

**PENGEMBANGAN SISTEM PELAPORAN ABSENSI SISWA BERBASIS *SMS*
GATEWAY UNTUK PENGAWASAN ORANG TUA DI SMK N 4 KLATEN**

TUGAS AKHIR SKRIPSI

Diajukan kepada Fakultas Teknik Universitas Negeri Yogyakarta
untuk Memenuhi Sebagian Persyaratan Guna Memperoleh Gelar Sarjana
Pendidikan



Oleh :
Budi Cahyono
NIM 11520241045

**PROGRAM STUDI PENDIDIKAN TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS NEGERI YOGYAKARTA
2017**

SURAT PERNYATAAN

Saya yang bertanda tangan dibawah ini:

Nama : Budi Cahyono

NIM : 11520241045

Program Studi : Pendidikan Teknik Informatika

Judul TAS :

Pengembangan Sistem Pelaporan Absensi Siswa Berbasis SMS Gateway untuk
Pengawasan Orang Tua di SMK N 4 Klaten

menyatakan bahwa skripsi ini benar-benar karya saya sendiri. Sepanjang pengetahuan saya tidak terdapat karya atau pendapat yang ditulis atau diterbitkan orang lain kecuali sebagai acuan kutipan dengan mengikuti tata penulisan karya ilmiah yang telah lazim dan telah disebutkan dalam daftar pustaka.

Yogyakarta, Maret 2017

Yang menyatakan,

Budi Cahyono
NIM. 11520241045

LEMBAR PERSETUJUAN

Tugas Akhir Skripsi dengan Judul

PENGEMBANGAN SISTEM PELAPORAN ABSENSI SISWA BERBASIS SMS GATEWAY UNTUK PENGAWASAN ORANG TUA DI SMK N 4 KLATEN

Disusun oleh :

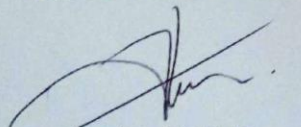
Budi Cahyono
NIM. 11520241045

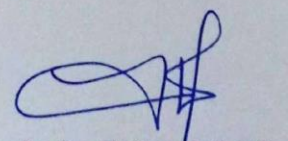
Telah memenuhi syarat dan disetujui oleh Dosen Pembimbing untuk dilaksanakan ujian akhir tugas skripsi bagi yang bersangkutan.

Yogyakarta, Februari 2017

Mengetahui,
Ketua Program Studi,
Pendidikan Teknik Informatika

Disetujui,
Pembimbing


Handaru Jati, Ph.D
NIP. 19740511 199903 1 002


Dr. Drs. EkoMarpanaji, M.T.
NIP. 19670608 199303 1 001

HALAMAN PENGESAHAN

Tugas Akhir Skripsi

PENGEMBANGAN SISTEM PELAPORAN ABSENSI SISWA BERBASIS SMS GATEWAY UNTUK PENGAWASAN ORANG TUA DI SMK N 4 KLATEN

Disusun oleh:

Budi Cahyono

NIM. 11520241045

Telah dipertahankan di depan Tim Penguji Tugas Akhir Skripsi Program Studi
Pendidikan Teknik Informatika Fakultas Teknik Universitas Negeri Yogyakarta
pada tanggal 22 Februari 2017

TIM PENGUJI

Nama/Jabatan	Tanda Tangan	Tanggal
Dr. Eko Marpanaji Ketua Penguji/Pembimbing		2/03/2017
Nurhasanah, M.Pd Sekretaris		28/02/2017
Adi Dewanto, M.Kom Penguji		1/03/2017

Yogyakarta, 2017

Fakultas Teknik Universitas Negeri Yogyakarta

Dekan,



Dr. Widarto, M.Pd

NIP. 19631230 1988121 1 001

HALAMAN MOTTO

"Man Jadda Wajada"

Barang siapa yang bersungguh-sungguh, dia (akan) mendapatkannya

"Jangan pernah mengejar kesuksesan, tetapi perbanyaklah ilmu dan keterampilan maka kesuksesan akan mengejarmu dengan sendirinya"

"Manusia yang baik adalah yang selalu memperbaiki diri"

"Jangan mengulangi kesalahan yang sama, karena masih banyak kesalahan lain yang bisa kita coba dan perbaiki"

PENGEMBANGAN SISTEM PELAPORAN ABSENSI SISWA BERBASIS SMS GATEWAY UNTUK PENGAWASAN ORANG TUA DI SMK N 4 KLATEN

Oleh:
Budi Cahyono
NIM. 11520241045

ABSTRAK

Tujuan dari penelitian ini adalah: (1) merancang dan membangun aplikasi pelaporan absensi berbasis SMS *Gateway* untuk pengawasan orang tua siswa di SMK N 4 Klaten, (2) memberikan pengetahuan mengenai tingkat kualitas aplikasi yang digunakan kepada guru SMK N 4 klaten sesuai standar ISO 25010 dari aspek *functional suitability*, *performance efficiency*, *usability*, *maintainability*, dan *compatibility*. Metode penelitian yang digunakan adalah metode penelitian pengembangan RnD (*Research and Development*). Model pengembangan yang digunakan pada penelitian ini adalah model-V yang terdiri dari tahap pengembangan dan tahap pengujian. Tahap pengembangan: (1) *requirements*, (2) *specification*, (3) *design*, (4) *implementation*. Tahap pengujian (5) *unit testing*, (6) *integration testing*, (7) *system testing*, (8) *acceptance testing*. Hasil penelitian ini adalah: (1) menghasilkan sistem pelaporan absensi siswa berbasis SMS *Gateway* dan (2) tingkat kelayakan sistem pelaporana absensi siswa ditinjau dari aspek *functional suitability* sebesar 100% dengan kategori sangat layak, *performance efficiency* 82 dengan *grade B*, *compatibility* sebesar 100% dengan kategori sangat layak, *maintainability* sebesar 100% dengan kategori sangat layak dan *usability* dengan koefisien *alpha cronbach* 0,885 dengan kategori "**Good**". Dari hasil-hasil tingkat kelayakan tersebut menunjukkan bahwa sistem pelaporan absensi siswa layak untuk digunakan.

Kata kunci: sistem, *research and development*, model-V, ISO 25010

HALAMAN PERSEMBAHAN

Alhamdulillah, segala puji bagi Allah SWT atas segala karunia-Nya sehingga penulis mampu menyelesaikan penulisan skripsi ini. Karya ini penulis persembahkan untuk:

1. Bapak Bagyo Sumaryoto dan keluarga (Mas Tono, Mas Yadi, Mas Gun, Mbak Peni, Mbak Anjar dan Lik Kadi) yang selalu memberikan dukungan dan doa demi terselesaikannya karya ini.
2. Bapak Dr. Eko Marpanaji yang selalu membimbing dan memberikan masukan demi terselesaikannya karya ini.
3. Aditio Agung Nugroho, Annis Nuraini, Soraya Mir'atun Miswah, Udin, Fariz Budi Widada, Rochmat Budiman, Tri Nugroho dan Musthofa yang selalu berbagi ilmu, memberikan bantuan dan semangat.
4. Teman-teman kelas F (Akhi, Mbak Nana, Mareta, Andri, Isnan, Farid, Ipeh, Faza, Hilma, Mamak, Fiani, Mifta, dll) yang luar biasa menginspirasi dan memotivasi.
5. Semua karyawan SMK N 4 Klaten, kelompok peternakan Sumber Rezeki, karang taruna PSMM, yang selalu mengajak untuk melangkah maju.
6. Kelompok "Warbyasah" perjuangan skripsi (Meganingtyas, Gatra, Puji, Ratna, Ina, Epik, Rizqon dan teman-teman yang tidak saya ketahui namanya) yang selalu mengingatkan dan mengajak untuk berprogress dalam pengerjaan skripsi.

KATA PENGANTAR

Puji syukur kehadiran Allah SWT atas berkat rahmat dan karunia-Nya, sehingga penulis dapat menyelesaikan Tugas Akhir Skripsi dalam rangka untuk memenuhi sebagian persyaratan untuk mendapatkan gelar Sarjana Pendidikan Program Studi Pendidikan Teknik Informatika Universitas Negeri Yogyakarta dengan judul "Pengembangan Sistem Pelaporan Absensi Siswa Berbasis SMS *Gateway* untuk Pengawasan Orang Tua di SMK N 4 Klaten" dapat disusun sesuai dengan harapan. Tugas Akhir Skripsi ini dapat diselesaikan tidak lepas dari bantuan, bimbingan, masukan, arahan dan kerjasama dengan pihak lain. Penulis menyampaikan ucapan terimakasih kepada yang terhormat:

1. Bapak Dr. Eko Marpanaji selaku Dosen Pembimbing Tugas Akhir Skripsi yang telah membantu, membimbing, memberikan saran, masukan selama penyusunan Tugas Akhir Skripsi ini.
2. Bapak Adi Dewanto, M.Kom selaku Penguji Utama dan Ibu Nurhasanah, M.Pd selaku Sekretaris yang memberikan koreksi perbaikan Tugas Akhir Skripsi ini.
3. Bapak Dr. Fatchul Arifin, selaku ketua Jurusan Pendidikan Teknik Elektronika dan Bapak Handaru Jati, Ph.D. selaku ketua Program Studi Pendidikan Teknik Informatika, beserta dosen dan staff yang telah memberikan bantuan, saran, masukan dan fasilitas selama proses penyusunan pra proposal sampai dengan selesainya Tugas Akhir Skripsi ini.

4. Bapak Totok Sukardiyono, M.T., Dosen Penasehat Akademik, yang selalu memberikan bimbingan dan arahan dalam menempuh studi ini.
5. Bapak Dr. Widarto, M.Pd. selaku Dekan Fakultas Teknik Universitas Negeri Yogyakarta yang memberikan persetujuan pelaksanaan Tugas Akhir Skripsi ini.
6. Bapak Drs. Sarbin M.M. ,guru-guru dan staff SMK Negeri 4 Klaten yang telah memberikan bantuan, memperlancar pengambilan data selama proses penelitian Tugas Akhir Skripsi.
7. Keluarga Bapak Bagyo Sumaryoto yang senantiasa memberikan dukungan dan doa.
8. Teman-teman Kelas F PTI UNY 2011 yang saya banggakan dan rindukan.
9. Semua pihak, secara langsung dan tidak langsung, yang tidak dapat disebutkan disini atas bantuan dan perhatiannya selama penyusunan Tugas Akhir Skripsi ini.

Akhir kata, semoga segala bantuan yang telah diberikan oleh semua pihak diatas menjadi amalan yang bermanfaat dan mendapatkan balasan dari Allah SWT. Demikian Tugas Akhir Skripsi ini penulis susun, besar harapan Tugas Akhir Skripsi ini bisa menjadi informasi bermanfaat bagi pembaca atau pihak lain yang membutuhkan

Yogyakarta, Maret 2017

Yang menyatakan,

Budi Cahyono
NIM. 11520241045

DAFTAR ISI

HALAMAN JUDUL.....	i
SURAT PERNYATAAN.....	ii
HALAMAN PENGESAHAN.....	iv
HALAMAN MOTTO.....	vi
ABSTRAK.....	vii
HALAMAN PERSEMBAHAN	viii
KATA PENGANTAR	ix
DAFTAR ISI.....	xi
DAFTAR TABEL.....	xiv
DAFTAR GAMBAR	xv
BAB I PENDAHULUAN	1
A. Latar Belakang Masalah.....	1
B. Identifikasi Masalah.....	3
C. Pembatasan Masalah.....	3
D. Rumusan Masalah.....	4
E. Tujuan Penelitian.....	4
F. Spesifikasi produk yang dikembangkan.....	4
G. Manfaat Penelitian.....	5
1. Teoritis	5
2. Praktis	6

BAB II KAJIAN PUSTAKA.....	7
A. Kajian Teori.....	7
1. Presensi Siswa.....	7
2. Teknologi SMS.....	8
3. <i>SMS Gateway</i>	8
4. <i>Gammu</i>	9
5. <i>Codeigniter</i>	9
6. MVC (Model <i>View Controller</i>)	10
7. Bahasa Pemrograman PHP	12
8. MySQL& Apache	13
9. Metode Pengembangan Sistem	14
10. Parameter Pengujian Perangkat Lunak.....	20
11. Desain Produk	30
12. <i>Class Diagram</i>	31
13. <i>Sequence Diagram</i>	32
14. <i>Activity Digram</i>	34
B. Penelitian yang Relevan.....	35
C. Kerangka Pikir.....	36
D. Pertanyaan Penelitian.....	39
BAB III METODE PENELITIAN.....	40
A. Model Pengembangan.....	40
B. Prosedur pengembangan.....	40
1. <i>Requirements</i>	40
2. <i>Specification</i>	41

3.	Desain Produk	42
4.	Implementasi	42
5.	Pengujian.....	42
6.	Waktu dan tempat penelitian	43
7.	Sumber data/subjek penelitian.....	44
8.	Metode dan alat pengumpul data	44
9.	Instrumen Penelitian	45
10.	Teknik Analisis Data	48
BAB IV HASIL PENELITIAN DAN PEMBAHASAN		51
1.	<i>Requirements</i>	51
2.	<i>Specification</i>	52
3.	Design	52
4.	Database	69
5.	Desain <i>Interface</i>	69
6.	Implementasi	71
7.	Pengujian.....	79
BAB V KESIMPULAN DAN SARAN.....		93
A.	Kesimpulan.....	93
B.	Saran.....	94
Daftar Pustaka.....		95
LAMPIRAN.....		97

DAFTAR TABEL

Tabel 1. ISO 25010 (ISO,2011)	20
Tabel 2. <i>The Computer System Usability Questionare</i> (CSUQ)	24
Tabel 3. Instrumen <i>maintainability</i> (Land, 2002)	26
Tabel 4. Simbol-simbol <i>use case</i> (S, Rosa, 2013: 156-158)	31
Tabel 5. Simbol-simbol <i>class diagram</i> (S, Rosa, 2013: 146-147)	32
Tabel 6. Simbol-simbol <i>sequence diagram</i> (S, Rosa, 2013: 165-167).....	33
Tabel 7. Simbol-simbol diagram aktivitas (S, Rosa, 2013: 162-163)	34
Tabel 8. Metode Pengujian.....	43
Tabel 9. Format standar <i>test case functional suitability</i> (Ghazy Assassa)	46
Tabel 10. Instrumen <i>usability</i>	46
Tabel 11. Instrumen <i>maintainability</i> (Land, 2002)	47
Tabel 12. <i>Interprestasi Alpha Cronbach</i>	49
Tabel 13 Desain <i>interface</i>	70
Tabel 14 Simbol <i>flowgraph</i>	80
Tabel 15 Keterangan setiap <i>node</i>	81
Tabel 16 <i>Independent path</i>	83
Tabel 17 <i>Test case</i>	84
Tabel 18 Hasil pengujian <i>efficiency</i>	87
Tabel 19 Instrumen <i>maintainability</i>	89
Table 20 Hasil pengujian <i>acceptance testing</i>	91
Tabel 21 Konsistensi <i>alpha cronbach</i>	92
Tabel 22 Hasil kuesioner angket <i>usability</i>	103

DAFTAR GAMBAR

Gambar 1. Konsep MVC	12
Gambar 2. Model V (Watkins, 2011:146)	15
Gambar 3 Tahapan pengujian	17
Gambar 4 <i>use case diagram</i>	53
Gambar 5 <i>Sequence diagram login</i>	54
Gambar 6 <i>Sequence diagram</i> presensi	55
Gambar 7 <i>Sequence diagram input data</i>	56
Gambar 8 <i>Sequence diagram edit data</i>	57
Gambar 9 <i>Sequence diagram delete data</i>	58
Gambar 10 <i>Sequence diagram cari data</i>	59
Gambar 11 <i>Sequence diagram reset password</i>	60
Gambar 12 <i>Sequence diagram absensi</i>	61
Gambar 13 <i>Activity diagram login</i>	62
Gambar 14 <i>Activity diagram input</i>	63
Gambar 15 <i>Activity diagram edit data</i>	64
Gambar 16 <i>Activity diagram delete data</i>	65
Gambar 17 <i>Activity diagram reset password</i>	66
Gambar 18 <i>Activity diagram cari</i>	67
Gambar 19 <i>Activity Diagram Absensi</i>	68
Gambar 20 ERD Absensi	69
Gambar 21 Halaman <i>login</i>	72

Gambar 22 Halaman <i>superadmin</i>	72
Gambar 23 Halaman <i>admin</i>	73
Gambar 24 Halaman data siswa.....	73
Gambar 25 Halaman <i>input</i> data siswa	74
Gambar 26 Lihat data <i>user</i>	74
Gambar 27 Halaman <i>input</i> data user	75
Gambar 28 Halaman Jurusan.....	75
Gambar 29 Halaman lihat absensi.....	76
Gambar 30 Halaman <i>input</i> absensi.....	76
Gambar 31 Tampilan SMS	77
Gambar 32 Tabel guru.....	78
Gambar 33 Tabel jurusan.....	78
Gambar 34 Tabel presensi.....	78
Gambar 35 Tabel siswa.....	79
Gambar 36 Tabel <i>user</i>	79
Gambar 37 <i>Flowgraph</i>	81
Gambar 38 Pengujian <i>efficiency</i>	86
Gambar 39 Statistik <i>efficiency</i>	87
Gambar 40 <i>Error</i> memasukkan <i>login</i>	89
Gambar 41 <i>Error</i> edit data.....	89

DAFTAR LAMPIRAN

Lampiran 1 <i>Use case</i>	98
Lampiran 2 Hasil pengujian <i>compatibility</i>	99
Lampiran 3 Matriks ISO 9126 dan ISO 25010.....	100
Lampiran 4 Hasil kuesioner angket <i>usability</i>	103
Lampiran 5 Surat izin penelitian UNY.....	106
Lampiran 6 Surat izin penelitian BAPEDDA.....	107
Lampiran 7 Instrumen <i>Usability</i>	108
Lampiran 8 Lanjutan instrumen <i>usability</i>	109
Lampiran 9 Daftar hadir	110
Lampiran 10 Surat keterangan selesai penelitian	111
Lampiran 11 <i>Source code</i>	112

BAB I

PENDAHULUAN

A. Latar Belakang Masalah

Salah satu kegiatan guru saat berada di kelas adalah mendata kehadiran siswa atau biasa disebut presensi. Menurut Kamus Besar Bahasa Indonesia (KBBI) absensi berarti ketidakhadiran yang berasal dari kata *absent* (tidak hadir). Absensi merupakan salah satu tolak ukur aktivitas siswa dalam proses pembelajaran. Absensi siswa terdiri dari siswa masuk, sakit, izin maupun alpha (tidak masuk tanpa keterangan). Sekolah memberikan laporan ketidakhadiran siswa pada akhir studi atau pada saat murid menerima rapor di akhir semester. Begitu juga di SMKN 4 Klaten, pelaporan absensi dimasukkan dalam rapor yang diberikan pada akhir semester. Orang tua mengalami kesulitan dalam mengingat siswa yang tidak berangkat selama satu semester baik dengan keterangan izin, sakit dan bahkan memungkinkan siswa membolos pada saat berlangsungnya kegiatan belajar mengajar jika laporan diberikan pada akhir semester (Akhmad Sudrajat, 2010).

Proses guru di SMKN 4 Klaten dalam memberikan laporan absensi kepada orang tua saat ini masih menggunakan cara manual, yaitu guru menghitung absensi siswa setiap harinya dan dimasukkan dalam rapor siswa berupa total keseluruhan absensi pada akhir semester. Pada pemberian rapor tersebut hanya terdapat kolom keterangan (sakit, izin dan tanpa keterangan) serta masih belum terdapat keterangan waktu dan tanggal pada masing-masing jenis absen siswa. Keterangan absensi siswa tersebut seharusnya dilaporkan secara detail kepada

orang tua sebagai pertanggung jawaban seorang guru terhadap murid. Berdasarkan observasi ketika PPL orang tua (cenderung) percaya dengan keadaan siswa disekolah tanpa memperdulikan keadaan sebenarnya apakah siswa benar-benar datang disekolah atau mengikuti kegiatan dari sekolah dan bahkan membolos.

Penyampaian informasi kehadiran siswa dari guru kepada orang tua sulit dilakukan jika melalui surat. Kesulitan disebabkan karena pengiriman surat membutuhkan waktu (surat diberikan kepada siswa saat masuk) dan juga sering tidak sampainya surat kepada orang tua wali. Selain itu, hal tersebut membuat orang tua tidak bisa memantau anaknya setiap hari secara penuh (*real time*).

Sejalan dengan bidang teknologi informasi dan komunikasi yang semakin maju, pemanfaatan teknologi dalam dunia pendidikan juga mulai berkembang. Teknologi dapat diterapkan untuk membantu mengatasi permasalahan dalam dunia pendidikan. Salah satunya adalah mampu membantu dan memberikan informasi absensi siswa kepada orang tua secara *real time*. Teknologi yang bisa dimanfaatkan untuk memberikan laporan secara *real time* kepada orang tua siswa adalah SMS. SMS (*Short Message Service*) adalah pesan singkat yang dimiliki oleh sebuah *provider* telekomunikasi dalam bentuk pengiriman berupa layanan *text* ke alat telekomunikasi lain yaitu *handphone* (Sofyan, 2015:184). Sifatnya yang personal serta penggunaanya yang mudah membuat SMS tetap populer sampai sekarang. Aplikasi *SMS Gateway* dapat digunakan sebagai media untuk mempermudah informasi absensi siswa selama kegiatan pembelajaran. Selain murah, layanan SMS ini juga bermanfaat untuk sekolah dalam memberikan informasi seputar absensi siswa kepada orang tua.

Berdasarkan observasi yang dilakukan, aplikasi ini dibutuhkan oleh pihak sekolah dan diperlukan sebuah sistem rekayasa aplikasi yang dapat memudahkan dalam pelaporan absensi. Serta mengukur sejauh mana kualitas sistem informasi absensi dengan memanfaatkan teknologi informasi yaitu *SMS Gateway*. Dengan judul: "Pengembangan Sistem Pelaporan Absensi Siswa Berbasis *SMS Gateway* untuk Pengawasan Orang Tua di SMK N 4 Klaten".

B. Identifikasi Masalah

Berdasarkan uraian latar belakang di atas, maka ada beberapa permasalahan yang dapat diidentifikasi yaitu sebagai berikut:

1. Terbatasnya kemampuan sekolah dalam memberikan informasi absensi kepada orang tua secara *real time*.
2. Kurangnya pengawasan dari orang tua murid terhadap siswa selama kegiatan belajar mengajar.
3. Sudah ada pemanfaatan teknologi *SMS Gateway* untuk pelaporan absensi siswa namun masih bisa dikembangkan.
4. Belum diketahui kelayakan dari sistem pelaporan *SMS Gateway* dikarenakan sistem yang digunakan masih baru di SMK N 4 Klaten.

C. Pembatasan Masalah

Berdasarkan identifikasi masalah, peneliti membatasi masalah pada belum adanya penggunaan *SMS Gateway* dalam bidang pendidikan terutama dalam hal absensi siswa dan kurangnya pengetahuan dan perhatian guru terhadap kualitas aplikasi yang digunakan.

D. Rumusan Masalah

Berdasarkan uraian diatas tentang pentingnya sistem pelaporan absensi siswa berbasis SMS *Gateway* untuk pengawasan orang tua di SMK N 4 Klaten, maka permasalahan dititikberatkan pada kemampuan aplikasi berbasis web dan mengangani pelaporan absensi siswa. Permasalahan yang diselesaikan dalam penelitian ini dapat dirumuskan sebagai berikut:

1. Bagaimana merancang dan membangun aplikasi pelaporan absensi SMS *Gateway* untuk pengawasan orang tua siswa?
2. Bagaimana mengetahui tingkat kualitas aplikasi yang digunakan?

E. Tujuan Penelitian

Berdasarkan rumusan masalah diatas, tujuan dari penelitian ini adalah:

1. Mampu merancang dan membangun aplikasi pelaporan absensi berbasis *SMS Gateway* untuk pengawasan orang tua siswa di SMK N 4 Klaten.
2. Memberikan pengetahuan mengenai tingkat kualitas aplikasi yang digunakan kepada guru SMK N 4 Klatensesuai standar ISO 25010 dari aspek *functional suitability, performance efficiency, usability, maintainability, dan compatibility*.

F. Spesifikasi produk yang dikembangkan

Produk yang dihasilkan dari penelitian ini adalah aplikasi pelaporan absensi berbasis *SMS Gateway* yang mempunyai fungsi sebagai berikut:

1. Memiliki 2 *user* yaitu *user* biasa (*administrator*) dan *superadmin*.
2. *User* biasa (*administrator*) dapat :
 - Melakukan manajemen absensi (menambah, menghapus, melihat, dan mengubah absensi/mengubah keterangan serta jam absensi).

3. *Superadmin* dapat :

- Melakukan manajemen absensi (sama seperti *administrator*).
- Melakukan manajemen *user* (menambah, mencari, mengubah, menghapus dan melihat data *user*).
- Melakukan *reset password* dan menentukan *user* yang menjadi *administrator* atau *superadmin*.
- Melakukan manajemen data siswa (menambah, mencari, mengubah, menghapus dan melihat data siswa).
- Melakukan manajemen jurusan (menambah, mencari, mengubah, menghapus dan melihat data jurusan).
- Melakukan set pesan yang akan dikirimkan.

Pengembangan dari absensi *SMS Gateway* yang dapat mengubah dan meng-set pesan yang akan dikirimkan kepada orang tua yang menerima pesan. Pesan tersebut dapat disesuaikan dengan kondisi anak tersebut ketika izin (misalnya mengikuti lomba, perkemahan, rapat OSIS dan lain-lain).

G. Manfaat Penelitian

Tugas akhir ini diharapkan dapat memberikan manfaat untuk kepentingan:

1. Teoritis

- a. Hasil penelitian ini dapat berguna dalam bidang pendidikan dan bagi ilmu pengetahuan dan teknologi.

- b. Hasil penelitian ini dapat menjadi bahan perbandingan dan referensi dalam pengembangan pembuatan aplikasi *SMS Gateway* dalam hal absensi siswa.

2. Praktis

- a. Bagi orang tua siswa
 - 1. Aplikasi yang telah dibuat dapat digunakan oleh orang tua siswa dalam mengawasi siswa selama di sekolah.
 - 2. Orang tua siswa akan mendapatkan informasi terkait kondisi siswa selama di sekolah.
- b. Bagi guru
 - 1. Guru dapat memberikan informasi kepada orang tua siswa.
 - 2. Guru dapat lebih efektif dan efisien dalam menyampaikan informasi absensi kepada orang tua siswa.
- c. Bagi pengembang
 - 1. Pengembang dapat membantu memecahkan masalah dalam bidang pendidikan yang berkaitan dengan absensi siswa.

BAB II

KAJIAN PUSTAKA

A. Kajian Teori

1. Presensi Siswa

Menurut Kamus Besar Bahasa Indonesia (KBBI), presensi memiliki arti kehadiran sedangkan absensi berarti ketidakhadiran. Kehadiran siswa di sekolah (*school attendance*) adalah kehadiran dan keikutsertaan siswa secara fisik dan mental terhadap aktivitas sekolah pada jam-jam efektif di sekolah. Sedangkan ketidakhadiran adalah ketiadaan partisipasi secara fisik siswa terhadap kegiatan-kegiatan sekolah. Pada jam efektif sekolah, siswa harus berada di sekolah. Kalau tidak ada di sekolah, siswa harus dapat memberikan keterangan yang sah serta diketahui oleh orang tua atau walinya (Akhmad Sudrajat, 2010).

Salah satu kegunaan absensi untuk pihak pelajar antara lain penghitungan kemungkinan pelajar untuk mengikuti ujian. Selain itu kegunaan informasi absensi bagi sekolah antara lain untuk melakukan evaluasi kepada pelajar terhadap suatu mata pelajaran. Dalam peraturan sekolah terdapat sanksi dalam hal kedisiplinan kehadiran. Bagi siswa yang melanggar hukuman berupa *point* yang diakumulasikan dari tiap pelanggaran yang dilakukan. Skorsing dan dikeluarkan dari sekolah menjadi sanksi terberat yang diterima oleh siswa yang melanggar.

Terdapat beberapa jenis absensi, yang membedakan jenis-jenis absensi tersebut adalah cara penggunaannya dan tingkat daya gunanya. Secara umum jenis-jenis absensi dapat dikelompokkan menjadi dua (Sofyan, 2015:19), yaitu:

- a. Absensi manual, adalah cara pengentrian kehadiran dengan cara menggunakan pena (tanda tangan).
- b. Absensi non manual, adalah suatu cara pengentrian kehadiran dengan menggunakan alat dan sistem terkomputerisasi, bisa menggunakan *barcode*, *finger print* atau dengan mengentrikan NIP dan sebagainya.

2. Teknologi SMS

SMS (*Short Message Service*) adalah pesan singkat yang dimiliki oleh sebuah *provider* telekomunikasi dalam bentuk pengiriman berupa layanan *text* ke alat telekomunikasi lain yaitu *handphone* (Sofyan, 2015:184). Layanan SMS banyak diaplikasikan pada sistem komunikasi tanpa kabel, memungkinkan dilakukannya pengiriman pesan dalam bentuk *alphanumeric* antar terminal pelanggan atau antar terminal pelanggan dengan sistem eksternal seperti *email*, *pagging*, *voice mail* dan lain-lain. Pada mulanya layanan SMS ini digunakan sebagai standar untuk telepon *wireless* berbasis GSM (*Global System for Mobile Communication*). Alasan utama dalam pemilihan SMS sebagai alat komunikasi adalah tidak membutuhkan *handphone* dengan spesifikasi yang tinggi, penggunaan lebih mudah dan tidak membutuhkan koneksi internet.

3. SMS Gateway

SMS Gateway adalah suatu *platform* yang menyediakan mekanisme untuk mengirim dan menerima SMS (Wahana, 2014:1). Pada *SMS Gateway* ini memungkinkan untuk berkomunikasi dengan perangkat lain yang memiliki *platformSMS*. *SMS Gateway* dapat mengirim dan menerima pesan SMS dengan sangat mudah karena antarmuka yang mudah dan standar digunakan.

Fitur yang biasa ditemukan pada *SMS Gateway* (Wahana, 2014:2):

- a. *Auto Reply*, fitur ini banyak digunakan pada program *Quiz* yang menggunakan format SMS tertentu. Contoh : Ketik REG, NSP1212
- b. *Broadcast Message*, fitur ini biasa disebut dengan pengiriman massal. Fitur ini banyak disediakan pada *handphone* dan juga *gadget*. Namun, fitur ini masih terdapat kekurangan yaitu SMS yang dikirim harus sama. Jika ingin mengirimkan SMS secara massal dan sekaligus menyebutkan nama pengirim didalam SMS, hal tersebut tidak dapat dilakukan.
- c. Pengiriman Terjadwal, fitur ini memungkinkan pengiriman SMS dalam waktu yang sudah ditetapkan. Fitur ini biasa digunakan pada ucapan selamat ulang tahun sebagai pesan pengingat.

4. Gammu

Gammu adalah semacam *service* yang disediakan untuk membangun aplikasi *SMS Gateway*. *Gammu* merupakan media penghubung antara komputer dengan perangkat telepon untuk dapat digunakan mengirim dan menerima pesan SMS (Wahana, 2014:3). Selain sebagai penghubung, *gammu* juga dapat mengelola fungsi *phonebook*. Para pengembang memilih *gammu* karena dapat dijalankan di sistem operasi Windows dan Linux. Selain sifatnya yang *open source*, *gammu* juga tidak memerlukan banyak *hardware* dan juga banyak perangkat yang kompatibel.

5. Codeigniter

Menurut Wahana Komputer (2014:4) *codeigniter* adalah sebuah *framework* bahasa pemrograman PHP. *Codeigniter* menawarkan kemudahan serta

standarisasi dalam proses pengembangan *website* dan aplikasi berbasis web. *Codeigniter* melakukan pengembangan *website* secara cepat dan terstruktur sehingga sangat berguna dalam pengembangan yang dikerjakan secara tim. Pada fasilitas *codeigniter* menyediakan *library* dan *helper* yang berguna dan mempermudah proses pengembangan. Para pengembang cukup memilih komponen-komponen lalu dikombinasikan menjadi sebuah *website* atau aplikasi yang diinginkan.

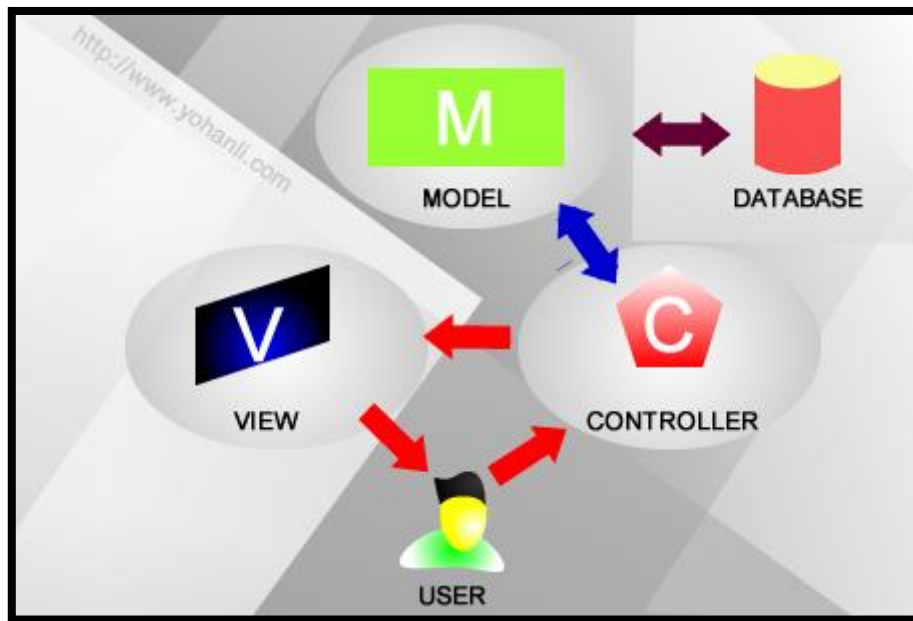
Kelebihan penggunaan *codeigniter* dalam pengembangan suatu proyek (Wahana, 2014:4):

- a. Menghemat waktu, dengan menggunakan *library* dan *helper* yang disediakan, pengembang hanya perlu memikirkan logika yang digunakan.
- b. *Code reuse*, *codeigniter* menggunakan standar baku/tetap yang dapat digunakan pada proyek yang akan datang.
- c. Komunitas, banyak komunitas yang dapat membantu memecahkan masalah dalam *codeigniter*.
- d. *Best practice*, pengembang *codeigniter* dapat meningkatkan kualitas pemrograman karena banyak kumpulan kode *best practice*.

