

**PENGEMBANGAN MODEL TES ESSAY BERBANTUAN KOMPUTER
UNTUK MATA PELAJARAN ELEKTRONIKA DASAR DI SMK N 2
YOGYAKARTA**

TUGAS AKHIR SKRIPSI

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



Oleh:

Fajar Prastya

NIM 10518241003

**PROGRAM STUDI PENDIDIKAN TEKNIK MEKATRONIKA
FAKULTAS TEKNIK
UNIVERSITAS NEGERI YOGYAKARTA
2016**

LEMBAR PERSETUJUAN

Tugas Akhir Skripsi dengan Judul

**PENGEMBANGAN MODEL TES ESSAY BERBANTUAN KOMPUTER UNTUK
MATA PELAJARAN ELEKTRONIKA DASAR DI SMK N 2 YOGYAKARTA**

Disusun Oleh

Fajar Prastya

NIM 10518241003

Telah memenuhi syarat dan disetujui oleh pembimbing untuk dilaksanakan Ujian
Akhir Tugas Akhir Skripsi bagi yang Bersangkutan.

Yogyakarta, November 2015

Mengetahui
Ketua Program Studi,
Pendidikan Teknik Mekatronika

Disetujui,
Dosen Pembimbing



Herlambang Sigit P, S.T,M.Cs
NIP. 19650829 199903 1 001



Dr. Haryanto, M.Pd., MT
NIP. 19620310 198601 1 001

SURAT PERNYATAAN

Saya yang bertanda tangan di bawah ini:

Nama : Fajar Prastya

NIM : 10518241003

Program Studi : Pendidikan Teknik Mekatronika

Judul TAS : Pengembangan Model Tes Essay Berbantuan Komputer
Untuk Mata Pelajaran Elektronika Dasar Di Smk N 2
Yogyakarta

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

Yogyakarta, Agustus 2015

Yang Menyatakan,



Fajar Prastya
NIM. 10518241003

HALAMAN PENGESAHAN

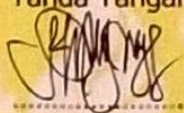

Tugas Akhir Skripsi

**PENGEMBANGAN MODEL TES ESSAY BERBANTUAN KOMPUTER UNTUK
MATA PELAJARAN ELEKTRONIKA DASAR DI SMK N 2 YOGYAKARTA**

Disusun Oleh
Fajar Prastya
NIM 10518241003

Telah dipertahankan di depan Tim Penguji Tugas Akhir Skripsi Program Studi
Pendidikan Teknik Mekatronika Fakultas Teknik Universitas Negeri Yogyakarta
pada tanggal 4 Desember 2015

TIM PENGUJI

| Nama Jabatan | Tanda Tangan | Tanggal |
|---|--|---------|
| <u>Dr. Haryanto, M.Pd., MT</u> Ketua Penguji/Pembimbing |  | 26/1-16 |
| <u>Hartoyo, M.Pd., MT</u> Sekretaris |  | 26-1-16 |
| <u>Zamtinah, M.Pd</u> Penguji |  | 19/1-16 |

Yogyakarta, 27 Januari 2016

Fakultas Teknik Universitas Negeri Yogyakarta

Dekan,



Dr. Moch Bruri Triyono
NIP. 19560216 198603 1 003 &

HALAMAN MOTTO

- ✓ It's hard but not imposible
- ✓ "Kegagalan hanya terjadi bila kita menyerah". Lessing

HALAMAN PERSEMBAHAN

Ayah dan Ibu sebagai tanda bakti, hormat dan rasa terima kasih yang tiada terhingga yang senantiasa memberikan kasih sayang dan dukungan.

Adik dan keluargaku yang selalu memberikan semangat.

Teman-teman seperjuangan Mekatronika 10, kalian selalu luar biasa.

Dosen pembimbing Bpk. Dr. Haryanto, M.Pd., MT. yang selalu sabar membimbing saya.

PENGEMBANGAN MODEL TES ESSAY BERBANTUAN KOMPUTER UNTUK MATA PELAJARAN ELEKTRONIKA DASAR DI SMK N 2 YOGYAKARTA

Oleh
Fajar Prasty
NIM. 10518241003

ABSTRAK

Penelitian pengembangan ini bertujuan untuk memperoleh 1) butir soal tes *essay* yang valid dan reliabel untuk mata pelajaran elektronika dasar, 2) model *software* untuk pengujian tes *essay* yang tepat untuk mata pelajaran elektronika dasar, dan 3) hasil unjuk kerja *software* dalam pengembangan tes *essay* berbantuan komputer.

Metode penelitian yang digunakan adalah 1) penelitian dan pengembangan (*research and development*) model *waterfall* mengacu pada Roger S. Presman (2001) untuk pengembangan *software*, sedangkan pembuatan butir tes *essay* menggunakan langkah-langkah yang dikemukakan oleh Saifuddin Azwar. Penelitian tahap I dimulai dengan pengembangan *software* tes *essay* berbantuan komputer dan pembuatan butir soal. Penelitian tahap II implementasi untuk menguji kinerja program. Responden penelitian yaitu 2 guru dan 30 siswa kelas X Teknik Audio Video SMK N 2 Yogyakarta. Instrumen yang digunakan pengumpulan data meliputi: lembar penilaian untuk ahli perangkat lunak, angket untuk guru, angket untuk siswa dan soal tes *essay*. Responden siswa menggunakan *software* untuk diuji pencapaian kompetensi dengan tes. Validasi instrumen dilakukan oleh expert judgement dan reliabilitas instrumen dihitung dengan rumus alpha cronbach. Teknik analisis data yang digunakan merupakan teknik analisis deskriptif kuantitatif.

Hasil yang diperoleh dari penelitian adalah 1) soal tes *essay* dinyatakan valid oleh expert judgement setelah dilakukan perbaikan sesuai rekomendasi dan reliabilitas soal tes *essay* adalah 0,84 (sangat reliabel); 2) model *software* tes *essay* yang tepat digunakan adalah model tes yang memiliki: halaman *login*, halaman cetak, halaman soal, halaman mata pelajaran, halaman *user log*, halaman *history*, halaman pengguna, halaman admin, halaman konfigurasi. Algoritma dalam proses koreksi adalah: *levenshtein*, *n-gram* dan *jaccard.*, 3) unjuk kerja *software* berdasarkan angket guru diperoleh persentase skor 76,54% dengan kategori baik, sementara angket siswa diperoleh persentase skor 81,08% dengan kategori baik. Berdasarkan hasil tes yang diberikan pada 30 responden diperoleh keakuratan hasil koreksi sebesar 96,67%. Kesalahan dalam proses koreksi ini disebabkan karena *database* kamus yang masih terbatas.

Kata kunci: *research and development*, butir soal *essay*, tes *essay* berbantuan komputer.

KATA PENGANTAR

Puji syukur kehadirat Allah SWT atas berkat rahmat dan karunia-Nya, sehingga Tugas Akhir Skripsi dengan judul "Pengembangan Model Tes Essay Berbantuan komputer Untuk Mata Pelajaran Elektronika Dasar Di SMK N 2 Yogyakarta" dapat disusun sesuai dengan harapan.

Penyusunan Tugas Akhir Skripsi ini dapat diselesaikan sebagai mana mestinya tidak lepas dari bantuan dan kerjasama dengan berbagai pihak. Untuk itu penulis menyampaikan ucapan terima kasih kepada:

1. Dr. Haryanto, M.Pd., MT selaku Dosen Pembimbing Tugas Akhir Skripsi.
2. Dr. Istanto Wahyu Djatmiko, M.Pd. selaku Pembimbing Akademik.
3. Herlambang Sigit Pramono, M.Cs selaku Ketua Program Studi Pendidikan Teknik Mekatronika.
4. Totok Heru Tri Maryadi, M.Pd selaku Ketua Jurusan Pendidikan Teknik Elektro Fakultas Teknik Universitas Negeri Yogyakarta.
5. Dr. Moch Bruri Triyono selaku Dekan Fakultas Teknik Universitas Negeri Yogyakarta.
6. Prof. Dr. Rochmad Wahab, M.Pd, MA., selaku Rektor Universitas Negeri Yogyakarta.
7. Semua pihak, secara langsung maupun tidak langsung, yang tidak dapat disebutkan disini atas bantuan dan perhatiannya selama penyusunan Tugas Akhir Skripsi ini.

Penulis menyadari bahwa penyusunan Tugas Akhir Skripsi ini masih jauh dari kesempurnaan. Oleh karena itu penulis mengharapkan saran dan kritik yang membangun dari para pembaca. Penulis juga berharap agar Tugas akhir Skripsi

ini dapat bermanfaat bagi para pembaca juga bagi pihak lain yang memerlukannya.

Yogyakarta, Agustus 2015

Penulis,

Fajar Prastya
NIM. 10518241003

DAFTAR ISI

| | Halaman |
|--|---------|
| HALAMAN JUDUL..... | i |
| HALAMAN PERSETUJUAN | ii |
| HALAMAN PERNYATAAN..... | iii |
| HALAMAN PENGESAHAN | iv |
| HALAMAN MOTTO | v |
| HALAMAN PERSEMBAHAN | vi |
| ABSTRAK | vii |
| KATA PENGANTAR..... | viii |
| DAFTAR ISI..... | x |
| DAFTAR TABEL..... | xii |
| DAFTAR GAMBAR | xiii |
| DAFTAR LAMPIRAN..... | xiv |
| | |
| BAB I PENDAHULUAN | 1 |
| A. Latar Belakang | 1 |
| B. Identifikasi Masalah | 6 |
| C. Batasan Masalah | 6 |
| D. Rumusan Masalah | 7 |
| E. Tujuan Penelitian | 7 |
| F. Manfaat Penelitian..... | 7 |
| G. Spesifikasi Produk | 8 |
| | |
| BAB II KAJIAN PUSTAKA | 9 |
| A. Landasan Teori | 9 |
| 1. Evaluasi Hasil Belajar | 9 |
| 2. Pengujian Berbasis Komputer..... | 20 |
| 3. Mata Pelajaran Elektronika Dasar | 26 |
| B. Penelitian Yang Relevan..... | 28 |
| 1. Penelitian Relevan Pertama..... | 28 |
| 2. Penelitian Relevan Kedua..... | 29 |
| 3. Penelitian Relevan Ketiga..... | 30 |
| C. Kerangka Berpikir | 31 |
| D. Pertanyaan Penelitian | 33 |
| | |
| BAB III METODE PENELITIAN..... | 35 |
| A. Model Pengembangan | 35 |
| B. Prosedur Pengembangan Instrumen Butir Tes | 43 |
| C. Tempat Penelitian | 44 |
| D. Subjek dan Objek Penelitian | 44 |
| E. Teknik Pengumpulan Data | 44 |
| F. Instrumen Penelitian..... | 45 |
| G. Validitas dan Reliabilitas Instrumen..... | 50 |
| H. Teknik Analisis Data | 51 |
| | |
| BAB IV HASIL PENELITIAN | 53 |

| | |
|---|-----|
| A. Deskripsi Data Uji Coba..... | 53 |
| B. Analisis Data | 89 |
| C. Kajian Produk..... | 99 |
| D. Pembahasan Hasil Penelitian | 100 |
| | |
| BAB V KESIMPULAN DAN SARAN | 106 |
| A. Kesimpulan | 106 |
| B. Keterbatasan Produk | 108 |
| C. Pengembangan Produk Lebih Lanjut | 108 |
| D. Saran | 109 |
| | |
| DAFTAR PUSTAKA | 110 |
| LAMPIRAN | 111 |

DAFTAR TABEL

| | Halaman |
|---|---------|
| Tabel 1. Penguraian Isi dan Bobot Bagian dalam Perancangan Tes | 16 |
| Tabel 2. Levenshtein Algoritm..... | 24 |
| Tabel 3. Kompetensi Elektronika Dasar..... | 27 |
| Tabel 4. Perbandingan Persentase Jawaban dan Nilai | 41 |
| Tabel 5. Data Skor Akhir yang Ditampilkan | 42 |
| Tabel 6. Aspek-aspek Identifikasi Kebutuhan untuk Pembuatan Produk .. | 46 |
| Tabel 7. Aspek-aspek Pengujian Produk..... | 47 |
| Tabel 8. Aspek-aspek Unjuk Kerja <i>Software</i> | 48 |
| Tabel 9. Aspek-aspek Tabel <i>Checklist</i> Dokumen | 49 |
| Tabel 10. Kisi-Kisi Butir Bank Soal | 49 |
| Tabel 11. Validasi Instrumen | 50 |
| Tabel 12. Kategori Data Hasil Penelitian | 52 |
| Tabel 13. Hasil Pengujian Fungsional Program | 87 |
| Tabel 14. Interpretasi Koefisien Korelasi (r_{xy}) untuk Uji Validitas | 90 |
| Tabel 15. Validitas Empiris Butir Soal Tes <i>Essay</i> | 90 |
| Tabel 16. Interpretasi Koefisien Reliabilitas (r_{11}) untuk Uji Reliabilitas | 91 |
| Tabel 17. Klasifikasi Tingkat Kesulitan | 92 |
| Tabel 18. Tingkat kesulitan Butir Soal Tes <i>Essay</i> | 92 |
| Tabel 19. Kriteria Daya Beda Butir Soal | 93 |
| Tabel 20. Daya Beda Butir Soal Tes <i>Essay</i> | 94 |
| Tabel 21. Konversi Skor Aspek Performansi Penggunaan | 94 |
| Tabel 22. Konversi Skor Aspek Materi Tes | 95 |
| Tabel 23. Konversi Skor Aspek Kemanfaatan | 96 |
| Tabel 24. Ketepatan Koreksi Jawaban tes..... | 97 |
| Tabel 25. Nilai Hasil Tes Siswa..... | 98 |
| Tabel 26. Persentase Butir Soal Yang Dikeluarkan Program..... | 100 |

DAFTAR GAMBAR

| | Halaman |
|---|---------|
| Gambar 1. Diagram Jenis-jenis Alat Penilaian | 11 |
| Gambar 2. Langkah Standar Dalam Perencanaan dan Penyusunan Tes..... | 14 |
| Gambar 3. Diagram Pengembangan <i>Software</i> | 32 |
| Gambar 4. Diagram <i>The Linear Sequential Model</i> | 35 |
| Gambar 5. <i>Entity Relational Diagram</i> (ERD)..... | 37 |
| Gambar 6. Data Flow Diagram Level 0 | 38 |
| Gambar 7. Piramida Model Proses Pengembangan <i>Software</i> ... | 38 |
| Gambar 8. Desain Tampilan Halaman Login | 54 |
| Gambar 9. Desain Tampilan Halaman Cetak..... | 56 |
| Gambar 10. Desain Tampilan Halaman Soal | 57 |
| Gambar 11. Desain Tampilan Tambah/ Ubah Soal | 58 |
| Gambar 12. Desain Tampilan Halaman Pelajaran..... | 58 |
| Gambar 13. Desain Tampilan Halaman <i>User Log</i> | 59 |
| Gambar 14. Desain Tampilan Halaman <i>History</i> | 60 |
| Gambar 15. Desain Tampilan Halaman Pengguna..... | 61 |
| Gambar 16. Desain Tampilan Halaman Konfigurasi | 62 |
| Gambar 17. <i>Data Flow Diagram</i> Level 0 | 63 |
| Gambar 18. <i>Data Flow Diagram</i> Level 1 | 63 |
| Gambar 19. File Basis Data..... | 66 |
| Gambar 20. Flow Chart Program Tes Essay Berbantuan Komputer ... | 67 |
| Gambar 21. Halaman Login | 78 |
| Gambar 22. Halaman Cetak..... | 78 |
| Gambar 23. Halaman Soal | 79 |
| Gambar 24. Halaman Tambah dan Ubah Soal | 80 |
| Gambar 25. Halaman Mata Pelajaran..... | 80 |
| Gambar 26. Halaman <i>User Log</i> | 81 |
| Gambar 27. Halaman History | 82 |
| Gambar 28. Halaman Pengguna..... | 83 |
| Gambar 29. Halaman Admin | 83 |
| Gambar 30. Halaman konfigurasi | 84 |
| Gambar 31. Halaman <i>Logout</i> | 85 |
| Gambar 32. Halaman Awal Tes | 86 |
| Gambar 33. Halaman Tes..... | 86 |
| Gambar 34. Halaman Hasil Tes | 87 |
| Gambar 35. Hasil Angket Kemudahan Penggunaan..... | 95 |
| Gambar 36. Hasil Angket Kemudahan Pengubahan Soal..... | 95 |
| Gambar 37. Hasil Angket Kemampuan <i>Software</i> Membantu Guru | 96 |

DAFTAR LAMPIRAN

| | Halaman |
|--|---------|
| Lampiran I. <i>Source Code</i> Program | 111 |
| Lampiran II. Angket..... | 145 |
| Lampiran III. Checklist pengujian produk..... | 151 |
| Lampiran IV. Soal tes dan kunci jawaban | 154 |
| Lampiran V. Surat Keterangan Expert Judgement | 156 |
| Lampiran VI. Surat Izin Penelitian..... | 159 |
| Lampiran VII. Hasil Penelitian dan Analisis Data | 163 |
| Lampiran VIII. User Manual..... | 173 |

BAB I

PENDAHULUAN

A. Latar Belakang

Fakta menyebutkan bahwa sebagian besar (87%) permasalahan guru masih kesulitan dalam memahami cara penilaian kurikulum 2013 (<http://kampus.okezone.com/>, 2013). Hal ini terungkap berdasarkan sebuah survei yang dilakukan oleh dosen Universitas Negeri Semarang. Survei melalui angket itu dilakukan oleh Ani Rusilowati (2013) dalam kegiatan *Professor Go to Schools*.

Penilaian dalam pendidikan merupakan suatu proses pengukuran keberhasilan proses belajar dan mengajar. Penilaian terdiri dari tes dan non tes (Djemari, 2008: 67). Penilaian yang termasuk dalam non tes yaitu observasi, wawancara, skala sikap, dll. Penilaian tes dibagi menjadi tes tertulis dan tes lisan. Tes tertulis secara umum dikelompokkan dalam tes objektif dan subjektif. Tes subjektif terdiri dari tes uraian pendek dan tes uraian panjang.

Bentuk tes objektif maupun subjektif (*essay*) disusun agar dapat mengukur kemampuan siswa yang sebenarnya. Permasalahan lain yang dihadapi guru adalah sulitnya membuat soal tes *essay*. Hal itu terungkap melalui wawancara dengan guru mengenai tes *essay* pada saat peneliti melakukan PPL. Untuk itu pengembangan mengenai soal tes subjektif menarik untuk diteliti lebih lanjut.

Pembelajaran dalam kelas terdiri dari tiga proses yang dilakukan guru yaitu penentuan tujuan pembelajaran, proses kegiatan pembelajaran dan evaluasi hasil pembelajaran. Instrument yang umum digunakan untuk evaluasi

hasil pembelajaran adalah tes *essay*. Guru SMK dalam membuat tes *essay* yang baik merasa kesulitan. Hasil wawancara peneliti saat melaksanakan PPL di SMK N 2 Yogyakarta, guru merasa kesulitan dalam melakukan analisis butir tiap soal terutama pada soal *essay*. Pengamatan di SMK N 2 Yogyakarta guru masih belum melakukan analisis butir soal terutama soal bentuk *essay*. Membuat soal yang memiliki validitas dan reliabilitas tidaklah mudah. Tes buatan guru umumnya mempunyai reliabilitas sedang atau rendah dan disusun tanpa bantuan tenaga ahli (Arikunto, 2004: 147).

Kesulitan guru SMK dalam melakukan analisis butir soal *essay* dapat diselesaikan dengan memanfaatkan bantuan komputer untuk melakukan analisis. *Software* yang sering digunakan untuk melakukan analisis butir soal umumnya berupa aplikasi *Microsoft office excel*. Aplikasi *Microsoft office excel* dalam pengoperasiannya masih menyulitkan guru, sehingga dibutuhkan *software* yang mudah pengoperasiannya.

Model tes *essay* dalam evaluasi pembelajaran merupakan instrumen yang penting untuk mengetahui tingkat perkembangan siswa dalam penguasaan materi. Tes *essay* memiliki keunggulan untuk mengukur aspek kognitif, mengembangkan kemampuan berbahasa, melatih berpikir teratur atau penalaran dan mengembangkan keterampilan problem solving (Sudjana, 2005: 36). Pemberian soal *essay* sebaiknya intensitasnya lebih banyak dari soal objektif karena bisa digunakan untuk memantau kemampuan kognitif siswa.

Sejak tahun 1960-an tes *essay* mulai ditinggalkan karena munculnya bentuk tes objektif (Sudjana, 2005: 35). Kondisi yang sama juga terjadi di SMK N 2 Yogyakarta, guru lebih memilih menggunakan tes bentuk objektif. Menurut

guru tes bentuk objektif dipilih karena lebih mudah dalam melakukan koreksi dan analisis butir soal. Soal *essay* jarang digunakan karena analisisnya dianggap sulit. Penggunaan tes bentuk objektif secara berlebihan dapat menurunkan kemampuan berpikir siswa karena adanya unsur tebakan dalam menjawab pertanyaan.

Soal tes *essay* dapat digunakan untuk meningkatkan Kemampuan siswa SMK dalam aspek kognitif, berpikir, berbahasa yang baik dan benar, dan pemecahan masalah. Soal tes *essay* sebaiknya digunakan guru untuk evaluasi pembelajaran tiap bab, karena lingkup materi yang dapat disusun sempit. Soal tes objektif sebaiknya digunakan untuk evaluasi pembelajaran secara menyeluruh, karena kelebihanannya dapat memuat banyak materi.

Koreksi soal *essay* membutuhkan waktu yang relatif lebih lama. Keburukan soal *essay* adalah waktu koreksinya lama dan tidak bisa diwakilkan oleh orang lain (Arikunto, 2012: 178). Peneliti juga mengalami koreksi soal model *essay* yang lama saat melaksanakan kegiatan Praktik Pengajaran Lapangan (PPL) di SMK N 2 Yogyakarta. Senada dengan ucapan guru-guru di SMK N 2 Yogyakarta koreksi soal *essay* membutuhkan waktu yang lama. Hasil wawancara dengan siswa apabila diadakan tes dengan soal *essay* maka waktu koreksi jawaban setidaknya dibutuhkan waktu satu minggu.

Koreksi jawaban *essay* menurut para ahli bisa dipercepat yaitu dengan memanfaatkan bantuan komputer. Komputer bisa diprogram untuk melakukan koreksi jawaban soal tes model *essay* dengan perbandingan kata. Koreksi soal *essay* dengan memanfaatkan komputer dapat menghemat waktu dan lebih akurat dalam penilaian (Page dan Paulus, 1968: 190).

Siswa SMK dituntut untuk menjadi siswa yang aktif. Siswa dalam proses pembelajaran sebaiknya aktif mempelajari dan menguasai materi sesuai dengan kurikulum. Level kemampuan siswa dalam suatu mata pelajaran dapat diketahui dengan melakukan evaluasi atau tes. Nilai yang diperoleh dari tes dapat menunjukkan penguasaan materi pelajaran seorang siswa. Fungsi penilaian dapat dipergunakan untuk diagnostik kemampuan (Arikunto, 2004: 10). Nilai tes evaluasi pembelajaran dapat diketahui kemampuan siswa dan kelemahan siswa.

Wawancara dan pengamatan siswa di SMKN 2 Yogyakarta diketahui siswa tidak terlalu peduli pada interpretasi nilai, yaitu siswa kurang memperhatikan kelemahan penguasaan materi. Siswa SMK seyogyanya mengambil tindakan untuk mempelajari materi pelajaran yang belum dikuasai. Hasil ulangan hanya diberikan guru berupa sebuah nilai angka tanpa memberi tahu siswa materi apa yang sudah dikuasai dan materi mana yang belum dikuasai.

Kondisi level penguasaan materi pelajaran yang tidak diketahui siswa menyebabkan siswa belum tahu materi apa yang belum dikuasai. Diagnostik nilai yang bisa langsung diketahui oleh siswa diperlukan sebagai bahan evaluasi diri, sehingga diperlukan pengujian tes *essay* berbantuan komputer. Jadi dari suatu nilai maka siswa bisa langsung tahu materi mana yang sebaiknya diperdalam dalam waktu selanjutnya.

Mata pelajaran Elektronika dasar menjadi pelajaran yang diajarkan kepada siswa Jurusan Teknik Audio Video SMKN 2 Yogyakarta. Materi pelajaran elektronika dasar sebaiknya dikuasai siswa. Siswa perlu latihan dalam mengerjakan tes latihan soal terutama soal tes *essay* untuk meningkatkan

penguasaan materi. Bentuk soal uraian objektif sangat tepat digunakan untuk bidang sains, karena kunci jawabannya hanya satu (Djemari, 2008: 70).

Hasil pengamatan di SMKN 2 Yogyakarta siswa jarang melakukan latihan soal untuk mata pelajaran elektronika dasar. Pengujian atau latihan soal *essay* dengan berbantuan komputer belum banyak dikembangkan didunia pendidikan. Komputer sebagai media latihan soal akan mempermudah siswa dalam latihan soal secara mandiri. Siswa secara mandiri dapat mengerjakan latihan soal-soal elektronika dasar. Soal-soal latihan berbantuan komputer yang dikerjakan secara mandiri masih membutuhkan pihak yang melakukan koreksi jawaban.

Penggunaan komputer untuk melakukan pengujian soal tes *essay* sebaiknya dikembangkan lebih lanjut pada proses koreksi. Pengujian soal tes *essay* mata pelajaran elektronika dasar berbantuan perlu dikembangkan sebagai media latihan soal tes *essay* siswa. Media latihan soal ini hendaknya membantu siswa dalam latihan mengerjakan soal elektronika dasar secara mandiri.

Menanggapi permasalahan tentang, guru merasa kesulitan dalam evaluasi pembelajaran dengan soal *essay*, guru jarang memberikan tes *essay*, koreksi soal *essay* yang lama, siswa tidak mengetahui level kompetensi penguasaan materi, perlunya pengembangan tes *essay* mata pelajaran elektronika dasar berbantuan komputer, peneliti bermaksud mengembangkan *software* tes *essay* berbantuan komputer. *Software* ini nantinya memungkinkan guru mudah dalam melakukan penilaian dan siswa dapat berlatih mengerjakan soal secara mandiri karena fungsi koreksi jawaban dilakukan oleh komputer.

B. Identifikasi Masalah

Usaha untuk mengukur tingkat kompetensi dalam suatu mata pelajaran adalah dengan sering mengerjakan soal baik secara klasikal maupun secara mandiri, sedangkan bila terlalu sering latihan soal, maka beban koreksi lembar pekerjaan semakin banyak. Oleh karena itu dibutuhkan aplikasi yang mampu memberika latihan soal maupun tes yang mengoreksi jawaban dengan bantuan komputer. Rincian masalah dalam proses tes *essay* adalah sebagai berikut.

1. Guru merasa sulit melakukan evaluasi pembelajaran dengan tes *essay*
2. Tes bentuk *essay* pada umumnya sulit dianalisis sehingga jarang dilakukan oleh guru.
3. Waktu koreksi untuk soal bentuk *essay* relatif lebih lama.
4. Kebanyakan siswa kesulitan untuk mengetahui level kompetensi penguasaan materi elektronika dasar, sehingga diperlukan pengujian tes *essay* berbantuan komputer.
5. Penggunaan komputer untuk pengujian soal tes *essay* masih belum banyak dikembangkan untuk materi mata pelajaran Elektronika Dasar.

C. Batasan Masalah

Berdasarkan identifikasi permasalahan di atas, maka peneliti perlu membatasi terhadap masalah lain yang mungkin timbul saat penelitian berlangsung. Pembatasan masalah dilakukan pada pengembangan tes *essay* berbantuan komputer untuk mengetahui aspek fungsional *software* dalam melakukan pengujian tes *essay* untuk kemampuan kognitif mata pelajaran Elektronika Dasar Program Studi Keahlian Teknik Audio Video SMK N 2 Yogyakarta.

D. Rumusan Masalah

Penjabaran masalah dapat dirangkum dalam rumusan masalah dengan rincian sebagai berikut.

1. Bagaimanakah validitas dan reliabilitas butir-butir tes *essay* yang disusun untuk pengujian mata pelajaran Elektronika Dasar di SMK?
2. Bagaimana model/bentuk *software* tes *essay* yang tepat?
3. Bagaimana unjuk kerja *software* tes *essay* mata pelajaran Elektronika Dasar di Program Studi Keahlian Elektronika Dasar SMK N 2 Yogyakarta?

E. Tujuan Penelitian

Penelitian pengembangan tes *essay* berbantuan komputer ini memiliki tujuan sebagai berikut.

1. Memperoleh butir soal tes *essay* yang valid dan reliabel untuk mata pelajaran Elektronika Dasar.
2. Memperoleh model *software* untuk pengujian tes *essay* yang tepat untuk mata pelajaran Elektronika Dasar di Program Studi Keahlian Audio Video SMK N 2 Yogyakarta.
3. Memperoleh hasil unjuk kerja *software* dalam pengembangan tes *essay* berbantuan komputer.

F. Manfaat Penelitian

1. Menambah kekayaan *software* sebagai alat bantu pada fungsi evaluasi atau penilaian dalam dunia pendidikan dalam rangka mengikuti perkembangan IPTEK.

2. Mengenalkan tes *essay* dengan bantuan komputer sebagai media tes dan sebagai penilai jawaban untuk mempercepat proses evaluasi pembelajaran.
3. Mempermudah guru dalam melakukan evaluasi atau penilaian siswa terhadap materi yang telah diajarkan sehingga pengambilan keputusan proses KBM pada tahap selanjutnya dapat lebih cepat.
4. Mempermudah siswa yang ingin melakukan latihan mengerjakan soal secara mandiri sehingga tahu level kemampuan dan penguasaan materi.

G. Spesifikasi Produk

Software tes *essay* berbantuan komputer dikembangkan untuk memenuhi kriteria sebagai berikut.

1. Bentuk soal tes merupakan *short essay* dengan jawaban objektif.
2. Memiliki halaman *login* pengguna untuk guru sebagai pembuat soal dan siswa sebagai pengguna *software* tes *essay*.
3. *Software* mampu menyimpan data soal dalam database dan mengambil kembali database soal.
4. *Software* mampu melakukan penilaian jawaban dalam proses tes dengan melakukan perbandingan dengan kunci jawaban yang telah dibuat.

BAB II

KAJIAN PUSTAKA

A. Landasan Teori

1. Evaluasi Hasil Belajar

a. Penilaian Hasil Belajar

Penilaian dari sudut bahasa diartikan sebagai proses menentukan nilai suatu objek. Untuk dapat menentukan suatu nilai atau harga suatu objek diperlukan adanya pembandingan. Penilaian adalah proses membandingkan objek atau program yang dinilai dan kriteria sebagai dasar dalam membandingkan antara kondisi sebenarnya atau apa adanya dengan kriteria atau apa kondisi seharusnya.

Menurut Sudjana (2005: 3) Penilaian adalah proses memberikan atau menentukan nilai kepada objek tertentu berdasarkan suatu kriteria tertentu, proses pemberian nilai tersebut berlangsung dalam bentuk *interpretasi* yang diakhiri *judgment*. Penilaian menurut Azwar (1966: 157) merupakan konversi dari skor mentah kedalam klasifikasi evaluatif menurut norma atau kriteria yang relevan. Klasifikasi evaluatif didasarkan pada suatu norma kelas atau suatu kriteria yang ditentukan terlebih dahulu. Penilaian terhadap hasil belajar dapat digolongkan menjadi dua pendekatan yaitu mengacu pada kriteria dan penilaian mengacu kepada norma.

Menurut Sudjana (2005: 3) Hasil belajar siswa merupakan perubahan tingkah laku, tingkah laku yang dimaksud mencakup beberapa aspek yaitu aspek kognitif, afektif dan psikomotoris. Pengertian senada juga diungkapkan Jihad dan Haris (2008: 14) hasil belajar merupakan pencapaian bentuk perubahan

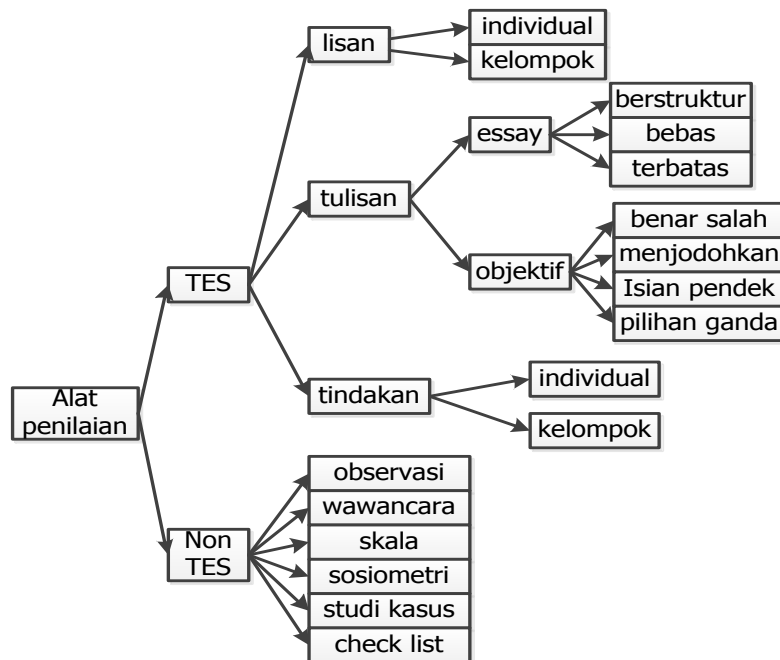
perilaku yang cenderung menetap dari ranah kognitif, afektif, dan psikomotorik dari proses belajar yang dilakukan dalam waktu tertentu. Hasil belajar dapat disimpulkan merupakan berubahnya perilaku seseorang dalam ranah kognitif, afektif, dan psikomotorik yang mana perubahan ini terjadi karena proses belajar dan bersifat tetap.

Penilaian hasil belajar menurut Sudjana (2005: 3) merupakan proses pemberian nilai terhadap hasil belajar yang dicapai siswa dengan kriteria tertentu. Hal ini menunjukkan bahwa hasil belajar siswa merupakan objek penilaian. Penilaian dilakukan dengan menggunakan alat penilaian yang merupakan alat ukur. Alat penilaian dalam pendidikan secara garis besar dapat dijabarkan pada Gambar 1.

Fungsi penilaian menurut Sudjana (2005: 3) adalah: alat untuk mengetahui tercapai tidaknya tujuan instruksional, umpan balik bagi perbaikan proses belajar mengajar dan dasar dalam penyusunan laporan kemajuan belajar siswa kepada orang tuanya. Fungsi penilaian menurut Jihad dan Haris (2008: 56) sebagai pemantauan kinerja komponen-komponen kegiatan proses belajar mengajar dalam mencapai tujuan yang diharapkan dalam proses belajar mengajar. Fungsi penilaian dapat disimpulkan untuk memantau dan mengetahui perkembangan komponen dalam kegiatan belajar dan mengajar sehingga dapat diambil tindakan selanjutnya sesuai dengan tujuan yang ditentukan.

Tujuan penilaian menurut Sudjana (2005: 4) adalah untuk: 1) mendeskripsikan kecakapan belajar siswa, 2) mengetahui keberhasilan proses pendidikan dan pengajaran di sekolah, 3) menentukan tindak lanjut hasil penilaian, 4) memberikan pertanggungjawaban dari pihak sekolah ke pihak yang

berkepentingan. Kecakapan belajar siswa yaitu untuk mengetahui kelebihan dan kekurangan siswa pada mata pelajaran dan level kemampuan siswa dibandingkan dengan siswa lain. Keberhasilan proses pendidikan dan pengajaran di sekolah yaitu untuk mengetahui keefektifan komponen pendidikan dalam mencapai tujuan yang telah ditentukan. Tindak lanjut hasil penilaian yaitu tindakan berupa penyempurnaan proses belajar mengajar maupun merubah hal-hal yang dirasa kurang tepat. Pertanggungjawaban kepada pihak yang berkepentingan yaitu berupa laporan keberhasilan pendidikan maupun hambatan yang terjadi dalam proses pendidikan



Gambar 1. Diagram Jenis-jenis Alat Penilaian (Sudjana, 2005: 6)

b. Penilaian Non Tes

Penggunaan non tes untuk penilaian proses dan hasil belajar mengajar masih terbatas dibandingkan dengan tes. Guru umumnya lebih memilih tes untuk menilai proses dan hasil belajar mengajar karena lebih mudah untuk dibuat dan lebih praktis. Penilaian non tes terdiri dari observasi, wawancara, skala,

sosiometri, studi kasus dan chek list (Sudjana, 2005: 67). Kelebihan non tes adalah kemampuannya untuk menilai dari berbagai aspek kognitif, afektif, maupun psikomotorik.

Penggunaan nontes dalam proses dan hasil pembelajaran antara lain: 1) Observasi umumnya digunakan untuk penilaian perilaku individu (siswa); 2) Wawancara digunakan untuk penilaian dalam ranah kognitif dan afektif siswa; 3) Skala (skala penilaian, skala sikap, skala minat) bisa digunakan untuk menilai siswa dari aspek afektif; 4) Sosiometri biasanya digunakan untuk menilai siswa dari aspek sikap terutama yang berkaitan dengan hubungan social; 5) Studi kasus digunakan untuk mengumpulkan data individu mengenai kasus-kasus tertentu; 6) chek list merupakan kumpulan catatan untuk memperoleh data dan informasi individu secara mendalam.

c. Penilaian Tes

Penilaian tes terdiri dari tes objektif dan tes subjektif. Tes merupakan pertanyaan untuk mendapat jawaban dari siswa dalam bentuk lisan, tulisan atau tindakan (Sudjana, 2005: 35). Ranah yang diukur menggunakan tes umumnya adalah ranah kognitif atau pengetahuan. Tes sebenarnya juga bisa untuk mengukur ranah afektif dan psikomotorik.

d. Tes Objektif

Tes objektif merupakan tes yang banyak digunakan dalam menilai hasil belajar. Tes objektif dipilih karena cakupan materi pelajaran yang luas dan mudah dalam melakukan penilaian. Bentuk soal objektif menurut Nana Sudjana (2005: 44) antara lain: 1) jawaban singkat, 2) benar-salah, 3) menjodohkan, 4)

pilihan ganda. Bentuk tes selain jawaban singkat dalam soal telah tersedia kemungkinan jawaban yang dapat dipilih.

e. Tes Subjektif

1) Tes *Essay*

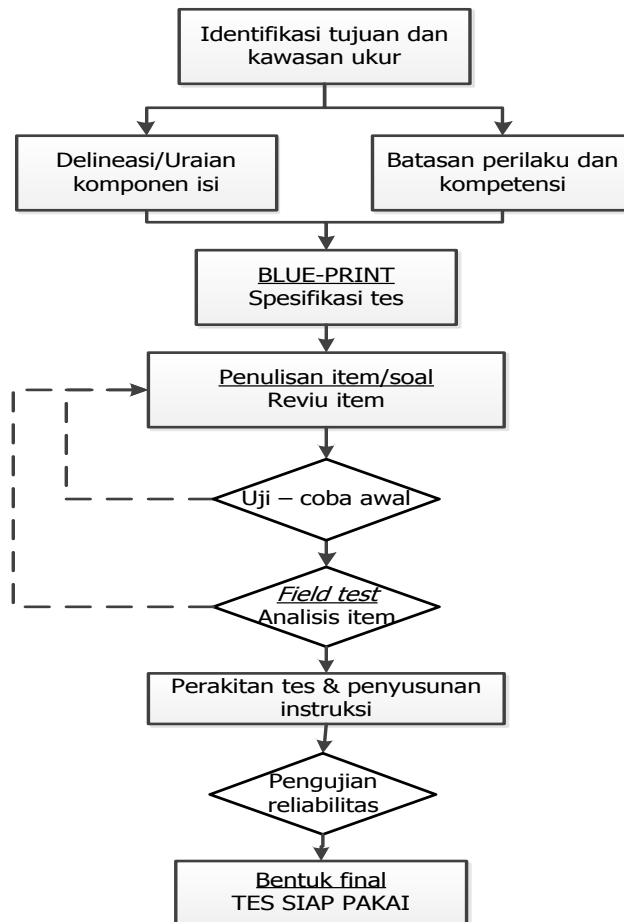
Tes *essay* merupakan salah satu model tes yang paling tua dan sering digunakan dalam evaluasi pembelajaran. Menurut Nana Sudjana (2005: 35) tes *essay* merupakan pertanyaan yang menuntut siswa menjawabnya dalam bentuk menguraikan, menjelaskan, mendiskusikan membandingkan, memberikan alasan dan bentuk lain yang sejenis sesuai dengan tuntutan pertanyaan dengan menggunakan kata dan bahasa sendiri.

Model tes *essay* dibedakan menjadi dua yaitu: a) tes *essay* bebas (uraian bebas); b) tes *essay* terbatas dan terstruktur; (Sudjana, 2005: 37). Tes *essay* bebas jawaban siswa tidak dibatasi, tergantung pada pandangan siswa sendiri. Tes *essay* terbatas memiliki bentuk pertanyaan yang diarahkan pada hal-hal tertentu atau adanya pembatasan tertentu. Model tes *essay* yang lain adalah tes *essay* terstruktur, tes ini dipandang sebagai bentuk antara soal soal objektif dan soal-soal *essay*. Soal terstruktur merupakan soal jawaban singkat, meskipun jawaban bersifat bebas tetapi soal memiliki unsur: a) pengantar soal, b) seperangkat data, c) serangkaian subsoal.

Kelebihan dan kekurangan model tes *essay* menurut Sudjana (2005: 36) adalah sebagai berikut: a) dapat mengukur proses mental yang tinggi atau aspek kognitif tingkat tinggi; b) dapat mengembangkan kemampuan berbahasa, baik lisan maupun tulisan dengan baik dan benar sesuai dengan kaidah-kaidah bahasa; c) dapat melatih kemampuan berpikir teratur atau penalaran, yakni

berpikir logis, analitis dan sistematis; d) mengembangkan kemampuan pemecahan masalah (*problem solving*); e) adanya keuntungan teknis seperti mudah dalam penyusunan soal, guru dapat langsung melihat proses berpikir siswa. Kelemahan soal tes *essay* antara lain: a) sampel tes sangat terbatas sebab dengan tes ini tidak mungkin dapat menguji semua bahan; b) sifatnya sangat subjektif, baik dalam menanyakan, membuat pertanyaan maupun cara memeriksanya; c) tes ini biasanya kurang reliabel, mengungkap aspek yang terbatas, pemeriksaannya memerlukan waktu lama.

f. Penyusunan Tes *Essay*



Gambar 2. Langkah Standar Dalam Perencanaan dan Penyusunan Tes (Azwar, 1996: 54)

Menyusun tes yang berkualitas memerlukan langkah-langkah yang tepat. Tahap awal dalam penyusunan tes adalah dengan perencanaan, yaitu menentukan karakteristik tes yang akan dibuat. Penyusunan tes menurut Azwar (1996: 53-54) dengan mengikuti langkah-langkah standar dalam mengembangkan tes ditunjukkan dengan Gambar 2.

1) Identifikasi Tujuan dan Aspek Pengukuran

Penyusunan tes diawali dengan mengidentifikasi tujuan dan aspek pengukuran. Tujuan tes menurut Azwar (1996: 56) adalah fungsi penempatan, fungsi formatif, fungsi diagnostik, dan fungsi sumatif. Fungsi formatif tes yaitu memperoleh umpan balik kemajuan belajar siswa, tes harus dirancang meliputi semua unit instruksional yang telah diajarkan pada siswa. Identifikasi tujuan dan aspek pengukuran merupakan batasan-batasan yang disusun untuk mempersempit lingkup tes. Batasan-batasan tes yaitu: (a) menentukan mata pelajaran, (b) menentukan di mana soal akan digunakan, (c) menentukan kelas yang akan diuji, (d) menentukan materi pada semester berapa.

Peneliti menentukan batasan dalam penyusunan soal yaitu: (a) mata pelajaran yang ditentukan adalah elektronika dasar, (b) tes akan digunakan pada SMK N 2 Yogyakarta, (c) tes akan diujikan pada kelas 1, (d) materi tes diajarkan pada semester dua. Pembatasan tes yang akan disusun oleh peneliti adalah "Tes Elektronika Dasar SMK N 2 Yogyakarta Kelas X Semester 1".

2) Penguraian Komponen Isi

Penguraian komponen isi pelajaran yang diujikan sangatlah penting. Menurut Azwar (1996: 57-59) penguraian isi tes yaitu usaha untuk menjaga agar tes yang ditulis tidak keluar dari lingkup yang telah ditentukan. Penguraian tes

berarti pula mengusahakan agar jangan sampai ada bagian materi pelajaran penting yang terlewatkan dan tidak tertuang dalam tes. Materi tes yang baik sebaiknya bersifat komprehensif dan berisi item-item yang relevan. Komprehensif menurut Azwar (1996: 56) yaitu tes mencakup seluruh isi atau bahan pelajaran yang telah diidentifikasi, sedangkan relevan yaitu item yang ditulis benar-benar menanyakan materi yang telah diidentifikasi dan segala sesuatu yang dianggap perlu.

Penguraian komponen isi dengan menuangkan dalam bentuk tabel seperti contoh Tabel 1. Nilai bobot dalam penguraian komponen menentukan nilai proporsi jumlah item soal yang disusun. Nilai bobot yang besar maka jumlah item yang harus dibuat juga lebih banyak, dan sebaliknya.

Tabel 1. Penguraian Isi dan Bobot Bagian dalam Perancangan Tes

| No | Uraian Isi | Bobot (%) |
|-------|---|-----------|
| 1. | Identifikasi komponen pasif | 35 |
| 2. | Identifikasi komponen aktif | 35 |
| 3. | Menjelaskan fungsi komponen pasif dan aktif | 30 |
| Total | | 100% |

3) Batasan Perilaku dan Kompetensi

Batasan perilaku dan kompetensi menjadi langkah selanjutnya setelah menguraikan isi. Batasan perilaku Azwar (1996: 59-60) merupakan operasionalisasi tujuan instruksional (*behavioral objectives*). Tujuan instruksional ini biasanya sudah tersedia pada silabus atau garis besar pokok pengajaran. Kompetensi merupakan penerjemahan tujuan instruksional kedalam bentuk konkrit sehingga bisa diukur. Bentuk konkrit dari tujuan instruksional menyatakan dalam taraf kompetensi kognitif atau mental. Dalam kenyataannya tidak semua materi dapat dibuat dalam tingkat kompetensi yang tinggi. Tingkat kompetensi suatu tes juga harus memperhatikan usia dan pendidikan siswa yang akan

dikenai tes. Salah satu pedoman dalam menentukan tingkat kompetensi item tes adalah dengan taksonomi tujuan pendidikan yang dirumuskan oleh Bloom. Taksonomi bloom yaitu mencakup aspek afektif, kognitif dan psikomotorik. Aspek kognitif menurut Bloom yaitu aspek pengetahuan (*knowledge*), pemahaman (*comprehension*), aplikasi (*application*), analisis (*analysis*), sintesis (*synthesis*), dan evaluasi (*evaluation*).

