

Design and Development of Computer-Based Examination Services Application in Higher Education Academic Portal

Rancang Bangun Aplikasi Layanan Ujian Berbasis Komputer di Portal Akademik Perguruan Tinggi

Gabriel F. Z. Sadu, Sary D.E. Paturusi, Alwin M. Sambul

Dept. of Electrical Engineering, Sam Ratulangi University Manado, Kampus Bahu St., 95115, Indonesia

e-mails : gabrielsadu026@student.unsrat.ac.id, sarypaturusi@unsrat.ac.id, asambul@unsrat.ac.id

Received: 18 May 2023; revised: 1 June 2023; accepted: 29 June 2023

Abstract — Sam Ratulangi University has several rooms equipped with computers that can be used for online exams but are rarely utilized. This study aims to develop a computer-based exam service application that provides the option for conducting exams in specific computer rooms. For the convenience of lecturers and students, the application can be integrated as a feature within the university's academic portal. Based on the research and development conducted, it can be concluded that the development of a computer-based exam service application in the Academic Portal of the University can provide numerous benefits. The MVC structure in the CodeIgniter Framework facilitates a more organized application development process. Additionally, the use of the Summernote Editor makes exam questions visually appealing and interactive. By integrating this application into the university's Academic Portal, it can support exam activities within the university and optimize the utilization of computer rooms that were previously underutilized.

Key words— Design and Development, Computer-Based Exam, Academic Portal, CodeIgniter 4.

Abstrak — Universitas Sam Ratulangi memiliki sejumlah ruangan berisikan komputer-komputer yang dapat digunakan untuk ujian secara online namun jarang dipakai. Penelitian ini bertujuan mengembangkan suatu aplikasi layanan ujian berbasis komputer yang memiliki opsi untuk pelaksanaan ujian di ruangan-ruangan komputer tertentu. Untuk kemudahan para Dosen dan Mahasiswa dalam mengaksesnya, maka aplikasi tersebut dapat diintegrasikan menjadi sebuah fitur yang berada di portal akademik perguruan tinggi. Berdasarkan hasil penelitian dan pengembangan yang telah dilakukan, dapat disimpulkan bahwa pembangunan aplikasi layanan ujian berbasis komputer di Portal Akademik Perguruan Tinggi dapat memberikan banyak manfaat. Struktur MVC dalam Framework CodeIgniter membuat pengembangan aplikasi menjadi lebih teratur. Juga penggunaan Editor Summernote dapat membuat soal-soal menjadi lebih menarik secara visual dan lebih interaktif. Dengan integrasi aplikasi ini pada Portal Akademik Perguruan Tinggi, dapat mendukung kegiatan ujian dalam Perguruan Tinggi tersebut, dan memberdayakan ruangan-ruangan komputer yang sebelumnya jarang dipakai.

Kata kunci — Rancang Bangun, Ujian Berbasis Komputer, Portal Akademik, CodeIgniter 4

I. PENDAHULUAN

Integrasi teknologi dalam pendidikan telah menyebabkan bergantinya sistem ujian menjadi secara online sebagai sarana

evaluasi kinerja siswa. Salah satu manfaat sistem ujian secara online adalah mengurangi sampah kertas yang dipakai dalam ujian berbasis kertas. Sistem ujian secara online juga dapat menilai hasil kerja peserta ujian segera setelah ujian itu selesai, jika soal-soal yang diberikan merupakan soal pilihan ganda. Sedangkan untuk soal berbentuk esai, pemeriksa jawaban dapat membaca dengan lebih jelas, dikarenakan tulisan yang ada berbentuk digital dan teratur, dibandingkan dengan tulisan tangan yang seringkali sulit dibaca.

Penerapan sistem ujian secara online juga memiliki tantangan tersendiri. Salah satu tantangan utama adalah memastikan keamanan proses ujian. Dengan maraknya kecurangan online, penting untuk menerapkan langkah-langkah untuk mencegah kecurangan dan memastikan integritas ujian [1]. Tantangan lain adalah aspek teknis pelaksanaannya, seperti memastikan bahwa sistem ujian secara online kompatibel dengan infrastruktur yang ada dan sistem dapat menangani pengguna dalam jumlah besar pada saat yang bersamaan.

Universitas Sam Ratulangi memiliki sejumlah ruangan berisikan komputer-komputer yang dapat digunakan untuk ujian secara online. Komputer-komputer tersebut telah memenuhi standar keamanan dan spesifikasi minimum untuk menjalankan sistem ujian secara online. Hal ini dibuktikan dengan pemakaiannya untuk ujian calon Mahasiswa melalui Seleksi Bersama Masuk Perguruan Tinggi Negeri (SBMTPN), maupun Tumou Tou (T2) setiap tahunnya. Namun, selain itu, komputer-komputer tersebut tidak pernah digunakan lagi.

Salah satu solusi untuk penggunaan kembali fasilitas diatas adalah adanya suatu aplikasi layanan ujian berbasis komputer yang memiliki opsi untuk pelaksanaan ujian di ruangan tersebut. Untuk kemudahan para Dosen dan Mahasiswa dalam mengaksesnya, maka aplikasi tersebut dapat diintegrasikan menjadi sebuah fitur yang berada di portal akademik perguruan tinggi.

Penelitian ini bertujuan untuk membangun sebuah aplikasi layanan ujian berbasis komputer untuk digunakan oleh Mahasiswa dan Dosen. Dari penelitian ini diharapkan memiliki manfaat yaitu menghasilkan aplikasi layanan ujian berbasis komputer yang dapat dimanfaatkan oleh civitas akademik Universitas Sam Ratulangi dan memberdayakan ruangan-ruangan komputer yang sebelumnya jarang dipakai.

A. Penelitian Terkait

- 1) “Pengembangan Aplikasi Ujian Berbasis Komputer Beserta Analisis Uji Guna Sistem Perangkat Lunaknya Menggunakan Metode SUMI (Software Usability Measurement Inventory)” [2], dimana aplikasi ini dibuat dengan mengikuti tahapan-tahapan metodologi SDLC dan dikembangkan dengan bahasa pemrograman JAVA dan RDBMS MySQL. Aplikasi ini juga diuji menggunakan metode SUMI (Software Usability Measurement Inventory).
- 2) “Perancangan Dan Implementasi Aplikasi Ujian Sekolah Berbasis Komputer atau Computer Based Test (CBT) di SMK Negeri 1 Kabupaten Sorong” [3], dimana dengan menggunakan metode Waterfall, dapat dikembangkan suatu aplikasi Computer Based Test (CBT) dalam platform web.
- 3) “Perancangan Tampilan Aplikasi Ujian Berbasis Komputer untuk Ujian Harian Sekolah Menengah Atas” [4], dimana penelitian ini menghasilkan model tampilan aplikasi (Graphic User Interface) dari aplikasi ujian berbasis komputer yang User Friendly.
- 4) “Aplikasi Ujian Berbasis Computer Based Test (Cbt) Pada SMAS Zulhijjah Batanghari” [5], dimana penelitian ini berfokus pada siswa sekolah sebagai target pengguna aplikasi tersebut.
- 5) “Pengembangan Aplikasi TOEFL Practice Exam Berbasis Website Pada Universitas Teknologi Sumbawa” [6], dimana hasil dari penelitian ini adalah aplikasi yang dapat melakukan pemberkasan soal sesuai paket soal, penilaian hasil ujian sesuai standar, penentuan batas score kelulusan dan batas partisipasi daftar ujian.
- 6) “Aplikasi Ujian Dan Penilaian TOEFL Berbasis Komputer Client Server Dengan Metode Linear Congruent Method” [7], dimana pembuatan aplikasi menggunakan metode incremental serta perancangan sistem menggunakan diagram konteks, data flow diagram dan entity relationship diagram. Aplikasi tersebut dapat mengarsipkan soal dan hasil ujian dapat tersimpan dalam server kampus. Sehingga kegiatan ujian TOEFL dapat dilaksanakan secara online.
- 7) “Perancangan Aplikasi Ujian Mahasiswa Baru Pada Universitas Serambi Mekkah Berbasis Web” [8], dimana tujuan penelitian tersebut adalah membuat perancangan aplikasi ujian mahasiswa baru berbasis web pada Universitas Serambi Mekkah menggunakan bahasa pemrograman PHP dengan database MySQL dan framewok CodeIgniter.
- 8) “Perancangan Sistem Informasi Ujian Akhir Semester Berbasis Komputer Pada Perguruan Tinggi” [9], dimana tujuan dari penelitian tersebut yaitu merancang sebuah sistem informasi ujian berbasis komputer atau yang dikenal dengan sebutan computer based test pada perguruan tinggi, dimana sistem ini dapat digunakan dalam proses pelaksanaan ujian akhir semester. Adapun metode yang digunakan pada penelitian tersebut dalam perancangan sistemnya yaitu metode Waterfall, dan hasil dari penelitian ini yaitu sebuah sistem informasi yang dapat mawadahi dosen dan mahasiswa dalam

pelaksanaan ujian akhir semester dimana dalam suatu s006Fal dapat dibuat oleh beberapa dosen pengampu, dan dengan adanya sistem ini proses penilaian ujian semester juga lebih efektif karena mahasiswa dapat langsung melihat hasil ujian mereka setelah melaksanakan ujian, serta dapat mereduksi penggunaan kertas.

