

**Rekomendasi E-book pada Digital Repository Rumah Sakit
Umum Universitas Kristen Indonesia dengan menggunakan
Metode VIKOR**

SKRIPSI

Oleh:

SOYID WAHYU DARMAWAN NIM. 1741720151



**PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNOLOGI INFORMASI
POLITEKNIK NEGERI MALANG
2021**

**Rekomendasi E-book pada Digital Repository Rumah Sakit
Umum Universitas Kristen Indonesia dengan menggunakan
Metode VIKOR**

SKRIPSI

Digunakan Sebagai Syarat Maju Ujian Diploma IV

Politeknik Negeri Malang

Oleh:

SOYID WAHYU DARMAWAN NIM. 1741720141



**PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNOLOGI INFORMASI
POLITEKNIK NEGERI MALANG
JUNI 2021**

HALAMAN PENGESAHAN

Rekomendasi E-book pada Digital Repository Rumah Sakit Umum Universitas Kristen Indonesia dengan menggunakan Metode VIKOR

Disusun oleh:

SOYID WAHYU DARMAWAN NIM. 1741720151

Proposal Skripsi ini telah diuji pada ..., .. 2021

Disetujui oleh:

- | | | | |
|-----------------------------|---|---|-------|
| 1. Pembimbing
Utama | : | <u>Ridwan Rismanto, S.ST., M.Kom.</u>
NIP. 19860318 201212 1 001 | |
| 2. Pembimbing
Pendamping | : | <u>Habibie Ed Dien, S.Kom., MT.</u>
NIP. 19920412 201903 1 013 | |
| 3. Penguji Utama | : | <u>.....</u>
NIP. | |
| 4. Penguji
Pendamping | : | <u>.....</u>
NIP. | |

Mengetahui,

Ketua Jurusan
Teknologi Informasi

Ketua Program Studi
Teknik Informatika

Rudy Ariyanto, S.T., M.CS
NIP. 19711110 199903 1 002

Imam Fahrur Rozi, ST., MT.
NIP. 198406102008121004

PERNYATAAN

Dengan ini saya menyatakan bahwa Laporan Skripsi ini tidak terdapat karya yang pernah diajukan untuk memperoleh gelar Ahli Madya/Sarjana Terapan di suatu Perguruan Tinggi dan sepanjang pengetahuan saya juga tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis diacu dalam naskah ini dan disebutkan dalam daftar pustaka.

Malang, 2021

Soyid Wahyu Darmawan

ABSTRAK

Wahyu D., Soyid. “Rekomendasi E-book pada Digital Repository Rumah Sakit Umum Universitas Kristen Indonesia dengan menggunakan Metode VIKOR”.

Pendamping: (1) Ridwan Rismanto, S.ST., M.Kom. (2) Habibie Ed Dien, S.Kom., MT.

Skripsi, Program Studi Teknik Informatika, Jurusan Teknologi Informasi, Politeknik Negeri Malang, 2021.

Salah satu layanan di perpustakaan digital rumah sakit adalah sistem informasi untuk menemukan ketersediaan koleksi dari buku elektronik yang dimaksud. Tetapi tidak banyak sistem menyediakan mekanisme yang dapat merekomendasikan karyawan rumah sakit mengenai buku elektronik yang mereka inginkan untuk mereka jadikan literatur, tetapi juga buku elektronik lainnya yang mungkin perlukan. beberapa judul buku elektronik dengan kandungan isi yang berbeda-beda. Perbedaan dapat dilihat dari beberapa kriteria seperti kelayakan isi, kebahasaan, penyajian, dan kegrafikaan. Sehingga dibuatlah sistem rekomendasi untuk merekomendasikan *E-book* sebagai bahan pertimbangan sesuai yang diinginkan karyawan rumah sakit itu sendiri. Penggunaan metode VIKOR diterapkan karena termasuk dari *Multi Criteria Decision Making* untuk mengambil keputusan bersifat diskrit berdasarkan beberapa kriteria sebagai hasil terbaik dari rekomendasi sistem. Pembuatan sistem menggunakan bahasa pemrograman Python framework Flask sebagai backend API, Javascript framework ReactJs sebagai frontend, dan database MySql.

Kata Kunci : Perpustakaan Digital, Sistem Informasi, Rumah Sakit, E-book, VIKOR, Flask, ReactJS, MySql

ABSTRACT

Wahyu D., Soyid. *“E-book recommendations at the Digital Warehouse of the General Hospital of the Christian University of Indonesia using the VIKOR Method”*. **Counseling Lecturer: (1) Ridwan Rismanto, S.ST., M.Kom. (2) Habibie Ed Dien, S.Kom., MT.**

Thesis, Informatics Management Study Program, Department of Information Technology, State Polytechnic of Malang, 2021.

One of the services in the hospital's digital library is an information system to find the availability of a collection of the electronic books in question. However, not many systems provide a mechanism by which hospital employees can recommend the e-books they want them to turn into literature, but also any other e-books they may need. several titles of electronic books with different contents. The differences can be seen from several criteria such as content feasibility, language, presentation, and graphics. So that a recommendation system was created to recommend the E-book as a consideration according to what the hospital employees wanted. The use of the VIKOR method is applied because it is included in the Multi Criteria Decision Making to make discrete decisions based on several criteria as the best results from system recommendations. The system development uses the Python Flask framework programming language as the API backend, the Javascript framework ReactJs as the frontend, and the MySql database.

Keywords: *Digital Library, Information System, Hospital, E-book, VIKOR, Flask, ReactJS, MySql*

KATA PENGANTAR

Puji Syukur kami panjatkan kehadiran Allah SWT atas segala rahmat dan hidayah-Nya penulis dapat menyelesaikan laporan akhir dengan judul “Rekomendasi E-book pada Digital Repository Rumah Sakit Umum Universitas Kristen Indonesia dengan menggunakan Metode VIKOR”. Skripsi ini disusun sebagai persyaratan untuk menyelesaikan studi program Diploma IV Program Studi Teknik Informatika, Jurusan Teknologi Informasi, Politeknik Negeri Malang.

Kami menyadari tanpa adanya dukungan dan kerja sama dari berbagai pihak, kegiatan laporan akhir ini tidak akan dapat berjalan baik. Untuk itu, kami ingin menyampaikan rasa terima kasih kepada:

1. Allah SWT yang telah memberikan rahmat dan hidayat sehingga mampu menyelesaikan tugas akhir ini sehingga selesai tepat waktu.
2. Orang tua saya yang telah memberikan doa dan dukungan, baik secara moril maupun materil.
3. Bapak Rudy Ariyanto, ST., M.Cs., selaku ketua jurusan Teknologi Informasi
4. Bapak Imam Fahrur Rozi, S.T., M.T., selaku ketua program studi Teknik Informatika.
5. Bapak Ridwan Rismanto, S.ST., M.Kom. selaku dosen pembimbing utama.
6. Bapak Habibie Ed Dien, S.Kom., MT. selaku dosen pembimbing pendamping.
7. Teman – teman yang telah memberikan bantuan dan dukungan dalam membantu menyelesaikan laporan skripsi.
8. Seluruh dosen, staf, dan karyawan Politeknik Negeri Malang yang telah membantu penulis dalam menyelesaikan laporan skripsi ini.
9. Dan seluruh pihak yang telah membantu dan mendukung lancarnya pembuatan Laporan Akhir dari awal hingga akhir yang tidak dapat kami sebutkan satu persatu.

Penulis menyadari bahwa dalam penyusunan laporan akhir ini, masih banyak terdapat kekurangan dan kelemahan yang dimiliki penulis baik itu sistematika penulisan maupun penggunaan bahasa. Untuk itu penulis mengharapkan saran dan

kritik dari berbagai pihak yang bersifat membangun demi penyempurnaan laporan ini. Semoga laporan ini berguna bagi pembaca secara umum dan penulis secara khusus. Akhir kata, penulis ucapkan banyak terima kasih.

Malang, 2021

Soyid Wahyu Darmawan

DAFTAR ISI

HALAMAN PENGESAHAN.....	i
PERNYATAAN	ii
ABSTRAK.....	iii
<i>ABSTRACT</i>	iv
KATA PENGANTAR	v
DAFTAR ISI.....	vii
DAFTAR TABEL.....	ix
DAFTAR GAMBAR	x
DAFTAR LAMPIRAN.....	xii
BAB I. PENDAHULUAN.....	1
1.1. Latar Belakang	1
1.2. Rumusan Masalah	3
1.3. Tujuan	3
1.4. Batasan Masalah	4
1.5. Sistematika Penulisan	5
BAB II. LANDASAN TEORI.....	7
2.1. Penelitian Terdahulu	7
2.2. Sistem Pendukung Keputusan.....	9
2.3. Metode Vlsekriterijumska Optimizacija Kompromisno Resenje (VIKOR)	10
2.4. Python	11
2.5. JAVASCRIPT	12
2.6. MySQL	12
BAB III. METODOLOGI PENELITIAN	13
3.1. Data	13
3.2. Metode Pengembangan Perangkat Lunak.....	13
3.3. Metode Pengumpulan Data.....	15
3.4. Metode Pengolahan Data	15
3.5. Metode Pengujian	16
BAB IV. ANALISIS DAN PERANCANGAN	17
4.1. Analisis	17
4.2. Perancangan	37
BAB V. IMPLEMENTASI DAN PENGUJIAN	52
5.1. Implementasi Basis Data.....	52

5.2.	Implementasi Pengkodean Sistem	56
5.3.	Implementasi tampilan.....	68
5.4.	Pengujian Fungsionalitas sistem	72
BAB VI. HASIL DAN PEMBAHASAN		73
6.1.	Hasil Pengujian Akurasi.....	73
BAB VII. KESIMPULAN DAN SARAN.....		75
7.1.	Kesimpulan	75
7.2.	Saran	75
DAFTAR PUSTAKA		76
Lampiran		79

DAFTAR TABEL

Tabel 2.1.1. Tabel Penelitian terdahulu	7
Tabel 4.1.2.1. Tabel Analisis Pengguna	17
Tabel 4.1.3.1. Tabel Kebutuhan Fungsional	18
Tabel 4.1.4.1. Tabel Analisis Kebutuhan Perangkat Lunak	19
Tabel 4.1.4.2. Tabel Analisis Kebutuhan Perangkat Keras	19
Tabel 4.1.5.1. Tabel Deskripsi Aktor	21
Tabel 4.1.6.1. Tabel Skenario – mengelola data user	21
Tabel 4.1.6.2. Tabel Skenario – mengelola data nakes	22
Tabel 4.1.6.3. Tabel Skenario – mengelola data jenis buku	22
Tabel 4.1.6.4. Tabel Skenario – mengelola data penulis	23
Tabel 4.1.6.5. Tabel Skenario – mengelola data penerbit	23
Tabel 4.1.6.6. Tabel Skenario – mengelola data buku	24
Tabel 4.1.6.7. Tabel Skenario – mengelola data review	24
Tabel 4.1.6.8. Tabel Skenario – mengelola data admin	24
Tabel 4.1.6.9. Tabel Skenario – mengelola data nilai buku	25
Tabel 4.1.6.10. Tabel Skenario – melihat data buku	25
Tabel 4.1.6.11. Tabel Skenario – mengakses isi buku	26
Tabel 4.1.7.1. Tabel Spesifikasi use case	27
Tabel 4.2.1.1. Tabel Admin	37
Tabel 4.2.1.2. Tabel User	37
Tabel 4.2.1.3. Tabel Ulasan	38
Tabel 4.2.1.4. Tabel Jenis Buku	38
Tabel 4.2.1.5. Tabel Tenaga Kesehatan	38
Tabel 4.2.1.6. Tabel Penerbit	38
Tabel 4.2.1.7. Tabel Penulis	38
Tabel 4.2.1.8. Tabel Buku	38
Tabel 4.2.1.9. Tabel Kriteria	39
Tabel 4.2.1.10. Tabel Nilai Buku	39
Tabel 6.1.1. Tabel Pengujian Akurasi	73
Tabel 6.1.1. Tabel Perhitungan Pengujian Akurasi	74

DAFTAR GAMBAR

Gambar 3.1. Gambar Bagan Metodologi Penelitian	13
Gambar 3.2.1. Gambar Metode RAD	14
Gambar 3.4.1. Gambar Data Buku	16
Gambar 4.1.5.1. Gambar Diagram Use case	20
Gambar 4.1.8.1. Gambar Activity Diagram login admin	29
Gambar 4.1.8.2. Gambar Activity Diagram data nakes	30
Gambar 4.1.8.3. Gambar Activity Diagram mengelola data buku	31
Gambar 4.1.8.4. Gambar Activity Diagram mengelola nilai buku	32
Gambar 4.1.8.5. Gambar Activity Diagram login user	33
Gambar 4.1.8.6. Gambar Activity Diagram melihat data buku	34
Gambar 4.1.8.7. Gambar Activity Diagram melihat detail buku	35
Gambar 4.1.8.8. Gambar Activity Diagram mengisi nilai buku	36
Gambar 4.2.1.1. Gambar Perancangan Basis Data	37
Gambar 4.2.2.1.1. Gambar tampilan login	40
Gambar 4.2.2.1.2. Gambar tampilan dashboard	40
Gambar 4.2.2.1.3. Gambar tampilan data user	41
Gambar 4.2.2.1.4. Gambar tampilan data admin	41
Gambar 4.2.2.1.5. Gambar tampilan data ulasan	42
Gambar 4.2.2.1.6. Gambar tampilan data nakes	42
Gambar 4.2.2.1.7. Gambar tampilan data jenis buku	43
Gambar 4.2.2.1.8. Gambar tampilan data penerbit	43
Gambar 4.2.2.1.9. Gambar tampilan data penulis	44
Gambar 4.2.2.1.10. Gambar tampilan data buku	44
Gambar 4.2.2.1.11. Gambar tampilan data kriteria	45
Gambar 4.2.2.1.12. Gambar tampilan data nilai buku	45
Gambar 4.2.2.1.13. Gambar tampilan data Rekomendasi	46
Gambar 4.2.2.2.1. Gambar tampilan register	46
Gambar 4.2.2.2.2. Gambar tampilan login	47
Gambar 4.2.2.2.3. Gambar tampilan lupa password	47
Gambar 4.2.2.2.4. Gambar tampilan beranda	48
Gambar 4.2.2.2.5. Gambar tampilan data buku	48
Gambar 4.2.2.2.6. Gambar tampilan data buku kedokteran	49

Gambar 4.2.2.2.7. Gambar tampilan data buku keperawatan	49
Gambar 4.2.2.2.8. Gambar tampilan data buku farmasi	50
Gambar 4.2.2.2.9. Gambar tampilan data buku rekomendasi	50
Gambar 4.2.2.2.10. Gambar tampilan detail buku	51
Gambar 4.2.2.2.11. Gambar tampilan ulasan buku	51
Gambar 5.3.1.1. Gambar Tampilan Dashboard	68
Gambar 5.3.1.2. Gambar Tampilan Kriteria Buku	69
Gambar 5.3.1.3. Gambar Tampilan Rekomendasi Buku	69
Gambar 5.3.2.1. Gambar Tampilan Buku	70
Gambar 5.3.2.2. Gambar Rekomendasi Buku	71
Gambar 5.3.2.3. Gambar Tampilan Detail Buku	71

DAFTAR LAMPIRAN

- Lampiran 1 Data Buku
- Lampiran 1 Menu Utama Aplikasi
- Lampiran 2 Keluaran Sistem
- Lampiran 3 Kuesioner Uji Pengguna
- Lampiran 4 *Listing Program*

BAB I. PENDAHULUAN

1.1. Latar Belakang

Tantangan baru teknologi informasi khususnya penyedia informasi adalah bagaimana menyalurkan informasi secara cepat, tepat dan global. Dengan pesatnya perkembangan teknologi informasi memberikan peluang untuk mengelola informasi dalam format digital (Radjatadoe et al., 2017). Digital library merupakan kumpulan kumpulan koleksi digital berupa konten dan jasa (fungsi) terkait dengan penyimpanan, penemuan, pencarian dan pelestarian ilmu pengetahuan (Wicaksana et al., 2017).

Digital Repository seringkali dikaitkan dengan upaya menghimpun karya-karya intelektual dan materi ilmiah dalam format digital yang dapat diakses secara online (Qurotianti, 2019). Peran lain dengan Adanya digital repository ini memungkinkan pengguna untuk mengakses seluruh informasi secara online dalam format digital sehingga masyarakat semakin dimudahkan dengan kemudahan akses dan keamanan tempat (Artikel, 2017).

Studi Kasus dari penelitian ini adalah di Rumah Sakit Umum Universitas Kristen Indonesia. Berada di bagian P4 (Pendidikan, Perencanaan, Pengembangan, dan Penelitian) ini sebagai divisi yang mendalami urusan internal dari Rumah Sakit UKI. Demi mewujudkan Rumah Sakit yang mempunyai keunggulan dalam pelayanan dan pendidikan di bidang kesehatan (*Informasi Lengkap Dan Buat Janji Di RSU Universitas Kristen Indonesia (UKI). Biaya, Tindakan Medis, Daftar Dokter, Selengkapnya.*, n.d.).

Berdasarkan penjelasan tersebut, penulis memutuskan untuk memilih topik dari penelitian adalah dengan menggunakan Sistem Pendukung Keputusan (SPK). dengan menerapkan Sistem Pendukung Keputusan (SPK), suatu sistem informasi berbasis komputer yang mengkombinasikan model dan data untuk menyediakan dukungan kepada pengambil keputusan dalam memecahkan masalah terstruktur disebut Sistem Pendukung Keputusan (Pohan & Sinaga, 2020).

Alasan adanya Rekomendasi E-book ini supaya user mengetahui literatur untuk dibaca dengan mudah berdasarkan kriteria penilaian E-book tersebut. Gambaran sistem ini adalah saat user menginputkan nilai kriteria dan nilai tersebut tersimpan pada database sehingga untuk memunculkan hasil penilaian tersebut, ditampilkan sebuah combo box dengan nama “rekomendasi” sehingga user mengetahui Rekomendasi E-book berdasarkan penilaian tersebut. Kriteria penilaian sebuah E-book adalah melihat dari segi kriteria Kelayakan isi, Kebahasaan, Penyajian, dan Kegrafikaan (Arsanti, 2018; Meningkatkan et al., 2020).

Metode yang tepat untuk penelitian ini adalah dengan menggunakan metode VIKOR (VlseKriterijumska Optimizacija Kompromisno Resenje). metode perankingan dengan menggunakan indeks peringkat multikriteria berdasarkan ukuran tertentu dari kedekatan dengan solusi yang ideal (Arisandi & Pribadi, 2020). Lalu metode VIKOR adalah metode untuk optimasi kriteria majemuk dalam suatu sistem yang kompleks dengan Konsep yang menentukan peringkat dari sampel yang ada dengan melihat hasil dari nilai regrets measure (R) dari setiap sampel (Sharma, 2017).

Alasan saya memilih Metode VIKOR karena Metode VIKOR sangat berguna pada situasi dimana pengambil keputusan tidak memiliki kemampuan untuk menentukan pilihan pada saat desain sebuah sistem dimulai (Tumanggor et al., 2018). Sehingga metode VIKOR sangat berpengaruh pada hasil penentuan rekomendasi E-book yang diteliti.

Berdasarkan hal tersebut, penulis memutuskan untuk membuat suatu Sistem Pendukung Keputusan untuk merekomendasikan E-book pada Digital Repository Rumah Sakit Umum UKI dengan menggunakan metode VIKOR. Dengan adanya sistem ini, diharapkan dapat membantu dan mempermudah pihak Rumah Sakit Umum UKI dalam merekomendasi E-book pada Digital Repository dengan baik.

1.2. Rumusan Masalah

Berdasarkan Latar Belakang di atas, Rumusan Masalah yang dapat diambil adalah:

1. Bagaimana membuat sistem rekomendasi yang menghasilkan E-book dengan nilai tertinggi berdasarkan kriteria dari E-book tersebut?
2. Bagaimana membangun sistem rekomendasi E-book pada Digital Repository berdasarkan kriteria dengan menggunakan VIKOR?
3. Bagaimana tingkat akurasi VIKOR untuk diimplementasikan ke dalam sistem rekomendasi E-book berdasarkan kriteria dari E-book tersebut?

1.3. Tujuan

Tujuan dari penelitian dengan judul Rekomendasi E-book pada Digital Repository Rumah Sakit Umum Universitas Kristen Indonesia dengan menggunakan Metode VIKOR adalah sebagai berikut:

1. Untuk membuat sistem yang menghasilkan rekomendasi E-book yang sesuai berdasarkan kriteria penilaian buku. Hasil penentuan tersebut digunakan menjadi salah satu tolak ukur dalam menentukan Rekomendasi E-book.
2. Untuk memberikan rekomendasi E-book yang sesuai dengan kriteria penilaian buku dengan menggunakan Sistem Rekomendasi yang diimplementasikan dengan metode VIKOR
3. Dapat mengetahui tingkat akurasi dalam implementasi metode VIKOR kedalam sistem rekomendasi E-book berdasarkan kriteria penilaian buku (Kelayakan isi, Kebahasaan, Penyajian, Dan Kegrafikaan)

1.4. Batasan Masalah

Agar skripsi yang berjudul Rekomendasi E-book pada Digital Repository Rumah Sakit Umum Universitas Kristen Indonesia dengan menggunakan Metode VIKOR dapat berjalan sesuai rencana dan tujuan awal, maka penulis menetapkan batasan-batasan masalah yaitu:

1. Aplikasi yang dibuat berbasis website dengan menggunakan Bahasa pemrograman JAVASCRIPT dan framework ReactJS
2. Perhitungan metode hanya dapat dilakukan oleh satu user sehingga tidak adanya suatu GDSS (Group Decision Support System) atau Sistem Pendukung Keputusan secara berkelompok
3. Penentuan Rekomendasi E-book pada Digital Repository berdasarkan kriteria penilaian menggunakan metode VIKOR.
4. Keluaran dari sistem berupa rekomendasi E-book yang ditampilkan di laman Website Digital Repository.
5. Jumlah minimal buku yang di digitalisasi berjumlah 50 buku
6. Jumlah ukuran E-book yang dapat di upload ke sistem Digital Repository berukuran maksimal 350 MB (*Mega Byte*)

1.5. Sistematika Penulisan

Sistematika Penulisan skripsi dengan judul “Rekomendasi E-book pada Digital Repository Rumah Sakit Umum Universitas Kristen Indonesia dengan menggunakan Metode VIKOR” dengan susunan sebagai berikut :

BAB I : PENDAHULUAN

Pendahuluan berisi tentang latar belakang, rumusan masalah, tujuan, batasan masalah, sistematika penulisan. Bagian pendahuluan akan menjadi dasar penelitian yang dilakukan.

BAB II : LANDASAN TEORI

Berisikan teori-teori yang relevan yang melengkapi latar belakang. Sekaligus memberi *review* tentang pustaka yang telah dibaca selama masa pencarian solusi terhadap masalah yang diangkat dalam tugas akhir. Pustaka yang dijadikan rujukan atau referensi merupakan publikasi ilmiah. Publikasi ilmiah dapat berbentuk jurnal, prosiding, modul ajar, diktat dan lain-lain.

BAB III : METODOLOGI PENELITIAN

Terdiri dari langkah-langkah yang akan membimbing penulis memilih metode, teknik, prosedur apa yang tepat, dan *tools* apa yang akan digunakan sehingga setiap tahapan dapat dilakukan dengan tepat, termasuk desain dan perancangan sistem yang akan dibuat.

