



**IMPLEMENTASI *K-NEAREST NEIGHBOR* PADA SISTEM REKOMENDASI
PENENTUAN JURUSAN PADA SISWA SEKOLAH MENENGAH ATAS
(SMA)**

SKIRPSI

Oleh

Gede Bagoes Santoso

NIM 122410101010

PROGRAM STUDI SISTEM INFORMASI

UNIVERSITAS JEMBER

2017



**IMPLEMENTASI *K-NEAREST NEIGHBOR* PADA SISTEM REKOMENDASI
PENENTUAN JURUSAN PADA SISWA SEKOLAH MENENGAH ATAS
(SMA)**

SKIRPSI

Diajukan guna melengkapi tugas akhir dan memenuhi salah satu syarat
untuk menyelesaikan pendidikan di Program Studi Sistem Informasi Universitas
Jember dan mendapat gelar Sarjana Sistem Informasi

Oleh

Gede Bagoes Santoso

NIM 122410101010

PROGRAM STUDI SISTEM INFORMASI

UNIVERSITAS JEMBER

2017

PERSEMBAHAN

Skripsi ini saya persembahkan untuk :

1. Allah SWT yang senantiasa memberikan rahmat dan hidayah-Nya untuk mempermudah dan melancarkan dalam mengerjakan skripsi.
2. Abah Sariyanto Raharjo dan Ibu Suni.
3. Sahabat - sahabatku dengan dukungan beserta doanya.
4. Guru - guruku baik dari pendidikan formal maupun informal.
5. Almamater Program Studi Sistem Informasi Universitas Jember.

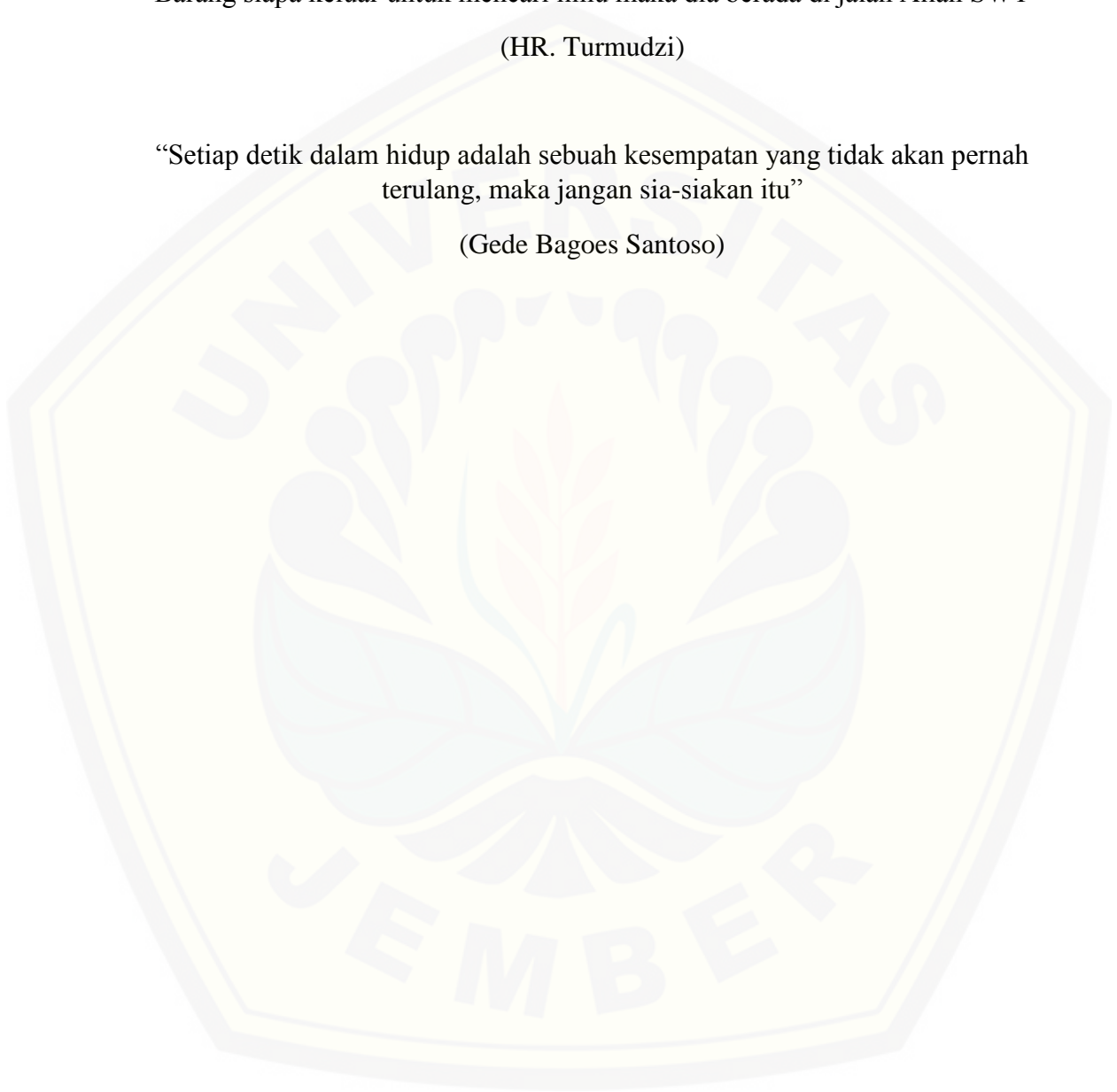
MOTTO

“Barang siapa keluar untuk mencari ilmu maka dia berada di jalan Allah SWT”

(HR. Turmudzi)

“Setiap detik dalam hidup adalah sebuah kesempatan yang tidak akan pernah terulang, maka jangan sia-siakan itu”

(Gede Bagoes Santoso)



PERNYATAAN

Saya yang bertanda tangan di bawah ini:

Nama : Gede Bagoes Santoso

NIM : 122410101010

menyatakan dengan sesungguhnya bahwa karya ilmiah yang berjudul “Implementasi *K-Nearest Neighbor* pada Sistem Rekomendasi Penentuan Jurusan pada Siswa Menengah Atas (SMA)” adalah benar-benar hasil karya sendiri, kecuali jika dalam pengutipan substansi disebutkan 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 sebenarnya, tanpa adanya tekanan dan paksaan dari pihak manapun serta bersedia mendapat sanksi akademik jika di kemudian hari pernyataan ini tidak benar.

Jember, 13 Juli 2017

Yang menyatakan,

Gede Bagoes Santoso

NIM 122410101010

PENGESAHAN PEMBIMBING

Skripsi berjudul “Implementasi *K-Nearest Neighbor* pada Sistem Rekomendasi Penentuan Jurusan pada Siswa Menengah Atas (SMA)”, telah diuji dan disahkan pada:

hari, tanggal : Jum’at, 21 Juli 2017

tempat : Program Studi Sistem Informasi Universitas Jember.

Disetujui oleh:

Pembimbing I,

Pembimbing II,

Anang Andrianto, ST., M.T
NIP. 196906151997021002

Yanuar Nurdiansyah, S.T., M.Cs
NIP. 196704201992011001

PENGESAHAN PENGUJI

Skripsi berjudul “Implementasi *K-Nearest Neighbor* pada Sistem Rekomendasi Penentuan Jurusan pada Siswa Menengah Atas (SMA)”, telah diuji dan disahkan pada:

hari, tanggal : Jum’at, 21 Juli 2017

tempat : Program Studi Sistem Informasi Universitas Jember

Penguji I,

Penguji II,

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

Fahrobby Adnan, S.Kom., M.MSI
NIP. 198706192014041001

Mengesahkan
Ketua Program Studi

Prof. Drs. Slamin, M.Comp.Sc.,Ph.D
NIP. 19670420 1992011001

RINGKASAN

Implementasi *K-Nearest Neighbor* pada Sistem Rekomendasi Penentuan Jurusan pada Siswa Menengah Atas (SMA); Gede Bagoes Santoso, 122410101010, 135 halaman ; Program Studi Sistem Informasi Universitas Jember.

Penentuan Jurusan merupakan salah satu upaya agar siswa dapat fokus dalam mempelajari dan mendalami bakat mereka masing – masing pada bidang akademik. Sistem Rekomendasi Penentuan Jurusan dengan pengimplementasian *K-Nearest Neighbor* ini nantinya akan membantu dalam proses Jurusan tersebut, sehingga dapat mempermudah dalam proses Jurusan. Ada beberapa kriteria yang dipakai dalam penelitian dan pembangunan sistem ini. Proses *K-Nearest Neighbor* dimulai dari penentuan kriteria, penentuan nilai K, normalisasi data uji dan data *training*, perhitungan *euclidean distance*, *sorting* dan penarikan kesimpulan. *K-Nearest Neighbor* bekerja dengan mencari jarak terpendek sampai terjauh antara data uji dengan data *training*. Metode ini mengambil data teratas sesuai K yang telah diinputkan dan menarik kesimpulan hasil perhitungangan dilihat dari jumlah data yang sama yg telah diambil sesuai dengan K. Penelitian ini menggunakan jenis penelitian pengembangan. Model pengembangan perangkat lunak untuk perancangan dan pembangunan dari sistem ini menggunakan model *waterfall*. Terdapat 5 tahapan penelitian yaitu : analisis kebutuhan, desain sistem, implementasi sistem, pengujian sistem, dan pemeliharaan sistem. Analisis kebutuhan merupakan tahap untuk mengumpulkan data dan informasi yang dibutuhkan dalam membangun sebuah perangkat lunak. Analisis kebutuhan terdiri dari studi pustaka, wawancara, dan gambaran umum sistem. Desain sistem menggunakan model *Unified Modeling Language* (UML) yang dirancang menggunakan konsep *Object Oriented Programming* (OOP). Implementasi sistem menggunakan Bahasa pemrograman *Page Hyper Text Pre-Processor* (PHP) dan manajemen basis data MySQL sistem dilakukan dengan menggunakan 2 cara yaitu *White Box* dan *Black Box Testing*.

Pemeliharaan sistem ini dilakukan dengan melakukan perbaikan pada *error* yang terdapat dalam perangkat lunak. Sistem ini dibangun berbasis website.



PRAKATA

Puji syukur kehadiran Allah SWT atas segala rahmat dan karunia-Nya, sehingga penulis dapat menyelesaikan skripsi dengan judul “Implementasi K-Nearest Neighbor pada Sistem Rekomendasi Penentuan Jurusan pada Siswa Menengah Atas (SMA)”. Skripsi ini disusun untuk memenuhi salah satu syarat menyelesaikan pendidikan Strata Satu (S1) pada Program Studi Sistem Informasi Universitas Jember.

Penyusunan skripsi ini tidak lepas dari bantuan berbagai pihak. Oleh karena itu, penulis menyampaikan terima kasih kepada:

1. Prof. Drs. Slamir, M.Comp.Sc., Ph.D., selaku Ketua Program Studi Sistem Informasi Universitas Jember sekaligus Dosen Pembimbing Akademik;
2. Anang Andrianto, ST., M.T selaku Dosen Pembimbing Utama dan Yanuar Nurdiansyah, S.T., M.Cs. selaku Dosen Pembimbing Pendamping yang telah meluangkan waktu, pikiran, dan perhatian dalam penulisan skripsi;
3. Seluruh Bapak dan Ibu dosen beserta staf karyawan di Program Studi Sistem Informasi Universitas Jember.
4. Abah Sariyanto Raharjo dan Ibu Suni yang selalu mendukung dan mendoakan.
5. Keluarga penulis yang selama ini memberikan nasehat dan perhatian.
6. Sri Wulan Nawang Sari yang selalu memberikan semangat dan memotivasi.
7. Anke Nawanda, Hanif Farid, Satria Aji, M. Eko, Afandi yang selalu memberikan dukungan.
8. Sahabat seperjuangan yang selalu menemani dan memberikan semangat serta doa Bagus Ananda, Bagus Cahyo, Tri Setyoko, Riska Arimanudin, Dimas Arifianto, Yohanis Permadi, Renaldi Dwi, Rizki Kurniawan, Brelyanez Fambudi

9. Teman - teman yang sudah membantu dan mendampingi penulis dalam penyusunan tugas akhir Hofi, Aglendy, Bryan, Aji, Ainul Yaqin, Hipolitus.
10. Teman - teman Crazy Squad yang selalu memberikan dukungan Byan, Candra Dwi, Firdaus, Duta, Rayen.
11. Teman - teman seperjuangan Formation angkatan 2012.
12. Teman-teman Program Studi Sistem Informasi di semua angkatan.
13. Teman-teman Unit Kegiatan Mahasiswa Olahraga MACO.
14. Ibu Hermin dan segenap keluarga besar SMAN 1 Sukodadi - Lamongan
15. Semua pihak yang tidak dapat disebutkan satu persatu.

DAFTAR ISI

SKIRPSI.....	ii
PERSEMBAHAN.....	iii
MOTTO	iv
PERNYATAAN.....	v
PENGESAHAN PEMBIMBING.....	vi
PENGESAHAN PENGUJI.....	vii
RINGKASAN	viii
PRAKATA.....	x
DAFTAR ISI.....	xii
DAFTAR GAMBAR	xvii
DAFTAR TABEL.....	xxi
BAB 1. PENDAHULUAN	1
1.1 Latar Belakang	1
3.2 Rumusan Masalah	2
3.3 Tujuan.....	2
3.4 Batasan Masalah.....	2
3.5 Sistematika Penulisan.....	3
BAB 2. TINJAUAN PUSTAKA	5
3.1 Penelitian Terdahulu.....	5
3.2 Datamining	6

3.3	Klasifikasi.....	6
3.4	K-Nearest Neighbor	7
BAB 3. METODOLOGI PENELITIAN.....		10
3.1	Jenis Penelitian	10
3.2	Tempat dan Waktu Penelitian	10
3.3	Tahapan Penelitian	10
3.3.1	Analisis Kebutuhan (<i>Requirements</i>).....	11
3.3.2	Desain Sistem	11
3.3.3	Implementasi.....	12
3.3.4	Pengujian	12
3.3.5	Pemeliharaan.....	13
BAB 4. PENGEMBANGAN SISTEM.....		14
3.4	4.1 Analisis Kebutuhan Sistem	14
4.1.1	<i>Statement of Purpose (SOP)</i>	14
4.1.2	Kebutuhan Fungsional	15
4.1.3	Kebutuhan Non-Fungsional	15
3.5	4.2 Desain Sistem.....	16
4.2.1	<i>Business Process</i>	16
4.2.3	Skenario Sistem.....	20
4.2.4	<i>Sequence Diagram</i>	22
4.2.5	<i>Activity Diagram</i>	25
4.2.6	<i>Class Diagram</i>	27
4.2.7	<i>Entity Relationship Diagram</i>	28

3.6	4.3 Implementasi	28
3.7	4.4 Pengujian Sistem	40
4.4.1	Pengujian <i>White Box</i>	40
4.4.2	Pengujian <i>Black Box</i>	43
BAB 5. HASIL DAN PEMBAHASAN.....		49
5.1	Hasil Pembuatan Sistem Klasifikasi Penentuan Jurusan pada Siswa Sekolah Menengah Atas (SMA)	49
5.1.1.	Tampilan Halaman Login	49
5.1.2.	Tampilan Halaman Utama Admin	50
5.1.3.	Tampilan Data Siswa dan Nilai Siswa	51
5.1.4.	Tampilan Tabel	54
5.1.5.	Tampilan perhitungan	57
5.1.6.	Tampilan Hasil Perhitungan.....	62
5.2	Hasil Penerapan Perhitungan K-Nearest Neighbor	63
5.2.1	Penentuan Data Kriteria	63
5.2.2	Penentuan Nilai K	63
5.2.3	Normalisasi <i>Z-Score</i>	64
5.2.4	Perhitungan <i>Euclidean Distance</i>	68
5.2.5	Sorting Data	68
5.2.6	Tarik Kesimpulan.....	87
5.2.7	Implementasi Metode <i>K-Nearest Neighbor</i> dalam Sistem Rekomendasi Penentuan Jurusan pada Siswa Menengah Atas (SMA).....	88
5.4	Pembahasan Penerapan Desain Sistem	90

5.4.1	Pembahasan Penerapan Desain Halaman Login	91
5.4.2	Pembahasan Penerapan Desain Halaman Utama Admin.....	91
5.4.3	Pembahasan Penerapan Desain Halaman Data Siswa dan Nilai Siswa	91
5.4.4	Pembahasan Penerapan Desain Halaman Tabel	92
5.4.5	Pembahasan Penerapan Desain Halaman Perhitungan	92
5.4.6	Pembahasan Penerapan Desain Halaman Hasil Perhitungan.....	93
5.5	Pembahasan Implementasi Penerapan Metode <i>K-Nearest Neighbor</i> dan Pengembangan Sistem.....	93
5.5.1	Pembahasan Penerapan Metode <i>K-Nearest Neighbor</i> dalam Rekomendasi Penentuan Jurusan pada Siswa Menengah Atas (SMA)	94
5.5.2	Pembahasan <i>Software Development Life Cycle Model Waterfall</i>	94
5.6	Pengujian Sistem Implementasi Metode <i>K-Nearest Neighbor</i> Pada Rekomendasi Penentuan Jurusan Pada Siswa Menengah Atas (SMA).....	95
BAB 6. PENUTUP		98
6.1	Kesimpulan.....	98
6.2	Saran	98
DAFTAR PUSTAKA		100
LAMPIRAN.....		101
LAMPIRAN A		101
A.1	Skenario Login.....	101
A.2	Skenario Menambah Data Siswa	102
A.3	Skenario Menambah Nilai (dataset).....	104
A.4	Skenario Verifikasi	108
A.5	Skenario Melihat Tabel Data Training	109

A.6 Skenario Melihat Tabel Dataset.....	110
A.7 Skenario Mengelola Perhitungan.....	110
A.8 Skenario Melihat Data Hasil Perhitungan.....	115
LAMPIRAN B	117
B.1 <i>Sequence Diagram Login</i>	117
B.2 <i>Sequence Diagram Menambah Data Siswa</i>	118
B.3 <i>Sequence Diagram Menambah Nilai</i>	120
B.4 <i>Sequence Diagram Verifikasi</i>	122
B.5 <i>Sequence Diagram Melihat Data Training</i>	123
B.6 <i>Sequence Diagram Melihat Dataset</i>	123
B.7 <i>Sequence Diagram Mengelola Perhitungan</i>	124
B.8 <i>Sequence Diagram Melihat Hasil Perhitungan</i>	125
LAMPIRAN C	126
C.1 <i>Activity Diagram Login</i>	126
C.2 <i>Activity Diagram Menambah Data Siswa</i>	128
C.3 <i>Activity Diagram Menambah Nilai Siswa</i>	130
C.4 <i>Activity Diagram Verifikasi</i>	132
C.5 <i>Activity Diagram Melihat Data Training</i>	133
C.5 <i>Activity Diagram Melihat Dataset</i>	133
C.7 <i>Activity Diagram Melihat Hasil Perhitungan</i>	135

DAFTAR GAMBAR

Gambar 2.1 Blok Diagram Klasifikasi (Bustami, 2014)..... 7

Gambar 3.1 Tahapan Metode Waterfall (Sommerville, 2011)..... 11

Gambar 3.2 Flowchart penerapan KNN pada sistem..... 11

Gambar 4.1 Bussiness Process 17

Gambar 4. 2 Use Case Diagram 18

Gambar 4.3 *Sequence Diagram* Melihat Data Hasil Perhitungan..... 24

Gambar 4.4 *Actifity Diagram* Melihat Data Hasil Perhitungan 26

Gambar 4.5 Class Diagram 27

Gambar 4.6 *Entitiy Relationship Diagram* 28

Gambar 4.8 Kode Program cperhitungan..... 29

Gambar 4.9 Kode Program mdataset 37

Gambar 4.10 Kode Program mdatatraining 39

Gambar 4.11 *Listing* Program method knn 41

Gambar 4.12 Diagram Alir method knn()..... 41

Gambar 5.1 Halaman *Login* Sistem 50

Gambar 5.2 Halaman utama admin..... 51

Gambar 5.3 Tampilan input data siswa..... 52

Gambar 5.4 Alert ketika form kosong pada tampilan input data siswa 52

Gambar 5. 5 Tampilan input nilai siswa (1)..... 53

Gambar 5.6 Tampilan input nilai siswa (2)..... 53

Gambar 5.7 *Alert* ketika form kosong pada tampilan input nilai siswa 54

Gambar 5.8 Tampilan halaman verifikasi (1)	55
Gambar 5.9 Tampilan halaman verifikasi (2)	55
Gambar 5.10 Tampilan halaman edit (1)	56
Gambar 5.11 Tampilan halaman edit (2)	56
Gambar 5.12 <i>Alert</i> ketika form kosong pada halaman edit.....	56
Gambar 5.13 Tampilan halaman dataset.....	57
Gambar 5.14 Tampilan halaman data training.....	57
Gambar 5.15 Tampilan halaman hitung data training (1).....	58
Gambar 5.16 Tampilan halaman hitung data training (2).....	58
Gambar 5.17 Tampilan halaman kuadrat (1)	59
Gambar 5.18 Tampilan kuadrat (2).....	59
Gambar 5.19 Tampilan halaman normalisasi (1).....	60
Gambar 5.20 Tampilan halaman normalisasi (2).....	60
Gambar 5.21 Tampilan halaman knn (1)	61
Gambar 5.22 Tampilan halaman knn (2)	61
Gambar 5.23 Tampilan halaman knn (3)	62
Gambar 5.24 Tampilan halaman hasil perhitungan	62
Gambar 5.25 Input nilai siswa (1).....	64
Gambar 5.26 Input nilai siswa (2).....	65
Gambar 5.27 Rata-rata setiap mata pelajaran	65
Gambar 5.28 Hasil perhitungan data training (1).....	66
Gambar 5.29 Hasil perhitungan data training (2).....	66
Gambar 5. 30 Kode Program Metode K-Nearest Neighbor (1).....	88

Gambar 5. 31 Kode Program Metode K-Nearest Neighbor (2).....	89
Gambar 5. 32 Kode Program Metode K-Nearest Neighbor (3).....	89
Gambar 5. 33 Kode Program Metode K-Nearest Neighbor (4).....	90
Gambar 5. 34 Kode Program Metode K-Nearest Neighbor (5).....	90
Gambar 5. 35 Grafik hasil perhitungan tingkat akurasi	97
Gambar B.1 <i>Sequence Diagram</i> Login Admin.....	117
Gambar B.2 <i>Sequence Diagram</i> Login Siswa.....	117
Gambar B.3 <i>Sequence Diagram</i> Menambah Data Siswa (Admin).....	118
Gambar B.4 <i>Sequence Diagram</i> Menambah Data Siswa (Siswa)	119
Gambar B.5 <i>Sequence Diagram</i> Menambah Nilai (Admin).....	120
Gambar B.6 <i>Sequence Diagram</i> Menambah Nilai (Siswa)	121
Gambar B.7 <i>Sequence Diagram</i> Verifikasi (admin).....	122
Gambar B.8 <i>Sequence Diagram</i> Melihat Data <i>Training</i> (admin).....	123
Gambar B.9 <i>Sequence Diagram</i> Melihat Data <i>Training</i> (admin).....	123
Gambar B.10 <i>Sequence Diagram</i> Mengelola Perhitungan (admin).....	124
Gambar B.11 <i>Sequence Diagram</i> Melihat Hasil Perhitungan (admin).....	125
Gambar B.12 <i>Sequence Diagram</i> Melihat Hasil Perhitungan (Siswa)	125
Gambar C.1 <i>Activity Diagram</i> Login (Admin)	126
Gambar C.2 <i>Activity Diagram</i> Login (Siswa).....	127
Gambar C.3 <i>Activity Diagram</i> Logout (Admin)	127
Gambar C.4 <i>Activity Diagram</i> Logout (Siswa).....	128
Gambar C.5 <i>Activity Diagram</i> Menambah Data Siswa (Admin).....	128
Gambar C.6 <i>Activity Diagram</i> Menambah Data Siswa (Siswa)	129

Gambar C.7 <i>Activity Diagram</i> Menambah Nilai Siswa (Admin)	130
Gambar C.8 <i>Activity Diagram</i> Menambah Nilai Siswa (Siswa).....	131
Gambar C.9 <i>Activity Diagram</i> Verifikasi (Admin).....	132
Gambar C.10 <i>Activity Diagram</i> Melihat Data <i>Training</i> (Admin).....	133
Gambar C.11 <i>Activity Diagram</i> Melihat <i>Dataset</i> (Admin).....	133
Gambar C.12 <i>Activity Diagram</i> Mengelola Perhitungan (Admin)	134
Gambar C.13 <i>Activity Diagram</i> Melihat Hasil Perhitungan (Admin).....	135
Gambar C.14 <i>Activity Diagram</i> Melihat Hasil Perhitungan (Admin).....	135

DAFTAR TABEL

Tabel 4.1 Definisi Aktor.....	19
Tabel 4.2 Definisi <i>Use Case</i>	19
Tabel 4.3 Melihat Data Hasil Perhitungan	22
Tabel 4.4 Test Case Function knn()	42
Tabel 4. 5 Pengujian <i>Black Box</i>	44
Tabel 5.1 Data Kriteria.....	63
Tabel 5. 2 Tabeli Hasil Hitung Nilai Euclidean Distance Data Uji dengan Data Training.....	79
Tabel 5. 3 Tabel Hasil Sorting Sesuai Nilai Euclidean Distance	68
Tabel 5. 4 Hasil Kesimpulan Sesuai $K=7$	87
Tabel 5. 5 Hasil Pengujian Data.....	95
Tabel A.1 Skenario Login Admin (Guru)	101
Tabel A.2 Skenario Login User (Siswa).....	101
Tabel A.3 Skenario Menambah Data Siswa.....	102
Tabel A.4 Skenario Menambah Data Siswa.....	103
Tabel A.5 Skenario Menambah Nilai (Dataset)	104
Tabel A.6 Skenario Menambah Nilai (Dataset)	106
Tabel A.7 Skenario Menambah Nilai (Dataset)	108
Tabel A.8 Skenario Melihat Tabel Data Training	109
Tabel A.9 Skenario Melihat Tabel Dataset	110
Tabel A.10 Skenario Mengelola Perhitungan	110
Tabel A.11 Melihat Data Hasil Perhitungan	115

Tabel A.12 Skenario Melihat Data Hasil Perhitungan 116



BAB 1. PENDAHULUAN

Bab ini merupakan langkah awal dari penulisan tugas akhir. Bab ini berisi latar belakang, rumusan masalah, tujuan dan manfaat, batasan masalah, metodologi penelitian, dan sistematika penulisan.

