] = $this->paging("bos/dusun", $total, $perpage); 6
71     $data['data'] = $this->model->get_data("v_dusun", array(), array(
72         "order_by" => array("id_dusun", "asc")
73     ));
74     $this->page('dusun', $data); 8
75 } 9
  
```



9. Function *add\_dusun*(Controllers/Bos)

```

77 public function add_dusun($id) { 1
78     if (isset($id)) { 2
79         $data = $this->model->get_detail("dsn", array("id_dusun" => $id)); 3
80     } 4
81     $data['title'] = "Tambah Dusun"; 5
82     $data['desa'] = $this->model->get_data("des", array()); 6
83     $this->page("add_dusun", $data); 7
84 } 8
  
```

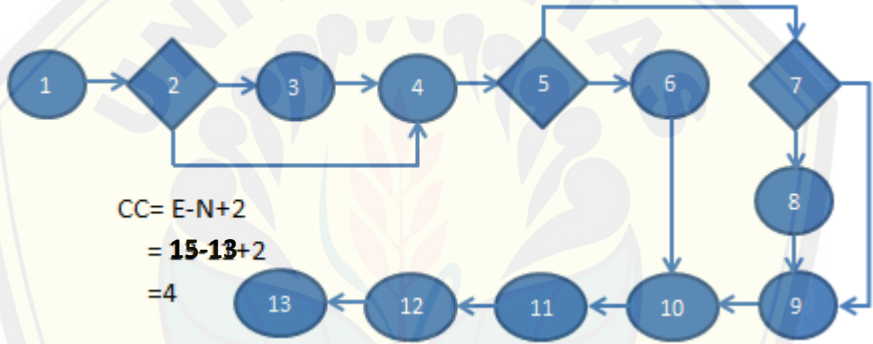


10. Function babyControllers/user)

```

18 public function baby($id) { 1
19     if (isset($id)) { 2
20         $data['status'] = $this->model->delete('balita', array('id_balita' => $id)); 3
21     } 4
22     if($this->input->get_post('filter')){ 5
23         $data['filter']=$this->input->post('filter');
24         $filter=array("id_jk"=>$data['filter']); 6
25     }else{ 7
26         $data['filter']="";
27         $filter=array(); 8
28     } 9
29     $data['title'] = "Data Bayi"; 10
30     $data['data'] = $this->model->get_data("v_balita", $filter, array(
31         "order_by" => array("id_balita", "asc")
32     )); 11
33     $this->page("baby", $data); 12
34 } 13

```

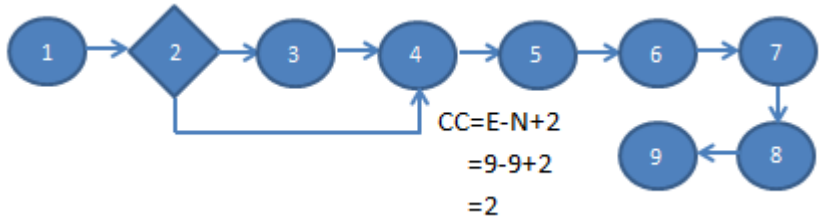


11. Function add\_baby(Controllers/user)

```

36 public function add_baby($id) { 1
37     if (isset($id)) { 2
38         $data = $this->model->get_detail("balita", array("id_balita" => $id)); 3
39     } 4
40     $data['data']=$this->model->get_data("des"); 5
41     $data['dat']=$this->model->get_data("dsn"); 6
42     $data['title'] = "Tambah bayi"; 7
43     $this->page("add_baby", $data); 8
44 } 9

```

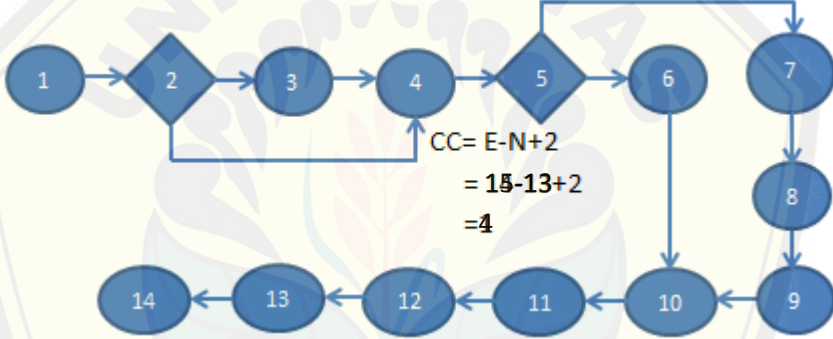


12. Function cek\_gizi (Controllers/user)

```

1 public function cek_gizi($id) { 1
46 if (isset($id)) { 2
47     $data['status'] = $this->model->delete('cek', array('id_balita' => $id)); 3
48 } 4
49 if($this->input->get_post('filter')){ 5
50     $data['filter']=$this->input->post('filter'); 6
51     $filter=array("id_jk"=>$data['filter']);
52 }else{ 7
53     $data['filter']="";
54     $filter=array(8
55 ) 9
56 $data['title'] = "Data Bayi"; 10
57 $data['data'] = $this->model->get_data("balita", $filter, array(
58     "order_by" => array("id_balita", "asc")
59 )); 11
60 $data['desa'] = $this->model->get_data("des"); 12
61 $this->page("cek_gizi", $data); 13
62 } 14

```

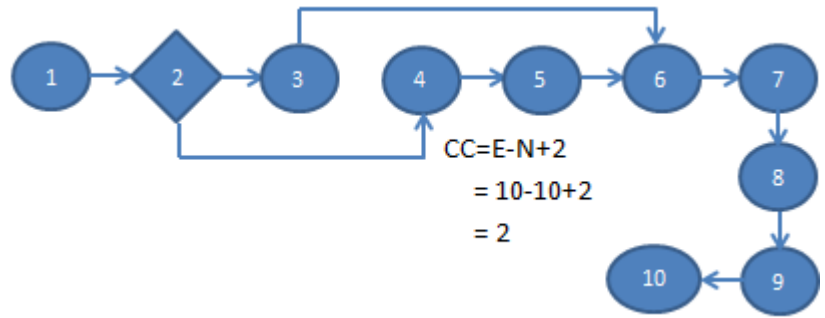


13. Function rekap(Controllers/user)