BAB IV : ANALISIS DAN PERANCANGAN

Pada bagian ini diuraikan dengan jelas sistem yang akan dibuat dan kebutuhan sistem yang meliputi kebutuhan fungsional dan kebutuhan nonfungsional. Rancangan sistem meliputi rancangan model sistem, rancangan arsitektur sistem, rancangan proses, rancangan prosedural, rancangan data dan rancangan antarmuka pengguna

BAB V : IMPLEMENTASI DAN PENGUJIAN

Pada bagian ini, implementasi sistem dipaparkan secara detil sesuai rancangan dan komponen (*tools*) bahasa pemrograman yang dipakai. Implementasi rancangan proses dapat disertai dengan potongan kode pada proses yang dimaksud. Pengujian merupakan proses untuk menentukan apakah hasil dari tugas akhir sudah sesuai dengan kebutuhan sistem dan berjalan sesuai lingkungan yang diinginkan.

Pengujian dapat berupa pengujian fungsional, penerimaan pengguna, ataupun performa sistem. Pada pengujian dipaparkan secara detil mengenai metode pengujian, tujuan pengujian, proses pengujian serta analisis hasil pengujian.

BAB VI : HASIL DAN PEMBAHASAN

Pembahasan merupakan argumentasi rasional dari penulis yang disusun secara sistematis berdasarkan fakta ilmiah yang diperoleh dari hasil pengujian. Pembahasan hasil pengujian dapat disajikan dalam bentuk uraian teoritik, baik secara kualitatif maupun kuantitatif. Secara ilmiah, hasil pengujian yang diperoleh dalam penelitian dapat berupa temuan baru atau perbaikan, penegasan atau penolakan interpretasi suatu fenomena ilmiah yang diteliti pada penelitian sebelumnya. Penyajian hasil penelitian dapat diperjelas dengan menggunakan tabel, kurva, grafik, foto atau bentuk lain dapat digunakan sesuai keperluan secara lengkap dan jelas.

BAB VII : KESIMPULAN DAN SARAN

Berisi uraian singkat dan jelas tentang hasil tugas akhir yang diperoleh sesuai dengan tujuan penelitian. Apabila diperlukan, saran dapat digunakan untuk menyampaikan hal-hal yang dapat diperbaiki, dikembangkan atau dijadikan penelitian lebih lanjut.

BAB II. LANDASAN TEORI

Tinjauan pustaka merupakan bagian yang akan membahas tentang penyelesaian masalah yang akan memberikan jalan keluar atau hasil. Dalam hal ini akan dikemukakan beberapa teori yang akan berkaitan dengan permasalahan yang akan diangkat.

2.1. Penelitian Terdahulu

Penelitian terdahulu ini menjadi salah satu acuan penulis untuk melakukan penelitian sehingga penulis dapat memperluas teori yang digunakan dalam mengkaji penelitian yang dilakukan. Dari penelitian terdahulu, penulis tidak menemukan penelitian dengan judul yang sama dengan judul penelitian penulis terutama di bagian metode penelitian. Namun, penulis mengangkat beberapa penelitian untuk dijadikan referensi dalam menambah kajian pada penelitian penulis. Berikut merupakan penelitian terdahulu berupa beberapa jurnal terkait dengan penelitian yang dilakukan penulis.

Tabel 2.1.1. Tabel Penelitian terdahulu

Nama Peneliti	Judul Penelitian	Hasil Penelitian
Farmy Setiawan Radjatadoe, Dony M. Sihotang, Meiton Boru 2017	Pembuatan aplikasi digital library di jurusan ilmu Komputer universitas nusa cendana berbasis web	Dalam menjalankan aplikasi berjalan dengan baik karena hasil pengujian sesuai dengan hasil yang diharapkan dan dari aspek fungsionalitas persentase tertinggi pada penilaian tanggapan sangat baik

Fikri Firgiawan, Wina Witanti dan Gunawan Abdillah 2018	Sistem pendukung keputusan rekomendasi pemilihan buku komputer di amazon.com menggunakan metode technique for order preference by similarity to ideal solution (topis)	Dalam penelitian menjelaskan kalau pembuatan Aplikasi Sistem Rekomendasi dengan menggunakan Sistem Pendukung Keputusan dapat memberikan hasil yang baik
Boyma Simamora 2017	Rancang bangun sistem rekomendasi televisi led dengan metode vikor berbasis web	Berdasarkan penelitian yang telah dilakukan, dapat disimpulkan bahwa rancang bangun sistem rekomendasi pembelian televisi LED dengan metode VIKOR dengan kriteria sudah dapat diverifikasi dengan membandingkan hasil perhitungan sistem dengan perhitungan manual
Betrik Juliana Hutapea, Mentari Ananda Hasmi , Abdul Karim, Suginam 2018	Sistem pendukung keputusan penentuan jenis kulit terbaik untuk pembuatan sepatu dengan menggunakan metode vikor	hasil penelitian adalah dengan membahas sistem pendukung yang dapat membantu untuk menentukan jenis kulit terbaik dengan menggunakan metode VIKOR.

Dari beberapa judul penelitian yang telah dipaparkan, terpilih penelitian yang dilakukan oleh Boyma Simamora yang berjudul “Rancang bangun sistem rekomendasi televisi led dengan metode vikor berbasis web” sebagai referensi utama penulis dalam melakukan penelitian. Hal ini dikarenakan adanya pembahasan yang serupa dari segi ide dan konsep dari penelitian yang dilakukan. Dan Jurnal yang berjudul “Sistem Pendukung Keputusan Rekomendasi Pemilihan Buku Komputer Di Amazon.Com Menggunakan Metode Technique For Order Preference By Similarity To Ideal Solution (Topsis)” menjelaskan bahwa dalam mencari rekomendasi dapat digunakan dengan menggunakan Sistem Pendukung Keputusan (Firgiawan et al., 2018).

2.2.Sistem Pendukung Keputusan

Sistem pendukung keputusan adalah bagian dari sistem informasi berbasis komputer (termasuk sistem pengetahuan) yang dipakai untuk mendukung pengambilan keputusan dalam suatu organisasi atau perusahaan. SPK merupakan penggabungan sumber- sumber kecerdasan individu dengan kemampuan komponen untuk memperbaiki kualitas keputusan (Hutapea Mentari Ananda; Karim, Abdul; Suginam, Suginam, 2018).

Sistem ini digunakan untuk membantu pengambilan keputusan dalam situasi semi terstruktur dan situasi yang tidak terstruktur, dimana tak seorang pun tahu secara pasti bagaimana keputusan seharusnya dibuat (Hamria Hamria et al., 2020). Sistem ini digunakan untuk membantu pengambilan keputusan dalam situasi yang semi terstruktur dan situasi yang tidak terstruktur (Safii & Zulhamsyah, 2018).

2.3. Metode VlseKriterijumska Optimizacija Kompromisno Resenje (VIKOR)

VIKOR berasal dari kata VlseKriterijumska Optimizacija Kompromisno Resenje yang merupakan metode analisis pengambilan keputusan dengan multi atribut decision making yang dikembangkan oleh Serafim Opricovic untuk memecahkan permasalahan keputusan dengan kriteria yang saling bertentangan dan dari unit yang berbeda, dengan asumsi bahwa kompromi dapat diterima sebagai resolusi dari konflik yang ada (Bako & Lubis, 2018).

Metode vikor memiliki kelebihan dalam mengkompromi alternatif yang ada, serta dapat menyelesaikan pengambilan keputusan bersifat diskret pada kriteria yang bertentangan dan non commensurable, yaitu perbedaan unit antar kriteria (Sutrikanti et al., 2018).

Langkah-langkah yang digunakan dalam Metode VIKOR adalah sebagai berikut (Simamora, 2017):

1. Dengan menentukan tabel pengamatan dari database dengan mencari nilai data terbaik (f_i^*) dan terburuk (f_i^-) atau dengan istilah *Cost* dan *Benefit*
2. Menentukan bobot kriteria yang diperoleh dari pengguna sesuai dengan kebutuhan atau kriteria yang diinginkan
3. Menghitung Normalisasi matriks

$$R_{ij} = \frac{(f_i^*) - (f_{ij})}{(f_i^*) - (f_i^-)} \quad (\text{Rumus 2.3.1})$$

Keterangan :

R_{ij} = nilai normalisasi sampel i pada kriteria j

f_{ij} = nilai data sampel i pada kriteria j

f_i^* = nilai terbaik dalam satu kriteria

f_i^- = nilai terjelek dalam satu kriteria

4. Menghitung Normalisasi bobot ($W_j \times R_{ij}$)

Melakukan perkalian antara nilai data yang telah dinormalisasi dengan nilai bobot kriteria yang telah ditentukan

5. Menghitung nilai *Utility Measure (S)* dan *Regret Measure (R)*

$$S_j = \sum_{i=1}^n W_i \left(\frac{(f_i^+) - (f_{ij})}{(f_i^+) - (f_i^-)} \right) \quad (\text{Rumus 2.3.2})$$

6. Menghitung indeks VIKOR

$$Q_j = \left[\frac{S_j - S^*}{S^- - S^*} \right] \times v + \left[\frac{R_j - R^*}{R^- - R^*} \right] \times (1 - v) \quad (\text{Rumus 2.3.3})$$

Keterangan :

S^* = nilai S terkecil

S^- = nilai S terbesar

R^* = nilai R terkecil

R^- = nilai R terbesar

7. Perankingan alternatif

Setelah Q_j dihitung, maka pengurutan perankingan ditentukan dari nilai yang paling rendah dengan solusi kompromi sebagai solusi ideal dari perankingan Q_j dengan nilai terendah. Karena nilai S_j merupakan solusi yang diukur dari titik terjauh solusi ideal, sedangkan nilai R_j merupakan solusi yang diukur dari titik terdekat solusi ideal.

2.4.Python

Python adalah bahasa pemrograman tujuan umum yang ditafsirkan, tingkat tinggi dan Python menekankan keterbacaan kode dengan penggunaan spasi putih yang signifikan. (*Python (Bahasa Pemrograman)* - Wikipedia Bahasa Indonesia, *Ensiklopedia Bebas*, n.d.). Sintaks Python yang sederhana dan mudah dipelajari menekankan pada keterbacaan dan karena itu mengurangi biaya pemeliharaan program. Python mendukung modul dan paket, yang mendorong modularitas program dan penggunaan kembali kode (*What Is Python? Executive Summary* / *Python.Org*, n.d.).

Flask merupakan microframework Python yang digunakan untuk membangun sebuah web service. Pada web service, Flask bertindak sebagai penengah antara website dengan basis data, flask juga bertindak untuk mengambil data dari basis data yang kemudian dijadikan umpan balik. (Parlika et al., 2020)

2.5.JAVASCRIPT

JavaScript yaitu salah satu bahasa pemrograman yang paling populer digunakan dalam kurun waktu dua puluh tahun ini. Bahkan JavaScript juga dikenal juga sebagai salah satu dari tiga bahasa pemrograman yang utama bagi web developer (*Apa Itu JavaScript ? Pemahaman Dasar Bagi Pemula*, n.d.). Sebagai bahasa multi-paradigma, JavaScript mendukung gaya pemrograman yang digerakkan oleh peristiwa, fungsional, dan imperative (*JavaScript - Wikipedia*, n.d.).

Secara fungsional, JavaScript digunakan untuk menyediakan script pada objek yang dibenamkan (Ubaya et al., 1907). ReactJs merupakan kerangka kerja open source yang menggunakan library javascript untuk membuat user interface dan React biasa digunakan untuk menangani pengembangan pada aplikasi single-page dan aplikasi mobile. ReactJS memiliki keunggulan dimana kerangka kerja ini memberikan kecepatan, simplicity, dan scalability (Nursaid et al., 2020).

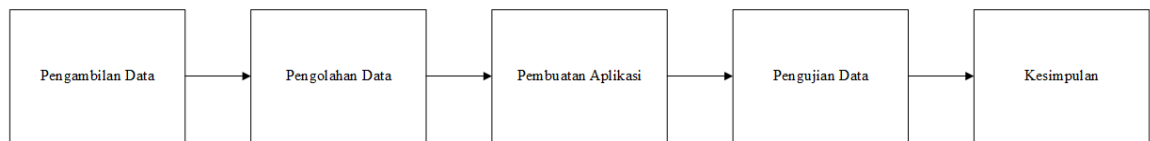
2.6.MySQL

MySQL adalah DBMS yang open source dengan dua bentuk lisensi, yaitu Free Software (perangkat lunak bebas) dan Shareware (perangkat lunak berpemilik yang penggunaannya terbatas) (*Pengertian MySQL, Fungsi, Dan Cara Kerjanya (Lengkap)*, n.d.).

MySQL lebih sering digunakan dengan program lain untuk mengimplementasikan aplikasi yang membutuhkan kemampuan database relasional (*MySQL - Wikipedia*, n.d.). MySQL merupakan perangkat lunak manajemen basis data yang berlisensi open source. MySQL berkinerja yang sangat cepat, reliable serta mudah digunakan untuk melakukan organisasi basis data (Aprida & Febriliyan, 2013).

BAB III. METODOLOGI PENELITIAN

Proses pengambilan data dilakukan dengan cara melakukan mengambil data yang sudah dimiliki oleh pihak P4 (Pendidikan, Perencanaan, Pengembangan, dan Penelitian) Rumah Sakit Umum Universitas Kristen Indonesia untuk mendapatkan data yang dibutuhkan rekomendasi *E-book*, di dalam aplikasi digital repository, meliputi pengolahan data untuk menghitung perankingan rekomendasi *E-book*. Kesimpulan didapatkan sesuai dengan hasil dan kendala pada proses penelitian.



Gambar 3.1. Gambar Bagan Metodologi Penelitian

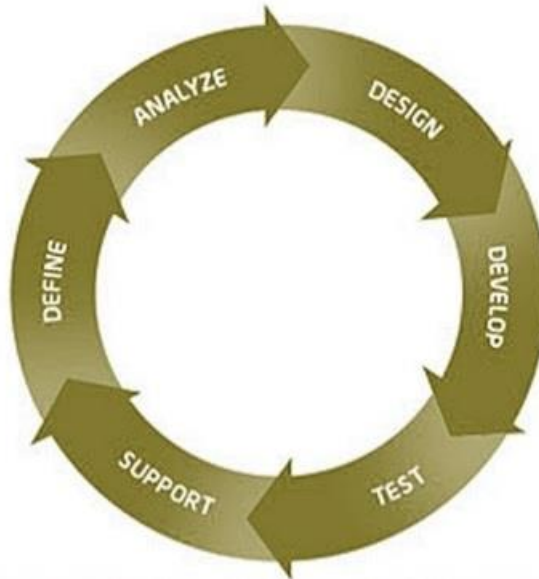
3.1. Data

Data yang diolah merupakan data *E-book* yang didapatkan dari pihak P4 (Pendidikan, Perencanaan, Pengembangan, dan Penelitian) Rumah Sakit Umum Universitas Kristen Indonesia selama 5 bulan dimulai pada bulan Januari 2021 sampai dengan Mei 2021.

3.2. Metode Pengembangan Perangkat Lunak

Metode pengembangan perangkat lunak yang digunakan pada penelitian ini menerapkan SDLC (*System Development Life Cycle*) dengan model RAD (*Rapid Application Development*). Model RAD adalah metode yang berfokus pada pengembangan aplikasi secara cepat, melalui pengulangan dan feedback berulang-ulang. Metode ini merupakan semacam cikal bakal *agile project management*, karena bisa mengikuti pace bisnis yang terus berkembang dan juga kebutuhan pasar yang terus meningkat. Pengembangan software pada umumnya seperti waterfall model membutuhkan perencanaan yang terbilang cukup kaku. Klien atau pelanggan seakan ‘dipaksa’ untuk menyetujui banyak hal di awal, tetapi mereka tidak bisa melihat proses pembuatannya.

Keuntungan utama menjalankan rapid application development adalah jangka waktu pengembangan lebih cepat. Hal ini dikarenakan feedback dari pelanggan cepat didapatkan dan semua perubahan yang dilakukan akan sesuai hasil tersebut. Akan tetapi, salah satu kekurangan RAD adalah kamu membutuhkan tim berisikan developer yang benar-benar memiliki skill tinggi dan juga metode ini hanya bisa digunakan untuk proyek yang bisa termodulasi.



Gambar 3.2.1. Gambar Metode RAD

Metode RAD memiliki tahapan-tahapan sebagai berikut :

- a. Menentukan project requirement
berisikan apa saja yang harus dicapai dalam sebuah proyek dan strategi untuk menghadapi permasalahan yang mungkin akan muncul. Persyaratan ini juga mencakup timeline dan budget yang ada. Orang-orang yang membuat project requirements adalah para developer, klien, dan juga software user.
- b. Membuat prototype
mengembangkan prototipe secara cepat, dengan fitur dan fungsi yang dibutuhkan. Setelah itu, prototipe tersebut akan diberikan kepada klien untuk mengetahui apa saja yang mereka suka dan apa yang tidak. Pada

tahap ini, hasil yang diberikan sama sekali belum sempurna, hanya menunjukkan fitur dan fungsi yang akan ada saja.

c. *Rapid construction* dan pengumpulan *feedback*

mencakup fitur, fungsi, visual, dan juga interface dari program yang sedang dikembangkan. Setelah itu, prototipe akan dikembangkan lagi sampai klien memberikan persetujuan untuk finalisasi produk.

d. Implementasi

Fitur, fungsi, visual, dan *interface* akan diulas kembali oleh klien. Pada tahap ini, uji coba akan dilakukan jika memang dibutuhkan. Uji cobanya mencakup kestabilan, *usability testing*, dan pengujian lainnya untuk memastikan semua hal sudah terkontrol.

3.3. Metode Pengumpulan Data

Metode Pengumpulan data dalam penyusunan penulisan skripsi ini adalah :

1. Studi Literatur dengan tujuan memahami algoritma VIKOR dengan mengumpulkan beberapa jurnal dan E-book melalui website.
2. Pengumpulan data dimulai dengan pengumpulan buku elektronik (*e-book*) dalam bentuk digital (.pdf) yang sudah dimiliki pihak P4 (Pendidikan, Perencanaan, Pengembangan, dan Penelitian) sebagai data dalam proses dalam rekomendasi E-book pada Digital Repository Rumah Sakit Umum Universitas Kristen Indonesia.

3.4. Metode Pengolahan Data

Tahap ini merupakan tahap pengumpulan data yang digunakan untuk proses pengolahan data yang diperoleh dari *e-book* yang dimiliki pihak P4 (Pendidikan, Perencanaan, Pengembangan, dan Penelitian) yang akan diolah pada sistem. Data buku ini digunakan sebagai tolak ukur dalam pembuatan aplikasi rekomendasi *E-book* pada Digital Repository Rumah Sakit Umum Universitas Kristen Indonesia

No	Kode Buku	Nama Buku	Jenis Buku	Tenaga Kesehatan	Penerbit	Penulis
1	B1	Leading and Managing in Nursing	Medikal Surgikal	Keperawatan	Elsevier	Patricia
2	B2	Assessment and Management of Clinical Problems	Medikal Surgikal	Keperawatan	Elsevier	Harding
3	B3	Nursing Interventions and Collaborative Management	Medikal Surgikal	Keperawatan	Elsevier	Marianne
4	B4	Netter's Pediatrics	Pediatrics	Keperawatan	Elsevier	Todd a Florin
5	B5	Drug-Drug Interaction Primer	Drugs	Farmasi	American Psychiatric	Neil B Sandsom
6	B6	Clinical Manual of Drug Interaction Principles	Drugs	Farmasi	American Psychiatric	Gary Winn
7	B7	Atlas of Human Anatomy	Anatomy	Kedokteran	Elsevier	Frank Netter
8	B8	Guyton and Hall Textbook of Medical Physiology	Psychology	Kedokteran	Elsevier	John Hall
9	B9	Drugs And Human Location	Drugs	Farmasi	Elsevier	Todd a Florin
10	B10	PRESCRIBING FOR ELDERLY PATIENTS	Prescribing	Farmasi	Wiley blackwell	Stephen Jackson
11	B11	Bailey & Loves Essential Clinical Anatomy	Anatomy	Kedokteran	Paperback	John Lunley
12	B12	Larsens Human Embryology	Embryology	Kedokteran	Wiley blackwell	Gary Schoenwolf

Gambar 3.4.1. Gambar Data Buku

3.5. Metode Pengujian

Pengujian sistem merupakan pengujian tingkat kualitas sistem dari cara kerja dan penggunaan sistem yang dibuat. Pengujian sistem ini meliputi pengujian fungsionalitas, pengujian akurasi, dan usability testing.

1. Pengujian fungsionalitas dilakukan dengan menggunakan metode pengujian black box. Pengujian ini menggunakan data uji berupa data input dari user pada sistem. Hasil yang diperoleh dari pengujian black box adalah berjalannya fungsi pada sistem sesuai dengan yang diharapkan.
2. Pengujian akurasi digunakan untuk mengetahui tingkat keakuratan perhitungan algoritma VIKOR dengan melakukan pencocokan hasil data Excel.
3. Usability testing digunakan untuk melakukan pengujian pada tingkat manfaat aplikasi sistem bagi user dalam menentukan Rekomendasi E-book pada Digital Repository.

BAB IV. ANALISIS DAN PERANCANGAN

4.1. Analisis

4.1.1. Deskripsi Sistem

Sistem akan ada dua pengguna yaitu admin dan user atau karyawan Rumah Sakit Umum Universitas Kristen Indonesia. Dimana admin mengelola data admin, user, nakes, jenis buku, penerbit, penulis, buku, kriteria, nilai buku, dan perhitungan metode VIKOR. User atau karyawan dapat melihat E-book, melihat E-book berdasarkan rekomendasi, melihat detail E-book, membaca E-book, dan menilai E-book berdasarkan kriteria-kriteria.

4.1.2. Analisis Pengguna

Berikut hasil analisis pengguna yang telah diidentifikasi :

Tabel 4.1.2.1. Tabel Analisis Pengguna

No	Pengguna	Deskripsi
1	Admin	Mengelola pengolahan data meliputi data admin, data user, data nakes, data jenis buku, data penerbit, data penulis, data buku, data kriteria, data nilai buku, dan data perhitungan metode VIKOR
2	Karyawan Rumah Sakit UKI	Mengolah dengan melihat data buku, data buku berdasarkan rekomendasi, data buku berdasarkan tenaga kesehatan, data buku berdasarkan tahun terbit, data buku berdasarkan urutan buku, melihat detail buku, dan menilai buku

4.1.3. Analisis Kebutuhan Fungsional

Kebutuhan fungsional merupakan kebutuhan yang membahas mengenai proses-proses yang berkaitan dengan sistem. Dalam sistem yang dibuat terdapat dua pengguna yaitu :

1. Admin

Pada sistem ini, admin melakukan pengolahan data meliputi data admin, data user, data nakes, data jenis buku, data penerbit, data penulis, data buku, data kriteria, data nilai buku, dan data perhitungan metode VIKOR

2. Karyawan

Pada sistem ini, karyawan melakukan Mengolah dengan melihat data buku, data buku berdasarkan rekomendasi, data buku berdasarkan tenaga kesehatan, data buku berdasarkan tahun terbit, data buku berdasarkan urutan buku, melihat detail buku, dan menilai buku

Tabel 4.1.3.1. Tabel Kebutuhan Fungsional

Data yang digunakan	<ul style="list-style-type: none"> - Data nilai buku - Data kriteria - Data buku
Admin	Menginputkan data yang digunakan meliputi data admin, data user, data nakes, data jenis buku, data penerbit, data penulis, data buku, data kriteria, data nilai buku, dan data perhitungan metode VIKOR
Karyawan	Menginputkan data nilai buku berdasarkan kriteria penilaian buku

4.1.4. Analisis Kebutuhan Non-Fungsional

Kebutuhan non-fungsional merupakan kebutuhan yang fokus pada pelaku yang terdapat pada sistem. Berikut merupakan kebutuhan non-fungsional :

1. Keamanan

Keamanan pada sistem ini telah dilengkapi dengan adanya penggunaan password ketika admin dan pihak pelaksana melakukan proses login dengan penggunaan JWT (Json Web Token)

2. Spesifikasi untuk Developer

Spesifikasi yang dibutuhkan untuk developer dalam membangun sistem aplikasi ini akan dijelaskan pada tabel dibawah ini :

a. Analisis Kebutuhan Perangkat Lunak

Berikut adalah analisis kebutuhan perangkat lunak yang dibutuhkan, akan dijelaskan pada tabel 4.1.4.1.

