

**PERANCANGAN APLIKASI PEMILIHAN PEMINATAN DAN
UJIAN KOMPREHENSIF MAHASISWA BERBASIS WEBSITE
PADA PROGRAM STUDI PTI**

SKRIPSI

Diajukan Oleh

**TEUKU MUKSAL MINA
NIM. 190212026
REKAYASA PERANGKAT LUNAK**

**Mahasiswa Fakultas Tarbiyah dan Keguruan
Program Studi Pendidikan Teknologi Informasi**



**UNIVERSITAS ISLAM NEGERI AR-RANIRY
FAKULTAS TARBIYAH DAN KEGURUAN
PROGRAM STUDI PENDIDIKAN TEKNOLOGI INFORMASI
2023 M/ 1443 H**

SKRIPSI
PERANCANGAN APLIKASI PEMILIHAN PEMINATAN DAN UJIAN
KOMPREHENSIF MAHASISWA BERBASIS WEBSITE PADA PROGRAM
STUDI PTI

Diajukan Oleh:

TEUKU MUKSAL MINA

NIM. 190212026

BIDANG PEMINATAN:


REKAYASA PERANGKAT LUNAK

Mahasiswa Fakultas Tarbiyah Dan Keguruan
Program Studi Pendidikan Teknologi Informasi

Disetujui Oleh:

Pembimbing 1

Pembimbing 2


Ridwan, S.ST., M.T.

NIP. 198402242019031004


Aulia Syarif Aziz, S.Kom., M.Sc.

NIP. 1993005212022031001

**PERANCANGAN APLIKASI PEMILIHAN PEMINATAN DAN UJIAN
KOMPREHENSIF MAHASISWA BERBASIS WEBSITE PADA
PROGRAM STUDI PTI**

SKRIPSI

Telah di uji oleh Panitia Ujian Munaqasyah Skripsi Fakultas Tarbiyah dan Keguruan Universitas Islam Negeri Ar-Raniry Banda Aceh dan dinyatakan lulusserta diterima sebagai salah satu beban studi Program Sarjana (S-1) dalam Pendidikan Teknologi Informasi

Pada :


Sabtu, 16 Desember 2023

3 Jumadil Akhir 1445 H


Darussalam-Banda Aceh

Panitia Ujian Munaqasyah Skripsi


Ketua


Ridwan, S.ST., M.T.
NIP. 198402242019031004

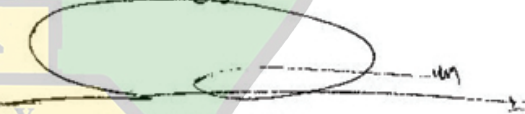
Sekretaris


Aulia Syarif Aziz, S.Kom., M.Sc.
NIP. 1993005212022031001

Penguji 1


Raihan Islamadina, M.T.
NIP. 198901312020122011

Penguji 2


Firmansyah, M.T.
NIP. 198704212015031002

Mengetahui,

Dekan Fakultas Tarbiyah dan Keguruan UIN Ar-Raniry
Darussalam Banda Aceh




Prof. Saiful Mulki, S.Ag., M.A., M.Ed., Ph.D.
NIP. 1979010219997031003

LEMBAR PERNYATAAN KEASLIAN KARYA ILMIAH

Yang bertandatangan di bawah ini:

Nama : Teuku Muksal Mina

NIM 190212026

Program Studi : Pendidikan Teknologi Informasi Fakultas

Tarbiyah dan Keguruan Judul Skripsi: **Perancangan Aplikasi Pemilihan Peminatan dan Ujian Komprehensif Mahasiswa Berbasis Website Pada Program Studi PTI**

Dengan ini menyatakan bahwa dalam penulisan skripsi ini, saya:

1. Tidak menggunakan ide orang lain tanpa mampu mengembangkan dan mempertanggungjawabkan.
2. Tidak melakukan plagiat terhadap naskah karya orang lain.
3. Tidak menggunakan karya orang lain tanpa menyebutkan sumber asli atau tanpa izin pemilik karya.
4. Tidak memanipulasi dan memalsukan data.
5. Mengerjakan karya ini dan mampu bertanggung jawab atas karya ini.

Bila dikemudian hari ada tuntutan dari pihak lain atas karya saya, dan telah melalui pembuktian yang dapat dipertanggungjawabkan dan ternyata memang ditemukan bukti bahwa saya telah melanggar pernyataan ini, maka saya siap dikenai sanksi berdasarkan atauran yang berlaku di Fakultas Tarbiyah dan Keguruan UIN Ar-Raniry Banda Aceh. Demikian Surat pernyataan saya buat dengan sesungguhnya.

جامعة الرانيري

AR - RANIRY

Banda Aceh, 16 Desember 2023



Teuku Muksal Mina
NIM. 190212026

ABSTRAK

Nama : Teuku Muksal Mina
NIM : 190212026
Fakultas/Prodi : Tarbiyah dan Keguruan/Pendidikan Teknologi
Informasi Judul : Perancangan Aplikasi Pemilihan Peminatan dan Ujian
Komprehensif Mahasiswa Berbasis Website Pada
Program Studi PTI
Pembimbing I : Ridwan, S.ST. M.T
Pembimbing II : Aulia Syarif Aziz, S.kom., M.Sc.
Kata Kunci : Sistem Pendukung Keputusan, penentuan peminatan,
Simple Additive *Weighting*, *PHP*, *MySQL*

Program studi Pendidikan Teknologi Informasi (PTI) merupakan prodi yang menghasilkan tenaga pendidik sekaligus memiliki kompetensi tenaga profesional IT. Setiap tahun rutin melakukan penentuan peminatan dan ujian komprehensif mahasiswa. Masalah yang dihadapi oleh Prodi PTI yaitu proses penentuan peminatan dan ujian komprehensif tersebut memiliki subjektifitas yang tinggi. Dari permasalahan tersebut, pada penelitian ini dibuat sebuah Aplikasi Ujian Komprehensif dan Peminatan Mahasiswa berbasis website dengan memakai framework Laravel dan pengembangan dengan metode *Rapid Application Development* (RAD). Dalam proses penentuan peminatan, aplikasi ini menggunakan Sistem Pendukung Keputusan (SPK) dengan menerapkan *Metode Simple Additive Weighting* (SAW). Metode SAW menggunakan bobot dari beberapa kriteria, di antaranya adalah nilai rata-rata mata kuliah peminatan, nilai ujian minat, nilai minat, dan pilihan karir. Pengujian fungsi-fungsi pada website pemilihan peminatan dan ujian komprehensif dengan metode Black Box Testing

menghasilkan hasil yang valid. Aplikasi ini didesain untuk memberikan dukungan kepada mahasiswa dalam mengambil keputusan yang tepat berdasarkan kriteria Peminatan dan Ujian Komprehensif.



KATA PENGANTAR

Alhamdulillahirabbil'alamin. Dengan mengucapkan puji syukur kehadiran Allah SWT, yang telah memberi kesehatan, kesempatan, serta taufiq dan hidayah, sehingga penulis dapat menyusun proposal ini. Shalawat serta salam yang tercurahkan kepada baginda Nabi besar Muhammad S.A.W yang merupakan sosok yang amat mulia yang menjadi panutan setiap muslim serta telah membuat perubahan yang besar di dunia ini. Berkat rahmat dan hidayah yang Allah berikan sehingga penulis dapat menyelesaikan skripsi ini yang berjudul: “PERANCANGAN APLIKASI PEMILIHAN PEMINATAN DAN UJIAN KOMPREHENSIF MAHASISWA BERBASIS WEBSITE PADA PROGRAM STUDI PTI”.

Pada kesempatan ini, penulis mengucapkan terimakasih yang sebesar-besarnya kepada:

1. Yang istimewa dan yang tercinta sosok pejuang dibalik layar dan panutan penulis Ayahanda Teuku Idrus, Beliau memang tidak sempat melihat aktivitas dari penulis (sibuk), namun beliau mampu mendidik penulis, memberikan motivasi dan semangat tiada henti dari jarak yang jauh tapi begitu dekat hingga penulis dapat menyelesaikan studi hingga sarjana.
2. Pintu Syurgaku, Ibunda tercinta Mardhiah penulis ucapkan beribu terima kasih telah mengasuh, mendidik, membimbing, membina, memberikan semangat serta doa-doa tulus yang diberikan selama ini. Terimakasih atas motivasi dan nasihat yang diberikan selama ini walaupun terkadang pikiran kita tidak sejalan. Terimakasih atas kebesaran hati dan kesabaran menghadapi sikap kekanak-

kanakan penulis. Ibu adalah sosok yang menjadi penguat dan pengingat paling hebat. Terimakasih sudah menjadi tempat pulang ibu.

3. Bapak Prof Dr. H. Mujiburrahman M.Ag. Selaku Rektor Universitas Islam Negeri (UIN) Ar-Raniry.
4. Bapak Safrul Muluk, MA., M.Ed., Ph.D. Selaku Dekan Fakultas Tarbiyah dan Keguruan Universitas Islam Negeri (UIN) Ar-Raniry
5. Ibu Mira Maisura, M.Sc selaku Ketua Prodi Studi Pendidikan Teknologi Informasi atas kesempatan dan bantuan yang diberikan kepada penulis dalam melakukan penelitian dan memperoleh informasi yang diperlukan selama penulisan skripsi ini.
6. Bapak Ridwan, S.ST. M.T sebagai Dosen Pembimbing I yang telah memberikan arahan dan semangat dalam penyusunan skripsi ini.
7. Bapak Aulia Syarif Aziz, S.kom., M.Sc sebagai Dosen Pembimbing II yang telah memberikan arahan dan semangat dalam penyusunan skripsi ini.
8. Bapak/Ibu Dosen program studi Pendidikan Teknologi Informasi yang telah mendidik dan memberikan bimbingan selama masa perkuliahan.
9. Kepada seluruh keluarga yang telah memberikan semangat dan dukungan kepada penulis sehingga penulis dapat menyelesaikan skripsi ini.
10. Sahabat dan teman-teman Mahasiswa Prodi Pendidikan Teknologi Informasi seangkatan tahun 2019 yang saling bekerja sama dan saling membantu, memberi masukan untuk penulis.

11. Kepada tuan putri pemilik nama Niswatul Afifah sebagai *partner* dan *support system* penulis yang telah kebersamai penulis pada hari-hari yang tidak mudah selama proses pengerjaan tugas akhir, terimakasih telah menjadi rumah tempat berkeluh kesah, menjadi pendengar yang baik, dan banyak berkontribusi dalam penulisan skripsi ini. Mendahulukan kepentinganku, meluangkan waktu, tenaga, pikiran maupun materi. Terimakasih telah menjadi bagian dari perjalananku hingga saat ini. Tetaplah nyaman menjadi kamu, seperti kamu membuat saya nyaman menjadi saya dengan segala kekurangan.

12. Kepada *my brother* Rikki satria dan Faisal fajri dari grup ulti flicker yang senantiasa selalu menemani dan belajar tentang segala hal tentang dunia maupun akhirat.

13. *Last but not least, I wanna thank me, I wanna thank me for believing in me, I wanna thank me for doing al this hard work, I wanna thank me for having no days off, I wanna thank me for never quitting.*

Pada fase perskripsian ini, kadangkala kita lupa akan arti kebaikan, kebenaran, kesetiaan, persahabatan, ketenangan dan cinta, karena terlalu banyak sandiwara yang menghadiri. Tetapi satu hal yang harus diingat, “*Dunia tidak pernah kekurangan orang-orang baik, hanya saja kita yang terlalu sering bertemu dengan orang jahat.*” Dedi Irawan

Meskipun telah berusaha menyelesaikan Skripsi ini sebaik mungkin, penulis menyadari bahwa skripsi ini masih ada kekurangan. Oleh karena itu, penulis mengharapkan kritik dan saran yang membangun dari para pembaca guna

menyempurnakan segala kekurangan dalam penyusunan Skripsi ini. Akhir kata, penulis berharap semoga Skripsi ini berguna bagi para pembaca dan pihak-pihak lain yang berkepentingan. Semoga Allah SWT meridhai penulisan ini dan senantiasa memberikan Rahmat dan hidayah-Nya kepada kita semua. Aamiin ya rabbal'amin

Banda Aceh, 16 Desember 2023

Penulis,

Teuku Muksal Mina



DAFTAR ISI

HALAMAN SAMPUL JUDUL	
LEMBAR PENGESAHAN PEMBIMBING	
LEMBAR PENGESAHAN SIDANG	
LEMBAR PERNYATAAN KEASLIAN	
ABSTRAK	i
KATA PENGANTAR	iii
DAFTAR ISI	vii
DAFTAR GAMBAR	x
DAFTAR TABEL	xi
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah.....	3
1.3 Tujuan Penelitian	3
1.4 Batasan Penelitian.....	4
1.5 Manfaat Penelitian	4
1.6 Relevansi Penelitian Terdahulu	5
BAB II LANDASAN TEORI	7
2.1 Metode <i>Rapid Application Development</i> (RAD)	7
2.1.1 <i>Requirement Planning</i> (Rencana Kebutuhan).....	8
2.1.2 <i>User Design</i> (Desain Pengguna).....	8
2.1.3 <i>Construction</i> (Pengembangan).....	9
2.1.4 <i>Cutover</i> (Implementasi dan Pengujian).....	9
2.2 Framework Laravel.....	9
2.3 Pengertian Aplikasi.....	11
2.4 Pengertian Aplikasi Berbasis Web.....	11
2.5 Konsep MVC (<i>Model-View-Controller</i>).....	12
2.6 PHP	13
2.7 <i>JavaScript</i>	14
2.7.1 <i>AJAX</i>	15

2.8 Database MySQL.....	15
2.9 Visual Studio Code.....	16
2.10 Sistem Pendukung Keputusan.....	16
2.11 Simple Additive Weighthing (SAW)	16
2.12 Kriteria dan Bobot.....	19
BAB III METODOLOGI PENELITIAN	20
3.1 Tahapan Penelitian.....	20
3.2 Analisa Kebutuhan.....	21
3.2.1 Kebutuhan Proses.....	21
3.2.2 Kebutuhan Data	21
3.2.3 Kebutuhan Output.....	21
3.2.4 Kebutuhan Fungsional	21
3.2.5 Kebutuhan Perangkat Lunak (<i>Software</i>).....	22
3.2.6 Kebutuhan Perangkat Keras (<i>Hardware</i>).....	22
3.2.7 Metode Pengumpulan Data.....	23
3.3 Perancangan Sistem	23
3.3.1 Use Case Diagram.....	23
3.3.2 Activity Diagram	25
3.3.3 Data Model.....	36
3.4 Impelementasi <i>Simple Additive Weighting (SAW)</i>	41
3.5 Perancangan Aplikasi.....	43
BAB IV HASIL DAN PEMBAHASAN	48
3.6 Hasil Implementasi dan Pembahasan.....	48
3.6.1 Tampilan Login dan Dashboard.....	48
3.6.2 Tampilan Assesment dan CetakPDF.....	50
3.6.3 Tampilan List Ujian dan Pin Ujian	52
3.6.4 Tampilan Ujian dan Hasil Ujian	53
3.7 Pengujian sistem	54
3.7.1 Hasil Pengujian sistem.....	55
BAB V KESIMPULAN DAN SARAN	60

