

PROTOTIPE APLIKASI UJIAN CODING PHP BERBASIS WEB INTRANET DI STMIK ATMA LUHUR

Yohanes Setiawan Japriadi¹; Lukas Tommy²

Program Studi Teknik Informatika^{1,2}

STMIK Atma Luhur^{1,2}

www.atmaluhur.ac.id^{1,2}

ysetiawanj@atmaluhur.ac.id¹; lukastommy@atmaluhur.ac.id²

Abstract—Modernization causes the examination process in some educational institutions that are usually done conventionally now start to change to CBT (Computer Based Test). At STMIK Atma Luhur, CBT in the form of a coding test carried out on a computer laboratory was felt to be not effective enough due to a short correction time, as well as the possibility of losing the students work due to sudden power failure. Collection of jobs using an LAN (Local Area Network) intranet network can not be realized due to the difficult network configuration and requires not a small cost. One solution that can be offered in overcoming the above problem is to create a web-based exam application that is connected to a Wireless Local Area Network (WLAN) intranet network. The system development model and tools used in this study are prototype models and UML (Unified Modeling Language). The proposed application is effective in facilitating and accelerating the process of collecting student answer files from several client computers to a server computer. In addition, the usage of the proposed application allows the lecturer not to correct students' answers in a hurry, so that the scores that are truly objective and can be accounted for are obtained.

Keywords: CBT, centralized data communication, coding test, intranet, WLAN.

Intisari—Modernisasi menyebabkan proses ujian di beberapa lembaga pendidikan yang biasanya dilakukan secara konvensional kini mulai berganti menjadi CBT (Computer Based Test). Pada STMIK Atma Luhur, CBT yang berupa ujian coding yang dilaksanakan pada laboratorium komputer dirasakan belum cukup efektif akibat dari waktu koreksi yang singkat, sekaligus kemungkinan hilangnya pekerjaan mahasiswa akibat mati listrik yang tiba-tiba. Pengumpulan pekerjaan menggunakan jaringan intranet LAN (Local Area Network) belum dapat direalisasikan akibat konfigurasi jaringannya yang sulit dan membutuhkan biaya yang tidak sedikit. Salah satu solusi yang dapat ditawarkan dalam mengatasi masalah di atas adalah dengan membuat aplikasi

ujian berbasis web yang terhubung dengan jaringan intranet WLAN (Wireless Local Area Network). Model dan alat bantu pengembangan sistem yang digunakan dalam penelitian ini adalah model prototipe dan UML (Unified Modelling Language). Aplikasi yang diusulkan efektif dalam memudahkan dan mempercepat proses pengumpulan file jawaban mahasiswa dari beberapa komputer klien ke sebuah komputer server. Selain itu, pemanfaatan aplikasi usulan memungkinkan dosen pengampu untuk tidak mengoreksi jawaban mahasiswa secara terburu-buru, sehingga diperoleh nilai yang benar-benar objektif dan dapat dipertanggung jawabkan.

Kata Kunci: CBT, intranet, komunikasi data terpusat, ujian koding, WLAN.

PENDAHULUAN

Semakin pesatnya perkembangan teknologi informasi di dunia, tak terkecuali di Indonesia mempengaruhi berbagai aspek kehidupan masyarakat menjadi lebih modern. Pendidikan adalah salah satu contoh aspek yang mengalami modernisasi, dimana proses ujian di beberapa lembaga pendidikan yang biasanya dilakukan secara konvensional kini mulai berganti menjadi CBT (Computer Based Test) (Kelen & Belalawe, 2018).

STMIK Atma Luhur hingga laporan ini dibuat belum sepenuhnya menerapkan CBT untuk kegiatan akademiknya, dimana CBT hanya diterapkan untuk menguji calon mahasiswa baru yang mendaftar serta ujian mata kuliah bidang komputer. Hal ini menyebabkan kurang efisiennya biaya ATK (Alat Tulis Kantor) (Riyadi, Hermaliani, & Utami, 2019) yang dikeluarkan oleh pihak kampus. Selain itu, CBT berupa ujian coding yang dilakukan di laboratorium komputer dirasakan belum cukup efektif akibat waktu koreksi yang singkat (Perkasa, Saputra, & Fronita, 2015; Riyadi et al., 2019). Masalah ini telah dirasakan oleh beberapa dosen pengampu mata kuliah komputer, salah satunya mata kuliah Algoritma dan Struktur Data yang diajarkan oleh peneliti dengan

menggunakan bahasa pemrograman PHP: *Hypertext Preprocessor*.

Pihak laboratorium komputer sebelum ini memang telah berinisiatif untuk mengumpulkan pekerjaan mahasiswa dari berbagai komputer ke sebuah *flash disk*, namun hal ini memakan waktu dan rentan terhadap serangan *malware* (Widijuanarto, 2012). Pengumpulan pekerjaan menggunakan jaringan *intranet* LAN (*Local Area Network*) belum dapat direalisasikan akibat konfigurasi jaringannya yang sulit dan membutuhkan biaya yang tidak sedikit (Ariawal & Purbo, 2016) dimana keempat laboratorium yang ada memiliki paling tidak 45 buah komputer.