Tabel 4.1.4.1. Tabel Analisis Kebutuhan Perangkat Lunak

No	Perangkat	Perangkat lunak
1	<i>Web browser</i>	Brave Browser
2	<i>Web server</i>	Apache
3	<i>DBMS</i>	MySQL
4	<i>Text Edior dan IDE</i>	Jetbrains Pycharm dan Jetbrains IntelliJIDEA
5	<i>Framework</i>	Flask dan ReactJS

b. Analisis Kebutuhan Perangkat Keras

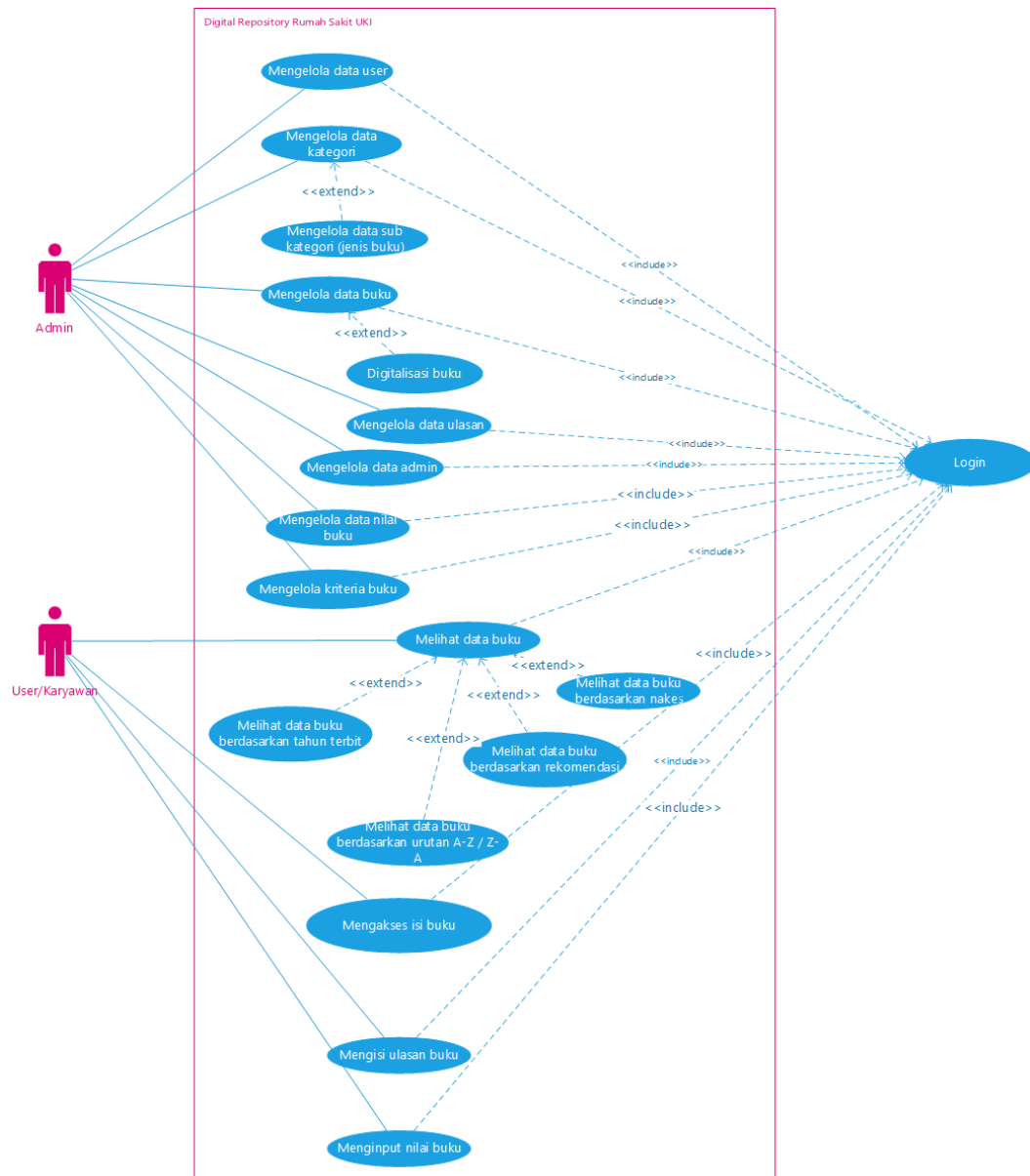
Berikut ini adalah analisis kebutuhan perangkat lunak yang dibutuhkan, akan dijelaskan pada Tabel 4.1.4.2.

Tabel 4.1.4.2. Tabel Analisis Kebutuhan Perangkat Keras

No	Perangkat	Perangkat Keras
1	<i>Processor</i>	Intel Core I7 – 9750H
2	<i>Memory</i>	24 GB
3	<i>SSD</i>	512 GB
4	<i>Harddisk</i>	1 TB
5	Perangkat Input	Mouse, Harddisk External, Flashdisk, dan mousepad

4.1.5. Diagram *Use case*

Diagram *usecase* merupakan interaksi yang menggambarkan hubungan antara aktor dengan kegiatan yang terdapat di dalam sistem. Berikut tampilan diagram usecase pada gambar 4.1.5.1 :



Gambar 4.1.5.1. Gambar Diagram Use case

Berikut penjelasan aktor dari use case diagram pada tabel 4.1.5.1.

Tabel 4.1.5.1. Tabel Deskripsi Aktor

Aktor	Deskripsi
Admin	Admin adalah orang yang bertanggung jawab atas segala urusan yang berkaitan tentang mengelola data-data sistem yaitu admin, user, nakes, jenis buku, penulis, penerbit, buku, nilai buku, kriteria, dan perhitungan metode VIKOR
Karyawan Rumah Sakit UKI	Karyawan merupakan orang yang memiliki keterkaitan tentang pekerjaan yang berada di tempat tersebut (Rumah Sakit UKI). Karyawan melihat data buku dan melihat data buku berdasarkan rekomendasi. Selain itu, karyawan juga dapat menilai hasil rekomendasi data buku.

4.1.6. Skenario *Use case*

Berikut merupakan skenario yang akan dilakukan untuk setiap *use case*:

Tabel 4.1.6.1. Tabel Skenario – mengelola data user

Mengelola data user	
Tujuan	Menampilkan, mengubah, dan menghapus data user
Aktor	Admin
Kondisi awal	1. Login sebagai admin
Skenario utama	1. Admin memilih data user 2. Sistem menampilkan data user
Skenario alternatif	1. Jika admin ingin mengubah data user karyawan dengan menekan tombol “edit” 2. Jika admin ingin menghapus data user karyawan dengan menekan tombol “delete”
Kondisi akhir	Sistem menampilkan data user sesuai yang tersimpan dalam <i>database</i>

Tabel 4.1.6.2. Tabel Skenario – mengelola data nakes

Mengelola data nakes	
Tujuan	Menampilkan, menambah, mengubah, dan menghapus data nakes
Aktor	Admin
Kondisi awal	1. Login sebagai admin
Skenario utama	1. Admin memilih data nakes 2. Admin mengisi data nakes 3. Sistem menampilkan data nakes
Skenario alternatif	1. Jika admin ingin menambah data nakes dengan mengisi form nakes dan menekan tombol “add” 2. Jika admin ingin mengubah data nakes dengan menekan tombol “edit” 3. Jika admin ingin menghapus data nakes dengan menekan tombol “delete”
Kondisi akhir	Sistem menampilkan data nakes sesuai yang tersimpan dalam <i>database</i>

Tabel 4.1.6.3. Tabel Skenario – mengelola data jenis buku

Mengelola data jenis buku	
Tujuan	Menampilkan, menambah, mengubah, dan menghapus data jenis buku
Aktor	Admin
Kondisi awal	1. Login sebagai admin
Skenario utama	1. Admin memilih data jenis buku 2. Admin mengisi data jenis buku 3. Sistem menampilkan data jenis buku
Skenario alternatif	1. Jika admin ingin menambah data jenis buku dengan mengisi form jenis buku dan menekan tombol “add” 2. Jika admin ingin mengubah data jenis buku dengan menekan tombol “edit” 3. Jika admin ingin menghapus data jenis buku dengan menekan tombol “delete”
Kondisi akhir	Sistem menampilkan data jenis buku sesuai yang tersimpan dalam <i>database</i>

Tabel 4.1.6.4. Tabel Skenario – mengelola data penulis

Mengelola data penulis	
Tujuan	Menampilkan, menambah, mengubah, dan menghapus data penulis
Aktor	Admin
Kondisi awal	1. Login sebagai admin
Skenario utama	1. Admin memilih data penulis 2. Admin mengisi data penulis 3. Sistem menampilkan data penulis
Skenario alternatif	1. Jika admin ingin menambah data penulis dengan mengisi form penulis dan menekan tombol “add” 2. Jika admin ingin mengubah data penulis dengan menekan tombol “edit” 3. Jika admin ingin menghapus data penulis dengan menekan tombol “delete”
Kondisi akhir	Sistem menampilkan data penulis sesuai yang tersimpan dalam <i>database</i>

Tabel 4.1.6.5. Tabel Skenario – mengelola data penerbit

Mengelola data penerbit	
Tujuan	Menampilkan, menambah, mengubah, dan menghapus data penerbit
Aktor	Admin
Kondisi awal	1. Login sebagai admin
Skenario utama	1. Admin memilih data penerbit 2. Admin mengisi data penerbit 3. Sistem menampilkan data penerbit
Skenario alternatif	1. Jika admin ingin menambah data penerbit dengan mengisi form penerbit dan menekan tombol “add” 2. Jika admin ingin mengubah data penerbit dengan menekan tombol “edit” 3. Jika admin ingin menghapus data penerbit dengan menekan tombol “delete”
Kondisi akhir	Sistem menampilkan data penerbit sesuai yang tersimpan dalam <i>database</i>

Tabel 4.1.6.6. Tabel Skenario – mengelola data buku

Mengelola data buku	
Tujuan	Menampilkan, menambah, mengubah, dan menghapus data buku
Aktor	Admin
Kondisi awal	2. Login sebagai admin
Skenario utama	1. Admin memilih data buku 2. Admin mengisi data buku 3. Sistem menampilkan data buku
Skenario alternatif	1. Jika admin ingin menambah data buku dengan mengisi form buku dan menekan tombol “add” 2. Jika admin ingin mengubah data buku dengan menekan tombol “edit” 3. Jika admin ingin menghapus data buku dengan menekan tombol “delete”
Kondisi akhir	Sistem menampilkan data buku sesuai yang tersimpan dalam <i>database</i>

Tabel 4.1.6.7. Tabel Skenario – mengelola data review

Mengelola data nakes	
Tujuan	Menampilkan dan menghapus data user
Aktor	Admin
Kondisi awal	1. Login sebagai admin
Skenario utama	1. Admin memilih data review 2. Sistem menampilkan data review
Skenario alternatif	1. Jika admin ingin menghapus data review dengan menekan tombol “delete”
Kondisi akhir	Sistem menampilkan data review sesuai yang tersimpan dalam <i>database</i>

Tabel 4.1.6.8. Tabel Skenario – mengelola data admin

Mengelola data admin	
Tujuan	Menampilkan data admin
Aktor	Admin
Kondisi awal	1. Login sebagai admin
Skenario utama	1. Admin memilih data nakes 2. Sistem menampilkan data nakes
Kondisi akhir	Sistem menampilkan data admin yang tersimpan dalam <i>database</i>

Tabel 4.1.6.9. Tabel Skenario – mengelola data nilai buku

Mengelola data nilai buku	
Tujuan	Menampilkan, menambah, mengubah, dan menghapus data user
Aktor	Admin
Kondisi awal	1. Login sebagai admin
Skenario utama	1. Admin memilih data nilai buku 2. Admin mengisi data nilai buku 3. Sistem menampilkan data nilai buku
Skenario alternatif	1. Jika admin ingin menambah data nilai buku dengan mengisi form nilai buku dan menekan tombol “add” 2. Jika admin ingin mengubah data nilai buku dengan menekan tombol “edit” 3. Jika admin ingin menghapus data nilai buku dengan menekan tombol “delete”
Kondisi akhir	Sistem menampilkan data nilai buku sesuai yang tersimpan dalam <i>database</i>

Tabel 4.1.6.10. Tabel Skenario – melihat data buku

Melihat data buku	
Tujuan	Menampilkan data buku
Aktor	User
Kondisi awal	1. Login sebagai user
Skenario utama	1. User memilih data buku 2. Sistem menampilkan data nakes
Skenario alternatif	1. Jika user ingin melihat menu buku dengan menekan tombol “data buku” 2. Jika user ingin melihat menu rekomendasi buku dengan menekan tombol “data rekomendasi buku” 3. Jika user ingin melihat menu buku berdasarkan nakes dengan menekan tombol “data buku tenaga kesehatan” 4. Jika user ingin melihat menu buku berdasarkan tahun terbit dengan menekan tombol “data buku berdasarkan tahun terbit” 5. Jika user ingin melihat menu buku berdasarkan urutan buku dengan menekan tombol “data urutan buku”
Kondisi akhir	Sistem menampilkan data buku sesuai yang tersimpan dalam <i>database</i>

Tabel 4.1.6.11. Tabel Skenario – mengakses isi buku

Mengakses isi buku	
Tujuan	Menampilkan, mengakses, mengunduh, dan menilai buku
Aktor	User
Kondisi awal	1. Login sebagai user
Skenario utama	<ol style="list-style-type: none"> 1. User memilih detail buku setelah menekan buku yang ingin dibaca 2. User mengisi nilai buku berdasarkan kriteria 3. Sistem menampilkan data detail buku
Skenario alternatif	<ol style="list-style-type: none"> 1. Jika user ingin menilai buku dengan mengisi form penilaian buku berdasarkan kriteria berbentuk combo box 2. Jika admin ingin mengakses buku dengan menampilkan detail buku 3. Jika admin ingin mengunduh buku dengan menekan tombol “download”
Kondisi akhir	Sistem menampilkan data buku berdasarkan id sesuai yang tersimpan dalam <i>database</i>

Tabel 4.1.6.12. Tabel Skenario – mengisi ulasan

Mengisi ulasan	
Tujuan	Menambah data ulasan
Aktor	User
Kondisi awal	1. Login sebagai user
Skenario utama	1. User mengisi data ulasan
Skenario alternatif	1. Jika admin ingin menambah data ulasan dengan mengisi form ulasan dan menekan tombol “add”
Kondisi akhir	Sistem menampilkan data ulasan sesuai yang tersimpan dalam <i>database</i>

4.1.7. Spesifikasi use case

Deskripsi dari *use case* diagram yang menjelaskan bagaimana aktor dan *use case* terlibat dalam sistem.

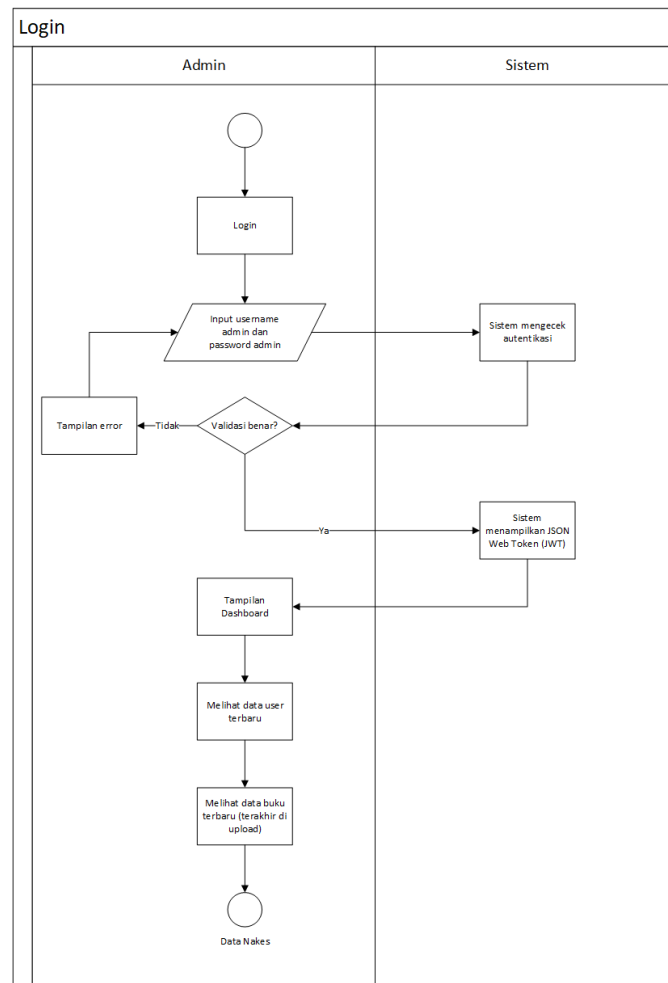
Tabel 4.1.7.1. Tabel Spesifikasi use case

No	Aktor	<i>Use Case</i>	Keterangan
1	Admin	Mengelola data user	Admin dapat melihat dan mengubah data user
2	Admin	Mengelola data nakes	Admin dapat melihat, menambah, mengubah, dan menghapus data nakes
3	Admin	Mengelola data jenis buku	Admin dapat melihat, menambah, mengubah, dan menghapus data jenis buku
4	Admin	Mengelola data penulis	Admin dapat melihat, menambah, mengubah, dan menghapus data penulis
5	Admin	Mengelola data penerbit	Admin dapat melihat, menambah, mengubah, menghapus data penerbit
6	Admin	Mengelola data buku	Admin dapat melihat, menambah, mengubah, dan menghapus data buku
7	Admin	Mengelola data nilai buku	Admin dapat melihat, menambah, mengubah, dan menghapus data nilai buku
8	Admin	Mengelola data review	Admin dapat melihat dan menghapus data review

9	Admin	Mengelola data admin	Admin dapat melihat dan menghapus data admin
10	User	Melihat data buku	User dapat melihat data buku
11	User	Mengakses isi buku	User dapat mengakses isi buku berdasarkan id buku tersebut
12	User	Mengisi nilai buku	User dapat mengisi nilai buku berdasarkan kriteria
13	User	Mengisi ulasan buku	User dapat mengisi ulasan buku

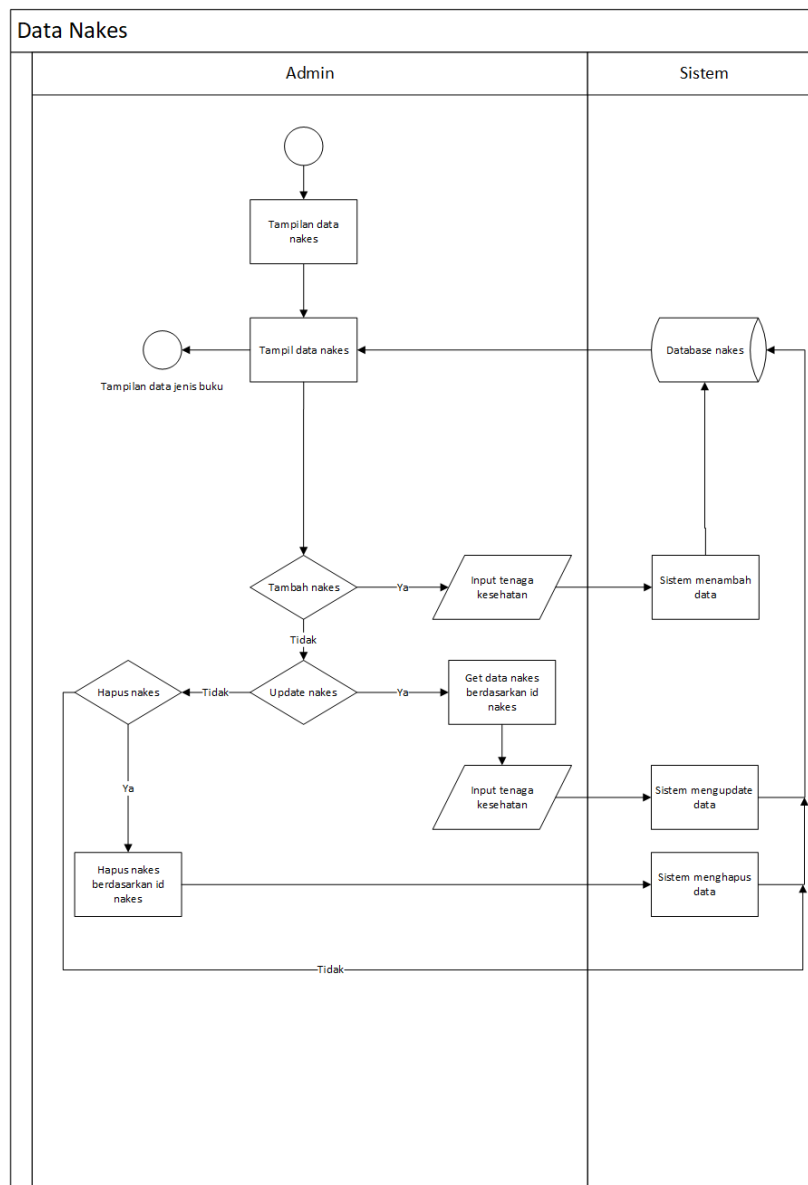
4.1.8. Activity Diagram

Activity Diagram digunakan untuk menjelaskan aktivitas apa saja yang dapat dilakukan dalam aplikasi. Kebutuhan yang telah diidentifikasi sebelumnya, dirancang beberapa activity diagram untuk menjelaskan alur proses dari kegiatan antara aktor dengan sistem, berikut proses yang akan berjalan yaitu:



