

BAB I

PENDAHULUAN

1.1 Latar Belakang

Air merupakan sumber kehidupan. Semua makhluk hidup dari mikroorganisme sampai dengan makhluk hidup yang lebih kompleks seperti manusia membutuhkan air. Air merupakan senyawa kimia yang melimpah di alam [1], namun hanya 2,53% air di bumi yang berupa air tawar dan dari 2,53% itu hanya 39,526% yang dapat dimanfaatkan dengan biaya rendah, yaitu: air di danau, sungai, waduk dan sumber air tanah dangkal. Air dibutuhkan di semua aspek kehidupan manusia mulai dari konsumsi, industri, transportasi, pertanian, dan peternakan, sehingga perlu dilakukan upaya bersama untuk mempertahankan keberadaannya demi kelangsungan kehidupan [2].

Meskipun jumlah air di alam melimpah, adakalanya suatu daerah mengalami kekeringan karena berbagai faktor seperti kurangnya curah hujan karena kemarau, pola iklim yang berubah dan lain sebagainya. Pada umumnya bencana kekeringan tidak dapat diketahui bagaimana awal mulanya, namun bisa dikatakan bahwa kekeringan terjadi ketika air yang ada sudah tidak lagi mampu mencukupi kebutuhan sehari-hari [3]. Kekeringan masih menjadi masalah di beberapa daerah di Indonesia salah satunya di wilayah Jawa Tengah. Menurut data dari Badan Nasional Penanggulangan Bencana (BNPB) pada tahun 2019 terdapat 7 desa/kelurahan yang masuk ke dalam kategori desa rawan bencana kekeringan kelas bahaya tinggi dan 2.809 desa/kelurahan yang masuk ke dalam kategori desa rawan bencana kekeringan kelas bahaya sedang di Provinsi Jawa Tengah [4].

Kekeringan merupakan salah satu permasalahan lingkungan hidup yang secara spesifik akan menimbulkan permasalahan yang serius bila terjadi dalam waktu yang berkepanjangan. Maka dari itu, diperlukan tindakan berkelanjutan dan bermanfaat sehingga dampak negatif dari kekeringan dapat ditekan seminimal mungkin. Salah satu solusi untuk mengatasi kekeringan adalah dengan membangun embung [5]. Embung merupakan waduk berukuran mikro di lahan pertanian (*small farm reservoir*) yang memiliki multifungsi serta dibangun untuk digunakan sebagai

pengendali kelebihan air ketika musim penghujan dan menjadi sumber air irigasi pada musim kemarau [6]. Dalam mengimplementasikan rencana pembangunan embung, sehubungan dengan banyaknya calon embung yang teridentifikasi sedangkan biaya yang tersedia terbatas maka tidak semua calon embung dapat dibangun dalam pembangunan 5 tahun anggaran. Oleh sebab itu perlu disusun prioritas pembangunan embung [2].

Pembangunan embung harus tepat guna dan efisien mulai dari lokasinya, anggarannya, serta dampak nyatanya nanti bila embung tersebut selesai dibangun. Maka dari itu perlu dilakukan pengkajian terhadap lokasi-lokasi alternatif untuk pembangunan embung dengan memperhatikan variabel-variabel tertentu. Dalam penentuan prioritas pembangunan embung terdapat 12 variabel berpengaruh yang dikelompokkan menjadi 5 faktor [2] yaitu:

1. Faktor topografi yang terdiri dari *abutmen* bukit tumpuan, volume material timbunan, dan luas daerah yang akan dibebaskan.
2. Faktor geologi yang hanya terdiri dari variabel jenis tanah dasar pondasi.
3. Faktor hidrologi yang terdiri dari variabel debit banjir rencana Q_{50th} , volume tampungan efektif, volume sedimen, dan luas DTA embung.
4. Faktor efektivitas yang terdiri dari lamanya waktu operasi yang dibutuhkan dan harga air/m³.
5. Faktor aksesibilitas yang terdiri dari jarak *quarry* dari lokasi ke *site* pembangunan embung jauh atau dekat dan akses jalan menuju *site* bendungan apakah tersedia jalan atau tidak.

Dari 12 variabel yang telah ada dipilih 7 variabel utama yang selanjutnya akan disebut dengan kriteria. Kriteria yang dipilih adalah vegetasi area genangan embung, volume material timbunan, luas daerah yang akan dibebaskan, volume tampungan efektif, lama operasi, harga air/m³, dan akses jalan menuju *site* bendungan. Kriteria-kriteria tersebut digunakan untuk menentukan prioritas pembangunan embung di 8 lokasi alternatif yang berada di beberapa kecamatan di Kabupaten Semarang. Lokasi dari ke-8 alternatif berada di Kecamatan Dadapayam, Mluweh, Lebak, Pakis, Jatikurung, Gogodalem, Kandangan, dan Ngrawan.

Penentuan prioritas lokasi embung dapat dilakukan dengan lebih mudah dan cepat jika terdapat suatu sistem pendukung keputusan yang dapat membantu pihak yang berwenang dalam mengambil kebijakan. Hasil dari sistem pendukung keputusan ini nantinya dapat menjadi bahan pertimbangan untuk menentukan prioritas lokasi pembangunan embung.

Dalam sistem pendukung keputusan penentuan lokasi embung pada penelitian ini menggunakan metode *Višekriterijumsko Kompromisno Rangiranje* (VIKOR) yang mana merupakan salah satu metode pengambilan keputusan multi kriteria atau yang lebih dikenal dengan istilah *Multi Criteria Decision Making* (MCDM). Metode ini berfokus pada peringkat dan pemilihan dari sekumpulan alternatif dengan kriteria yang saling bertentangan untuk dapat mengambil keputusan untuk mencapai keputusan akhir. Metode VIKOR sebelumnya pernah digunakan untuk menyelesaikan permasalahan serupa seperti pada penelitian yang berjudul *Implementasi Metode VIKOR untuk Seleksi Penerima Beasiswa* [7], serta pada penelitian berjudul *Multi-Criteria Decision Making with the VIKOR and SMARTER Methods for Optimal Seller Selection from Several E-Marketplaces* yang menggunakan metode VIKOR untuk memilih penjual yang optimal dari beberapa *marketplace* [8], dan pada penelitian berjudul *Multi-Criteria Optimization of Insulation Options for Warmth of Buildings to Increase Energy Efficiency* yang mana pada penelitian tersebut metode VIKOR digunakan untuk memilih material yang terbaik untuk digunakan sebagai material insulasi pada bangunan untuk memaksimalkan efisiensi energi yang digunakan [9]. Dari beberapa penelitian sebelumnya didapatkan kesimpulan bahwa metode VIKOR dapat diterapkan dan efektif untuk menyelesaikan suatu permasalahan yang multi kriteria dan dengan mengimplementasikan metode VIKOR ke dalam sebuah sistem informasi pendukung keputusan, proses penentuan prioritas embung terbaik di Kabupaten Semarang bisa dilakukan dengan efisien.

1.2 Rumusan Masalah

1. Bagaimana menggunakan metode *Višekriterijumsko Kompromisno Rangiranje* (VIKOR) untuk mengolah data dalam penentuan lokasi embung di Kabupaten Semarang?
2. Bagaimana cara mengimplementasikan metode *Višekriterijumsko Kompromisno Rangiranje* (VIKOR) ke dalam sebuah sistem pendukung keputusan?

1.3 Tujuan Tugas Akhir

Tujuan dari penelitian tugas akhir ini adalah membuat sebuah sistem pendukung keputusan penentuan prioritas pembangunan embung yang mengimplementasikan metode *Višekriterijumsko Kompromisno Rangiranje* (VIKOR) untuk mengolah data alternatif embung di Kabupaten Semarang.

1.4 Batasan Masalah

1. Pembuatan sistem pendukung keputusan penentuan lokasi embung dengan menggunakan metode *Višekriterijumsko Kompromisno Rangiranje* (VIKOR) digunakan untuk penentuan prioritas pembangunan embung di Kabupaten Semarang.
2. Pembuatan sistem pendukung keputusan penentuan lokasi embung dengan menggunakan metode *Višekriterijumsko Kompromisno Rangiranje* (VIKOR) di Kabupaten Semarang menampilkan data 8 lokasi alternatif embung, 7 kriteria, dan peta Kabupaten Semarang.
3. Pembuatan sistem pendukung keputusan penentuan lokasi embung dengan menggunakan metode *Višekriterijumsko Kompromisno Rangiranje* (VIKOR) menggunakan bahasa pemrograman PHP menggunakan *framework Codeigniter* dan dengan basis data MySQL.
4. Kriteria pada penelitian ini dibatasi hanya menggunakan 7 kriteria yaitu vegetasi area genangan embung, volume material timbunan, luas daerah yang akan dibebaskan, volume tampungan efektif, lama operasi, harga air/m³, dan akses jalan menuju *site* bendungan.

5. Sistem ini diujikan dalam jaringan lokal menggunakan *web browser*.

1.5 Manfaat Penelitian

Manfaat dari penelitian tugas akhir mengenai sistem pendukung keputusan penentuan lokasi embung dengan menggunakan metode *Višekriterijumsko Kompromisno Rangiranje* (VIKOR) di Kabupaten Semarang antara lain sebagai berikut:

1. Merancang sistem pendukung keputusan untuk penentuan lokasi embung di Kabupaten Semarang dalam bentuk sistem informasi.
2. Menampilkan data secara visual dalam bentuk sistem informasi untuk penentuan lokasi embung di Kabupaten Semarang.
3. Membantu instansi terkait untuk menentukan prioritas lokasi pembangunan embung Kabupaten Semarang dengan efektif dan efisien.

1.6 Metode Pengembangan Perangkat Lunak

Metode yang digunakan untuk mengembangkan perangkat lunak pada penelitian Tugas Akhir ini adalah metode *Rapid Application Development* (RAD).

1.7 Sistematika Penulisan

Laporan Tugas Akhir ini disusun menurut sistematika penulisan yang terdiri atas lima bab dengan susunan sebagai berikut:

BAB I PENDAHULUAN

Bab ini berisi tentang latar belakang, rumusan masalah, tujuan tugas akhir, batasan masalah, manfaat penelitian, metode pengembangan perangkat lunak, dan sistematika penulisan.

BAB II KAJIAN PUSTAKA

Bab ini berisi uraian mengenai penelitian terdahulu yang serupa, teknologi yang dipakai serta metode sistem pendukung keputusan yang diimplementasikan.

BAB III PERANCANGAN SISTEM

Bab ini berisi mengenai perancangan “Sistem Pendukung Keputusan Penentuan Lokasi Embung dengan Menggunakan Metode *Višekriterijumsko Kompromisno Rangiranje (VIKOR)*”.

BAB IV IMPLEMENTASI DAN PENGUJIAN SISTEM

Bab ini berisi tentang implementasi dan pengujian “Sistem Pendukung Keputusan Penentuan Lokasi Embung dengan Menggunakan Metode *Višekriterijumsko Kompromisno Rangiranje (VIKOR)*”. Pengujian sistem dilakukan dengan 2 dua jenis pengujian yaitu *Black Box Testing* yang meliputi fungsi-fungsi setiap komponen pada sistem ini telah berjalan dengan baik atau tidak dan pengujian performansi meliputi apakah sistem yang dibuat sudah sesuai dengan yang diharapkan dan pengujian *System Usability Scale (SUS)* meliputi pengujian pada kegunaan sistem.

BAB V PENUTUP

Bab ini berisi tentang kesimpulan mengenai sistem informasi yang telah dibuat apakah sudah sesuai dengan perancangannya serta saran untuk perbaikan maupun pengembangan di masa yang akan datang.

BAB II

KAJIAN PUSTAKA

2.1. Penelitian Terdahulu

Dalam mengerjakan penelitian ini, penelitian-penelitian terdahulu yang sudah dilakukan sebelumnya digunakan sebagai kajian serta referensi terhadap penelitian ini. Terdapat penelitian terdahulu dengan topik yang sama yaitu penelitian mengenai penentuan prioritas lokasi pembangunan embung, diantaranya penelitian yang dilakukan oleh Anjasmoro dkk [2], dengan judul Analisis Prioritas Pembangunan Embung Metode *Cluster Analysis*, AHP, dan *Weighted Average* (Studi Kasus: Embung di Kabupaten Semarang), penelitian tersebut menggunakan 3 metode yang berbeda untuk menganalisis penentuan embung prioritas, yaitu metode *Cluster Analysis*, AHP, dan *Weighted Average*. Kesimpulan dari penelitian tersebut adalah didapatkan variabel-variabel yang berpengaruh dalam pembangunan embung dengan metode *cluster analysis* metode non hierarki adalah vegetasi area genangan embung, volume material timbunan, luas daerah yang akan dibebaskan, volume tampungan efektif, lama operasi, harga air/m³, akses jalan masuk menuju site bendungan, status lahan di site dan genangan, biaya konstruksi embung, biaya OP, cakupan daerah irigasi, dan manfaat air baku [2].

Penelitian terdahulu lainnya mengenai embung adalah penelitian yang berjudul *Analysis of Fuzzy TOPSIS Method in Determining Priority of Small Dams Construction* yang dilakukan oleh Desyta Ulfiana dkk [10]. Penelitian tersebut dilakukan untuk mengetahui prioritas pembangunan bendungan kecil di Kabupaten Semarang menggunakan metode TOPSIS. Aspek teknis atau kriteria yang digunakan yaitu vegetasi di daerah genangan, volume material tanggul, area pembebasan lahan, penyimpanan hidup, reservoir seumur hidup, biaya air dan jalan akses ke situs bendungan. Untuk mengakomodasi jenis kriteria yang memiliki variabel linguistik, logika *fuzzy* digunakan untuk mengukur. Logika *fuzzy* kemudian diimplementasikan dalam metode TOPSIS sehingga analisis terbaik dapat diperoleh [10].

Penelitian terdahulu mengenai metode VIKOR antara lain penelitian yang dilakukan Salvius Paulus Lengkong dkk [7] yang berjudul Implementasi Metode VIKOR untuk Seleksi Penerima Beasiswa. Pada penelitian tersebut metode VIKOR digunakan untuk membantu proses seleksi penerimaan beasiswa dari 40 siswa berdasarkan kriteria-kriteria yang telah ditentukan yaitu: indeks prestasi (IP), semester, daya listrik dan tagihan listrik rumah [7].

Penelitian lain mengenai VIKOR juga pernah dilakukan oleh Miftahul Arif [8] dengan judul *Multi-Criteria Decision Making with the VIKOR and SMARTER Methods for Optimal Seller Selection from Several E-Marketplaces* yang menggunakan metode VIKOR untuk memilih penjual yang optimal dari beberapa *marketplace*. Selain menggunakan metode VIKOR, dalam penelitian ini juga menggunakan metode SMARTER yang digunakan untuk menentukan level prioritas tiap kriteria yang dibutuhkan untuk melakukan perhitungan ROC (*Rank Order Centroid*) yang mana dengan melakukan kuesioner kepada responden yang berkompeten [8].

Penelitian lainnya mengenai VIKOR adalah penelitian yang berjudul *Multi-Criteria Optimization of Insulation Options for Warmth of Buildings to Increase Energy Efficiency* [9]. Penelitian tersebut bertujuan untuk memilih material yang terbaik untuk digunakan sebagai material insulasi pada bangunan. Opsi alternatif bahan insulasi yang dipertimbangkan yaitu *styrofoam*, *mineral wool*, *pluto panels*, *polyester*, *polyurethane*, *perlite*, dan *wood wool* dengan kriteria yang ditentukan adalah harga bahan insulasi, emisi, koefisien konduktivitas termal, kalor spesifik, faktor ketahanan difusi uap air, dan kepadatan. Dalam penelitian ini metode VIKOR digunakan untuk mendapatkan material insulasi terbaik guna memaksimalkan efisiensi energi yang digunakan dan mengurangi biaya dan emisi CO₂. Dari penelitian ini didapatkan bahwa material *styrofoam* menjadi alternatif terbaik untuk digunakan sebagai material insulasi pada bangunan [9]. Metode VIKOR atau *Višekriterijumsko Kompromisno Rangiranje* yang di implementasikan ke dalam sebuah sistem pendukung keputusan diharapkan dapat digunakan untuk membantu menentukan solusi dari penentuan prioritas pembangunan embung di Kabupaten Semarang dengan efektif dan efisien.

2.2. Embung

Embung adalah salah satu teknologi konservasi air yang dibangun sebagai solusi/pemecah masalah kekeringan dengan menampung air hujan dan air limpasan (*run off*) sebagai cadangan persediaan pada musim kemarau. Embung merupakan waduk berukuran mikro yang dibangun untuk digunakan sebagai pengendali kelebihan air ketika musim penghujan dengan teknik pemanenan air (*water harvesting*) dan menjadi sumber air irigasi ketika kemarau. Embung juga memiliki tugas untuk mengumpulkan air dari Daerah Pengaliran Sungai (DPS) di bagian hulu yang berasal dari limpasan air hujan [6].

2.3. Sistem Pendukung Keputusan

Sistem Pendukung Keputusan (SPK) atau *Decision Support System* (DSS) ialah serangkaian kelas tertentu dari sistem informasi terkomputerisasi yang mendukung kegiatan pengambilan keputusan. Aplikasi Sistem Pendukung Keputusan (SPK) mengimplementasikan sistem informasi berbasis komputer atau *Computer Based Information Systems* (CBIS) untuk membantu menyelesaikan masalah manajemen tidak terstruktur melalui penyediaan solusi yang efektif [11].

Sistem Pendukung Keputusan (SPK) merupakan sebuah sistem yang mampu memberikan kemampuan pemecahan masalah, sistem ini dibuat untuk meningkatkan produktivitas dengan melakukan proses pembuatan keputusan secara otomatis [12].

Karakteristik dari sistem pendukung keputusan [11] antara lain :

- a. Membantu dalam proses pembuatan keputusan bagi organisasi atau perusahaan.
- b. Adanya antarmuka/*interface* agar mudah dipahami bagi pengguna dengan pengguna tetap memegang kontrol terhadap proses pengambilan keputusan.
- c. Membantu dalam pengambilan keputusan untuk membahas masalah terstruktur dan semi terstruktur serta mendukung beberapa keputusan yang saling terkait dan berinteraksi.

- d. Memiliki kapasitas dialog untuk mengumpulkan informasi yang sesuai dengan kebutuhan.
- e. Memiliki sub-sistem yang terintegrasi dengan baik sehingga dapat bekerja sebagai kesatuan sistem yang utuh.
- f. Memiliki dua buah komponen utama yaitu data dan model.

Tujuan dari sistem pendukung keputusan [11] adalah:

1. Membantu memberikan solusi dalam pengambilan keputusan pada suatu masalah.
2. Menyediakan dukungan atas pertimbangan yang dilakukan oleh manajer tanpa bertujuan untuk menggantikan fungsi manajer.
3. Meningkatkan efektivitas keputusan diambil daripada perbaikan efisiensinya.
4. Kecepatan proses perhitungan oleh komputer memungkinkan pengambil keputusan melakukan banyak perhitungan secara cepat dengan biaya yang ekonomis.
5. Peningkatan produktivitas dengan membangun suatu kelompok pengambil keputusan terutama pakar bisa sangat mahal. Namun dengan sistem pendukung keputusan komputerisasi, ukuran kelompok bisa dikurangi dan anggotanya dapat berada di lokasi yang berbeda-beda, sehingga memperkecil biaya perjalanan.

2.4. Višekriterijumsko Kompromisno Rangiranje (VIKOR)

VIKOR atau *Višekriterijumsko Kompromisno Rangiranje* dalam bahasa Serbia yang berarti “perangkingan kompromis multi-kriteria” merupakan salah satu metode pengambilan keputusan multikriteria. Landasan dari solusi kompromi dalam VIKOR dibuat oleh Yu (1973) dan Zeleny (1982) kemudian diteruskan oleh Opricovic dan Tzeng (2002, 2003, 2004, dan 2007) [13]. Metode VIKOR merupakan metode *Multi-Criteria Decision Making* (MCDM) yang telah digunakan secara luas untuk menyelesaikan berbagai macam pengambilan keputusan berdasarkan banyak kriteria dengan mengajukan solusi kompromi berdasarkan solusi ideal yang diperkirakan. Metode VIKOR mampu mengatasi

kriteria yang bertentangan dalam melakukan proses perangkingan, maksud dari kriteria bertentangan adalah tiap kriteria dapat menggunakan penilaian berbeda dengan kriteria yang lain yakni kriteria dapat menggunakan tren *benefit* (semakin besar nilainya maka semakin baik) atau tren *cost* (semakin kecil nilainya maka semakin baik). Metode VIKOR sendiri memiliki kelemahan dalam melakukan pembobotan kriteria karena tidak ada perhitungan khusus untuk menghitung nilai bobot setiap kriteria, pembobotan kriteria dalam metode VIKOR diberikan begitu saja oleh pengambil keputusan sehingga diperlukan metode lain untuk memeriksa konsistensi bobot [8].

Langkah-langkah yang dilakukan dalam penyelesaian masalah menggunakan metode VIKOR [7][8] adalah sebagai berikut:

1. Menyusun Matriks Keputusan (F)

Setiap alternatif dan kriteria disusun ke dalam bentuk matriks keputusan F . A_i menyatakan alternatif ke 1,2,3, ..., i dan C_{xj} menyatakan kriteria ke 1,2,3, ..., j dan x_{ij} menyatakan respons alternatif i pada kriteria j .

$$F = \begin{matrix} & \begin{matrix} C_{x1} & C_{x2} & \cdots & C_{xj} \end{matrix} \\ \begin{matrix} A_1 \\ A_2 \\ \vdots \\ A_i \end{matrix} & \begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1j} \\ x_{21} & x_{22} & \cdots & x_{2j} \\ \vdots & \vdots & \ddots & \vdots \\ x_{i1} & x_{i2} & \cdots & x_{ij} \end{pmatrix} \end{matrix} \quad (2.1)$$

Keterangan:

F = Matriks keputusan

A_i = Alternatif ke - i

C_j = Kriteria ke - j

x_{ij} = Respons alternatif i pada kriteria j

i = 1,2,3, ..., i adalah nomor urutan alternatif

j = 1,2,3, ..., j adalah nomor urutan kriteria

2. Menentukan Bobot Kriteria (W)

Menentukan bobot kriteria yang diperoleh dari pengguna sistem sesuai dengan kebutuhan atau kriteria yang diinginkan. Rumusan umum untuk bobot kriteria adalah berlaku persamaan:

$$\sum_{j=1}^n W_j = 1 \quad (2.2)$$

Keterangan:

W_j = Bobot kriteria j

3. Membuat Matriks Normalisasi (N)

Membuat matriks normalisasi dengan menentukan nilai positif (f_j^+) dan nilai negatif (f_j^-) sebagai solusi ideal untuk setiap kriteria. Penentuan nilai data terbaik/positif (f_j^+) dan terburuk/negatif (f_j^-) atau dengan istilah *Cost* dan *Benefit* ditentukan oleh jenis data kriteria apakah *higher-the-better* (HB) atau *lower-the-better* (LB). Untuk menentukan nilai positif dan nilai negatif masing-masing kriteria digunakan persamaan berikut:

- Jika kriteria memiliki tren *benefit*, maka gunakan fungsi berikut:

$$f_j^+ = \max(f_{1j}, f_{2j}, f_{3j}, \dots, f_{ij}) \quad (2.3)$$

$$f_j^- = \min(f_{1j}, f_{2j}, f_{3j}, \dots, f_{ij}) \quad (2.4)$$

- Jika kriteria memiliki tren *cost*, maka gunakan fungsi berikut:

$$f_j^+ = \min(f_{1j}, f_{2j}, f_{3j}, \dots, f_{ij}) \quad (2.5)$$

$$f_j^- = \max(f_{1j}, f_{2j}, f_{3j}, \dots, f_{ij}) \quad (2.6)$$

Selanjutnya melakukan normalisasi pada matriks F dengan persamaan berikut:

$$N_{ij} = \frac{(f_j^+ - f_{ij})}{(f_j^+ - f_j^-)} \quad (2.7)$$

Keterangan:

N = Matriks ternormalisasi

f_{ij} = Fungsi respons alternatif i pada kriteria j

f_j^+ = Nilai terbaik dalam satu kriteria j

f_j^- = Nilai terjelek dalam satu kriteria j

4. Menghitung Normalisasi Bobot (F^*)

Menentukan nilai terbobot dari data ternormalisasi untuk setiap alternatif terhadap kriteria dengan melakukan perkalian antara nilai data ternormalisasi (N_{ij}) dengan nilai bobot kriteria (W_j) yang telah ditentukan, dengan persamaan sebagai berikut:

$$F_{ij}^* = W_j \cdot N_{ij} \quad (2.8)$$

Keterangan:

F_{ij}^* = Nilai data ternormalisasi sudah terbobot untuk alternatif i pada kriteria j

W_j = Nilai bobot kriteria j

N_{ij} = Nilai data ternormalisasi untuk alternatif i pada kriteria j

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

Menghitung nilai *utility measure* (S) dan *regret measure* (R) untuk setiap alternatif yang mana nilai S_i menyatakan nilai jarak alternatif ke solusi ideal positif sedangkan R_i menyatakan nilai jarak alternatif ke solusi ideal negatif. Untuk menghitung S_i dan R_i digunakan persamaan berikut:

$$S_i = \sum_{j=1}^n F_{ij}^* \quad (2.9)$$

$$R_i = \max_j [F_{ij}^*] \quad (2.10)$$

Keterangan:

S_i = Nilai *Utility Measure* untuk alternatif ke - i

R_i = Nilai *Regret Measure* untuk alternatif ke - i

F_{ij}^* = Nilai data ternormalisasi sudah terbobot untuk alternatif i pada kriteria j

6. Menghitung Nilai Indeks VIKOR (Q)

Menghitung nilai indeks VIKOR (Q) untuk setiap alternatif dengan menggunakan nilai S_i , S^+ , S^- , R_i , R^+ , dan R^- yang didapat dari perhitungan *utility measures* dan *regret measure* serta nilai V yang merupakan bobot yang nilainya antara 0-1 (umumnya bernilai 0.5). Nilai V merupakan nilai bobot

strategy of the maximum group sedangkan nilai $(1 - V)$ adalah bobot *individual regret*. Semakin kecil nilai indeks VIKOR (Q_i), maka semakin baik pula solusi alternatif tersebut. Untuk mencari nilai Q_i digunakan persamaan berikut:

$$Q_i = V \left[\frac{(S_i - S^-)}{(S^+ - S^-)} \right] + (1 - V) \left[\frac{(R_i - R^-)}{(R^+ - R^-)} \right] \quad (2.11)$$

Keterangan:

Q_i = Nilai Indeks VIKOR alternatif

V = Bobot berkisar antara 0-1 (umumnya bernilai 0.5)

$S^+ = \max_i(S_i)$

$S^- = \min_i(S_i)$

$R^+ = \max_i(R_i)$

$R^- = \min_i(R_i)$

7. Perangkingan Alternatif

Setelah menghitung nilai Q_i , maka terdapat 3 perangkingan: S_i , R_i , dan Q_i . Solusi kompromi dilihat pada perangkingan Q_i . Perangkingan ditentukan dari nilai Q_i yang paling rendah sebagai solusi ideal.

8. Mengajukan Solusi Kompromi

Solusi kompromi ditentukan dari alternatif yang memiliki peringkat terbaik dengan mengukur indeks VIKOR yang minimum dengan mengujinya dengan 2 kondisi berikut:

- Kondisi 1: *Acceptable Advantage*

Menghitung selisih antara peringkat alternatif pertama dan kedua yakni $Q_{(a_1)}$ dan $Q_{(a_2)}$ lalu membandingkannya dengan nilai DQ . Jika nilai selisih yang didapat lebih besar atau sama dengan nilai DQ , maka kondisi *acceptable advantage* terpenuhi. Persamaan dari kondisi dapat dituliskan sebagai berikut:

$$Q_{(a_2)} - Q_{(a_1)} \geq DQ \quad (2.12)$$

$$DQ = \frac{1}{m - 1} \quad (2.13)$$

Keterangan:

$Q_{(a_2)}$ = Alternatif peringkat kedua

$Q_{(a_1)}$ = Alternatif peringkat pertama

m = jumlah alternatif

- Kondisi 2: *Acceptable Stability in Decision Making*

Menguji stabilitas perankingan alternatif dengan menggunakan nilai V yang berbeda yakni: nilai $V > 0,5$ (*voting by majority rule*), nilai $V = 0,5$ (*by consensus*), dan nilai $V < 0,5$ (*with veto*). Jika alternatif peringkat pertama atau $Q_{(a_1)}$ tetap menjadi peringkat terbaik dalam 3 macam pemeringkatan dengan nilai V yang berbeda, maka kondisi *acceptable stability in decision making* terpenuhi.

Jika salah satu kondisi tidak terpenuhi, maka solusi kompromi dapat diajukan sebagai berikut:

- Jika hanya kondisi 2 yang tidak terpenuhi, maka memilih alternatif peringkat pertama dan kedua atau $Q_{(a_1)}$ dan $Q_{(a_2)}$.
- Jika kondisi 1 tidak terpenuhi, maka memilih alternatif $Q_{(a_1)}, Q_{(a_2)}, \dots, Q_{(a_m)}$. Dimana alternatif $Q_{(a_m)}$ ditentukan dengan persamaan berikut:

$$Q_{(a_m)} - Q_{(a_1)} < DQ \quad (2.14)$$

$$DQ = \frac{1}{m-1} \quad (2.13)$$

Keterangan:

$Q_{(a_m)}$ = Alternatif m maksimum yang berada dalam kondisi berdekatan

m = jumlah alternatif

2.5. Basis Data

Basis data terdiri dari dua kata yaitu basis dan data. Basis dapat diartikan sebagai suatu markas atau gudang, tempat bersarang atau tempat berkumpul. Data dapat diartikan merupakan representasi dari fakta dunia yang mewakili suatu objek

(manusia, barang, peristiwa, keadaan, dan sebagainya) yang direkam dalam bentuk angka, huruf, simbol, teks, gambar, bunyi atau kombinasinya. Bisa disimpulkan basis data adalah himpunan kelompok data (arsip) yang saling berhubungan yang diorganisasi sedemikian rupa agar kelak dapat dimanfaatkan kembali dengan cepat dan mudah [14].