4) Spesifikasi Tes

Spesifikasi tes merupakan tabel yang memuat uraian isi tes dan tingkat kompetensi yang akan diungkap. Table spesifikasi tes sering disebut sebagai *test blue-print*. Pembuatan *blue-print* tes menurut Azwar (1996: 69-72) menjadi suatu pegangan yang sangat membantu dalam penyusunan item tes, yaitu pedoman untuk menjaga agar penulisan item tes menjadi terarah. Tipe item dalam tes ada dua macam yaitu objektif dan subjektif. Pemilihan item yang akan digunakan dalam tes ini menurut Azwar (1996: 74) mempertimbangkan beberapa hal, pertama menyangkut hakikat hasil belajar dan kedua adalah kualitas item yang mungkin dibuat.

5) Penulisan Item Soal

Penulisan item soal dibuat setelah *blue print* telah tersedia. Penyusunan item *essay* menurut Azwar (1996: 106-109) sebaiknya mengikuti beberapa petunjuk, antara lain: (a) Memberikan pertanyaan yang menghendaki siswa agar menunjukkan penguasaan pengetahuan yang penting; (b) Membuat pertanyaan yang arah jawabannya jelas, sehingga para ahli dapat setuju satu jawaban; (c) Jangan menanyakan sikap atau pendapat; (d) Jangan memberi kesempatan siswa untuk memilih dan menjawab sebagian hanya diantara pertanyaan yang

disediakan; (e) Sebaiknya menulis satu jawaban ideal yang dikehendaki, kemudian menyusun pertanyaannya.

Pemberian skor yaitu memberikan angka atau harga dari suatu jawaban. Pemberian skor menurut Azwar (1996: 111) adalah harga kuantitatif suatu jawaban terhadap item dalam tes, dan memperoleh deskripsi mengenai performansi siswa dalam tes. Beberapa pedoman dalam melakukan pemberian skor terhadap tes tipe *essay* menurut Azwar (1966: 118) adalah sebagai berikut: (a) Membuat pedoman penskoran terlebih dahulu yang berisi garis besar atau pokok jawaban yang dikehendaki, kemudian tentukan bobot nilai untuk setiap item; (b) Bila jawaban dapat dibatasi maka diperlukan kriteria-kriteria jawaban yang dianggap benar; (c) Sebelum melakukan pemeriksaan jawaban sebaiknya tidak mengetahui pemilik kertas jawaban; (d) Memeriksa jawaban item pertama untuk seluruh siswa baru kemudian memeriksa nomor berikutnya; (e) Jangan melakukan pemberian skor pada saat kondisi tidak tenang, gembira atau sedang lelah, dsb.

6) Analisis Item Soal

Analisis item yaitu untuk mengetahui keberhasilan suatu tes. Menurut Azwar (1996: 129), tes dikatakan berhasil menjalankan fungsinya bila mampu memberikan hasil ukur yang cermat dan akurat. Kualitas dari suatu tes dapat dilihat dari dua parameter yaitu indeks kesukaran item dan indeks daya diskriminasi.

a) Indeks Kesukaran

Indeks kesukaran item merupakan rasio antara penjawab item soal yang benar dan jumlah penjawab item soal. Indeks kesukaran disimbolkan

dengan huruf p . Secara teori dikatakan bahwa p merupakan probabilitas empirik untuk lulus item soal tertentu bagi kelompok siswa tertentu. Rumus indeks kesukaran item soal adalah:

$$p = n_i/N$$

n_i = banyaknya siswa yang menjawab item soal dengan benar

w = banyaknya siswa yang menjawab item soal

b) Indeks Daya Beda

Indeks daya beda merupakan kemampuan item soal untuk membedakan siswa berkemampuan rendah dan siswa berkemampuan tinggi. Item soal dikatakan mempunyai daya diskriminasi tinggi apabila siswa berkemampuan tinggi bisa menjawab dengan benar sementara siswa dengan kemampuan rendah tidak bisa menjawab dengan benar. Rumus daya diskriminasi item soal adalah:

$$d = \frac{n_{iT}}{N_T} - \frac{n_{iR}}{N_R}$$

n_{iT} = banyaknya penjawab item soal dengan benar dari kelompok tinggi.

n_T = banyaknya penjawab dari kelompok tinggi

n_{iR} = banyaknya penjawab item soal dengan benar dari kelompok rendah.

n_R = banyaknya penjawab dari kelompok rendah.

g. Validitas dan Reliabilitas Tes

Validitas berasal dari kata *validity* yang berarti ketepatan suatu instrument pengukur. Validitas menurut Azwar (1996: 173) validitas mempunyai arti sejauhmana ketepatan dan kecermatan suatu instrument pengukur (tes) dalam melakukan fungsi ukurnya. Suatu tes dikatakan mempunyai validitas yang tinggi apabila tes tersebut mampu menjalankan fungsi ukurnya atau memberikan hasil ukur yang tepat dan akurat sesuai maksud dikenakannya tes tersebut.

Kecermatan suatu tes yaitu bila tes mampu mendeteksi perbedaan-perbedaan yang kecil yang ada pada atribut yang diukurnya.

Reliabilitas berasal dari kata *reliability* yaitu keandalan. Pengukuran yang memiliki reliabilitas tinggi bila dapat menghasilkan data yang reliabel. Reliabilitas menurut Azwar (1996: 180) yaitu bila suatu tes yang diujikan beberapa kali pada suatu subjek yang sama maka akan diperoleh hasil yang sama. Menentukan reliabilitas dapat dilakukan melalui pendekatan umum, yaitu pendekatan tes-ulang (*test-retest*), pendekatan tes sejajar (*alternate-forms*), dan pendekatan konsistensi internal (*internal consistency*). Pendekatan tes ulang dilakukan dengan cara melakukan tes dua kali pada subjek dengan jeda waktu tertentu. Kelemahan tes ulang adalah kurang praktis karena harus melakukan tes dua kali. Pendekatan tes sejajar hanya bisa dilakukan bila tersedia dua instrument pengukur yang dianggap sejajar. Kelemahan pendekatan tes sejajar terletak pada sulitnya menyusun dua alat ukur yang memenuhi persyaratan sejajar. Pendekatan konsistensi internal didasarkan pada data sekali pengenaan satu bentuk alat ukur pada sekelompok subjek, yaitu dengan membelah instrument yang digunakan menjadi beberapa bagian. Teknik komputasi reliabilitas konsistensi internal adalah penggunaan Formula Spearman-Brown, Formula Rulon, Formula Alpha, Formula-formula Kuder Richardson, Formula analisis Varians, dan sebagainya.

2. Pengujian Berbasis Komputer

Penilaian dengan menggunakan komputer digunakan untuk menggantikan PPT (*paper-pencil test*). Menurut Samsul Hadi (2011: 11) pemanfaatan komputer untuk melakukan pengujian disebut dengan *Computerized Based Testing* (CBT),

yaitu secara prinsip sama dengan PPT tetapi soal disajikan dengan bantuan komputer. Penyajian dengan bantuan komputer ini memungkinkan dilakukan pengacakan soal yang diberikan, sehingga kesempatan peserta tes mencontek menjadi lebih kecil.

Pengujian berbasis komputer menurut Lawrence M. Rudner (gmac.com: 2012) dibagi menjadi beberapa jenis, antara lain: *computerized based testing* (CBT), *fixed form computerized based testing*, *linear on the fly test* (LOFT), *computerized adaptive testing* (CAT). Tempat penyimpanan butir soal dalam CBT dibagi menjadi: 1) dalam komputer individu, 2) dalam komputer server melalui *local area network* (LAN), 3) dalam komputer server melalui internet.

a. Model *Computerized Based Testing*

Computerized based testing merupakan model paling simpel dari pengujian berbasis komputer. Test (CBT) dikelola dan umumnya penilaian dilakukan secara elektronik. Berbagai tipe pertanyaan dapat ditampilkan melalui layar komputer dan merupakan konversi dari tes dengan menggunakan media kertas. Model pertanyaan berupa pertanyaan objektif maupun subjektif. Pertanyaan mungkin disimpan pada komputer individu, *local area network* (LAN), atau pada komputer server luar dengan media komunikasi internet. CBT memungkinkan tes dapat dikelola dan dinilai lebih akurat, cepat dan lebih aman dari tes menggunakan kertas.

b. Model *Fixed Form Computerized Based Testing*

Fixed form CBT merupakan tes dimana semua peserta tes memperoleh soal yang sama. *Fixed form* CBT merupakan model *paper and pencil* (PPT) test yang dalam penyajiannya diubah menggunakan bantuan komputer. Pola

penyajian tes *fixed form* CBT yang membedakannya dengan PPT adalah dengan perubahan urutan penyajian butir soal kepada peserta tes. Kesempatan menyontek pada model *fixed form* CBT cukup besar karena butir soal secara isi sama. Kelebihan *fixed form* CBT butir soal yang dikembangkan relatif lebih sedikit.

c. Model *Linear On the Fly Test* (LOFT)

LOFT merupakan pengembangan lebih lanjut dari CBT. Pada LOFT komputer memiliki koleksi butir soal yang berjumlah banyak dan bervariasi dari tingkat kesulitan butir soal. Peserta tes pada model LOFT menerima butir soal yang bervariasi dan berbeda pada setiap peserta tes. Model LOFT butir soal yang disajikan walaupun berbeda secara isi namun secara karakteristik statistik memiliki bobot yang sama. Kelebihan tes ini yaitu peserta tes memiliki kesempatan menyontek yang kecil karena butir soal yang berbeda. Kelemahan LOFT yaitu butir soal yang dikembangkan untuk *bank* soal berjumlah banyak.

d. Model *Computerized Adaptive Testing* (CAT)

Pengujian komputer yang lebih maju adalah *Computerized Adaptive Testing* (CAT). CAT memanfaatkan kemampuan komputasi dari komputer untuk memilih butir soal yang tepat. Butir soal yang diberikan pada peserta tes disesuaikan dengan tingkat kemampuan peserta tes. Tingkat kemampuan peserta tes ini dilihat dengan data respon tiap butir soal. peserta tes jika menjawab butir soal dengan benar maka tingkat kesulitan butir soal dinaikkan, sebaliknya jika menjawab butir soal tidak tepat maka tingkat kesulitan butir diturunkan. Peserta tes model CAT akan lebih aman dari kemungkinan menyontek karena butir soal berbeda. CAT memiliki kelemahan yaitu lebih mahal

dalam pengembangannya dan butir soal yang dikembangkan untuk *bank* soal berjumlah banyak.

e. Pengembangan Tes *Essay* Berbantuan Komputer

Tes *essay* berbantuan komputer merupakan bentuk dari CBT yang dikembangkan peneliti. Tes *essay* menggunakan bantuan komputer untuk menyajikan tes dan menilai tes. Tes *essay* berbantuan komputer mempunyai dua fungsi utama yaitu untuk membuat soal *essay* dan menilai jawaban *essay* soal. Jawaban *essay* dalam melakukan koreksinya menggunakan metode *string similarity* yaitu dengan membandingkan jawaban soal dengan jawaban yang ada pada *bank* soal.

Software Model Tes Essay Berbantuan Komputer dalam memberi skor berdasarkan perbandingan text kunci jawaban dan jawaban. Perbandingan text menurut Gomaa dan Fahmy (2012) dibagi menjadi dua yaitu: 1) *String based Similarity*, merupakan penentuan kesamaan kata berdasarkan urutan huruf dan komposisi huruf, *String based similarity* dibagi menjadi 13 jenis yang terdiri dari 6 *character based distance measures* dan 7 *term based distance measures*; 2) *Corpus Based Similarity*, merupakan penentuan persamaan kalimat dengan mencari sinonim dari tiap kata dari *database (library)*.

Character based distance measures diantaranya: 1) *Damerau-Lavenshtein distance*, 2) *Jaro algorithm*, 3) *Jaro-Winkler distance*, 4) *Needleman-Wunsch algorithm*, 5) *Smith-Waterman algorithm*, 6) *N-Gram similarity algorithm*. *Term based distance measures* diantaranya: 1) *Block Distance*, 2) *Cosine Similarity*, 3) *Dice's coefficient*, 4) *Euclidean distance*, 5) *Jaccard similarity*, 6) *Matching coefficient*, 7) *Overlap coefficient*. *Corpus based similarity* diantaranya: 1) *Latent*

semantic analysis, 2) *Explicit semantic analysis*, 3) *Pointwise mutual information*, 4) *Extracting DIS tributionally similar word using CO-occurrences (DISCO)*.

Algoritma yang akan digunakan pada Model Tes *Essay* Berbantuan Komputer adalah: 1) *Damerau-Levenshtein*, 2) *N-Gram similarity Algorithm*, 3) *Jaccard similarity* dan 4) *Extracting DIS tributionally similar word using CO-occurrences (DISCO)*. Algoritma dipilih berdasarkan penelitian yang dilakukan oleh Gomaa dan Fahmy (2012) dengan hasil algoritma terpilih mempunyai hasil penilaian soal *essay* yang paling baik.

Algoritma yang pertama yaitu teknik yang dikenal dengan sebutan *Damerau-Levenshtein Algorithm*. Proses koreksi yaitu dengan menempatkan kunci jawaban dan jawaban pada matrik untuk kemudian dicari perbedaannya. Cara kerja *Levenshtein algorithm* dapat dilihat pada Tabel 2. Apabila ada kesamaan teks maka nilai dalam matrik diagonal akan tetap. Contoh Tabel 2 menunjukkan bila pada kata RESISTOR dan INDUKTOR terdapat 5 perbedaan huruf dari 8 huruf yang ada, hal ini menunjukkan dua kata tersebut berbeda.

Tabel 2. Levenshtein Algoritm

| | | R | E | S | I | S | T | O | R | |
|---|---|---|---|---|---|---|---|---|---|---|
| I | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | |
| N | 1 | 1 | 2 | 3 | 3 | 4 | 5 | 6 | 7 | |
| D | 2 | 2 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | |
| U | 3 | 3 | 3 | 3 | 4 | 5 | 6 | 7 | 8 | |
| K | 4 | 4 | 4 | 4 | 4 | 5 | 6 | 7 | 8 | |
| T | 5 | 5 | 5 | 5 | 5 | 5 | 6 | 7 | 8 | |
| O | 6 | 6 | 6 | 6 | 6 | 6 | 5 | 6 | 7 | |
| R | 7 | 7 | 7 | 7 | 7 | 7 | 6 | 5 | 6 | |
| R | 8 | 8 | 7 | 8 | 8 | 8 | 8 | 7 | 6 | 5 |

Algoritma kedua yaitu *N-Gram similarity Algorithm*. Fungsi dari *N-Gram Similarity Algorithm* adalah membentuk kombinasi kata yang bisa dibentuk dari suatu kalimat. N merupakan jumlah kata yang dibentuk, untuk 2 kata maka

disebut *2-Gram* dan 3 kata disebut *3-Gram*. Misalkan: "Resistor adalah komponen elektronik dua kutub yang didesain untuk mengatur tegangan listrik dan arus listrik" maka *2-Gram* adalah: "Resistor adalah, adalah komponen, komponen elektronik, dst". *N-Gram* kunci jawaban dan *N-Gram* jawaban siswa inilah yang nanti dibandingkan untuk mencari ada kesamaan atau tidak.

Algoritma ketiga yaitu *Jaccard similarity*. *Jaccard similarity* berfungsi untuk mencari perbandingan kata yang sama diantara dua kalimat. *Jaccard* merupakan jumlah kata yang sama antara kunci jawaban dan jawaban siswa berbanding terbalik dengan jumlah kata keseluruhan. Rumus dalam perhitungan *Jaccard similarity* adalah:

$$Jaccard = \frac{|A \cap B|}{|A \cup B|}$$

A= Kata yang terdapat pada kalimat 1 (Kunci jawaban)

B= Kata yang terdapat pada kalimat 2 (jawaban siswa)

Algoritma keempat yang digunakan yaitu *Extracting DIS tributionally similar word using CO-occurrences* atau disingkat dengan nama *DISCO*. *DISCO* merupakan merupakan proses mencari kata yang mempunyai arti sama (bersinonim). Sinonim kata dalam jawaban siswa dicari dari sebuah *database* kamus (*library*). Apabila kata dalam kunci jawaban memiliki arti yang sama dengan kunci jawaban maka jawaban dianggap sama.

Proses kerja *software* Model Tes *Essay* Berbantuan Komputer terdiri dari tiga tahap. Tahap pertama adalah menentukan persamaan jawaban berdasarkan *string based similarity* atau persamaan huruf dengan kunci jawaban. Tahap pertama ini jawaban siswa diolah berdasarkan tiga metode yaitu 1) *Raw*, merupakan data asli dari jawaban siswa tanpa melakukan Natural Language

Processing (NLP); 2) *Stop word*, merupakan proses untuk menghilangkan kata yang termasuk dalam *database stop word* pada kalimat jawaban siswa. Daftar kata *stop word* yang digunakan dalam *database* merujuk hasil penelitian Fadillah (2003); 3) *Stem*, merupakan proses untuk menghilangkan imbuhan dalam suatu kata dan mencari kata dasar. Contoh imbuhan dalam suatu kata yaitu: (a) kata ganti orang: dia, -nya, mereka, dll; (b) imbuhan: kah, lah, pun, dll.

Tahap kedua yaitu mencari persamaan kata dalam jawaban siswa dalam *database (library)* yang telah ada. Dalam tahap ini dilakukan proses menghilangkan kata yang termasuk dalam *stop word*, kemudian memisahkan tiap kata dan membuat matrik persamaan. Setelah membuat matrik persamaan kemudian menentukan nilai berdasarkan nilai maksimal kesamaan dan nilai rata-rata kesamaan.

Tahap ketiga yaitu menentukan nilai akhir kesamaan jawaban siswa dengan kunci jawaban berdasarkan nilai dari *String Based Similarity* dan *Corpus Based Similarity*. Nilai hasil ketiga ini merupakan hasil akhir dan merupakan nilai kesamaan jawaban siswa dengan kunci jawaban soal.

3. Mata Pelajaran Elektronika Dasar

SMK Negeri 2 Yogyakarta merupakan salah satu SMK yang membuka Program Keahlian Teknik Audio Video. Tujuan Program Keahlian Teknik Audio Video secara umum mengacu pada isi Undang-undang Sistem Pendidikan Nasional (UU SPN) pasal 3 mengenai Tujuan Pendidikan Nasional dan penjelasan pasal 15 yang menyebutkan bahwa pendidikan kejuruan merupakan pendidikan

menengah yang mempersiapkan peserta didik terutama untuk bekerja dalam bidang tertentu.

Tabel 3. Kompetensi Elektronika Dasar

| No | Kompetensi Dasar | Indikator |
|----|--|---|
| 1. | Menggambar simbol dan sifat komponen pasif dan aktif | Menggambar simbol dan mengetahui sifat resistor, kapasitor, induktor, transformator, dioda, transistor, UJT, FET, MOSFET, Thyristor, photo resistor, photo dioda, photo transistor, SCR |
| 2. | Mengidentifikasi komponen elektronika semikonduktor dan optik sesuai <i>data sheet</i> | Mengidentifikasi resistor, kapasitor, induktor, transformator, dioda, transistor, UJT, FET, MOSFET, Thyristor, photo resistor, photo dioda, photo transistor, SCR |
| 3. | Mengoperasikan CRO, dan <i>frequency counter</i> untuk pengukuran tegangan dan frekuensi pada rangkaian elektronika analog dasar | - Memahami pentingnya CRO - Mengetahui besaran yang dapat diukur dengan CRO - Mengetahui pengaturan tombol CRO - Mengetahui cara pembacaan CRO |
| 4. | Menggambarkan beberapa simbol gerbang logika kedalam skema rangkaian digital | Menggambar simbol gerbang logika NOT, AND, OR, NAND, NOR, EX-OR, EX-NOR |
| 5. | Menentukan gerbang logika TTL dari gerbang dasar berdasarkan hukum-hukum logika dasar, dan aljabar boolean | Memahami karakteristik input dan output masing-masing gerbang logika dasar |
| 6. | Menentukan ekivalen gerbang logika dasar dengan menggunakan komponen relay dan semikonduktor | Memahami ekivalen gerbang dasar dengan komponen relay dan semikonduktor |
| 7. | Menerapkan sensor dan tranduser pada rangkaian elektronika analog | Menerapkan sensor dan tranduser pada rangkaian elektronika analog |

Program Keahlian Teknik Audio Video SMK N 2 Yogyakarta memiliki tujuan khusus membekali peserta didik dengan keterampilan pengetahuan dan sikap agar kompeten dalam hal: a) Program Keahlian Teknik Audio Video agar dapat bekerja baik secara mandiri atau mengisi lowongan pekerjaan yang ada di

dunia usaha dan dunia insdutri sebagai tenaga kerja tingkat menengah; b) memilih karir, berkompetisi, dan mengembangkan sikap profesional dalam Program Keahlian Audio Video.

Mata pelajaran yang sebaiknya dikuasai oleh siswa Program Studi Audio Video untuk mencapai tujuan tersebut diatas adalah Teknik Elektronika yang memiliki tujuh kompetensi dasar seperti Tabel 3.

Penelitian pada mata pelajaran ini bertujuan untuk mengembangkan soal *essay* pada mata pelajaran elektronika dasar SMK N 2 Yogyakarta prodi TAV kelas X. Selanjutnya peneliti mengembangkan *software* untuk membantu penyajian tes *essay* dan koreksinya. Soal kemudian digunakan pada *software* yang dikembangkan oleh peneliti.

B. Penelitian Yang Relevan

1. Penelitian Relevan Pertama

Penelitian pengembangan pernah dilakukan oleh Heru Miftakhudin dengan judul "Pengembangan *Software* Evaluasi Hasil Belajar Ekonomi SMA Berbasis Teknologi Informasi". Penelitian ini jenisnya adalah penelitian dan pengembangan. Penelitian dilaksanakan di SMA Negeri 1 Buluspesantren pada bulan Desember 2013, subjek penelitian terdiri dari 2 dosen ahli, 2 guru mata pelajaran, dan 40 siswa kelas XI IPS SMA Negeri 1 Buluspesantren. Teknik pengumpulan data menggunakan angket dan wawancara. Teknik analisis data menggunakan analisis deskriptif persentase.

Hasil penelitian berupa *software* evaluasi hasil belajar yang dapat digunakan dalam evaluasi pembelajaran. Ahli media menilai kelayakan *software* dari aspek umum dikatakan "sangat baik", dari aspek komunikasi visual

dikatakan "baik". Sedangkan ahli materi menilai kelayakan *software* dari aspek umum dikatakan "baik", dan dari aspek substansi materi dikatakan "sangat baik". Hasil uji coba lapangan diperoleh komentar guru mata pelajaran terhadap *software* dari aspek umum dikatakan "sangat baik", dari aspek substansi materi dikatakan "baik", dari aspek rakayasa perangkat lunak dikatakan sangat baik, dan dari aspek visual di katakan "baik". Sedangkan siswa mengomentari dari aspek umum dan aspek komunikasi visual keduanya dikatakan "baik".

2. Penelitian Relevan Kedua

Penelitian pengembangan kedua dilakukan oleh Tika Puspita Sari dengan judul "Pengembangan *Software* Notasi Angka". Penelitian ini menggunakan metode *Research and Development* (R&D) produk yang telah divalidasi oleh para ahli, kemudian diujikan kepada responden untuk mengetahui tingkat kelayakan dan keefektifan dari produk yang dihasilkan. Responden dalam penelitian ini berjumlah 30 orang terdiri dari 10 musisi yang memahami not angka dan not balok, 10 musisi yang memahami not angka serta 10 musisi yang tidak memahami notasi balok dan notasi angka. Setiap responden mengisi angket dan *checklist* yang dibagikan berdasarkan pengamatan terhadap produk yang dipakai. Dari 18 butir yang dianalisis ternyata butir nomor 5,6 dan nomor 17 tidak valid, maka angket yang dibagikan mempunyai 15 butir soal.

Hasil penelitian menemukan bahwa skor total reponden berada pada rentang baik mendekati sangat baik yaitu 90,39% yang berarti produk yang dihasilkan layak pakai dan efektif untuk membantu musisi dalam hal menulis karya dalam bentuk notasi angka. Skor total dari semua responden berjumlah 1627, dari skor maksimal adalah 1800.

3. Penelitian Relevan Ketiga

Penelitian pengembangan kedua dengan judul " Pengembangan *Computerized Adaptive Testing (CAT)* dengan algoritma Logika *Fuzzy* dilakukan oleh Haryanto, Desertasi Progam Pascasarjana Universitas Negeri Yogyakarta pada tahun 2009. Penelitian ini menggunakan pendekatan *Research and Development (R & D)*. Penelitian terdiri dari dua tahap: 1) Pengembangan program CAT dengan algoritma logika *Fuzzy*; 2) Pengujian produk program CAT dilakukan di Sekolah Menengah Atas (SMA) Negeri 6 Yogyakarta kelas XII sebagai sampel. Teknik pengambilan data dilakukan dengan observasi, dokumentasi dan angket. Teknik analisis data yang digunakan adalah deskriptif kuantitatif. Analisis data hasil tes dilakukan menggunakan dua pendekatan, yaitu: logika *Fuzzy* untuk analisis data butir-butir tes hasil kerja program CAT, dan likelihood maksimum untuk analisis estimasi kemampuan siswa berdasar butir-butir tes yang diperoleh.

Hasil penelitian pengembangan produk program CAT dengan logika *Fuzzy* meliputi: 1) Pengembangan program CAT berdasarkan pada kebutuhan pemakai yaitu: mudah digunakan, tampilan interaktif, memiliki sistem keamanan, mudah diakses dan mengacu standar kekinian; 2) Program CAT dapat mengenali tiga macam pengguna yaitu pengguna administrator, pengguna guru, dan pengguna siswa; 3) Program CAT memiliki tiga macam basis data yaitu: data pengguna, bank soal, dan hasil tes siswa; 4) Program CAT memiliki sembilan menu utama. Masing-masing memiliki submenu sesuai dengan keperluan dan jenis penggunaanya; 5) Sistem kerja CAT dalam memilih butir tes menggunakan model sistem inferensi algoritma logika *Fuzzy* dengan metode Tsukamoto; 6) model

tampilan program CAT meliputi: tampilan halaman awal, halaman login, halaman menu dan sub-menu, serta halaman tes; 7) Program CAT yang dibuat dapat bekerja melakukan tugasnya sesuai dengan fungsi dan jenis pengguna; 8) dalam pengelolaan tes.

Produk program CAT mampu melakukan fungsinya dalam pengelolaan tes yaitu: (a) administrasi *bank* soal, (b) mengemas butir-butir tes secara otomatis berdasar hasil inferensi *Fuzzy*, (c) pengemasan jumlah butir-butir tes secara otomatis sesuai dengan kemampuan siswa, (d) pilihan jawaban pada setiap butir tes dimunculkan secara acak, dan (e) menyimpan rekaman hasil tes baik secara individu maupun secara bersama-sama.

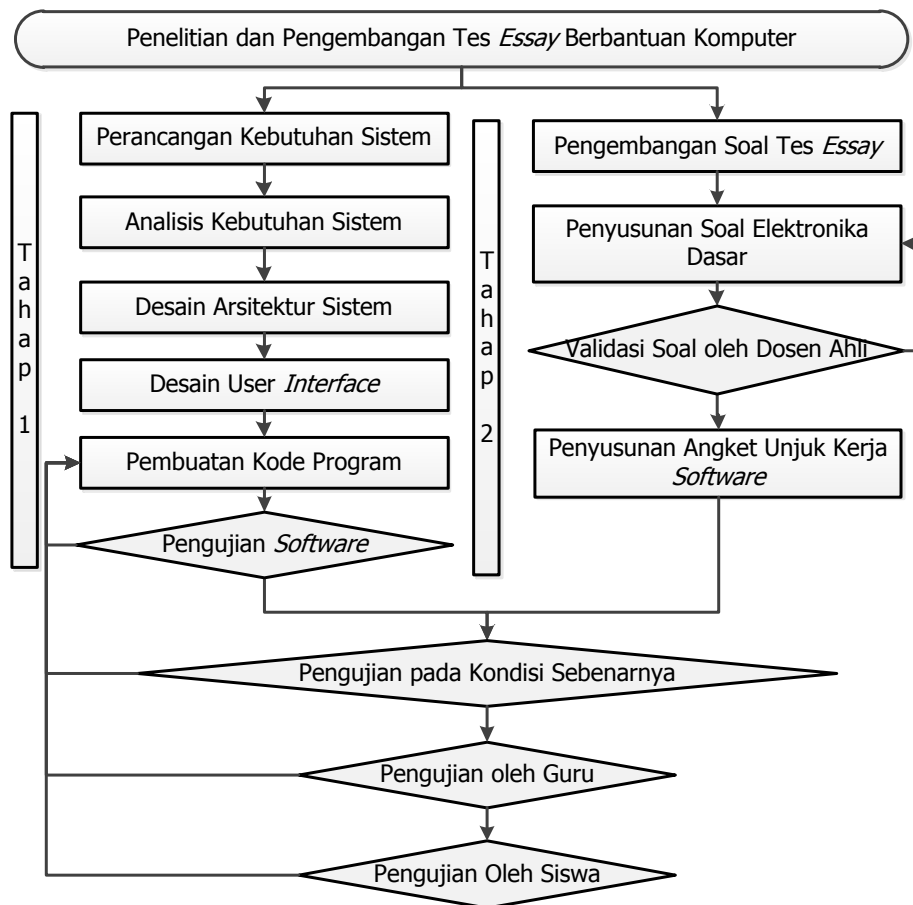
Berdasar hasil analisis data dari penerapan program CAT untuk pengujian kemampuan siswa menunjukkan bahwa: 1) sistem inferensi program CAT dengan algoritma *Fuzzy* mampu memberikan butir-butir tes yang adaptif kepada siswa dengan tepat berdasar benar-salah respon jawaban yang diberikan; 2) kumpulan butir-butir tes hasil kerja program CAT yang diperoleh setiap siswa, dapat mendeskripsikan dengan baik kemampuan masing-masing siswa.

C. Kerangka Berpikir

Berdasarkan kajian teori tersebut di atas, maka kerangka berpikir penelitian ini adalah sebagai berikut: Langkah awal penelitian dimulai dengan mengembangkan aplikasi model tes *essay*. Pengembangan aplikasi dilakukan dengan urutan: 1) pengembangan *form* login, 2) pengembangan *form* guru untuk fungsi pembuatan dan pengeditan soal, 3) pengembangan *form* siswa untuk pengerjaan soal, 4) pengembangan fungsi akses database soal dan nilai,

5) pengembangan fungsi koreksi soal, 6) pengembangan *form* menampilkan nilai siswa, 7) pengembangan *form* admin untuk fungsi administrator.

Langkah selanjutnya adalah pengembangan soal-soal *essay* untuk mata pelajaran elektronika dasar. Mata pelajaran elektronika dasar diajarkan pada siswa SMK N 2 Yogyakarta program studi Teknik Audio Video kelas X. Tes sebelum dimasukkan pada database *software* telah divalidasi oleh dosen ahli. Selanjutnya dilakukan penyusunan angket untuk mengetahui unjuk kerja *software* model tes *essay* untuk mata pelajaran elektronika dasar.



Gambar 3. Diagram Pengembangan *Software*

Langkah yang dilakukan selanjutnya adalah menguji kelayakan *software* model tes *essay*. Fungsi *software* divalidasi oleh dosen ahli untuk menilai unjuk kerja *software*. Masukan para ahli kemudian digunakan untuk memperbaiki kerja *software* sehingga menjadi lebih baik.

Langkah selanjutnya yaitu menguji *software* pada penggunaan sebenarnya. Tempat mengambil sampel yaitu SMK N 2 Yogyakarta Program studi Teknik Audio Video. *Software* akan diujicoba oleh guru dan siswa, selanjutnya guru dan siswa diminta untuk mengisi angket.

D. PERTANYAAN PENELITIAN

Berdasarkan uraian sebelumnya fokus pengembangan dapat dirumuskan dalam pertanyaan penelitian:

1. Bagaimanakah tingkat validitas soal model tes *essay* berbantuan komputer dalam pengujian Elektronika Dasar?
2. Bagaimanakah tingkat reliabilitas soal model tes *essay* berbantuan komputer dalam pengujian Elektronika Dasar?
3. Bagaimanakah tingkat kesulitan soal model tes *essay* berbantuan komputer dalam pengujian Elektronika Dasar?
4. Bagaimanakah daya beda soal model tes *essay* berbantuan komputer dalam pengujian Elektronika Dasar?
5. Bagaimanakah kemudahan pengoperasian *software* tes *essay* bagi pengguna?
6. Bagaimanakah kemudahan pengubahan soal tes dalam *software* tes *essay* berbantuan komputer?

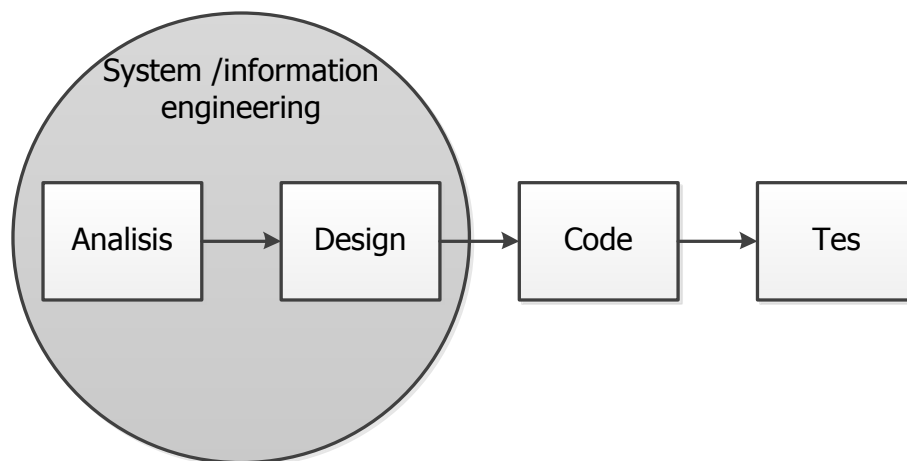
7. Bagimanakah kemampuan *software* dalam membantu guru melakukan tes *essay*?
8. Bagimanakah ketepatan koreksi hasil tes *essay* pada *software* yang dikembangkan?
9. Bagimanakah kecepatan koreksi hasil tes *essay* pada *software* yang dikembangkan?
10. Bagimanakah keakuratan koreksi hasil tes *essay* pada *software* yang dikembangkan?

BAB III

METODE PENELITIAN

A. Model Pengembangan

Penelitian ini menggunakan pendekatan penelitian *Research and Development*. Penelitian ini menggunakan dua model yaitu: 1) Pengembangan *Software Tes Essay Berbantuan Komputer*, 2) Pengembangan instrumen test. Penelitian terdiri dari dua tahap yaitu: pengembangan produk dan implementasi produk. Pada pengembangan tes *essay* menggunakan model Azwar. Tahap pengembangan produk yaitu dilakukan pengembangan *software Tes Essay Berbantuan Komputer*. Tahap pertama penelitian yaitu pengembangan *software model tes essay berbantuan komputer*.



Gambar 4. Diagram *The Linear Sequential Model* (Pressman, 2001: 29)

Pengembangan aplikasi model tes *essay* berbantuan komputer menggunakan metode the *linear sequential model* yang seringkali disebut dengan *classic life cycle* atau *waterfall model* mengacu pada Pressman (2001, 28-30).

Proses pengembangan *software* mengikuti langkah-langkah sesuai Gambar 3, yaitu: 1) *System/information engineering*, yaitu menentukan persyaratan yang harus dipenuhi oleh *software* untuk bekerja pada suatu perangkat yang telah ada seperti perangkat keras, manusia dan database; 2) Analisis, yaitu merupakan proses menyusun spesifikasi yang diperlukan oleh *software*, menentukan fungsi yang diperlukan, perilaku *software*, performa, dan antarmuka; 3) *Design*, yaitu menentukan struktur data, arsitektur *software*, antarmuka, dan algoritma; 4) *Code*, yaitu mengubah *design* yang dibuat menjadi bahasa yang bisa diproses oleh mesin (komputer); 5) Testing, yaitu pengujian *software* (*Debugging*) dilakukan untuk menemukan kesalahan yang terjadi.

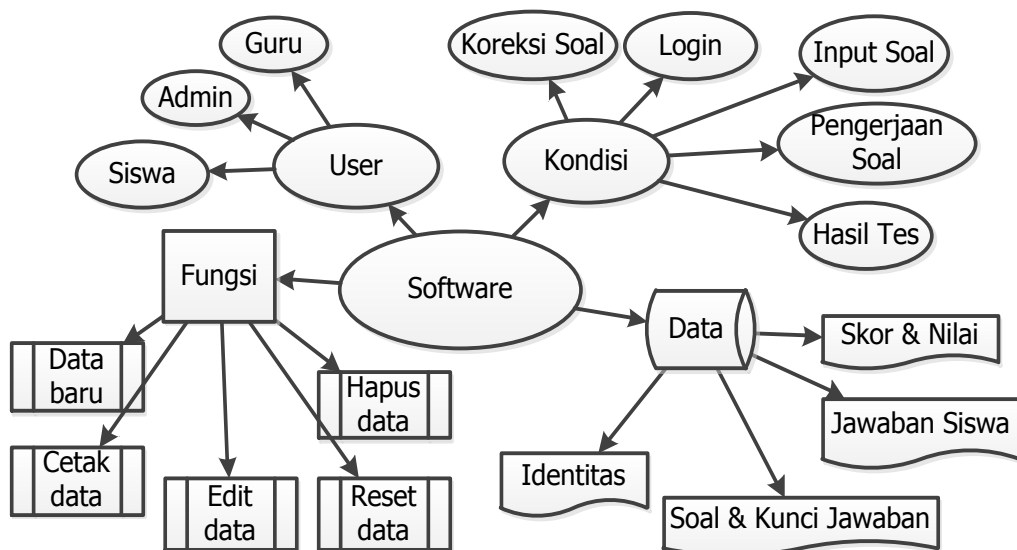
Kesalahan yang kemungkinan terjadi dalam sebuah *software* adalah: a) *Syntax error* (kesalahan kode), b) run time error (kesalahan saat *software* dijalankan), c) *logic error* (kesalahan dari sisi logika). Pemeriksaan pada bagian logika internal *software* berfungsi meyakinkan semua baris program bisa dieksekusi dan fungsi eksternal *software*. Proses implementasi *code* program menggunakan *software Microsoft Visual C# Express 2010* dengan basis bahasa pemrograman C.

1. Analisis

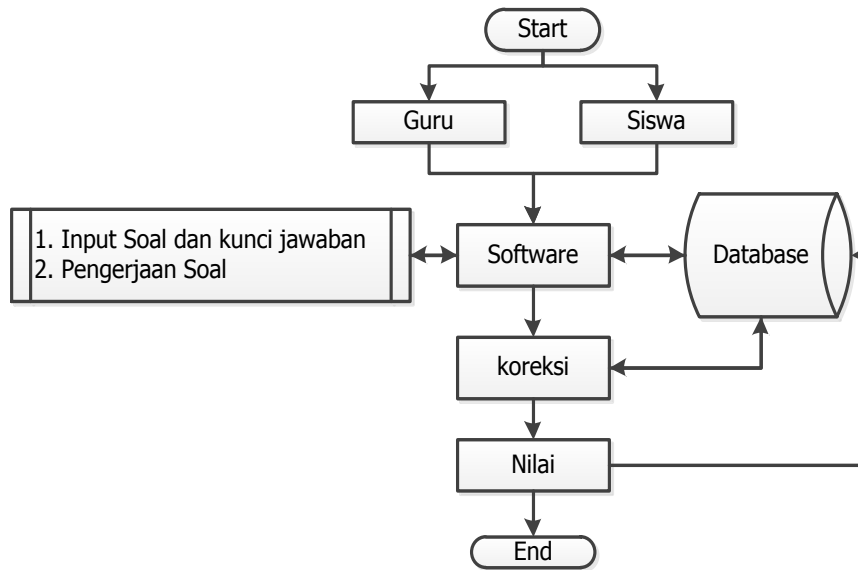
Tahap awal pengembangan *software* model tes *essay* berbantuan komputer yaitu dengan menentukan konsep dan prinsip kerja *software* yang diuraikan sebagai berikut: fungsi yang dibutuhkan, yaitu: (a) akses database, untuk keperluan penyimpanan serta pengambilan soal dan kunci jawaban; (b) menampilkan halaman pengerjaan soal *essay*; (c) pencarian dan pengenalan kata dalam suatu kalimat; (d) menentukan skor jawaban;

Data yang perlu diolah oleh *software* yaitu: (a) data text soal, kunci jawaban, jawaban, nilai huruf, identitas pengguna; (b) data angka skor jawaban, nilai; Perilaku *software* yaitu: (a) menampilkan halaman *login* saat dioperasikan pertama kali, (b) menampilkan butir soal *essay* satu persatu, (c) melakukan pengenalan kata dari jawaban dan membandingkan dengan kunci jawaban.

Elemen model tes *essay* berbantuan komputer yaitu: *Data dictionary*, terdiri dari: (a) data soal, (b) data kunci jawaban, (c) data nilai, (d) data user. *Entity relation diagram*, dapat dilihat pada Gambar 5; *Data flow diagram* level 0, dapat dilihat pada Gambar 6; *State transition diagram*, merupakan status saat *software* bekerja, terdiri dari: (a) *login*, (b) menginput data soal, (c) mengerjakan soal, (d) koreksi jawaban, (e) skor atau nilai akhir.



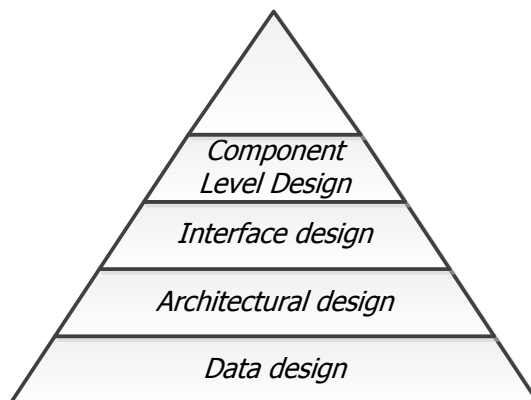
Gambar 5. *Entity Relational Diagram* (ERD)



Gambar 6. *Data Flow Diagram Level 0*

2. Desain

Tahap desain dibagi menjadi dua bagian yaitu tahap desain produk dan tahap desain uji coba. Tahap pertama yaitu desain *software* model tes *essay* berbantuan komputer.



Gambar 7. Piramida Model Proses Pengembangan *Software* (Pressman, 2001: 337)

Desain model pengembangan *software* dapat dilihat pada Gambar 6. Pembuatan *software* dimulai dengan: (a) *Data design*, yaitu menentukan kelompok-kelompok data yang akan diolah oleh komputer dan menentukan

susunan data; Dilanjutkan (b) *Architectural design*, yaitu menentukan pembagian data berdasarkan kategori yang ditentukan untuk mempermudah pengolahan data dan menentukan hubungan antar data yang telah disusun; Tahap selanjutnya (c) *Interface design*, yaitu bagaimana interaksi data antara komputer dan pengguna dilakukan; Tahap terakhir (d) *Component Design*, yaitu membuat komponen yang berfungsi untuk melakukan interaksi komputer dan pengguna.

Tahap kedua adalah uji coba produk untuk mengetahui unjuk kerja Model Tes *Essay* Berbantuan Komputer. Proses tes dengan menggunakan media komputer untuk mengetahui keberhasilan siswa pada mata pelajaran yang diajarkan. Proses pengujian dimulai dengan instalasi *software* dan guru melakukan input data soal dan kunci jawaban yang telah disusun. Setelah konfigurasi *software* selesai siswa mengerjakan butir soal tes sesuai dengan waktu yang telah ditentukan.

Data kunci jawaban yang dimasukkan oleh guru dibagi menjadi beberapa bagian yaitu: (a) Jawaban harus sama persis, yaitu jawaban dianggap benar bila kalimat sama persis dari segi urutan kata; (b) Jawaban terdapat semua kata, yaitu jawaban dianggap benar bila semua kata ada pada kunci jawaban walaupun urutannya tidak sama; (c) Jawaban terdapat salah satu kata, yaitu jawaban dianggap benar bila terdapat salah satu kata pada kunci jawaban; (d) Sinonim kata, yaitu jawaban yang dianggap benar memiliki istilah atau sinonim kata.

a. Desain User Interface

Desain UI software terdiri dari beberapa bagian (a) Halaman awal *software*, (b) Halaman *login*, (c) Halaman menu utama, (d) Halaman Soal, (e)

Halaman tambah dan edit soal, (f) Halaman pelajaran, (g) Halaman user log, (h) halaman *history*, (i) Halaman pengguna/admin, (j) Halaman konfigurasi.

Halaman awal *software* merupakan halaman yang ditampilkan saat *software* pertama kali dijalankan. Pada halaman ini ditampilkan nama *software* dan deskripsi *software*. Halaman kedua yaitu halaman login. Halaman ini berfungsi untuk mengenali pengguna *software* dengan menggunakan nama pengguna dan password.

Halaman ketiga adalah menu utama. Menu utama tiap pengguna akan berbeda berdasarkan hak yaitu administrator, guru dan siswa. Menu utama merupakan halaman yang ditampilkan saat pengguna *software* sukses melakukan login. Halaman keempat adalah halaman soal. Halaman soal berisi daftar soal yang ada pada database, pada halaman soal selanjutnya dapat melakukan navigasi pada halaman tambah soal, edit soal maupun hapus soal.

Halaman kelima adalah halaman tambah soal dan edit soal, merupakan halaman yang ditampilkan apabila pengguna guru ingin menambahkan soal pada database. Halaman keenam merupakan halaman pelajaran. Halaman pelajaran menampilkan daftar pelajaran yang telah ada dalam database dan menampilkan fungsi untuk melakukan penambahan dan pengeditan daftar pelajaran.

Halaman ketujuh merupakan halaman user log. Halaman user log menampilkan daftar tiap mata pelajaran soal mana yang telah dikerjakan disertai detail id soal, jumlah jawaban benar, jumlah jawaban salah, jumlah jawaban kosong, waktu dan nilai. Halaman kedelapan merupakan halaman *history*. Halaman *history* menampilkan daftar pengguna siswa yang telah melakukan tes

disertai dengan detail kode soal, jumlah jawaban benar, jumlah jawaban salah, waktu pengerjaan dan nilai siswa yang bersangkutan.

Halaman kesembilan merupakan halaman pengguna/admin. Halaman pengguna/admin berfungsi bila pengguna adalah admin maka memiliki hak untuk menambah pengguna guru, bila pengguna adalah guru maka memiliki hak untuk menambah pengguna siswa. Halaman kesepuluh merupakan halaman konfigurasi. Halaman konfigurasi menampilkan pengaturan saat guru akan mengadakan tes bagi siswa.

3. Implementasi

Siswa menggunakan software Model Tes Essay Berbantuan Komputer dimulai dengan membuka *software*, dilanjutkan *login* dan mengisi identitas. Pertama kali sebelum mengerjakan soal tes siswa memilih *file* soal tes. *Software* akan menampilkan soal satu persatu untuk dikerjakan siswa. Setelah siswa mengerjakan soal maka siswa akan melakukan *submit* jawaban, selanjutnya software akan melakukan koreksi dengan memeriksa kesamaan kata dan kalimat.

