

**PENGEMBANGAN *SOFTWARE TES DENGAN ANALISIS BUTIR SOAL*
BERBASIS *WEB* DI MAN YOGYAKARTA 1**

TUGAS AKHIR SKRIPSI

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



Oleh:

Muhammad Thoriq Romadhon

NIM. 10520244005

**PROGRAM STUDI PENDIDIKAN TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS NEGERI YOGYAKARTA
2015**

LEMBAR PERSETUJUAN

Tugas Akhir Skripsi dengan Judul

PENGEMBANGAN DAN ANALISIS KUALITAS SOFTWARE TES DENGAN ANALISIS BUTIR SOAL BERBASIS WEB DI MAN YOGYAKARTA 1

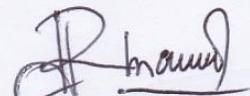
Disusun oleh:

Muhammad Thoriq Romadhon
NIM 10520244005

telah memenuhi syarat dan disetujui oleh Dosen Pembimbing untuk dilaksanakan

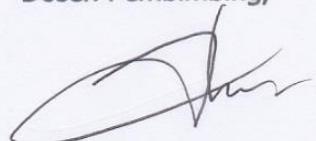
Ujian Akhir Tugas Akhir Skripsi bagi yang bersangkutan.

Mengetahui,
Ketua Program Studi
Pendidikan Teknik Informatika,



Dr. Ratna Wardani
NIP. 19701218 200501 2 001

Yogyakarta, 3 Desember 2014
Disetujui,
Dosen Pembimbing,



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

SURAT PERNYATAAN

Saya yang bertanda tangan di bawah ini:

Nama : Muhammad Thoriq Romadhon

NIM : 10520244005

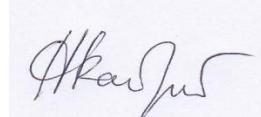
Program Studi: Pendidikan Teknik Informatika

Judul TAS : Pengembangan *Software Tes dengan Analisis Butir Soal Berbasis Web* di MAN Yogyakarta 1

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

Yogyakarta, 3 Desember 2014

Yang menyatakan,



Muhammad Thoriq Romadhon
NIM. 10520244005

HALAMAN PENGESAHAN

Tugas Akhir Skripsi

**PENGEMBANGAN SOFTWARE ESES DENGAN ANALISIS BUTIR SOAL
BERBASIS WEB DI MAN YOGYAKARTA 1**

Disusun oleh:

Muhammad Thoriq Romadhon
NIM 10520244005

Telah dipertahankan di depan Tim Penguji Tugas Akhir Skripsi Program Studi
Pendidikan Teknik Informatika Fakultas Teknik Universitas Negeri Yogyakarta
Pada tanggal 5 Januari 2015

TIM PENGUJI

Nama/Jabatan

Handaru Jati, Ph.D.
Ketua Penguji/Pembimbing

Tanda Tangan



Tanggal

12/1/2015

Athika Dwi Wiji Utami, M.Pd.
Sekretaris

12/1/2015

Herman Dwi Surjono, Ph.D.
Penguji

8/1/2015

Yogyakarta, Januari 2015

Fakultas Teknik Universitas Negeri Yogyakarta

Dekan,



HALAMAN MOTTO

"Sukses adalah kemampuan untuk pergi dari kegagalan tanpa kehilangan semangat" – Sir Winston Churchill

"Sesungguhnya Allah tidak akan mengubah nasib suatu kaum hingga mereka mengubah diri mereka sendiri" – Q.S Ar-Ra'd:11

"Jika saya diberi waktu delapan jam untuk menebang pohon, saya akan menggunakan yang enam jam untuk mengasah kapak" – Abraham Lincoln

"Jika Anda melakukan sesuatu dan ternyata menghasilkan hal yang mengagumkan, maka Anda harus mencoba melakukan hal luar biasa lainnya, jangan terpaku terlalu lama. Temukan saja apa hal selanjutnya." – Steve Jobs

"Life was never meant to be easy. The trials of life were meant to separate the earnest from lazy and dishonest." – Mario Teguh

HALAMAN PERSEMBAHAN

Skripsi ini saya persembahkan untuk :

1. Kedua orangtua saya yang tercinta, Bapak Alm. Munadjat dan Ibu Temu Widati yang selalu mendidik saya, memberikan doa, dukungan, nasihat dan semangat yang tiada henti.
2. Semua kakak saya, Yan Rusdi Kamal, Hamdan Amin, Farah Susilowati, dan Muhammad Sholeh Dipraja yang turut serta memberi doa dan dukungan.
3. Seluruh keluarga Bani Marsudi yang telah memberikan do'aanya.
4. Dayan Ramly Ramadhan, Mirza Hikmatyar, Yanuar Arifin, Damar Purba Pamungkas, dan Rama Bramantara dalam tim Craterio terimakasih untuk selalu memberikan doa, dukungan dan bantuan dalam menyelesaikan skripsi ini.
5. Ilham Prabowo, Galih Nugroho, Fuat Hermawan, Nuning Arumsari, Refany Anhar, Abdul Rachman Pambudi, Pradana Setialana, Tika Novita Sari, Neutrina Nilamsari, Lisa Noor Ardhiani, Galih Amindyah D.P., Aditia Nurjaman, Aprianto Dwi Atmaji, dan Wahyu Andi Saputra yang telah membantu, memberikan motivasi serta do'a untuk menyelesaikannya skripsi ini.
6. Keluarga Informatika F 2010 UNY yang selalu memberikan keceriaan, kebersamaan dan motivasi.
7. Seluruh teman UNY yang telah memberikan do'a, dukungan, dan semangat.

PENGEMBANGAN *SOFTWARE* TES DENGAN ANALISIS BUTIR SOAL BERBASIS *WEB* DI MAN YOGYAKARTA 1

Oleh:

Muhammad Thoriq Romadhon
NIM 10520244005

ABSTRAK

Tes dengan menggunakan kertas dapat memakan banyak biaya dan waktu karena soal perlu dicetak untuk seluruh siswa dan jawaban tes siswa harus dikoreksi secara manual oleh guru. Pemanfaatan komputer untuk tes bisa menghemat biaya dan waktu karena soal tidak perlu dicetak dan jawaban siswa dikoreksi oleh komputer. Penelitian ini bertujuan untuk: (1) merancang dan membuat *software* tes dengan analisis butir soal berbasis *web* di MAN Yogyakarta 1, dan (2) mengetahui kualitas *software* yang dikembangkan pada *software* tes dengan analisis butir soal berbasis *web* berdasarkan standar ISO 9126 pada aspek *functionality, reliability, usability, efficiency, maintainability, dan portability*.

Metode yang digunakan adalah *Research and Development* (R&D). Prosedur pengembangan *software* menggunakan model *waterfall*. Model *waterfall* memiliki tahapan yang terdiri dari analisis kebutuhan, desain, implementasi, dan pengujian.

Hasil penelitian diketahui bahwa: (1) *software* tes dengan analisis butir soal berbasis *web* dikembangkan dengan menggunakan *framework* Laravel dan tahap pengembangannya terdiri dari tahap analisis kebutuhan, tahap desain, tahap implementasi, dan tahap pengujian, dan (2) hasil pengujian aspek *functionality, fungsi berjalan 100%* dan tidak memiliki celah terhadap serangan *SQL Injection* dan *XSS (Cross Site Scripting)*. Dalam aspek *reliability*, saat diuji dengan *stress testing*, tidak terdapat *error* dan 100% *software* dapat berjalan dengan baik. Pada aspek *usability*, tingkat persetujuan pengguna sebesar 73% dan nilai *cronbach's alpha* sebesar 0,934 dengan kategori *excellent*. Pada aspek *efficiency*, waktu *load time* rata-rata sebesar 1,65 detik (diterima). Pada aspek *maintainability*, *software* berada pada tingkat mudah dalam perbaikan. Pada aspek *portability*, *software* berhasil diakses melalui semua *browser desktop* dan *mobile* yang diujicobakan tanpa *error*.

Kata Kunci : *Software, tes, analisis butir soal, web, dan ISO 9126.*

KATA PENGANTAR

Puji syukur kepada Allah SWT atas rahmat, hidayah, dan karunia-Nya kepada kita semua sehingga kami dapat menyelesaikan Tugas Akhir Skripsi dengan judul "Pengembangan *Software* Tes dengan Analisis Butir Soal Berbasis *Web* di MAN Yogyakarta 1". Tugas Akhir Skripsi ini disusun sebagai salah satu syarat untuk mendapatkan gelar Sarjana Pendidikan di Fakultas Teknik Universitas Negeri Yogyakarta.

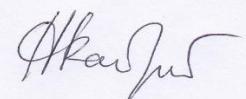
Penulis menyadari dalam penyusunan proposal skripsi ini tidak akan selesai tanpa bantuan dari berbagai pihak. Oleh karena itu pada kesempatan ini kami ingin mengucapkan terima kasih kepada yang terhormat:

1. Handaru Jati, Ph.D selaku Dosen Pembimbing TAS yang telah banyak membantu selama penyusunan Tugas Akhir Skripsi ini.
2. Muhammad Munir, M.Pd dan Dr. Ratna Wardani selaku Ketua Jurusan Pendidikan Teknik Elektronika dan Ketua Program Studi Pendidikan Teknik Informatika beserta dosen dan staf yang telah memberikan bantuan dan fasilitas selama proses penyusunan pra proposal sampai dengan selesaiya TAS ini.
3. Dr. Moch. Bruri Triyono selaku Dekan Fakultas Teknik Universitas Negeri Yogyakarta yang memberikan persetujuan pelaksanaan Tugas Akhir Skripsi.
4. Drs. H. Imam Suja'i Fadly, M. Pd. I selaku Kepala MAN Yogyakarta 1 yang telah memberi ijin dan bantuan dalam pelaksanaan penelitian Tugas Akhir Skripsi ini.

5. Para guru dan siswa MAN Yogyakarta 1 yang telah memberi bantuan memperlancar pengambilan data selama proses penelitian Tugas Akhir Skripsi ini.
6. Semua pihak, secara langsung maupun tidak langsung, yang tidak dapat disebutkan di sini atas bantuan dan perhatiannya selama penyusunan Tugas Akhir Skripsi ini.

Akhirnya, semoga segala bantuan yang telah berikan semua pihak di atas menjadi amalan yang bermanfaat dan mendapatkan balasan dari Allah SWT dan Tugas Akhir Skripsi ini menjadi informasi bermanfaat bagi pembaca atau pihak lain yang membutuhkannya.

Yogyakarta, Desember 2014
Penulis,



Muhammad Thoriq Romadhon
NIM 10520244005

DAFTAR ISI

	Halaman
LEMBAR PERSETUJUAN	ii
SURAT PERNYATAAN	iii
HALAMAN PENGESAHAN	iv
HALAMAN MOTTO	iv
HALAMAN PERSEMBAHAN	vi
ABSTRAK	vii
KATA PENGANTAR	viii
DAFTAR ISI	x
DAFTAR TABEL	xii
DAFTAR GAMBAR	xiii
DAFTAR LAMPIRAN	xv
BAB I PENDAHULUAN	1
A. Latar Belakang Masalah	1
B. Identifikasi Masalah	3
C. Batasan Masalah	3
D. Rumusan Masalah	4
E. Tujuan Penelitian	4
F. Spesifikasi Produk yang Dikembangkan	4
G. Manfaat Penelitian	5
BAB II KAJIAN TEORI	6
A. Kajian Teori	6
1. Definisi <i>Software</i>	6
2. Pengertian Tes	6
3. Analisis Butir Soal	7
4. Framework Laravel	12
5. Model Pengembangan Perangkat Lunak	16
6. Analisis Kualitas Perangkat Lunak	18
B. Hasil Penelitian yang Relevan	27
C. Kerangka Pikir	28
D. Pertanyaan Penelitian	30
BAB III METODE PENELITIAN	31
A. Model Pengembangan	31
B. Prosedur Pengembangan	31
1. Analisis Kebutuhan	31
2. Desain	32
3. Implementasi	32
4. Pengujian	33
C. Sumber Data/ Subjek Penelitian	33
D. Metode dan Alat Pengumpul Data	34
1. Aspek <i>Functionality</i>	34

2. Aspek <i>Reliability</i>	36
3. Aspek <i>Usability</i>	36
4. Aspek <i>Efficiency</i>	38
5. Aspek <i>Maintainability</i>	38
6. Aspek <i>Portability</i>	38
E. Teknik Analisis Data.....	38
1. <i>Functionality</i>	38
2. <i>Reliability</i>	39
3. <i>Usability</i>	39
4. <i>Efficiency</i>	41
5. <i>Maintainability</i>	42
6. <i>Portability</i>	42
BAB IV HASIL PENELITIAN DAN PEMBAHASAN	43
A. Tahap Analisis Kebutuhan	43
B. Tahap Desain.....	44
C. Implementasi	55
D. Pengujian	58
E. Pembahasan Hasil Penelitian	71
BAB V SIMPULAN DAN SARAN	75
A. Simpulan	75
B. Keterbatasan Produk.....	76
C. Pengembangan Produk Lebih Lanjut	76
D. Saran	76
DAFTAR PUSTAKA	77
LAMPIRAN.....	81

DAFTAR TABEL

	Halaman
Tabel 1. Indeks Kesulitan Soal	9
Tabel 2. Indeks Daya Beda Soal (Sudijono, 1996).....	11
Tabel 3. Subkarakteristik ISO 9126 dan Penjelasannya (Padayachee, Kotze, & van Der Merwe, 2010).....	19
Tabel 4. <i>Website Performance Benchmark</i> . (Optimize, 2010)	24
Tabel 5. <i>Browser Statistic</i> (W3schools, 2014)	27
Tabel 6. Instrumen <i>functionality</i>	34
Tabel 7. Kuesioner USE	36
Tabel 8. <i>Internal Consistency Cronbach's Alpha</i> (Gliem & Gliem, 2003).....	41
Tabel 9. Definisi Aktor	44
Tabel 10. Hasil Pengujian <i>Functionality</i>	59
Tabel 11. Hasil Pengujian <i>Usability</i>	64
Tabel 12. Hasil Pengujian <i>Efficiency</i>	69
Tabel 13. Persentase Tingkat Keberhasilan Pengujian <i>Reliability</i>	72
Tabel 14. Perolehan Skor Kuesioner	72
Tabel 15. Definisi <i>Use Case</i>	89
Tabel 16. Skenario <i>Use Case Login</i>	90
Tabel 17. Skenario <i>Use Case Logout</i>	90
Tabel 18. Skenario <i>Use Case Ganti username dan password</i>	91
Tabel 19. Skenario <i>Use Case Memeriksa Status Login</i>	91
Tabel 20. Skenario <i>Use Case Reset Akun</i>	92
Tabel 21. Skenario <i>Use Case Tambah Data</i>	92
Tabel 22. Skenario <i>Use Case Edit Data</i>	93
Tabel 23. Skenario <i>Use Case Hapus Data</i>	94
Tabel 24. Lihat Data	94
Tabel 25. Skenario <i>Use Case Duplikat Tes</i>	95
Tabel 26. Skenario <i>Use Case Lihat Analisis Butir Soal</i>	95
Tabel 27. Skenario <i>Use Case Memilih Tes</i>	96
Tabel 28. Skenario <i>Use Case Mengisi Identitas</i>	96
Tabel 29. Skenario <i>Use Case Melakukan Tes</i>	96
Tabel 30. Skenario <i>Use Case Melihat Hasil Tes</i>	96
Tabel 31. Daftar Responden Pengujian <i>Usability</i>	133
Tabel 32. Data Pengujian <i>Usability</i>	134

DAFTAR GAMBAR

	Halaman
Gambar 1. Interaksi MVC Laravel	14
Gambar 2. Contoh <i>Item Analysis</i> pada Moodle (sumber: docs.moodle.org)	16
Gambar 3. Model ISO 9126 (Chua & Dyson, 2004)	19
Gambar 4. Kerangka Pikir	29
Gambar 5. Ilustrasi Model Waterfall.....	31
Gambar 6. Contoh Tingkat Persetujuan Responden pada <i>Software</i>	41
Gambar 7. <i>Use Case Diagram Software</i> Tes dengan Analisis Butir Soal	45
Gambar 8. <i>Class Diagram Software</i> Tes dengan Analisis Butir Soal	46
Gambar 9. <i>Activity Diagram Software</i> Tes dengan Analisis Butir Soal.....	47
Gambar 10. <i>Sequence Diagram</i> Memilih Tes	48
Gambar 11. <i>Sequence Diagram</i> Mengisi Identitas	49
Gambar 12. <i>Sequence Diagram</i> Melaksanakan Tes	50
Gambar 13. <i>Sequence Diagram</i> Melihat Hasil Tes	51
Gambar 14. Desain Halaman Peserta Tes Bagian Daftar Tes.....	51
Gambar 15. Desain Halaman Peserta Tes Bagian Identitas Peserta Tes	52
Gambar 16. Desain Halaman Peserta Tes Bagian Pelaksanaan Tes.....	53
Gambar 17. Desain Halaman Peserta Tes Bagian Hasil Tes.....	53
Gambar 18. Desain <i>ERD Software</i> Tes dengan Analisis Butir Soal.....	54
Gambar 19. Halaman Daftar Tes	55
Gambar 20. Halaman Identitas Peserta Tes.....	56
Gambar 21. Halaman Pelaksanaan Tes	57
Gambar 22. Halaman Hasil Tes	58
Gambar 23. Contoh Hasil Analisis Butir Soal	60
Gambar 24. Perhitungan Tingkat Kesulitan dan Daya Beda Manual	61
Gambar 25. Perhitungan Persentase Pilihan Jawaban dengan <i>microsoft excel</i> ...61	61
Gambar 26. Hasil <i>Scan</i> Serangan <i>SQL Injection</i>	62
Gambar 27. Hasil <i>Scan</i> Serangan <i>XSS</i>	62
Gambar 28. <i>Retesting Vulnerabilities</i>	63
Gambar 29. Hasil Pengujian <i>Reliability</i> WAPT	63
Gambar 30. Hasil Perhitungan <i>Cronbach's Alpha</i>	65
Gambar 31. Laporan GTMetrix Halaman <i>Login</i>	65
Gambar 32. Laporan GTMetrix Halaman Pengelolaan Tes	66
Gambar 33. Laporan GTMetrix Halaman <i>Form</i> Tes	67
Gambar 34. Laporan GTMetrix Halaman Pengelolaan soal	67
Gambar 35. Laporan GTMetrix Halaman <i>Form</i> Soal	68
Gambar 36. <i>PhpMetrics Report – Maintainability Index</i>	69
Gambar 37. Laporan BrowseEmAll pada Pengujian <i>Portability Desktop</i>	70
Gambar 38. Laporan BrowseEmAll pada Pengujian <i>Portability Mobile</i>	71

Gambar 39. Tingkat Persetujuan Responden pada <i>Software</i>	73
Gambar 40. <i>Sequence Diagram Login</i>	98
Gambar 41. <i>Sequence Diagram Lihat Data</i>	98
Gambar 42. <i>Sequence Diagram Tambah Data</i>	99
Gambar 43. <i>Sequence Diagram Edit Data</i>	100
Gambar 44. <i>Sequence Diagram Hapus Data</i>	100
Gambar 45. <i>Sequence Diagram Duplikat Tes</i>	101
Gambar 46. <i>Sequence Diagram Lihat Analisis Butir Soal</i>	101
Gambar 47. <i>Sequence Diagram Mengganti Username dan Password</i>	102
Gambar 48. <i>Sequence Diagram Logout</i>	102
Gambar 49. <i>Sequence Diagram Reset Akun</i>	103
Gambar 50. Desain Halaman Guru Bagian <i>Login</i>	105
Gambar 51. Desain Halaman Guru Bagian Tes	105
Gambar 52. Desain Halaman Guru Bagian <i>Form Tes</i>	106
Gambar 53. Desain Halaman Bagian Guru Soal	106
Gambar 54. Desain Halaman Guru Bagian <i>Form Soal</i>	107
Gambar 55. Desain Halaman Guru Bagian Soal	107
Gambar 56. Desain Halaman Guru Bagian Hasil Tes.....	108
Gambar 57. Desain Halaman Guru Bagian Detai Hasil Tes	108
Gambar 58. Desain Halaman Guru Bagian Analisis Butir Soal	109
Gambar 59. Desain Halaman Guru Bagian Pengaturan Akun.....	109
Gambar 60. Desain Halaman Guru Bagian Menu.....	110
Gambar 61. Halaman Guru Bagian <i>Login</i>	112
Gambar 62. Halaman Guru Bagian Pengelolaan Tes.....	112
Gambar 63. Halaman Guru Bagian Tambah Tes	113
Gambar 64. Halaman Guru Bagian Edit Tes.....	114
Gambar 65. Halaman Guru Bagian Pengelolaan Soal.....	115
Gambar 66. Halaman Guru Bagian Tambah Soal	116
Gambar 67. Halaman Guru Bagian Edit Soal.....	117
Gambar 68. Halaman Guru Bagian Halaman Laporan.....	118
Gambar 69. Halaman Guru Bagian Hasil Tes	119
Gambar 70. Halaman Guru Bagian Detail Hasil Tes	120
Gambar 71. Halaman Guru Bagian Analisis Butir Soal 1	121
Gambar 72. Halaman Guru Bagian Analisis Butir Soal 2	121
Gambar 73. Halaman Guru Bagian Pengaturan Akun	122
Gambar 74. Dokumentasi Pengujian <i>Usability</i> 1	141
Gambar 75. Dokumentasi Pengujian <i>Usability</i> 2	141
Gambar 76. Dokumentasi Pengujian <i>Usability</i> 3.....	142
Gambar 77. Dokumentasi Pengujian <i>Usability</i> 4.....	142

DAFTAR LAMPIRAN

	Halaman
Lampiran 1. Surat – surat Perijinan	82
Lampiran 2. Desain <i>Use Case Diagram</i>	88
Lampiran 3. Desain <i>Sequence Diagram</i>	97
Lampiran 4. Desain Antarmuka	104
Lampiran 5. Implementasi Antarmuka	111
Lampiran 6. Analisis Kebutuhan	123
Lampiran 7. Angket Pengujian <i>Usability</i>	129
Lampiran 8. Data Pengujian <i>Usability</i>	132
Lampiran 9. Angket Pengujian <i>Functionality</i>	136
Lampiran 10. Dokumentasi	140

BAB I

PENDAHULUAN

A. Latar Belakang Masalah

Evaluasi pembelajaran merupakan kegiatan yang dilakukan oleh pendidik untuk untuk memantau proses, kemajuan, dan perbaikan hasil belajar peserta didik. Penjelasan tentang evaluasi pembelajaran ini tertulis pada undang-undang Republik Indonesia nomor 20 tahun 2003 tentang sistem pendidikan nasional bab XVI pasal 58 ayat 1. Dalam evaluasi pembelajaran, tes atau ujian digunakan memantau proses, kemajuan, dan perbaikan hasil belajar peserta didik.

Dalam pelaksanaan tes tertulis, pendidik dapat menggunakan kertas untuk menyajikan soal dan sebagai tempat jawaban siswa. Selain kertas, komputer juga dapat digunakan sebagai alternatif untuk melaksanakan tes. Guru di MAN (Madrasah Aliyah Negeri) Yogyakarta 1 melaksanakan tes guna mengetahui hasil belajar peserta didiknya. Berdasarkan hasil observasi yang dilakukan di MAN Yogyakarta 1 diperoleh bahwa guru MAN Yogyakarta 1 dalam melaksanakan tes masih kurang memanfaatkan komputer. Guru MAN Yogyakarta 1 masih menggunakan kertas untuk melakukan tes. Tes yang dilaksanakan menggunakan kertas ini memiliki kekurangan karena saat pembuatan soal-soal untuk tes, guru membuat soal terlebih dahulu kemudian mencetak dan menggandakannya sehingga dapat memakan biaya dan waktu. Kemudian dalam pengoreksian jawaban, guru harus mengoreksi jawaban satu persatu jawaban siswa secara manual, kemudian merekap hasilnya. Proses tersebut akan memakan waktu yang cukup lama.

Guru MAN Yogyakarta 1 melakukan analisis butir soal untuk mengetahui butir soal apakah sudah baik. Analisis butir soal yang dilakukan oleh guru MAN Yogyakarta 1 biasanya menggunakan bantuan *software microsoft excel*. *Software microsoft excel* membantu guru dalam melakukan perhitungan analisis butir soal dan mempermudah ketika hasilnya akan dicetak. Namun, *software* ini masih memiliki kekurangan yakni dalam melakukan analisis butir soal, guru perlu melakukan *input* data jawaban hasil ujian para peserta didik satu per satu sehingga memakan banyak waktu.

Berdasarkan hasil wawancara yang dilakukan di MAN Yogyakarta 1 dengan guru TIK yang juga menjabat sebagai Kepala Lab. Komputer yaitu Nung Indarti, S.Kom., diperoleh bahwa tipe soal yang sering digunakan untuk tes berupa pilihan ganda, uraian, dan kadang menggunakan tipe soal menjodohkan. Beliau menyarankan agar pengembangan *software* tes dengan analisis butir soal berbasis *web* ini diutamakan pada tipe soal pilihan ganda. Menurut beliau tipe soal pilihan ganda akan mudah dikoreksi oleh komputer karena jawabannya pasti. Tidak seperti soal *essay* yang jawabannya bisa bervariasi. *Software* tes dengan analisis butir soal berbasis *web* menurutnya diperlukan agar dapat mempercepat kinerja guru dalam memperoleh hasil tes dan melakukan analisis butir soal. Karena hasil tes dan hasil analisis butir soal dapat diperoleh secara langsung setelah siswa selesai melaksanakan tes. Selain itu menurut ibu Nung, *software* juga akan dimanfaatkan untuk memberikan layanan *try out* dari *website* sekolah yang bisa diakses oleh para siswa dari dalam sekolah maupun luar sekolah dan untuk evaluasi pelatihan guru di sekolah.

Pengembangan *software* perlu diuji agar *software* tersebut baik saat digunakan. Menurut Iacob & Constantinescu (2008), pengujian pada *software* diperlukan agar mengurangi resiko masalah sebelum *software* dirilis. Ada berbagai macam standar pengujian diantaranya *McCall*, *Boehm*, *FURPS*, *Dromey*, *The Bayesian Belief Network (BBN)*, dan ISO 9126. Dari semua standar tersebut ISO 9126 merupakan standar Internasional. ISO 9126 memiliki enam karakteristik dalam pengujian kualitas *software*. Enam karakteristik tersebut adalah *functionality*, *reliability*, *usability*, *efficiency*, *maintainability*, dan *portability*.

B. Identifikasi Masalah

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

1. Tes masih menggunakan kertas sehingga banyak memakan biaya dan waktu.
2. Tes yang menggunakan kertas masih perlu dikoreksi satu per satu secara manual yang dapat menyita waktu.
3. Analisis butir soal memerlukan *input* data secara manual sehingga rawan terjadi kekeliruan dalam memasukkan data (*human error*).
4. MAN Yogyakarta 1 belum memiliki *software* tes dengan analisis butir soal berbasis *web* yang memenuhi standar kualitas.

C. Batasan Masalah

1. *Software* tes dengan analisis butir soal yang dikembangkan di MAN Yogyakarta 1 merupakan *software* yang berbasis *web*.
2. Analisis Kualitas *software* tes dengan analisis butir soal di MAN Yogyakarta 1 berdasarkan standar kualitas.

D. Rumusan Masalah

Berdasarkan uraian di atas maka dalam penelitian ini, masalah dapat dirumuskan sebagai berikut:

1. Bagaimana mengembangkan *software* tes dengan analisis butir soal berbasis *web* di MAN Yogyakarta 1?
2. Bagaimana tingkat kualitas *software* yang dikembangkan?

E. Tujuan Penelitian

Tujuan penelitian pengembangan *software* tes dengan analisis butir soal berbasis *web* ini adalah:

1. Untuk merancang *software* tes dengan analisis butir soal berbasis *web* di MAN Yogyakarta 1.
2. Untuk mengetahui kualitas *software* yang dikembangkan berdasarkan standar kualitas *software* ISO 9126.

F. Spesifikasi Produk yang Dikembangkan

Produk yang dikembangkan dari penelitian ini adalah *software* tes dengan analisis butir soal berbasis *web* di MAN Yogyakarta 1 dengan fungsi sebagai berikut:

1. Melihat data
2. Memasukkan data
3. Mengubah data
4. Menghapus data
5. Mengunduh data

G. Manfaat Penelitian

Pengembangan *software* tes dengan analisis butir soal berbasis *web* di MAN Yogyakarta 1 ini diharapkan dapat bermanfaat bagi beberapa pihak antara lain:

1. Guru
 - a. Guru dapat menggunakan *software* untuk membuat tes dan soalnya.
 - b. Guru dapat menggunakan *software* untuk menyelenggarakan tes.
 - c. Menghemat biaya guru karena tidak menggunakan kertas untuk menyajikan soal.
 - d. Menghemat waktu guru karena hasil tes dan analisis butir soal bisa didapatkan secara langsung setelah siswa mengerjakan tes.
2. Siswa
 - a. Siswa dapat menggunakan *software* untuk mengerjakan tes.
 - b. Siswa dapat mengetahui hasil tes secara cepat (langsung setelah selesai mengerjakan tes).
3. Peneliti
 - a. Mengenal dan memahami lebih jauh teknologi pengembangan *software*.
 - b. Mengetahui teknik merancang *software* tes dengan analisis butir soal.
 - c. Mengetahui teknik pengujian kualitas *software*.

BAB II

KAJIAN TEORI

A. Kajian Teori

1. Definisi *Software*

Software dalam bahasa Indonesia sering disebut dengan perangkat lunak. Menurut Rosa & Shalahuddin (2013), "Perangkat lunak (*software*) adalah program komputer yang terasosiasi dengan dokumentasi perangkat lunak seperti dokumentasi kebutuhan, model desain, dan cara penggunaan (*user manual*)". Roger (2010) menyatakan bahwa:

"Software is: (1) instructions (computer programs) that when executed provide desired features, function, and performance; (2) data structures that enable the programs to adequately manipulate information, and (3) descriptive information in both hard copy and virtual forms that describes the operation and use of the programs."

Berdasarkan pendapat para ahli di atas, pengertian *software* yaitu program komputer yang menyediakan fitur, fungsi, dan pelaksanaan struktur data yang memungkinkan program untuk memanipulasi informasi. Program tersebut juga terasosiasi dengan dokumentasi perangkat lunak.

2. Pengertian Tes

Dalam kegiatan belajar mengajar, tes sering digunakan untuk kegiatan evaluasi peserta didik. Tes dalam dunia pendidikan sering disebut dengan ujian. Ada berbagai macam pengertian tes menurut beberapa ahli. Arikunto (1999) menjelaskan bahwa "Tes adalah alat atau prosedur yang digunakan untuk mengetahui atau mengukur sesuatu dalam suasana, dengan cara dan aturan-aturan yang sudah ditentukan". Ahli lain yaitu Sudijono (1996) mengemukakan

pengetian tes dalam uraian yang lebih panjang. Sudijono mendefinisikan pengertian tes sebagai berikut:

"tes adalah cara (yang dapat dipergunakan) atau prosedur (yang perlu ditempuh) dalam rangka pengukuran dan penilaian di bidang pendidikan, yang berbentuk pemberian tugas atau serangkaian tugas baik berupa pertanyaan-pertanyaan (yang harus dijawab), atau perintah-perintah (yang harus dikerjakan) oleh *testee*, sehingga (atas dasar data yang diperoleh dari hasil pengukuran tersebut) dapat dihasilkan nilai yang melambangkan tingkah laku atau prestasi *testee*; nilai mana dapat dibandingkan dengan nilai-nilai yang dicapai oleh *testee* lainnya atau dibandingkan dengan nilai standar tertentu."

Overton (2012) mengemukakan tentang pengertian tes "*A method to determine a student's ability to complete certain tasks or demonstrate mastery of a skill or knowledge of content*". Jadi, pengertian tes berdasarkan pendapat para ahli di atas yaitu sebuah metode, cara atau alat yang berupa serangkaian tugas yang harus dikerjakan oleh peserta tes agar dapat menghasilkan nilai yang menggambarkan prestasi atau kemampuan peserta tes.

3. Analisis Butir Soal

Dalam melaksanakan tes, butir soal yang digunakan untuk tes dapat dikategorikan apakah baik atau tidak dengan melakukan analisis butir soal. Menurut Daryanto (2001), "analisis butir soal adalah suatu prosedur yang sistematis, yang akan memberikan informasi-informasi yang sangat khusus terhadap butir tes yang kita susun." Daryanto menjelaskan bahwa untuk memutuskan bahwa sebuah soal dikatakan baik perlu dilakukan analisis soal mengenai taraf kesukaran, daya pembeda, dan pola jawaban soal. Ahli lain yaitu Sudijono (1996) menjelaskan bahwa analisis butir-butir *item* tes hasil belajar dapat dilakukan dari segi derajat kesukaran *item*-nya, daya pembeda *item*-nya, fungsi distraktornya. Dari beberapa pendapat ahli tersebut dapat disimpulkan bahwa analisis butir soal merupakan prosedur sistematis yang dilakukan untuk

memberikan informasi-informasi terhadap butir tes yang disusun. Informasi dari analisis butir soal tersebut adalah tingkat kesulitan, daya beda, dan pola jawaban soal.

a. Tingkat kesulitan

Tingkat kesulitan sebuah soal akan mempengaruhi mutu atau baik tidaknya soal tersebut. Soal yang baik adalah soal yang tidak terlalu mudah atau terlalu sukar (Daryanto, 2001). Tingkat kesulitan suatu soal dapat diketahui dengan menggunakan suatu perhitungan. Perhitungan sering berhubungan dengan angka atau bilangan. Bilangan yang menunjukkan tingkat kesulitan suatu soal disebut dengan indeks kesulitan. Indeks kesulitan ini memiliki nilai dari 0,0 sampai 1,0 dengan ketentuan bahwa indeks kesulitan dengan nilai 0,0 menunjukkan bahwa soal terlalu sulit dan 1,0 menunjukkan bahwa soal terlalu mudah (Arikunto, 1999). Indeks kesulitan ini tentunya dapat dihitung dari rumus. Menurut Sudijono (1996), Indeks kesulitan soal dapat dihitung dengan rumus berikut:

$$P = \frac{B}{JS}$$

Di mana :

P = Proporsi / Indeks kesulitan.