1.1 Latar Belakang

Penjurusan pada Sekolah Menengah Atas (SMA) dilakukan sebagai upaya untuk mengarahkan siswa pada minat dan bakat akademik siswa. Proses penjurusan diharapkan nantinya dapat membantu siswa mempelajari dan mengembangkan minat dan bakat siswa dalam bidang tersebut. Penjurusan ini juga berguna dalam menentukan pandangan dalam memilih jurusan saat masuk pada jenjang perguruan tinggi dan juga untuk jenjang karir nantinya.

Berpedoman dengan kurikulum yang dipakai sekarang yaitu kurikulum K-13, proses Jurusan pada tingkat Sekolah Menengah Atas (SMA) dilakukan sejak dini, yaitu sejak awal masuk pada kelas X. Setelah proses tahap seleksi masuk ke sekolah, siswa langsung melakukan tes psikologi sebagai rangkaian proses Jurusan. Selain itu ada juga proses perhitungan dengan melihat nilai raport dari Sekolah Menengah Pertama (SMP), nilai ujian nasional dan hasil angket setiap siswa. Pada proses perhitungan sebagai operator atau petugasnya adalah guru Bimbingan Konseling (BK). Guru BK bertugas untuk melihat dan mengitung serta mempertimbangkan nilai raport dan nilai ujian nasional masing-masing siswa sebagai salah satu proses Jurusan. Dikarenakan data siswa yang begitu banyak maka dirasakan proses tersebut terbilang cukup lama pengerjaannya karena masih dilakukan dengan bantuan *Microsoft Exel*. Proses perhitungan tersebut masih terpecah-pecah antara *sheet* satu dengan *sheet* yang lainnya dalam *Microsoft Exel*, tidak dalam satu proses sekaligus.

Berdasarkan uraian di atas dibutuhkan sistem yang dapat membantu dalam proses rekomendasi pemilihan jurusan pada Siswa Menengah Atas (SMA) agar prosesnya lebih efektif. Dimana pada proses pemilihan jurusan tersebut dibutuhkan metode yang nantinya dapat mempermudah sistem dalam proses rekomendasi. *K-Nearest Neighbor* salah metode yang menggunakan data latih untuk mencari jarak terdekat sebanyak K yang telah ditentukan.

3.2 Rumusan Masalah

Berdasarkan uraian latar belakang di atas, maka rumusan masalah yang ditimbulkan adalah sebagai berikut:

1. Bagaimana implementasi metode *K-Nearest Neighbor* pada rekomendasi penentuan jurusan pada siswa sekolah menengah atas (SMA)?
2. Bagaimana hasil pengujian akurasi dari Sistem rekomendasi penentuan jurusan pada siswa sekolah menengah atas (SMA)?

3.3 Tujuan

Berdasarkan rumusan serta batasan masalah yang telah dipaparkan sebelumnya maka tujuannya, yaitu sebagai berikut:

1. Mengimplementasikan metode *K-Nearest Neighbor* dalam proses rekomendasi penentuan jurusan pada siswa sekolah menengah atas (SMA)
2. Merancang dan membangun sebuah sistem rekomendasi penentuan jurusan pada siswa sekolah menengah atas (SMA) menggunakan metode *K-Nearest Neighbor*

3.4 Batasan Masalah

Beberapa hal yang membatasi penelitian ini diantaranya adalah:

1. Sistem ini menggunakan metode *K-Nearest Neighbor*
2. Objek yang digunakan dalam penelitian ini adalah SMA Negeri 1 Sukodadi

3. Sistem hanya digunakan untuk membantu proses rekomendasi penentuan jurusan pada siswa sekolah menengah atas (SMA)
4. Parameter yang digunakan untuk menunjang penelitian ini adalah pelajaran IPA dan pelajaran IPS
5. Sistem dapat membantu pengolahan data kriteria yang digunakan untuk proses rekomendasi penentuan jurusan pada siswa sekolah menengah atas (SMA)

3.5 Sistematika Penulisan

Adapun sistematika penulisan skripsi ini adalah sebagai berikut:

1. Pendahuluan

Bab ini memuat uraian tentang latar belakang, rumusan masalah, tujuan, manfaat, batasan masalah, dan sistematika penulisan skripsi yang masing-masing tertuang secara eksplisit dalam sub bab tersendiri.

2. Tinjauan Pustaka

Bab ini memaparkan tinjauan terhadap hasil-hasil penelitian terdahulu berkaitan dengan masalah yang dibahas, landasan materi dan konsep pemetaan strata desa siaga aktif, dan kajian teori metode analisis data yang berkaitan dengan masalah dalam penelitian.

3. Metodologi Penelitian

Bab ini menguraikan tentang tempat dan waktu penelitian, metode penelitian, metode pengumpulan data, metode analisis data, dan teknik pengembangan sistem yang digunakan dalam penelitian.

4. Pengembangan Sistem

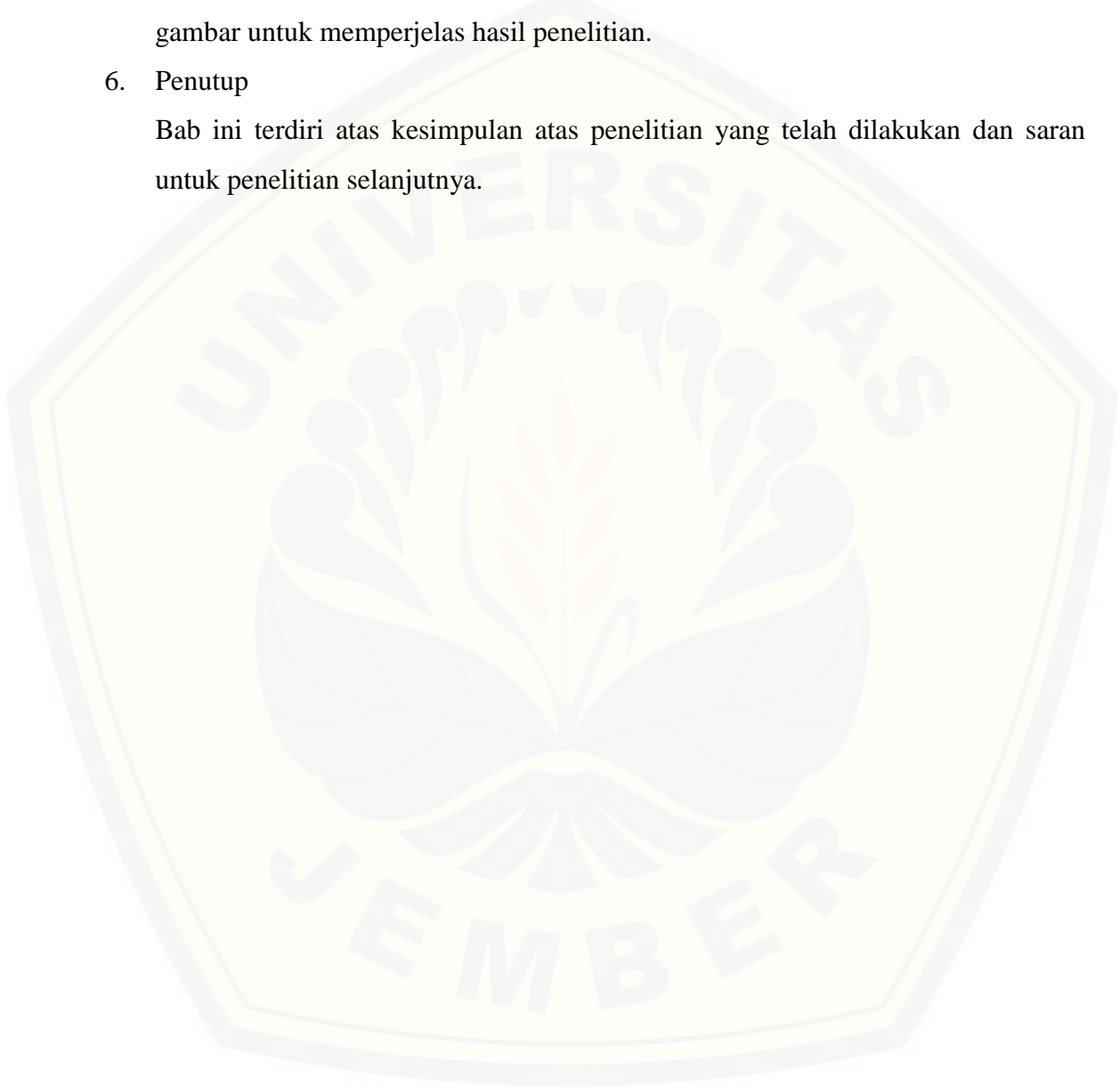
Bab ini akan menguraikan mengenai analisis kebutuhan, desain, implementasi, dan pengujian sistem yang digunakan dalam proses pengembangan sistem pengelompokan posyandu berdasarkan tingkat PHBS menggunakan metode *Naive Bayes Classifier*. Tahapan analisis hingga pengujian dilakukan sesuai dengan metode pengembangan *waterfall*.

5. Hasil dan Pembahasan

Bab ini memaparkan secara rinci pemecahan masalah melalui analisis yang disajikan dalam bentuk deskripsi dibantu dengan ilustrasi berupa tabel dan gambar untuk memperjelas hasil penelitian.

6. Penutup

Bab ini terdiri atas kesimpulan atas penelitian yang telah dilakukan dan saran untuk penelitian selanjutnya.



BAB 2. TINJAUAN PUSTAKA

Pada bagian ini dijelaskan teori-teori dan pustaka yang digunakan dalam penelitian. Teori yang dibahas adalah teori tentang data mining, klasifikasi, dan metode *K-Nearest Neighbor*.

3.1 Penelitian Terdahulu

Penelitian sebelumnya yang berjudul “Sistem Pendukung Keputusan Penempatan Jurusan Mahasiswa Baru Menggunakan Metode *K-Nearest Neighbor*, yang dilakukan oleh Ratih Kumalasari Niswatin, mahasiswa Teknik Informatika Universitas Nusantara PGRI Kediri, telah menghasilkan sistem pendukung keputusan yang digunakan untuk memberikan rekomendasi penempatan jurusan pada mahasiswa baru. Pada penelitian ini menunjukkan bahwa metode *K-Nearest Neighbor* digunakan untuk menyelesaikan permasalahan klasifikasi, dan metode ini cukup baik digunakan untuk menyelesaikan penelitian ini. Karena pada penelitian ini tersedia data training yang baik dan akurat, karena pada metode *K-Nearest Neighbor* hasil klasifikasi diperoleh dengan menghitung kedekatan antara permasalahan baru (*data testing*) dengan permasalahan lama (*data training*) berdasarkan pada kecocokan bobot nilai/fitur-fitur yang telah ditentukan.

Penelitian terdahulu juga yang berjudul “Sistem Pendukung Keputusan Status Gizi Balita menggunakan *K-Nearest Neighbor*, yang dilakukan oleh Yusa Dwi mahasiswi Program Studi Sistem Informasi Universitas Jember, telah menghasilkan sistem yang digunakan untuk mengklasifikasikan status gizi balita dengan beberapa

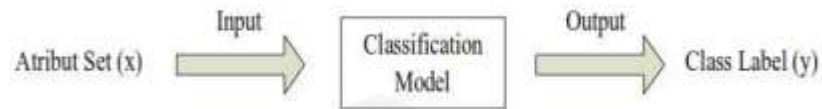
parameter yg digunakan dalam sistem tersebut. Pada sistem ini juga terdapat data *testing* dan data *training* yang digunakan dalam proses perhitungan pada metode *K-Nearest Neighbor*. Pada penelitian ini juga dicari kedekatan antara data lama dengan data baru melalui nilai k yang telah ditentukan.

3.2 Datamining

Basis data merupakan suatu ilmu yang mempelajari tentang data yang saling berhubungan, ada disiplin ilmu lain yang berhubungan dengan basis data, tetapi dengan tujuan menganalisis sebuah data yang ada untuk diolah menjadi sebuah informasi yaitu datamining. Turban mengatakan bahwa Datamining adalah proses yang menggunakan teknik statistik, matematika, kecerdasan buatan, dan *machine learning* untuk mengekstrasi dan mengidentifikasi informasi yang bermanfaat dan pengetahuan yang terkait (Kusrini, 2009). Tujuan dari ilmu datamining sendiri untuk menganalisa sebuah data dari berbagai sudut pandang, mengkategorikannya dan menyimpulkan relasi yang teridentifikasi.

3.3 Klasifikasi

Datamining merupakan ilmu yang digunakan untuk menganalisa sebuah data mengkategorikannya, mengelompokkan dan menyimpulkannya, dalam beberapa proses tersebut terdapat teknik mengelompokkan data pada datamining yang biasanya disebut klasifikasi, dimana Classification adalah proses untuk menemukan model atau fungsi yang menjelaskan atau membedakan konsep atau kelas data, dengan tujuan untuk dapat memperkirakan kelas dari suatu objek yang labelnya tidak diketahui (Kamber, 2006).



Gambar 2.1 Blok Diagram Klasifikasi (Bustami, 2014)

Gambar 2.1 menunjukkan gambaran dari alur sebuah proses klasifikasi dimana atribut set merupakan data input yang digunakan, setelah atribut set telah diinputkan maka akan diproses oleh metode klasifikasi dan akan menghasilkan sebuah output yang berupa class.

3.4 K-Nearest Neighbor

Klasifikasi merupakan proses untuk menyatakan suatu objek ke dalam salah satu kategori *datamining*. Klasifikasi data terdiri dari 2 langkah proses. Pertama ada 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). Algoritma K-Nearest Neighbor merupakan sebuah algoritma yang sering digunakan untuk klasifikasi teks dan data. Penggunaan K-Nearest Neighbor mempunyai sifat self-learning dimana jika semakin banyak dokumen, maka makin banyak pula sumber yang dapat digunakan untuk dibandingkan. KNearest Neighbor berarti mencari tetangga yang paling dekat dengan sets yang akan di klasifikasi (Miah, 2009).

K-Nearest Neighbor (K-NN) merupakan 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 atau yang paling mirip dengan objek pada data baru atau data *testing*. 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 ada pada persamaan 1 (Han J dan Kamber, 2000) :

$$d(\chi_i \chi_j) = \sqrt{\sum_r^n (\alpha_r(\chi_i) - \alpha_r(\chi_j))^2} \dots\dots\dots(1)$$

Keterangan:

$d(\chi_i \chi_j)$: Jarak <i>Euclidean Distance</i>
χ_i	: <i>record</i> ke- i
χ_j	: <i>record</i> ke- j
α_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):

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

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 persamaan 2 berikut :

$$V' = \frac{v - \bar{A}}{\sigma_{\bar{A}}} \dots\dots\dots(2)$$

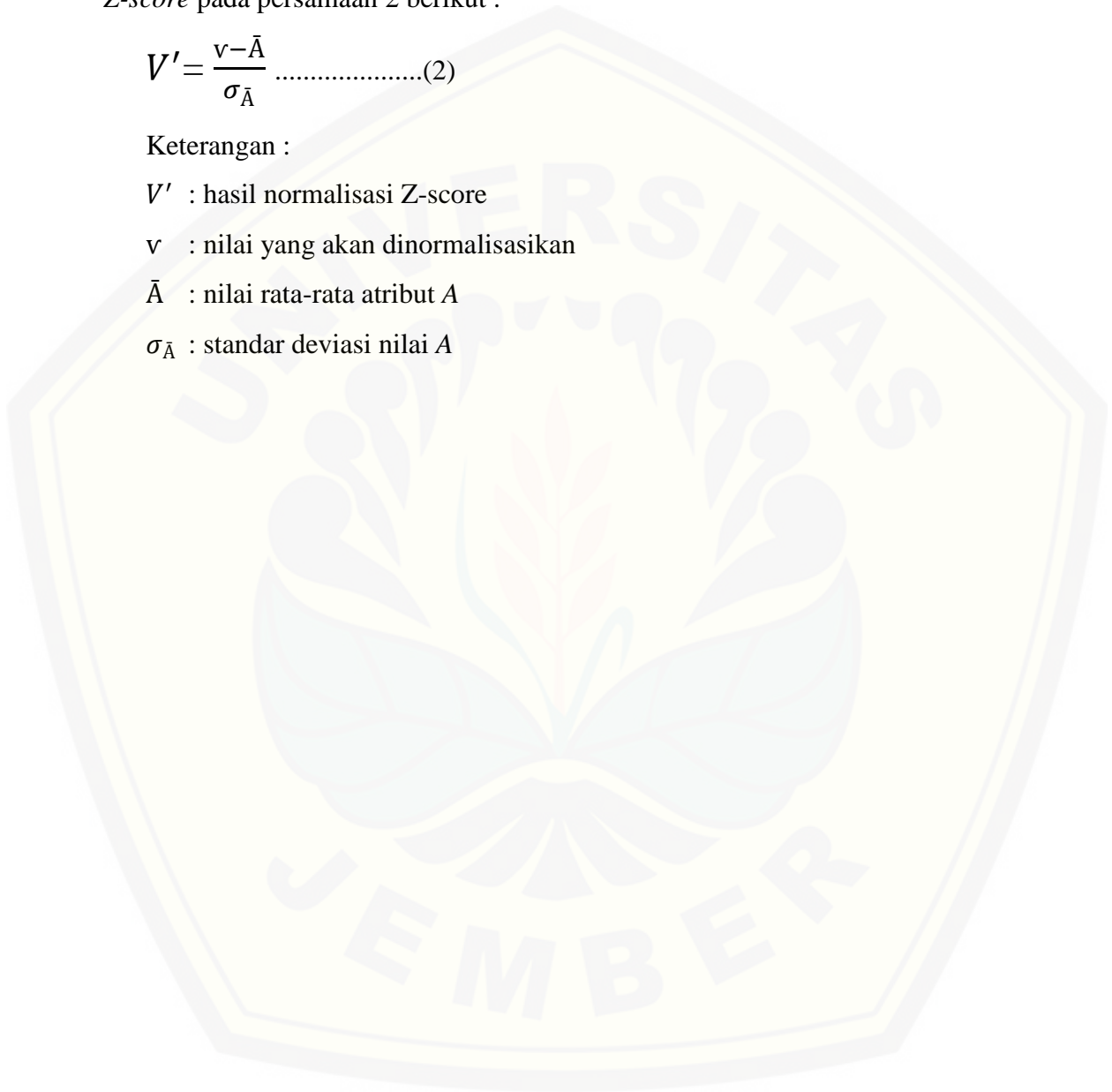
Keterangan :

V' : hasil normalisasi Z-score

v : nilai yang akan dinormalisasikan

\bar{A} : nilai rata-rata atribut A

$\sigma_{\bar{A}}$: standar deviasi nilai A



BAB 3. METODOLOGI PENELITIAN

Metode penelitian merupakan langkah dan prosedur yang akan dilakukan dalam mengumpulkan data atau informasi empiris guna memecahkan permasalahan, dan mengumpulkan informasi yang diperlukan untuk menyusun penelitian ini. Proses dimulai dari pengumpulan informasi sampai proses pembangunan sistem.

3.1 Jenis Penelitian

Pada penelitian ini digunakan dua jenis penelitian, yaitu penelitian kualitatif dan penelitian kuantitatif. Jenis penelitian kualitatif digunakan karena penelitian ini menganalisa studi literatur dan melakukan *interview* untuk pengumpulan sampel data dan jenis penelitian kuantitatif digunakan karena dalam penelitian ini menerapkan serta mengkaji teori yang sudah ada sebelumnya.

3.2 Tempat dan Waktu Penelitian

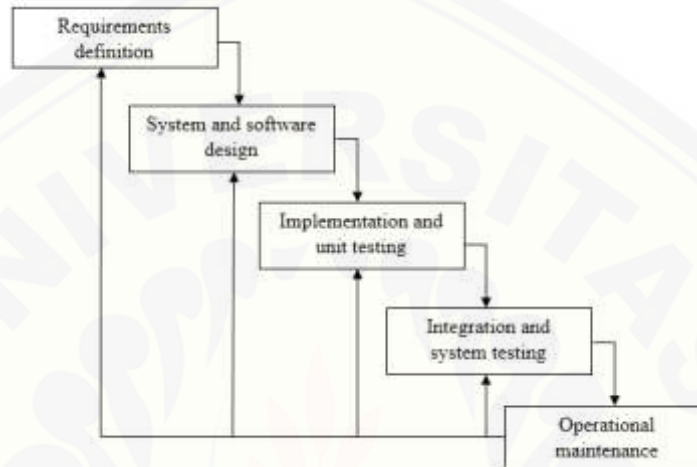
Tempat yang dilaksanakan untuk penelitian ini adalah di SMA Negeri 1 Sukodadi. Waktu penelitian dilakukan selama 3 bulan, dimulai bulan Agustus 2016 sampai dengan bulan Oktober 2016.

3.3 Tahapan Penelitian

Penelitian ini akan dilakukan dalam beberapa tahap yang disesuaikan dengan metode *Software Defelopment Life Cycle* (SDLC) waterfall, yang dibagi menjadi beberapa tahap yaitu : analisis kebutuhan, desain sistem, implementasi, dan pengujian.

Pembangunan perangkat lunak pada penelitian ini adalah metode waterfall. Waterfall model merupakan salah satu model proses perangkat lunak yang mengambil kegiatan proses dasar seperti spesifikasi, pengembangan, validasi dan

evolusi dengan mempresentasikannya sebagai fase-fase proses yang berbeda seperti analisis dan definisi persyaratan, perancangan perangkat lunak, implementasi dan pengujian unit, integrasi dan pengujian sistem, operasi dan pemeliharaan (Sommerville, 2011). Sebagaimana terlihat pada Gambar 2



Gambar 3.1 Tahapan Metode Waterfall (Sommerville, 2011)

3.3.1 Analisis Kebutuhan (*Requirements*)

Tahap pertama pada proses perancangan sistem adalah tahap analisis kebutuhan. Pada tahap ini dilakukan penelitian mencari permasalahan yang ada untuk dapat dianalisis kebutuhan yang diperlukan sebagai solusi dari permasalahan yang muncul. Data-data yang telah didapat kemudian dikelompokkan menjadi kebutuhan fungsional dan non-fungsional. Teknik pengumpulan data yang dilakukan antara lain:

1. Studi pustaka

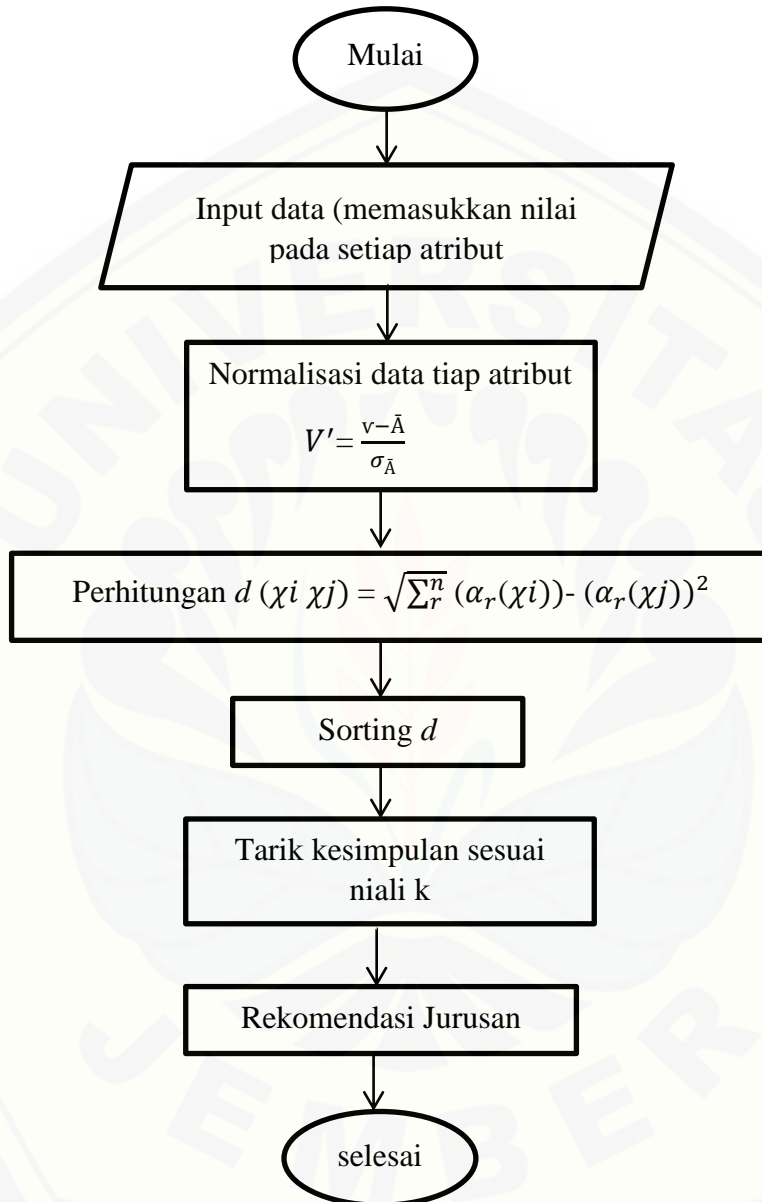
Teknik ini dilakukan dengan tujuan sebagai dasar pembaasan penyusunan dasar teori yang digunakan dalam penelitian. Sumber yang digunakan dalam studi pustaka berupa buku, jurnal, karya ilmiah, penelitian sebelumnya dan situs website.