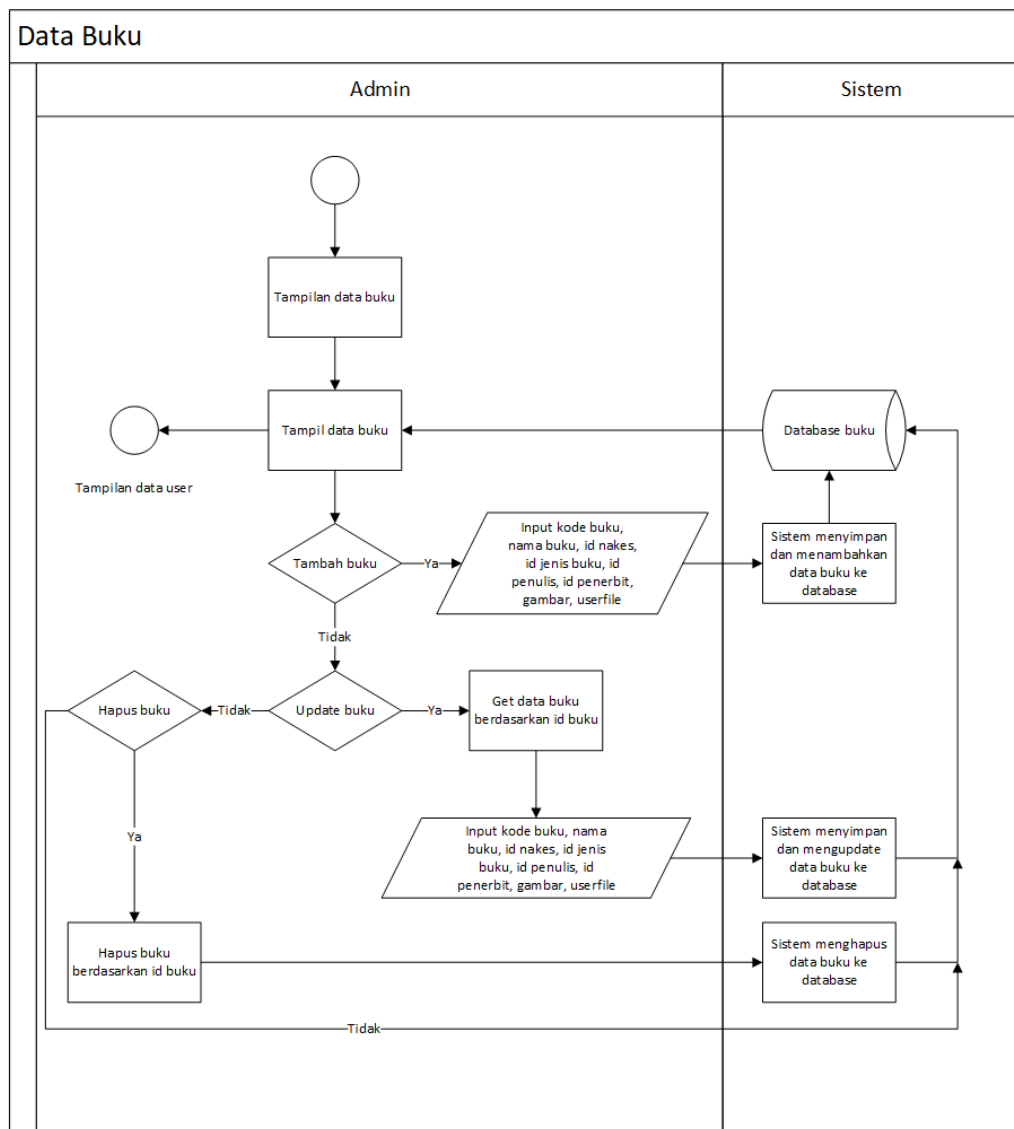
Gambar 4.1.8.1. Gambar Activity Diagram login admin

Gambar 4.1.8.1 merupakan activity diagram login yaitu admin memasukan username dan password untuk melakukan login sistem. Sistem akan melakukan validasi. Jika sukses, akan di arahkan ke dalam aplikasi



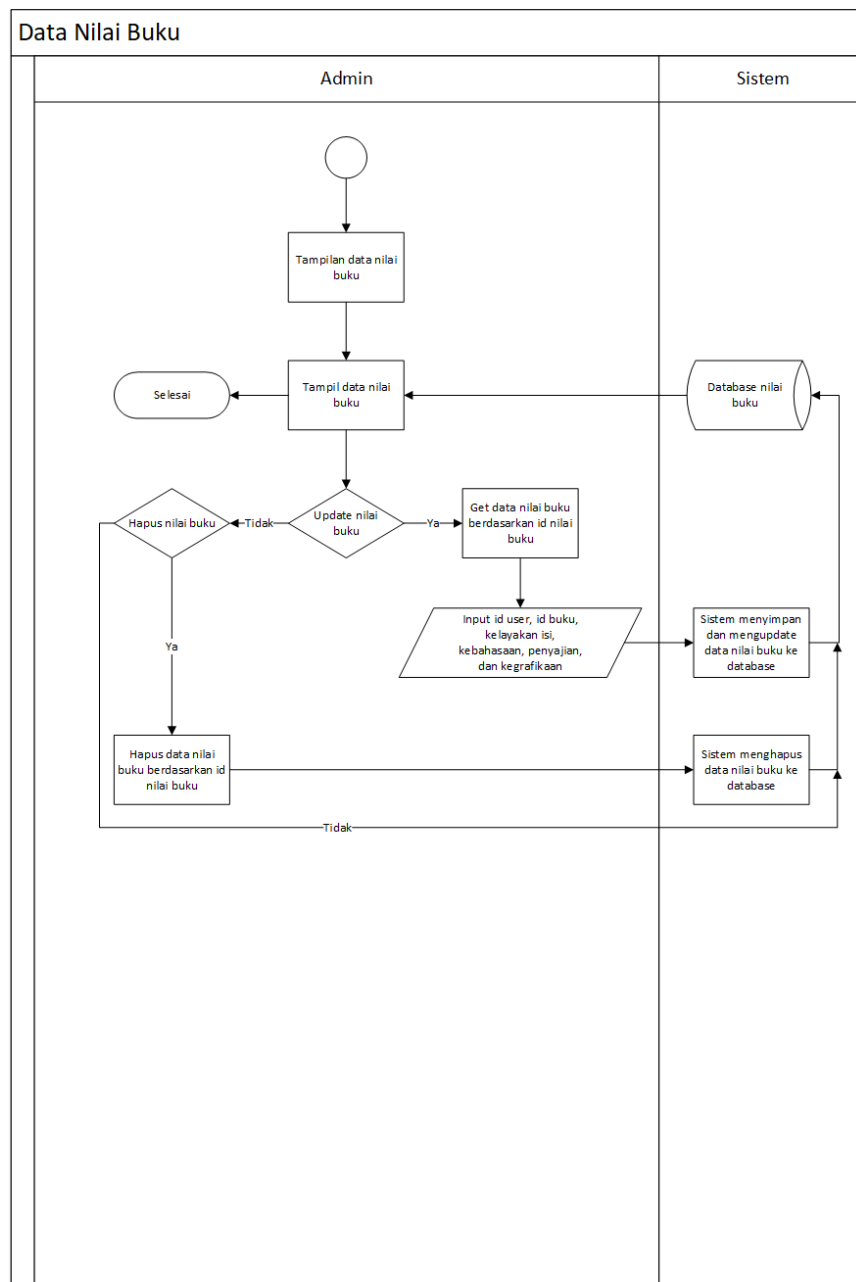
Gambar 4.1.8.2. Gambar Activity Diagram data nakes

Gambar 4.1.8.2. merupakan Activity Diagram untuk mengelola data nakes yang dimana admin dapat menambahkan, mengubah, dan menghapus data nakes yang tersimpan pada database



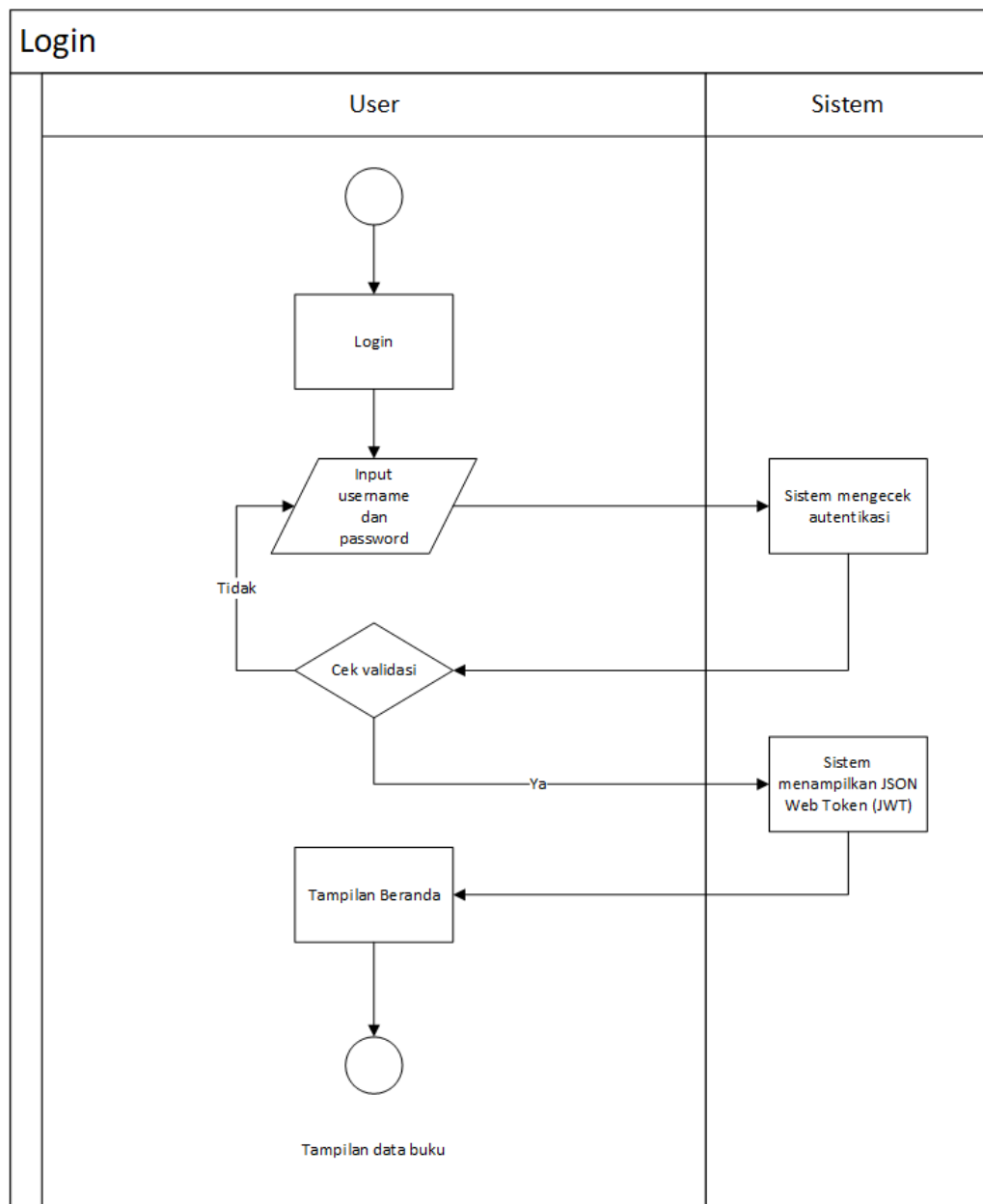
Gambar 4.1.8.3. Gambar Activity Diagram mengelola data buku

Gambar 4.1.8.3 merupakan activity diagram dalam mengelola data buku yang dimana admin dapat menambah, mengunggah gambar, mengunggah file, mengubah, dan menghapus data buku yang tersimpan pada database



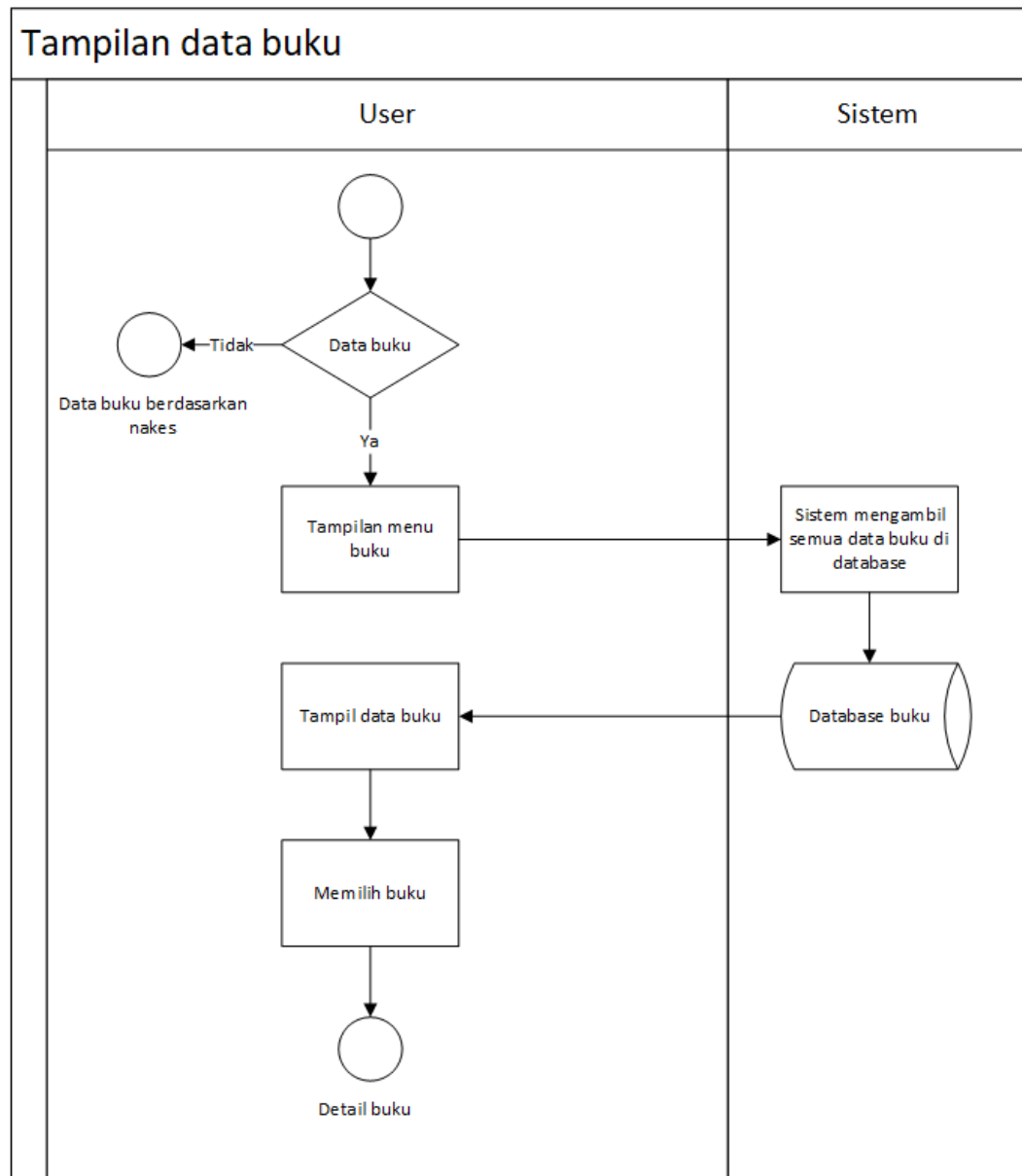
Gambar 4.1.8.4. Gambar Activity Diagram mengelola nilai buku

Gambar 4.1.8.4 merupakan activity diagram dalam mengelola nilai buku, admin dapat menambah data nilai pada kriteria (kelayakan isi, kebahasaan, penyajian, dan kegrafikaan), mengubah, dan menghapus data nilai buku yang tersimpan pada database.



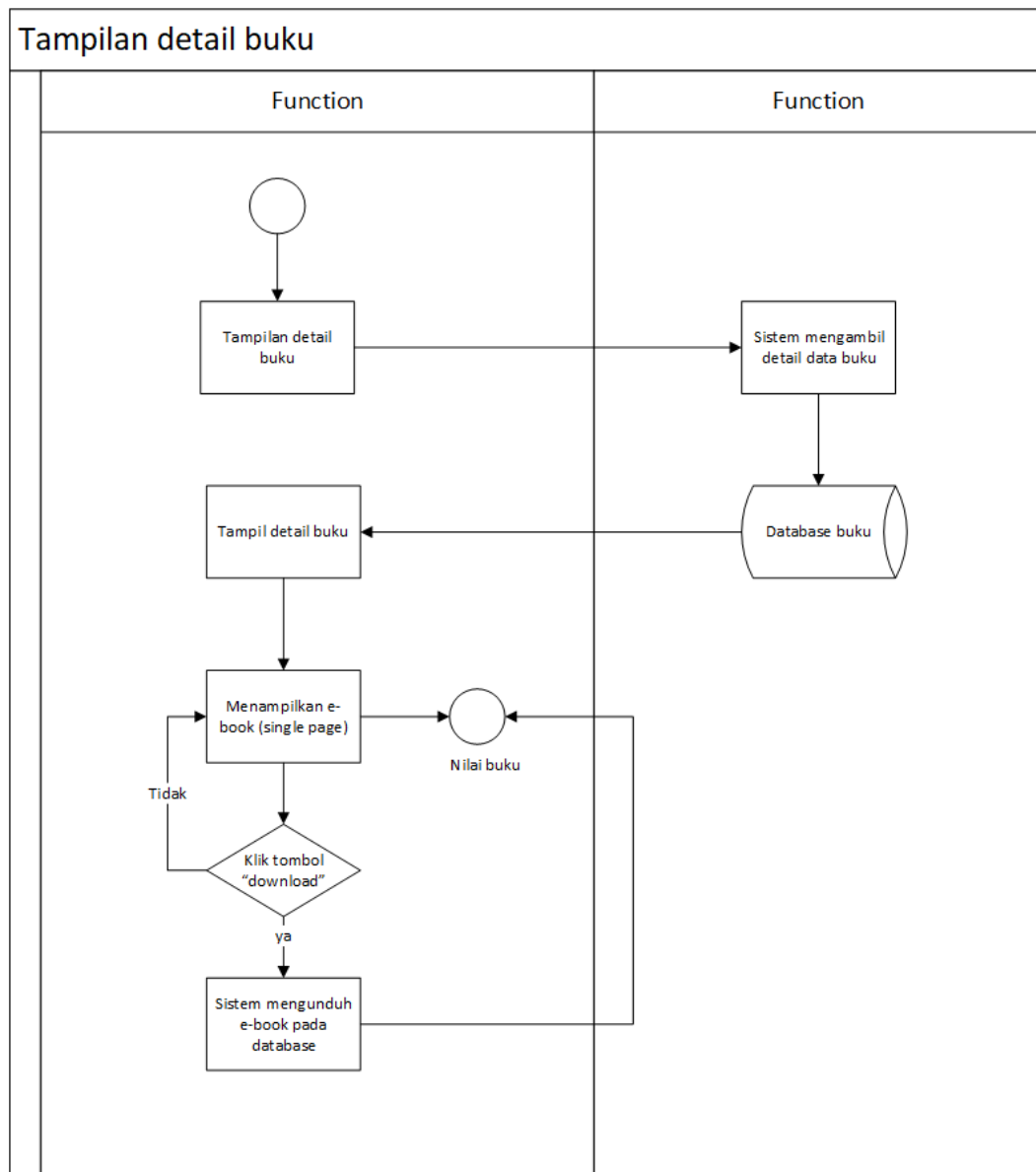
Gambar 4.1.8.5. Gambar Activity Diagram login user

Gambar 4.1.8.5 merupakan activity diagram login yaitu user memasukan username dan password untuk melakukan login sistem. Sistem akan melakukan validasi. Jika sukses, akan di arahkan ke dalam aplikasi



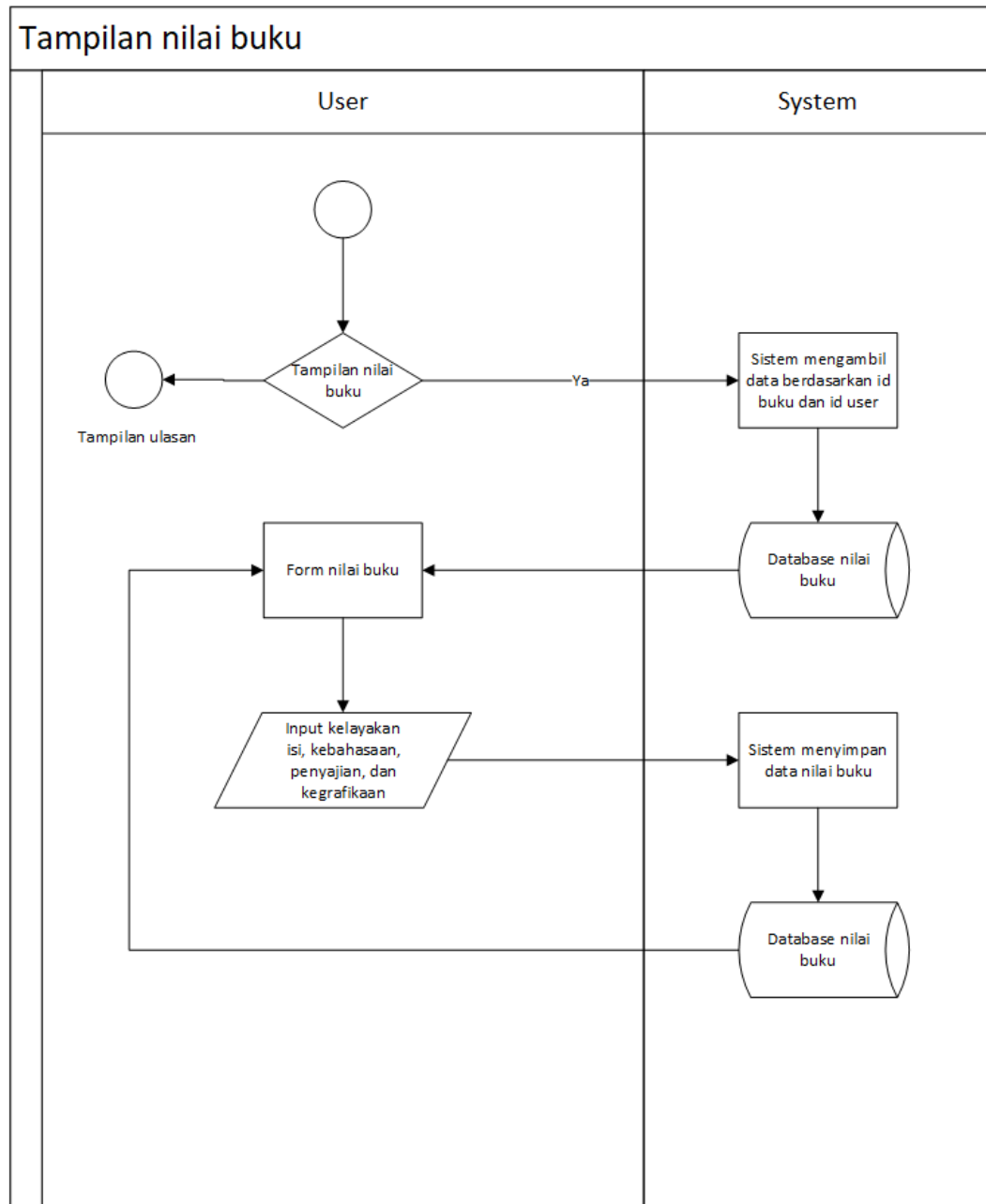
Gambar 4.1.8.6. Gambar Activity Diagram melihat data buku

Gambar 4.1.8.6 merupakan activity diagram melihat data buku yaitu user memilih judul buku yang diinginkan. Jika user sudah memilih, sistem akan menampilkan detail buku yang terdapat pada gambar 4.1.8.7.



Gambar 4.1.8.7. Gambar Activity Diagram melihat detail buku

Gambar 4.1.8.7 merupakan activity diagram melihat detail buku yaitu user melihat detail buku tersebut berdasarkan data yang ada pada database. User dapat mengunduh buku dengan menekan tombol “download”.



Gambar 4.1.8.8. Gambar Activity Diagram mengisi nilai buku

Gambar 4.1.8.8 merupakan activity diagram mengisi nilai buku yaitu user mengisi nilai buku berdasarkan kriteria (kelayakan isi, kebahasaan, penyajian, dan kegrafikaan). Nilai tersebut akan di hitung dengan metode VIKOR dan hasil ranking metode VIKOR tersebut akan ditampilkan di menu rekomendasi buku.