Salah satu solusi yang dapat ditawarkan dalam mengatasi masalah di atas adalah dengan membuat aplikasi ujian berbasis *web* yang terhubung dengan jaringan *intranet* WLAN (*Wireless Local Area Network*). Pemanfaatan jaringan *intranet* WLAN ini memungkinkan aplikasi dapat berjalan di setiap komputer ataupun laptop yang terpasang *Wifi Card*, dan terkoneksi dalam dalam satu jaringan yang sama meskipun tidak terhubung dengan jaringan *internet* (Ariawal & Purbo, 2016). Terdapat soal ujian berupa teks dan gambar dari dosen pengampu yang ditampilkan pada aplikasi yang akan dibuat, sehingga mahasiswa tidak lagi membutuhkan *print out* soal, akibatnya biaya kertas dan percetakan yang dikeluarkan pihak kampus menjadi berkurang. Selain itu terdapat pula *editor* sekaligus *compiler* PHP yang tertanam di aplikasi yang akan dibuat, sehingga meringankan pekerjaan pihak laboratorium karena tidak membutuhkan instalasi aplikasi *editor* atau *compiler* PHP eksternal. Setelah di-*compile*, *file coding* dengan ekstensi *.php* akan disimpan ke komputer *server* pada *folder* baru bernama NIM (Nomor Induk Mahasiswa) mahasiswa tersebut untuk nantinya dikoreksi oleh dosen pengampu. *Smartphone* Android tidak dipilih sebagai *platform*, karena meskipun populer, praktis, dan bermobilitas tinggi (Tommy, Kirana, & Lindawati, 2019), *keyboard virtual*-nya kurang nyaman jika digunakan dalam meng-*input* data dalam jumlah besar dibandingkan dengan *keyboard* pada komputer.

Penelitian mengenai pembuatan sistem ujian *online* yang terkait dengan penelitian yang akan dilakukan telah pernah dilakukan sebelumnya. Prototipe sistem ujian seleksi SIP (Sekolah Inspektur Polisi) *online* dapat menggantikan sistem ujian berbasis kertas dan dapat diakses oleh peserta menggunakan jaringan *intranet*. Pengacakan soal pada basis data membuat soal-soal ujian yang keluar dapat bervariasi, dinamis, tidak mudah dihapal, dan mencegah peserta untuk saling mencontek

sehingga transparansi pada nilai yang diperoleh dapat tercapai (Y. Setiawan, 2016).

Sistem ujian online berbayar menggunakan metode *Technology Acceptance Model* (TAM) dan Webqual sudah bisa diterima dikarenakan ada empat hipotesis yang menunjukkan pengaruh yang positif antar konstruk penerimaan dan dua hipotesis yang menunjukkan pengaruh negatif (E. Setiawan, Antoni, & Mirza, 2019). Sistem informasi ujian *online* memberikan kemudahan di dalam pelaksanaan dan pengolahan data ujian bagi siswa maupun guru, sekaligus menjadi pilihan terbaik untuk menekan berbagai persoalan dan hambatan dari pelaksanaan ujian manual (Susilowati & Hidayat, 2018).

Penerapan aplikasi ujian berbasis komputer meningkatkan kualitas pelaksanaan ujian yang dapat mengatasi faktor kecurangan siswa khususnya mencontek dalam proses ujian, meningkatkan objektivitas guru dalam penilaian, penghematan biaya dalam proses pembuatan soal/penggandaan soal dan lembar jawaban, mempercepat proses pemeriksaan dan penilaian hasil ujian (Sampebua & Membala, 2018). Aplikasi ujian *online* mengurangi biaya operasional dan memberikan kemudahan bagi siswa serta guru untuk melakukan penilaian tanpa buku. Selain itu, proses ujian menjadi lebih cepat dan efisien, meminimalisir kecurangan dalam proses ujian, dan siswa tidak perlu menunggu lama untuk mengetahui nilai mereka karena nilai akan muncul secara *realtime* (Riyadi et al., 2019).

Implementasi *Model-View-Controller* (MVC) di ujian *online* mempermudah guru dan siswa dalam melaksanakan ujian yang fleksibel. Proses tatap muka di kelas menjadi lebih efektif dan guru dapat mengetahui kemampuan siswa dengan kuis yang bisa dikerjakan dari rumah (Kelen & Belalawe, 2018). Implementasi algoritma *fisher-yates* untuk mengacak soal ujian *online* penerimaan mahasiswa baru membuat mahasiswa dalam menjawab soal memiliki nomor yang sama tetapi bentuk soal yang berbeda sehingga mengurangi tindakan kecurangan dalam proses pelaksanaan ujian penerimaan mahasiswa baru (Hasan, Supriadi, & Zamzami, 2017).

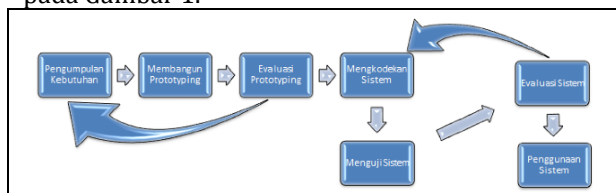
Pemanfaatan *Latent Semantic Analysis* (LSA) dalam penilaian sistem ujian *online essay* memiliki pengukuran kedekatan antara kunci jawaban dengan *vector* jawaban menggunakan *cosine similarity*. Sistem meningkatkan validitas hasil ujian dengan menampilkan soal secara acak tiap mahasiswa berdasarkan paket soal yang dibuat. Kesalahan pengetikan saat meng-*input*-kan jawaban juga sangat berpengaruh terhadap hasil akhir dari sebuah ujian (Perkasa et al., 2015). Algoritma *Linear Congruent Method* (LCM) dapat diterapkan pada sistem ujian *online* seleksi

penerimaan mahasiswa baru. Penentuan konstanta pada LCM sangat menentukan baik tidaknya bilangan acak yang diperoleh dalam arti memperoleh bilangan acak yang seakan-akan tidak akan terjadi pengulangan (Gunawan & Prabowo, 2017).