B. Rancang Bangun

Dalam konteks pembuatan sistem komputer, rancang merupakan serangkaian prosedur untuk menterjemahkan hasil analisa dan sebuah sistem ke dalam bahasa pemrograman untuk mendeskripsikan dengan detail bagaimana komponen-komponen sistem di implementasikan, sedangkan bangun adalah kegiatan menciptakan sistem baru maupun mengganti atau memperbaiki sistem yang telah ada secara keseluruhan [10].

Jadi dapat disimpulkan bahwa Rancang Bangun merupakan kegiatan menerjemahkan hasil analisa ke dalam bentuk paket perangkat lunak kemudian menciptakan sistem tersebut atau memperbaiki sistem yang sudah ada.

C. Aplikasi

Aplikasi adalah program yang dibuat oleh pemakai yang ditujukan untuk melakukan suatu tugas khusus [11]. Aplikasi juga dapat diartikan sebagai penggunaan atau penerapan suatu konsep yang menjadi pokok pembahasan atau sebagai program komputer yang disusun secara sistematis untuk menolong manusia dalam melaksanakan tugas tertentu. Aplikasi software yang dirancang untuk penggunaan praktisi khusus, klasifikasi luas ini dapat dibagi menjadi 2 (dua) yaitu:

- a. Aplikasi software spesialis, program dengan dokumentasi tergabung yang dirancang untuk menjalankan tugas tertentu.
- b. Aplikasi paket, suatu program dengan dokumentasi tergabung yang dirancang untuk jenis masalah tertentu.

D. Ujian Berbasis Komputer

Ujian Berbasis Komputer (Computer Based Test atau CBT) didefinisikan sebagai penggunaan teknologi komputer dan perangkat lunak khusus dalam pelaksanaan ujian dan penilaian hasil belajar siswa [12]. Ujian Berbasis Komputer saat ini telah banyak dipilih oleh banyak lembaga pendidikan dan institusi pengujian dibandingkan dengan ujian dengan cara tradisional yang berbasis kertas.

Salah satu keuntungan utama dari Ujian Berbasis Komputer dibandingkan dengan ujian berbasis kertas adalah proses penilaian yang lebih cepat dan otomatis. Selain itu, Ujian Berbasis Komputer juga memberikan fleksibilitas dalam penyusunan soal dan variasi ujian. Pergantian ke ujian berbasis komputer juga dapat memberikan pengalaman pengujian yang lebih interaktif dan menarik bagi peserta, dengan integrasi elemen multimedia seperti gambar, video, dan audio dalam soal, juga umpan balik instan setelah peserta menyelesaikan ujian.

Meskipun demikian, Ujian Berbasis Komputer juga menimbulkan tantangan dan pertimbangan khusus. Infrastruktur teknologi yang memadai harus disiapkan,

termasuk fasilitas komputer, jaringan internet yang stabil, dan perangkat lunak ujian yang handal. Selain itu, perlu memastikan keamanan dan integritas sistem untuk mencegah kecurangan dan manipulasi dalam pelaksanaan ujian.

E. Portal Akademik

Portal Akademik adalah sebuah sistem informasi yang menyediakan informasi seputar akademik universitas yang dapat diakses melalui internet [13]. Portal Akademik dapat digunakan oleh seluruh anggota komunitas akademik, termasuk mahasiswa, dosen, staf, dan pihak-pihak terkait lainnya. Melalui Portal Akademik, pengguna dapat mengakses beragam fitur dan layanan yang mendukung proses belajar-mengajar, pengelolaan akademik, dan interaksi antara anggota komunitas akademik.

Salah satu fitur utama dari Portal Akademik adalah sistem manajemen akademik yang menyediakan informasi tentang berbagai mata kuliah yang dikontrak mahasiswa. Dosen juga dapat menggunakan Portal Akademik untuk memasukkan dan mengelola informasi kuliah, mengumumkan tugas, serta mengakses dan menilai pekerjaan mahasiswa secara daring.

Selain itu, beberapa Portal Akademik juga dapat menyediakan layanan tambahan, seperti billing untuk membayar hal-hal yang berkaitan dengan perkuliahan seperti Uang Kuliah Tunggal (UKT), dan pendaftaran acara atau seminar akademik. Pengguna dapat menggunakan portal ini untuk mengakses sumber daya informasi dan layanan lainnya yang relevan dengan kegiatan akademik dan penunjang kehidupan kampus.

F. PHP

PHP adalah sebuah bahasa pemrograman yang sering digunakan untuk pengembangan web. PHP dibuat oleh Rasmus Lerdorf pada tahun 1993 dan dirilis pada tahun 1995. PHP merupakan singkatan dari PHP: Hypertext Preprocessor [14].

PHP bekerja pada sisi server, artinya kode PHP dijalankan di server web sebelum hasilnya dikirim ke browser pengguna. Hal ini memungkinkan PHP untuk berinteraksi dengan berbagai jenis server dan basis data, termasuk Apache, Nginx, MySQL, PostgreSQL, dan banyak lagi. Dengan dukungan untuk berbagai database, PHP memungkinkan pengembang untuk membangun aplikasi web yang kuat dan kompleks dengan mudah.

PHP memiliki sintaks yang mudah dipahami dan dipelajari, yang membuatnya menjadi bahasa pemrograman yang ramah bagi pemula. Kode PHP dapat disisipkan langsung ke dalam dokumen HTML, sehingga memudahkan pengembangan aplikasi web dinamis. Selain itu, PHP juga memiliki dukungan yang luas dari komunitas pengembang dan tersedia sebagai perangkat lunak open source, sehingga mudah diakses dan digunakan secara gratis.

PHP juga memiliki banyak Framework untuk membantu pengembangan. Beberapa di antaranya adalah PRADO, CakePHP, Symfony, CodeIgniter, Laravel, Yii Framework, Phalcon dan Laminas.

G. CodeIgniter

CodeIgniter adalah Framework Open-source untuk PHP yang dirancang untuk menyederhanakan dan mempercepat

pengembangan aplikasi web. CodeIgniter dirilis oleh EllisLab pada 28 Februari 2006 [15]. Sejak saat itu CodeIgniter telah mendapatkan banyak pengikut di kalangan para pengembang karena kesederhanaannya, fleksibilitas, dan kinerjanya yang baik.

CodeIgniter dikenal karena dokumentasinya yang sangat baik, yang komprehensif dan mudah dipahami, menjadikannya Framework yang ramah bagi para pengembang, terutama bagi mereka yang baru mengenal Framework PHP atau pengembangan web. Dokumentasi ini mencakup contoh-contoh dan tutorial yang jelas, memungkinkan para pengembang untuk dengan cepat memahami konsep-konsep kerangka kerja dan mengaplikasikannya dalam praktik.

Seperti teknologi lainnya, CodeIgniter memiliki beberapa keterbatasan. Sejak dirilis, beberapa Framework PHP lainnya muncul dengan fitur-fitur modern dan praktik pengembangan yang lebih mutakhir. Namun, kesederhanaan dan kemudahan penggunaan CodeIgniter terus membuatnya menjadi pilihan yang sangat baik untuk proyek-proyek kecil, prototyping cepat, dan para pengembang yang lebih suka pendekatan yang langsung dalam pengembangan web.

