

## **BAB 3**

### **METODOLOGI PENELITIAN**

#### **3.1 Studi Literatur**

Penelitian sistem pemesanan berbasis website menjadi langkah strategis yang semakin relevan di tengah transformasi digital. Sistem ini tidak hanya membantu mengotomatiskan proses pemesanan, tetapi juga meningkatkan efisiensi operasional dan pengalaman pengguna. Berbagai literatur menyebutkan bahwa keberhasilan implementasi sistem pemesanan bergantung pada kemampuannya untuk menyediakan informasi real-time mengenai ketersediaan produk, mempermudah proses transaksi, serta menyediakan antarmuka yang ramah pengguna.

Integrasi sistem pemesanan dengan fitur manajemen inventaris menjadi elemen kunci dalam memastikan ketersediaan barang secara akurat. Selain itu, penelitian menunjukkan bahwa pemanfaatan teknologi ini dapat membantu bisnis meningkatkan kecepatan layanan, mengurangi kesalahan manusia, serta mendukung pengambilan keputusan berbasis data. Hal ini menunjukkan pentingnya penelitian sistem yang tidak hanya memenuhi kebutuhan pelanggan, tetapi juga mampu memberikan nilai tambah bagi organisasi.

Metodologi Agile merupakan pendekatan yang semakin populer dalam penelitian perangkat lunak karena fleksibilitasnya dalam menghadapi perubahan kebutuhan proyek. Agile berfokus pada iterasi pendek, kolaborasi antar tim, dan umpan balik langsung dari pengguna, sehingga memungkinkan produk dikembangkan secara bertahap dan sesuai dengan kebutuhan yang dinamis.

Dalam konteks penelitian sistem pemesanan berbasis website, penerapan Agile memungkinkan penyesuaian spesifikasi yang berkelanjutan berdasarkan masukan pengguna. Studi sebelumnya mengungkapkan bahwa pendekatan ini mempercepat waktu penelitian dan meningkatkan kualitas hasil akhir. Kolaborasi intensif antara tim peneliti dan pemangku kepentingan juga menjadi salah satu faktor keberhasilan penerapan Agile, terutama dalam memastikan bahwa setiap iterasi mampu memberikan nilai yang signifikan bagi pengguna.

Black Box Testing adalah metode pengujian perangkat lunak yang fokus pada pengujian fungsi sistem berdasarkan spesifikasi tanpa memeriksa struktur internalnya. Pendekatan ini sangat efektif dalam mengidentifikasi kesalahan pada

input-output, validasi data, serta fungsi utama sistem.

Dalam penelitian sistem pemesanan berbasis website, Black Box Testing digunakan untuk memastikan bahwa fitur utama seperti pencarian produk, pengelolaan keranjang belanja, proses pembayaran, dan pemberitahuan berjalan sesuai harapan. Penelitian terdahulu menunjukkan bahwa pengujian ini mampu meningkatkan keandalan sistem dengan mendeteksi potensi kesalahan yang dapat memengaruhi pengalaman pengguna.

Skala Likert adalah metode evaluasi yang sering digunakan untuk mengukur persepsi, opini, atau tingkat kepuasan pengguna terhadap suatu sistem. Dalam penelitian sistem pemesanan berbasis website, metode ini dapat membantu peneliti memahami pengalaman pengguna secara lebih mendalam, seperti kemudahan navigasi, kecepatan sistem, dan desain antarmuka.

Penelitian menunjukkan bahwa hasil evaluasi dengan Skala Likert dapat digunakan untuk mengidentifikasi area yang memerlukan perbaikan, misalnya dalam aspek kejelasan antarmuka atau waktu respons sistem. Evaluasi ini biasanya dilakukan setelah sistem diuji melalui User Acceptance Test (UAT) untuk mendapatkan umpan balik langsung dari pengguna, sehingga proses penelitian dapat terus disempurnakan berdasarkan kebutuhan nyata.

Manajemen inventaris merupakan elemen krusial dalam mendukung operasional sistem pemesanan. Literatur menyebutkan bahwa integrasi antara sistem pemesanan berbasis website dan manajemen inventaris memungkinkan organisasi untuk meminimalkan kesalahan stok, mempercepat pembaruan data, dan meningkatkan efisiensi operasional.

Beberapa studi menggarisbawahi pentingnya penggunaan teknologi pemantauan stok real-time dan algoritma prediktif untuk memperkirakan kebutuhan stok di masa mendatang. Dengan pendekatan ini, organisasi dapat memastikan kelancaran proses pemesanan tanpa adanya gangguan akibat ketidaksesuaian stok. Praktik terbaik dalam manajemen inventaris mencakup penggunaan alat otomatisasi untuk pembaruan stok dan analisis data guna mendukung keputusan strategis.

Berdasarkan studi literatur, penelitian sistem pemesanan berbasis website memerlukan pendekatan yang terintegrasi. Hal ini mencakup penerapan metodologi Agile untuk memastikan fleksibilitas dalam penelitian, pengujian fungsional menggunakan Black Box Testing, serta evaluasi pengguna dengan Skala Likert untuk meningkatkan pengalaman dan kepuasan pengguna. Selain itu, manajemen inventaris yang terintegrasi menjadi salah satu komponen penting untuk menjamin kelancaran operasional dan kepuasan pelanggan.

Dengan mengacu pada teori dan praktik terbaik yang telah teruji, diharapkan sistem yang dikembangkan mampu memenuhi kebutuhan bisnis CV. Marannu Karya Sejahtera, memberikan pengalaman pengguna yang optimal, serta mendukung tujuan strategis perusahaan. Studi literatur ini menjadi landasan kuat untuk memastikan bahwa setiap aspek penelitian dilakukan secara komprehensif dan selaras dengan standar industri yang berlaku.

### **3.2 Analisis Kebutuhan**

Analisis kebutuhan dilakukan berdasarkan data yang diperoleh dari wawancara serta hasil studi literatur. Informasi yang dikumpulkan digunakan untuk mengidentifikasi kebutuhan fungsional dan non-fungsional sistem. Proses ini bertujuan untuk memastikan bahwa sistem yang dikembangkan dapat memenuhi ekspektasi pengguna dan mendukung tujuan bisnis yang diinginkan.

Dalam tahap ini, *user stories* dan *use cases* disusun untuk memetakan kebutuhan pengguna ke dalam skenario konkret. Hal ini melibatkan keterlibatan langsung pemangku kepentingan sehingga penelitian sistem dapat berjalan selaras dengan kebutuhan bisnis dan memberikan solusi yang efektif. Dengan pendekatan ini, rancangan sistem tidak hanya berorientasi pada efisiensi teknis tetapi juga pada pengalaman pengguna dan dampak bisnis secara keseluruhan.

#### **3.2.1 Proses Bisnis CV. Marannu Karya Sejahtera**

CV Marannu Karya Sejahtera adalah sebuah usaha mikro kecil dan menengah (UMKM) yang bergerak di bidang makanan dan minuman, khususnya penyediaan jasa boga (catering). Proses bisnis yang diterapkan dalam operasional catering perusahaan ini mencakup beberapa tahap utama, dimulai dari penerimaan pesanan, pengelolaan stok bahan makanan, hingga pengiriman produk ke pelanggan. Seluruh proses ini bertujuan untuk memastikan bahwa layanan yang diberikan berjalan secara efisien, dengan meminimalkan potensi kesalahan yang dapat menghambat operasional.

Proses bisnis dimulai dengan tahap penerimaan pesanan. Dalam tahap ini, pelanggan melakukan pemesanan melalui sistem yang telah dirancang. Pesanan mencakup layanan-layanan yang ditawarkan oleh CV Marannu Karya Sejahtera, seperti penyediaan nasi kotak, prasmanan untuk acara besar, tumpeng sebagai simbol perayaan, dan snack box untuk rapat atau kegiatan serupa. Sistem mencatat

semua informasi yang berkaitan dengan pesanan, termasuk jenis layanan, jumlah kebutuhan, serta tanggal dan waktu pengiriman yang diinginkan pelanggan.

Selanjutnya, tahap pengelolaan stok bahan makanan menjadi elemen penting dalam operasional. Sistem pengelolaan inventaris yang diterapkan oleh perusahaan berfungsi secara real-time untuk mencatat ketersediaan bahan makanan serta mencatat barang yang masuk dan keluar. Barang masuk mencakup bahan baku yang diperoleh dari pemasok, sementara barang keluar mengacu pada bahan baku yang digunakan untuk memenuhi pesanan. Sistem ini juga dirancang untuk memberikan notifikasi otomatis ketika stok bahan mendekati batas minimum, sehingga memungkinkan pemilik usaha untuk segera mengambil langkah proaktif, seperti melakukan pemesanan ulang kepada pemasok. Hal ini bertujuan untuk memastikan bahwa bahan baku selalu tersedia dan risiko kekurangan stok dapat diminimalkan.

Tahap selanjutnya adalah produksi pesanan, di mana bahan makanan yang tersedia diolah sesuai dengan kebutuhan pelanggan. Proses ini dilakukan berdasarkan spesifikasi pesanan yang tercatat dalam sistem, dengan memastikan bahwa kualitas produk tetap terjaga. Setelah proses produksi selesai, produk yang telah disiapkan akan dikemas dan disiapkan untuk pengiriman.

