

BAB IV

ANALISIS DAN PERANCANGAN

Bab ini menjelaskan hasil analisis sistem yang telah dilakukan serta perancangan sistem yang dikembangkan. Penjabaran disusun dalam beberapa subbab untuk menguraikan komponen-komponen utama secara sistematis.

4.1 Analisis Sistem Terdahulu

Penelitian ini telah melakukan analisis terhadap beberapa sistem rekomendasi yang relevan sebagai acuan dalam pengembangan sistem rekomendasi kafe berbasis web di wilayah Bintaro Jaya. Analisis ini dilakukan untuk memahami pendekatan, metode, dan cakupan fitur dari sistem-sistem sebelumnya yang menggunakan metode AHP, SAW, ataupun kombinasi keduanya.

Tiga penelitian dipilih sebagai fokus analisis utama karena mewakili variasi metode yang digunakan: SAW saja, AHP saja, dan gabungan AHP-SAW. Analisis difokuskan pada aspek metode, objek studi, jumlah kriteria, penerapan aplikasi, dan kelemahan maupun kelebihan. Hasil perbandingan dapat dilihat pada tabel berikut:

Tabel 4. 1 Analisis sistem terdahulu

Aspek	Noerul Hanin, Ahmad Cahyono Adi (2023)	Hanifah Zahro (2021)	Putri Septiana (2023)
Objek Studi	Kafe untuk mahasiswa di Pontianak	Kafe di Tangerang Selatan	Rumah kost mahasiswa di Metro
Metode yang Digunakan	SAW	AHP	AHP + SAW
Jumlah Kriteria	5 (Wifi, harga, bangunan, luas, jarak ke kampus)	9 (fasilitas, lokasi, variasi menu, rasa, dll.)	6 (biaya, fasilitas, jarak, keamanan, lingkungan, aturan)
Aplikasi Berbasis Web	Tidak disebutkan	Ya	Tidak disebutkan
Perankingan Alternatif	Ya, dengan SAW	Tidak ada perankingan (hanya bobot AHP)	Ya, menggunakan SAW
Keunikan/Kelebihan	Survei responden mahasiswa; hasil predikat cafe ideal	Skor UAT tinggi (86.9%) menunjukkan penerimaan pengguna	Kombinasi AHP & SAW; logika multikriteria berjalan utuh

Kelemahan	Tidak ada pembobotan eksplisit (hanya SAW)	Tidak menyelesaikan proses rekomendasi	Objek bukan kafe; tidak berbasis web
-----------	--	--	--------------------------------------

Berdasarkan hasil perbandingan tersebut, dapat disimpulkan bahwa sistem rekomendasi terdahulu belum sepenuhnya menggabungkan keunggulan dari metode AHP dan SAW secara terpadu dalam konteks pemilihan kafe berbasis web. Penelitian oleh Noerul Hanin dan Ahmad Cahyono Adi (2023) hanya menggunakan metode SAW tanpa proses pembobotan yang sistematis, sedangkan penelitian oleh Hanifah Zahro (2021) menggunakan AHP tetapi tidak melanjutkan ke tahap perankingan alternatif. Sementara itu, penelitian oleh Putri Septiana (2023) telah menggabungkan kedua metode tersebut, namun fokusnya pada objek rumah kost, bukan kafe, dan belum diimplementasikan dalam bentuk aplikasi web.

Oleh karena itu, penelitian ini menghadirkan kebaruan dalam bentuk penerapan gabungan AHP untuk pembobotan kriteria dan SAW untuk perankingan alternatif dalam sistem rekomendasi kafe berbasis web. Sistem yang dikembangkan juga memperluas jumlah kriteria menjadi 13 dan secara spesifik berfokus pada wilayah Bintaro Jaya sebagai lokasi studi, yang belum banyak dijadikan objek penelitian sebelumnya.

4.2 Spesifikasi Kebutuhan Sistem

Sistem yang dikembangkan dalam penelitian ini merupakan sistem pendukung keputusan berbasis web untuk memberikan rekomendasi kafe di wilayah Bintaro Jaya. Sistem ini menggunakan kombinasi metode Analytic Hierarchy Process (AHP) dan Simple Additive Weighting (SAW) untuk mengolah preferensi pengguna secara sistematis dan menghasilkan daftar peringkat kafe yang sesuai dengan kebutuhan pengguna.

Spesifikasi sistem ini dikategorikan ke dalam dua bagian utama, yaitu spesifikasi fungsional dan spesifikasi non-fungsional. Keduanya disusun berdasarkan hasil analisis kebutuhan serta tujuan fungsional dari sistem.

4.2.1 Spesifikasi Fungsional

Spesifikasi fungsional menjelaskan fitur-fitur inti yang wajib dimiliki oleh sistem. Fitur-fitur ini dirancang untuk mencakup proses input, pengolahan data, serta penyajian hasil kepada pengguna dan admin.

- 1) Sistem memungkinkan pengguna untuk memilih dan mengatur preferensi berdasarkan 13 kriteria (misalnya harga, Wi-Fi, variasi menu, waktu operasional, dsb).
- 2) Sistem memproses pembobotan preferensi pengguna menggunakan metode AHP.
- 3) Sistem melakukan perankingan alternatif kafe dengan metode SAW berdasarkan bobot dari AHP.
- 4) Sistem menampilkan daftar rekomendasi kafe dalam urutan peringkat secara real-time.
- 5) Pengguna dapat melihat detail setiap kafe, termasuk fasilitas, deskripsi, dan gambar.
- 6) Pengguna dapat melakukan pencarian dan filter berdasarkan kategori tertentu.
- 7) Sistem menyimpan histori rekomendasi berdasarkan input pengguna (opsional).
- 8) Admin dapat mengelola data kafe (CRUD: create, read, update, delete).
- 9) Admin dapat menambah atau menyesuaikan kriteria penilaian yang digunakan dalam sistem.

4.2.2 Spesifikasi Non-Fungsional

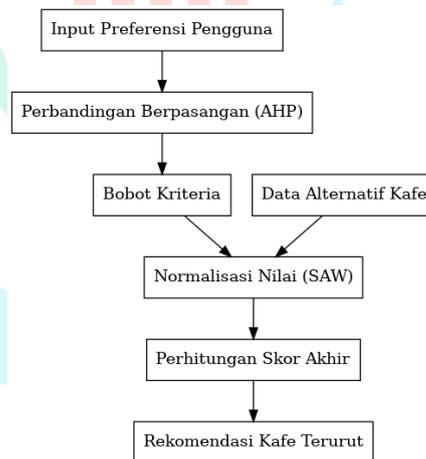
Spesifikasi non-fungsional mendefinisikan batasan teknis dan standar kualitas dari sistem, termasuk performa, keamanan, kompatibilitas, dan skalabilitas.

- 1) Kinerja (*Performance*): Sistem menampilkan hasil rekomendasi maksimal dalam waktu kurang dari 2 detik.

- 2) Kompatibilitas: Sistem dapat diakses melalui peramban web modern (Chrome, Firefox, Edge) di berbagai perangkat, termasuk laptop dan smartphone.
- 3) Keamanan: Informasi pengguna (nama, email, nomor telepon, dan kata sandi) disimpan dengan aman dan tidak dibagikan ke pihak ketiga.
- 4) Skalabilitas: Sistem dirancang agar dapat diperluas ke wilayah lain di luar Bintaro Jaya dengan menambahkan dataset kafe baru.
- 5) Usability: Antarmuka pengguna (UI) dirancang modern, minimalis, dan *mobile-friendly*, dibangun dengan ReactJS.
- 6) Reliabilitas: Sistem tetap dapat menampilkan rekomendasi meskipun koneksi jaringan melambat, dengan *fallback caching* untuk hasil terakhir.

4.2.3 Spesifikasi Proses

Gambar 4.1 berikut menunjukkan alur proses sistem pendukung keputusan dalam memberikan rekomendasi kafe, dimulai dari preferensi pengguna hingga keluaran akhir berupa daftar rekomendasi.



Gambar 4.1 Diagram Alur Proses AHP & SAW

Alur pengambilan keputusan dalam sistem ini terdiri dari tiga tahapan utama, yaitu input, proses, dan output. Pada tahap input, pengguna memberikan preferensi terhadap sejumlah kriteria, seperti harga, fasilitas Wi-Fi, kehalalan makanan, dan lainnya. Preferensi tersebut digunakan dalam metode AHP untuk menentukan bobot prioritas dari masing-masing kriteria berdasarkan matrix perbandingan (*Pairwise Matrix*). Setelah bobot kriteria diperoleh, sistem melanjutkan ke tahap

proses kedua menggunakan metode SAW, di mana data atribut dari alternatif kafe dinormalisasi dan dikalikan dengan bobot tersebut untuk menghasilkan skor akhir. Tahap akhir dari alur ini adalah output berupa daftar rekomendasi kafe yang telah diurutkan berdasarkan skor tertinggi.

Pemilihan kombinasi metode AHP dan SAW dalam sistem ini didasarkan pada pertimbangan karakteristik masing-masing metode yang saling melengkapi. AHP berperan penting dalam menyediakan struktur hierarki kriteria yang sistematis dan logis. Melalui teknik perbandingan berpasangan, AHP memungkinkan pengguna atau pengelola sistem untuk menilai tingkat kepentingan relatif dari setiap kriteria secara konsisten, yang kemudian dikalkulasi menjadi bobot numerik yang merepresentasikan prioritas masing-masing kriteria.

Setelah bobot kriteria diperoleh, metode SAW digunakan untuk melanjutkan proses pengambilan keputusan ke tahap perankingan alternatif. SAW mengubah data atribut dari setiap alternatif menjadi nilai terukur, kemudian menggabungkannya dengan bobot kriteria dari AHP untuk menghasilkan skor total yang objektif. Karena proses ini bersifat linear dan mudah diterapkan, SAW sangat efektif untuk menghasilkan peringkat alternatif berdasarkan berbagai kriteria dengan cara yang transparan dan dapat dipertanggungjawabkan.

Kekuatan dari penggabungan kedua metode ini terletak pada sinergi antara struktur logis AHP dalam menetapkan bobot dan efisiensi SAW dalam menghitung skor akhir. Dengan pendekatan ini, sistem dapat menyaring dan mengurutkan pilihan kafe secara adil berdasarkan preferensi pengguna, tanpa harus bergantung pada ulasan subjektif yang sering kali tidak lengkap atau bias.

Dibandingkan metode lain seperti TOPSIS atau Fuzzy AHP, kombinasi AHP dan SAW dipilih karena menawarkan pendekatan yang saling melengkapi sekaligus efisien. AHP efektif dalam menangkap preferensi kualitatif pengguna melalui pembobotan kriteria secara terstruktur, sedangkan SAW unggul dalam melakukan perankingan alternatif secara kuantitatif dan transparan. TOPSIS memerlukan pembentukan solusi ideal positif dan negatif yang kurang intuitif bagi pengguna umum, sementara Fuzzy AHP menuntut pemrosesan data linguistik dan logika fuzzy yang kompleks, yang kurang praktis dalam konteks aplikasi publik

seperti pemilihan kafe. Oleh karena itu, AHP-SAW dinilai lebih tepat untuk kebutuhan sistem rekomendasi praktis dengan banyak kriteria dan data atribut eksplisit seperti ini.

4.3 Perancangan Sistem

4.3.1 Arsitektur Sistem dan Alasan Pemilihan Teknologi

Dalam merancang sistem pendukung keputusan rekomendasi kafe ini, arsitektur sistem dipisahkan menjadi dua bagian utama: backend dan frontend. Untuk bagian backend, digunakan framework Laravel, sedangkan bagian frontend menggunakan ReactJS. Pemilihan kedua teknologi ini tidak hanya didasarkan pada popularitasnya, tetapi pada keunggulan teknis dan konseptual yang ditawarkan dalam pengembangan sistem berbasis web modern.