6. MVC (Model View Controller)

MVC (Model *View Controller*) adalah metode pembuatan *software* yang memisahkan antara alur aplikasi dengan tampilan (Wahana, 2014:17). Intinya MVC meminimalisir koding PHP pada halaman web karena sudah dipisahkan menjadi *Controller* dan *View*. Pada pengolahan data juga menjadi lebih rapi karena dipisahkan kedalam *Controller* dan model.

- a. Model berfungsi untuk menangani urusan yang berhubungan dengan interaksi ke *database*. Model mempresentasikan struktur data dari aplikasi yang berupa basis data maupun data lain. Dalam model biasanya berisi *class* dan fungsi untuk mengambil, meng-*update*, dan menghapus data.
- b. *View* berhubungan dengan segala sesuatu yang akan ditampilkan kepada *user*. Pada *View* bisa berupa halaman web, RS, JSON dan lain-lain. Pada pemrograman *View* harus menghindari adanya pemrosesan logika atau data. *View* hanya berisi variable-variabel yang berisi data yang siap ditampilkan. Jadi *View* hanya dikhususkan untuk menampilkan data hasil model dan *Controller*.
- c. *Controller* bertindak sebagai penghubung antara data dan *View*. *Controller* terdapat *class* dan fungsi yang memproses permintaan dari *View* ke data di dalam model (Wahana, 2014:17-18). Pada *Controller* tidak boleh berisi kode untuk mengakses *database* karena tugas tersebut sudah ada pada model. Sementara *Controller* berisi variabel yang akan ditampilkan di *View* dan juga memanggil model untuk melakukan akses basis data. Selain itu *Controller* juga menyediakan penanganan *error/kesalahan*. Mengerjakan proses logika serta melakukan *validasi* atau pengecekan input juga merupakan tugas *Controller*.



Gambar 1. Konsep MVC

Alur kerja konsep MVC, yaitu ketika *user* melakukan permintaan halaman web ke *web server* yang pertama kali dijalankan adalah skrip *Controller*. Kemudian *Controller* melakukan *validasi* apakah memerlukan informasi dari *database* atau tidak. Jika diperlukan *Controller* memanggil skrip model. Model melakukan pengolahan basis data lalu mengembalikan hasilnya kepada *Controller*. *Controller* yang mempunyai hasil *database* dari model tadi mengalokasikan ke *View* untuk ditampilkan kepada *user*.

7. Bahasa Pemrograman PHP

PHP adalah singkatan dari *Hypertext Preprocessor* (suatu akronif rekursif) yang dibangun oleh Rasmus Lerdorf pada tahun 1994. Pada awal pengembangannya PHP disebut sebagai *Personal Home Page*. PHP merupakan bahasa pemrograman yang disisipkan dalam HTML dan bersifat *open source* sehingga dapat mengakses *source code* tanpa harus membayar sepeserpun. PHP banyak digunakan untuk pembuatan situs web yang bersifat dinamis (Didik, 2008).

Kelebihan bahasa pemrograman PHP dibandingkan bahasa pemrograman yang lainnya (Madcoms, 2004:2):

- a. Bahasa pemrograman PHP merupakan sebuah bahasa *script* yang tidak melakukan kompilasi dalam penggunaannya.
- b. *Server* web yang mendukung PHP dapat ditemukan di mana-mana mulai dari IIS sampai dengan apache, dengan konfigurasi yang relative mudah.
- c. Dalam sisi pengembangan lebih mudah, karena banyaknya *developer* yang siap membantu dalam pengembangan.
- d. Dalam sisi pemahaman, PHP adalah bahasa *scripting* yang paling mudah karena referensinya banyak.

Selain itu PHP juga memiliki tingkat akses yang lebih cepat dibandingkan bahasa pemrograman yang lainnya. Banyak *developer* yang menggunakan bahasa pemrograman PHP salah satunya karena PHP mendukung akses ke beberapa *database* yang sudah ada, baik yang *free* maupun yang komersial. Contohnya: MySQL, PostgreSQL, mSQL, Infomix dan MicrosoftSQL *server*.

8. MySQL& Apache

Apache merupakan web server yang dikembangkan oleh *The Apache Software Foundation* melalui sebuah proyek bernama HTTP Server Project. HTTP Server Project adalah usaha pengembangan perangkat lunak kolaboratif bertujuan untuk menciptakan kekuatan, tingkat komersial, banyak fitur, dan tersedia secara gratis yang merupakan implementasi dari kode sumber dari HTTP server. Proyek ini dikelola bersama oleh sekelompok sukarelawan yang terletak di seluruh dunia, dengan menggunakan internet dan web untuk berkomunikasi,

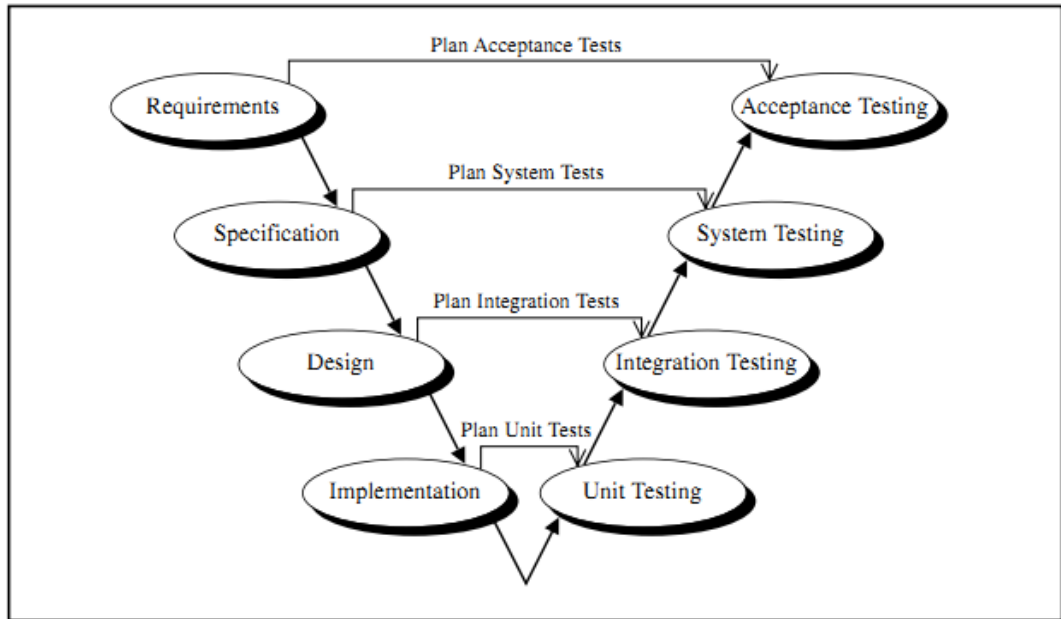
merencanakan, dan mengembangkan server dan dokumentasi terkait (Apache, 2015).

MySQL (*My Structured Query Language*) adalah sebuah program pembuat *database* yang bersifat *open source*, artinya bebas digunakan dan *free* (Bunafit, 2004:29). MySQL merupakan suatu sistem manajemen basis data relasional (RDBMS-*Relational Database Management System*) yang mampu bekerja dengan cepat, kokoh dan mudah digunakan. Basis data merupakan system komputerisasi untuk memelihara informasi dan membuat informasi tersedia ketika dibutuhkan (Dessy dan Yuniar, 2014). Basis data memungkinkan untuk menyimpan, menelusuri, mengurutkan, dan mengambil data secara efisien. Bahasa yang digunakan oleh MySQL adalah standar SQL (standar bahasa basis data relasional di seluruh dunia saat ini).

9. Metode Pengembangan Sistem

SDLC (*Software Development Life Cycle* atau sering disebut *System Development Life Cycle*) adalah proses mengembangkan atau mengubah suatu sistem perangkat lunak dengan menggunakan model-model dan metodologi yang digunakan orang untuk mengembangkan sistem-sistem perangkat lunak sebelumnya (S, Rosa, 2013:26). Proses pengembangan system melewati beberapa tahapan dari mulai system itu direncanakan sampai dengan system tersebut diterapkan, dioperasikan, dan dipelihara (Syahrina Ramadhina, 2015).

Terdapat beberapa model yang bisa digunakan untuk pengembangan suatu aplikasi. Salah satunya adalah V-model (*Validation & Verification Model*) adalah sebuah pengembangan dari model *waterfall*.



Gambar 2. Model V (Watkins, 2011:146)

Model-V menggambarkan relasi aksi-aksi jaminan kualitas pada aksi yang berkaitan dengan komunikasi permodelan, serta aktivitas konstruksi lebih awal. Model-V pada dasarnya menyediakan cara secara visual bagaimana tindakan verifikasi dan validasi diterapkan pada bagian-bagian rekayasa perangkat lunak yang lebih awal (Pressman, 2012:46). Pemilihan model-V dalam penelitian ini karena model ini mendukung *tailoring project* (pekerjaan proyek) dan penambahan serta pengurangan *method* atau *tool* dilakukan secara dinamik. Model-V memudahkan pengembang dalam menambah atau mengurangi *method* atau *tool* agar sesuai dengan suatu proyek tertentu.

a. Tahap Pengembangan

i. *Requirement*

Merupakan frase pertama dalam pengembangan model-V yang digunakan untuk mengetahui aplikasi yang diinginkan oleh *user*. Pada tahap ini memiliki peran yang sangat krusial dalam analisis kebutuhan dari aplikasi

yang dikembangkan. Hasil dari tahap *requirement* ini adalah dokumentasi kebutuhan *user*.

ii. Specification

Merupakan frase dimana pengembang mendapatkan spesifikasi yang diinginkan dari *user*. Spesifikasi digunakan untuk mendapatkan hasil yang diinginkan *user* saat analisis kebutuhan. Keluaran dari tahap *specification* adalah mendapatkan spesifikasi dari produk yang dikembangkan.

iii. Design

Pada fase ini semua sistem internal dan external sudah memiliki kerangka sistem yang dikembangkan. Pada tahap desain ini *developer* mengembangkan spesifikasi yang telah di setujui oleh *user*. Keluaran dari tahap design adalah penggambaran produk yang dikembangkan sebelum diimplementasikan.

iv. Implementation

Pada tahap *implementation* secara keseluruhan yang terjadi adalah masalah pengkodean (koding). Tahap implementasi mengacu pada desain yang dibuat berdasarkan spesifikasi dari *user*.

b. Tahapan pengujian

Pengujian adalah serangkaian kegiatan yang dapat direncanakan di muka (awal) dan dilakukan secara sistematis (Pressman, 2012:550). Sedangkan menurut Rosa (2015:272), pengujian adalah satu set aktivitas yang direncanakan dan sistematis untuk menguji atau mengevaluasi kebenaran yang diinginkan. Pengujian perangkat lunak adalah sebuah elemen topik yang

memiliki cakupan luas dan sering di kaitkan dengan verifikasi (*verification*) dan validasi (*validation*) (V&V).

Verifikasi adalah sekumpulan tugas yang memastikan bahwa perangkat lunak benar menerapkan fungsi yang ditentukan, sedangkan validasi adalah sekumpulan tugas yang berbeda yang memastikan bahwa perangkat lunak yang telah dikembangkan sesuai dengan persyaratan pengguna (Pressman, 2012:551).

Tahapan pengujian secara keseluruhan adalah sebagai berikut (Rosa,2013:274):



Gambar 3 Tahapan pengujian

i. Pengujian unit (*unit testing*)

Pengujian ini berkaitan dengan fungsi, prosedur atau komponen yang terdapat dalam sistem. Pengujian ini berfokus pada upaya verifikasi terhadap unit terkecil dari perancangan perangkat lunak, yaitu komponen atau modul (Pressman, 2012:557). Pengujian unit dilakukan oleh *developer* atau pengembang sendiri dengan tujuan untuk menghasilkan produk yang sesuai. Pengujian unit dapat digunakan untuk mengukur *functional*, *performance*, *strees* dan *reliability* perangkat lunak yang dilakukan dengan teknik *white box testing* (Watkins, 2011:50).

- ii. Pengujian integrasi (*Integration Testing*)

Menurut Pressman (2011:560) pengujian integrasi adalah teknik sistematis untuk membangun arsitektur perangkat lunak, pada saat yang bersamaan dilakukan pengujian untuk menemukan kesalahan-kesalahan yang terkait dengan antarmuka yang bertujuan untuk mengambil komponen yang diuji dan membangun struktur program yang telah ditentukan oleh perancang. Pengujian integrasi adalah pengujian yang dilakukan kepada semua unit yang saling berhubungan dengan fokus pada antarmuka. Pengujian integrasi dapat digunakan untuk mengukur *functional*, *performance*, dan *reliability* perangkat lunak yang dilakukan dengan teknik *black box testing* (Watkins, 2011:57).
- iii. Pengujian sistem (*System Testing*)

Pengujian sistem merupakan pengujian dimana unit-unit proses yang sudah terintegrasi diuji dengan antarmuka yang sudah dibuat sehingga pengujian ini dimaksudkan untuk menguji sistem perangkat lunak secara keseluruhan dan diuji secara satu sistem (Rosa, 2015:275). Pengujian sistem dapat digunakan untuk mengukur volume, *performance* dan *stress* perangkat lunak yang dilakukan dengan teknik *black box testing* (Watkins, 2011:63).
- iv. Pengujian penerimaan (*Acceptance Testing*)

Pengujian penerimaan digunakan untuk mengetahui kepuasan pelanggan atau *user* terhadap perangkat lunak yang telah dikembangkan (Rosa, 2015:275). Pengujian penerimaan dapat digunakan untuk mengukur

usability perangkat lunak yang dilakukan dengan teknik *black box testing* (Watkins, 2011:69-70).

c. Teknik atau metode pengujian

Menurut Rosa (2015,275) pengujian perangkat lunak ada 2 pendekatan, yaitu *white box testing* dan *black box testing*

i. *White box testing*

White box testing merupakan pengujian perangkat lunak dari segi desain dan kode program apakah mampu menghasilkan fungsi-fungsi, masukan, dan keluaran yang sesuai dengan spesifikasi kebutuhan. Pengujian *white box testing* dilakukan oleh *tester* terhadap cara kerja internal dari perangkat yang dikembangkan.

ii. *Black box testing*

Black box testing berkaitan dengan pengujian yang dilakukan pada antar muka perangkat lunak (Pressman, 2012:587). Pengujian ini berfungsi untuk mengetahui apakah fungsi-fungsi, masukan dan keluaran dari perangkat lunak sesuai dengan spesifikasi kebutuhan (Rosa, 2015:275).

Black box testing merupakan pengujian yang dilakukan pada *user interface* tanpa mengetahui internal dari produk tersebut.

Terdapat dua jenis pengujian validasi yaitu *alpha testing* dan *beta testing*.

Alpha testing dilakukan dari sisi pengembang oleh sekelompok perwakilan dari pengguna akhir, sedangkan *beta testing* dilakukan oleh satu atau lebih dari pengguna akhir (Pressman, 2011:570).

10. Parameter Pengujian Perangkat Lunak

Pengujian untuk mengetahui kualitas perangkat lunak dapat dilakukan dengan berbagai model, diantaranya ISO 25010, McCall, Bohem dan Dromey. Dalam proses pengujian kualitas perangkat lunak diperlukan sebuah parameter yang nantinya digunakan untuk melakukan uji kelayakan perangkat lunak yang dibuat. Pengujian diperlukan tidak hanya untuk meminimalisir kesalahan secara teknis tapi juga kesalahan non teknis (Rosa, 2013:272).

Kualitas perangkat lunak dapat dinilai melalui parameter dan metode tertentu serta melalui pengujian *software*. Salah satu tolak ukur kualitas perangkat lunak adalah ISO 25010, yang dibuat oleh ISO (*International Organization for Standardization*) dan IEC (*International Electrotechnical Commission*).

Pada penelitian ini menggunakan ISO 25010 dalam melakukan kualitas perangkat lunak.

Tabel 1. ISO 25010 (ISO,2011)

No	Faktor	Sub faktor
1	<i>Functional suitability</i>	<i>Functional completeness</i>
		<i>Functional correctness</i>
		<i>Functional appropriateness</i>
2	<i>Performance efficiency</i>	<i>Time behavior</i>
		<i>Resource utilization</i>
		<i>Capacity</i>
3	<i>Compatibility</i>	<i>Co-existence</i>
		<i>Interoperability</i>
4	<i>Usability</i>	<i>Appropriateness recognizability</i>
		<i>Learnability</i>
		<i>Operability</i>
		<i>User error protection</i>
		<i>User interface aesthetics</i>
5	<i>Reliability</i>	<i>Accessibility</i>
		<i>Maturity</i>
		<i>Availability</i>
		<i>Fault tolerance</i>
		<i>Recoverability</i>

No	Faktor	Sub faktor
6	Security	<i>Confidentiality</i>
		<i>Integrity</i>
		<i>Non-repudiation</i>
		<i>Accountability</i>
		<i>Authenticity</i>
7	Maintainability	<i>Modularity</i>
		<i>Reusability</i>
		<i>Analiysability</i>
		<i>Modifiability</i>
		<i>Testability</i>
8	Portability	<i>Adaptability</i>
		<i>Instability</i>
		<i>Replaceability</i>

ISO 25010 mendefinisikan sebuah model kualitas produk yang terdiri dari delapan karekeristik (yang dibagi lagi menjadi sub karekeristik) yang berhubungan dengan sifat statis perangkat lunak dan sifat dinamis dari sistem komputer (ISO/IEC 2505:2011). Perangkat lunak yang dikembangkan akan diuji menggunakan lima dari delapan aspek didalam ISO 25010 yakni *functional suitability*, *performance efficiency*, *compatibility*, *maintainability*, dan *usability*. Pemilihan kelima aspek yang duji tersebut berdasarkan kesesuaian aspek pengujian dengan sistem yang terdapat pada aplikasi yang dikembangkan. Penjelasan dari kelima karakteristik tersebut adalah:

a. *Functional suitability* (kesesuaian fungsionalitas)

Aspek *functional suitability* merupakan tingkat dimana perangkat lunak dapat menyediakan fungsionalitas yang dibutuhkan ketika perangkat lunak digunakan (ISO 25010, 2011). Terdapat tiga indikator *functional suitability* (ISO 25010, 2011), yaitu:

- 1) *Functional completeness*, sejauh mana perangkat lunak dapat mencakup semua tugas dan tujuan pengguna.

- 2) *Functional correctness*, sejauh mana perangkat lunak dapat memberikan hasil yang tepat dan teliti terhadap tingkat kebutuhan.
- 3) *Functional appropriateness*, sejauh mana fungsi memfasilitasi pemenuhan tujuan tertentu. Seperti pengguna hanya diberikan langkah-langkah penting untuk melakukan perintah tertentu tanpa melalui langkah-langkah yang tidak diperlukan.

Pengujian *functionality* menguji bagaimana kerja setiap fungsi yang sudah dirancang bisa berjalan dengan baik. *Functional testing* dilakukan untuk memastikan spesifikasi fungsionalitas pada analisis kebutuhan berjalan dengan baik (Williams, 2006,39). Pada pengujian ini dilakukan dengan melakukan tes pada setiap fungsi perangkat lunak yang didokumentasikan dalam *test case*.

b. *Performance efficiency*

Menurut ISO 25010 (2011) mendefinisikan tingkat kinerja relatif terhadap sumber daya yang digunakan dalam kondisi yang ditetapkan. Sedangkan menurut Pressman (2012:673) pengujian kinerja dilakukan untuk memastikan sistem dapat menangani beban ekstrem tanpa penurunan kemampuan operasi yang tidak dapat diterima. Pengujian kinerja digunakan untuk mengungkap masalah-masalah kinerja yang muncul akibat kurangnya sumber daya pada server, *bandwidth* jaringan yang kurang, kemampuan basis data yang tidak memadai, kemampuan sistem operasi yang rusak atau lemah, fungsionalitas aplikasi yang dirancang buruk, dan masalah perangkat keras lainnya atau perangkat lunak yang mengakibatkan rusaknya kinerja *client-server*.

Pengujian aspek *performance efficiency* menggunakan aplikasi GTMetrik karena aplikasi tersebut menghasilkan dua pengujian berdasarkan aturan YSlow dan PageSpeed. Aturan YSlow dikembangkan oleh Yahoo Developer Network dan PageSpeed Insights dikembangkan oleh Google.

c. *Usability*

Aspek *usability* merupakan sejauh mana sistem dapat digunakan oleh pengguna tertentu untuk memperoleh tujuan tertentu dengan efektif, efisien dan memuaskan (ISO 25010, 2011). Indikator dalam aspek ini (ISO 25010, 2011):

- 1) *Appropriatness recognizability*, kemampuan perangkat lunak untuk dianalisis oleh pengguna terkait dengan pemenuhan kebutuhan.
- 2) *Learnability*, sejauh mana perangkat lunak memberikan kemudahan untuk dipelajari penggunaannya oleh pengguna (*user*).
- 3) *Operability*, sejauh mana perangkat lunak dapat digunakan dan dioperasikan oleh pengguna dengan mudah.
- 4) *User error protection*, sejauh mana perangkat lunak dapat menghindarkan pengguna dari kesalahan.
- 5) *User interface aesthetics*, sejauh mana inteaktivitas tampilan pada perangkat lunak dalam memberikan interaksi yang menyenangkan dan memuaskan bagi pengguna.
- 6) *Accessibility*, sejauh mana pengguna perangkat lunak berdasarkan perbedaan karakteristik penggunaannya untuk mencapai tujuan tertentu.

Usability dapat didefinisikan sebagai kemudahan dalam penggunaan.

Usability merupakan usaha yang dibutuhkan untuk mempelajari,

mengoperasikan, menyiapkan input, dan menginterpretasikan output suatu program. Semakin baik tingkat usability maka semakin mudah dan nyaman program tersebut digunakan.

James R. Lewis dan timnya dalam dokumen yang diterbitkan oleh IBM yang berjudul "*IBM Computer Usability Satisfaction Questionnaires: Psychometrics Evaluation and Instruction for Use*". Dalam dokumen tersebut, J R. Lewis menyusun sebuah kuisisioner yang bernama *The Computer System Usability Questionnaire* (CSUQ).

Tabel 2. *The Computer System Usability Questionnaire* (CSUQ)

No	Kriteria	Pernyataan
1	<i>Operability</i>	Secara keseluruhan, saya merasa puas dengan kemudahan penggunaan sistem ini.
2	<i>Operability</i>	Cara penggunaan sistem ini sangat simpel.
3	<i>Appropriateness recognizability</i>	Saya dapat menyelesaikan tugas saya dengan efektif ketika menggunakan sistem ini.
4	<i>Appropriateness recognizability</i>	Saya dapat dengan cepat menyelesaikan pekerjaan saya menggunakan sistem ini.
5	<i>Appropriateness recognizability</i>	Saya dapat menyelesaikan tugas saya dengan efisien ketika menggunakan sistem ini.
6	<i>Learnability</i>	Saya merasa nyaman menggunakan sistem ini.
7	<i>Learnability</i>	Sistem ini sangat mudah dipelajari.
8	<i>Appropriateness recognizability</i>	Saya yakin saya akan lebih produktif ketika menggunakan sistem ini.
9	<i>User error protection</i>	Jika terjadi <i>error</i> , sistem ini memberikan pesan pemberitahuan tentang langkah yang saya lakukan untuk mengatasi masalah.
10	<i>User error protection</i>	Kapanpun saya melakukan kesalahan, saya bisa kembali dan pulih dengan cepat.
11	<i>Operability</i>	Informasi yang disediakan sistem ini sangat jelas.
12	<i>Operability</i>	Mudah untuk menemukan informasi yang saya butuhkan.
13	<i>Accessibility</i>	Informasi yang diberikan oleh sistem ini sangat mudah dipahami.
14	<i>Accessibility</i>	Informasi yang diberikan sangat efektif dalam membantu menyelesaikan pekerjaan saya.
15	<i>User interface aesthetics</i>	Tata letak informasi yang terdapat di layar monitor sangat jelas.

No	Kriteria	Pernyataan
16	<i>User interface aesthetics</i>	Tampilan sistem ini sangat memudahkan.
17	<i>User interface aesthetics</i>	Saya suka menggunakan tampilan sistem semacam ini.
18	<i>Appropriateness recognizability</i>	Sistem ini memberikan semua fungsi dan kapabilitas yang saya perlukan.
19	<i>Appropriateness recognizability</i>	Secara keseluruhan, saya sangat puas dengan kinerja sistem ini.

d. *Maintability,*

Pengujian aspek *maintainability* yaitu tingkat efektivitas dan efisiensi pada suatu produk atau sistem untuk dapat dimodifikasi oleh pengembang (ISO 25010, 2011). Modifikasi mencakup koreksi, perbaikan atau adaptasi terhadap perubahan lingkungan, persyaratan dan spesifikasi (ISO 25010, 2011). Terdapat indikator dalam aspek ini (ISO 25010, 2011):

- 1) *Modularity*, tingkat dimana sistem atau program terdiri dari komponen yang berlainan sehingga perubahan terhadap satu komponen minimal memiliki pengaruh terhadap komponen lain.
- 2) *Reusability*, tingkat dimana sebuah *asset* dapat digunakan pada lebih dari satu sistem perangkat lunak pada pembangunan asset lainnya.
- 3) *Analyzability*, merupakan kemampuan dimana perangkat lunak dapat dianalisis untuk mengetahui apa yang menyebabkan kegagalan pada perangkat lunak atau untuk mengidentifikasi bagian yang dapat dimodifikasi.
- 4) *Changeability*, merupakan tingkat dimana perangkat lunak memungkinkan sebuah modifikasi yang spesifik untuk dapat

diimplementasikan. Hal ini bertujuan untuk memudahkan dalam modifikasi.

- 5) *Testability*, tingkat dimana modifikasi perangkat lunak untuk dilakukan validasi.
- 6) *Compliance*, merupakan tingkatan dimana sebuah perangkat lunak dapat memenuhi standarisasi yang berhubungan dengan *maintainability* pada perangkat lunak lain.

Dalam penelitian ini yang dimaksud pengujian aspek *maintainability* merupakan pengujian secara operasional untuk mengetahui apabila terjadi kegagalan dalam input data, tampilan sistem, dan apakah sistem mudah untuk dilakukan modifikasi. Pengujian ini dilakukan secara operasional dengan menggunakan instrumen *maintainability* (Land, 2002). Aspek yang terdapat dalam instrumen tersebut adalah *instrumentation*, *consistency*, dan *simplicity*.

Tabel 3. Instrumen *maintainability* (Land, 2002)

Aspek	Aspek yang dinilai	Hasil yang akan diperoleh
<i>instrumentation</i>	Terdapat peringatan pada sistem pengolahan data untuk mengidentifikasi kesalahan	Ketika <i>user</i> melakukan kesalahan dalam operasionalnya, sistem akan memberikan peringatan.
<i>consistency</i>	Penggunaan satu bentuk perancangan pada seluruh rancangan sistem.	Bentuk rancangan sistem pengolahan data mempunyai satu bentuk yang sama.
<i>simplicity</i>	Kemudahan dalam pengolahan, perbaikan, dan pengembangan sistem.	Mudah untuk diolah, diperbaiki, dan dikembangkan. Hal ini dapat diidentifikasi melalui penulisan kode program.

e. *Compatibility*

Aspek *compatibility* merupakan kemampuan dari dua atau lebih komponen perangkat lunak untuk dapat melakukan pertukaran informasi dan melakukan fungsi yang dibutuhkan ketika digunakan pada *hardware* atau lingkungan perangkat lunak yang sama.

- 1) *Co-existence*, merupakan tingkat dimana perangkat lunak dapat berdampingan dengan perangkat lain dan berbagai sumber daya pada lingkungan yang sama tanpa dampak yang merugikan.
- 2) *Interoperability*, merupakan tingkat dimana perangkat lunak dapat beroperasi secara kooperatif dengan satu atau lebih perangkat lunak lain.

Pada pengujian *compatibility* menjadi sangat penting dalam konteks perangkat lunak berbasis *web* karena *user* menggunakan *browser*. Pada pengujian ini dibutuhkan untuk mengetahui seberapa cocok perangkat lunak dapat dijalankan pada beberapa *browser* yang mungkin digunakan oleh *user*. *Browser* yang digunakan pada pengujian ini adalah chrome, mozilla firefox dan safari.

f. *Portability*

Aspek *portability* merupakan tingkat efektivitas dan efisiensi dengan sistem, produk atau komponen dapat ditransfer dari satu perangkat keras, perangkat lunak atau lingkungan operasional atau penggunaan ke yang lainnya (ISO 25010, 2011). Terdapat indikator dalam aspek ini (ISO 25010, 2011), yaitu:

- 1) *Adaptability*, sejauh mana sistem beradaptasi dengan *hardware*, *software*, atau lingkungan yang bervariasi.
- 2) *Instability*, sejauh mana sistem dapat di-*install* ataupun di-*uninstall* dengan baik dalam berbagai kondisi lingkungan perangkat.
- 3) *Replaceability*, sejauh mana perangkat lunak dapat menggantikan produk lain yang memiliki kesamaan, selain itu *replaceability* dapat diartikan pada kemampuan aplikasi untuk dapat di-*update* ketika versi baru dari aplikasi tersebut sudah di-*release*.

g. Reliability

Merupakan tingkatan dimana perangkat lunak dapat bertahan pada tingkatan tertentu ketika digunakan oleh pengguna pada kondisi yang spesifik.

- 1) *Availability*, merupakan tingkatan dimana komponen perangkat lunak dapat beroperasi dan siap ketika dibutuhkan untuk digunakan.
- 2) *Fault tolerance*, merupakan tingkat dimana sebuah perangkat lunak dapat bertahan pada tingkat kemampuan tertentu terhadap kegagalan atau kesalahan yang terdapat pada perangkat lunak.
- 3) *Recoverability*, merupakan tingkat dimana perangkat lunak dapat memenuhi standar ketahanan yang dimiliki oleh perangkat lunak sejenis.

h. Security

Merupakan perlindungan terhadap perangkat lunak dari berbagai ancaman, akses atau penggunaan dari pengguna yang tidak dikenal.

- 1) *Confidentiality*, merupakan tingkat dimana perangkat lunak memiliki perlindungan terhadap data atau informasi dari pengguna, apakah sebagai ancaman atau kejengajaan.
- 2) *Integrity*, merupakan tingkat dimana kelengkapan dan ketepatan dari sejumlah *asset* telah dijaga.
- 3) *Non-repudiation*, merupakan tingkatan dimana aksi atau tindakan yang dilakukan telah terbukti sehingga hal tersebut tidak dapat ditolak.
- 4) *Accountability*, merupakan tingkatan dimana aksi dari sebuah entitas dapat ditelusuri keunikannya terhadap entitas.
- 5) *Authenticity*, merupakan tingkat dimana identitas dari sebuah subjek atau sumber dapat diterima.

ISO 25010 merupakan standar kualitas yang menggantikan ISO 9126 yang merupakan standar penting untuk perangkat lunak. Sedangkan pada ISO 25010 menetapkan delapan karakteristik yaitu *functional suitability, reliability, performance efficiency, security, compability, maintainability, dan compatibility*. Perubahan ICT yang sangat cepat dan sangat berbeda dari satu dekade yang lalu merupakan salah satu alasan mengapa ISO 9126 sudah tidak relevan lagi untuk dijadikan standar dalam melakukan pengujian kualitas perangkat lunak. Memory yang besar, tampilan yang lebih baik, dan processor yang lebih cepat memungkinkan pengembangan sistem aplikasi baru yang juga membutuhkan pengujian kualitas yang berbeda yaitu ISO 25010 (Veenendaal, 2014).

Model ISO 25010 memiliki kriteria penilaian yang lebih spesifik. Oleh karena itu pada penelitian ini model kualitas yang digunakan adalah model ISO 25010. Matriks faktor kualitas ISO 9126 dan ISO 25010 dapat dilihat pada lampiran 3.





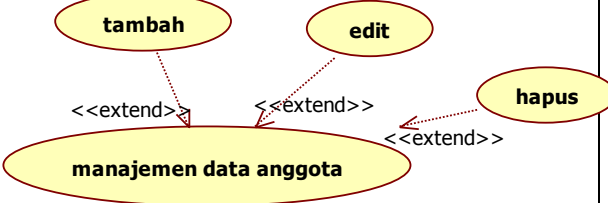
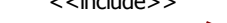
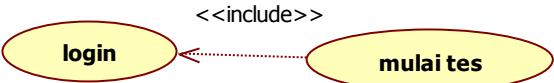
11. Desain Produk

a. Permodelan UML (*Unified Modelling Language*)

“UML muncul karena adanya kebutuhan permodelan visual untuk mempresentasikan, menggambarkan, membangun, dan dokumentasi dari sistem perangkat lunak. UML merupakan bahasa visual untuk permodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks pendukung. UML hanya berfungsi untuk melakukan permodelan dan penggunaan UML tidak terbatas pada metodologi tertentu.”(S, Rosa, 2013: 137).

