

BAB I

PENDAHULUAN

1.1 Latar Belakang

Logistik ialah disiplin ilmu yang bisa mengendalikan serta mengatur arus benda, tenaga, data serta sumber energi yang lain (semacam jasa serta sumber daya manusia). Disamping itu logistik juga mencakup integrasi operasi transportasi, inventori, informasi serta pergudangan. Tanpa dukungan logistik akan sulit mengoptimalkan penggunaan modal yang dikeluarkan. Jika dikaitkan dengan *supply chain*, terdapat banyaknya pihak yang terlibat dalam menjalankan proses bisnis suatu perusahaan.

Dalam mata rantai operasional logistik dan rantai pasok terdapat fasilitas gudang (*warehouse*) yang memiliki peranan penting sebagai fasilitas untuk menunjang operasional bisnis suatu perusahaan. Gudang sendiri merupakan tempat untuk menyimpan barang-barang untuk produksi (*raw material atau work-in process*), bahan habis pakai, maupun hasil produksi dalam kuantitas dan periode waktu tertentu yang kemudian akan didistribusikan ke tujuan tertentu seperti distribution center, retail, maupun end customer. Sebagai fasilitas yang memiliki peranan vital bagi keberlangsungan bisnis perusahaan, maka operasional di gudang harus menerapkan manajemen yang efektif dan efisien dalam pengontrolan kegiatan pergudangan sehingga diharapkan terjadinya pengurangan biaya-biaya operasional yang ada didalam gudang, aliran masuk dan keluar barang yang lancar, serta kemudahan akses dan keakuratan informasi stok barang digudang. Salah satu tantangan yang dihadapi dalam operasional pergudangan ialah banyaknya jenis barang yang ditangani serta pergerakan barang yang cepat sehingga dapat menimbulkan kesalahan baik dari sisi penanganan fisik barang maupun perekaman status informasi barang.

Dengan adanya tantangan tersebut, pengelolaan terhadap barang dan informasi membutuhkan sistem yang bisa mengatasi tantangan tersebut serta mampu mengatur semua aktivitas di gudang. Sistem yang dibangun diharapkan bisa menampilkan kondisi stok barang secara langsung dan sesuai antara jumlah yang ada pada sistem dan yang ada di gudang sehingga bisa memberikan informasi yang jelas untuk pengguna. Perancangan sistem informasi tersebut, diharapkan juga mampu meningkatkan produktivitas dan pelayanan gudang, sehingga dapat mempengaruhi proses bisnis perusahaan/industri secara keseluruhan. PT Pertamina Lubricants sudah membuat sistem terkait tantangan yang ada

pada operasional pergudangan hanya saja sistem tersebut masih masih mempunyai kekurangan dan belum sempurna. Maka dari itu dilakukan penelitian terhadap perancangan sistem informasi pada PT Pertamina Lubricants serta pengaplikasiannya untuk mendukung proses bisnis dari PT Pertamina Lubricants.

1.2 Rumusan Masalah

Berdasarkan uraian identifikasi masalah diatas, maka rumusan masalah dalam penelitian ini adalah:

1. Bagaimana menghasilkan sistem yang dapat memberikan keakuratan informasi secara real time, menampilkan grafik stok serta terakomodirnya pemindahan material dalam pengelolaan logistik PT. Pertamina Lubricants?
2. Bagaimana menghasilkan sistem yang dapat memenuhi informasi mengenai logistik PT. Pertamina Lubricants?

1.3 Batasan Masalah

Agar tidak menyimpang dari permasalahan yang diangkat dan bisa menggapai target yang diharapkan maka penelitian ini membatasi permasalahan sebagai berikut:

1. Menggunakan bahasa PHP.
2. Membangun aplikasi Web.
3. Informasi yang disajikan hanya bagian logistik PT. Pertamina Lubricants.
4. Tidak membahas keamanan jaringan.

1.4 Tujuan Perancangan

Tujuan penulisan yang hendak dicapai dalam tugas akhir ini yaitu membangun aplikasi sistem informasi logistik secara *online* dengan memberikan kemudahan berupa meminimasi *human error* pada penggunaan aplikasi sistem informasi logistik serta dapat membantu para pekerja PT. Pertamina Lubricants.

1.5 Manfaat Perancangan

Manfaat yang bisa diperoleh dari pembuatan sistem ini adalah:

1. Memudahkan para operator gudang dalam mengelola stok barang dan informasi logistik PT. Pertamina Lubricants.
2. Dengan bisa melihat kondisi stok secara langsung bisa meningkatkan efektifitas proses bisnis PT. Pertamina Lubricants.
3. Memberikan informasi yang dapat diakses oleh para pengguna kapanpun dan dimanapun.

1.6. Waktu Pelaksanaan Perancangan

Waktu pelaksanaan perancangan dalam tugas akhir ini dimulai pada bulan September 2020 – Januari 2021.

Task	2020				2021
	September	Oktober	November	Desember	Januari
<i>Planning</i>					
Melakukan wawancara					
Analisis kebutuhan sistem					
<i>Design</i>					
Pembuatan UML					
Pembuatan ERD					
Pembuatan Class Diagram					
Pembuatan <i>Mockup</i>					
<i>Coding</i>					
Pembuatan aplikasi Sistem Informasi Logistik					
Pengembangan aplikasi Sistem Informasi Logistik					
<i>Testing</i>					
Pengujian atau <i>testing</i>					

BAB II

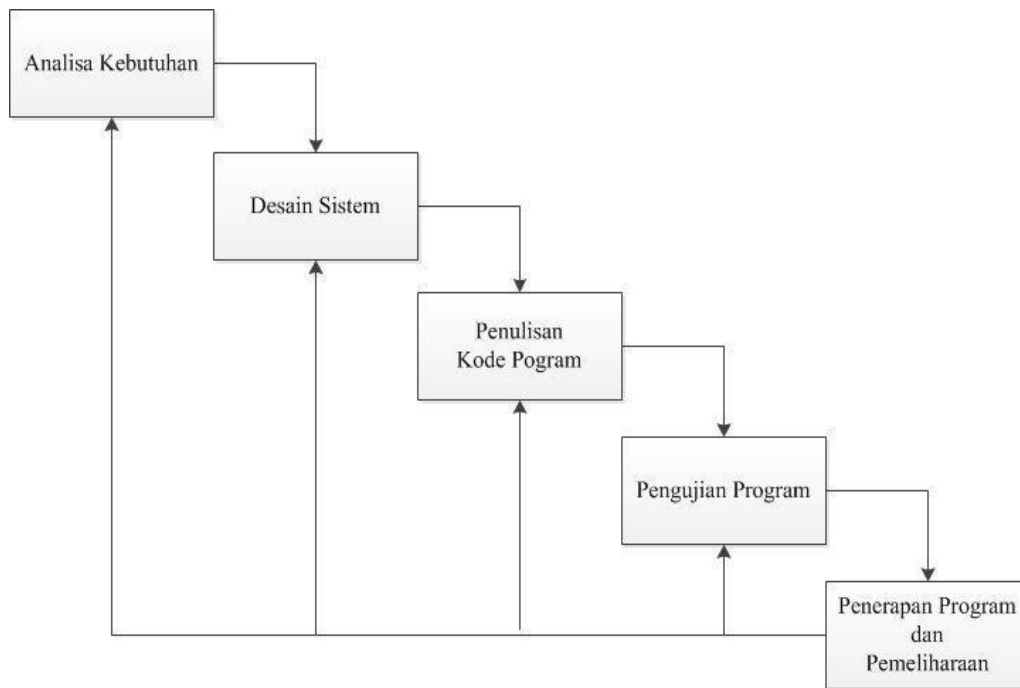
TINJAUAN PUSTAKA

Pada penelitian ini sistem yang terbuat berbasis website dan menggunakan bahasa pemrograman PHP. Pasalnya, bahasa PHP memiliki kemampuan lintas platform, yang artinya PHP dapat dengan mudah diterapkan pada berbagai platform sistem operasi, dan hampir semua browser juga mendukung PHP, dan merupakan kode *open source free* atau gratis. PHP memiliki tingkat akses yang cepat dan didukung oleh beberapa web server. PHP mendukung beberapa web server, seperti Apache, IIS, Lighttpd, Xitami.

2.1 Metode *Waterfall*

Metode yang digunakan dalam penelitian ini adalah metode *waterfall* karena kelebihanannya. Keuntungan yang ada pada penggunaan metode *waterfall* ialah prosesnya yang lebih terstruktur, yang membuat perangkat lunak menjadi berkualitas dan dapat dipelihara. *User* juga lebih diuntungkan karena bisa merencanakan dan menyiapkan data yang dibutuhkan untuk kebutuhan proses dari awal. Penjadwalan juga menjadi lebih deterministik dikarenakan penjadwalan masing-masing proses bisa ditentukan sehingga tujuan penyelesaian pengembangan program dapat terlihat dengan jelas. Diberikan rangkaian tertentu untuk melihat kemajuan yang tepat dari setiap tahapan. Model tersebut merupakan model tipe dokumen yang utuh, sehingga proses perawatan bisa dilaksanakan dengan mudah.

Dalam penelitian ini metode *waterfall* merupakan metode yang pada umumnya digunakan oleh analis sistem. Pokok inti dari metode *waterfall* ialah pekerjaan sistem dilakukan secara berurutan atau linier. Oleh karena itu, jika langkah pertama belum diselesaikan, maka langkah 2, 3, dan seterusnya tidak akan dapat dilakukan jika fase pertama dan kedua sudah dijalankan, fase ketiga akan dijalankan secara otomatis.



Gambar 2.2 Langkah-langkah metode *Waterfall*

Secara garis besar metode *waterfall* mempunyai proses-proses sebagai berikut : Analisa Kebutuhan, Desain Sistem, Kode Program, Pengujian Program dan Persiapan Program dan Pemeliharaan (Pressman, 2015).