Tabel 4. Perbandingan Persentase Jawaban dan Nilai

| No | Kesamaan Kata (%) | Nilai 1-100 | Nilai 1-4 | Predikat |
|----|-------------------|-------------|-----------|----------|
| 1 | 96%-100% | 96-100 | 4 | A |
| 2 | 91%-95% | 91-95 | 3,66 | A- |
| 3 | 86%-90% | 86-90 | 3,33 | B+ |
| 4 | 81%-85% | 81-85 | 3 | B |
| 5 | 76%-80% | 76-80 | 2,66 | B- |
| 6 | 71%-75% | 71-75 | 2,33 | C+ |
| 7 | 66%-70% | 66-70 | 2 | C |
| 8 | 61%-65% | 61-65 | 1,66 | C- |
| 9 | 56%-60% | 56-60 | 1,33 | D+ |
| 10 | ≤55% | ≤55 | 1 | D |

Proses perbandingan kata dan kalimat akan menghasilkan nilai berupa persentase kesamaan antara jawaban dengan kunci jawaban. Menentukan skor

dari persentase kesamaan jawaban dilakukan sesuai Tabel 4. Skor akhir yang ditampilkan software meliputi nilai dalam skala 1-4 dan predikat dalam skala A-D. Tabel 5 merupakan data yang ditampilkan dalam skor akhir.

Tabel 5. Data Skor Akhir yang Ditampilkan

| Capaian | | |
|------------------|---|-------------------------|
| Mata Pelajaran | : | Elektronika Dasar |
| Pengetahuan | : | 1-4 |
| Predikat | : | A-D |
| Deskripsi | | |
| Pengetahuan | | |
| Catatan | : | Penguasaan materi cukup |

4. Evaluasi

Evaluasi yang dilakukan terhadap program meliputi: (1) *Syntax error*, yaitu mencari apabila ada kesalahan kode pada baris program; (2) *Run time error*, yaitu menguji software dengan dengan menjalankannya dan memberikan variasi masukan yang mungkin terjadi; (3) *Logic error*, yaitu mencari kesalahan software dari segi logika. Pemeriksaan internal software dilihat dengan *white box testing*, yaitu memeriksa urutan eksekusi program langkah demi langkah dan validasi data diolah dalam tiap proses.

Evaluasi dari sisi fungsionalitas program dilakukan dengan uji coba yaitu: (1) *alfa test*, merupakan pengujian *software* pada pengguna pertama (guru) dan (2) *beta test*, merupakan pengujian *software* pada pengguna terakhir (siswa).

Data yang diperlukan untuk melakukan evaluasi yaitu: (1) Data kualitatif berupa fungsi dan modul yang diperlukan *software*, calon pihak pengguna *software*, persyaratan *hardware* minimal yang diperlukan untuk menjalankan *software*, serta hasil pengujian internal *software*; (2) Data kuantitatif yaitu tanggapan yang diberikan oleh dosen ahli dan guru terhadap produk *software*

Model Tes *Essay* Berbantuan Komputer dan hasil unjuk kerja *software* untuk melakukan tes dan melakukan koreksi jawaban.

B. Prosedur Pengembangan Instrumen Butir Tes

Tahap kedua penelitian merupakan pengembangan butir tes yang akan digunakan dalam program. Prosedur pengembangan butir tes mengikuti langkah-langkah yang dikemukakan oleh Azwar (1996, 54) yaitu:

1. Identifikasi Tujuan dan Kawasan Ukur

merupakan pembatasan tes yang disusun berdasarkan standar kompetensi : mengenal komponen dasar elektronika. "Tes Elektronika Dasar SMK N 2 Yogyakarta kelas X Semester 1".

2. Penguraian Komponen Isi, Batasan Perilaku dan Kompetensi

merupakan proses menentukan materi tes berdasarkan kompetensi dasar, mengidentifikasi komponen pasif dan aktif serta menentukan tingkat kesulitan soal berdasarkan taksonomi bloom.

3. Spesifikasi Tes atau *Blue Print Test*

yaitu penyusunan matrik soal dengan mempertimbangkan lingkup materi dan tingkat kesulitan.

4. Penulisan Soal

Proses menulis butir-butir soal berdasarkan matrik yang telah dibuat, butir soal yang disusun digunakan sebagai data bank soal program.

5. Analisis Butir Tes

merupakan proses validasi butir tes. Proses validasi yang digunakan merupakan validasi isi dan konstruk oleh ahli materi selaku validator.

6. Perakitan Tes

proses penyusunan tes untuk digunakan pada proses pengujian, soal tes disusun dengan mengambil lima butir soal dari bank soal dengan komposisi satu soal mudah, tiga soal sedang dan satu soal sulit.

C. Tempat Penelitian

Penelitian *research and development* ini dalam pelaksanaannya dilakukan di: (1) Laboratorium Komputer Pendidikan Teknik Elektro UNY, sebagai tempat pengembangan produk tes essay berbantuan komputer; (2) SMK N 2 Yogyakarta, sebagai tempat implementasi produk tes *essay* berbantuan komputer dalam situasi pembelajaran; (3) Waktu penelitian dilakukan pada bulan Januari 2015 sampai dengan Maret 2015.

D. Subjek dan Objek Penelitian

Subjek yang digunakan dalam penelitian meliputi: (1) Dosen ahli, untuk validasi produk., (2) Guru Mata Pelajaran Elektronika Dasar, untuk identifikasi kebutuhan produk dan ujicoba produk., (3) Siswa kelas X Audio Video, untuk uji fungsionalitas produk.

E. Teknik Pengumpulan Data

Penelitian Tes *Essay* Berbantuan Komputer ini menggunakan metode angket untuk mengumpulkan data penelitian. Metode angket digunakan untuk memperoleh data unjuk kerja *software*. Skala angket yang digunakan adalah angket dengan skala likert. Dokumen yang diperlukan dalam penelitian adalah silabus mata pelajaran Elektronika Dasar. Silabus merupakan sumber dalam menyusun butir tes soal *essay*.

Pengumpulan data juga dilakukan dengan metode tes untuk mengukur kemampuan siswa dan fungsionalitas *software*. Nilai hasil tes siswa merupakan data unjuk kerja *software* dalam melakukan penilaian terhadap jawaban tes *essay*.

F. Instrument Penelitian

Penelitian ini dibagi menjadi dua fase, yaitu: 1) Pengembangan produk berupa software Model Tes Essay Berbantuan Komputer; 2) Pengembangan butir tes. Teknik pengumpulan data yang digunakan saat pengembangan produk adalah: a) *check list* ketepatan kode program (*syntax error*), ketepatan proses (*run time error*), ketepatan hasil (*logic error*) dan prosedur internal software menggunakan *white box testing*; b) Kuesioner fungsi dan unjuk kerja *software*; c) Data tes butir soal. Teknik pengumpulan data yang digunakan saat implementasi produk: a) Data tes unjuk kerja *software* untuk melakukan tes, fungsi koreksi jawaban, dan menilai jawaban; b) Data tes mengenai, jawaban siswa dan nilai siswa.

Berdasarkan metode pengumpulan data yang diuraikan, instrument yang digunakan untuk mengumpulkan data dalam penelitian meliputi:

1. Instrumen Penelitian Identifikasi Kebutuhan

Instrumen identifikasi kebutuhan digunakan untuk menentukan spesifikasi dalam proses pengembangan *software*. Instrument ini berupa form aspek-aspek kebutuhan produk yang diperlukan dalam pengembangan *software*. Tabel 6. merupakan Aspek-aspek Identifikasi untuk Pembuatan Produk

Tabel 6. Aspek-aspek Identifikasi Kebutuhan untuk Pembuatan Produk

| No | Aspek | Sub Aspek | Indikator | Deskripsi | No Butir |
|----|---|--|--|---|----------------|
| 1 | Pemilihan materi | Materi tes | Mata pelajaran yang diujikan | Penentuan mata pelajaran yang akan menjadi bahan tes dalam <i>software</i> . | 1, 2 |
| | | Ranah pengujian | Ranah hasil pelajaran yang diujikan | Penentuan ranah kemampuan yang diuji | 3, 4 |
| 2 | Pemilihan <i>hardware</i> dan <i>software</i> pengembangan produk | Spesifikasi <i>hardware</i> | Spesifikasi prosesor, RAM, Hardisk, dan kecepatan komputer | Penentuan kriteria minimal spesifikasi <i>hardware</i> yang harus dipenuhi untuk mengembangkan Model Tes <i>Essay</i> Berbantuan Komputer | 5, 6, 7, 8 |
| | | Spesifikasi <i>software</i> | Software pemrograman, basis data dan sistem operasi | Penentuan kriteria <i>software</i> yang akan digunakan untuk mengembangkan Model Tes <i>Essay</i> Berbantuan Komputer | 9, 10, 11 |
| 3 | Pemilihan rancangan <i>software</i> | Algoritma | Algoritma yang digunakan | Perancangan algoritma dalam program Tes <i>Essay</i> | 12 |
| | | <i>database</i> | <i>Database</i> yang digunakan | Pemilihan <i>database</i> untuk penyimpanan data tes <i>essay</i> | 13 |
| | | antarmuka | Desain tampilan | Penentuan <i>layout</i> tampilan program | 14 |
| 4 | Pemilihan <i>database</i> | Struktur data | Format penyimpanan data | Penentuan format data dalam <i>database</i> yang akan dipergunakan | 15, 16 |
| 5 | Pemilihan teknik pengenalan kata dan kalimat | Algoritma pengolah jawaban | Algoritma untuk memisahkan kalimat menjadi kata | Penentuan algoritma yang berfungsi menguraikan kalimat jawaban menjadi kata | 17, 18, 19 |
| 6 | Pemilihan teknik perbandingan jawaban dan kunci jawaban | Algoritma koreksi jawaban <i>essay</i> | Algoritma untuk menentukan nilai kesamaan | Penentuan algoritma dalam perbandingan jawaban dan kunci jawaban | 20, 21, 22, 23 |
| 7 | Pemilihan cara penskoran jawaban <i>essay</i> | Algoritma penskoran jawaban <i>essay</i> | Algoritma untuk menentukan skor jawaban <i>essay</i> | Penentuan metode dalam memberi skor jawaban | 24, 25, 26, 27 |
| 8 | Pemilihan cara menilai | Algoritma konversi skor tes | Konversi skor menjadi nilai angka | Penentuan metode konversi skor menjadi nilai angka | 28 |
| | | | Konversi skor menjadi nilai huruf | Penentuan metode konversi skor menjadi nilai huruf | 29 |
| | | | Estimasi kemampuan | Mengestimasi kemampuan siswa dari hasil akhir tes | 30 |

2. Instrumen Penelitian Pengujian Fungsional Produk

Instrumen Pengujian Produk digunakan untuk melakukan proses pengujian *software*. Instrumen ini berupa checklist aspek-aspek pengujian

software pada saat pengembangan produk dan saat penilaian oleh ahli *software*.

Tabel 7. merupakan Aspek-aspek Pengujian *Software*.

Tabel 7. Aspek-aspek Pengujian Produk

| No | Aspek | Sub Aspek | Indikator | Deskripsi | No Butir |
|----|--------------------------|-----------------------------------|---|---|------------|
| 1 | <i>Syntax error</i> | Sistem Operasi | Program dapat berjalan pada sistem operasi Windows | Program dapat berjalan pada sistem operasi Windows 7 | 1 |
| | | Mengenali user | Program mampu mengenali user | Program mampu mengenali user guru maupun siswa | 2, 3, 4 |
| | | Menutup program | User dapat menutup program | User dapat menutup program setelah selesai penggunaan | 5, 6 |
| 2 | <i>Run time error</i> | Akses program pada <i>library</i> | Program mampu merespon perintah user | Program dapat menampilkan halaman sesuai perintah user | 7, 8, 9 |
| 3 | <i>Logic error</i> | <i>Output</i> program | Program mampu menampilkan hasil akhir yang sesuai | Output program sesuai dengan rancangan yang telah dibuat | 10, 11, 12 |
| 4 | <i>White box testing</i> | <i>Data flow</i> | Program mampu membaca masukan memproses dan menghasilkan keluaran | Program membaca masukan berupa jawaban, memproses jawaban dan menghasilkan keluaran skor | 13, 14, 15 |
| | | <i>Logical decision</i> | Program mampu mengambil keputusan pada proses tes <i>essay</i> | Program mampu mengambil keputusan pemilihan butir soal, membatasi waktu tes dan menilai jawaban | 16, 17, 18 |
| | | Perulangan (<i>loop</i>) | Program mampu melakukan fungsinya secara berulang-ulang | Program bisa digunakan secara berulang-ulang | 19, 20, 21 |
| | | Validitas data | Program mampu menghasilkan data yang valid | Data yang dihasilkan program valid | 22, 23, 24 |

3. Instrumen Penelitian Angket Unjuk Kerja *Software*

Instrumen Unjuk Kerja *Software* digunakan untuk melakukan proses penilaian *software* oleh pengguna guru dan siswa. Instrument ini berupa angket penilaian *software*. Tabel 8. merupakan Aspek-aspek Unjuk Kerja *Software*.

Tabel 8. Aspek-aspek Unjuk Kerja *Software*

| No | Aspek | Sub Aspek | Indikator | Deskripsi | No Butir | Responden |
|----|----------------------------|---------------------|---|--|----------|---------------------------------|
| 1 | Cara penggunaan | Kelengkapan program | Kelengkapan fasilitas | Kelengkapan fasilitas pada program | 1 | Guru dan Siswa |
| | | | Kelengkapan menu | Kelengkapan menu pada program | 2 | |
| | | Kemudahan program | Kemudahan bahasa | Kemudahan bahasa pada program | 3 | |
| | | | Kemudahan navigasi | Kemudahan navigasi pada program | 4, 5 | |
| 2 | Tampilan <i>software</i> | Penggunaan teks | Ukuran teks | Ukuran teks pada program mudah dibaca | 6, 7 | Guru dan Siswa |
| | | | Warna teks | Warna teks pada program menarik | 8, 9 | |
| | | Penggunaan gambar | Warna background | Warna background pada program kontras dengan warna teks | 10, 11 | |
| | | | Warna layout tampilan | Warna background pada program menarik | 12 | |
| 3 | Materi tes | Pokok bahasan | Sesuai dengan pokok bahasan | Materi tes <i>essay</i> sesuai dengan pokok bahasan | 13, 14 | Guru dan Siswa |
| | | Kompetensi dasar | Sesuai kompetensi dasar | Materi tes <i>essay</i> sesuai dengan kompetensi dasar | 15 | |
| | | Ranah kognitif | Sesuai dengan ranah kognitif | Materi tes <i>essay</i> dapat mengukur aspek kognitif | 16 | |
| 4 | Manfaat | Bagi guru | Membantu proses tes | Program membantu proses tes | 17, 20 | Guru dan Siswa |
| | | | Memudahkan guru dalam menilai | Program mempermudah proses penilaian | 19 | |
| | | Bagi siswa | Memudahkan siswa latihan | | 18, 21 | |
| 5 | Kelebihan tes <i>essay</i> | Aspek yang diukur | Mengukur aspek kognitif tinggi | Tes <i>essay</i> dapat mengukur aspek kognitif tinggi | 22 | Guru dan Siswa (22, 23, 24, 25) |
| | | Aspek bahasa | Mengembangkan kemampuan berbahasa | Tes <i>essay</i> dapat mengembangkan kemampuan berbahasa | 23 | |
| | | Kemampuan berpikir | Melatih kemampuan berpikir | Tes <i>essay</i> melatih kemampuan berpikir | 24, 26 | |
| | | Pemecahan masalah | Mengembangkan kemampuan pemecahan masalah | Tes <i>essay</i> dapat mengembangkan kemampuan pemecahan masalah | 25 | |
| | | Penyusunan soal | Kemudahan penyusunan soal | Tes <i>essay</i> mempermudah guru dalam pembuatan soal | 27 | |

4. Dokumentasi

Dokumentasi yaitu berupa silabus mata pelajaran elektronika dasar. Materi pelajaran yang diajarkan pada siswa bersumber dari silabus. Oleh karena itu silabus menjadi dasar dalam penyusunan butir tes yang akan digunakan dalam proses pengujian *software*.

Tabel 9. Aspek-aspek Tabel *Checklist* Dokumen

| No | Aspek | Sub Aspek | Indikator | Deskripsi |
|----|---------|--------------------|------------------|----------------------|
| 1 | Silabus | Standar Kompetensi | Kompetensi Dasar | Materi pembuatan tes |

5. Tes

Tes merupakan cara untuk mengukur kemampuan hasil belajar siswa. Data tes merupakan butir soal dan respons jawaban dari siswa. Proses pengujian menggunakan bantuan komputer sebagai media dalam mengerjakan tes. Data jawaban siswa kemudian dinilai secara langsung oleh komputer dan hasilnya dapat dilihat secara langsung.

Tabel 10. Kisi-Kisi Butir Bank Soal

| No | Standar Kompetensi | Kelas/Sem | Kompetensi Dasar | Materi | Tingkat kesulitan | Bobot | No Soal |
|-------|-------------------------------------|-----------|--|---------------|-------------------|-------|----------------|
| 1 | Mengenal komponen dasar elektronika | X/1 | Mengidentifikasi Komponen Pasif, aktif dan optik | Resistor | Mudah | 5 | 10 |
| | | | | | Sedang | 10 | 8 |
| | | | | | Sulit | 15 | 1, 2 |
| | | | | Kapasitor | Mudah | 5 | 9 |
| | | | | | Sedang | 10 | 7 |
| | | | | | Sulit | 15 | 20 |
| | | | | Induktor | Mudah | 5 | 25 |
| | | | | | Sedang | 10 | 22 |
| | | | | Transformator | Sedang | 10 | 3, 4, 5, 6 |
| | | | | Dioda | Mudah | 5 | 24 |
| | | | | | Sedang | 10 | 15, 16, 21, 23 |
| | | | | | Sulit | 15 | 11, 19 |
| | | | | Transistor | Sedang | 10 | 13, 14 |
| Sulit | 15 | 12 | | | | | |
| Optik | Mudah | 5 | 17, 18 | | | | |

G. Validitas dan Reliabilitas Instrumen

1. Validitas Instrumen

Validitas instrumen penelitian dilakukan dengan menunjukkan alat pengumpul data kepada expert judgement yaitu Bapak Dr. Istanto Wahyu Djatmiko, M.Pd dan Bapak Dr. Samsul Hadi, M.Pd, MT. Instrumen yang digunakan dalam penelitian ini berupa kuesioner dan soal *essay* yang diujikan kepada beliau. Hasil validitas ini merupakan alat pengumpul data untuk mengetahui unjuk kerja *software* tes *essay* berbantuan komputer dan soal *essay*. Hasil validasi dapat dilihat pada Tabel 11.

Tabel 11. Validasi Instrumen

| No | Expert Judgement | Validasi/Saran |
|----|----------------------------------|--|
| 1. | Dr. Istanto Wahyu Djatmiko, M.Pd | <ol style="list-style-type: none">1. Perbaiki kisi-kisi instrument, terdiri atas: aspek, sub-aspek, indikator, dan deskripsinya.2. Format angket dibuat lebih menarik (tidak dalam bentuk tabel).3. Sesuaikan kajian teori dengan sub-aspek yang ada dalam kisi-kisi instrumen.4. Kata obyek yang dinyatakan letakkan pada awal pernyataan. |
| 2. | Dr. Samsul Hadi, M.Pd, MT | <ol style="list-style-type: none">1. Cek redaksi dan layout2. Perbaiki petunjuk pengisian |

Saran pada Tabel 12. Kemudian digunakan untuk memperbaiki kuesioner agar dapat digunakan untuk penelitian. Hasil revisi berupa (1) perbaikan penulisan kata yang salah, (2) perbaikan kisi-kisi instrument, (3) merubah format kuesioner, (4) perbaikan petunjuk pengisian kuesioner.

2. Reliabilitas Instrumen

Penelitian ini menggunakan teknik pengujian reliabilitas dengan rumus Alpha Cronbach. Pengujian reliabilitas ini bertujuan untuk menguji instrument pengumpul data berupa kuesioner dan soal *essay*.

$$r = \left[\frac{k}{(k-1)} \right] \left[1 - \frac{\sum \sigma_b^2}{\sigma_t^2} \right]$$

Keterangan :

r = Realibilitas instrumen (*Cronbach alpha*)
 k = Banyaknya butir pertanyaan
 $\sum \sigma_b^2$ = Jumlah varians butir
 σ_t^2 = Varians total

Ujicoba kuesioner dan soal *essay* dilakukan sebanyak satu kali. Berdasarkan hasil perhitungan diketahui bahwa nilai reliabilitas kuesioner adalah 0,91 atau dapat dikatakan sangat reliabel dan reliabilitas soal *essay* adalah 0,84 atau dapat dikatakan sangat reliabel.

H. Teknik Analisis Data

Teknik analisis data yang digunakan merupakan teknik analisis deskriptif kuantitatif. Tahap pertama merupakan uji kelayakan *software* yang diperoleh dengan pengisian lembar penilaian oleh ahli *software* yaitu Bapak Dr. Samsul Hadi, M.Pd, MT dan Bapak Rustam Asnawi, M.T, Ph.D dalam bentuk *checklist* dan uraian saran. Kemudian data unjuk kerja *software* diperoleh dengan angket oleh guru dan siswa.

Tahap kedua, penelitian berfungsi untuk menguji penerapan *software* untuk melakukan Tes. Hasil tes kemudian dianalisis secara statistik deskriptif untuk mengetahui hasil pengujian tes menggunakan *software*. Data hasil penelitian dianalisis menggunakan perhitungan nilai mean, median, dan simpangan baku. Kategori data hasil penelitian diolah menggunakan rumus seperti Tabel 12 (Arikunto , 2004: 290).

Tabel 12. Kategori Data Hasil Penelitian

| Interval Skor | Kategori |
|--|-----------------|
| $M_i + 1,5 SD_i < X \leq M_i + 3 SD_i$ | Sangat Baik |
| $M_i < X \leq M_i + 1,5 SD_i$ | Baik |
| $M_i - 1,5 SD_i < X \leq M_i$ | Cukup Baik |
| $M_i - 3 SD_i < X \leq M_i - 1,5 SD_i$ | Kurang Baik |

Keterangan:

M_i : Rata-rata ideal

SD_i : Simpangan baku ideal

M_i : $\frac{1}{2} \times$ (jumlah skor maksimal ideal + jumlah skor minimal ideal)

SD_i : $\frac{1}{6} \times$ (jumlah skor maksimal ideal - jumlah skor minimal ideal)

Skor penilaian kelayakan pada Tabel 12 digunakan sebagai acuan untuk

menentukan unjuk kerja *software tes essay*.

BAB IV

HASIL PENELITIAN

A. Deskripsi Data Uji Coba

1. Pengembangan Perangkat Lunak

Perangkat lunak yang dapat membantu guru dalam melakukan tes hasil belajar sangat diperlukan untuk menurunkan beban pekerjaan. Dalam penelitian ini, perangkat lunak Tes *Essay* Berbantuan komputer diharapkan mampu memberikan kemudahan guru dalam proses pengujian dan koreksi jawaban soal *essay*. Pengembangan perangkat lunak dimulai dari *System/information engineering*, analisis, *design*, *code*, dan *testing*.

a. System/ Information Engineering

System/information engineering merupakan proses untuk mengumpulkan informasi yang berkaitan dengan pengembangan perangkat lunak. Informasi yang dikumpulkan merupakan bahan kajian dalam analisis dan design dalam pengembangan perangkat lunak. Informasi itu antara lain: perangkat keras, pengguna, database. Proses *System/information engineering* meliputi tahap Analisis dan tahap *Design*.

b. Analisis Sistem

Analisis sistem merupakan proses menyusun spesifikasi yang diperlukan oleh perangkat lunak, menentukan fungsi yang diperlukan, perilaku perangkat lunak, performa dan antarmuka

1) Analisis Kebutuhan Pemakai

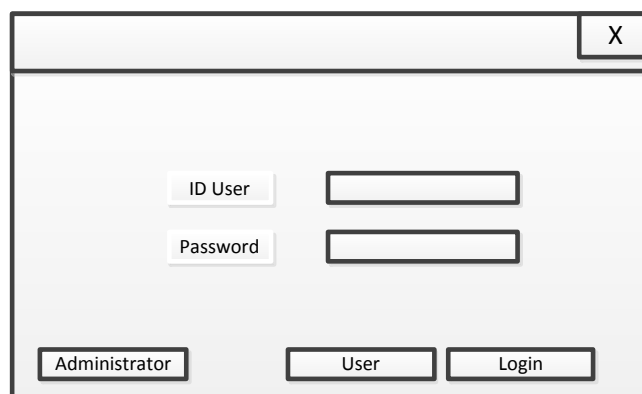
Hasil identifikasi dari tahap analisis kebutuhan pemakai adalah sebagai berikut: a) perangkat lunak dapat membantu proses pengujian kemampuan

siswa, b) perangkat lunak mudah dioperasikan oleh pengguna, c) perangkat lunak memiliki kemudahan untuk dikembangkan dan disempurnakan lebih lanjut, d) perangkat lunak mudah di akses, e) perangkat lunak memiliki standar kelayakan, f) perangkat lunak memiliki keamanan yang baik, g) perangkat lunak dapat diaplikasikan untuk berbagai mata pelajaran.

2) Analisis Kerja

Hasil identifikasi dari tahap analisis kerja adalah:

- a) Pada bagian awal program saat dijalankan maka akan menampilkan halaman login yang berisi: ID (User/Admin), password dan tombol Admin, Login serta tombol Close.



The image shows a wireframe of a login window. At the top right is a close button labeled 'X'. The main area contains two input fields: the first is labeled 'ID User' and the second is labeled 'Password'. At the bottom, there are three buttons: 'Administrator', 'User', and 'Login'.

Gambar 8. Desain Tampilan Halaman Login

- b) Tombol Admin digunakan oleh pengguna yang akan login sebagai admin, yaitu orang yang diberi wewenang mengatur keseluruhan perangkat lunak. Tombol Login digunakan oleh pengguna guru dan siswa. User Guru memiliki wewenang mengatur segala sesuatu tentang ujian, sedangkan user Siswa hanya memiliki wewenang mengerjakan soal saja. Tombol Close berfungsi untuk menutup halaman login.
- c) Jika tombol admin atau login ditekan maka perangkat lunak akan mengecek database user untuk membandingkan data ID dan data *password*.

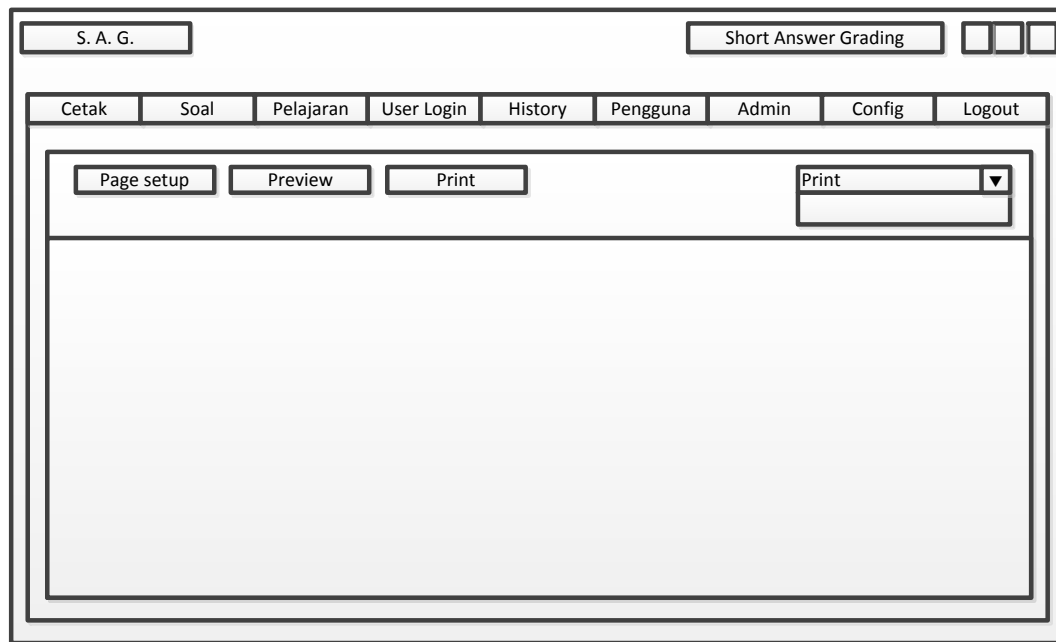
Perangkan lunak akan menampilkan ID salah jika ID belum terdaftar, bila ID terdaftar tetapi *password* salah maka akan ditampilkan pesan *password* salah dan harus mengulangi proses login. Halaman utama akan ditampilkan apabila ID dan *password* tepat sesuai dengan database user.

d) Jika pengguna merupakan administrator maka akan tampil halaman utama berupa 8 *tabpage* yaitu: cetak, soal, pelajaran, userlog, history, pengguna, admin, konfigurasi, tombol log out dan close. Area dibawah *tabpage* merupakan tempat menampilkan data dari menu *tabpage* yang dipilih.

e) Jika pengguna merupakan guru maka menu yang ditampilkan sama dengan menu pada pengguna admin, tetapi dengan pengurangan fitur yaitu menu *tabpage* "admin" tidak aktif dan tidak dapat digunakan.

f) Jika pengguna merupakan siswa maka 8 menu *tabpage* akan disembunyikan dan diganti dengan teks "SAS Short Answer Scoring". Pengguna siswa memulai mengerjakan tes dengan menekan tombol "Mulai Tes" dan perangkat lunak akan menampilkan *window* "Tes" yang terdiri dari *textbox* no soal, waktu tes, soal, jawaban, tombol "*prev*" untuk memilih soal sebelumnya, "*Next*" untuk memilih soal selanjutnya. Waktu tes akan menghitung waktu yang tersisa untuk mengerjakan, bila waktu telah habis maka otomatis *window* "Tes" akan ditutup. Bila *window* "Tes" ditutup dengan tombol submit atau karena waktu habis maka perangkat lunak akan melakukan koreksi jawaban dari pertanyaan-pertanyaan yang diberikan. Saat proses koreksi selesai maka perangkat akan menampilkan nilai dari jawaban tes.

g) *Tabpage* "Cetak" merupakan fasilitas untuk mencetak data yang ada pada perangkat lunak berupa data soal, hasil tes, data user siswa, data user guru. *Tabpage* "Cetak" hanya ditampilkan pada pengguna Admin dan Guru.



Gambar 9. Desain Tampilan Halaman Cetak

h) *Tabpage* "Soal" merupakan fasilitas pengelolaan data soal berupa tiga tombol "Tambah", "Ubah", "Hapus", kolom pemilihan mata pelajaran dan tabel untuk menampilkan data soal.

Pertama user harus memilih mata pelajaran terlebih dahulu, bila pada mata pelajaran terpilih telah terdapat soal dalam database maka soal akan ditampilkan dalam tabel. Tombol "Tambah" digunakan untuk menambah soal baru, *window* baru akan muncul untuk mengisi data soal baru berupa: *textbox* ID, Mapel, Bbt, Ik, DB, Butir Soal, Kunci jawaban, tombol Add dan Cancel. ID menunjukkan nomor soal, Mapel, menunjukkan golongan mata pelajaran soal, Bbt merupakan bobot nilai maksimal jawaban, Ik merupakan indeks kesukaran soal, DB merupakan daya beda soal, Butir Soal merupakan tempat menulis

butir soal, Kunci Jawaban merupakan kata kunci jawaban yang tepat, Add merupakan tombol untuk menambahkan soal kedalam database, tombol Add akan berfungsi bila seluruh kolom telah terisi, Cancel digunakan untuk membatalkan proses menambah soal baru. Tombol "Ubah" digunakan untuk mengubah soal, pertama soal yang akan diubah dipilih dengan pada tabel data soal kemudian tekan tombol ubah, *window* ubah soal memiliki desain sama dengan tambah soal. Tombol "Hapus" berfungsi untuk menghapus soal yang telah dipilih pada tabel data soal. *TabPage* "Soal" ditampilkan hanya pada pengguna Admin dan Guru.

The screenshot shows a web application window titled "S. A. G." with a sub-header "Short Answer Grading". Below the header is a navigation menu with tabs: "Cetak", "Soal" (selected), "Pelajaran", "User Login", "History", "Pengguna", "Admin", "Config", and "Logout".

Below the navigation menu, there are three buttons: "Tambah", "Ubah", and "Hapus". To the right of these buttons is a dropdown menu labeled "Mata Pelajaran".

Below the buttons and dropdown menu is a table with the following structure:

| ID | Bbt | IK | DB | Butir soal | kunci |
|----|-----|----|----|------------|-------|
| | | | | | |
| | | | | | |
| | | | | | |

Gambar 10. Desain Tampilan Halaman Soal

| | | | | | | | | | | | |
|---------|----------------------|-------|----------------------|-----|----------------------|----|----------------------|----|----------------------|----------|-------|
| ID | <input type="text"/> | Mapel | <input type="text"/> | Bbt | <input type="text"/> | IK | <input type="text"/> | DB | <input type="text"/> | Masukkan | Tutup |
| Soal | | | | | | | | | | | |
| Jawaban | | | | | | | | | | | |

Gambar 11. Desain Tampilan Tambah/Ubah Soal

- i) *Tabpage* "Pelajaran" merupakan fasilitas untuk melakukan penambahan, perubahan dan penghapusan mata pelajaran.

| | | | | | | | | |
|--------------|----------------------|----------------------|----------------|--------------------------|--------------------------|--------------------------|--------|--------|
| S. A. G. | | Short Answer Grading | | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | | |
| Cetak | Soal | Pelajaran | User Login | History | Pengguna | Admin | Config | Logout |
| Tambah | | Ubah | | Hapus | | | | |
| ID Pelajaran | <input type="text"/> | ID | Mata Pelajaran | | | | | |
| Nama Mapel | <input type="text"/> | | | | | | | |
| | Add | | | | | | | |

Gambar 12. Desain Tampilan Halaman Pelajaran

Pada sisi kiri terdapat tombol "Tambah", "Ubah", "Hapus", kolom "ID Pelajaran", "Nama Mata Pelajaran", Tombol "Add", "Cancel", dan sisi kanan merupakan tabel untuk menampilkan daftar mata pelajaran yang telah tersimpan dalam database. User Admin dan Guru dapat melakukan penambahan pelajaran dengan menekan tombol "Tambah" dilanjutkan

mengisi nama mata pelajaran dan menekan tombol "Add". Bila pengguna ingin mengubah nama mata pelajaran maka terlebih dahulu memilih mata pelajaran pada tabel dan menekan tombol "Ubah" dilanjutkan menekan tombol "Add". Bila ingin membatalkan proses maka pengguna dapat menekan tombol "Cancel". Daftar mata pelajaran dapat dihapus dengan memilih nama mata pelajaran kemudian menekan tombol "Hapus". Pengguna yang memiliki wewenang menggunakan *TabPage* "Pelajaran" adalah Admin dan Guru.

j) *TabPage* "User Log" merupakan fasilitas untuk menampilkan hasil tes berdasarkan kategori Mata Pelajaran, terdiri dari tombol "Hapus", "Reset", pilihan Mata Pelajaran dan tabel hasil tes. Hasil tes yang akan ditampilkan dipilih dahulu melalui berdasarkan mata pelajaran pada combobox yang tersedia. Hasil tes ditampilkan dalam bentuk tabel dengan kolom ID Siswa, Soal, Jawaban dan nilai.

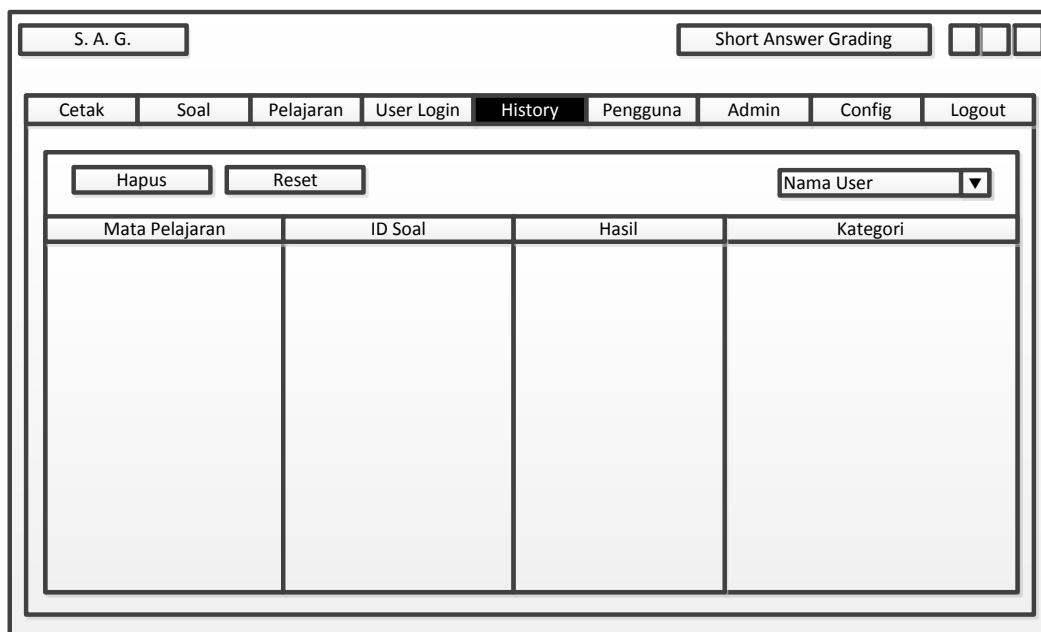
The screenshot shows a web application interface for 'Short Answer Grading'. At the top, there is a user identifier 'S. A. G.' and the page title 'Short Answer Grading'. Below this is a navigation menu with tabs: 'Cetak', 'Soal', 'Pelajaran', 'User Login' (which is currently selected), 'History', 'Pengguna', 'Admin', 'Config', and 'Logout'. Under the 'User Login' tab, there are two buttons: 'Hapus' and 'Reset', and a dropdown menu labeled 'Mata Pelajaran'. Below these controls is a table with the following structure:

| ID | Soal | Jawaban | Nilai |
|----|------|---------|-------|
| | | | |

Gambar 13. Desain Tampilan Halaman *User Log*

Data tes dapat dihapus dengan memilih data yang akan dihapus dan menekan tombol hapus. Tombol "Reset" berfungsi untuk menghapus seluruh data pada mata pelajaran yang dipilih. Pengguna yang memiliki wewenang menggunakan *TabPage* "User Log" merupakan Admin dan Guru.

k) *TabPage* "History" merupakan fasilitas untuk menampilkan hasil tes berdasarkan kategori Siswa, terdiri dari tombol "Reset", "Hapus", pilihan "nama User" dan tabel hasil.



Gambar 14. Desain Tampilan Halaman *History*

Tabel hasil tes memuat "Mata Pelajaran", "ID Soal", "Hasil" dan "Kategori". Data tes dapat dihapus maupun direset dengan tombol yang tersedia. Pengguna yang memiliki wewenang *tabpanel* "History" adalah Admin dan Guru.

l) *TabPage* "Pengguna" merupakan fasilitas untuk mengatur data pengguna siswa. Pada *tabpanel* pengguna disediakan menu untuk menambah user siswa

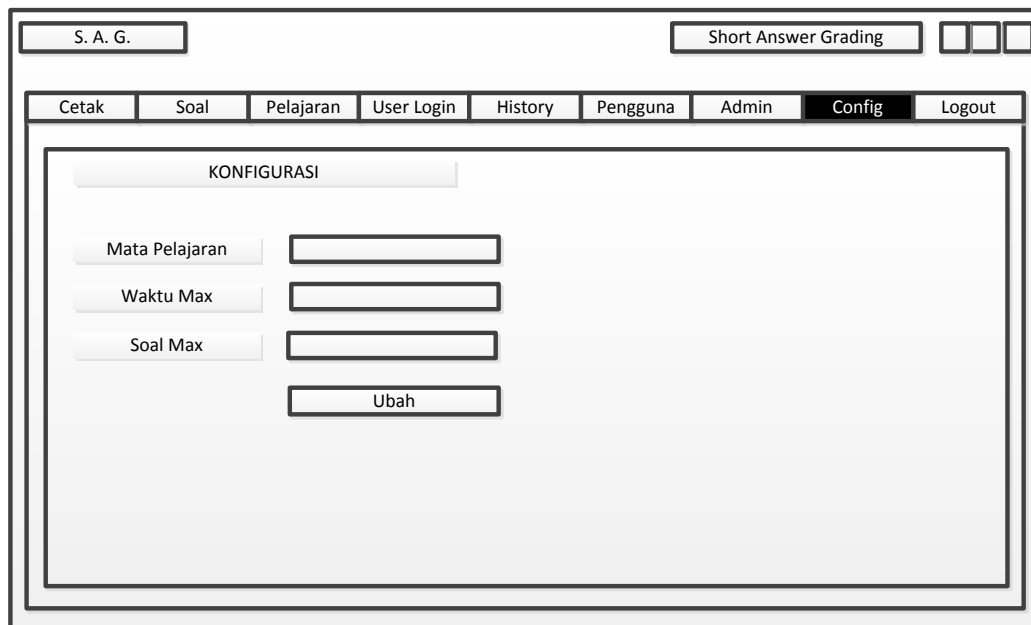
mengedit *password* maupun menghapus data user siswa. Pengguna yang memiliki wewenang *tabpage* "Pengguna" adalah Admin dan Guru.

The screenshot shows a web application interface for user management. At the top, there is a user profile section with the name "S. A. G." and a "Short Answer Grading" button. Below this is a navigation menu with tabs: "Cetak", "Soal", "Pelajaran", "User Login", "History", "Pengguna" (which is the active tab), "Admin", "Config", and "Logout". Under the "Pengguna" tab, there are three buttons: "Tambah", "Ubah", and "Hapus". The main content area is split into two columns. The left column contains three input fields labeled "ID Pengguna", "Nama Pengguna", and "Password", with a "Masukkan" button below them. The right column contains a table with two columns, "ID" and "Nama", and a large empty space below the table.

Gambar 15. Desain Tampilan Halaman Pengguna

m) *Tabpage* "Admin" merupakan fasilitas untuk mengatur data pengguna Guru. Pada *tabpage* admin disediakan menu sama dengan menu *tabpage* "Pengguna". *Tabpage* "Admin" merupakan bagian yang hanya bisa diakses bila pengguna merupakan Admin.

n) *Tabpage* "Konfigurasi" merupakan fasilitas yang bisa diakses oleh user Admin dan Guru, berfungsi untuk melakukan pengaturan tes. Pengaturan tes antara lain mata pelajaran, jumlah soal dan waktu maksimal. Tombol simpan digunakan untuk menyimpan pengaturan tes.



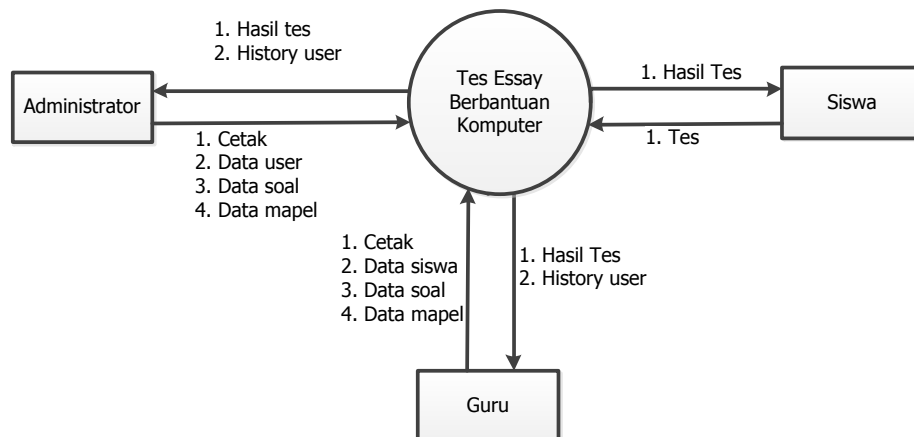
Gambar 16. Desain Tampilan Halaman Konfigurasi

- o) Tombol "Mulai Tes" merupakan tombol apabila pengguna akan memulai proses tes.
- p) Tombol "LogOut" merupakan tombol untuk keluar dari menu utama ke halaman login.

3) Analisis Data Flow Diagram

Data flow diagram meliputi tiga kategori pengguna perangkat lunak yaitu Administrator, Guru dan Siswa (Gambar 8).

Data flow diagram level 0 merupakan gambar yang menunjukkan interaksi pengguna dengan perangkat lunak. Pihak pertama Administrator memiliki wewenang mengatur semua data yang ada pada perangkat lunak data membaca semua data. Pengguna kedua Guru memiliki wewenang sama dengan administrator kecuali pada bagian pengaturan data guru. Pengguna ketiga Siswa hanya memiliki wewenang untuk mengerjakan soal dan melihat hasilnya saat akhir.



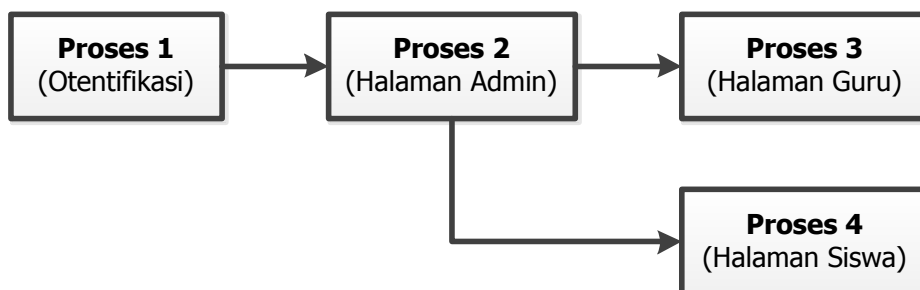
Gambar 17. *Data Flow Diagram* Level 0

c. Desain Sistem

Desain sistem merupakan tahap penerapan prosedur dalam pengembangan perangkat lunak dengan tujuan untuk merinci setiap peralatan, komponen dan proses sehingga mempermudah tahap implementasi.

1) *Data Flow Diagram* (DFD) Level 1

Desain DFD Level 1 merupakan penjelasan lebih lanjut dari DFD Level 0. DFD level 1 merupakan penjabaran secara spesifik proses program Tes Essay Berbantuan Komputer. Gambar 9 menunjukkan DFD level 1 yang menggambarkan proses program berjalan saat digunakan oleh user. Proses program dimulai dari user *login* hingga user *logout*, yang merupakan proses autentifikasi user sehingga dapat menentukan hak yang dimiliki oleh user.



Gambar 18. *Data Flow Diagram* Level 1

a) Proses 1

Merupakan proses awal program dijalankan. Pada bagian ini merupakan proses otentifikasi sebagai berikut:

- (1) Proses login, yaitu kondisi awal program. Proses yang berjalan adalah prosedur mengenali identitas user. Hasil identifikasi menentukan proses selanjutnya.
- (2) Hasil identifikasi akan diketahui user memiliki ID dan *password*.
- (3) Program tidak akan terbuka apabila ID dan *password* tidak cocok.
- (4) Menu program yang ditampilkan setelah login bergantung pada ID dan *password* yang dimasukkan.

b) Proses 2

Merupakan proses pembacaan login pengguna untuk menentukan halaman dan menu program yang akan terbuka selanjutnya.