BAHAN DAN METODE

A. Model Pengembangan Sistem

Model prototipe dipilih sebagai model pengembangan perangkat lunak pada penelitian ini, yaitu salah satu metode siklus hidup sistem yang berlandaskan pada model konsep bergerak demi membangun model ke bentuk sistem akhir. Tahap-tahap pada model prototipe dapat dilihat pada Gambar 1.



Sumber: (Bastian & Budiman, 2017)

Gambar 1. Tahap-Tahap Model Prototipe

1. Model Pengembangan Sistem

Teknik pengumpulan data yang digunakan adalah observasi dan wawancara, dimana dilakukan pengamatan secara langsung terhadap proses ujian praktek mata kuliah yang dilaksanakan pada laboratorium komputer. Selain itu, diadakan juga wawancara terhadap rekan-rekan dosen yang pernah melaksanakan ujian serupa di laboratorium komputer. Berdasarkan pengumpulan kebutuhan yang telah dilakukan, diketahui masalah yang ada adalah sulitnya mengoreksi dengan teliti dan rinci terhadap ujian yang dilaksanakan oleh mahasiswa mata kuliah praktek di laboratorium komputer. Contoh mata kuliah praktek ini adalah mata kuliah yang menggunakan bahasa pemrograman PHP seperti desain dan pemrograman web. Masalah ini diakibatkan waktu yang disediakan oleh pihak kampus, yaitu selama 30 menit dirasakan tidak cukup. Kurang banyaknya jumlah laboratorium komputer merupakan alasan waktu yang singkat ini, sehingga sesaat setelah sebuah ujian mata kuliah berakhir, ujian mata kuliah berikutnya akan segera dimulai. Pindahan hasil ujian mahasiswa ke sebuah flash disk juga dirasakan kurang optimal dikarenakan memakan waktu, kemungkinan tidak ikut ter-copy, dan rentan terhadap serangan malware. Implementasi konfigurasi jaringan intranet dengan kabel LAN cukup sulit dan membutuhkan biaya yang cukup besar mengingat jumlah komputer di laboratorium cukup banyak untuk dihubungkan dengan kabel LAN dan router,

yakni 45 komputer per ruangan. Selain masalah-masalah yang telah dijelaskan, ujian berbasis komputer yang menyimpan hasil ujiannya di komputer itu sendiri (lokal) rentan akan kecurangan, misalkan mahasiswa yang melihat komputer di dekatnya, bahkan meng-copy hasil ujian tersebut dengan flash disk.

Salah satu solusi yang dirasakan cukup efektif untuk mengatasi permasalahan-permasalahan tersebut adalah dengan membuat prototipe aplikasi ujian coding berbasis web yang terhubung pada jaringan intranet WLAN dimana penyimpanan hasil ujian pesertanya adalah di komputer pusat (server). Aplikasi ini nantinya dapat digunakan oleh para dosen untuk meminimalisir kecurangan dan menampung hasil ujian mahasiswa mata kuliah yang diampu mereka, dimana hasil ujian tersebut dapat dikoreksi oleh dosen yang bersangkutan dengan tidak terburu-buru

2. Membangun *Prototyping*

Pada tahap ini akan dilaksanakan desain tampilan halaman sekaligus fitur-fitur pada aplikasi. Selain itu, di tahap ini juga akan didesain skema jaringan serta kebutuhan perangkat keras dan lunak dari aplikasi yang akan dibuat agar dapat berjalan dengan baik.

3. Evaluasi *Prototyping*

Prototipe dari tahap membangun prototyping selanjutnya dievaluasi oleh para *stakeholders* STMIK Atma Luhur, yaitu bagian prodi (program studi), bagian lab komputer, dosen, mahasiswa, dan pihak lain yang berkaitan dengan aplikasi yang akan dibuat. Faktor yang dievaluasi antara lain dalam hal antarmuka, skema jaringan, spesifikasi *hardware* dan *software* yang dibutuhkan, sekaligus fungsi-fungsi yang nantinya akan dimiliki oleh aplikasi yang dibuat. Pengembang aplikasi akan kembali melakukan pengumpulan kebutuhan dengan umpan balik dari para *stakeholders* apabila prototipe tidak disetujui.

4. Mengkodekan Sistem

Pada tahap ini, rancangan halaman dan fungsi pada prototipe diterapkan menjadi aplikasi dengan fungsi yang berjalan seutuhnya. Proses desain tampilan dan *coding* menggunakan Adobe Dreamweaver dan *Sublime Text Editor* dengan bahasa pemrograman PHP dan *library/framework* Bootstrap.

