

Pembangunan Sistem E-Learning Untuk Kebutuhan TOT (Training Of Trainer) Guru Pada Smk Yp 17-1 Malang Dengan Menggunakan Framework Codeigniter

Niar Ariati¹, Heru Nurwarsito², Denny Sagita Rusdianto³

Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Brawijaya
Email: ¹niar.ariati@gmail.com, ²heru@ub.ac.id, ³denny.sagita@ub.ac.id

Abstrak

SMK YP 17-1 Malang merupakan suatu yayasan pendidikan formal yang menyelenggarakan pendidikan kejuruan pada jenjang pendidikan menengah sebagai lanjutan dari SMP, MTs, atau bentuk lain yang sederajat. Sebagai lembaga pendidikan formal, sudah menjadi kewajiban untuk mengikuti kebijakan pendidikan yang telah ditetapkan pemerintah. Dalam pelaksanaan kebijakan tersebut didapatkan berbagai kendala seperti pada pelaksanaan kebijakan pendidikan dalam hal kurikulum yang berkembang secara dinamis serta pengembangan profesional staf yang saling berkaitan. Melihat beberapa kendala dan masalah yang terjadi, diperlukan kemampuan, kerjasama atau timbal balik antar guru dalam mengajarkan/mentransfer pengetahuannya. Oleh karena itu, perlu dibangun sebuah sistem perangkat lunak yang mampu melakukan manajemen pembelajaran/pelatihan dan pelayanan informasi yang berbasis web untuk kebutuhan TOT (*Training of Trainer*) guru. Sistem yang dibangun dengan menggunakan kerangka kerja Codeigniter yang mengimplementasikan model proses air terjun ini telah memenuhi kebutuhan fungsional sistem dengan presentase 100% valid dan ketiga fungsi utama memiliki kompleksitas di bawah 21 yang menunjukkan bahwa program cukup mudah untuk dikelola. Selain itu, menurut kepuasan calon pengguna, sistem memiliki rata-rata skor *System Usability Scale* (SUS) sebesar 84,17 yang mengindikasikan bahwa sistem perangkat lunak yang dibangun termasuk dalam kategori "Good" dan bernilai *adjective* "B".

Kata kunci: *e-learning*, TOT, Codeigniter, model proses air terjun, SUS

Abstract

SMK YP-17-1 Malang is an organization that operates a formal education. It holds a vocational education on a senior-high grade as a continuation from Junior high (SMP, MTs or any comparable junior high). As an organization that holds a formal education, it is imperative for SMK YP 17-1 to implement any governmental policy in its operation. In the process, there will be problems such as difficulties on implementing the dynamically changing curriculum and the problem of developing the staff's professionalism. Based on that fact, we need the skill, cooperation, and feedback among teachers on their daily routine so that they can share their knowledge. Hence, need to implemented a software to manage education/training process and also acts as a web-based information gateway for Training of trainers system. The completed system uses CodeIgniter framework that implements Waterfall development cycle and fulfills the functional requirement with 100% validation percentage. All three of the main feature has a complexity level under 21 which proves that this system can be operated with easy maintenance. Moreover, user satisfaction value reaches 84.17% which indicates that the completed system is categorized as "good" and has an adjective value of "B".

Keywords: *e-learning*, TOT, Codeigniter, waterfall development cycle, SUS

1. PENDAHULUAN

SMK YP 17-1 Malang merupakan suatu yayasan pendidikan formal yang menyelenggarakan pendidikan kejuruan dimana memiliki kewajiban untuk mengikuti kebijakan

pendidikan yang telah ditetapkan pemerintah. Beberapa kebijakan publik dalam bidang pendidikan meliputi anggaran pendidikan, kurikulum, rekrutmen tenaga kependidikan, pengembangan profesional staf, pengelolaan sumber daya, dan kebijakan lain yang

bersentuhan langsung maupun tak langsung atas pendidikan (Solichin, 2015). Dalam pelaksanaan kebijakan tersebut didapatkan berbagai kendala, contohnya pada pelaksanaan kebijakan pendidikan dalam hal kurikulum serta pengembangan profesional staf.

Dalam lingkungan SMK YP 17-1 Malang, guru yang produktif dan mengajar mata pelajaran yang diujikan dalam Ujian Nasional telah menerima pelatihan kurikulum 2013 sedangkan guru lainnya belum menerima pelatihan. Saat ini agar guru yang belum menerima pelatihan memiliki dasar pemahaman yang sama tentang pengimplementasian kurikulum 2013, diberikan sosialisasi oleh wakil ketua kurikulum melalui rapat. Selanjutnya secara mandiri mencari sumber belajar lain melalui internet ataupun bertanya/berdiskusi dengan guru yang sudah menerima pelatihan kurikulum.