5.1 Kesimpulan	60
5.2 Saran	61
DAFTAR PUSTAKA	62
RIWAYAT HIDUP PENULIS.....	65

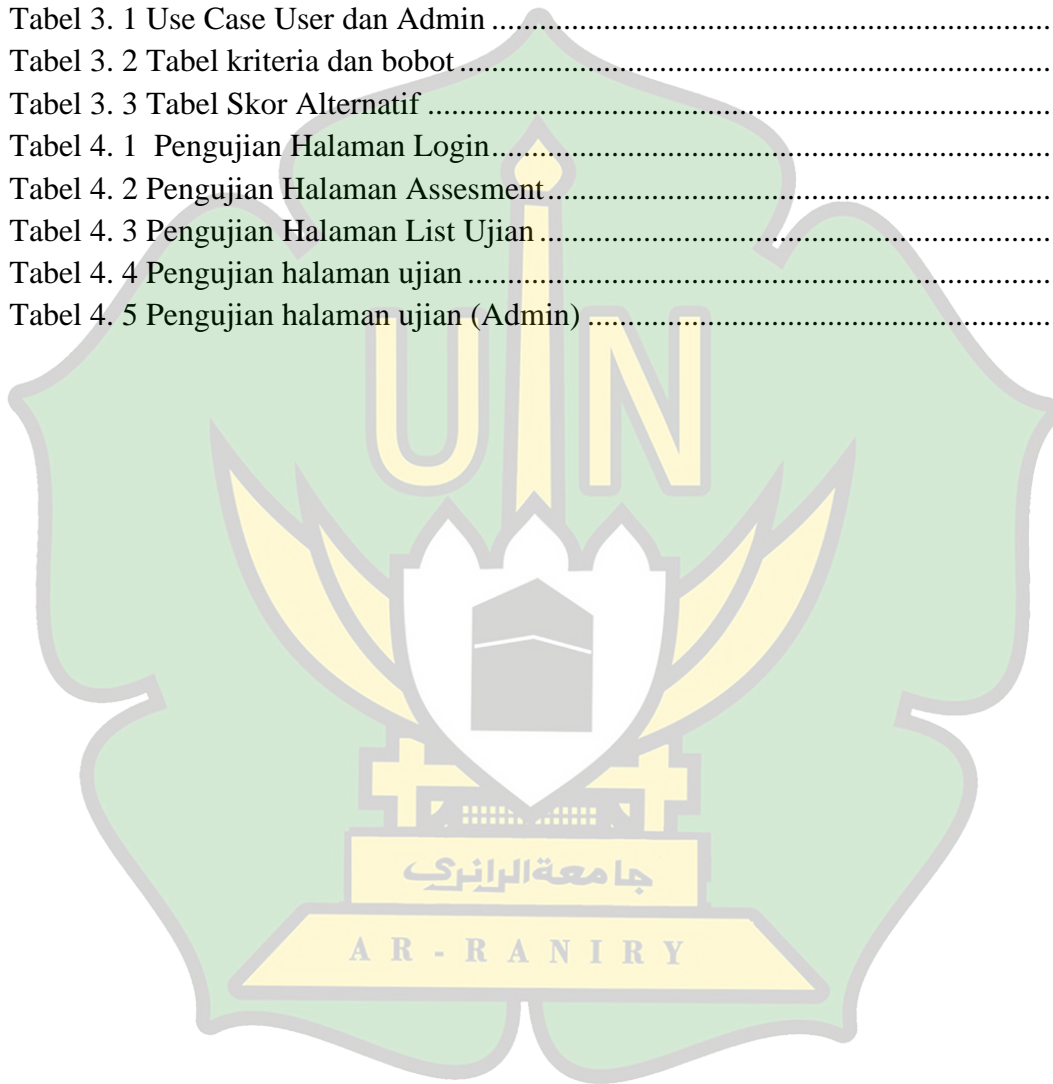


DAFTAR GAMBAR

Gambar 2. 1 Tahapan Metode RAD	7
Gambar 3. 1 Tahapan Penelitian	20
Gambar 3. 2 Use Case Diagram SPK PTI.....	24
Gambar 3. 3 Activity Diagram user login.....	26
Gambar 3. 4 Activity Diagram User Assesment.....	27
Gambar 3. 5 Activity Diagram User Ujian	29
Gambar 3. 6 Activity Diagram Admin Mengelola Soal	31
Gambar 3. 7 Activity Diagram Admin Mengelola Ujian.....	32
Gambar 3. 8 Activity Diagram Admin Mengelola Users	33
Gambar 3. 9 Activity Diagram Admin Mengelola Peminatan.....	34
Gambar 3. 10 Data Model.....	36
Gambar 3. 11 Halaman <i>Login</i>	43
Gambar 3. 12 Halaman <i>Frontend, Backend</i>	44
Gambar 3. 13 Halaman <i>Home</i>	44
Gambar 3. 14 Halaman <i>List Ujian</i>	45
Gambar 3. 15 Halaman <i>Assessment</i>	45
Gambar 3. 16 Halaman <i>Ujian</i>	46
Gambar 3. 17 Halaman <i>Hasil</i>	46
Gambar 3. 18 Halaman <i>Admin, List Ujian, Tambah</i>	47
Gambar 4. 1 Halaman <i>Login dan Dashboard</i>	48
Gambar 4. 2 Halaman <i>Assessment</i>	50
Gambar 4. 3 Halaman <i>CetakPDF</i>	52
Gambar 4. 4 Halaman <i>List Ujian</i>	52
Gambar 4. 5 Halaman <i>Memasukkan Pin Ujian</i>	53
Gambar 4. 6 Halaman <i>Ujian</i>	53
Gambar 4. 7 Halaman <i>Hasil Ujian</i>	54

DAFTAR TABEL

Tabel 1. 1 Penelitian Terdahulu	5
Tabel 2. 1 Tabel kriteria dan bobot	19
Tabel 3. 1 Use Case User dan Admin	24
Tabel 3. 2 Tabel kriteria dan bobot	41
Tabel 3. 3 Tabel Skor Alternatif	42
Tabel 4. 1 Pengujian Halaman Login.....	55
Tabel 4. 2 Pengujian Halaman Assesment.....	55
Tabel 4. 3 Pengujian Halaman List Ujian	56
Tabel 4. 4 Pengujian halaman ujian	57
Tabel 4. 5 Pengujian halaman ujian (Admin)	58



BAB I

PENDAHULUAN

1.1 Latar Belakang

Era digitalisasi memiliki pengaruh yang signifikan terhadap teknologi. Digitalisasi telah mengubah cara kita berinteraksi, bekerja, dan berkomunikasi secara fundamental. Komputer dan internet menjadi lebih cepat, lebih kuat, dan lebih terjangkau, sehingga memungkinkan pengembangan teknologi baru seperti kecerdasan buatan (AI), *machine learning*, dan *blockchain* [1]. Dalam keseluruhan, era digitalisasi telah mempercepat perkembangan teknologi dan memungkinkan pengembangan solusi baru yang lebih inovatif. Hal ini juga telah memperluas jangkauan teknologi dan memberikan akses yang lebih luas ke informasi dan aplikasi.

Program studi Pendidikan Teknologi Informasi (PTI) merupakan prodi yang menghasilkan tenaga pendidik sekaligus memiliki kompetensi tenaga profesional IT, entrepreneur, kompetensi pedagogik dan kompetensi sosial. Semua pelaku pendidikan, mulai dari mereka yang berada di tingkat kebijakan, manajemen, hingga pelaksana seperti guru mengacu pada tujuan pendidikan nasional sebagaimana dijelaskan dalam Pasal 3 Undang-Undang Sistem Pendidikan Nasional Nomor 20 Tahun 2003. Tujuan tersebut adalah menciptakan peserta didik yang menjadi individu yang beriman dan bertaqwa kepada Tuhan Yang Maha Esa, berakhlak mulia, sehat, berilmu, cakap, kreatif, mandiri, serta menjadi warga

negara yang demokratis dan bertanggung jawab [2].

Prodi Pendidikan Teknologi Informasi memiliki tiga peminatan atau jurusan, yaitu Multimedia (MM), Rekayasa Perangkat Lunak (RPL), dan Teknik Jaringan Komputer (TKJ). Mahasiswa wajib untuk mengikuti tes peminatan untuk memilih salah satu bidang. Pada saat ini pemilihan peminatan dilaksanakan secara manual, dimana mahasiswa saat memilih hanya berkonsultasi dengan dosen pembimbing akademik yaitu disesuaikan dengan pencapaian semester.

Pelaksanaan ujian komprehensif juga merupakan tahapan penting dalam perjalanan akademik mahasiswa, ujian komprehensif bertujuan mengukur pemahaman mendalam terhadap materi pembelajaran selama masa studi. Namun, proses manual pelaksanaan ujian tersebut dapat memakan waktu dan berpotensi mengalami kesalahan administratif.

Berdasarkan permasalahan tersebut, pada penelitian ini dirancang Aplikasi Ujian Komprehensif dan Sistem Pendukung Keputusan (SPK). SPK ini akan menggunakan metode *Simple Additive Weighting* (SAW) dan akan disesuaikan dengan proses peminatan yang ada di Program Studi Pendidikan Teknologi Informasi. Untuk menentukan peminatan mahasiswa, beberapa kriteria digunakan, seperti nilai rata-rata mata kuliah peminatan dari semester 1 hingga 4, nilai ujian minat, nilai pilihan minat dan karir di masa depan. Kriteria ini sesuai dengan karakteristik metode SAW, yang merupakan salah satu metode sederhana dalam sistem pengambilan keputusan yang dapat menangani masalah yang tidak terstruktur [3].

1.2 Rumusan Masalah

Rumusan masalah penelitian adalah sebagai berikut :

1. Bagaimana merancang aplikasi peminatan dan ujian komprehensif berbasis web ?
2. Bagaimana menerapkan sistem pendukung keputusan (SPK) dalam penentuan peminatan pada Program Studi Pendidikan Teknologi Informasi ?
3. Sejauh mana validitas fungsi-fungsi yang diuji pada aplikasi peminatan dan ujian komprehensif berbasis web menggunakan *Black box testing* ?

1.3 Tujuan Penelitian

Tujuan penelitian adalah sebagai berikut :

1. Merancang aplikasi penentuan peminatan mahasiswa dan ujian komprehensif.
2. Mengimplementasikan SPK menggunakan metode SAW pada penentuan bidang peminatan.
3. Melakukan pengujian fungsi - fungsi pada aplikasi dengan Black box testing.

1.4 Batasan Penelitian

Adapun batasan masalah pada penelitian ini, yaitu:

1. Implementasi aplikasi dan situs web akan terbatas pada pemilihan peminatan dan pelaksanaan ujian komprehensif di PTI.
2. Kriteria yang digunakan dalam spk yaitu nilai ujian minat, nilai rata – rata mata kuliah bidang minat dan pilihan minat beserta pilihan karir di masa depan.
3. Pengembangan aplikasi website menggunakan framework Laravel.

1.5 Manfaat Penelitian

Penelitian ini diharapkan memberikan kontribusi positif dalam meningkatkan efisiensi pemilihan peminatan dan ujian komprehensif mahasiswa di Program Studi Pendidikan Teknologi Informasi di Universitas Islam Negeri Ar-Raniry. Aplikasi yang dikembangkan diharapkan dapat menyederhanakan proses tersebut, meningkatkan akurasi, dan secara keseluruhan, meningkatkan kualitas pendidikan di program studi tersebut. Dengan aplikasi ini, mahasiswa diharapkan dapat lebih fokus pada pemahaman materi dan pengembangan keterampilan sesuai dengan bidang peminatan mereka, berdampak positif pada perkembangan akademik dan profesional di masa mendatang.

1.6 Relevansi Penelitian Terdahulu

Tabel 1. 1 Penelitian Terdahulu

No	Nama Penulis	Judul	Tahun	Uraian Penulis
1	Muhammad Dwika Ilyas Ruhyat	Rancang bangun Aplikasi ujian online untuk siswa Smp menggunakan framework laravel (studi kasus : smp negeri 4 cikalongkulon)	2019	Membangun aplikasi ujian online berbasis web menggunakan metode Waterfall dan bahasa pemrograman PHP dengan framework Laravel, siswa dapat mengakses ujian secara elektronik.
2	Hazar Hamzah	Rancang bangun Aplikasi pembelajaran online menggunakan Framework laravel (studi kasus : web programming unpas)	2019	Membangun aplikasi berbasis web yang mengatasi kendala pengelolaan materi dengan memanfaatkan framework laravel dan menggunakan metode waterfall
3	Mahendar Dwi Payana dan M. Bayu Wibawa	Sistem Ujian Online Tes Masuk Universitas Ubudiyah Indonesia Bagi Calon Mahasiswa Baru Berbasis Web Menggunakan CI (Code Igniter)	2020	Pembuatan sistem ujian online berbasis web menggunakan framework Code Igniter.

Penelitian ini dikembangkan dan disesuaikan dengan studi kasus sebelumnya. Kriteria yang digunakan untuk perhitungan SAW disesuaikan dengan prosedur yang ada di Program Studi Pendidikan Teknologi Informasi Universitas Islam Negeri Ar-Raniry. Salah satu perkembangan baru dalam penelitian ini adalah perancangan dengan metode *Rapid Application Development (RAD)*, penggunaan

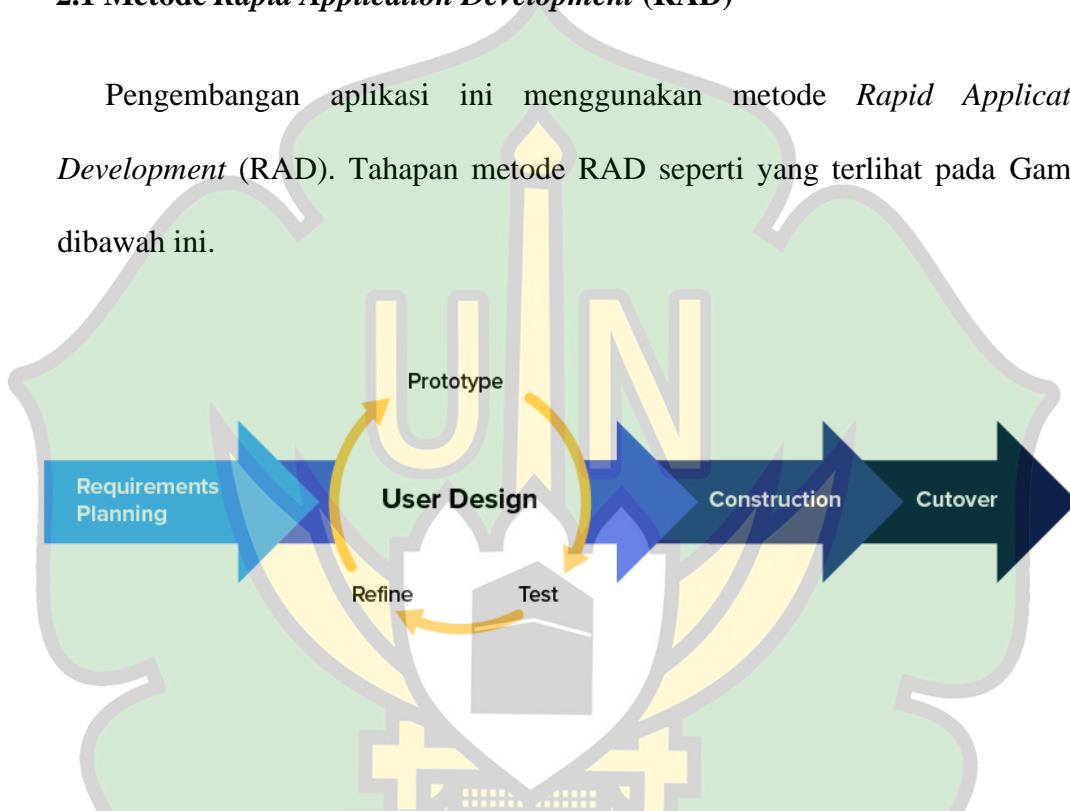
framework Laravel, Bahasa pemrograman *Javascript/Ajax*, ujian komprehensif dan penerapan ujian minat sebagai kriteria pemilihan peminatan.



BAB II LANDASAN TEORI

2.1 Metode *Rapid Application Development* (RAD)

Pengembangan aplikasi ini menggunakan metode *Rapid Application Development* (RAD). Tahapan metode RAD seperti yang terlihat pada Gambar dibawah ini.



Gambar 2. 1 Tahapan Metode RAD

Metode *Rapid Application Development* (RAD) adalah suatu model proses pembangunan perangkat lunak yang termasuk dalam kategori teknik incremental atau bertahap. Fokus utama dari tahapan-tahapan dalam metode RAD adalah untuk mempercepat dan menyederhanakan siklus pengembangan [4]. RAD menerapkan pendekatan berulang dalam mengembangkan sistem, di mana model sistem dibangun pada awal tahap pengembangan untuk mengidentifikasi

kebutuhan (*requirement*) [5]. Proses ini akan merinci tahapan perancangan sistem. Pemilihan metode *Rapid Application Development* (RAD) dipengaruhi oleh beberapa keunggulan, seperti siklus pengembangan yang lebih cepat, fleksibilitas yang lebih tinggi, dan potensi pengurangan kesalahan. Berikut adalah beberapa tahapan pengembangan sistem yang dijelaskan dalam metode RAD:

2.1.1 *Requirement Planning* (Rencana Kebutuhan)

Pada fase ini, kolaborasi antara pengguna dan pengembang dilakukan untuk menyelesaikan masalah yang dihadapi dan menetapkan kebutuhan apa saja yang diperlukan untuk menciptakan sistem aplikasi. Tahap ini merupakan langkah awal yang krusial untuk kesuksesan pengembangan sistem dan mencegah terjadinya kesalahan komunikasi antara pengguna dan pengembang.

2.1.2 *User Design* (Desain Pengguna)

Langkah ini melibatkan pembuatan rencana desain yang diusulkan untuk memenuhi kebutuhan, mengikuti rencana, dan bertujuan untuk mengatasi permasalahan yang dihadapi. Dalam penelitian ini, sistem dijelaskan melalui penggunaan *Unified Modeling Language* (UML) [6].

2.1.3 *Construction* (Pengembangan)

Pada fase ini, pengembang memulai pembuatan sistem sesuai dengan perencanaan. Ini dimulai dengan penyusunan kode program, atau biasa disebut coding, dan mengubah desain sistem yang telah dibuat menjadi aplikasi yang dapat digunakan.

2.1.4 *Cutover* (Implementasi dan Pengujian)

Pada langkah ini, semua komponen sistem yang telah dikembangkan perlu mengalami pengujian *Black Box* secara menyeluruh dengan tujuan mengurangi potensi risiko kegagalan sistem. *Black Box Testing* merupakan salah satu metode pengujian perangkat lunak yang berorientasi pada spesifikasi fungsional dari perangkat lunak[7].

2.2 Framework Laravel

Laravel adalah kerangka kerja PHP yang dirilis di bawah lisensi MIT yang berfungsi untuk membuat situs web berbasis *Model View Presenter* (MVP) yang ditulis dalam PHP. [8]. Tujuannya adalah untuk meningkatkan kualitas perangkat lunak dengan mengurangi biaya pembuatan dan perawatan awal serta meningkatkan pengalaman pengguna dengan memberikan sintaks yang cepat, jelas, dan efisien.

MVC adalah pendekatan perangkat lunak yang membedakan logika aplikasi dari presentasi. Ini mengorganisir aplikasi berdasarkan komponen seperti

pengontrolan (*Controller*), tampilan ke pengguna (*View*), dan manipulasi data (*Model*). [9].