B = Banyaknya peserta tes yang menjawab benar terhadap suatu soal.

JS = Jumlah peserta tes.

Butir soal yang telah diketahui indeks kesulitannya tentunya sudah dapat diklasifikasikan mana soal yang sangat sulit, soal yang sangat mudah atau bisa saja soal yang sedang. Pada buku yang dikarang oleh Robert L.Thorndike dan Elizabeth Hagen berjudul *Measurement and Evaluation in Psychology and*

Education yang dikutip oleh Sudijono (1996), Indeks kesulitan suatu soal diklasifikasikan. Klasifikasi tersebut dapat dilihat pada Tabel 1.

Tabel 1. Indeks Kesulitan Soal

Indeks kesulitan soal	Klasifikasi soal
Kurang dari 0,30	Sukar
0,30 – 0,70	Sedang
Lebih dari 0,70	Mudah

Butir soal yang baik adalah soal yang tidak terlalu mudah atau terlalu sukar. Oleh karena itu soal yang memiliki indeks kesulitan soal 0,30 – 0,70 atau memiliki klasifikasi soal sedang dapat dikatakan butir soal yang baik.

b. Daya beda

Daya beda soal merupakan kemampuan suatu soal untuk membedakan antara siswa yang pandai dengan siswa yang bodoh (Daryanto, 2001). Menurut Arikunto (1999), bilangan yang menunjukkan besarnya daya beda disebut indeks diskriminasi. Indeks diskriminasi memiliki nilai dari -1,00 – 1,00. Bila soal dijawab benar oleh siswa pandai maupun siswa yang bodoh, maka soal tersebut tidak memiliki daya beda sehingga soal tersebut tidak baik dan daya bedanya bernilai 0,00. Begitu juga bila soal dijawab salah oleh siswa pandai maupun siswa bodoh. Jika Kelompok siswa pandai menjawab soal dengan benar dan kelompok siswa bodoh menjawab soal tersebut salah maka soal tersebut mempunyai daya beda 1,00. Tetapi apabila kelompok siswa pandai menjawab soal tersebut dengan salah sedangkan kelompok siswa bodoh menjawabnya dengan benar maka nilai daya beda soal adalah -1,00. Daya beda soal 0,00 berarti soal tidak memiliki daya beda sama sekali, daya beda 1,00 berarti soal memiliki daya beda paling tinggi dan daya beda -1,00 berarti soal tersebut memiliki daya beda yang jelek sekali karena siswa

yang bodoh lebih menjawab soal tersebut dengan benar dari pada siswa yang pandai.

Penentuan indeks daya beda soal adalah dengan membagi peserta tes menjadi dua kelompok, yaitu kelompok pandai (peserta tes yang memiliki nilai hasil tes tinggi) dan kelompok bodoh (peserta tes yang memiliki nilai hasil tes rendah). Setelah itu indeks daya beda soal dapat dihitung dengan menggunakan rumus:

$$D = \frac{B_A}{J_A} - \frac{B_B}{J_B} = P_A - P_B$$

Di mana :

J_A = Banyaknya peserta tes kelompok pandai

J_B = Banyaknya peserta tes kelompok bodoh

B_A = Banyaknya peserta tes kelompok pandai yang menjawab soal dengan benar

B_B = Banyaknya peserta tes kelompok bodoh yang menjawab soal dengan benar

P_A = Indeks kesukaran soal untuk kelompok pandai

P_B = Indeks kesukaran soal untuk kelompok bodoh

Setelah diketahui indeks daya bedanya, selanjutnya soal dapat diklasifikasikan baik buruknya daya beda soal tersebut. Klasifikasi indeks daya beda soal memiliki patokan seperti pada Tabel 2.

Tabel 2. Indeks Daya Beda Soal (Sudijono, 1996)

Indeks Daya Beda Soal	Klasifikasi
Kurang dari 0,20	Jelek
0,20 – 0,40	Cukup
0,40 – 0,70	Baik
0,70 – 1,00	Baik sekali
Bernilai negatif	-

Butir soal yang memiliki klasifikasi cukup, baik dan baik sekali merupakan soal yang sudah memiliki daya beda yang baik (Sudijono, 1996). Oleh karena itu soal dapat dikatakan memiliki daya beda yang baik bila indeks daya bedanya memiliki nilai 0,20 – 1,00 dan dikatakan jelek apabila memiliki nilai kurang dari 0,20 dan negatif.

c. Pola Jawaban Soal

Pola jawaban soal maksudnya adalah distribusi jawaban peserta tes pada soal bentuk pilihan ganda. Pola jawaban soal didapatkan dari perhitungan banyaknya peserta tes yang memilih jawaban a, b, c, d, e ataupun yang tidak memilih jawaban manapun atau dalam istilah evaluasi disebut dengan omit, disingkat O (Daryanto, 2001). Menurut Arikunto (1999), dari pola jawaban dapat ditentukan apakah jawaban pengecoh pada soal pilihan ganda dapat mengecoh peserta tes dengan baik atau tidak. Pengecoh yang tidak dipilih sama sekali oleh peserta tes dapat disimpulkan bahwa pengecoh tersebut jelek sedangkan pengecoh yang baik merupakan pengecoh yang memiliki daya tarik yang besar untuk dipilih oleh peserta tes. Pengecoh dapat dikatakan baik apabila paling sedikit dipilih oleh peserta tes sebesar 5% dari total peserta tes dan jika omitnya tidak lebih dari 10% dari total peserta tes.

4. *Framework* Laravel

Dalam pengembangan *web* banyak sekali alat-alat yang digunakan untuk mempermudah pengembangannya. Salah satunya adalah dalam bentuk *framework*. Menurut Björemo & Trninić (2010), "*A framework in context of software development is a set of prewritten code or libraries which provide functionality common to a whole class of applications. The framework can be seen as a base or a skeleton to build upon*". Rosa & Shalahuddin (2013) mengatakan bahwa *framework* berupa kerangka kerja yang digunakan oleh *programmer* untuk mempermudahnya dalam membuat sebuah aplikasi dan mempermudah dalam melakukan perubahan. Dapat disimpulkan bahwa *framework* merupakan kerangka kerja yang menyediakan fungsi – fungsi umum untuk *programmer* yang dapat digunakan untuk mempermudahnya dalam membuat sebuah aplikasi/program.

Menurut Björemo & Trninić (2010), keuntungan penggunaan *framework* adalah sebagai berikut:

- a. *Reuse Code*. Kode program yang telah dibuat, diuji, dan digunakan oleh *programmer* lain dapat digunakan/dikembangkan kembali sehingga dapat memangkas waktu dalam membuat program.
- b. *Get free help*. Bantuan misalnya berupa adanya fungsi keamanan bawaan dari *framework* sehingga *programmer* secara tidak langsung mendapat bantuan secara gratis.
- c. *Specific architecture or design pattern*. Membuat *programmer* mudah dalam memahami dan mulai membuat program dengan menggunakan *framework* dengan adanya konsep arsitektur pembuatan kode program pada *framework*.

- d. *Support high level programming.* Terdapat beberapa modul dasar misalnya seperti *login* ataupun *database handling* yang dapat digunakan untuk mempercepat pengembangan program.
- e. Terdapat *update* fitur dari *framework*.
- f. Mudah dipahami oleh *new inexperienced developers*.

Ada berbagai jenis *framework* yang digunakan untuk pembuatan *software* berbasis *web*. Salah satu *framework* yang digunakan untuk pengembangan *software* berbasis *web* adalah Laravel. Rees (2012) mengatakan bahwa "*Laravel is a PHP 5.3 web application framework*", berarti bahwa laravel merupakan *framework* untuk aplikasi *web* yang menggunakan bahasa pemrograman PHP. Laravel memiliki merupakan *framework* dengan *architecture pattern* yang berupa MVC atau Model *View Controller* (Armel, 2014). Menurut Armel (2014), *developer* dapat menuliskan kode program dalam tiga basis berikut dalam konsep MVC:

a. *Model*

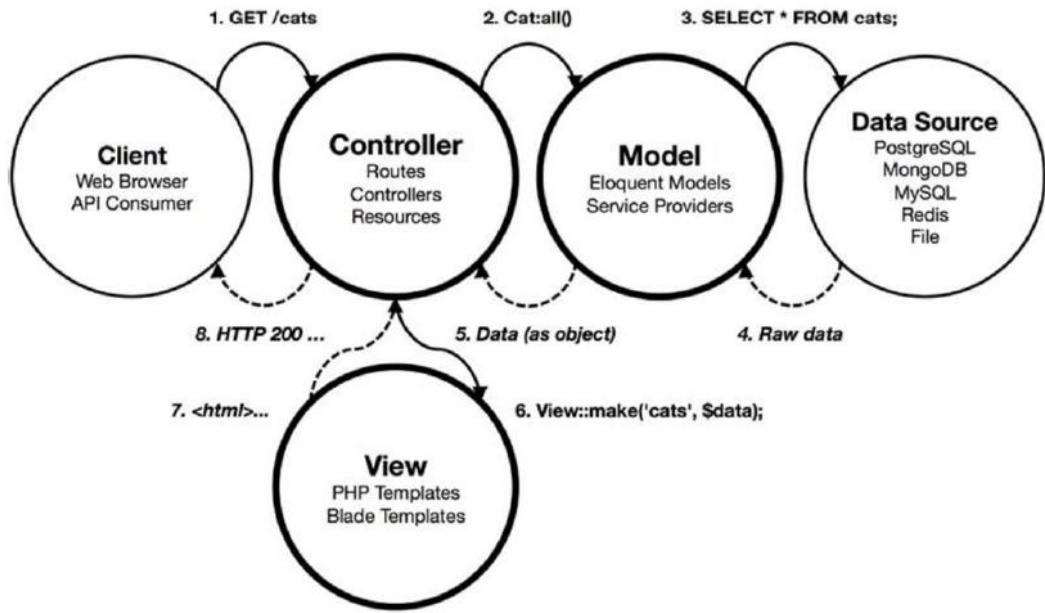
Model merupakan bagian yang dapat digunakan untuk memanipulasi data. *Model* terdiri dari layer yang terletak antara data dan aplikasi.

b. *Views*

Views merupakan gambaran visual untuk aplikasi *web*. *Views* bertanggungjawab untuk menampilkan data yang didapatkan oleh *controller* dari *model*.

c. *Controller*

Fungsi utama dari *controller* adalah menangani *requests* dan menampaikan data dari *model* menuju ke *views*. Interaksi yang terjadi pada MVC dapat dilihat pada Gambar 1.



Gambar 1. Interaksi MVC Laravel

Keunggulan laravel dari *web* menurut Otwell (2014) adalah sebagai berikut:

a. *RESTful Routing*

Menggunakan cara yang mudah untuk merespon *requests* pada aplikasi.

b. *Command Your Data*

Menggunakan *Eloquent ORM* dan sistem migrasi database yang dapat bekerja pada *MySQL*, *Postgres*, *SQL Server*, dan *SQLite*.

c. *Beautiful Templating*

Dapat menggunakan *native PHP* atau *blade template engine*

d. *Ready For Tomorrow*

Sempurna untuk semua ukuran kerja *software* seperti *enterprise applications* ataupun *simple JSON APIs*.

e. *Proven Foundation*

Dibuat dengan *Symfony components* yang mana komponen tersebut *well-tested and reliable code*.

f. *Composer Powered*

Composer merupakan alat yang digunakan untuk mengatur paket-paket pihak ketiga yang dapat diintegrasikan dengan Laravel.

g. *Great Community*

Terdapat forum dan *IRC Chat* yang dapat digunakan untuk *sharing* dan bertanya.

h. *Red, Green, Refactor*

Terdapat *unit testing* bawaan Laravel yang mudah digunakan.

Keunggulan lain menurut Surguy (2013) adalah lisensi Laravel adalah *MIT License* sehingga laravel bersifat *Open Sources* (gratis).

Beberapa contoh *framework* PHP selain laravel adalah Symfony, Zend, Slim, Kohana, Lithium, dan CodeIgniter. Menurut Surguy (2014), Laravel dikembangkan berdasarkan fungsi yang belum ada pada beberapa *framework* tersebut dan beberapa *framework* tersebut masih fokus pada pengembangan untuk versi PHP yang lama (sebelum versi PHP 5.3). Berbeda dengan Laravel yang dikembangkan untuk versi PHP terbaru.

Jadi, dapat disimpulkan bahwa Laravel dapat digunakan untuk membangun *software* berbasis *web* karena bahasa pemrograman Laravel adalah PHP. Fitur-fitur bawaan Laravel seperti *composer*, *eloquent ORM*, *routing*, dan fitur lainnya, serta sifatnya yang gratis dan adanya forum yang mendukung akan membantu dan mempermudah dalam pengembangan *software* berbasis *web*. Selain itu Laravel memiliki keunggulan yaitu Laravel dikembangkan untuk melengkapi fungsi-fungsi yang belum ada pada beberapa *framework* seperti Symfony, Zend, Slim, Kohana, Lithium, dan CodeIgniter dan dikembangkan untuk versi PHP terbaru.

5. Moodle

Moodle merupakan *Course Management System (CMS)* yang bersifat gratis. Menurut Cole & Foster (2007), moodle merupakan CMS yang sering digunakan oleh universitas, sekolah, dunia bisnis, dan juga para pengajar untuk membuat *course* berbasis *web*. Moodle memiliki banyak fitur yang dapat menunjang pembelajaran. Diantara fitur-fitur yang dimiliki moodle, fitur yang berhubungan dengan *software* tes dengan analisis butir soal adalah moodle bisa digunakan untuk melakukan kegiatan tes dan membuat soal-soal yang digunakan untuk tes. Selain itu di dalam moodle juga terdapat fitur *item analysis* (analisis butir soal). Analisis butir soal pada moodle berupa *random guess score*, *facility index*, *standar deviation*, *discrimination index*, dan *discriminative efficiency*. Gambar 2 merupakan contoh laporan *item analysis* pada moodle.

Q#	Question text	Answer's text	partial credit	R. Counts	R. %	% Correct Facility	SD	Disc. Index	Disc. Coeff.
(2)	Life, the universe and everything : According to Douglas Adams, what was the meaning of life?	1999	(0.00)	0/1	(0%)	0%	0.000	0.00	-999.00
		1942	(0.00)	1/1	(100%)				
		42	(1.00)	0/1	(0%)				
		41	(0.00)	0/1	(0%)				
		40	(0.00)	0/1	(0%)				
		24	(0.00)	0/1	(0%)				
		21	(0.00)	0/1	(0%)				

Gambar 2. Contoh *Item Analysis* pada Moodle (sumber: docs.moodle.org)

Analisis butir soal menurut Sudijono (1996) dapat diukur dengan tingkat kesulitan, daya beda, dan pola jawaban soal. Pada Gambar 2 tingkat kesulitan pada moodle ditampilkan pada kolom *% correct facility*, daya beda pada kolom *Disc. Index* dan pola jawaban soal pada kolom *R. %*.

Moodle memiliki banyak kelengkapan fitur bila digunakan untuk pembelajaran. Moodle juga sudah memiliki fitur untuk membuat dan melaksanakan tes serta fitur

untuk melakukan analisis butir soal. Meskipun moodle memiliki fitur yang lengkap dan banyak, moodle tidak dimanfaatkan di MAN Yogyakarta 1. Berdasarkan hasil wawancara dengan pihak sekolah MAN Yogyakarta 1, moodle sulit untuk digunakan karena terlalu banyak memiliki fitur dan kurang sederhana.

6. Model Pengembangan Perangkat Lunak

Model pengembangan perangkat lunak atau *software* dibutuhkan agar menghasilkan perangkat lunak yang berkualitas (Rosa & Shalahuddin 2013). Model pengembangan perangkat lunak sering disebut dengan model SDLC atau *Software Development Life Cycle*. Salah satu model SDLC ini adalah model air terjun (*waterfall*). Menurut Rosa & Shalahuddin (2013), model *waterfall* adalah model SDLC yang cocok untuk pengembangan dengan spesifikasi yang tidak berubah-ubah. Model *waterfall* memiliki alur sebagai berikut:

a. Analisis kebutuhan perangkat lunak

Berupa proses pengumpulan kebutuhan perangkat lunak agar sesuai dengan apa yang dibutuhkan oleh *user*. Analisis kebutuhan perangkat lunak ini didokumentasikan.

b. Desain

Berupa proses untuk membuat desain perangkat lunak yang terdiri dari struktur data, arsitektur perangkat lunak, representasi antar muka, dan prosedur pengkodean.

c. Pembuatan Kode Program

Pada tahap ini desain yang telah dibuat akan ditranslasikan ke dalam program perangkat lunak sehingga menghasilkan program komputer yang sesuai dengan desain yang telah dibuat pada tahap desain.

d. Pengujian

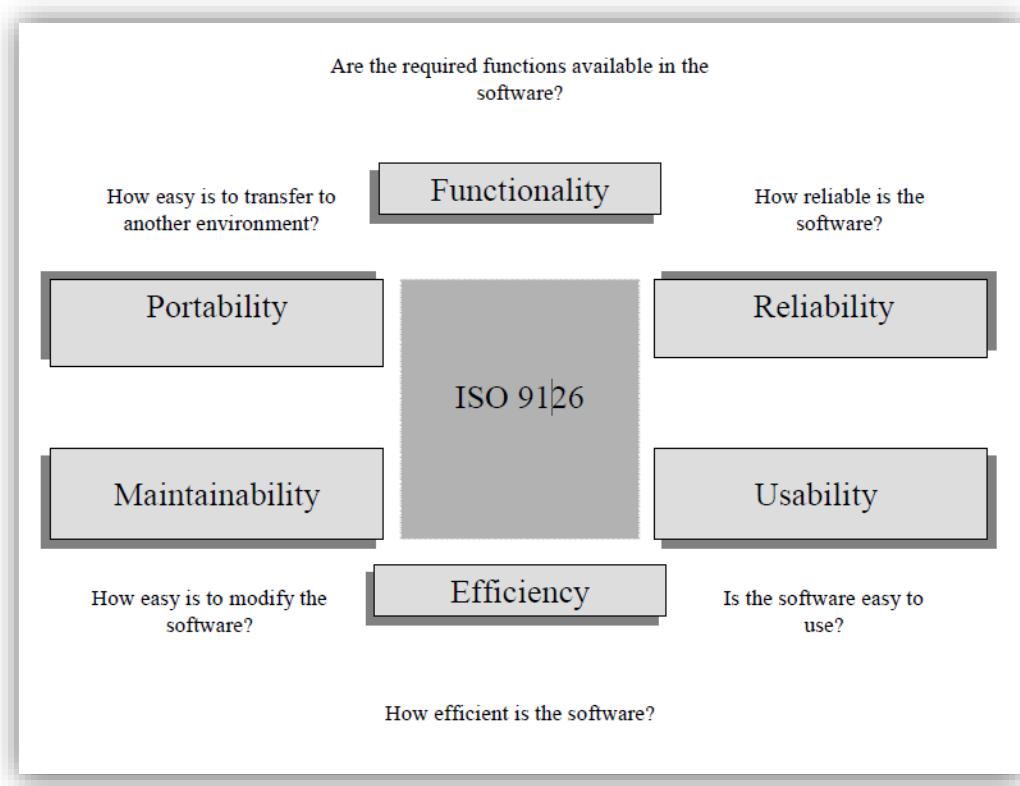
Pengujian dilakukan untuk meminimalisir kesalahan atau *error* dan memastikan keluaran dari program sesuai dengan yang diinginkan.

e. Pendukung (*support*) atau pemeliharaan (*maintenance*)

Tahap ini digunakan untuk melakukan proses pemeliharaan program. Ketika program sudah diterima oleh *user*, terkadang *user* menginginkan perubahan pada program. Perubahan bisa terjadi karena adanya kesalahan atau bisa saja karena program harus beradaptasi dengan lingkungan baru. Tahap pemeliharaan dapat mengulangi proses pengembangan mulai dari tahap analisis spesifikasi perangkat lunak yang ada, tapi tidak untuk membuat perangkat lunak baru.

7. Analisis Kualitas Perangkat Lunak

Software bisa dikatakan baik apabila memenuhi standar kualitas *software*. ISO 9126 merupakan standar internasional yang memiliki tujuan untuk menyediakan sebuah kerangka kerja untuk evaluasi dari kualitas *software* (Chua & Dyson, 2004). ISO 9126 mendefinisikan kualitas produk perangkat lunak, model, karakteristik mutu, dan metrik terkait digunakan untuk mengevaluasi dan menetapkan kualitas sebuah produk *software*. ISO 9126 menetapkan enam karakteristik pengujian kualitas *software*. Karakteristik tersebut yang dapat dilihat Gambar 3.



Gambar 3. Model ISO 9126 (Chua & Dyson, 2004)

Dari enam karakteristik tersebut, dibagi lagi menjadi beberapa subkarakteristik seperti pada Tabel 3.

Tabel 3. Subkarakteristik ISO 9126 dan Penjelasannya (Padayachee, Kotze, & van Der Merwe, 2010)

Characteristic	Sub Characteristics	Explanation
Functionality	Suitability Accurateness Interoperability Compliance Security	‘Can software perform the tasks required?’ ‘Is the result as expected?’ ‘Can the system interact with another system?’ ‘Is the system compliant with standards?’ ‘Does the system prevent unauthorized access?’
Reliability	Maturity Fault tolerance Recoverability	‘Have most of the faults in the software been eliminated over time?’ ‘Is the software capable of handling errors?’ ‘Can the software resume working & restore lost data after failure?’
Usability	Understandability Learnability Operability Attractiveness	‘Does the user comprehend how to use the system easily?’ ‘Can the user learn to use the system easily?’ ‘Can the user use the system without much effort?’ ‘Does the interface look good?’
Efficiency	Time Behaviour Resource utilization	‘How quickly does the system respond?’ ‘Does the system utilize resources efficiently?’
Maintainability	Analyzability Changeability Stability Testability	‘Can faults be easily diagnosed?’ ‘Can the software be easily modified?’ ‘Can the software continue functioning if changes are made?’ ‘Can the software be tested easily?’
Portability	Adaptability Installability Conformance Replaceability	‘Can the software be moved to other environments?’ ‘Can the software be installed easily?’ ‘Does the software comply with portability standards?’ ‘Can the software easily replace other software?’

a. Aspek *Functionality*

Functionality merupakan kemampuan *software* untuk menyediakan fungsi yang sesuai dengan kebutuhan baik yang tersurat maupun yang tersirat ketika digunakan dalam kondisi tertentu (ISO, 2001). Pada aspek *functionality* berdasarkan ISO (2003), *functionality* suatu *software* dapat dihitung dengan menggunakan rumus berikut:

$$X = 1 - \frac{A}{B}$$

Keterangan:

$X = \text{functionality}$

A = Jumlah fungsi yang gagal uji

B = Jumlah seluruh fungsi

$0 \leq X \leq 1$. *Functionality* dikatakan baik jika mendekati 1.

Dalam ISO (2003), fungsi-fungsi tersebut diuji dengan menggunakan *test case*.

Bhagwat (2009) menyatakan bahwa pengertian *test case* merupakan pernyataan singkat tentang sesuatu yang harus diuji.

Aspek *functionality* juga memperhatikan *security* sebuah *software*. Menurut Vieira, Antunes, & Madeira (2009), kerentanan berbahaya dalam aplikasi *web* adalah *SQL Injection* dan *Cross Site Scripting (XSS)*. *SQL Injection* adalah kerentanan yang terjadi ketika suatu *software* memberikan kemampuan pada penyerang untuk mempengaruhi *Structured Query Language (SQL)* (Clarke, 2009). Dengan memiliki kemampuan tersebut, penyerang dapat memasukkan kode *SQL* ke dalam parameter *input* pengguna yang kemudian diteruskan ke *back-end SQL Server* untuk dijalankan (Clarke, 2009). Kerentanan tersebut bisa berbahaya karena penyerang dapat memasukkan kode *SQL* yang tidak sesuai, misalnya kode *SQL* tersebut menampilkan *password* dari suatu *software*. Kemudian *Cross Site Scripting (XSS)* adalah serangan yang berupa penyuntikan kode berbahaya, biasanya dalam bentuk *script* yang disuntikkan dari sisi *client* ke dalam aplikasi *web* dari sumber luar aplikasi *web* tersebut (Cook, 2003). Untuk menguji *security* pada *software* dapat digunakan *Acunetix Web Vulnerability Scanner* (Vieira, Antunes, & Madeira, 2009). *Acunetix Web Vulnerability Scanner* dapat memindai

seluruh bagian *web* untuk diuji dengan melakukan serangan *SQL Injection* dan *Cross Site Scripting (XSS)*.

b. Aspek *Reliability*

Dalam ISO (2001), *Reliability* diartikan sebagai kemampuan produk perangkat lunak untuk mempertahankan performa pada tingkat tertentu saat digunakan dalam kondisi tertentu. ISO (2003) menyatakan bahwa untuk menguji *reliability* dengan melakukan *stress testing*. Menurut Pradhan (2013) untuk melakukan *stress testing* dapat digunakan *software Web Application Load, Stress and Performance Testing* (WAPT). Pengertian *software* WAPT adalah sebagai berikut “*WAPT is a load and stress testing tool that provides an easy-to-use and cost-effective way to test any website*”. Jadi WAPT merupakan *software* yang dapat digunakan untuk melakukan *stress testing* pada semua *website*. Asthana & Olivieri (2009) mengungkapkan bahwa standar *reliability* yang dapat diterima bedasarkan standar Telcordia adalah 95% atau lebih tes pada *software* harus lolos. Sehingga nantinya *software* dikatakan lolos uji pada aspek *reliability* jika minimal 95% *software* dapat berjalan dengan baik ketika diuji *stress testing* menggunakan WAPT.

c. Aspek *Usability*

Usability menurut ISO (2001) adalah kemampuan dari produk perangkat lunak untuk dipahami, dipelajari, digunakan, dan menarik bagi pengguna bila digunakan dalam kondisi tertentu. Kemudahan pengguna dalam menggunakan produk perangkat lunak tersebut tentunya depengaruhi dengan apa yang dilihat langsung oleh pengguna. Bagian *software* berbasis *web* yang akan langsung bisa dirasakan oleh *user* adalah *user interface*. Sehingga *user interface* *software* akan

berpengaruh dengan respon *user* terhadap *software*. Arnold M. Lund (2001) membuat kuesioner yang bernama USE *Questionnaire*. USE merupakan kepanjangan dari *Usefulness, Satisfaction, and Ease of use*. Menurut Lund (2001), kuesioner ini dapat digunakan untuk mengukur aspek *usability* dan telah banyak digunakan banyak perusahaan. *Usability* didefinisikan oleh empat kualitas komponen: *usefulness, ease of use, easy of learning, dan satisfaction*.

d. Aspek *Efficiency*

Menurut ISO (2001), *efficiency* adalah kemampuan produk *software* untuk memberikan kinerja yang sesuai, relatif terhadap jumlah sumber daya yang digunakan, dalam kondisi yang telah ditetapkan. Menurut Padayachee, Kotze, & van Der Merwe (2010), *efficiency* mencakup berapa waktu yang dibutuhkan sistem untuk melakukan respon dan bagaimana *resources* dimanfaatkan secara efisien artinya dalam *software* berbasis *web* dapat diketahui berapa waktu untuk *load* halaman *web* dan bagaimana *source code* diprogram secara efisien. Singh (2014) menjelaskan bahwa untuk mengetahui berapa waktu untuk *load* halaman *web* dapat menggunakan GTMetrix. Selain itu dapat digunakan untuk mengukur waktu *load* halaman *web*, GTMetrix juga dapat memberikan rekomendasi agar *source code* diprogram agar lebih efisien. Normalnya semakin cepat waktu *website* dapat diakses oleh pengguna maka akan semakin baik. Nielsen (2006) menjelaskan ada tiga batasan waktu yang perlu diingat ketika mengoptimalkan *web* berdasarkan persepsi manusia. Batasan pertama yaitu 0,1 detik. Pada batasan ini pengguna merasakan respon dari *web* yang cepat. Batasan kedua yaitu 1 detik. Pada batasan ini pengguna bisa merasakan adanya jeda namun proses ini membuat pengguna dapat merasakan komputer menjalankan suatu

proses yang kemudian mengeluarkan hasil dan pikiran pengguna belum merasa terganggu dengan jeda ini. Batasan ketiga yaitu 10 detik. Pada batasan ini pengguna mulai memperhatikan jeda yang ada. Pengguna mulai berharap agar respon dapat berjalan lebih cepat. Jika lebih dari batasan tersebut atau lebih dari 10 detik maka seringnya pengguna akan meninggalkan *web*. Apabila ada pengguna yang masih berada di *web* tersebut, pekerjaan mereka akan terhambat dengan adanya jeda tersebut. Menurut Aptimize (2010), standar kecepatan *load* suatu *website* adalah seperti pada Tabel 4.

Tabel 4. *Website Performance Benchmark*. (Aptimize, 2010)

Benchmark	Load Time
<i>First view load time (Domestic)</i>	<i>7,066 seconds</i>
<i>First view load time (International)</i>	<i>9,462 seconds</i>

e. Aspek *Maintainability*

Maintainability merupakan kemampuan produk perangkat lunak untuk dimodifikasi. Modifikasi dapat mencakup koreksi, perbaikan atau adaptasi dari perangkat lunak terhadap perubahan lingkungan dan persyaratan dan spesifikasi fungsional (ISO, 2001). Menurut Coleman (1994), pengertian *Maintainability* berdasarkan standar IEEE merupakan kemudahan yang sistem perangkat lunak agar dapat dimodifikasi untuk memperbaiki kesalahan, meningkatkan performa atau atribut lainnya, atau beradaptasi dengan perubahan lingkungan. Jadi pada intinya *maintainability* merupakan tingkat kemampuan *software* agar dapat dimodifikasi atau dikembangkan kembali. Tingkat kemampuan *software* dapat diukur dengan berbagai macam cara. Menurut Najm (2014), *software metric* yang paling sering digunakan untuk mengukur *maintainability* adalah *maintainability Index (MI)*. MI merupakan *software metric* yang mengukur bagaimana suatu

source code dari *software* mudah untuk di-*maintenance*. MI dihitung dari rumus yang terdiri dari *Lines of Code (LOC)*, *Cyclomatic Complexity (CC)* dan *Halstead Volume (HV)*. Rumus *maintainability index* adalah sebagai berikut Coleman (1994):

$$MI = 171 - 5.2 \times \ln(aveVol) - 0.23 \times aveV(g') - 16.2 \times \ln(aveLOC) \\ + (50 \times \sin(\sqrt{2.46 \times perCM}))$$

Keterangan:

MI	= <i>maintainability index</i>
aveVol	= rata-rata <i>Halstead Volume (HV)</i>
ave V(g')	= rata-rata <i>Cyclomatic Complexity (CC)</i>
aveLOC	= rata-rata <i>Lines of Code (LOC)</i>
perCM	= persentase <i>comment</i> pada <i>source code</i>

Verifysoft (2010) menjelaskan bahwa *Halstead metric* didapatkan dari penginterpretasian *source code* menjadi urutan *token* dan mengklasifikasikan setiap *token* menjadi *operator* atau *operand*. *Halstead Volume* dihitung berdasarkan *operator* dan *operand* dengan rumus sebagai berikut:

$$V = N * \log_2(n)$$

Keterangan:

V	= <i>Halstead Volume</i>
N	= <i>Program length</i> (Jumlah seluruh <i>operator</i> dan <i>operand</i>)
n	= <i>Vocabulary size</i> (Jumlah seluruh <i>unique/distinct operator</i> dan <i>operand</i>)

Cyclomatic complexity didapatkan dari mengukur jumlah logika keputusan dalam modul perangkat lunak tunggal. Pada *cyclomatic complexity* didefinisikan untuk setiap modul dihitung dengan $e - n + 2$, dengan e dan n merupakan jumlah *edges* dan *nodes* pada *control flow graph*. *Control flow graphs* merupakan pendeskripsi

struktur logika dari modul *software* (Watson, McCabe, & Wallace, 1996). Untuk mempermudah perhitungan MI dapat digunakan *tool* yang bernama PHPMetric (Lepine, 2014). PHPMetric merupakan *software* dengan lisensi *MIT license (open source)* yang disebut "*Static analysis tool for PHP*" yang artinya merupakan alat untuk menganalisis PHP. Pada aspek *maintainability* PHPMetric dapat digunakan untuk menghitung *maintainability index (MI)* yang dihitung dari *Halstead Volume (HV)*, *Cyclomatic Complexity (CC)*, *count of source Lines Of Code (LOC)* dan *Percent Line of Comment (CM)*. Coleman (1994) menyatakan bahwa "*All components above the 85 maintainability index are highly maintainable, components between 85 and 65 are moderately maintainable, and components below 65 are difficult to maintain*". Jadi nilai MI minimal agar *software* tidak sulit di-*maintenance* adalah 65.

f. Aspek *Portability*

Portability merupakan kemampuan produk *software* untuk ditransfer dari satu lingkungan yang lain (ISO, 2001). *Software* berbasis *web* diakses oleh *web browser* agar dapat digunakan. Agar dapat memenui *portability* perlu dicoba pada berbagai macam *web browser*. Salonen (2012) mengungkapkan bahwa pengembang harus mencoba *software*-nya untuk bisa diakses dengan tujuh *browser* dengan versi berbeda dari tiga mayoritas *browser* yang digunakan untuk perangkat *desktop* dan lima *browser* untuk perangkat *mobile*. Berdasarkan *browser statistic and trends* pada bulan juni 2014 di situs w3schools *google chrome*, *Mozilla Firefox* dan *Internet Explorer* menempati urutan tiga teratas seperti pada Tabel 5.