Selain itu, kurangnya pengetahuan penggunaan teknologi informasi untuk mandiri mencari sumber belajar dari beberapa guru serta kesulitan dalam bertanya/berdiskusi dengan guru lain dalam lingkungan sekolah dikarenakan terkadang guru berada di sekolah hanya pada jam-jam mengajar saja atau guru mengajar di sekolah lain untuk memenuhi kewajiban mengajar 24 jam tatap muka (bagi guru yang telah/akan sertifikasi). Melihat masalah ini, diperlukan kemampuan atau kerjasama atau timbal balik antar guru dalam mentransfer pengetahuannya akan pelatihan kurikulum 2013 yang pernah diterima maupun mengajarkan ranah materi keahliannya untuk mendukung pengembangan profesional staf yang dilakukan saat ini (sertifikasi guru maupun keahlian ganda) kepada guru lain.

Berdasarkan pemaparan diatas, perlu dibangun sebuah sistem berbasis web untuk memberikan wadah bertanya, berdiskusi, atau menyampaikan pembelajaran/pelatihan dan informasi dengan berfokus pada pengumpulan data CRUD (*Create, Read, Update, Delete*) dan jenis soal *multiple choice* untuk evaluasi kegiatan pembelajaran dengan penilaian menggunakan penskoran beda bobot. Sistem ini dibangun menggunakan *framework* Codeigniter yang memiliki konsep MVC (*Model-View-Controller*) serta tersedianya semua class dan modul-modul umum sehingga pembangunan web akan lebih terstruktur serta mempercepat dan memudahkan pengembang dalam pengkodean program. Dalam implementasinya, kerangka kerja Codeigniter dikombinasikan

dengan model proses *waterfall* sebagai acuan dalam pengembangan sistem dimana setiap tahap prosesnya dilakukan secara sistematis dan berurutan. Dan terakhir hasil implementasinya diuji dengan metode kuisioner *System Usability Scale* (SUS) dengan kriteria *efficiency* dan *satisfaction* untuk mengukur *usability* desain sistem serta tingkat kemudahan setiap fungsi desain sistem sehingga dapat diketahui kualitas sistem perangkat lunak menurut penilaian pengguna. Pengujian SUS ini digunakan dengan pertimbangan jumlah sampel yang kecil, waktu dan biaya yang diperlukan kecil pula karena kuisioner sudah tersedia beserta perhitungan pengkonversian ke dalam kriteria kelayakan penerapan suatu sistem perangkat lunak yang dapat diandalkan dengan penelitian Bangor, dkk (2009) mengenai penggunaan SUS dengan berbagai variasi sistem dan teknologi.

Sehingga diharapkan dengan adanya sistem yang akan dibangun ini dapat membantu menyelesaikan permasalahan manajemen pembelajaran atau pelatihan dan layanan informasi yang ada di SMK YP 17-1 Malang.

2. PENSKORAN BEDA BOBOT

Guna mengukur tingkat kephahaman/daya serap peserta terhadap materi mentoring yang telah diberikan dan sebagai bahan evaluasi proses mentoring, disajikan sebuah ujian yang mengakomodasi soal pilihan ganda. Dimana pemberian skor pada tes domain kognitif ini disesuaikan dengan pedoman yang dikeluarkan oleh Depdiknas yang telah dimodifikasi seperti dikutip Ainur Rofieq (2008), yaitu penskoran butir beda bobot. Pada jenis penskoran ini pemberian skor dengan memberikan bobot berbeda pada sekelompok butir soal dan biasanya bobot butir soal tersebut menyesuaikan dengan tingkatan kognitif (pengetahuan, pemahaman, penerapan, analisis, sintesis, dan evaluasi). Adapun rumusnya:

$$\text{Skor} = \sum \frac{(B_i \times b_i)}{St} \times 100\% \quad (1)$$

Dengan B_i menyatakan banyaknya butir soal yang dijawab benar, b_i menyatakan bobot setiap butir soal, dan St menyatakan skor teoritis (skor bila menjawab benar semua). Contoh perhitungan skor menggunakan penskoran beda bobot ditunjukkan sebagai berikut.

Pada ujian Umum berjumlah 30 butir yang terdiri dari enam tingkat domain kognitif dengan masing-masing bobot sebagai berikut: pengetahuan dengan bobot 1, pemahaman 2, penerapan 3, analisis 4, sintesis 5, dan evaluasi

6. Peserta A dapat menjawab benar 6 dari 10 butir soal domain pengetahuan, 10 dari 12 butir soal pemahaman, 2 dari 4 butir soal penerapan, 1 dari 2 butir soal analisis, dan soal sintesis dan evaluasi masing-masing 1 butir soal. Untuk perhitungan skor disusun Tabel 1 berikut.