- Laravel mendukung pengembangan aplikasi berbasis RESTful API yang modular, aman, dan efisien. Dengan dukungan Eloquent ORM, sistem dapat mengelola data dengan struktur relasional yang kompleks secara konsisten dan rapi. Fitur-fitur seperti routing dinamis, middleware, dan CSRF protection mendukung pengembangan sistem yang skalabel dan aman.

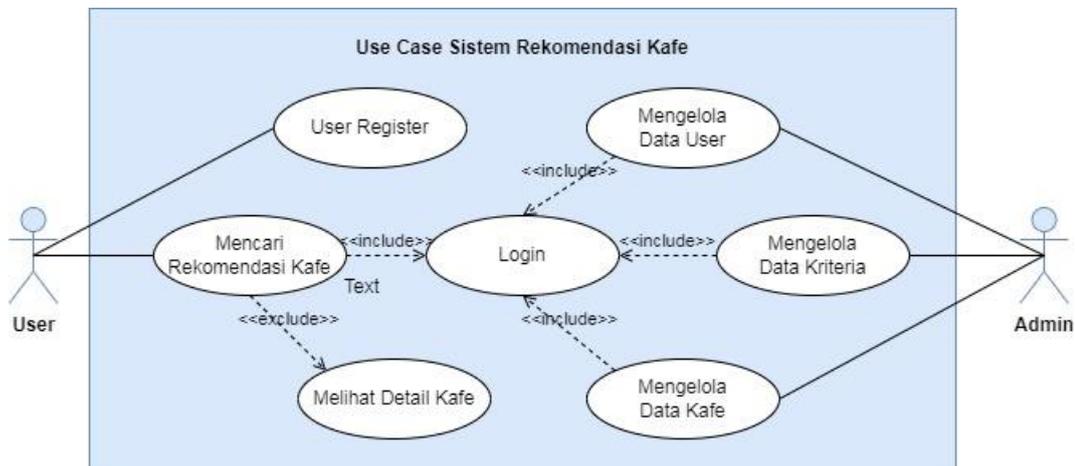
Sementara itu, ReactJS memungkinkan pembuatan Single Page Application (SPA) yang ringan dan cepat, memberikan pengalaman pengguna yang responsif tanpa perlu memuat ulang halaman secara penuh. Kemampuan React untuk membangun komponen antarmuka yang dapat digunakan ulang (reusable components) juga mempermudah proses pengembangan dan pemeliharaan.

Kombinasi Laravel–React menghasilkan sistem dengan arsitektur terpisah antara client dan server, sehingga memudahkan integrasi dan pengembangan lebih lanjut, termasuk potensi pengembangan versi mobile atau integrasi AI di masa depan.

4.3.2 Use Case

Use case merupakan metode yang digunakan untuk memetakan dan menggambarkan interaksi antara pengguna dengan sistem, termasuk bagaimana

sistem merespons tindakan dari setiap aktor yang terlibat. Pendekatan ini membantu menjelaskan peran dan alur aktivitas yang dilakukan oleh pengguna dalam mencapai tujuan tertentu melalui sistem. Diagram *Use Case* untuk sistem rekomendasi kafe disajikan pada Gambar 4.1.



Gambar 4. 2 Use Case Diagram Sistem Rekomendasi Kafe

Dari gambar 4.2 di atas, dapat dilihat bahwa sistem memiliki dua aktor utama, yaitu User dan Admin. Aktor User dapat melakukan registrasi, login, mencari rekomendasi kafe berdasarkan kriteria tertentu, serta melihat detail kafe. Use case “Melihat Detail Kafe” memiliki relasi “*exclude*” terhadap use case utama “Mencari Rekomendasi Kafe”. Relasi ini menunjukkan bahwa fitur tersebut bersifat opsional, dan hanya dijalankan apabila pengguna ingin melihat informasi lebih lanjut tentang kafe tertentu. Artinya, pengguna dapat langsung memilih hasil rekomendasi tanpa melihat detail, atau sebaliknya, tergantung pada kebutuhan atau preferensi penggunaan.

Sementara itu, aktor Admin memiliki hak akses untuk login ke sistem dan mengelola data pengguna, kriteria penilaian, serta informasi kafe, termasuk fungsi menambah, mengubah, dan menghapus data. Dengan pendekatan ini, Use Case Diagram memberikan gambaran awal yang jelas mengenai fungsi utama dalam sistem serta interaksi antara pengguna dan aplikasi.

4.3.3 *Skenario Use Case*

Skenario *Use Case* adalah serangkaian langkah-langkah atau situasi yang menjelaskan bagaimana pengguna (aktor) akan berinteraksi dengan sebuah sistem atau aplikasi untuk mencapai tujuan tertentu. Skenario ini menunjukkan serangkaian aksi yang diambil oleh pengguna dan sistem dalam konteks penggunaan aplikasi atau sistem tersebut. Ini membantu dalam memahami secara rinci bagaimana aplikasi atau sistem akan digunakan dalam berbagai situasi atau kondisi yang berbeda. Skenario *Use Case* sering digunakan dalam proses pengembangan perangkat lunak untuk merancang fungsionalitas sistem dan memvalidasi kebutuhan pengguna.

4.3.3.1 Skenario Use Case User Register

Tabel 4. 2 Skenario *Use Case User Register*

Aktor	User
Use Case	User melakukan login ke aplikasi
Kondisi Awal	User belum memiliki akun
Tahapan	<ol style="list-style-type: none"> 1. User mengakses aplikasi 2. User memilih menu login 3. User memilih registrasi akun 4. User mengisikan kredensial berupa nama lengkap, alamat e-mail, nomor telfon, dan password 5. User berhasil mendaftar dan diarahkan ke beranda
Kondisi Akhir	User berhasil mendaftarkan akun nya

4.3.3.2 Skenario Use Case User Login

Tabel 4. 3 Skenario *Use Case User Login*

Aktor	User
Use Case	User melakukan login ke aplikasi
Kondisi Awal	User sudah melakukan pendaftaran dan belum login ke akun nya
Tahapan	<ol style="list-style-type: none"> 1. User mengakses aplikasi 2. User memilih menu login 3. User mengisikan kredensial berupa e-mail dan password 4. User berhasil login dan diarahkan ke beranda
Kondisi Akhir	User berhasil login ke akun nya

4.3.3.3 Skenario Use Case Mencari Rekomendasi Kafe

Tabel 4. 4 Skenario *Use Case Mencari Rekomendasi Kafe*

Aktor	User
Use Case	User mengakses halaman rekomendasi
Kondisi Awal	User harus sudah login ke akun nya
Tahapan	<ol style="list-style-type: none">1. User memilih menu rekomendasi2. User menginput kriteria yang diinginkan
Kondisi Akhir	User mendapat hasil rekomendasi berdasarkan kriteria

4.3.3.4 Skenario Use Case Melihat Detail Kafe

Tabel 4. 5 Skenario *Use Case Melihat Detail Kafe*

Aktor	User
Use Case	User melihat detail kafe
Kondisi Awal	User berada di beranda aplikasi
Tahapan	<ol style="list-style-type: none">1. User memencari kafe yang diinginkan2. User memilih tombol selengkapnya
Kondisi Akhir	User dapat melihat halaman detail kafe

4.3.3.5 Skenario Use Case Admin Login

Tabel 4. 6 Skenario *Use Case Admin Login*

Aktor	Admin
Use Case	Admin melakukan login
Kondisi Awal	Admin belum melakukan login
Tahapan	<ol style="list-style-type: none">1. Admin mengakses aplikasi2. Admin memilih menu login3. Admin mengisikan kredensial berupa e-mail dan password4. Admin berhasil login dan diarahkan ke halaman admin
Kondisi Akhir	Admin berhasil login ke halaman admin

4.3.3.6 Skenario Use Case Admin Mengelola Data User

Tabel 4. 7 Skenario *Use Case Mengelola Data User*

Aktor	Admin
Use Case	Admin mengakses data user
Kondisi Awal	Admin harus sudah melakukan login ke halaman admin
Tahapan	<ol style="list-style-type: none">1. Admin mengakses halaman user2. Admin memilih tombol Tambah Data dan menginput data user3. Admin memilih data user dan klik tombol edit untuk mengubah data user4. Admin memilih data user dan klik tombol hapus untuk menghapus data user
Kondisi Akhir	Admin berhasil mengelola data user

4.3.3.7 Skenario Use Case Admin Mengelola Data Kriteria

Tabel 4. 8 Skenario *Use Case Mengelola Data Kriteria*

Aktor	Admin
Use Case	Admin mengakses data kriteria
Kondisi Awal	Admin harus sudah melakukan login ke halaman admin
Tahapan	<ol style="list-style-type: none">1. Admin mengakses halaman kriteria2. Admin memilih tombol Tambah Data dan menginput data kriteria3. Admin memilih data kriteria dan klik tombol edit untuk mengubah data kriteria4. Admin memilih data kriteria dan klik tombol hapus untuk menghapus data kriteria
Kondisi Akhir	Admin berhasil mengelola data kriteria

4.3.3.8 Skenario Use Case Admin Mengelola Data Kafe

Tabel 4. 9 Skenario *Use Case Mengelola Data Kafe*

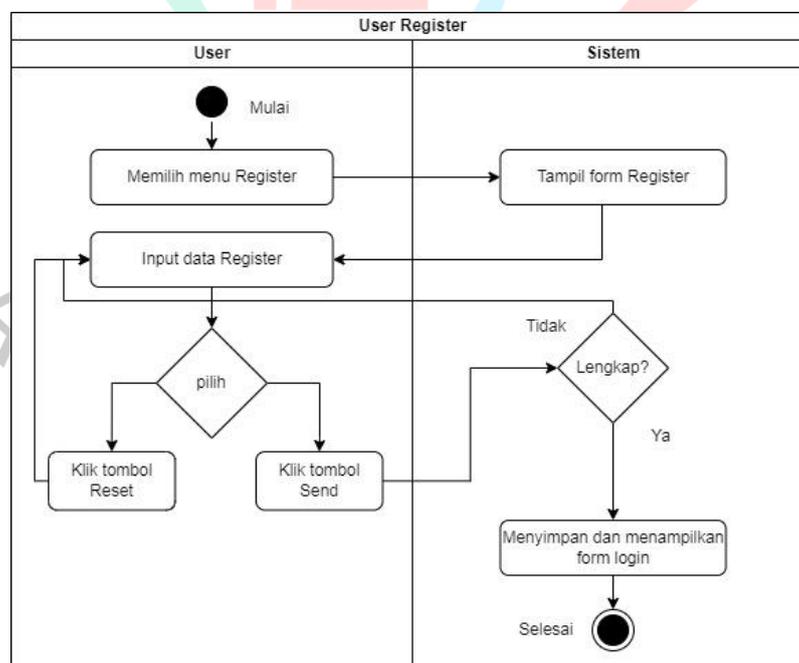
Aktor	Admin
Use Case	Admin mengakses data kafe
Kondisi Awal	Admin harus sudah melakukan login ke halaman admin

Tahapan	<ol style="list-style-type: none"> 1. Admin mengakses halaman kafe 2. Admin memilih tombol Tambah Data dan menginput data kafe 3. Admin memilih data kafe dan klik tombol edit untuk mengubah data kafe 4. Admin memilih data kafe dan klik tombol hapus untuk menghapus data kafe
Kondisi Akhir	Admin berhasil mengelola data kafe

4.3.4 Activity Diagram

Setelah perancangan use case dan skenario interaksi disusun, langkah selanjutnya adalah pembuatan activity diagram untuk merepresentasikan alur aktivitas dalam setiap skenario yang telah ditentukan. Diagram ini digunakan untuk menggambarkan urutan proses serta logika kerja sistem pada masing-masing kegiatan. Penjabaran lebih lanjut akan menjelaskan aktivitas yang dilakukan oleh tiap jenis pengguna dalam sistem.