UML adalah bahasa yang digunakan untuk memodelkan perangkat lunak. Permodelan UML meliputi pembuatan *use case* diagram. *Use case* digunakan untuk mengetahui dan menggambarkan apa yang dilakukan sistem dilihat dari sudut pandang *user/pengguna*. Elemen utama dari *use case* ada empat, yaitu: sistem, aktor yang berinteraksi dalam sistem, *use case* atau *service* yang tersedia dalam sistem, dan relasi antar elemen. Sistem direpresentasikan sebagai kotak. Nama sistem ditulis di bagian bawah atau di dalam kotak. *Use case* menggambarkan urutan transaksi dari aktor pada sistem.

Tabel 4. Simbol-simbol *use case* (S, Rosa, 2013: 156-158)

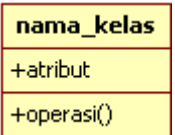






Simbol	Deskripsi
<p><i>Use case</i></p> 	<p>fungsi yang disediakan sistem sebagai unit – unit yang saling bertukar pesan antar unit atau aktor; biasanya dinyatakan dengan menggunakan kata kerja diawal frase nama <i>use case</i></p>
<p>Aktor / <i>Actor</i></p>  <p>nama aktor</p>	<p>orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang, biasanya dinyatakan menggunakan kata benda di awal frase nama aktor</p>
<p>Asosiasi / <i>association</i></p> 	<p>komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor</p>
<p>Ekstensi / <i>extend</i></p> <p><<extend>></p> 	<p>relasi tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan itu; di dalam extend, suatu <i>use case</i> hanya dapat diakses apabila telah melewati <i>use case</i> lainnya, misal :</p>  <p><i>Use case</i> tambah, edit, dan hapus adalah <i>use case</i> tambahan dari <i>use case</i> manajemen data anggota.</p>
<p>Menggunakan / <i>include</i> / <i>uses</i></p> <p><<include>></p> 	<p>relasi tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsinya atau sebagai syarat dijalankan <i>use case</i> ini, contoh :</p>  <p><i>Use case</i> mulai tes dapat dijalankan apabila <i>use case</i> login sudah dijalankan.</p>

12. Class Diagram

Class diagram atau diagram kelas menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Diagram kelas dibuat agar pembuat program (*programmer*) membuat kelas-

kelas sesuai rancangan di dalam diagram kelas agar antara dokumentasi perancangan dan perangkat lunak sinkron (S, Rosa, 2013: 141-142).

Tabel 5. Simbol-simbol *class* diagram (S, Rosa, 2013: 146-147)


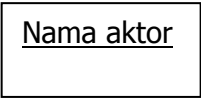

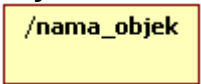


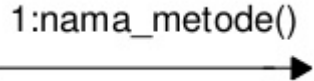
Simbol	Deskripsi
Kelas 	Kelas pada struktur sistem
Antarmuka/ <i>interface</i> 	Sama dengan konsep <i>interface</i> dalam pemrograman berorientasi objek
Asosiasi/ <i>association</i> 	relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i>
Asosiasi berarah/ <i>directed association</i> 	Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya disertai dengan <i>multiplicity</i>
Generalisasi 	Relasi antar kelas dengan makna generalisasi-spesialisasi (umum-khusus)
Kebergantungan/ <i>dependency</i> 	Relasi antar kelas dengan makna kebergantungan antar kelas
Agregasi/ <i>aggregation</i> 	Relasi antar kelas dengan makna semua-bagian (<i>whole-part</i>)

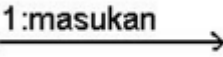
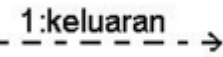
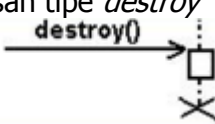
13. Sequence Diagram

Menurut S, Rosa (2013:165) Diagram sekuen menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek. Jadi dalam menggambar *sequence* diagram harus diketahui objek-objek yang terlibat dalam sebuah

use case beserta metode-metode yang dimiliki kelas yang diinstansiasi menjadi objek tersebut. Selain itu membuat diagram *sequence* juga dibutuhkan untuk melihat skenario yang ada pada *use case*.

Tabel 6. Simbol-simbol *sequence* diagram (S, Rosa, 2013: 165-167)





Simbol	Deskripsi
<p>Aktor</p>  <p>atau</p>  <p>Tanpa waktu aktif</p> <p>Garis hidup/<i>lifeline</i></p> 	<p>Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi itu sendiri. Jadi walaupun simbol dari aktor adalah orang, tapi aktor belum tentu merupakan orang. Biasanya dinyatakan menggunakan kata benda di awal fase nama aktor.</p>
<p>Objek</p> 	<p>Menyatakan objek yang berinteraksi pesan.</p>
<p>Waktu aktif</p> 	<p>Menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan didalamnya.</p>
<p>Pesan tipe <i>create</i></p> 	<p>Menyatakan suatu objek membuat objek yang lain., arah panah mengarah pada objek yang dibuat.</p>
<p>Pesan tipe <i>call</i></p> 	<p>Menyatakan suatu objek memanggil operasi/metode yang ada pada objek lain atau dirinya sendiri, arah panah mengarah pada objek yang memiliki operasi/metode, karena ini memanggil operasi/metode maka operasi/metode yang dipanggil harus ada pada diagram kelas sesuai dengan kelas objek yang berinteraksi.</p>


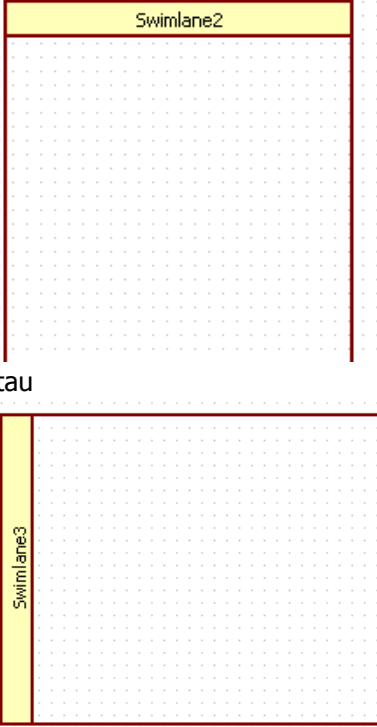
Simbol	Deskripsi
Pesan tipe <i>send</i> 	Menyatakan bahwa suatu objek mengirimkan data/masukkan/informasi ke objek lainnya, arah panah mengarah pada objek yang dikirim.
Pesan tipe <i>return</i> 	Menyatakan bahwa suatu objek yang telah menjalankan suatu operasi atau metode menghasilkan suatu kembalian ke objek tertentu, arah panah mengarah pada objek yang menerima kembalian.
Pesan tipe <i>destroy</i> 	Menyatakan suatu objek mengakhiri hidup objek yang lain, arah panah mengarah pada objek yang diakhiri, sebaiknya jika ada <i>create</i> maka ada <i>destroy</i> .

14. Activity Digram

Activity diagram atau diagram aktivitas menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Jadi diagram aktivitas menggambarkan aktivitas yang dilakukan oleh sistem, bukan yang dilakukan aktor.

Tabel 7. Simbol-simbol diagram aktivitas (S, Rosa, 2013: 162-163)

Simbol	Deskripsi
Status awal 	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal.
Aktivitas 	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
Pencabangan / <i>decision</i> 	Asosiasi pencabangan dimana jika ada pilihan aktivitas lebih dari satu.
Penggabungan / <i>join</i> 	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu.

Simbol	Deskripsi
Status akhir 	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.
Swimlane 	Memisahkan organisasi yang bertanggung jawab terhadap aktivitas yang terjadi.

B. Penelitian yang Relevan

- a. Penelitian yang dilakukan oleh (Tommy, STMIK AMIKOM Yogyakarta, 2013) membuat sebuah *application* absensi berbasis web menggunakan *framework Codeigniter* yang mengadakan penelitiannya di SMP BOBKRI 1 Wates, membuat sistem absensi dengan mengumpulkan data-data, diolah, disimpan, atau dilihat kembali hingga menghasilkan informasi. Semua hal tersebut juga merupakan bagian dari kebutuhan sekolah untuk mengelola absensi kehadiran siswa yang ada setiap hari dan memberikan sebuah inovasi teknologi informasi yang baru.

- b. Penelitian yang dilakukan oleh (Yuyiandrika, 2013) membangun system informasi presensi mahasiswa STMIK Atma Luhur Pangkal Pinang Berbasis Web Informasi. Tentang absensi mahasiswa dibutuhkan sangat akurat, karena mempengaruhi nilai akhir yang mereka peroleh untuk mata kuliah yang mereka ambil.
- c. Penelitian yang dilakukan oleh (Hesta Arum Ekawaty, 2011) mengembangkan sistem *SMS Gateway* nilai siswa. Tentang hasil nilai, baik nilai ulangan harian maupun ulangan semester yang dilakukan menggunakan SMS kepada siswa yang bersangkutan.
- d. Penelitian yang dilakukan oleh (Arif Saeful Miftah, 2014) mengembangkan aplikasi *SMS Gateway* presensi siswa berbasis web dengan PHP dan MySQL. Aplikasi ini ditujukan kepada orang tua agar aktif tahu keadaan siswa dengan cara mengetikkan format SMS tertentu.

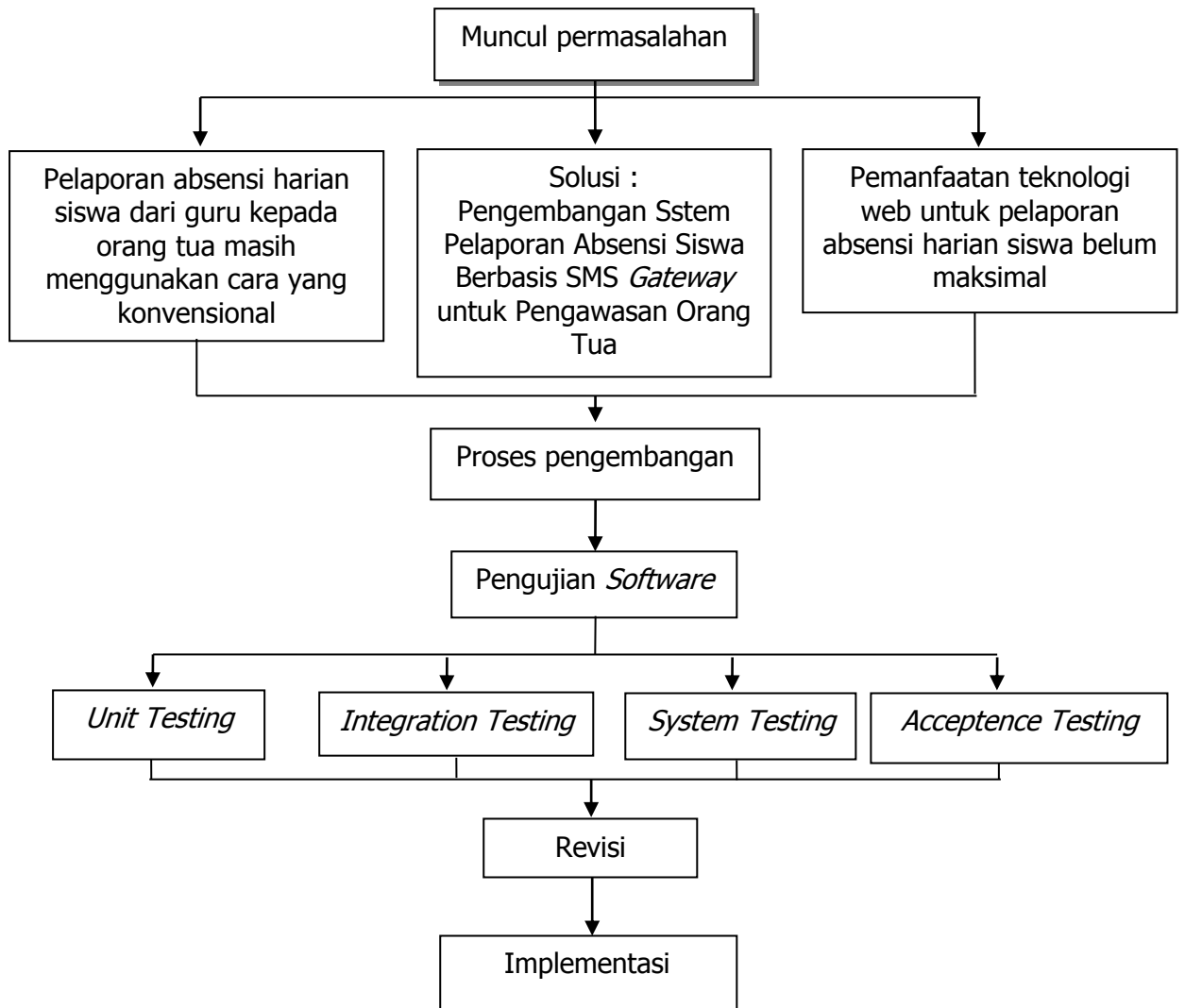
C. Kerangka Pikir

Kerangka pikir merupakan bentuk proses dari keseluruhan proses dalam penelitian, kerangka pikir mencakup variabel yang saling berhubungan. Kerangka pikir disusun berdasarkan teori yang telah dideskripsikan. Pada pengembangan aplikasi *SMS Gateway* ini menggunakan metode model-V. Dalam model-V ini terdiri dari *requirements, specification, design, implementation*. Pengujian kualitas menggunakan standar ISO 25010. Ada lima aspek kualitas yang diujikan pada penelitian ini, yaitu *functional suitability, performance efficiency, usability, maintainability, dan compatibility*. Pengujian kualitas menggunakan standar ISO 25010 karena aspek dalam ISO 25010 mewakili aspek internal dan eksternal dari perangkat

lunak yang diujikan, serta memiliki kelebihan dalam hal struktur hierarki, kriteria evaluasi, bentuk dan ekspresi yang komprehensif, definisi yang akurat dan sederhana.

Penelitian ini diawali dengan adanya permasalahan yang muncul sehingga diperlukan sebuah alternatif penyelesaian. Alternatif penyelesaian masalah yang dilakukan adalah dengan mengembangkan aplikasi *SMS Gateway* sebagai pengawasan orang tua. Setelah aplikasi selesai dirancang kemudian dilakukan pengujian dari perangkat lunak tersebut. Pengujian dilakukan dengan standar kualitas ISO 25010. Setelah dilakukan pengujian, maka dilanjutkan dengan revisi dan implementasi kepada pengguna sehingga menghasilkan aplikasi yang memiliki kualitas baik.

Bagan1. Kerangka Pikir



D. Pertanyaan Penelitian

Berdasarkan latar belakang dan tujuan penelitian maka pertanyaan penelitian dalam penelitian ini adalah sebagai berikut:

1. Bagaimana pengembangan aplikasi pelaporan absensi harian siswa berbasis *SMS Gateway* untuk pengawasan orang tua di SMK N 4 Klaten?
2. Bagaimana tingkat kualitas sistem pelaporan absensi harian siswa berbasis *SMS Gateway* untuk pengawasan orang tua di SMK N 4 Klaten berdasarkan standar pengujian kualitas perangkat lunak ISO/IEC 25010 pada aspek *functional suitability*, *performance efficiency*, *usability*, *maintainability*, dan *compatibility*?

BAB III METODE PENELITIAN

A. Model Pengembangan

Metode yang digunakan pada penelitian "Pengembangan Sistem Pelaporan Absensi Siswa Berbasis *SMS Gateway* untuk Pengawasan Orang Tua di SMK N 4 Klaten" ini menggunakan metode *Research and Development* (R&D). Menurut Sugiyono (2014:333) metode penelitian dan pengembangan adalah metode yang digunakan untuk menghasilkan produk tertentu, dan menguji keefektifan produk tersebut. Model pengembangan perangkat lunak merupakan strategi yang digunakan untuk mengembangkan perangkat lunak yang mencakup proses, *methods* dan *tools*, sehingga memungkinkan pengembangan selesai tepat waktu dan berkualitas. (Pressman, 2012).

Model pengembangan perangkat lunak pada penelitian ini adalah pengembangan sistem pelaporan absensi siswa berbasis *SMS Gateway* untuk pengawasan orang tua di SMK N 4 Klaten. Model pengembangan yang digunakan adalah model-V yang terdiri dari *requirements*, *specification*, *design*, dan *implementation*. Pada tahap pengujian juga sudah masuk dalam alur V-model yaitu pengujian *unit testing*, *integration testing*, *system testing*, dan *acceptance testing*.

B. Prosedur pengembangan

1. Requirements

Analisis yang dilakukan pada tahap ini adalah identifikasi permasalahan yang ada, analisis terhadap fungsionalitas perangkat lunak, desain antar muka

(*interface*) perangkat lunak, kebutuhan perangkat lunak dan perangkat keras untuk dapat mengembangkan dan menjalankan sistem absensi yang berbentuk *website*. Permasalahan yang diidentifikasi dalam penelitian ini adalah pelaporan ketidakhadiran di SMK N 4 Klaten yang dilakukan di akhir studi pada saat siswa menerima rapor. Orang tua mengalami kesulitan dalam mengingatkan siswa yang tidak berangkat selama satu semester baik dengan keterangan izin, sakit dan memungkinkan siswa membolos pada saat berlangsungnya kegiatan belajar mengajar jika laporan diberikan pada akhir semester. Proses guru dalam memberikan laporan absensi kepada orang tua masih menggunakan cara manual, yaitu memasukkan pelaporan dalam rapor berupa total keseluruhan absensi pada akhir semester. Pada kolom tersebut hanya terdapat kolom keterangan (sakit, izin dan alpha) dan masih belum terdapat keterangan waktu pada masing-masing jenis absensi. Keterangan absensi siswa seharusnya dilaporkan secara detail kepada orang tua sebagai pertanggungjawaban seorang guru terhadap murid. Orang tua (cenderung) percaya dengan keadaan siswa di sekolah tanpa memperdulikan keadaan sebenarnya apakah siswa benar-benar datang di sekolah atau mengikuti kegiatan dari sekolah dan bahkan membolos. Pada analisis kebutuhan ini diharapkan semua kebutuhan pada proses pengembangan perangkat lunak terpenuhi. Metode pengumpulan data yang dilakukan pada penelitian ini dengan menggunakan studi literatur, observasi dan kuisioner.

2. Specification

Setelah dilakukan *requirement* yang bertujuan untuk mengumpulkan informasi dan masalah, langkah selanjutnya adalah *specification*. Pada metode V-Model

pengumpulan data digunakan untuk menentukan spesifikasi dari aplikasi yang dikembangkan. Spesifikasi tahapan ini menjawab "apa" saja yang dibutuhkan untuk mengembangkan aplikasi. Tahapan ini menghasilkan "apa" saja fitur dan kebutuhan dari absensi *SMS Gateway* yang dikembangkan. Metode pengumpulan data menggunakan studi literatur terhadap kebutuhan *SMS Gateway* dan wawancara/observasi terhadap fitur yang dibutuhkan.

3. Desain Produk

Berdasarkan *requirement* dan *specification* yang telah dilakukan, langkah selanjutnya adalah desain sistem yang merupakan gambaran dari analisis kebutuhan. Desain sistem meliputi perancangan UML (*Unified Modeling Language*) untuk menggambarkan proses kerja dari sisi perangkat lunak yang terdiri dari empat diagram yaitu *Use Case Diagram*, *Class Diagram*, *Activity Diagram* dan *Sequence Diagram*. Perancangan *database* memberikan gambaran tentang data yang digunakan, serta perancangan antar muka (*user interface*) untuk memberikan gambaran tampilan dari sistem yang akan dikembangkan.

4. Implementasi

Setelah tahap desain selesai dijalankan langkah selanjutnya adalah tahap implementasi (koding). Pembuatan aplikasi dimulai dengan mengacu pada desain yang telah dibuat. Pembuatan aplikasi harus sesuai dengan model proses yang ada dalam rekayasa perangkat lunak.

5. Pengujian

Setelah aplikasi selesai dibuat maka selanjutnya dilakukan pengujian. Pada tahap ini aplikasi yang telah dikembangkan kemudian diberikan berbagai rangkaian

validasi kualitas perangkat lunak yang menggunakan instrumen penelitian sesuai standar ISO 25010 sehingga dapat dilakukan evaluasi sistem sebelum akhirnya dapat digunakan oleh banyak pengguna. Validasi atau pengujian dilakukan untuk meminimalisir kesalahan yang terjadi dan untuk menguji kualitas dari sistem yang dibuat.

Pengujian perangkat dilakukan melalui empat tahap, yaitu :

Tabel 8. Metode Pengujian

Teori Pressman	Tahap Pengujian (Teori Rosa A. S)	Teknik Pengujian	Aspek ISO 25010
Verifikasi	<i>Unit Testing</i>	<i>White Box</i>	<i>Functional Suitability</i> <i>Performance efficiency</i>
	<i>Integration Testing</i>	<i>Black Box</i>	
Validasi	<i>System Testing</i>	<i>Black Box</i>	<i>Maintainability</i> <i>Compatibility</i>
	<i>Acceptance Testing</i>	<i>Alpha & Beta Testing</i>	<i>Usability</i>

Pada *Unit Testing* dan *Integration Testing* merupakan tahap verifikasi pengujian perangkat lunak yang dilakukan dengan teknik *white box* dan *black box testing* untuk pengujian aspek *functional suitability* dan *performance efficiency*. Pada tahap validasi pengujian perangkat lunak yaitu *system testing* dan *acceptance testing* dilakukan dengan teknik *black box testing*, *alpha* dan *beta testing* untuk pengujian *maintainability*, *compatibility* dan *usability*.

6. Waktu dan tempat penelitian

Penelitian ini dilakukan pada Bulan Oktober 2016. Lokasi penelitian di SMK Negeri 4 Klaten untuk proses observasi, pengembangan aplikasi, validasi dan revisi.

7. Sumber data/subjek penelitian

Sumber data dalam penelitian merupakan subjek untuk memperoleh data yang dibutuhkan. Sumber data penelitian ini adalah:

- a. Guru-guru SMK N 4 Klaten untuk aspek *usability*.

Jumlah sampel guru adalah 22 orang, hal ini sesuai dengan pendapat yang dikemukakan oleh Nielsen (2010) yang menyatakan jumlah sampel minimal untuk pengujian *usability/acceptance* sejumlah 20 orang.

- b. Ahli pemrograman (*developer/programmer*) untuk aspek *functional suitability*.
- c. Dokumentasi terhadap aplikasi yang dikembangkan untuk aspek *maintainability, efficiency* dan *compatibility*.

8. Metode dan alat pengumpul data

Metode pengumpulan data dalam penelitian ini adalah sebagai berikut:

- a. Studi literatur

Studi literatur ini dilakukan untuk mengumpulkan temuan riset, jurnal, maupun informasi lain yang berkaitan dengan pengembangan produk. Dalam penelitian ini, studi literatur digunakan untuk mempelajari uji kualitas perangkat lunak *website* menggunakan ISO 25010. Selain itu studi literatur digunakan untuk mempelajari langkah dan metode rekayasa perangkat lunak. Mempelajari sistem absensi sekolah berbasis *SMS Gateway* juga dilakukan dengan studi literatur.

- b. Observasi

Pengumpulan data menggunakan teknik observasi dilakukan untuk mengumpulkan data berupa permasalahan dan kebutuhan perangkat absensi yang muncul dilapangan. Teknik observasi dilakukan dengan pengamatan

terhadap sistem absensi yang ada di sekolah dan penyampaian informasi terkait absensi guru kepada orang tua. Selain itu, teknik observasi juga digunakan untuk mengumpulkan data terkait dengan pengujian kualitas sistem pada aspek *performance efficiency, functional suitability, maintainability dan compatibility*.

c. Wawancara

Teknik wawancara ini digunakan secara bebas kepada guru-guru dan murid murid. Teknik wawancara yang dilakukan adalah teknik tidak terstruktur, yang dilakukan secara spontan dan tanpa membuat catatan sebelumnya. Teknik wawancara ini digunakan untuk mengetahui kebutuhan *user* mengenai sistem informasi yang akan dikembangkan.

d. Kuisisioner

Teknik kuisisioner pada pengumpulan data ini digunakan untuk mengumpulkan data terkait pengujian kualitas perangkat lunak pada aspek *usability*.

9. Instrumen Penelitian

Instrumen yang digunakan dalam penelitian ini terdiri dari instrumen untuk aspek *functional suitability, performance efficiency, maintainability, usability* dan *compatibility*. Berikut rincian instrumen yang digunakan:

a. Instrumen aspek *functional suitability*

Pada pengujian ini berfokus pada kesesuaian satu set fungsi untuk dapat melakukan tugas-tugas tertentu. Pengujian faktor kualitas *functionality* dilakukan dengan melakukan tes pada setiap fungsi perangkat lunak yang didokumentasikan dalam *test case*.

Tabel 9. Format standar *test case functional suitability* (Ghazy Assassa)

Test Case Name	Menunjukkan nama unik dari fungsi / case yang diuji
<i>Pre-Condition</i>	Syarat kondisi awal yang harus terpenuhi sebelum test dapat dijalankan
<i>Step</i>	Nomor langkah atau nomor aksi yang dijalankan
<i>Action</i>	Langkah-langkah yang dijalankan
<i>Expected Response</i>	Hasil yang seharusnya didapatkan yang menunjukkan tidak ada <i>error</i> / kesalahan pada aplikasi
<i>Pass/ Fail</i>	Hasil yang didapatkan saat pengujian
<i>Comment</i>	Komentar untuk pengujian

b. Instrumen *usability*

Pengujian *usability* menggunakan instrumen kuisioner yang diberikan kepada setiap guru di SMK N 4 Klaten. Instrumen yang digunakan untuk menguji aspek *usability* menggunakan kuisioner dari J.R. Lewis berikut ini:

Tabel 10. Instrumen *usability*

No	Pernyataan	SS	S	R	TS	STS
1	Secara keseluruhan, saya puas dengan kemudahan pemakaian aplikasi ini.					
2	Penggunaan aplikasi ini sangat sederhana.					
3	Saya dapat menyelesaikan pekerjaan dengan sempurna menggunakan aplikasi ini.					
4	Saya dapat menyelesaikan pekerjaan dengan cepat menggunakan aplikasi ini.					
5	Saya dapat menyelesaikan pekerjaan saya secara efisien menggunakan aplikasi ini.					
6	Saya merasa nyaman menggunakan aplikasi ini.					
7	Sangat mudah mempelajari penggunaan aplikasi ini.					
8	Saya yakin akan lebih produktif ketika menggunakan aplikasi ini.					
9	Pesan kesalahan yang diberikan aplikasi ini menjelaskan dengan jelas cara mengatasinya.					
10	Kapanpun saya membuat kesalahan, saya bisa memperbaikinya dengan cepat dan mudah.					
11	Informasi yang disediakan aplikasi ini cukup jelas.					
12	Sangat mudah mencari informasi yang dibutuhkan di aplikasi ini.					
13	Informasi yang disediakan aplikasi sangat mudah dipahami.					
14	Informasi yang disediakan efektif membantu menyelesaikan tugas.					
15	Tata letak informasi yang ditampilkan dilayar aplikasi sangat jelas.					

No	Pernyataan	SS	S	R	TS	STS
16	Tampilan/antarmuka aplikasi menyenangkan.					
17	Saya suka menggunakan tampilan aplikasi seperti ini.					
18	Aplikasi ini memiliki fungsi dan kemampuan (kapabilitas) sesuai harapan saya.					
19	Secara keseluruhan, saya puas dengan aplikasi ini.					

c. Instrumen *performance efficiency*

Pengujian aspek *performance efficiency* menggunakan aplikasi GTMetrik karena aplikasi tersebut menghasilkan dua pengujian berdasarkan aturan YSlow dan PageSpeed. Aturan YSlow dikembangkan oleh Yahoo Developer Network dan PageSpeed Insights dikembangkan oleh Google.

d. Instrumen *maintainability*

Pengujian dalam aspek *maintainability* berfokus pada kemampuan pengkat lunak untuk dimodifikasi baik secara koreksi, perbaikan maupun adaptasi. Pengujian ini dilakukan dengan cara operasional dengan menggunakan instrumen *maintainability* (Land, 2002). Berikut instrumen aspek *maintainability*.

Tabel 11. Instrumen *maintainability* (Land, 2002)

Aspek	Aspek yang dinilai	Hasil yang akan diperoleh
<i>instrumentation</i>	Terdapat peringatan pada sistem pengolahan data untuk mengidentifikasi kesalahan	Ketika <i>user</i> melakukan kesalahan dalam operasionalnya, sistem akan memberikan peringatan.
<i>consistency</i>	Penggunaan satu bentuk perancangan pada seluruh rancangan sistem.	Bentuk rancangan sistem pengolah data mempunyai satu bentuk yang sama.
<i>simplicity</i>	Kemudahan dalam pengolahan, perbaikan, dan pengembangan sistem.	Mudah untuk diolah, diperbaiki, dan dikembangkan. Hal ini dapat diidentifikasi melalui penulisan kode program.

e. Pengujian *compatibility*

Pengujian *compatibility* menggunakan berbagai *browser* pada *desktop* untuk menguji kualitas perangkat lunak berjalan baik.

10. Teknik Analisis Data

Jenis data dalam penelitian ini adalah data kuantitatif yang diperoleh dari skor pengujian menggunakan instrumen penelitian. Analisis dengan pengekatan ini sesuai dengan pengukuran yang digunakan pada angket yaitu skala Guttman dan Likert.

1. Pengujian aspek kualitas *functional suitability*

Pengujian aspek kualitas *functional suitability* dilakukan dengan melakukan tes pada setiap fungsi perangkat lunak yang didokumentasikan dalam *test case*. Skala yang digunakan dalam pengujian ini adalah skala Guttman yang dengan tegas mengenai berjalan atau tidaknya fungsi dari suatu aplikasi. Menurut Sugiyono (2014:111) Skala Guttman digunakan untuk mendapatkan jawaban yang tegas seperti "YA" dan "Tidak". Kriteria lolos jika semua fungsi berjalan sesuai yang diharapkan.

2. Pengujian aspek *usability*

Pada pengujian kualitas aspek *usability* menggunakan instrumen kuisisioner yang diberikan kepada setiap *user* dari aplikasi yaitu guru SMK N 4 Klaten. Menurut Nielsen (2012), untuk penelitian kuantitatif jumlah responden untuk menguji faktor *usability* minimal 20 orang. Skala yang digunakan dalam pengujian faktor *usability* adalah skala Likert. Dari hasil yang didapat menggunakan kuisisioner tersebut maka dilakukan perhitungan reliabilitas

terhadap instrumen. Analisa aspek *usability* menggunakan kuisisioner kemudian dihitung dengan konsistensi *Alpha Cronbach*:

$$r_{11} = \left[\frac{n}{(n-1)} \right] \left[1 - \frac{\sum \sigma^2 i}{\sigma^2 t} \right]$$

Dimana:

- r_{11} = reliabilitas instrumen
- n = banyaknya butir pertanyaan
- $\sum \sigma_b^2$ = jumlah varian butir/item
- $\sigma^2 t$ = varian total

Suatu instrumen penelitian dinyatakan reliabel dengan menggunakan teknik ini, bila koefisien reliabilitas (r_{11}) > 0,6 (Setyawan, 2013). Penghitungan *Alpha Cronbach* yang dihitung dengan interpretasi nilai reliabilitas *Aplha Cronbach* sebagai berikut (Gilem&Gilem, 2003):

Tabel 12. *Interprestasi Alpha Cronbach*

No	Nilai R	Interprestasi
1.	R > 0.9	<i>Excellent</i>
2.	0.9 > R > 0.8	<i>Good</i>
3.	0.8 > R > 0.7	<i>Acceptabel</i>
4.	0.7 > R > 0.6	<i>Questionable</i>
5.	0.6 > R > 0.5	<i>Poor</i>
6.	R < 5	<i>Unacceptabel</i>

3. Instrumen *performance efficiency*

Pengujian aspek *performance efficiency* menggunakan aplikasi GTMetrix karena aplikasi tersebut menghasilkan dua pengujian berdasarkan aturan

YSlow dan PageSpeed. Aturan YSlow dikembangkan oleh Yahoo Developer Network, sementara PageSpeed Insight dikembangkan oleh Google.

4. Instrumen *maintainability*

Analisis *maintainability* dilakukan dengan menganalisis hasil pengujian secara operasional dengan instrumen *maintainability*. Apabila web yang diuji lolos untuk semua aspek yang terdapat dalam instrumen, maka web dinyatakan memenuhi uji *maintainability* (Land, 2002).

5. Instrumen *compatibility*

Pengujian *compatibility* menggunakan beberapa *browser* pada *desktop* untuk menguji kualitas perangkat lunak berjalan baik. *Web-based applications* memenuhi aspek *compatibility* jika dapat berjalan baik pada berbagai *browser* (Schach, 2008.248).

BAB IV

HASIL PENELITIAN DAN PEMBAHASAN

A. Deskripsi Data Uji Coba

Pada tahap pengembangan sistem pelaporan absensi siswa berbasis *SMS Gateway* untuk pengawasan orang tua di SMK N 4 Klaten ini menggunakan model-V yang terdiri dari tahap pengembangan (*requirements, specification, design, dan implementation*) dan tahap pengujian (*unit testing, integration testing, system testing, dan acceptance testing*). Tahapan dalam pengembangan dijelaskan sebagai berikut.

