

repository.ub.ac.id

**PEMBANGUNAN SISTEM SELEKSI PENERIMAAN SISWA  
BARU SMA BERBASIS WEB DENGAN PENGUJIAN KUALITAS  
BERBASIS ISO 25010**

(Studi Kasus SMA Negeri 1 Purwosari Kecamatan Purwosari  
Kabupaten Pasuruan)

**SKRIPSI**

Untuk memenuhi sebagian persyaratan  
memperoleh gelar Sarjana Komputer

Disusun oleh:  
Ovan Jaya Wardana  
NIM: 125150201111068



PROGRAM STUDI TEKNIK INFORMATIKA  
JURUSAN TEKNIK INFORMATIKA  
FAKULTAS ILMU KOMPUTER  
UNIVERSITAS BRAWIJAYA  
MALANG  
2017



## PENGESAHAN

PEMBANGUNAN SISTEM SELEKSI PENERIMAAN SISWA BARU SMA BERBASIS WEB  
DENGAN PENGUJIAN KUALITAS BERBASIS ISO 25010

SKRIPSI

Diajukan untuk memenuhi sebagian persyaratan  
memperoleh gelar Sarjana Komputer

Disusun Oleh:

Ovan Jaya Wardana

NIM: 125150201111068

Skripsi ini telah diuji dan dinyatakan lulus pada  
17 Januari 2017

Telah diperiksa dan disetujui oleh:

Dosen Pembimbing I

Dosen Pembimbing II

Fajar Pradana, S.ST, M.Eng  
NIP: 19871121 201504 1 004

Bayu Priyambadha, S.Kom, M.Kom  
NIP. 198209092008121004

Mengetahui  
Ketua Jurusan Teknik Informatika

Tri Astoto Kurniawan, S.T, M.T, Ph.D  
NIP: 19710518 200312 1 001

## PERNYATAAN ORISINALITAS

Saya menyatakan dengan sebenar-benarnya bahwa sepanjang pengetahuan saya, di dalam naskah skripsi ini tidak terdapat karya ilmiah yang pernah diajukan oleh orang lain untuk memperoleh gelar akademik di suatu perguruan tinggi, dan tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis disitasi dalam naskah ini dan disebutkan dalam daftar pustaka.

Apabila ternyata didalam naskah skripsi ini dapat dibuktikan terdapat unsur-unsur plagiasi, saya bersedia skripsi ini digugurkan dan gelar akademik yang telah saya peroleh (sarjana) dibatalkan, serta diproses sesuai dengan peraturan perundang-undangan yang berlaku (UU No. 20 Tahun 2003, Pasal 25 ayat 2 dan Pasal 70).

Malang, 10 Januari 2017



Ovan Jaya Wardana

NIM: 125150201111068

## KATA PENGANTAR

Puji syukur penulis panjatkan kepada Tuhan Yang Maha Esa, karena atas berkah, rahmat serta hidayah-Nya sehingga penulis dapat menyelesaikan laporan Skripsi dengan judul “Pembangunan Sistem Seleksi Penerimaan Siswa Baru SMA Berbasis Web Dengan Pengujian Kualitas Berbasis ISO 25010” dengan baik. Keberhasilan tersebut tak lepas dari bantuan dan dukungan dari berbagai pihak baik secara moril maupun materil. Oleh karena itu dalam kesempatan ini penulis ingin mengucapkan terima kasih kepada:

1. Bapak Fajar Pradana, S.ST, M.Eng selaku dosen pembimbing I yang telah memberikan ilmu dan saran selama penyusunan skripsi ini.
2. Bapak Bayu Priyambadha, S.Kom, M.Kom selaku dosen pembimbing II yang telah memberikan ilmu dan saran selama penyusunan skripsi ini.
3. Ibu Diah Priharsari, S.T, M.T selaku dosen dosen pembimbing II yang kemudian digantikan oleh Bapak Bayu Priyambadha, S.Kom, M.Kom karena beliau melanjutkan *study* ke luar negeri.
4. Bapak Imron Rosyadi selaku Guru SMA Negeri 1 Purwosari yang telah memberikan informasi mengenai seleksi penerimaan siswa SMA baru di SMA Negeri 1 Purwosari
5. Alm. Bapak Wardoyo dan Ibu Saprida, orang tua penulis dan seluruh keluarga atas dukungan dan do’a yang telah diberikan.
6. Faisol Andi Sefihara yang membantu mengajari saya *framework* CodeIgniter, membantu dalam proses memperoleh informasi mengenai seleksi penerimaan siswa SMA baru di SMA Negeri 1 Purwosari dan bersedia menjadi penguji pada subkarakteristik *functional suitability*.
7. Muklas Sutra Wirawan, Abdul Khoir, Nova Anwar Karim, dan Afrizal Fath Rahman yang bersedia menjadi penguji pada karakteristik *functional suitability*.
8. Maulidya Larasty S., Adit Ramadhan, dan Mohammad Fahmi Chasan yang telah memperbolehkan saya menggunakan handphonenya untuk pengujian pada karakteristik *portability*.
9. Seluruh pihak yang telah membantu kelancaran skripsi saya yang tidak dapat kami sebutkan satu persatu.

Penulis menyadari bahwa dalam penyusunan laporan ini masih banyak kekurangan baik format laporan maupun isinya. Oleh karena itu, kami mengharapkan saran dan kritik yang membangun dari para pembaca guna perbaikan laporan selanjutnya. Semoga laporan ini dapat memberikan manfaat bagi semua pihak, Aamiin.

Malang, 10 Januari 2017

Ovan Jaya Wardana  
ovan@ovanskyline.com

## ABSTRAK

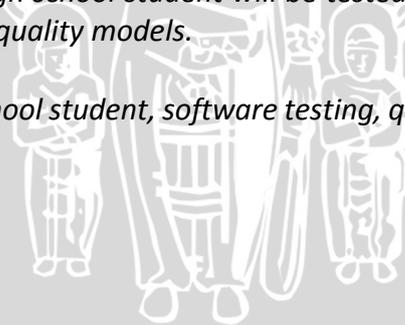
Pendidikan adalah salah satu bagian penting yang harus didapatkan oleh setiap orang untuk dapat menciptakan kemajuan peradaban dan kualitas hidup bangsa. Pemerintah juga telah menetapkan wajib belajar 12 tahun. Oleh karena itu, setiap anak yang sudah menempuh pendidikan di Sekolah Menengah Pertama, harus melanjutkan ke jenjang pendidikan yang lebih tinggi, yaitu Sekolah Menengah Atas. Untuk dapat memperoleh pendidikan di tingkat SMA, tahap yang harus dilalui adalah tahap seleksi penerimaan siswa baru. Beberapa SMA yang tergolong berkualitas, memiliki banyak peminat. Sehingga, banyak anak yang baru lulus SMP mendaftar ke SMA yang tergolong favorit tersebut. Dalam penyelenggaraan seleksi penerimaan siswa baru, terdapat beberapa masalah. Masalah tersebut, antara lain: kesulitan pihak sekolah pada proses pendaftaran calon siswa baru dalam hal pengadaan formulir pendaftaran, calon siswa harus mengantri cukup lama saat registrasi dengan calon siswa lain, sering terjadi contek mencontek saat berlangsungnya ujian tulis seleksi penerimaan siswa baru, dan sering terjadinya dorong mendorong ketika calon siswa ataupun orang tua calon siswa melihat pengumuman penerimaan hasil seleksi siswa baru yang ditempelkan di papan sekolah. Dari masalah tersebut, pada penelitian ini akan dibangun sistem seleksi penerimaan siswa SMA baru. Agar sistem dapat bekerja secara optimal, sistem yang dibuat harus memiliki kualitas yang baik. Kualitas sistem tersebut dapat diuji dengan berbagai metode/jenis pengujian perangkat lunak. Salah satu jenis pengujian perangkat lunak adalah pengujian black box dan white box. Namun, pengujian black box dan white box tidak memiliki karakteristik-karakteristik untuk standart produk. Maka dari itu, sistem yang akan dibangun, yaitu sistem seleksi penerimaan siswa SMA baru akan diuji berdasarkan model kualitas, yaitu model kualitas ISO 25010.

Kata kunci: penerimaan siswa SMA, pengujian perangkat lunak, model kualitas, ISO 25010

## ABSTRACT

*Education is one of the important parts that must be obtained by any person to be able to create the advancement of civilization and the quality of life of the nation. The government has also set a 12-year compulsory study. Therefore, every child who has been studying in junior high schools, should continue to pursue a higher education, namely High School. To be able to receive education at high school level, are stages to go through the selection process for new students. Some high school belonging to quality, has many takers. Thus, many children who just graduated from junior high school to enroll belonging to the favorite. In the selection of new high school students, there are some problems. These issues, among others: the difficulty of the school in the process of registration of new students in the procurement of the registration form, prospective students have to queue long enough during registration with the applicant another, often cheating cheating during the course of the written test the selection of new high school student, and frequent pushing and shoving when prospective students and parents of prospective students see the results of the selection new students which posted on the school board. Because of that problems, in research will build system for selection new high school student. For the system to work optimally, the system created must have good quality. The quality of the system can be tested with different methods/types of software testing. One type of software testing is black box testing and white box. However, black box testing and white box does not have the characteristics to standard products. Therefore, the system will be built, the system of selection new high school student will be tested based on the model of quality, namely ISO 25010 quality models.*

*Keyword: selection high school student, software testing, quality model, ISO 25010*



## DAFTAR ISI

PENGESAHAN .....	ii
PERNYATAAN ORISINALITAS .....	iii
KATA PENGANTAR.....	iv
ABSTRAK.....	v
ABSTRACT .....	vi
DAFTAR ISI .....	vii
DAFTAR TABEL.....	x
DAFTAR GAMBAR .....	xii
DAFTAR LAMPIRAN .....	xiv
<b>BAB 1 PENDAHULUAN.....</b>	<b>1</b>
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	3
1.3 Tujuan .....	3
1.4 Manfaat.....	3
1.5 Batasan masalah .....	3
1.6 Sistematika pembahasan .....	4
<b>BAB 2 LANDASAN KEPUSTAKAAN .....</b>	<b>5</b>
2.1 Kajian Pustaka .....	5
2.2 Penerimaan Siswa Baru SMA.....	5
2.3 Model <i>Waterfall</i> .....	8
2.4 <i>Unified Modelling Language (UML)</i> .....	9
2.5 Pola Arsitektur MVC ( <i>Model, View, Controller</i> ) .....	17
2.6 Pengujian Perangkat Lunak.....	17
2.6.1 Pengujian <i>White Box</i> .....	18
2.6.2 Pengujian <i>Black Box</i> .....	19
2.6.3 Pengujian Integrasi.....	20
2.7 ISO 25010 .....	20
2.7.1 <i>Functional Suitability</i> .....	21
2.7.2 <i>Performance Efficiency</i> .....	22
2.7.3 <i>Usability</i> .....	23
2.7.4 <i>Security</i> .....	27
2.7.5 <i>Portability</i> .....	28
<b>BAB 3 METODOLOGI .....</b>	<b>29</b>
3.1 Studi Literatur .....	30
3.2 Pengumpulan Data .....	32
3.3 Rekayasa Kebutuhan.....	32
3.4 Perancangan dan Implementasi .....	32
3.4.1 Perancangan.....	32
3.4.2 Implementasi .....	33
3.5 Pengujian dan Analisis .....	33
3.5.1 Pengujian Unit.....	33
3.5.2 Pengujian Integrasi.....	33

3.5.3 Pengujian Validasi .....	33
3.5.4 Pengujian Berbasis ISO 25010.....	33
3.6 Pengambilan Kesimpulan.....	35
<b>BAB 4 REKAYASA KEBUTUHAN.....</b>	<b>36</b>
4.1 Proses Rekayasa Kebutuhan .....	36
4.1.1 Elisitasi Kebutuhan .....	36
4.1.2 Spesifikasi Kebutuhan .....	41
4.1.3 Validasi .....	44
4.1.4 Manajemen Kebutuhan .....	45
4.2 Model Kebutuhan .....	48
4.2.1 Pemodelan Use Case Diagram .....	49
4.2.2 Pemodelan Diagram Sekuen.....	57
4.2.3 Pemodelan Diagram Kelas .....	63
<b>BAB 5 PERANCANGAN DAN IMPLEMENTASI .....</b>	<b>64</b>
5.1 Perancangan Sistem.....	64
5.1.1 Perancangan Arsitektur Sistem.....	64
5.1.2 Perancangan Diagram Kelas.....	65
5.1.3 Pemodelan Data.....	67
5.1.4 Perancangan Komponen.....	71
5.1.5 Perancangan Antarmuka.....	72
5.2 Implementasi .....	75
5.2.1 Spesifikasi Sistem .....	75
5.2.2 Batasan Implementasi.....	76
5.2.3 Implementasi Program.....	76
5.2.4 Implementasi Antarmuka .....	78
<b>BAB 6 PENGUJIAN DAN ANALISIS.....</b>	<b>84</b>
6.1 Pengujian .....	84
6.1.1 Pengujian Unit.....	85
6.1.2 Pengujian Integrasi.....	93
6.1.3 Pengujian Validasi .....	96
6.1.4 Pengujian Berbasis ISO 25010.....	99
6.2 Analisis .....	112
6.2.1 Analisis Hasil Uji Pengujian Unit.....	112
6.2.2 Analisis Hasil Uji Pengujian Integrasi .....	112
6.2.3 Analisis Hasil Pengujian Validasi .....	112
6.2.4 Analisis Hasil Uji Pengujian Berbasis ISO 25010.....	113
<b>BAB 7 PENUTUP .....</b>	<b>117</b>
7.1 Kesimpulan.....	117
7.2 Saran .....	118
<b>DAFTAR PUSTAKA.....</b>	<b>119</b>
<b>LAMPIRAN A FORM PENGUJIAN FUNCTIONAL SUITABILITY.....</b>	<b>122</b>
<b>LAMPIRAN B HASIL PENGUJIAN TIME BEHAVIOUR .....</b>	<b>124</b>
<b>LAMPIRAN C HASIL PENGUJIAN CAPACITY .....</b>	<b>139</b>
<b>LAMPIRAN D FORM PENGUJIAN USABILITY.....</b>	<b>150</b>



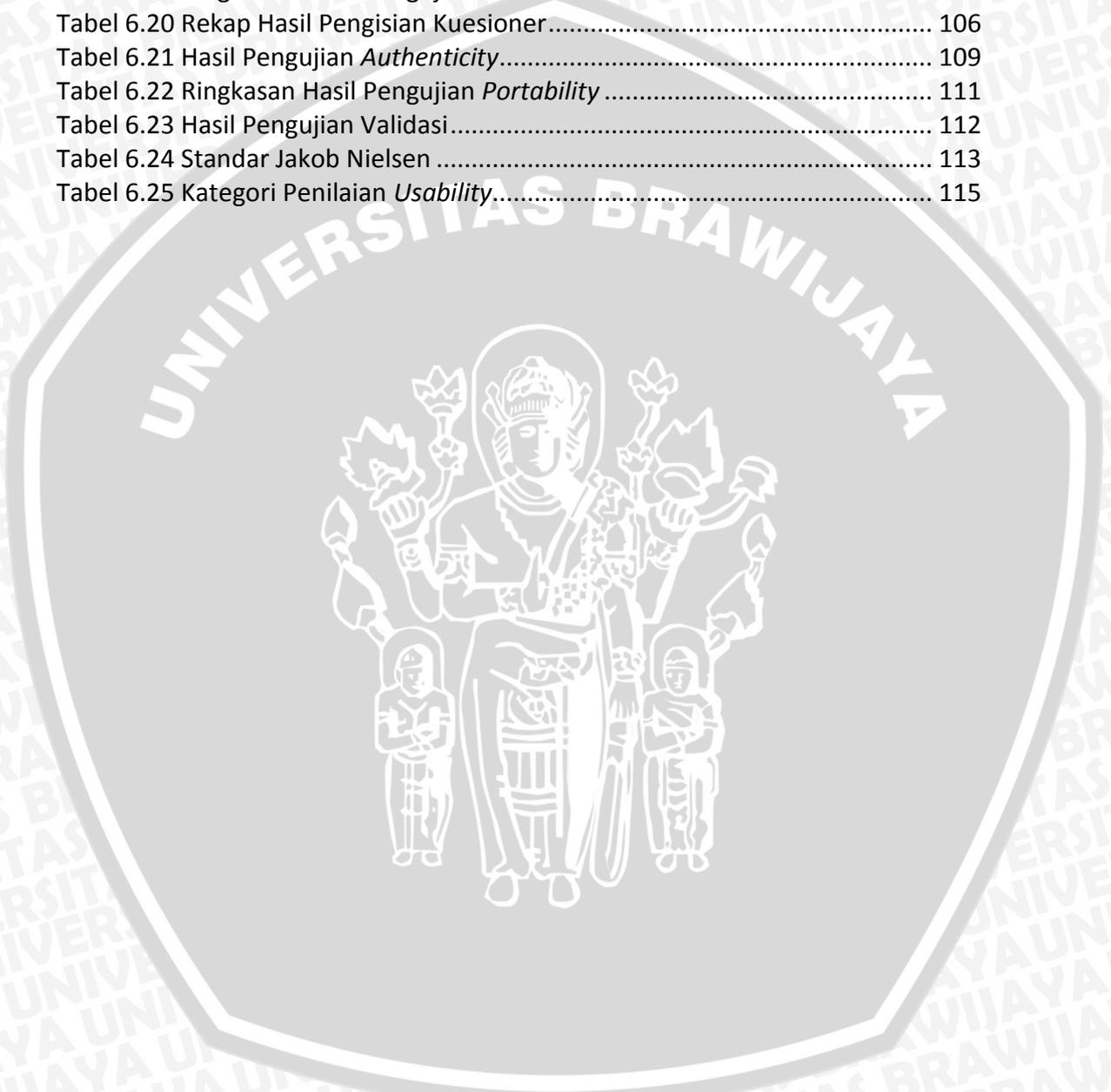
LAMPIRAN E HASIL PENGUJIAN PORTABILITY .....	152
LAMPIRAN F DOKUMENTASI.....	173



## DAFTAR TABEL

Tabel 2.1 Komponen <i>Use Case Diagram</i> .....	10
Tabel 2.2 Komponen <i>Activity Diagram</i> .....	12
Tabel 2.3 Komponen <i>Sequence Diagram</i> .....	13
Tabel 2.4 Komponen <i>Class Diagram</i> .....	15
Tabel 2.5 Standar Jakob Nielsen .....	22
Tabel 2.6 <i>USE Questionnaire</i> .....	24
Tabel 2.7 Kategori Penilaian <i>Usability</i> .....	26
Tabel 4.1 Identifikasi <i>Stakeholder</i> .....	37
Tabel 4.2 <i>Requirement Discovery</i> .....	37
Tabel 4.3 Spesifikasi Kebutuhan Fungsional .....	41
Tabel 4.4 Spesifikasi Kebutuhan Non-Fungsional .....	44
Tabel 4.5 Validasi Kebutuhan Fungsional .....	44
Tabel 4.6 Validasi Kebutuhan Non Fungsional.....	45
Tabel 4.7 Identifikasi User.....	46
Tabel 4.8 Identifikasi Kebutuhan .....	46
Tabel 4.9 Pemetaan <i>Use Case</i> dengan Kebutuhan Fungsional.....	49
Tabel 4.10 Skenario <i>Use Case</i> Registrasi.....	51
Tabel 4.11 Skenario <i>Use Case</i> Kerjakan Soal.....	53
Tabel 4.12 Skenario <i>Use Case</i> Posting Pengumuman.....	54
Tabel 4.13 Skenario <i>Use Case</i> Edit Soal .....	55
Tabel 4.14 Skenario <i>Use Case</i> Menentukan Jumlah Siswa Diterima .....	56
Tabel 4.15 Pemetaan <i>Use Case</i> dengan Diagram Sekuen.....	57
Tabel 5.1 Atribut Data Calon Siswa.....	69
Tabel 5.2 Atribut Data Admin .....	70
Tabel 5.3 Atribut Data Staff TU .....	70
Tabel 5.4 Spesifikasi Perangkat Keras .....	75
Tabel 5.5 Spesifikasi Perangkat Lunak .....	76
Tabel 6.1 Kasus Uji Algoritma Penilaian.....	87
Tabel 6.2 Kasus Uji Algoritma Menentukan Kuota Per Gelombang Ujian.....	90
Tabel 6.3 Kasus Uji Algoritma Menentukan Jadwal.....	93
Tabel 6.4 Interaksi Method Tambah Pengumuman dan Melihat Pengumuman .	94
Tabel 6.5 Hasil Pengujian Integrasi Method Tambah Pengumuman dan Melihat Pengumuman .....	94
Tabel 6.6 Interaksi Method Validasi Pembayaran dan Lihat Pembayaran .....	95
Tabel 6.7 Hasil Pengujian Integrasi Method Validasi Pembayaran dan Lihat Pembayaran .....	95
Tabel 6.8 Pengujian Lihat Pengumuman.....	96
Tabel 6.9 Pengujian Lihat Profil.....	97
Tabel 6.10 Pengujian Edit Profil .....	97
Tabel 6.11 Pengujian Upload Bukti Pembayaran.....	97
Tabel 6.12 Pengujian Lihat Jadwal Ujian.....	97
Tabel 6.13 Pengujian Cetak Kartu Ujian.....	98

Tabel 6.14 Pengujian Buat Soal.....	98
Tabel 6.15 Pengujian Edit Pengumuman .....	98
Tabel 6.16 Pengujian Menentukan Jumlah Siswa Diterima.....	99
Tabel 6.17 Pengujian Validasi Pembayaran .....	99
Tabel 6.18 Rekap Hasil Pengujian <i>Functional Suitability</i> .....	100
Tabel 6.19 Ringkasan Hasil Pengujian <i>Time Behaviour</i> .....	102
Tabel 6.20 Rekap Hasil Pengisian Kuesioner.....	106
Tabel 6.21 Hasil Pengujian <i>Authenticity</i> .....	109
Tabel 6.22 Ringkasan Hasil Pengujian <i>Portability</i> .....	111
Tabel 6.23 Hasil Pengujian Validasi.....	112
Tabel 6.24 Standar Jakob Nielsen .....	113
Tabel 6.25 Kategori Penilaian <i>Usability</i> .....	115



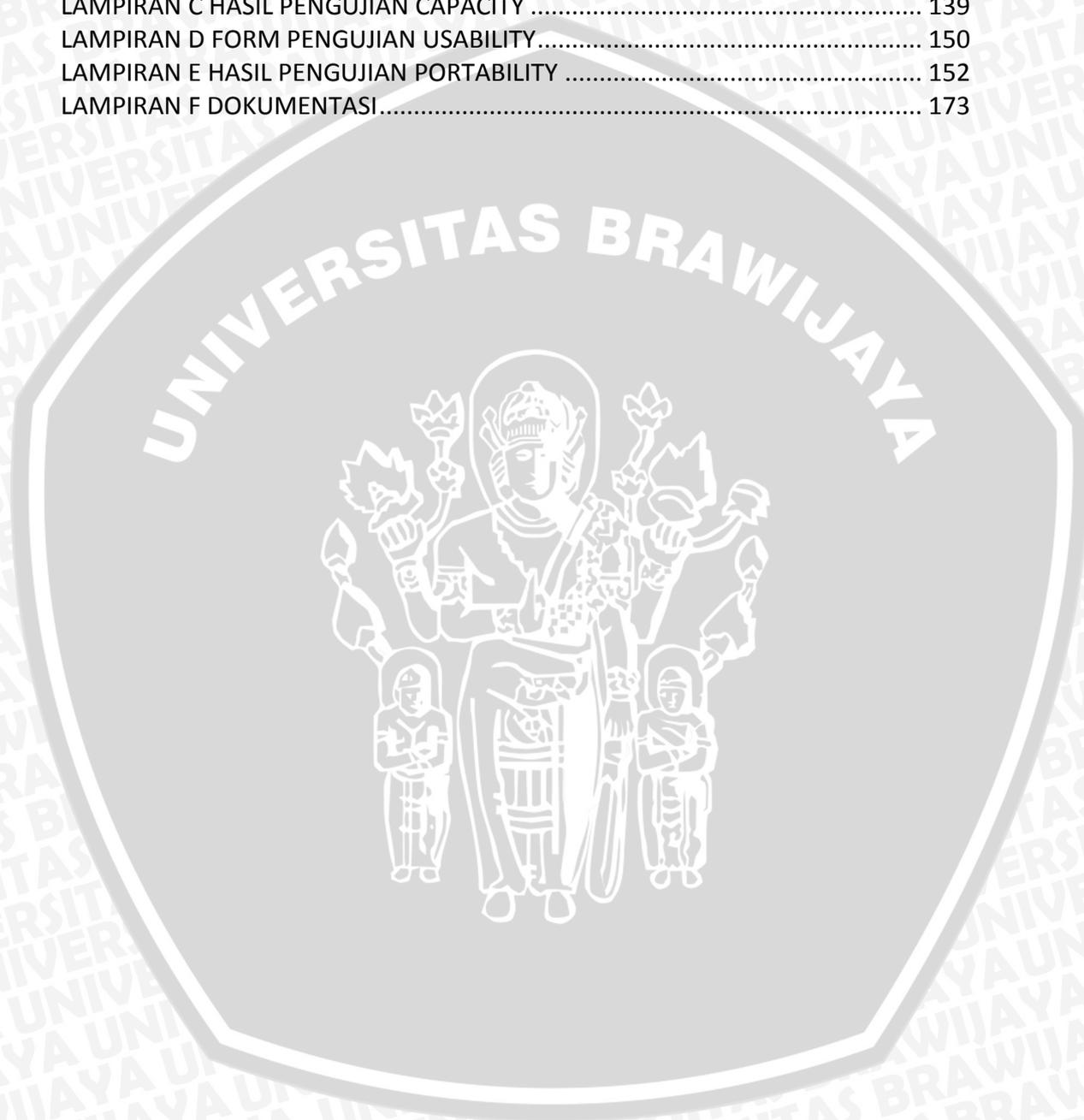
## DAFTAR GAMBAR

Gambar 2.1 Alur Proses Penerimaan Siswa Baru di SMA Negeri 1 Purwosari .....	7
Gambar 2.2 Ilustrasi Model <i>Waterfall</i> .....	8
Gambar 2.3 <i>Use Case Diagram</i> .....	11
Gambar 2.4 Contoh <i>Sequence Diagram</i> .....	14
Gambar 2.5 Contoh <i>Class Diagram</i> .....	16
Gambar 2.6 Pola Arsitektur MVC ( <i>Model, View, Controller</i> ) .....	17
Gambar 2.7 Contoh Notasi <i>Flow Graph</i> .....	18
Gambar 2.8 Karakteristik Model Kualitas ISO 25010 .....	21
Gambar 3.1 Diagram Metodologi Penelitian .....	29
Gambar 3.2 Diagram Alir Pengembangan Sistem Model Waterfall .....	31
Gambar 4.1 Diagram Rekayasa Kebutuhan .....	36
Gambar 4.2 Proses Bisnis Pembuatan Soal Ujian Seleksi .....	38
Gambar 4.3 Proses Bisnis Mengumumkan Tanggal Registrasi, Ujian Seleksi, dan Tanggal Pengumuman Hasil Seleksi .....	39
Gambar 4.4 Proses Bisnis Registrasi Calon Siswa .....	39
Gambar 4.5 Proses Bisnis Ujian Seleksi .....	40
Gambar 4.6 Proses Bisnis Mengumumkan Hasil Seleksi .....	40
Gambar 4.7 Identifikasi Aktor .....	46
Gambar 4.8 Diagram Use Case Sistem Seleksi Penerimaan Siswa Baru SMA .....	51
Gambar 4.9 Diagram Sekuen Posting Pengumuman .....	59
Gambar 4.10 Diagram Sekuen Edit Soal .....	60
Gambar 4.11 Menentukan Jumlah Siswa Diterima .....	61
Gambar 4.12 Analisis Diagram Kelas .....	63
Gambar 5.1 Arsitektur Sistem Seleksi Penerimaan Siswa Baru SMA .....	65
Gambar 5.2 Diagram Kelas Sistem Seleksi Penerimaan Siswa Baru SMA .....	66
Gambar 5.3 ERD Sistem Seleksi Penerimaan Siswa Baru SMA .....	68
Gambar 5.4 Pemodelan Database Sistem Seleksi Penerimaan Siswa Baru SMA .....	69
Gambar 5.5 Algoritma Kerjakan Ujian .....	71
Gambar 5.6 Algoritma Lihat Hasil Seleksi .....	72
Gambar 5.7 Algoritma Menentukan Jumlah Siswa Diterima .....	72
Gambar 5.8 Antarmuka Halaman Lihat Pengumuman .....	73
Gambar 5.9 Antarmuka Halaman Lihat Profil .....	73
Gambar 5.10 Antarmuka Halaman Login .....	74
Gambar 5.11 Antarmuka Halaman Registrasi .....	74
Gambar 5.12 Antarmuka Halaman Edit Soal .....	75
Gambar 5.13 Kode Program Mengerjakan Soal .....	77
Gambar 5.14 Kode Program Menentukan Jumlah Siswa Diterima .....	77
Gambar 5.15 Kode Program Melihat Hasil Seleksi .....	78
Gambar 5.16 Antarmuka Halaman Lihat Pengumuman .....	79
Gambar 5.17 Implementasi Antarmuka Registrasi .....	79
Gambar 5.18 Implementasi Antarmuka Lihat Profil .....	80
Gambar 5.19 Implementasi Antarmuka Kerjakan Soal .....	80

Gambar 5.20 Implementasi Antarmuka Lihat Hasil Seleksi .....	81
Gambar 5.21 Implementasi Antarmuka Lihar Seluruh Staff TU .....	81
Gambar 5.22 Implementasi Antarmuka Lihat Seluruh Calon Siswa .....	82
Gambar 5.23 Implementasi Antarmuka Validasi Pembayaran.....	82
Gambar 5.24 Implementasi Antarmuka Edit Pengumuman.....	83
Gambar 5.25 Implementasi Antarmuka Menentukan Jumlah Siswa Diterima ....	83
Gambar 6.1 Tahap-Tahap Pengujian Dan Analisis .....	84
Gambar 6.2 Pembentukan Node Algoritma Penilaian.....	85
Gambar 6.3 Flowgraph Algoritma Penilaian .....	86
Gambar 6.4 Pembentukan Node Algoritma Menentukan Kuota Per Gelombang Ujian .....	88
Gambar 6.5 Flowgraph Algoritma Menentukan Kuota Per Gelombang Ujian .....	89
Gambar 6.6 Pembentukan Node Algoritma Menentukan Jadwal .....	91
Gambar 6.7 Flowgraph Algoritma Menentukan Jadwal .....	92
Gambar 6.8 Tampilan Hasil Pengujian Integrasi Method Tambah Pengumuman dan Melihat Pengumuman.....	94
Gambar 6.9 Tampilan Hasil Pengujian Integrasi Method Validasi Pembayaran dan Lihat Pembayaran .....	96
Gambar 6.10 RAM Yang Digunakan Mozilla Tanpa Membuka Aplikasi Berbasis Web Apapun.....	104
Gambar 6.11 CPU Yang Digunakan Mozilla Tanpa Membuka Aplikasi Berbasis Web Apapun.....	104
Gambar 6.12 RAM Yang Digunakan Mozilla Dengan Hanya Membuka Sistem Seleksi.....	104
Gambar 6.13 CPU Yang Digunakan Mozilla Dengan Hanya Membuka Sistem Seleksi.....	105
Gambar 6.14 Ringkasan Hasil Pengujian <i>Capacity</i> .....	106
Gambar 6.15 Hasil Pengujian <i>Confidentiality, Integrity, Dan Non-Repudiation</i> .	108
Gambar 6.16 Hasil Pengujian <i>Accountability</i> .....	108
Gambar 6.17 Hasil Pengujian <i>Capacity</i> Dengan 131 <i>User</i> .....	114

## DAFTAR LAMPIRAN

LAMPIRAN A FORM PENGUJIAN FUNCTIONAL SUITABILITY.....	122
LAMPIRAN B HASIL PENGUJIAN TIME BEHAVIOUR .....	124
LAMPIRAN C HASIL PENGUJIAN CAPACITY .....	139
LAMPIRAN D FORM PENGUJIAN USABILITY.....	150
LAMPIRAN E HASIL PENGUJIAN PORTABILITY .....	152
LAMPIRAN F DOKUMENTASI.....	173



## BAB 1 PENDAHULUAN

### 1.1 Latar Belakang

Pendidikan adalah salah satu bagian penting yang harus didapatkan oleh setiap orang untuk dapat menciptakan kemajuan peradaban dan kualitas hidup bangsa (Lesmana, 2013). Pemerintah juga telah menetapkan wajib belajar 12 tahun. Oleh karena itu, setiap anak yang sudah menempuh pendidikan di Sekolah Menengah Pertama, harus melanjutkan ke jenjang pendidikan yang lebih tinggi, yaitu Sekolah Menengah Atas. Untuk dapat memperoleh pendidikan di tingkat SMA, tahap yang harus dilalui adalah tahap seleksi penerimaan siswa baru. Beberapa SMA yang tergolong berkualitas, memiliki banyak peminat (Suharsono dan Rahayu, 2014). Sehingga, banyak anak yang baru lulus SMP mendaftar ke SMA yang tergolong favorit tersebut.

Dalam penyelenggaraan seleksi penerimaan siswa baru, terdapat beberapa masalah. Masalah pertama adalah kesulitan pihak sekolah pada proses pendaftaran calon siswa baru dalam hal pengadaan formulir pendaftaran sehingga terjadinya penghamburan biaya, penumpukan arsip, dan sulit dalam proses pencarian calon siswa yang memenuhi kriteria (Suharsono dan Rahayu, 2014). Masalah kedua adalah calon siswa harus mengantri cukup lama saat registrasi dengan calon siswa lain. Masalah ketiga adalah sering terjadi contek mencontek saat berlangsungnya ujian tulis seleksi penerimaan siswa baru. Masalah keempat adalah calon siswa kesulitan untuk mendapatkan informasi pendaftaran dan penerimaan hasil seleksi siswa baru dikarenakan calon siswa harus datang langsung ke sekolah. Masalah kelima adalah sering terjadinya dorong mendorong ketika calon siswa ataupun orang tua calon siswa melihat pengumuman penerimaan hasil seleksi siswa baru yang ditempelkan di papan sekolah.

Untuk mengurangi masalah-masalah tersebut, maka dibutuhkan sebuah sistem dalam pelaksanaan penerimaan siswa baru (Suharsono dan Rahayu, 2014). Dimana sistem membantu pihak sekolah dalam pengadaan formulir sehingga biaya dapat diminimalisir. Sistem juga harus dapat membantu calon siswa dalam registrasi, sehingga calon siswa tidak perlu mengantri cukup lama. Sistem harus menyediakan fitur ujian online dimana soal dan pilihan jawaban diacak, sehingga dapat meminimalisir contek mencontek saat ujian seleksi berlangsung. Agar calon siswa mudah mendapatkan informasi mengenai pendaftaran dan penerimaan hasil seleksi siswa baru, sistem harus dapat menampilkan informasi mengenai hal tersebut.

Maka dari itu, solusi yang dapat dilakukan adalah dengan membuat sistem seleksi penerimaan siswa SMA baru. Sistem ini diharapkan mampu membantu proses seleksi mulai dari proses pendaftaran sampai proses pengumuman penerimaan siswa baru.

Agar sistem dapat bekerja secara optimal, sistem yang dibuat harus memiliki kualitas yang baik. Sebab, menurut Jamwal (2010) kualitas sistem yang

kurang baik dapat menyebabkan kerugian finansial dan tujuan dari pembuatan sistem tersebut gagal.

Kualitas sistem tersebut dapat diuji dengan berbagai metode/jenis pengujian perangkat lunak. Salah satu jenis pengujian perangkat lunak adalah pengujian black box dan white box. Namun, pengujian black box dan white box tidak memiliki karakteristik-karakteristik untuk standart produk (Jamwal, 2010). Jenis pengujian yang memiliki karakteristik-karakteristik untuk standart produk adalah pengujian berbasis model kualitas (Wahono, 2006).

Terdapat beberapa model kualitas, yaitu Mc Call, Boehm, Dromey, FURPS, ISO 9126, dan ISO 25010. McCall adalah model kualitas yang muncul pertama. Model ini dibentuk dengan beberapa fitur. Fitur ini dikelompokkan menjadi tiga perspektif, yaitu *maintenance*, *flexibility*, dan *testing*. McCall memiliki kelemahan, yaitu kurang tepat dalam pengukuran kualitas, karena didasarkan pada tanggapan ya atau tidak. Model McCall juga tidak mempertimbangkan fungsi, sehingga pandangan pengguna berkurang (Miguel, et all, 2014).

Kemudian model kualitas Boehm menetapkan karakteristik dalam skala besar yang merupakan perbaikan dari model McCall. Terdapat penambahan faktor pada model Boehm, yaitu *Utility*, *Maintainability*, dan *Portability*. *Utility* menunjukkan kemudahan, kehandalan dan efisiensi penggunaan produk perangkat lunak. *Maintainability* menggambarkan fasilitas untuk memodifikasi, dan *Portability* dalam arti mampu digunakan apabila terdapat perubahan lingkungan. Namun, terdapat kekurangan dari model Boehm, yaitu tidak adanya karakteristik *functionality*, *securtity*, dan *compability*(Miguel, et all, 2014).

Lalu Dromey yang didasarkan pada perspektif kualitas produk. Dengan cara ini evaluasi kualitas untuk setiap produk berbeda dan evaluasi yang lebih dinamis dapat ditetapkan. Model ini menyatakan bahwa untuk produk yang berkualitas baik, semua elemen yang membentuk itu harus dapat berjalan sesuai dengan fungsinya. Namun, tidak ada diskusi tentang bagaimana hal ini dapat dilakukan dalam prakteknya (Miguel, et all, 2014).

Model FURPS (*Functionality, Usability, Reliability, Performance, Product Support*) adalah model untuk mengklasifikasikan atribut kualitas perangkat lunak. Model ini mengkategorikan karakteristik sebagai Kebutuhan Fungsional dan Non Fungsional. Kebutuhan Fungsional didefinisikan oleh input dan output yang diharapkan atau *Functionality*(F). Sedangkan Kebutuhan Non Fungsional dikelompokkan sebagai *Usability* (U), *Reliability* (R), *Performance* (P), dan *Product Support* (S). Kelemahan dari Model FURPS yaitu beberapa fitur utama seperti portabilitas, diabaikan (Miguel, et all, 2014).

Model ISO 9126 didasarkan pada McCall dan Boehm model. Model ini memiliki dua bagian utama yang terdiri dari atribut kualitas internal dan eksternal ,dan kualitas dalam atribut digunakan (Miguel, et all, 2014). Kelemahan dari Model ISO 9126 yaitu tidak adanya karakteristik *security* dan *compatibility*.

Kemudian, muncullah Model ISO 25010 yang memperbarui model ISO 9126. Model ISO 25010 merupakan model yang memiliki karakteristik yang paling lengkap dibanding model-model yang lain. Model ISO 25010 merupakan model yang mengarahkan dalam pengembangan produk perangkat lunak dengan spesifikasi dan evaluasi persyaratan mutu (Miguel, et all, 2014).

Berdasarkan uraian sebelumnya, ISO 25010 dapat dipilih untuk pengujian kualitas dari sistem seleksi penerimaan siswa baru karena model ISO 25010 merupakan model kualitas memiliki karakteristik yang paling lengkap dibanding model-model yang lain, serta mengarahkan dalam pengembangan produk perangkat lunak dengan spesifikasi dan evaluasi persyaratan mutu.

## 1.2 Rumusan Masalah

Berdasarkan paparan latar belakang tersebut, maka rumusan masalah yang dikaji adalah sebagai berikut :

1. Apa kebutuhan sistem seleksi penerimaan siswa baru SMA?
2. Apa hasil perancangan sistem seleksi penerimaan siswa baru SMA?
3. Apa hasil implementasi sistem seleksi penerimaan siswa baru SMA?
4. Apa hasil pengujian dengan model kualitas ISO 25010 terhadap sistem yang dikembangkan?

## 1.3 Tujuan

Tujuan dari penelitian ini adalah :

1. Merancang dan mengimplementasikan sistem seleksi penerimaan siswa baru SMA.
2. Menguji sistem seleksi penerimaan siswa baru berbasis web.

## 1.4 Manfaat

Manfaat dari penelitian ini adalah :

1. Membantu pihak sekolah sehingga proses seleksi penerimaan siswa baru tidak membuang banyak biaya untuk pengadaan formulir dan soal serta lembar jawaban ujian. Hasil yang didapatkan pada proses ujian lebih valid, karena berkurangnya kegiatan mencontek yang dilakukan calon siswa.
2. Mempermudah calon siswa dalam proses registrasi, melihat informasi seputar seleksi, dan dalam melihat pengumuman hasil seleksi penerimaan siswa baru.

## 1.5 Batasan masalah

Agar permasalahan yang dirumuskan lebih terfokus, penelitian dalam skripsi ini dibatasi dalam hal :

1. Sistem yang dikembangkan berbasis web.
2. Karakteristik model kualitas ISO 25010 yang digunakan adalah *functional suitability, performance efficiency, usability, security, dan portability* yang disesuaikan dengan kebutuhan dari stakeholder.

## 1.6 Sistematika pembahasan

Sistematika penulisan berguna untuk memberikan gambaran dan uraian penulisan skripsi. Penulisan skripsi ini meliputi beberapa bab sebagai berikut :

**BAB I : PENDAHULUAN**

Pada bab ini berisi tentang latar belakang rumusan masalah, batasan masalah, tujuan , manfaat dan sistematika penulisan.

**BAB II : LANDASAN KEPUSTAKAAN**

Bab ini berisi tentang penjelasan teori – teori sebagai referensi dalam pembuatan sistem informasi.

**BAB III : METODOLOGI PENELITIAN**

Berisi metodologi penelitian pengerjaan skripsi yang mencakup studi literature, pengumpulan data, analisis dan perancangan, implementasi, pengujian, serta pengambilan kesimpulan.

**BAB IV : PERANCANGAN SISTEM**

Membahas tentang bagaimana rancangan Sistem Informasi Pelayanan Kesehatan Ibu Dan Anak Bidan Desa yang mencakup analisis kebutuhan perangkat lunak dan perancangan sistem.

**BAB V : IMPLEMENTASI PROGRAM**

Berisi implementasi yang dilakukan berdasarkan perancangan system. Bab ini berisi spesifikasi system, batasan implementasi, implementasi program dan implementasi antar muka

**BAB VI : PENGUJIAN DAN ANALISIS**

Menjelaskan proses pengujian sistem serta analisis yang dibuat berdasarkan pengujian.

**BAB VII : PENUTUP**

Berisi kesimpulan dari pembahasan bab – bab yang telah disampaikan dan saran terhadap desain aplikasi yang telah ditawarkan.

## BAB 2 LANDASAN KEPUSTAKAAN

### 2.1 Kajian Pustaka

Pada penelitian pertama membahas tentang pembuatan perangkat lunak pendaftaran siswa baru online telah dilakukan oleh Suharsono dan Rahayu. Perangkat lunak tersebut ditujukan kepada SMA Negeri 1 Baleendah. Suharsono dan Rahayu merancang sistem menggunakan pendekatan metode *waterfall* dengan metode pendekatan sistem berupa pendekatan analisis dan pemrograman terstruktur sebagai alat bantu proses. Hasil dari implementasi perangkat lunak ini adalah dapat mengelola calon siswa untuk melakukan pendaftaran secara terkomputerisasi, baik itu dalam kota maupun luar kota dan dapat mengurangi beberapa kendala yang masih menjadi gangguan dalam menyimpan data. Pengujian sistem menggunakan metode pengujian *black box* dan pengujian *betha* yaitu pengujian lapangan dengan pengolahan data sebagai nilai akurasi hasil pilihan.

Penelitian kedua dengan judul “Rancang Bangun Sistem Informasi Penerimaan Siswa Baru Online, Modul: Rekomendasi” telah dilakukan oleh Solikhah dan Ghozali. Metode untuk mengembangkan perangkat lunak menggunakan ReadySET. ReadySET merupakan sebuah standart pendokumentasian perangkat lunak yang disusun menggunakan metode Unified Process Model (UPM). Metode ini terdiri dari empat tahap yaitu Tahap Awal (Inception Phase), Tahap Perluasan (Elaboration Phase), Tahap Konstruksi (Construction Phase), dan Tahap Transisi (Transition Phase). Solikhah dan Ghozali menguji sistem menggunakan metode pengujian *black box*.

Penelitian ketiga dengan judul “Assessing the Quality of M-Learning Systems using ISO/IEC 25010” telah dilakukan oleh Acharya dan Sinha. Metode penelitian yang digunakan oleh Acharya dan Sinha adalah analisis teknologi M-Learning, terdiri dari *framework* M-Learning dan faktor kualitas yang mempengaruhi M-Learning. Kemudian analisis model kualitas ISO/IEC 25010. Hasil dari kedua analisis tersebut digunakan untuk menentukan karakteristik dan subkarakteristik yang relevan dengan M-Learning. Setelah itu mengidentifikasi metrik dan perangkat untuk pengukuran. Kemudian, menggunakan pengukuran tersebut untuk mengukur kualitas M-Learning.

### 2.2 Penerimaan Siswa Baru SMA

Penerimaan Siswa Baru (PSB) tingkat SMA adalah proses seleksi akademis calon siswa dengan tujuan untuk menuju jenjang pendidikan yang lebih tinggi dalam tingkatan SMA. (Sholikhah dan Ghozali, 2009). Setiap sekolah pasti mempunyai standarisasi dan kriteria sendiri dalam menyeleksi siswa baru guna menjaga kualitas dan kredibilitas sekolah. Pelaksanaan seleksi penerimaan siswa baru dapat dilakukan berbagai cara, mulai dari seleksi dari nilai UN, tes langsung, maupun secara online melalui internet. Namun seleksi penerimaan ini memiliki kelebihan dan kelemahan.

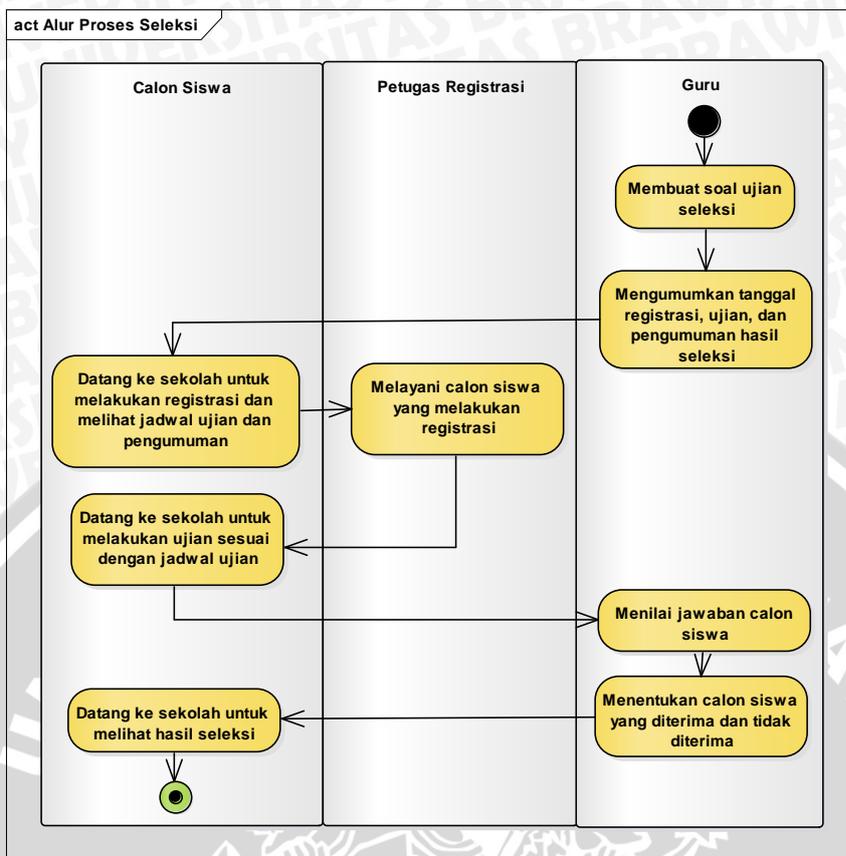
Sebagai kelebihan yaitu diantaranya dapat meningkatkan kepercayaan masyarakat terhadap mutu atau kualitas sekolah dengan adanya penjarangan siswa baru. Selain mendapat kepercayaan dari masyarakat, tentu saja standarisasi kualitas pendidikan di sekolah akan terjaga. Tidak jarang orang tua yang beranggapan ketika anaknya mengikuti seleksi penerimaan siswa di sekolah lanjutan, bahwa anaknya tengah bersaing dengan ratusan anak-anak lainnya, sehingga ketika ia lulus akan ada kebanggaan tersendiri bagi si anak maupun orang tua. Bagi pihak sekolah pun akan mempermudah pengklasifikasian siswa baru sesuai dengan kemampuan masing-masing.

Namun tetap saja ada kelemahan dari sistem seleksi tersebut, di mana ketika sekolah menjadikan seleksi penerimaan siswa baru sebagai formalitas, padahal bagi kerabat yang mempunyai kemampuan dibawah rata-rata dapat menduduki kursi dengan mudah. Tentu hal tersebut akan menimbulkan diskriminasi dan berkurangnya kesempatan bagi anak-anak yang murni mengikuti seleksi.

Dalam sistem seleksi melalui nilai UN juga terdapat kelemahan yaitu ketidakakuratan nilai UN yang diperoleh siswa. Karena bisa saja nilai tersebut diperoleh atas hasil tebak-tebakan atau malah mencontek dari siswa lain, padahal sebenarnya kemampuan siswa tersebut terbatas. Hal ini dapat menyebabkan siswa tidak mampu menerima pelajaran yang dirasa sulit baginya.

Penerimaan siswa baru di SMA Negeri 1 Purwosari terdiri dari 5 tahap. Tahap pertama adalah tahap pembuatan soal ujian seleksi. Tahap kedua adalah pengumuman tanggal registrasi, tanggal ujian seleksi, dan tanggal pengumuman hasil seleksi. Tahap ketiga adalah registrasi calon siswa. Tahap keempat adalah ujian seleksi. Tahap kelima adalah pengumuman hasil seleksi. Alur tahapan proses penerimaan siswa baru di SMA Negeri 1 Purwosari dapat dilihat pada gambar 2.1.





**Gambar 2.1 Alur Proses Penerimaan Siswa Baru di SMA Negeri 1 Purwosari**

Pada tahap pertama, guru-guru SMA Negeri 1 Purwosari membuat soal soal untuk ujian seleksi penerimaan siswa baru SMA. Soal soal tersebut adalah soal pilihan ganda. Soal soal tersebut meliputi bahasa indonesia, matematika, fisika, biologi, kimia, dan bahasa inggris.

Pada tahap kedua, pihak sekolah mengumumkan tanggal registrasi, tanggal ujian seleksi, dan tanggal pengumuman hasil seleksi. Calon siswa datang ke sekolah untuk melihat tanggal-tanggal tersebut.

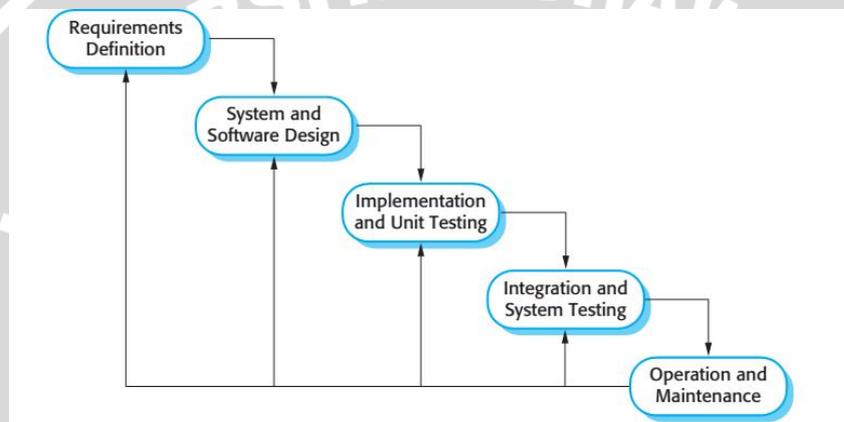
Pada tahap ketiga, seluruh calon siswa datang ke sekolah untuk melakukan registrasi. Calon siswa datang ke sekolah untuk mengambil formulir pendaftaran. Kemudian calon siswa mengisi formulir tersebut. Apabila calon siswa sudah mengisi formulir pendaftaran, formulir tersebut diberikan kepada petugas registrasi. Pada saat memberikan formulir, calon siswa juga membayar biaya registrasi.

Pada tahap keempat, seluruh calon siswa yang telah melakukan registrasi datang ke sekolah untuk mengikuti ujian seleksi. Calon siswa mengerjakan sejumlah soal pilihan ganda.

Pada tahap kelima, seluruh calon siswa datang ke sekolah untuk melihat pengumuman hasil seleksi, apakah calon siswa tersebut diterima atau tidak diterima di SMA Negeri 1 Purwosari sebagai siswa baru.

## 2.3 Model Waterfall

Model *Waterfall* merupakan salah satu model dari Software Development Life Cycle (SDLC). SDLC atau Software Development Life Cycle adalah proses mengubah atau mengembangkan suatu sistem perangkat lunak dengan menggunakan metodologi dan model-model yang digunakan orang dengan tujuan untuk mengembangkan sistem-sistem perangkat lunak sebelumnya (berdasarkan best practice atau cara-cara yang sudah teruji baik) (Rosa dan Shalahuddin, 2014). Sedangkan, Model *Waterfall* adalah sebuah model proses pengembangan perangkat lunak yang melibatkan tahap diskrit: spesifikasi, desain, implementasi, pengujian, dan pemeliharaan. Pada prinsipnya, tahap satu harus lengkap sebelum maju ke tahap berikutnya adalah mungkin. Dalam prakteknya, ada iterasi yang signifikan antara tahap (Sommerville, 2011). Gambar 2.2 mengilustrasikan tahapan pada model *waterfall*.



**Gambar 2.2 Ilustrasi Model Waterfall**

Sumber: Sommerville (2011)

Menurut Sommerville (2011) tahapan-tahapan dari model waterfall adalah sebagai berikut:

1. Analisis kebutuhan  
Sistem pelayanan, kendala, dan tujuan ditetapkan melalui konsultasi dengan pengguna sistem. Mereka kemudian didefinisikan secara rinci dan berfungsi sebagai spesifikasi sistem. Tahapan analisis kebutuhan terdiri dari tahap elisitasi, spesifikasi, validasi dan verifikasi serta manajemen kebutuhan.
2. Sistem dan perancangan perangkat lunak  
Proses desain sistem mengalokasikan persyaratan baik perangkat keras atau perangkat lunak sistem dengan membentuk sistem secara keseluruhan arsitektur. Software desain melibatkan identifikasi dan menggambarkan abstraksi sistem perangkat lunak yang mendasar dan hubungan mereka. Perancangan terdiri dari perancangan arsitektur, perancangan data, perancangan komponen dan perancangan antarmuka.
3. Implementasi dan unit testing  
Selama tahap ini, desain perangkat lunak direalisasikan sebagai serangkaian program atau unit program. Unit pengujian melibatkan memverifikasi bahwa setiap unit memenuhi spesifikasinya.

4. Integrasi dan pengujian sistem  
Unit program individu atau program diintegrasikan dan diuji sebagai sistem yang lengkap untuk memastikan bahwa perangkat lunak persyaratan telah dipenuhi. Setelah pengujian, sistem perangkat lunak yang dikirimkan ke pelanggan.
5. Operasi dan pemeliharaan  
Biasanya (meskipun tidak selalu), ini adalah fase terpanjang dalam siklus hidup. Sistem ini dipasang dan dimasukkan ke dalam penggunaan praktis. Pemeliharaan melibatkan kesalahan mengoreksi yang tidak ditemukan di awal tahap siklus hidup, meningkatkan pelaksanaan unit sistem dan meningkatkan pelayanan sistem sebagai requirementsare baru yang ditemukan.

Kelemahan yang ada pada model *Waterfall* adalah ketika adanya perubahan spesifikasi. Perubahan tersebut sangat sulit dilakukan karena tahapan pada proses pengembangan ini bersifat statis. (Sommerville, 2011)

Model *Waterfall* harus digunakan hanya ketika spesifikasi kebutuhan dipahami dengan baik dan tidak berubah secara radikal selama pengembangan sistem (Sommerville, 2011). Pada penelitian ini, sistem yang dikembangkan mempunyai spesifikasi kebutuhan yang tidak berubah, sehingga model SDLC yang digunakan adalah model *Waterfall*.

## 2.4 Unified Modelling Language (UML)

UML (*Unified Modelling Language*) adalah salah satu standar bahasa yang banyak digunakan di dunia industri. Tujuan penggunaan bahasa tersebut adalah untuk mendefinisikan kebutuhan (*requirement*), membuat analisis dan desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek (Rosa dan Shalahuddin, 2014).

Pembuatan UML dikontrol oleh suatu konsorsium terbuka yang terdiri dari banyak perusahaan yang disebut *Object Management Group* (OMG). OMG dibentuk untuk membuat standar-standar yang mendukung interoperabilitas, khususnya interoperabilitas sistem berorientasi objek (Flower, 2014). UML memiliki 13 jenis diagram yang berfungsi dalam memetakan bahasa pemrograman tertentu, diantaranya :

### 1. Use Case

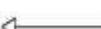
*Use case* adalah teknik persyaratan penemuan yang pertama kali diperkenalkan di Metode Objectory. *Use case* sekarang telah menjadi fitur mendasar dari bahasa pemodelan terpadu. Dalam bentuk yang paling sederhana, use case mengidentifikasi pelaku yang terlibat dalam interaksi dan nama jenis interaksi. Ini kemudian dilengkapi dengan informasi tambahan yang menggambarkan interaksi dengan sistem. Informasi tambahan mungkin deskripsi tekstual atau satu atau lebih model grafis seperti UML urutan atau state chart. (Sommerville, 2011).

Syarat penamaan pada use case adalah nama didefinisikan sesederhana mungkin dan dapat dipahami. Ada dua hal utama pada use case yaitu pendefinisian pada aktor dan use case.

- Aktor adalah orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dar aktor adalah gambar orang, tapi aktor belum tentu merupakan orang.
- Use case, yaitu fungsionalitas yang disediakan oleh sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor. (Rosa dan Shalahuddin, 2014).

Simbol-simbol yang ada pada diagram use case terdapat pada Tabel 2.1.

**Tabel 2.1 Komponen Use Case Diagram**

No	Gambar	Nama	Keterangan
1		<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i> .
2		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri( <i>independent</i> ) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri ( <i>independent</i> ).
3		<i>Generalization</i>	Hubungan dimana objek anak ( <i>descendent</i> ) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk ( <i>ancestor</i> ).
4		<i>Include</i>	Menspesifikasikan bahwa <i>use cases</i> sumber secara <i>eksplisit</i> .
5		<i>Extend</i>	Menspesifikasikan bahwa <i>use case</i> target memperluas perilaku dari <i>use case</i> sumber pada suatu titik yang diberikan.
6		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.
7		<i>System</i>	Menspesifikasikan paket yang menampilkan sistem secara terbatas.
8		<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor

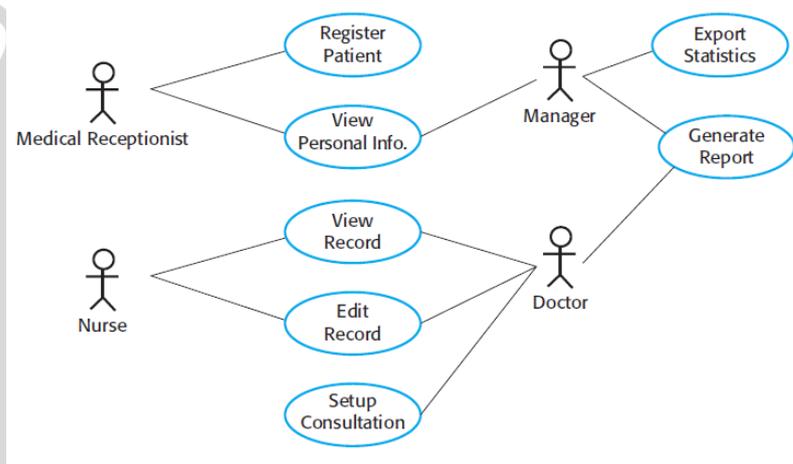
Tabel 2.1 Komponen Use Case Diagram (lanjutan)

No	Gambar	Nama	Keterangan
9		Collaboration	Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemen-elemennya (sinergi).
10		Note	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi

Sumber: Rosa dan Shalahuddin (2014).

Selain menggambarkan proses dalam bentuk diagram, use case diagram juga disertai dengan tabel yang menjelaskan rangkaian isi tiap use case. Dalam tabel tersebut dijelaskan beberapa hal, diantaranya adalah aktor mana yang menggunakan use case apa dan use case mana yang memasukkan use case lain.

Contoh use case diagram dapat dilihat pada Gambar 2.3 berikut :



Gambar 2.3 Use Case Diagram

Sumber: Sommerville (2011)

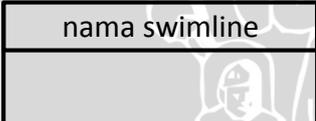
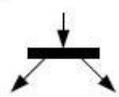
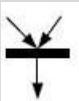
## 2. Activity Diagram

Activity diagram adalah teknik untuk menggambarkan logika prosedural, proses bisnis, dan jalur kerja (Fowler, 2004). Activity Diagram juga dapat menggambarkan proses paralel yang mungkin terjadi pada beberapa eksekusi. Activity Diagram merupakan state diagram khusus, di mana sebagian besar state adalah action dan sebagian besar transisi di-trigger oleh selesainya state sebelumnya (internal processing). Oleh karena itu activity Diagram tidak menggambarkan behaviour internal sebuah sistem (dan interaksi antar subsistem) secara eksak, tetapi lebih menggambarkan proses-proses dan jalur-jalur aktivitas dari level atas secara umum. Menggambarkan proses bisnis dan urutan aktivitas dalam sebuah proses. Dipakai pada business modeling untuk memperlihatkan urutan aktifitas proses bisnis. Struktur diagram ini mirip flowchart atau Data Flow Diagram pada perancangan

terstruktur. Sangat bermanfaat apabila kita membuat diagram ini terlebih dahulu dalam memodelkan sebuah proses untuk membantu memahami proses secara keseluruhan. *Activity Diagram* dibuat berdasarkan sebuah atau beberapa use case pada use case diagram.

Simbol-simbol yang ada pada *activity diagram* dapat dilihat pada Tabel 2.2

**Tabel 2.2 Komponen Activity Diagram**

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
percabangan/ <i>decision</i> 	asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu
penggabungan/ <i>join</i> 	asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu
status akhir 	status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir
swimlane 	memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi
fork 	digunakan untuk menunjukkan kegiatan yang dilakukan secara paralel
join 	digunakan untuk menunjukkan kegiatan yang digabungkan

Sumber: Rosa dan Salahuddin (2014)

### 3. Sequence Diagram

*Sequence diagram* menggambarkan kelakuan objek yang berada pada pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek (Rosa dan Salahuddin, 2014).

Diagram ini juga menunjukkan serangkaian pesan yang dipertukarkan oleh obyek-obyek yang melakukan suatu tugas atau aksi tertentu. Obyek-obyek tersebut kemudian diurutkan dari kiri ke kanan, aktor yang menginisiasi interaksi biasanya ditaruh di paling kiri dari diagram.

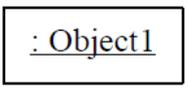
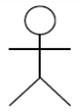
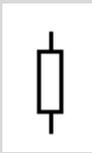
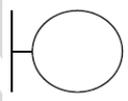


Pada diagram ini, dimensi vertikal merepresentasikan waktu. Bagian paling atas dari diagram menjadi titik awal dan waktu berjalan ke bawah sampai dengan bagian dasar dari diagram. Garis vertikal, disebut *lifeline*, dilekatkan pada setiap obyek atau aktor. Kemudian *lifeline* tersebut digambarkan menjadi kotak ketika obyek melakukan suatu operasi, kotak tersebut disebut activation. Obyek dikatakan mempunyai *live activation* pada saat tersebut.

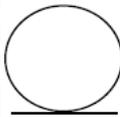
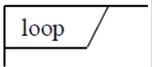
Pesan yang dipertukarkan antar obyek digambarkan sebagai sebuah anak panah antara activation box pengirim dan penerima. Kemudian di atasnya diberikan label pesan.

Komponen *sequence diagram* dapat dilihat pada Tabel 2.3.

**Tabel 2.3 Komponen Sequence Diagram**

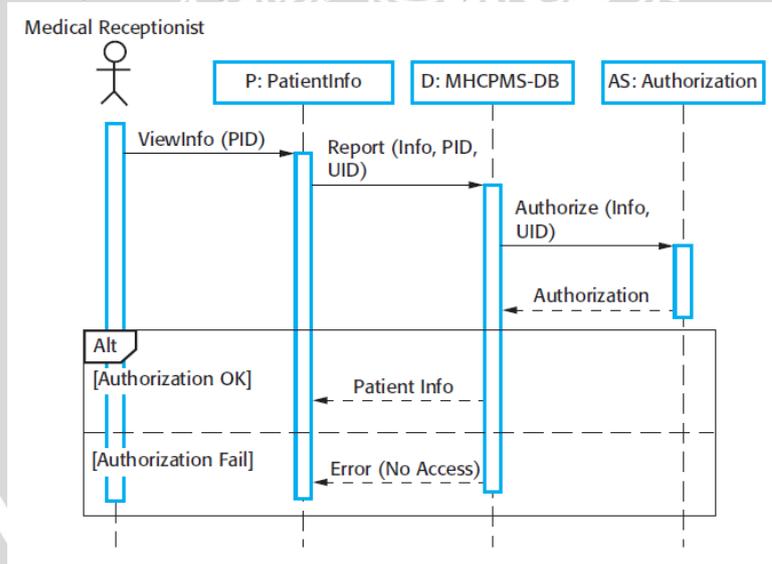
No	Komponen	Nama Komponen	Keterangan
1.		<i>Object</i>	Merupakan instance dari sebuah class dan dituliskan tersusun secara horizontal. Digambarkan sebagai sebuah class (kotak) dengan nama objek didalamnya yang diawali dengan sebuah titik koma.
2.		Actor	Actor juga dapat berkomunikasi dengan object, maka actor juga dapat diurutkan sebagai kolom.
3.		<i>Lifeline</i>	Mengindikasikan keberadaan sebuah object dalam basis waktu. Notasi untuk <i>Lifeline</i> adalah garis putus-putus vertikal yang ditarik dari sebuah objek.
4.		<i>Activation</i>	Dinotasikan sebagai sebuah kotak segi empat yang digambar pada sebuah lifeline. Activation mengindikasikan sebuah objek yang akan melakukan sebuah aksi.
5.		<i>Boundary</i>	Boundary terletak di antara sistem dengan dunia sekelilingnya. Semua form, laporan-laporan, antar muka ke perangkat keras seperti printer atau scanner dan antar muka ke sistem lainnya adalah termasuk dalam kategori.
6.		<i>Control</i>	Control berhubungan dengan fungsionalitas seperti pemanfaatan sumber daya, pemrosesan terdistribusi, atau penanganan kesalahan.

Tabel 2.3 Komponen *Sequence Diagram* (lanjutan)

No	Komponen	Nama Komponen	Keterangan
7.		<i>Entity</i>	Entity digunakan menangani informasi yang mungkin akan disimpan secara permanen. Entity bisa juga merupakan sebuah tabel pada struktur basis data.
8.		<i>Message</i>	Message, digambarkan dengan anak panah horizontal antara Activation. Message mengindikasikan komunikasi antara object-object.
9.		<i>Self-message</i>	Self-message atau panggilan mandiri mengindikasikan komunikasi kembali kedalam sebuah objek itu sendiri.
10.		<i>Loop</i>	Operator loop adalah <i>fragmen</i> yang dapat mengeksekusi berulang kali dan penjaga menunjukkan dasar iterasi.

Sumber: Ulum (2010)

Contoh *sequence diagram* dapat dilihat pada Gambar 2.4 berikut.



Gambar 2.4 Contoh *Sequence Diagram*

Sumber: Sommerville (2011)

#### 4. Class Diagram

*Class diagram* atau yang disebut dengan diagram kelas adalah diagram yang menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang nantinya akan dibuat untuk membangun sebuah sistem (Sukanto & Shalahuddin, 2013).

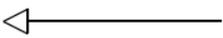
Class diagram dibuat agar *programmer* membuat kelas-kelas sesuai rancangan di dalam diagram, sehingga antara dokumentasi perancangan dan perangkat lunak sinkron. (Rosa dan Shalahuddin, 2014).