Pengiriman produk ke pelanggan merupakan tahap akhir dalam proses bisnis ini. Sistem yang diterapkan juga mendukung pencatatan detail barang yang keluar, termasuk tujuan pengiriman dan waktu yang telah ditentukan. Hal ini bertujuan untuk memastikan bahwa pengiriman dilakukan secara tepat waktu dan sesuai dengan kebutuhan pelanggan.

Sebagai hasil dari seluruh proses tersebut, sistem menghasilkan berbagai jenis laporan yang bermanfaat bagi pemilik usaha. Laporan-laporan ini mencakup histori pesanan, status stok bahan makanan, dan riwayat pergerakan barang. Dengan adanya laporan ini, pemilik usaha dapat melakukan evaluasi dan perencanaan bisnis yang lebih efektif, mendukung transformasi digital perusahaan menuju pengelolaan yang lebih modern dan efisien.

### **3.2.2 Kebutuhan Fungsional**

Kebutuhan fungsional dari sistem yang akan dikembangkan mencakup fitur-fitur utama yang diperlukan untuk mendukung pengelolaan inventaris bahan makanan secara efisien. Salah satu kebutuhan yang paling mendesak adalah pengelolaan stok secara *real-time*. Sistem ini harus mampu mencatat, memperbarui,

dan menampilkan data stok bahan makanan maupun peralatan dengan akurasi tinggi. Fitur ini penting untuk memastikan bahwa bahan yang mudah rusak dapat dimonitor secara kontinu sehingga pemborosan akibat stok berlebih dapat dihindari. Selain itu, pemantauan stok secara *real-time* memungkinkan catering untuk mempersiapkan diri menghadapi permintaan mendadak, terutama pada jam operasional yang sibuk.

Penelitian ini cocok untuk menggunakan metode Agile dalam pengembangan situs web. Hal ini dikarenakan metode Agile memiliki fleksibilitas terhadap perubahan kebutuhan. Agile memungkinkan adaptasi yang cepat terhadap kebutuhan pengguna yang dinamis, sebagaimana tercermin dari kebutuhan pengelolaan inventaris CV. Marannu Karya Sejahtera yang dapat berubah sesuai dengan kondisi bisnis. Selain itu, iterasi singkat untuk evaluasi cepat juga menjadi poin utama dalam metode Agile. Metode Agile membagi pengembangan menjadi sprint-sprint pendek yang memungkinkan pengembangan secara bertahap, sehingga fitur seperti pengelolaan stok dan pencatatan barang dapat segera diimplementasikan dan dievaluasi. Metode agile juga memiliki pengujian dan perbaikan yang dilakukan secara berkelanjutan. Dengan Agile, setiap iterasi mencakup proses pengujian seperti Black Box Testing yang memastikan bahwa fitur yang dikembangkan sesuai dengan requirement, dan UAT yang relevan dengan tujuan penelitian untuk mengevaluasi tingkat kepuasan pengguna secara berkala.

Fitur penting lainnya adalah pencatatan barang masuk dan keluar. Sistem ini menyediakan mekanisme untuk mencatat setiap transaksi, baik berupa bahan makanan yang ditambahkan ke dalam stok (barang masuk) maupun bahan yang digunakan atau dikeluarkan dari stok (barang keluar). Pencatatan ini harus disimpan dalam basis data untuk memberikan histori transaksi yang jelas dan dapat diakses kapan saja. Hal ini tidak hanya berguna untuk memantau aliran bahan, tetapi juga mendukung pembuatan laporan yang lebih akurat.

Selanjutnya, sistem juga harus dilengkapi dengan kemampuan memberikan notifikasi otomatis ketika stok suatu bahan mencapai ambang batas minimum yang telah ditentukan. Fitur ini sangat penting untuk membantu pemilik usaha dan staf melakukan tindakan proaktif, seperti memesan ulang bahan makanan, sebelum stok benar-benar habis. Dengan adanya notifikasi ini, risiko gangguan operasional akibat kekurangan bahan dapat diminimalisasi.

Selain itu, sistem harus memiliki fungsi pelaporan inventaris yang memungkinkan pengguna untuk menghasilkan laporan secara periodik, baik harian, mingguan, maupun bulanan. Laporan ini akan membantu pemilik

usaha dalam memahami pola penggunaan bahan makanan, mengidentifikasi potensi pemborosan, dan melakukan perencanaan pembelian secara lebih efektif. Fitur pencarian dan filter juga diperlukan untuk mempermudah pengguna dalam menemukan data tertentu di antara banyaknya catatan inventaris. Sistem ini harus dapat menyaring informasi berdasarkan kategori, tanggal, atau parameter lain yang relevan.

Untuk memastikan bahwa sistem dapat digunakan oleh berbagai pihak yang terlibat dalam operasional catering, fitur akses multi-pengguna juga harus disediakan. Sistem ini harus mendukung pembagian hak akses berdasarkan peran pengguna, sehingga hanya pengguna yang berwenang, seperti pemilik usaha, yang dapat mengakses data sensitif, sementara *customer* hanya dapat membuat dan melihat daftar transaksi yang dimiliki.

### **3.2.3 Kebutuhan Non-Fungsional**

Selain kebutuhan fungsional, sistem juga harus memenuhi serangkaian kebutuhan non-fungsional yang memastikan kualitas dan keandalan sistem dalam mendukung operasional catering. Salah satu aspek yang paling penting adalah kemudahan penggunaan atau *usability*. Antarmuka pengguna sistem ini harus dirancang sedemikian rupa sehingga mudah dipahami oleh pengguna dari berbagai latar belakang, termasuk mereka yang tidak memiliki keahlian teknis. Tampilan yang sederhana namun intuitif akan membantu meningkatkan produktivitas pengguna dalam mengoperasikan sistem.

Dari sisi ketersediaan atau *availability*, sistem berbasis web ini harus selalu dapat diakses selama jam operasional catering. Hal ini penting untuk memastikan bahwa data stok dapat diperbarui dan diakses kapan saja tanpa gangguan. Ketersediaan sistem yang konsisten akan mendukung kelancaran operasional catering dan mencegah kendala yang disebabkan oleh ketidakterediaan informasi.

Keamanan data juga menjadi aspek penting dalam kebutuhan non-fungsional. Sistem ini harus memiliki mekanisme autentikasi dan otorisasi yang ketat untuk memastikan bahwa hanya pengguna yang berwenang yang dapat mengakses informasi tertentu. Sebagai tambahan, data transaksi dan informasi sensitif lainnya harus dienkripsi untuk melindungi dari risiko pencurian atau kebocoran data. Keamanan yang kuat akan meningkatkan kepercayaan pengguna terhadap sistem.

Kinerja sistem atau *performance* juga harus menjadi perhatian utama. Sistem harus dirancang untuk dapat menangani banyak transaksi secara simultan, terutama selama jam sibuk, tanpa menurunkan kecepatan atau responsivitas. Hal ini sangat penting untuk menjaga efisiensi operasional dan kepuasan pengguna.

Terakhir, sistem harus bersifat portabel dan kompatibel dengan berbagai perangkat, termasuk komputer desktop, laptop, dan smartphone. Dengan kompatibilitas ini, pengguna dapat mengakses sistem dari mana saja, baik dari kantor maupun di lokasi lain. Portabilitas ini memberikan fleksibilitas yang lebih besar bagi pemilik usaha dan staf dalam menjalankan tugas mereka, terutama ketika mereka tidak berada di kantor catering.

Dengan pemenuhan kebutuhan non-fungsional ini, sistem yang dikembangkan tidak hanya akan memberikan solusi teknis yang andal tetapi juga mendukung peningkatan kualitas operasional secara keseluruhan.

#### **3.2.4 Rekomendasi Solusi Sistem**

Berdasarkan analisis kebutuhan di atas, solusi yang direkomendasikan adalah website yang dirancang khusus untuk pengelolaan inventaris catering. Website ini akan mengintegrasikan seluruh fitur yang dibutuhkan, termasuk pencatatan stok *real-time*, pelaporan inventaris, dan pemberian notifikasi stok rendah.

Desain sistem ini diharapkan dapat meningkatkan efisiensi pengelolaan bahan makanan, mengurangi pemborosan, dan mendukung perencanaan operasional yang lebih baik untuk CV. Marannu Karya Sejahtera.

### **3.3 Wawancara**

Wawancara dilakukan dengan pihak-pihak terkait, seperti pemilik usaha, karyawan, dan calon pengguna sistem, guna memperoleh pemahaman yang lebih mendalam mengenai dinamika bisnis, tantangan operasional, serta kebutuhan spesifik terkait pengelolaan inventaris. Pertanyaan-pertanyaan yang diajukan disusun secara sistematis, mencakup aspek kritis seperti kendala dalam pengelolaan stok, preferensi penggunaan sistem baru, serta harapan terhadap peningkatan efektivitas operasional melalui sistem yang diusulkan.