```

93 public function rekap(){ 1
94     if($this->input->get_post('filter')){ 2
95         $data['filter']=$this->input->post('filter');
96         $filter=array("id_jk"=>$data['filter']); 3
97     }else{ 4
98         $data['filter']="";
99         $filter=array(); 5
100     } 6
101     $data['title'] = "rekap gizi"; 7
102     $data['data'] = $this->model->get_data("cek", $filter, array(
103         "order_by" => array("id_balita", "asc")
104     ));
105     $this->page("rekap", $data); 9
106 } 10

```

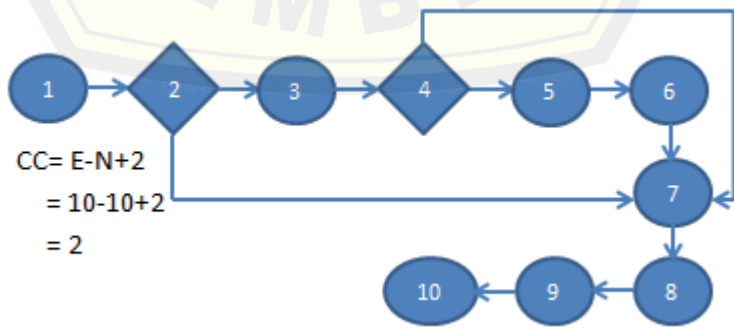


14. Function proses\_cek(Controllers/user)

```

64 public function proses_cek($id) { 1
65     if($this->input->get_post('BB')){ 2
66         $data['dt']=$this->model->hitung_dt($this->input->post('id_balita'));
67         $data['du']=$this->model->hitung_du(array(
68             'id_balita'=>$this->input->post('id_balita'),
69             'BB'=>$this->input->post('BB'),
70             'PB'=>$this->input->post('PB'),
71             'LLA'=>$this->input->post('LLA'),
72             'LK'=>$this->input->post('LK'));
73         $data['hasil']=$this->model->hitung($data['dt'],$data['du'],$this->input->post('k'));
74         $this->page("hasil_cek", $data); 3
75     // redirect(base_url("user/rekap")); //buat muculin proses kalo d comment
76     // SIMPAN KE TABEL CEK
77     if ($this->input->get_post('nama_balita')) { 4
78         $input = array('id_balita' => $this->input->post('id_balita'),
79             'tgl_cek' => $this->input->post('tgl_cek'),
80             'BB' => $this->input->post('BB'),
81             'PB' => $this->input->post('PB'),
82             'LLA' => $this->input->post('LLA'),
83             'LK' => $this->input->post('LK'),
84             'status' => $data['hasil']['statusnya']);
85         $status = $this->model->update('cek', $input); 5
86     } 6
87     }else{ 7
88         $data = $this->model->get_detail("balita", array("id_balita"=>$id));
89         $this->page("proses_cek", $data); 8
90     } 9
91 } 10

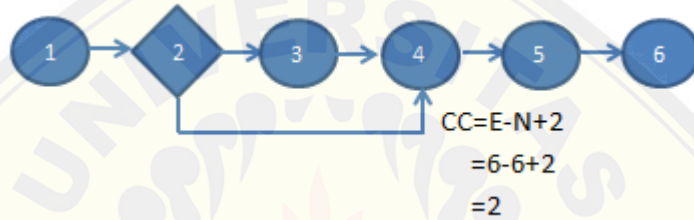
```



15. *Function dat\_training(Controllers/simpan)*

```

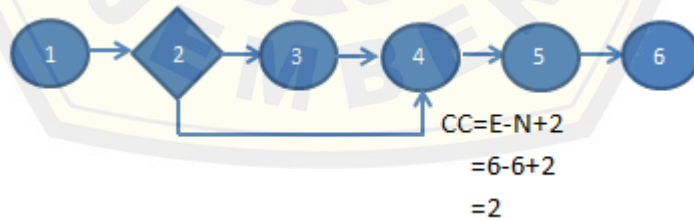
26 public function dat_training() { 1
27     if ($this->input->get_post('umur')) { 2
28         $input = array('id_dat_training' => $this->input->post('id_dat_training'),
29             'id_jk' => $this->input->post('id_jk'),
30             'umur' => $this->input->post('umur'),
31             'PB' => $this->input->post('PB'),
32             'BB' => $this->input->post('BB'),
33             'LLA' => $this->input->post('LLA'),
34             'LK' => $this->input->post('LK'),
35             'id_status' => $this->input->post('status'));
36
37         $status = $this->model->update('dat_training', $input); 3
38     } 4
39     redirect(base_url() . "bos/dat_training?status=$status"); 5
} 6
    
```



16. *Function des(Controllers/simpan)*

```

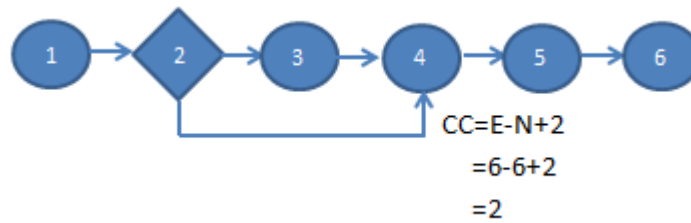
41 public function des() { 1
42     if ($this->input->get_post('desa')) { 2
43         $input = array('id_des' => $this->input->post('id_des'),
44             'desa' => $this->input->post('desa'));
45         $status = $this->model->update('des', $input); 3
46     } 4
47     redirect(base_url() . "bos/des?status=$status"); 5
48 } 6
    
```



17. *Function dsn(Controllers/simpan)*

```

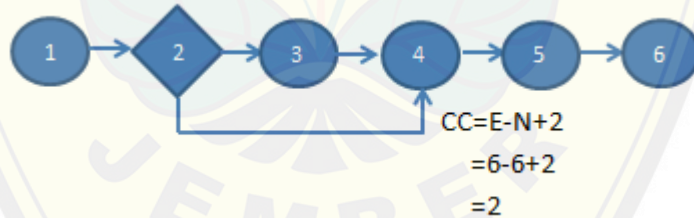
50 public function dsn() { 1
51     if ($this->input->get_post('dusun')) { 2
52         $input = array('id_dusun' => $this->input->post('id_dusun'),
53             'id_des' => $this->input->post('id_des'),
54             'dusun' => $this->input->post('dusun'));
55         $status = $this->model->update('dsn', $input); 3
56     } 4
57     redirect(base_url() . "bos/dusun?status=$status"); 5
58 } 6
    