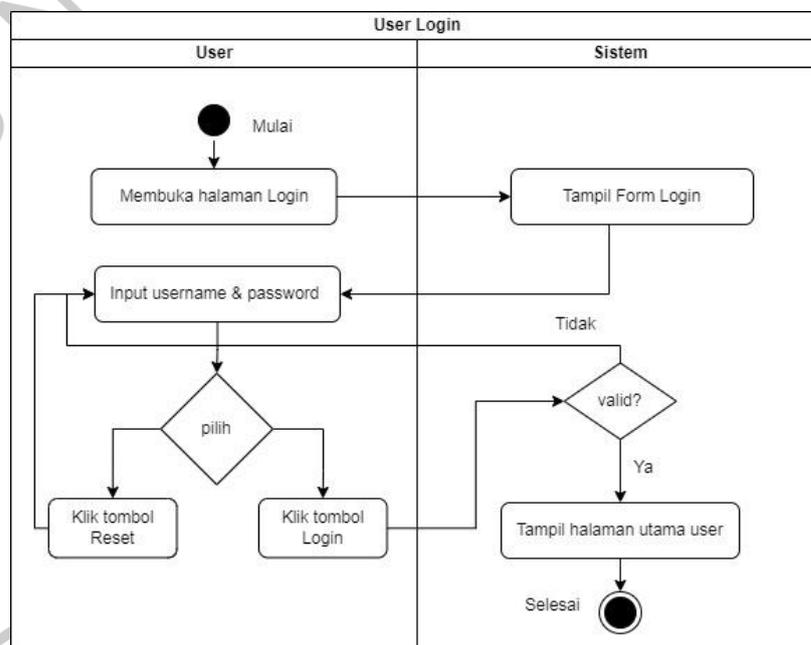
4.3.4.1 Activity Diagram User Register



Gambar 4. 3 Activity Diagram User Register

Gambar 4.3 menunjukkan alur aktivitas saat pengguna melakukan registrasi. Proses dimulai ketika pengguna memilih menu Register, lalu sistem menampilkan form registrasi. Pengguna mengisi data yang diminta, kemudian memilih untuk mengirim atau mereset data. Jika pengguna menekan tombol Reset, proses kembali ke tahap pengisian. Jika tombol Send diklik, sistem akan memeriksa kelengkapan data. Jika data belum lengkap, pengguna diarahkan kembali untuk melengkapi form. Jika data lengkap, sistem menyimpan data dan menampilkan form login sebagai langkah akhir proses registrasi.

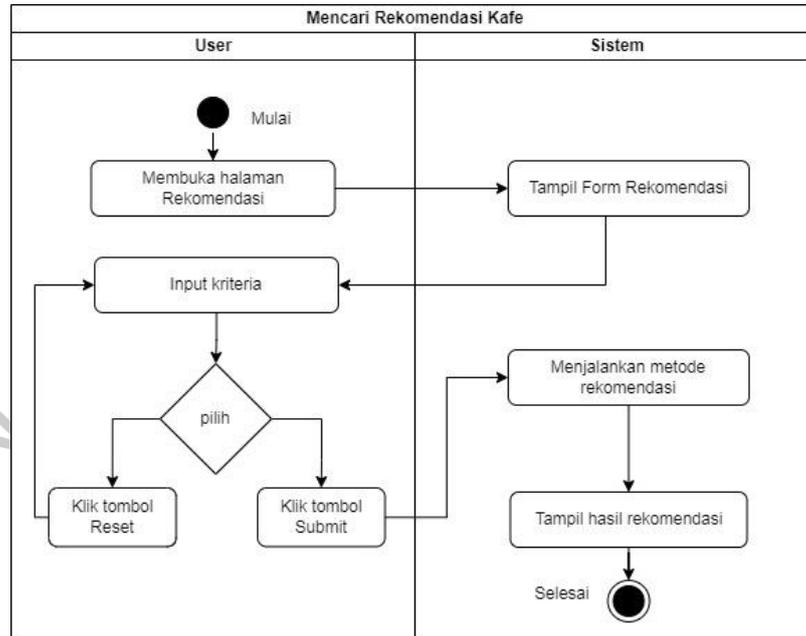
4.3.4.2 Activity Diagram User Login



Gambar 4. 4 Activity diagram User Login

Gambar 4.4 menunjukkan alur aktivitas ketika pengguna melakukan login ke dalam sistem. Proses diawali dengan membuka halaman login, lalu sistem menampilkan form login. Pengguna kemudian mengisi username dan password, serta memilih untuk mengirim data melalui tombol Login atau mengulang isian melalui tombol Reset. Jika data dikirim, sistem akan melakukan validasi terhadap informasi yang dimasukkan. Jika data tidak valid, pengguna diarahkan kembali untuk mengisi ulang. Jika valid, sistem akan menampilkan halaman utama pengguna dan proses login selesai.

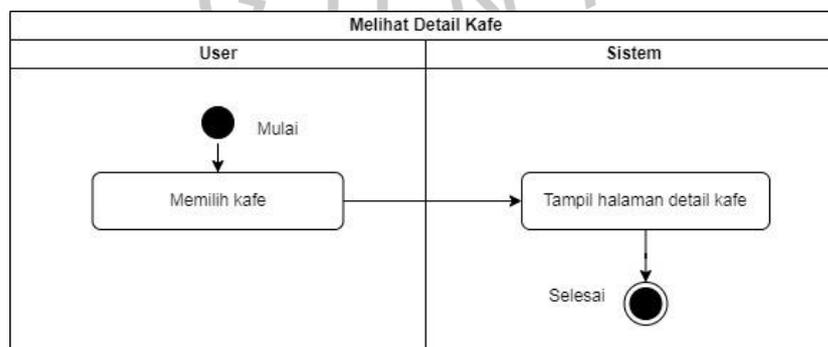
4.3.4.3 Activity Diagram Mencari Rekomendasi Kafe



Gambar 4. 5 Activity Diagram Mencari Rekomendasi Kafe

Gambar 4.5 menunjukkan alur aktivitas saat pengguna mencari rekomendasi kafe. Proses diawali ketika pengguna membuka halaman rekomendasi, dan sistem menampilkan form untuk memasukkan kriteria pencarian. Setelah pengguna mengisi kriteria, terdapat dua opsi: mengklik tombol Reset untuk mengulang pengisian, atau tombol Submit untuk mengirim data. Jika tombol Submit dipilih, sistem akan menjalankan proses perhitungan menggunakan metode rekomendasi, lalu menampilkan hasil rekomendasi kafe kepada pengguna. Proses berakhir setelah hasil ditampilkan.

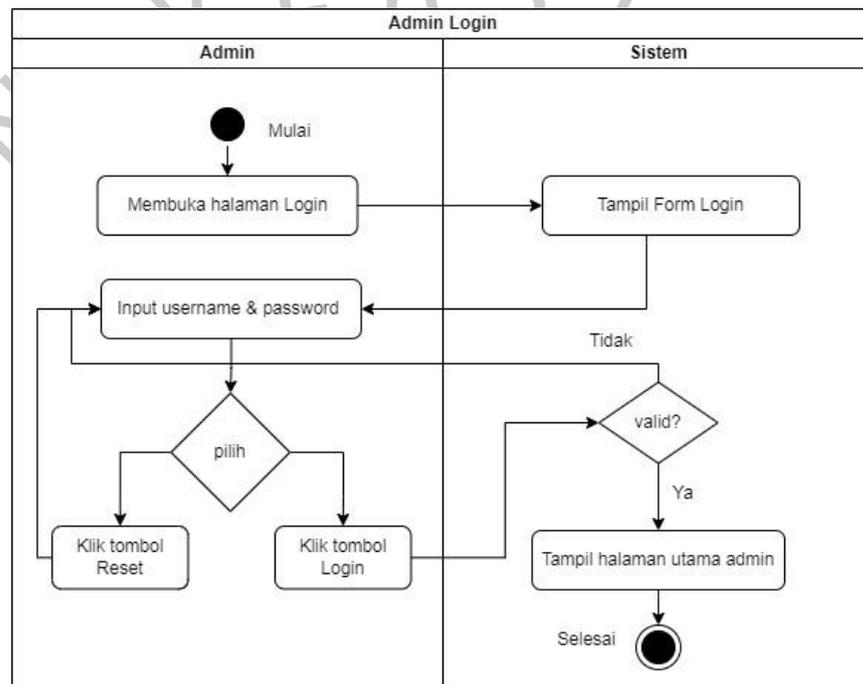
4.3.4.4 Activity Diagram Melihat Detail Kafe



Gambar 4. 6 Activity Diagram Melihat Detail Kafe

Gambar 4.6 menunjukkan aktivitas ketika pengguna ingin melihat detail informasi dari salah satu kafe. Proses dimulai saat pengguna memilih kafe dari daftar yang tersedia. Sistem kemudian menampilkan halaman detail kafe yang berisi informasi tambahan seperti nama, alamat, menu, dan fasilitas kafe.

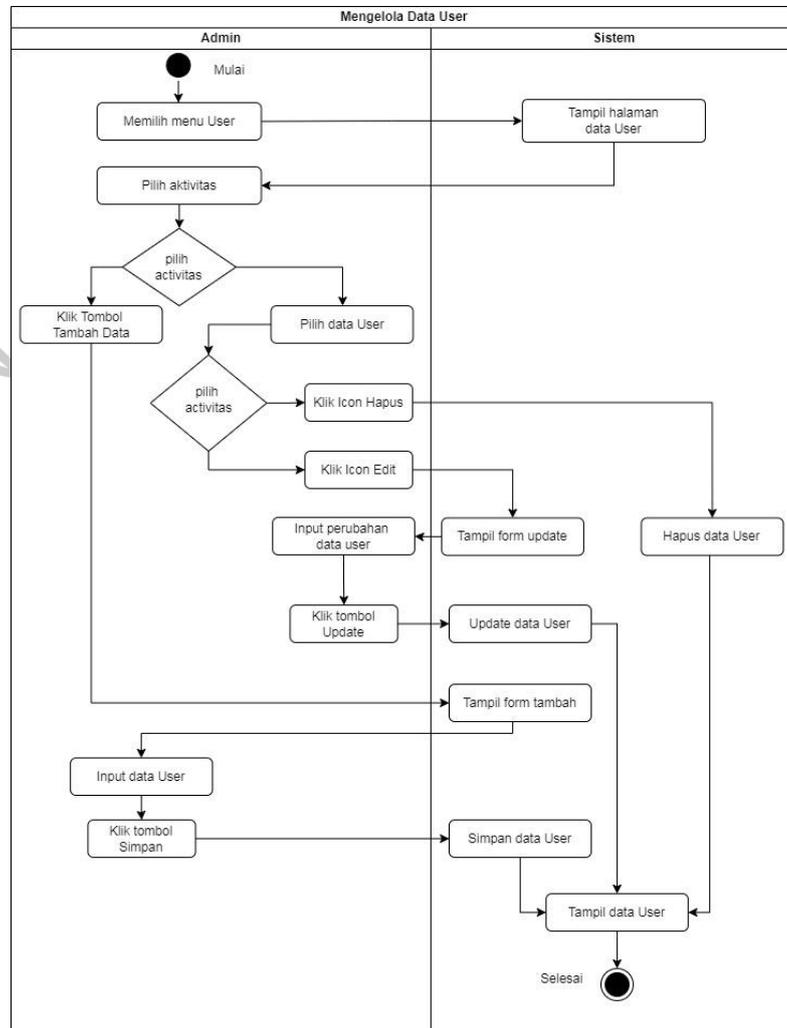
4.3.4.5 Activity Diagram Admin Login



Gambar 4. 7 Activity diagram Admin Login

Gambar 4.7 memperlihatkan alur aktivitas saat admin melakukan proses login ke dalam sistem. Proses diawali dengan membuka halaman login, di mana sistem menampilkan form untuk diisi. Admin memasukkan username dan password, lalu memilih untuk mengirim data atau mereset input. Jika tombol Login diklik, sistem akan memvalidasi data yang dimasukkan. Jika tidak valid, admin akan diarahkan kembali untuk mengisi ulang. Jika data valid, sistem menampilkan halaman utama admin dan proses login selesai.