1. Requirements

Permasalahan yang diidentifikasi dalam penelitian ini adalah pelaporan ketidakhadiran di SMK N 4 Klaten yang dilakukan diakhir studi pada saat siswa menerima rapor. Orang tua mengalami kesulitan dalam mengingat siswa yang tidak berangkat selama satu semester baik dengan keterangan izin, sakit dan memungkinkan siswa membolos pada saat berlangsungnya kegiatan belajar mengajar jika laporan diberikan pada akhir semester. Proses guru dalam memberikan laporan absensi kepada orang tua masih menggunakan cara manual, yaitu memasukkan pelaporan absensi dalam rapor berupa jumlah keseluruhan absensi selama satu semester. Pada rapor yang diberikan guru kepada orang tua hanya terdapat keterangan sakit, izin dan alpha. Pada rapor tersebut belum terdapat keterangan membolos pada jam tertentu. Orang tua cenderung percaya dengan keadaan siswa disekolah tanpa memperdulikan

keadaan sebenarnya apakah siswa benar-benar datang disekolah atau mengikuti kegiatan dari sekolah dan bahkan membolos.

2. Specification

Pada tahap *specification* memberikan gambaran “apa” saja yang dibutuhkan untuk mengembangkan aplikasi. Produk yang dikembangkan membutuhkan spesifikasi perangkat *dekstop* dengan sistem operasi minimum *Windows 7*. Spesifikasi *hardware* dan *software* yang dibutuhkan:

a. Kebutuhan *hardware*

- 1) Laptop dengan sistem operasi *windows*
- 2) Modem GSM+SIM *card*
- 3) *Handphone*

b. Kebutuhan *software*

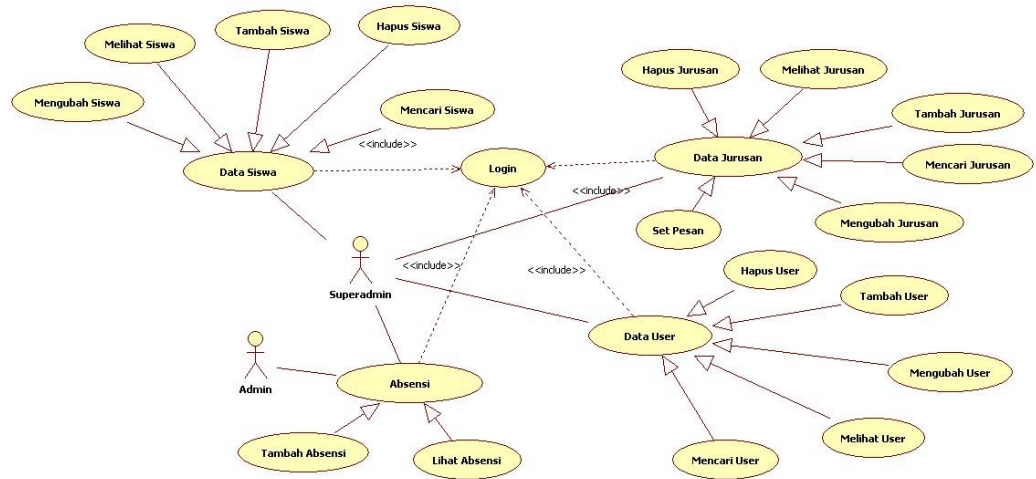
- 1) Gammu-1.33.0
- 2) Sublime
- 3) Xampp
- 4) Star UML

3. Design

Desain dilakukan untuk mempermudah pengembangan dalam mengembangkan produk. Pada tahap desain ini yang dilakukan adalah membuat desain *experience* (UX) dan *interface* (UI). Pada pembuatan desain *experience* dilakukan dengan UML sebagai bahasa permodelannya. Pembuatan UML diagram menggunakan bantuan *software* Star UML (*use case* dan *activity diagram*) dan menggunakan websequencediagrams.com (*sequence diagram*).

a. Use case diagram

User case diagram secara keseluruhan dapat digambarkan sebagai berikut:



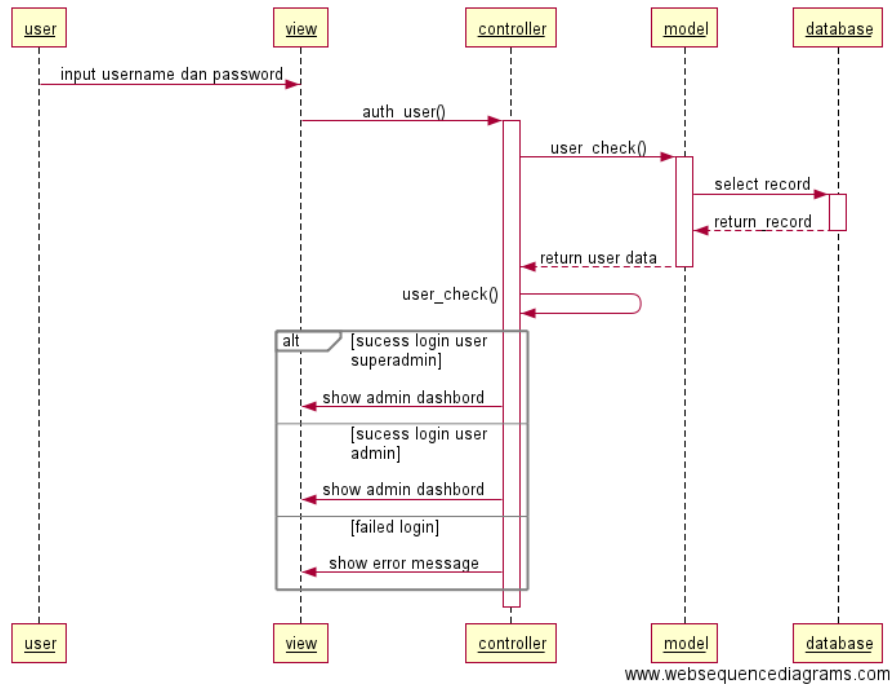
Gambar 4 use case diagram

Use case diagram sistem menggambarkan semua aktor dengan skenario masing-masing. Berdasarkan use case tersebut menggambarkan bahwa aplikasi SMS Gateway ini memiliki dua aktor yaitu administrator dan superadmin. Administrator dapat melakukan absensi yang berupa melihat dan menambah absensi. Sementara superadmin dapat melakukan manajemen data siswa, user, dan jurusan (input, edit, cari, lihat dan delete). Selain itu superadmin dapat melakukan set pesan yang akan dikirimkan. Semua fungsi atau use case administrator dan superadmin tersebut harus melalui use case login.

b. Sequence diagram

Berdasarkan hasil analisis skenario use case maka dapat dibuat diagram sequence. Berikut beberapa desain sequence diagram:

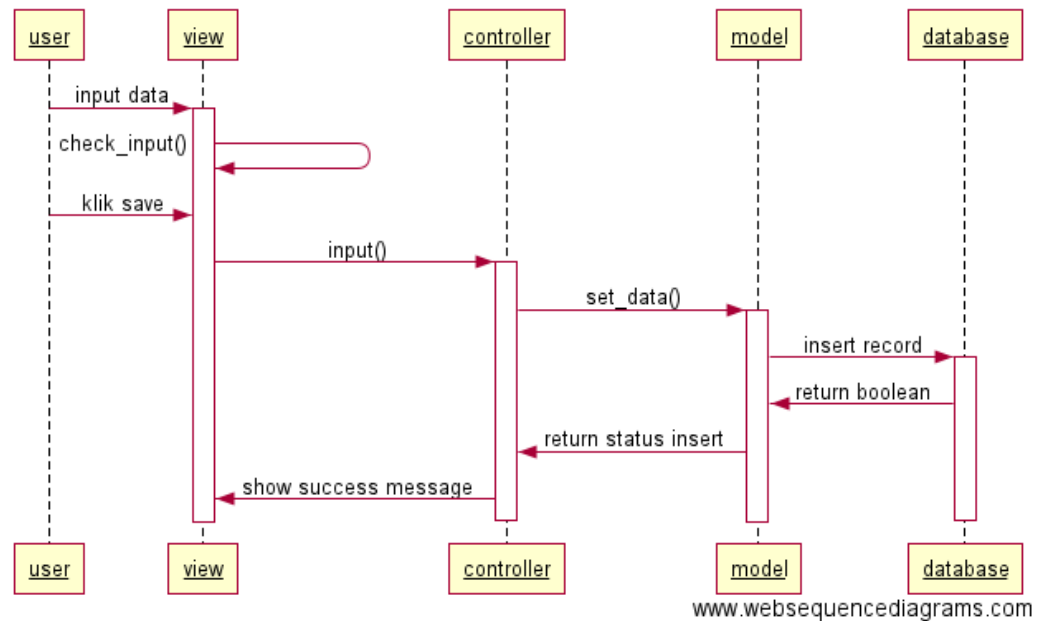
1) Login



Gambar 5 *Sequence diagram login*

Sequence diagram login menggambarkan bagaimana cara *login* bekerja, yaitu ketika pengguna mengisi *username* dan *password* kemudian mengklik *sign in* maka sistem akan mengecek (*username* dan *password*) pada *database*. Setelah dicek di *database* sistem akan menampilkan halaman sesuai dengan level *user*.

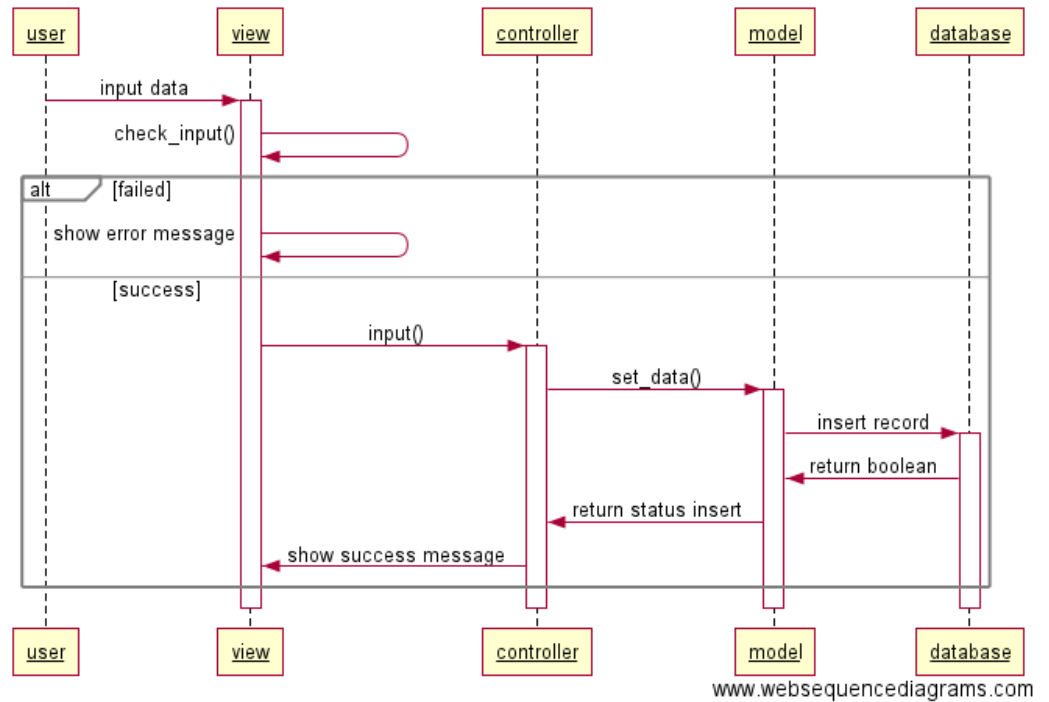
2) Presensi



Gambar 6 *Sequence diagram* presensi

Pada *sequence diagram* tersebut menggambarkan bagaimana fungsi tambah presensi bekerja, yaitu saat pengguna mengisi data sistem akan melakukan pengecekan. Setelah itu mengklik save dan data akan disimpan dalam *database*, setelah selesai dalam menyimpan data akan memunculkan peringatan pesan sukses.

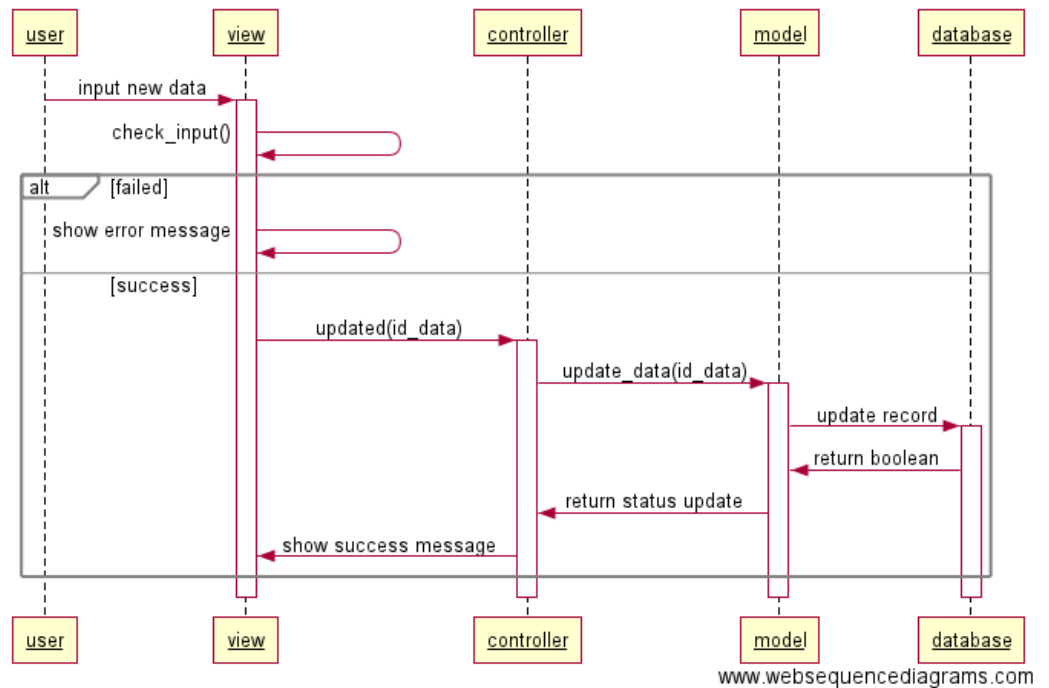
3) *Input data*



Gambar 7 *Sequence diagram input data*

Sequence diagram tersebut menggambarkan bagaimana fungsi *input data* bekerja, yaitu ketika *user (superadmin)* memasukkan data sistem akan melakukan pengecekan. Jika data yang dimasukkan salah sistem akan memunculkan pesan peringatan gagal. Sedangkan jika data yang dimasukkan benar data akan dimasukkan dalam *database* dan sistem akan memunculkan pesan sukses.

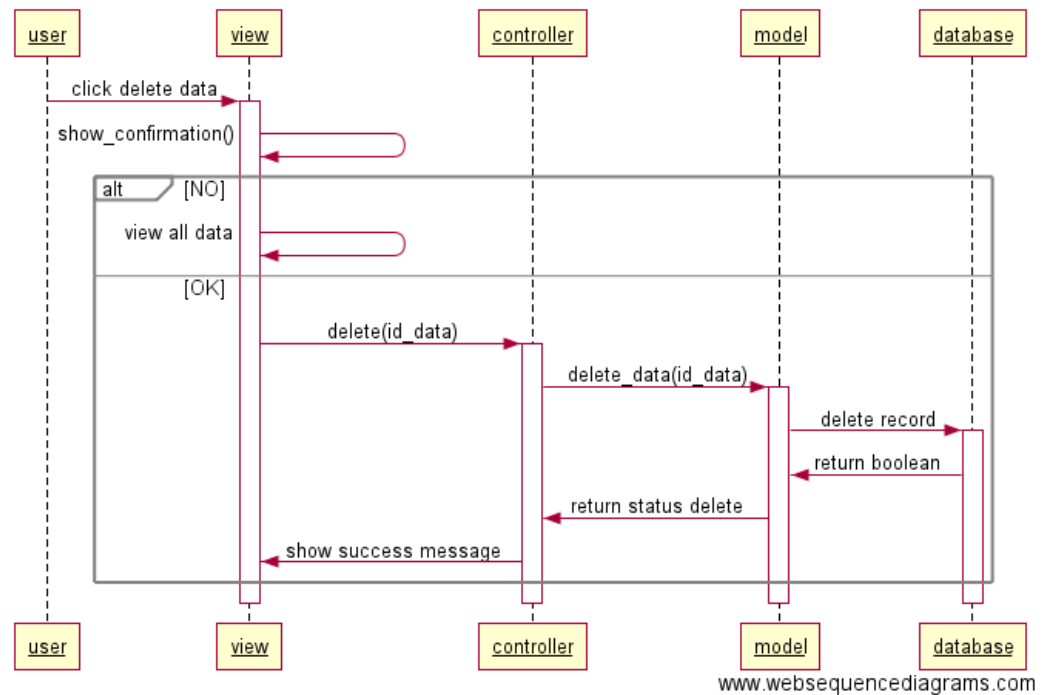
4) *Edit data*



Gambar 8 *Sequence diagram edit data*

Pada *sequence diagram* tersebut menggambarkan bagaimana fungsi *edit data* bekerja, yaitu ketika *user* memasukkan data yang baru sistem akan melakukan pengecekan. Jika data yang dimasukkan salah sistem akan memunculkan pesan peringatan gagal, sedangkan jika data baru yang dimasukkan benar sistem akan menyimpan ke dalam *database* dan sistem akan memunculkan pesan pemberitahuan sukses.

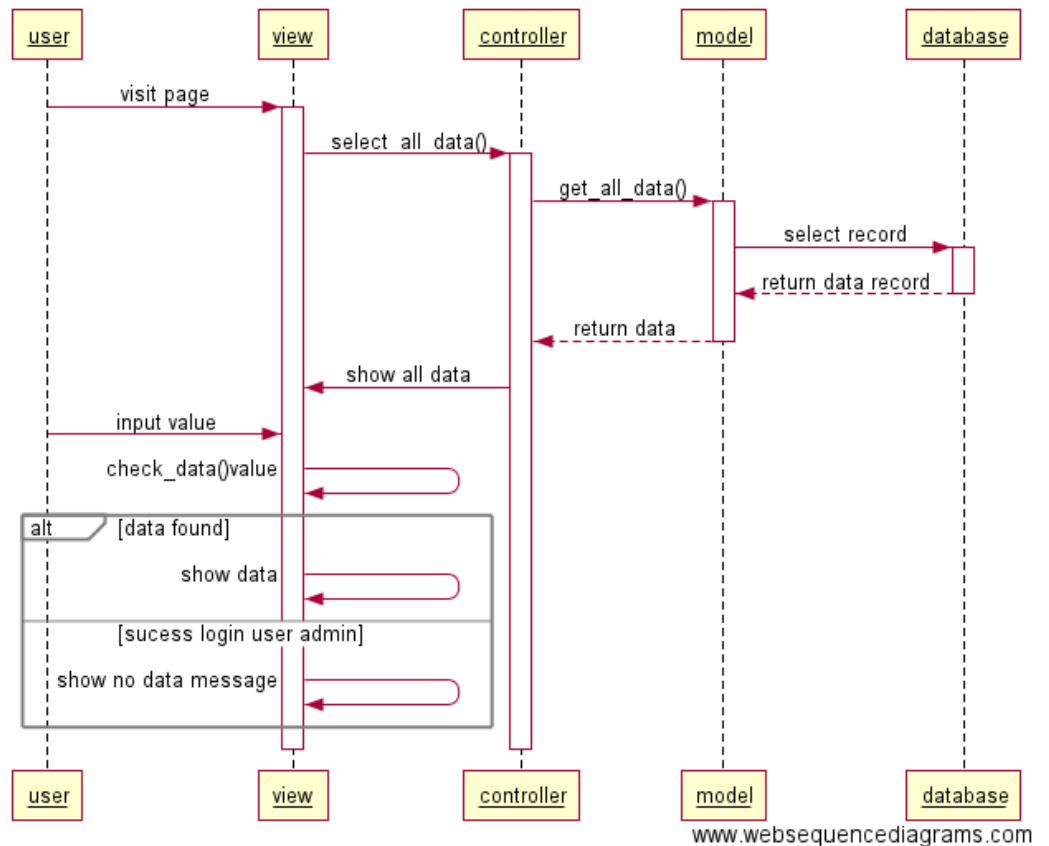
5) Delete data



Gambar 9 *Sequence diagram delete data*

Pada *sequence diagram* tersebut menggambarkan bagaimana *delete data* bekerja, yaitu ketika *user* mengklik delete sistem akan menampilkan pesan konfirmasi. Jika *user* memilih "NO" data akan ditampilkan, sedangkan jika *user* memilih "YES" sistem akan menghapus data pada *database* dan sistem akan memunculkan pesan pemberitahuan data sukses dihapus.

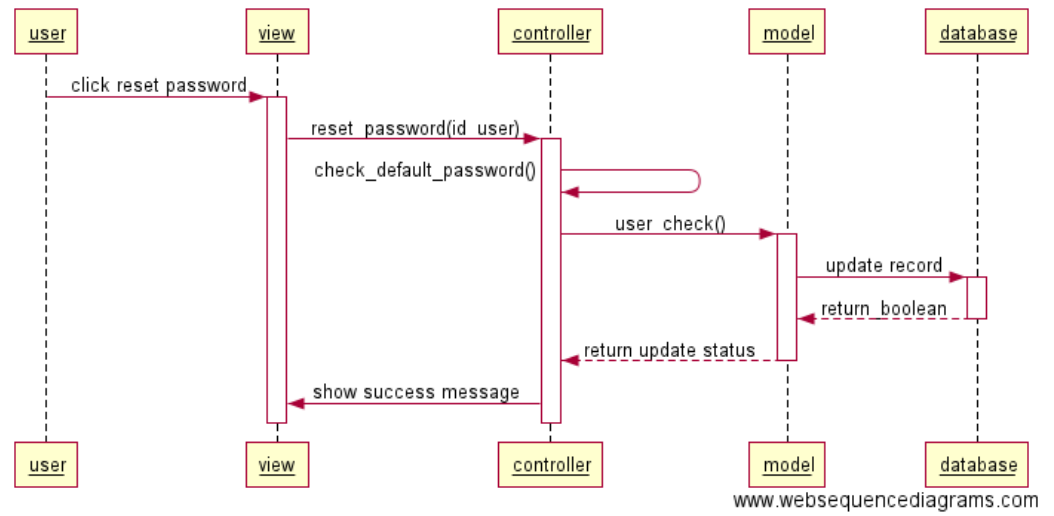
6) Cari data



Gambar 10 *Sequence diagram* cari data

Sequence diagram tersebut menggambarkan bagaimana fungsi cari data, yaitu ketika kita memasukkan data kita cari sistem mencari data dalam *database*. Jika data tidak ditemukan maka sistem menampilkan pesan data tidak ditemukan, sedangkan jika data yang dicari ada dalam *database* sistem menampilkan data yang dicari.

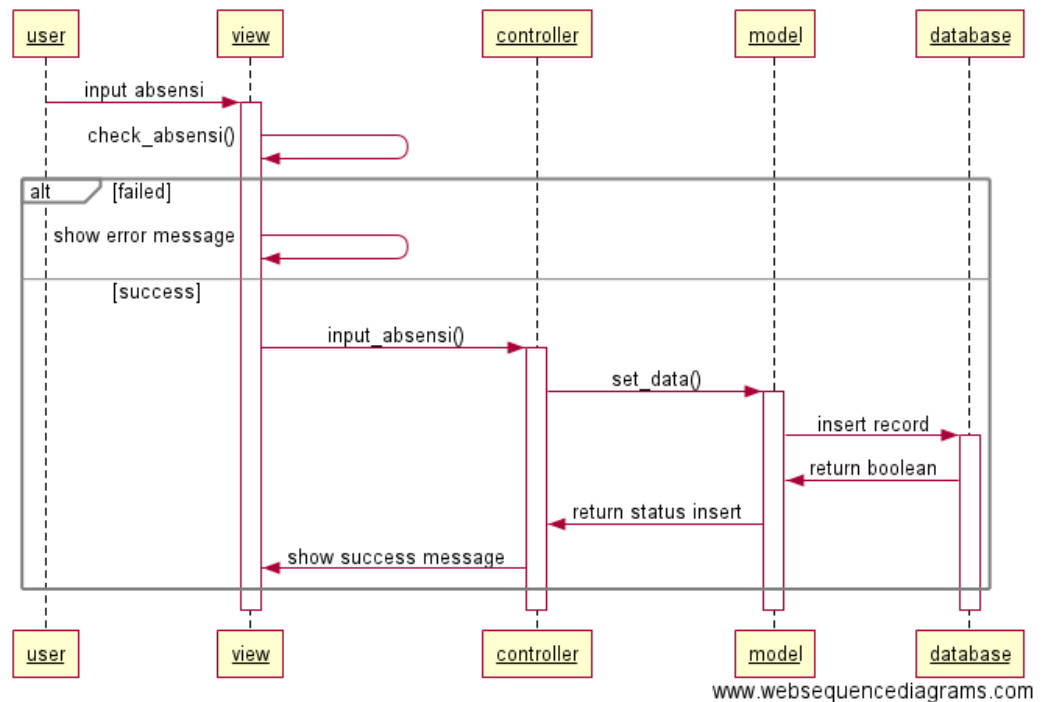
7) Reset password



Gambar 11. Sequence diagram reset password

Sequence diagram tersebut menggambarkan bagaimana fungsi *reset password* berjalan, yaitu ketika *user* mengklik *reset password* sistem akan mengecek *default password* dan sistem akan memunculkan pesan sukses.

8) Absensi



Gambar 12 *Sequence diagram* absensi

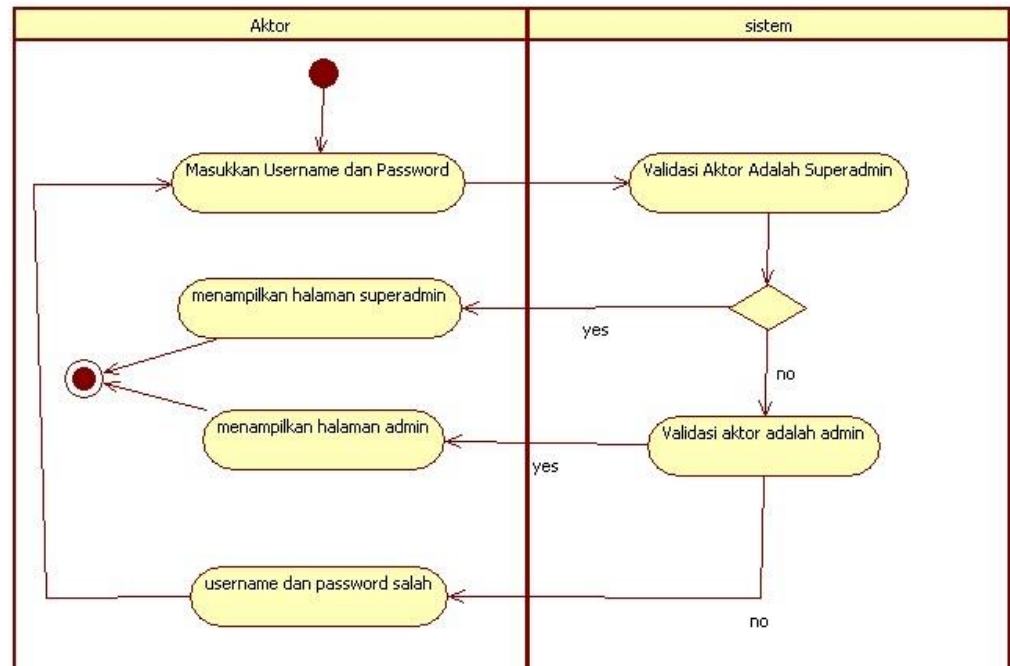
Sequence diagram tersebut menggambarkan bagaimana fungsi *input absensi* bekerja, yaitu ketika *user (superadmin/admin)* memasukkan data sistem akan melakukan pengecekan. Jika data yang dimasukkan salah sistem akan memunculkan pesan peringatan gagal. Sedangkan jika data yang dimasukkan benar data akan dimasukkan dalam *database* dan sistem akan memunculkan pesan sukses.

c. *Activity diagram*

Activity diagram menjelaskan tentang tingkah laku dinamis dari sistem.

Berikut beberapa desain *activity diagram*:

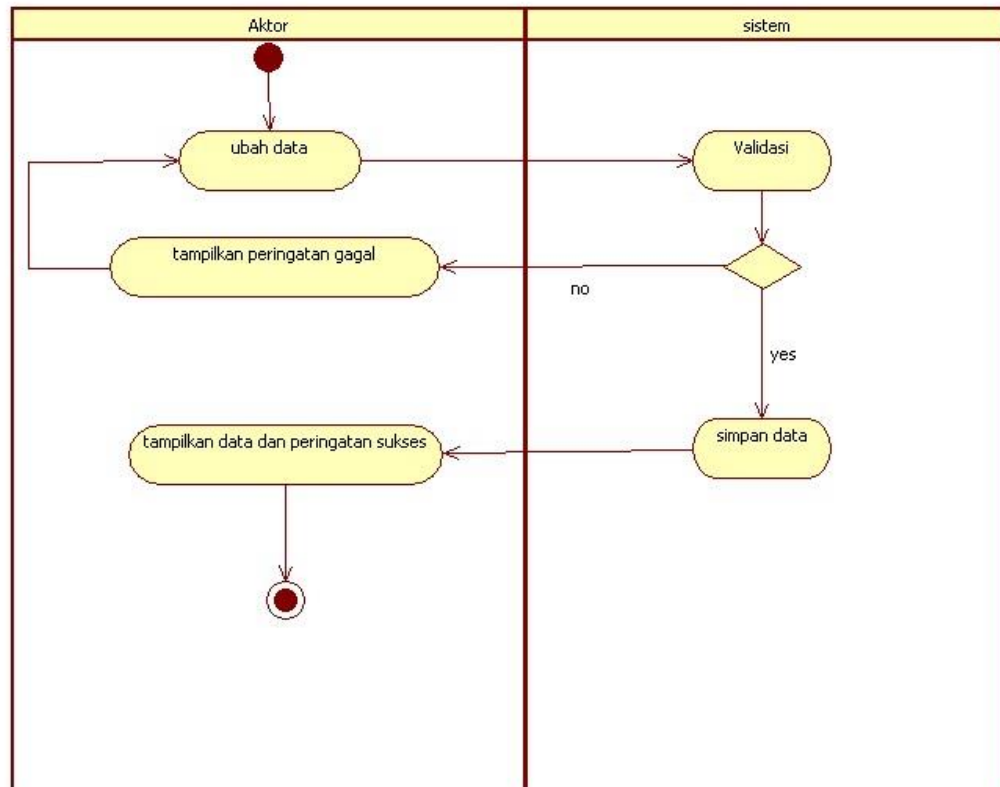
1) Login



Gambar 13 Activity diagram login

Activity diagram login menggambarkan alur login dari user superadmin dan administrator. Aktor harus memasukkan username dan password terlebih dahulu kemudian sistem melakukan validasi dan menentukan level user tersebut. Jika login berhasil maka aktor dapat masuk ke dashboard sesuai level user, namun jika gagal sistem akan memunculkan pesan kesalahan kemudian aktor diminta kembali memasukkan username dan password.

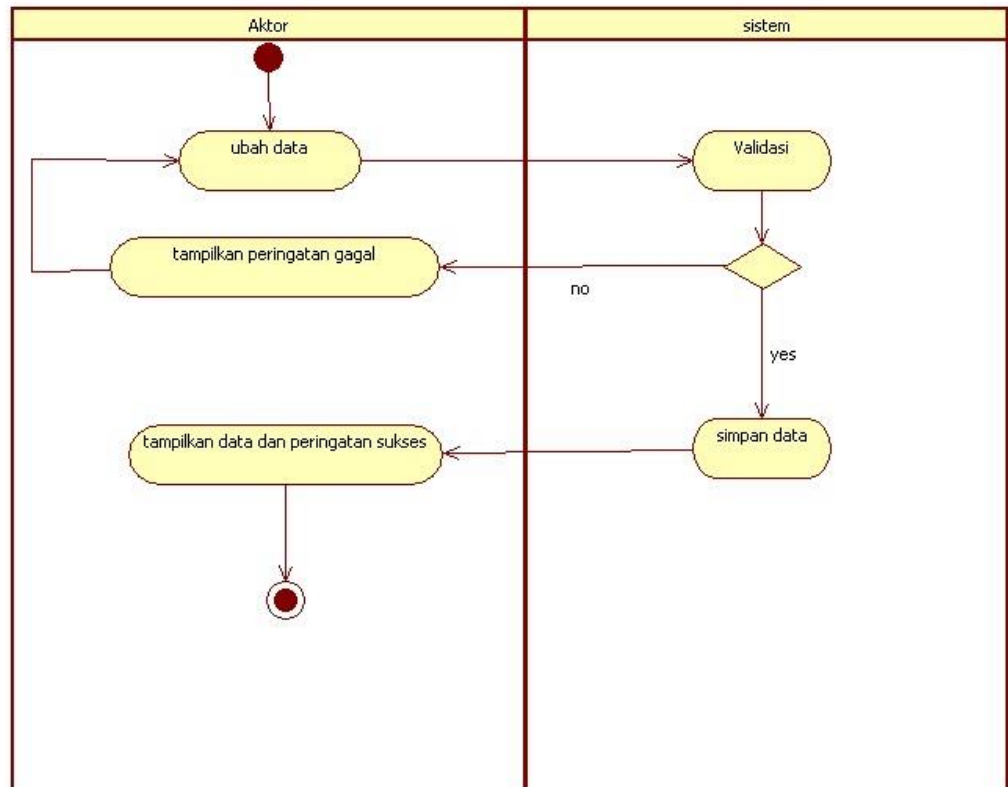
2) Input



Gambar 14 Activity diagram input

Activity diagram input menggambarkan aktivitas aktor yang memiliki hak untuk memasukkan data ke dalam sistem. Berdasarkan diagram tersebut, aktor memasukkan data kemudian sistem melakukan validasi. Jika validasi berhasil maka data akan disimpan ke dalam sistem kemudian akan dimunculkan pesan sukses dan data dimasukkan dalam *database*, tetapi jika validasi gagal akan dimunculkan pesan kegagalan dan kesalahan kemudian aktor diminta memasukkan data kembali.