5. Menguji Sistem

Model pengujian *blackbox* dipilih untuk menguji fungsi-fungsi (masukan dan keluaran) yang dimiliki aplikasi.

6. Mengevaluasi Sistem

Aplikasi yang melalui pengujian *blackbox* dan dinyatakan sesuai akan diuji sekali lagi oleh para *stakeholders* apakah sudah memenuhi kebutuhan sebelum diimplementasikan di laboratorium komputer. Pengembang akan mengulangi tahap mengkodekan sistem dengan berbekal umpan balik yang diberikan seandainya aplikasi tidak sesuai dengan keinginan para *stakeholders*.

7. Penggunaan Sistem

Aplikasi yang telah dievaluasi dan dinyatakan lulus oleh para *stakeholders* selanjutnya diimplementasikan pada laboratorium komputer dan disosialisasikan kepada dosen-dosen pengampu mata kuliah yang diajarkan di laboratorium komputer.

B. Tools Pengembangan Sistem

UML (*Unified Modelling Language*) merupakan alat bantu pengembangan sistem yang digunakan dalam membuat aplikasi ini. UML merupakan bahasa resmi untuk memodelkan proses bisnis dan perangkat lunak yang umumnya digunakan di pendekatan berorientasi objek. Diagram *use case*, diagram *activity*, dan diagram *sequence* merupakan diagram-diagram UML yang digunakan dalam membuat aplikasi ini. Prototipe aplikasi yang dibuat hanya menggunakan *database MySQL* untuk menyimpan *username* dan *password* untuk keperluan autentifikasi mahasiswa sehingga *class diagram* dianggap belum perlu ditambahkan untuk mendeskripsikan *database* dari aplikasi.

HASIL DAN PEMBAHASAN

A. Analisis Kebutuhan

Kebutuhan dari aplikasi yang nantinya dibuat didefinisikan melalui proses analisis kebutuhan. Kebutuhan fungsional dan kebutuhan non-fungsional merupakan pengelompokan dari analisis kebutuhan.

1. Model Pengembangan Sistem

Aplikasi ujian *coding PHP* yang akan dibangun memiliki kebutuhan-kebutuhan fungsional seperti berikut:

- Aplikasi yang dibuat dapat dioperasikan oleh para dosen untuk manajemen soal ujian yang ditampilkan di aplikasi *web* dan mengakses *file PHP* hasil ujian mahasiswa yang disimpan pada media penyimpanan internal komputer *server*.
- Aplikasi yang dibuat dapat dioperasikan oleh mahasiswa untuk *login* ke aplikasi *web*, melihat soal, menjawab soal tersebut dengan

koding, sekaligus memeriksa berikut menyimpan hasil kompilasi kodingnya menggunakan *editor* dan *compiler PHP embedded*.

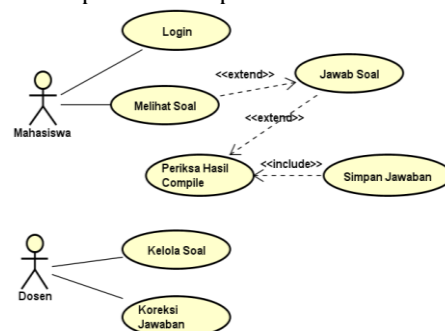
2. Analisa Kebutuhan Non-Fungsional

Aplikasi ujian *coding PHP* yang akan dibangun memiliki kebutuhan-kebutuhan non-fungsional yang berupa spesifikasi perangkat keras dan perangkat lunak seperti berikut:

- Server*: Satu buah laptop Asus X45A-VX058D dengan *processor Intel Celeron dual core 1000M*, RAM DDR3 4GB, WiFi, *harddisk 500GB* (minimal kapasitas tersisa 1 GB), sistem operasi Windows 7 Ultimate 64 Bits, dan XAMPP versi 1.7.4 sebagai DBMS (*Database Management System*) *database MySQL* dan *web server* lokal.
- Klien: 20 buah laptop Fujitsu LH531 (atau laptop pribadi mahasiswa) *processor Intel dual core B970*, RAM DDR3 6GB dengan WiFi di laboratorium komputer 2 (Lab 2), Windows 7 Professional 64 Bits dan *browser Mozilla Firefox*.
- Jaringan: 1 buah kabel RJ45 dengan panjang 1 meter untuk menghubungkan *WiFi router Tenda N300* laju transmisi 300 Mbps dengan komputer *server*.

3. Use Case Diagram

Relasi yang terjadi antara aktor dan sistem yang ada diuraikan dengan *use case diagram*. *Use case diagram* dari aplikasi ujian koding yang diusulkan diperlihatkan pada Gambar 2.



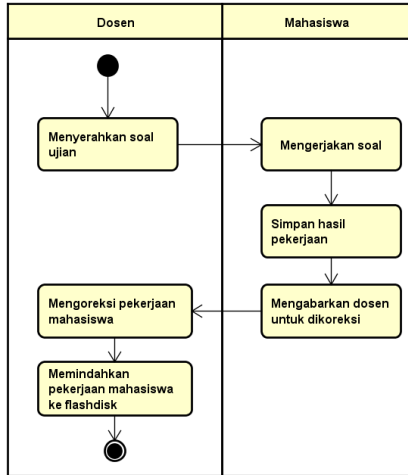
Sumber: (Japriadi & Tommy, 2019)

Gambar 2. Use Case Diagram Aplikasi Usulan

4. Analisis Sistem Berjalan

Ujian koding berbasis komputer di laboratorium komputer STMIK Atma Luhur sampai dengan penelitian ini dibuat masih dilakukan secara konvensional. Dosen mula-mula menyerahkan kertas soal ujian koding kepada mahasiswa, kemudian mahasiswa menjawab soal tersebut dengan melakukan koding dan menyimpan jawabannya ke dalam suatu *file* dengan ekstensi tertentu, misalkan *.php*.