Tabel 5. *Browser Statistic* (W3schools, 2014)

Browser Statistics

2014	Chrome	Internet Explorer	Firefox	Safari	Opera
June	59.3 %	8.8 %	25.1 %	3.7 %	1.8 %
May	59.2 %	8.9 %	24.9 %	3.8 %	1.8 %
April	58.4 %	9.4 %	25.0 %	4.0 %	1.8 %
March	57.5 %	9.7 %	25.6 %	3.9 %	1.8 %

BrowseEmAll dapat digunakan untuk melakukan *testing web* dengan berbagai macam *browser* dan *multiple platforms* termasuk perangkat *mobile* (Moss, 2013). Sehingga *BrowseEmAll* dapat mempermudah dan membantu dalam *testing software* dari aspek *portability*.

Selain ISO 9126 ada standar kualitas *software* lainnya seperti *McCall, Boehm, FURPS, Dromey*, dan *The Bayesian Belief Network (BBN)*. Masing-masing standar kualitas memiliki karakteristik yang berbeda-beda. Menurut Fahmy (2012), ISO 9126 dibuat berdasarkan standar kualitas *software* tersebut sehingga karakteristik ISO merupakan gabungan dari beberapa standar kualitas yang telah ada.

B. Hasil Penelitian yang Relevan

1. Aplikasi Tes Potensi Akademik *Online* Sebagai Alat Bantu Proses Pembelajaran Bagi Calon Peserta Ujian oleh Husnah Tuti Dianak di Universitas Pembangunan Nasional "Veteran" Jawa Timur pada tahun 2012. Penelitian ini bertujuan untuk membuat aplikasi yang dapat mengukur kemampuan seseorang di bidang akademik umum atau sering juga diidentifikasi dengan tes kecerdasan seseorang. Aplikasi ini berbasis *web* yang bekerja agar dapat menyajikan informasi tentang tes potensi akademik. Program dirancang dengan menggunakan konsep struktural. Hasil dari penelitian ini didapat

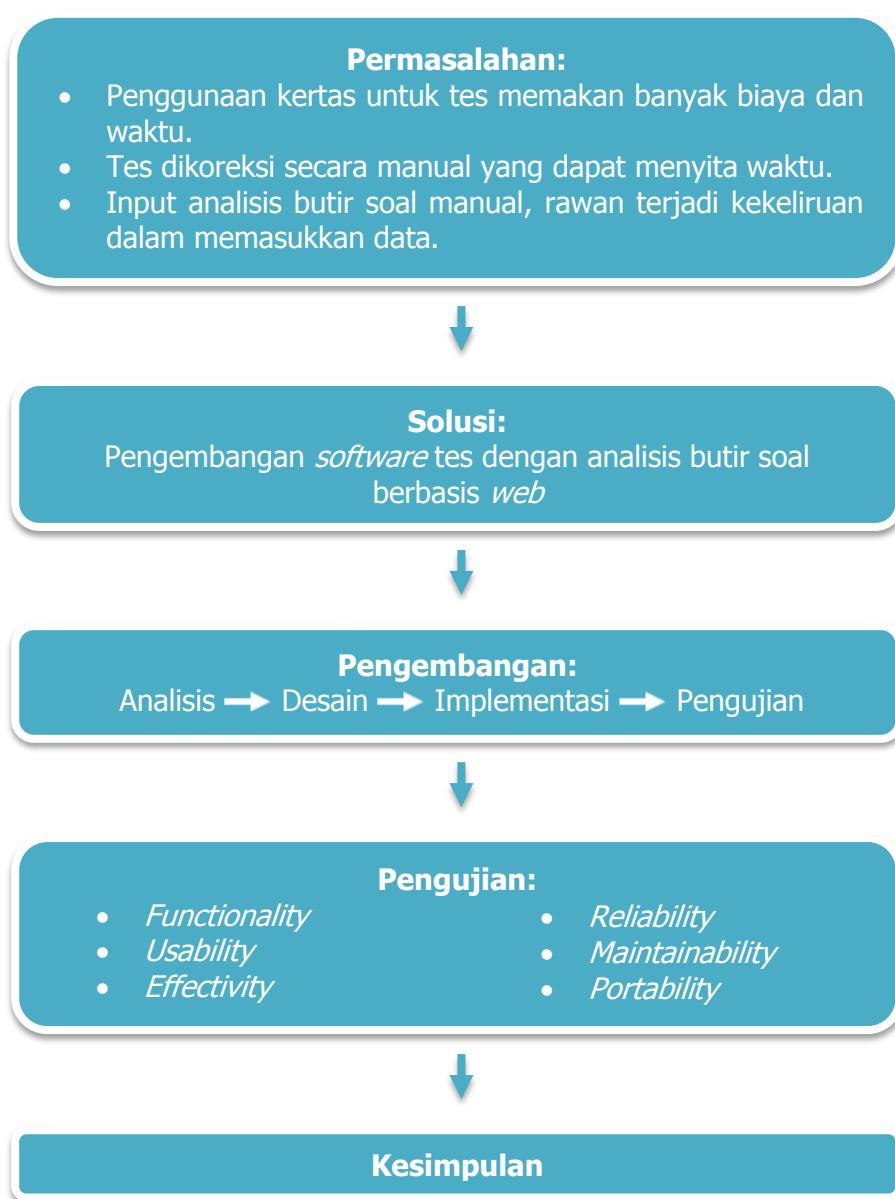
- bahwa aplikasi dapat memudahkan *user* dalam hal belajar tes potensi akademik dengan cepat dan mudah karena memanfaatkan internet.
2. Perancangan Aplikasi Simulasi TOEFL (*Test Of English as Foreign Language*) oleh Toni Parwanto di Universitas Islam Negeri Syarif Hidayatullah Jakarta pada tahun 2011. Penelitian ini menghasilkan simulasi TOEFL berbasis *web* yang dapat menguji kemampuan *listening*, *structure* dan *reading* untuk mengukur kemampuan bahasa inggris seseorang. Dalam pengembangannya aplikasi ini menggunakan bahasa pemrograman PHP dan *database* MySQL.
 3. Rancang Bangun Simulasi Tes *Online* Calon Pegawai Negeri Sipil (CPNS) oleh Imam Ahfas di Universitas Muria Kudus pada tahun 2012. Penelitian ini menghasilkan Simulasi Tes *Online* CPNS berbasis *web* yang bersifat gratis yang nantinya bisa digunakan oleh *user* untuk melakukan latihan soal tentang ujian untuk tes CPNS. Pengembangan aplikasi menggunakan pemodelan UML dan menggunakan alat *Dreamwaver 8.0* dan *XAMPP*.

C. Kerangka Pikir

MAN Yogyakarta 1 dalam melaksanakan tes masih menggunakan kertas dan analisis butir soal masih perlu melakukan *input* secara manual. Hal tersebut mengakibatkan terjadinya pemborosan waktu dan biaya serta rawan terjadinya kekeliruan saat melakukan *input* data. *Software* tes dengan analisis butir soal berbasis *web* ini diharapkan dapat mengurangi biaya dan waktu yang dibutuhkan untuk melakukan tes serta menghilangkan terjadinya kekeliruan *input* data untuk analisis butir soal karena data dari hasil tes langsung diolah secara otomatis oleh *software* untuk melakukan analisis butir soal. *Software* dikembangkan dengan model *waterfall* dengan pengujian yang menggunakan standar kualitas ISO 9126

yang memiliki karakteristik *functionality*, *usability*, *effectivity*, *reliability*, *maintainability*, dan *portability* agar menjadi *software* yang baik. Hasil pengujian *software* dibandingkan dengan standar pada setiap karakteristik ISO 9126 sehingga dapat diketahui apakah *software* tersebut baik sesuai dengan ISO 9126.

Ulasan tersebut digambarkan pada Gambar 4.



Gambar 4. Kerangka Pikir

D. Pertanyaan Penelitian

Pertanyaan penelitian dari pengembangan *software* tes dengan analisis butir soal berbasis *web* di MAN Yogyakarta 1 sebagai berikut :

1. Apakah *software* tes dengan analisis butir soal berbasis *web* di MAN Yogyakarta 1 memenuhi aspek *functionality*?
2. Apakah *software* tes dengan analisis butir soal berbasis *web* di MAN Yogyakarta 1 memenuhi aspek *efficiency*?
3. Apakah *software* tes dengan analisis butir soal berbasis *web* di MAN Yogyakarta 1 memenuhi aspek *usability*?
4. Apakah *software* tes dengan analisis butir soal berbasis *web* di MAN Yogyakarta 1 memenuhi aspek *reliability*?
5. Apakah *software* tes dengan analisis butir soal berbasis *web* di man yogyakarta 1 memenuhi aspek *maintainability*?
6. Apakah *software* tes dengan analisis butir soal berbasis *web* di man yogyakarta 1 memenuhi aspek *portability*?

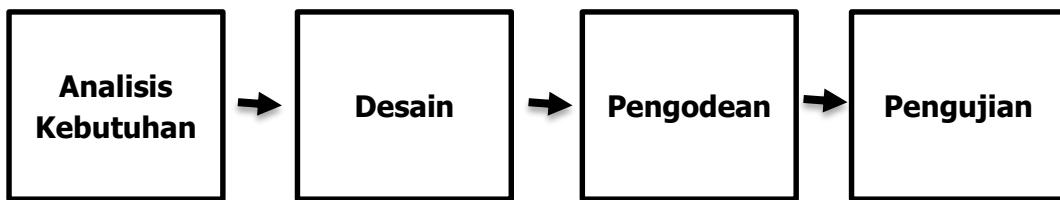
BAB III

METODE PENELITIAN

A. Model Pengembangan

Penelitian ini memiliki tujuan untuk mengembangkan *software* tes dengan analisis butir soal berbasis *web* di MAN Yogyakarta 1. Berdasarkan tujuan itu, digunakan metode *Research and Development* (R&D) agar pengembangan dari *software* berbasis *web* tersebut dapat memenuhi standar kualitas. Sugiyono (2009) menyatakan bahwa R&D adalah penelitian yang digunakan untuk menghasilkan produk tertentu dan menguji keefektifan produk tersebut.

Tahapan – tahapan dalam pengembangan perangkat lunak sering disebut dengan SDLC atau *Software Development Life Cycle*. Model SDLC yang digunakan dalam penelitian ini adalah *waterfall*. Model *waterfall* digambarkan oleh Rosa & Shalahuddin (2013) seperti pada Gambar 5.



Gambar 5. Ilustrasi Model Waterfall

B. Prosedur Pengembangan

1. Analisis Kebutuhan

Analisis kebutuhan dilakukan dengan mengumpulkan kebutuhan perangkat lunak yang sesuai dengan kebutuhan *user* atau pengguna. Pada tahap analisis kebutuhan ini dilakukan dengan cara observasi dan wawancara langsung dengan pihak sekolah. Sugiyono (2009) menjelaskan bahwa teknik observasi dilakukan bila penelitian berkenaan dengan perilaku manusia, proses kerja, gejala-gejala alam,

dan bila responden yang diamati tidak terlalu besar. Observasi dilakukan dengan mengamati kegiatan belajar mengajar khususnya proses ulangan harian yang dilakukan oleh guru di MAN Yogyakarta 1. Kemudian wawancara merupakan teknik pengumpulan data untuk melakukan studi pendahuluan untuk menemukan permasalahan yang diteliti dan juga untuk mengetahui hal-hal yang lebih mendalam dari responden (Sugiyono, 2009). Dalam analisis kebutuhan ini, wawancara dilakukan secara langsung dengan guru di MAN Yogyakarta 1. Hasil wawancara dapat dilihat pada Lampiran 6. Hasil dari analisis kebutuhan berupa spesifikasi yang dibutuhkan dalam pengembangan *software*.

2. Desain

Spesifikasi yang didapatkan dari analisis kebutuhan diubah menjadi desain *software* pada tahap ini. Dalam membuat desain, agar hasilnya baik digunakan suatu pemodelan. Pemodelan yang digunakan adalah *Unified Modeling Language (UML)*. Menurut Rosa & Shalahuddin (2013), "UML merupakan bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung".

3. Implementasi / Pengodean

Pada tahap implementasi, desain *software* yang telah dibuat pada tahap sebelumnya dibuat sebagai kode program sehingga akan menghasilkan *software* sesuai dengan desain yang telah dibuat. *Software* yang dirancang adalah *software* yang berbasis *web*. Dalam implementasi digunakan *framework* Laravel agar implementasi dapat berjalan cepat dan mudah.

4. Pengujian

Pada tahap pengujian, hasil dari implementasi *software* dianalisis kualitasnya agar dapat diketahui apakah *software* dapat dikategorikan sebagai *software* yang baik. Analisis kualitas *software* dilakukan dengan menggunakan standar kualitas *software* ISO 9126. Dalam pengujian berdasarkan standar ISO 9126 *software* harus memenuhi aspek *functionality, reliability, usability, efficiency, maintainability*, dan *portability*.

C. Sumber Data/ Subjek Penelitian

Dalam pengujian untuk aspek *usability* dan *functionality* dari *software* digunakan responden ahli dalam bidang pengembangan *software* berbasis *web* dan pengguna dari sekolah yaitu siswa dan guru. Mengacu pada Nielsen (2006) yang menyatakan bahwa jumlah sampel paling sedikit dalam penelitian adalah 20 orang agar mendapatkan data yang signifikan secara statistik. Dalam penelitian ini untuk pengujian *usability* menggunakan 30 responden yang terdiri dari 28 siswa dan 2 guru. Data responden untuk pengujian *usability* dapat dilihat pada Lampiran 8. Subjek penelitian pengujian *functionality* menggunakan 3 responden ahli di bidang pengembangan *software* berbasis *web*. Untuk pengujian lainnya yaitu *efficiency*, sumber data didapatkan dari hasil pengujian dengan menggunakan alat ukur GTMetrix. Kemudian untuk pengujian *reliability* sumber data didapatkan dari hasil pengujian menggunakan WAPT. Uji *maintainability* sumber data didapatkan dari perhitungan *Maintainability Index (MI)*. Yang terakhir yaitu uji *portability* sumber data didapatkan dari hasil percobaan menggunakan beberapa *web browser*.

D. Metode dan Alat Pengumpul Data

Sesuai dengan pengujian *software* yang menggunakan ISO 9126, dalam penelitian ini dijabarkan metode dan alat pengumpulan data berdasarkan masing-masing aspek pengujian.

1. Aspek *Functionality*

Pengujian pada aspek ini dilakukan oleh *programmer* yang menguasai bidang *web*. Pengujian dilakukan dengan mengisi kuesioner yang sesuai dengan fungsi dari *software* yang diuji sehingga dapat diketahui apakah fungsi-fungsi pada *software* berjalan dengan baik atau tidak. Instrumen untuk mengukur aspek *functionality* dapat dilihat pada Tabel 6.

Tabel 6. Instrumen *functionality*

No.	Fungsi	Pernyataan	Lolos	
			Ya	Tidak
Halaman Peserta Tes				
1.	Daftar tes	Fungsi untuk melihat daftar tes yang telah dibuat oleh <i>admin/guru</i> dari halaman peserta tes guna mengikuti tes sudah berjalan dengan benar		
2.	Identitas Peserta Tes	Fungsi untuk melakukan <i>input</i> data peserta tes sudah berjalan dengan benar		
3.	Pelaksanaan tes	Fungsi untuk pelaksanaan tes bagi peserta tes sudah berjalan dengan benar		
4.	Hasil tes	Fungsi untuk melihat hasil tes dari halaman peserta tes sudah berjalan dengan benar		
Halaman Guru (<i>Admin</i>)				
5.	<i>Login</i>	Fungsi untuk masuk ke halaman <i>admin/guru</i> sudah berjalan dengan benar		
6.	Mengubah Akun	Fungsi untuk mengubah akun yang digunakan untuk <i>login</i> sudah berjalan dengan benar		

Lanjutan Tabel 6.

No.	Fungsi	Pernyataan	Lolos	
			Ya	Tidak
Halaman Guru (Admin)				
7.	Reset Akun	Fungsi untuk melakukan <i>reset</i> akun sudah berjalan dengan benar		
8.	Logout	Fungsi untuk keluar dari halaman <i>admin/guru</i> sudah berjalan dengan benar		
9.	Tambah Tes	Fungsi untuk menambah tes sudah berjalan dengan benar		
10.	Edit Tes	Fungsi untuk mengedit tes sudah berjalan dengan benar		
11.	Hapus Tes	Fungsi untuk menghapus tes sudah berjalan dengan benar		
12.	Melihat Tes	Fungsi untuk melihat tes sudah berjalan dengan benar		
13.	Pengaturan Tes	Fungsi untuk mengatur tes sudah berjalan dengan benar		
14.	Duplikat Tes	Fungsi untuk menduplikat tes sudah berjalan dengan benar		
15.	Tambah Soal	Fungsi untuk menambah butir soal sudah berjalan dengan benar		
16.	Edit Soal	Fungsi untuk mengedit butir soal sudah berjalan dengan benar		
17.	Hapus Soal	Fungsi untuk menghapus butir soal sudah berjalan dengan benar		
18.	Melihat Soal	Fungsi untuk melihat butir soal sudah berjalan dengan benar		
19.	Lihat Laporan	Fungsi untuk melihat daftar laporan tes sudah berjalan dengan benar		
20.	Hasil Tes	Fungsi untuk melihat rapor hasil tes peserta tes sudah berjalan dengan benar		
21.	Detail Hasil Tes	Fungsi untuk melihat hasil tes secara detail sudah berjalan dengan benar		
22.	Hasil Analisis Butir Soal	Fungsi untuk melihat hasil analisis butir soal sudah berjalan dengan benar		
23.	Cetak Hasil Tes	Fungsi untuk mencetak hasil tes sudah berjalan dengan benar		
24.	Cetak Detail Hasil Tes	Fungsi untuk mencetak detail hasil tes sudah berjalan dengan benar		
25.	Cetak Hasil Analisis Butir Soal	Fungsi untuk mencetak hasil analisis butir soal sudah berjalan dengan benar		

Dalam segi *security*, *software* diuji dengan menggunakan *Acunetix Web Vulnerability Scanner*. *Acunetix Web Vulnerability Scanner* mengukur segi *security* dengan menguji *software* berbasis *web* dengan serangan *XSS* dan *SQL Injection*. Sehingga dapat diketahui apakah *software* dapat menahan serangan tersebut.

2. Aspek *Reliability*

Pengujian aspek *reliability* menggunakan aplikasi WAPT atau *Web Application Load, Stress and Performance Testing*. WAPT dapat memberikan sejumlah beban kepada *software* sehingga dapat diketahui apakah *software* dapat bekerja dengan baik apabila diberi beban.

3. Aspek *Usability*

Pengujian aspek *usability* menggunakan kuesioner yang dibuat oleh Lund (2001) yaitu USE (*Usefulness, Satisfaction, and Ease of use*). Kuesioner USE menggunakan Skala likert dengan bentuk *checklist*. Pilihan jawabannya yaitu Sangat Setuju (SS), Setuju (S), Ragu-ragu (RG), Tidak Setuju (TS), dan Sangat Tidak Setuju (STS). kuesioner USE dapat dilihat pada Tabel 7.

Tabel 7. Kuesioner USE

No.	Pernyataan	Jawaban				
		STS	TS	RG	S	SS
1.	<i>Software</i> ini membantu saya bekerja lebih efektif					
2.	<i>Software</i> ini membantu saya bekerja lebih produktif					
3.	<i>Software</i> ini sangat berguna					
4.	<i>Software</i> ini memberikan saya pengendalian lebih atas aktivitas saya					
5.	<i>Software</i> ini mempermudah saya dalam menyelesaikan apa yang ingin saya selesaikan					
6.	<i>Software</i> ini menghemat waktu saya ketika saya menggunakan					

Lanjutan Tabel 7:

No.	Pernyataan	Jawaban				
		STS	TS	RG	S	SS
7.	<i>Software</i> ini sesuai dengan kebutuhan saya					
8.	<i>Software</i> ini melakukan segala sesuatu yang saya harapkan untuk dilakukan					
9.	<i>Software</i> ini mudah digunakan					
10.	<i>Software</i> ini praktis digunakan					
11.	<i>Software</i> ini mudah dipahami					
12.	<i>Software</i> ini membutuhkan langkah-langkah yang sedikit untuk mencapai apa yang ingin saya lakukan dengan <i>software</i> ini					
13.	<i>Software</i> ini fleksibel					
14.	Tidak ada kesulitan dalam menggunakan <i>Software</i> ini					
15.	Saya dapat menggunakan <i>Software</i> ini tanpa panduan tertulis					
16.	Saya tidak melihat adanya inkonsistensi saat saya gunakan <i>software</i> ini					
17.	Baik pengguna yang sesekali menggunakan dan pengguna yang biasa menggunakan akan menyukai <i>software</i> ini					
18.	Saya dapat menangani kesalahan dengan cepat dan mudah					
19.	Saya dapat menggunakan <i>software</i> ini secara benar setiap saat					
20.	Saya belajar untuk menggunakan <i>software</i> ini secara cepat					
21.	Saya mudah mengingat bagaimana menggunakan ini					
22.	<i>Software</i> ini mudah untuk dipelajari bagaimana penggunaannya					
23.	Saya menjadi terampil menggunakan <i>Software</i> ini secara cepat					
24.	Saya puas dengan <i>software</i> ini					
25.	Saya akan merekomendasikan <i>software</i> ini ke teman					
26.	<i>Software</i> ini menyenangkan untuk digunakan					
27.	<i>Software</i> ini bekerja seperti yang saya inginkan					
28.	<i>Software</i> ini memiliki tampilan yang sangat bagus					
29.	Menurut saya, saya perlu memiliki <i>software</i> ini					
30.	<i>Software</i> ini nyaman untuk digunakan					

4. Aspek *Efficiency*

Pengujian pada aspek ini menggunakan alat ukur GTMetrix. Pengujian dilakukan dengan memasukkan *url software* pada *website* GTMetrix, kemudian GTMetrix akan mengukur kecepatan *load web* dan optimasi *source code* pemrograman *software*. Dari hasil pengukuran tersebut dapat diketahui bagaimana *efficiency software*.

5. Aspek *Maintainability*

Pengujian aspek *Maintainability* menggunakan perhitungan *Maintainability index*. Perhitungan *Maintainability index* berdasarkan pada perhitungan dari *Lines of Code (LOC)*, *Cyclomatic Complexity (CC)*, dan *Halstead Volume (HV)* *source code* program dengan menggunakan PHPMetric.

6. Aspek *Portability*

Pengujian pada aspek ini dilakukan dengan mengakses *software* dengan berbagai macam *web browser*. Dalam melakukan ini alat bantu yang digunakan adalah BrowseEmAll. BrowseEmAll dapat menjalankan *virtual web browser* sehingga *software* dapat diuji dengan berbagai macam *web browser*.

E. Teknik Analisis Data

Standar pengujian kualitas *software* berdasarkan ISO 9126. Berikut adalah penjelasan teknik analisis data dalam pengujian *software* berdasarkan masing-masing kriteria dari ISO 9126.

1. *Functionality*

Pengujian aspek *functionality* dilakukan dengan menguji fungsi-fungsi pada *software* berbasis *web* sesuai *test case* yang dilakukan oleh ahli dalam bidang tersebut. *Test case* berbentuk *checklist* dengan jawaban tegas yaitu "Ya-Tidak".

Skala pengukuran ini disebut dengan skala Guttman (Sugiyono, 2009). Dengan menggunakan skala tersebut dapat diketahui jawaban apakah fungsi berjalan dengan baik atau tidak. Setelah didapatkan jawaban dari *test case* kemudian hasil dihitung dengan menggunakan rumus (ISO, 2003):

$$X = 1 - \frac{A}{B}$$

Keterangan:

X = *functionality*

A = Jumlah fungsi yang gagal uji

B = Jumlah seluruh fungsi

Hasil dari perhitungan tersebut menghasilkan nilai **0 <= X <= 1**. *Functionality* semakin baik jika mendekati nilai 1.

Selain itu pengujian juga dilakukan pada aspek *security*. Pada aspek ini *software* diuji dari serangan *SQL Injection* dan *XSS* dengan menggunakan *Acunetix Web Vulnerability Scanner*. *Software* ini digunakan untuk mendeteksi apakah masih ada kemungkinan terjadinya serangan *SQL Injection* dan *XSS* atau tidak.

2. *Reliability*

Pengujian pada tahap ini dilakukan dengan pengujian *stress testing* yang dilakukan dengan WAPT. Hasil dari *stress testing* menggunakan WAPT ini kemudian harus berhasil minimal 95% sesuai standar Telcordia.

3. *Usability*

Pengujian *usability* menggunakan kuesioner USE dengan skala likert sebagai skala pengukuran dalam pengujian. Menurut Sugiyono (2009), jawaban tiap *item* pada instrumen yang menggunakan skala likert memiliki gradasi dari sangat positif

sampai sangat negatif. Jawaban tersebut diberi skor agar dapat dianalisis. Berikut contoh skala likert yang dikemukakan oleh Sugiyono (2009) :

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

Sugiyono (2009) menjelaskan untuk melakukan analisis data hasil pengujian *usability* dengan menghitung jumlah rata-rata jawaban berdasarkan skor. Misalnya dari 100 responden didapatkan hasil seperti pada Tabel 8.

Tabel 8. Contoh Hasil Jawaban Responden

Jawaban	Jumlah penjawab
SS	25
ST	40
RG	5
TS	20
STS	10

Berdasarkan skor yang telah ditetapkan dapat dihitung sebagai berikut :

$$\text{Jumlah skor yang menjawab SS} = 25 \times 5 = 125$$

$$\text{Jumlah skor yang menjawab S} = 40 \times 4 = 160$$

$$\text{Jumlah skor yang menjawab RR} = 5 \times 3 = 15$$

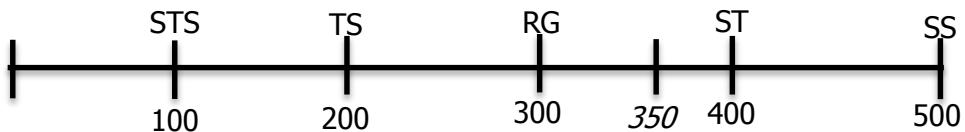
$$\text{Jumlah skor yang menjawab TS} = 20 \times 2 = 20$$

$$\text{Jumlah skor yang menjawab STS} = 10 \times 1 = 10$$

$$\text{Jumlah skor Total} = 350$$

Jumlah skor ideal untuk seluruh *item* = $5 \times 100 = 500$. Jadi tingkat persetujuannya berdasarkan data tersebut = $(350 : 500) \times 100\% = 70\%$ dari yang diharapkan.

Secara kontium dapat digambarkan seperti pada Gambar 6.



Gambar 6. Contoh Tingkat Persetujuan Responden pada *Software*

Berdasarkan data yang diperoleh dari 100 responden maka skor 350 terletak pada daerah setuju.

Selain perhitungan tingkat persetujuannya, jawaban kuesioner USE juga dihitung *cronbach's alpha*-nya agar diketahui reliabilitas instrumen yang digunakan. Perhitungan *cronbach's alpha* menggunakan *tool* SPSS 20 dan kemudian hasil dari perhitungan yang didapatkan dibandingkan dengan tabel *internal consistency cronbach's alpha* seperti pada Tabel 8.

Tabel 8. *Internal Consistency Cronbach's Alpha* (Gliem & Gliem, 2003)

<i>Cronbach's Alpha</i>	<i>Internal Consistency</i>
$\alpha \geq .9$	<i>Excellent</i>
$.9 > \alpha \geq .8$	<i>Good</i>
$.8 > \alpha \geq .7$	<i>Acceptable</i>
$.7 > \alpha \geq .6$	<i>Questionable</i>
$.6 > \alpha \geq .5$	<i>Poor</i>
$.5 > \alpha$	<i>Unacceptable</i>

4. *Efficiency*

Pengujian pada *efficiency* dilakukan dengan pengujian *load* halaman *web* dengan menggunakan GTMetrix. GTMetrix akan menghasilkan waktu load dari halaman *web*. *Web* dikatakan baik apabila waktu *load*-nya setidaknya 10 detik menurut Nielsen (2010) dan setidaknya 7 detik sesuai rata-rata *load* *web* (Aptimize, 2010).

5. *Maintainability*

Pengujian *maintainability* menggunakan perhitungan *Maintainability Index* (MI). MI dihitung dari rumus yang terdiri dari *Lines of Code (LOC)*, *Cyclomatic Complexity (CC)*, dan *Halstead Volume (HV)*. Perhitungan LOC, CC, dan HV menggunakan PHPMetric. Hasil perhitungan tersebut kemudian dimasukkan ke dalam rumus *Maintainability Index* adalah sebagai berikut Coleman (1994):

$$MI = 171 - 5.2 \times \ln(aveVol) - 0.23 \times aveV(g') - 16.2 \times \ln(aveLOC) \\ + (50 \times \sin(\sqrt{2.46 \times perCM}))$$

Keterangan:

MI	= <i>Maintainability Index</i>
aveVol	= rata-rata <i>Halstead Volume (HV)</i>
ave V(g')	= rata-rata <i>Cyclomatic Complexity (CC)</i>
aveLOC	= rata-rata <i>Lines of Code (LOC)</i>
perCM	= persentase <i>comment</i> pada <i>source code</i>

Hasil perhitungan MI agar *software* dapat lolos uji *maintainability* adalah 65 seperti yang diungkapkan oleh Coleman (1994).

6. *Portability*

Pengujian pada aspek *portability* dilakukan dengan mengakses *software* dengan berbagai tujuh *browser desktop* dari tiga mayoritas *browser* yang digunakan yaitu *google chrome*, *Mozilla Firefox*, dan *Internet Explorer* dan lima *browser* perangkat *mobile* seperti yang diungkapkan oleh Salonen (2012). *Software* harus dapat diakses dengan berbagai macam *browser* tersebut agar lolos uji *portability*. Pengaksesan dilakukan dengan menggunakan *BrowseEmAll* yang dapat menjalakan *virtual web browser* untuk *desktop* dan *mobile*.

BAB IV

HASIL PENELITIAN DAN PEMBAHASAN

A. Tahap Analisis Kebutuhan

1. Analisis Kebutuhan Fungsi

Kebutuhan fungsi pada pengembangan *software tes* dengan analisis butir soal berbasis *web* di MAN Yogyakarta 1 adalah sebagai berikut:

- a. Guru dapat menyelenggarakan tes dengan memasukkan beberapa soal pilihan ganda dengan lima alternatif jawaban.
- b. Tes memiliki pengaturan KKM, waktu, dan petunjuk penggerjaan.
- c. Laporan hasil tes pada halaman guru berupa laporan per tes yang diselenggarakan dan laporan per siswa yang melakukan tes.
- d. Laporan hasil tes per tes memuat informasi : alokasi waktu, KKM, jumlah soal, nama peserta tes, waktu selesai, nilai, dan status ketuntasan.
- e. Laporan hasil tes per siswa memuat informasi : nilai, KKM, Status ketuntasan, alokasi waktu, lama penggerjaan, identitas peserta tes, soal berserta kunci jawaban, dan jawaban peserta tes.
- f. Laporan hasil analisis butir soal memuat informasi : soal, tingkat kesulitan, daya beda, alternatif jawaban tidak efektif, kesimpulan, dan persebaran jawaban.
- g. Sebelum melakukan tes, peserta tes memasukkan identitas yang berupa nama, kelas, dan sekolah.
- h. Laporan hasil tes untuk peserta tes berupa nilai, KKM, Status ketuntasan, alokasi waktu, lama penggerjaan, dan identitas peserta tes.

2. Analisis Kebutuhan *Software* dan *Hardware*

Tools (software) dan *hardware* yang digunakan dalam pengembangan *software* tes dengan analisis butir soal antara lain:

- a. PC / Laptop.
- b. *Framework* Laravel.
- c. XAMPP (*Apache & MySQL Server*).
- d. Visual Paradigm for UML, *software* yang digunakan untuk membuat diagram UML seperti *user case*, *class diagram*, *activity diagram*, dan *sequence diagram* pada tahap desain dalam pengembangan *software*.
- e. Balsamiq Mockup, *software* yang digunakan untuk membuat desain tampilan *software* pada tahap desain dalam pengembangan *software*.
- f. *Web Browser* (Internet Explorer, Google Chrome, dan Mozilla Firefox).

B. Tahap Desain

1. Diagram UML

a. *Use Case*

1) Definisi aktor

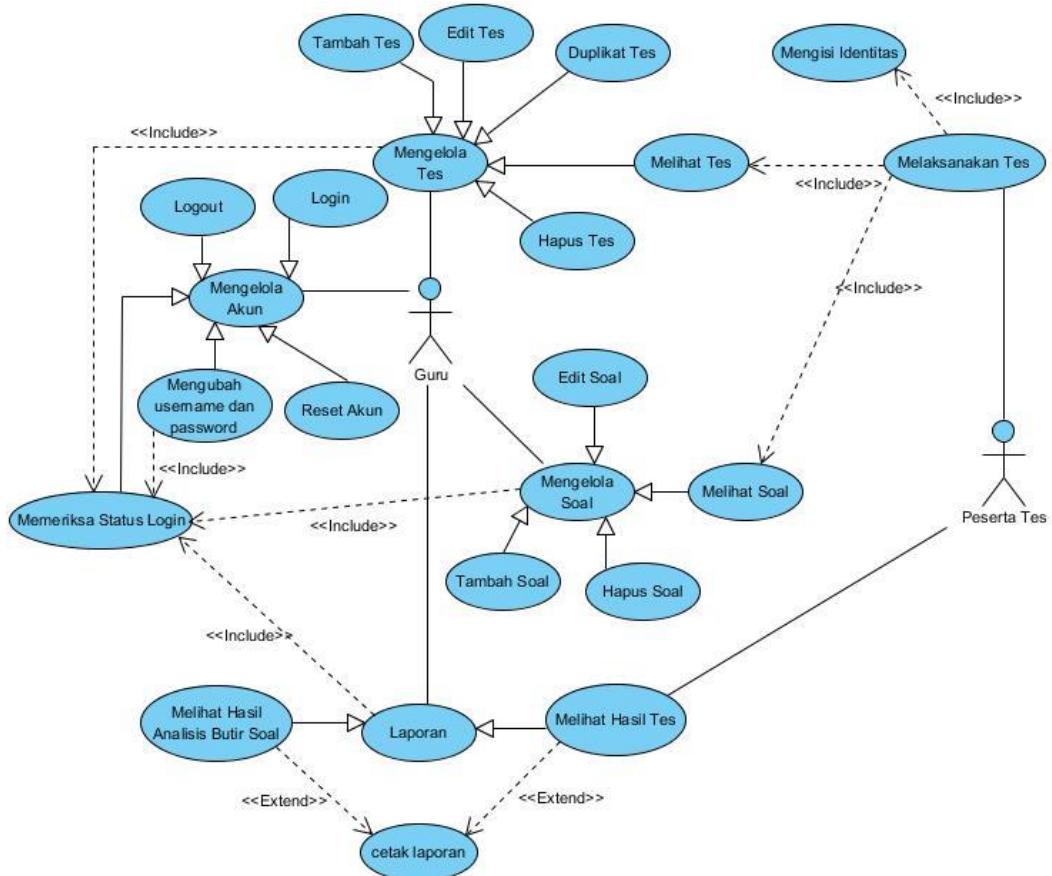
Deskripsi aktor untuk *use case software* tes dengan analisis butir soal terdapat pada Tabel 9.