```



18. *Function balita* (Controllers/simpan)

```

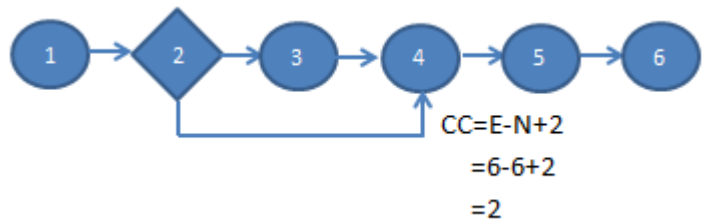
60 public function balita() { 1
61     if ($this->input->get_post('nama_balita')) { 2
62         $input = array('id_balita' => $this->input->post('id_balita'),
63             'nama_balita' => $this->input->post('nama_balita'),
64             'Tmpt_Lahir' => $this->input->post('Tmpt_Lahir'),
65             'TglLahir' => $this->input->post('TglLahir'),
66             'id_jk' => $this->input->post('id_jk'),
67             'id_jk' => $this->input->post('id_jk'),
68             'id_jk' => $this->input->post('id_jk'),
69             'nama_ibu' => $this->input->post('nama_ibu'),
70             'BB' => $this->input->post('BB'),
71             'PB' => $this->input->post('PB'),
72             'LLA' => $this->input->post('LLA'),
73             'LK' => $this->input->post('LK'));
74         $status = $this->model->update('balita', $input); 3
75     } 4
76     redirect(base_url() . "user/baby?status=$status"); 5
77 } 6
    
```



19. *Function cek\_balita*(Controllers/simpan)

```

78 public function cek_balita() { 1
79     if ($this->input->get_post('nama_balita')) { 2
80         $input = array('id_balita' => $this->input->post('id_balita'),
81             'tgl_cek' => $this->input->post('tgl_cek'),
82             'BB' => $this->input->post('BB'),
83             'PB' => $this->input->post('PB'),
84             'LLA' => $this->input->post('LLA'),
85             'LK' => $this->input->post('LK'));
86         $status = $this->model->update('cek', $input); 3
87     } 4
88     redirect(base_url() . "user/hasil_cek"); 5
89 } 6
    
```

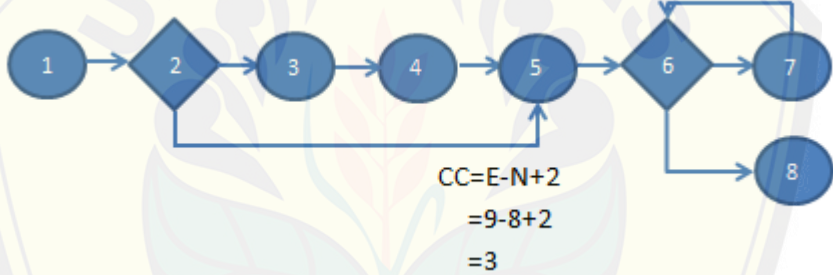


20. Function get\_detail (Models/model)

```

18 public function get_detail($table, $cond) {
19     if (isset($cond)) { 2
20         $this->db->where($cond); 3
21     } 4
22     $query = $this->db->get($table); 5
23     foreach ($query->result_array() as $r)
24         ; 6
25     return $r; 7
26 } 8

```

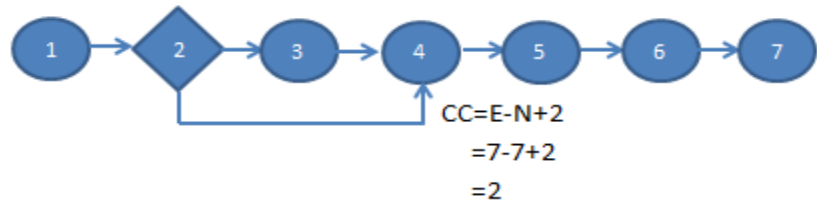


21. Function delete(Models/model)

```

69 public function delete($table, $cond) 1 {
70     if (isset($cond)) { 2
71         $this->db->where($cond); 3
72     } 4
73     $query = $this->db->delete($table); 5
74     return $query; 6
75 } 7

```

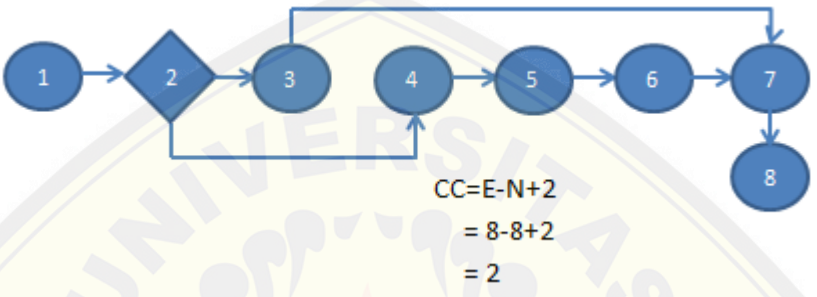


22. Function update(Models/model)

```

77 public function update($table, $insert) { 1
78     if (trim($insert["id_$table"]) == "") { 2
79         $query = $this->db->insert($table, $insert); 3
80     } else { 4
81         $this->db->where(array("id_$table" => $insert["id_$table"]));
82         $query = $this->db->update($table, $insert); 5
83     } 6
84     return $query; 7
85 } 8

```



23. Function get\_data(Models/model)