- (1) Jika pengguna yang login merupakan administrator, maka program akan memberi kewenangan hak akses lengkap (semua menu terbuka).
- (2) Pengguna administrator dapat membuat, mengubah dan menghapus pengguna guru maupun pengguna siswa.
- (3) Pengguna administrator dapat membuat, mengubah, menghapus dan mengatur mata pelajaran serta konfigurasi soal.

c) Proses 3

Seperti pada proses 2, jika proses pembacaan login pengguna telah terbaca maka akan terbuka menu program selanjutnya

- (1) Jika pengguna yang login adalah guru, maka program akan diberi kewenangan hak akses terbatas (tidak semua menu dibuka)

- (2) Pengguna guru tidak dapat membuat, mengatur dan menghapus pengguna untuk sesama guru.
- (3) Pengguna guru memiliki wewenang membuat, mengubah dan menghapus pengguna siswa.
- (4) Pengguna guru memiliki wewenang untuk membuat, mengubah, menghapus dan mencetak mata pelajaran.
- (5) Pengguna guru memiliki wewenang untuk melihat hasil tes untuk semua peserta tes atau setiap individu peserta tes.
- (6) Pengguna guru memiliki kewenangan untuk proses konfigurasi soal yang akan diujikan.

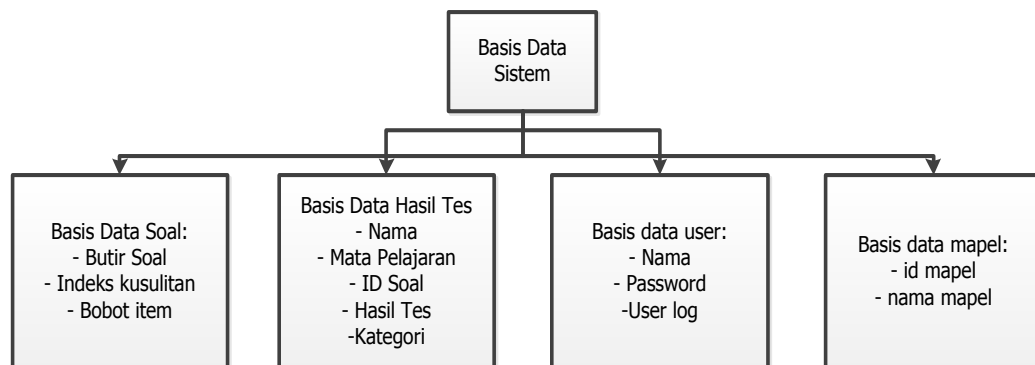
d) Proses 4

Merupakan proses login untuk pengguna siswa dalam proses tes.

- (1) Jika pengguna yang login adalah siswa, maka hak akses akan terbatas.
- (2) Pengguna siswa hanya dapat menjalankan program untuk melakukan tes sesuai konfigurasi yang diatur guru.
- (3) Pengguna siswa memiliki kewenangan untuk menjawab soal-soal yang diujikan.
- (4) Pada saat menjawab soal , siswa diberi fasilitas "*next*", "*prev*", "*submit*" dan "*cancel*". Tombol *next* digunakan untuk membuka soal selanjutnya, tombol *prev* digunakan untuk membuka soal sebelumnya, tombol *submit* digunakan untuk mengakhiri pengerjaan soal dan tombol *cancel* untuk membatalkan ujian.
- (5) Jika proses menjawab soal telah selesai maka program akan melakukan koreksi jawaban dan menampilkan nilai.

2) Perancangan Basis Data

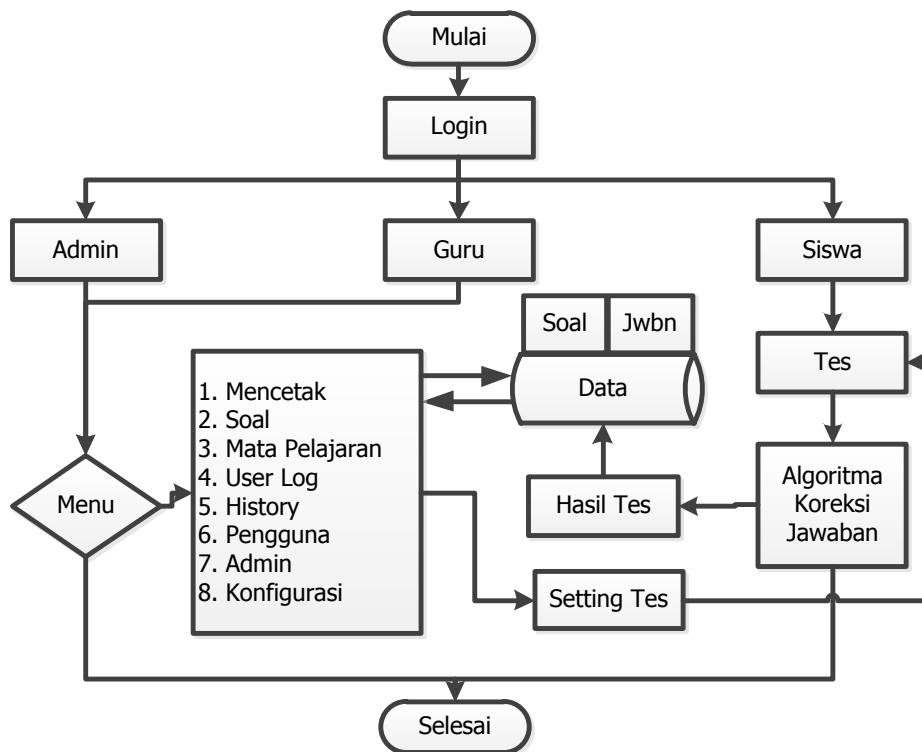
Basis data diperlukan dalam program untuk menyimpan hasil proses program. Rancangan basis data merupakan proses untuk menentukan jenis data yang digunakan sesuai dengan analisis yang diperlukan. Rancangan basis data dalam penelitian ini meliputi (a) data mata pelajaran, (b) data pengguna, (c) data soal, (d) data hasil tes. Data mata pelajaran dan data pengguna disimpan dalam bentuk file db_user dan db_mapel, sementara data soal dan data hasil tes disimpan dalam bentuk file biner dengan ASCII code.



Gambar 19. File Basis Data

3) Perancangan *Flow Chart*

Flow chart merupakan diagram atau gambar yang disusun dengan simbol-simbol baku untuk menggambarkan urutan proses. *Flow chart* digunakan sebagai acuan dalam pembuatan program. *Flow chart* secara garis besar merupakan gambaran program Tes Essay Berbantuan Komputer dalam proses kerja logika.



Gambar 20. Flow Chart Program Tes Essay Berbantuan Komputer

Flow chart dibuat berdasarkan pada analisis kebutuhan sistem yang telah diuraikan diatas, disertai dengan disain aliran data dalam program. *Flow chart* menggambarkan proses kerja program mulai dari saat program dibuka (*loading program*), program bekerja (*running program*) dan program selesai (*logout*). Gambar 11, merupakan diagram secara garis besar program tes essay berbantuan komputer.

4) Rancangan Algoritma Penilaian

Algoritma penilaian merupakan mekanisme program untuk memberikan nilai jawaban pertanyaan siswa berdasarkan algoritma yang telah diprogramkan. Jawaban pertanyaan user siswa akan dibandingkan dengan kunci jawaban dari soal yang bersangkutan. Ketepatan jawaban siswa kemudian diskala dari skala 1%-100%, selanjutnya dikonversi menjadi nilai oleh

program. Algoritma penskoran yang dipakai dalam program adalah (a) *Damerau-Levenshtein Algorithm*, (b) *N-Gram Similarity Algorithm*, (c) *Jaccard similarity* dan (d) *Extracting DIS tributionally similar word using CO-occurrences (DISCO)*.

Hasil penilaian jawaban soal tes oleh siswa berupa nilai persentase dari 0% hingga 100%. Nilai 0% berarti jawaban siswa salah dan 100% berarti jawaban siswa benar. Nilai persentase ini dihitung menjadi skor (nilai) berdasarkan informasi skor maksimum tiap soal. Skor seluruh tes selanjutnya ditambahkan kemudian dibagi dengan jumlah soal dalam tes untuk menghitung nilai akhir.

5) Desain Tampilan Program

Desain tampilan merupakan gambaran perangkat lunak yang menjadi dasar dalam proses implementasi. Desain tampilan berisi rancangan halaman yang akan ditampilkan sesuai dengan fasilitas menu perangkat lunak.

d. Kode

Berdasarkan rancangan tampilan yang telah diuraikan, selanjutnya dibuat halaman tampilan yang sesungguhnya. Kode-kode program selanjutnya diimplementasikan pada bagian ini menggunakan bahasa pemrograman, dan perintah program untuk akses basis data. Bahasa yang digunakan untuk menulis program adalah Microsoft Visual C# Express. Program basis data digunakan program MySQL dari produk microsoft.

Implementasi program dibuat berdasarkan rancangan diagram aliran data (*flow chart*) dan rancangan tampilan program. Terdapat dua hal dalam

implementasi yaitu: (1) tahap penulisan kode program dan (2) tampilan hasil program.

1) Penulisan Kode Program

Kode program merupakan tata cara penulisan menggunakan bahasa tertentu (bahasa komputer) dan disusun sedemikian rupa sehingga dapat dimengerti dan dilaksanakan oleh komputer. Berikut ini disampaikan kode program konseptual yang menggambarkan proses mulai dari awal hingga selesai.

(a) Proses Login

Berdasarkan analisis kebutuhan program tes essay berbantuan komputer mengklasifikasikan user menjadi administrator, guru dan siswa. Klasifikasi user menentukan proses program selanjutnya. Untuk mendeteksi user digunakan pembeda yaitu nama dan password. Kode proses login adalah sebagai berikut:

```
private void mButtonLogin_Click(object sender, EventArgs e)
{
    Cmd = new OleDbCommand(StrCmd, MyConn);
    MyConn.Open();
    DataReader = Cmd.ExecuteReader();
    if (DataReader.HasRows)
    {
        while (DataReader.Read())
        {
            Username = DataReader.GetString(0).ToLower();
            Password = DataReader.GetString(1);
            Status = DataReader.GetString(2);
            if ((Username == mTextBoxUser.Text.ToLower()) && (Password ==
mTextBoxPass.Text))
            {
                if (Status == "1") { StatusGuru = true; }
                if (Status == "2") { StatusSiswa = true; }
                mLogUserName = Username;
            }
        }
    }
    MyConn.Close();
    if ((StatusGuru == true) || (StatusSiswa == true)){DialogResult = DialogResult.OK;}
```

```

}
private void mButtonAdmin_Click(object sender, EventArgs e)
{
    Cmd = new OleDbCommand(StrCmd, MyConn);
    MyConn.Open();
    DataReader = Cmd.ExecuteReader();
    if (DataReader.HasRows)
    {
        while (DataReader.Read())
        {
            Username = DataReader.GetString(0).ToLower();
            Password = DataReader.GetString(1);
            if ((Username == mTextBoxUser.Text.ToLower()) && (Password ==
mTextBoxPass.Text))
            {StatusAdmin = true; mLogUserName = Username;
            }
        }
    }
    MyConn.Close();
    if (StatusAdmin == true){DialogResult = DialogResult.OK;}
}
}

```

(b) Halaman Utama

Berdasarkan desain, halaman utama dibagi menjadi tiga yaitu: administrator, guru dan siswa. Tampilan halaman administrator dan guru hampir sama, perbedaannya untuk administrator semua menu aktif sedangkan guru menu admin tidak aktif. Tampilan untuk siswa hanya berupa halaman untuk tes saja. Berikut ini kode proses untuk klasifikasi ketiga user tersebut:

```

private void Login()
{
    this.Hide();
    if (LoginForm.ShowDialog() == DialogResult.OK)
    {
        this.Show();
        mTabControl1.SelectedIndex = 0;
        LogUserName = LoginForm.LogUserName;
        mTextBoxUserId.Text = LogUserName;
        if (LoginForm.StatusGuru == true)
        {
            pictureBoxWelcome.Visible = false;
            mTabControl1.Visible = true;
            MessageBox.Show("Login As Teacher");
        }
        else if (LoginForm.StatusSiswa == true)
        {

```

```

        pictureBoxWelcome.Visible = true;
        mTabControl1.Visible = false;
        MessageBox.Show("Login As Student");
    }
    else if (LoginForm.StatusAdmin == true)
    {
        pictureBoxWelcome.Visible = false;
        mTabControl1.Visible = true;
        MessageBox.Show("Login As Admin");
    }
}
else { this.Close(); }
LoadDataSoal();
LoadExistDataTes();
}

```

(c) Halaman Soal

Halaman soal hanya bisa dibuka oleh administrator dan guru. Halaman soal menampilkan dua hal pokok yaitu:

- (1) Menampilkan soal berdasarkan mata pelajaran yang dipilih

```

private void LoadToListviewSoal()
{
    listViewSoal.Items.Clear();
    string[] DataSoal = new string[8];
    string Words;
    ListViewItem DataSoalToWrite;
    int hitung = 0;
    foreach (string Baris in rTextBoxSoal.Lines) { hitung++; }
    for (int i = 0; i < hitung - 1; i++)
    {
        string[] Lines;
        Words = rTextBoxSoal.Lines[i];
        Lines = Words.Split('~');
        DataSoal[1] = Lines[0];
        DataSoal[2] = Lines[1];
        DataSoal[3] = Lines[2];
        DataSoal[4] = Lines[3];
        DataSoal[5] = Lines[4];
        DataSoal[6] = Lines[5];
        DataSoalToWrite = new ListViewItem(DataSoal);
        listViewSoal.Items.Add(DataSoalToWrite);
    }
}

```

- (2) Menampilkan butir soal berdasarkan ID butir yang dipilih

```

private void Tambah_Soal_Load(object sender, EventArgs e)
{
    if (MainForm.EditSoal == true)
    {

```

```

        mTextBoxIDSoal.Text = MainForm.mIDSoal;
        mTextBoxMapel.Text = MainForm.mMapelSoal;
        mTextBoxBbt.Text = MainForm.mBbtSoal;
        mTextBoxIK.Text = MainForm.mIKSoal;
        mTextBoxDB.Text = MainForm.mDBSoal;
        rTBoxButirSoal.Text = MainForm.mButirSoal.Replace(" .~. ", "\r").Replace(" .!. ",
"\n").Replace(" .@. ", "\t");
        rTBoxKunciJawaban.Text = MainForm.mKunciJawaban.Replace(" .~. ",
"\r").Replace(" .!. ", "\n").Replace(" .@. ", "\t");
    }
    else
    {
        mTextBoxIDSoal.Clear();
        mTextBoxMapel.Clear();
        mTextBoxBbt.Clear();
        mTextBoxIK.Clear();
        mTextBoxDB.Clear();
        rTBoxButirSoal.Clear();
        rTBoxKunciJawaban.Clear();
        rTBoxButirSoal.Text = "";
        rTBoxKunciJawaban.Text = "";
        mTextBoxMapel.Text = MainForm.nMapelSoal;
    }
    mTextBoxIDSoal.Focus();
}

```

Halaman soal juga terdapat fungsi untuk menambah, mengubah dan menghapus soal. Untuk menghapus soal tampilan sama dengan saat mata pelajaran dipilih pada butir (1) diatas. Untuk menambah dan mengubah soal, tampilan sama dengan tampilan butir (2) diatas

(d) Halaman Pelajaran

Halaman pelajaran digunakan oleh administrator dan guru untuk memasukkan data mata pelajaran yang akan dibuat data soal. Berdasarkan desain, halaman pelajaran terdiri dari ID pelajaran dan nama pelajaran.

```

private void MoveUserToRTBoxMapel()
{
    listViewMapel.Items.Clear();
    string[] DataMapel = new string[2];
    string Words;
    ListViewItem DataMapelToWrite;
    int hitung = 0;
    foreach (string Baris in rTextBoxMapel.Lines) { hitung++; }
    for (int i = 0; i < hitung - 1; i++)
    {
        string[] Lines;

```

```

        Words = rTextBoxMapel.Lines[i];
        Lines = Words.Split('~');
        DataMapel[0] = Lines[0];
        DataMapel[1] = Lines[1];
        DataMapelToWrite = new ListViewItem(DataMapel);
        listViewMapel.Items.Add(DataMapelToWrite);
    }
}

```

Pada tampilan halaman pelajaran administrator dan guru dapat menambah, mengubah dan menghapus data mata pelajaran.

(e) Halaman *User Log*

Halaman user log berfungsi untuk menampilkan data hasil tes berdasarkan mata pelajaran untuk semua user(siswa). Data yang ditampilkan berupa ID siswa, butir soal, jawaban siswa dan nilai.

```

private void LoadToListviewUserLog()
{
    listViewUserLog.Items.Clear();
    string[] DataHasilTes = new string[17];
    string Words;
    ListViewItem DataHasilTesToWrite;
    int hitung = 0;
    foreach (string Baris in rTextBoxViewHasilTes.Lines) { hitung++; }
    for (int i = 0; i < hitung - 1; i++)
    {
        string[] Lines;
        Words = rTextBoxViewHasilTes.Lines[i];
        Lines = Words.Split('~');
        DataHasilTes[1] = Lines[14];
        DataHasilTes[2] = Lines[4];
        DataHasilTes[3] = Lines[8];
        DataHasilTes[4] = Lines[13];
        DataHasilTesToWrite = new ListViewItem(DataHasilTes);
        listViewUserLog.Items.Add(DataHasilTesToWrite);
    }
}

```

Administrator dan guru dapat menghapus data dari satu siswa atau semua data (reset).

(f) Halaman History

Halaman history berfungsi untuk menampilkan data hasil tes berdasarkan seorang user (siswa). Data yang ditampilkan berupa mata pelajaran, id soal, hasil, dan kategori.

```
private void LoadToListviewHistory()
{
    listViewHistory.Items.Clear();
    string[] DataHistory = new string[17];
    string Words;
    ListViewItem DataHistoryToWrite;
    int hitung = 0;
    foreach (string Baris in rTextBoxHistory.Lines) { hitung++; }
    for (int i = 0; i < hitung - 1; i++)
    {
        string[] Lines;
        Words = rTextBoxHistory.Lines[i];
        Lines = Words.Split('~');
        DataHistory[0] = Lines[6];
        DataHistory[1] = Lines[0];
        DataHistory[2] = Lines[13];
        DataHistory[3] = Lines[16];
        DataHistoryToWrite = new ListViewItem(DataHistory);
        listViewHistory.Items.Add(DataHistoryToWrite);
    }
}
```

Administrator dan guru dapat menghapus satu data atau menghapus semua data (reset).

(g) Halaman Pengguna/Admin

Halaman pengguna/admin berfungsi untuk menambah user, mengubah user, maupun menghapus user. Data user yaitu berupa username dan password.

```
private void MoveUserToRTBoxGuru()
{
    listViewGuru.Items.Clear();
    string[] DataGuru = new string[4];
    string Words;
    ListViewItem DataGuruToWrite;
    int hitung = 0;
    foreach (string Baris in rTextBoxGuru.Lines) { hitung++; }
    for (int i = 0; i < hitung - 1; i++)
    {
```



```

string[] Lines;
Words = rTextBoxGuru.Lines[i];
Lines = Words.Split('~');
DataGuru[0] = Lines[3];
DataGuru[1] = Lines[0];
DataGuruToWrite = new ListViewItem(DataGuru);
listViewGuru.Items.Add(DataGuruToWrite);
}
}

```

Pada halaman pengguna/admin mempunyai tampilan sama. User guru hanya dapat menambah, mengubah dan menghapus data user siswa. Sedangkan administrator dapat menambah, mengubah atau menghapus data user siswa maupun user guru.

(h) Halaman Konfigurasi

Halaman konfigurasi berfungsi untuk melakukan pengaturan soal saat akan diujikan, antara lain mata pelajaran, jumlah soal yang diujikan, waktu mengerjakan.

```

private void mButtonSaveSetting_Click(object sender, EventArgs e)
{
    string FileName, Mapel;
    int WaktuTes, JumlahSoal;
    if (!(string.IsNullOrEmpty(mComboBoxSettingMapel.Text)) &&
        !(string.IsNullOrEmpty(mTextBoxSettingWaktu.Text)) &&
        !(string.IsNullOrEmpty(mTextBoxSettingJumlahSoal.Text)))
    {
        FileName = "Data\\Setting.bin";
        Mapel = mComboBoxSettingMapel.Text;
        WaktuTes = Convert.ToInt32(mTextBoxSettingWaktu.Text);
        JumlahSoal = Convert.ToInt32(mTextBoxSettingJumlahSoal.Text);
        SaveDataSetting(FileName, Mapel, WaktuTes, JumlahSoal);
        MessageBox.Show("Setting baru disimpan");
    }
    else { MessageBox.Show("Setting belum lengkap"); }
    LoadDataSetting();
}

```

(i) Algoritma Koreksi Jawaban

Algoritma koreksi jawaban merupakan kode program yang menentukan skor jawaban siswa dalam tes. Siswa yang telah menyelesaikan

tes atau apabila waktu tes telah habis maka program akan mengakhiri proses tes dan menilai jawaban siswa.

```
private void Koreksi()
{
    int NilaiStage1, NilaiStage2, NilaiStage3, NilaiStage4, NilaiStage5, NilaiStage6,
    NilaiMax;
    int j = RtBoxTes.Lines.Length - 1;
    List<string> MyListTes = RtBoxTes.Lines.ToList();
    for (int i = 0; i < j; i++)
    {
        string Words = RtBoxTes.Lines[i];
        string[] Word = Words.Split('~');
        NilaiMax = Convert.ToInt32(Word[1]);
        RtBoxRaw.Clear();
        int Hitung = 0;
        string Input_Words = Word[8].ToLower();
        string[] Kalimat = Input_Words.Split(delimiterChars);
        foreach (string Kata in Kalimat) { RtBoxRaw.Text += Kata + "\n"; }
        List<string> myList = RtBoxRaw.Lines.ToList();
        foreach (string baris in RtBoxRaw.Lines)
        {
            if (baris == "") { myList.RemoveAt(Hitung); Hitung--; }
            Hitung++;
        }
        RtBoxRaw.Lines = myList.ToArray();
        myList = RtBoxRaw.Lines.Distinct().ToList();
        RtBoxRaw.Lines = myList.ToArray();
        RtBoxAnswerRaw.Clear();
        Hitung = 0;
        Input_Words = Word[5].ToLower();
        Kalimat = Input_Words.Split(delimiterChars);
        foreach (string Kata in Kalimat) { RtBoxAnswerRaw.Text += Kata + "\n"; }
        myList = RtBoxAnswerRaw.Lines.ToList();
        foreach (string baris in RtBoxAnswerRaw.Lines)
        {
            if (baris == "") { myList.RemoveAt(Hitung); Hitung--; }
            Hitung++;
        }
        RtBoxAnswerRaw.Lines = myList.ToArray();
        myList = RtBoxAnswerRaw.Lines.Distinct().ToList();
        RtBoxAnswerRaw.Lines = myList.ToArray();
        NilaiStage1 = LevenshteinRaw();
        StopWord();
        NilaiStage2 = LevenshteinStop();
        NgramRaw();
        NilaiStage4 = LevenshteinNgram();
        NgramStop();
        NilaiStage5 = LevenshteinNgram();
        NilaiAkhir = HitungNilaiAkhir(NilaiStage1, NilaiStage2, NilaiStage4, NilaiStage5,
        NilaiMax);
        mWaktuPengerjaan = (Convert.ToInt32(mTextBoxWaktuMax.Text)) * 60 -
        WaktuTesMax;
        string[] LevelPredikat = { "A", "A-", "B+", "B", "B-", "C+", "C", "C-", "D+", "D-" };
    }
}
```

```

string Predikat="";
int NilaiPredikat = (NilaiAkhir * 100) / NilaiMax;
if (NilaiPredikat <= 100 && NilaiPredikat >= 96) { Predikat = LevelPredikat[0]; }
else if (NilaiPredikat < 96 && NilaiPredikat >= 91) { Predikat = LevelPredikat[1]; }
else if (NilaiPredikat < 91 && NilaiPredikat >= 86) { Predikat = LevelPredikat[2]; }
else if (NilaiPredikat < 86 && NilaiPredikat >= 81) { Predikat = LevelPredikat[3]; }
else if (NilaiPredikat < 81 && NilaiPredikat >= 76) { Predikat = LevelPredikat[4]; }
else if (NilaiPredikat < 76 && NilaiPredikat >= 71) { Predikat = LevelPredikat[5]; }
else if (NilaiPredikat < 71 && NilaiPredikat >= 66) { Predikat = LevelPredikat[6]; }
else if (NilaiPredikat < 66 && NilaiPredikat >= 61) { Predikat = LevelPredikat[7]; }
else if (NilaiPredikat < 61 && NilaiPredikat >= 56) { Predikat = LevelPredikat[8]; }
else { Predikat = LevelPredikat[9]; }
MyListTes[j] += "~" + NilaiStage1.ToString() + "~" + NilaiStage2.ToString() +
"~" + NilaiStage4.ToString() + "~" + NilaiStage5.ToString() +
"~" + NilaiAkhir.ToString() +
"~" + MainForm.UserName.ToString() + "~" + mWaktuPengerjaan.ToString() +
"~" + Predikat;
mNilaiTes = mNilaiTes + NilaiAkhir;
NilaiTesTotal = NilaiTesTotal + NilaiMax;
}
RtBoxTes.Lines = MyListTes.ToArray();
mNilaiTes = (mNilaiTes * 100) / NilaiTesTotal;
}

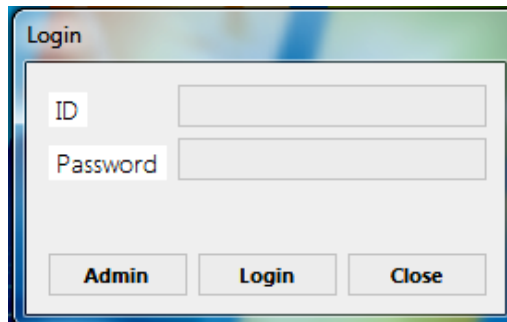
```

2) Tampilan Antar Muka Program

Berdasarkan kode yang telah dibuat, program kemudian dikompilasi menjadi satu kesatuan dan dijalankan serta diatur agar sesuai dengan yang diharapkan. Berikut ini tampilan program bila dijalankan/dieksekusi (running program).

a) Halaman Login

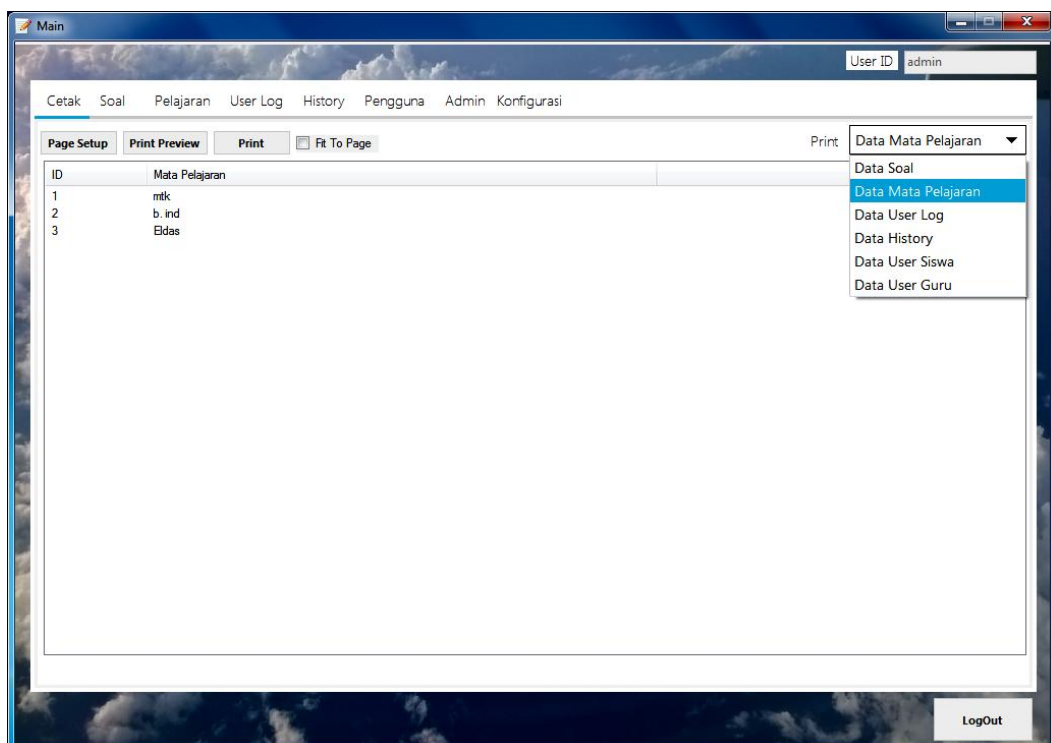
Halaman login menampilkan kotak dialog untuk memasukkan ID dan password. Terdapat tiga buah tombol yaitu tombol admin bila akan login sebagai administrator. Tombol login bila akan login sebagai guru atau siswa. Tombol cancel untuk membatalkan (menutup) program.



Gambar 21. Halaman Login

b) Halaman Cetak

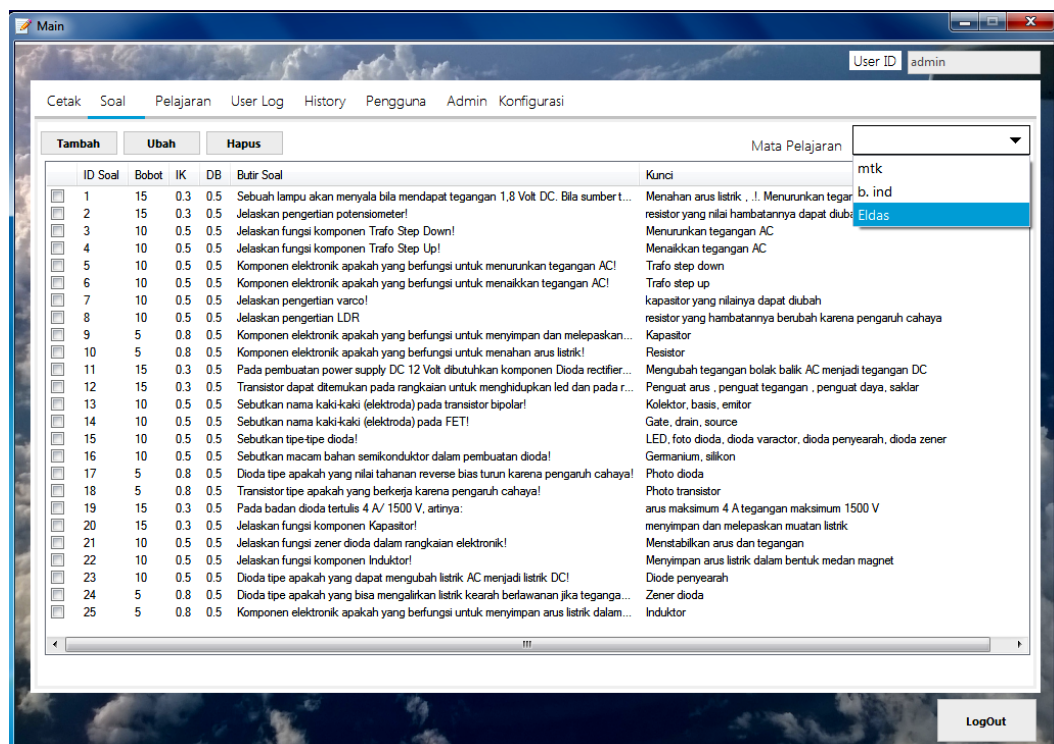
Halaman cetak menampilkan menu untuk mencetak data pada program, data itu antara lain: soal, mata pelajaran, user log, history, user siswa dan user guru. Tombol yang tersedia adalah page setup, print preview, print dan kotak pemilihan data yang akan dicetak.



Gambar 22. Halaman Cetak

c) Halaman Soal

Halaman soal merupakan tabel yang memuat butir soal. Data yang ada pada butir soal antara lain ID soal, bobot, indeks kesulitan, daya beda, butir soal dan kunci jawaban. Butir soal akan tampil dengan terlebih dahulu memilih mata pelajaran pada kotak pilihan.

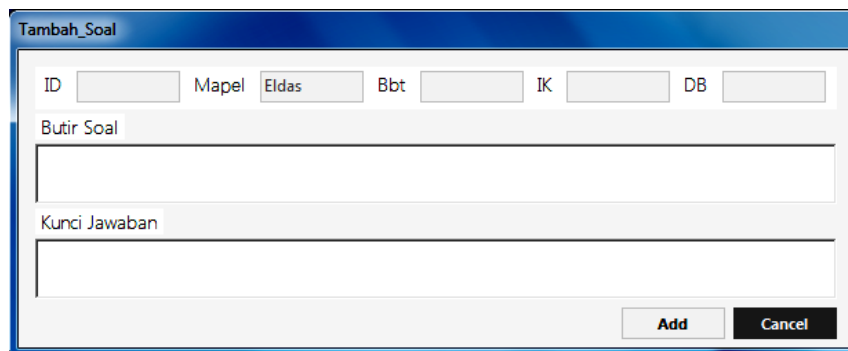


Gambar 23. Halaman Soal

Butir soal disusun dalam tabel yang terletak pada bagian tengah halaman. Halaman soal memiliki tombol untuk menambah, mengubah, dan menghapus soal. Apabila tombol tambah soal atau ubah soal ditekan, maka akan muncul halaman baru untuk menambah soal.

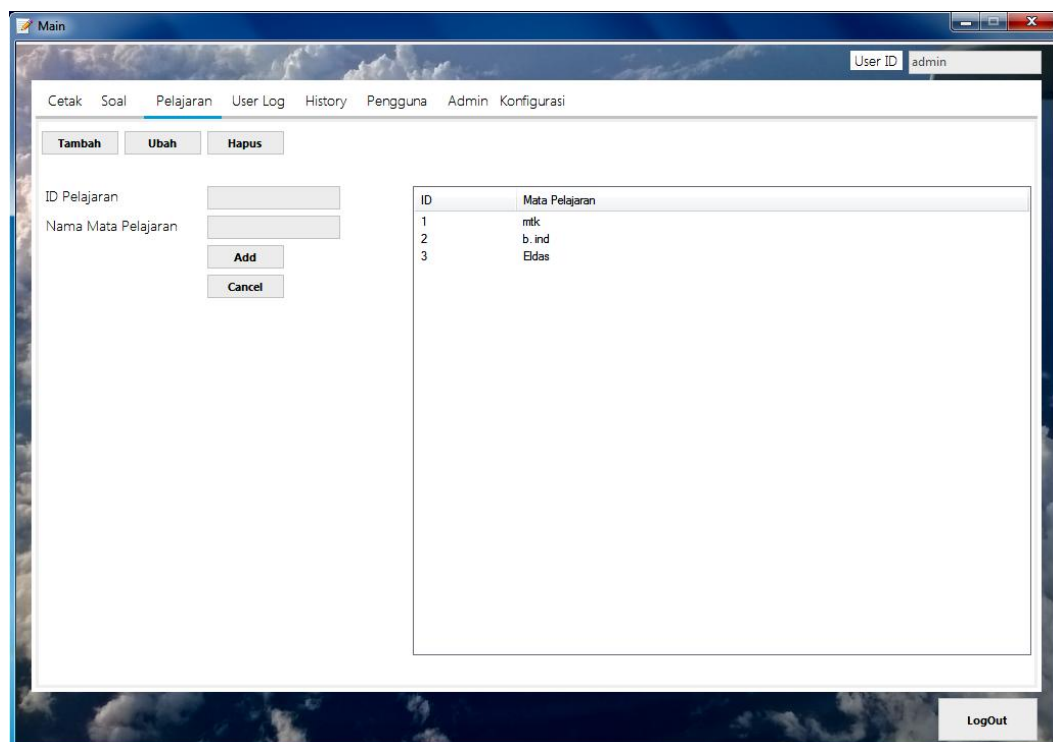
Gambar 33 pada bagian tengah merupakan tempat untuk menuliskan butir soal, sedangkan bagian bawah merupakan tempat untuk menuliskan kunci jawaban. Identitas soal yang harus dilengkapi antara lain ID soal, mata pelajaran,

bobot, indeks kesukaran dan daya beda. Tombol add digunakan untuk memasukkan soal kedalam database. Tombol cancel digunakan untuk membatalkan tambah atau ubah soal.



Gambar 24. Halaman Tambah dan Ubah Soal

d) Halaman Mata Pelajaran



| ID | Mata Pelajaran |
|----|----------------|
| 1 | mtk |
| 2 | b. ind |
| 3 | Eldas |

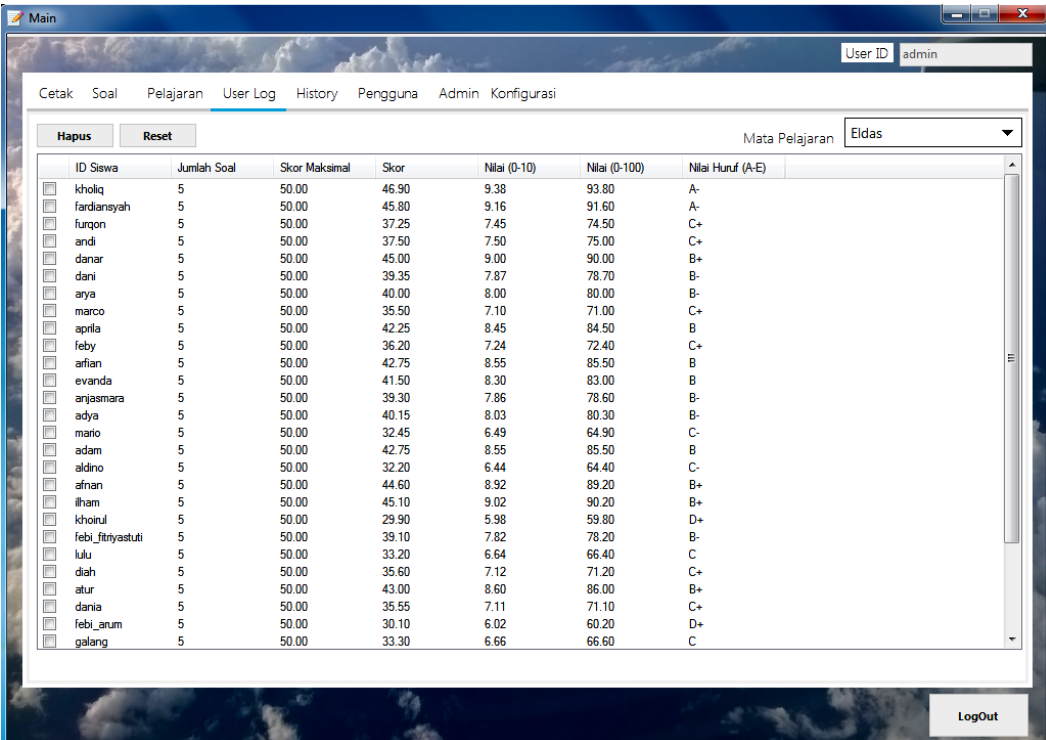
Gambar 25. Halaman Mata Pelajaran

Halaman mata pelajaran berupa tabel yang terdiri dari ID pelajaran dan nama mata pelajaran. Administrator atau guru dapat menambah, mengubah maupun menghapus mata pelajaran. Tombol add digunakan untuk

memasukkan data mata pelajaran baru atau mengubah data lama. Tombol cancel digunakan bila membatalkan proses menambah atau mengubah data mata pelajaran.

e) Halaman *User Log*

Halaman *user log* merupakan tabel untuk menampilkan hasil tes semua siswa bila dilihat berdasarkan satu mata pelajaran. Data tes yang ditampilkan meliputi: nama peserta, soal, jawaban dan skor. Untuk menampilkan halaman ini terlebih dahulu memilih mata pelajaran pada kotak yang disediakan.



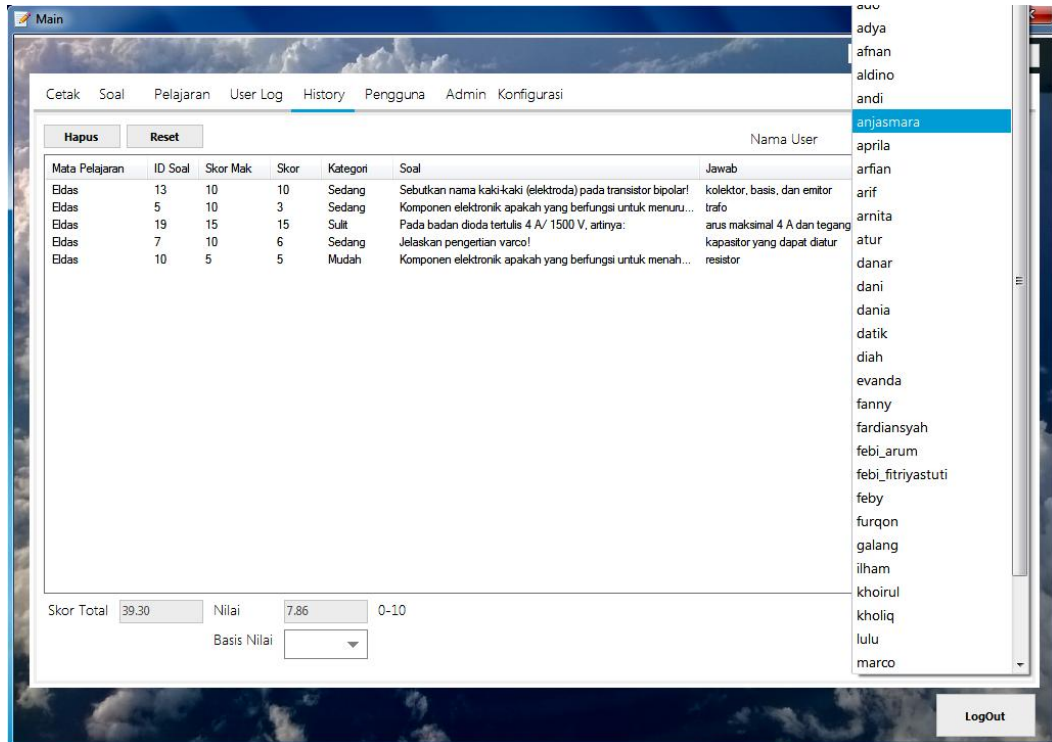
| ID Siswa | Jumlah Soal | Skor Maksimal | Skor | Nilai (0-10) | Nilai (0-100) | Nilai Huruf (A-E) |
|--|-------------|---------------|-------|--------------|---------------|-------------------|
| <input type="checkbox"/> kholiq | 5 | 50.00 | 46.90 | 9.38 | 93.80 | A- |
| <input type="checkbox"/> fardiansyah | 5 | 50.00 | 45.80 | 9.16 | 91.60 | A- |
| <input type="checkbox"/> furqon | 5 | 50.00 | 37.25 | 7.45 | 74.50 | C+ |
| <input type="checkbox"/> andi | 5 | 50.00 | 37.50 | 7.50 | 75.00 | C+ |
| <input type="checkbox"/> danar | 5 | 50.00 | 45.00 | 9.00 | 90.00 | B+ |
| <input type="checkbox"/> dari | 5 | 50.00 | 39.35 | 7.87 | 78.70 | B- |
| <input type="checkbox"/> arya | 5 | 50.00 | 40.00 | 8.00 | 80.00 | B- |
| <input type="checkbox"/> marco | 5 | 50.00 | 35.50 | 7.10 | 71.00 | C+ |
| <input type="checkbox"/> aprila | 5 | 50.00 | 42.25 | 8.45 | 84.50 | B |
| <input type="checkbox"/> feby | 5 | 50.00 | 36.20 | 7.24 | 72.40 | C+ |
| <input type="checkbox"/> arifan | 5 | 50.00 | 42.75 | 8.55 | 85.50 | B |
| <input type="checkbox"/> evanida | 5 | 50.00 | 41.50 | 8.30 | 83.00 | B |
| <input type="checkbox"/> anjasmara | 5 | 50.00 | 39.30 | 7.86 | 78.60 | B- |
| <input type="checkbox"/> adya | 5 | 50.00 | 40.15 | 8.03 | 80.30 | B- |
| <input type="checkbox"/> mano | 5 | 50.00 | 32.45 | 6.49 | 64.90 | C- |
| <input type="checkbox"/> adam | 5 | 50.00 | 42.75 | 8.55 | 85.50 | B |
| <input type="checkbox"/> aldino | 5 | 50.00 | 32.20 | 6.44 | 64.40 | C- |
| <input type="checkbox"/> afnan | 5 | 50.00 | 44.60 | 8.92 | 89.20 | B+ |
| <input type="checkbox"/> ilham | 5 | 50.00 | 45.10 | 9.02 | 90.20 | B+ |
| <input type="checkbox"/> khoiril | 5 | 50.00 | 29.90 | 5.98 | 59.80 | D+ |
| <input type="checkbox"/> febi_fitriyastuti | 5 | 50.00 | 39.10 | 7.82 | 78.20 | B- |
| <input type="checkbox"/> lulu | 5 | 50.00 | 33.20 | 6.64 | 66.40 | C |
| <input type="checkbox"/> diah | 5 | 50.00 | 35.60 | 7.12 | 71.20 | C+ |
| <input type="checkbox"/> atur | 5 | 50.00 | 43.00 | 8.60 | 86.00 | B+ |
| <input type="checkbox"/> dania | 5 | 50.00 | 35.55 | 7.11 | 71.10 | C+ |
| <input type="checkbox"/> febi_anum | 5 | 50.00 | 30.10 | 6.02 | 60.20 | D+ |
| <input type="checkbox"/> galang | 5 | 50.00 | 33.30 | 6.66 | 66.60 | C |

Gambar 26. Halaman *User Log*

f) Halaman *History*

Halaman *history* digunakan untuk menampilkan data hasil tes dari satu siswa untuk semua mata pelajaran. Untuk menampilkan halaman ini terlebih dahulu memilih nama siswa yang akan ditampilkan. Informasi yang

ditampilkan dalam history meliputi: mata pelajaran, ID soal, hasil (skor) dan kategori.