Beberapa fitur yang terdapat di Laravel antara lain:

- *Bundles*: Fitur ini menyediakan sistem pengemasan modular dengan beragam aplikasi.
- *Eloquent ORM*: Implementasi PHP lanjutan yang menyediakan metode internal dari pola "active record" untuk mengatasi masalah hubungan objek database.
- *Application Logic*: Bagian aplikasi yang menggunakan controller atau bagian Route untuk mengatur logika.
- *Reverse Routing*: Mendefinisikan relasi antara Link dan Route.
- *Restful controllers*: Memisahkan logika dalam melayani HTTP GET dan POST.
- *Class Auto Loading*: Menyediakan loading otomatis untuk kelas PHP.
- *View Composer*: Kode unit logika yang dapat dieksekusi ketika view sedang dimuat.

Sebuah kerangka kerja (*framework*) digunakan oleh pengembang untuk membuat pembangunan aplikasi web lebih mudah dengan menyediakan *repository* yang berisi *class-class*, fungsi, dan alat. Solusi untuk akses *database*, *autentikasi*, *templating*, *control*, dan fungsi lainnya dapat termasuk dalam *framework*. Diharapkan bahwa penggunaan *framework* akan membuat pengembangan aplikasi

lebih teratur dan rapi, dengan struktur yang ideal dan dapat digunakan kembali. Struktur MVC digunakan untuk memisahkan logika bisnis dan presentasi, tujuannya adalah agar perubahan pada logika presentasi atau bisnis tidak mempengaruhi satu sama lain secara kompleks.

2.3 Pengertian Aplikasi

"Sebuah aplikasi merupakan kumpulan atribut yang terdiri dari beberapa formulir dan laporan yang diatur dengan baik sehingga dapat mengakses data". Didasarkan pada definisi tersebut, aplikasi adalah perangkat lunak yang digunakan di komputer dan memiliki set instruksi yang dirancang untuk melakukan berbagai tugas atau tugas. Misalnya, mereka dapat menerapkan, menggunakan, dan menambah data [10].

2.4 Pengertian Aplikasi Berbasis Web

Aplikasi web adalah sebuah program yang menggunakan teknologi browser untuk beroperasi dan dapat diakses melalui jaringan komputer [11]. Menurut definisi lain dari Rouse (2011), aplikasi web merupakan program yang disimpan di server, dikirim melalui internet, dan diakses melalui antarmuka browser. Dengan demikian, dapat disimpulkan bahwa aplikasi web adalah perangkat lunak yang dapat diakses melalui web browser menggunakan jaringan internet atau intranet. Selain itu, aplikasi web juga merupakan perangkat lunak komputer yang dikodekan dalam bahasa pemrograman yang mendukung pengembangan perangkat lunak

berbasis web, seperti *HTML*, *JavaScript*, *CSS*, *Ruby*, *Python*, *PHP*, *Java*, dan berbagai bahasa lainnya.

2.5 Konsep MVC (*Model-View-Controller*)

Model View Controller (MVC) adalah suatu konsep yang populer dalam pengembangan aplikasi web, berasal dari bahasa pemrograman *Smalltalk* [12]. MVC memisahkan pengembangan aplikasi menjadi komponen utama, seperti manipulasi data, antarmuka pengguna, dan kontrol aplikasi [13]. Terdapat tiga jenis komponen yang membentuk pola MVC dalam suatu aplikasi:

- *Model*: Biasanya terkait langsung dengan database untuk melakukan manipulasi data (*insert*, *update*, *delete*, *search*) dan menangani validasi dari bagian *controller*. Namun, model tidak dapat berinteraksi langsung dengan bagian *view*.
- *View*: Bagian ini menangani logika presentasi. Pada aplikasi web, view sering berupa file template HTML yang diatur oleh controller. View bertanggung jawab menerima dan menyajikan data kepada pengguna. Bagian ini tidak memiliki akses langsung ke bagian model.
- *Controller*: Bagian yang mengelola hubungan antara *model* dan *view*. *Controller* menerima permintaan dan data dari pengguna, kemudian menentukan proses apa yang akan dilakukan oleh aplikasi.

Secara umum, alur kontrol arsitektur MVC melibatkan langkah-langkah berikut:

- Pengguna berinteraksi dengan antarmuka pengguna melalui beberapa cara, misalnya, dengan menekan tombol *mouse*.
- *Controller* menangani event input dari antarmuka pengguna, seringkali melalui *handler* atau *callback* yang terdaftar. *Controller* kemudian mengonversi event tersebut menjadi tindakan pengguna yang dapat dimengerti oleh model.
- *Controller* memberitahu model tentang tindakan pengguna yang dapat mengubah keadaan model. Sebagai contoh, controller dapat memperbarui keranjang belanja pengguna.
- *Model* menghasilkan antarmuka pengguna dengan benar melalui pertanyaan yang sesuai, seperti daftar konten yang ada di keranjang belanja. Data view diambil dari model, dan dalam beberapa implementasi, controller memerintahkan view untuk menghasilkan hasil pertanyaan. Selain itu, ada model yang secara otomatis memberi tahu tampilan tentang perubahan keadaan yang memerlukan pembaruan tampilan..
- Siklus alur kontrol dimulai kembali setelah pengguna menunggu komentar tambahan.

2.6 PHP

PHP adalah bahasa pemrograman *server-side* yang dirancang khusus untuk pengembangan aplikasi web. Dalam lingkungan server, PHP berfungsi untuk memproses kode pada sisi server sebelum mengirimkan hasilnya ke klien (biasanya dalam bentuk HTML yang sudah di-generate).PHP memiliki sintaks yang mirip

dengan bahasa pemrograman C, dan telah menjadi salah satu bahasa pemrograman yang populer dalam pengembangan web. Ia dapat digunakan untuk berbagai tujuan, seperti membuat halaman web dinamis, mengakses dan mengelola basis data, berkomunikasi dengan server lain melalui protokol HTTP, dan banyak lagi [14]

2.7 JavaScript

JavaScript merupakan bahasa skrip yang populer di dunia internet dan dapat beroperasi pada sebagian besar browser terkenal, seperti *Internet Explorer (IE)*, *Mozilla*, *FireFox*, *Netscape*, dan *Opera*. Kode JavaScript dapat dimasukkan ke dalam halaman web menggunakan tag *SCRIPT* [15].

Beberapa informasi mengenai *JavaScript* meliputi:

- *JavaScript* dirancang untuk meningkatkan interaktivitas suatu situs web.
- *JavaScript* merupakan jenis bahasa skrip.
- Bahasa skrip adalah jenis bahasa pemrograman yang ringan.
- Kode *JavaScript* dieksekusi di komputer (web browser).
- *JavaScript* umumnya ditanamkan (*embedded*) dalam halaman HTML.
- *JavaScript* berfungsi sebagai bahasa interpreter, yang berarti skrip dieksekusi tanpa melalui proses kompilasi.
- Penggunaan *JavaScript* tidak memerlukan pembayaran lisensi dan dapat diakses oleh siapa saja.

2.7.1 AJAX

AJAX (Asynchronous JavaScript And XML) merupakan suatu teknik pemrograman berbasis web yang bertujuan untuk meningkatkan interaktivitas dan dinamika aplikasi web. Teknik ini mengarah pada pemindahan sebagian besar interaksi antara pengguna dan server web ke latar belakang, memungkinkan pertukaran data dengan server tanpa perlu memuat ulang seluruh halaman web setiap kali pengguna melakukan perubahan [16].

Penggunaan *AJAX* memberikan beberapa keuntungan, antara lain:

- **Interaktivitas Tinggi:** Aplikasi yang menggunakan *AJAX* menjadi lebih interaktif dibandingkan dengan aplikasi web konvensional.
- **Usabilitas Tinggi:** Pembaruan data tidak memerlukan pembacaan ulang seluruh halaman, melainkan hanya yang relevan untuk meningkatkan pengalaman pengguna.
- **Kecepatan Tinggi:** Aplikasi *AJAX* memiliki kinerja lebih cepat jika dibandingkan dengan aplikasi web konvensional.

2.8 Database MySQL

Database adalah suatu pengolah data untuk memudahkan penyimpanan dan pengaksesan data. MySQL adalah suatu database server yang berfungsi untuk menangani database. MySQL adalah salah satu jenis database server terkenal yang banyak digunakan untuk membuat aplikasi sumber dan pengolahan data yang mudah diakses dan digunakan dengan cepat, dan bersifat *open-source* [17].

2.9 Visual Studio Code

Visual Studio Code merupakan kode editor yang ringan namun kuat yang di jalankan di *desktop* dan tersedia untuk *Windows*, *macOS* dan *Linux*. Muncul dengan dukungan bawaan untuk *JavaScript*, *TypeScript* dan *Node.js* dan memiliki ekstensi yang banyak untuk bahasa lain seperti *C ++*, *C #*, *Java*, *Python*, *PHP*, dan lain sebagainya [18].

2.10 Sistem Pendukung Keputusan

Dari beberapa definisi yang telah ditemukan mengenai pengambilan keputusan, dapat disimpulkan bahwa proses pengambilan keputusan dalam sebuah organisasi merupakan hasil dari komunikasi dan partisipasi yang berkelanjutan dari seluruh anggota organisasi. Esensi dari pengambilan keputusan adalah memilih di antara berbagai alternatif tindakan yang mungkin, dengan menggunakan mekanisme tertentu, dengan harapan dapat menghasilkan keputusan terbaik. Proses pengambilan keputusan melibatkan tahapan prosedur, dimulai dengan menemukan masalah, mendefinisikan tujuan yang diinginkan, menilai berbagai cara untuk mencapainya, dan mengakhiri proses dengan menetapkan tindakan. [19].

2.11 Simple Additive Weighting (SAW)

Metode yang dipilih untuk merancang sistem guna menentukan peminatan mahasiswa adalah metode *Simple Additive Weighting (SAW)* [20]. Keunggulan metode ini terletak pada sederhananya, serta kemudahan penerapannya berkat

algoritmanya yang tidak terlalu rumit. Prinsip dasar SAW melibatkan penjumlahan terbobot dari rating kinerja pada setiap alternatif untuk semua atribut yang relevan. Tahap awal dalam metode ini adalah proses menormalisasi matriks (X) untuk membawa data ke dalam skala yang dapat dibandingkan dengan semua rating alternatif yang tersedia. SAW dikenal sebagai metode yang sering digunakan dalam situasi *Multiple Attribute Decision Making* (MADM).

MADM merupakan pendekatan untuk menemukan alternatif optimal dari sejumlah opsi dengan kriteria yang ditetapkan, memerlukan penentuan bobot untuk setiap atribut [21]. Nilai akhir untuk setiap opsi didapat dengan mengalikan nilai bobot sebelumnya dengan nilai rating kinerja yang telah dinormalisasi dari matriks. Untuk normalisasi matriks, nilai atribut alternatif dibagi dengan atribut yang sesuai berdasarkan jenisnya. Terdapat dua jenis atribut, yaitu keuntungan/benefit (maksimum) atau biaya/cost (minimum). Dalam kriteria keuntungan, nilai atribut kriteria dari setiap kolom dibagi dengan nilai maksimum ($\text{Max } X_{ij}$) dari setiap kolom. Sebaliknya, dalam kriteria biaya, nilai atribut kriteria dari setiap kolom dibagi dengan nilai minimum atribut kriteria ($\text{Min } X_{ij}$).

Diberikan persamaan berikut :

$$r_{ij} = \begin{cases} \frac{x_{ij}}{\text{Max}_i x_{ij}} \\ \frac{x_{ij}}{\text{Min}_i x_{ij}} \end{cases}$$

Jika j adalah atribut keuntungan (benefit)

Jika j adalah atribut biaya (cost)

Dimana r_{ij} adalah rating kinerja ternormalisasi dari alternatif A_i pada atribut C_j ,
 $i=1,2,\dots,m$ $j=1,2,\dots,n$.

Nilai preferensi untuk setiap alternatif (V_i) diberikan sebagai berikut :

$$V_i = \sum_{j=1}^n w_j r_{ij}$$

Keterangan :

W_j = nilai bobot dari setiap kriteria

R_{ij} = rating kinerja ternormalisasi

X_{ij} = nilai atribut yang dimiliki setiap kriteria

Keuntungan = jika nilai terbesar adalah terbaik (Benefit)

Biaya = jika nilai terkecil adalah terbaik (Cost)

Max X_{ij} = nilai terbesar dari setiap kriteria

Min X_{ij} = nilai terkecil dari setiap kriteria

V_i = ranking untuk setiap alternatif

Nilai V_i yang lebih tinggi menunjukkan preferensi yang lebih tinggi terhadap alternatif A_i . Beberapa langkah yang diusulkan oleh Fishburn dan MacCrimmon untuk menerapkan metode *Simple Additive Weighting* (SAW) yakni:

1. Menetapkan standar yang menjadi dasar pengambilan keputusan, yaitu C_i .
2. Menghitung nilai kecocokan dari setiap alternatif untuk masing-masing kriteria.
3. Membuat matriks keputusan berdasarkan kriteria (C_i), kemudian melakukan normalisasi matriks dengan mengubah persamaan sesuai dengan jenis atribut (keuntungan atau biaya). Prosedur ini menghasilkan matriks normalisasi R .
4. Untuk mendapatkan hasil akhir, proses penjumlahan matriks ternormalisasi R dengan vektor bobot digunakan untuk mendapatkan nilai tertinggi. Nilai ini kemudian dipilih sebagai alternatif terbaik (A_i) sebagai solusi

2.12 Kriteria dan Bobot

Tabel berikut menunjukkan jumlah kriteria dan nilai bobot yang diperlukan untuk melakukan perhitungan dengan metode *Simple Additive Weighting* (SAW):

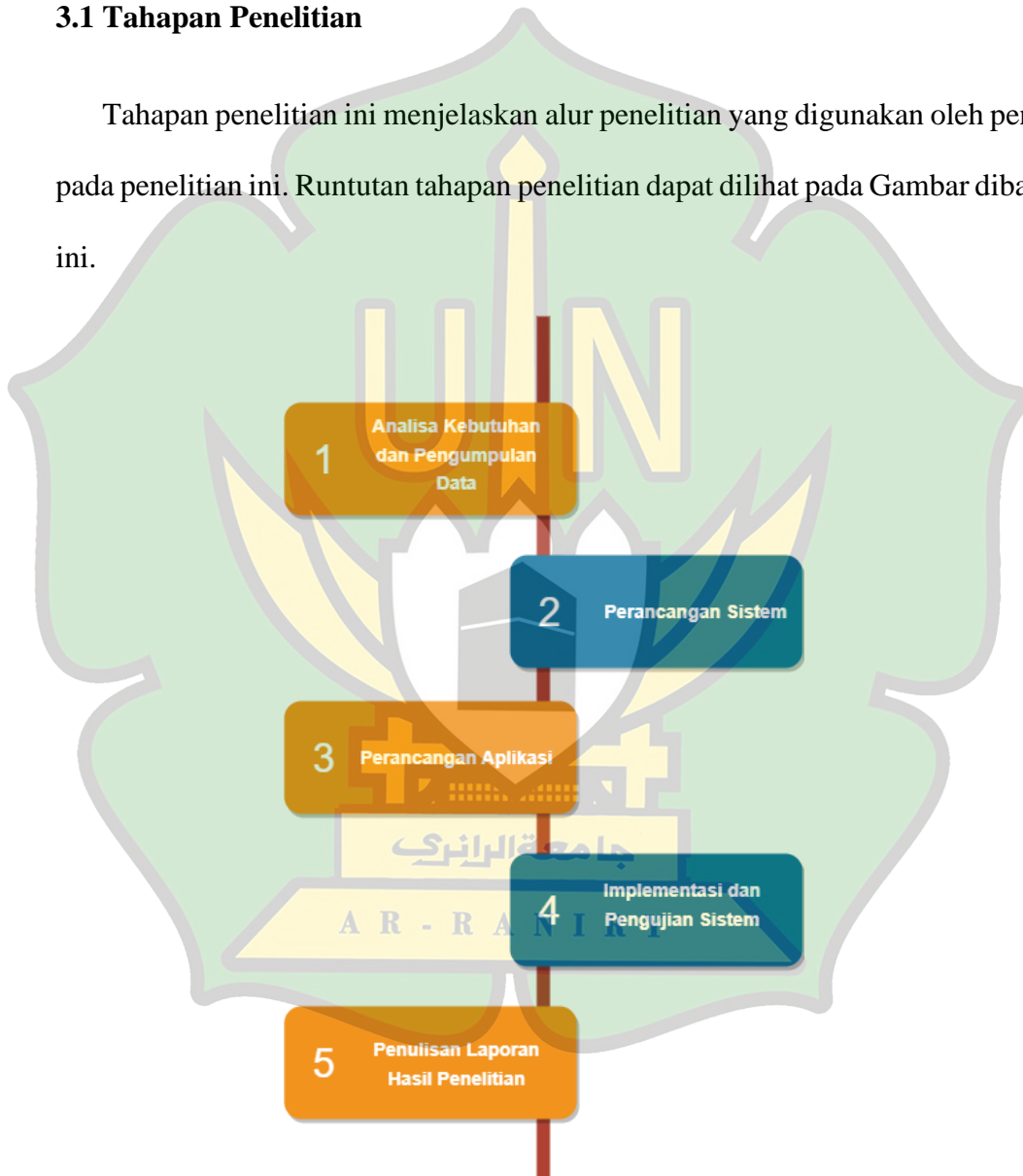
Tabel 2. 1 Tabel kriteria dan bobot

Kode Kriteria	Kriteria	Bobot
C1	Ujian Minat	40%
C2	Nilai rata – rata mata kuliah bidang minat	50%
C3	Pilihan minat dan pilihan karir di masa depan	10%

BAB III METODOLOGI PENELITIAN

3.1 Tahapan Penelitian

Tahapan penelitian ini menjelaskan alur penelitian yang digunakan oleh penulis pada penelitian ini. Runtutan tahapan penelitian dapat dilihat pada Gambar dibawah ini.