Basis data merupakan salah satu komponen yang sangat penting dalam sebuah sistem informasi. Basis data adalah kumpulan data yang saling berhubungan yang disimpan secara bersama sedemikian rupa dan tanpa pengulangan (*redundancy*) yang tidak perlu, untuk memenuhi berbagai kebutuhan. Basis data memiliki tujuan untuk mengatur data sehingga didapatkan kecepatan dan kemudahan (*speed*), efisiensi ruang penyimpanan (*space*), keakuratan (*accuracy*), ketersediaan (*availability*), kelengkapan (*completeness*), keamanan (*security*), dan kebersamaan (*sharability*) [14].

2.6. CodeIgniter

Codeigniter adalah sebuah aplikasi *open source* yang berupa *framework* atau kerangka kerja yang digunakan untuk membangun *website* dengan bahasa pemrograman PHP. Tujuan penggunaan *framework* adalah untuk mempercepat pengembangan proyek dengan menyediakan *library* yang dapat digunakan dibanding penulisan kode dasar atau kode terstruktur. Codeigniter relatif mudah digunakan dan dipelajari karena memiliki *interface* yang sederhana, dokumentasi yang lengkap, dan *library* atau kumpulan *tools* yang dapat digunakan untuk membuat sebuah proses kerja untuk *website* yang dibuat [15].

Codeigniter menggunakan pola desain MVC (*Model, View, Controller*). MVC merupakan suatu konsep yang memisahkan pengembangan aplikasi berdasarkan komponen utama yang membangun sebuah aplikasi, yakni [16]:

1. *Model*: bagian yang berhubungan langsung dengan basis data untuk memanipulasi data (*insert, update, delete, search*), menangani validasi dari *controller*, namun tidak berhubungan langsung dengan *view*.
2. *View*: bagian yang menangani *presentation logic*, pada suatu aplikasi web biasanya berupa file HTML. *View* berfungsi untuk menerima dan

merepresentasikan data kepada pengguna. Bagian ini tidak memiliki akses langsung terhadap *model*.

3. *Controller*: bagian yang mengatur hubungan antara bagian *model* dan bagian *view*. Controller berfungsi untuk menerima permintaan (*request*) dan data dari pengguna.

Beberapa keuntungan yang bisa didapatkan dari penggunaan Codeigniter antara lain sebagai berikut [17]:

- Codeigniter berukuran kecil, cepat, sederhana, dan mudah dipelajari.
- Mudah saat melakukan migrasi dari satu server ke server yang lain, cukup hanya mengubah *URL*.
- Proses instalasi yang mudah.
- Dokumentasi yang lengkap dan dukungan komunitas yang sudah kuat.
- Koleksi pustaka atau *library* yang tersedia sudah banyak.

2.7. Rapid Application Development

Rapid Application Development (RAD) atau *rapid prototyping* adalah suatu model pembangunan perangkat lunak yang tergolong dalam teknik inkremental atau bertingkat. *Rapid Application Development* (RAD) menggunakan metode iteratif atau berulang dalam mengembangkan sistem dimana model kerja (*working model*) sistem dikonstruksikan diawal tahap pengembangan dengan tujuan menetapkan kebutuhan (*requirement*) pengguna. Model kerja hanya digunakan sesekali sebagai basis desain dan implementasi sistem akhir [18].

Rapid Application Development (RAD) merupakan metode yang menguntungkan setiap pengguna dalam melakukan pengembangan sistem, hal tersebut dinyatakan pada permintaan pengguna yang harus dipenuhi oleh pengguna dan juga mengadopsi konsep iterasi dan *prototype*. Pada RAD terdapat metode yang bersifat sekuensial linear, diartikan pengembangannya memakan waktu relatif singkat dan hasil perangkat lunak yang sudah dikembangkan memiliki kualitas cukup baik [19].

Kelebihan menggunakan *Rapid Application Development* (RAD) adalah sebagai berikut [20]:

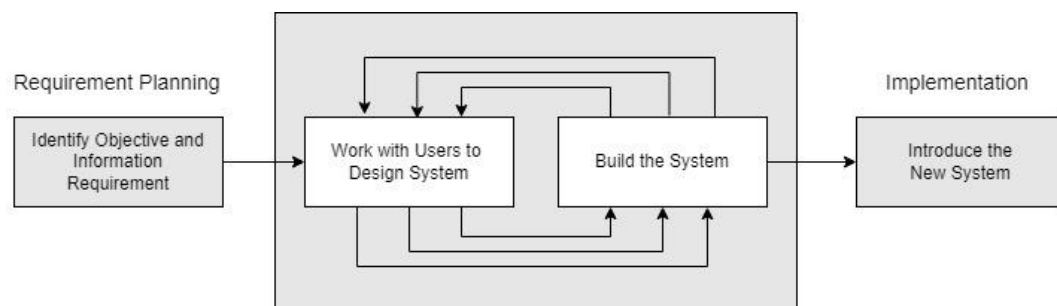
- Kemudahan dalam implementasi
- Aplikasi dikembangkan berdasarkan kebutuhan dan keinginan pengguna
- Waktu pengembangan aplikasi relatif cepat dan efisien.

Kekurangan menggunakan *Rapid Application Development* (RAD) adalah sebagai berikut [20]:

- Waktu pengembangannya yang cepat justru dapat menghasilkan produk dengan kualitas yang buruk
- Membutuhkan tim pengembangan yang berpengalaman
- Dibutuhkan kontrol dan manajemen proyek yang bagus

Berikut adalah bagian-bagian yang terdapat dalam *Rapid Application Development* (RAD) [19]:

- Requirement planning*: ketentuan-ketentuan yang didapatkan dari hasil analisis kebutuhan.
- Desain RAD: tahapan pengguna ikut terlibat dalam perancangan sistem di setiap fase.
- Implementasi: sebuah proses menerapkan setiap bagian pada sebuah sistem.



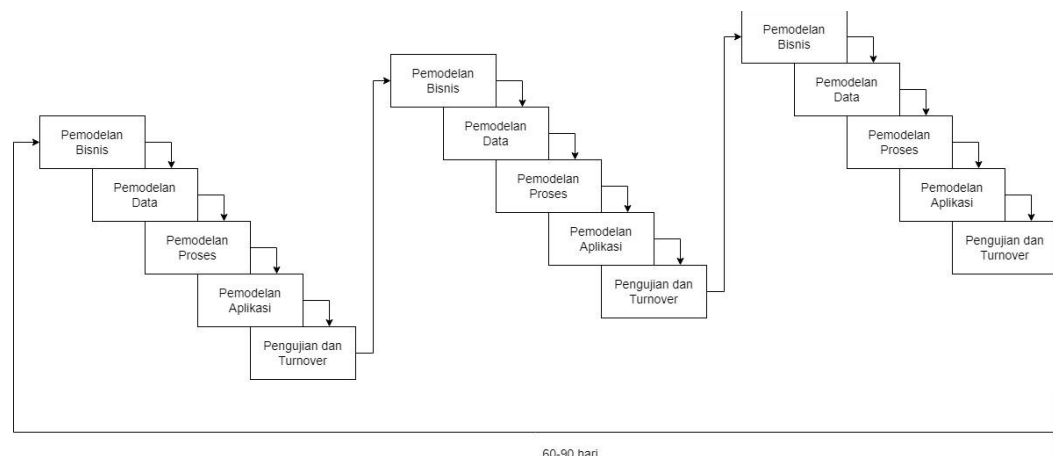
Gambar 2.1 Tahapan *Rapid Application Development* (RAD)

Berikut adalah tahap yang terkait pada *Rapid Application Development* (RAD) [19]:

- Bussiness Modelling* bertujuan untuk menyelesaikan permasalahan dari sistem yang sedang dikembangkan dalam proses bisnis pada penelitian yang sedang berlangsung.
- Data Modelling* menghasilkan objek dari informasi yang sudah dikumpulkan yakni objek yang memiliki hubungan dengan informasi yang diolah. Pada pemodelan objek tersebut akan dilakukan implementasi dalam

berbagai database dan objek-objek yang ada memiliki rangkaian atribut masing- masing.

- c. *Process Modelling* merupakan langkah yang harus dilakukan oleh peneliti guna memodelkan setiap kebutuhan pada sebuah sistem melalui berbagai perancangan.
- d. *Application Generation* merupakan implementasi yang dilakukan pada kebutuhan dan perancangan yang sudah didefinisikan kemudian akan dibuat kode program dan basis data.
- e. *Testing and Turnover* adalah tahap penting yang dilakukan pada akhir penelitian dengan menguji fungsional sistem supaya tidak terjadi *error* pada sistem, jika sistem sudah melewati tahap pengujian maka sistem dapat digunakan oleh pengguna.



Gambar 2.2 Model *Rapid Application Development* (RAD)

BAB III

PERANCANGAN SISTEM

Pada bab ini akan dijelaskan tahap perancangan pada sistem pendukung keputusan penentuan lokasi embung dengan menggunakan metode *Višekriterijumsko Kompromisno Rangiranje (VIKOR)* dengan menggunakan metode pengembangan aplikasi *Rapid Application Development (RAD)* [21]. Bab ini akan menjelaskan lebih khusus pada tahap analisis dan perancangan sistem atau desain.

3.1 Analisis

Pada tahap ini dilakukan analisis data yang sesuai dengan kondisi yang ada di Kabupaten Semarang terkait dengan sistem yang dikembangkan. Proses pengambilan data dilakukan dengan pemanfaatan dari penelitian Dosen Departemen Teknik Sipil, Fakultas Teknik, Universitas Diponegoro dalam penelitian berjudul *Analysis of Fuzzy TOPSIS Method in Determining Priority of Small Dams Construction* [10] dan penelitian lain berjudul Analisis Prioritas Pembangunan Embung Metode *Cluster Analysis*, AHP dan *Weighted Average* (Studi Kasus: Embung di Kabupaten Semarang) [2] yang kemudian disesuaikan dengan kebutuhan sistem pendukung keputusan.

3.1.1 Analisis Situasi

Kabupaten Semarang merupakan salah satu kabupaten di Provinsi Jawa Tengah dengan ibu kotanya adalah Kota Ungaran. Kabupaten Semarang secara geografis terletak pada 110°14'54,75" sampai dengan 110°39'3" Bujur Timur dan 7°3'57" sampai dengan 7°30' Lintang Selatan. Kabupaten ini berbatasan dengan Kota Semarang di utara; Kabupaten Demak dan Kabupaten Grobogan di timur, Kabupaten Boyolali di timur dan selatan, Kota Salatiga di tengah Kabupaten Semarang, serta Kabupaten Magelang, Kabupaten Temanggung, dan Kabupaten Kendal di barat. Kabupaten Semarang terdiri atas 19 kecamatan, yang dibagi lagi atas 208 desa dan 27 kelurahan. Kabupaten Semarang mempunyai sebuah lembaga

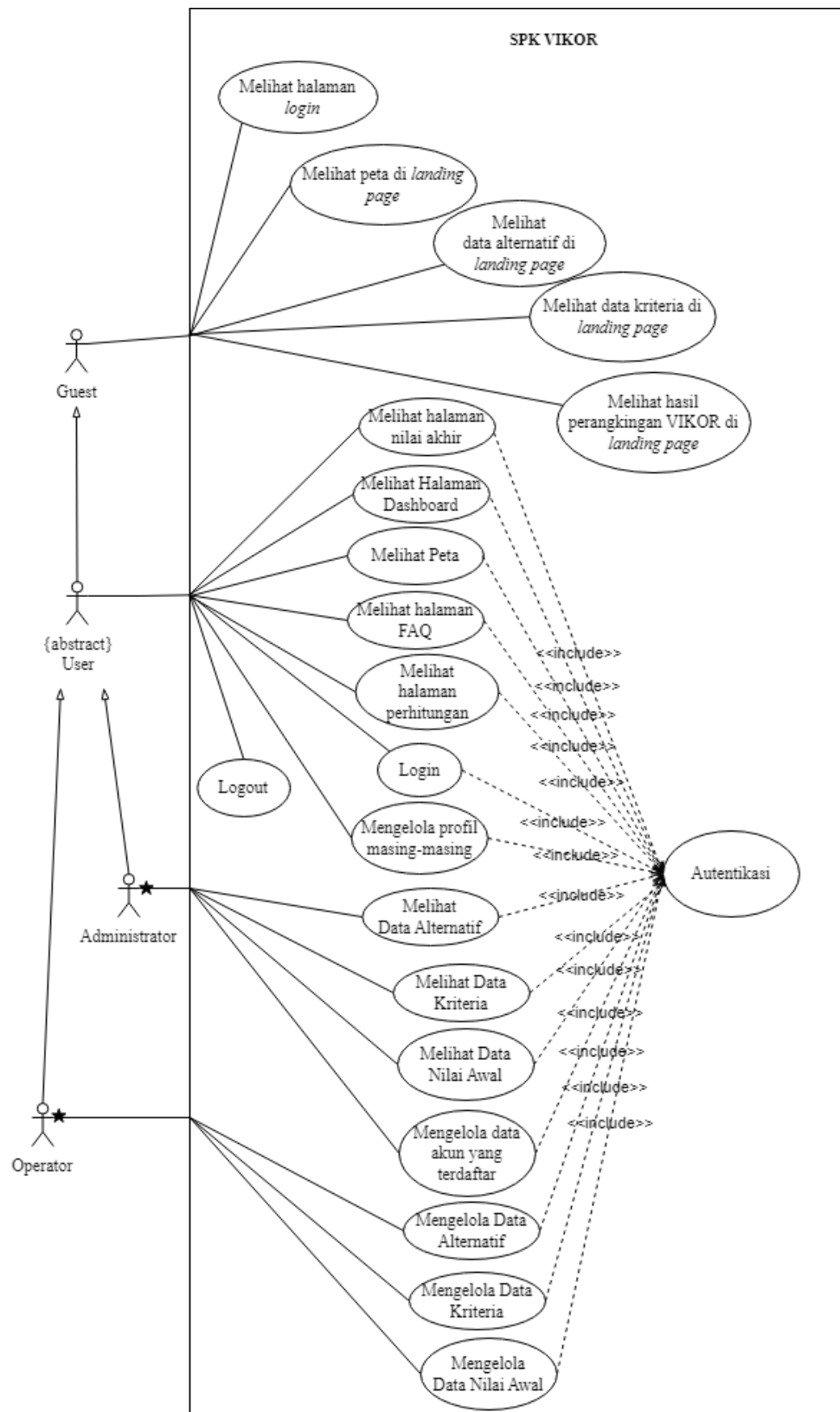
teknis daerah di bidang penelitian dan perencanaan pembangunan daerah atau yang disingkat BAPPEDA (Badan Perencanaan Pembangunan Daerah) yang dipimpin oleh seorang kepala badan yang berada di bawah dan bertanggung jawab kepada gubernur/bupati/wali kota melalui sekretaris daerah. Kabupaten Semarang sedang berupaya untuk meningkatkan jumlah embung di daerahnya, sedangkan dana yang dimiliki terbatas sehingga perlu adanya penentuan prioritas pembangunan embung dengan menggunakan sistem pendukung keputusan (SPK) sehingga diperoleh lokasi embung yang terbaik.

3.1.2 Kebutuhan Pengguna

Sistem yang dibutuhkan ialah sebuah sistem informasi penentuan lokasi embung yang dapat melakukan pengolahan data terkait meliputi mengolah data alternatif lokasi embung, mengolah data kriteria, dan mengolah nilai kriteria tiap alternatif sehingga dari perhitungan data-data di atas diperoleh urutan prioritas lokasi pembangunan embung yang ditampilkan dalam bentuk tabel.

Pada sistem pendukung keputusan ini terdapat 3 tingkatan pengguna, yaitu administrator, operator, dan *guest*. Administrator dapat melihat dan mengubah *role* akun terdaftar, melihat peta, melihat data alternatif, melihat data kriteria, melihat data nilai awal, nilai *v*, perhitungan, dan juga nilai akhir. Untuk operator dapat melihat, membuat, memperbarui, dan menghapus data alternatif, kriteria, nilai awal setiap alternatif terhadap kriteria dan nilai *v* serta melihat peta, perhitungan, dan juga nilai akhir sedangkan *guest* hanya bisa melihat data alternatif, data kriteria, hasil perangkingan, dan peta di halaman awal/*landing page* saja. Untuk administrator dan operator perlu melakukan *login* pada sistem untuk dilakukan *session authentication* sebelum memasuki sistem, untuk *guest* tidak perlu melakukan *login* karena hanya dapat melihat data yang hanya ditampilkan di halaman awal.

Berdasarkan analisis kebutuhan di atas, maka dapat dijelaskan lebih lanjut melalui diagram *use case*. Diagram ini akan mendeskripsikan pemodelan *use case* yang dimaksudkan untuk menunjukkan hubungan antara fungsi yang dapat dilakukan aktor dalam sistem. *Use case* sistem ini dapat dilihat pada gambar 3.1.



Gambar 3.1 Diagram use case

3.1.3 Deskripsi Use Case

Deskripsi *use-case* menjelaskan masing-masing fungsi komponen *use case* pada gambar 3.1 yang dijelaskan pada tabel 3.1 – tabel 3.20.

1. Prosedur proses melihat data alternatif di *landing page*

Tabel 3.1 Deskripsi *use case* melihat data alternatif di *landing page*

<i>Use Case Name</i>	Melihat data alternatif di <i>landing page</i>	
<i>Use Case Description</i>	Proses melihat data alternatif di <i>landing page</i> yang dapat dilakukan oleh semua pengguna.	
<i>Actors</i>	Guest, Operator, Administrator	
<i>Pre-conditions</i>	Menampilkan halaman awal/ <i>landing page</i> untuk melihat data alternatif	
<i>Post-Condition</i>	Menampilkan data alternatif yang terdapat pada section SPK VIKOR pada <i>landing page</i> secara <i>modal table</i> .	
<i>Main Scenario</i>	<i>Serial No</i>	<i>Steps</i>
<i>Actors/Guest</i>	1	Buka halaman utama (<i>landing page</i>) melalui URL.
	2	Klik menu Data pada <i>navigation bar</i> SPK VIKOR
	3	Klik button ‘Lihat Data’ pada <i>section</i> Alternatif, data akan ditampilkan secara <i>modal table</i> .
<i>Extensions</i>	1a	Sistem tidak dapat dibuka karena kesalahan input URL.

2. Prosedur proses melihat data kriteria di *landing page*

Tabel 3.2 Deskripsi *use case* melihat data kriteria di *landing page*

<i>Use Case Name</i>	Melihat data kriteria di <i>landing page</i>	
<i>Use Case Description</i>	Proses melihat data kriteria di <i>landing page</i> yang dapat dilakukan oleh semua pengguna.	
<i>Actors</i>	Guest, Operator, Administrator	
<i>Pre-conditions</i>	Menampilkan halaman awal/ <i>landing page</i> untuk melihat data kriteria.	
<i>Post-Condition</i>	Menampilkan data kriteria yang terdapat pada <i>section</i> SPK VIKOR pada <i>landing page</i> secara <i>modal table</i> .	
<i>Main Scenario</i>	<i>Serial No</i>	<i>Steps</i>
<i>Actors/Guest</i>	1	Buka halaman utama (<i>landing page</i>) melalui URL.
	2	Klik menu Data pada <i>navigation bar</i> SPK VIKOR
	3	Klik button ‘Lihat Data’ pada <i>section</i> Kriteria, data akan ditampilkan secara <i>modal table</i> .
<i>Extensions</i>	1a	Sistem tidak dapat dibuka karena kesalahan input URL.

3. Prosedur proses melihat hasil perangkingan di *landing page*

Tabel 3.3 Deskripsi *use case* melihat hasil perangkingan di *landing page*

<i>Use Case Name</i>	Melihat data perangkingan di <i>landing page</i>	
<i>Use Case Description</i>	Proses melihat data perangkingan di <i>landing page</i> yang dapat dilakukan oleh semua pengguna.	
<i>Actors</i>	Guest, Operator, Administrator	
<i>Pre-conditions</i>	Menampilkan halaman awal/ <i>landing page</i> untuk melihat data perangkingan.	
<i>Post-Condition</i>	Menampilkan data perangkingan yang terdapat pada <i>section</i> SPK VIKOR pada <i>landing page</i> secara <i>modal table</i> .	
<i>Main Scenario</i>	<i>Serial No</i>	<i>Steps</i>
<i>Actors/Guest</i>	1	Buka halaman utama (<i>landing page</i>) melalui <i>URL</i> .
	2	Klik menu Rank pada <i>navigation bar</i> SPK VIKOR
	3	Klik button ‘Lihat Semua’ pada <i>section</i> Hasil Perangkingan, data akan ditampilkan secara <i>modal table</i> .
<i>Extensions</i>	1a	Sistem tidak dapat dibuka karena kesalahan <i>input URL</i> .

4. Prosedur proses melihat peta di *landing page*

Tabel 3.4 Deskripsi *use case* melihat peta di *landing page*

<i>Use Case Name</i>	Melihat peta di <i>landing page</i>	
<i>Use Case Description</i>	Proses melihat peta di <i>landing page</i> yang dapat dilakukan oleh semua pengguna.	
<i>Actors</i>	Guest, Operator, Administrator	
<i>Pre-conditions</i>	Menampilkan halaman awal/ <i>landing page</i> untuk melihat peta.	
<i>Post-Condition</i>	Menampilkan peta yang terdapat pada <i>section</i> SPK VIKOR pada <i>landing page</i> .	
<i>Main Scenario</i>	<i>Serial No</i>	<i>Steps</i>
<i>Actors/Guest</i>	1	Buka halaman utama (<i>landing page</i>) melalui <i>URL</i> .
	2	Klik menu Sistem pada <i>navigation bar</i> SPK VIKOR
<i>Extensions</i>	1a	Sistem tidak dapat dibuka karena kesalahan <i>input URL</i> .

5. Prosedur proses melihat halaman masuk (*login*)

Tabel 3.5 Deskripsi *use case* melihat halaman masuk (*login*)

<i>Use Case Name</i>	Melihat halaman masuk (<i>login</i>)
<i>Use Case Description</i>	Proses melihat halaman masuk (<i>login</i>) yang digunakan untuk masuk ke dalam sistem dengan menggunakan akun yang sudah terdaftar.
<i>Actors</i>	Guest, Operator, Administrator

<i>Pre-conditions</i>	Menampilkan halaman awal/ <i>landing page</i> yang terdapat <i>button login</i> pada <i>navigation bar</i>	
<i>Post-Condition</i>	Menampilkan halaman masuk (<i>login</i>) yang akan digunakan untuk melakukan proses fungsi <i>login</i> .	
Main Scenario	Serial No	Steps
<i>Actors/Guest</i>	1	Buka halaman utama (<i>landing page</i>) melalui <i>URL</i> .
	2	Klik <i>button Login</i> pada <i>navigation bar</i> SPK VIKOR
<i>Extensions</i>	1a	Sistem tidak dapat dibuka karena kesalahan <i>input URL</i> atau <i>button “Login”</i> di <i>landing page</i> tidak berfungsi.

6. Prosedur proses masuk (*login*)

Tabel 3.6 Deskripsi *use case* proses masuk (*login*)

Use Case Name	Proses masuk (<i>login</i>)	
<i>Use Case Description</i>	Proses masuk (<i>login</i>) ke dalam sistem dengan melakukan autentikasi data dengan menggunakan akun yang sudah terdaftar.	
<i>Actors</i>	Operator dan Administrator	
<i>Pre-conditions</i>	Menampilkan halaman masuk (<i>login</i>) yang memuat form alamat <i>email</i> dan <i>password</i> dan tombol “ <i>login</i> ” untuk proses fungsi <i>login</i> ke dalam sistem.	
<i>Post-Condition</i>	Menampilkan halaman <i>dashboard</i> sesuai dengan akun role yang melakukan <i>login</i> : administrator atau operator.	
Main Scenario	Serial No	Steps
<i>Actors</i>	1	Memasukkan alamat <i>email</i> dan <i>password</i> .
	2	Autentikasi alamat <i>email</i> dan <i>password</i> ..
	3	Menekan tombol <i>login</i> dan pengguna masuk ke dalam sistem sesuai dengan data akun yang digunakan untuk masuk.
<i>Extensions</i>	1a	<i>Email</i> atau <i>password</i> yang dimasukkan salah/tidak sesuai dengan data yang ada di <i>database</i> .
	2b	Form <i>email</i> dan <i>password</i> tidak diisi saat melakukan <i>login</i>

7. Prosedur proses melihat halaman *dashboard*

Tabel 3.7 Deskripsi *use case* melihat halaman *dashboard*

Use Case Name	Proses melihat halaman <i>dashboard</i>
<i>Use Case Description</i>	Proses menampilkan halaman <i>dashboard</i> setelah berhasil melakukan <i>login</i> .
<i>Actors</i>	Operator dan Administrator
<i>Pre-conditions</i>	Melakukan fungsi <i>login</i> sebagai operator atau administrator untuk masuk ke dalam sistem.
<i>Post-Condition</i>	Menampilkan halaman <i>dashboard</i> sesuai dengan akun <i>role</i> yang melakukan <i>login</i> : administrator atau operator.

<i>Main Scenario</i>	<i>Serial No</i>	<i>Steps</i>
<i>Actors</i>	1	Klik menu <i>dashboard</i> pada <i>sidebar</i> dalam SPK VIKOR.
<i>Extensions</i>	1a	Data tidak dapat ditampilkan karena kesalahan sistem.
	2b	<i>Session login</i> akun telah kedaluwarsa karena tidak ada aktivitas dalam rentang waktu tertentu sehingga sistem melakukan <i>logout</i> otomatis pada akun.

8. Prosedur proses melihat data alternatif

Tabel 3.8 Deskripsi *use case* melihat data alternatif

<i>Use Case Name</i>	Proses melihat data alternatif	
<i>Use Case Description</i>	Proses menampilkan halaman alternatif didalam SPK VIKOR.	
<i>Actors</i>	Administrator	
<i>Pre-conditions</i>	Melakukan fungsi <i>login</i> sebagai administrator untuk masuk ke dalam sistem dan membuka halaman data alternatif.	
<i>Post-Condition</i>	Menampilkan halaman alternatif.	
<i>Main Scenario</i>	<i>Serial No</i>	<i>Steps</i>
<i>Actors</i>	1	Klik menu data - alternatif pada <i>sidebar</i> dalam SPK VIKOR.
<i>Extensions</i>	1a	Data alternatif tidak dapat ditampilkan karena kesalahan sistem.
	2b	<i>Session login</i> akun telah kedaluwarsa karena tidak ada aktivitas dalam rentang waktu tertentu sehingga sistem melakukan <i>logout</i> otomatis pada akun.

9. Prosedur proses melihat data kriteria

Tabel 3.9 Deskripsi *use case* melihat data kriteria

<i>Use Case Name</i>	Proses melihat data kriteria	
<i>Use Case Description</i>	Proses menampilkan halaman kriteria didalam SPK VIKOR.	
<i>Actors</i>	Administrator	
<i>Pre-conditions</i>	Melakukan fungsi <i>login</i> sebagai administrator untuk masuk ke dalam sistem dan membuka halaman data kriteria.	
<i>Post-Condition</i>	Menampilkan halaman kriteria serta parameternya.	
<i>Main Scenario</i>	<i>Serial No</i>	<i>Steps</i>
<i>Actors</i>	1	Klik menu data - kriteria pada <i>sidebar</i> dalam SPK VIKOR.
<i>Extensions</i>	1a	Data kriteria dan parameter tidak dapat ditampilkan karena kesalahan sistem.
	2b	<i>Session login</i> akun telah kedaluwarsa karena tidak ada aktivitas dalam rentang waktu tertentu sehingga sistem melakukan <i>logout</i> otomatis pada akun.

10. Prosedur proses melihat data nilai awal

Tabel 3.10 Deskripsi *use case* melihat data nilai awal

<i>Use Case Name</i>	Proses melihat data nilai awal	
<i>Use Case Description</i>	Proses menampilkan halaman nilai awal didalam SPK VIKOR.	
<i>Actors</i>	Administrator	
<i>Pre-conditions</i>	Melakukan fungsi <i>login</i> sebagai administrator untuk masuk ke dalam sistem dan membuka halaman data nilai awal.	
<i>Post-Condition</i>	Menampilkan halaman nilai awal semua alternatif.	
<i>Main Scenario</i>	<i>Serial No</i>	<i>Steps</i>
<i>Actors</i>	1	Klik menu nilai awal pada <i>sidebar</i> dalam SPK VIKOR.
<i>Extensions</i>	1a	Data nilai awal tidak dapat ditampilkan karena kesalahan sistem.
	2b	<i>Session login</i> akun telah kedaluwarsa karena tidak ada aktivitas dalam rentang waktu tertentu sehingga sistem melakukan <i>logout</i> otomatis pada akun.

11. Prosedur proses mengelola data alternatif

Tabel 3.11 Deskripsi *use case* mengelola data alternatif

<i>Use Case Name</i>	Proses mengelola data alternatif	
<i>Use Case Description</i>	Proses yang terdiri dari melihat, menambah, memperbarui, dan menghapus data alternatif didalam SPK VIKOR.	
<i>Actors</i>	Operator	
<i>Extend Use Case</i>	Tambah alternatif, edit alternatif, dan hapus alternatif.	
<i>Pre-conditions</i>	Melakukan fungsi <i>login</i> sebagai operator untuk masuk ke dalam sistem dan membuka halaman data alternatif.	
<i>Post-Condition</i>	Lihat data: Menampilkan seluruh data alternatif.	
	Tambah data: Menampilkan halaman tambah alternatif dan berhasil menambahkan data alternatif.	
	Edit data: Menampilkan halaman edit alternatif dan berhasil mengubah data alternatif.	
	Hapus data: Data alternatif terpilih berhasil dihapus.	
<i>Main Scenario</i>	<i>Serial No</i>	<i>Steps</i>
<i>Actors</i>	1	Klik menu data - alternatif pada <i>sidebar</i> dalam SPK VIKOR.
	2	Tambah data: <ul style="list-style-type: none"> – Klik <i>icon</i> tambah alternatif – Masukkan nama, kecamatan, <i>latitude</i>, <i>longitude</i> alternatif serta isikan nilai alternatif terhadap tiap kriteria – Klik <i>button</i> “Simpan”

	3	Edit data: <ul style="list-style-type: none"> – Klik <i>icon</i> "edit" pada alternatif yang ingin diubah – Ubah data yang ingin diperbarui – Klik <i>button</i> "Simpan"
	4	Hapus data: <ul style="list-style-type: none"> – Klik <i>icon</i> "hapus" pada alternatif yang ingin diubah – Klik "hapus" pada <i>pop-up</i> konfirmasi.
<i>Extensions</i>	1a	Data alternatif tidak dapat ditampilkan, ditambah, diubah atau dihapus karena kesalahan sistem.
	2b	<i>Session login</i> akun telah kedaluwarsa karena tidak ada aktivitas dalam rentang waktu tertentu sehingga sistem melakukan <i>logout</i> otomatis pada akun.