Tabel 9. Definisi Aktor

No.	Aktor	Deskripsi
1	Guru	Pengguna yang bertanggungjawab sebagai <i>admin</i> . Pengguna tersebut memiliki hak akses sebagai <i>admin</i> yaitu mengelola tes, mengelola soal, dan melihat laporan baik hasil tes maupun hasil analisis butir soal.
2	Peserta Tes	Pengguna yang memiliki hak akses untuk melaksanakan tes. Dalam pelaksanaannya pengguna dapat melihat daftar tes yang ada, mengisi identitas untuk melaksanakan tes, melihat soal dan memasukkan jawaban, dan melihat hasil tes.

2) Diagram *Use Case*

Diagram *use case* software tes dengan analisis butir soal dapat dilihat pada Gambar 7. Pada diagram *use case*, aktor terdiri dari guru dan siswa. Aktor guru dapat mengelola tes, mengelola akun, mengelola soal, dan mengelola laporan. Aktor siswa dapat melaksanakan tes dan melihat hasil tes.

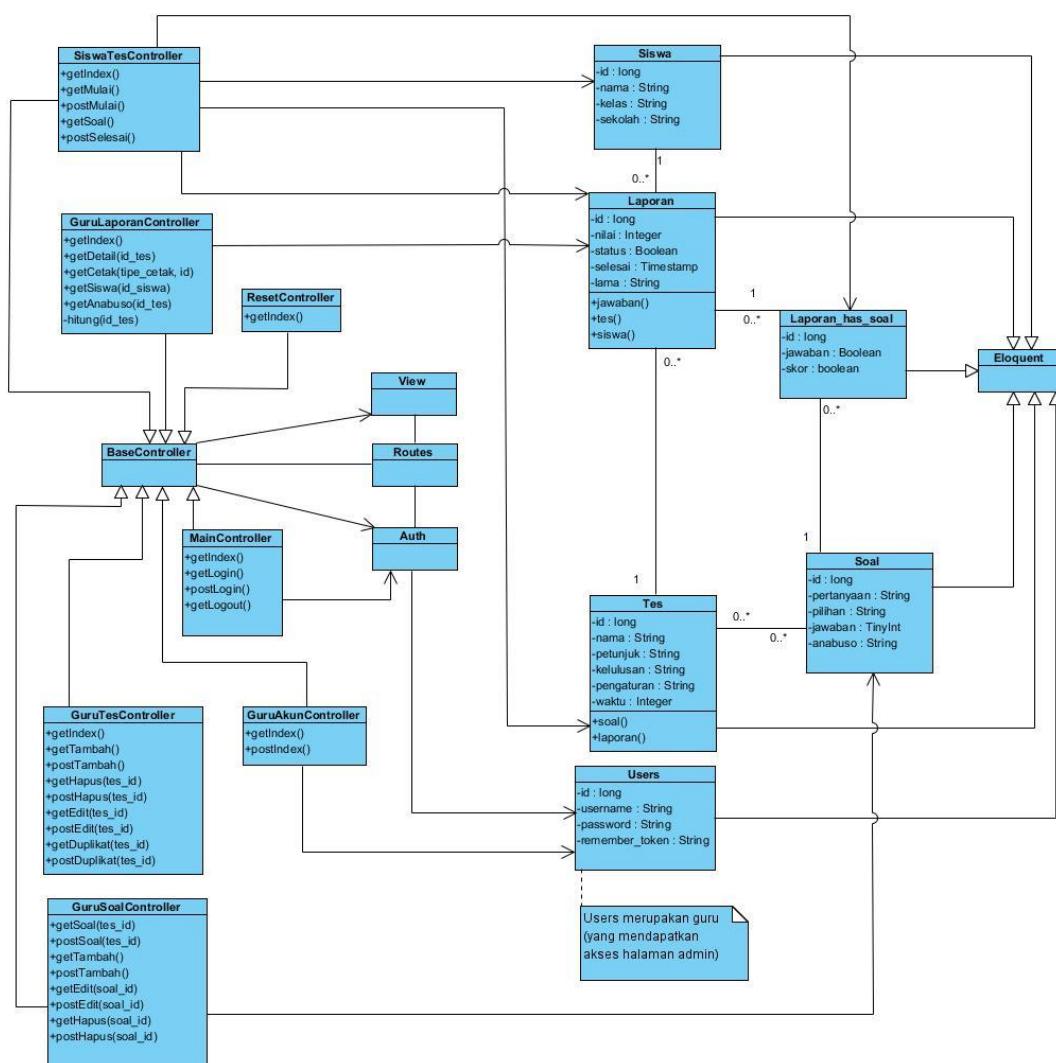


Gambar 7. *Use Case Diagram Software Tes dengan Analisis Butir Soal*

b. *Class Diagram*

Framework Laravel menggunakan *architecture pattern* MVC (Model View Controller). MVC Laravel berupa *class-class* yang telah ada di laravel yang bisa digunakan oleh pengembang untuk membantu pengembangan. *Class* model dapat

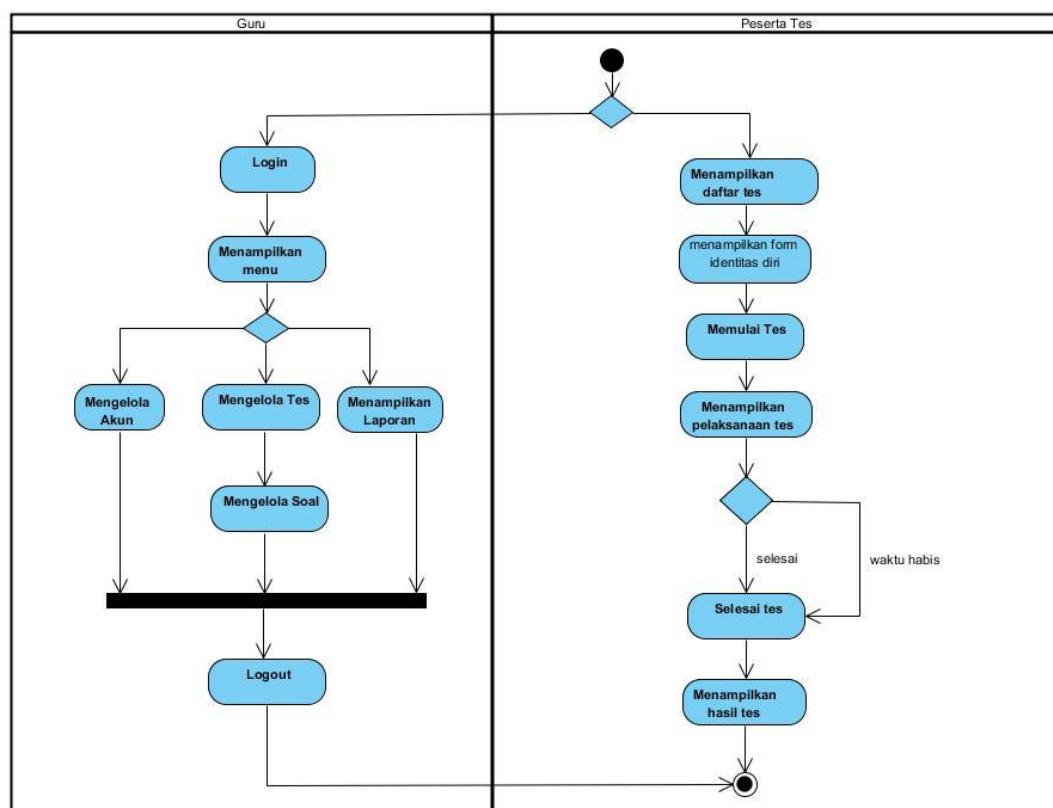
diakses melalui *class Eloquent*, *Controller* melalui *class BaseController*, dan *View* melalui *class View*. Selain itu Pada laravel semua *request* dari pengguna ditangani oleh *class Routes* sehingga *class* ini yang akan mengarahkan pengguna kepada suatu *controller*. Selain itu laravel memiliki kelas Auth yang dapat digunakan untuk melakukan manajemen *user* seperti *login*, *logout*, dan sebagainya. *Class-class* bawaan laravel tersebut mempengaruhi desain *class diagram*. Desain *class diagram* dapat dilihat pada Gambar 8.



Gambar 8. *Class Diagram Software Tes dengan Analisis Butir Soal*

c. Activity Diagram

Pengguna pada *software tes dengan analisis butir soal* memiliki dua pengguna yaitu peserta tes dan guru. Aktivitas masing-masing pengguna tentunya memiliki bagian halaman yang berbeda. Untuk mempermudah pemahaman *activity diagram* digunakan *swimlane* yang membedakan bagian kedua pengguna tersebut. Gambar 9 menggambarkan *activity diagram* untuk *software tes dengan analisis butir soal*.

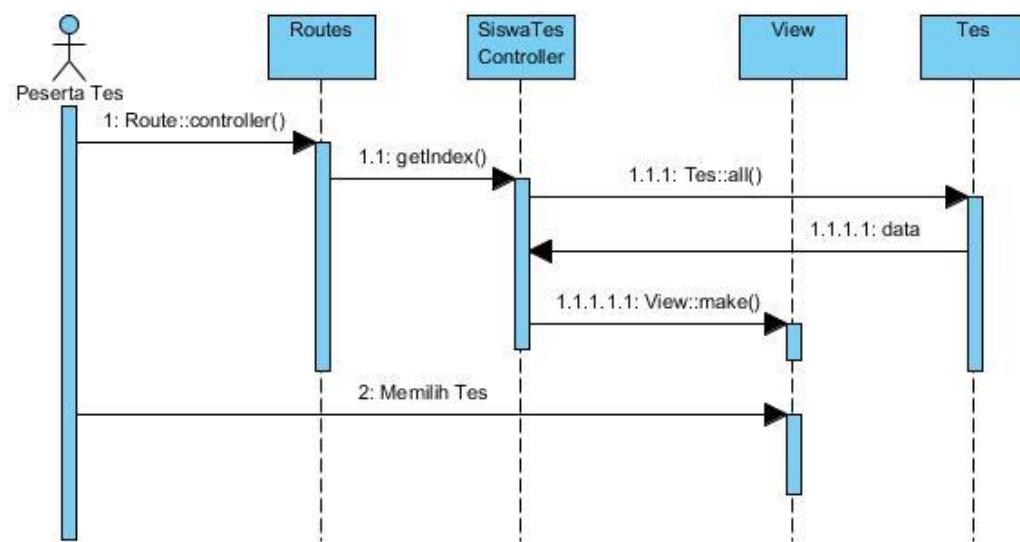


Gambar 9. *Activity Diagram Software Tes dengan Analisis Butir Soal*

d. *Sequence Diagram*

1) Memilih Tes

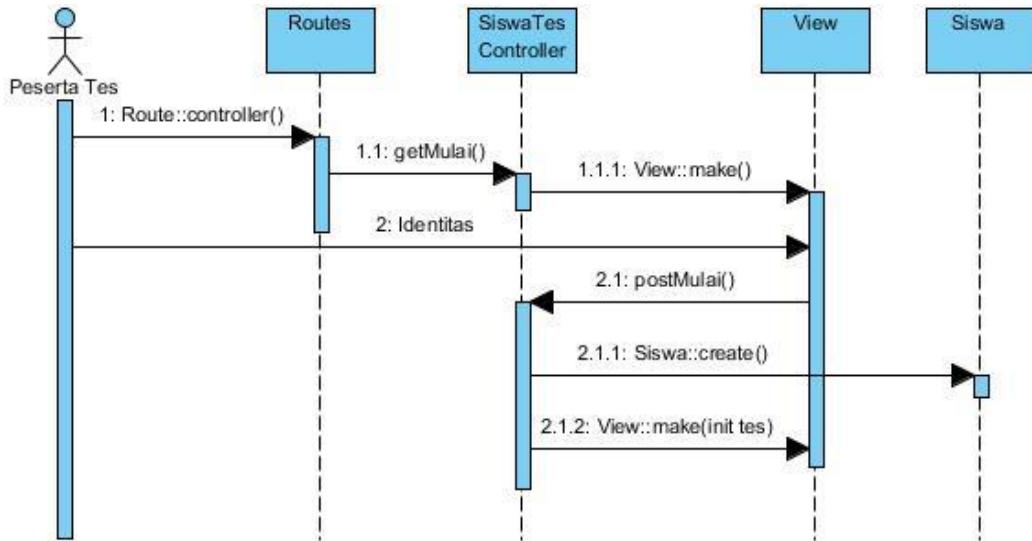
Gambar 10 merupakan *sequence diagram* untuk fungsi memilih tes. Pada *sequence diagram* memilih tes, yang berperan sebagai aktor adalah peserta tes. *Class* yang terlibat pada *sequence diagram* ini adalah Routes, SiswaTesController, View, dan Tes. *Class* SiswaTesController berperan sebagai *controller* untuk fungsi memilih tes dan *class* Tes berperan sebagai model yang terhubung ke *database*.



Gambar 10. *Sequence Diagram* Memilih Tes

2) Mengisi Identitas

Gambar 11 merupakan *sequence diagram* untuk fungsi mengisi identitas. Pada *sequence diagram* mengisi identitas, yang berperan sebagai aktor adalah peserta tes. *Class* yang terlibat pada *sequence diagram* ini adalah Routes, SiswaTesController, View, dan Siswa. *Class* SiswaTesController berperan sebagai *controller* untuk fungsi mengisi identitas dan *class* Siswa berperan sebagai model yang terhubung ke *database*.

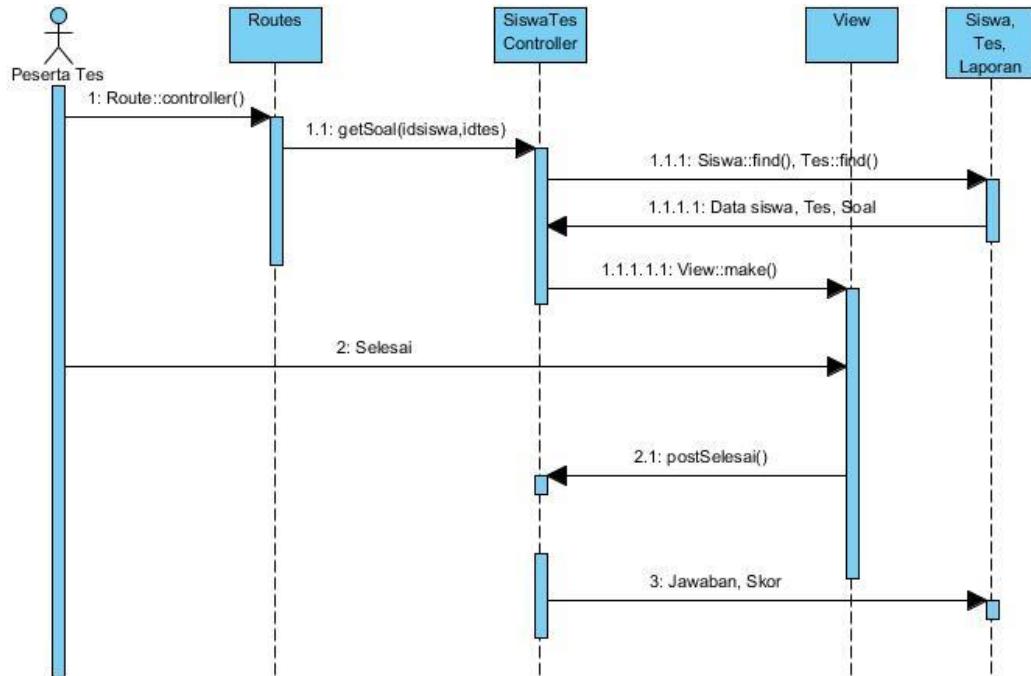


Gambar 11. *Sequence Diagram* Mengisi Identitas

3) Melaksanakan Tes

Gambar 12 merupakan *sequence diagram* untuk fungsi melaksanakan tes.

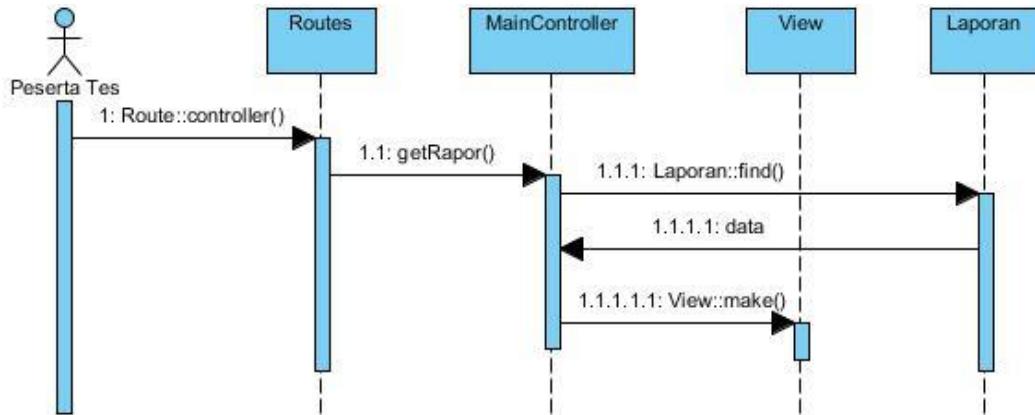
Pada *sequence diagram* melaksanakan tes, yang berperan sebagai aktor adalah peserta tes. *Class* yang terlibat pada *sequence diagram* ini adalah Routes, SiswaTesController, View, Siswa, Tes, dan Laporan. *Class* SiswaTesController berperan sebagai *controller* untuk fungsi melaksanakan tes dan *class* Siswa, Tes, dan Laporan berperan sebagai model yang terhubung ke *database*.



Gambar 12. *Sequence Diagram* Melaksanakan Tes

4) Melihat Hasil Tes

Gambar 13 merupakan *sequence diagram* untuk fungsi melihat hasil tes. Pada *sequence diagram* melihat hasil tes, yang berperan sebagai aktor adalah peserta tes. *Class* yang terlibat pada *sequence diagram* ini adalah Routes, MainController, View, dan Laporan. *Class* MainController berperan sebagai *controller* untuk fungsi melihat hasil tes dan *class* Laporan berperan sebagai model yang terhubung ke *database*.

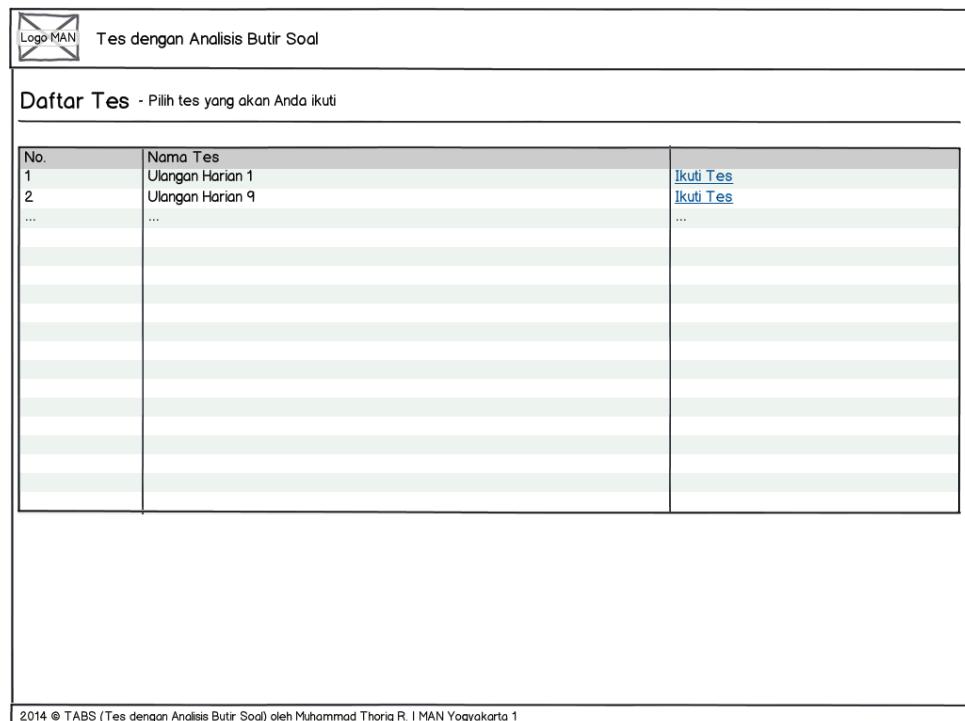


Gambar 13. *Sequence Diagram* Melihat Hasil Tes

2. Desain *Interface*

1) Halaman Peserta Tes

a) Daftar Tes



Gambar 14. Desain Halaman Peserta Tes Bagian Daftar Tes

Pada *interface* halaman daftar tes terdapat tampilan tabel dengan kolom no, nama tes, dan *link* untuk mengikuti tes.

b) Identitas Peserta Tes

Logo MAN Tes dengan Analisis Butir Soal

{Nama Tes}

Ketentuan

Alokasi Waktu
KKM
Jumlah Soal

Petunjuk

Identitas Peserta Tes

Nama

Kelas

Sekolah

Mulai Tes

2014 © TABS (Tes dengan Analisis Butir Soal) oleh Muhammad Thoriq R. | MAN Yogyakarta 1

Gambar 15. Desain Halaman Peserta Tes Bagian Identitas Peserta Tes

Pada *interface* halaman identitas peserta tes, terdapat informasi tes yang berupa alokasi waktu, KKM, jumlah soal, dan petunjuk. Selain itu juga terdapat isian identitas peserta tes yang terdiri dari nama, kelas, dan sekolah.

c) Pelaksanaan Tes

The screenshot shows a window titled 'Tes dengan Analisis Butir Soal'. At the top left is the 'Logo MAN'. The main area displays a question: '1. Perhatikan gambar berikut!' followed by a square containing an 'X'. Below the question is the text 'Bentuk di atas disebut....' and a list of five options (A-E) for the student to choose from. The right side of the window shows a 'Soal' column and a 'Sisa Waktu' column. At the bottom right is a 'Selesai' button.

Gambar 16. Desain Halaman Peserta Tes Bagian Pelaksanaan Tes

Pada *interface* pelaksanaan tes, ditampilkan soal, pilihan jawaban, waktu penggerjaan, dan tombol selesai.

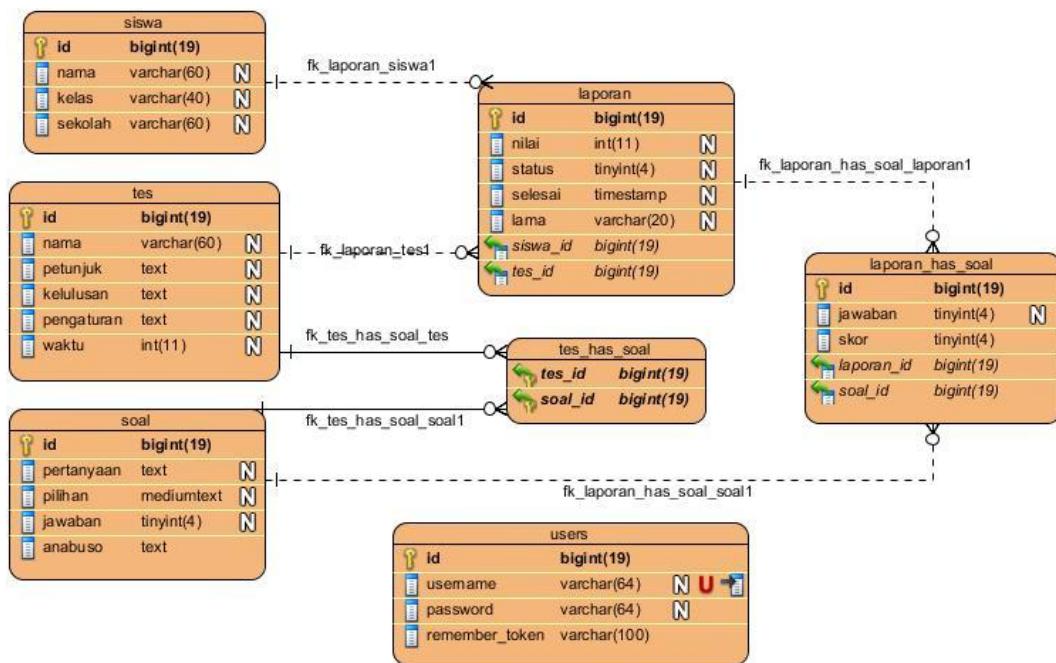
d) Hasil Tes

The screenshot shows a window titled 'Tes dengan Analisis Butir Soal'. At the top left is the 'Logo MAN'. The main area displays a header 'Hasil : {Nama Tes}' and two sections: 'Informasi Tes' and 'Identitas'. The 'Informasi Tes' section contains fields for 'Nilai', 'KKM', 'Status Ketuntasan', 'Alokasi Waktu', and 'Lama Pengerojan'. The 'Identitas' section contains fields for 'Nama', 'Kelas', and 'Sekolah'. At the bottom right is a 'Selesai' button.

Gambar 17. Desain Halaman Peserta Tes Bagian Hasil Tes

Pada *interface* halaman hasil tes ditampilkan informasi tes yang terdiri dari nama tes, nilai, KKM, status ketuntasan, alokasi waktu, dan lama penggeraan. Selain itu juga terdapat identitas peserta tes yang terdiri dari nama, kelas, dan sekolah.

3. Desain *Entity Relational Diagram*

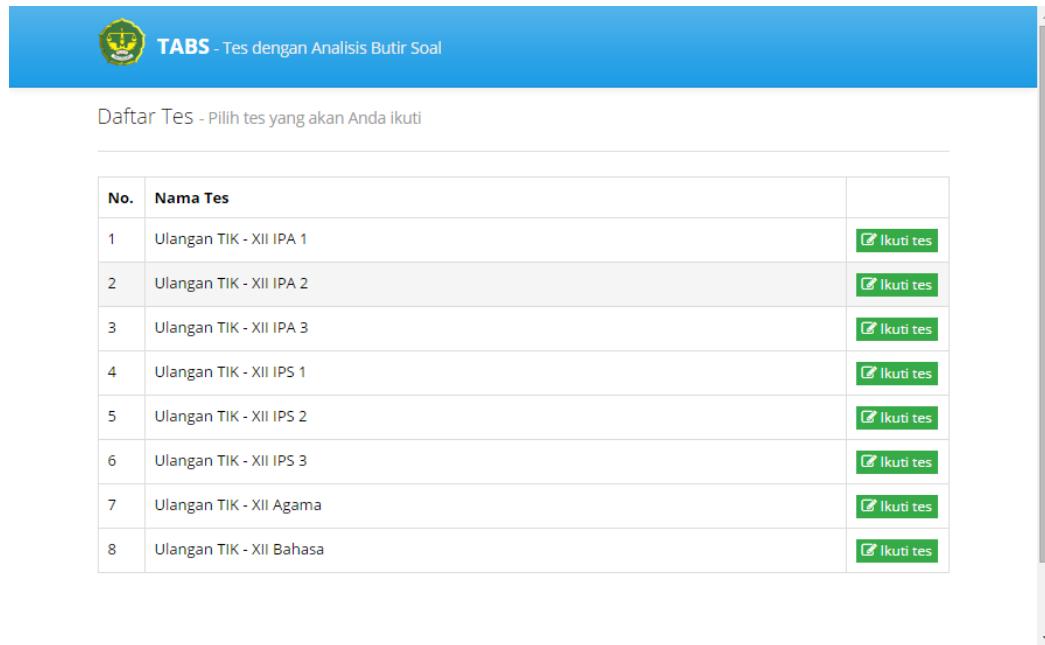


Gambar 18. Desain *ERD Software Tes dengan Analisis Butir Soal*

Pada desain *Entity Relational Diagram* (ERD), tabel terdiri dari siswa, tes, soal, laporan, tes_has_soal, laporan_has_soal, dan users. Siswa merupakan tabel yang menyimpan data peserta tes. Tes merupakan tabel yang menyimpan data tes. Soal merupakan tabel yang menyimpan data soal yang digunakan untuk tes serta hasil analisis butir soal. Laporan merupakan tabel yang menyimpan data hasil tes. Users merupakan tabel yang digunakan untuk menyimpan data *username* dan *password* guru untuk login ke *software*. Tes_has_soal dan laporan_has_soal merupakan tabel yang berperan sebagai penghubung.

C. Implementasi

1. Halaman Peserta Tes
 - a. Halaman Daftar Tes



No.	Nama Tes	
1	Ulangan TIK - XII IPA 1	<input checked="" type="checkbox"/> Ikuti tes
2	Ulangan TIK - XII IPA 2	<input checked="" type="checkbox"/> Ikuti tes
3	Ulangan TIK - XII IPA 3	<input checked="" type="checkbox"/> Ikuti tes
4	Ulangan TIK - XII IPS 1	<input checked="" type="checkbox"/> Ikuti tes
5	Ulangan TIK - XII IPS 2	<input checked="" type="checkbox"/> Ikuti tes
6	Ulangan TIK - XII IPS 3	<input checked="" type="checkbox"/> Ikuti tes
7	Ulangan TIK - XII Agama	<input checked="" type="checkbox"/> Ikuti tes
8	Ulangan TIK - XII Bahasa	<input checked="" type="checkbox"/> Ikuti tes

Gambar 19. Halaman Daftar Tes

Halaman daftar tes berisi tentang tes-tes apa saja yang bisa diikuti oleh siswa.

Tes yang tampil merupakan tes yang telah dibuat oleh guru di halaman *admin*.

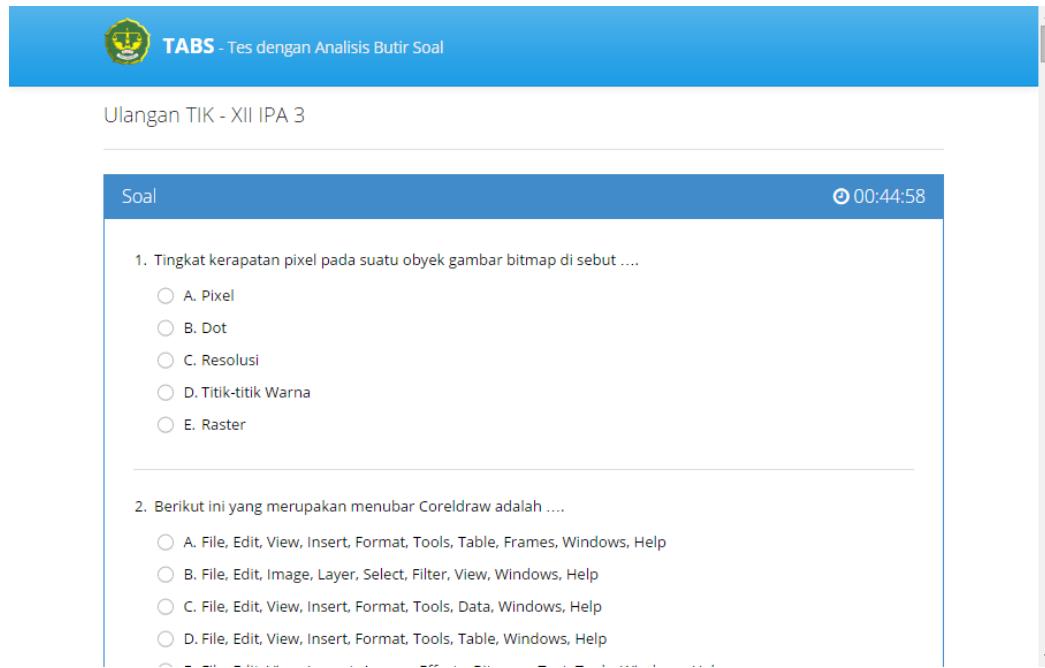
b. Halaman Identitas Peserta Tes

The screenshot shows the TABS (Tes dengan Analisis Butir Soal) test interface. At the top, there is a blue header bar with the TABS logo and the text "TABS - Tes dengan Analisis Butir Soal". Below the header, the text "Ulangan TIK - XII IPA 3" is displayed. The main content area is divided into two sections: "Ketentuan" (left) and "Petunjuk" (right). The "Ketentuan" section contains the following text: "Alokasi waktu :45 menit", "Nilai kelulusan :75", and "Jumlah soal :35". The "Petunjuk" section contains the text: "Berdo'alah sebelum mengerjakan. Selamat mengerjakan! Semoga berhasil!". Below these sections, there is a form titled "Identitas Peserta Tes" with fields for "Nama", "Kelas", and "Sekolah", each with an input box. A green "Mulai Tes" button is located at the bottom right of the form. At the bottom of the page, there is a dark footer bar with the text "2014 © TABS (Tes dengan Analisis Butir Soal) oleh Muhammad Thoriq R. | Bantuan" and a small circular icon.