Gambar 3. 1 Tahapan Penelitian

3.2 Analisa Kebutuhan

Penerapan metode *Rapid Application Development* (RAD) dalam membuat aplikasi ini diawali dengan tahap perencanaan syarat dan kebutuhan yaitu dengan menganalisa kebutuhan Admin dan User. Tahap ini adalah hasil analisis kebutuhan sistem, kebutuhan data dan lain-lain.

3.2.1 Kebutuhan Proses

Rekomendasi berdasarkan data—data kebutuhan pengguna yang akan diolah untuk memberikan saran yang tepat bagi pengguna—dikenal sebagai kebutuhan proses.

3.2.2 Kebutuhan Data

Kebutuhan Data pada aplikasi ini yaitu mahasiswa UIN, Mata kuliah setiap peminatan dan soal.

3.2.3 Kebutuhan Output

Kebutuhan output yang ingin dihasilkan dari sistem yaitu hasil rekomendasi peminatan dari sistem dan juga hasil nilai ujian komprehensif.

3.2.4 Kebutuhan Fungsional

- Halaman Login untuk semua user
- Halaman Dashboard untuk semua user
- Halaman List Ujian untuk semua user
- Halaman Assesment untuk semua user

- Halaman Dashboard untuk admin
- Halaman Users untuk admin
- Halaman Peminatan untuk admin
- Halaman Laporan untuk admin
- Halaman Ujian untuk admin
- Halaman Bank Soal untuk admin

3.2.5 Kebutuhan Perangkat Lunak (*Software*)

Beberapa perangkat lunak yang di butuhkan di dalam penelitian ini adalah sebagai berikut :

- a. *Laravel*
- b. *Visual Studio Code*
- c. Bahasa pemograman PHP dan *Javascript*
- d. *Database MYSQL*

3.2.6 Kebutuhan Perangkat Keras (*Hardware*)

Perangkat keras yang di perlukan untuk penelitian ini adalah

- Sebuah laptop yang memiliki spesifikasi berikut:
- Prosesor : 2,4 GHz Dual-Core.
- RAM : 8 GB.
- SSD : 256 GB.

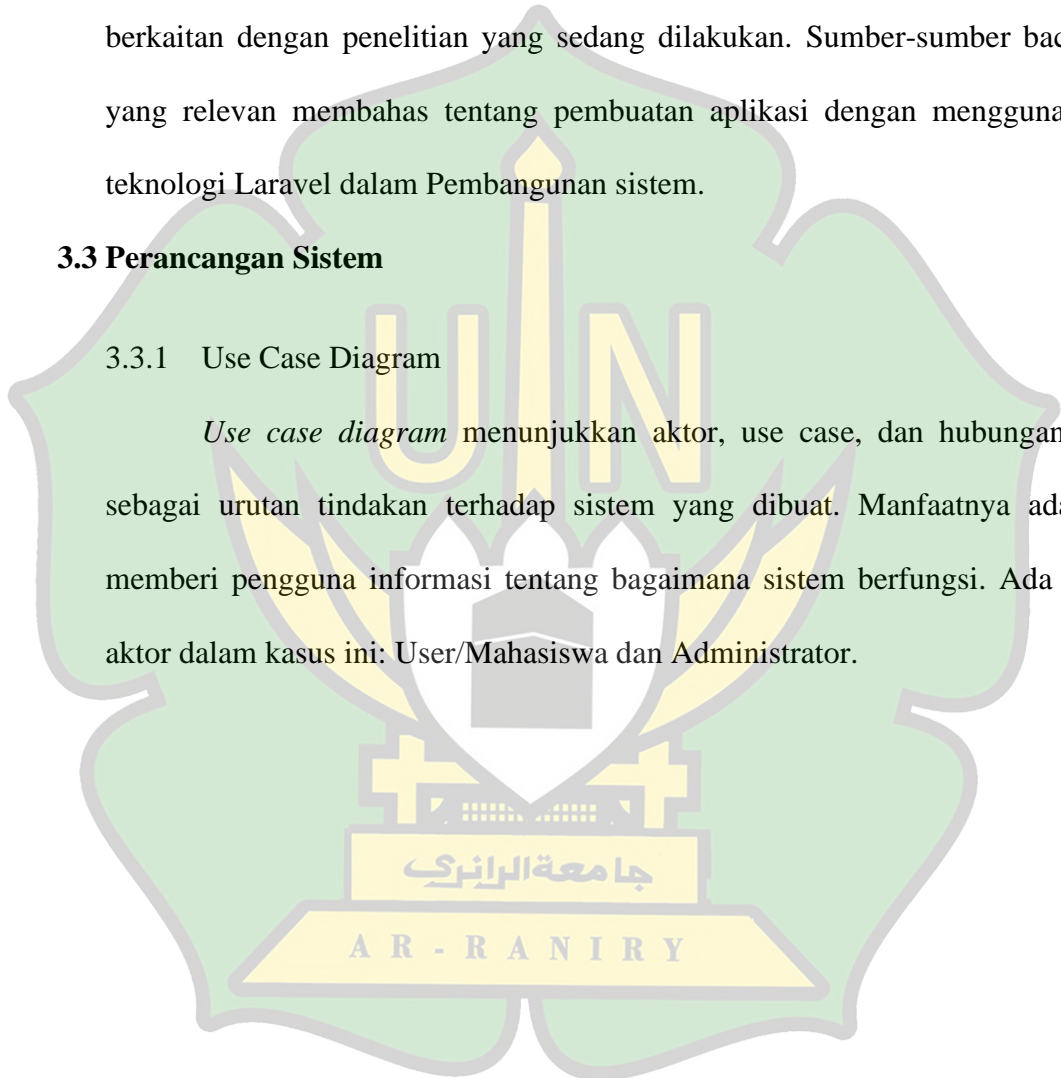
3.2.7 Metode Pengumpulan Data

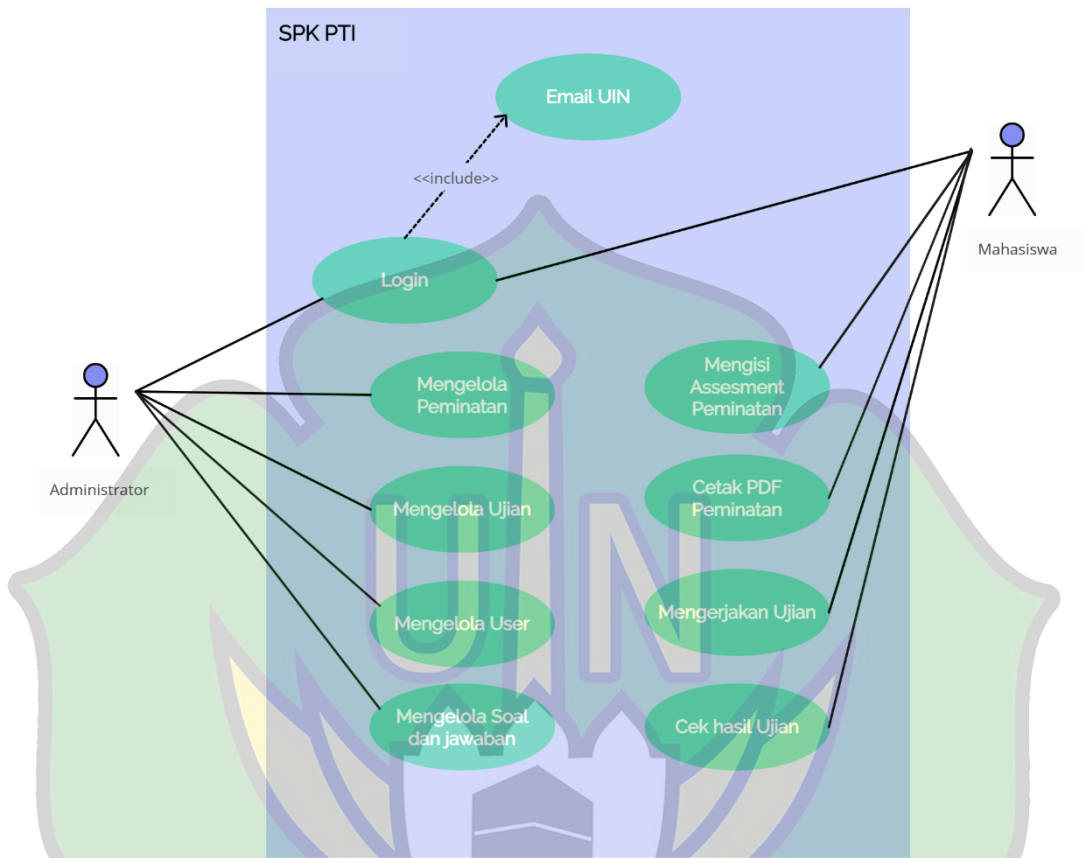
Peneliti menggunakan studi pustaka untuk mengumpulkan data. Ini diperoleh dari berbagai sumber, seperti buku, internet, jurnal, dan e-book yang berkaitan dengan penelitian yang sedang dilakukan. Sumber-sumber bacaan yang relevan membahas tentang pembuatan aplikasi dengan menggunakan teknologi Laravel dalam Pembangunan sistem.

3.3 Perancangan Sistem

3.3.1 Use Case Diagram

Use case diagram menunjukkan aktor, use case, dan hubungannya sebagai urutan tindakan terhadap sistem yang dibuat. Manfaatnya adalah memberi pengguna informasi tentang bagaimana sistem berfungsi. Ada dua aktor dalam kasus ini: User/Mahasiswa dan Administrator.





Gambar 3. 2 Use Case Diagram SPK PTI

Berikut ini adalah skenario penggunaan di mana Pengguna atau Mahasiswa menggunakan SPK PTI:

Tabel 3. 1 Use Case User dan Admin

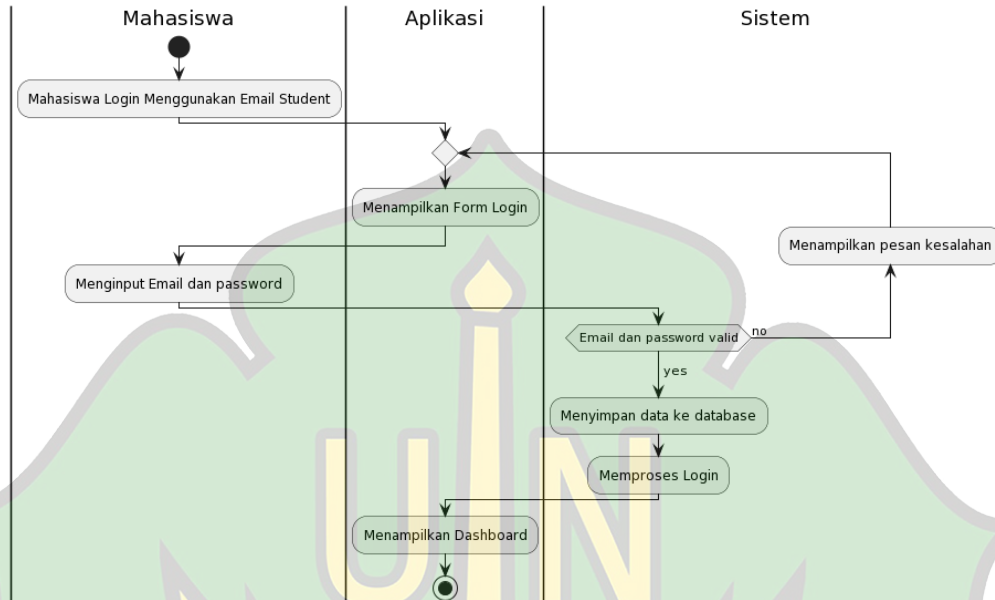
No.	Use Case	Aktor	Keterangan
1	Login/Logout	User/Admin	User/Admin melakukan login dengan memasukkan email dan password akun student. Dan juga bisa logout dari akun mereka.

2	Mengelola Peminatan	Admin	Admin menambahkan, mengedit dan menghapus data yang dibutuhkan dalam assessment peminatan
3	Mengisi Assement Peminatan	User	User menginput nilai dengan kriteria yang tertera di halaman assessment peminatan
4	Mengerjakan Ujian	User	User mengerjakan ujian yang sudah dibuat oleh admin
5	Mengelola User	Admin	Admin menambahkan, mengedit dan menghapus data user
6	Mengelola Ujian	Admin	Admin menambahkan, mengedit dan menghapus ujian
7	Mengelola Soal dan Jawaban	Admin	Admin menambahkan, mengedit dan menghapus soal dan jawaban untuk ujian
8	Cetak PDF Peminatan	User	User mencetak pdf setelah melakukan assessment peminatan
9	Cek Hasil Ujian	User	User dapat melihat hasil ujian yang telah diikuti

3.3.2 Activity Diagram

Activity Diagram dapat digunakan untuk menjelaskan proses dan alur kerja operasional secara langkah demi langkah dari komponen suatu sistem. Activity Diagram adalah representasi grafis dari seluruh tahapan alur kerja yang mengandung aktivitas, pilihan tindakan, perulangan, dan hasil dari aktivitas tersebut.

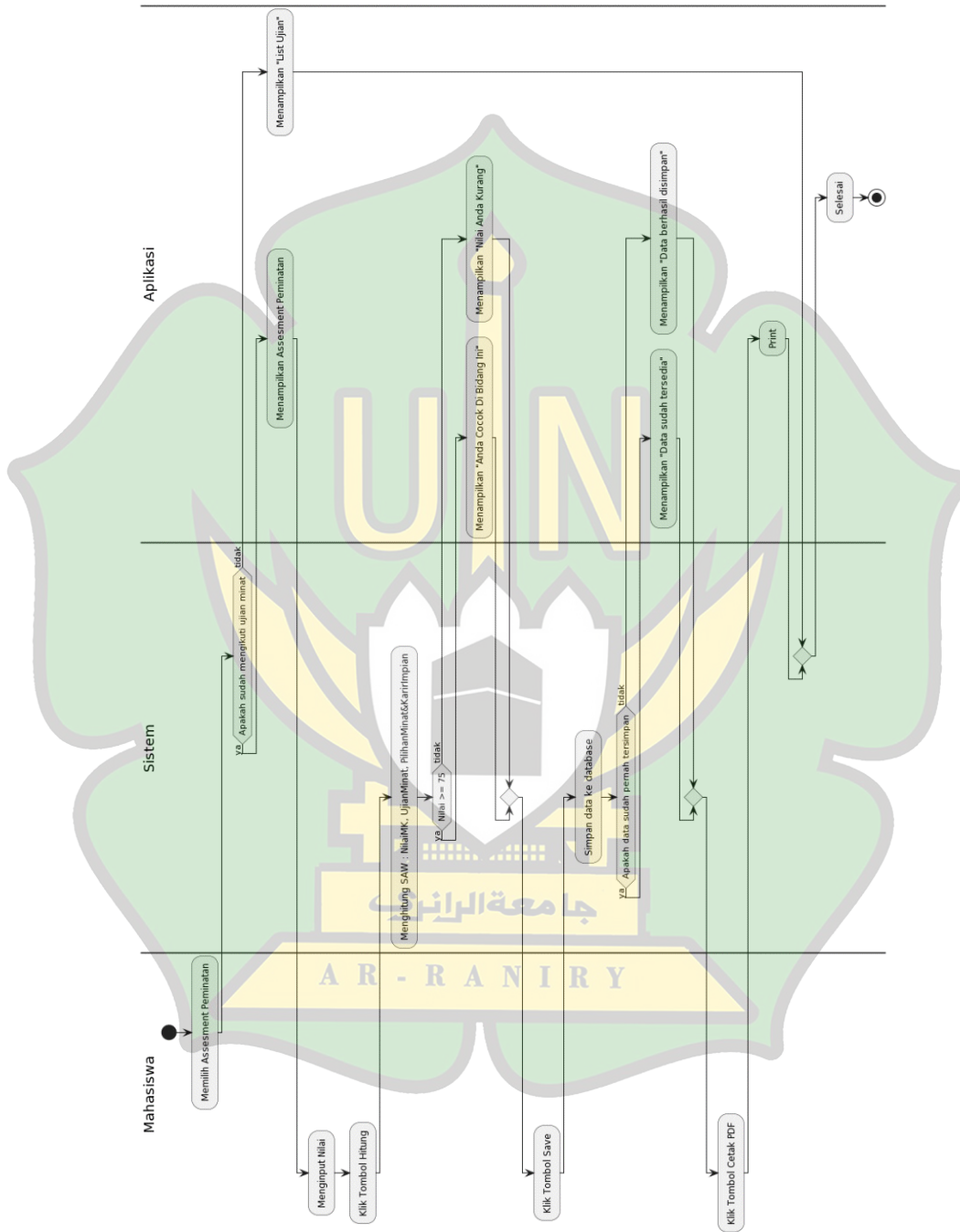
a. Diagram User Login



Gambar 3. 3 Activity Diagram *user login*

User mengisi *email* dan *password* dari akun *student* UIN kemudian di validasi oleh sistem. Jika data benar maka aplikasi akan menampilkan halaman Dashboard dan data user akan disimpan ke dalam database, namun jika salah maka akan menampilkan pesan “harap menggunakan email student” dan juga user akan diminta untuk mengisinya lagi.

