

**SISTEM INFORMASI AKADEMIK
SMPN 3 SETELUK**

SKRIPSI

Diajukan untuk Memenuhi Salah Satu Syarat
dalam Menempuh Program Pendidikan Strata I
pada Jurusan Teknik Informatika
di ITN Malang

Oleh :

DELIS KURNIAWAN SAMANTHA
N I M : 0818078



**INSTITUT TEKNOLOGI NASIONAL MALANG
FAKULTAS TEKNOLOGI INDUSTRI
JURUSAN TEKNIK INFORMATIKA S1**

2013

ମୁଖ୍ୟମାନ ପାତ୍ରଙ୍କ ଅଧ୍ୟକ୍ଷ

ମୁଖ୍ୟ ୩ ମହୀ

ବେଶ୍ୱର

ମୁଖ୍ୟ ୩ ମହୀ ଲିଖାନକ ମୁଖ୍ୟ ଅଧ୍ୟକ୍ଷ
ମୁଖ୍ୟ ୩ ମହୀ ମୁଖ୍ୟ ଅଧ୍ୟକ୍ଷ ଏବଂ
ମୁଖ୍ୟ ୩ ମହୀ ମୁଖ୍ୟ ଅଧ୍ୟକ୍ଷ ଏବଂ
ମୁଖ୍ୟ ୩ ମହୀ ମୁଖ୍ୟ ଅଧ୍ୟକ୍ଷ

୧୦୦

ମୁଖ୍ୟ ୩ ମହୀ ମୁଖ୍ୟ ଅଧ୍ୟକ୍ଷ

ମୁଖ୍ୟ ୩ ମହୀ ମୁଖ୍ୟ ଅଧ୍ୟକ୍ଷ

ମୁଖ୍ୟ ୩ ମହୀ ମୁଖ୍ୟ ଅଧ୍ୟକ୍ଷ

୧୦୦

**SISTEM INFORMASI AKADEMIK
SMPN 3 SETELUK**

SKRIPSI

**Diajukan untuk Memenuhi Salah Satu Syarat
dalam Menempuh Program Pendidikan Strata I
pada Jurusan Teknik Informatika
di ITN Malang**

Oleh :

**DELIS KURNIAWAN SAMANTHA
N I M : 0818078**



**INSTITUT TEKNOLOGI NASIONAL MALANG
FAKULTAS TEKNOLOGI INDUSTRI
JURUSAN TEKNIK INFORMATIKA S1
2013**

LEMBAR PERSETUJUAN

SISTEM INFORMASI AKADEMIK
SMPN 3 SETELUK

SKRIPSI

*Disusun dan Diajukan Sebagai Salah Satu Syarat Untuk Memperoleh
Gelar Sarjana Teknik Informatika Strata Satu (S-1)*

Disusun Oleh :

DELIS KURNIAWAN S.

NIM : 08.18.078

Dosen Pembimbing I

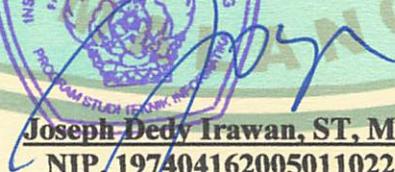

Dr. Ir. Dhayal Gustopo, MT
NIP. 0713086103

Dosen Pembimbing II


Ahmad Raisol, ST,
NIP. 1031000431

Diperiksa dan Disetujui


Ketua Prodi Teknik Informatika S-1


Joseph Dedy Irawan, ST, MT
NIP. 197404162005011022

PROGRAM STUDI TEKNIK INFORMATIKA S-1

FAKULTAS TEKNOLOGI INDUSTRI

INSTITUT TEKNOLOGI NASIONAL MALANG

2013



INSTITUT TEKNOLOGI NASIONAL MALANG
Fakultas Teknologi Industri
Program Studi Teknik Informatika S1
Jl. Raya Karanglo Km. 2 Malang

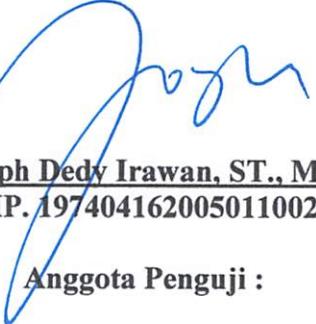
**BERITA ACARA UJIAN SKRIPSI
FAKULTAS TEKNOLOGI INDUSTRI**

Nama : Delis Kurniawan Samantha
NIM : 08.18.078
Jurusan : Teknik Informatika S-1
Judul : Sistem Informasi Akademik SMPN 3 Seteluk

Dipertahankan dihadapan Majelis Penguji Skripsi Jenjang Strata Satu (S-1) pada :
Hari : Jumat
Tanggal : 15 Februari 2013
Nilai : 78,72 (B+)

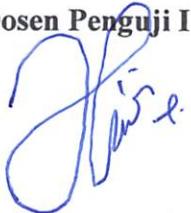
Panitia Ujian Skripsi :

Ketua Majelis Penguji


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

Anggota Penguji :

Dosen Penguji I



Ali Mahmudi, BEng, PhD.
NIP. 1031000429

Dosen Penguji II



Sandy Nataly Mantja, Skom.
NIP. 1030800418



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

PERNYATAAN KEASLIAN SKRIPSI

Saya yang bertanda tangan di bawah ini:

Nama : Delis Kurniawan Samantha
Nim : 08.18.078
Program Studi : Teknik Informatika S-1
Fakultas : Teknologi Industri

Menyatakan dengan sesungguhnya bahwa Skripsi saya yang berjudul:

“SISTEM INFORMASI AKADEMIK SMPN 3 SETELUK”

Adalah Skripsi saya sendiri bukan duplikat serta mengutip atau menyadur seluruhnya karya orang lain kecuali dari sumber aslinya.

Malang, 01 Maret 2013

Yang membuat pernyataan



Delis Kurniawan S

**SISTEM INFORMASI AKADEMIK
SMPN 3 SETELUK**

**DELIS KURNIAWAN S
(0818078)**

Program Studi Teknik Informatika S-1
Fakultas Teknologi Industri
Institut Teknologi Nasional Malang
Email: deliskurniawansamantha@gmail.com

**Dosen Pembimbing : 1. Dr. Ir. Dhayal Gustopo, MT
2. Ahmad Faisol, ST,**

Abstrak

Kegiatan akademik di SMPN 3 Seteluk merupakan hal yang rutin diantaranya adalah proses penilaian siswa. Masalah yang sering terjadi adalah tak tepat waktunya jadwal pelaporan hasil prestasi di akademik misalnya hasil belajar (raport) siswa, kinerja guru, kinerja karyawan yang tak tepat waktu. Dari permasalahan tersebut, maka dibuat suatu sistem akademik sekolah untuk mengatur proses penilaian siswa, dimana nantinya data nilai akan diproses melalui program DBMS sehingga data dapat lebih cepat diproses dan laporan dapat dibuat dengan mudah. Keuntungan dari program ini pun data nilai dapat tersimpan dengan rapih. Metode dalam penelitian yang digunakan yaitu waterfall model dengan perangkat desain menggunakan UML (Unified Modeling Language). Adapun tools yang digunakan untuk membangun sistem ini memakai PhP sebagai bahasa pemrograman web serta MySQL sebagai database servernya.

Sistem ini diterapkan dalam proses penilaian siswa, sehingga data nilai siswa dapat terintegrasi dengan baik. Selain itu proses pengolahan nilai pun dapat lebih akurat dalam bentuk presentase.

Kata kunci : Akademik Sekolah, UML, PhP, MySQL

KATA PENGANTAR

Puji syukur penulis panjatkan kepada Tuhan Yang Maha Esa atas limpahan anugrah-Nya, sehingga dapat menyelesaikan skripsi yang berjudul “**SISTEM INFORMASI AKADEMIK SMPN 3 SETELUK**”.

Adapun maksud dan tujuan dari penulisan laporan ini merupakan salah satu syarat untuk dapat menyelesaikan studi dan mendapatkan gelar Sarjana Jurusan Teknik Informatika S-1 ITN Malang.

Sebagai pihak penyusun, kami menyadari tanpa adanya kemauan dan usaha serta bantuan dari berbagai pihak, maka laporan ini tidak dapat diselesaikan dengan baik. Oleh karena itu, kami mengucapkan Terimakasih kepada yang terhormat :

1. Ir. Soeparno Djijo, MT., selaku Rektor Institut Teknologi Nasional Malang.
2. Ir. H. Anang Subardi, MT., selaku Dekan Fakultas Teknologi Industri Institut Teknologi Nasional Malang.
3. Joseph Dedy Irawan, ST, MT., selaku Ketua Jurusan Teknik Informatika S-1 Institut Teknologi Nasional Malang.
4. Dr. Ir. Dhayal Gustopo, MT., selaku Dosen Pembimbing I.
5. Ahmad Faisol, ST., selaku Dosen Pembimbing II.
6. Kepada Orang tua yang tak pernah berhenti memberikan doa serta dukungannya.
7. Teman – teman Aslab Database dan Sistem Informasi yang telah memberi dukungan.
8. Semua pihak yang telah membantu dalam menyelesaikan skripsi ini yang tidak dapat penulis sebutkan satu persatu.

Penulis menyadari bahwa skripsi ini masih mempunyai banyak kekurangan dan kesalahan, untuk itu penulis mengharapkan adanya saran yang membangun demi pengembangan ilmu pengetahuan. Akhir kata, penulis mohon maaf kepada semua pihak apabila selama penyusunan skripsi ini penyusun membuat kesalahan dan semoga skripsi ini dapat bermanfaat bagi kita semua.

Malang, Maret 2013

Penulis

DAFTAR ISI

LEMBAR PERSETUJUAN	i
LEMBAR BERITA ACARA	ii
SURAT PERNYATAAN ORISINALITAS	iii
<i>Abstrak</i>	iv
KATA PENGANTAR	v
DAFTAR ISI	vii
DAFTAR TABEL	x
DAFTAR GAMBAR	xi
BAB I PENDAHULUAN	1
1.1. Latar Belakang	1
1.2. Rumusan Masalah	2
1.3. Batasan Masalah	2
1.4. Tujuan	3
1.5. Metode Penelitian.....	3
1.6. Sistematika Penulisan	3
BAB II LANDASAN TEORI	5
2.1. Manajemen Sekolah	5
2.1.1. Kebijakan Mutu SMPN 3 Seteluk	5
2.1.2. Struktur Organisasi	5
2.1.3. Uraian Kerja	6
2.1.3.1. Kepala Sekolah	6
2.1.3.2. Komite	6
2.1.3.3. Tata Usaha	6
2.1.3.4. Guru Kelas	7
2.1.3.5. Guru Bidang	7
2.2. Internet	7
2.3. Sistem	8
2.3.1. Pengertian Sistem	8
2.3.2. Pengertian Sistem Informasi	8
2.3.3. Pengertian Sistem Informasi Akademik.....	9

2.4. Hypertext Markup Language (HTML)	9
2.5. Hypertext Transfer Protocol (HTTP)	9
2.5.1. Hypertext Transfer Protocol Secure (HTTPS)	10
2.6. Database	11
2.7. Tinjauan Perangkat Lunak	11
2.7.1. PHP (PHP: Hypertext Preprocessor)	12
2.7.2. MySQL	12
2.8. Data Flow Diagram (DFD)	14
2.8.1. Diagram Konteks (Context Diagram)	17
2.8.2. Diagram Nol (0)	17
BAB III METODOLOGI	18
3.1. Gambaran Umum	18
3.2. Kebutuhan Sistem	18
3.2.1. Deskripsi Kebutuhan	19
3.3. Pemodelan Kebutuhan Fungsional	20
3.3.1. Use Case Diagram	21
3.4. Skenario	22
3.4.1. Skenario Usecase Diagram Tata Usaha	22
3.4.2. Skenario Usecase Diagram Guru	27
3.4.3. Skenario Usecase Diagram Siswa	30
3.5. Perancangan Sistem	31
3.5.1. Konteks Diagram	32
3.5.2. Data Flow Diagram (DFD)	32
3.5.3. Data Flow Diagram Proses 8 Level 1	33
3.6. Entity Relational Diagram (ERD)	34
3.7. Perancangan Basis Data	35
3.8. Struktur Menu	39
3.9. Antarmuka Pemakai	39
BAB IV IMPLEMENTASI DAN PENGUJIAN	43
4.1. Implementasi Sistem	43
4.1.1. Spesifikasi Perangkat Keras	43
4.1.2. Spesifikasi Perangkat Lunak	43

4.2. Implementasi Program	44
4.2.1. Halaman Login Siakad	44
4.2.2. Halaman Admin	47
4.2.3. Halaman Data Siswa	47
4.2.4. Halaman Data Wali Kelas	51
4.2.5. Halaman Data Kelas	55
4.2.6. Halaman Data Mata Pelajaran	59
4.2.7. Halaman Data Matpel-Kelas	63
4.2.8. Halaman Data Tahun Ajaran	67
4.2.9. Halaman identitas Siswa	71
4.2.10. Halaman Prestasi Siswa	72
4.3. Pengujian Sistem	73
4.3.1 Rencana Pengujian	74
4.3.2 Pengujian Login	74
4.3.3 Pengujian Input Data Guru	75
4.3.4 Pengujian Input Data Siswa	75
4.3.5 Pengujian Nilai Siswa	76
4.3.6 Kesimpulan Hasil Pengujian	77
BAB V PENUTUP	78
5.1. Kesimpulan	78
5.2. Saran	78
DAFTAR PUSTAKA	79
LEMBAR PERSEMBAHAN	80
LAMPIRAN	

DAFTAR TABEL

Tabel 3.1. Kebutuhan Fungsional Perangkat Lunak Tata Usaha	19
Tabel 3.2. Kebutuhan Fungsional Perangkat Lunak Guru	20
Tabel 3.3. Kebutuhan Fungsional Perangkat Lunak Siswa	20
Tabel 3.4. Skenario Usecase Diagram Tata Usaha	22
Tabel 3.5. Skenario Usecase Diagram Guru	27
Tabel 3.6. Skenario Usecase Diagram Siswa	30
Tabel 3.7. Rancangan Database Siswa	35
Tabel 3.8. Rancangan Database Guru	35
Tabel 3.9. Rancangan Database Tahun Ajaran	36
Tabel 3.10. Rancangan Database Kelas	37
Tabel 3.11. Rancangan Database Mata Pelajaran	37
Tabel 3.12. Rancangan Database Nilai	37
Tabel 3.13. Rancangan Database Wali Kelas	38
Tabel 3.14. Rancangan Database Mata Pelajaran Kelas	38
Tabel 4.1 Rencana Pengujian	74
Tabel 4.2 Pengujian Login Admin	74
Tabel 4.3 Pengujian Input Data Guru	75
Tabel 4.4 Pengujian Input Data Siswa	76
Tabel 4.5 Pengujian Nilai Siswa	76

DAFTAR GAMBAR

Gambar 2.1.	Diagram Organisasi SMPN 3 Seteluk	5
Gambar 2.2.	Simbol Entitas	14
Gambar 2.3.	Simbol Proses	15
Gambar 2.4.	Simbol Data Flow	16
Gambar 2.5.	Simbol Data Store	17
Gambar 3.1.	Gambaran Umum	18
Gambar 3.2.	Usecase Diagram Tata Usaha	21
Gambar 3.3.	Usecase Diagram Guru	21
Gambar 3.4.	Usecase Diagram Siswa	22
Gambar 3.5.	Diagram Konteks	32
Gambar 3.6.	Data Flow Diagram	33
Gambar 3.7.	DFD Level 1 Proses 8	34
Gambar 3.8.	Entity Relational Diagram	34
Gambar 3.9.	Struktur Menu	39
Gambar 3.10.	Tata Letak Halaman Login	40
Gambar 3.11.	Tata Letak Halaman Daftar Siswa	40
Gambar 3.12.	Tata Letak Halaman Guru	41
Gambar 3.13.	Tata Letak Halaman Daftar Kelas	41
Gambar 3.14.	Tata Letak Halaman Daftar Nilai	41
Gambar 3.15.	Tata Letak Halaman Rubah Siswa	42
Gambar 3.16.	Tata Letak Halaman Rubah Guru	42
Gambar 3.17.	Tata letak Halamn Rubah Nilai	42
Gambar 4.1.	Halaman Login Siakad	45
Gambar 4.2.	Halaman Menu Utama Saat Login berhasil	46
Gambar 4.3.	Halaman Peringatan Bila Password Anda Salah	46
Gambar 4.4.	Data Guru	47
Gambar 4.5.	Halaman Data Siswa	48

Gambar 4.6. Halaman Aksi Insert Data Siswa	49
Gambar 4.7. Halaman Aksi Update Data Siswa	50
Gambar 4.8. Halaman Aksi Delete Data Siswa	51
Gambar 4.9. Halaman Data Wali Kelas	52
Gambar 4.10. Halaman Aksi Insert Data Wali Kelas	53
Gambar 4.11. Halaman Aksi Update Data Wali Kelas	54
Gambar 4.12. Halaman Aksi Delete Data Wali Kelas	55
Gambar 4.13. Halaman Data Kelas	56
Gambar 4.14. Halaman Aksi Insert Data Kelas	57
Gambar 4.15. Halaman Aksi Update Data Kelas	58
Gambar 4.16. Halaman Aksi Delete Data Kelas	59
Gambar 4.17. Halaman Data Mata Pelajaran	60
Gambar 4.18. Halaman Aksi Insert Data Mata Pelajaran	61
Gambar 4.19. Halaman Aksi Update Data Mata Pelajaran	62
Gambar 4.20. Halaman Aksi Delete Data Mata Pelajaran	63
Gambar 4.21. Halaman Data Matpel-Kelas	64
Gambar 4.22. Halaman Aksi Insert Data Matpel-Kelas	65
Gambar 4.23. Halaman Aksi Update Data Matpel-Kelas	66
Gambar 4.24. Halaman Aksi Delete Data Matpel-Kelas	67
Gambar 4.25. Halaman Data Tahun Ajaran	68
Gambar 4.26. Halaman Aksi Insert Data Tahun Ajaran	69
Gambar 4.27. Halaman Aksi Update Data Tahun Ajaran	70
Gambar 4.28. Halaman Aksi Delete Data Tahun Ajaran	71
Gambar 4.29. Halaman Identitas Siswa	72
Gambar 4.30. Halaman Prestasi siswa	73

BAB I

PENDAHULUAN

1.1 Latar Belakang

Dengan semakin berkembangnya era globalisasi, maka kebutuhan akan informasi pun semakin penting termasuk pula bagi mereka yang menekuni dunia pendidikan. Maka apabila ada ketidak lancaran pengelolaan dapat mengakibatkan kekacauan dalam pengendalian arus informasi, pengambilan keputusan serta perencanaan teknis dan strategis.

Menurut John F.Nash dan Martil B.Robert (1988) yang diterjemahkan oleh Jogiyanto H.M, menyatakan bahwa Sistem informasi adalah suatu sistem dalam Suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang mendukung fungsi operasi organisasi yang bersifat manajerial dengan kegiatan strategi dari suatu organisasi untuk dapat menyediakan kepada pihak luar tertentu dengan informasi yang diperlukan untuk pengambilan keputusan. Sistem informasi dalam suatu organisasi dapat dikatakan sebagai suatu sistem yang menyediakan informasi bagi semua tingkatan dalam organisasi tersebut kapan saja diperlukan. Sistem ini menyimpan, mengambil, mengubah, mengolah dan mengkomunikasikan informasi yang diterima dengan menggunakan sistem informasi atau peralatan sistem lainnya.

Pentingnya implementasi sistem informasi menurut statistik dinas pendidikan nasional (DIKNAS) suatu model solusi untuk meningkatkan kualitas pendidikan, apapun interpretasi kualitas pendidikan itu yaitu input, proses, ataupun output pendidikan. Dengan diterapkannya Sistem informasi akademik Sekolah, diharapkan terwujudnya transformasi dari pola pendidikan lama menuju pola pendidikan baru. Pola pendidikan lama yang masih menggunakan sistem secara manual dalam melakukan proses penyimpanan data dan nilai siswa sedangkan pola pendidikan baru yang menggunakan sistem terkomputerisasi sangat efektif dan aman dalam melakukan proses penyimpanan data dan nilai siswa, dan mempermudahkan siswa dalam mendapatkan tentang informasi nilai dan materi pelajaran. Sistem Informasi Akademik (SIAKAD) ini diharapkan

sekolah dapat memperoleh manfaat terwujudnya suatu pusat informasi yang dapat diakses oleh siswa, pegawai maupun guru untuk memperoleh data-data pendidikan yang terkait dengan porsi dan kewenangannya.

Dalam upaya untuk memantau (monitoring) pelayanan SMPN 3 Seteluk Kabupaten Sumbawa Barat - NTB, sistem yang kita sebut adalah sistem informasi akademik. Dimana sistem ini sangat dibutuhkan untuk memonitor semua hasil simpan basis data proses belajar mengajar di SMPN 3 Seteluk khususnya oleh pihak internal dan eksternal yaitu guru, pegawai dan siswa. dengan beragam tujuan yang diperlukan oleh guru untuk mempermudah dalam memberikan informasi nilai dan materi kepada siswa, pegawai untuk mempermudah dalam melakukan proses pengelolahan database siswa dan sekolah secara terkomputerisasi, dan siswa dalam mendapatkan suatu informasi nilai dan materi pelajaran yang benar dan akurat. beragam permasalahan yang harus segera diselesaikan pada setiap kinerja dari sirkulasi tersebut harus dibantu dengan sistem komputerisasi yang dapat mempermudah alur kerja yang sudah berjalan.

Oleh karena itu penulis bermaksud mengangkat masalah tersebut guna sebagai laporan penelitian dengan judul : “ **SISTEM INFORMASI AKADEMIK SMPN 3 SETELUK** ”.

1.2 Rumusan Masalah

Berdasarkan latar belakang diatas, penulis merumuskan masalah dalam hal penelitian ini adalah “Bagaimana merancang sebuah aplikasi sistem informasi akademik SMPN 3 Seteluk untuk pengelolaan data dengan menggunakan cara terkomputerisasi”.

1.3 Batasan Masalah

Dikarenakan sistem yang digunakan hingga saat ini pada SMPN 3 Seteluk adalah bersifat manual. Maka dalam hal memusatkan masalah yang ada dan agar tidak menyimpang dari pokok bahasan maka bentuk perancangan sistem penulis hanya dibatasi pada prosedur:

1. Pembuatan suatu aplikasi sistem informasi akademik yang berbasis *web* hanya untuk jaringan internal sekolah.
2. Pembuatan *database server* MySQL untuk mendukung sistem informasi dan menggunakan bahasa pemrograman PHP.
3. Sistem ini mencakup menampilkan data nilai, mengelola data nilai, menampilkan daftar siswa dan mengelola data siswa.

1.4 Tujuan

Tujuan dari penelitian ini adalah sebagai berikut :

1. Membuat aplikasi Sistem Informasi Akademik SMPN 3 Seteluk.
2. Bagaimana menyediakan informasi yang berkualitas bagi manajemen dari SMPN 3 Seteluk.

1.5 Metode Penelitian

1. *Survey* lapangan, yaitu melihat dan mengamati secara langsung proses pengolahan data yang ada.
2. Wawancara, yaitu dengan bertanya tentang sesuatu hal kepada yang berkepentingan.
3. Studi pustaka, yaitu membaca buku – buku yang terkait secara langsung maupun tidak langsung untuk mengetahui secara teoritis permasalahan yang sedang dihadapi.

1.6 Sistematika Penulisan

Untuk mengetahui dan memahami gambaran secara umum mengenai isi dari Tugas Akhir ini, maka penulisan Tugas Akhir ini dibagi dalam lima bab dengan sistematika penulisan sebagai berikut :

Bab I Pendahuluan

Bab ini berisikan tentang latar belakang, rumusan masalah, batasan masalah, tujuan, metode penelitian, serta sistematika penulisan.

Bab II Landasan Teori

Bab ini menjelaskan tentang landasan teori yang digunakan, seperti penjelasan mengenai : Manajemen sekolah, internet, sistem, HTML, HTTP, HTTPS, PHP, MySQL, basis data.

Bab III Analisis Sistem

Bab ini berisikan Gambaran umum, kebutuhan sistem, perancangan sistem mencakup diagram konteks dan data flow diagram (DFD), perancangan basis data.

Bab IV Implementasi dan Pengujian

Bab ini berisi lingkungan pendukung pembuatan website media informasi, pembuatan database dan tabel, Implementasi web.

Bab V Kesimpulan dan Saran

Bab ini akan diberikan penjelasan mengenai kesimpulan yang diperoleh dari hasil pembahasan yang berkaitan dengan judul Tugas Akhir serta saran yang diharapkan dapat memberikan kemajuan dan perbaikan bagi lembaga.

BAB II

LANDASAN TEORI

2.1 Manajemen Sekolah

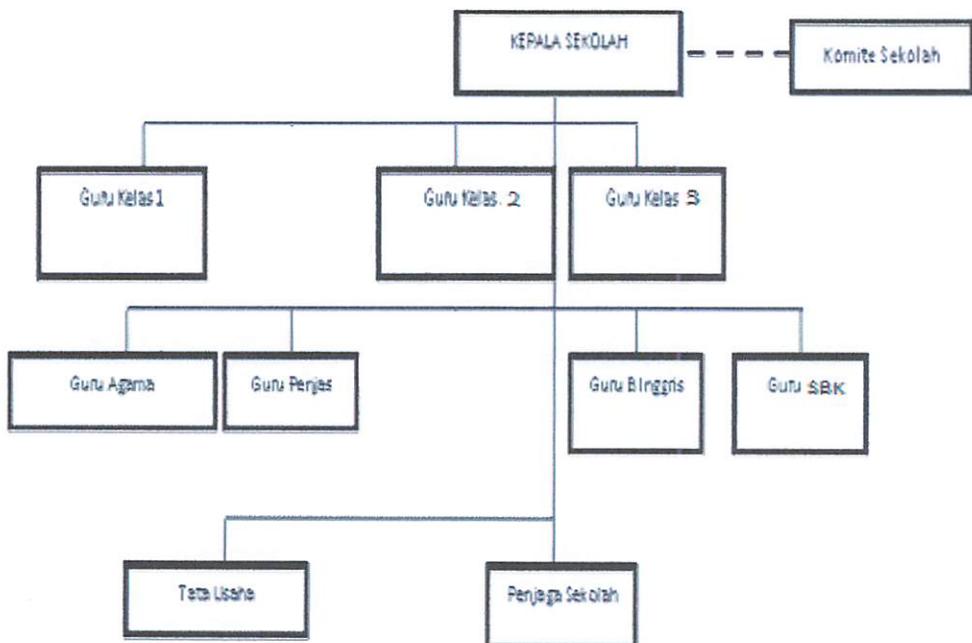
2.1.1 Kebijakan Mutu SMPN 3 Seteluk

SMPN 3 Seteluk Kabupaten Sumbawa Barat – NTB berupaya mewujudkan sekolah bertaraf internasional yang dilandasi Imtaq, Iptek, dan Berwawasan lingkungan dengan :

1. Terwujudnya aspek Input yang bertaraf nasional.
2. Terwujudnya aspek proses yang bertaraf nasional.
3. Terwujudnya semua aspek Output yang bertaraf internasional yang beriman.

2.1.2 Struktur Organisasi

SMPN 3 Seteluk Kabupaten Sumbawa Barat - NTB didalam pelaksanaan tugas-tugasnya mempunyai struktur kerja sebagai berikut :



Gambar 2.1 Diagram Organisasi SMPN 3 Seteluk

2.1.3 Uraian Kerja

Adapun uraian kerja dari tiap bagian pada struktur organisasi diatas adalah sebagai berikut :

2.1.3.1 Kepala Sekolah

Kepala sekolah adalah pimpinan dari sebuah sekolah dimana segala keputusan dan perintah adalah berasal dari dirinya. Kepala Sekolah pun bersama komite pada satu waktu perlu melakukan koordinasi dalam mengambil keputusan untuk kepentingan sekolah.

Kepala sekolah juga mempunyai tugas untuk mengawasi kerja seluruh stafnya yaitu dari tata usaha dan semua guru agar dapat bekerja dengan baik.

2.1.3.2 Komite

Komite merupakan suatu kelompok orang yang diambil dari masyarakat sekitar atau bisa juga orangtua murid, yang bertugas untuk mengawasi jalannya kegiatan di sekolah. Komite pun bersama kepala sekolah dapat mengajukan keputusan tertentu untuk kepentingan bersama.

2.1.3.3 Tata Usaha

Tata usaha merupakan bagian yang mengurus segala administrasi sekolah, dimana ia membantu kepala sekolah dan seluruh unsur di sekolah termasuk guru-guru.

Dalam tugasnya ia harus mampu mengerti semua arsip dan jenis jenisnya yang ada dalam sekolah. Arsip-arsip tersebut ada yang berhubungan dengan proses belajar mengajar atau berupa laporan tentang sekolah termasuk keuangan sekolah.

2.1.3.4 Guru Kelas

Di Sekolah Dasar guru kelas dari kelas 1 sampai dengan kelas 3 bertugas untuk memberikan pengajaran kepada murid-muridnya sesuai dengan kelas yang diembannya berdasarkan kurikulum yang berlaku.

Selain tugas pokoknya sebagai pendidik bagi murid-muridnya, guru kelas pun bertanggung jawab kepada administrasi kelas misalnya mengurus nilai-nilai ujian muridnya.

2.1.3.5 Guru Bidang

Yang termasuk guru bidang, yaitu guru olahraga, guru agama, guru bahasa inggris, guru SBK. Tugasnya hampir sama dengan guru kelas yaitu mengajar murid-muridnya sesuai dengan mata pelajaran yang diambilnya. Perbedaannya ia tidak bertanggung jawab kepada administrasi kelas, ia hanya bertanggung jawab sesuai dengan mata pelajaran yang dipegangnya.