12. Prosedur proses mengelola data kriteria

Tabel 3.12 Deskripsi *use case* mengelola data kriteria

<i>Use Case Name</i>	Proses mengelola data kriteria	
<i>Use Case Description</i>	Proses yang terdiri dari melihat, menambah, memperbarui, dan menghapus data kriteria didalam SPK VIKOR.	
<i>Actors</i>	Operator	
<i>Extend Use Case</i>	Tambah kriteria, edit kriteria, dan hapus kriteria.	
<i>Pre-conditions</i>	Melakukan fungsi <i>login</i> sebagai operator untuk masuk ke dalam sistem dan membuka halaman data kriteria.	
<i>Post-Condition</i>	Lihat data: Menampilkan seluruh data kriteria dan parameter.	
	Tambah data: Menampilkan halaman tambah kriteria dan berhasil menambahkan data kriteria.	
	Edit data: Menampilkan halaman edit kriteria dan berhasil mengubah data kriteria.	
	Hapus data: Data kriteria terpilih berhasil dihapus.	
<i>Main Scenario</i>	<i>Serial No</i>	<i>Steps</i>
<i>Actors</i>	1	Klik menu data - kriteria pada <i>sidebar</i> dalam SPK VIKOR.
	2	Tambah data: <ul style="list-style-type: none"> – Klik <i>icon</i> tambah kriteria. – Pilih jenis kriteria yang ingin ditambahkan: berparameter atau tidak berparameter. – Masukkan jumlah parameter jika kriteria yang diitambahkan memiliki parameter. – Masukkan nama, bobot, tren kriteria serta isikan parameternya jika ada. – Klik <i>button</i> "Simpan"
	3	Edit data: <ul style="list-style-type: none"> – Klik <i>icon</i> "edit" pada kriteria yang ingin diubah – Ubah data yang ingin diperbarui

		– Klik <i>button</i> “Simpan”
	4	Hapus data: – Klik <i>icon</i> "hapus" pada kriteria yang ingin dihapus – Klik “hapus” pada <i>pop-up</i> konfirmasi.
<i>Extensions</i>	1a	Data kriteria dan parameter tidak dapat ditampilkan, ditambah, diubah atau dihapus karena kesalahan sistem.
	2b	<i>Session login</i> akun telah kedaluwarsa karena tidak ada aktivitas dalam rentang waktu tertentu sehingga sistem melakukan <i>logout</i> otomatis pada akun.

13. Prosedur proses mengelola data nilai awal

Tabel 3.13 Deskripsi *use case* mengelola data nilai awal

<i>Use Case Name</i>	Proses mengelola data nilai awal	
<i>Use Case Description</i>	Proses yang terdiri dari melihat dan memperbarui data nilai awal alternatif terhadap kriteria dan nilai V didalam SPK VIKOR.	
<i>Actors</i>	Operator	
<i>Extend Use Case</i>	Edit nilai alternatif dan edit nilai V.	
<i>Pre-conditions</i>	Melakukan fungsi <i>login</i> sebagai operator untuk masuk ke dalam sistem dan membuka halaman nilai awal.	
<i>Post-Condition</i>	Lihat data: Menampilkan seluruh data nilai awal alternatif.	
	Edit data: Menampilkan halaman edit nilai awal dan berhasil mengubah data nilai awal.	
<i>Main Scenario</i>	<i>Serial No</i>	<i>Steps</i>
<i>Actors</i>	1	Klik menu nilai awal pada <i>sidebar</i> dalam SPK VIKOR.
	2	Edit data nilai awal alternatif: – Klik <i>icon</i> "edit" pada alternatif yang ingin diubah – Ubah data nilai alternatif yang ingin diperbarui – Klik <i>button</i> “Simpan”
	2	Edit data nilai V: – Klik <i>icon</i> "edit" pada <i>section</i> nilai V – Ubah nilai V ₁ , V ₂ , dan V ₃ – Klik <i>button</i> “Edit”
<i>Extensions</i>	1a	Data nilai awal tidak dapat ditampilkan atau diubah karena kesalahan sistem.
	2b	<i>Session login</i> akun telah kedaluwarsa karena tidak ada aktivitas dalam rentang waktu tertentu sehingga sistem melakukan <i>logout</i> otomatis pada akun.

14. Prosedur proses mengelola profil masing-masing

Tabel 3.14 Deskripsi *use case* mengelola profil masing-masing

<i>Use Case Name</i>	Proses mengelola profil masing-masing	
<i>Use Case Description</i>	Proses yang terdiri dari melihat dan memperbarui data profil masing-masing meliputi <i>username</i> , foto profil, dan <i>password</i> didalam SPK VIKOR.	
<i>Actors</i>	Operator	
<i>Pre-conditions</i>	Melakukan fungsi <i>login</i> sebagai operator untuk masuk ke dalam sistem dan membuka halaman profil melalui <i>dropdown menu</i> di pojok kanan atas <i>website</i> .	
<i>Post-Condition</i>	Lihat data: Menampilkan data profil masing-masing.	
	Edit data: Menampilkan pop-up form untuk memperbarui <i>username</i> , foto profil, dan <i>password</i> .	
<i>Main Scenario</i>	<i>Serial No</i>	<i>Steps</i>
<i>Actors</i>	1	Klik <i>dropdown menu</i> di pojok kanan atas <i>website</i> , pilih lihat profil.
	2	Klik <i>icon</i> "edit" untuk memperbarui <i>username</i> dan foto profil lalu klik <i>button</i> "Edit" untuk menyimpan.
	3	Klik <i>icon</i> "edit password" untuk memperbarui <i>password</i> lalu klik <i>button</i> "Edit" untuk menyimpan.
<i>Extensions</i>	1a	Data profil masing-masing tidak dapat ditampilkan atau diubah karena kesalahan sistem.
	2b	<i>Session login</i> akun telah kedaluwarsa karena tidak ada aktivitas dalam rentang waktu tertentu sehingga sistem melakukan <i>logout</i> otomatis pada akun.

15. Prosedur proses mengelola data akun terdaftar

Tabel 3.15 Deskripsi *use case* mengelola data akun terdaftar

<i>Use Case Name</i>	Proses mengelola data akun terdaftar	
<i>Use Case Description</i>	Proses yang terdiri dari melihat, menambah, menghapus, dan memperbarui data akun didalam SPK VIKOR.	
<i>Actors</i>	Administrator	
<i>Extend Use Case</i>	Tambah akun, edit <i>role</i> akun, dan hapus akun.	
<i>Pre-conditions</i>	Melakukan fungsi <i>login</i> sebagai administrator untuk masuk ke dalam sistem dan membuka halaman akun.	
<i>Post-Condition</i>	Lihat data: Menampilkan data seluruh akun terdaftar.	
	Tambah data: Menampilkan <i>pop-up form</i> untuk menambahkan akun baru.	
	Edit data: Menampilkan halaman edit akun untuk mengubah <i>role</i> akun tersebut.	
<i>Main Scenario</i>	<i>Serial No</i>	<i>Steps</i>
<i>Actors</i>	1	Klik menu akun pada <i>sidebar</i> dalam SPK VIKOR.

	2	Tambah data: – Klik <i>icon</i> tambah operator. – Masukkan email, nama, dan <i>password</i> baru. – Klik <i>button</i> “Tambah”
	3	Edit data: – Klik <i>icon</i> "edit" pada akun yang ingin diubah – Ubah <i>role</i> akun pada data terpilih – Klik <i>button</i> “Simpan”
	4	Hapus data: – Klik <i>icon</i> "hapus" pada akun yang ingin dihapus – Klik “hapus” pada <i>pop-up</i> konfirmasi.
<i>Extensions</i>	1a	Data akun tidak dapat ditampilkan atau diubah karena kesalahan sistem.
	2b	<i>Session login</i> akun telah kedaluwarsa karena tidak ada aktivitas dalam rentang waktu tertentu sehingga sistem melakukan <i>logout</i> otomatis pada akun.

16. Prosedur proses melihat halaman perhitungan

Tabel 3.16 Deskripsi *use case* melihat halaman perhitungan

<i>Use Case Name</i>	Proses melihat halaman perhitungan	
<i>Use Case Description</i>	Proses menampilkan halaman perhitungan didalam SPK VIKOR.	
<i>Actors</i>	Operator dan Administrator	
<i>Pre-conditions</i>	Melakukan fungsi <i>login</i> untuk masuk ke dalam sistem dan membuka halaman perhitungan.	
<i>Post-Condition</i>	Menampilkan halaman perhitungan.	
<i>Main Scenario</i>	<i>Serial No</i>	<i>Steps</i>
<i>Actors</i>	1	Klik menu perhitungan pada <i>sidebar</i> dalam SPK VIKOR.
<i>Extensions</i>	1a	Data perhitungan tidak dapat ditampilkan karena kesalahan sistem.
	2b	<i>Session login</i> akun telah kedaluwarsa karena tidak ada aktivitas dalam rentang waktu tertentu sehingga sistem melakukan <i>logout</i> otomatis pada akun.

17. Prosedur proses melihat halaman nilai akhir

Tabel 3.17 Deskripsi *use case* melihat halaman nilai akhir

<i>Use Case Name</i>	Proses melihat halaman nilai akhir
<i>Use Case Description</i>	Proses menampilkan halaman nilai akhir didalam SPK VIKOR.
<i>Actors</i>	Operator dan Administrator
<i>Pre-conditions</i>	Melakukan fungsi <i>login</i> untuk masuk ke dalam sistem dan membuka halaman nilai akhir.
<i>Post-Condition</i>	Menampilkan halaman perhitungan.

<i>Main Scenario</i>	<i>Serial No</i>	<i>Steps</i>
<i>Actors</i>	1	Klik menu nilai akhir pada <i>sidebar</i> dalam SPK VIKOR.
<i>Extensions</i>	1a	Data nilai akhir perhitungan tidak dapat ditampilkan karena kesalahan sistem.
	2b	<i>Session login</i> akun telah kedaluwarsa karena tidak ada aktivitas dalam rentang waktu tertentu sehingga sistem melakukan <i>logout</i> otomatis pada akun.

18. Prosedur proses melihat halaman peta

Tabel 3.18 Deskripsi *use case* melihat halaman peta

<i>Use Case Name</i>	Proses melihat halaman peta	
<i>Use Case Description</i>	Proses menampilkan halaman peta didalam SPK VIKOR.	
<i>Actors</i>	Operator dan Administrator	
<i>Pre-conditions</i>	Melakukan fungsi <i>login</i> untuk masuk ke dalam sistem dan membuka halaman peta.	
<i>Post-Condition</i>	Menampilkan halaman peta.	
<i>Main Scenario</i>	<i>Serial No</i>	<i>Steps</i>
<i>Actors</i>	1	Klik menu peta pada <i>sidebar</i> dalam SPK VIKOR.
<i>Extensions</i>	1a	Data peta tidak dapat ditampilkan karena kesalahan sistem.
	2b	<i>Session login</i> akun telah kedaluwarsa karena tidak ada aktivitas dalam rentang waktu tertentu sehingga sistem melakukan <i>logout</i> otomatis pada akun.

19. Prosedur proses melihat halaman FAQ

Tabel 3.19 Deskripsi *use case* melihat melihat halaman FAQ

<i>Use Case Name</i>	Proses melihat halaman FAQ	
<i>Use Case Description</i>	Proses menampilkan halaman FAQ didalam SPK VIKOR.	
<i>Actors</i>	Operator dan Administrator	
<i>Pre-conditions</i>	Melakukan fungsi <i>login</i> untuk masuk ke dalam sistem dan membuka halaman FAQ.	
<i>Post-Condition</i>	Menampilkan halaman perhitungan.	
<i>Main Scenario</i>	<i>Serial No</i>	<i>Steps</i>
<i>Actors</i>	1	Klik <i>dropdown menu</i> di pojok kanan atas website, pilih FAQ.
<i>Extensions</i>	1a	halaman FAQ tidak dapat ditampilkan karena kesalahan sistem.
	2b	<i>Session login</i> akun telah kedaluwarsa karena tidak ada aktivitas dalam rentang waktu tertentu sehingga sistem melakukan <i>logout</i> otomatis pada akun.

20. Prosedur proses keluar (*logout*)

Tabel 3.20 Deskripsi *use case* proses keluar (*logout*)

<i>Use Case Name</i>	Proses keluar (<i>logout</i>)	
<i>Use Case Description</i>	Proses keluar (<i>logout</i>) dari sistem.	
<i>Actors</i>	Operator dan Administrator	
<i>Pre-conditions</i>	Melakukan fungsi <i>login</i> untuk masuk ke dalam SPK VIKOR.	
<i>Post-Condition</i>	Akun berhasil keluar dari sistem dan kembali halaman <i>landing page</i> .	
<i>Main Scenario</i>	<i>Serial No</i>	<i>Steps</i>
<i>Actors</i>	1	Klik <i>dropdown menu</i> di pojok kanan atas website, pilih Keluar.
<i>Extensions</i>	1a	Sistem tidak dapat merespons perintah keluar (<i>logout</i>)

3.1.4 Analisis Kebutuhan Sistem

Dalam memahami fungsi sistem informasi penentuan lokasi embung ini, analisis yang dilakukan adalah pemahaman tentang informasi dan perilaku yang diperlukan dengan mengacu terhadap analisis kebutuhan sistem. Pada sistem ini menggunakan metode VIKOR, pada metode VIKOR akan melakukan beberapa tahap perhitungan untuk mendapatkan nilai indeks VIKOR tiap alternatif untuk menentukan peringkat prioritas pembangunan embung. Tahap perhitungan dari metode VIKOR yaitu; membuat matriks keputusan (F), menentukan bobot kriteria (W), membuat matriks normalisasi (N), membuat normalisasi bobot (F*), mencari nilai *utility measure* (S) dan *regret measure* (R), menghitung nilai indeks (Q) hingga didapatkan perangkingan alternatif lalu menentukan solusi kompromi dengan melakukan 2 buah pengujian sehingga didapat konklusi peringkat embung terbaik.

3.1.5 Kebutuhan Non-Fungsional

Kebutuhan non-fungsional adalah kebutuhan spesifikasi sistem yang akan diimplementasikan meliputi komponen-komponennya, sehingga untuk menjalankan sistem ini diperlukan perangkat lunak sebagai berikut:

a. Windows OS

Sistem ini dapat berjalan di sistem operasi yang memiliki browser dan web server lokal. Sistem Operasi yang digunakan untuk pengembangan dan implementasi sistem ini adalah Windows 11.

b. *Local Web Server*

Sistem pendukung keputusan ini memerlukan *local web server* atau web service yang menyediakan Apache dan MySQL. *Local Web Server* yang digunakan pada implementasi sistem ini adalah XAMPP.

c. *Browser*

Sistem ini dapat dijalankan menggunakan berbagai *browser* seperti Firefox, Google Chrome, Microsoft Edge ataupun Safari. *Browser* yang digunakan pada implementasi sistem ini adalah Google Chrome.

Pada tahap ini juga dilakukan instalasi terhadap semua kebutuhan non-fungsional perangkat lunak sesuai dengan bagian yang telah disediakan. Berikut ini merupakan bagian untuk menentukan kebutuhan non-fungsional yang ditunjukkan pada tabel 3.21.

Tabel 3.21 Kebutuhan non-fungsional sistem

KN-F	Parameter	Kebutuhan
KN-F-01	<i>Portability</i>	Fitur dan fungsi yang terdapat dalam sistem dapat berfungsi dengan baik dan benar.
KN-F-02	<i>Usability</i>	Sistem memiliki tampilan atau interface dan experience yang mudah dipahami dan digunakan oleh pengguna.
KN-F-03	<i>Reliability:</i> Autentikasi	Sistem ini melakukan proses autentikasi pada saat proses <i>login</i> berlangsung guna melakukan validasi terhadap pengguna yang ingin masuk ke dalam sistem serta melakukan pengecekan terhadap role atau hak akses yang dimiliki.
KN-F-04	<i>Reliability: Login</i>	Sistem ini menggunakan proses <i>login</i> sebagai pintu masuk untuk pengguna yang telah terdaftar untuk dapat masuk ke dalam sistem.
KN-F-05	<i>Flexibility</i>	Sistem ini dapat berubah sesuai dengan kebutuhan.
KN-F-06	<i>Supportability:</i> Komunikasi	Bahasa Indonesia dan Bahasa Inggris

3.2 Perancangan (Desain)

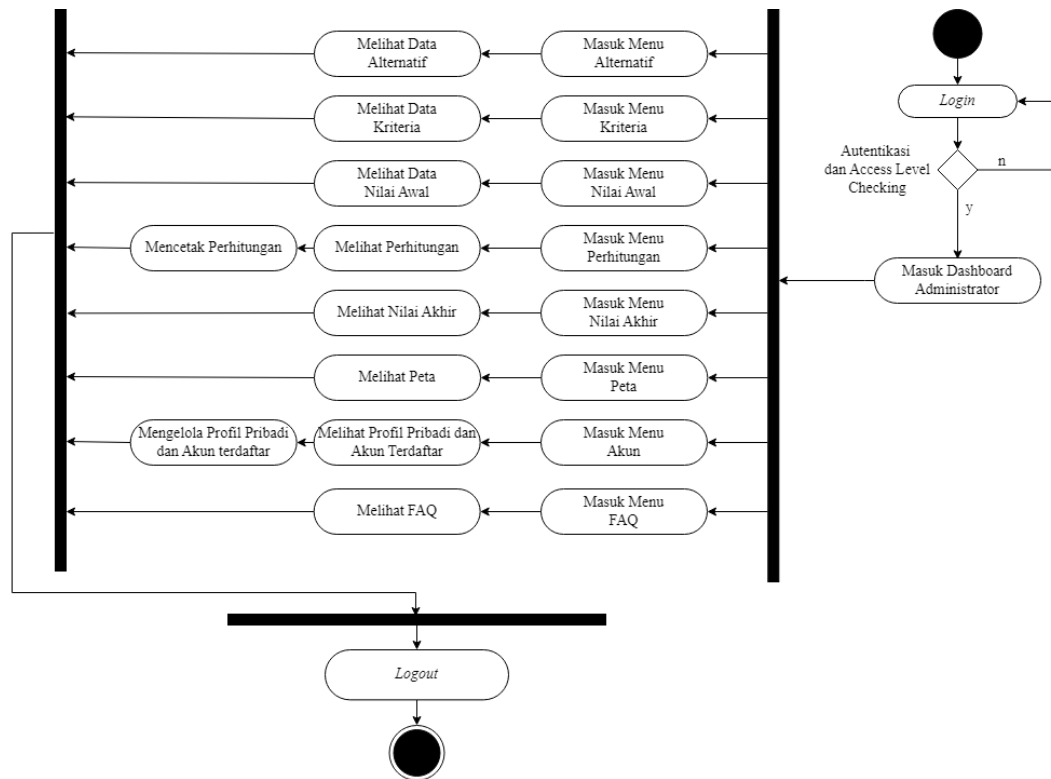
Tahap ini merupakan tahap perancangan perangkat lunak yang terdiri dari perancangan proses kerja (*business process*), perancangan aktivitas diagram

(*diagram activity*), perancangan basis data (*database design*), dan perancangan antarmuka (*user interface design*).

3.2.1 Perancangan Proses Kerja (*Bussiness Process*)

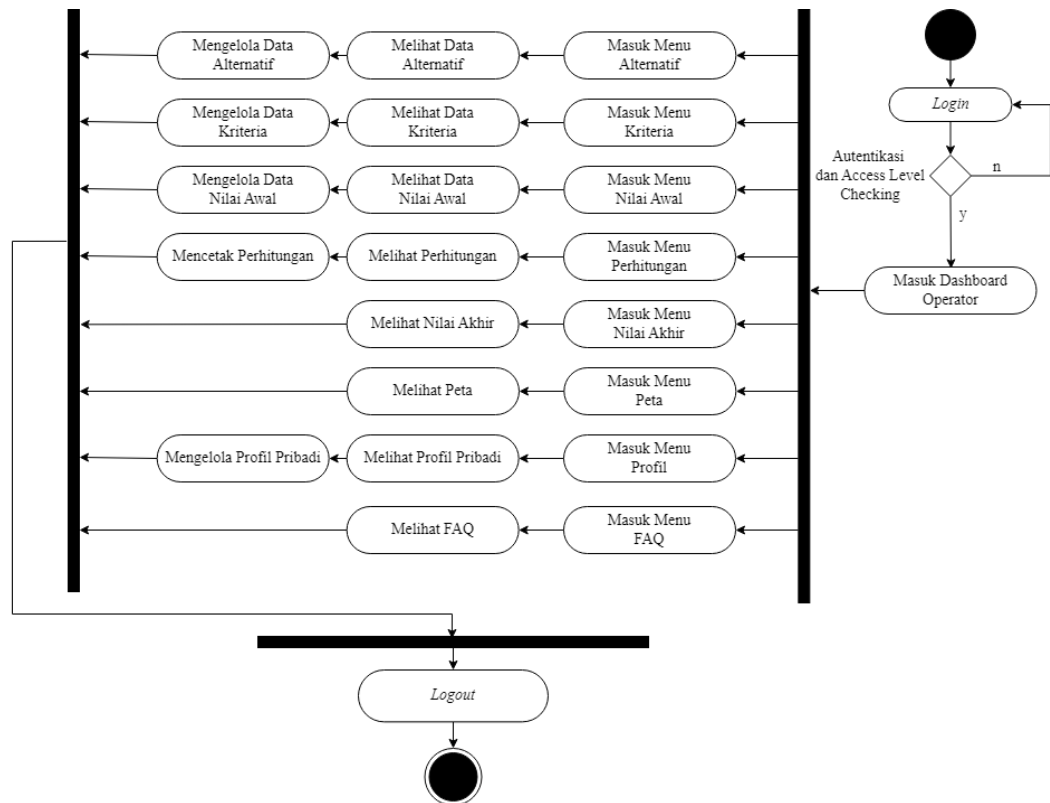
Dalam desain perangkat lunak, perancangan proses kerja perlu dibangun terlebih dahulu karena merupakan perancangan mendasar dari perilaku dan aktivitas sistem yang terjadi ketika sistem dijalankan. Perilaku dan aktivitas sistem tersebut digambarkan dalam bentuk *activity diagram*. Permulaan dari sistem ini dimulai dengan halaman *landing page* yang bisa diakses oleh siapapun dan berisi daftar kriteria dan alternatif serta hasil akhir perangkungan dan visualisasi peta. Kemudian proses *login* dengan melakukan cek autentikasi pengguna, jika gagal akan kembali pada halaman *login* dan jika berhasil autentikasi akan masuk pada *dashboard* sesuai dengan *role* nya masing-masing. Pada menu *dashboard* akan ditampilkan hasil akhir rangking dari metode VIKOR serta visualisasi peta lokasi alternatif. Untuk *role* administrator akan menampilkan menu pada *sidebar* yaitu *dashboard*, data alternatif, data kriteria, nilai awal, perhitungan, nilai akhir, peta, dan akun akan tetapi administrator hanya dapat melakukan *create*, *update*, dan *delete* pada data akun saja di menu akun. Untuk *role* operator akan menampilkan menu pada *sidebar* yaitu *dashboard*, data alternatif, data kriteria, nilai awal, perhitungan, nilai akhir, dan peta. Operator dapat melakukan *create*, *read*, *update*, dan *delete* pada data alternatif, kriteria, nilai awal, dan nilai v di halaman masing-masing data. Untuk *role guest* hanya menampilkan data alternatif, data kriteria, dan hasil akhir tabel perangkungan di halaman awal/*landing page* dikarenakan *guest* tidak melakukan *login* ke dalam sistem. Menu yang berada didalam sistem yang terdapat proses *create*, *read*, *update* dan *delete* telah ditentukan hak aksesnya sesuai dengan *role* yang sudah ditentukan pada *use case diagram*. Di dalam sistem terdapat juga halaman *frequently asked question* (FAQ) yang berisi sejumlah pertanyaan umum mengenai sistem pendukung keputusan VIKOR beserta dengan jawabannya, halaman ini dapat diakses oleh administrator dan operator.

Aktivitas yang dilakukan oleh administrator meliputi masuk ke halaman *login*, masuk ke dalam *dashboard*, melihat data alternatif, melihat data kriteria, mengelola data akun pengguna, melihat *map*, melihat nilai awal, melihat *FAQ*, melihat dan mencetak proses perhitungan VIKOR, dan melihat nilai akhir. Aktivitas sistem yang dapat dilakukan oleh administrator ditunjukkan pada gambar 3.2.



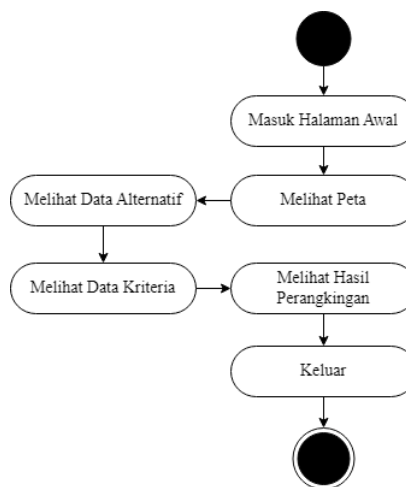
Gambar 3.2 Diagram activity administrator

Aktivitas yang dilakukan oleh operator meliputi masuk ke halaman *login*, masuk ke dalam *dashboard*, mengelola data alternatif, mengelola data kriteria, mengelola data nilai awal, melihat *map*, melihat *FAQ*, mengelola akun pribadi, melihat dan mencetak proses perhitungan VIKOR, dan melihat nilai akhir. Aktivitas sistem yang dapat dilakukan oleh operator ditunjukkan pada gambar 3.3.



Gambar 3.3 Diagram activity operator

Aktivitas yang dilakukan oleh *guest* hanya sebatas melihat data yang ditampilkan di halaman awal yaitu hanya data alternatif, data kriteria, hasil perbandingan, dan *map*. Aktivitas sistem yang dapat dilakukan oleh *guest* ditunjukkan pada gambar 3.4.



Gambar 3.4 Diagram activity guest

3.2.2 Perancangan Basis Data (*Database Design*)

Basis data merupakan komponen dasar dalam pengembangan sistem pendukung keputusan yang menggambarkan kebutuhan sistem. Proses perancangan basis data terdapat enam langkah [22]:

1. Pengumpulan Data dan Analisis

a. Analisis Kebutuhan Sistem

Dalam memahami fungsi sistem informasi ini, analisis yang dilakukan adalah pemahaman tentang informasi dan perilaku yang diperlukan dengan mengacu terhadap analisis kebutuhan sistem. Pada tahap ini dilakukan analisa sistem informasi yang akan berinteraksi dengan sistem basis data yaitu aktor dari sistem ini sendiri. Sistem ini dibangun dan digunakan untuk tiga aktor yaitu administrator, operator, dan *guest*. Deskripsi dari masing-masing tingkatan pengguna (*role*) ditunjukkan pada tabel 3.22.

Tabel 3.22 Tingkatan pengguna (*role*) pada sistem

No	Aktor	Deskripsi
1	Administrator	Administrator memiliki akses melihat data alternatif, melihat data kriteria, melihat data nilai awal, mengelola akun pribadi dan akun terdaftar, melihat <i>map</i> , melihat FAQ, melihat dan mencetak perhitungan.
2	Operator	Operator memiliki hak akses mengelola data alternatif, mengelola data kriteria, mengelola data nilai awal, mengelola data akun pribadi, melihat <i>map</i> , melihat FAQ, serta melihat dan mencetak perhitungan.
3	<i>Guest</i>	<i>Guest</i> hanya memiliki hak akses untuk melihat data alternatif, data kriteria, melihat hasil perangkingan, melihat <i>map</i> di halaman awal.

b. Kebutuhan Fungsional

Berdasarkan analisis kebutuhan sistem yang dijelaskan pada tabel 3.22, maka kebutuhan fungsional yang muncul dapat dibedakan menjadi 3 tingkatan pengguna (*role*):

1. Administrator

- a. *Login* berfungsi untuk autentikasi keamanan dan pengecekan *session* administrator ketika memasuki sistem pendukung keputusan.
- b. Data akun berfungsi untuk mengelola seluruh akun terdaftar dan mengelola profil akun pribadi.

- c. Data alternatif berfungsi untuk melihat data alternatif.
- d. Data kriteria berfungsi untuk melihat data kriteria serta parameternya.
- e. Data nilai awal untuk melihat nilai tiap alternatif terhadap kriteria dan nilai v.
- f. Data perhitungan untuk melihat perhitungan secara detail dan mencetak hasil perhitungannya.
- g. Data peta berfungsi untuk melihat peta lokasi alternatif.

2. Operator

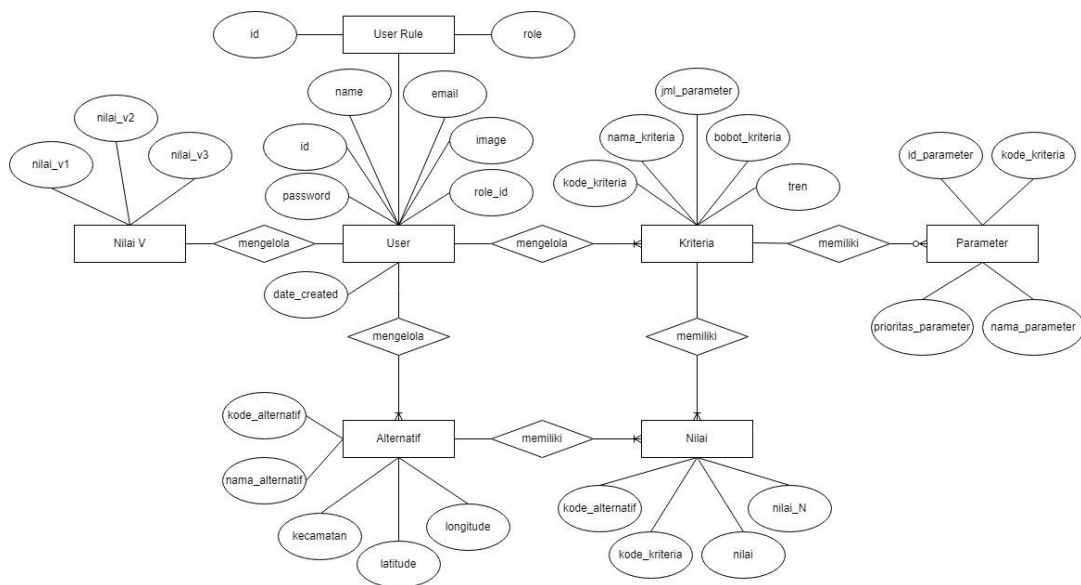
- a. *Login* berfungsi untuk autentikasi keamanan dan pengecekan *session* operator ketika memasuki sistem pendukung keputusan.
- b. Data akun berfungsi untuk mengelola profil akun pribadi..
- c. Data alternatif berfungsi untuk melihat, menambah, memperbarui dan menghapus data alternatif.
- d. Data kriteria berfungsi untuk melihat, menambah, memperbarui dan menghapus data kriteria serta parameternya.
- e. Data nilai awal untuk melihat, menambah, memperbarui dan menghapus nilai tiap alternatif terhadap kriteria dan nilai v.
- f. Data perhitungan untuk melihat perhitungan secara detail dan mencetak hasil perhitungannya.
- g. Data peta berfungsi untuk melihat peta lokasi alternatif.