2. Wawancara

Wawancara merupakan teknik mengumpulkan data dengan cara mengajukan pertanyaan-pertanyaan kepada narasumber untuk memperoleh data yang dibutuhkan guna menyelesaikan penelitian ini.

3. Tahap Analisi Data

Tahap ini dimulai dengan menelaah data secara keseluruhan, pada tahap ini juga dilakukan penentuan kriteria apa saja yang layak untuk dijadikan parameter dari data yang sudah dikumpulkan, dan menentukan kebutuhan sistem berdasarkan studi literatur dan wawancara yang telah dilakukan. Selanjutnya menganalisis data dengan metode *K-Nearest Neighbor*. Penerapan metode *K-Nearest Neighbor* dalam sistem ini dapat dilihat pada gambar 3.2



Gambar 3.2 Flowchart penerapan KNN pada sistem

3.3.2 Desain Sistem

Desain sistem merupakan tahap pembuatan desain sistem, dimana pada penelitian ini menggunakan *Unified Modeling Language* (UML) yang dirancang

dengan konsep *Object-Oriented Programming* (OOP). Pemodelan UML yang digunakan sebagai berikut:

1. *Business Process*
2. *Usecase Diagram* menggambarkan hubungan antara sistem dengan aktor berkaitan dengan fungsi atau tugas yang dilakukan oleh aktor
3. *Scenario* menjelaskan alur sistem dan keadaan yang akan terjadi ketika terjadi suatu event tertentu.
4. *Sequence Diagram* menampilkan pesan yang dikirim dan diterima antar object
5. *Activity Diagram* menjelaskan mengenai alur jalannya setiap *usecase*
6. *Class Diagram* menggambarkan relasi antar objek dan struktur semantik yang umum.
7. *Entity Relationship Diagram*

3.3.3 Implementasi

Tahap implementasi pada penelitian ini adalah dengan membangun sebuah website dari apa yang sudah ditulis sehingga terbentuk sebuah sistem. Beberapa hal yang dilakukan yaitu menulis kode (coding) sistem menggunakan bahasa pemrograman *HTML*, *CSS*, dan *PHP*. Manajemen basis data yang digunakan dalam membangun sistem yaitu aplikasi *DBMS MySQL*.

3.3.4 Pengujian

Tahap pengujian adalah tahap terakhir dari pembangunan sistem. Pengujian yang dilakukan pada sistem yaitu dengan proses *white box* dan *black box*. Pengujian *white box* adalah proses meneliti kode program dan menganalisa adanya kesalahan atau tidak sedangkan *black box* merupakan pengujian dengan melakukan *running* program untuk menguji kesalahan.

1. *White Box Testing*

Dilakukan oleh tim penguji dari pengembang sistem, dimana tidak hanya memperhatikan masukan/keluaran (I/O) tetapi juga algoritma yang digunakan apakah sesuai dengan rancangan yang dibuat atau tidak. Teknik pengujian ini menggunakan pengujian jalur dasar (*basis path testing*) dimana kompleksitas dari perangkat lunak yang dibangun akan dihitung menggunakan *Cyclomatic Complexity*.

2. *Black Box Testing*

Black-box testing merupakan pengujian program yang melihat dari segi fungsional tanpa melihat desain dan kode program. Tujuan dari black-box testing adalah untuk mengetahui apakah input, output, dan fitur-fitur pada program yang telah dibangun sesuai dengan kebutuhan user. Black-box testing dilakukan pihak dari tempat penelitian.

3.3.5 Pemeliharaan

Pemeliharaan merupakan proses perawatan sistem digunakan oleh pengguna. Pemeliharaan dilakukan dengan mengecek kinerja sistem secara berkala. Pengecekan dilakukan apakah kinerja sistem masih berjalan dengan baik dan memperbaiki jika terdapat kerusakan.

BAB 4. PENGEMBANGAN SISTEM

Bab ini menguraikan mengenai analisis kebutuhan, desain, implementasi, dan pengujian sistem yang digunakan dalam proses implementasi *K-Nearest Neighbor* pada sistem rekomendasi penentuan jurusan pada Siswa Menengah Atas (SMA). Tahapan analisis hingga pengujian dilakukan sesuai dengan metode pengembangan *waterfall*.

4.1 Analisis Kebutuhan Sistem

Berdasarkan metode pengembangan sistem model *waterfall*, tahapan awal yang dilakukan adalah tahapan analisis. Tahapan analisis ini dilakukan terhadap objek penelitian untuk memperoleh kebutuhan-kebutuhan dari sistem yang dibangun, baik berupa kebutuhan fungsional maupun kebutuhan non-fungsional. Dimana hasil analisis tersebut sangat mempengaruhi fungsionalitas sistem yang dibangun untuk dapat digunakan sesuai dengan fungsi dan kebutuhan pengguna.

4.1.1 *Statement of Purpose (SOP)*

Sistem rekomendasi penentuan Jurusan pada Siswa Menengah Atas (SMA) dengan mengimplementasikan metode *K-Nearest Neighbor* adalah sistem yang dapat mengolah data nilai kriteria yang digunakan dalam sistem ini untuk menghasilkan Jurusan pada siswa. Tujuan pengembangan sistem ini adalah guna mempermudah proses perhitungan kriteria dalam proses Jurusan yang sebelum menggunakan sistem ini dirasa memakan waktu yang cukup lama. Data yang dibutuhkan dalam sistem ini adalah data nilai raport Sekolah Menengah Pertama (SMP), nilai Ujian Nasional, angket minat dan saran dari hasil psikotes setiap siswa.

Sistem ini memiliki dua hak akses yaitu admin dan *user*, dimana yang bertugas sebagai admin adalah guru Bimbingan Konseling (BK) dan *user* adalah siswa. Sistem

ini dapat menginputkan data siswa, data nilai kriteria, melihat tabel data *training*, tabel *dataset*, mengelola perhitungan dan melihat hasil perhitungan. Sistem dibangun berbasis *web* dan menggunakan tampilan yang menarik atau *user friendly*.

4.1.2 Kebutuhan Fungsional

Kebutuhan fungsional sistem berisi fitur-fitur inti yang harus dipenuhi dalam sistem agar sistem mampu difungsikan sesuai dengan tujuan dan kebutuhan pengguna terhadap sistem itu sendiri. Kebutuhan fungsional dari sistem ini yaitu:

1. Sistem mampu melakukan login untuk admin dan *user* yang akan masuk sistem.
2. Sistem mampu menambahkan data siswa.
3. Sistem mampu menambahkan nilai kriteria.
4. Sistem mampu menampilkan data *training*.
5. Sistem mampu menampilkan *dataset*.
6. Sistem mampu mengelola perhitungan menggunakan metode *K-Nearest Neighbor*.
7. Sistem dapat menampilkan hasil dari kalsifikasi Jurusan menggunakan metode *K-Nearest Neighbor*.
8. Sistem dapat menampilkan daftar hasil kalsifikasi Jurusan menggunakan metode *K-Nearest Neighbor*.

4.1.3 Kebutuhan Non-Fungsional

Kebutuhan non-fungsional merupakan fitur-fitur yang dimiliki untuk mendukung sistem dalam memenuhi fungsionalitasnya untuk dapat memenuhi kebutuhan dari pengguna. Kebutuhan non-fungsional dari sistem ini yaitu:

1. Sistem memiliki batasan hak akses pengguna dengan menggunakan *username* dan *password*.

2. Sistem berbasis *website*.
3. Sistem memiliki tampilan yang *user friendly*.
4. Sistem menggunakan *framework Code Igniter*.

4.2 Desain Sistem

Tahapan yang dilakukan setelah melakukan analisis kebutuhan sistem yaitu tahap perencanaan pembangunan sistem yang dapat digambarkan dengan desain sistem. Desain pada sistem ini meliputi *use case diagram*, *use case skenario*, *activity diagram*, *sequence diagram*, *class diagram*, dan *entity relationship diagram*.

4.2.1 Business Process

Business process merupakan diagram yang menggambarkan kebutuhan data yang dibutuhkan oleh sistem. *Business process* didalamnya terdiri atas beberapa poin diantaranya:

1. *Input* : Data yang dimasukkan ke dalam sistem
2. *Output* : Data yang dihasilkan oleh sistem
3. *Goal* : Tujuan dibangun suatu sistem
4. *Used* : *Platform* yang menjadi basis sistem
5. *Process* : Sistem yang bekerja

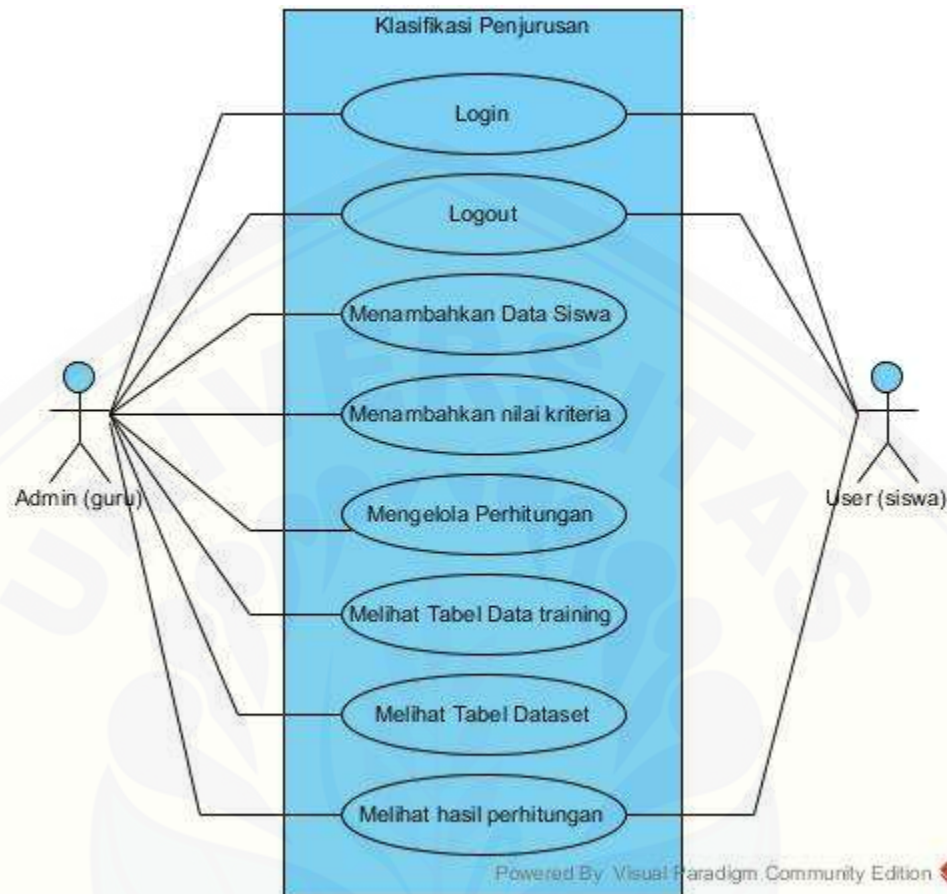


Gambar 4.1 Bussiness Process

Gambar 4.1 merupakan *business process* dari sistem rekomendasi penentuan Jurusan pada siswa Sekolah Menengah Atas (SMA). *Business process* menjelaskan proses *input*, *output*, *goal*, dan *uses* yang diaplikasikan ke dalam sistem.

4.2.2 Use case Diagram

Use case diagram digunakan untuk dapat menggambarkan interaksi antara aktor dengan sistem. Melalui *use case diagram* dapat diketahui interaksi yang dapat dilakukan aktor terhadap sistem sesuai dengan hak akses yang dimiliki oleh masing-masing aktor atau pengguna. *Use case diagram* ditunjukkan pada Gambar 4.2



Gambar 4. 2 Use Case Diagram

Gambar 4.2 menunjukkan *use case diagram* sistem rekomendasi penentuan Jurusan pada siswa Sekolah Menengah Atas (SMA) yang terdiri dari 2 aktor.

Use case sistem rekomendasi penentuan Jurusan pada siswa Sekolah Menengah Atas (SMA) ini mempunyai penjelasan berupa tabel definisi aktor yang menggambarkan tugas-tugas aktor dalam mengoperasikan sistem tersebut dapat dilihat pada Tabel 4.1 Definisi Aktor. Penjelasan lainnya yaitu disebut definisi *use case* yang menggambarkan fungsionalitas dari setiap *usecase* dapat dilihat pada Tabel 4.2 Definisi *Use Case*.

Tabel 4.1 Definisi Aktor

No.	Aktor	Definisi Tugas
1.	<i>Admin (Guru)</i>	Menambahkan data siswa, menambahkan nilai kriteria, melihat tabel data <i>training</i> , melihat tabel <i>dataset</i> , mengelola perhitungan, melihat tabel hasil perhitungan
2.	<i>User (Siswa)</i>	Melihat tabel hasil perhitungan

Tabel 4.2 Definisi Use Case

No.	Use Case	Deskripsi
1.	<i>Login</i>	<i>Usecase</i> Login merupakan <i>usecase</i> yang digunakan untuk mengelola data login admin, user.
2.	<i>Logout</i>	<i>Usecase</i> Logout merupakan <i>usecase</i> yang digunakan untuk mengelola data <i>logout</i> admin, user.
3.	Menambahkan Data Siswa	<i>Usecase</i> menambahkan data siswa merupakan <i>usecase</i> yang digunakan untuk menambahkan data siswa baru yang akan dimasukkan ke dalam sistem
4.	Menambahkan Nilai Kriteria	<i>Usecase</i> menambahkan nilai kriteria merupakan <i>usecase</i> yang digunakan untuk menambahkan nilai kriteria yang ada pada sistem
5.	Mengelola Perhitungan	<i>Usecase</i> mengelola perhitungan merupakan <i>usecase</i> yang digunakan untuk mengelola perhitungan yang ada pada sistem.
6.	Melihat Tabel Data <i>Training</i>	<i>Usecase</i> melihat tabel data <i>training</i> merupakan <i>usecase</i> yang digunakan untuk melihat tabel

		data <i>training</i> yang ada dalam sistem.
7.	Melihat Tabel <i>Dataset</i>	<i>Usecase</i> melihat tabel <i>dataset</i> merupakan <i>usecase</i> yang digunakan untuk melihat tabel <i>dataset</i> yang ada dalam sistem.
8.	Melihat Hasil Perhitungan	<i>Usecase</i> melihat hasil perhitungan merupakan <i>usecase</i> yang digunakan untuk melihat tabel hasil perhitungan yang telah dilakukan oleh sistem.

4.2.3 Skenario Sistem

Skenario sistem berfungsi untuk menjelaskan alur dari sebuah sistem serta alur alternatif yang dilakukan oleh para aktor yang menggunakan sistem ini. Skenario sistem sesuai dengan yang ada pada *Use case diagram* seperti pada Gambar 4.2.

1. Skenario *Login*

Skenario *Login* merupakan alur dari aktor dan sistem jika akan masuk dalam sistem. Penjelasan urutan aksi aktor dan reaksi sistem pada skenario utama dan skenario alternatif skenario *login* ditunjukkan pada lampiran A.

2. Skenario *Logout*

Skenario *logout* merupakan alur dari sistem dan aktor jika akan keluar dari sistem. Penjelasan urutan aksi aktor dan reaksi sistem pada skenario utama dan skenario alternatif skenario *logout* ditunjukkan pada lampiran A.

3. Skenario Menambah Data Siswa

Skenario menambah data siswa merupakan alur dari aktor dan sistem jika akan menambahkan data siswa. Penjelasan urutan aksi aktor dan reaksi sistem pada skenario utama dan skenario alternatif skenario menambahkan data siswa ditunjukkan pada lampiran A.

4. Skenario Menambahkan Nilai Kriteria

Skenario menambahkan nilai kriteria merupakan alur dari aktor dan sistem jika akan menambahkan nilai kriteria. Penjelasan urutan aksi aktor dan reaksi sistem pada skenario utama dan skenario alternatif skenario menambahkan nilai kriteria ditunjukkan pada lampiran A.

5. Skenario Verifikasi

Skenario verifikasi merupakan alur dari aktor dan sistem jika akan melakukan proses verifikasi terhadap data uji. Penjelasan urutan aksi aktor dan reaksi sistem pada skenario utama dan skenario alternatif skenario menambahkan nilai kriteria ditunjukkan pada lampiran A.

6. Skenario Melihat Data *Training*

Skenario melihat data *training* merupakan alur dari aktor dan sistem jika akan melihat data *training*. Penjelasan urutan aksi aktor dan reaksi sistem pada skenario utama dan skenario alternatif skenario melihat data *training* ditunjukkan pada lampiran A.

7. Skenario Melihat *Dataset*

Skenario melihat *dataset* merupakan alur dari aktor dan sistem jika akan melihat *dataset*. Penjelasan urutan aksi aktor dan reaksi sistem pada skenario utama dan skenario alternatif skenario melihat *dataset* ditunjukkan pada lampiran A.

8. Skenario Mengelola Perhitungan

Skenario mengelola perhitungan merupakan alur dari aktor dan sistem jika akan mengelola perhitungan. Penjelasan urutan aksi aktor dan reaksi sistem pada skenario utama dan skenario alternatif skenario mengelola perhitungan ditunjukkan pada lampiran A.

9. Skenario Melihat Data Hasil Perhitungan

Skenario melihat data hasil perhitungan merupakan alur dari aktor dan sistem jika akan melihat data hasil perhitungan. Penjelasan urutan aksi aktor dan reaksi sistem pada skenario utama dan skenario alternatif skenario melihat data hasil perhitungan ditunjukkan pada tabel 4.3

Tabel 4.3 Melihat Data Hasil Perhitungan

Nama Use Case	Melihat Data Hasil Perhitungan
Aktor	User (siswa)
Deskripsi Singkat	Siswa akan melihat data hasil perhitungan
Prekondisi	Siswa memilih menu tabel hasil perhitungan
Prakondisi	Melihat data hasil perhitungan
Flow Events	
Skenario Normal : Melihat Data Hasil Perhitungan	
Aksi Aktor	Reaksi Sistem
1. Klik menu tabel hasil perhitungan	
	2. Menampilkan halaman hasil, meliputi tabel data hasil perhitungan (no, nama, Jurusan)

4.2.4 Sequence Diagram

Sequence Diagram pada sistem ini digunakan untuk menggambarkan skenario atau rangkaian langkah-langkah yang dilakukan sebagai sebuah respon dari suatu kejadian untuk menghasilkan *output* tertentu. *Sequence Diagram* diawali dari apa yang me-trigger aktivitas tersebut, proses dan perubahan apa saja yang terjadi secara internal dan *output* apa yang dihasilkan.

1. *Sequence Diagram Login*

Sequence diagram login merupakan langkah-langkah yang dilakukan sebagai sebuah respon dari suatu kejadian/event untuk melakukan proses *login*. Penggambaran *sequence diagram login* digunakan untuk menjelaskan fungsi atau *method* yang akan dibuat seperti yang ditunjukkan pada lampiran B.

2. *Sequence Diagram Logout*

Sequence diagram logout merupakan langkah-langkah yang dilakukan sebagai sebuah respon dari suatu kejadian/event untuk melakukan proses *logout*. Penggambaran *sequence diagram logout* digunakan untuk menjelaskan fungsi atau *method* yang akan dibuat seperti yang ditunjukkan pada lampiran B.

3. *Sequence Menambahkan Data Siswa*

Sequence menambahkan data siswa merupakan langkah-langkah yang dilakukan sebagai sebuah respon dari suatu kejadian/event untuk melakukan proses menambahkan data siswa. Penggambaran *sequence diagram* menambahkan data siswa digunakan untuk menjelaskan fungsi atau *method* yang akan dibuat seperti yang ditunjukkan pada lampiran B.

4. *Sequence* Menambahkan Nilai Siswa

Sequence menambahkan nilai kriteria merupakan langkah-langkah yang dilakukan sebagai sebuah respon dari suatu kejadian/event untuk melakukan proses menambahkan nilai kriteria. Penggambaran *sequence diagram* menambahkan nilai kriteria digunakan untuk menjelaskan fungsi atau *method* yang akan dibuat seperti yang ditunjukkan pada lampiran B.

5. *Sequence* Melakukan Verifikasi

Sequence menambahkan nilai kriteria merupakan langkah-langkah yang dilakukan sebagai sebuah respon dari suatu kejadian/event untuk melakukan proses verifikasi data uji. Penggambaran *sequence diagram* verifikasi untuk menjelaskan fungsi atau *method* yang akan dibuat seperti yang ditunjukkan pada lampiran B.

6. *Sequence* Melihat Tabel Data *Training*

Sequence melihat tabel data *training* merupakan langkah-langkah yang dilakukan sebagai sebuah respon dari suatu kejadian/event untuk melakukan proses melihat tabel data *training*. Penggambaran *sequence diagram* melihat tabel data *training* digunakan untuk menjelaskan fungsi atau *method* yang akan dibuat seperti yang ditunjukkan pada lampiran B.

7. *Sequence* Melihat Tabel *Dataset*

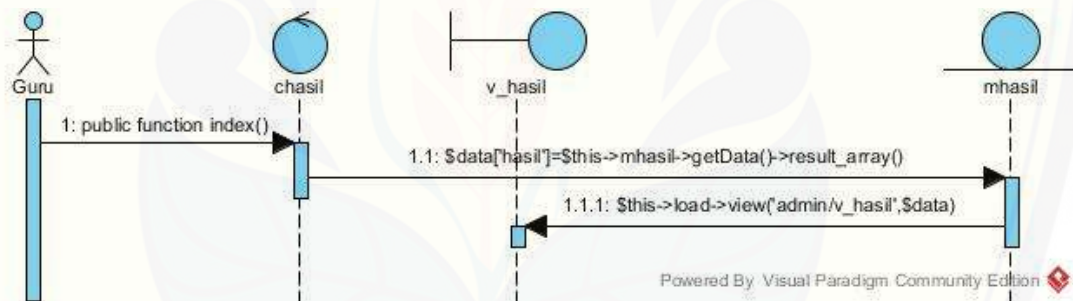
Sequence melihat tabel *dataset* merupakan langkah-langkah yang dilakukan sebagai sebuah respon dari suatu kejadian/event untuk melakukan proses melihat tabel *dataset*. Penggambaran *sequence diagram* melihat tabel *dataset* digunakan untuk menjelaskan fungsi atau *method* yang akan dibuat seperti yang ditunjukkan pada lampiran B.

8. *Sequence* Mengelola Perhitungan

Sequence mengelola perhitungan merupakan langkah-langkah yang dilakukan sebagai sebuah respon dari suatu kejadian/event untuk melakukan proses mengelola perhitungan. Penggambaran *sequence diagram* mengelola perhitungan digunakan untuk menjelaskan fungsi atau *method* yang akan dibuat seperti yang ditunjukkan pada lampiran B.

9. *Sequence* Melihat Data Hasil Perhitungan

Sequence melihat data hasil perhitungan merupakan langkah-langkah yang dilakukan sebagai sebuah respon dari suatu kejadian/event untuk melakukan proses melihat data hasil perhitungan. Penggambaran *sequence diagram* melihat data hasil perhitungan digunakan untuk menjelaskan fungsi atau *method* yang akan dibuat seperti yang ditunjukkan pada gambar 4.3,



Gambar 4.3 *Sequence Diagram* Melihat Data Hasil Perhitungan

Gambar 4.3 merupakan *sequence diagram* melihat data hasil perhitungan untuk hak akses admin dan user. *Sequence* ini menggambarkan alur method dalam proses melihat data hasil perhitungan. Pada *sequence* ini terdapat *class view* hasil, *class controller* chasil, dan *class model* mhasil serta di dalam class tersebut terdapat beberapa method yang dipanggil.

4.2.5 Activity Diagram

Activity diagram pada sistem ini berfungsi untuk menggambarkan berbagai alir aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, decision yang mungkin terjadi, dan bagaimana masing-masing alir berakhir.

1. Activity diagram login

Activity diagram ini menggambarkan aktivitas *login* jika ingin mengakses sistem informasi ini dengan *username* dan *password* yang telah disediakan serta login sesuai hak akses yang ada seperti yang ditunjukkan pada lampiran C.

2. Activity diagram Logout

Activity diagram ini menggambarkan aktivitas *logout* jika ingin keluar sistem seperti yang ditunjukkan pada lampiran C.