1. Analisa Kebutuhan

Proses ini merupakan analisis kebutuhan sistem. Pengumpulan data pada tahap ini dapat dilakukan melalui penelitian, wawancara atau penelitian kepustakaan. Analisis sistem akan mengumpulkan informasi sebanyak mungkin dari pengguna sehingga sistem komputer dapat dibuat untuk melakukan tugas-tugas yang diinginkan pengguna. Tahapan ini akan menciptakan dokumen *user requirement* yang berkaitan dengan keinginan *user* terhadap pengerjaan sistem. Dokumen ini tentunya akan menjadi referensi bagi analisis sistem untuk mengkonversi ke bahasa pemrograman.

2. Desain Sistem

Proses desain mengubah *user requirement* menjadi desain perangkat lunak yang dapat diperkirakan sebelum proses *coding*. Proses tersebut berfokus pada: struktur data, arsitektur perangkat lunak, representasi antarmuka, dan detail proses (algoritma). Pada tahap ini, dokumen yang disebut kebutuhan perangkat lunak akan dibuat. Programmer akan menggunakan dokumen ini untuk kegiatan pembuatan sistem.

3. Kode Program

Coding merupakan terjemahan desain dalam bahasa yang dapat dikenali oleh komputer. Dijalankan oleh *programmer*, kemudian menerjemahkan transaksi yang diinginkan oleh *user*. Tahap ini merupakan tahap pengerjaan sistem yang sebenarnya. Artinya, komputer akan digunakan secara maksimal pada tahap ini. Setelah pengodean selesai maka akan dilakukan pengujian terhadap sistem yang telah diselesaikan sebelumnya. Tujuan *testing* ialah untuk menemukan kesalahan dalam sistem dan kemudian memperbaikinya.

4. Pengujian Program

Tahapan *testing* merupakan tahapan terakhir dari pembuatan suatu sistem. Setelah dilakukan analisis, perancangan dan pengodean, sistem yang telah selesai akan digunakan oleh *user* kemudian *user* akan menguji secara langsung sistem yang telah selesai dan memberikan *feedback* terhadap sistem tersebut.

5. Penerapan Program dan Pemeliharaan

Perangkat lunak yang dikirimkan ke *user* pasti akan berubah. Perubahan tersebut mungkin disebabkan oleh kesalahan, dikarenakan perangkat lunak harus beradaptasi dengan lingkungan (periferal atau sistem operasi baru), atau *user* membutuhkan pengembangan fitur.

2.1.1 Keuntungan Metode *Waterfall*

1. Kualitas sistem yang dihasilkan akan sangat baik. Hal ini dikarenakan pada implementasi sistemnya yang dilakukan secara bertahap sehingga metode *waterfall* tidak hanya fokus pada tahapan tertentu.
2. Dokumen pengembangan sistem sangat terorganisir, karena setiap tahapan harus diselesaikan secara utuh sebelum memasuki tahapan berikutnya dan setiap tahapan akan memiliki dokumen tertentu.

2.1.2 Kelemahan Metode *Waterfall*

1. Manajemen yang baik diperlukan karena proses pengembangan tidak dapat diulangi sebelum produk muncul.
2. Jika kesalahan tidak ditemukan lebih awal, maka kesalahan kecil akan menjadi masalah besar.
3. Pelanggan merasa kesulitan untuk menyatakan kebutuhannya dengan jelas, sehingga tidak dapat beradaptasi dengan ketidakpastian di awal pengembangan.

2.2 Pengertian Logistik

Logistik ialah proses yang secara strategis mengatur pengadaan, perpindahan dan penyimpanan material, suku cadang dan produk jadi, dan aliran informasi terkait melalui organisasi dan saluran pemasarannya dengan tujuan meningkatkan keuntungan perusahaan (Christopher, 2015). Logistik juga mencakup integrasi operasi transportasi, inventori, informasi, pergudangan hingga *reverse logistics*. Tanpa dukungan logistik akan sulit mengoptimalkan penggunaan modal yang dikeluarkan. Jika dikaitkan dengan *supply chain*, terdapat banyaknya pihak yang terlibat dalam menjalankan proses bisnis suatu perusahaan.

Logistik ialah salah satu dari bagian *supply chain* yang terlibat dalam kegiatan distribusi dan transportasi, yaitu mulai dari perencanaan hingga penentuan keputusan terkait pengiriman barang. Kemudian peranan terhadap sistem inventori, hal tersebut memiliki hubungan dengan distribusi dan transportasi, karena hampir seluruh industri memproduksi barang di tempat yang berbeda dengan barang tersebut dikonsumsi. Sehingga timbul adanya keterkaitan antar sistem inventory dan sistem distribusi. Keberadaan gudang dianggap sebagai hal yang sangat penting, karena gudang merupakan fasilitas yang selalu dilalui selama proses pemenuhan kebutuhan konsumen dilakukan.

2.3 Website

Website atau situs ialah kumpulan halaman yang isinya menampilkan informasi tentang data teks, data gambar statis atau dinamis, data animasi, suara, video, dan kombinasinya (statis dan dinamis), yang merupakan rangkaian dari bangunan yang saling terkait, setiap link dari setiap bangunan dihubungkan oleh suatu jaringan halaman web atau *hyperlink*.

Jika isi konten informasi website konstan, jarang berubah, dan konten informasi searah dengan pemilik website, berarti statis. Isi informasi website selalu berubah-ubah, dan isi informasinya merupakan interaksi dua arah antara pemilik dengan pengguna website yang bersifat dinamis. Pengembangan situs web statis hanya dapat diperbarui oleh pemiliknya, sedangkan situs web dinamis hanya dapat diperbarui oleh pengguna dan pemiliknya (Rosa & M, 2018).

2.4 PHP (*Hipertext Preprocessor*)

PHP ialah singkatan dari "*Hypertext Preprocessor*", yang merupakan sebuah bahasa pemrograman yang berjalan di sisi server (*server side scripting*). PHP tidak dapat berjalan sendiri dan membutuhkan server website agar dapat menjalankannya.

PHP terintegrasi dengan kode HTML untuk membuat situs web dinamis. Kode HTML digunakan untuk membuat bingkai atau membangun situs web, sedangkan kode PHP digunakan untuk membuatnya menjadi dinamis. Makna dinamis bisa berupa perhitungan logis, menambah data, menampilkan hingga menghapus data.

PHP adalah perangkat lunak *open source* (gratis) dengan fungsi lintas *platform* dan dapat digunakan dengan semua jenis sistem operasi dan server web. PHP dapat berjalan di Windows dan beberapa versi Linux. PHP dapat dioperasikan dengan dua cara yakni sebagai modul apache pada web server dan sebagai *file* biner pada CGI.

”PHP dikenal sebagai bahasa pemrograman *script-script* yang membuat dokumen HTML secara *on the fly* yang dieksekusi di server web, dokumen HTML, yang dihasilkan dari suatu aplikasi bukan dokumen HTML, yang dibuat dengan menggunakan editor teks atau editor HTML, dikenal sebagai bahasa pemrograman *server side*“ (Sidik, 2018).

2.5 Perangkat Lunak XAMPP

XAMPP ialah alat untuk mengemas paket perangkat lunak menjadi paket perangkat lunak. Jika telah dilakukan instalasi XAMPP, tidak perlu menginstal dan mengkonfigurasi server web Apache, PHP, dan MySQL secara manual. XAMPP akan menginstal dan mengkonfigurasinya secara otomatis (Rosa & M, 2018).

2.5.1 APACHE

Apache adalah *tools* yang bersifat *open source*, yang berarti semua orang dapat menggunakannya, mengambil, dan bahkan mengubah kode program. Tugas utama Apache adalah membuat halaman web yang benar untuk pengguna berdasarkan kode PHP yang ditulis oleh pembuat halaman web. *Database* dapat diakses terlebih dahulu jika penulisan kode berdasarkan kode PHP diperlukan (misalnya di MySQL) untuk mendukung halaman web yang dibuat (Rosa & M, 2018).

2.5.2 PHP dan MySQL

Sistem manajemen *database* yang sering digunakan PHP adalah MySQL. Tetapi PHP juga mendukung sistem manajemen *database* Oracle, Microsoft Access, Interbase, d-Base, PostgreSQL, dan lain-lain. PHP sepenuhnya mendukung pemrograman berorientasi objek, integrasi XML, mendukung semua ekstensi dari MySQL terbaru, pengembangan layanan Web menggunakan SOAP dan REST, dan ratusan fungsi. Seperti server web lainnya, PHP juga bersifat *open source*, yang berarti gratis sehingga semua orang dapat menggunakannya (Rosa & M, 2018).

Perkembangannya disebut SQL, yang merupakan singkatan dari *Structured Query Language*. SQL ialah bahasa terstruktur yang didedikasikan untuk memproses *database*. SQL awalnya didefinisikan pada tahun 1986 oleh *American National Standards Institute* (ANSI). MySQL ialah sistem manajemen *database open source*. MySQL sangat cocok untuk PHP. MySQL dibuat dan dikembangkan oleh MySQL AB di Swedia.

MySQL dapat dipakai untuk membuat serta mengelola *database* beserta isinya. MySQL dapat digunakan juga untuk menambah, memodifikasi dan menghapus data di *database*. MySQL adalah sistem manajemen basis data relasional. Artinya data yang dikelola dalam *database* akan ditempatkan pada beberapa tabel terpisah, sehingga pengolahan data akan lebih cepat.

MySQL dapat dipakai untuk mengelola *database* kecil hingga besar. MySQL juga dapat menjalankan perintah SQL untuk mengelola *database*-nya. MySQL 5 ini sudah menunjang pemicu, yang dapat dengan mudah mengelola tabel di *database*.