Tabel 4.2.1.3. Tabel Ulasan

Nama Atribut	Tipe Data	Keterangan
Id_ulasan	Int(11)	Not null, primary key
Nama_depan	Varchar(100)	Not null
Nama_belakang	Varchar(100)	Not null
email	Varchar(100)	Not null
Judul	Varchar(100)	Not null
pesan	Varchar(255)	Not null

Tabel 4.2.1.4. Tabel Jenis Buku

Nama Atribut	Tipe Data	Keterangan
Id_jenis_buku	Int(11)	Not null, primary key
Jenis_buku	Varchar(100)	Not null

Tabel 4.2.1.5. Tabel Tenaga Kesehatan

Nama Atribut	Tipe Data	Keterangan
Id_nakes	Int(11)	Not null, primary key
Tenaga_kesehatan	Varchar(100)	Not null

Tabel 4.2.1.6. Tabel Penerbit

Nama Atribut	Tipe Data	Keterangan
Id_penerbit	Int(11)	Not null, primary key
Nama_penerbit	Varchar(100)	Not null
Tahun_terbit	Varchar(100)	Not null

Tabel 4.2.1.7. Tabel Penulis

Nama Atribut	Tipe Data	Keterangan
Id_penulis	Int(11)	Not null, primary key
Nama_penulis	Varchar(100)	Not null

Tabel 4.2.1.8. Tabel Buku

Nama Atribut	Tipe Data	Keterangan
Id_buku	Int(11)	Not null, primary key
Kode_buku	Varchar(100)	Not null
Nama_buku	Varchar(255)	Not null
Id_jenis	Int(11)	Not null, foreign key
Id_nakes	Int(11)	Not null, foreign key
Id_penerbit	Int(11)	Not null, foreign key
Id_penulis	Int(11)	Not null, foreign key

Gambar	Varchar(255)	Not null
userfile	Varchar(255)	Not null

Tabel 4.2.1.9. Tabel Kriteria

Nama Atribut	Tipe Data	Keterangan
Id_kriteria	Int(11)	Not null, primary key
Kode	Varchar(100)	Not null
Nama_kriteria	Varchar(100)	Not null
bobot	Float	Not null
Tipe kriteria	Varchar(100)	Not null

Tabel 4.2.1.10. Tabel Nilai Buku

Nama Atribut	Tipe Data	Keterangan
Id_nilai_buku	Int(11)	Not null, primary key
Id_user	Int(11)	Not null, foreign key
Id_buku	Int(11)	Not null, foreign key
Kelayakan_isi	Float	Not null
Kebahasaan	Float	Not null
Penyajian	Float	Not null
kegrafikaan	Float	Not null

4.2.2. Perancangan Antar Muka

Perancangan *user interface* dijelaskan dalam bentuk tampilan mockup meliputi halaman admin dan user (karyawan)

4.2.2.1. Halaman Admin

a. tampilan login

Admin Digital Repository Rumah Sakit Umum Universitas Kristen Indonesia

https://localhost:3001/login

Username

Password

Login

Gambar 4.2.2.1.1. Gambar tampilan login

b. tampilan dashboard

Admin Digital Repository Rumah Sakit Umum Universitas Kristen Indonesia

https://localhost:3001/dashboard

Dashboard

Log out

Home/Dashboard

Tabel Admin

Tabel User

Tabel Buku

Tabel Nilai Buku

Grafik Babot

Grafik Nilai buku

Tabel Buku Terbaru

No	Kode Buku	Nama Buku	Jenis Buku	Tenaga Kesehatan	Penerbit	Tahun	Penulis	Gambar	Userfile
2	B2	Buku 2	Jenis B	Nakes B	Penerbit B	2018	Penulis B	Gambar-2.jpg	Userfile-2.pdf
1	B1	Buku 1	Jenis A	Nakes A	Penerbit A	2017	Penulis A	Gambar-1.jpg	Userfile-1.pdf

Tabel Nilai Buku Terbaru

No	user	Buku	Kelayakan Isi	Kebahasaan	Penyajian	Kegrafikaan
2	user 2	Buku 2	5	8	7	4
1	user 1	Buku 1	5	7	9	7

Gambar 4.2.2.1.2. Gambar tampilan dashboard

c. tampilan data user

Admin Digital Repository Rumah Sakit Umum Universitas Kristen Indonesia

https://localhost:3001/data-user

Dashboard Log out

Home/User Table

Form User

NIK

Nama

Username

Email

Password

Alamat

Nomor Telepon

Update

Tabel User

No	NIK	Nama User	Username User	Email User	Password User	Alamat	Telp	Update	Delete
1	NIK-1	User-1	Username-1	Email-1	asnlisdzkzndmzm	Alamat-1	Telp-1	Edit	Delete

Gambar 4.2.2.1.3. Gambar tampilan data user

d. tampilan data admin

Admin Digital Repository Rumah Sakit Umum Universitas Kristen Indonesia

https://localhost:3001/data-admin

Dashboard Log out

Home/Admin Table

Form Admin

Nama

Username

Email

Password

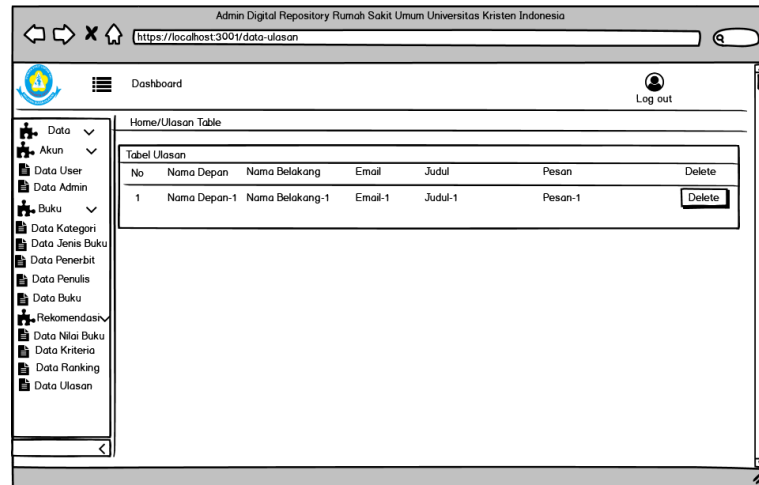
Add

Tabel Admin

No	Nama Admin	Username Admin	Email Admin	Password Admin	Update	Delete
1	Admin-1	Username-1	Email-1	asnlisdzndrikisa	Edit	Delete

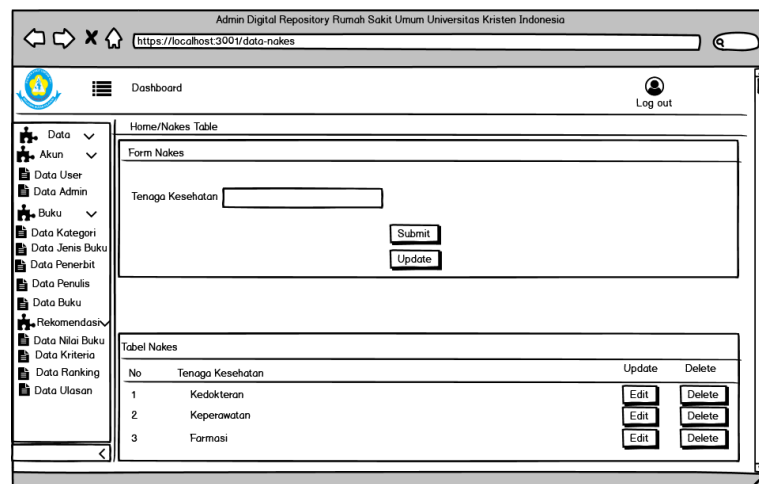
Gambar 4.2.2.1.4. Gambar tampilan data admin

e. tampilan data ulasan



Gambar 4.2.2.1.5. Gambar tampilan data ulasan

f. tampilan data nakes



Gambar 4.2.2.1.6. Gambar tampilan data nakes

g. tampilan data jenis buku

Admin Digital Repository Rumah Sakit Umum Universitas Kristen Indonesia

Dashboard Log out

Home/Jenis Buku Table

Form Jenis Buku

Jenis Buku

Submit Update

Tabel Jenis Buku

No	Jenis Buku	Update	Delete
1	Psychology	Edit	Delete
2	Anatomy	Edit	Delete
3	Drugs	Edit	Delete

Gambar 4.2.2.1.7. Gambar tampilan data jenis buku

h. tampilan data penerbit

Admin Digital Repository Rumah Sakit Umum Universitas Kristen Indonesia

Dashboard Log out

Home/Penerbit Table

Form Penerbit

Nama Penerbit

Tahun Terbit

Submit Update

Tabel Penerbit

No	Nama Penerbit	Tahun Terbit	Update	Delete
1	Erlangga	2021	Edit	Delete
2	Yudhistira	2018	Edit	Delete
3	Simar Dunia	2018	Edit	Delete

Gambar 4.2.2.1.8. Gambar tampilan data penerbit

i. tampilan data penulis

Admin Digital Repository Rumah Sakit Umum Universitas Kristen Indonesia

Dashboard

Log out

Home/Penulis Table

Form Penulis

Nama Penulis

Submit

Update

Tabel Penulis

No	Nama Penulis	Update	Delete
1	Raditya Dika	<input type="button" value="Edit"/>	<input type="button" value="Delete"/>
2	Soyid Wahyu	<input type="button" value="Edit"/>	<input type="button" value="Delete"/>
3	Wahyu Dharma	<input type="button" value="Edit"/>	<input type="button" value="Delete"/>

Gambar 4.2.2.1.9. Gambar tampilan data penulis

j. tampilan data buku

Admin Digital Repository Rumah Sakit Umum Universitas Kristen Indonesia

Dashboard

Log out

Home/Buku Table

Form Buku

Kode Buku

Nama Buku

Jenis Buku

Tenaga Kesehatan

Penerbit

Penulis

Gambar

Userfile

Submit

Update

Tabel Buku

No	Kode Buku	Nama Buku	Jenis Buku	Tenaga Kesehatan	Penerbit	Tahun	Penulis	Gambar	Userfile	Update	Delete
1	B1	Buku 1	Jenis A	Nakes A	Penerbit A	2017	Penulis A	Gambar-1.jpg	Userfile-1.pdf	<input type="button" value="Edit"/>	<input type="button" value="Delete"/>

Gambar 4.2.2.1.10. Gambar tampilan data buku

k. tampilan data kriteria

Admin Digital Repository Rumah Sakit Umum Universitas Kristen Indonesia

https://localhost:3001/data-kriteria

Dashboard Log out

Home/Kriteria Table

Form Kriteria

Kode Buku

Nama Kriteria

Bobot Buku

Tipe Kriteria

Grafik Kriteria Buku

Tabel Kriteria

No	Kode	Nama Kriteria	Bobot	Tipe Kriteria
1	C1	Kelayakan isi	0.3	benefit
2	C2	Kebahasaan	0.25	benefit
3	C3	Penyajian	0.28	benefit
4	C4	Kegrafikaan	0.17	benefit

Gambar 4.2.2.1.11. Gambar tampilan data kriteria

l. tampilan data nilai buku

Admin Digital Repository Rumah Sakit Umum Universitas Kristen Indonesia

https://localhost:3001/data-nilai-buku

Dashboard Log out

Home/Nilai Buku Table

Form Nilai Buku

User

Buku

Kelayakan isi

Kebahasaan

Penyajian

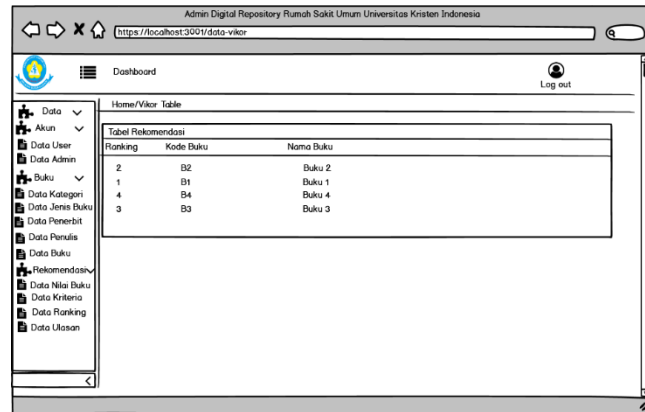
Kegrafikaan

Tabel Nilai Buku

No	user	Buku	Kelayakan Isi	Kebahasaan	Penyajian	Kegrafikaan	Update	Delete
1	User 1	Buku 1	5	7	9	7	<input type="button" value="Edit"/>	<input type="button" value="Delete"/>

Gambar 4.2.2.1.12. Gambar tampilan data nilai buku

m. tampilan data Rekomendasi



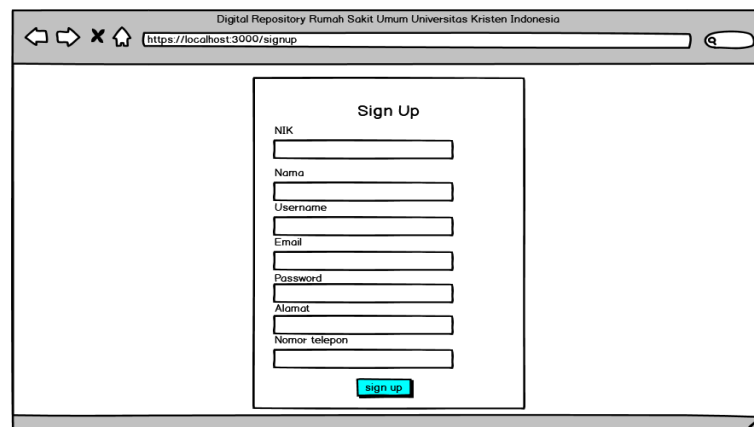
The screenshot shows a web application interface for an Admin Digital Repository. The browser address bar displays 'https://localhost:3001/data-viktor'. The page title is 'Admin Digital Repository Rumah Sakit Umum Universitas Kristen Indonesia'. The dashboard includes a sidebar menu with options like 'Data', 'Akun', 'Data User', 'Data Admin', 'Buku', 'Data Kategori', 'Data Jenis Buku', 'Data Penerbit', 'Data Penulis', 'Data Buku', 'Rekomendasi', 'Data Nilai Buku', 'Data Kriteria', 'Data Ranking', and 'Data Ulasan'. The main content area is titled 'Home/Visitor Table' and contains a table with the following data:

Ranking	Kode Buku	Nama Buku
2	B2	Buku 2
1	B1	Buku 1
4	B4	Buku 4
3	B3	Buku 3

Gambar 4.2.2.1.13. Gambar tampilan data Rekomendasi

4.2.2.2. Halaman user (karyawan)

- Tampilan Register



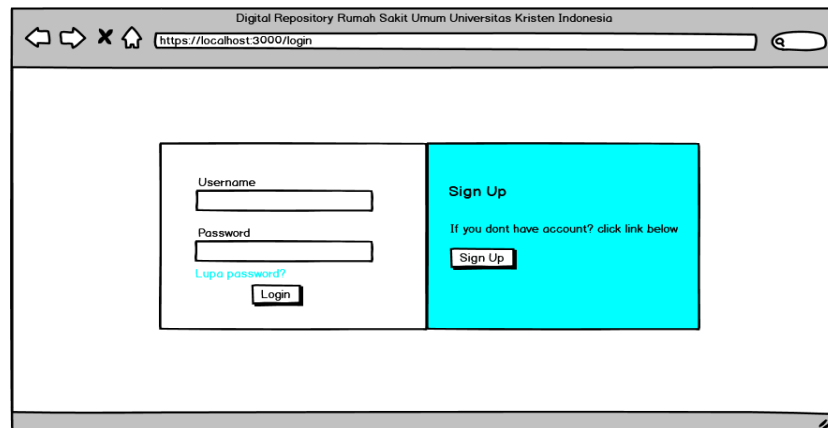
The screenshot shows a web application interface for a user registration page. The browser address bar displays 'https://localhost:3000/signup'. The page title is 'Digital Repository Rumah Sakit Umum Universitas Kristen Indonesia'. The form is titled 'Sign Up' and contains the following input fields:

- NIK
- Nama
- Username
- Email
- Password
- Alamat
- Nomor telepon

Below the input fields is a blue button labeled 'sign up'.

Gambar 4.2.2.2.1. Gambar tampilan register

- Tampilan Login



Digital Repository Rumah Sakit Umum Universitas Kristen Indonesia

<https://localhost:3000/login>

Username

Password

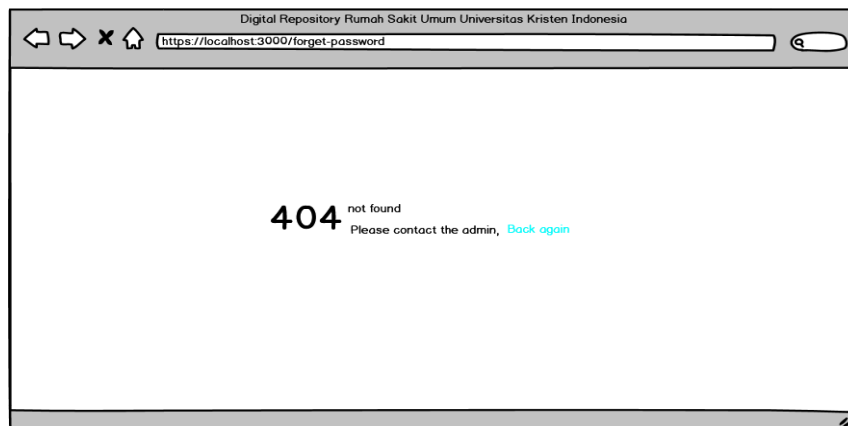
[Lupa password?](#)

Sign Up

If you dont have account? click link below

Gambar 4.2.2.2.2. Gambar tampilan login

- Tampilan lupa password



Digital Repository Rumah Sakit Umum Universitas Kristen Indonesia

<https://localhost:3000/forget-password>

404 not found
Please contact the admin, [Back again](#)

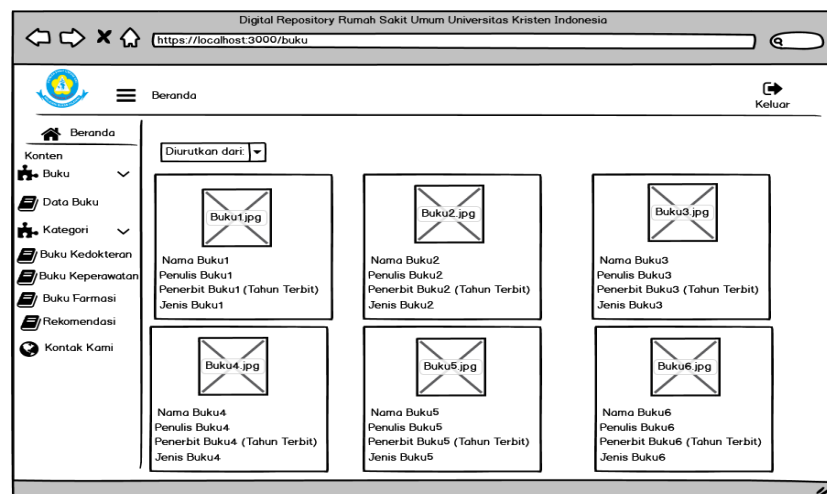
Gambar 4.2.2.2.3. Gambar tampilan lupa password

- Tampilan Beranda



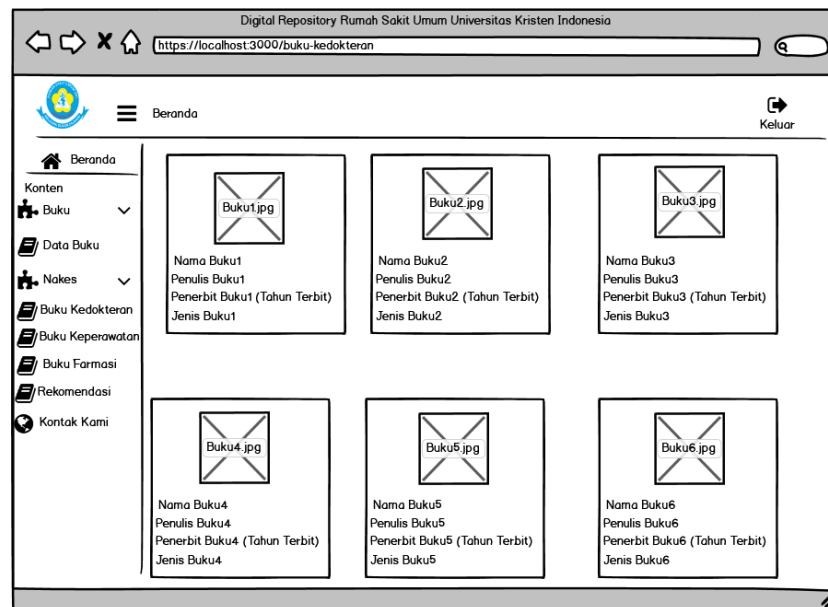
Gambar 4.2.2.2.4. Gambar tampilan beranda

- Tampilan data buku



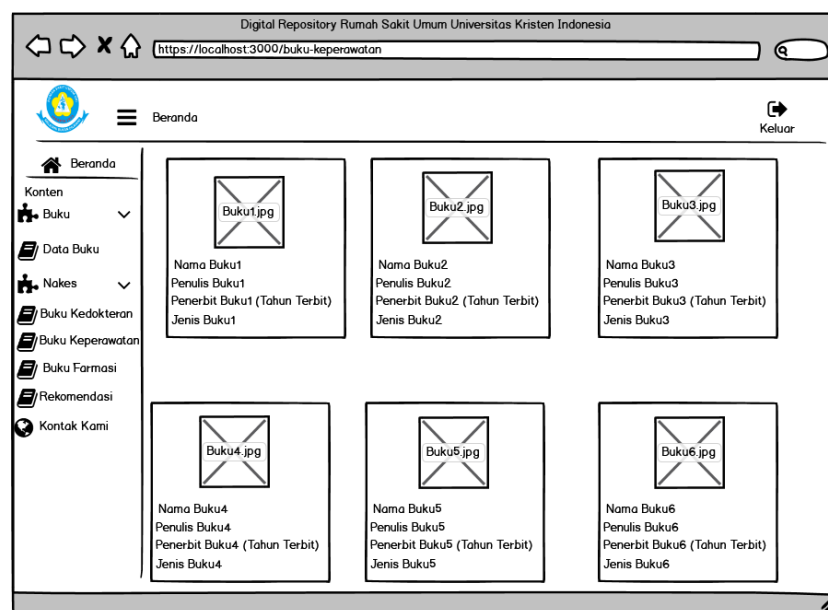
Gambar 4.2.2.2.5. Gambar tampilan data buku

- Tampilan data buku kedokteran



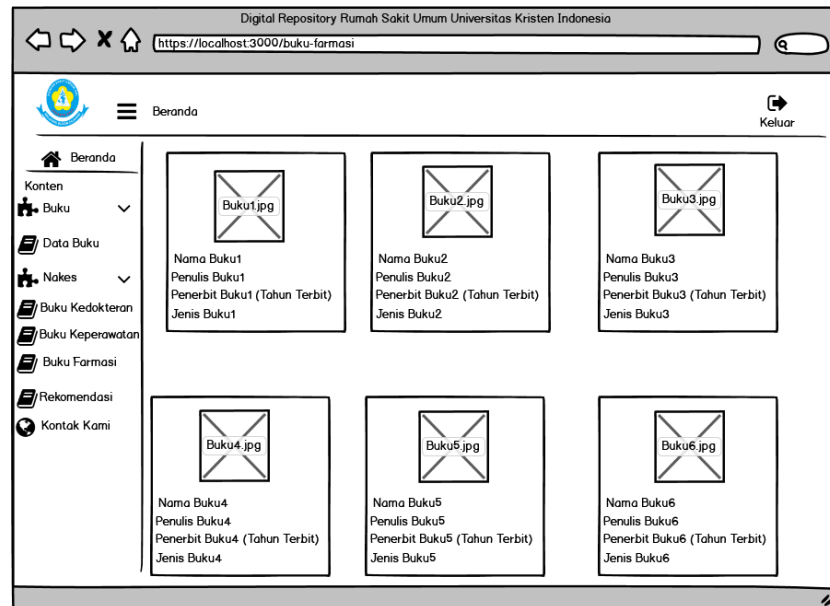
Gambar 4.2.2.2.6. Gambar tampilan data buku kedokteran