b. Diagram User Assement Peminatan

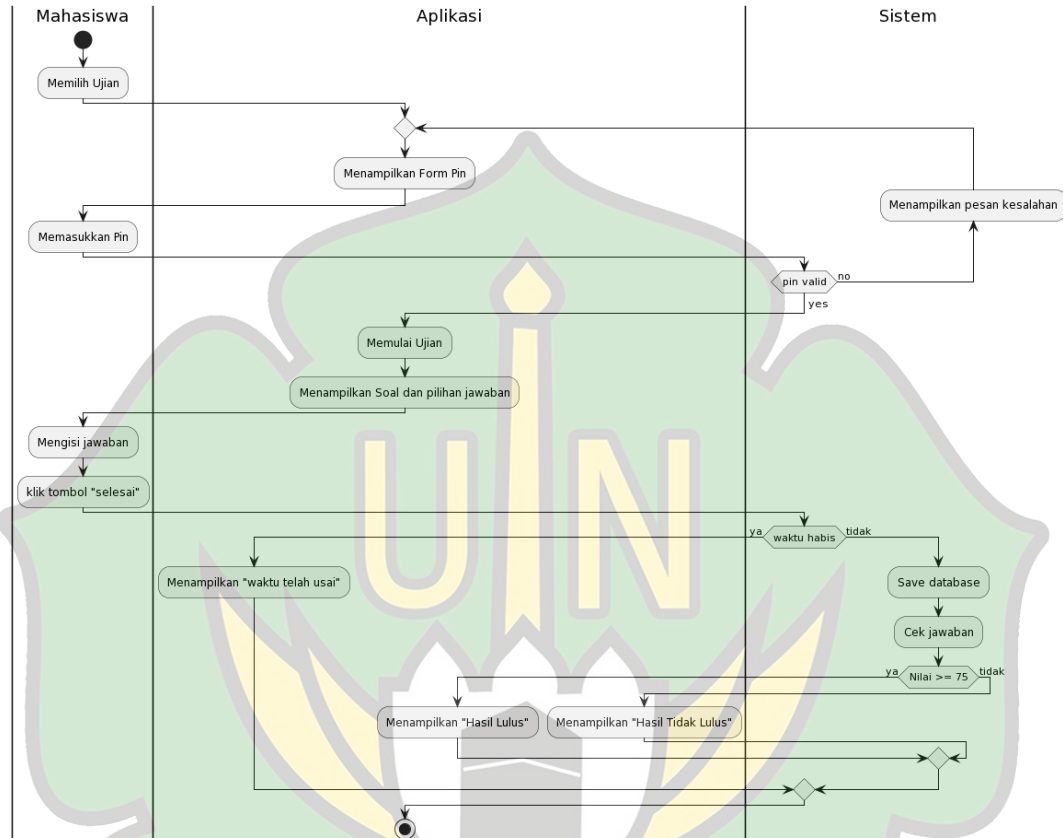


Gambar 3. 4 Activity Diagram User Assesment

User memilih halaman "assessment" sistem akan mengecek apakah user/mahasiswa telah mengikuti ujian minat ? jika belum maka akan dialihkan ke halaman "List Ujian" sebagai syarat bisa menginput assessment peminatan. Jika user telah mengikuti ujian maka user tinggal memasukkan nilai-nilai yang tercantum di halaman tersebut. User memilih peminatan dan karir di masa depan yang diinginkan. Selanjutnya, user mengklik tombol "hitung" untuk memproses nilai-nilai yang telah dimasukkan. Selama proses ini, sistem memeriksa setiap bobot yang telah dihitung menggunakan metode SAW. Hasil dari proses ini akan menampilkan pesan kepada user apakah nilainya mencukupi atau tidak untuk memilih peminatan tersebut. Setelah memperoleh nilai yang memadai untuk peminatan yang dipilih oleh user, user diminta untuk mengklik "simpan" agar data nilai dan peminatan tersimpan di database sistem. User juga dapat mengklik "cetak PDF" untuk menyimpan informasi dari halaman penilaian peminatan.



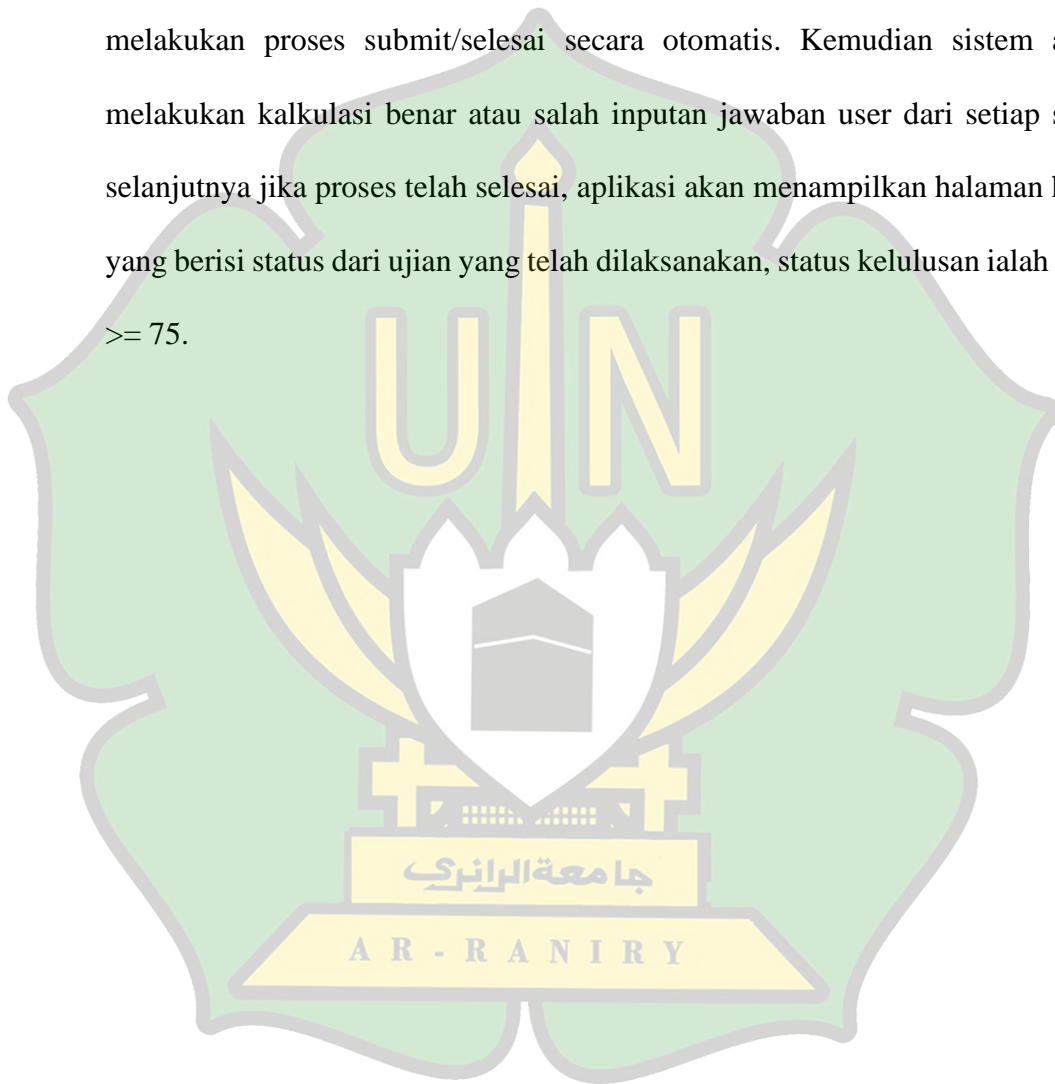
c. Diagram User Melaksanakan Ujian



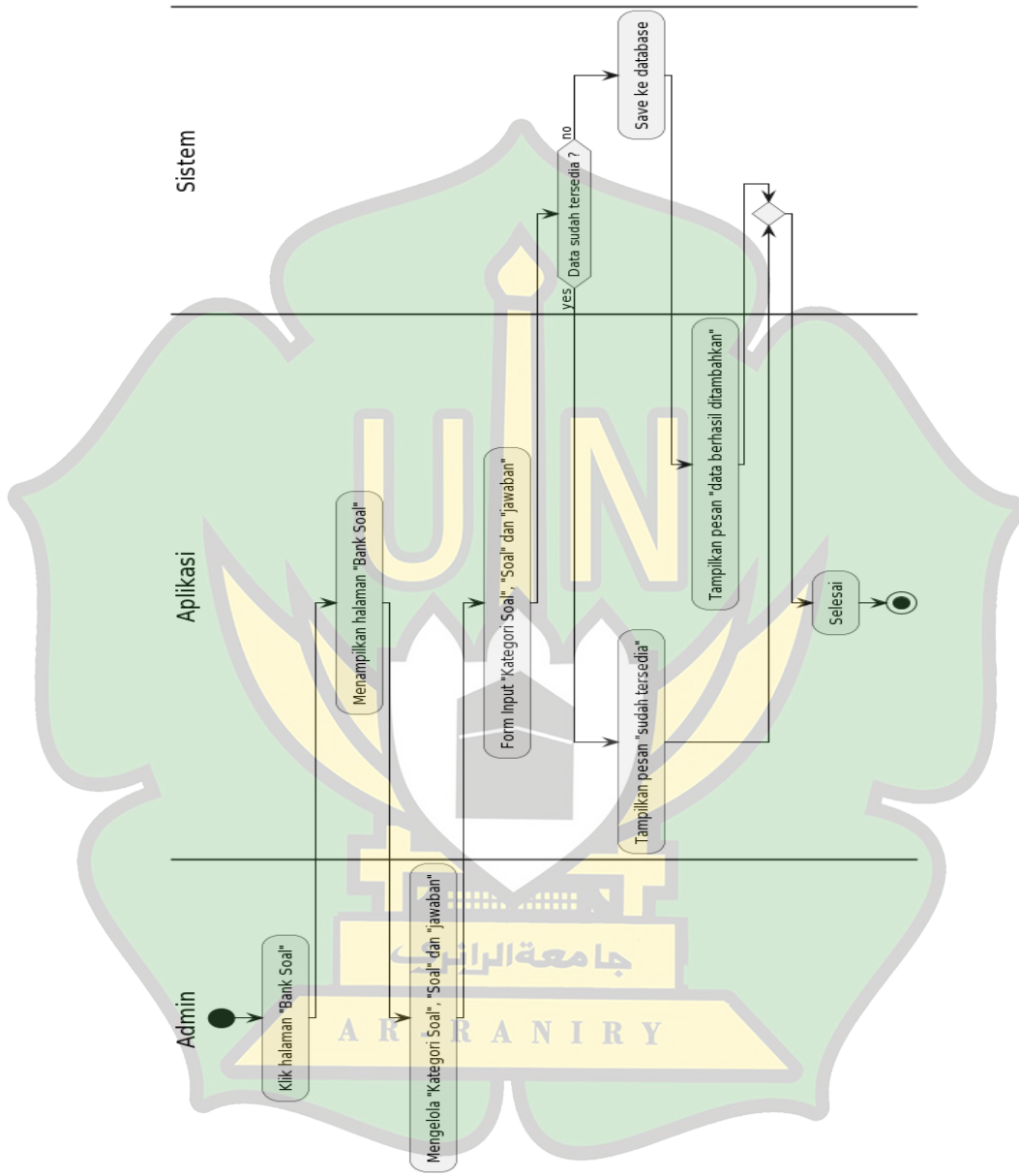
Gambar 3. 5 Activity Diagram User Ujian

User memilih halaman "List Ujian" dan memilih ujian yang akan diikuti, kemudian user diminta untuk memasukkan pin ujian, selanjutnya sistem akan mengecek pin beserta status user, apakah pin yang dimasukkan user benar? dan apakah user sudah lulus dari ujian ini? jika status user lulus dari ujian ini maka sistem akan menampilkan hasil status nilai ujian sebelumnya, jika status user tidak lulus dari ujian yang dipilih dan juga pin yang dimasukkan benar maka akan ditampilkan halaman ujian. User diharuskan melakukan

penginputan jawaban dari soal – soal yang disediakan, selanjutnya jika user telah mengisi semua jawaban user diharuskan mengklik tombol “selesai” apabila user kehabisan waktu dan lupa mengklik tombol “selesai” sistem akan melakukan proses submit/selesai secara otomatis. Kemudian sistem akan melakukan kalkulasi benar atau salah inputan jawaban user dari setiap soal, selanjutnya jika proses telah selesai, aplikasi akan menampilkan halaman hasil yang berisi status dari ujian yang telah dilaksanakan, status kelulusan ialah nilai ≥ 75 .



d. Diagram Admin Mengelola Soal

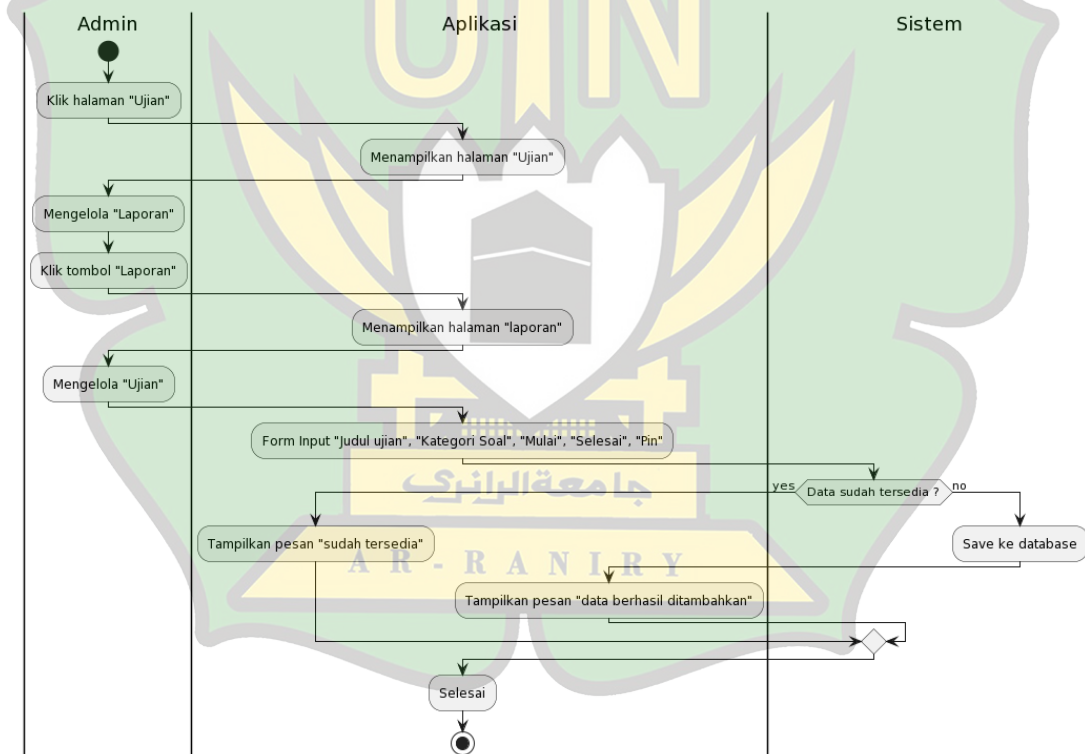


Gambar 3. 6 Activity Diagram Admin Mengelola Soal

Admin memilih halaman “Bank Soal” untuk melakukan pengelolaan soal yang akan diujikan, setiap soal memiliki kategori dan pilihan jawaban.

Pertama admin diharuskan untuk membuat kategori soal, kemudian admin melakukan input penambahan soal beserta pilihan jawaban yang benar dan salah, selesai menginput admin mengklik tombol simpan yang akan diproses dan dikirimkan ke database. Jika data soal telah tersedia maka sistem akan menampilkan pesan “data telah tersedia” namun jika tidak maka data soal beserta jawaban dari admin akan dikirimkan ke database oleh sistem dan menampilkan pesan “data berhasil ditambahkan”.

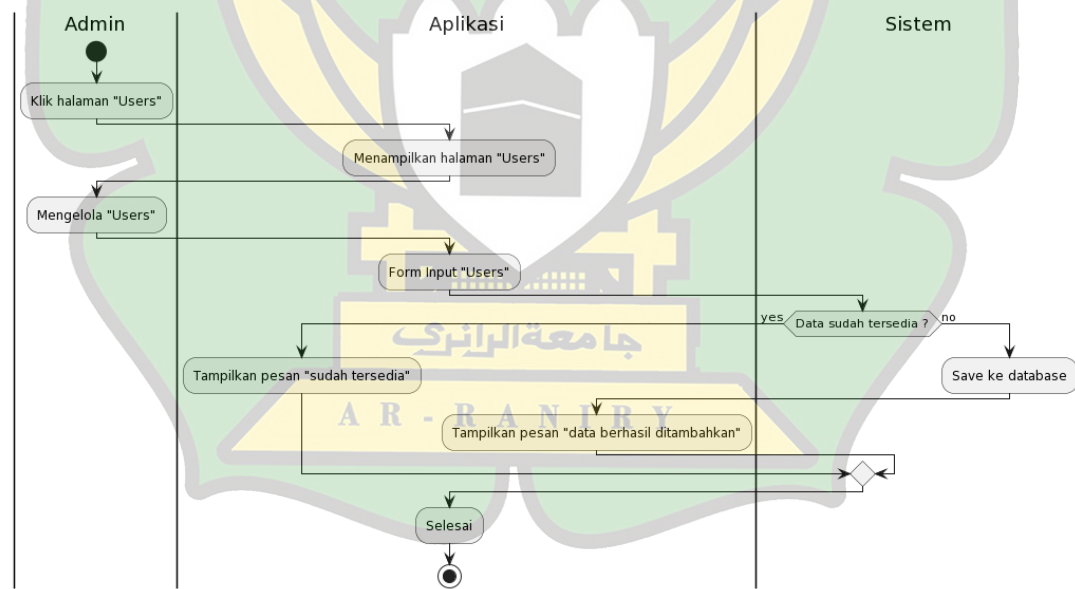
e. Diagram Admin Mengelola Ujian



Gambar 3. 7 Activity Diagram Admin Mengelola Ujian

Admin memilih halaman “Ujian” untuk melakukan pengelolaan Ujian dimana kategori soal apa saja yang akan diujikan, Admin bisa menambahkan judul ujian, kategori soal, tanggal mulai, tanggal selesai dan pin, selesai menginput admin mengklik tombol simpan yang akan diproses dan dikirimkan ke database. Jika data Ujian telah tersedia maka sistem akan menampilkan pesan “data telah tersedia” namun jika tidak maka data ujian hasil inputan dari admin akan dikirimkan ke database oleh sistem dan menampilkan pesan “data berhasil ditambahkan”.