2.5.3 PHPMYAdmin

Menggunakan MySQL untuk manajemen *database* harus dilaksanakan dengan mengetikkan *command line* (baris perintah) yang sesuai untuk setiap tujuan tertentu. Jika ingin membuat *database*, ketikkan *command line* yang sesuai untuk membuat *database*. Jika tabel ingin dihapus, maka ketik saja *command line* yang sesuai untuk menghapus tabel. Hal ini pasti sangat sulit karena harus mengingat dan mengetik perintah satu persatu.

Ada banyak *software* yang dapat digunakan untuk mengelola *database* di MySQL, salah satunya ialah phpMyAdmin. Menggunakan phpMyAdmin, ini akan memberikan kemudahan dalam pembuatan tabel, mengisi data, dan lain-lain. Tanpa harus mengingat perintah. PhpMyAdmin dapat diaktifkan dengan langkah pertama yaitu mengaktifkan *web server* Apache dan MySQL dari *control panel* XAMPP setelah menginstal XAMPP. Langkah kedua, memulai browser (IE, Mozilla Firefox atau Opera), dan masukkan URL berikut: <http://localhost/phpmyadmin/> di *address bar*, dan tekan *Enter*. Langkah ketiga ialah saat muncul *interface* phpMyAdmin, bisa dimulai dengan memasukkan nama *database*, nama tabel, dan lain-lain (Rosa & M, 2018).

2.6 Unified Modeling Language (UML)

Unified Modeling Language (UML) digunakan untuk memodelkan sistem/perangkat lunak menggunakan alat yang ada. Pemodelan UML dalam penggunaannya, akan

berfokus pada pengembangan dan desain perangkat lunak untuk menyelesaikan rekayasa dan pengembangan perangkat lunak terhadap (Nugroho, 2017) :

1. Gambaran umum arsitektur sistem secara keseluruhan.
2. Memahami bagaimana objek dalam sistem mengirim pesan dan berkolaborasi antara satu sama lainnya.
3. Uji apakah sistem/perangkat lunak beroperasi secara normal.
4. Dokumentasi sistem/perangkat lunak untuk tujuan khusus di masa mendatang.

Setiap sistem yang kompleks harus dilihat dari sudut pandang yang berbeda sehingga dapat dipahami secara menyeluruh. Untuk tujuan ini, UML menyediakan sembilan jenis diagram, yang dapat dikelompokkan menurut sifat statis atau dinamisnya. Jenis diagram UML diantaranya adalah :

1. *Use Case Diagram*

Statis, menampilkan kumpulan *use case* dan *actor*. Diagram ini sangat penting, terutama saat memodelkan atau mengatur perilaku sistem yang dibutuhkan oleh *user*. *Use case* diagram memodelkan perilaku sistem informasi yang akan dibuat. *Use case* digunakan untuk menentukan fungsi dalam sistem informasi dan siapa yang berhak menggunakan fungsi tersebut.

2. *Class Diagram*

Statis, tetapi biasanya berisi kelas-kelas aktif dan menampilkan sekumpulan kelas, antarmuka, kolaborasi, dan hubungan kelasnya. Diagram kelas menggambarkan struktur sistem dengan menentukan kelas yang dibutuhkan untuk membangun sistem. Kelas memiliki bagian yang disebut sebagai atribut, metode, atau operasi.

3. *Statechart Diagram*

Dinamis, menunjukkan *state* dari sistem, termasuk *state*, transisi, *event*, dan aktivitas. Penting untuk menunjukkan sifat dinamis dari *interface* (antarmuka), kelas, dan kolaborasi, yang sangat penting dalam pemodelan sistem yang reaktif. Menjelaskan transisi dan perubahan keadaan (dari satu keadaan ke keadaan lain) dari objek dalam sistem karena rangsangan yang diterima. Umumnya, *statechart* menggambarkan kelas tertentu (kelas dapat memiliki beberapa *statechart*).

4. *Activity Diagram*

Sifatnya dinamis. *Activity* diagram ialah suatu diagram keadaan khusus yang menunjukkan aliran dari satu aktivitas ke aktivitas lainnya dalam sebuah sistem. *Activity* diagram menggambarkan *workflow* (alur kerja) atau aktivitas sistem atau proses bisnis atau menu pada perangkat lunak. Perlu diperhatikan bahwa *activity*

diagram menggambarkan aktivitas sistem, bukan perilaku *actor*, sehingga sistem dapat melakukan aktivitasnya.

5. *Sequence Diagram*

Sifatnya dinamis dan menekankan pengiriman *message* (pesan) dalam jangka waktu tertentu. *Sequence* diagram menjelaskan mengenai perilaku suatu objek dalam *use case* dengan memvisualisasikan *lifeline* objek dan *message* yang akan dikirim dan diterima di antara objek tersebut. Mengilustrasikan *sequence* diagram, perlu memahami objek yang terlibat dalam *use case* dan model yang dimiliki kelas yang dipakai untuk objek tersebut. Desain *sequence* diagram juga dapat bertujuan untuk melihat skenario yang ada dalam *use case*.

6. *Collaboration Diagram*

Bersifat dinamis dan berfokus pada organisasi struktural objek yang menerima dan mengirim pesan. Menjelaskan interaksi antar objek, seperti diagram *sequence*, tetapi peran setiap objek perlu ditekankan, bukan waktu pengiriman pesan. Setiap pesan memiliki nomor urut, di mana pesan tingkat tertinggi memiliki nomor 1. Pesan dengan level yang sama memiliki awalan yang sama.

7. *Component Diagram*

Bersifat statis. Diagram ini terkait dengan diagram kelas, di mana komponen biasanya dipetakan ke satu atau lebih dari satu kelas, antarmuka, atau kolaborasi. *Component* diagram ialah diagram yang menggambarkan struktur fisik sistem, digunakan untuk menjelaskan bagaimana cara membagi kode program menjadi beberapa komponen, dan untuk menggambarkan hubungan antar komponen. *Component* diagram memberikan ilustrasi umum sistem tingkat tinggi yang sederhana. Klasifikasi sekumpulan kelas menjadi beberapa komponen untuk memfasilitasi pertukaran dan penggunaan kembali kode. Tujuan dari *component* diagram ialah untuk memperlihatkan hubungan struktural antar komponen dalam sistem.

8. *Deployment Diagram*

Bersifat statis, diagram ini menunjukkan konfigurasi *run-time* aplikasi. Diagram ini sangat berguna ketika aplikasi bertindak sebagai aplikasi yang dapat berjalan di banyak komputer (*distributed computing*). *Deployment* diagram adalah diagram yang digunakan untuk membagi perangkat lunak menjadi node pemrosesan (*processing node*). Diagram ini juga digunakan untuk menggambarkan pengaturan elemen pemrosesan dari *run-time* perangkat lunak.

2.7 Framework

Framework dapat didefinisikan sebagai kumpulan fungsi/prosedur dan *class* yang dapat digunakan untuk tujuan tertentu, sehingga dapat menyederhanakan dan mempercepat pekerjaan seorang programmer tanpa harus membuat lagi fungsi atau *class* dari awal (Hakim, 2010:8).

Ada beberapa alasan untuk menggunakan *framework* yaitu:

1. Mempercepat dan mempermudah pembuatan atau pengembangan sebuah aplikasi web.
2. Proses pemeliharaan relatif mudah, karena pola tertentu sudah ada di *framework* (dengan syarat pola standar yang sudah ada diikuti oleh programmer).
3. Secara umum, *framework* menyediakan fungsi yang umum digunakan, jadi tidak perlu membangun lagi dari awal.

Lebih bebas dalam pengembangan jika dibandingkan dengan CMS (*Content Management System*).

2.7.1 CodeIgniter

CodeIgniter merupakan *framework* PHP yang dapat membantu developer menjadi lebih cepat untuk mengembangkan aplikasi web berbasis PHP (Hakim, 2010:8).



Gambar 2.2 Logo CodeIgniter

Sumber : Hakim “Membangun Web Berbasis *PHP* dengan Framework CodeIgniter” (2010:9)

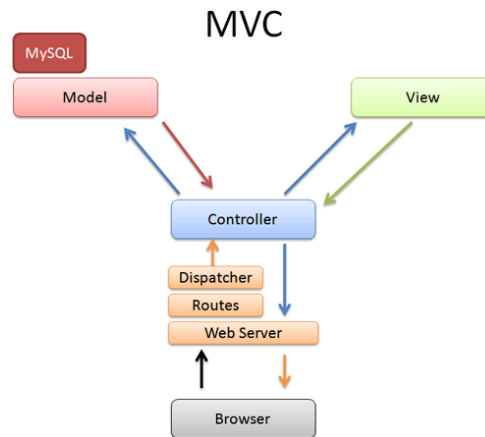
CodeIgniter merupakan *framework* PHP yang dapat mengimplementasikan sistem berbasis MVC (*Model-View-Controller*). Artinya, CodeIgniter ini mampu memisahkan komponen-komponen dalam *coding* aplikasi berbasis web, sehingga diharapkan dapat dikelola dengan lebih mudah kedepannya.

2.7.2 Design Patern : MVC (*Model, View, Controller*)

CodeIgniter ialah *framework* PHP yang didasarkan aturan MVC (*Model-View-Cont*). Penggunaan MVC dapat memisahkan lapisan logika aplikasi dan lapisan presentasi.

Oleh karena itu, dalam pengembangan Web, programmer dapat fokus pada sistem inti, dan desainer Web dapat berfokus pada tampilan Web. Menariknya, skrip PHP, *query*, MySQL, Javascript, dan CSS dapat dipisahkan satu sama lain, tidak dilakukan dalam skrip besar yang membutuhkan banyak *resource* untuk dijalankan.