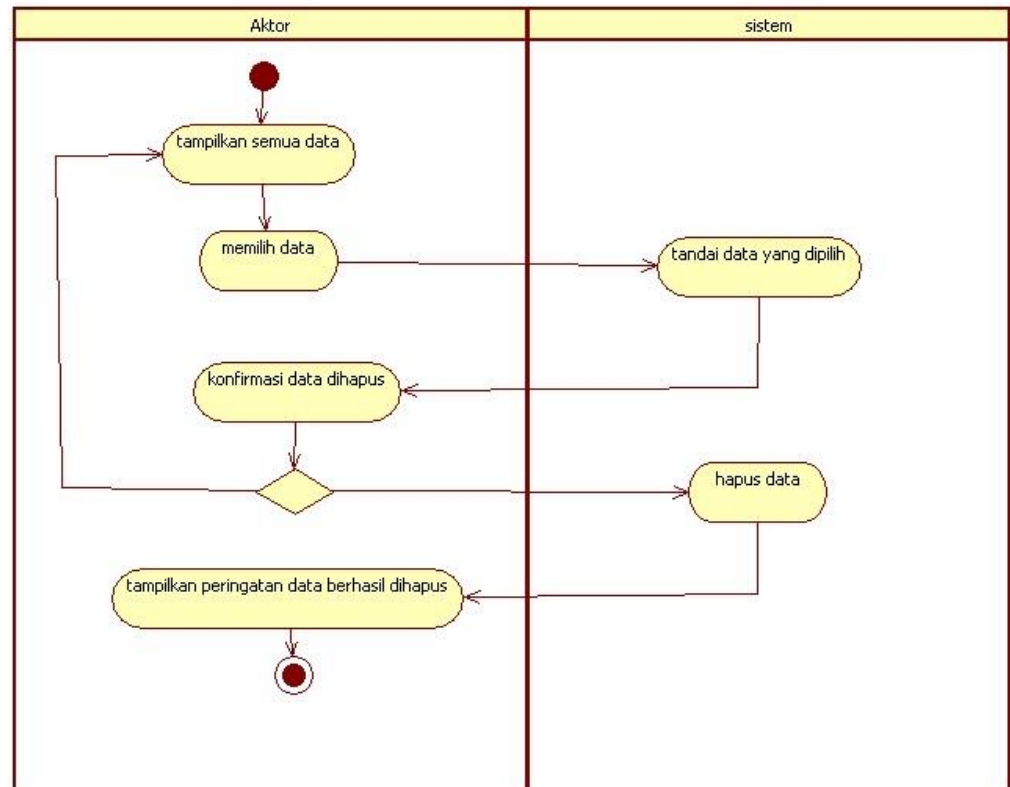
3) Edit



Gambar 15 Activity diagram edit data

Dalam melakukan *edit* data, aktor harus mengubah data terlebih dahulu kemudian sistem akan melakukan validasi terhadap data baru dimasukkan. Jika validasi berhasil data akan di-*update* dengan data yang baru kemudian dimunculkan pesan sukses dan data yang baru di-*edit*, sedangkan jika validasi gagal maka akan dimunculkan pesan kegagalan kesalahan kemudian aktor akan diminta memasukkan data kembali.

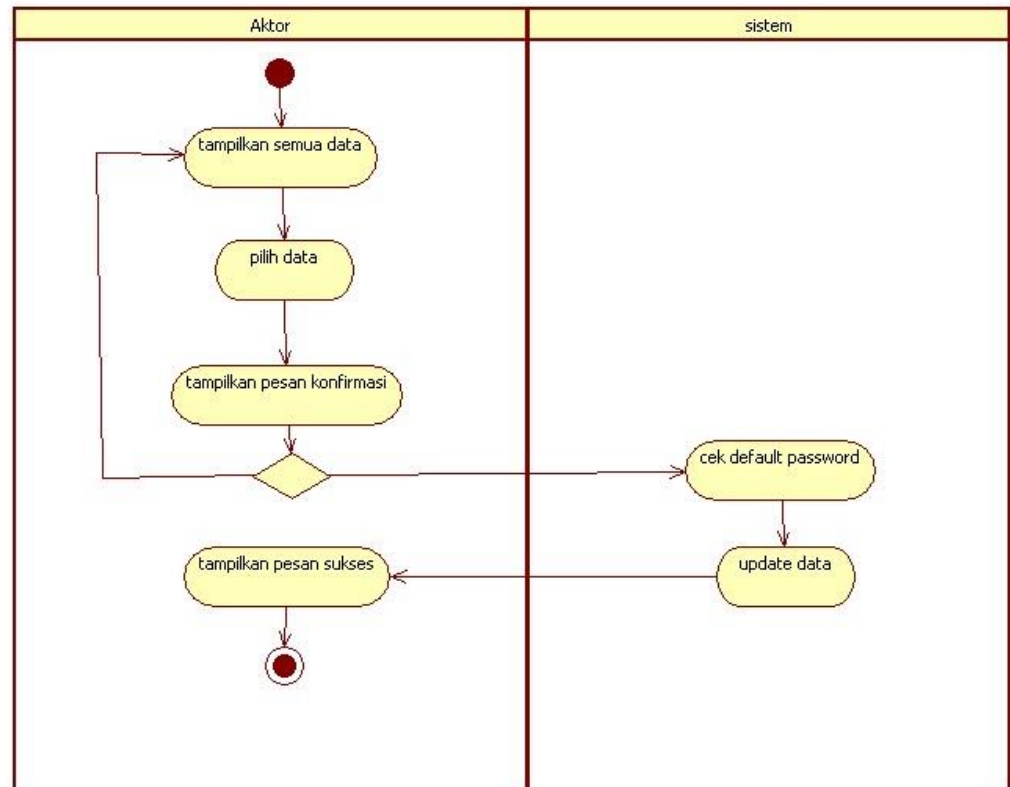
4) Delete



Gambar 16 Activity diagram delete data

Activity diagram delete dilakukan oleh aktor yang memiliki hak akses untuk menghapus data. Berdasarkan diagram tersebut aktor terlebih dahulu melihat keseluruhan data dan memilih data yang akan dihapus kemudian sistem akan menandai data tersebut serta memunculkan pesan konfirmasi apakah yakin akan menghapus data tersebut. Apabila aktor membatalkan konfirmasi tersebut maka aktor dapat melihat kembali data yang akan dihapus. Sedangkan apabila aktor mengkonfirmasi penghapusan data, maka user akan menghapus data tersebut kemudian memunculkan pesan sukses setelah eksekusinya.

5) Reset password

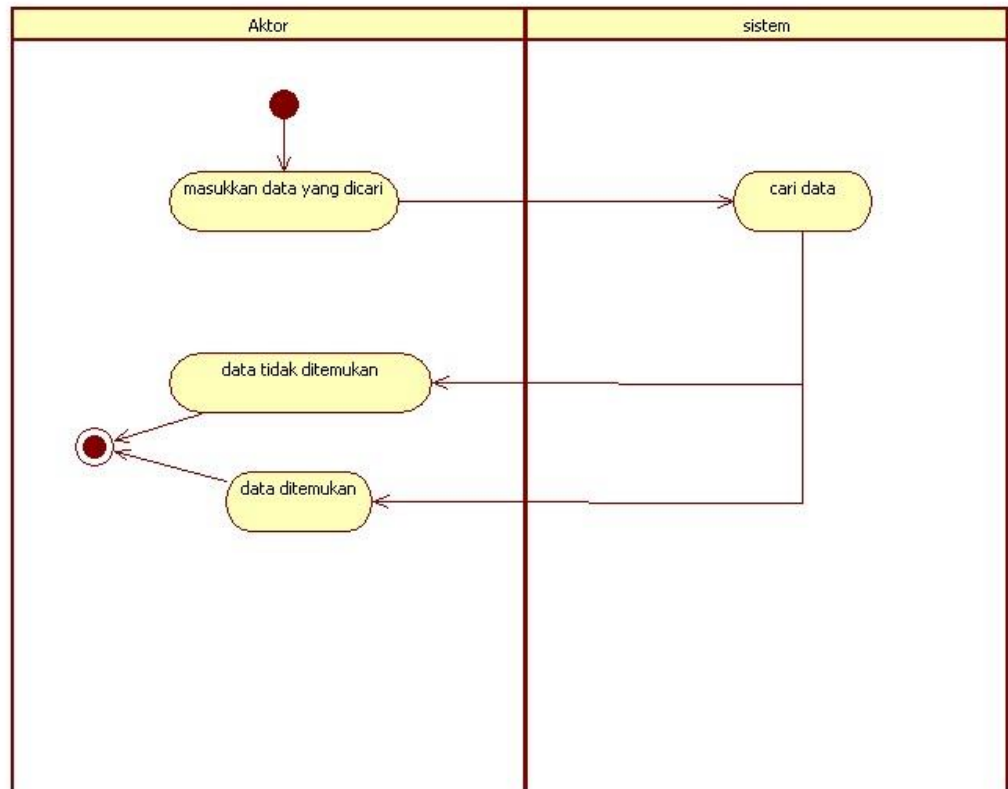


Gambar 17 Activity diagram reset password

Activity Diagram Reset Password hanya dapat dilakukan oleh aktor dengan *user level superadmin*. Pertama - tama aktor akan melihat seluruh data dari *user* sistem kemudian memilih data *user* yang akan di-*reset password*-nya. Setelah memilih, aktor akan langsung dihadapkan dengan konfirmasi apakah yakin akan me-*reset password* *user* tersebut. Apabila dibatalkan maka aktor dapat kembali melihat dan memilih data *user* kembali, sedangkan jika dikonfirmasi maka sistem akan mencari dan mengecek terlebih dahulu *default password* yang akan diberikan untuk *user*. Jika *default password* sudah dicek, sistem akan mengubah

data *password user* menjadi *default password* kemudian menampilkan pesan sukses telah mengubah data *password* kepada aktor.

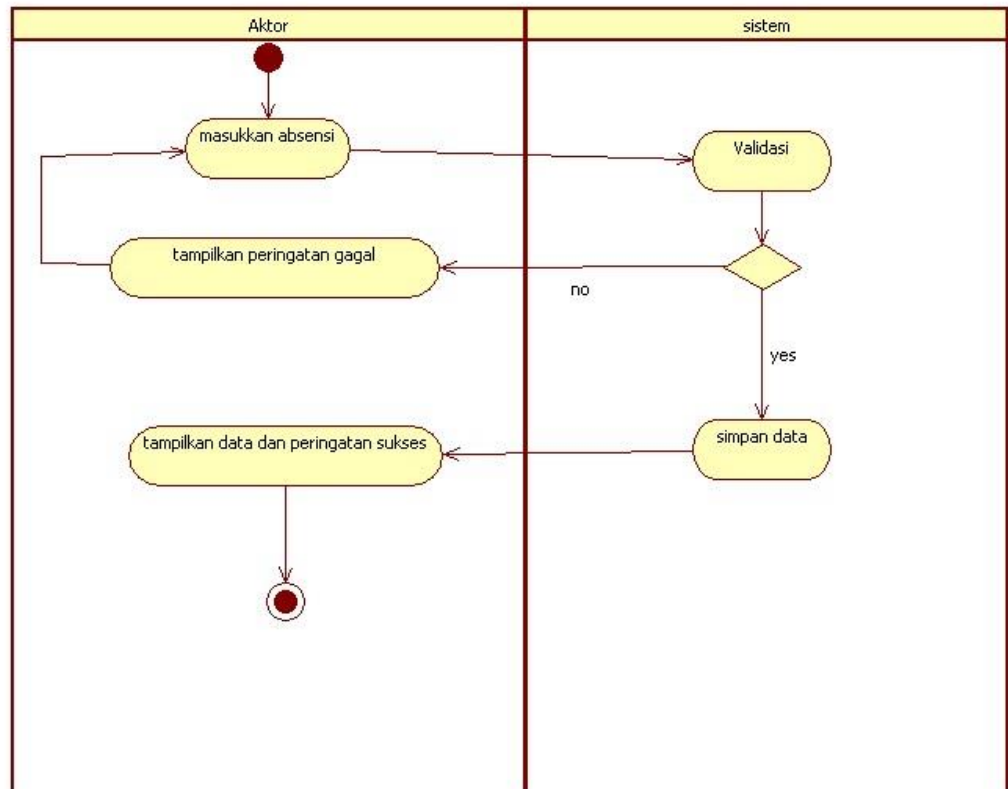
6) Cari



Gambar 18 *Activity diagram* cari

Activity diagram cari menggambarkan bahwa untuk mencari data aktor harus memasukkan *value* tersebut. Apabila tidak ditemukan maka dimunculkan pesan tidak terdapat data, sedangkan jika ditemukan data akan ditampilkan.

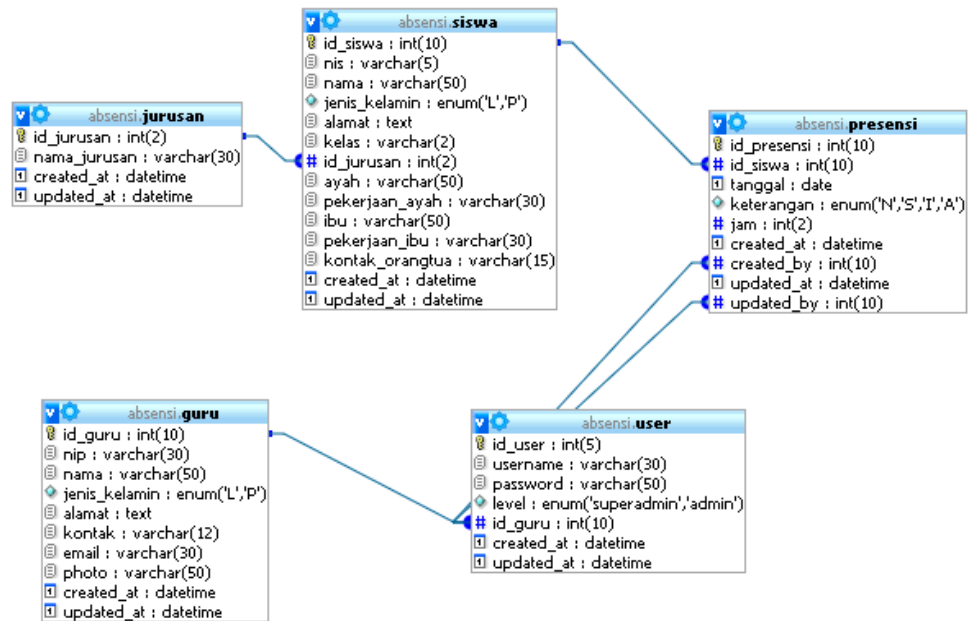
7) Absensi



Gambar 19 *Activity Diagram* Absensi

Activity diagram absensi menggambarkan aktivitas aktor yang memiliki hak untuk memasukkan absensi ke dalam sistem. Berdasarkan diagram tersebut, aktor memasukkan data absensi kemudian sistem melakukan validasi. Jika validasi berhasil maka data akan disimpan ke dalam sistem kemudian akan dimunculkan pesan sukses dan data dimasukkan dalam *database*, tetapi jika validasi gagal akan dimunculkan pesan kegagalan dan kesalahan kemudian aktor diminta memasukkan data absensi kembali.

4. Database

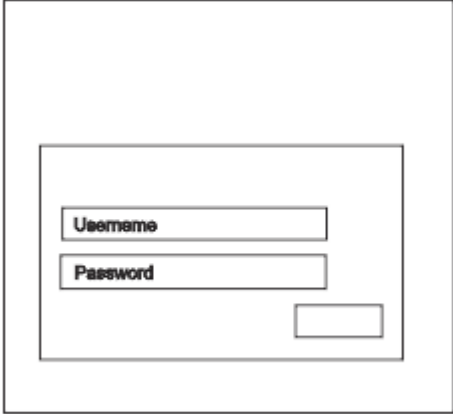
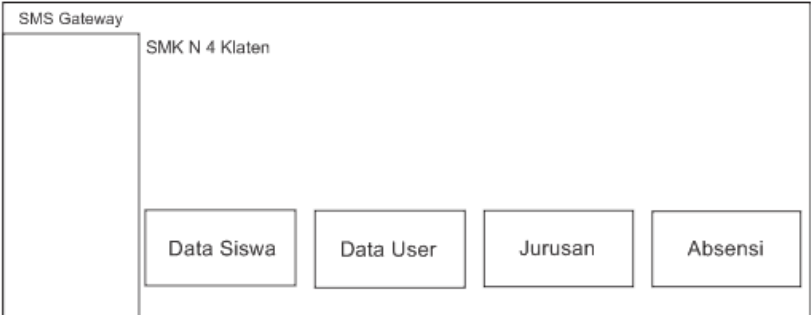
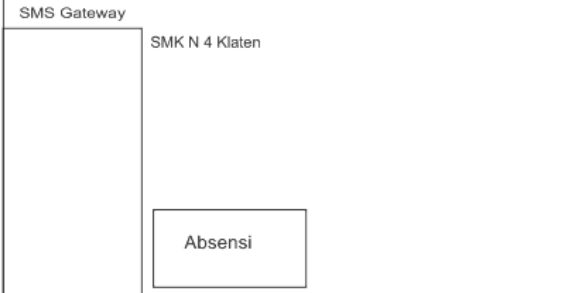



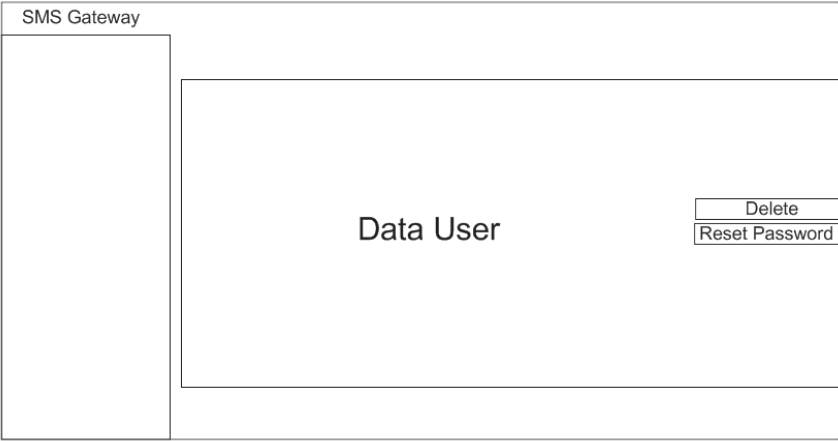
Gambar 20 ERD Absensi

5. Desain Interface

Desain *interface* pada pengembangan ini dibuat dengan *storyboard*, berikut *storyboard* desain *interface*:

Tabel 13 Desain *interface*

No	Desain	Deskripsi
1		<p>Pada halaman <i>login</i> terdapat menu untuk memasukkan <i>username</i> dan <i>password</i>, serta terdapat tombol untuk <i>sign in</i>.</p>
2		<p>Pada halaman <i>dashboard (superadmin)</i> terdapat empat menu utama yaitu data siswa, data user, jurusan dan absensi.</p>
3		<p>Pada halaman <i>dashboard (administrator)</i> hanya terdapat menu untuk melakukan absensi.</p>
4		<p>Pada halaman tambah data siswa terdapat <i>form</i> pengisian untuk mengisi data dan juga tombol <i>save</i>.</p>

No	Desain	Deskripsi
5		<p>Pada halaman data <i>user</i> terdapat data <i>user</i> (<i>administrator</i> maupun <i>superadmin</i>), tombol <i>delete</i>, dan <i>reset password</i>.</p>

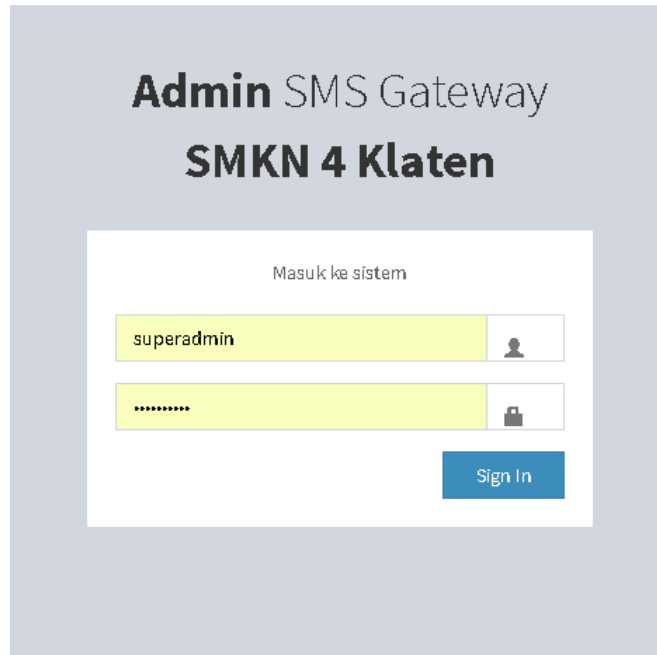
6. Implementasi

Setelah melakukan analisis kebutuhan dan membuat rancangan sistem dari aspek rekayasa perangkat lunak dengan menggunakan *Unified Modelling Language* (UML), desain basis data, dan desain antar muka (*interface*), maka selanjutnya dilakukan implementasi sistem termasuk di dalamnya proses pengkodean. Berikut hasil dari proses implementasi sistem:

- a. Implementasi fungsi dan rancangan antarmuka untuk aplikasi pelaporan SMS *Gateway* adalah sebagai berikut:

- 1) Halaman *login*

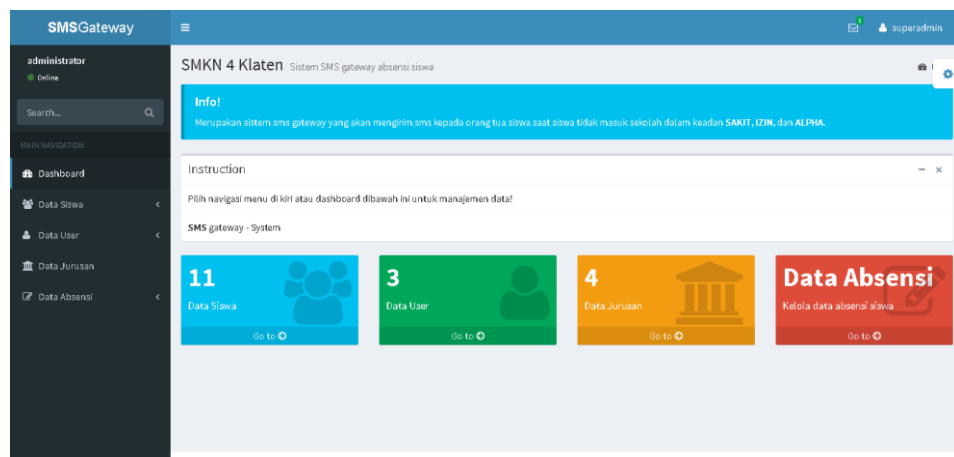
Implementasi antar muka halaman *login* adalah sebagai berikut:



Gambar 21 Halaman *login*

2) Halaman utama *superadmin*

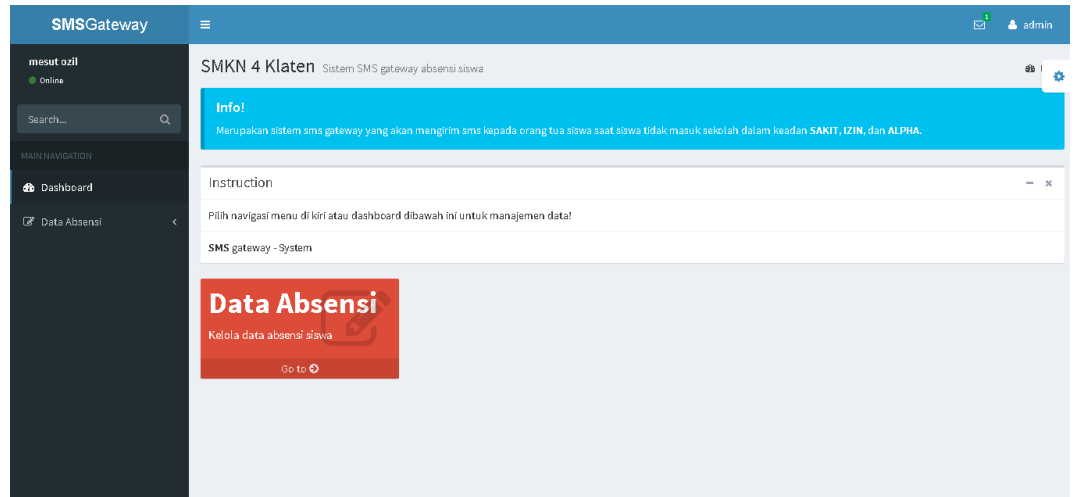
Implementasi antar muka halaman utama *superadmin* adalah sebagai berikut:



Gambar 22 Halaman *superadmin*

3) Halaman utama *admin*

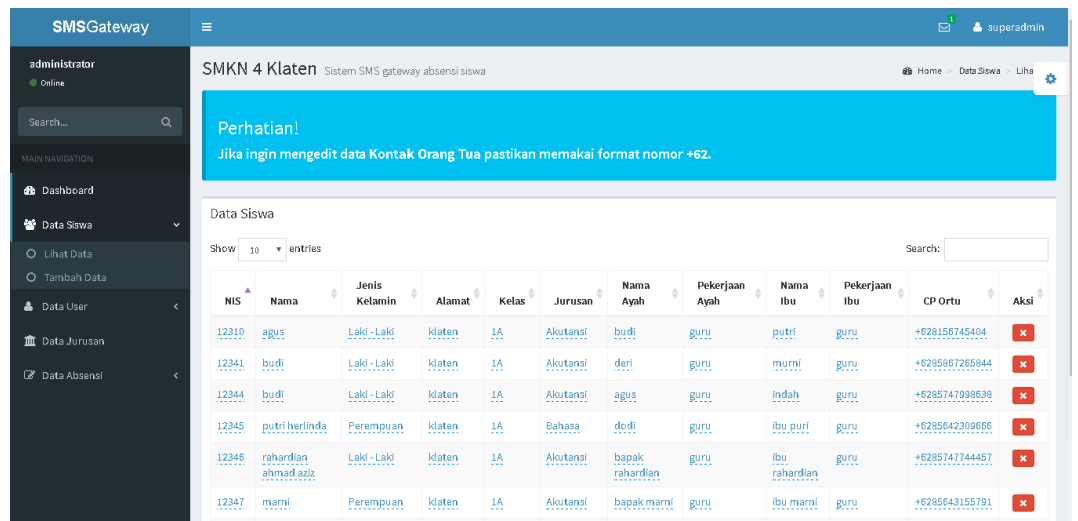
Implementasi antar muka halaman utama *admin* adalah sebagai berikut:



Gambar 23 Halaman *admin*

4) Halaman lihat data siswa

Implementasi antar muka halaman lihat data siswa adalah sebagai berikut:



Gambar 24 Halaman data siswa

7) Halaman input data *user*

Implementasi antar muka halaman input data *user* adalah sebagai berikut:

The screenshot shows the 'Tambah Data User' form in the SMSGateway application. The form includes fields for NIP, Nama, Jenis Kelamin (radio buttons for Laki and Perempuan), Alamat, Email (pre-filled with 'superadmin'), Password User (masked with dots), Level User (dropdown menu), and Kontak (pre-filled with '+62'). There are 'Save' and 'Cancel' buttons at the bottom right. An info banner at the top states: 'Merupakan halaman untuk mengelola data user yang dapat mengakses sistem. User dibagi menjadi user level SUPERADMIN dan user level ADMIN.'

Gambar 27 Halaman *input* data user

8) Halaman jurusan

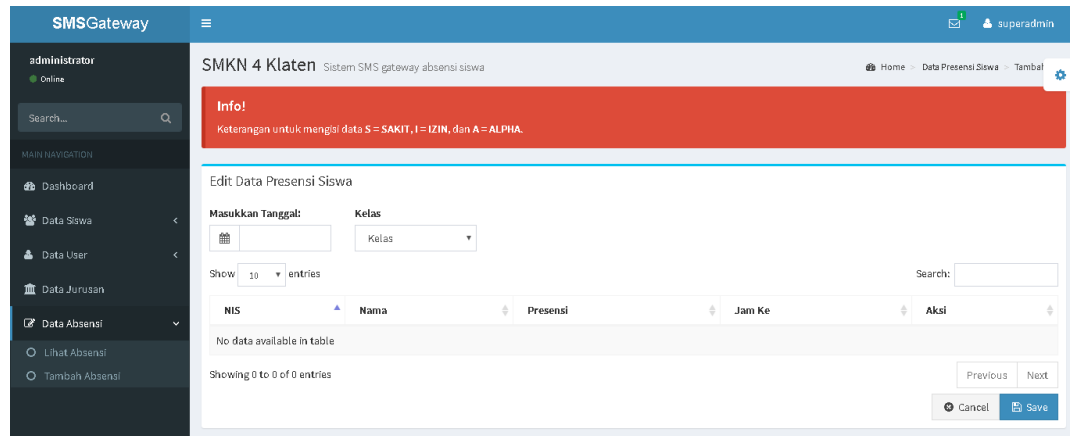
Implementasi antar muka halaman jurusan adalah sebagai berikut:

The screenshot shows the 'Manajemen Data Jurusan' page in the SMSGateway application. It features a 'Tambah Jurusan' form with a 'Save' button. Below the form, there is a section for 'Set data pesan' with a text area and a 'Save' button. To the right, there is a 'Daftar Jurusan' table with columns for 'No', 'Nama Jurusan', and 'Aksi'. The table contains four entries: 1. Absentansi, 2. TKJ, 3. RPL, and 4. Bahasa. The 'Aksi' column contains red 'X' icons. The table is paginated, showing 'Showing 1 to 4 of 4 entries'.