4.3.4.6 Activity Diagram Mengelola Data User

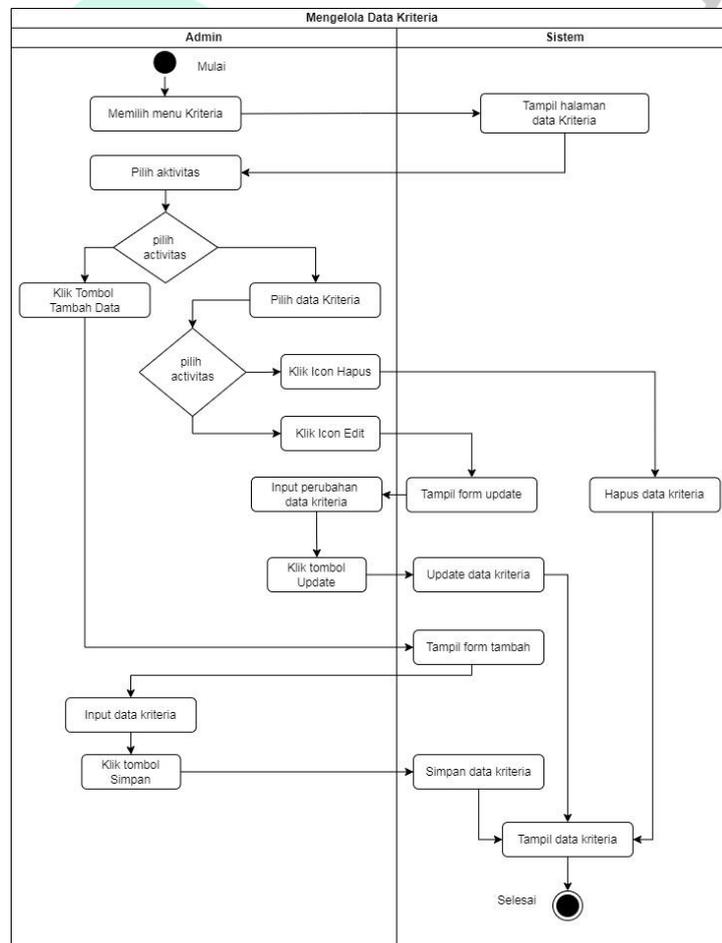


Gambar 4. 8 Activity Diagram Mengelola Data User

Gambar 4.8 menunjukkan alur aktivitas saat admin melakukan pengelolaan data pengguna. Proses dimulai ketika admin memilih menu User, lalu sistem menampilkan halaman data user. Admin kemudian memilih jenis aktivitas yang ingin dilakukan, yaitu menambah, mengubah, atau menghapus data pengguna. Untuk menambah data, admin mengisi form melalui opsi “Tambah Data” dan menekan tombol Simpan. Sistem akan menyimpan data baru ke dalam basis data dan kembali menampilkan daftar pengguna yang telah diperbarui.

Untuk mengubah atau menghapus data, admin terlebih dahulu memilih data pengguna yang tersedia. Jika memilih untuk menghapus, sistem langsung memproses dan menghapus data tersebut dari basis data. Jika memilih untuk mengedit, sistem akan menampilkan form update, kemudian admin dapat menginput perubahan dan menekan tombol Update. Sistem akan memperbarui data dan kembali menampilkan data user. Diagram ini menggambarkan tiga jalur aktivitas berbeda dalam satu skenario pengelolaan data, yang saling terintegrasi dalam satu tampilan sistem.

4.3.4.7 Activity Diagram Mengelola Data Kriteria



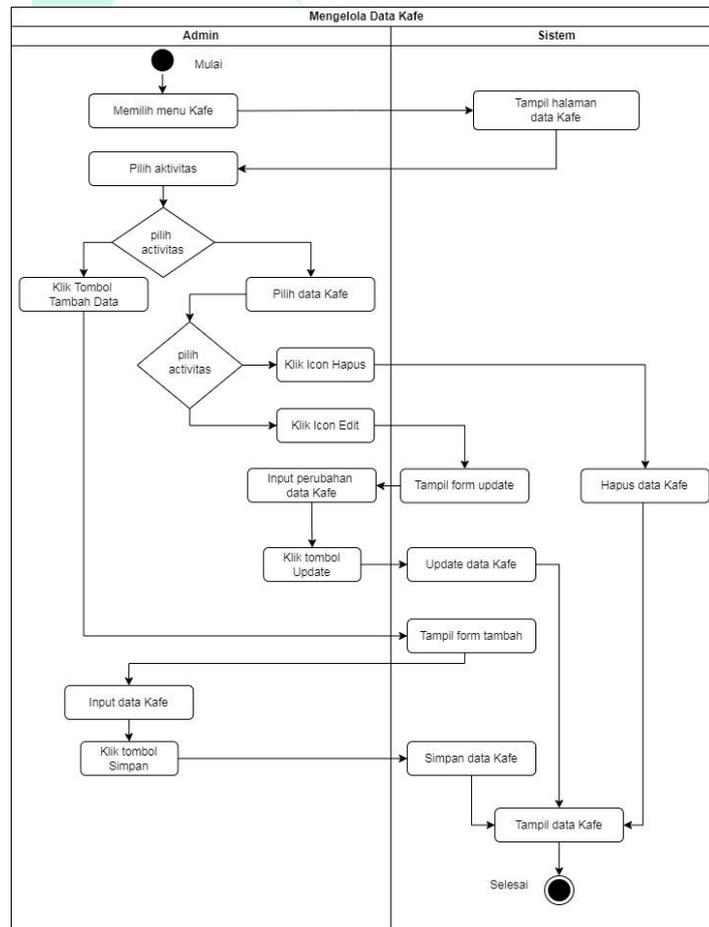
Gambar 4. 9 Activity Diagram Mengelola Data Kriteria

Gambar 4.9 menunjukkan alur aktivitas admin dalam mengelola data kriteria. Proses dimulai ketika admin memilih menu Kriteria, lalu sistem

menampilkan halaman data kriteria. Selanjutnya, admin memilih aktivitas yang ingin dilakukan, yaitu menambah, mengubah, atau menghapus data kriteria.

Jika admin memilih menambah data, sistem menampilkan form input untuk diisi. Setelah admin mengisi data dan menekan tombol Simpan, sistem akan menyimpan data kriteria ke dalam basis data dan memperbarui tampilan data. Jika admin memilih untuk mengubah data, maka setelah memilih kriteria tertentu dan menekan ikon Edit, sistem akan menampilkan form update. Admin melakukan perubahan dan mengklik tombol Update, kemudian sistem akan memperbarui data kriteria tersebut. Sementara itu, jika admin memilih menghapus data, sistem akan langsung menghapus data yang dipilih setelah admin menekan ikon Hapus.

4.3.4.8 Activity Diagram Mengelola Data Kafe



Gambar 4. 10 Activity Diagram Mengelola Data Kafe

Gambar 4.10 menunjukkan alur aktivitas saat admin melakukan pengelolaan data kafe. Proses dimulai dengan memilih menu Kafe, kemudian sistem menampilkan halaman data kafe. Admin selanjutnya memilih aktivitas yang ingin dilakukan, yaitu menambah, mengubah, atau menghapus data kafe.

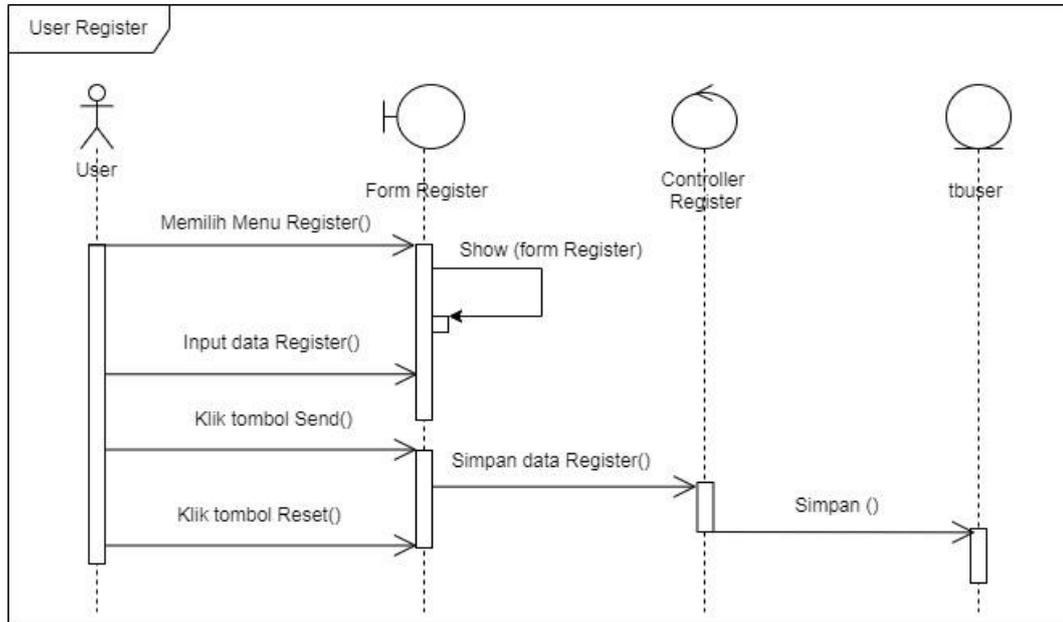
Jika admin memilih menambah data, sistem akan menampilkan form tambah yang dapat diisi, kemudian data dikirim melalui tombol Simpan dan disimpan ke dalam basis data. Jika admin memilih untuk mengubah data, ia akan memilih data kafe yang diinginkan, menekan ikon Edit, lalu menginput perubahan pada form yang ditampilkan. Setelah menekan tombol Update, sistem akan memperbarui data kafe tersebut. Jika admin memilih menghapus data, maka setelah memilih data dan menekan ikon Hapus, sistem akan langsung menghapus data kafe dari basis data.

4.3.5 Sequence Diagram

Sequence diagram merupakan salah satu jenis diagram interaksi dalam *Unified Modeling Language* (UML) yang digunakan untuk memvisualisasikan urutan komunikasi antar objek atau aktor dalam sistem selama jangka waktu tertentu. Diagram ini memperlihatkan aliran pesan yang terjadi dalam suatu skenario atau fungsi spesifik, sehingga memudahkan pemahaman terhadap urutan proses dan hubungan antar komponen dalam sistem.

Beberapa elemen utama dalam diagram urutan melibatkan objek (yang dapat berupa instansi kelas atau aktor), pesan yang dikirim antar objek, dan urutan waktu eksekusi pesan. Diagram ini membantu dalam memvisualisasikan alur eksekusi dan kolaborasi antar objek, membantu pengembang untuk memahami dan merancang perilaku suatu sistem.

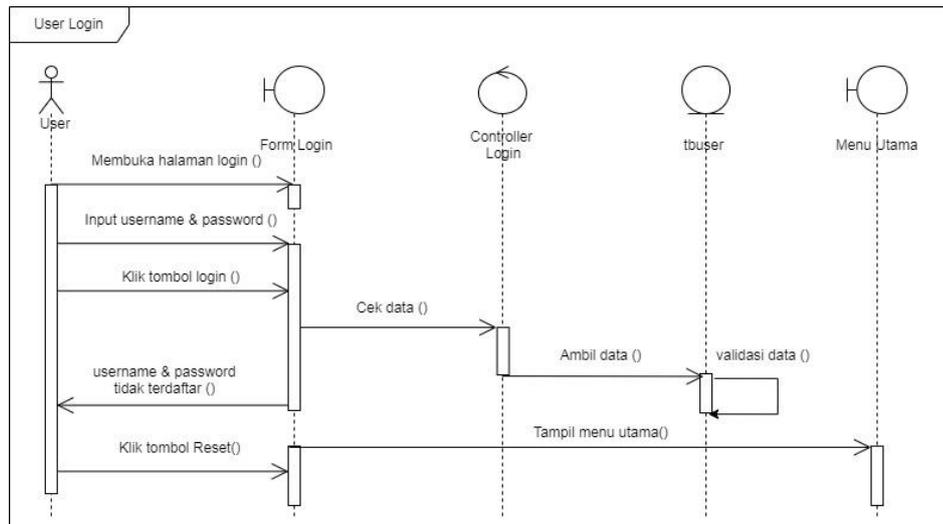
4.3.5.1 Sequence Diagram User Register



Gambar 4. 1 Sequence Diagram User Register

Gambar 4.11 menunjukkan alur utama saat pengguna melakukan registrasi. Proses dimulai ketika pengguna memilih menu register dan mengisi data pada form yang ditampilkan oleh sistem. Setelah pengguna mengklik tombol "Send", data dikirim ke controller untuk diproses dan disimpan ke dalam tabel `tbuser`. Terdapat juga opsi untuk mereset form jika pengguna ingin menghapus input yang telah dimasukkan.