```

44 public function get_data($table, $cond, $no, $perpage) { 1
45     if (isset($cond)) { 2
46         $this->db->where($cond); 3
47     } 4
48     if (is_array($no)) { 5
49         if (isset($no['order_by'])) { 6
50             $this->db->order_by($no['order_by'][0], $no['order_by'][1]); 7
51         } 8
52         if (isset($no['group_by'])) { 9
53             $this->db->group_by($no['group_by']); 10
54         } 11
55         if (isset($no['limit'])) { 12
56             $query = $this->db->get($table, $no['limit'][1], $no['limit'][1]); 13
57         } else { 14
58             $query = $this->db->get($table); 15
59         } 16
60     } else { 17
61         if (isset($no) && isset($perpage)) { 18
62             $query = $this->db->get($table, $perpage, $no); 19
63         } else { 20
64             $query = $this->db->get($table); 21
65         } 22
66     } 23
67     return $query; 24
} 25

```





### E. Lampiran Pengujian *Black Box*

#### Pengujian *Black Box* Login

Login		Status	
Aktor	Sistem	Berhasil	Gagal
1. Membuka sistem cek gizi			
	2. Menampilkan form log in dengan field yang tersedia antara lain : - <i>Username</i> (input text: harus diisi) - <i>Password</i> (input password: harus diisi )		
3. Mengisi form login yang tersedia ( <i>username</i> , <i>password</i> )			
4. Klik tombol login			
	5. Verifikasi <i>username</i> dan <i>password</i>		
	6. Menampilkan menu yang tersedia sesuai dengan jenis <i>user</i>		
<b>Username/password salah</b>			
3. Salah memasukkan <i>Username</i> yang tidak tersedia atau <i>password</i> yang salah			
4. Klik tombol login			
	5. Verifikasi <i>username</i> dan <i>password</i>		
	6. Menampilkan warning message " <i>username/password</i> salah"		
	7. Menampilkan		

	kembali halaman login		
<b>Hanya mengisi salah satu field/tidak mengisi</b>			
3. Tidak mengisi /hanya mengisi salah satu field			
4. Klik tombol login			
	5. Menampilkan form login dan warning "harap isi bidang ini"		

#### Pengujian *Black Box* Data Balita

Data balita		Status	
Aktor	Sistem	Berhasil	Gagal
1. Melakukan login			
	2. Menampilkan halaman <i>user</i> dengan menu <ul style="list-style-type: none"> <li>- Baby</li> <li>- Cek gizi</li> <li>- Rekap gizi</li> <li>- Logout</li> </ul>		
3. Memilih menu baby			
	4. Menampilkan tabel nama balita, submenu filter tampilan sesuai jenis kelamin, tombol tambah data balita, serta tombol <i>edit</i> dan hapus pada tiap nama balita		
<b>Skenario Input Data Balita</b>			
5. Klik Tambah Data Balita			
	6. Menampilkan form tambah data balita yang berisi <i>field</i> : <ul style="list-style-type: none"> <li>- Id balita</li> <li>- Nama balita (<i>input</i> nama: harus diisi)</li> </ul>		

	<ul style="list-style-type: none"> <li>- Tempat lahir (<i>input</i> nama: harus diisi)</li> <li>- Tanggal lahir (jquery: harus dipilih)</li> <li>- Jenis kelamin (jquery: harus dipilih)</li> <li>- Desa (jquery: harus dipilih)</li> <li>- Dusun (jquery: harus dipilih)</li> <li>- Nama ibu (<i>input</i> nama: harus diisi)</li> <li>- Kondisi awal balita (berupa BB, TB, LLA, LK) (<i>input</i> nomor: harus diisi)</li> </ul>		
<p>7. Menginputkan data balita dengan prosedur sebagai berikut:</p> <ul style="list-style-type: none"> <li>- Isi Nama balita tsb</li> <li>- Isi nama tempat lahir balita tsb</li> <li>- Pilih Tanggal lahir sesuai kelahiran balita tsb</li> <li>- Pilih Jenis kelamin balita tsb</li> <li>- Pilih Nama Desa balita tsb</li> <li>- Pilih Nama Dusun balita tsb</li> <li>- Isi Nama ibu dari balita tsb</li> <li>- Isi Kondisi awal balita (berupa BB, TB, LLA, LK)</li> </ul>			
8. Klik <i>Submit</i>			
	9. Data disimpan ke database		

	10. Sistem secara otomatis menampilkan menu bayi yang telah diperbarui		
<b>Input data dengan format salah</b>			
7. Menginputkan dengan format yang tidak sesuai			
	8. Sistem menampilkan peringatan data yang diinputkan tidak sesuai format pada sisi field yang salah format		
	9. Menampilkan kembali form tambah data balita		
<b>Data tidak lengkap</b>			
10b. Tidak mengisi data dengan lengkap			
11b. Klik <i>Submit</i>			
	12b. Menampilkan kembali form input data <i>user</i> dan <i>alert</i> harap isi bidang ini		
<b>Batal saat sedang menginputkan data balita</b>			
7. Klik <i>back</i>			
	8. Menampilkan kembali halaman menu bayi		

<b>Skenario Edit Data Balita</b>			
5. Klik tombol <i>edit</i> pada baris nama balita yang datanya akan <i>diedit</i>			