Gambar 28 Halaman Jurusan

9) Halaman lihat absensi

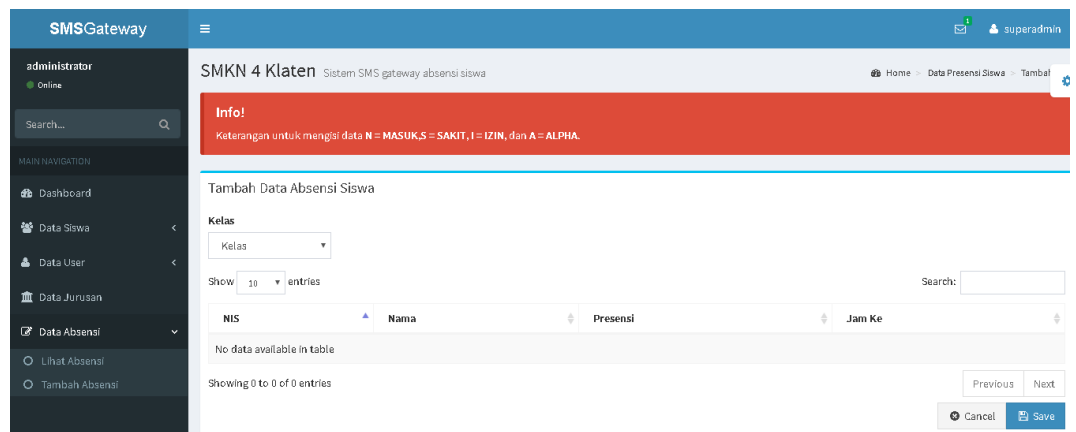
Implementasi antar muka halaman lihat absensi adalah sebagai berikut:



Gambar 29 Halaman lihat absensi

10) Halaman input absensi

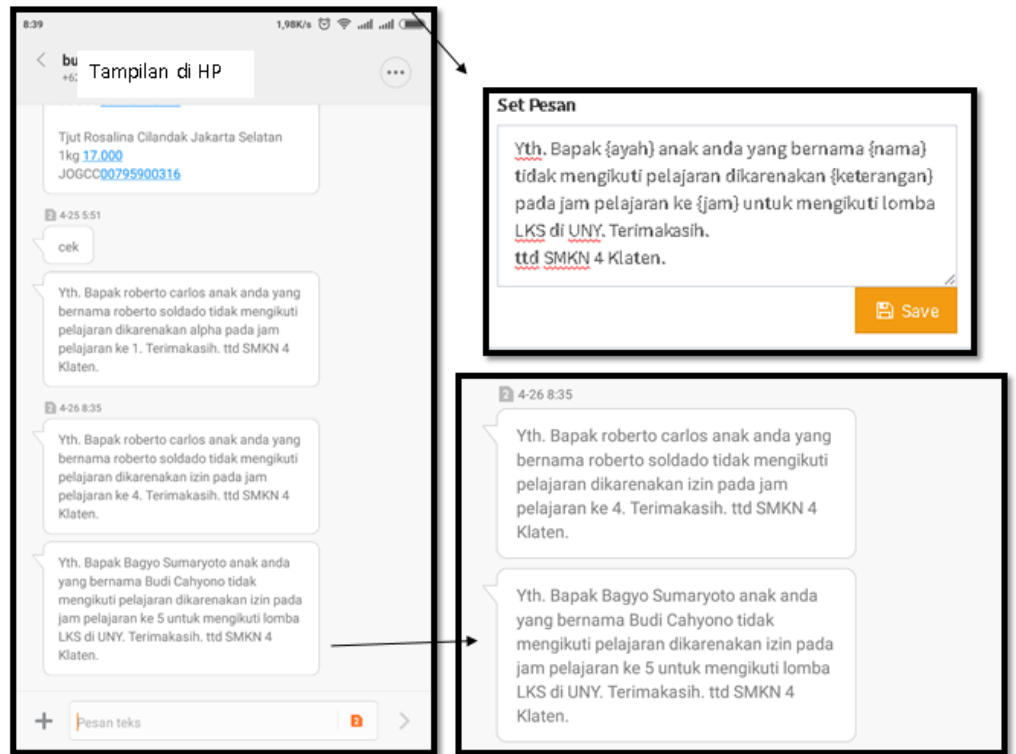
Implementasi antar muka halaman input absensi adalah sebagai berikut:



Gambar 30 Halaman *input* absensi

11) Tampilan *Handphone*

Implementasi antar muka halaman SMS pada *handphone* adalah sebagai berikut

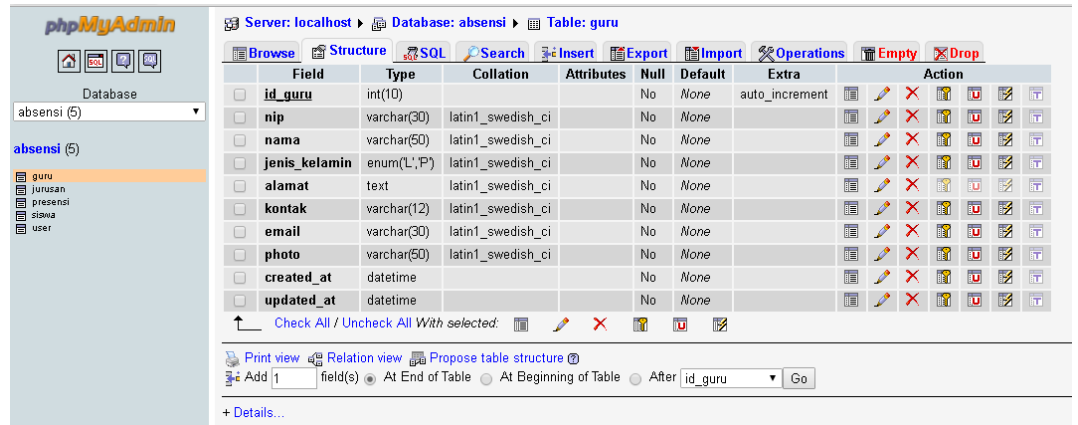


Gambar 31 Tampilan SMS

b. Implementasi basis data

Berikut ini merupakan implementasi dari perancangan basis data yaitu:

1) Tabel guru

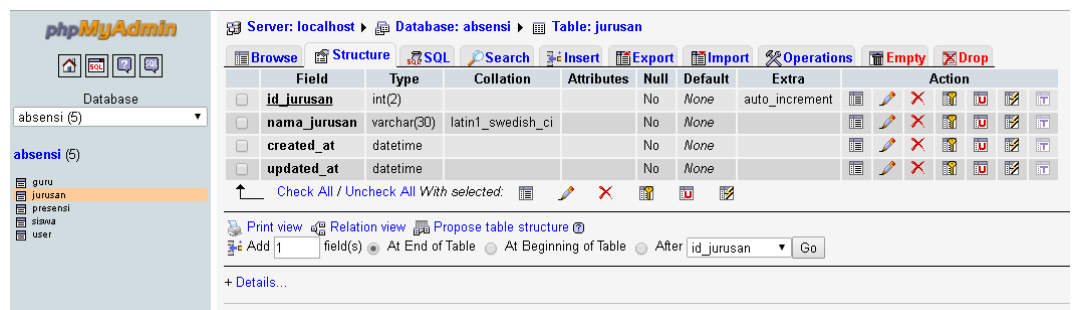


The screenshot shows the phpMyAdmin interface for the 'absensi' database. The 'guru' table structure is displayed with the following fields:

Field	Type	Collation	Attributes	Null	Default	Extra	Action
<input type="checkbox"/> id_guru	int(10)			No	None	auto_increment	
<input type="checkbox"/> nip	varchar(30)	latin1_swedish_ci		No	None		
<input type="checkbox"/> nama	varchar(50)	latin1_swedish_ci		No	None		
<input type="checkbox"/> jenis_kelamin	enum('L','P')	latin1_swedish_ci		No	None		
<input type="checkbox"/> alamat	text	latin1_swedish_ci		No	None		
<input type="checkbox"/> kontak	varchar(12)	latin1_swedish_ci		No	None		
<input type="checkbox"/> email	varchar(30)	latin1_swedish_ci		No	None		
<input type="checkbox"/> photo	varchar(50)	latin1_swedish_ci		No	None		
<input type="checkbox"/> created_at	datetime			No	None		
<input type="checkbox"/> updated_at	datetime			No	None		

Gambar 32 Tabel guru

2) Tabel jurusan

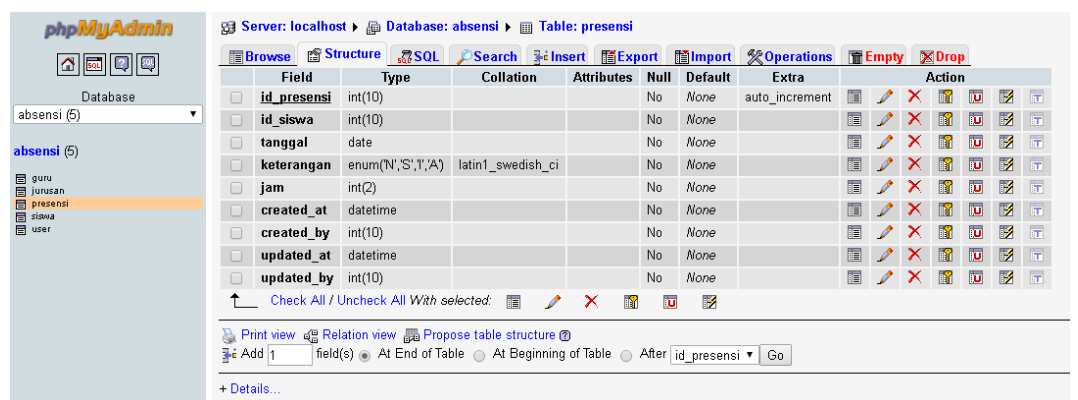


The screenshot shows the phpMyAdmin interface for the 'absensi' database. The 'jurusan' table structure is displayed with the following fields:

Field	Type	Collation	Attributes	Null	Default	Extra	Action
<input type="checkbox"/> id_jurusan	int(2)			No	None	auto_increment	
<input type="checkbox"/> nama_jurusan	varchar(30)	latin1_swedish_ci		No	None		
<input type="checkbox"/> created_at	datetime			No	None		
<input type="checkbox"/> updated_at	datetime			No	None		

Gambar 33 Tabel jurusan

3) Tabel presensi

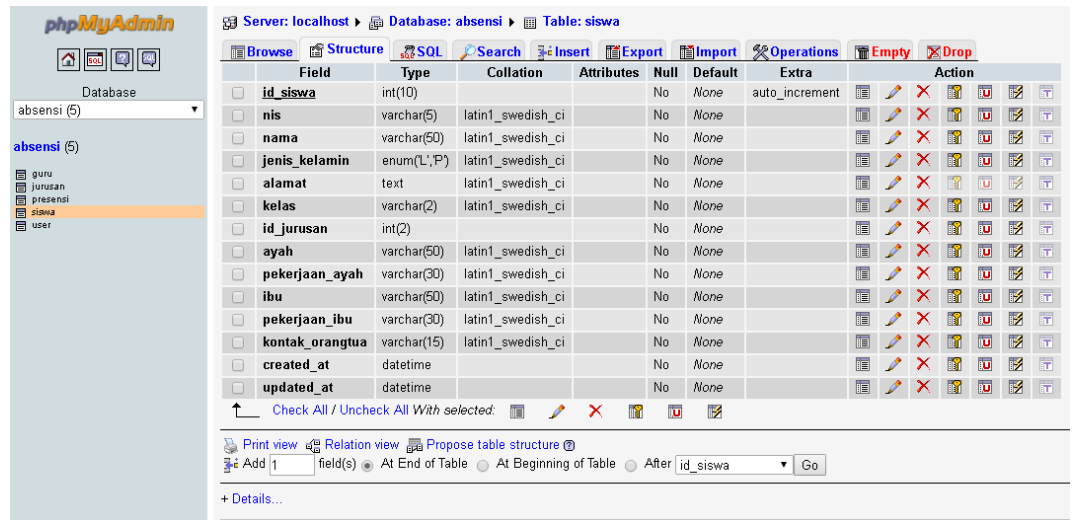


The screenshot shows the phpMyAdmin interface for the 'absensi' database. The 'presensi' table structure is displayed with the following fields:

Field	Type	Collation	Attributes	Null	Default	Extra	Action
<input type="checkbox"/> id_presensi	int(10)			No	None	auto_increment	
<input type="checkbox"/> id_siswa	int(10)			No	None		
<input type="checkbox"/> tanggal	date			No	None		
<input type="checkbox"/> keterangan	enum('N','S','I','A')	latin1_swedish_ci		No	None		
<input type="checkbox"/> jam	int(2)			No	None		
<input type="checkbox"/> created_at	datetime			No	None		
<input type="checkbox"/> created_by	int(10)			No	None		
<input type="checkbox"/> updated_at	datetime			No	None		
<input type="checkbox"/> updated_by	int(10)			No	None		

Gambar 34 Tabel presensi

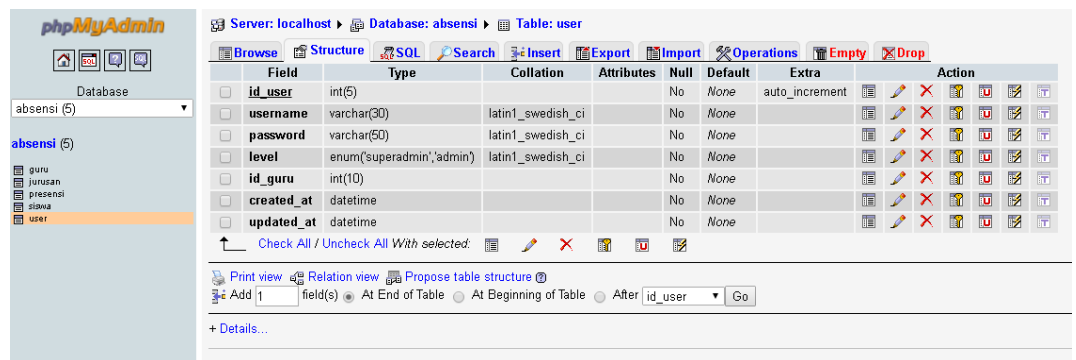
4) Tabel siswa



Field	Type	Collation	Attributes	Null	Default	Extra	Action				
<input type="checkbox"/> id_siswa	int(10)			No	None	auto_increment					
<input type="checkbox"/> nis	varchar(5)	latin1_swedish_ci		No	None						
<input type="checkbox"/> nama	varchar(50)	latin1_swedish_ci		No	None						
<input type="checkbox"/> jenis_kelamin	enum('L','P')	latin1_swedish_ci		No	None						
<input type="checkbox"/> alamat	text	latin1_swedish_ci		No	None						
<input type="checkbox"/> kelas	varchar(2)	latin1_swedish_ci		No	None						
<input type="checkbox"/> id_jurusan	int(2)			No	None						
<input type="checkbox"/> ayah	varchar(50)	latin1_swedish_ci		No	None						
<input type="checkbox"/> pekerjaan_ayah	varchar(30)	latin1_swedish_ci		No	None						
<input type="checkbox"/> ibu	varchar(50)	latin1_swedish_ci		No	None						
<input type="checkbox"/> pekerjaan_ibu	varchar(30)	latin1_swedish_ci		No	None						
<input type="checkbox"/> kontak_orangtua	varchar(15)	latin1_swedish_ci		No	None						
<input type="checkbox"/> created_at	datetime			No	None						
<input type="checkbox"/> updated_at	datetime			No	None						

Gambar 35 Tabel siswa

5) Tabel user



Field	Type	Collation	Attributes	Null	Default	Extra	Action				
<input type="checkbox"/> id_user	int(5)			No	None	auto_increment					
<input type="checkbox"/> username	varchar(30)	latin1_swedish_ci		No	None						
<input type="checkbox"/> password	varchar(50)	latin1_swedish_ci		No	None						
<input type="checkbox"/> level	enum('superadmin','admin')	latin1_swedish_ci		No	None						
<input type="checkbox"/> id_guru	int(10)			No	None						
<input type="checkbox"/> created_at	datetime			No	None						
<input type="checkbox"/> updated_at	datetime			No	None						

Gambar 36 Tabel user

7. Pengujian

Setelah aplikasi selesai dikembangkan, maka langkah selanjutnya yaitu pengujian aplikasi. Tahapan pengujian pada pengembangan ini meliputi *unit testing*, *integration testing*, *system testing*, dan *acceptance testing*.

a. Pengujian unit (*unit testing*)

Pengujian unit dilakukan dengan menggunakan metode *white box*. Pengujian ini dilakukan dengan menelusuri *source code* pada aplikasi untuk menemukan kesalahan. Pengujian ini sudah dimulai saat implementasi pemrograman berlangsung sampai aplikasi selesai dikembangkan. Salah satu metode pengujian *white box* yaitu melalui *basis path*. Hasil pengujian *white box* adalah sebagai berikut:

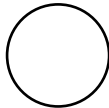

1) *Flowchart*

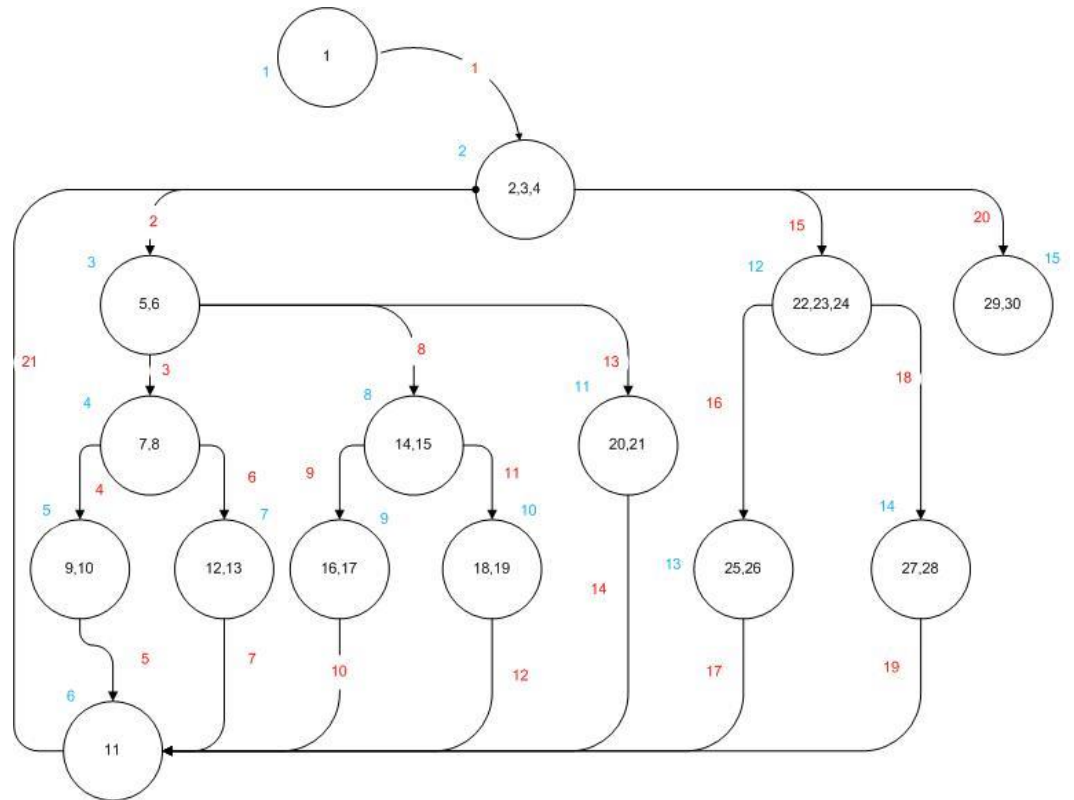
Flowchart program pada absensi SMS *Gateway* dapat dilihat pada bagian lampiran. Pada pembuatan *flowchart* menggunakan bantuan www.draw.io. *Flowchart* program terdapat pada lampiran 1.

2) *Flowgraph*

Flowgraph atau notasi grafik alir dapat diartikan dengan notasi yang merepresentasikan aliran kontrol. Setiap cabang ditunjukkan oleh *path* yang terpisah dan loop ditunjukkan oleh panah yang kembali ke kondisi node. Pembuatan *flowgraph* dapat menggunakan bantuan *flowchart* program.

Tabel 14 Simbol *flowgraph*

	Simpul atau node Menggambarkan satu atau lebih perintah prosedural. Urutan proses dan keputusan dapat dipetakan dalam satu node.
	Tanda panah atau <i>edge</i> Menggambarkan aliran kontrol. Setiap node harus mempunyai tujuan



Gambar 37 *Flowgraph*

Pada *flowgraph* terdapat angka yang berwarna biru dan merah. Angka merah merepresentasikan jumlah *edge* yaitu 21 buah, sedangkan angka biru merepresentasikan jumlah *node* yaitu 15 buah. Berikut keterangan untuk setiap *node*.

Tabel 15 Keterangan setiap *node*

No	Keterangan	No	Keterangan
1	Mulai	16	User melakukan klik lihat data user
2	Memunculkan halaman <i>login</i>	17	Menampilkan halaman data user yang tersimpan dalam <i>database</i>
3	Memasukkan <i>username</i> dan <i>password</i>	18	User melakukan klik tambah data user
4	Melakukan pengecekan apakah user masuk sebagai <i>admin</i> atau <i>superadmin</i>	19	Menampilkan <i>form</i> tambah data user

No	Keterangan	No	Keterangan
5	User masuk sebagai <i>superadmin</i>	20	User melakukan klik data jurusan
6	Menampilkan halaman utama <i>superadmin</i>	21	Menampilkan halaman jurusan
7	User melakukan klik data siswa	22	User masuk sebagai <i>admin</i>
8	Menampilkan sub data siswa	23	User melakukan klik data absensi
9	User melakukan klik lihat data siswa	24	Menampilkan sub data absensi
10	Menampilkan data siswa yang tersimpan dalam <i>database</i>	25	User melakukan klik lihat absensi
11	User melakukan klik home	26	Menampilkan data absensi yang terdapat pada <i>database</i>
12	User melakukan klik tambah data siswa	27	User melakukan klik tambah absensi
13	Menampilkan form data siswa	28	Menampilkan form tambah data absensi
14	User melakukan klik data <i>user</i>	29	User melakukan klik logout
15	Menampilkan sub data <i>user</i>	30	Selesai

3) Cyclomatic Complexity (CC)

Merupakan ukuran yang menunjukkan kompleksitas logic suatu program. CC dapat diperoleh dengan menghitung daerah yang dapat dibentuk oleh *flowgraph*.

Cyclomatic Complexity dapat dihitung dengan rumus

$$V(G) = E - N + 2$$

Dimana:

E=jumlah *edge* pada *flowgraph*

N=jumlah *node* pada *flowgraph*

Maka penghitungan *Cyclomatic Complexity* adalah:

Diketahui: $E=21$, $N=15$

Ditanya: $V(G)=?$

Jawab: $V(G)=E-N+2$
 $=21-15+2=8$

Jadi nilai $V(G)$ atau *Cyclomatic Complexity* adalah 8. Nilai $V(G)$ ini akan digunakan untuk menentukan *independent path* pada tahap selanjutnya.

4) *Independent Path*

Independent path merupakan jalur pada program yang menghubungkan *node* awal dengan *node* akhir. *Independent path* setidaknya bergerak melalui satu *edge* yang belum dilalui. Jumlah *path* sesuai dengan nilai $V(G)$, yaitu 8 buah.

Tabel 16 *Independent path*

Basis Flow	Independent path
Jalur 1	1-2-3-4-5-6-29-30
Jalur 2	1-2-3-4-5-6-7-8-9-10-11-29-30
Jalur 3	1-2-3-4-5-6-7-8-12-13-11-29-30
Jalur 4	1-2-3-4-5-6-14-15-16-11-29-30
Jalur 5	1-2-3-4-5-6-14-17-18-11-29-30
Jalur 6	1-2-3-4-5-6-20-21-11-29-30
Jalur 7	1-2-3-4-5-6-23-24-25-26-11-29-30
Jalur 8	1-2-3-4-5-6-23-24-27-28-29-30

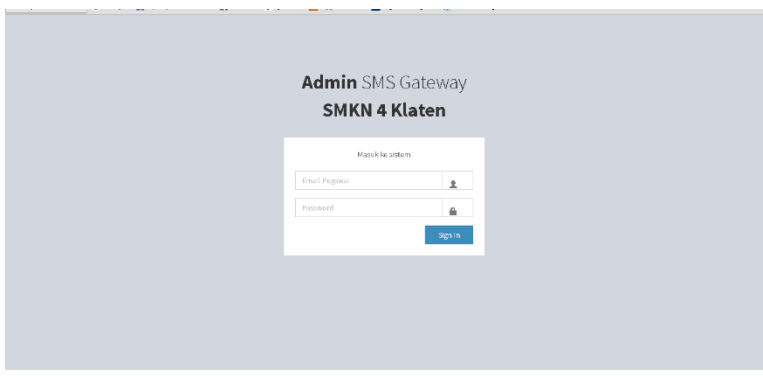
b. *Integration testing*

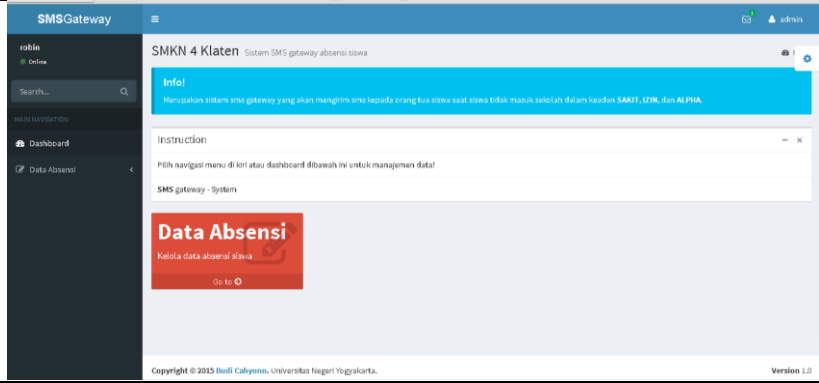
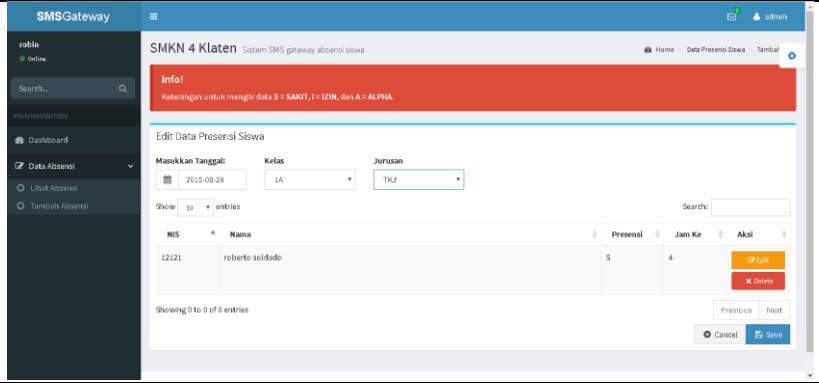
Tahap kedua pada pengujian aplikasi ini adalah *Integration testing*. Pada pengujian ini dilakukan untuk menguji aspek *functional suitability* dan *performance efficiency*. Metode pengujian yang dilakukan yaitu menggunakan *black box testing*.

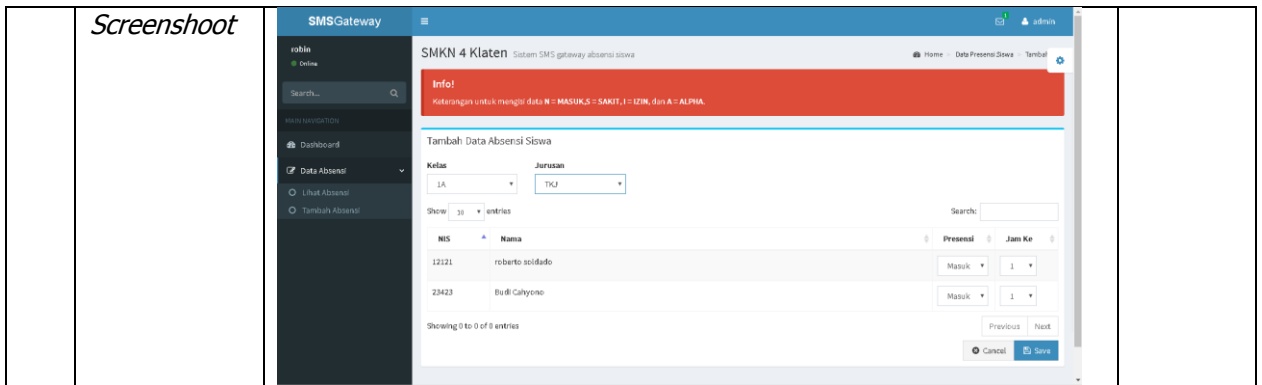
1) *Functional suitability*

Pada pengujian aspek *Functional suitability* menggunakan *test case*.

Tabel 17 *Test case*

No	Pengujian		Sukses/ Gagal
1	Nama tes	Buka halaman <i>Login</i>	Sukses
	Tujuan	Menguji halaman <i>login</i> dapat berjalan dengan sukses	
	Asumsi	Dapat terhubung dengan <i>database</i> pada saat pertama kali dijalankan dan menampilkan halaman <i>login</i>	
	Data tes	Jendela <i>browser</i> Google Chrome	
	<i>Steps</i>	Mengisi bar dengan alamat <i>web</i>	
	Hasil yang diharapkan	Tampil halaman <i>login</i>	
	Hasil pengamatan	Sesuai dengan harapan	
	<i>Screenshot</i>		
2	Nama tes	<i>Login administrator</i> dan <i>superadmin</i>	Sukses
	Tujuan	Menguji apakah <i>administrator</i> dapat <i>login</i> ke halaman <i>login</i> dengan benar	
	Asumsi	Dapat mencocokkan <i>username</i> dan <i>password administrator</i> dengan benar dan menampilkan halaman <i>administrator</i>	
	Data tes	- Jendela <i>login</i> - <i>Username</i> dan <i>password administrator</i>	
	<i>Steps</i>	- Mengisi kolom <i>username</i> dan kolom <i>password</i> - Klik <i>sign in</i>	
	Hasil yang diharapkan	<i>Administrator</i> dapat masuk ke halaman <i>administrator</i>	

	Hasil pengamatan	Sesuai dengan harapan	
	<i>Screenshoot</i>		
3	Nama tes	<i>Administrator</i> melihat data absensi	Sukses
	Tujuan	Menguji apakah halaman lihat absensi berjalan dengan benar	
	Asumsi	Dapat berjalan dan terhubung dengan <i>database</i> dengan benar	
	Data tes	Lihat absensi	
	<i>Steps</i>	<ul style="list-style-type: none"> - Klik data absensi - Klik lihat absensi 	
	Hasil yang diharapkan	Data lihat absensi dapat ditampilkan dengan benar	
	Hasil pengamatan	Sesuai dengan harapan	
	<i>Screenshoot</i>		
4	Nama tes	<i>Administrator</i> melakukan absensi	
	Tujuan	Menguji apakah halaman tambah absensi berjalan dengan benar	
	Asumsi	Dapat berjalan dan terhubung dengan <i>database</i>	
	Data tes	Data tambah absensi	
	<i>Steps</i>	Halaman <i>administrator</i> <ul style="list-style-type: none"> - Klik data absensi - Klik tambah absensi - Isi <i>form</i> - Klik <i>save</i> 	
	Hasil yang diharapkan	<i>Administrator</i> dapat menambah data absensi	
	Hasil pengamatan	Sesuai dengan harapan	



2) Performance efficiency

Pada pengujian ini menggunakan bantuan *tool* Yslow. Hasil pengujian dari pengembangan pelaporan sistem absensi SMS Gateway adalah sebagai berikut

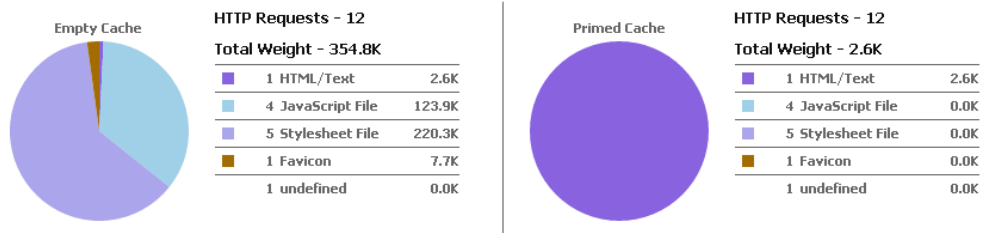


Gambar 38 Pengujian *efficiency*

Berdasarkan gambar berikut, hasil pengujian *efficiency* menggunakan bantuan *yslow* memperoleh *grade* B dengan skor *performance* 82.

Statistics The page has a total of 12 HTTP requests and a total weight of 354.8K bytes with empty cache

WEIGHT GRAPHS



Copyright © 2017 Yahoo! Inc. All rights reserved.

Gambar 39 Statistik *efficiency*

Hasil statistik tersebut menunjukkan terdapat 12 HTTP *request* dan total dokumen sebesar 354,8 K tanpa *cache* .Jika menggunakan ukuran dokumen menjadi 3,6 K.

Selanjutnya tabel berikut merupakan hasil pengujian *efficiency*.

Tabel 18 Hasil pengujian *efficiency*

<u>TYPE</u>	<u>SIZE (KB)</u>	<u>GZIP (KB)</u>	<u>COOKIE SENT (bytes)</u>	<u>EXPIRES (Y/M/D)</u>	<u>RESPONSE TIME (ms)</u>	<u>ETAG</u>
doc (1)	2.6K					
Doc	2.6K		53	11/19/1981	0	
js (4)	123.9K					
Js	0.5K		53	no expires	0	"208-5113c8219eb80"
Js	84.3K		53	no expires	0	"1495f-510d009ba4e80"
Js	3.6K		53	no expires	0	"e6f-4f995aa36bd00"
Js	35.4K		53	no expires	0	"8a7c-510d009ba4e80"
css (5)	220.3K					
css	0.0K		53	no expires	0	
css	21.9K		53	no expires	0	"55e0-50180fddb4280"
css	5.4K		53	no expires	0	"152c-4f995aa36bd00"

<u>TYPE</u>	<u>SIZE (KB)</u>	<u>GZIP (KB)</u>	<u>COOKIE SENT (bytes)</u>	<u>EXPIRES (Y/M/D)</u>	<u>RESPONSE TIME (ms)</u>	<u>ETAG</u>
css	75.8K		53	no expires	0	"12844-510d009ba4e80"
css	117.1K		53	no expires	0	"1c99e-510d009ba4e80"
favicon (1)	7.7K					
favicon	7.7K		53	no expires	0	"1e66-4d922b19fd4c0"
font (1)	0.0K					
font	0.0K		53	no expires	0	

c. *System testing*

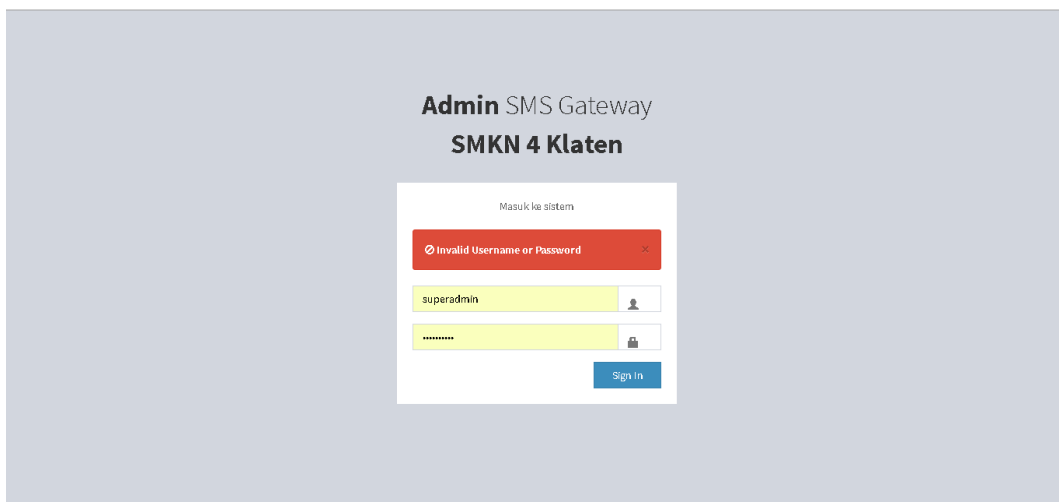
Pada pengujian *system testing* merupakan tahap pengujian ke tiga dalam pengujian perangkat lunak. Pengujian ini dilakukan dengan menggunakan metode *black box testing* untuk menguji aspek *maintainability* dan *compatibility*.