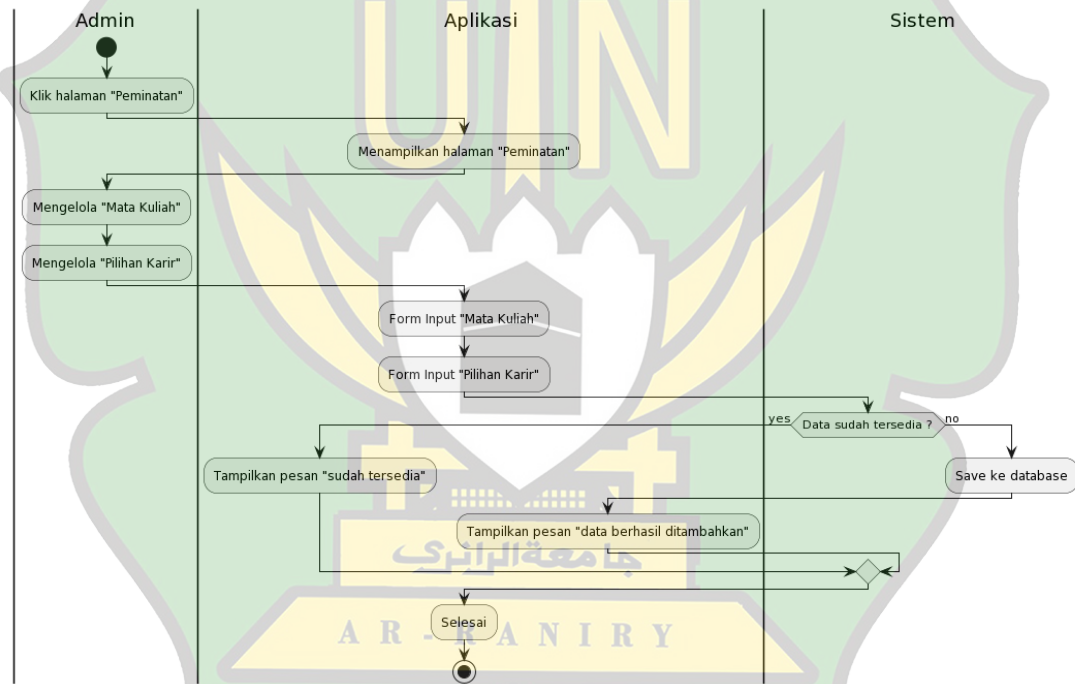
f. Diagram Admin Mengelola Users



Gambar 3. 8 Activity Diagram Admin Mengelola Users

Admin memilih halaman “Users” untuk melakukan pengelolaan user, dimana data user telah tersimpan otomatis ke dalam database jika user telah login ke dalam aplikasi. Admin bisa mengedit dan menghapus data user di halaman Users. Sistem juga melakukan pengecekan setiap editan dari Admin yang mana jika data telah tersedia di database maka tidak akan tersimpan,

g. Diagram Admin Mengelola Peminatan



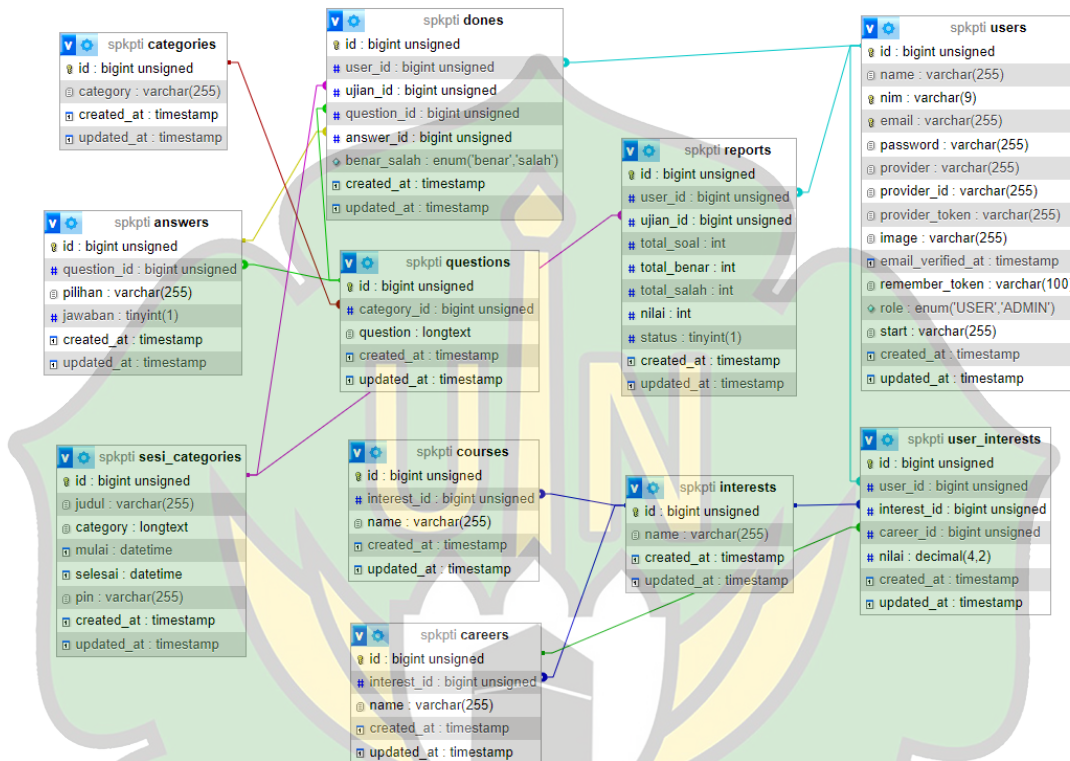
Gambar 3. 9 Activity Diagram Admin Mengelola Peminatan

Admin memilih halaman “Peminatan” untuk melakukan pengelolaan peminatan, Admin bisa menambahkan mata kuliah dan pilihan karir dimasa depan yang akan digunakan sebagai kriteria assessment peminatan, selesai

menginput admin mengklik tombol simpan yang akan diproses dan dikirimkan ke database. Jika data telah tersedia maka sistem akan menampilkan pesan “data telah tersedia” namun jika tidak maka data Mata kuliah atau pilihan karir hasil inputan dari admin akan dikirimkan ke database oleh sistem dan menampilkan pesan “data berhasil ditambahkan”.



3.3.3 Data Model



Gambar 3. 10 Data Model

Berikut adalah penjelasan mengenai data model beserta relasi untuk entitas *users*, *categories*, *questions*, *answers*, *sesi_categories*, *dones*, *reports*, *interests*, *courses*, *careers*, *user_interests*:

a. *Data Model Users*:

- *Name*: Nama lengkap dari mahasiswa
- *Nim*: Nim lengkap dari mahasiswa
- *Email*: email student lengkap dari mahasiswa

- *Image*: gambar dari *email student* mahasiswa
 - *Provider*: *provider* yang digunakan mahasiswa untuk *login*
 - *Role*: level akses user
 - *Start*: log waktu ujian user
- b. *Data Model Categories*:
- *Id*: id kategori yang akan direlasikan ke model *questions*
 - *Category*: nama untuk kategori setiap soal
- c. *Data Model Questions*:
- *Id*: id soal yang akan direlasikan ke *model answers*
 - *Category_id*: id *foreign key* dari model *categories*
 - *Question*: isi pertanyaan yang terhubung dengan *category_id*
- d. *Data Model Answers*:
- *Id*: id jawaban yang akan direlasikan ke *model done*s
 - *Question_id*: id *foreign key* dari *model questions*
 - *Pilihan*: jawaban pilihan dari pertanyaan yang terhubung dengan *question_id*
 - *Jawaban*: jawaban pilihan yang bernilai benar dan salah (*Boolean*)
- e. *Data Model sesi_categories*:
- *Id*: id ujian yang akan direlasikan ke *model done*s dan *reports*
 - *Judul*: judul dari ujian yang akan dibuat
 - *Category*: list dari kategori soal yang akan diujikan

- Mulai: tanggal dan waktu dilaksanakan ujian
- Selesai: tanggal dan waktu selesai ujian
- Pin: pin yang digunakan sebagai syarat masuk ujian

f. *Data Model Dones:*

- *User_id*: id foreign key dari model users
- *Ujian_id*: id foreign key dari model sesi_categories
- *Question_id*: id foreign key dari model questions
- *Answer_id*: id foreign key dari model answers
- *Benar_salah*: berisi benar atau salah dari setiap jawaban yang dilakukan user

g. *Data Model Reports:*

- *User_id*: id foreign key dari model users
- *Ujian_id*: id foreign key dari model sesi_categories
- *Total_soal*: berisi total soal yang telah dikerjakan oleh user
- *Total_benar*: berisi total benar dari jawaban yang telah dikerjakan user
- *Total_salah*: berisi total salah dari jawaban yang telah dikerjakan user
- *Nilai*: berisi hasil kalkulasi benar/salah dari jawaban yang telah dikerjakan user
- *Status*: *Boolean* yang merepresentasikan lulus atau tidaknya nilai user

h. *Data Model Interests:*

- *Id:* id peminatan yang akan direlasikan ke *model courses*, *careers* dan *user_interests*
- *Name:* nama dari peminatan

i. *Data Model Courses:*

- *Interest_id:* id foreign key dari *model interests*
- *Name:* nama mata kuliah yang akan menjadi syarat kriteria peminatan

j. *Data Model Careers:*

- *Interest_id:* id foreign key dari *model interests*
- *Name:* nama pilihan karir masa depan yang akan menjadi syarat kriteria peminatan

k. *Data Model User_interests:*

- *User_id:* id foreign key dari *model users*
- *Interest_id:* id foreign key dari *model interests*
- *Career_id:* id foreign key dari *model careers*
- *Nilai:* hasil kalkulasi dengan menggunakan metode SAW di halaman *assessment*

Jenis – jenis relasi yang terjadi pada setiap model adalah sebagai berikut :

- Model Users:

One To Many -> Sesi_Categories, User_Interests, Reports, Dones

- Model Categories:

One To Many -> Questions

Many To One -> Sesi_Categories

- Model Questions:

One To Many -> Answers

Many To One -> Categories

- Model Answers:

Many To One -> Questions

- Model Dones:

Many To One -> Users, Questions, Answers

- Model Reports:

Many To One -> Users

- Model Interests:

One To Many -> Courses, Careers, User_Interests

- Model Courses dan Careers:

Many To One -> Interests

- Model User_Interests:

Many To One -> Users, Interests, Careers

3.4 Implementasi *Simple Additive Weighting* (SAW)

Tahap awal implementasi *Simple Additive Weighting* (SAW) adalah dengan menentukan kriteria yang dibutuhkan. Kriteria yang dibutuhkan merupakan nilai dari ujian minat yang diikuti oleh mahasiswa, kriteria selanjutnya ialah rata – rata nilai mata kuliah prasyarat untuk pengambilan mata kuliah peminatan yang telah ditempuh mahasiswa program studi Pendidikan teknologi informasi pada semester 1 – 4. Terakhir ialah kriteria dari pilihan minat dan karir masa depan mahasiswa.

Bobot Kriteria:

Tabel 3. 2 Tabel kriteria dan bobot

Kode Kriteria	Kriteria	Bobot
C1	Ujian Minat	40%
C2	Nilai rata – rata mata kuliah bidang minat	50%
C3	Pilihan minat dan pilihan karir di masa depan	10%

Skor Alternatif pada Setiap Kriteria:

Tabel 3. 3 Tabel Skor Alternatif

#	Ujian Minat	Rata – rata mata kuliah	Pilihan karir sesuai dengan minat
A1	80	90	100
A2	50	70	0

Langkah-langkah perhitungan:

1. Normalisasi Matriks Keputusan (R) :

$$R = \begin{vmatrix} 0.4 & 0.5 & 0.1 \\ 80 & 90 & 100 \\ 100 & 100 & 100 \\ 50 & 70 & 0 \\ 100 & 100 & 100 \end{vmatrix}$$

$$R = \begin{vmatrix} 0.4 & 0.5 & 0.1 \\ 0.8 & 0.9 & 0.1 \\ 0.5 & 0.7 & 0 \end{vmatrix}$$

2. Bobot Terbobot (W):

$$\text{Bobot Terbobot (W)} = \begin{vmatrix} 0.4 \\ 0.5 \\ 0.1 \end{vmatrix}$$

3. Mengalikan Matriks Normalisasi (R) dengan Bobot Terbobot (W):

$$R \times W = \begin{vmatrix} 0.4 \times 0.8 + 0.5 \times 0.9 + 0.1 \times 1 \\ 0.4 \times 0.5 + 0.5 \times 0.7 + 0.1 \times 0 \end{vmatrix}$$

$$R \times W = \begin{vmatrix} 0.732 \\ 0.561 \end{vmatrix}$$

Hasil Akhir

$$A1 = 0.732$$

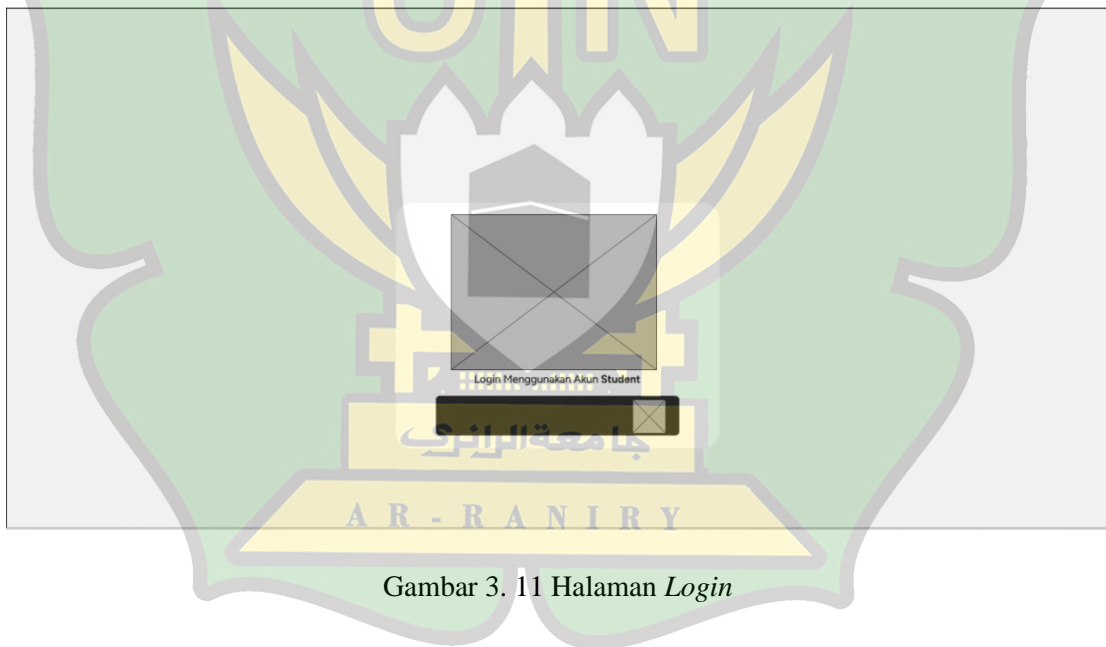
$$A2 = 0.561$$

Jadi, berdasarkan perhitungan SAW, alternatif A1 memiliki nilai tertinggi dan dapat dipilih sebagai alternatif terbaik.