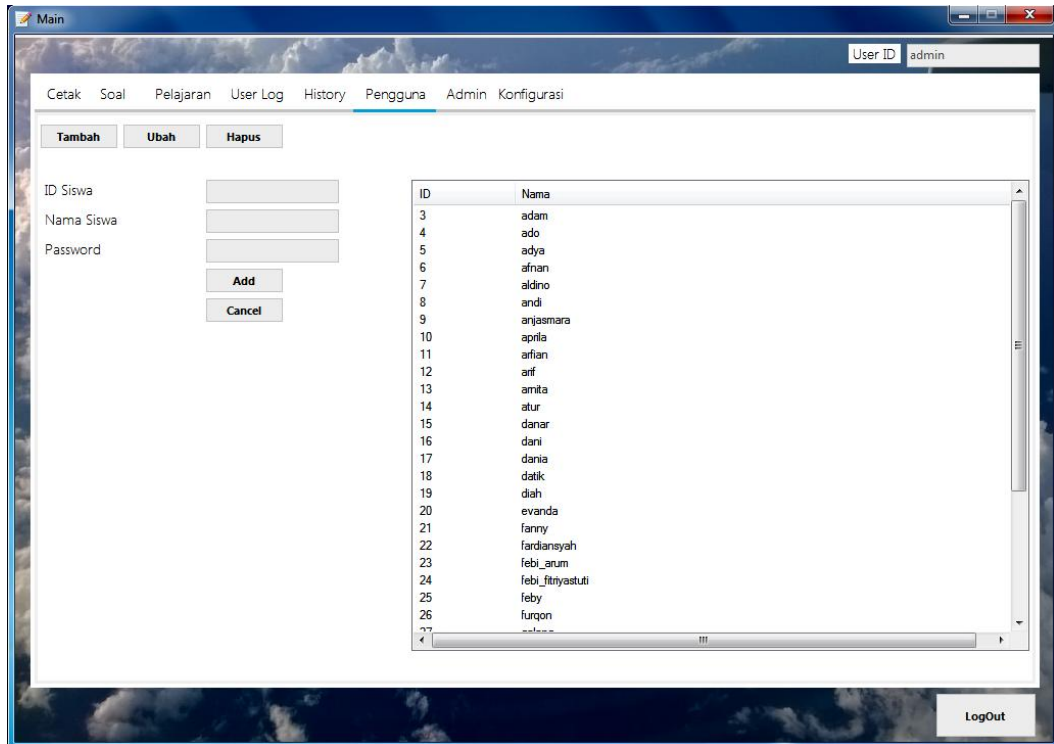


Gambar 27. Halaman History

g) Halaman Pengguna

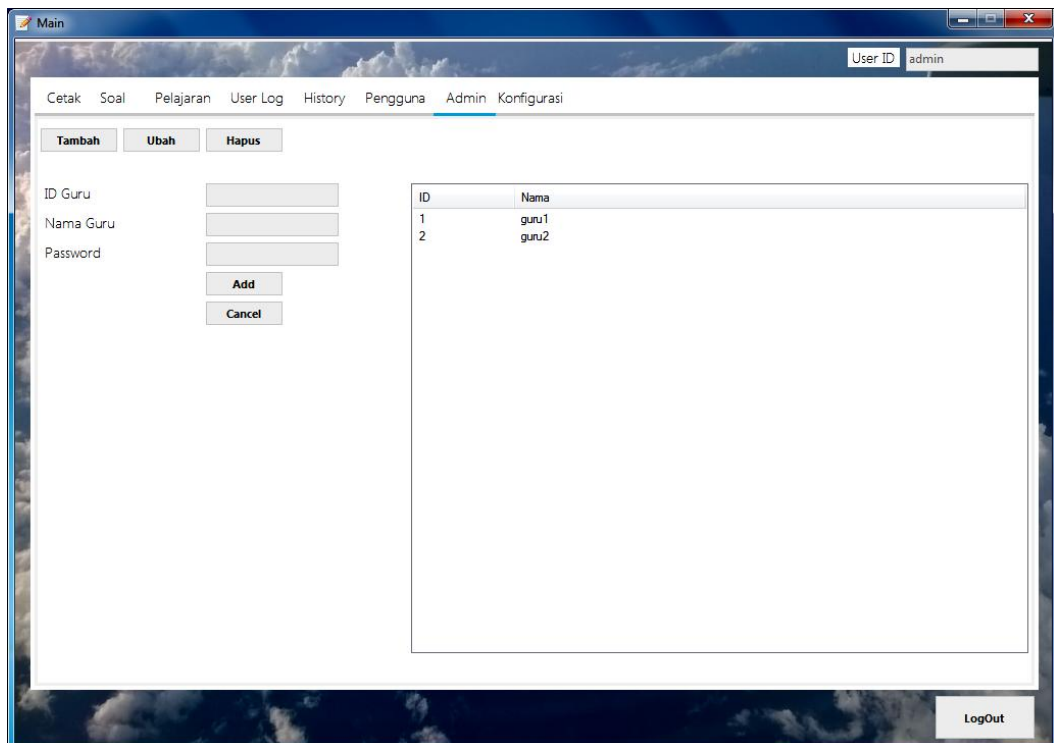
Halaman pengguna hanya bisa digunakan oleh administrator dan guru. Sebelum siswa melaksanakan tes, terlebih dahulu siswa harus dibuatkan ID dan password oleh administrator atau guru. Bila siswa belum memiliki ID dan password maka siswa belum bisa menggunakan program ini.

Halaman pengguna memiliki tiga fungsi yaitu untuk menambah user siswa, mengubah user siswa dan menghapus user siswa. Bila administrator atau guru akan menghapus data seorang siswa maka sebaiknya data hasil tes siswa yang bersangkutan dicetak terlebih dahulu apabila nanti diperlukan.



Gambar 28. Halaman Pengguna

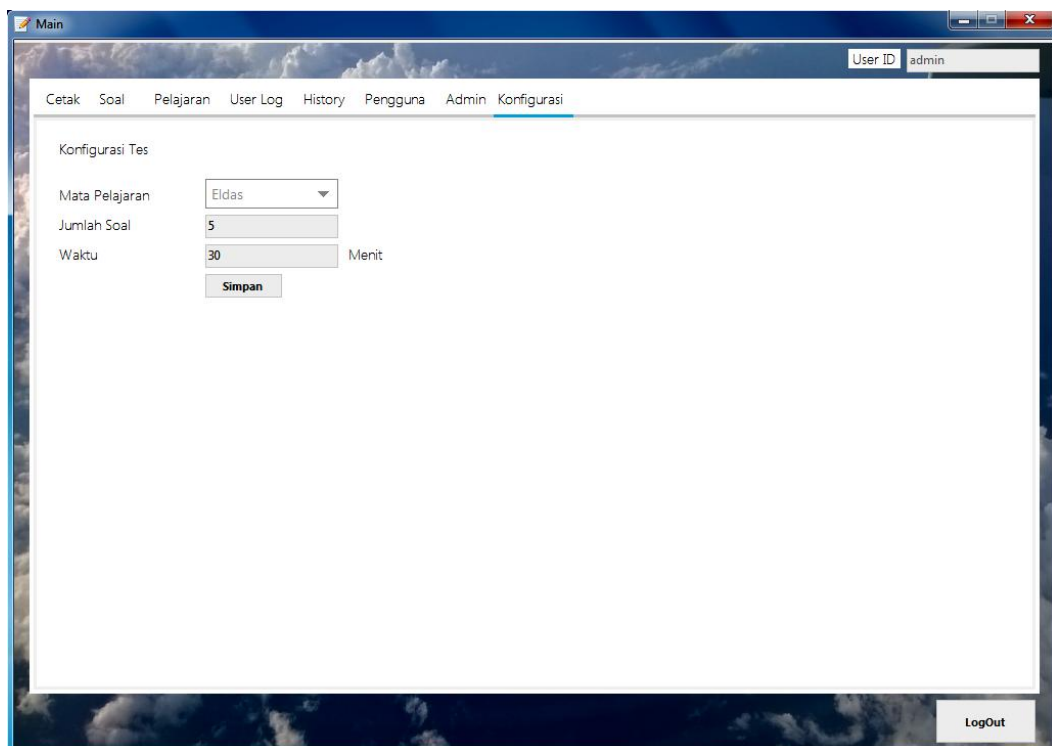
h) Halaman Admin



Gambar 29. Halaman Admin

Halaman admin hanya bisa dibuka oleh administrator. Bagi administrator halaman ini berguna untuk membuat ID user guru. Administrator dapat menambah, mengubah atau menghapus ID user guru. Proses ini mirip dengan halaman pengguna. Data user guru berupa ID guru dan nama guru.

i) Halaman konfigurasi



Gambar 30. Halaman konfigurasi

Halaman konfigurasi merupakan bagian untuk mengatur tes yang akan dilakukan. Pengaturan tes antara lain mata pelajaran yang diujikan, lama waktu maksimal dan jumlah soal tes. Perubahan pengaturan dilakukan dengan mengisi pengaturan pada kotak yang telah disediakan. Pengaturan baru akan disimpan bila tombol simpan telah ditekan.

j) Halaman *Logout*

Halaman logout digunakan apabila user administrator, user guru maupun user siswa telah selesai menggunakan program. User yang telah selesai menggunakan program, apabila tidak melakukan logout maka semua data masih dalam posisi terbuka. Logout dilakukan untuk mengamankan data agar tidak dipergunakan oleh pihak yang tidak memiliki wewenang. Tombol logout terletak pada bagian paling bawah dari tampilan program.

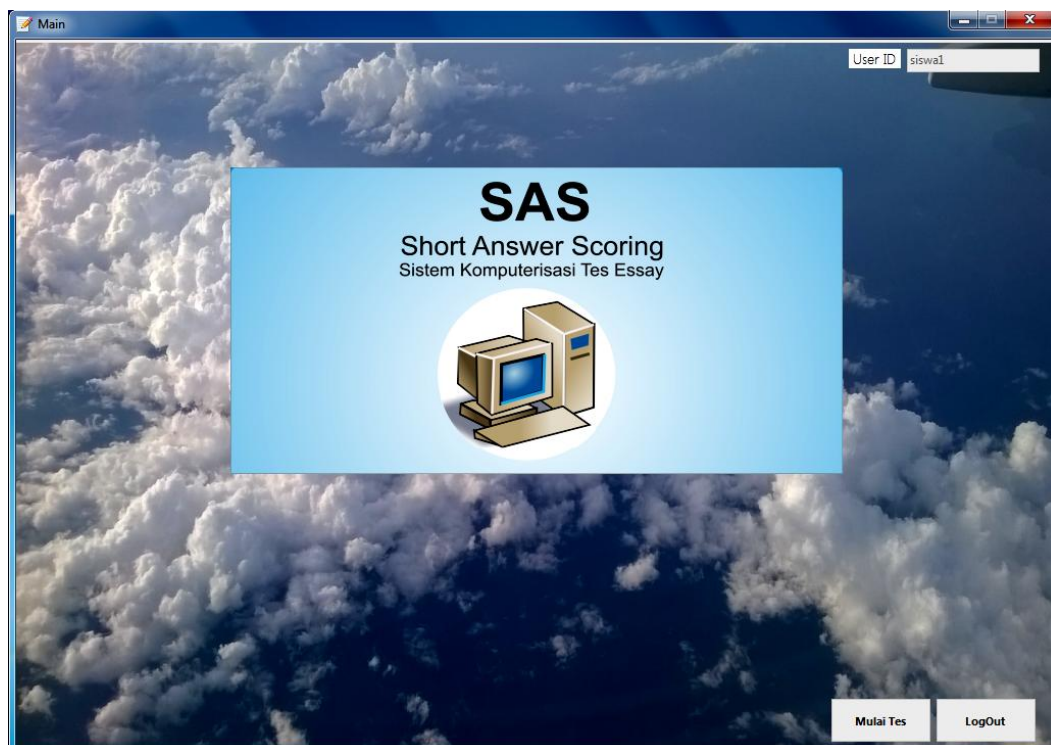


Gambar 31. Halaman *Logout*

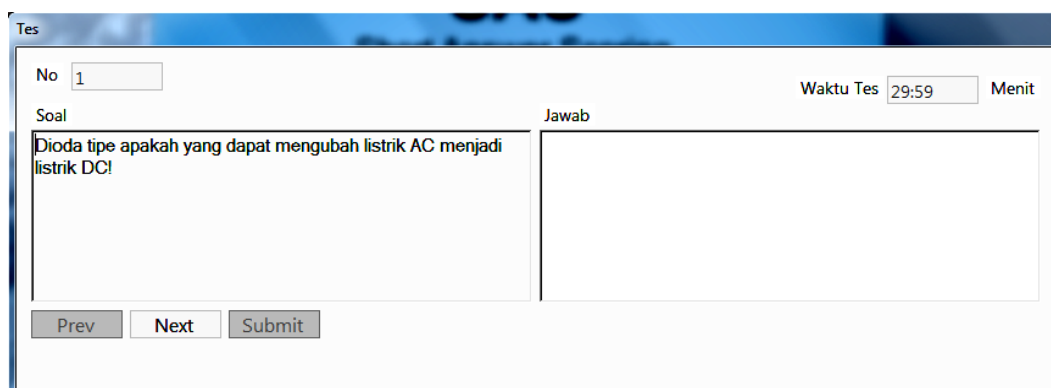
k) Tampilan Halaman Tes

Halaman tes digunakan khusus oleh user siswa. Pada saat siswa telah melakukan proses login maka akan tampil halaman awal tes. Siswa dapat memulai tes dengan menekan tombol mulai tes. Saat siswa telah menekan tombol mulai tes, saat itu juga waktu tes mulai dihitung.

Halaman soal tes terdiri dari tombol prev, next dan submit. Tombol prev digunakan untuk menuju ke soal sebelumnya. Tombol next digunakan untuk menuju soal selanjutnya. Tombol submit digunakan bila tes telah selesai dikerjakan. Bagian tengah kiri halaman merupakan tempat untuk menampilkan butir soal. Bagian tengah kanan digunakan sebagai tempat menuliskan jawaban dari pertanyaan tes.



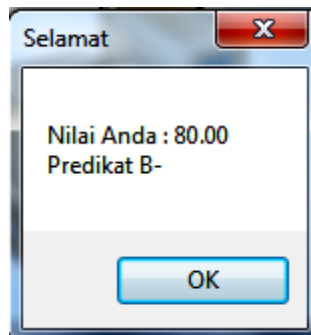
Gambar 32. Halaman Awal Tes



Gambar 33. Halaman Tes

I) Halaman Hasil Tes

Halaman hasil tes hanya muncul setelah siswa selesai mengerjakan soal tes. Program akan melakukan koreksi jawaban siswa dan menghitung skor total. Skor akhir merupakan angka dari 0 sampai 100 dan nilai predikat berupa huruf dari A-E.



Gambar 34. Halaman Hasil Tes

e. Testing

Tabel 13. Hasil Pengujian Fungsional Program

| No | Fungsi program | Hasil | |
|-----|--|-------|-----|
| | | Ya | Tdk |
| 1. | Program dapat dijalankan di lingkungan <i>windows</i> | ✓ | |
| 2. | User dapat <i>login</i> sesuai dengan kondisinya | ✓ | |
| 3. | User guru dapat membuka program sesuai kewenangannya | ✓ | |
| 4. | User siswa dapat membuka program sesuai kewenangannya | ✓ | |
| 5. | User dapat melakukan <i>logout</i> sesuai dengan kondisinya | ✓ | |
| 6. | Program dapat ditutup kembali melalui tombol yang disediakan | ✓ | |
| 7. | Program dapat menampilkan hasil kerja sesuai dengan menu yang dipilih | ✓ | |
| 8. | Program dapat menyimpan hasil kerja sesuai dengan masukan yang diberikan | ✓ | |
| 9. | Program dapat memproses masukan menjadi keluaran sesuai dengan fungsi yang dikehendaki | ✓ | |
| 10. | Hasil kerja yang ditampilkan program adalah benar sesuai fungsi yang dikehendaki | ✓ | |
| 11. | Data yang disimpan program adalah benar sesuai dengan fungsinya | ✓ | |
| 12. | Keluaran hasil program adalah benar sesuai dengan fungsinya | ✓ | |
| 13. | Program dapat membaca masukan berupa jawaban soal untuk proses selanjutnya | ✓ | |
| 14. | Program dapat menyimpan hasil tes yang telah dikerjakan siswa | ✓ | |
| 15. | Program dapat menampilkan hasil koreksi jawaban tes | ✓ | |
| 16. | Program dapat menampilkan butir soal secara acak | ✓ | |
| 17. | Program dapat menampilkan butir soal dengan bobot sama | ✓ | |
| 18. | Program dapat menampilkan nilai akhir dari jawaban soal | ✓ | |
| 19. | Program dapat menampilkan soal selanjutnya walaupun soal sebelumnya belum dijawab | ✓ | |
| 20. | Program dapat menampilkan kembali soal yang sebelumnya belum dijawab | ✓ | |
| 21. | Program dapat menampilkan kembali hasil tes yang telah dikoreksi | ✓ | |
| 22. | Hasil koreksi jawaban soal adalah benar sesuai dengan koreksi yang dilakukan secara manual | ✓ | |
| 23. | Hasil tes yang ditampilkan dapat menggambarkan kemampuan siswa peserta tes | ✓ | |
| 24. | Hasil analisis program terhadap butir soal mampu mengestimasi kemampuan siswa peserta tes | ✓ | |

Proses pengujian merupakan penerapan program yang dibuat dalam kondisi sebenarnya, guna mendapatkan fungsionalitas program. Pengujian program meliputi dua langkah, yaitu pengujian internal untuk pelacakan kesalahan (*debugging*) dan pengujian eksternal untuk mengetahui fungsionalitas dan kehandalan program (validasi program).

Pengujian program dilakukan bersamaan dengan proses pembuatan program. Program dibuat berdasarkan analisis kebutuhan, desain sistem, *data flow diagram* (DFD), serta flow chart program. Kode program disusun sedemikian rupa sehingga meminimalkan kesalahan pada *syntax error*. Kode program juga diuji dalam *run error* serta *logic error* sehingga hasil program sesuai dengan harapan. Tabel 13 merupakan hasil pengujian fungsional program yang telah dilakukan

2. Deskripsi Data Hasil Pengujian

Pengujian program dilakukan dalam dua tahapan yaitu pengujian internal program dan pengujian eksternal program. Pengujian internal merupakan pengujian *debugging* yang dilakukan oleh pengembang (peneliti). Pengujian eksternal dilakukan dari sisi pengguna (user).

Pengujian internal digunakan untuk mengetahui bahwa program yang dibuat telah bekerja sesuai yang dikehendaki. Pengujian yang dilakukan oleh pengembang antara lain: melacak kesalahan kode program (*debugging*). Urutan pengujian internal adalah a) menguji kalimat perintah program (*syntax error*), b) menguji program saat dijalankan (*running error*), dan c) menguji kebenaran hasil program saat dijalankan (*logic error*). Pengujian internal dilakukan sejak awal

pembuatan program hingga program selesai. Tabel 14 merupakan data hasil pengujian yang dilakukan secara internal.

Pengujian eksternal meliputi dua tahap yaitu: a) pengujian *alfa* yang dilakukan oleh pengguna pertama (*first user/guru*), dan b) pengujian *beta* yang dilakukan oleh pengguna akhir (*end user/siswa*). Pengujian *alfa* dilakukan oleh guru mata pelajaran elektronika dasar di SMK N 2 Yogyakarta mulai dari memanggil program, mengisi data mata pelajaran, mengisi data soal, membuat user siswa, menkonfigurasi tes, dan melihat hasil tes, hingga program ditutup. Pengujian *beta* dilakukan melalui kegiatan tes menggunakan program. Pengujian dilakukan untuk melihat kemampuan algoritma koreksi jawaban tes.

B. Analisis Data

1. Validitas Soal Tes *Essay*

a. Expert Judgement

Validitas butir soal tes *essay* sebelum dilakukan ujicoba pada siswa terlebih dahulu dilakukan dengan menunjukkan butir soal kepada expert judgement. Hasil validitas ini digunakan untuk menyempurnakan butir soal tes *essay*. Hasil validitas oleh dosen ahli berupa pembenahan redaksi dalam penulisan butir soal.

b. Empiris

Data validitas soal tes *essay* secara empiris didapatkan dengan menghitung data hasil ujicoba tes. Ujicoba tes dilakukan dengan memberikan tes pada 30 siswa. Rumus yang digunakan dalam perhitungan validitas soal tes *essay* merupakan rumus korelasi *Pearson product-moment*, sebagai berikut:

$$r_{xy} = \frac{N \sum XY - (\sum X)(\sum Y)}{\sqrt{\{N \sum X^2 - (\sum X)^2\}\{N \sum Y^2 - (\sum Y)^2\}}}$$

r_{xy} = Koefisien korelasi Pearson
 ΣXY = Jumlah hasil kali skor X dan Y
 ΣX = Jumlah skor X
 ΣY = Jumlah skor Y
 ΣX^2 = Jumlah kuadrat skor X
 ΣY^2 = Jumlah kuadrat skor y
 N = jumlah siswa

Tabel 14. Interpretasi Koefisien Korelasi (r_{xy}) untuk Uji Validitas

| No | Koefisien Korelasi Pearson (r_{xy}) | Kategori |
|----|---|---------------|
| 1 | 0,80-1,00 | Sangat Tinggi |
| 2 | 0,60-0,80 | Tinggi |
| 3 | 0,40-0,60 | Cukup |
| 4 | 0,20-0,40 | Rendah |
| 5 | 0,00-0,20 | Sangat Rendah |

Sumber: (Arikunto, 2004: 89)

Tabel 15. Validitas Empiris Butir Soal Tes *Essay*

| No Soal | Validitas (r_{xy}) | Kategori |
|---------|------------------------|---------------|
| 1 | 0,11 | Sangat Rendah |
| 2 | 0,25 | Rendah |
| 3 | 0,60 | Tinggi |
| 4 | 0,57 | Cukup |
| 5 | 0,53 | Cukup |
| 6 | 0,43 | Cukup |
| 7 | 0,53 | Cukup |
| 8 | 0,70 | Tinggi |
| 9 | 0,25 | Rendah |
| 10 | 0,51 | Cukup |
| 11 | 0,10 | Sangat Rendah |
| 12 | 0,14 | Sangat Rendah |
| 13 | 0,56 | Cukup |
| 14 | 0,56 | Cukup |
| 15 | 0,52 | Cukup |
| 16 | 0,46 | Cukup |
| 17 | 0,51 | Cukup |
| 18 | 0,44 | Cukup |
| 19 | 0,34 | Rendah |
| 20 | 0,13 | Sangat Rendah |
| 21 | 0,61 | Tinggi |
| 22 | 0,65 | Tinggi |
| 23 | 0,54 | Cukup |
| 24 | 0,49 | Cukup |
| 25 | 0,42 | Cukup |

Hasil Perhitungan validitas soal diperoleh data 4 soal kategori sangat rendah, 3 soal kategori rendah, 14 soal kategori cukup dan 4 soal kategori tinggi.

2. Reliabilitas Soal Tes *Essay*

Reliabilitas soal tes *essay* diperoleh setelah melakukan perhitungan data hasil uji soal. Ujicoba tes dilakukan dengan memberikan tes pada 30 siswa. Rumus yang digunakan dalam perhitungan soal *essay* adalah *Alpha Cronbach*. Hasil perhitungan sebagai berikut:

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

r_{11} = Reliabilitas Soal

k = Banyaknya Butir Soal = 25

$\sum \sigma_n^2$ = Jumlah Varian Butir = 9,82

σ_t^2 = Varians Total = 51,56

Tabel 16. Interpretasi Koefisien Reliabilitas (r_{11}) untuk Uji Reliabilitas

| No | Koefisien Reliabilitas (r_{11}) | Kategori |
|----|-------------------------------------|---------------|
| 1 | 0,00-0,20 | Kecil |
| 2 | 0,20-0,40 | Rendah |
| 3 | 0,40-0,60 | Sedang |
| 4 | 0,60-0,80 | Tinggi |
| 5 | 0,80-1,00 | Sangat Tinggi |

Sumber: (Guilford dalam Ruseffendi, 2005: 160)

$$r_{11} = \left[\frac{25}{25-1} \right] \left[1 - \frac{9,82}{51,56} \right]$$

$$r_{11} = 0,84$$

Hasil perhitungan tingkat reliabilitas diperoleh nilai koefisien reliabilitas $r_{11} = 0,84$. Oleh karena itu tingkat reliabilitas soal adalah termasuk kategori Sangat tinggi.

3. Tingkat Kesulitan Butir Soal Tes *Essay*

Tingkat kesulitan butir soal tes *essay* diperoleh dari perhitungan data hasil ujicoba soal tes. Ujicoba tes dilakukan dengan memberikan tes pada 30 siswa.

Rumus yang digunakan dalam perhitungan tingkat kesulitan soal adalah sebagai berikut:

$$p = \frac{n_i}{N}$$

p = tingkat kesulitan

n_i = banyaknya siswa yang menjawab item soal dengan benar

N = banyaknya siswa yang menjawab item soal

Tabel 17. Klasifikasi Tingkat Kesulitan

| No | Tingkat Kesulitan (p) | Klasifikasi |
|----|---------------------------|-------------|
| 1 | 0,00-0,29 | Sulit |
| 2 | 0,30-0,69 | Sedang |
| 3 | 0,70-1,00 | Mudah |

Sumber: (Arikunto, 2004: 210)

Tabel 18. Tingkat kesulitan Butir Soal Tes *Essay*

| No Soal | Tingkat Kesulitan (p) | Kategori |
|---------|---------------------------|----------|
| 1 | 0,28 | Sulit |
| 2 | 0,29 | Sulit |
| 3 | 0,69 | Sedang |
| 4 | 0,68 | Sedang |
| 5 | 0,67 | Sedang |
| 6 | 0,67 | Sedang |
| 7 | 0,69 | Sedang |
| 8 | 0,67 | Sedang |
| 9 | 0,83 | Mudah |
| 10 | 0,80 | Mudah |
| 11 | 0,27 | Sulit |
| 12 | 0,29 | Sulit |
| 13 | 0,65 | Sedang |
| 14 | 0,62 | Sedang |
| 15 | 0,63 | Sedang |
| 16 | 0,60 | Sedang |
| 17 | 0,83 | Mudah |
| 18 | 0,82 | Mudah |
| 19 | 0,28 | Sulit |
| 20 | 0,27 | Sulit |
| 21 | 0,67 | Sedang |
| 22 | 0,68 | Sedang |
| 23 | 0,67 | Sedang |
| 24 | 0,81 | Mudah |
| 25 | 0,82 | Mudah |

Hasil Perhitungan tingkat kesulitan soal diperoleh data 6 soal kategori sulit, 13 soal kategori sedang dan 6 soal kategori mudah.

4. Daya Beda Butir Soal Tes *Essay*

Daya beda butir soal tes essay diperoleh dari perhitungan data hasil ujicoba soal tes. Ujicoba tes dilakukan dengan memberikan tes pada 30 siswa. Rumus yang digunakan dalam perhitungan tingkat kesulitas soal adalah sebagai berikut:

$$d = \frac{n_{iT}}{N_T} - \frac{n_{iR}}{N_R}$$

d = daya beda

n_{iT} = banyaknya penjawab item soal dengan benar dari kelompok tinggi.

n_T = banyaknya penjawab dari kelompok tinggi

n_{iR} = banyaknya penjawab item soal dengan benar dari kelompok rendah.

n_R = banyaknya penjawab dari kelompok rendah.

Tabel 19. Kriteria Daya Beda Butir Soal

| No | Daya Beda (d) | Kategori |
|----|-------------------|-------------|
| 1 | 0,00-0,19 | Rendah |
| 2 | 0,20-0,39 | Cukup |
| 3 | 0,40-0,69 | Baik |
| 4 | 0,70-1,00 | Baik Sekali |
| 5 | Negatif | Tidak Baik |

Sumber: (Arikunto, 2004: 213)

Hasil Perhitungan daya beda soal diperoleh data 6 soal dengan daya beda rendah, 12 soal dengan daya beda cukup dan 7 soal dengan daya beda baik.

Tabel 20. Daya Beda Butir Soal Tes *Essay*

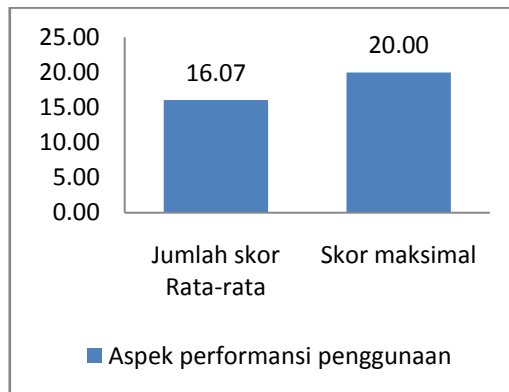
| No Soal | Daya Beda (<i>d</i>) | Keterangan |
|---------|------------------------|------------|
| 1 | 0,10 | Rendah |
| 2 | 0,10 | Rendah |
| 3 | 0,42 | Baik |
| 4 | 0,36 | Cukup |
| 5 | 0,42 | Baik |
| 6 | 0,33 | Cukup |
| 7 | 0,31 | Cukup |
| 8 | 0,44 | Baik |
| 9 | 0,23 | Cukup |
| 10 | 0,42 | Baik |
| 11 | 0,06 | Rendah |
| 12 | 0,11 | Rendah |
| 13 | 0,31 | Cukup |
| 14 | 0,33 | Cukup |
| 15 | 0,33 | Cukup |
| 16 | 0,24 | Cukup |
| 17 | 0,43 | Baik |
| 18 | 0,36 | Cukup |
| 19 | 0,12 | Rendah |
| 20 | 0,12 | Rendah |
| 21 | 0,43 | Baik |
| 22 | 0,42 | Baik |
| 23 | 0,33 | Cukup |
| 24 | 0,32 | Cukup |
| 25 | 0,33 | Cukup |

5. Kemudahan Pengoperasian *Software* Tes *Essay*

Kemudahan pengoperasian *software* tes *essay* diketahui berdasarkan angket yang diisi oleh pengguna setelah menggunakan produk *software*. Pertanyaan yang diajukan merujuk pada hal-hal yang menyangkut kemudahan penggunaan *software*. Hasil penilaian melalui angket adalah sebagai berikut:

Tabel 21. Konversi Skor Aspek Performansi Penggunaan

| Interval Skor | Kategori |
|-----------------------|-------------|
| $16,25 < X \leq 20$ | Sangat baik |
| $12,5 < X \leq 16,25$ | Baik |
| $8,75 < X \leq 12,5$ | Cukup |
| $5 < X \leq 8,75$ | Kurang |



Gambar 35. Hasil Angket Kemudahan Penggunaan

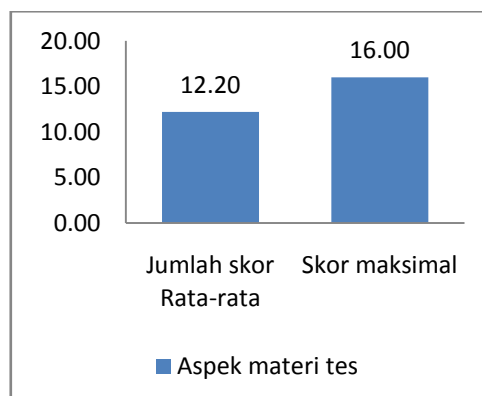
Jumlah rerata skor aspek performansi penggunaan adalah 16,07 atau kategori baik, berdasarkan tabel konversi skor.

6. Kemudahan Pengubahan Soal Tes dalam *Software*

Kemudahan pengubahan soal tes dalam *software* diketahui berdasarkan angket yang diisi oleh pengguna setelah menggunakan produk *software*. Pertanyaan yang diajukan merujuk pada hal-hal yang menyangkut materi pembuatan butir soal tes. Hasil penilaian melalui angket adalah sebagai berikut:

Tabel 22. Konversi Skor Aspek Materi Tes

| Interval Skor | Kategori |
|------------------|-------------|
| $13 < X \leq 16$ | Sangat baik |
| $10 < X \leq 13$ | Baik |
| $7 < X \leq 10$ | Cukup |
| $4 < X \leq 7$ | Kurang |



Gambar 36. Hasil Angket Kemudahan Pengubahan Soal

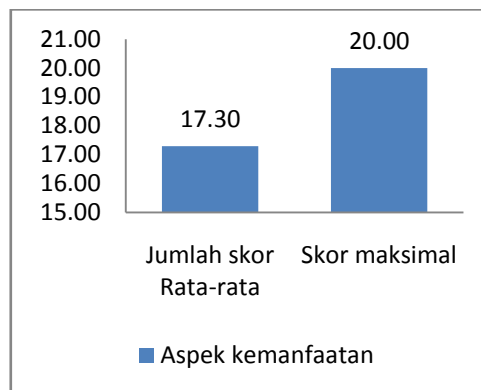
Jumlah rerata skor aspek materi tes adalah 12,20 atau kategori baik, berdasarkan tabel konversi skor.

7. Kemampuan *Software* dalam Membantu Guru

Kemampuan *software* dalam membantu guru diketahui berdasarkan angket yang diisi oleh pengguna setelah menggunakan produk *software*. Pertanyaan yang diajukan merujuk pada hal-hal yang menyangkut manfaat penggunaan *software*. Hasil penilaian melalui angket adalah sebagai berikut:

Tabel 23. Konversi Skor Aspek Kemanfaatan

| Interval Skor | Kategori |
|-----------------------|-------------|
| $16,25 < X \leq 20$ | Sangat baik |
| $12,5 < X \leq 16,25$ | Baik |
| $8,75 < X \leq 12,5$ | Cukup |
| $5 < X \leq 8,75$ | Kurang |



Gambar 37. Hasil Angket Kemampuan *Software* Membantu Guru

Jumlah rerata skor aspek kemanfaatan adalah 17,30 atau kategori sangat baik, berdasarkan tabel konversi skor.

8. Ketepatan Koreksi Hasil Tes *Essay* pada *Software*

Ketepatan koreksi hasil tes *essay* pada *software* diketahui dengan membandingkan hasil koreksi tes pada *software* dengan hasil koreksi secara manual. Berdasarkan Tabel 17 persentase ketepatan koreksi hasil tes adalah 96,67%.

$$\begin{aligned} \text{Persentase ketepatan koreksi} &= \frac{\text{Jumlah siswa hasil koreksi tepat}}{\text{Jumlah siswa}} \times 100\% \\ &= \frac{29}{30} \times 100\% = 96,67\% \end{aligned}$$

Tabel 24. Ketepatan Koreksi Jawaban tes

| No Siswa | Hasil koreksi jawaban | Persentase (%) |
|-------------|-----------------------|----------------|
| Siswa ke-1 | Tepat | 100% |
| Siswa ke-2 | Tepat | 100% |
| Siswa ke-3 | Tepat | 100% |
| Siswa ke-4 | Tepat | 100% |
| Siswa ke-5 | Tepat | 100% |
| Siswa ke-6 | Tepat | 100% |
| Siswa ke-7 | Tepat | 100% |
| Siswa ke-8 | Tepat | 100% |
| Siswa ke-9 | Tepat | 100% |
| Siswa ke-10 | Tepat | 100% |
| Siswa ke-11 | Tepat | 100% |
| Siswa ke-12 | Tepat | 100% |
| Siswa ke-13 | Tepat | 100% |
| Siswa ke-14 | Tepat | 100% |
| Siswa ke-15 | Tepat | 100% |
| Siswa ke-16 | Tepat | 100% |
| Siswa ke-17 | Tepat | 100% |
| Siswa ke-18 | Tepat | 100% |
| Siswa ke-19 | Tepat | 100% |
| Siswa ke-20 | Tepat | 100% |
| Siswa ke-21 | Tepat | 100% |
| Siswa ke-22 | Tidak Tepat | 0% |
| Siswa ke-23 | Tepat | 100% |
| Siswa ke-24 | Tepat | 100% |
| Siswa ke-25 | Tepat | 100% |
| Siswa ke-26 | Tepat | 100% |
| Siswa ke-27 | Tepat | 100% |
| Siswa ke-28 | Tepat | 100% |
| Siswa ke-29 | Tepat | 100% |
| Siswa ke-30 | Tepat | 100% |
| Rata-rata | | 96,67% |

9. Kecepatan Koreksi Hasil Tes *Essay* pada *Software*

Kecepatan koreksi hasil tes *essay* pada *software* diketahui dengan menggunakan *stopwatch*. Perhitungan waktu dimulai dari siswa yang telah selesai mengerjakan soal kemudian melakukan submit jawaban. Waktu yang

dibutuhkan dalam proses koreksi satu butir soal adalah dalam interval 0 detik sampai 6 detik.

10. Keakuratan Koreksi Hasil Tes *Essay* pada *Software*

Keakuratan koreksi hasil tes *essay* pada *software* diketahui dengan membandingkan nilai hasil koreksi dengan menggunakan *software* dan nilai hasil koreksi secara manual. Tabel 18 merupakan nilai hasil koreksi dengan menggunakan *software* dan dengan cara koreksi manual.

Tabel 25. Nilai Hasil Tes Siswa

| No | Nilai (Koreksi Software) | Nilai (Koreksi Manual) | Deviasi (%) |
|-----------|--------------------------|------------------------|-------------|
| 1 | 84 | 84 | 0% |
| 2 | 58 | 58 | 0% |
| 3 | 76 | 76 | 0% |
| 4 | 68 | 68 | 0% |
| 5 | 62 | 62 | 0% |
| 6 | 74 | 74 | 0% |
| 7 | 78 | 78 | 0% |
| 8 | 82 | 82 | 0% |
| 9 | 84 | 84 | 0% |
| 10 | 68 | 68 | 0% |
| 11 | 78 | 78 | 0% |
| 12 | 84 | 84 | 0% |
| 13 | 90 | 90 | 0% |
| 14 | 76 | 76 | 0% |
| 15 | 68 | 68 | 0% |
| 16 | 68 | 68 | 0% |
| 17 | 82 | 82 | 0% |
| 18 | 90 | 90 | 0% |
| 19 | 58 | 58 | 0% |
| 20 | 76 | 76 | 0% |
| 21 | 70 | 70 | 0% |
| 22 | 70 | 78 | 11,5% |
| 23 | 64 | 64 | 0% |
| 24 | 88 | 88 | 0% |
| 25 | 56 | 56 | 0% |
| 26 | 92 | 92 | 0% |
| 27 | 66 | 66 | 0% |
| 28 | 70 | 70 | 0% |
| 29 | 60 | 60 | 0% |
| 30 | 76 | 76 | 0% |
| Rata-rata | | | 0,38% |

C. Kajian Produk

Produk *software* tes *essay* berbantuan komputer yang telah berhasil dibuat. Pada awalnya *software* tidak mempunyai kemampuan untuk memilih butir soal sesuai dengan tingkat kesukaran butir soal (mudah, sedang dan sulit) dan hanya memilih soal secara acak dari bank soal. Proses pemilihan butir soal setelah revisi yaitu dilakukan dengan memilih butir soal dengan menggunakan komposisi soal mudah 20%, soal sedang 60% dan soal sulit 20%. Tingkat kesulitan soal diketahui dari input data soal saat proses penambahan soal baru oleh user guru.

Revisi program pertama dilakukan saat pengguna siswa akan memulai proses tes. Siswa terlebih dahulu ditanya kesiapan mengerjakan tes dengan menggunakan "*message box*". Bila siswa belum siap maka siswa dapat melakukan "*Log out*" dan bila siswa telah siap maka selanjutnya halaman tes akan ditampilkan.

Revisi program kedua dilakukan saat pengguna siswa saat melakukan tes. Tombol "*Submit*" yang digunakan untuk mengakhiri tes dinonaktifkan bila siswa belum membuka semua soal yang diberikan. Saat siswa telah membuka semua soal tes, maka tombol "*Submit*" akan diaktifkan. Saat siswa menekan tombol "*Submit*" untuk mengakhiri tes maka akan ditampilkan peringatan untuk meyakinkan bahwa soal telah selesai dijawab.

Produk hasil akhir program dikembangkan dengan mengacu *software* pada umumnya, yaitu: terdapat prosedur operasi baku sebelum digunakan. Tahap awal adalah persiapan *software* dan program pendukung (.Net Framework 4.0). Tahap kedua dengan melakukan instalasi program pada komputer. Tahap

ketiga adalah menguji program untuk memastikan *software* telah bekerja dan siap digunakan.

D. Pembahasan Hasil Penelitian

Kelebihan *software* ini yaitu dalam proses pengujian atau tes soal yang diterima oleh siswa berbeda karena pemilihan butir soal dilakukan secara acak dari bank soal. komposisi tingkat kesulitan soal sama antar siswa dengan perbandingan 20% soal sulit, 60% soal sedang dan 20% soal mudah. Karena butir soal yang dikeluarkan oleh program berbeda maka akan memperkecil kesempatan siswa untuk mencontek jawaban teman lainnya. Kombinasi jumlah soal yang dikeluarkan oleh program untuk 30 siswa ditunjukkan oleh Tabel 19.

Tabel 26. Persentase Butir Soal Yang Dikeluarkan Program

| No Butir | Jumlah Soal | Persentase(%) | No Butir | Jumlah Soal | Persentase(%) |
|----------|-------------|---------------|----------|-------------|---------------|
| 1 | 3 | 2% | 14 | 7 | 5% |
| 2 | 6 | 4% | 15 | 13 | 9% |
| 3 | 6 | 4% | 16 | 7 | 5% |
| 4 | 9 | 6% | 17 | 2 | 1% |
| 5 | 3 | 2% | 18 | 4 | 3% |
| 6 | 5 | 3% | 19 | 7 | 5% |
| 7 | 5 | 3% | 20 | 4 | 3% |
| 8 | 7 | 5% | 21 | 8 | 5% |
| 9 | 5 | 3% | 22 | 6 | 4% |
| 10 | 7 | 5% | 23 | 7 | 5% |
| 11 | 5 | 3% | 24 | 7 | 5% |
| 12 | 5 | 3% | 25 | 5 | 3% |
| 13 | 7 | 5% | | | |

Penyusunan pembahasan hasil penelitian ini mengacu pada pertanyaan penelitian yang telah dijabarkan pada BAB II. Pembahasan ini merupakan jawaban dari pertanyaan penelitian yang telah dibuat.

1. Validitas Soal Tes *Essay*

Tingkat validitas soal tes *essay* diketahui dengan penilaian dosen dari jurusan pendidikan teknik elektro sebagai ahli materi. Jumlah ahli materi yang menguji tingkat validitas soal tes *essay* adalah 2 orang. Berdasarkan penilaian dan masukan dalam proses pembuatan soal tes *essay* maka soal tes *essay* dinyatakan valid.

Validitas soal tes *essay* yang telah disusun dinyatakan valid dari segi isi dan konstruk. Soal tes *essay* dinyatakan telah valid dari segi isi karena materi tes *essay* telah disesuaikan dengan silabus mata pelajaran elektronika dasar di SMK N 2 Yogyakarta. Sedangkan soal tes *essay* dinyatakan valid dari segi konstruksi tes yang telah disesuaikan dengan matrik soal yang telah disusun.

Hasil Perhitungan validitas soal diperoleh data 4 soal kategori sangat rendah, 3 soal kategori rendah, 14 soal kategori cukup dan 4 soal kategori tinggi. Berdasarkan data empiris validitas soal tes *essay* maka perlu perbaikan pada butir soal agar tingkat validitasnya dapat menjadi lebih baik.

2. Reliabilitas Soal Tes *Essay*

Butir soal *essay* yang dikembangkan perlu diketahui nilai reliabilitasnya. Nilai reliabilitas soal tes *essay* diperoleh diketahui dengan teknik perhitungan *Alpha Cronbach*. Skor butir yang diperoleh siswa dikonversi menjadi skala 1-4, sehingga mempermudah proses penghitungan nilai reliabilitas soal tes *essay* yang dikembangkan. Soal tes *essay* terdiri dari tiga tipe yaitu soal mudah dengan skor maksimal 5, soal sedang dengan skor maksimal 10 dan soal sulit dengan skor maksimal 15. Soal mudah dikonversi menjadi skala 1-4 dengan mengubah yaitu skor 0,00-1,25 menjadi skala 1, skor 1,26-2,50 menjadi skala

2, skor 2,51-3,75 menjadi skala 3 dan skor 3,76-5,00 menjadi skala 4. Soal sedang dikonversi menjadi skala 1-4 dengan mengubah yaitu skor 0,00-2,50 menjadi skala 1, skor 2,51-5,00 menjadi skala 2, skor 5,01-7,50 menjadi skala 3 dan skor 7,51-10,00 menjadi skala 4. Soal sulit dikonversi menjadi skala 1-4 dengan mengubah yaitu skor 0,00-3,75 menjadi skala 1, skor 3,76-7,50 menjadi skala 2, skor 7,51-11,25 menjadi skala 3 dan skor 11,26-15,00 menjadi skala 4.

Hasil perhitungan tingkat reliabilitas diperoleh nilai koefisien reliabilitas $r_{11} = 0,84$. Oleh karena itu berdasarkan tabel kategori tingkat reliabilitas soal adalah termasuk kategori Sangat tinggi.

3. Tingkat Kesulitan Soal Tes *Essay*

Tingkat kesulitan butir tes *essay* yang digunakan dalam pengujian diketahui berdasarkan matrik yang disusun saat pengembangan butir tes. Tingkat kesulitan butir tes dikategorikan menjadi tiga golongan yaitu soal mudah, soal sedang dan soal sulit. Berdasarkan kategori tingkat kesulitan soal maka setiap siswa akan menerima soal berjumlah 5 dengan komposisi tingkat kesulitan 1 soal mudah, 3 soal sedang dan 1 soal sulit.

Tingkat kesulitan butir soal juga diuji secara empirik setelah didapatkan hasil tes yang dikerjakan oleh siswa. Hasil Perhitungan tingkat kesulitan soal diperoleh data 6 soal kategori sulit, 13 soal kategori sedang dan 6 soal kategori mudah.

4. Daya Beda Soal Tes *Essay*

Daya beda butir soal diketahui dengan membagi nilai siswa menjadi kelompok atas dan kelompok bawah. Masing masing kelompok atas dan

kelompok bawah berjumlah 15 orang siswa berdasarkan tingkat kesulitan yang diperoleh kelompok atas dan tingkat kesulitan yang diperoleh kelompok bawah maka daya beda dapat diketahui.

Daya beda butir tes *essay* yang digunakan dalam pengujian dihitung berdasarkan hasil dari ujicoba. Hasil Perhitungan daya beda soal diperoleh data 6 soal dengan daya beda rendah, 12 soal dengan daya beda cukup dan 7 soal dengan daya beda baik.

5. Kemudahan pengoperasian *software*

Software tes *essay* berbantuan komputer memiliki kemudahan dalam pengoperasiannya karena pengguna dapat melakukan banyak fungsi pada *software* antara lain: mampu membedakan pengguna berdasarkan ID dan password, pengguna dapat melakukan pencetakan data tes siswa, pengguna dapat mengubah data soal dengan fungsi tambah, ubah dan hapus, pengguna dapat mengubah data pengguna siswa maupun guru dengan fungsi tambah, ubah dan hapus, pengguna dapat mengubah data mata pelajaran pada *software* dengan fungsi tambah, ubah dan hapus dan pengguna dapat mengatur konfigurasi tes sesuai kebutuhan.

Jumlah rerata skor aspek performansi penggunaan adalah 16,07 atau kategori baik, berdasarkan tabel konversi skor. Oleh karena itu kemudahan pengoperasian *software* berdasarkan angket adalah baik.

6. Kemudahan perubahan soal pada *software*

Software tes *essay* memiliki fungsi untuk mengubah data soal dalam bank soal. Pengguna (guru) memiliki wewenang dalam menambahkan soal baru, mengubah soal lama dan menghapus soal lama menggunakan tombol

yang telah disediakan dalam *software*. Data yang ada dalam setiap butir soal antara lain adanya data ID soal, skor tiap butir, tingkat kesukaran, daya beda, soal dan kunci jawaban. Soal akan dimasukkan dalam *database* bila guru telah mengisi semua kolom, sehingga dapat mengurangi kesalahan dalam input data soal. Pada kolom tertentu misalnya pada kolom skor, tingkat kesukaran dan daya beda, karakter yang boleh diketikkan hanya angka saja dan otomatis akan menghapus bila karakter yang bukan angka.