Gambar berikut menunjukkan proses aplikasi berdasarkan *framework* CodeIgniter:



Gambar 2.3 Arsitektur MVC

Sumber : Hakim “Membangun Web Berbasis *PHP* dengan Framework CodeIgniter” (2010:4)

Gambar di atas menunjukkan bahwa ketika *user request* datang, hal itu akan diproses oleh *Controller*, dan kemudian, jika operasi *database* diperlukan, *Controller* akan memanggil *Model*. Kemudian hasil *query* oleh *Model* akan dikembalikan ke *Controller*. *Controller* kemudian akan memanggil tampilan yang sesuai dan menggabungkannya dengan hasil *query* *Model*. Hasil akhir dari proses ini akan ditampilkan pada browser.

Dalam konteks *CodeIgniter* dan aplikasi dengan berbasis web, konsep MVC menyebabkan kode program terbagi dalam tiga kategori, yaitu :

1. *Model*

Berisi kode (dalam bentuk OOP *class*) dari aplikasi atau program yang digunakan untuk mengoperasikan *database*.

2. *View*

Berisi *template* HTML/XML atau PHP yang bertujuan untuk menampilkan data di browser

3. *Controller*

Berisi kode (dalam bentuk OOP *class*) dengan tujuan untuk mengatur aliran dari aplikasi (sebagai pengontrol *View* dan *Model*).

2.7.3 Smarty

Smarty merupakan salah satu *PHP template engine* yang memiliki fungsi untuk memisahkan antara *file-file* proses (.php) dan *file-file* tampilan (.html atau .css). Beberapa tujuan dari *smarty* :

1. Memisahkan *file* presentasi dari *file* kode aplikasi.
2. Perkembangan cepat / penyebaran untuk *programmer* dan desainer.
3. Cepat dan mudah dalam perawatannya.
4. Sintaks mudah dimengerti.
5. Fleksibel untuk pengembangan khusus.

Gratis dan *open source*. (<http://www.smarty.net/>)

2.7.4 CiSmart

CiSmart merupakan modifikasi dari *framework CodeIgniter* yang digabungkan bersama dengan *smarty template engine*. Dengan demikian bahwa dasar *framework* yang digunakan tetap menggunakan *CodeIgniter* memiliki konsep MVC (*Model-View-Controller*), sedangkan *smarty template engine* digunakan sebagai pengganti dalam pengelolaan *View*.

2.8 Normalisasi

Normalisasi ialah proses pengelompokan elemen data ke dalam tabel. Tabel ini menampilkan entitas dan hubungannya untuk menghilangkan redundansi data, menentukan kunci unik untuk mengakses data atau membentuk relasi, dan memudahkan untuk memodifikasi *database* (Connolly, 2010).

a. Tujuan dari Normalisasi

- Menghilangkan duplikasi data
- Mengurangi kerumitan pada data
- Memudahkan dalam proses modifikasi data

b. Proses Normalisasi

- Data dideskripsikan dalam bentuk tabel dan kemudian dianalisis menjadi beberapa tingkatan sesuai dengan kebutuhan tertentu.
- Jika tabel yang akan diuji tidak memenuhi persyaratan tertentu, perlu dilakukan pemecahan tabel menjadi beberapa tabel yang lebih sederhana untuk mendapatkan bentuk terbaik.

c. Tahapan Normalisasi

Tahapan normalisasi dilaksanakan dari tahap yang paling ringan (1NF) hingga tahap paling ketat (5NF). Normalisasi biasanya hanya mencapai level 3NF atau

BCNF (*Boyce-Codd Normal Form*), karena sudah cukup untuk menghasilkan *form* yang berkualitas tinggi.

1) Bentuk Normal Kesatu (1 NF)

- Jika tabel tidak memiliki atribut multi-nilai, atribut gabungan, atau kombinasinya dalam bidang data yang sama, tabel memenuhi bentuk standar 1NF.
- Masing-masing atribut dalam tabel harus berupa atribut *atomic* (tidak dapat dibagi lagi).

2) Bentuk Normal Kedua (2 NF)

- Jika format umum 2NF memenuhi format 1NF, itu dipenuhi dalam tabel, dan semua atribut kecuali *primary key* memiliki ketergantungan fungsional yang lengkap pada *primary key*.
- Jika ketergantungan fungsional dari beberapa atribut hanya sebagian (bergantung pada hanya sebagian dari *primary key*), tabel tidak memenuhi persyaratan 2NF.
- Jika ada atribut yang tidak bergantung pada *primary key*, maka perlu dilakukan pemindahan atau penghapusan untuk atribut tersebut.

3) Bentuk Normal Ketiga (3NF)

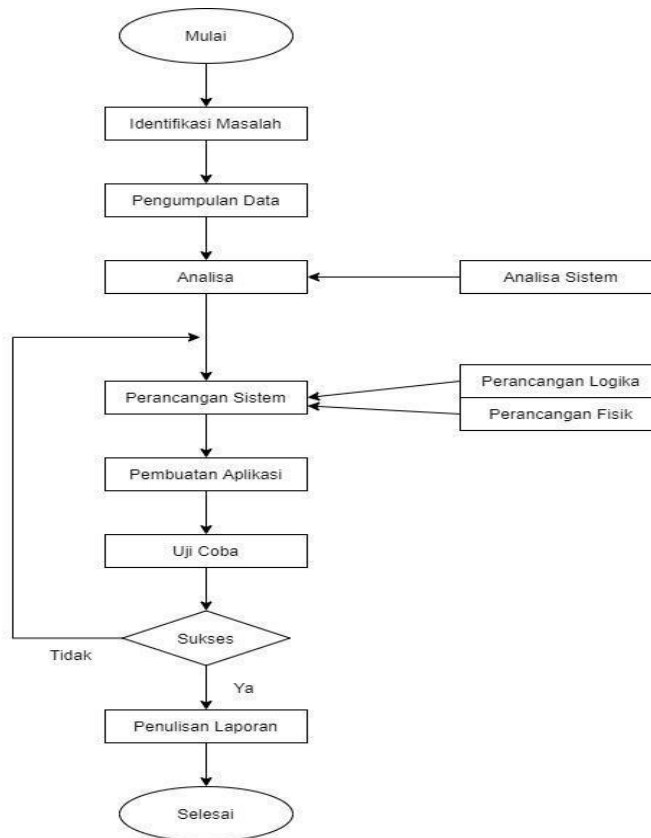
Jika memenuhi bentuk 2NF, maka akan memenuhi bentuk konvensional 3NF, dan jika tidak ada atribut *non primary key* yang bergantung pada atribut *non primary key* lainnya, itu disebut ketergantungan transitif, yang dapat diartikan sebagai hubungan antara dua atau lebih atribut *non key* dengan ketergantungan fungsional (*key/primary key*).

BAB III

METODOLOGI PENELITIAN

3.1 Kerangka Pemikiran

Berikut gambar kerangka pemikiran penelitian :



Gambar 3.1 Kerangka Pemikiran

Gambar 3.1 merupakan kerangka pemikiran dalam melakukan penelitian. Digunakan untuk menggambarkan alur proses sistem yang dirancang.

3.2 Identifikasi Masalah

Dilakukan identifikasi permasalahan berdasarkan latar belakang masalah yang telah disusun sebagai berikut:

- a. Aplikasi yang sekarang menggunakan basis desktop sehingga program sulit untuk dipantau dari jarak jauh karena masih offline.
- b. Sistem pengelolaan yang ada belum bisa memberikan informasi secara langsung dan ketepatan antara jumlah yang ada pada sistem dan di gudang.

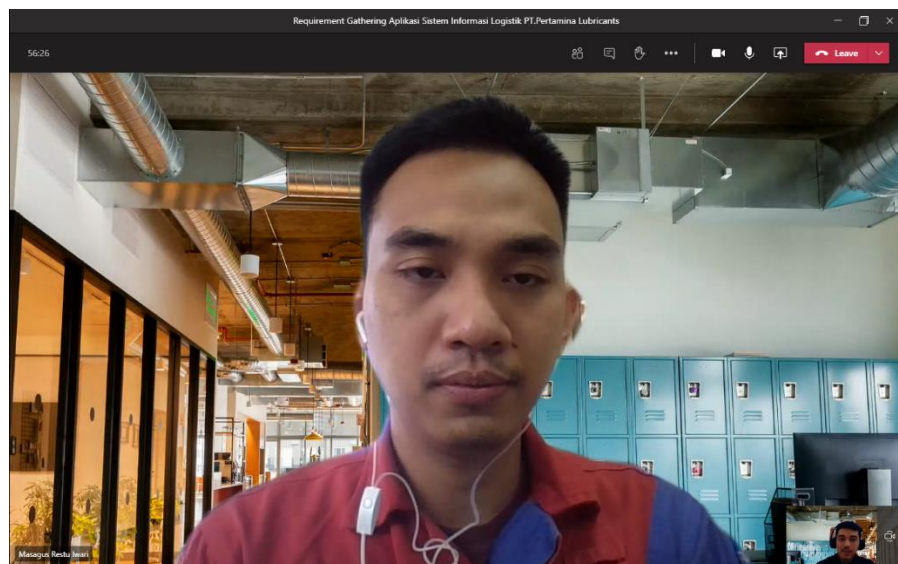
- c. Dengan kurangnya informasi logistik yang ada mempengaruhi pelayanan dan proses bisnis perusahaan.