Tabel 1. Perhitungan pemberian skor

Domain butir soal	Jumlah butir	bi	Jml butir x bi	Bi	Bi x bi
Pengetahuan	10	1	10	6	6
Pemahaman	12	2	24	10	20
Penerapan	4	3	12	2	6
Analisis	2	4	8	1	4
Sintesis	1	5	5	1	5
Evaluasi	1	6	6	1	6
Jumlah	30	-	St = 65	-	47

$$\text{Skor} = \sum \frac{(6 \times 1) + (10 \times 2) + (2 \times 3) + (1 \times 4) + (1 \times 5) + (1 \times 6)}{65} \times 100\%$$

Dari perhitungan diatas didapatkan skor yang diperoleh Peserta A 72,31% dimana berarti Peserta A dapat menguasai mata pelajaran Umum sebesar 72,31%.

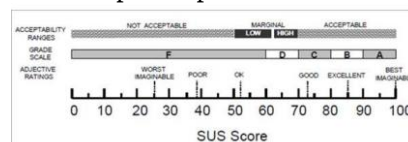
3. SYSTEM USABILITY SCALE (SUS)

Arti kata *usability* mengacu pada metode untuk melakukan improvisasi terhadap *easy-of-use* sistem. Aspek *Usability* dapat diukur menggunakan *tools System Usability Scale* (SUS). Tools ini dikembangkan oleh John Brooke pada tahun 1986. Pengujian dilakukan dengan memberikan 10 pertanyaan dimana setiap pertanyaan memiliki 5 opsi jawaban dimulai dari skala nomor 1 yang menunjukkan “Sangat tidak setuju” hingga 5 menunjukkan “Sangat setuju”. Sementara untuk pertanyaan yang (sama sekali) tidak dipahami oleh responden ditentukan skala nomor 3.

SUS memiliki strategi penilaian dimana keseluruhan item pertanyaan terbagi dalam 2 kelompok pertanyaan positif pada nomor ganjil (1,3,5,7,9) serta pertanyaan negatif pada nomor genap (2,4,6,8,10). Kontribusi setiap item pertanyaan positif dihitung dengan mengurangi poin jawaban responden dengan 1 poin, sedangkan untuk pertanyaan negatif kontribusi skornya dihitung dengan mengurangi poin 5 dengan poin jawaban responden. Setelah poin tiap responden diperoleh, seluruh poin tiap pertanyaan dijumlahkan lalu dikali dengan 2,5 sehingga yang awalnya kisaran nilai 0-40 berubah menjadi 0-100. Kemudian dihitung rerata skor SUS keseluruhan responden dengan membagi jumlah skor seluruh responden dengan jumlah responden. (Brooke, 2013)

Setelah diketahui skor rata-rata, kemudian dibandingkan dengan *range* nilai yang

menunjukkan interpretasi grade dari skor SUS seperti dikemukakan oleh Bangor, dkk (2009) yang telah mengumpulkan lebih dari 3500 data hasil *scaling* dari penggunaan SUS lebih dari 1 dekade dengan berbagai variasi sistem dan teknologi. Hal ini bertujuan mengevaluasi hasil *scaling* sebagai acuan menentukan adjektivitas dari nilai yang didapat seperti “Poor”, “Good”, dan “Excellent”. Berikut interpretasi grade dari skor SUS ditampilkan pada Gambar 2.

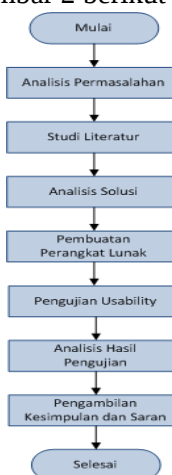


Gambar 1. Interpretasi grade skor SUS

Sumber: Bangor, dkk (2009)

4. METODOLOGI

Metodologi penelitian yang memuat sejumlah praktik manajemen dan teknis pengerjaan dalam penelitian ditunjukkan pada diagram alir Gambar 2 berikut ini.



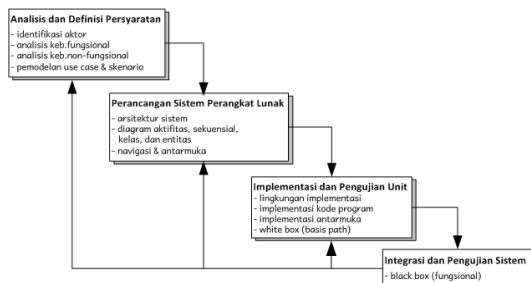
Gambar 2. Diagram alir metodologi penelitian

Langkah awal yang dilakukan pada penelitian seperti tergambar pada Gambar 2 yaitu analisis permasalahan. Pada langkah ini dilakukan analisis terhadap permasalahan dengan cara observasi dan wawancara pada lingkungan SMK YP 17-1 Malang. Selanjutnya dilakukan studi literatur dengan menyusun dasar pengetahuan yang berkaitan dengan informasi dari hasil analisis permasalahan.