Gambar 20. Halaman Identitas Peserta Tes

Halaman identitas peserta tes digunakan oleh siswa untuk mengisi identitas dirinya sebelum melaksanakan tes. Informasi yang harus diisi siswa adalah nama, kelas, dan sekolah.

c. Halaman Pelaksanaan Tes



The screenshot shows a web-based test application. At the top, there is a blue header bar with the logo of a university or institution on the left and the text "TABS - Tes dengan Analisis Butir Soal" in the center. Below the header, the title "Ulangan TIK - XII IPA 3" is displayed. The main content area is titled "Soal" and shows a timer "00:44:58" in the top right corner. The test consists of two questions, each with five multiple-choice options (A, B, C, D, E). Question 1 asks about the density of pixels in a bitmap image, with options A through E. Question 2 asks what menu bar is present in CorelDRAW, with options A through D.

Gambar 21. Halaman Pelaksanaan Tes

Halaman pelaksanaan tes merupakan halaman yang digunakan siswa untuk menjawab soal-soal tes. Pada halaman ini terdapat *timer* yang apabila habis waktunya maka jawaban siswa yang sudah ada akan terkirim ke *server* dan otomatis tes yang sedang berlangsung akan selesai.

d. Halaman Hasil Tes

The screenshot shows a results page from the TABS (Tes dengan Analisis Butir Soal) system. At the top, there is a logo and the text "TABS - Tes dengan Analisis Butir Soal". Below this, the text "Hasil : Ulangan TIK - XII IPA 3" is displayed. The page is divided into two main sections: "Informasi Tes" (Test Information) and "Identitas" (Identity).
Informasi Tes (Test Information):

Nilai	:	74
KKM	:	75
Status Ketuntasan	:	Tidak Tuntas
Alokasi Waktu	:	00:45:00
Lama Pengerjaan	:	00:00:44

Identitas (Identity):

Nama	:	Darojat
Kelas	:	12 IPA 2
Sekolah	:	MAN Yogyakarta 1

Gambar 22. Halaman Hasil Tes

Halaman hasil tes merupakan halaman yang menampilkan perolehan nilai yang didapatkan siswa setelah melaksanakan tes. Pada halaman ini siswa dapat melihat informasi tes dan identitas dirinya.

D. Pengujian

1. *Functionality*

Hasil pengujian *functionality* yang dilakukan oleh tiga orang yang bekerja di PT. Sebangsa Bersama yang merupakan perusahaan yang bergerak di bidang IT *software* dapat dilihat pada Tabel 10.

Tabel 10. Hasil Pengujian *Functionality*

No. Pernyataan	Ya	Tidak
1	3	0
2	3	0
3	3	0
4	3	0
5	3	0
6	3	0
7	3	0
8	3	0
9	3	0
10	3	0
11	3	0
12	3	0
13	3	0
14	3	0
15	3	0
16	3	0
17	3	0
18	3	0
19	3	0
20	3	0
21	3	0
22	3	0
23	3	0
24	3	0
25	3	0
Total	75	0

Pengujian *functionality* didapatkan hasil bahwa semua fungsi berjalan dengan baik.

Selain pengujian tersebut pengujian fungsi untuk pada analisis butir soal juga dibuktikan dengan perhitungan manual menggunakan *software* microsoft excel.

Salah satu hasil perhitungan analisis butir soal pada *software* tes dengan analisis butir soal dapat dilihat pada Gambar 23.

No.	Soal	Tingkat Kesulitan		Daya Beda		Alternatif Jawaban Tidak Efektif	Kesimpulan
		Koefisien	Ket.	Koefisien	Ket.		
1	Dalam grafis yang disebut Raster adalah	0.6	Sedang	0.2	Cukup	-	Baik

Persentase Jawaban

No. Soal	Pilihan 1	Pilihan 2	Pilihan 3	Pilihan 4	Pilihan 5	Tidak menjawab
1	60 %	17 %	7 %	10 %	7 %	0 %

Gambar 23. Contoh Hasil Analisis Butir Soal

Pada hasil perhitungan didapatkan bahwa koefisien tingkat kesulitan 0,6 dan daya beda 0,2. Selain itu hasil pola jawaban untuk pilihan jawaban 1 sebesar 80%, pilihan 2 sebesar 70%, pilihan 3 sebesar 7%, pilihan 4 sebesar 10%, dan pilihan 5 sebesar 7%. Dengan cara melakukan input jawaban yang sama dengan jawaban yang ada pada *software*, butir soal dianalisis dengan menggunakan *microsoft excel* yang dihitung dengan rumus perhitungan tingkat kesulitan, daya beda, dan pola jawaban soal. Gambar 24 dan Gambar 25 merupakan hasil perhitungan analisis butir soal pada *microsoft excel*. Perhitungan ini menunjukkan hasil yang sama dengan perhitungan pada *software* tes dengan analisis butir soal.

AK	AL	AM	AN	AO	AP	AQ
No. soal	1	2	3	4	5	6
Tingkat Kesulitan						
	0,6	0,33333	0,5	0,63333	0,7	0,7
	Sedang	Sedang	Sedang	Sedang	Sedang	Sedang
Daya Beda						
	0,2	0,26667	-0,2	0,46667	0,6	0,6
	Cukup	Cukup	Jelek	Baik	Baik	Baik

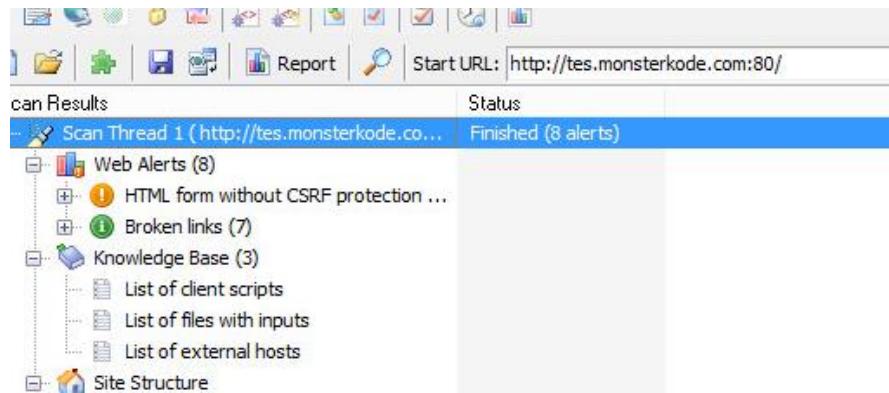
Gambar 24. Perhitungan Tingkat Kesulitan dan Daya Beda Manual

Z	AA	AB	AC	AD	AE	AF	AG
no.soal	1	2	3	4	5	6	7
Pilihan Jawaban							
1	60	33,3	6,67	6,67	6,67	6,67	6,67
2	16,7	20	50	63,3	6,67	6,67	10
3	6,67	20	20	6,67	70	70	10
4	10	13,3	13,3	10	0	6,67	66,7
5	6,67	13,3	10	10	10	10	6,67
Total	100	100	100	96,7	93,3	100	100
OMIT	0	0	0	3,33	6,67	0	0

Gambar 25. Perhitungan Persentase Pilihan Jawaban dengan *Microsoft Excel*

Pada aspek *security software* diuji dengan menggunakan *software Acunetix Web Vulnerability Scanner*. Berikut adalah hasil pengujinya:

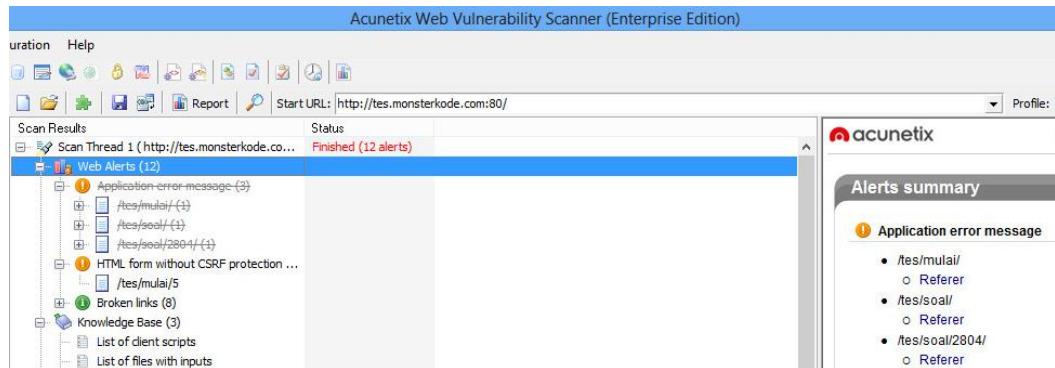
a. Hasil *Scan Serangan SQL Injection*



Gambar 26. Hasil *Scan Serangan SQL Injection*

Pada bagian *Web Alert* tidak terdapat *application error message* terhadap serangan *SQL Injection* sehingga *software* aman dari serangan *SQL Injection*.

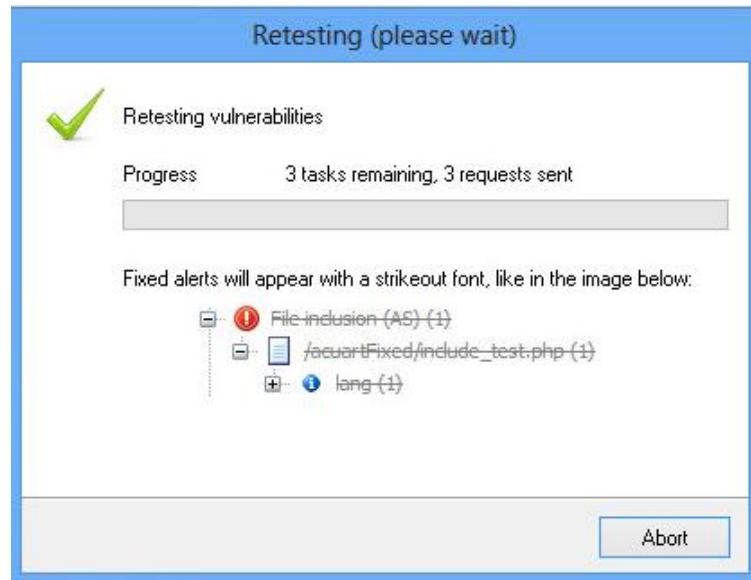
b. Hasil *Scan Serangan XSS*



Gambar 27. Hasil *Scan Serangan XSS*

Pada Gambar 27 terdapat *application error message* berjumlah tiga, yang merupakan letak serangan *XSS*. Masalah celah keamanan ini telah diperbaiki. Perbaikan ini ditunjukkan pada Gambar 27 bagian *application error message* telah dicoret. Pada Gambar 28 yang menyatakan bahwa "fixed alerts will appear with a

strikeout font". Jadi *application error message* telah dicoret merupakan celah keamanan yang telah diperbaiki.



Gambar 28. *Retesting Vulnerabilities*

2. *Reliability*

Hasil pengujian menggunakan *software* WAPT dapat dilihat pada Gambar 29.

Test execution parameters:

Test status: finished
Test started at: 10/11/2014 10:50:24
Scenario name: tes.wps
Test run comment:
Test executed by: thoriq (MONSTER)
Test executed on: localhost
Test duration: 0:10:00

Summary

Profile	Successful sessions	Failed sessions	Successful pages	Failed pages	Successful hits	Failed hits	Total KBytes sent
tes	56	0	1465	0	3033	0	3978

Gambar 29. Hasil Pengujian *Reliability* WAPT

Pada hasil pengujian dapat dilihat bahwa terdapat hasil pengujian *stress testing* yang berupa keberhasilan *sessions*, *pages*, dan *hits*. Dari hasil tersebut terlihat bahwa hasil *successful sessions* 56 dan 0 *failed sessions*, *successful pages* 1465 dan 0 *failed pages*, serta *successful hits* 3033 dan 0 *failed hits*.

3. *Usability*

Pengujian *usability* dengan menggunakan kuesioner USE yang diisi oleh 30 responden yang terdiri dari 28 siswa dan 2 guru TIK di MAN Yogyakarta 1. Jawaban responden dapat dilihat pada Tabel 11.

Tabel 11. Hasil Pengujian *Usability*

Pertanyaan	SS	TS	RG	ST	STS
1	5	18	6	1	0
2	5	19	6	0	0
3	6	21	3	0	0
4	6	16	6	2	0
5	3	21	6	0	0
6	3	18	9	0	0
7	3	14	12	1	0
8	3	14	12	1	0
9	3	18	9	0	0
10	2	23	5	0	0
11	3	16	11	0	0
12	1	13	15	1	0
13	2	23	5	0	0
14	2	9	12	7	0
15	3	8	10	6	3
16	0	7	21	1	1
17	2	11	16	1	0
18	2	9	18	1	0
19	2	7	20	1	0
20	2	19	8	1	0
21	2	15	9	4	0
22	4	14	11	1	0
23	4	10	13	3	0
24	4	12	14	0	0
25	4	13	11	2	0
26	5	19	5	1	0
27	3	15	12	0	0
28	0	20	9	1	0
29	2	21	6	1	0
30	5	19	4	1	1
Total	91	462	304	38	5

Perhitungan *cronbach's alpha* hasil pengujian *usability* dengan menggunakan *tool* SPSS 20 didapatkan hasil 0,934 seperti yang terlihat pada Gambar 30. Hasil 0,934 masuk dalam kategori *excellent* jika dibandingkan dengan Tabel 8 (*internal consistency cronbach's alpha*).

Case Processing Summary		
	N	%
Cases	30	100,0
Valid	30	,0
Excluded ^a	0	
Total	30	100,0

a. Listwise deletion based on all variables in the procedure.

Reliability Statistics	
Cronbach's Alpha	N of Items
,934	30

Gambar 30. Hasil Perhitungan *Cronbach's Alpha*

4. *Efficiency*

a. Hasil Pengujian *Efficiency* Halaman *Login*

Performance Report for: <http://tes.monsterkode.com/login>

Report generated: Wednesday, November 5, 2014, 8:11 PM -0800

Test Server Region: Vancouver, Canada

Using: Firefox (Desktop) 25.0.1, Page Speed 1.12.16, YSlow 3.1.8



Page load time: 0.73s | Total page size: 378KB | Total number of requests: 40

Gambar 31. Laporan GTMetrix Halaman *Login*

Pengujian *efficiency* halaman login pada Gambar 31, *grade* yang didapatkan adalah A (98%) untuk Page Speed dan B (84%) untuk Yslow. *Grade* ini menunjukkan bahwa tingkat efisiensi *resource software* yang diujikan sudah berada di atas rata-rata *grade* GTMetrix (*grade* GTMetrix untuk Page Speed dan YSlow bernilai 79%). Waktu untuk *load* halaman 0,73 detik, yang berarti sudah memenuhi standar Optimize (2010) karena waktu di bawah 7 detik.

b. Hasil Pengujian *Efficiency* Halaman Pengelolaan Tes

Performance Report for:

<http://tes.monsterkode.com/tes>

Report generated: Wednesday, November 5, 2014, 7:08 PM -0800

Test Server Region: Vancouver, Canada

Using: Firefox (Desktop) 25.0.1, Page Speed 1.12.16, YSlow 3.1.8



Gambar 32. Laporan GTMetrix Halaman Pengelolaan Tes

Pengujian *efficiency* pengelolaan tes pada Gambar 32, *grade* yang didapatkan adalah A (93%) untuk Page Speed dan B (83%) untuk Yslow. *Grade* ini menunjukkan bahwa tingkat efisiensi *resource software* yang diujikan sudah berada di atas rata-rata *grade* GTMetrix (*grade* GTMetrix untuk Page Speed dan YSlow bernilai 79%). Waktu untuk *load* halaman 2,09 detik, yang berarti sudah memenuhi standar Optimize (2010) karena waktu di bawah 7 detik.

c. Hasil Pengujian *Efficiency* Halaman *Form* Tes

Performance Report for:
<http://tes.monsterkode.com/guru/tes/tambah>

Report generated: Tuesday, November 4, 2014, 9:15 PM -0800

Test Server Region: Vancouver, Canada

Using: Firefox (Desktop) 25.0.1, Page Speed 1.12.16, YSlow 3.1.8

Advanced options: Cookies



Gambar 33. Laporan GTMetrix Halaman *Form* Tes

Pengujian *efficiency* halaman *form* tes pada Gambar 33, *grade* yang didapatkan adalah A (95%) untuk Page Speed dan B (84%) untuk Yslow. *Grade* ini menunjukkan bahwa tingkat efisiensi *resource software* yang diujikan sudah berada di atas rata-rata *grade* GTMetrix (*grade* GTMetrix untuk Page Speed dan YSlow bernilai 79%). Waktu untuk *load* halaman 1,85 detik, yang berarti sudah memenuhi standar Optimize (2010) karena waktu di bawah 7 detik.

d. Hasil Pengujian *Efficiency* Halaman Pengelolaan Soal

Performance Report for:
<http://tes.monsterkode.com/guru/tes/soal/15>

Report generated: Tuesday, November 4, 2014, 9:20 PM -0800

Test Server Region: Vancouver, Canada

Using: Firefox (Desktop) 25.0.1, Page Speed 1.12.16, YSlow 3.1.8

Advanced options: Cookies



Gambar 34. Laporan GTMetrix Halaman Pengelolaan soal

Pengujian *efficiency* halaman pengelolaan soal pada Gambar 34, *grade* yang didapatkan adalah A (95%) untuk Page Speed dan B (83%) untuk Yslow. *Grade* ini menunjukkan bahwa tingkat efisiensi *resource software* yang diujikan sudah berada di atas rata-rata *grade* GTMetrix (*grade* GTMetrix untuk Page Speed dan YSlow bernilai 79%). Waktu untuk *load* halaman 0,90 detik, yang berarti sudah memenuhi standar Optimize (2010) karena waktu di bawah 7 detik.

e. Hasil Pengujian *Efficiency* Halaman *Form* Soal

Performance Report for:

<http://tes.monsterkode.com/guru/tes/soal/edit/362>

Report generated: Tuesday, November 4, 2014, 9:24 PM -0800

Test Server Region: Vancouver, Canada

Using: Firefox (Desktop) 25.0.1, Page Speed 1.12.16, YSlow 3.1.8

Advanced options: Cookies



Gambar 35. Laporan GTMetrix Halaman *Form* Soal

Pengujian *efficiency* halaman *form* soal pada Gambar 31, *grade* yang didapatkan adalah A (94%) untuk Page Speed dan B (84%) untuk Yslow. *Grade* ini menunjukkan bahwa tingkat efisiensi *resource software* yang diujikan sudah berada di atas rata-rata *grade* GTMetrix (*grade* GTMetrix untuk Page Speed dan YSlow bernilai 79%). Waktu untuk *load* halaman 1,06 detik, yang berarti sudah memenuhi standar Optimize (2010) karena waktu di bawah 7 detik.

Hasil pengujian *efficiency* seluruhnya dapat dilihat pada Tabel 12.

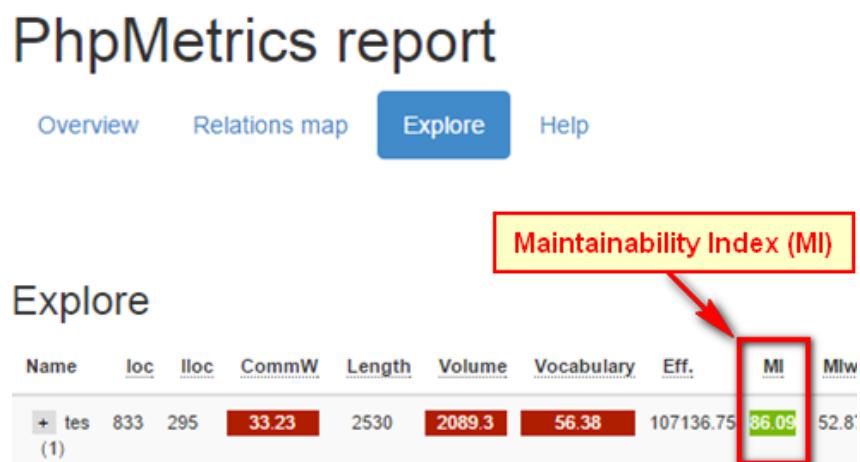
Tabel 12. Hasil Pengujian *Efficiency*

No.	Halaman web	Page Load Time (detik)	Page Size (Kb)	Page Speed Grade	Yslow Grade
1	Login	0,73	378	A (98%)	B (84%)
2	Tes	2,09	305	A (93%)	B (83%)
3	Form tes	1,85	412	A (95%)	B (84%)
4	Soal	0,90	418	A (95%)	B (83%)
5	Form soal	1,06	559	A (94%)	B (84%)
6	Laporan	1,08	413	A (95%)	B (84%)
7	Hasil tes	1,42	415	A (95%)	B (84%)
8	Detail hasil tes	1,26	466	A (95%)	B (84%)
9	Analisis butir soal	3,55	467	A (95%)	B (84%)
10	Daftar tes	2,09	305	A (93%)	B (83%)
11	Identitas peserta tes	0,73	305	A (94%)	B (84%)
12	Pelaksanaan tes	4,03	507	A (92%)	C (78%)
13	Hasil tes peserta	0,72	254	A (94%)	B (84%)
Rata-rata		1,65	400	A (94%)	B (83%)

Hasil pengujian *efficiency* pada Tabel 12 menunjukkan bahwa rata-rata waktu *page load* halaman *web* 1,65 detik. Selain itu masing-masing *page load time* tidak ada yang melebihi 7 detik (sesuai standar *efficiency* menurut Optimize).

5. *Maintainability*

Gambar 36 merupakan hasil pengujian *Maintainability* dengan menggunakan *PhpMetrics*.



Gambar 36. *PhpMetrics Report – Maintainability Index*

Dari hasil pengujian menunjukkan bahwa *Maintainability Index* (MI) memiliki nilai 86,09. Coleman (1994) menyatakan bahwa "*All components above the 85 maintainability index are highly maintainable, components between 85 and 65 are moderately maintainable, and components below 65 are difficult to maintain*". Jadi karena MI bernilai 86,09 maka hasil perhitungan MI tersebut masuk dalam kategori *highly maintainable*.

6. *Portability*

Berikut adalah hasil pengujian *portability* dengan menggunakan BrowseEmAll.

a. Hasil Pengujian *Portability Browser* Berbasis *Desktop*



BrowseEmAll cross-browser report

Tested: http://localhost/tes/tes	
No errors found	
Internet Explorer 10	0 Errors, 0 warning(s)
Firefox 31	0 Errors, 0 warning(s)
Firefox 30	0 Errors, 0 warning(s)
Firefox 29	0 Errors, 0 warning(s)
Chrome 36	0 Errors, 0 warning(s)
Chrome 35	0 Errors, 0 warning(s)
Chrome 34	0 Errors, 0 warning(s)

Gambar 37. Laporan BrowseEmAll pada Pengujian *Portability Desktop*

Dari Gambar 37 menunjukkan bahwa pada pengujian dengan menggunakan tujuh *browser* berbasis *desktop* tidak terjadi *error*.

b. Hasil Pengujian *Portability Browser* Berbasis *Mobile*



BrowseEmAll cross-browser report

Tested: http://localhost/tes/tes	
No errors found	
iPad 4 (iOS 7)	0 Errors, 0 warning(s)
iPhone 5S (iOS 7)	0 Errors, 0 warning(s)
Android 4.4	0 Errors, 0 warning(s)
Android 4.3	0 Errors, 0 warning(s)
Android 4.4 tablet	0 Errors, 0 warning(s)

Gambar 38. Laporan BrowseEmAll pada Pengujian *Portability Mobile*

Gambar 38 menunjukkan bahwa pada pengujian dengan menggunakan lima *browser* berbasis *mobile* tidak terjadi *error*.

E. Pembahasan Hasil Penelitian

1. *Functionality*

Hasil pengujian *functionallity* yang ada pada Tabel 10 dihitung dengan menggunakan rumus ISO (2003) pada aspek *functionality*.

$$X = 1 - \frac{A}{B}$$

$$X = 1 - \frac{0}{75}$$

$$X = 1 - 0$$

$$X = 1$$

ISO 9126 dalam aspek *functionality* menyatakan bahwa *software* dikatakan baik apabila hasil perhitungannya semakin mendekati 1. Karena hasil pengujian memiliki nilai maksimal yaitu 1 maka *software* sudah memenuhi aspek *functionality*.

Hasil pengujian aspek *security* menunjukkan bahwa tidak ada celah serangan *SQL Injection* dan *XSS*. Pada aspek *security* website harus aman dari serangan *Injection* dan *XSS* (Vieira, Antunes, & Madeira, 2009) sehingga *website* telah memenuhi syarat tersebut.

2. *Reliability*

Dari hasil pengujian dengan menggunakan WAPT, persentase keberhasilan dapat dilihat seperti pada Tabel 13.

Tabel 13. Persentase Tingkat Keberhasilan Pengujian *Reliability*

No.	Komponen	Berhasil	Gagal	Total	Persentase
1	<i>Sessions</i>	56	0	56	100%
2	<i>Pages</i>	1465	0	1465	100%
3	<i>Hits</i>	3033	0	3033	100%

Dari Tabel 13 dapat disimpulkan bahwa persentase keberhasilan seluruh komponen pada pengujian *reliability* sebesar 100%. *Software* dapat dikatakan telah memenuhi standar pada aspek *reliability* karena persentase minimal yang harus dicapai berdasarkan standar Telcordia adalah 95%.

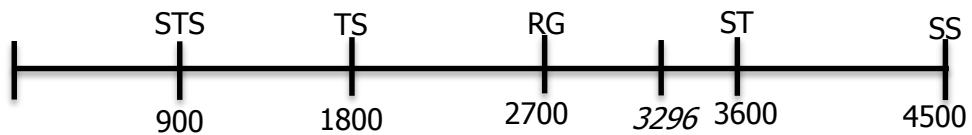
3. *Usability*

Dari hasil jawaban responden pada kuesioner USE, jumlah jawaban dikalikan dengan skor agar didapatkan skor total dari hasil jawaban responden. Perhitungan tersebut dapat dilihat pada Tabel 14.

Tabel 14. Perolehan Skor Kuesioner

	Jumlah	Skor	Jumlah x Skor
SS	91	5	455
ST	462	4	1848
RG	304	3	912
TS	38	2	76
STS	5	1	5
Skor Total:			3296

Jumlah skor ideal untuk seluruh *item* = $5 \times 30 \times 30 = 4500$. Jadi tingkat persetujuannya berdasarkan data tersebut = $(3296 : 4500) \times 100\% = 73\%$ dari yang diharapkan. Secara kontium dapat dilihat pada Gambar 39.



Gambar 39. Tingkat Persetujuan Responden pada *Software*

Berdasarkan data yang diperoleh dari 30 responden maka skor 3296 terletak pada daerah setuju.

Perhitungan *cronbach's alpha* dengan menggunakan *tool* SPSS 20 didapatkan nilai 0,934. Jika dibandingkan dengan Tabel 8 (*internal consistency cronbach's alpha*) maka hasil tersebut masuk dalam kategori *excellent*.

4. *Efficiency*

Hasil Pengujian *efficiency* didapatkan bahwa rata-rata waktu *page load* halaman *web* sebesar 1,65 detik. *Web* dikatakan baik apabila waktu *load*-nya setidaknya 10 detik menurut Nielsen (2010) dan setidaknya 7 detik sesuai rata-rata *load web* yang diungkapkan oleh Optimize (2010) sehingga *software* telah menuhi standar pada aspek *efficiency*.

5. *Maintainability*

Hasil Pengujian *maintainability* menunjukkan bahwa nilai MI sebesar 86,09. Coleman (1994) menyatakan bahwa nilai minimal MI adalah 65 agar memenuhi standar *maintainability* sehingga *software* dikatakan telah memenuhi aspek *maintainability*.

6. *Portability*

Salonen (2012) mengungkapkan bahwa agar memenuhi standar *portability* web dapat berjalan pada tujuh *browser desktop* dari tiga mayoritas *browser* yang digunakan yaitu *google chrome*, *Mozilla Firefox*, dan *Internet Explorer* dan lima *browser* perangkat *mobile*. Dari hasil pengujian menunjukkan bahwa pengujian pada tujuh *browser desktop* dan lima *browser mobile* berjalan dengan sukses tanpa *error* sehingga *software* telah memenuhi standar pada aspek *portability*.

BAB V

SIMPULAN DAN SARAN

A. Simpulan

Berdasarkan hasil penelitian dan pembahasan yang telah dilakukan, dapat disimpulkan bahwa :

1. *Software* tes dengan analisis butir soal berbasis *web* di MAN Yogyakarta 1 dalam pengembangannya menggunakan *framework* Laravel. Proses pengembangan *software* menggunakan model *waterfall* yang terdiri dari empat tahap yaitu, (1) Analisis kebutuhan; (2) Desain; (3) Implementasi; dan (4) Pengujian. *Software* ini memiliki fitur untuk menyelenggarakan tes, mengelola tes dan soalnya serta menghasilkan laporan hasil tes dan analisis butir soal.
2. *Software* lolos dalam uji kualitas dengan standar ISO 9126 yang memiliki enam aspek dalam standarnya. Pada aspek *functionality*, fungsi dapat berjalan 100% dan tidak memiliki celah terhadap serangan *SQL Injection* dan *XSS (Cross Site Scripting)*. Dalam aspek *reliability*, saat diuji dengan *stress testing* tidak terdapat *error* dan 100% *software* dapat berjalan dengan baik. Pada aspek *usability*, tingkat persetujuan pengguna sebesar 73% dan nilai *cronbach's alpha* sebesar 0,934 dengan kategori *excellent*. Dalam aspek *efficiency*, waktu *load time* rata-rata sebesar 1,65 detik (diterima). Pada aspek *maintainability*, *software* berada pada tingkat mudah dalam perbaikan. Pada aspek *portability*, *website* berhasil diakses melalui semua *browser desktop* dan *mobile* yang diujicobakan tanpa *error*.

B. Keterbatasan Produk

Software yang dikembangkan masih memiliki keterbatasan dalam hal kelengkapan fitur yang berupa tipe soal yang kurang beragam. Belum ada tipe soal seperti isian singkat, *essay*, menjodohkan, dan sebagainya.

C. Pengembangan Produk Lebih Lanjut

Pengembangan *software* selanjutnya dapat melengkapi fitur yang berupa tipe soal yang kurang beragam sehingga *software* semakin bisa memberikan manfaat yang lebih dari *software* sebelumnya.

D. Saran

Berdasarkan berbagai keterbatasan yang dimiliki penulis baik dari segi waktu maupun pikiran, maka penulis menyarankan untuk pengembangan penelitian yang akan datang sebagai berikut :

1. Perlu adanya penambahan fitur lain berupa tipe soal yang lebih beragam.
2. Teknik pengujian kualitas *software* menggunakan *tools* yang lebih beragam.

DAFTAR PUSTAKA

- Afhas, I. (2012). Rancang Bangun Simulasi Tes Online Calon Pegawai Negeri Sipil (CPNS). Kudus: Universitas Muria Kudus.
- Aptimize. (2010). *Website Performance Benchmark*. Diakses dari <http://www.aptimize.com/upload/docs/2010-website-performance-benchmarks.pdf>. Pada tanggal 1 Agustus 2014 jam 15.06 WIB.
- Arikunto, S. (1999). *Dasar-Dasar Evaluasi Pendidikan*. Jakarta: Bumi Aksara.
- Armel, J. (2014). *Web Application Development with Laravel PHP Framework Version 4*. Diakses dari <http://www.theseus.fi/handle/10024/74052>. Pada tanggal 11 Juli 2014, Jam 22.32 WIB.
- Asthana, A., & Olivieri, J. (2009). Quantifying software reliability and readiness. *Communications Quality and Reliability, 2009. CQR 2009. IEEE International Workshop Technical Committee on* (pp. 1–6). IEEE.
- Bhagwat, A. (2009). *Software Test Case Engineering: Treating Test Cases as a Product (or An Approach for Finding Defects that have Low Albedo Value)*. Diakses dari <http://www.stickyminds.com/article/software-test-case-engineering-treating-test-cases-product-or-approach-finding-defects-have>. Pada tanggal 9 agustus 2014 jam 08.10 WIB.
- Björemo, M., & Trninić, P. (2010). Evaluation of web application frameworks- Evaluation of web application frameworks with regards to rapid development. Sweden: Department of Computer Science and Engineering Göteborg.
- Chua, B., & Dyson, L. (2004). Applying the ISO 9126 model to the evaluation of an e-learning system. *In Beyond the Comfort Zone in Proceedings of the 21st ASCILITE Conference*, (pp. 184-190). Perth.
- Clarke, J. (2009). *SQL Injection Attacks And Defense*. US: Elsevier.
- Cole, J., & Foster, H. (2007). *Using Moodle: Teaching with the popular open source course management system*. " O'Reilly Media, Inc.".
- Coleman, D. et al. (1994). Using metrics to evaluate software system maintainability. 27(8), 44-49.
- Cook, S. (2003). *A Web developers guide to cross-site scripting*. Singapore: SANS Institute.

- Daryanto. (2001). *Evaluasi Pendidikan*. Jakarta: Rineka Cipta.
- Fahmy, S., et al. (2012). Evaluating the Quality of Software in e-Book Using the ISO 9126 Model. *International Journal of Control and Automation*, 5(2), 115-122.
- Gliem, J. A., & Gliem, R. R. (2003). Calculating, interpreting, and reporting Cronbach's alpha reliability coefficient for Likert-type scales. *Midwest Research-to-Practice Conference in Adult, Continuing, and Community Education*.
- Iacob, I., & Constantinescu, R. (2008). Testing: First step towards software quality. *Journal of Applied Quantitative Methods*, 3(3), 241-253.
- ISO. (2001). IEC 9126-1: Software Engineering-Product Quality-Part 1: Quality Model. *Geneva, Switzerland: International Organization for Standardization*.
- ISO. (2003). ISO/IEC 9126-2: Software Engineering-Product Quality-Part 2: External Metrics. *Geneva, Switzerland: International Organization for Standardization*.
- Lepine, J.-F. (2014). *PHPMetric*. Diakses dari <http://www.phpmetrics.org/>. Pada tanggal 2 Agustus 2014 jam 13.55 WIB.
- Lund, A. (2001). *Measuring usability with the USE questionnaire*. Diakses dari http://www.stcsig.org/usability/newsletter/0110_measuring_with_use.html. Pada tanggal 2 Agustus 2014 jam 13.57 WIB.
- Moss, B. (2013). *Deal Of The Week: Browseemall*. Dikses dari <http://www.webdesignerdepot.com/2013/08/deal-of-the-week-browseemall/>. Pada tanggal 2 desember 2014 jam 13.35 WIB.
- Najm, N. (2014). Measuring Maintainability Index of a Software Depending on Line of Code Only. *IOSR Journal of Computer Engineering*.
- Nielsen, J. (2006). *Quantitative studies: How many users to test?*. Diakses dari <http://www.nngroup.com/articles/quantitative-studies-how-many-users/>. Pada tanggal 21 Juli 2014, Jam 23:03 WIB.
- Nielsen, J. (2010). *Website Response Time*. Diakses dari <http://www.nngroup.com/articles/website-response-times/>. Pada tanggal 1 Agustus 2014 jam 14.08 WIB.
- Otwell, T. (2014). *The PHP Framework for Web Artisans*. Diakses dari <http://laravel.com/>. Pada tanggal 30 Juli 2014, Jam 12:21 WIB.