Mahasiswa mengabarkan kepada dosen pengampu untuk mengoreksi hasil ujiannya dan kemudian dosen langsung mengoreksinya ataupun memindahkannya terlebih dahulu ke *flash disk*. *Activity diagram* sistem berjalan dapat dilihat pada Gambar 3.



Sumber: (Japriadi & Tommy, 2019)
Gambar 3. *Activity Diagram* Sistem Berjalan

B. Rancangan Sistem

Rancangan yang dibuat dalam membangun aplikasi ujian koding, yaitu representasi tingkah laku aplikasi menggunakan *activity diagram* dan *sequence diagram*. Selain itu, dirancang juga skema jaringan dari aplikasi yang dibuat sehingga komputer *server* dan klien dapat terkoneksi.

1. Activity Diagram

Beberapa aliran fungsi dari aplikasi yang dibuat dideskripsikan menggunakan *activity diagram*. *Activity diagram* aplikasi yang dibuat dengan mahasiswa dan dosen sebagai *user* ditunjukkan Gambar 4 dan Gambar 5.

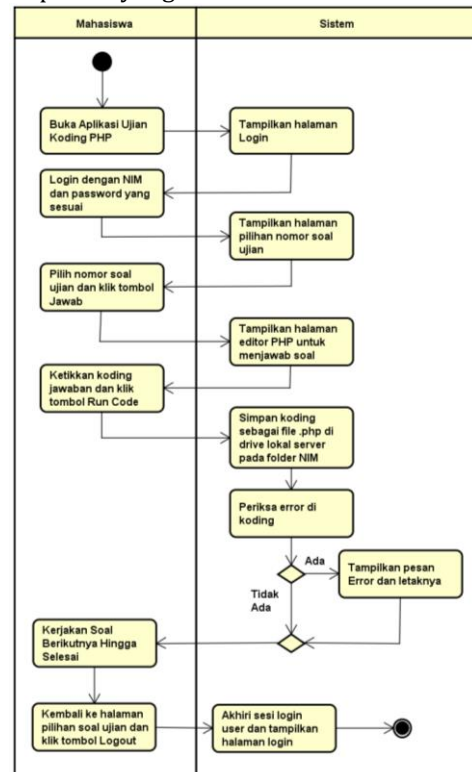
2. Sequence Diagram

Sequence diagram menggambarkan aliran fungsi di *use case* dan mendeskripsikan rangkaian keterkaitan objek yang disusun pada rangkaian periode. *Sequence diagram* aplikasi yang dibuat dengan mahasiswa dan dosen sebagai *user* ditunjukkan gambar 6 dan gambar 7.

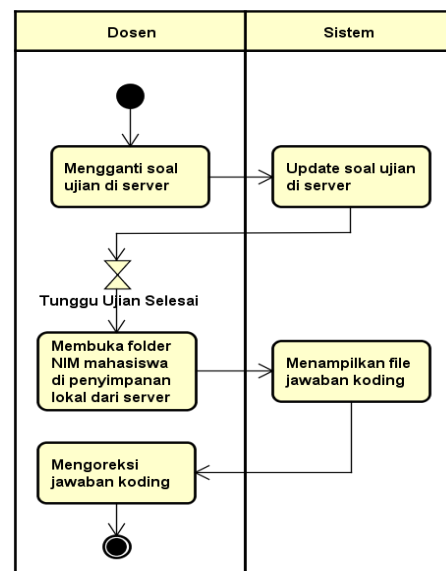
3. Skema Jaringan Aplikasi Usulan

Skema jaringan dari aplikasi yang dibuat ditunjukkan seperti Gambar 8, dimana WiFi *router* memiliki alamat IP (*Internet Protocol*) 192.168.1.1, laptop dosen (*server*) alamat IP-nya 192.168.1.2. Alamat IP dari komputer/ laptop mahasiswa ditentukan otomatis, dimana dalam satu sesi ujian ada 20 mahasiswa yang mengikutinya, yakni 192.168.1.3 sampai dengan 192.168.1.22. Pembatasan jumlah mahasiswa yang mengikuti ujian dalam satu waktu ini dimaksudkan agar tidak

terjadi *traffic* atau melambatnya performa dari *web server* aplikasi yang diusulkan.