Tahap ketiga yang dilakukan yaitu analisis solusi dimana pada tahap ini kegiatan awal yang dilakukan adalah mengkaji hasil dari analisis permasalahan dan studi literatur. Kemudian dilanjutkan dengan mengemukakan solusi penyelesaian permasalahan dan

melakukan diskusi dengan *client* guna memperoleh kesepakatan mengenai teknologi dasar, teknologi pengembangan perangkat lunak yang digunakan serta jadwal pelaksanaan kegiatan pengembangan perangkat lunak.

Keempat, pembuatan perangkat lunak, dilakukan pengembangan perangkat lunak yang terdiri dari beberapa tahap kegiatan sesuai dengan model proses yang dijadikan acuan penulis berdasarkan hasil analisis solusi, yaitu model proses *waterfall* yang diadaptasi dari Sommerville (2007) namun tahap operasi dan pemeliharaan tidak disertakan karena tahapan tersebut tidak dilakukan dalam penelitian ini. Berikut representasi model proses *waterfall* tersebut beserta kegiatan yang dilakukan pada tiap tahap dapat dilihat pada Gambar 3.



Gambar 3. Model proses *waterfall* pembuatan perangkat lunak

Tahap selanjutnya yang dilakukan dalam penelitian yaitu melakukan pengujian *usability* menggunakan metode kuisioner SUS (*System Usability Scale*) dan melakukan analisis terhadap hasil pengujian yang telah dilakukan diantaranya yaitu pengujian *basis path*, pengujian fungsional, serta pengujian *usability*. Terakhir, dilakukan pengambilan kesimpulan dari hasil implementasi serta pengujian yang dilakukan dan saran sebagai pertimbangan atas pengembangan lebih lanjut serta memperbaiki kesalahan yang masih terdapat dalam sistem.

5. ANALISIS KEBUTUHAN

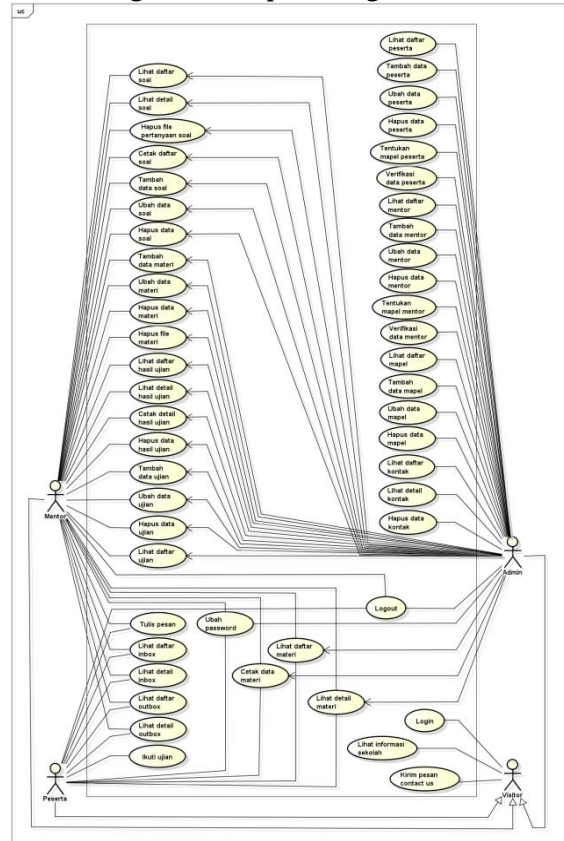
5.1 Kebutuhan Non-Fungsional

Sistem perangkat lunak manajemen layanan informasi dan pelatihan/pembelajaran yang akan dikembangkan ini nantinya akan diserahkan kepada *customer*, SMK YP 17-1 Malang. Sehingga diperlukan pengujian *usability* dari sistem perangkat lunak yang akan dikembangkan untuk mengetahui tingkat kepuasan pelanggan dengan parameter pengujian dilakukan dengan menggunakan

metode *System Usability Scale* (SUS) menghasilkan skor diatas 70,

5.2 Kebutuhan Fungsional

Pada tahap ini diuraikan kebutuhan yang harus disediakan oleh sistem serta pengguna yang dapat melakukan interaksi dengan sistem tersebut digambarkan pada diagram *use case*.