1) *Maintainability*

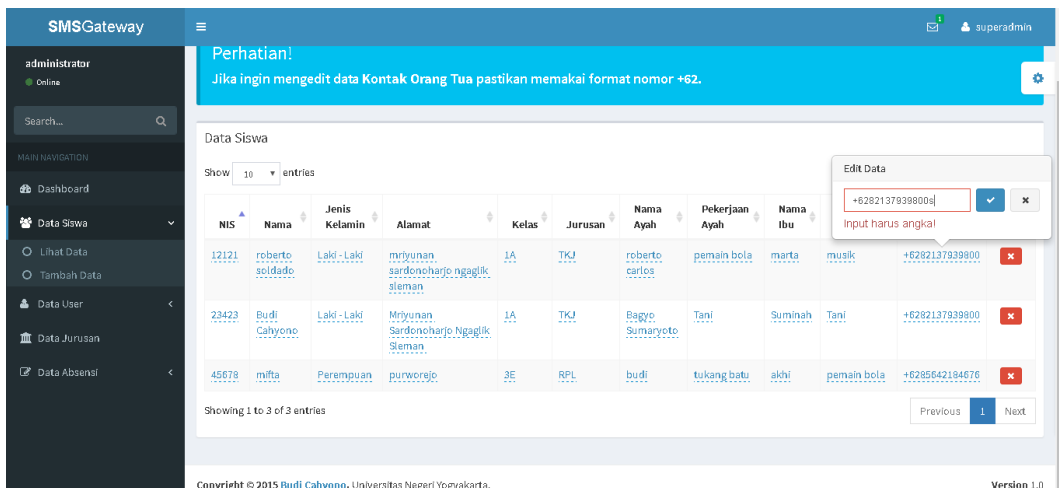
Pada pengujian *maintainability* ini dimaksudkan untuk menguji apabila mengalami kegagalan dalam input data, tampilan sistem, dan apakah sistem mudah untuk dilakukan modifikasi. Pengujian ini dilakukan secara operasional dengan menggunakan instrumen *maintainability* (Land, 2002). Aspek yang terdapat dalam instrumen tersebut adalah *instrumentation*, *consistency*, dan *simplicity*.

Tabel 19 Instrumen *maintainability*

Aspek	Aspek yang dinilai	Hasil yang akan diperoleh
<i>instrumentation</i>	Terdapat peringatan pada sistem pengolahan data untuk mengidentifikasi kesalahan	Ketika <i>user</i> melakukan kesalahan dalam operasionalnya, sistem akan memberikan peringatan.
<i>consistency</i>	Penggunaan satu bentuk perancangan pada seluruh rancangan sistem.	Bentuk rancangan sistem pengolah data mempunyai satu bentuk yang sama.
<i>Simplicity</i>	Kemudahan dalam pengolahan, perbaikan, dan pengembangan sistem.	Mudah untuk diolah, diperbaiki, dan dikembangkan. Hal ini dapat diidentifikasi melalui penulisan kode program.



Gambar 40 *Error* memasukkan *login*



Gambar 41 *Error* edit data

2) *Compatibility*

Pengujian ini dilakukan dengan pengamatan secara langsung dengan percobaan pada berbagai kondisi lingkungan, diantaranya adalah pada berbagai *browser* yang berbeda.

Hasil pengujian *compatibility* dapat dilihat pada lampiran 2

3) *Acceptance Testing*

Pada pengujian ini dilakukan untuk menguji aspek *usability*. Pengujian ini dilakukan dengan menggunakan kuesioner yang disusun oleh J R. Lewis dan timnya yang berjudul "*IBM Computer Usability Satisfaction Questionnaires: Psychometrics Evaluation and Istruction for Use*".

Hasil dari pengujian *acceptance testing* terdapat pada tabel dibawah ini.

Table 20 Hasil pengujian *acceptance testing*

No	Item	Varians
1	$\sigma_{(1)}^2$	0,21
2	$\sigma_{(2)}^2$	0,46
3	$\sigma_{(3)}^2$	0,23
4	$\sigma_{(4)}^2$	0,49
5	$\sigma_{(5)}^2$	0,24
6	$\sigma_{(6)}^2$	0,36
7	$\sigma_{(7)}^2$	0,59
8	$\sigma_{(8)}^2$	0,35
9	$\sigma_{(9)}^2$	0,51
10	$\sigma_{(10)}^2$	0,59
11	$\sigma_{(11)}^2$	0,26
12	$\sigma_{(12)}^2$	0,46
13	$\sigma_{(13)}^2$	0,65
14	$\sigma_{(14)}^2$	0,19
15	$\sigma_{(15)}^2$	0,13
16	$\sigma_{(16)}^2$	0,39
17	$\sigma_{(17)}^2$	0,40
18	$\sigma_{(18)}^2$	0,51
19	$\sigma_{(19)}^2$	0,23
Total	$\sum \sigma_i^2$	7,22

$$\sigma_i^2 = \frac{\sum X^2 - \frac{(\sum X)^2}{N}}{N}$$

$$= \frac{122266 - \frac{(1558)^2}{20}}{20}$$

$$= 44,89$$

Untuk menentukan reliabilitas menggunakan rumus *Alpha Cronbach* sebagai berikut :

$$r_{11} = \left[\frac{n}{n-1} \right] \left[1 - \frac{\sum \sigma_i^2}{\sigma_t^2} \right]$$

$$= \left[\frac{19}{19-1} \right] \left[1 - \frac{7,22}{44,89} \right]$$

$$= (1,0556)(0,8392)$$

$$= 0,885$$

Hasil perhitungan koefisien *alpha cronbach* dari data tersebut hasil reliabilitas adalah 0,885 dari 19 *items*. Berdasarkan perhitungan tersebut kemudian dikonversi menjadi nilai kualitatif menggunakan tabel konsistensi *alpha cronbach*.

Tabel 21 Konsistensi *alpha cronbach*

No	Nilai R	Interprestasi
1.	R > 0.9	<i>Excellent</i>
2.	0.9 > R > 0.8	<i>Good</i>
3.	0.8 > R > 0.7	<i>Acceptabel</i>
4.	0.7 > R > 0.6	<i>Questionable</i>
5.	0.6 > R > 0.5	<i>Poor</i>
6.	R < 0.5	<i>Unacceptabel</i>

Jika dibandingkan dengan tabel konsistensi *alpha cronbach* tersebut, hasil uji *usability* aplikasi bernilai **0,885** menunjukkan nilai "**Good**".

BAB V

KESIMPULAN DAN SARAN

A. Kesimpulan

Berdasarkan hasil penelitian dan pembahasan yang telah dilakukan mengenai pengembangan dan analisis kualitas pengembangan sistem pelaporan absensi siswa berbasis SMS *Gateway* untuk pengawasan orang tua di SMK N 4 Klaten, maka dapat diambil beberapa kesimpulan sebagai berikut:

1. Aplikasi pelaporan absensi siswa berbasis SMS *Gateway* untuk pengawasan orang tua di SMK N 4 Klaten dibangun menggunakan *framework* CodeIgniter berdasarkan model pengembangan V-Model yaitu tahap pengembangan (*requirements, specification, design, dan implementation*) dan tahap pengujian (*unit testing, integration testing, system testing, dan acceptance testing*). Aplikasi ini dikembangkan dengan memiliki 2 *user* (admin dan *superadmin*), *user superadmin* dapat melakukan manajemen siswa, *user*, jurusan, dan absensi. Sedangkan user admin hanya dapat melakukan manajemen absensi.
2. Pengujian kualitas aplikasi diuji menggunakan standar ISO 25010 melalui tahapan *unit testing, integration testing, system testing, dan acceptance testing*. Pada tahap *acceptance testing* melalui beberapa aspek yaitu *functional suitability, performance efficiency, usability, maintainability, dan compatibility*. Pengujian aspek *functional suitability* dilakukan dengan menggunakan *test case*. Pada pengujian *performance efficiency* menggunakan bantuan *tool* yslow didapat *grade* B dengan nilai 82. Pada

pengujian aspek *usability* yang dilakukan menggunakan angket kuisisioner dari James R. Lewis kepada 20 orang guru SMK N 4 Klaten diperoleh koefisien *alpha cronbach* sebesar 0,885 yang menunjukkan nilai "Good". Pada pengujian *maintainability* menggunakan instrumen Land dengan menunjukkan dokumentasi aplikasi. Pada pengujian *compatibility* menggunakan dokumentasi untuk menjalankan aplikasi pada *browser* yang berbeda (Chrome, Safari, dan Mozilla).

B. Saran

1. Pembuatan SMS *Gateway* menggunakan *sender ID*. Ketika pesan masuk menampilkan nama instansi lembaga bukan nomor pribadi.
2. Tampilan lihat absensi dapat me-*filter* tiap hari, minggu, bulan, semester dan tahun.

Daftar Pustaka

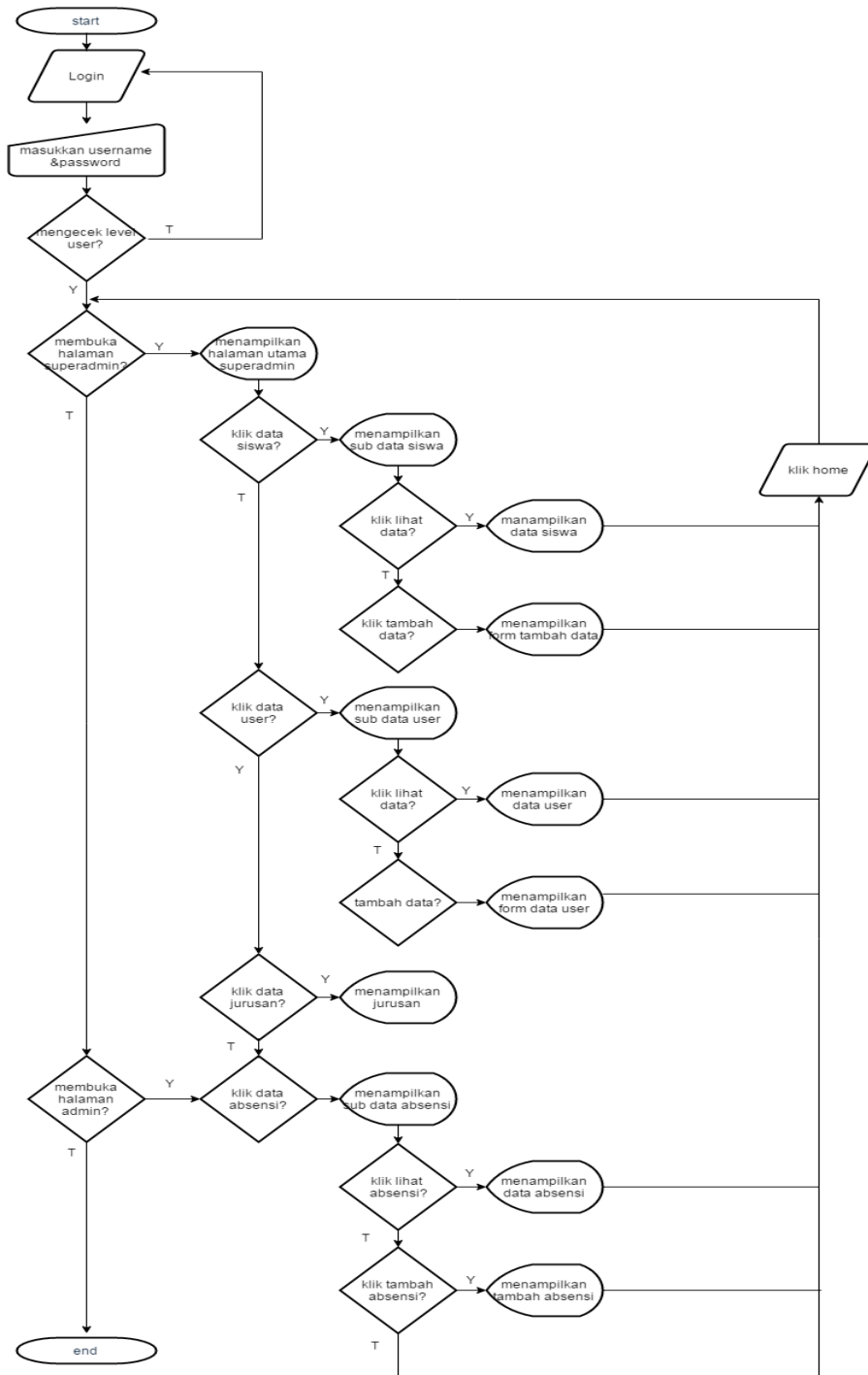
- Assassa, Ghazy.(2007). Test Case Template. Diakses dari http://faculty.ksu.edu.sa/ghazy/CSC342_Tools/Forms/AllItems.aspx pada 29 Juli 2015 jam 7.50
- Sudrajat, Akhmad. (2007). Tentang Kehadiran dan Ketidakhadiran Siswa di Sekolah. Diakses dari <https://akhmadsudrajat.wordpress.com/2010/10/17/tentang-kehadiran-dan-ketidakhadiran-siswa-di-sekolah> Pada Tanggal 7 Juli 2016, Jam 8:16 WIB.
- Aminudin, (2014). *Program Absensi Siswa Realtime dengan PHP &SMS Gateway*. Yogyakarta: Lokomedia.
- Dessy Irmawati, Yuniar Indri Hapsari. (2014). *Sistem Informasi Kearsipan untuk Meningkatkan Kualitas Pelayanan*. Jurnal Pendidikan Teknologi dan Kejuruan, 138.
- Gliem, J. A., & Gliem, R. R. (2003). Calculating, Interpreting, and Reporting Cronbach's Alpha Reliability Coefficient for Likert-Type Scales. 2003. Midwest Research to Practice Conference in Adult, Continuing, and Community Education. Pg. 82.
- Hariyanto, Didik. (2008). *Pengembangan Sistem Informasi Akademik Mahasiswa Berbasis Teknologi WAP di Jurusan Pendidikan Teknik Elektro FT UNY*. Jurnal Pendidikan Teknologi dan Kejuruan, 147.
- ISO. (2011). ISO 25010. Diakses dari: <https://www.iso.org/obp/ui/#iso:std:iso-iec:25010:ed-1:v1:en>. Pada tanggal 21 Juli 2016. Jam 20.00 WIB.
- Lewis, J.R. (1995). IBM Computer Usability Satisfaction Questionnaires: Psychometric Evaluation and Instruction for Use. International Journal of Human-Computer Interaction. Pg. 57-58.
- Madcoms, Litbang. 2011. *Aplikasi Web Database dengan Dreamweaver dan PHP--MySQL*. Yogyakarta: Andi.
- Maulana, Sofyan (2015). *Trik kolaborasi VB.Net &SMS Gateway*. Yogyakarta: Afsa Solution.
- Niknejad, Aida. (2011). A Quality Evaluation of An Android Smartphone Application. Swedia.
- Nugroho, Bunafit. (2005). *Database Relational dengan MySQL*. Yogyakarta: Andi.

- Pressman, R. S. (2012). (Rekayasa Perangkat Lunak - Buku Dua, Pendekatan Praktis Edisi 7). Andi.
- Rachidan, Adhi & Sikumbang, Andy. (2007). *Mastering CMS dengan Mambo/Joomla*. Jakarta: Elex Media Komputindo.
- Ramadhina, S. (2015). Pembuatan Sistem Informasi Manajemen di Bengkel Sekolah Menengah Kejuruan Negeri 3 Yogyakarta. *Jurnal Pendidikan Teknologi dan Kejuruan*, 329.
- Riduwan, & Akdon. (2008). *Rumus dan Data dalam Analisis Statistika*. Bandung: Alfabeta.
- Rizoldi, Romzi Imron. (2009). *Membuat Sendiri SMS Gateway (ESME) Berbasis Protokol SMPP*. Yogyakarta: Andi.
- S, Rosa A., & Shalahuddin, M. (2013). *Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek*. Bandung: Informatika.
- Saputra, Agus. (2013). *Step by Step Membangun Aplikasi Bioskop dan SMS untuk Panduan Skripsi*. Jakarta: Elex Media Komputindo.
- Sugiyono. (2014). *Metode Penelitian Administratorisasi*. Bandung: Alfabeta
- Supriyono, Joko. (2015). *Perancangan Sistem Informasi Absensi Karyawan Menggunakan Konsep Data Mining Pada PT. Kalila Indonesia. Sekolah Tinggi Manajemen dan Ilmu Komputer (STMIK) Raharja*. Tangerang: Skripsi.
- Triwahyuni, Atin. (2012). *Sistem Informasi Absensi Siswa (Studi Kasus Perguruan Islam Mathali'ul Falah Pati Jawa Tengah)*. *Jurnal Teknologi dan Informatika (TEKNOMATIKA)*. Hlm 1-8.
- Veenendaal, E. v. (2014). *The New Standard for Software Product Quality. Testing Experience*. Diakses 25 oktober 2016, dari http://www.erikvanveenendaal.nl/NL/files/TE25_van_Veenendaal.pdf
- Wahana. (2015). *Mudah Membuat Aplikasi SMS Gateway dengan CodeIgniter*. Yogyakarta: Wahana Komputer.
- Winarti, Dwi. (2011). *Pengembangan Aplikasi Absensi Pegawai Berbasis Web dan Absensi Siswa Berbasis SMS SMA Negeri 1 Imogiri*. *Artikel Teknik Informatika*. Hlm. 1-15.
- Watkins, John., & Simon Mills. (2011). *Testing IT*. New York: Cambridge University Press.

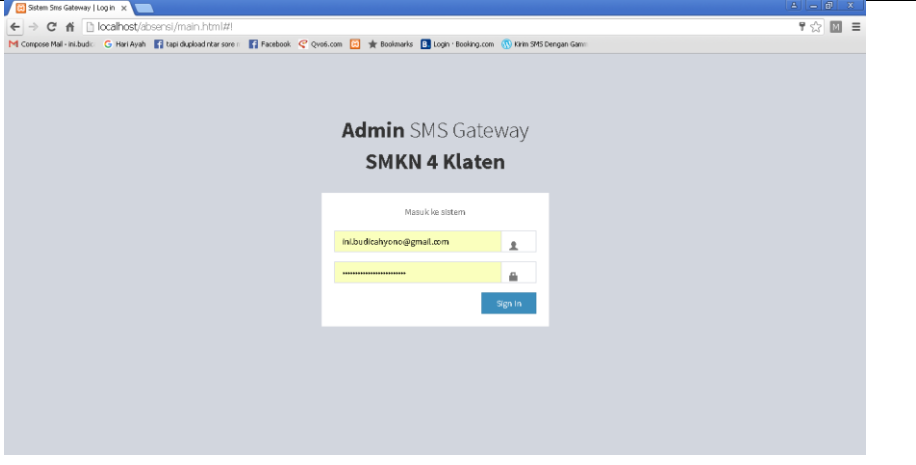
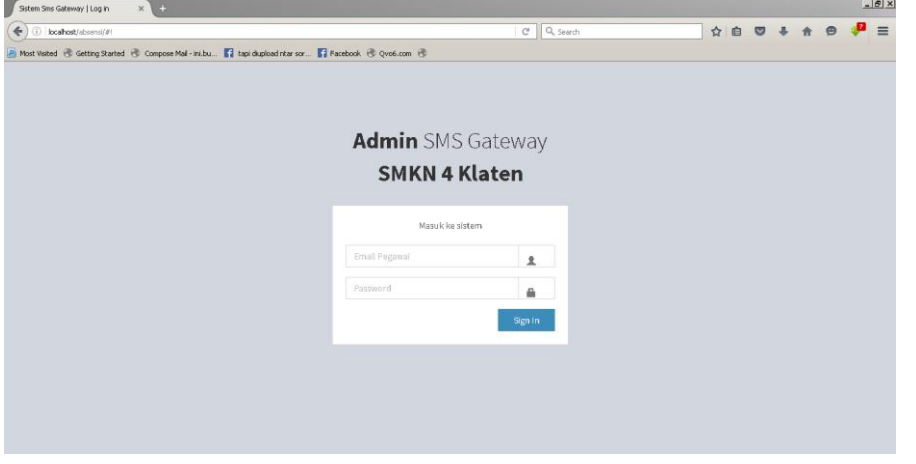
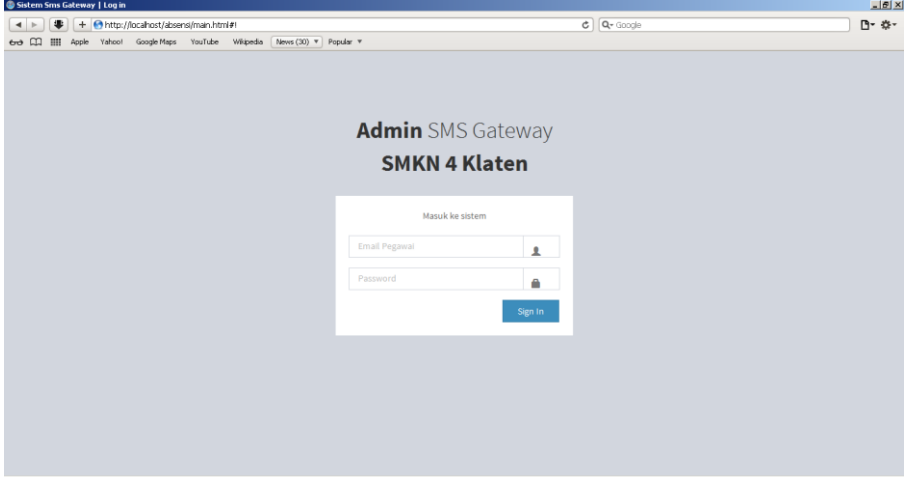
LAMPIRAN

GAMBAR USE CASE

PENGEMBANGAN SISTEM PELAPORAN ABSENSI SISWA BERBASIS SMS GATEWAY UNTUK PENGAWASAN ORANG TUA DI SMK N 4 KLATEN



Lampiran 2 Hasil pengujian *compatibility*

No	Nama <i>Browser</i>	Hasil pengujian
1	Google Chrome	
2	Mozilla Firefox	
3	Safari browser	

Lampiran 3 Matriks ISO 9126 dan ISO 25010

No	Model ISO-9126	Model ISO-25010	Keterangan
1	Functionality	Functional Suitability	Pada model ISO-25010 <i>Functionality</i> berubah menjadi <i>Functional Suitability</i> karena lebih menunjukkan spesifikasi dari fungsionalitas.
	<i>Suitability</i>	<i>Appropriateness</i>	Pada model ISO-25010 <i>Suitability</i> berubah nama menjadi <i>Appropriateness</i> .
	<i>Accuracy</i>	<i>Accuracy</i>	Pada model ISO-25010 sub faktor <i>Accuracy</i> masih sama seperti model ISO-9126.
	<i>Interoperability</i>		Pada model ISO-25010 sub faktor <i>Interoperability</i> beralih menjadi sub faktor dari <i>Compability</i> .
	<i>Security</i>		Pada model ISO-25010 sub faktor <i>Security</i> berubah menjadi faktor kualitas tersendiri.
2	Reliability	Reliability	Faktor <i>Reliability</i> pada model ISO-9126 masih tetap ada pada model ISO-25010.
	<i>Maturity</i>	<i>Availability</i>	Pada model ISO-25010 sub faktor <i>Maturity</i> berubah menjadi <i>Availability</i> karena berdasarkan maksud dari <i>Maturity</i> pada model ISO-9126 lebih cenderung pada kegagalan perangkat lunak yang bisa diwakili oleh <i>Fault Tolerance</i> .
	<i>Fault tolerance</i>	<i>Fault tolerance</i>	Sub-faktor <i>Fault Tolerance</i> masih tetap ada pada model ISO-25010 seperti pada model ISO-9126.
	<i>Recoverability</i>	<i>Recoverability</i>	Sub-faktor <i>Recoverability</i> masih tetap ada pada model ISO-25010 seperti pada model ISO-9126.
3	Usability	Usability	Sub-faktor <i>Usability</i> masih tetap ada pada model ISO-25010 seperti pada model ISO-9126.
	<i>Understandability</i>	<i>Appropriateness Recognizability</i>	Pada model ISO-25010 sub faktor <i>Understandability</i> berubah nama menjadi <i>Appropriateness Recognizability</i> karena lebih spesifik
	<i>Learnability</i>	<i>Learnability</i>	Sub-faktor <i>Learnability</i> masih tetap ada pada model ISO-25010 seperti pada model ISO-9126.

	<i>Operability</i>	<i>Operability</i>	Sub-faktor <i>Operability</i> masih tetap ada pada model ISO-25010 seperti pada model ISO-9126.
		<i>User Error Protection</i>	<i>User Error Protection</i> merupakan sub faktor baru dari <i>Usability</i> dari model ISO-25010
		<i>User Interface Aesthetics</i>	<i>User Interface Aesthetics</i> merupakan sub faktor baru dari <i>Usability</i> dari model ISO-25010
		<i>Accessibility</i>	<i>Accessibility</i> merupakan sub faktor baru dari <i>Usability</i> dari model ISO-25010
4	<i>Efficiency</i>	<i>Performance Efficiency</i>	Pada model ISO-25010 faktor <i>Efficiency</i> berubah nama menjadi <i>Performance Efficiency</i>
	<i>Time-behaviour</i>	<i>Time-behaviour</i>	Sub-faktor <i>Time-behaviour</i> masih tetap ada pada model ISO-25010 seperti pada model ISO-9126.
	<i>Resource-utilisation</i>	<i>Resource-utilisation</i>	Sub-faktor <i>Resource-utilisation</i> masih tetap ada pada model ISO-25010 seperti pada model ISO-9126.
5	<i>Maintainability</i>	<i>Maintainability</i>	Sub-faktor <i>Maintainability</i> masih tetap ada pada model ISO-25010 seperti pada model ISO-9126.
	<i>Analyzability</i>	<i>Analyzability</i>	Sub-faktor <i>Analyzability</i> masih tetap ada pada model ISO-25010 seperti pada model ISO-9126.
6	<i>Portability</i>	<i>Portability</i>	Sub-faktor <i>Portability</i> masih tetap ada pada model ISO-25010 seperti pada model ISO-9126.
	<i>Adaptability</i>	<i>Adaptability</i>	Sub-faktor <i>Adaptability</i> masih tetap ada pada model ISO-25010 seperti pada model ISO-9126.
	<i>Installability</i>	<i>Installability</i>	Sub-faktor <i>Installability</i> masih tetap ada pada model ISO-25010 seperti pada model ISO-9126.
	<i>Replaceability</i>	<i>Replaceability</i>	Sub-faktor <i>Replaceability</i> masih tetap ada pada model ISO-25010 seperti pada model ISO-9126.
	<i>Co-existence</i>		Pada model ISO-25010 sub faktor <i>Co-existence</i> beralih menjadi sub faktor dari <i>Compatibility</i> .

7		Security	Sebelumnya <i>Security</i> merupakan sub-faktor dari <i>Functionality</i> pada model ISO-9126
		<i>Confidentiality</i>	Sub-faktor <i>Confidentiality</i> merupakan sub-faktor baru pada ISO-25010, yaitu terdapat pada faktor <i>Security</i>
		<i>Integrity</i>	Sub-faktor <i>Integrity</i> merupakan sub-faktor baru pada ISO-25010, yaitu terdapat pada faktor <i>Security</i>
		<i>Non-repudiation</i>	Sub-faktor <i>Non-repudiation</i> merupakan sub-faktor baru pada ISO-25010, yaitu terdapat pada faktor <i>Security</i>
		<i>Accountability</i>	Sub-faktor <i>Accountability</i> merupakan sub-faktor baru pada ISO-25010, yaitu terdapat pada faktor <i>Security</i>
		<i>Authenticity</i>	Sub-faktor <i>Authenticity</i> merupakan sub-faktor baru pada ISO-25010, yaitu terdapat pada faktor <i>Security</i>
8		Compatibility	<i>Compatibility</i> merupakan faktor baru pada ISO 25010
		<i>Co-existence</i>	<i>Co-existence</i> sebelumnya merupakan sub faktor dari <i>Portability</i>
		<i>Interoperability</i>	<i>Interoperability</i> sebelumnya merupakan sub faktor dari <i>Functionality</i>

Lampiran 4 Hasil kuesioner angket *usability*

Tabel 22 Hasil kuesioner angket *usability*

No Responden	Item Pertanyaan																			Total	Total kuadrat
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19		
1	5	5	5	5	4	5	4	4	4	4	4	4	4	4	4	4	4	4	4	81	6561
2	4	4	4	2	4	4	4	4	2	2	4	4	4	4	5	4	4	4	4	71	5041
3	4	4	3	4	4	3	2	2	3	3	4	3	2	3	4	3	3	4	3	61	3721
4	4	4	5	5	4	4	5	4	4	4	5	4	4	4	4	3	4	3	4	78	6084
5	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	76	5776
6	5	4	4	4	4	4	4	4	2	4	4	2	2	4	4	4	4	2	4	69	4761
7	4	4	4	4	4	3	4	4	4	4	4	4	5	4	4	5	2	2	4	73	5329
8	4	4	4	4	4	4	3	4	3	3	4	4	4	4	4	3	4	3	4	71	5041
9	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	76	5776
10	5	4	5	5	5	4	5	4	4	4	5	5	4	4	4	5	4	4	5	85	7225
11	4	4	4	4	4	5	4	4	3	4	4	4	4	5	4	5	5	4	4	79	6241
12	4	2	4	4	5	5	5	4	4	3	4	4	5	5	4	4	4	4	4	78	6084
13	4	4	4	5	5	4	5	4	4	5	4	4	4	4	4	4	4	4	5	81	6561
14	5	5	5	5	5	5	5	4	5	5	4	5	5	4	5	4	4	5	5	90	8100
15	4	4	4	5	5	4	4	5	4	5	5	4	4	4	4	4	4	4	4	81	6561
16	4	5	4	4	5	4	4	4	4	4	4	4	4	4	4	4	4	3	4	77	5929
17	4	5	4	4	4	5	5	5	4	3	3	5	5	5	4	5	5	4	4	83	6889
18	5	5	4	4	5	4	5	5	4	5	5	4	4	4	4	4	4	4	4	83	6889
19	5	5	4	5	5	5	5	4	4	4	5	5	5	4	5	5	5	4	5	89	7921
20	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	76	5776
jumlah	86	84	83	85	88	84	85	81	74	78	84	81	81	82	83	82	80	74	83		
jumlah kuadrat	7396	7056	6889	7225	7744	7056	7225	6561	5476	6084	7056	6561	6561	6724	6889	6724	6400	5476	6889	1558	122266
variasi soal	0.21	0.46	0.23	0.49	0.24	0.36	0.59	0.35	0.51	0.59	0.26	0.45	0.65	0.19	0.13	0.39	0.40	0.51	0.23		

Dengan data yang terdapat pada tabel, dicari varians tiap-tiap item terlebih dahulu, kemudian varians tiap-tiap item dijumlahkan.

$$\sigma_{(1)}^2 = \frac{374 - \frac{(86)^2}{20}}{20} = 0,21$$

$$\sigma_{(11)}^2 = \frac{358 - \frac{(84)^2}{20}}{20} = 0,26$$

$$\sigma_{(2)}^2 = \frac{362 - \frac{(84)^2}{20}}{20} = 0,46$$

$$\sigma_{(12)}^2 = \frac{337 - \frac{(81)^2}{20}}{20} = 0,46$$

$$\sigma_{(3)}^2 = \frac{349 - \frac{(83)^2}{20}}{20} = 0,23$$

$$\sigma_{(13)}^2 = \frac{341 - \frac{(81)^2}{20}}{20} = 0,65$$

$$\sigma_{(4)}^2 = \frac{371 - \frac{(85)^2}{20}}{20} = 0,49$$

$$\sigma_{(14)}^2 = \frac{340 - \frac{(82)^2}{20}}{20} = 0,19$$

$$\sigma_{(5)}^2 = \frac{392 - \frac{(88)^2}{20}}{20} = 0,24$$

$$\sigma_{(15)}^2 = \frac{347 - \frac{(83)^2}{20}}{20} = 0,13$$

$$\sigma_{(6)}^2 = \frac{360 - \frac{(84)^2}{20}}{20} = 0,36$$

$$\sigma_{(16)}^2 = \frac{344 - \frac{(82)^2}{20}}{20} = 0,39$$

$$\sigma_{(7)}^2 = \frac{373 - \frac{(85)^2}{20}}{20} = 0,59$$

$$\sigma_{(17)}^2 = \frac{328 - \frac{(80)^2}{20}}{20} = 0,40$$

$$\sigma_{(8)}^2 = \frac{335 - \frac{(81)^2}{20}}{20} = 0,35$$

$$\sigma_{(18)}^2 = \frac{284 - \frac{(74)^2}{20}}{20} = 0,51$$

$$\sigma_{(9)}^2 = \frac{284 - \frac{(74)^2}{20}}{20} = 0,51$$

$$\sigma_{(19)}^2 = \frac{349 - \frac{(83)^2}{20}}{20} = 0,23$$

$$\sigma_{(10)}^2 = \frac{316 - \frac{(74)^2}{20}}{20} = 0,59$$

Jumlah varians semua item :

$$\begin{aligned}\sum \sigma_i^2 &= 0,21 + 0,46 + 0,23 + 0,49 + 0,24 + 0,36 + 0,59 + 0,35 \\ &\quad + 0,51 + 0,59 + 0,26 + 0,46 + 0,65 + 0,19 + 0,13 + 0,39 \\ &\quad + 0,40 + 0,51 + 0,23 \\ &= 7,22\end{aligned}$$

$$\sigma_i^2 = \frac{\sum X^2 - \frac{(\sum X)^2}{N}}{N}$$

$$= \frac{122266 - \frac{(1558)^2}{20}}{20}$$

$$= 44,89$$

Untuk menentukan reliabilitas menggunakan rumus *Alpha Cronbach* sebagai berikut :

$$r_{11} = \left[\frac{n}{n-1} \right] \left[1 - \frac{\sum \sigma_i^2}{\sigma_t^2} \right]$$



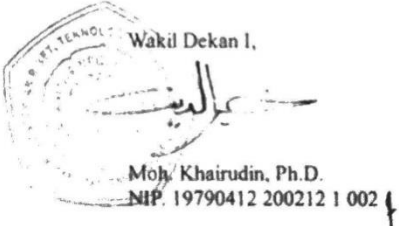
$$= \left[\frac{19}{19-1} \right] \left[1 - \frac{7,22}{44,89} \right]$$

$$= (1,0556)(0,8392)$$

$$= 0,885$$

Berdasarkan penghitungan *alpha cronbach* diatas diperoleh koefisien *alpha cronbach* sebesar **0.885**.