- Overton, T. (2012). Assessing Learners with Special Needs Seventh Edition. In *Assessing Learners with Special Needs Seventh Edition*. United States: Pearson.
- Padayachee, I., Kotze, P., & van Der Merwe, A. (2010). ISO 9126 external systems quality characteristics, sub-characteristics and domain specific criteria for evaluating e-Learning systems. *The Southern African Computer Lecturers' Association, University of Pretoria, South Africa*.
- Parwanto, T. (2011). Perancangan Aplikasi Simulasi TOEFL (Test of English as Foreign Language). Jakarta: Universitas Islam Negeri Syarif Hidayatullah.
- Pradhan, D. (2013). *WAPT; A Load Testing Tool That Delivers! [Review]*. Diakses dari <http://www.softwaretestingtricks.com/2013/02/best-load-stress-performance-testing-tool-WAPT.html>. Pada tanggal 30 Juli 2014, Jam 12:21 WIB.
- Rees, D. (2012). *Code Happy Application Development with The Laravel PHP Framework for Beginners*. Leanpub.
- Roger, S. P. (2010). *Software Engineering : A Practitioner's Approach Seventh Edition*. New York: McGraw-Hill Companies, Inc.
- Rosa, A., & Shalahuddin, M. (2013). *Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek*. Bandung: Informatika.
- Salonen, V. (2012). Automatic Portability Testing. Jyväskylä: University of Jyväskylä.
- Singh, T. (2014). *Performance testing of any website "GTmetrix Tool"*. Diakses dari <http://www.oodlestechnologies.com/blogs/Performance-testing-of-any-website-%22GTmetrix-Tool%22>. Pada tanggal 1 Agustus 2014 jam 11.03 WIB.
- Sudijono, A. (1996). *Pengantar Evaluasi Pendidikan*. Jakarta: PT RajaGrafindo Persada.
- Sugiyono. (2009). *Pendekatan Kuantitatif, Kualitatif, dan R&D*. Bandung: CV. Alfabeta.
- Surguy, M. (2013). *Why use Laravel?*. Diakses dari <http://maxoffsky.com/code-blog/why-use-laravel/>. Pada tanggal 12 Juli 2014, Jam 10:21 WIB.
- Surguy, M. (2014). *Laravel - my first framework*. Leanpub.

Tutdianah, H. (2012). Aplikasi Tes Potensi Akademik Online sebagai Alat Bantu Proses Pembelajaran Bagi Calon Peserta Ujian. Jawa Timur: Universitas Pembangunan Nasional "Veteran".

Verifysoft. (2010). *Measurement of Halstead Metrics with Testwell CMT++ and CMTJava (Complexity Measures Tool)*. Diakses dari http://www.verifysoft.com/en_halstead_metrics.html. Pada tanggal 2 Agustus 2014 jam 22.26 WIB.

Vieira, M., Antunes, N., & Madeira, H. (2009). Using web security scanners to detect vulnerabilities in web services. (pp. 566-571). IEEE.

W3schools. (2014). *Browser Statistics*. Diakses dari http://www.w3schools.com/browsers/browsers_stats.asp. Pada tanggal 3 agustus 2014 Jam 13.55 WIB.

Watson, A., McCabe, T., & Wallace, D. (1996). Structured testing: A testing methodology using the cyclomatic complexity metric. *NIST special Publication, 500(235)*, 1-114.

LAMPIRAN

Lampiran 1. Surat – surat Perijinan

1. Surat Permohonan Izin Observasi/Survey

2083

**KEMENTERIAN PENDIDIKAN DAN KEBUDAYAAN
UNIVERSITAS NEGERI YOGYAKARTA
FAKULTAS TEKNIK**

Alamat : Kampus Karangmalang, Yogyakarta, 55281
Telp. (0274) 586168 psw. 276,289,292 (0274) 586734 Fax. (0274) 586734
website : <http://ft.uny.ac.id> e-mail: ft@uny.ac.id ; teknik@uny.ac.id

SUCOFINDO
SICS
ISO 9001:2000
Certificate No. Q5C 00592

Nomor : 4178/UN34.15/PL/ 2013

Hal : Permohonan Ijin Observasi/Survey

Lamp. :

Yth. Pimpinan /Direktur /Kepala /Ketua *) : MAN Yogyakarta 1
JL. C. SIMANJUNTAK NO.60
YOGYAKARTA

Dalam rangka pelaksanaan Mata Kuliah Tugas Akhir Skripsi, kami mohon dengan hormat bantuan Saudara memberikan ijin untuk melaksanakan observasi/Survei dengan fokus permasalahan " Analisis Pengembangan Software Tes Obyektif Dengan Analisis Butir Soal Berbasis Web Di MAN Yogyakarta 1 ", bagi mahasiswa Fakultas Teknik Universitas Negeri Yogyakarta sebagai berikut:

No.	Nama	NIM	Jurusan/Program Studi
1	Muhammad Thoriq Romadhon	10520244005	Pend. Teknik Informatika - S1

Dosen Pembimbing/Dosen Pengampu:
Nama : Handaru Jati, Ph.D.
NIP : 19740511 199903 1 002
Demikian permohonan kami, atas bantuan dan kerjasama yang baik selama ini, kami mengucapkan terima kasih.

Dekan,

Wakil Dekan I,
Dr. Sunaryo Soenarto
NIP. 19580630 198601 1 0014

Tembusan:
Ketua Jurusan

*) Coret yang tidak perlu
10520244005 No. 2083

2. Surat Permohonan Izin Penelitian dari Fakultas



3. Surat Keterangan Telah Melakukan Penelitian



4. Surat Keterangan Pembimbing Tugas Akhir Skripsi

**9KEPUTUSAN DEKAN
FAKULTAS TEKNIK
UNIVERSITAS NEGERI YOGYAKARTA
NOMOR : 274/ELK/Q-I/XI/2013
TENTANG
PENGANGKATA, N PEMBIMBING TUGAS AKHIR SKRIPSI
BAGI MAHASISWA FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA**

**DEKAN FAKULTAS TEKNIK
UNIVERSITAS NEGERI YOGYAKARTA**

Menimbang : 1. Bawa sehubungan dengan telah dipenuhi syarat untuk penulisan Tugas Akhir Skripsi bagi mahasiswa Fakultas Teknik Universitas Negeri Yogyakarta, perlu diangkat pembimbing.
2. Bawa untuk keperluan dimaksud perlu ditetapkan dengan Keputusan Dekan.

Mengingat : 1. Undang-undang Nomor 20 tahun 2003.
2. Peraturan Pemerintah RI Nomor 60 tahun 1999.
3. Keputusan Presiden RI: a. Nomor 93 tahun 1999; b. 305/M tahun 1999.
4. Keputusan Menteri Pendidikan dan Kebudayaan RI: Nomor 274/O/1999.
5. Keputusan Mendiknas RI Nomor 003/O/2001.
6. Keputusan Rektor UNY Nomor : 1160/UN34/KP/2011

M E M U T U S K A N

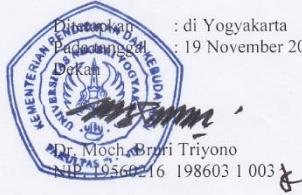
Menetapkan
Pertama : Mengangkat Pembimbing Tugas Akhir Skripsi bagi mahasiswa Fakultas Teknik Universitas Negeri Yogyakarta sebagai berikut :

Nama Pembimbing	:	Handaru Jati, Ph.D
Bagi mahasiswa	:	
Nama/No.Mahasiswa	:	Muhammad Thoriq Romadhon / 10520244005
Jurusan/ Prodi	:	Pendidikan Teknik Elektronika / Pendidikan Teknik Informatika
Judul Skripsi	:	<i>Analisis Pengembangan software Tes Objektif dengan Analisis Butir Soal Berbasis Web Di MAN Yogyakarta 1</i>

Kedua : Dosen pembimbing diserahi tugas membimbing penulisan Tugas Akhir Skripsi sesuai dengan Pedoman Tugas Akhir Skripsi.

Ketiga : Keputusan ini berlaku sejak ditetapkan

Keempat : Segala sesuatu akan diubah dan dibetulkan sebagaimana mestinya apabila di kemudian hari ternyata terdapat kekeliruan dalam Keputusan ini.

: di Yogyakarta : 19 November 2013

Dr. Moch. Arifri Triyono
NIP. 19560216 198603 1 003

Tembusan Yth :

1. Wakil Dekan II FT UNY
2. Ketua Jurusan Pendidikan Teknik Elektronika
3. Kasub. Bag Pendidikan FT UNY
4. Yang bersangkutan

5. Lembar Persetujuan Dilaksanakan Penelitian



Lampiran 2. Desain *Use Case Diagram*

1) Definisi *Use Case*

Tabel 15. Definisi *Use Case*

No.	Use Case	Deskripsi
1	Mengelola Akun	merupakan proses untuk pengelolaan akun <i>admin/guru</i> yang bertugas sebagai pengelola <i>web</i> . Mengelola akun merupakan generalisasi dari proses <i>login</i> , <i>logout</i> , mengubah <i>username</i> dan <i>password</i> , <i>reset</i> akun, dan memeriksa status <i>login</i>
2	<i>Login</i>	merupakan proses untuk masuk ke dalam halaman <i>admin</i> pada <i>web</i>
3	<i>Logout</i>	merupakan proses untuk keluar dari halaman <i>admin web</i>
4	Mengubah <i>username</i> dan <i>password</i>	merupakan proses untuk mengganti <i>username</i> dan <i>password</i> yang digunakan untuk <i>login</i>
5	<i>Reset</i> akun	merupakan proses yang digunakan untuk mengembalikan <i>username</i> dan <i>password</i> ke pengaturan awal.
6	Memeriksa status <i>login</i>	merupakan proses untuk memeriksa apakah pengguna <i>web</i> sudah melakukan <i>login</i> atau belum
7	Mengelola tes	merupakan proses generalisasi dari tambah tes, edit tes, hapus tes, duplikat tes, dan melihat tes
8	Tambah tes	merupakan proses membuat tes pada <i>web</i>
9	Edit tes	merupakan proses mengubah tes yang sudah ada pada <i>web</i>
10	Hapus tes	merupakan proses untuk menghapus tes yang sudah ada pada <i>web</i>
11	Duplikat tes	merupakan proses untuk menggandakan tes dari tes yang sudah ada pada <i>web</i>
12	Melihat tes	merupakan proses untuk menampilkan tes yang sudah ada pada <i>web</i>
13	Mengelola soal	merupakan proses generalisasi dari tambah tes, edit soal, hapus soal, dan melihat soal
14	Tambah soal	merupakan proses membuat soal pada <i>web</i>
15	Edit soal	merupakan proses mengubah soal yang sudah ada pada <i>web</i>
16	Hapus soal	merupakan proses untuk menghapus soal yang sudah ada pada <i>web</i>
17	Melihat soal	merupakan proses untuk menampilkan soal yang sudah ada pada <i>web</i>
18	Laporan	merupakan proses generalisasi dari melihat hasil tes, dan melihat hasil analisis butir soal
19	Melihat hasil tes	merupakan proses melihat hasil tes

Lanjutan Tabel 15.

No.	Use Case	Deskripsi
20	Melihat hasil analisis butir soal	merupakan proses untuk melihat hasil analisis butir soal
21	Cetak laporan	merupakan proses yang digunakan untuk mencetak laporan
22	Melaksanakan tes	merupakan proses pengeraan tes
23	Mengisi identitas	merupakan proses pengisian identitas peserta tes untuk pelaksanaan tes

2) Skenario *Use Case*

Nama *Use Case* : *login*

Tabel 16. Skenario *Use Case Login*

Aksi aktor	Reaksi sistem
Skenario normal	
1. Memasukkan <i>username</i> dan <i>password</i>	
	2. Mengecek valid tidaknya data masukkan
	3. Masuk ke halaman <i>admin web</i>
Skenario alternatif	
1. Memasukkan <i>username</i> dan <i>password</i>	
	2. Mengecek valid tidaknya data masukkan
	3. Menampilkan pesan login tidak valid
4. Memasukkan <i>username</i> dan <i>password</i> yang valid	
	5. Mengecek valid tidaknya data masukkan
	6. Masuk ke halaman <i>admin web</i>

Nama *Use Case* : *logout*

Tabel 17. Skenario *Use Case Logout*

Aksi aktor	Reaksi sistem
Skenario normal	
1. Memilih menu <i>logout</i>	
	2. Keluar dari halaman <i>admin web</i> / <i>logout</i>

Nama *Use Case* : ganti *username* dan *password*

Tabel 18. Skenario *Use Case* Ganti *username* dan *password*

Aksi aktor	Reaksi sistem
Skenario normal	
1. Memasukkan <i>username baru</i> , <i>password lama</i> dan <i>password baru</i>	
	2. Mengecek valid tidaknya data masukkan
	3. <i>Username</i> dan <i>password</i> baru disimpan
Skenario alternatif	
1. Memasukkan <i>username baru</i> , <i>password lama</i> dan <i>password baru</i>	
	2. Mengecek valid tidaknya data masukkan
	3. Menampilkan pesan data tidak valid
4. Memasukkan <i>username baru</i> , <i>password lama</i> dan <i>password baru</i> yang valid	
	5. Mengecek valid tidaknya data masukkan
	6. <i>Username</i> dan <i>password</i> baru disimpan

Nama *Use Case* : memeriksa status *login*

Tabel 19. Skenario *Use Case* Memeriksa Status *Login*

Aksi aktor	Reaksi sistem
Skenario normal	
	1. Memeriksa apakah ada <i>session login</i> yang ada
	2. Mengembalikan status apakah sudah <i>login</i> atau belum

Nama *Use Case* : *reset akun*

Tabel 20. Skenario *Use Case Reset Akun*

Aksi aktor	Reaksi sistem
Skenario normal	
	1. Me-reset <i>Username</i> dan <i>Password</i>

Nama *Use Case* : tambah data

Tabel 21. Skenario *Use Case Tambah Data*

Aksi aktor	Reaksi sistem
Skenario normal	
1. Memasukkan data sesuai dengan kolom yang ada	
	2. Mengecek valid tidaknya data masukkan
	3. Menyimpan data ke basis data
	4. Menampilkan pesan data berhasil disimpan
Skenario alternatif	
1. Memasukkan data sesuai dengan kolom yang ada	
	2. Mengecek valid tidaknya data yang dimasukkan
	3. Menampilkan pesan bahwa data tidak valid
4. Memperbaiki data yang tidak valid	
	5. Mengecek valid tidaknya data yang dimasukkan
	6. Menyimpan data ke <i>database</i>
	7. Menampilkan pesan data berhasil disimpan

Nama *Use Case* : edit data

Tabel 22. Skenario *Use Case* Edit Data

Aksi aktor	Reaksi sistem
Skenario normal	
1. Memilih data yang akan diedit	
	2. Menampilkan data dalam <i>form</i> yang akan diedit
3. Mengedit data	
	4. Memeriksa valid tidaknya data yang dimasukkan
	5. Menyimpan data yang diedit ke <i>database</i>
	6. Menampilkan pesan data berhasil diedit
Skenario alternatif	
1. Memilih data yang akan diedit	
	2. Menampilkan data dalam <i>form</i> yang akan dedit
3. Mengedit data	
	4. Memeriksa valid tidaknya data yang dimasukkan
	5. Menampilkan pesan data yang dimasukkan tidak valid
6. Memperbaiki data masukan yang diedit dan tidak valid	
	7. Memeriksa valid tidaknya data yang di mamasukkan
	8. Menyimpan data yang telah deditke dalam basis data
	9. Menampilkan pesan data berhasil diedit

Nama *Use Case* : hapus data

Tabel 23. Skenario *Use Case* Hapus Data

Aksi aktor	Reaksi sistem
Skenario normal	
1. Memilih data yang akan dihapus	
	2. Menampilkan pesan konfirmasi apakah data benar-benar akan dihapus
3. Mengeklik pilihan 'ya'	
	4. Menghapus data dari <i>database</i>
	5. Menampilkan pesan data berhasil dihapus
Skenario alternatif	
1. Memilih data yang akan dihapus	
	2. Menampilkan pesan konfirmasi apakah data benar-benar akan dihapus
3. Mengeklik pilihan 'tidak'	
	4. Kembali ke tampilan sebelumnya

Nama *Use Case* : lihat data

Tabel 24. Lihat Data

Aksi aktor	Reaksi sistem
Skenario normal	
	1. Menampilkan data

Nama *Use Case* : duplikat tes

Tabel 25. Skenario *Use Case* Duplikat Tes

Aksi aktor	Reaksi sistem
Skenario normal	
1. Memilih data yang akan diduplikat	
	2. Menampilkan data dalam <i>form</i> duplikat data
3. Mengedit data	
	4. Memeriksa valid tidaknya data yang dimasukkan
	5. Menyimpan data yang diduplikat ke <i>database</i>
	6. Menampilkan pesan data berhasil diduplikat
Skenario alternatif	
1. Memilih data yang akan diduplikat	
	2. Menampilkan data dalam <i>form</i> duplikat data
3. Mengedit data	
	4. Memeriksa valid tidaknya data yang dimasukkan
	5. Menampilkan pesan data yang dimasukkan tidak valid
6. Memperbaiki data masukan	
	7. Memeriksa valid tidaknya data yang dimasukkan
	8. Menyimpan data yang diduplikat ke dalam basis data
	9. Menampilkan pesan data berhasil diduplikat

Nama *Use Case* : lihat analisis butir soal

Tabel 26. Skenario *Use Case* Lihat Analisis Butir Soal

Aksi aktor	Reaksi sistem
Skenario normal	
1. Memilih data yang akan dilihat	
	2. Menampilkan data Analisis butir soal

Nama *Use Case* : memilih tes

Tabel 27. Skenario *Use Case* Memilih Tes

Aksi aktor	Reaksi sistem
Skenario normal	
	1. Menampilkan data tes
2. Memilih salah satu tes	

Nama *Use Case* : mengisi identitas

Tabel 28. Skenario *Use Case* Mengisi Identitas

Aksi aktor	Reaksi sistem
Skenario normal	
	1. Menampilkan <i>form</i> isian identitas
2. Mengisi identitas	3. Menyimpan data ke database

Nama *Use Case* : melaksanakan tes

Tabel 29. Skenario *Use Case* Melakukan Tes

Aksi aktor	Reaksi sistem
Skenario normal	
	1. Menampilkan soal tes
2. Mengisi jawaban	3. Menghitung nilai dan menyimpan jawaban dan nilai ke database

Nama *Use Case* : melihat hasil tes

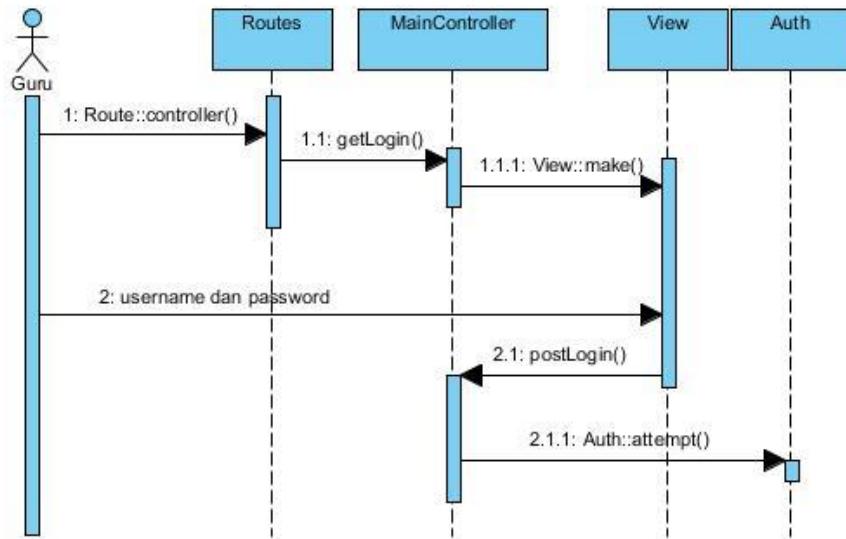
Tabel 30. Skenario *Use Case* Melihat Hasil Tes

Aksi aktor	Reaksi sistem
Skenario normal	
	1. Menampilkan hasil tes

Lampiran 3. Desain *Sequence Diagram*

1. Login

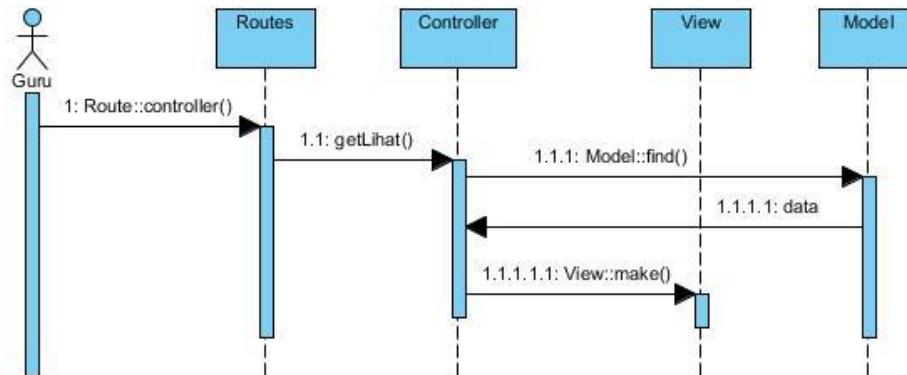
Gambar berikut ini merupakan *sequence diagram* untuk fungsi *login*.



Gambar 40. *Sequence Diagram Login*

2. Lihat Data

Gambar berikut ini merupakan *sequence diagram* untuk fungsi lihat data.

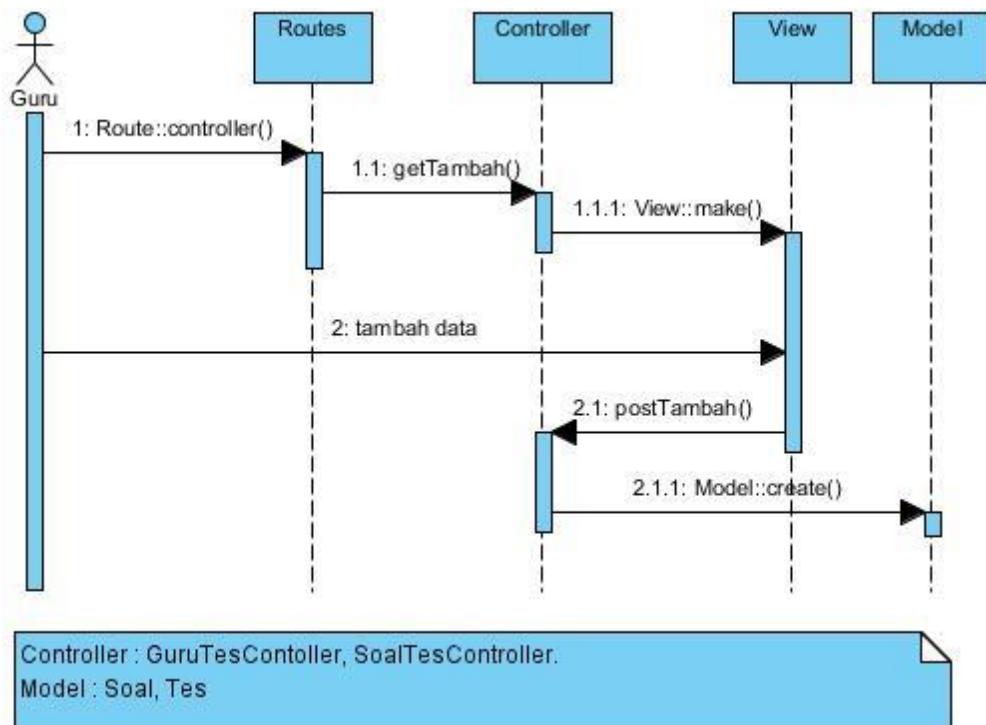


- Controller : GuruTesController, SoalTesController, GuruLaporanController
- Sequence diagram untuk Lihat Tes, Lihat Soal, Lihat Hasil Tes
- Model : Soal, Tes, Laporan
- 1.1 : Method untuk melihat data. (nama method tergantung pada masing-masing controller.
- 1.1.1 : Method dapat diganti Model::all() ataupun yang lain sesuai kebutuhan

Gambar 41. *Sequence Diagram Lihat Data*

3. Tambah Data

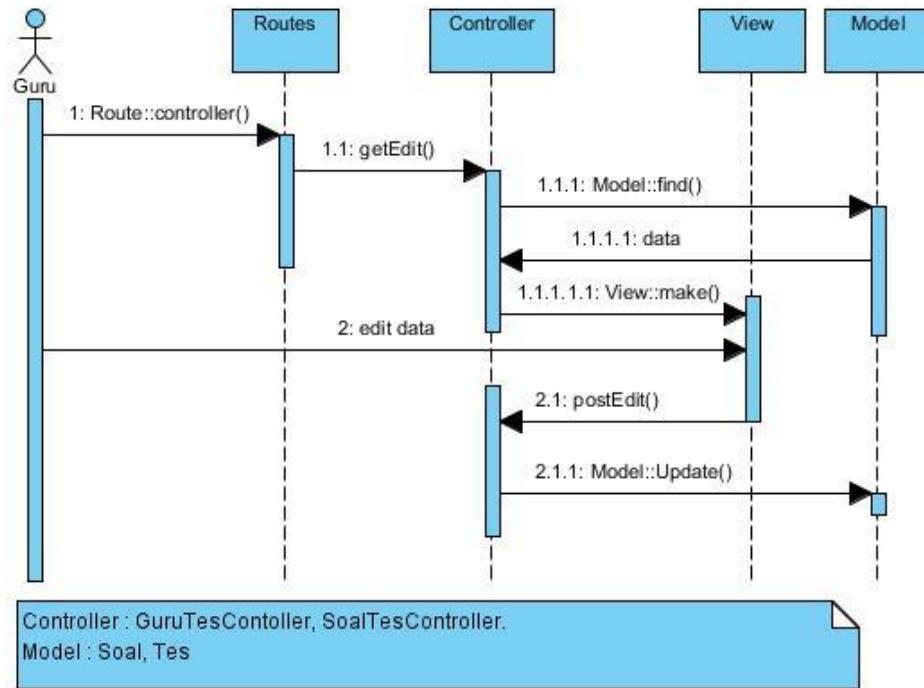
Gambar berikut ini merupakan *sequence diagram* untuk fungsi tambah data.



Gambar 42. *Sequence Diagram* Tambah Data

4. Edit Data

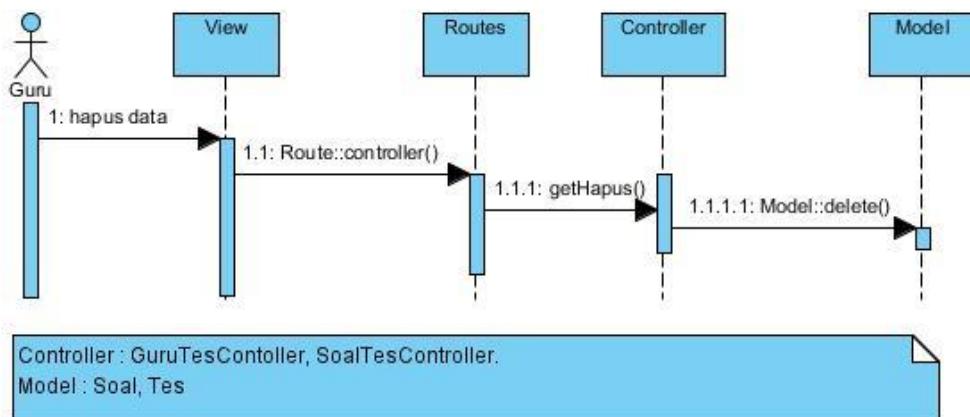
Gambar berikut ini merupakan *sequence diagram* untuk fungsi edit data.



Gambar 43. *Sequence Diagram* Edit Data

5. Hapus Data

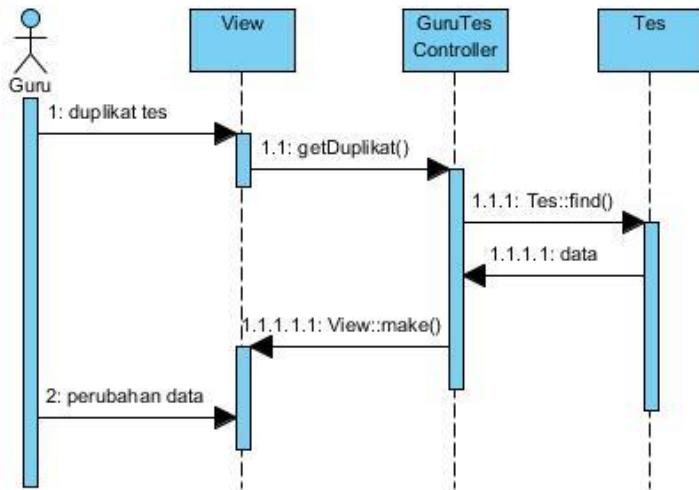
Gambar berikut ini merupakan *sequence diagram* untuk fungsi hapus data.



Gambar 44. *Sequence Diagram* Hapus Data

6. Duplikat Tes

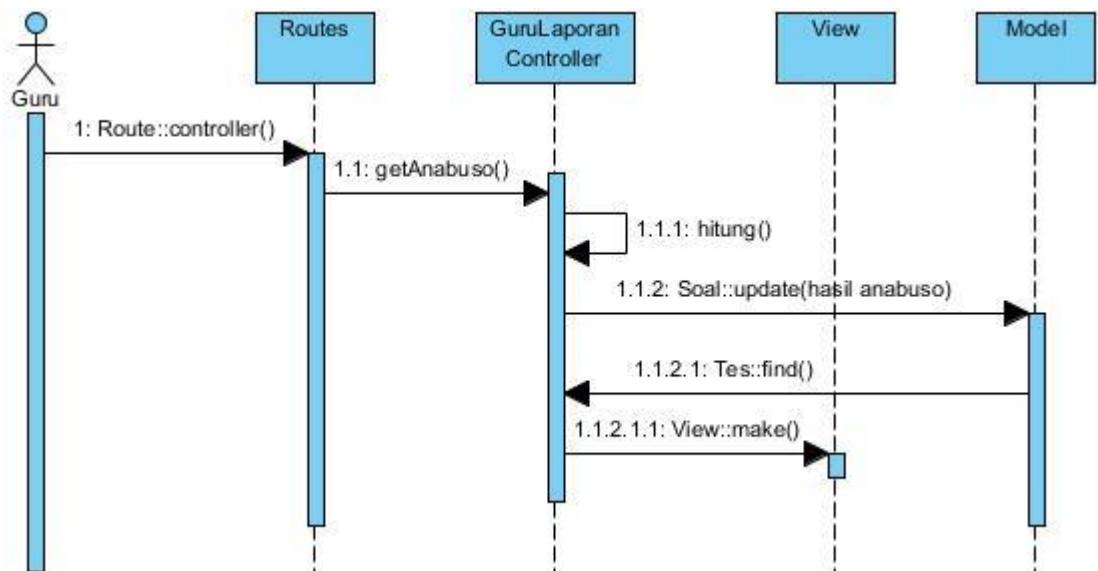
Gambar berikut ini merupakan *sequence diagram* untuk fungsi duplikat tes.



Gambar 45. *Sequence Diagram* Duplikat Tes

7. Lihat Analisis Butir Soal

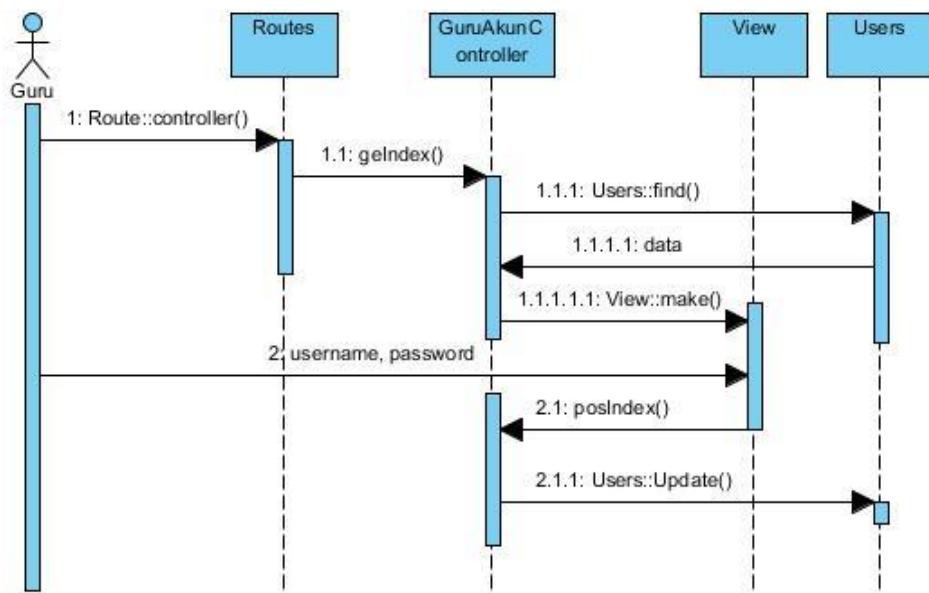
Gambar berikut ini merupakan *sequence diagram* untuk fungsi lihat analisis butir soal.