Jumlah rerata skor aspek materi tes adalah 12,20 atau kategori baik, berdasarkan tabel konversi skor. Oleh karena itu kemudahan pengubahan soal pada *software* berdasarkan angket adalah baik.

7. Kemampuan *software* dalam membantu guru

Software tes *essay* bisa membantu guru karena beberapa fungsinya yaitu: soal yang dikeluarkan dalam tes dipilih secara acak dan berbeda antar komputer sehingga meminimalkan kerjasama dalam pengerjaannya, jawaban tes akan dinilai oleh *software* berdasarkan algoritma yang telah disusun sehingga guru tidak perlu melakukan penilaian, data tes dapat dilihat kembali oleh guru apabila diperlukan.

Jumlah rerata skor aspek kemanfaatan adalah 17,30 atau kategori sangat baik, berdasarkan tabel konversi skor. Oleh karena itu kemampuan *software* dalam membantu guru berdasarkan angket adalah baik.

8. Ketepatan koreksi hasil tes

Fungsi yang ada pada *software* tes *essay* yaitu kemampuan untuk melakukan koreksi hasil tes. Ketepatan koreksi hasil tes dapat dilihat dari data jawaban tes dari 30 siswa. *Software* tes *essay* mampu memberikan skor sesuai

dengan rentang skor yang ditentukan, software mampu menjumlahkan skor hasil tes pada tahap akhir dan diberi nilai dengan rentang 0-10 atau 0-100 dengan tepat. Pada halaman *user log* dan *history software* menampilkan konversi nilai hasil tes menjadi nilai huruf A-E dengan tepat.

9. Kecepatan koreksi hasil tes

Pengukuran kecepatan koreksi hasil tes berdasarkan pada proses perhitungan oleh dengan bantuan *stopwatch*. Kecepatan koreksi soal sangat dipengaruhi oleh panjang jawaban tes yang dibuat oleh siswa, yang berarti jika kalimat yang dibuat semakin banyak maka waktu yang dibutuhkan untuk melakukan koreksi bertambah. Jumlah soal yang dikerjakan siswa berjumlah 5 butir. Berdasarkan interval waktu saat siswa menekan tombol submit sampai nilai akhir ditampilkan maka diketahui lama waktu koreksi jawaban siswa bervariasi antara 10 detik hingga 30 detik. Sehingga waktu yang dibutuhkan untuk koreksi satu butir soal diperkirakan 2 sampai 6 detik.

10. Keakuratan koreksi hasil tes

Keakuratan koreksi jawaban siswa dapat dilihat berdasarkan skor yang diberikan oleh *software* pada jawaban siswa. Keakuratan hasil koreksi software tes essay setelah dibandingkan dengan koreksi secara manual memperoleh persentase 96,67%. Hal ini disebabkan karena terdapat kesalahan hasil koreksi pada satu siswa. Kesalahan hasil koreksi ini terjadi dikarenakan software tes essay masih memiliki kekurangan dalam *database* kamus sinonim kata, sehingga terjadi kesalahan dalam memberikan nilai.

BAB V

KESIMPULAN DAN SARAN

A. Kesimpulan

1. Validitas soal tes *essay*

Validitas soal tes *essay* yang digunakan dalam perangkat lunak diperoleh melalui proses expert judgement. Hasil Perhitungan validitas secara empiris soal diperoleh data 4 soal kategori sangat rendah, 3 soal kategori rendah, 14 soal kategori cukup dan 4 soal kategori tinggi.

2. Reliabilitas soal tes *essay*

Reliabilitas butir soal tes *essay* dihitung berdasarkan data hasil uji coba soal tes *essay*. Nilai yang diperoleh berdasarkan perhitungan dengan rumus *Alpha Cronbach* adalah 0.84 atau dapat dikatakan butir soal sangat reliabel.

3. Tingkat kesulitan soal tes *essay*

Tingkat kesulitan soal tes *essay* yang digunakan dalam pengujian diuji secara empiris dengan hasil 6 soal kategori sulit, 13 soal kategori sedang dan 6 soal kategori mudah.

4. Daya beda soal tes *essay*

Hasil Perhitungan daya beda soal diperoleh data 6 soal dengan daya beda rendah, 12 soal dengan daya beda cukup dan 7 soal dengan daya beda baik.

5. Kemudahan pengoperasian *software tes essay*

Software tes *essay* berbantuan komputer memiliki kemudahan dalam pengoperasian karena dapat melakukan banyak fungsi antara lain: mampu membedakan pengguna, fungsi cetak, pengubahan data soal, pengguna, mata

pelajaran dan konfigurasi tes. Jumlah rerata skor aspek performansi penggunaan adalah 16,07 atau kategori baik, berdasarkan tabel konversi skor.

6. Kemudahan pengubahan soal tes dalam *software*

Software tes essay memiliki fungsi untuk mengubah data soal dalam bank soal. Pengguna guru merupakan pengguna yang memiliki wewenang dalam menambah, mengubah dan menghapus soal. Jumlah rerata skor aspek materi tes adalah 12,20 atau kategori baik, berdasarkan tabel konversi skor.

7. Kemampuan *software* dalam membantu guru

Kemampuan *software* dalam membantu guru karena memiliki fungsi yaitu: soal yang dikeluarkan acak sehingga meminimalkan kerjasama, *software* akan memberi nilai pada jawaban siswa dan hasil tes dapat ditampilkan kembali. Jumlah rerata skor aspek kemanfaatan adalah 17,30 atau kategori sangat baik, berdasarkan tabel konversi skor.

8. Ketepatan koreksi hasil tes

Ketepatan koreksi hasil tes *software* tes *essay* dapat dilihat berdasarkan data jawaban tes dari 30 siswa dan ketepatan *software* dalam melakukan pemberian skor dan penilaian.

9. Kecepatan koreksi hasil tes

Kecepatan koreksi hasil tes dipengaruhi oleh jawaban tes yang dibuat oleh siswa. Berdasarkan uji coba saat proses tes maka diketahui kecepatan koreksi satu butir soal adalah 2 hingga 6 detik.

10. Keakuratan koreksi hasil tes

Keakuratan hasil koreksi tes diketahui dengan membandingkan hasil nilai koreksi oleh *software* dan secara manual. Diketahui persentase keakuratan hasil koreksi adalah 96,67%.

B. Keterbatasan Produk

Produk *software* yang telah berhasil dibuat masih memiliki kelemahan yaitu *database* kamus sinonim untuk proses penskoran jawaban siswa masih terbatas. Keterbatasan *database* kamus sinonim membuat *software* belum bisa mendeteksi semua kata yang dimasukkan dan kata dalam bahasa asing. Akibat dari masih terbatasnya *database* sinonim ini *software* dapat melakukan kesalahan dalam penskoran jawaban siswa jika belum mampu mendeteksi kata yang belum tercatat dalam *database*.

Kelemahan *software* lainnya yaitu belum mendukung apabila akan digunakan pada komputer yang telah memanfaatkan jaringan baik LAN maupun internet. *Software* tes *essay* harus dilakukan penginstalan dan dilakukan pengaturan pada tiap komputer bila akan digunakan.

C. Pengembangan Produk Lebih Lanjut

Arah pengembangan *software* tes *essay* lebih lanjut dapat dikembangkan dengan meningkatkan jumlah *database* kamus sinonim untuk meningkatkan ketepatan dalam melakukan proses penskoran hasil tes. Pengembangan *software* tes *essay* lainnya yaitu dengan menambahkan kemampuan untuk memanfaatkan jaringan berupa LAN maupun internet untuk mempermudah melakukan proses tes.

D. Saran

Software tes essay berbantuan komputer dapat digunakan lebih lanjut sebagai media untuk melakukan evaluasi atau penilaian di SMK sehingga sekolah dapat mengikuti perkembangan IPTEK.

Tes *essay* dengan media komputer merupakan masih jarang digunakan oleh sekolah dalam proses evaluasi pembelajaran. Sekolah dapat menggunakan *software tes essay* berbantuan komputer untuk mengenalkan pada siswa tes *essay* dengan media komputer.

Guru dapat memanfaatkan fasilitas yang terdapat pada *software tes essay* berbantuan komputer untuk mempercepat dan mempermudah proses evaluasi pembelajaran, sehingga proses pengambilan keputusan proses KBM pada tahap selanjutnya lebih cepat.

Software tes essay berbantuan komputer dapat menjadi media bagi siswa untuk latihan mengerjakan soal secara mandiri sehingga mampu mengetahui kemampuan dan penguasaan materi.

DAFTAR PUSTAKA

- Asep Jihad dan Abdul Haris. (2008). *Evaluasi Pembelajaran*. Yogyakarta: Multi Pressindo
- Djemari Mardapi. (2008). *Teknik Penyusunan Instrumen Tes dan Nontes*. Yogyakarta: Mitra Cendikia
- Gomaa, Wael H. dan Fahmy, Aly A. (2012). *Short Answer Grading Using String Similarity And Corpus-Based Similarity*. International Journal of Advanced Computer Science and Applications, Vol. 3, No. 11, 2012
- Lawrence M. Rudner. (2012). *Computer-Based Testing Terms*. Diakses dari [http:// www.gmac.com/ why-gmac/ gmac-news/ gmnews/ 2012/ january/ demystifying- the- gmat- computer- based- testing- terms.aspx](http://www.gmac.com/why-gmac/gmac-news/gmnews/2012/january/demystifying-the-gmat-computer-based-testing-terms.aspx). pada tanggal 18 Maret 2014, Jam 21.30 WIB.
- Nana Sudjana. (2005). *Penilaian Hasil Proses Belajar Mengajar*. Bandung: Remaja Rosdakarya
- Page, E.B. dan Paulus, D.H. (1968). *The Analysis of Essays by Computer*. USA: U.S. Department of Health, Education, and Welfare, Office of Education, Bureau of Research
- Pressman, Roger S. (2001). *Software Engineering: a Practitioner's Approach-5th edition*. USA: McGraw Hill
- Puspitarini. (2013). *Masih Banyak Guru Kesulitan Implementasi Kurikulum Baru*. Diakses dari [http:// news.okezone.com/ read/ 2013/ 12/ 16/ 560/ 913092/ masih- banyak- guru- kesulitan- implementasi- kurikulum- baru](http://news.okezone.com/read/2013/12/16/560/913092/masih-banyak-guru-kesulitan-implementasi-kurikulum-baru). pada tanggal 12 Januari 2014, Jam 21.30 WIB.
- Saifuddin Azwar. (1997). *Reliabilitas dan Validitas*. Yogyakarta: Pustaka Pelajar
- Saifuddin Azwar. (2002). *Tes Prestasi Fungsi Pengembangan Pengukuran Prestasi Belajar*. Yogyakarta: Pustaka Pelajar
- Samsul Hadi. (2013). *Pengembangan Computerized Adaptive Test Berbasis Web*. Yogyakarta: Aswaja Pressindo
- Suharsimi Arikunto. (2004). *Dasar-dasar Evaluasi Pendidikan*. Jakarta: Bumi Aksara
- Walter R.Borg dan Meredith D. Gall. (1983). *Educational Research An Introduction Chapter 18*. New York: Longman Inc

LAMPIRAN

Lampiran I. *Source Code* Program

1. Code program Login

```
OleDbConnection MyConn = new OleDbConnection("Provider =
Microsoft.ACE.OLEDB.12.0; Data Source = ShortAnswerGrading.mdb");
public bool StatusAdmin = false, StatusGuru = false, StatusSiswa = false;
private string mLogUserName;
public string LogUserName { get { return mLogUserName; } }
public Login()
{
InitializeComponent();
}
private void mButtonLogin_Click(object sender, EventArgs e)//Login sebagai guru
/ siswa
{
Tutup = false;
string StrCmd, Username, Password, Status;
OleDbCommand Cmd;
OleDbDataReader DataReader;
if (string.IsNullOrEmpty(mTextBoxUser.Text)) //cek username kosong
{
MetroFramework.MetroMessageBox.Show(this, "Masukkan Username",
"Perhatian", MessageBoxButtons.OK, MessageBoxIcon.Information);
mTextBoxUser.Focus();
goto akhir;
}
if (string.IsNullOrEmpty(mTextBoxPass.Text)) //cek password kosong
{
MetroFramework.MetroMessageBox.Show(this, "Masukkan Passwords",
"Perhatian", MessageBoxButtons.OK, MessageBoxIcon.Information);
mTextBoxPass.Focus();
goto akhir;
}
StrCmd = "SELECT * FROM tbUser WHERE User = '" + mTextBoxUser.Text + "'";
Cmd = new OleDbCommand(StrCmd, MyConn);
MyConn.Open();
DataReader = Cmd.ExecuteReader();
if (DataReader.HasRows)
{
while (DataReader.Read())
{
Username = DataReader.GetString(0).ToLower();
Password = DataReader.GetString(1);
Status = DataReader.GetString(2);
if ((Username == mTextBoxUser.Text.ToLower()) && (Password ==
mTextBoxPass.Text))
{
if (Status == "1") { StatusGuru = true; }

```

```

if (Status == "2") { StatusSiswa = true; }
mLogUserName = Username;
}
else
{
MetroFramework.MetroMessageBox.Show(this, "Password salah", "Login gagal",
MessageBoxButtons.OK, MessageBoxIcon.Information);
mTextBoxPass.Clear();
mTextBoxPass.Focus();
}
}
}
else
{
MetroFramework.MetroMessageBox.Show(this, "Username atau password salah",
"Login gagal", MessageBoxButtons.OK, MessageBoxIcon.Information);
mTextBoxUser.Clear();
mTextBoxUser.Focus();
mTextBoxPass.Clear();
}
MyConn.Close();
if ((StatusGuru == true) || (StatusSiswa == true))//step saat login sukses
{
DialogResult = DialogResult.OK;
}
akhir: ;
}
bool Tutup = false;
private void mButtonCancel_Click(object sender, EventArgs e)
{
DialogResult = DialogResult.Cancel;
Tutup = true;
}
private void Login_Load(object sender, EventArgs e)
{
mTextBoxUser.Clear();
mTextBoxPass.Clear();
StatusGuru = false;
StatusSiswa = false;
StatusAdmin = false;
}
private void mButtonAdmin_Click(object sender, EventArgs e)//Login sebagai
admin
{
Tutup = false;
string StrCmd, Username, Password;
OleDbCommand Cmd;
OleDbDataReader DataReader;

```

```

if (string.IsNullOrEmpty(mTextBoxUser.Text))           //cek username kosong
{
MetroFramework.MetroMessageBox.Show(this, "Masukkan Username",
"Perhatian", MessageBoxButtons.OK, MessageBoxIcon.Information);
mTextBoxUser.Focus();
goto akhir;
}
if (string.IsNullOrEmpty(mTextBoxPass.Text))           //cek password kosong
{
MetroFramework.MetroMessageBox.Show(this, "Masukkan Passwords",
"Perhatian", MessageBoxButtons.OK, MessageBoxIcon.Information);
mTextBoxPass.Focus();
goto akhir;
}
StrCmd = "SELECT * FROM tbAdmin WHERE User = '" + mTextBoxUser.Text +
''''";
Cmd = new OleDbCommand(StrCmd, MyConn);
MyConn.Open();
DataReader = Cmd.ExecuteReader();
if (DataReader.HasRows)
{
while (DataReader.Read())
{
Username = DataReader.GetString(0).ToLower();
Password = DataReader.GetString(1);
if ((Username == mTextBoxUser.Text.ToLower()) && (Password ==
mTextBoxPass.Text))
{
StatusAdmin = true;
mLogUserName = Username;
}
else
{
MetroFramework.MetroMessageBox.Show(this, "Password salah", "Login gagal",
MessageBoxButtons.OK, MessageBoxIcon.Information);
mTextBoxPass.Clear();
mTextBoxPass.Focus();
}
}
}
else
{
MetroFramework.MetroMessageBox.Show(this, "Username atau password salah",
"Login gagal", MessageBoxButtons.OK, MessageBoxIcon.Information);
mTextBoxUser.Clear();
mTextBoxUser.Focus();
mTextBoxPass.Clear();
}
}

```



```

MyConn.Close();
if (StatusAdmin == true)//step saat login sukses
{
DialogResult = DialogResult.OK;
}
akhir: ;
}
private void Login_FormClosing(object sender, FormClosingEventArgs e)
{
if (Tutup == true)
{
if (MessageBox.Show("Tutup Program?", "Peringatan",
MessageBoxButtons.YesNo) == DialogResult.Yes)
{
e.Cancel = false;
}
else { e.Cancel = true; }
}
}
}

```

2. Code program utama

```

public MainForm()
{
InitializeComponent();
}
Login LoginForm = new Login();
Tambah_Soal Tambah_Soal = new Tambah_Soal();
Tes Tes = new Tes();
OleDbConnection MyConn = new OleDbConnection("Provider =
Microsoft.ACE.OLEDB.12.0; Data Source = ShortAnswerGrading.mdb");
bool EditMapel = false, EditGuru = false, EditSiswa = false;
string IDMapelOld, NamaMapelOld, IDGuruOld, NamaGuruOld, IDSiswaOld,
NamaSiswaOld;
int PointerMapelOld, PointerGuruOld, PointerSiswaOld;
string LogUserName;
public static string nMapelSoal;
public static bool EditSoal = false;
public static string mIDSoal, mMapelSoal, mBbtSoal, mIKSoal, mDBSoal,
mButirSoal, mKunciJawaban;
public static string Soal, UserName;
public static int WaktuTes;
public static bool IsOdd(int value)
{
return value % 2 != 0;
}
private string KategoriHuruf(float NilaiPredikat)
{

```

```

string[] LevelPredikat = { "A", "A-", "B+", "B", "B-", "C+", "C", "C-", "D+", "D-" };
string Predikat = "";
if (NilaiPredikat <= 100 && NilaiPredikat >= 96) { Predikat = LevelPredikat[0]; }
else if (NilaiPredikat < 96 && NilaiPredikat >= 91) { Predikat = LevelPredikat[1];
}
else if (NilaiPredikat < 91 && NilaiPredikat >= 86) { Predikat = LevelPredikat[2];
}
else if (NilaiPredikat < 86 && NilaiPredikat >= 81) { Predikat = LevelPredikat[3];
}
else if (NilaiPredikat < 81 && NilaiPredikat >= 76) { Predikat = LevelPredikat[4];
}
else if (NilaiPredikat < 76 && NilaiPredikat >= 71) { Predikat = LevelPredikat[5];
}
else if (NilaiPredikat < 71 && NilaiPredikat >= 66) { Predikat = LevelPredikat[6];
}
else if (NilaiPredikat < 66 && NilaiPredikat >= 61) { Predikat = LevelPredikat[7];
}
else if (NilaiPredikat < 61 && NilaiPredikat >= 56) { Predikat = LevelPredikat[8];
}
else { Predikat = LevelPredikat[9]; }
return Predikat;
}
private void mButtonClose_Click(object sender, EventArgs e)
{
    if (MessageBox.Show("Tutup Program?", "Peringatan",
        MessageBoxButtons.YesNo) == DialogResult.Yes)
    {
        Close();
    }
}
private void MainForm_Load(object sender, EventArgs e)
{
    Login();
    MoveUserToRTBoxMapel();
    MoveUserToRTBoxGuru();
    MoveUserToRTBoxSiswa();
    LoadDataSetting();
    LoadDataSoal();
    LoadExistDataTes();
}
private void MainForm_FormClosing(object sender, FormClosingEventArgs e)
{
    if (Tutup != true)
    {
        if (MessageBox.Show("Tutup Program?", "Peringatan",
            MessageBoxButtons.YesNo) == DialogResult.Yes)
        {
            e.Cancel = false;
        }
    }
}

```

```

}
else { e.Cancel = true; }
}
}
private void mButtonLogOut_Click(object sender, EventArgs e)
{
Login();
}
bool Tutup = false;
private void Login()
{
this.Hide();
if (LoginForm.ShowDialog() == DialogResult.OK)
{
Tutup = false;
this.Show();
mTabControl1.SelectedIndex = 0;
LogUserName = LoginForm.LogUserName;
mTextBoxUserId.Text = LogUserName;
if (LoginForm.StatusGuru == true)
{
pictureBoxWelcome.Visible = false;
mTabControl1.Visible = true;
mButtonStartTest.Visible = false;
MessageBox.Show("Login As Teacher");
mComboBoxPrint.Items.Clear();
string[] Data = { "Data Soal", "Data Mata Pelajaran", "Data User Log", "Data
History", "Data User Siswa" };
mComboBoxPrint.Items.Clear();
foreach (string Menu in Data) { mComboBoxPrint.Items.Add(Menu); }
}
else if (LoginForm.StatusSiswa == true)
{
pictureBoxWelcome.Location = new Point(210, 125);
pictureBoxWelcome.Visible = true;
mTabControl1.Visible = false;
mButtonStartTest.Visible = true;
MessageBox.Show("Login As Student");
}
else if (LoginForm.StatusAdmin == true)
{
pictureBoxWelcome.Visible = false;
mTabControl1.Visible = true;
mButtonStartTest.Visible = false;
MessageBox.Show("Login As Admin");
string[] Data = { "Data Soal", "Data Mata Pelajaran", "Data User Log", "Data
History", "Data User Siswa", "Data User Guru" };
mComboBoxPrint.Items.Clear();

```

```

foreach (string Menu in Data) { mComboBoxPrint.Items.Add(Menu); }
}
}
else { Tutup = true; this.Close(); }
LoadDataSoal();
LoadExistDataTes();
}
private void mTabControl1_Selecting(object sender, TabControlCancelEventArgs
e)
{
if (LoginForm.StatusGuru == true)
{
if (mTabControl1.SelectedIndex == 6) { e.Cancel = true; }
}
}
private void mButtonAddMapel_Click(object sender, EventArgs e)
{
int hitung = 0;
foreach (string Baris in rTextBoxMapel.Lines) { hitung++; }
if (hitung == 0) { mTextBoxIDMapel.Text = (hitung + 1).ToString(); }
if (hitung > 0) { mTextBoxIDMapel.Text = (hitung).ToString(); }
}
private void MoveUserToRTBoxMapel()
{
string StrCmd, IDMapel, Mapel;
OleDbCommand Cmd;
OleDbDataReader DataReader;
rTextBoxMapel.Clear();
StrCmd = "SELECT * FROM tbMapel";
Cmd = new OleDbCommand(StrCmd, MyConn);
MyConn.Open();
DataReader = Cmd.ExecuteReader();
if (DataReader.HasRows)
{
while (DataReader.Read())
{
IDMapel = DataReader.GetString(0);
Mapel = DataReader.GetString(1);
rTextBoxMapel.Text += IDMapel + "~" + Mapel + "\n";
}
}
MyConn.Close();
listViewMapel.Items.Clear();
string[] DataMapel = new string[2];
string Words;
ListItem DataMapelToWrite;
int hitung = 0;
foreach (string Baris in rTextBoxMapel.Lines) { hitung++; }

```

```

for (int i = 0; i < hitung - 1; i++)
{
string[] Lines;
Words = rTextBoxMapel.Lines[i];
Lines = Words.Split('~');
DataMapel[0] = Lines[0];
DataMapel[1] = Lines[1];
DataMapelToWrite = new ListViewItem(DataMapel);
listViewMapel.Items.Add(DataMapelToWrite);
}
}
private void mButtonAddNewMapel_Click(object sender, EventArgs e)
{
string StrCmd;
OleDbCommand Cmd;
OleDbDataReader DataReader;
if (!(string.IsNullOrEmpty(mTextBoxIDMapel.Text)) &&
!(string.IsNullOrEmpty(mTextBoxNamaMapel.Text)))
{
if (EditMapel == false)
{
StrCmd = "INSERT INTO tbMapel ([ID], [Mapel]) VALUES ('" +
mTextBoxIDMapel.Text + "', '" + mTextBoxNamaMapel.Text + "')";
}
else
{
StrCmd = "UPDATE tbMapel SET [Mapel]='" + mTextBoxNamaMapel.Text + "
WHERE [ID]='" + mTextBoxIDMapel.Text + "'";
}
Cmd = new OleDbCommand(StrCmd, MyConn);
MyConn.Open();
try
{
DataReader = Cmd.ExecuteReader();
if (EditMapel == true) { RefreshDataMapel>NamaMapelOld,
mTextBoxNamaMapel.Text); }
EditMapel = false;
mTextBoxIDMapel.Clear();
mTextBoxNamaMapel.Clear();
}
catch { MessageBox.Show("ID Mapel Sudah dipakai"); }
MyConn.Close();
}
else { MessageBox.Show("Data belum lengkap"); }
MoveUserToRTBoxMapel();
}
private void mButtonEditMapel_Click(object sender, EventArgs e)
{

```

```

EditMapel = true;
string IDMapel, Mapel;
if (listViewMapel.SelectedItems.Count > 0)
{
int i = listViewMapel.SelectedIndices[0];
IDMapel = listViewMapel.Items[i].SubItems[0].Text;
Mapel = listViewMapel.Items[i].SubItems[1].Text;
mTextBoxIDMapel.Text = IDMapel;
mTextBoxNamaMapel.Text = Mapel;
IDMapelOld = IDMapel;
NamaMapelOld = Mapel;
PointerMapelOld = i;
}
}
private void mButtonCancelAddMapel_Click(object sender, EventArgs e)
{
mTextBoxIDMapel.Clear();
mTextBoxNamaMapel.Clear();
EditMapel = false;
}
private void mButtonDellMapel_Click(object sender, EventArgs e)
{
string IDMapel;
string StrCmd;
OleDbCommand Cmd;
OleDbDataReader DataReader;
if (listViewMapel.SelectedItems.Count > 0)
{
int i = listViewMapel.SelectedIndices[0];
IDMapel = listViewMapel.Items[i].SubItems[0].Text;
StrCmd = "DELETE * FROM tbMapel WHERE [ID]='" + IDMapel + "'";
Cmd = new OleDbCommand(StrCmd, MyConn);
MyConn.Open();
DataReader = Cmd.ExecuteReader();
MyConn.Close();
MoveUserToRTBoxMapel();
}
}
private void mComboBoxMapel_DropDown(object sender, EventArgs e)
{
string[] DataMapel = new string[2];
string Words;
int hitung = 0;
mComboBoxMapel.Items.Clear();
foreach (string Baris in rTextBoxMapel.Lines) { hitung++; }
for (int i = 0; i < hitung - 1; i++)
{
string[] Lines;

```

```

Words = rTextBoxMapel.Lines[i];
Lines = Words.Split('~');
DataMapel[0] = Lines[0];
DataMapel[1] = Lines[1];
mComboBoxMapel.Items.Add(Lines[1]);
}
}
private void mButtonAddSoal_Click(object sender, EventArgs e)
{
EditSoal = false;
nMapelSoal = mComboBoxMapel.Text;
if (mComboBoxMapel.SelectedIndex >= 0)
{
if (Tambah_Soal.ShowDialog() == DialogResult.OK)
{
string[] DataSoal = new string[6];
string Mapel = mComboBoxMapel.Text;
DataSoal[0] = Tambah_Soal.IDSoal;
DataSoal[1] = Tambah_Soal.BbtSoal;
DataSoal[2] = Tambah_Soal.IKSoal;
DataSoal[3] = Tambah_Soal.DBSoal;
DataSoal[4] = Tambah_Soal.ButirSoal;
DataSoal[5] = Tambah_Soal.KunciJawaban;
rTextBoxBankSoal.Text +=
DataSoal[0] + "~" + DataSoal[1] + "~" +
DataSoal[2] + "~" + DataSoal[3] + "~" +
DataSoal[4] + "~" + DataSoal[5] + "~" +
Mapel + "\n";
SaveDataSoal();
MoveSoalToBuffer(mComboBoxMapel.Text);
}
}
else { MessageBox.Show("Pilih Mata Pelajaran"); }
}
private void LoadToListviewMapel()
{
listViewSoal.Items.Clear();
string[] DataSoal = new string[8];
string Words;
ListViewItem DataSoalToWrite;
int hitung = 0;
foreach (string Baris in rTextBoxSoal.Lines) { hitung++; }
for (int i = 0; i < hitung - 1; i++)
{
string[] Lines;
Words = rTextBoxSoal.Lines[i];
Lines = Words.Split('~');
DataSoal[1] = Lines[0];

```

```

DataSoal[2] = Lines[1];
DataSoal[3] = Lines[2];
DataSoal[4] = Lines[3];
DataSoal[5] = Lines[4];
DataSoal[6] = Lines[5];
DataSoalToWrite = new ListViewItem(DataSoal);
listViewSoal.Items.Add(DataSoalToWrite);
}
}
private void MoveSoalToBuffer(string SelectedMapel)
{
string[] DataSoal = new string[7];
string Words;
int hitung = 0;//sekaligus pointer soal
rTextBoxSoal.Clear();
foreach (string Baris in rTextBoxBankSoal.Lines) { hitung++; }
for (int i = 0; i < hitung - 1; i++)
{
string[] Lines;
Words = rTextBoxBankSoal.Lines[i];
Lines = Words.Split('~');
if (Lines[6] == SelectedMapel)
{
rTextBoxSoal.Text += Words + "~" + i.ToString() + "\n";
}
}
LoadToListviewMapel();
}
private void SaveDataSoal()
{
string FileName="Data\\Data.bin";
string Soal = rTextBoxBankSoal.Text;
BinaryWriter Writer = new BinaryWriter(File.Open(FileName, FileMode.Create));
Writer.Write(Soal);
Writer.Close();
}
private void LoadDataSoal()
{
string FileName = "Data\\Data.bin";
string Soal;
BinaryReader Reader = new BinaryReader(File.Open(FileName, FileMode.Open));
Soal = Reader.ReadString();
Reader.Close();
rTextBoxBankSoal.Clear();
rTextBoxBankSoal.Text = Soal;
}
private void mComboBoxMapel_SelectedIndexChanged(object sender, EventArgs e)

```



```

{
MoveSoalToBuffer(mComboBoxMapel.Text);
}
private void rTextBoxBankSoal_TextChanged(object sender, EventArgs e)
{
MoveSoalToBuffer(mComboBoxMapel.Text);
}
private void mButtonUbahSoal_Click(object sender, EventArgs e)
{
List<string> myListSoal;
EditSoal = true;
string[] DataSoalToEdit = new string[8];
string Words;
string[] Lines;
int SoalToDel;
if (listViewSoal.SelectedItems.Count > 0)
{
int i = listViewSoal.SelectedIndices[0];
mMapelSoal = mComboBoxMapel.Text;
mIDSoal = listViewSoal.Items[i].SubItems[1].Text;
mBbtSoal = listViewSoal.Items[i].SubItems[2].Text;
mIKSoal = listViewSoal.Items[i].SubItems[3].Text;
mDBSoal = listViewSoal.Items[i].SubItems[4].Text;
mButirSoal = listViewSoal.Items[i].SubItems[5].Text;
mKunciJawaban = listViewSoal.Items[i].SubItems[6].Text;
if (Tambah_Soal.ShowDialog() == DialogResult.OK)
{
string[] DataSoal = new string[6];
string Mapel = mComboBoxMapel.Text;
Words = rTextBoxSoal.Lines[i];
Lines = Words.Split('~');
SoalToDel = Convert.ToInt32(Lines[7]);
myListSoal = rTextBoxBankSoal.Lines.ToList();
myListSoal.RemoveAt(SoalToDel);
rTextBoxBankSoal.Lines = myListSoal.ToArray();
DataSoal[0] = Tambah_Soal.IDSoal;
DataSoal[1] = Tambah_Soal.BbtSoal;
DataSoal[2] = Tambah_Soal.IKSoal;
DataSoal[3] = Tambah_Soal.DBSoal;
DataSoal[4] = Tambah_Soal.ButirSoal;
DataSoal[5] = Tambah_Soal.KunciJawaban;
rTextBoxBankSoal.Text +=
DataSoal[0] + "~" + DataSoal[1] + "~" +
DataSoal[2] + "~" + DataSoal[3] + "~" +
DataSoal[4] + "~" + DataSoal[5] + "~" +
Mapel + "\n";
SaveDataSoal();
MoveSoalToBuffer(mComboBoxMapel.Text);
}
}
}

```

```

}
}
EditSoal = false;
}
private void mButtonHapusSoal_Click(object sender, EventArgs e)
{
List<string> myListSoal;
string[] DataSoalToEdit = new string[8];
string Words;
string[] Lines;
int SoalToDel;
if (listViewSoal.SelectedItems.Count > 0)
{
int i = listViewSoal.SelectedIndices[0];
Words = richTextBoxSoal.Lines[i];
Lines = Words.Split('~');
SoalToDel = Convert.ToInt32(Lines[7]);
myListSoal = richTextBoxBankSoal.Lines.ToList();
myListSoal.RemoveAt(SoalToDel);
richTextBoxBankSoal.Lines = myListSoal.ToArray();
SaveDataSoal();
MoveSoalToBuffer(mComboBoxMapel.Text);
}
}
private void mComboBoxMapelGuru_DropDown(object sender, EventArgs e)
{
string[] DataMapel = new string[2];
string Words;
int hitung = 0;
mComboBoxIDMapelGuru.Items.Clear();
foreach (string Baris in richTextBoxMapel.Lines) { hitung++; }
for (int i = 0; i < hitung - 1; i++)
{
string[] Lines;
Words = richTextBoxMapel.Lines[i];
Lines = Words.Split('~');
DataMapel[0] = Lines[0];
DataMapel[1] = Lines[1];
mComboBoxIDMapelGuru.Items.Add(Lines[0]);
}
}
private void mButtonAddGuru_Click(object sender, EventArgs e)
{
int hitung = 0;
foreach (string Baris in richTextBoxGuru.Lines) { hitung++; }
if (hitung == 0) { mTextBoxIdGuru.Text = (hitung + 1).ToString(); }
if (hitung > 0) { mTextBoxIdGuru.Text = (hitung).ToString(); }
}
}

```

```

private void MoveUserToRTBoxGuru()
{
string StrCmd, Username, Password, Status, ID;
OleDbCommand Cmd;
OleDbDataReader DataReader;
rTextBoxGuru.Clear();
StrCmd = "SELECT * FROM tbUser";
Cmd = new OleDbCommand(StrCmd, MyConn);
MyConn.Open();
DataReader = Cmd.ExecuteReader();
if (DataReader.HasRows)
{
while (DataReader.Read())
{
Username = DataReader.GetString(0).ToLower();
Password = DataReader.GetString(1);
Status = DataReader.GetString(2);
ID = DataReader.GetString(3);
if (Status == "1")
{
rTextBoxGuru.Text += Username + "~" + Password + "~" + Status + "~" + ID
+ "\n";
}
}
}
MyConn.Close();
listViewGuru.Items.Clear();
string[] DataGuru = new string[4];
string Words;
ListItem DataGuruToWrite;
int hitung = 0;
foreach (string Baris in rTextBoxGuru.Lines) { hitung++; }
for (int i = 0; i < hitung - 1; i++)
{
string[] Lines;
Words = rTextBoxGuru.Lines[i];
Lines = Words.Split('~');
DataGuru[0] = Lines[3];
DataGuru[1] = Lines[0];
DataGuruToWrite = new ListItem(DataGuru);
listViewGuru.Items.Add(DataGuruToWrite);
}
}
private void mButtonAddNewGuru_Click(object sender, EventArgs e)
{
string StrCmd;
OleDbCommand Cmd;
OleDbDataReader DataReader;

```

```

if (!(string.IsNullOrEmpty(mTextBoxIdGuru.Text)) &&
!(string.IsNullOrEmpty(mTextBoxNamaGuru.Text)) &&
!(string.IsNullOrEmpty(mTextBoxPassGuru.Text)))
{
if (EditGuru == false)
{
StrCmd = "INSERT INTO tbUser ([User], [Password], [Status], [ID]) VALUES ("
+ mTextBoxNamaGuru.Text + ", " + mTextBoxPassGuru.Text + ", '1', "
+ mTextBoxIdGuru.Text + ")";
}
else
{
StrCmd = "UPDATE tbUser SET [Password]='" + mTextBoxPassGuru.Text + ",
[Status]='1', [ID]='" + mTextBoxIdGuru.Text + " WHERE [User]='" +
mTextBoxNamaGuru.Text + "'";
}
Cmd = new OleDbCommand(StrCmd, MyConn);
MyConn.Open();
try
{
DataReader = Cmd.ExecuteReader();
EditGuru = false;
mTextBoxIdGuru.Clear();
mTextBoxNamaGuru.Clear();
mTextBoxPassGuru.Clear();
}
catch { MessageBox.Show("Nama Sudah dipakai"); }
MyConn.Close();
}
else { MessageBox.Show("Data belum lengkap"); }
MoveUserToRTBoxGuru();
}
private void mButtonEditGuru_Click(object sender, EventArgs e)
{
EditGuru = true;
string IDGuru, NamaGuru;
if (listViewGuru.SelectedItems.Count > 0)
{
int i = listViewGuru.SelectedIndex[0];
IDGuru = listViewGuru.Items[i].SubItems[0].Text;
NamaGuru = listViewGuru.Items[i].SubItems[1].Text;
mTextBoxIdGuru.Text = IDGuru;
mTextBoxNamaGuru.Text = NamaGuru;
IDGuruOld = IDGuru;
NamaGuruOld = NamaGuru;
PointerGuruOld = i;
string[] Lines;
string Words = rTextBoxGuru.Lines[i];

```

```

Lines = Words.Split('~');
mTextBoxPassGuru.Text = Lines[1];
}
}
private void mButtonDelGuru_Click(object sender, EventArgs e)
{
string NamaGuru;
string StrCmd;
OleDbCommand Cmd;
OleDbDataReader DataReader;
if (listViewGuru.SelectedItems.Count > 0)
{
int i = listViewGuru.SelectedIndices[0];
NamaGuru = listViewGuru.Items[i].SubItems[1].Text;
StrCmd = "DELETE * FROM tbUser WHERE [User]='" + NamaGuru + "'";
Cmd = new OleDbCommand(StrCmd, MyConn);
MyConn.Open();
DataReader = Cmd.ExecuteReader();
MyConn.Close();
MoveUserToRTBoxGuru();
}
}
private void mButtonCancelAddGuru_Click(object sender, EventArgs e)
{
EditGuru = false;
mTextBoxIdGuru.Clear();
mTextBoxNamaGuru.Clear();
mTextBoxPassGuru.Clear();
}
private void mButtonAddSiswa_Click(object sender, EventArgs e)
{
int hitung = 0;
foreach (string Baris in rTextBoxSiswa.Lines) { hitung++; }
if (hitung == 0) { mTextBoxIdSiswa.Text = (hitung + 1).ToString(); }
if (hitung > 0) { mTextBoxIdSiswa.Text = (hitung).ToString(); }
}
private void MoveUserToRTBoxSiswa()
{
string StrCmd, Username, Password, Status, ID;
OleDbCommand Cmd;
OleDbDataReader DataReader;
rTextBoxSiswa.Clear();
StrCmd = "SELECT * FROM tbUser";
Cmd = new OleDbCommand(StrCmd, MyConn);
MyConn.Open();
DataReader = Cmd.ExecuteReader();
if (DataReader.HasRows)
{

```

```

while (DataReader.Read())
{
    Username = DataReader.GetString(0).ToLower();
    Password = DataReader.GetString(1);
    Status = DataReader.GetString(2);
    ID = DataReader.GetString(3);
    if (Status == "2")
    {
        rTextBoxSiswa.Text += Username + "~" + Password + "~" + Status + "~" + ID
        + "\n";
    }
}
MyConn.Close();
listViewSiswa.Items.Clear();
string[] DataSiswa = new string[4];
string Words;
ListViewItem DataSiswaToWrite;
int hitung = 0;
foreach (string Baris in rTextBoxSiswa.Lines) { hitung++; }
for (int i = 0; i < hitung - 1; i++)
{
    string[] Lines;
    Words = rTextBoxSiswa.Lines[i];
    Lines = Words.Split('~');
    DataSiswa[0] = Lines[3];
    DataSiswa[1] = Lines[0];
    DataSiswaToWrite = new ListViewItem(DataSiswa);
    listViewSiswa.Items.Add(DataSiswaToWrite);
}
private void mButtonAddNewSiswa_Click(object sender, EventArgs e)
{
    string StrCmd;
    OleDbCommand Cmd;
    OleDbDataReader DataReader;
    if (!(string.IsNullOrEmpty(mTextBoxIdSiswa.Text)) &&
        !(string.IsNullOrEmpty(mTextBoxNamaSiswa.Text)) &&
        !(string.IsNullOrEmpty(mTextBoxPassSiswa.Text)))
    {
        if (EditSiswa == false)
        {
            StrCmd = "INSERT INTO tbUser ([User], [Password], [Status], [ID]) VALUES ("
            + mTextBoxNamaSiswa.Text + ", '" + mTextBoxPassSiswa.Text + "', '2', '"
            + mTextBoxIdSiswa.Text + ")";
        }
        else
        {

```

```

StrCmd = "UPDATE tbUser SET [Password]=" + mTextBoxPassSiswa.Text + ",
[Status]='2', [ID]=" + mTextBoxIdSiswa.Text + " WHERE [User]=" +
mTextBoxNamaSiswa.Text + """;
}
Cmd = new OleDbCommand(StrCmd, MyConn);
MyConn.Open();
try
{
DataReader = Cmd.ExecuteReader();
EditSiswa = false;
mTextBoxIdSiswa.Clear();
mTextBoxNamaSiswa.Clear();
mTextBoxPassSiswa.Clear();
}
catch { MessageBox.Show("Nama Sudah dipakai"); }
MyConn.Close();
}
else { MessageBox.Show("Data belum lengkap"); }
MoveUserToRTBoxSiswa();
}
private void mButtonEditSiswa_Click(object sender, EventArgs e)
{
EditSiswa = true;
string IDSiswa, NamaSiswa;
if (listViewSiswa.SelectedItems.Count > 0)
{
int i = listViewSiswa.SelectedIndices[0];
IDSiswa = listViewSiswa.Items[i].SubItems[0].Text;
NamaSiswa = listViewSiswa.Items[i].SubItems[1].Text;
mTextBoxIdSiswa.Text = IDSiswa;
mTextBoxNamaSiswa.Text = NamaSiswa;
IDSiswaOld = IDSiswa;
NamaSiswaOld = NamaSiswa;
PointerSiswaOld = i;
string[] Lines;
string Words = rTextBoxSiswa.Lines[i];
Lines = Words.Split('~');
mTextBoxPassSiswa.Text = Lines[1];
}
}
private void mButtonDelSiswa_Click(object sender, EventArgs e)
{
string NamaSiswa;
string StrCmd;
OleDbCommand Cmd;
OleDbDataReader DataReader;
if (listViewSiswa.SelectedItems.Count > 0)
{

```

```

int i = listViewSiswa.SelectedIndices[0];
NamaSiswa = listViewSiswa.Items[i].SubItems[1].Text;
StrCmd = "DELETE * FROM tbUser WHERE [User]='" + NamaSiswa + "'";
Cmd = new OleDbCommand(StrCmd, MyConn);
MyConn.Open();
DataReader = Cmd.ExecuteReader();
MyConn.Close();
MoveUserToRTBoxSiswa();
}
}
private void mButtonCancelAddSiswa_Click(object sender, EventArgs e)
{
EditSiswa = false;
mTextBoxIdSiswa.Clear();
mTextBoxNamaSiswa.Clear();
mTextBoxPassSiswa.Clear();
}
private void mComboBoxSettingMapel_DropDown(object sender, EventArgs e)
{
string[] DataMapel = new string[2];
string Words;
int hitung = 0;
mComboBoxSettingMapel.Items.Clear();
foreach (string Baris in rTextBoxMapel.Lines) { hitung++; }
for (int i = 0; i < hitung - 1; i++)
{
string[] Lines;
Words = rTextBoxMapel.Lines[i];
Lines = Words.Split('~');
DataMapel[0] = Lines[0];
DataMapel[1] = Lines[1];
mComboBoxSettingMapel.Items.Add(Lines[1]);
}
}
private void mTextBoxSettingJumlahSoal_KeyPress(object sender,
KeyPressEventArgs e)
{
char KeyPress = e.KeyChar;
if (char.IsDigit(KeyPress) || e.KeyChar == Convert.ToChar(Keys.Back)) { }
else { e.Handled = true; }
}
private void mTextBoxSettingWaktu_KeyPress(object sender, KeyPressEventArgs
e)
{
char KeyPress = e.KeyChar;
if (char.IsDigit(KeyPress) || e.KeyChar == Convert.ToChar(Keys.Back)) { }
else { e.Handled = true; }
}
}

```



```

private void SaveDataSetting(string FileName, string Mapel, int WaktuTes, int
JumlahSoal)
{
BinaryWriter Writer = new BinaryWriter(File.Open(FileName, FileMode.Create));
Writer.Write(Mapel);
Writer.Write(WaktuTes);
Writer.Write(JumlahSoal);
Writer.Close();
}
private void LoadDataSetting()
{
string FileName, Mapel;
int WaktuTes, JumlahSoal;
FileName = "Data\\Setting.bin";
BinaryReader Reader = new BinaryReader(File.Open(FileName, FileMode.Open));
Mapel = Reader.ReadString();
WaktuTes = Reader.ReadInt32();
JumlahSoal = Reader.ReadInt32();
Reader.Close();
mComboBoxSettingMapel.Items.Add(Mapel);
mComboBoxSettingMapel.SelectedIndex = 0;
mTextBoxSettingWaktu.Text = WaktuTes.ToString();
mTextBoxSettingJumlahSoal.Text = JumlahSoal.ToString();
}
private void mButtonSaveSetting_Click(object sender, EventArgs e)
{
string FileName, Mapel;
int WaktuTes, JumlahSoal;
if (!(string.IsNullOrEmpty(mComboBoxSettingMapel.Text)) &&
!(string.IsNullOrEmpty(mTextBoxSettingWaktu.Text)) &&
!(string.IsNullOrEmpty(mTextBoxSettingJumlahSoal.Text)))
{
FileName = "Data\\Setting.bin";
Mapel = mComboBoxSettingMapel.Text;
WaktuTes = Convert.ToInt32(mTextBoxSettingWaktu.Text);
JumlahSoal = Convert.ToInt32(mTextBoxSettingJumlahSoal.Text);
SaveDataSetting(FileName, Mapel, WaktuTes, JumlahSoal);
MessageBox.Show("Setting baru disimpan");
}
else { MessageBox.Show("Setting belum lengkap"); }
LoadDataSetting();
}
private void MulaiTes()
{
LoadDataSetting();
LoadDataSoal();
MoveSoalToBuffer(mComboBoxSettingMapel.Text);
UserName = LogUserName;
}