2.2 Internet

Internet (Interconnected Network) merupakan jaringan komputer yang terdiri dari ribuan jaringan komputer independen yang dihubungkan satu dengan yang lainnya. Secara etimologis, *internet* berasal dari bahasa inggris yakni *inter* berarti antar dan *net* berarti jaringan sehingga dapat diartikan hubungan antar jaringan. Jaringan komputer ini terdiri dari lembaga pendidikan, pemerintahan, militer, organisasi dan bisnis dan organisasi lainnya. *Internet* atau nama pendeknya *Net* merupakan jaringan komputer yang tersebar didunia. Sampai saat ini *internet* sudah menghubungkan lebih dari 100000 jaringan komputer dengan pemakai lebih dari 100 juta orang. *Internet* adalah jaringan luas di komputer yang lazim disebut dengan *world wide network* secara lebih ringkas *internet* adalah sumber informasi dan alatkomunikasi serta hiburan. Jaringan *internet* menyediakan beberapa aplikasi yang dapat digunakan oleh user internet : E-mail,

mailing list, newsgroup, file transfer protocol (FTP), gopher, telnet, talk, chat, *world wide web*.

World wide web atau yang biasa disebut web saja adalah sistem pengiriman dokumen tersebar keseluruh dunia. Web menggunakan arsitektur hyperlink, yaitu adanya kaitan antara satu dokumen web dengan dokumen web lain. Pemakai internet yang menggunakan fasilitas ini dapat dengan mudah mencari informasi yang diperlukan dengan mengunjungi web satu ke web lain. Informasi yang disajikan dalam halaman web selain ditampilkan dalam bentuk teks, dapat juga dalam bentuk grafis, video maupun suara. Dengan adanya fasilitas-fasilitas ini menjadi www sebagai salah satu layanan diinternet yang makin berkembang, paling popular dan paling banyak digunakan.

2.3 Sistem

2.3.1 Pengertian Sistem

Pengertian sistem dalam pengertian yang paling umum adalah sekumpulan benda yang memiliki hubungan di antara mereka. Kata sistem sendiri berasal dari bahasa Latin (*systēma*) dan bahasa Yunani (*sustēma*) adalah suatu kesatuan yang terdiri komponen atau elemen yang dihubungkan bersama untuk memudahkan aliran informasi, materi atau energi.

2.3.2 Pengertian Sistem Informasi

Sistem informasi adalah suatu sistem dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang mendukung fungsi operasi organisasi yang bersifat manajerial dengan kegiatan strategi dari suatu organisasi untuk dapat menyediakan kepada pihak luar tertentu dengan informasi yang diperlukan untuk pengambilan keputusan. Sistem informasi dalam suatu organisasi dapat dikatakan sebagai suatu sistem yang menyediakan informasi bagi semua tingkatan dalam organisasi tersebut kapan saja diperlukan. Sistem ini menyimpan, mengambil, mengubah, mengolah

dan mengkomunikasikan informasi yang diterima dengan menggunakan sistem informasi atau peralatan sistem lainnya.

2.3.3 Pengertian Sistem Informasi Akademik

Sistem Informasi Akademik adalah suatu sistem yang dirancang untuk keperluan pengeloaan data-data Akademik dengan penerapan teknologi komputer baik *hardware* maupun *software*, *hardware* (perangkat keras) adalah peralatan-peralatan seperti komputer (PC maupun Laptop), Printer, CD ROM, HardDisk, Handphone dan sebagainya. Sedangkan *Software* (perangkat lunak) merupakan program komputer yang memfungsikan *hardware* tersebut, sehingga seluruh proses kegiatan akademik dapat terkelola menjadi informasi yang bermanfaat dalam pengelolaan manajemen sekolah dan pengambilan keputusan-keputusan bagi pengambil keputusan atau top manajemen di lingkungan sekolah.

2.4 Hypertext Markup Language (HTML)

Hyper Text Markup Language (HTML) merupakan salah satu varian dari SGML (*Standard Generalized Markup Language*), yaitu sebuah standar dari ISO (*International Organization for Standardization*) untuk pertukaran dokumen secara elektronik melalui protocol HTTP. HTML sendiri merupakan tata penulisan yang digunakan dalam dokumen web. Dokumen ini, akan menghasilkan suatu dokumen sesuai dengan keinginan yang mendesain *page*. Dokumen ini mempunyai kemampuan menampilkan gambar, suara, teks, maupun penyediaan lirik terhadap halaman web lainnya, baik dengan alamat yang sama serta alamat yang berbeda. HTML sendiri secara formal diumumkan sebagai RFC 1866.

2.5 Hypertext Transfer Protocol (HTTP)

Hypertext Transfer Protocol adalah protokol TCP/IP pada layer aplikasi yang merupakan dasar *World Wide Web*. Di implementasikan pertama kali pada

akhir tahun 1990. HTTP merupakan protokol request-response. Sebuah program klien HTTP membuat koneksi dengan HTTP Server dan meminta halaman tertentu dari server, program server HTTP menerima koneksi tersebut dan menjawab permintaan dari klien. Secara singkat, HTTP adalah protokol yang melakukan hubungan antara klien dan server, yang mana pada sisi server, akan menerima permintaan berupa halaman web dari klien, lalu kemudian akan mengirim permintaan tersebut ke klien. Permintaan halaman web dari klien dilakukan dengan program yang sering disebut browser. Program ini banyak jenisnya, diantaranya adalah Microsoft Internet Explorer, Netscape, Mozilla, Konqueror, Lynx, dsb. Secara umum, HTTP bekerja pada port 80, pada jaringan TCP/IP.

2.5.1 Hypertext Transfer Protocol Secure (HTTPS)

HTTPS singkatan dari *HyperText Transport Protocol Secure*, memiliki pengertian sama dengan HTTP tetapi dengan alasan keamanan (*security*), HTTPS memberi tambahan *Secure Socket Layer* (SSL). Umumnya website yang menggunakan HTTPS ini adalah website yang memiliki tingkat kerawanan tinggi yang berhubungan dengan masalah keuangan dan privacy dari pelanggannya seperti website perbankan dan investasi.

Teknologi HTTPS protocol mencegah kemungkinan "dicurinya" informasi penting (*credit card* adalah contoh yang paling sering disebut-sebut) yang dikirimkan selama proses komunikasi berlangsung antara *user* dengan *web server* (atau sebaliknya). Secara teknis, *website* yang menggunakan HTTPS akan melakukan enkripsi terhadap informasi (data) menggunakan teknik enkripsi SSL. Dengan cara ini meskipun seseorang berhasil "mencuri" data tersebut selama dalam perjalanan user *web server*, orang tersebut tidak akan bisa membacanya karena sudah diubah oleh teknik enkripsi SSL. Untuk mengenali suatu *website* menggunakan SSL atau tidak, cukup dengan memperhatikan informasi di depan alamat *website* pada *browser*. Apabila alamat website diawali dengan <https://>

maka bisa dipatikan website tersebut sudah menggunakan teknologi SSL. Cara lain yaitu dengan melihat "icon kunci" di pojok kanan bawah browser, apabila "icon" tersebut terlihat dalam posisi terkunci maka website tersebut aman.

2.6 Database

Database adalah susunan *record* data operasional lengkap dari suatu organisasi atau perusahaan, yang diorganisir dan disimpan secara terintegrasi dengan menggunakan metode tertentu dalam komputer sehingga mampu memenuhi informasi yang optimal yang dibutuhkan oleh para pengguna. Banyak sekali kegiatan manusia yang menggunakan komputer sebagai sarana pengolahan data, sehingga diperlukan suatu perangkat lunak database. Jika dikaji lebih mendasar tentang batasan suatu database, maka dapat disebutkan bahwa segala bentuk koleksi data adalah suatu database. Definisi dasar struktur database yaitu:

- a) Data : sekumpulan fakta mengenai objek tertentu, orang dan lain-lain yang dinyatakan dengan angka, huruf, gambar, film, suara dan sebagainya yang relevan dan belum mempunyai arti.
- b) Informasi : hasil pengolahan data yang konkret dan sudah mempunyai arti untuk mencapai suatu tujuan tertentu.
- c) Tabel : merupakan hal yang paling mendasar dalam hal penyimpanan data yang terdiri dari *field* dan *record*.
- d) *Field* (kolom) : merupakan elemen dari tabel yang berisikan informasi tertentu yang spesifik tentang subjudul table pada sebuah item data.
- e) *Record* : Adalah suatu kumpulan beberapa *field* yang mempunyai tujuan yang sama.

2.7 Tinjauan Perangkat Lunak

Pada bagian ini penulis akan menguraikan penjelasan singkat mengenai perangkat lunak yang digunakan dalam pembuatan aplikasi

2.7.1 PHP (PHP: Hypertext Preprocessor)

PHP adalah kependekan dari PHP: *Hypertext Preprocessor*, bahasa interpreter yang mempunyai kemiripan dengan C dan Perl. PHP dapat digunakan bersama dengan HTML sehingga memudahkan dalam membuat aplikasi *web* dengan cepat. Dapat digunakan untuk menciptakan *dynamic website* baik itu yang memerlukan penggunaan database ataupun tidak.

Kelebihan PHP dari bahasa pemrograman lain yaitu :

- a) Bahasa pemrograman PHP adalah sebuah bahasa *script* yang tidak melakukan sebuah kompilasi dalam penggunaanya.
- b) *Web Server* yang mendukung PHP dapat ditemukan dimana - mana dari mulai *apache*, IIS, Lighttpd, hingga Xitami dengan konfigurasi yang relatif mudah.
- c) Dalam sisi pengembangan lebih mudah, karena banyaknya milis - milis dan *developer* yang siap membantu dalam pengembangan.
- d) Dalam sisi pemahamanan, PHP adalah bahasa *scripting* yang paling mudah karena memiliki referensi yang banyak.
- e) PHP adalah bahasa *open source* yang dapat digunakan di berbagai mesin (Linux, Unix, Macintosh, Windows) dan dapat dijalankan secara *runtime* melalui *console* serta juga dapat menjalankan perintah-perintah sistem.

2.7.2 MySQL

MySQL adalah sebuah database *server*, dapat juga berperan sebagai *client* sehingga sering disebut database *client / server*, yang *open source* dengan kemampuan dapat berjalan baik di OS (*Operating System*) manapun, dengan *Platform Windows* maupun Linux. Selain itu database ini memiliki beberapa Kelebihan di banding *database* lain, di antaranya adalah:

- a) MySQL sebagai *Database Management System (DBMS)*
- b) MySQL sebagai *Relation Database Management System (RDBMS)*

- c) MySQL adalah sebuah Software Database yang *Open Source*, artinya program ini bersifat *free* atau bebas digunakan oleh siapa saja tanpa harus membeli dan membayar lisensi kepada pembuatnya.
- d) MySQL merupakan sebuah *database server*, jadi dengan menggunakan *database* ini anda dapat menghubungkannya ke media internet sehingga dapat diakses dari jauh.
- e) MySQL merupakan sebuah *database client*. Selain menjadi *server* yang melayani permintaan, MySQL juga dapat melakukan *query* yang mengakses database pada *server*. Jadi MySQL dapat juga berperan sebagai *client*.
- f) MySQL mampu menerima *query* yang bertumpuk dalam satu permintaan atau yang disebut *Multi-Threading*.
- g) MySQL merupakan sebuah database yang mampu menyimpan data berkapasitas sangat besar hingga berukuran gigabyte sekalipun.

MySQL merupakan program *database server* dengan SQL sebagai bahasa yang digunakan di dalamnya. SQL dibagi menjadi dua bentuk *query*, yaitu :

1. DDL (*Data Definition Language*)

DDL adalah sebuah Metode Query SQL yang berguna untuk mendefinisikan data pada sebuah database, adapun Query yang dimiliki adalah :

- a) *CREATE* : Digunakan untuk melakukan pembuatan tabel dan database.
- b) *DROP* : Digunakan untuk melakukan penghapusan tabel maupun database.
- c) *ALTER*: Digunakan untuk melakukan pengubahan struktur tabel yang telah dibuat, baik menambah *Field* (*add*), mengganti nama *Field* (*change*) ataupun menamakannya kembali (*rename*), serta menghapus (*drop*).

2. DML (*Data Manipulation Language*)

DML adalah sebuah metode *Query* yang dapat digunakan apabila DDL telah terjadi, sehingga fungsi dari *Query* ini adalah untuk melakukan pemanipulasi database yang telah ada atau yang telah dibuat sebelumnya. Berikut *Query* yang termasuk di dalamnya adalah :

- a) *SELECT* : Digunakan untuk menampilkan data pada tabel.
- b) *INSERT* : Digunakan untuk melakukan penginputan / pemasukan data pada tabel *database*.
- c) *UPDATE* : Digunakan untuk melakukan pengubahan atau peremajaan terhadap data yang ada pada tabel.
- d) *DELETE* : Digunakan untuk melakukan penghapusan data pada tabel.

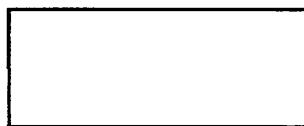
2.8 Data Flow Diagram (DFD)

Untuk memudahkan penggambaran suatu sistem yang ada atau sistem yang baru yang akan dikembangkan secara *logika* tanpa memperhatikan lingkungan fisik dimana data tersebut mengalir atau lingkungan fisik dimana data tersebut akan disimpan, maka kita menggunakan Data Flow Diagram. Data Flow Diagram merupakan alat yang cukup populer sekarang, karena dapat menggambarkan arus data di dalam suatu sistem dengan terstruktur dan jelas, itulah sebabnya DFD merupakan alat bantu yang paling penting bagi seorang analis sistem. DFD dapat merepresentasikan suatu sistem yang otomatis maupun manual dengan menggunakan gambar yang berbentuk jaringan grafik. Ada beberapa simbol yang digunakan dalam penggambaran DFD, yaitu :

Entitas Luar (*External Entity*) dan Terminal

Entitas Luar adalah entitas yang berada di luar sistem yang memberikan data kepada sistem (*source*) atau yang menerima informasi dari sistem (*sink*). Entitas Luar ini bukanlah bagian dari sistem, bila suatu sistem informasi dirancang untuk satu bagian/departemen maka bagian lain yang masih terkait menjadi entitas luar. Sedangkan terminal adalah *entitas* yang merupakan bagian dari sistem.

Simbol *entitas* dapat dilihat pada gambar 2.2 di bawah ini :



Gambar 2.2 Simbol *Entitas*

Pedoman pemberian nama terminal dan entitas luar

- a) Nama terminal berupa kata benda
- b) Terminal tidak boleh memiliki nama yang sama kecuali memang objeknya sama (digambarkan dua kali, dimaksudkan untuk membuat diagram lebih jelas). Bila demikian, maka terminal ini perlu diberi garis miring pada pojok kiri atas.
- c) Nama aliran data yang masuk ke dalam suatu proses tidak boleh sama dengan nama aliran data yang keluar dari proses tersebut.
- d) Data flow yang masuk ke atau keluar dari *data store* tidak perlu diberi nama bila:
 1. Aliran datanya sederhana dan mudah dipahami
 2. Aliran data menggambarkan seluruh data item (satu *record* utuh)
 3. Tidak boleh ada aliran data dari terminal ke *data store* atau sebaliknya karena terminal bukan bagian dari sistem. Hubungan terminal dengan *data store* harus melalui proses.

Proses

Proses menggambarkan apa yang dilakukan oleh sistem. Berfungsi mentransformasikan satu atau beberapa data masukan menjadi satu atau beberapa data keluaran sesuai dengan spesifikasi yang diinginkan. Setiap proses memiliki satu atau beberapa data masukan serta menghasilkan satu atau beberapa data keluaran. Proses sering pula disebut **bubble**.

Simbol *process* dapat dilihat pada gambar 2.3 di bawah ini :



Gambar 2.3 Simbol Proses

Pedoman pemberian nama proses :

- a) Nama proses terdiri dari kata kerja dan kata benda yang mencerminkan fungsi proses tersebut, misalnya : Hitung Gaji, Pendataan *Order*, Cetak Laporan Penjualan, dll.

- b) Jangan menggunakan kata ‘proses’ sebagai bagian dari nama suatu proses.
- c) Tidak boleh ada beberapa proses yang memiliki nama yang sama.
- d) Proses harus diberi nomor. Urutan nomor sedapat mungkin mengikuti aliran/urutan proses, namun demikian, urutan nomor tidak berarti secara mutlak merupakan urutan proses secara *kronologis*.
- e) Penomoran proses pada tingkat pertama (diagram nol) adalah 1.0, 2.0, 3.0, dst.
- f) Penomoran proses pada tingkat kedua dari proses 1.0 (rincian dari proses 1.0) adalah 1.1, 1.2, 1.3, dst.
- g) Diagram konteks (*context diagram*) tidak perlu diberi nomor.
- h) Proses 2.x adalah proses terendah, tidak dirinci lagi.

Alir Data (*Data Flow*)

Alir Data menggambarkan aliran data dari suatu *entitas* ke *entitas* lainnya.

Arah panah menggambarkan aliran data. Ada beberapa aliran data, yaitu :

1. Antara dua proses yang berurutan
2. Dari penyimpanan data (*data store*) ke proses dan sebaliknya
3. Dari *source* ke proses
4. Dari proses ke *sink*

Simbol Data Flow dapat dilihat pada gambar 2.4 di bawah ini:



Gambar 2.4 Simbol Data flow

Pedoman pemberian nama aliran data :

- a) Nama aliran data yang terdiri dari beberapa kata dihubungkan dengan garis sambung.
- b) Tidak boleh ada aliran data yang namanya sama dan pemberian nama harus mencerminkan isinya.
- c) Aliran data yang terdiri dari beberapa elemen dapat dinyatakan dengan grup elemen.

- d) Hindari penggunaan kata ‘data’ dan ‘informasi’ untuk memberi nama pada aliran data.
- e) Sedapat mungkin nama aliran data ditulis lengkap.

Penyimpan Data (*Data Store*)

Data store adalah tempat menyimpan data. Proses dapat mengambil data dari atau memberikan data ke *data store*.

Simbol *data store* dapat dilihat pada gambar 2.5 berikut:

DATA STORE

Gambar 2.5 Simbol *Data Store*

Pedoman pemberian nama *data store* :

- a) Nama harus mencerminkan isi *data store* tersebut.
- b) Bila namanya lebih dari satu kata, maka harus diberi tanda sambung

2.8.1 Diagram Konteks (*Context Diagram*)

- a) Merupakan level tertinggi dari DAD yang menggambarkan seluruh input ke atau dari sistem.
- b) Memberikan gambaran tentang keseluruhan sistem
- c) Terminal yang memberikan masukan kepada sistem disebut *source*, terminal yang menerima keluaran dari sistem disebut *sink*.
- d) Hanya ada satu proses.
- e) Tidak boleh ada *data store*.

2.8.2 Diagram nol (0)

- a) Sudah menunjukkan *data store* yang digunakan.
- b) Untuk proses yang tidak dirincé lagi pada level selanjutnya (*functional primitive*), tambahkan * pada akhir nomor proses.
- c) Keseimbangan input dan output (*balancing*) antara diagram 1.0 dengan diagram hubungan harus terpelihara

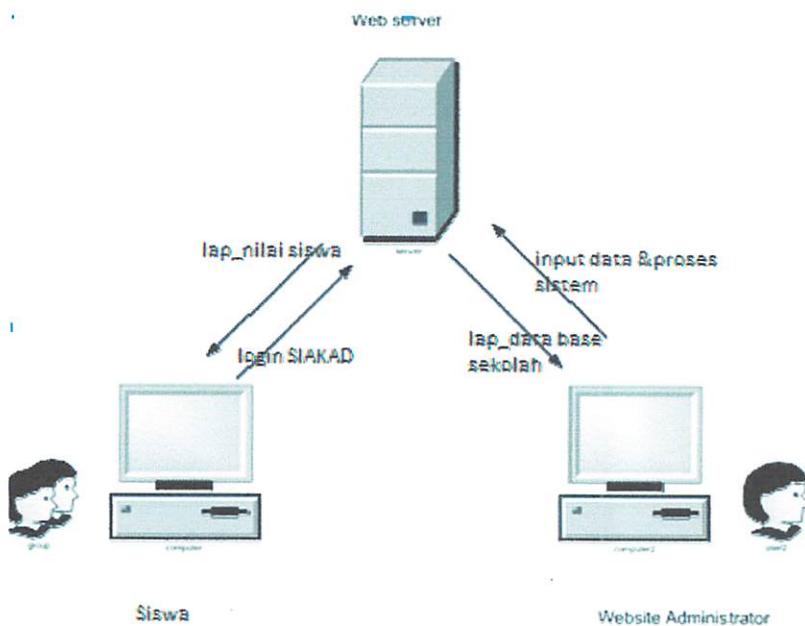
BAB III

ANALISIS SISTEM

3.1 Gambaran Umum

Website yang akan dibangun akan mengikuti prinsip CMS (*Content Management System*) dimana terdapat siswa , dan *administrator website*. *Registered user* artinya siswa yang telah memiliki *username* dan *password* khusus sehingga dapat mengakses layanan yang lebih lanjut. Sedangkan *administrator website* adalah *super administrator* memegang akses penuh. *Administrator website* dapat menambah/merubah *content website*, menambah/menghapus *user*, dan sebagainya.

Berikut ini adalah skema gambaran umum dari *website* ini.



Gambar 3.1 Gambaran Umum

3.2 Kebutuhan Sistem

Definisi kebutuhan sistem adalah faktor-faktor yang diperlukan oleh sistem dan diperlukan untuk perancangan perangkat lunak sehingga perangkat lunak tersebut sesuai dengan maksud dan tujuan pembuatannya.

3.2.1 Deskripsi Kebutuhan

Penjelasan mengenai kebutuhan sistem dibagi menjadi dua :

1. Kebutuhan Antar Muka

Website yang dibangun membutuhkan perangkat lain sebagai pendukung agar berjalan sesuai fungsinya. Kebutuhan tersebut adalah :

a. **Antarmuka pemakai**

Dibutuhkan sebuah komputer yang dilengkapi *web browser* dan terkoneksi dengan internet

b. **Aplikasi Server.**

Aplikasi server mengatur *request* ke *server* dan respon dari *server* ke *request*. Dibutuhkan *web server* dan Database server MySQL dan semuanya disediakan oleh pihak ketiga yaitu *web hosting* yang kita sewa.

2. Kebutuhan Fungsional

Merupakan daftar kebutuhan fungsional dari *website* yang akan dibangun.

Kebutuhan fungsional tersebut dijabarkan sebagai berikut:

a. **Tata Usaha**

Tabel 3.1 Kebutuhan Fungsional Perangkat Lunak Tata Usaha :

No	Deskripsi Kebutuhan	Keterangan
T01	Menampilkan data sekolah	halaman data sekolah
T02	Mengelola data sekolah	Form untuk merubah data sekolah
T03	Menampilkan Daftar Siswa	Halaman melihat daftar siswa
T04	Mengelola Data siswa	Form untuk merubah data siswa
T05	Menampilkan Biodata	Halaman melihat

	Siswa	biodata siswa
T06	Menampilkan Data Nilai	Halaman melihat data nilai
T07	Mengelola Data Nilai	Form untuk merubah data nilai
T08	Menampilkan laporan	Halaman melihat laporan

b. Guru

Tabel 3.2 Kebutuhan Fungsional Perangkat Lunak Guru :

No	Deskripsi Kebutuhan	Keterangan
G01	Menampilkan Daftar Siswa	Halaman melihat daftar siswa
G02	Menampilkan Data Nilai	Halaman melihat data nilai
G03	Mengelola Data Nilai	Form untuk merubah data nilai
G04	Menampilkan laporan	Halaman melihat laporan

c. Siswa

Tabel 3.3 Kebutuhan Fungsional Perangkat Lunak Siswa :

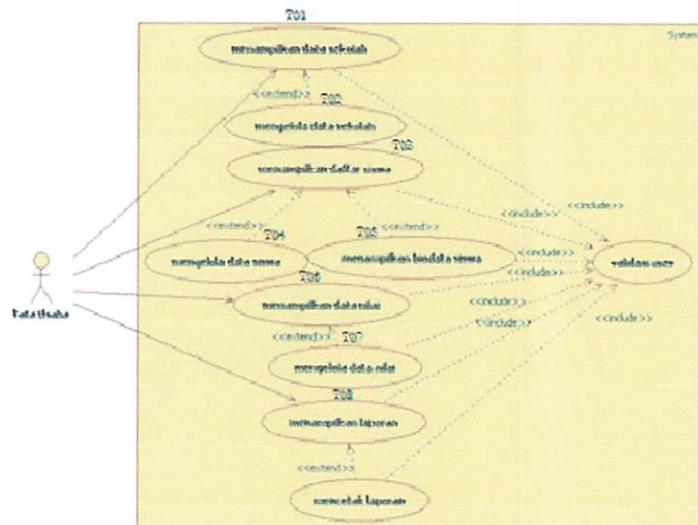
No	Deskripsi Kebutuhan	Keterangan
S01	Menampilkan Data Nilai	Halaman melihat data nilai
S02	Menampilkan Biodata Siswa	Halaman melihat biodata siswa

3.3 Pemodelan Kebutuhan Fungsional

Merupakan penggambaran dari kebutuhan fungsional dalam sistem pengolahan nilai siswa di Sekolah. Meliputi *Use case Diagram*.

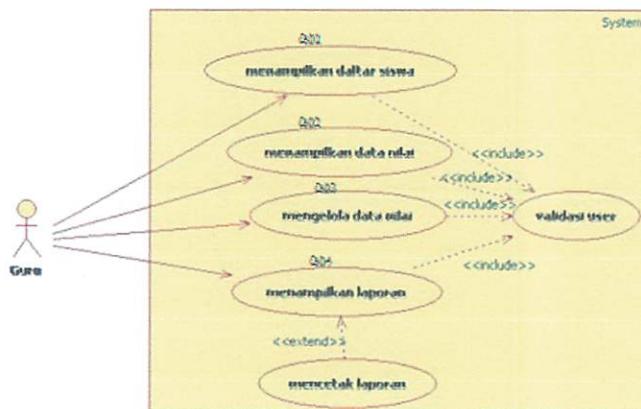
3.3.1 Use Case Diagram

a. Tata Usaha



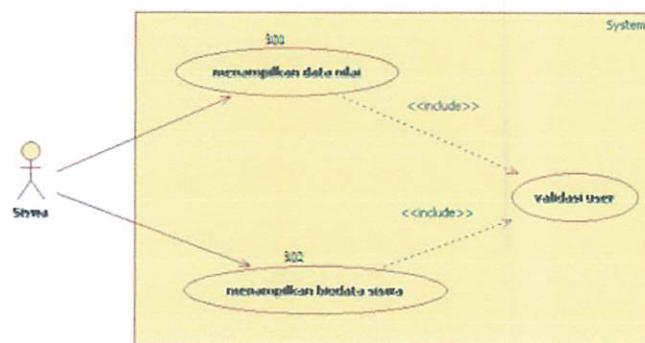
Gambar 3.2 Usecase Diagram Tata Usaha

b. Guru



Gambar 3.3 Usecase Diagram Guru

c. Siswa



Gambar 3.4 Usecase Diagram Siswa

3.4 Skenario

3.4.1 Skenario Usecase Diagram Tata Usaha

Tabel 3.4 Skenario Usecase Diagram Tata Usaha

Identifikasi	
No Use Case	T01
Nama Use Case	Menampilkan Data sekolah
Tujuan	Untuk melihat data sekolah
Deskripsi	Halaman berisi data umum sekolah
Jenis	Primer, esensial
Aktor	Tata Usaha
Skenario Utama	
Kondisi Awal	Halaman Utama admin
Aksi Aktor	
1. Memasukkan data login	2. Mengecek validasi
4. Sorot Data sekolah	3. Menampilkan menu utama
5. Klik data sekolah	6. Menampilkan Data Sekolah
Kondisi Akhir	Ditampilkan Halaman berisi data sekolah

Identifikasi	
No Use Case	T02
Nama Use Case	Mengelola Data Sekolah
Tujuan	Untuk merubah data sekolah yang ada
Deskripsi	Berupa form isian untuk merubah data sekolah
Jenis	Primer, esensial
Aktor	Tata Usaha
Skenario Utama	
Kondisi Awal	Halaman Utama admin
Aksi Aktor	
1. Memasukkan data login	2. Mengecek validasi
4. Sorot Data Sekolah	3. Menampilkan Menu utama
5. Klik data Sekolah	7. Menampilkan Form Isian Data Sekolah
6. Klik Ubah Data Sekolah	
Kondisi Akhir	Ditampilkan form Isian Data Sekolah

Identifikasi	
No Use Case	T03
Nama Use Case	Menampilkan Daftar Siswa
Tujuan	Untuk Menampilkan Daftar Siswa
Deskripsi	Menampilkan Nama-nama siswa yang aktif
Jenis	Primer, esensial
Aktor	Tata Usaha
Skenario Utama	
Kondisi Awal	Halaman Utama admin

Aksi Aktor	Respon Sistem
1. Memasukkan data login 4. Sorot Data Siswa 5. Klik daftar Siswa	2. Mengecek validasi 3. Menampilkan menu utama 6. Menampilkan Daftar Siswa
Kondisi Akhir	Ditampilkan halaman Berisi Daftar Siswa

Identifikasi	
No Use Case	T04
Nama Use Case	Mengelola Data Siswa
Tujuan	Untuk merubah Data siswa
Deskripsi	Menampilkan form untuk merubah data siswa
Jenis	Primer, esensial
Aktor	Tata Usaha
Skenario Utama	
Kondisi Awal	Halaman Utama admin
Aksi Aktor	Respon Sistem
1. Memasukkan data login 4. Sorot Data Siswa 5. Klik daftar Siswa 7. Klik Edit Data Siswa	2. Mengecek validasi 3. Menampilkan menu utama 6. Menampilkan Daftar Siswa 8. Menampilkan form edit siswa
Kondisi Akhir	Ditampilkan halaman form edit siswa