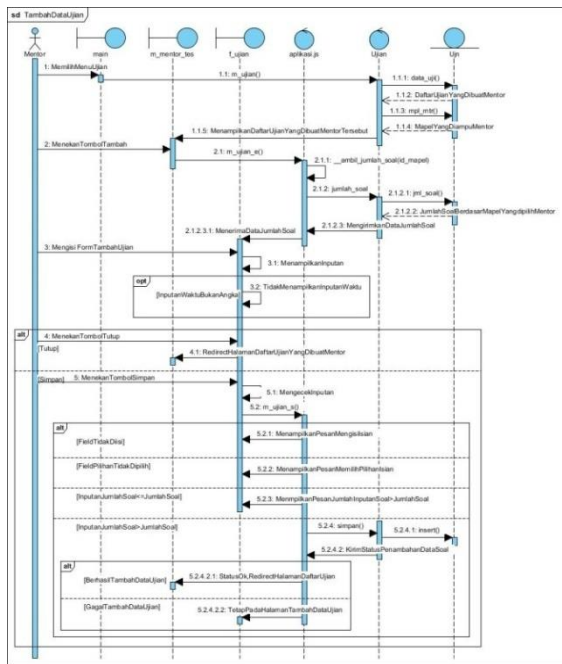


Gambar 4. Diagram *use case*

6. PERANCANGAN

6.1 Diagram Sequence

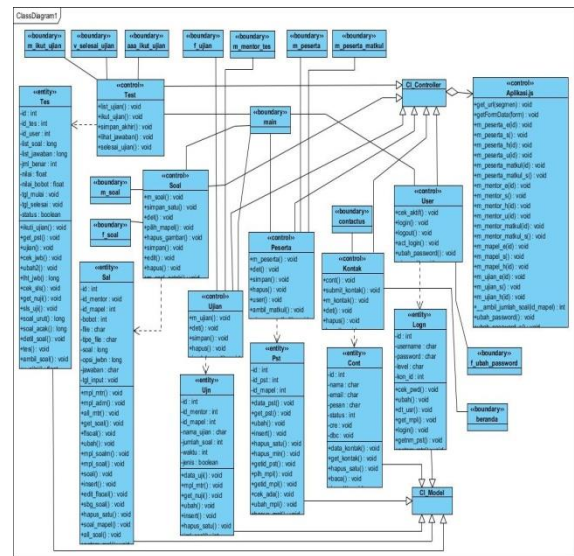
Pada bagian ini diilustrasikan rangkaian langkah-langkah yang dilakukan sebagai respon dari sebuah event untuk menghasilkan keluaran (*output*) tertentu. Berikut salah satu pengilustrasian dalam fungsi tambah data ujian.



Gambar 5. Diagram sequence tambah data ujian

6.2 Diagram Class

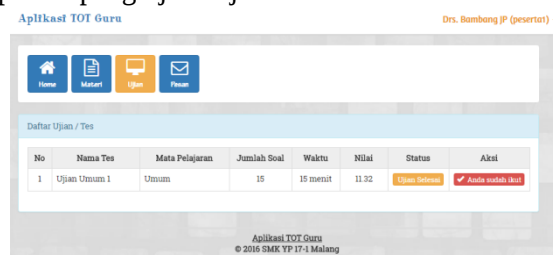
Berdasarkan hasil perancangan diagram sequence, digambarkan struktur dan deskripsi kelas, objek beserta hubungannya satu sama lain dalam sebuah diagram kelas. Pada gambar 6 dijelaskan bahwa terdapat hubungan generalisasi semua *class controller* (*control*) terhadap *class CI_Controller* dimana semua *class controller* (*control*) adalah *child class* *CI_Controller*. *Class child* dari *CI_Controller* memiliki hubungan agregasi dengan *class aplikasi.js*. *Class* *aplikasi.js* merupakan kelas penyusun dari *class child* *CI_Controller* dimana setiap *class child controller* memerlukan fungsi `getFormData($form)` untuk dapat menampilkan data *form* dan `get_url(segmen)` untuk mengambil url segmen Codeigniter. Sehingga dapat digambarkan bahwa *class CI_Controller* memiliki hubungan agregasi dengan *class aplikasi.js*. Sementara itu, semua *class model* (*entity*) memiliki hubungan generalisasi terhadap *class CI_Model*, *class control* memiliki hubungan asosiasi yang menyatakan hubungan statis dengan *class boundary* (*view*), dan *class control* memiliki hubungan dependensi dengan *class entity* karena *class control* bergantung terhadap *class entity* dimana operasi dalam *class control* menggunakan *class entity*.