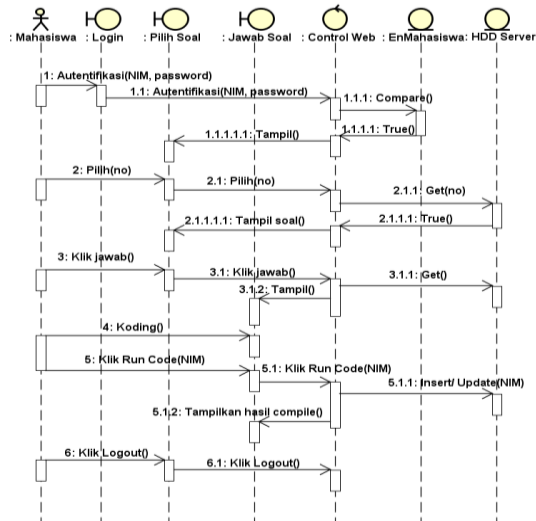
Sumber: (Japriadi & Tommy, 2019)
Gambar 4. *Activity Diagram* Aplikasi Usulan User Mahasiswa



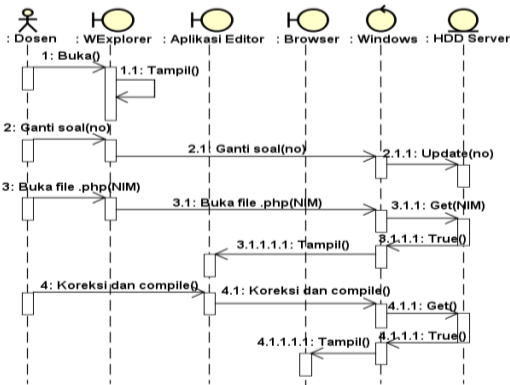
Sumber: (Japriadi & Tommy, 2019)
Gambar 5. *Activity Diagram* Aplikasi Usulan User Dosen

C. Implementasi Sistem

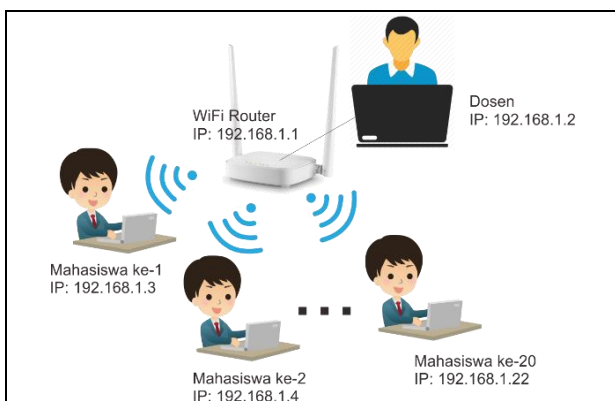
Implementasi sistem yang sudah dibuat pada tahap rancangan sistem nantinya akan dilaksanakan di tahap implementasi sehingga dapat digunakan.



Sumber: (Japriadi & Tommy, 2019)
Gambar 6. Sequence Diagram Aplikasi Usulan User Mahasiswa



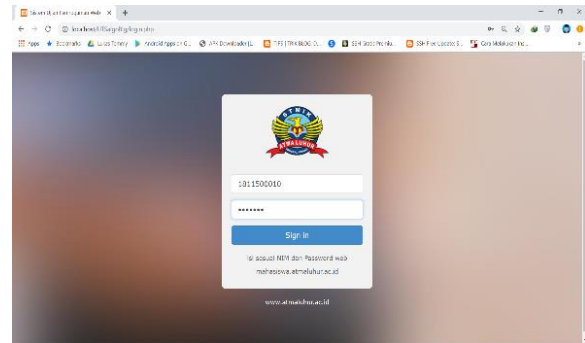
Sumber: (Japriadi & Tommy, 2019)
Gambar 7. Sequence Diagram Aplikasi Usulan User Dosen



Sumber: (Japriadi & Tommy, 2019)
Gambar 8. Skema Jaringan Aplikasi Usulan

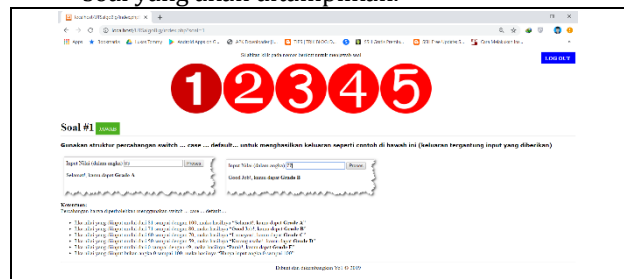
1. Tampilan Layar
Beberapa tampilan layar dari aplikasi yang dibuat adalah sebagai berikut:

a. Tampilan layar halaman login mahasiswa, yang ditunjukkan seperti Gambar 9.



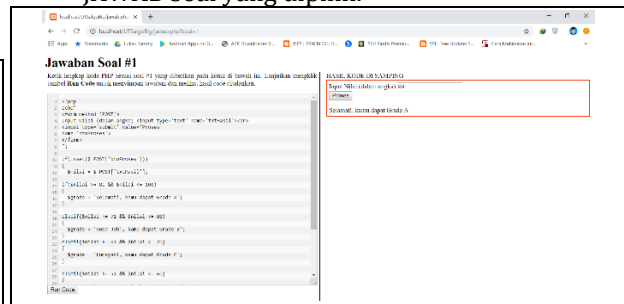
Sumber: (Japriadi & Tommy, 2019)
Gambar 9. Tampilan Layar Halaman Login Mahasiswa

b. Tampilan layar halaman pilih soal, yang ditunjukkan seperti Gambar 10. Pada prototipe aplikasi ujian koding ini, hanya 5 soal yang akan ditampilkan.