	6. Menampilkan form <i>edit</i> data balita dengan data sebelumnya		

7. Menginputkan data pada bagaian yang akan <i>diedit</i>			
8. Klik <i>Submit</i>			
	9. Data disimpan ke database		
	10. Sistem secara otomatis menampilkan menu bayi yang telah diperbarui		
<b>Edit data dengan format salah</b>			
7. Mengeditkan dengan format yang tidak sesuai			
	8. Sistem menampilkan peringatan data yang <i>diedit</i> kan tidak sesuai format pada sisi field yang salah format		
<b>Data tidak lengkap</b>			
7. Tidak mengisi data dengan lengkap			
8. Klik <i>Submit</i>			
	9. Menampilkan kembali form <i>edit</i> data <i>user</i> dan <i>alert</i> harap isi bidang ini		
<b>Batal saat sedang mengeditkan data balita</b>			
9c. Klik <i>back</i>			
	10c. Menampilkan kembali halaman menu bayi		

<b>Skenario Hapus Data Balita</b>			
5. Klik tombol delete pada baris nama balita yang datanya akan dihapus			
	6. Menampilkan		

	warning “yakin?”		
7. Klik ok			
	8. Data terhapus		
	9. Menampilkan dan merefresh halaman menu baby		
<b>Batal hapus</b>			
8a. Klik batal			
	9. Menampilkan kembali menu bayi		

Pengujian *Black Box* Cek Gizi

Cek Status Gizi		Status	
Aktor	Sistem	Berhasil	Gagal
1. Melakukan Login			
	2. Menampilkan halaman <i>user</i> dengan menu <ul style="list-style-type: none"> <li>- Baby</li> <li>- Cek gizi</li> <li>- Rekap gizi</li> <li>- Logout</li> </ul>		
3. Memilih menu Cek Gizi			
	4. Menampilkan tabel nama bayi, submenu filter desa dan dusun, dan tambah data bayi bila data bayi belum ada.		
5. Klik tombol cek gizi pada baris nama bayi yang akan dicek			
	6. Menampilkan form data <i>user</i> yang berisi field: <ul style="list-style-type: none"> <li>- Nama balita</li> <li>- Tanggal cek</li> <li>- Kondisi balita saat dicek,</li> </ul>		

	<i>user</i> menginputkan data balita berupa BB, TB, LLA, LK		
7. Menginputkan kondisi balita yang dicek			
8. Klik cek			
	<p>9. Menampilkan proses perhitungan sebagai berikut:</p> <ul style="list-style-type: none"> <li>- Menampilkan data <i>training</i></li> <li>- Menampilkan nilai kondisi yang telah dinormalisasi</li> <li>- Menghitung jarak terdekat dari tiap data <i>training</i></li> <li>- Mengurutkan data yang terdekat</li> <li>- Mengambil kesimpulan sesuai k yang ditentukan</li> </ul>		
	10. Menampilkan data kondisi dan status gizi balita yang dicek, serta tombol cek balita lainnya atau keluar		
11. Klik cek			
	12. Data disimpan ke database		
	13. Sistem secara otomatis menampilkan menu cek gizi bayi		
<b>Batal mengecek karena kesalahan data yang akan dicek</b>			
7. Klik <i>back</i>			
	8. Menampilkan kembali halaman cek gizi bayi		



<b>Input data dengan format salah</b>			
7. Menginputkan dengan format yang tidak sesuai			
	8. Sistem menampilkan peringatan data yang diinputkan tidak sesuai format pada sisi field yang salah format		
<b>Data tidak lengkap</b>			
7. Tidak mengisi data dengan lengkap			
8. Klik cek			
	9. Menampilkan kembali form cek gizi		
<b>Batal cek karena salah penginputan nilai kondisi</b>			
11. Klik <i>back</i>			
	12. Menampilkan form data <i>user</i> yang berisi field: <ul style="list-style-type: none"> <li>- Nama balita</li> <li>- Tanggal cek</li> <li>- Kondisi balita saat dicek, <i>user</i> menginputkan data balita berupa BB, TB, LLA, LK</li> </ul>		
<b>Data direkap</b>			
11. Klik rekap			
	18a. Data disimpan ke database		
	19a. Menampilkan rekap data bayi yang dipilih		

Pengujian *Black Box* View Rekap Gizi

View Rekap Gizi		Status	
Aktor	Sistem	Berhasil	Gagal
1. Melakukan Login			
	2. Menampilkan halaman <i>user</i> dengan menu - Baby - Cek gizi - Rekap gizi - Logout		
3. Memilih menu Rekap Gizi			
	4. Menampilkan tabel data gizi balit		
5. Klik detail dari baris nama bayi yang akan direkap pada dusun dan desa yang dipilih			
	6. Menampilkan rekap gizi dari bayi yang dipilih		

Pengujian *Black Box* Data Desa

Data Desa		Status	
Aktor	Sistem	Berhasil	Gagal
1. Melakukan login			
	2. Menampilkan halaman <i>admin</i> dengan menu - Home - Data <i>training</i> - Desa - Dusun		
3. Memilih menu desa			
	4. Menampilkan tabel data desa dan tombol tambah data desa		

<b>Input Data Desa</b>			
5. Klik Tambah Data Desa			
	6. Menampilkan form tambah data desa yang berisi <i>field</i> : - Id desa - Desa ( <i>input</i> nama: harus diisi)		
7. Menginputkan nama desa			
8. Klik <i>Submit</i>			
	9. Data disimpan ke database		