3. *Guest*

- a. Data alternatif berfungsi untuk melihat data alternatif.
- b. Data kriteria berfungsi untuk melihat data kriteria.
- c. Data perangkian akhir berfungsi hanya untuk melihat hasil akhir dari perhitungan VIKOR.
- d. Data peta berfungsi untuk melihat peta lokasi alternatif.

2. Perancangan Basis Data secara Konseptual

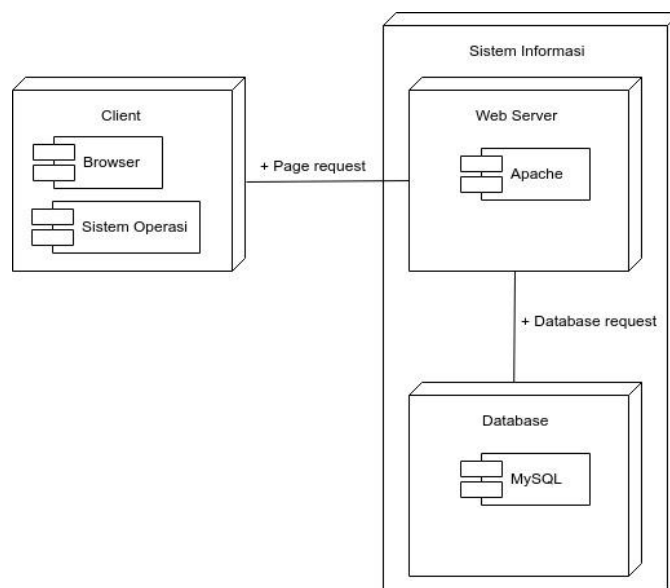
Melalui tahap ini akan dihasilkan skema konseptual basis data yang akan memperinci kebutuhan sistem. Skema konseptual basis data sering menggunakan ERD (*Entity Relation Diagram*) model, seperti yang ditunjukkan pada gambar 3.5.



Gambar 3.5 Entity relation diagram

3. Pemilihan DBMS (*Database Management System*)

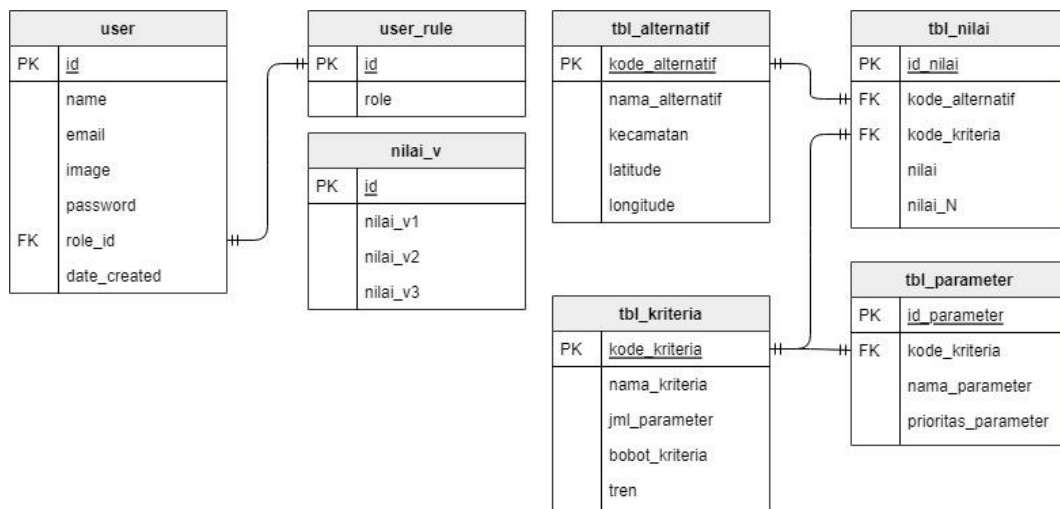
Database Management System digunakan untuk menyimpan, menampilkan, dan mengelola data. Pemilihan *DBMS* dapat ditentukan dengan memperhatikan faktor teknik, ekonomi, politik dan organisasi. Dalam hal ini untuk menjalankan tugasnya, *DBMS* dapat digambarkan melalui *Deployment Diagram* yang menjelaskan *relational, network, hierarchy*, struktur penyimpanan dan jalur akses seperti yang ditunjukkan pada gambar 3.6.



Gambar 3.6 Deployment diagram

4. Perancangan Basis Data secara Logika

Pada tahap ini menggambarkan skema secara konseptual ke dalam DBMS yang dipilih yang akan memetakan sistem perancangan basis data dengan model skema basis data yang ditunjukkan pada gambar 3.7.



Gambar 3.7 Skema basis data

5. Perancangan Basis Data secara Fisik

Perancangan secara fisik didefinisikan dalam hal pemilihan struktur penyimpanan dan tabel yang dapat digambarkan melalui tabel-tabel berikut:

1. Tabel users

Tabel user dideklarasikan kolom atau atribut, tipe data beserta panjang datanya yang ditunjukkan pada tabel 3.23.

Tabel 3.23 Struktur tabel user

No	Kolom	Tipe Data (<i>Length</i>)	Keterangan
1	id	int (11)	<i>primary key</i>
2	name	varchar (128)	-
3	email	varchar (128)	-
4	image	varchar (128)	-
5	password	varchar (256)	-
6	role_id	int(11)	-
8	date_created	int(11)	-

2. Tabel user_rule

Tabel user_rule dideklarasikan kolom atau atribut, tipe data beserta panjang datanya yang ditunjukkan pada tabel 3.24.

Tabel 3.24 Struktur tabel user_rule

No	Kolom	Tipe Data (<i>Length</i>)	Keterangan
1	id	int (11)	<i>primary key</i>
2	role	varchar (128)	-

3. Tabel tbl_alternatif

Tabel tbl_alternatif dideklarasikan kolom atau atribut, tipe data beserta panjang datanya yang ditunjukkan pada tabel 3.25.

Tabel 3.25 Struktur tabel tbl_alternatif

No	Kolom	Tipe Data (<i>Length</i>)	Keterangan
1	kode_alternatif	varchar(255)	<i>primary key</i>
2	nama_alternatif	varchar(255)	-
3	kecamatan	varchar(255)	-
4	latitude	double	-
5	longitude	double	-

4. Tabel tbl_kriteria

Tabel tbl_kriteria dideklarasikan kolom atau atribut, tipe data beserta panjang datanya yang ditunjukkan pada tabel 3.26.

Tabel 3.26 Struktur tabel tbl_kriteria

No	Kolom	Tipe Data (<i>Length</i>)	Keterangan
1	kode_kriteria	varchar(255)	<i>primary key</i>
2	nama_kriteria	varchar(255)	-
3	jml_parametert	int(11)	-
4	bobot_kriteria	double	-
5	tren	varchar(126)	-

5. Tabel tbl_parameter

Tabel tbl_parameter dideklarasikan kolom atau atribut, tipe data beserta panjang datanya yang ditunjukkan pada tabel 3.27.

Tabel 3.27 Struktur tabel tbl_parameter

No	Kolom	Tipe Data (<i>Length</i>)	Keterangan
1	id_parameter	int(155)	<i>primary key</i>
2	kode_kriteria	varchar(255)	<i>foreign key</i>
3	nama_parameter	varchar(255)	-
4	prioritas_parameter	int(50)	-

6. Tabel tbl_nilai

Tabel tbl_nilai dideklarasikan kolom atau atribut, tipe data beserta panjang datanya yang ditunjukkan pada tabel 3.28.

Tabel 3.28 Struktur tabel nilai

No	Kolom	Tipe Data (<i>Length</i>)	Keterangan
1	id_nilai	int(50)	<i>primary key</i>
2	kode_alternatif	varchar(255)	<i>foreign key</i>
3	kode_kriteria	varchar(255)	<i>foreign key</i>
4	nilai	double	-
5.	nilai_N	double	-

7. Tabel nilai_v

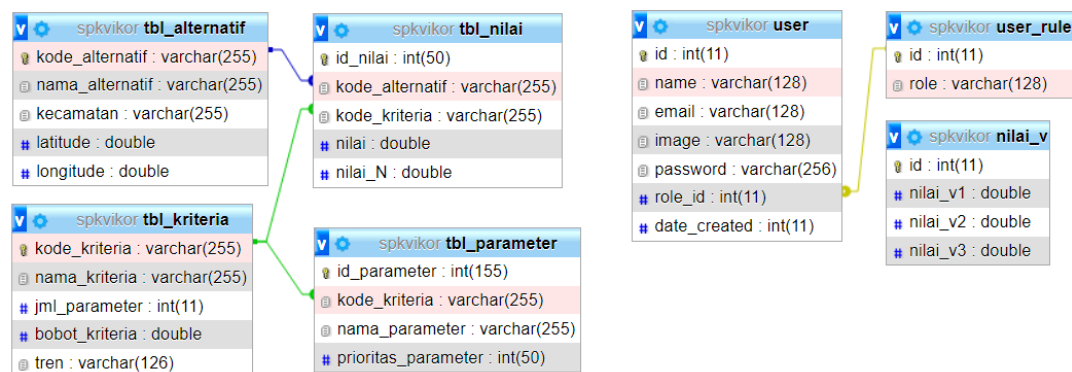
Tabel nilai_v dideklarasikan kolom atau atribut, tipe data beserta panjang datanya yang ditunjukkan pada tabel 3.29.

Tabel 3.29 Struktur tabel nilai_v

No	Kolom	Tipe Data (<i>Length</i>)	Keterangan
1	id	int(11)	<i>primary key</i>
2	nilai_v1	double	-
3	nilai_v2	double	-
4	nilai_v3	double	-

6. Implementasi Sistem Basis Data

Proses pembuatan sistem basis data dapat dilakukan dengan membuat *Class Diagram* dan *ECB (Entity Control Boundary)*. *Class Diagram* merupakan *diagram* yang menggambarkan struktur sistem dari segi pendeklarasian kelas-kelas, kolom, atribut, tipe data, panjang data dan *attribute key* yang akan dibuat untuk membangun sistem. Diagram implementasi basis data dari sistem ini dapat dilihat pada gambar 3.8.



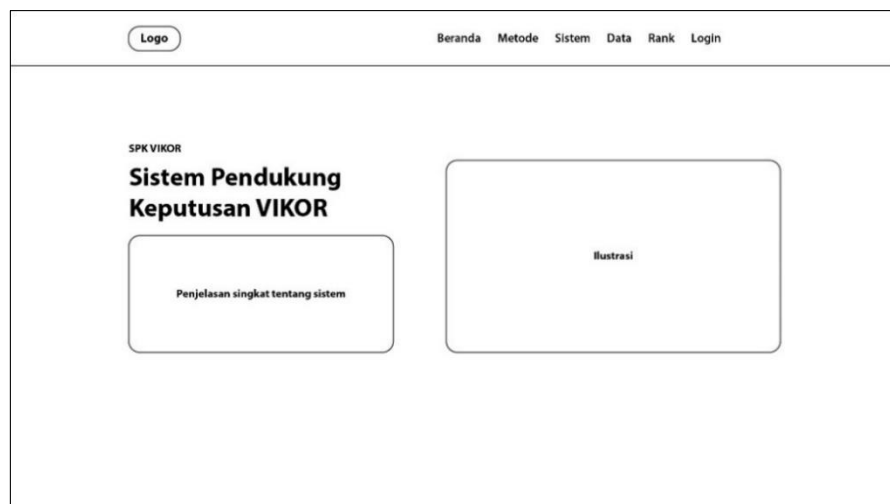
Gambar 3.8 Diagram class

3.2.3 Perancangan Antarmuka (*User Interface*)

Pembuatan antarmuka adalah pembuatan desain tampilan dari sistem yang terdiri dari desain tampilan setiap halaman. Pembuatan tampilan masukan berupa *form-form* sedangkan tampilan keluaran adalah tampilan untuk menampilkan hasil dari *input* oleh administrator, operator, dan *guest*. Berikut adalah gambaran dari tampilan antarmuka sistem informasi penentuan lokasi embung:

1. Tampilan Halaman Awal/*Landing Page*

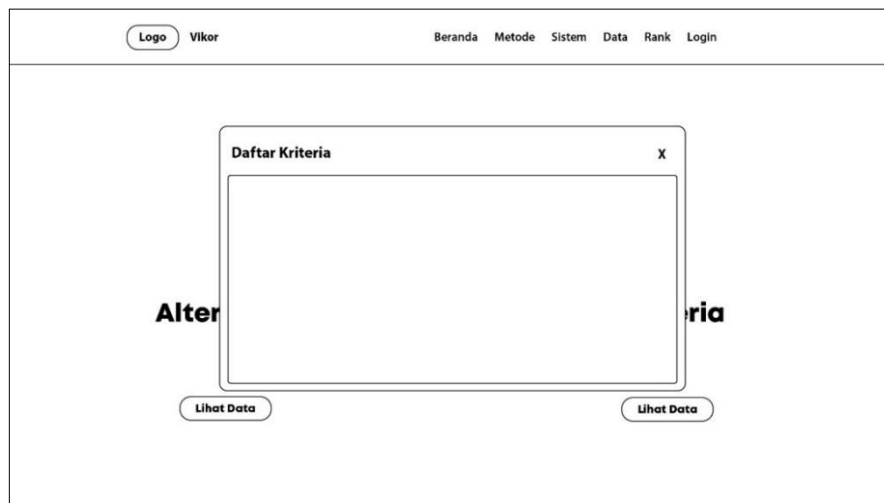
Halaman awal/*landing page* merupakan halaman pertama ketika pengguna mengakses sistem informasi. Pada halaman ini berisi penjelasan mengenai fitur yang ada di sistem informasi ini, penjelasan langkah-langkah perhitungan pada metode VIKOR, visualisasi peta, data alternatif dan kriteria yang diolah hingga hasil perangkingan akhir. Perancangan halaman ini ditunjukkan oleh gambar 3.9.



Gambar 3.9 Rancang tampilan halaman awal/*landing page*

2. Tampilan *Landing Page Section* Kriteria

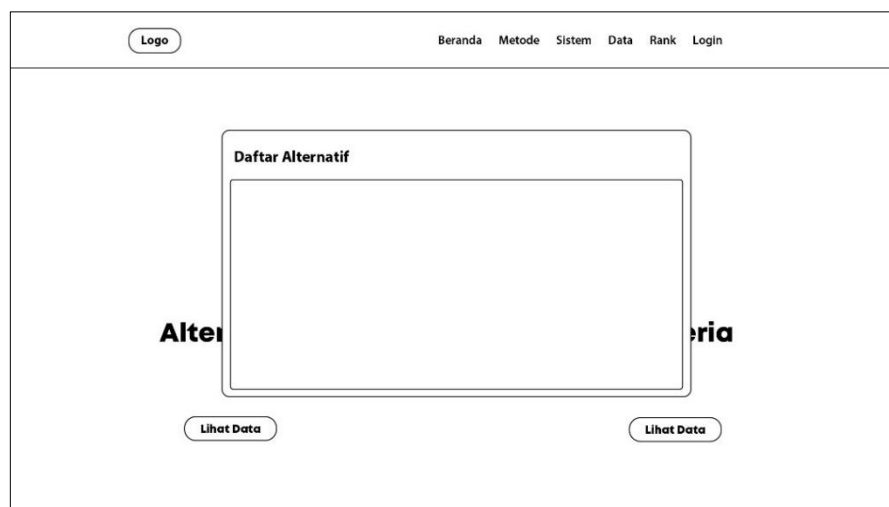
Halaman ini merupakan halaman untuk melihat daftar kriteria tanpa harus melakukan *login* dahulu yang bisa dilihat oleh siapa pun. Perancangan halaman ini ditunjukkan oleh gambar 3.10.



Gambar 3.10 Rancang tampilan halaman *landing page section* kriteria

3. Tampilan *Landing Page Section* Alternatif

Halaman ini merupakan halaman untuk melihat daftar alternatif tanpa harus melakukan *login* dahulu yang bisa dilihat oleh siapa pun. Perancangan halaman ini ditunjukkan oleh gambar 3.11.



Gambar 3.11 Rancang tampilan halaman *landing page section* alternatif

4. Tampilan *Landing Page Section* Hasil Perangkingan

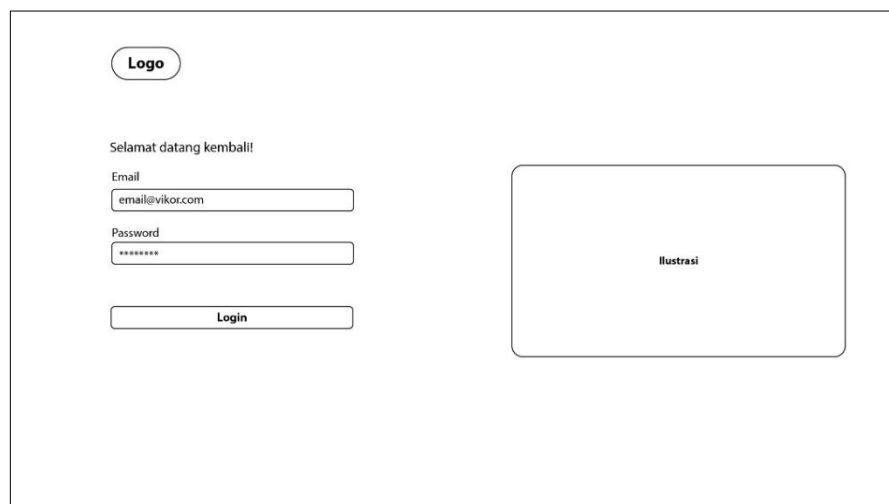
Halaman ini merupakan halaman untuk melihat hasil perangkingan yang diperoleh dari perhitungan tanpa harus melakukan *login* dahulu yang bisa dilihat oleh siapa pun. Perancangan halaman ini ditunjukkan oleh gambar 3.12.



Gambar 3.12 Rancang tampilan halaman *landing page section* perangkingan

5. Tampilan *Login*

Halaman *login* digunakan untuk pengguna melakukan proses *input email* dan *password* untuk masuk ke dalam sistem. Pada halaman ini dilakukan juga proses *checking multilevel* berdasar *role* yang sudah ditentukan. Perancangan halaman ini ditunjukkan oleh gambar 3.13.

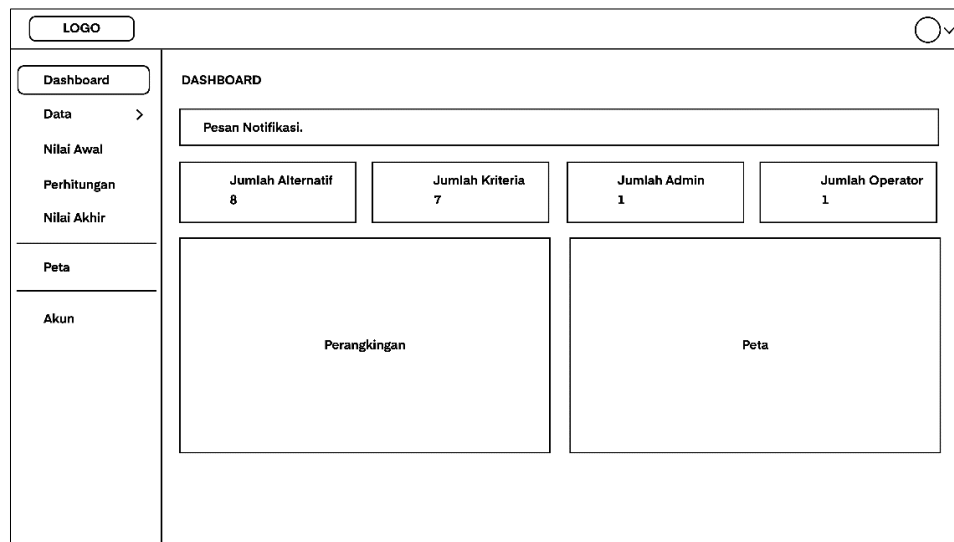


Gambar 3.13 Rancang tampilan halaman *login*

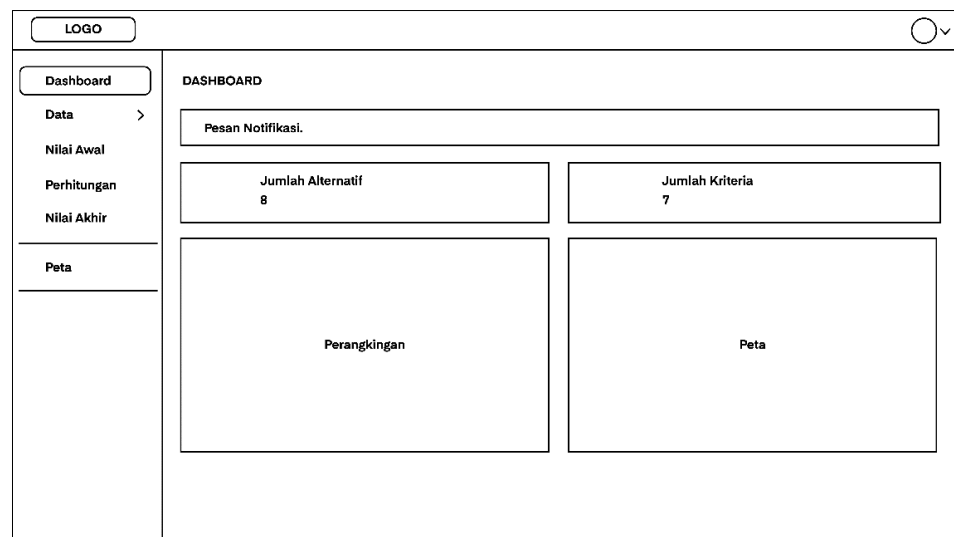
6. Tampilan *Dashboard*

Halaman *dashboard* adalah halaman yang pertama kali dilihat oleh pengguna setelah *login*. Menu akun hanya akan bisa dilihat dan diakses oleh administrator, untuk operator data jumlah yang bisa dilihat hanya jumlah data alternatif dan data kriteria. Pada halaman ini terdapat *sidebar* yang berisi menu-menu sistem, menu-

menu ini akan menyesuaikan sesuai dengan *role* akun yang melakukan *login*. Isi dari halaman *dashboard* adalah jumlah data yang dikelola, hasil perangkingan, serta visualisasi peta. Perancangan halaman ini ditunjukkan oleh gambar 3.14 dan gambar 3.15.



Gambar 3.14 Rancang tampilan halaman *dashboard* administrator

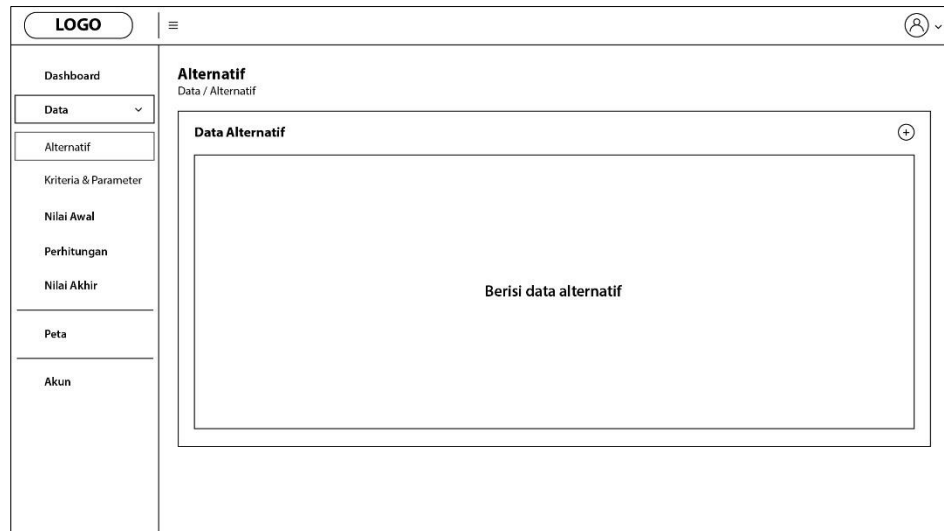


Gambar 3.15 Rancang tampilan halaman *dashboard* operator

7. Tampilan Data Alternatif

Halaman data alternatif berisi data alternatif yang berupa kode alternatif, nama alternatif, *latitude*, *longitude*, dan kecamatan yang bisa dilihat oleh administrator dan operator serta tombol *action* seperti tambah alternatif, edit

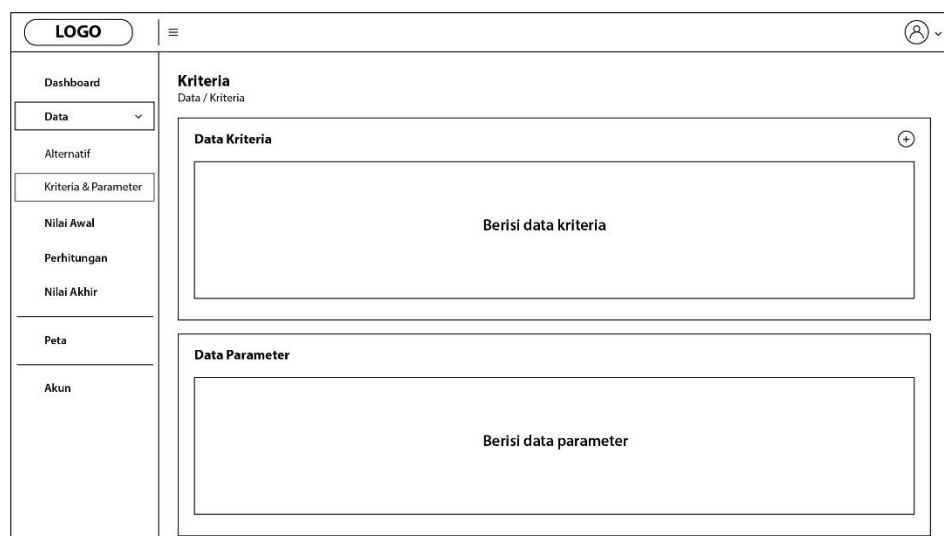
alternatif, dan hapus alternatif yang hanya bisa dilihat dan diakses oleh operator. Perancangan halaman ini ditunjukkan oleh gambar 3.16.



Gambar 3.16 Rancang tampilan halaman data alternatif

8. Tampilan Data Kriteria dan Parameter

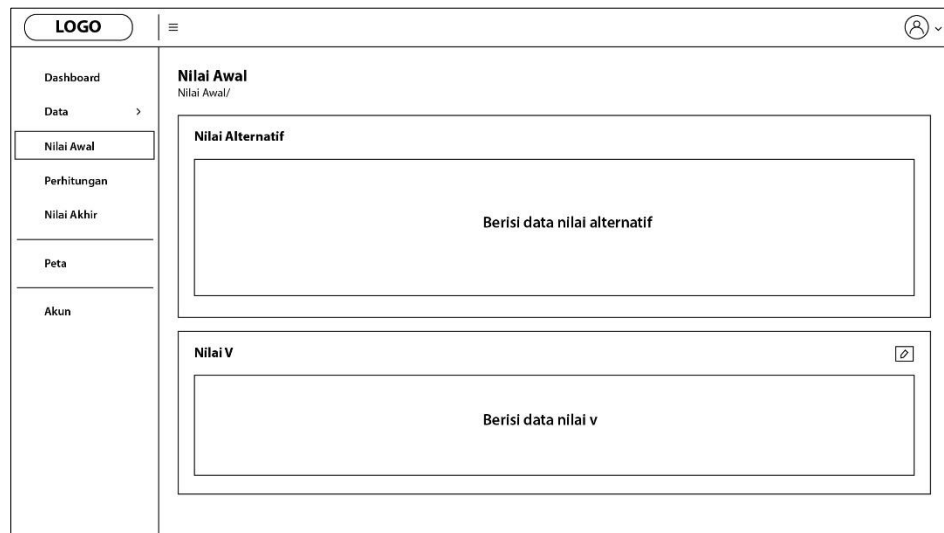
Halaman data kriteria dan parameter berisi data kriteria yang berupa kode kriteria, nama kriteria, bobot kriteria, tren kriteria, jumlah parameter, dan parameter yang bisa dilihat oleh administrator dan operator serta tombol *action* seperti tambah kriteria, edit kriteria, dan hapus kriteria yang hanya bisa dilihat dan diakses oleh operator. Perancangan halaman ini ditunjukkan oleh gambar 3.17.



Gambar 3.17 Rancang tampilan halaman data kriteria dan parameter

9. Tampilan Data Nilai Awal

Halaman data nilai awal berisi data nilai tiap alternatif terhadap kriteria. Halaman ini bisa dilihat oleh administrator dan operator, hanya saja untuk tombol *action edit* hanya dapat dilihat dan diakses operator. Perancangan halaman ini ditunjukkan oleh gambar 3.18.



Gambar 3.18 Rancang tampilan halaman data nilai awal

10. Tampilan Perhitungan

Halaman perhitungan berisi seluruh tahap perhitungan VIKOR mulai dari membuat matriks keputusan, menentukan bobot kriteria, membuat matriks normalisasi, membuat normalisasi bobot, mencari nilai *utility measure* dan *regret measure*, menghitung nilai indeks hingga didapatkan perangkingan alternatif lalu menentukan solusi kompromi serta pengujiannya. Setiap perhitungan ditampilkan dalam bentuk *card* yang dapat di *minimize*. Di halaman ini juga terdapat tombol cetak untuk mencetak hasil perhitungan. Halaman ini bisa diakses oleh administrator dan operator. Perancangan halaman ini ditunjukkan oleh gambar 3.19.