Gambar 6. Diagram Class

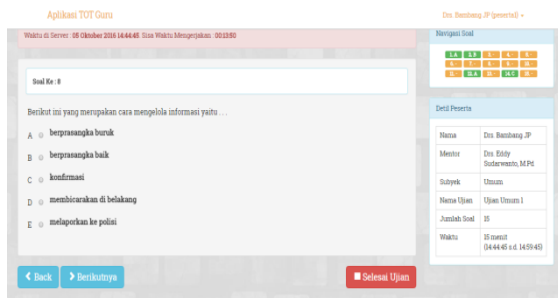
7. IMPLEMENTASI

Sebagai bentuk evaluasi terhadap proses mentoring, seorang peserta diberikan sebuah test terhadap materi yang diberikan. Pada gambar 7 berikut menampilkan utama menu ujian seorang peserta pada kolom aksi menampilkan aksi yang dapat dilakukan peserta berdasarkan status ujian apakah ujian telah dikerjakan, belum dikerjakan, atau dalam proses pengerjaan ujian.



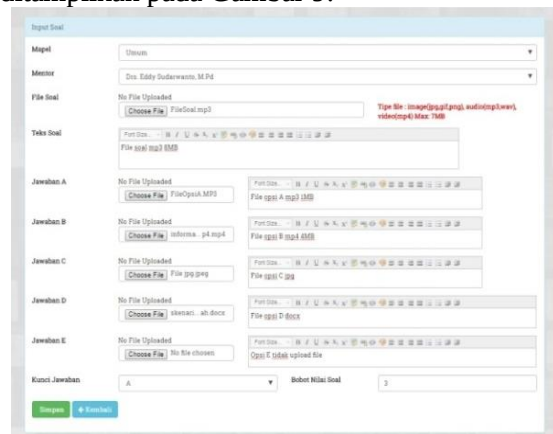
Gambar 7. Antarmuka halaman utama ujian

Pada Gambar 8 menampilkan antarmuka halaman saat peserta mengerjakan ujian. Soal ditampilkan satu persatu. Dalam halaman ini terdapat navigasi soal, detail peserta, dan waktu pengerjaan ujian. Waktu pengerjaan soal menampilkan waktu pengerjaan saat ini dan hitung mundur waktu pengerjaan soal. Navigasi soal digunakan untuk melihat atau melakukan pengecekan soal yang belum/telah dijawab dengan cara menekan nomor soal maka akan diarahkan ke nomor soal yang dipilih. Jika waktu pengerjaan soal sudah habis maka akan muncul peringatan untuk mengakhiri ujian dan pengerjaan ujian pun berakhir.



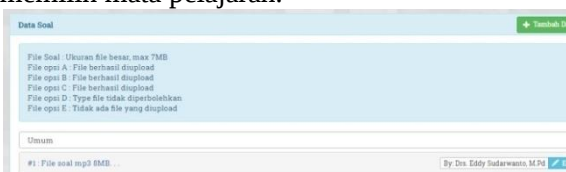
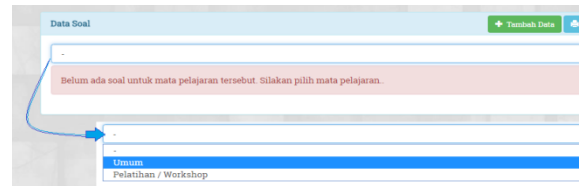
Gambar 8. Antarmuka halaman pengerjaan

Halaman tambah/edit soal yang digunakan oleh user yang berperan sebagai admin dan mentor terdapat beberapa isian dengan terdapat ketentuan *type* dan ukuran *file* dalam pengunggahan. Berikut halaman tersebut ditampilkan pada Gambar 9.



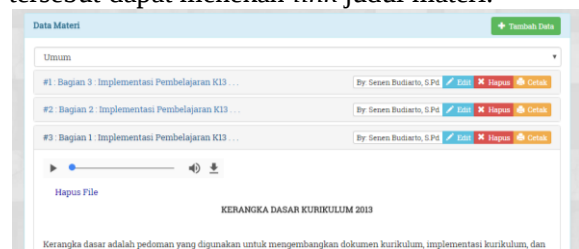
Gambar 9. Antarmuka halaman tambah/edit soal

Setelah proses tambah atau edit data soal (menekan tombol simpan) dilakukan maka akan di-redirect ke halaman daftar soal dari mata pelajaran yang baru ditambahkan dengan menampilkan pesan dalam pengunggahan *file*, baik itu keberhasilan ataupun kegagalan pengunggahan seperti ditunjukkan pada Gambar 10. Selanjutnya pada Gambar 11 ditampilkan daftar soal menurut mata pelajaran. Jika belum terdapat soal untuk mata pelajaran tersebut atau item mata pelajaran tidak dipilih maka akan ditampilkan pesan bahwa belum ada soal untuk mata pelajaran dan dianjurkan untuk memilih mata pelajaran.