- Tampilan data buku keperawatan



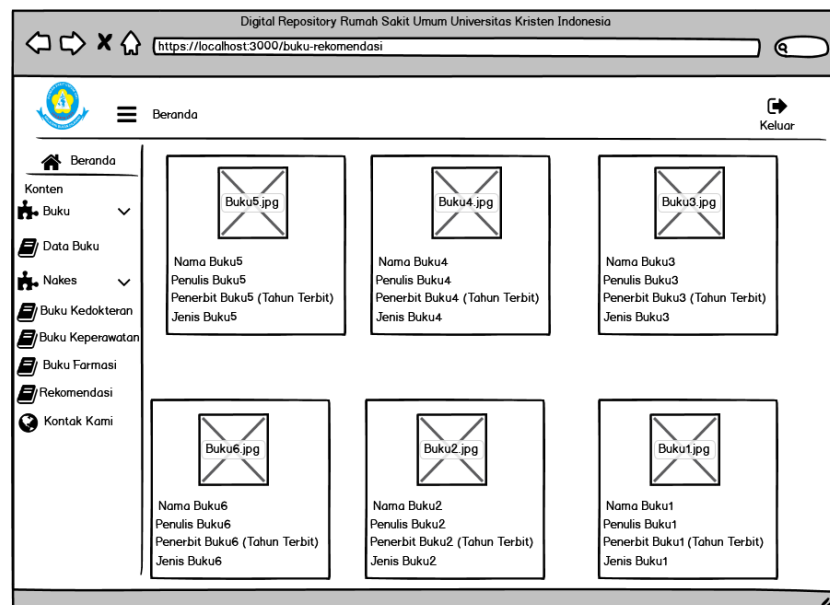
Gambar 4.2.2.2.7. Gambar tampilan data buku keperawatan

- Tampilan data buku farmasi



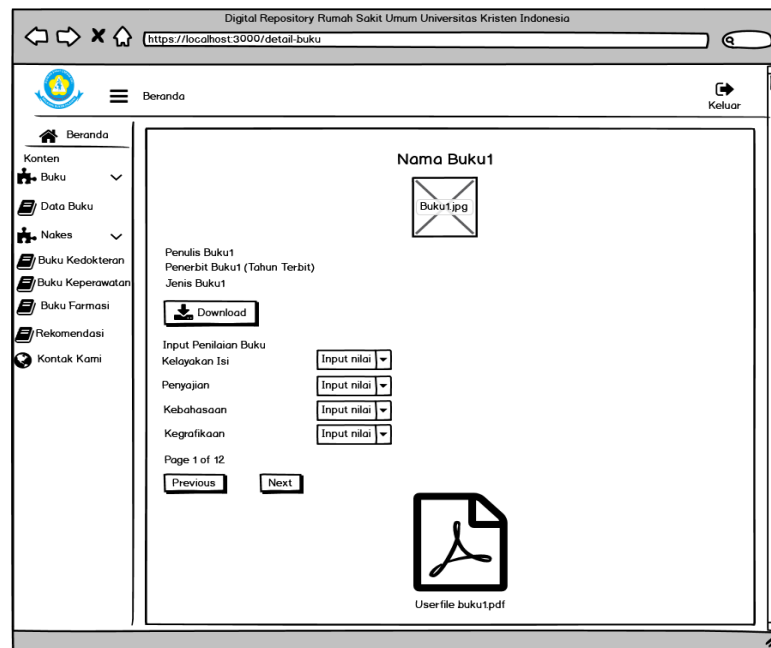
Gambar 4.2.2.2.8. Gambar tampilan data buku farmasi

- Tampilan data buku rekomendasi



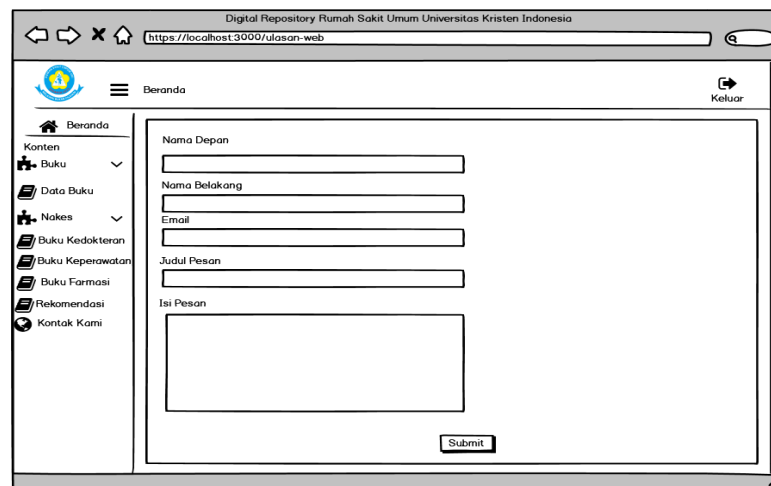
Gambar 4.2.2.2.9. Gambar tampilan data buku rekomendasi

- Tampilan detail buku



Gambar 4.2.2.2.10. Gambar tampilan detail buku

- Tampilan ulasan buku

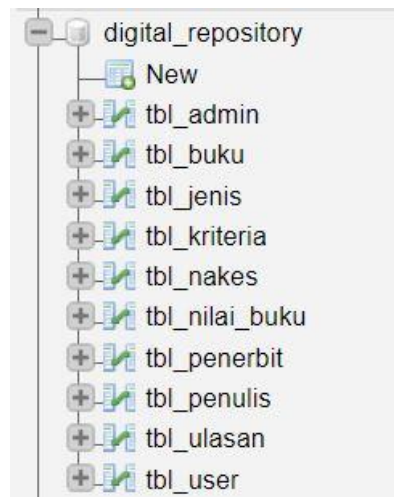


Gambar 4.2.2.2.11. Gambar tampilan ulasan buku

BAB V. IMPLEMENTASI DAN PENGUJIAN

5.1. Implementasi Basis Data

Berikut ini merupakan penjelasan mengenai implementasi *database* yang telah dilakukan pada pembuatan sistem aplikasi ini :




Gambar 5.1.1. Gambar Desain Basis Data

Pada Gambar 5.1.1 merupakan implementasi *database* yang digunakan untuk menyimpan data dan digunakan dalam pembuatan aplikasi. Tabel yang digunakan meliputi *tbl_admin*, *tbl_buku*, *tbl_jenis*, *tbl_kriteria*, *tbl_nakes*, *tbl_nilai_buku*, *tbl_penerbit*, *tbl_penulis*, *tbl_ulasan*, *tbl_user*. Berikut merupakan penjelasan setiap tabelnya:

#	Name	Type	Collation	Attributes	Null	Default
<input type="checkbox"/> 1	id_admin	int(11)			No	None
<input type="checkbox"/> 2	nama_admin	varchar(100)	latin1_swedish_ci		No	None
<input type="checkbox"/> 3	username_admin	varchar(100)	latin1_swedish_ci		No	None
<input type="checkbox"/> 4	email_admin	varchar(100)	latin1_swedish_ci		No	None
<input type="checkbox"/> 5	password_admin	varchar(100)	latin1_swedish_ci		No	None


Gambar 5.1.2. Gambar tabel admin

Gambar 5.1.2 merupakan tabel dalam menyimpan data admin. Tabel ini berisikan `id_admin`, `nama_admin`, `username_admin`, `email_admin`, `password_admin`.

	#	Name	Type	Collation	Attributes	Null	Default
<input type="checkbox"/>	1	id_buku 	int(11)			No	None
<input type="checkbox"/>	2	kode_buku	varchar(100)	latin1_swedish_ci		No	None
<input type="checkbox"/>	3	nama_buku	varchar(255)	latin1_swedish_ci		No	None
<input type="checkbox"/>	4	id_jenis	int(11)			No	None
<input type="checkbox"/>	5	id_nakes	int(11)			No	None
<input type="checkbox"/>	6	id_penerbit	int(11)			No	None
<input type="checkbox"/>	7	id_penulis	int(11)			No	None
<input type="checkbox"/>	8	gambar	varchar(255)	latin1_swedish_ci		No	None
<input type="checkbox"/>	9	userfile	varchar(255)	latin1_swedish_ci		No	None

Gambar 5.1.3. Gambar tabel buku

Gambar 5.1.3 merupakan tabel menyimpan data buku, tabel ini berisi `id_buku`, `kode_buku`, `nama_buku`, `id_jenis`, `id_nakes`, `id_penerbit`, `id_penulis`, `gambar`, dan `userfile`.

	#	Name	Type	Collation	Attributes	Null	Default
<input type="checkbox"/>	1	id_jenis 	int(11)			No	None
<input type="checkbox"/>	2	jenis_buku	varchar(100)	latin1_swedish_ci		No	None

Gambar 5.1.4. Gambar tabel jenis buku

Gambar 5.1.4 merupakan tabel untuk menyimpan data jenis buku. Tabel ini berisi `id_jenis` dan `jenis buku`.

	#	Name	Type	Collation	Attributes	Null	Default
<input type="checkbox"/>	1	id_kriteria	int(11)			No	None
<input type="checkbox"/>	2	kode	varchar(10)	latin1_swedish_ci		No	None
<input type="checkbox"/>	3	nama_kriteria	varchar(100)	latin1_swedish_ci		No	None
<input type="checkbox"/>	4	bobot	float			No	None
<input type="checkbox"/>	5	tipe_kriteria	varchar(100)	latin1_swedish_ci		No	None

Gambar 5.1.5. Gambar tabel kriteria

Gambar 5.1.5 merupakan tabel untuk menyimpan kriteria. Tabel ini berisi id_kriteria, kode, nama_kriteria, bobot, tipe_kriteria.

	#	Name	Type	Collation	Attributes	Null	Default
<input type="checkbox"/>	1	id_nakes	int(11)			No	None
<input type="checkbox"/>	2	tenaga_kesehatan	varchar(100)	latin1_swedish_ci		No	None

Gambar 5.1.6. Gambar tabel tenaga kesehatan

Gambar 5.1.6 merupakan tabel untuk menyimpan data tenaga kesehatan. Tabel ini berisi id_nakes dan tenaga_kesehatan.

	#	Name	Type	Collation	Attributes	Null	Default
<input type="checkbox"/>	1	id_nilai_buku	int(11)			No	None
<input type="checkbox"/>	2	id_user	int(11)			No	None
<input type="checkbox"/>	3	id_buku	int(11)			No	None
<input type="checkbox"/>	4	kelayakan_isi	varchar(10)	latin1_swedish_ci		No	None
<input type="checkbox"/>	5	kebahasaan	varchar(10)	latin1_swedish_ci		No	None
<input type="checkbox"/>	6	penyajian	varchar(10)	latin1_swedish_ci		No	None
<input type="checkbox"/>	7	kegrafikaan	varchar(10)	latin1_swedish_ci		No	None

Gambar 5.1.7. Gambar tabel nilai buku

Gambar 5.1.7 merupakan tabel untuk menyimpan data nilai buku yang dimana data nilai buku ini akan dihitung dengan menggunakan metode VIKOR dan hasilnya akan ditampilkan di menu rekomendasi buku. Tabel ini berisi id_nilai_buku, id_user, id_buku, kelayakan_isi, kebahasaan, penyajian, dan kegrafikaan.

	#	Name	Type	Collation	Attributes	Null	Default
<input type="checkbox"/>	1	id_penerbit	int(11)			No	None
<input type="checkbox"/>	2	nama_penerbit	varchar(100)	latin1_swedish_ci		No	None
<input type="checkbox"/>	3	tahun_terbit	varchar(5)	latin1_swedish_ci		No	None

Gambar 5.1.8. Gambar tabel penerbit

Gambar 5.1.8 merupakan tabel untuk menyimpan data penerbit, tabel ini berisi id_penerbit, nama_penerbit, dan tahun_terbit.

	#	Name	Type	Collation	Attributes	Null	Default
<input type="checkbox"/>	1	id_penulis	int(11)			No	None
<input type="checkbox"/>	2	nama_penulis	varchar(100)	latin1_swedish_ci		No	None

Gambar 5.1.9. Gambar tabel penulis

Gambar 5.1.9 merupakan tabel untuk menyimpan data penulis, tabel ini berisi id_penulis dan nama_penulis.

	#	Name	Type	Collation	Attributes	Null	Default
<input type="checkbox"/>	1	id_ulasan	int(11)			No	None
<input type="checkbox"/>	2	nama_depan	varchar(100)	latin1_swedish_ci		No	None
<input type="checkbox"/>	3	nama_belakang	varchar(100)	latin1_swedish_ci		No	None
<input type="checkbox"/>	4	email	varchar(100)	latin1_swedish_ci		No	None
<input type="checkbox"/>	5	judul	varchar(100)	latin1_swedish_ci		No	None
<input type="checkbox"/>	6	pesan	varchar(255)	latin1_swedish_ci		No	None

Gambar 5.1.10. Gambar tabel ulasan

Gambar 5.1.10 merupakan tabel untuk menyimpan data ulasan, tabel ini berisi id_ulasan, nama_depan, nama_belakang, email, judul, dan pesan.

#	Name	Type	Collation	Attributes	Null	Default
<input type="checkbox"/> 1	id_user 	int(11)			No	None
<input type="checkbox"/> 2	nik	varchar(11)	latin1_swedish_ci		No	None
<input type="checkbox"/> 3	nama_user	varchar(100)	latin1_swedish_ci		No	None
<input type="checkbox"/> 4	username_user	varchar(100)	latin1_swedish_ci		No	None
<input type="checkbox"/> 5	email_user	varchar(100)	latin1_swedish_ci		No	None
<input type="checkbox"/> 6	password_user	varchar(100)	latin1_swedish_ci		No	None
<input type="checkbox"/> 7	id_jabatan	int(11)			No	None
<input type="checkbox"/> 8	alamat	varchar(100)	latin1_swedish_ci		No	None
<input type="checkbox"/> 9	notelp	varchar(12)	latin1_swedish_ci		No	None

Gambar 5.1.11. Gambar tabel user

Gambar 5.1.11 merupakan tabel untuk menyimpan data user, tabel ini berisi id_user, nik, nama_user, username_user, email_user, password_user, alamat, dan notelp.

5.2. Implementasi Pengkodean Sistem

Aplikasi ini dibuat dan dibangun dengan menggunakan Bahasa pemrograman Python dan Javascript dengan menggunakan *framework Flask dan Javascript*. Implementasi yang dilakukan sesuai dengan analisis yang telah dilakukan sebelumnya. Beberapa proses dilakukan berikut :

1. Membuat API untuk melakukan pengolahan data semua tabel pada database dengan menggunakan flask sebagai backend dan web server
2. Membuat perhitungan metode VIKOR dan diintegrasikan ke API supaya terhubung dengan database
3. Membuat tampilan website dengan menggunakan ReactJS sebagai frontend

Backend

1. Deklarasi Database

Deklarasi Database merupakan proses untuk menghubungkan backend dengan database sehingga pengolahan data menjadi mudah Proses menghubungkan database yaitu

```

from flask import Flask, render_template
from flask_mysqlldb import MySQL
from flask_bcrypt import Bcrypt
from flask_jwt_extended import JWTManager
from flask_cors import CORS

db = MySQL()
bcrypt = Bcrypt()
jwt = JWTManager()
# app = Flask()
app = Flask(__name__)

def create_app():
    cors = CORS(app, resources=r'/*')
    app.config['MYSQL_HOST'] = 'localhost'
    app.config['MYSQL_USER'] = 'root'
    app.config['MYSQL_PASSWORD'] = ''
    app.config['MYSQL_DB'] = 'digital_repository'
    app.config['MYSQL_CURSORCLASS'] = 'DictCursor'

    app.config['MAX_CONTENT_LENGTH'] = 350 * 1024 * 1024

    app.config["JWT_SECRET_KEY"] = 'secret'
    db.init_app(app)
    bcrypt.init_app(app)
    jwt.init_app(app)

```

2. Mengolah user

Mengolah user merupakan proses login dan register user yang terdapat pada database. Proses mengolah user dibagi menjadi beberapa potongan yaitu:

```

from flask import Flask, render_template, Blueprint
from flask import jsonify, request
from flask_jwt_extended import create_access_token
from ..run import db, jwt, bcrypt

user = Blueprint('user', __name__)

CREATE / REGISTER
@user.route('/user/register', methods=['POST'])
def add_user():
    cur = db.connection.cursor()
    nik = request.get_json()['nik']
    nama_user = request.get_json()['nama_user']
    username_user = request.get_json()['username_user']
    email_user = request.get_json()['email_user']
    password_user =
bcrypt.generate_password_hash(request.get_json()['password_user
']).decode('utf-8')
    alamat = request.get_json()['alamat']
    notelp = request.get_json()['notelp']

```

```

cur.execute("INSERT INTO tbl_user (id_user,nik, nama_user,
username_user,email_user,password_user, alamat, notelp) "
           " SELECT MAX(id_user)+1, "
           "'" + str(nik) + "', "
           "'" + str(nama_user) + "', "
           "'" + str(username_user) + "', "
           "'" + str(email_user) + "', "
           "'" + str(password_user) + "', "
           "'" + str(alamat) + "', "
           "'" + str(notelp) + "' " +
           "FROM tbl_user")

db.connection.commit()
result = \
    {
        'nik': nik,
        'nama_user': nama_user,
        'username_user': username_user,
        'email_user': email_user,
        'password_user': password_user,
        'alamat': alamat,
        'notelp': notelp
    }

return jsonify({"result": result})

```

```

# LOGIN
@user.route('/user/login', methods=['POST'])
def login():
    cur = db.connection.cursor()
    username_user = request.get_json()['username_user']
    password_user = request.get_json()['password_user']
    result = ""

    cur.execute("SELECT * FROM tbl_user where username_user = '"
+ str(username_user) + "'")
    rv = cur.fetchone()

    if bcrypt.check_password_hash(rv['password_user'],
password_user):
        access_token = create_access_token(
            identity={'id_user': rv['id_user'],
                    'nik': rv['nik'],
                    'nama_user': rv['nama_user'],
                    'username_user': rv['username_user'],
                    'email_user': rv['email_user'],
                    'alamat': rv['alamat'],
                    'notelp': rv['notelp']})
        result = jsonify({"result": access_token})
    else:
        result = jsonify({"error": "Invalid username and
password"})

    return result

```


3. Mengolah buku

Mengolah buku merupakan proses melihat, menambah, mengubah, dan menghapus data user yang terdapat pada database. Proses mengolah user dibagi menjadi beberapa potongan yaitu:

a. Proses tambah

Proses menambah buku digunakan untuk menambahkan kedalam database. Proses tersebut melakukan perintah *query insert*

```
# CREATE
@buku.route('/buku', methods=['POST'])
def add_buku():
    if request.method == 'POST':
        if app.config['MAX_CONTENT_LENGTH']:
            cur = db.connection.cursor()

            kode_buku = request.form['kode_buku']
            nama_buku = request.form['nama_buku']
            id_jenis = request.form['id_jenis']
            id_nakes = request.form['id_nakes']
            id_penerbit = request.form['id_penerbit']
            id_penulis = request.form['id_penulis']
            gambar = save_images(request.files['gambar'])
            userfile = save_userfile(request.files['userfile'])

            cur.execute(
                "INSERT INTO tbl_buku
                (id_buku,kode_buku,nama_buku,id_jenis,id_nakes,id_penerbit,
                id_penulis, gambar, userfile) "
                " SELECT MAX(id_buku)+1, "
                "'" + str(kode_buku) + "', "
                "'" + str(nama_buku) + "', "
                "'" + str(id_jenis) + "', "
                "'" + str(id_nakes) + "', "
                "'" + str(id_penerbit) + "', "
                "'" + str(id_penulis) + "', "
                "'" + str(gambar) + "', "
                "'" + str(userfile) + "' "
                "FROM tbl_buku")

            db.connection.commit()
            result = \
            {

                'kode_buku': kode_buku,
                'nama_buku' : nama_buku,
                'id_jenis' : id_jenis,
                'id_nakes': id_nakes,
                'id_penerbit' : id_penerbit,
                'id_penulis' : id_penulis,
                'gambar' : gambar,
                'userfile' : userfile
            }

            return jsonify({"result" : result})
```

b. Proses ubah

```

# UPDATE
@buku.route('/buku/<id_buku>', methods=['PUT'])
def update_buku(id_buku):
    if request.method == 'PUT':
        if app.config['MAX_CONTENT_LENGTH']:
            cur = db.connection.cursor()
            kode_buku = request.form['kode_buku']
            nama_buku = request.form['nama_buku']
            id_jenis = request.form['id_jenis']
            id_nakes = request.form['id_nakes']
            id_penerbit = request.form['id_penerbit']
            id_penulis = request.form['id_penulis']

            # gambar
            gambar = save_images(request.files['gambar'])

            # file
            userfile = save_userfile(request.files['userfile'])

            cur.execute("UPDATE tbl_buku SET kode_buku = '" +
str(kode_buku) + "',"
                        + "nama_buku ='" + str(nama_buku) +
",',"
                        + "id_jenis ='" + str(id_jenis) + "',"
                        + "id_nakes ='" + str(id_nakes) + "',"
                        + "id_penerbit ='" + str(id_penerbit)
+ "',"
                        + "id_penulis ='" +
str(id_penulis) + "',"
                        + "gambar ='" + str(gambar) + "',"
                        + "userfile ='" + str(userfile)
                        + "'WHERE id_buku=" + id_buku)

            db.connection.commit()
            result = \
                {
                    'id_buku' : id_buku,
                    'kode_buku': kode_buku,
                    'nama_buku' : nama_buku,
                    'id_jenis' : id_jenis,
                    'id_nakes': id_nakes,
                    'id_penerbit' : id_penerbit,
                    'id_penulis' : id_penulis,
                    'gambar' : gambar,
                    'userfile' : userfile
                }

            return jsonify({"result": result})

            + "id_penulis ='" + str(id_penulis) +
",',"
            + "gambar ='" + str(gambar) + "',"
            + "userfile ='" + str(userfile)
            + "'WHERE id_buku=" + id_buku)

            db.connection.commit()

```

```

result = \
    {
        'id_buku'          : id_buku,
        'kode_buku': kode_buku,
        'nama_buku'       : nama_buku,
        'id_jenis'        : id_jenis,
        'id_nakes': id_nakes,
        'id_penerbit'      : id_penerbit,
        'id_penulis'       : id_penulis,
        'gambar'          : gambar,
        'userfile'         : userfile
    }

    return jsonify({"result": result})

```

c. Proses hapus

```

# DELETE
@buku.route('/buku/<id_buku>', methods=['DELETE'])
def delete_buku(id_buku):
    cur = db.connection.cursor()
    response = cur.execute("DELETE FROM tbl_buku where id_buku="
+ id_buku)
    db.connection.commit()

    if response > 0:
        result = {'message' : 'record deleted'}
    else:
        result = {'message' : 'no record found'}
    return jsonify({"result": result})