Proses wawancara ini tidak hanya berfokus pada permasalahan teknis seperti pencatatan stok bahan baku, tetapi juga memperhatikan faktor-faktor kualitatif,

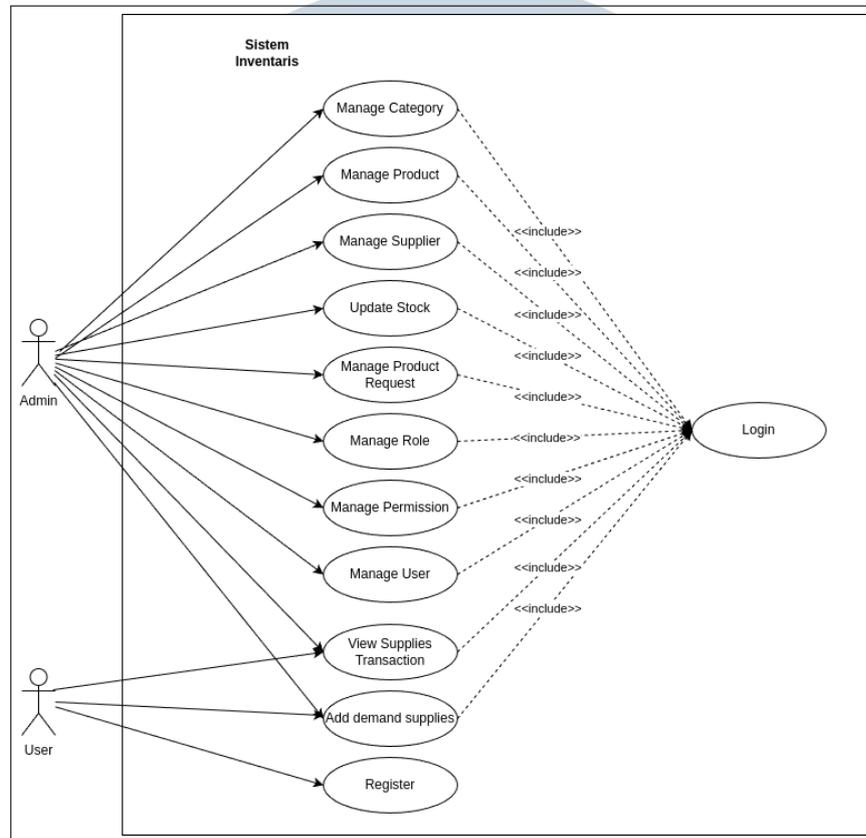
seperti kepuasan pengguna dan pengaruh sistem terhadap produktivitas usaha. Dengan pendekatan ini, penelitian dapat menghasilkan gambaran yang lebih komprehensif tentang kebutuhan pengguna dan permasalahan bisnis yang ada, sehingga menjadi landasan utama dalam perancangan sistem informasi inventaris yang lebih responsif, terstruktur, dan efisien. Dalam penelitian ini, wawancara dilakukan secara langsung melalui tatap muka bersama narasumber, khususnya pemilik usaha CV. Marannu Karya Sejahtera, untuk memastikan keakuratan dan kedalaman data yang diperoleh.

### **3.4 Perancangan Website**

Tahap ini mencakup pembuatan berbagai diagram UML untuk menggambarkan alur kerja sistem, termasuk Activity Diagram untuk alur aktivitas, Use Case Diagram untuk aktor dan fungsionalitas, dan Entity Relationship Diagram (ERD) untuk struktur dan hubungan data.



### 3.4.1 Use Case Diagram



Gambar 3.1. Use Case Diagram

Dalam Gambar 3.1, admin diwajibkan untuk login terlebih dahulu sebelum dapat mengakses fitur-fitur yang tersedia. Setelah berhasil login, admin memiliki kemampuan untuk menambahkan kategori, menambahkan pemasok, dan menambahkan persediaan. Admin juga dapat memperbarui stok yang tersedia dan melihat transaksi persediaan yang telah terjadi. Selain itu, admin memiliki opsi untuk keluar (logout) dari sistem setelah menyelesaikan tugas-tugasnya. Seluruh fitur ini memerlukan autentikasi awal melalui proses login, yang menjadi langkah wajib sebelum menggunakan fungsi lainnya.

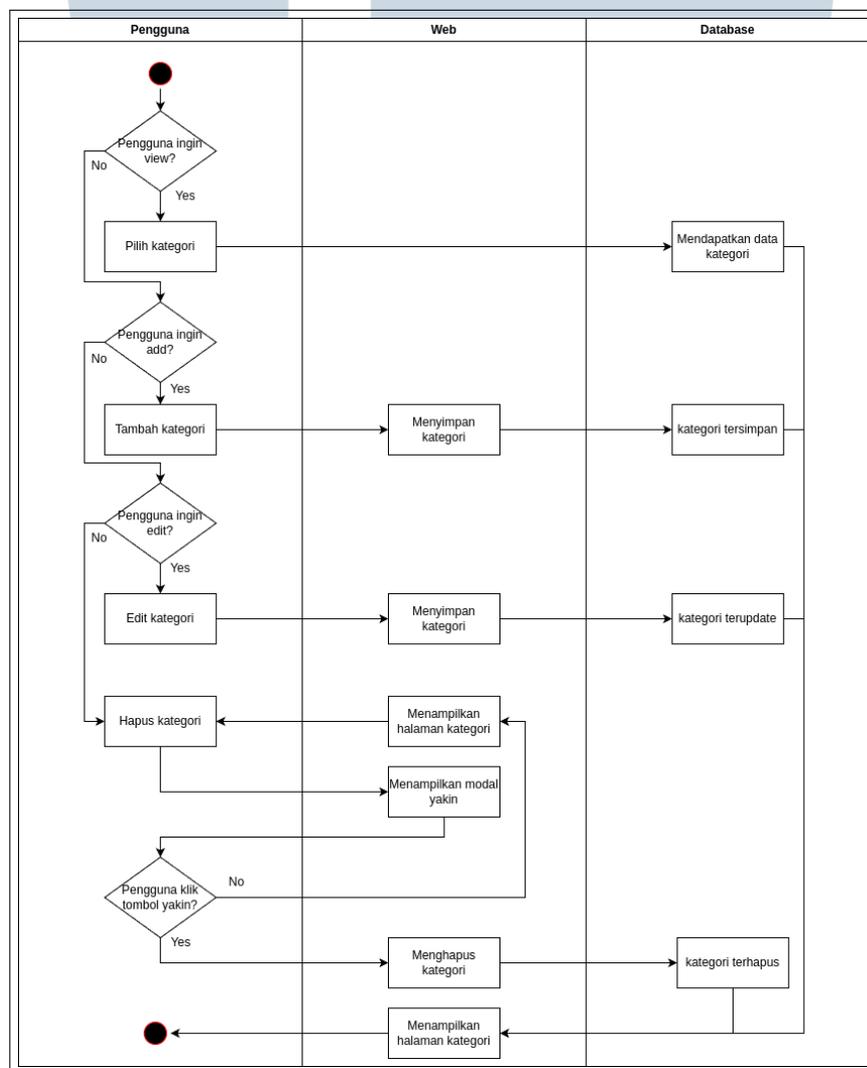
Dalam Gambar 3.1 juga, karyawan memiliki beberapa fitur yang dapat diakses setelah melakukan *login*. Pertama, karyawan dapat menambahkan permintaan persediaan untuk kebutuhan operasional melalui fitur *Add Demand Supplies*. Selain itu, karyawan juga dapat melihat transaksi persediaan yang telah terjadi menggunakan fitur *View Supplies Transaction*. Setelah menyelesaikan tugasnya, karyawan memiliki opsi untuk keluar dari sistem dengan fitur *Logout*.

Semua fitur ini hanya dapat diakses setelah karyawan berhasil *login*. Diagram ini memberikan gambaran interaksi antara karyawan dan sistem, sementara detail teknis seperti validasi data dan mekanisme keamanan *login* tidak dijelaskan secara rinci.

### 3.4.2 Activity Diagram

#### A Activity Diagram Admin

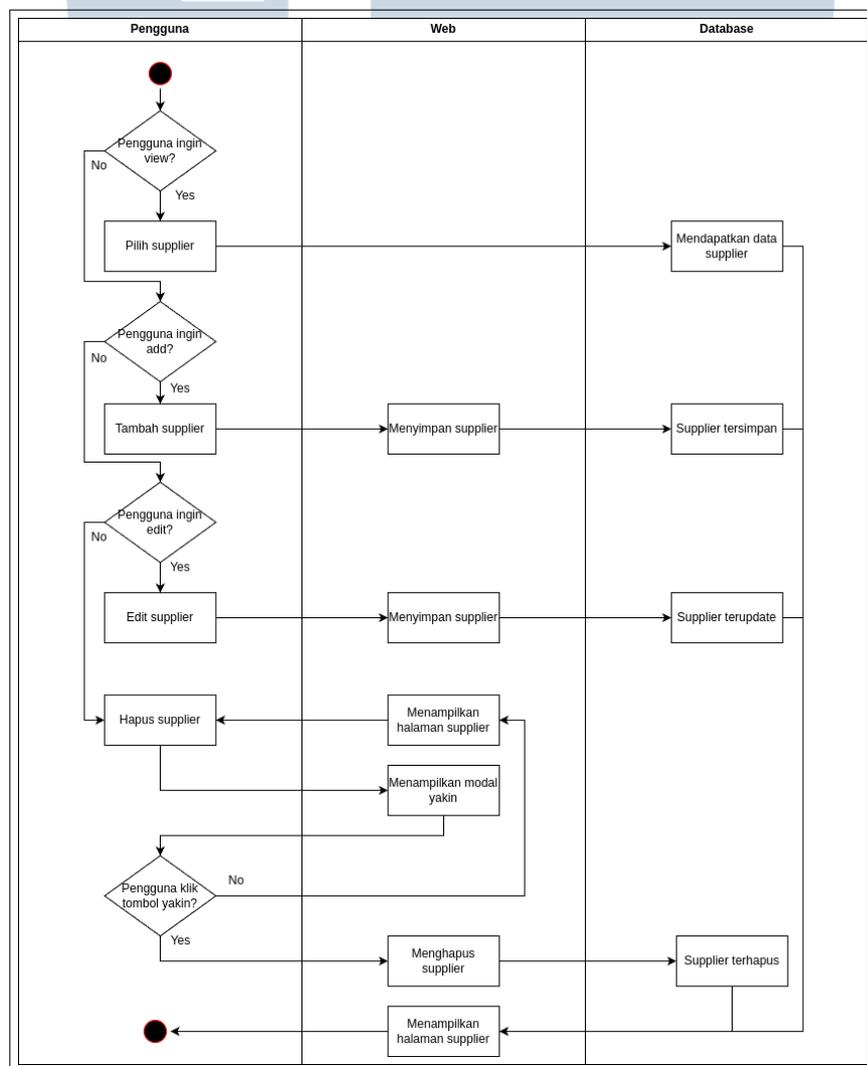
##### A.1 Activity Diagram Manage Category