Penjelasan mengenai komponen *class diagram* dapat dilihat pada Tabel 2.4 berikut :

**Tabel 2.4 Komponen Class Diagram**

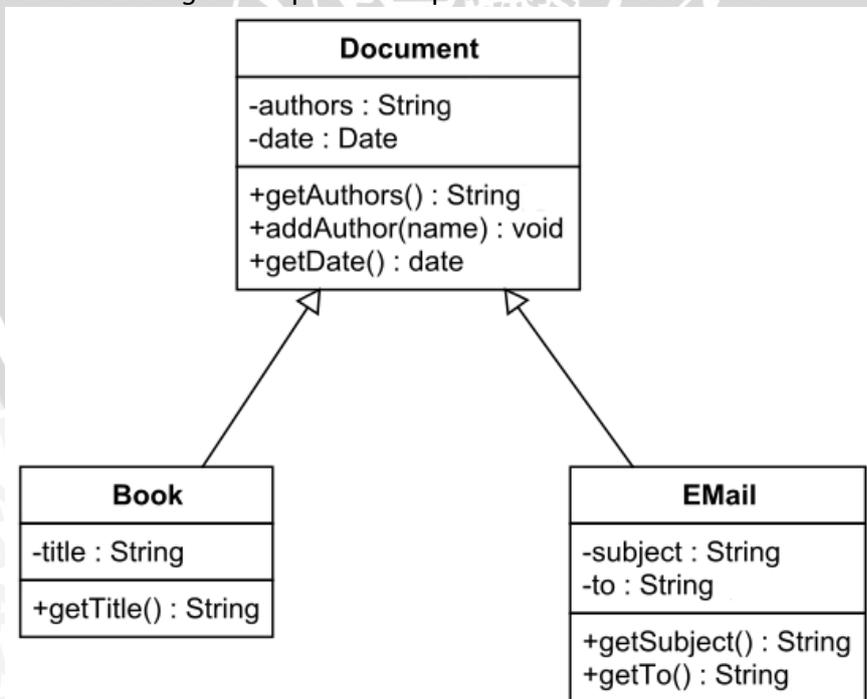
No	Komponen	Nama Komponen	Keterangan
1.	<div style="border: 1px solid black; padding: 5px; width: fit-content;"> <p>Nama Class</p> <p>- atribut</p> <p>- atribut</p> <p>- atribut</p> <hr/> <p>+ method</p> <p>+ method</p> </div>	Class	Merupakan blok – blok pembangun pada pemrograman berorientasi objek. Sebuah class digambarkan sebagai sebuah kotak yang terbagi atas 3 bagian. Bagian atas adalah bagian nama dari class. Bagian tengah mendefinisikan property/atribut class. Bagian akhir mendefinisikan method-method dari sebuah class.
2.	<u>1..n Owned by 1</u>	Assosiation	Asosiasi merupakan sebuah <i>relationship</i> paling umum antara 2 class, dan dilambangkan oleh sebuah garis yang menghubungkan antara 2 class.
3.		Composition	<i>Composition</i> merupakan kondisi dimana suatu kelas bergantung pada kelas yang lain. Sebuah <i>relationship composition</i> digambarkan sebagai garis dengan ujung berbentuk jajaran genjang berisi / <i>solid</i> .
4.		Dependency	<i>Dependency</i> merupakan kondisi dimana suatu kelas menggunakan kelas lain. Penggunaan <i>dependency</i> digunakan untuk menunjukkan operasi pada suatu <i>class</i> yang menggunakan <i>class</i> yang lain. <i>Dependency</i> dilambangkan sebagai sebuah panah bertitik-titik.

Tabel 2.4 Komponen *Class Diagram* (lanjutan)

No	Komponen	Nama Komponen	Keterangan
5.		<i>Agregation</i>	<i>Aggregation</i> mengindikasikan keseluruhan bagian relationship dan biasanya disebut sebagai relasi “memiliki sebuah” atau “bagian dari”. Sebuah <i>aggregation</i> digambarkan sebagai sebuah garis dengan sebuah jajaran genjang yang tidak berisi/tidak solid.
6.		<i>Generalizatio n</i>	Sebuah relasi <i>generalization</i> sepadan dengan sebuah relasi <i>inheritance</i> pada konsep berorientasi objek. Sebuah <i>generalization</i> dilambangkan dengan sebuah panah dengan kepala panah yang tidak solid yang mengarah ke kelas “parent”- nya/induknya.

Sumber: Ulum (2010)

Contoh *class diagram* dapat dilihat pada Gambar 2.5 berikut.

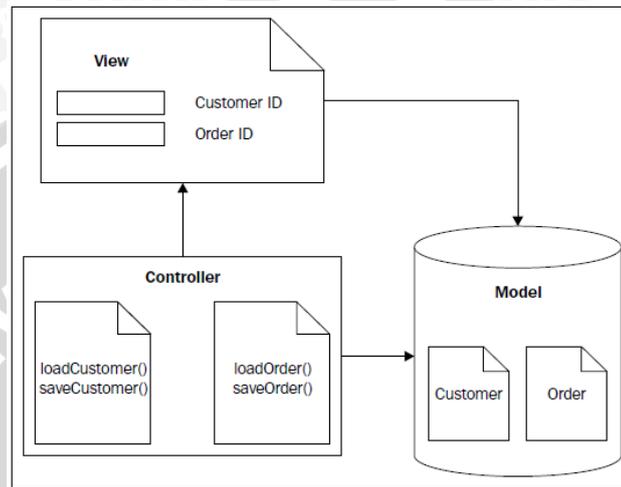


Gambar 2.5 Contoh *Class Diagram*

Sumber: cs.utsa.edu

## 2.5 Pola Arsitektur MVC (*Model, View, Controller*)

MVC (*Model, View, Controller*) adalah salah satu pola yang dikenal luas sebagai konsep yang memisahkan proses antara lapisan antarmuka, logika bisnis, dan data pada satu aplikasi (Abeyasinghe, 2009). MVC dapat mempermudah pekerjaan apabila terdapat perubahan, perbaikan, dan pengembangan dalam sebuah sistem (Abeyasinghe, 2009). Gambar 2.6 mempresentasikan pola arsitektur MVC.



**Gambar 2.6 Pola Arsitektur MVC (*Model, View, Controller*)**

Sumber: Abeyasinghe (2009)

Terdapat tiga komponen yang menunjukkan lapisan komponen dalam MVC, yaitu:

*Model*, merupakan komponen yang mengelola akses data yang terjadi pada aplikasi. *Model* bertugas terhadap respon jika *view* melakukan permintaan data dan respon jika *controller* melakukan perintah untuk perubahan data (Abeyasinghe, 2009).

*View*, merupakan komponen yang berfungsi untuk menampilkan data dari *model* ke dalam bentuk interaksi antarmuka. Dengan kata lain, *view* melakukan pengelolaan terhadap informasi yang ditampilkan (Abeyasinghe, 2009).

*Controller*, merupakan komponen yang bertugas untuk merespon dari perintah-perintah pengguna aplikasi seperti pemanggilan *view* atau perubahan/permintaan data yang melibatkan pemanggilan *model*. Dalam konteks nyata, *controller* berisi kode yang terkait dengan logika proses bisnis aplikasi (Abeyasinghe, 2009).

## 2.6 Pengujian Perangkat Lunak

Pengujian perangkat lunak adalah proses penting untuk menjamin kualitas dari sistem yang dibangun serta merepresentasikan spesifikasi, desain dan pengkodean perangkat lunak (Gunadarma, 2014 dalam Aulia, 2014). Pengujian ini memiliki tujuan untuk mencari kesalahan-kesalahan yang mungkin terjadi pada perangkat lunak dan dilakukan untuk mengatasi kegagalan yang akan mengakibatkan adanya pekerjaan tambahan dan biaya tambahan yang tidak

perlu. Pengujian perangkat lunak dilakukan berdasarkan prinsip-prinsip pengujian diantaranya (Wisnu, 2011 dalam Aulia, 2014):

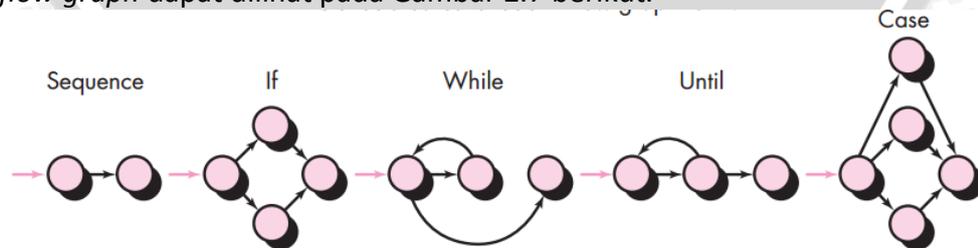
- Semua pengujian harus dapat diruntut sampai kepada spesifikasi kebutuhan perangkat lunak
- Pengujian harus dimulai dari lingkup yang besar
- Pengujian yang mendalam tidak mungkin dilakukan karena tidak mungkin mengeksekusi semua jalur permutasi
- Supaya efektif (memiliki probabilitas yang tinggi dalam menemukan kesalahan), pengujian dilakukan oleh pihak lain yang independen
- Pengujian harus direncanakan jauh sebelum dilakukan

### 2.6.1 Pengujian *White Box*

Pengujian *white box* atau disebut dengan *glass-box testing* adalah desain *test case* yang menggunakan struktur kontrol, dideskripsikan sebagai bagian dari desain komponen tingkat untuk menurunkan uji kasus (Pressman, 2010). Menggunakan metode pengujian *white box* dapat memperoleh uji kasus yang menjamin bahwa semua jalur independen dalam sebuah modul telah dieksekusi setidaknya sekali, melaksanakan semua keputusan logis pada sisi benar dan yang salah, melaksanakan semua perulangan dalam batas-batas operasional dan tes struktur data internal untuk memastikan validitasnya (Pressman, 2010).

Pengujian *white box* juga dapat dikatakan sebagai pengujian unit. Pengujian unit ini berfokus pada upaya verifikasi pada unit terkecil dari desain perangkat lunak, modul atau komponen perangkat lunak (Pressman, 2010). Uji unit berfokus pada logika proses internal dan struktur data dalam batas-batas komponen. Jenis pengujian dapat dilakukan secara paralel untuk beberapa komponen. Salah satu metode uji coba unit yaitu uji coba berbasis alur (*Basis Path Testing*). Uji coba berbasis alur merupakan teknik uji coba *white box* pertama yang diusulkan oleh Tom McCabe (Pressman, 2010). Langkah-langkah pengujian *basis path* menurut Pressman (2010) adalah sebagai berikut.

1. Mendefinisikan node pada kode program  
Hal ini dilakukan dengan menentukan kode program yang akan berdiri sebagai node-node yang akan digambarkan pada flow graph.
2. Membuat notasi *flow graph*  
Notasi *flow graph* mendefinisikan kontrol alur dari program. Contoh notasi *flow graph* dapat dilihat pada Gambar 2.7 berikut.



Gambar 2.7 Contoh Notasi *Flow Graph*

Sumber: Pressman (2010)

3. Mendefinisikan jalur independen

Jalur independen adalah setiap jalan melalui program yang memperkenalkan setidaknya satu set baru pernyataan. Contoh dari independent path seperti, Path 1 : 1-2-4-6-7; Path 2 : 1-2-3-4-5-6-7.

4. Menentukan *cyclomatic complexity*  $V(G)$

*Cyclomatic complexity* dapat dihitung menggunakan tiga cara yaitu:

$$V(G) = \text{jumlah region dari flow graph} \quad (2.1)$$

$$V(G) = E - N + 2 \quad (2.2)$$

$$V(G) = P + 1 \quad (2.3)$$

Keterangan:

E = edge atau garis

N = node atau simpul

P = predicate node

5. Membuat *test case*

*Test case* adalah merupakan tindakan yang dilakukan dalam kasus uji sesuai dengan jalur yang didefinisikan

### 2.6.2 Pengujian *Black Box*

Pengujian *black box*, juga disebut pengujian validasi, berfokus pada persyaratan fungsional perangkat lunak. Artinya, teknik pengujian *black box* memungkinkan untuk menurunkan set kondisi input yang sepenuhnya akan melaksanakan semua persyaratan fungsional untuk suatu program. Pengujian *black box* bukan merupakan alternatif untuk teknik *white box*. Sebaliknya, pengujian *black box* adalah pendekatan komplementer yang kemungkinan akan mengungkap kelas yang berbeda dari kesalahan daripada metode *white box*.

Menurut Wisnu dalam Aulia (2014) pengujian *black box* berusaha menemukan kesalahan dalam beberapa kategori, antara lain: (1) Kesalahan fungsi atau fungsi yang hilang, (2) kesalahan antarmuka, (3) kesalahan dalam struktur data atau akses database eksternal, (4) perilaku atau kinerja kesalahan, dan (5) inisialisasi dan kesalahan terminasi. Tidak seperti pengujian *white box*, yang dilakukan pada awal proses pengujian, pengujian *black box* cenderung diterapkan selama tahap akhir pengujian. Karena pengujian *black box* sengaja mengabaikan struktur kontrol, perhatian difokuskan pada informasi domain. Tes ini dirancang untuk menjawab pertanyaan-pertanyaan berikut:

- Bagaimana validitas fungsional diuji?
- Bagaimana perilaku dan kinerja sistem diuji?
- Apa kelas input akan membuat kasus uji yang baik?
- Apakah sistem sangat sensitif terhadap nilai input tertentu?
- Bagaimana batas kelas data diisolasi?
- Apa kecepatan data dan volume data yang dapat mentolerir sistem?
- Apa efek akan kombinasi spesifik data terhadap operasi sistem?

Dengan menerapkan teknik *black box*, dapat diperoleh satu set kasus uji yang memenuhi kriteria berikut: (1) uji kasus yang mengurangi, dengan jumlah yang lebih besar dari satu, jumlah kasus uji tambahan yang harus dirancang untuk

mencapai wajar pengujian, dan (2) uji kasus yang memberitahu sesuatu tentang ada atau tidak adanya kelas kesalahan, bukan kesalahan terkait hanya dengan tes khusus di tangan (Pressman, 2010).

### 2.6.3 Pengujian Integrasi

Pengujian integrasi adalah tahap dalam pengujian perangkat lunak di mana kelas atau komponen perangkat lunak digabungkan dan diuji sebagai sebuah satu kesatuan (Taiwar, 2013 dalam Tribowo, 2015). Pengujian ini bertujuan untuk memastikan hubungan antar kelas dalam sistem bekerja sesuai dengan spesifikasi kebutuhan perangkat lunak. Dalam pengujian integrasi, pengujian dapat dilakukan menggunakan teknik pengujian *Black Box* (Taiwar, 2013 dalam Tribowo, 2015).

Terdapat empat jenis pendekatan dalam pengujian integrasi antara lain: *Big-Bang*, *Top-Down*, *Bottom-Up*, dan *Sandwich*. Pendekatan *Big-Bang* dilakukan dengan menguji semua atau sebagian besar kelas yang sudah diimplementasikan secara bersamaan. Pendekatan ini sangat efektif untuk menghemat waktu dalam pengujian integrasi. Namun, apabila kasus uji dan hasilnya tidak didokumentasikan dengan baik maka proses integrasi keseluruhan akan lebih rumit. Sehingga pendekatan ini cocok untuk sistem yang tidak terlalu besar. Sedangkan pada pendekatan *Top-Down*, pengujian dilakukan pada kelas dengan level tertinggi terlebih dahulu kemudian bertahap turun ke kelas dengan level terendah. Kemudian pada pendekatan *Bottom-Up*, pengujian dimulai dari pengujian kelas secara individu dan bertahap naik kepada beberapa kelas. Pada pendekatan *Sandwich*, pengujian dilakukan berdasarkan kombinasi dari pendekatan *Top-Down* dan *Bottom-Up* (Taiwar, 2013 dalam Tribowo, 2015).

### 2.7 ISO 25010

ISO 25010 merupakan standar internasional yang digunakan untuk mengevaluasi kualitas sebuah produk perangkat lunak. Standar ini muncul pada tahun 2011 memperbarui ISO 9126 (Wagner, 2013).

Standar ini terdiri dari 8 karakteristik. Masing-masing karakteristik dibagi menjadi beberapa sub karakteristik. Karakteristik tersebut antara lain: *functional suitability*, *performance efficiency*, *compatibility*, *usability*, *reliability*, *security*, *maintability*, dan *portability*.

Pada penelitian ini, karakteristik yang digunakan untuk mengevaluasi Sistem Seleksi Penerimaan Siswa Baru SMA yaitu: *functional suitability*, *performance efficiency*, *usability*, *security*, dan *portability*. Kelima karakteristik tersebut dipilih berdasarkan kebutuhan dari *stakeholder*.



Gambar 2.8 Karakteristik Model Kualitas ISO 25010

Sumber: ISO/IEC 25010

### 2.7.1 Functional Suitability

Merupakan kemampuan produk perangkat lunak atau sistem untuk dapat menyediakan fitur-fitur yang dibutuhkan oleh pengguna (Alves, 2014).

Karakteristik ini terdiri dari 3 subkarakteristik, antara lain (ISO/IEC 25010) :

a. *Functional completeness.*

Merupakan kemampuan sistem untuk mencakup semua tugas yang ditentukan pengguna.

b. *Functional correctness*

Merupakan kemampuan suatu produk atau sistem memberikan hasil yang benar.

c. *Functional appropriateness*

Merupakan kemampuan sistem untuk memberikan fungsi yang dapat menyelesaikan tugas.

Ketiga subkarakteristik tersebut dalam sebuah sistem sebuah sistem dapat diketahui dengan Persamaan 2.1 (Acharya dan Sinha, 2013). Semua tugas yang ditentukan pengguna telah didapatkan dengan melakukan analisis kebutuhan, sehingga menghasilkan kebutuhan sistem. Kebutuhan sistem nantinya akan dijadikan pedoman dalam membuat fitur fitur sistem. Sistem akan memberikan hasil yang benar, apabila fitur yang dirancang, berhasil diimplementasikan. Serta, sistem dapat menyelesaikan tugas apabila fitur sistem berhasil diimplementasikan. Pada Persamaan 2.1, terdapat aturan, yaitu semakin mendekati angka 1, maka perangkat lunak tersebut semakin baik (Acharya dan Sinha, 2013).

$$X = \frac{I}{P} \tag{2.4}$$

Keterangan :

X = Tingkat *functional suitability*

I = Jumlah fitur yang dirancang

P = Jumlah fitur yang berhasil diimplementasikan

Contoh :



Fitur yang diimplementasi akan diuji oleh penguji. Penguji akan menghitung jumlah fitur yang berhasil diimplementasikan. Apabila jumlah penguji 10 orang, jumlah fitur yang dirancang berjumlah 40 dan masing-masing penguji mendapatkan nilai 40 dari hasil menguji, maka cara menghitungnya adalah sebagai berikut (Arifin, 2015).

$$I = 40 \tag{2.5}$$

$$P = \frac{\text{Jumlah dari hasil tiap penguji}}{\text{jumlah penguji}} = 40 \tag{2.6}$$

Maka didapatkan nilai X (*functional suitability*) yaitu 1. Sehingga perangkat lunak tersebut termasuk perangkat lunak yang sangat baik.

### 2.7.2 Performance Efficiency

Merupakan kemampuan produk perangkat lunak memberikan kinerja yang tepat sesuai dengan jumlah *resource* yang digunakan dalam kondisi yang telah ditetapkan (Alves, 2014). Karakteristik ini terdiri dari 3 subkarakteristik, antara lain (ISO/IEC 25010) :

a. *Time behaviour*

Merupakan respon dan processing time produk atau sistem, ketika menjalankan fungsinya.

Subkarakteristik ini dapat diuji dengan menggunakan *tools* WAPT dengan kriteria waktu respon (*response time*) (Setialana, 2014 dalam Tribowo, 2015). Hasil dari kriteria waktu repson tersebut akan dibandingkan dengan standart Jakob Nielsen yang dapat dilihat pada Tabel 2.5.

**Tabel 2.5 Standar Jakob Nielsen**

Waktu Respon	Pandangan Pengguna
<0,1 detik	Pengguna merasakan respon yang sangat cepat dari web
<1,0 detik	Pengguna merasakan jeda namun masih fokus kepada web tersebut
<10 detik	Perhatian pengguna terhadap web akan menurun tajam
≥ 10 detik	Kemungkinan besar pengguna akan beralih dari web tersebut

Sumber: Nielsen (1993) dalam Tribowo (2015)

b. *Resource utilization*

Merupakan jumlah dan jenis sumber daya yang digunakan oleh produk atau sistem ketika menjalankan fungsinya.

Untuk mengetahui jumlah sumber daya yang digunakan, dapat digunakan aplikasi Resource Monitor (Gordon, et all, 2014). Resource Monitor dapat mengetahui jumlah RAM dan CPU yang digunakan oleh sebuah aplikasi/sistem. Karena Sistem Seleksi Penerimaan Siswa Baru SMA dijalankan menggunakan browser, cara untuk mengetahui jumlah RAM dan CPU yang digunakan oleh Sistem Seleksi Penerimaan Siswa Baru SMA



adalah dengan cara membuka browser (Mozilla Firefox) tanpa membuka tab apapun, kemudian dilihat jumlah RAM dan CPU yang dipakai oleh Mozilla Firefox. Kemudian, Sistem Seleksi Penerimaan Siswa Baru SMA dibuka pada browser Mozilla tersebut dan dilihat jumlah RAM dan CPU yang dipakai. Selisih dari jumlah RAM sebelum dan sesudah membuka Sistem Seleksi pada Mozilla Firefox adalah jumlah RAM dan CPU yang dibutuhkan oleh Sistem Seleksi Penerimaan Siswa Baru SMA.

c. *Capacity*

Merupakan berapa banyak pengguna sistem yang dapat ditangani dan sistem masih bekerja dengan stabil.

Untuk mengetahui jumlah pengguna yang dapat ditangani oleh sistem dapat menggunakan Apache jMeter (Anicas, 2014).

### 2.7.3 Usability

Merupakan tingkat kualitas dari sistem yang mudah dipelajari, mudah digunakan dan mendorong pengguna untuk menggunakan sistem sebagai alat bantu dalam menyelesaikan tugas (Alves, 2014). Karakteristik ini terdiri dari 3 subkarakteristik, antara lain (ISO/IEC 25010) :

a. *Appropriateness recognizability*

Sejauh mana pengguna dapat mengenali apakah produk atau sistem yang sesuai untuk kebutuhan mereka.

b. *Learnability*

Sejauh mana produk atau sistem dapat dipelajari dengan mudah oleh pengguna tertentu untuk mencapai tujuan tertentu.

c. *Operability*

Merupakan kemampuan suatu produk atau sistem memiliki atribut yang membuatnya mudah dioperasikan dan dikontrol.

d. *User error protection*

Merupakan kemampuan sistem untuk melindungi pengguna sehingga pengguna tidak dapat membuat / melakukan kesalahan atau error.

e. *User interface aesthetics*

Sejauh mana *user interface* memungkinkan interaksi menyenangkan dan memuaskan bagi pengguna.

f. *Accessibility*

Sejauh mana suatu produk atau sistem dapat digunakan oleh orang-orang dengan karakteristik yang berbeda-beda.

Karakteristik ini dapat dievaluasi dengan berbagai macam model kuesioner. Berdasarkan penelitian yang dilakukan oleh Arifin (2015), karakteristik ini dapat diuji menggunakan *USE Questionnaire*. *USE Questionnaire* mempunyai pernyataan sebanyak 30 pernyataan. Semua pernyataan yang ada pada *USE*

Questionnaire telah mencakup semua subkarakteristik yang ada pada karakteristik Usability. Tabel 2.6 menunjukkan pernyataan dari USE Questionnaire.

**Tabel 2.6 USE Questionnaire**

Pernyataan		Subkarakteristik
<b>USEFULNESS</b>		
1.	Sistem ini membantu saya menjadi lebih efektif.	<i>Appropriateness recognizability</i>
2.	Sistem ini membantu saya menjadi lebih produktif.	<i>Appropriateness recognizability</i>
3.	Sistem ini bermanfaat	<i>Appropriateness recognizability</i>
4.	Ini memberi saya kontrol lebih besar terhadap kegiatan dalam hidup saya.	<i>Appropriateness recognizability</i>
5.	Sistem ini mempermudah pekerjaan saya	<i>Appropriateness recognizability</i>
6.	Sistem ini mempercepat pekerjaan saya	<i>Appropriateness recognizability</i>
7.	Sistem ini sesuai dengan kebutuhan dalam pekerjaan saya	<i>Appropriateness recognizability</i>
8.	Ia melakukan segala sesuatu yang saya harapkan untuk dilakukan.	<i>Appropriateness recognizability</i>
<b>EASE OF USE</b>		
9.	Sistem ini mudah digunakan.	<i>Learnability</i>
10.	Sistem ini mudah digunakan.	<i>Learnability</i>
11.	Sistem ini user friendly.	<i>Operability</i>
12.	Hal ini membutuhkan langkah-langkah yang mungkin paling sedikit untuk mencapai apa yang ingin saya lakukan dengan itu.	<i>Learnability</i>
13.	Fleksibel.	<i>Operability</i>
14.	Menggunakannya tanpa menggunakan usaha yang besar.	<i>Operability</i>
15.	Saya dapat menggunakannya tanpa instruksi tertulis.	<i>Learnability</i>
16.	Saya tidak melihat adanya inkonsistensi ketika saya menggunakannya.	<i>Operability</i>
17.	Pengguna akan menyukainya.	<i>User interface aesthetics</i>
18.	Saya dapat pulih dari kesalahan dengan cepat dan mudah.	<i>Operability</i>
19.	Saya dapat menggunakannya dengan sukses setiap kali.	<i>User error protection</i>
<b>EASE OF LEARNING</b>		
20.	Saya dapat dengan cepat terbiasa menggunakan sistem ini.	<i>Learnability</i>

Tabel 2.6 USE Questionnaire (lanjutan)

	Pernyataan	Subkarakteristik
21.	Saya mudah mengingat bagaimana menggunakannya.	<i>Learnability</i>
22.	Sangat mudah untuk belajar menggunakannya.	<i>Learnability</i>
23.	Saya cepat menjadi terampil dengan itu.	<i>Learnability</i>
<b>SATISFACTION</b>		
24.	Saya puas dengan itu.	<i>User interface aesthetics</i>
25.	Saya akan merekomendasikan hal ini kepada teman.	<i>User interface aesthetics</i>
26.	Sistem ini menyenangkan untuk digunakan.	<i>User interface aesthetics</i>
27.	Sistem ini bekerja sesuai dengan yang saya inginkan	<i>User interface aesthetics</i>
28.	Sistem ini bagus.	<i>User interface aesthetics</i>
29.	Saya rasa saya menginginkan sistem ini.	<i>User interface aesthetics</i>
30.	Sistem ini menyenangkan untuk digunakan.	<i>User interface aesthetics</i>

Sumber : Lund (2001)

*Usefulness* merupakan sejauh mana seseorang percaya bahwa menggunakan sistem tertentu akan meningkatkan atau kinerja pekerjaannya (Laitenberger & Dreyer, 2013). *Ease of Use* merupakan sejauh mana seseorang percaya bahwa menggunakan sistem tertentu akan dapat mengurangi usaha/upaya (Laitenberger & Dreyer, 2013). *Ease of Learning* merupakan seberapa cepat pengguna yang tidak pernah melihat antarmuka pengguna menyelesaikan tugas-tugas dasar (Motive Glosarry, 2010). *Satisfaction* merupakan seberapa suka pengguna dalam menggunakan sistem (Motive Glosarry, 2010).

USE *Questionnaire* diberikan kepada calon siswa, guru, dan staff tata usaha. Dengan demikian, subkarakteristik *accessibility* dapat diujikan pada sistem ini.

Skala Likert pada USE *Questionnaire* dapat menggunakan 7 skala atau 5 skala. Penelitian ini menggunakan skala 5. Skala 5 dipilih karena pada penelitian yang telah dilakukan oleh Muderedzwa & Nyakwende (2010) dalam Arifin (2015) membuktikan bahwa skala 5 adalah skala yang paling sering digunakan. Pada penelitian ini, jawaban yang disediakan berupa persetujuan terhadap item yang digunakan. Jawaban untuk persetujuan disebut sebagai *Level of Agreement* (Vagias, 2006 dalam Arifin, 2015). *Level of Agreement* tersebut yaitu :

- a. Sangat Tidak Setuju
- b. Tidak Setuju
- c. Ragu-ragu
- d. Setuju
- e. Sangat Setuju

Untuk keperluan analisis kuantitatif, menurut Sugiyono (2011) dalam Arifin (2015) jawaban di atas dapat diberi skor sebagai berikut :

- a. Sangat Tidak Setuju (STS) diberi skor 1
- b. Tidak Setuju (TS) diberi skor 2
- c. Ragu-ragu (RR) diberi skor 3
- d. Setuju (S) diberi skor 4
- e. Sangat Setuju (SS) diberi skor 5

Data hasil dari pengujian usability dianalisis dengan menghitung rata-rata jawaban berdasarkan skor setiap jawaban dari responden. Berdasarkan skor yang telah ditetapkan dapat dihitung sebagai berikut (Sugiyono, 2011 dalam Arifin, 2015) :

$$Skor_{total} = (J_{SS} \times 5) + (J_S \times 4) + (J_{RR} \times 3) + (J_{STS} \times 1) \quad (2.7)$$

Keterangan :

$J_{SS}$  = jumlah responden menjawab Sangat Setuju

$J_S$  = jumlah responden menjawab Setuju

$J_{RR}$  = jumlah responden menjawab Ragu-ragu

$J_{TS}$  = jumlah responden menjawab Tidak Setuju

$J_{STS}$  = jumlah responden menjawab Sangat Tidak Setuju

Setelah skor total didapatkan kemudian mencari persentase skor untuk mendapatkan interpretasi hasil pengujian *Usability* menggunakan rumus (Sugiyono, 2011 dalam Arifin, 2015) :

$$P_{skor} = \frac{Skor_{total}}{i \times r \times 5} \times 100\% \quad (2.8)$$

Keterangan :

$Skor_{total}$  = Skor total hasil responden menjawab

$i$  = Jumlah Pertanyaan

$r$  = Jumlah Responden

Setelah persentase skor didapatkan kemudian dibandingkan dengan Tabel 2.7.

**Tabel 2.7 Kategori Penilaian *Usability***

Presentase Skor	Interpretasi
0% - 20%	Sangat Tidak Layak
21% - 40%	Tidak Layak
41% - 60%	Cukup Layak
61% - 80%	Layak
81% - 100%	Sangat Layak

Sumber : Guritno, Sudaryono, & Rahardja (2011) dalam Arifin (2015)

### 2.7.4 Security

Merupakan kemampuan produk atau sistem melindungi informasi dan data sehingga orang atau produk atau sistem lainnya memiliki tingkat akses data yang sesuai dengan jenis dan tingkat otorisasi mereka (Alves, 2014). Karakteristik ini terdiri dari 5 subkarakteristik, antara lain (ISO/IEC 25010) :

a. *Confidentiality*

Kemampuan suatu produk atau sistem memastikan bahwa data dapat diakses hanya untuk mereka yang memiliki akses.

b. *Integrity*

Kemampuan sistem atau produk untuk mencegah akses tidak sah dan modifikasi sistem.

c. *Non-repudiation*

Kemampuan sistem dapat melindungi terhadap penolakan aktifitas yang sudah pernah dilakukan. Artinya sistem harus dapat menjaga agar pengguna sistem tidak dapat menyangkal telah melakukan sebuah aktifitas.

d. *Accountability*

Sejauh mana tindakan entitas dapat ditelusuri secara unik. Artinya setiap aktifitas pengguna di dalam sistem akan direkam (logged).

e. *Authenticity*

Sejauh mana identitas subjek dapat terbukti menjadi salah satu yang dapat diklam. Dengan kata lain, sistem dapat membuktikan bahwa pengguna memang benar-benar orang yang memiliki identitas yang mereka klaim.

Menurut Arifin (2015) Pengujian security dapat dilakukan menggunakan aplikasi Acunetix Web Vulnerability Scanner. Acunetix Web Vulnerability Scanner adalah aplikasi untuk memindai/*scanning* situs, termasuk aplikasi web yang terintegrasi, server web dan server Acunetix Web Vulnerability juga dapat memindai aplikasi berbasis web untuk menemukan *vulnerability* (kerentanan). *Vulnerability*. *Vulnerability* merupakan kelemahan yang memungkinkan penyerang (attacker) untuk mengurangi jaminan informasi sebuah sistem. *Vulnerability* adalah titik temu dari tiga unsur, yaitu kerentanan sistem atau kelemahan sistem, penyerang mengakses kelemahan sistem, dan kemampuan penyerang untuk mengeksploitasi kelemahan sistem (The Three Tenets of Cyber Security, 2009). Dengan demikian, aplikasi Acunetix Web Vulnerability Scanner dapat untuk menguji subkarakteristik *confidentiality*, *integrity*, dan *non-repudiation*.

Apabila sistem dapat menunjukkan aktifitas pengguna di dalam sistem, maka sistem tersebut lulus dalam pengujian karakteristik *accountability*.

Apabila sistem dapat membuktikan bahwa data yang digunakan masuk (login) ke dalam sistem merupakan data yang memiliki identitas yang benar, maka sistem tersebut lulus dalam pengujian karakteristik *authenticity*.

### 2.7.5 Portability

Merupakan kemampuan sebuah produk perangkat lunak untuk dapat dipindahkan dalam lingkungan yang berbeda (ISO/IEC 25010). Karakteristik ini terdiri dari 3 subkarakteristik, antara lain (ISO/IEC 25010) :

a. *Adaptability*

Sejauh mana suatu produk atau sistem dapat bekerja secara efektif dan efisien pada saat dijalankan pada hardware, software atau lingkungan operasional yang berbeda.

b. *Installability*

Sistem atau produk dapat diinstall atau dihapus dalam lingkungan tertentu.

c. *Replaceability*

Sejauh mana suatu produk dapat mengganti produk perangkat lunak lain yang ditentukan untuk tujuan yang sama di lingkungan yang sama.

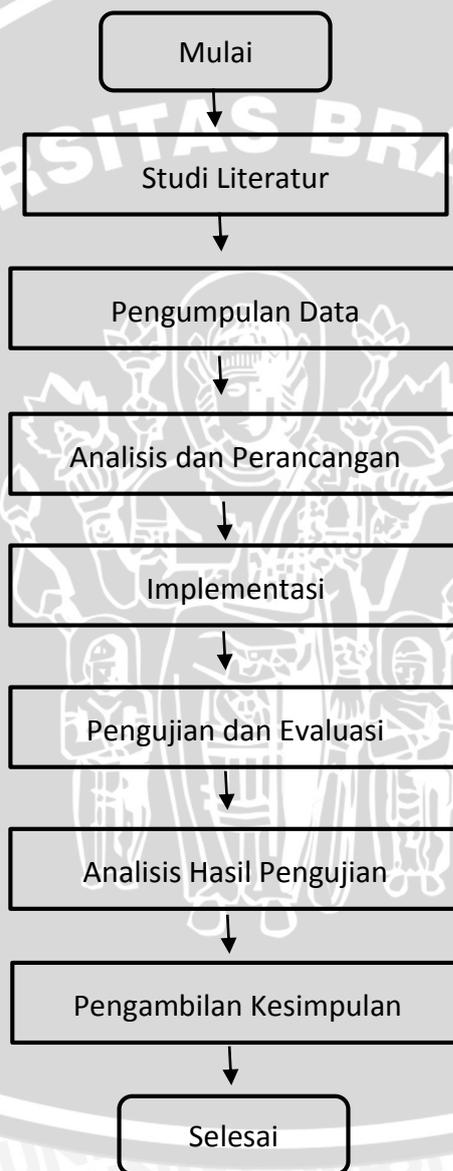
Sistem Seleksi Penerimaan Siswa Baru SMA merupakan sistem berbasis web. Dan sistem ini tidak menggantikan perangkat lunak yang lain. Maka dari itu, subkarakteristik *instability* dan *replaceability* tidak diujikan.

Sistem ini diakses oleh software yang berbeda. Software tersebut adalah browser. Pengujian terhadap subkarakteristik *adaptability* dapat diujikan dengan *crossbrowser* dengan cara menjalankan sistem pada jenis *browser* yang berbeda-beda (Arifin, 2015). Apabila sistem dapat berfungsi dengan baik, maka sistem tersebut lulus dalam pengujian *adaptability* sekaligus *portability*.



## BAB 3 METODOLOGI

Bagian ini menjelaskan langkah-langkah yang akan dilakukan dalam pembuatan Sistem Seleksi Penerimaan Siswa Baru SMA Berbasis Web dengan Pengujian Berbasis ISO 25010. Tahapan yang akan dilakukan antara lain studi literatur, pengumpulan data, analisis dan perancangan, implementasi, pengujian dan evaluasi, serta pengambilan kesimpulan. Tahapan metodologi penelitian dapat digambarkan sebagai berikut:



Gambar 3.1 Diagram Metodologi Penelitian

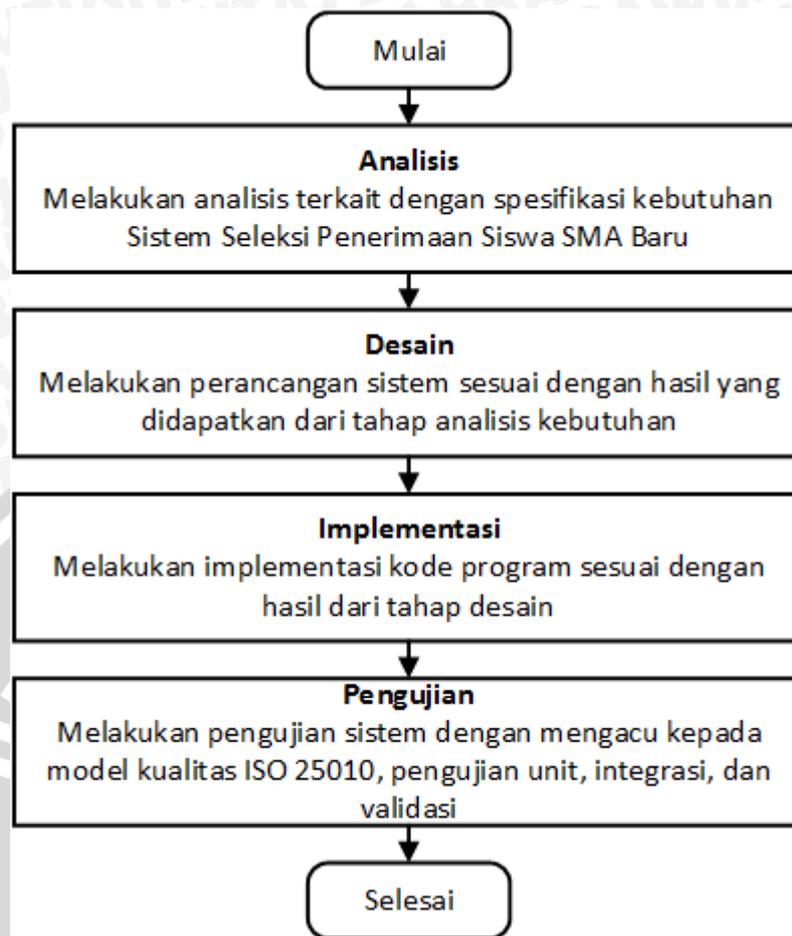
### 3.1 Studi Literatur

Studi literatur merupakan langkah untuk mencari literatur dan memahami konsep-konsep rancangan sistem yang harus dipelajari. Literatur dapat berupa buku, teks, paper, dan literatur online.

Studi literatur dilakukan dengan mengumpulkan dan mempelajari literatur-literatur yang berkaitan dengan perancangan Sistem Seleksi Penerimaan Siswa Baru dan literatur-literatur yang berkaitan dengan pengujian ISO 25010. Kegiatan ini dilakukan untuk mempelajari teori serta keperluan analisis untuk mendapatkan data yang sesungguhnya. Literatur yang digunakan dalam perencanaan dan pembangunan Sistem Seleksi Penerimaan Siswa Baru, antara lain :

1. Penerimaan Siswa Baru
2. Model Waterfall
3. *Unified Modelling Language* (UML)
  - a. Diagram *Use case*
  - b. Diagram Kelas
  - c. Diagram Aktivitas
  - d. *Sequence Diagram*
4. Pengujian Integrasi
5. ISO 25010
  - a. Functional Suitability
  - b. Performance Efficiency
  - c. Compatibility
  - d. Usability
  - e. Reliability
  - f. Security
  - g. Maintability
  - h. Portability

Model yang digunakan dalam mengembangkan sistem pada penelitian ini adalah model *waterfall*. Model *waterfall* cocok digunakan dalam mengembangkan sistem ini didasarkan karena kebutuhan sistem yang dibangun tidak berubah ubah. Tahapan pengembangan yang dilakukan menggunakan model *waterfall* dapat dilihat pada Gambar 3.2.



**Gambar 3.2 Diagram Alir Pengembangan Sistem Model Waterfall**

Pengembangan sistem dengan model *waterfall* pada penelitian ini terdiri dari empat tahap, yaitu :

1. Analisis

Pada tahap ini, spesifikasi kebutuhan dari Sistem Seleksi Penerimaan Siswa Baru SMA dianalisis dengan mengkaji penelitian-penelitian terdahulu. Metode tersebut dipilih karena proses dan fitur di sebagian besar penelitian yang membahas tentang sistem seleksi penerimaan siswa baru dapat ditemukan lebih jelas dan efisien dibandingkan dengan survey pada beberapa SMA. Sehingga, studi literatur dipilih dalam tahap ini. Hasil dari tahap ini adalah spesifikasi kebutuhan perangkat lunak yang disertai dengan identifikasi aktor pada sistem dan diagram *use case* beserta skenarionya.

2. Desain

Pada tahap ini, sistem akan dirancang berdasarkan hasil yang didapatkan pada tahap analisis. Hasil dari tahap ini yaitu desain arsitektur, desain diagram kelas, diagram aktivitas, diagram sekuen, dan desain antarmuka sistem.

### 3. Implementasi

Pada tahap implementasi, hasil yang didapatkan pada tahap desain akan diubah menjadi kode program. Kode program tersebut diimplementasikan dengan bahasa PHP, sintaks HTML, CSS, dan Javascript. Hasi dari tahap ini adalah sistem yang sesuai dengan rancangan sistem pada tahap desain.

### 4. Pengujian

Pada tahap ini, sistem diuji menggunakan model kualitas ISO 25010 serta pengujian unit, pengujian integrasi, dan pengujian validasi. Sistem diuji hanya pada 5 karakteristik ISO 25010, yaitu *functional suitability, performance efficiency, usability, security, dan portability*. Hasi dari tahap ini berupa sistem yang telah memnuhi standar model ISO 25010.

## 3.2 Pengumpulan Data

Pengumpulan data mempunyai peranan yang penting dalam menyelesaikan penelitian ini. Pengumpulan data ini dilakukan kepada guru dan staff tata usaha di SMAN 1 Purwosari. Data pada penelitian ini menggunakan data pertanyaan-pertanyaan yang dapat menentukan kebutuhan sistem seleksi penerimaan siswa SMA baru. Pengumpulan data dilakukan dengan metode wawancara. Metode wawancara adalah metode yang digunakan untuk melakukan wawancara secara langsung kepada pihak terkait seperti kepada guru dan staff tata usaha di SMAN 1 Purwosari.

## 3.3 Rekayasa Kebutuhan

Pada tahap ini dilakukan rekayasa kebutuhan Sistem Seleksi Penerimaan Siswa Baru SMA. Kegiatan analisis kebutuhan bertujuan untuk mengetahui kebutuhan yang diperlukan untuk membangun sistem. Langkah ini dilakukan agar sistem yang dibuat dapat sesuai dengan kebutuhan pengguna. Proses analisis kebutuhan meliputi beberapa tahap elisitasi perangkat lunak, spesifikasi perangkat lunak, validasi, serta manajemen kebutuhan perangkat lunak. Proses analisis kebutuhan dilakukan dengan metode *Object Oriented Analysis*. Proses pemodelan kebutuhan ini menggunakan UML (*Unified Modelling Language*) yang terdiri dari *use case diagram* dan skenario use case.

## 3.4 Perancangan dan Implementasi

### 3.4.1 Perancangan

Perancangan sistem yang akan dibuat berdasarkan analisa kebutuhan sistem yang telah ditentukan sebelumnya. Perancangan sistem meliputi perancangan arsitektur sistem, perancangan diagram sekuen, perancangan diagram kelas, pemodelan data, perancangan komponen dan perancangan antarmuka.

### 3.4.2 Implementasi

Implementasi sistem dibuat berdasarkan dengan perancangan yang sudah dijabarkan pada bab sebelumnya. Implementasi meliputi spesifikasi sistem, batasan implementasi, implementasi program dan implementasi antarmuka. Implementasi dilakukan dengan menggunakan bahasa pemrograman PHP dan memanfaatkan *framework* Bootstrap dan CodeIgniter untuk pembuatan website, serta menggunakan MySQL sebagai pendukung basis data.

### 3.5 Pengujian dan Analisis

Setelah tahap implementasi selesai dilakukan maka sistem siap diuji. Pengujian sistem pada penelitian ini dibagi menjadi dua tahap utama. Tahap pertama sistem diuji menggunakan paradigma rekayasa perangkat lunak yaitu pengujian Unit, pengujian integrasi dan pengujian validasi. Dan pada tahap kedua akan dilakukan pengujian dengan model kualitas ISO 25010.

#### 3.5.1 Pengujian Unit

Pengujian unit pada penelitian ini dilakukan untuk memastikan unit-unit yang memiliki prioritas tinggi memiliki hasil implementasi yang sesuai dengan harapan di awal analisis kebutuhan. Pengujian unit pada penelitian dilakukan dengan menggunakan teknik *white box* dengan jenis pengujian *basis path*. Unit yang akan diujikan pada pengujian ini akan dipilih berdasarkan tiga unit dengan prioritas tinggi.

#### 3.5.2 Pengujian Integrasi

Pengujian integrasi yang dilakukan pada penelitian ini menggunakan pendekatan *Big-Bang* dimana kelas yang diujikan adalah kelas kelas yang saling berinteraksi. Pada pengujian ini, teknik yang digunakan adalah teknik *black box*. Tahapan yang dilakukan pada pengujian integrasi diawali dengan menganalisis proses yang melibatkan interaksi antar kelas. Kemudian membuat kasus uji yang berisi data *input* dan *expect result*. Setelah itu mengeksekusi pengujian. Tahap selanjutnya yaitu membandingkan *actual result* dengan *expect result* berdasarkan hasil eksekusi.

#### 3.5.3 Pengujian Validasi

Pengujian validasi pada penelitian ini dilakukan untuk memastikan seluruh spesifikasi kebutuhan fungsional perangkat lunak telah diimplementasikan sesuai dengan yang diharapkan. Pada pengujian ini, teknik yang digunakan adalah teknik *black box*. Tahapan yang dilakukan pada pengujian validasi yaitu membuat kasus uji setiap kebutuhan fungsional. Setelah itu mengeksekusi pengujian tersebut.

#### 3.5.4 Pengujian Berbasis ISO 25010

Tahap awal pengujian pada penelitian ini mengacu kepada karakteristik karakteristik yang ada pada model kualitas ISO 25010. Karakteristik atau kriteria

yang diuji meliputi *functional suitability*, *performance efficiency*, *usability*, *security*, dan *portability*.

#### 3.5.4.1 *Functional Suitability*

Pada pengujian karakteristik *functionality* semua implementasi dari spesifikasi kebutuhan perangkat lunak akan diuji oleh penguji. Tahapan yang dilakukan yaitu penguji menguji fitur yang diimplementasikan. Kemudian dihitung total nilai yang didapatkan dari seluruh penguji. Setelah itu dihitung tingkat *functional suitability*.

#### 3.5.4.2 *Performance Efficiency*

##### a. *Time Behaviour*

Pengujian subkarakteristik *time behaviour* dilakukan untuk menilai waktu respon (*response time*) sistem. Tahapan yang dilakukan yaitu menguji *response time* dengan WAPT. Setelah itu membandingkan hasil uji dengan Standar Jakob Nielsen.

##### b. *Resource Utilization*

*Resource utilization* dapat diketahui dengan *tool* Resource Monitor. Tahapan yang dilakukan yaitu membuka Mozilla Firefox tanpa membuka tab apapun. Kemudian dicatat jumlah RAM dan CPU yang digunakan. Setelah itu, membuka Sistem Seleksi Penerimaan Siswa Baru SMA pada Mozilla Firefox yang tadi dan dicatat jumlah yang digunakan. Kemudian, dihitung selisih jumlah RAM dan CPU yang digunakan.

##### c. *Capacity*

*Capacity* Sistem Seleksi Penerimaan Siswa Baru SMA dapat diketahui dengan *tool* Apache jMeter. Langkah langkahnya yaitu Sistem Seleksi diuji dengan beban 120 *user* menggunakan *tool* Apache jMeter. 120 *user* dipilih karena pada saat ujian seleksi, jumlah calon siswa yang melakukan ujian seleksi berjumlah 120 calon siswa pada waktu yang sama. Kemudian, jumlah *user* ditambah 1 secara terus menerus sampai batas stabil sistem. Jumlah *user* yang didapatkan pada batas stabil sistem adalah nilai dari *capacity*.

#### 3.5.4.3 *Usability*

Pengujian karakteristik *usability* pada penelitian ini menggunakan model USE *Questionnaire* yang terdiri dari tiga puluh butir pernyataan. Setiap kuesioner yang telah dijawab oleh responden akan dihitung berdasarkan aturan USE *Questionnaire*. Lalu dihitung skor totalnya. Kemudian, dihitung prosentase skor total. Setelah prosentase skor total didapatkan, prosentase tersebut dibandingkan dengan tabel interpretasi.

#### 3.5.4.4 Security

##### a. Confidentiality, Integrity, dan Non-Repudiation

Pengujian subkarakteristik *confidentiality*, *integrity*, dan *non-repudiation* pada penelitian ini menggunakan tool Acunetix Web Vulnerability Scanner. Tahapan yang dilakukan pada pengujian subkarakteristik *confidentiality*, *integrity*, dan *non-repudiation* yaitu menguji perlindungan data dan akses sistem menggunakan Acunetix Web Vulnerability Scanner. Kemudian membandingkan hasil uji dengan Standar yang ada di Acunetix User Manual Book.

##### b. Accountability

Pengujian subkarakteristik *accountability* pada penelitian ini dilakukan dengan cara melihat apakah sistem dapat merekam aktifitas pengguna. Tahapan yang dilakukan adalah *user* melakukan sebuah aktifitas. Setelah itu melihat apakah sistem dapat merekam aktifitas pengguna atau tidak.

##### c. Authenticity

Pengujian subkarakteristik *authenticity* pada penelitian ini dilakukan dengan cara skenario pengujian. Tahapan yang dilakukan yaitu melakukan skenario pengujian dengan beberapa pengguna masuk sistem dengan username dan password milik pengguna tersebut. Kemudian, membandingkan hasil yang diharapkan dengan hasil yang didapatkan.

#### 3.5.4.5 Portability

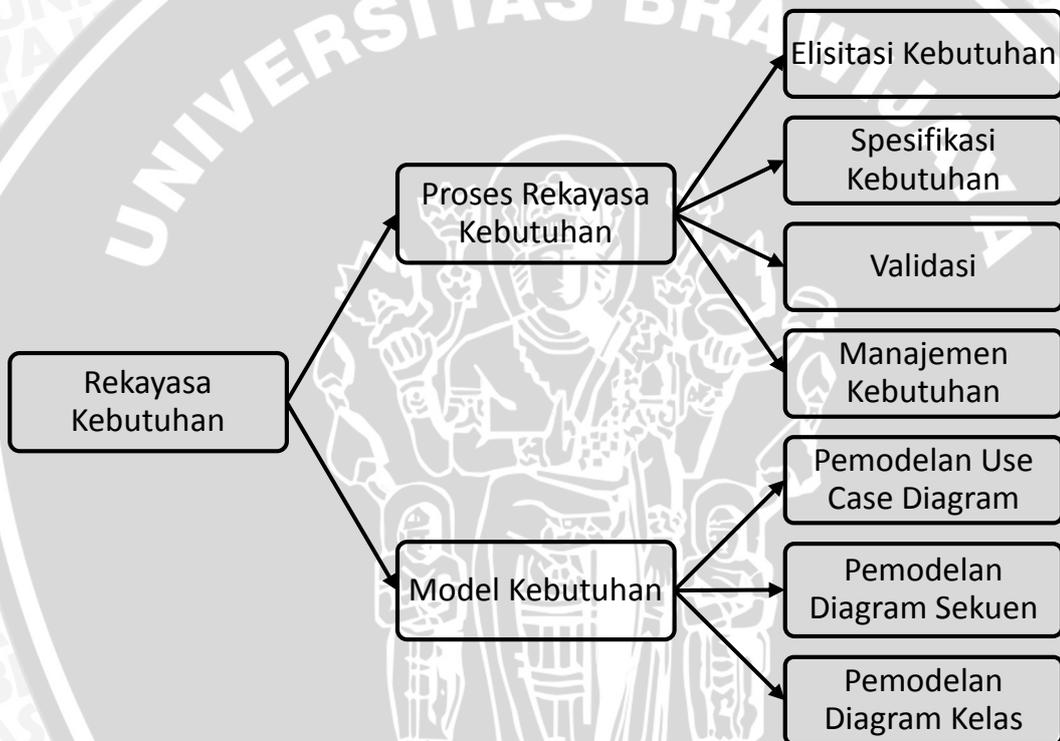
Pengujian pada karakteristik *portability* dilakukan dengan menguji sistem apakah sistem dapat berfungsi dengan baik pada *browser* yang berbeda beda. Tahapan yang dilakukan yaitu menjalankan sistem pada 3 jenis dekstop browser dan 3 jenis mobile browser. Sistem memenuhi *portability* bila sistem dapat berfungsi dengan baik.

### 3.6 Pengambilan Kesimpulan

Pengambilan kesimpulan dilakukan setelah semua tahapan perancangan, implementasi, dan pengujian sistem telah selesai dilakukan. Kesimpulan diambil dari hasil pengujian dan analisis terhadap sistem yang dibangun. Tahap terakhir dari penulisan adalah saran yang dimaksudkan untuk memperbaiki kesalahan yang terjadi dan menyempurnakan penulisan serta untuk memberikan pertimbangan atas pengembangan sistem selanjutnya. Setelah pengambilan kesimpulan selesai, penelitian diakhiri dengan proses akhir yakni pembuatan laporan berupa karya ilmiah.

## BAB 4 REKAYASA KEBUTUHAN

Bab ini membahas proses analisa kebutuhan Sistem Seleksi Penerimaan Siswa Baru SMA Berbasis Web dengan Pengujian Kualitas Berbasis ISO 25010. Analisa kebutuhan perangkat lunak dibuat untuk menggambarkan kebutuhan-kebutuhan yang harus disediakan oleh sistem agar dapat memenuhi kebutuhan pengguna. Dalam rekayasa kebutuhan terbagi menjadi proses rekayasa kebutuhan dan model kebutuhan. Proses rekayasa kebutuhan terdiri dari elisitasi, spesifikasi, validasi dan manajemen kebutuhan. Pemodelan kebutuhan terdiri dari pemodelan *use case diagram*. Tahap-tahap rekayasa kebutuhan Sistem Seleksi Penerimaan Siswa Baru SMA Berbasis Web dengan Pengujian Kualitas Berbasis ISO 25010 dapat dilihat pada gambar 4.1 berikut.



Gambar 4.1 Diagram Rekayasa Kebutuhan

### 4.1 Proses Rekayasa Kebutuhan

Proses rekayasa kebutuhan merupakan tahap untuk mengumpulkan kebutuhan-kebutuhan sistem dalam membangun Sistem Seleksi Penerimaan Siswa Baru SMA. Tahap-tahap dalam proses ini adalah elisitasi kebutuhan, spesifikasi kebutuhan, validasi verifikasi dan manajemen kebutuhan.

#### 4.1.1 Elisitasi Kebutuhan

Elisitasi kebutuhan adalah proses mengumpulkan informasi tentang sistem yang diperlukan dan sistem yang ada serta untuk menemukan kebutuhan suatu

sistem (Sommerville, 2011). Untuk menemukan kebutuhan sistem, cara yang dapat dilakukan yaitu komunikasi dengan pelanggan, pengguna sistem dan pihak lain yang memiliki kepentingan dalam pengembangan sistem. Pada tahap elisitasi kebutuhan sistem ini terdiri dari identifikasi stakeholder, teknik elisitasi, dan proses bisnis sistem.

#### 4.1.1.1 Identifikasi Stakeholder

Sistem Seleksi Penerimaan Siswa Baru SMA digunakan untuk mempermudah proses registrasi calon siswa SMA baru sampai pengumuman penerimaan siswa baru. Identifikasi *stakeholder* menjelaskan siapa saja yang terlibat dalam sistem yang akan dibangun tersebut. Identifikasi *stakeholder* dapat dilihat lebih lengkap pada tabel 4.1.

**Tabel 4.1 Identifikasi Stakeholder**

<i>Stakeholder</i>	<i>Deskripsi</i>
Kepala Sekolah	Kepala Sekolah adalah orang yang bertanggung jawab terhadap proses seleksi penerimaan siswa baru.
Calon Siswa	Calon siswa merupakan subjek utama dalam sistem seleksi penerimaan siswa SMA baru.
Guru	Guru merupakan aktor utama yang akan membuat soal dan menentukan kriteria penilaian.
Staff Tata Usaha Sekolah	Staff Tata Usaha bertanggung jawab atas pengecekan data registrasi calon siswa baru dan proses validasi pembayaran pendaftaran.

Terdapat 4 *stakeholder* pada Sistem Seleksi Penerimaan Siswa Baru SMA, antara lain: kepala sekolah, calon siswa, guru, dan staff tata usaha sekolah. Kepala sekolah adalah orang yang bertanggung jawab terhadap proses seleksi penerimaan siswa baru. Calon siswa adalah anak yang baru lulus SMP dan ingin mendaftar ke SMA menjadi siswa baru. Guru memiliki peran untuk membuat soal dan menentukan kriteria penilaian. Sedangkan, staff tata usaha berperan sebagai orang yang mengecek data registrasi calon siswa dan memvalidasi pembayaran pendaftaran.

#### 4.1.1.2 Teknik Elisitasi

Pada penelitian ini elisitasi dilakukan dengan metode wawancara dan pengamatan langsung (observasi). Wawancara dilakukan kepada pengguna utama yaitu calon siswa, guru, serta staff tata usaha sekolah SMA di SMAN 1 Purwosari. Selain itu identifikasi masalah juga dilakukan dengan pengamatan langsung yang dilakukan ketika pihak SMA melaksanakan proses penerimaan siswa baru. Dari kedua metode tersebut maka dapat ditemukan kebutuhan yang dapat dilihat pada Tabel 4.2.

**Tabel 4.2 Requirement Discovery**

No	Temuan Kebutuhan (Requirement Discovery)
1	Proses registrasi calon siswa dapat disimpan dalam sistem
2	Staff tata usaha dapat mengelola data registrasi calon siswa pada sistem

**Tabel 4.2 Requirement Discovery (lanjutan)**

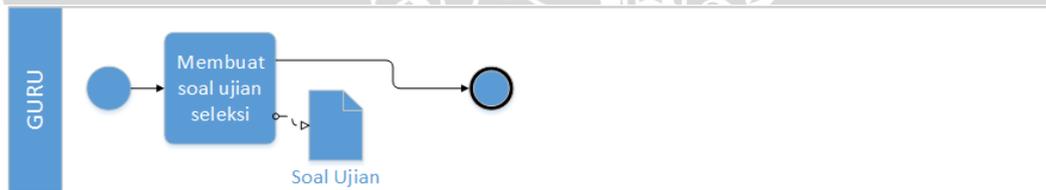
No	Temuan Kebutuhan (Requirement Discovery)
3	Staff tata usaha dapat mengelola pengumuman terkait pelaksanaan seleksi penerimaan siswa baru
4	Guru dapat mengelola soal ujian seleksi penerimaan siswa baru.
5	Calon siswa dapat mengerjakan soal ujian seleksi pada sistem sesuai dengan waktu yang ditentukan.
6	Guru dapat menentukan jumlah siswa yang diterima.
7	Calon siswa dapat melihat hasil seleksi penerimaan siswa baru.
8	Sistem memiliki tampilan <i>user friendly</i>
9	Sistem memiliki keamanan yang hanya diakses oleh user yang teridentifikasi
10	Sistem dapat diakses di beberapa browser dan mobile.

Berdasarkan temuan kebutuhan pada tabel 4.2, masih perlu dispesifikasikan lagi seperti mengelola pengumuman. Pada sistem, mengelola pengumuman akan dispesifikasikan menjadi lihat pengumuman, buat pengumuman, edit pengumuman, dan hapus pengumuman.

**4.1.1.3 Analisa Proses Bisnis**

Penerimaan siswa baru di SMA Negeri 1 Purwosari terdiri dari 5 tahap. Tahap pertama adalah tahap pembuatan soal ujian seleksi. Tahap kedua adalah pengumuman tanggal registrasi, tanggal ujian seleksi, dan tanggal pengumuman hasil seleksi. Tahap ketiga adalah registrasi calon siswa. Tahap keempat adalah ujian seleksi. Tahap kelima adalah pengumuman hasil seleksi.

Proses bisnis pembuatan soal ujian seleksi dapat dilihat pada gambar 4.2.

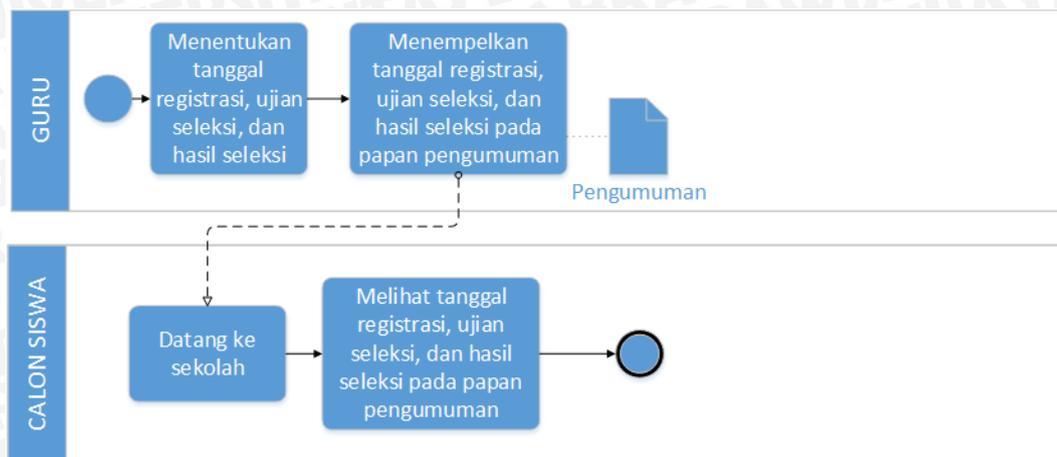


**Gambar 4.2 Proses Bisnis Pembuatan Soal Ujian Seleksi**

Pada proses bisnis pembuatan soal ujian seleksi, guru-guru SMA Negeri 1 Purwosari membuat soal soal untuk ujian seleksi penerimaan siswa baru SMA. Soal soal tersebut adalah soal pilihan ganda. Soal soal tersebut meliputi bahasa indonesia, matematika, fisika, biologi, kimia, dan bahasa inggris.

Proses selanjutnya adalah mengumumkan tanggal registrasi, ujian seleksi, dan tanggal pengumuman hasil seleksi. Proses bisnis mengumumkan tanggal registrasi, ujian seleksi, dan tanggal pengumuman hasil seleksi dapat dilihat pada gambar 4.3.

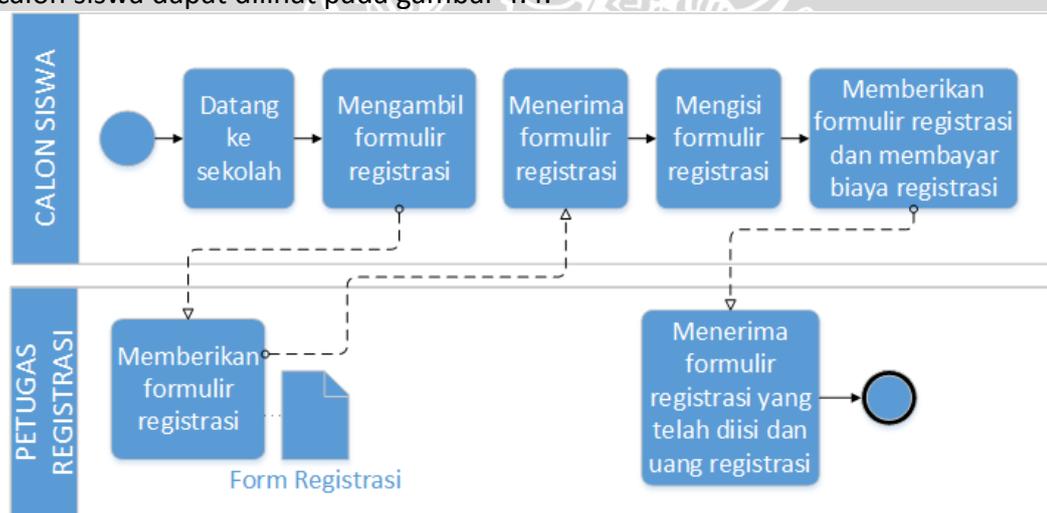




**Gambar 4.3 Proses Bisnis Mengumumkan Tanggal Registrasi, Ujian Seleksi, dan Tanggal Pengumuman Hasil Seleksi**

Proses bisnis mengumumkan tanggal registrasi, ujian seleksi, dan tanggal pengumuman hasil seleksi diawali dengan guru menentukan tanggal registrasi, ujian seleksi, dan tanggal pengumuman hasil seleksi. Setelah tanggal tanggal tersebut ditentukan, guru akan menempelkan tanggal tanggal tersebut pada papan pengumuman. Kemudian, calon siswa akan datang ke sekolah untuk melihat tanggal tanggal tersebut.

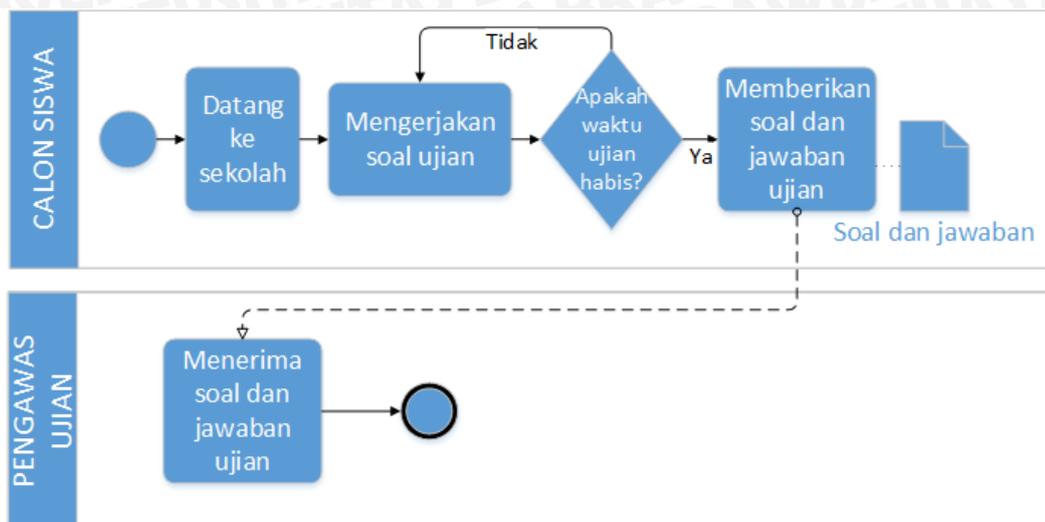
Proses selanjutnya adalah registrasi calon siswa. Proses bisnis registrasi calon siswa dapat dilihat pada gambar 4.4.



**Gambar 4.4 Proses Bisnis Registrasi Calon Siswa**

Proses bisnis registrasi calon siswa diawali dengan seluruh calon siswa datang ke sekolah. Kemudian calon siswa mengambil formulir pendaftaran ke petugas registrasi. Selanjutnya calon siswa mengisi formulir tersebut. Apabila calon siswa sudah mengisi formulir registrasi, formulir tersebut diberikan kepada petugas registrasi. Pada saat memberikan formulir, calon siswa juga membayar biaya registrasi.

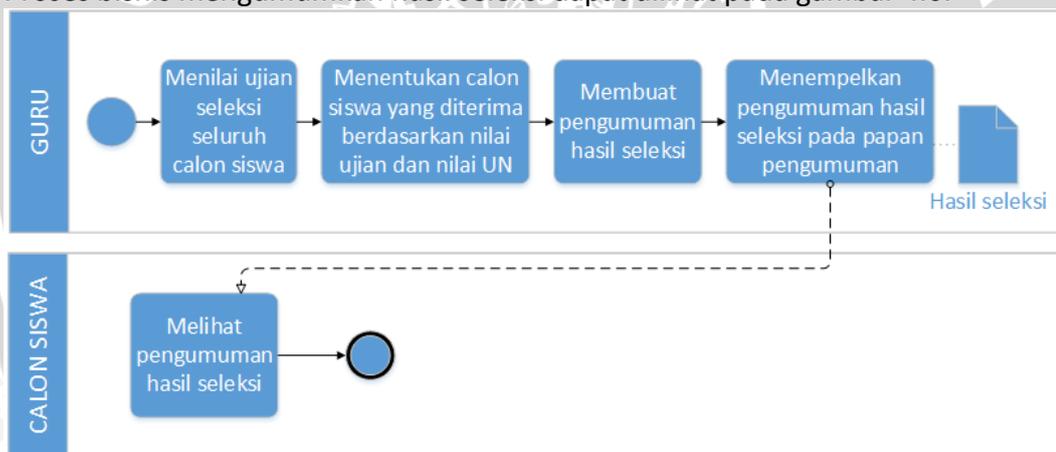
Proses selanjutnya adalah ujian seleksi. Proses bisnis ujian seleksi dapat dilihat pada gambar 4.5.