3.3 Pengumpulan Data

Penelitian ini menggunakan teknik pengumpulan data berupa :

- a. Wawancara

Metode wawancara bertujuan untuk mengumpulkan informasi proses barang masuk dan keluar serta informasi kegiatan yang dilakukan masing-masing bagian dengan melakukan tanya jawab secara langsung dengan user *warehouse* yang menggunakan aplikasi Sistem Informasi Logistik sebelumnya yang bernama Masagus Restu Iwari. Wawancara dilakukan pada tanggal 26 November 2020 melalui *Microsoft Teams* pada pukul 10.29 pagi.



Gambar 3.2 Dokumentasi Wawancara dengan *User* aplikasi Sistem Informasi Logistik PT. Pertamina Lubricants

- b. Observasi

Pengamatan yang dilakukan adalah mengamati dan memahami secara langsung pengoperasian sistem yang saat ini beroperasi di *warehouse* PT. Pertamina Lubricants.

- c. Studi Pustaka

Melakukan pencarian buku, artikel, dan sumber literatur Internet yang berkaitan dengan topik penelitian. Selanjutnya memahami dan mempelajari materi yang kemudian dijadikan sebagai penunjang penelitian ini.

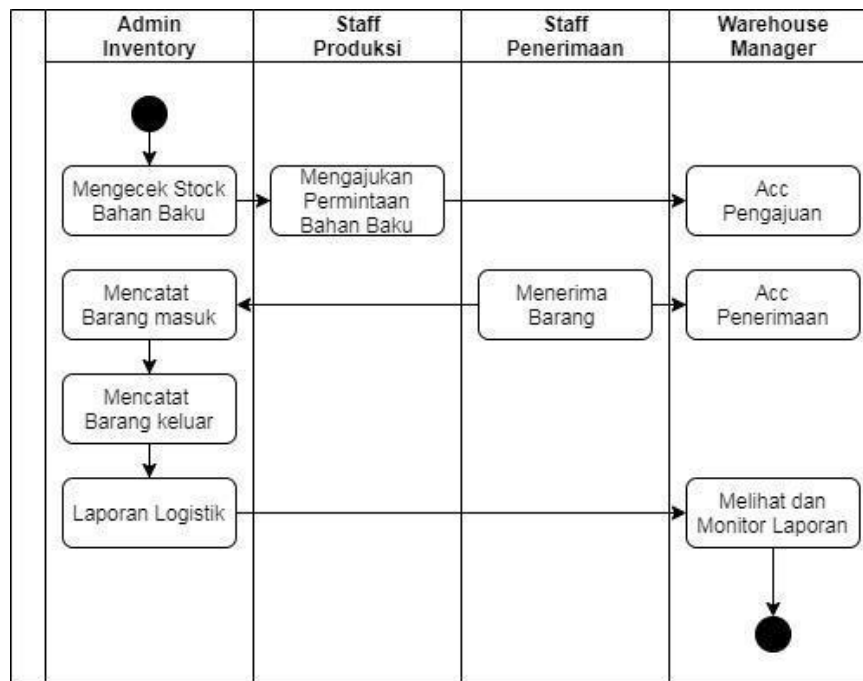
3.4 Analisa

Analisis kebutuhan sistem bertujuan untuk mengetahui semua aspek yang mempengaruhi sistem. Proses pengembangan sistem dilakukan berdasarkan standar yang merujuk pada aspek-aspek yang mempengaruhi sistem tersebut. Tahap analisis sistem ini akan melakukan analisis proses sistem yang sedang beroperasi, analisis basis data dan analisis kebutuhan non fungsional. Dalam bab ini, sub-bab berikut memperkenalkan pembahasan analisis sistem:

3.4.1 Analisa Sistem Saat Ini

Sistem yang beroperasi saat ini di PT. Pertamina Lubricants dalam mendapatkan informasi logistik belum maksimal dikarenakan UI (*User Interface*) yang sulit sehingga membutuhkan *learning period* yang lama, lalu sistem belum mampu untuk *real time*, ketika melakukan *update* data harus dari *database* langsung sehingga prosesnya menjadi lebih rumit dan operatornya hanya sebatas di aplikasi. Berikut adalah alur sistem yang sedang berjalan di PT. Pertamina Lubricants:

- a. Admin *inventory* mengecek stok bahan baku.
- b. Staff produksi mengajukan permintaan bahan baku ke admin gudang dan memesan bahan baku ke pemasok yang nantinya diverifikasi oleh *warehouse manager*.
- c. Staff penerimaan menerima bahan baku dari pemasok sesuai orderan yang diverifikasi oleh *warehouse manager*.
- d. Produksi melakukan permintaan bahan baku ke admin gudang.
- e. Admin *inventory* mencatat barang masuk dan pengeluaran bahan baku sesuai orderan dari produksi.
- f. *Warehouse manager* melakukan *monitoring* dan melihat seluruh laporan.

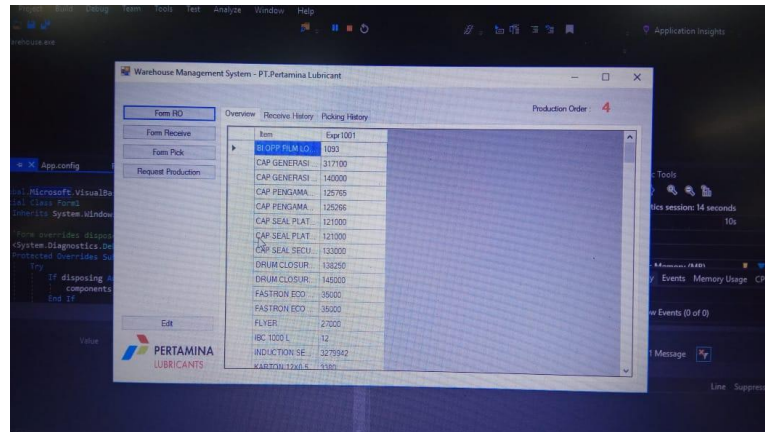


Gambar 3.3 Diagram Alur Analisa Sistem Saat Ini

3.4.2 Tampilan Aplikasi Saat Ini

Pada aplikasi yang lama menggunakan bahasa pemrograman visual studio.Net berbasis dekstop. Aplikasi ini dibangun sejak tahun 2016. Sudah mengalami beberapa penyesuaian tentang fitur sistemnya. Jadwal terakhir maintenance aplikasi ini sudah satu tahun yang lalu. Beberapa kekurangan dari aplikasi saat ini adalah jika perlu dilakukan pemantauan atau mengambil *database* dan laporan dari program itu sendiri, tidak dapat dilakukan karena program hanya bisa diakses dari jarak jauh. Hal ini menyulitkan dalam penerapan *software* dikarenakan setiap pengguna harus terlebih dahulu menginstal dan menyesuaikan serta menyiapkan *database* (jika tidak terhubung secara otomatis). Selain itu, akan sulit bagi developer untuk memperbarui dan memelihara *software* dan sistem itu sendiri. Sehingga dengan alasan itu juga penelitian ini menggunakan aplikasi berbasis web agar bisa dipantau dari jarak jauh dan lebih fleksibel dalam penggunaan aplikasi, karena *file* dan *database software* akan terpusat di satu titik server, selain itu proses pembaruan atau perawatan *software* menjadi lebih mudah.

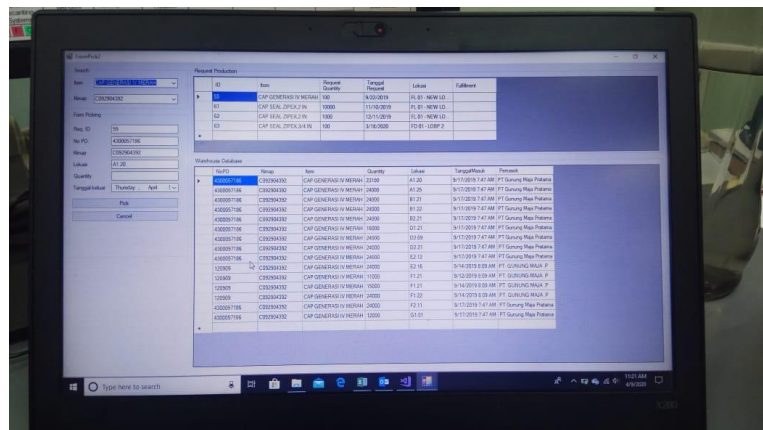
1. Form Menu Utama



Gambar 3.4 Form Menu Utama

Pada tampilan menu utama aplikasi lama, terdapat empat tombol menu yaitu tombol form RO, Form Receive, Form Pick dan Request Production serta menampilkan item barang, receive history dan packing history. Terdapat juga informasi jumlah Production Order yang ada saat ini.

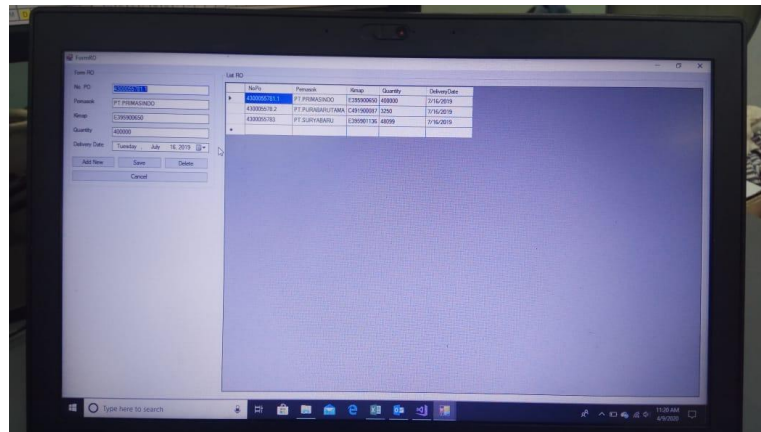
2. Form Request Production



Gambar 3.5 Form Request Production

Form *Request Production* terdapat beberapa input yaitu item, kuantitas, lokasi (LOBP1, LOBP2, New LOBP), dan tanggal request.