CodeIgniter mengikuti pola arsitektur model-view-controller (MVC), yang membuatnya sangat terorganisir dan membantu para pengembang menciptakan aplikasi yang mudah dipelihara dan dapat berkembang. Dalam MVC, aplikasi dibagi menjadi tiga komponen utama: Model, View, dan Controller.

Model adalah komponen yang bertanggung jawab untuk mengelola data, dan berinteraksi langsung dengan Database untuk mengambil, menyimpan, dan memanipulasi data.

View adalah komponen yang bertanggung jawab untuk menampilkan data kepada pengguna. View menghasilkan output yang dapat dilihat oleh pengguna, seperti halaman HTML. Dalam pengembangan aplikasi ini, view adalah berupa file template atau file tampilan yang memanfaatkan sintaks PHP dan HTML untuk menampilkan data yang dikirim oleh controller.

Controller adalah komponen yang berperan sebagai pengendali antara model dan view. Controller menerima permintaan dari pengguna, mengatur interaksi dengan model untuk mendapatkan atau memanipulasi data, dan menentukan view mana yang akan ditampilkan kepada pengguna.

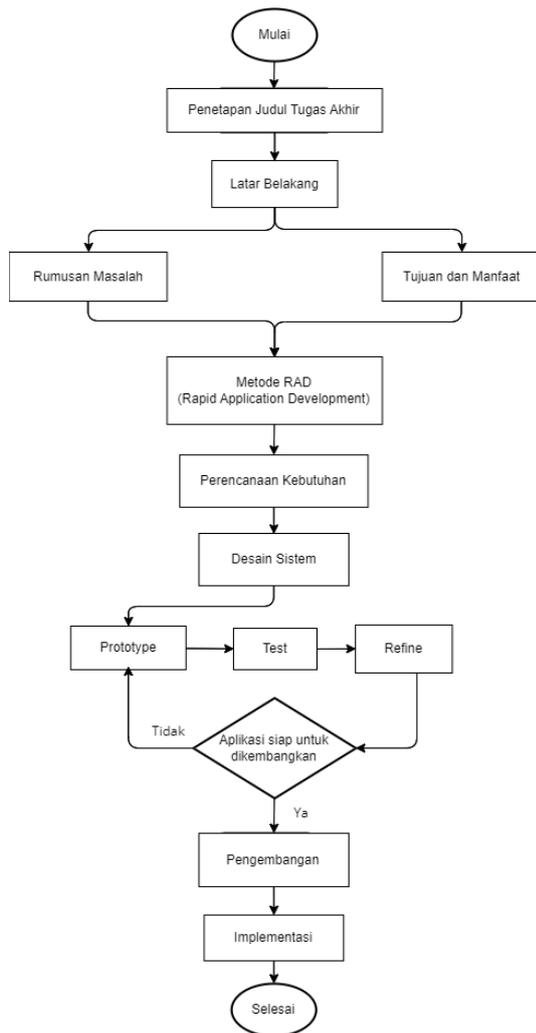
II. METODE

A. Kerangka Pemikiran

Dalam melakukan penelitian, dibuatlah kerangka pemikiran seperti pada Gambar 1, dengan tujuan untuk mengarahkan penelitian dengan lebih efisien, menggunakan metode yang ada.

B. Metode Penelitian

Metode pengembangan perangkat lunak yang akan digunakan dalam penelitian ini adalah RAD (Rapid Application Development). RAD adalah salah satu model hasil perkembangan System Development Life Cycle (SDLC)



Gambar 1. Kerangka Pemikiran

[16]. Metode ini bertujuan untuk mempercepat proses pengembangan dengan menggunakan siklus pengembangan yang singkat, dengan menekankan kolaborasi intensif antara pengembang dan stakeholder dalam membangun aplikasi yang diinginkan. Tahapan-tahapan dalam metode RAD meliputi :

- 1) Perencanaan Kebutuhan, yang melibatkan identifikasi kebutuhan bisnis yang harus dipenuhi oleh aplikasi yang akan dikembangkan. Pada tahap ini, tujuan dan ruang lingkup proyek ditentukan, serta kebutuhan-kebutuhan fungsional dan nonfungsional yang harus dipenuhi oleh sistem diteliti. Stakeholder dan pengembang berkolaborasi untuk mengidentifikasi kebutuhan yang paling krusial dan mendesak untuk dituang dalam aplikasi yang akan dibuat.
- 2) Desain Sistem, yang melibatkan perancangan konsep sistem secara umum, termasuk arsitektur, antarmuka, dan struktur database. Prototype awal sistem dikembangkan untuk memberikan gambaran awal kepada stakeholder tentang bagaimana aplikasi akan berfungsi. Stakeholder terlibat dalam pengujian dan memberikan umpan balik yang dapat digunakan untuk menyempurnakan desain

dan mengidentifikasi kebutuhan tambahan. Siklus pengujian dan perbaikan dilakukan secara berulang hingga sistem memenuhi kebutuhan yang ditetapkan.

- 3) Pengembangan, yang dimulai setelah desain sistem dan prototype awal telah diperbaiki. Pengembang bekerja untuk mengimplementasikan desain yang telah disetujui dan membangun aplikasi yang lengkap. Komponen aplikasi dikodekan, diuji, dan diintegrasikan untuk menciptakan solusi yang berfungsi sepenuhnya. Dalam metode RAD, pengembangan dilakukan dalam iterasi-iterasi yang cepat untuk menghasilkan aplikasi yang dapat berjalan dengan cepat.
- 4) Implementasi, yang melibatkan penerapan aplikasi ke lingkungan produksi atau operasional. Setelah aplikasi dianggap siap dan telah lulus pengujian, dilakukan migrasi data dan instalasi aplikasi ke infrastruktur yang dituju. Stakeholder dilibatkan dalam pelatihan pengguna dan proses adaptasi ke aplikasi baru. Setelah implementasi, aplikasi dioperasikan dan dipelihara sesuai dengan kebutuhan yang ditetapkan.

Adapun alasan penggunaan pendekatan metode ini adalah karena pendekatan ini memiliki beberapa kelebihan, diantaranya adalah : siklus pengembangan lebih pendek, lebih fleksibel, meningkatkan keterlibatan pengguna, serta dapat menekan kemungkinan kesalahan [17].

C. Unified Modeling Language (UML)

Unified Modeling Language (UML) adalah bahasa pemodelan yang dikembangkan untuk menyederhanakan proses desain perangkat lunak [18]. UML mencakup beberapa jenis diagram seperti use case diagram dan activity diagram.

Use case diagram adalah diagram yang menggambarkan kebutuhan fungsional perangkat lunak, dan dapat digunakan untuk memahami bagaimana sistem seharusnya berfungsi [19]. Sebagai bagian dalam desain aplikasi layanan ujian berbasis komputer, maka dibuatlah use case diagram seperti yang terlihat pada Gambar 2.

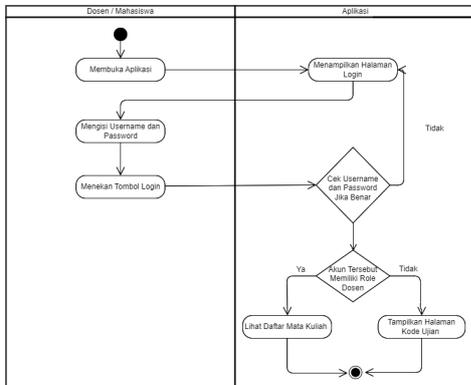
Activity diagram menggambarkan aktifitas-aktifitas, objek, state, transisi state dan event [20]. Sebagai bagian dalam desain aplikasi layanan ujian berbasis komputer, maka dibuatlah beberapa activity diagram untuk merepresentasikan alur kerja untuk setiap use case, seperti pada Gambar 3, 4, 5, dan 6.

Untuk bagian perancangan antarmuka aplikasi, maka dibuatlah rancangan dalam bentuk sketsa digital kasar seperti pada Gambar 7 dan Gambar 8. Dibuat juga prototype untuk mengembangkan contoh kerja dari aplikasi yang lebih lengkap dan fungsional menggunakan Figma. Hasil prototype dapat dilihat pada Gambar 9.