**Gambar 4.5 Proses Bisnis Ujian Seleksi**

Proses bisnis ujian seleksi diawali dengan seluruh calon siswa datang ke sekolah. Kemudian calon siswa mengerjakan soal ujian. Apabila waktu ujian telah habis, calon siswa mengumpulkan soal beserta jawaban ke pengawas ujian.

Proses selanjutnya adalah pihak sekolah mengumumkan hasil seleksi. Proses bisnis mengumumkan hasil seleksi dapat dilihat pada gambar 4.6.



**Gambar 4.6 Proses Bisnis Mengumumkan Hasil Seleksi**

Proses bisnis mengumumkan hasil seleksi diawali dengan guru menilai ujian seleksi seluruh calon siswa. Kemudian guru menentukan calon siswa yang diterima dan yang tidak diterima berdasarkan nilai ujian dan nilai UN. Setelah itu guru membuat pengumuman hasil seleksi. Kemudian, pengumuman hasil seleksi tersebut ditempel pada papan pengumuman. Lalu calon siswa melihat pengumuman hasil seleksi tersebut.

Seluruh proses bisnis mulai dari proses bisnis pembuatan soal ujian seleksi sampai proses bisnis mengumumkan hasil seleksi dibuat berdasarkan observasi dan wawancara dengan guru guru SMA Negeri 1 Purwosari.

#### 4.1.1.4 Analisa Pengujian Kualitas Berbasis ISO 25010

Stakeholder dari Sistem Seleksi Penerimaan Siswa Baru SMA ini yaitu pembangun sistem (system builder), pemilik sistem, dan pengguna sistem. Pemilik sistem adalah pihak sekolah. Sedangkan, pengguna sistem adalah calon siswa, guru, dan staff tata usaha.

Pihak sekolah membutuhkan sistem yang sesuai dengan kebutuhan, yaitu kebutuhan dalam menangani seleksi penerimaan siswa SMA baru. Pihak sekolah juga membutuhkan sistem dengan kinerja yang tepat, dapat melindungi informasi dan data, sistem yang efektif dan efisien, dan dapat merekam aktifitas pengguna.

Calon siswa membutuhkan sistem dengan kinerja yang tepat, dapat melindungi informasi dan data, serta sistem tersebut harus dapat dijalankan pada *browser* yang berbeda beda. Karena setiap calon siswa tidak mungkin menggunakan *browser* yang sama.

Dengan demikian, karakteristik yang perlu diujikan pada Sistem Seleksi Penerimaan Siswa Baru SMA yaitu functional suitability, performance efficiency, usability, security, dan portability.

#### 4.1.2 Spesifikasi Kebutuhan

Spesifikasi kebutuhan sistem ini mengacu kepada elisitasi kebutuhan pada bagian sebelumnya. Berikut spesifikasi kebutuhan sistem dari hasil elisitasi yang dilakukan.

##### 4.1.2.1 Kebutuhan Fungsional

Daftar kebutuhan fungsional Sistem Seleksi Penerimaan Siswa Baru SMA dijelaskan pada tabel 4.3.

**Tabel 4.3 Spesifikasi Kebutuhan Fungsional**

Kode	Kebutuhan Fungsional	Deskripsi	Aktor	Prioritas
FR-01	Registrasi calon siswa	Sistem mampu menerima dan menyimpan data registrasi calon siswa, yaitu NISN, nama lengkap, tempat lahir, tanggal lahir, alamat, jenis kelamin, nomor hp, email, nama ayah, pekerjaan ayah, nama ibu, pekerjaan ibu, asal sekolah, nilai rata-rata UN, ijazah, foto, dan password.	User	Tinggi
FR-02	Melihat pengumuman	Sistem dapat melihat pengumuman mengenai seleksi penerimaan siswa SMA baru.	User	Tinggi
FR-03	Melihat profil	Sistem dapat melihat profil calon siswa.	Calon Siswa	Tinggi

Tabel 4.3 Spesifikasi Kebutuhan Fungsional (lanjutan)

Kode	Kebutuhan Fungsional	Deskripsi	Aktor	Prioritas
FR-04	Mengedit profil	Sistem dapat mengedit data profil calon siswa	Calon Siswa	Rendah
FR-05	Menghapus profil	Sistem dapat menghapus profil calon siswa apabila calon siswa batal mendaftar.	Calon Siswa	Rendah
FR-06	Mengupload bukti pembayaran	Sistem dapat melakukan upload bukti pembayaran dari Bank atau ATM sebagai bukti bahwa pendaftar telah melakukan pembayaran.	Calon Siswa	Sedang
FR-07	Melihat jadwal ujian	Sistem dapat menampilkan jadwal ujian calon siswa	Calon Siswa	Tinggi
FR-08	Mencetak kartu ujian	Sistem dapat mencetak kartu ujian.	Calon Siswa	Sedang
FR-09	Mengerjakan soal	Sistem dapat menampilkan soal soal ujian seleksi yang nantinya akan dikerjakan calon siswa.	Calon Siswa	Tinggi
FR-10	Melihat hasil seleksi	Sistem dapat menampilkan hasil seleksi calon siswa yang login ke dalam sistem.	Calon Siswa	Tinggi
FR-11	Melihat profil admin	Sistem dapat menampilkan profil admin	Admin	Rendah
FR-12	Melihat seluruh admin	Sistem dapat menampilkan data seluruh admin	Admin	Rendah
FR-13	Menambahkan user admin	Sistem dapat menambah user untuk admin.	Admin	Rendah
FR-14	Mengedit profil admin	Sistem dapat mengedit profil admin	Admin	Rendah
FR-15	Menghapus profil admin	Sistem dapat menghapus profil admin	Admin	Rendah
FR-16	Melihat seluruh staff TU	Sistem dapat menampilkan data seluruh staff TU	Admin	Rendah
FR-17	Menambah user staff TU	Sistem dapat menambah user staff TU.	Admin	Rendah
FR-18	Menghapus user staff TU	Sistem dapat menghapus user staff TU	Admin	Rendah
FR-19	Memposting pengumuman	Sistem dapat memposting pengumuman mengenai seleksi penerimaan siswa SMA baru.	Admin	Sedang
FR-20	Mengedit pengumuman	Sistem dapat mengedit pengumuman yang telah diposting.	Admin	Sedang

Tabel 4.3 Spesifikasi Kebutuhan Fungsional (lanjutan)

Kode	Kebutuhan Fungsional	Deskripsi	Aktor	Prioritas
FR-21	Menghapus pengumuman	Sistem dapat menghapus pengumuman yang telah diposting.	Admin	Sedang
FR-22	Membuat soal	Sistem dapat memasukkan soal ujian ke dalam sistem	Admin	Tinggi
FR-23	Mengedit soal	Sistem dapat mengedit soal ujian.	Admin	Rendah
FR-24	Menghapus soal	Sistem dapat menghapus soal ujian.	Admin	Rendah
FR-25	Melihat hasil seleksi seluruh calon siswa	Sistem dapat melihat hasil seleksi seluruh peserta ujian.	Admin	Sedang
FR-26	Menentukan jumlah siswa diterima	Sistem dapat menentukan jumlah pendaftar yang diterima sebagai siswa baru SMA	Admin	Sedang
FR-27	Menentukan jadwal pendaftaran	Sistem dapat menentukan jadwal pendaftaran.	Admin	Sedang
FR-28	Menentukan jadwal ujian	Sistem dapat menentukan jadwal ujian	Admin	Sedang
FR-29	Menentukan jadwal pengumuman seleksi	Sistem dapat menentukan jadwal pengumuman seleksi penerimaan siswa SMA baru.	Admin	Sedang
FR-30	Melihat profil staff TU	Sistem dapat menampilkan profil staff TU	Staff TU	Rendah
FR-31	Mengedit profil staff TU	Sistem dapat mengubah profil staff TU	Staff TU	Rendah
FR-32	Melihat seluruh calon siswa	Sistem dapat melihat data calon siswa	Staff TU	Rendah
FR-33	Menghapus profil calon siswa	Sistem dapat menghapus profil calon siswa.	Staff TU	Rendah
FR-34	Memvalidasi pembayaran	Sistem dapat memvalidasi bukti pembayaran yang telah diupload oleh pendaftar	Staff TU	Sedang
FR-35	Login	Sistem harus menyediakan fitur autentikasi berupa login	User	Rendah
FR-36	Logout	Sistem harus menyediakan fitur keluar dari sistem	Calon Siswa, Staff TU, & Admin	Rendah

#### 4.1.2.2 Kebutuhan Non-Fungsional

Daftar kebutuhan non-fungsional Sistem Seleksi Penerimaan Siswa Baru SMA dapat dilihat pada tabel 4.4.

Tabel 4.4 Spesifikasi Kebutuhan Non-Fungsional

Kode	Parameter	Deskripsi
NFR-01	<i>Usability</i>	Sistem memiliki tampilan yang mudah digunakan dan memiliki tampilan yang konsisten saat digunakan oleh pengguna.
NFR-02	<i>Performance Efficiency</i>	Sistem dapat melayani perintah pengguna dengan batas waktu respons di bawah 10 detik.
NFR-03	<i>Security</i>	Sistem hanya dapat diakses oleh pengguna yang mempunyai akun yang sudah terdaftar di sistem.
NFR-04	<i>Portability</i>	Sistem dapat dijalankan pada berbagai <i>web browser</i> yang berbeda.

Kebutuhan non-fungsional Sistem Seleksi Penerimaan Siswa Baru SMA nantinya akan diuji pada pengujian karakteristik ISO 25010, karena *usability*, *performance efficiency*, *security* dan *portability* merupakan karakteristik karakteristik pada ISO 25010 yang akan diujikan.

#### 4.1.3 Validasi

Spesifikasi kebutuhan yang telah didefinisikan perlu dilakukan proses validasi. Validasi adalah proses pengecekan bahwa sistem memenuhi kebutuhan dan harapan dari *customer* (Sommerville, 2011). Proses ini memastikan kebutuhan - kebutuhan yang telah disepakati sudah ada pada sistem seperti fitur – fitur yang diperlukan. Proses validasi pada penelitian ini dilakukan oleh guru dan staff TU.

##### 4.1.3.1 Validasi Kebutuhan Fungsional

Validasi dari kebutuhan fungsional sistem yang telah dilakukan dapat dilihat pada tabel 4.5.

Tabel 4.5 Validasi Kebutuhan Fungsional

Kode	Kebutuhan Fungsional	Validasi
FR-01	Registrasi calon siswa	√
FR-02	Melihat pengumuman	√
FR-03	Melihat profil	√
FR-04	Mengedit profil	√
FR-05	Menghapus profil	√
FR-06	Mengupload bukti pembayaran	√
FR-07	Melihat jadwal ujian	√
FR-08	Mencetak kartu ujian	√
FR-09	Mengerjakan soal	√
FR-10	Melihat hasil seleksi	√
FR-11	Melihat profil admin	√
FR-12	Melihat seluruh admin	√
FR-13	Menambahkan user admin	√
FR-14	Mengedit profil admin	√
FR-15	Menghapus profil admin	√
FR-16	Melihat seluruh staff TU	√
FR-17	Menambah user staff TU	√
FR-18	Menghapus user staff TU	√

Tabel 4.5 Validasi Kebutuhan Fungsional (lanjutan)

Kode	Kebutuhan Fungsional	Validasi
FR-19	Memposting pengumuman	√
FR-20	Mengedit pengumuman	√
FR-21	Menghapus pengumuman	√
FR-22	Membuat soal	√
FR-23	Mengedit soal	√
FR-24	Menghapus soal	√
FR-25	Melihat hasil seleksi seluruh calon siswa	√
FR-26	Menentukan jumlah siswa diterima	√
FR-27	Menentukan jadwal pendaftaran	√
FR-28	Menentukan jadwal ujian	√
FR-29	Menentukan jadwal pengumuman seleksi	√
FR-30	Melihat profil staff TU	√
FR-31	Mengedit profil staff TU	√
FR-32	Melihat seluruh calon siswa	√
FR-33	Menghapus profil calon siswa	√
FR-34	Memvalidasi pembayaran	√
FR-35	Login	√
FR-36	Logout	√

#### 4.1.3.2 Validasi Kebutuhan Non Fungsional

Validasi dari kebutuhan fungsional sistem yang telah dilakukan dapat dilihat pada tabel 4.6.

Tabel 4.6 Validasi Kebutuhan Non Fungsional

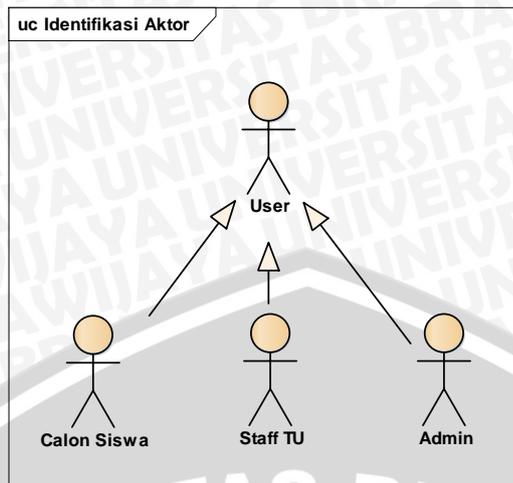
Kode	Parameter	Validasi
NFR-01	<i>Usability</i>	√
NFR-02	<i>Performance Efficiency</i>	√
NFR-03	<i>Security</i>	√
NFR-04	<i>Portability</i>	√

#### 4.1.4 Manajemen Kebutuhan

Manajemen kebutuhan adalah tahap yang dikerjakan setelah menyelesaikan proses validasi dan verifikasi. Fungsi utama dilakukan manajemen kebutuhan ini adalah untuk mengidentifikasi, mengontrol, mencari kebutuhan dan perubahan kebutuhan yang diperlukan seiring berjalannya proses pengerjaan. Dalam manajemen kebutuhan dimodelkan dalam sebuah *use case diagram*.

##### 4.1.4.1 Identifikasi Aktor

Identifikasi aktor bertujuan untuk memberikan gambaran tentang pengguna yang akan terlibat dalam penggunaan sistem. Identifikasi aktor yang akan berperan dalam sistem dapat dilihat pada gambar 4.7.



**Gambar 4.7 Identifikasi Aktor**

Penjelasan gambar 4.7 dapat dilihat pada tabel 4.7.

**Tabel 4.7 Identifikasi User**

Aktor	Spesifikasi	Deskripsi
User	Calon Siswa	Calon siswa merupakan pengguna sistem yang akan melakukan registrasi (pendaftaran), mengupload bukti pembayaran, dan melakukan ujian seleksi penerimaan siswa baru.
	Staff TU	Staff TU merupakan pihak yang akan melakukan pengecekan dan pengelolaan data registrasi calon siswa memvalidasi pembayaran pendaftaran
	Admin	Admin merupakan pihak yang mengelola pengumuman terkait seleksi penerimaan siswa baru, menambahkan soal ujian ke dalam sistem, menentukan jadwal pendaftaran, jadwal ujian, dan jadwal pengumuman, serta menginputkan jumlah siswa yang diterima ke dalam sistem.

#### 4.1.4.2 Identifikasi Kebutuhan

Identifikasi kebutuhan adalah tahap untuk membuat kebutuhan yang telah didefinisikan harus dapat diidentifikasi dengan mudah. Identifikasi tersebut yaitu terdiri dari penyetaraan antara kebutuhan sistem dan kebutuhan interface. Identifikasi kebutuhan Sistem Seleksi Penerimaan Siswa SMA Baru dapat dilihat pada Tabel 4.8 berikut.

**Tabel 4.8 Identifikasi Kebutuhan**

Kode Kebutuhan	Kebutuhan Fungsional	Deskripsi	Kode Interface
FR-01	Registrasi calon siswa	Sistem mampu menerima dan menyimpan data registrasi calon siswa, yaitu NISN, nama lengkap, tempat lahir, tanggal lahir, alamat, jenis kelamin,	IR-01

Tabel 4.8 Identifikasi Kebutuhan (lanjutan)

Kode Kebutuhan	Kebutuhan Fungsional	Deskripsi	Kode Interface
		nomor hp, email, nama ayah, pekerjaan ayah, nama ibu, pekerjaan ibu, asal sekolah, nilai rata-rata UN, ijazah, foto, dan password.	
FR-02	Melihat pengumuman	Sistem dapat melihat pengumuman mengenai seleksi penerimaan siswa SMA baru.	IR-02
FR-03	Melihat profil	Sistem dapat melihat profil calon siswa.	IR-03
FR-04	Mengedit profil	Sistem dapat mengedit data profil calon siswa	IR-04
FR-05	Menghapus profil	Sistem dapat menghapus profil calon siswa apabila calon siswa batal mendaftar.	IR-05
FR-06	Mengupload bukti pembayaran	Sistem dapat melakukan upload bukti pembayaran dari Bank atau ATM sebagai bukti bahwa pendaftar telah melakukan pembayaran.	IR-06
FR-07	Melihat jadwal ujian	Sistem dapat menampilkan jadwal ujian calon siswa	IR-07
FR-08	Mencetak kartu ujian	Sistem dapat mencetak kartu ujian.	IR-08
FR-09	Mengerjakan soal	Sistem dapat menampilkan soal soal ujian seleksi yang nantinya akan dikerjakan calon siswa.	IR-09
FR-10	Melihat hasil seleksi	Sistem dapat menampilkan hasil seleksi calon siswa yang login ke dalam sistem.	IR-10
FR-11	Melihat profil admin	Sistem dapat menampilkan profil admin	IR-11
FR-12	Melihat seluruh admin	Sistem dapat menampilkan data seluruh admin	IR-12
FR-13	Menamambah user admin	Sistem dapat menambah user untuk admin.	IR-13
FR-14	Mengedit profil admin	Sistem dapat mengedit profil admin	IR-14
FR-15	Menghapus profil admin	Sistem dapat menghapus profil admin	IR-15
FR-16	Melihat seluruh staff TU	Sistem dapat menampilkan data seluruh staff TU	IR-16
FR-17	Menambah user staff TU	Sistem dapat menambah user staff TU.	IR-17
FR-18	Menghapus user staff TU	Sistem dapat menghapus user staff TU	IR-18
FR-19	Memposting pengumuman	Sistem dapat memposting pengumuman mengenai seleksi penerimaan siswa SMA baru.	IR-19
FR-20	Mengedit pengumuman	Sistem dapat mengedit pengumuman yang telah diposting.	IR-20
FR-21	Menghapus pengumuman	Sistem dapat menghapus pengumuman yang telah diposting.	IR-21

Tabel 4.8 Identifikasi Kebutuhan (lanjutan)

Kode Kebutuhan	Kebutuhan Fungsional	Deskripsi	Kode Interface
FR-22	Membuat soal	Sistem dapat memasukkan soal ujian ke dalam sistem	IR-22
FR-23	Mengedit soal	Sistem dapat mengedit soal ujian.	IR-23
FR-24	Menghapus soal	Sistem dapat menghapus soal ujian.	IR-24
FR-25	Melihat hasil seleksi seluruh calon siswa	Sistem dapat melihat hasil seleksi seluruh peserta ujian.	IR-25
FR-26	Menentukan jumlah siswa diterima	Sistem dapat menentukan jumlah pendaftar yang diterima sebagai siswa baru SMA	IR-26
FR-27	Menentukan jadwal pendaftaran	Sistem dapat menentukan jadwal pendaftaran.	IR-27
FR-28	Menentukan jadwal ujian	Sistem dapat menentukan jadwal ujian	IR-28
FR-29	Menentukan jadwal pengumuman seleksi	Sistem dapat menentukan jadwal pengumuman seleksi penerimaan siswa SMA baru.	IR-29
FR-30	Melihat profil staff TU	Sistem dapat menampilkan profil staff TU	IR-30
FR-31	Mengedit profil staff TU	Sistem dapat mengubah profil staff TU	IR-31
FR-32	Melihat seluruh calon siswa	Sistem dapat melihat data calon siswa	IR-32
FR-33	Menghapus profil calon siswa	Sistem dapat menghapus profil calon siswa.	IR-33
FR-34	Memvalidasi pembayaran	Sistem dapat memvalidasi bukti pembayaran yang telah diupload oleh pendaftar	IR-34
FR-35	Login	Sistem harus menyediakan fitur autentikasi berupa login	IR-35
FR-36	Logout	Sistem harus menyediakan fitur keluar dari sistem	IR-36

Identifikasi kebutuhan pada Tabel 4.8 memetakan kebutuhan sistem dengan kebutuhan antarmuka. Pemetaan ini akan mempermudah jika adanya kemungkinan perubahan kebutuhan sehingga dapat diidentifikasi dari kebutuhan antarmukanya.

## 4.2 Model Kebutuhan

Model kebutuhan dilakukan untuk memudahkan pembacaan kebutuhan dalam tahap perancangan. Proses pemodelan kebutuhan ini menggunakan *Unified Modelling Language (UML)* yang terdiri dari *usecase diagram*, *sequence diagram*, dan *class diagram*.

#### 4.2.1 Pemodelan Use Case Diagram

Diagram *use case* merupakan model diagram UML (*Unified Modeling Language*) yang memodelkan perilaku sistem dari tampak luar. Diagram ini menggambarkan aktor, *use case*, dan hubungan terkait keduanya pada sistem yang dibangun. Pada bagian ini akan diuraikan bagaimana gambaran *use case* diagram pada Sistem Seleksi Penerimaan Siswa Baru SMA.

##### 4.2.1.1 Diagram Use Case Sistem

Diagram *use case* menggambarkan ruang lingkup dari sebuah sistem yang sedang dibangun. Diagram *use case* terdiri dari dua komponen utama, yaitu *use case* dan aktor. *Use case* meliputi semua hal yang ada pada sistem, sedangkan aktor meliputi semua hal diluar sistem. Aktor termasuk seseorang atau apa saja yang berhubungan dengan sistem yang dibangun. *Use case* dibuat berdasarkan kebutuhan fungsional sistem. Pemetaan *use case* dan kebutuhan fungsional sistem dapat dilihat pada tabel 4.9.

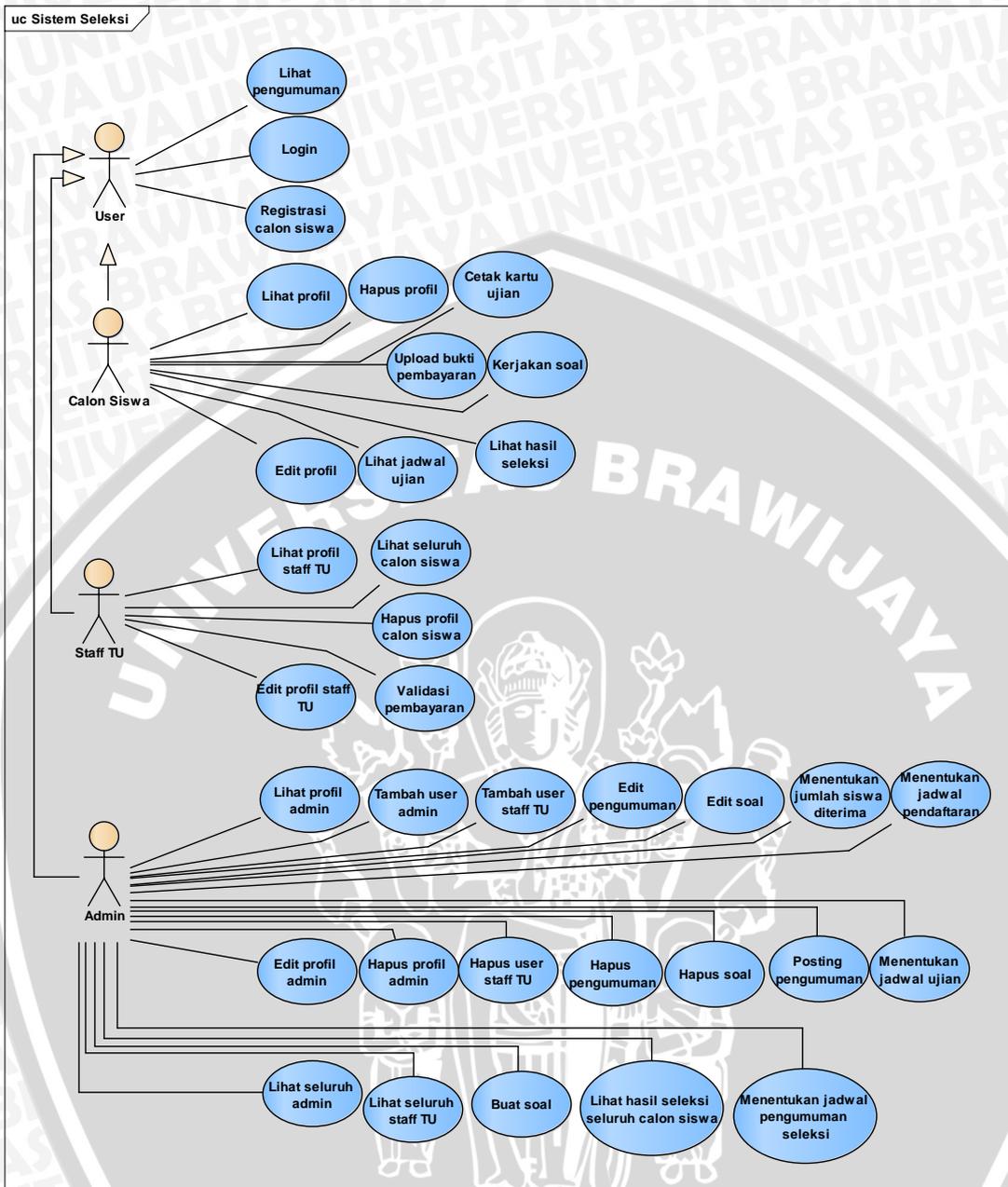
**Tabel 4.9 Pemetaan *Use Case* dengan Kebutuhan Fungsional**

Kode Kebutuhan	Kebutuhan	Kode Use Case	Nama Use Case	Aktor
FR-01	Registrasi calon siswa	UC-01	Registrasi	User
FR-02	Melihat pengumuman	UC-02	Lihat pengumuman	User
FR-03	Melihat profil	UC-03	Lihat profil	Calon Siswa
FR-04	Mengedit profil	UC-04	Edit profil	Calon Siswa
FR-05	Menghapus profil	UC-05	Hapus profil	Calon Siswa
FR-06	Mengupload bukti pembayaran	UC-06	Upload bukti pembayaran	Calon Siswa
FR-07	Melihat jadwal ujian	UC-07	Lihat jadwal ujian	Calon Siswa
FR-08	Mencetak kartu ujian	UC-08	Cetak kartu ujian	Calon Siswa
FR-09	Mengerjakan soal	UC-09	Kerjakan soal	Calon Siswa
FR-10	Melihat hasil seleksi	UC-10	Lihat hasil seleksi	Calon Siswa
FR-11	Melihat profil admin	UC-11	Lihat profil admin	Admin
FR-12	Melihat seluruh admin	UC-12	Lihat seluruh admin	Admin
FR-13	Menamambah user admin	UC-13	Tambah user admin	Admin
FR-14	Mengedit profil admin	UC-14	Edit profil admin	Admin
FR-15	Menghapus profil admin	UC-15	Hapus profil admin	Admin
FR-16	Melihat seluruh staff TU	UC-16	Lihat seluruh staff TU	Admin
FR-17	Menambah user staff TU	UC-17	Tambah user staff TU	Admin
FR-18	Menghapus user staff TU	UC-18	Hapus user staff TU	Admin
FR-19	Memposting pengumuman	UC-19	Posting pengumuman	Admin
FR-20	Mengedit pengumuman	UC-20	Edit pengumuman	Admin

Tabel 4.9 Pemetaan *Use Case* dengan Kebutuhan Fungsional (lanjutan)

Kode Kebutuhan	Kebutuhan	Kode Use Case	Nama Use Case	Aktor
FR-21	Menghapus pengumuman	UC-21	Hapus pengumuman	Admin
FR-22	Membuat soal	UC-22	Buat soal	Admin
FR-23	Mengedit soal	UC-23	Edit soal	Admin
FR-24	Menghapus soal	UC-24	Hapus soal	Admin
FR-25	Melihat hasil seleksi seluruh calon siswa	UC-25	Lihat hasil seleksi seluruh calon siswa	Admin
FR-26	Menentukan jumlah siswa diterima	UC-26	Menentukan jumlah siswa diterima	Admin
FR-27	Menentukan jadwal pendaftaran	UC-27	Menentukan jadwal pendaftaran	Admin
FR-28	Menentukan jadwal ujian	UC-28	Menentukan jadwal ujian	Admin
FR-29	Menentukan jadwal pengumuman seleksi	UC-29	Menentukan jadwal pengumuman seleksi	Admin
FR-30	Melihat profil staff TU	UC-30	Lihat profil staff TU	Staff TU
FR-31	Mengedit profil staff TU	UC-31	Edit profil staff TU	Staff TU
FR-32	Melihat seluruh calon siswa	UC-32	Lihat seluruh calon siswa	Staff TU
FR-33	Menghapus profil calon siswa	UC-33	Hapus profil calon siswa	Staff TU
FR-34	Memvalidasi pembayaran	UC-34	Validasi pembayaran	Staff TU
FR-35	Login	UC-35	Login	User
FR-36	Logout	UC-36	Logout	Calon Siswa, Staff TU, & Admin

Sistem ini memiliki beberapa kegiatan utama. Penggambaran kegiatan tersebut dapat dilihat dalam diagram *use case* pada gambar 4.8.



Gambar 4.8 Diagram Use Case Sistem Seleksi Penerimaan Siswa Baru SMA

#### 4.2.1.2 Skenario Use Case

Penjelasan *use case diagram* pada gambar 4.2 dapat dilihat pada tabel spesifikasi *use case* berikut,

##### 1. [UC-01] Registrasi

Tabel 4.10 Skenario Use Case Registrasi

Skenario Kasus pada Sistem	
Nomor FR	FR-01
Nama	Registrasi
Tujuan	Calon siswa dapat melakukan registrasi ke dalam sistem

Deskripsi	Use case ini menjelaskan bagaimana proses registrasi calon siswa
Aktor	User
<b>Skenario Utama</b>	
Kondisi Awal	Aktor membuka halaman Home.
<b>Aksi Aktor</b>	<b>Reaksi Sistem</b>
1. Aktor memilih menu Registrasi	2. Sistem akan menampilkan halaman registrasi
3. Aktor mengisi form registrasi yang terdiri dari NISN, nama lengkap, tempat lahir, tanggal lahir, alamat, jenis kelamin, nomor hp, email, nama ayah, pekerjaan ayah, nama ibu, pekerjaan ibu, asal sekolah, nilai rata-rata UN, ijazah, foto, dan password kemudian mengklik tombol Daftar.	4. Sistem akan mengecek NISN pada database.
	5. Sistem akan menyimpan data siswa di database
	6. Sistem memunculkan pesan bahwa aktor berhasil melakukan registrasi.
<b>Skenario Alternatif 1: Jika aktor memilih menu Registrasi sebelum pendaftaran dibuka</b>	
	2. Sistem akan menampilkan pesan bahwa pendaftaran belum dibuka
<b>Skenario Alternatif 2: Jika aktor memilih menu Registrasi setelah hari pendaftaran berakhir</b>	
	2. Sistem akan menampilkan pesan bahwa pendaftaran telah ditutup
<b>Skenario Alternatif 3: Jika input aktor tidak valid</b>	
	6. Sistem menampilkan notifikasi bahwa data yang dimasukkan tidak valid.
Kondisi Akhir	Aktor telah terdaftar dalam sistem sebagai calon siswa.

Pada Tabel 4.10 menggambarkan skenario dari use case Registrasi baru. Tujuan dari kebutuhan ini adalah agar calon siswa dapat melakukan registrasi ke dalam sistem. Skenario diawali dengan aktor memilih menu Registrasi. Sistem akan menampilkan halaman registrasi. Langkah selanjutnya adalah aktor mengisi form registrasi. Setelah form terisi, aktor mengklik tombol Daftar. Sistem akan mengecek NISN pada database.



Kemudian sistem akan menyimpan data calon siswa ke dalam database. Setelah itu, sistem akan menampilkan pesan bahwa aktor berhasil melakukan registrasi. Ada tiga skenario alternatif, yaitu jika aktor memilih menu Registrasi sebelum pendaftaran dibuka, maka sistem akan menampilkan pesan bahwa pendaftaran belum dibuka. Skenario alternatif kedua yaitu jika aktor memilih menu Registrasi setelah hari pendaftaran berakhir, maka sistem akan menampilkan pesan bahwa pendaftaran telah ditutup. Skenario alternatif yang ketiga adalah jika input aktor tidak valid, maka sistem akan menampilkan notifikasi bahwa data yang dimasukkan tidak valid. Kondisi akhir dari skenario ini adalah aktor telah terdaftar dalam sistem sebagai calon siswa.

2. [UC-09] Kerjakan soal

Tabel 4.11 Skenario Use Case Kerjakan Soal

Skenario Kasus pada Sistem	
Nomor FR	FR-09
Nama	Kerjakan soal
Tujuan	Calon siswa dapat mengerjakan soal ujian seleksi penerimaan siswa SMA baru
Deskripsi	Use case ini menjelaskan bagaimana proses calon siswa mengerjakan ujian seleksi
Aktor	Calon siswa
Skenario Utama	
Kondisi Awal	Aktor telah berhasil login ke sistem dan telah membuka halaman Home
Aksi Aktor	Reaksi Sistem
1. Calon siswa memilih menu Ujian.	2. Sistem menampilkan halaman yang berisi informasi mengenai ujian
3. Calon siswa mengklik tombol Mulai Ujian.	4. Sistem menampilkan halaman yang berisi soal-soal ujian.
5. Calon siswa menjawab soal ujian dengan batasan waktu. Setelah menjawab seluruh soal, calon siswa mengklik tombol Selesai Mengerjakan.	6. Sistem menyimpan jawaban dan menampilkan pesan bahwa calon siswa telah selesai mengerjakan ujian.
Skenario Alternatif 1: Jika sesi login tidak valid	
	2. Sistem menampilkan halaman login
Skenario Alternatif 1: Jika aktor belum mengklik tombol Selesai Mengerjakan saat waktu ujian habis.	
	6. Sistem menyimpan jawaban dan menampilkan pesan bahwa calon



	siswa telah selesai mengerjakan ujian.
Kondisi Akhir	Sistem mengkalkulasi nilai ujian dan disimpan ke dalam database.

Pada Tabel 4.11 menggambarkan skenario dari use case kerjakan soal. Tujuan dari kebutuhan ini adalah agar calon siswa dapat melakukan ujian seleksi penerimaan siswa SMA baru. Skenario diawali dengan aktor login ke dalam sistem. Kemudian aktor memilih menu Ujian dan sistem akan menampilkan halaman yang berisi informasi mengenai ujian. Kemudian aktor mengklik tombol Mulai Ujian. Sistem akan menampilkan halaman yang berisi soal soal ujian. Aktor menjawab soal ujian dengan batasan waktu. Apabila aktor telah selesai mengerjakan soal, aktor mengklik tombol Selesai Mengerjakan. Setelah itu, sistem akan menyimpan jawaban dan menampilkan pesan bahwa calon siswa telah selesai mengerjakan ujian. Ada dua skenario alternatif, yaitu jika sesi login aktor tidak valid maka sistem akan menampilkan halaman login sedangkan alternatif kedua yaitu jika aktor belum mengklik tombol Selesai Mengerjakan saat waktu ujian habis, maka sistem akan tetap menyimpan jawaban dan menampilkan pesan bahwa calon siswa telah selesai mengerjakan ujian. Kondisi akhir dari skenario ini adalah sistem berhasil mengkalkulasi nilai ujian dan disimpan ke dalam database.

### 3. [UC-19] Posting Pengumuman

Tabel 4.12 Skenario Use Case Posting Pengumuman

Skenario Kasus pada Sistem	
Nomor FR	FR-19
Nama	Posting Pengumuman
Tujuan	Admin dapat memposting pengumuman
Deskripsi	Use case ini menjelaskan bagaimana proses admin dapat memposting pengumuman
Aktor	Admin
Skenario Utama	
Kondisi Awal	Aktor telah berhasil <i>login</i> ke sistem dan telah membuka halaman Home
Aksi Aktor	Reaksi Sistem
1. Aktor memilih menu Pengumuman	2. Sistem akan menampilkan halaman yang berisi pengumuman pengumuman yang telah dibuat sebelumnya.
3. Aktor mengklik tombol Tambah Pengumuman.	4. Sistem menampilkan form untuk membuat pengumuman baru.



5. Aktor mengisi form membuat pengumuman baru berupa judul pengumuman dan isi pengumuman kemudian mengklik tombol Tambah.	6. Sistem menampilkan pesan bahwa pengumuman berhasil ditambahkan.
<b>Skenario Alternatif 1: Jika sesi login tidak valid</b>	
	2. Sistem menampilkan halaman login
Kondisi Akhir	Sistem berhasil menyimpan pengumuman yang telah dibuat

Pada Tabel 4.12 menggambarkan skenario dari use case posting pengumuman. Tujuan dari kebutuhan ini adalah agar admin dapat memposting pengumuman mengenai seleksi penerimaan siswa SMA baru. Skenario diawali dengan aktor login ke dalam sistem. Kemudian aktor memilih menu Pengumuman dan sistem akan menampilkan halaman yang berisi pengumuman-pengumuman yang telah dibuat sebelumnya. Kemudian aktor mengklik tombol Tambah Pengumuman. Setelah aktor mengklik tombol Tambah Pengumuman, sistem akan menampilkan form untuk membuat pengumuman baru. Setelah itu, aktor mengisi form membuat pengumuman baru berupa judul pengumuman dan isi pengumuman. Setelah mengisi form, aktor mengklik tombol tambah. Kemudian, sistem menampilkan pesan bahwa pengumuman berhasil ditambahkan. Ada satu skenario alternatif, yaitu apabila sesi login tidak valid, maka sistem akan menampilkan halaman login. Kondisi akhir dari skenario ini adalah sistem berhasil menyimpan pengumuman yang telah dibuat.

4. [UC-23] Edit soal

Tabel 4.13 Skenario Use Case Edit Soal

Skenario Kasus pada Sistem	
Nomor FR	FR-24
Nama	Edit soal
Tujuan	Agar admin dapat mengedit soal ujian apabila ada perubahan
Deskripsi	Use case ini menjelaskan bagaimana proses mengedit soal
Aktor	Admin
Skenario Utama	
Kondisi Awal	Aktor telah berhasil <i>login</i> ke sistem dan telah membuka halaman Home
Aksi Aktor	Reaksi Sistem
1. Aktor memilih menu Soal	2. Sistem akan menampilkan halaman yang berisi soal-soal ujian seleksi



3. Aktor menekan tombol Edit pada soal yang ingin diedit.	4. Sistem menampilkan form edit soal berupa soal, opsi jawaban, dan jawaban.
5. Aktor mengedit soal kemudian menekan tombol Simpan	6. Sistem menampilkan pesan bahwa soal berhasil diedit
<b>Skenario Alternatif 1: Jika sesi login tidak valid</b>	
	2. Sistem akan menampilkan halaman login
Kondisi Akhir	Soal, opsi jawaban, dan jawaban berhasil disimpan ke dalam sistem

Pada Tabel 4.13 menggambarkan skenario dari use case edit soal. Tujuan dari kebutuhan ini adalah agar admin dapat mengedit soal ujian. Skenario diawali dengan aktor login ke dalam sistem. Kemudian aktor memilih menu Soal dan sistem akan menampilkan halaman yang berisi akan menampilkan halaman yang berisi soal soal ujian seleksi. Kemudian aktor menekan tombol Edit pada soal yang ingin diedit. Setelah itu, sistem akan menampilkan form edit soal berupa soal, opsi jawaban, dan jawaban. Lalu aktor mengedit soal kemudian menekan tombol Simpan. Sistem akan menampilkan pesan bahwa soal berhasil diedit. Ada satu skenario alternatif, yaitu jika sesi login tidak valid, maka sistem akan menampilkan halaman login. Kondisi akhir dari skenario ini adalah soal, opsi jawaban, dan jawaban berhasil disimpan ke dalam sistem.

5. [UC-26] Menentukan jumlah siswa diterima

Tabel 4.14 Skenario Use Case Menentukan Jumlah Siswa Diterima

Skenario Kasus pada Sistem	
Nomor FR	FR-27
Nama	Menentukan jumlah siswa diterima
Tujuan	Agar admin dapat menentukan jumlah siswa yang diterima
Deskripsi	Use case ini menjelaskan bagaimana proses admin menentukan jumlah siswa diterima
Aktor	Admin
Skenario Utama	
Kondisi Awal	Aktor telah berhasil login ke sistem dan telah membuka halaman Home
Aksi Aktor	Reaksi Sistem
1. Aktor memilih menu Jadwal & Ujian	2. Sistem menampilkan halaman untuk mengatur jadwal dan ujian
3. Aktor memasukkan jumlah siswa yang diterima kemudian menekan tombol Simpan	4. Sistem akan memunculkan pesan bahwa jumlah siswa yang diterima berhasil dirubah
<b>Skenario Alternatif 1: Jika sesi login tidak valid</b>	

	2. Sistem akan menampilkan halaman login.
<b>Skenario Alternatif 2: Jika <i>input</i> aktor tidak valid</b>	
	4. Sistem akan menampilkan pesan bahwa jumlah siswa yang diterima harus lebih dari 0.
Kondisi Akhir	Sistem berhasil menentukan siswa yang diterima dan tidak diterima

Pada Tabel 4.14 menggambarkan skenario dari use case menentukan jumlah siswa diterima. Tujuan dari kebutuhan ini adalah agar admin dapat menentukan jumlah siswa yang diterima di SMA sebagai siswa baru. Skenario diawali dengan aktor login ke dalam sistem. Kemudian aktor memilih menu Jadwal & Ujian dan sistem akan menampilkan halaman untuk mengatur jadwal dan ujian. Setelah itu, aktor memasukkan jumlah siswa yang diterima kemudian menekan tombol Simpan. Apabila aktor telah menekan tombol Simpan, sistem akan memunculkan pesan bahwa jumlah siswa yang diterima berhasil dirubah. Ada dua skenario alternatif, yaitu jika sesi login tidak valid, maka sistem akan menampilkan halaman login. Sedangkan skenario alternatif kedua adalah jika *input* aktor tidak valid, maka sistem akan menampilkan pesan bahwa jumlah siswa yang diterima harus lebih dari 0. Kondisi akhir dari skenario ini adalah sistem berhasil menentukan siswa yang diterima dan tidak diterima.

#### 4.2.2 Pemodelan Diagram Sekuen

Diagram sekuen mempresentasikan model alir jalannya proses interaksi antar objek. Diagram sekuen dibuat berdasarkan *use case* diagram sistem. Pemetaan *use case* dan diagram sekuen dapat dilihat pada Tabel 4.15 berikut.

**Tabel 4.15 Pemetaan Use Case dengan Diagram Sekuen**

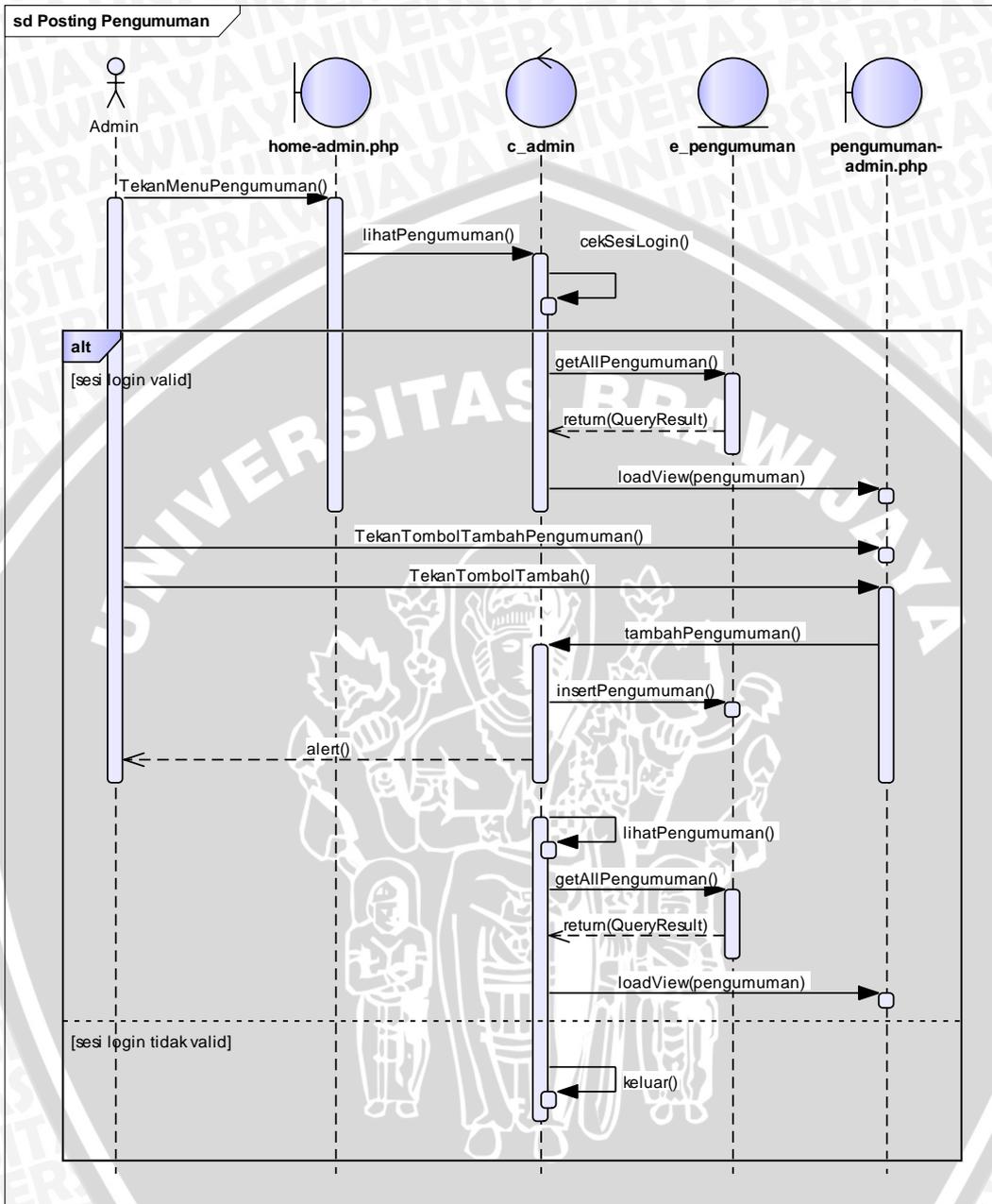
Kode Use Case	Nama Use Case	Kode Sequence	Nama Diagram Sekuen	Aktor
UC-01	Registrasi	SC-01	Registrasi	User
UC-02	Lihat pengumuman	SC-02	Lihat pengumuman	User
UC-03	Lihat profil	SC-03	Lihat profil	Calon Siswa
UC-04	Edit profil	SC-04	Edit profil	Calon Siswa
UC-05	Hapus profil	SC-05	Hapus profil	Calon Siswa
UC-06	Upload bukti pembayaran	SC-06	Upload bukti pembayaran	Calon Siswa
UC-07	Lihat jadwal ujian	SC-07	Lihat jadwal ujian	Calon Siswa
UC-08	Cetak kartu ujian	SC-08	Cetak kartu ujian	Calon Siswa
UC-09	Kerjakan soal	SC-09	Kerjakan soal	Calon Siswa
UC-10	Lihat hasil seleksi	SC-10	Lihat hasil seleksi	Calon Siswa
UC-11	Lihat profil admin	SC-11	Lihat profil admin	Admin
UC-12	Lihat seluruh admin	SC-12	Lihat seluruh admin	Admin
UC-13	Tambah user admin	SC-13	Tambah user admin	Admin
UC-14	Edit profil admin	SC-14	Edit profil admin	Admin
UC-15	Hapus profil admin	SC-15	Hapus profil admin	Admin

Tabel 4.15 Pemetaan *Use Case* dengan Diagram Sekuen (lanjutan)

Kode <i>Use Case</i>	Nama <i>Use Case</i>	Kode <i>Sequenc</i>	Nama Diagram Sekuen	Aktor
UC-16	Lihat seluruh staff TU	SC-16	Lihat seluruh staff TU	Admin
UC-17	Tambah user staff TU	SC-17	Tambah user staff TU	Admin
UC-18	Hapus user staff TU	SC-18	Hapus user staff TU	Admin
UC-19	Posting pengumuman	SC-19	Posting pengumuman	Admin
UC-20	Edit pengumuman	SC-20	Edit pengumuman	Admin
UC-21	Hapus pengumuman	SC-21	Hapus pengumuman	Admin
UC-22	Buat soal	SC-22	Buat soal	Admin
UC-23	Edit soal	SC-23	Edit soal	Admin
UC-24	Hapus soal	SC-24	Hapus soal	Admin
UC-25	Lihat hasil seleksi seluruh calon siswa	SC-25	Lihat hasil seleksi seluruh calon siswa	Admin
UC-26	Menentukan jumlah siswa diterima	SC-26	Menentukan jumlah siswa diterima	Admin
UC-27	Menentukan jadwal pendaftaran	SC-27	Menentukan jadwal pendaftaran	Admin
UC-28	Menentukan jadwal ujian	SC-28	Menentukan jadwal ujian	Admin
UC-29	Menentukan jadwal pengumuman seleksi	SC-29	Menentukan jadwal pengumuman seleksi	Admin
UC-30	Lihat profil staff TU	SC-30	Lihat profil staff TU	Staff TU
UC-31	Edit profil staff TU	SC-31	Edit profil staff TU	Staff TU
UC-32	Lihat seluruh calon siswa	SC-32	Lihat seluruh calon siswa	Staff TU
UC-33	Hapus profil calon siswa	SC-33	Hapus profil calon siswa	Staff TU
UC-34	Validasi pembayaran	SC-34	Validasi pembayaran	Staff TU
UC-35	Login	SC-35	Login	User
UC-36	Logout	SC-36	Logout	Calon Siswa, Staff TU, & Admin

Dari pemetaan tersebut maka dapat dimodelkan dalam diagram sekuen. Berikut beberapa diagram sekuen pada Sistem Seleksi Penerimaan Siswa Baru SMA.

### 1. [SC-19] Posting Pengumuman



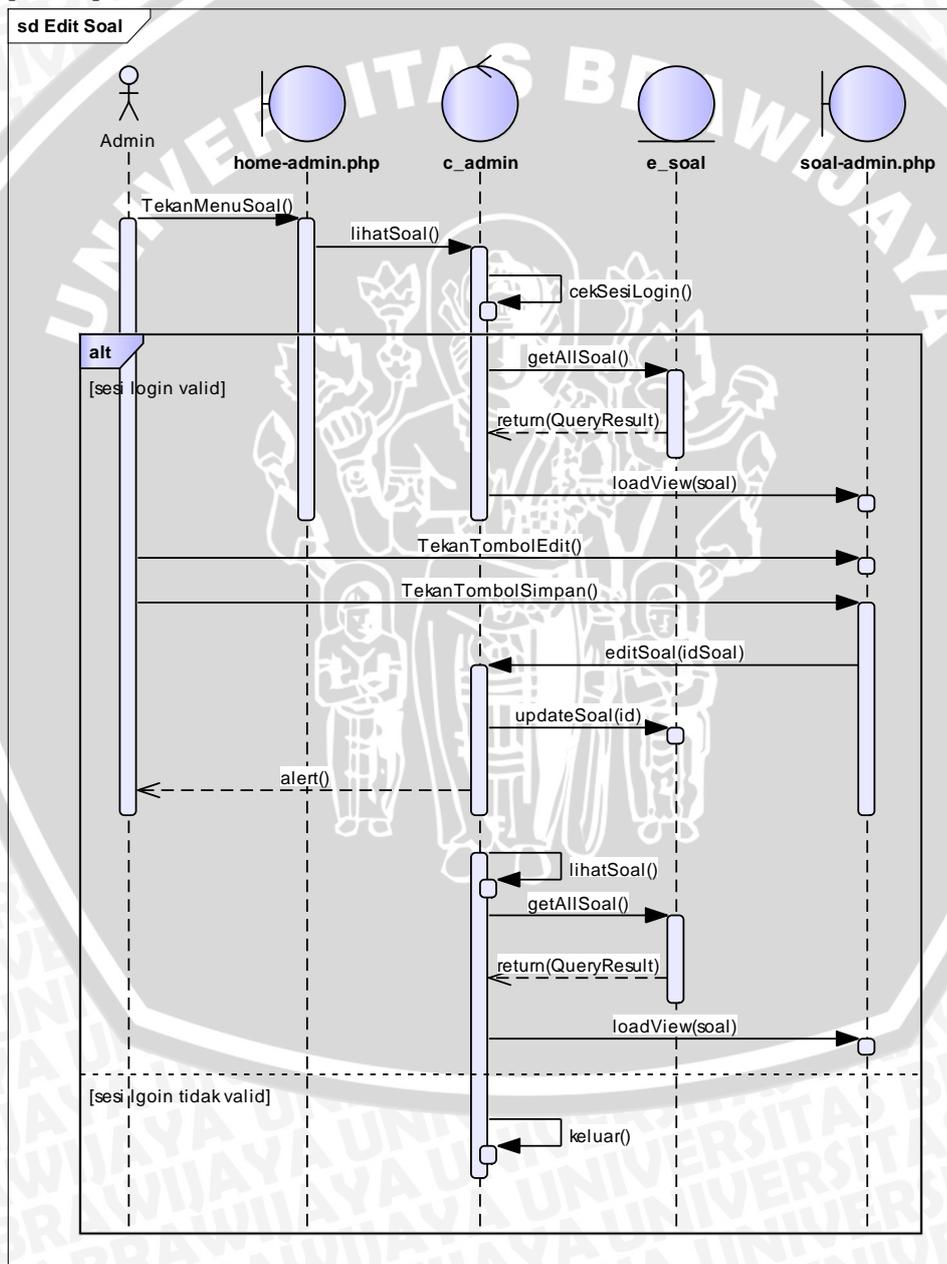
Gambar 4.9 Diagram Sekuen Posting Pengumuman

Diagram sekuen posting pengumuman pada gambar 4.9 diawali aktor admin menekan memilih menu Pengumuman di boundary home-admin.php. Selanjutnya sistem akan mendirect ke fungsi lihatPengumuman() pada objek c\_admin sebagai controller. Setelah itu c\_admin akan memanggil fungsi cekSesiLogin() untuk mengecek validasi user apakah sesi login valid. Jika sesi login tidak valid maka c\_admin akan memanggil fungsi keluar dan sistem akan menampilkan halaman login. Tetapi jika sesi login valid maka sistem akan meneruskan ke alur selanjutnya. Setelah itu c\_admin akan memanggil fungsi getAllPengumuman(). Fungsi getAllPengumuman() akan mengembalikan nilai



kepada objek `c_admin`. Kemudian memanggil boundary pengumuman-`admin.php`. Admin menekan tombol Tambah Pengumuman, kemudian menekan tombol Tambah. Selanjutnya sistem akan mendirect ke fungsi `insertPengumuman` pada objek `c_admin`. Kemudian sistem memberikan alert atau pesan kepada admin bahwa pengumuman berhasil ditambahkan. Kemudian objek `c_admin` memanggil fungsi `lihatPengumuman()`. Fungsi `lihatPengumuman()` akan memanggil fungsi `getAllPengumuman()` dan akan mengembalikan nilai kepada objek `c_admin`. Setelah itu objek akan memanggil boundary `pengumuman-admin.php`.

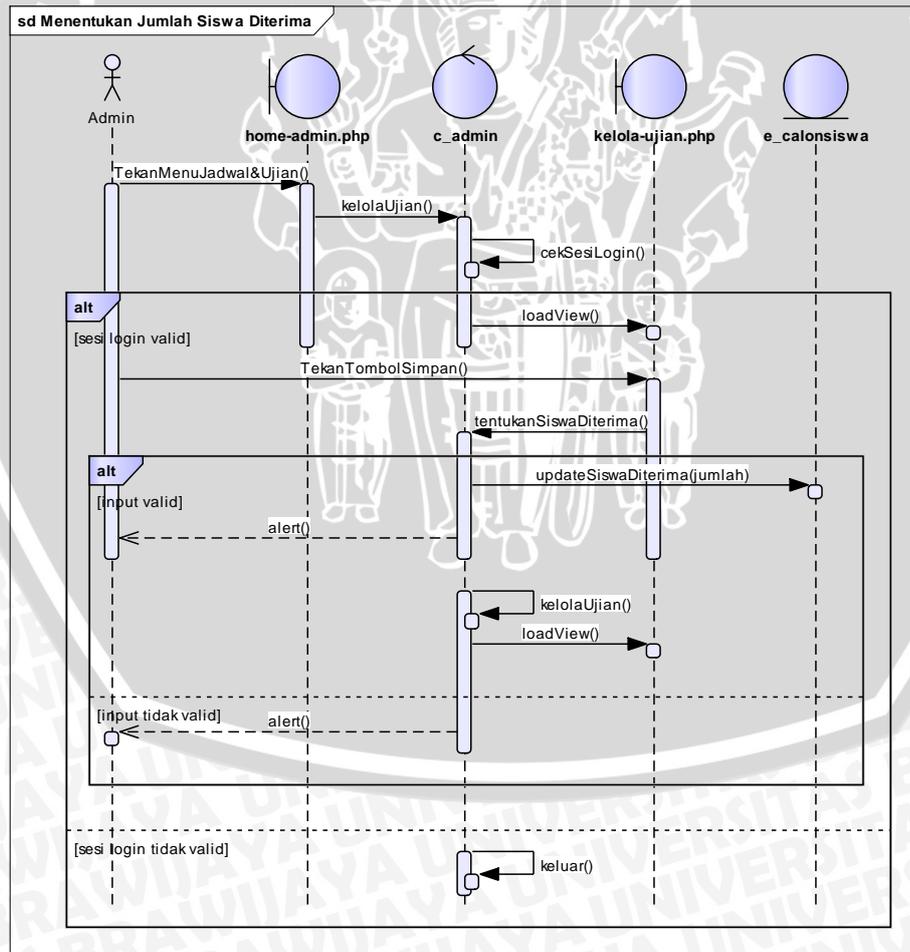
## 2. [SC-23] Edit Soal



Gambar 4.10 Diagram Sekuen Edit Soal

Diagram sekuen edit soal pada gambar 4.10 diawali aktor admin menekan memilih menu Soal di boundary home-admin.php. Selanjutnya sistem akan mendirect ke fungsi lihatSoal() pada objek c\_admin sebagai controller. Kemudian objek c\_admin memanggil fungsi cekSesiLogin() untuk mengecek validasi user apakah sesi login valid. Jika sesi login tidak valid maka c\_admin akan memanggil fungsi keluar() dan menampilkan halaman login. Tetapi jika sesi login valid maka sistem akan meneruskan ke alur selanjutnya. Setelah itu c\_admin akan memanggil fungsi getAllSoal(). Fungsi getAllSoal() akan mengembalikan nilai kepada objek c\_admin. Kemudian memanggil boundary soal-admin.php. Admin menekan tombol Edit, kemudian menekan tombol Simpan. Selanjutnya sistem akan mendirect ke fungsi editSoal(idSoal) pada objek c\_admin. Kemudian objek c\_admin memanggil fungsi updateSoal(id). Admin akan mendapatkan alert atau pesan bahwa soal berhasil diedit. Kemudian objek c\_admin memanggil fungsi lihatSoal(). Setelah itu c\_admin akan memanggil fungsi getAllSoal(). Fungsi getAllSoal() akan mengembalikan nilai kepada objek c\_admin. Kemudian memanggil boundary soal-admin.php.

### 3. [SC-26] Menentukan Jumlah Siswa Diterima



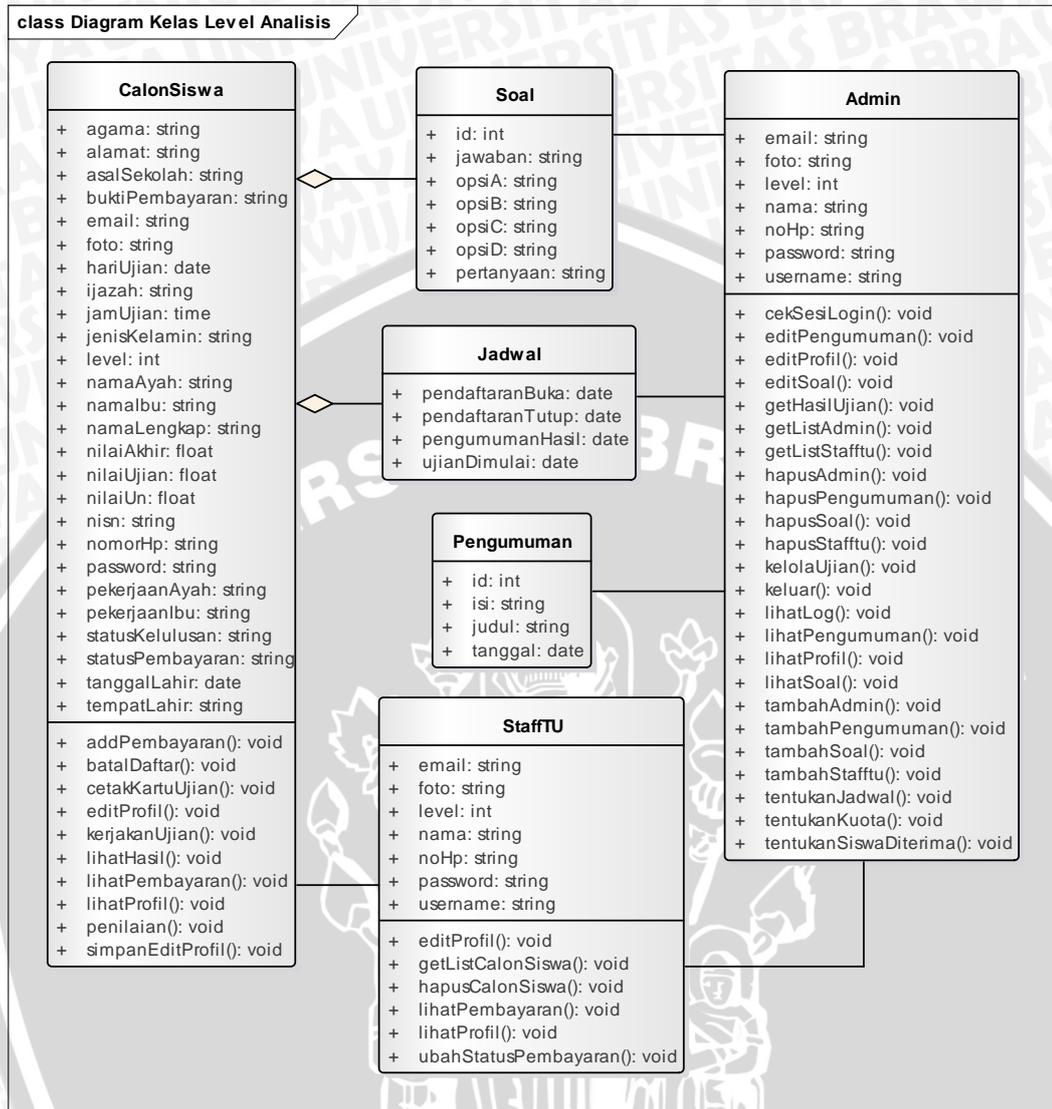
Gambar 4.11 Menentukan Jumlah Siswa Diterima



Diagram sekuen menentukan jumlah siswa diterima pada gambar 4.11 diawali aktor admin menekan memilih menu Jadwal & Ujian di boundary home-admin.php. Selanjutnya sistem akan mendirect ke fungsi kelolaUjian() pada objek c\_admin sebagai controller. Kemudian objek c\_admin memanggil fungsi cekSesiLogin() untuk mengecek validasi user apakah sesi login valid. Jika sesi login tidak valid maka c\_admin akan memanggil fungsi keluar() kemudian menampilkan halaman login. Tetapi jika sesi login valid maka sistem akan meneruskan ke alur selanjutnya. Setelah itu c\_admin akan memanggil boundary keloaja-ujian.php. Setelah itu, admin menekan tombol Simpan. Sistem akan memanggil fungsi tentukanSiswaDiterima(). Apabila input admin tidak valid, maka sistem akan menampilkan pesan bahwa input tidak valid. Sedangkan, apabila input admin valid, maka akan meneruskan ke alur selanjutnya. Setelah itu, objek c\_admin akan memanggil fungsi updateSiswaDiterima(jumlah). Admin akan mendapatkan alert atau pesan bahwa jumlah siswa yang diterima berhasil dirubah. Selanjutnya, objek c\_admin akan memanggil fungsi kelolaUjian(). Kemudian, memanggil boundary kelola-ujian.php.



### 4.2.3 Pemodelan Diagram Kelas



Gambar 4.12 Analisis Diagram Kelas

Dari hasil analisis yang dilakukan pada gambar 4.12, kelas yang dibuat antara lain CalonSiswa, Admin, StaffTU, Soal, Jadwal, dan Pengumuman. Kelas CalonSiswa dan kelas Soal mempunyai hubungan agregasi. Sedangkan kelas Calon Siswa dan kelas Jadwal mempunyai hubungan yang sama juga, yaitu agregasi.

## BAB 5 PERANCANGAN DAN IMPLEMENTASI

Bab ini membahas perancangan dan implementasi berdasarkan hasil analisis kebutuhan Sistem Seleksi Penerimaan Siswa Baru SMA. Proses perancangan sistem yang dilakukan meliputi perancangan arsitektur sistem, pemodelan kelas, pemodelan data, perancangan komponen, dan perancangan antarmuka. Implementasi sistem terdiri dari spesifikasi lingkungan implementasi, batasan implementasi, implementasi program, dan implementasi antarmuka. Tahapan-tahapan pada proses perancangan dan implementasi dapat dilihat pada diagram 5.1.

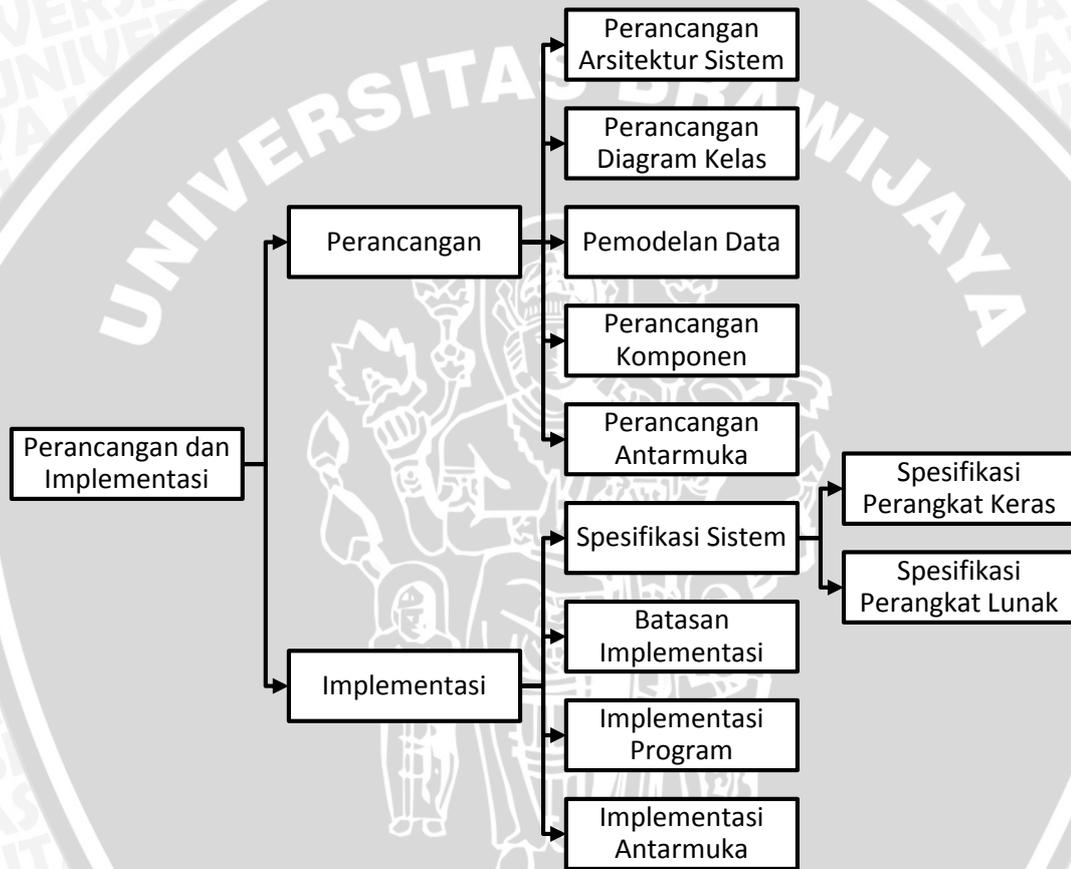


Diagram 5.1 Perancangan dan Implementasi

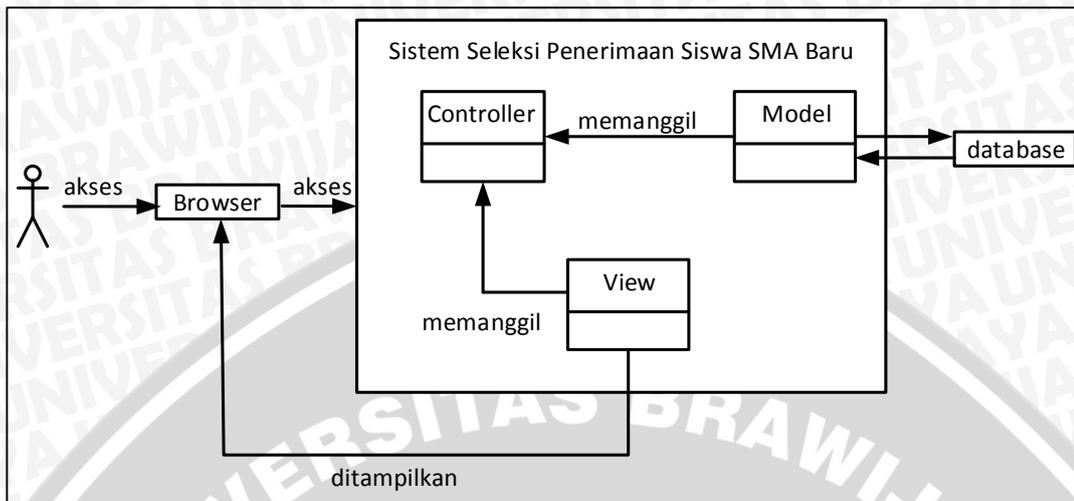
### 5.1 Perancangan Sistem

Proses perancangan sistem yang dilakukan meliputi perancangan arsitektur sistem, perancangan diagram kelas, perancangan basis data, dan perancangan antarmuka.