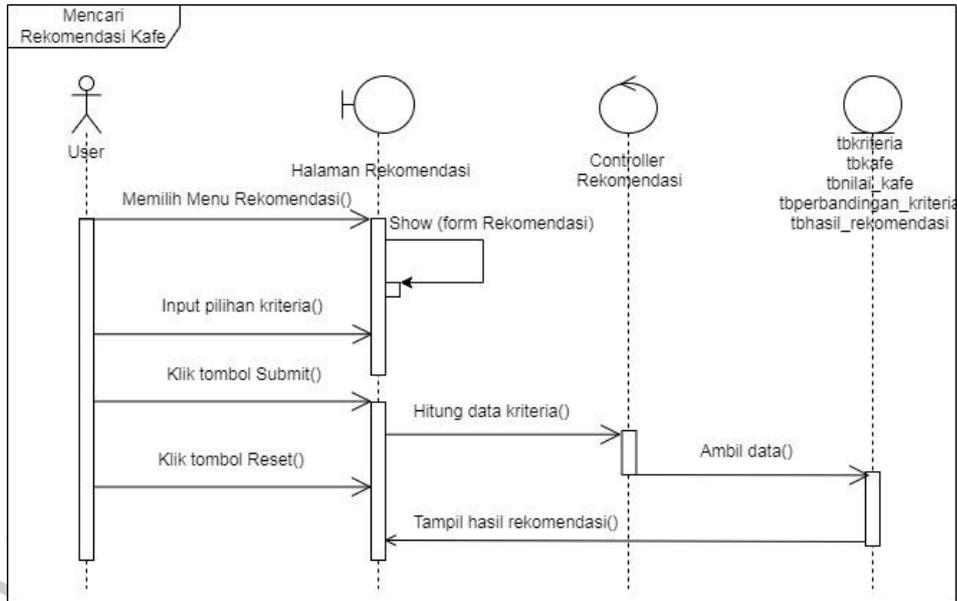
4.3.5.2 Sequence Diagram User Login



Gambar 4. 2 Sequence Diagram User Login

Gambar 4.12 memperlihatkan alur interaksi saat pengguna melakukan proses login. Setelah membuka halaman login dan menginput username serta password, sistem mengirimkan data tersebut ke controller untuk divalidasi terhadap data yang tersimpan di tabel `tbuser`. Jika data valid, sistem akan menampilkan menu utama. Jika tidak valid, pengguna akan menerima notifikasi bahwa data tidak ditemukan. Diagram ini juga menunjukkan opsi untuk mereset form login. Fokus utama interaksi berada pada proses pengecekan dan validasi kredensial pengguna.

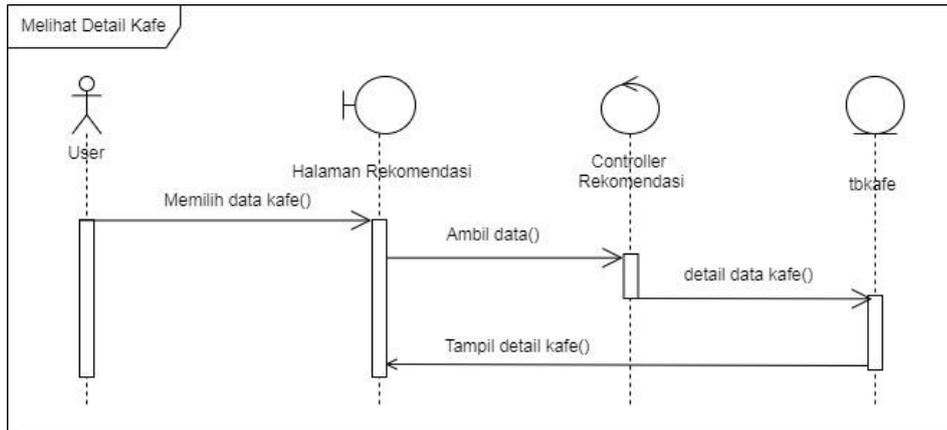
4.3.5.3 Sequence Diagram Mencari Rekomendasi Kafe



Gambar 4. 3 Sequence Diagram Mencari Rekomendasi Kafe

Gambar 4.13 menggambarkan proses saat pengguna mencari rekomendasi kafe berdasarkan kriteria tertentu. Setelah memilih menu rekomendasi dan mengisi preferensi, sistem memproses input tersebut menggunakan metode AHP untuk menentukan bobot setiap kriteria, lalu menerapkan metode SAW untuk menghitung skor dan melakukan perankingan kafe. Proses ini melibatkan pengambilan data dari beberapa tabel, termasuk kriteria, kafe, pembobotan, dan hasil rekomendasi. Setelah perhitungan selesai, sistem menampilkan daftar rekomendasi yang sesuai dengan preferensi pengguna. Terdapat juga opsi untuk mereset pilihan kriteria, sehingga pengguna dapat mengulang proses pemilihan sesuai preferensi yang baru.

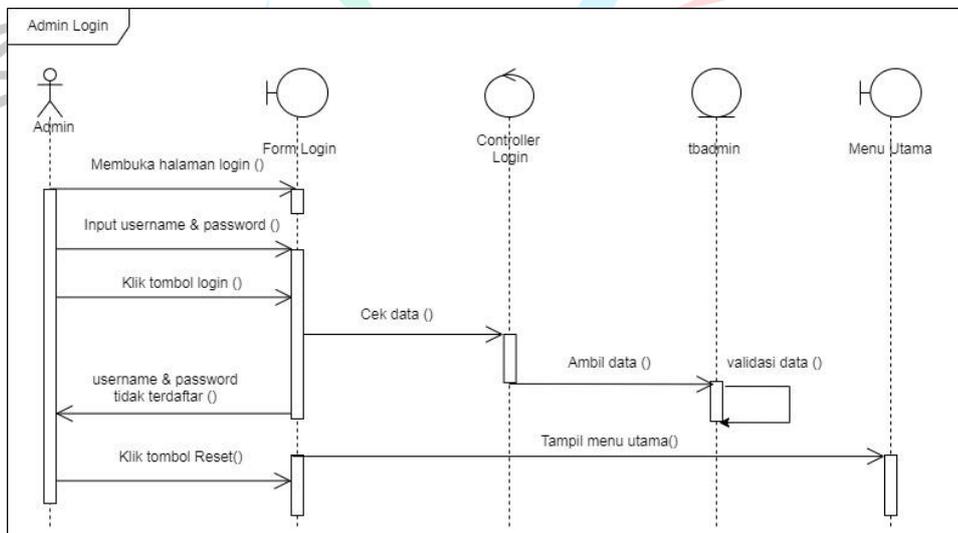
4.3.5.4 Sequence Diagram Melihat Detail Kafe



Gambar 4. 4 Sequence Diagram Melihat Detail Kafe

Gambar 4.14 menunjukkan alur ketika pengguna memilih salah satu kafe dari daftar rekomendasi untuk melihat detail informasinya. Setelah pengguna memilih data kafe, sistem mengirimkan permintaan ke controller untuk mengambil detail informasi dari tabel `tbkafe`, lalu menampilkannya pada halaman rekomendasi. Proses ini bersifat opsional dan berfungsi untuk membantu pengguna mengenali kafe sebelum membuat keputusan.

4.3.5.5 Sequence Diagram Admin Login

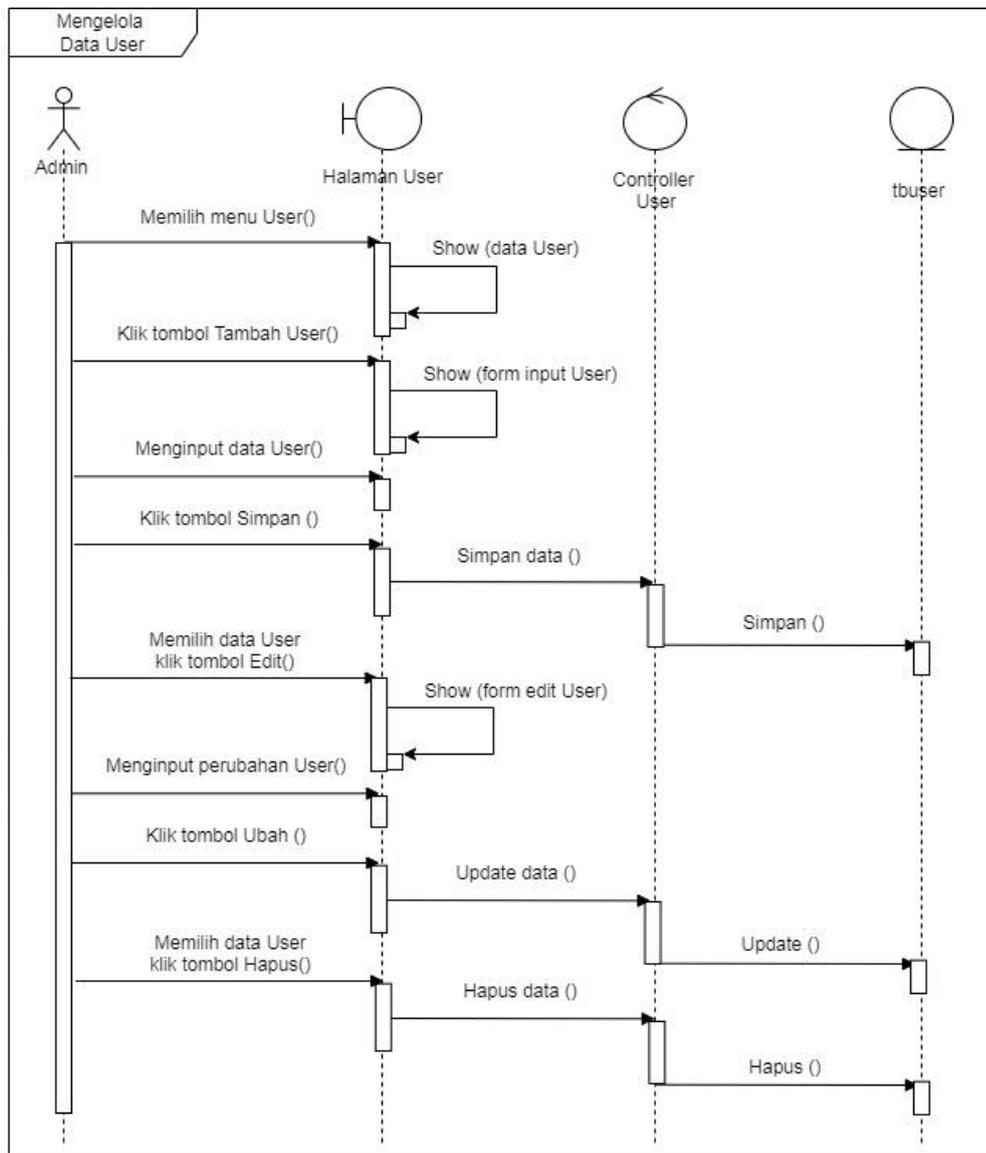


Gambar 4. 5 Sequence Diagram Admin Login

Gambar 4.15 memperlihatkan alur saat admin melakukan proses login ke sistem. Admin membuka halaman login dan mengisi username serta password, lalu

data dikirim ke controller untuk divalidasi terhadap informasi di tabel `tbadmin`. Jika data valid, sistem menampilkan menu utama admin. Jika tidak, admin akan menerima informasi bahwa data tidak ditemukan. Tersedia juga opsi reset untuk kembali ke halaman utama.