Gambar 46. *Sequence Diagram* Lihat Analisis Butir Soal

8. Mengganti *Username* dan *Password*

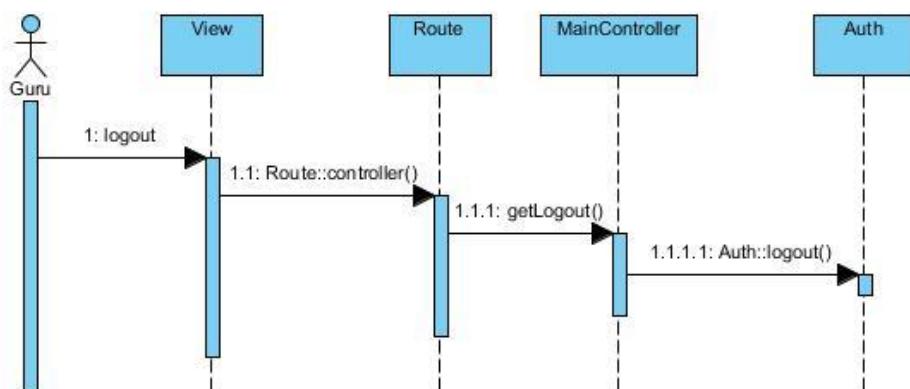
Gambar berikut ini merupakan *sequence diagram* untuk fungsi mengganti *username* dan *password*.



Gambar 47. *Sequence Diagram* Mengganti *Username* dan *Password*

9. Logout

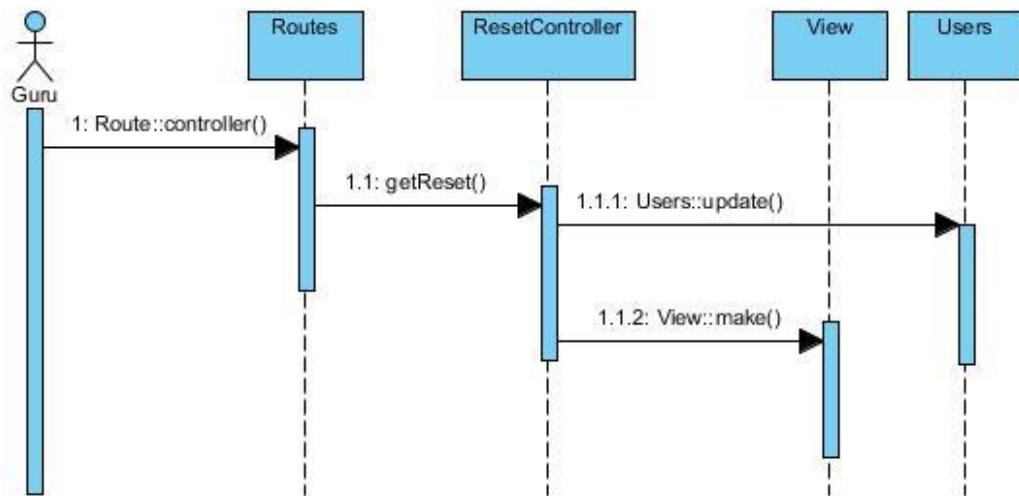
Gambar berikut ini merupakan *sequence diagram* untuk fungsi *logout*.



Gambar 48. *Sequence Diagram* Logout

10. Reset Akun

Gambar berikut ini merupakan *sequence diagram* untuk fungsi *reset akun*.

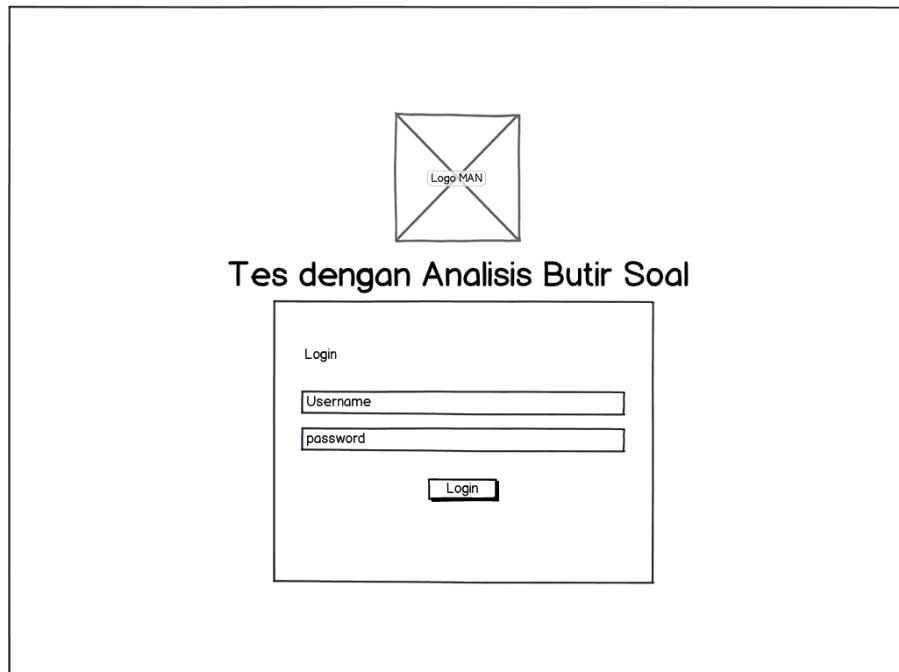


Gambar 49. *Sequence Diagram Reset Akun*

Lampiran 4. Desain Antarmuka

1. Halaman Guru

a. *Login*



Logo MAN

Tes dengan Analisis Butir Soal

Login

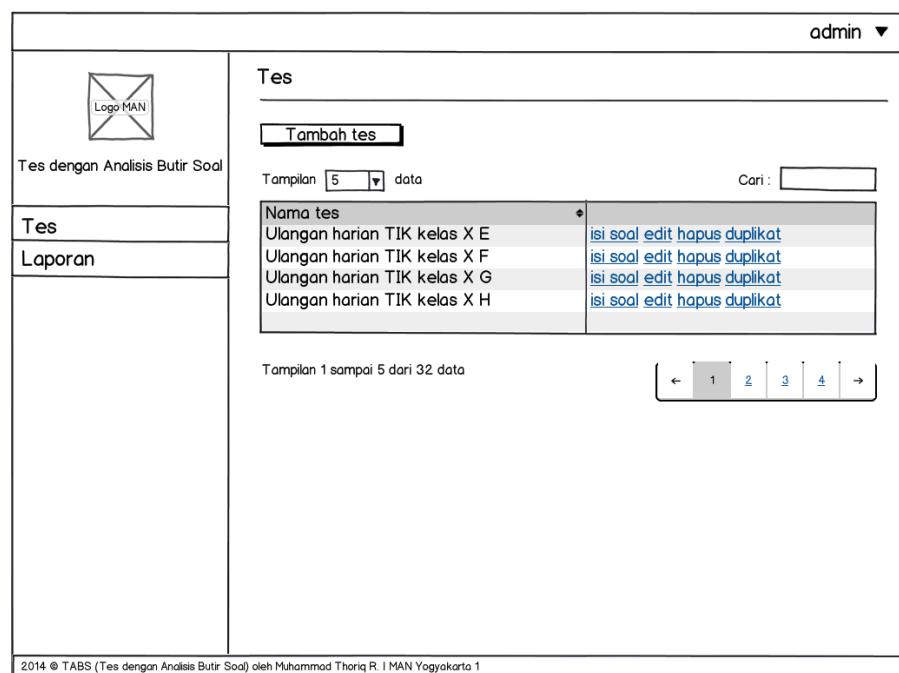
Username

password

Login

Gambar 50. Desain Halaman Guru Bagian *Login*

b. Tes



admin ▾

Logo MAN

Tes dengan Analisis Butir Soal

Tes

Tambah tes

Cari :

Tampilan 5 data

Nama tes	
Ulangan harian TIK kelas X E	isi soal edit hapus duplicat
Ulangan harian TIK kelas X F	isi soal edit hapus duplicat
Ulangan harian TIK kelas X G	isi soal edit hapus duplicat
Ulangan harian TIK kelas X H	isi soal edit hapus duplicat

Tampilan 1 sampai 5 dari 32 data

← 1 2 3 4 →

2014 © TABS (Tes dengan Analisis Butir Soal) oleh Muhammad Thoriq R. I MAN Yogyakarta 1

Gambar 51. Desain Halaman Guru Bagian Tes

C. Form Tes

admin ▼

Tambah Tes

Tes dengan Analisis Butir Soal

Tes

Laporan

Nama Tes

Petunjuk

KKM

waktu

Pengaturan

acak soal

acak jawaban

Simpan

2014 © TABS (Tes dengan Analisis Butir Soal) oleh Muhammad Thoriq R. I MAN Yogyakarta 1

Gambar 52. Desain Halaman Guru Bagian *Form Tes*

d. Soal

admin ▼

Soal

Tambah Soal

Informasi Tes

Nama Tes

KKM

Waktu

Petunjuk

Tampilan **5** data Cari :

Nama soal	
Contoh sample soal	isi soal edit hapus duplikat
Contoh sample soal	isi soal edit hapus duplikat
Contoh sample soal	isi soal edit hapus duplikat
Contoh sample soal	isi soal edit hapus duplikat

Tampilan 1 sampai 5 dari 32 data

← 1 2 3 4 →

2014 © TABS (Tes dengan Analisis Butir Soal) oleh Muhammad Thoriq R. I MAN Yogyakarta 1

Gambar 53. Desain Halaman Bagian Guru Soal

e. *Form Soal*

Gambar 54. Desain Halaman Guru Bagian *Form Soal*

f. *Laporan*

Gambar 55. Desain Halaman Guru Bagian Soal

g. Hasil Tes

Gambar 56. Desain Halaman Guru Bagian Hasil Tes

h. Detail Hasil Tes

Gambar 57. Desain Halaman Guru Bagian Detai Hasil Tes

i. Analisis Butir Soal

Logo MAN

Tes dengan Analisis Butir Soal

Informasi Tes

Nama tes

Alokasi Waktu

Nilai Kelulusan

Jumlah Soal

No. Soal	Sodi	Tingkat Kesulitan	Daya Beda	Alternatif Jawaban Tidak Efektif	Kesimpulan

No. Soal	Pilihan1	Pilihan2	Pilihan3	Pilihan4	Pilihan5	Tidak Menjawab

2014 © TABS (Tes dengan Analisis Butir Soal) oleh Muhammad Thoriq R. I MAN Yogyakarta 1

Gambar 58. Desain Halaman Guru Bagian Analisis Butir Soal

j. Pengaturan Akun

Logo MAN

Tes dengan Analisis Butir Soal

Username

Password

>Password lama

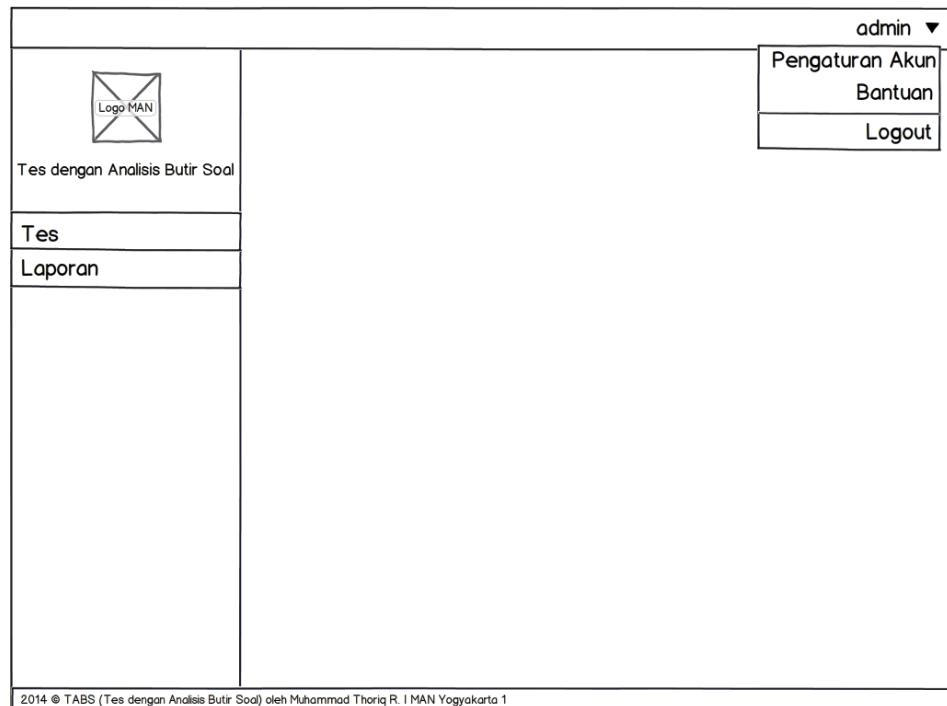
Password baru

Simpan

2014 © TABS (Tes dengan Analisis Butir Soal) oleh Muhammad Thoriq R. I MAN Yogyakarta 1

Gambar 59. Desain Halaman Guru Bagian Pengaturan Akun

k. Menu *Logout*, Bantuan dan Pengaturan Akun

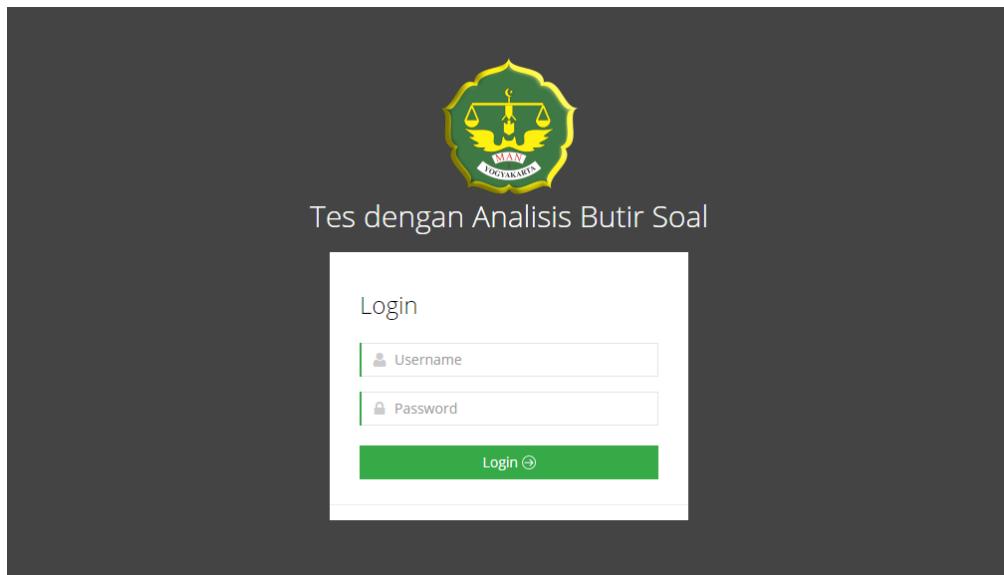


Gambar 60. Desain Halaman Guru Bagian Menu

Lampiran 5. Implementasi Antarmuka

1. Halaman Guru

a. Halaman *Login*



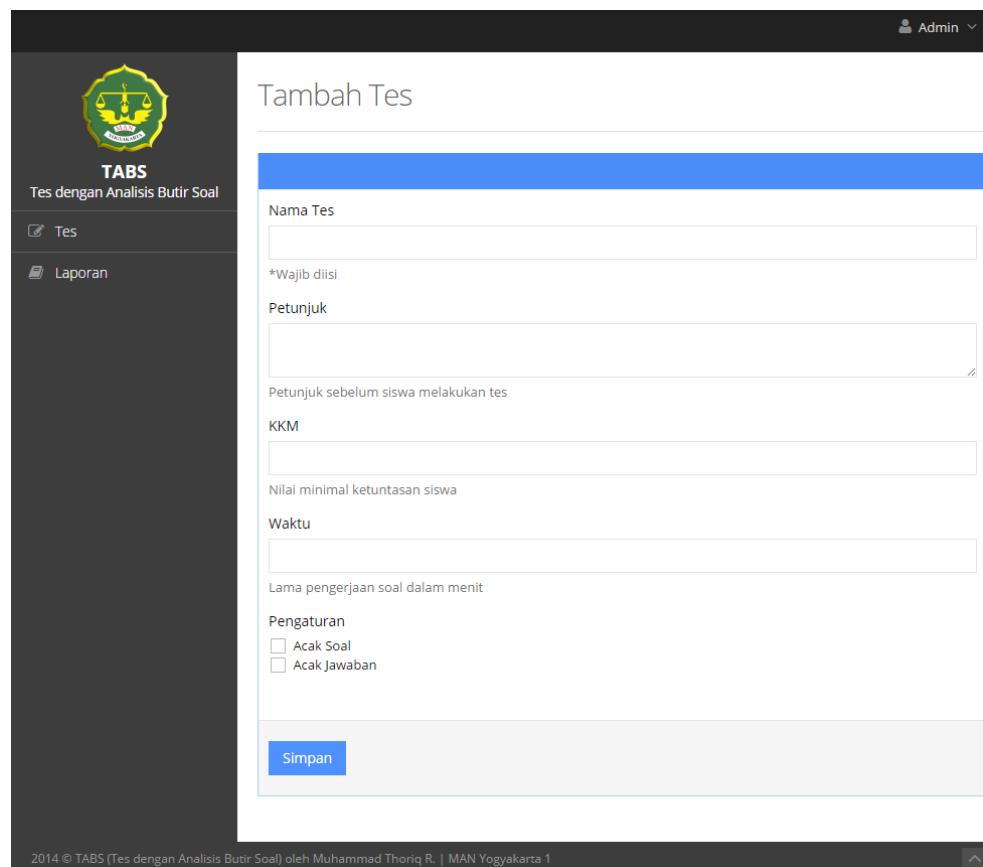
Gambar 61. Halaman Guru Bagian *Login*

b. Halaman Pengelolaan Tes

Nama tes	Waktu	KKM	
Ulangan TIK - XII Agama	45 Menit	75	Isi Soal Edit Hapus Duplikat
Ulangan TIK - XII Bahasa	45 Menit	75	Isi Soal Edit Hapus Duplikat
Ulangan TIK - XII IPA 1	45 Menit	75	Isi Soal Edit Hapus Duplikat
Ulangan TIK - XII IPA 2	45 Menit	75	Isi Soal Edit Hapus Duplikat
Ulangan TIK - XII IPA 3	45 Menit	75	Isi Soal Edit Hapus Duplikat

Gambar 62. Halaman Guru Bagian Pengelolaan Tes

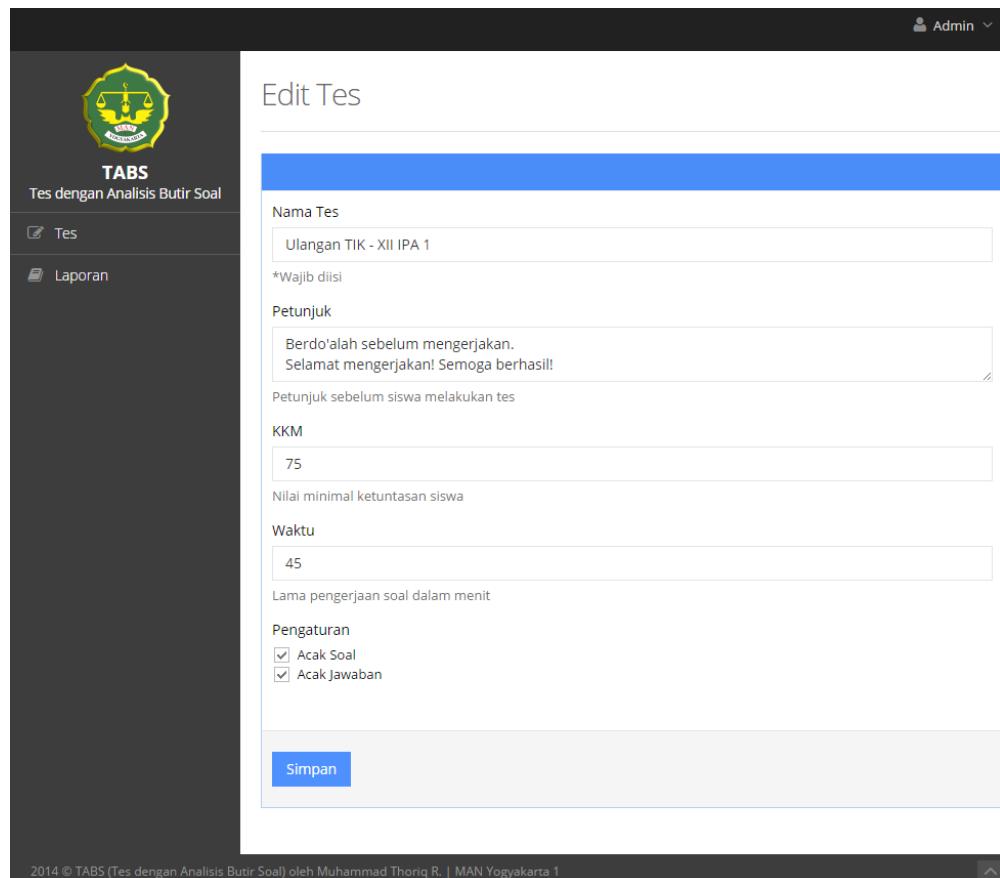
c. Halaman Tambah Tes



The screenshot shows the 'Tambah Tes' (Add Test) page of the TABS application. The left sidebar has a logo and navigation links for 'Tes' and 'Laporan'. The main area has a title 'Tambah Tes' and fields for 'Nama Tes' (Test Name), 'Petunjuk' (Instructions), 'KKM' (Minimum Passing Grade), 'Waktu' (Time), and 'Pengaturan' (Settings) with checkboxes for 'Acak Soal' (Shuffle Questions) and 'Acak Jawaban' (Shuffle Answers). A 'Simpan' (Save) button is at the bottom.

Gambar 63. Halaman Guru Bagian Tambah Tes

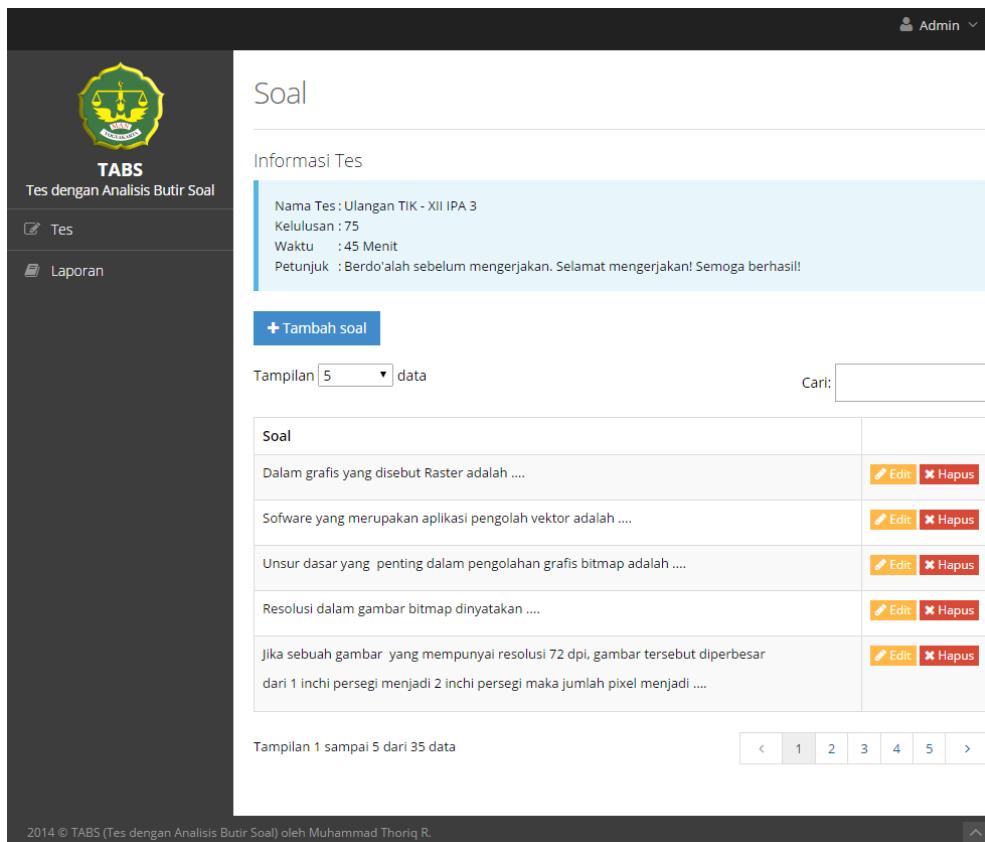
d. Halaman Edit Tes



The screenshot shows the 'Edit Tes' (Edit Test) page of the TABS application. The left sidebar has a logo and navigation links for 'Tes' and 'Laporan'. The main content area has a blue header bar. The first section is 'Nama Tes' (Test Name) with a text input containing 'Ulangan TIK - XII IPA 1'. Below it is a note: '*Wajib diisi'. The next section is 'Petunjuk' (Instructions) with a text area containing 'Berdo'alah sebelum mengerjakan. Selamat mengerjakan! Semoga berhasil!'. Below that is 'Petunjuk sebelum siswa melakukan tes' (Instructions before students take the test). The next section is 'KKM' (Minimum Grade) with a text input containing '75'. Below it is a note: 'Nilai minimal ketuntasan siswa'. The next section is 'Waktu' (Time) with a text input containing '45'. Below it is a note: 'Lama pengerjaan soal dalam menit'. The final section is 'Pengaturan' (Settings) with two checked checkboxes: 'Acak Soal' (Shuffle Questions) and 'Acak Jawaban' (Shuffle Answers). At the bottom is a blue 'Simpan' (Save) button.

Gambar 64. Halaman Guru Bagian Edit Tes

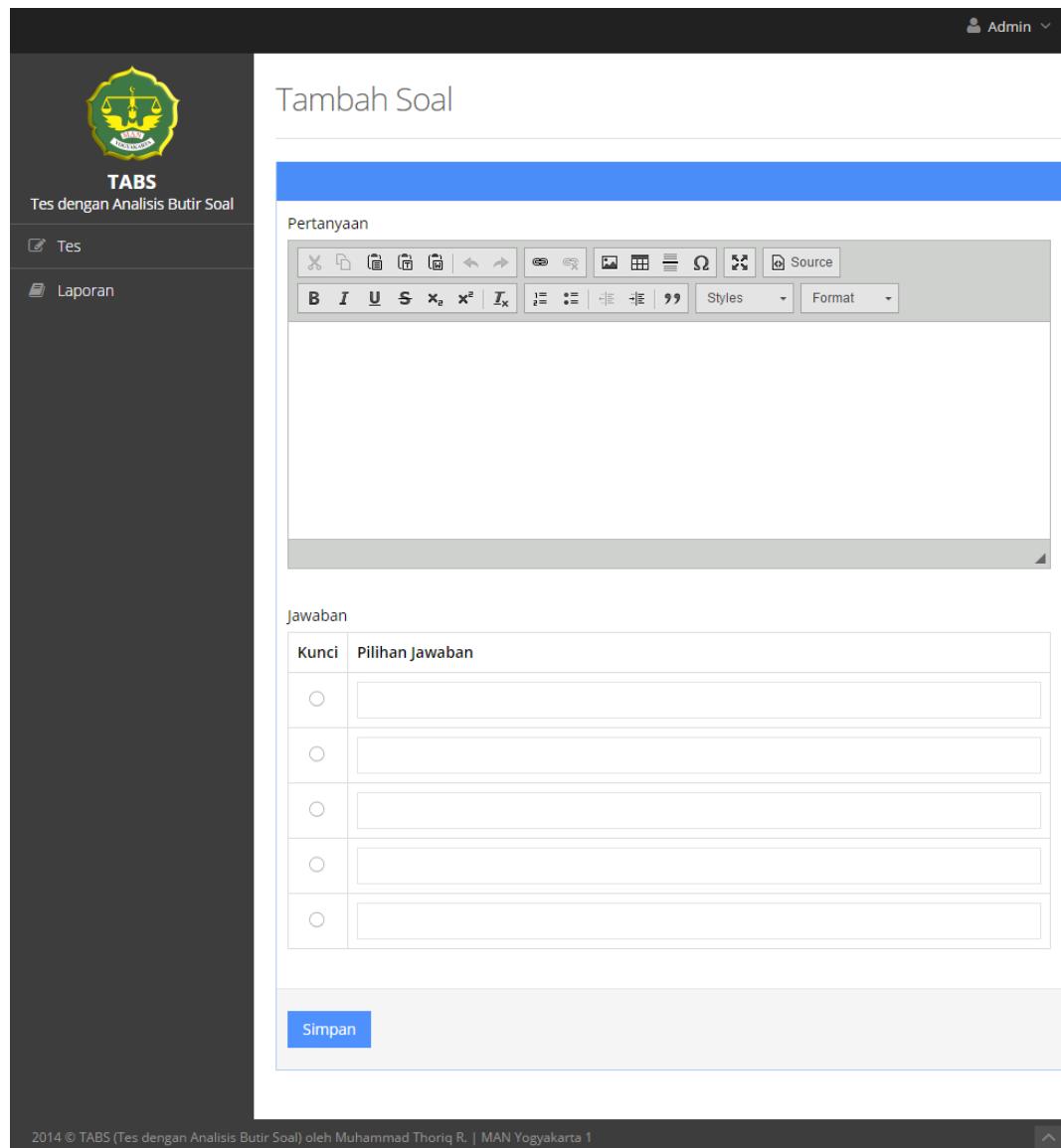
e. Halaman Pengelolaan Soal



The screenshot shows the 'Soal' (Questions) page of the TABS (Tes dengan Analisis Butir Soal) application. The interface is in Indonesian. At the top, there is a header with a user icon labeled 'Admin'. On the left, a sidebar has the 'TABS' logo and the text 'Tes dengan Analisis Butir Soal'. Below the logo are two menu items: 'Tes' (selected) and 'Laporan'. The main content area is titled 'Soal' and contains a section for 'Informasi Tes' (Test Information) with details: 'Nama Tes : Ulangan TIK - XII IPA 3', 'Kelulusan : 75', 'Waktu : 45 Menit', and a note: 'Petunjuk : Berdo'alah sebelum mengerjakan. Selamat mengerjakan! Semoga berhasil!'. Below this is a blue button labeled '+ Tambah soal' (Add question). A search bar with the placeholder 'Cari:' and a dropdown menu for 'Tampilan' (Display) set to '5' are also present. The main table lists 35 questions, each with an 'Edit' button and a 'Hapus' (Delete) button. The questions are: 1. Dalam grafis yang disebut Raster adalah 2. Sofware yang merupakan aplikasi pengolah vektor adalah 3. Unsur dasar yang penting dalam pengolahan grafis bitmap adalah 4. Resolusi dalam gambar bitmap dinyatakan 5. Jika sebuah gambar yang mempunyai resolusi 72 dpi, gambar tersebut diperbesar dari 1 inchi persegi menjadi 2 inchi persegi maka jumlah pixel menjadi

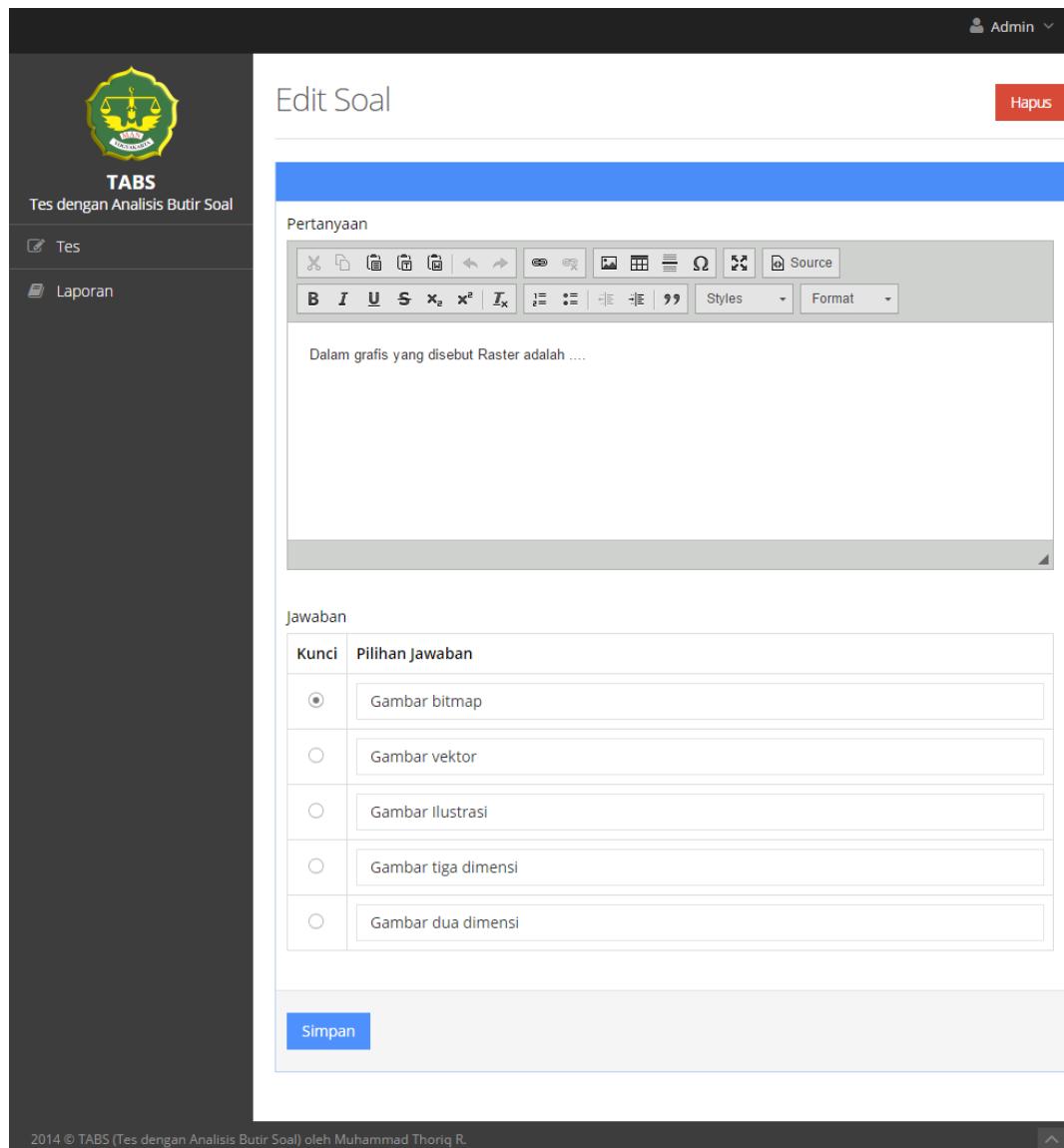
Gambar 65. Halaman Guru Bagian Pengelolaan Soal

f. Halaman Tambah Soal



Gambar 66. Halaman Guru Bagian Tambah Soal

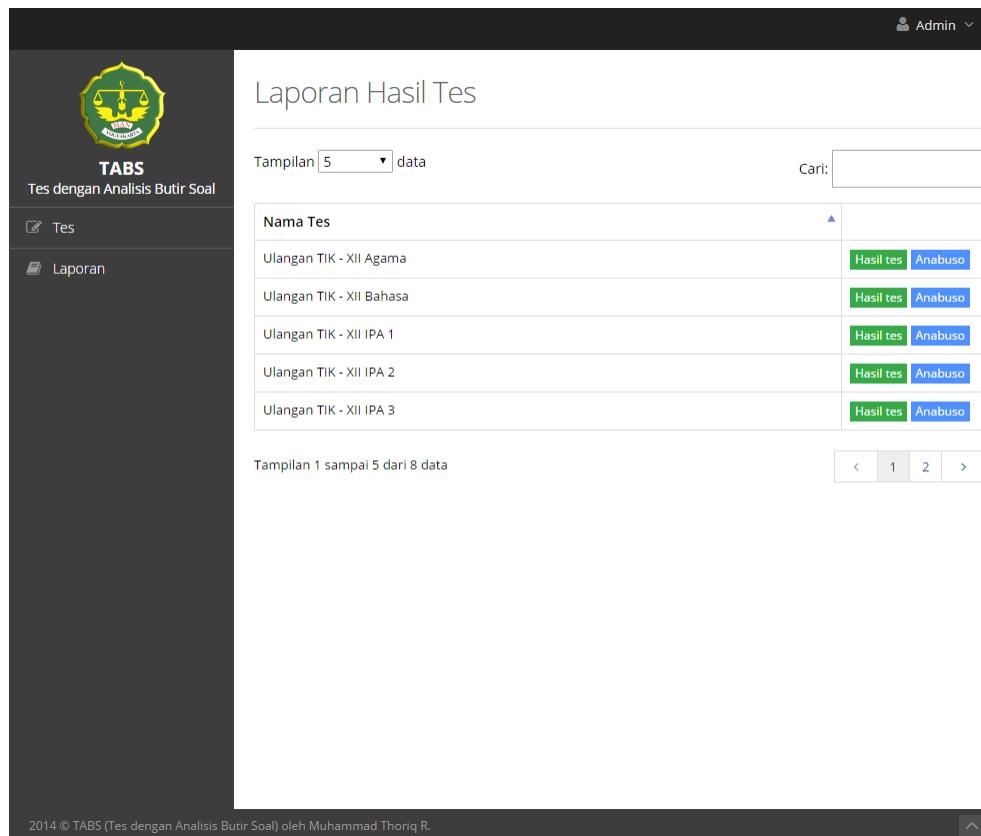
g. Halaman Edit Soal



The screenshot shows the 'Edit Soal' (Edit Question) page of the TABS (Tes dengan Analisis Butir Soal) software. The interface is in Indonesian. At the top right, there is a user icon labeled 'Admin'. Below the header, the page title 'Edit Soal' is displayed. On the left, there is a sidebar with the TABS logo and two menu items: 'Tes' (Test) and 'Laporan' (Report). The main content area is titled 'Pertanyaan' (Question). The question text is: 'Dalam grafis yang disebut Raster adalah'. Below the question is a rich text editor toolbar with various formatting options. The 'Jawaban' (Answers) section contains a table with five rows, each representing an answer option with a radio button and a text input field. The table has two columns: 'Kunci' (Correct Answer) and 'Pilihan Jawaban' (Answer Options). The correct answer is 'Gambar bitmap'. The other options are 'Gambar vektor', 'Gambar ilustrasi', 'Gambar tiga dimensi', and 'Gambar dua dimensi'. At the bottom of the page is a blue 'Simpan' (Save) button. The footer of the page includes the copyright notice '2014 © TABS (Tes dengan Analisis Butir Soal) oleh Muhammad Thoriq R.' and a small upward arrow icon.