#### 5.1.1 Perancangan Arsitektur Sistem

Perancangan arsitektur sistem dilakukan untuk menggambarkan jalannya dari awal, yakni dari proses pengguna mengakses sistem, pemrosesan data, hingga

menghasilkan output bagi pengguna. Perancangan arsitektur Sistem Seleksi Penerimaan Siswa Baru SMA dapat dilihat pada gambar 5.1.



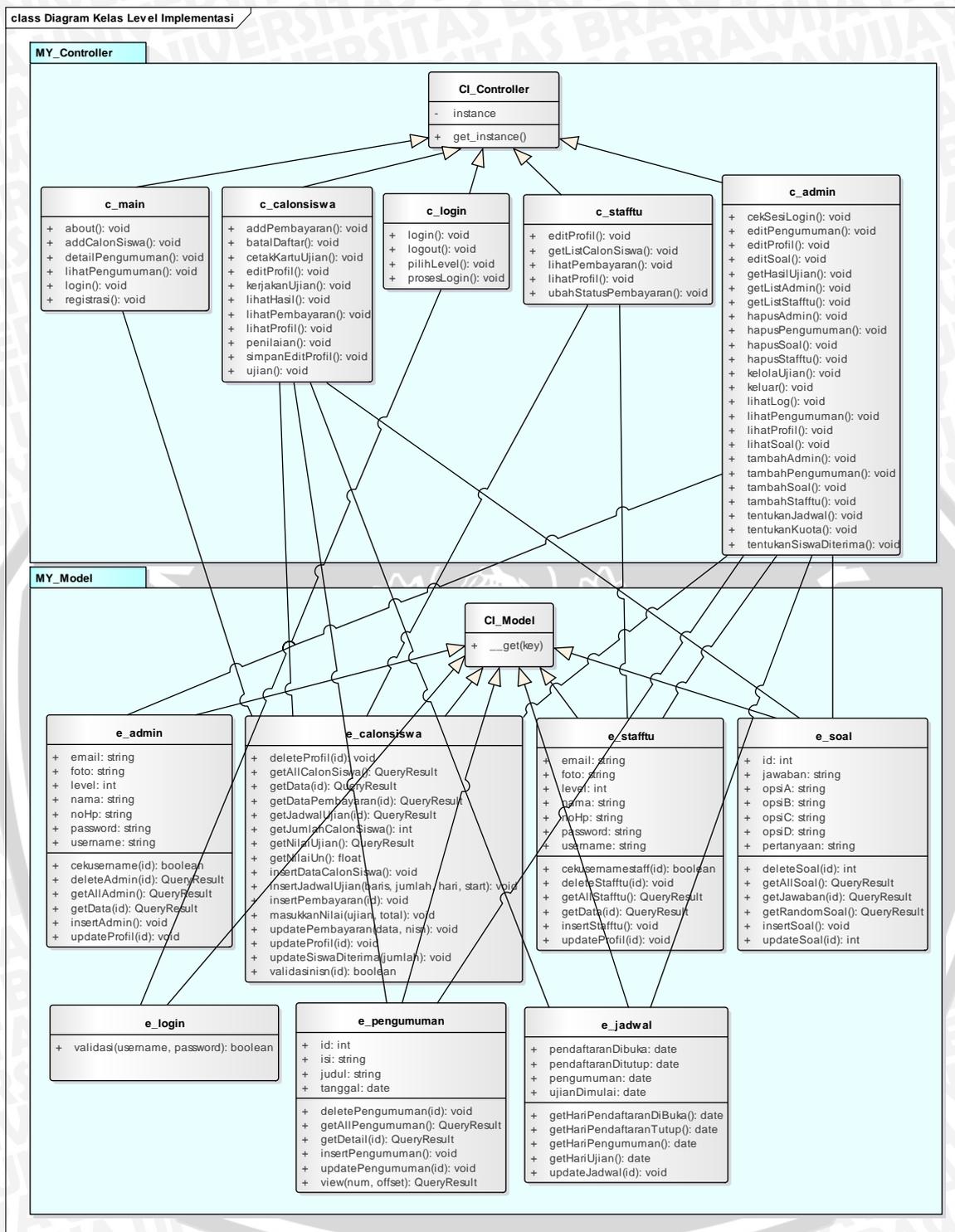
**Gambar 5.1 Arsitektur Sistem Seleksi Penerimaan Siswa Baru SMA**

Sistem Seleksi ini dapat dijalankan oleh aktor melalui beberapa proses yaitu awalnya aktor akan mengakses halaman browser untuk menampilkan Sistem Seleksi Penerimaan Siswa Baru SMA. Sistem Seleksi ini menggunakan *framework* CodeIgniter, sehingga browser akan mengakses sistem sesuai struktur file pada CodeIgniter yang menggunakan MVC (Model View Controller). Dalam CI, controller akan memanggil model yang berisi fungsi-fungsi untuk pengolahan data, diantaranya proses CRUD (*Create, Read, Update, Delete*). Data yang telah dipanggil di dalam model akan ditampilkan oleh view di dalam browser.

### 5.1.2 Perancangan Diagram Kelas

Diagram kelas menggambarkan sistem dan relasi relasi yang ada di dalamnya. Suatu kelas dalam sistem memiliki relasi dengan kelas lainnya dan dapat memiliki pewarisan (*inheritance*) atau ketergantungan (*dependency*) dengan komponen yang dimilikinya.

Sistem ini menggunakan MVC (*Model, View, Controller*) sehingga merepresentasikan ketiga komponen tersebut. Diagram kelas Sistem Seleksi Penerimaan Siswa Baru SMA dapat dilihat pada gambar 5.2.



Gambar 5.2 Diagram Kelas Sistem Seleksi Penerimaan Siswa Baru SMA

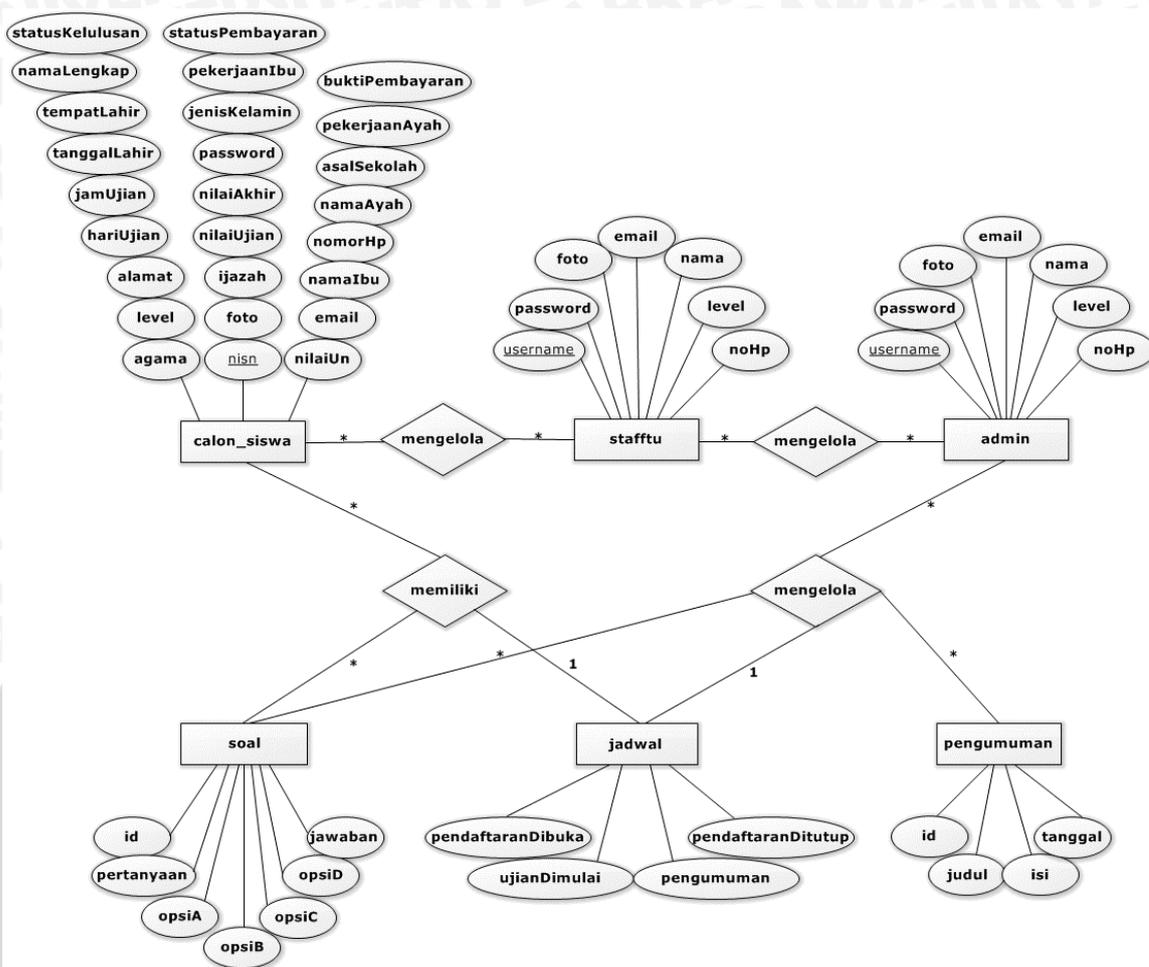
Berdasarkan gambar 5.2 diagram kelas Sistem Seleksi Penerimaan Siswa Baru SMA terdiri dari dua *package* kelas, yaitu MY\_Controller dan MY\_Model. Pada *package* MY\_Controller yang terdiri dari kelas-kelas controller yang semuanya generalisasi atau mewarisi dari kelas CI\_Controller. Kelas-kelas pada *package* MY\_Controller berfungsi penghubung dan pengontrol dari data dalam

model dengan tampilan pada view. Kelas controller yakni kelas CI\_Controller memiliki anak kelas c\_main berfungsi untuk manajemen fitur yang dapat digunakan sebelum user login, kelas c\_calonsiswa berfungsi untuk manajemen fitur yang dimiliki oleh calon siswa, kelas c\_login berfungsi untuk keperluan operasi ketika user ingin masuk ke dalam sistem, kelas c\_stafftu berfungsi untuk manajemen fitur yang dimiliki oleh staff TU, dan kelas c\_admin berfungsi untuk manajemen fitur yang dimiliki oleh admin.

*Package* selanjutnya adalah *package* MY\_Model. *Package* ini terdiri dari kelas-kelas model yang semuanya generalisasi atau mewarisi dari kelas CI\_Model. Kelas-kelas pada *package* MY\_Model berfungsi untuk operasi data pada database baik memasukkan data ke database, mengambil data dari database, ataupun menghapus data dari database. Kelas model yakni kelas CI\_Model memiliki anak kelas e\_admin yang berfungsi untuk pengelolaan data admin, kelas e\_calonsiswa yang berfungsi untuk pengelolaan data calon siswa, kelas e\_stafftu yang berfungsi untuk pengelolaan data staff TU, kelas e\_soal yang berfungsi untuk pengelolaan data soal ujian, kelas e\_login yang berfungsi untuk validasi user ketika ingin login ke sistem, kelas e\_pengumuman yang berfungsi untuk pengelolaan data pengumuman, kelas e\_jadwal yang berfungsi untuk pengelolaan data jadwal.

### 5.1.3 Pemodelan Data

Pemodelan data bertujuan merancang manajemen data yang akan digunakan dalam sistem. Manajemen data mencakup pemodelan basis data yang terdiri dari data relevan yang akan digunakan dalam sistem dan didukung oleh *Database Management System* (DBMS). Pemodelan data dalam sistem ini digambarkan dalam *Entity Relationship Diagram* (ERD). ERD Sistem Seleksi Penerimaan Siswa Baru SMA dapat dilihat pada gambar 5.3.

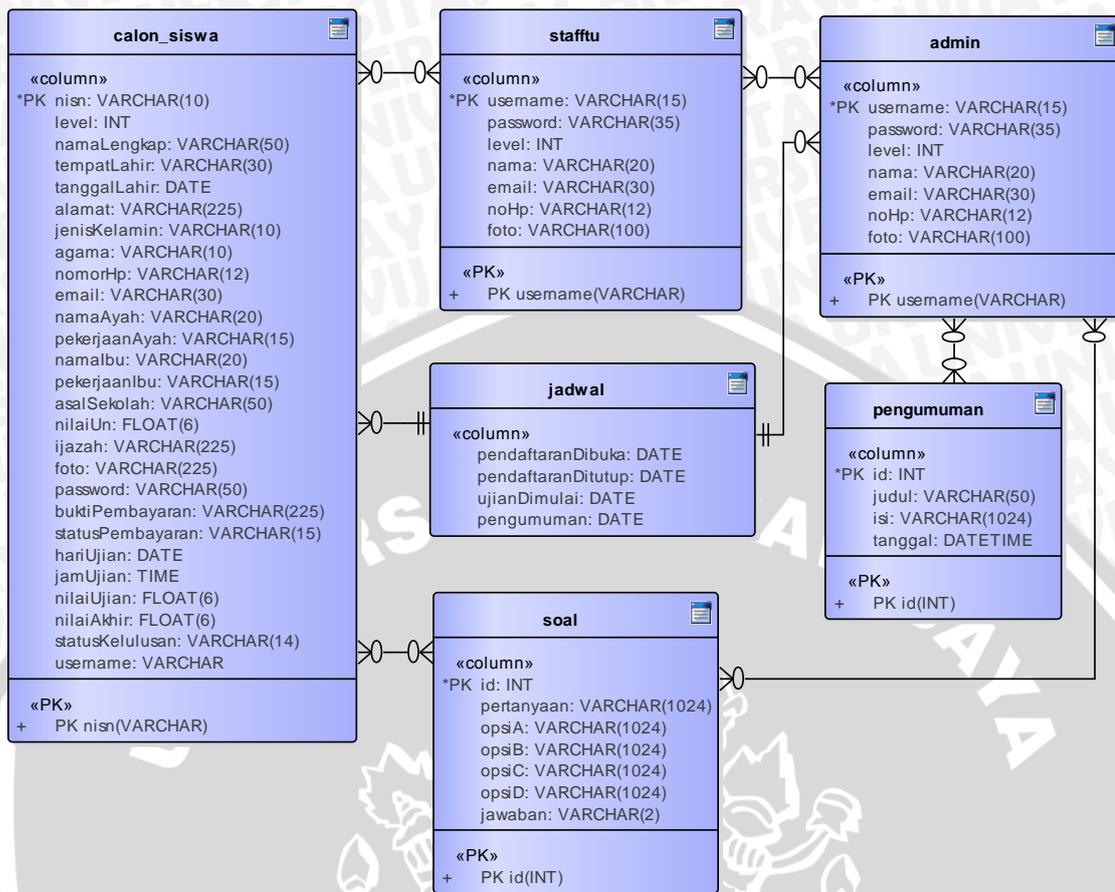


Gambar 5.3 ERD Sistem Seleksi Penerimaan Siswa Baru SMA

Pada gambar 5.7 calon\_siswa dengan soal dan jadwal mempunyai relasi, yaitu calon siswa memiliki soal dan jadwal. Admin mempunyai relasi dengan stafftu, soal, jadwal, dan pengumuman dimana admin akan mengelola stafftu, soal, jadwal, dan pengumuman. Sedangkan stafftu mempunyai relasi dengan calon\_siswa, yaitu stafftu mengelola calon\_siswa.

### 5.1.3.1 Pemodelan Database

Berdasarkan dari ERD diagram pada maka dapat dimodelkan dalam database. Pemodelan database Sistem Seleksi Penerimaan Siswa Baru SMA dapat dilihat pada gambar 5.4.



**Gambar 5.4 Pemodelan Database Sistem Seleksi Penerimaan Siswa Baru SMA**

Pada gambar 5.8 tabel calon\_siswa merupakan tabel yang berisi pengelolaan data dari calon siswa. Tabel stafftu berisi data dari staff tu. Tabel admin berisi data dari admin. Tabel soal berisi data dari soal. Tabel jadwal berisi data hari pendaftaran dibuka, pendaftaran ditutup, ujian dimulai, dan pengumuman penerimaan siswa SMA baru. Tabel pengumuman merupakan tabel yang berisi data pengumuman mengenai seleksi penerimaan siswa SMA baru.

**5.1.3.2 Struktur Tabel**

Berdasarkan rancangan database diatas, berikut struktur tabel yang terdapat pada database sistem dan penjelasannya dapat dilihat pada Tabel 5.1, Tabel 5.2, dan Tabel 5.3 berikut.

**1. Tabel calon\_siswa**

**Tabel 5.1 Atribut Data Calon Siswa**

Field	Type	Deskripsi	Keterangan
nisn	varchar(10)	Nomor Induk Siswa Nasional	PK
level	int(1)	Level user	
namaLengkap	varchar(50)	Nama lengkap calon siswa	
tempatLahir	varchar(30)	Tempat lahir calon siswa	
tanggalLahir	date	Tanggal lahir calon siswa	



**Tabel 5.1 Atribut Data Calon Siswa (lanjutan)**

Field	Type	Deskripsi	Keterangan
alamat	varchar(225)	Alamat rumah calon siswa	
jenisKelamin	varchar(10)	Jenis kelamin calon siswa	
agama	varchar(10)	Agama calon siswa	
nomorHp	varchar(12)	Nomor hp calon siswa	
email	varchar(30)	Email calon siswa	
namaAyah	varchar(20)	Nama ayah calon siswa	
pekerjaanAyah	varchar(15)	Pekerjaan ayah	
namaIbu	varchar(20)	Nama ibu calon siswa	
pekerjaanIbu	varchar(15)	Pekerjaan ibu	
asalSekolah	varchar(20)	Asal sekolah calon siswa	
nilaiUn	float	Nilai rata-rata ujian nasional	
ijazah	varchar(225)	File gambar ijazah	
foto	varchar(225)	Pas foto 3x4 calon siswa	
password	varchar(50)	Password untuk masuk sistem	
buktiPembayaran	varchar(225)	Slip pembayaran	
statusPembayaran	varchar(15)	Status pembayaran	
hariUjian	date	Hari melaksanakan ujian	
jamUjian	time	Jam melaksanakan ujian	
nilaiUjian	float	Nilai ujian seleksi	
nilaiAkhir	float	Nilai akhir	
statusKelulusan	varchar(14)	Status kelulusan	

2. Tabel admin

**Tabel 5.2 Atribut Data Admin**

Field	Type	Deskripsi	Keterangan
username	varchar(15)	Username admin	PK
password	varchar(35)	Password untuk masuk sistem	
level	int(1)	Level user	
nama	varchar(20)	Nama admin	
email	varchar(30)	Email admin	
noHp	varchar(12)	Nomor hp admin	
foto	varchar(100)	File foto admin	

3. Tabel stafftu

**Tabel 5.3 Atribut Data Staff TU**

Field	Type	Deskripsi	Keterangan
username	varchar(15)	Username staff TU	PK

**Tabel 5.4 Atribut Data Staff TU (lanjutan)**

Field	Type	Deskripsi	Keterangan
password	varchar(35)	Password untuk masuk sistem	
level	int(1)	Level user	
nama	varchar(20)	Nama staff TU	
email	varchar(30)	Email staff TU	
noHp	varchar(12)	Nomor hp staff TU	
foto	varchar(100)	File foto staff TU	

#### 5.1.4 Perancangan Komponen

Perancangan komponen adalah suatu kegiatan untuk mengambil setiap komponen dalam sistem tentang bagaimana komponen tersebut akan berjalan (Sommerville, 2011). Model desain dapat digunakan untuk mempermudah proses implementasi oleh programmer. Perancangan komponen berisi algoritma-algoritma yang akan diimplementasikan pada kelas-kelas yang didefinisikan. Algoritma yang dituliskan merupakan pseudocode yang menjembatani bahasa manusia dengan kode program. Dalam menuliskan pseudocode, standart yang digunakan merujuk pada web users.csc.calpoly.ed yang ditulis oleh Dr. John Dalbey. Dalam perancangan komponen Sistem Seleksi Penerimaan Siswa Baru SMA, sampel yang diambil yaitu algoritma fungsi kerjakan ujian, lihat hasil seleksi, dan menentukan jumlah siswa diterima.

##### 5.1.4.1 Algoritma Kerjakan Ujian

Algoritma kerjakan ujian ini merupakan algoritma untuk mengerjakan soal-soal saat ujian seleksi. Gambar 5.5 menjelaskan algoritma kerjakan ujian.

<b>Fungsi Kerjakan Ujian</b>
Ambil nilai ujian
Jika nilai ujian tidak ada
Mengambil soal secara acak
Menampilkan halaman ujian dengan soal yang diacak
Jika tidak
Menampilkan notifikasi bahwa calon siswa telah melaksanakan ujian

**Gambar 5.5 Algoritma Kerjakan Ujian**

##### 5.1.4.2 Algoritma Lihat Hasil Seleksi

Algoritma lihat hasil seleksi ini berfungsi untuk melihat hasil seleksi apakah calon siswa diterima atau tidak setelah mengikuti ujian seleksi. Gambar 5.6 menjelaskan algoritma lihat hasil seleksi.

<b>Fungsi Lihat Hasil Seleksi</b>
Menetapkan asia/jakarta sebagai zona waktu
Panggil fungsi getHariPengumuman yang hasilnya dimasukkan ke dalam variabel haripengumuman

<p>Mendeklarasikan variabel sekarang yang berisi tanggal hari ini</p> <p>Jika sekarang variabel sekarang lebih dari sama dengan variabel hari pengumuman</p> <p>Variabel id sama dengan nisn</p> <p>Mengambil data calon siswa berdasarkan nisn</p> <p>Menampilkan halaman hasil seleksi</p> <p>Jika tidak</p> <p>Menampilkan notifikasi bahwa pengumuman hasil seleksi belum ada</p>
---

**Gambar 5.6 Algoritma Lihat Hasil Seleksi**

#### 5.1.4.3 Algoritma Menentukan Jumlah Siswa Diterima

Algoritma menentukan jumlah siswa diterima ini berfungsi untuk jumlah siswa yang diterima di SMA sebagai siswa baru berdasarkan nilai ujian nasional dan nilai ujian seleksi. Gambar 5.7 menjelaskan algoritma menentukan jumlah siswa diterima.

##### Fungsi Menentukan Jumlah Siswa Diterima

<p>Jika masukan jumlah lebih dari 0</p> <p>Variabel jumlah sama dengan masukan jumlah</p> <p>Memanggil fungsi updateSiswaDiterima dengan parameter variabel jumlah</p> <p>Menampilkan notifikasi bahwa jumlah siswa yang diterima berhasil diubah</p> <p>Jika tidak</p> <p>Menampilkan pesan bahwa jumlah siswa yang diterima harus lebih dari 0</p>
--

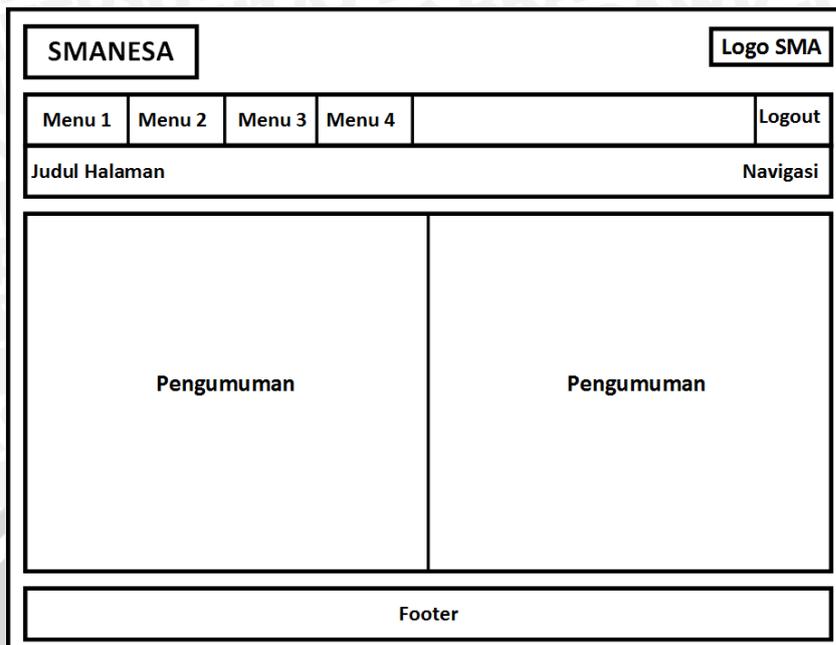
**Gambar 5.7 Algoritma Menentukan Jumlah Siswa Diterima**

#### 5.1.5 Perancangan Antarmuka

Proses ini bertujuan untuk menghasilkan antarmuka sistem agar *output* yang dihasilkan sistem dapat diterima dengan baik oleh pengguna. *Output* yang dihasilkan diantaranya adalah pengumuman mengenai seleksi, soal ujian yang dikerjakan oleh pendaftar saat ujian seleksi, dan hasil seleksi penerimaan siswa SMA baru. Berikut adalah rancangan antarmuka halaman-halaman pada Sistem Seleksi Penerimaan Siswa Baru SMA.

##### 5.1.5.1 Antarmuka Halaman Lihat Pengumuman

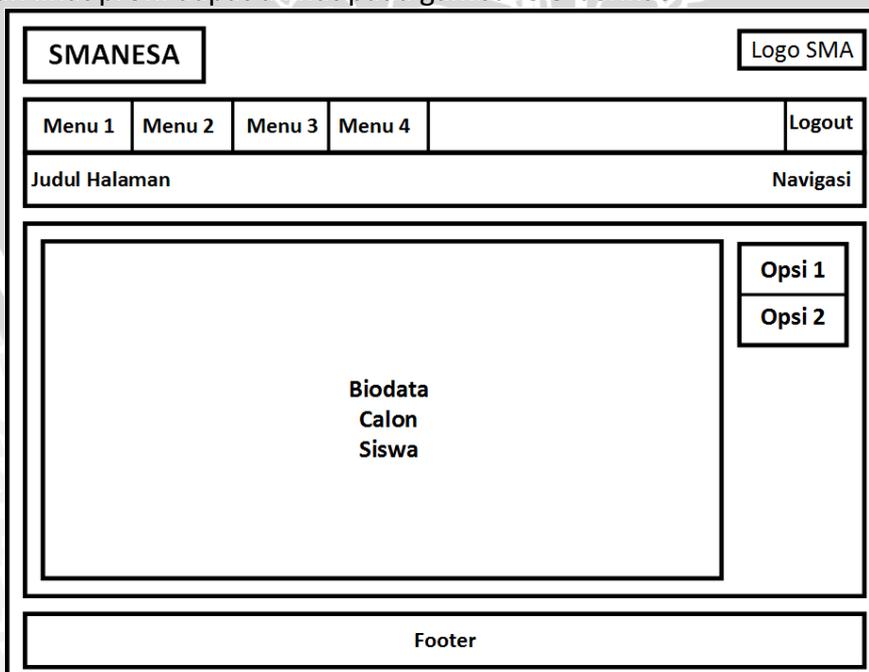
Halaman ini merupakan halaman untuk melihat pengumuman mengenai seleksi penerimaan siswa SMA baru yang telah diposting oleh admin. Pengumuman ditampilkan dalam 2 kolom. Tombol-tombol yang ada pada halaman ini adalah tombol menu, login, dan baca selengkapnya. Rancangan antarmuka halaman lihat pengumuman dapat dilihat pada gambar 5.8 berikut.



Gambar 5.8 Antarmuka Halaman Lihat Pengumuman

### 5.1.5.2 Antarmuka Halaman Lihat Profil

Halaman ini merupakan halaman untuk melihat profil atau biodata calon siswa yang telah melakukan registrasi ke dalam sistem. Pada halaman ini akan menampilkan data-data dari calon siswa seperti nish, nama lengkap, tempat lahir, tanggal lahir, sampai foto ijazah. Tombol-tombol yang ada pada halaman ini antara lain : menu, logout, edit biodata, dan batal mendaftar. Rancangan antarmuka halaman lihat profil dapat dilihat pada gambar 5.9 berikut.



Gambar 5.9 Antarmuka Halaman Lihat Profil

### 5.1.5.3 Antarmuka Halaman Lihat Seluruh Admin

Halaman ini merupakan halaman untuk melihat seluruh admin. Pada halaman ini akan menampilkan data-data seluruh admin seperti foto, username, nama, email, dan nomor hp. Tombol-tombol yang ada pada halaman ini antara lain: menu, logout, navigasi, tambah admin, dan hapus admin. Rancangan antarmuka halaman lihat profil dapat dilihat pada gambar 5.10 berikut.

The screenshot shows a web interface for viewing all administrators. It features a sidebar menu with six items labeled 'Menu 1' through 'Menu 6'. The main content area is titled 'Navigasi' and contains a 'Judul Halaman' section with a 'Tombol' button. Below this is a table with six columns: 'Label 1', 'Label 2', 'Label 3', 'Label 4', 'Label 5', and 'Aksi'. The table is currently empty. At the top left is a 'Logo SMA' and at the top right is a 'Foto' button.

Gambar 5.10 Antarmuka Halaman Login

### 5.1.5.4 Antarmuka Halaman Tambah Admin

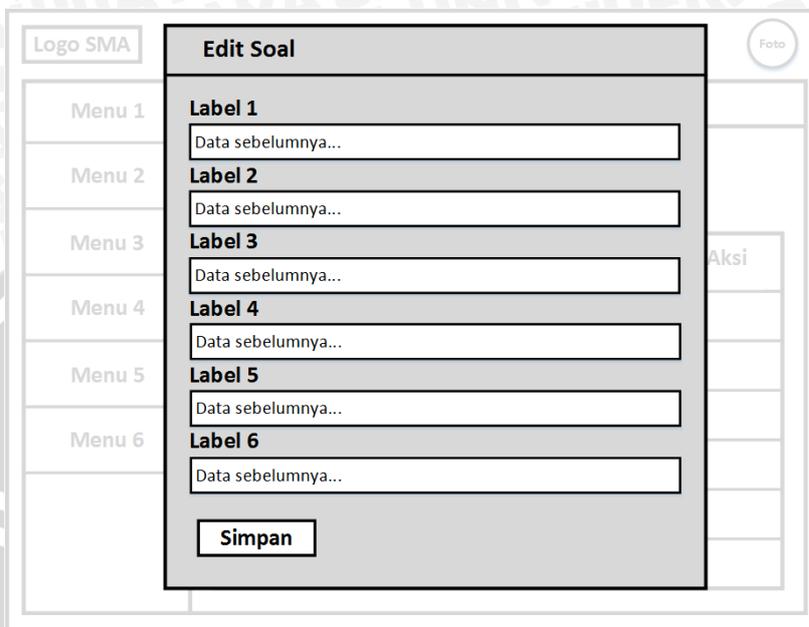
Halaman ini merupakan halaman atau form untuk menambahkan admin. Form ini adalah modals atau pop up. Dalam modals ini terdapat data yang harus diisi antara lain: username, nama lengkap, email, password, nomor hp, dan foto. Rancangan antarmuka halaman tambah admin dapat dilihat pada gambar 5.11 berikut.

The screenshot shows a modal form for adding a new administrator. The form is titled 'Tambah Admin' and is overlaid on a background that includes the same menu as seen in the previous image. The form contains six input fields, each preceded by a label: 'Label 1', 'Label 2', 'Label 3', 'Label 4', 'Label 5', and 'Label 6'. At the bottom right of the form is a 'Submit' button. The background also shows a 'Foto' button and a partial 'Aksi' label.

Gambar 5.11 Antarmuka Halaman Registrasi

### 5.1.5.5 Antarmuka Halaman Edit Soal

Halaman ini merupakan halaman atau form untuk mengedit soal. Form ini adalah modals atau pop up. Dalam modals ini terdapat data yang harus diisi antara lain: soal, opsi A, opsi B, opsi C, opsi D, dan jawaban dari soal. Rancangan antarmuka halaman tambah admin dapat dilihat pada gambar 5.12 berikut.



Gambar 5.12 Antarmuka Halaman Edit Soal

## 5.2 Implementasi

Bagian ini memaparkan tahapan implementasi Sistem Seleksi Penerimaan Siswa Baru SMA yang mengacu pada hasil analisis kebutuhan dan proses perancangan. Pembahasan pada sub bab ini terdiri dari spesifikasi sistem, batasan implementasi, implementasi program, dan implementasi antarmuka.

### 5.2.1 Spesifikasi Sistem

Spesifikasi sitem terdiri dari dua bagian, yaitu spesifikasi perangkat keras dan spesifikasi perangkat lunak. Berikut adalah spesifikasi perangkat keras dan spesifikasi perangkat lunak.

#### 5.2.1.1 Spesifikasi Perangkat Keras

Pembangunan Sistem Seleksi Penerimaan Siswa Baru SMA menggunakan spesifikasi perangkat keras dapat dilihat pada Tabel 5.4 berikut.

Tabel 5.4 Spesifikasi Perangkat Keras

Nama Komponen	Spesifikasi
Prosesor	Intel Core i3
Memori (RAM)	4 GB
Harddisk	640 GB
Kartu Grafis	Nvidia Geforce GT 520M
Monitor	14.0", Resolusi 1366 x 768

### 5.2.1.2 Spesifikasi Perangkat Lunak

Pembangunan Sistem Seleksi Penerimaan Siswa Baru SMA menggunakan spesifikasi perangkat lunak dapat dilihat pada Tabel 5.5 berikut.

**Tabel 5.5 Spesifikasi Perangkat Lunak**

Nama Komponen	Spesifikasi
Sistem Operasi	Windows 7 Ultimate
Browser	Google Chrome versi 52.0.2743.24, Firefox 46.0.1
Code Editor	Sublime Text 3
Image Editor	Adobe Photoshop CS 3
Basis Data	MySQL 5.6
Pemodelan, UML	Ideas Modeler 9.20, Enterprise Architect 12.0
Framework	Code Igniter 2.1.4, Bootstrap
Dokumentasi	MS Word 2013

### 5.2.2 Batasan Implementasi

Beberapa batasan implementasi Sistem Seleksi Penerimaan Siswa Baru SMA diantaranya :

1. Sistem Seleksi Penerimaan Siswa Baru SMA dirancang dan dibangun menggunakan bahasa pemrograman PHP, *framework* Code Igniter dan Bootstrap untuk pembuatan website, serta menggunakan MySQL sebagai pendukung basis data.
2. Sistem Seleksi Penerimaan Siswa Baru SMA memiliki fitur utama berupa registrasi calon siswa, upload bukti pembayaran, validasi pembayaran, menentukan jadwal ujian, mengerjakan soal, dan menentukan jumlah siswa diterima.
3. Data yang digunakan dalam Sistem Seleksi Penerimaan Siswa Baru SMA diambil dari SMA Negeri 1 Purwosari.

### 5.2.3 Implementasi Program

Proses implementasi Sistem Seleksi Penerimaan Siswa Baru SMA dilakukan dengan menggunakan bahasa pemrograman PHP dengan *framework* Code Igniter yang menggunakan konsep MVC. Pada sistem ini, terdapat fungsi-fungsi seperti registrasi calon siswa, mengerjakan soal, dan menentukan jumlah siswa diterima. Berikut adalah implementasi program dari beberapa fungsi yang ada di dalam Sistem Seleksi.

#### 5.2.3.1 Implementasi Program Mengerjakan Soal

Mengerjakan soal merupakan kegiatan yang dilakukan oleh calon siswa dimana calon siswa menjawab soal-soal pilihan ganda yang telah disediakan oleh

sistem. Fungsi ini tidak memiliki parameter dan tidak memiliki nilai kembalain. Fungsi ini akan memanggil fungsi untuk mengacak soal, kemudian menampilkan halaman untuk mengerjakan soal apabila nilai ujian calon siswa masih kosong. Implementasi program mengerjakan soal dapat dilihat pada gambar 5.13 berikut.

```
function kerjakanUjian(){
    $calonsiswa['nilaiUj']=$this->e_calonsiswa->getNilaiUjian();
    $nilaiujian;
    foreach ($calonsiswa['nilaiUj'] as $nilaiuj) {
        $nilaiujian = $nilaiuj->nilaiUjian;
    }
    if($nilaiujian==NULL){
        $soal['data'] = $this->e_soal->getRandomSoal();
        $this->load->view('kerjakan-ujian',$soal);
    }
    else{
        echo"<script>alert('Anda sudah melaksanakan
ujian');window.close();</script>";
    }
}
```

**Gambar 5.13 Kode Program Mengerjakan Soal**

### 5.2.3.2 Implementasi Menentukan Jumlah Siswa Diterima

Menentukan jumlah siswa diterima adalah kegiatan yang dilakukan oleh admin dengan memasukkan jumlah siswa diterima. Jumlah siswa yang diterima ditentukan berdasarkan nilai ujian nasional dan nilai ujian seleksi. Fungsi ini tidak memiliki parameter dan tidak memiliki nilai kembalian. Di dalam fungsi ini terdapat fungsi untuk menentukan calon siswa yang diterima di SMA berdasarkan nilai ujian nasional dan nilai ujian. Implementasi program menentukan jumlah siswa diterima dapat dilihat pada gambar 5.14 berikut.

```
function tentukanSiswaDiterima(){
    if (($this->input->post('jumlah'))>0) {
        $jumlah = $this->input->post('jumlah');
        $this->e_calonsiswa->updateSiswaDiterima($jumlah);
        echo"<script>alert('Jumlah siswa yang diterima berhasil
diubah.');

```

**Gambar 5.14 Kode Program Menentukan Jumlah Siswa Diterima**

### 5.2.3.3 Implementasi Melihat Hasil Seleksi

Melihat hasil seleksi adalah kegiatan yang dilakukan yang dilakukan oleh calon siswa untuk mengetahui apakah calon siswa tersebut diterima atau tidak

diterima di SMA sebagai siswa baru. Fungsi ini akan menampilkan halaman hasil seleksi. Pada halaman tersebut calon siswa juga dapat melihat nilai ujian dan nilai akhir. Implementasi program melihat hasil seleksi dapat dilihat pada gambar 5.15 berikut.

```
function lihatHasil(){
    date_default_timezone_set('asia/jakarta');
    $haripengumuman = date($this->e_jadwal->getHariPengumuman());
    $sekarang = date('Y-m-d');
    if($sekarang>=$haripengumuman){
        $id = $this->session->userdata('nispn');
        $calonsiswa['data'] = $this->e_calonsiswa->getData($id);
        $this->load->view('hasil-seleksi', $calonsiswa);
    }
    else{
        echo"<script>alert('Maaf, pengumuman hasil seleksi belum
ada.');
```

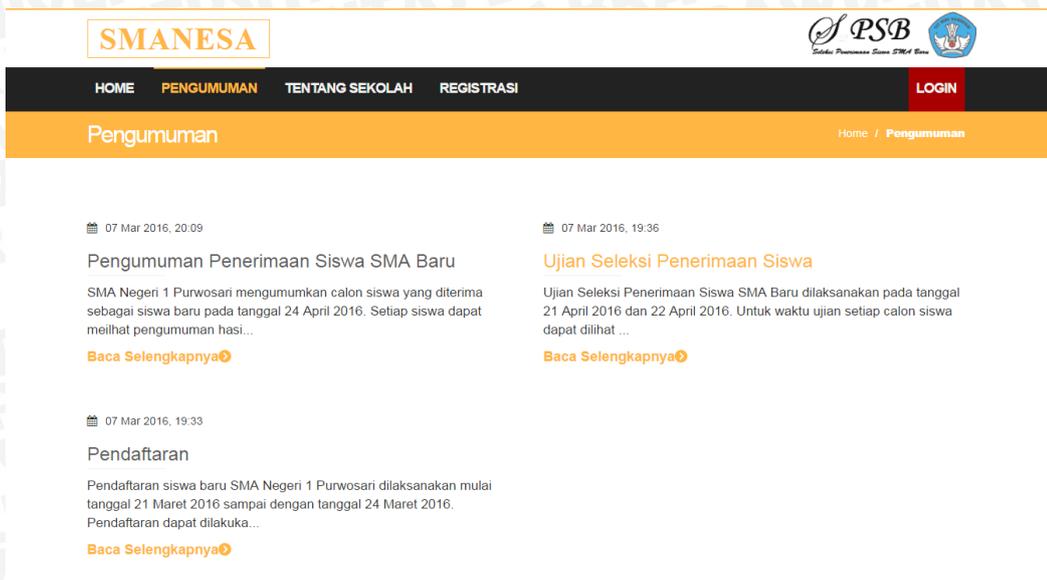
**Gambar 5.15 Kode Program Melihat Hasil Seleksi**

## 5.2.4 Implementasi Antarmuka

Antarmuka sistem adalah sarana yang digunakan oleh pengguna untuk berinteraksi dengan sistem. Berikut beberapa penjelasan mengenai implementasi antarmuka yang telah dibuat pada Sistem Seleksi Penerimaan Siswa Baru SMA.

### 5.2.4.1 Implementasi Antarmuka Lihat Pengumuman

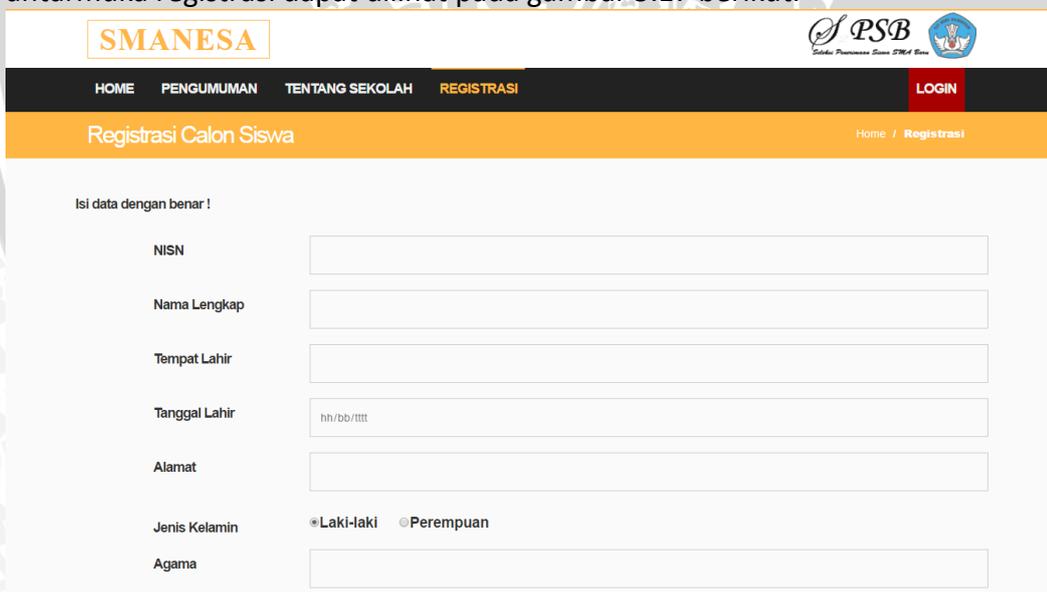
Antarmuka lihat pengumuman merupakan halaman yang berisi pengumuman-pengumuman mengenai seleksi penerimaan siswa baru yang telah diposting oleh admin. Implementasi antarmuka lihat pengumuman dapat dilihat pada gambar 5.16 berikut.



Gambar 5.16 Antarmuka Halaman Lihat Pengumuman

### 5.2.4.2 Implementasi Antarmuka Registrasi

Antarmuka registrasi merupakan halaman untuk melakukan pendaftaran. Pada halaman ini terdapat data-data yang harus diisi oleh calon siswa seperti nisn, nama lengkap, tempat lahir, tanggal lahir, dan sebagainya. Implementasi antarmuka registrasi dapat dilihat pada gambar 5.17 berikut.

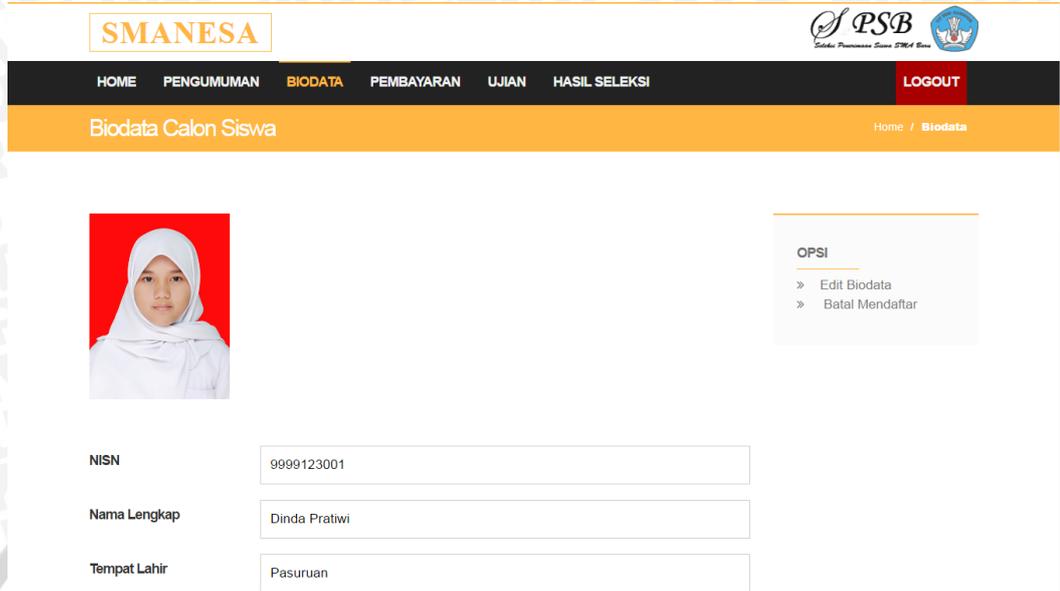


Gambar 5.17 Implementasi Antarmuka Registrasi

### 5.2.4.3 Implementasi Antarmuka Lihat Profil

Antarmuka lihat profil merupakan halaman yang menampilkan data-data calon siswa seperti nisn, nama lengkap, tempat lahir, tanggal lahir, dan

sebagainya. Implementasi antarmuka lihat profil dapat dilihat pada gambar 5.18 berikut.



**Gambar 5.18 Implementasi Antarmuka Lihat Profil**

#### 5.2.4.4 Implementasi Antarmuka Kerjakan Soal

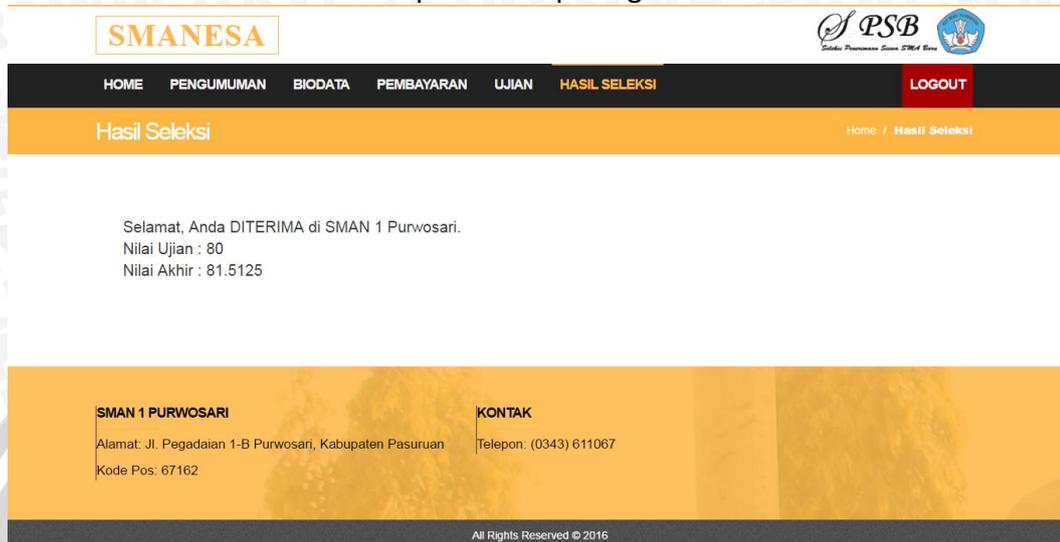
Antarmuka kerjakan soal merupakan halaman yang menampilkan soal-soal ujian secara acak yang harus dikerjakan oleh calon siswa dengan batasan waktu tertentu. Apabila waktu mengerjakan habis, sedangkan calon siswa belum selesai mengerjakan, soal yang telah dijawab otomatis akan tersimpan ke dalam sistem dan akan dinilai. Implementasi antarmuka kerjakan soal dapat dilihat pada gambar 5.19 berikut.



**Gambar 5.19 Implementasi Antarmuka Kerjakan Soal**

### 5.2.4.5 Implementasi Antarmuka Lihat Hasil Seleksi

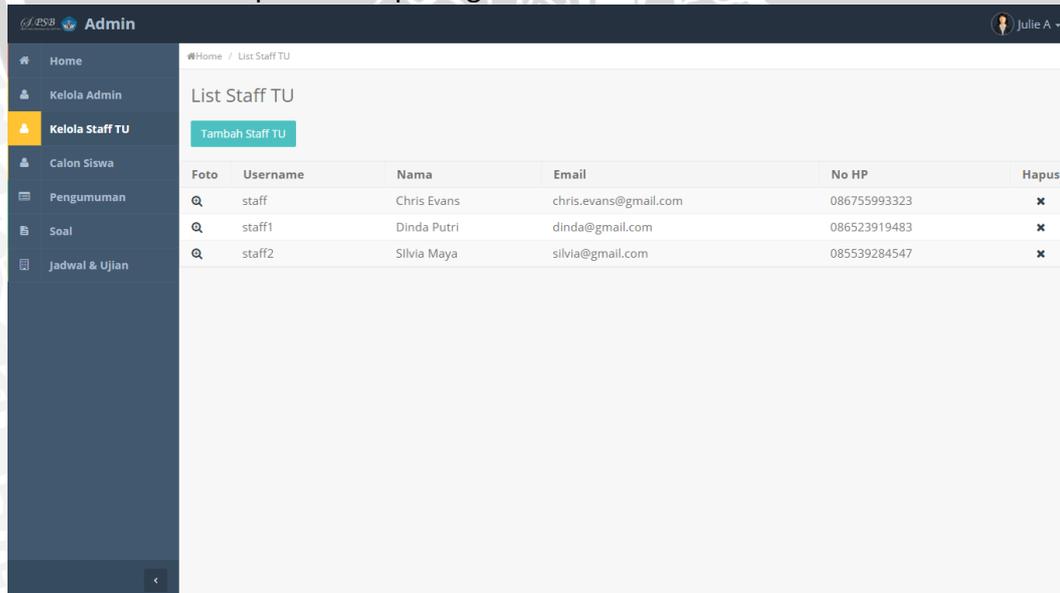
Antarmuka lihat hasil seleksi merupakan halaman yang menampilkan hasil akhir dari proses seleksi, apakah calon siswa diterima atau tidak. Pada halaman ini juga menampilkan nilai ujian seleksi dan nilai akhir calon siswa. Implementasi antarmuka lihat hasil seleksi dapat dilihat pada gambar 5.20 berikut.



Gambar 5.20 Implementasi Antarmuka Lihat Hasil Seleksi

### 5.2.4.6 Implementasi Antarmuka Lihat Seluruh Staff TU

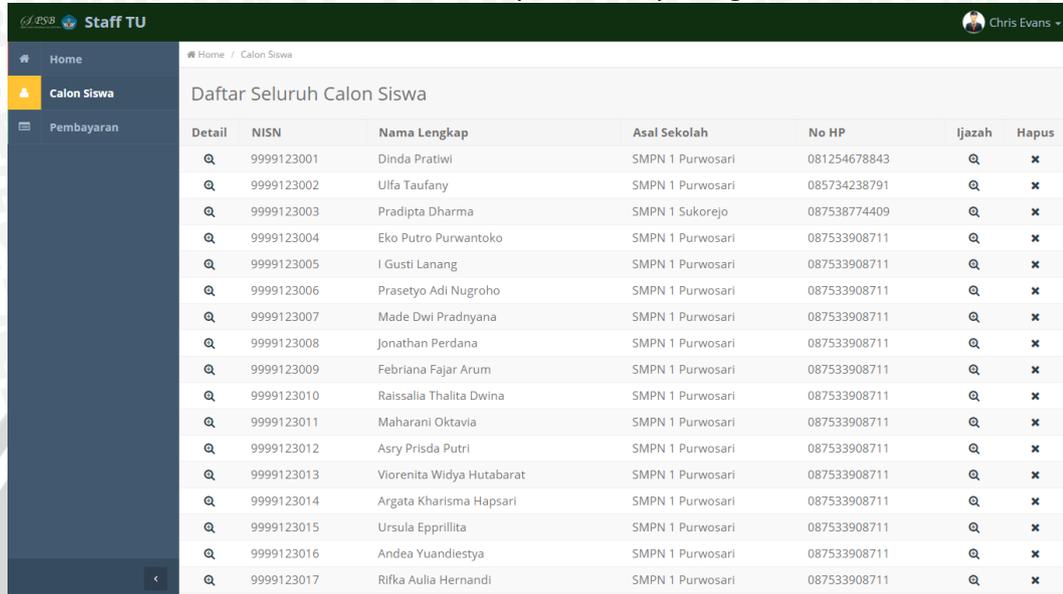
Antarmuka lihat seluruh staff tu merupakan halaman yang menampilkan seluruh staff TU beserta data-data dari staff TU. Data-data tersebut antara lain: foto, username, nama, email, dan nomor hp. Implementasi antarmuka lihat seluruh staff TU dapat dilihat pada gambar 5.21 berikut.



Gambar 5.21 Implementasi Antarmuka Lihlar Seluruh Staff TU

### 5.2.4.7 Implementasi Antarmuka Lihat Seluruh Calon Siswa

Antarmuka lihat seluruh calon siswa merupakan halaman yang menampilkan data-data dari seluruh calon siswa. Pada halaman ini, staff TU juga dapat menghapus calon siswa apabila data tersebut bukan data asli. Implementasi antarmuka lihat seluruh calon siswa dapat dilihat pada gambar 5.22 berikut.

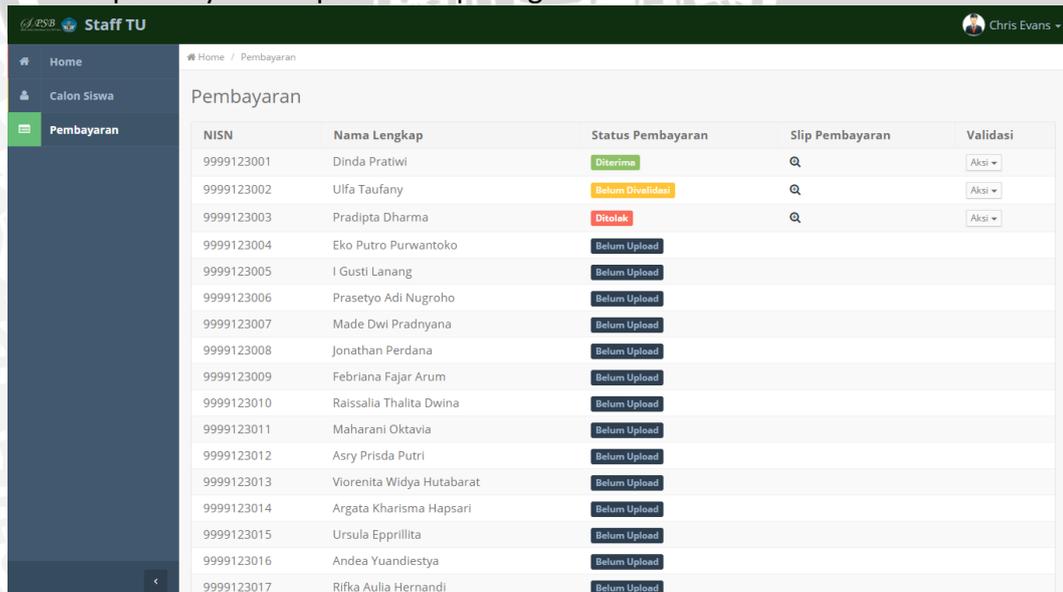


Detail	NISN	Nama Lengkap	Asal Sekolah	No HP	Ijazah	Hapus
	9999123001	Dinda Pratiwi	SMPN 1 Purwosari	081254678843		
	9999123002	Ulfa Taufany	SMPN 1 Purwosari	085734238791		
	9999123003	Pradipta Dharma	SMPN 1 Sukorejo	087538774409		
	9999123004	Eko Putro Purwantoko	SMPN 1 Purwosari	087533908711		
	9999123005	I Gusti Lanang	SMPN 1 Purwosari	087533908711		
	9999123006	Prasetyo Adi Nugroho	SMPN 1 Purwosari	087533908711		
	9999123007	Made Dwi Pradnyana	SMPN 1 Purwosari	087533908711		
	9999123008	Jonathan Perdana	SMPN 1 Purwosari	087533908711		
	9999123009	Febriana Fajar Arum	SMPN 1 Purwosari	087533908711		
	9999123010	Raissalia Thalita Dwina	SMPN 1 Purwosari	087533908711		
	9999123011	Maharani Oktavia	SMPN 1 Purwosari	087533908711		
	9999123012	Asry Prisda Putri	SMPN 1 Purwosari	087533908711		
	9999123013	Viorenita Widya Hutabarat	SMPN 1 Purwosari	087533908711		
	9999123014	Argata Kharisma Hapsari	SMPN 1 Purwosari	087533908711		
	9999123015	Ursula Epprillita	SMPN 1 Purwosari	087533908711		
	9999123016	Andea Yuandiestya	SMPN 1 Purwosari	087533908711		
	9999123017	Rifka Aulia Hernandi	SMPN 1 Purwosari	087533908711		

Gambar 5.22 Implementasi Antarmuka Lihat Seluruh Calon Siswa

### 5.2.4.8 Implementasi Antarmuka Validasi Pembayaran

Antarmuka validasi pembayaran merupakan halaman untuk memvalidasi pembayaran dari seluruh calon siswa. Pada halaman ini staff TU dapat mengubah status pembayaran menjadi diterima atau ditolak. Implementasi antarmuka validasi pembayaran dapat dilihat pada gambar 5.23 berikut.

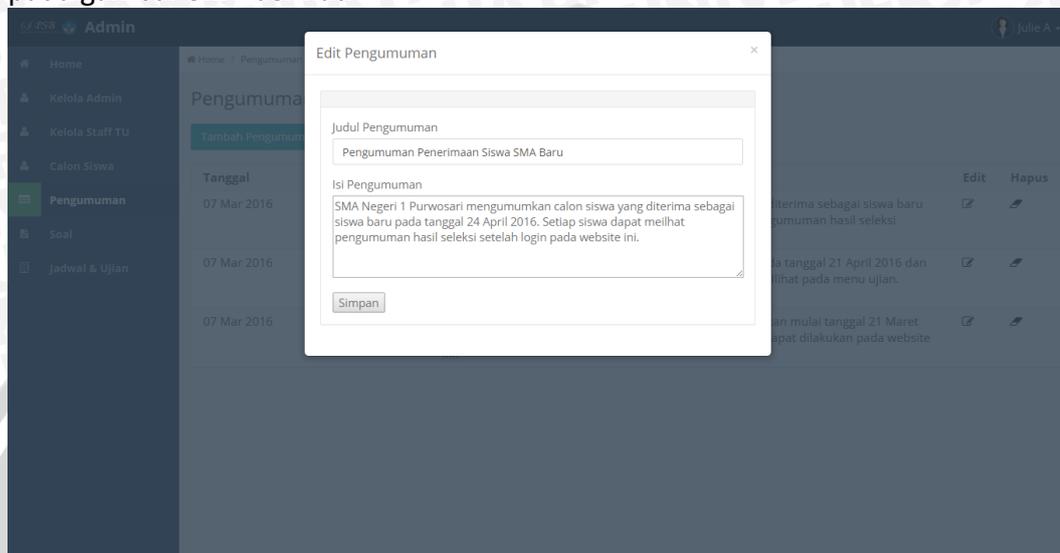


NISN	Nama Lengkap	Status Pembayaran	Slip Pembayaran	Validasi
9999123001	Dinda Pratiwi	Diterima		Aksi
9999123002	Ulfa Taufany	Belum Divalidasi		Aksi
9999123003	Pradipta Dharma	Ditolak		Aksi
9999123004	Eko Putro Purwantoko	Belum Upload		
9999123005	I Gusti Lanang	Belum Upload		
9999123006	Prasetyo Adi Nugroho	Belum Upload		
9999123007	Made Dwi Pradnyana	Belum Upload		
9999123008	Jonathan Perdana	Belum Upload		
9999123009	Febriana Fajar Arum	Belum Upload		
9999123010	Raissalia Thalita Dwina	Belum Upload		
9999123011	Maharani Oktavia	Belum Upload		
9999123012	Asry Prisda Putri	Belum Upload		
9999123013	Viorenita Widya Hutabarat	Belum Upload		
9999123014	Argata Kharisma Hapsari	Belum Upload		
9999123015	Ursula Epprillita	Belum Upload		
9999123016	Andea Yuandiestya	Belum Upload		
9999123017	Rifka Aulia Hernandi	Belum Upload		

Gambar 5.23 Implementasi Antarmuka Validasi Pembayaran

### 5.2.4.9 Implementasi Antarmuka Edit Pengumuman

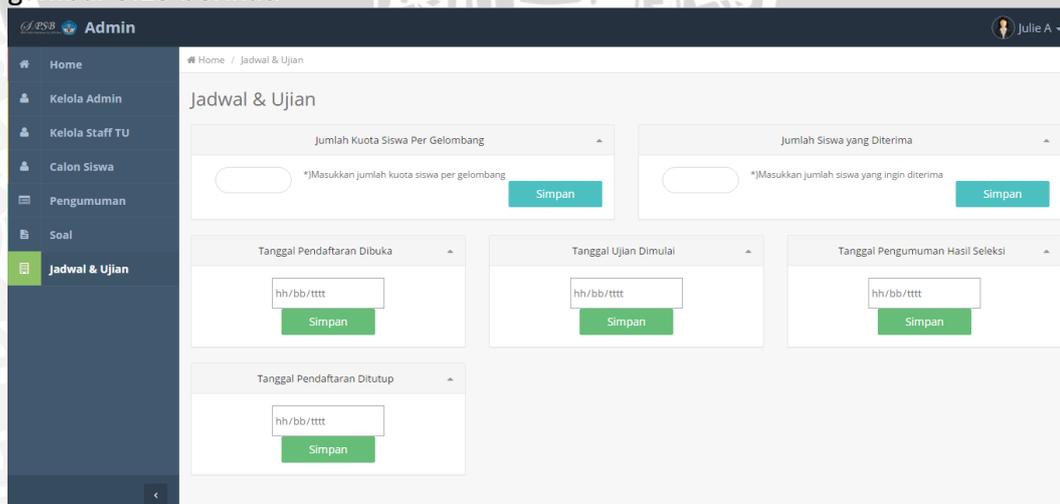
Antarmuka edit pengumuman merupakan halaman untuk mengubah pengumuman apabila terdapat perubahan atau kesalahan. Halaman ini berupa pop up atau modals dimana *user* diminta untuk mengisi form perubahan dari pengumuman terakhir. Implementasi antarmuka edit pengumuman dapat dilihat pada gambar 5.24 berikut.



Gambar 5.24 Implementasi Antarmuka Edit Pengumuman

### 5.2.4.10 Implementasi Antarmuka Menentukan Jumlah Siswa Diterima

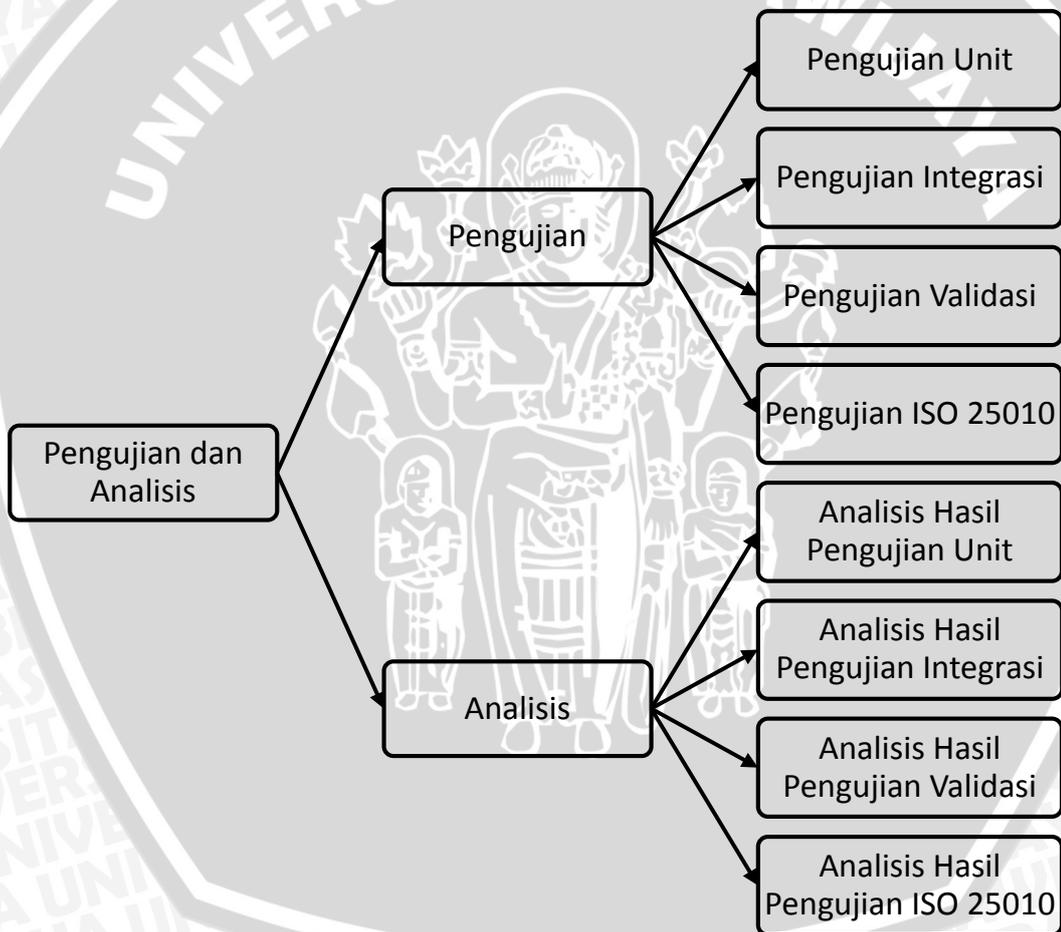
Antarmuka menentukan jumlah siswa diterima merupakan halaman untuk menentukan jumlah calon siswa yang diterima di SMA sebagai siswa baru. Implementasi antarmuka menentukan jumlah siswa diterima dapat dilihat pada gambar 5.25 berikut.



Gambar 5.25 Implementasi Antarmuka Menentukan Jumlah Siswa Diterima

## BAB 6 PENGUJIAN DAN ANALISIS

Pada bab ini menjelaskan tentang pengujian dan analisis dari hasil implementasi yang telah dilakukan pada bab sebelumnya. Proses pengujian dimulai dari tahap pengujian code program yang akan dilakukan pada pengujian unit, kemudian dilanjutkan pengujian arsitektur atau desain yang akan dilakukan pada pengujian integrasi, setelah itu dilanjutkan dengan pengujian sistem atau fungsionalitas yang akan dilakukan pada pengujian validasi dan yang terakhir adalah pengujian berbasis ISO 25010. Sedangkan proses analisis bertujuan untuk mendapatkan kesimpulan dari semua hasil pengujian yang telah dilakukan. Tahapan pengujian dan analisis Sistem Seleksi Penerimaan Siswa Baru SMA dapat dilihat pada Gambar 6.1 berikut.