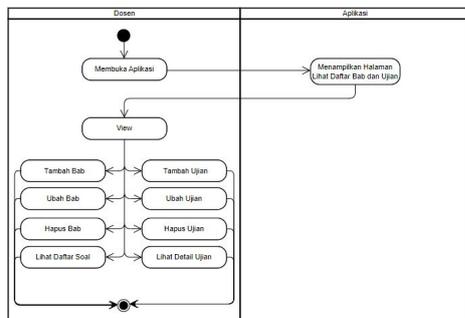
Lalu sebagai bagian dalam desain aplikasi layanan ujian berbasis komputer, maka dibuatlah ERD yang menggambarkan entitas, atribut, dan hubungan di antara entitas tersebut, dan dapat membantu perancang dan pengembang dalam memahami struktur dan hubungan antar entitas secara visual, seperti yang terlihat pada Gambar 10.



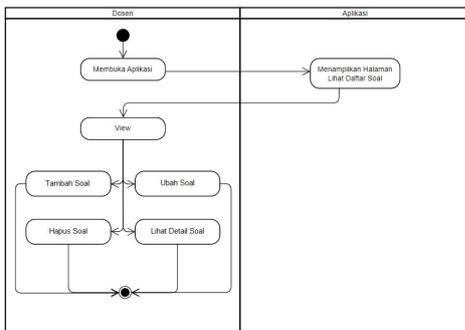
Gambar 2. Use Case Diagram



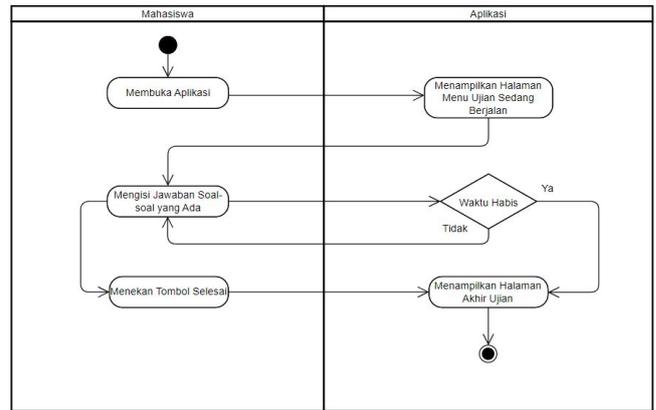
Gambar 3. Activity Diagram Halaman Login



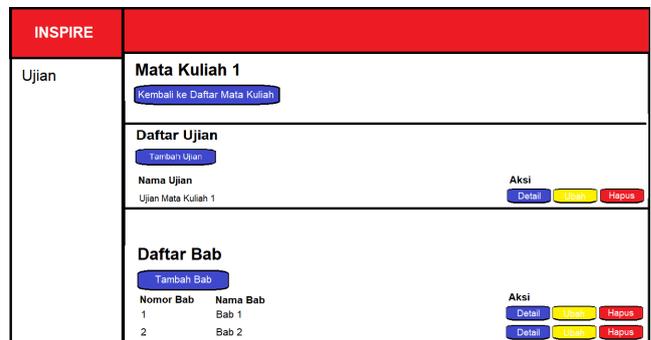
Gambar 4. Activity Diagram Lihat Daftar Bab dan Ujian



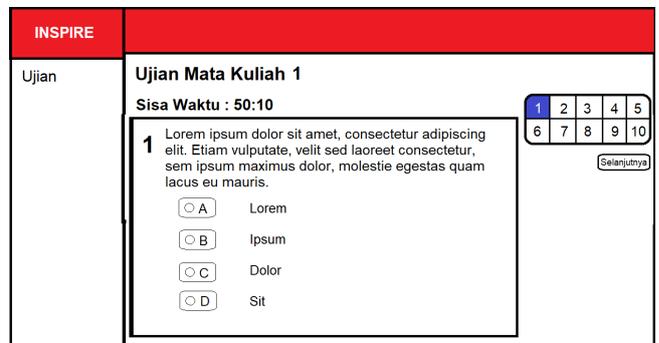
Gambar 5. Activity Diagram Lihat Daftar Soal



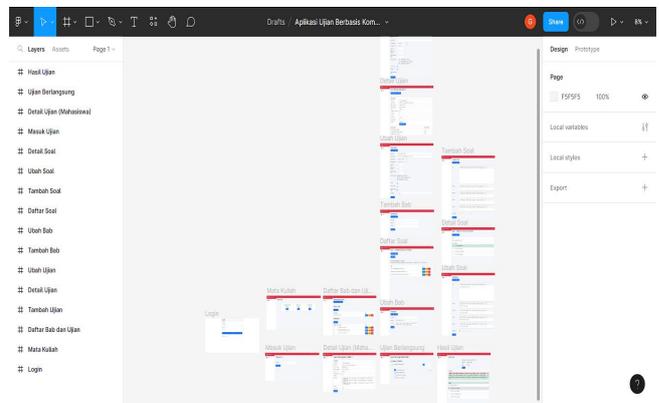
Gambar 6. Activity Diagram Menu Ujian Sedang Berjalan



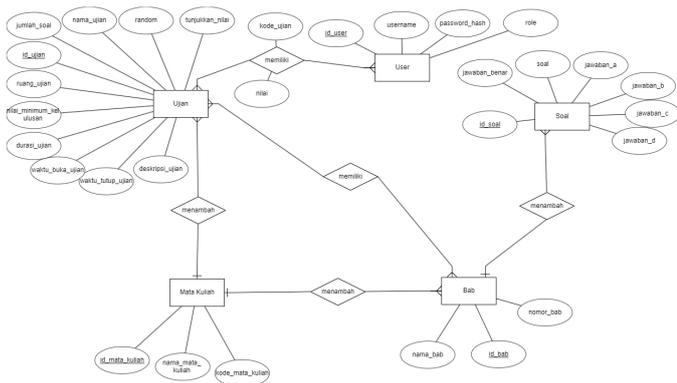
Gambar 7. Sketsa Halaman Lihat Daftar Bab dan Ujian



Gambar 8. Sketsa Halaman Menu Ujian Sedang Berjalan



Gambar 9. Prototype Aplikasi



Gambar 10. ERD untuk Aplikasi

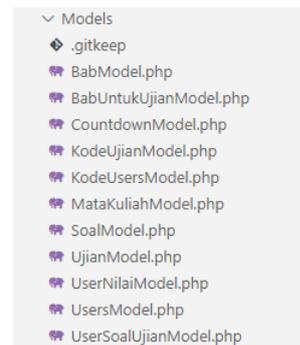
III. HASIL DAN PEMBAHASAN

A. Penerapan struktur Model-View-Controller (MVC)

Dalam penerapan struktur Model-View-Controller (MVC) untuk aplikasi, file kodingan utama terbagi menjadi 3, yaitu Model, Controller, dan View.

Dalam Gambar 11, file kodingan yang ada dalam folder Model merepresentasikan tabel-tabel dengan nama yang sama dalam database, yang dipakai untuk operasi-operasi dalam aplikasi, yaitu :

- 1) BabModel.php, yang dipakai untuk operasi-operasi yang menggunakan tabel 'bab', yang berisi data dari semua bab yang ada.
- 2) BabUntukUjianModel.php, yang dipakai untuk operasi-operasi yang menggunakan tabel 'bab_untuk_ujian', yang berisi data dari bab yang akan digunakan dalam suatu ujian.
- 3) CountdownModel.php, yang dipakai untuk operasi-operasi yang menggunakan tabel 'countdown', yang berisi data dari waktu tersisa untuk suatu sesi ujian.
- 4) KodeUjianModel.php, yang dipakai untuk operasi-operasi yang menggunakan tabel 'kode_ujian', yang berisi data dari kode yang digunakan dalam suatu ujian.
- 5) KodeUsersModel.php, yang dipakai untuk operasi-operasi yang menggunakan tabel 'kode_users', yang berisi data gabungan dari kode ujian dan pengguna, yang menandakan suatu sesi ujian.
- 6) MataKuliahModel.php, yang dipakai untuk operasi-operasi yang menggunakan tabel 'mata_kuliah', yang berisi data dari semua mata kuliah yang ada.
- 7) SoalModel.php, yang dipakai untuk operasi-operasi yang menggunakan tabel 'soal', yang berisi data dari semua soal yang ada.
- 8) UjianModel.php, yang dipakai untuk operasi-operasi yang menggunakan tabel 'ujian', yang berisi data dari semua ujian yang ada.
- 9) UserNilaiModel.php, yang dipakai untuk operasi-operasi yang menggunakan tabel 'user_nilai', yang berisi data dari nilai yang dihasilkan sesudah ujian.
- 10) UsersModel.php, yang dipakai untuk operasi-operasi yang menggunakan tabel 'users', yang berisi data dari pengguna.