Lampiran 5 Surat izin penelitian UNY

	KEMENTERIAN RISET, TEKNOLOGI DAN PENDIDIKAN TINGGI UNIVERSITAS NEGERI YOGYAKARTA FAKULTAS TEKNIK Alamat Karangmalang, Yogyakarta 55281 Telp (0274) 568168 psw 276, 289, 292 (0274) 586734 Fax (0274) 586734 Website http://ft.uny.ac.id , email ft@uny.ac.id , teknik@uny.ac.id			
No	: 1727/H34/PL/2016	9 Nopember 2016		
Lamp	: -			
Hal	: Ijin Penelitian			
Yth.	1. Gubernur DIY c.q. Ka. Badan Kesatuan Bangsa dan Perlindungan Masyarakat (Kesbanglinmas) DIY 2. Bupati Kabupaten Klaten c.q. Kepala Badan Perencanaan dan Pembangunan Daerah (Bappeda) Kabupaten Klaten 3. Kepala Dinas Pendidikan, Pemuda, dan Olahraga Kabupaten Klaten 4. Kepala Sekolah SMK Negeri 4 Klaten			
Dalam rangka pelaksanaan Tugas Akhir Skripsi kami mohon dengan hormat bantuan Saudara memberikan ijin untuk melaksanakan penelitian dengan judul Pengembangan Sistem Pelaporan Absensi Siswa Berbasis SMS Gateway untuk Pengawasan Orang Tua di SMK N 4 Klaten, bagi Mahasiswa Fakultas Teknik Universitas Negeri Yogyakarta tersebut di bawah ini:				
No	Nama	No. Mhs.	Program Studi	Lokasi
1.	Budi Cahyono	11520241045	Pend. Teknik Informatika	SMK Negeri 4 Klaten
Dosen Pembimbing/Dosen Pengampu Nama : Dr. Eko Marpanaji, MT NIP : 19670608 199303 1 001				
Adapun pelaksanaan penelitian dilakukan mulai Bulan November 2016 s/d Desember 2016 Demikian permohonan ini, atas bantuan dan kerjasama yang baik selama ini, kami mengucapkan terima kasih.				
 Wakil Dekan I, Moh. Khairudin, Ph.D. NIP. 19790412 200212 1 002				
Tembusan : Ketua Jurusan				

Lampiran 6 Surat izin penelitian BAPPEDA



PEMERINTAH KABUPATEN KLATEN
BADAN PERENCANAAN PEMBANGUNAN DAERAH
(BAPPEDA)

Jl. Pemuda No. 294 Gedung Pemda II Lt. 2 Telp. (0272)321046 Psw 314-318 Faks 328730
KLATEN 57424

Nomor : 072/931/XI/09
Lampiran : -
Penhal : Ijin Penelitian

Klaten, 10 Nopember 2016
Kepada Yth
Ka SMKN 4 Klaten
Di

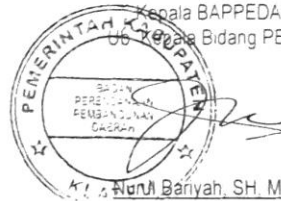
KLATEN

Menunjuk Surat dari Dekan Fak Teknik UNY Nomor 1727/H34/PL/2016 Tanggal 9 Nopember 2016 Perihal Permohonan Ijin Penelitian dengan hormat kami beritahukan bahwa di Instansi/Nilayah yang Saudara pimpin akan dilaksanakan Penelitian oleh :

Nama : Budi Cahyono
Alamat : Karangmalang, Yogyakarta
Pekerjaan : Mahasiswa UNY
Penanggungjawab : Moh Khairudin, Ph D
Judul/Topik : Pengembangan Sistem Pelaporan Absensi Siswa Berbasis SMS Gateway Untuk Pengawasan Orang Tua di SMKN 4 Klaten
Jangka Waktu : 3 Bln (10 Nopember 2016 s/d 10 Februan 2017)
Catatan : Menyerahkan Hasil Penelitian Berupa **Hard Copy** dan **Soft Copy** Ke Bidang PEPP BAPPEDA Kabupaten Klaten

Demikian atas kerjasama yang baik selama ini kami ucapkan terima kasih

An BUPATI KLATEN
Kepala BAPPEDA
Bidang PEPP



Nunul Bariyah, SH, M.Si
Pembina
NIP 195910271987032003

Tembusan disampaikan Kepada Yth :

- 1 Ka Kantor Kesbangpol Kab. Klaten
- 2 Ka Dinas Pendidikan Kab. Klaten
- 3 Dekan Fak. Teknik UNY
- 4 Yang bersangkutan
- 5 Arsip

Lampiran 7 Instrumen *Usability*

INSTRUMEN PENGUJIAN ASPEK *USABILITY*
PENGEMBANGAN SISTEM PELAPORAN ABSENSI SISWA BERBASIS SMS GATEWAY
UNTUK PENGAWASAN ORANG TUA DI SMK N 4 KLATEN

Tanda Tangan



Nama : Rahmawan A Aziz

Profesi : Guru

Berilah tanda centang (✓) pada kolom skala yang disediakan sesuai dengan penilaian untuk:
 pengujian *usability* pada "PENGEMBANGAN SISTEM PELAPORAN ABSENSI SISWA
 BERBASIS SMS GATEWAY UNTUK PENGAWASAN ORANG TUA DI SMK N 4
 KLATEN" yang disusun oleh Budi Cahyono, dengan keterangan sebagai berikut:

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

No	Pertanyaan	SS	S	N	KS	TS
1.	Secara keseluruhan, saya puas dengan kemudahan pemakaian aplikasi ini.		✓			
2.	Sangat sederhana penggunaan aplikasi ini.		✓			
3.	Saya dapat dengan sempurna menyelesaikan pekerjaan dengan aplikasi ini.		✓			
4.	Saya dapat menyelesaikan pekerjaan saya dengan cepat menggunakan aplikasi ini.		✓			
5.	Saya dapat menyelesaikan pekerjaan saya secara efisien menggunakan aplikasi ini.		✓			
6.	Saya merasa nyaman menggunakan aplikasi ini.		✓			
7.	Sangat mudah mempelajari penggunaan aplikasi ini.				✓	
8.	Saya yakin saya bisa menjadi produktif dengan cepat berkat aplikasi ini.		✓			
9.	Pesan kesalahan yang diberikan aplikasi ini menjelaskan dengan gamblang cara mengatasinya.			✓		
10.	Kapanpun saya membuat kesalahan, saya bisa memperbaikinya dengan cepat dan mudah.			✓		
11.	Informasi yang disediakan aplikasi ini cukup jelas.		✓			

Lampiran 8 Lanjutan instrumen *usability*

12.	Sangat mudah mencari informasi di aplikasi ini.		✓			
13.	Informasi yang disediakan aplikasi sangat mudah dipahami.		✓			
14.	Informasi yang disediakan efektif membantu saya menyelesaikan tugas dan skenario.		✓			
15.	Pengorganisasian informasi yang ditampilkan aplikasi jelas.		✓			
16.	Antarmuka aplikasi menyenangkan.		✓			
17.	Saya menyukai menggunakan antarmuka aplikasi ini.			✓		
18.	Aplikasi ini memiliki fungsi dan kapabilitas sesuai harapan saya.		✓			
19.	Secara keseluruhan, saya puas dengan aplikasi ini.			✓		

Saran: Sender ID ketika pesan masuk menampilkan
nama Lembaga bukan No. Pribadi

Lampiran 9 Daftar hadir

DAFTAR HADIR PESERTA
SIMULASI PENGEMBANGAN SISTEM PELAPORAN ABSENSI SISWA BERBASIS SMS
GATEWAY UNTUK PENGAWASAN ORANG TUA DI SMK N 4 KLATEN

No	Nama	Guru Mata Pelajaran	Tanda Tangan
1.	Rahadian Aziz	Simulasi Digital	1.
2.	Putri Herwinda	Bahasa Jawa	2.
3.	YUNI UTAROWATI	BK	3.
4.	LIDHARSI	BK	4.
5.	Wahid	PPKn	5.
6.	SUNGIRO	Adm Perkant	6.
7.	SURYA PR	Produktif AP	7.
8.	Ferry S	Produktif AP	8.
9.	Nurzulmi	Adm perkantoran	9.
10.	Lilik DP	Adm perkantoran	10.
11.	BATOT JP	PKn	11.
12.	Indriastri L	Bhs Inggris	12.
13.	Indah Nugrahani	Adm Perkantoran	13.
14.	Waga Ari C	AP	14.
15.	Danny Citra P	Kewirausahaan	15.
16.	Ninuk Flaryanti	Matematika	16.
17.	Ninuk Flaryanti	Matematika	17.
18.	BYAH ERGA N	Seni Budaya	18.
19.	WANDA M W	English / Jawa	19.
20.	Fajar Indro	Sejarah	20.
21.	Widodo	Matematika	21.
22.	Sri Astuti R.	B. Inggris	22.
23.			23.
24.			24.
25.			25.
26.			26.



PEMERINTAH KABUPATEN KLATEN
DINAS PENDIDIKAN
SEKOLAH MENENGAH KEJURUAN (SMK) NEGERI 4 KLATEN
Jl. Mataram No. 5, Belangwetan, Klaten Utara. Telp. (0272) 324084 Klaten
Email: smkn4klaten@ymail.com



SURAT KETERANGAN

Nomor : 423.1 / 635 / 13

Yang bertanda tangan di bawah ini Kepala SMK Negeri 4 Klaten menerangkan bahwa:

N a m a : Budi Cahyono
Alamat : Karang Malang, Yogyakarta
NIM : 11520241045
Program Studi : Pend. Teknik Informatika
Pekerjaan : Mahasiswa UNY
Penanggungjawab : Moh. Khairudin, Ph.D

bahwa yang namanya tersebut di atas pada bulan Nopember 2016 telah melaksanakan Penelitian dengan judul : **Pengembangan Sistem Pelaporan Absensi Siswa Berbasis SMS Gateway Untuk Pengawasan Orang Tua di SMK Negeri 4 Klaten Tahun Ajaran 2016/2017.**

Demikian Surat Keterangan ini dibuat kepada yang berkepentingan harap dipergunakan sebagaimana mestinya.

Klaten, 15 Nopember 2016

Kepala Sekolah



M. WORO NUGROHO, S.Pd, M.Eng
NIP. 19720316 199702 1 002



Lampiran 11 *Source code*

1. Model

Model berfungsi untuk menangani urusan yang berhubungan dengan interaksi ke *database*.

Admin_model.php

```

<?php if ( ! defined('BASEPATH')) exit('No direct script access allowed');
class Admin_model extends CI_Model {
    // =====
    =====
    public function set_data_siswa()
    {
        $data_siswa = array(
            'nis' => $this->input->post('nis'),
            'nama' => $this->input->post('nama'),
            'jenis_kelamin' => $this->input->post('jenis_kelamin'),
            'alamat' => $this->input->post('alamat'),
            'kelas' => $this->input->post('kelas1').$this->input-
>post('kelas2'),
            'id_jurusan' => $this->input->post('id_jurusan'),
            'ayah' => $this->input->post('ayah'),
            'pekerjaan_ayah' => $this->input->post('pekerjaan_ayah'),
            'ibu' => $this->input->post('ibu'),
            'pekerjaan_ibu' => $this->input->post('pekerjaan_ibu'),
            'kontak_orangtua' => $this->input-
>post('kontak_orangtua'),
            'created_at' => date('Y-m-d H:i:s'),
            'updated_at' => '0000-00-00 00:00:00'
        );
        return $this->db->insert('siswa', $data_siswa);
    }

    public function get_data_siswa($id_siswa = "")
    {
        if ($id_siswa == "")
        {
            $this->db->join('jurusan j', 'j.id_jurusan = s.id_jurusan');
            return $this->db->get('siswa s');
        }
        else
        {
            $this->db->where('s.id_siswa', $id_siswa);
            $this->db->join('jurusan j', 'j.id_jurusan = s.id_jurusan');
            return $this->db->get('siswa s');
        }
    }

    public function get_siswa($kelas, $id_jurusan)
    {
        $this->db->where('s.kelas', $kelas);
        $this->db->where('s.id_jurusan', $id_jurusan);
        // $this->db->where('p.tanggal !=', date('Y-m-d'));
        // $this->db->join('presensi p', 'p.id_siswa = s.id_siswa', 'left');
        return $this->db->get('siswa s');
    }

    public function get_kelas_siswa()

```

Siswa

```

    {
        $this->db->select('kelas');
        $this->db->distinct();
        return $this->db->get('siswa');
    }

function update_data_siswa($pk, $name, $value)
{
    $data[$name] = $value;
    $data['updated_at'] = date('Y-m-d H:i:s');

    $this->db->where('id_siswa', $pk);
    $this->db->update('siswa', $data);

    return TRUE;
}

public function delete_data_siswa($id_siswa)
{
    $this->db->where('id_siswa', $id_siswa);
    return $this->db->delete('siswa');
}

// ===== User
=====

public function set_data_guru()
{
    $data_guru = array(
        'nip' => $this->input->post('nip'),
        'nama' => $this->input->post('nama'),
        'jenis_kelamin' => $this->input->post('jenis_kelamin'),
        'alamat' => $this->input->post('alamat'),
        'kontak' => '+62' . $this->input->post('kontak'),
        'email' => $this->input->post('email'),
        'photo' => "",
        'created_at' => date('Y-m-d H:i:s'),
        'updated_at' => '0000-00-00 00:00:00'
    );
    $this->db->insert('guru', $data_guru);
    return $this->db->insert_id();
}

public function set_data_user($id_guru)
{
    $data_user = array(
        'username' => $this->input->post('email'),
        'password' => sha1($this->input->post('password')),
        'id_guru' => $id_guru,
        'created_at' => date('Y-m-d H:i:s'),
        'updated_at' => '0000-00-00 00:00:00'
    );
    return $this->db->insert('user', $data_user);
}

```

```

}

public function get_data_user($id_user = "")
{
    if ($id_user == "")
    {
        $this->db->not_like('nama', 'administrator');
        $this->db->join('guru g', 'g.id_guru = u.id_guru');
        return $this->db->get('user u');
    }
    else
    {
        $this->db->where('id_user', $id_user);
        $this->db->join('guru g', 'g.id_guru = u.id_guru');
        return $this->db->get('user u');
    }
}

public function get_password_user()
{
    $this->db->where('id_user', $this->session->userdata('id_login'));
    $this->db->select('password');
    return $this->db->get('user');
}

public function update_data_guru($pk, $name, $value)
{
    $data[$name] = $value;
    $data['updated_at'] = date('Y-m-d H:i:s');

    $this->db->where('id_guru', $pk);
    $this->db->update('guru', $data);

    return TRUE;
}

public function reset_password($id_user)
{
    $this->db->where('id_user', $id_user);
    $default_password = $this->db->get('user')->row()->username;
    $this->db->where('id_user', $id_user);
    return $this->db->update('user', array('password' => sha1($default_password)));
}

public function update_data_user($pk, $name, $value)
{
    $data[$name] = $value;
    $data['updated_at'] = date('Y-m-d H:i:s');
    $this->db->where('id_user', $pk);
    if ($name == 'username')
    {
        $this->db->update('user', $data);
        $this->db->where('id_user', $pk);
    }
}

```

```

        return $this->db->get('user')->row()->id_guru;
    }
    else
    {
        $this->db->update('user', $data);
        return TRUE;
    }
}

public function delete_data_guru($id_guru)
{
    $this->db->where('id_guru', $id_guru);
    return $this->db->delete('guru');
}

// ===== Jurusan
=====

public function set_data_jurusan()
{
    $data_jurusan = array(
        'nama_jurusan' => $this->input->post('nama_jurusan'),
        'created_at' => date('Y-m-d H:i:s'),
        'updated_at' => '0000-00-00 00:00:00'
    );
    return $this->db->insert('jurusan', $data_jurusan);
}

public function get_data_jurusan()
{
    return $this->db->get('jurusan');
}

public function update_data_jurusan($pk, $name, $value)
{
    $data[$name] = $value;
    $data['updated_at'] = date('Y-m-d H:i:s');

    $this->db->where('id_jurusan', $pk);
    $this->db->update('jurusan', $data);

    return TRUE;
}

public function delete_data_jurusan($id_jurusan)
{
    $this->db->where('id_jurusan', $id_jurusan);
    return $this->db->delete('jurusan');
}

// ===== Presensi
=====

```

```

public function set_data_presensi()
{
    $id_siswa = $this->input->post('id_siswa');
    $keterangan = $this->input->post('keterangan');
    $jam = $this->input->post('jam');

    for ($i=0; $i < count($id_siswa); $i++)
    {
        if ($keterangan[$i] != 'N')
        {
            $data_presensi = array(
                'id_siswa' => $id_siswa[$i],
                'tanggal' => date('Y-m-d'),
                'keterangan' => $keterangan[$i],
                'jam' => $jam[$i],
                'created_at' => date('Y-m-d H:i:s'),
                'created_by' => $this->session->userdata('id_login'),
                'updated_at' => '0000-00-00 00:00:00',
                'updated_by' => $this->session->userdata('id_login')
            );

            if ($this->db->insert('presensi', $data_presensi) === TRUE)
            {
                $nomor = $this->get_data_siswa($id_siswa[$i])->row()-
>kontak_orangtua;
                $nama = $this->get_data_siswa($id_siswa[$i])->row()-
>nama;
                $ayah = $this->get_data_siswa($id_siswa[$i])->row()-
>ayah;
                $dynamic_data = array('ayah' => $ayah, 'nama' =>
$nama, 'keterangan' => $keterangan[$i], 'jam' => $jam[$i]);
                $this->load->helper('file');
                $message = read_file(FCPATH.'sms.txt');
                foreach ($dynamic_data as $key_msg => $value_msg) {
                    switch ($value_msg) {
                        case 'I':
                            $value_msg = 'izin';
                            break;

                        case 'S':
                            $value_msg = 'sakit';
                            break;

                        case 'A':
                            $value_msg = 'alpha';
                            break;
                    }
                    $message = str_replace('{'.$key_msg.'}',
$value_msg, $message);
                }
                exec(FCPATH.'.bin\gammu-smsd-inject.exe -c
'.FCPATH.'.bin\smsdrc EMS '.$nomor.' -text "'.$message.'"');
            }
        }
    }
}

```

```

    }
    }
    return TRUE;
}

public function get_presensi($id_siswa)
{
    $this->db->where('id_siswa', $id_siswa);
    $this->db->where('tanggal', date('Y-m-d'));
    return $this->db->get('presensi');
}

public function get_data_presensi($kelas, $id_jurusan, $tanggal)
{
    $this->db->join('presensi p', 'p.id_siswa = s.id_siswa');
    $this->db->where('s.kelas', $kelas);
    $this->db->where('s.id_jurusan', $id_jurusan);
    $this->db->where('p.tanggal', $tanggal);
    return $this->db->get('siswa s');
}

public function update_data_presensi()
{
    $id_siswa = $this->input->post('id_siswa');
    $data_presensi = array(
        'keterangan' => $this->input->post('keterangan'),
        'jam' => $this->input->post('jam'),
        'updated_at' => date('Y-m-d H:i:s'),
        'updated_by' => $this->session->userdata('id_login')
    );

    $this->db->where('id_siswa', $id_siswa);
    return $this->db->update('presensi', $data_presensi);
}

public function delete_data_presensi($id_siswa)
{
    $this->db->where('id_siswa', $id_siswa);
    return $this->db->delete('presensi');
}
}

```

Main_model.php

```

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

class Main_model extends CI_Model {

    // ===== Main

```

```

=====

public function user_check()
{
    $username = $this->input->post('username');
    $password = sha1($this->input->post('password'));

    if ($this->config->item('default_password') == "") {
        $this->config->set_item('default_password', $username);
    }

    $user = $this->db->get_where('user', array('username' => $username,
'password' => $password));

    return $user;
}

public function get_user_identity($id_guru)
{
    $this->db->where('id_guru', $id_guru);
    $user = $this->db->get('guru');

    return $user;
}
}

```

8. Controller

Controller bertindak sebagai penghubung antara data dan *View*.

Admin.php

```

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

class Admin extends CI_Controller {

    public function __construct()
    {
        parent::__construct();
        $this->load->model('admin_model');
        if ($this->session->userdata('logged_in') != TRUE) {
            redirect('/main');
        }
    }

    // =====
}
=====

```

Index

```

public function index()
{
    $this->load->helper('file');
    $data['smk'] = read_file(FCPATH.'smk.txt');
    $data['num_siswa'] = $this->admin_model->get_data_siswa()-
>num_rows();
    $data['num_jurusan'] = $this->admin_model->get_data_jurusan()-
>num_rows();
    $data['num_user'] = $this->admin_model->get_data_user()-
>num_rows();
    $this->load->view('admin/index', $data);
}

public function logout()
{
    $this->session->sess_destroy();
    redirect('/main');
}

//          =====
=====
}

public function profil()
{
    $this->load->helper('file');
    $data['smk'] = read_file(FCPATH.'smk.txt');
    $data['user'] = $this->admin_model->get_data_user($this->session-
>userdata('id_login'))->row();
    $this->load->view('admin/profil', $data);
}

public function upload_photo()
{
    $id = $this->input->post('id_guru');

    unlink(FCPATH.'uploads/images/'. $this->input->post('namafile'));
    $config['upload_path'] = FCPATH.'uploads/images';
    $config['file_name'] = $this->input->post('nama').'_pp';

    $config['allowed_types'] = 'jpg|png|gif|jpeg';
    $config['remove_spaces'] = FALSE;
    $this->load->library('upload', $config);
    $field_name = "import";

    if ( ! $this->upload->do_upload($field_name))
    {
        $this->session->set_flashdata('message_error', 'An error
occured! '.$this->upload->display_errors());
    }
    else
    {
        $name = $this->upload->data('file_name');
        $this->admin_model->update_data_guru($id, 'photo', $name);
    }
}

```

```

        $this->session->set_flashdata('message_success', 'Photo Profil
        '.$this->input->post('jenis').' Uploaded!');
    }
    redirect('/profil');
}

public function change_password()
{
    $password = $this->admin_model->get_password_user()->row();
    $old_password = sha1($this->input->post('password'));
    $new_password = sha1($this->input->post('password_baru'));
    if ($old_password == $password->password)
    {
        $this->admin_model->update_data_user($this->session-
        >userdata('id_login'), 'password', $new_password);
        $this->session->set_flashdata('message_success', 'Password
        User Berhasil Diubah!');
        redirect('/profil');
    }
    else
    {
        $this->session->set_flashdata('message_error', 'Gagal, Password
        Lama Salah!');
        redirect('/profil');
    }
}

public function reset_password($id_user)
{
    if ($this->admin_model->reset_password($id_user) === TRUE)
    {
        $this->session->set_flashdata('message_success', 'Password
        User Berhasil Direset!');
        redirect('/user');
    }
    else
    {
        $this->session->set_flashdata('message_error', 'Gagal, Terjadi
        Kesalahan!');
        redirect('/user');
    }
}

// ===== Siswa
=====

public function siswa()
{
    $this->load->helper('file');
    $data['smk'] = read_file(FCPATH.'smk.txt');
    $data['jurusan'] = $this->admin_model->get_data_jurusan()->result();
    $data['siswa'] = $this->admin_model->get_data_siswa()->result();
    $this->load->view('admin/siswa', $data);
}

```

```

}

public function add_siswa()
{
    $this->load->helper('file');
    $data['smk'] = read_file(FCPATH.'smk.txt');
    $data['jurusan'] = $this->admin_model->get_data_jurusan()->result();
    $this->load->view('admin/add_siswa', $data);
}

public function get_siswa($kelas, $id_jurusan)
{
    echo          json_encode($this->admin_model->get_siswa($kelas,
$id_jurusan)->result_array());
}

public function set_data_siswa()
{
    if ($this->admin_model->set_data_siswa() === TRUE)
    {
        $this->session->set_flashdata('message_success', 'Data Siswa
Berhasil Dimasukkan!');
        redirect('/add_siswa');
    }
    else
    {
        $this->session->set_flashdata('message_error', 'Gagal, Terjadi
Kesalahan!');
        redirect('/add_siswa');
    }
}

public function update_data_siswa()
{
    $pk = $this->input->post('pk');
    $name = $this->input->post('name');
    $value = $this->input->post('value');

    $retval = $this->admin_model->update_data_siswa($pk, $name,
$value);
    echo $retval;
}

public function delete_data_siswa($id_siswa)
{
    if ($this->admin_model->delete_data_siswa($id_siswa) === TRUE)
    {
        $this->session->set_flashdata('message_success', 'Delete Data
Siswa Berhasil!');
        redirect('/siswa');
    }
    else
    {

```

```

        $this->session->set_flashdata('message_error', 'Gagal, Terjadi
Kesalahan!');
        redirect('/siswa');
    }
}

// ===== User
=====

public function user()
{
    $this->load->helper('file');
    $data['smk'] = read_file(FCPATH.'smk.txt');
    $data['user'] = $this->admin_model->get_data_user()->result();
    $this->load->view('admin/user', $data);
}

public function add_user()
{
    $this->load->helper('file');
    $data['smk'] = read_file(FCPATH.'smk.txt');
    $this->load->view('admin/add_user', $data);
}

public function set_data_user()
{
    $id_guru = $this->admin_model->set_data_guru();
    if ($this->admin_model->set_data_user($id_guru) === TRUE)
    {
        $this->session->set_flashdata('message_success', 'Data User
Berhasil Dimasukkan!');
        redirect('/add_user');
    }
    else
    {
        $this->session->set_flashdata('message_error', 'Gagal, Terjadi
Kesalahan!');
        redirect('/add_user');
    }
}

public function update_data_guru()
{
    $pk = $this->input->post('pk');
    $name = $this->input->post('name');
    $value = $this->input->post('value');

    $retval = $this->admin_model->update_data_guru($pk, $name,
$value);
    echo $retval;
}

public function update_data_user()

```

```

    {
        $pk = $this->input->post('pk');
        $name = $this->input->post('name');
        $value = $this->input->post('value');

        $retval = $this->admin_model->update_data_user($pk, $name, $value);
        echo $retval;
    }

    public function update_data_email()
    {
        $pk = $this->input->post('pk');
        $name = $this->input->post('name');
        $value = $this->input->post('value');

        $id_guru = $this->admin_model->update_data_user($pk, $name,
$value);
        $retval = $this->admin_model->update_data_guru($id_guru, 'email',
$value);
        echo $retval;
    }

    public function delete_data_guru($id_guru)
    {
        if ($this->admin_model->delete_data_guru($id_guru) === TRUE)
        {
            $this->session->set_flashdata('message_success', 'Delete Data
User Berhasil!');
            redirect('/user');
        }
        else
        {
            $this->session->set_flashdata('message_error', 'Gagal, Terjadi
Kesalahan!');
            redirect('/user');
        }
    }

    // ===== Jurusan
    =====

    public function jurusan()
    {
        $this->load->helper('file');
        $data['smk'] = read_file(FCPATH.'smk.txt');
        $data['sms'] = read_file(FCPATH.'sms.txt');
        $data['jurusan'] = $this->admin_model->get_data_jurusan()->result();
        $this->load->view('admin/jurusan', $data);
    }

    public function set_data_jurusan()
    {
        if ($this->admin_model->set_data_jurusan() === TRUE)

```

```

        {
            $this->session->set_flashdata('message_success', 'Data Jurusan
Berhasil Dimasukkan!');
            redirect('/jurusan');
        }
        else
        {
            $this->session->set_flashdata('message_error', 'Gagal, Terjadi
Kesalahan!');
            redirect('/jurusan');
        }
    }

    public function set_data_message()
    {
        $message = $this->input->post('message');
        $this->load->helper('file');
        if (write_file(FCPATH.'sms.txt', $message) === TRUE)
        {
            $this->session->set_flashdata('message_success', 'Setting Pesan
Berhasil Disimpan!');
            redirect('/jurusan');
        }
        else
        {
            $this->session->set_flashdata('message_error', 'Gagal, Terjadi
Kesalahan!');
            redirect('/jurusan');
        }
    }

    public function update_data_jurusan()
    {
        $pk = $this->input->post('pk');
        $name = $this->input->post('name');
        $value = $this->input->post('value');

        $retval = $this->admin_model->update_data_jurusan($pk, $name,
$value);
        echo $retval;
    }

    public function delete_data_jurusan($id_jurusan)
    {
        if ($this->admin_model->delete_data_jurusan($id_jurusan) === TRUE)
        {
            $this->session->set_flashdata('message_success', 'Delete data
jurusan Berhasil!');
            redirect('/jurusan');
        }
        else
        {
            $this->session->set_flashdata('message_error', 'Gagal, Terjadi

```

```

Kesalahan!);
        redirect('/jurusan');
    }
}

// ===== Presensi
=====

public function presensi()
{
    $this->load->helper('file');
    $data['smk'] = read_file(FCPATH.'smk.txt');
    $data['jurusan'] = $this->admin_model->get_data_jurusan()->result();
    $data['kelas'] = $this->admin_model->get_kelas_siswa()->result();
    $this->load->view('admin/presensi', $data);
}

public function add_presensi()
{
    $this->load->helper('file');
    $data['smk'] = read_file(FCPATH.'smk.txt');
    $data['jurusan'] = $this->admin_model->get_data_jurusan()->result();
    $data['kelas'] = $this->admin_model->get_kelas_siswa()->result();
    $this->load->view('admin/add_presensi', $data);
}

public function set_presensi()
{
    if ($this->admin_model->set_data_presensi() === TRUE)
    {
        $this->session->set_flashdata('message_success', 'Data Presensi
Berhasil Dimasukkan!');
        redirect('/add_presensi');
    }
    else
    {
        $this->session->set_flashdata('message_error', 'Gagal, Terjadi
Kesalahan!');
        redirect('/add_presensi');
    }
}

public function update_presensi()
{
    if ($this->admin_model->update_data_presensi() === TRUE)
    {
        $this->session->set_flashdata('message_success', 'Data Presensi
Berhasil Diubah!');
        redirect('/presensi');
    }
    else
    {
        $this->session->set_flashdata('message_error', 'Gagal, Terjadi

```

```

Kesalahan!);
        redirect('/presensi');
    }
}

public function delete_presensi($id_siswa)
{
    if ($this->admin_model->delete_data_presensi($id_siswa) === TRUE)
    {
        $this->session->set_flashdata('message_success', 'Data Presensi
Berhasil Dihapus!');
        redirect('/presensi');
    }
    else
    {
        $this->session->set_flashdata('message_error', 'Gagal, Terjadi
Kesalahan!');
        redirect('/presensi');
    }
}

public function get_presensi($id_siswa)
{
    echo    json_encode($this->admin_model->get_presensi($id_siswa)-
>num_rows());
}

public function get_data_presensi($kelas, $id_jurusan, $tanggal)
{
    echo    json_encode($this->admin_model->get_data_presensi($kelas,
$id_jurusan, $tanggal)->result_array());
}
}

```

Main.php

```

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

class Main extends CI_Controller {

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

    //      =====
    =====

```

Index

```

public function index()
{
    $this->load->helper('file');
    $data['smk'] = read_file(FCPATH.'smk.txt');
    $this->load->view('login', $data);
}

// ===== Login
=====

public function auth_user()
{
    $user = $this->main_model->user_check();

    if ($user->num_rows() > 0) {
        if ($this->input->post('password') == $this->config->
item('default_password')) {
            $this->session->set_userdata('default_password',
TRUE);
        }

        if ($user->row()->level == 'superadmin') {
            $id_guru = $user->row()->id_guru;
            $user_identity = $this->main_model->
get_user_identity($id_guru)->row_array();
            $this->session->set_userdata($user_identity);
            $this->session->set_userdata('id_login', $user->row()-
>id_user);
            $this->session->set_userdata('level', $user->row()-
>level);
            $this->session->set_userdata('logged_in', TRUE);

            redirect('/superadmin');
        } else {
            $id_guru = $user->row()->id_guru;
            $user_identity = $this->main_model->
get_user_identity($id_guru)->row_array();
            $this->session->set_userdata($user_identity);
            $this->session->set_userdata('id_login', $user->row()-
>id_user);
            $this->session->set_userdata('level', $user->row()-
>level);
            $this->session->set_userdata('logged_in', TRUE);
            redirect('/admin');
        }
    }
    else
    {
        $this->session->set_flashdata('message_error', 'Invalid
Username or Password');
        redirect('/main');
    }
}

```

}