Gambar 3.2. Activity Diagram Kategori Admin

Gambar 3.2 menunjukkan activity diagram pengelolaan kategori oleh admin. Admin dapat memilih menu kategori untuk melihat data kategori, menambah kategori baru, atau menghapus kategori yang ada. Penambahan kategori dilakukan dengan mengisi data yang kemudian disimpan ke database. Untuk penghapusan kategori, sistem meminta konfirmasi dari admin sebelum menghapus data dari database. Setelah proses selesai, halaman kategori akan diperbarui sesuai perubahan yang dilakukan.

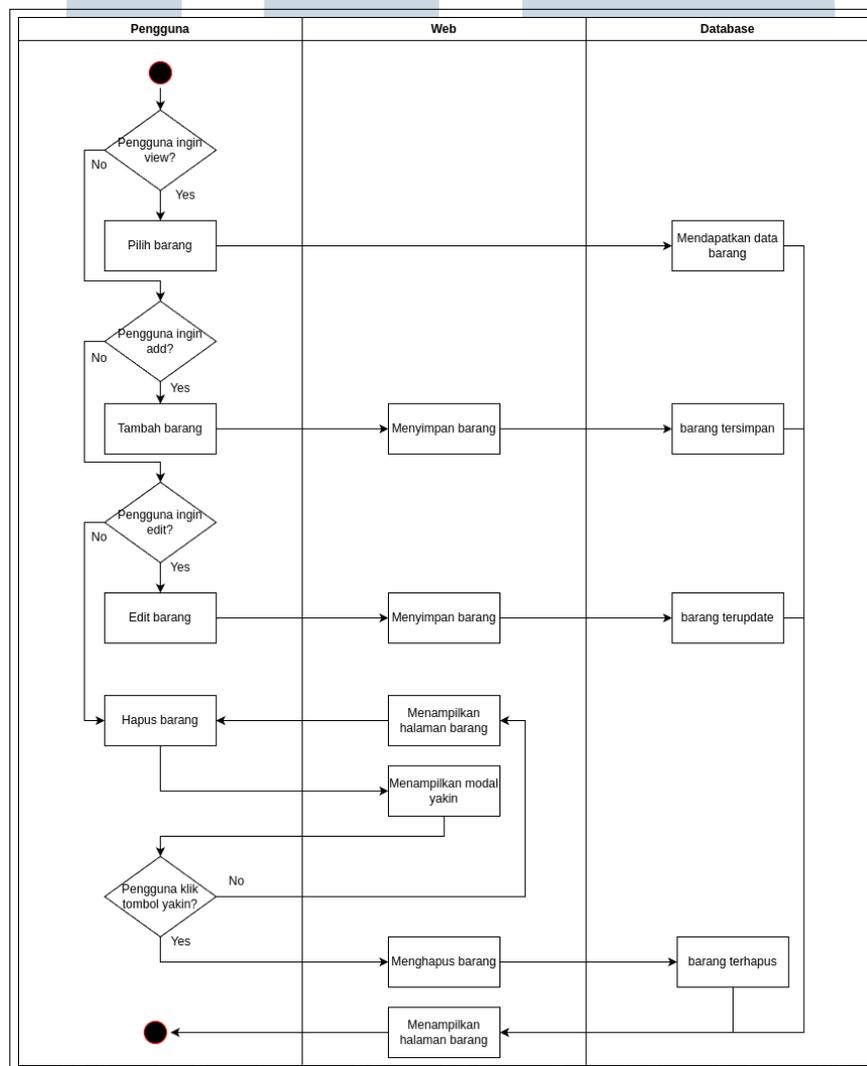
## A.2 Activity Diagram Manage Supplier



Gambar 3.3. Activity Diagram Supplier Admin

Gambar 3.3 menunjukkan activity diagram pengelolaan pemasok (*supplier*) oleh admin. Admin dapat memilih menu pemasok untuk melihat data pemasok, menambahkan pemasok baru dengan menyimpan data ke database, atau menghapus pemasok yang sudah ada. Untuk penghapusan, sistem meminta konfirmasi terlebih dahulu sebelum menghapus data pemasok dari database. Setelah proses selesai, data pemasok pada halaman akan diperbarui sesuai perubahan.

### A.3 Activity Diagram Manage Product

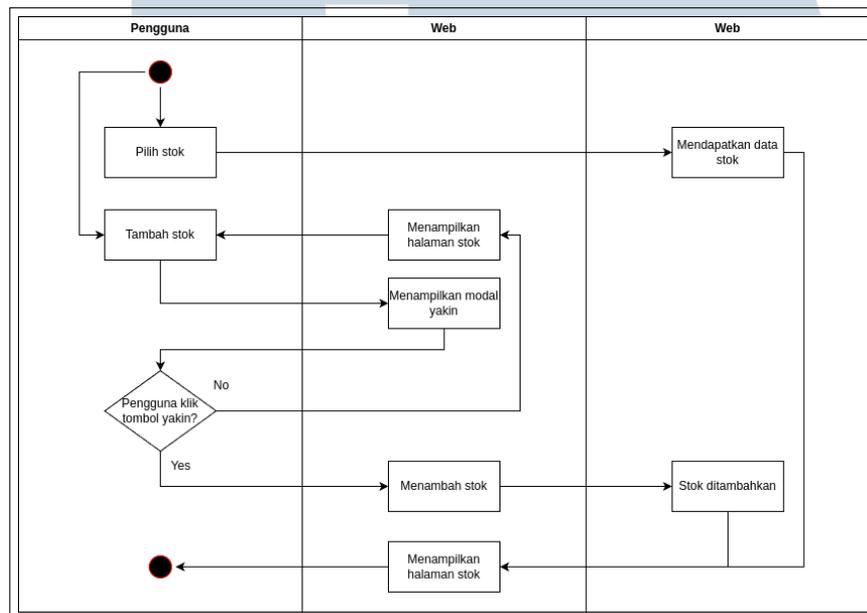


Gambar 3.4. Activity Diagram Pengelolaan Barang Admin

Gambar 3.4 menunjukkan activity diagram pengelolaan barang oleh admin. Admin dapat memilih barang untuk melihat data barang yang tersimpan di sistem,

menambahkan barang baru dengan menyimpan data ke dalam database, atau menghapus barang yang sudah ada. Untuk penghapusan barang, sistem akan meminta konfirmasi terlebih dahulu sebelum menghapus data dari database. Setelah proses selesai, data pada halaman barang akan diperbarui sesuai perubahan.

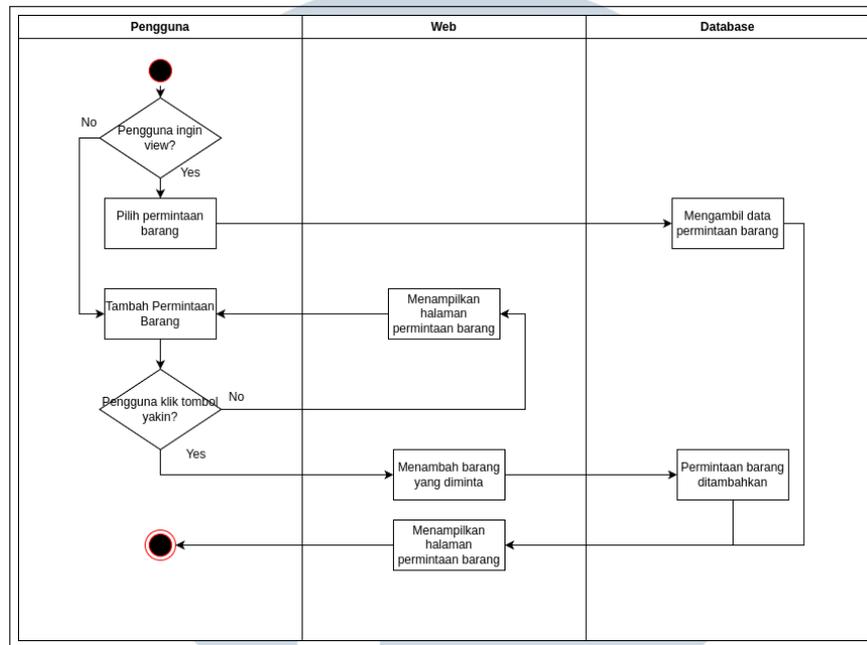
#### A.4 Activity Diagram Update Stock



Gambar 3.5. Activity Diagram Pengelolaan Stok Admin

Gambar 3.5 menunjukkan activity diagram pengelolaan stok oleh admin. Admin dapat memilih stok untuk melihat data yang tersimpan di sistem, menambahkan stok baru, atau memperbarui data stok yang ada. Setelah memilih stok, admin diarahkan ke halaman stok. Admin kemudian dapat menambah stok, di mana sistem akan meminta konfirmasi terlebih dahulu sebelum data stok diperbarui di database. Jika admin mengonfirmasi penambahan, sistem akan menyimpan perubahan, dan stok terbaru akan tercatat dalam database. Proses ini memastikan bahwa data stok yang tampil selalu sesuai dengan perubahan yang dilakukan.