Gambar 11. Bagian Model dalam Pengembangan Aplikasi

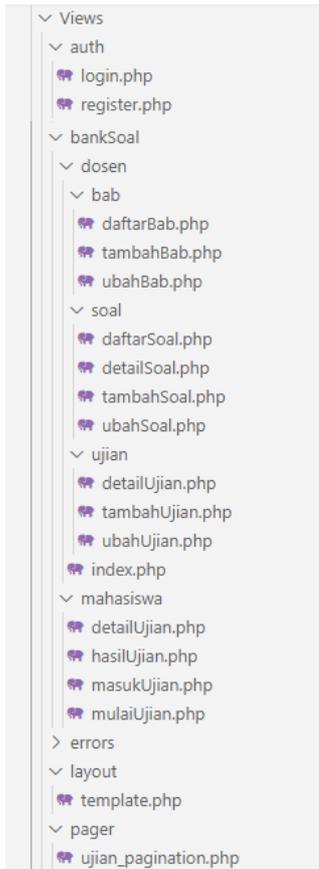
- 11) UserSoalUjianModel.php, yang dipakai untuk operasi-operasi yang menggunakan tabel 'user_soal_ujian', yang berisi data dari soal-soal yang digunakan dalam suatu sesi ujian.

Dalam Gambar 12, file kodingan yang ada dalam folder View terbagi dalam 5 folder, yaitu :

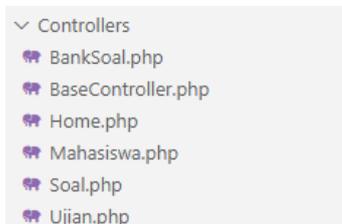
- 1) Auth, yang berisi login.php dan register.php, dimana file-file tersebut menampilkan halaman masuk dan membuat akun.
- 2) BankSoal, yang dibagi menjadi 2 folder, yaitu Dosen dan Mahasiswa. Dalam folder Dosen, terdapat folder Bab, Ujian, dan Soal, yang masing-masing berisi view untuk menambah, mengubun, dan menghapus bab, ujian, atau soal, dengan satu view untuk menampilkan daftar soal yang ada. Dalam folder Mahasiswa, terdapat view untuk memasukkan kode ujian, menampilkan detail ujian, menjalanknkan ujian, dan menampilkan hasil ujian.
- 3) Errors, yang merupakan built-in view dari CodeIgniter untuk menampilkan pesan-pesan jika dalam aplikasi terjadi error.
- 4) Layout, yang berisi file template.php, dimana file tersebut berfungsi sebagai template dasar untuk semua view yang lain.
- 5) Pager, yang berisi ujian_pagination.php, dimana file tersebut berfungsi untuk menampilkan navigasi nomor untuk ujian.

Dalam Gambar 13, file kodingan yang ada dalam folder Controller terbagi dalam 6 file, yaitu :

- 1) BankSoal.php, yang mengatur bagian dosen, terutama pada bagian mata kuliah dan bagian operasi-operasi untuk bab.
- 2) BaseController.php, yang merupakan built-in controller dari CodeIgniter untuk memberikan fungsi-fungsi yang diperlukan dalam pembuatan controller yang lain.
- 3) Home.php, yang mengatur aplikasi ketika pertama kali dibuka.
- 4) Mahasiswa.php, yang mengatur fungsionalitas pada bagian mahasiswa.
- 5) Soal.php, yang mengatur fungsionalitas bagian operasi-operasi untuk soal.
- 6) Ujian.php yang mengatur fungsionalitas bagian operasi-operasi untuk ujian.



Gambar 12. Bagian View dalam Pengembangan Aplikasi



Gambar 13. Bagian Controller dalam Pengembangan Aplikasi

B. Framework, Library, dan Package yang digunakan

Dalam pengembangan aplikasi, digunakan beberapa *Framework, Library, dan Package* seperti Bootstrap, jQuery, Popper, dan Myth/Auth.

Bootstrap adalah framework front-end untuk pengembangan aplikasi web responsif. Bootstrap menyediakan kumpulan tools & styles untuk CSS yang siap dipakai untuk membangun antarmuka pengguna yang menarik dan responsif.

Untuk menggunakan Bootstrap, aplikasi juga harus turut menyertakan library JQuery dan Popper. JQuery adalah library yang menyederhanakan pengembangan aplikasi web dengan menyediakan fungsi-fungsi JavaScript yang mudah digunakan untuk berbagai tugas seperti manipulasi DOM, penanganan peristiwa, animasi, dan pengambilan data dari server. Sedangkan Popper adalah library yang digunakan untuk pengaturan tata letak dan pengaturan popper (tooltip, popover, dll.).

Summernote adalah library editor teks WYSIWYG (What You See Is What You Get) yang menyediakan antarmuka yang intuitif dan ramah pengguna untuk membuat dan mengedit konten rich text. Summernote digunakan dalam aplikasi untuk pembuatan soal agar memungkinkan pengguna untuk memformat teks, menyisipkan gambar, membuat tabel, menerapkan style, dan melakukan operasi pengeditan teks lainnya.

Dalam Gambar 14 terdapat file `template.php` yang memuat Bootstrap 4, JQuery, Popper, dan Summernote pada bagian head menggunakan HTML. Dikarenakan file `template.php` akan digunakan oleh setiap halaman dari aplikasi, maka Bootstrap 4, JQuery, Popper, dan Summernote juga akan terus dimuat dalam setiap kali aplikasi dibuka.

Myth/Auth adalah package autentikasi untuk framework CodeIgniter 4 yang menyediakan sejumlah fitur dan fungsionalitas yang diperlukan untuk mengimplementasikan sistem autentikasi dalam aplikasi. Myth/Auth memudahkan pengembangan aplikasi dalam mengelola login, logout, dan otorisasi pengguna. Myth/Auth juga menyediakan metode untuk mengatur dan mengelola peran dan izin pengguna. Menggunakan Myth/Auth untuk pengembangan aplikasi dapat dengan cepat mengimplementasikan fitur autentikasi yang aman dan dapat disesuaikan dalam aplikasi.

C. Pembuatan Database

Database yang telah dirancang dengan ERD kemudian dibuat dalam PhpMyAdmin, yang adalah sebuah aplikasi web berbasis PHP untuk mengelola dan mengelola basis data MySQL.

Dalam Gambar 15, terdapat tabel-tabel yang merupakan hasil migrasi tabel-tabel dari Myth/Auth, dimana migrasi tersebut diperlukan agar library Myth/Auth dapat berjalan dengan semestinya. Tabel-tabel yang merupakan hasil migrasi tabel-tabel dari Myth/Auth adalah `auth_activation_attempts`, `auth_groups`, `auth_groups_permissions`, `auth_groups_users`, `auth_logins`, `auth_permissions`, `auth_reset_attempts`, `auth_tokens`, `auth_users_permissions`, `migrations`, dan `users`.

Selain itu terdapat tabel-tabel yang merepresentasikan entitas yang telah dibuat di ERD, yaitu `mata_kuliah`, `bab`, `soal`, dan `ujian`. Juga terdapat tabel-tabel yang merupakan penggabungan antara dua tabel tertentu atau tabel untuk kepentingan lain, yaitu `bab_untuk_ujian`, `kode_ujian`, `kode_users`, `user_nilai`, dan `user_soal_ujian`.

D. Pengkodean dan Hasil

Dalam tahap pengkodean, desain dan logika yang telah dirancang sebelumnya ditulis dalam bentuk kode. Contoh bagian kode dapat dilihat dalam Gambar 16.