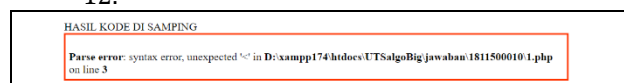
Sumber: (Japriadi & Tommy, 2019)
Gambar 10. Tampilan Layar Halaman Pilih Soal

c. Tampilan layar halaman hasil running jawaban, yang ditunjukkan Gambar 11. Baris koding di halaman ini akan kosong ketika mahasiswa tersebut pertama mengklik JAWAB soal yang dipilih.



Sumber: (Japriadi & Tommy, 2019)
Gambar 11. Tampilan Layar Halaman Hasil Running Jawaban

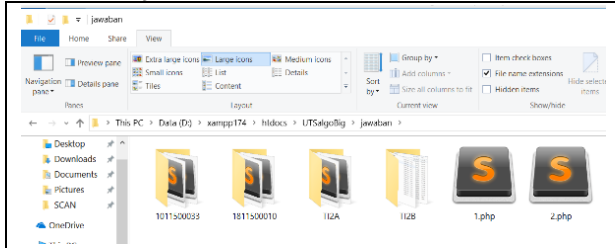
d. Tampilan layar halaman hasil running jawaban jika error, yang ditunjukkan Gambar 12.



Sumber: (Japriadi & Tommy, 2019)
Gambar 12. Tampilan Layar Halaman Hasil Running Jawaban Jika Error

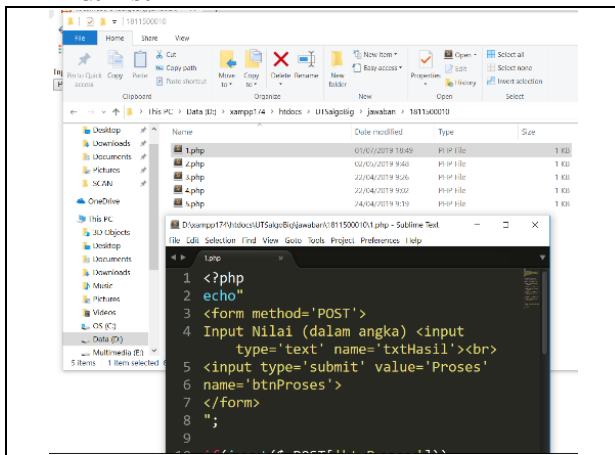
Terdapat juga beberapa tampilan layar yang bukan bagian dari aplikasi yang dibuat, namun masih memiliki keterkaitan dengannya, yaitu sebagai berikut:

1. Tampilan layar lokasi penyimpanan *folder* jawaban mahasiswa di *server*, yang ditunjukkan Gambar 13. *Folder* ini berisikan 5 buah *file* jawaban mahasiswa dengan NIM nama *folder*.



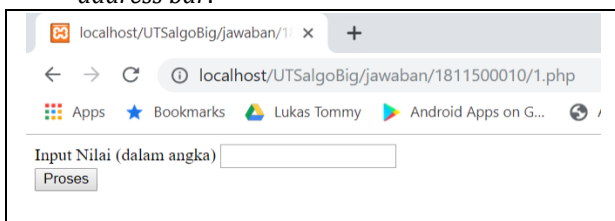
Sumber: (Japriadi & Tommy, 2019)
Gambar 13. Tampilan Layar Lokasi Penyimpanan *Folder* Jawaban Mahasiswa di *Server*

2. Tampilan layar isi *folder* dan *file* jawaban mahasiswa di *server*, yang ditunjukkan Gambar 14.



Sumber: (Japriadi & Tommy, 2019)
Gambar 14. Tampilan Layar Isi *Folder* dan *File* Jawaban Mahasiswa di *Server*

3. Tampilan layar hasil *compile file* jawaban pada *browser*, yang ditunjukkan Gambar 15. Layar ini juga ditampilkan jika ada mahasiswa yang berusaha mengakses jawaban mahasiswa lainnya menggunakan *browser* dan manipulasi *address bar*.



Sumber: (Japriadi & Tommy, 2019)
Gambar 15. Tampilan Layar Hasil *Compile* Jawaban pada *Browser*

2. Pengujian

Model pengujian yang digunakan dalam penelitian ini yaitu *black box* dari segi fungsionalitas aplikasi yang dibuat. Hasil pengujian *black box* terhadap aplikasi yang dibuat ditunjukkan Tabel 1.

Tabel 1. Hasil Pengujian *Black Box*

No	Pengujian	Hasil yang diharapkan	Keterangan
1	Login ke aplikasi dengan <i>username</i> atau <i>password</i> yang tak sesuai	Aplikasi menampilkan pesan "Login Gagal!"	Valid
2	Login ke aplikasi dengan <i>username</i> atau <i>password</i> yang sesuai	Aplikasi menampilkan halaman pilih soal	Valid
3	Klik nomor soal di halaman pilih soal	Aplikasi menampilkan soal yang dipilih	Valid
4	Klik tombol JAWAB di halaman pilih soal	Aplikasi menampilkan halaman jawab soal. Jika mahasiswa yang login telah mengerjakan soal ini sebelumnya, jawaban sebelumnya akan ditampilkan.	Valid
5	Ketik koding dengan tepat dan klik tombol Run Code	Menyimpan jawaban di <i>folder</i> bernama NIM mahasiswa yang login pada <i>harddisk</i> lokal milik <i>server</i> dan menampilkan hasil <i>compile</i> di sebelah kanan jendela koding	Valid
6	Ketik koding dengan kurang tepat dan klik tombol Run Code	Menyimpan jawaban di <i>folder</i> bernama NIM mahasiswa yang login pada <i>harddisk</i> lokal milik <i>server</i> dan menampilkan <i>error</i> di sebelah kanan jendela koding	Valid
7	Klik tombol LOG OUT di halaman pilih soal	Sesi <i>login</i> mahasiswa diakhiri dan sistem menampilkan halaman <i>login</i>	Valid