Gambar 6.1 Tahap-Tahap Pengujian Dan Analisis

### 6.1 Pengujian

Pengujian yang dilakukan terdiri dari pengujian unit, pengujian integrasi, pengujian validasi, dan pengujian ISO 25010.

### 6.1.1 Pengujian Unit

Tujuan dari pengujian unit yaitu untuk memastikan algoritma algoritma yang memiliki prioritas tinggi telah diimplementasikan sesuai dengan yang diharapkan. Pengujian unit dilakukan menggunakan pengujian basis path. Algoritma yang diuji dalam pengujian ini adalah algoritma penilaian, menentukan kuota, dan menentukan jadwal.

#### 1. Pengujian Algoritma Penilaian

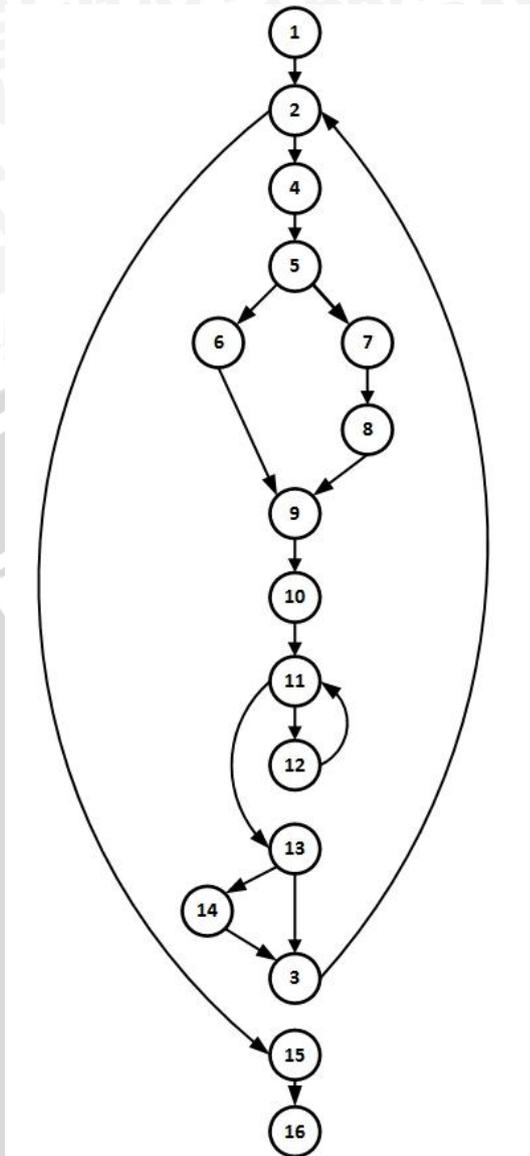
Algoritma penilaian merupakan algoritma dengan prioritas tinggi. Algoritma ini berfungsi untuk menghitung jawaban ujian seleksi calon siswa kemudian menghitung nilai akhir. Gambar 6.2 memaparkan algoritma penilaian beserta node flowgraph.

```
function penilaian(){
(1) skor=0; (2) (3)
  for (i=1; i<=50; i++){
    (4) idsoal[i]= post(soalno.i);
    (5) if (!empty (post(opsino.i)))
      (6) then jawaban[i]= post(opsino.i);
      end if
    (7) else then
      (8) jawaban[i] = "";
    (9) jawaban[no] = mengambil jawaban yang benar (idsoal[i]);
      (11)
    (10) foreach (jawaban[no] as jawab) {
      (12) jawabanno[i] = jawab->jawaban;
    }
    (13) if (jawaban[i] == jawabanno[i])
      (14) then skor++;
      end if
    (15) end for

    nilaiun = mengambil nilai un calon siswa
    nilaiujian = (100/50*skor);
    (16) nilaitotal = (100/50*skor*(75/100))+((25/100)*nilaiun);
    Menyimpan nilaiujian dan nilaitotal ke dalam database
    Menampilkan pesan bahwa calon siswa telah selesai mengerjakan ujian
  }
}
```

Gambar 6.2 Pembentukan Node Algoritma Penilaian

Pada pembentukan node algoritma penilaian, node yang terbentuk sebanyak 16. Dari node yang sudah terbentuk langkah selanjutnya adalah membuat flowgraph dari algoritma penilaian berdasarkan node-node yang ditentukan. Flowgraph algoritma penilaian dapat dilihat pada Gambar 6.3 berikut.



**Gambar 6.3 Flowgraph Algoritma Penilaian**

Berdasarkan flowgraph algoritma penilaian yang dapat dilihat pada Gambar 6.3 maka dapat dihitung nilai cyclometric complexity  $V(G)$  sebagai berikut,

$$\begin{aligned}
 V(G) &= E - N + 2 \\
 &= 19 - 16 + 2 \\
 &= 5
 \end{aligned}$$

$$\begin{aligned}
 V(G) &= P + 1 \\
 &= 4 + 1 \\
 &= 5
 \end{aligned}$$

Sehingga dari hasil persamaan cyclometric complexity maka didapatkan jalur independen yaitu:

1. 1-2-15-16



2. 1-2-4-5-6-9-10-11-13-3-2-15-16
3. 1-2-4-5-6-9-10-11-12-13-3-2-15-16
4. 1-2-4-5-7-8-9-10-11-12-11-13-3-2-15-16
5. 1-2-4-5-6-9-10-11-12-11-13-14-3-2-15-16

Berdasarkan 5 jalur independen yang telah didefinisikan tersebut maka dapat dibentuk kasus ujinya. Tabel 6.1 memaparkan kasus uji algoritma penilaian.

**Tabel 6.1 Kasus Uji Algoritma Penilaian**

No	Jalur	Data input	Hasil yang diharapkan	Hasil yang diperoleh	Status
1	1-2-15-16	Nilai $i$ lebih dari 50	Sistem menampilkan pesan bahwa calon siswa telah selesai mengerjakan ujian	Sistem menampilkan pesan bahwa calon siswa telah selesai mengerjakan ujian	Valid
2	1-2-4-5-6-9-10-11-13-3-2-15-16	Nilai $i \geq 1$ , jawaban tidak kosong, jawaban salah	Skor tidak bertambah	Skor tidak bertambah	Valid
3	1-2-4-5-6-9-10-11-12-13-3-2-15-16	Nilai $i \leq 50$ , jawaban tidak kosong, jawaban salah	Skor tidak bertambah	Skor tidak bertambah	Valid
4	1-2-4-5-7-8-9-10-11-12-11-13-3-2-15-16	Jawaban kosong	Skor tidak bertambah	Skor tidak bertambah	Valid
5	1-2-4-5-6-9-10-11-12-11-13-14-3-2-15-16	Jawaban tidak kosong dan jawaban benar	Skor ditambah dengan 1	Skor ditambah dengan 1	Valid

## 2. Pengujian Algoritma Menentukan Kuota Per Gelombang Ujian

Algoritma menentukan kuota per gelombang ujian merupakan algoritma dengan prioritas tinggi dengan tujuan untuk menentukan jumlah calon siswa per gelombang ujian. Gambar 6.4 memaparkan algoritma menentukan kuota per gelombang ujian.

```

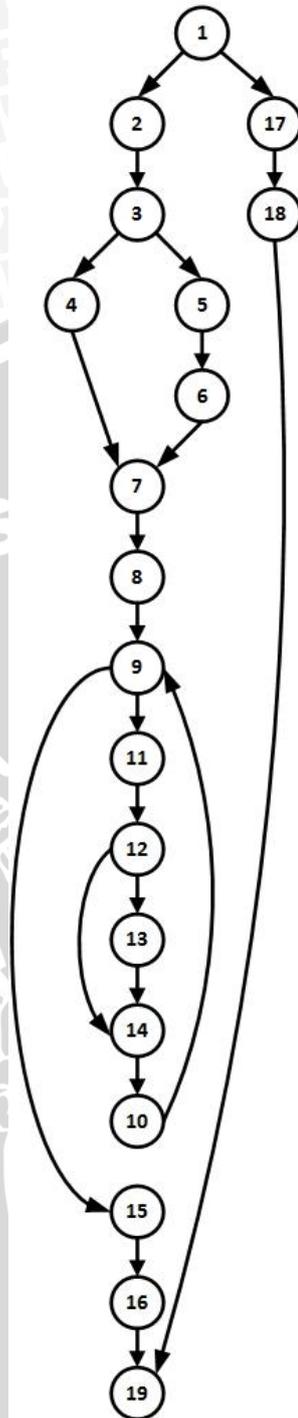
function tentukanKuota(){
(1)if (( post(jumlahkuota)) > 0 )
(2) {
    then jumlah_kuota = post(jumlahkuota);
    hari_ujian = mengambil hari ujian
    jumlah = mengambil jumlah calon siswa
(3)if (jumlah % jumlah_kuota == 0)
(4)then jumlah_gelombang = jumlah / jumlah_kuota;
    end if
(5)else then
(6)jumlah_gelombang = floor((jumlah / jumlah_kuota) + 1);

    {Set timezone menjadi asia/jakarta
(7) {
    start = 07:00:00;
    baris = 0;
(9)
(10)
(8)for (i=1; i<=jumlah_gelombang; i++)
    {
    Memasukkan jadwal ujian dengan fungsi insertJadwalUjian dengan parameter
($baris,$jumlah_kuota,$hari_ujian,$start);
(11) start1 = start ditambah 90 menit
    start = start1;
    baris = baris + jumlah_kuota;
(12)if (start == 17:30:00)
(13) {
    then start = '07:00:00';
    hari_ujian1= hari_ujian ditambah 1 hari
    hari_ujian = hari_ujian1
(14)end if
(15)end for
(16)end if
(17)else then
(18)Menampilkan pesan Jumlah kuota siswa per gelombang harus lebih dari 0
(19)
}
}
}
}

```

**Gambar 6.4 Pembentukan Node Algoritma Menentukan Kuota Per Gelombang Ujian**

Pada pembentukan node algoritma menentukan kuota per gelombang ujian, node yang terbentuk sebanyak 19. Dari node yang sudah terbentuk langkah selanjutnya adalah membuat flowgraph dari algoritma menentukan kuota per gelombang ujian berdasarkan node-node yang ditentukan. Flowgraph algoritma menentukan kuota per gelombang ujian dapat dilihat pada Gambar 6.5 berikut.



**Gambar 6.5 Flowgraph Algoritma Menentukan Kuota Per Gelombang Ujian**

Berdasarkan flowgraph algoritma menentukan kuota per gelombang ujian yang dapat dilihat pada Gambar 6.5 maka dapat dihitung nilai *cyclometric complexity*  $V(G)$  sebagai berikut.

$$\begin{aligned}
 V(G) &= E - N + 2 \\
 &= 22 - 19 + 2 \\
 &= 5
 \end{aligned}$$



$$\begin{aligned}
 V(G) &= P + 1 \\
 &= 4 + 1 \\
 &= 5
 \end{aligned}$$

Sehingga dari hasil persamaan cyclometric complexity maka didapatkan jalur independen yaitu:

1. 1-17-18-19
2. 1-2-3-4-7-8-9-15-16-19
3. 1-2-3-4-7-8-9-11-12-14-10-9-15-16-19
4. 1-2-3-5-6-7-8-9-11-12-14-10-9-15-16-19
5. 1-2-3-5-6-7-8-9-11-12-13-14-10-9-15-16-19

Berdasarkan 5 jalur independen yang telah didefinisikan tersebut maka dapat dibentuk kasus uji nya. Tabel 6.2 memaparkan kasus uji algoritma menentukan kuota per gelombang ujian.

**Tabel 6.2 Kasus Uji Algoritma Menentukan Kuota Per Gelombang Ujian**

No	Jalur	Data input	Hasil yang diharapkan	Hasil yang diperoleh	Status
1	1-17-18-19	Input 'jumlahkuota' kurang dari 0	Sistem menampilkan pesan bahwa jumlah kuota siswa per gelombang harus lebih dari 0	Sistem menampilkan pesan bahwa jumlah kuota siswa per gelombang harus lebih dari 0	Valid
2	1-2-3-4-7-8-9-15-16-19	Input 'jumlahkuota' lebih dari 0, 'jumlah' habis dibagi dengan 'jumlah_kuota' dan 'i' lebih dari 'jumlah_gelombang'	Jadwal ujian calon siswa tidak berubah	Jadwal ujian calon siswa tidak berubah	Valid
3	1-2-3-4-7-8-9-11-12-14-10-9-15-16-19	Input 'jumlahkuota' lebih dari 0, 'jumlah' habis dibagi dengan 'jumlah_kuota', 'i' kurang dari sama dengan 'jumlah_gelombang' dan 'start' tidak sama dengan '17:30:00'	Jadwal jam ujian calon siswa berubah	Jadwal jam ujian calon siswa berubah	Valid
4	1-2-3-5-6-7-8-9-11-12-14-10-9-15-16-19	Input 'jumlahkuota' lebih dari 0, 'jumlah' tidak habis dibagi dengan 'jumlah_kuota', 'i' kurang dari sama dengan 'jumlah_gelombang' dan 'start' tidak sama dengan '17:30:00'	Jadwal jam ujian calon siswa berubah	Jadwal jam ujian calon siswa berubah	Valid

**Tabel 6.2 Kasus Uji Algoritma Menentukan Kuota Per Gelombang Ujian (lanjutan)**

No	Jalur	Data input	Hasil yang diharapkan	Hasil yang diperoleh	Status
5	1-2-3-5-6-7-8-9-11-12-13-14-10-9-15-16-19	Input 'jumlahkuota' lebih dari 0, 'jumlah' tidak habis dibagi dengan 'jumlah_kuota', 'i' kurang dari sama dengan 'jumlah_gelombang' dan 'start' tidak sama dengan '17:30:00'	Jadwal hari dan jam ujian calon siswa berubah	Jadwal hari dan jam ujian calon siswa berubah	Valid

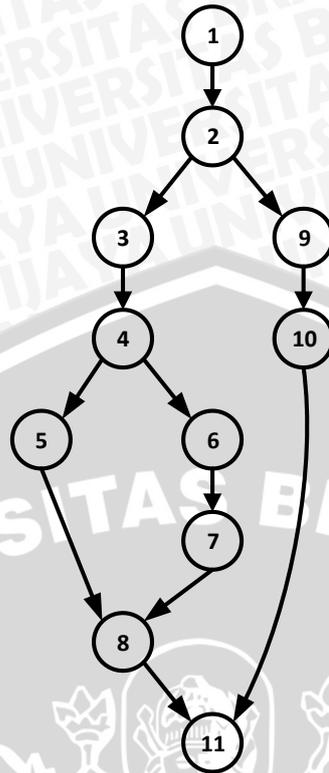
### 3. Pengujian Algoritma Menentukan Jadwal

Algoritma menentukan jadwal merupakan algoritma dengan prioritas tinggi dengan tujuan untuk menentukan tanggal pendaftaran dibuka, tanggal pendaftaran ditutup, tanggal ujian dimulai, dan tanggal pengumuman hasil seleksi. Gambar 6.6 memaparkan algoritma menentukan jadwal.

```
function tentukanJadwal(){
(1)id=GET[id];
(2)if (id == pendaftarantutup)
(3)then hariPendaftaranBuka = mengambil hari pendaftaran dibuka
(4)if (post(tanggal) >= hariPendaftaranBuka)
(5)then melakukan update jadwal dengan fungsi updateJadwal(id);
end if
(6)else then
(7)Menampilkan pesan bahwa tanggal pendaftaran ditutup harus sama atau
lebih dari tanggal pendaftaran dibuka
(8)end if
(9)else then
(10)melakukan update jadwal dengan fungsi updateJadwal(id);
(11)Menampilkan halaman kelola ujian
}
```

**Gambar 6.6 Pembentukan Node Algoritma Menentukan Jadwal**

Pada pembentukan node algoritma menentukan jadwal, node yang terbentuk sebanyak 11. Dari node yang sudah terbentuk langkah selanjutnya adalah membuat flowgraph dari algoritma menentukan jadwal berdasarkan node-node yang ditentukan. Flowgraph algoritma menentukan jadwal dapat dilihat pada Gambar 6.7 berikut.



**Gambar 6.7 Flowgraph Algoritma Menentukan Jadwal**

Berdasarkan flowgraph algoritma menentukan kuota per gelombang ujian yang dapat dilihat pada Gambar 6.7 maka dapat dihitung nilai *cyclometric complexity*  $V(G)$  sebagai berikut.

$$\begin{aligned}
 V(G) &= E - N + 2 \\
 &= 12 - 11 + 2 \\
 &= 3
 \end{aligned}$$

$$\begin{aligned}
 V(G) &= P + 1 \\
 &= 2 + 1 \\
 &= 3
 \end{aligned}$$

Sehingga dari hasil persamaan *cyclometric complexity* maka didapatkan jalur independen yaitu:

1. 1-2-9-10-11
2. 1-2-3-4-5-8-11
3. 1-2-3-4-6-7-8-11

Berdasarkan 5 jalur independen yang telah didefinisikan tersebut maka dapat dibentuk kasus ujinya. Tabel 6.3 memaparkan kasus uji algoritma menentukan jadwal.



Tabel 6.3 Kasus Uji Algoritma Menentukan Jadwal

No	Jalur	Data input	Hasil yang diharapkan	Hasil yang diperoleh	Status
1	1-2-9-10-11	id tidak sama dengan 'pendaftarantut up'	Sistem memanggil fungsi updateJadwal(\$id)	Sistem memanggil fungsi updateJadwal(\$id)	Valid
2	1-2-3-4-5-8-11	Id = 'pendaftarantut up' dan 'tanggal' >= \$hariPendaftarannBuka	Sistem memanggil fungsi updateJadwal(\$id)	Sistem memanggil fungsi updateJadwal(\$id)	Valid
3	1-2-3-4-6-7-8-11	Id = 'pendaftarantut up' dan 'tanggal' < \$hariPendaftarannBuka	Sistem menampilkan pesan bahwa tanggal pendaftaran ditutup harus sama atau lebih dari tanggal pendaftaran dibuka	Sistem menampilkan pesan bahwa tanggal pendaftaran ditutup harus sama atau lebih dari tanggal pendaftaran dibuka	Valid

### 6.1.2 Pengujian Integrasi

Pengujian integrasi adalah sebuah teknik sistematis untuk membangun arsitektur perangkat lunak dengan melakukan tes untuk mengungkap kesalahan yang berkaitan dengan tampilan program. Tujuan dari pengujian integrasi adalah untuk mengambil komponen unit diuji dan membangun program terstruktur yang sebelumnya sudah ditentukan pada tahap desain (Pressman, 2010). Pada kasus ini teknik yang digunakan adalah *black box*. Sistem memenuhi aspek pengujian integrasi apabila seluruh hasil yang diharapkan (*expected result*) sesuai dengan hasil yang diperoleh (*actual result*). Namun, apabila terdapat hasil yang tidak sesuai dengan yang diharapkan maka proses integrasi antar kelas harus diperbaiki.

#### 1. Pengujian Integrasi Antara Method Tambah Pengumuman dan Melihat Pengumuman

Method tambah pengumuman atau tambahPengumuman() merupakan method untuk membuat pengumuman baru mengenai seleksi penerimaan siswa SMA baru. Method ini berada pada kelas `c_admin`. Sedangkan, method melihat pengumuman atau lihatPengumuman() adalah method untuk melihat pengumuman mengenai seleksi penerimaan siswa SMA baru. Method ini berada pada kelas `c_main`. Tabel 6.4 menjelaskan interaksi method tambah pengumuman dan melihat pengumuman.

**Tabel 6.4 Interaksi Method Tambah Pengumuman dan Melihat Pengumuman**

No	Method dari Class c_admin	Method dari Class c_main	Goal
1	tambahPengumuman()	lihatPengumuman()	Melihat pengumuman yang baru saja ditambahkan

Prosedur uji yang dilakukan yaitu:

1. Penguji menyiapkan *class c\_admin* untuk mengaktifkan method tambah pengumuman
2. Penguji menyiapkan *input-input* yang diperlukan method tambahPengumuman()
3. Penguji mengeksekusi method tambahPengumuman()
4. Penguji melihat apakah pengumuman dapat dilihat menggunakan method lihatPengumuman() yang berada pada *class c\_main*

Hasil pengujian integrasi method tambah pengumuman dan melihat pengumuman berdasarkan prosedur uji dapat dilihat pada Tabel 6.5.

**Tabel 6.5 Hasil Pengujian Integrasi Method Tambah Pengumuman dan Melihat Pengumuman**

Input Pertama	Method dari class c_admin	Output Pertama / Input Kedua	Method dari class c_main	Expected Result	Result	Status
Input Judul: Pendaftaran Input Isi: 16 November 2016	tambahPengumuman()	Judul: Pendaftaran Isi: 16 November 2016	lihatPengumuman()	Sistem menampilkan pengumuman dengan Judul: Pendaftaran Isi: 16 November 2016	Sistem menampilkan pengumuman dengan Judul: Pendaftaran Isi: 16 November 2016	Valid

Dari hasil pengujian pada Tabel 6.5 integrasi pada method tambah pengumuman dan melihat pengumuman dapat tervalidasi. Untuk tampilan interface dari hasil uji dapat dilihat pada Gambar 6.8 berikut.



**Gambar 6.8 Tampilan Hasil Pengujian Integrasi Method Tambah Pengumuman dan Melihat Pengumuman**

## 2. Pengujian Integrasi Antara Method Validasi Pembayaran dan Method Lihat Pembayaran

Method validasi pembayaran atau `ubahStatusPembayaran()` adalah method untuk memvalidasi bukti pembayaran yang telah diupload oleh calon siswa. Method ini berada pada kelas `c_stafftu`. Sedangkan, method lihat pembayaran atau `lihatPembayaran()` adalah method untuk melihat bukti pembayaran yang telah diupload beserta hasil validasi yang telah dilakukan oleh staff TU. Method ini berada pada kelas `c_calonsiswa`. Tabel 6.6 menjelaskan interaksi method validasi pembayaran dan lihat pembayaran.

**Tabel 6.6 Interaksi Method Validasi Pembayaran dan Lihat Pembayaran**

No	Method dari Class <code>c_stafftu</code>	Method dari Class <code>c_calonsiswa</code>	Goal
1	<code>ubahStatusPembayaran()</code>	<code>lihatPembayaran()</code>	Melihat status pembayaran yang telah divalidasi / telah diubah statusnya

Prosedur uji yang dilakukan yaitu:

1. Penguji menyiapkan kelas `c_stafftu` untuk mengaktifkan method `ubahStatusPembayaran()`
2. Penguji menyiapkan *input-input* yang diperlukan method `ubahStatusPembayaran()`
3. Penguji mengeksekusi method `ubahStatusPembayaran()`
4. Penguji melihat apakah status pembayaran mengalami perubahan dengan menggunakan method `lihatPembayaran()`

Hasil pengujian integrasi method validasi pembayaran dan lihat pembayaran berdasarkan prosedur uji dapat dilihat pada Tabel 6.7.

**Tabel 6.7 Hasil Pengujian Integrasi Method Validasi Pembayaran dan Lihat Pembayaran**

Input Pertama	Method dari class <code>c_stafftu</code>	Output Pertama / Input Kedua	Method dari class <code>c_main</code>	Expected Result	Result	Status
Diterima	<code>ubahStatusPembayaran()</code>	Diterima	<code>lihatPembayaran()</code>	Sistem menampilkan informasi bahwa pembayaran diterima	Sistem menampilkan informasi bahwa pembayaran diterima	Valid

Dari hasil pengujian pada Tabel 6.7 integrasi pada method validasi pembayaran dan lihat pembayaran dapat tervalidasi. Untuk tampilan interface dari hasil uji dapat dilihat pada Gambar 6.9 berikut.



**Gambar 6.9** Tampilan Hasil Pengujian Integrasi Method Validasi Pembayaran dan Lihat Pembayaran

### 6.1.3 Pengujian Validasi

Pengujian validasi memiliki tujuan untuk mengetahui apakah sistem yang telah dibangun telah menyediakan fungsi-fungsi yang dibutuhkan. Item-item yang telah pada daftar kebutuhan dan merupakan hasil analisis kebutuhan akan digunakan sebagai acuan untuk melakukan pengujian validasi. Pengujian validasi menggunakan pengujian *black box*, karena pengujian ini tidak memfokuskan alur jalannya algoritma program dan lebih ditekankan untuk menemukan kesesuaian antara kinerja sistem dengan daftar kebutuhan sistem. Berikut akan dijelaskan bagaimana proses pengujian validasi untuk tiap-tiap kebutuhan sistem. Dalam melakukan pengujian validasi terdapat kasus-kasus yang digunakan untuk mengetahui kemungkinan-kemungkinan tindakan yang akan dilakukan oleh pengguna sistem. Berikut ini merupakan beberapa kasus yang sebagian besar akan dilakukan oleh pengguna sistem.

1. Kasus Uji Lihat Pengumuman

**Tabel 6.8** Pengujian Lihat Pengumuman

Nama Kasus Uji	Uji Lihat Pengumuman
Objek Uji	Uji Halaman Lihat Pengumuman
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa sistem dapat memenuhi kebutuhan fungsional untuk menampilkan pengumuman yang terdiri dari tanggal, judul, dan isi pengumuman
Prosedur Uji	Menekan menu Pengumuman
Hasil yang Diharapkan	Sistem dapat menampilkan halaman lihat pengumuman yang terdiri dari tanggal, judul, dan isi pengumuman
Hasil Uji Coba	Valid

Sumber : Pengujian



## 2. Kasus Uji Lihat Profil

**Tabel 6.9 Pengujian Lihat Profil**

Nama Kasus Uji	Uji Lihat Profil
Objek Uji	Uji Halaman Lihat Profil
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa sistem dapat memenuhi kebutuhan fungsional untuk menampilkan data calon siswa.
Prosedur Uji	Menekan tombol Biodata
Hasil yang Diharapkan	Sistem dapat menampilkan halaman lihat profil beserta data calon siswa
Hasil Uji Coba	Valid

**Sumber : Pengujian**

## 3. Kasus Uji Edit Profil

**Tabel 6.10 Pengujian Edit Profil**

Nama Kasus Uji	Uji Edit Profil
Objek Uji	Uji Halaman Edit Profil
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa sistem dapat memenuhi kebutuhan fungsional untuk menampilkan form edit profil dan menyimpan data calon siswa yang telah diubah ke dalam database
Prosedur Uji	Menekan tombol Biodata, menekan tombol Edit Biodata, mengisi form, menekan tombol Simpan
Hasil yang Diharapkan	Sistem dapat menampilkan halaman form edit profil dan dapat menyimpan perubahan data ke dalam database
Hasil Uji Coba	Valid

**Sumber : Pengujian**

## 4. Kasus Uji Upload Bukti Pembayaran

**Tabel 6.11 Pengujian Upload Bukti Pembayaran**

Nama Kasus Uji	Uji Upload Bukti Pembayaran
Objek Uji	Uji Halaman Upload Bukti Pembayaran
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa sistem dapat memenuhi kebutuhan fungsional untuk menampilkan halaman untuk upload bukti pembayaran.
Prosedur Uji	Menekan tombol Pembayaran, menekan tombol Pilih File, memilih file, menekan tombol Upload
Hasil yang Diharapkan	Sistem dapat menampilkan halaman upload bukti pembayaran dan dapat menyimpan file bukti pembayaran
Hasil Uji Coba	Valid

**Sumber : Pengujian**

## 5. Kasus Uji Lihat Jadwal Ujian

**Tabel 6.12 Pengujian Lihat Jadwal Ujian**

Nama Kasus Uji	Uji Lihat Jadwal Ujian
Objek Uji	Uji Halaman Lihat Jadwal Ujian

Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa sistem dapat memenuhi kebutuhan fungsional untuk menampilkan informasi mengenai tanggal dan jam ujian calon siswa
Prosedur Uji	Menekan menu Ujian
Hasil yang Diharapkan	Sistem dapat menampilkan halaman lihat jadwal ujian yang menampilkan informasi mengenai tanggal dan jam ujian calon siswa
Hasil Uji Coba	Valid

**Sumber : Pengujian**

#### 6. Kasus Uji Cetak Kartu Ujian

**Tabel 6.13 Pengujian Cetak Kartu Ujian**

Nama Kasus Uji	Uji Cetak Kartu Ujian
Objek Uji	Uji Cetak Kartu Ujian
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa sistem dapat memenuhi kebutuhan fungsional untuk mencetak ( <i>export</i> ) kartu ujian ke dalam sebuah file yang dipilih
Prosedur Uji	Menekan menu Ujian, menekan tombol Cetak Kartu Ujian
Hasil yang Diharapkan	Sistem dapat mencetak kartu ujian calon siswa ke dalam file yang dipilih oleh pengguna
Hasil Uji Coba	Valid

**Sumber : Pengujian**

#### 7. Kasus Uji Buat Soal

**Tabel 6.14 Pengujian Buat Soal**

Nama Kasus Uji	Uji Buat Soal
Objek Uji	Uji Halaman Buat Soal
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa sistem dapat memenuhi kebutuhan fungsional untuk menampilkan form buat soal dan menyimpan soal baru ke database
Prosedur Uji	Menekan menu Soal, menekan tombol Tambah Soal, mengisi form, menekan tombol Tambah soal
Hasil yang Diharapkan	Sistem dapat menampilkan form buat soal dan menyimpan soal baru ke database
Hasil Uji Coba	Valid

**Sumber : Pengujian**

#### 8. Kasus Uji Edit Pengumuman

**Tabel 6.15 Pengujian Edit Pengumuman**

Nama Kasus Uji	Uji Edit Pengumuman
Objek Uji	Uji Halaman Edit Pengumuman
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa sistem dapat memenuhi kebutuhan fungsional untuk menampilkan form edit pengumuman dan dapat menyimpan perubahan data pengumuman baru ke database
Prosedur Uji	Menekan menu Pengumuman, menekan tombol Edit pada pengumuman yang ingin diedit, menekan tombol Simpan
Hasil yang Diharapkan	Sistem dapat menampilkan form edit pengumuman dan dapat menyimpan perubahan data pengumuman baru ke database

Hasil Uji Coba	Valid
----------------	-------

**Sumber : Pengujian**

#### 9. Kasus Uji Menentukan Jumlah Siswa Diterima

**Tabel 6.16 Pengujian Menentukan Jumlah Siswa Diterima**

Nama Kasus Uji	Uji Menentukan Jumlah Siswa Diterima
Objek Uji	Uji Halaman Menentukan Jumlah Siswa Diterima
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa sistem dapat memenuhi kebutuhan fungsional untuk menentukan calon siswa yang diterima berdasarkan jumlah yang telah ditentukan
Prosedur Uji	Menekan menu Jadwal & Ujian, mengisi form pada form Jumlah Siswa yang Diterima, menekan tombol Simpan
Hasil yang Diharapkan	Sistem dapat menentukan calon siswa yang diterima berdasarkan jumlah yang telah ditentukan
Hasil Uji Coba	Valid

**Sumber : Pengujian**

#### 10. Kasus Uji Validasi Pembayaran

**Tabel 6.17 Pengujian Validasi Pembayaran**

Nama Kasus Uji	Uji Validasi Pembayaran
Objek Uji	Uji Halaman Validasi Pembayaran
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa sistem dapat memenuhi kebutuhan fungsional untuk memvalidasi bukti pembayaran calon siswa
Prosedur Uji	Menekan menu Pembayaran, memilih aksi pada bukti pembayaran yang divalidasi
Hasil yang Diharapkan	Sistem dapat memvalidasi bukti pembayaran calon siswa
Hasil Uji Coba	Valid

**Sumber : Pengujian**

### 6.1.4 Pengujian Berbasis ISO 25010

Berdasarkan pada analisa yang telah dilakukan pada rekayasa kebutuhan, pengujian ini dilakukan pada 5 karakteristik yang terdapat pada model kualitas ISO 25010 yaitu *functional suitability*, *performance efficiency*, *usability*, *security*, dan *portability*.

#### 6.1.4.1 *Functional Suitability*

Tujuan dilakukannya pengujian pada karakteristik *functional suitability* adalah untuk memastikan kebutuhan fungsional atau fitur-fitur sistem telah diimplementasikan sesuai dengan yang diharapkan. Karakteristik ini terdiri dari 3 subkarakteristik, antara lain *functional completeness*, *functional correctness*, dan *functional appropriateness*. Ketiga subkarakteristik tersebut dapat diuji dengan Persamaan 6.1

$$X = \frac{I}{P} \tag{6.1}$$

Dengan :

X = Tingkat *functional suitability*

I = Jumlah fitur yang dirancang

P = Jumlah fitur yang berhasil diimplementasikan

Fitur yang diimplementasikan akan diuji oleh penguji. Penguji berjumlah 5 orang. Kelima orang ini adalah orang yang ahli dalam pengembangan sistem berbasis web. Hasil rekapitan pengujian yang dilakukan oleh 5 orang penguji dapat dilihat pada tabel 6.18.

**Tabel 6.18 Rekap Hasil Pengujian *Functional Suitability***

No	Fungsi	Deskripsi	Ya	Tidak
1	Registrasi	Sistem mampu menerima dan menyimpan data registrasi dari calon siswa	5	0
2	Lihat pengumuman	Sistem dapat melihat pengumuman mengenai seleksi penerimaan siswa SMA baru.	5	0
3	Lihat profil	Sistem dapat melihat profil calon siswa.	5	0
4	Edit profil	Sistem dapat mengedit data profil calon siswa	5	0
5	Hapus profil	Sistem dapat menghapus profil calon siswa apabila calon siswa batal mendaftar.	5	0
6	Upload bukti pembayaran	Sistem dapat melakukan upload bukti pembayaran dari Bank atau ATM sebagai bukti bahwa pendaftar telah melakukan pembayaran.	5	0
7	Lihat jadwal ujian	Sistem dapat menampilkan jadwal ujian calon siswa	5	0
8	Cetak kartu ujian	Sistem dapat mencetak kartu ujian.	5	0
9	Kerjakan soal	Sistem dapat menampilkan soal soal ujian seleksi yang nantinya akan dikerjakan calon siswa.	5	0
10	Lihat hasil seleksi	Sistem dapat menampilkan hasil seleksi calon siswa yang login ke dalam sistem.	5	0
11	Lihat profil admin	Sistem dapat menampilkan profil admin	5	0
12	Lihat seluruh admin	Sistem dapat menampilkan data seluruh admin	5	0
13	Tambah user admin	Sistem dapat menambah user untuk admin.	5	0
14	Edit profil admin	Sistem dapat mengedit profil admin	5	0
15	Hapus profil admin	Sistem dapat menghapus profil admin	5	0



Tabel 6.18 Rekap Hasil Pengujian *Functional Suitability* (lanjutan)

No	Fungsi	Deskripsi	Ya	Tidak
16	Lihat seluruh staff TU	Sistem dapat menampilkan data seluruh staff TU	5	0
17	Tambah user staff TU	Sistem dapat menambah user staff TU.	5	0
18	Hapus user staff TU	Sistem dapat menghapus user staff TU	5	0
19	Posting pengumuman	Sistem dapat memposting pengumuman mengenai seleksi penerimaan siswa SMA baru.	5	0
20	Edit pengumuman	Sistem dapat mengedit pengumuman yang telah diposting.	5	0
21	Hapus pengumuman	Sistem dapat menghapus pengumuman yang telah diposting.	5	0
22	Buat soal	Sistem dapat memasukkan soal ujian ke dalam sistem	5	0
23	Edit soal	Sistem dapat mengedit soal ujian.	5	0
24	Hapus soal	Sistem dapat menghapus soal ujian.	5	0
25	Lihat hasil seleksi seluruh calon siswa	Sistem dapat melihat hasil seleksi seluruh peserta ujian.	5	0
26	Menentukan jumlah siswa diterima	Sistem dapat menentukan jumlah pendaftar yang diterima sebagai siswa baru SMA	5	0
27	Menentukan jadwal pendaftaran	Sistem dapat menentukan jadwal pendaftaran.	5	0
28	Menentukan jadwal ujian	Sistem dapat menentukan jadwal ujian	5	0
29	Menentukan jadwal pengumuman seleksi	Sistem dapat menentukan jadwal pengumuman seleksi penerimaan siswa SMA baru.	5	0
30	Lihat profil staff TU	Sistem dapat menampilkan profil staff TU	5	0
31	Edit profil staff TU	Sistem dapat mengubah profil staff TU	5	0
32	Lihat seluruh calon siswa	Sistem dapat melihat data calon siswa	5	0
33	Hapus profil calon siswa	Sistem dapat menghapus profil calon siswa.	5	0
34	Validasi pembayaran	Sistem dapat memvalidasi bukti pembayaran yang telah diupload oleh pendaftar	5	0
35	Login	Sistem dapat menyediakan fitur autentikasi berupa login	5	0
36	Logout	Sistem dapat menyediakan fitur keluar dari sistem	5	0

Berdasarkan hasil rekapan pada tabel 6.18, seluruh fitur yang dirancang berhasil diimplementasikan. Kemudian dapat dimasukkan ke dalam persamaan untuk mencari tingkat *functional suitability*.

$$I = 36$$

$$P = \frac{\text{Jumlah dari hasil tiap penguji}}{\text{Jumlah penguji}}$$

$$P = \frac{36 + 36 + 36 + 36 + 36}{5}$$

$$P = 36$$

$$X = \frac{I}{P}$$

$$X = \frac{36}{36}$$

$$X = 1$$

#### 6.1.4.2 Performance Efficiency

##### a. Time Behaviour

Pengujian pada subkarakteristik ini dilakukan menggunakan *tool* WAPT (*Web Application Load, Stress and Performance Testing*). Beban sistem yang diujikan sebesar 20 *virtual user* dan setiap *user* melakukan satu *action* setiap 30 detik. Jika kriteria yang diuji telah didapatkan nilainya maka akan dibandingkan dengan standar waktu respon Jakob Nielsen. Jika sistem memiliki waktu respon kurang dari 10 detik maka sistem memenuhi subkarakteristik *time behaviour*. Namun jika sistem memiliki waktu respon lebih dari sama dengan 10 detik, maka proses atau antarmuka pada sistem harus disederhanakan. Tabel 6.19 menjelaskan ringkasan hasil pengujian *time behaviour*.

**Tabel 6.19 Ringkasan Hasil Pengujian Time Behaviour**

URL	Rata-Rata Hasil (Detik)
http://localhost/seleksibaru/	0,07
http://localhost/seleksibaru/c_main/pengumuman	0,81
http://localhost/seleksibaru/c_main/about	0,06
http://localhost/seleksibaru/c_main/registrasi	0,11
http://localhost/seleksibaru/c_login	0,06
http://localhost/seleksibaru/c_login/prosesLogin	0,18
http://localhost/seleksibaru/c_main/pengumuman	0,83
http://localhost/seleksibaru/c_calonsiswa/lihatProfil	0,06
http://localhost/seleksibaru/c_calonsiswa/lihatPembayaran	0,08

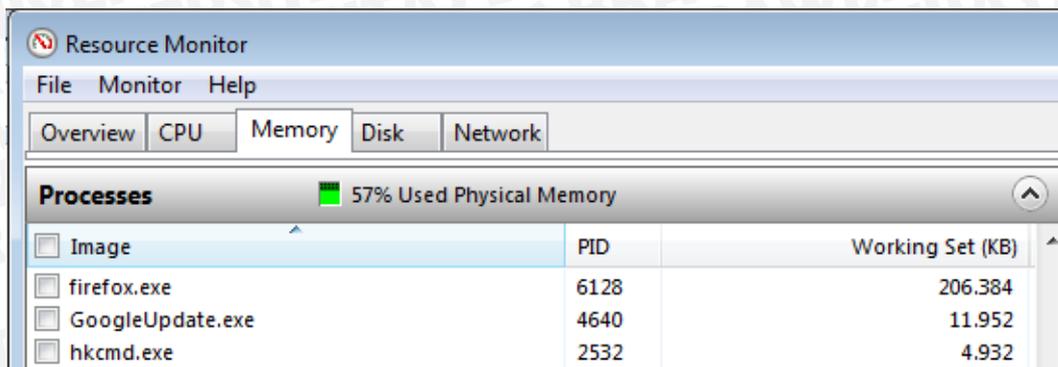
Tabel 6.19 Ringkasan Hasil Pengujian *Time Behaviour* (lanjutan)

URL	Rata-Rata Hasil (Detik)
http://localhost/seleksibaru/c_calonsiswa/cetakKartuUjian	0,18
http://localhost/seleksibaru/c_calonsiswa/ujian	0,07
http://localhost/seleksibaru/c_calonsiswa/lihatHasil	0,06
http://localhost/seleksibaru/c_login/logout	0,07
http://localhost/seleksibaru/c_main	0,07
http://localhost/seleksibaru/c_admin/getListAdmin	0,07
http://localhost/seleksibaru/c_admin/hapusAdmin	0,09
http://localhost/seleksibaru/c_admin/getListStafftu	0,07
http://localhost/seleksibaru/c_admin/hapusStafftu	0,09
http://localhost/seleksibaru/c_admin/getHasilUjian	0,11
http://localhost/seleksibaru/c_admin/lihatPengumuman	0,06
http://localhost/seleksibaru/c_admin/tambahPengumuman	0,09
http://localhost/seleksibaru/c_admin/hapusPengumuman	0,09
http://localhost/seleksibaru/c_admin/lihatSoal	0,06
http://localhost/seleksibaru/c_admin/tambahSoal	0,09
http://localhost/seleksibaru/c_admin/hapusSoal	0,09
http://localhost/seleksibaru/c_admin/kelolaUjian	0,05
http://localhost/seleksibaru/c_admin/tentukanKuota	0,09
http://localhost/seleksibaru/c_admin/tentukanSiswaDiterima	0,08
http://localhost/seleksibaru/c_admin/tentukanJadwal	0,08
http://localhost/seleksibaru/c_stafftu	0,05
http://localhost/seleksibaru/c_stafftu/lihatPembayaran	0,06
http://localhost/seleksibaru/c_stafftu/ubahStatusPembayaran	0,07
http://localhost/seleksibaru/c_stafftu/getListCalonSiswa	0,08
Rata-Rata Seluruh URL (Detik)	0,13

Berdasarkan ringkasan hasil pengujian *time behaviour* pada Tabel 6.18, nilai rata-rata waktu respon sistem seluruh URL adalah 0,13 detik.

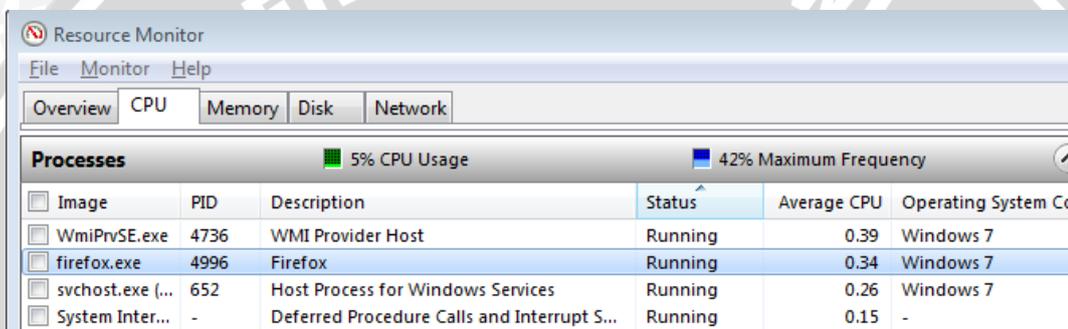
#### b. Resource Utilization

Pengujian pada subkarakteristik ini menggunakan *tool* Resource Monitor. Gambar 6.10 menunjukkan jumlah RAM yang digunakan oleh Mozilla Firefox tanpa membuka aplikasi berbasis web apapun.



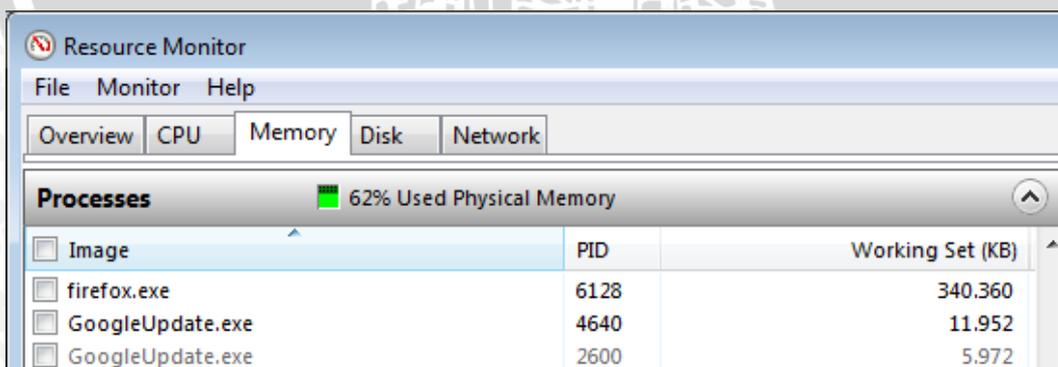
**Gambar 6.10 RAM Yang Digunakan Mozilla Tanpa Membuka Aplikasi Berbasis Web Apapun**

Sedangkan pada gambar 6.11 menunjukkan jumlah CPU yang digunakan oleh Mozilla Firefox tanpa membuka aplikasi berbasis web apapun.



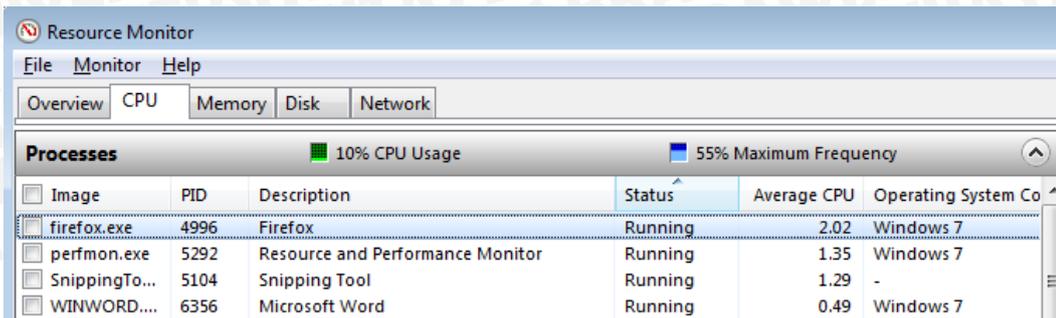
**Gambar 6.11 CPU Yang Digunakan Mozilla Tanpa Membuka Aplikasi Berbasis Web Apapun**

Gambar 6.12 menunjukkan jumlah RAM yang digunakan oleh Mozilla Firefox dengan hanya membuka Sistem Seleksi Penerimaan Siswa Baru SMA.



**Gambar 6.12 RAM Yang Digunakan Mozilla Dengan Hanya Membuka Sistem Seleksi**

Gambar 6.13 menunjukkan jumlah CPU yang digunakan oleh Mozilla Firefox dengan hanya membuka Sistem Seleksi Penerimaan Siswa Baru SMA.



Gambar 6.13 CPU Yang Digunakan Mozilla Dengan Hanya Membuka Sistem Seleksi

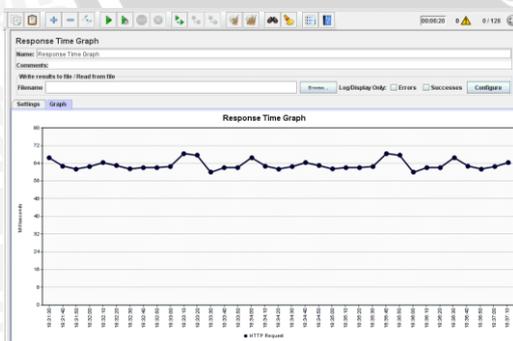
RAM yang dibutuhkan Sistem Seleksi = 340.360 – 206.384  
 = 133.976 KB  
 = 133,976 MB

CPU yang dibutuhkan Sistem Seleksi = 2,02% - 0,34%  
 = 1,68 %

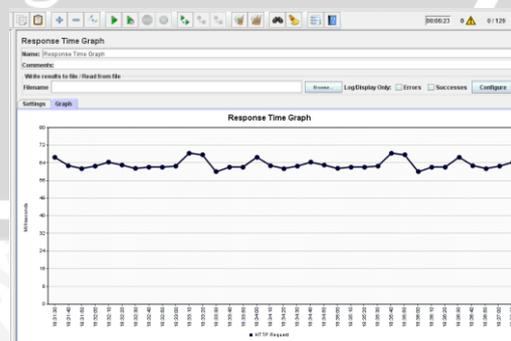
Jadi, RAM yang dibutuhkan oleh Sistem Seleksi Penerimaan Siswa Baru SMA adalah ± 133,976 MB dan CPU yang dibutuhkan oleh Sistem Seleksi Penerimaan Siswa Baru SMA adalah ± 1,68% dengan spesifikasi perangkat keras prosesor Intel Core i3 dan memori (RAM) sebesar 4GB.

**c. Capacity**

Pengujian pada subkarakteristik ini menggunakan *tool* Apache jMeter. Sistem Seleksi diuji dengan beban dimulai 120 *user* dan setiap *user* melakukan *action* setiap 3 detik sekali. 120 *user* dipilih karena pada saat ujian seleksi, jumlah calon siswa yang melakukan ujian seleksi berjumlah 120 calon siswa pada waktu yang sama. Kemudian, jumlah *user* ditambah 1 secara terus menerus sampai batas stabil sistem. Jumlah *user* yang didapatkan pada batas stabil sistem adalah nilai dari *capacity*. Hasil pengujian *capacity* dapat dilihat secara lengkap pada Lampiran. Gambar 6.14 menjelaskan ringkasan hasil pengujian *capacity*.



128 user



129 user





Tabel 6.20 Rekap Hasil Pengisian Kuesioner (lanjutan)

Responden	Total Jawaban				
	STS	TS	RR	S	SS
Responden 8	0	0	0	17	13
Responden 9	0	0	0	10	20
Responden 10	0	0	0	13	17
Responden 11	0	0	0	15	15
Responden 12	0	0	0	9	21
Responden 13	0	0	0	15	15
Responden 14	0	0	0	14	16
Responden 15	0	0	0	15	15
Total	0	0	0	174	276

Berdasarkan hasil rekap pengisian kuesioner pada Tabel 6.20, dapat diketahui skor total.

$$\begin{aligned} \text{Skor}_{\text{total}} &= (J_{\text{ss}} \times 5) + (J_{\text{s}} \times 4) + (J_{\text{RR}} \times 3) + (J_{\text{STS}} \times 1) \\ \text{Skor}_{\text{total}} &= (276 \times 5) + (174 \times 4) + (0 \times 3) + (0 \times 1) \\ \text{Skor}_{\text{total}} &= 696 + 1380 \\ \text{Skor}_{\text{total}} &= 2076 \end{aligned}$$

Setelah skor total didapatkan, persentase skor untuk mendapatkan interpretasi hasil pengujian *Usability* dapat dihitung.

$$\begin{aligned} P_{\text{skor}} &= \frac{\text{Skor}_{\text{total}}}{i \times r \times 5} \times 100\% \\ P_{\text{skor}} &= \frac{2076}{30 \times 15 \times 5} \times 100\% \\ P_{\text{skor}} &= 92,27\% \end{aligned}$$

#### 6.1.4.4 Security

Karakteristik *security* memiliki 5 subkarakteristik, yaitu *confidentiality*, *integrity*, *non-repudiation*, *accountability*, dan *authenticity*. Pengujian hanya dilakukan pada subkarakteristik *confidentiality*, *integrity*, *non-repudiation*, dan *authenticity*.

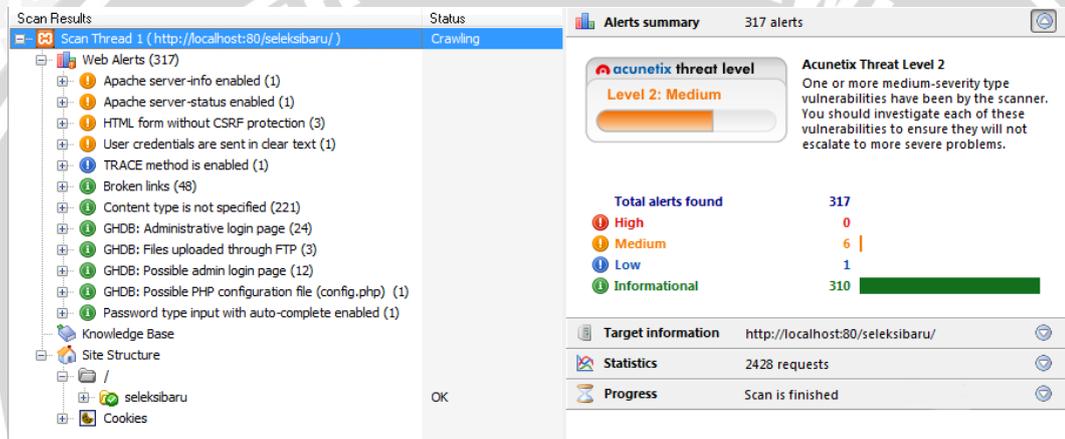
Menurut Arifin (2015) Pengujian *security* dapat dilakukan menggunakan aplikasi Acunetix Web Vulnerability Scanner. Acunetix Web Vulnerability Scanner adalah aplikasi untuk memindai/*scanning* situs, termasuk aplikasi web yang terintegrasi, server web dan server Acunetix Web Vulnerability juga dapat memindai aplikasi berbasis web untuk menemukan *vulnerability* (kerentanan). *Vulnerability*. *Vulnerability* merupakan kelemahan yang memungkinkan penyerang (*attacker*) untuk mengurangi jaminan informasi sebuah sistem.

*Vulnerability* adalah titik temu dari tiga unsur, yaitu kerentanan sistem atau kelemahan sistem, penyerang mengakses kelemahan sistem, dan kemampuan penyerang untuk mengeksploitasi kelemahan sistem (The Three Tenets of Cyber Security, 2009). Dengan demikian, aplikasi Acunetix Web Vulnerability Scanner dapat untuk menguji subkarakteristik *confidentiality*, *integrity*, dan *non-repudiation*.

Subkarakteristik *accountability* diuji dengan melihat sistem, apakah sistem dapat merekam aktifitas pengguna, dan subkarakteristik *authenticity* diuji dengan skenario pengujian *authenticity*.

**a. Confidentiality, Integrity, dan Non-repudiation**

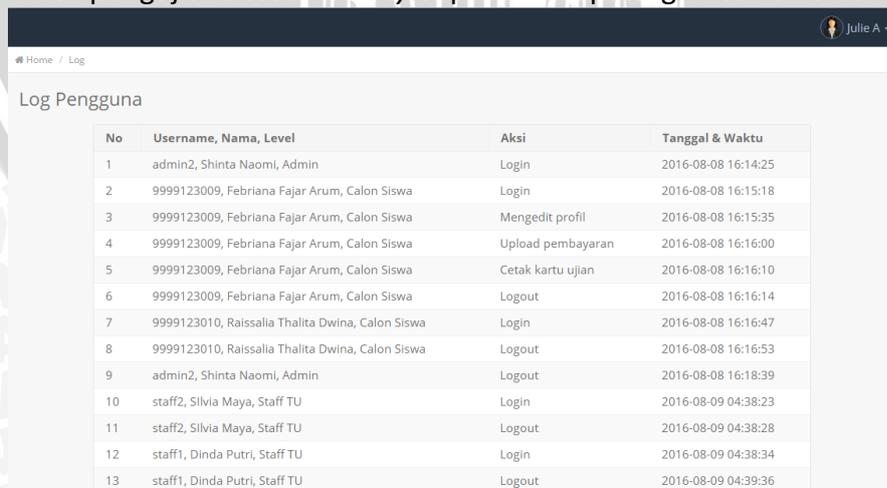
Hasil pengujian *confidentiality*, *integrity*, dan *non-repudiation* dapat dilihat pada gambar 6.15 berikut.



**Gambar 6.15 Hasil Pengujian Confidentiality, Integrity, Dan Non-Repudiation**

**b. Accountability**

Hasil pengujian *accountability* dapat dilihat pada gambar 6.16 berikut.



**Gambar 6.16 Hasil Pengujian Accountability**

Gambar 6.16 merupakan fitur pada sistem seleksi yang berfungsi untuk merekam aktifitas pengguna, seperti login, mengedit profil, upload pembayaran, cetak kartu ujian, logout, dan aktifitas lainnya.

**c. Authenticity**

Hasil pengujian *authenticity* dapat dilihat pada tabel 6.21 berikut.

**Tabel 6.21 Hasil Pengujian Authenticity**

No	Data Input	Hasil yang Diharapkan	Hasil yang Diperoleh	Status
1	Memasukkan NISN "9999123001" pada bagian username dan "123" pada bagian password. Username dan password tersebut adalah milik Dinda Pratiwi	Sistem dapat membuktikan bahwa username "9999123001" dan password "123" adalah milik Dinda Pratiwi dengan cara sistem dapat menampilkan biodata dari Dinda Pratiwi.	Sistem dapat membuktikan bahwa username "9999123001" dan password "123" adalah milik Dinda Pratiwi dengan cara sistem dapat menampilkan biodata dari Dinda Pratiwi.	Valid
2	Memasukkan NISN "9999123002" pada bagian username dan "123" pada bagian password. Username dan password tersebut adalah milik Ulfa Taufany	Sistem dapat membuktikan bahwa username "9999123002" dan password "123" adalah milik Dinda Pratiwi dengan cara sistem dapat menampilkan biodata dari Ulfa Taufany	Sistem dapat membuktikan bahwa username "9999123002" dan password "123" adalah milik Dinda Pratiwi dengan cara sistem dapat menampilkan biodata dari Ulfa Taufany	Valid
3	Memasukkan NISN "9999123003" pada bagian username dan "123" pada bagian password. Username dan password tersebut adalah milik Pradipta Dharma	Sistem dapat membuktikan bahwa username "9999123003" dan password "123" adalah milik Dinda Pratiwi dengan cara sistem dapat menampilkan biodata dari Pradipta Dharma	Sistem dapat membuktikan bahwa username "9999123003" dan password "123" adalah milik Dinda Pratiwi dengan cara sistem dapat menampilkan biodata dari Pradipta Dharma	Valid

Tabel 6.21 Hasil Pengujian *Authenticity* (lanjutan)

No	Data Input	Hasil yang Diharapkan	Hasil yang Diperoleh	Status
4	Memasukkan NISN "9999123004" pada bagian username dan "123" pada bagian password. Username dan password tersebut adalah milik Eko Putro Purwantoko	Sistem dapat membuktikan bahwa username "9999123004" dan password "123" adalah milik Dinda Pratiwi dengan cara sistem dapat menampilkan biodata dari Eko Putro Purwantoko	Sistem dapat membuktikan bahwa username "9999123004" dan password "123" adalah milik Dinda Pratiwi dengan cara sistem dapat menampilkan biodata dari Eko Putro Purwantoko	Valid
5	Memasukkan NISN "9999123004" pada bagian username dan "123" pada bagian password. Username dan password tersebut adalah milik I Gusti Larasati	Sistem dapat membuktikan bahwa username "9999123005" dan password "123" adalah milik Dinda Pratiwi dengan cara sistem dapat menampilkan biodata dari I Gusti Larasati	Sistem dapat membuktikan bahwa username "9999123005" dan password "123" adalah milik Dinda Pratiwi dengan cara sistem dapat menampilkan biodata dari I Gusti Larasati	Valid

#### 6.1.4.5 Portability

Karakteristik ini memiliki 6 subkarakteristik, yaitu *adaptability*, *installability*, dan *replaceability*. Pengujian hanya dilakukan pada subkarakteristik *adaptability*. *Adaptability* pada sistem akan dinilai dengan melakukan pengujian pada kompatibilitas antar *browser*. Pengujian dilakukan dengan menjalankan sistem pada 3 *desktop browser* dan 3 *mobile browser*. *Browser* tersebut yaitu Chrome 53, Firefox 47.0.1, Opera 38.0, iPhone 4S (iOS 9.3.3), Xperia Z (Android 5.1.1) dan Lenovo A536 (Android 4.2.2). Setiap antarmuka yang dimiliki oleh sistem akan dijalankan pada semua *browser* tersebut. Tetapi, tidak semua antarmuka diuji menggunakan *mobile browser*, karena ada beberapa fungsi yang sulit untuk dilakukan apabila menggunakan *mobile browser*, seperti mengerjakan ujian. Jika pada seluruh antarmuka tidak terdapat error, maka sistem telah memenuhi karakteristik *adaptability* sekaligus *portability*.

Hasil pengujian *adaptability* dapat dilihat secara lengkap pada Lampiran. Tabel 6.22 menjelaskan ringkasan hasil pengujian *adaptability*.

Tabel 6.22 Ringkasan Hasil Pengujian *Portability*

No	URL	Status
1	<a href="http://localhost/seleksibaru/">http://localhost/seleksibaru/</a>	Valid
2	<a href="http://localhost/seleksibaru/c_main">http://localhost/seleksibaru/c_main</a>	Valid
3	<a href="http://localhost/seleksibaru/c_main/pengumuman">http://localhost/seleksibaru/c_main/pengumuman</a>	Valid
4	<a href="http://localhost/seleksibaru/c_main/detailPengumuman?id=3">http://localhost/seleksibaru/c_main/detailPengumuman?id=3</a>	Valid
5	<a href="http://localhost/seleksibaru/c_main/about">http://localhost/seleksibaru/c_main/about</a>	Valid
6	<a href="http://localhost/seleksibaru/c_main/registrasi">http://localhost/seleksibaru/c_main/registrasi</a>	Valid
7	<a href="http://localhost/seleksibaru/c_login">http://localhost/seleksibaru/c_login</a>	Valid
8	<a href="http://localhost/seleksibaru/c_calonsiswa">http://localhost/seleksibaru/c_calonsiswa</a>	Valid
9	<a href="http://localhost/seleksibaru/c_calonsiswa/lihatProfil">http://localhost/seleksibaru/c_calonsiswa/lihatProfil</a>	Valid
10	<a href="http://localhost/seleksibaru/c_calonsiswa/lihatPembayaran">http://localhost/seleksibaru/c_calonsiswa/lihatPembayaran</a>	Valid
11	<a href="http://localhost/seleksibaru/c_calonsiswa/addPembayaran">http://localhost/seleksibaru/c_calonsiswa/addPembayaran</a>	Valid
12	<a href="http://localhost/seleksibaru/c_calonsiswa/ujian">http://localhost/seleksibaru/c_calonsiswa/ujian</a>	Valid
13	<a href="http://localhost/seleksibaru/c_calonsiswa/cetakKartuUjian">http://localhost/seleksibaru/c_calonsiswa/cetakKartuUjian</a>	Valid
14	<a href="http://localhost/seleksibaru/c_calonsiswa/lihatHasil">http://localhost/seleksibaru/c_calonsiswa/lihatHasil</a>	Valid
15	<a href="http://localhost/seleksibaru/c_admin">http://localhost/seleksibaru/c_admin</a>	Valid
16	<a href="http://localhost/seleksibaru/c_admin/getListAdmin">http://localhost/seleksibaru/c_admin/getListAdmin</a>	Valid
17	<a href="http://localhost/seleksibaru/c_admin/tambahAdmin">http://localhost/seleksibaru/c_admin/tambahAdmin</a>	Valid
18	<a href="http://localhost/seleksibaru/c_admin/hapusAdmin?id=admin3">http://localhost/seleksibaru/c_admin/hapusAdmin?id=admin3</a>	Valid
19	<a href="http://localhost/seleksibaru/c_admin/getListStafftu">http://localhost/seleksibaru/c_admin/getListStafftu</a>	Valid
20	<a href="http://localhost/seleksibaru/c_admin/tambahStafftu">http://localhost/seleksibaru/c_admin/tambahStafftu</a>	Valid
21	<a href="http://localhost/seleksibaru/c_admin/hapusStafftu?id=staff3">http://localhost/seleksibaru/c_admin/hapusStafftu?id=staff3</a>	Valid
22	<a href="http://localhost/seleksibaru/c_admin/getHasilUjian">http://localhost/seleksibaru/c_admin/getHasilUjian</a>	Valid
23	<a href="http://localhost/seleksibaru/c_admin/lihatPengumuman">http://localhost/seleksibaru/c_admin/lihatPengumuman</a>	Valid
24	<a href="http://localhost/seleksibaru/c_admin/tambahPengumuman">http://localhost/seleksibaru/c_admin/tambahPengumuman</a>	Valid
25	<a href="http://localhost/seleksibaru/c_admin/editPengumuman?id=7">http://localhost/seleksibaru/c_admin/editPengumuman?id=7</a>	Valid
26	<a href="http://localhost/seleksibaru/c_admin/hapusPengumuman?id=7">http://localhost/seleksibaru/c_admin/hapusPengumuman?id=7</a>	Valid
27	<a href="http://localhost/seleksibaru/c_admin/lihatSoal">http://localhost/seleksibaru/c_admin/lihatSoal</a>	Valid
28	<a href="http://localhost/seleksibaru/c_admin/tambahSoal">http://localhost/seleksibaru/c_admin/tambahSoal</a>	Valid
29	<a href="http://localhost/seleksibaru/c_admin/editSoal?id=51">http://localhost/seleksibaru/c_admin/editSoal?id=51</a>	Valid
30	<a href="http://localhost/seleksibaru/c_admin/hapusSoal?id=51">http://localhost/seleksibaru/c_admin/hapusSoal?id=51</a>	Valid
31	<a href="http://localhost/seleksibaru/c_admin/kelolaUjian">http://localhost/seleksibaru/c_admin/kelolaUjian</a>	Valid
32	<a href="http://localhost/seleksibaru/c_admin/tentukanKuota">http://localhost/seleksibaru/c_admin/tentukanKuota</a>	Valid
33	<a href="http://localhost/seleksibaru/c_admin/tentukanSiswaDiterima">http://localhost/seleksibaru/c_admin/tentukanSiswaDiterima</a>	Valid
34	<a href="http://localhost/seleksibaru/c_stafftu">http://localhost/seleksibaru/c_stafftu</a>	Valid
35	<a href="http://localhost/seleksibaru/c_stafftu/getListCalonSiswa">http://localhost/seleksibaru/c_stafftu/getListCalonSiswa</a>	Valid
36	<a href="http://localhost/seleksibaru/c_stafftu/lihatPembayaran">http://localhost/seleksibaru/c_stafftu/lihatPembayaran</a>	Valid
37	<a href="http://localhost/seleksibaru/c_stafftu/ubahStatusPembayaran">http://localhost/seleksibaru/c_stafftu/ubahStatusPembayaran</a>	Valid

Berdasarkan ringkasan hasil pengujian *portability* pada Tabel 6.22, seluruh antarmuka sistem tidak terdapat error.

## 6.2 Analisis

Analisis dilakukan untuk mendapatkan kesimpulan dari hasil uji yang dilakukan oleh sistem, yakni pengujian unit, pengujian integrasi, pengujian validasi, dan pengujian berbasis ISO 25010. Proses analisis dilakukan sesuai dengan pengujian yang dilakukan pada sistem yakni analisis hasil uji pengujian unit, analisis hasil uji pengujian integrasi, analisis uji pengujian validasi dan analisis hasil uji pengujian berbasis ISO 25010.

### 6.2.1 Analisis Hasil Uji Pengujian Unit

Dari kasus uji yang telah dilaksanakan sesuai dengan prosedur pengujian basis path yang telah disebutkan dalam sub pokok bahasan 6.1.1 pada algoritma penilaian, algoritma menentukan kuota per gelombang ujian, dan algoritma menentukan jadwal menunjukkan bahwa seluruh kasus uji memiliki hasil yang telah sesuai dengan yang diharapkan atau bernilai valid.

### 6.2.2 Analisis Hasil Uji Pengujian Integrasi

Berdasarkan hasil pengujian integrasi, proses inti yang melibatkan interaksi antarkelas, yaitu integrasi antara method tambah pengumuman dan melihat pengumuman serta integrasi method validasi pembayaran dan lihat pembayaran pada sistem telah terintegrasi dengan baik.

### 6.2.3 Analisis Hasil Pengujian Validasi

Dari kasus uji yang telah dilaksanakan sesuai dengan prosedur pengujian validasi yang telah disebutkan dalam sub pokok bahasan 6.1.3 maka didapatkan hasil seperti yang ditunjukkan dalam tabel 6.23 berikut.

**Tabel 6.23 Hasil Pengujian Validasi**

No.	Kasus Uji	Hasil yang Didapatkan	Status
1.	Uji Lihat Pengumuman	Sistem dapat menampilkan halaman lihat pengumuman yang terdiri dari tanggal, judul, dan isi pengumuman	Valid
2.	Uji Lihat Profil	Sistem dapat menampilkan halaman lihat profil beserta data calon siswa	Valid
3.	Uji Edit Profil	Sistem dapat menampilkan halaman form edit profil dan dapat menyimpan perubahan data ke dalam database	Valid
4.	Uji Upload Bukti Pembayaran	Sistem dapat menampilkan halaman upload bukti pembayaran dan dapat menyimpan file bukti pembayaran	Valid
5.	Uji Lihat Jadwal Ujian	Sistem dapat menampilkan halaman lihat jadwal ujian yang menampilkan informasi mengenai tanggal dan jam ujian calon siswa	Valid
6.	Uji Cetak Kartu Ujian	Sistem dapat mencetak kartu ujian calon siswa ke dalam file yang dipilih oleh pengguna	Valid

**Tabel 6.23 Hasil Pengujian Validasi (lanjutan)**

No.	Kasus Uji	Hasil yang Didapatkan	Status
7.	Uji Buat Soal	Sistem dapat menampilkan form buat soal dan menyimpan soal baru ke database	Valid
8.	Uji Edit Pengumuman	Sistem dapat menampilkan form edit pengumuman dan dapat menyimpan perubahan data pengumuman baru ke database	Valid
9.	Uji Menentukan Jumlah Siswa Diterima	Sistem dapat menentukan calon siswa yang diterima berdasarkan jumlah yang telah ditentukan	Valid
10.	Uji Validasi Pembayaran	Sistem dapat memvalidasi bukti pembayaran calon siswa	Valid

Berdasarkan hasil pengujian validasi, seluruh kasus uji yang didefinisikan telah valid.