Gambar 67. Halaman Guru Bagian Edit Soal

h. Halaman Laporan



Laporan Hasil Tes

Tampilan 5 data Cari:

Nama Tes	Hasil tes	Anabuso
Ulangan TIK - XII Agama	Hasil tes	Anabuso
Ulangan TIK - XII Bahasa	Hasil tes	Anabuso
Ulangan TIK - XII IPA 1	Hasil tes	Anabuso
Ulangan TIK - XII IPA 2	Hasil tes	Anabuso
Ulangan TIK - XII IPA 3	Hasil tes	Anabuso

Tampilan 1 sampai 5 dari 8 data

2014 © TABS (Tes dengan Analisis Butir Soal) oleh Muhammad Thoriq R.

Gambar 68. Halaman Guru Bagian Halaman Laporan

i. Halaman Hasil Tes

The screenshot shows a web-based application for managing tests. The sidebar on the left features the logo of the National Institute of Higher Education (INSTITUT PERTAMA NEGERI) and the text "TABS" (Tes dengan Analisis Butir Soal). It has two main navigation links: "Tes" and "Laporan". The main content area is titled "Hasil : Ulangan TIK - XII IPA 3". It includes a "Cetak" (Print) button. Below this, there is a section titled "Informasi Tes" with the following details: "Alokasi waktu : 45 menit", "Nilai kelulusan : 75", and "Jumlah soal : 35". There is also a search bar with a "Cari:" placeholder and a dropdown menu for "Tampilan" set to "5". The main table displays student results with columns for "Nama", "Waktu Selesai", "Nilai", and "Ketuntasan". Each row includes a "Detail" button. The table shows the following data:

Nama	Waktu Selesai	Nilai	Ketuntasan	
Abdurrahman Hanif	23-09-2014 11:19:48	71	Tidak tuntas	Detail
Adelia Zelika	23-09-2014 11:39:16	74	Tidak tuntas	Detail
ahmad iqbal fawaid	23-09-2014 11:46:04	71	Tidak tuntas	Detail
Aisyah Nur Atiko'	23-09-2014 11:47:31	94	Tuntas	Detail
Aliza Putri Nur Hanifa	23-09-2014 11:47:31	80	Tuntas	Detail

Below the table, it says "Tampilan 1 sampai 5 dari 34 data" and a navigation bar with buttons for <, 1, 2, 3, 4, 5, and >. The footer of the page includes the URL "localhost/tes/guru/laporan/siswa/42", the name "Muhammad Thoriq R. | MAN Yogyakarta 1", and a small logo.

Gambar 69. Halaman Guru Bagian Hasil Tes

j. Halaman Detail Hasil Tes

The screenshot shows the 'Hasil' (Results) page of the TABS (Tes dengan Analisis Butir Soal) application. At the top, there is a header with a user icon and the text 'Admin'. Below the header, the page title is 'Hasil : Ulangan TIK - XII IPA 3'. On the right, there is a 'Cetak' (Print) button. The main content is divided into three sections: 'Informasi Tes' (Test Information), 'Identitas' (Identity), and 'Detail' (Details). The 'Informasi Tes' section contains the following data:

Nilai	:	71
KKM	:	75
Status Ketuntasan	:	Tidak Tuntas
Alokasi Waktu	:	00:45:00
Lama Pengerjaan	:	00:32:43

The 'Identitas' section contains the following data:

Nama	:	Abdurrahman Hanif
Kelas	:	XII IPA 3
Sekolah	:	MAN YOGYAKARTA 1

The 'Detail' section displays a table of 9 questions with their respective answers, keys, and results:

No.	Pertanyaan	Jawaban	Kunci Jawaban	Hasil
1	Dalam grafis yang disebut Raster adalah	Gambar bitmap	Gambar bitmap	Benar
2	Sofware yang merupakan aplikasi pengolah vektor adalah	Macromedia Freehand	Macromedia Freehand	Benar
3	Unsur dasar yang penting dalam pengolahan grafis bitmap adalah	pixel, resolusi dan intensitas warna.	pixel, resolusi dan Intensitas warna.	Benar
4	Resolusi dalam gambar bitmap dinyatakan	Dpi	Dpi	Benar
5	Jika sebuah gambar yang mempunyai resolusi 72 dpi, gambar tersebut diperbesar dari 1 inchi persegi menjadi 2 inchi persegi maka jumlah pixel menjadi	36	5184	Salah
6	Warna dasar model RGB untuk kepentingan tampilan display di monitor adalah	merah, kuning dan biru	merah, hijau, dan biru	Salah
7	Pemodelan warna yang didasarkan pada teknik pencetakan pada kertas yang dikenal dengan Separation Printing, adalah	CMYK	CMYK	Benar
8	Corel Draw secara umum digunakan untuk mengolah grafik dalam bentuk	Grafik Vektor	Grafik Vektor	Benar
9	Tingkat kerapatan pixel pada suatu obyek gambar bitmap di sebut	Dot	Resolusi	Salah

Gambar 70. Halaman Guru Bagian Detail Hasil Tes

k. Halaman Analisis Butir Soal

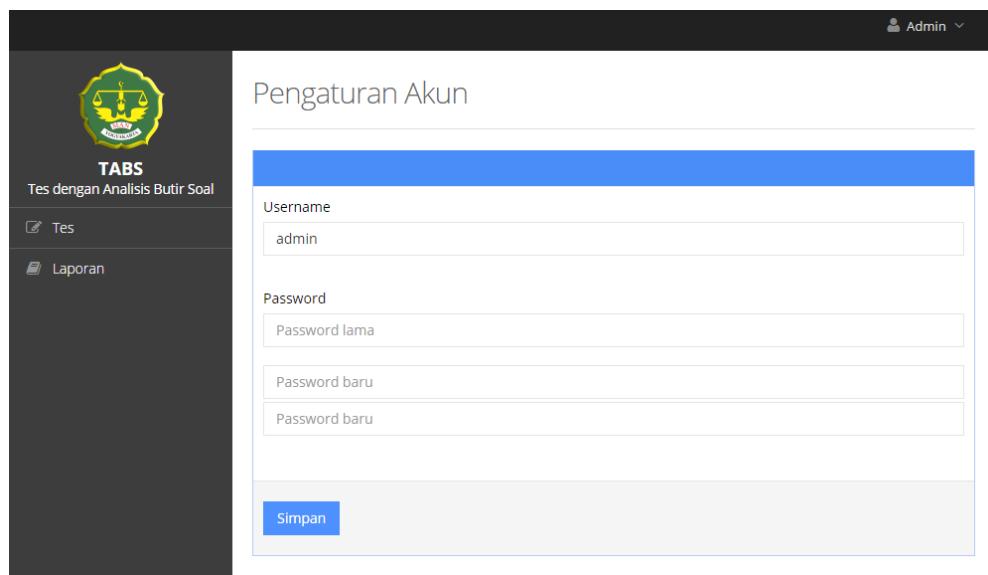
No.	Soal	Tingkat Kesulitan		Daya Beda		Alternatif Jawaban Tidak Efektif	Kesimpulan
		Koefisien	Keterangan	Koefisien	Keterangan		
1	Dalam grafis yang disebut Raster adalah	0.5	Sedang	-1	Negatif	3,4,5	Tidak Baik
2	Software yang merupakan aplikasi pengolah vektor adalah	0.5	Sedang	-1	Negatif	2,4,5	Tidak Baik
3	Unsur dasar yang penting dalam pengolahan grafis bitmap adalah	1	Mudah	0	Jelek	2,3,4,5	Tidak Baik

Gambar 71. Halaman Guru Bagian Analisis Butir Soal 1

No. Soal	Pilihan 1	Pilihan 2	Pilihan 3	Pilihan 4	Pilihan 5	Tidak menjawab
1	50 %	50 %	0 %	0 %	0 %	0 %
2	50 %	0 %	50 %	0 %	0 %	0 %
3	100 %	0 %	0 %	0 %	0 %	0 %
4	50 %	50 %	0 %	0 %	0 %	0 %
5	0 %	0 %	0 %	50 %	50 %	0 %
6	100 %	0 %	0 %	0 %	0 %	0 %
7	100 %	0 %	0 %	0 %	0 %	0 %
8	0 %	100 %	0 %	0 %	0 %	0 %
9	0 %	0 %	100 %	0 %	0 %	0 %
10	0 %	0 %	50 %	0 %	50 %	0 %
11	0 %	0 %	100 %	0 %	0 %	0 %
12	0 %	0 %	0 %	0 %	100 %	0 %
13	0 %	0 %	0 %	50 %	50 %	0 %

Gambar 72. Halaman Guru Bagian Analisis Butir Soal 2

I. Halaman Pengaturan akun



The screenshot shows the 'Pengaturan Akun' (Account Settings) page of the TABS (Tes dengan Analisis Butir Soal) application. The page has a dark header with a user icon and 'Admin'. On the left, there's a sidebar with the TABS logo, the text 'Tes dengan Analisis Butir Soal', and two menu items: 'Tes' and 'Laporan'. The main content area is titled 'Pengaturan Akun'. It contains four input fields: 'Username' (with 'admin' typed in), 'Password' (with 'Password lama' typed in), 'Password baru' (with 'Password baru' typed in), and 'Password baru' (with 'Password baru' typed in). Below these fields is a blue 'Simpan' (Save) button.

Gambar 73. Halaman Guru Bagian Pengaturan Akun

Lampiran 6. Analisis Kebutuhan

ANALISIS KEBUTUHAN
SOFTWARE/TES DENGAN ANALISIS BUTIR SOAL BERBASIS WEB
DI MAN YOGYAKARTA 1

A. Identifikasi Permasalahan

1. Bagaimana ujian/tes di MAN Yogyakarta 1?

tes masih menggunakan kertas, kadang menuntutkan komputer untuk tes dan tryout diluar jam sekolah.

2. Bagaimana pelaporan hasil ulangan siswa ke kurikulum?

Biasanya dilaporkan dalam bentuk printout yang dijadikan satu dengan laporan pembelajaran seperti RPL ketahiran siswa dan sebagainya.

3. Apa Kendala dari ujian/tes di MAN Yogyakarta 1?

- Masih menggunakan kertas sehingga membebani biaya dan boros waktu dalam kerusakan

4. Apa Kendala pelaporan hasil ulangan siswa ke kurikulum?

- Hasil ulangan perlu di-receip satu per satu sehingga lama
- Analisis butir soal juga perlu input dari seorang siswa sehingga lama

5. Apakah *software* yang sudah ada dan apa kekurangannya?

Menggunakan excel untuk melakukan analisis butir soal. Soal bagus namun Memakan waktu lama untuk input jawaban siswa.

B. Penggunaan dan Pemanfaatan

1. Siapakah yang akan menggunakan *software* ini ?

Turu dan siswa guru bertindak sebagai penyelenggara terhadap siswa sebagai peserta tes

2. Selain untuk tes/ujian dan pelaporan ke kurikulum, *software* ini dimanfaatkan untuk apa?

Software dimanfaatkan untuk layout di website sekolah bagi siswa dalam maupun luar sekolah dan digunakan untuk evaluasi. Setiap ada pelatihan guru

3. Di manakah *software* akan digunakan?

Di jaringan lokal sekolah dan website sekolah (untuk tryout)

.....
.....
.....
.....
.....
.....

3. Identifikasi Kebutuhan *Software*

Input

1. Apa saja tipe soal yang digunakan untuk tes/ulangan?

Pisanya menggunakan pilihan ganda, essay dan kalang menjodohkan. Mampu untuk pengembangan *Software* tersebutnya. pilihan ganda saja agar mudah & dicoreksi. Kippernter. Pilihan ganda dibuat bisa 5 pilikan A-E

.....
.....
.....

2. Siapakah yang melakukan input biodata siswa?

sebaiknya siswa sendiri saja karena kalau admin/guru yang menginput akan repot dan memakan banyak waktu

.....
.....
.....
.....
.....

3. Apa saja biodata siswa yang diperlukan?

Nama, kelas, sekolah

.....
.....
.....
.....
.....

Proses

1. Bagaimana proses tes/ujian siswa? Apakah ada batasan waktu dan batasan lain?

- Ada batasan waktu mengerjakan tes
- Ada RKM
- Soal tes soal dan jawaban dicetak agar mengurangi keruangan
- Pada software ter-clone itu bisa melanjutkan tesnya
- Karena ketika komputer error sehingga komputer kadang..
perlu di restart

2. Bagaimana perhitungan analisis butir soal?

Dibuat seperti perhitungan yang ada pada analisis di excel.
laporannya juga seperti pada excel.

Output

1. Bagaimanakah gambaran bentuk hasil tes siswa yang diperlukan?

Laporan ada 2 jenis:
- Laporan per siswa → Tampil biodata siswa, nilai jawaban siswa dan informasi lain yang terl...

2. Bagaimanakah hasil laporan analisis butir soal?

Dibuat seperti laporan yang sudah ada di sekolah.
(contoh hasil laporan diberikan oleh guru).

3. Apakah laporan perlu dicetak?

lalu, sebaiknya dicetak ke format Microsoft Word. Agar
sebelum diprint bisa dicetak.....
.....
.....
.....
.....

Yogyakarta, 21 Januari 2014

Narasumber


NURUL INDARTI, S.kom

Lampiran 7. Angket Pengujian

Usability

INSTRUMEN PENGUJIAN USABILITY

Pengembangan dan Analisis Kualitas Software Tes dengan Analisis Butir Soal Berbasis Web di MAN Yogyakarta 1

Nama :
Jabatan : SISWA / GURU *)

Berilah tanda *checklist* (✓) pada kolom pilihan yang sesuai dengan pendapat Anda selaku responden terhadap penggunaan *Software* Tes dengan Analisis Butir Soal Berbasis *Web* di MAN Yogyakarta 1

Keterangan pilihan:

SS : Sangat Setuju RG : Ragu-ragu STS : Sangat Tidak Setuju
S : Setuju TS : Tidak Setuju

No.	Pernyataan	Jawaban				
		STS	TS	RG	S	SS
1.	<i>Software</i> ini membantu saya bekerja lebih efektif					✓
2.	<i>Software</i> ini membantu saya bekerja lebih produktif					✓
3.	<i>Software</i> ini sangat berguna					✓
4.	<i>Software</i> ini memberikan saya pengendalian lebih atas aktivitas saya					✓
5.	<i>Software</i> ini mempermudah saya dalam menyelesaikan apa yang ingin saya selesaikan					✓
6.	<i>Software</i> ini menghemat waktu saya ketika saya menggunakan					✓
7.	<i>Software</i> ini sesuai dengan kebutuhan saya					✓
8.	<i>Software</i> ini melakukan segala sesuatu yang saya harapkan untuk dilakukan					✓
9.	<i>Software</i> ini mudah digunakan					✓
10.	<i>Software</i> ini praktis digunakan					✓
11.	<i>Software</i> ini mudah dipahami					✓
12.	<i>Software</i> ini membutuhkan langkah-langkah yang sedikit untuk mencapai apa yang ingin saya lakukan dengan <i>software</i> ini					✓
13.	<i>Software</i> ini fleksibel					✓
14.	Tidak ada kesulitan dalam menggunakan <i>Software</i> ini					✓

15.	Saya dapat menggunakan <i>Software</i> ini tanpa panduan tertulis					✓
16.	Saya tidak melihat adanya inkonsistensi saat saya gunakan <i>software</i> ini			✓		
17.	Baik pengguna yang sese kali menggunakan dan pengguna yang biasa menggunakan akan menyukai <i>software</i> ini			✓		
18.	Saya dapat menangani kesalahan dengan cepat dan mudah		✓			
19.	Saya dapat menggunakan <i>software</i> ini secara benar setiap saat			✓		
20.	Saya belajar untuk menggunakan <i>software</i> ini secara cepat				✓	
21.	Saya mudah mengingat bagaimana menggunakan ini				✓	
22.	<i>Software</i> ini mudah untuk dipelajari bagaimana penggunaannya				✓	
23.	Saya menjadi terampil menggunakan <i>Software</i> ini secara cepat				✓	
24.	Saya puas dengan <i>software</i> ini				✓	
25.	Saya akan merekomendasikan <i>software</i> ini ke teman				✓	
26.	<i>Software</i> ini menyenangkan untuk digunakan				✓	
27.	<i>Software</i> ini bekerja seperti yang saya inginkan				✓	
28.	<i>Software</i> ini memiliki tampilan yang sangat bagus			✓		
29.	Menurut saya, saya perlu memiliki <i>software</i> ini				✓	
30.	<i>Software</i> ini nyaman untuk digunakan				✓	

*) coret yang tidak perlu

Terima kasih saya ucapkan atas bantuan dan partisipasi Anda dalam penelitian ini.

Yogyakarta, September 2014

Responden,

(S. Agus Suntoro, M.Pd)

Lampiran 8. Data Pengujian *Usability*

Tabel 31. Daftar Responden Pengujian *Usability*

No. Responden	Nama	Jabatan
1	Nung Indarti, S. Kom	Guru
2	S. Agus Santosa, M.Pd.	Guru
3	Afidatul MZ	Siswa
4	Danar	Siswa
5	Chintia Mauriend DC	Siswa
6	Hermawati Septiana Putri	Siswa
7	Rari Prajna	Siswa
8	Dina Widya Fellasufah	Siswa
9	Norma S.F.	Siswa
10	Kunti Siti Sundari	Siswa
11	Muh Faris Wafiq	Siswa
12	Chelsy	Siswa
13	Aisyah Nur Atiko	Siswa
14	Winda Kurnia Sari	Siswa
15	Taris Aditama	Siswa
16	Adelia Zelika	Siswa
17	Andi Mustafa Falah	Siswa
18	Dea Rizka Annisa	Siswa
19	Masykuri Imam M	Siswa
20	Rohisotul M. Z.	Siswa
21	Deni Yoga Uwara	Siswa
22	Bagus Ismail A.	Siswa
23	Rr. Jannah Fathiyah S.	Siswa
24	Dyeong	Siswa
25	Qisti Lativa Wardani	Siswa
26	Dewo Suryo P	Siswa
27	Naufal	Siswa
28	Ami	Siswa
29	Rukhi Sholikhah	Siswa
30	Melati Astria J.	Siswa

Tabel 32. Data Pengujian *Usability*

No. Responden	Pertanyaan																													Total	
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	
1	5	5	5	4	4	5	4	5	5	5	5	4	5	4	4	1	4	5	4	4	4	4	5	4	5	5	4	5	5	133	
2	5	5	5	5	5	5	5	4	4	4	4	4	4	5	5	4	4	3	4	5	5	5	5	5	5	5	4	4	4	137	
3	4	4	4	4	4	4	3	3	4	4	3	3	4	2	2	3	3	3	3	4	2	3	2	3	4	4	4	4	4	102	
4	2	4	5	5	4	4	3	4	4	4	4	2	4	3	1	3	5	4	3	4	4	4	3	5	5	5	4	2	5	1	110
5	4	4	4	4	4	4	3	3	4	4	3	3	4	2	2	3	3	3	3	4	2	3	2	3	4	4	4	4	4	102	
6	4	3	3	3	3	3	3	3	4	4	4	3	4	3	3	3	4	3	3	3	3	3	3	3	4	4	4	4	3	4	101
7	3	3	4	4	4	3	4	3	4	4	4	3	4	4	5	3	3	4	4	4	4	4	4	3	3	3	4	3	3	4	109
8	4	4	4	4	4	3	4	4	4	4	4	4	4	4	2	2	3	3	3	3	3	3	3	3	4	4	3	4	4	106	
9	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	4	5	5	5	5	5	5	5	5	5	5	5	4	4	5	147
10	4	4	4	3	3	3	3	3	3	4	3	4	3	3	3	3	3	3	3	3	3	4	3	4	3	4	3	4	3	100	
11	4	4	4	4	4	4	4	4	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	4	3	4	3	3	4	3	101
12	4	4	4	4	4	4	3	3	4	4	3	3	4	2	2	3	3	3	3	4	2	3	2	3	4	4	4	4	4	102	
13	4	5	5	5	4	3	4	4	3	3	3	3	4	2	4	4	3	3	2	4	4	2	4	3	2	2	3	3	4	3	102
14	4	4	4	4	4	4	3	3	4	4	4	4	4	3	4	3	3	3	3	4	4	4	3	3	4	4	3	4	4	5	111
15	4	4	4	4	4	4	4	4	4	4	4	4	4	4	3	3	3	3	3	4	4	4	4	4	4	4	4	4	4	4	115
16	5	3	3	3	3	3	3	3	4	4	3	4	4	3	4	3	3	3	3	3	4	4	3	3	3	3	3	3	4	100	
17	5	5	5	5	4	5	4	5	4	5	4	4	4	4	4	4	4	4	5	4	4	5	4	5	4	5	3	3	5	130	
18	3	3	4	5	3	4	4	4	4	4	4	4	3	4	3	1	3	3	3	3	2	3	5	5	4	2	3	4	3	3	102
19	4	4	4	4	4	4	4	4	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	4	3	4	4	4	102	
20	4	4	4	4	4	4	4	4	3	4	4	4	3	2	3	4	4	3	4	4	4	3	4	4	4	4	4	4	4	112	

21	4	4	4	4	4	4	4	3	3	3	3	4	4	4	3	3	3	4	4	4	4	4	3	4	4	4	4	4	111	
22	3	4	4	4	4	4	2	3	4	4	4	4	4	4	3	3	4	4	3	4	4	4	4	3	3	4	3	4	5	111
23	4	4	4	3	4	3	3	2	4	4	4	3	3	3	3	2	4	4	4	4	4	4	3	3	3	3	3	2	2	100
24	3	3	4	3	4	4	4	3	4	4	4	3	3	2	2	3	3	3	3	3	2	4	3	3	3	4	4	3	4	99
25	4	4	4	4	4	4	4	3	4	3	3	4	2	1	3	3	3	3	3	3	3	4	4	4	4	4	4	4	105	
26	4	4	4	4	4	4	4	4	4	4	4	4	4	4	3	4	4	4	4	4	3	3	4	4	4	4	4	4	117	
27	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	120	
28	3	4	3	2	3	3	3	3	3	3	4	3	4	3	4	3	4	3	3	4	3	3	3	3	3	3	3	3	96	
29	3	3	4	3	3	3	4	3	4	3	3	4	3	3	3	4	3	3	4	4	3	3	4	4	4	3	4	4	103	
30	4	4	4	2	4	4	3	4	4	4	4	4	4	2	4	4	4	2	3	4	4	4	4	3	3	4	4	4	110	

Lampiran 9. Angket Pengujian

Functionality

INSTRUMEN PENGUJIAN ASPEK *FUNCTIONALITY*

SOFTWARE TES DENGAN ANALISIS BUTIR SOAL BERBASIS *WEB* DI MAN YOGYAKARTA 1

Nama : NUNING APRUMSARI
Pekerjaan : JUNIOR SYSTEM ANALYST
Instansi : PT. SEBANGSA BERSAMA

Petunjuk Pengisian

Berilah tanda centang (v) pada kolom **Ya** jika fungsi berjalan dengan benar dan **Tidak** jika fungsi tidak berjalan dengan benar.

No.	Fungsi	Pernyataan	Lolos	
			Ya	Tidak
Halaman Peserta Tes				
1.	Daftar tes	Fungsi untuk melihat daftar tes yang telah dibuat oleh admin/guru dari halaman peserta tes guna mengikuti tes sudah berjalan dengan benar	✓	
2.	Identitas Peserta Tes	Fungsi untuk melakukan <i>input</i> data peserta tes sudah berjalan dengan benar	✓	
3.	Pelaksanaan tes	Fungsi untuk pelaksanaan tes bagi peserta tes sudah berjalan dengan benar	✓	
4.	Hasil tes	Fungsi untuk melihat hasil tes dari halaman peserta tes sudah berjalan dengan benar	✓	
Halaman Guru (Admin)				
5.	<i>Login</i>	Fungsi untuk masuk ke halaman admin/guru sudah berjalan dengan benar	✓	
6.	Mengubah Akun	Fungsi untuk mengubah akun yang digunakan untuk <i>login</i> sudah berjalan dengan benar	✓	
7.	<i>Reset Akun</i>	Fungsi untuk melakukan <i>reset</i> akun sudah berjalan dengan benar	✓	
8.	<i>Logout</i>	Fungsi untuk keluar dari halaman admin/guru sudah berjalan dengan benar	✓	
9.	Tambah Tes	Fungsi untuk menambah tes sudah berjalan dengan benar	✓	

10.	Edit Tes	Fungsi untuk mengedit tes sudah berjalan dengan benar	✓	
11.	Hapus Tes	Fungsi untuk menghapus tes sudah berjalan dengan benar	✓	
12.	Melihat Tes	Fungsi untuk melihat tes sudah berjalan dengan benar	✓	
13.	Pengaturan Tes	Fungsi untuk mengatur tes sudah berjalan dengan benar	✓	
14.	Duplikat Tes	Fungsi untuk menduplikat tes sudah berjalan dengan benar	✓	
15.	Tambah Soal	Fungsi untuk menambah butir soal sudah berjalan dengan benar	✓	
16.	Edit Soal	Fungsi untuk mengedit butir soal sudah berjalan dengan benar	✓	
17.	Hapus Soal	Fungsi untuk menghapus butir soal sudah berjalan dengan benar	✓	
18.	Melihat Soal	Fungsi untuk melihat sudah berjalan dengan benar	✓	
19.	Lihat Laporan	Fungsi untuk melihat daftar laporan tes sudah berjalan dengan benar	✓	
20.	Hasil Tes	Fungsi untuk melihat rapor hasil tes peserta tes sudah berjalan dengan benar	✓	
21.	Detail Hasil Tes	Fungsi untuk melihat hasil tes secara detail sudah berjalan dengan benar	✓	
22.	Hasil Analisis Butir Soal	Fungsi untuk melihat hasil analisis butir soal sudah berjalan dengan benar	✓	
23.	Cetak Hasil Tes	Fungsi untuk mencetak hasil tes sudah berjalan dengan benar	✓	
24.	Cetak Detail Hasil Tes	Fungsi untuk mencetak detail hasil tes sudah berjalan dengan benar	✓	
25.	Cetak Hasil Analisis Butir Soal	Fungsi untuk mencetak hasil analisis butir soal sudah berjalan dengan benar	✓	

Komentar dan Saran

UX pada tampilan mengerjakan soal /tes perlu dibuat lebih
menarik dan menyenangkan lagi. Atau memfasilitasi jika scrollnya
ke bawah dan panjang pula."

Yogyakarta, 17 September 2014

Responden,



NUNING ARUMSARI

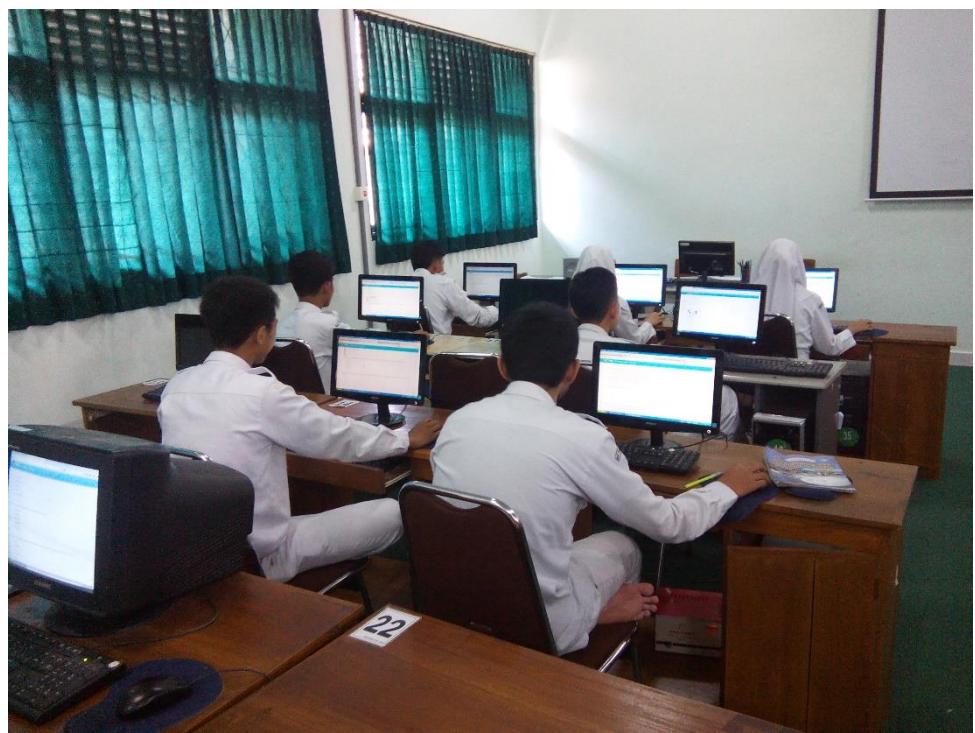
Lampiran 10. Dokumentasi



Gambar 74. Dokumentasi Pengujian *Usability* 1



Gambar 75. Dokumentasi Pengujian *Usability* 2



Gambar 76. Dokumentasi Pengujian *Usability* 3



Gambar 77. Dokumentasi Pengujian *Usability* 4