Identifikasi	
No Use Case	T05
Nama Use Case	Menampilkan Biodata Siswa
Tujuan	Untuk menampilkan biodata siswa
Deskripsi	Menampilkan halaman biodata per siswa
Jenis	Primer, esensial
Aktor	Tata Usaha
Skenario Utama	
Kondisi Awal	Halaman utama admin
Aksi Aktor	
1. Memasukkan data login 4. Sorot Data Siswa 5. Klik daftar Siswa 7. Klik Nama Siswa	
2. Mengecek validasi 3. Menampilkan menu utama 6. Menampilkan Daftar Siswa 8. Tampilkan Biodata siswa	
Kondisi Akhir	Ditampilkan Halaman biodata Siswa

Identifikasi	
No Use Case	T06
Nama Use Case	Menampilkan Data Nilai
Tujuan	Menampilkan Data Nilai per kelas
Deskripsi	Halaman berisi daftar nilai siswa per kelas
Jenis	Primer, esensial
Aktor	Tata Usaha
Skenario Utama	

Kondisi Awal	Halaman Utama admin
Aksi Aktor	Respon Sistem
1. Memasukkan data login 4. Sorot Data Nilai 5. Klik Data Nilai 6. Klik Jenis Angkatan dan Kelas	2. Mengecek validasi 3. Menampilkan menu utama 7. Menampilkan Data Nilai Siswa
Kondisi Akhir	Ditampilkan Halaman Data Nilai Per kelas

Identifikasi	
No Use Case	T07
Nama Use Case	Mengelola Data nilai
Tujuan	Untuk merubah Nilai siswa
Deskripsi	Berisi form inputan untuk merubah nilai siswa per kelas per matapelajaran
Jenis	Primer, esensial
Aktor	Tata Usaha
Skenario Utama	
Kondisi Awal	Halaman Utama admin
Aksi Aktor	Respon Sistem
1. Memasukkan data login 4. Sorot Data Nilai 5. Klik Data Nilai 6. Klik Jenis Angkatan dan Kelas 8. Klik Edit Nilai	2. Mengecek validasi 3. Menampilkan menu utama 7. Menampilkan Data Nilai Siswa 9. Tampilkan form isian data nilai
Kondisi Akhir	Ditampilkan Form Edit Nilai

Identifikasi	
No Use Case	T08
Nama Use Case	Menampilkan Laporan
Tujuan	Untuk melihat laporan sekolah
Deskripsi	Berisi berbagai macam laporan dari laporan data siswa sampai laporan nilai
Jenis	Primer, esensial
Aktor	Tata Usaha
Skenario Utama	
Kondisi Awal	Halaman Utama admin
Aksi Aktor	
1. Memasukkan data login 4. Sorot Laporan 5. Klik Laporan	
2. Mengecek validasi 3. Menampilkan menu utama 6. Menampilkan Laporan	
Kondisi Akhir	Menampilkan halaman berisi laporan

3.4.2 Skenario Usecase Diagram Guru

Tabel 3.5 Skenario Usecase Diagram Guru

Identifikasi	
No Use Case	G01
Nama Use Case	Menampilkan Daftar Siswa
Tujuan	Untuk Menampilkan Daftar Siswa
Deskripsi	Menampilkan Nama-nama siswa yang aktif
Jenis	Primer, esensial
Aktor	Guru
Skenario Utama	

Kondisi Awal	Halaman Utama guru
Aksi Aktor	Respon Sistem
1. Memasukkan data login 4. Sorot Data Siswa 5. Klik daftar Siswa	2. Mengecek validasi 3. Menampilkan menu utama 6. Menampilkan Daftar Siswa

Kondisi Akhir Ditampilkan halaman Berisi Daftar Siswa

Identifikasi	
No Use Case	G02
Nama Use Case	Menampilkan Data Nilai
Tujuan	Menampilkan Data Nilai per kelas
Deskripsi	Halaman berisi daftar nilai siswa per kelas
Jenis	Primer, esensial
Aktor	Guru
Skenario Utama	
Kondisi Awal	Halaman Utama guru
Aksi Aktor	Respon Sistem
1. Memasukkan data login 4. Sorot Data Nilai 5. Klik Data Nilai 6. Klik Jenis Angkatan dan Kelas	2. Mengecek validasi 3. Menampilkan menu utama 7. Menampilkan Data Nilai Siswa
Kondisi Akhir	Ditampilkan Halaman Data Nilai Per kelas

Identifikasi	
No Use Case	G03
Nama Use Case	Mengelola Data nilai
Tujuan	Untuk merubah Nilai siswa
Deskripsi	Berisi form inputan untuk merubah nilai siswa per kelas per matapelajaran
Jenis	Primer, esensial
Aktor	Guru
Skenario Utama	
Kondisi Awal	Halaman Utama guru
Aksi Aktor	
1. Memasukkan data login 4. Sorot Data Nilai 5. Klik Data Nilai 6. Klik Jenis Angkatan dan Kelas 8. Klik Edit Nilai	2. Mengecek validasi 3. Menampilkan menu utama 7. Menampilkan Data Nilai Siswa 9. Tampilkan form isian data nilai
Kondisi Akhir	Ditampilkan Form Edit Nilai

Identifikasi	
No Use Case	G04
Nama Use Case	Menampilkan Laporan
Tujuan	Untuk melihat laporan sekolah
Deskripsi	Berisi berbagai macam laporan dari laporan data siswa sampai laporan nilai
Jenis	Primer, esensial
Aktor	Guru
Skenario Utama	

Kondisi Awal	Halaman Utama guru
Aksi Aktor	Respon Sistem
1. Memasukkan data login 4. Sorot Laporan 5. Klik Laporan	2. Mengecek validasi 3. Menampilkan menu utama 6. Menampilkan Laporan
Kondisi Akhir	Menampilkan halaman berisi laporan

3.4.3 Skenario Usecase Diagram Siswa

Tabel 3.6 Skenario Usecase Diagram Siswa

Identifikasi	
No Use Case	S01
Nama Use Case	Menampilkan Data Nilai
Tujuan	Menampilkan Data Nilai per siswa
Deskripsi	Halaman berisi daftar nilai per siswa
Jenis	Primer, esensial
Aktor	Siswa
Skenario Utama	
Kondisi Awal	Halaman Utama siswa
Aksi Aktor	Respon Sistem
1. Memasukkan data login 4. Sorot Data Nilai 5. Klik Data Nilai 6. Klik Jenis Kelas	2. Mengecek validasi 3. Menampilkan menu utama 7. Menampilkan Data Nilai Siswa
Kondisi Akhir	Ditampilkan Halaman Data Nilai Per siswa

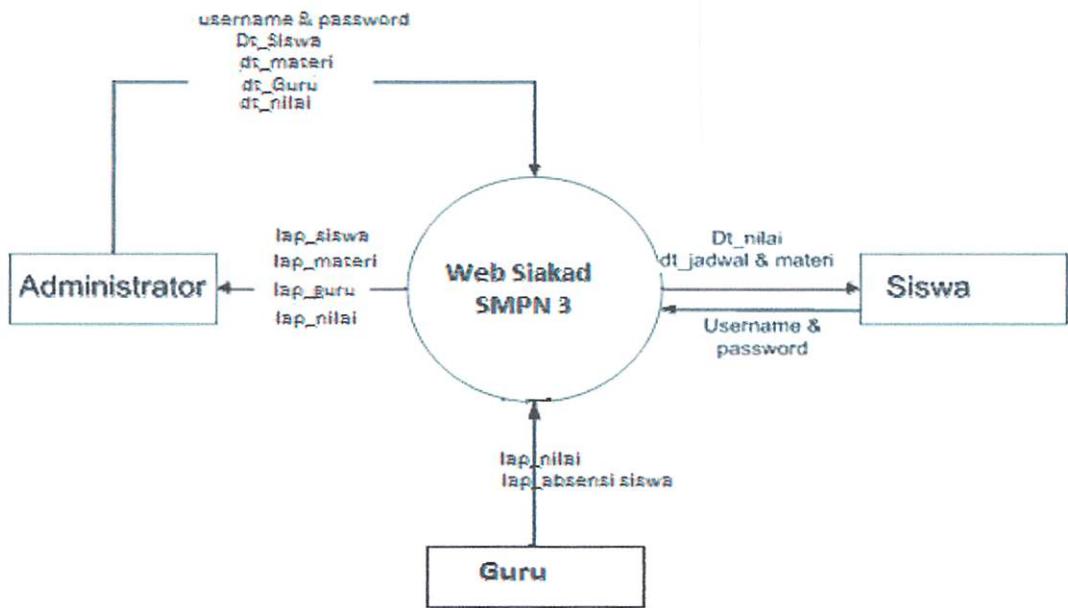
Identifikasi	
No Use Case	S02

Nama Use Case	Menampilkan Biodata Siswa				
Tujuan	Untuk menampilkan biodata siswa				
Deskripsi	Menampilkan halaman biodata per siswa				
Jenis	Primer, esensial				
Aktor	Siswa				
Skenario Utama					
Kondisi Awal	Halaman utama siswa				
<table border="1"> <thead> <tr> <th>Aksi Aktor</th> <th>Respon Sistem</th> </tr> </thead> <tbody> <tr> <td> 1. Memasukkan data login 4. Sorot Biodata Siswa 5. Klik Biodata Siswa </td> <td> 2. Mengecek validasi 3. Menampilkan menu utama 6. Tampilkan Biodata siswa </td> </tr> </tbody> </table>		Aksi Aktor	Respon Sistem	1. Memasukkan data login 4. Sorot Biodata Siswa 5. Klik Biodata Siswa	2. Mengecek validasi 3. Menampilkan menu utama 6. Tampilkan Biodata siswa
Aksi Aktor	Respon Sistem				
1. Memasukkan data login 4. Sorot Biodata Siswa 5. Klik Biodata Siswa	2. Mengecek validasi 3. Menampilkan menu utama 6. Tampilkan Biodata siswa				
Kondisi Akhir	Ditampilkan Halaman biodata Siswa				

3.5 Perancangan Sistem

Perancangan sistem di sini menggunakan beberapa *tools* yang digunakan untuk membantu *analisis*, yaitu *diagram konteks*, *data level diagram* serta *entity relational diagram*.

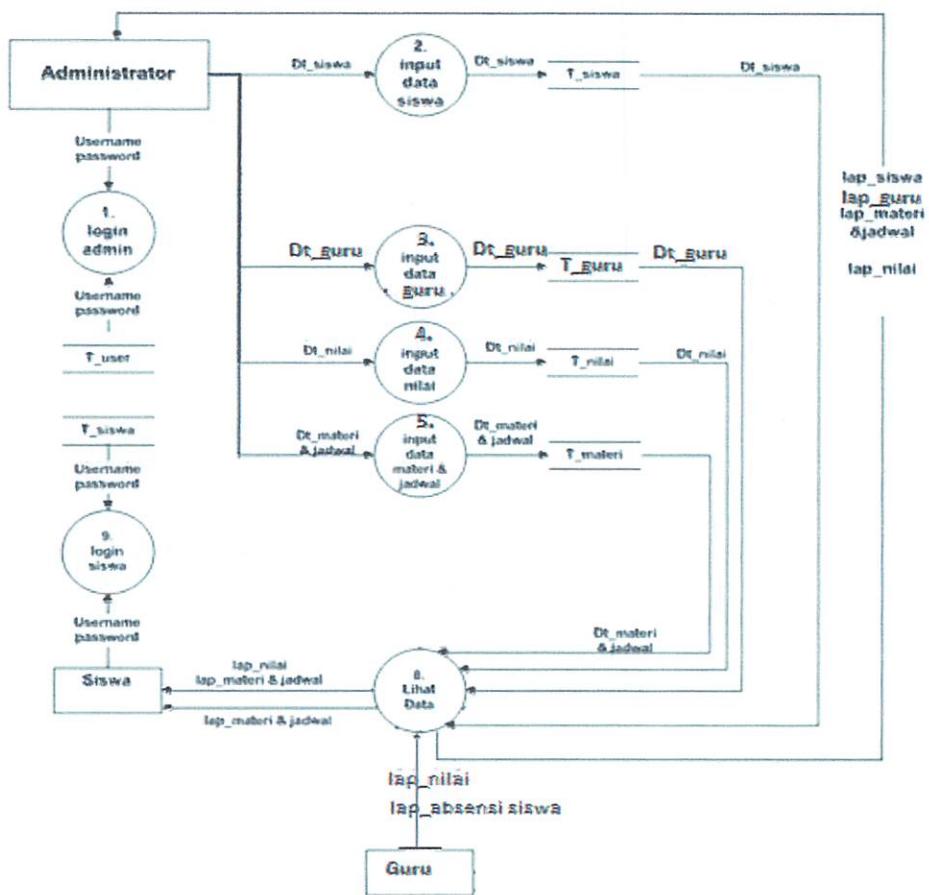
3.5.1 Konteks Diagram



Gambar 3.5 Diagram Konteks

3.5.2 Data Flow Diagram (DFD)

Data flow diagram website yang diusulkan untuk pengembangan sistem yang akan dirancang dengan mengacu kepada *diagram konteks* atau merupakan pengembangan dari *diagram konteks* atau bisa juga disebut sebagai turunan dari *diagram konteks*.

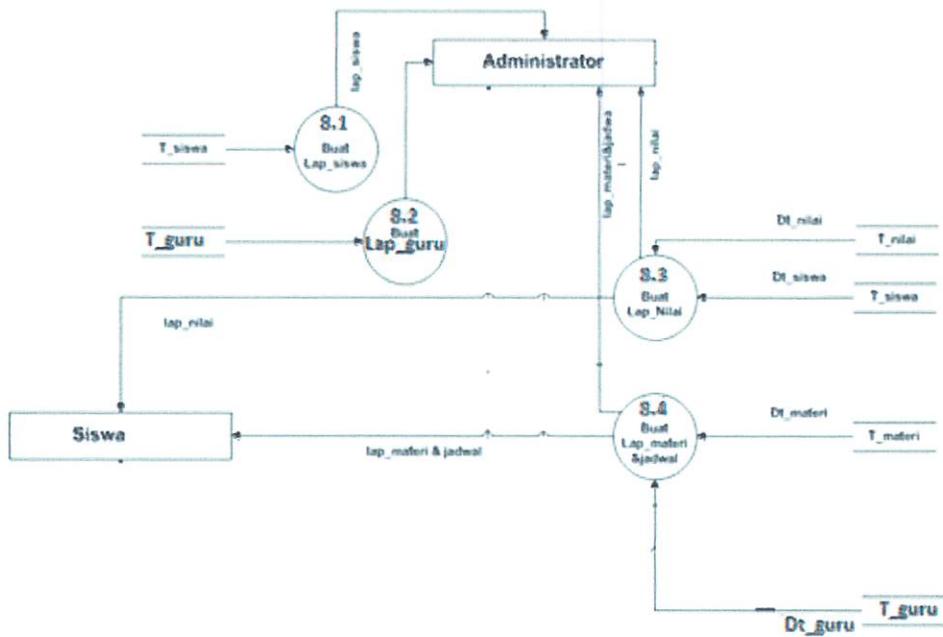


Gambar 3.6 Data Flow Diagram

3.5.3 Data Flow Diagram Proses 8 Level 1

Data Flow Diagram Proses 8 Level 1 merupakan turunan dari Data Flow Diagram Level 0 proses. Lebih jelasnya dapat dilihat pada gambar di bawah ini

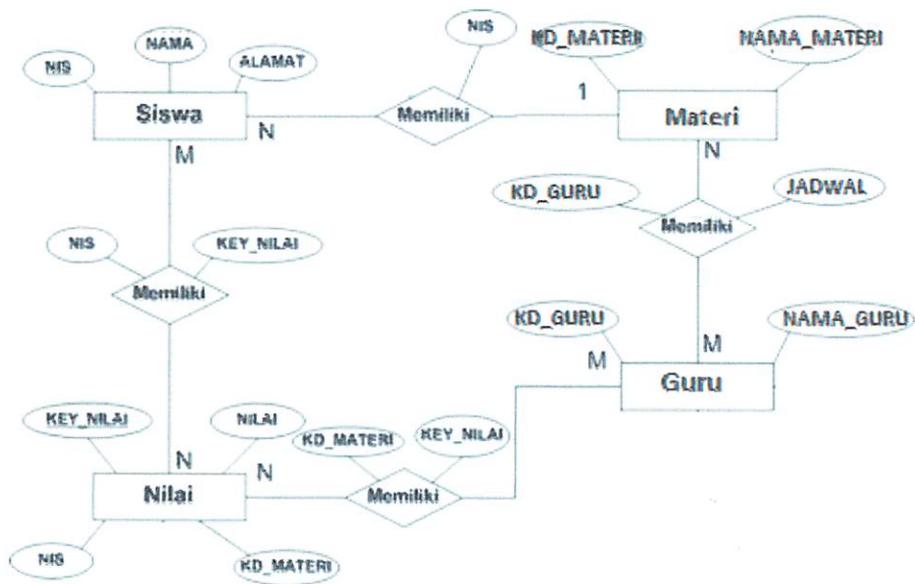
:



Gambar 3.7 DFD Level 1 Proses 8

3.6 Entity Relational Diagram (ERD)

Entity relational diagram merupakan salah satu tools untuk analisis perancangan yang menggambarkan relasi antar entitas.



Gambar 3.8 Entity Relational Diagram

3.7 Perancangan Basis Data

Pada tahap perancangan basis data ini, akan dijelaskan mengenai perancangan tentang database yang akan digunakan dalam *website* SMPN 3 Seteluk. Sebelum pengembangan *website*, terlebih dahulu dilakukan penyusunan struktur database secara benar.

Rancangan tabel-tabel dalam database adalah sebagai berikut :

1. Tabel 3.7 Rancangan *Database* Siswa

Fungsi : Menyimpan data siswa

Primary key : nim

Foreign key :

Jenis tabel : Master

Struktur tabel :

No	Nama field	Type	Size	Keterangan
1	Nim	Char	10	Nomor induk murid
2	Nisn	Char	11	Nomor NISN
3	nama_murid	Char	35	Nama lengkap murid
4	jenis_kelamin	Tinyint	1	Jenis kelamin murid
5	tmp_lahir	Char	30	Tempat lahir
6	tgl_lahir	Date		Tanggal lahir
7	Agama	Tinyint	1	Agama murid
8	Alamat	Text		Alamat tempat tinggal
9	Telepon	Char	12	Telepon / HP
10	Nama_ortu	Char	35	Nama Ayah
11	tanggal_edit	Date		Tanggal data di edit
12	Pic_siswa	varchar	100	Nama photo

2. Tabel 3.8 Rancangan *Database* Guru

Fungsi : Menyimpan data guru

Primary key : NIP

Foreign key :

Jenis tabel : Master

Struktur tabel :

No	Nama field	Type	Size	Keterangan
1	NIP	char	18	Nomor induk Pegawai
2	NUPTK	char	18	Nomor Unit Pegawai
3	Nama	char	35	Nama lengkap Guru
4	jenis_kelamin	tinyint	1	Jenis kelamin murid
5	tmp_lahir	char	30	Tempat lahir
6	tgl_lahir	date		Tanggal lahir
7	Agama	tinyint	1	Agama
8	Alamat	text		Alamat tempat tinggal
9	Telepon	char	12	Telepon / HP
10	tanggal_edit	date		Tanggal data di edit
11	Pic_guru	varchar	100	Nama photo

3. Tabel 3.9 Rancangan Database Tahun Ajaran

Fungsi : Menyimpan data Tahun Ajaran

Primary key : Id_thn_ajaran

Foreign key :

Jenis tabel : Master

Struktur tabel :

No	Nama field	Type	Size	Keterangan
1	id_thn_ajaran	Char	8	Kode tahun Ajaran
2	Tahun_ajaran	Char	10	Nama tahun ajaran

4. Tabel 3.10 Rancangan Database kelas

Fungsi : Menyimpan data kelas

Primary key : id_kelas

Foreign key :

Jenis tabel : Master

Struktur tabel :

No	Nama field	Type	Size	Keterangan
1	id_kelas	Int	4	Kode kelas
2	Tingkat_kelas	tinyint	1	Tingkatan kelas
3	Kapasitas	Int	2	Kapasitas siswa per kelas

5. Tabel 3.11 Rancangan Database Mata Pelajaran

Fungsi : Menyimpan data mata pelajaran

Primary key : id_matpel

Foreign key :

Jenis tabel : Master

Struktur tabel :

No	Nama field	Type	Size	Keterangan
1	id_matpel	Int	4	Kode mata pelajaran
2	Nama_matpel	varchar	35	Nama matapelajaran
3	Jenis_matpel	tinyint	1	Jenis matapelajaran

6. Tabel 3.12 Rancangan Database Nilai

Fungsi : Menyimpan data nilai siswa

Primary key : id_nilai

Foreign key : id_thn_ajaran, id_kelas, id_parallel, id_matpel, nim

Jenis tabel : Transaksi

Struktur tabel :

No	Nama field	Type	Size	Keterangan
1	id_nilai	Int	255	Kode nilai

2	Nim	char	10	Nomor Induk Murid
3	Id_thn_ajaran	tinyint	1	Kode Tahun Ajaran
4	Id_kelas	int	4	Kode kelas
5	Id_parallel	int	4	Kode kelas parallel
6	Sem	tinyint	1	Semester
7	Id_matpel	int	4	Kode mata pelajaran
8	Jenis_nilai	tinyint	1	Jenis penilaian
9	Nilai	float	5	Besar nilai

7. Tabel 3.13 Rancangan Database Wali Kelas

Fungsi : Menyimpan data guru wali kelas

Primary key : id_guru_kelas

Foreign key : id_thn_ajaran, id_kelas, id_parallel, NIP

Jenis tabel : Transaksi

Struktur tabel :

No	Nama field	Type	Size	Keterangan
1	id_guru_kelas	int	99	Kode guru kelas
2	Id_thn_ajaran	tinyint	1	Kode Tahun Ajaran
3	Id_kelas	int	4	Kode kelas
4	Id_parallel	int	4	Kode kelas parallel
5	NIP	char	18	Nomor Induk Pegawai

8. Tabel 3.14 Rancangan Database Mata Pelajaran Kelas

Fungsi : Menyimpan data mata pelajaran per kelas

Primary key : id_matpel_kls

Foreign key : id_kelas, id_matpel

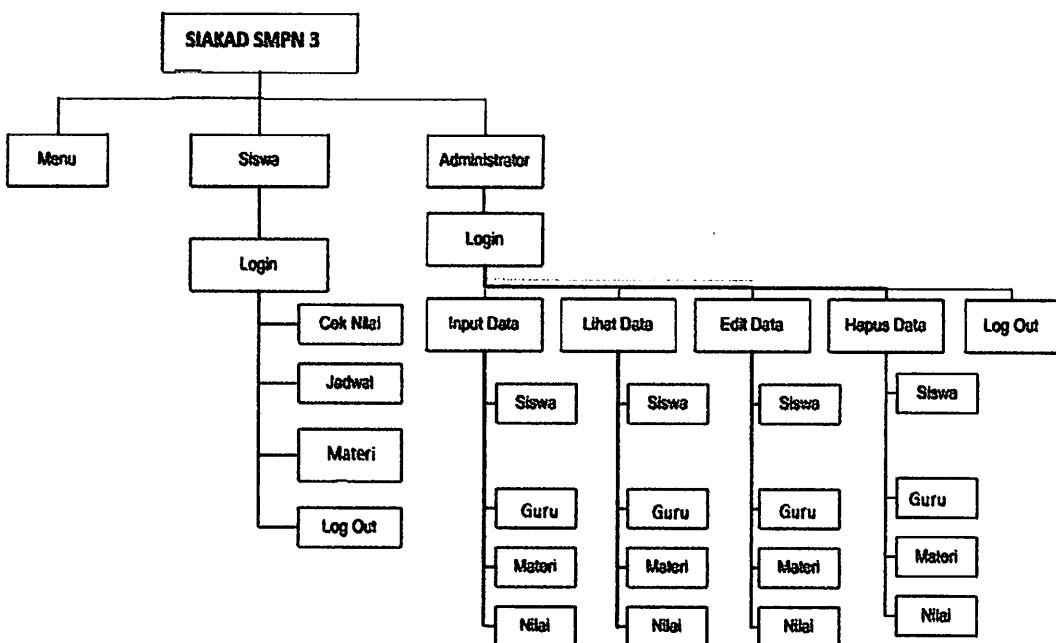
Jenis tabel : Transaksi

Struktur tabel :

No	Nama field	Type	Size	Keterangan
1	id_matpel_kls	Int	5	Kode matapelajaran per kelas
2	Id_kelas	Int	4	Kode Kelas
3	Id_matpel	Int	4	Kode mata pelajaran

3.8 Struktur Menu

Struktur menu website dibuat sederhana agar setiap pemakai dapat menggunakannya tanpa kesulitan. Untuk lebih jelasnya tentang bentuk rancangan menu dapat dilihat pada gambar di bawah ini

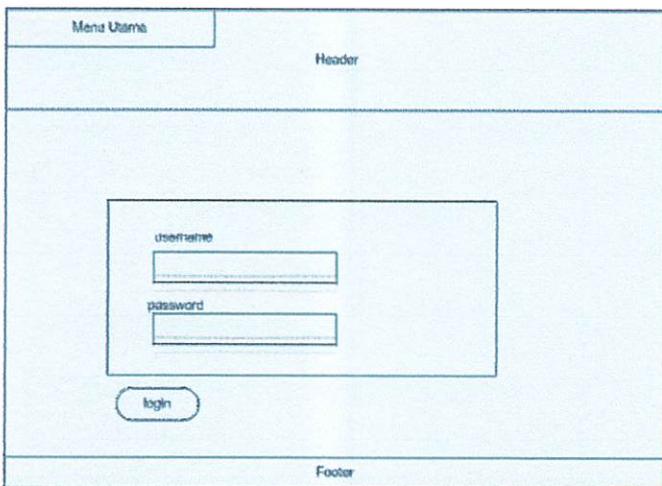


Gambar 3.9 Struktur Menu

3.9 Antarmuka Pemakai

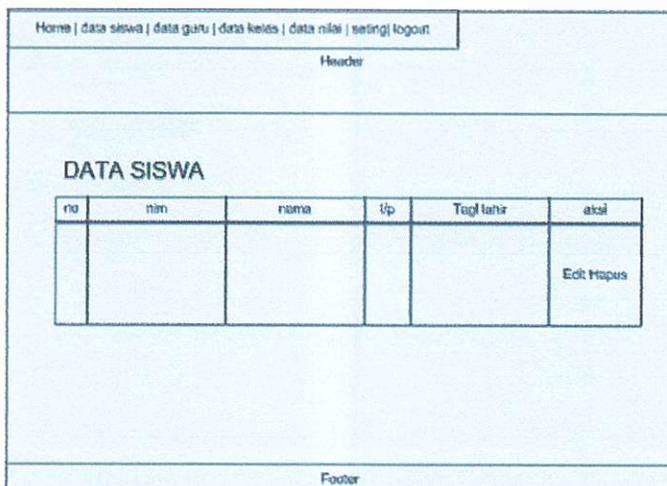
Tata letak layar merupakan rancangan dalam menata tampilan layar yang akan digunakan dalam pembuatan program sebagai sarana memasukkan data. Di dalam program aplikasi yang akan dibuat, sebagai berikut :

1. Tata letak layar halaman Login



Gambar. 3.10 Tata Letak Halaman *Login*

2. Tata letak layar halaman daftar siswa



Gambar. 3.11 Tata Letak Halaman Daftar Siswa

3. Tata letak layar halaman daftar guru

Home data siswa data guru data kelas data nilai setting logout																	
Header																	
DATA GURU																	
<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>no</th> <th>NIP</th> <th>nama</th> <th>I/p</th> <th>Tgl lahir</th> <th>aksi</th> </tr> </thead> <tbody> <tr> <td></td> <td></td> <td></td> <td></td> <td></td> <td>Edit Hapus</td> </tr> </tbody> </table>						no	NIP	nama	I/p	Tgl lahir	aksi						Edit Hapus
no	NIP	nama	I/p	Tgl lahir	aksi												
					Edit Hapus												
Footer																	

Gambar. 3.12 Tata Letak Halaman Daftar Guru

4. Tata letak layar halaman daftar kelas

Home data siswa data guru data kelas data nilai setting logout													
Header													
DATA KELAS													
<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>no</th> <th>kelas</th> <th>paralel</th> <th>aksi</th> </tr> </thead> <tbody> <tr> <td></td> <td></td> <td></td> <td>Edit Hapus</td> </tr> </tbody> </table>						no	kelas	paralel	aksi				Edit Hapus
no	kelas	paralel	aksi										
			Edit Hapus										
Footer													

Gambar. 3.13 Tata Letak Halaman Daftar Kelas

5. Tata letak layar halaman daftar nilai

Home data siswa data guru data kelas data nilai setting logout																																									
Header																																									
DATA NILAI																																									
<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td>Tahun Ajaran</td> <td><input type="text"/></td> </tr> <tr> <td>Kelas</td> <td><input type="text"/></td> </tr> <tr> <td colspan="6"> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>no</th> <th>NIM</th> <th>nama</th> <th>MAT</th> <th>IND</th> <th>IPA</th> <th>IPS</th> <th>PKN</th> <th>PNJ</th> <th>aksi</th> </tr> </thead> <tbody> <tr> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td>Edit Hapus</td> </tr> </tbody> </table> </td> </tr> <tr> <td colspan="6" style="text-align: center;">Footer</td> </tr> </table>						Tahun Ajaran	<input type="text"/>	Kelas	<input type="text"/>	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>no</th> <th>NIM</th> <th>nama</th> <th>MAT</th> <th>IND</th> <th>IPA</th> <th>IPS</th> <th>PKN</th> <th>PNJ</th> <th>aksi</th> </tr> </thead> <tbody> <tr> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td>Edit Hapus</td> </tr> </tbody> </table>						no	NIM	nama	MAT	IND	IPA	IPS	PKN	PNJ	aksi										Edit Hapus	Footer					
Tahun Ajaran	<input type="text"/>																																								
Kelas	<input type="text"/>																																								
<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>no</th> <th>NIM</th> <th>nama</th> <th>MAT</th> <th>IND</th> <th>IPA</th> <th>IPS</th> <th>PKN</th> <th>PNJ</th> <th>aksi</th> </tr> </thead> <tbody> <tr> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td>Edit Hapus</td> </tr> </tbody> </table>						no	NIM	nama	MAT	IND	IPA	IPS	PKN	PNJ	aksi										Edit Hapus																
no	NIM	nama	MAT	IND	IPA	IPS	PKN	PNJ	aksi																																
									Edit Hapus																																
Footer																																									

Gambar. 3.14 Tata Letak Halaman Daftar Nilai

6. Tata letak layar halaman rubah siswa

Header

RUBAH DATA SISWA

Nama

Jenis kelamin

Tempat lahir

Tanggal lahir

alamat

Nama ayah

Footer

Gambar. 3.15 Tata Letak Halaman Rubah Siswa

7. Tata letak layar halaman rubah guru

Header

RUBAH DATA GURU

NIP

Nama

Jenis kelamin

Tempat lahir

Tanggal lahir

alamat

Footer

Gambar. 3.16 Tata Letak Halaman Rubah Guru

8. Tata letak layar halaman rubah nilai

Header

RUBAH DATA NILAI

Tahun Ajaran

Kelas

no	NIM	nama	MAT	IND	IPA	IPS	PKN	PNJ	aksi

Footer

Gambar. 3.17 Tata Letak Halaman Rubah Nilai

BAB IV

IMPLEMENTASI DAN PENGUJIAN

4.1 Implementasi Sistem

Implementasi sistem adalah proses penerapan racangan sistem yang telah dibuat menjadi suatu aplikasi yang siap untuk dijalankan pada dunia nyata IT (Teknik Informatika). Implementasi sistem berfungsi untuk menerapkan sistem sesuai dengan tujuan sistem, tujuan sistem yang kami maksud adalah merancang sistem informasi akademik. Disamping implementasi sistem ini berfungsi juga untuk di terapkan guna memudahkan siswa, guru, tata usaha maupun masyarakat luas yang membutuhkan segala informasi yang ada di SMPN 3 Seteluk.