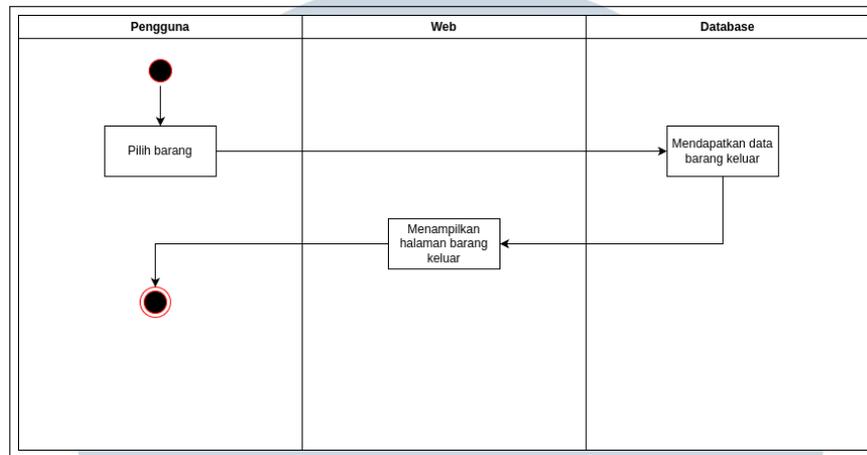
## A.5 Activity Diagram Manage Product Request



Gambar 3.6. Activity Diagram Pengelolaan Permintaan Barang oleh Admin

Gambar 3.6 menunjukkan activity diagram pengelolaan permintaan barang oleh admin. Proses dimulai dari admin yang memilih permintaan barang untuk dilihat. Sistem kemudian mengirimkan *request* ke database untuk mendapatkan data permintaan barang dan menampilkan halaman permintaan barang kepada admin. Setelah halaman permintaan barang ditampilkan, admin dapat memilih untuk menyetujui permintaan tersebut jika ada. Jika permintaan disetujui, sistem akan menambahkan barang yang diminta dengan konfirmasi terlebih dahulu. Pada langkah ini, admin diberikan opsi untuk membatalkan atau melanjutkan proses penambahan barang. Jika admin mengonfirmasi, data permintaan barang akan diperbarui di database, memastikan bahwa data yang tersimpan sudah sesuai dengan keputusan admin.

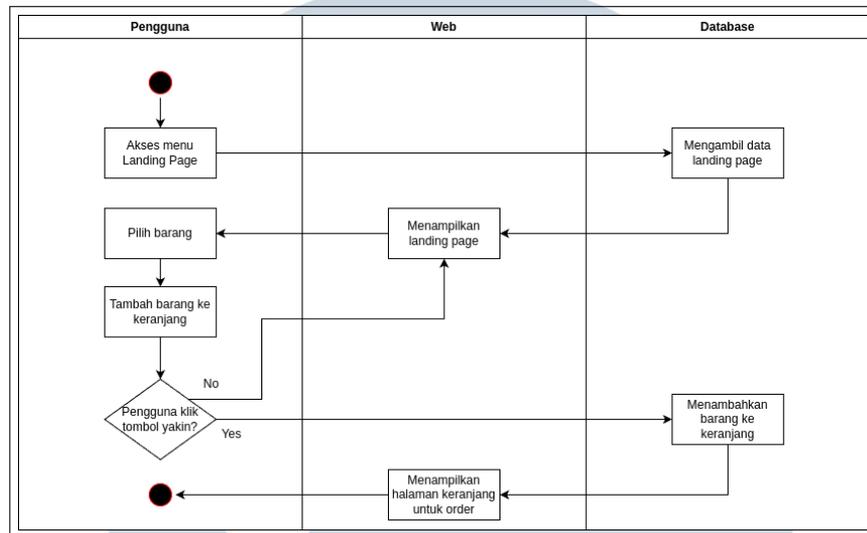
## A.6 Activity Diagram Add Demand Supplies



Gambar 3.7. Activity Diagram Pengelolaan Barang Keluar oleh Admin

Gambar 3.7 menunjukkan activity diagram untuk pengelolaan barang keluar oleh admin. Proses dimulai ketika admin memilih barang keluar yang ingin dikelola. Sistem kemudian mengirimkan *request* ke database untuk mengambil data yang relevan. Setelah data diterima dari database, sistem menampilkan halaman barang keluar kepada admin. Pada tahap ini, admin dapat melihat data yang telah diambil. Proses ini berakhir setelah halaman barang keluar berhasil ditampilkan, memastikan bahwa admin mendapatkan informasi yang diperlukan untuk pengelolaan barang keluar.

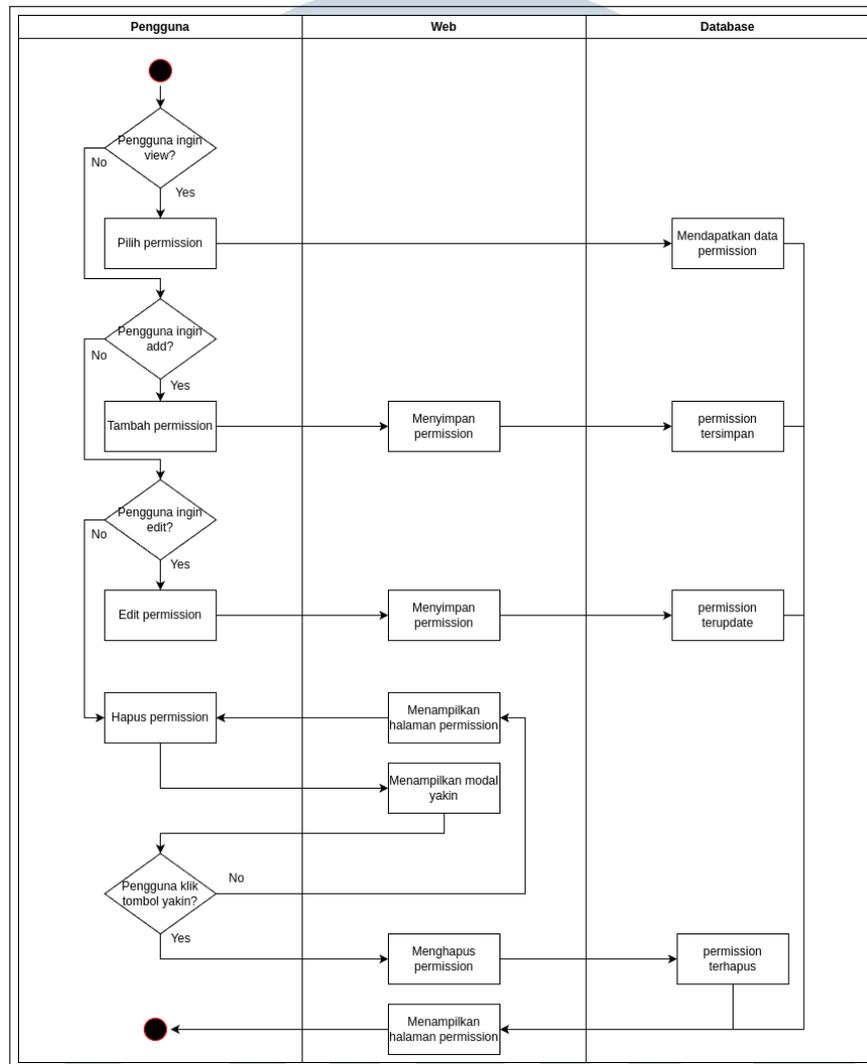
## A.7 Activity Diagram View Supplies Transaction



Gambar 3.8. Activity Diagram Pengelolaan Keranjang oleh Admin

Gambar 3.8 menunjukkan aktivitas pengelolaan keranjang oleh admin. Proses dimulai dengan admin memilih halaman *landing page*, kemudian sistem mengirimkan *request* ke database untuk eksekusi. Selanjutnya, sistem menampilkan halaman *landing page* di mana admin dapat menambahkan barang ke keranjang. Admin kemudian dapat memproses pesanan melalui halaman keranjang. Jika admin yakin, data pesanan disimpan dan ditampilkan di halaman barang keluar, dan stok terbaru diperbarui di database, mengakhiri proses ini.