3. Form Production Order

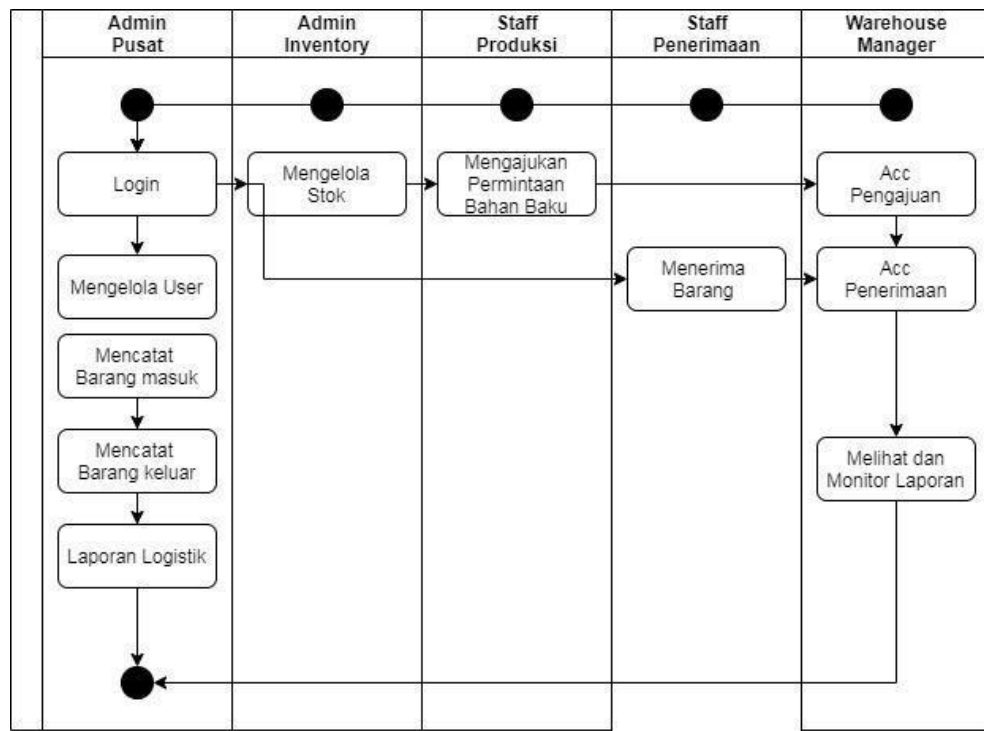


Gambar 3.6 Form Production Order

Form *Production Order* (RO) terdapat beberapa input yaitu nomor PO, daftar pemasok, kimap (ID Barang), kuantitas, dan *delivery date*.

3.4.3 Analisa Sistem Usulan

Dengan melihat masalah diatas, disini pada penelitian ini mengusulkan sebuah sistem logistik berbasis web yang bisa diakses oleh admin dan operator gudang yang terkait. Diharapkan dengan menggunakan basis *web* bisa memudahkan admin dalam mendapatkan informasi ketersediaan barang dan meningkatkan pelayanan perusahaan dan bisa menjawab masalah-masalah yang ada diperusahaan.



Gambar 3.7 Diagram Alur Analisa Sistem Usulan

3.4.4 Analisa Kebutuhan Pengguna

Pengguna pada sistem ini adalah admin pusat, admin *inventory*, staff produksi, staff penerimaan dan *warehouse manager*. Admin pusat adalah entitas yang memiliki peran penuh dalam mengelola sistem logistik. Adapun penjelasan mengenai kebutuhan pengguna dibagi menjadi dua yaitu :

1. *Functional Requirements*

- a. Terdapat Login dan Logout untuk user.
- b. Sistem bisa mengelola data *user*.
- c. Sistem bisa melihat daftar material masuk.
- d. Sistem bisa melihat daftar material keluar.
- e. Sistem bisa mengelola data tipe material.
- f. Sistem bisa mengelola data lokasi.
- g. Sistem bisa mengelola data pemasok.
- h. Sistem bisa mengelola data material.
- i. Sistem bisa mengelola stok barang.
- j. Sistem bisa melihat posisi stok material.
- k. Sistem bisa membuat *request production*.
- l. Sistem bisa mengelola material.
- m. Sistem bisa membuat *release order*.
- n. Sistem bisa melihat *receive history*.
- o. Sistem bisa mengelola aktivitas gudang seperti verifikasi *release order* dan *request production*.
- p. Sistem mampu melihat *packing history*.

2. *Non-Functional Requirements*

- a. Dapat digunakan kapanpun dan dimanapun.
- b. Sistem ini dapat menyediakan informasi mengenai lokasi material sehingga memudahkan operator dalam mengelola logistik gudang.
- c. Tampilan sistem mudah dipahami dan digunakan seperti tampilan yang *responsive* dan ada cara penggunaannya.

3.4.5 Analisa Kebutuhan Perangkat Lunak

Penelitian ini menggunakan beberapa software. berikut adalah spesifikasi:

1. HTML 5
2. CSS3
3. PHP 7.2

4. CI (Codeigniter) 4
5. XAMPP 7.2.34
6. Database MYSQL 7.2.34
7. Notepad ++ 7.9.2
8. Javascript ECMAScript 5
9. Google Chrome 76

3.4.6 Analisa Kebutuhan Perangkat Keras

Untuk melakukan penelitian ini diperlukan perangkat keras (*hardware*). Spesifikasi perangkat keras yang diperlukan adalah:

1. System Manufactur Lenovo Z10
2. Processor Intel Core i7 CPU 2.00Ghz 2.60 GHz
3. RAM 4 GB
4. Main board Intel
5. VGA NVIDIA
6. Hardisk 1 TB
7. Keyboard
8. Mouse

3.5 Perancangan Sistem

3.5.1 Perancangan Basis Data

Dalam hal ini dibahas mengenai Normalisasi, *entity relationship diagram* (ERD) dan relasi tabel.

1. Bentuk Un-Normalisasi

Tabel material	=	*id_material, nama_material, id_tipe, nama_tipe, id_lokasi, nama_lokasi, deskripsi, id_pemasok, nama_pemasok, id_permintaan, jumlah_permintaan, tgl_permintaan, id_ro, no_po, jumlah_ro, tgl_kirim, id_keluar, tgl_keluar, jumlah_keluar, id_masuk, jumlah_masuk, id_user, nama_user, username, password, level.
----------------	---	--

2. Bentuk Normal Pertama (1st NF)

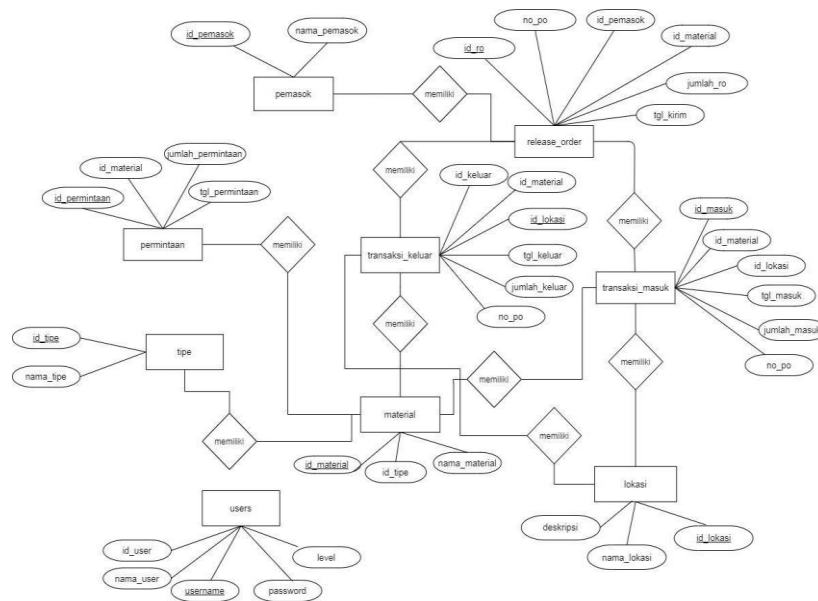
Tabel material	=	*id_material, nama_material, *id_tipe, nama_tipe, *id_lokasi, nama_lokasi, deskripsi, *id_pemasok, nama_pemasok, *id_permintaan, jumlah_permintaan, tgl_permintaan, *id_ro, no_po, jumlah_ro, tgl_kirim, *id_keluar, tgl_keluar, jumlah_keluar, *id_masuk, jumlah_masuk, *id_user, nama_user, username, password, level.
----------------	---	--

3. Bentuk Normalisasi Kedua (2nd NF)

Tabel material	=	*id_material, nama_material.
Tabel tipe		*id_tipe, nama_tipe.
Tabel lokasi		*id_lokasi, nama_lokasi.
Tabel pemasok		*id_pemasok, nama_pemasok.
Tabel permintaan		*id_permintaan, id_material, jumlah_permintaan, tgl_permintaan.
Tabel release_order		*id_ro, no_po, id_pemasok, id_material, jumlah_ro, tgl_kirim.
Tabel transaksi_keluar		*id_keluar, tgl_keluar, id_material, jumlah_keluar, no_po, id_lokasi
Tabel transaksi_masuk		*id_masuk, tgl_masuk, id_material, jumlah_masuk, no_po, id_lokasi.
Tabel users		*id_user, nama_user, username, password, level.