4.3.5.6 Sequence Diagram Mengelola Data User

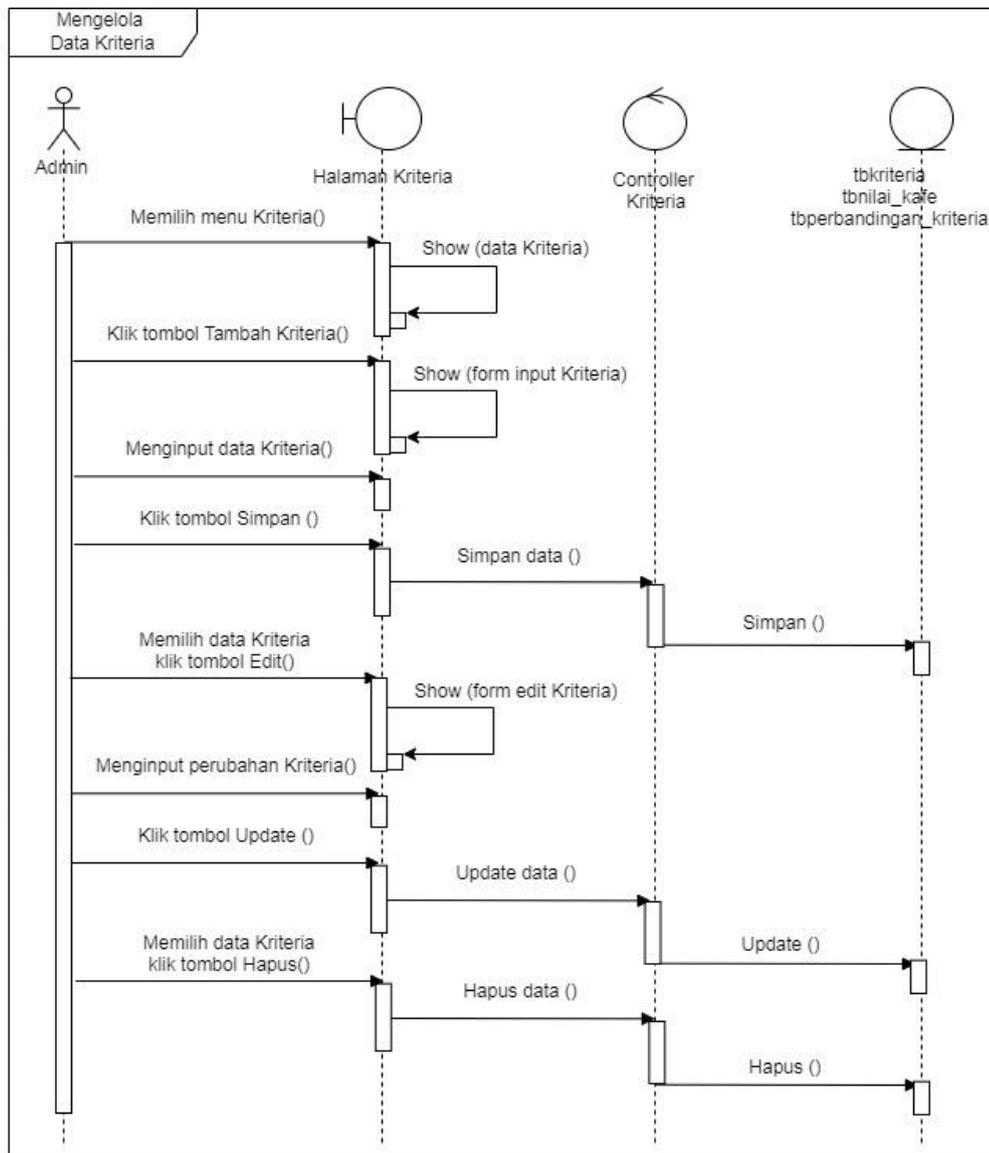


Gambar 4. 6 Sequence Diagram Mengelola Data User

Gambar 4.16 menggambarkan proses saat admin mengelola data pengguna. Proses dimulai ketika admin memilih menu user, lalu halaman data user ditampilkan. Admin dapat menambahkan data pengguna baru dengan mengisi form dan mengklik tombol Simpan. Selain itu, admin juga dapat memilih data pengguna

untuk diperbarui melalui tombol Edit, kemudian melakukan perubahan dan menyimpannya. Jika diperlukan, admin juga dapat menghapus data pengguna yang dipilih. Setiap aksi akan dikirim ke controller untuk diteruskan ke basis data pada tabel `tbuser`.

4.3.5.7 Sequence Diagram Mengelola Data Kriteria

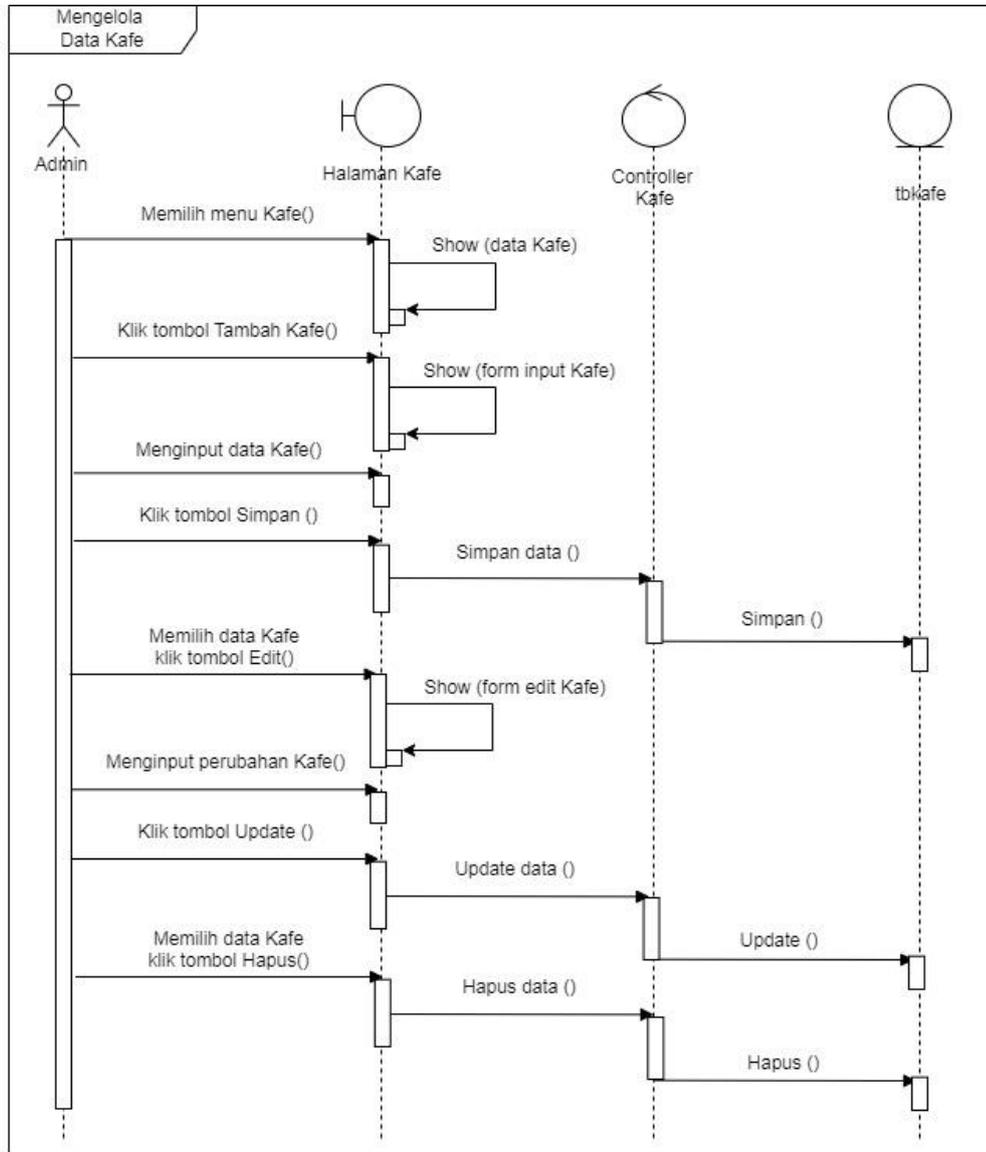


Gambar 4. 7 Sequence Diagram Mengelola Data Kriteria

Gambar 4.17 memperlihatkan proses pengelolaan data kriteria oleh admin. Setelah memilih menu kriteria, sistem menampilkan halaman data yang tersedia. Admin dapat menambahkan kriteria baru dengan membuka form input dan menyimpannya. Selain itu, admin juga dapat memilih kriteria yang ingin diperbarui

atau dihapus. Setiap aksi dikirimkan ke controller untuk diproses dan disimpan ke dalam tabel `tbkriteria`, serta berpotensi memengaruhi data pada tabel pendukung seperti `tbperbandingan_kriteria` dan `tbnilai_kafe`.

4.3.5.8 Sequence Diagram Mengelola Data Kafe



Gambar 4. 8 Sequence Diagram Mengelola Data Kafe

Gambar 4.18 menggambarkan proses pengelolaan data kafe oleh admin. Setelah memilih menu kafe, sistem menampilkan daftar data yang tersedia. Admin dapat menambahkan kafe baru dengan mengisi form dan menyimpannya, serta mengubah atau menghapus data kafe yang sudah ada. Setiap aksi dikirimkan ke controller untuk diproses dan diteruskan ke tabel `tbkafe` pada basis data.

4.3.6 Perancangan Database

Database atau basis data merupakan bagian yang tidak kalah penting dalam pembuatan perangkat lunak ini. Basis data dalam perangkat lunak ini menyimpan data mengenai pengguna dan informasi kafe yang akan dijelaskan dibawah.

4.3.6.1 Struktur Database

Database aplikasi ini dirancang untuk menggunakan 6 *table* dengan nama *user*, *kafe*, *kriteria*, *nilai_kafe*, *perbandingan_kriteria* dan *hasil_rekomendasi* yang kamus datanya dirancang seperti tabel dibawah.