## A.8 Activity Diagram Manage Permission

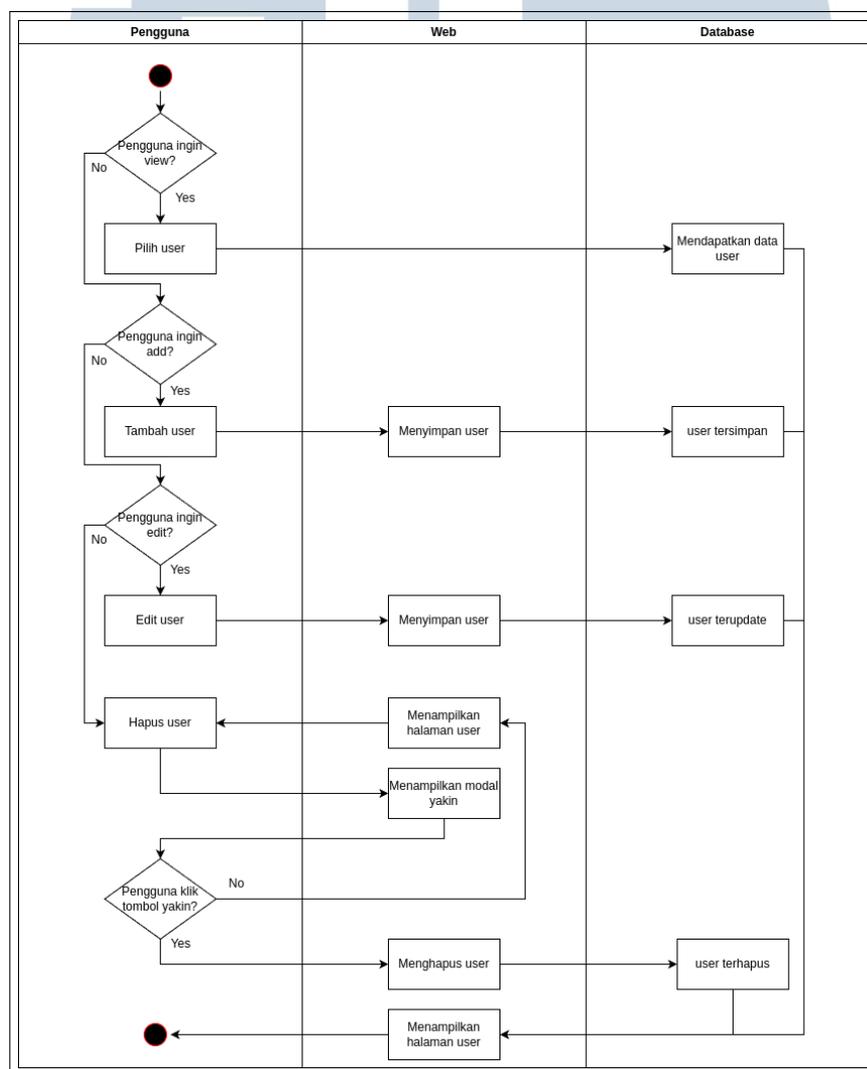


Gambar 3.9. Activity Diagram Pengelolaan Keranjang oleh Admin

Gambar 3.9 menunjukkan alur proses pengelolaan *permission* dalam sebuah sistem berbasis web yang melibatkan tiga entitas utama, yaitu pengguna, aplikasi web, dan database. Proses dimulai ketika pengguna memilih *permission*, yang kemudian memicu aplikasi web untuk mengambil data dari database dan menampilkan halaman *permission*. Jika pengguna menambahkan *permission*, data baru disimpan melalui interaksi antara aplikasi web dan database, dan halaman diperbarui untuk mencerminkan perubahan tersebut. Sebaliknya, jika pengguna ingin menghapus *permission*, sistem menampilkan *modal* konfirmasi untuk memastikan tindakan tersebut. Jika pengguna memilih untuk melanjutkan, aplikasi

web berkomunikasi dengan database untuk menghapus data yang dimaksud, lalu memperbarui halaman dengan perubahan terbaru. Alur ini berakhir setelah proses penambahan atau penghapusan selesai, yang menunjukkan bagaimana setiap langkah dikelola secara sistematis dengan kolaborasi antara pengguna, web, dan database untuk memastikan integritas data dalam sistem.

### A.9 Activity Diagram Manage User



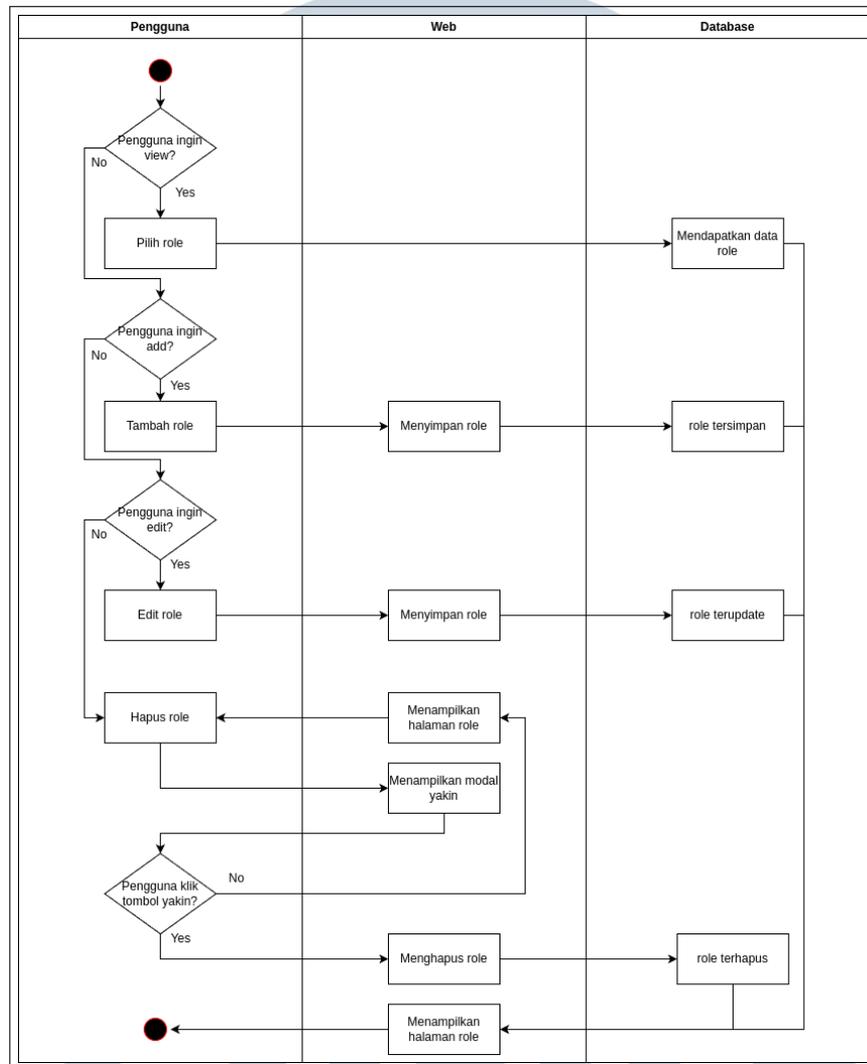
Gambar 3.10. Activity Diagram Pengelolaan Keranjang oleh Admin

Gambar 3.10 menggambarkan alur proses pengelolaan *user* dalam sebuah sistem berbasis web yang melibatkan tiga entitas utama, yaitu pengguna, aplikasi web, dan database. Proses dimulai ketika pengguna memilih *user*, yang memicu

aplikasi web untuk mengambil data terkait dari database dan menampilkan halaman *user*. Jika pengguna menambahkan *user*, aplikasi web akan menyimpan data tersebut melalui komunikasi dengan database. Setelah data *user* tersimpan, halaman diperbarui untuk mencerminkan perubahan tersebut. Sebaliknya, jika pengguna ingin menghapus *user*, sistem menampilkan sebuah *modal* konfirmasi untuk memastikan tindakan tersebut. Jika pengguna memilih untuk melanjutkan, aplikasi web akan memproses penghapusan dengan menghapus data *user* dari database dan memperbarui tampilan halaman. Proses ini berakhir setelah langkah penambahan atau penghapusan selesai, yang menunjukkan bagaimana sistem mengelola setiap langkah secara terstruktur melalui kolaborasi antara pengguna, aplikasi web, dan database untuk memastikan integritas data.



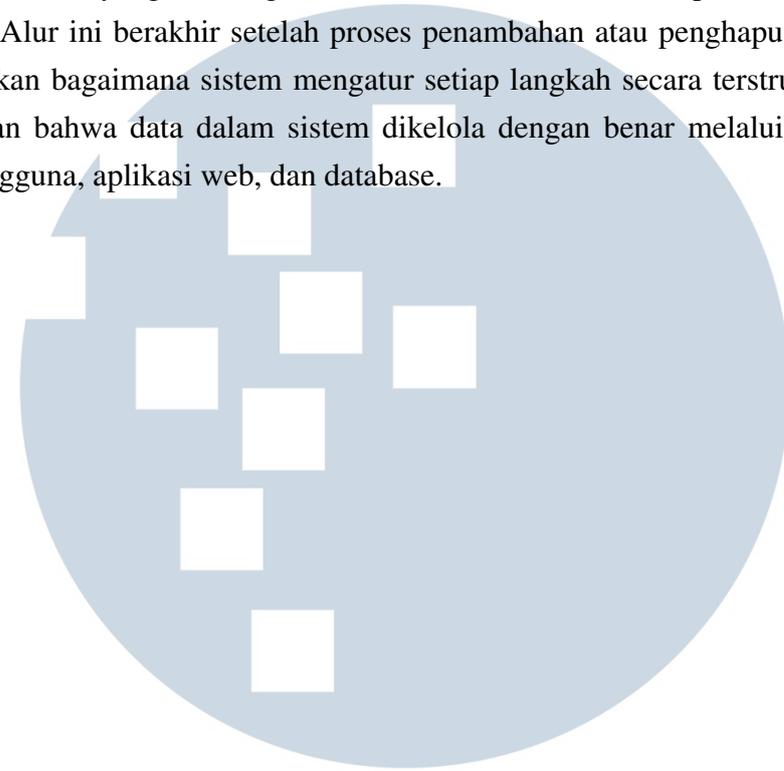
## A.10 Activity Diagram Manage Role



Gambar 3.11. Activity Diagram Pengelolaan Keranjang oleh Admin

Gambar 3.11 menunjukkan alur proses pengelolaan *role* dalam sebuah sistem berbasis web yang melibatkan tiga entitas utama, yaitu pengguna, aplikasi web, dan database. Proses diawali dengan pengguna memilih *role*, yang kemudian memicu aplikasi web untuk mengambil data terkait dari database dan menampilkan halaman *role*. Jika pengguna menambahkan *role*, sistem web akan menyimpan data tersebut melalui komunikasi dengan database, di mana data baru *role* disimpan, dan halaman diperbarui untuk mencerminkan penambahan tersebut. Sebaliknya, jika pengguna ingin menghapus *role*, sistem akan menampilkan sebuah *modal* konfirmasi untuk memastikan tindakan tersebut. Jika pengguna memilih untuk