3. Activity diagram Menambahkan Data Siswa

Activity diagram ini menggambarkan alur aktivitas yang dilakukan oleh aktor dan sistem jika ingin mengakses fitur input data siswa seperti yang ditunjukkan pada lampiran C.

4. Activity diagram Menambahkan Nilai Siswa

Activity diagram ini menggambarkan alur aktivitas yang dilakukan oleh aktor dan sistem jika ingin mengakses fitur input nilai siswa dimulai dari memasukkan nilai dan edit nilai seperti yang ditunjukkan pada lampiran C

5. Activity diagram Verifikasi

Activity diagram ini menggambarkan alur aktivitas yang dilakukan oleh aktor dan sistem jika ingin melakukan proses verifikasi seperti yang ditunjukkan pada lampiran C

6. Activity diagram Melihat Tabel Data Training

Activity diagram ini menggambarkan alur aktivitas yang dilakukan oleh aktor dan sistem jika ingin melihat tabel data *training* seperti yang ditunjukkan pada lampiran C.

7. Activity diagram Melihat Tabel Dataset

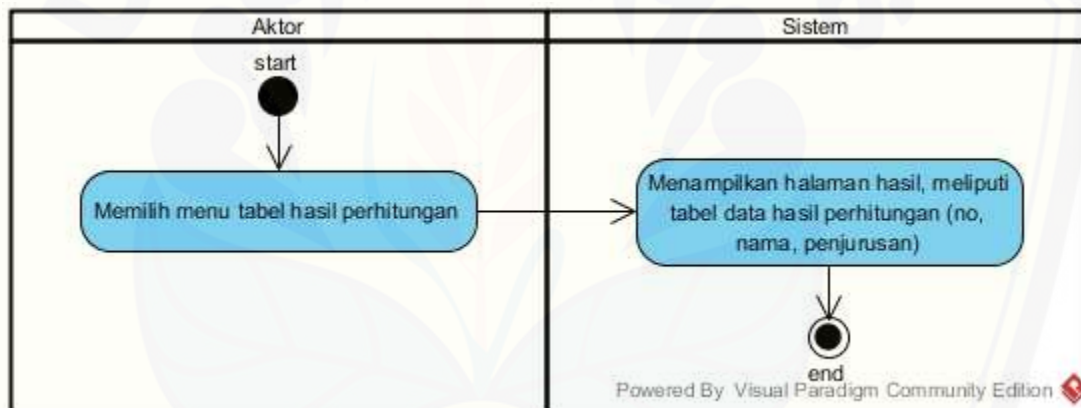
Activity diagram ini menggambarkan alur aktivitas yang dilakukan oleh aktor dan sistem jika ingin melihat tabel *dataset* seperti yang ditunjukkan pada lampiran C.

8. *Activity diagram* Mengelola Perhitungan

Activity diagram ini menggambarkan alur aktivitas yang dilakukan oleh aktor dan sistem jika ingin mengakses dan mengelola data perhitungan dimulai dari *view* hitung, kuadrat, normalisasi, knn, seperti yang ditunjukkan pada lampiran C.

9. *Activity diagram* Melihat Data Hasil Perhitungan

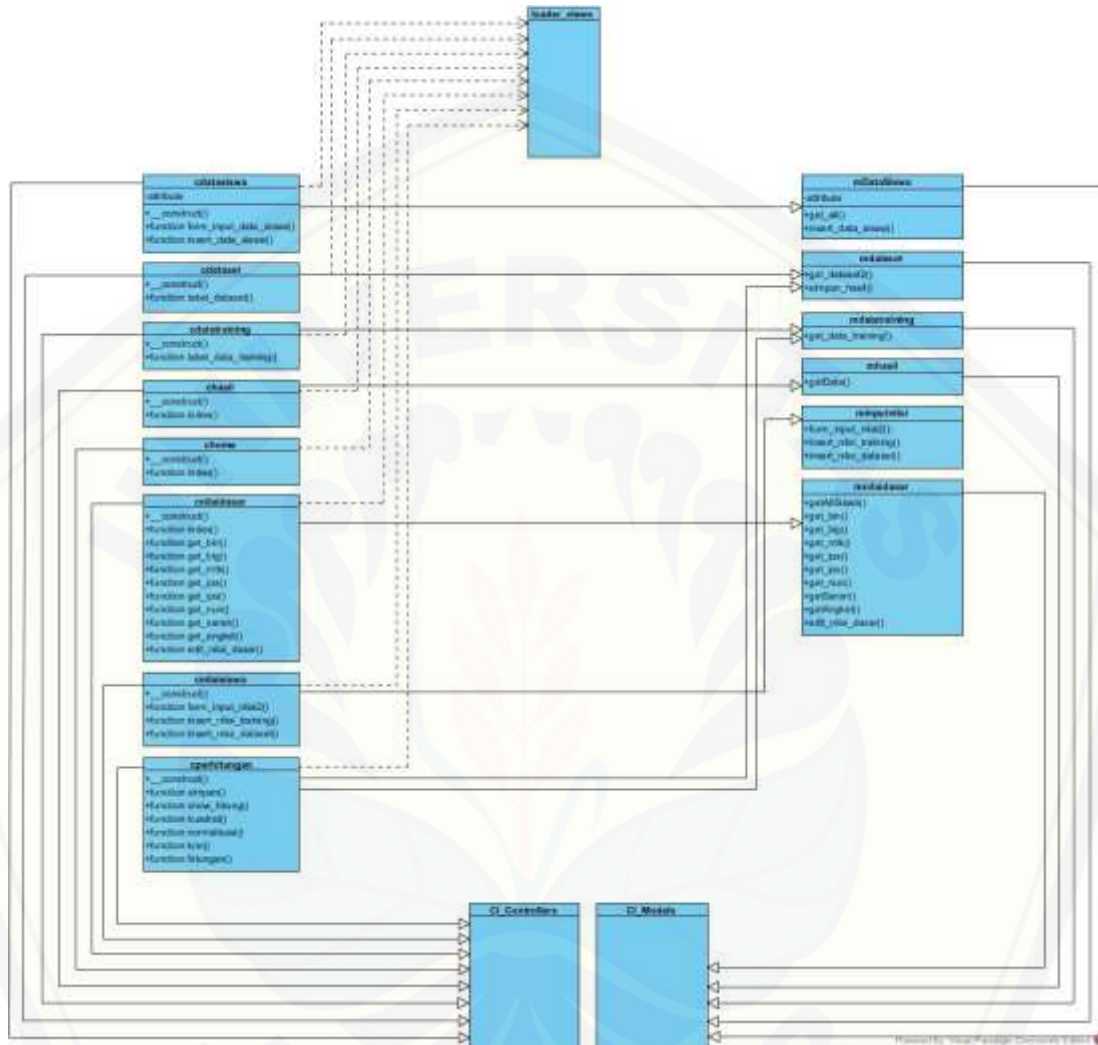
Activity diagram ini menggambarkan alur aktivitas yang dilakukan oleh aktor dan sistem jika ingin melihat data hasil perhitungan seperti yang ditunjukkan pada gambar 4.4,



Gambar 4.4 *Actifity Diagram* Melihat Data Hasil Perhitungan

Gambar 4.4 menggambarkan *activity diagram* data hasil perhitungan dimana pada *activity diagram* ini menggambarkan aktivitas admin dan user jika ingin melihat data hasil perhitungan

4.2.6 Class Diagram

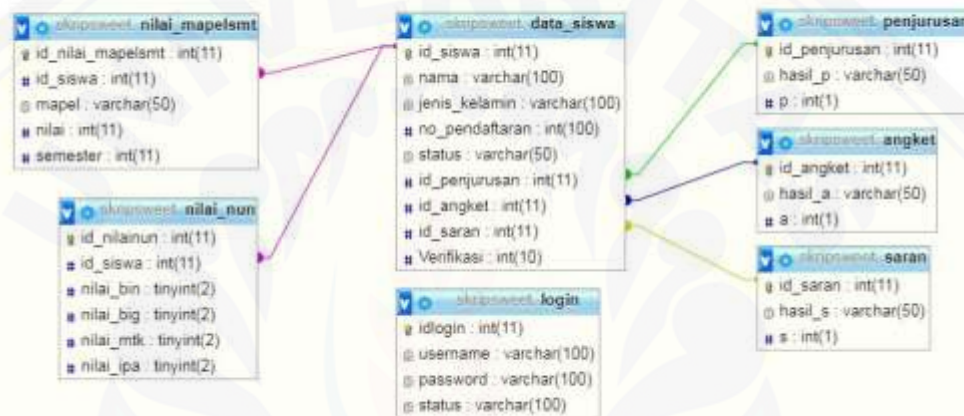


Gambar 4.5 Class Diagram

Gambar 4.5 menggambarkan tentang relasi antar *class* di dalam sistem. Relasi terjadi antar *controller*, *model*, dan *view*. Berdasarkan gambar tersebut bisa dipahami keterkaitan dan ketergantungan antar *class* di dalam sistem.

4.2.7 Entity Relationship Diagram

Entity Relationship Diagram (ERD) pada sistem rekomendasi penentuan jurusan pada siswa menengah atas (SMA) ini menjelaskan hubungan antar data dalam basis data berdasarkan objek-objek dasar data yang mempunyai hubungan antar relasi. ERD aplikasi ditunjukkan pada gambar 4.6.



Gambar 4.6 Entity Relationship Diagram

4.3 Implementasi

Setelah tahap perancangan selesai, tahap selanjutnya dalam penelitian ini yaitu tahap pengimplementasian desain perancangan ke dalam bahasa pemrograman. Bahasa pemrograman yang dipakai adalah bahasa pemrograman *PHP (Hypertext Preprocessor)* dan menggunakan database *MySQL*. Dalam perancangan sistem informasi pengelompokan posyandu berdasarkan tingkat PHBS ini menggunakan *framework Code Igniter* untuk memudahkan di dalam pengembangan dan penulisan *coding*. Pada tahap implementasi perancangan ini menjelaskan tentang fitur-fitur yang terdapat pada sistem. Salah satu fitur yang ada dalam sistem ini adalah perhitungan yang menggunakan metode *K-Nearest Neighbor*. Kode program perhitungan metode *K-Nearest Neighbor* terdapat di *class cperhitungan* pada *package controller* dan *class mdataset, mdatatraining* pada *package models*. Penulisan kode

program perhitungan metode *K-Nearest Neighbor* dapat dilihat pada gambar 4.8 sampai dengan gambar 4.10,

1. *Class Controller* cperhitungan

Pada *Controller* cperhitungan terdapat method `simpan()` yang berfungsi untuk menampilkan hasil perhitungan, method `show_hitung()` yang berfungsi untuk menampilkan data training, method `kuadrat()` yang berfungsi untuk menampilkan kuadrat setiap data training, method `normalisasi()` yang berfungsi untuk menampilkan hasil normalisasi dataset dan data training, method `knn()` yang berfungsi untuk menampilkan hasil perhitungan metode, method `hitung()` yang berfungsi untuk menampilkan perhitungan dalam metode *K-Nearest Neighbor* yang ditampilkan pada halaman kuadrat, normalisasi, dan knn. Penulisan kode program pada cperhitungan dapat dilihat pada gambar 4.8

Gambar 4.8 Kode Program cperhitungan

```
<?php if ( ! defined('BASEPATH')) exit('No direct script access allowed');

class Cperhitungan extends CI_Controller {

public function __construct(){
parent::__construct();
$this->load->model('mdataset');
$this->load->model('mdatatraining');
}

public function simpan(){
$hasilAkhir = $this->input->get("hasil");
$cek = $this->mdataset->simpan_hasil($hasilAkhir);
if ($cek==true) {
$this->session->set_flashdata('message', "Insert Berhasil!");
redirect('admin/cperhitungan/normalisasi');
}else{
$this->session->set_flashdata('message', "Insert Gagal!");
```



```

redirect('admin/cperhitungan/knn/sorted');
    }
}

public function show_hitung(){
    $data = $this->hitung();
    $this->load->view('admin/v_hitung', $data);
}

public function kuadrat(){
    $data['data_training']=$this->mdatatraining->get_data_training()->result_array();
    $data['hitungan'] = $this->hitung();
    $this->load->view('admin/v_kuadrat', $data);
}

public function normalisasi(){
    $data['dataset']=$this->mdataset->get_dataset2()->result_array();
    $data['data_training']=$this->mdatatraining->get_data_training()->result_array();
    $data['hitungan'] = $this->hitung();
    $this->load->view('admin/v_normalisasi', $data);
}

public function knn($sorted=FALSE){
    $data['data_training']=$this->mdatatraining->get_data_training()->result_array();
    $data['hitungan'] = $this->hitung();
    foreach ($data['data_training'] as $key => $row ) {
        $data['knn'][$row['id_siswa']]=sqrt(((($data['hitungan']['normalisasi_rbin'][$key]-
        $data['hitungan']['uji_normalisasi_rbin'][0])) * ((($data['hitungan']['normalisasi_rbin'][$key]-
        $data['hitungan']['uji_normalisasi_rbin'][0])) +
        ((($data['hitungan']['normalisasi_rbig'][$key]-$data['hitungan']['uji_normalisasi_rbig'][0])) *
        ((($data['hitungan']['normalisasi_rbig'][$key]-$data['hitungan']['uji_normalisasi_rbig'][0])) +
        ((($data['hitungan']['normalisasi_rmtk'][$key]-$data['hitungan']['uji_normalisasi_rmtk'][0])) *
        ((($data['hitungan']['normalisasi_rmtk'][$key]-$data['hitungan']['uji_normalisasi_rmtk'][0])) +
        ((($data['hitungan']['normalisasi_ripa'][$key]-$data['hitungan']['uji_normalisasi_ripa'][0])) *

```



```

(($data['hitungan']['normalisasi_ripa'][$key]-$data['hitungan']['uji_normalisasi_ripa'][0])) +
(($data['hitungan']['normalisasi_rips'][$key]-$data['hitungan']['uji_normalisasi_rips'][0])) *
(($data['hitungan']['normalisasi_rips'][$key]-$data['hitungan']['uji_normalisasi_rips'][0])) +
(($data['hitungan']['normalisasi_jumNun'][$key]-$data['hitungan']['uji_normalisasi_jumNun'][0])) *
(($data['hitungan']['normalisasi_jumNun'][$key]-$data['hitungan']['uji_normalisasi_jumNun'][0])) +
(($data['hitungan']['normalisasi_rerata_ipa'][$key]-$data['hitungan']['uji_normalisasi_rerata_ipa'][0]))
*
(($data['hitungan']['normalisasi_rerata_ipa'][$key]-$data['hitungan']['uji_normalisasi_rerata_ipa'][0]))
+
Dilanjutkan
Lanjutan
(($data['hitungan']['normalisasi_rerata_ips'][$key]-$data['hitungan']['uji_normalisasi_rerata_ips'][0])) *
(($data['hitungan']['normalisasi_rerata_ips'][$key]-$data['hitungan']['uji_normalisasi_rerata_ips'][0])) +

(($data['hitungan']['normalisasi_a'][$key]-$data['hitungan']['uji_normalisasi_a'][0])) *
(($data['hitungan']['normalisasi_a'][$key]-$data['hitungan']['uji_normalisasi_a'][0])) +
(($data['hitungan']['normalisasi_s'][$key]-$data['hitungan']['uji_normalisasi_s'][0])) *
(($data['hitungan']['normalisasi_s'][$key]-$data['hitungan']['uji_normalisasi_s'][0]));

$data['jurusan'][$row['id_siswa']]=$row['hasil'];
$data['nama'][$row['id_siswa']]=$row['nama'];
}
if ($sorted) {
    asort($data['knn']);
}
$data['limit'] = $this->input->get('k_value');
$data['limit2'] = $this->input->get('k_value');
if (empty($data['limit'])) {
    $data['limit'] = count($data['knn']);
}
$data['sorted'] = $sorted;
$this->load->view('admin/v_knn', $data);
}

```

```
public function hitungan(){
    $data['dataset']=$this->mdataset->get_dataset2()->result_array();// yg satu
    $data['data_training']=$this->mdatatraining->get_data_training()->result_array();

    $jumlah_data = count($data['data_training']);
    $data['jumlah_rbin'] = 0;
    $data['jumlah_rbig'] = 0;
    $data['jumlah_rmtk'] = 0;
    $data['jumlah_ripa'] = 0;
    $data['jumlah_rips'] = 0;
    $data['jumlah_jumNun'] = 0;
    $data['jumlah_rerata_ipa'] = 0;
    $data['jumlah_rerata_ips'] = $data['jumlah_a'] =
    0;$data['jumlah_s'] = 0;

    $data['jumlah_rbin2'] = 0;
    $data['jumlah_rbig2'] = 0;
    $data['jumlah_rmtk2'] = 0;
    $data['jumlah_ripa2'] = 0;
    $data['jumlah_rips2'] = 0;
    $data['jumlah_jumNun2'] = 0;
    $data['jumlah_rerata_ipa2'] = 0;
    $data['jumlah_rerata_ips2'] = 0;
    $data['jumlah_a2'] = 0;
    $data['jumlah_s2'] = 0;
    $data['kuadrat_rbin'] = array();
    $data['kuadrat_rbig'] = array();
    $data['kuadrat_rmtk'] = array();
    $data['kuadrat_ripa'] = array();
    $data['kuadrat_rips'] = array();
    $data['kuadrat_jumNun'] = array();
    $data['kuadrat_rerata_ipa'] = array();
    $data['kuadrat_rerata_ips'] = array();
    $data['kuadrat_a'] = array();
```

```
$data['kuadrat_s'] = array();
$data['normalisasi_rbin'] = array();
$data['normalisasi_rbig'] = array();
$data['normalisasi_rmtk'] = array();
$data['normalisasi_ripa'] = array();
$data['normalisasi_rips'] = array();
$data['normalisasi_jumNUn'] = array();
$data['normalisasi_rerata_ipa'] = array();
$data['normalisasi_rerata_ips'] = array();
$data['normalisasi_a'] = array();
$data['normalisasi_s'] = array();

foreach ($data['data_training'] as $key => $row) {

    $data['jumlah_rbin'] += $row['rbin'];
    $data['jumlah_rbig'] += $row['rbig'];
    $data['jumlah_rmtk'] += $row['rmtk'];
    $data['jumlah_ripa'] += $row['ripa'];
    $data['jumlah_rips'] += $row['rips'];
    $data['jumlah_jumNun'] += $row['jumNun'];
    $data['jumlah_rerata_ipa'] += $row['rerata_ipa'];
    $data['jumlah_rerata_ips'] += $row['rerata_ips'];
    $data['jumlah_a'] += $row['a'];
    $data['jumlah_s'] += $row['s'];
}

$data['jumlah_rbin2'] = $data['jumlah_rbin']*$data['jumlah_rbin'];
$data['jumlah_rbig2'] = $data['jumlah_rbig']*$data['jumlah_rbig'];
$data['jumlah_rmtk2'] = $data['jumlah_rmtk']*$data['jumlah_rmtk'];
$data['jumlah_ripa2'] = $data['jumlah_ripa']*$data['jumlah_ripa'];
$data['jumlah_rips2'] = $data['jumlah_rips']*$data['jumlah_rips'];
$data['jumlah_jumNun2'] = $data['jumlah_jumNun']*$data['jumlah_jumNun'];
$data['jumlah_rerata_ipa2'] = $data['jumlah_rerata_ipa']*$data['jumlah_rerata_ipa'];
$data['jumlah_rerata_ips2'] = $data['jumlah_rerata_ips']*$data['jumlah_rerata_ips'];
$data['jumlah_a2'] = $data['jumlah_a']*$data['jumlah_a'];
```

```

$data['jumlah_s2']          = $data['jumlah_s']*$data['jumlah_s'];

$data['mean_rbin'] = $data['jumlah_rbin']/$jumlah_data;
$data['mean_rbig'] = $data['jumlah_rbig']/$jumlah_data;
$data['mean_rmtk'] = $data['jumlah_rmtk']/$jumlah_data;
$data['mean_ripa'] = $data['jumlah_ripa']/$jumlah_data;
$data['mean_rips'] = $data['jumlah_rips']/$jumlah_data;
$data['mean_jumNun'] = $data['jumlah_jumNun']/$jumlah_data;
$data['mean_rerata_ipa'] = $data['jumlah_rerata_ipa']/$jumlah_data;
$data['mean_rerata_ips'] = $data['jumlah_rerata_ips']/$jumlah_data;
$data['mean_a'] = $data['jumlah_a']/$jumlah_data;
$data['mean_s'] = $data['jumlah_s']/$jumlah_data;
foreach ($data['data_training'] as $key => $row) {
    $data['kuadrat_rbin'][$key] = $row['rbin'] * $row['rbin'];
    $data['kuadrat_rbig'][$key] = $row['rbig'] * $row['rbig'];
    $data['kuadrat_rmtk'][$key] = $row['rmtk'] * $row['rmtk'];
    $data['kuadrat_ripa'][$key] = $row['ripa'] * $row['ripa'];
    $data['kuadrat_rips'][$key] = $row['rips'] * $row['rips'];
    $data['kuadrat_jumNun'][$key] = $row['jumNun'] * $row['jumNun'];
    $data['kuadrat_rerata_ipa'][$key] = $row['rerata_ipa'] * $row['rerata_ipa'];
    $data['kuadrat_rerata_ips'][$key] = $row['rerata_ips'] * $row['rerata_ips'];
    $data['kuadrat_a'][$key] = $row['a'] * $row['a'];
    $data['kuadrat_s'][$key] = $row['s'] * $row['s'];

    $data['jk_rbin']          = array_sum($data['kuadrat_rbin']);
    $data['jk_rbig']         = array_sum($data['kuadrat_rbig']);
    $data['jk_rmtk']         = array_sum($data['kuadrat_rmtk']);
    $data['jk_ripa']         = array_sum($data['kuadrat_ripa']);
    $data['jk_rips']         = array_sum($data['kuadrat_rips']);
    $data['jk_jumNun']       = array_sum($data['kuadrat_jumNun']);
    $data['jk_rerata_ipa']   = array_sum($data['kuadrat_rerata_ipa']);
    $data['jk_rerata_ips']   = array_sum($data['kuadrat_rerata_ips']);
    $data['jk_a']            = array_sum($data['kuadrat_a']);
    $data['jk_s']            = array_sum($data['kuadrat_s']);
}

```

```
$data['defiasi_rbin']  
=sqrt(($jumlah_data*$data['jk_rbin']-$data['jumlah_rbin2'])/$jumlah_data*($jumlah_data-1));  
$data['defiasi_rbig']  
=sqrt(($jumlah_data*$data['jk_rbig']-$data['jumlah_rbig2'])/$jumlah_data*($jumlah_data-1));  
$data['defiasi_rmtk']  
=sqrt(($jumlah_data*$data['jk_rmtk']-$data['jumlah_rmtk2'])/$jumlah_data*($jumlah_data-1));  
$data['defiasi_ripa']  
=sqrt(($jumlah_data*$data['jk_ripa']-$data['jumlah_ripa2'])/$jumlah_data*($jumlah_data-1));  
$data['defiasi_rips']  
=sqrt(($jumlah_data*$data['jk_rips']-$data['jumlah_rips2'])/$jumlah_data*($jumlah_data-1));  
$data['defiasi_jumNun']  
=sqrt(($jumlah_data*$data['jk_jumNun']-$data['jumlah_jumNun2'])/$jumlah_data*($jumlah_data-  
1));  
$data['defiasi_rerata_ipa']  
=sqrt(($jumlah_data*$data['jk_rerata_ipa']-$data['jumlah_rerata_ipa2'])/$jumlah_data*($jumlah_data-  
1));  
$data['defiasi_rerata_ips']  
=sqrt(($jumlah_data*$data['jk_rerata_ips']-$data['jumlah_rerata_ips2'])/$jumlah_data*($jumlah_data-  
1));  
$data['defiasi_a']  
=sqrt(($jumlah_data*$data['jk_a']-$data['jumlah_a2'])/$jumlah_data*($jumlah_data-1));  
$data['defiasi_s']  
=sqrt(($jumlah_data*$data['jk_s']-$data['jumlah_s2'])/$jumlah_data*($jumlah_data-1));  
  
foreach ($data['data_training'] as $key => $row) {  
$data['normalisasi_rbin'][$key]  
=number_format((float)($row['rbin']-$data['mean_rbin'])/$data['defiasi_rbin'], 10, '.', '');  
$data['normalisasi_rbig'][$key]  
=number_format((float)($row['rbig']-$data['mean_rbig'])/$data['defiasi_rbig'], 10, '.', '');  
$data['normalisasi_rmtk'][$key]  
=number_format((float)($row['rmtk']-$data['mean_rmtk'])/$data['defiasi_rmtk'], 10, '.', '');  
$data['normalisasi_ripa'][$key]  
=number_format((float)($row['ripa']-$data['mean_ripa'])/$data['defiasi_ripa'], 10, '.', '');
```



```

$data['normalisasi_rips'][$key]
=number_format((float)($row['rips']-$data['mean_rips'])/$data['defiasi_rips'], 10, '.', '');
$data['normalisasi_jumNun'][$key]
=number_format((float)($row['jumNun']-$data['mean_jumNun'])/$data['defiasi_jumNun'], 10, '.', '');
$data['normalisasi_rerata_ipa'][$key]
=number_format((float)($row['rerata_ipa']-$data['mean_rerata_ipa'])/$data['defiasi_rerata_ipa'], 10, '.',
");
$data['normalisasi_rerata_ips'][$key]
=number_format((float)($row['rerata_ips']-$data['mean_rerata_ips'])/$data['defiasi_rerata_ips'], 10, '.',
");
$data['normalisasi_a'][$key]
= number_format((float)($row['a']-$data['mean_a'])/$data['defiasi_a'], 10, '.', '');
$data['normalisasi_s'][$key]
= number_format((float)($row['s']-$data['mean_s'])/$data['defiasi_s'], 10, '.', '');
}

foreach ($data['dataset'] as $key => $row) {
$data['uji_normalisasi_rbin'][$key]
= number_format((float)($row['rbin'] - $data['mean_rbin']) / $data['defiasi_rbin'], 10, '.', '');
$data['uji_normalisasi_rbig'][$key]
= number_format((float)($row['rbig'] - $data['mean_rbig']) / $data['defiasi_rbig'], 10, '.', '');
$data['uji_normalisasi_rmtk'][$key]
= number_format((float)($row['rmtk'] - $data['mean_rmtk']) / $data['defiasi_rmtk'], 10, '.', '');
$data['uji_normalisasi_ripa'][$key]
= number_format((float)($row['ripa'] - $data['mean_ripa']) / $data['defiasi_ripa'], 10, '.', '');
$data['uji_normalisasi_rips'][$key]
= number_format((float)($row['rips'] - $data['mean_rips']) / $data['defiasi_rips'], 10, '.', '');
$data['uji_normalisasi_jumNun'][$key]
= number_format((float)($row['jumNun'] - $data['mean_jumNun']) / $data['defiasi_jumNun'], 10, '.',
");
$data['uji_normalisasi_rerata_ipa'][$key]
=number_format((float)($row['rerata_ipa']-$data['mean_rerata_ipa'])/$data['defiasi_rerata_ipa'], 10, '.',
");
$data['uji_normalisasi_rerata_ips'][$key]