```

4. Perhitungan metode vikor

Pada proses ini untuk menghitung nilai buku dengan menggunakan metode VIKOR

```

def hitung_vikor(self, alternatif, kriteria, tipe_kriteria,
bobot_kriteria, matriks_keputusan):
    nilai_maksimum = []
    nilai_minimum = []
    for i in range(len(kriteria)):
        nilai_maksimum.append(0)
        nilai_minimum.append(0)
        for j in range(len(alternatif)):
            if ((j == 0) or (nilai_maksimum[i] <
matriks_keputusan[j][i])):
                nilai_maksimum[i] = matriks_keputusan[j][i]
            if ((j == 0) or (nilai_minimum[i] >
matriks_keputusan[j][i])):
                nilai_minimum[i] = matriks_keputusan[j][i]

    matriks_normalisasi = []
    for i in range(len(alternatif)):
        matriks_normalisasi.append([])
        for j in range(len(kriteria)):
            matriks_normalisasi[i].append(0)
            if tipe_kriteria[j] == "cost":
                matriks_normalisasi[i][j] =

```

```

    (matriks_keputusan[i][j] - nilai_minimum[j]) /
    (nilai_maksimum[j] - nilai_minimum[j])
        elif tipe_kriteria[j] == "benefit": #
tipe_kriteria[j] == "benefit":
            matriks_normalisasi[i][j] =
            (nilai_maksimum[j] - matriks_keputusan[i][j]) /
            (nilai_maksimum[j] - nilai_minimum[j])

matriks_normalisasi_terbobot = []
    for i in range(len(alternatif)):
        matriks_normalisasi_terbobot.append([])
        for j in range(len(kriteria)):
            matriks_normalisasi_terbobot[i].append(0)
            matriks_normalisasi_terbobot[i][j] =
matriks_normalisasi[i][j] * bobot_kriteria[j]

Si = []
    Ri = []
    for i in range(len(alternatif)):
        Si.append(0)
        Ri.append(0)
        for j in range(len(kriteria)):
            Si[i] = Si[i] +
matriks_normalisasi_terbobot[i][j]
            if ((j == 0) or (Ri[i] <
matriks_normalisasi_terbobot[i][j])):
                Ri[i] = matriks_normalisasi_terbobot[i][j]

Splus = max(Si)
    Smin = min(Si)
    Rplus = max(Ri)
    Rmin = min(Ri)
    V = 0.5

    hasil_akhir = []
    for i in range(len(alternatif)):
        hasil_akhir.append(0)
        hasil_akhir[i] = (V * ((Si[i] - Smin) / (Splus -
Smin))) + ((1 - V) * ((Ri[i] - Rmin) / (Rplus - Rmin)))

alternatif_ranking = []
    hasil_akhir_ranking = []
    for i in range(len(alternatif)):
        alternatif_ranking.append(alternatif[i])
        hasil_akhir_ranking.append(hasil_akhir[i])

    for i in range(len(alternatif)):
        for j in range(len(alternatif)):
            if j > i:
                if (hasil_akhir_ranking[i] >
hasil_akhir_ranking[j]):
                    tmp_alternatif = alternatif_ranking[i]
                    tmp_hasil_akhir = hasil_akhir_ranking[i]
                    alternatif_ranking[i] =
alternatif_ranking[j]
                    hasil_akhir_ranking[i] =
hasil_akhir_ranking[j]
                    alternatif_ranking[j] = tmp_alternatif
                    hasil_akhir_ranking[j] = tmp_hasil_akhir

    return alternatif_ranking

```

Pada proses ini adalah untuk menghasilkan ranking dari metode VIKOR untuk ditampilkan pada aplikasi Digital Repository Rumah Sakit Umum Universitas Kristen Indonesia

```
@vikor.route('/buku_vikor', methods=['GET'])
def get_buku_vikor_data():
    try:
        ObjekVikor = CobaHitungVIKOR()
        alternatif = ObjekVikor.data_IdBuku()
        kriteria = ObjekVikor.data_Kriteria()
        tipe_kriteria = ObjekVikor.data_TipeKriteria()
        bobot_kriteria = ObjekVikor.data_Bobot()
        matriks_keputusan = ObjekVikor.data_matriks_keputusan()

        arrayData = ObjekVikor.hitung_vikor(alternatif,
        kriteria, tipe_kriteria, bobot_kriteria, matriks_keputusan)
        data = [str(int) for int in arrayData]
        res = ",".join(data)

        cur = db.connection.cursor()
        cur.execute(
            "SELECT id_buku, kode_buku, nama_buku, jenis_buku,
            tenaga_kesehatan, nama_penerbit, tahun_terbit, nama_penulis,
            gambar, userfile "
            "FROM tbl_buku as b "
            "INNER JOIN tbl_jenis as j "
            "on b.id_jenis = j.id_jenis "
            "INNER JOIN tbl_nakes as n "
            "on b.id_nakes = n.id_nakes "
            "INNER JOIN tbl_penerbit as p1 "
            "on b.id_penerbit = p1.id_penerbit "
            "INNER JOIN tbl_penulis as p2 "
            "on b.id_penulis = p2.id_penulis WHERE id_buku in
            (" + res + ") ORDER BY FIELD(id_buku," + res + ")")

        db.connection.commit()

        rv = cur.fetchall()
        return jsonify(rv)
        # return jsonify({'message': 'data pada ' + arrayData + '
        ada'})

    except:
        cur = db.connection.cursor()
        cur.execute(
            "SELECT id_buku, kode_buku, nama_buku, jenis_buku,
            tenaga_kesehatan, nama_penerbit, tahun_terbit, nama_penulis,
            gambar, userfile "
            "FROM tbl_buku as b "
            "INNER JOIN tbl_jenis as j "
            "on b.id_jenis = j.id_jenis "
            "INNER JOIN tbl_nakes as n "
            "on b.id_nakes = n.id_nakes "
            "INNER JOIN tbl_penerbit as p1 "
            "on b.id_penerbit = p1.id_penerbit "
            "INNER JOIN tbl_penulis as p2 "
            "on b.id_penulis = p2.id_penulis ")
```

```

db.connection.commit()

rv = cur.fetchall()
return jsonify(rv)

```

Frontend

1. Fungsi mengolah kriteria buku

```

import axios from 'axios'
export const getKriteria = () => {
  return axios
    .get('/kriteria', {
      headers: { "Content-type": "application/json" }
    })
    .then(res => {
      var tbl_kriteria = []
      Object.keys(res.data).forEach((key) => {
        var val = res.data[key]
        tbl_kriteria.push([
          val.id_kriteria,
          val.kode,
          val.nama_kriteria,
          val.bobot,
          val.tipe_kriteria])
      })
      return tbl_kriteria
    })
}
export const addKriteria = newKriteria => {
  return axios
    .post(
      '/kriteria', {
        kode: newKriteria.kode,
        nama_kriteria: newKriteria.nama_kriteria,
        bobot: newKriteria.bobot,
        tipe_kriteria: newKriteria.tipe_kriteria,
      }, {
        headers: { "Content-type": "application/json" }
      })
    .then((res) => {
      console.log(res)
    })
}
export const deleteKriteria = id_kriteria => {
  axios
    .delete(
      `/kriteria/${id_kriteria}`, {
        headers: { "Content-type": "application/json" }
      })
    .then((res) => {
      console.log(res)
    })
    .catch((res) => {
      console.log(res)
    })
}
}

```

2. Fungsi mengolah buku

```
export const addBuku = newBuku => {
  return axios
    .post(
      '/buku', {
        kode_buku: newBuku.kode_buku,
        nama_buku: newBuku.nama_buku,
        id_jenis: newBuku.id_jenis,
        id_nakes: newBuku.id_nakes,
        id_penerbit: newBuku.id_penerbit,
        id_penulis: newBuku.id_penulis,
        gambar: newBuku.gambar,
        userfile: newBuku.userfile
      }, {
        headers: {
          "Content-Type": "multipart/form-data",
          "Accept": "application/json",
          "type": "formData"
        }
      })
    .then((res) => {
      console.log(res)
    })
}
```

3. Tampilan buku (User)

```
import React, {Component} from "react";
import { Badge, Card, CardBody, CardHeader, Col, Pagination,
  PaginationItem, PaginationLink, Row, Table } from 'reactstrap';
import {Link, withRouter} from 'react-router-dom';

const buku = {
  marginLeft: 85,
  marginRight: 25,
  textAlign: 'center',
  width: 200,
  height: 250
}

const Book = (props) =>{
  return(
    <Col xs="12" xl="4">
      <Card>
        <CardBody>
          <img className="center" src={process.env.PUBLIC_URL +
            '/Images/' + props.gambar } style={buku} alt="buku_menu"></img>
          <br/>
          <br/>
          <h6><Link to={props.LinkDetail}>{props.nama_buku}
            ({props.tenaga_kesehatan})</Link></h6>
          <h6>{props.nama_penulis}</h6>
          <h6>{props.nama_penerbit} ({props.tahun_terbit})</h6>
          <h6>{props.jenis_buku}</h6>

          </CardBody>
        </Card>
      </Col>
    )
  )
}
```

```

import React, { Component } from 'react';
import { Badge, Card, CardBody, CardHeader, Col, Pagination,
PaginationItem, PaginationLink, Row, Table } from 'reactstrap';
import Book from "../Buku";

class BukuTables extends Component {
  state = {
    ListBuku:[]
  }
  ambilDataDariServerAPI = () =>{
    fetch('http://127.0.0.1:5000/buku')
      .then(response => response.json())
      .then(jsonHasilAmbilDariAPI => {
        this.setState({
          ListBuku: jsonHasilAmbilDariAPI
        })
      })
  }

  ambilDataTahunTerbaru = () => {
    fetch('http://127.0.0.1:5000/tahun_ascending')
      .then(response => response.json())
      .then(jsonHasilAmbilDariAPI => {
        this.setState({
          ListBuku: jsonHasilAmbilDariAPI
        })
      })
  }

  ambilDataTahunTerlama = () => {
    fetch('http://127.0.0.1:5000/tahun_descending')
      .then(response => response.json())
      .then(jsonHasilAmbilDariAPI => {
        this.setState({
          ListBuku: jsonHasilAmbilDariAPI
        })
      })
  }

  ambilDataBukuAscending = () => {
    fetch('http://127.0.0.1:5000/buku_ascending')
      .then(response => response.json())
      .then(jsonHasilAmbilDariAPI => {
        this.setState({
          ListBuku: jsonHasilAmbilDariAPI
        })
      })
  }

  ambilDataBukuDescending = () => {
    fetch('http://127.0.0.1:5000/buku_descending')
      .then(response => response.json())
      .then(jsonHasilAmbilDariAPI => {
        this.setState({
          ListBuku: jsonHasilAmbilDariAPI
        })
      })
  }
}

```



```

handleChange(filter){
  if(filter == "tahun_ascending"){
    this.ambilDataTahunTerbaru()
  }
  else if(filter == "tahun_descending"){
    this.ambilDataTahunTerlama()
  }
  else if(filter == "judul_ascending"){
    this.ambilDataBukuAscending()
  }
  else if(filter == "judul_descending"){
    this.ambilDataBukuDescending()
  }
  else{
    this.ambilDataDariServerAPI()
  }
}
componentDidMount() {
  // this.getBukuMenu()
  this.ambilDataDariServerAPI()
}
render() {
  return (
    <div className="animated fadeIn">
      <br/>
      <from>
        <select onChange={ (val) =>
this.handleChange(val.target.value)}>
          <option value="0">Order By:</option>
          <option value="Buku">Buku</option>
          <option value="tahun_ascending">Tahun Terbit
Terlama</option>
          <option value="tahun_descending">Tahun Terbit
Terbaru</option>
          <option value="judul_ascending">Judul Buku A -
Z</option>
          <option value="judul_descending">Judul Buku Z -
A</option>

```

4. Tampilan rekomendasi buku (user)

```

import React, {Component} from "react";
import { Badge, Card, CardBody, CardHeader, Col, Pagination,
PaginationItem, PaginationLink, Row, Table } from 'reactstrap';
import {Link, withRouter} from 'react-router-dom';
const Book = (props) =>{
  return(
    <Col xs="12" xl="4">
      <Card>
        <CardBody>
          <img className="center" src={process.env.PUBLIC_URL +
'/Images/' + props.gambar } style={buku} alt="buku_menu"></img>
          <br/>
          <br/>
          <h6><Link to={props.LinkDetail}>{props.nama_buku}
({props.tenaga_kesehatan})</Link></h6>
          <h6>{props.nama_penulis}</h6>
          <h6>{props.nama_penerbit} ({props.tahun_terbit})</h6>
          <h6>{props.jenis_buku}</h6>
        </CardBody>
      </Card>
    </Col>
  )
}
export default Book

```

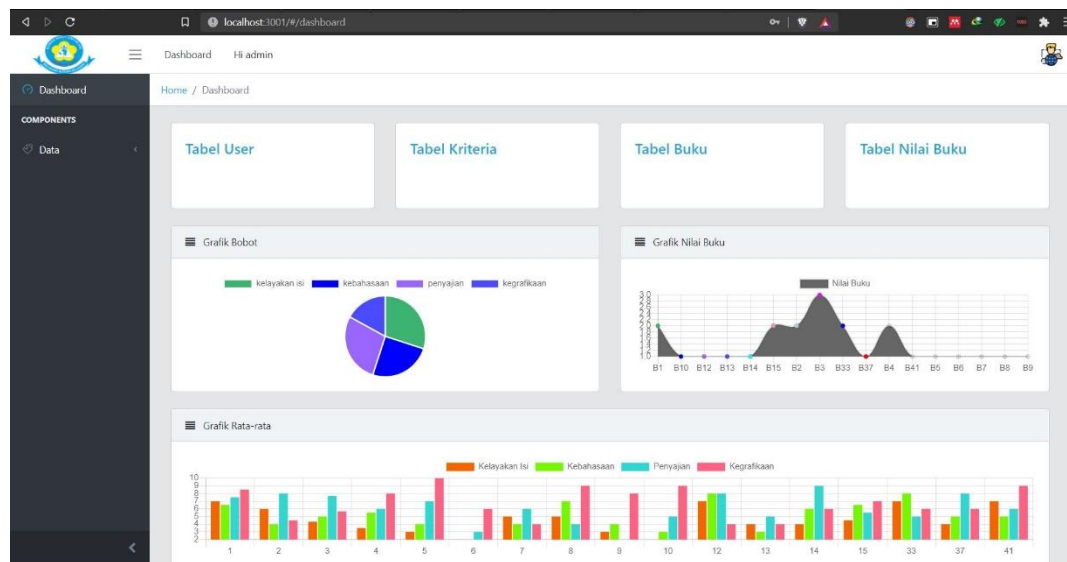
5.3. Implementasi tampilan

Implementasi tampilan dibuat berdasarkan perancangan antar muka yang telah dibuat sebelumnya.

5.3.1. Tampilan Admin

1. Tampilan Dashboard

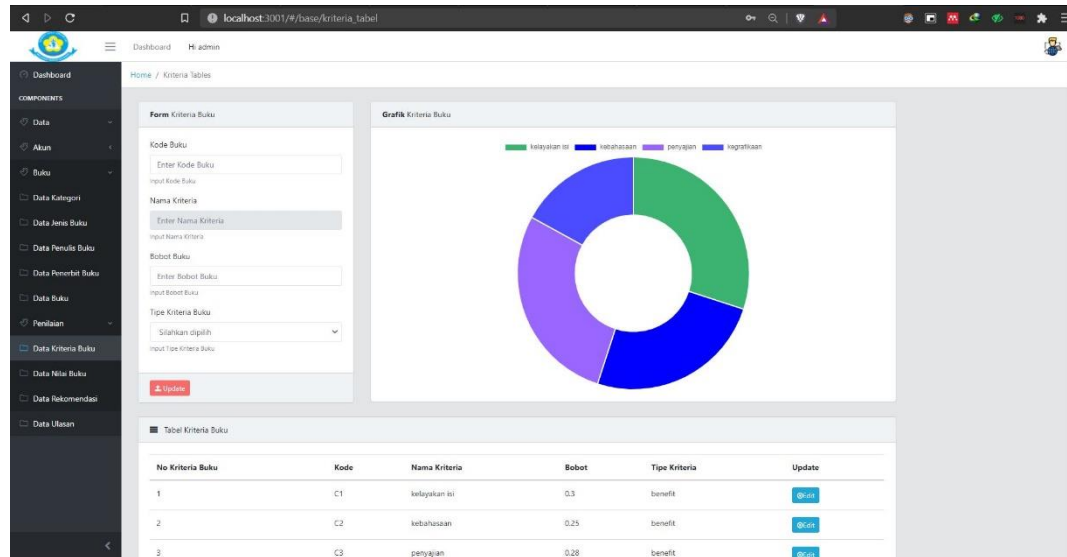
Tampilan Dashboard menampilkan data buku terbaru, grafik data buku, grafik, data kriteria, grafik data penilaian buku berdasarkan kriteria penilaian buku.



Gambar 5.3.1.1. Gambar Tampilan Dashboard

2. Tampilan Kriteria Buku

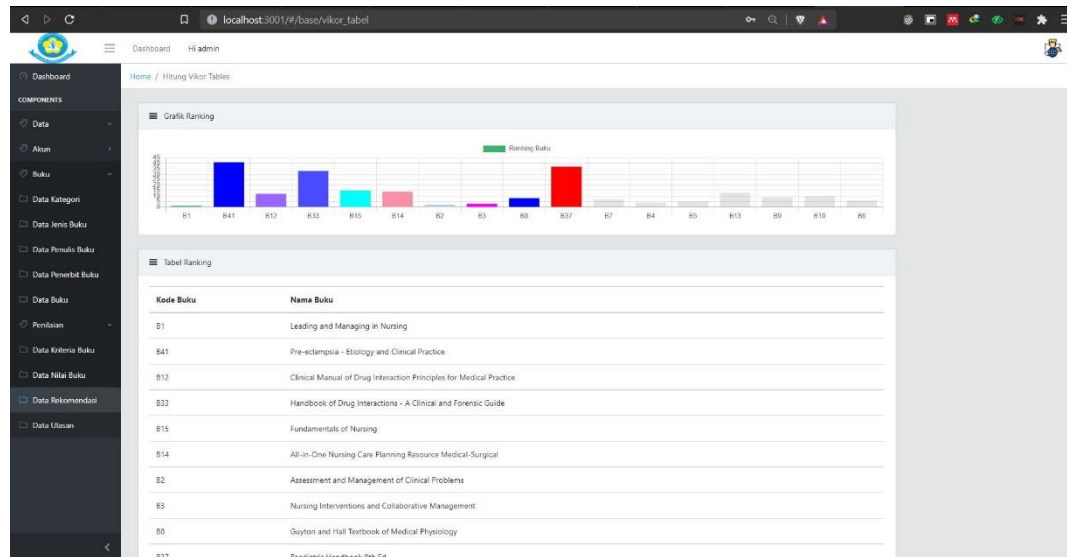
Tampilan Kriteria Buku menampilkan kriteria data buku dan bobot kriteria, admin dapat mengubah bobot kriteria dan tipe kriteria yang dimana akan mempengaruhi pada perhitungan metode VIKOR



Gambar 5.3.1.2. Gambar Tampilan Kriteria Buku

3. Tampilan rekomendasi buku

Tampilan rekomendasi buku menampilkan hasil ranking dari metode VIKOR dan admin dapat memonitoring hasil perangkingan dengan menggunakan metode VIKOR di website

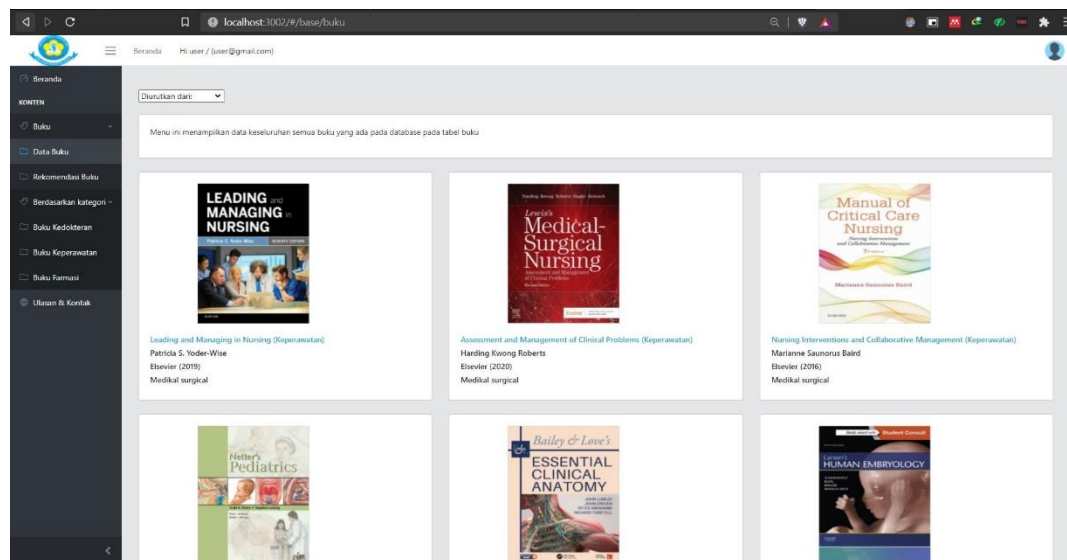


Gambar 5.3.1.3. Gambar Tampilan Rekomendasi Buku

5.3.2. Tampilan User

1. Tampilan buku

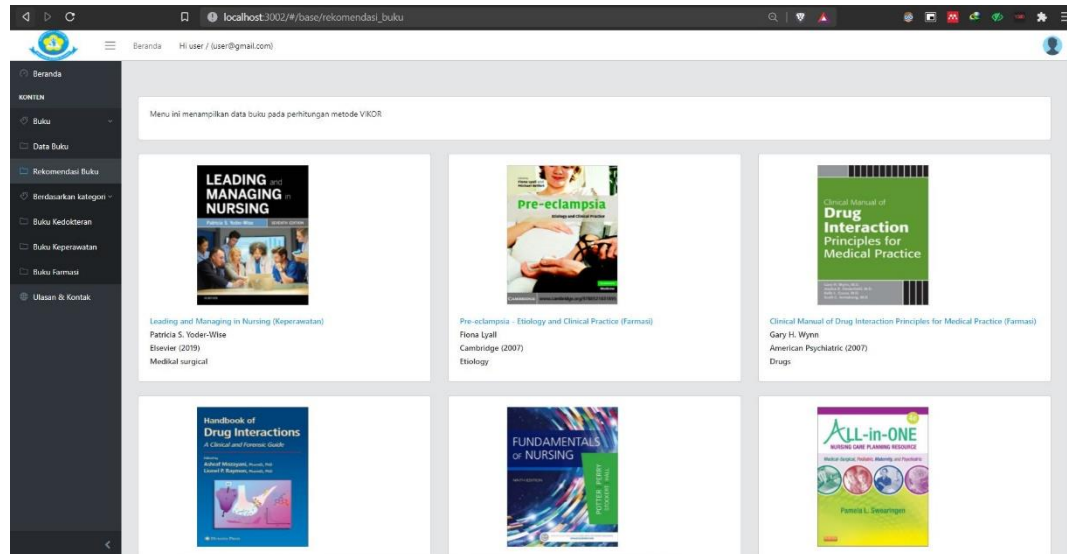
Tampilan Buku ini menampilkan semua buku elektronik berada pada tabel buku di database *Digital_Repository* dan pengguna dapat memilih buku mana yang ingin dibaca.