## 6.2.4 Analisis Hasil Uji Pengujian Berbasis ISO 25010

### 6.2.4.1 Analisis Hasil Uji Pengujian *Functional Suitability*

Berdasarkan hasil pengujian pada karakteristik *functional suitability* didapatkan tingkat *functional suitability* sebesar 1. Untuk memenuhi standar *functional suitability*, diperlukan tingkat *functional suitability* yang mendekati atau bernilai 1. Sehingga, dari hasil pengujian tersebut maka sistem seleksi penerimaan siswa SMA baru telah memenuhi standar untuk karakteristik *functional suitability*.

### 6.2.4.2 Analisis Hasil Uji Pengujian *Performance Efficiency*

#### 1. Analisis Hasil Uji Pengujian *Time Behaviour*

Berdasarkan hasil pengujian pada subkarakteristik *time behaviour*, didapatkan rata-rata waktu respons sistem sebesar 0,13 detik. Kemudian rata-rata waktu respon tersebut dibandingkan dengan Tabel Standar Jakob Nielsen. Tabel 6.24 menunjukkan Tabel Standar Jakob Nielsen.

**Tabel 6.24 Standar Jakob Nielsen**

Waktu Respon	Pandangan Pengguna
<0,1 detik	Pengguna merasakan respon yang sangat cepat dari web
<1,0 detik	Pengguna merasakan jeda namun masih fokus kepada web tersebut
<10 detik	Perhatian pengguna terhadap web akan menurun tajam
≥ 10 detik	Kemungkinan besar pengguna akan beralih dari web tersebut

Sehingga jika hasil tersebut dibandingkan dengan standar Jakob Nielsen, maka dapat dikatakan pengguna merasakan adanya jeda namun masih fokus dalam

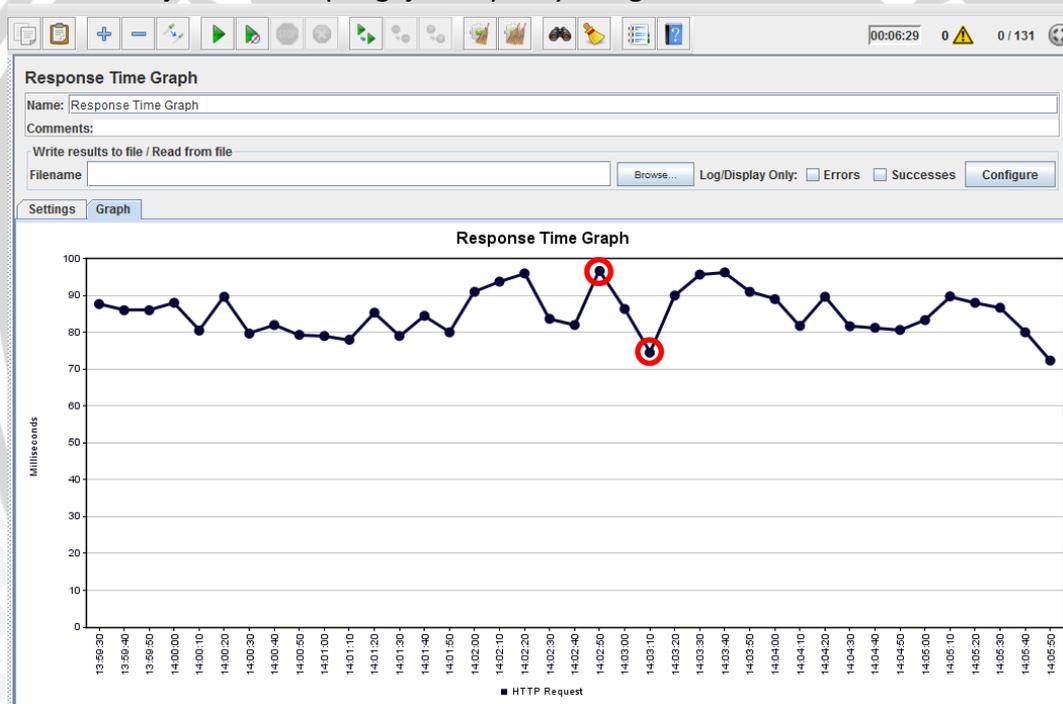
sistem. Dan untuk memenuhi standar, dibutuhkan nilai kurang dari 10 detik. Dengan demikian, dapat disimpulkan bahwa sistem seleksi penerimaan siswa SMA baru memenuhi subkarakteristik *time behaviour*.

## 2. Analisis Hasil Uji Pengujian *Resource Utilization*

Berdasarkan hasil pengujian pada subkarakteristik *resource utilization*, jumlah RAM yang dibutuhkan oleh Sistem Seleksi Penerimaan Siswa Baru SMA adalah  $\pm 133,976$  MB dan CPU yang dibutuhkan oleh Sistem Seleksi Penerimaan Siswa Baru SMA adalah  $\pm 1,68\%$  dengan spesifikasi perangkat keras prosesor Intel Core i3 dan memori (RAM) sebesar 4GB.

## 3. Analisis Hasil Uji Pengujian *Capacity*

Berdasarkan hasil pengujian pada subkarakteristik *capacity*, sistem yang diberi beban 131 user menunjukkan hasil *response time* yang tidak stabil. Gambar 6.17 menunjukkan hasil pengujian *capacity* dengan 131 user.



Gambar 6.17 Hasil Pengujian *Capacity* Dengan 131 User

Ketidakstabilan *response time* apabila diberi beban 131 user dapat dilihat pada detik ke 210 (14:02:50) dengan *response time* sebesar 98 ms dan pada detik ke 230 (14:03:10) dengan *response time* sebesar 74 ms. Dengan demikian, *capacity* dari Sistem Seleksi Penerimaan Siswa Baru SMA adalah 130 user.

### 6.2.4.3 Analisis Hasil Uji Pengujian *Usability*

Berdasarkan hasil pengujian pada karakteristik *usability*, didapatkan prosentase skor sebesar 92,27%. Kemudian, prosentase skor tersebut

dibandingkan dengan tabel kategori penilaian *usability*. Tabel 6.25 menunjukkan tabel kategori penilaian *usability*.

**Tabel 6.25 Kategori Penilaian *Usability***

Presentase Skor	Interpretasi
0% - 20%	Sangat Tidak Layak
21% - 40%	Tidak Layak
41% - 60%	Cukup Layak
61% - 80%	Layak
81% - 100%	Sangat Layak

Apabila prosentase skor tersebut dibandingkan dengan tabel kategori penilaian *usability*, maka dapat dikatakan sistem sangat layak. Dengan demikian, dapat disimpulkan bahwa sistem seleksi memenuhi karakteristik *usability*.

#### 6.2.4.4 Analisis Hasil Uji Pengujian *Security*

##### 1. Analisis Hasil Uji Pengujian *Confidentiality, Integrity, dan Non-repudiation*

Berdasarkan hasil pengujian pada subkarakteristik *confidentiality, integrity, dan non-repudiation*, didapatkan hasil yaitu tingkat keamanan level 2 (medium). Keamanan dengan tingkat level 2 menandakan adanya kesalahan pada konfigurasi server dan kekurangan pengodean pada aplikasi sehingga terjadi gangguan pada server (Acunetix User Manual Book, 2015). Dan untuk memenuhi standar, dibutuhkan hasil tingkat keamanan level 3 (high). Dengan demikian, sistem seleksi memenuhi subkarakteristik *confidentiality, integrity, dan non-repudiation*.

##### 2. Analisis Hasil Uji Pengujian *Accountability*

Berdasarkan hasil pengujian pada subkarakteristik *accountability*, sistem dapat merekam aktifitas pengguna. Dengan adanya fitur yang dapat merekam aktifitas pengguna, apabila ada pengguna yang tidak bertanggung jawab menghapus data data pada sistem seleksi, maka pengguna yang tidak bertanggung jawab tersebut dapat diketahui. Dengan demikian, sistem seleksi telah memenuhi subkarakteristik *accountability* karena sistem dapat merekam aktifitas pengguna.

##### 3. Analisis Hasil Uji Pengujian *Authenticity*

Dari kasus uji yang telah dilaksanakan pada sub pokok bahasan 6.1.4 menunjukkan bahwa seluruh kasus uji memiliki hasil yang telah sesuai dengan yang diharapkan atau bernilai valid. Dengan demikian, sistem seleksi memenuhi subkarakteristik *authenticity*.

#### 6.2.4.5 Analisis Hasil Uji Pengujian *Portability*

Berdasarkan hasil pengujian pada karakteristik *portability*, semua antarmuka dapat dibuka dan dijalankan tanpa adanya *error*. Dan untuk dapat memenuhi standar *portability*, diperlukan sistem yang memiliki antarmuka yang dapat berjalan dengan baik tanpa adanya *error* jika dioperasikan di berbagai *browser*. Dengan demikian, sistem seleksi telah memenuhi karakteristik *portability*.

Keuntungan yang didapatkan dengan menerapkan ISO 25010 berdasarkan hasil pengujian yang sudah dilakukan adalah fitur sistem yang berhasil diimplementasikan sesuai dengan fitur yang dirancang, waktu respon sistem memenuhi standar, yaitu standar Jakob Nielsen, dapat mengetahui jumlah RAM yang dibutuhkan oleh sistem, dapat mengetahui jumlah *user* yang dapat ditangani oleh sistem, sistem mudah dipelajari dan mudah digunakan, sistem lebih aman karena aktifitas seluruh pengguna direkam oleh sistem dan sistem dapat dijalankan tidak hanya pada satu *browser*, tetapi pada beberapa *browser* yang berbeda.



## BAB 7 PENUTUP

### 7.1 Kesimpulan

Berdasarkan hasil rekayasa kebutuhan, perancangan, implementasi dan pengujian sistem yang telah dilakukan, maka dapat diambil kesimpulan sebagai berikut:

1. Pembangunan Sistem Seleksi Penerimaan Siswa Baru SMA Berbasis Web diawali dengan melakukan wawancara ke guru SMA Negeri 1 Purwosari serta observasi langsung ke SMA Negeri 1 Purwosari. Proses rekayasa kebutuhan diawali dari tahap elisitasi kebutuhan, spesifikasi kebutuhan, validasi, manajemen kebutuhan dan terakhir pemodelan kebutuhan.
2. Proses perancangan Sistem Seleksi Penerimaan Siswa Baru SMA terdiri dari perancangan arsitektur sistem, perancangan diagram sekuen, perancangan diagram kelas, pemodelan data, perancangan komponen dan perancangan antarmuka.
3. Sistem diimplementasikan berbasis web dengan menggunakan *framework* CodeIgniter dan template bootstrap serta database MySQL. Dengan adanya sistem ini, calon siswa lebih mudah untuk melakukan registrasi, kecurangan dalam ujian seleksi dapat dikurangi, dan calon siswa lebih mudah melihat hasil seleksi penerimaan siswa baru.
4. Berdasarkan hasil pengujian berbasis ISO 25010, Sistem Seleksi Penerimaan Siswa Baru SMA telah memenuhi standar dari karakteristik ISO 25010 yang telah ditentukan. Pada karakteristik *functional suitability*, didapatkan nilai 1 yang berarti seluruh fungsi berjalan dengan baik. Kemudian pada subkarakteristik *time behaviour*, didapatkan rata-rata waktu respons sistem sebesar 0,13 detik. Pada subkarakteristik *resource utilization*, didapatkan jumlah RAM yang dibutuhkan sistem sebesar  $\pm 133,976$  MB dan CPU yang dibutuhkan sistem sebesar  $\pm 1,68\%$ . Lalu pada subkarakteristik *capacity*, didapatkan hasil bahwa *capacity* sistem adalah 130 user. Kemudian pada karakteristik *usability* yang diujikan dengan menggunakan model kuesioner USE *Questionnaire*, didapatkan nilai sebesar 92,27% yang berarti sistem sangat layak. Pada subkarakteristik *confidentiality*, *integrity*, dan *non-repudiation* yang diujikan dengan *tool* Acunetix Web Vulnerability Scanner, didapatkan hasil keamanan level 2 (medium). Kemudian pada subkarakteristik *accountability*, didapatkan hasil bahwa sistem dapat merekam aktifitas pengguna. Lalu pada subkarakteristik *authenticity*, didapatkan hasil bahwa bahwa seluruh kasus uji memiliki hasil yang telah sesuai dengan yang diharapkan atau bernilai valid. Dan pada karakteristik *portability*, didapatkan hasil bahwa semua antarmuka dapat dibuka dan dijalankan tanpa adanya *error*.

## 7.2 Saran

Saran yang dapat diberikan setelah menyelesaikan penelitian ini antara lain:

1. Sistem Seleksi Penerimaan Siswa Baru SMA ini sebaiknya dapat diimplementasikan di semua SMA di Kabupaten Pasuruan
2. Untuk mendapatkan hasil uji yang lebih detail, pengujian dilakukan pada seluruh karakteristik yang ada pada ISO 25010.



## DAFTAR PUSTAKA

- Abeyasinghe, S. 2009. *PHP Team Development: Easy and Effective Team Work using MVC, Agile Development, Source Control, Testing, Bug Tracking, and More*. Birmingham: PACKT Publishing.
- Acharya, A., Sinha, D., 2013. Assessing the Quality of M-Learning Systems using ISO/IEC 25010. *International Journal of Advanced Computer Research* 3, 67.
- Acunetix, 2015. *Acunetix Web Vulnerability Scanner Manual*. Tersedia di : [www.acunetix.com/resources/wvsmanual.pdf](http://www.acunetix.com/resources/wvsmanual.pdf) diakses pada 8 Juli 2016.
- Alves, T.L., Silva, P., Dias, M.S., 2014. Applying ISO/IEC 25010 Standard to Prioritize and Solve Quality Issues of Automatic ETL Processes. *IEEE*, pp. 573–576. doi:10.1109/ICSME.2014.98
- Anicas, Mitchell. 2014. How To Use Apache JMeter To Perform Load Testing on a Web Server. Tersedia di: <https://www.digitalocean.com/community/tutorials/how-to-use-apache-jmeter-to-perform-load-testing-on-a-web-server> [Diakses 8 September 2016]
- Arifin, Yanuar. 2015. *Pengembangan Aplikasi Pengelolaan Data Prestasi Mahasiswa Universitas Negeri Yogyakarta*. S1. Universitas Negeri Yogyakarta.
- Aulia, R.S., 2014. Rancang Bangun Sistem Informasi Pemetaan Sebaran Guru Kabupaten Lumajang. Skripsi sarjana komputer PTIIK Universitas Brawijaya.
- Cs.utsa.edu. Unified Modeling Language. <<http://www.cs.utsa.edu/~cs3443/uml/uml.html>>. [Diakses 25 Mei 2016].
- Dalbey, Dr. John. 2003. Pseudocode Standard. Tersedia di : [http://users.csc.calpoly.edu/~jdalbey/SWE/pdl\\_std.html](http://users.csc.calpoly.edu/~jdalbey/SWE/pdl_std.html) diakses pada 31 Mei 2016
- Fowler, M., 2004. “UML Distilled Edisi 3 Panduan Singkat Bahasa Pemodelan Objek Standar”. Yogyakarta : Penerbit Andi.
- Gordon, Whitson, et all. 2014. Monitoring Your PC with Resource Monitor and Task Manager. Tersedia di: <http://www.howtogeek.com/school/using-windows-admin-tools-like-a-pro/lesson6/all/> [Diakses 8 September 2016]
- International Standart Office, 25000. *ISO/IEC 25000*. System and Software Quality Requirements and Evaluation (SQuaRE). Tersedia di: <http://iso25000.com/> [Diakses 24 November 2015].
- J. P. Miguel, D. Mauricio, and G. Rodríguez, 2014. “A Review of Software Quality Models for the Evaluation of Software Products,” *International Journal of Software Engineering & Applications*, vol. 5, no. 6, pp. 31–53.
- Jamwal, D., 2010. Analysis of Software Quality Models for Organizations. *International Journal of Latest Trends in Computing*, 1(2), pp. 19-23.

- Laitenberger, O. dan Dreyer, H.M. 2013. *Evaluating the Usefulness and the Ease of Use of a Web-based Inspection Data Collection Tool*. Tersedia di: <<http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.31.263&rep=rep1&type=pdf>> [Diakses 12 Januari 2016].
- Lesmana, A.R., 2013. *Efektifitas Bimbingan Kelompok Melalui Teknik Bercerita Untuk Mengembangkan Karakter Siswa*. S1. Universitas Pendidikan Indonesia. Tersedia di <<http://repository.upi.edu/>> [Diakses 25 Januari 2016]
- Lund, A.M. 2001. *Measuring Usability with the USE Questionnaire*. [online] Tersedia di: <<http://garyperlman.com/quest/quest.cgi?form=USE>> [Diakses 10 Mei 2015].
- Motive Glosarry. 2010. *The Motive Web Design Glossary*. Tersedia di: <<http://www.motive.co.nz/glossary/usable.php>> [Diakses 12 Januari 2016].
- Nielsen, Jacob. 2012. How Many Test Users in a Usability Study?. Tersedia di: <<https://www.nngroup.com/articles/how-many-test-users/>> [Diakses 20 September 2016]
- Pressman, R.S., 2010. *Software Engineering A Practitioner's Approach*. Seventh Edition. New York: The McGraw-Hill Companies, Inc.,
- Rosa A.S. , M. Salahudin, 2014. "Rekayasa Perangkat Lunak", Bandung : Informatika Bandung.
- Sholikhah, P., Ghozali, K., 2014. *Rancang Bangun Sistem Informasi Penerimaan Siswa Baru Online, Modul: Rekomendasi*. S1. Institut Teknologi Sepuluh Nopember.
- Sommerville, Software engineering, 9th ed. 2011. Boston: Pearson.
- Suharsono, T.N. & Rahayu, W. 2014. *Perangkat Lunak Pendaftaran Siswa Baru (PSB) Online Di SMA Negeri 1 Baleendah*. Bandung: Politeknik Komputer Niaga.
- Sukamto, R.A. & Salahuddin, M. 2013. *Rekayasa Perangkat Lunak: Terstruktur dan Berorientasi Objek*. Bandung: Informatika.
- The Three Tenets of Cyber Security, 2009. *U.S. Air Force Software Protection Initiative*. Tersedia di: <<http://www.spi.dod.mil/tenets.htm>> [Diakses 14 Januari 2016].
- Tribowo, Anhar. 2015. *Pembangunan Sistem Pengelolaan Restoran Terintegrasi dengan Pengujian Kualitas Berbasis ISO 9126*. S1. Universitas Brawijaya.
- Ulum, Asep Saepul. 2010. Daftar Simbol Enterprise Architect. <[http://elib.unikom.ac.id/files/disk1/459/jbptunikompp-gdl-asepseaful-22933-12-unikom\\_a-1.pdf](http://elib.unikom.ac.id/files/disk1/459/jbptunikompp-gdl-asepseaful-22933-12-unikom_a-1.pdf)> [Diakses 5 Mei 2015].
- Wagner, S., 2013. *Software Product Quality Control*. Springer Berlin Heidelberg, Berlin, Heidelberg.

Wahono, R.S., 2006. *Teknik Pengukuran Kualitas Perangkat Lunak*. [online].  
Tersedia di: <<http://romisatriawahono.net/2006/06/05/teknik-pengukuran-kualitas-perangkat-lunak/>> [Diakses 25 Januari 2016].



## LAMPIRAN A FORM PENGUJIAN FUNCTIONAL SUITABILITY

### PENGUJIAN FUNCTIONAL SUITABILITY SISTEM SELEKSI PENERIMAAN SISWA SMA BARU

Nama : \_\_\_\_\_

Umur : \_\_\_\_\_

Berilah tanda (√) pada kolom yang disediakan sesuai dengan penilaian untuk pengujian *functional suitability* dengan ketentuan penilaian sebagai berikut:

Ya = Jika berfungsi dengan benar

Tidak = Jika tidak berfungsi dengan benar

No	Fungsi	Deskripsi	Ya	Tidak
1	Membuat data calon siswa	Sistem mampu menerima dan menyimpan data registrasi dari calon siswa		
2	Lihat pengumuman	Sistem dapat melihat pengumuman mengenai seleksi penerimaan siswa SMA baru.		
3	Lihat profil	Sistem dapat melihat profil calon siswa.		
4	Edit profil	Sistem dapat mengedit data profil calon siswa		
5	Hapus profil	Sistem dapat menghapus profil calon siswa apabila calon siswa batal mendaftar.		
6	Upload bukti pembayaran	Sistem dapat melakukan upload bukti pembayaran dari Bank atau ATM sebagai bukti bahwa pendaftar telah melakukan pembayaran.		
7	Lihat jadwal ujian	Sistem dapat menampilkan jadwal ujian calon siswa		
8	Cetak kartu ujian	Sistem dapat mencetak kartu ujian.		
9	Kerjakan soal	Sistem dapat menampilkan soal soal ujian seleksi yang nantinya akan dikerjakan calon siswa.		
10	Lihat hasil seleksi	Sistem dapat menampilkan hasil seleksi calon siswa yang login ke dalam sistem.		
11	Lihat profil admin	Sistem dapat menampilkan profil admin		
12	Lihat seluruh admin	Sistem dapat menampilkan data seluruh admin		
13	Tambah user admin	Sistem dapat menambah user untuk admin.		
14	Edit profil admin	Sistem dapat mengedit profil admin		
15	Hapus profil admin	Sistem dapat menghapus profil admin		
16	Lihat seluruh staff TU	Sistem dapat menampilkan data seluruh staff TU		
17	Tambah user staff TU	Sistem dapat menambah user staff TU.		
18	Hapus user staff TU	Sistem dapat menghapus user staff TU		
19	Posting pengumuman	Sistem dapat memposting pengumuman mengenai seleksi penerimaan siswa SMA baru.		
20	Edit pengumuman	Sistem dapat mengedit pengumuman yang telah diposting.		
21	Hapus pengumuman	Sistem dapat menghapus pengumuman yang telah diposting.		
22	Buat soal	Sistem dapat memasukkan soal ujian ke dalam sistem		
23	Edit soal	Sistem dapat mengedit soal ujian.		

24	Hapus soal	Sistem dapat menghapus soal ujian.		
25	Lihat hasil seleksi seluruh calon siswa	Sistem dapat melihat hasil seleksi seluruh peserta ujian.		
26	Menentukan jumlah siswa diterima	Sistem dapat menentukan jumlah pendaftar yang diterima sebagai siswa baru SMA		
27	Menentukan jadwal pendaftaran	Sistem dapat menentukan jadwal pendaftaran.		
28	Menentukan jadwal ujian	Sistem dapat menentukan jadwal ujian		
29	Menentukan jadwal pengumuman seleksi	Sistem dapat menentukan jadwal pengumuman seleksi penerimaan siswa SMA baru.		
30	Lihat profil staff TU	Sistem dapat menampilkan profil staff TU		
31	Edit profil staff TU	Sistem dapat mengubah profil staff TU		
32	Lihat seluruh calon siswa	Sistem dapat melihat data calon siswa		
33	Hapus profil calon siswa	Sistem dapat menghapus profil calon siswa.		
34	Validasi pembayaran	Sistem dapat memvalidasi bukti pembayaran yang telah diupload oleh pendaftar		
35	Login	Sistem dapat menyediakan fitur autentikasi berupa login		
36	Logout	Sistem dapat menyediakan fitur keluar dari sistem		



## LAMPIRAN B HASIL PENGUJIAN TIME BEHAVIOUR

Response time, sec (with page elements)		0:00:00-0:01:00	0:01:00-0:02:00	0:02:00-0:03:00	0:03:00-0:04:00	0:04:00-0:05:00	0:05:00-0:06:00	0:06:00-0:07:00	0:07:00-0:08:00	0:08:00-0:09:00	0:09:00-0:10:00	Total
TimeBehaviour.page_1: http://localhost/seleksibaru/		6	16	21	31	36	42	51	57	61	60	381
Min		0.05(0.05)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)
Max		0.08(0.08)	0.08(0.08)	0.08(0.08)	0.09(0.09)	0.08(0.08)	0.11(0.11)	0.09(0.09)	0.14(0.14)	0.75(0.75)	1.12(1.12)	1.12(1.12)
Avg		0.07(0.07)	0.06(0.06)	0.05(0.05)	0.05(0.05)	0.05(0.05)	0.05(0.05)	0.05(0.05)	0.05(0.05)	0.09(0.09)	0.15(0.15)	0.07(0.07)
Avg90		0.08(0.08)	0.08(0.08)	0.07(0.07)	0.07(0.07)	0.06(0.06)	0.07(0.07)	0.07(0.07)	0.07(0.07)	0.22(0.22)	0.43(0.43)	0.20(0.20)
TimeBehaviour.page_2: http://localhost/seleksibaru/c_main/pengumuman		6	15	21	31	37	42	50	57	61	60	380
Min		0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)
Max		0.06(0.06)	0.08(0.08)	0.08(0.08)	0.09(0.09)	0.08(0.08)	0.08(0.08)	0.17(0.17)	0.25(0.25)	0.42(0.42)	0.48(0.48)	0.48(0.48)
Avg		0.05(0.05)	0.06(0.06)	0.05(0.05)	0.05(0.05)	0.05(0.05)	0.05(0.05)	0.05(0.05)	0.06(0.06)	0.08(0.08)	0.11(0.11)	0.07(0.07)
Avg90		0.06(0.06)	0.08(0.08)	0.07(0.07)	0.07(0.07)	0.07(0.07)	0.07(0.07)	0.08(0.08)	0.10(0.10)	0.17(0.17)	0.24(0.24)	0.14(0.14)
TimeBehaviour.page_3: http://localhost/seleksibaru/c_main/about		6	14	21	32	37	40	52	57	60	60	379
Min		0.05(0.05)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)
Max		0.08(0.08)	0.09(0.09)	0.08(0.08)	0.08(0.08)	0.08(0.08)	0.09(0.09)	0.08(0.08)	0.09(0.09)	1.12(1.12)	0.30(0.30)	1.12(1.12)
Avg		0.07(0.07)	0.06(0.06)	0.05(0.05)	0.05(0.05)	0.04(0.04)	0.05(0.05)	0.05(0.05)	0.05(0.05)	0.09(0.09)	0.09(0.09)	0.06(0.06)
Avg90		0.08(0.08)	0.08(0.08)	0.07(0.07)	0.07(0.07)	0.06(0.06)	0.07(0.07)	0.07(0.07)	0.07(0.07)	0.30(0.30)	0.17(0.17)	0.15(0.15)
TimeBehaviour.page_4: http://localhost/seleksibaru/c_main/registrasi		6	14	21	31	38	39	52	58	59	60	378
Min		0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)
Max		0.08(0.11)	0.08(0.11)	0.08(0.11)	0.08(0.14)	0.09(0.13)	0.08(0.11)	0.13(0.17)	0.30(0.45)	0.41(0.61)	0.70(0.94)	0.70(0.94)
Avg		0.05(0.08)	0.06(0.09)	0.05(0.08)	0.05(0.09)	0.05(0.08)	0.05(0.08)	0.05(0.09)	0.06(0.10)	0.09(0.14)	0.12(0.20)	0.07(0.11)
Avg90		0.08(0.11)	0.08(0.11)	0.07(0.10)	0.07(0.11)	0.07(0.10)	0.07(0.10)	0.07(0.11)	0.10(0.17)	0.20(0.31)	0.30(0.46)	0.16(0.25)
TimeBehaviour.page_5: http://localhost/seleksibaru/c_login		6	14	21	30	38	40	51	58	59	60	377
Min		0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)
Max		0.08(0.08)	0.08(0.08)	0.06(0.06)	0.06(0.06)	0.08(0.08)	0.11(0.11)	0.09(0.09)	0.36(0.36)	0.44(0.44)	0.50(0.50)	0.50(0.50)
Avg		0.06(0.06)	0.05(0.05)	0.04(0.04)	0.04(0.04)	0.05(0.05)	0.05(0.05)	0.05(0.05)	0.06(0.06)	0.08(0.08)	0.11(0.11)	0.06(0.06)
Avg90		0.08(0.08)	0.07(0.07)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.07(0.07)	0.07(0.07)	0.14(0.14)	0.20(0.20)	0.24(0.24)	0.15(0.15)

TimeBehaviour_page_6: <a href="http://localhost/seleksibaru/c_login/proseslogin">http://localhost/seleksibaru/c_login/proseslogin</a>														
Min	0.11(0.11)	0.11(0.11)	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.09(0.09)
Max	0.14(0.14)	0.16(0.16)	0.17(0.17)	0.19(0.19)	0.26(0.26)	0.26(0.26)	0.26(0.26)	0.26(0.26)	0.26(0.26)	0.26(0.26)	0.26(0.26)	0.26(0.26)	0.26(0.26)	0.26(0.26)
Avg	0.13(0.13)	0.12(0.12)	0.13(0.13)	0.13(0.13)	0.13(0.13)	0.13(0.13)	0.13(0.13)	0.13(0.13)	0.13(0.13)	0.13(0.13)	0.13(0.13)	0.13(0.13)	0.13(0.13)	0.13(0.13)
Avg90	0.14(0.14)	0.15(0.15)	0.16(0.16)	0.16(0.16)	0.17(0.17)	0.17(0.17)	0.17(0.17)	0.17(0.17)	0.17(0.17)	0.17(0.17)	0.17(0.17)	0.17(0.17)	0.17(0.17)	0.17(0.17)
TimeBehaviour_page_7: <a href="http://localhost/seleksibaru/c_main/pengumuman">http://localhost/seleksibaru/c_main/pengumuman</a>														
Min	0.03(0.51)	0.05(0.53)	0.03(0.51)	0.03(0.51)	0.03(0.51)	0.03(0.51)	0.03(0.51)	0.03(0.51)	0.03(0.51)	0.03(0.51)	0.03(0.51)	0.03(0.51)	0.03(0.51)	0.03(0.51)
Max	0.06(0.61)	0.09(0.64)	0.08(0.69)	0.08(0.70)	0.11(1.00)	0.11(1.00)	0.11(1.00)	0.11(1.00)	0.11(1.00)	0.11(1.00)	0.11(1.00)	0.11(1.00)	0.11(1.00)	0.11(1.00)
Avg	0.05(0.56)	0.06(0.57)	0.05(0.57)	0.05(0.59)	0.05(0.60)	0.05(0.60)	0.05(0.60)	0.05(0.60)	0.05(0.60)	0.05(0.60)	0.05(0.60)	0.05(0.60)	0.05(0.60)	0.05(0.60)
Avg90	0.06(0.60)	0.08(0.61)	0.07(0.63)	0.07(0.64)	0.08(0.66)	0.08(0.66)	0.08(0.66)	0.08(0.66)	0.08(0.66)	0.08(0.66)	0.08(0.66)	0.08(0.66)	0.08(0.66)	0.08(0.66)
TimeBehaviour_page_8: <a href="http://localhost/seleksibaru/c_calonsiswa/lihatProfil">http://localhost/seleksibaru/c_calonsiswa/lihatProfil</a>														
Min	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)
Max	0.08(0.08)	0.08(0.08)	0.08(0.08)	0.08(0.08)	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.09(0.09)
Avg	0.06(0.06)	0.05(0.05)	0.05(0.05)	0.05(0.05)	0.05(0.05)	0.05(0.05)	0.05(0.05)	0.05(0.05)	0.05(0.05)	0.05(0.05)	0.05(0.05)	0.05(0.05)	0.05(0.05)	0.05(0.05)
Avg90	0.08(0.08)	0.08(0.08)	0.08(0.08)	0.08(0.08)	0.08(0.08)	0.08(0.08)	0.08(0.08)	0.08(0.08)	0.08(0.08)	0.08(0.08)	0.08(0.08)	0.08(0.08)	0.08(0.08)	0.08(0.08)
TimeBehaviour_page_9: <a href="http://localhost/seleksibaru/c_calonsiswa/lihatPembayaran">http://localhost/seleksibaru/c_calonsiswa/lihatPembayaran</a>														
Min	0.03(0.03)	0.05(0.05)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)
Max	0.08(0.08)	0.08(0.08)	0.08(0.08)	0.09(0.09)	0.11(0.11)	0.13(0.13)	0.13(0.13)	0.13(0.13)	0.13(0.13)	0.13(0.13)	0.13(0.13)	0.13(0.13)	0.13(0.13)	0.13(0.13)
Avg	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)
Avg90	0.08(0.08)	0.08(0.08)	0.07(0.07)	0.07(0.07)	0.08(0.08)	0.08(0.08)	0.08(0.08)	0.08(0.08)	0.08(0.08)	0.08(0.08)	0.08(0.08)	0.08(0.08)	0.08(0.08)	0.08(0.08)
TimeBehaviour_page_10: <a href="http://localhost/seleksibaru/c_calonsiswa/ujian">http://localhost/seleksibaru/c_calonsiswa/ujian</a>														
Min	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)
Max	0.05(0.05)	0.11(0.11)	0.06(0.06)	0.06(0.06)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.11(0.11)
Avg	0.04(0.04)	0.05(0.05)	0.04(0.04)	0.04(0.04)	0.04(0.04)	0.04(0.04)	0.04(0.04)	0.04(0.04)	0.04(0.04)	0.04(0.04)	0.04(0.04)	0.04(0.04)	0.04(0.04)	0.04(0.04)
Avg90	0.05(0.05)	0.07(0.07)	0.06(0.06)	0.06(0.06)	0.07(0.07)	0.07(0.07)	0.07(0.07)	0.07(0.07)	0.07(0.07)	0.07(0.07)	0.07(0.07)	0.07(0.07)	0.07(0.07)	0.07(0.07)



TimeBehaviour_page_14: http://localhost/seleksibaru/c_login	6	13	21	27	36	41	49	58	58	367
Min	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)
Max	0.06(0.06)	0.08(0.08)	0.08(0.08)	0.11(0.11)	0.08(0.08)	0.08(0.08)	0.09(0.09)	0.33(0.33)	0.31(0.31)	0.36(0.36)
Avg	0.05(0.05)	0.05(0.05)	0.05(0.05)	0.05(0.05)	0.04(0.04)	0.04(0.04)	0.04(0.04)	0.06(0.06)	0.06(0.06)	0.09(0.09)
Avg90	0.06(0.06)	0.07(0.07)	0.06(0.06)	0.07(0.07)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.12(0.12)	0.12(0.12)	0.18(0.18)
TimeBehaviour_page_15: http://localhost/seleksibaru/c_login/prosesLogin	6	13	20	28	35	42	47	58	58	365
Min	0.12(0.12)	0.09(0.09)	0.09(0.09)	0.11(0.11)	0.11(0.11)	0.09(0.09)	0.11(0.11)	0.11(0.11)	0.09(0.09)	0.11(0.11)
Max	0.16(0.16)	0.16(0.16)	0.19(0.19)	0.16(0.16)	0.19(0.19)	0.22(0.22)	0.25(0.25)	1.37(1.37)	1.37(1.37)	1.42(1.42)
Avg	0.14(0.14)	0.13(0.13)	0.13(0.13)	0.13(0.13)	0.13(0.13)	0.14(0.14)	0.14(0.14)	0.15(0.15)	0.23(0.23)	0.34(0.34)
Avg90	0.16(0.16)	0.15(0.15)	0.16(0.16)	0.14(0.14)	0.16(0.16)	0.17(0.17)	0.18(0.18)	0.19(0.19)	0.52(0.52)	0.70(0.70)
TimeBehaviour_page_16: http://localhost/seleksibaru/c_admin/getListAdmin	5	13	20	29	34	41	49	56	58	364
Min	0.06(0.06)	0.05(0.05)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)
Max	0.08(0.08)	0.08(0.08)	0.09(0.09)	0.11(0.11)	0.13(0.13)	0.11(0.11)	0.13(0.13)	0.28(0.28)	0.55(0.55)	0.61(0.61)
Avg	0.07(0.07)	0.06(0.06)	0.05(0.05)	0.05(0.05)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.09(0.09)	0.11(0.11)
Avg90	0.07(0.07)	0.08(0.08)	0.07(0.07)	0.08(0.08)	0.08(0.08)	0.08(0.08)	0.08(0.08)	0.10(0.10)	0.20(0.20)	0.25(0.25)
TimeBehaviour_page_17: http://localhost/seleksibaru/c_admin/getListStaffu	5	13	19	30	33	41	50	56	58	364
Min	0.06(0.06)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)
Max	0.08(0.08)	0.06(0.06)	0.08(0.08)	0.08(0.08)	0.11(0.11)	0.08(0.08)	0.09(0.09)	0.61(0.61)	0.70(0.70)	0.70(0.70)
Avg	0.07(0.07)	0.05(0.05)	0.05(0.05)	0.05(0.05)	0.05(0.05)	0.05(0.05)	0.05(0.05)	0.07(0.07)	0.09(0.09)	0.11(0.11)
Avg90	0.08(0.08)	0.06(0.06)	0.07(0.07)	0.07(0.07)	0.07(0.07)	0.07(0.07)	0.08(0.08)	0.17(0.17)	0.25(0.25)	0.24(0.24)
TimeBehaviour_page_18: http://localhost/seleksibaru/c_admin/getHasilUjian	4	14	19	30	33	41	49	56	58	362
Min	0.08(0.08)	0.05(0.05)	0.06(0.06)	0.05(0.05)	0.05(0.05)	0.06(0.06)	0.05(0.05)	0.05(0.05)	0.06(0.06)	0.06(0.06)
Max	0.11(0.11)	0.09(0.09)	0.12(0.12)	0.14(0.14)	0.14(0.14)	0.14(0.14)	0.17(0.17)	0.59(0.59)	1.03(1.03)	2.65(2.65)
Avg	0.09(0.09)	0.07(0.07)	0.08(0.08)	0.08(0.08)	0.08(0.08)	0.08(0.08)	0.09(0.09)	0.10(0.10)	0.14(0.14)	0.19(0.19)
Avg90	0.09(0.09)	0.07(0.07)	0.08(0.08)	0.08(0.08)	0.08(0.08)	0.08(0.08)	0.09(0.09)	0.10(0.10)	0.14(0.14)	0.19(0.19)

TimeBehaviour1.page_4: http://localhost/seleksibaru/c_admin/lihatPengumuman		6	13	19	25	35	42	51	52	62	72	377
Min	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)
Max	0.09(0.09)	0.09(0.09)	0.08(0.08)	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.08(0.08)	0.09(0.09)	0.11(0.11)	0.20(0.20)	0.09(0.09)	0.20(0.20)
Avg	0.06(0.06)	0.07(0.07)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.05(0.05)	0.05(0.05)	0.06(0.06)	0.06(0.06)	0.05(0.05)	0.06(0.06)
Avg90	0.09(0.09)	0.09(0.09)	0.08(0.08)	0.08(0.08)	0.08(0.08)	0.08(0.08)	0.07(0.07)	0.07(0.07)	0.08(0.08)	0.09(0.09)	0.07(0.07)	0.08(0.08)
TimeBehaviour1.page_5: http://localhost/seleksibaru/c_admin/lihatSoal		6	12	20	24	35	43	51	52	62	70	375
Min	0.05(0.05)	0.03(0.03)	0.05(0.05)	0.05(0.05)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.05(0.05)	0.03(0.03)
Max	0.08(0.08)	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.08(0.08)	0.08(0.08)	0.08(0.08)	0.08(0.08)	0.09(0.09)	0.22(0.22)	0.13(0.13)	0.22(0.22)
Avg	0.07(0.07)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)
Avg90	0.08(0.08)	0.09(0.09)	0.08(0.08)	0.08(0.08)	0.08(0.08)	0.08(0.08)	0.08(0.08)	0.08(0.08)	0.08(0.08)	0.09(0.09)	0.08(0.08)	0.08(0.08)
TimeBehaviour1.page_6: http://localhost/seleksibaru/c_admin/keolaujian		6	11	21	24	35	43	48	54	63	68	373
Min	0.05(0.05)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)
Max	0.08(0.08)	0.08(0.08)	0.08(0.08)	0.08(0.08)	0.08(0.08)	0.08(0.08)	0.09(0.09)	0.09(0.09)	0.08(0.08)	0.13(0.13)	0.08(0.08)	0.13(0.13)
Avg	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.05(0.05)	0.05(0.05)	0.05(0.05)	0.05(0.05)	0.06(0.06)	0.05(0.05)	0.05(0.05)	0.05(0.05)
Avg90	0.08(0.08)	0.08(0.08)	0.08(0.08)	0.08(0.08)	0.07(0.07)	0.07(0.07)	0.07(0.07)	0.07(0.07)	0.08(0.08)	0.07(0.07)	0.07(0.07)	0.07(0.07)
TimeBehaviour1.page_7: http://localhost/seleksibaru/c_login/logout		6	11	21	24	33	45	44	57	63	69	373
Min	0.05(0.05)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)
Max	0.08(0.08)	0.08(0.08)	0.09(0.09)	0.08(0.08)	0.08(0.08)	0.08(0.08)	0.08(0.08)	0.12(0.12)	0.09(0.09)	0.11(0.11)	0.08(0.08)	0.12(0.12)
Avg	0.06(0.06)	0.05(0.05)	0.05(0.05)	0.05(0.05)	0.05(0.05)	0.05(0.05)	0.04(0.04)	0.04(0.04)	0.05(0.05)	0.04(0.04)	0.04(0.04)	0.05(0.05)
Avg90	0.08(0.08)	0.07(0.07)	0.08(0.08)	0.06(0.06)	0.07(0.07)	0.07(0.07)	0.06(0.06)	0.07(0.07)	0.07(0.07)	0.06(0.06)	0.06(0.06)	0.07(0.07)
TimeBehaviour1.page_8: http://localhost/seleksibaru/c_main		6	11	21	24	33	45	44	57	63	69	373
Min	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)
Max	0.05(0.05)	0.05(0.05)	0.05(0.05)	0.05(0.05)	0.06(0.06)	0.06(0.06)	0.08(0.08)	0.08(0.08)	0.09(0.09)	0.22(0.22)	0.11(0.11)	0.22(0.22)
Avg	0.04(0.04)	0.04(0.04)	0.04(0.04)	0.04(0.04)	0.04(0.04)	0.04(0.04)	0.04(0.04)	0.04(0.04)	0.04(0.04)	0.05(0.05)	0.05(0.05)	0.04(0.04)
Avg90	0.05(0.05)	0.05(0.05)	0.05(0.05)	0.05(0.05)	0.05(0.05)	0.05(0.05)	0.06(0.06)	0.06(0.06)	0.07(0.07)	0.06(0.06)	0.08(0.08)	0.06(0.06)



TimeBehaviour2_page_1: http://localhost/seleksibaru/		2	4	4	5	9	7	12	6	14	13	76
Min	0.05(0.05)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)
Max	0.09(0.09)	0.06(0.06)	0.05(0.05)	0.05(0.05)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.05(0.05)	0.09(0.09)
Avg	0.07(0.07)	0.05(0.05)	0.04(0.04)	0.04(0.04)	0.04(0.04)	0.04(0.04)	0.04(0.04)	0.04(0.04)	0.05(0.05)	0.04(0.04)	0.04(0.04)	0.04(0.04)
Avg90	0.09(0.09)	0.06(0.06)	0.05(0.05)	0.05(0.05)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.05(0.05)	0.06(0.06)	0.06(0.06)	0.05(0.05)	0.06(0.06)
TimeBehaviour2_page_2: http://localhost/seleksibaru/c_admin/getListAdmin		2	3	5	5	9	7	12	6	14	11	74
Min	0.08(0.08)	0.09(0.09)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.08(0.08)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)
Max	0.09(0.09)	0.09(0.09)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.13(0.13)	0.12(0.12)	0.11(0.11)	0.09(0.09)	0.11(0.11)	0.13(0.13)	0.13(0.13)
Avg	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.08(0.08)	0.09(0.09)
Avg90	0.09(0.09)	0.09(0.09)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.12(0.12)	0.11(0.11)	0.09(0.09)	0.11(0.11)	0.11(0.11)	0.11(0.11)
TimeBehaviour2_page_4: http://localhost/seleksibaru/c_admin/getListAdmin		1	4	4	6	6	9	8	10	13	9	70
Min	0.09(0.09)	0.08(0.08)	0.06(0.06)	0.06(0.06)	0.08(0.08)	0.08(0.08)	0.06(0.06)	0.08(0.08)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)
Max	0.09(0.09)	0.11(0.11)	0.09(0.11)	0.09(0.11)	0.09(0.09)	0.11(0.11)	0.17(0.19)	0.12(0.12)	0.13(0.13)	0.13(0.13)	0.12(0.12)	0.17(0.19)
Avg	0.09(0.09)	0.09(0.10)	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.10(0.10)	0.10(0.10)	0.09(0.09)	0.09(0.09)	0.10(0.10)	0.09(0.10)
Avg90	0.09(0.09)	0.11(0.11)	0.09(0.11)	0.09(0.11)	0.09(0.09)	0.11(0.11)	0.14(0.15)	0.12(0.12)	0.11(0.11)	0.12(0.12)	0.12(0.12)	0.12(0.12)
TimeBehaviour2_page_5: http://localhost/seleksibaru/c_admin/hapusAdmin		1	4	4	5	7	7	8	12	12	9	69
Min	0.09(0.09)	0.06(0.06)	0.08(0.08)	0.06(0.06)	0.06(0.06)	0.08(0.08)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)
Max	0.09(0.09)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.09(0.09)	0.11(0.11)	0.09(0.09)	0.11(0.11)	0.14(0.14)	0.13(0.13)	0.13(0.13)	0.14(0.14)
Avg	0.09(0.09)	0.08(0.08)	0.10(0.10)	0.10(0.10)	0.07(0.07)	0.10(0.10)	0.08(0.08)	0.09(0.09)	0.09(0.09)	0.08(0.08)	0.09(0.09)	0.09(0.09)
Avg90	0.09(0.09)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.09(0.09)	0.11(0.11)	0.09(0.09)	0.11(0.11)	0.12(0.12)	0.11(0.11)	0.12(0.12)	0.11(0.11)
TimeBehaviour2_page_6: http://localhost/seleksibaru/c_admin/getListAdmin		1	4	4	5	7	7	8	12	12	9	69
Min	0.09(0.09)	0.06(0.06)	0.09(0.09)	0.06(0.06)	0.06(0.06)	0.08(0.08)	0.08(0.08)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)
Max	0.09(0.09)	0.11(0.11)	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.14(0.14)	0.11(0.11)	0.11(0.11)	0.13(0.13)	0.11(0.11)	0.14(0.14)
Avg	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.08(0.08)	0.09(0.09)	0.09(0.09)	0.08(0.08)	0.09(0.09)	0.08(0.08)	0.08(0.08)	0.09(0.09)
Avg90	0.09(0.09)	0.11(0.11)	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.12(0.12)	0.10(0.10)	0.11(0.11)	0.11(0.11)	0.10(0.10)	0.11(0.11)

TimeBehaviour2_page_7: http://localhost/seleksibaru/c_admin/getListStafftu													69
Min	0.08(0.08)	0.06(0.06)	0.08(0.08)	0.06(0.06)	0.08(0.08)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)
Max	0.08(0.08)	0.09(0.09)	0.16(0.16)	0.11(0.11)	0.11(0.11)	0.09(0.09)	0.13(0.13)	0.13(0.13)	0.14(0.14)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.16(0.16)
Avg	0.08(0.08)	0.08(0.08)	0.11(0.11)	0.08(0.08)	0.09(0.09)	0.08(0.08)	0.10(0.10)	0.08(0.08)	0.09(0.09)	0.08(0.08)	0.09(0.09)	0.09(0.09)	0.09(0.09)
Avg90	0.08(0.08)	0.09(0.09)	0.15(0.15)	0.11(0.11)	0.10(0.10)	0.09(0.09)	0.12(0.12)	0.09(0.09)	0.12(0.12)	0.10(0.10)	0.10(0.10)	0.11(0.11)	0.11(0.11)
TimeBehaviour2_page_9: http://localhost/seleksibaru/c_admin/getListStafftu													66
Min	0.11(0.11)	0.09(0.09)	0.08(0.08)	0.08(0.08)	0.06(0.06)	0.08(0.08)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.08(0.08)	0.06(0.06)	0.06(0.06)	0.06(0.06)
Max	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.17(0.17)	0.09(0.09)	0.11(0.11)	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.11(0.11)	0.17(0.17)
Avg	0.11(0.11)	0.10(0.10)	0.09(0.10)	0.09(0.10)	0.09(0.10)	0.10(0.10)	0.08(0.08)	0.09(0.09)	0.08(0.08)	0.08(0.08)	0.08(0.08)	0.08(0.08)	0.09(0.09)
Avg90	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.15(0.15)	0.09(0.09)	0.11(0.11)	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.10(0.10)	0.11(0.11)
TimeBehaviour2_page_10: http://localhost/seleksibaru/c_admin/hapusStafftu													66
Min	0.08(0.08)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.08(0.08)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)
Max	0.08(0.08)	0.09(0.09)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.33(0.33)	0.11(0.11)	0.33(0.33)	0.14(0.14)	0.09(0.09)	0.09(0.09)	0.11(0.11)	0.33(0.33)
Avg	0.08(0.08)	0.08(0.08)	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.12(0.12)	0.08(0.08)	0.12(0.12)	0.09(0.09)	0.08(0.08)	0.09(0.09)	0.09(0.09)	0.09(0.09)
Avg90	0.08(0.08)	0.09(0.09)	0.11(0.11)	0.10(0.10)	0.10(0.10)	0.25(0.25)	0.10(0.10)	0.10(0.10)	0.12(0.12)	0.09(0.09)	0.10(0.10)	0.10(0.10)	0.13(0.13)
TimeBehaviour2_page_11: http://localhost/seleksibaru/c_admin/getListStafftu													66
Min	0.11(0.11)	0.08(0.08)	0.06(0.06)	0.06(0.06)	0.09(0.09)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)
Max	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.13(0.13)	0.08(0.08)	0.11(0.11)	0.09(0.09)	0.11(0.11)	0.09(0.09)	0.11(0.11)	0.11(0.11)	0.13(0.13)
Avg	0.11(0.11)	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.10(0.10)	0.07(0.07)	0.08(0.08)	0.08(0.08)	0.09(0.09)	0.08(0.08)	0.09(0.09)	0.09(0.09)	0.08(0.08)
Avg90	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.12(0.12)	0.08(0.08)	0.10(0.10)	0.09(0.09)	0.11(0.11)	0.09(0.09)	0.10(0.10)	0.10(0.10)	0.10(0.10)
TimeBehaviour2_page_12: http://localhost/seleksibaru/c_admin/lihatPengumuman													66
Min	0.09(0.09)	0.08(0.08)	0.06(0.06)	0.08(0.08)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)
Max	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.11(0.11)	0.09(0.09)	0.13(0.13)	0.09(0.09)	0.11(0.11)	0.09(0.09)	0.11(0.11)	0.14(0.14)	0.14(0.14)
Avg	0.09(0.09)	0.09(0.09)	0.08(0.08)	0.09(0.09)	0.09(0.09)	0.08(0.08)	0.09(0.09)	0.08(0.08)	0.09(0.09)	0.08(0.08)	0.09(0.09)	0.09(0.09)	0.09(0.09)
Avg90	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.11(0.11)	0.12(0.12)	0.09(0.09)	0.10(0.10)	0.11(0.11)	0.09(0.09)	0.10(0.10)	0.10(0.10)	0.11(0.11)

TimeBehaviour2_page_13: <a href="http://localhost/seleksibaru/c_admin/tambahPengumuman">http://localhost/seleksibaru/c_admin/tambahPengumuman</a>	1	2	4	5	6	6	9	11	7	15	66
Min	0.11(0.11)	0.09(0.09)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)
Max	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.09(0.09)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.11(0.11)
Avg	0.11(0.11)	0.10(0.10)	0.10(0.10)	0.09(0.09)	0.09(0.09)	0.08(0.08)	0.08(0.08)	0.09(0.09)	0.08(0.08)	0.08(0.08)	0.09(0.09)
Avg90	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.09(0.09)	0.10(0.10)	0.11(0.11)	0.10(0.10)	0.10(0.10)	0.11(0.11)
TimeBehaviour2_page_14: <a href="http://localhost/seleksibaru/c_admin/lihatPengumuman">http://localhost/seleksibaru/c_admin/lihatPengumuman</a>	1	2	4	5	6	6	9	11	7	15	66
Min	0.09(0.09)	0.06(0.06)	0.09(0.09)	0.08(0.08)	0.08(0.08)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)
Max	0.09(0.09)	0.06(0.06)	0.11(0.11)	0.11(0.11)	0.13(0.13)	0.11(0.11)	0.13(0.13)	0.13(0.13)	0.11(0.11)	0.13(0.13)	0.13(0.13)
Avg	0.09(0.09)	0.06(0.06)	0.10(0.10)	0.10(0.10)	0.11(0.11)	0.08(0.08)	0.09(0.09)	0.08(0.08)	0.08(0.08)	0.09(0.09)	0.09(0.09)
Avg90	0.09(0.09)	0.06(0.06)	0.11(0.11)	0.11(0.11)	0.13(0.13)	0.11(0.11)	0.12(0.12)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.11(0.11)
TimeBehaviour2_page_15: <a href="http://localhost/seleksibaru/c_admin/hapusPengumuman">http://localhost/seleksibaru/c_admin/hapusPengumuman</a>	1	2	4	5	6	6	9	11	7	15	66
Min	0.09(0.09)	0.09(0.09)	0.08(0.08)	0.08(0.08)	0.06(0.06)	0.08(0.08)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)
Max	0.09(0.09)	0.11(0.11)	0.11(0.11)	0.09(0.09)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.16(0.16)	0.11(0.11)	0.11(0.11)	0.16(0.16)
Avg	0.09(0.09)	0.10(0.10)	0.09(0.09)	0.09(0.09)	0.10(0.10)	0.09(0.09)	0.08(0.08)	0.10(0.10)	0.08(0.08)	0.08(0.08)	0.09(0.09)
Avg90	0.09(0.09)	0.11(0.11)	0.11(0.11)	0.09(0.09)	0.11(0.11)	0.11(0.11)	0.10(0.10)	0.13(0.13)	0.10(0.10)	0.10(0.10)	0.11(0.11)
TimeBehaviour2_page_16: <a href="http://localhost/seleksibaru/c_admin/lihatPengumuman">http://localhost/seleksibaru/c_admin/lihatPengumuman</a>	1	2	4	5	6	6	10	11	7	15	66
Min	0.08(0.08)	0.09(0.09)	0.09(0.09)	0.06(0.06)	0.06(0.06)	0.08(0.08)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)
Max	0.08(0.08)	0.11(0.11)	0.11(0.11)	0.08(0.08)	0.11(0.11)	0.09(0.09)	0.13(0.13)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.13(0.13)
Avg	0.08(0.08)	0.10(0.10)	0.10(0.10)	0.07(0.07)	0.09(0.09)	0.09(0.09)	0.10(0.10)	0.08(0.08)	0.09(0.09)	0.09(0.09)	0.09(0.09)
Avg90	0.08(0.08)	0.11(0.11)	0.11(0.11)	0.08(0.08)	0.11(0.11)	0.09(0.09)	0.12(0.12)	0.10(0.10)	0.11(0.11)	0.10(0.10)	0.11(0.11)
TimeBehaviour2_page_17: <a href="http://localhost/seleksibaru/c_admin/lihatSoal">http://localhost/seleksibaru/c_admin/lihatSoal</a>	1	2	4	5	6	6	10	11	7	15	66
Min	0.08(0.08)	0.08(0.08)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)
Max	0.08(0.08)	0.09(0.09)	0.08(0.08)	0.08(0.08)	0.08(0.08)	0.08(0.08)	0.08(0.08)	0.09(0.09)	0.08(0.08)	0.08(0.08)	0.09(0.09)
Avg	0.08(0.08)	0.09(0.09)	0.07(0.07)	0.07(0.07)	0.07(0.07)	0.07(0.07)	0.07(0.07)	0.07(0.07)	0.07(0.07)	0.07(0.07)	0.07(0.07)
Avg90	0.08(0.08)	0.09(0.09)	0.08(0.08)	0.08(0.08)	0.08(0.08)	0.08(0.08)	0.07(0.07)	0.07(0.07)	0.07(0.07)	0.07(0.07)	0.07(0.07)

TimeBehaviour2_page_18: <a href="http://localhost/seleksibaru/c_admin/tambahSoal">http://localhost/seleksibaru/c_admin/tambahSoal</a>												
Min	-	0.08(0.08)	0.08(0.08)	0.08(0.08)	0.08(0.08)	0.08(0.08)	0.08(0.08)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)
Max	-	0.09(0.09)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.09(0.09)	0.11(0.11)
Avg	-	0.08(0.08)	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.08(0.08)	0.08(0.08)	0.08(0.08)	0.08(0.08)	0.09(0.09)
Avg90	-	0.09(0.09)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.10(0.10)	0.10(0.10)	0.11(0.11)	0.09(0.09)	0.11(0.11)
TimeBehaviour2_page_19: <a href="http://localhost/seleksibaru/c_admin/lihatSoal">http://localhost/seleksibaru/c_admin/lihatSoal</a>												
Min	-	0.08(0.08)	0.06(0.06)	0.08(0.08)	0.08(0.08)	0.06(0.06)	0.06(0.06)	0.08(0.08)	0.08(0.08)	0.06(0.06)	0.06(0.06)	0.06(0.06)
Max	-	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.09(0.09)	0.09(0.09)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.11(0.11)
Avg	-	0.09(0.09)	0.09(0.09)	0.10(0.10)	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.08(0.08)	0.08(0.08)	0.09(0.09)
Avg90	-	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.09(0.09)	0.09(0.09)	0.11(0.11)	0.11(0.11)	0.10(0.10)	0.10(0.10)	0.11(0.11)
TimeBehaviour2_page_20: <a href="http://localhost/seleksibaru/c_admin/hapusSoal">http://localhost/seleksibaru/c_admin/hapusSoal</a>												
Min	-	0.08(0.08)	0.08(0.08)	0.08(0.08)	0.09(0.09)	0.08(0.08)	0.08(0.08)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)
Max	-	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.12(0.12)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.13(0.13)	0.11(0.11)	0.11(0.11)	0.13(0.13)
Avg	-	0.09(0.09)	0.09(0.09)	0.10(0.10)	0.10(0.10)	0.10(0.10)	0.10(0.10)	0.09(0.09)	0.09(0.09)	0.08(0.08)	0.08(0.08)	0.09(0.09)
Avg90	-	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.12(0.12)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.10(0.10)	0.11(0.11)
TimeBehaviour2_page_21: <a href="http://localhost/seleksibaru/c_admin/lihatSoal">http://localhost/seleksibaru/c_admin/lihatSoal</a>												
Min	-	0.11(0.11)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)
Max	-	0.11(0.11)	0.09(0.09)	0.11(0.11)	0.09(0.09)	0.14(0.14)	0.14(0.14)	0.11(0.11)	0.11(0.11)	0.12(0.12)	0.13(0.13)	0.14(0.14)
Avg	-	0.11(0.11)	0.08(0.08)	0.09(0.09)	0.08(0.08)	0.08(0.08)	0.08(0.08)	0.08(0.08)	0.08(0.08)	0.08(0.08)	0.08(0.08)	0.08(0.08)
Avg90	-	0.11(0.11)	0.09(0.09)	0.11(0.11)	0.09(0.09)	0.12(0.12)	0.12(0.12)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.10(0.10)	0.11(0.11)
TimeBehaviour2_page_22: <a href="http://localhost/seleksibaru/c_admin/keblajjian">http://localhost/seleksibaru/c_admin/keblajjian</a>												
Min	-	0.09(0.09)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)
Max	-	0.12(0.12)	0.09(0.09)	0.11(0.11)	0.11(0.11)	0.09(0.09)	0.09(0.09)	0.11(0.11)	0.11(0.11)	0.13(0.13)	0.11(0.11)	0.13(0.13)
Avg	-	0.10(0.10)	0.08(0.08)	0.09(0.09)	0.09(0.09)	0.08(0.08)	0.08(0.08)	0.10(0.10)	0.10(0.10)	0.09(0.09)	0.08(0.08)	0.09(0.09)
Avg90	-	0.12(0.12)	0.09(0.09)	0.11(0.11)	0.11(0.11)	0.09(0.09)	0.09(0.09)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.10(0.10)	0.11(0.11)

TimeBehaviour2, page_23: <a href="http://localhost/seleksibaru/c_admin/tentukanKuota">http://localhost/seleksibaru/c_admin/tentukanKuota</a>												
Min	-	0.08(0.08)	0.06(0.06)	0.06(0.06)	0.08(0.08)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)
Max	-	0.09(0.09)	0.09(0.09)	0.11(0.11)	0.13(0.13)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.13(0.13)	0.11(0.11)	0.13(0.13)
Avg	-	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.10(0.10)	0.09(0.09)	0.08(0.08)	0.08(0.08)	0.09(0.09)	0.09(0.09)	0.08(0.08)	0.09(0.09)
Avg90	-	0.09(0.09)	0.09(0.09)	0.11(0.11)	0.12(0.12)	0.11(0.11)	0.11(0.11)	0.10(0.10)	0.11(0.11)	0.10(0.10)	0.10(0.10)	0.11(0.11)
TimeBehaviour2, page_24: <a href="http://localhost/seleksibaru/c_admin/kelelaUjian">http://localhost/seleksibaru/c_admin/kelelaUjian</a>												
Min	-	0.06(0.06)	0.06(0.06)	0.09(0.09)	0.06(0.06)	0.08(0.08)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)
Max	-	0.09(0.09)	0.09(0.09)	0.11(0.11)	0.11(0.11)	0.16(0.16)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.16(0.16)
Avg	-	0.08(0.08)	0.08(0.08)	0.10(0.10)	0.09(0.09)	0.10(0.10)	0.08(0.08)	0.08(0.08)	0.09(0.09)	0.08(0.08)	0.09(0.09)	0.09(0.09)
Avg90	-	0.09(0.09)	0.09(0.09)	0.11(0.11)	0.11(0.11)	0.14(0.14)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.10(0.10)	0.11(0.11)	0.11(0.11)
TimeBehaviour2, page_25: <a href="http://localhost/seleksibaru/c_admin/tentukanSiswaDiterima">http://localhost/seleksibaru/c_admin/tentukanSiswaDiterima</a>												
Min	-	0.09(0.09)	0.09(0.09)	0.08(0.08)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)
Max	-	0.13(0.13)	0.09(0.09)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.08(0.08)	0.09(0.09)	0.09(0.09)	0.11(0.11)	0.13(0.13)
Avg	-	0.11(0.11)	0.09(0.09)	0.10(0.10)	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.07(0.07)	0.08(0.08)	0.08(0.08)	0.08(0.08)	0.08(0.08)
Avg90	-	0.13(0.13)	0.09(0.09)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.08(0.08)	0.09(0.09)	0.09(0.09)	0.10(0.10)	0.10(0.10)
TimeBehaviour2, page_26: <a href="http://localhost/seleksibaru/c_admin/kelelaUjian">http://localhost/seleksibaru/c_admin/kelelaUjian</a>												
Min	-	0.08(0.08)	0.08(0.08)	0.06(0.06)	0.06(0.06)	0.08(0.08)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)
Max	-	0.11(0.11)	0.11(0.11)	0.13(0.13)	0.09(0.09)	0.09(0.09)	0.13(0.13)	0.13(0.13)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.13(0.13)
Avg	-	0.09(0.09)	0.09(0.09)	0.10(0.10)	0.08(0.08)	0.08(0.08)	0.09(0.09)	0.07(0.07)	0.08(0.08)	0.08(0.08)	0.09(0.09)	0.09(0.09)
Avg90	-	0.11(0.11)	0.11(0.11)	0.13(0.13)	0.09(0.09)	0.09(0.09)	0.11(0.11)	0.08(0.08)	0.11(0.11)	0.10(0.10)	0.10(0.10)	0.11(0.11)
TimeBehaviour2, page_27: <a href="http://localhost/seleksibaru/c_admin/tentukanJadwal">http://localhost/seleksibaru/c_admin/tentukanJadwal</a>												
Min	-	0.09(0.09)	0.09(0.09)	0.06(0.06)	0.08(0.08)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)
Max	-	0.13(0.13)	0.13(0.13)	0.09(0.09)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.08(0.08)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.13(0.13)
Avg	-	0.11(0.11)	0.10(0.10)	0.08(0.08)	0.09(0.09)	0.09(0.09)	0.08(0.08)	0.08(0.08)	0.09(0.09)	0.08(0.08)	0.08(0.08)	0.09(0.09)
Avg90	-	0.13(0.13)	0.13(0.13)	0.09(0.09)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.08(0.08)	0.11(0.11)	0.10(0.10)	0.11(0.11)	0.11(0.11)

TimeBehaviour2.page_28: <a href="http://localhost/seleksibar/c_admin/kelolaUjian">http://localhost/seleksibar/c_admin/kelolaUjian</a>												
Min	0	0.09(0.09)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)
Max	-	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.09(0.09)	0.11(0.11)	0.09(0.09)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.11(0.11)
Avg	-	0.10(0.10)	0.08(0.08)	0.08(0.08)	0.09(0.09)	0.09(0.09)	0.08(0.08)	0.09(0.09)	0.08(0.08)	0.09(0.09)	0.09(0.09)	0.08(0.08)
Avg90	-	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.09(0.09)	0.11(0.11)	0.09(0.09)	0.11(0.11)	0.11(0.11)	0.10(0.10)	0.10(0.10)
TimeBehaviour2.page_30: <a href="http://localhost/seleksibar/c_admin/tentukanJadwal">http://localhost/seleksibar/c_admin/tentukanJadwal</a>												
Min	0	0.09(0.09)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)
Max	-	0.11(0.11)	0.09(0.09)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.16(0.16)	0.11(0.11)	0.14(0.14)
Avg	-	0.10(0.10)	0.08(0.08)	0.09(0.09)	0.08(0.08)	0.09(0.09)	0.08(0.08)	0.09(0.09)	0.10(0.10)	0.08(0.08)	0.09(0.09)	0.09(0.09)
Avg90	-	0.11(0.11)	0.09(0.09)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.13(0.13)	0.13(0.13)	0.11(0.11)	0.11(0.11)	0.11(0.11)
TimeBehaviour2.page_31: <a href="http://localhost/seleksibar/c_admin/kelolaUjian">http://localhost/seleksibar/c_admin/kelolaUjian</a>												
Min	0	0.06(0.06)	0.06(0.06)	0.08(0.08)	0.08(0.08)	0.09(0.09)	0.08(0.08)	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.06(0.06)	0.06(0.06)
Max	-	0.08(0.08)	0.09(0.09)	0.11(0.11)	0.09(0.09)	0.11(0.11)	0.11(0.11)	0.09(0.09)	0.11(0.11)	0.13(0.13)	0.09(0.09)	0.11(0.11)
Avg	-	0.07(0.07)	0.08(0.08)	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.10(0.10)	0.09(0.09)	0.08(0.08)	0.09(0.09)
Avg90	-	0.08(0.08)	0.09(0.09)	0.11(0.11)	0.11(0.11)	0.09(0.09)	0.11(0.11)	0.10(0.10)	0.11(0.11)	0.10(0.10)	0.09(0.09)	0.10(0.10)
TimeBehaviour2.page_32: <a href="http://localhost/seleksibar/c_admin/kelolaUjian">http://localhost/seleksibar/c_admin/kelolaUjian</a>												
Min	0	0.08(0.08)	0.09(0.09)	0.09(0.09)	0.08(0.08)	0.06(0.06)	0.08(0.08)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)
Max	-	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.09(0.09)	0.12(0.12)	0.11(0.11)
Avg	-	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.10(0.10)	0.09(0.09)	0.08(0.08)	0.09(0.09)
Avg90	-	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.09(0.09)	0.10(0.10)	0.11(0.11)
TimeBehaviour2.page_32: <a href="http://localhost/seleksibar/c_admin/kelolaUjian">http://localhost/seleksibar/c_admin/kelolaUjian</a>												
Min	0	0.09(0.09)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)
Max	-	0.11(0.11)	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.14(0.14)	0.09(0.09)	0.13(0.13)	0.09(0.09)
Avg	-	0.10(0.10)	0.09(0.09)	0.08(0.08)	0.07(0.07)	0.09(0.09)	0.08(0.08)	0.07(0.07)	0.09(0.09)	0.09(0.09)	0.08(0.08)	0.08(0.08)
Avg90	-	0.11(0.11)	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.11(0.11)	0.12(0.12)	0.12(0.12)	0.12(0.12)	0.11(0.11)	0.11(0.11)	0.11(0.11)