3.5 Perancangan Aplikasi

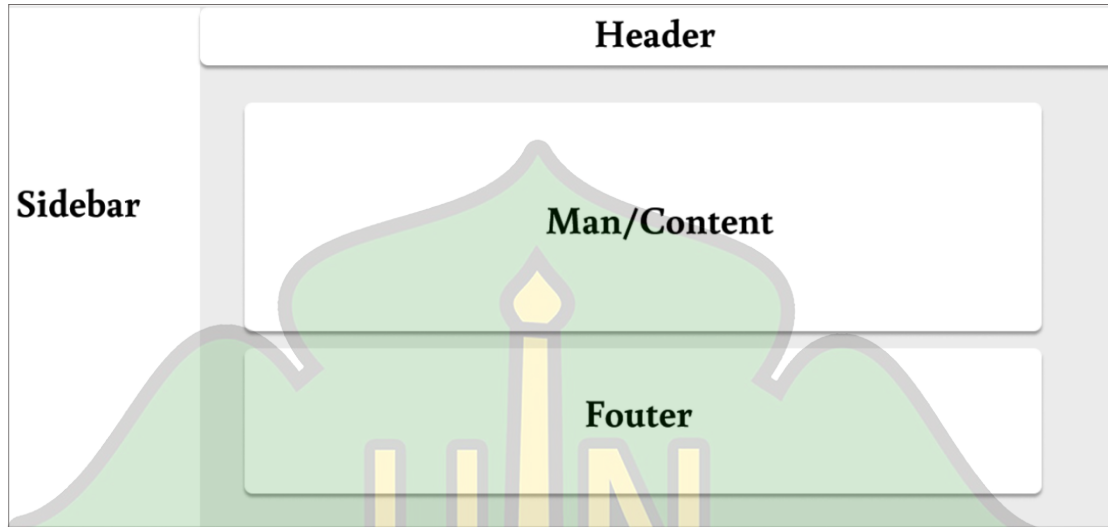
Perancangan aplikasi merupakan langkah yang digunakan dalam menciptakan tampilan antarmuka pengguna, atau user interface, dari sebuah aplikasi [22]. Adapun perancangan terlihat seperti gambar di bawah ini.

a. Halaman Login



Gambar 3. 11 Halaman *Login*

b. Halaman *Frontend, Backend*



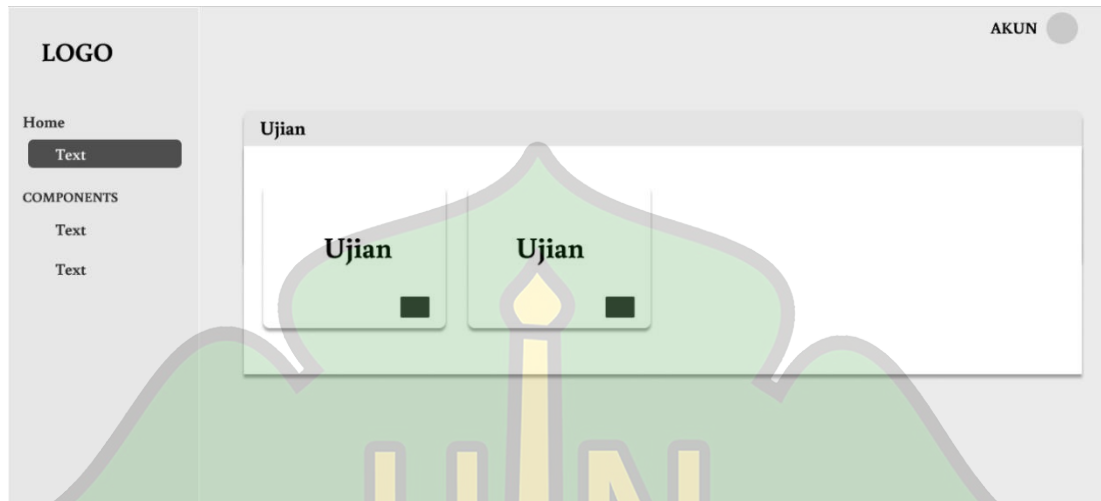
Gambar 3. 12 Halaman *Frontend, Backend*

c. Halaman *Home*



Gambar 3. 13 Halaman *Home*

d. Halaman List Ujian



Gambar 3. 14 Halaman *List Ujian*

e. Halaman Assessment peminatan mahasiswa



Gambar 3. 15 Halaman *Assessment*

f. Halaman melaksanakan ujian



Gambar 3. 16 Halaman *Ujian*

g. Halaman hasil melaksanakan ujian



Gambar 3. 17 Halaman *Hasil*

h. Halaman *Admin, List Ujian, Tambah*



Judul
Ujian Minat

Apa Saja yang Di ujikan
Select box

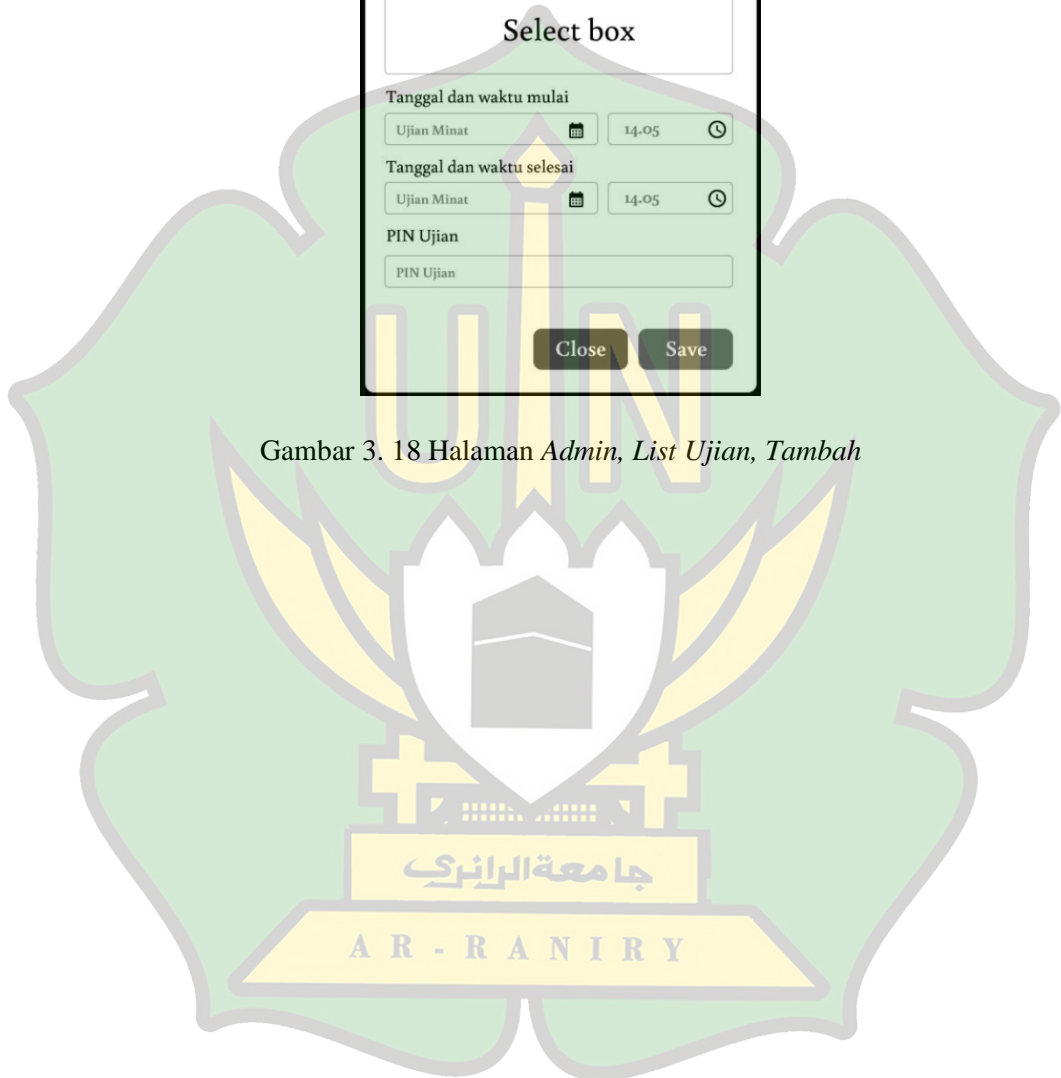
Tanggal dan waktu mulai
Ujian Minat 14-05

Tanggal dan waktu selesai
Ujian Minat 14-05

PIN Ujian
PIN Ujian

Close Save

Gambar 3. 18 Halaman *Admin, List Ujian, Tambah*



BAB IV HASIL DAN PEMBAHASAN

3.6 Hasil Implementasi dan Pembahasan

Sistem dikembangkan sesuai dengan rencana dan desain sebelumnya selama tahap pelaksanaan sistem. Proses ini dapat dibagi menjadi beberapa bagian berdasarkan fungsionalitas yang telah ditetapkan [23].

3.6.1 Tampilan Login dan Dashboard



Gambar 4. 1 Halaman *Login* dan *Dashboard*

Login adalah antarmuka yang digunakan ketika pengguna atau administrator hendak mengakses aplikasi SPK PTI. Jika pengguna atau



administrator berhasil memasukkan email dan password dengan benar, sistem akan mengarahkannya ke halaman *dashboard*. Sebaliknya, jika terdapat kesalahan dalam memasukkan email dan password, sistem akan menampilkan pesan kesalahan "Email tidak valid".

3.6.2 Tampilan Assesment dan CetakPDF

Assesment Peminatan Mahasiswa

Masukkan nilai tiap mata kuliah anda kedalam input yang tersedia, pilih minat dan karir, klik tombol Hitung kemudian pilih save

Rekayasa Perangkat Lunak	Multimedia	Teknik Komputer dan Jaringan
Basis Data I 90,00	Design 90,00	Pengantar Jaringan 90,00
Basis Data II 90,00	Pengantar Multimedia 90,00	
Rekayasa Web 90,00		

Rekayasa Perangkat Lunak 90,00

Multimedia 90,00

Teknik Komputer dan Jaringan 90,00

Nilai Ujian Minat: 100

Pilih bidang yang paling diminati

Pilihan Karir Dimasa Depan

Pilihan Peminatan

Pilihan Karir

Cetak PDF Hitung Save

Gambar 4. 2 Halaman *Assesment*

Input dari aktivitas pemilihan peminatan siswa ditampilkan melalui metode *Simple Additive Weighting* (SAW). Nilai mata kuliah, nilai ujian minat, pilihan minat, dan pilihan karier di masa depan adalah kriteria yang digunakan

untuk menilai. Bobot yang diterapkan untuk masing-masing kriteria adalah sebagai berikut:

- Nilai Rata – Rata Mata Kuliah Bidang Peminatan: 0,50
- Nilai Ujian Minat: 0,40
- Pilihan Minat dan Karir: 0,10

Jika hasil total penjumlahan skor kriteria memenuhi atau melebihi nilai 70 dan tombol "Hitung" ditekan, notifikasi akan muncul menyatakan "Nilai Anda (total) memenuhi untuk minat ini!" Tombol "Simpan" akan berhasil ditekan karena mengikuti syarat total di atas. Proses yang dilakukan oleh tombol "Simpan" adalah menyimpan hasil inputan dari pengguna ke dalam database.

Kemudian apabila tombol "CetakPDF" ditekan, pengguna akan diarahkan ke halaman "cetakPDF" yang membawa atribut atau nilai yang berasal dari data yang tersimpan dalam database.

Nama	Nim	Fakultas	Prodi
TEUKU MUKSALMINA	190212026	Tarbiyah dan Keguruan	Pendidikan Teknologi Informasi

Rekayasa Perangkat Lunak		Multimedia		Teknik Komputer dan Jaringan	
Matakuliah	Nilai	Matakuliah	Nilai	Matakuliah	Nilai
Basis Data I	90.00	Design	90.00	Pengantar Jaringan	90.00
Basis Data II	90.00	Pengantar Multimedia	90.00		
Rekayasa Web	90.00				
Rata-rata	90.00	Rata-rata	90.00	Rata-rata	90.00

Nilai Ujian	Peminatan	Pilihan Karir Masa Depan	Total Nilai
100	Rekayasa Perangkat Lunak	Web Development	95.00

Gambar 4. 3 Halaman *CetakPDF*

3.6.3 Tampilan List Ujian dan Pin Ujian

List Pelaksanaan Ujian

Ujian Minat	Ujian Komprehensif
<input checked="" type="checkbox"/> Mulai : 2023-11-05 14:05:00	<input checked="" type="checkbox"/> Mulai : 2023-11-05 14:13:00
<input checked="" type="checkbox"/> Selesai : -	<input checked="" type="checkbox"/> Selesai : -

Gambar 4. 4 Halaman *List Ujian*

Halaman List Ujian menampilkan daftar ujian yang diadakan oleh administrator, dengan informasi krusial mengenai waktu mulai dan selesai ujian, serta pin ujian yang diperlukan sebagai kunci untuk melaksanakan ujian.



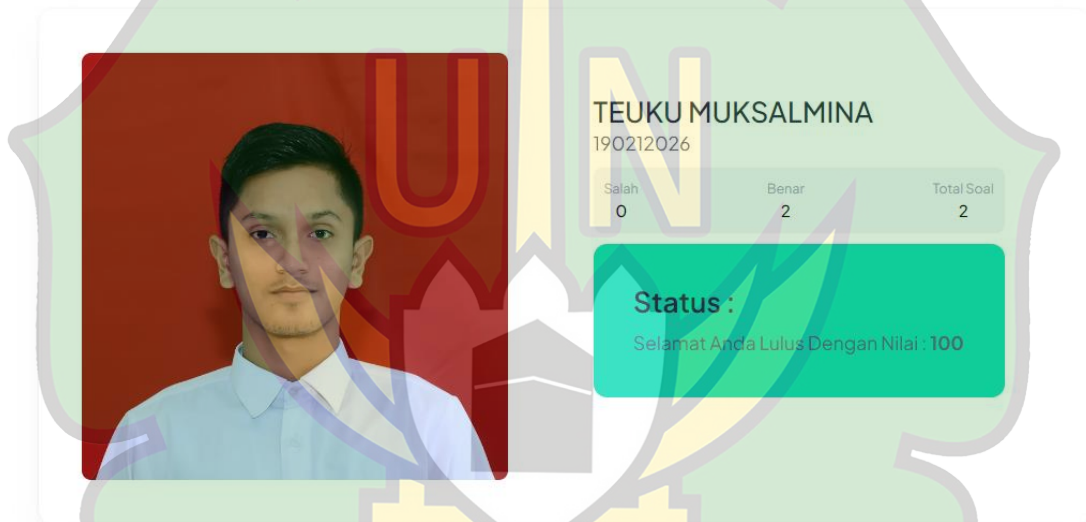
Gambar 4. 5 Halaman Memasukkan *Pin Ujian*

3.6.4 Tampilan Ujian dan Hasil Ujian

Gambar 4. 6 Halaman *Ujian*

Ujian adalah antarmuka pengguna yang digunakan ketika pengguna mengikuti ujian yang diselenggarakan oleh administrator melalui halaman ini. Pengguna dapat memilih jawaban yang benar dari pertanyaan yang diberikan.

Jika pengguna merasa ragu, mereka dapat menekan tombol "Lewati". Setelah mengerjakan semua atau sebagian pertanyaan, pengguna dapat menekan tombol "Selesai" untuk memulai proses identifikasi jumlah jawaban yang benar dan yang salah, yang nantinya akan ditampilkan di halaman hasil ujian. Jika nilai yang diperoleh oleh pengguna setara atau lebih besar dari 70, maka status pengguna dianggap "Lulus ujian".



Gambar 4.7 Halaman *Hasil Ujian*

3.7 Pengujian sistem

Uji sistem adalah tahap implementasi aplikasi untuk mengevaluasi kesesuaiannya dengan spesifikasi yang direncanakan dan kemampuan beroperasi dalam lingkungan yang diinginkan. Identifikasi bug, ketidaksempurnaan, dan kesalahan yang dapat menyebabkan kegagalan eksekusi perangkat lunak biasanya terkait dengan proses pengujian aplikasi.

Setiap komponen aplikasi diuji secara menyeluruh untuk memastikan bahwa semuanya berjalan dengan sempurna. Pengujian *Black Box* digunakan untuk menguji aplikasi berdasarkan spesifikasi fungsional tanpa mengakses desain atau kode program. Tujuan pengujian ini adalah untuk memastikan apakah fungsi, input, dan output aplikasi memenuhi spesifikasi yang telah ditetapkan.