Dalam Gambar 16, terdapat bagian file `Routes.php` yang mengatur URI routing untuk aplikasi. File `Routes.php` sendiri merupakan file bawaan dari CodeIgniter 4, yang dapat dimodifikasi dalam pengembangan aplikasi. Dapat dilihat bahwa terdapat dua route group, yaitu "banksoal", yang ditujukan untuk Dosen, dan "ujian", yang ditujukan untuk Mahasiswa.

Setelah proses pengkodean selesai, pengembangan yang

dilakukan telah berhasil mengimplementasikan fitur-fitur yang



Gambar 14. Pemuatan Bootstrap 4, JQuery, Popper, dan Summernote

telah dirancang dan membangun sebuah aplikasi layanan ujian berbasis komputer. Contoh hasil dari pengkodean dapat dilihat pada Gambar 15 dan Gambar 16.

E. Pengujian

Setelah proses pengembangan selesai, aplikasi perlu menjalani serangkaian pengujian untuk memverifikasi bahwa semua fitur dan fungsionalitas aplikasi bekerja dengan benar dan sesuai dengan spesifikasi yang telah ditetapkan. Metode pengujian yang digunakan adalah *Alpha Testing* dengan menggunakan *Black Box Testing*, dan *Beta Testing* dengan menyebarkan kuisioner.

Dalam *Black Box Testing*, fokus utama adalah pada perilaku dan input/output yang dihasilkan oleh aplikasi, tanpa memperhatikan struktur internal atau logika implementasi. Hasil pengujian menggunakan Black Box Testing dapat dilihat pada Tabel 1.

TABEL 1

HASIL PENGUJIAN MENGGUNAKAN BLACK BOX TESTING

No.	Use Case	Hasil yang Diharapkan	Hasil yang Didapatkan
1.	Login	Dosen / Mahasiswa masuk ke halaman selanjutnya jika <i>username</i> dan <i>password</i> benar, dan jika <i>username</i> atau <i>password</i> salah maka diberi pesan.	Sesuai
2.	Logout	Dosen / Mahasiswa dikeluarkan dari halaman tersebut dan langsung dibawa ke halaman <i>login</i> .	Sesuai
3.	Lihat Daftar Mata Kuliah	Aplikasi menampilkan daftar mata kuliah yang terdiri dari nama mata kuliah dan kode mata kuliah. Terdapat tombol untuk menuju ke masing-masing mata kuliah.	Sesuai
4.	Lihat Daftar Bab dan Ujian	Aplikasi hanya menampilkan daftar bab dan ujian dari mata kuliah sebelumnya, dimana ujian terdiri dari nama nama ujian, dan	Sesuai

bab terdiri dari nomor bab dan nama bab. Untuk masing-masing ujian dan bab terdapat tombol untuk melihat detail, mengubah, dan menghapus ujian atau bab tersebut. Terdapat juga tombol untuk menambah ujian atau bab baru.

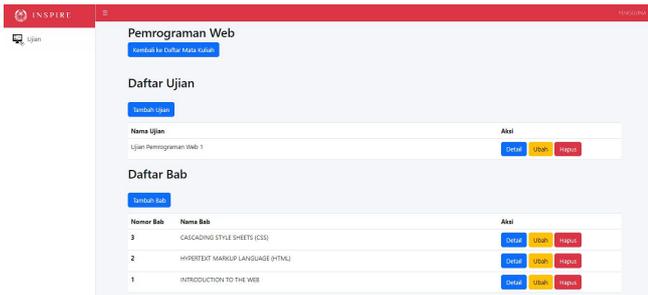
5.	Tambah Bab	Aplikasi menampilkan formulir kosong yang terdiri dari nomor bab dan nama bab. Jika nomor bab atau nama bab yang diisi telah ada di Database maka akan ditolak, namun jika nomor bab atau nama bab yang diisi belum ada di Database maka data tersebut akan diisi oleh aplikasi di Database.	Sesuai
6.	Ubah Bab	Aplikasi menampilkan formulir yang berisi data bab yang dipilih, yang terdiri dari nomor bab dan nama bab. Jika nomor bab dan nama bab yang diisi sama dengan yang ada di Database maka akan ditolak, namun jika nomor bab atau nama bab yang diisi tidak sama dengan data yang ada di Database maka data tersebut akan diisi oleh aplikasi di Database.	Sesuai
7.	Hapus Bab	Aplikasi menghapus data bab tersebut dari Database.	Sesuai
8.	Lihat Daftar Soal	Aplikasi hanya menampilkan daftar soal dari bab sebelumnya. Untuk masing-masing soal terdapat tombol untuk melihat detail, mengubah, dan menghapus soal tersebut. Terdapat juga tombol untuk menambah soal baru.	Sesuai
9.	Tambah Soal	Aplikasi menampilkan formulir kosong yang terdiri dari detail untuk soal. Formulir untuk	Sesuai

		pertanyaan dan jawaban menggunakan Summernote. Jika data soal yang diisi telah ada di Database maka akan ditolak, namun jika data soal yang diisi belum ada di Database maka data tersebut akan diisi oleh aplikasi di Database.				
10.	Ubah Soal	Aplikasi menampilkan formulir yang berisi data soal yang dipilih. Jika data soal yang diisi sama dengan yang ada di Database maka akan ditolak, namun jika data soal yang diisi tidak sama dengan data yang ada di Database maka data tersebut akan diisi oleh aplikasi di Database.	Sesuai	15.	Hapus Ujian Database. Aplikasi menghapus data ujian tersebut dari Database.	Sesuai
11.	Hapus Soal	Aplikasi menghapus data soal tersebut dari Database.	Sesuai	16.	Lihat Detail Ujian Aplikasi Menampilkan detail dari ujian yang telah dipilih. Jika telah ada kode ujian, maka kode ujian akan ditampilkan bersamasama dengan detail ujian. Namun jika belum ada kode ujian, maka terdapat tombol untuk <i>generate</i> kode ujian. Jika tombol itu ditekan, maka kode ujian akan di- <i>generate</i> secara otomatis, dan kode ujian tersebut akan ditampilkan bersamasama dengan detail ujian.	Sesuai
12.	Lihat Detail Soal	Aplikasi Menampilkan detail dari Soal yang telah dipilih. Dapat dilihat pertanyaan, seluruh opsi jawaban, dan jawaban benar yang ditandai dengan <i>highlight</i> warna hijau pada salah satu jawaban.	Sesuai	17.	Halaman Kode Ujian Aplikasi menampilkan kolom untuk mengisi kode ujian.	Sesuai
13.	Tambah Ujian	Aplikasi menampilkan formulir kosong yang terdiri dari detail untuk ujian. Jika data ujian yang diisi telah ada di Database maka akan ditolak, namun jika data ujian yang diisi belum ada di Database maka data tersebut akan diisi oleh aplikasi di Database.	Sesuai	18.	Lihat Informasi Ujian Aplikasi menampilkan detail ujian yang akan dijalankan dan tombol untuk memulai ujian.	Sesuai
14.	Ubah Ujian	Aplikasi menampilkan formulir yang berisi data ujian yang dipilih. Jika data ujian yang diisi sama dengan yang ada di Database maka akan ditolak, namun jika data ujian yang diisi tidak sama dengan data yang ada di Database maka data tersebut akan diisi oleh aplikasi di Database.	Sesuai	19.	Menu Ujian Sedang Berjalan Aplikasi menampilkan soal satu-persatu, dengan waktu hitung mundur dan navigasi ke soal-soal lainnya.	Sesuai
				20.	Halaman Akhir Ujian Aplikasi memberitahukan bahwa ujian telah selesai, dan jika diperbolehkan, menampilkan nilai dan <i>review</i> ujian yang telah dijalankan sebelumnya.	Sesuai

Pengujian dengan menyebarkan kuisisioner adalah salah satu cara untuk mengumpulkan umpan balik dari pengguna atau terkait pengalaman mereka dalam menggunakan aplikasi. Kuisisioner ini dibuat menggunakan skala Likert dari skala 1 sampai 4, dan disebarkan kepada 30 orang responden yang merupakan mahasiswa aktif dari Universitas Sam Ratulangi. Untuk skor penilaian dapat dilihat pada tabel 2, dan untuk tampilan kuisisioner dapat dilihat pada Gambar 17.