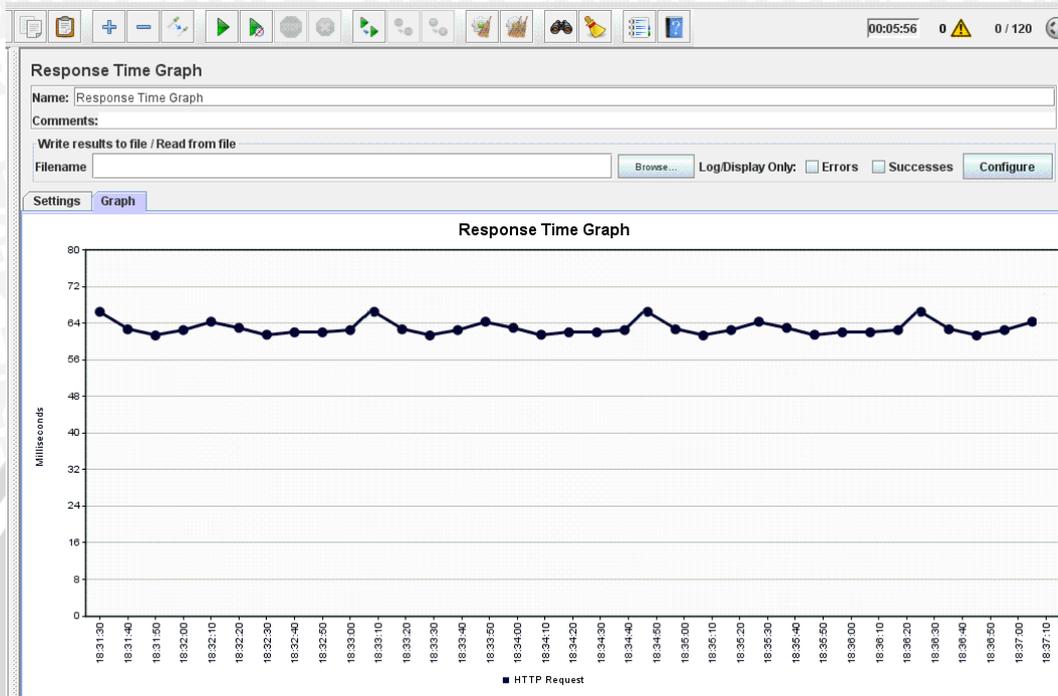
TimeBehaviour2_page_33: http://localhost/seleksibaru/c_admin/tentukanJadwal												
Min	0	2	4	3	5	6	11	5	13	13	62	
Max	-	0.09(0.09)	0.08(0.08)	0.08(0.08)	0.06(0.06)	0.08(0.08)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	
Avg	-	0.14(0.14)	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.14(0.14)	
Avg90	-	0.12(0.12)	0.09(0.09)	0.09(0.09)	0.07(0.07)	0.09(0.09)	0.08(0.08)	0.09(0.09)	0.08(0.08)	0.08(0.08)	0.08(0.08)	
TimeBehaviour2_page_34: http://localhost/seleksibaru/c_admin/kelolaJalan												
Min	0	2	4	3	5	6	11	5	13	13	62	
Max	-	0.08(0.08)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.05(0.05)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.05(0.05)	
Avg	-	0.08(0.08)	0.06(0.06)	0.08(0.08)	0.08(0.08)	0.08(0.08)	0.08(0.08)	0.08(0.08)	0.09(0.09)	0.11(0.11)	0.11(0.11)	
Avg90	-	0.08(0.08)	0.06(0.06)	0.07(0.07)	0.07(0.07)	0.06(0.06)	0.06(0.06)	0.07(0.07)	0.07(0.07)	0.07(0.07)	0.07(0.07)	
TimeBehaviour2_page_35: http://localhost/seleksibaru/c_login/logout												
Min	0	2	4	3	5	6	11	5	13	12	61	
Max	-	0.06(0.06)	0.03(0.03)	0.03(0.03)	0.05(0.05)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	
Avg	-	0.09(0.09)	0.06(0.06)	0.06(0.06)	0.08(0.08)	0.06(0.06)	0.11(0.11)	0.08(0.08)	0.06(0.06)	0.06(0.06)	0.11(0.11)	
Avg90	-	0.08(0.08)	0.05(0.05)	0.05(0.05)	0.07(0.07)	0.05(0.05)	0.06(0.06)	0.09(0.09)	0.04(0.04)	0.05(0.05)	0.05(0.05)	
TimeBehaviour2_page_36: http://localhost/seleksibaru/c_main												
Min	0	2	4	3	5	6	11	5	13	12	61	
Max	-	0.05(0.05)	0.03(0.03)	0.05(0.05)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.05(0.05)	0.03(0.03)	0.03(0.03)	
Avg	-	0.05(0.05)	0.05(0.05)	0.08(0.08)	0.05(0.05)	0.08(0.08)	0.06(0.06)	0.05(0.05)	0.06(0.06)	0.08(0.08)	0.08(0.08)	
Avg90	-	0.05(0.05)	0.04(0.04)	0.06(0.06)	0.04(0.04)	0.05(0.05)	0.04(0.04)	0.04(0.04)	0.05(0.05)	0.04(0.04)	0.04(0.04)	
TimeBehaviour2_page_37: http://localhost/seleksibaru/c_login												
Min	0	2	3	4	5	6	11	5	13	12	61	
Max	-	0.05(0.05)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	
Avg	-	0.08(0.08)	0.08(0.08)	0.06(0.06)	0.08(0.08)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.08(0.08)	0.08(0.08)	
Avg90	-	0.06(0.06)	0.05(0.05)	0.05(0.05)	0.06(0.06)	0.05(0.05)	0.05(0.05)	0.05(0.05)	0.05(0.05)	0.05(0.05)	0.05(0.05)	
TimeBehaviour2_page_37: http://localhost/seleksibaru/c_login												
Min	0	2	3	4	5	6	11	5	13	12	61	
Max	-	0.08(0.08)	0.08(0.08)	0.06(0.06)	0.08(0.08)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.08(0.08)	0.08(0.08)	
Avg	-	0.06(0.06)	0.05(0.05)	0.05(0.05)	0.06(0.06)	0.05(0.05)	0.05(0.05)	0.05(0.05)	0.05(0.05)	0.05(0.05)	0.05(0.05)	
Avg90	-	0.08(0.08)	0.08(0.08)	0.06(0.06)	0.08(0.08)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.07(0.07)	0.07(0.07)	

TimeBehaviour2_page_38: http://localhost/seleksibaru/c_login/prosesLogin	0	2	2	5	5	6	10	6	12	13	61
Min	-	0.13(0.13)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.11(0.11)	0.09(0.09)
Max	-	0.14(0.14)	0.11(0.11)	0.19(0.19)	0.17(0.17)	0.16(0.16)	0.16(0.16)	0.16(0.16)	0.19(0.19)	0.16(0.16)	0.19(0.19)
Avg	-	0.13(0.13)	0.11(0.11)	0.14(0.14)	0.13(0.13)	0.13(0.13)	0.13(0.13)	0.13(0.13)	0.14(0.14)	0.13(0.13)	0.13(0.13)
Avg90	-	0.14(0.14)	0.11(0.11)	0.18(0.18)	0.16(0.16)	0.15(0.15)	0.15(0.15)	0.15(0.15)	0.17(0.17)	0.16(0.16)	0.16(0.16)
TimeBehaviour2_page_39: http://localhost/seleksibaru/c_stafftu/lihatPembayaran	0	2	2	4	6	5	11	4	14	12	60
Min	-	0.08(0.08)	0.06(0.06)	0.05(0.05)	0.06(0.06)	0.05(0.05)	0.05(0.05)	0.05(0.05)	0.05(0.05)	0.05(0.05)	0.05(0.05)
Max	-	0.08(0.08)	0.08(0.08)	0.09(0.09)	0.08(0.08)	0.09(0.09)	0.08(0.08)	0.08(0.08)	0.09(0.09)	0.08(0.08)	0.09(0.09)
Avg	-	0.08(0.08)	0.07(0.07)	0.07(0.07)	0.08(0.08)	0.07(0.07)	0.06(0.06)	0.06(0.06)	0.07(0.07)	0.07(0.07)	0.07(0.07)
Avg90	-	0.08(0.08)	0.08(0.08)	0.09(0.09)	0.08(0.08)	0.09(0.09)	0.08(0.08)	0.08(0.08)	0.09(0.09)	0.08(0.08)	0.09(0.09)
TimeBehaviour2_page_40: http://localhost/seleksibaru/assets/	0	2	2	4	6	5	11	4	14	12	60
Min	-	0.17(0.17)	0.20(0.20)	0.17(0.17)	0.17(0.17)	0.17(0.17)	0.17(0.17)	0.17(0.17)	0.17(0.17)	0.17(0.17)	0.17(0.17)
Max	-	0.20(0.20)	0.20(0.20)	0.17(0.17)	0.19(0.19)	0.20(0.20)	0.22(0.22)	0.20(0.20)	0.31(0.31)	0.22(0.22)	0.31(0.31)
Avg	-	0.19(0.19)	0.20(0.20)	0.17(0.17)	0.18(0.18)	0.19(0.19)	0.19(0.19)	0.19(0.19)	0.22(0.22)	0.19(0.19)	0.19(0.19)
Avg90	-	0.20(0.20)	0.20(0.20)	0.17(0.17)	0.19(0.19)	0.20(0.20)	0.21(0.21)	0.20(0.20)	0.27(0.27)	0.21(0.21)	0.23(0.23)
TimeBehaviour2_page_41: http://localhost/seleksibaru/c_stafftu/ubahStatusPembayaran	0	2	2	4	6	5	10	5	13	10	57
Min	-	0.08(0.08)	0.11(0.11)	0.03(0.03)	0.03(0.03)	0.05(0.05)	0.05(0.05)	0.05(0.05)	0.03(0.03)	0.03(0.03)	0.03(0.03)
Max	-	0.11(0.11)	0.16(0.16)	0.09(0.09)	0.16(0.16)	0.13(0.13)	0.12(0.12)	0.06(0.06)	0.11(0.11)	0.06(0.06)	0.16(0.16)
Avg	-	0.09(0.09)	0.13(0.13)	0.07(0.07)	0.08(0.08)	0.08(0.08)	0.07(0.07)	0.06(0.06)	0.06(0.06)	0.05(0.05)	0.07(0.07)
Avg90	-	0.11(0.11)	0.16(0.16)	0.09(0.09)	0.13(0.13)	0.13(0.13)	0.10(0.10)	0.06(0.06)	0.09(0.09)	0.06(0.06)	0.10(0.10)
TimeBehaviour2_page_42: http://localhost/seleksibaru/c_stafftu/lihatPembayaran	0	2	2	4	6	5	10	5	13	10	57
Min	-	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.05(0.05)	0.05(0.05)	0.05(0.05)	0.05(0.05)	0.05(0.05)	0.05(0.05)	0.05(0.05)
Max	-	0.06(0.06)	0.08(0.08)	0.09(0.09)	0.08(0.08)	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.08(0.08)	0.09(0.09)
Avg	-	0.06(0.06)	0.07(0.07)	0.08(0.08)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.07(0.07)	0.07(0.07)	0.07(0.07)
Avg90	-	0.06(0.06)	0.08(0.08)	0.09(0.09)	0.07(0.07)	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.08(0.08)	0.09(0.09)

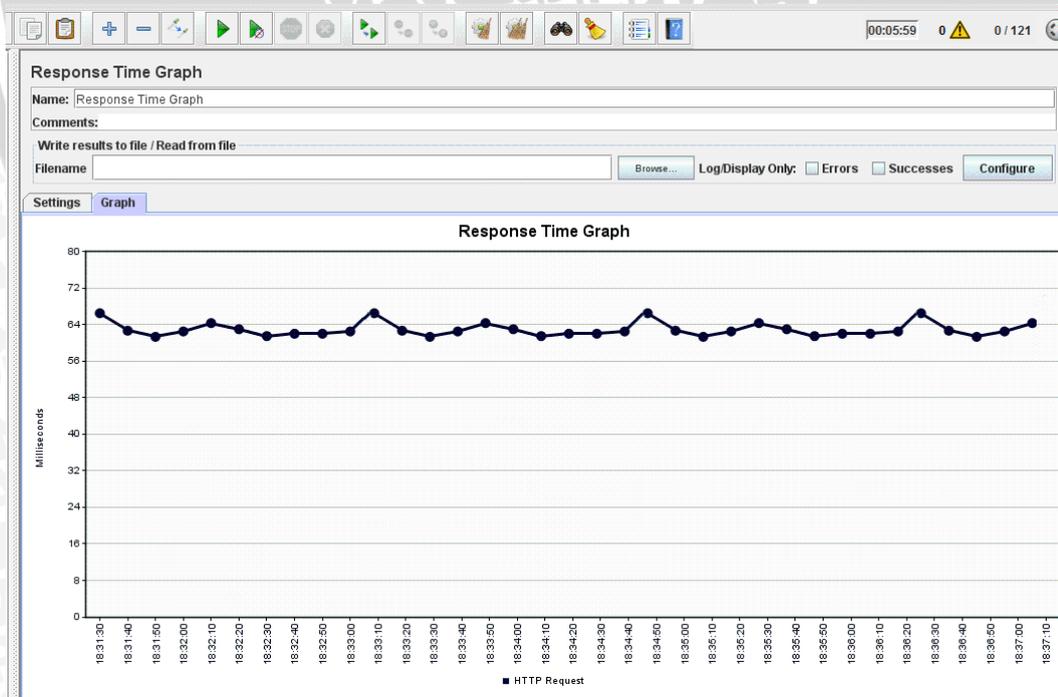
Avg	-	0.09(0.09)	0.13(0.13)	0.07(0.07)	0.08(0.08)	0.07(0.07)	0.06(0.06)	0.05(0.05)	0.07(0.07)			
Avg90	-	0.11(0.11)	0.16(0.16)	0.09(0.09)	0.13(0.13)	0.10(0.10)	0.09(0.09)	0.06(0.06)	0.10(0.10)			
TimeBehaviour2_page_42: <a href="http://localhost/seleksibaru/c_staffu/lihatPembayaran">http://localhost/seleksibaru/c_staffu/lihatPembayaran</a>	0	2	2	4	6	5	5	10	13	10	57	
Min	-	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.05(0.05)	0.05(0.05)	0.05(0.05)	0.05(0.05)	0.05(0.05)	0.05(0.05)	0.05(0.05)	
Max	-	0.06(0.06)	0.08(0.08)	0.09(0.09)	0.08(0.08)	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.08(0.08)	0.09(0.09)	
Avg	-	0.06(0.06)	0.07(0.07)	0.08(0.08)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.07(0.07)	0.07(0.07)	0.07(0.07)	0.07(0.07)	
Avg90	-	0.06(0.06)	0.08(0.08)	0.09(0.09)	0.07(0.07)	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.08(0.08)	0.09(0.09)	
TimeBehaviour2_page_43: <a href="http://localhost/seleksibaru/assets/">http://localhost/seleksibaru/assets/</a>	0	2	2	4	6	5	5	10	5	13	10	57
Min	-	0.17(0.17)	0.17(0.17)	0.17(0.17)	0.17(0.17)	0.17(0.17)	0.19(0.19)	0.17(0.17)	0.17(0.17)	0.17(0.17)	0.17(0.17)	
Max	-	0.17(0.17)	0.19(0.19)	0.19(0.19)	0.23(0.23)	0.20(0.20)	0.22(0.22)	0.22(0.22)	0.31(0.31)	0.23(0.23)	0.31(0.31)	
Avg	-	0.17(0.17)	0.18(0.18)	0.18(0.18)	0.21(0.21)	0.18(0.18)	0.20(0.20)	0.19(0.19)	0.20(0.20)	0.19(0.19)	0.19(0.19)	
Avg90	-	0.17(0.17)	0.19(0.19)	0.19(0.19)	0.23(0.23)	0.20(0.20)	0.22(0.22)	0.25(0.25)	0.23(0.23)	0.22(0.22)		
TimeBehaviour2_page_44: <a href="http://localhost/seleksibaru/c_staffu/ubahStatusPembayaran">http://localhost/seleksibaru/c_staffu/ubahStatusPembayaran</a>	0	2	2	3	7	5	4	10	4	13	10	56
Min	-	0.09(0.09)	0.11(0.11)	0.06(0.06)	0.03(0.03)	0.03(0.03)	0.05(0.05)	0.03(0.03)	0.03(0.03)	0.03(0.03)	0.03(0.03)	
Max	-	0.11(0.11)	0.13(0.13)	0.08(0.08)	0.09(0.09)	0.11(0.11)	0.08(0.08)	0.09(0.09)	0.08(0.08)	0.08(0.08)	0.20(0.20)	
Avg	-	0.10(0.10)	0.12(0.12)	0.07(0.07)	0.07(0.07)	0.07(0.07)	0.06(0.06)	0.06(0.06)	0.06(0.06)	0.05(0.05)	0.07(0.07)	
Avg90	-	0.11(0.11)	0.13(0.13)	0.08(0.08)	0.09(0.09)	0.10(0.10)	0.08(0.08)	0.09(0.09)	0.08(0.08)	0.08(0.08)	0.11(0.11)	
TimeBehaviour2_page_45: <a href="http://localhost/seleksibaru/c_staffu/lihatPembayaran">http://localhost/seleksibaru/c_staffu/lihatPembayaran</a>	0	2	2	3	7	5	4	10	4	12	11	56
Min	-	0.05(0.05)	0.06(0.06)	0.06(0.06)	0.05(0.05)	0.05(0.05)	0.08(0.08)	0.05(0.05)	0.05(0.05)	0.05(0.05)	0.05(0.05)	
Max	-	0.05(0.05)	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.09(0.09)	
Avg	-	0.05(0.05)	0.08(0.08)	0.08(0.08)	0.07(0.07)	0.06(0.06)	0.08(0.08)	0.07(0.07)	0.07(0.07)	0.07(0.07)	0.07(0.07)	
Avg90	-	0.05(0.05)	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.08(0.08)	0.09(0.09)	0.09(0.09)	0.09(0.09)	0.08(0.08)	0.09(0.09)	
TimeBehaviour2_page_46: <a href="http://localhost/seleksibaru/assets/">http://localhost/seleksibaru/assets/</a>	0	2	2	3	7	5	4	10	4	12	11	56
Min	-	0.17(0.17)	0.19(0.19)	0.17(0.17)	0.17(0.17)	0.17(0.17)	0.17(0.17)	0.17(0.17)	0.17(0.17)	0.17(0.17)	0.17(0.17)	
Max	-	0.17(0.17)	0.20(0.20)	0.19(0.19)	0.22(0.22)	0.19(0.19)	0.19(0.19)	0.22(0.22)	0.25(0.25)	0.20(0.20)	0.25(0.25)	
Avg	-	0.17(0.17)	0.19(0.19)	0.18(0.18)	0.18(0.18)	0.19(0.19)	0.18(0.18)	0.19(0.19)	0.20(0.20)	0.18(0.18)	0.19(0.19)	
Avg90	-	0.17(0.17)	0.20(0.20)	0.19(0.19)	0.21(0.21)	0.19(0.19)	0.19(0.19)	0.24(0.24)	0.20(0.20)	0.20(0.20)	0.21(0.21)	

## LAMPIRAN C HASIL PENGUJIAN CAPACITY

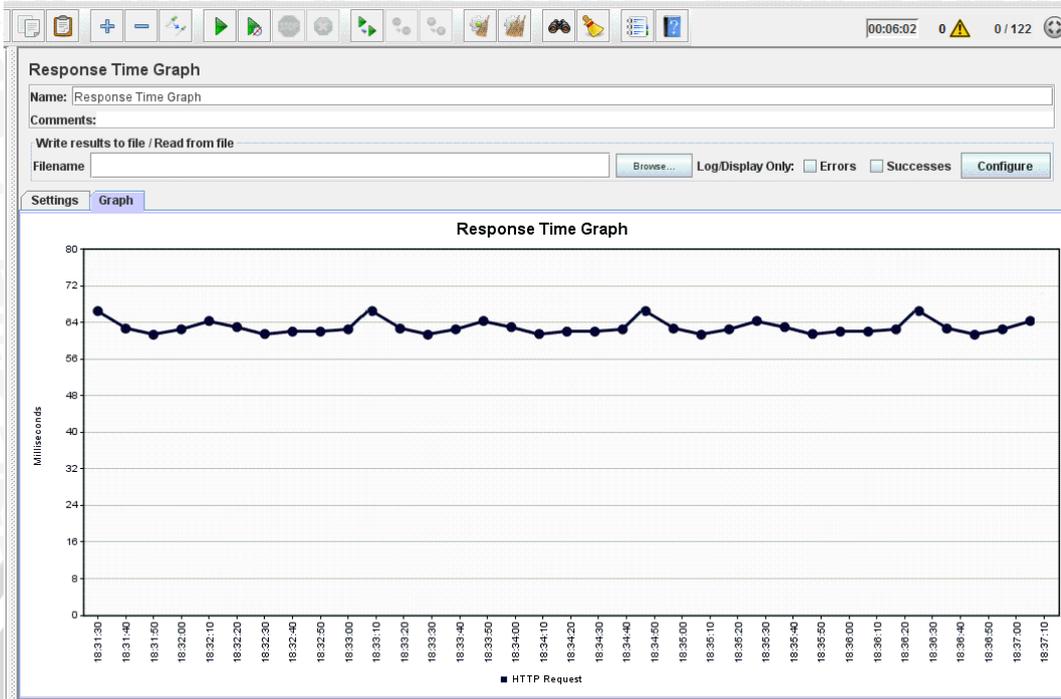
### C.1 Hasil Pengujian Capacity dengan 120 User



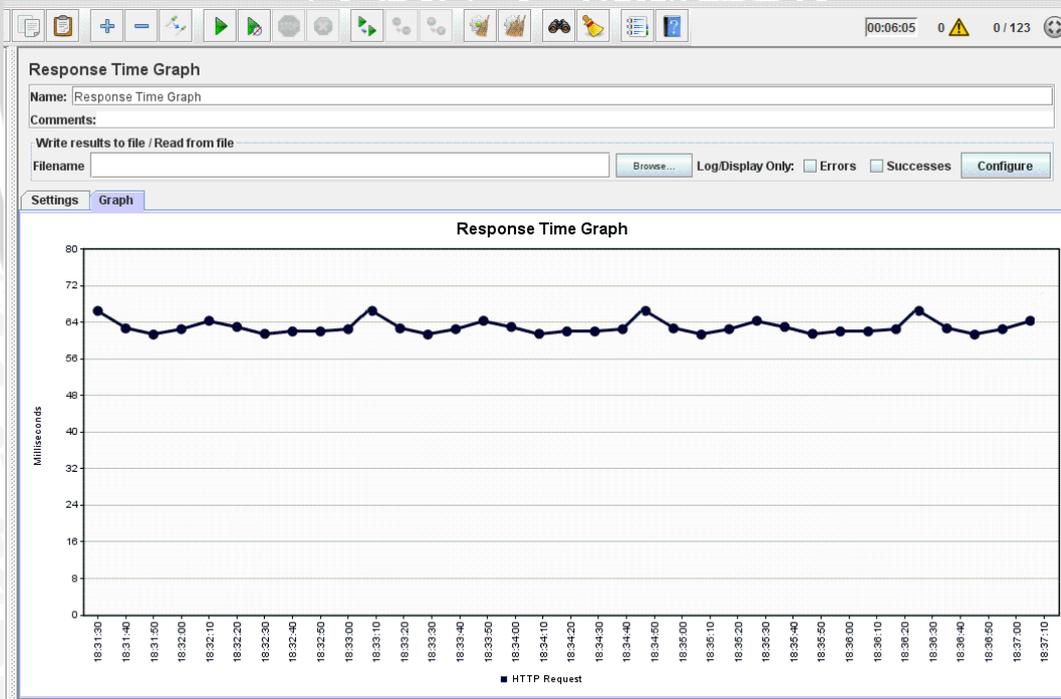
### C.2 Hasil Pengujian Capacity dengan 121 User



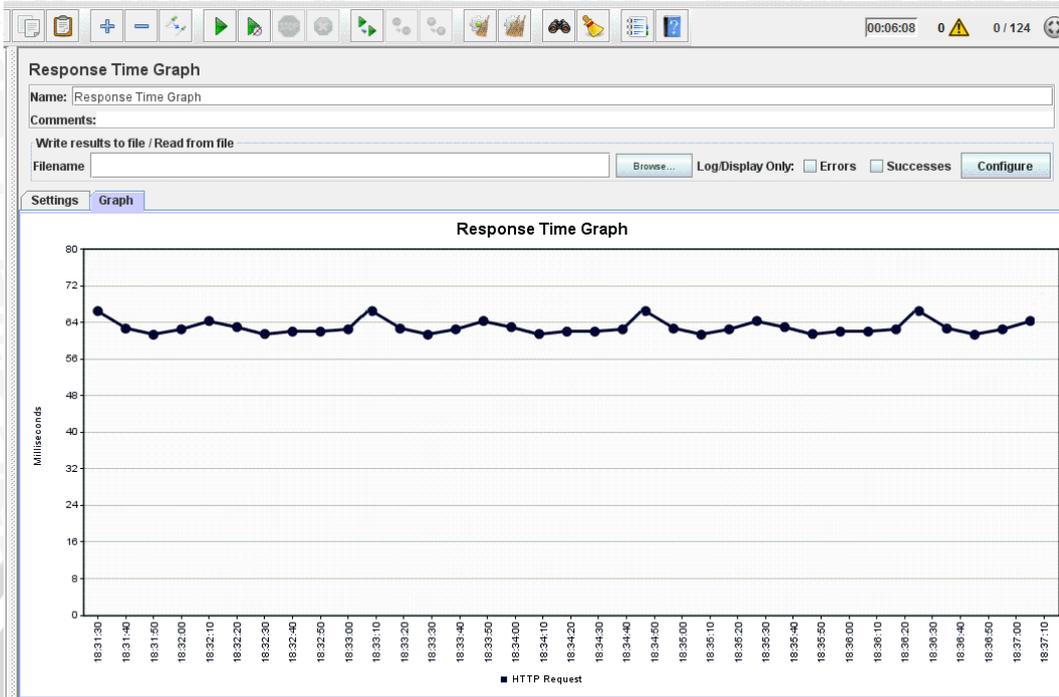
### C.3 Hasil Pengujian Capacity dengan 122 User



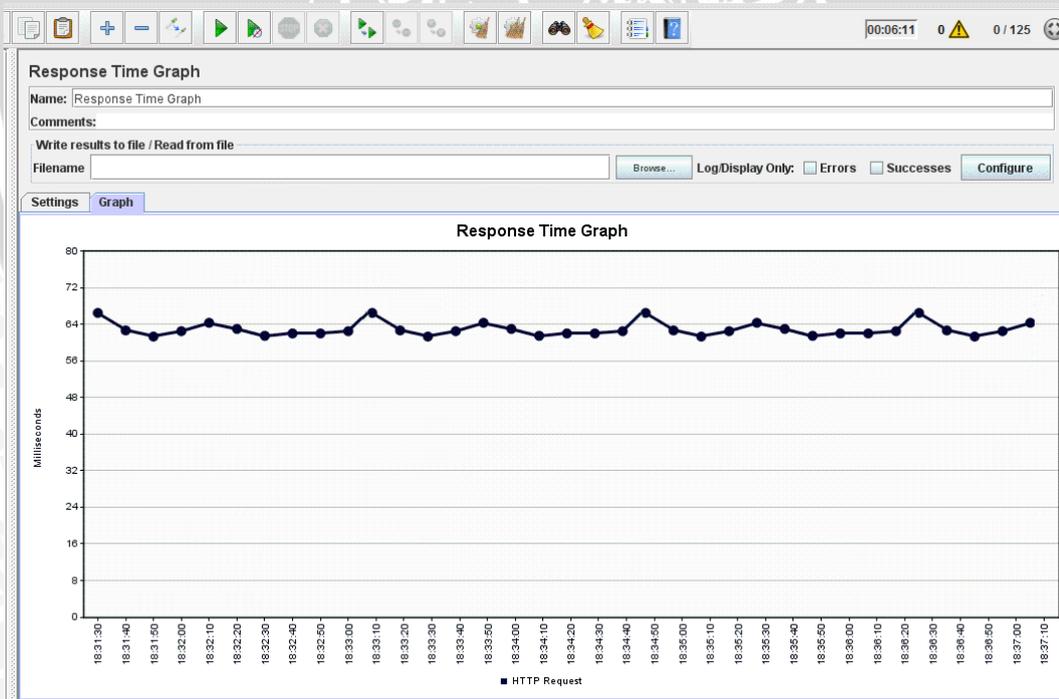
### C.4 Hasil Pengujian Capacity dengan 123 User



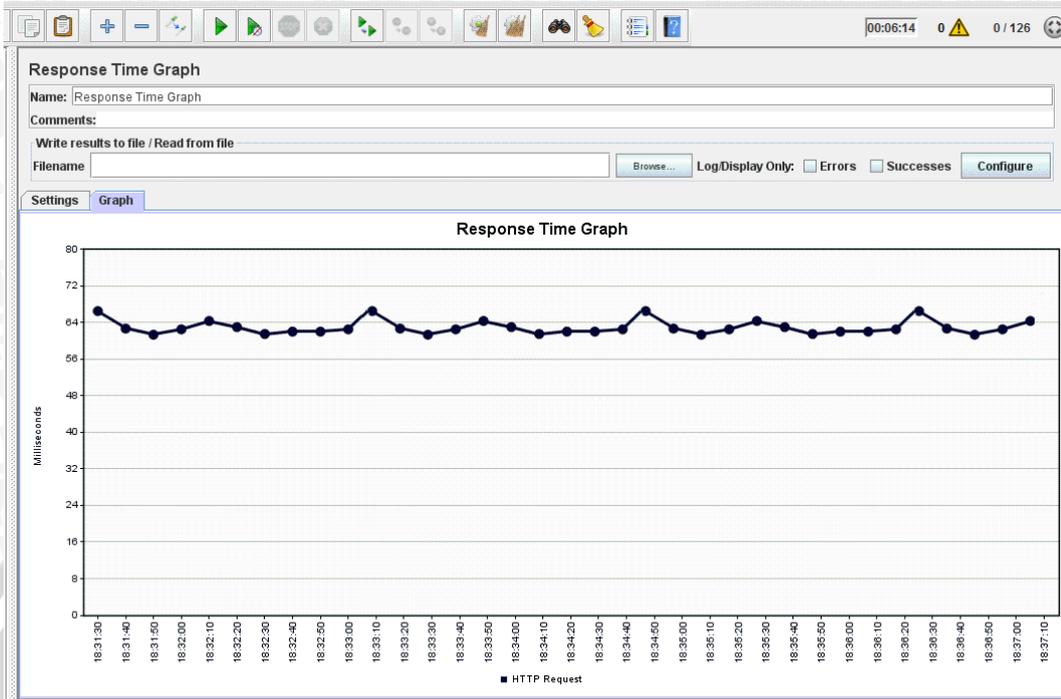
## C.5 Hasil Pengujian Capacity dengan 124 User



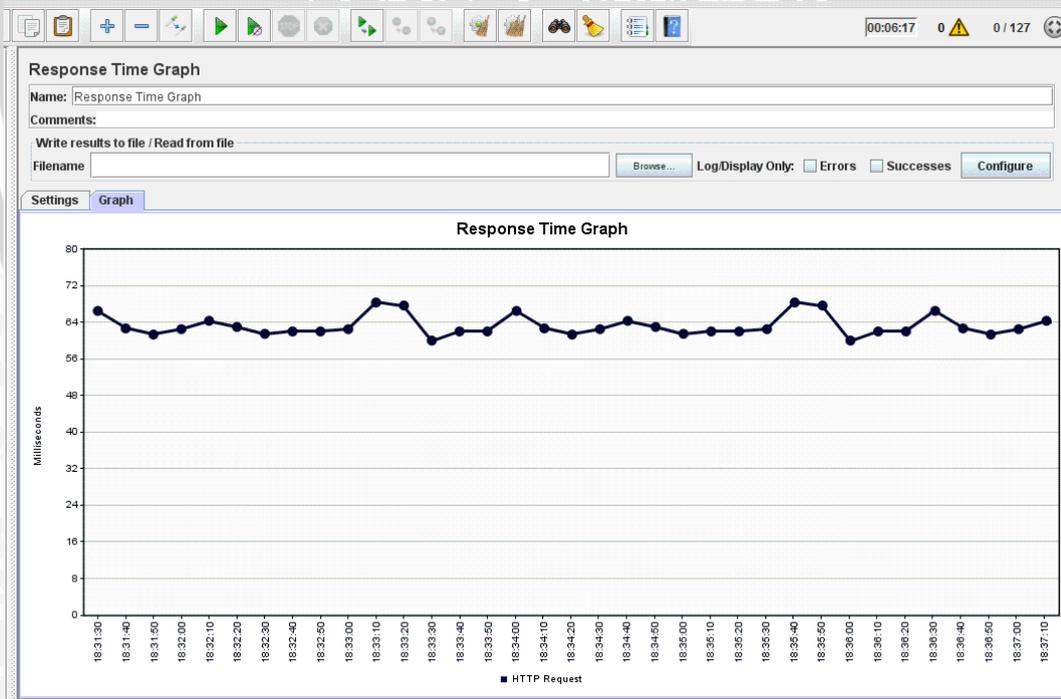
## C.6 Hasil Pengujian Capacity dengan 125 User



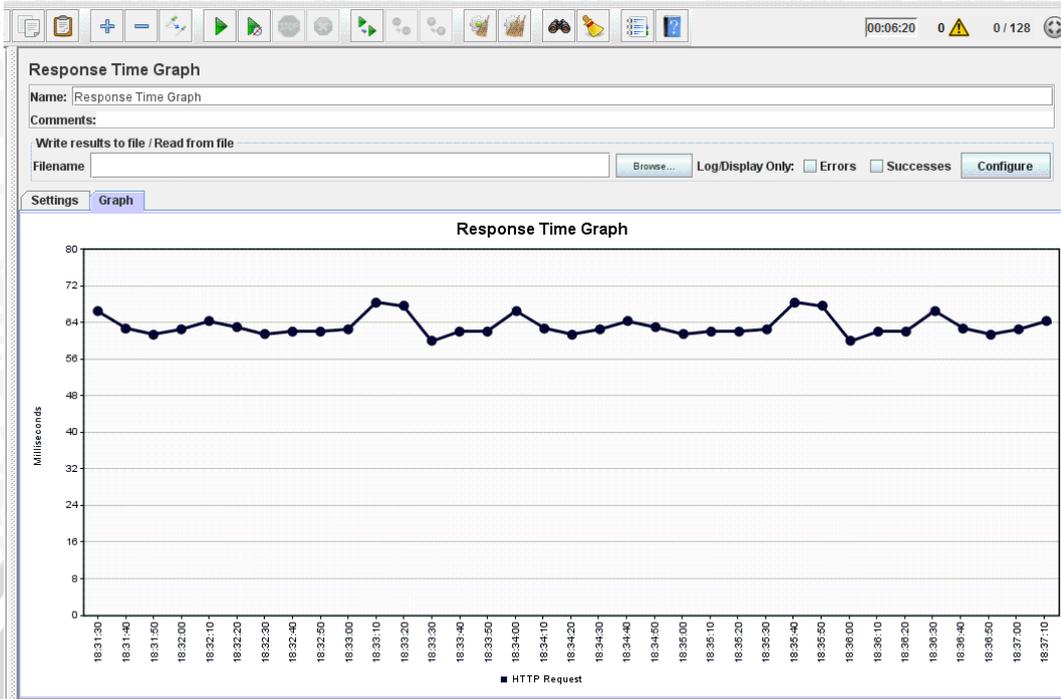
### C.7 Hasil Pengujian Capacity dengan 126 User



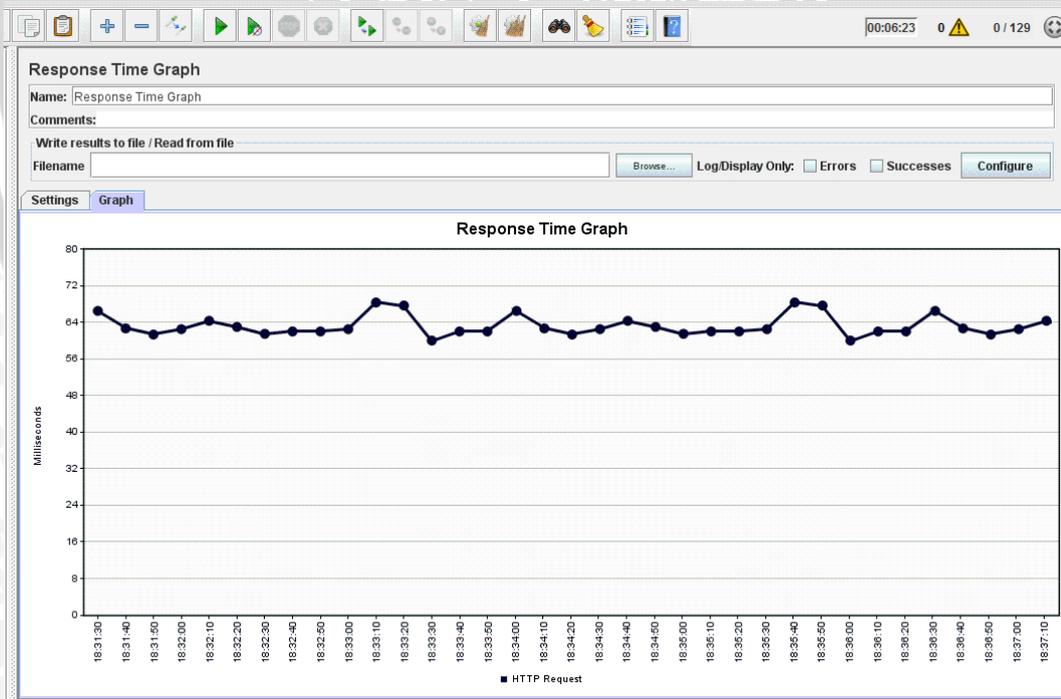
### C.8 Hasil Pengujian Capacity dengan 127 User



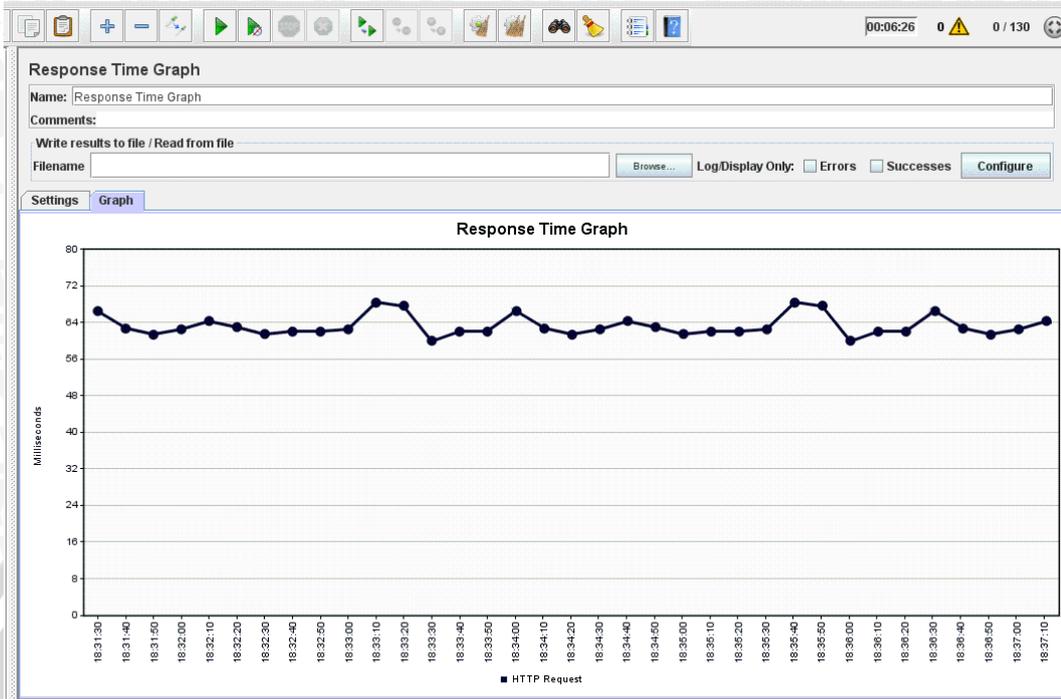
### C.9 Hasil Pengujian Capacity dengan 128 User



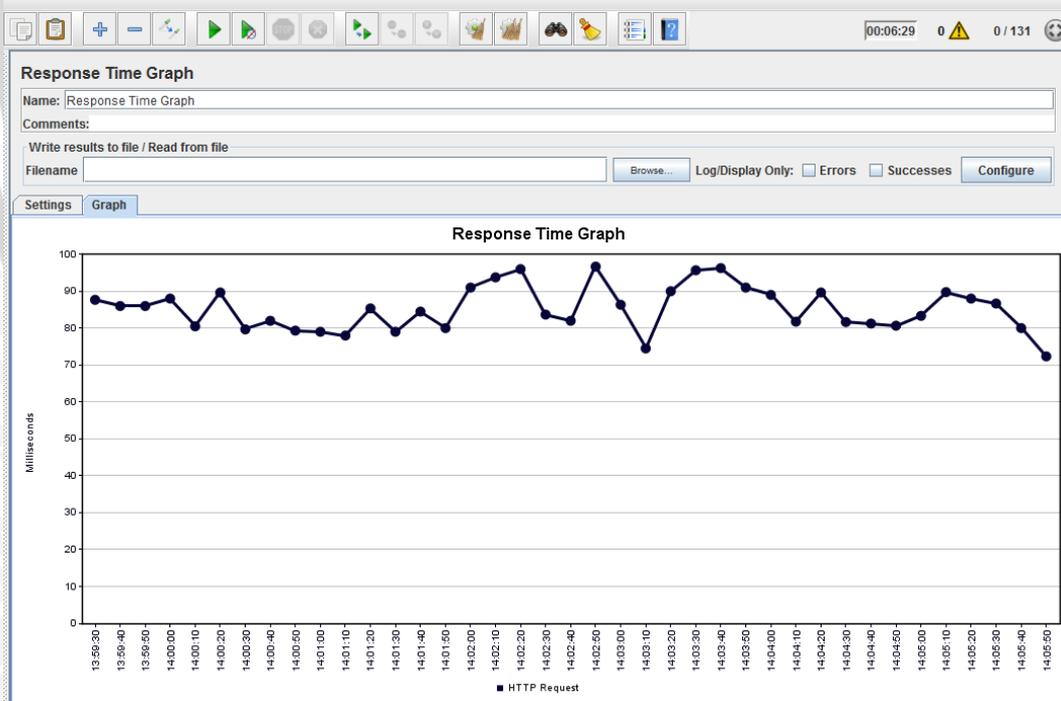
### C.10 Hasil Pengujian Capacity dengan 129 User



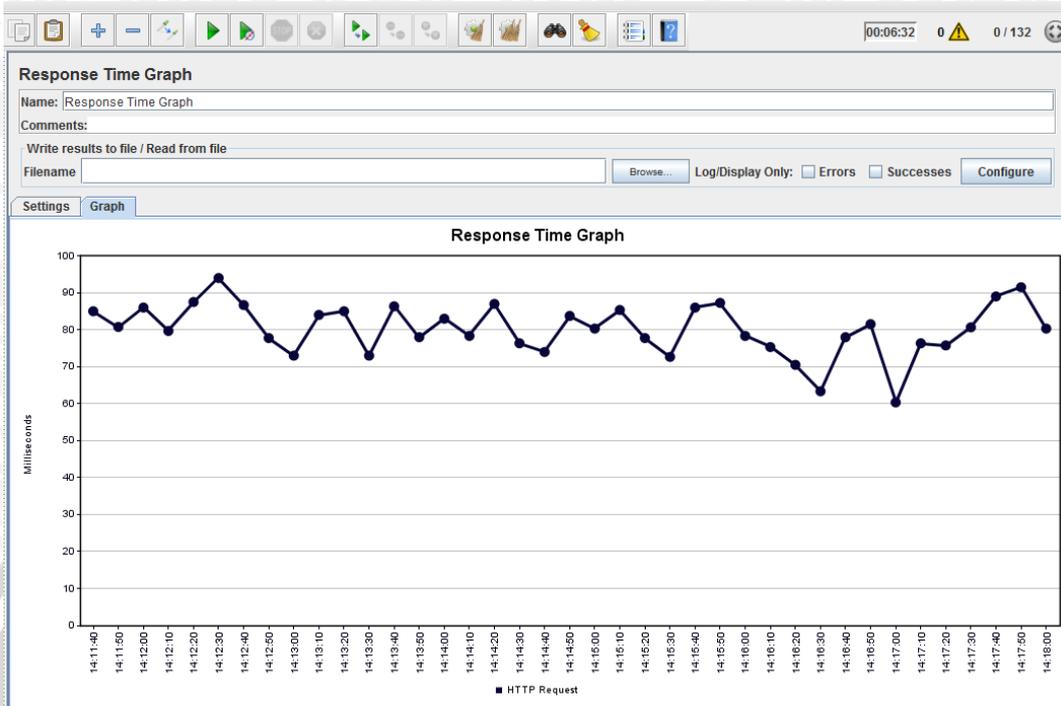
### C.11 Hasil Pengujian Capacity dengan 130 User



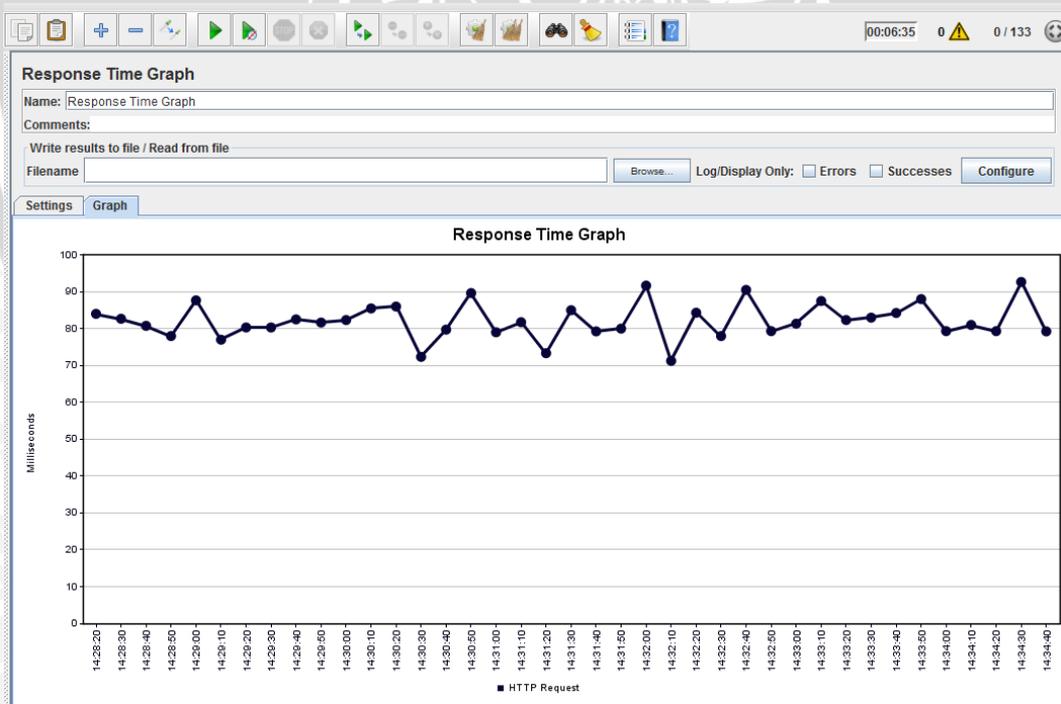
### C.12 Hasil Pengujian Capacity dengan 131 User



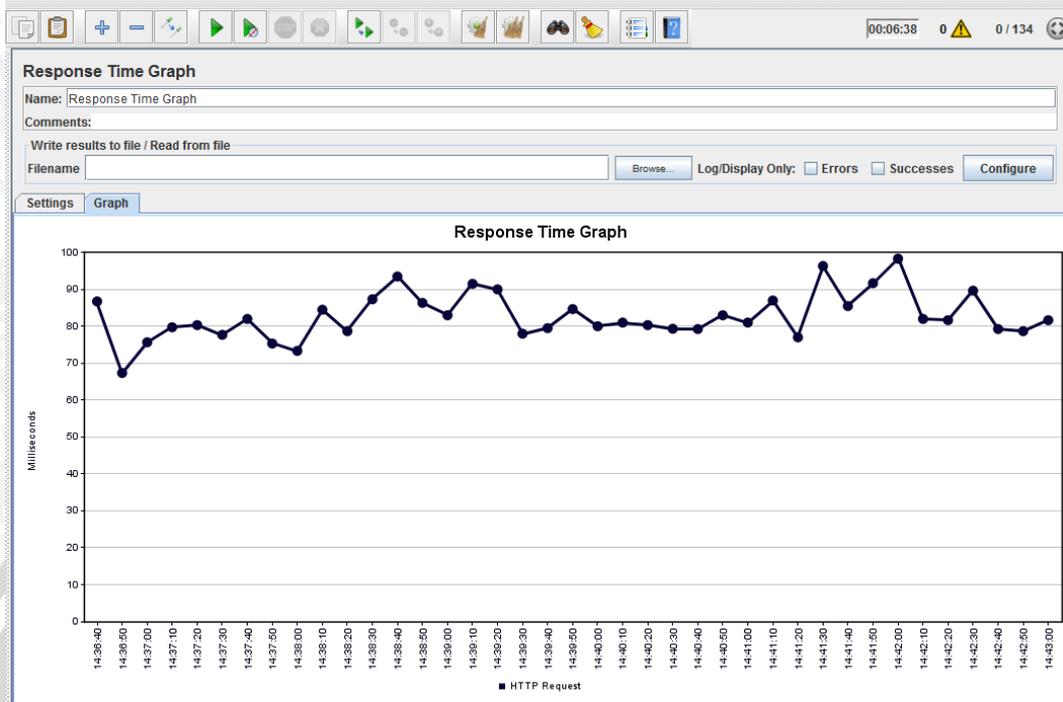
### C.13 Hasil Pengujian *Capacity* dengan 132 User



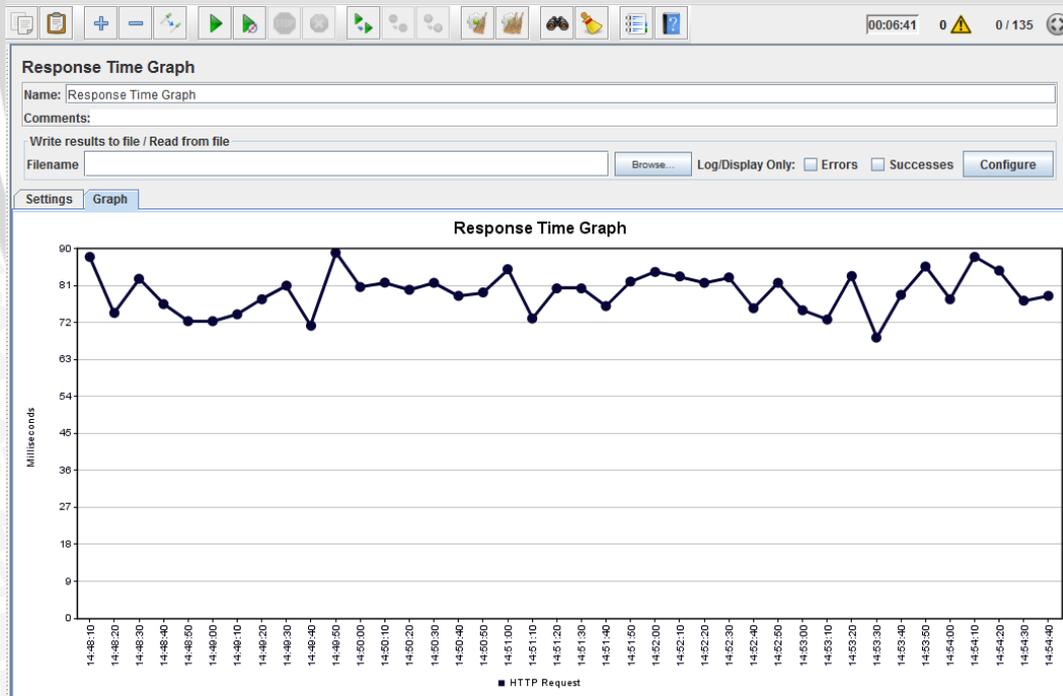
### C.14 Hasil Pengujian *Capacity* dengan 133 User



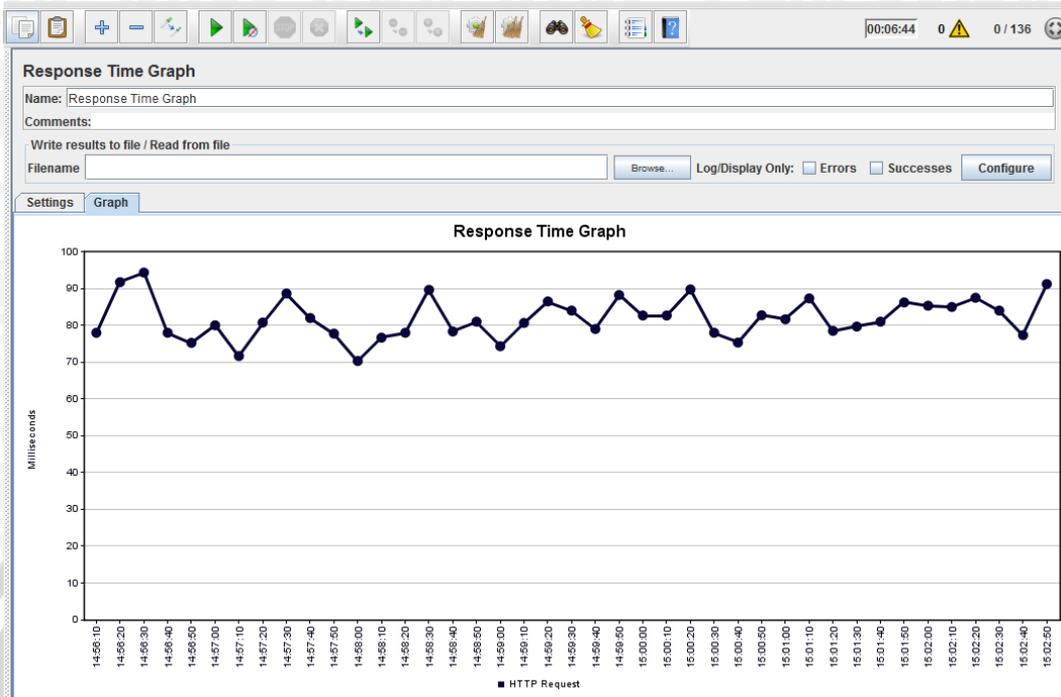
### C.15 Hasil Pengujian Capacity dengan 134 User



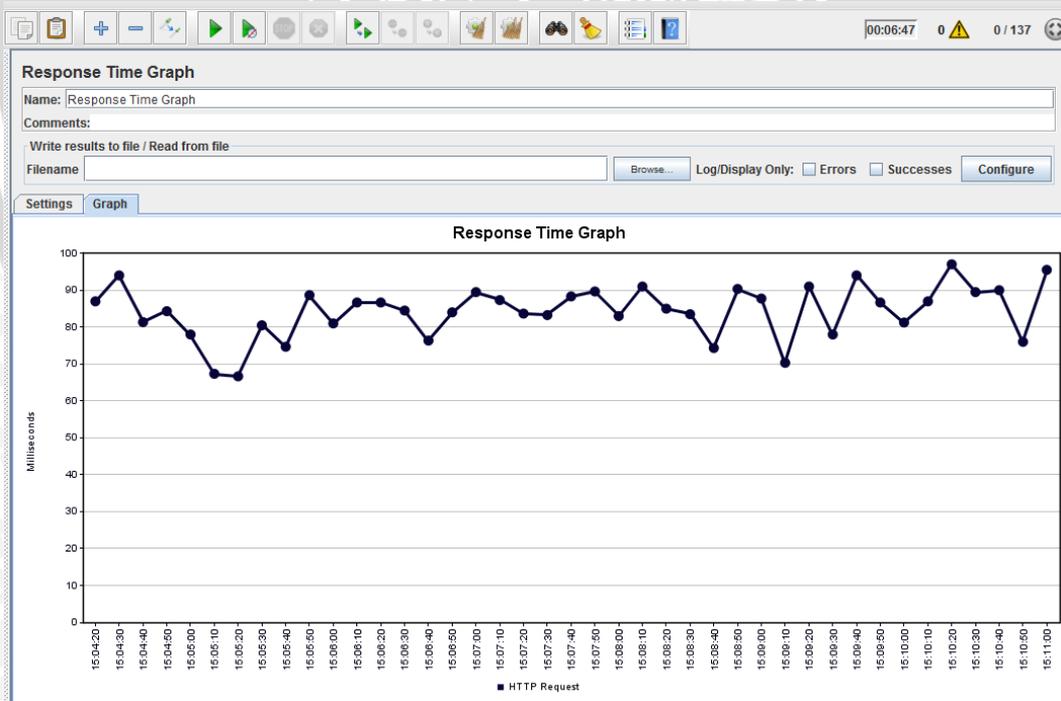
### C.16 Hasil Pengujian Capacity dengan 135 User



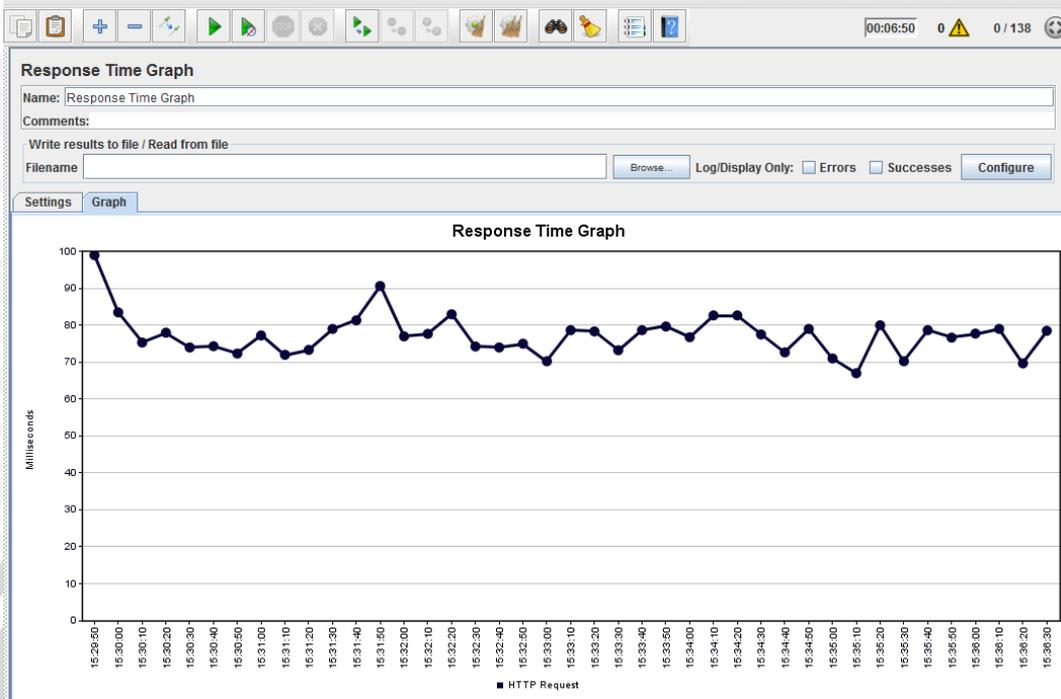
### C.17 Hasil Pengujian Capacity dengan 136 User



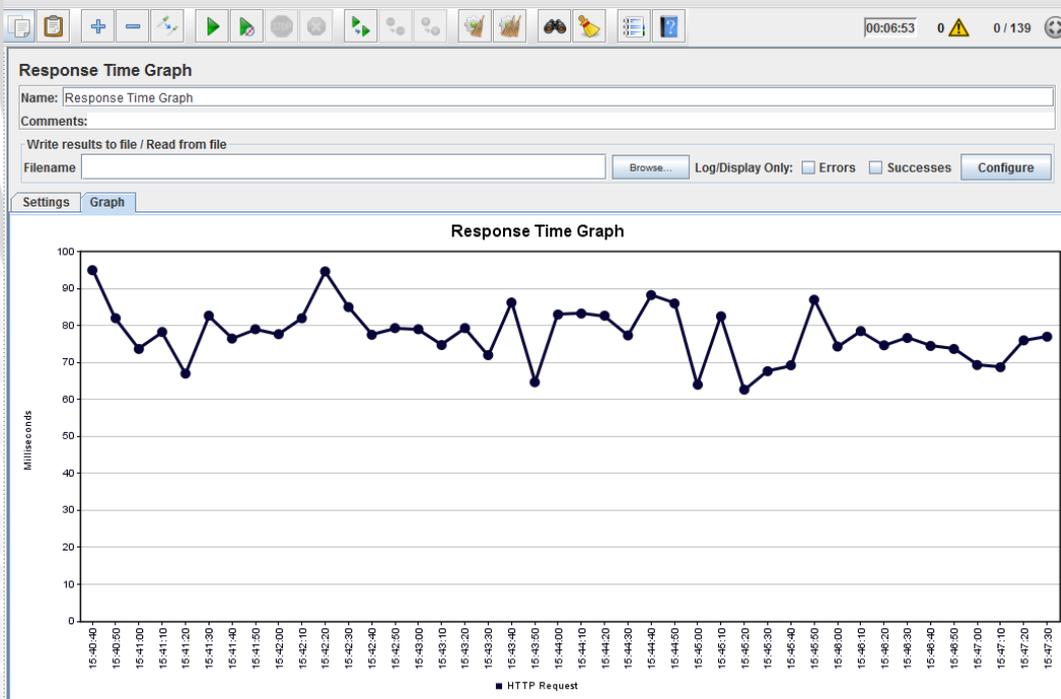
### C.18 Hasil Pengujian Capacity dengan 137 User



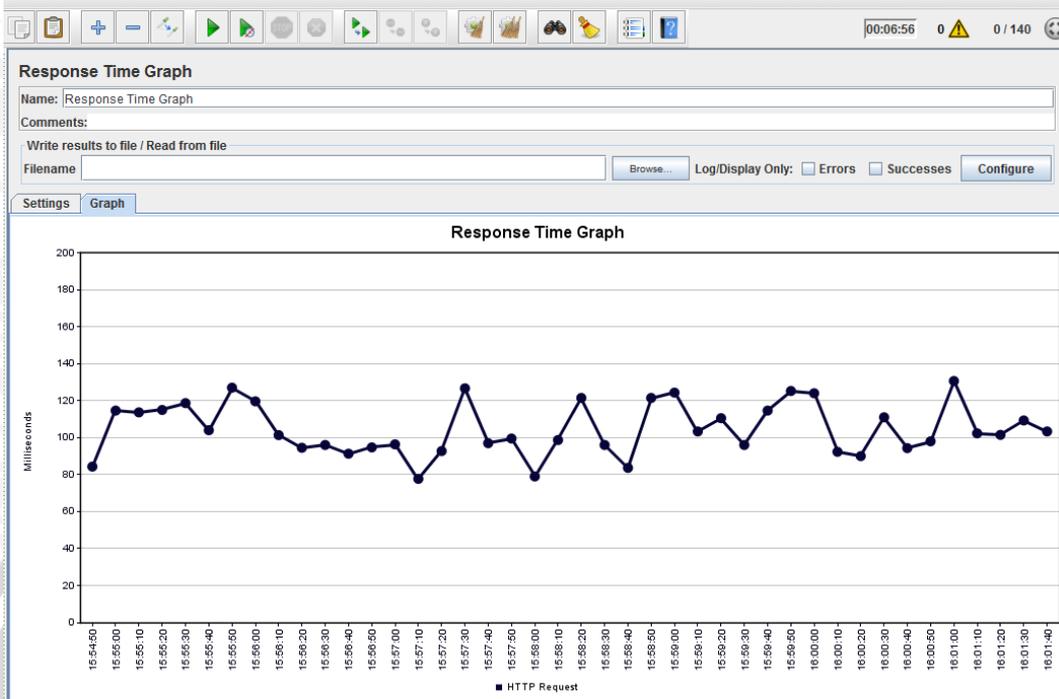
### C.19 Hasil Pengujian Capacity dengan 138 User



### C.20 Hasil Pengujian Capacity dengan 139 User



## C.21 Hasil Pengujian *Capacity* dengan 140 User



## LAMPIRAN D FORM PENGUJIAN USABILITY

### KUESIONER

Evaluasi *Usability* Sistem Seleksi Penerimaan Siswa SMA Baru dengan Model Kuesioner USE Questionnaire.

Nama : \_\_\_\_\_  
 Umur : \_\_\_\_\_  
 Pekerjaan : \_\_\_\_\_

Berilah tanda (√) pada bagian jawaban yang Anda anggap sesuai!  
 STS (Sangat Tidak Setuju), TS (Tidak Setuju), RR (Ragu-Ragu), S (Setuju), SS (Sangat Setuju).