4.1.1 Spesifikasi Perangkat Keras

Implementasi sistem aplikasi SIAKAD SMPN 3 Seteluk membutuhkan perangkat keras dengan spesifikasi minimal sebagai berikut:

- a) Processor Pentium 4 1.8 Ghz
- b) Memory 512 MB
- c) Sisa ruang Hard Disk 1 GB
- d) Monitor SVGA dengan resolusi 1024x768
- e) Keyboard

4.1.2 Spesifikasi Perangkat Lunak

Implementasi sistem aplikasi SIAKAD SMPN 3 Seteluk, *software / piranti lunak* yang digunakan adalah sebagai berikut:

- a) Sistem Operasi Windows (Xp dan 7)

Sistem operasi yang digunakan untuk membuat aplikasi SIAKAD SMPN 3 Seteluk menggunakan sistem operasi Windows (Xp dan 7). Sistem

operasi Windows (Xp dan 7) berfungsi untuk pembuatan *website* dan *platform* dari *web* yang di buat.

b) Macromedia Dreamweaver 8

Macromedia Dreamweaver 8 adalah *software* yang digunakan untuk pembuatan aplikasi SIAKAD SMPN 3 Seteluk berbasis web. *Software* ini sangat mendukung pembuatan *website*, seperti pembuatan *script php* dan pembuatan tampilan *website* menggunakan GUI.

c) PHP 4.2

Perl Hypertext Preprocessor (PHP) merupakan suatu *script* yang bersifat *open source* dan dapat digunakan untuk membuat sebuah web menjadi lebih menarik, dinamis, dan interaktif.

d) XAMPP

XAMPP adalah perangkat lunak bebas yang mendukung banyak sistem operasi. Merupakan komplikasi dari beberapa program yang berfungsi sebagai *server* yang berdiri sendiri (*localhost*), yang terdiri atas program apache http server, MySQL database, dan penerjemah bahasa yang ditulis dengan bahasa bahasa pemrograman PHP dan Perl.

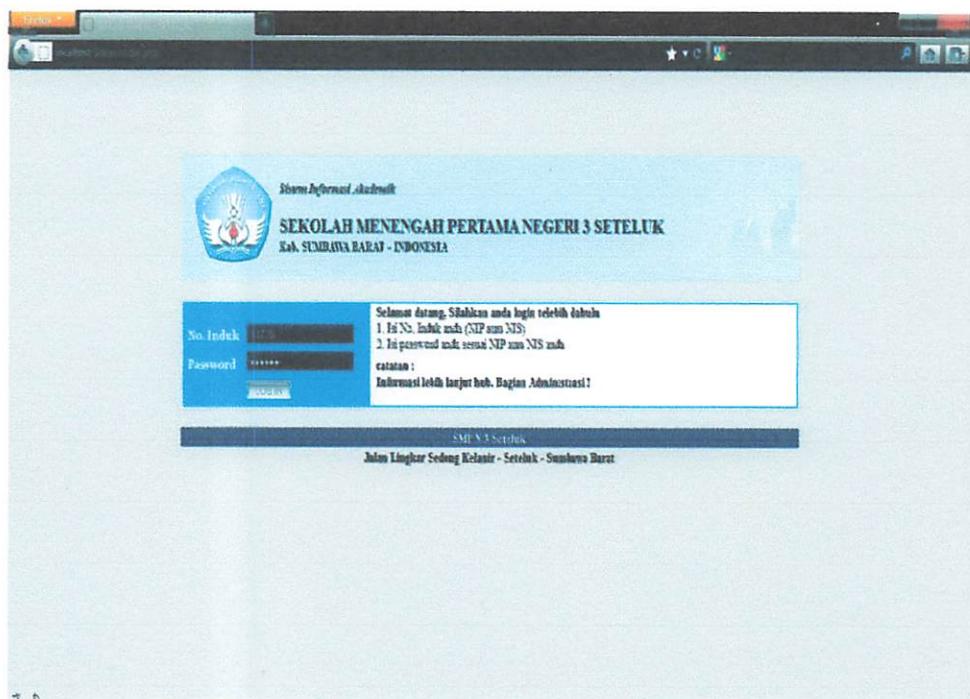
4.2 Implementasi Program

Implementasi ini dilakukan agar pemakai sistem dapat mengerti tentang bagaimana cara kerja dari sistem ini. Berikut adalah implementasi dari sistem informasi akademik sekolah berbasiskan web :

4.2.1 Halaman Login Siakad

Halaman login adalah halaman awal ketika sistem akademik di akses. Pada pengujian kali ini masih menggunakan *localhost* jadi untuk mengaksesnya dengan mengetik *localhost/siakad/index.php* setelah itu *browser* akan mengakses halaman *login* atau indek seperti pada gambar 4.1 berikut ini.





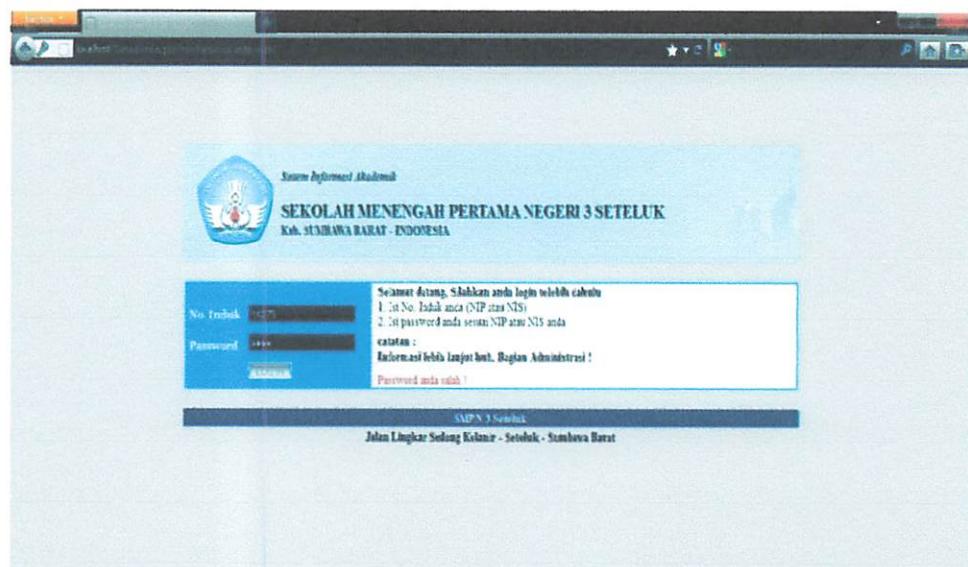
Gambar 4.1 Halaman *Login* Siakad

Pada halaman *login* terdapat inputan untuk *user*, inputan tersebut yaitu inputan no. induk dan password. *User* wajib memasukan no induk dan *login* agar dapat mengakses web siakad, setelah mereka *login* maka sistem akan mengecek apakah *user* telah terdaftar atau belum. Jika *user* telah terdaftar maka sistem akan mengakses hak akses *user*, dan *user* langsung masuk ke Sistem Informasi Akademik SMPN 3 Seteluk seperti gambar 4.2 berikut ini. Pada sistem akademik ini terdapat tiga hak akses yaitu admin, guru dan murid seperti yang telah dijelaskan pada bab 3 sebelumnya.



Gambar 4.2 Halaman Menu Utama Saat *login* Berhasil

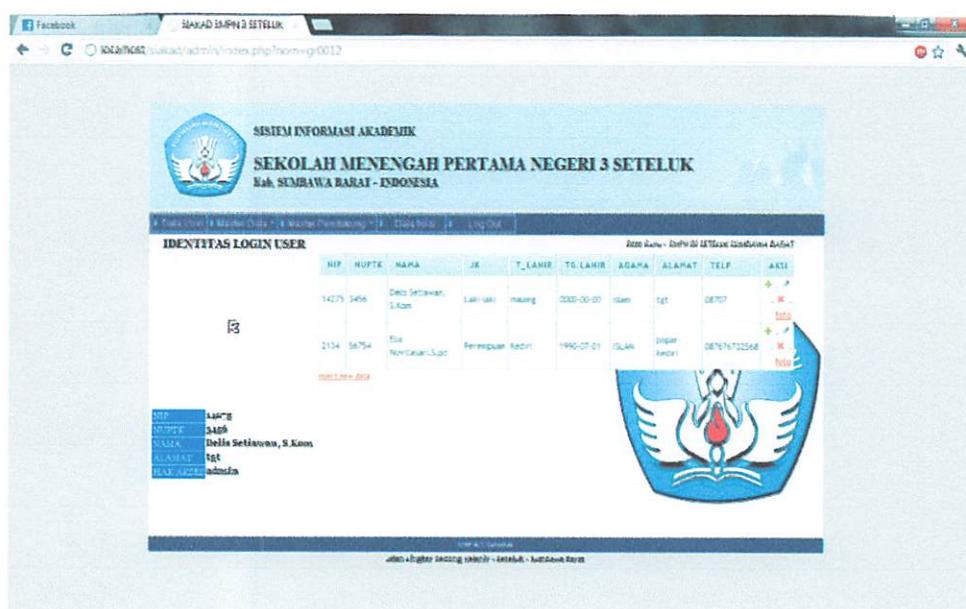
Akan tetapi bila user tidak terdaftar akan muncul sebuah peringatan “*password* anda salah” pada halaman *login* Sistem Informasi Akademik SMPN 3 Seteluk seperti gambar 4.3 berikut ini.



Gambar 4.3 Halaman Peringatan Bila *Password* Anda Salah

4.2.2 Halaman admin

Halaman admin adalah halaman khusus untuk hak akses admin, pada halaman admin ini admin dapat menginputkan data seperti data siswa, guru, kelas dan *fitur -fitur* lainnya, akan tetapi admin tidak diberi hak untuk menginputkan nilai. Nilai siswa hanya bisa diinputkan oleh guru wali kelas saja, jadi bukan sekedar guru saja. Berikut ini tampilan halaman *admin* seperti gambar 4.4 .



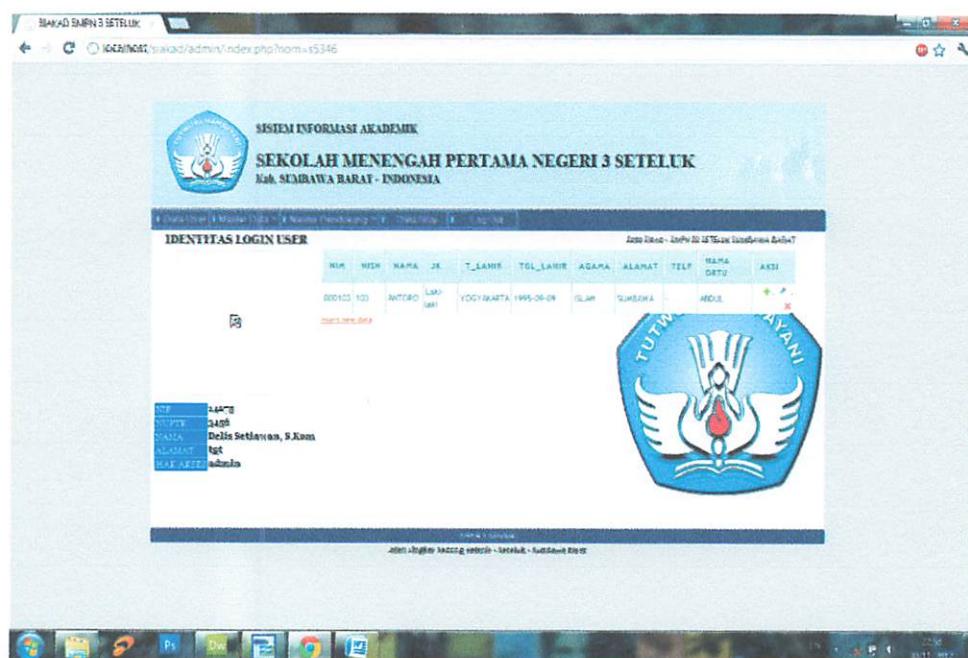
Gambar 4.4 Data Guru

Screen shot diatas menunjukan data admin yang *login* dan juga foto admin. Pada tampilan diatas juga terdapat menu-menu yang dapat digunakan oleh admin diantara data user, master data seperti data guru, murid, wali kelas dan seterusnya. Sebelah kanan identitas login user adalah halaman isi menu yang di pilih seperti tampilan gambar 4.4 adalah isi dari menu master data – guru.

4.2.3 Halaman Data Siswa

Halaman data siswa adalah salah satu menu yang disediakan kepada admin, tujuannya adalah agar admin dapat mendata siswa sebagai data master pada aplikasi siakad ini. Halaman data siswa berisi list siswa dan menu aksi/*action* pada list tersebut, *action* pada *list* terdiri dari *insert*

data, update data dan *delete data*. Berikut ini gambar 4.5 adalah tampilan halaman data siswa.

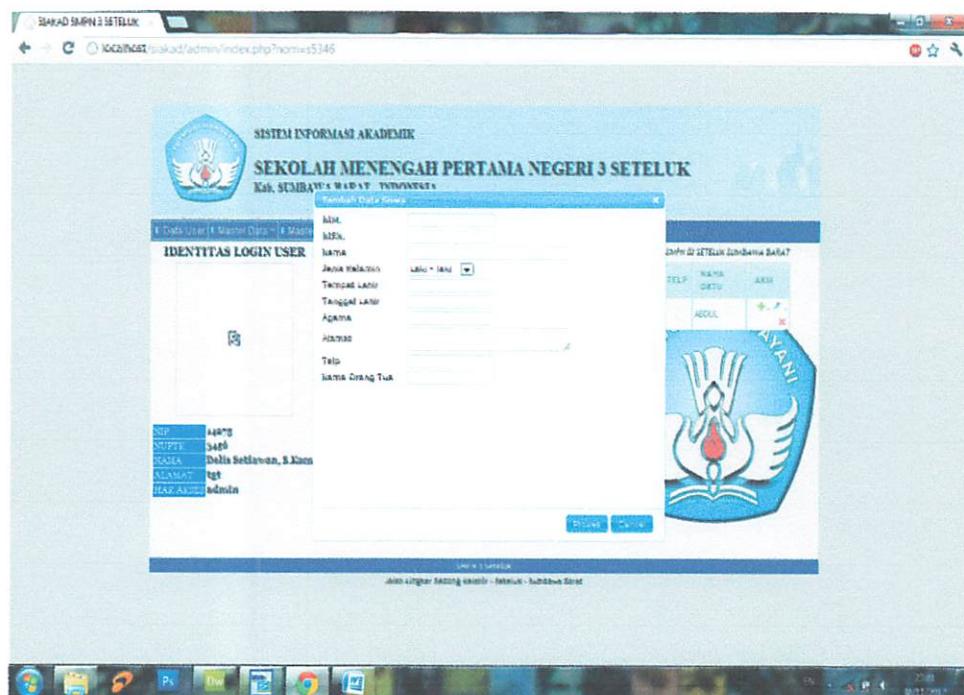


Gambar 4.5 Halaman Data Siswa

Seperti penjelasan sebelumnya pada halaman data siswa terdapat aksi atau *action* untuk memanipulasi data siswa. *Action* tersebut adalah *insert data*, *update data* dan *delete data*, tujuannya adalah agar *admin* dapat memanipulasi data dengan mudah dan lebih aman jika dibandingkan dengan mengubah data dari *database* langsung. Berikut ini penjelasan dari tiap aksi atau *action* pada halaman data siswa.

a) **Aksi/action Insert Data**

Aksi/ *action* ini digunakan untuk menambahkan data siswa ke dalam *database* siswa. Dengan menggunakan aksi ini admin cukup mengisikan data pada setiap *field* yang ada pada *database*. Aksi ini juga berguna untuk menjaga keamanan data atau mengurangi resiko kesalahan penginputan data langsung ke database. Berikut ini 4.6 tampilan aksi *insert* data pada aplikasi ini.

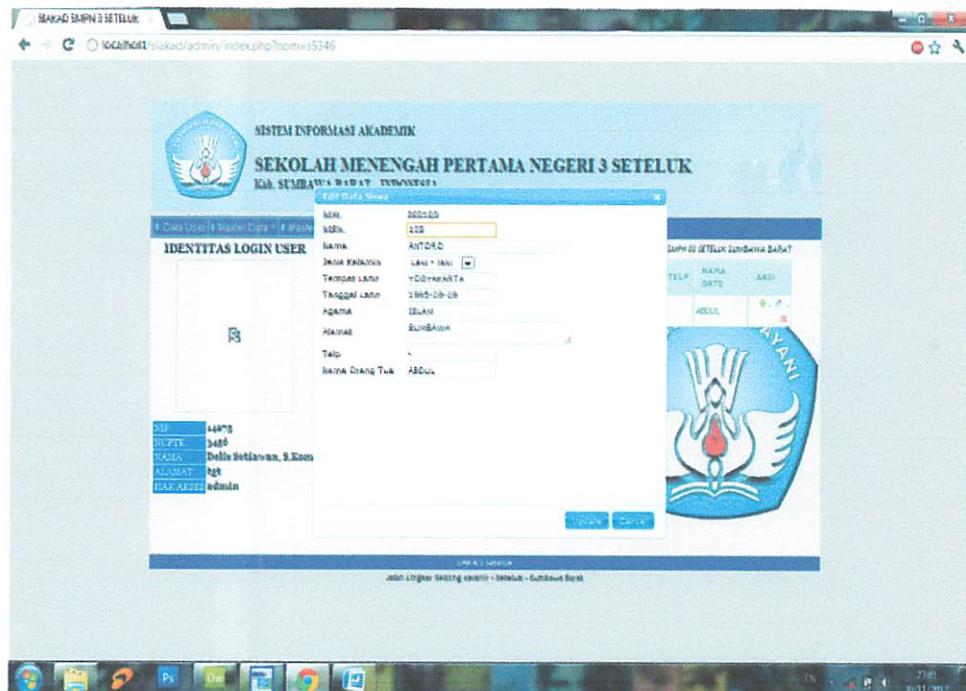


Gambar 4.6 Halaman Aksi *Insert* Data Siswa

Tampilan aksi/*action insert* data pada gambar 4.6 menampilkan *field – field* data yang perlu diisi oleh *admin* untuk pendaftaran siswa. Pada saat *admin* menekan aksi/*action insert* data maka *otomatis* akan keluar *form insert* data siswa, *form* tersebut terdapat dua pilihan tombol proses dan *cancel*. Tombol proses digunakan untuk memproses data yang ada pada *form* untuk dipindah ke dalam *database*. Sedangkan tombol *cancel* digunakan untuk membatalkan penambahan data atau *insert* data.

b) **Aksi/*action Update* Data**

Aksi/*action* ini digunakan untuk mengupdate atau mengedit data siswa ke dalam *database* siswa. Dengan menggunakan aksi ini *admin* cukup mengisikan data perubahan pada setiap *field* yang ada pada *database*. Aksi ini juga berguna untuk menjaga keamanan data atau mengurangi resiko kesalahan pengubahan data langsung ke *database*. Berikut ini 4.7 tampilan aksi *update* data pada aplikasi ini.

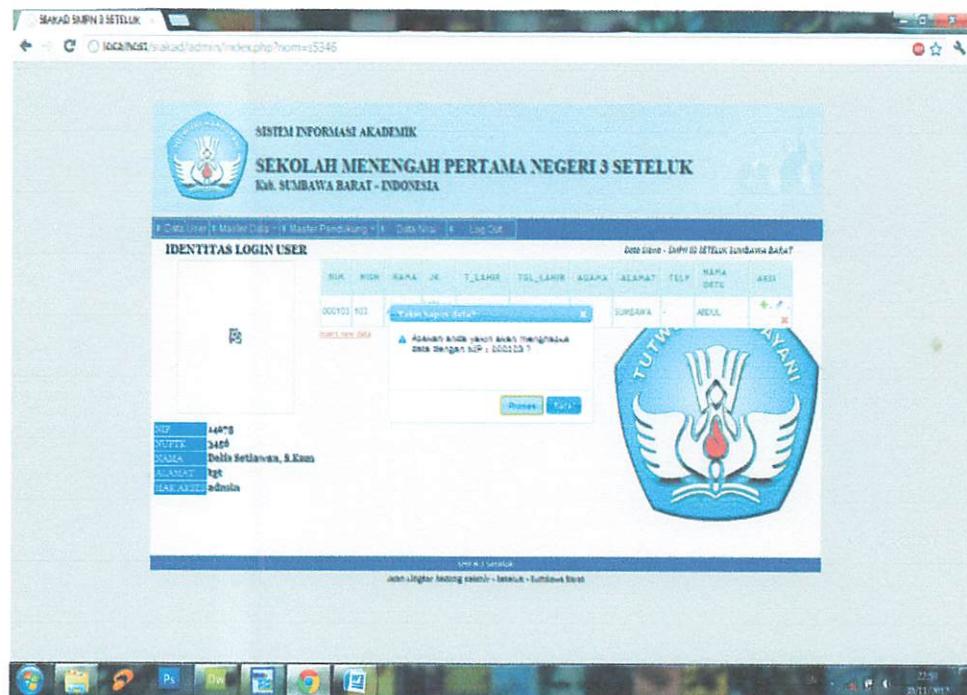


Gambar 4.7 Halaman Aksi *Update Data Siswa*

Tampilan aksi/*action update* data pada gambar 4.7 menampilkan *field – field* data yang perlu diisi oleh *admin* untuk pengubahan data siswa. Pada saat *admin* menekan aksi/*action update* data maka *otomatis* akan keluar *form update* data siswa. Pada *form* ini tiap *field* data akan berisi data siswa lama, sehingga dapat memudahkan *admin* untuk mengubah data tanpa harus melihat data sebelumnya pada *database*. *Form* tersebut terdapat dua pilihan tombol proses dan *cancel*. Tombol proses digunakan untuk memproses data yang ada pada *form* untuk dipindah kedalam *database*. Sedangkan tombol *cancel* digunakan untuk membatalkan perubahan data siswa.

c) Aksi/*action Delete Data*

Aksi/ *action* ini digunakan untuk menghapus data siswa ke dalam *database* siswa. Dengan menggunakan aksi ini *admin* cukup memilih data pada *list* siswa dari halaman data siswa. Aksi ini juga berguna untuk menjaga keamanan data atau mengurangi resiko kesalahan penghapusan data jika dibandingkan menghapus data langsung ke *database*. Berikut ini 4.8 tampilan aksi *delete* data pada aplikasi ini.

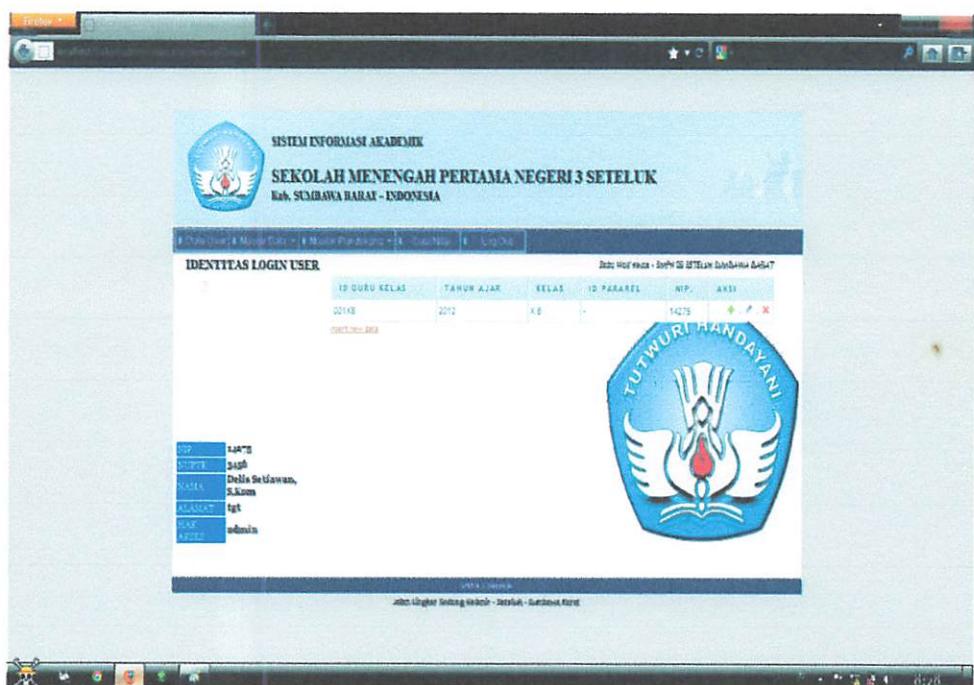


Gambar 4.8 Halaman Aksi Delete Data siswa

Tampilan aksi/*action delete* data pada gambar 4.8 menampilkan pesan menghapus data siswa dengan nomor induk tertentu. Pada saat admin menekan aksi/*action delete* data maka *otomatis* akan keluar pesan peringatan yang menunjukan bahwa data siswa dengan nomor induk tertentu akan dihapus. Jadi dengan adanya pesan tersebut maka *admin* perlu mengecek apakah data yang akan dihapus benar atau salah. Pada pesan tersebut terdapat tombol proses digunakan untuk memproses penghapusan data pada *database*. Sedangkan tombol cancel digunakan untuk membatalkan penghapusan data siswa.

4.2.4 Halaman Data Wali Kelas

Halaman data wali kelas adalah salah satu menu yang disediakan kepada *admin*, tujuannya adalah agar *admin* dapat mendata wali kelas sebagai data master pada aplikasi siakad ini. Halaman data wali kelas berisi *list* wali kelas dan menu aksi/*action* pada *list* tersebut, *action* pada *list* terdiri dari *insert data*, *update data* dan *delete data*. Berikut ini gambar 4.9 adalah tampilan halaman data wali kelas.

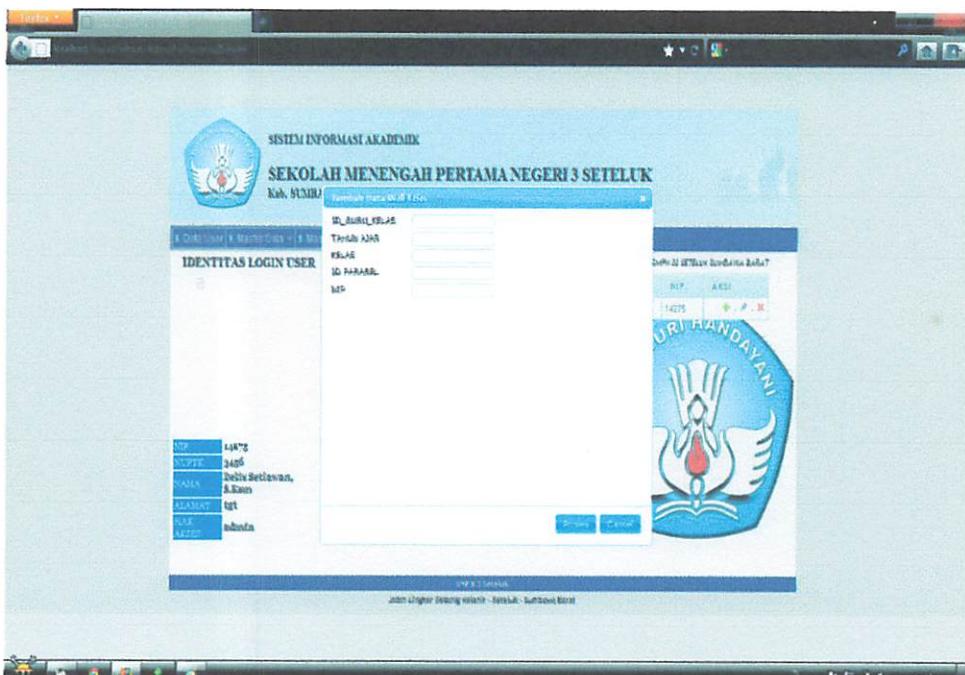


Gambar 4.9 Halaman Data Wali Kelas

Seperti penjelasan sebelumnya pada halaman data wali kelas terdapat aksi atau *action* untuk memanipulasi data wali kelas. *Action* tersebut adalah *insert data*, *update data* dan *delete data*, tujuannya adalah agar *admin* dapat memanipulasi data dengan mudah dan lebih aman jika dibandingkan dengan mengubah data dari *database* langsung. Berikut ini penjelasan dari tiap aksi atau *action* pada halaman data wali kelas.

a) **Aksi/action Insert Data**

Aksi/ *action* ini digunakan untuk menambahkan data wali kelas ke dalam database wali kelas. Dengan menggunakan aksi ini *admin* cukup mengisikan data pada setiap *field* yang ada pada *database*. Aksi ini juga berguna untuk menjaga keamanan data atau mengurangi resiko kesalahan penginputan data langsung ke *database*. Berikut ini 4.10 tampilan aksi *insert data* pada aplikasi ini.