Gambar 3.19 Rancang tampilan halaman perhitungan

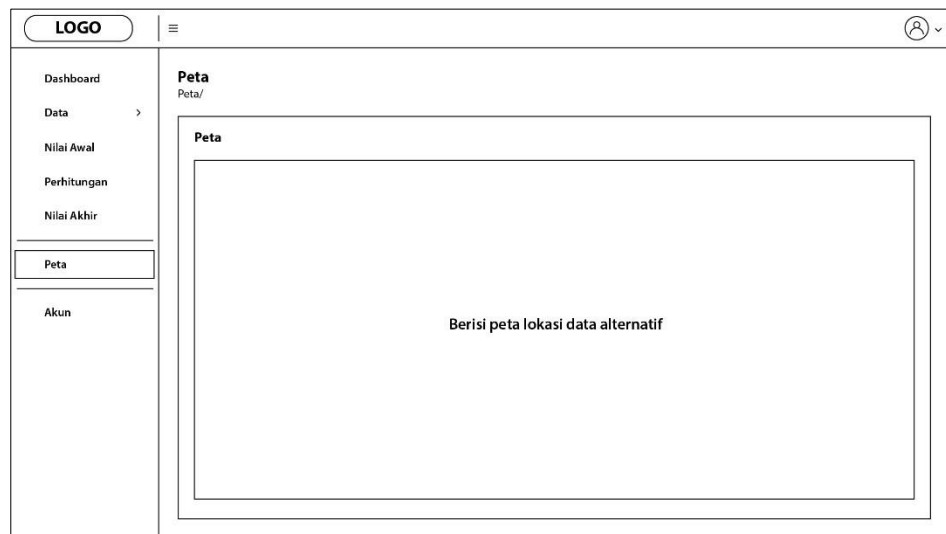
11. Tampilan Nilai Akhir

Halaman nilai akhir berisi hasil akhir dari tahap perhitungan VIKOR yakni hasil perangkingan disertai dengan 2 buah pengujian serta konklusinya. Halaman ini bisa diakses oleh administrator dan operator. Perancangan halaman ini ditunjukkan oleh gambar 3.20.

Gambar 3.20 Rancang tampilan halaman nilai akhir

12. Tampilan Peta

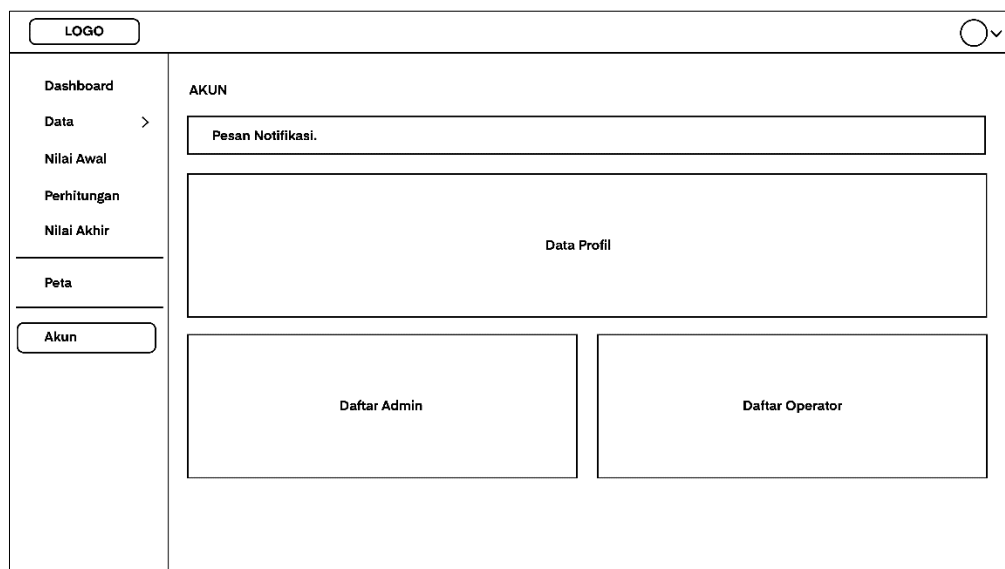
Halaman peta berisi tampilan peta lokasi alternatif sebagai visualisasi data dan halaman ini bisa diakses oleh administrator dan operator. Perancangan halaman ini ditunjukkan oleh gambar 3.21.



Gambar 3.21 Rancang tampilan halaman peta

13. Tampilan Akun

Halaman akun berisi data administrator dan operator yang terdaftar dan halaman ini hanya bisa diakses oleh administrator saja. Perancangan halaman ini ditunjukkan oleh gambar 3.22.



Gambar 3.22 Rancang tampilan halaman akun

BAB IV

HASIL PENELITIAN DAN PEMBAHASAN

4.1 Pembuatan Sistem


4.1.1 Pembuatan Basis Data

Basis data dibuat menggunakan Bahasa *SQL* (*Structure Query Language*) dengan memanfaatkan DBMS (*Database Management System*) pada *MySQL*. Server lokal yang digunakan untuk menjalankan basis data tersebut adalah *Apache* 2.4.46 dan *MySQL* 10.4.18 menggunakan *XAMPP* 7.3.27. Dengan berpedoman pada ERD (*Entity Relationship Diagram*) yang sebelumnya telah dirancang, maka dibuat basis data untuk setiap tabel atau entitas serta atribut dan relasinya. Berikut adalah tabel-tabel basis data sistem pendukung keputusan penentuan lokasi embung dengan menggunakan metode *Višekriterijumsko Kompromisno Rangiranje* (VIKOR):

1. Tabel user

Nama tabel: user

Berisi data user dengan *primary key* yaitu id. Struktur tabel user ditunjukkan pada gambar 4.1.

#	Name	Type	Collation	Attributes
1	id 	int(11)		
2	name	varchar(128)	utf8mb4_general_ci	
3	email	varchar(128)	utf8mb4_general_ci	
4	image	varchar(128)	utf8mb4_general_ci	
5	password	varchar(256)	utf8mb4_general_ci	
6	role_id	int(11)		
7	is_active	int(1)		
8	date_created	int(11)		

Gambar 4.1 Struktur tabel user

2. Tabel user_rule

Nama tabel: user_rule

Berisi data user_rule dengan *primary key* yaitu id. Struktur tabel user_rule ditunjukkan pada gambar 4.2.

#	Name	Type	Collation	Attributes
1	id	int(11)		
2	role	varchar(128)	utf8mb4_general_ci	

Gambar 4.2 Struktur tabel user_rule

3. Tabel alternatif

Nama tabel: tbl_alternatif

Berisi data alternatif dengan *primary key* yaitu kode_alternatif. Struktur tabel alternatif ditunjukkan pada gambar 4.3.

#	Name	Type	Collation	Attributes
1	kode_alternatif	varchar(255)	utf8mb4_general_ci	
2	nama_alternatif	varchar(255)	utf8mb4_general_ci	
3	kecamatan	varchar(255)	utf8mb4_general_ci	
4	latitude	double		
5	longitude	double		

Gambar 4.3 Struktur tabel alternatif

4. Tabel kriteria

Nama tabel: tbl_kriteria

Berisi data kriteria dengan *primary key* yaitu kode_kriteria. Struktur tabel kriteria ditunjukkan pada gambar 4.4.


#	Name	Type	Collation	Attributes
1	kode_kriteria	varchar(255)	utf8mb4_general_ci	
2	nama_kriteria	varchar(255)	utf8mb4_general_ci	
3	jml_parameter	int(11)		
4	bobot_kriteria	double		
5	tren	varchar(126)	utf8mb4_general_ci	

Gambar 4.4 Struktur tabel kriteria

5. Tabel nilai

Nama tabel: tbl_nilai

Berisi data nilai alternatif terhadap tiap kriteria dengan *primary key* yaitu id_nilai. Struktur tabel nilai ditunjukkan pada gambar 4.5.


#	Name	Type	Collation	Attributes
1	id_nilai 	int(50)		
2	kode_alternatif	varchar(255)	utf8mb4_general_ci	
3	kode_kriteria	varchar(255)	utf8mb4_general_ci	
4	nilai	double		
5	nilai_N	double		

Gambar 4.5 Struktur tabel nilai

6. Tabel parameter

Nama tabel: tbl_parameter

Berisi data parameter dengan *primary key* yaitu id_parameter. Struktur tabel parameter ditunjukkan pada gambar 4.6.

#	Name	Type	Collation	Attributes
1	id_parameter 	int(155)		
2	kode_kriteria	varchar(255)	utf8mb4_general_ci	
3	nama_parameter	varchar(255)	utf8mb4_general_ci	
4	prioritas_parameter	int(50)		

Gambar 4.6 Struktur tabel parameter

7. Tabel nilai v

Nama tabel: nilai_v

Berisi data nilai v dengan *primary key* yaitu id. Struktur tabel nilai v ditunjukkan pada gambar 4.7.

#	Name	Type	Collation	Attributes
1	id	int(11)		
2	nilai_v1	double		
3	nilai_v2	double		
4	nilai_v3	double		

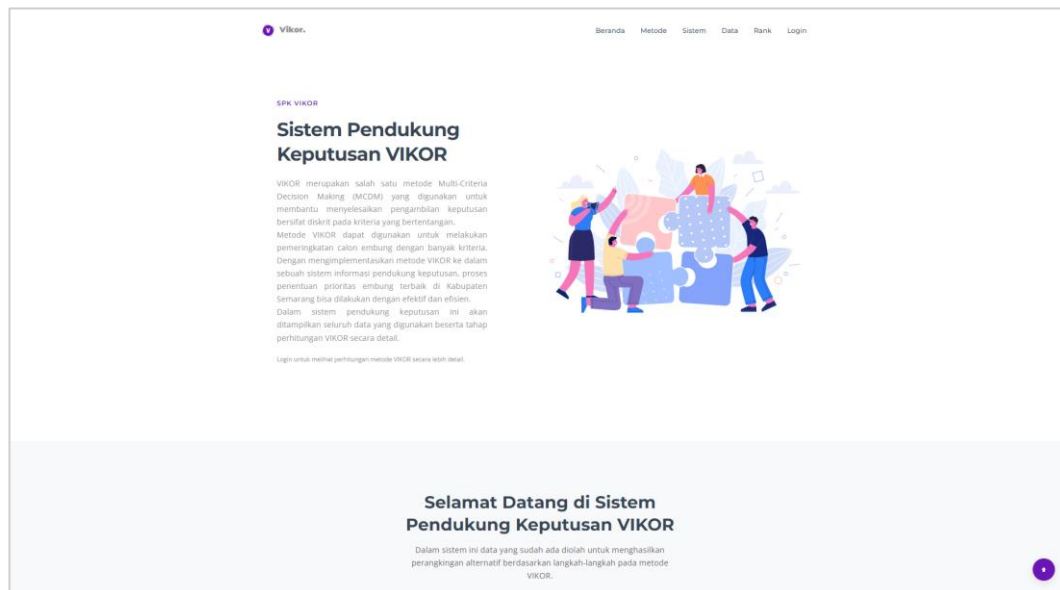
Gambar 4.7 Struktur tabel nilai v

4.1.2 Pembuatan Program

Pada perancangan program dijelaskan bagaimana struktur pembuatan sistem pendukung keputusan penentuan lokasi embung dengan menggunakan metode *Višekriterijumsko Kompromisno Rangiranje* (VIKOR). Program ini dibuat dengan menggunakan kerangka kerja *CodeIgniter* dengan konsep *model*, *view*, dan *controller* atau MVC. Berikut ini merupakan implementasi program yang terdapat dalam sistem informasi ini:

1. Tampilan Halaman Awal (*Landing Page*)

Halaman awal/*landing page* adalah halaman yang akan ditampilkan saat pertama kali mengakses sistem informasi ini. Halaman ini menampilkan gambaran mengenai sistem informasi pendukung ini dan metode VIKOR secara umum, serta menampilkan data alternatif, kriteria, dan hasil perhitungan berupa peringkat lokasi pembangunan embung yang juga divisualisasikan berupa peta lokasi embung di Kabupaten Semarang. Pada bagian atas halaman terdapat tombol *login* yang dapat digunakan oleh administrator dan operator melakukan *login* agar dapat masuk ke halaman *dashboard*/beranda. Halaman awal atau *landing page* ditunjukkan pada gambar 4.8.

Gambar 4.8 Halaman awal (*landing page*)

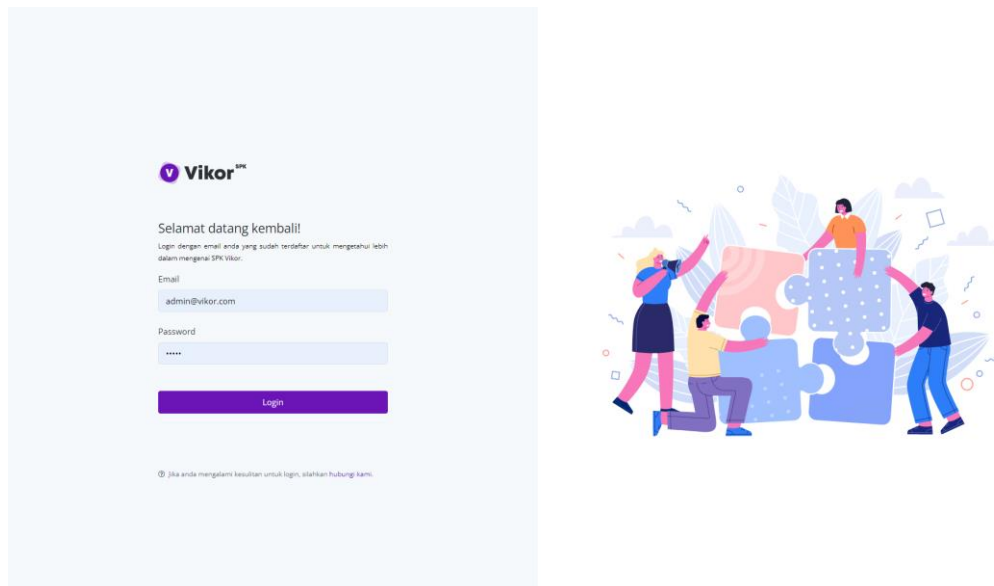
File hubungan antara *model*, *view*, dan *controller* pada halaman awal (*landing page*) ini ditunjukkan pada tabel 4.1.

Tabel 4.1 Tabel MVC *landing page*

<i>MVC</i>	Nama File	Fungsi
<i>Model</i>	Alternatif_model.php Kriteria_model.php Nilai_model.php	Memanggil data pada <code>tbl_alternatif</code> , <code>tbl_kriteria</code> , dan <code>tbl_nilai</code> untuk dikirimkan ke <i>controller</i> .
<i>View</i>	landingpage/index.php	Menampilkan halaman awal (<i>landing page</i>) beserta data-data yang disertakan.
<i>Controller</i>	Home.php	Menerima data dari model dan selanjutnya ditampilkan ke dalam <i>view</i> .

2. Tampilan Halaman *Login*

Halaman *login* merupakan halaman yang ditampilkan kepada pengguna ketika ingin masuk ke halaman *dashboard/beranda*. Pada halaman ini terdapat proses *input* email dan *password* yang dimiliki pengguna yang nantinya akan dilakukan proses autentikasi dan pengecekan *role* pengguna. Jika proses autentikasi dan pengecekan *role* berhasil maka akan diteruskan ke halaman *dashboard/beranda* operator atau administrator sesuai dengan *role* masing-masing akun. Halaman *login* ditunjukkan pada gambar 4.9.

Gambar 4.9 Halaman *login*

File hubungan antara *model*, *view*, dan *controller* halaman *login* ini ditunjukkan pada tabel 4.2.

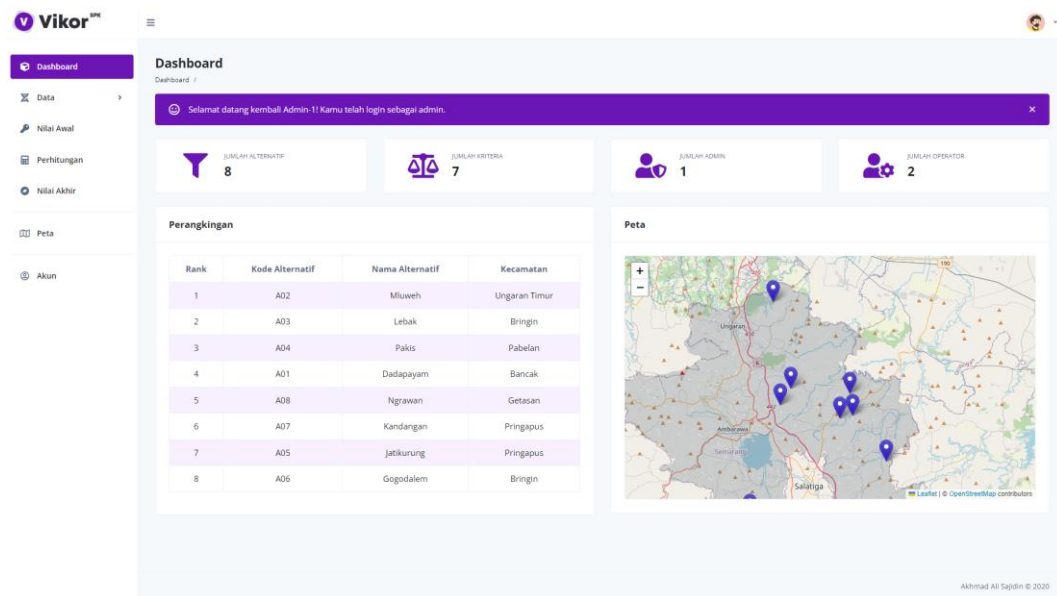
Tabel 4.2 Tabel MVC halaman *login*

<i>MVC</i>	Nama File	Fungsi
<i>Model</i>	Akun.php	Memanggil data pada tbl_user dari basis data.
<i>View</i>	login.php	Menampilkan halaman <i>login</i> .
<i>Controller</i>	Auth.php	Memanggil data dari model untuk digunakan sebagai proses autentikasi <i>username</i> dan <i>password</i> kepada user yang melakukan proses <i>login</i> .

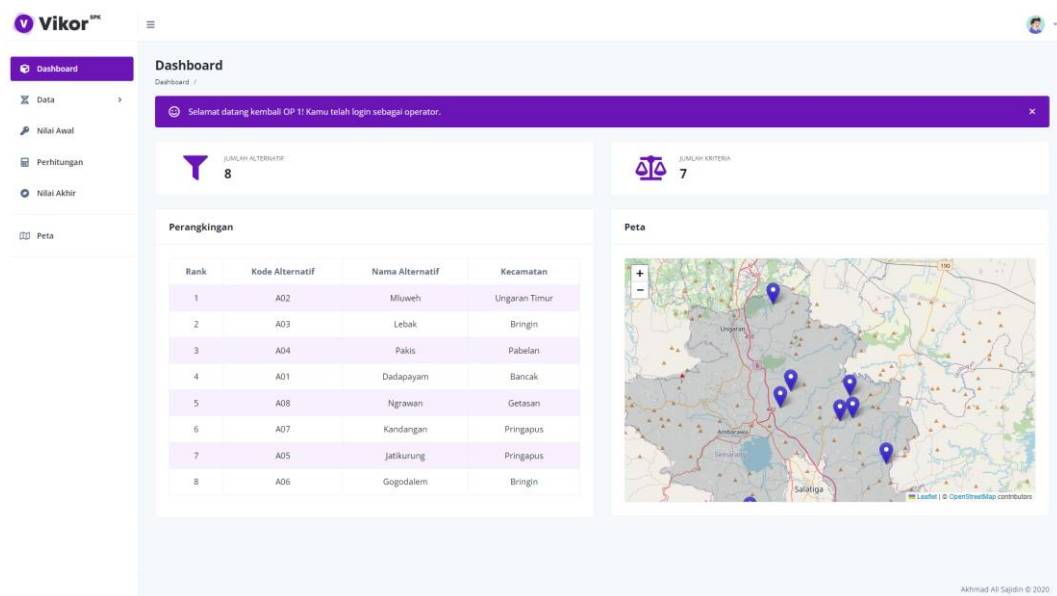
3. Tampilan Halaman Beranda/*Dashboard*

Halaman beranda/*dashboard* merupakan halaman yang akan ditampilkan ketika proses *login* berhasil. Halaman ini menampilkan *dashboard* administrator atau operator sesuai dengan *role* pengguna yang *login*. Perbedaan antar *dashboard* untuk administrator dan operator yaitu, untuk administrator terdapat menu akun pada *sidebar* yang digunakan untuk mengelola data pengguna yang terdaftar, dimana untuk *role* operator tidak ada menu tersebut. Pada bagian atas kiri terdapat logo sistem informasi dan bagian kanan atas terdapat *dropdown* yang berisi menu untuk membuka profil pengguna, menu untuk keluar dari sistem, atau menu untuk membuka halaman FAQ (*Frequently Asked Questions*). Bagian konten halaman beranda/*dashboard* berisi rangkuman mengenai jumlah pengguna terdaftar, jumlah

alternatif, jumlah kriteria, hasil perangkingan dengan metode VIKOR dan visualisasi Kabupaten Semarang. Halaman beranda/*dashboard* ditunjukkan pada gambar 4.10 dan 4.11.



Gambar 4.10 Halaman *dashboard* untuk pengguna dengan role administrator



Gambar 4.11 Halaman *dashboard* untuk pengguna dengan role operator

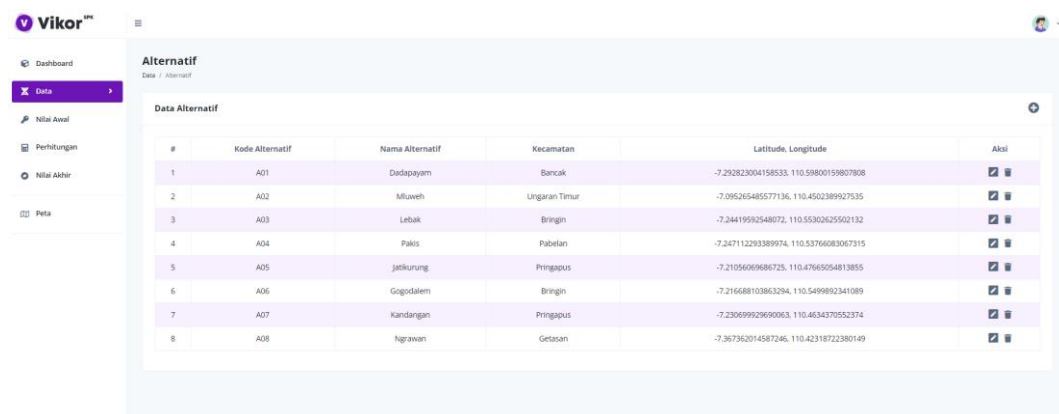
File hubungan antara *model*, *view*, dan *controller* halaman beranda/*dashboard* ini ditunjukkan pada tabel 4.3.

Tabel 4.3 Tabel MVC halaman beranda/*dashboard*

MVC	Nama File	Fungsi
Model	Alternatif_model.php Kriteria_model.php Akun_model.php Nilai_model.php	Memanggil data tbl_alternatif, tbl_kriteria, tbl_akun, dan tbl_nilai dari basis data untuk kemudian dikirimkan ke controller untuk diproses.
View	operator/index.php admin/index.php	Menampilkan halaman <i>dashboard</i> sesuai dengan <i>role user</i> yang melakukan <i>login</i> .
Controller	Operator.php Admin.php	Menerima data dari model dan selanjutnya ditampilkan ke dalam <i>view</i> .

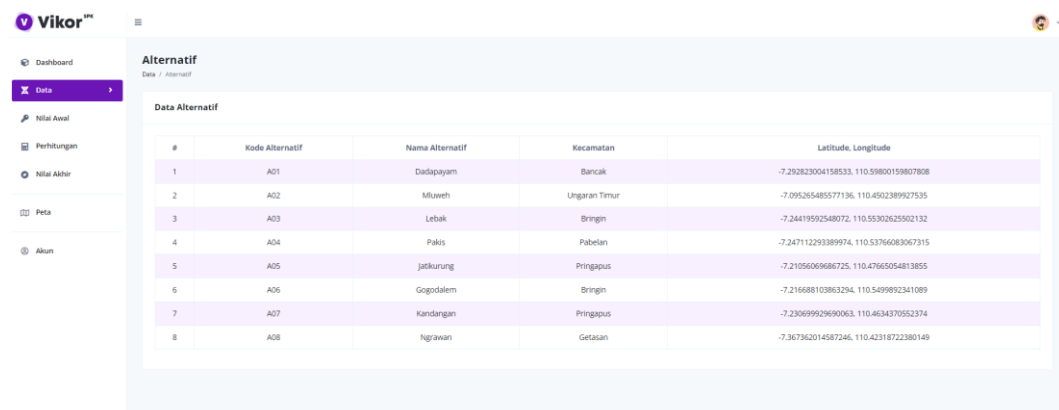
4. Tampilan Halaman Alternatif

Halaman alternatif merupakan halaman yang menampilkan informasi dari tabel alternatif. Administrator dapat melihat daftar lokasi alternatif, menambah alternatif beserta nilai alternatifnya, menghapus alternatif, dan memperbarui alternatif yang dipilih sedangkan administrator hanya mampu melihat data alternatif. Halaman alternatif ditunjukkan oleh gambar 4.12 sampai gambar 4.15.



#	Kode Alternatif	Nama Alternatif	Kecamatan	Latitude, Longitude	Aksi
1	A01	Dadapayam	Bancak	-7.292823004158533, 110.59800159807808	[Edit] [Delete]
2	A02	Miwueh	Ungaran Timur	-7.095265485577136, 110.4502389927535	[Edit] [Delete]
3	A03	Lebak	Bringin	-7.24419592548072, 110.55302625502132	[Edit] [Delete]
4	A04	Pakis	Pabelan	-7.247112293389974, 110.53766083067315	[Edit] [Delete]
5	A05	Jatikurung	Pringapus	-7.21056069686725, 110.47665054813855	[Edit] [Delete]
6	A06	Gogodalem	Bringin	-7.216688103863294, 110.5499892341089	[Edit] [Delete]
7	A07	Kandangan	Pringapus	-7.230699929690063, 110.4634370552374	[Edit] [Delete]
8	A08	ngrawan	Getasan	-7.367362014587246, 110.42318722380149	[Edit] [Delete]

Gambar 4.12 Halaman alternatif operator



#	Kode Alternatif	Nama Alternatif	Kecamatan	Latitude, Longitude	Aksi
1	A01	Dadapayam	Bancak	-7.292823004158533, 110.59800159807808	[Edit] [Delete]
2	A02	Miwueh	Ungaran Timur	-7.095265485577136, 110.4502389927535	[Edit] [Delete]
3	A03	Lebak	Bringin	-7.24419592548072, 110.55302625502132	[Edit] [Delete]
4	A04	Pakis	Pabelan	-7.247112293389974, 110.53766083067315	[Edit] [Delete]
5	A05	Jatikurung	Pringapus	-7.21056069686725, 110.47665054813855	[Edit] [Delete]
6	A06	Gogodalem	Bringin	-7.216688103863294, 110.5499892341089	[Edit] [Delete]
7	A07	Kandangan	Pringapus	-7.230699929690063, 110.4634370552374	[Edit] [Delete]
8	A08	ngrawan	Getasan	-7.367362014587246, 110.42318722380149	[Edit] [Delete]

Gambar 4.13 Halaman alternatif administrator

Gambar 4.14 Halaman tambah alternatif

Gambar 4.15 Halaman ubah alternatif

Halaman alternatif memuat informasi alternatif yang terdapat di basis data. Data alternatif lokasi yang ditampilkan berjumlah 8 lokasi yang meliputi Dadapayam, Lebak, Mluweh, Pakis, Jatikurung, Gogodalem, Kandangan, dan Ngrawan.

File hubungan antara *model*, *view*, dan *controller* halaman data alternatif ditunjukkan pada tabel 4.4.

Tabel 4.4 Tabel MVC Halaman alternatif

MVC	Nama File	Fungsi
Model	Alternatif_model.php	Memanggil data tbl_alternatif dari basis data untuk kemudian dikirimkan ke <i>controller</i> untuk diproses.
View	operator/alternatif.php admin/alternatif.php operator/addalternatif operator/editdataalternatif	Menampilkan halaman alternatif sesuai dengan <i>role user</i> yang melakukan <i>login</i> , halaman tambah alternatif, dan halaman edit alternatif.
Controller	Operator.php Admin.php	Menerima data dari model dan selanjutnya ditampilkan ke dalam <i>view</i> serta mengkoordinasi parameter saat terjadi penambahan, pengurangan, atau perubahan data alternatif.

5. Tampilan Halaman Kriteria dan Parameter

Halaman kriteria dan parameter merupakan halaman yang menampilkan informasi dari tabel kriteria dan tabel parameter. Halaman ini akan menampilkan kriteria-kriteria yang terdaftar beserta parameternya jika kriteria. Pada halaman ini operator dapat menambahkan kriteria berparameter maupun tidak berparameter, menghapus kriteria, mengubah kriteria termasuk mengubah bobot dari tiap kriteria sedangkan administrator hanya mampu melihat data kriteria dan parameter. Saat operator menambahkan kriteria, operator akan mendapatkan *pop-up modal* pilihan apakah kriteria yang akan ditambahkan memiliki parameter atau tidak, jika memiliki parameter maka operator harus mengisi jumlah parameternya pada *form* setelahnya. Halaman kriteria dan parameter ditunjukkan oleh gambar 4.16 sampai gambar 4.22..