	10. Sistem secara otomatis menampilkan menu data desa yang telah diperbarui		
<b>Input data dengan format salah</b>			
7. Menginputkan dengan format yang tidak sesuai			
	8. Sistem menampilkan peringatan data yang diinputkan tidak sesuai format pada sisi <i>field</i> yang salah format		
<b>Data tidak lengkap</b>			
7. Tidak mengisi data dengan lengkap			
8. Klik <i>Submit</i>			
	9. Menampilkan kembali form input data desa dan <i>alert</i> harap isi bidang ini		
<b>Batal saat sedang menginputkan data desa</b>			
7. Klik <i>back</i>			
	8. Menampilkan kembali halaman menu data desa		

<b>Skenario Edit Data Desa</b>			
5. Klik ikon pensil untuk mengedit nama desa pada nama desa yang akan diubah			
	6. Menampilkan form <i>edit</i> data desa yang berisi <i>field</i> : - Id desa - Desa ( <i>input</i> nama: harus diisi)		
7. Mengubah nama desa			
8. Klik <i>Submit</i>			
	9. Data disimpan ke database		
	10. Sistem secara otomatis menampilkan menu data desa yang telah diperbarui		
<b>Edit data dengan format salah</b>			
7. Mengubah dengan format yang tidak sesuai			
	8. Sistem menampilkan peringatan data yang diinputkan tidak sesuai format pada sisi <i>field</i> yang salah format		
<b>Data tidak lengkap</b>			
7. Tidak mengisi data dengan lengkap			
8. Klik <i>Submit</i>			
	9. Menampilkan kembali form <i>edit</i> data desa dan <i>alert</i> harap isi bidang ini		

<b>Batal saat sedang menginputkan data desa</b>			
7. Klik <i>back</i>			
	8. Menampilkan kembali halaman menu data dusun		

<b>Skenario Hapus Data Desa</b>			
5. Klik tombol delete pada baris data desa yang datanya akan dihapus			
	6. Menampilkan warning “yakin?”		
7. Klik ok			
	8. Data terhapus		
	9. Menampilkan dan mereshfresh halaman menu data desa		
<b>Batal hapus</b>			
7. Klik batal			
	8. Menampilkan kembali menu data desa		

#### Pengujian *Black Box* Data Dusun

<b>Data Dusun</b>		<b>Status</b>	
Aktor	Sistem	Berhasil	Gagal
1. Melakukan login			
	2. Menampilkan halaman <i>admin</i> dengan menu <ul style="list-style-type: none"> <li>- Home</li> <li>- Data <i>training</i></li> <li>- Desa</li> <li>- Dusun</li> </ul>		
3. Memilih menu desa			
	4. Menampilkan tabel data desa dan tombol tambah data desa		

<b>Input Data Desa</b>			
5. Klik Tambah Data Dusun			
	6. Menampilkan form tambah data dusun yang berisi <i>field</i> : - Id dusun - Desa (jquery: harus dipilih) - Dusun ( <i>input</i> nama: harus diisi)		
7. Menginputkan data dusun dengan prosedur sebagai berikut: - Pilih nama desa - Isi nama dusun			
8. Klik <i>Submit</i>			
	9. Data disimpan ke database		
	10. Sistem secara otomatis menampilkan menu data dusun yang telah diperbarui		
<b>Input data dengan format salah</b>			
7. Menginputkan dengan format yang tidak sesuai			
	8. Sistem menampilkan peringatan data yang diinputkan tidak sesuai format pada sisi field yang salah format		
<b>Data tidak lengkap</b>			
7. Tidak mengisi data dengan lengkap			
8. Klik <i>Submit</i>			
	9. Menampilkan kembali form input data dusun dan		

	<i>alertharap</i> isi bidang ini		
<b>Batal saat sedang menginputkan data dusun</b>			
7. Klik <i>back</i>			
	8. Menampilkan kembali halaman menu data dusun		

<b>Skenario Edit Data Desa</b>			
5. Klik ikon pensil untuk mengedit nama dusun pada nama dusun yang akan diubah			
	6. Menampilkan form tambah data dusun yang berisi <i>field</i> : - Id dusun - Desa (jquery: harus dipilih) - Dusun ( <i>input</i> nama: harus diisi		
7. Mengubahkan nama dusun			
8. Klik <i>Submit</i>			
	9. Data disimpan ke database		
	10. Sistem secara otomatis menampilkan menu data desa yang telah diperbarui		
<b>Edit data dengan format salah</b>			
7. Mengubahkan dengan format yang tidak sesuai			
	8. Sistem menampilkan peringatan data yang diinputkan tidak sesuai format pada sisi field yang salah		

	format		
<b>Data tidak lengkap</b>			
7. Tidak mengisi data dengan lengkap			
8. Klik <i>Submit</i>			
	9. Menampilkan kembali form <i>edit</i> data dusun dan <i>alert</i> harap isi bidang ini		
<b>Batal saat sedang menginputkan data desa</b>			
7. Klik <i>back</i>			
	8. Menampilkan kembali halaman menu data dusun		

<b>Skenario Hapus Data Desa</b>			
5. Klik tombol delete pada baris data dusun yang datanya akan dihapus			
	6. Menampilkan warning “yakin?”		
7. Klik ok			
	8. Data terhapus		
	9. Menampilkan dan mererefresh halaman menu data dusun		
<b>Batal hapus</b>			