No	Pertanyaan	Jawaban				
		STS	TS	RR	S	SS
1	Sistem ini membantu saya menjadi lebih efektif					
2	Sistem ini membantu saya menjadi lebih produktif					
3	Sistem ini bermanfaat					
4	Sistem ini memberi saya kontrol lebih besar terhadap proses tersebut (proses seleksi penerimaan siswa SMA baru)					
5	Sistem ini mempermudah pekerjaan saya					
6	Sistem ini mempercepat pekerjaan saya					
7	Sistem ini sesuai dengan kebutuhan dalam pekerjaan saya					
8	Sistem ini melakukan segala sesuatu yang saya harapkan					
9	Sistem ini mudah digunakan					
10	Sistem ini tidak rumit untuk dioperasikan					
11	Sistem ini <i>user friendly</i> (tampilan sistem yang mudah digunakan)					
12	Sistem ini tidak membutuhkan langkah yang panjang untuk mencapai apa yang ingin saya lakukan					
13	Fleksibel					
14	Menggunakannya tanpa menggunakan usaha yang besar					
15	Saya dapat menggunakannya tanpa instruksi tertulis					
16	Saya tidak melihat adanya inkonsistensi ketika saya menggunakannya					
17	Pengguna akan menyukainya					
18	Sistem ini dapat menangani apabila ada kesalahan pengguna					
19	Saya dapat menggunakannya dengan sukses setiap kali					
20	Saya dapat dengan cepat terbiasa menggunakan sistem ini					
21	Saya mudah mengingat bagaimana menggunakannya					
22	Sangat mudah untuk belajar menggunakannya					
23	Saya cepat menjadi terampil menggunakan sistem ini					
24	Saya puas dengan sistem ini					

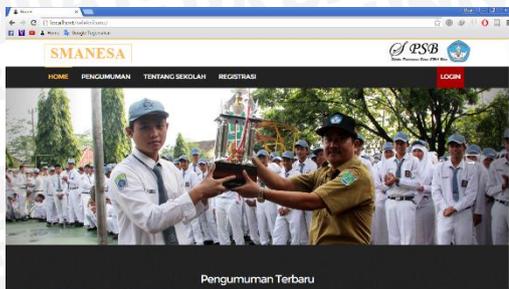


25	Saya akan merekomendasikan sistem ini kepada teman						
26	Sistem ini menyenangkan untuk digunakan						
27	Sistem ini bekerja sesuai dengan yang saya inginkan						
28	Sistem ini bagus						
29	Saya rasa saya menginginkan sistem ini						
30	Sistem ini menyenangkan untuk digunakan						



## LAMPIRAN E HASIL PENGUJIAN PORTABILITY

### E.1 Hasil Pengujian *Portability* Halaman Home



Chrome 53



Firefox 47.0.1



Opera 38.0



iOS 9.3.3

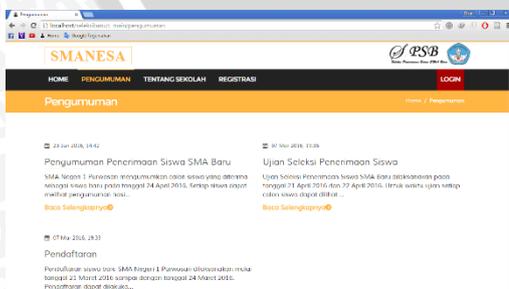


Android 5.1.1

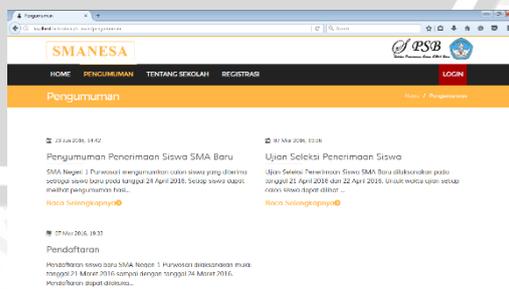


Android 4.2.2

### E.2 Hasil Pengujian *Portability* Halaman Pengumuman

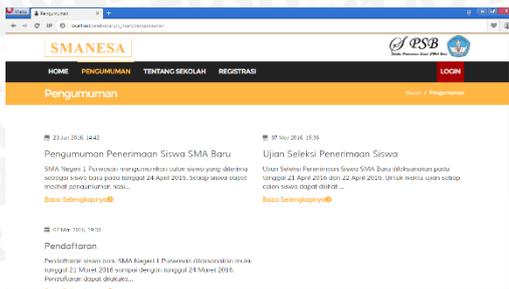


Chrome 53



Firefox 47.0.1





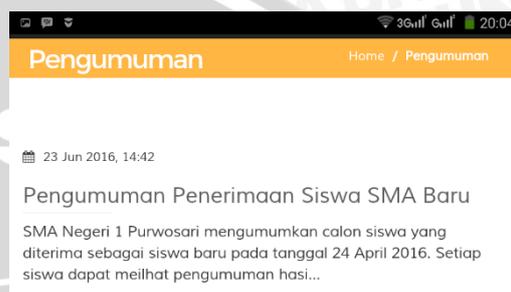
Opera 38.0



iOS 9.3.3



Android 5.1.1

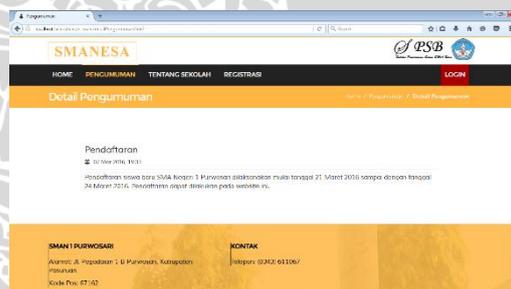


Android 4.2.2

### E.3 Hasil Pengujian *Portability* Halaman Detail Pengumuman



Chrome 53



Firefox 47.0.1



Opera 38.0



iOS 9.3.3



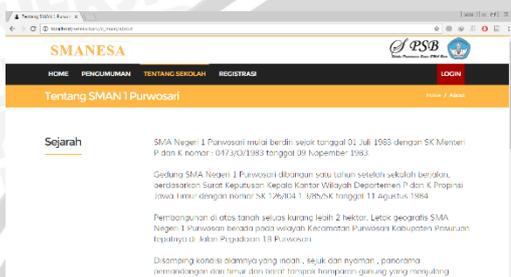


Android 5.1.1



Android 4.2.2

### E.4 Hasil Pengujian *Portability* Halaman Tentang Sekolah



Chrome 53



Firefox 47.0.1



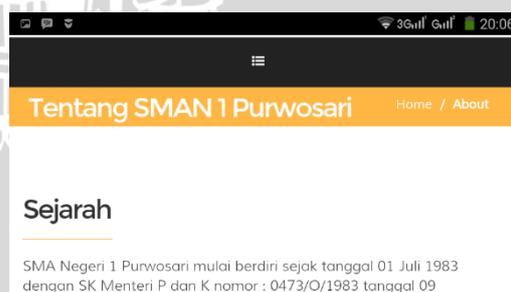
Opera 38.0



iOS 9.3.3



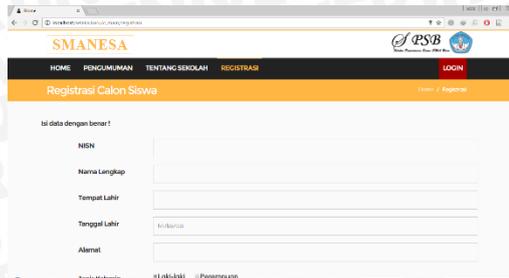
Android 5.1.1



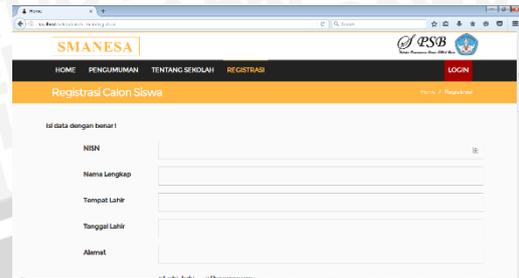
Android 4.2.2



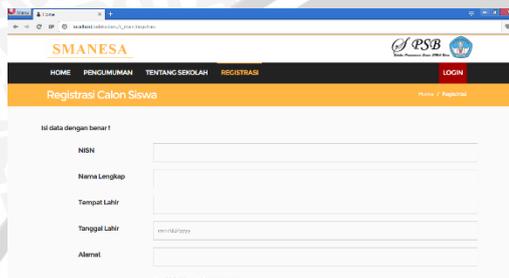
## E.5 Hasil Pengujian *Portability* Halaman Registrasi



Chrome 53



Firefox 47.0.1



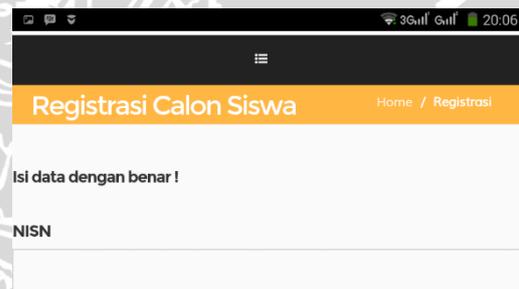
Opera 38.0



iOS 9.3.3

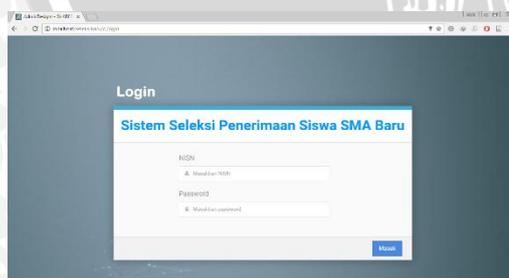


Android 5.1.1

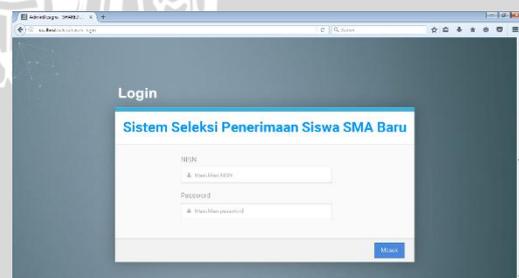


Android 4.2.2

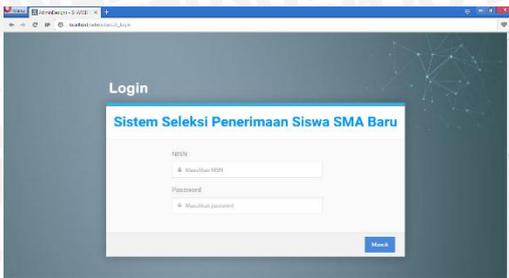
## E.6 Hasil Pengujian *Portability* Halaman Login



Chrome 53



Firefox 47.0.1



Opera 38.0



iOS 9.3.3



Android 5.1.1



Android 4.2.2

### E.7 Hasil Pengujian *Portability* Halaman Home Calon Siswa



Chrome 53



Firefox 47.0.1



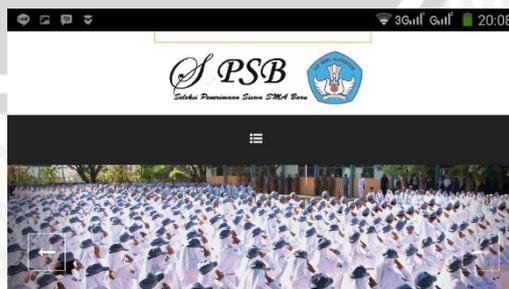
Opera 38.0



iOS 9.3.3



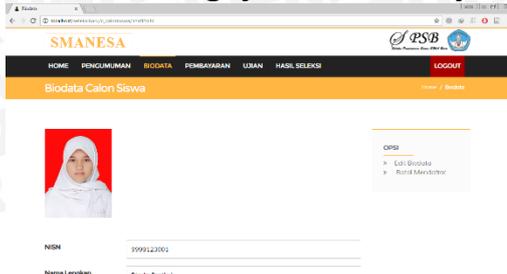
Android 5.1.1



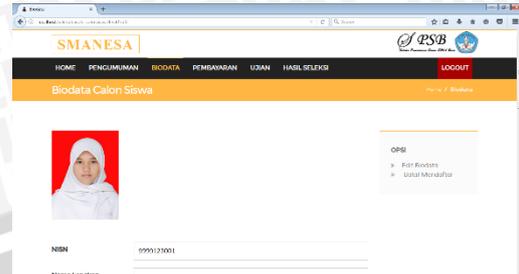
Android 4.2.2



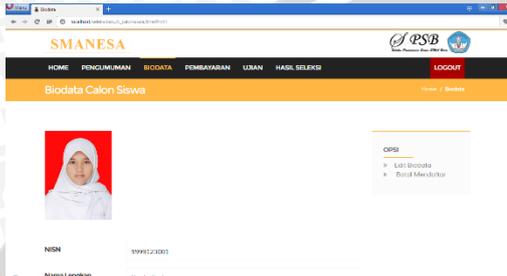
## E.8 Hasil Pengujian Portability Halaman Biodata



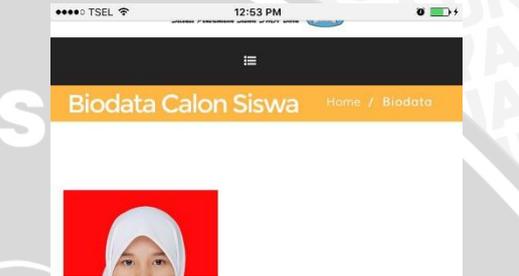
Chrome 53



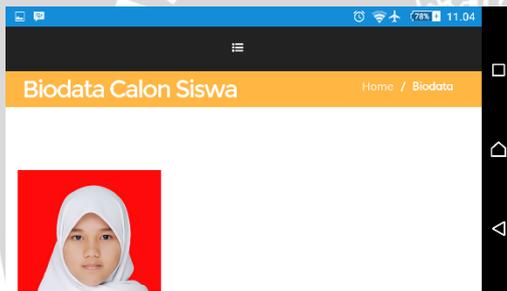
Firefox 47.0.1



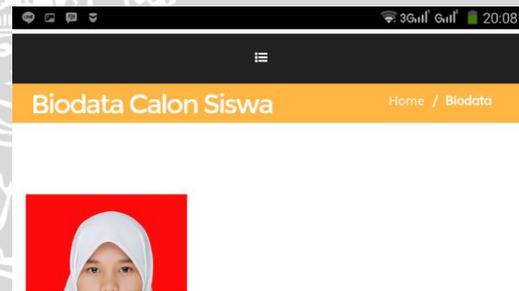
Opera 38.0



iOS 9.3.3



Android 5.1.1



Android 4.2.2

## E.9 Hasil Pengujian Portability Halaman Pembayaran



Chrome 53



Firefox 47.0.1



Opera 38.0

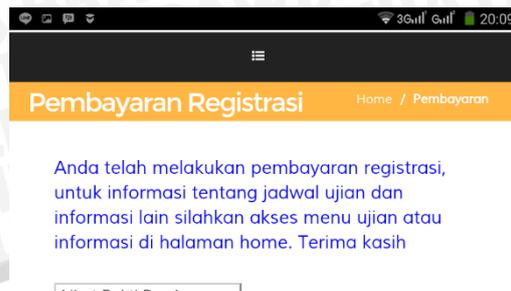


iOS 9.3.3

Anda telah melakukan upload bukti pembayaran. Upload bukti pembayaran akan segera diverifikasi oleh pihak sekolah. Terima kasih

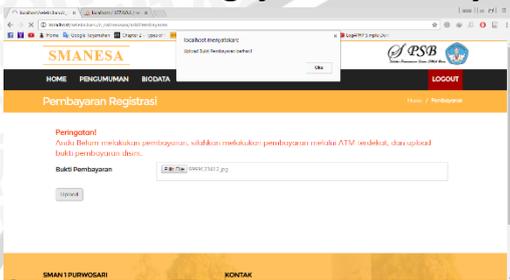


Android 5.1.1

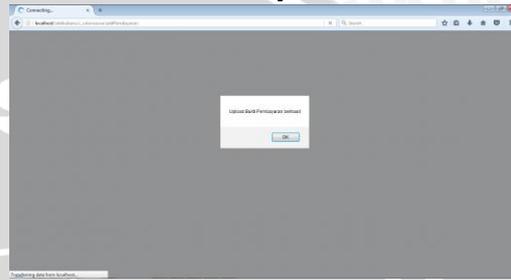


Android 4.2.2

**E.10 Hasil Pengujian Portability Upload Bukti Pembayaran**



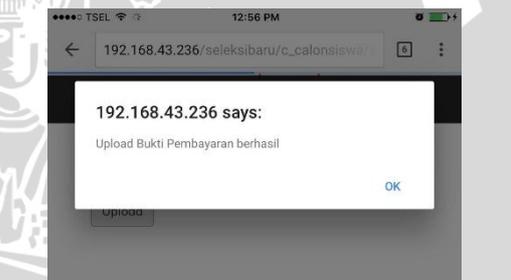
Chrome 53



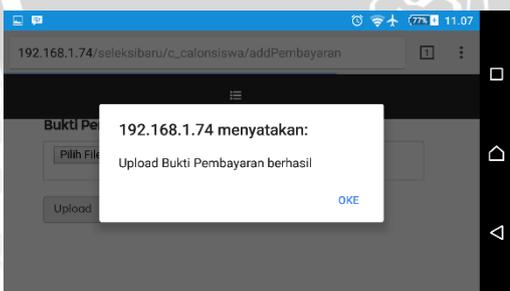
Firefox 47.0.1



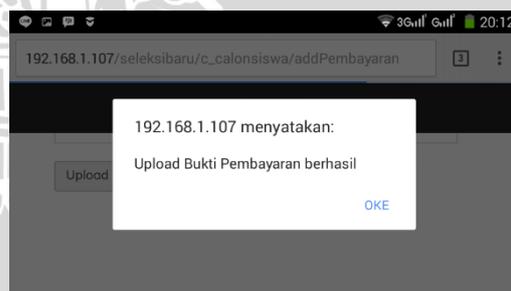
Opera 38.0



iOS 9.3.3

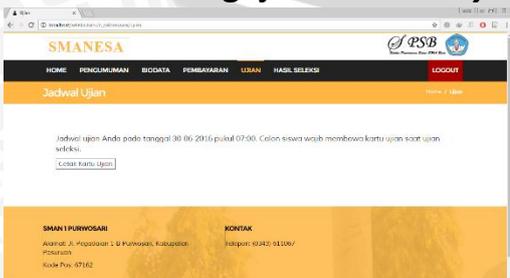


Android 5.1.1



Android 4.2.2

**E.11 Hasil Pengujian Portability Halaman Jadwal Ujian**

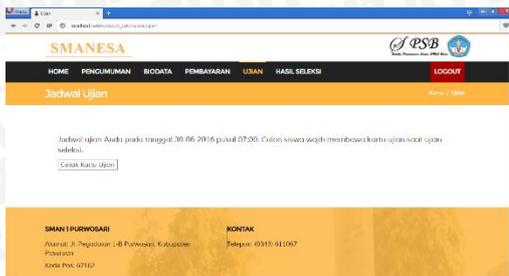


Chrome 53



Firefox 47.0.1





Opera 38.0



iOS 9.3.3



Android 5.1.1

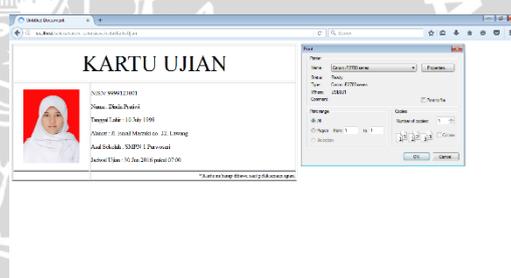


Android 4.2.2

**E.12 Hasil Pengujian Portability Halaman Cetak Kartu Ujian**



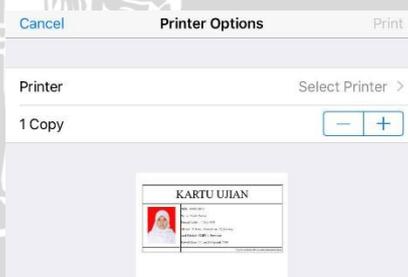
Chrome 53



Firefox 47.0.1



Opera 38.0



iOS 9.3.3



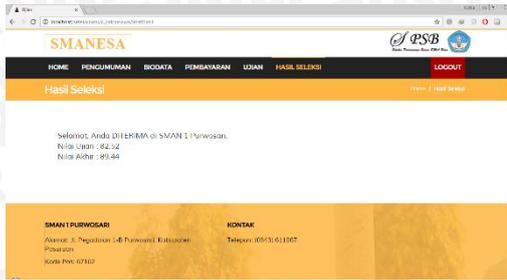
Android 5.1.1



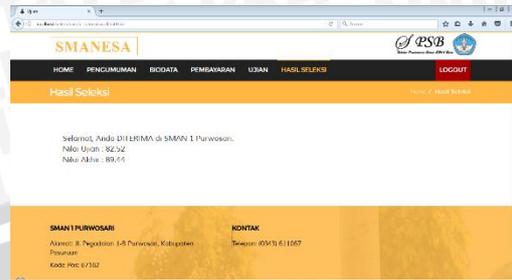
Android 4.2.2



### E.13 Hasil Pengujian Portability Halaman Lihat Hasil Seleksi



Chrome 53



Firefox 47.0.1



Opera 38.0



iOS 9.3.3

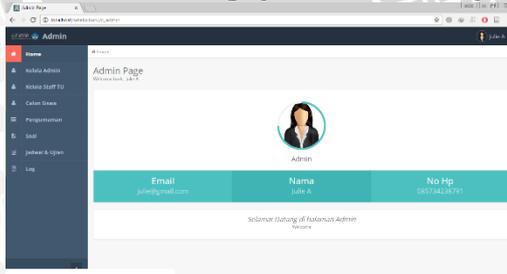


Android 5.1.1

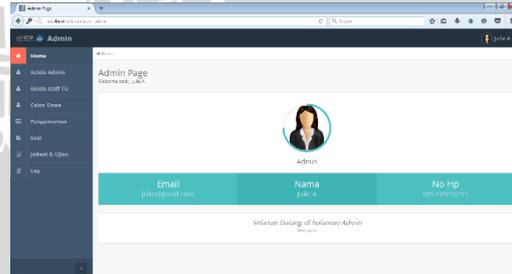


Android 4.2.2

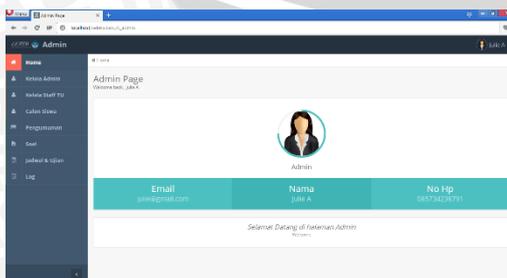
### E.14 Hasil Pengujian Portability Halaman Home Admin



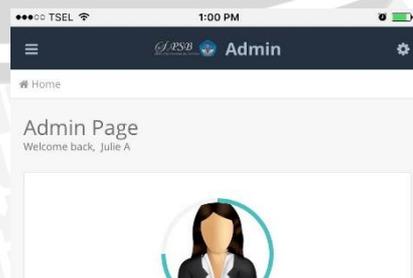
Chrome 53



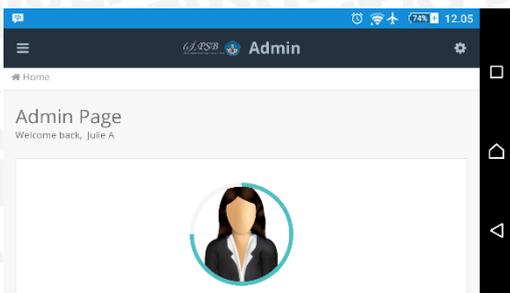
Firefox 47.0.1



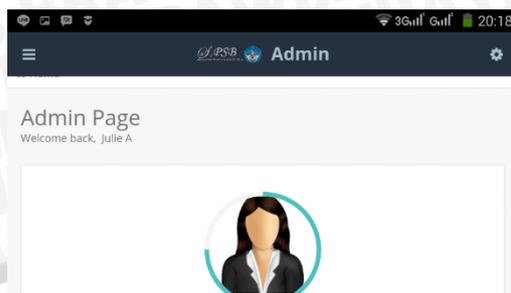
Opera 38.0



iOS 9.3.3

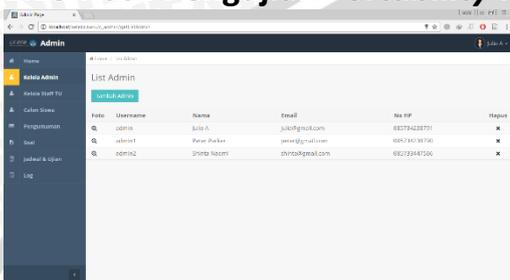


Android 5.1.1

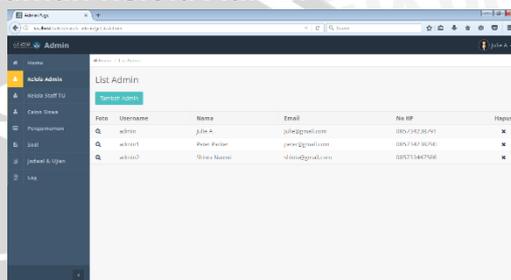


Android 4.2.2

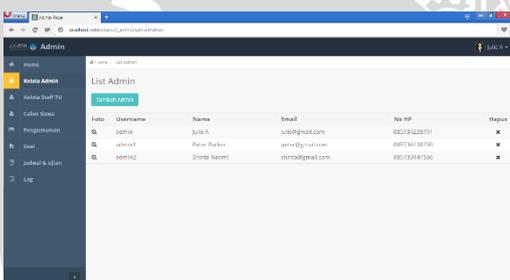
**E.15 Hasil Pengujian Portability Halaman Kelola Admin**



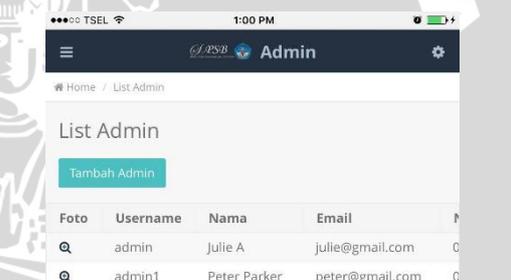
Chrome 53



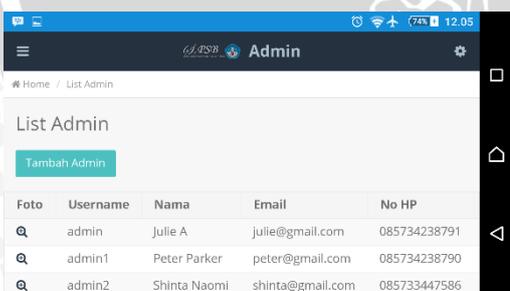
Firefox 47.0.1



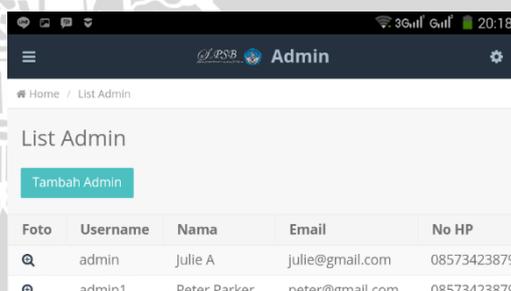
Opera 38.0



iOS 9.3.3

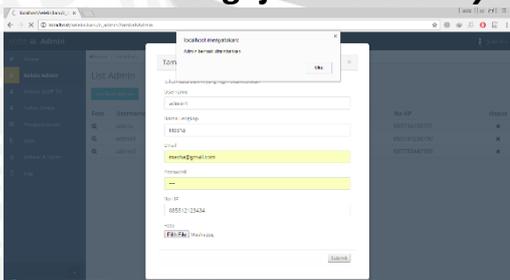


Android 5.1.1

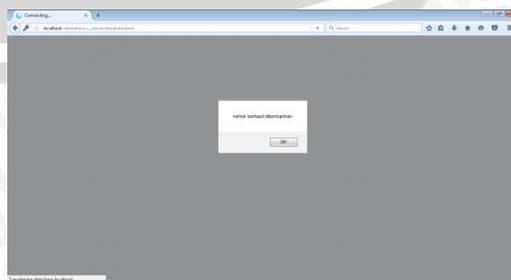


Android 4.2.2

**E.16 Hasil Pengujian Portability Halaman Tambah Admin**

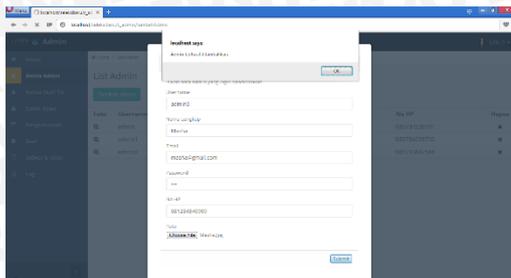


Chrome 53



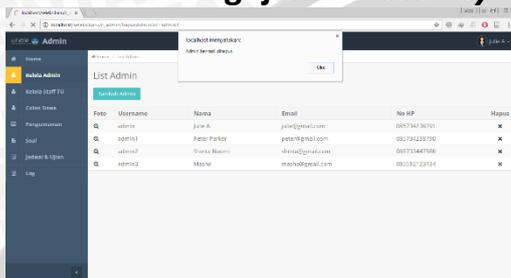
Firefox 47.0.1



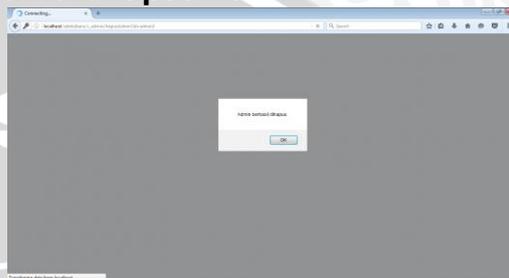


Opera 38.0

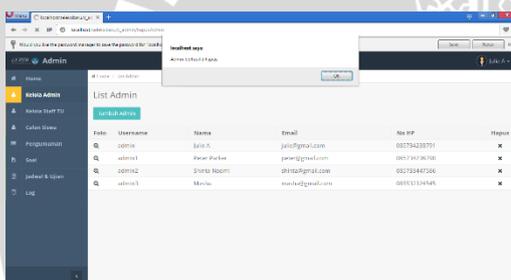
**E.17 Hasil Pengujian Portability Halaman Hapus Admin**



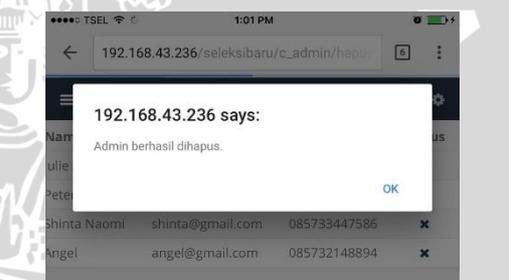
Chrome 53



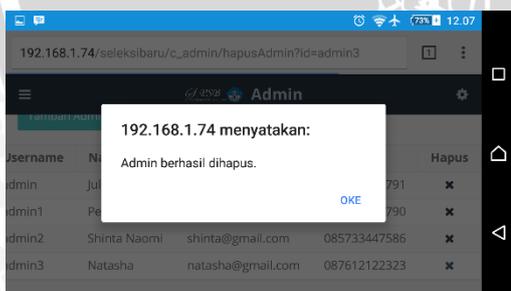
Firefox 47.0.1



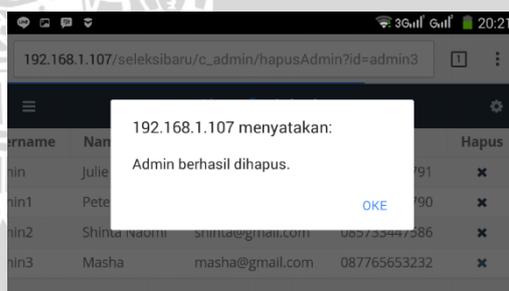
Opera 38.0



iOS 9.3.3

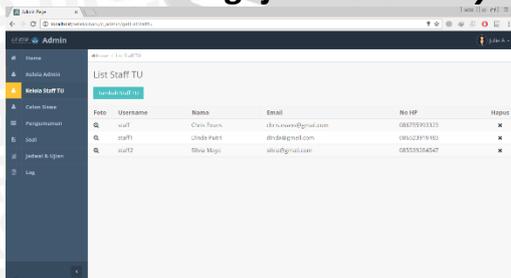


Android 5.1.1

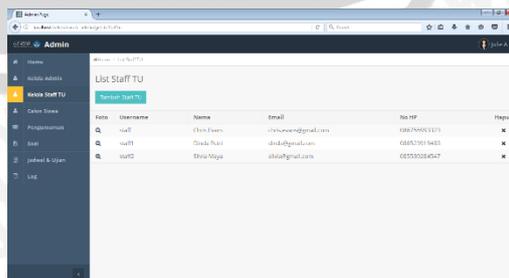


Android 4.2.2

**E.18 Hasil Pengujian Portability Halaman Kelola Staff TU**

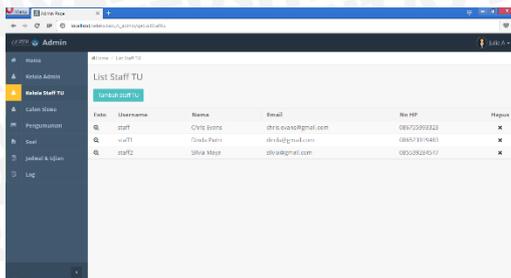


Chrome 53

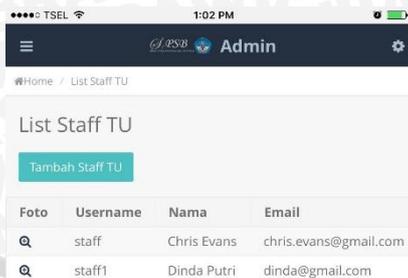


Firefox 47.0.1

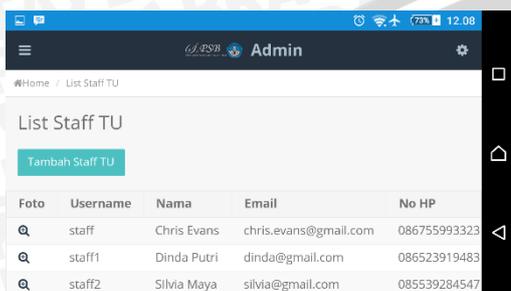




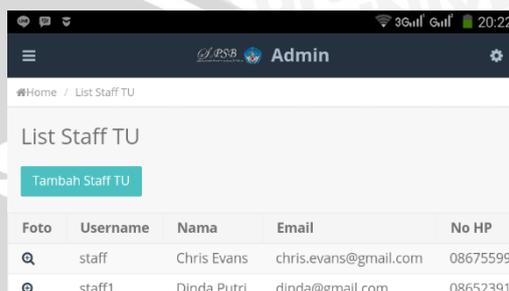
Opera 38.0



iOS 9.3.3

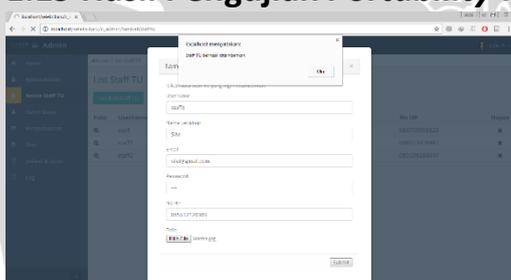


Android 5.1.1

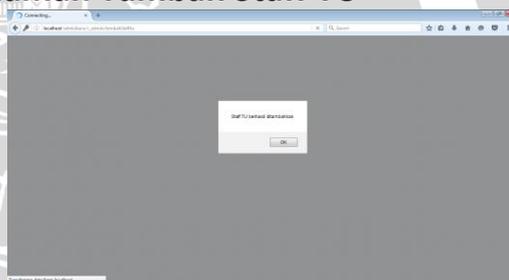


Android 4.2.2

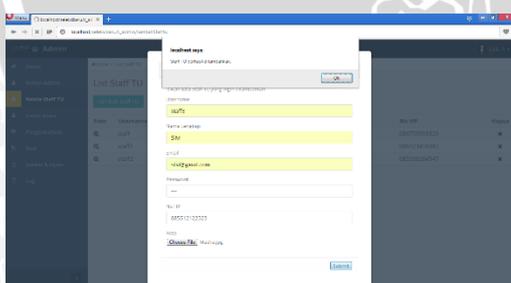
**E.19 Hasil Pengujian Portability Halaman Tambah Staff TU**



Chrome 53

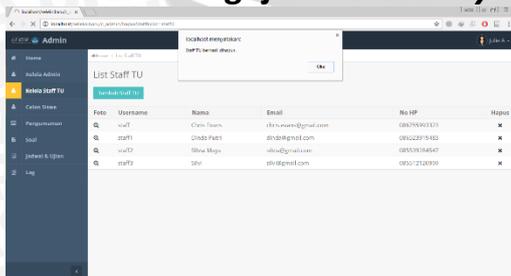


Firefox 47.0.1

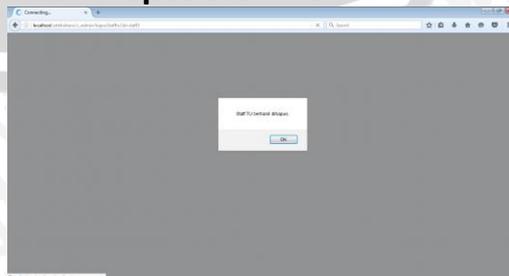


Opera 38.0

**E.20 Hasil Pengujian Portability Halaman Hapus Staff TU**

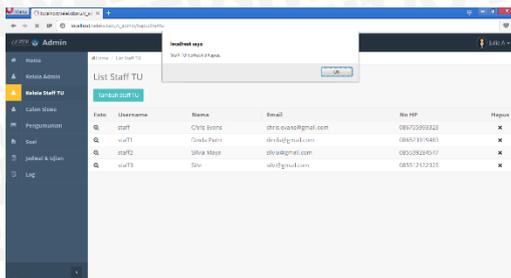


Chrome 53

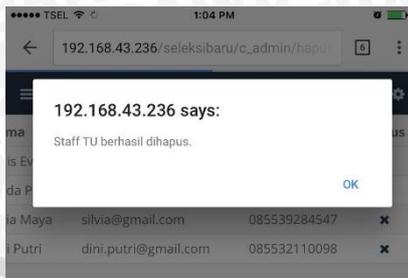


Firefox 47.0.1

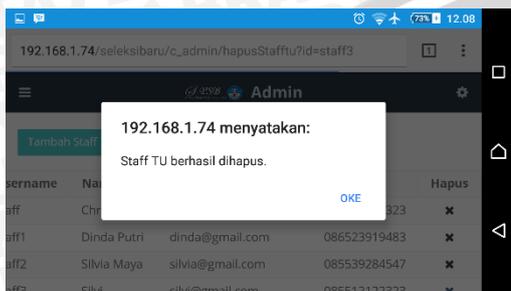




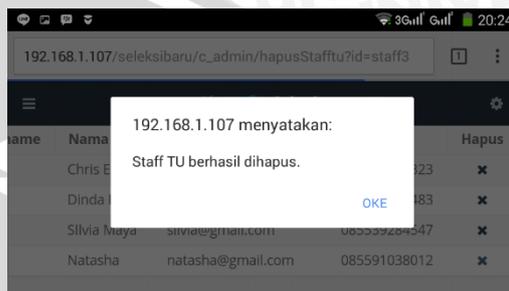
Opera 38.0



iOS 9.3.3

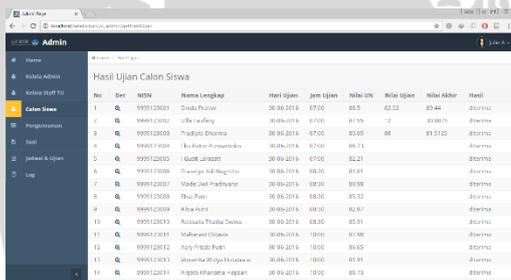


Android 5.1.1

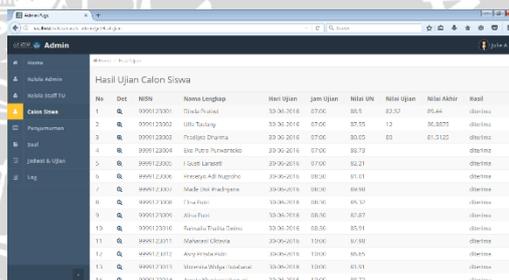


Android 4.2.2

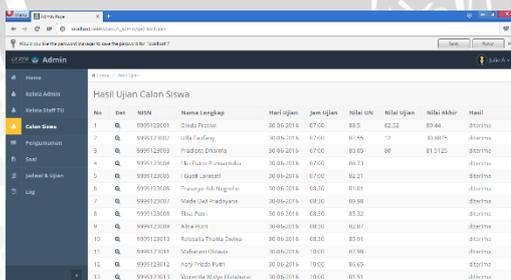
### E.21 Hasil Pengujian Portability Halaman Hasil Ujian Calon Siswa



Chrome 53



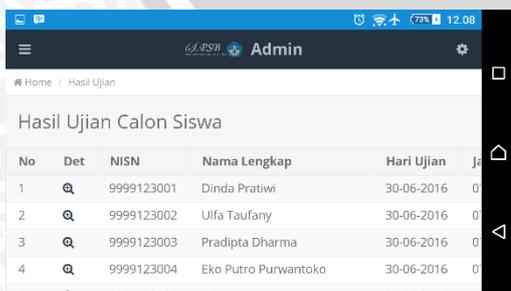
Firefox 47.0.1



Opera 38.0



iOS 9.3.3

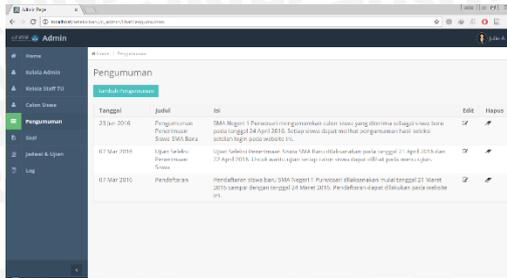


Android 5.1.1

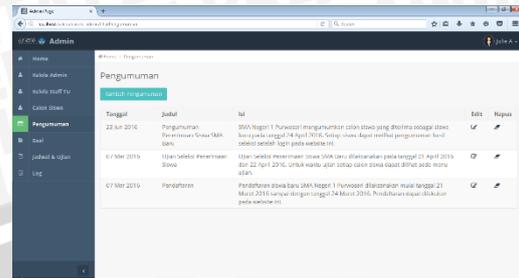


Android 4.2.2

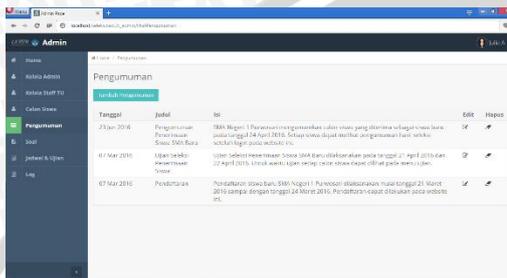
## E.22 Hasil Pengujian Portability Halaman Kelola Pengumuman



Chrome 53



Firefox 47.0.1



Opera 38.0



iOS 9.3.3

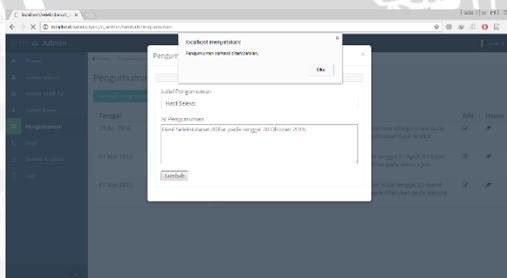


Android 5.1.1

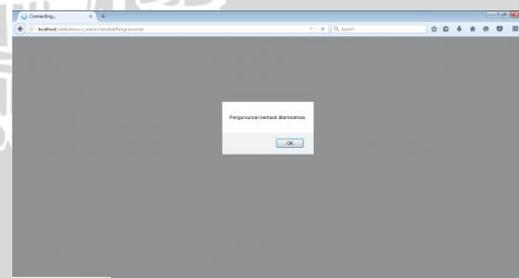


Android 4.2.2

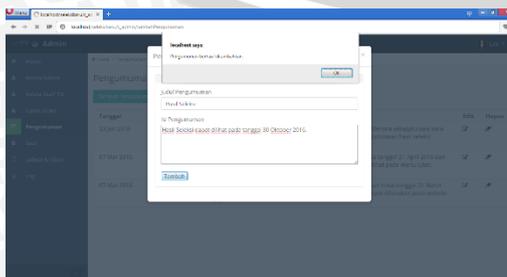
## E.23 Hasil Pengujian Portability Halaman Tambah Pengumuman



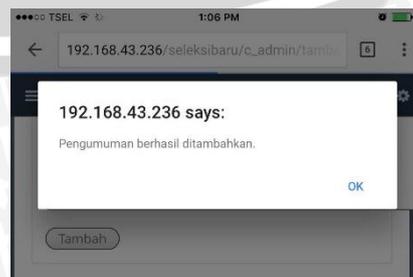
Chrome 53



Firefox 47.0.1

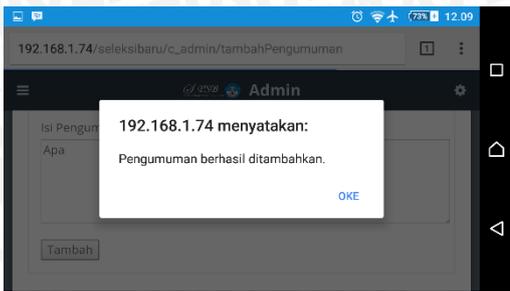


Opera 38.0

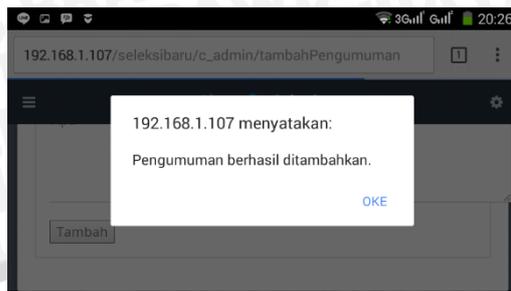


iOS 9.3.3



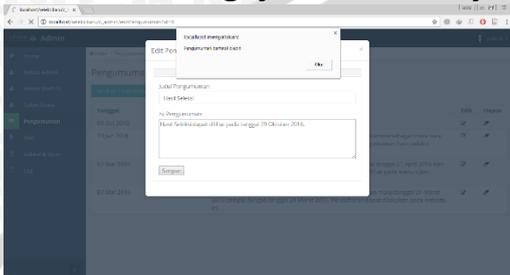


Android 5.1.1

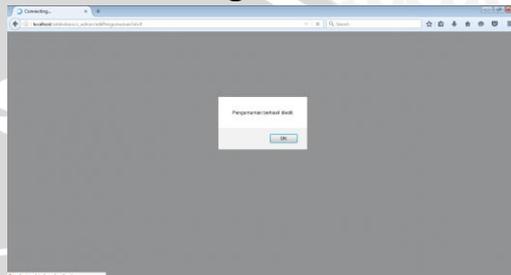


Android 4.2.2

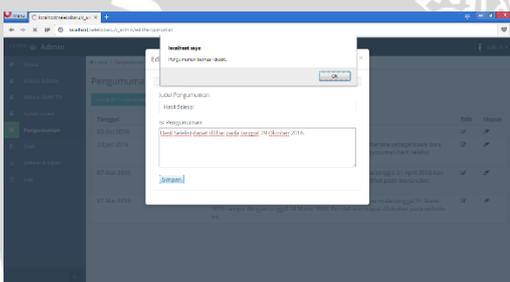
**E.24 Hasil Pengujian Portability Halaman Edit Pengumuman**



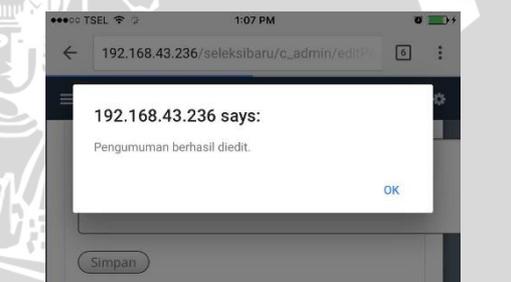
Chrome 53



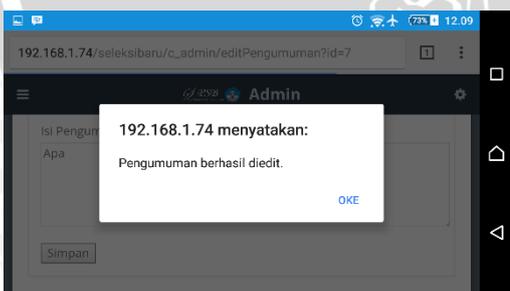
Firefox 47.0.1



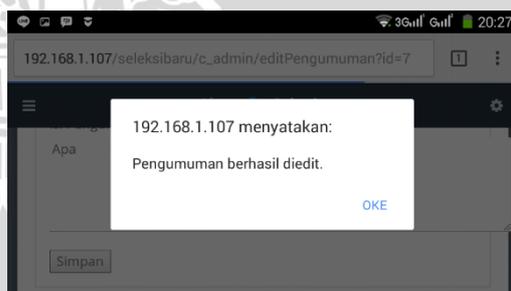
Opera 38.0



iOS 9.3.3

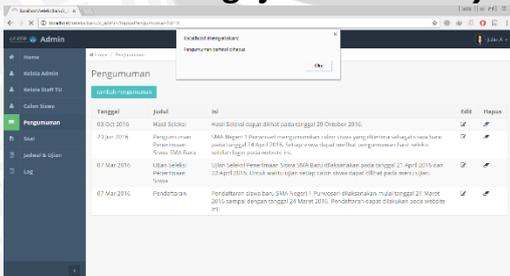


Android 5.1.1

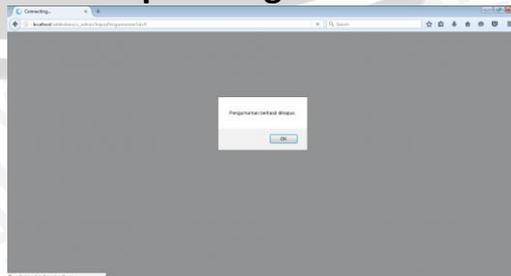


Android 4.2.2

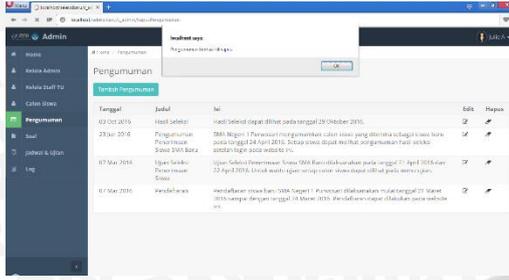
**E.25 Hasil Pengujian Portability Halaman Hapus Pengumuman**



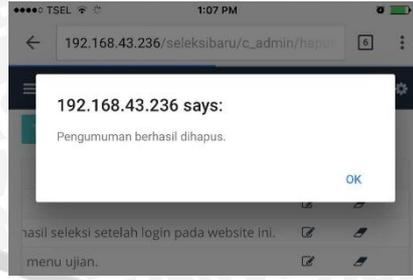
Chrome 53



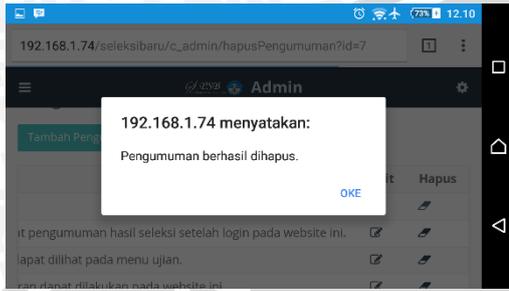
Firefox 47.0.1



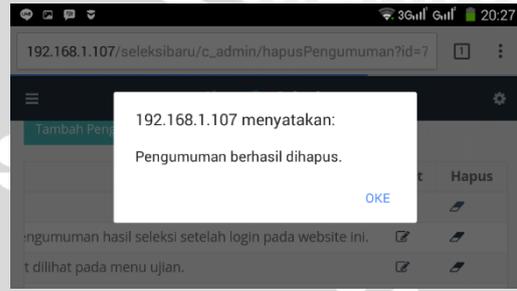
Opera 38.0



iOS 9.3.3

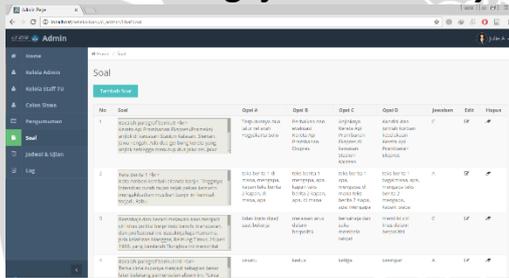


Android 5.1.1

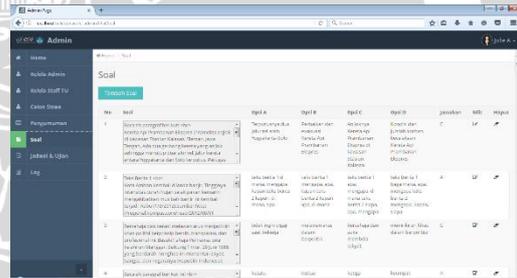


Android 4.2.2

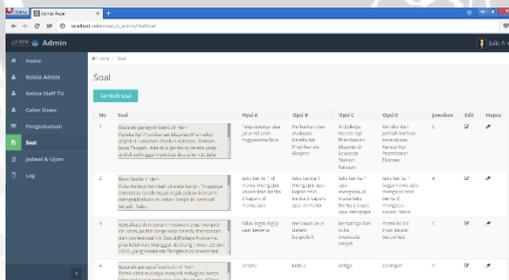
### E.26 Hasil Pengujian Portability Halaman Kelola Soal



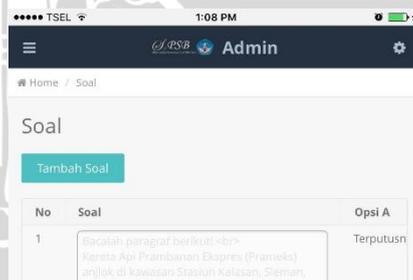
Chrome 53



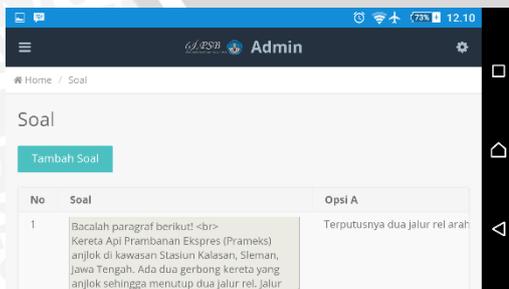
Firefox 47.0.1



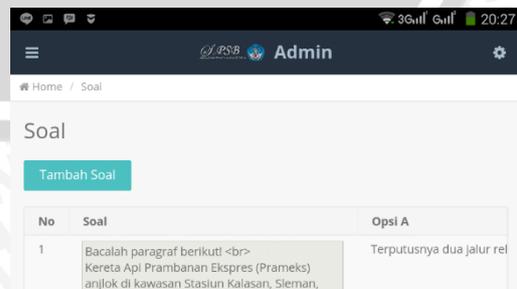
Opera 38.0



iOS 9.3.3



Android 5.1.1



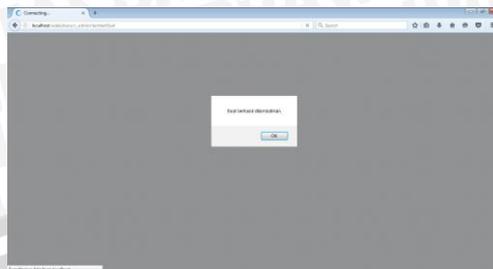
Android 4.2.2



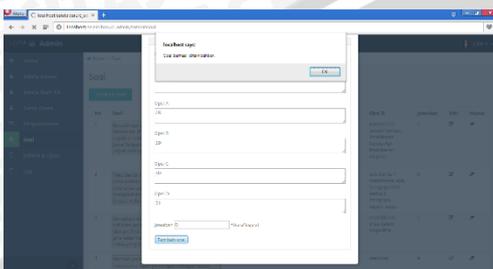
### E.27 Hasil Pengujian Portability Halaman Tambah Soal



Chrome 53



Firefox 47.0.1

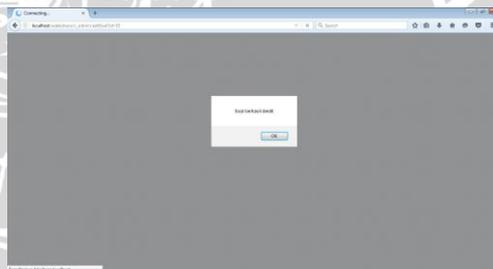


Opera 38.0

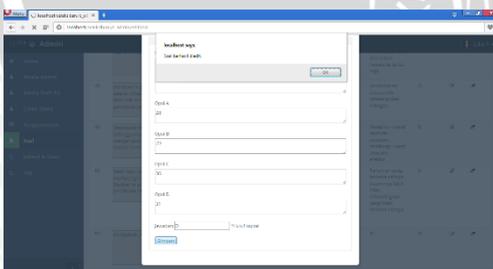
### E.28 Hasil Pengujian Portability Halaman Edit Soal



Chrome 53

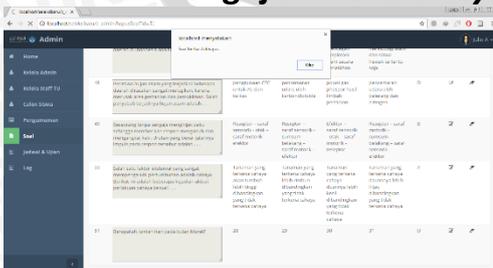


Firefox 47.0.1

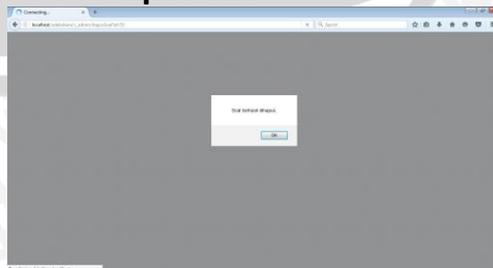


Opera 38.0

### E.29 Hasil Pengujian Portability Halaman Hapus Soal

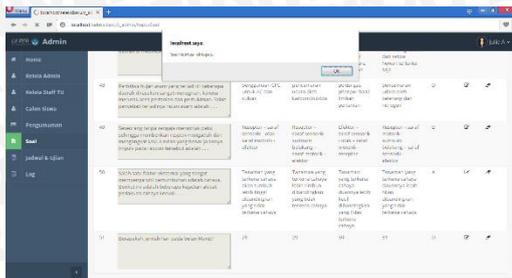


Chrome 53



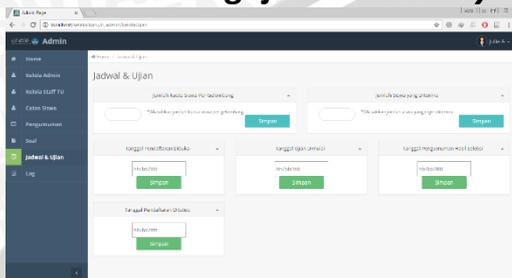
Firefox 47.0.1



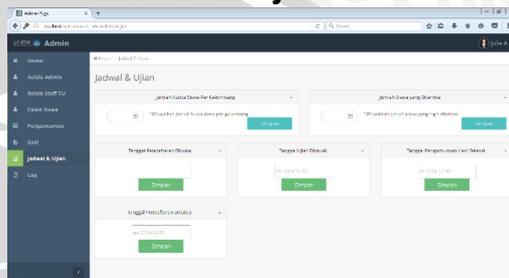


Opera 38.0

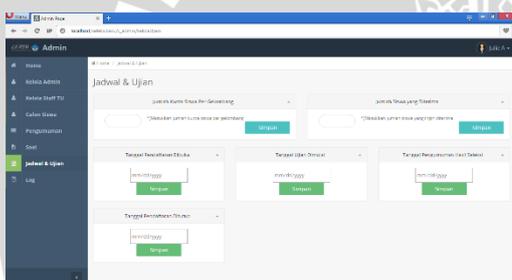
### E.30 Hasil Pengujian Portability Halaman Jadwal & Ujian



Chrome 53



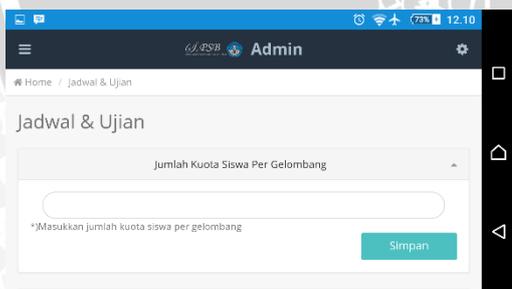
Firefox 47.0.1



Opera 38.0



iOS 9.3.3

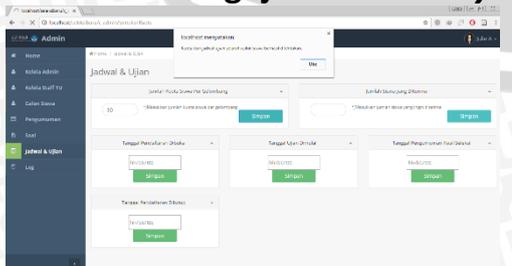


Android 5.1.1



Android 4.2.2

### E.31 Hasil Pengujian Portability Menentukan Kuota Per Gelombang

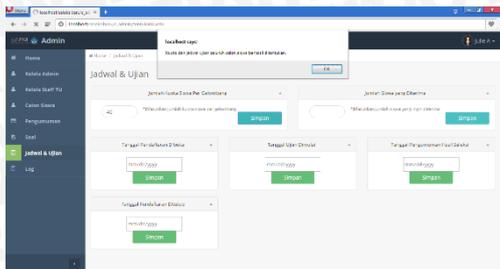


Chrome 53



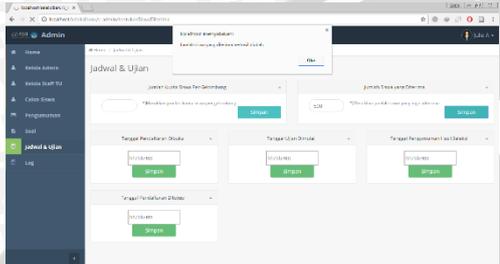
Firefox 47.0.1



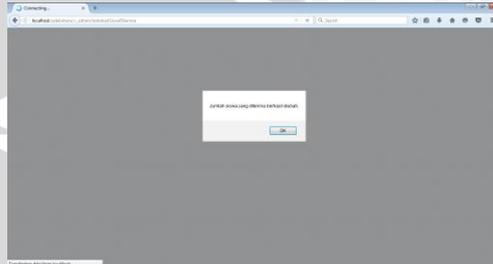


Opera 38.0

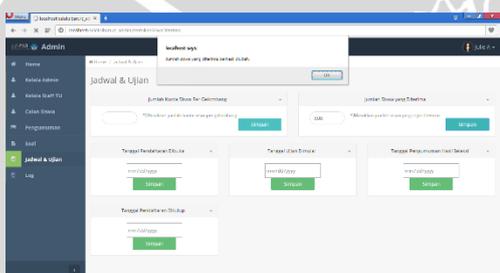
### E.32 Hasil Pengujian Portability Menentukan Jumlah Siswa Diterima



Chrome 53

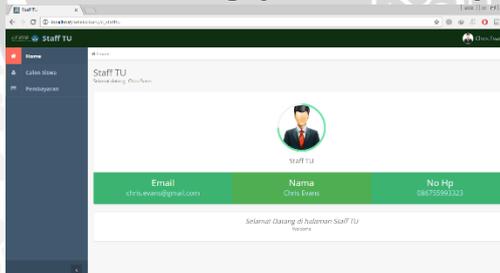


Firefox 47.0.1

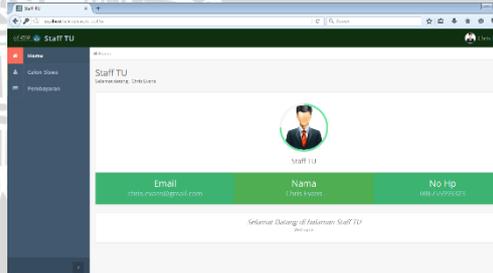


Opera 38.0

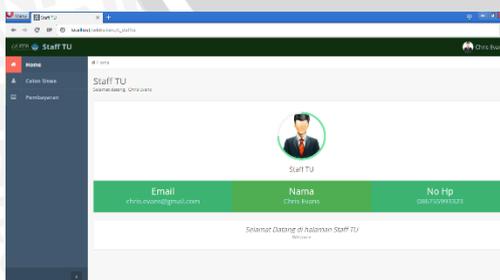
### E.33 Hasil Pengujian Portability Halaman Home Staff TU



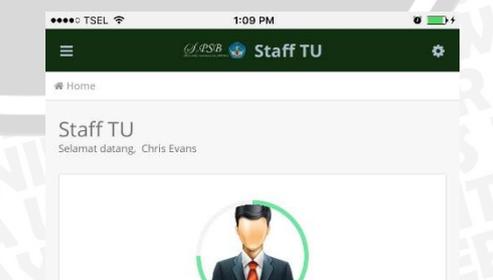
Chrome 53



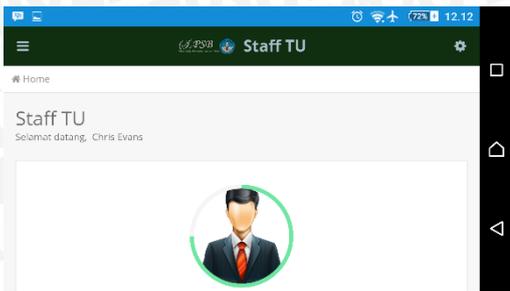
Firefox 47.0.1



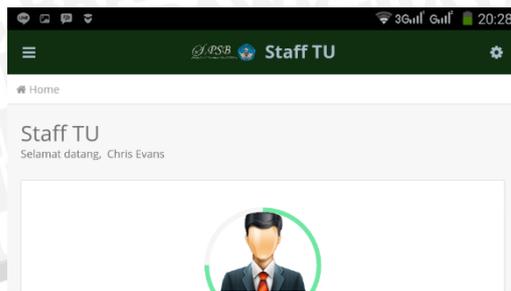
Opera 38.0



iOS 9.3.3

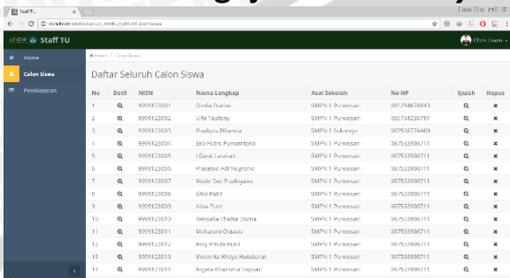


Android 5.1.1

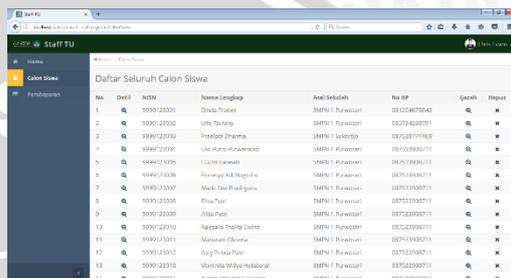


Android 4.2.2

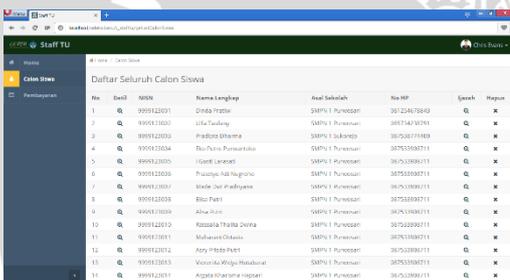
**E.34 Hasil Pengujian Portability Halaman Daftar Seluruh Calon Siswa**



Chrome 53



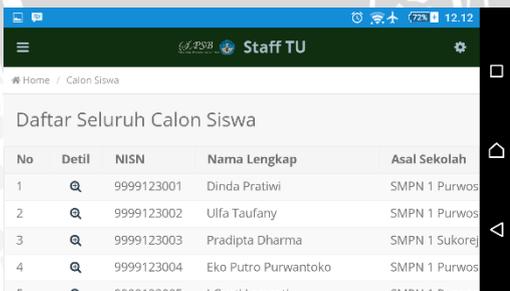
Firefox 47.0.1



Opera 38.0



iOS 9.3.3

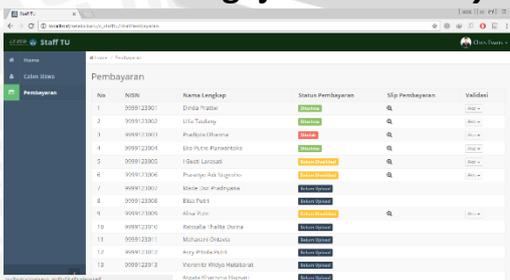


Android 5.1.1

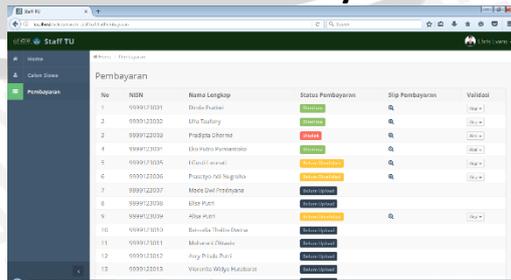


Android 4.2.2

**E.35 Hasil Pengujian Portability Halaman Validasi Pembayaran**



Chrome 53



Firefox 47.0.1

No	NISN	Nama Lengkap	Status Pembayaran	Slip Pembayaran	Validasi
1	9999123001	Dinda Pratiwi	Diterima		
2	9999123002	Ulfa Taufany	Ditolak		
3	9999123003	Pradipta Dharma	Ditolak		
4	9999123004	Ilia Nur Permawati	Ditolak		
5	9999123005	Iqbal Lansari	Ditolak		
6	9999123006	Prasetya Adi Nugroho	Ditolak		
7	9999123007	Maula Devi Pradiyana	Ditolak		
8	9999123008	Elisa Putri	Ditolak		
9	9999123009	Alfa Rizki	Ditolak		
10	9999123010	Babakha Thalia Oema	Ditolak		
11	9999123011	Mohazri Ochiada	Ditolak		
12	9999123012	Ary Prada Purca	Ditolak		
13	9999123013	Vicentia Widya Hastabari	Ditolak		
14	9999123014	Agatha Wilhelmina Hapsari	Ditolak		

Opera 38.0

No	NISN	Nama Lengkap	Status
1	9999123001	Dinda Pratiwi	Diterima
2	9999123002	Ulfa Taufany	Ditolak
3	9999123003	Pradipta Dharma	Ditolak

iOS 9.3.3

No	NISN	Nama Lengkap	Status Pembayaran
1	9999123001	Dinda Pratiwi	Diterima
2	9999123002	Ulfa Taufany	Ditolak
3	9999123003	Pradipta Dharma	Ditolak
4	9999123004	Eko Putro Purwantoko	Diterima

Android 5.1.1

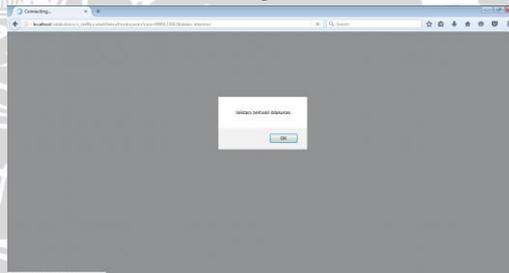
No	NISN	Nama Lengkap	Status Pembayaran
1	9999123001	Dinda Pratiwi	Diterima
2	9999123002	Ulfa Taufany	Diterima
3	9999123003	Pradipta Dharma	Ditolak

Android 4.2.2

### E.36 Hasil Pengujian Portability Ubah Status Pembayaran

No	NISN	Nama Lengkap	Status Pembayaran	Slip Pembayaran	Validasi
1	9999123001	Dinda Pratiwi	Diterima		
2	9999123002	Ulfa Taufany	Ditolak		
3	9999123003	Pradipta Dharma	Ditolak		
4	9999123004	Ilia Nur Permawati	Ditolak		
5	9999123005	Iqbal Lansari	Ditolak		
6	9999123006	Prasetya Adi Nugroho	Ditolak		
7	9999123007	Maula Devi Pradiyana	Ditolak		
8	9999123008	Elisa Putri	Ditolak		
9	9999123009	Alfa Rizki	Ditolak		
10	9999123010	Babakha Thalia Oema	Ditolak		
11	9999123011	Mohazri Ochiada	Ditolak		
12	9999123012	Ary Prada Purca	Ditolak		
13	9999123013	Vicentia Widya Hastabari	Ditolak		
14	9999123014	Agatha Wilhelmina Hapsari	Ditolak		

Chrome 53



Firefox 47.0.1

No	NISN	Nama Lengkap	Status Pembayaran	Slip Pembayaran	Validasi
1	9999123001	Dinda Pratiwi	Diterima		
2	9999123002	Ulfa Taufany	Ditolak		
3	9999123003	Pradipta Dharma	Ditolak		
4	9999123004	Ilia Nur Permawati	Ditolak		
5	9999123005	Iqbal Lansari	Ditolak		
6	9999123006	Prasetya Adi Nugroho	Ditolak		
7	9999123007	Maula Devi Pradiyana	Ditolak		
8	9999123008	Elisa Putri	Ditolak		
9	9999123009	Alfa Rizki	Ditolak		
10	9999123010	Babakha Thalia Oema	Ditolak		
11	9999123011	Mohazri Ochiada	Ditolak		
12	9999123012	Ary Prada Purca	Ditolak		
13	9999123013	Vicentia Widya Hastabari	Ditolak		
14	9999123014	Agatha Wilhelmina Hapsari	Ditolak		

Opera 38.0

## LAMPIRAN F DOKUMENTASI

### F.1 Pengujian Functional Suitability



### F.2 Pengujian Usability