Tabel 4. 10 Kamus Data *Database User*

No	Field	Tipe data	Length	Keterangan
1	Id_user (PK)	INT	10	User ID
2	Fname	VARCHAR	255	Nama depan user
3	Lname	VARCHAR	255	Nama belakang user
4	Email	VARCHAR	255	Email user
5	Notelp	INT	20	No telp user
6	Pass	VARCHAR	255	Password user
7	authlevel	INT	1	Identifikasi admin dan user

Tabel 4. 1 Kamus Data *Database Kafe*

No	Field	Tipe data	Length	Keterangan
1	Id_kafe (PK)	INT	10	Id kafe
2	Nama	VARCHAR	50	Nama kafe
3	Alamat	VARCHAR	100	Alamat Kafe
4	Wifi	BOOLEAN		Ketersediaan wifi
5	Rentang_harga	INT	10	Rentang harga

6	Area_rokok	BOOLEAN		Ketersediaan <i>smoking area</i>
7	Luas_area	INT	4	Luas area
8	Jumlah_kursi	VARCHAR	4	Jumlah kursi
9	Deskripsi	VARCHAR	1000	Deskripsi kafe
10	Gambar	VARCHAR	100	Directory file gambar kafe

Tabel 4. 2 Kamus Data Database *Kriteria*

No	Field	Tipe data	Length	Keterangan
1	Id_kriteria (PK)	INT	10	Id kriteria
2	Nama_kriteria	VARCHAR	50	Nama kafe
3	Bobot	FLOAT		Bobot kriteria kafe

Tabel 4. 3 Kamus Data Database *Nilai Kafe*

No	Field	Tipe data	Length	Keterangan
1	Id_nilai (PK)	INT	10	Id nilai
2	Id_kafe (FK)	INT	10	Id kafe
3	Id_kriteria (FK)	INT	10	Id kriteria
4	Nilai	FLOAT		Nilai kafe

Tabel 4. 4 Kamus Data Database *Perbandingan Kriteria*

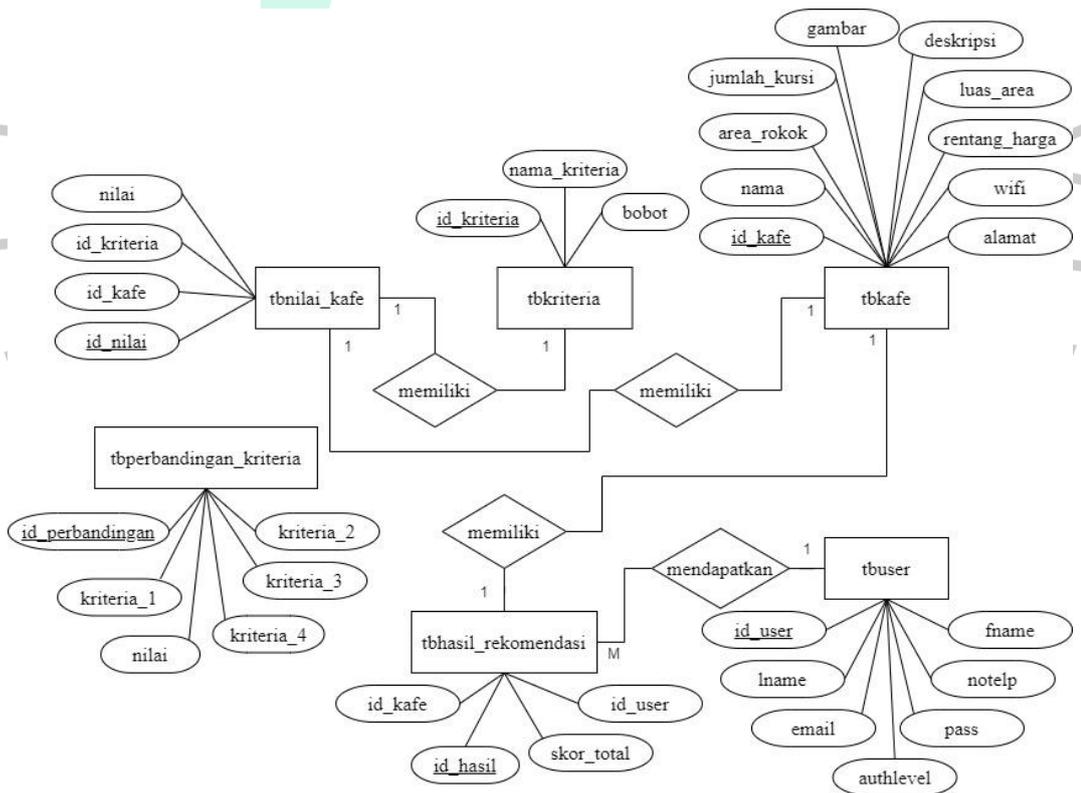
No	Field	Tipe data	Length	Keterangan
1	Id_perbandingan	INT	20	Id perbandingan
2	Kriteria_1	INT	20	Kriteria 1
3	Kriteria_2	INT	20	Kriteria 2
4	Kriteria_3	INT	20	Kriteria 3
5	Kriteria_4	INT	20	Kriteria 4
6	Nilai	FLOAT		Nilai perbandingan kafe

Tabel 4. 5 Kamus Data Database Hasil Rekomendasi

No	Field	Tipe data	Length	Keterangan
1	Id_hasil (PK)	INT	10	Id hasil
2	Id_user (FK)	INT	10	Id user
3	Id_kafe (FK)	INT	10	Id kafe
4	Skor_total	FLOAT		Nilai skor total

4.3.6.2 Entity Relationship Diagram

Entity Relationship Diagram (ERD) merupakan representasi visual yang digunakan untuk menggambarkan struktur data dan keterkaitan antar entitas dalam sebuah sistem. Diagram ini membantu merancang basis data dengan menunjukkan hubungan logis antara entitas dan atribut yang dimilikinya.



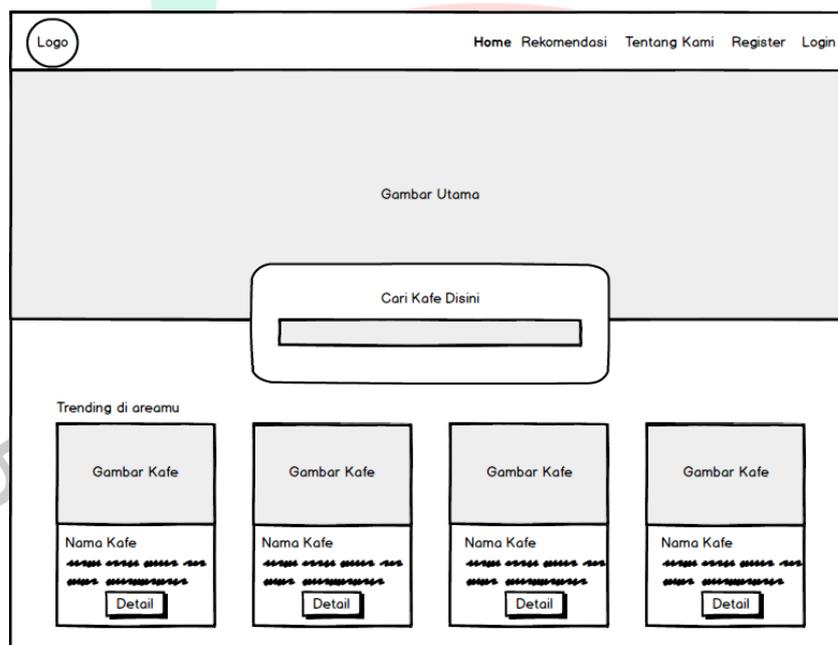
Gambar 4. 9 Entity Relationship Diagram Aplikasi Rekomendasi Kafe

Pada gambar diatas, terdapat 6 buah entitas berupa tbuser, tbkafe, tbkriteria, tbnilai_kafe, tbperbandingan_kriteria dan tbhasil_rekomendasi. beberapa

entitas tersebut memiliki relationship *one to one* dan ada satu yang hubungan antar tabel *one to many*.

4.3.7 Perancangan Interface

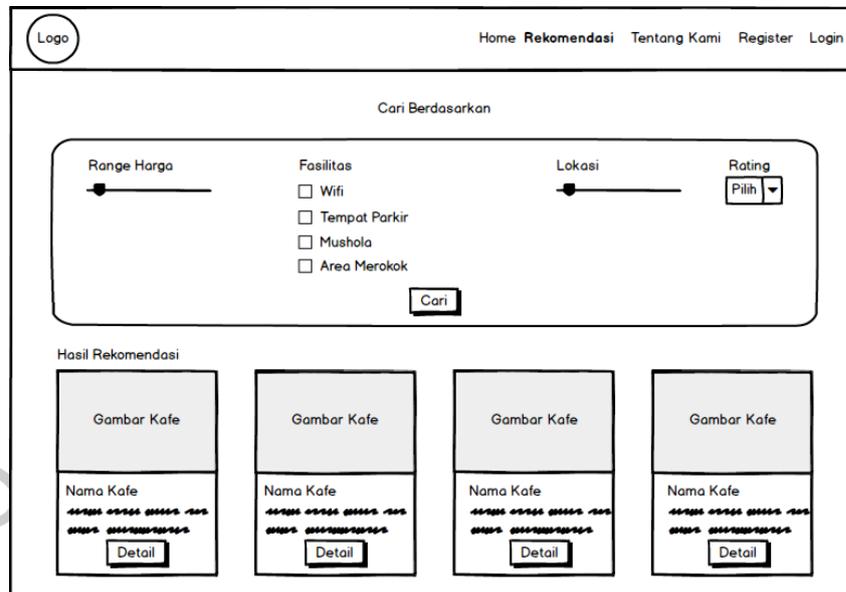
Perancangan interface merupakan proses merancang tampilan dan interaksi pengguna pada sebuah aplikasi atau sistem. Ini mencakup desain visual seperti tata letak elemen-elemen antarmuka, pemilihan warna dan jenis huruf, serta navigasi antarhalaman. Selain itu, perancangan antarmuka juga melibatkan pembuatan wireframe atau prototipe untuk menggambarkan bagaimana pengguna akan berinteraksi dengan aplikasi tersebut. Tujuan utamanya adalah untuk menghadirkan antarmuka yang intuitif, efisien, dan nyaman digunakan, sehingga pengguna dapat berinteraksi dengan aplikasi atau sistem secara lebih mudah dan efektif.



Gambar 4. 20 Rancangan Halaman Utama Aplikasi

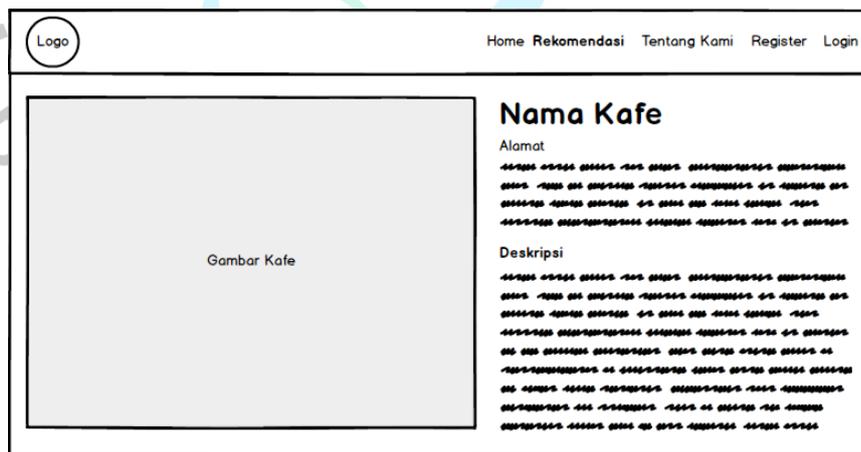
Gambar 4.20 adalah rancangan dari halaman utama aplikasi ketika dibuka oleh user. Halaman utama aplikasi menampilkan gambar utama yang merupakan salah satu dari kafe yang sudah terdaftar. Pada halaman ini juga ditampilkan kolom pencarian agar pengguna dapat mencari kafe secara langsung.

Terdapat elemen berupa *navbar* dimana pengguna dapat mengakses rekomendasi, tentang kami, register, dan menu *login*.



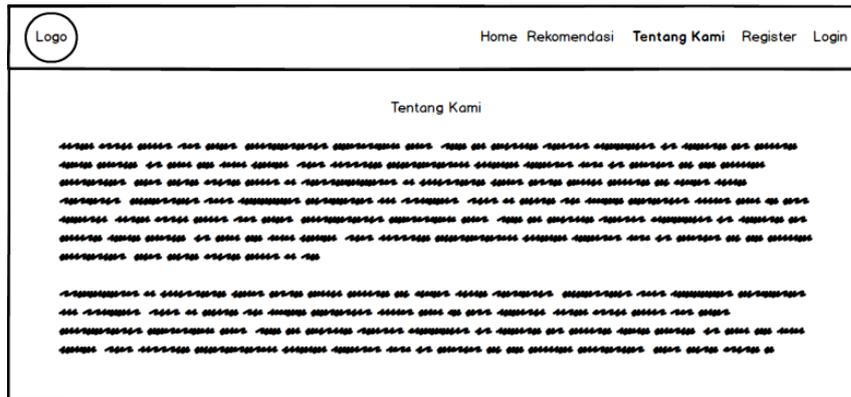
Gambar 4. 10 Rancangan Halaman Rekomendasi

Gambar 4.21 merupakan halaman rekomendasi yang menampilkan pilihan kriteria pencarian. Terdapat kriteria rentang harga, fasilitas, lokasi dan rating dari user.