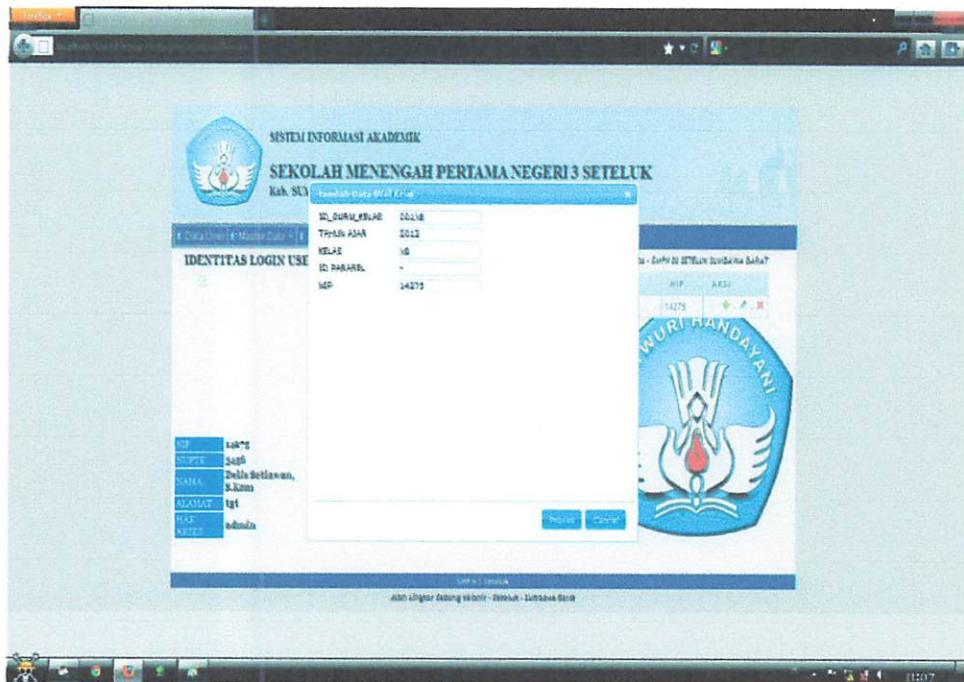


Gambar 4.10 Halaman Aksi Insert Data Wali Kelas

Tampilan aksi/*action insert* data pada gambar 4.10 menampilkan *field-field* data yang perlu diisi oleh admin untuk pendataan wali kelas. Pada saat *admin* menekan aksi/*action insert* data maka *otomatis* akan keluar *form insert* data wali kelas, *form* tersebut terdapat dua pilihan tombol proses dan *cancel*. Tombol proses digunakan untuk memproses data yang ada pada form untuk dipindah ke dalam *database*. Sedangkan tombol *cancel* digunakan untuk membatalkan penambahan data atau *insert* data.

b) Aksi/*action Update Data*

Aksi/ *action* ini digunakan untuk mengupdate atau mengedit data siswa ke dalam *database* siswa. Dengan menggunakan aksi ini *admin* cukup mengisikan data perubahan pada setiap *field* yang ada pada *database*. Aksi ini juga berguna untuk menjaga keamanan data atau mengurangi resiko kesalahan pengubahan data langsung ke *database*. Berikut ini 4.11 tampilan aksi *update* data pada aplikasi ini.

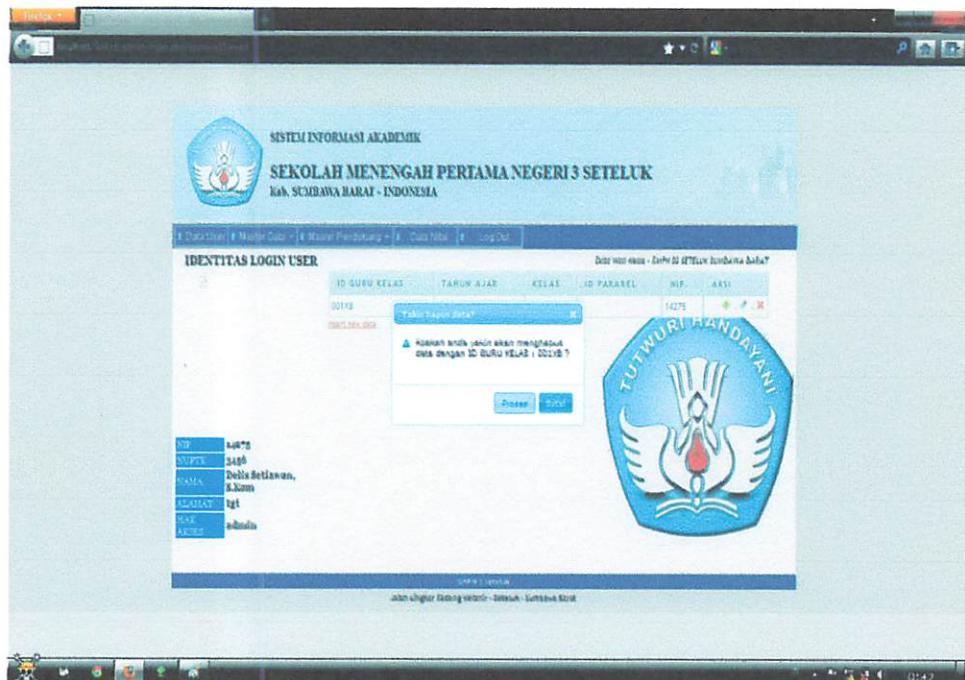


Gambar 4.11 Halaman Aksi *Update Data Wali Kelas*

Tampilan aksi/*action update* data pada gambar 4.11 menampilkan *field – field* data yang perlu diisi oleh admin untuk pengubahan data wali kelas. Pada saat *admin* menekan aksi/*action update* data maka otomatis akan keluar form update data siswa. Pada *form* ini tiap *field* data akan berisi data wali kelas lama, sehingga dapat memudahkan admin untuk mengubah data tanpa harus melihat data sebelumnya pada database. *Form* tersebut terdapat dua pilihan tombol *proses* dan *cancel*. Tombol *proses* digunakan untuk memproses data yang ada pada *form* untuk dipindah kedalam database. Sedangkan tombol *cancel* digunakan untuk membatalkan perubahan data siswa.

c) Aksi/*action Delete Data*

Aksi/ *action* ini digunakan untuk menghapus data wali kelas ke dalam *database* wali kelas. Dengan menggunakan aksi ini *admin* cukup memilih data pada *list* wali kelas dari halaman data wali kelas. Aksi ini juga berguna untuk menjaga keamanan data atau mengurangi resiko kesalahan penghapusan data jika dibandingkan menghapus data langsung ke *database*. Berikut ini 4.12 tampilan aksi *delete* data pada aplikasi ini

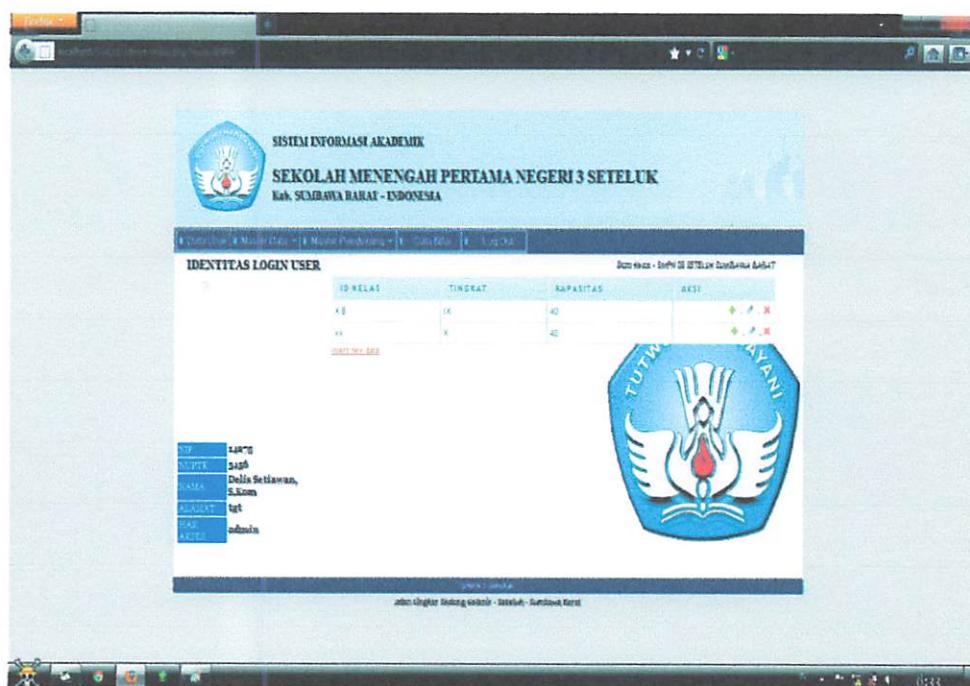


Gambar 4.12 Halaman Aksi *Delete* Data Wali Kelas

Tampilan aksi/*action delete* pada gambar 4.12 menampilkan pesan menghapus data wali kelas dengan ID guru kelas tertentu. Pada saat *admin* menekan aksi/*action delete* data maka *otomatis* akan keluar pesan peringatan yang menunjukkan bahwa data wali kelas dengan ID guru kelas tertentu akan dihapus. Jadi dengan adanya pesan tersebut maka *admin* perlu mengecek apakah data yang akan dihapus benar atau salah. Pada pesan tersebut terdapat tombol proses digunakan untuk memproses penghapusan data pada *database*. Sedangkan tombol *cancel* digunakan untuk membatalkan penghapusan data wali kelas.

4.2.5 Halaman Data Kelas

Halaman data kelas adalah salah satu menu yang disediakan kepada *admin*, tujuannya adalah agar *admin* dapat mendata kelas dan jumlah murid pada setiap masing-masing kelas dari kelas 1 hingga kelas 3 yang ada pada SMPN 3 Seteluk sebagai data master pada aplikasi siakad ini. Halaman data kelas berisi *list* kelas dan menu aksi/*action* pada *list* tersebut, *action* pada *list* terdiri dari *insert data*, *update data* dan *delete data*. Berikut ini gambar 4.13 adalah tampilan halaman data kelas.

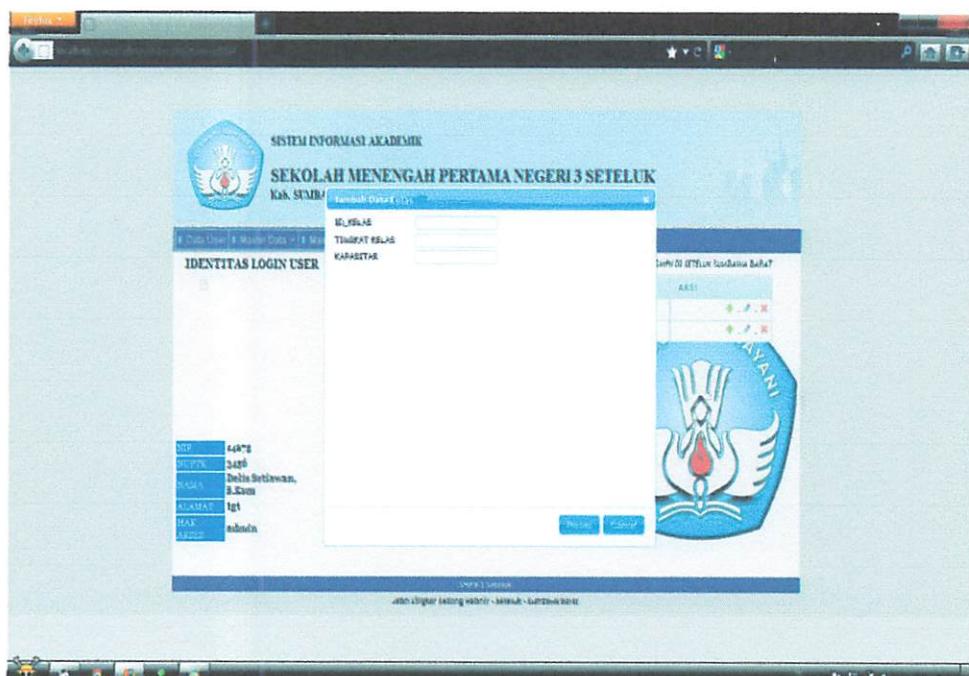


Gambar 4.13 Halaman Data Kelas

Seperti penjelasan sebelumnya pada halaman data kelas terdapat aksi atau *action* untuk memanipulasi data kelas. *Action* tersebut adalah *insert data*, *update data* dan *delete data*, tujuannya adalah agar *admin* dapat memanipulasi data dengan mudah dan lebih aman jika dibandingkan dengan mengubah data dari *database* langsung. Berikut ini penjelasan dari tiap aksi atau *action* pada halaman data kelas.

a) **Aksi/action Insert Data**

Aksi/ *action* ini digunakan untuk menambahkan data kelas ke dalam *database* kelas. Dengan menggunakan aksi ini admin cukup mengisikan data pada setiap *field* yang ada pada *database*. Aksi ini juga berguna untuk menjaga keamanan data atau mengurangi resiko kesalahan penginputan data langsung ke *database*. Berikut ini 4.14 tampilan aksi *insert* data pada aplikasi ini.

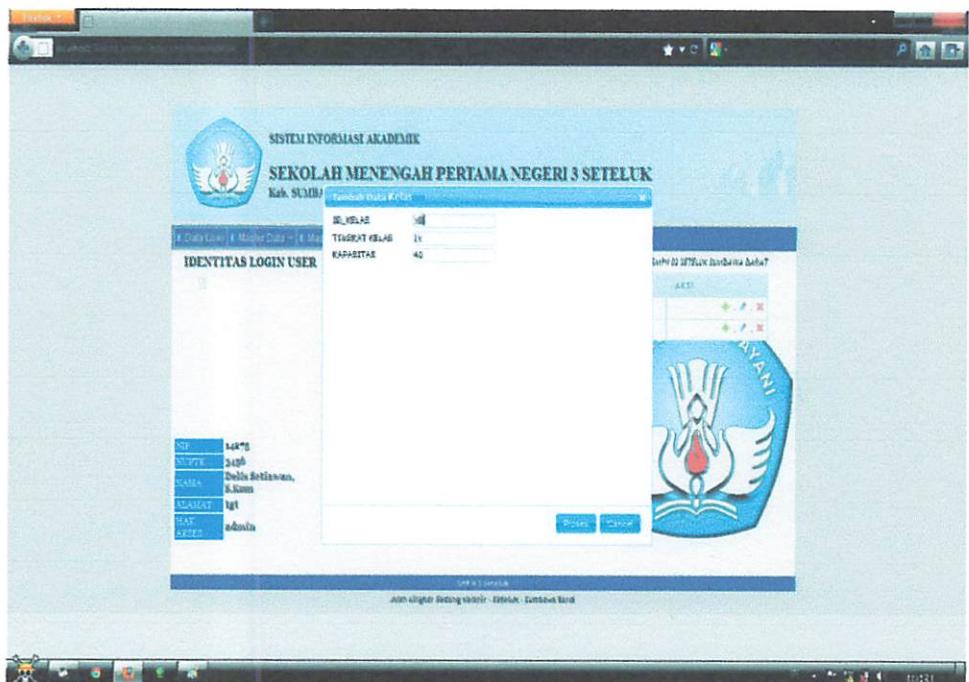


Gambar 4.14 Halaman Aksi *Insert* Data Kelas

Tampilan aksi/*action insert* data pada gambar 4.14 menampilkan *field – field* data yang perlu diisi oleh *admin* untuk pendaftaran kelas. Pada saat *admin* menekan aksi/*action insert* data maka *otomatis* akan keluar *form insert* data kelas, *form* tersebut terdapat dua pilihan tombol proses dan *cancel*. Tombol proses digunakan untuk memproses data yang ada pada *form* untuk dipindah ke dalam *database*. Sedangkan tombol *cancel* digunakan untuk membatalkan penambahan data atau *insert* data.

b) **Aksi/*action Update* Data**

Aksi/*action* ini digunakan untuk mengupdate atau mengedit data kelas ke dalam *database* kelas. Dengan menggunakan aksi ini *admin* cukup mengisikan data perubahan pada setiap *field* yang ada pada *database*. Aksi ini juga berguna untuk menjaga keamanan data atau mengurangi resiko kesalahan pengubahan data langsung ke *database*. Berikut ini 4.15 tampilan aksi *update* data pada aplikasi ini.

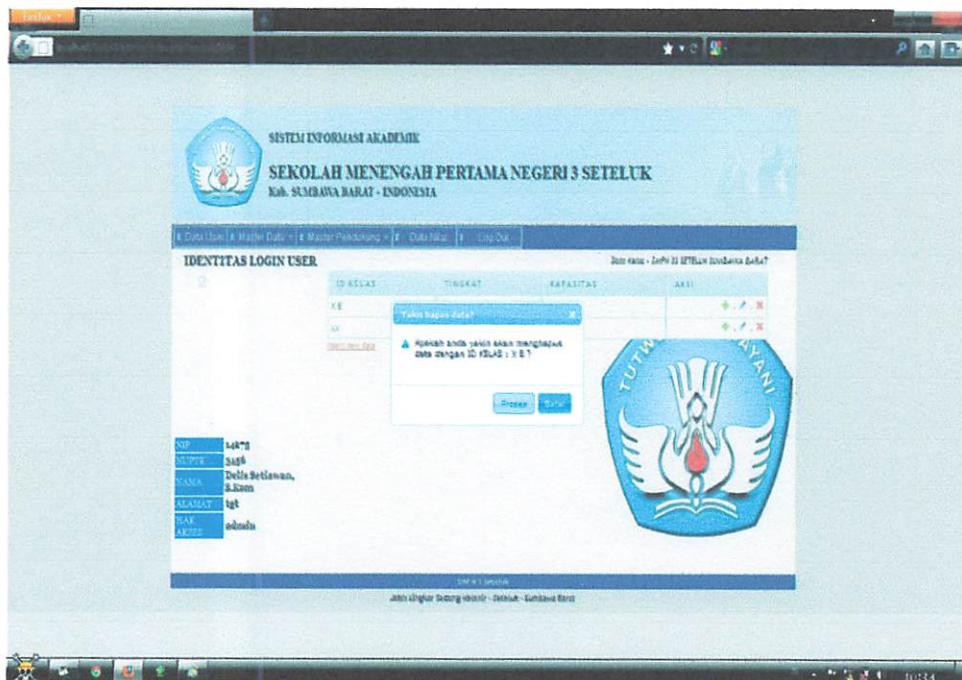


Gambar 4.15 Halaman Aksi *Update Data Kelas*

Tampilan aksi/*action update* data pada gambar 4.15 menampilkan *field – field* data yang perlu diisi oleh *admin* untuk pengubahan data kelas. Pada saat admin menekan aksi/*action update* data maka *otomatis* akan keluar *form update* data kelas. Pada *form* ini tiap *field* data akan berisi data kelas, sehingga dapat memudahkan admin untuk mengubah data tanpa harus melihat data sebelumnya pada *database*. *Form* tersebut terdapat dua pilihan tombol *proses* dan *cancel*. Tombol *proses* digunakan untuk memproses data yang ada pada *form* untuk dipindah kedalam *database*. Sedangkan tombol *cancel* digunakan untuk membatalkan perubahan data kelas.

c) Aksi/*action Delete Data*

Aksi/ *action* ini digunakan untuk menghapus data kelas di dalam *database* kelas. Dengan menggunakan aksi ini *admin* cukup memilih data pada *list* kelas dari halaman data kelas. Aksi ini juga berguna untuk menjaga keamanan data atau mengurangi resiko kesalahan penghapusan data jika dibandingkan menghapus data langsung ke *database*. Berikut ini 4.16 tampilan aksi *delete* data pada aplikasi ini

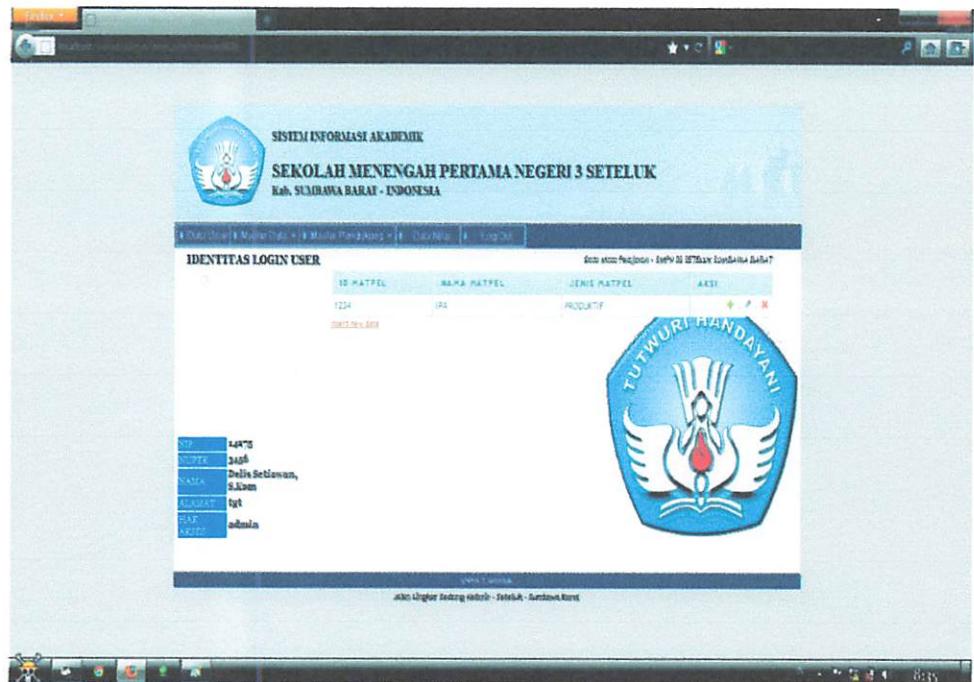


Gambar 4.16 Halaman Aksi Delete Data Kelas

Tampilan aksi/*action delete* pada gambar 4.16 menampilkan pesan menghapus data kelas dengan ID kelas tertentu. Pada saat *admin* menekan aksi/*action delete* data maka *otomatis* akan keluar pesan peringatan yang menunjukan bahwa data kelas dengan ID kelas tertentu akan dihapus. Jadi dengan adanya pesan tersebut maka *admin* perlu mengecek apakah data yang akan dihapus benar atau salah. Pada pesan tersebut terdapat tombol proses digunakan untuk memproses penghapusan data pada database. Sedangkan tombol *cancel* digunakan untuk membatalkan penghapusan data kelas.

4.2.6 Halaman Data Mata Pelajaran

Halaman data mata pelajaran adalah salah satu menu yang disediakan kepada *admin*, tujuannya adalah agar *admin* dapat mendata mata pelajaran dan jenis mata pelajaran sebagai data master pada aplikasi siakad ini. Halaman data mata pelajaran berisi *list* mata pelajaran dan menu aksi/*action* pada *list* tersebut, *action* pada *list* terdiri dari *insert data*, *update data* dan *delete data*. Berikut ini gambar 4.17 adalah tampilan halaman data mata pelajaran.

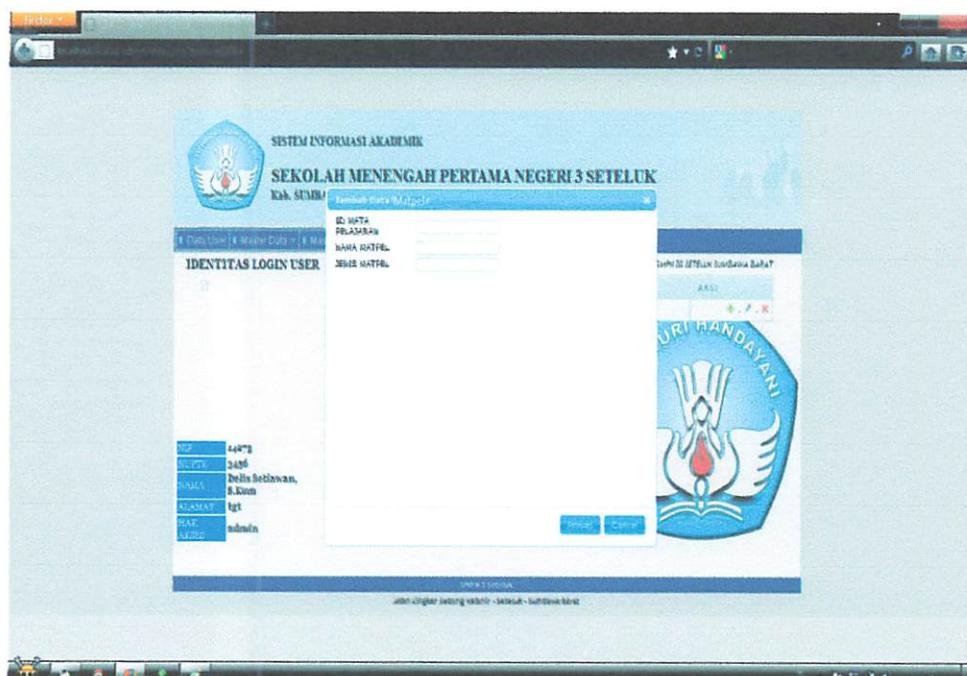


Gambar 4.17 Halaman Data Mata Pelajaran

Seperti penjelasan sebelumnya pada halaman data mata pelajaran terdapat aksi atau *action* untuk memanipulasi data mata pelajaran. *Action* tersebut adalah *insert data*, *update data* dan *delete data*, tujuannya adalah agar *admin* dapat memanipulasi data dengan mudah dan lebih aman jika dibandingkan dengan mengubah data dari *database* langsung. Berikut ini penjelasan dari tiap aksi atau *action* pada halaman data mata pelajaran.

a) **Aksi/action Insert Data**

Aksi/ *action* ini digunakan untuk menambahkan data mata pelajaran ke dalam *database* mata pelajaran. Dengan menggunakan aksi ini *admin* cukup mengisikan data pada setiap *field* yang ada pada *database*. Aksi ini juga berguna untuk menjaga keamanan data atau mengurangi resiko kesalahan penginputan data langsung ke *database*. Berikut ini tampilan aksi *insert data* pada aplikasi ini.

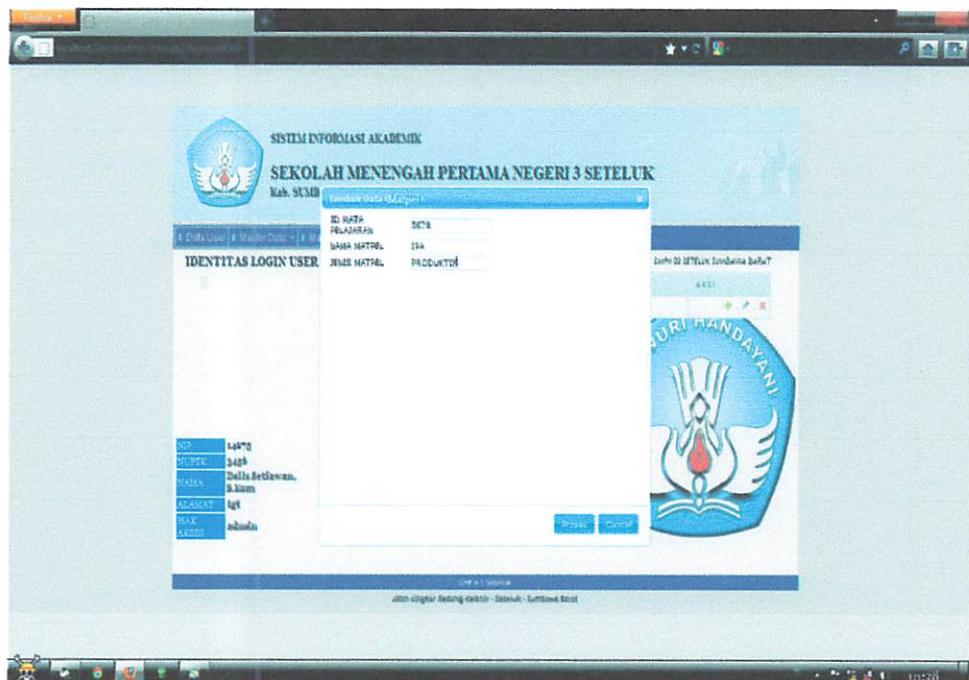


Gambar 4.18 Halaman Aksi *Insert* Data Mata Pelajaran

Tampilan aksi/*action insert* data pada gambar 4.18 menampilkan *field – field* data yang perlu diisi oleh *admin* untuk pendataan mata pelajaran. Pada saat admin menekan aksi/*action insert* data maka *otomatis* akan keluar *form insert* data matpel, *form* tersebut terdapat dua pilihan tombol proses dan *cancel*. Tombol proses digunakan untuk memproses data yang ada pada *form* untuk dipindah ke dalam *database*. Sedangkan tombol *cancel* digunakan untuk membatalkan penambahan data atau *insert* data.

b) Aksi/*action Update Data*

Aksi/ *action* ini digunakan untuk mengupdate atau mengedit data mata pelajaran ke dalam *database* mata pelajaran. Dengan menggunakan aksi ini *admin* cukup mengisikan data perubahan pada setiap *field* yang ada pada *database*. Aksi ini juga berguna untuk menjaga keamanan data atau mengurangi resiko kesalahan pengubahan data langsung ke *database*. Berikut ini 4.19 tampilan aksi *update* data pada aplikasi ini.