```

```

int JmlSoalBuffer;
int JmlSoal = Convert.ToInt32(mTextBoxSettingJumlahSoal.Text);
int WktTes = Convert.ToInt32(mTextBoxSettingWaktu.Text);
string SoalToSend = "";
string[] PointSoal;
JmlSoalBuffer = rTextBoxSoal.Lines.Length - 1;
int SoalSulit, SoalSedang, SoalMudah;
SoalSulit = (20 * JmlSoal) / 100;
SoalSedang = (60 * JmlSoal) / 100;
SoalMudah = JmlSoal - (SoalSulit + SoalSedang);
rTextBoxIndekKesukaran.Clear();
decimal IK;
string Words;
int hitung = 0;
int JmlSoalKategori;
foreach (string Baris in rTextBoxSoal.Lines) { hitung++; }
for (int a = 0; a < hitung - 1; a++)
{
string currentDot1 =
System.Globalization.CultureInfo.CurrentCulture.NumberFormat.NumberDecimalS
eparator;
string[] Lines;
Words = rTextBoxSoal.Lines[a];
Lines = Words.Split('~');
IK = Convert.ToDecimal(Lines[2].Replace('.', Convert.ToChar(currentDot1)));
if (IK <= 0.3m) { rTextBoxIndekKesukaran.Text += Words + "\n"; }
}
JmlSoalKategori = rTextBoxIndekKesukaran.Lines.Length - 1;
PointSoal = rTextBoxIndekKesukaran.Lines;
List<int> RandNumber1 = Enumerable.Range(0, JmlSoalKategori).ToList();
RandNumber1.Shuffle();
int i = 0;
foreach (int Number in RandNumber1)
{
i++;
if (i <= SoalSulit) { SoalToSend += PointSoal[Number] + "\n"; }
}
rTextBoxIndekKesukaran.Clear();
hitung = 0;
foreach (string Baris in rTextBoxSoal.Lines) { hitung++; }
for (int a = 0; a < hitung - 1; a++)
{
string currentDot1 =
System.Globalization.CultureInfo.CurrentCulture.NumberFormat.NumberDecimalS
eparator;
string[] Lines;
Words = rTextBoxSoal.Lines[a];
Lines = Words.Split('~');

```

```

IK = Convert.ToDecimal(Lines[2].Replace('.', Convert.ToChar(currentDot1)));
if (IK > 0.3m && IK < 0.7m) { rTextBoxIndekKesukaran.Text += Words + "\n";
}
}
JmlSoalKategori = rTextBoxIndekKesukaran.Lines.Length - 1;
PointSoal = rTextBoxIndekKesukaran.Lines;
List<int> RandNumber2 = Enumerable.Range(0, JmlSoalKategori).ToList();
RandNumber2.Shuffle();
i = 0;
foreach (int Number in RandNumber2)
{
i++;
if (i <= SoalSedang) { SoalToSend += PointSoal[Number] + "\n"; }
}
rTextBoxIndekKesukaran.Clear();
hitung = 0;
foreach (string Baris in rTextBoxSoal.Lines) { hitung++; }
for (int a = 0; a < hitung - 1; a++)
{
string currentDot1 =
System.Globalization.CultureInfo.CurrentCulture.NumberFormat.NumberDecimalS
eparator;
string[] Lines;
Words = rTextBoxSoal.Lines[a];
Lines = Words.Split('~');
IK = Convert.ToDecimal(Lines[2].Replace('.', Convert.ToChar(currentDot1)));
if (IK > 0.7m) { rTextBoxIndekKesukaran.Text += Words + "\n"; }
}
JmlSoalKategori = rTextBoxIndekKesukaran.Lines.Length - 1;
PointSoal = rTextBoxIndekKesukaran.Lines;
List<int> RandNumber3 = Enumerable.Range(0, JmlSoalKategori).ToList();
RandNumber3.Shuffle();
i = 0;
foreach (int Number in RandNumber3)
{
i++;
if (i <= SoalMudah) { SoalToSend += PointSoal[Number] + "\n"; }
}
Soal = SoalToSend;
WaktuTes = WktTes;
if (Tes.ShowDialog() == DialogResult.OK)
{
float Nilai = (Convert.ToSingle(Tes.NilaiTes))/100;
MessageBox.Show("Nilai Anda : " + Nilai.ToString("N2") + "\nPredikat " +
KategoriHuruf(Nilai), "Selamat");
rTextBoxHasilTes.Text = Tes.HasilTes;
LoadExistDataTes();
rTextBoxStoreTes.Text += Tes.HasilTes;
}

```

```

SaveDataTes();
}
}
private bool CekBelumTes(string Mapel, string Nama)
{
bool BelumTes = true;
string[] DataHasilTes = new string[16];
string Words;
int hitung = 0;//sekaligus pointer data tes
foreach (string Baris in rTextBoxStoreTes.Lines) { hitung++; }
for (int i = 0; i < hitung - 1; i++)
{
string[] Lines;
Words = rTextBoxStoreTes.Lines[i];
Lines = Words.Split('~');
if (Lines[6] == Mapel && Lines[14] == Nama) { BelumTes = false; }
}
return BelumTes;
}
private void mButtonStartTest_Click(object sender, EventArgs e)
{
bool StatusTes = CekBelumTes(mComboBoxSettingMapel.Text, LogUserName);
if (StatusTes == true)
{
DialogResult SiapTes = MessageBox.Show("Apakan anda siap?", "Peringatan",
MessageBoxButtons.YesNo);
if (SiapTes == DialogResult.Yes) { MulaiTes(); }
}
else { MessageBox.Show("Anda sudah melaksanakan tes\nHubungi Guru jika
akan tes ulang", "Perhatian"); }
}
private void LoadExistDataTes()
{
string FileName = "Data\\DataTes.bin";
string DataTes = "";
try
{
BinaryReader Reader = new BinaryReader(File.Open(FileName, FileMode.Open));
DataTes = Reader.ReadString();
Reader.Close();
}
catch { }
rTextBoxStoreTes.Clear();
rTextBoxStoreTes.Text = DataTes;
}
private void SaveDataTes()
{
string FileName = "Data\\DataTes.bin";

```

```

string DataTes = rTextBoxStoreTes.Text;
BinaryWriter Writer = new BinaryWriter(File.Open(FileName, FileMode.Create));
Writer.Write(DataTes);
Writer.Close();
}
private void mComboBoxMapelUserLog_DropDown(object sender, EventArgs e)
{
string[] DataMapel = new string[2];
string Words;
int hitung = 0;
mComboBoxMapelUserLog.Items.Clear();
foreach (string Baris in rTextBoxMapel.Lines) { hitung++; }
for (int i = 0; i < hitung - 1; i++)
{
string[] Lines;
Words = rTextBoxMapel.Lines[i];
Lines = Words.Split('~');
DataMapel[0] = Lines[0];
DataMapel[1] = Lines[1];
mComboBoxMapelUserLog.Items.Add(Lines[1]);
}
}
private void LoadToListviewUserLog()
{
listViewUserLog.Items.Clear();
string[] DataHasilTes = new string[17];
string Words;
ListViewItem DataHasilTesToWrite;
int hitung = 0;
foreach (string Baris in rTextBoxNewUserLog.Lines) { hitung++; }
for (int i = 1; i < hitung; i++)
{
float Skor = 0, SkorMak = 0;
string[] Lines;
Words = rTextBoxNewUserLog.Lines[i];
Lines = Words.Split('~');
int SizeLines = Lines.Length;
for (int a = 1; a < SizeLines; a++)
{
if (IsOdd(a)) { Skor += Convert.ToSingle(Lines[a]); }
else { SkorMak += Convert.ToSingle(Lines[a]); }
}
DataHasilTes[1] = Lines[0];
DataHasilTes[2] = ((SizeLines - 1) / 2).ToString();
DataHasilTes[3] = SkorMak.ToString("N2");
DataHasilTes[4] = Skor.ToString("N2");
DataHasilTes[5] = (Skor * 10 / SkorMak).ToString("N2");
DataHasilTes[6] = (Skor * 100 / SkorMak).ToString("N2");
}
}

```

```

float NilaiPredikat = (Skor * 100) / SkorMak;
DataHasilTes[7] = KategoriHuruf(NilaiPredikat);
DataHasilTesToWrite = new ListViewItem(DataHasilTes);
listViewUserLog.Items.Add(DataHasilTesToWrite);
}
}
private void MoveDataTesToBuffer(string SelectedMapel)
{
string[] DataHasilTes = new string[16];
string Words;
int hitung = 0;//sekaligus pointer data tes
rTextBoxViewHasilTes.Clear();
foreach (string Baris in rTextBoxStoreTes.Lines) { hitung++; }
for (int i = 0; i < hitung - 1; i++)
{
string[] Lines;
Words = rTextBoxStoreTes.Lines[i];
Lines = Words.Split('~');
if (Lines[6] == SelectedMapel)
{
rTextBoxViewHasilTes.Text += Words + "~" + i.ToString() + "\n";
}
}
hitung = 0;
rTextBoxNewUserLog.Clear();
foreach (string Baris in rTextBoxViewHasilTes.Lines) { hitung++; }
string OldName = "";
for (int i = 0; i < hitung - 1; i++)
{
string[] Lines;
Words = rTextBoxViewHasilTes.Lines[i];
Lines = Words.Split('~');
float[] SkorSiswa = new float[] { Convert.ToSingle(Lines[9]),
Convert.ToSingle(Lines[10]), Convert.ToSingle(Lines[11]),
Convert.ToSingle(Lines[12]) };
float SkorAkhir = SkorSiswa.Max();
SkorAkhir = SkorAkhir * Convert.ToSingle(Lines[1]) / 100;
if (Lines[14] == OldName)
{
rTextBoxNewUserLog.Text += "~" + SkorAkhir.ToString("N2") + "~" + Lines[1];
}
else
{
rTextBoxNewUserLog.Text += "\n" + Lines[14] + "~" +
SkorAkhir.ToString("N2") + "~" + Lines[1];
}
OldName = Lines[14];
}
}

```

```

LoadToListviewUserLog();
}
private void mComboBoxMapelUserLog_SelectedIndexChanged(object sender,
EventArgs e)
{
MoveDataTesToBuffer(mComboBoxMapelUserLog.Text);
}
private void mButtonDelUserLog_Click(object sender, EventArgs e)
{
List<string> myListHasilTes;
string[] DataTesToEdit = new string[16];
string Words;
string[] Lines;
int DataTesToDel;
if (listViewUserLog.SelectedItems.Count > 0)
{
int i = listViewUserLog.SelectedIndices[0];
Words = rTextBoxViewHasilTes.Lines[i];
Lines = Words.Split('~');
DataTesToDel = Convert.ToInt32(Lines[17]);
myListHasilTes = rTextBoxStoreTes.Lines.ToList();
myListHasilTes.RemoveAt(DataTesToDel);
rTextBoxStoreTes.Lines = myListHasilTes.ToArray();
SaveDataTes();
MoveDataTesToBuffer(mComboBoxMapelUserLog.Text);
}
else { MessageBox.Show("Pilih data yang akan dihapus"); }
}
private void mButtonResetUserLog_Click(object sender, EventArgs e)
{
if (mComboBoxMapelUserLog.SelectedIndex >= 0)
{
string DataTesNotDel = "";
string Words;
int hitung = 0;//sekaligus pointer data tes
foreach (string Baris in rTextBoxStoreTes.Lines) { hitung++; }
for (int i = 0; i < hitung - 1; i++)
{
string[] Lines;
Words = rTextBoxStoreTes.Lines[i];
Lines = Words.Split('~');
if (Lines[6] != mComboBoxMapelUserLog.Text)
{
DataTesNotDel += Words + "\n";
}
}
}
rTextBoxStoreTes.Text = DataTesNotDel;
SaveDataTes();
}

```

```

MoveDataTesToBuffer(mComboBoxMapelUserLog.Text);
}
else { MessageBox.Show("Pilih Mata Pelajaran"); }
}
private void mComboBoxUserHistory_DropDown(object sender, EventArgs e)
{
string[] DataUser = new string[2];
string Words;
int hitung = 0;
mComboBoxUserHistory.Items.Clear();
foreach (string Baris in rTextBoxSiswa.Lines) { hitung++; }
for (int i = 0; i < hitung - 1; i++)
{
string[] Lines;
Words = rTextBoxSiswa.Lines[i];
Lines = Words.Split('~');
DataUser[0] = Lines[0];
DataUser[1] = Lines[1];
mComboBoxUserHistory.Items.Add(Lines[0]);
}
}
private void LoadToListviewHistory()
{
listViewHistory.Items.Clear();
string[] DataHistory = new string[17];
string Words;
ListViewItem DataHistoryToWrite;
int hitung = 0;
foreach (string Baris in rTextBoxHistory.Lines) { hitung++; }
for (int i = 0; i < hitung - 1; i++)
{
string[] Lines;
Words = rTextBoxHistory.Lines[i];
Lines = Words.Split('~');
string currentDot1 =
System.Globalization.CultureInfo.CurrentCulture.NumberFormat.NumberDecimalS
eparator;
decimal IK = Convert.ToDecimal(Lines[2].Replace('.',
Convert.ToChar(currentDot1)));
if (IK <= 0.3m) { DataHistory[4] = "Sulit"; }
else if (IK > 0.3m && IK < 0.7m) { DataHistory[4] = "Sedang"; }
else if (IK > 0.7m) { DataHistory[4] = "Mudah"; }
DataHistory[0] = Lines[6];
DataHistory[1] = Lines[0];
DataHistory[2] = Lines[1];
DataHistory[3] = Lines[13];
DataHistory[5] = Lines[4];
DataHistory[6] = Lines[8];
}
}
}
}

```



```

DataHistoryToWrite = new ListViewItem(DataHistory);
listViewHistory.Items.Add(DataHistoryToWrite);
}
}
private void MoveHistoryToBuffer(string SelectedUser)
{
string[] DataHistory = new string[16];
string Words;
int hitung = 0;//sekaligus pointer data tes
float JmlSkor = 0, JmlSkorMak = 0;
rTextBoxHistory.Clear();
foreach (string Baris in rTextBoxStoreTes.Lines) { hitung++; }
for (int i = 0; i < hitung - 1; i++)
{
string[] Lines;
Words = rTextBoxStoreTes.Lines[i];
Lines = Words.Split('~');
if (Lines[14] == SelectedUser)
{
rTextBoxHistory.Text += Words + "~" + i.ToString() + "\n";
float[] SkorSiswa = new float[] { Convert.ToSingle(Lines[9]),
Convert.ToSingle(Lines[10]), Convert.ToSingle(Lines[11]),
Convert.ToSingle(Lines[12]) };
float SkorAkhir = SkorSiswa.Max();
SkorAkhir = SkorAkhir * Convert.ToSingle(Lines[1]) / 100;
JmlSkor += SkorAkhir;
JmlSkorMak += Convert.ToSingle(Lines[1]);
}
}
LoadToListviewHistory();
mTextBoxSkorTotal.Text = JmlSkor.ToString("N2");
mTextBoxSkorMak.Text = JmlSkorMak.ToString();
mTextBoxNilai.Text = (JmlSkor * 10 / JmlSkorMak).ToString("N2");
mLabelNilai.Text = "0-10";
}
private void mComboBoxUserHistory_SelectedIndexChanged(object sender,
EventArgs e)
{
MoveHistoryToBuffer(mComboBoxUserHistory.Text);
}
private void mButtonDelHistory_Click(object sender, EventArgs e)
{
List<string> myListHistory;
string[] DataHistoryToEdit = new string[16];
string Words;
string[] Lines;
int DataHistoryToDel;
if (listViewHistory.SelectedItems.Count > 0)

```

```

{
int i = listViewHistory.SelectedIndices[0];
Words = rTextBoxHistory.Lines[i];
Lines = Words.Split('~');
DataHistoryToDel = Convert.ToInt32(Lines[17]);
myListHistory = rTextBoxStoreTes.Lines.ToList();
myListHistory.RemoveAt(DataHistoryToDel);
rTextBoxStoreTes.Lines = myListHistory.ToArray();
SaveDataTes();
MoveHistoryToBuffer(mComboBoxUserHistory.Text);
}
else { MessageBox.Show("Pilih data yang akan dihapus"); }
}
private void mButtonResetHistory_Click(object sender, EventArgs e)
{
if (mComboBoxUserHistory.SelectedIndex >= 0)
{
string DataHistoryNotDel = "";
string Words;
int hitung = 0;//sekaligus pointer data tes
foreach (string Baris in rTextBoxStoreTes.Lines) { hitung++; }
for (int i = 0; i < hitung - 1; i++)
{
string[] Lines;
Words = rTextBoxStoreTes.Lines[i];
Lines = Words.Split('~');
if (Lines[14] != mComboBoxUserHistory.Text)
{
DataHistoryNotDel += Words + "\n";
}
}
rTextBoxStoreTes.Text = DataHistoryNotDel;
SaveDataTes();
MoveHistoryToBuffer(mComboBoxUserHistory.Text);
}
else { MessageBox.Show("Pilih Nama User"); }
}
private void mComboBoxPrint_SelectedIndexChanged(object sender, EventArgs e)
{
MoveUserToRTBoxMapel();
MoveUserToRTBoxGuru();
MoveUserToRTBoxSiswa();
LoadDataSetting();
LoadDataSoal();
LoadExistDataTes();
if (mComboBoxPrint.Text == "Data Soal") { PrintSoal(); }
if (mComboBoxPrint.Text == "Data Mata Pelajaran") { PrintMapel(); }
}

```

```

if (mComboBoxPrint.Text == "Data User Log") { PrintUserLog(); }
if (mComboBoxPrint.Text == "Data History") { PrintHistory(); }
if (mComboBoxPrint.Text == "Data User Siswa") { PrintUserSiswa(); }
if (mComboBoxPrint.Text == "Data User Guru") { PrintUserGuru(); }
}
private void PrintSoal()
{
printableListViewPrint.Clear();
printableListViewPrint.CheckBoxes = true;
printableListViewPrint.Columns.Insert(0, "1", "", 30);
printableListViewPrint.Columns.Insert(1, "2", "ID Soal", 50);
printableListViewPrint.Columns.Insert(2, "3", "Bobot", 40);
printableListViewPrint.Columns.Insert(3, "4", "IK", 30);
printableListViewPrint.Columns.Insert(4, "5", "DB", 30);
printableListViewPrint.Columns.Insert(5, "6", "Butir Soal", 400);
printableListViewPrint.Columns.Insert(6, "7", "Kunci", 400);
printableListViewPrint.Items.Clear();
string[] DataSoal = new string[8];
string Words;
ListViewItem DataSoalToWrite;
int hitung = 0;
foreach (string Baris in rTextBoxBankSoal.Lines) { hitung++; }
for (int i = 0; i < hitung - 1; i++)
{
string[] Lines;
Words = rTextBoxBankSoal.Lines[i];
Lines = Words.Split('~');
DataSoal[1] = Lines[0];
DataSoal[2] = Lines[1];
DataSoal[3] = Lines[2];
DataSoal[4] = Lines[3];
DataSoal[5] = Lines[4];
DataSoal[6] = Lines[5];
DataSoalToWrite = new ListViewItem(DataSoal);
printableListViewPrint.Items.Add(DataSoalToWrite);
}
printableListViewPrint.Title = "Print Soal";
}
private void PrintMapel()
{
printableListViewPrint.Clear();
printableListViewPrint.CheckBoxes = false;
printableListViewPrint.Columns.Insert(0, "1", "ID", 100);
printableListViewPrint.Columns.Insert(1, "2", "Mata Pelajaran", 500);
printableListViewPrint.Items.Clear();
string[] DataMapel = new string[2];
string Words;
ListViewItem DataMapelToWrite;

```

```

int hitung = 0;
foreach (string Baris in rTextBoxMapel.Lines) { hitung++; }
for (int i = 0; i < hitung - 1; i++)
{
string[] Lines;
Words = rTextBoxMapel.Lines[i];
Lines = Words.Split('~');
DataMapel[0] = Lines[0];
DataMapel[1] = Lines[1];
DataMapelToWrite = new ListViewItem(DataMapel);
printableListViewPrint.Items.Add(DataMapelToWrite);
}
printableListViewPrint.Title = "Print Mata Pelajaran";
}
private void PrintUserLog()
{
printableListViewPrint.Clear();
printableListViewPrint.CheckBoxes = true;
printableListViewPrint.Columns.Insert(0, "1", "", 30);
printableListViewPrint.Columns.Insert(1, "2", "ID Siswa", 100);
printableListViewPrint.Columns.Insert(2, "3", "Soal", 365);
printableListViewPrint.Columns.Insert(3, "4", "Jawaban", 365);
printableListViewPrint.Columns.Insert(4, "5", "Nilai", 100);
printableListViewPrint.Items.Clear();
string[] DataHasilTes = new string[17];
string Words;
ListViewItem DataHasilTesToWrite;
int hitung = 0;
foreach (string Baris in rTextBoxStoreTes.Lines) { hitung++; }
for (int i = 0; i < hitung - 1; i++)
{
string[] Lines;
Words = rTextBoxStoreTes.Lines[i];
Lines = Words.Split('~');
DataHasilTes[1] = Lines[14];
DataHasilTes[2] = Lines[4];
DataHasilTes[3] = Lines[8];
DataHasilTes[4] = Lines[13];
DataHasilTesToWrite = new ListViewItem(DataHasilTes);
printableListViewPrint.Items.Add(DataHasilTesToWrite);
}
printableListViewPrint.Title = "Print User Log";
}
private void PrintHistory()
{
printableListViewPrint.Clear();
printableListViewPrint.CheckBoxes = false;
printableListViewPrint.Columns.Insert(0, "1", "Mata Pelajaran", 240);

```

```

printableListViewPrint.Columns.Insert(1, "2", "ID Soal", 240);
printableListViewPrint.Columns.Insert(2, "3", "Hasil", 240);
printableListViewPrint.Columns.Insert(3, "4", "Kategori", 240);
printableListViewPrint.Items.Clear();
string[] DataHistory = new string[17];
string Words;
ListViewItem DataHistoryToWrite;
int hitung = 0;
foreach (string Baris in rTextBoxStoreTes.Lines) { hitung++; }
for (int i = 0; i < hitung - 1; i++)
{
string[] Lines;
Words = rTextBoxStoreTes.Lines[i];
Lines = Words.Split('~');
DataHistory[0] = Lines[6];
DataHistory[1] = Lines[0];
DataHistory[2] = Lines[13];
DataHistory[3] = Lines[16];
DataHistoryToWrite = new ListViewItem(DataHistory);
printableListViewPrint.Items.Add(DataHistoryToWrite);
}
printableListViewPrint.Title = "Print History";
}
private void PrintUserSiswa()
{
printableListViewPrint.Clear();
printableListViewPrint.CheckBoxes = false;
printableListViewPrint.Columns.Insert(0, "1", "ID", 100);
printableListViewPrint.Columns.Insert(1, "2", "Nama", 500);
printableListViewPrint.Items.Clear();
string[] DataSiswa = new string[4];
string Words;
ListViewItem DataSiswaToWrite;
int hitung = 0;
foreach (string Baris in rTextBoxSiswa.Lines) { hitung++; }
for (int i = 0; i < hitung - 1; i++)
{
string[] Lines;
Words = rTextBoxSiswa.Lines[i];
Lines = Words.Split('~');
DataSiswa[0] = Lines[3];
DataSiswa[1] = Lines[0];
DataSiswaToWrite = new ListViewItem(DataSiswa);
printableListViewPrint.Items.Add(DataSiswaToWrite);
}
}
private void PrintUserGuru()
{

```

```

printableListViewPrint.Clear();
printableListViewPrint.CheckBoxes = false;
printableListViewPrint.Columns.Insert(0, "1", "ID", 100);
printableListViewPrint.Columns.Insert(1, "2", "Nama", 500);
printableListViewPrint.Items.Clear();
string[] DataGuru = new string[4];
string Words;
ListViewItem DataGuruToWrite;
int hitung = 0;
foreach (string Baris in rTextBoxGuru.Lines) { hitung++; }
for (int i = 0; i < hitung - 1; i++)
{
string[] Lines;
Words = rTextBoxGuru.Lines[i];
Lines = Words.Split('~');
DataGuru[0] = Lines[3];
DataGuru[1] = Lines[0];
DataGuruToWrite = new ListViewItem(DataGuru);
printableListViewPrint.Items.Add(DataGuruToWrite);
}
}
private void mButtonPrint_Click(object sender, EventArgs e)
{
if (mComboBoxPrint.SelectedIndex >= 0)
{
printableListViewPrint.FitToPage = checkBoxFitToPage.Checked;
printableListViewPrint.Print();
}
else { MessageBox.Show("Pilih Data yang akan di print"); }
}
private void mButtonPageSetup_Click(object sender, EventArgs e)
{
printableListViewPrint.PageSetup();
}
private void mButtonPrintPreview_Click(object sender, EventArgs e)
{
if (mComboBoxPrint.SelectedIndex >= 0)
{
printableListViewPrint.FitToPage = checkBoxFitToPage.Checked;
printableListViewPrint.PrintPreview();
}
else { MessageBox.Show("Pilih Data yang akan di print"); }
}
private void RefreshDataMapel(string OldMapel, string NewMapel)
{
int hitung = 0;
List<string> MyListMapel, MyListTes;
MyListMapel = rTextBoxBankSoal.Lines.ToList();

```

```

MyListTes = rTextBoxStoreTes.Lines.ToList();
foreach (string Baris in rTextBoxBankSoal.Lines) { hitung++; }
for (int i = 0; i < hitung - 1; i++)
{
MyListMapel[i] = MyListMapel[i].Replace(OldMapel, NewMapel);
rTextBoxBankSoal.Lines = MyListMapel.ToArray();
}
hitung = 0;
foreach (string Baris in rTextBoxStoreTes.Lines) { hitung++; }
for (int i = 0; i < hitung - 1; i++)
{
MyListTes[i] = MyListTes[i].Replace(OldMapel, NewMapel);
rTextBoxStoreTes.Lines = MyListTes.ToArray();
}
SaveDataSoal();
SaveDataTes();
}
private void RefreshDataSiswa(string OldName, string NewName)
{
int hitung = 0;
List<string> MyListSiswa;
MyListSiswa = rTextBoxStoreTes.Lines.ToList();
foreach (string Baris in rTextBoxStoreTes.Lines) { hitung++; }
for (int i = 0; i < hitung - 1; i++)
{
MyListSiswa[i] = MyListSiswa[i].Replace(OldName, NewName);
rTextBoxStoreTes.Lines = MyListSiswa.ToArray();
}
SaveDataTes();
}
private void metroComboBox1_SelectedIndexChanged(object sender, EventArgs e)
{
float JmlSkor = Convert.ToSingle(mTextBoxSkorTotal.Text), JmlSkorMak =
Convert.ToSingle(mTextBoxSkorMak.Text);
if (mComboBoxNilai.SelectedIndex == 0)
{
mTextBoxNilai.Text = (JmlSkor * 10 / JmlSkorMak).ToString("N2");
mLabelNilai.Text = "0-10";
}
if (mComboBoxNilai.SelectedIndex == 1)
{
mTextBoxNilai.Text = (JmlSkor * 100 / JmlSkorMak).ToString("N2");
mLabelNilai.Text = "0-100";
}
}
}

```

Lampiran II. Angket

1. Angket Siswa

PETUNJUK PENGISIAN ANGKET SISWA Unjuk Kerja *Software*

- Berilah tanda silang (X) pada kolom yang sesuai dengan pendapat Saudara
- Penjelasan alternatif jawaban:
1 = Tidak Setuju
2 = Kurang Setuju
3 = Setuju
4 = Sangat Setuju

Contoh

A. Aspek performansi penggunaan

| No | Pernyataan | Jawaban |
|----|--|---|
| 1. | Kelengkapan fasilitas yang mendukung operasional | ① ② <input checked="" type="checkbox"/> ③ ④ |

- Jika kolom pengisian angket terdapat kesalahan, berilah tanda (=) pada kolom yang Saudara jawab salah, selanjutnya berilah tanda silang (X) pada kolom yang sesuai dengan pendapat saudara

A. Aspek performansi penggunaan

| No | Pernyataan | Jawaban |
|----|--|---|
| 1. | Kelengkapan fasilitas yang mendukung operasional | ① ② <input checked="" type="checkbox"/> ③ <input checked="" type="checkbox"/> ④ |

| A. Aspek performansi penggunaan | | |
|--|--|----------------|
| No | Pernyataan | Jawaban |
| 1. | Kelengkapan fasilitas yang mendukung operasional | ① ② ③ ④ |
| 2. | Kelengkapan menu untuk mendukung operasional | ① ② ③ ④ |
| 3. | Pemakaian bahasa yang digunakan pada program | ① ② ③ ④ |
| 4. | Penggunaan navigasi/pengarah pada bagian yang penting | ① ② ③ ④ |
| 5. | Kemudahan menggunakan navigasi untuk link antar komponen | ① ② ③ ④ |
| B. Aspek performansi tampilan | | |
| 6. | Ketepatan ukuran teks yang digunakan | ① ② ③ ④ |
| 7. | Kejelasan ukuran teks untuk dibaca | ① ② ③ ④ |
| 8. | Ketepatan bentuk teks yang digunakan | ① ② ③ ④ |
| 9. | Ketepatan warna teks yang digunakan | ① ② ③ ④ |
| 10. | Ketepatan warna teks dengan warna <i>background</i> yang digunakan | ① ② ③ ④ |
| 11. | Kualitas gambar terhadap tampilan | ① ② ③ ④ |
| 12. | Komposisi warna gambar | ① ② ③ ④ |
| C. Aspek materi tes | | |
| 13. | Butir soal relevan dengan pokok bahasan yang diujikan | ① ② ③ ④ |
| 14. | Materi tes relevan dengan pokok bahasan | ① ② ③ ④ |
| 15. | Butir soal relevan dengan tuntutan kompetensi | ① ② ③ ④ |
| 16. | Butir soal relevan dengan pengujian aspek kognitif | ① ② ③ ④ |
| D. Aspek kemanfaatan | | |
| 17. | Penggunaan program dapat membantu untuk latihan ujian | ① ② ③ ④ |
| 18. | Penggunaan program dapat memberi motivasi untuk belajar | ① ② ③ ④ |
| 19. | Penggunaan program dapat mengetahui level kemampuan siswa mengerjakan soal | ① ② ③ ④ |
| 20. | Materi tes meningkatkan pemahaman siswa pada pokok bahasan | ① ② ③ ④ |
| 21. | Program membatu siswa dalam latihan mengerjakan soal | ① ② ③ ④ |
| E. Aspek Kelebihan Tes Essay | | |
| 22. | Tes <i>essay</i> dapat mengembangkan kemampuan berbahasa | ① ② ③ ④ |
| 23. | Tes <i>essay</i> dapat melatih kemampuan penalaran | ① ② ③ ④ |
| 24. | Tes <i>essay</i> dapat mengembangkan kemampuan pemecahan masalah | ① ② ③ ④ |
| 25. | Tes <i>essay</i> dapat mengembangkan tingkat kemampuan berpikir tinggi | ① ② ③ ④ |

F. Kesimpulan

26. Menurut saya program ini dinyatakan

- Layak digunakan tanpa revisi
- Layak digunakan dengan revisi
- Tidak layak

G. Komentar / saran

.....
.....

Yogyakarta,2014
Pengguna

(.....)

Beri tanda (√)

2. Angket Guru

PETUNJUK PENGISIAN ANGKET GURU Unjuk Kerja *Software*

4. Berilah tanda silang (X) pada kolom yang sesuai dengan pendapat Saudara
5. Penjelasan alternatif jawaban:
1 = Tidak Setuju
2 = Kurang Setuju
3 = Setuju
4 = Sangat Setuju

Contoh

B. Aspek performansi penggunaan

| No | Pernyataan | Jawaban |
|----|--|---|
| 1. | Kelengkapan fasilitas yang mendukung operasional | ① ② <input checked="" type="checkbox"/> ④ |

6. Jika kolom pengisian angket terdapat kesalahan, berilah tanda (=) pada kolom yang Saudara jawab salah, selanjutnya berilah tanda silang (X) pada kolom yang sesuai dengan pendapat saudara

B. Aspek performansi penggunaan

| No | Pernyataan | Jawaban |
|----|--|---|
| 1. | Kelengkapan fasilitas yang mendukung operasional | ① ② <input checked="" type="checkbox"/> <input checked="" type="checkbox"/> |

| A. Aspek performansi penggunaan | | |
|--|--|----------------|
| No | Pernyataan | Jawaban |
| 27. | Kelengkapan fasilitas yang mendukung operasional | ① ② ③ ④ |
| 28. | Kelengkapan menu untuk mendukung operasional | ① ② ③ ④ |
| 29. | Pemakaian bahasa yang digunakan pada program | ① ② ③ ④ |
| 30. | Penggunaan navigasi/pengarah pada bagian yang penting | ① ② ③ ④ |
| 31. | Kemudahan menggunakan navigasi untuk link antar komponen | ① ② ③ ④ |
| B. Aspek performansi tampilan | | |
| 32. | Ketepatan ukuran teks yang digunakan | ① ② ③ ④ |
| 33. | Kejelasan ukuran teks untuk dibaca | ① ② ③ ④ |
| 34. | Ketepatan bentuk teks yang digunakan | ① ② ③ ④ |
| 35. | Ketepatan warna teks yang digunakan | ① ② ③ ④ |
| 36. | Ketepatan warna teks dengan warna <i>background</i> yang digunakan | ① ② ③ ④ |
| 37. | Kualitas gambar terhadap tampilan | ① ② ③ ④ |
| 38. | Komposisi warna gambar | ① ② ③ ④ |
| C. Aspek materi tes | | |
| 39. | Butir soal relevan dengan pokok bahasan yang diujikan | ① ② ③ ④ |
| 40. | Materi tes relevan dengan pokok bahasan | ① ② ③ ④ |
| 41. | Butir soal relevan dengan tuntutan kompetensi | ① ② ③ ④ |
| 42. | Butir soal relevan dengan pengujian aspek kognitif | ① ② ③ ④ |
| D. Aspek kemanfaatan | | |
| 43. | Penggunaan program dapat membantu untuk proses pengujian | ① ② ③ ④ |
| 44. | Penggunaan program dapat memotivasi siswa untuk belajar | ① ② ③ ④ |
| 45. | Program mempermudah guru melakukan penilaian | ① ② ③ ④ |
| 46. | Program membantu guru dalam pembuatan bank soal | ① ② ③ ④ |
| 47. | Program membantu siswa dalam latihan soal | ① ② ③ ④ |
| E. Aspek Kelebihan Tes Essay | | |
| 48. | Tes <i>essay</i> dapat mengukur aspek kognitif tinggi | ① ② ③ ④ |
| 49. | Tes <i>essay</i> dapat mengembangkan kemampuan berbahasa | ① ② ③ ④ |
| 50. | Tes <i>essay</i> dapat melatih kemampuan penalaran | ① ② ③ ④ |
| 51. | Tes <i>essay</i> dapat mengembangkan tingkat kemampuan pemecahan masalah | ① ② ③ ④ |
| 52. | Tes <i>essay</i> dapat mengembangkan tingkat kemampuan berpikir tinggi | ① ② ③ ④ |
| 53. | Tes <i>essay</i> mempermudah guru dalam pembuatan soal | ① ② ③ ④ |

F. Kesimpulan

54. Menurut saya program ini dinyatakan

- Layak digunakan tanpa revisi
- Layak digunakan dengan revisi
- Tidak layak

G. Komentar / saran

.....
.....

Yogyakarta,2014
Pengguna

(.....)

Beri tanda (√)

Lampiran III. Checklist pengujian produk

Petunjuk Pengisian Lembar Pengujian Fungsional Produk Oleh Ahli Perangkat Lunak

7. Berilah tanda silang (X) pada kolom yang sesuai dengan pendapat Saudara
8. Penjelasan alternatif jawaban:
1 = Ya
2 = Tidak

Contoh

| No | Pernyataan | Hasil |
|----|---|--|
| 1. | Program dapat dijalankan di lingkungan <i>windows</i> | <input checked="" type="checkbox"/> <input type="checkbox"/> |

9. Jika kolom pengisian angket terdapat kesalahan, berilah tanda (=) pada kolom yang Saudara jawab salah, selanjutnya berilah tanda silang (X) pada kolom yang sesuai dengan pendapat saudara

| No | Pernyataan | Hasil |
|----|---|--|
| 1. | Program dapat dijalankan di lingkungan <i>windows</i> | <input type="checkbox"/> <input checked="" type="checkbox"/> |

Lembar Pengujian Fungsional Produk

| No | Pernyataan | Hasil |
|-----|--|-------|
| 1. | Program dapat dijalankan di lingkungan <i>windows</i> | ① ② |
| 2. | User dapat <i>login</i> sesuai dengan kondisinya | ① ② |
| 3. | User guru dapat membuka program sesuai kewenangannya | ① ② |
| 4. | User siswa dapat membuka program sesuai kewenangannya | ① ② |
| 5. | User dapat melakukan <i>logout</i> sesuai dengan kondisinya | ① ② |
| 6. | Program dapat ditutup kembali melalui tombol yang disediakan | ① ② |
| 7. | Program dapat menampilkan hasil kerja sesuai dengan menu yang dipilih | ① ② |
| 8. | Program dapat menyimpan hasil kerja sesuai dengan masukan yang diberikan | ① ② |
| 9. | Program dapat memproses masukan menjadi keluaran sesuai dengan fungsi yang dikehendaki | ① ② |
| 10. | Hasil kerja yang ditampilkan program adalah benar sesuai fungsi yang dikehendaki | ① ② |
| 11. | Data yang disimpan program adalah benar sesuai dengan fungsinya | ① ② |
| 12. | Keluaran hasil program adalah benar sesuai dengan fungsinya | ① ② |
| 13. | Program dapat membaca masukan berupa jawaban soal untuk proses selanjutnya | ① ② |
| 14. | Program dapat menyimpan hasil tes yang telah dikerjakan siswa | ① ② |
| 15. | Program dapat menampilkan hasil koreksi jawaban tes | ① ② |
| 16. | Program dapat menampilkan butir soal secara acak | ① ② |
| 17. | Program dapat membatasi waktu maksimal tes <i>essay</i> | ① ② |
| 18. | Program dapat menampilkan nilai akhir dari jawaban soal | ① ② |
| 19. | Program dapat menampilkan soal selanjutnya walaupun soal sebelumnya belum dijawab | ① ② |
| 20. | Program dapat menampilkan kembali soal yang sebelumnya belum dijawab | ① ② |
| 21. | Program dapat menampilkan kembali hasil tes yang telah dikoreksi | ① ② |
| 22. | Hasil koreksi jawaban soal adalah benar sesuai dengan koreksi yang dilakukan secara manual | ① ② |
| 23. | Hasil tes yang ditampilkan dapat menggambarkan kemampuan siswa peserta tes | ① ② |
| 24. | Hasil analisis program terhadap butir soal mampu mengestimasi kemampuan siswa peserta tes | ① ② |

Kesimpulan

Menurut saya program ini dinyatakan

- Layak digunakan tanpa revisi
- Layak digunakan dengan revisi
- Tidak layak

Komentar / saran

.....
.....

Yogyakarta, ... Desember 2014

Ahli Perangkat Lunak

(.....)

Beri tanda (√)

Lampiran IV. Soal tes dan kunci jawaban

Soal

1. Sebuah lampu akan menyala bila mendapat tegangan 1,8 Volt DC. Bila sumber tegangan adalah 5 Volt DC maka lampu dirangkai seri dengan resistor 330 Ohm. Jelaskan fungsi resistor dalam rangkaian tersebut!
2. Jelaskan pengertian potensiometer!
3. Jelaskan fungsi komponen Trafo Step Down!
4. Jelaskan fungsi komponen Trafo Step Up!
5. Komponen elektronik apakah yang berfungsi untuk menurunkan tegangan AC!
6. Komponen elektronik apakah yang berfungsi untuk menaikkan tegangan AC!
7. Jelaskan pengertian varco!
8. Jelaskan pengertian LDR
9. Komponen elektronik apakah yang berfungsi untuk menyimpan dan melepaskan muatan listrik!
10. Komponen elektronik apakah yang berfungsi untuk menahan arus listrik!
11. Pada pembuatan power supply DC 12 Volt dibutuhkan komponen Dioda rectifier/ Dioda Penyearah. Jelaskan fungsi komponen tersebut dalam rangkaian power supply!
12. Transistor dapat ditemukan pada rangkaian untuk menghidupkan led dan pada rangkaian amplifier. Jelaskan apa saja fungsi dari transistor dari contoh rangkaian tersebut!
13. Sebutkan nama kaki-kaki (elektroda) pada transistor bipolar!
14. Sebutkan nama kaki-kaki (elektroda) pada FET!
15. Sebutkan tipe-tipe diode!
16. Sebutkan macam bahan semikonduktor dalam pembuatan dioda!
17. Dioda tipe apakah yang nilai tahanan reverse bias turun karena pengaruh cahaya!
18. Transistor tipe apakah yang berkerja karena pengaruh cahaya!
19. Pada badan dioda tertulis 4 A/ 1500 V, artinya:
20. Jelaskan fungsi komponen Kapasitor!
21. Jelaskan fungsi zener dioda dalam rangkaian elektronik!
22. Jelaskan fungsi komponen Induktor!
23. Dioda tipe apakah yang dapat mengubah listrik AC menjadi listrik DC!
24. Dioda tipe apakah yang bisa mengalirkan listrik kearah berlawanan jika tegangan melampaui batas (breakdown voltage)!
25. Komponen elektronik apakah yang berfungsi untuk menyimpan arus listrik dalam bentuk magnet!

Kunci Jawaban

1. Menahan arus listrik, Menurunkan tegangan
2. resistor yang nilai hambatannya dapat diubah
3. Menurunkan tegangan AC
4. Menaikkan tegangan AC
5. Trafo step down
6. Trafo step up

7. kapasitor yang nilainya dapat diubah
8. resistor yang hambatannya berubah karena pengaruh cahaya
9. Kapasitor
10. Resistor
11. Mengubah tegangan bolak balik AC menjadi tegangan DC
12. Penguat arus , penguat tegangan , penguat daya, saklar
13. Kolektor, basis, emitor
14. Gate, drain, source
15. LED, foto dioda, dioda varactor, dioda penyearah, dioda zener
16. Germanium, silikon
17. Photo dioda
18. Photo transistor
19. arus maksimum 4 A tegangan maksimum 1500 V
20. menyimpan dan melepaskan muatan listrik
21. Menstabilkan arus dan tegangan
22. Menyimpan arus listrik dalam bentuk medan magnet
23. Diode penyearah
24. Zener dioda
25. Induktor

Lampiran V. Surat Keterangan Expert Judgement

SURAT PERNYATAAN *Judgment* INSTRUMEN PENELITIAN TUGAS AKHIR SKRIPSI

Saya yang bertanda tangan dibawah ini:

Nama : Dr. Samsul Hadi, M.Pd, M.T
NIP : 19600529 198403 1 003
Jurusan : Pendidikan Teknik Elektro

menyatakan bahwa instrumen penelitian TAS atas nama mahasiswa:

Nama : Fajar Prasty
NIM : 10518241003
Program Studi : Pendidikan Teknik Mekatronika
Judul TAS : PENGEMBANGAN MODEL TES ESSAY BERBANTUAN
KOMPUTER MATA PELAJARAN ELEKTRONIKA DASAR DI
SMK N 2 YOGYAKARTA

Setelah dilakukan kajian atas instrumen penelitian TAS tersebut dapat dinyatakan:

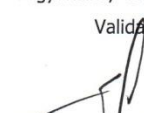
- Layak digunakan untuk penelitian
 Layak digunakan dengan perbaikan
 Tidak layak digunakan untuk penelitian yang bersangkutan

dengan saran/perbaikan sebagaimana terlampir.

Demikian agar dapat digunakan sebagaimana mestinya.

Yogyakarta, Desember 2014

Validator,


Dr. Samsul Hadi, M.Pd, M.T
NIP. 19600529 198403 1 003

Catatan:

- Beri tanda ✓

SURAT PERNYATAAN *Judgment*
INSTRUMEN PENELITIAN TUGAS AKHIR SKRIPSI

Saya yang bertanda tangan dibawah ini:

Nama : Dr. Istanto Wahyu Djatmiko, M.Pd
NIP : 19590219 198603 1 001
Jurusan : Pendidikan Teknik Elektro

menyatakan bahwa instrumen penelitian TAS atas nama mahasiswa:

Nama : Fajar Prastya
NIM : 10518241003
Program Studi : Pendidikan Teknik Mekatronika
Judul TAS : PENGEMBANGAN MODEL TES ESSAY BERBANTUAN
KOMPUTER MATA PELAJARAN ELEKTRONIKA DASAR DI
SMK N 2 YOGYAKARTA

Setelah dilakukan kajian atas instrumen penelitian TAS tersebut dapat dinyatakan:

- Layak digunakan untuk penelitian
 Layak digunakan dengan perbaikan
 Tidak layak digunakan untuk penelitian yang bersangkutan

dengan saran/perbaikan sebagaimana terlampir.

Demikian agar dapat digunakan sebagaimana mestinya.

Yogyakarta, Desember 2014

Validator,



Dr. Istanto Wahyu Djatmiko, M.Pd

NIP. 19590219 198603 1 001

Catatan:

- Beri tanda ✓

SURAT PERNYATAAN *Judgment*
INSTRUMEN PENELITIAN TUGAS AKHIR SKRIPSI

Saya yang bertanda tangan dibawah ini:

Nama : Rustam Asnawi, M.T, Ph.D
NIP : 19720127 199702 1 001
Jurusan : Pendidikan Teknik Elektro

menyatakan bahwa instrumen penelitian TAS atas nama mahasiswa:

Nama : Fajar Prastya
NIM : 10518241003
Program Studi : Pendidikan Teknik Mekatronika
Judul TAS : PENGEMBANGAN MODEL TES ESSAY BERBANTUAN
KOMPUTER MATA PELAJARAN ELEKTRONIKA DASAR DI
SMK N 2 YOGYAKARTA

Setelah dilakukan kajian atas instrumen penelitian TAS tersebut dapat dinyatakan:

- Layak digunakan untuk penelitian
 Layak digunakan dengan perbaikan
 Tidak layak digunakan untuk penelitian yang bersangkutan

dengan saran/perbaikan sebagaimana terlampir.

Demikian agar dapat digunakan sebagaimana mestinya.

Yogyakarta, Desember 2014

Validator,



Rustam Asnawi, M.T, Ph.D

NIP. 19720127 199702 1 001

Catatan:

- Beri tanda ✓

Lampiran VI. Surat Izin Penelitian



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

Nomor : 3273/H34/PL/2014

28 Nopember 2014

Lamp. :

Hal : Ijin Penelitian

Yth.

- 1 . Gubernur DIY c.q. Ka. Biro Adm. Pembangunan Setda DIY
- 2 . Gubernur Provinsi DIY c.q. Ka. Bappeda Provinsi DIY
- 3 . Walikota Kota Yogyakarta c.q. Kepala Badan Pelayanan Terpadu Kota Yogyakarta
- 4 . Kepala Dinas Pendidikan, Pemuda , dan Olahraga Provinsi DIY
- 5 . Kepala Dinas Pendidikan, Pemuda , dan Olahraga Kota Yogyakarta
- 6 . Kepala SMK N 2 Yogyakarta


Dalam rangka pelaksanaan Tugas Akhir Skripsi kami mohon dengan hormat bantuan Saudara memberikan ijin untuk melaksanakan penelitian dengan judul Pengembangan Model Tes Essay Berbantuan Komputer Mata Pelajaran Elektronika Dasar di SMK N 2 Yogyakarta, bagi mahasiswa Fakultas Teknik Universitas Negeri Yogyakarta tersebut di bawah ini:

| No. | Nama | NIM | Jurusan | Lokasi |
|-----|---------------|-------------|------------------------------------|--------------------|
| 1 | Fajar Prastya | 10518241003 | Pendidikan Teknik Mekatronika - S1 | SMK N 2 Yogyakarta |

Dosen Pembimbing/Dosen Pengampu :
Nama : Dr. Haryanto, M.Pd, MT
NIP : 19620310 198601 1 001

Adapun pelaksanaan penelitian dilakukan mulai Tanggal 8 Desember 2014 s/d 31 Januari 2015.