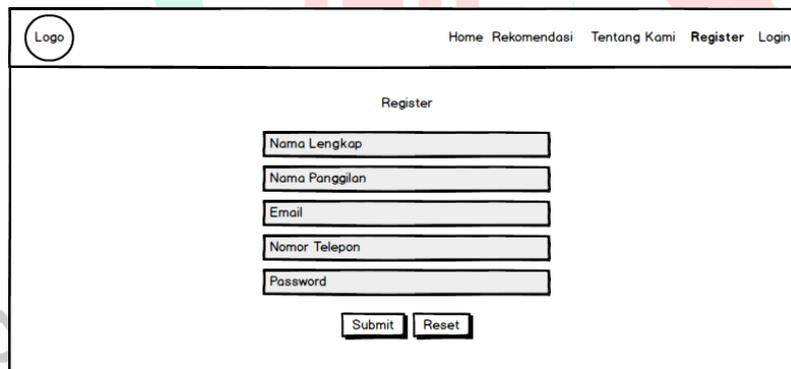
Gambar 4. 11 Rancangan Halaman Detail Kafe

Gambar 4.22 merupakan rancangan tampilan untuk detail kafe. Halaman ini menampilkan informasi berupa gambar, nama, alamat, deskripsi, dan detail lainnya dari sebuah kafe.



Gambar 4. 12 Rancangan Halaman Tentang Kami

Gambar 4.23 merupakan rancangan halaman tentang kami. Halaman ini menampilkan informasi tentang aplikasi ini, mulai cara pengaturan dan cara penggunaan.



Gambar 4. 13 Rancangan Halaman Register

Gambar 4.24 merupakan rancangan halaman register. Halaman ini menampilkan form register untuk membuat akun. Form register terdiri dari data profil user.

The image shows a web page layout for a login form. At the top left is a circular 'Logo' placeholder. To its right is a navigation menu with links: 'Home', 'Rekomendasi', 'Tentang Kami', 'Register', and 'Login'. The main content area is titled 'Login' and contains two input fields: 'Username' and 'Password'. Below these fields are two buttons: 'Login' and 'Reset'.

Gambar 4. 14 Rancangan Halaman Login

Gambar 4.25 merupakan rancangan halaman login. Halaman ini menampilkan form login. Form login terdiri dari isian username dan password.

The image shows a web page layout for an admin login form. It features a central rounded rectangular box containing the title 'Login', two input fields for 'Username' and 'Password', and two buttons labeled 'Login' and 'Reset'.

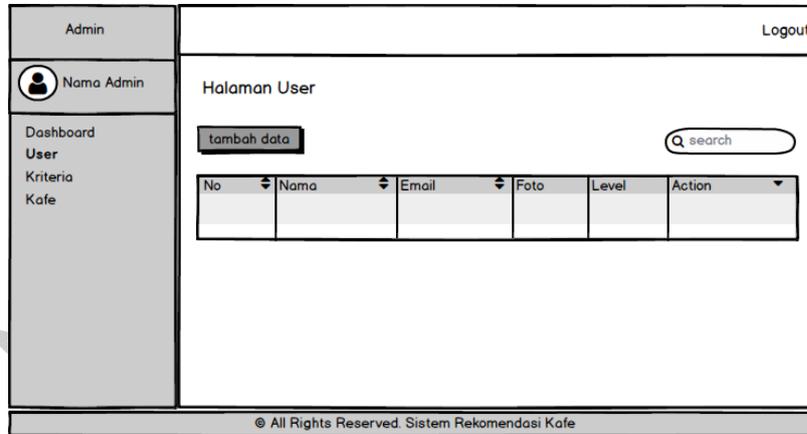
Gambar 4. 15 Rancangan Halaman Login Admin

Gambar 4.26 merupakan rancangan halaman login admin. Halaman ini menampilkan form login. Form login terdiri dari isian username dan password.

The image shows a dashboard layout. On the left is a vertical sidebar with a top section labeled 'Admin' containing a 'Logout' link. Below this is a user profile section with a person icon and the text 'Nama Admin'. The main content area is titled 'Halaman Dashboard' and displays three data boxes: 'User' with the value '2', 'Kriteria' with the value '4', and 'Kafe' with the value '50'. At the bottom of the page is a footer with the text '© All Rights Reserved. Sistem Rekomendasi Kafe'.

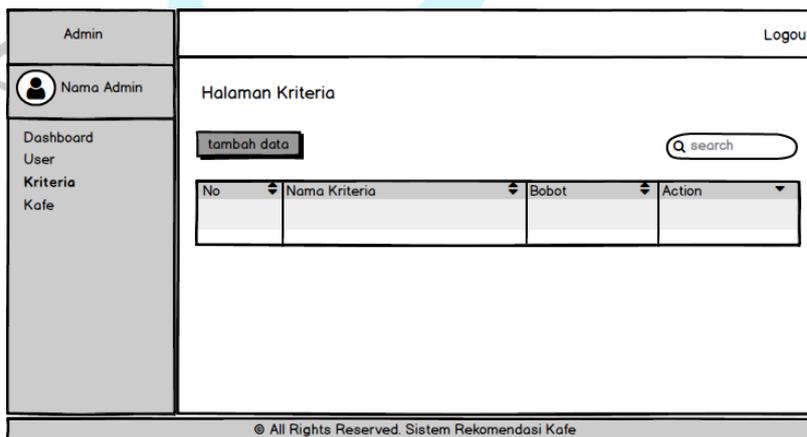
Gambar 4. 16 Rancangan Halaman Dashboard

Gambar 4.27 merupakan rancangan halaman dashboard. Halaman ini menampilkan informasi jumlah data user, kriteria dan kafe.



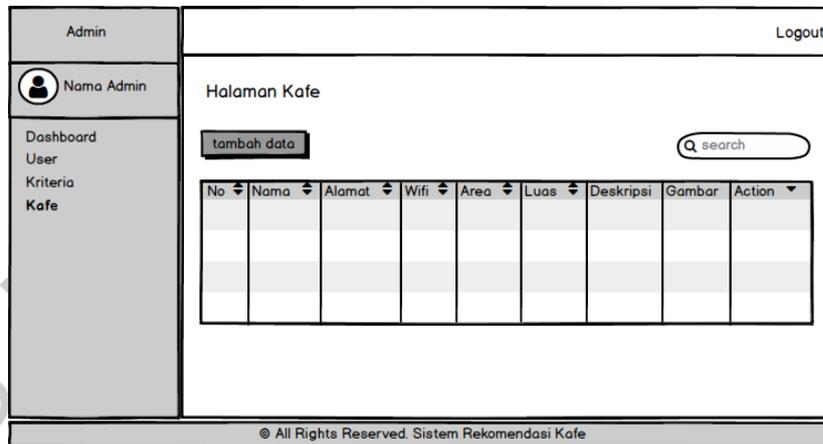
Gambar 4. 17 Rancangan Halaman Halaman User

Gambar 4.28 menampilkan rancangan antarmuka untuk halaman data pengguna. Halaman ini berfungsi sebagai panel administrasi yang memungkinkan admin melakukan pengelolaan data pengguna, seperti menambahkan, mengedit, dan menghapus informasi yang tersimpan dalam sistem.



Gambar 4. 18 Rancangan Halaman Halaman Kriteria

Gambar 4.29 merupakan rancangan halaman kriteria. Halaman ini menampilkan data kriteria. halaman ini digunakan admin untuk mengelola data kriteria yaitu menambah, mengubah dan menghapus data kriteria.



Gambar 4. 30 Rancangan Halaman Halaman Kafe

Gambar 4.30 merupakan rancangan antarmuka halaman kafe. Halaman ini digunakan oleh admin untuk menampilkan dan mengelola data kafe, termasuk menambah, mengubah, serta menghapus informasi yang tersimpan dalam sistem.

4.3.8 Perancangan *Testing*

4.3.8.1 Perancangan *Whitebox Testing*

White Box Testing dilakukan untuk menguji logika internal program, terutama pada implementasi algoritma AHP dan SAW dalam sistem rekomendasi. Pengujian ini bertujuan memastikan bahwa proses perhitungan dan alur logika berjalan sesuai yang diharapkan, bebas dari kesalahan, dan menghasilkan output yang konsisten.

Berikut merupakan tabel skenario *White Box Testing* beserta deskripsi dan indikator keberhasilannya:

Tabel 4. 6 Tabel *White box Testing*

No	Skenario Pengujian	Deskripsi Pengujian	Indikator Keberhasilan
1	Path Testing	Menguji semua jalur logika dalam modul perhitungan AHP dan SAW, termasuk cabang dan fungsi internal.	Semua jalur dieksekusi minimal sekali tanpa menghasilkan error atau output yang salah.
2	Loop Testing	Menguji logika perulangan (loop) seperti for, while, dan foreach dalam proses input, perbandingan berpasangan, dan iterasi skor.	Setiap loop berakhir sesuai kondisi yang ditetapkan, tanpa infinite loop atau crash.
3	Condition Testing	Menguji semua percabangan kondisi seperti if, else, dan switch pada proses validasi input dan pemrosesan data.	Semua kondisi dieksekusi dan memberikan hasil sesuai logika yang dirancang.
4	Hitung Bobot AHP	Menguji fungsi yang menghitung bobot kriteria berdasarkan perbandingan berpasangan dan normalisasi matriks.	Nilai bobot sama dengan perhitungan manual (selisih ≤ 0.01).
5	Hitung Skor SAW	Menguji fungsi yang menghitung skor total berdasarkan bobot AHP dan nilai alternatif dari masing-masing kafe.	Peringkat yang dihasilkan sesuai dengan skenario referensi atau hasil manual.

4.3.8.2 Perancangan *Blackbox Testing*

Black Box Testing melibatkan tahap input, eksekusi aplikasi, dan observasi keluaran dari sebuah aplikasi. Berikut merupakan tabel *Black Box* pengujian fitur.

Tabel 4. 77 Tabel *Testing Black Box*

No	Skenario Pengujian Aplikasi	Hasil Yang Diharapkan
1	<i>Register</i>	Pengguna berhasil melakukan registrasi
2	<i>Login aplikasi</i>	Pengguna berhasil melakukan login
3	Home	Setelah login, pengguna diarahkan ke halaman utama
4	Mencari kafe	Aplikasi menampilkan hasil pencarian user
5	Menggunakan fitur rekomendasi	Aplikasi menampilkan hasil rekomendasi berdasar inputan user
6	<i>Login admin</i>	Admin berhasil login ke halaman dashboard

No	Skenario Pengujian Aplikasi	Hasil Yang Diharapkan
1	<i>Register</i>	Pengguna berhasil melakukan registrasi
2	<i>Login aplikasi</i>	Pengguna berhasil melakukan login
7	Kelola User	Admin berhasil menambah, mengubah dan menghapus data user
8	Kelola Kriteria	Admin berhasil mengelola kriteria yaitu menambah, mengubah dan menghapus data kriteria

4.3.9 Perencanaan Pengembangan Lanjut

Penelitian ini menghasilkan sebuah sistem rekomendasi kafe berbasis web yang dibangun sebagai *proof of concept* (POC). Meski sistem telah mampu menjalankan fungsi dasar rekomendasi berdasarkan metode AHP dan SAW, terdapat sejumlah pengembangan lanjutan yang dapat dilakukan agar sistem dapat digunakan secara nyata oleh pengguna umum.

Beberapa rencana keberlanjutan sistem di antaranya:

1. Meningkatkan tampilan dan pengalaman pengguna (UI/UX)
2. Mengotomatisasi proses pengambilan data menggunakan *automated web scraping*
3. Melakukan uji coba penerimaan pengguna menggunakan *User Acceptance Testing* (UAT)
4. Menyesuaikan bobot kriteria berdasarkan preferensi individual
5. Mengintegrasikan analisis sentimen dari ulasan pengguna dengan metode NLP

Langkah-langkah tersebut diharapkan mampu meningkatkan skalabilitas dan keandalan sistem, serta menjadikannya lebih adaptif terhadap kebutuhan pengguna di masa depan.