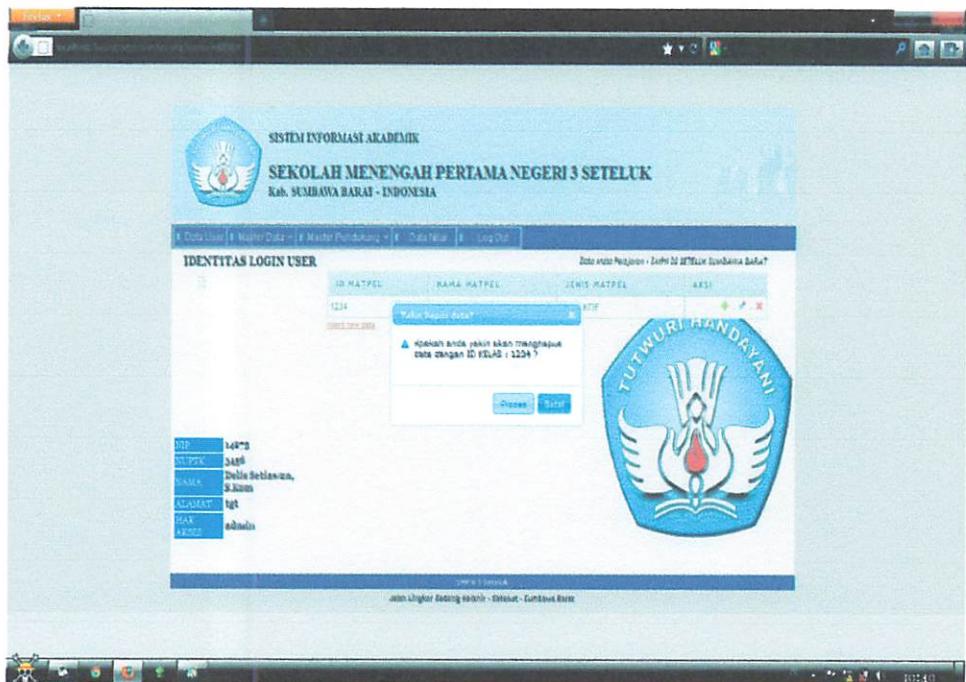


Gambar 4.19 Halaman Aksi Update Data Mata Pelajaran

Tampilan aksi/*action update* data pada gambar 4.19 menampilkan *field – field* data yang perlu diisi oleh *admin* untuk pengubahan data mata pelajaran. Pada saat *admin* menekan aksi/*action update* data maka otomatis akan keluar form update data mata pelajaran. Pada *form* ini tiap *field* data akan berisi data mata pelajaran, sehingga dapat memudahkan *admin* untuk mengubah data tanpa harus melihat data sebelumnya pada *database*. *Form* tersebut terdapat dua pilihan tombol *proses* dan *cancel*. Tombol *proses* digunakan untuk memproses data yang ada pada *form* untuk dipindah kedalam *database*. Sedangkan tombol *cancel* digunakan untuk membatalkan perubahan data mata pelajaran.

c) Aksi/*action Delete Data*

Aksi/ *action* ini digunakan untuk menghapus data mata pelajaran di dalam *database* mata pelajaran. Dengan menggunakan aksi ini *admin* cukup memilih data pada *list* mata pelajaran dari halaman data mata pelajaran. Aksi ini juga berguna untuk menjaga keamanan data atau mengurangi resiko kesalahan penghapusan data jika dibandingkan menghapus data langsung ke *database*. Berikut ini 4.20 tampilan aksi *delete* data pada aplikasi ini



Gambar 4.20 Halaman Aksi *Delete* Data Mata Pelajaran

Tampilan aksi/*action delete* data pada gambar 4.20 menampilkan pesan menghapus data kelas dengan ID matpel tertentu. Pada saat *admin* menekan aksi/*action delete* data maka *otomatis* akan keluar pesan peringatan yang menunjukan bahwa data mata pelajaran dengan ID matpel tertentu akan dihapus. Jadi dengan adanya pesan tersebut maka *admin* perlu mengecek apakah data yang akan dihapus benar atau salah. Pada pesan tersebut terdapat tombol proses digunakan untuk memproses penghapusan data pada *database*. Sedangkan tombol *cancel* digunakan untuk membatalkan penghapusan data matpel.

4.2.7 Halaman Data Matpel – Kelas

Halaman data matpel - kelas adalah salah satu menu yang disediakan kepada *admin*, tujuannya adalah agar *admin* dapat mendata mata pelajaran sesuai dengan ID kelasnya masing-masing agar mempermudah *admin* dalam membuat jadwal pelajaran dan sebagai data master pada aplikasi siakad ini. Halaman data mata pelajaran berisi list mata pelajaran dan menu aksi/*action* pada *list* tersebut, *action* pada *list*

terdiri dari *insert data*, *update data* dan *delete data*. Berikut ini gambar 4.21 adalah tampilan halaman data matpel - kelas.

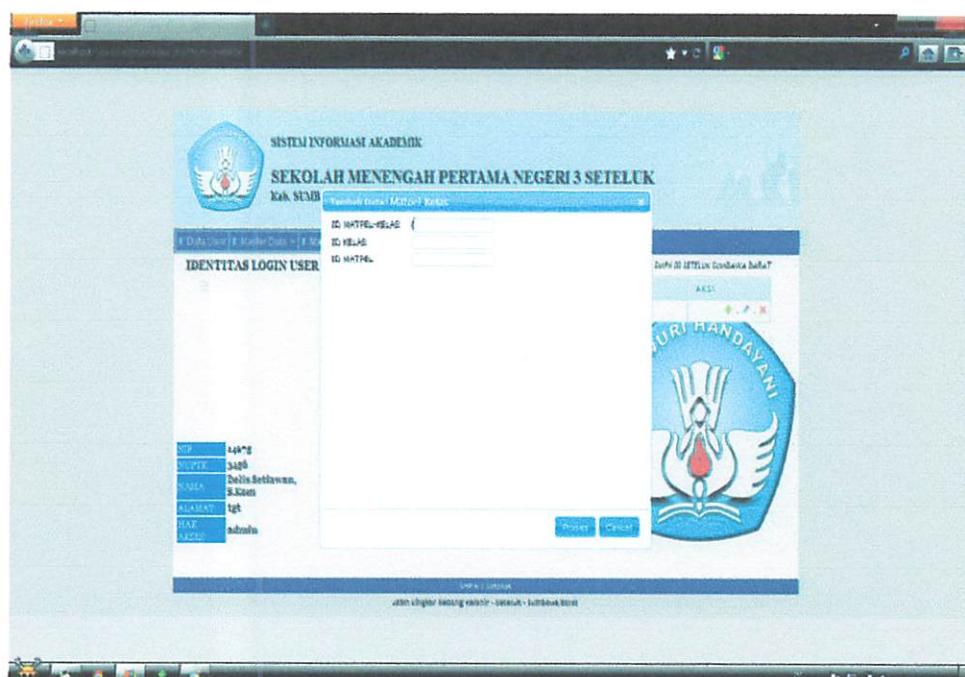


Gambar 4.21 Halaman Data Matpel-Kelas

Seperti penjelasan sebelumnya pada halaman data matpel-kelas terdapat aksi atau *action* untuk memanipulasi data matpel-kelas. *Action* tersebut adalah *insert data*, *update data* dan *delete data*, tujuannya adalah agar *admin* dapat memanipulasi data dengan mudah dan lebih aman jika dibandingkan dengan mengubah data dari *database* langsung. Berikut ini penjelasan dari tiap aksi atau *action* pada halaman data matpel-kelas.

a) **Aksi/action Insert Data**

Aksi/ *action* ini digunakan untuk menambahkan data mata pelajaran ke dalam database matpel-kelas. Dengan menggunakan aksi ini *admin* cukup mengisikan data pada setiap *field* yang ada pada *database*. Aksi ini juga berguna untuk menjaga keamanan data atau mengurangi resiko kesalahan penginputan data langsung ke *database*. Berikut ini 4.22 tampilan aksi *insert data* pada aplikasi ini.

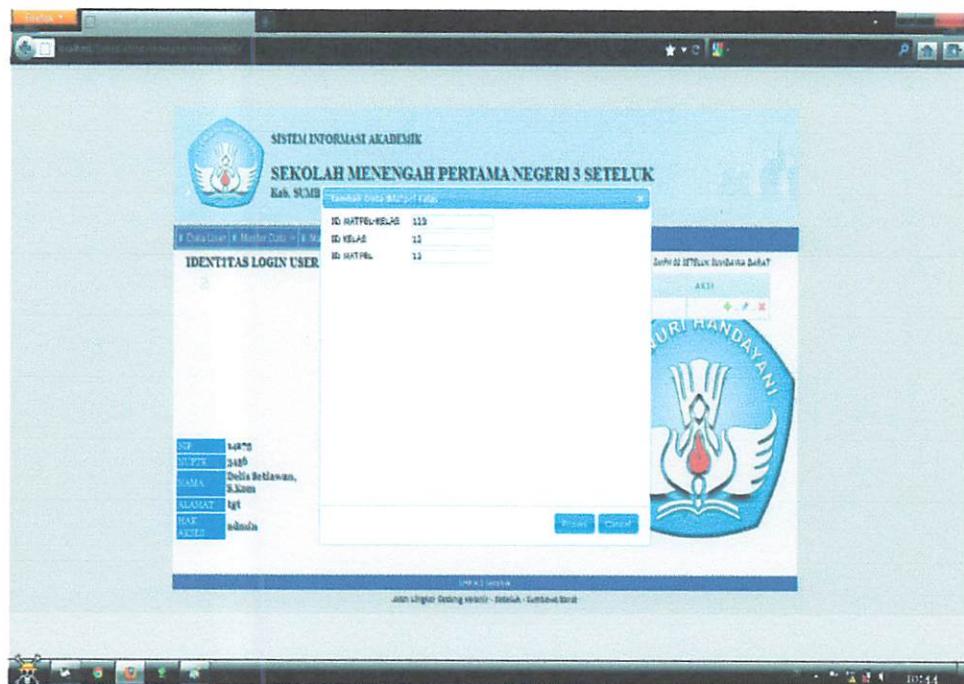


Gambar 4.22 Halaman Aksi *Insert* Data Matpel-Kelas

Tampilan aksi/*action insert* data pada gambar 4.22 menampilkan *field – field* data yang perlu diisi oleh admin untuk pendataan mata pelajaran. Pada saat *admin* menekan aksi/*action insert* data maka *otomatis* akan keluar *form insert* data matpel, *form* tersebut terdapat dua pilihan tombol proses dan *cancel*. Tombol proses digunakan untuk memproses data yang ada pada *form* untuk dipindah ke dalam *database*. Sedangkan tombol *cancel* digunakan untuk membatalkan penambahan data atau *insert* data.

b) **Aksi/*action Update* Data**

Aksi/ *action* ini digunakan untuk mengupdate atau mengedit data mata pelajaran ke dalam *database* matpel-kelas. Dengan menggunakan aksi ini *admin* cukup mengisikan data perubahan pada setiap *field* yang ada pada *database*. Aksi ini juga berguna untuk menjaga keamanan data atau mengurangi resiko kesalahan pengubahan data langsung ke *database*. Berikut ini 4.23 tampilan aksi *update* data pada aplikasi ini.

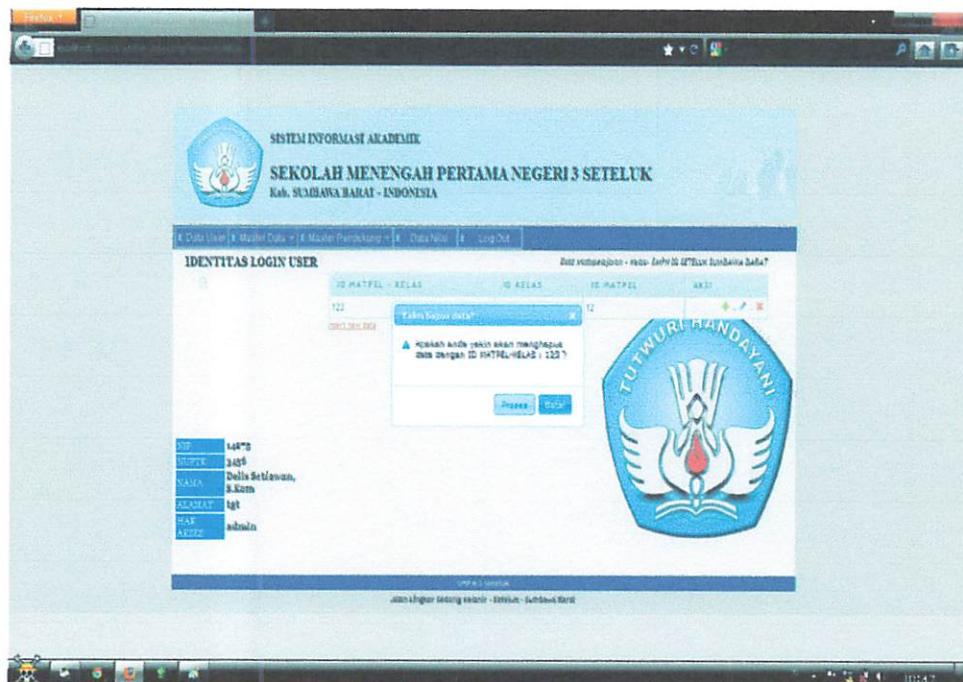


Gambar 4.23 Halaman Aksi *Update* Data Matpel-Kelas

Tampilan aksi/*action update* data pada gambar 4.23 menampilkan *field – field* data yang perlu diisi oleh *admin* untuk pengubahan data mata pelajaran. Pada saat *admin* menekan aksi/*action update* data maka *otomatis* akan keluar *form update* data matpel-kelas. Pada *form* ini tiap *field* data akan berisi data mata pelajaran, sehingga dapat memudahkan *admin* untuk mengubah data tanpa harus melihat data sebelumnya pada *database*. *Form* tersebut terdapat dua pilihan tombol proses dan *cancel*. Tombol proses digunakan untuk memproses data yang ada pada *form* untuk dipindah kedalam *database*. Sedangkan tombol *cancel* digunakan untuk membatalkan perubahan data matpel-kelas.

c) Aksi/*action Delete Data*

Aksi/ *action* ini digunakan untuk menghapus data mata pelajaran di dalam *database* matpel-kelas. Dengan menggunakan aksi ini *admin* cukup memilih data pada *list* matpel-kelas dari halaman data matpel-kelas. Aksi ini juga berguna untuk menjaga keamanan data atau mengurangi resiko kesalahan penghapusan data jika dibandingkan menghapus data langsung ke *database*. Berikut ini 4.24 tampilan aksi *delete* data pada aplikasi ini

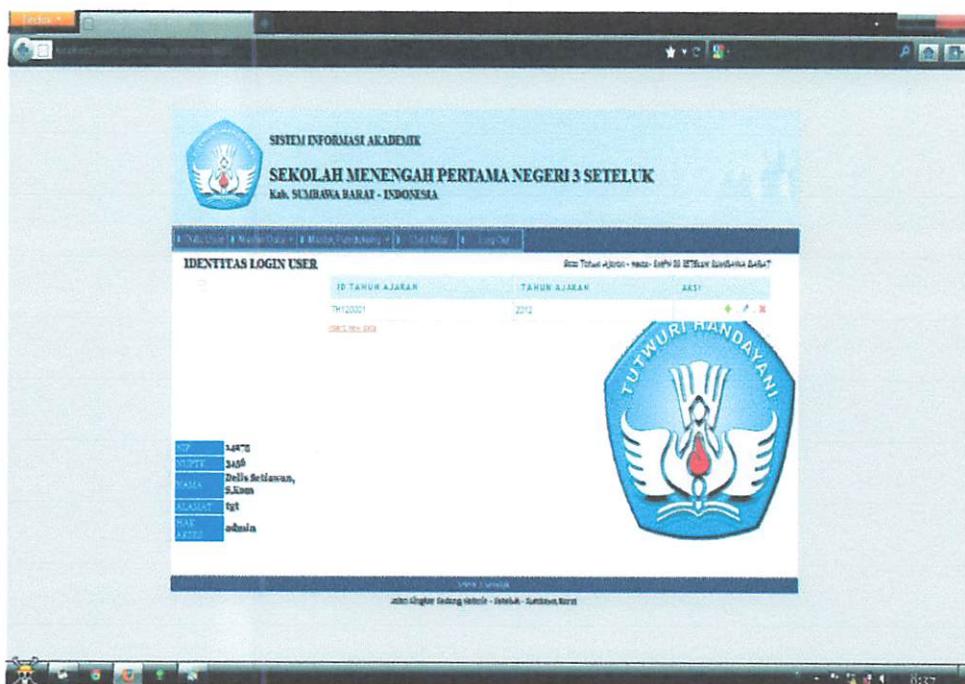


Gambar 4.24 Halaman Aksi Delete Data Matpel-Kelas

Tampilan aksi/*action delete* data pada gambar 4.24 menampilkan pesan menghapus data kelas dengan ID matpel-kelas tertentu. Pada saat *admin* menekan aksi/*action delete* data maka *otomatis* akan keluar pesan peringatan yang menunjukan bahwa data matpel-kelas dengan ID matpel-kelas tertentu akan dihapus. Jadi dengan adanya pesan tersebut maka *admin* perlu mengecek apakah data yang akan dihapus benar atau salah. Pada pesan tersebut terdapat tombol *proses* digunakan untuk memproses penghapusan data pada database. Sedangkan tombol *cancel* digunakan untuk membatalkan penghapusan data matpel-kelas.

4.2.8 Halaman Data Tahun Ajaran

Halaman data tahun ajaran adalah salah satu menu yang disediakan kepada *admin*, tujuannya adalah agar *admin* dapat mengelolah dan mendata murid yang masuk dan lulus sesuai dengan ID tahun ajaran masing-masing sebagai data master pada aplikasi siakad ini. Halaman data tahun ajaran berisi list tahun ajaran dan menu aksi/*action* pada list tersebut, action pada *list* terdiri dari *insert data*, *update data* dan *delete data*. Berikut ini gambar 4.25 adalah tampilan halaman data tahun ajaran.

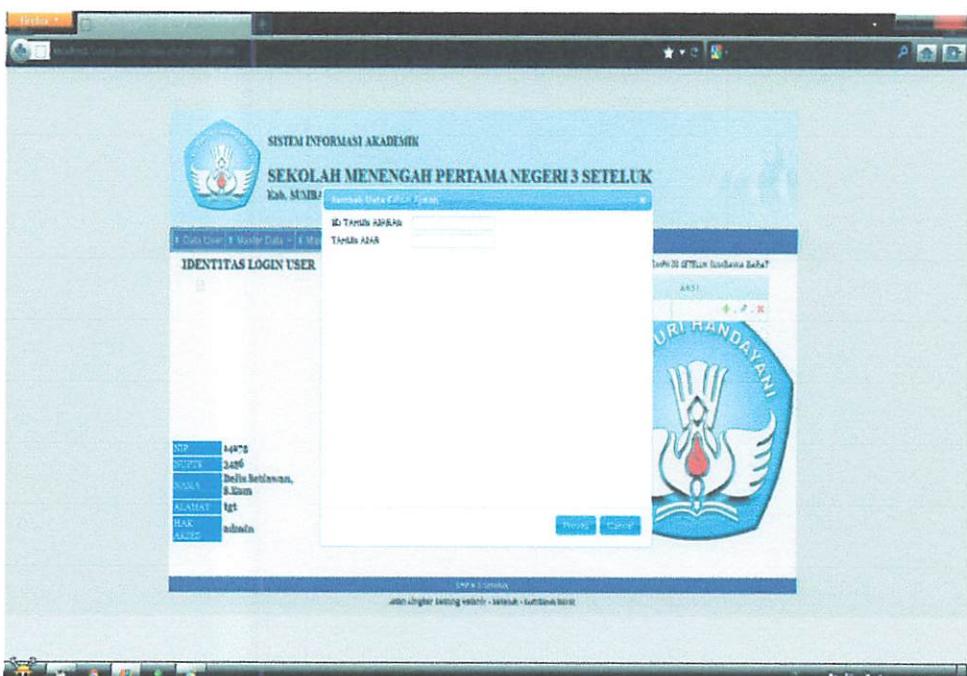


Gambar 4.25 Halaman Data Tahun Ajaran

Seperti penjelasan sebelumnya pada halaman data tahun ajaran terdapat aksi atau *action* untuk memanipulasi data tahun ajaran. *Action* tersebut adalah *insert data*, *update data* dan *delete data*, tujuannya adalah agar *admin* dapat memanipulasi data dengan mudah dan lebih aman jika dibandingkan dengan mengubah data dari *database* langsung. Berikut ini penjelasan dari tiap aksi atau *action* pada halaman data tahun ajaran.

a) **Aksi/action Insert Data**

Aksi/ *action* ini digunakan untuk menambahkan data tahun ajaran ke dalam *database* tahun ajaran. Dengan menggunakan aksi ini *admin* cukup mengisikan data pada setiap *field* yang ada pada *database*. Aksi ini juga berguna untuk menjaga keamanan data atau mengurangi resiko kesalahan penginputan data langsung ke *database*. Berikut ini 4.26 tampilan aksi *insert data* pada aplikasi ini.

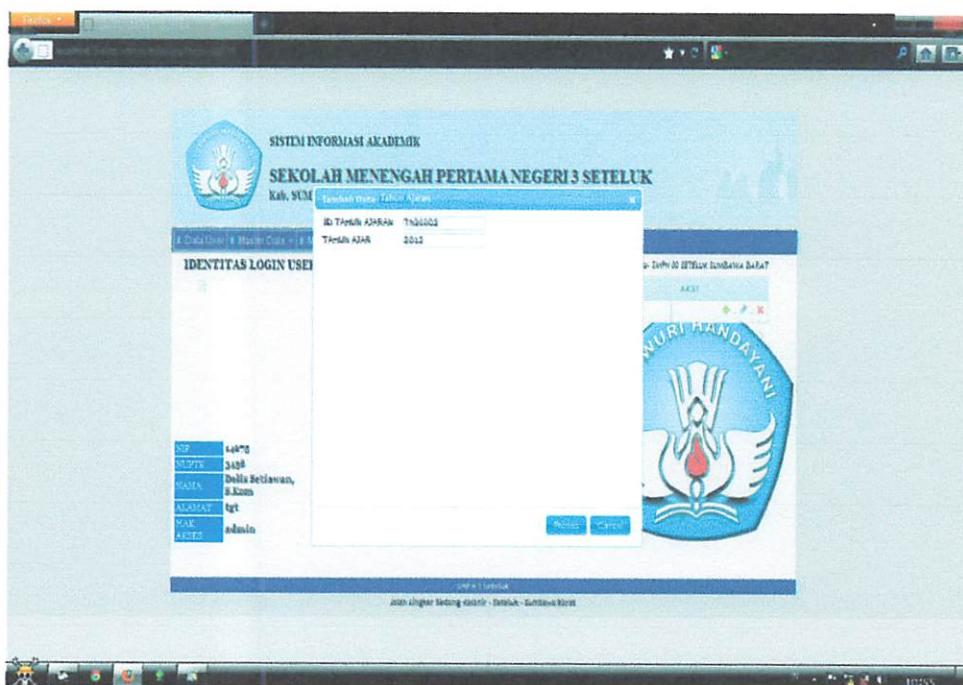


Gambar 4.26 Halaman Aksi *Insert* Data Tahun Ajaran

Tampilan aksi/*action insert* data pada gambar 4.26 menampilkan *field – field* data yang perlu diisi oleh admin untuk pendaftaran tahun ajaran. Pada saat *admin* menekan aksi/*action insert* data maka *otomatis* akan keluar *form insert* data tahun ajaran, *form* tersebut terdapat dua pilihan tombol proses dan *cancel*. Tombol proses digunakan untuk memproses data yang ada pada *form* untuk dipindah ke dalam *database*. Sedangkan tombol *cancel* digunakan untuk membatalkan penambahan data atau *insert data*.

b) Aksi/*action Update Data*

Aksi/ *action* ini digunakan untuk mengupdate atau mengedit data tahun ajaran ke dalam *database* tahun ajaran. Dengan menggunakan aksi ini *admin* cukup mengisikan data perubahan pada setiap *field* yang ada pada *database*. Aksi ini juga berguna untuk menjaga keamanan data atau mengurangi resiko kesalahan pengubahan data langsung ke *database*. Berikut ini 4.27 tampilan aksi *update* data pada aplikasi ini.

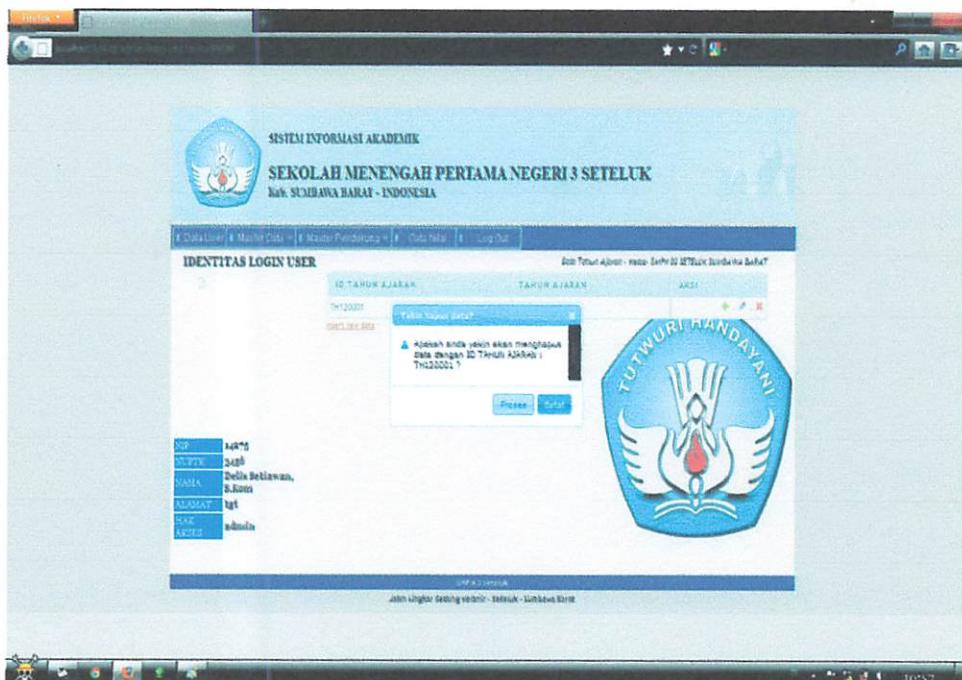


Gambar 4.27 Halaman Aksi *Update* Data Tahun Ajaran

Tampilan aksi/*action update* data pada gambar 4.27 menampilkan *field – field* data yang perlu diisi oleh *admin* untuk pengubahan data tahun ajaran. Pada saat *admin* menekan aksi/*action update* data maka *otomatis* akan keluar *form update* data tahun ajaran. Pada *form* ini tiap *field* data akan berisi data tahun ajaran, sehingga dapat memudahkan *admin* untuk mengubah data tanpa harus melihat data sebelumnya pada *database*. *Form* tersebut terdapat dua pilihan tombol *proses* dan *cancel*. Tombol *proses* digunakan untuk memproses data yang ada pada *form* untuk dipindah kedalam *database*. Sedangkan tombol *cancel* digunakan untuk membatalkan perubahan data tahun ajaran.

c) Aksi/*action Delete Data*

Aksi/ *action* ini digunakan untuk menghapus data tahun ajaran di dalam *database* tahun ajaran. Dengan menggunakan aksi ini *admin* cukup memilih data pada *list* tahun ajaran dari halaman data tahun ajaran. Aksi ini juga berguna untuk menjaga keamanan data atau mengurangi resiko kesalahan penghapusan data jika dibandingkan menghapus data langsung ke *database*. Berikut ini 4.28 tampilan aksi *delete* data pada aplikasi ini



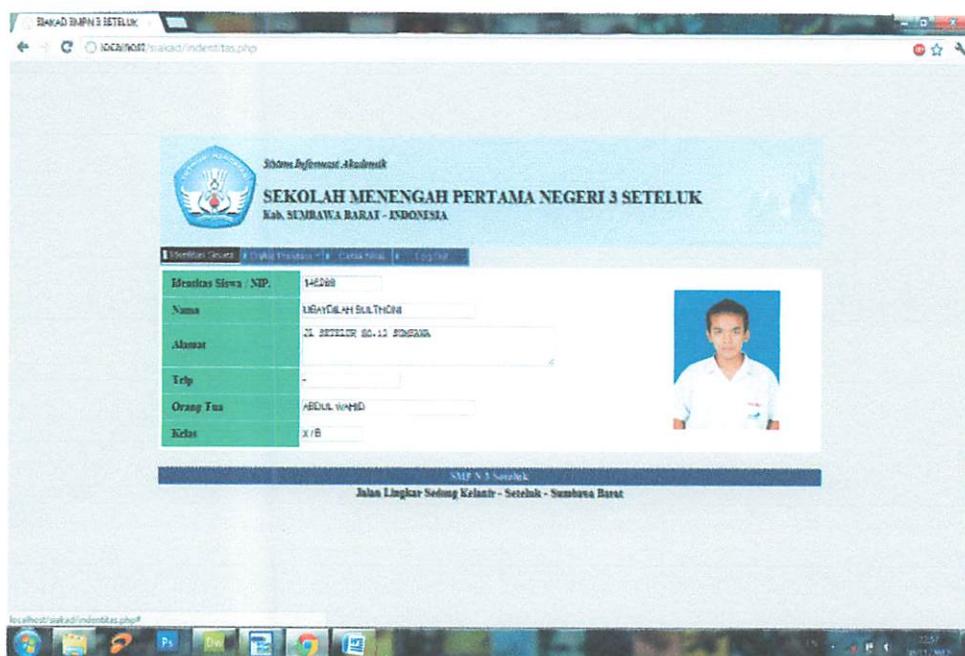
Gambar 4.28 Halaman Aksi *Delete* Data Tahun Ajaran

Tampilan aksi/*action delete* data pada gambar 4.28 menampilkan pesan menghapus data tahun ajaran dengan ID tahun ajaran tertentu. Pada saat *admin* menekan aksi/*action delete* data maka *otomatis* akan keluar pesan peringatan yang menunjukkan bahwa data tahun ajaran dengan ID tahun ajaran tertentu akan dihapus. Jadi dengan adanya pesan tersebut maka *admin* perlu mengecek apakah data yang akan dihapus benar atau salah. Pada pesan tersebut terdapat tombol *proses* digunakan untuk memproses penghapusan data pada *database*. Sedangkan tombol *cancel* digunakan untuk membatalkan penghapusan data tahun ajaran.