Kriteria & Parameter
Data / Kriteria & Parameter

Data Kriteria

#	Kode Kriteria	Nama Kriteria	Bobot Kriteria	Tren	Jumlah Parameter	Aksi
1	C01	Vegetasi area genangan embung	0.127530395	Benefit	5	
2	C02	Volume material timbunan	0.095448311	Cost	5	
3	C03	Luas daerah yang akan dibebaskan	0.251508247	Cost	5	
4	C04	Volume tampungan efektif	0.133663994	Benefit	5	
5	C05	Lama operasi	0.159709096	Benefit	5	
6	C06	Harga air/m ³	0.12971751	Cost	5	
7	C07	Akses jalan menuju site bendungan	0.102422447	Benefit	4	

Data Parameter

#	Kode Kriteria	Nama Kriteria	Parameter 1	Parameter 2	Parameter 3	Parameter 4	Parameter 5
1	C01	Vegetasi area genangan embung	Hutan	Semak belukar	Ladang/tegalan	Sawah tadah hujan	Perkampungan
2	C02	Volume material timbunan	< 40.000 m ³	40.000 m ³ ≤ X < 80.000 m ³	80.000 m ³ ≤ X < 120.000 m ³	120.000 m ³ ≤ X < 160.000 m ³	160.000 m ³ ≤ X < 200.000 m ³
3	C03	Luas daerah yang akan dibebaskan	1,5 Ha ≤ X < 3 Ha	3 Ha ≤ X < 4,5 Ha	4,5 Ha ≤ X < 6 Ha	6 Ha ≤ X < 7,5 Ha	≥ 7,5 Ha
4	C04	Volume tampungan efektif	> 1 400 000 m ³	750 000 m ³ ≤ X < 1 400 000 m ³	500 000 m ³ ≤ X < 750 000	250 000 m ³ ≤ X < 500 000	< 250 000 m ³

Gambar 4.16 Halaman kriteria dan parameter operator

Vikor™

Dashboard

Data

Nilai Awal

Perhitungan

Nilai Akhir

Peta

Akun

Kriteria & Parameter

Data / Kriteria & Parameter

Data Kriteria

#	Kode Kriteria	Nama Kriteria	Bobot Kriteria	Tren	Jumlah Parameter
1	C01	Vegetasi area genangan embung	0.12753	Cost	5
2	C02	Volume material timbunan	0.09545	Cost	5
3	C03	Luas daerah yang akan dibebaskan	0.25151	Cost	5
4	C04	Volume tampungan efektif	0.13366	Benefit	5
5	C05	Lama Operasi	0.15971	Benefit	5
6	C06	Harga air/m3	0.12972	Cost	5
7	C07	Akses jalan menuju site bendungan	0.10242	Cost	4

Data Parameter

#	Kode Kriteria	Nama Kriteria	Parameter 1	Parameter 2	Parameter 3	Parameter 4	Parameter 5
1	C01	Vegetasi area genangan embung	Hutan	Semak Belukar	Ladang/Tagalan	Sawah Tadiah Hujan	Perkampungan
2	C02	Volume material timbunan	< 40.000 m3	40.000 m3 ≤ X < 80.000 m3	80.000 m3 ≤ X < 120.000 m3	120.000 m3 ≤ X < 160.000 m3	160.000 m3 ≤ X < 200.000 m3
3	C03	Luas daerah yang akan dibebaskan	1.5 Ha ≤ X < 3 Ha	3 Ha ≤ X < 4.5 Ha	4.5 Ha ≤ X < 6 Ha	6 Ha ≤ X < 7.5 Ha	≥ 7.5 Ha
4	C04	Volume tampungan efektif	< 250.000 m3	250.000 m3 ≤ X < 500.000 m3	500.000 m3 ≤ X < 750.000 m3	750.000 m3 ≤ X < 1.500.000 m3	≥ 1.500.000 m3
5	C05	Lama Operasi	< 40 hr	40 hr ≤ X < 60 hr	60 hr ≤ X < 80 hr	80 hr ≤ X < 100 hr	≥ 100 hr
6	C06	Harga air/m3	< Rp10.000,00	Rp10.000,00 ≤ X < Rp20.000,00	Rp20.000,00 ≤ X < Rp30.000,00	Rp30.000,00 ≤ X < Rp40.000,00	≥ Rp40.000,00
7	C07	Akses jalan menuju site bendungan	Tersedia jalan aspal sampai site	jalan makadam/tanah sampai site	jalan setapak	Tidak tersedia jalan	

Gambar 4.17 Halaman kriteria dan parameter administrator

Vikor™

Dashboard

Data

Nilai Awal

Perhitungan

Nilai Akhir

Peta

Kriteria & Parameter

Data / Kriteria & Parameter

Data Kriteria

#	Kode Kriteria	Nama Kriteria	Bobot Kriteria	Tren	Jumlah Parameter	Aksi
1	C01	Vegetasi area genangan embung	0.12753	Cost	5	
2	C02	Volume material timbunan	0.09545	Cost	5	
3	C03	Luas daerah yang akan dibebaskan	0.25151	Cost	5	
4	C04	Volume tampungan efektif	0.13366	Benefit	5	
5	C05	Lama Operasi	0.15971	Benefit	5	
6	C06	Harga air/m3	0.12972	Cost	5	
7	C07	Akses jalan menuju site bendungan	0.10242	Cost	4	

Apakah kriteria yang akan anda tambahkan memiliki parameter? ☒ Ya ☐ Tidak

Data Parameter

#	Kode Kriteria	Nama Kriteria	Parameter 1	Parameter 2	Parameter 3	Parameter 4	Parameter 5
1	C01	Vegetasi area genangan embung	Hutan	Semak Belukar	Ladang/Tagalan	Sawah Tadiah Hujan	Perkampungan
2	C02	Volume material timbunan	< 40.000 m3	40.000 m3 ≤ X < 80.000 m3	80.000 m3 ≤ X < 120.000 m3	120.000 m3 ≤ X < 160.000 m3	160.000 m3 ≤ X < 200.000 m3
3	C03	Luas daerah yang akan dibebaskan	1.5 Ha ≤ X < 3 Ha	3 Ha ≤ X < 4.5 Ha	4.5 Ha ≤ X < 6 Ha	6 Ha ≤ X < 7.5 Ha	≥ 7.5 Ha
4	C04	Volume tampungan efektif	< 250.000 m3	250.000 m3 ≤ X < 500.000 m3	500.000 m3 ≤ X < 750.000 m3	750.000 m3 ≤ X < 1.500.000 m3	≥ 1.500.000 m3
5	C05	Lama Operasi	< 40 hr	40 hr ≤ X < 60 hr	60 hr ≤ X < 80 hr	80 hr ≤ X < 100 hr	≥ 100 hr
6	C06	Harga air/m3	< Rp10.000,00	Rp10.000,00 ≤ X < Rp20.000,00	Rp20.000,00 ≤ X < Rp30.000,00	Rp30.000,00 ≤ X < Rp40.000,00	≥ Rp40.000,00
7	C07	Akses jalan menuju site bendungan	Tersedia jalan aspal sampai site	jalan makadam/tanah sampai site	jalan setapak	Tidak tersedia jalan	

Gambar 4.18 Pop-up konfirmasi jenis kriteria yang akan ditambahkan

Kriteria & Parameter

Data / Kriteria & Parameter

Data Kriteria

#	Kode Kriteria	Nama Kriteria	Bobot Kriteria	Tren	Jumlah Parameter	Aksi
1	C01	Vegetasi area genangan embung	0.12753	Cost	5	[Edit] [Hapus]
2	C02	Volume material timbunan	0.09545	Cost	5	[Edit] [Hapus]
3	C03	Luas daerah yang akan dibeaskan	0.25151	Cost	5	[Edit] [Hapus]
4	C04	Volume tanggungan efektif	0.13366	Benefit	5	[Edit] [Hapus]
5	C05	Lama Operasi	0.15971	Benefit	5	[Edit] [Hapus]
6	C06	Harga air/m3	0.12972	Cost	5	[Edit] [Hapus]
7	C07	Akses jalan menuju site bendungan	0.10242	Cost	4	[Edit] [Hapus]

Data Parameter

#	Kode Kriteria	Nama Kriteria	Parameter 1	Parameter 2	Parameter 3	Parameter 4	Parameter 5
1	C01	Vegetasi area genangan embung	Hutan	Semak Belukar	Ladang/Tegalan	Sawah Tadah Hujan	Perkampungan
2	C02	Volume material timbunan	< 40.000 m3	40.000 m3 ≤ X < 80.000 m3	80.000 m3 ≤ X < 120.000 m3	120.000 m3 ≤ X < 160.000 m3	160.000 m3 ≤ X < 200.000 m3
3	C03	Luas daerah yang akan dibeaskan	1,5 Ha ≤ X < 3 Ha	3 Ha ≤ X < 4,5 Ha	4,5 Ha ≤ X < 6 Ha	6 Ha ≤ X < 7,5 Ha	≥ 7,5 Ha
4	C04	Volume tanggungan efektif	< 250.000 m3	250.000 m3 ≤ X < 500.000 m3	500.000 m3 ≤ X < 750.000 m3	750.000 m3 ≤ X < 1.500.000 m3	≥ 1.500.000 m3
5	C05	Lama Operasi	< 40 hr	40 hr ≤ X < 60 hr	60 hr ≤ X < 80 hr	80 hr ≤ X < 100 hr	≥ 100 hr
6	C06	Harga air/m3	< Rp10.000,00	Rp10.000,00 ≤ X < Rp20.000,00	Rp20.000,00 ≤ X < Rp30.000,00	Rp30.000,00 ≤ X < Rp40.000,00	≥ Rp40.000,00
7	C07	Akses jalan menuju site bendungan	Tersedia jalan aspal sampai site	jalan makadam/tanah sampai site	jalan setapak	Tidak tersedia jalan	

Gambar 4.19 Form jumlah parameter saat menambahkan kriteria berparameter

Tambah Kriteria

Data / Kriteria & Parameter / Tambah Kriteria

Tambah Kriteria

Kode Kriteria: C08

Nama Kriteria: Nama kriteria

Bobot Kriteria: Bobot kriteria

Tren Kriteria: Cost

Parameter 1: Nama Parameter 1

Parameter 2: Nama Parameter 2

Parameter 3: Nama Parameter 3

Parameter 4: Nama Parameter 4

Parameter 5: Nama Parameter 5

[← Kembali](#) [Simpan](#)

Gambar 4.20 Halaman tambah kriteria berparameter

Tambah Kriteria

Data / Kriteria & Parameter / Tambah Kriteria

Tambah Kriteria

Kode Kriteria: C08

Nama Kriteria: Masukkan nama kriteria

Bobot Kriteria: Masukkan bobot kriteria

Tren Kriteria: Cost

[← Kembali](#) [Simpan](#)

Gambar 4.21 Halaman tambah kriteria tanpa parameter

Gambar 4.22 Halaman ubah kriteria

Pada halaman kriteria memuat informasi kriteria beserta dengan parameternya masing-masing yang terdapat di basis data. Kriteria yang ditampilkan berjumlah tujuh kriteria yaitu, volume material timbunan, luas daerah yang akan dibebaskan, volume tampungan efektif, lama operasi, harga air/m³, vegetasi area genangan embung dan akses jalan menuju *site* bendungan.

File hubungan antara *model*, *view*, dan *controller* halaman kriteria dan parameter ditunjukkan pada tabel 4.5.

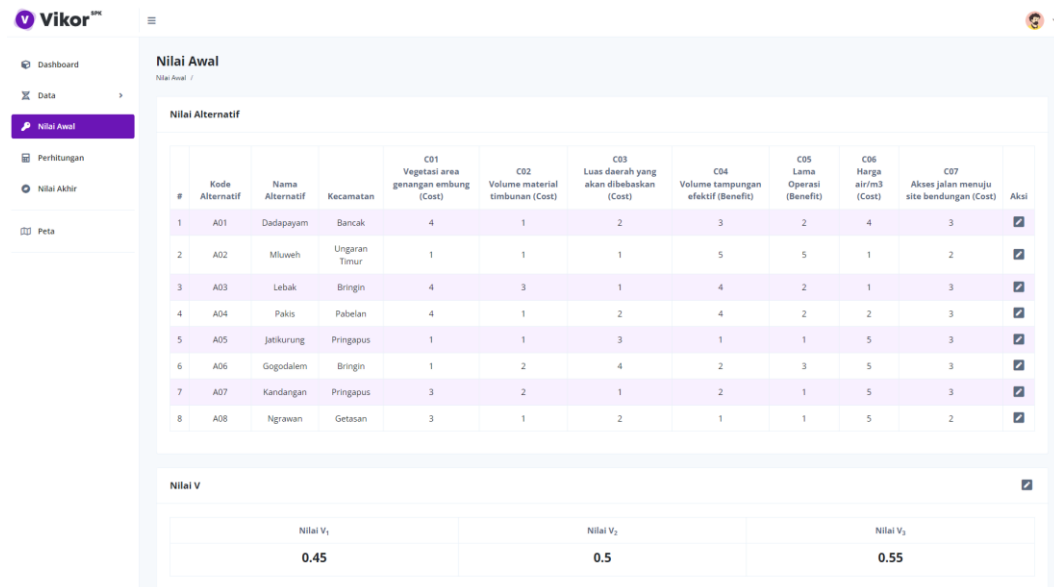
Tabel 4.5 Tabel MVC halaman kriteria

MVC	Nama File	Fungsi
<i>Model</i>	Kriteria_model.php	Memanggil data <code>tbl_kriteria</code> dan <code>tbl_parameter</code> dari basis data untuk kemudian dikirimkan ke <i>controller</i> untuk diproses.
<i>View</i>	operator/kriteria.php admin/kriteria.php operator/addkriteria.php operator/addkriteriaparam.php operator/editkriteria.php	Menampilkan halaman kriteria sesuai dengan <i>role user</i> yang melakukan <i>login</i> , halaman tambah kriteria berparameter dan tidak berparameter, serta halaman edit kriteria.
<i>Controller</i>	Operator.php Admin.php	Menerima data dari model yang selanjutnya ditampilkan ke dalam <i>view</i> serta mengkoordinasi parameter saat terjadi penambahan, pengurangan, atau perubahan pada data kriteria dan parameter.

6. Tampilan Halaman Nilai Awal

Halaman nilai awal merupakan halaman yang menampilkan informasi dari tabel nilai, yaitu nilai awal dari masing-masing alternatif terhadap kriteria dan nilai V. Pada halaman ini, administrator hanya dapat melihat data nilai awal sedangkan

operator dapat mengubah nilai awal alternatif melalui tombol edit pada kolom aksi dan mengubah nilai V melalui tombol edit pada pojok *card* nilai V. Halaman nilai awal ditunjukkan pada gambar 4.23 dan 4.24.



Nilai Awal

Nilai Awal /

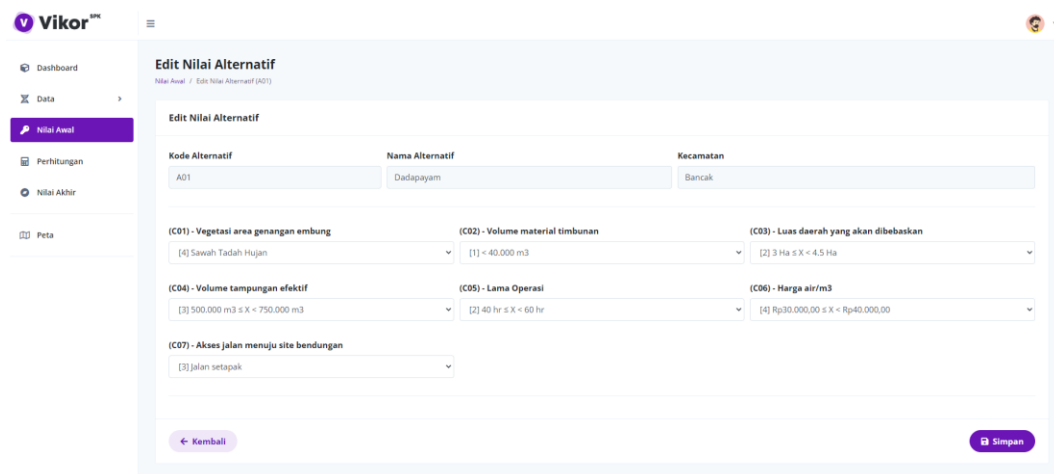
Nilai Alternatif

#	Kode Alternatif	Nama Alternatif	Kecamatan	C01 Vegetasi area genangan embung (Cost)	C02 Volume material timbunan (Cost)	C03 Luas daerah yang akan dibebaskan (Cost)	C04 Volume tampungan efektif (Benefit)	C05 Lama Operasi (Benefit)	C06 Harga air/m3 (Cost)	C07 Akses jalan menuju site bendungan (Cost)	Aksi
1	A01	Dadapayam	Bancak	4	1	2	3	2	4	3	<input checked="" type="checkbox"/>
2	A02	Mluweh	Ungaran Timur	1	1	1	5	5	1	2	<input checked="" type="checkbox"/>
3	A03	Lebak	Bringin	4	3	1	4	2	1	3	<input checked="" type="checkbox"/>
4	A04	Pakis	Pabelan	4	1	2	4	2	2	3	<input checked="" type="checkbox"/>
5	A05	Jatikurung	Pringapus	1	1	3	1	1	5	3	<input checked="" type="checkbox"/>
6	A06	Gogodalem	Bringin	1	2	4	2	3	5	3	<input checked="" type="checkbox"/>
7	A07	Kandangan	Pringapus	3	2	1	2	1	5	3	<input checked="" type="checkbox"/>
8	A08	Nigrawan	Getasan	3	1	2	1	1	5	2	<input checked="" type="checkbox"/>

Nilai V

Nilai V_1	Nilai V_2	Nilai V_3
0.45	0.5	0.55

Gambar 4.23 Halaman nilai awal



Edit Nilai Alternatif

Nilai Awal / Edit Nilai Alternatif (A01)

Edit Nilai Alternatif

Kode Alternatif: A01 Nama Alternatif: Dadapayam Kecamatan: Bancak

(C01) - Vegetasi area genangan embung: [4] Sawah Tadah Hujan (C02) - Volume material timbunan: [1] < 40,000 m3 (C03) - Luas daerah yang akan dibebaskan: [2] 3 Ha ≤ X < 4.5 Ha

(C04) - Volume tampungan efektif: [3] 500,000 m3 ≤ X < 750,000 m3 (C05) - Lama Operasi: [2] 40 hr ≤ X < 60 hr (C06) - Harga air/m3: [4] Rp30,000,00 ≤ X < Rp40,000,00

(C07) - Akses jalan menuju site bendungan: [3] Jalan setapak

[← Kembali](#) [Simpan](#)

Gambar 4.24 Halaman ubah nilai awal

Halaman nilai awal memuat informasi data nilai tiap alternatif terhadap kriteria yang ada pada basis data dan nilai V. Nilai ini yang akan dilakukan perhitungan dengan menggunakan metode VIKOR.

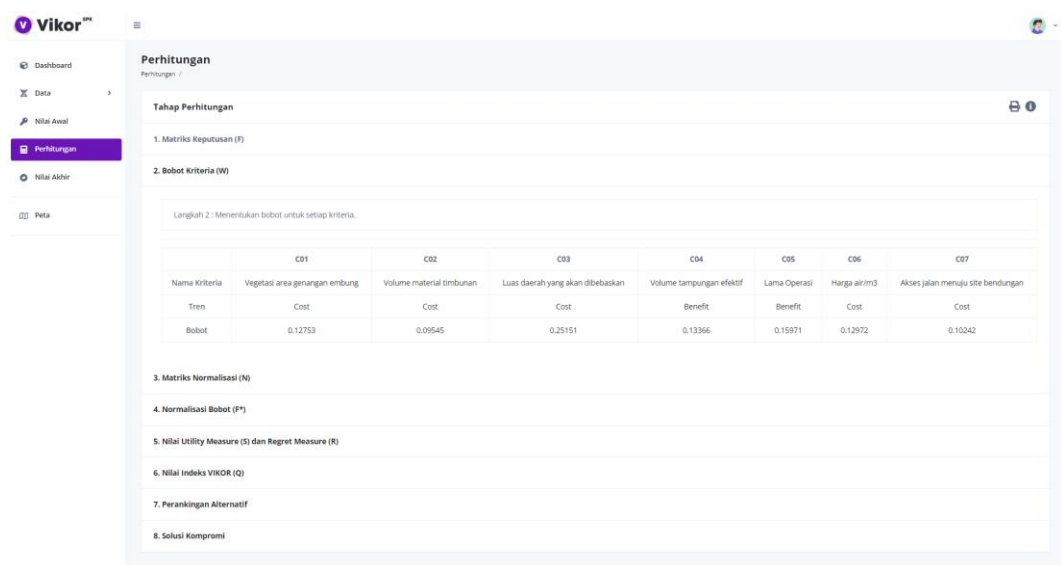
File hubungan antara *model*, *view*, dan *controller* halaman nilai awal ditunjukkan pada tabel 4.6.

Tabel 4.6 Tabel MVC halaman nilai awal

MVC	Nama File	Fungsi
Model	Nilai_model.php Alternatif.php Kriteria.php	Memanggil data tbl_alternatif, tbl_kriteria, dan tbl_nilai dari basis data untuk kemudian dikirimkan ke <i>controller</i> untuk diproses.
View	admin/nilai.php operator/nilai.php operator/editalternatif.php	Menampilkan halaman nilai awal sesuai dengan <i>role user</i> yang melakukan <i>login</i> , serta halaman edit nilai awal alternatif.
Controller	Operator.php Admin.php	Menerima data dari model dan selanjutnya ditampilkan ke dalam <i>view</i> serta mengkoordinasi parameter saat terjadi perubahan nilai awal.

7. Tampilan Halaman Perhitungan

Halaman perhitungan merupakan halaman yang digunakan untuk menunjukkan tahap-tahap perhitungan dengan menggunakan metode VIKOR terhadap data yang sudah dimasukkan. Halaman ini menampilkan data dari tabel alternatif, tabel kriteria, tabel nilai dan tabel nilai V. Halaman perhitungan ditunjukkan pada gambar 4.25.



Gambar 4.25 Halaman perhitungan

Dalam halaman ini ditunjukkan setiap tahap perhitungan dengan metode VIKOR beserta penjelasannya. Tahap-tahap perhitungan yang ditunjukkan yaitu: membuat matriks keputusan (F), bobot kriteria (W), matriks normalisasi (N), normalisasi bobot (F^*), nilai *utility measure* (S) dan *regret measure* (R), nilai indeks VIKOR (Q), perangkingan alternatif serta solusi kompromi.

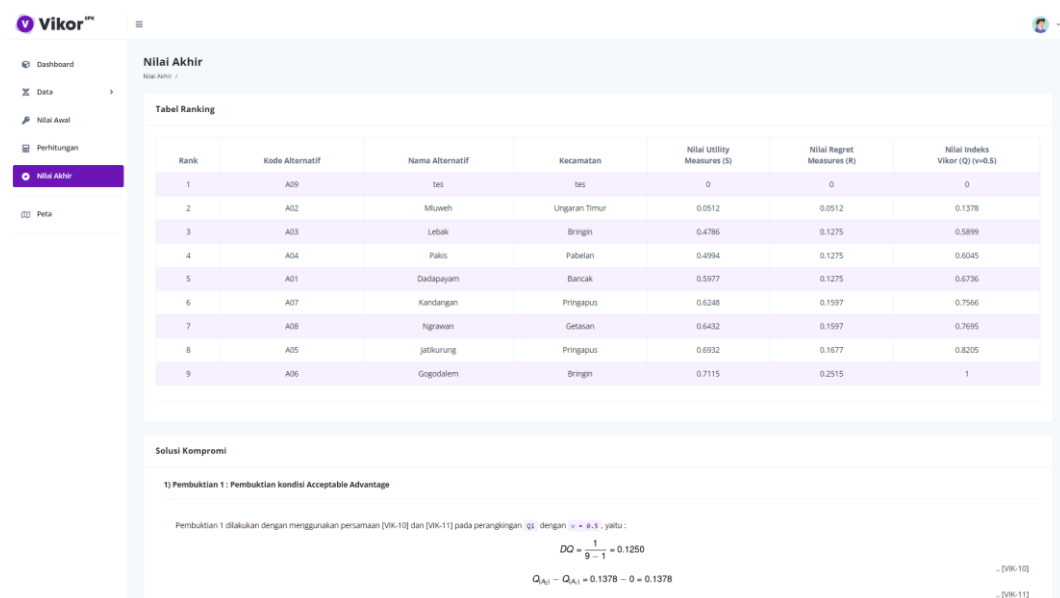
File hubungan antara *model*, *view*, dan *controller* halaman nilai awal ditunjukkan pada tabel 4.7.

Tabel 4.7 Tabel MVC halaman perhitungan

MVC	Nama File	Fungsi
<i>Model</i>	Alternatif_model.php Kriteria_model.php Nilai_model.php	Memanggil data tbl_alternatif, tbl_kriteria, tabel nilai_v, dan tbl_nilai dari basis data untuk kemudian dikirimkan ke <i>controller</i> untuk diproses.
<i>View</i>	perhitungan.php	Menampilkan halaman perhitungan yang berisi setiap tahap perhitungan metode VIKOR.
<i>Controller</i>	Perhitungan.php	Menerima data alternatif, kriteria, dan nilai dari model dan selanjutnya ditampilkan ke dalam <i>view</i> .

8. Tampilan Halaman Nilai Akhir

Halaman nilai akhir merupakan halaman yang digunakan untuk menunjukkan hasil akhir dari perhitungan VIKOR. Halaman ini menampilkan data dari tabel alternatif, tabel kriteria, tabel nilai dan tabel nilai V. Halaman nilai akhir ditunjukkan pada gambar 4.26.



Nilai Akhir

Tabel Ranking

Rank	Kode Alternatif	Nama Alternatif	Kecamatan	Nilai Utility Measures (S)	Nilai Regret Measures (R)	Nilai Indeks Vikor (Q) (n=0.5)
1	A09	tes	tes	0	0	0
2	A02	Muwah	Ungaran Timur	0.0512	0.0512	0.1378
3	A03	Lebak	Bringin	0.4786	0.1275	0.5899
4	A04	Pakis	Pabelan	0.4994	0.1275	0.6045
5	A01	Dadapayam	Banicak	0.5977	0.1275	0.6736
6	A07	Kandangan	Pringapus	0.6248	0.1597	0.7566
7	A08	ngrawan	Getasan	0.6432	0.1597	0.7695
8	A05	Jatikurung	Pringapus	0.6932	0.1677	0.8205
9	A06	Gogodalem	Bringin	0.7115	0.2515	1

Solusi Kompromi

1) Pembuktian 1: Pembuktian kondisi Acceptable Advantage

Pembuktian 1 dilakukan dengan menggunakan persamaan [VK-10] dan [VK-11] pada perbandingan Q_i dengan $Q_j = 9.5$, yaitu:

$$Q_i = \frac{1}{9-1} = 0.1250$$

$$Q_{A_0} - Q_{A_1} = 0.1378 - 0 = 0.1378$$

... [VK-10]

... [VK-11]

Gambar 4.26 Halaman nilai akhir

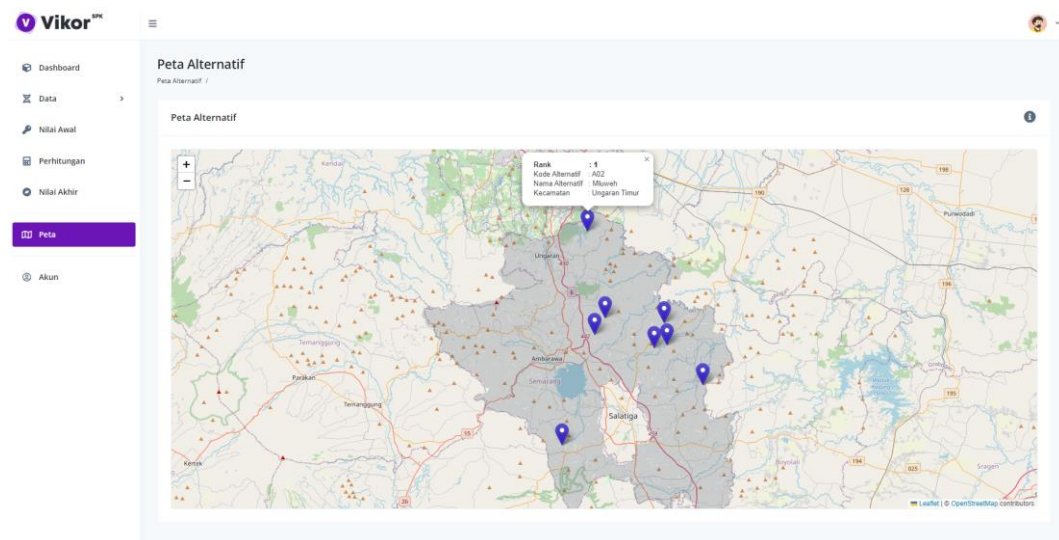
File hubungan antara *model*, *view*, dan *controller* halaman nilai akhir ditunjukkan pada tabel 4.8.

Tabel 4.8 Tabel MVC halaman nilai akhir

MVC	Nama File	Fungsi
Model	Alternatif_model.php Kriteria_model.php Nilai_model.php	Memanggil data tbl_alternatif, tbl_kriteria, tabel nilai_v, dan tbl_nilai dari basis data untuk kemudian dikirimkan ke <i>controller</i> untuk diproses.
View	nilai_akhir.php	Menampilkan halaman nilai akhir yang berisi hasil dari perhitungan metode VIKOR.
Controller	Nilaiakhir.php	Menerima data dari model dan selanjutnya ditampilkan ke dalam view.

9. Tampilan Halaman Peta

Halaman peta merupakan halaman yang menampilkan informasi dari tabel alternatif. Visualisasi peta menggunakan *javascript library* dari Leaflet dan untuk menandai lokasi dari tiap alternatif digunakan sistem koordinat geografis berupa garis lintang (*latitude*) dan garis bujur (*longitude*) yang tersimpan dalam tabel alternatif. Halaman peta ditunjukkan pada gambar 4.27.



Gambar 4.27 Halaman peta Kabupaten Semarang

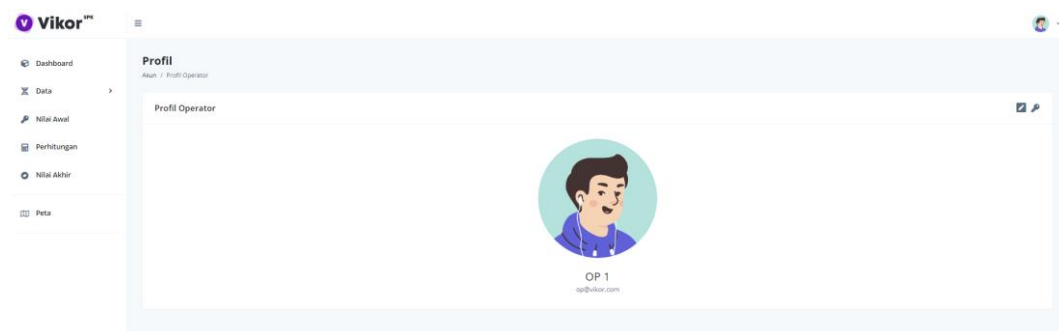
File hubungan antara *model*, *view*, dan *controller* halaman peta ditunjukkan pada tabel 4.9.