Gambar 10. Antarmuka pesan pengunggahan *file* pada daftar soal

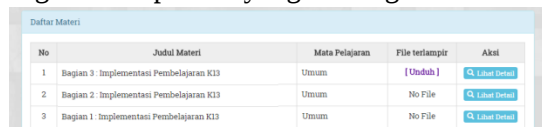
Gambar 11. Antarmuka halaman awal daftar soal

Sama halnya pada halaman daftar soal, halaman daftar materi untuk admin dan mentor juga ditampilkan berdasarkan mata pelajaran yang dipilih, dan untuk melihat detail materi tersebut dapat menekan *link* judul materi.



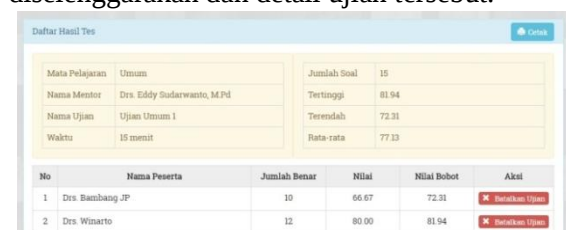
Gambar 12. Antarmuka daftar detail materi admin dan mentor

Sedangkan untuk peserta, seluruh daftar materi ditampilkan menurut mata pelajaran yang diambil peserta yang bersangkutan.



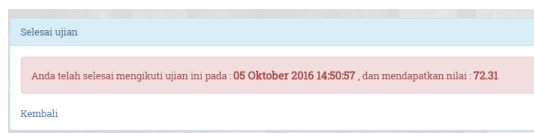
Gambar 13. Antarmuka daftar materi peserta

Pada implementasi halaman nilai, terdapat 2 jenis halaman yang berbeda antara halaman untuk admin dan mentor serta untuk peserta. Gambar 14 merupakan tampilan halaman nilai untuk admin dan mentor yang menampilkan daftar nilai peserta terhadap ujian yang telah diselenggarakan dan detail ujian tersebut.



Gambar 14. Antarmuka halaman nilai admin dan mentor

Untuk peserta, tampilan halaman nilai ditunjukkan pada Gambar 15 yang memuat waktu selesainya dilakukan ujian serta nilai ujian peserta tersebut. Tombol kembali digunakan untuk kembali ke halaman daftar ujian peserta berdasarkan mata pelajaran yang diambilnya.



Gambar 15. Antarmuka halaman nilai peserta

8. ANALISIS HASIL PENGUJIAN

Pengujian perangkat lunak dilakukan dengan menggunakan pengujian *black box* berupa pengujian fungsional, pengujian *white box* berupa pengujian basis path pada 3 fungsi utama sistem perangkat lunak, serta pengujian *usability* menggunakan metode kuisioner SUS (*System Usability Scale*).

8.1 Analisis Pengujian *Black Box*

Pada tahap pengujian *black box* dilakukan pengujian terhadap seluruh fungsi sistem perangkat lunak yang berjumlah 52 fungsi dan terbagi ke dalam total 161 kasus uji dimana setiap fungsi memiliki jumlah *alternatif flow* yang berbeda-beda. Kasus uji tersebut diantaranya adalah mencoba memberikan masukan dengan data yang benar, memberikan masukan dengan data yang salah, ataupun tidak memberikan masukan apapun ke dalam sistem. Hasil dari pengujian tersebut untuk seluruh kasus uji adalah menghasilkan hasil/keluaran sesuai dengan yang diharapkan sehingga dapat dikatakan bahwa seluruh fungsi dalam sistem yang dikembangkan telah teraplikasikan dan bekerja dengan baik (*valid*).

8.2 Analisis Pengujian *White Box*

Pada tahap pengujian *white box* ditampilkan pengujian terhadap 3 fungsi sistem perangkat lunak yaitu ikuti ujian, tambah data soal, serta tentukan mapel peserta. Hasilnya diperoleh data untuk fungsi ikuti ujian memiliki nilai kompleksitas 11 dengan jumlah *edge* 32 dan *nodes* 23 memperlihatkan bahwa fungsi ini cukup sulit untuk dikelola. Sementara fungsi tambah data soal nilai kompleksitas 10 dengan jumlah *edge* 24 dan *nodes* 16, serta fungsi tentukan mapel peserta nilai kompleksitas 7 dengan *edge* 20 dan *nodes* 15 menunjukkan bahwa kedua fungsi ini mudah untuk dikelola.