4.2.9 Halaman Identitas Siswa

Halaman identitas siswa adalah halaman yang berisi data pribadi siswa. Halaman ini dapat diakses oleh siswa. pada saat *login* siswa dapat memasukan nomor induk dan *password* yang telah di berikan oleh pihak sekolah. Nomor induk dan *password* nantinya akan digunakan sebagai data yang akan dicek oleh sistem apakah nomor induk dan *password* sudah benar, dan siapa pemegang hak akses dari nomor induk tersebut. Jika hak akses tersebut adalah siswa maka akan keluar menu khusus untuk hak

akses siswa. menu tersebut diantaranya adalah menu identitas siswa, indek prestasi siswa, cetak nilai dan *logout*. Pada bahasan kali ini kita akan membahas halaman identitas siswa. pada halaman ini siswa akan mengetahui data siswa pada *database*, siswa juga dapat mengecek apakah data yang telah diinputkan oleh admin sudah benar atau belum. Berikut ini gambar 4.29 menampilkan halaman identitas siswa pada siakad.

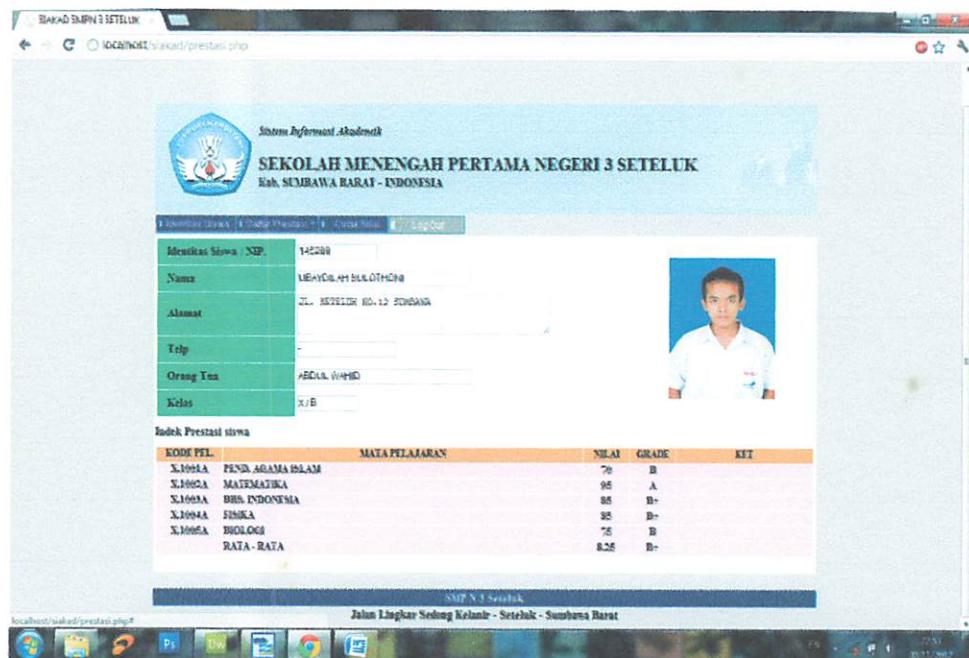


Gambar 4.29 Halaman Identitas Siswa

Pada gambar 4.29 terdapat *form* identitas siswa dan isi data pada tiap *field*. Gambar tersebut juga menampilkan foto siswa dan data – data siswa, jadi dengan adanya identitas tersebut siswa dapat melihat dan mengetahui apakah data pada siakad sudah benar atau belum.

4.2.10 Halaman Prestasi Siswa

Halaman prestasi siswa adalah halaman yang disediakan untuk hak akses siswa. halaman ini berisi identitas siswa dan nilai dari setiap pelajaran sesuai dengan kelas siswa tersebut. Dengan adanya halaman ini siswa dapat mengetahui prestasi siswa pada setiap tahun ajaran. Berikut ini gambar 4.30 halaman prestasi siswa pada siakad.



Gambar 4.30 Halaman Prestasi Siswa

Gambar 4.30 menampilkan halaman prestasi siswa, pada halaman tersebut terdapat identitas siswa dan nilai siswa sesuai dengan tahun ajaran yang baru. Pada nilai siswa terdapat kode pelajaran, mata pelajaran, nilai, *grade* dan keterangan. Kode pelajaran berisi kode pelajaran yang telah diinputkan *admin* kedalam master data pelajaran. Mata pelajaran berisi nama pelajaran dari kode pelajaran tersebut. Nilai berisi nilai dari matapelajaran. *Grade* berisi *konfensi* huruf dari nilai mata pelajaran dan keterangan berisi keterangan yang mungkin saja diinputkan oleh wali kelas.

4.3 Pengujian Sistem

Pengujian sistem merupakan tahap uji coba terhadap sistem yang telah dibuat. Pengujian sistem dilakukan bertujuan untuk mengidentifikasi masalah pada sistem bila terjadi kesalahan pada sistem dan bertujuan untuk mengetahui sejauh mana keberhasilan aplikasi ini bisa diterapkan dalam sistem.

Pengujian perangkat lunak ini menggunakan metode pengujian *Black Box*. Pengujian *Black Box* berfokus pada persyaratan *fungsional* perangkat lunak yang dibuat.

4.3.1 Rencana Pengujian

Rencana pengujian yang akan dilakukan dengan menguji sistem. Pengujian Sistem Informasi Akademik berikut menggunakan data uji berupa masukan dari *admin* dan *user*. Rencana pengujian selengkapnya terlihat pada tabel dibawah ini :

Tabel 4.1 Rencana Pengujian

Kelas Uji	Detail Pengujian	Jenis Pengujian
Login Admin	Verifikasi data login admin dengan memasukkan username dan password beserta pilih hak aksesnya oleh admin.	<i>Black Box</i>
Pengujian pengisian data guru dan siswa	Proses input data guru dan siswa sekaligus proses simpan yang dilakukan oleh admin.	<i>Black Box</i>
Pengujian Pengisian nilai siswa	Proses input nilai siswa sekaligus proses simpan.	<i>Black Box</i>

4.3.2 Pengujian Login

Pengujian *Login admin* yang mempunyai hak akses penuh dalam pengelolaan data. Pengujian *login* ini dilakukan dengan menggunakan *validasi* terhadap data *username* dan *password* dan hak akses yang dimasukan, apabila data masukan *valid* maka *login* sukses tetapi apabila data masukan tidak *valid* maka *login* gagal.

Tabel 4.2 Pengujian *Login Admin*

Kasus dan Hasil Uji Login Admin (Data Normal)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Username, dan Password yang dimasukan benar (valid)	Dapat masuk ke halaman utama administrator	User name dan password sesuai dengan hak akses.	[x] Diterima [] Ditolak

Kasus dan Hasil Uji Login Admin (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Username dan Password yang dimasukan salah (tidak valid)	Tidak dapat login sebagai admin, dan menampilkan pesan “Password anda salah”.	User name dan password tidak sesuai dengan hak akses.	[x] Diterima [] Ditolak

4.3.3 Pengujian Input Data Guru

Pengujian ini adalah pengujian dengan memasukkan sejumlah data guru ke dalam *database*, dengan beberapa *validasi*.

Tabel 4.3 Pengujian *Input* Data Guru

Kasus dan Hasil Uji Input Data Guru (Data Normal)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Klik Laporan Data Guru	Dapat masuk ke halaman data guru	Melihat isi data guru	[x] Diterima [] Ditolak
Klik Tombol Insert	Dapat masuk ke halaman form data guru dan pengisian data guru	Tombol insert sesuai dengan yang diharapkan	[x] Diterima [] Ditolak
Klik tombol Proses	Data yang telah di Inputkan pada kolom form tersimpan di database.	Tombol proses sesuai dengan yang diharapkan	[x] Diterima [] Ditolak

4.3.4 Pengujian input Data Siswa

Pengujian ini adalah pengujian dengan memasukkan sejumlah data siswa ke dalam *database*, dengan beberapa *validasi*.

Tabel 4.4 Pengujian *Input* Data Siswa

Kasus dan Hasil Uji Input Data Siswa (Data Normal)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Klik Laporan Data Guru	Dapat masuk ke halaman data siswa	Melihat isi data siswa	[x] Diterima [] Ditolak
Klik Tombol Insert	Dapat masuk ke halaman form data siswa dan pengisian data siswa	Tombol insert sesuai dengan yang diharapkan	[x] Diterima [] Ditolak
Klik tombol Proses	Data yang telah di Inputkan pada kolom form tersimpan di database.	Tombol proses sesuai dengan yang diharapkan	[x] Diterima [] Ditolak

4.3.5 Pengujian Nilai Siswa

Pengujian ini adalah pengujian dengan memasukkan sejumlah nilai untuk data siswa ke dalam *database*, dengan beberapa *validasi*.

Tabel 4.5 Pengujian Nilai Siswa

Kasus dan Hasil Uji Input Laporan Nilai (Data Normal)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Klik Laporan Nilai	Dapat masuk ke halaman laporan nilai siswa	Melihat isi laporan nilai siswa	[x] Diterima [] Ditolak
Klik Tombol Insert	Dapat masuk ke halaman form laporan nilai siswa dan pengisian nilai siswa	Tombol insert sesuai dengan yang diharapkan	[x] Diterima [] Ditolak
Klik tombol Proses	Data yang telah di Inputkan pada kolom form tersimpan di database.	Tombol proses sesuai dengan yang diharapkan	[x] Diterima [] Ditolak

4.3.6 Kesimpulan Hasil Pengujian

Pengujian yang telah dilakukan merupakan proses yang terdapat dalam sistem informasi akademik berbasis *web*. Dan setelah dilakukan pengujian, maka berdasarkan hasil dari pengujian dapat diambil kesimpulan bahwa aplikasi dapat digunakan dengan baik, disamping terdapat beberapa proses lain yang mendukung dalam berjalannya aplikasi ini. Namun demikian pengujian tersebut di atas dapat dikatakan belum sempurna, dikarenakan hanya dilakukan pada sisi pengujian. Dan semua yang dilakukan dalam pengujian ini diharapkan dapat mewakili pengujian fungsi yang lain dalam sistem informasi akademik berbasis *web*.

BAB V

P E N U T U P

5.1 Kesimpulan

Berdasarkan analisis dan pengujian yang telah dilakukan maka dapat diambil beberapa kesimpulan antara lain:

1. Sistem Informasi akademik dapat membantu dalam pengelolaan data akademik sehingga menghasilkan informasi yang *valid*.
2. Sistem Informasi akademik dapat mempercepat dan mempermudah guru bagian kurikulum dalam pembuatan laporan.
3. Dengan diterapkannya sistem ini diharapkan segala kendala tentang keterlambatan dan ketidak akuratan laporan-laporan yang berhubungan dengan masalah akademik dapat diatasi.
4. Sistem informasi akademik ini dapat membantu proses data siswa, data guru, data pelajaran, data kelas, dan peniaian sehingga lebih optimal.
5. Sistem informasi akademik ini dapat mengurangi kesalahan dalam pegolahan data.

5.2 Saran

Adapun saran-saran yang dapat disampaikan yang berkaitan dengan Program ini adalah sebagai berikut :

1. Perlu adanya pengembangan layanan pada *customer* (orang tua siswa) yang akan melengkapi kemudahan dalam mengakses informasi seperti SMS *gateway*.
2. Aplikasi ini perlu ditambahkan informasi tentang daftar keuangan siswa yang belum dan sudah membayar DPP/SPP sekolah.
3. Sistem informasi akademik yang telah dibangun ini agar dijadikan bahan untuk pengembangan sistem lebih lanjut

DAFTAR PUSTAKA

1. Hakim, Lukmanul. 2010. *Bikin Website Super Keren dengan PHP dan jQuery*. Penerbit Lokomedia, Yogyakarta
2. Hakim, Lukmanul. 2011. *Trik Dahsyat Menguasai Ajax dengan jQuery*. Penerbit Lokomedia, Yogyakarta
3. HM, Jogiyanto. 2001. *Analisa & Desain Sistem Informasi: pendekatan terstruktur teori dan praktik aplikasi bisnis*. Penerbit ANDI Yogyakarta
4. Saputra, Agus dan Agustin, Feni. 2011. *Pemograman CSS untuk Pemula*. PT Elex Media Komputindo, Jakarta
5. Yuana, Rosihan Ari. 2010. *67 Trik dan Ide Brilian Master PHP*. Penerbit Lokomedia, Yogyakarta
6. MySQL official website (www.mysql.com), Diakses tanggal 23 September 2012
7. PHP official website (www.php.net), Diakses tanggal 15 November 2012

LEMBAR PERSEMPAHAN

Dengan Menyebut Nama Allah Yang Maha Pengasih Lagi Maha Penyanyang



"Tugas kita bukanlah untuk berhasil, Tugas kita adalah untuk mencoba, Karena didalam mencoba itulah kita menemukan dan belajar membangun kesempatan untuk berhasil"

-MARIO TEGUH-

KUPERSEMPAHKAN SKRIPSI INI KEPADA:

- ❖ PAPA DAN MAMAKU TERCINTA
- ❖ KELUARGA BESARKU
- ❖ KAWAN-KAWAN JURUSAN
TEKNIK INFORMATIKA
- ❖ ORANG YANG MEMOTIVASIKU
- ❖ ALMAMATERKU

www.deliskutriawansamantha.wordpress.com

LAMPIRAN



INSTITUT TEKNOLOGI NASIONAL MALANG
Fakultas Teknologi Industri
Program Studi Teknik Informatika S1
Jl. Raya Karanglo Km. 2 Malang

FORMULIR PERBAIKAN SKRIPSI

Nama : Delis Kurniawan Samantha
NIM : 08.18.078
Prodi : Teknik Informatika S-1
Judul : Sistem Informasi Akademik SMPN 3 Seteluk

Tanggal	Penguji	Uraian
15 Februari 2013	I	<ul style="list-style-type: none">- Nilai di inputkan oleh guru tidak muncul di laporan nilai siswa- Pengujian browser (Mozilla firefox, Opera, Internet explorer, Google chrome)- Tidak bisa menambahkan matpel pada siswa
15 Februari 2013	II	<ul style="list-style-type: none">- Tambahkan teori sistem informasi akademik pada BAB II- Tambahkan DFD dan ERD- Revisi kesimpulan dan saran- Tambahkan Pengujian user pada BAB IV

Anggota Penguji :

Dosen Penguji I

Ali Mahmudi, BEng, PhD.
NIP. 1031000429

Dosen Penguji II

Sandy Nataly Mantja, Skom.
NIP. 1030800418

Mengetahui

Dosen Pembimbing I

Dr. Ir. Dhayal Gustopo, MT.
NIP. 0713086103

Dosen Pembimbing II

Ahmad Faisol, ST.
NIP. 1031000431



INSTITUT TEKNOLOGI NASIONAL MALANG
Fakultas Teknologi Industri
Program Studi Teknik Informatika S1
Jl. Raya Karanglo Km. 2 Malang

FORMULIR BIMBINGAN SKRIPSI

Nama : Delis Kurniawan Samantha
Nim : 08.18.078
Masa Bimbingan : 29 Oktober s/d 29 April 2013
Judul Skripsi : Sistem Informasi Akademik SMPN 3 Seteluk

NO	TANGGAL	URAIAN	PARAF PEMBIMBING
1	06-06-2012	Acc BAB I-II, Revisi BAB III	
2	13-06-2012	Acc BAB III	
3	12-01-2013	Revisi BAB IV	
4	16-01-2013	Acc Makalah Seminar Hasil	
5	31-01-2013	Acc Kompre	
6			
7			
8			
9			
10			

Malang, 01 Maret 2013
Dosen Pembimbing II

Ahmad Faisol, ST.
NIP. 1031000431



INSTITUT TEKNOLOGI NASIONAL MALANG
Fakultas Teknologi Industri
Program Studi Teknik Informatika S1
Jl. Raya Karanglo Km. 2 Malang

FORMULIR BIMBINGAN SKRIPSI

Nama : Delis Kurniawan Samantha
Nim : 08.18.078
Masa Bimbingan : 29 Oktober s/d 29 April 2013
Judul Skripsi : Sistem Informasi Akademik SMPN 3 Seteluk

NO	TANGGAL	URAIAN	PARAF PEMBIMBING
1	22-05-2012	Revisi BAB I-II	
2	11-06-2012	Acc BAB I-II, Revisi BAB III	
3	05-07-2012	Acc BAB III-IV	
4	04-12-2012	Revisi BAB V	
5	16-01-2013	Acc Makalah Seminar Hasil	
6	11-02-2013	Acc Kompre	
7			
8			
9			
10			

Malang, 01 Maret 2013

Dosen Pembimbing I

Dr. Ir. Dhayal Gustopo S, MT
NIP. 0713086103



PERKUMPULAN PENGELOLA PENDIDIKAN UMUM DAN TEKNOLOGI NASIONAL MALANG

INSTITUT TEKNOLOGI NASIONAL MALANG

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

BNI (PERSERO) MALANG
BANK NIAGA MALANG

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

Nomor : ITN-98/T.Inf/TA/2012
Lampiran : -
Perihal : Bimbingan Skripsi

29 Oktober 2012

Kepada : Yth. Sdr. Dr. Ir. Dhayal Gustopo S, MT
Dosen Pembimbing Program Studi Teknik Informatika S1
Institut Teknologi Nasional
M a l a n g

Dengan hormat

Sesuai dengan permohonan dan persetujuan dalam Proposal Skripsi untuk mahasiswa :

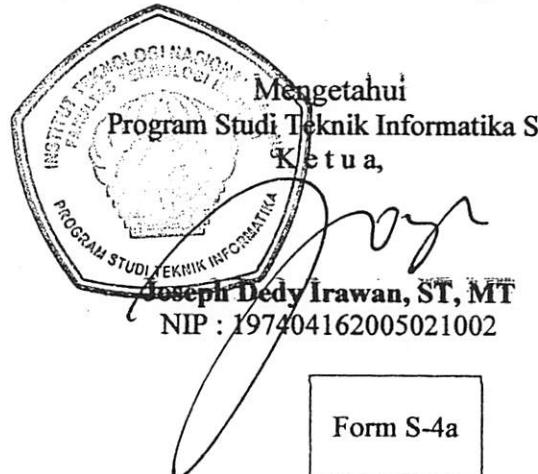
Nama : Delis Kurniawan
Nim : 0818078
Prodi : Teknik Informatika S1
Fakultas : Teknologi Industri

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

29 Oktober 2012 s/d 29 April 2013

Sebagai satu syarat untuk menempuh Ujian Sarjana Teknik, Program Studi Teknik Informatika S1.

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



Form S-4a



PERKUMPULAN PENGELOLA PENDIDIKAN UMUM DAN TEKNOLOGI NASIONAL MALANG
INSTITUT TEKNOLOGI NASIONAL MALANG
FAKULTAS TEKNOLOGI INDUSTRI
FAKULTAS TEKNIK SIPIL DAN PERENCANAAN
PROGRAM PASCASARJANA MAGISTER TEKNIK

BNI (PERSERO) MALANG
BANK NIAGA MALANG

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

Nomor : ITN-98/T.Inf/TA/2012
ampiran : -
terhal : Bimbingan Skripsi

29 Oktober 2012

epada : Yth. Sdr. Ahmad Faisol, ST
Dosen Pembimbing Program Studi Teknik Informatika S1
Institut Teknologi Nasional
M a l a n g

Dengan hormat

Sesuai dengan permohonan dan persetujuan dalam Proposal Skripsi untuk mahasiswa :

Nama : Delis Kurniawan
Nim : 0818078
Prodi : Teknik Informatika S1
Fakultas : Teknologi Industri

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

29 Oktober 2012 s/d 29 April 2013

Sebagai satu syarat untuk menempuh Ujian Sarjana Teknik, Program Studi Teknik Informatika S1.

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

Mengetahui
Program Studi Teknik Informatika S1
Ketua,
Joseph Dedy Irawan, ST, MT
NIP : 197404162005021002

Form S-4a

Nama : Olfie Prasyar

Pekerjaan : Mahasiswa

Berilah tanda pada salah satu kolom "Nilai" menurut anda paling benar untuk penilaian Aplikasi Sistem Informasi Akademik SMPN 3 Seteluk.

No	Pertanyaan	Nilai
1	Apakah Tampilan dari Aplikasi Sistem Informasi Akademik ini Menarik	S
2	Apakah Aplikasi Sistem Informasi Akademik ini mudah di gunakan	S
3	Apakah Aplikasi Sistem Informasi Akademik ini dapat membantu kegiatan pengolahan data	S
4	Apakah Sistem Informasi Akademik ini dapat membantu dalam penyajian informasi secara akurat, mudah dan cepat	SS

Ket:

SS = Sangat Setuju S = Setuju KS = Kurang Setuju TS= Tidak Setuju

Saran : Penerapannya nanti lebih memudahkan penggunaan oleh siswa

Terima Kasih

Nama : Sutran Rahmatullah

Pekerjaan : Mahasiswa

Berilah tanda pada salah satu kolom "Nilai" menurut anda paling benar untuk penilaian Aplikasi Sistem Informasi Akademik SMPN 3 Seteluk.

No	Pertanyaan	Nilai
1	Apakah Tampilan dari Aplikasi Sistem Informasi Akademik ini Menarik	d
2	Apakah Aplikasi Sistem Informasi Akademik ini mudah di gunakan	f
3	Apakah Aplikasi Sistem Informasi Akademik ini dapat membantu kegiatan pengolahan data	f
4	Apakah Sistem Informasi Akademik ini dapat membantu dalam penyajian informasi secara akurat, mudah dan cepat	ff

Ket:

SS = Sangat Setuju S = Setuju KS = Kurang Setuju TS= Tidak Setuju

Saran : Pehiburan & kemaya-

Terima Kasih

Nama : Aditya Dwi Marganto

Pekerjaan : Mahasiswa

Berilah tanda pada salah satu kolom "Nilai" menurut anda paling benar untuk penilaian Aplikasi Sistem Informasi Akademik SMPN 3 Seteluk.

No	Pertanyaan	Nilai
1	Apakah Tampilan dari Aplikasi Sistem Informasi Akademik ini Menarik	S
2	Apakah Aplikasi Sistem Informasi Akademik ini mudah di gunakan	S
3	Apakah Aplikasi Sistem Informasi Akademik ini dapat membantu kegiatan pengolahan data	S
4	Apakah Sistem Informasi Akademik ini dapat membantu dalam penyajian informasi secara akurat, mudah dan cepat	C

Ket:

SS = Sangat Setuju S = Setuju KS = Kurang Setuju TS= Tidak Setuju

Saran :

Terima Kasih

Nama : Irwan Rakhmawan

Pekerjaan : Mahasiswa

Berilah tanda pada salah satu kolom "Nilai" menurut anda paling benar untuk penilaian Aplikasi Sistem Informasi Akademik SMPN 3 Seteluk.

No	Pertanyaan	Nilai
1	Apakah Tampilan dari Aplikasi Sistem Informasi Akademik ini Menarik	S
2	Apakah Aplikasi Sistem Informasi Akademik ini mudah di gunakan	SS
3	Apakah Aplikasi Sistem Informasi Akademik ini dapat membantu kegiatan pengolahan data	S
4	Apakah Sistem Informasi Akademik ini dapat membantu dalam penyajian informasi secara akurat, mudah dan cepat	S

Ket:

SS = Sangat Setuju S = Setuju KS = Kurang Setuju TS= Tidak Setuju

Saran :

Terima Kasih

Nama : Zakariah

Pekerjaan : Mahasiswa

Berilah tanda pada salah satu kolom "Nilai" menurut anda paling benar untuk penilaian Aplikasi Sistem Informasi Akademik SMPN 3 Seteluk.

No	Pertanyaan	Nilai
1	Apakah Tampilan dari Aplikasi Sistem Informasi Akademik ini Menarik	SS
2	Apakah Aplikasi Sistem Informasi Akademik ini mudah di gunakan	S
3	Apakah Aplikasi Sistem Informasi Akademik ini dapat membantu kegiatan pengolahan data	S
4	Apakah Sistem Informasi Akademik ini dapat membantu dalam penyajian informasi secara akurat, mudah dan cepat	S

Ket:

SS = Sangat Setuju S = Setuju KS = Kurang Setuju TS= Tidak Setuju

Saran :

Terima Kasih

1. Script Check Login

```
<?php session_start();
include "db.php";
$sql = "select no_induk,password,hak_akses from user where no_induk=\"".$_POST[induk]."" and
password=\"".$_POST[pass]."";;
$kueri = mysql_query($sql);
if (mysql_num_rows($kueri) == 0)
{
    $hsl = "Password anda salah !";
    header("location:../index.php?rs1=".$hsl);
} else
{
    $kueri = mysql_query($sql);
    $data=mysql_fetch_array($kueri);
    if($data['hak_akses'] != 'siswa'){
        $_SESSION['NI']=$data[0];
        $_SESSION['HK']=$data[2];
        header("location:index.php");
    } else
    {
        $_SESSION['NI']=$data[0];
        $_SESSION['HK']=$data[2];
        header("location:../indentitas.php");
    }
}
?>
```

2. Script Menu Admin

```
<?php session_start();
$mn = $_GET['x'];
if ($mn == 'us001') {
    if ($_SESSION['HK']=='admin') {
        echo '<SCRIPT LANGUAGE="JavaScript"
TYPE="text/javascript">
<!--
        document.location = "index.php?nom=us1233 " ;
//-->
</SCRIPT>';
    }
    else
    {
        echo '<SCRIPT LANGUAGE="JavaScript"
TYPE="text/javascript">
<!--
        document.location = "index.php" ;
//-->
</SCRIPT>';
    }
}
if ($mn == 'guru') {
```

```
echo '<SCRIPT LANGUAGE="JavaScript"
TYPE="text/javascript">
<!--
        document.location = "index.php?nom=gr0012 " ;
//-->
</SCRIPT>';
}
if ($mn == 'sw001') {
    echo '<SCRIPT LANGUAGE="JavaScript"
TYPE="text/javascript">
<!--
        document.location = "index.php?nom=s5346 " ;
//-->
</SCRIPT>';
}
if ($mn == 'wk001') {
    echo '<SCRIPT LANGUAGE="JavaScript"
TYPE="text/javascript">
<!--
        document.location = "index.php?nom=w35wwe " ;
//-->
</SCRIPT>';
}
if ($mn == 'kl001') {
    echo '<SCRIPT LANGUAGE="JavaScript"
TYPE="text/javascript">
<!--
        document.location = "index.php?nom=kl989 " ;
//-->
</SCRIPT>';
}
if ($mn == 'mp001') {
    echo '<SCRIPT LANGUAGE="JavaScript"
TYPE="text/javascript">
<!--
        document.location = "index.php?nom=m0898 " ;
//-->
</SCRIPT>';
}
if ($mn == 'mk001') {
    echo '<SCRIPT LANGUAGE="JavaScript"
TYPE="text/javascript">
<!--
        document.location = "index.php?nom=m4663 " ;
//-->
</SCRIPT>';
}
if ($mn == 'th001') {
    echo '<SCRIPT LANGUAGE="JavaScript"
TYPE="text/javascript">
<!--
```

```

        document.location = "index.php?nom=9897t" ;
    //-->
    </SCRIPT>';
}
?
```

3. Script Guru

```

<html>
<head>
<link rel="stylesheet" type="text/css" href="../admin/css/styles.css">
<link rel="stylesheet" type="text/css" href="../admin/css/demos.css">
<link rel="stylesheet" type="text/css" href="../admin/css/start/jquery-ui-1.8.20.custom.css">
<script type="text/javascript" src="../admin/js/jquery-1.7.2.min.js"></script>
<script type="text/javascript" src="../admin/js/jquery-ui-1.8.20.custom.min.js"></script>
<script type="text/javascript">
// Popup window code
function newPopup(url) {
    popupWindow = window.open(
        url,'popUpWindow','height=250,width=500,top=auto,resizable=yes,scrollbars=yes,toolbar=y
es,menubar=no,location=no,directories=no,status=yes')
}
</script>
<title>GURU</title>
</head>
<body>
<div id="main">
<!--Dibawah ini adalah dialog setiap aksi-->
<div id="dialog-add" title="Tambah Data Customer">
    <form>
        <table border="0">
            <tr>
                <td width="122"><label for="no">NIP.</label></td>
                <td width="355"><input type="text" name="noadd" id="noadd" class="text ui-
widget-content ui-corner-all" />
                    * </td>
            </tr>
            <tr>
                <td><label for="nama">NUPTK.</label></td>
                <td><input name="nuptk" type="text" class="text ui-widget-content ui-corner-all"
id="nuptk" value="" /></td>
                    <tr>
                        <td><label for="nama">NAMA</label></td>
                        <td><input name="nama" class="text ui-widget-content ui-corner-all" id="nama"
size="40">
                            * </td>
                        <tr>
                            <td><label for="nama">Jenis Kelamin</label></td>
                            <td><select name="jk" id="jk" value="" class="text ui-widget-content ui-corner-all"
/>
                                <option value="Laki-laki"> Laki - laki </option>