3.5.2 Entity Relationship Diagram (ERD)

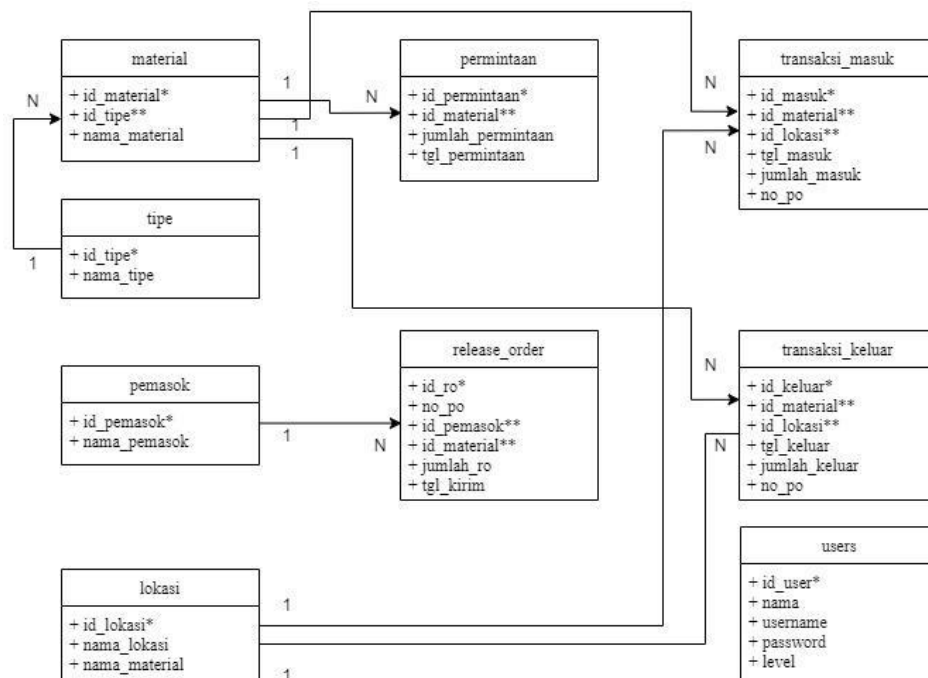
Entity Relationship Diagram (ERD) ialah diagram yang menunjukkan entitas yang terlibat dalam sistem dan hubungan atau hubungan antara entitas tersebut. ERD yang dibentuk untuk sistem yang akan dirancang adalah sebagai berikut:



Gambar 3.8 Entity Relationship Diagram (ERD)

3.5.3 Logical Record Structure (LRS)

Database dengan beberapa tabel biasanya terdapat pada sebuah sistem informasi. Beberapa dari tabel ini berhubungan satu sama lain. Hubungan antar tabel digunakan untuk menghubungkan satu tabel ke tabel lainnya, yang mampu mempercepat akses *database*. Aturan *relational database* menyatakan bahwa satu tabel dan tabel lain terkait dengan kunci unik atau kunci utama. Hubungan tabel ditunjukkan pada Gambar 3.9.



Gambar 3.9 Logical Record Structure (LRS)

3.5.4 Struktur Tabel

1. Tabel Users

Nama_Field	Tipe	Primary Key	Foreign Key
id_user	int (10)	y	
nama_user	varchar (50)		
username	varchar (50)		
password	varchar (50)		
level	varchar (20)		

2. Tabel Lokasi

Nama_Field	Tipe	Primary Key	Foreign Key
id_lokasi	int (10)	Y	
nama_lokasi	varchar (50)		
deskripsi	varchar (100)		

3. Tabel Material

Nama_Field	Tipe	Primary Key	Foreign Key
id_material	int (10)	y	
nama_material	varchar (50)		
id_tipe	int (10)		Y

4. Tabel Pemasok

Nama_Field	Tipe	Primary Key	Foreign Key
id_pemasok	int (10)	y	
nama_pemasok	varchar (50)		

5. Tabel Permintaan

Nama_Field	Tipe	Primary Key	Foreign Key
id_permintaan	int (10)	y	
id_material	int (10)		y
jumlah_permintaan	int (10)		
tgl_permintaan	date		

6. Tabel Release Order

Nama_Field	Tipe	Primary Key	Foreign Key
id_ro	int (10)	Y	
no_po	varchar (10)		
id_pemasok	int (10)		y
id_material	int (10)		y
jumlah_ro	int (10)		
tgl_kirim	date		

7. Tabel Tipe

Nama_Field	Tipe	Primary Key	Foreign Key
id_tipe	int (10)	y	
nama_tipe	varchar (50)		

8. Tabel Transaksi Masuk

Nama_Field	Tipe	Primary Key	Foreign Key
id_masuk	int (10)	y	
tgl_keluar	date		
id_material	int (10)		y

jumlah_masuk	int (10)		
no_po	int (10)		
id_lokasi	int (10)		y

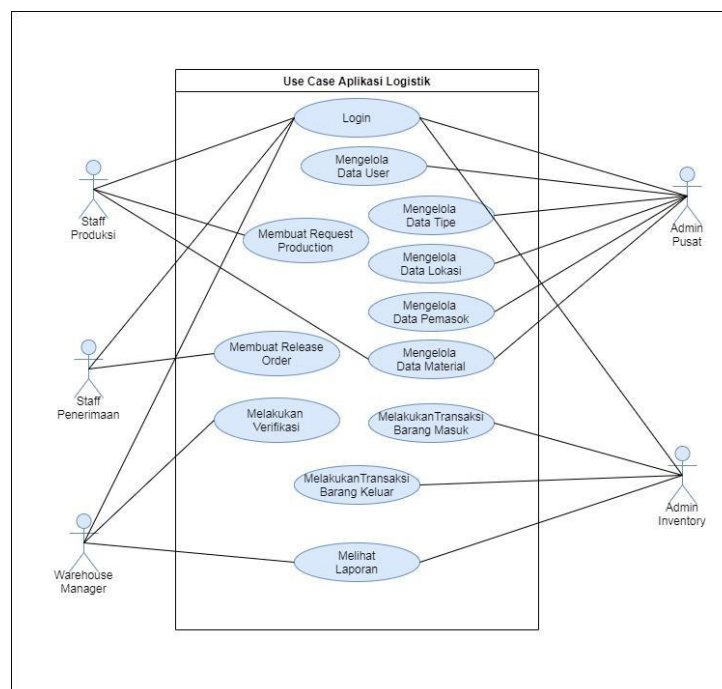
9. Tabel Transaksi keluar

Nama_Field	Type	Primary Key	Foreign Key
id_keluar	int (10)	y	
tgl_keluar	date		
id_material	int (10)		y
jumlah_keluar	int (10)		
no_po	int (10)		
id_lokasi	int (10)		y

3.6 Perancangan Aplikasi

3.6.1 Use Case Diagram

Use Case Diagram menjelaskan interaksi antara komponen aplikasi dan menjelaskan bagaimana mereka berinteraksi dengan pengguna.

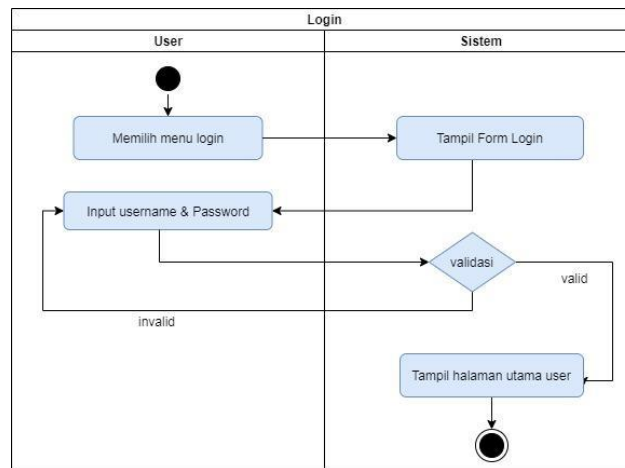


Gambar 3.10 Use Case Diagram

Gambar 3.10 adalah *use case diagram* dari aplikasi sistem informasi logistik. Pada *use case diagram* diatas terdiri dari 5 aktor yang memiliki level akses yang berbeda-beda. Use case ini terdapat dua belas aktifitas yang dilakukan.

3.6.2 Activity Diagram

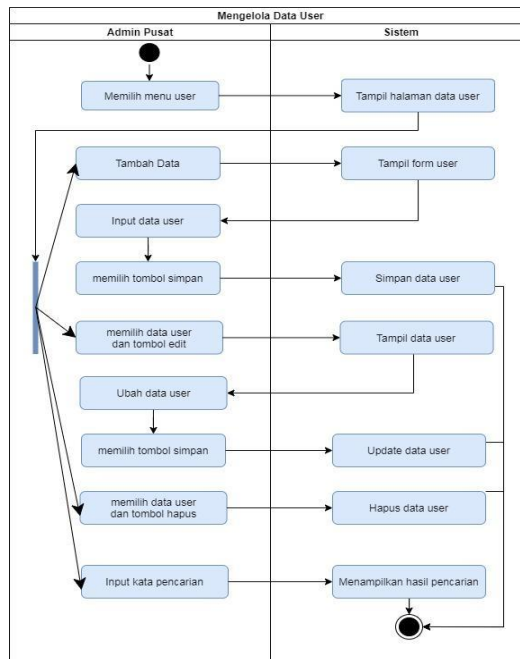
1. Activity Diagram Login



Gambar 3.11 Activity Diagram Login

Gambar 3.11 adalah *activity diagram* untuk proses *login*. Pada diagram ini menggambarkan kegiatan alur aktor dan sistem. Disini *user* diminta untuk mengisi *username* dan *password* kemudian sistem akan melakukan validasi. Jika *username* dan *password* benar maka akan masuk kedalam sistem, tetapi jika *username* dan *password* salah maka akan mengulang kembali untuk memasukkan *username* dan *password*.