melanjutkan penghapusan, aplikasi web akan memproses permintaan dengan menghapus data yang bersangkutan dari database dan memperbarui tampilan halaman. Alur ini berakhir setelah proses penambahan atau penghapusan selesai, menunjukkan bagaimana sistem mengatur setiap langkah secara terstruktur untuk memastikan bahwa data dalam sistem dikelola dengan benar melalui kolaborasi antara pengguna, aplikasi web, dan database.

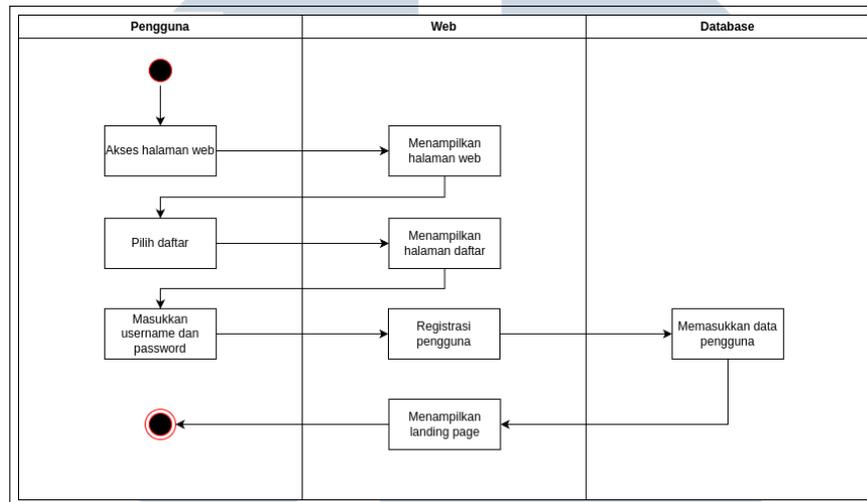


# UMMN

UNIVERSITAS  
MULTIMEDIA  
NUSANTARA

## B Activity Diagram User

### B.1 Activity Diagram Register User

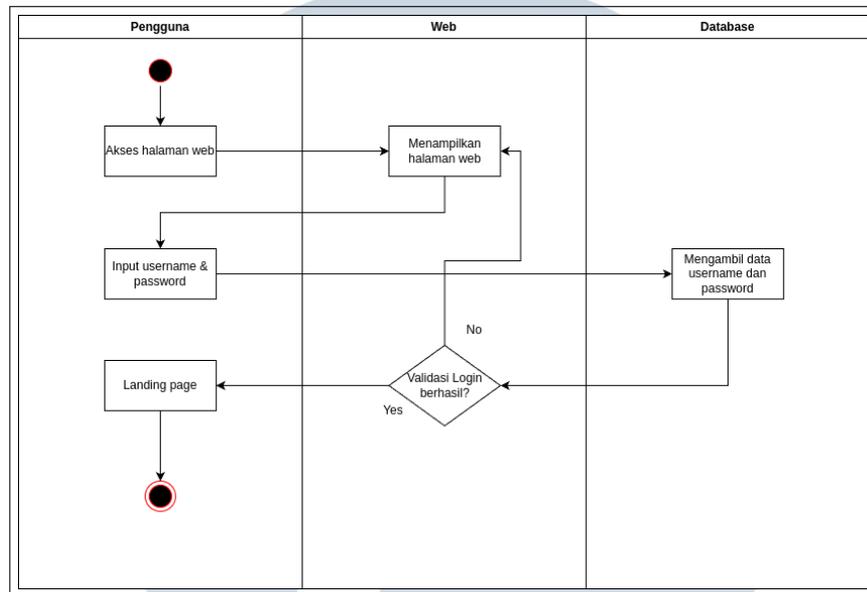


Gambar 3.12. Activity Diagram Proses Registrasi oleh User

Gambar 3.12 menunjukkan activity diagram untuk proses registrasi oleh user. Proses dimulai ketika user mengakses halaman web. Sistem kemudian menampilkan halaman web tersebut. Setelah itu, user memilih opsi daftar, dan sistem menampilkan halaman daftar. Selanjutnya, user memasukkan username dan password. Sistem menyimpan data tersebut ke dalam database. Setelah data berhasil disimpan, sistem menampilkan halaman *landing page*. Proses ini berakhir ketika user berhasil diarahkan ke halaman *landing page*.

U N I V E R S I T A S  
M U L T I M E D I A  
N U S A N T A R A

## B.2 Activity Diagram Login User

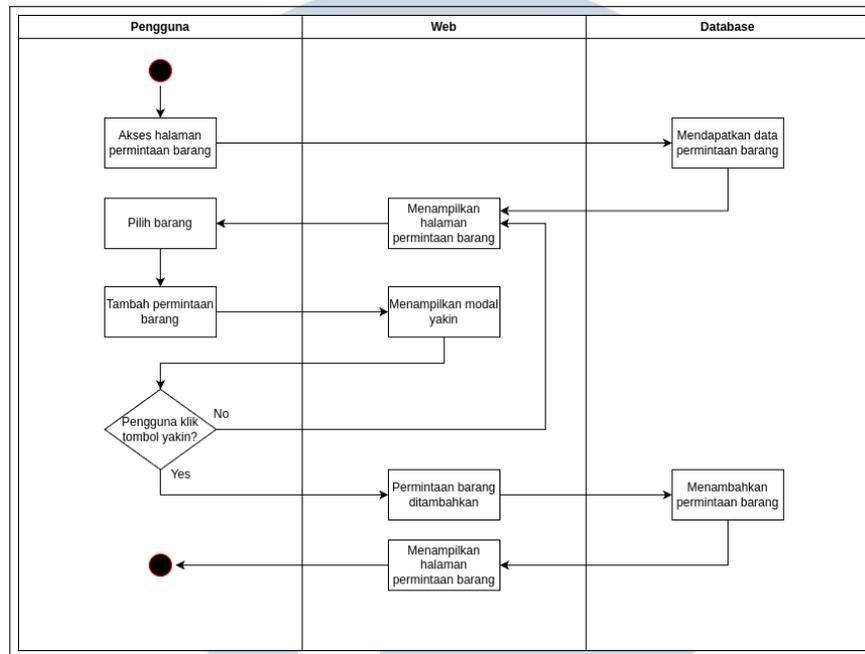


Gambar 3.13. Activity Diagram Proses Login oleh User

Gambar 3.13 menunjukkan activity diagram untuk proses login oleh user. Proses dimulai ketika user mengakses halaman web. Sistem kemudian menampilkan halaman web tersebut, memungkinkan user untuk menginput username dan password. Sistem mengirimkan permintaan validasi username dan password ke database. Jika data valid, user diarahkan ke halaman *landing page*. Jika tidak valid, user diminta untuk mengulang proses login. Proses berakhir ketika user berhasil masuk ke halaman *landing page*.