```



```

=number_format((float)($row['rerata_ips']-$data['mean_rerata_ips'])/$data['defiasi_rerata_ips'], 10, '.',
");
$data['uji_normalisasi_a'][$key]
= number_format((float)($row['a'] -$data['mean_a']) /$data['defiasi_a'], 10, '.', "");
$data['uji_normalisasi_s'][$key]
= number_format((float)($row['s'] -$data['mean_s']) /$data['defiasi_s'], 10, '.', "");
}
return $data;
}
}

```

2. Class model mdataset

Pada class model mdataset terdapat method `get_dataset2` yang digunakan untuk mengambil nilai dataset yang diinputkan pada sistem, method `simpan_hasil()` yang digunakan untuk menyimpan hasil perhitungan ke dalam database. Kode sistem pada mdataset dapat dilihat pada gambar 4.9

Gambar 4.9 Kode Program mdataset

```

<?php
if(!defined('BASEPATH')) exit('No direct script access allowed');

class Mdataset extends CI_Model{

function get_dataset2(){
$sql = "SELECT nama, jenis_kelamin, id_Jurusan, s.id_siswa, bin.nilai as rbin, big.nilai as rbig,
        mtk.nilai as rmtk, ipa.nilai as ripa, ips.nilai as rips,
        (nun.nilai_bin + nun.nilai_big + nun.nilai_mtk + nun.nilai_ipa) as jumNun,
        ((bin.nilai + mtk.nilai + ipa.nilai + nun.nilai_bin + nun.nilai_mtk + nun.nilai_ipa) /
6) AS rerata_ipa, ((bin.nilai + big.nilai + ips.nilai + nun.nilai_bin + nun.nilai_big) / 5) AS rerata_ips,
hasil_s, s, hasil_a, a
FROM data_siswa s JOIN ( SELECT id_siswa, AVG(nilai) AS nilai FROM nilai_mapelsmt
WHERE mapel = 'bin' GROUP BY id_siswa) AS bin ON s.id_siswa = bin.id_siswa JOIN (

```

```
SELECT id_siswa, AVG(nilai) AS nilai FROM nilai_mapelsmt WHERE mapel = 'big' GROUP BY
id_siswa ) AS big ON s.id_siswa = big.id_siswa JOIN (
SELECT id_siswa, AVG(nilai) AS nilai FROM nilai_mapelsmt WHERE mapel = 'mtk' GROUP BY
id_siswa ) AS mtk ON s.id_siswa = mtk.id_siswa JOIN (
SELECT id_siswa, AVG(nilai) AS nilai FROM nilai_mapelsmt WHERE mapel = 'ipa' GROUP BY
id_siswa ) AS ipa ON s.id_siswa = ipa.id_siswa JOIN (
SELECT id_siswa, AVG(nilai) AS nilai FROM nilai_mapelsmt WHERE mapel = 'ips' GROUP BY
id_siswa ) AS ips ON s.id_siswa = ips.id_siswa JOIN nilai_nun AS nun ON s.id_siswa = nun.id_siswa
join
angket on angket.id_angket=s.id_angket join
saran on saran.id_saran=s.id_saran
where s.Verifikasi='1' && s.status='0' && id_Jurusan is NULL ORDER BY id_siswa ASC" ; return
$this->db->query($sql);
}

function simpan_hasil($hasil){
if($hasil=="IPA"){
    $p=1;
}else{
    $p=2;
}
$sql = " UPDATE data_siswa SET id_Jurusan = $p where id_Jurusan is NULL ORDER BY id_siswa
ASC limit 1";
return $this->db->query($sql);
}
}

?>
```

3. Class model mdatatraining

Pada *class* model mdatatraining terdapat method `get_data_training()` yang digunakan untuk mengambil nilai datatraining yang ada dalam database. Kode program mdatatraining dapat dilihat pada gambar 4.10

Gambar 4.10 Kode Program mdatatraining

```

<?php
if(!defined('BASEPATH')) exit ('No direct script access allowed');

class Mdatatraining extends CI_Model{
function get_data_training(){
$sql = "SELECT nama, jenis_kelamin, s.id_siswa, j.hasil_p as hasil, bin.nilai as rbin, big.nilai as rbig,
mtk.nilai as rmtk, ipa.nilai as ripa, ips.nilai as rips,(nun.nilai_bin + nun.nilai_big + nun.nilai_mtk +
nun.nilai_ipa) as jumNun, ((bin.nilai + mtk.nilai + ipa.nilai + nun.nilai_bin + nun.nilai_mtk +
nun.nilai_ipa) / 6) AS rerata_ipa, ((bin.nilai + big.nilai + ips.nilai + nun.nilai_bin + nun.nilai_big) / 5)
AS rerata_ips, hasil_a, a, hasil_s, s
FROM data_siswa s JOIN ( SELECT id_siswa, AVG(nilai) AS nilai FROM nilai_mapelsmt WHERE
mapel = 'bin' GROUP BY id_siswa) AS bin ON s.id_siswa = bin.id_siswa JOIN (
SELECT id_siswa, AVG(nilai) AS nilai FROM nilai_mapelsmt WHERE mapel = 'big'
GROUP BY id_siswa ) AS big ON s.id_siswa = big.id_siswa JOIN (
SELECT id_siswa, AVG(nilai) AS nilai FROM nilai_mapelsmt WHERE mapel = 'mtk' GROUP BY
id_siswa ) AS mtk ON s.id_siswa = mtk.id_siswa JOIN (
SELECT id_siswa, AVG(nilai) AS nilai FROM nilai_mapelsmt WHERE mapel = 'ipa' GROUP BY
id_siswa ) AS ipa on s.id_siswa = ipa.id_siswa JOIN (
SELECT id_siswa, AVG(nilai) AS nilai FROM nilai_mapelsmt WHERE mapel = 'ips' GROUP BY
id_siswa ) AS ips ON s.id_siswa = ips.id_siswa JOIN nilai_nun AS nun ON s.id_siswa = nun.id_siswa
join
angket on angket.id_angket=s.id_angket join
saran on saran.id_saran=s.id_saran join
Jurusan j on j.id_Jurusan=s.id_Jurusan
where s.status='1'";
return $this->db->query($sql);
    }
} ?>

```

3.4 4.4 Pengujian Sistem

Pengujian dilakukan untuk mengevaluasi aplikasi yang telah dibuat. Proses pengujian dilakukan dengan pengujian *whitebox* terlebih dahulu kemudian akan dilanjutkan dengan pengujian *blackbox*.

4.4.1 Pengujian White Box

Pengujian *white box* pada sistem ini dengan cara menggambar diagram alir, menghitung kompleksitas siklomatiknya (CC), dan membuat tabel pengujian *test case*. Pengujian *listing program* ditunjukkan pada Gambar 4.10, gambar diagram alir ditunjukkan pada Gambar 4.11 dan tabel pengujian alur *test case* ditunjukkan pada Tabel 4.4.

CC = 4

Jalur 1 = 39 – (40-42) – 49 – (50-62) – 66 – (70-71)

Jalur 2 = 39 – (40-42) – 49 – (50-62) – 66 - 67

Jalur 3 = 39 – (40-42) – 49 – (50-62) – 66 – 67 – (70-71) – 73 – 74

Jalur 4 = 39 – (40-42) – 49 – (50-62) – 66 – 67 – (70-71) – 73 – (77-78)

Tabel 4.4 Test Case Function knn()

Jalur 1	
Test Case	Menghitung <i>euclidean distance</i> dari data normalisasi <i>dataset</i> atau data uji dengan setiap masing – masing data <i>training</i>
Target yang diharapkan	Menampilkan euclidean distance
Hasil Pengujian	Benar
Path/Jalur	Jalur 1 = 39 – (40-42) – 49 – (50-62) – 66 – (70-71)

Jalur 2	
Test Case	Menghitung <i>euclidean distance</i> dari data normalisasi <i>dataset</i> atau data uji dengan setiap masing – masing data <i>training</i>
Target yang diharapkan	Menampilkan <i>euclidean distance</i> berdasarkan jarak terpendek sampai yang terjauh antara <i>dataset</i> atau data uji dengan data <i>training</i>
Hasil Pengujian	Benar
Path/Jalur	Jalur 2 = 39 – (40-42) – 49 – (50-62) – 66 – 67

Jalur 3	
<i>Test Case</i>	Menghitung <i>euclidean distance</i> dari data normalisasi <i>dataset</i> atau data uji dengan setiap masing – masing data <i>training</i>
Target yang diharapkan	Menampilkan <i>euclidean distance</i> berdasarkan jarak terpendek sampai yang terjauh antara <i>dataset</i> atau data uji dengan data <i>training</i> dengan nilai k kosong atau nilai k tidak diisi
Hasil Pengujian	Benar
Path/Jalur	Jalur 3 = 39 – (40-42) – 49 – (50-62) – 66 – 67 – (70-71) – 73 – 74

Jalur 4	
<i>Test Case</i>	Menghitung <i>euclidean distance</i> dari data normalisasi <i>dataset</i> atau data uji dengan setiap masing – masing data <i>training</i>
Target yang diharapkan	Menampilkan <i>euclidean distance</i> berdasarkan jarak terpendek sampai yang terjauh antara <i>dataset</i> atau data uji dengan data <i>training</i> dengan nilai k sudah diisi
Hasil Pengujian	Benar
Path/Jalur	Jalur 4 = 39 – (40-42) – 49 – (50-62) – 66 – 67 – (70-71) – 73 – (77-78)

4.4.2 Pengujian *Black Box*

Pengujian *black box* berfungsi untuk menguji sistem dari segi spesifikasi fungsional sistem dengan tujuan mengetahui apakah fungsi-fungsi, inputan, dan

keluaran sistem sesuai dengan spesifikasi yang dibutuhkan oleh pengguna. Hasil pengujian dengan metode *black box* dapat dilihat pada Tabel 4.5

Tabel 4. 1 Pengujian *Black Box*

No	Fitur	Aksi	Hasil	Ket
1.	Login	Mengisi kolom <i>username</i> dan <i>password</i> lalu klik tombol <i>login</i>	<i>Login</i> berhasil dan menampilkan halaman awal.	[√] Berhasil [] Gagal
		Kolom <i>username</i> atau <i>password</i> tidak sesuai	Menampilkan pesan <i>Login</i> gagal	[√] Berhasil [] Gagal
2.	Menampilkan Home	Klik menu home	Menampilkan halaman home	[√] Berhasil [] Gagal
3.	Data dan nilai	Klik menu data dan nilai	Menampilkan submenu input nilai, input data siswa	[√] Berhasil [] Gagal
		Klik submenu input data siswa	Menampilkan halaman input data siswa	[√] Berhasil [] Gagal
		Mengisi form nama, nomor pendaftaran, jenis kelamin	Insert berhasil, menampilkan halaman input data siswa	[√] Berhasil [] Gagal
		Form nama, nomor pendaftaran, jenis kelamin kosong	Menampilkan pesan “harap isi bidang ini”, “pilih salah	[√] Berhasil [] Gagal

			satu”	
		Klik submenu input nilai	Menampilkan halaman input nilai	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Gagal
		Memilih nama siswa, mengisi form bahasa indonesia, bahasa inggris, matematika, ipa, ips, nun, angket minat, saran	Insert berhasil, menampilkan halaman input nilai siswa	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Gagal
		Form bahasa indonesia, bahasa inggris, matematika, ipa, ips, nun, angket minat, saran	Menampilkan pesan “harap isi bidang ini”, “pilih salah satu”	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Gagal
4.	Tabel	Klik menu tabel	Menampilkan submenu verifikasi, dataset, data training	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Gagal
		Klik submenu verifikasi	Menampilkan halaman verifikasi	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Gagal
		Memilih nama siswa, klik	Verifikasi berhasil,	<input checked="" type="checkbox"/> Berhasil

		verifikasi	menampilkan halaman verifikasi	<input type="checkbox"/> Gagal
		Memilih nama siswa, klik edit, mengisi form, klik submit	Insert berhasil, menampilkan halaman verifikasi	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Gagal
		Memilih nama siswa, klik edit, mengisi form, klik submit (form kosong)	Menampilkan pesan “harap isis bidang ini”, “pilih salah satu”	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Gagal
		Klik submenu dataset	Menampilkan halaman tabel dataset	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Gagal
		Klik submenu data training	Menampilkan halaman tabel data training	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Gagal
5.	Perhitungan	Klik menu perhitungan	Menampilkan halaman tabel hitung data training	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Gagal
		Klik kuadrat	Menampilkan halaman kuadrat	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Gagal
		Klik normalisasi	Menampilkan halaman normalisasi	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Gagal

		Klik back	Menampilkan halaman kuadrat	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Gagal
		Klik KNN	Menampilkan halaman knn	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Gagal
		Klik sort	Menampilkan halaman knn - sorted	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Gagal
		Klik back	Menampilkan halaman normalisasi	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Gagal
		Mengisi nilai k, klik filter	Menampilkan halaman knn sorted berdasarkan nilai k, menampilkan hasil	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Gagal
		Klik submit	Menyimpan hasil ke dalam database, menampilkan halaman normalisasi	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Gagal
		Klik unsort	Menampilkan halaman knn	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Gagal

6.	Hasil perhitungan	Klik menu perhitungan	Menampilkan halaman hasil perhitungan	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Gagal
7.	<i>Logout</i>	Klik <i>Logout</i>	Menampilkan halaman <i>login</i>	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Gagal



BAB 6. PENUTUP

6.1 Kesimpulan

Kesimpulan dari penelitian ini adalah:

- a. Penerepan metode *K-Nearest Neighbor* dalam sistem rekomendasi penentuan Jurusan pada siswa menengah atas (SMA) ini mampu menentukan hasil Jurusan berdasarkan parameter yang digunakan yang nantinya akan digunakan pada rumus *Z-score* untuk menormalisasikan nilai setiap parameter. Setelah tahap normalisasi kemudian perhitungan akan dilanjutkan pada penghitungan jarak menggunakan rumus *Euclidean Distance* pada setiap parameter. Pengkalsifikasian dilakukan apabila nilai jarak antar data *training* dengan data yang diuji telah diurutkan berdasarkan nilai jarak terdekat yang kemudian ditarik kesimpulannya berdasarkan nilai K yang telah ditentukan.
- b. Pada penelitian ini metode *K-Nearest Neighbor* memiliki tingkat akurasi yaitu 90% dengan nilai $K=7$. Nilai akurasi tersebut akan bergantung dengan semakin besarnya data dan nilai K, akurasi data akan berbeda – beda hasilnya.

6.2 Saran

Beberapa saran dan masukan berikut diharapkan dapat memberikan perbaikan dalam penelitian selanjutnya, yaitu :

- a. Pengembangan sistem rekomendasi penentuan Jurusan pada siswa menengah atas (SMA) yang akan dikembangkan selanjutnya diharapkan dapat menambahkan jenis kriteria yang digunakan dalam perhitungan tanpa merubah kode pemrograman. Sehingga menjadikan sistem ini menjadi sistem yang dinamis terhadap perubahan dalam penambahan kriteria. Karena penambahan kriteria sewaktu-waktu bisa terjadi apabila ada kriteria yang perlu ditambahkan untuk proses penentuan jurusan tersebut.

- b. Penambahan jumlah *data training* yang lebih banyak sehingga dapat membantu meningkatkan tingkat akurasi hasil perhitungan pada sistem ini dengan menggunakan metode *K-Nearest Neighbor*. Penambahan jumlah data *training* perlu ditambahkan karena semakin banyak data *training* yang dipakai maka semakin banyak data *training* yg dibandingkan atau dihitung jaraknya dengan data uji atau *dataset*.



DAFTAR PUSTAKA

- Bramer, M. (2007). *Principle of Data Mining : Undergraduate Topics in Computer Science*. London: Springer Verlag.
- Desiani, A. d. (2006). *Konsep Kecerdasan Buatan*. Yogyakarta: Andi.
- Fahrurrozi, I., & SN, A. (2012). *Proses Pemodelan Software dengan Metode Waterfall dan Extreme Programming Studi Perbandingan*.
- Fahrurrozi, I., & SN, A. (Tanpa Tahun). *Proses Pemodelan Software dengan Metode Waterfall dan Extreme Programming Studi Perbandingan*.
- Han, J. a. (2006). *Datamining : Concept and Techniques*. Second Edition. Elsevier Inc. .
- Kusrini. (2009). *Algoritma Datamining*. Yogyakarta. CV. Andi Offset.
- Miah, M. (2009). *Improved k-nearest neighbor Algorithm for Text Classification*. *Journal Department of Science and Engineering*. University of Texas.
- Sommerville, I. (2011). *Software Engeenering (Rekayasa Perangkat Lunak)*. Jakarta: Erlangga.

LAMPIRAN

LAMPIRAN A

A.1 Skenario Login

Tabel A.1 Skenario Login Admin (Guru)

Nama Use Case	Login
Aktor	Guru (admin)
Deskripsi Singkat	Admin akan mengakses aplikasi
Prekondisi	<i>Username</i> dan <i>password</i> yang akan digunakan untuk <i>login</i>
Prakondisi	Berhasil <i>login</i>
Flow Events	
Skenario Normal : <i>Login</i>	
Aksi Aktor	Reaksi Sistem
1. Membuka halaman website	
	2. Menampilkan halaman utama <i>login</i>
3. Menginputkan <i>username</i> dan <i>password</i>	
4. Klik tombol <i>login</i>	
	5. Mengecek ke database
	6. Menampilkan pesan hello admin pada halaman home guru (admin)
Skenario Alternatif : <i>Username dan Password Salah</i>	
Aksi Aktor	Reaksi Sistem
6a. Klik tombol <i>login</i>	
	7a. Menampilkan pesan “login gagal”
Skenario Alternatif : <i>Memilih tombol Logout</i>	
Aksi Aktor	Reaksi Sistem
1. Klik tombol <i>logout</i>	
	2. Menampilkan halaman login

Tabel A.2 Skenario Login User (Siswa)

Nama Use Case	Login
Aktor	Siswa (user)
Deskripsi Singkat	Bidan akan mengakses aplikasi
Prekondisi	<i>Username</i> dan <i>password</i> yang akan

	digunakan untuk <i>login</i>
Prakondisi	Berhasil <i>login</i>
Flow Events	
Skenario Normal : <i>Login</i>	
Aksi Aktor	Reaksi Sistem
1. Membuka halaman website	
	2. Menampilkan halaman utama <i>login</i>
3. Menginputkan <i>username</i> dan password	
4. Klik tombol <i>login</i>	
	5. Mengecek ke database
	6. Menampilkan pesan hello user pada halaman home siswa (user)
Skenario Alternatif : <i>Username dan Password Salah</i>	
Aksi Aktor	Reaksi Sistem
6a. Klik tombol <i>login</i>	
	7a. Menampilkan span “login gagal”
Skenario Alternatif : <i>Memilih tombol Logout</i>	
Aksi Aktor	Reaksi Sistem
1. Klik tombol <i>logout</i>	
	2. Menampilkan halaman login

A.2 Skenario Menambah Data Siswa

Tabel A.3 Skenario Menambah Data Siswa

Nama Use Case	Tambah data siswa
Aktor	Admin (guru)
Deskripsi Singkat	Admin akan menambahkan data siswa
Prekondisi	Admin memilih menu data dan nilai
Prakondisi	Data siswa berhasil ditambahkan
Flow Events	
Skenario Normal : <i>Tambah Data Siswa</i>	
Aksi Aktor	Reaksi Sistem
1. Klik menu data dan nilai	
	2. Menampilkan submenu input data siswa, input nilai (dataset)
3. Klik menu input data siswa	
	4. Menampilkan halaman form input data siswa, menampilkan form biodata siswa meliputi nama siswa,

	no pendaftaran, jenis kelamin dan tombol submit
5. Mengisi form	
6. Klik tombol Submit	
	7. Menyimpan ke dalam database
	8. Menampilkan pesan “input berhasil, selanjutnya silahkan mengisi form nilai”, menampilkan halaman form input data siswa, menampilkan form biodata siswa meliputi nama lengkap, no pendaftaran, jenis kelamin dan tombol submit
Skenario Alternatif : Form Data Tidak Lengkap	
Aksi Aktor	Reaksi Sistem
6a. Klik tombol Submit	
	7a. Menampilkan pesan “Pilih salah satu opsi berikut, harap isi bidang ini”
Skenario Alternatif : Nomor Pendaftaran Sudah Digunakan	
6a. Klik tombol Submit	
	7a. Menampilkan pesan “Insert Gagal, data yang dimasukkan sudah ada dalam database”

Tabel A.4 Skenario Menambah Data Siswa

Nama Use Case	Tambah data siswa
Aktor	User (Siswa)
Deskripsi Singkat	User akan menambahkan data siswa
Prekondisi	User memilih menu data dan nilai
Prakondisi	Data siswa berhasil ditambahkan
Flow Events	
Skenario Normal : Tambah Data Siswa	
Aksi Aktor	Reaksi Sistem
1. Klik menu data dan nilai	
	2. Menampilkan submenu input data siswa, input nilai (dataset)
3. Klik menu input data siswa	
	4. Menampilkan halaman form input data siswa, menampilkan form biodata siswa meliputi nama siswa,

	no pendaftaran, jenis kelamin dan tombol submit
5. Mengisi form	
6. Klik tombol Submit	
	7. Menyimpan ke dalam database
	8. Menampilkan pesan “input berhasil, selanjutnya silahkan mengisi form nilai”, menampilkan halaman form input data siswa, menampilkan form biodata siswa meliputi nama lengkap, no pendaftaran, jenis kelamin dan tombol submit
Skenario Alternatif : Form Data Tidak Lengkap	
Aksi Aktor	Reaksi Sistem
6a. Klik tombol Submit	
	7a. Menampilkan pesan “Pilih salah satu opsi berikut, harap isi bidang ini”
Skenario Alternatif : Nomor Pendaftaran Sudah Digunakan	
6a. Klik tombol Submit	
	7a. Menampilkan pesan “Insert Gagal, data yang dimasukkan sudah ada dalam database”

A.3 Skenario Menambah Nilai (dataset)

Tabel A.5 Skenario Menambah Nilai (Dataset)

Nama Use Case	Tambah nilai (dataset)
Aktor	Admin (guru)
Deskripsi Singkat	Admin akan menambahkan nilai
Prekondisi	Admin memilih menu data dan nilai
Prakondisi	Nilai berhasil ditambahkan
Flow Events	
Skenario Normal : Tambah Nilai (dataset)	
Aksi Aktor	Reaksi Sistem
1. Klik menu data dan nilai	
	2. Menampilkan submenu input data siswa, input nilai (dataset)
3. Klik menu input nilai (dataset)	