7. Klik batal			
	8. Menampilkan kembali menu data desa		



## F. Lampiran Data Set

Tabel data gizi anak laki-laki

No.	Umur	Berat Badan	Tinggi Badan	Lingkar Lengan Atas	Lingkar Kepala	Klasifikasi st. Gizi menurut who
1	36	12.6	92.5	25	40	<b>BAIK</b>
2	36	12.2	88	24	42	<b>LEBIH</b>
3	16	10	81	18	30	<b>LEBIH</b>
4	31	11.4	87.8	23	47	<b>LEBIH</b>
5	29	11	82	22	40	<b>LEBIH</b>
6	34	12.5	94.2	24	50	<b>LEBIH</b>
7	24	12	86	21	40	<b>BURUK</b>
8	26	12	88.5	21	40	<b>BAIK</b>
9	17	10.6	77.7	18	38	<b>BAIK</b>
10	18	9.5	78	10	43	<b>BAIK</b>
11	24	13	86	16	40	<b>BAIK</b>
12	16	11.8	73	18	35	<b>BAIK</b>
13	30	12.5	88	25	42	<b>BAIK</b>
14	19	7.6	73	18	38	<b>BAIK</b>
15	20	10.1	82	16	38	<b>LEBIH</b>
16	17	14	75	17	38	<b>BAIK</b>
17	20	11.6	72	15	40	<b>BAIK</b>
18	36	11.5	80.5	25	47	<b>LEBIH</b>
19	23	11	78	20	38	<b>BAIK</b>
20	28	9	71	20	43	<b>LEBIH</b>
21	18	9.4	75	15	40	<b>LEBIH</b>
22	28	12	80	17	45	<b>LEBIH</b>
23	29	10	81	19	47	<b>LEBIH</b>
24	29	10	88	19	44	<b>BAIK</b>
25	33	12	92	23	50	<b>BAIK</b>
26	30	12	76.5	20	45	<b>BAIK</b>
27	23	10	77	24	42	<b>BURUK</b>
28	25	10	83	15	41	<b>BAIK</b>
29	36	10	86	25	47	<b>BAIK</b>
30	19	7.8	71.7	15	40	<b>LEBIH</b>
31	21	11	82.5	22	39	<b>LEBIH</b>
32	30	13	83	22	47	<b>LEBIH</b>
33	20	8.1	82	19	39	<b>BAIK</b>
34	15	9	76	15	33	<b>LEBIH</b>

35	16	9.2	73	20	37	<b>LEBIH</b>
36	12	8.7	75	15	35	<b>LEBIH</b>
37	33	13	90	24	49	<b>BAIK</b>
38	16	11.6	74	19	41	<b>LEBIH</b>
39	24	9.1	79	20	44	<b>BAIK</b>
40	18	7.2	74	15	39	<b>BAIK</b>
41	16	9.5	76	16	36	<b>LEBIH</b>
42	21	9.9	76.5	21	49	<b>BAIK</b>
43	18	8.8	72	23	39	<b>LEBIH</b>
44	29	12.5	85.2	21	50	<b>LEBIH</b>
45	15	9.5	70	16	40	<b>BURUK</b>
46	15	9.2	76	15	36	<b>LEBIH</b>
47	44	12	94.6	23	50	<b>LEBIH</b>
48	15	9.2	76	15	36	<b>LEBIH</b>
49	24	13	91.2	16	34	<b>BURUK</b>
50	18	15.1	82	18	35	<b>BAIK</b>
51	24	13	82	19	34	<b>LEBIH</b>

Tabel Data gizi anak Perempuan

No.	Umur	Berat Badan	Tinggi Badan	Lingkar Lengan Atas	Lingkar Kepala	Klasifikasi st. Gizi menurut who
1	33	11.3	85	20	40	<b>LEBIH</b>
2	30	9.3	82	21	39	<b>BAIK</b>
3	21	8.2	74	16	34	<b>LEBIH</b>
4	34	12.5	94	24	45	<b>BAIK</b>
5	12	9.6	75.2	15	32	<b>BAIK</b>
6	12	7	67	15	33	<b>BAIK</b>
7	32	9.6	86	25	45	<b>BAIK</b>
8	16	11.3	79.7	16	45	<b>BAIK</b>
9	25	10.2	80	20	43	<b>BAIK</b>
10	27	9.6	82	20	43	<b>BAIK</b>
11	24	10	79	20	42	<b>BAIK</b>
12	26	10	82	17	44	<b>BAIK</b>
13	30	11.5	88	22	46	<b>BAIK</b>
14	18	9.8	81	17	40	<b>LEBIH</b>
15	17	9.4	78.5	19	39	<b>BAIK</b>
16	26	10.8	84	18	46	<b>BAIK</b>
17	30	10.5	83	25	48	<b>LEBIH</b>

18	19	10	83	22	40	<b>LEBIH</b>
19	16	10	75	15	36	<b>BAIK</b>
20	36	12	86.3	25	47	<b>KURANG</b>
21	20	8	83	20	42	<b>BAIK</b>
22	14	8	71	15	35	<b>LEBIH</b>
23	17	11	71	16	38	<b>KURANG</b>
24	12	8	65	15	32	<b>BAIK</b>
25	31	11	83	23	39	<b>LEBIH</b>
26	26	11	84	16	41	<b>BURUK</b>
27	34	13	82	25	45	<b>LEBIH</b>
28	26	11.3	82	22	38	<b>BAIK</b>
29	12	9	65.7	15	35	<b>BAIK</b>
30	12	7.5	62	19	35	<b>BAIK</b>
31	12	8.6	67.6	20	38	<b>BAIK</b>
32	29	12.5	89	24	43	<b>BAIK</b>
33	26	10	78	25	40	<b>BAIK</b>
34	12	8	65	17	32	<b>BAIK</b>
35	12	8.5	66.4	20	39	<b>BAIK</b>
36	19	9	74.5	20	35	<b>BAIK</b>
37	13	7.5	69	18	34	<b>BAIK</b>
38	17	9	72	18	39	<b>BAIK</b>
39	17	7.5	70	15	40	<b>LEBIH</b>
40	18	8	76	23	37	<b>BAIK</b>
41	21	8.6	71.7	20	46	<b>KURANG</b>
42	48	12.5	92.2	25	48	<b>BAIK</b>
43	22	10	74	25	38	<b>BAIK</b>
44	36	12.3	91	25	40	<b>BAIK</b>
45	26	10	82	20	40	<b>BAIK</b>
46	21	14	65	15	33	<b>BAIK</b>
47	28	12	82	20	40	<b>BAIK</b>
48	17	10.7	74	17	45	<b>BURUK</b>
49	28	12	82	20	40	<b>BAIK</b>

**G. Lampiran Normalisasi**

## Normalisasi Data Gizi Anak Laki-laki

Berat Badan	Tinggi Badan	Lingkar Lengan Atas	Lingkar Kepala	BBn	TBn	LLAn	LKn
12.6	92.5	25	40	0.020413	0.035594	0.032225	-0.00409
12.2	88	24	42	0.015882	0.022049	0.026616	0.003929
10	81	18	30	-0.00904	0.00098	-0.00704	-0.04417
11.4	87.8	23	47	0.251464	0.507095	0.237012	0.352787
11	82	22	40	0.002288	0.00399	0.015398	-0.00409
12.5	94.2	24	50	0.01928	0.040711	0.026616	0.035994
12	86	21	40	0.013616	0.016029	0.009788	-0.00409
12	88.5	21	40	0.013616	0.023554	0.009788	-0.00409
10.6	77.7	18	38	-0.00224	-0.00895	-0.00704	-0.0121
9.5	78	10	43	-0.0147	-0.00805	-0.05191	0.007938
13	86	16	40	0.024944	0.016029	-0.01826	-0.00409
11.8	73	18	35	0.01135	-0.0231	-0.00704	-0.02413
12.5	88	25	42	0.01928	0.022049	0.032225	0.003929
7.6	73	18	38	-0.03623	-0.0231	-0.00704	-0.0121
10.1	82	16	38	0.236738	0.489637	0.197748	0.316715