Tabel 4.9 Tabel MVC halaman peta

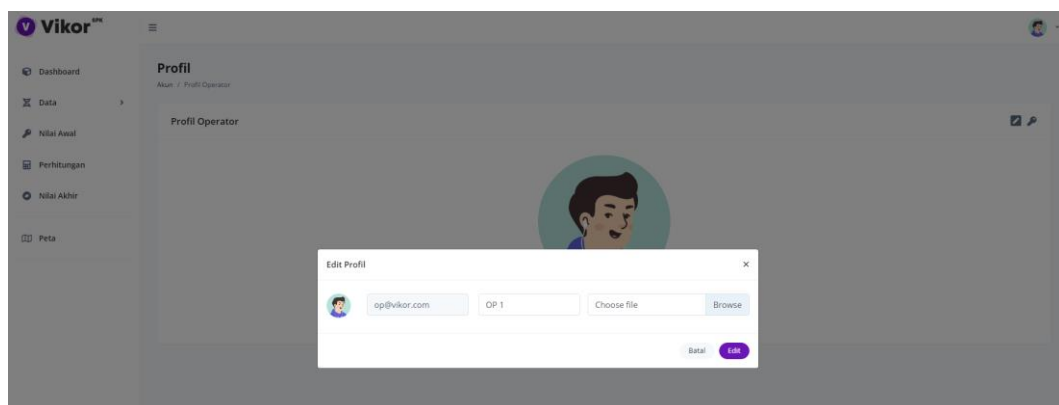
MVC	Nama File	Fungsi
Model	Alternatif_model.php	Memanggil data tbl_alternatif dari basis data untuk kemudian dikirimkan ke <i>controller</i> untuk diproses.
View	map.php	Menampilkan halaman peta yang berisi lokasi alternatif terdaftar.
Controller	Map.php	Menerima data dari model dan selanjutnya ditampilkan ke dalam view.

10. Tampilan Halaman Profil

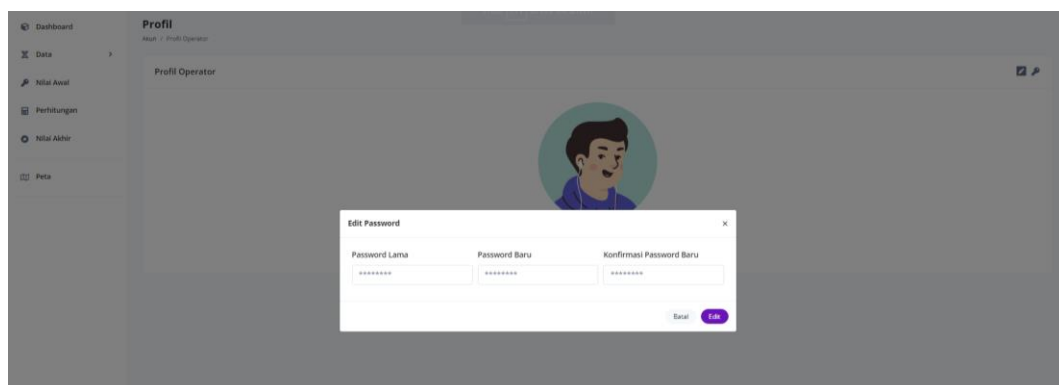
Halaman profil merupakan halaman yang menampilkan informasi akun operator yang sedang digunakan untuk masuk ke sistem. Operator dapat melihat dan memperbarui data profilnya sendiri yaitu nama, foto profil, dan *password* yang mana form untuk memperbarui data profil akan ditampilkan dalam bentuk *pop-up modal*. Halaman profil ditunjukkan pada gambar 4.28 sampai gambar 4.30.



Gambar 4.28 Halaman profil operator



Gambar 4.29 Form ubah nama dan foto profil operator



Gambar 4.30 Form ubah password

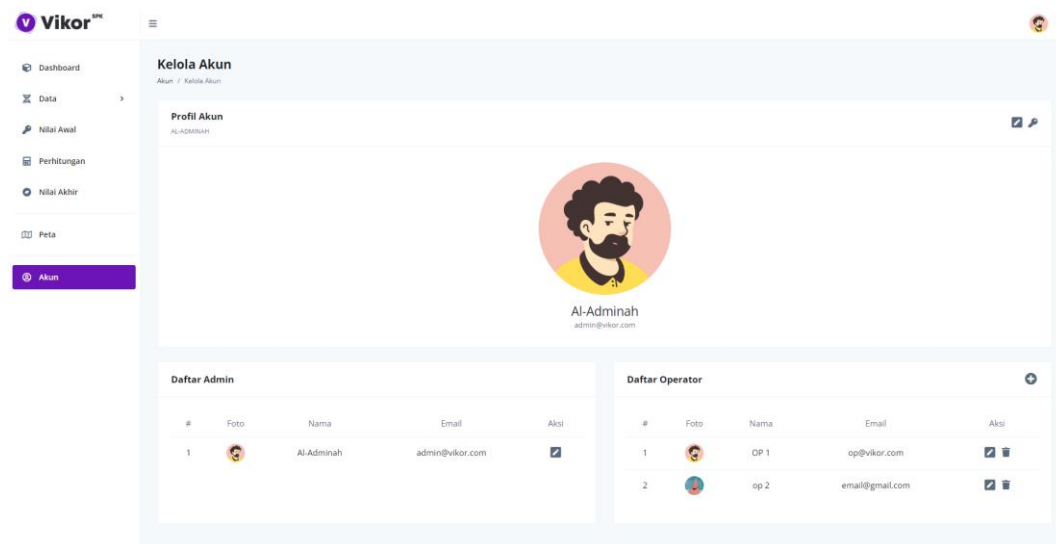
File hubungan antara *model*, *view*, dan *controller* halaman profil ditunjukkan pada tabel 4.10.

Tabel 4.10 Tabel MVC halaman profil

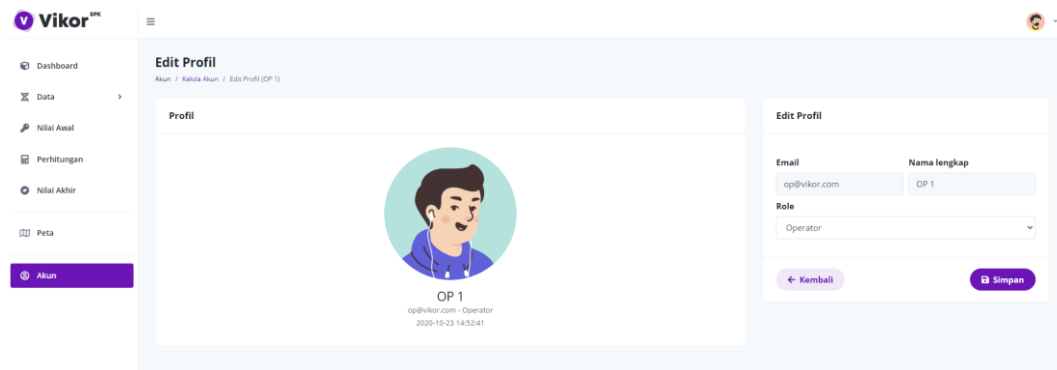
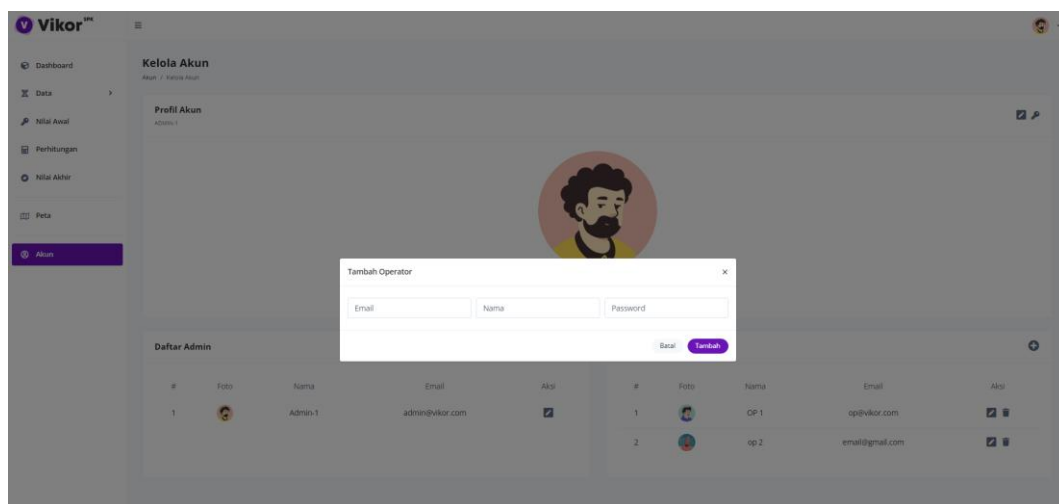
MVC	Nama File	Fungsi
Model	Akun_model.php	Memanggil data tabel <i>user</i> dari basis data untuk kemudian dikirimkan ke <i>controller</i> untuk diproses.
View	operator/akun.php	Menampilkan halaman profil operator
Controller	Operator.php	Menerima data dari model dan selanjutnya ditampilkan ke dalam <i>view</i> serta mengkoordinasi parameter saat terjadi perubahan pada data akun.

11. Tampilan Halaman Akun

Halaman akun merupakan halaman yang menampilkan informasi dari tabel *user*. Administrator dapat melihat dan memperbarui akunnya sendiri seperti memperbarui nama, foto profil, dan *password* yang mana form untuk memperbarui data profil akan ditampilkan dalam bentuk *pop-up modal*. Administrator juga dapat menambah, menghapus, dan memperbarui *role* dari akun terdaftar. Halaman akun ditunjukkan pada gambar 4.31 sampai gambar 4.33.



Gambar 4.31 Halaman akun administrator

Gambar 4.32 Halaman ubah *role* akun terdaftar

Gambar 4.33 Form tambah operator

File hubungan antara *model*, *view*, dan *controller* halaman akun ditunjukkan pada tabel 4.11.

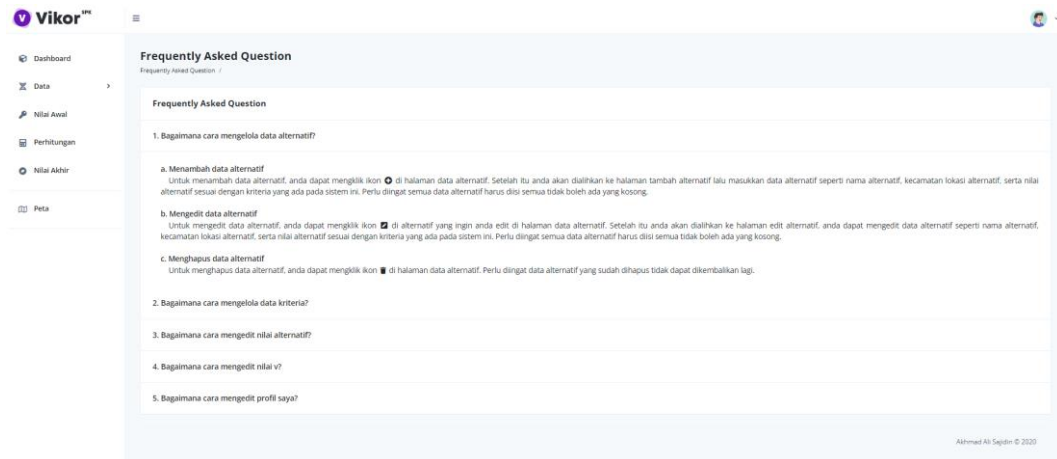
Tabel 4.11 Tabel MVC halaman akun

<i>MVC</i>	Nama File	Fungsi
<i>Model</i>	Akun_model.php	Memanggil data tabel <i>user</i> dari basis data untuk kemudian dikirimkan ke <i>controller</i> untuk diproses.
<i>View</i>	admin/akun.php admin/edituser.php	Menampilkan halaman akun administrator dan seluruh data akun terdaftar serta halaman edit akun.
<i>Controller</i>	Admin.php	Menerima data dari model dan selanjutnya ditampilkan ke dalam <i>view</i> serta mengkoordinasi parameter saat terjadi perubahan pada data akun.

12. Tampilan Halaman FAQ (*Frequently Asked Question*)

Halaman FAQ atau *Frequently Asked Question* merupakan halaman yang digunakan untuk menampilkan jawaban dari pertanyaan-pertanyaan yang sering

ditanyakan mengenai sistem informasi pendukung keputusan ini. Halaman *FAQ* ditunjukkan pada gambar 4.34.



Gambar 4.34 Halaman FAQ

File hubungan antara *model*, *view*, dan *controller* halaman FAQ ditunjukkan pada tabel 4.12.

Tabel 4.12 Tabel MVC halaman FAQ

<i>MVC</i>	Nama File	Fungsi
<i>Model</i>	Akun_model.php	Memanggil data tabel <i>user</i> dari basis data untuk kemudian dikirimkan ke <i>controller</i> untuk diproses.
<i>View</i>	faq.php	Menampilkan halaman <i>FAQ</i> yang berisi pertanyaan serta jawaban yang umum ditanyakan mengenai sistem informasi.
<i>Controller</i>	Faq.php	Menerima data dari model dan selanjutnya ditampilkan ke dalam view halaman <i>FAQ</i> .

4.2 Pengujian Sistem

Tahap pengujian sistem dilakukan untuk menguji sistem pendukung keputusan sebelum dipublikasikan. Pengujian sistem dilakukan dengan menggunakan metode *Black box*. Pengujian dilakukan untuk mengetahui apakah sistem yang sudah dibuat mampu berjalan sesuai dengan rancangan pengembangan sistem atau belum. Penggunaan metode *Black box* bertujuan untuk mengetahui apakah setiap fungsi dari sistem pendukung keputusan sudah dapat berjalan (secara fungsional) dengan baik atau belum.

Pada pengujian sistem diperlukan untuk indikator untuk tiap bagian yang diuji. Pengujian dilakukan pada tiap bagian termasuk fungsi dari menu, form, dan tombol yang ada di dalam sistem. Pengujian dimulai dengan pengujian kebutuhan

fungsional sistem. Berikut adalah tabel pengujian kebutuhan sistem yang ditunjukkan oleh tabel 4.13.

Tabel 4.13 Tabel pengujian fungsional sistem

No.	Pengujian Fungsional	Keterangan
1	Tersedia halaman awal (<i>landing page</i>) sebagai halaman utama dan sekaligus halaman yang menampilkan informasi bagi <i>guest</i> pada sistem	Tersedia
2	Tersedia halaman <i>login</i> untuk melakukan autentikasi pengguna dan menentukan tingkatan pengguna (<i>role</i>) pada sistem	Tersedia
3	Tersedia halaman-halaman administrator yang hanya pengguna dengan tingkatan pengguna (<i>role</i>) administrator yang dapat mengaksesnya	Tersedia
4	Tersedia halaman-halaman operator yang hanya pengguna dengan tingkatan pengguna (<i>role</i>) operator yang dapat mengaksesnya	Tersedia
5	Tersedia halaman untuk melakukan perubahan data alternatif, data kriteria, data nilai, dan data akun	Tersedia
6	Tersedia halaman yang menampilkan perhitungan VIKOR secara detail dan halaman yang menampilkan peta visualisasi dari lokasi alternatif lokasi embung	Tersedia

Setelah dilakukan pengujian fungsional terhadap sistem, tahap selanjutnya adalah melakukan pengujian pada setiap halaman.

1. Pengujian Halaman Awal (*Landing Page*)

Pengujian pada halaman awal dilakukan dengan membuka sistem pertama kali, apakah sistem berhasil mengarah ke halaman awal atau tidak. Hasil pengujian halaman awal (*landing page*) ditunjukkan pada tabel 4.14.

Tabel 4.14 Tabel pengujian halaman awal (*landing page*)

Nama Pengujian	Bentuk Pengujian	Hasil yang Diharapkan	Hasil Pengujian
Pengujian buka sistem	Mengakses sistem melalui <i>address bar</i>	Menampilkan halaman awal	Berhasil
Pengujian masuk <i>platform</i>	Melakukan klik pada tombol <i>login</i>	Masuk ke halaman <i>login</i>	Berhasil
Pengujian <i>link</i> Metode	Melakukan klik pada tombol Metode	Menampilkan halaman awal (<i>landing page</i>) bagian Metode	Berhasil
Pengujian <i>link</i> Sistem	Melakukan klik pada tombol Sistem	Menampilkan halaman awal (<i>landing page</i>) bagian Sistem	Berhasil
Pengujian <i>link</i> Data	Melakukan klik pada tombol Data	Menampilkan halaman awal (<i>landing page</i>) bagian Data	Berhasil
Pengujian tombol Alternatif	Melakukan klik pada tombol lihat data alternatif	Menampilkan pop-up data alternatif	Berhasil
Pengujian tombol Kriteria	Melakukan klik pada tombol lihat data kriteria	Menampilkan pop-up data kriteria	Berhasil

Pengujian tombol Ranking	Melakukan klik pada tombol lihat data ranking	Menampilkan pop-up data perangkingan	Berhasil
--------------------------	---	--------------------------------------	----------

2. Pengujian Halaman *Login*

Pengujian pada halaman *login* dilakukan dengan melakukan proses autentikasi dan pemeriksaan tingkatan pengguna (*role*) yang akan memasuki sistem. Hasil pengujian halaman *login* ditunjukkan pada tabel 4.15.

Tabel 4.15 Tabel pengujian halaman *login*

Nama Pengujian	Bentuk Pengujian	Hasil yang Diharapkan	Hasil Pengujian
Pengujian masuk ke halaman <i>login</i>	Mengakses halaman <i>login</i> melalui tombol <i>login</i> di halaman awal	Menampilkan halaman <i>login</i>	Berhasil
Pengujian masuk ke sistem	Melakukan pengisian username dan password serta menekan tombol <i>login</i> di halaman <i>login</i>	Masuk ke dashboard sesuai dengan tingkatan pengguna yang melakukan <i>login</i>	Berhasil

3. Pengujian Halaman *Dashboard*

Pengujian pada halaman *dashboard* dilakukan dengan menguji apakah beranda menampilkan data yang sesuai. Konten halaman *dashboard* antara administrator dan operator akan terdapat sedikit perbedaan. Konten halaman operator yaitu data jumlah alternatif dan kriteria, hasil perangkingan, dan visualisasi peta, sedangkan halaman administrator menunjukkan konten yang sama dengan halaman *dashboard* operator tetapi terdapat konten tambahan yaitu data jumlah admin dan jumlah operator. Hasil pengujian halaman *dashboard* ditunjukkan pada tabel 4.16 dan tabel 4.17.

Tabel 4.16 Tabel pengujian halaman *dashboard* administrator

Nama Pengujian	Bentuk Pengujian	Hasil yang Diharapkan	Hasil Pengujian
Pengujian masuk ke <i>dashboard</i>	Mengakses melalui tombol “ <i>Login</i> ” di halaman awal (<i>landing page</i>)	Menampilkan halaman <i>dashboard</i> administrator	Berhasil
Pengujian peta, data akun, dan data alternatif	Mengakses halaman <i>dashboard</i> administrator dan memeriksa konten yang ditampilkan	Menampilkan peta, data lokasi alternatif embung, dan jumlah data akun yang terdaftar	Berhasil
Pengujian keluar dari sistem	Melakukan klik tombol “keluar” pada <i>dropdown</i> di <i>navigation bar</i>	Keluar dari sistem dan kembali ke halaman <i>login</i>	Berhasil

Tabel 4.17 Tabel pengujian halaman *dashboard* operator

Nama Pengujian	Bentuk Pengujian	Hasil yang Diharapkan	Hasil Pengujian
Pengujian masuk ke <i>dashboard</i>	Mengakses melalui tombol “Login” di halaman awal (<i>landing page</i>)	Menampilkan halaman <i>dashboard</i> operator	Berhasil
Pengujian petadan data alternatif	Mengakses halaman <i>dashboard</i> operator dan memeriksa konten yang ditampilkan	Menampilkan peta dan data lokasi alternatif embung	Berhasil
Pengujian keluar dari sistem	Melakukan klik tombol “keluar” pada <i>dropdown</i> di <i>navigation bar</i>	Keluar dari sistem dan kembali ke halaman <i>login</i>	Berhasil

4. Pengujian Halaman Alternatif

Pengujian pada halaman alternatif dilakukan dengan menampilkan halaman alternatif yang berisi data alternatif dan menguji seluruh aksi yang ada di dalamnya. Hasil pengujian halaman alternatif ditunjukkan pada tabel 4.18.

Tabel 4.18 Tabel pengujian halaman alternatif

Nama Pengujian	Bentuk Pengujian	Hasil yang Diharapkan	Hasil Pengujian
Pengujian halaman alternatif	Mengakses halaman alternatif melalui menu di <i>sidebar</i>	Menampilkan halaman alternatif beserta isi data alternatif	Berhasil
Pengujian aksi tambah, ubah, dan hapus data alternatif	Mengakses melalui tombol tambah, edit, atau hapus di halaman alternatif	Menambah, mengubah, dan menghapus data alternatif	Berhasil
Pengujian keluar dari sistem	Melakukan klik tombol “keluar” pada <i>dropdown</i> di <i>navigation bar</i>	Keluar dari sistem dan kembali ke halaman <i>login</i>	Berhasil

5. Pengujian Halaman Kriteria dan Parameter

Pengujian pada halaman kriteria dilakukan dengan menampilkan halaman kriteria dan parameter yang berisi data kriteria dan parameter dan menguji seluruh aksi di dalamnya. Hasil pengujian halaman kriteria dan parameter ditunjukkan pada tabel 4.19.

Tabel 4.19 Tabel pengujian halaman kriteria dan parameter

Nama Pengujian	Bentuk Pengujian	Hasil yang Diharapkan	Hasil Pengujian
Pengujian halaman kriteria dan parameter	Mengakses halaman kriteria melalui menu di <i>sidebar</i>	Menampilkan halaman kriteria dan parameter beserta isi data kriteria dan parameternya	Berhasil

Pengujian aksi tambah, ubah, dan hapus data kriteria berparameter dan tidak berparameter	Mengakses melalui tombol tambah, edit, atau hapus di halaman kriteria dan parameter	Menambah, mengubah, dan menghapus data kriteria dan parameter	Berhasil
Pengujian keluar dari sistem	Melakukan klik tombol “keluar” pada <i>dropdown</i> di <i>navigation bar</i>	Keluar dari sistem dan kembali ke halaman <i>login</i>	Berhasil

6. Pengujian Halaman Nilai Awal

Pengujian pada halaman nilai awal dilakukan dengan menampilkan halaman nilai dengan isi data nilai awal tiap alternatif terhadap kriteria dan nilai V serta menguji seluruh aksi di dalamnya. Hasil pengujian halaman nilai awal ditunjukkan pada tabel 4.20.

Tabel 4.20 Tabel pengujian halaman nilai awal

Nama Pengujian	Bentuk Pengujian	Hasil yang Diharapkan	Hasil Pengujian
Pengujian halaman nilai awal	Mengakses halaman nilai awal melalui menu di <i>sidebar</i>	Menampilkan halaman nilai awal yang berisi data nilai awal alternatif dan nilai V	Berhasil
Pengujian aksi ubah nilai awal	Mengakses melalui tombol edit di halaman nilai awal	Mengubah data nilai awal	Berhasil
Pengujian keluar dari sistem	Melakukan klik tombol “keluar” pada <i>dropdown</i> di <i>navigation bar</i>	Keluar dari sistem dan kembali ke halaman <i>login</i>	Berhasil

7. Pengujian Halaman Perhitungan

Pengujian pada halaman perhitungan dilakukan dengan menampilkan halaman perhitungan dengan isi data tiap tahap perhitungan dan menguji setiap aksi di dalamnya. Hasil pengujian halaman perhitungan ditunjukkan pada tabel 4.21.

Tabel 4.21 Tabel pengujian halaman perhitungan

Nama Pengujian	Bentuk Pengujian	Hasil yang Diharapkan	Hasil Pengujian
Pengujian halaman perhitungan	Mengakses halaman perhitungan melalui menu di <i>sidebar</i>	Menampilkan halaman perhitungan yang berisi data tiap tahap perhitungan	Berhasil
Pengujian aksi cetak perhitungan	Mengakses melalui tombol cetak di halaman perhitungan	Mencetak perhitungan	Berhasil
Pengujian keluar dari sistem	Melakukan klik tombol “keluar” pada <i>dropdown</i> di <i>navigation bar</i>	Keluar dari sistem dan kembali ke halaman <i>login</i>	Berhasil

8. Pengujian Halaman Nilai Akhir

Pengujian pada halaman nilai akhir dilakukan dengan menampilkan halaman nilai akhir dari perhitungan dengan isi hasil dari perhitungan metode VIKOR dan menguji setiap aksi di dalamnya. Hasil pengujian halaman nilai akhir ditunjukkan pada tabel 4.22.

Tabel 4.22 Tabel pengujian halaman nilai akhir

Nama Pengujian	Bentuk Pengujian	Hasil yang Diharapkan	Hasil Pengujian
Pengujian halaman nilai akhir	Mengakses halaman nilai akhir melalui menu di <i>sidebar</i>	Menampilkan halaman nilai akhir yang berisi data dari hasil perhitungan	Berhasil
Pengujian keluar dari sistem	Melakukan klik tombol “keluar” pada <i>dropdown</i> di <i>navigation bar</i>	Keluar dari sistem dan kembali ke halaman <i>login</i>	Berhasil

9. Pengujian Halaman Peta

Pengujian pada halaman peta dilakukan dengan menampilkan halaman peta yang berisi visualisasi data alternatif yang terdaftar. Hasil pengujian halaman peta ditunjukkan pada tabel 4.23.

Tabel 4.23 Tabel pengujian halaman peta

Nama Pengujian	Bentuk Pengujian	Hasil yang Diharapkan	Hasil Pengujian
Pengujian halaman peta	Mengakses halaman peta melalui menu di <i>sidebar</i>	Menampilkan halaman peta yang berisi visualisasi peta data alternatif terdaftar	Berhasil
Pengujian keluar dari sistem	Melakukan klik tombol “keluar” pada <i>dropdown</i> di <i>navigation bar</i>	Keluar dari sistem dan kembali ke halaman <i>login</i>	Berhasil

10. Pengujian Halaman Profil

Pengujian pada halaman profil dilakukan dengan menampilkan halaman profil yang berisi data profil pribadi pengguna dan menguji setiap aksi di dalamnya. Hasil pengujian halaman profil ditunjukkan pada tabel 4.24.

Tabel 4.24 Tabel pengujian halaman profil

Nama Pengujian	Bentuk Pengujian	Hasil yang Diharapkan	Hasil Pengujian
Pengujian halaman profil	Melakukan klik tombol “lihat profil” pada <i>dropdown</i> di <i>navigation bar</i>	Menampilkan halaman profil yang berisi data profil pribadi pengguna	Berhasil

Pengujian ubah <i>password</i> dan ubah profil pengguna	Mengakses tombol edit dan edit <i>password</i> di halaman profil	Memperbarui <i>password</i> dan profil pengguna	Berhasil
Pengujian keluar dari sistem	Melakukan klik tombol “keluar” pada <i>dropdown</i> di <i>navigation bar</i>	Keluar dari sistem dan kembali ke halaman <i>login</i>	Berhasil

11. Pengujian Halaman Akun

Pengujian pada halaman akun dilakukan dengan menampilkan halaman akun yang berisi seluruh akun terdaftar dan menguji setiap aksi di dalamnya. Hasil pengujian halaman akun ditunjukkan pada tabel 4.25.

Tabel 4.25 Tabel pengujian halaman akun

Nama Pengujian	Bentuk Pengujian	Hasil yang Diharapkan	Hasil Pengujian
Pengujian masuk ke halaman akun	Mengakses halaman akun melalui menu di <i>sidebar</i>	Menampilkan halaman akun beserta isi data akun	Berhasil
Pengujian aksi tambah, ubah, dan hapus data akun	Mengakses melalui tombol yang tersedia di halaman akun	Menambah, mengubah, dan menghapus data akun	Berhasil
Pengujian keluar dari sistem	Melakukan klik tombol “keluar” pada <i>dropdown</i> di <i>navigation bar</i>	Keluar dari sistem dan kembali ke halaman <i>login</i>	Berhasil

12. Pengujian Halaman FAQ

Pengujian pada halaman FAQ (*Frequently Asked Question*) dilakukan dengan menampilkan halaman FAQ yang berisi sejumlah pertanyaan mengenai sistem yang disertai dengan penjelasan jawaban pertanyaan-pertanyaan di atas. Hasil pengujian halaman FAQ ditunjukkan pada tabel 4.26.

Tabel 4.26 Tabel pengujian halaman FAQ

Nama Pengujian	Bentuk Pengujian	Hasil yang Diharapkan	Hasil Pengujian
Pengujian halaman FAQ	Melakukan klik tombol “FAQ” pada <i>dropdown</i> di <i>navigation bar</i>	Menampilkan halaman FAQ yang berisi pertanyaan terkait sistem dan jawabannya	Berhasil
Pengujian keluar dari sistem	Melakukan klik tombol “keluar” pada <i>dropdown</i> di <i>navigation bar</i>	Keluar dari sistem dan kembali ke halaman <i>login</i>	Berhasil

4.3 Pengujian Metode VIKOR

Pengujian metode VIKOR dilakukan secara manual untuk melakukan proses validasi terhadap rumus-rumus dari metode VIKOR yang telah diterapkan dalam sistem. Berikut adalah tahapan penggunaan metode VIKOR untuk mengolah data penentuan prioritas lokasi pembangunan embung di Kabupaten Semarang:

1. Penentuan Kriteria dan Parameter

Tahap pertama disusun data kriteria meliputi kode kriteria, nama kriteria, tren kriteria, parameter tiap kriteria dan nilai parameternya jika kriteria memiliki parameter. Data-data tersebut ditunjukkan pada tabel 4.27.