TABEL 2
KETERANGAN SKOR PENILAIAN

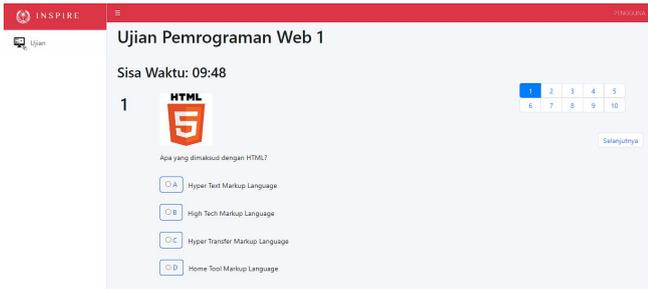
Tingkat kepuasan	Skala
Sangat Setuju	1
Setuju	2
Tidak Setuju	3
Sangat Tidak Setuju	4



Gambar 15. Halaman Lihat Daftar Bab dan Ujian



Gambar 18. Soal Pertama Kuisisioner



Gambar 16. Halaman Menu Ujian Sedang Berjalan



Gambar 19. Soal Kedua Kuisisioner



Gambar 17. Formulir Kuisisioner

Hasil kuisisioner dihitung menggunakan rumus:

$$y = (\sum(N.R)) / (\text{Skor ideal}) \times 100\%$$

Keterangan:

Y = Nilai persentase yang dicari

X = Jumlah nilai kategori jawaban dikalikan dengan frekuensi ($\sum = N.R$)

N = Nilai dari setiap jawaban

R = Frekuensi / Jumlah Partisipan

Skor ideal = Nilai tertinggi dikalikan dengan jumlah sampel ($4 \times 30 = 120$) [21].

Hasil pengujian pada Gambar 18 menunjukkan bahwa 1 dari 30 responden memberikan nilai Tidak Setuju ($N.R = 2 \times 1 = 2$), 14 dari 30 responden memberikan nilai Setuju ($N.R = 3 \times 14 = 42$), dan 15 dari 30 responden memberikan nilai Sangat Setuju ($N.R = 4 \times 15 = 60$), sehingga diperoleh $\sum(N.R) = 104$. Untuk total persentase kepuasan dari soal nomor 1 jika dimasukkan pada rumus $Y = \sum(N.R) / \text{total skor ideal} \times 100\%$ yakni $Y = 104 / 120 \times 100\% = 86,6\%$.

Hasil pengujian pada Gambar 19 menunjukkan bahwa 8 dari 30 responden memberikan nilai Setuju ($N.R = 3 \times 8 = 24$), dan 22 dari 30 responden memberikan nilai Sangat Setuju ($N.R = 4 \times 22 = 88$), sehingga diperoleh $\sum(N.R) = 110$. Untuk

total persentase kepuasan dari soal nomor 2 jika dimasukkan pada rumus $Y = \sum(N.R) / \text{total skor ideal} \times 100\%$ yakni $Y = 110 / 120 \times 100\% = 91,6\%$.

Hasil pengujian pada Gambar 20 menunjukkan bahwa 1 dari 30 responden memberikan nilai Tidak Setuju ($N.R = 2 \times 1 = 2$), 9 dari 30 responden memberikan nilai Setuju ($N.R = 3 \times 9 = 27$), dan 20 dari 30 responden memberikan nilai Sangat Setuju ($N.R = 4 \times 20 = 80$), sehingga diperoleh $\sum(N.R) = 109$. Untuk total persentase kepuasan dari soal nomor 3 jika dimasukkan pada rumus $Y = \sum(N.R) / \text{total skor ideal} \times 100\%$ yakni $Y = 109 / 120 \times 100\% = 90,8\%$.

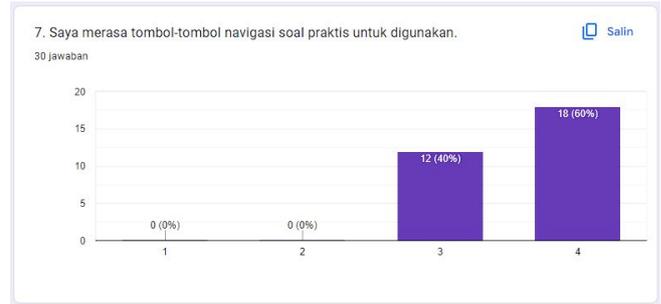
Hasil pengujian pada Gambar 21 menunjukkan bahwa 12 dari 30 responden memberikan nilai Setuju ($N.R = 3 \times 12 = 36$), dan 18 dari 30 responden memberikan nilai Sangat Setuju ($N.R = 4 \times 18 = 72$), sehingga diperoleh $\sum(N.R) = 108$. Untuk total persentase kepuasan dari soal nomor 4 jika dimasukkan pada rumus $Y = \sum(N.R) / \text{total skor ideal} \times 100\%$ yakni $Y = 108 / 120 \times 100\% = 90\%$.

Hasil pengujian pada Gambar 22 menunjukkan bahwa 2 dari 30 responden memberikan nilai Tidak Setuju ($N.R = 2 \times 2 = 4$), 8 dari 30 responden memberikan nilai Setuju ($N.R = 3 \times 8 = 24$), dan 20 dari 30 responden memberikan nilai Sangat Setuju ($N.R = 4 \times 20 = 80$), sehingga diperoleh $\sum(N.R) = 108$. Untuk total persentase kepuasan dari soal nomor 5 jika dimasukkan pada rumus $Y = \sum(N.R) / \text{total skor ideal} \times 100\%$ yakni $Y = 108 / 120 \times 100\% = 90\%$.

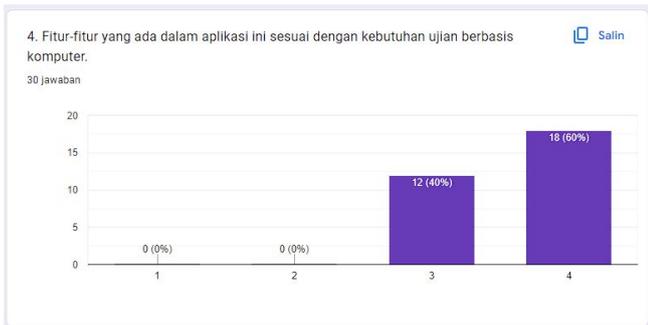
Hasil pengujian pada Gambar 23 menunjukkan bahwa 11 dari 30 responden memberikan nilai Setuju ($N.R = 3 \times 11 = 33$), dan 19 dari 30 responden memberikan nilai Sangat Setuju ($N.R = 4 \times 19 = 76$), sehingga diperoleh $\sum(N.R) = 109$. Untuk total persentase kepuasan dari soal nomor 6 jika dimasukkan pada rumus $Y = \sum(N.R) / \text{total skor ideal} \times 100\%$ yakni pada rumus $Y = \sum(N.R) / \text{total skor ideal} \times 100\%$ yakni $Y = 109 / 120 \times 100\% = 90,8\%$.



Gambar 20. Soal Ketiga Kuisisioner



Gambar 24. Soal Ketujuh Kuisisioner



Gambar 21. Soal Keempat Kuisisioner



Gambar 22. Soal Kelima Kuisisioner



Gambar 23. Soal Keenam Kuisisioner

Hasil pengujian pada Gambar 24 menunjukkan bahwa 12 dari 30 responden memberikan nilai Setuju (N.R = 3 × 12 = 36), dan 18 dari 30 responden memberikan nilai Sangat Setuju (N.R = 4 × 18 = 72), sehingga diperoleh $\sum(N.R) = 108$. Untuk total persentase kepuasan dari soal nomor 7 jika dimasukkan pada rumus $Y = \frac{\sum(N.R)}{\text{total skor ideal}} \times 100\%$ yakni $Y = \frac{108}{120} \times 100\% = 90\%$.

Untuk total persentase kepuasan pengguna Aplikasi Layanan Ujian Komputer ini jika dijumlahkan dari soal 1-10

adalah: $86,6 + 91,6 + 90,8 + 90 + 90 + 90,8 + 90 + 91,6 + 90 + 93,3 = 904,7 / 10 = 90,5\%$.