Gambar 5.3.2.1. Gambar Tampilan Buku

2. Tampilan Rekomendasi Buku

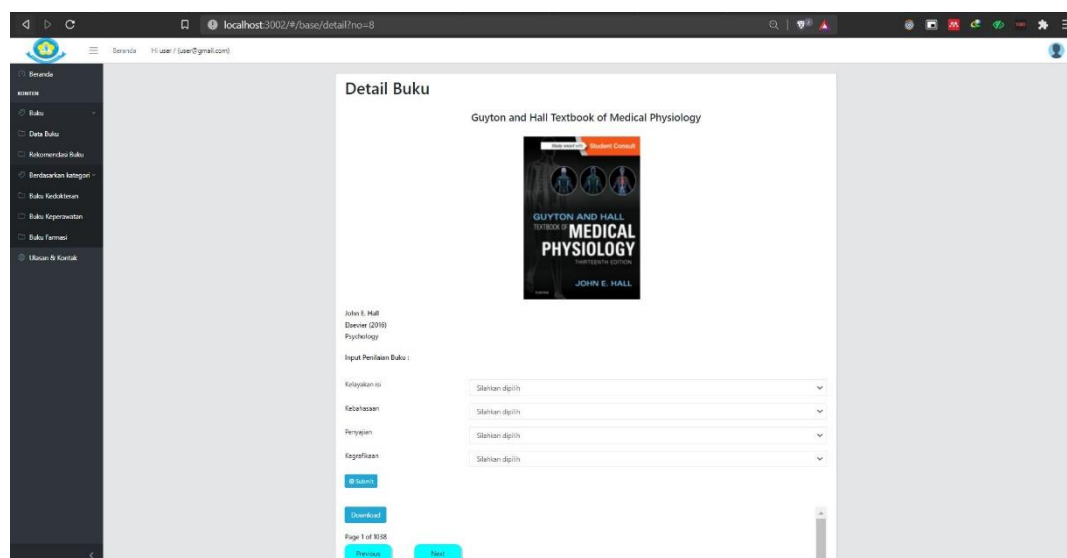
Tampilan Rekomendasi Buku menampilkan hasil rekomendasi berdasarkan ranking dengan menggunakan metode VIKOR sehingga pengguna dapat mengetahui buku mana yang direkomendasikan untuk dibaca.



Gambar 5.3.2.2. Gambar Rekomendasi Buku

3. Tampilan Detail Buku

Tampilan Detail Buku menampilkan isi buku (judul, penulis, penerbit, tahun terbit, dan isi pdf) dan terdapat form penilaian berdasarkan kriteria penilaian buku yang dimana penilaian dijadikan perhitungan dengan metode VIKOR.



Gambar 5.3.2.3. Gambar Tampilan Detail Buku

5.4. Pengujian Fungsionalitas sistem

Pengujian fungsionalitas sistem dilakukan dengan cara menjalankan tiap fitur dalam aplikasi dan melihat kesesuaian hasil yang terjadi dengan hasil yang diharapkan.

No	Test Case	Hasil yang diharapkan	Hasil yang didapatkan	keterangan
1	Mengolah user	Admin dapat mengolah data user seperti mengedit dan menghapus data user	Admin dapat melihat, edit, dan hapus user	Berhasil
2	Mengelola data kategori	Admin mengolah kategori (tenaga kesehatan) dan jenis buku dalam bentuk CRUD	Admin dapat melihat, tambah, edit, dan hapus kategori (kedokteran, keperawatan, dan farmasi) dan jenis buku	Berhasil
3	Mengelola data buku	Admin dapat menginput buku dan dapat mengunggah gambar dan file pdf ke sistem	Admin dapat melihat, tambah, edit, dan hapus data buku	Berhasil
4	Mengelola data ulasan	Admin dapat memonitoring ulasan yang diberikan user mengenai website digital repository	Admin dapat melihat dan hapus data ulasan	Berhasil
5	Mengelola data kriteria buku	Admin dapat mengolah data kriteria (bobot dan tipe kriteria) menjadi tolak ukur dalam perhitungan metode VIKOR	Admin dapat melihat, edit, dan hapus data kriteria buku	Berhasil
6	Mengelola data nilai buku	Admin dapat menginput penilaian berdasarkan kriteria penilaian	Admin dapat melihat, tambah, edit, dan hapus data nilai buku	Berhasil
7	Melihat data buku	User melihat daftar buku-buku yang ingin dibaca	User dapat melihat data buku (data buku berdasarkan tahun terbit, kategori, rekomendasi, dan urutan judul buku)	Berhasil
8	Mengakses isi buku	User dapat melihat isi buku dan isi pdf dari buku tersebut	User dapat melihat isi buku (judul, penulis, penerbit, dan tahun terbit), melihat isi pdf dan mengunduh buku	Berhasil
9	Mengisi ulasan buku	User dapat mengisi komentar mengenai website digital repository	User dapat input ulasan website	Berhasil
10	Menginput nilai buku	User dapat menginput penilaian buku untuk dijadikan tolak ukur dari rekomendasi dengan metode VIKOR	User dapat input nilai buku berdasarkan kriteria penilaian buku (kelayakan isi, kebahasaan, penyajian, dan kegrafikaan)	Berhasil

BAB VI. HASIL DAN PEMBAHASAN

Pembahasan ini mengenai argumentasi rasional dari penulis yang disusun secara sistematis berdasarkan fakta ilmiah yang diperoleh dari hasil pengujian. Pembahasan hasil pengujian dapat disajikan dalam bentuk uraian teoritik, baik secara kualitatif maupun kuantitatif

6.1. Hasil Pengujian Akurasi

Akurasi didapat dengan membandingkan 17 dari 50 buku yang dipilih dan dinilai oleh user dan dipilih dari sistem dengan hasil uji manual (menggunakan microsoft excel). Hasil pengujian akurasi ditunjukkan pada tabel:

Tabel 6.1.1. Tabel Pengujian Akurasi

Kode Buku	Uji Manual	Uji Sistem
B1	0	0
B2	0,393677478	0.3936774781196784
B3	0,406192111	0.4061921112035168
B4	0,531180943	0.5311809433289771
B5	0,591517168	0.5915171676369253
B6	1	1
B7	0,505096828	0.505096827848402
B8	0,441843887	0.4418438873707972
B9	0,849542592	0.8495425923725793
B10	0,860045147	0.8600451467268623
B12	0,25678983	0.2567898301057384
B13	0,681735377	0.6817353768167598
B14	0,390709279	0.39070927884044193
B15	0,368747772	0.3687477723654509
B33	0,278626589	0.27862658904597837
B37	0,446014019	0.44601401924676254
B41	0,204942379	0.20494237851966257

Tahapan Perhitungan Uji Akurasi

$$\text{Nilai Error} = \text{nilai uji sistem} - \text{nilai uji manual}$$

$$\text{Nilai Error \%} = (\text{nilai error} / \text{nilai uji sistem}) \times 100\%$$

$$\text{Nilai Akurasi} = 100 - \text{nilai Error \%}$$

Hasil Perhitungan akurasi dapat dilihat pada tabel:

Tabel 6.1.2. Tabel Perhitungan Pengujian Akurasi

Kode Buku	Uji Manual	Uji Sistem	Error	Error %
B1	0	0	0	0
B2	0,393677478	0.3936774781196784	0	0
B3	0,406192111	0.4061921112035168	0	0
B4	0,531180943	0.5311809433289771	0	0
B5	0,591517168	0.5915171676369253	0	0
B6	1	1	0	0
B7	0,505096828	0.505096827848402	0	0
B8	0,441843887	0.4418438873707972	0	0
B9	0,849542592	0.8495425923725793	0	0
B10	0,860045147	0.8600451467268623	0	0
B12	0,25678983	0.2567898301057384	0	0
B13	0,681735377	0.6817353768167598	0	0
B14	0,390709279	0.39070927884044193	0	0
B15	0,368747772	0.3687477723654509	0	0
B33	0,278626589	0.27862658904597837	0	0
B37	0,446014019	0.44601401924676254	0	0
B41	0,204942379	0.20494237851966257	0	0
Total Error				0
Rata-Rata Error				
Nilai Akurasi				100 %

Berdasarkan tabel 6.1.2. dapat diamati bahwa perhitungan manual dengan hasil perhitungan sistem memiliki persamaan dengan nilai kesesuaian sebesar 100 %. Akurasi pada perhitungan ini dinyatakan sangat baik dan sangat akurat karena nilai hasil akurasi $\geq 95\%$

BAB VII. KESIMPULAN DAN SARAN

7.1. Kesimpulan

Berdasarkan hasil penelitian dan pengujian yang telah dilakukan dapat ditarik kesimpulan sebagai berikut :

1. Sistem dapat merekomendasikan E-book sesuai dengan kriteria yang dihasilkan dari sistem dengan menggunakan metode *VIKOR* menentukan rekomendasi E-book sesuai dengan kriteria Penilaian buku (kelayakan isi, kebahasaan, penyajian, dan kegrafikaan).
2. Sistem dapat merekomendasikan E-book dengan menggunakan metode *VIKOR* dan dari hasil perhitungan manual (microsoft excel) dengan perhitungan sistem memiliki akurasi 99,98% persentase kesesuaian.
3. Rekomendasi E-book sesuai dengan kriteria penilaian buku (kelayakan isi, penyajian, kebahasaan, dan kegrafikaan) dinyatakan telah berhasil dilakukan dengan menggunakan Python Framework Flask sebagai Backend, Javascript Framework ReactJS sebagai Frontend, dan Mysql sebagai database

7.2. Saran

Saran yang dapat diberikan dari hasil penelitian untuk pengembangan sistem ini kedepan sebagai berikut :

1. Perbaiki tampilan pengguna dan tampilan admin yang lebih menarik.
2. Dapat dikembangkan dengan metode lain seperti metode *Borda* sebagai *Group Decision Support System* sehingga penilaian untuk merekomendasikan E-book dapat dilakukan oleh lebih dari satu pengguna tidak hanya dengan menghitung rata-rata.

DAFTAR PUSTAKA

- Apa Itu JavaScript ? Pemahaman Dasar Bagi Pemula.* (n.d.). Retrieved November 25, 2020, from <https://idwebhost.com/blog/apa-itu-javascript/>
- Aprida, C. D., & Febriliyan, S. (2013). Pembuatan Sistem Informasi Beasiswa Internal Direktorat Jenderal Perbendaharaan Menggunakan PHP dan MySQL. *Jurnal Teknik Pomits*, 2(2), 350–354.
<http://ejurnal.its.ac.id/index.php/teknik/article/view/4822>
- Arisandi, A., & Pribadi, E. S. (2020). Analisa Metode VIKOR pada Rekomendasi Alat Musik Keyboard Electone Terbaik. *KESATRIA: Jurnal Penerapan Sistem Informasi (Komputer & Manajemen)*, 1(1), 31–36.
<https://doi.org/10.30645/kesatria.v1i1.14>
- Arsanti, M. (2018). Pengembangan Bahan Ajar Mata Kuliah Penulisan Kreatif Bermuatan Nilai-Nilai Pendidikan Karakter Religius Bagi Mahasiswa Prodi Pbsi, Fkip, Unissula. *KREDO : Jurnal Ilmiah Bahasa Dan Sastra*, 1(2), 71–90. <https://doi.org/10.24176/kredo.v1i2.2107>
- Artikel, I. (2017). *Manajemen keuangan publik*.
- Bako, S., & Lubis, N. (2018). *Sistem Pendukung Keputusan Pemilihan Kamar Hotel Terbaik Di Kota Medan Dengan Menggunakan Metode Vikor*. 5(5), 525–531.
- Firgiawan, F., wina, W., & Abdillah, G. (2018). *Sistem Pendukung Keputusan Rekomendasi Pemilihan Buku Komputer Di Amazon.Com Menggunakan Metode Technique for Order Preference By Similarity To Ideal Solution (Topis)*. 2006, 255–260.
- Hamria Hamria, H., Azwar, A., & Arja, K. (2020). Sistem Pendukung Keputusan Penilaian Kinerja Jasa Pramubakti Menggunakan Metode Moora. *Jurnal Ilmiah Informatika*, 8(01), 25. <https://doi.org/10.33884/jif.v8i1.1824>
- Hutapea Mentari Ananda; Karim, Abdul; Suginam, Suginam, B. J. H. (2018). Sistem Pendukung Keputusan Penentuan Jenis Kulit Terbaik Untuk Pembuatan Sepatu Dengan Menggunakan Metode VIKOR. *JURIKOM (Jurnal Riset Komputer)*, 5(Vol 5, No 1 (2018): Februari 2018), 6–12.
<http://ejurnal.stmik-budidarma.ac.id/index.php/jurikom/article/view/562>
- Informasi lengkap dan buat janji di RSU Universitas Kristen Indonesia (UKI). Biaya, tindakan medis, daftar dokter, selengkapnya.* (n.d.). Retrieved November 24, 2020, from <https://www.halodoc.com/rumah-sakit/nama/rsu-universitas-kristen-indonesia-uki>
- JavaScript - Wikipedia.* (n.d.). Retrieved November 25, 2020, from <https://en.wikipedia.org/wiki/JavaScript>
- Meningkatkan, P. Y. U., Konsep, P., Pada, P., Sd, S., Darul, I., & Semarang, K. (2020). *Pendidikan dasar*. 7(1), 105–120.

- MySQL - Wikipedia*. (n.d.). Retrieved November 25, 2020, from <https://en.wikipedia.org/wiki/MySQL>
- Nursaid, F. F., Hendra Brata, A., & Kharisma, A. P. (2020). Pengembangan Sistem Informasi Pengelolaan Persediaan Barang Dengan ReactJS Dan React Native Menggunakan Prototype (Studi Kasus : Toko Uda Fajri). *J-Ptiik.Ub.Ac.Id*, 4(1), 46–55. <http://j-ptiik.ub.ac.id>
- Parlika, R., Ilham Pradika, S., Muhammad Hakim, A., & Rachman N.M., K. (2020). Bot Whatsapp Sebagai Pemberi Data Statistik Covid-19 Menggunakan Php, Flask, Dan Mysql. *Jurnal Informatika Dan Sistem Informasi (JIFoSI)*, 1(2 SE-Articles), 282–293. <http://jifosi.upnjatim.ac.id/index.php/jifosi/article/view/101>
- Pengertian MySQL, Fungsi, dan Cara Kerjanya (Lengkap)*. (n.d.). Retrieved November 25, 2020, from https://www.niagahoster.co.id/blog/mysql-adalah/#Pengertian_MySQL
- Pohan, H., & Sinaga, D. E. (2020). Penerapan Metode Moora Dalam Menentukan Parfume Terbaik Berdasarkan Kepribadian. *KESATRIA: Jurnal Penerapan Sistem Informasi (Komputer & Manajemen)*, 1(2), 59–63. <https://doi.org/10.30645/kesatria.v1i2.21>
- Python (bahasa pemrograman) - Wikipedia bahasa Indonesia, ensiklopedia bebas*. (n.d.). Retrieved November 25, 2020, from [https://id.wikipedia.org/wiki/Python_\(bahasa_pemrograman\)](https://id.wikipedia.org/wiki/Python_(bahasa_pemrograman))
- Qurotianti, A. (2019). Optimalisasi Pemanfaatan Akses Digital Repository Dalam Menunjang Tri Dharma Perguruan Tinggi (Studi Kasus Di Perpustakaan Universitas Muhammadiyah Yogyakarta). *Pustabiblia: Journal of Library and Information Science*, 3(2), 113–125. <https://doi.org/10.18326/pustabiblia.v3i2.113-125>
- Radjatadoe, F. S., Sihotang, D. M., Boru, M., Komputer, J. I., & Cendana, U. N. (2017). *PEMBUATAN APLIKASI DIGITAL LIBRARY DI JURUSAN ILMU menyalurkan informasi secara cepat , tepat dan global . Dengan pesatnya perkembangan teknologi didapat oleh pemakai komputer yang terhubung ke internet . Website pada awalnya adalah ruang informasi dalam .* 5(2), 34–41.
- Safii, M., & Zulhamsyah, A. (2018). Sistem Pendukung Keputusan Pemilihan Mekanik Sepeda Motor Yamaha Alfascorfii Dengan Metode Multi Objective Optimization On The Basis Of Ratio Analysis (MOORA). *J-SAKTI (Jurnal Sains Komputer Dan Informatika)*, 2(2), 162. <https://doi.org/10.30645/j-sakti.v2i2.79>
- Sharma, M. (2017). Multi attribute decision making techniques. *International Journal of Research in Management, Science & Technology*, 1(1), 49–51. http://www.ijrmst.org/vol1_no1.php
- Simamora, B. (2017). Rancang Bangun Sistem Rekomendasi Televisi LED Dengan Metode Vikor Berbasis Web. *Jurnal ULTIMATICS*, 9(1), 42–49.

<https://doi.org/10.31937/ti.v9i1.563>

- Sutrikanti, N., Situmorang, H., Fachrurrazi, Nurdiyanto, H., & Mesran, M. (2018). Implementasi Pendukung Keputusan Dalam Pemilihan Calon Peserta Cerdas Cermat Tingkat SMA Menerapkan Metode VIKOR. *JURIKOM (Jurnal Riset Komputer)*, 5(2407-389X), 109–113.
- Tumanggor, H., Haloho, M., Ramadhani, P., & Darma Nasution, S. (2018). Penerapan Metode VIKOR Dalam Penentuan Penerima Dana Bantuan Rumah Tidak Layak Huni. *Jurikom*, 5(1), 71–78.
- Ubaya, H., Siswati, S. D., Afriansyah, R., Studi, P., Komputer, T., Komputer, D., Komputer, F. I., Sriwijaya, U., Studi, P., Perangkat, R., Manufaktur, P., & Belitung, B. (1907). *Perancangan Monitoring Sensor berbasis Javascript dan Plotly*. 43–49.
- What is Python? Executive Summary* / *Python.org*. (n.d.). Retrieved November 25, 2020, from <https://www.python.org/doc/essays/blurb/>
- Wicaksana, Rudy, & Lukito. (2017). Mengukur Kesuksesan Layanan Digital Library Universitas Gadjah Mada (Ugm). *Prosiding SNATIF*, 4(2), 189–196.

Lampiran

Lampiran Buku

Buku Kedokteran					
kode	nama buku	jenis buku	penerbit	tahun terbit	penulis
B5	Bailey & Loves Essential Clinical Anatomy	anatomy	paperback	2018	John S.P. Lumley
B6	Larsens Human Embryology	embryology	wiley blackwell	2009	Gary C. Schoenwolf
B7	Atlas of Human Anatomy	anatomy	elsevier	2018	Frank H. Netter
B8	Guyton and Hall Textbook of Medical Physiology	psychology	elsevier	2016	John E. Hall
B17	Netters Concise Radiologic Anatomy	anatomy	elsevier	2014	Joel A. Vilensky
B18	Lecture Notes_ Radiology, 3rd Edition	radiology	wiley blackwell	2010	Pradip R Patel
B19	Diagnostic Radiology V.2	radiology	elsevier	2015	Andreas Adam
B20	Diagnostic the Dog and Cat	anatomy	saunders company	2000	kevin kealy
B21	PRIMER OF DIAGNOSTIC IMAGING, 4th Edition	diagnostic	elsevier	2007	Ralph Weissleder
B22	Atlas of Anatomy Head, Neck and Neuroanatomy	anatomy	elsevier	2017	Friedrich Paulsen
B23	Yoga Anatomy	anatomy	breath trust	2007	Leslie Kominoff
B50	Review of Medical Physiology	psychology	McGraw Hill	2019	Kim. E. Berret

Buku Keperawatan					
kode	nama buku	jenis buku	penerbit	tahun terbit	penulis
B1	Leading and Managing in Nursing	medikal surgikal	elsevier	2019	Patricia S. Yoder-Wise
B2	Assessment and Management of Clinical Problems	medikal surgikal	elsevier	2020	Harding Kwong Roberts
B3	Nursing Interventions and Collaborative Management	medikal surgikal	elsevier	2016	Marianne Saunorus Baird
B4	Netter's Pediatrics	pediatric	elsevier	2011	Todd A. Florin
B13	USMLE Step 2 CK Lecture Notes 2019 Pediatrics	pediatric	kaplan medical	2019	William G

B14	All-in-One Nursing Care Planning Resource Medical-Surgical	medikal surgikal	elsevier	2016	pamela L. Swearigan
B15	Fundamentals of Nursing	medikal surgikal	elsevier	2017	Patricia A. Potter
B16	Mosby's Pharmacology Memory Notecards Visual, Mnemonic, and Memory Aids for Nurses	pharmacology	elsevier	2019	JoAnn Zerwekh

Buku Farmasi					
kode	nama buku	jenis buku	penerbit	tahun terbit	penulis
B9	Drugs And Human Location	drugs	elsevier	2019	Todd A. Florin
B10	PRESCRIBING FOR ELDERLY PATIENTS	drugs	wiley- blackwell	2009	Stephen Jackson
B11	Drug–Drug Interaction Primer	prescribing	american psychiatric	2007	Neil B. Sandsom
B12	Clinical Manual of Drug Interaction Principles	drugs	american psychiatric	2007	Gary H. Wynn
B24	A Manual of Adverse Drug Interactions	drugs	elsevier	2017	J.P.Griffin
B25	Adherence to Pediatric Medical Regimens	medical regimen	paperback	2014	Michael Rapoff
B26	British National Formulary	formulary	BMJ Group	2009	Media GMBH
B27	British National Formulary for Children	formulary	BMJ Group	2009	Media GMBH
B28	Community Pharmacy Handbook	drugs	pharmaceutical press	2008	Jon Waterfield
B29	Drug interactions in infectious disease 2nd ed	drugs	Humana Press	2018	Stephen Pesticelli
B30	Drug-Drug Interactions 2nd ed	drugs	Informa Healthcare	2008	David Rodriguez
B31	Drug-related problems in the elderly	drugs	Springer	2009	Patrik Midlov
B32	Drugs and Drugs 2nd Ed	drugs	Springer	2006	Hamilton
B33	Handbook of Drug Interactions - A Clinical and Forensic Guide	drugs	Humana Press	2004	Ashraf Mozayani
B34	MedFacts - Pocket Guide of Drug Interaction	drugs	Humana Press	2004	George Bailie
B35	Neonatal Formulary 4 - Drugs in Pregnancy and the First Year of Life - A Pharmacopoeia	formulary	BMJ Books	2003	Catherine Hall
B36	Neonatal Formulary 5 - Drug Use in Pregnancy and the First Year of Life	formulary	wiley- blackwell	2007	Vicky Pittman

B37	Paediatric Handbook 8th Ed	prescribing	wiley-blackwell	2009	Kate Thompson
B38	Pharmacotherapies for the Treatment of Opioid Dependence	pharmacotherapy	Informa Healthcare	2009	Richard Matick
B39	Pharmacotherapy of Diabetes - New Developments	pharmacotherapy	Springer	2007	Erik Mogensen
B40	Pharmacotherapy of Obesity	pharmacotherapy	Birkhauser	2008	Michael Parnham
B41	Pre-eclampsia - Etiology and Clinical Practice	etiology	Cambridge	2007	Fiona Lyall
B42	Prescribing in Diabetes	prescribing	Cambridge	2008	Jill Hill
B43	Prescribing in Pregnancy 4th Ed	prescribing	wiley-blackwell	2008	Peter Rubin
B44	Stockley's Drug Interactions 8th Ed	drugs	pharmaceutical press	2008	Karen Baxter
B45	Stockley's Drug Interactions 2009 Pocket Companion	drugs	pharmaceutical press	2009	Karen Baxter
B46	Veterinary Drug Handbook (Desk Edition)	drugs	Lowa University	1999	Donald Plumb
B47	Davis's Drug Guide for Nurses	drugs	Davis Company	2019	Hazard Vallerand
B48	Principles of Pharmacology The Pathophysiologic Basis of Drug Therapy	pharmacotherapy	Wolters Kluwer	2017	David Golan
B49	The Addicted Brain Why We Abuse Drugs, Alcohol, and Nicotine (FT Press Science)	drugs	elsevier	2011	Michael Kuhar