	4. Menampilkan halaman form input nilai, meliputi dropdown nama siswa, menampilkan form Input Data Nilai Raport dan NUN SMP Siswa Baru (Dataset) (Bahasa Indonesia, Bahasa Inggris, Matematika, IPA, IPS, NUN, Angket, Minat) dan tombol submit
5. Mengisi form	
6. Klik tombol Submit	
	7. Menyimpan ke dalam database
	8. Menampilkan pesan “input berhasil”, menampilkan halaman verifikasi dan edit, meliputi drop down nama siswa, form edit disable (Bahasa Indonesia, Bahasa Inggris, Matematika, Ipa, Ips, Nun, angket minat, Saran), tombol edit, tombol verifikasi
9. Memilih nama siswa	
10. Klik tombol verifikasi	
	11. Menyimpan ke dalam database
	12. Menampilkan pesan “input berhasil”, menampilkan halaman verifikasi dan edit, meliputi drop down nama siswa, form edit disable (Bahasa Indonesia, Bahasa Inggris, Matematika, Ipa, Ips, Nun, angket minat, Saran), tombol edit, tombol verifikasi
Skenario Alternatif : Form Data Tidak Lengkap	
6a. Klik tombol Submit	
	7b. Menampilkan pesan "harap isi bidang, pilih salah satu opsi berikut", menampilkan halaman form input nilai, meliputi dropdown nama siswa, menampilkan form Input Data Nilai Raport dan NUN SMP Siswa Baru (Dataset) (Bahasa Indonesia, Bahasa Inggris, Matematika, IPA, IPS, NUN, Angket, Minat), tombol

	submit
Skenario Normal : Edit	
1. Memilih nama siswa	
2. Klik tombol edit	
	3. Menampilkan halaman verifikasi dan edit, meliputi drop down nama siswa, form edit available meliputi Bahasa Indonesia, Bahasa Inggris, Matematika, Ipa, Ips, Nun, angket minat, Saran, tombol edit, tombol verifikasi
4. Mengisi form	
5. Klik tombol Update	
	6. Menyimpan ke dalam database
	7. Menampilkan pesan “Verifikasi berhasil”, menampilkan halaman verifikasi dan edit, meliputi drop down nama siswa, form edit disable meliputi Bahasa Indonesia, Bahasa Inggris, Matematika, Ipa, Ips, Nun, angket minat, Saran, tombol edit, tombol update, tombol verifikasi
Skenario Alternatif : Form Data Tidak Lengkap	
Aksi Aktor	Reaksi Sistem
5a. Klik tombol update	
	6a. Menampilkan pesan "harap isi bidang, pilih salah satu opsi berikut", menampilkan halaman nilai dasar, meliputi drop down nama siswa, form edit available meliputi Bahasa Indonesia, Bahasa Inggris, Matematika, Ipa, Ips, Nun, angket minat, Saran, tombol edit, tombol verifikasi

Tabel A.6 Skenario Menambah Nilai (Dataset)

Nama Use Case	Tambah nilai (dataset)
Aktor	User (Siswa)
Deskripsi Singkat	Siswa akan menambahkan nilai
Prekondisi	Siswa memilih menu data dan nilai
Prakondisi	Nilai berhasil ditambahkan

Flow Events	
Skenario Normal : Tambah Nilai (dataset)	
Aksi Aktor	Reaksi Sistem
1. Klik menu data dan nilai	
	2. Menampilkan submenu input data siswa, input nilai (dataset)
3. Klik menu input nilai (dataset)	
	4. Menampilkan halaman form input nilai, meliputi dropdown nama siswa, menampilkan form Input Data Nilai Raport dan NUN SMP Siswa Baru (Dataset) (Bahasa Indonesia, Bahasa Inggris, Matematika, IPA, IPS, NUN, Angket, Minat) dan tombol submit
5. Mengisi form	
6. Klik tombol Submit	
	7. Menyimpan ke dalam database
	8. Menampilkan pesan "input berhasil", menampilkan halaman verifikasi dan edit, meliputi dropdown nama siswa, form edit disable (Bahasa Indonesia, Bahasa Inggris, Matematika, Ipa, Ips, Nun, angket minat, Saran), tombol edit, tombol verifikasi
Skenario Alternatif : Form Data Tidak Lengkap	
6a. Klik tombol Submit	
	7b. Menampilkan pesan "harap isi bidang, pilih salah satu opsi berikut", menampilkan halaman form input nilai, meliputi dropdown nama siswa, menampilkan form Input Data Nilai Raport dan NUN SMP Siswa Baru (Dataset) (Bahasa Indonesia, Bahasa Inggris, Matematika, IPA, IPS, NUN, Angket, Minat), tombol submit

A.4 Skenario Verifikasi

Tabel A.7 Skenario Menambah Nilai (Dataset)

Nama Use Case	Verifikasi
Aktor	Admin (guru)
Deskripsi Singkat	Admin akan melakukan verifikasi
Prekondisi	Admin memilih menu tabel
Prakondisi	Admin berhasil memverifikasi
Flow Events	
Skenario Normal : Tambah Nilai (dataset)	
Aksi Aktor	Reaksi Sistem
1. Klik menu tabel	
	2. Menampilkan submenu verifikasi, data training, dataset
3. Klik Verifikasi	
	4. Menampilkan halaman verifikasi dan edit, meliputi drop down nama siswa, form edit disable meliputi Bahasa Indonesia, Bahasa Inggris, Matematika, Ipa, Ips, Nun, angket minat, Saran, tombol edit, tombol verifikasi
5. Memilih nama	
6. Klik verifikasi	
	7. Menyimpan ke dalam database
	8. Menampilkan pesan “ Verifikasi berhasil”, menampilkan halaman verifikasi dan edit, meliputi drop down nama siswa, form edit disable meliputi Bahasa Indonesia, Bahasa Inggris, Matematika, Ipa, Ips, Nun, angket minat, Saran, tombol edit, tombol update, tombol verifikasi
Skenario Normal : Edit	
1. Memilih nama siswa	
2. Klik tombol edit	
	3. Menampilkan halaman verifikasi dan edit, meliputi drop down nama siswa, form edit available meliputi Bahasa Indonesia, Bahasa Inggris, Matematika, Ipa, Ips, Nun, angket

	minat, Saran, tombol edit, tombol update, tombol verifikasi
4. Mengisi form	
5. Klik tombol update	
	6. Menyimpan ke dalam database
	7. Menampilkan pesan “Input Data berhasil”, menampilkan halaman verifikasi dan edit, meliputi drop down nama siswa, form edit disable meliputi Bahasa Indonesia, Bahasa Inggris, Matematika, Ipa, Ips, Nun, angket minat, Saran, tombol edit, tombol update, tombol verifikasi
Skenario Alternatif : Form Data Tidak Lengkap	
Aksi Aktor	Reaksi Sistem
5a. Klik tombol update	
	6a. Menampilkan pesan "harap isi bidang, pilih salah satu opsi berikut", menampilkan halaman nilai dasar, meliputi drop down nama siswa, form edit available meliputi Bahasa Indonesia, Bahasa Inggris, Matematika, Ipa, Ips, Nun, angket minat, Saran, tombol edit, tombol update, tombol verifikasi

A.5 Skenario Melihat Tabel Data Training

Tabel A.8 Skenario Melihat Tabel Data Training

Nama Use Case	Lihat Tabel Data Training
Aktor	Admin (Guru)
Deskripsi Singkat	Admin akan melihat tabel data training
Prekondisi	Admin memilih menu tabel
Prakondisi	Melihat tabel data training
Flow Events	
Skenario Normal : Lihat Data Desa	
Aksi Aktor	Reaksi Sistem

1. Klik menu tabel	
	2. Menampilkan Submenu data training, dataset
3. Klik menu data training	
	4. Menampilkan tabel data training

A.6 Skenario Melihat Tabel Dataset

Tabel A.9 Skenario Melihat Tabel Dataset

Nama Use Case	Lihat Tabel Dataset
Aktor	Admin (Guru)
Deskripsi Singkat	Admin akan melihat tabel dataset
Prekondisi	Admin memilih menu tabel
Prakondisi	Melihat tabel dataset
Flow Events	
Skenario Normal : Lihat Data Desa	
Aksi Aktor	Reaksi Sistem
5. Klik menu tabel	
	6. Menampilkan Submenu data training, dataset
7. Klik menu dataset	
	8. Menampilkan tabel dataset

A.7 Skenario Mengelola Perhitungan

Tabel A.10 Skenario Mengelola Perhitungan

Nama Use Case	Mengelola Data Dusun
Aktor	Admin (guru)
Deskripsi Singkat	Admin akan mengelola perhitungan
Prekondisi	Admin memilih menu perhitungan
Prakondisi	<ol style="list-style-type: none"> 1. Menampilkan halaman tabel perhitungan data training 2. Menampilkan halaman tabel kuadrat data training 3. Menampilkan halaman tabel normalisasi 4. Menampilkan halaman tabel KNN - Sorting 5. Menampilkan halaman tabel KNN

- Sorting – Sorted	
Flow Events	
Skenario Normal : Melihat Tabel Perhitungan Data Training	
Aksi Aktor	Reaksi Sistem
1. Klik menu perhitungan	
	2. Menampilkan halaman hitung, meliputi tabel data training (nomor, nama, jenis kelamin, rbig, rbin, rmtk, ripa, rips, jumnun, rerataIPA, rerataIPS, saran, angket), tombol kuadrat
Skenario Normal : Melihat Tabel Kuadrat Data Training	
Aksi Aktor	Reaksi Sistem
1. Klik menu perhitungan	
	2. Menampilkan halaman hitung, meliputi tabel data training (nomor, nama, jenis kelamin, rbig, rbin, rmtk, ripa, rips, jumnun, rerataIPA, rerataIPS, saran, angket), tombol kuadrat
3. Klik tombol tombol kuadrat	
	4. Menampilkan halaman kuadrat, meliputi tabel data training (Menampilkan halaman hitung, meliputi tabel data training (nomor, nama, jenis kelamin, rbig, rbin, rmtk, ripa, rips, jumnun, rerataIPA, rerataIPS, saran, angket), tombol kuadrat), tombol back, tombol normalisasi
Skenario Alternatif : Memilih Tombol Back	
3a. Klik tombol back	
	4a. Menampilkan halaman hitung, meliputi tabel data training (nomor, nama, jenis kelamin, rbig, rbin, rmtk, ripa, rips, jumnun, rerataIPA, rerataIPS, saran, angket), tombol kuadrat
Skenario Normal : Melihat Tabel Normalisasi	
Aksi Aktor	Reaksi Sistem
1. Klik menu perhitungan	
	2. Menampilkan halaman hitung,

	meliputi tabel data training (nomor, nama, jenis kelamin, rbig, rbin, rmtk, ripa, rips, jumnun, rerataIPA, rerataIPS, saran, angket), tombol kuadrat
3. Klik tombol kuadrat	
	4. Menampilkan halaman kuadrat, meliputi tabel data training (nomor, nama, jenis kelamin, rbig, rbin, rmtk, ripa, rips, jumnun, rerataIPA, rerataIPS, saran, angket), tombol kuadrat), tombol back, tombol normalisasi
5. Klik tombol normalisasi	
	6. Menampilkan halaman normalisasi, meliputi tabel dataset (nomor, nama, jenis kelamin, rbig, rbin, rmtk, ripa, rips, jumnun, rerataIPA, rerataIPS, saran, angket), tabel data training (nomor, nama, jenis kelamin, rbig, rbin, rmtk, ripa, rips, jumnun, rerataIPA, rerataIPS, saran, angket), tombol back, tombol KNN
Skenario Alternatif : Memilih Tombol Back	
5a. Klik tombol back	
	6a. Menampilkan halaman kuadrat, meliputi tabel data training (nomor, nama, jenis kelamin, rbig, rbin, rmtk, ripa, rips, jumnun, rerataIPA, rerataIPS, saran, angket), tombol kuadrat), tombol back, tombol normalisasi
Skenario Normal : Melihat Tabel KNN – Sorting	
Aksi Aktor	Reaksi Sistem
1. Klik menu perhitungan	
	2. Menampilkan halaman hitung, meliputi tabel data training (nomor, nama, jenis kelamin, rbig, rbin, rmtk, ripa, rips, jumnun, rerataIPA, rerataIPS, saran, angket), tombol kuadrat

3. Klik tombol kuadrat	
	4. Menampilkan halaman kuadrat, meliputi tabel data training (nomor, nama, jenis kelamin, rbig, rbin, rmtk, ripa, rips, jumnun, rerataIPA, rerataIPS, saran, angket), tombol kuadrat), tombol back, tombol normalisasi
5. Klik tombol normalisasi	
	6. Menampilkan halaman normalisasi, meliputi tabel dataset (nomor, nama, jenis kelamin, rbig, rbin, rmtk, ripa, rips, jumnun, rerataIPA, rerataIPS, saran, angket), tabel data training (nomor, nama, jenis kelamin, rbig, rbin, rmtk, ripa, rips, jumnun, rerataIPA, rerataIPS, saran, angket), tombol back, tombol KNN
7. Klik tombol KNN	
	8. Menampilkan halaman tabel KNN, meliputi tabel euclidean distance (no, nama, euclidean distance, Jurusan), tombol back, tombol sort
Skenario Alternatif : Memilih Tombol Back	
7a. Klik tombol back	
	8a. Menampilkan halaman normalisasi, meliputi tabel dataset (nomor, nama, jenis kelamin, rbig, rbin, rmtk, ripa, rips, jumnun, rerataIPA, rerataIPS, saran, angket), tabel data training (nomor, nama, jenis kelamin, rbig, rbin, rmtk, ripa, rips, jumnun, rerataIPA, rerataIPS, saran, angket), tombol back, tombol KNN
Skenario Normal : Melihat Tabel KNN – Sorting – Sorted	
Aksi Aktor	Reaksi Sistem
1. Klik menu perhitungan	
	2. Menampilkan halaman hitung, meliputi tabel data training (nomor, nama, jenis kelamin, rbig, rbin,

	rmtk, ripa, rips, jumnun, rerataIPA, rerataIPS, saran, angket), tombol kuadrat
3. Klik tombol tombol kuadrat	
	4. Menampilkan halaman kuadrat, meliputi tabel data training (nomor, nama, jenis kelamin, rbig, rbin, rmtk, ripa, rips, jumnun, rerataIPA, rerataIPS, saran, angket), tombol kuadrat), tombol back, tombol normalisasi
5. Klik tombol normalisasi	
	6. Menampilkan halaman normalisasi, meliputi tabel dataset (nomor, nama, jenis kelamin, rbig, rbin, rmtk, ripa, rips, jumnun, rerataIPA, rerataIPS, saran, angket), tabel data training (nomor, nama, jenis kelamin, rbig, rbin, rmtk, ripa, rips, jumnun, rerataIPA, rerataIPS, saran, angket), tombol back, tombol KNN
7. Klik tombol KNN	
	8. Menampilkan halaman tabel KNN, meliputi tabel euclidean distance (no, nama, euclidean distance, Jurusan), tombol back, tombol sort
9. Klik tombol sort	
	10. Menampilkan halaman tabel KNN –sorted, meliputi tabel euclidean distance (no, nama, euclidean distance, Jurusan), kolom text field, tombol filter, tombol unsort
11. Masukkan nilai k	
12. Klik tombol filter	
	13. Menampilkan halaman tabel KNN – Sorted – k value, meliputi tabel euclidean distance (no, nama, euclidean distance, Jurusan)sesuai k value, kolom text field, tombol filter, text label hasil, tombol unsort, tombol submit

14. Klik Submit	
	15. Menyimpan ke dalam database
	16. Menampilkan halaman normalisasi, meliputi tabel dataset (nomor, nama, jenis kelamin, rbig, rbin, rmtk, ripa, rips, jumnun, rerataIPA, rerataIPS, saran, angket), tabel data training (nomor, nama, jenis kelamin, rbig, rbin, rmtk, ripa, rips, jumnun, rerataIPA, rerataIPS, saran, angket), tombol back, tombol KNN
Skenario Alternatif : Memilih Tombol Unsort	
11a. Klik tombol Unsort	
	12b. Menampilkan halaman tabel KNN - Sorting, meliputi tabel euclidean distance (no, nama, euclidean distance, Jurusan), tombol back, tombol sort

A.8 Skenario Melihat Data Hasil Perhitungan

Tabel A.11 Melihat Data Hasil Perhitungan

Nama Use Case	Melihat Data Hasil Perhitungan
Aktor	Admin (guru)
Deskripsi Singkat	Admin akan melihat data hasil perhitungan
Prekondisi	Admin memilih menu tabel hasil perhitungan
Prakondisi	Melihat data hasil perhitungan
Flow Events	
Skenario Normal : Melihat Data Hasil Perhitungan	
Aksi Aktor	Reaksi Sistem
1. Klik menu tabel hasil perhitungan	
	2. Menampilkan halaman hasil, meliputi tabel data hasil perhitungan (no, nama, Jurusan)

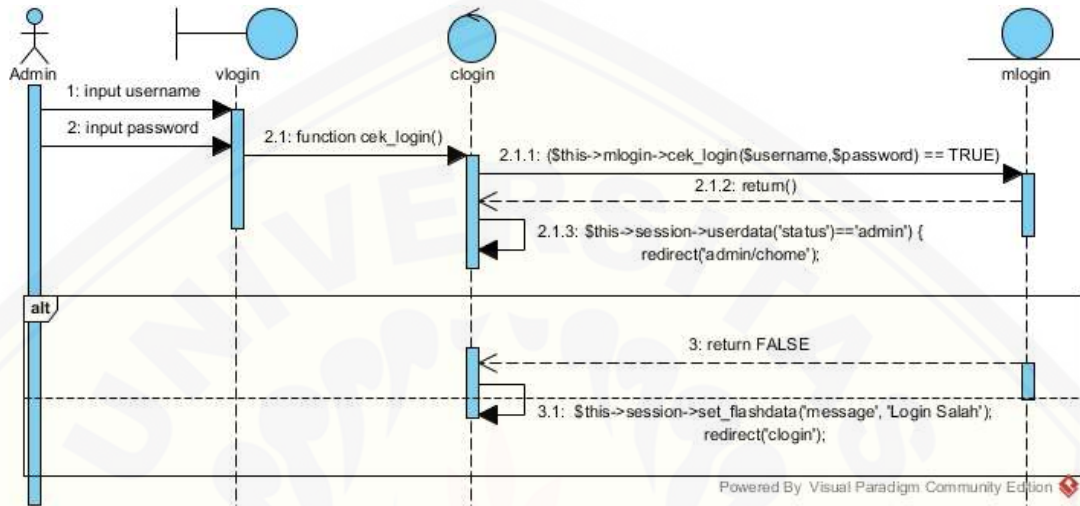
Tabel A.12 Skenario Melihat Data Hasil Perhitungan

Nama Use Case	Melihat Data Hasil Perhitungan
Aktor	User (siswa)
Deskripsi Singkat	Siswa akan melihat data hasil perhitungan
Prekondisi	Siswa memilih menu tabel hasil perhitungan
Prakondisi	Melihat data hasil perhitungan
Flow Events	
Skenario Normal : Melihat Data Hasil Perhitungan	
Aksi Aktor	Reaksi Sistem
3. Klik menu tabel hasil perhitungan	
	4. Menampilkan halaman hasil, meliputi tabel data hasil perhitungan (no, nama, Jurusan)

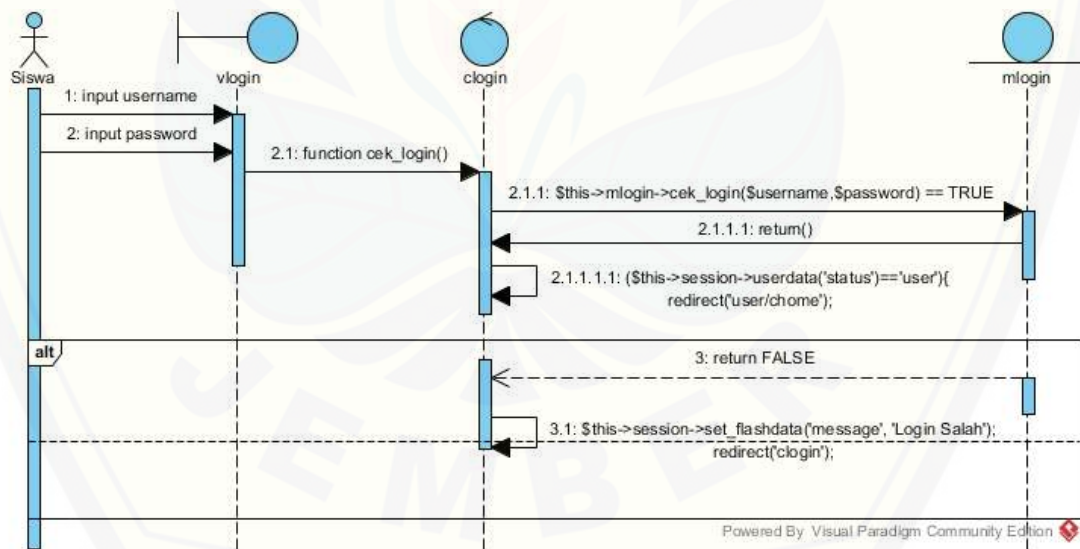
LAMPIRAN B

B.1 Sequence Diagram Login

Gambar B.1 Sequence Diagram Login Admin

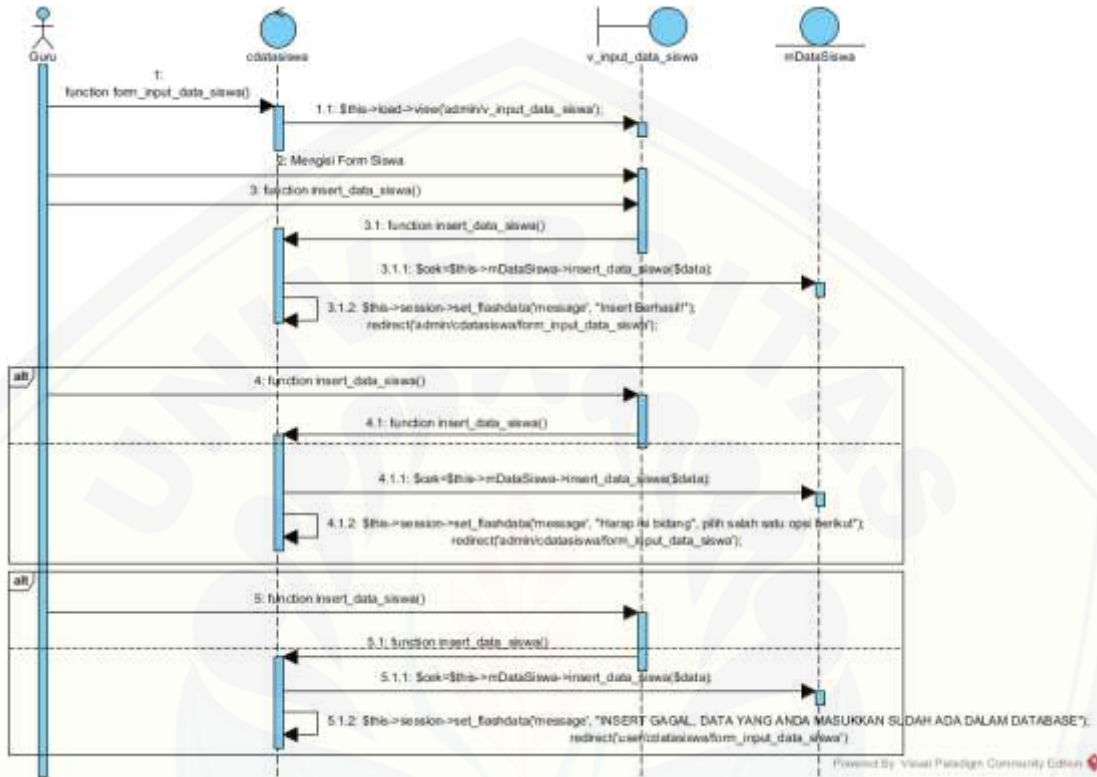


Gambar B.2 Sequence Diagram Login Siswa

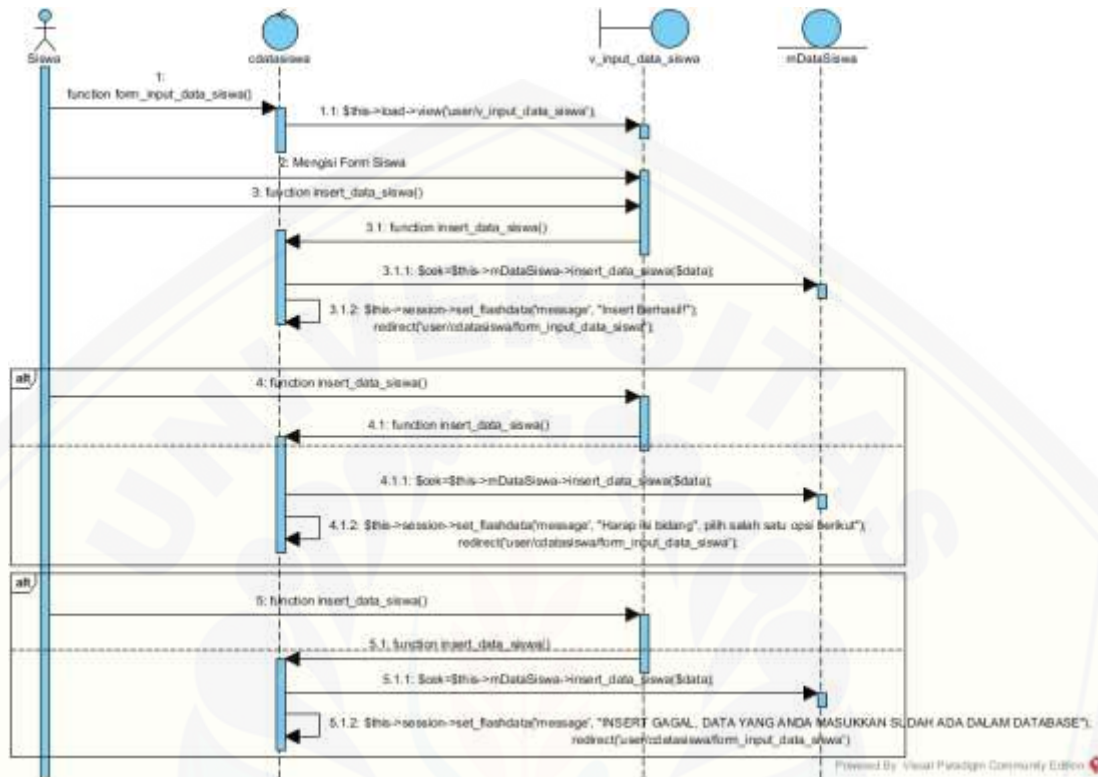


B.2 Sequence Diagram Menambah Data Siswa

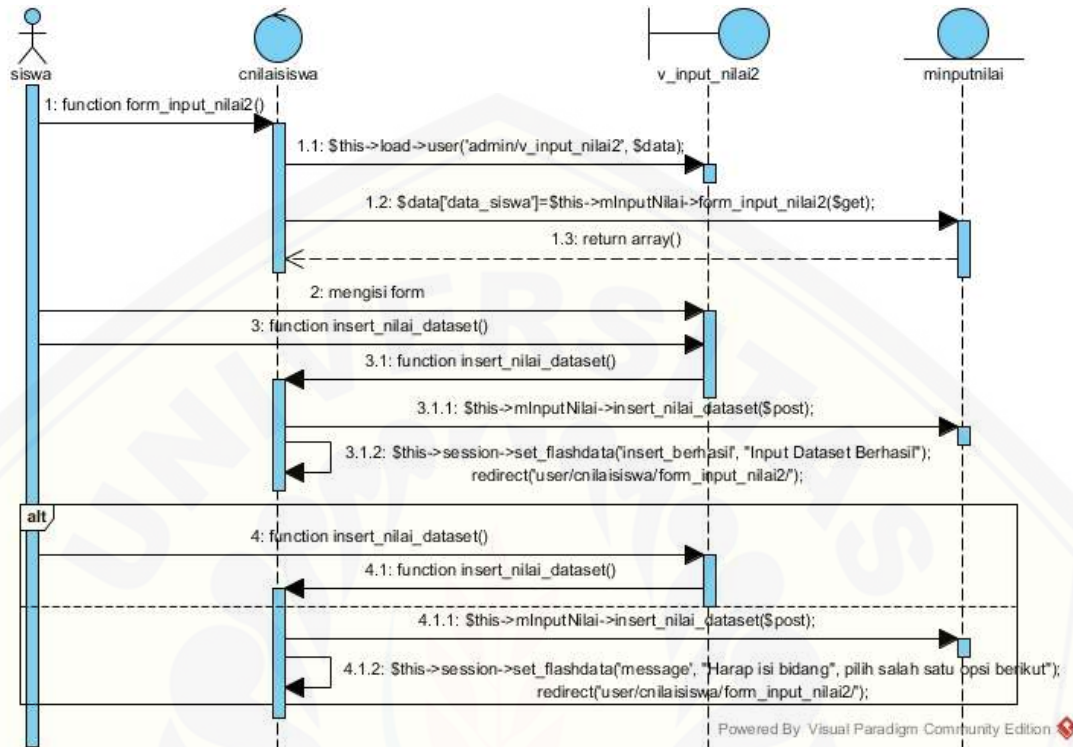
Gambar B.3 Sequence Diagram Menambah Data Siswa (Admin)