Demikian permohonan ini, atas bantuan dan kerjasama yang baik selama ini, kami mengucapkan terima kasih.

Wakil Dekan I

Dr. Sunaryo Soenarto
NIP. 19580630 198601 1 001

Tembusan :
Ketua Jurusan



PEMERINTAH DAERAH DAERAH ISTIMEWA YOGYAKARTA
SEKRETARIAT DAERAH
Kompleks Kepatihan, Danurejan, Telepon (0274) 562811 - 562814
(Hunting)
YOGYAKARTA 55213

openator2@yahoo.com

SURAT KETERANGAN / IJIN

070/REG/13/12/2014

Membaca Surat : **WAKIL DEKAN I FAKULTAS TEKNIK** Nomor : **3273/H34/PL/2014**
Tanggal : **28 NOVEMBER 2014** Perihal : **IJIN PENELITIAN/RISET**

- Mengingat :
1. Peraturan Pemerintah Nomor 41 Tahun 2006, tentang Perizinan bagi Perguruan Tinggi Asing, Lembaga Penelitian dan Pengembangan Asing, Badan Usaha Asing dan Orang Asing dalam melakukan Kegiatan Penelitian dan Pengembangan di Indonesia;
 2. Peraturan Menteri Dalam Negeri Nomor 20 Tahun 2011, tentang Pedoman Penelitian dan Pengembangan di Lingkungan Kementerian Dalam Negeri dan Pemerintah Daerah;
 3. Peraturan Gubernur Daerah Istimewa Yogyakarta Nomor 37 Tahun 2008, tentang Rincian Tugas dan Fungsi Satuan Organisasi di Lingkungan Sekretariat Daerah dan Sekretariat Dewan Perwakilan Rakyat Daerah;
 4. Peraturan Gubernur Daerah Istimewa Yogyakarta Nomor 18 Tahun 2009 tentang Pedoman Pelayanan Perizinan, Rekomendasi Pelaksanaan Survei, Penelitian, Pendataan, Pengembangan, Pengkajian, dan Studi Lapangan di Daerah Istimewa Yogyakarta.

DIIJINKAN untuk melakukan kegiatan survei/penelitian/pendataan/pengembangan/pengkajian/studi lapangan kepada:

Nama : **FAJAR PRASTYA** NIP/NIM : **10518241003**
Alamat : **FAKULTAS TEKNIK, PENDIDIKAN TEKNIK MEKATRONIKA, UNIVERSITAS NEGERI YOGYAKARTA**
Judul : **PENGEMBANGAN MODEL TES ESSAY BERBANTUAN KOMPUTER UNTUK MATA PELAJARAN ELEKTRONIKA DASAR DI SMK N 2 YOGYAKARTA**
Lokasi : **DINAS PENDIDIKAN, PEMUDA DAN OLAHRAGA DIY**
Waktu : **1 DESEMBER 2014 s/d 1 MARET 2015**

Dengan Ketentuan

1. Menyerahkan surat keterangan/ijin survei/penelitian/pendataan/pengembangan/pengkajian/studi lapangan *) dari Pemerintah Daerah DIY kepada Bupati/Walikota melalui institusi yang berwenang mengeluarkan ijin dimaksud;
2. Menyerahkan soft copy hasil penelitiannya baik kepada Gubernur Daerah Istimewa Yogyakarta melalui Biro Administrasi Pembangunan Setda DIY dalam compact disk (CD) maupun mengunggah (upload) melalui website adbang.jogjaprovo.go.id dan menunjukkan cetakan asli yang sudah disahkan dan dibubuhi cap institusi;
3. Ijin ini hanya dipergunakan untuk keperluan ilmiah, dan pemegang ijin wajib mentaati ketentuan yang berlaku di lokasi kegiatan;
4. Ijin penelitian dapat diperpanjang maksimal 2 (dua) kali dengan menunjukkan surat ini kembali sebelum berakhir waktunya setelah mengajukan perpanjangan melalui website adbang.jogjaprovo.go.id;
5. Ijin yang diberikan dapat dibatalkan sewaktu-waktu apabila pemegang ijin ini tidak memenuhi ketentuan yang berlaku.

Dikeluarkan di Yogyakarta
Pada tanggal **1 DESEMBER 2014**

A.n Sekretaris Daerah
Asisten Perekonomian dan Pembangunan
Ub.
Kepala Biro Administrasi Pembangunan



Desi Pili Astuti, M.Si
NIP. 19590525 198503 2 006

Tembusan :

1. GUBERNUR DAERAH ISTIMEWA YOGYAKARTA (SEBAGAI LAPORAN)
2. WALIKOTA YOGYAKARTA C.Q DINAS PERIJINAN KOTA YOGYAKARTA
3. DINAS PENDIDIKAN, PEMUDA DAN OLAHRAGA DIY
4. WAKIL DEKAN I FAKULTAS TEKNIK, UNIVERSITAS NEGERI YOGYAKARTA
5. YANG BERSANGKUTAN



PEMERINTAH KOTA YOGYAKARTA
DINAS PERIZINAN

Jl. Kenari No. 56 Yogyakarta Kode Pos : 55165 Telp. (0274) 555241, 515865, 515866, 562682
Fax (0274) 555241
EMAIL : perizinan@jogjakota.go.id
HOT LINE SMS : 081227625000 HOT LINE EMAIL : upik@jogjakota.go.id
WEBSITE : www.perizinan.jogjakota.go.id

SURAT IZIN

NOMOR : 070/3536
7278/34

Dasar : Surat izin / Rekomendasi dari Gubernur Kepala Daerah Istimewa Yogyakarta
Nomor : 070/REG/I/13/12/2014 Tanggal : 01/12/2014

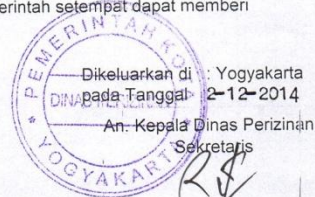
Mengingat : 1. Peraturan Daerah Kota Yogyakarta Nomor 10 Tahun 2008 tentang Pembentukan, -Susunan, Kedudukan dan Tugas Pokok Dinas Daerah
2. Peraturan Walikota Yogyakarta Nomor 85 Tahun 2008 tentang Fungsi, Rincian Tugas Dinas Perizinan Kota Yogyakarta;
3. Peraturan Walikota Yogyakarta Nomor 29 Tahun 2007 tentang Pemberian Izin Penelitian, Praktek Kerja Lapangan dan Kuliah Kerja Nyata di Wilayah Kota Yogyakarta;
4. Peraturan Walikota Yogyakarta Nomor 18 Tahun 2011 tentang Penyelenggaraan Perizinan pada Pemerintah Kota Yogyakarta;
5. Peraturan Gubernur Daerah Istimewa Yogyakarta Nomor: 18 Tahun 2009 tentang Pedoman Pelayanan Perizinan, Rekomendasi Pelaksanaan Survei, Penelitian, Pendataan, Pengembangan, Pengembangan, Pengkajian dan Studi Lapangan di Daerah Istimewa Yogyakarta;

Dijijinkan Kepada : Nama : FAJAR PRASTYA NO MHS / NIM : 10518241003
Pekerjaan : Mahasiswa Fak. Teknik - UNY
Alamat : Kampus Karangmalang, Yogyakarta
Penanggungjawab : Dr. Haryanto, M.Pd., M.T.
Keperluan : Melakukan Penelitian dengan judul Proposal : PENGEMBANGAN MODEL TES ESSAY BERBANTUAN KOMPUTER UNTUK MATA PELAJARAN ELEKTRONIKA DASAR DI SMK N 2 YOGYAKARTA

Lokasi/Responden : Kota Yogyakarta
Waktu : 01/12/2014 Sampai 01/03/2015
Lampiran : Proposal dan Daftar Pertanyaan
Dengan Ketentuan : 1. Wajib Memberi Laporan hasil Penelitian berupa CD kepada Walikota Yogyakarta (Cq. Dinas Perizinan Kota Yogyakarta)
2. Wajib Menjaga Tata tertib dan mentaati ketentuan-ketentuan yang berlaku setempat
3. Izin ini tidak disalahgunakan untuk tujuan tertentu yang dapat mengganggu kestabilan Pemerintah dan hanya diperlukan untuk keperluan ilmiah
4. Surat izin ini sewaktu-waktu dapat dibatalkan apabila tidak dipenuhinya ketentuan -ketentuan tersebut diatas
Kemudian diharap para Pejabat Pemerintah setempat dapat memberi bantuan seperlunya

Tanda tangan
Pemegang Izin

FAJAR PRASTYA



Dikeluarkan di : Yogyakarta

pada Tanggal : 2-12-2014

An. Kepala Dinas Perizinan

Sekretaris

ENY RETNO WATI, SH
NIP. 196103051988032004

Tembusan Kepada :
Yth. 1. Walikota Yogyakarta (sebagai laporan)
2. Ka. Biro Administrasi Pembangunan Setda DIY
3. Ka. Dinas Pendidikan Kota Yogyakarta
4. Kepala SMK Negeri 2 Yogyakarta
5. Ybs.



PEMERINTAH KOTA YOGYAKARTA
DINAS PENDIDIKAN
SMK NEGERI 2

JL. AM. Sangaji 47 Telp. (0274) 513490 Fax. (0274) 512639
E-mail : info@smk2-yk.sch.id Website : www.smk2-yk.sch.id,
Yogyakarta 55233

SURAT KETERANGAN

No. : 070/0252

Kepala SMK Negeri 2 Yogyakarta menerangkan bahwa :

Nama : **FAJAR PRASTYA**
No. Mahasiswa : 10518241003
Pekerjaan : Mahasiswa Fak. Teknik UNY

Berdasarkan surat izin dari Dinas Perizinan Kota Yogyakarta Nomor :
070/3536 tanggal 2 Desember 2014 perihal Permohonan Izin
Penelitian, bahwa mahasiswa tersebut telah selesai melaksanakan
pengambilan data pada tanggal 1 Desember 2014 sampai 1 Maret 2015
dengan judul :

**“ PENGEMBANGAN MODEL TES ESSAY BERBANTUAN KOMPUTER
UNTUK MATA PELAJARAN ELEKTRONIKA DASAR DI SMK NEGERI 2
YOGYAKARTA “**

Demikian surat keterangan ini dibuat untuk digunakan sebagaimana
mestinya.

Yogyakarta, 16 Februari 2015
Kepala Sekolah



Drs. RARYOTO, MT, M.Pd

NIP. 19641214 199003 1 007



SEGORO AMARTO

SEMANGAT GOTONG ROYONG AGAWA MAJUNE NGAYOGYAKARTA
KEMANDIRIAN – KEDISIPLINAN – KEPEDULIAN – KEBERSAMAAN



Lampiran VII. Hasil Penelitian dan Analisis Data

1. Penilaian kumulatif ahli perangkat lunak

a. *Syntax error*

| | No Butir | | | | | | |
|-------|----------|---|---|---|---|---|----|
| No | 1 | 2 | 3 | 4 | 5 | 6 | |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 6 |
| 2 | 1 | 1 | 1 | 1 | 1 | 1 | 6 |
| Total | | | | | | | 12 |

b. Run time error

| | No Butir | | | |
|-------|----------|---|---|---|
| No | 7 | 8 | 9 | |
| 1 | 1 | 1 | 1 | 3 |
| 2 | 1 | 1 | 1 | 3 |
| Total | | | | 6 |

c. Logic error

| | No Butir | | | |
|-------|----------|----|----|---|
| No | 10 | 11 | 12 | |
| 1 | 1 | 1 | 1 | 3 |
| 2 | 1 | 1 | 1 | 3 |
| Total | | | | 6 |

d. White box testing

| | No Butir | | | | | | | | | | | | |
|-------|----------|----|----|----|----|----|----|----|----|----|----|----|----|
| No | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 12 |
| 2 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 8 |
| Total | | | | | | | | | | | | | 20 |

2. Penilaian kumulatif guru

a. Aspek performansi penggunaan

| | No Butir | | | | | |
|-------|----------|---|---|---|---|----|
| No | 1 | 2 | 3 | 4 | 5 | |
| 1 | 3 | 3 | 3 | 4 | 3 | 16 |
| 2 | 3 | 3 | 4 | 3 | 2 | 15 |
| 3 | 3 | 3 | 3 | 3 | 2 | 14 |
| Total | | | | | | 45 |

b. Aspek peroramansi tampilan

| | No Butir | | | | | | | |
|-------|----------|---|---|---|----|----|----|----|
| No | 6 | 7 | 8 | 9 | 10 | 11 | 12 | |
| 1 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 21 |
| 2 | 3 | 3 | 3 | 2 | 2 | 3 | 2 | 18 |
| 3 | 3 | 3 | 3 | 2 | 2 | 3 | 3 | 19 |
| Total | | | | | | | | 58 |

c. Aspek materi tes

| | No Butir | | | | |
|-------|----------|----|----|----|----|
| No | 13 | 14 | 15 | 16 | |
| 1 | 3 | 3 | 3 | 3 | 12 |
| 2 | 3 | 3 | 3 | 3 | 12 |
| 3 | 3 | 3 | 3 | 3 | 12 |
| Total | | | | | 36 |

d. Aspek kemanfaatan

| | No Butir | | | | | |
|-------|----------|----|----|----|----|----|
| No | 17 | 18 | 19 | 20 | 21 | |
| 1 | 3 | 2 | 4 | 3 | 2 | 14 |
| 2 | 3 | 3 | 4 | 4 | 3 | 17 |
| 3 | 4 | 3 | 4 | 3 | 4 | 18 |
| Total | | | | | | 49 |

e. Aspek kelebihan tes essay

| | No Butir | | | | | | |
|-------|----------|----|----|----|----|----|----|
| No | 22 | 23 | 24 | 25 | 26 | 27 | |
| 1 | 3 | 3 | 3 | 3 | 3 | 4 | 19 |
| 2 | 3 | 3 | 3 | 3 | 3 | 3 | 18 |
| 3 | 3 | 3 | 3 | 4 | 3 | 4 | 20 |
| Total | | | | | | | 57 |

3. Penilaian kumulatif siswa
 a. Aspek performasi penggunaan

| No | No Butir | | | | |
|------------------------|----------|------|------|------|------|
| | 1 | 2 | 3 | 4 | 5 |
| 1 | 4 | 4 | 4 | 4 | 4 |
| 2 | 4 | 4 | 4 | 4 | 4 |
| 3 | 3 | 3 | 3 | 3 | 3 |
| 4 | 4 | 4 | 4 | 4 | 4 |
| 5 | 4 | 4 | 4 | 4 | 4 |
| 6 | 4 | 4 | 4 | 4 | 4 |
| 7 | 3 | 2 | 3 | 3 | 2 |
| 8 | 3 | 3 | 3 | 3 | 3 |
| 9 | 2 | 2 | 4 | 3 | 3 |
| 10 | 3 | 3 | 3 | 3 | 3 |
| 11 | 3 | 3 | 3 | 3 | 3 |
| 12 | 4 | 4 | 4 | 4 | 4 |
| 13 | 3 | 3 | 3 | 2 | 3 |
| 14 | 3 | 3 | 4 | 3 | 4 |
| 15 | 2 | 3 | 3 | 3 | 4 |
| 16 | 3 | 3 | 3 | 3 | 3 |
| 17 | 3 | 2 | 2 | 3 | 3 |
| 18 | 3 | 3 | 3 | 2 | 3 |
| 19 | 3 | 4 | 4 | 3 | 3 |
| 20 | 3 | 3 | 3 | 3 | 3 |
| 21 | 3 | 3 | 3 | 3 | 3 |
| 22 | 3 | 2 | 3 | 3 | 3 |
| 23 | 3 | 2 | 3 | 3 | 4 |
| 24 | 3 | 3 | 3 | 3 | 3 |
| 25 | 3 | 4 | 4 | 4 | 3 |
| 26 | 3 | 3 | 3 | 3 | 2 |
| 27 | 3 | 2 | 3 | 3 | 2 |
| 28 | 3 | 3 | 3 | 4 | 3 |
| 29 | 4 | 4 | 4 | 4 | 4 |
| 30 | 3 | 3 | 3 | 3 | 3 |
| Rerata Skor Tiap Butir | 3.17 | 3.10 | 3.33 | 3.23 | 3.23 |
| Rerata Tiap Aspek | 3.21 | | | | |
| Presentase Tiap Aspek | 80.33 | | | | |

b. Aspek performasi tampilan

| No | No Butir | | | | | | |
|------------------------|----------|------|------|------|------|------|------|
| | 6 | 7 | 8 | 9 | 10 | 11 | 12 |
| 1 | 4 | 4 | 4 | 3 | 4 | 3 | 3 |
| 2 | 3 | 4 | 4 | 4 | 3 | 3 | 2 |
| 3 | 3 | 3 | 4 | 3 | 3 | 3 | 4 |
| 4 | 4 | 4 | 4 | 4 | 4 | 3 | 3 |
| 5 | 4 | 4 | 4 | 4 | 3 | 4 | 4 |
| 6 | 4 | 4 | 4 | 3 | 3 | 4 | 3 |
| 7 | 3 | 3 | 3 | 3 | 4 | 2 | 2 |
| 8 | 3 | 3 | 3 | 3 | 3 | 3 | 2 |
| 9 | 3 | 3 | 2 | 3 | 1 | 4 | 2 |
| 10 | 3 | 4 | 3 | 3 | 3 | 3 | 3 |
| 11 | 3 | 3 | 3 | 3 | 3 | 3 | 3 |
| 12 | 4 | 4 | 4 | 4 | 4 | 4 | 4 |
| 13 | 4 | 4 | 3 | 3 | 3 | 3 | 3 |
| 14 | 3 | 4 | 3 | 3 | 3 | 2 | 2 |
| 15 | 3 | 3 | 3 | 3 | 3 | 3 | 3 |
| 16 | 2 | 3 | 3 | 3 | 3 | 3 | 3 |
| 17 | 3 | 3 | 2 | 3 | 3 | 2 | 3 |
| 18 | 3 | 3 | 2 | 3 | 3 | 2 | 2 |
| 19 | 3 | 3 | 3 | 3 | 2 | 3 | 2 |
| 20 | 3 | 3 | 3 | 3 | 3 | 3 | 2 |
| 21 | 4 | 4 | 4 | 3 | 4 | 3 | 3 |
| 22 | 3 | 3 | 3 | 3 | 3 | 3 | 2 |
| 23 | 2 | 3 | 4 | 3 | 2 | 3 | 2 |
| 24 | 3 | 3 | 3 | 3 | 3 | 3 | 3 |
| 25 | 3 | 3 | 4 | 3 | 2 | 3 | 4 |
| 26 | 3 | 3 | 3 | 3 | 3 | 3 | 2 |
| 27 | 3 | 3 | 3 | 3 | 3 | 2 | 3 |
| 28 | 3 | 3 | 3 | 3 | 3 | 3 | 3 |
| 29 | 4 | 4 | 4 | 4 | 4 | 4 | 4 |
| 30 | 3 | 3 | 3 | 3 | 2 | 2 | 3 |
| Rerata Skor Tiap Butir | 3.20 | 3.37 | 3.27 | 3.17 | 3.00 | 2.97 | 2.80 |
| Rerata Tiap Aspek | 3.11 | | | | | | |
| Presentase Tiap Aspek | 77.74 | | | | | | |

c. Aspek materi tes

| No | No Butir | | | |
|------------------------|----------|------|------|------|
| | 13 | 14 | 15 | 16 |
| 1 | 3 | 3 | 3 | 4 |
| 2 | 4 | 3 | 3 | 3 |
| 3 | 3 | 2 | 2 | 2 |
| 4 | 4 | 4 | 4 | 4 |
| 5 | 3 | 3 | 3 | 3 |
| 6 | 3 | 3 | 3 | 3 |
| 7 | 3 | 3 | 2 | 2 |
| 8 | 4 | 4 | 3 | 3 |
| 9 | 3 | 2 | 3 | 4 |
| 10 | 3 | 3 | 3 | 3 |
| 11 | 3 | 2 | 3 | 3 |
| 12 | 3 | 3 | 3 | 3 |
| 13 | 2 | 2 | 3 | 2 |
| 14 | 3 | 3 | 4 | 3 |
| 15 | 3 | 3 | 3 | 3 |
| 16 | 2 | 3 | 3 | 3 |
| 17 | 3 | 3 | 2 | 3 |
| 18 | 4 | 3 | 3 | 3 |
| 19 | 3 | 3 | 4 | 3 |
| 20 | 2 | 3 | 3 | 2 |
| 21 | 3 | 3 | 4 | 4 |
| 22 | 3 | 3 | 3 | 3 |
| 23 | 3 | 3 | 3 | 3 |
| 24 | 3 | 3 | 3 | 3 |
| 25 | 4 | 4 | 4 | 3 |
| 26 | 2 | 3 | 3 | 2 |
| 27 | 3 | 2 | 3 | 3 |
| 28 | 4 | 4 | 3 | 3 |
| 29 | 4 | 4 | 4 | 4 |
| 30 | 3 | 3 | 3 | 3 |
| Rerata Skor Tiap Butir | 3.10 | 3.00 | 3.10 | 3.00 |
| Rerata Tiap Aspek | 3.05 | | | |
| Presentase Tiap Aspek | 76.25 | | | |

d. Aspek kemanfaatan

| No | No Butir | | | | |
|------------------------|----------|------|------|------|------|
| | 17 | 18 | 19 | 20 | 21 |
| 1 | 4 | 4 | 4 | 4 | 4 |
| 2 | 3 | 3 | 4 | 3 | 3 |
| 3 | 3 | 4 | 4 | 4 | 4 |
| 4 | 4 | 4 | 3 | 3 | 4 |
| 5 | 4 | 4 | 3 | 4 | 4 |
| 6 | 4 | 4 | 4 | 4 | 4 |
| 7 | 3 | 3 | 4 | 3 | 4 |
| 8 | 4 | 4 | 4 | 4 | 4 |
| 9 | 1 | 3 | 2 | 2 | 3 |
| 10 | 4 | 4 | 3 | 3 | 4 |
| 11 | 3 | 3 | 3 | 3 | 3 |
| 12 | 4 | 4 | 4 | 4 | 4 |
| 13 | 3 | 3 | 3 | 3 | 3 |
| 14 | 4 | 4 | 4 | 4 | 4 |
| 15 | 4 | 3 | 4 | 4 | 3 |
| 16 | 3 | 4 | 4 | 3 | 4 |
| 17 | 3 | 3 | 3 | 3 | 3 |
| 18 | 4 | 3 | 4 | 3 | 3 |
| 19 | 4 | 4 | 3 | 4 | 4 |
| 20 | 3 | 3 | 3 | 2 | 3 |
| 21 | 4 | 4 | 4 | 4 | 4 |
| 22 | 4 | 3 | 3 | 3 | 2 |
| 23 | 3 | 2 | 3 | 3 | 3 |
| 24 | 3 | 4 | 4 | 4 | 4 |
| 25 | 4 | 3 | 4 | 3 | 3 |
| 26 | 3 | 3 | 3 | 3 | 3 |
| 27 | 3 | 3 | 3 | 3 | 3 |
| 28 | 4 | 3 | 4 | 4 | 4 |
| 29 | 4 | 3 | 4 | 3 | 4 |
| 30 | 4 | 4 | 4 | 3 | 3 |
| Rerata Skor Tiap Butir | 3.50 | 3.43 | 3.53 | 3.33 | 3.50 |
| Rerata Tiap Aspek | 3.46 | | | | |
| Presentase Tiap Aspek | 86.50 | | | | |

e. Aspek kelebihan tes essay

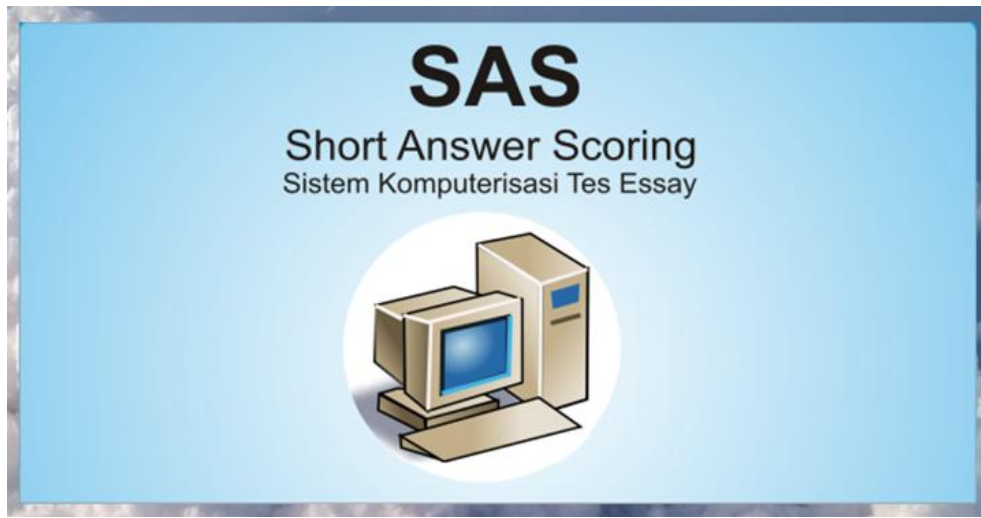
| No | No Butir | | | |
|------------------------|----------|------|------|------|
| | 22 | 23 | 24 | 25 |
| 1 | 3 | 4 | 4 | 3 |
| 2 | 4 | 3 | 3 | 3 |
| 3 | 4 | 4 | 4 | 4 |
| 4 | 4 | 4 | 4 | 4 |
| 5 | 3 | 3 | 3 | 4 |
| 6 | 4 | 4 | 4 | 4 |
| 7 | 2 | 4 | 4 | 4 |
| 8 | 4 | 4 | 4 | 4 |
| 9 | 3 | 3 | 4 | 4 |
| 10 | 3 | 3 | 3 | 3 |
| 11 | 3 | 3 | 3 | 3 |
| 12 | 4 | 4 | 4 | 4 |
| 13 | 2 | 3 | 2 | 2 |
| 14 | 4 | 4 | 4 | 4 |
| 15 | 3 | 4 | 3 | 2 |
| 16 | 3 | 3 | 3 | 3 |
| 17 | 3 | 3 | 3 | 2 |
| 18 | 4 | 4 | 4 | 4 |
| 19 | 3 | 4 | 4 | 4 |
| 20 | 3 | 3 | 2 | 2 |
| 21 | 4 | 4 | 4 | 4 |
| 22 | 3 | 3 | 3 | 3 |
| 23 | 3 | 3 | 3 | 3 |
| 24 | 4 | 4 | 4 | 4 |
| 25 | 3 | 4 | 3 | 4 |
| 26 | 2 | 3 | 3 | 3 |
| 27 | 4 | 4 | 3 | 3 |
| 28 | 2 | 3 | 4 | 4 |
| 29 | 4 | 4 | 3 | 3 |
| 30 | 2 | 3 | 3 | 3 |
| Rerata Skor Tiap Butir | 3.23 | 3.53 | 3.40 | 3.37 |
| Rerata Tiap Aspek | 3.38 | | | |
| Presentase Tiap Aspek | 84.58 | | | |

Hasil Uji Coba Tes

| No Siswa | No Soal | 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 |
|----------|----------|----|----|----|----|----|----|----|----|---|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| | Skor Mak | 15 | 15 | 10 | 10 | 10 | 10 | 10 | 10 | 5 | 5 | 15 | 15 | 10 | 10 | 10 | 10 | 5 | 5 | 15 | 15 | 10 | 10 | 10 | 5 | 5 |
| 1 | | 5 | 6 | 10 | 8 | 8 | 9 | 8 | 8 | 3 | 4 | 4 | 4 | 5 | 6 | 7 | 8 | 4 | 5 | 6 | 6 | 10 | 7 | 8 | 5 | 3 |
| 2 | | 5 | 6 | 9 | 10 | 10 | 5 | 10 | 8 | 4 | 5 | 4 | 4 | 9 | 5 | 7 | 5 | 3 | 3 | 5 | 5 | 5 | 8 | 6 | 3 | 3 |
| 3 | | 5 | 4 | 6 | 5 | 7 | 7 | 5 | 8 | 4 | 3 | 4 | 6 | 7 | 3 | 4 | 5 | 3 | 5 | 6 | 6 | 8 | 9 | 8 | 4 | 5 |
| 4 | | 4 | 5 | 9 | 8 | 10 | 8 | 9 | 8 | 4 | 4 | 4 | 4 | 8 | 10 | 8 | 8 | 5 | 5 | 5 | 5 | 8 | 10 | 8 | 5 | 4 |
| 5 | | 6 | 6 | 8 | 10 | 8 | 10 | 8 | 9 | 5 | 5 | 5 | 6 | 6 | 6 | 5 | 5 | 4 | 4 | 4 | 6 | 8 | 5 | 6 | 3 | 4 |
| 6 | | 6 | 6 | 10 | 8 | 10 | 8 | 8 | 10 | 5 | 5 | 4 | 7 | 7 | 7 | 5 | 5 | 4 | 4 | 6 | 5 | 9 | 9 | 9 | 5 | 5 |
| 7 | | 5 | 6 | 5 | 7 | 3 | 5 | 6 | 7 | 5 | 4 | 6 | 6 | 6 | 6 | 3 | 5 | 3 | 3 | 6 | 4 | 8 | 3 | 9 | 4 | 5 |
| 8 | | 5 | 4 | 6 | 7 | 6 | 5 | 7 | 5 | 4 | 3 | 4 | 6 | 9 | 9 | 6 | 6 | 5 | 5 | 6 | 6 | 8 | 9 | 8 | 5 | 5 |
| 9 | | 4 | 6 | 10 | 7 | 5 | 7 | 7 | 4 | 4 | 5 | 5 | 4 | 6 | 4 | 5 | 9 | 1 | 3 | 4 | 4 | 5 | 6 | 5 | 4 | 4 |
| 10 | | 6 | 5 | 5 | 5 | 6 | 6 | 8 | 5 | 3 | 3 | 4 | 4 | 7 | 5 | 6 | 5 | 5 | 5 | 5 | 5 | 9 | 7 | 5 | 3 | 3 |
| 11 | | 5 | 5 | 7 | 5 | 7 | 6 | 6 | 6 | 3 | 3 | 9 | 6 | 6 | 5 | 7 | 6 | 3 | 3 | 6 | 4 | 5 | 6 | 7 | 3 | 3 |
| 12 | | 6 | 6 | 8 | 10 | 9 | 8 | 8 | 10 | 4 | 5 | 5 | 6 | 6 | 6 | 5 | 6 | 4 | 5 | 5 | 5 | 9 | 10 | 9 | 5 | 5 |
| 13 | | 6 | 4 | 5 | 5 | 5 | 10 | 9 | 6 | 3 | 3 | 5 | 6 | 3 | 4 | 6 | 4 | 3 | 3 | 5 | 5 | 6 | 3 | 5 | 4 | 2 |
| 14 | | 5 | 4 | 9 | 5 | 10 | 5 | 9 | 5 | 4 | 3 | 5 | 5 | 6 | 6 | 8 | 6 | 4 | 4 | 6 | 6 | 10 | 10 | 9 | 4 | 4 |
| 15 | | 4 | 4 | 6 | 7 | 9 | 6 | 5 | 6 | 4 | 3 | 4 | 6 | 5 | 7 | 6 | 6 | 5 | 3 | 5 | 5 | 6 | 5 | 9 | 3 | 4 |
| 16 | | 4 | 5 | 6 | 6 | 7 | 4 | 7 | 5 | 3 | 3 | 5 | 6 | 5 | 7 | 6 | 6 | 3 | 4 | 6 | 4 | 9 | 5 | 7 | 3 | 3 |
| 17 | | 6 | 7 | 5 | 5 | 5 | 7 | 6 | 3 | 3 | 3 | 5 | 4 | 6 | 7 | 4 | 5 | 3 | 3 | 5 | 5 | 5 | 7 | 5 | 3 | 4 |
| 18 | | 4 | 5 | 6 | 3 | 5 | 6 | 5 | 3 | 4 | 3 | 5 | 6 | 8 | 6 | 6 | 5 | 5 | 3 | 6 | 6 | 5 | 7 | 8 | 9 | 5 |
| 19 | | 5 | 6 | 10 | 5 | 5 | 6 | 6 | 6 | 3 | 5 | 5 | 6 | 6 | 6 | 8 | 7 | 5 | 5 | 5 | 5 | 6 | 9 | 5 | 9 | 5 |
| 20 | | 5 | 5 | 7 | 7 | 6 | 5 | 5 | 6 | 4 | 3 | 5 | 5 | 4 | 6 | 7 | 5 | 3 | 3 | 4 | 4 | 6 | 7 | 5 | 7 | 4 |
| 21 | | 5 | 5 | 5 | 6 | 7 | 9 | 8 | 9 | 3 | 4 | 5 | 6 | 7 | 7 | 9 | 9 | 5 | 4 | 6 | 6 | 10 | 8 | 9 | 5 | 5 |
| 22 | | 4 | 4 | 6 | 6 | 6 | 7 | 5 | 7 | 4 | 3 | 5 | 5 | 5 | 5 | 5 | 5 | 4 | 3 | 4 | 4 | 5 | 4 | 6 | 7 | 3 |
| 23 | | 6 | 6 | 5 | 7 | 10 | 4 | 5 | 8 | 3 | 5 | 5 | 6 | 5 | 5 | 5 | 5 | 3 | 4 | 4 | 4 | 6 | 7 | 7 | 3 | 3 |
| 24 | | 5 | 5 | 6 | 6 | 5 | 7 | 7 | 7 | 4 | 3 | 5 | 6 | 7 | 6 | 6 | 7 | 3 | 5 | 6 | 6 | 10 | 9 | 8 | 5 | 4 |
| 25 | | 5 | 6 | 10 | 8 | 5 | 7 | 7 | 10 | 3 | 5 | 5 | 6 | 9 | 9 | 10 | 6 | 4 | 3 | 6 | 4 | 5 | 7 | 8 | 3 | 4 |
| 26 | | 5 | 5 | 6 | 5 | 5 | 7 | 7 | 5 | 4 | 3 | 4 | 5 | 5 | 5 | 6 | 5 | 3 | 3 | 4 | 6 | 7 | 4 | 6 | 3 | 3 |
| 27 | | 6 | 5 | 5 | 6 | 4 | 6 | 7 | 5 | 3 | 3 | 4 | 4 | 7 | 3 | 6 | 7 | 3 | 3 | 5 | 6 | 5 | 8 | 8 | 3 | 3 |
| 28 | | 5 | 5 | 6 | 10 | 5 | 5 | 6 | 6 | 4 | 3 | 5 | 6 | 9 | 9 | 7 | 5 | 4 | 3 | 5 | 4 | 9 | 5 | 6 | 5 | 5 |
| 29 | | 6 | 6 | 9 | 10 | 8 | 10 | 8 | 10 | 5 | 5 | 5 | 6 | 9 | 8 | 10 | 8 | 5 | 3 | 5 | 4 | 10 | 9 | 9 | 3 | 3 |
| 30 | | 6 | 5 | 6 | 6 | 6 | 5 | 5 | 5 | 5 | 5 | 6 | 4 | 6 | 7 | 7 | 7 | 5 | 5 | 5 | 4 | 6 | 5 | 6 | 3 | 3 |

**PENGEMBANGAN MODEL TES ESSAY BERBANTUAN KOMPUTER
UNTUK MATA PELAJARAN ELEKTRONIKA DASAR DI SMK N 2
YOGYAKARTA**

Petunjuk Penggunaan Software



Oleh:

Fajar Prastya

PROGRAM STUDI PENDIDIKAN TEKNIK MEKATRONIKA

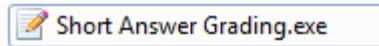
FAKULTAS TEKNIK

UNIVERSITAS NEGERI YOGYAKARTA

2015

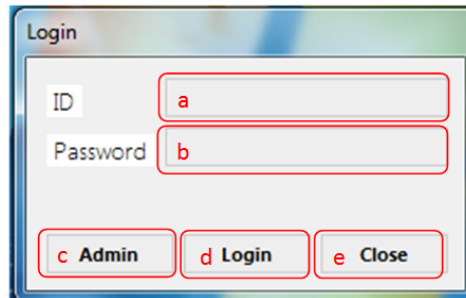
Petunjuk Penggunaan *Software Tes Essay* Berbantuan Komputer

1. Membuka Program



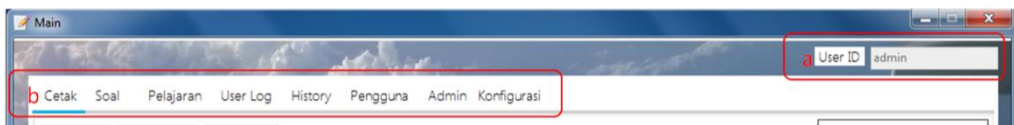
Klik Short Answer Grading

2. Tampilan Login



- a. Kolom ID -> Nama pengguna
- b. Kolom Password -> Password pengguna
- c. Tombol Admin -> Login sebagai Admin
- d. Tombol Login -> Login Sebagai Guru/ Siswa
- e. Tombol Close -> Menutup program

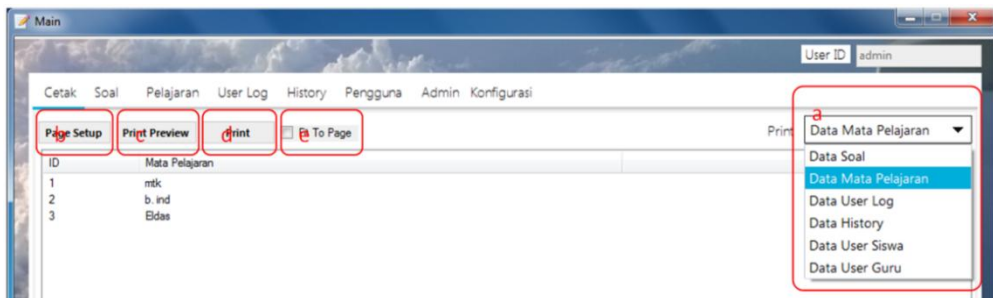
3. Tampilan Menu Utama



- a. Nama pengguna
- b. Menu Utama
Hak User
 - 1) Admin
 - Cetak : Mencetak data soal, daftar pelajaran, daftar user log, daftar history, daftar pengguna guru dan daftar pengguna siswa
 - Soal : membuat, mengedit dan menghapus soal
 - Pelajaran : menambah, mengedit dan menghapus daftar mata pelajaran
 - User Log : menghapus dan reset data hasil tes
 - History : menghapus dan reset data user
 - Pengguna : menambah, mengedit dan menghapus user siswa
 - Admin : menambah, mengedit dan menghapus user guru
 - Config : mengatur konfigurasi tes
 - 2) Guru
 - Cetak : Mencetak data soal, daftar pelajaran, daftar user log, daftar history, daftar pengguna guru dan daftar pengguna siswa
 - Soal : membuat, mengedit dan menghapus soal

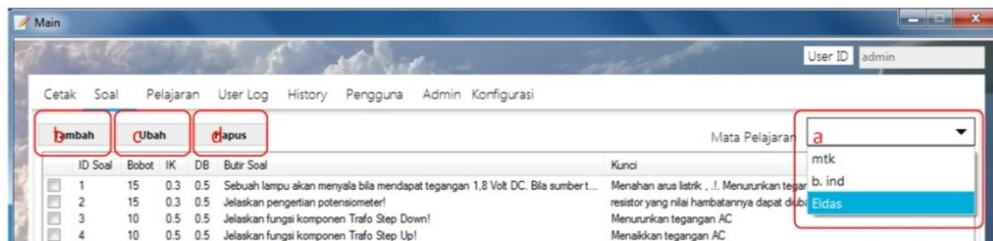
- Pelajaran : menambah, mengedit dan menghapus daftar mata pelajaran
 - User Log : menghapus dan reset data hasil tes
 - History : menghapus dan reset data user
 - Pengguna : menambah, mengedit dan menghapus user siswa
 - Config : mengatur konfigurasi tes
- 3) Siswa
Mengerjakan tes

4. Mencetak Data



- a. Pemilihan data yang akan dicetak -> Pilih pada Combo Box
- b. Mengatur halaman -> Klik Page Setup
- c. Menampilkan halaman sebelum di cetak -> Klik Print Preview
- d. Memulai proses cetak -> Klik Print
- e. Menyesuaikan data dengan ukuran kertas -> Centang Cek Box

5. Data Soal

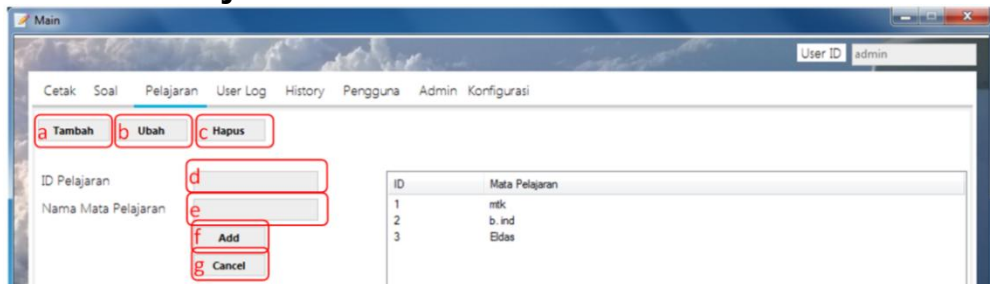


- a. Memilih daftar mata pelajaran -> Pilih pada Combo Box
- b. Menambah soal baru -> Klik Tambah
- c. Mengubah soal -> pilih pada List -> klik Ubah
- d. Menghapus soal -> pilih pada List -> klik Hapus

6. Tampilan Tambah Soal

- Kolom nomor soal
- Kolom mata pelajaran
- Kolom bobot soal
- Kolom indeks kesukaran soal
- Kolom daya beda soal
- Kolom butir soal
- Kolom kunci jawaban soal
- Tombol Add soal kedalam database
- Tombol batal

7. Data Mata Pelajaran



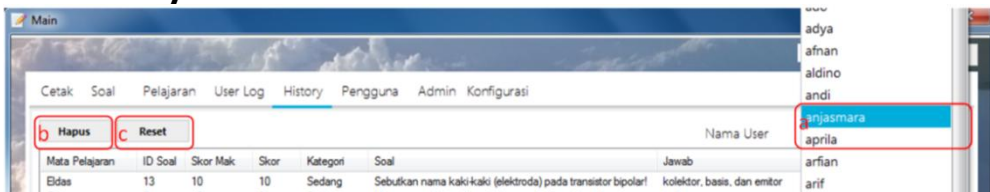
- Menambah mata pelajaran -> klik Tambah
- Mengubah mata pelajaran -> pilih pada List -> klik Ubah
- Menghapus mata pelajaran -> pilih pada List -> klik Hapus
- Kolom nomor pelajaran
- Kolom nama mata pelajaran
- Tombol Add mata pelajaran kedalam database
- Tombol Batal

8. Data User Log



- Memilih daftar mata pelajaran -> pilih pada combobox
- Tombol hapus data user log -> pilih pada List -> klik Hapus
- Tombol reset data -> klik untuk menghapus semua data mata pelajaran yang dipilih

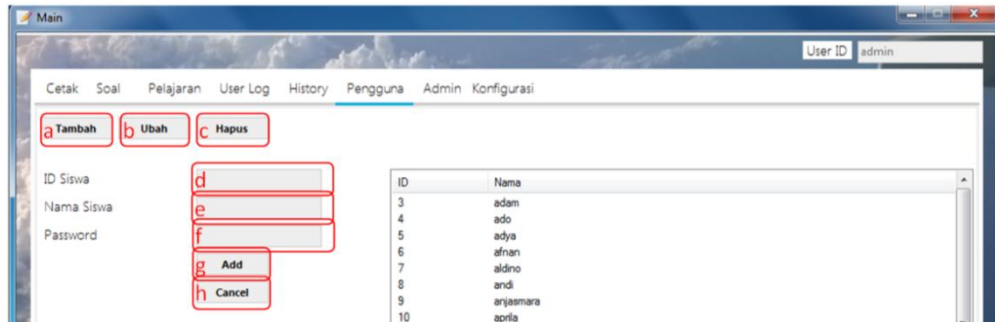
9. Data History



- Memilih daftar nama siswa -> pilih pada combobox
- Tombol hapus data history -> pilih pada List -> klik Hapus

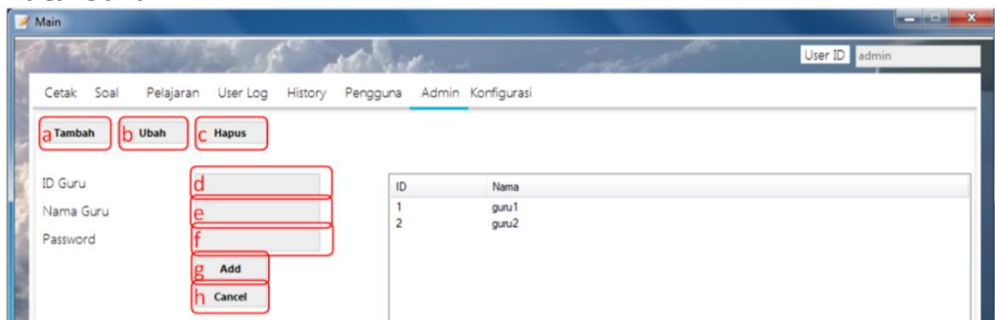
- c. Tombol reset data -> klik untuk menghapus semua data siswa yang dipilih

10. Data Siswa



- Menambah data user siswa -> klik Tambah
- Mengubah data user siswa -> pilih pada List -> klik Ubah
- Menghapus data user siswa -> pilih pada List -> klik Hapus
- Kolom nomor user siswa
- Kolom nama user siswa
- Kolom password user siswa
- Tombol Add data user siswa kedalam database
- Tombol Batal

11. Data Guru



- Menambah data user guru -> klik Tambah
- Mengubah data user guru -> pilih pada List -> klik Ubah
- Menghapus data user guru -> pilih pada List -> klik Hapus
- Kolom nomor user guru
- Kolom nama user guru
- Kolom password user guru
- Tombol Add data user guru kedalam database
- Tombol Batal

12. Konfigurasi

- Memilih daftar mata pelajaran untuk tes
- Menentukan jumlah soal
- Menentukan waktu tes maksimal
- Tombol simpan pengaturan

13. Tampilan Mulai Tes



- Tombol untuk memulai tes
- Tombol LogOut

14. Tampilan Tes

- Nomor soal
- Waktu tersisa
- Kolom soal
- Kolom jawaban
- Tombol soal sebelumnya
- Tombol soal selanjutnya
- Tombol selesai mengerjakan tes

Catatan:

- Versi Net Framework yang digunakan adalah .Net Framework 4.0
- Net Framework komputer perlu di update ke versi .Net Framework 4.0