Sumber: (Japriadi & Tommy, 2019)

KESIMPULAN

Beberapa hal yang bisa disimpulkan berdasarkan uraian yang telah dibahas di paper ini adalah aplikasi yang diusulkan efektif dalam memudahkan dan mempercepat proses pengumpulan *file* jawaban mahasiswa dari beberapa komputer klien ke sebuah komputer *server*. Aplikasi usulan juga dapat digunakan

sebagai solusi alternatif dari jaringan *intranet* LAN dimana biaya pemasangan dan konfigurasinya tidak terlalu besar jika dibandingkan dengan LAN. Selain itu, pemanfaatan aplikasi usulan memungkinkan dosen pengampu untuk tidak mengoreksi jawaban mahasiswa secara terburu-buru, sehingga diperoleh nilai yang benar-benar objektif dan dapat dipertanggung jawabkan.

REFERENSI

- Ariawal, D., & Purbo, O. W. (2016). *Simulasi Jaringan Komputer dengan Cisco Packet Tracer*. Jakarta: PT Elex Media Komputindo.
- Bastian, A., & Budiman, A. Y. (2017). Implementasi Piranti Jaringan Repeater Eco Village Menggunakan Model Prototype dan Konsep Green Computing. *Jurnal Ilmiah Teknologi Informasi Terapan*, 3(3), 233-240.
- Gunawan, & Prabowo, D. A. (2017). Sistem Ujian Online Seleksi Penerimaan Mahasiswa Baru Dengan Pengacakan Soal Menggunakan Linear Congruent Method (Studi Kasus Di Universitas Muhammadiyah Bengkulu). *Jurnal Informatika UPGRIS*, 3(2), 143-151.
- Hasan, M. A., Supriadi, & Zamzami. (2017). Implementasi Algoritma Fisher-Yates untuk Mengacak Soal Ujian Online Penerimaan Mahasiswa Baru (Studi Kasus: Universitas Lancang Kuning Riau). *Jurnal Nasional Teknologi Dan Sistem Informasi*, 3(2), 291-298.
- Japriadi, Y. S., & Tommy, L. (2019). *Laporan Akhir Penelitian Prototipe Aplikasi Ujian Coding Php Berbasis Web Intranet di STMIK Atma Luhur*. Pangkalpinang.
- Kelen, Y. R. L., & Belalawe, B. J. (2018). Implementasi Model-View-Controller (MVC) pada Ujian Online Melalui Penerapan Framework Codeigniter. *Jurnal Pendidikan Teknologi Informasi (JUKANTI)*, 1(1), 10-16
- Perkasa, D. A., Saputra, E., & Fronita, M. (2015). Sistem Ujian Online Essay Dengan Penilaian Menggunakan Metode Latent Semantic Analysis (LSA). *Jurnal Rekayasa Dan Manajemen Sistem Informasi*, 1(1), 1-9.
- Riyadi, A., Hermaliani, E. H., & Utami, D. Y. (2019). Pembuatan Aplikasi Sistem Ujian Online pada Smk Garuda Nusantara Bekasi. *Jurnal Ilmiah Sinus (JIS)*, 17(1), 23-36.
- Sampebua, M. R., & Membala, S. B. (2018). Penerapan Aplikasi Ujian Berbasis Komputer untuk Meningkatkan Kualitas Pendidikan pada SMP. *Jurnal Pengabdian Papua*, 2(1), 24-31.
- Setiawan, E., Antoni, D., & Mirza, A. H. (2019). Analisis Penerimaan Sistem Ujian Online Berbayar dengan Menggunakan Metode Technology Acceptance Model (TAM) dan Webqual. *Jurnal Bina Komputer JBK*, 1(1), 61-72.
- Setiawan, Y. (2016). Prototipe Sistem Ujian Online dan Penilaian Jawaban Peserta Secara Real-Time. *Jurnal Teknologi Informatika Dan Komputer Atma Luhur*, 3(1), 66-74.
- Susilowati, S., & Hidayat, T. (2018). Rancang Bangun Sistem Informasi Ujian Online (Studi Kasus pada SMAN 58 Jakarta). *Jurnal Teknik Komputer*, 4(1), 30-387.
- Tommy, L., Kirana, C., & Lindawati, V. (2019). Recommender System dengan Kombinasi Apriori dan Content-Based Filtering pada Aplikasi Pemesanan Produk. *Journal Teknoinfo*, 13(2), 84-95.
- Widijanuarto, A. S. (2012). *101 Tip dan Trik Seputar USB Flash Drive*. Jakarta: Elex Media Komputindo.