Gambar B.4 Sequence Diagram Menambah Data Siswa (Siswa)

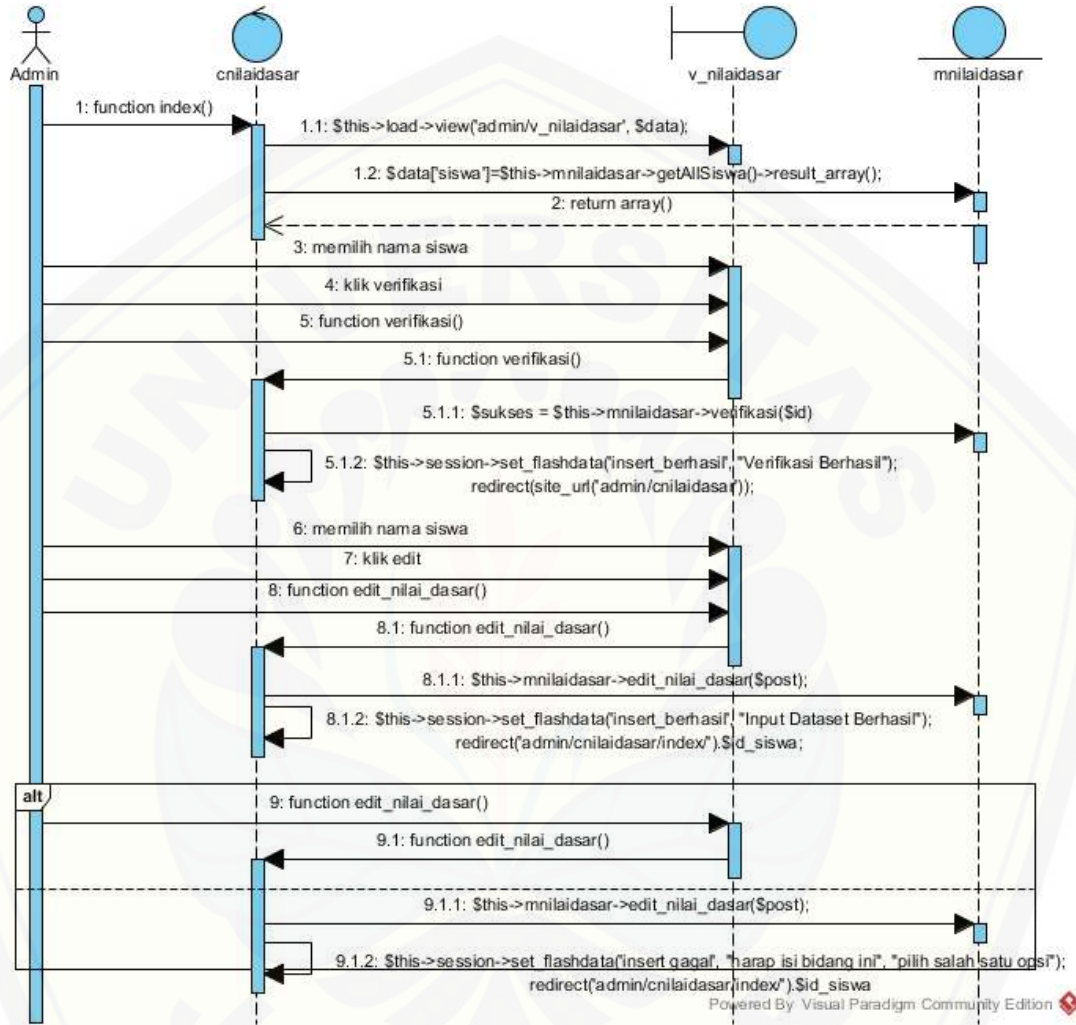


Gambar B.6 Sequence Diagram Menambah Nilai (Siswa)



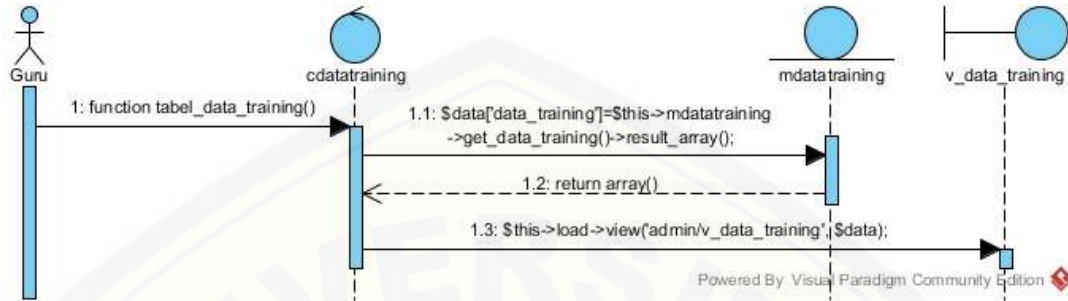
B.4 Sequence Diagram Verifikasi

Gambar B.7 Sequence Diagram Verifikasi (admin)



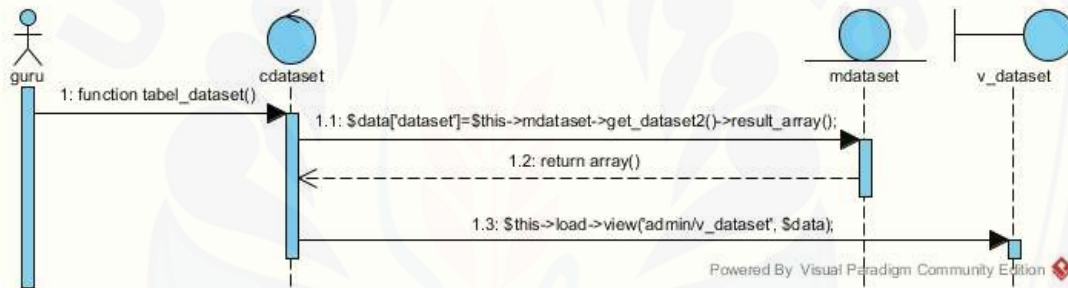
B.5 Sequence Diagram Melihat Data Training

Gambar B.8 Sequence Diagram Melihat Data Training (admin)



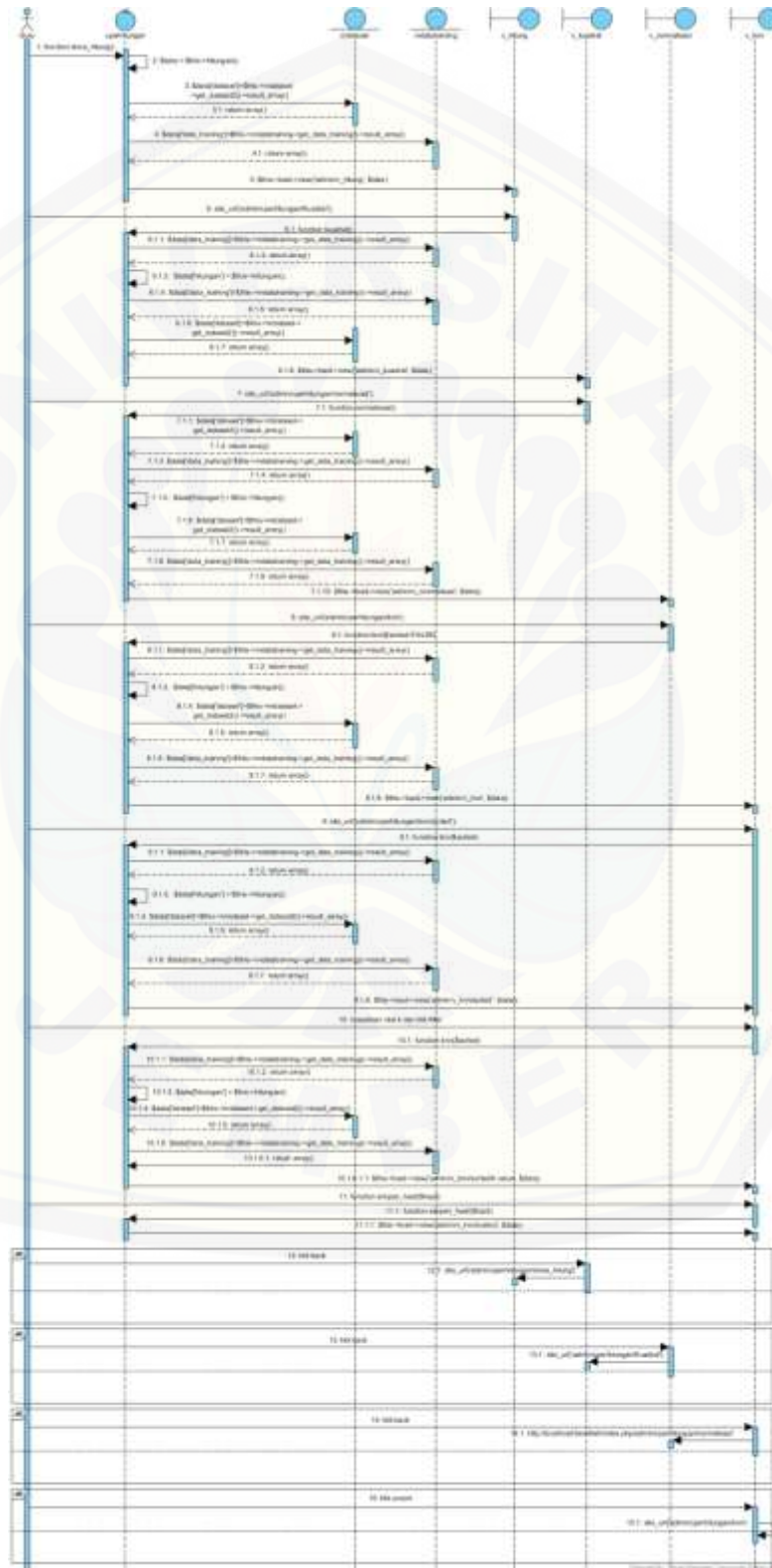
B.6 Sequence Diagram Melihat Dataset

Gambar B.9 Sequence Diagram Melihat Data Training (admin)



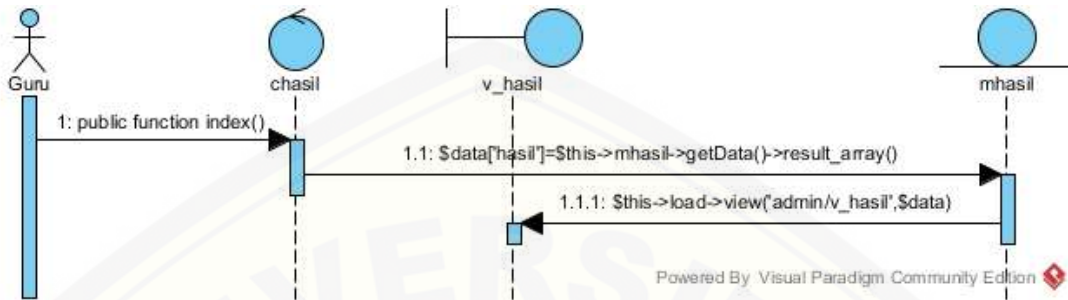
B.7 Sequence Diagram Mengelola Perhitungan

Gambar B.10 Sequence Diagram Mengelola Perhitungan (admin)

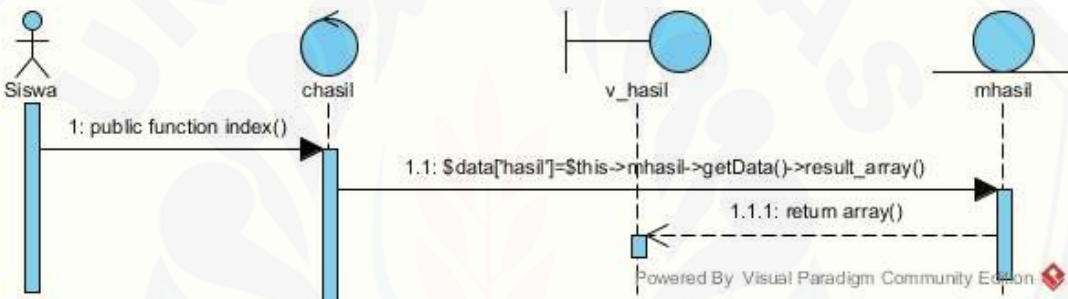


B.8 Sequence Diagram Melihat Hasil Perhitungan

Gambar B.11 Sequence Diagram Melihat Hasil Perhitungan (admin)



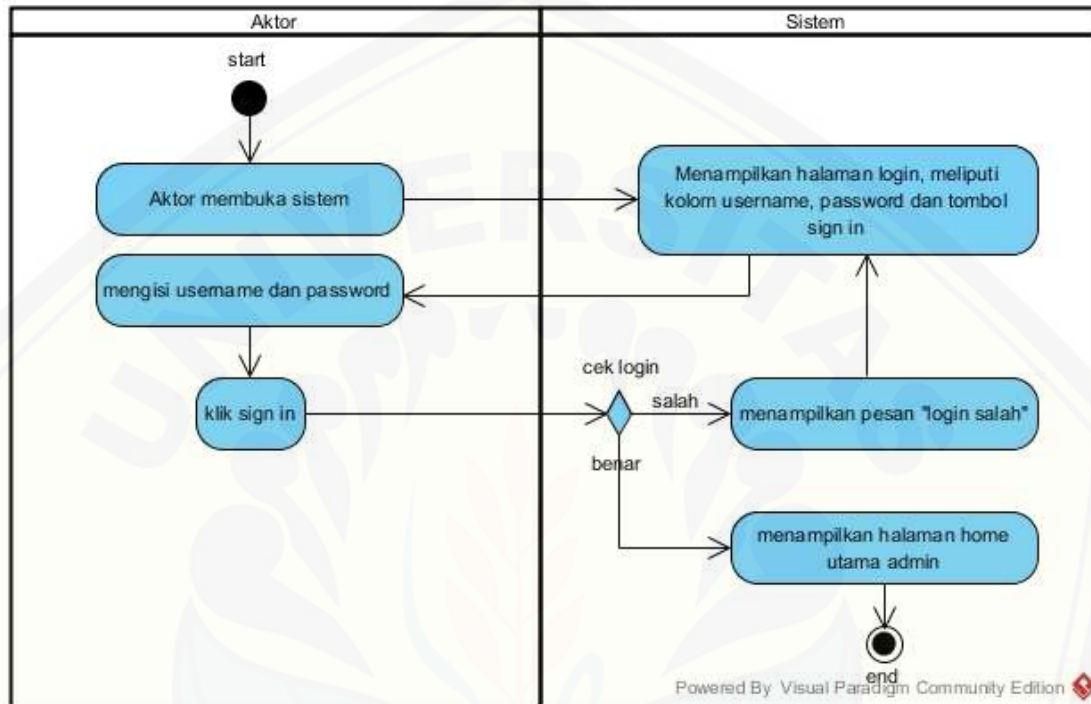
Gambar B.12 Sequence Diagram Melihat Hasil Perhitungan (Siswa)



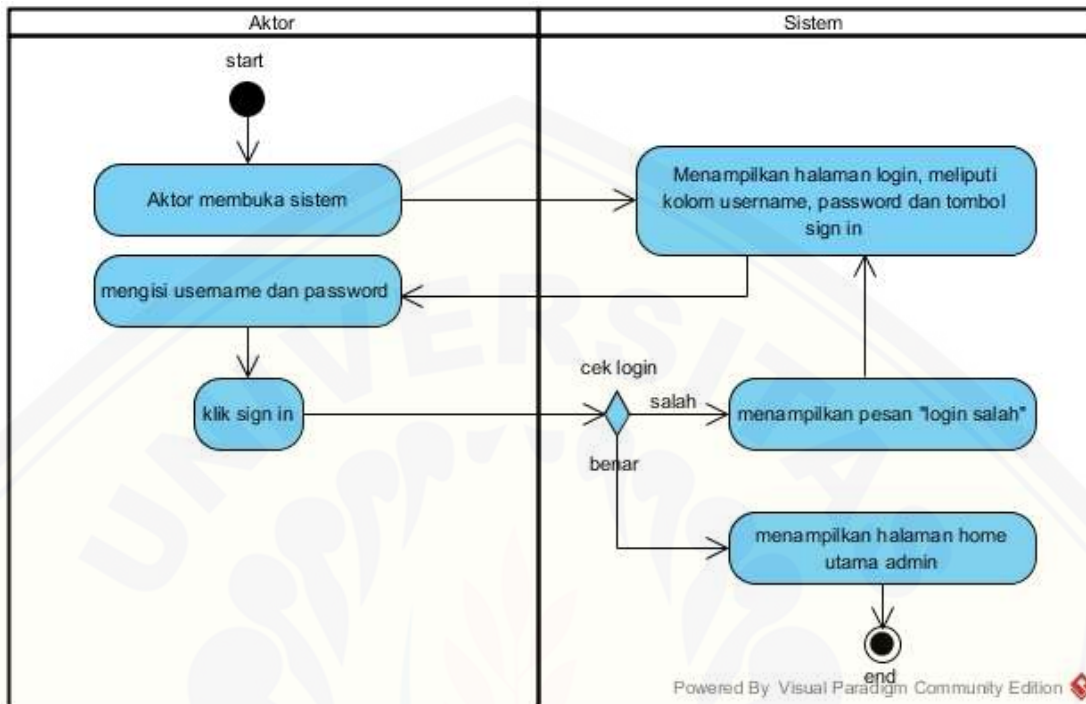
LAMPIRAN C

C.1 Activity Diagram Login

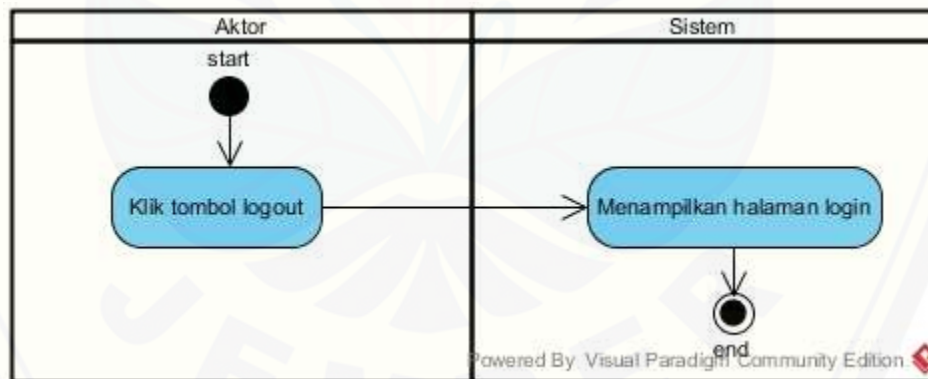
Gambar C.1 Activity Diagram Login (Admin)



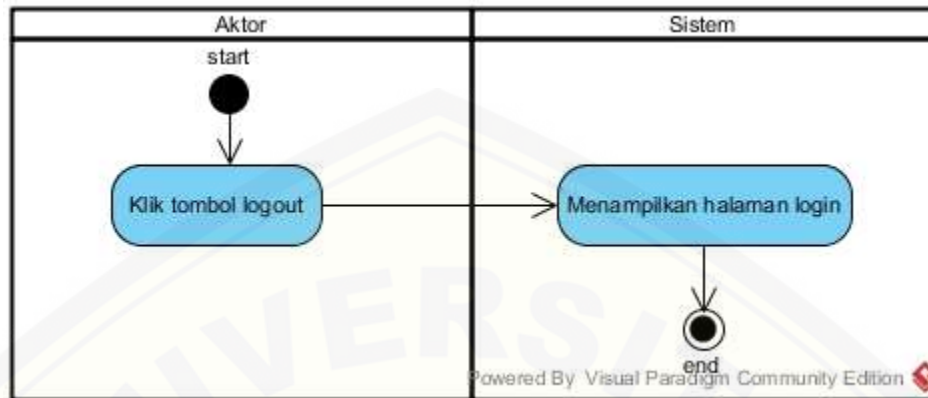
Gambar C.2 Activity Diagram Login (Siswa)



Gambar C.3 Activity Diagram Logout (Admin)

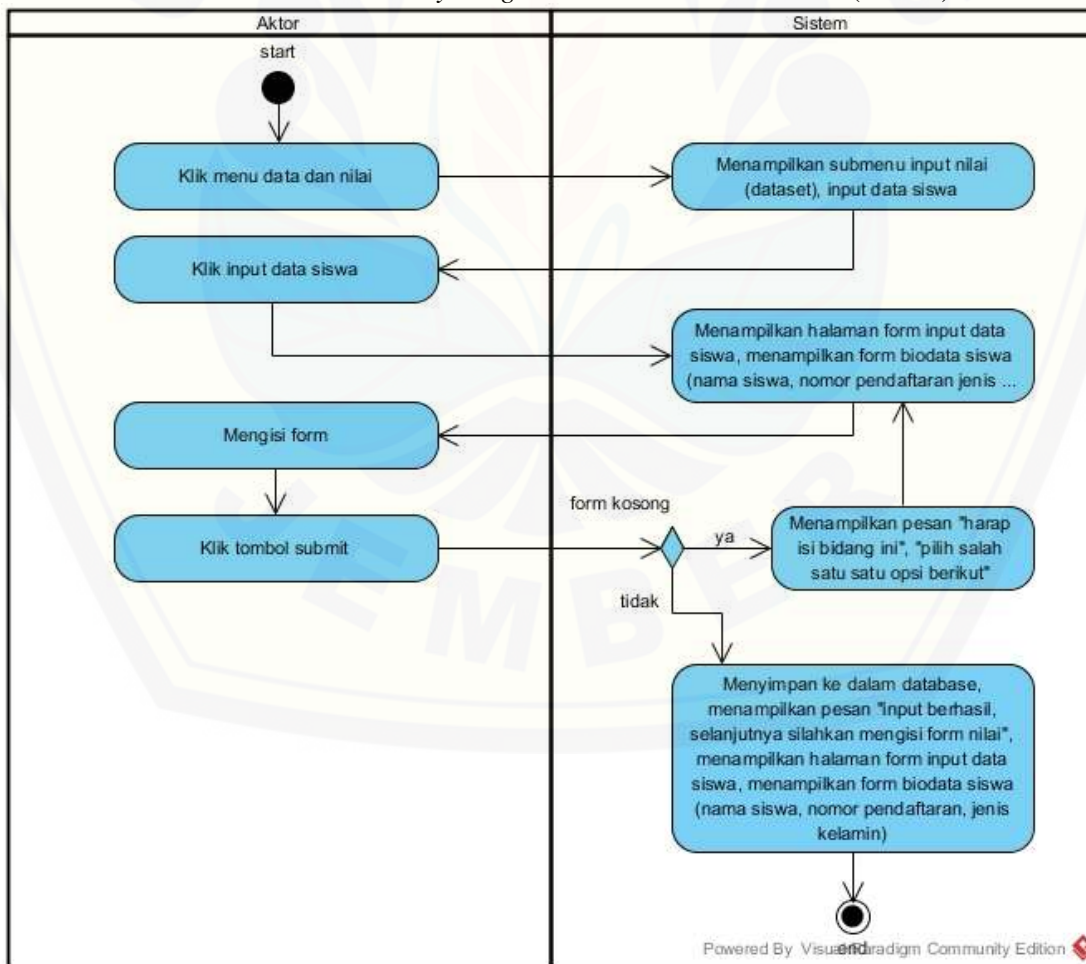


Gambar C.4 Activity Diagram Logout (Siswa)