Tabel 4.27 Tabel kriteria dan parameter

K1 (Benefit)	K2 (Cost)	K3 (Cost)	K4 (Benefit)	K5 (Benefit)	K6 (Cost)	K7 (Benefit)	Nilai Parameter
Perkampungan	(Kriteria tidak berparameter)	(Kriteria tidak berparameter)	(Kriteria tidak berparameter)	(Kriteria tidak berparameter)	(Kriteria tidak berparameter)	Tidak tersedia jalan	1
Sawah tadah hujan						Jalan setapak	2
Ladang/tegalan						Jalan makadam/tanah	3
Semak belukar						Tersedia jalan aspal	4
Hutan						-	5

Keterangan kode kriteria:

- K1 : Vegetasi area genangan embung
- K2 : Volume material timbunan (m^3)
- K3 : Luas daerah yang akan dibebaskan (Ha)
- K4 : Volume tampungan efektif (m^3)
- K5 : Lama operasi (Hari)
- K6 : Harga air/ m^3 (Rupiah)
- K7 : Akses jalan menuju *site* bendungan

2. Penentuan Alternatif

Pada tahap ini tentukan alternatif yang akan dilakukan perhitungan beserta dengan nilainya terhadap masing-masing kriteria berdasarkan data-data yang sudah ada[10]. Tabel data nilai alternatif ditunjukkan pada tabel 4.28.

Tabel 4.28 Tabel nilai alternatif

Nama Alternatif	K1	K2	K3	K4	K5	K6	K7
Dadapayam	Sawah Tadah Hujan	7.280	4,2	538.922,4	57	30.333,00	jalan setapak
Mluweh	Hutan	196.390	2,2	3.172.333,3	113	8.322,59	jalan makadam/ tanah
Lebak	Sawah Tadah Hujan	99.140	2,4	783.975,8	57	8.335,12	jalan setapak
Pakis	Sawah Tadah Hujan	11.430	3,4	1.346.651,1	57	10.092,48	jalan setapak
Jatikurung	Hutan	29.280	5,3	39.039,7	10	375.650,85	jalan setapak
Gogodalem	Hutan	54.722,35	7,3	318.778,0	63	74.434,54	jalan setapak
Kandangan	Ladang/ tegalan	46.406,3	2,8	35.907,0	2	549.291,92	jalan setapak
Ngrawan	Ladang/ tegalan	28.740	4,3	18.750,0	22	858.700,26	jalan makadam/ tanah

3. Menyusun Matriks Keputusan (F)

Pada tahap ini setiap alternatif lokasi di Kabupaten Semarang dan kriteria disusun ke dalam bentuk matriks keputusan. Pemberian nilai dari tiap alternatif terhadap kriteria berparameter didasarkan pada parameter yang sudah disusun sebelumnya. Matriks keputusan alternatif terhadap kriteria ditunjukkan pada tabel 4.29.

Tabel 4.29 Matriks keputusan (F)

Kode	Nama Alternatif	K1	K2	K3	K4	K5	K6	K7
A01	Dadapayam	2	7.280	4,2	538.922,4	57	30.333,00	2
A02	Mluweh	5	196.390	2,2	3.172.333,3	113	8.322,59	3
A03	Lebak	2	99.140	2,4	783.975,8	57	8.335,12	2
A04	Pakis	2	11.430	3,4	1.346.651,1	57	10.092,48	2
A05	Jatikurung	5	29.280	5,3	39.039,7	10	375.650,85	2
A06	Gogodalem	5	54.722,35	7,3	318.778,0	63	74.434,54	2
A07	Kandangan	3	46.406,3	2,8	35.907,0	2	549.291,92	2
A08	Ngrawan	3	28.740	4,3	18.750,0	22	858.700,26	3

4. Penentuan Bobot Kriteria (W)

Pada tahap ini setiap kriteria diberikan bobot berdasarkan data yang telah ada. Bobot tiap kriteria ditunjukkan pada tabel 4.30.

Tabel 4.30 Bobot kriteria (W)

Kode Kriteria	K1 (Benefit)	K2 (Cost)	K3 (Cost)	K4 (Benefit)	K5 (Benefit)	K6 (Cost)	K7 (Benefit)
Bobot Kriteria	0,12753	0,09545	0,25151	0,13366	0,15971	0,12972	0,10242

5. Menghitung Matriks Normalisasi (N)

Membuat matriks normalisasi dengan menentukan nilai positif (f_j^+) dan nilai negatif (f_j^-) sebagai solusi ideal untuk setiap kriteria terlebih dahulu menggunakan persamaan berikut:

- Jika kriteria memiliki tren *benefit*, maka menggunakan fungsi berikut:

$$f_j^+ = \max(f_{1j}, f_{2j}, f_{3j}, \dots, f_{ij})$$

$$f_j^- = \min(f_{1j}, f_{2j}, f_{3j}, \dots, f_{ij})$$

- Jika kriteria memiliki tren *cost*, maka menggunakan fungsi berikut:

$$f_j^+ = \min(f_{1j}, f_{2j}, f_{3j}, \dots, f_{ij})$$

$$f_j^- = \max(f_{1j}, f_{2j}, f_{3j}, \dots, f_{ij})$$

Perhitungan nilai positif (f_j^+) dan nilai negatif (f_j^-) untuk kriteria K1 (*benefit*) sebagai berikut:

$$\begin{aligned} f_1^+ &= \max(f_{1,1}, f_{2,1}, f_{3,1}, \dots, f_{8,1}) & f_1^- &= \min(f_{1,1}, f_{2,1}, f_{3,1}, \dots, f_{8,1}) \\ &= \max(2, 5, 2, 2, 5, 5, 2) & &= \min(2, 5, 2, 2, 5, 5, 2) \\ &= 5 & &= 2 \end{aligned}$$

Setelah dilakukan perhitungan nilai positif (f_j^+) dan nilai negatif (f_j^-) pada kriteria K2 hingga kriteria K7, hasil perhitungan nilai positif (f_j^+) dan nilai negatif (f_j^-) seluruh kriteria ditunjukkan pada tabel 4.31.

Tabel 4.31 Hasil perhitungan nilai positif (f_j^+) dan nilai negatif (f_j^-)

Kode Kriteria	K1 (<i>Benefit</i>)	K2 (<i>Cost</i>)	K3 (<i>Cost</i>)	K4 (<i>Benefit</i>)	K5 (<i>Benefit</i>)	K6 (<i>Cost</i>)	K7 (<i>Benefit</i>)
f_j^+	5	7.280	2,2	3.172.333,3	113	8.322,59	3
f_j^-	2	196.390	7,3	18.750	2	858.700,26	2

Setelah didapat nilai positif (f_j^+) dan nilai negatif (f_j^-) untuk setiap kriteria, tahap selanjutnya adalah menghitung normalisasi matriks keputusan N untuk masing-masing nilai alternatif dengan persamaan berikut:

$$N_{ij} = \frac{(f_j^+ - f_{ij})}{(f_j^+ - f_j^-)}$$

Untuk semua alternatif pada kriteria K1 dihitung nilai normalisasi dari $N_{1,1}$ sampai $N_{8,1}$ sebagai berikut:

$$N_{1,1} = \frac{(f_1^+ - f_{1,1})}{(f_1^+ - f_1^-)} = \frac{(5 - 2)}{(5 - 2)} = 1$$

dan seterusnya hingga:

$$N_{8,1} = \frac{(f_1^+ - f_{8,1})}{(f_1^+ - f_1^-)} = \frac{(5 - 3)}{(5 - 2)} = 0,6667$$

Setelah dilakukan perhitungan ke semua alternatif untuk kriteria K2 hingga kriteria K7 ($N_{1,2}, \dots, N_{8,2}$ s/d $N_{1,7}, \dots, N_{8,7}$) diperoleh hasil matriks normalisasi yang ditunjukkan pada tabel 4.32.

Tabel 4.32 Hasil perhitungan normalisasi (N)

Kode	Nama Alternatif	K1	K2	K3	K4	K5	K6	K7
A01	Dadapayam	1	0	0,3922	0,8351	0,5045	0,0259	1
A02	Mluweh	0	1	0	0	0	0	0
A03	Lebak	1	0,4857	0,0392	0,7573	0,5045	0	1
A04	Pakis	1	0,0219	0,2353	0,5789	0,5045	0,0021	1
A05	Jatikurung	0	0,1163	0,6078	0,9936	0,9279	0,432	1
A06	Gogodalem	0	0,2509	1	0,9049	0,4505	0,0777	1
A07	Kandangan	0,6667	0,2069	0,1176	0,9946	1	0,6362	1
A08	Ngrawan	0,6667	0,1135	0,4118	1	0,8198	1	0

6. Menghitung Normalisasi Bobot (F^*)

Pada tahap ini nilai tiap alternatif yang sudah ternormalisasi (N_{ij}) dikalikan dengan nilai bobot masing-masing kriteria (W_j) yang telah ditentukan dengan persamaan berikut:

$$F_{ij}^* = W_j \cdot N_{ij}$$

Hasil perhitungan normalisasi bobot ditunjukkan pada tabel 4.33.

Tabel 4.33 Hasil perhitungan normalisasi bobot (F^*)

Kode	Nama Alternatif	K1	K2	K3	K4	K5	K6	K7
A01	Dadapayam	0,1275	0	0,0986	0,1116	0,0806	0,0034	0,1024
A02	Mluweh	0	0,0955	0	0	0	0	0
A03	Lebak	0,1275	0,0464	0,0099	0,1012	0,0806	0	0,1024
A04	Pakis	0,1275	0,0021	0,0592	0,0774	0,0806	0,0003	0,1024
A05	Jatikurung	0	0,0111	0,1529	0,1328	0,1482	0,056	0,1024
A06	Gogodalem	0	0,0239	0,2515	0,1209	0,0719	0,0101	0,1024
A07	Kandangan	0,085	0,0197	0,0296	0,1329	0,1597	0,0825	0,1024
A08	Ngrawan	0,085	0,0108	0,1036	0,1337	0,1309	0,1297	0

7. Menghitung Nilai *Utility Measure* (S_i) dan *Regret Measure* (R_i)

Pada tahap ini dihitung nilai *utility measure* (S_i) dan nilai *regret measure* (R_i) untuk setiap alternatif dengan menggunakan nilai F^* yang didapat dari perhitungan sebelumnya. Untuk menghitung S_i dan R_i digunakan persamaan berikut:

$$S_i = \sum_{j=1}^n F_{ij}^*$$

$$R_i = \max_j [F_{ij}^*]$$

Untuk alternatif A01 dihitung nilai *utility measure* (S_i) dan nilai *regret measure* (R_i) sebagai berikut:

$$S_1 = F_{1,1}^* + F_{1,2}^* + F_{1,3}^* + F_{1,4}^* + F_{1,5}^* + F_{1,7}^* + F_{1,8}^*$$

$$S_1 = 0,1275 + 0 + 0,0986 + 0,1116 + 0,0806 + 0,0034 + 0,1024$$

$$S_1 = 0,5241$$

$$R_1 = \max[0,1275; 0; 0,0986; 0,1116; 0,0806; 0,0034; 0,1024]$$

$$R_1 = 0,1275$$

dan seterusnya hingga alternatif A08.

Hasil dari perhitungan nilai *utility measure* (S_i) dan nilai *regret measure* (R_i) ke masing-masing alternatif ditunjukkan pada tabel 4.34.

Tabel 4.34 Hasil perhitungan nilai *utility measure* (S_i) dan *regret measure* (R_i)

Kode	Nama Alternatif	Nilai <i>Utility Measure</i> (S_i)	Nilai <i>Regret Measure</i> (R_i)
A01	Dadapayam	0,5241	0,1275
A02	Mluweh	0,0955	0,0955
A03	Lebak	0,468	0,1275
A04	Pakis	0,4495	0,1275
A05	Jatikurung	0,6034	0,1529
A06	Gogodalem	0,5807	0,2515
A07	Kandangan	0,6118	0,1597
A08	Ngrawan	0,5937	0,1337

8. Menghitung Nilai Indeks VIKOR (Q)

Sebelum menghitung nilai indeks VIKOR dari tiap alternatif, perlu dihitung dahulu nilai S^+ , S^- , R^+ , dan R^- dengan menggunakan persamaan berikut:

$$S^+ = \max_i(S_i)$$

$$S^- = \min_i(S_i)$$

$$R^+ = \max_i(R_i)$$

$$R^- = \min_i(R_i)$$

Hasil dari perhitungan dari nilai S^+ , S^- , R^+ , dan R^- ditunjukkan pada tabel 4.35.

Tabel 4.35 Hasil perhitungan S^+ , S^- , R^+ , dan R^-

	Nilai <i>Utility Measure</i> (S)	Nilai <i>Regret Measure</i> (R)
Nilai Maksimal ($^+$)	0,6118	0,2515
Nilai Minimal ($^-$)	0,0955	0,0955

Setelah didapat nilai S^+ , S^- , R^+ , dan R^- selanjutnya dilakukan perhitungan untuk mencari nilai indeks VIKOR tiap alternatif. Untuk mencari nilai indeks VIKOR (Q) digunakan persamaan berikut:

$$Q_i = V \left[\frac{(S_i - S^-)}{(S^+ - S^-)} \right] + (1 - V) \left[\frac{(R_i - R^-)}{(R^+ - R^-)} \right]$$

Untuk alternatif A01 sampai alternatif A08 dilakukan perhitungan nilai indeks VIKOR sebagai berikut:

$$Q_1 = 0,5 \left[\frac{(0,5241 - 0,0955)}{(0,6118 - 0,0955)} \right] + (1 - 0,5) \left[\frac{(0,1275 - 0,0955)}{(0,2515 - 0,0955)} \right]$$

$$Q_1 = 0,5[0,8301] + (0,5)[0,2051]$$

$$Q_1 = 0,415 + 0,1026$$

$$Q_1 = 0,5176$$

...

$$Q_8 = 0,5 \left[\frac{(0,5937 - 0,0955)}{(0,6118 - 0,0955)} \right] + (1 - 0,5) \left[\frac{(0,1337 - 0,0955)}{(0,2515 - 0,0955)} \right]$$

$$Q_8 = 0,5[0,5163] + (0,5)[0,156]$$

$$Q_8 = 0,48247 + 0,12243$$

$$Q_8 = 0,6049$$

Setelah dihitung nilai indeks VIKOR (Q) dari masing-masing alternatif didapatkan nilai yang ditunjukkan pada tabel 4.36.

Tabel 4.36 Hasil perhitungan nilai indeks VIKOR (Q_i)

Kode Alternatif	Nama Alternatif	Nilai Indeks VIKOR ($V=0,5$)
A01	Dadapayam	0,5176
A02	Mluweh	0
A03	Lebak	0,4633
A04	Pakis	0,4454
A05	Jatikurung	0,6758
A06	Gogodalem	0,9699
A07	Kandangan	0,7058
A08	Ngrawan	0,6049

9. Perangkingan Alternatif

Perangkingan alternatif ditentukan dari nilai indeks VIKOR (Q), alternatif dengan nilai yang paling rendah merupakan solusi ideal. Hasil perangkingan berdasarkan nilai indeks VIKOR ditunjukkan pada tabel 4.37.

Tabel 4.37 Perangkingan alternatif berdasarkan nilai indeks VIKOR

Rank	Kode Alternatif	Nama Alternatif	Nilai Indeks VIKOR ($V=0,5$)
1	A02	Mluweh	0
2	A04	Pakis	0,4454
3	A03	Lebak	0,4633
4	A01	Dadapayam	0,5176
5	A08	Ngrawan	0,6049
6	A05	Jatikurung	0,6758
7	A07	Kandangan	0,7058
8	A06	Gogodalem	0,9699

10. Mengajukan Solusi Kompromi

Solusi kompromi ditentukan dari alternatif yang memiliki peringkat terbaik dengan mengukur indeks VIKOR yang minimum dengan mengujinya dengan 2 kondisi berikut:

- Pengujian Kondisi 1: *Acceptable Advantage*

Menghitung selisih antara peringkat alternatif pertama dan kedua yakni $Q_{(a_1)}$ dan $Q_{(a_2)}$ lalu membandingkannya dengan nilai DQ . Jika nilai selisih alternatif peringkat pertama dan kedua yang didapat lebih besar atau sama dengan nilai DQ , maka kondisi *acceptable advantage* terpenuhi. Persamaan dari kondisi dapat dituliskan sebagai berikut:

$$Q_{(a_2)} - Q_{(a_1)} \geq DQ$$

$$DQ = \frac{1}{m-1}$$

Pengujian terhadap alternatif terbaik pada kondisi *acceptable advantage* sebagai berikut:

$$Q_{(a_2)} - Q_{(a_1)} = 0,4454 - 0 = 0,4454$$

$$DQ = \frac{1}{m-1} = \frac{1}{8-1} = 0,1429$$

Dari perhitungan di atas diperoleh nilai DQ adalah 0,1429 dan selisih nilai $Q_{(a_1)}$ dan $Q_{(a_2)}$ adalah 0,4454. Dikarenakan nilai $Q_{(a_2)} - Q_{(a_1)} \geq DQ$, maka kondisi *acceptable advantage* terpenuhi.

- Pengujian Kondisi 2: *Acceptable Stability in Decision Making*

Menguji stabilitas perangkingan alternatif dengan menggunakan nilai V yang berbeda yakni: nilai $V > 0,5$ (*voting by majority rule*), nilai $V = 0,5$ (*by concensus*), dan nilai $V < 0,5$ (*with veto*). Jika alternatif peringkat pertama atau $Q_{(a_1)}$ tetap menjadi peringkat terbaik dalam 3 macam perangkingan dengan nilai V yang berbeda, maka kondisi *acceptable stability in decision making* terpenuhi.

Pengujian kondisi *acceptable stability in decision making* terhadap alternatif ditunjukkan pada tabel 4.38.

Tabel 4.38 Pengujian kondisi *acceptable stability in decision making*

Rank	(V=0,45)		(V=0,5)		(V=0,55)	
	Kode Alt	Q	Kode Alt	Q	Kode Alt	Q
1	A02	0	A02	0	A02	0
2	A04	0,4214	A04	0,4454	A04	0,4694
3	A03	0,4375	A03	0,4633	A03	0,4891
4	A01	0,4864	A01	0,5176	A01	0,5489
5	A08	0,5689	A08	0,6049	A08	0,6409
6	A05	0,6451	A05	0,6758	A05	0,7066
7	A07	0,6763	A07	0,7058	A07	0,7352
8	A06	0,9729	A06	0,9699	A06	0,9669

Dari hasil pemeringkatan dengan nilai V yang berbeda didapatkan alternatif A02 stabil berada di peringkat pertama, sehingga dapat disimpulkan bahwa kondisi *acceptable stability in decision making* terpenuhi.

Berdasarkan hasil pengujian kedua kondisi di atas dapat diketahui bahwa kedua kondisi terpenuhi, sehingga alternatif A02 atau Mluweh dapat diusulkan menjadi solusi kompromi dan merupakan peringkat terbaik dari perangkingan embung dengan metode VIKOR.

4.4 Pengujian *System Usability Scale* (SUS)

System Usability Scale (SUS) ialah sebuah metode pengujian dengan menggunakan kuesioner untuk mengukur *usability* sebuah sistem aplikasi. *System Usability Scale* (SUS) dikembangkan oleh John Brooke pada tahun 1986 dan digunakan untuk memberikan nilai terhadap fungsional dari sistem aplikasi dari pandangan responden [23]. *System Usability Scale* (SUS) terdiri dari 10 pertanyaan yang digunakan untuk menilai kegunaan dari produk atau sistem aplikasi oleh responden dengan memberikan skala nilai 1 (sangat tidak setuju) sampai 5 (sangat setuju) untuk setiap pertanyaan. Pertanyaan yang digunakan pada *System Usability Scale* (SUS) ditunjukkan pada tabel 4.39.

Tabel 4.39 Pertanyaan *System Usability Scale* (SUS)

No	Pertanyaan	STS	TS	RG	S	SS
1	Saya berpikir akan menggunakan sistem ini lagi					
2	Saya merasa sistem ini rumit untuk digunakan					

3	Saya merasa sistem ini mudah digunakan					
4	Saya membutuhkan bantuan dari orang lain atau teknisi dalam menggunakan sistem ini					
5	Saya merasa fitur-fitur sistem ini berjalan dengan semestinya					
6	Saya merasa ada banyak hal yang tidak konsisten (tidak serasi pada sistem ini)					
7	Saya merasa orang lain akan memahami cara menggunakan sistem ini dengan cepat					
8	Saya merasa sistem ini membingungkan					
9	Saya merasa tidak ada hambatan dalam menggunakan sistem ini					
10	Saya perlu membiasakan diri terlebih dahulu sebelum menggunakan sistem ini					

Setiap pertanyaan memiliki nilai kontribusi yang berkisar antara 0 hingga 4. Untuk pertanyaan bernomor 1, 3, 5, 7, dan 9 memiliki nilai kontribusi posisi skala dikurangi 1. Untuk pertanyaan bernomor 2, 4, 6, 8, dan 10 nilai kontribusinya adalah 5 dikurangi posisi skala. Hasil dari nilai *System Usability Scale* (SUS) merupakan jumlah seluruh nilai kontribusi dikali 2,5. Nilai *System Usability Scale* (SUS) berkisar antara 0 sampai 100 [23]. Berikut rumus perhitungan nilai *System Usability Scale* (SUS):

$$\begin{aligned} \text{Nilai atau Skor SUS} = & (((Q1 - 1) + (5 - Q2) + (Q3 - 1) + (5 - Q4) \\ & + (Q5 - 1) + (5 - Q6) + (Q7 - 1) + (5 - Q8) \\ & + (Q9 - 1) + (5 - Q10)) * 2.5) \end{aligned}$$

Keterangan:

Q1 – Q10 : Skor untuk pertanyaan 1 - 10

Pengujian *System Usability Scale* (SUS) dilakukan dengan melakukan pengambilan data melalui kuesioner kepada responden melalui *google form*. Responden melakukan pengujian ke sistem dengan menjalankan skenario tugas yang selanjutnya melakukan penilaian dengan menjawab pertanyaan yang digunakan dalam *System Usability Scale* (SUS). Target responden dalam pengujian ini adalah sejumlah 20 orang dengan rentang umur 20 – 30 tahun.

Dari pengujian yang telah dilakukan melalui penyebaran kuesioner didapatkan hasil seperti yang ditunjukkan dalam tabel 4.40 sebagai berikut:

Tabel 4.40 Hasil *System Usability Scale* (SUS) - SPK VIKOR

	R1	R2	R3	R4	R5	R6	R7	R8	R9	R10
<i>SUS RAW SCORE</i>	38	35	36	24	31	32	36	36	34	30
<i>SUS FINAL SCORE</i>	95	87.5	90	60	77,5	80	90	90	85	75

R11	R12	R13	R14	R15	R16	R17	R18	R19	R20	AVERAGE
34	39	40	38	29	38	29	40	33	20	33,6
85	97,5	100	95	72,5	95	72,5	100	82,5	50	84

Keterangan:

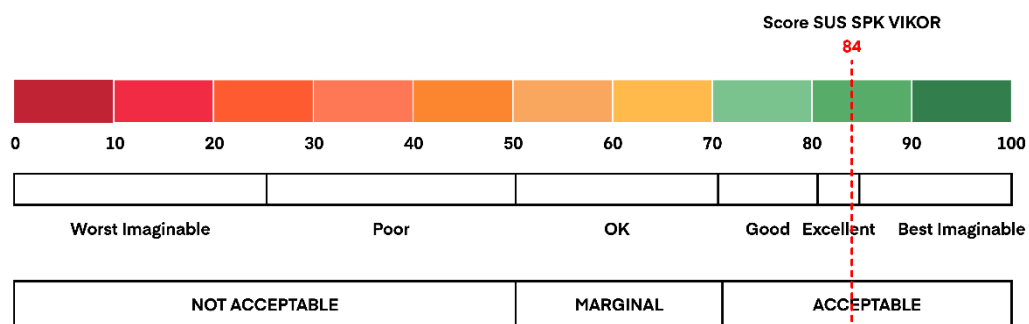
R1 - R20 : Responden ke-1 sampai responden ke-20

SUS raw score : Nilai responden terhadap 10 pertanyaan

SUS final score : $SUS\ raw\ score * 2,5$

Average : Nilai rata-rata dari seluruh *SUS Final Score*

Skor *System Usability Scale* (SUS) digunakan untuk menunjukkan tingkat penerimaan pengguna terhadap sistem. Supaya sistem yang diujikan bisa masuk ke kategori *acceptable* maka skor *System Usability Scale* yang didapat harus bernilai lebih dari 70. Berdasarkan tabel 4.40, sistem pendukung keputusan VIKOR mendapatkan rata-rata skor 84. Selanjutnya ditentukan kategori dan *grade* dari SPK VIKOR dengan menggunakan parameter yang telah ditentukan seperti yang ditunjukkan pada gambar 4.35 [24].



Gambar 4.35 Nilai SUS SPK VIKOR

Berdasarkan nilai yang diperoleh oleh SPK VIKOR dapat disimpulkan bahwa SPK VIKOR masuk ke dalam kategori *acceptable* dengan *grade excellent* dengan rata-rata nilai skor yang didapat senilai 84.

4.5 Pembahasan

Sistem pendukung keputusan ini menerapkan metode VIKOR atau *Višekriterijumsko Kompromisno Rangiranje* untuk mengolah data alternatif dan data kriteria untuk menentukan prioritas pembangunan embung di Kabupaten Semarang dan dibangun menggunakan *framework* Codeigniter dengan visualisasi peta menggunakan *library* dari Leaflet.

Alur sistem dimulai dengan menampilkan halaman awal/*landing page*, di halaman tersebut *guest* dapat melihat penjelasan tentang metode yang digunakan, data alternatif dan kriteria, hasil perangkingan embung, dan visualisasi peta lokasi alternatif. Untuk masuk ke dalam sistem, *user* perlu melakukan *login* terlebih dahulu menggunakan email dan *password* yang sudah terdaftar yang nantinya melalui data tersebut akan dilakukan autentikasi tingkatan pengguna (*role*) oleh sistem. Jika proses autentikasi berhasil maka selanjutnya akan diarahkan ke halaman *dashboard* sesuai dengan tingkatan pengguna (*role*) yang melakukan *login*.

Halaman *dashboard* operator menampilkan data jumlah alternatif, jumlah kriteria, hasil perangkingan, dan visualisasi peta lokasi alternatif sedangkan halaman *dashboard* administrator menampilkan menampilkan data seperti yang ada di halaman *dashboard* operator dengan tambahan data jumlah admin dan jumlah operator. Perbedaan utama operator dan administrator adalah *user* dengan *role* operator dapat melakukan perubahan pada data alternatif, kriteria dan parameter, dan nilai awal sedangkan administrator hanya dapat melihat data-data tersebut, akan tetapi administrator dapat memperbarui seluruh akun yang terdaftar di dalam sistem. Operator mengisi seluruh data alternatif dan kriteria yang dibutuhkan sistem untuk melakukan perhitungan data dengan menggunakan metode VIKOR.

Dari hasil perhitungan didapatkan alternatif Mluweh sebagai peringkat terbaik dengan nilai indeks VIKOR = 0 (saat nilai $V = 0.5$). Untuk memeriksa konsistensi alternatif Mluweh dilakukan dua pengujian yaitu; pengujian kondisi *acceptable advantage* dan pengujian kondisi *acceptable stability in decision making*. Setelah dilakukan pengujian, didapatkan bahwa kedua kondisi terpenuhi,

yang berarti bahwa alternatif Mluweh stabil berada di peringkat pertama sehingga alternatif Mluweh dapat diusulkan menjadi solusi kompromi dan merupakan peringkat terbaik dari perankingan embung menggunakan metode VIKOR.

Untuk memeriksa akurasi perhitungan sistem, dilakukan perhitungan manual menggunakan data yang sama dengan yang ada di dalam sistem. Sistem juga diuji coba menggunakan data penelitian terdahulu dan didapatkan hasil yang sama antara perhitungan manual dengan perhitungan sistem, sehingga dapat disimpulkan bahwa perhitungan sistem sudah benar dan sesuai dengan perhitungan metode VIKOR.

4.6 Demonstrasi dan Perbaikan (*Demonstrate and Refine*)

Setelah tahap pembuatan (*build*) sistem pendukung keputusan selesai, tahap selanjutnya adalah proses demonstrasi (*demonstrate*) dan perbaikan (*refine*). Berdasarkan metode penelitian RAD yang digunakan, setelah tahap pembuatan (*build*), sistem akan langsung didemonstrasikan dan ketika terdapat perubahan atau perbaikan pada sistem maka akan segera diperbaiki. Pada bagian demonstrasi dan perbaikan, nantinya akan berurusan langsung dengan pengguna atau *user*.

BAB V

PENUTUP

5.1 Kesimpulan

Berdasarkan hasil pengujian dan analisis dari penelitian ini, didapatkan kesimpulan sebagai berikut:

1. Sistem pendukung keputusan dengan menggunakan VIKOR dapat digunakan untuk melakukan pemeringkatan banyak alternatif dengan multi-kriteria secara efektif dan efisien.
2. Alternatif Mluweh menjadi peringkat terbaik dalam perangkaian menggunakan metode VIKOR dan tetap stabil menjadi peringkat terbaik setelah dilakukan pengujian kondisi *acceptable advantage* dan pengujian kondisi *acceptable stability in decision making*.
3. Dalam metode VIKOR tidak ada perhitungan khusus untuk menghitung nilai bobot masing-masing kriteria. Pemberian bobot hanya diberikan begitu saja oleh pengambil keputusan sehingga diperlukan metode lain untuk memeriksa konsistensi pembobotan seperti AHP dan sebagainya.

5.2 Saran

Dari hasil analisis dan pengujian program sistem pendukung keputusan dengan metode VIKOR (*Višekriterijumsko Kompromisno Rangiranje*) untuk penentuan prioritas pembangunan embung di Kabupaten Semarang, dapat diberikan saran sebagai berikut:

1. Pengembangan sistem pendukung dapat dilanjutkan dengan membuat sistem lebih *responsive* secara antarmuka, serta dapat dikembangkan secara lebih lanjut dalam bentuk aplikasi *mobile* berbasis Android atau iOS.
2. Hasil perhitungan metode *Višekriterijumsko Kompromisno Rangiranje* (VIKOR) untuk menentukan lokasi pembangunan embung dapat dibandingkan dengan hasil perhitungan yang didapat dengan metode lainnya.