3.7.1 Hasil Pengujian sistem

a) Pengujian halaman login

Tabel 4. 1 Pengujian Halaman Login

Skenario Pengujian	Hasil yang Diharapkan	Hasil Pengujian	Kesimpulan
Mengisi email dan password selain email student	Muncul alert login gagal	Muncul alert login gagal	Valid
Mengisi email dan password student	Login berhasil dan masuk ke dashboard	Login berhasil dan masuk ke dashboard	Valid

b) Pengujian halaman assesment

Tabel 4. 2 Pengujian Halaman Assesment

Skenario Pengujian	Hasil yang Diharapkan	Hasil Pengujian	Kesimpulan
Tidak mengikuti ujian minat dan langsung mengakses halaman assesment	Halaman di alihkan ke halaman List Ujian dan menampilkan pin dari ujian minat	Halaman di alihkan ke halaman List Ujian dan menampilkan pin dari ujian minat	Valid
Mengisi nilai mata kuliah dibawah rata – rata dan klik hitung	Muncul alert nilai anda tidak cukup	Muncul alert nilai anda tidak cukup	Valid
Memilih pilihan karir dimasa depan yang tidak sesuai	Muncul alert nilai anda tidak cukup	Muncul alert nilai anda tidak cukup	Valid

dengan pilihan peminatan dan klik hitung			
Mengisi nilai mata kuliah diatas rata – rata dan klik hitung	Muncul alert selamat nilai anda mencukupi	Muncul alert selamat nilai anda mencukupi	Valid
Mengisi nilai mata kuliah kategori rekayasa perangkat lunak lebih tinggi daripada nilai mata kategori yang lain	Di bawah logo rekayasa perangkat lunak akan menampilkan hasil rata – rata nilai yang lebih tinggi dari kategori mata kuliah yang lain	Di bawah logo rekayasa perangkat lunak akan menampilkan hasil rata – rata nilai yang lebih tinggi dari kategori mata kuliah yang lain	Valid
Menekan tombol save setelah menekan tombol hitung	Data sukses disimpan ke database dan menampilkan data berhasil disimpan	Data sukses disimpan ke database dan menampilkan data berhasil disimpan	Valid
Menekan tombol save ke dua kali agar terjadi duplikat data	Data gagal disimpan ke database karena terjadi duplikat data dan menampilkan data telah tersedia	Data gagal disimpan ke database karena terjadi duplikat data dan menampilkan data telah tersedia	Valid
Menekan tombol Cetak PDF	Menampilkan form untuk print di browser	Menampilkan form untuk print di browser	Valid

c) Pengujian halaman List Ujian

Tabel 4. 3 Pengujian Halaman List Ujian

Skenario Pengujian	Hasil yang Diharapkan	Hasil Pengujian	Kesimpulan
Menambahkan ujian pada admin dengan tanggal selesai yang sudah terlewati	Pada halaman List Ujian tidak menampilkan tombol masuk ujian	Pada halaman List Ujian tidak menampilkan tombol masuk ujian	Valid
Menambahkan ujian pada admin dengan tanggal dan waktu yang masih bisa di akses	Pada halaman List Ujian tidak menampilkan tombol masuk ujian	Pada halaman List Ujian tidak menampilkan tombol masuk ujian	Valid

Memasukkan pin ujian yang tidak sesuai pada ujian yang dipilih	Menampilkan pesan Pin yang anda masukkan salah	Menampilkan pesan Pin yang anda masukkan salah	Valid
Memasukkan pin ujian yang sesuai pada ujian yang dipilih	Sistem mengalihkan halaman ke halaman ujian	Aplikasi mengalihkan halaman ke halaman ujian	Valid

d) Pengujian halaman Ujian

Tabel 4. 4 Pengujian halaman ujian

Skenario Pengujian	Hasil yang Diharapkan	Hasil Pengujian	Kesimpulan
Menekan tombol selesai dengan tidak mengisi semua jawaban dari soal	Menampilkan halaman hasil dengan status tidak lulus	Menampilkan halaman hasil dengan status tidak lulus	Valid
Menunggu waktu habis dan tidak menekan tombol selesai	Sistem mengalihkan ke halaman hasil	Aplikasi langsung dialihkan ke halaman hasil	Valid
Mereload halaman ujian	Muncul alert sebagai trigger ya/tidak reload	Muncul alert sebagai trigger ya/tidak reload	Valid
Tidak menjawab soal dan menutup tab sampai waktu ujian habis	Sistem mengalihkan user ke halaman pin dan menampilkan pesan waktu telah usai	Aplikasi mengalihkan user ke halaman pin dan menampilkan pesan waktu telah usai	Valid
Mengikuti ujian walaupun user telah mengikuti ujian yang dipilih	Sistem mengalihkan ke halaman hasil	Sistem mengalihkan ke halaman hasil	Valid
Menekan tombol simpan tanpa memilih pilihan jawaban	Menampilkan pesan mohon maaf pilih jawaban terlebih dahulu	Menampilkan pesan mohon maaf pilih jawaban terlebih dahulu	Valid

e) Pengujian halaman ujian (Admin)

Tabel 4. 5 Pengujian halaman ujian (Admin)

Skenario Pengujian	Hasil yang Diharapkan	Hasil Pengujian	Kesimpulan
Menambahkan ujian dengan mengosongkan tanggal dan waktu selesai	Menampilkan pesan data berhasil disimpan dan menampilkan tombol Akhiri ujian pada aksi ujian tersebut	Menampilkan pesan data berhasil disimpan dan menampilkan tombol Akhiri ujian pada aksi ujian tersebut	Valid
Menekan tombol Akhiri ujian pada list ujian	Menampilkan tanggal dan waktu selesai pada list ujian tersebut	Menampilkan tanggal dan waktu selesai pada list ujian tersebut	Valid

Hasil pengujian aplikasi dengan metode black box testing telah berhasil. Selama proses pengujian, berbagai skenario dan kasus uji telah dijalankan tanpa akses ke kode sumber atau struktur internal aplikasi. Pengujian ini berfokus pada menguji fungsionalitas dan fitur-fitur eksternal aplikasi, serta mengidentifikasi masalah potensial yang dapat muncul dari perspektif pengguna akhir. Dalam hasil pengujian ini, ditemukan bahwa seluruh fungsionalitas aplikasi berjalan dengan baik sesuai dengan spesifikasi yang telah ditetapkan. Tidak ada masalah kritis yang ditemukan yang dapat menghambat penggunaan aplikasi secara keseluruhan.

Beberapa masalah minor yang ditemukan telah diidentifikasi dan akan diperbaiki dalam versi berikutnya. Hasil pengujian juga menunjukkan bahwa aplikasi telah merespons dengan cepat dan stabil pada berbagai platform dan sistem operasi yang diuji. Kinerja aplikasi pada perangkat dengan spesifikasi rendah dan tinggi terbukti memuaskan, tanpa adanya masalah performa yang signifikan. Dalam pengujian keamanan, aplikasi berhasil melewati tes yang dilakukan untuk memastikan adanya kelemahan keamanan yang potensial.

Berbagai langkah keamanan telah diterapkan dengan baik untuk melindungi data pengguna dan mencegah serangan yang mempengaruhi integritas aplikasi. Kesimpulannya, hasil pengujian dengan metode black box testing telah berhasil memvalidasi bahwa aplikasi berfungsi sesuai dengan harapan dan memenuhi kebutuhan fungsional pengguna. Hasil ini memberikan keyakinan bahwa aplikasi siap untuk dirilis dan dapat digunakan secara luas oleh pengguna akhir. Meskipun demikian, langkah-langkah pengembangan berkelanjutan akan dilakukan untuk terus memperbaiki dan meningkatkan kualitas aplikasi guna memberikan pengalaman pengguna yang optimal.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan dari hasil penelitian tentang pengembangan Aplikasi Peminatan dan Ujian Komprehensif Mahasiswa Berbasis Web Menggunakan Framework Laravel yang telah dilakukan, maka dapat diperoleh kesimpulan sebagai berikut :

1. Penelitian ini menghasilkan aplikasi SPK PTI Peminatan dan Ujian Komprehensif berbasis Web dengan menggunakan metode *Rapid Application Development* (RAD) antara lain :
 - *Requirement Planning* (Rencana Kebutuhan)
 - *User Design* (Desain Pengguna)
 - *Construction* (Pengembangan)
 - *Cutover* (Implementasi dan Pengujian)
2. Penerapan Sistem Pendukung Keputusan (SPK) Menggunakan metode *Simple Additive Weighting* (SAW) dengan bobot kriteria :
 - Ujian Minat = 40%
 - Nilai rata – rata mata kuliah bidang minat = 50%
 - Pilihan minat dan pilihan karir di masa depan = 10%
3. Hasil pengujian fungsi halaman login, assessment, list ujian, ujian, dan ujian (admin) pada Aplikasi Peminatan dan Ujian Komprehensif Mahasiswa Berbasis Web Menggunakan Framework Laravel adalah valid.

5.2 Saran

Dari hasil pengembangan menggunakan Metode *Rapid Application Development* (RAD) pada pembuatan aplikasi SPK PTI Peminatan dan Ujian Komprehensif berbasis web dengan *Framework Laravel* dalam penelitian ini, disarankan agar pengembangan selanjutnya memperhatikan aspek dinamisitas pada proses assessment peminatan. Hal ini bertujuan untuk meningkatkan efisiensi dan efektivitas dalam penginputan data. Selain itu, aplikasi ini dapat ditingkatkan pada sisi frontend dengan mempertimbangkan penggunaan teknologi seperti *React* atau *Vue*. Integrasi teknologi ini akan memberikan kompleksitas dan efisiensi yang lebih baik pada tampilan depan (*frontend*) aplikasi.

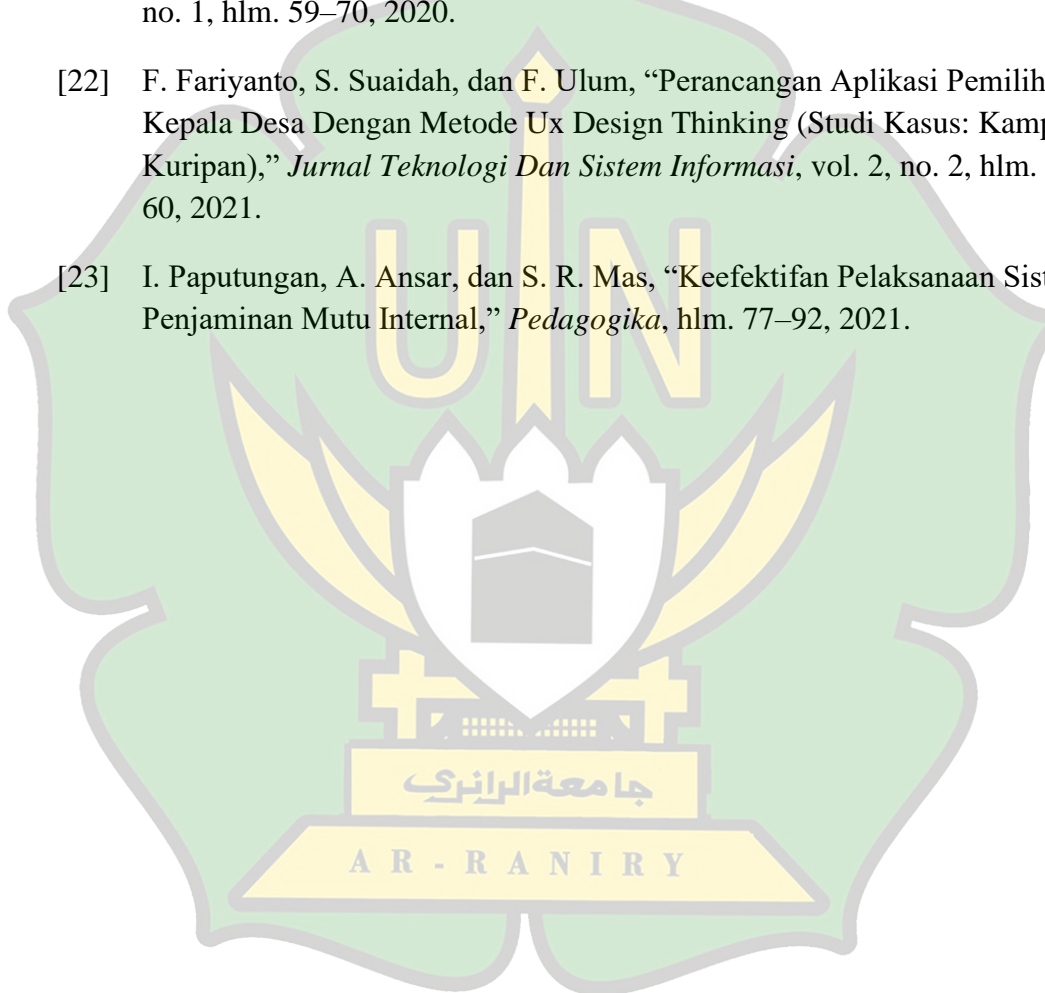
DAFTAR PUSTAKA

- [1] A. D. Hamdani, “Pendidikan di Era Digital Yang Mereduksi Nilai Budaya,” *Cermin: Jurnal Penelitian*, vol. 5, no. 1, hlm. 62–68, 2021.
- [2] T. Noor, “rumusan tujuan pendidikan nasional pasal 3 undang-undang sistem pendidikan nasional No 20 Tahun 2003,” *Wahana Karya Ilmiah Pendidikan*, vol. 2, no. 01, 2018.
- [3] V. Khuangnata, R. Alamsyah, dan V. Wijaya, “SPK Penentuan Pemberian Beasiswa Dengan Metode Saw,” *Jurnal Ilmiah Teknik Informatika METHOTIKA*, vol. 1, no. 2, hlm. 1–10, 2021.
- [4] A. R. S. Alam dkk., “Rancang Bangun Sistem Pendataan Jual Beli Tanah Menggunakan Metode Rapid Application Development,” *Jurnal Testing Dan Implementasi Sistem Informasi*, vol. 1, no. 1, hlm. 41–52, 2023.
- [5] 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.
- [6] M. D. Payana, M. B. Wibawa, dan A. Rahmi, “Sistem Ujian Online Tes Masuk Universitas Ubudiyah Indonesia Bagi Calon Mahasiswa Baru Berbasis Web Menggunakan CI (Code Igniter),” *Journal of Informatics and Computer Science*, vol. 6, no. 2, hlm. 91–103, 2020.
- [7] N. M. D. Febriyanti, A. A. K. O. Sudana, dan I. N. Piarsa, “Implementasi Black Box Testing pada Sistem Informasi Manajemen Dosen,” *Jurnal Ilmiah Teknologi Dan Komputer*, vol. 2, no. 3, hlm. 535–544, 2021.
- [8] H. Hamzah, S. G. Amalga, dan others, “RANCANG BANGUN APLIKASI PEMBELAJARAN ONLINE MENGGUNAKAN FRAMEWORK LARAVEL (STUDI KASUS: WEB PROGRAMMING UNPAS),” Fakultas Teknik Unpas, 2019.
- [9] G. B. Santoso, T. M. Sinaga, dan A. Zuhdi, “MVC Implementation In Laravel Framework For Development Web-Based E-Commerce Applications,” *Intelmatics*, vol. 1, no. 1, 2021.

- [10] M. T. Hidayatullah, M. Asbari, M. I. Ibrahim, dan A. H. H. Faidz, “Urgensi Aplikasi Teknologi dalam Pendidikan di Indonesia,” *Journal of Information Systems and Management (JISMA)*, vol. 2, no. 6, hlm. 70–73, 2023.
- [11] T. A. Kinaswara, “Rancang Bangun Aplikasi Inventaris Berbasis Website pada Kelurahan Bantengan,” dalam *Prosiding Seminar Nasional Teknologi Informasi dan Komunikasi (SENATIK)*, 2019, hlm. 71–75.
- [12] R. K. Atmaja, I. Komarudin, dan others, “Konsep MVC Pada Rancang Bangun Aplikasi Sistem Informasi Rekrutmen Karyawan Berbasis Web,” *IMTechno: Journal of Industrial Management and Technology*, vol. 2, no. 1, hlm. 30–36, 2021.
- [13] S. Irsyad dan A. S. Sitio, “Penerapan Konsep Mvc Pada Sistem Penjualan Online Dengan Sistem Keamanan Menggunakan Algoritma Rijndael,” *Informatika*, vol. 11, no. 2, hlm. 44–50, 2019.
- [14] R. Y. Endra, Y. Aprilinda, Y. Y. Dharmawan, dan W. Ramadhan, “Analisis Perbandingan Bahasa Pemrograman PHP Laravel dengan PHP Native pada Pengembangan Website,” *EXPERT: Jurnal Manajemen Sistem Informasi dan Teknologi*, vol. 11, no. 1, hlm. 48–55, 2021.
- [15] I. Y. Supardi, *Semua bisa menjadi programmer JavaScript & Node. js*. Elex Media Komputindo, 2021.
- [16] I. K. B. M. Aryawan, “Development of Lecturer Data Management System Using Asynchronous JavaScript & XML (AJAX) Concepts,” *JIKA (Jurnal Informatika)*, vol. 6, no. 2, hlm. 212–219, 2022.
- [17] R. Hermiati, A. Asnawati, dan I. Kanedi, “Pembuatan E-Commerce Pada Raja Komputer Menggunakan Bahasa Pemrograman Php Dan Database Mysql,” *Jurnal Media Infotama*, vol. 17, no. 1, 2021.
- [18] A. Del Sole dan D. Sole, *Visual Studio Code Distilled*. Springer, 2019.
- [19] L. T. S. Sarwandi dkk., *Sistem pendukung keputusan*. Graha Mitra Edukasi, 2023.
- [20] R. Rusliyawati, D. Damayanti, dan S. N. Prawira, “Implementasi Metode Saw Dalam Sistem Pendukung Keputusan Pemilihan Model Social

Customer Relationship Management,” *Jurnal Ilmiah Edutic: Pendidikan dan Informatika*, vol. 7, no. 1, hlm. 12–19, 2020.

- [21] S. G. Andika, K. Kusnadi, dan P. Sokibi, “Sistem Pendukung Keputusan Pemilihan Kegiatan Ekstrakurikuler Untuk Siswa Sma Menggunakan Metode Simple Multi Attribute Rating Technique (Studi Kasus: Sma Santa Maria Cirebon),” *Jurnal Digit: Digital of Information Technology*, vol. 9, no. 1, hlm. 59–70, 2020.
- [22] F. Fariyanto, S. Suaidah, dan F. Ulum, “Perancangan Aplikasi Pemilihan Kepala Desa Dengan Metode Ux Design Thinking (Studi Kasus: Kampung Kuripan),” *Jurnal Teknologi Dan Sistem Informasi*, vol. 2, no. 2, hlm. 52–60, 2021.
- [23] I. Paputungan, A. Ansar, dan S. R. Mas, “Keefektifan Pelaksanaan Sistem Penjaminan Mutu Internal,” *Pedagogika*, hlm. 77–92, 2021.



RIWAYAT HIDUP PENULIS

Nama: Teuku Muksal Mina

Tempat Tanggal Lahir: Aceh Utara, 28 Agustus 2000

Jenis Kelamin: Laki-laki

Alamat Rumah: Dusun Tanjong Reuneng,
Geudong, Aceh Utara, Samudera

Telp/HP: 085322420875

E-Mail institusi: 190212026@student.ar-raniry.ac.id



RIWAYAT PENDIDIKAN

Sekolah Dasar (SD)/Sederajat: MIN 6 Samudera

Sekolah Menengah Pertama (SMP)/Sederajat: MTsN 4 Samudera

Sekolah Menengah Atas (SMA)/Sederajat: SMK Negeri 1 Lhoksukon

Perguruan Tinggi: UIN Ar-Raniry Banda Aceh

Fakultas/Program Studi: Pendidikan Teknologi Informasi

RIWAYAT KELUARGA

Nama Ayah: Teuku Idrus

Pekerjaan Ayah: Pegawai Negeri Sipil

Nama Ibu: Mardhiah

Pekerjaan Ibu: Ibu Rumah Tangga

Alamat Lengkap: Dusun Tanjong Reuneng, Geudong, Aceh utara, Samudera