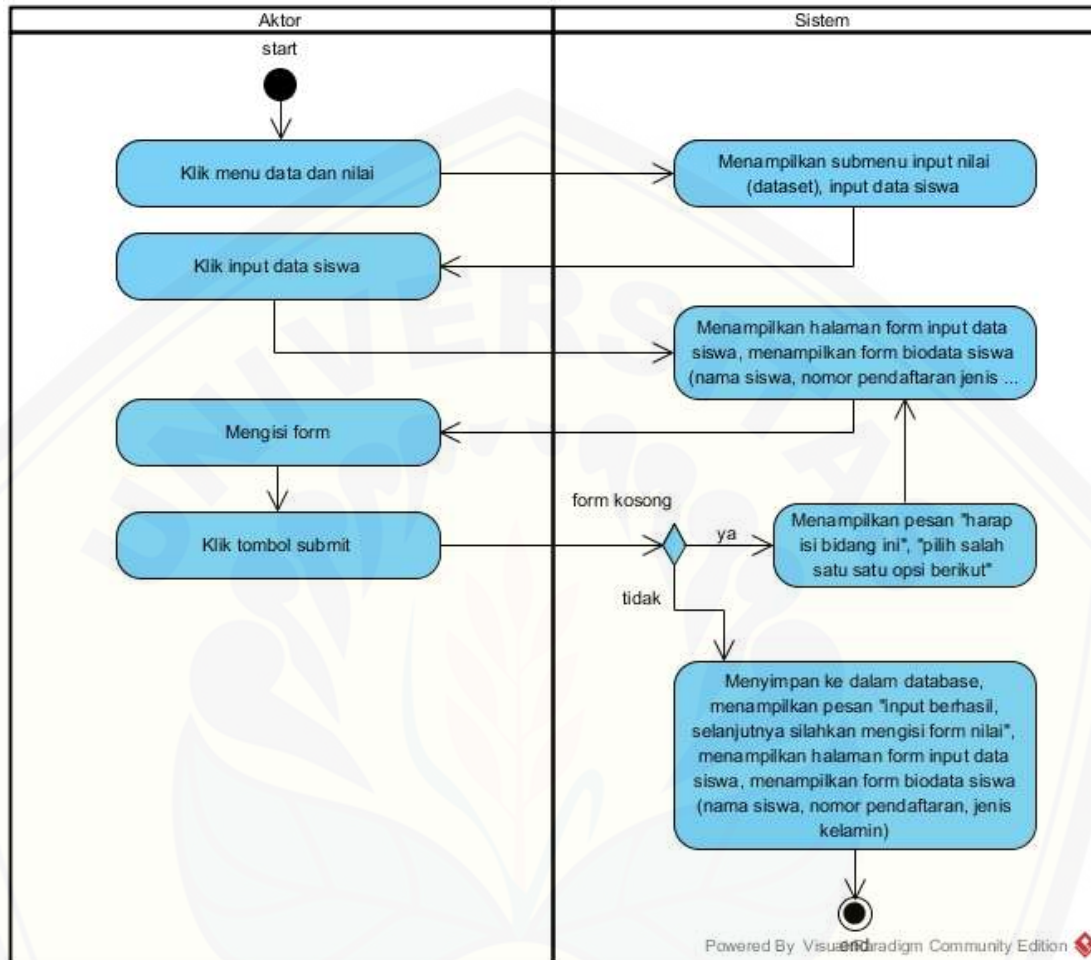


C.2 Activity Diagram Menambah Data Siswa

Gambar C.5 Activity Diagram Menambah Data Siswa (Admin)

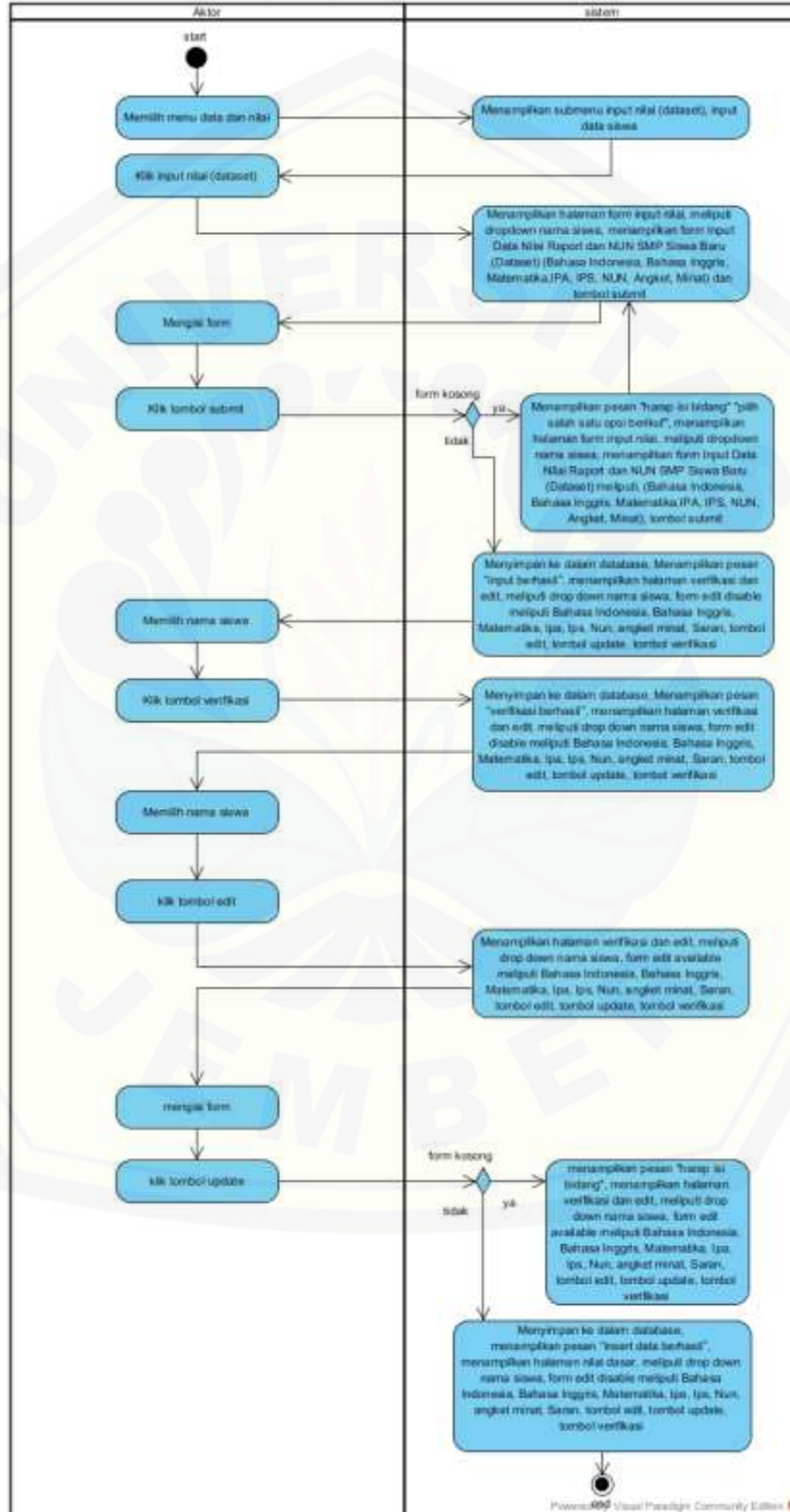


Gambar C.6 Activity Diagram Menambah Data Siswa (Siswa)

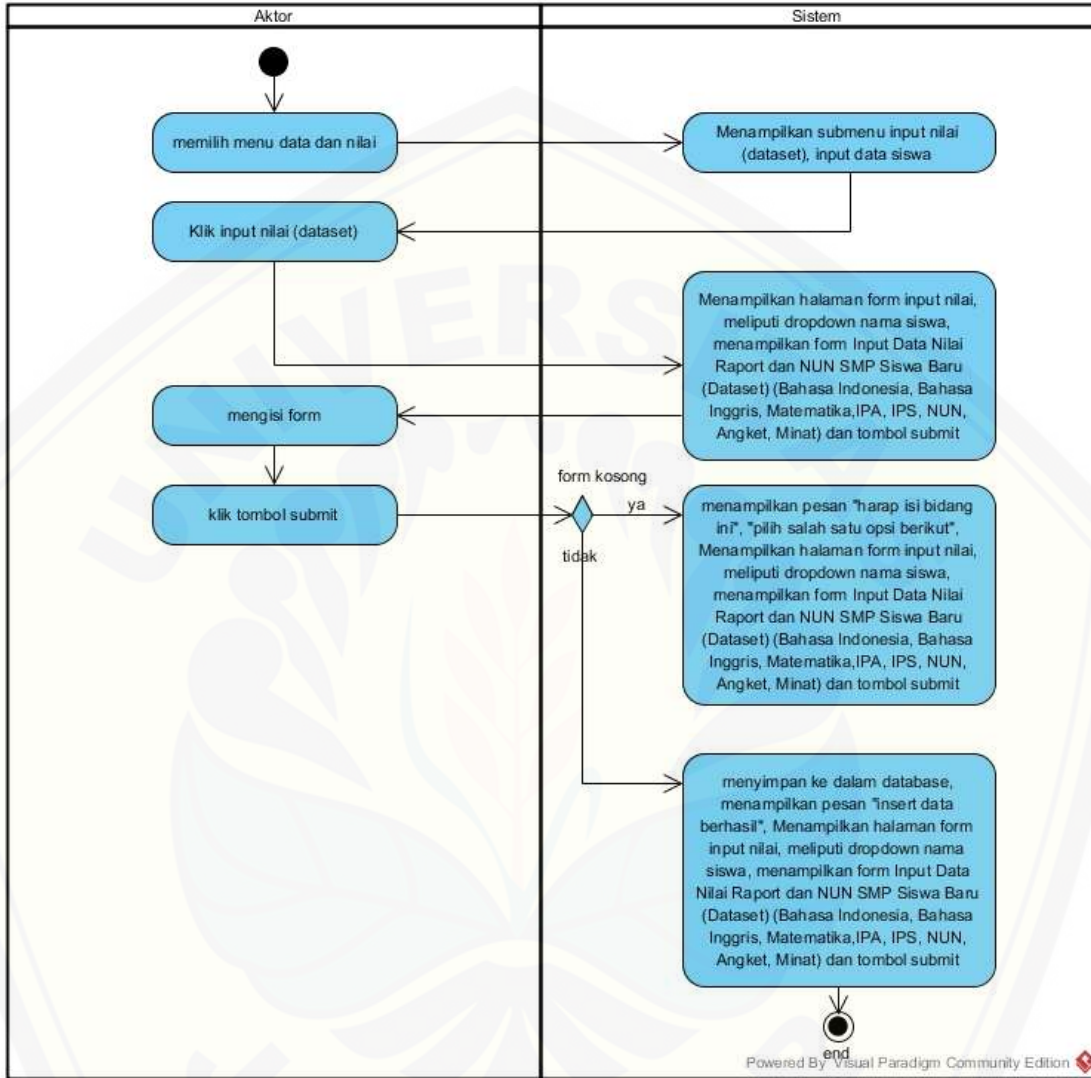


C.3 Activity Diagram Menambah Nilai Siswa

Gambar C.7 Activity Diagram Menambah Nilai Siswa (Admin)

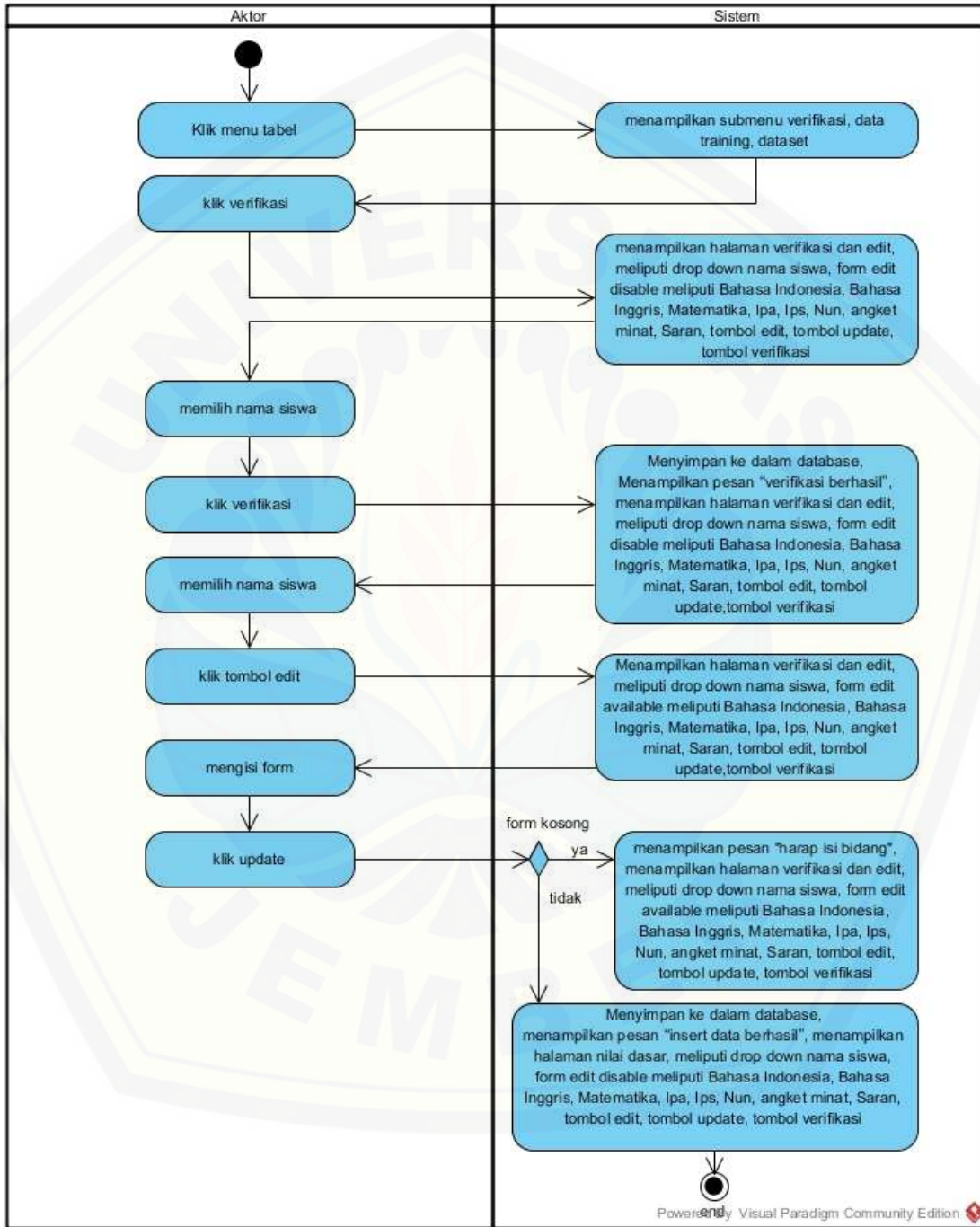


Gambar C.8 Activity Diagram Menambah Nilai Siswa (Siswa)



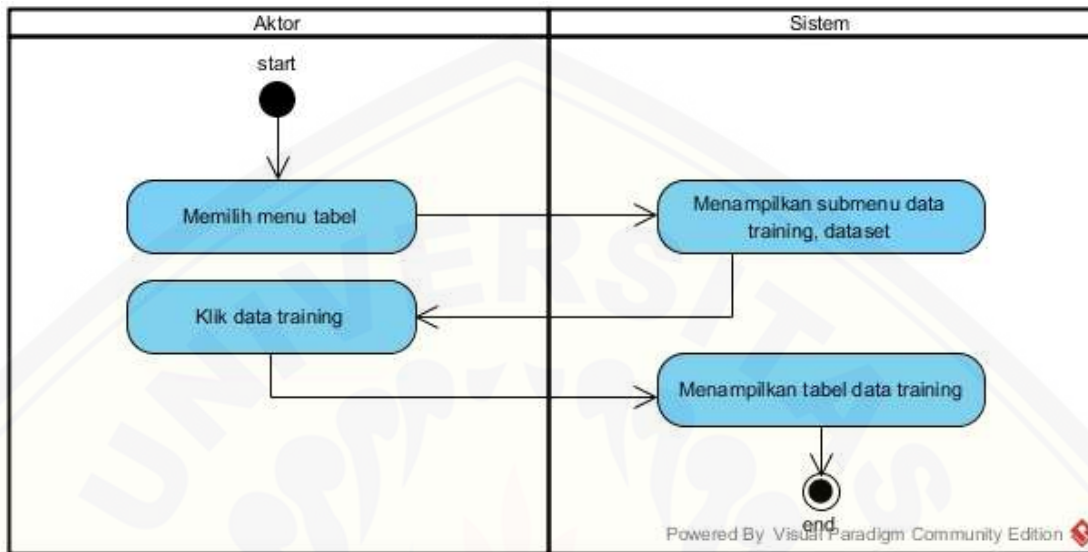
C.4 Activity Diagram Verifikasi

Gambar C.9 Activity Diagram Verifikasi (Admin)



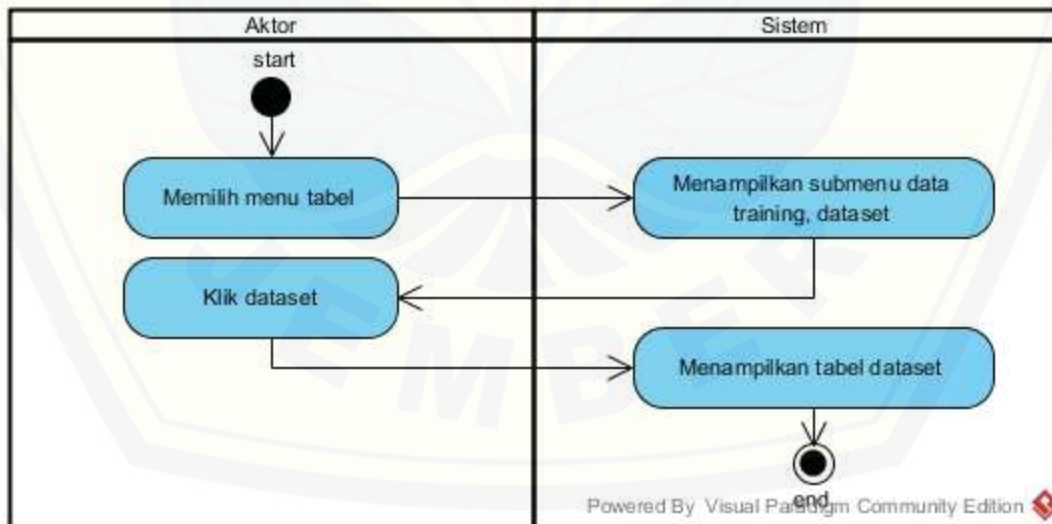
C.5 Activity Diagram Melihat Data Training

Gambar C.10 Activity Diagram Melihat Data Training (Admin)



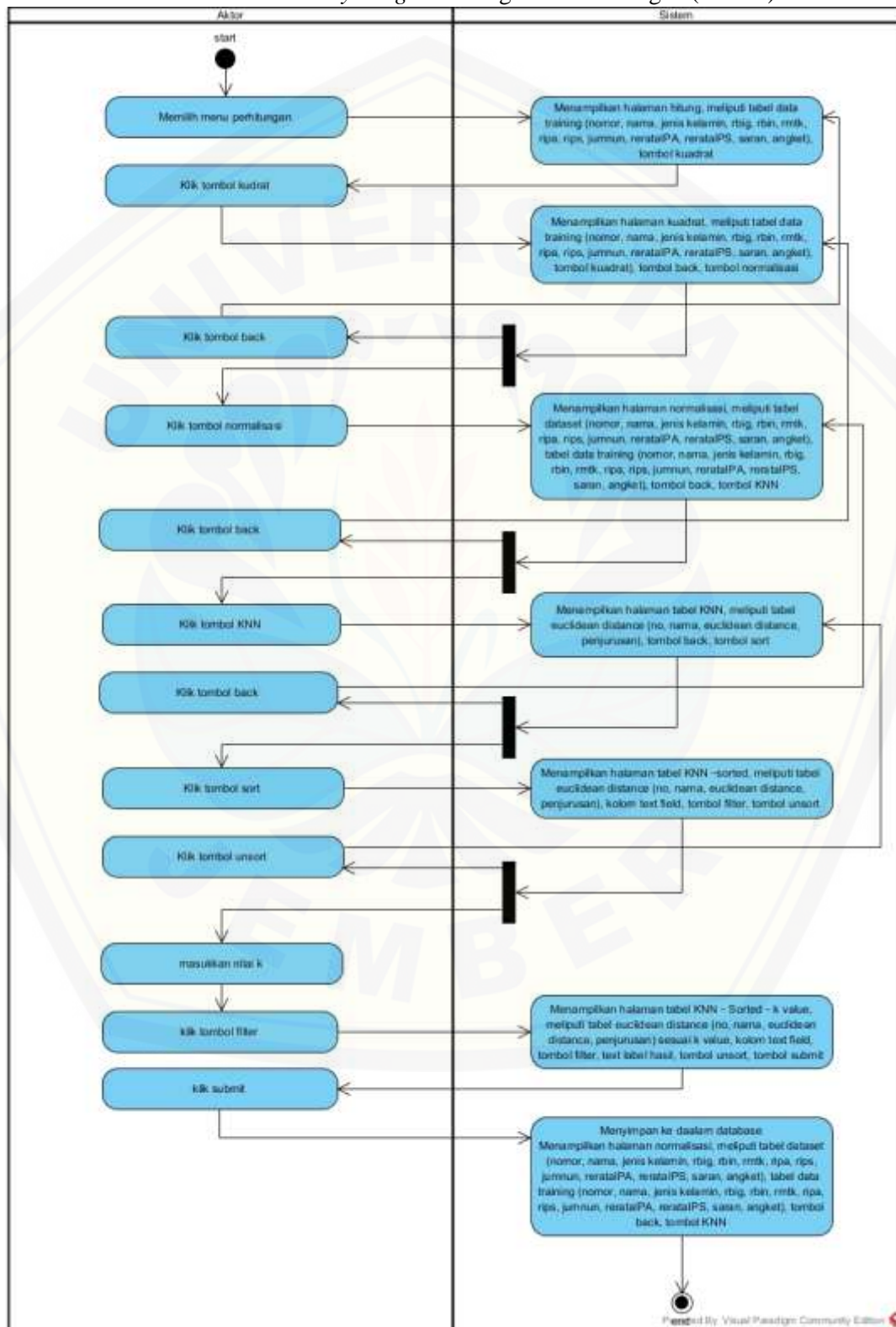
C.5 Activity Diagram Melihat Dataset

Gambar C.11 Activity Diagram Melihat Dataset (Admin)



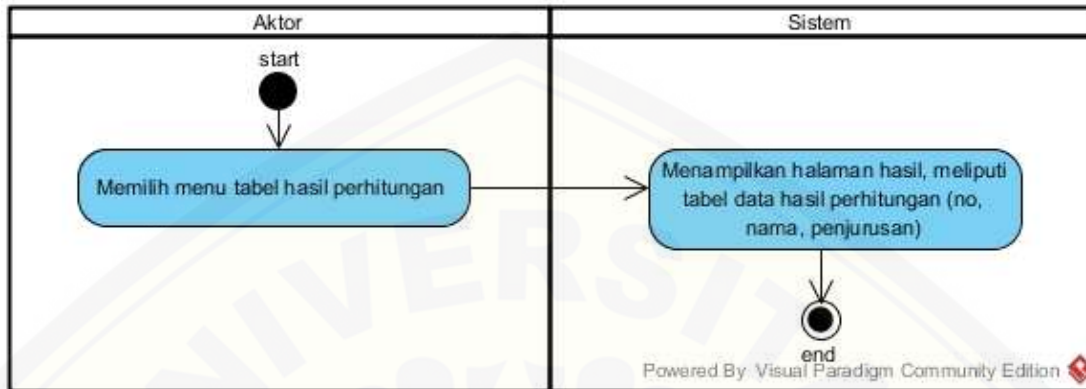
7 Activity Diagram Mengelola Perhitungan

Gambar C.12 Activity Diagram Mengelola Perhitungan (Admin)



C.7 Activity Diagram Melihat Hasil Perhitungan

Gambar C.13 Activity Diagram Melihat Hasil Perhitungan (Admin)



Gambar C.14 Activity Diagram Melihat Hasil Perhitungan (Admin)