U N I V E R S I T A S  
M U L T I M E D I A  
N U S A N T A R A

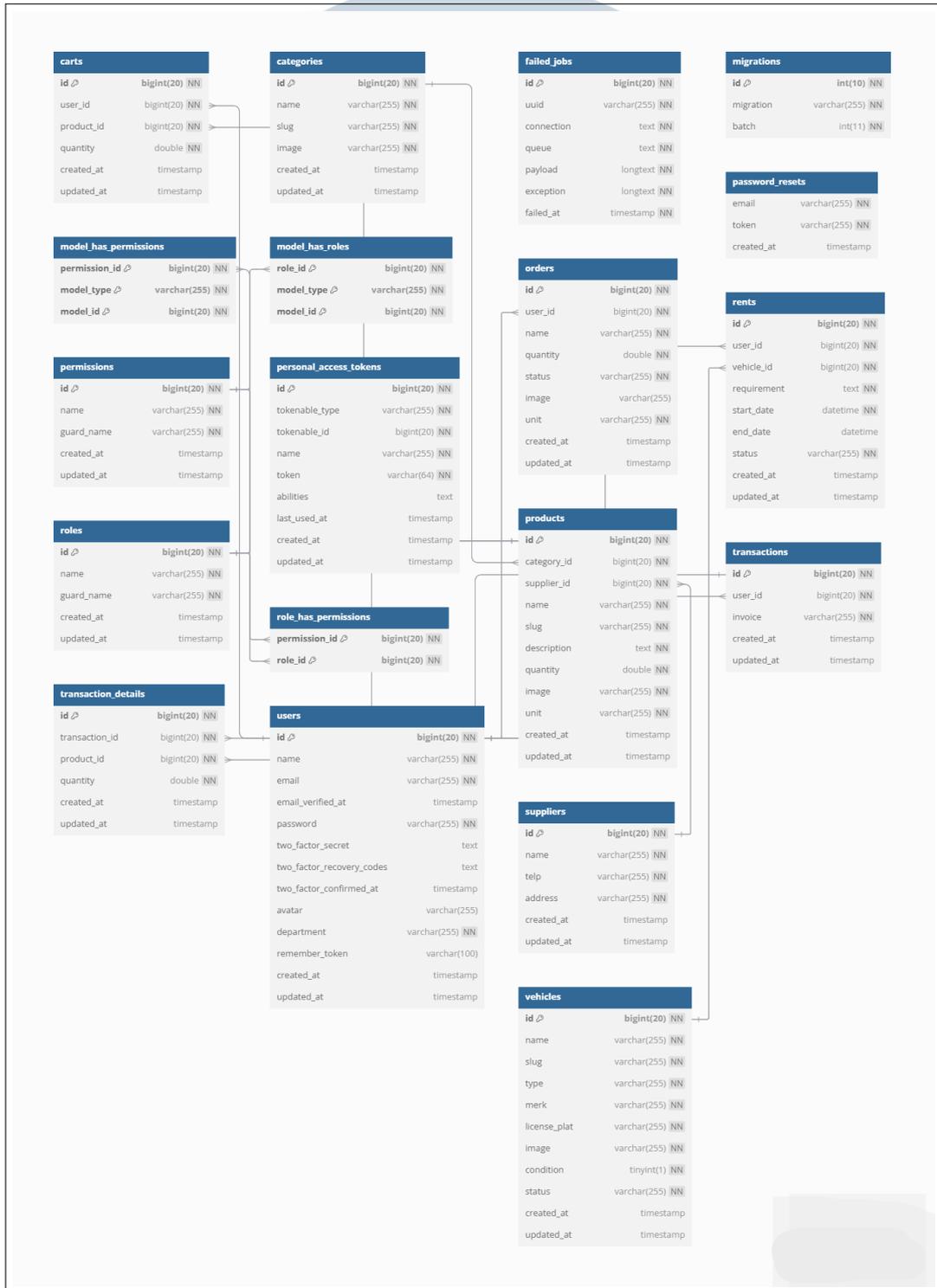
### B.3 Activity Diagram Add Demand Supplies User



Gambar 3.14. Activity Diagram Permintaan Barang User

Gambar yang ditunjukkan dalam 3.14 adalah sebuah Activity Diagram yang menggambarkan alur proses permintaan barang oleh pengguna. Diagram ini menunjukkan langkah-langkah yang dilalui oleh pengguna saat melakukan permintaan barang, dimulai dari pengguna mengakses sistem, mengisi formulir permintaan, hingga akhirnya permintaan barang diproses. Secara garis besar, diagram ini mencakup beberapa aktivitas penting, seperti memilih barang, mengonfirmasi permintaan, dan menunggu konfirmasi atau status pemrosesan. Setelah itu, sistem memverifikasi ketersediaan barang dan memberikan respons kepada pengguna apakah permintaan dapat dipenuhi atau tidak. Dengan menggunakan Activity Diagram ini, alur permintaan barang menjadi lebih jelas dan mudah dipahami, baik dari sisi pengguna maupun sistem yang memproses permintaan tersebut.

### 3.4.3 Entity Relationship Diagram (ERD)



Gambar 3.15. Entity Relationship Diagram

Diagram pada gambar 3.16 menggambarkan Entity Relationship Diagram (ERD) untuk database. Tabel `users` adalah tabel utama yang menyimpan informasi pengguna seperti nama, email, dan password. Tabel `products` berisi data produk yang terhubung dengan tabel `categories` melalui kolom `category_id`. Tabel `transactions` mencatat transaksi yang dilakukan oleh pengguna, dengan rincian transaksi disimpan di tabel `transaction_details`. Selain itu, tabel `suppliers` berisi data pemasok yang menyediakan produk, sedangkan tabel `vehicles` digunakan untuk manajemen data kendaraan yang terkait dengan tabel `rents` untuk penyewaan.

Beberapa tabel tambahan, seperti `model_has_roles` dan `model_has_permissions`, digunakan untuk mengelola hubungan antara model dengan peran dan izin, sedangkan `role_has_permissions` menghubungkan tabel `roles` dengan tabel `permissions`. Relasi penting meliputi hubungan antara `products.category_id` dengan `categories.id`, `transactions.user_id` dengan `users.id`, dan `carts.user_id` dengan `users.id`.

### 3.5 Pembuatan Program

Penelitian program dilakukan menggunakan framework Laravel untuk penelitian antarmuka pengguna (front-end) sekaligus back-end, serta MySQL sebagai sistem basis data. Metode yang diterapkan dalam penelitian program ini adalah metode Agile, yang dikenal fleksibel dan responsif terhadap perubahan kebutuhan pengguna. Dengan pendekatan ini, diharapkan sistem yang dibangun dapat lebih adaptif terhadap kebutuhan pengelolaan inventaris CV. Marannu Karya Sejahtera serta mendukung pencapaian tujuan bisnis melalui peningkatan efisiensi dan efektivitas operasional.

Metode *Agile* adalah pendekatan penelitian perangkat lunak yang menekankan fleksibilitas, kolaborasi, dan iterasi pendek untuk memastikan perangkat lunak yang dihasilkan sesuai dengan kebutuhan pengguna. Pendekatan ini sangat cocok untuk proyek yang dinamis, di mana kebutuhan dapat berubah berdasarkan umpan balik atau kondisi lapangan. Dalam penelitian ini, metode *Agile* diterapkan melalui siklus iterasi pendek (dikenal sebagai *sprint*). Berikut adalah tahapan penerapan metode *Agile*:

### 3.5.1 Inisiasi Proyek

Tahap ini adalah tahapan pertama dari metode Agile. Dimulai dengan mengidentifikasi kebutuhan pengguna melalui diskusi dengan pemangku kepentingan. Fitur utama yang dibutuhkan, seperti manajemen kategori, supplier, produk, stok, dan pengguna, diidentifikasi dan dimasukkan ke dalam *product backlog*. Berdasarkan wawancara yang telah dilakukan dengan pemilik usaha CV. Marannu Karya Sejahtera dalam Lampiran 1, diperoleh informasi penting mengenai permasalahan pengelolaan inventaris yang saat ini dihadapi oleh catering.

### 3.5.2 Perencanaan Sprint

Tahap ini adalah tahap kedua dari metode Agile. *Product backlog* dibagi menjadi tugas-tugas kecil yang dapat diselesaikan dalam iterasi pendek. Setiap sprint memiliki tujuan spesifik, seperti:

- Sprint 1: penelitian modul login dan register.
- Sprint 2: penelitian fitur kategori dan supplier.
- Sprint 3: penelitian modul produk dan stok produk.

### 3.5.3 Desain dan penelitian

Tahap ini adalah tahap ketiga dari metode Agile. Pada tahap ini, peneliti membuat desain antarmuka pengguna dan mengimplementasikan fungsionalitas berdasarkan backlog sprint. Beberapa kegiatan meliputi:

- Mendesain antarmuka menggunakan kerangka kerja modern (seperti *Bootstrap*).
- penelitian backend menggunakan kerangka kerja PHP (seperti *Laravel*).
- Membuat basis data untuk menyimpan data inventaris.

### 3.5.4 Pengujian

Tahap ini adalah tahap keempat dari metode Agile. Pada tahap ini, pengujian dilakukan untuk memastikan fungsionalitas yang dikembangkan sesuai kebutuhan. Pengujian dilakukan menggunakan metode *black-box testing* untuk menguji sistem dari sudut pandang pengguna tanpa memeriksa kode internal.

### 3.5.5 Umpan Balik dan Revisi

Tahap ini adalah tahap kelima dari metode Agile. Hasil sprint ditinjau oleh tim dan pengguna untuk mendapatkan umpan balik. Revisi dilakukan berdasarkan masukan sebelum sprint berikutnya dimulai.

### 3.5.6 Penyampaian

Tahap ini adalah tahap terakhir atau keenam dari metode Agile. Setelah beberapa iterasi, sistem yang telah memenuhi kebutuhan utama disampaikan kepada pengguna untuk digunakan. Dokumentasi disiapkan untuk mendukung pengguna dalam memahami sistem.

Dalam penelitian sistem ini, pendekatan *Agile* diterapkan dengan langkah-langkah sebagai berikut:

1. Mengidentifikasi kebutuhan pengguna melalui diskusi dengan pemangku kepentingan.
2. Membagi fitur utama menjadi backlog sprint, seperti login, kategori, produk, stok, dan pencarian.
3. Melaksanakan sprint iteratif untuk mengembangkan setiap modul secara bertahap.
4. Menggunakan umpan balik pengguna pada akhir setiap sprint untuk menyesuaikan fitur sesuai kebutuhan.
5. Menyelesaikan pengujian dan penyampaian sistem setelah beberapa iterasi.

Beberapa alat yang digunakan untuk mendukung penerapan *Agile* dalam penelitian ini adalah:

- **Trello/Jira:** Untuk manajemen backlog dan sprint.
- **GitHub:** Untuk manajemen versi kode sumber.
- **Postman:** Untuk pengujian API.
- **Google Meet/Slack:** Untuk komunikasi dengan tim dan pengguna.