Ketiga fungsi kompleks sistem perangkat lunak memiliki kompleksitas di bawah 21 dimana program mudah dalam pengelolaan. Seperti halnya yang dikemukakan Bray (1997) bahwa apabila nilai kompleksitas memiliki nilai sebesar 21 hingga 50 mengindikasikan bahwa

fungsi tersebut perlu dilakukan refaktor/desain ulang karena sulit dikelola dan program tersebut memiliki tingkat resiko yang tinggi.

8.3 Analisis Pengujian *Usability*

Pada tahap pengujian *usability* dilakukan pengujian dengan menggunakan metode SUS (*System Usability Scale*) yang melibatkan 20 responden calon pengguna sistem perangkat lunak dan 8 kegiatan pengujian yang harus diselesaikan. Hasilnya adalah rata-rata dari skor SUS yang didapatkan dari penjumlahan seluruh skor SUS dari seluruh responden dibagi dengan jumlah responden menunjuk angka 84,17 yang mengindikasikan bahwa sistem perangkat lunak yang dibangun termasuk dalam kategori “Good” dan bernilai *adjective* “B”.

9. KESIMPULAN

Berdasarkan hasil pengamatan selama proses penelitian mulai dari analisis kebutuhan, perancangan, implementasi, hingga pada proses pengujian dan analisis sistem, diperoleh kesimpulan bahwa sistem perangkat lunak yang dikembangkan dengan memanfaatkan model proses *waterfall* yang dimulai dari tahap analisis kebutuhan sesuai penjelasan sub-bab analisis kebutuhan sistem pada bab 4 penulisan penelitian, sistem ini dikembangkan dengan 4 (empat) aktor yang terlibat dengan total 52 fungsi. Dilanjutkan tahap perancangan yang dijelaskan pada sub-bab perancangan sistem bab 4 penulisan penelitian dimana terdapat perancangan antarmuka sistem serta 4 (empat) desain/perancangan diagram pengembangan sistem perangkat lunak yaitu diagram aktifitas, diagram *sequence*, diagram kelas, dan diagram entitas. Selanjutnya tahap implementasi sesuai dengan pembahasan implementasi pada bab 5 penulisan penelitian dimana ditampilkan serta dijelaskan beberapa hasil implementasi antarmuka sistem perangkat lunak dan 4 (empat) potongan kode program dari fungsi yang tersedia. Terakhir pada tahap integrasi dan pengujian sistem sesuai dengan penjelasan bab 6 penulisan penelitian dilakukan 3 (tiga) jenis pengujian sistem yaitu *black box testing*, *white box testing*, dan *usability testing*.

Kedua, setelah dilakukan pengembangan sistem perangkat lunak dengan memanfaatkan kerangka kerja Codeigniter sesuai dengan hasil rancangan sistem perangkat lunak pada bab 4 penulisan penelitian dimana struktur dari rancangan tersebut telah disesuaikan dengan

struktur kerangka kerja Codeigniter, penulis berpendapat bahwa proses implementasi kode program menjadi lebih mudah dan cepat dikarenakan sudah adanya modul MVC (*Model-View-Controller*) pada kerangka kerja Codeigniter sehingga penelusuran kesalahan (*error*) yang terjadi pada proses implementasi dapat dilakukan dengan cepat dan mudah.

Terakhir pengujian *usability* menggunakan metode *System Usability Scale* (SUS) yang dilakukan kepada 20 responden calon pengguna sistem perangkat lunak dengan menyelesaikan 8 kegiatan pengujian didapatkan hasil rata-rata skor SUS dari seluruh responden sebesar 84,17 yang menunjukkan bahwa sistem perangkat lunak yang dibangun termasuk dalam kategori “Good” dan bernilai *adjective* “B”.

10. DAFTAR PUSTAKA

- Solichin, Mujiyanto., 2015. *Implementasi Kebijakan Pendidikan dan Peran Birokrasi*. Jombang : Universitas Pesantren Tinggi Darul Ulum Jombang.
- Rofieq, Ainur., 2008. *Teknik Pemberian Skor dan Nilai Hasil Tes*. Jakarta : Depdiknas.
- Sommerville, Ian., 2007. *Software Engineering*. 8th ed. Harlow: Pearson Education Limited.
- Brooke, John., 2013. SUS: A Retrospective. [pdf] Tersedia melalui: <http://uxpajournal.org/wp-content/uploads/pdf/JUS_Brooke_February_2013.pdf> [Diakses 18 April 2016]
- Bangor, dkk., 2009. Determining What Individual SUS Scores Mean: Adding an Adjective Rating Scale. [pdf] Tersedia melalui: <http://uxpajournal.org/wp-content/uploads/pdf/JUS_Bangor_May2009.pdf> [Diakses 18 April 2016]
- Bray, Michael., dkk., 1997. *C4 Software Technology Reference Guide - A Prototype*. Pennsylvania: Carnaige Mellon University.