```

```

<option value="Perempuan"> Perempuan </option>
</select> * Pilih salah satu </td>
<tr>
<td><label for="nama">Tempat Lahir</label></td>
<td><input type="text" name="tlahir" id="tlahir" value="" class="text ui-widget-content ui-corner-all"/></td>
<tr>
<td><label for="nama">Tanggal Lahir</label></td>
<td><input type="text" name="tgllahir" id="tgllahir" value="" class="text ui-widget-content ui-corner-all" />
* Format date (hari/bln/tahun)</td>
<tr>
<td><label for="nama">Agama</label></td>
<td><input name="agama" type="text" class="text ui-widget-content ui-corner-all" id="agama" value="" /></td>
<tr>
<td><label for="nama">Alamat</label></td>
<td><textarea name="alamat" cols="40" rows="2" class="text ui-widget-content ui-corner-all" id="alamat"></textarea>
*</td>
<tr>
<td><label for="nama">Telp</label></td>
<td><input type="text" name="telp" id="telp" value="" class="text ui-widget-content ui-corner-all" /></td>
<tr>
<td>catatan </td>
<td>* Harus diisi</td>
<tr>
</table>
</form>
</div>
<div id="dialog-edit" title="Edit Data Customer">
<form>
<table border="0">
<tr>
<td width="115"><label for="no">NIP.</label></td>
<td width="352"><div id="noedit"></div></td>
</tr>
<tr>
<td><label for="label">NUPTK.</label></td>
<td><input name="nuptk2" type="text" class="text ui-widget-content ui-corner-all" id="nuptk2" value="" /></td>
<tr>
<td><label for="label">NAMA</label></td>
<td><input name="nama2" class="text ui-widget-content ui-corner-all" id="nama2" size="40">
*</td>
<tr>
<td><label for="label">Jenis Kelamin</label></td>
<td><select name="jk2" id="jk2" value="" class="text ui-widget-content ui-corner-all" />
<option value="Laki-laki"> Laki - laki </option>

```

```

<option value="Perempuan"> Perempuan </option>
</select></td>
<tr>
<td><label for="label">Tempat Lahir</label></td>
<td><input type="text" name="tlahir2" id="tlahir2" value="" class="text ui-widget-content ui-corner-all"/></td>
<tr>
<td><label for="label">Tanggal Lahir</label></td>
<td><input type="text" name="tgllahir2" id="tgllahir2" value="" class="text ui-widget-content ui-corner-all" />
* Format date (hari/bulan/tahun)</td>
<tr>
<td><label for="label">Agama</label></td>
<td><input name="agama2" type="text" class="text ui-widget-content ui-corner-all" id="agama2" value="" /></td>
<tr>
<td><label for="label">Alamat</label></td>
<td><textarea name="alamat2" cols="40" rows="2" class="text ui-widget-content ui-corner-all" id="alamat2"></textarea>
*</td>
<tr>
<td><label for="label">Telp</label></td>
<td><input type="text" name="telp2" id="telp2" value="" class="text ui-widget-content ui-corner-all" /></td>
<tr>
<td>catatan</td>
<td>* Harus diisi</td>
<tr>
</table>
</form>
</div>
<div id="dialog-delete" title="Yakin hapus data?">
<p><span class="ui-icon ui-icon-alert" style="float:left; margin:0 7px 20px 0;"></span><div id="isi">isi</div></p>
</div>
<!--Akhir dari dialog setiap aksi--&gt;
<!--Table penampilan data--&gt;
&lt;div id="tablenya"&gt;
&lt;table cellspacing="0" id="mytable"&gt;
&lt;caption&gt;
Data Guru - SMPN 03 SETELUK SUMBAWA BARAT
&lt;/caption&gt;
&lt;tr&gt;
&lt;th class="no" scope="col"&gt;NIP&lt;/th&gt;
&lt;th scope="col"&gt;NUPTK&lt;/th&gt;
&lt;th scope="col"&gt;Nama&lt;/th&gt;
&lt;th scope="col"&gt;JK&lt;/th&gt;
&lt;th scope="col"&gt;T_Lahir&lt;/th&gt;
&lt;th scope="col"&gt;TG.Lahir&lt;/th&gt;
&lt;th scope="col"&gt;Agama&lt;/th&gt;
</pre>

```

```

<th scope="col">Alamat</th>
<th scope="col">Telp</th>
<th scope="col">Aksi</th>
</tr>
<?php
    include "db.php";
    $sql = 'SELECT * FROM guru ORDER BY NIP ASC';
    $kueri = mysql_query($sql);
    while ($baris = mysql_fetch_array($kueri)){
        echo "<tr>
            <td id='rs' class='nipC'>".$baris['NIP']."</td>
            <td id='rs' class='nuptkC'>".$baris['NUPTK']."</td>
            <td id='rs' class='namaC'>".$baris['NAMA']."</td>
            <td id='rs' class='jkC'>".$baris['JENIS_KELAMIN']."</td>
            <td id='rs' class='tahirC'>".$baris['TMP_LAHIR']."</td>
            <td id='rs' class='tgllahirC'>".$baris['TGL_LAHIR']."</td>
            <td id='rs' class='agamaC'>".$baris['AGAMA']."</td>
            <td id='rs' class='alamatC'>".$baris['ALAMAT']."</td>
            <td id='rs' class='telpC'>".$baris['TELEPON']."</td>;
        if ($_SESSION['HK']=='admin') {
            echo"
                <td id='rs' align='right'><img src='PNG/action_add.png' onClick=AddNew()> -
                <img src='PNG/reply.png' onClick='EditData($(this))'> -
                <img src='PNG/action_delete.png' onClick='DeleteData($(this))'> -
                <a href=JavaScript:newPopup('form.php?id='".$baris['NIP']."'")>foto</a>
            </td>";
        } else {
            echo "<td id='rs' align='right'> no action </td>";
        }
        echo "</tr>";
    }
?>
</table>
<?php
    if ($_SESSION['HK']=='admin') {
        echo '<a href="#" onClick="AddNew()">Insert new data</a>';
    }
?>
<!--Akhir dari table-->
</div>
<div id="waiting">
<p> Saving data...</p>
</div>
<script>
var
    objRow, //variable row yg diisi pada saat dialog edit
    nipdelete; //variable nip yang diisi pada saat event delete
<!--Dibawah ini adalah inisialisasi dialog jQuery-->
$("#waiting").hide();
//inisialisasi dialog tambah data

```

```

$( "#dialog-add" ).dialog({
    autoOpen: false,
    resizable: false,
    height: 450,
    width: 520,
    modal: true,
    buttons: {
        "Proses": function() {
            var
                //isi variable dengan input data yang telah diisi oleh user
                ano = $("input#noadd").val();
                anuptk = $("input#nuptk").val();
                anama = $("input#nama").val();
                ajk = $("select#jk").val();
                atlahir = $("input#tahir").val();
                atgllahir = $("input#tgllahir").val();
                aagama = $("input#agama").val();
                aalamat = $("textarea#alamat").val();
                atelp = $("input#telp").val();
                //aksi
                aact= 'insert';
            //event untuk melakukan POST/GET pada postdata.php melalui jQuery
            $.ajax({
                type: "POST", //definisikan aksinya (POST/GET)
                url: "gurup.php", //definisikan urlnya
                data:
                {"no":ano,"nuptk":anuptk,"nama":anama,"jk":ajk,"tahir":atlahir,"tgllahir":atgllahir,"agama":aagama
                , "alamat":aalamat,"telp":atelp,"act":aact}, //data yang akan dikirim
                timeout: 10000, //timeout dari request
                beforeSend: function(){}, //event yang akan dieksekusi sebelum pengiriman data
                complete: function(){}, //event yang akan dieksekusi setelah pengiriman data
                cache: false, //cache
                success: function(result){ //event yang akan dieksekusi setelah data berhasil dikirim
                    if (result=='success'){ //apabila respon dari postdata.php
                        success maka
                            var
                            //link button aksi
                            sbutton = "
                                <td id='rs' align='right'><img
src='PNG/action_add.png' onClick=AddNew()> - <img src='PNG/reply.png'
onClick='EditData($(this))'> - <img src='PNG/action_delete.png' onClick='DeleteData($(this))'> -
<?php echo" <a href=JavaScript:newPopup('form.php?id=".$baris['NIP']."'>foto</a>"; ?></td>";
                            //isi data row baru
                            s = '<tr><td id="rs" class="nipC">' +ano+ '</td><td
id="rs" class="nuptkC">' +anuptk+ '</td><td id="rs" class="namaC">' +anama+ '</td><td id="rs"
class="jkC">' +ajk+ '</td><td id="rs" class="tahirC">' +atlahir+ '</td><td id="rs"
class="tgllahirC">' +atgllahir+ '</td><td id="rs" class="agamaC">' +aagama+ '</td><td id="rs"
class="alamatC">' +aalamat+ '</td><td id="rs" class="telpC">' +atelp+ '</td>' +sbutton+ '</tr>';
                            //tambahkan row baru pada baris terakhir
                }
            }
        }
    }
});
```

```

        $('#mytable > tbody:last').append(s);
    }else{alert(result);}
    //close waiting
    $('#waiting').hide();
},
error: function(error){alert('request timeout, please try again.');//$( this ).dialog( "close" );} //event
yang akan diseleksusi pada saat error berlangsung
}
);
$('#waiting').show();
$( this ).dialog( "close" );
},
Cancel: function() {
    $( this ).dialog( "close" );
}
}
});
//inisialisasi dialog edit data
//penjelasan dari code ini hampir sama dengan dialog tambah data
$( "#dialog-edit" ).dialog({
    autoOpen: false,
    resizable: false,
    height: 450,
    width: 520,
    modal: true,
    buttons: {
        "Update": function() {
            var
                ano = $("#noedit").html();
                anuptk = $("input#nuptk2").val();
                anama = $("input#nama2").val();
                ajk = $("select#jk2").val();
                atlahir = $("input#tlahir2").val();
                atgllahir = $("input#tgllahir2").val();
                aagama = $("input#agama2").val();
                aalamat = $("textarea#alamat2").val();
                atelp = $("input#telp2").val();
                aact= 'edit';
            $.ajax({
                type: "POST",
                url: "gurup.php",
                data:
                {"no":ano,"nuptk":anuptk,"nama":anama,"jk":ajk,"tlahir":atlahir,"tgllahir":atgllahir,"agama":aagama
                ,"alamat":aalamat,"telp":atelp,"act":aact}, //data yang akan dikirim
                timeout: 10000,
                beforeSend: function(){},
                complete: function(){},
                cache: false,
                success: function(result){
                    if (result=='success'){
                        //ubah isi data table sesuai dengan perubahan yang terjadi

```

```

objRow.find(".nuptkC").html($("#input#nuptk2").val())
objRow.find(".namaC").html($("#input#nama2").val());
                objRow.find(".jkC").html($("#select#jk2").val());
objRow.find(".tlahirC").html($("#input#tlahir2").val());
objRow.find(".tgllahirC").html($("#input#tgllahir2").val());
objRow.find(".agamaC").html($("#input#agama2").val());
objRow.find(".alamatC").html($("#textarea#alamat2").val());
                objRow.find(".telpC").html($("#input#telp2").val());
                $("#waiting").hide();
            }else{alert(result);}
        },
error: function(error){$("#waiting").hide();alert('request timeout, please try again.');
}
);
$("#waiting").show();
$( this ).dialog( "close" );
},
Cancel: function() {
    $( this ).dialog( "close" );
}
}
});

//inisialisasi dialog hapus data
//penjelasan dari code ini hampir sama dengan dialog tambah data
$( "#dialog-delete" ).dialog({
autoOpen:false,
resizable: false,
height:150,
modal: true,
buttons: {
    "Proses": function() {
        var
            aact= 'delete';
        $.ajax({
            type: "POST",
            url: "gurup.php",
            data: {"no":nipdelete,"act":aact},
            timeout: 10000,
            beforeSend: function(){}
            complete: function(){}
            cache: false,
            success: function(result){
                if (result=='success'){
                    //hapus baris table yang terpilih
                    objRow.remove();
                }else{
                    alert(result);
                }
                $("#waiting").hide();
            },
        });
    }
}
});
```

```

error: function(error){alert('request timeout, please try again.');//$("#waiting").hide();
}
);
$( this ).dialog( "close" );
$("#waiting").show();
},
"Batal": function() {
    $( this ).dialog( "close" );
}
}
});

<!--Akhir dari inisialisasi dialog jQuery-->
<!--Dibawah ini adalah event handler dari setiap aksi (add,edit,delete)-->
function AddNew(){
//untuk data baru, kosongkan fieldnya terlebih dahulu
$("input#noadd").val("");
$("input#nuptk").val("");
$("input#nama").val("");
$("input#tlahir").val("");
$("input#tgllahir").val('00/00/0000');
$("input#agama").val("");
$("textarea#alamat").val("");
$("input#telp").val("");
//buka dialog
$( "#dialog-add" ).dialog("open");
}

function EditData(obj){
var nc = obj.parent().parent().find(".nipC").html();
nuptk = obj.parent().parent().find(".nuptkC").html();
nama = obj.parent().parent().find(".namaC").html();
jk = obj.parent().parent().find(".jkC").html();
tlahir = obj.parent().parent().find(".tlahirC").html();
tgllahir = obj.parent().parent().find(".tgllahirC").html();
agama = obj.parent().parent().find(".agamaC").html();
alamat = obj.parent().parent().find(".alamatC").html();
telp = obj.parent().parent().find(".telpC").html();
$("#noedit").empty().append(nc);
$("input#nuptk2").val(nuptk);
$("input#nama2").val(nama);
$("input#tlahir2").val(tlahir);
$("input#tgllahir2").val(tgllahir);
$("input#agama2").val(agama);
$("textarea#alamat2").val(alamat);
$("input#telp2").val(telp);
$("select#jk2").val(jk);
objRow = obj.parent().parent(); //simpan object row
$( "#dialog-edit" ).dialog("open");
}

function DeleteData(obj){
nipdelete = obj.parent().parent().find(".nipC").html();

```

```

    $("#" + isi).empty().append('Apakah anda yakin akan menghapus data dengan NIP : ' +
nipdelete + ' ?');
    $("#" + dialogDelete).dialog( "open" );
    objRow = obj.parent().parent(); //simpan object row
}
<!--Akhir dari Event Handler-->
</script>
</div>
</body>
</html>

```

4. Script Siswa

```

<html>
<head>
<link rel="stylesheet" type="text/css" href="../admin/css/styles.css">
<link rel="stylesheet" type="text/css" href="../admin/css/demos.css">
<link rel="stylesheet" type="text/css" href="../admin/css/start/jquery-ui-1.8.20.custom.css">
<script type="text/javascript" src="../admin/js/jquery-1.7.2.min.js"></script>
<script type="text/javascript" src="../admin/js/jquery-ui-1.8.20.custom.min.js"></script>
<script type="text/javascript">
// Popup window code
function newPopup(url) {
    popupWindow = window.open(
        url,'popUpWindow','height=250,width=500,top=auto,resizable=yes,scrollbars=yes,toolbar=y
es,menubar=no,location=no,directories=no,status=yes')
}
</script>
<title>GURU</title>
</head>
<body>
<div id="main">
<!--Dibawah ini adalah dialog setiap aksi-->
<div id="dialog-add" title="Tambah Data Siswa">
    <form>
        <table border="0">
            <tr>
                <td width="122"><label for="no">NIM.</label></td>
                <td width="355"><input type="text" name="noadd" id="noadd" class="text ui-
widget-content ui-corner-all" />
                *</td>
            </tr>
            <tr>
                <td><label for="nama">NISN.</label></td>
            <td><input name="nuptk" type="text" class="text ui-widget-content ui-corner-all" id="nuptk"
value="" /></td>
                <tr>
                    <td><label for="nama">Nama</label></td>
                    <td><input name="nama" class="text ui-widget-content ui-corner-all" id="nama" size="40">
                    *</td>
                <tr>

```

```

<td><label for="nama">Jenis Kelamin</label></td>
<td><select name="jk" id="jk" value="" class="text ui-widget-content ui-corner-all">
/>
    <option value="Laki-laki"> Laki - laki </option>
    <option value="Perempuan"> Perempuan </option>
</select>    </td>
    <tr>
        <td><label for="nama">Tempat Lahir</label></td>
<td><input type="text" name="tlahir" id="tlahir" value="" class="text ui-widget-content ui-corner-all"/></td>
        <tr>
            <td><label for="nama">Tanggal Lahir</label></td>
<td><input type="text" name="tgllahir" id="tgllahir" value="" class="text ui-widget-content ui-corner-all" />
            * Format Date (hari/bulan/tahun)</td>
        <tr>
            <td><label for="nama">Agama</label></td>
<td><input name="agama" type="text" class="text ui-widget-content ui-corner-all" id="agama" value="" /></td>
            <tr>
                <td><label for="nama">Alamat</label></td>
<td><textarea name="alamat" cols="40" rows="2" class="text ui-widget-content ui-corner-all" id="alamat"></textarea>
                *</td>
            <tr>
                <td><label for="nama">Telp</label></td>
<td><input type="text" name="telp" id="telp" value="" class="text ui-widget-content ui-corner-all" /></td>
                <tr>
                    <td>Nama Orang Tua</td>
<td><input type="text" name="namaortu" id="namaortu" value="" class="text ui-widget-content ui-corner-all" />
                    *</td>
                <tr>
                    <td>Catatan</td>
                    <td>* Harus diisi</td>
                <tr>
                    </table>
                </form>
</div>
<div id="dialog-edit" title="Edit Data Siswa">
    <form>
        <table border="0">
            <tr>
                <td width="122"><label for="no">NIM.</label></td>
                <td width="355"><div id="noedit"></div></td>
            </tr>
            <tr>
                <td><label for="label">NISN.</label></td>

```

```

<td><input name="nuptk2" type="text" class="text ui-widget-content ui-corner-all" id="nuptk2" value="" /></td>
<tr>
<td><label for="label">Nama</label></td>
<td><input name="nama2" class="text ui-widget-content ui-corner-all" id="nama2" size="40">*</td>
<tr>
<td><label for="label">Jenis Kelamin</label></td>
<td><select name="jk2" id="jk2" value="" class="text ui-widget-content ui-corner-all" />
<option value="Laki-laki"> Laki - laki </option>
<option value="Perempuan"> Perempuan </option>
</select> </td>
<tr>
<td><label for="label">Tempat Lahir</label></td>
<td><input type="text" name="tlahir2" id="tlahir2" value="" class="text ui-widget-content ui-corner-all"/></td>
<tr>
<td><label for="label">Tanggal Lahir</label></td>
<td><input type="text" name="tgllahir2" id="tgllahir2" value="" class="text ui-widget-content ui-corner-all" />
* Format Date (hari/bulan/tahun)</td>
<tr>
<td><label for="label">Agama</label></td>
<td><input name="agama2" type="text" class="text ui-widget-content ui-corner-all" id="agama2" value="" /></td>
<tr>
<td><label for="label">Alamat</label></td>
<td><textarea name="alamat2" cols="40" rows="2" class="text ui-widget-content ui-corner-all" id="alamat2"></textarea>
*</td>
<tr>
<tr>
<td><label for="label">Telp</label></td>
<td><input type="text" name="telp2" id="telp2" value="" class="text ui-widget-content ui-corner-all" /></td>
<tr>
<td>Nama Orang Tua</td>
<td><input type="text" name="namaortu2" id="namaortu2" value="" class="text ui-widget-content ui-corner-all" />
*</td>
<tr>
<td>Catatan</td>
<td>* Harus diisi</td>
<tr>
</table>
</form>
</div>
<div id="dialog-delete" title="Yakin hapus data?">
<p><span class="ui-icon ui-icon-alert" style="float:left; margin:0 7px 20px 0;"></span><div id="isi">isi</div></p>
</div>

```

```

<!--Akhir dari dialog setiap aksi-->
<!--Table penampilan data-->
<div id="tablenya">
<table cellspacing="0" id="mytable">
<caption>
Data Siswa - SMPN 03 SETELUK SUMBAWA BARAT
</caption>
<tr>
<th scope="col">NIM</th>
<th scope="col">NISN</th>
<th scope="col">Nama</th>
<th scope="col">JK</th>
<th scope="col">T_Lahir</th>
<th scope="col">TGL_Lahir</th>
<th scope="col">Agama</th>
<th scope="col">Alamat</th>
<th scope="col">Telp</th>
<th scope="col">Nama OrTu</th>
<th scope="col">Aksi</th>
</tr>
<?php
    include "db.php";
    $sql = 'SELECT * FROM siswa ORDER BY NIM ASC';
    $kueri = mysql_query($sql);
    while ($baris = mysql_fetch_array($kueri)){
        echo "<tr>
            <td id='rs' class='nipC'>".$baris['NIM']."</td>
            <td id='rs' class='nuptkC'>".$baris['NISN']."</td>
            <td id='rs' class='namaC'>".$baris['NAMA_MURID']."</td>
            <td id='rs' class='jkC'>".$baris['JENIS_KELAMIN']."</td>
            <td id='rs' class='tlahirC'>".$baris['TMP_LAHIR']."</td>
            <td id='rs' class='tgllahirC'>".$baris['TGL_LAHIR']."</td>
            <td id='rs' class='agamaC'>".$baris['AGAMA']."</td>
            <td id='rs' class='alamatC'>".$baris['ALAMAT']."</td>
            <td id='rs' class='telpC'>".$baris['TELEPON']."</td>
            <td id='rs' class='namaortuC'>".$baris['NAMA_ORTU']."</td>";
        if ($_SESSION['HK']=='admin') {
            echo"
                <td id='rs' align='right'><img src='PNG/action_add.png' onClick=AddNew()> -
                <img src='PNG/reply.png' onClick='EditData($(this))'> -
                <img src='PNG/action_delete.png' onClick='DeleteData($(this))'> -
                <a href=JavaScript:newPopup('form_siswa.php?id='".$baris['NIM']."'>foto</a>
            </td>";
        } else
        {
            echo "<td id='rs' align='right'> no action </td>";
        }
        echo "</tr>";
    }
?>
</table>

```

```

<?php
    if ($_SESSION['HK']=='admin') {
        echo '<a href="#" onClick="AddNew()">Insert new data</a>';
    }
?>
<!--Akhir dari table-->
</div>
<div id="waiting">
<p> Saving data...</p>
</div>
<script>
var
objRow, //variable row yg diisi pada saat dialog edit
nipdelete; //variable nip yang diisi pada saat event delete
<!--Dibawah ini adalah inisialisasi dialog jQuery-->
$("#waiting").hide();
//inisialisasi dialog tambah data
$( "#dialog-add" ).dialog({
    autoOpen: false,
    resizable: false,
    height: 450,
    width: 520,
    modal: true,
    buttons: {
        "Proses": function() {
            var
                //isi variable dengan input data yang telah diisi oleh user
                ano = $("input#noadd").val();
                anuptk = $("input#nuptk").val();
                anama = $("input#nama").val();
                ajk = $("select#jk").val();
                atlahir = $("input#tlahir").val();
                atgllahir = $("input#tgllahir").val();
                aagama = $("input#agama").val();
                aalamat = $("textarea#alamat").val();
                atelp = $("input#telp").val();
                anamaortu = $("input#namaortu").val();
                //aksi
                aact= 'insert';
                //event untuk melakukan POST/GET pada postdata.php melalui jQuery
                $.ajax({
                    type: "POST", //definisikan aksinya (POST/GET)
                    url: "siswap.php", //definisikan urlnya
                    data:
                    {"no":ano,"nuptk":anuptk,"nama":anama,"jk":ajk,"tlahir":atlahir,"tgllahir":atgllahir,"agama":aagama
                    , "alamat":aalamat,"telp":atelp,"namaortu":anamaortu,"act":aact}, //data yang akan dikirim
                    timeout: 10000, //timeout dari request
                    beforeSend: function(){}, //event yang akan dieksekusi sebelum pengiriman data
                    complete: function(){}, //event yang akan dieksekusi setelah pengiriman data
                })
            }
        }
    }
}

```

```

cache: false, //cache
success: function(result){ //event yang akan dieksekusi setelah data
berhasil dikirim
    if (result=='success'){ //apabila respon dari postdata.php
success maka
        var
        //link button aksi
        sbutton = " <td id='rs' align='right'><img
src='PNG/action_add.png' onClick=AddNew()> - <img src='PNG/reply.png'
onClick='EditData($(this))> - <img src='PNG/action_delete.png' onClick='DeleteData($(this))>
</td>";
        //isi data row baru
        s = '<tr><td id="rs" class="nipC">' +ano+'</td><td id="rs"
class="jkC">' +ajk+'</td><td id="rs" class="tlahirC">' +atlahir+'</td><td id="rs"
class="tgllahirC">' +atgllahir+'</td><td id="rs" class="agamaC">' +aagama+'</td><td id="rs"
class="alamatC">' +aalamat+'</td><td id="rs" class="telpC">' +atelp+'</td><td id="rs"
class="telpC">' +anamaortu+'</td>' +sbutton+'</tr>';
        //tambahkan row baru pada baris terakhir
        $('#mytable > tbody:last').append(s);
    }else{alert(result);}
    //close waiting
    $("#waiting").hide();
},
error: function(error){alert('request timeout, please try again.');?>
(this ).dialog( "close" );} //event yang akan disejekusi pada saat error berlangsung
}
),
$("#waiting").show();
$( this ).dialog( "close" );
},
Cancel: function() {
    $( this ).dialog( "close" );
}
}
});
//inisialisasi dialog edit data
//penjelasan dari code ini hampir sama dengan dialog tambah data
$( "#dialog-edit" ).dialog({
autoOpen: false,
resizable: false,
height: 450,
width: 520,
modal: true,
buttons: {
"Update": function() {
var
ano = $("#noedit").html();
anuptk = $("input#nuptk2").val();
anama = $("input#nama2").val();
ajk = $("select#jk2").val();
}
}
}
);

```

```

atlahir = $("input#tlahir2").val();
atgllahir = $("input#tgllahir2").val();
aagama = $("input#agama2").val();
alamat = $("textarea#alamat2").val();
atelp = $("input#telp2").val();
anamaortu = $("input#namaortu2").val();
aact= 'edit';

$.ajax({
    type: "POST",
    url: "siswap.php",
    data:
    {"no":ano,"nuptk":anuptk,"nama":anama,"jk":ajk,"tlahir":atlahir,"tgllahir":atgllahir,"agama":aagama
    ,"alamat":alamat,"telp":atelp,"namaortu":anamaortu,"act":aact}, //data yang akan dikirim
    timeout: 10000,
    beforeSend: function(){},
    complete: function(){},
    cache: false,
    success: function(result){
        if (result=='success'){
            //ubah isi data table sesuai dengan perubahan yang terjadi
            objRow.find(".nuptkC").html($("#nuptk2").val());
            objRow.find(".namaC").html($("#nama2").val());
            objRow.find(".jkC").html($("#select#jk2").val());
            objRow.find(".tlahirC").html($("#tlahir2").val());
            objRow.find(".tgllahirC").html($("#tgllahir2").val());
            objRow.find(".agamaC").html($("#agama2").val());
            objRow.find(".alamatC").html($("#alamat2").val());
            objRow.find(".telpC").html($("#telp2").val());
            objRow.find(".namaortuC").html($("#namaortu2").val());
            $("#waiting").hide();
        }else{alert(result);}
    },
    error: function(error){$("#waiting").hide();alert('request timeout, please try again.');}
}
);
$("#waiting").show();
$( this ).dialog( "close" );
},
Cancel: function() {
    $( this ).dialog( "close" );
}
}
});

//inisialisasi dialog hapus data
//penjelasan dari code ini hampir sama dengan dialog tambah data
$( "#dialog-delete" ).dialog({
    autoOpen:false,
    resizable: false,
    height:150,
    modal: true,
    buttons: {

```

```

    "Proses": function() {
        var
            aact= 'delete';
        $.ajax({
            type: "POST",
            url: "siswap.php",
            data: {"no":nipdelete,"act":aact},
            timeout: 10000,
            beforeSend: function(){}
            complete: function(){}
            cache: false,
            success: function(result){
                if (result=='success'){
                    //hapus baris table yang terpilih
                    objRow.remove();
                }else{
                    alert(result);
                }
                $("#waiting").hide();
            },
            error: function(error){alert('request timeout, please try again.');//$("#waiting").hide();}
        });
        $( this ).dialog( "close" );
        $("#waiting").show();
    },
    "Batal": function() {
        $( this ).dialog( "close" );
    }
}
});

<!--Akhir dari inisialisasi dialog jQuery-->
<!--Dibawah ini adalah event handler dari setiap aksi (add,edit,delete)-->
function AddNew(){
    //untuk data baru, kosongkan fieldnya terlebih dahulu
    $("input#noadd").val("");
    $("input#nuptk").val("");
    $("input#nama").val("");
    $("input#tlahir").val("");
    $("input#tgllahir").val('00/00/0000');
    $("input#agama").val("");
    $("textarea#alamat").val("");
    $("input#telp").val("");
    $("input#namaortu").val("");
    //buka dialog
    $( "#dialog-add" ).dialog("open");
}

function EditData(obj){
    var nc = obj.parent().parent().find(".nipC").html();
    nuptk = obj.parent().parent().find(".nuptkC").html();
    nama = obj.parent().parent().find(".namaC").html();
}

```

```

jk = obj.parent().parent().find(".jkC").html();
tlahir = obj.parent().parent().find(".tlahirC").html();
tgllahir = obj.parent().parent().find(".tgllahirC").html();
agama = obj.parent().parent().find(".agamaC").html();
alamat = obj.parent().parent().find(".alamatC").html();
telp = obj.parent().parent().find(".telpC").html();
namaortu = obj.parent().parent().find(".namaortuC").html();
$("#noedit").empty().append(nc);
$("input#nuptk2").val(nuptk);
$("input#nama2").val(nama);
$("input#tlahir2").val(tlahir);
$("input#tgllahir2").val(tgllahir);
$("input#agama2").val(agama);
$("textarea#alamat2").val(alamat);
$("input#telp2").val(telp);
$("select#jk2").val(jk);
$("input#namaortu2").val(namaortu);
objRow = obj.parent().parent(); //simpan object row
$( "#dialog-edit" ).dialog("open");
}

function DeleteData(obj){
    nipdelete = obj.parent().parent().find(".nipC").html();
    $("#isi").empty().append('Apakah anda yakin akan menghapus data dengan NIP : ' +
    nipdelete + ' ?');
    $( "#dialog-delete" ).dialog( "open" );
    objRow = obj.parent().parent(); //simpan object row
}
<!--Akhir dari Event Handler-->
</script>
</div>
</body>
</html>
```