14	75	17	38	0.036273	-0.01708	-0.01265	-0.0121
11.6	72	15	40	0.009085	-0.02611	-0.02387	-0.00409
11.5	80.5	25	47	0.007952	-0.00053	0.032225	0.02397
11	78	20	38	0.002288	-0.00805	0.004179	-0.0121
9	71	20	43	-0.02037	-0.02912	0.004179	0.007938
9.4	75	15	40	-0.01584	-0.01708	-0.02387	-0.00409
12	80	17	45	0.013616	-0.00203	-0.01265	0.015954
10	81	19	47	-0.00904	0.00098	-0.00143	0.02397
10	88	19	44	-0.00904	0.022049	-0.00143	0.011946
12	92	23	50	0.013616	0.034089	0.021007	0.035994
12	76.5	20	45	0.013616	-0.01256	0.004179	0.015954
10	77	24	42	-0.00904	-0.01106	0.026616	0.003929
10	83	15	41	-0.00904	0.007	-0.02387	-7.9E-05
10	86	25	47	-0.00904	0.016029	0.032225	0.02397
7.8	71.7	15	40	-0.03396	-0.02701	-0.02387	-0.00409
11	82.5	22	39	0.002288	0.005495	0.015398	-0.00809
13	83	22	47	0.024944	0.007	0.015398	0.02397
8.1	82	19	39	-0.03056	0.00399	-0.00143	-0.00809

9	76	15	33	-0.02037	-0.01407	-0.02387	-0.03214
9.2	73	20	37	-0.0181	-0.0231	0.004179	-0.01611
8.7	75	15	35	-0.02377	-0.01708	-0.02387	-0.02413
13	90	24	49	0.024944	0.028069	0.026616	0.031986
11.6	74	19	41	0.009085	-0.02009	-0.00143	-7.9E-05
9.1	79	20	44	-0.01924	-0.00504	0.004179	0.011946
7.2	74	15	39	-0.04076	-0.02009	-0.02387	-0.00809
9.5	76	16	36	-0.0147	-0.01407	-0.01826	-0.02012
9.9	76.5	21	49	-0.01017	-0.01256	0.009788	0.031986
8.8	72	23	39	-0.02263	-0.02611	0.021007	-0.00809
12.5	85.2	21	50	0.01928	0.013621	0.009788	0.035994
9.5	70	16	40	-0.0147	-0.03213	-0.01826	-0.00409
9.2	76	15	36	-0.0181	-0.01407	-0.02387	-0.02012
12	94.6	23	50	0.013616	0.041915	0.021007	0.035994
9.2	76	15	36	-0.0181	-0.01407	-0.02387	-0.02012
13	91.2	16	34	0.024944	0.031681	-0.01826	-0.02813
15.1	82	18	35	0.048734	0.00399	-0.00704	-0.02413
13	82	19	34	0.024944	0.00399	-0.00143	-0.02813

## Normalisasi Data Gizi Anak Perempuan

Berat Badan	Tinggi Badan	Lingkar Lengan Atas	Lingkar Kepala	BBn	TBn	LLAn	LKn
11.3	85	20	40	0.01591268	0.018761	0.001847	0.000568
9.3	82	21	39	-0.00899855	0.010886	0.00788	-0.00407
8.2	74	16	34	-0.02269972	-0.01011	-0.02228	-0.02725
12.5	94	24	45	0.28053093	0.451116	0.263601	0.393506
9.6	75.2	15	32	-0.00526186	-0.00696	-0.02832	-0.03652
7	67	15	33	-0.03764646	-0.02849	-0.02832	-0.03189
9.6	86	25	45	-0.00526186	0.021386	0.032011	0.023748
11.3	79.7	16	45	0.01591268	0.004848	-0.02228	0.023748
10.2	80	20	43	0.00221151	0.005636	0.001847	0.014476
9.6	82	20	43	-0.00526186	0.010886	0.001847	0.014476
10	79	20	42	-0.00027962	0.003011	0.001847	0.00984
10	82	17	44	-0.00027962	0.010886	-0.01625	0.019112
11.5	88	22	46	0.0184038	0.026636	0.013913	0.028385
9.8	81	17	40	-0.00277074	0.008261	-0.01625	0.000568
9.4	78.5	19	39	0.24191852	0.410429	0.233436	0.365689

10.8	84	18	46	0.00968488	0.016136	-0.01022	0.028385
10.5	83	25	48	0.00594819	0.013511	0.032011	0.037657
10	83	22	40	-0.00027962	0.013511	0.013913	0.000568
10	75	15	36	-0.00027962	-0.00749	-0.02832	-0.01798
12	86.3	25	47	0.02463161	0.022173	0.032011	0.033021
8	83	20	42	-0.02519084	0.013511	0.001847	0.00984
8	71	15	35	-0.02519084	-0.01799	-0.02832	-0.02261
11	71	16	38	0.012176	-0.01799	-0.02228	-0.0087
8	65	15	32	-0.02519084	-0.03374	-0.02832	-0.03652
11	83	23	39	0.012176	0.013511	0.019946	-0.00407
11	84	16	41	0.012176	0.016136	-0.02228	0.005204
13	82	25	45	0.03708723	0.010886	0.032011	0.023748
11.3	82	22	38	0.01591268	0.010886	0.013913	-0.0087
9	65.7	15	35	-0.01273523	-0.0319	-0.02832	-0.02261
7.5	62	19	35	-0.03141865	-0.04161	-0.00419	-0.02261
8.6	67.6	20	38	-0.01771748	-0.02691	0.001847	-0.0087
12.5	89	24	43	0.03085942	0.029261	0.025978	0.014476
10	78	25	40	-0.00027962	0.000386	0.032011	0.000568
8	65	17	32	-0.02519084	-0.03374	-0.01625	-0.03652
8.5	66.4	20	39	-0.01896304	-0.03006	0.001847	-0.00407
9	74.5	20	35	-0.01273523	-0.0088	0.001847	-0.02261
7.5	69	18	34	-0.03141865	-0.02324	-0.01022	-0.02725
9	72	18	39	-0.01273523	-0.01536	-0.01022	-0.00407
7.5	70	15	40	-0.03141865	-0.02061	-0.02832	0.000568
8	76	23	37	-0.02519084	-0.00486	0.019946	-0.01334
8.6	71.7	20	46	-0.01771748	-0.01615	0.001847	0.028385
12.5	92.2	25	48	0.03085942	0.037661	0.032011	0.037657
10	74	25	38	-0.00027962	-0.01011	0.032011	-0.0087
12.3	91	25	40	0.0283683	0.034511	0.032011	0.000568
10	82	20	40	-0.00027962	0.010886	0.001847	0.000568
14	65	15	33	0.04954284	-0.03374	-0.02832	-0.03189
12	82	20	40	0.02463161	0.010886	0.001847	0.000568
10.7	74	17	45	0.00843931	-0.01011	-0.01625	0.023748
12	82	20	40	0.02463161	0.010886	0.001847	0.000568