IV. KESIMPULAN DAN SARAN

Berdasarkan hasil penelitian dan pengembangan yang telah dilakukan, dapat ditarik kesimpulan bahwa pembangunan aplikasi layanan ujian berbasis komputer di Portal Akademik Perguruan Tinggi dapat mendukung kegiatan ujian dalam Perguruan Tinggi tersebut, dan memberdayakan ruangan-ruangan komputer yang sebelumnya jarang dipakai. Integrasi aplikasi ini pada Portal Akademik Perguruan Tinggi dapat mempermudah Dosen dan Mahasiswa dalam menggunakan aplikasi layanan ujian. Pengembangan aplikasi layanan ujian dengan struktur MVC dalam Framework CodeIgniter membuat pengembangan aplikasi menjadi lebih teratur. Penggunaan Editor Summernote dalam pembuatan soal dapat membuat soal-soal menjadi lebih menarik secara visual dan lebih interaktif.

Terdapat saran yang bisa diberikan, terutama untuk pengembangan aplikasi layanan ujian ini secara lebih lanjut, yaitu pentingnya melibatkan semua pihak yang terkait, termasuk dosen, mahasiswa, dan pihak administrasi perguruan tinggi, dalam proses pengembangan dan implementasi. Direkomendasikan agar aplikasi ini dirancang dengan antarmuka yang lebih intuitif dan mudah digunakan, sehingga semua pengguna dapat dengan mudah mengakses dan memanfaatkannya. Perlu diberikan pelatihan dan dukungan yang memadai kepada pengguna, agar mereka dapat menguasai penggunaan aplikasi dengan baik. Perlu juga melakukan evaluasi dan pemantauan secara terus-menerus terhadap aplikasi ini setelah pengembangan, untuk manfaat yang maksimal.

V. KUTIPAN

- [1] R. Hakim, "Tingkat kecurangan tes online dibandingkan dengan tes tertulis di SMK Negeri 2 Yogyakarta," *Jurnal Pendidikan Teknik Elektro*, vol. 6, no. 3, 2016.
- [2] Pratomo dan R. Mantala, "Pengembangan aplikasi ujian berbasis komputer beserta analisis uji guna sistem perangkat lunaknya menggunakan metode SUMI (Software Usability Measurement Inventory)," *POSITIF: Jurnal Sistem dan Teknologi Informasi*, vol. 2, no. 1, 2016.

- [3] M. P. Saptono dan H. Widjasena, "Perancangan dan Implementasi Aplikasi Ujian Sekolah Berbasis Komputer Atau Computer Based Test (CBT) Di SMK Negeri 1 Kabupaten Sorong," *Electro Luceat*, vol. 5, no. 2, hlm. 5–13, 2019.
- [4] C. N. P. Dewi dan R. Raafi'udin, "Perancangan Tampilan Aplikasi Ujian Berbasis Komputer Untuk Ujian Harian Sekolah Menengah Atas," *ILKOM Jurnal Ilmiah*, vol. 10, no. 3, hlm. 298–305, 2018.
- [5] J. YANTO dan others, "APLIKASI UJIAN BERBASIS COMPUTER BASED TEST (CBT) PADA SMAS ZULHIJAH BATANGHARI," Universitas Nurdin Hamzah, 2022.
- [6] I. Budikusuma dan E. S. Susanto, "PENGEMBANGAN APLIKASI TOEFL PRACTICE EXAM BERBASIS WEBSITE PADA UNIVERSITAS TEKNOLOGI SUMBAWA," *JATI (Jurnal Mahasiswa Teknik Informatika)*, vol. 6, no. 1, hlm. 75–78, 2022.
- [7] K. Kirman dan G. Gunawan, "Aplikasi Ujian Dan Penilaian TOEFL Berbasis Komputer Client Server Dengan Metode Linear Congruent Method," *Jurnal Teknologi Sistem Informasi dan Sistem Komputer TGD*, vol. 5, no. 2, hlm. 205–213, 2022.
- [8] M. Fadhli, R. Islamadina, dan D. M. Apriliansyah, "PERANCANGAN APLIKASI UJIAN MAHASISWA BARU PADA UNIVERSITAS SERAMBI MEKKAH BERBASIS WEB," *Jurnal Nasional Komputasi Dan Teknologi Informasi*, vol. 2, no. 2620–8342, 2019.
- [9] A. A. M. Suradi, "Perancangan Sistem Informasi Ujian Akhir Semester Berbasis Komputer Pada Perguruan Tinggi," *Inspiration: Jurnal Teknologi Informasi dan Komunikasi*, vol. 12, no. 1, hlm. 30–40, 2022.
- [10] R. S. Pressman, *Software engineering: a practitioner's approach*. Palgrave macmillan, 2005.
- [11] K. Abdul, "Pengenalan sistem informasi," Penerbit Andi, Yogyakarta, 2003.
- [12] J. Bull dan C. McKenna, *A blueprint for computer-assisted assessment*. Routledge, 2003.
- [13] J. F. Saraun, A. M. Sambul, dan A. S. M. Lumenta, "Integrasi Layanan Google Classroom dengan Sistem Portal Akademik Perguruan Tinggi," *Jurnal Teknik Informatika*, vol. 17, no. 1, hlm. 27–34, 2022.
- [14] The PHP Group, "PHP: Hypertext Preprocessor," <https://www.php.net/>, 24 Januari 2023.
- [15] CodeIgniter Foundation, "Welcome to CodeIgniter," <https://codeigniter.com/>, 26 Januari 2023.
- [16] S. Aswati, M. S. Ramadhan, A. U. Firmansyah, dan K. Anwar, "Studi analisis model rapid application development dalam pengembangan sistem informasi," *MATRIK: Jurnal Manajemen, Teknik Informatika dan Rekayasa Komputer*, vol. 16, no. 2, hlm. 20–27, 2017.
- [17] N. Hidayat dan K. Hati, "Penerapan Metode Rapid Application Development (RAD) dalam Rancang Bangun Sistem Informasi Rapor Online (SIRALINE)," *Jurnal Sistem Informasi*, vol. 10, no. 1, hlm. 8–17, 2021.
- [18] B. Gosala, S. R. Chowdhuri, J. Singh, M. Gupta, dan A. Mishra, "Automatic classification of UML class diagrams using deep learning technique: convolutional neural network," *Applied Sciences*, vol. 11, no. 9, hlm. 4267, 2021.
- [19] R. Fauzan, D. Siahaan, S. Rochimah, dan E. Triandini, "A different approach on automated use case diagram semantic assessment," *International Journal of Intelligent Engineering and Systems*, 2021.
- [20] H. Haviluddin, "Memahami Penggunaan UML (Unified Modelling Language)," *Informatika Mulawarman: Jurnal Ilmiah Ilmu Komputer*, vol. 6, no. 1, hlm. 1–15, 2016.
- [21] A. Suandi, F. N. Khasanah, dan E. Retnoningsih, "Pengujian Sistem Informasi E-commerce Usaha Gudang Cokelat Menggunakan Uji Alpha dan Beta," *Information System For Educators And Professionals: Journal of Information System*, vol. 2, no. 1, hlm. 61–70, 2017.



Gabriel F. Z. Sadu lahir di Manado pada tanggal 5 Februari 2002. Penulis menempuh pendidikan pada tahun 2007 di SD Negeri 3 Manado dan SD GMIM Tumpengan Sea, pada tahun 2013 melanjutkan pendidikan di SMP Negeri 2 Nonapan dan SMP Garuda Manado, kemudian pada tahun 2016 melanjutkan pendidikan di SMA Negeri 9 Binsus Manado. Pada tahun 2019, penulis melanjutkan studi di jenjang sarjana di Program Studi Teknik Informatika, Fakultas Teknik, Jurusan Teknik Elektro, Universitas Sam Ratulangi Manado. Selama perkuliahan penulis tergabung dalam organisasi kemahasiswaan dan aktif berkegiatan dalam Himpunan Mahasiswa Elektro (HME) FT-UNSRAT. Di tingkat program studi penulis juga tergabung sebagai anggota dalam komunitas UNSRAT IT Community (UNITY).