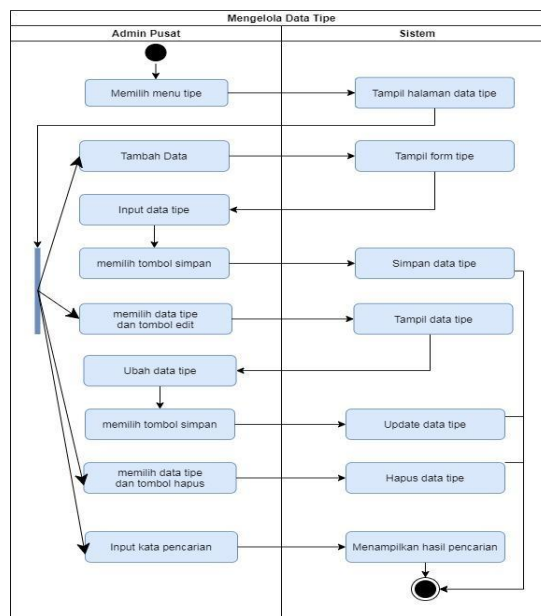
2. Activity Diagram Mengelola Data User



Gambar 3.12 Activity Diagram Mengelola Data User

Gambar 3.12 adalah *activity diagram* untuk proses mengelola data *user*. Pada diagram ini aktor admin pusat melakukan kegiatan menambah, mengubah, menghapus dan mencari data *user*.

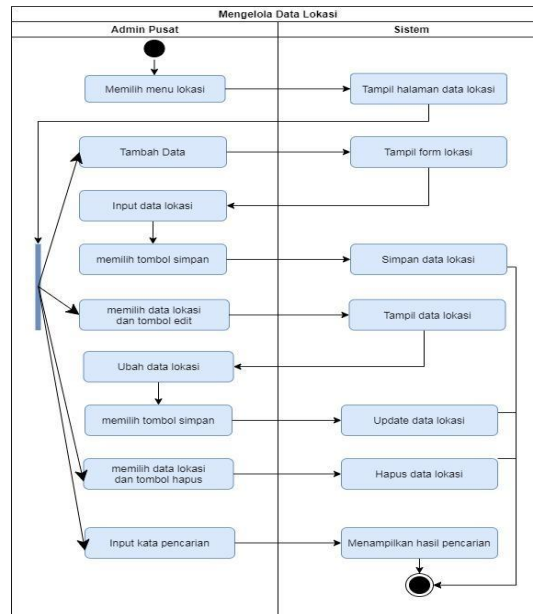
3. Activity Diagram Mengelola Data Tipe



Gambar 3.13 Activity Diagram Mengelola Data Tipe

Gambar 3.13 adalah *activity diagram* untuk proses mengelola data tipe. Pada diagram ini aktor admin pusat melakukan kegiatan menambah, mengubah, menghapus dan mencari data tipe.

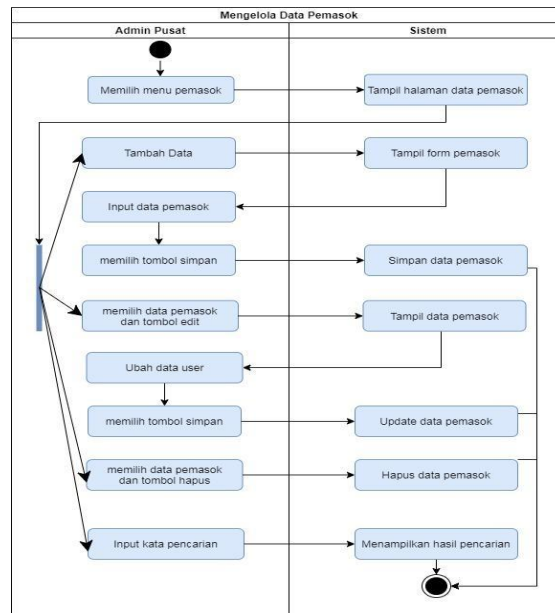
4. Activity Diagram Mengelola Data Lokasi



Gambar 3.14 Activity Diagram Mengelola Data Lokasi

Gambar 3.14 adalah *activity diagram* untuk proses mengelola data lokasi. Pada diagram ini aktor admin pusat melakukan kegiatan menambah, mengubah, menghapus dan mencari data lokasi.

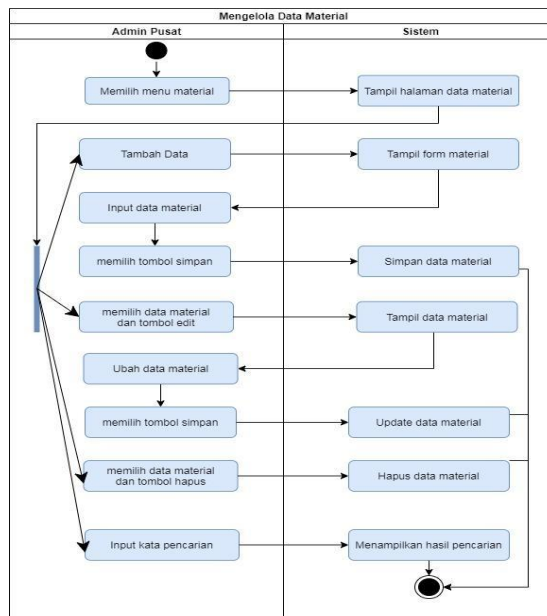
5. Activity Diagram Mengelola Data Pemasok



Gambar 3.15 Activity Diagram Mengelola Data Pemasok

Gambar 3.15 adalah *activity diagram* untuk proses mengelola data pemasok. Pada diagram ini aktor admin pusat melakukan kegiatan menambah, mengubah, menghapus dan mencari data pemasok.

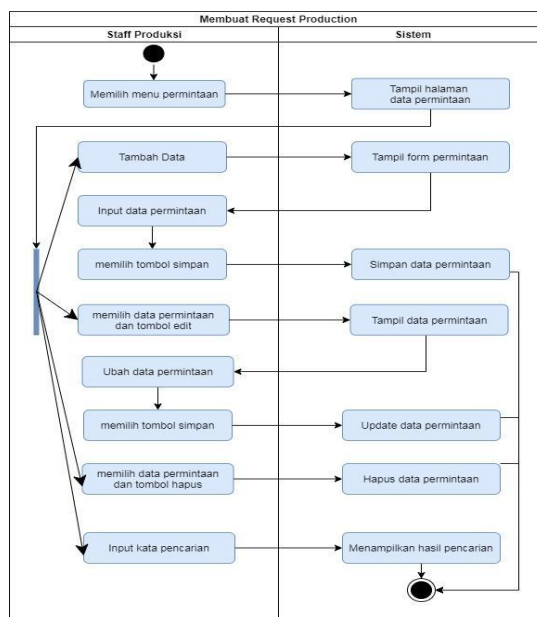
6. Activity Diagram Mengelola Data Material



Gambar 3.16 Activity Diagram Mengelola Data Material

Gambar 3.16 adalah *activity diagram* untuk proses mengelola data material. Pada diagram ini aktor admin pusat melakukan kegiatan menambah, mengubah, menghapus dan mencari data material.

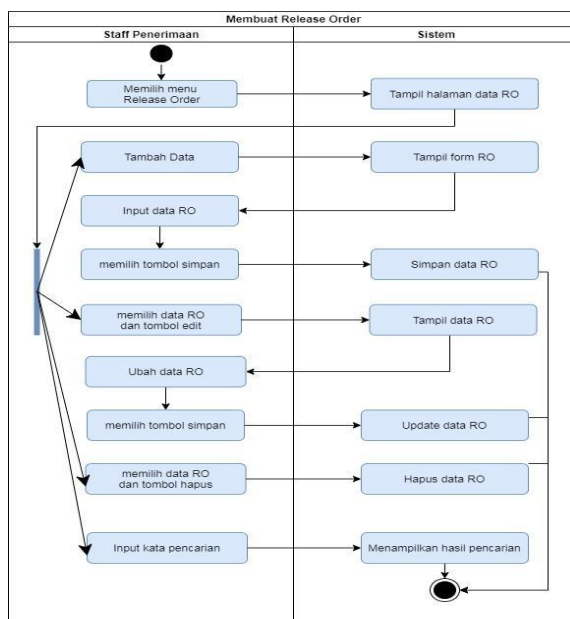
7. Activity Diagram Membuat Request Production



Gambar 3.17 Activity Diagram Membuat Request Production

Gambar 3.17 adalah *activity diagram* untuk proses membuat *request production*. Pada diagram ini aktor staf produksi melakukan kegiatan menambah, mengubah, menghapus dan mencari data *request production*.

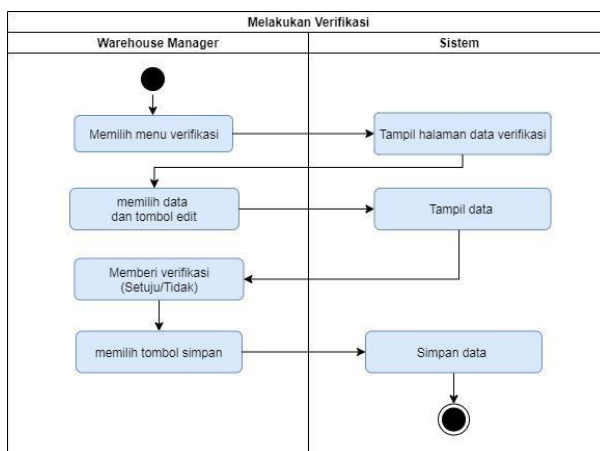
8. Activity Diagram Membuat Release Order



Gambar 3.18 Activity Diagram Membuat Release Order

Gambar 3.18 adalah *activity diagram* untuk proses membuat *release order*. Pada diagram ini aktor staf penerimaan melakukan kegiatan menambah, mengubah, menghapus dan mencari data *release order*.

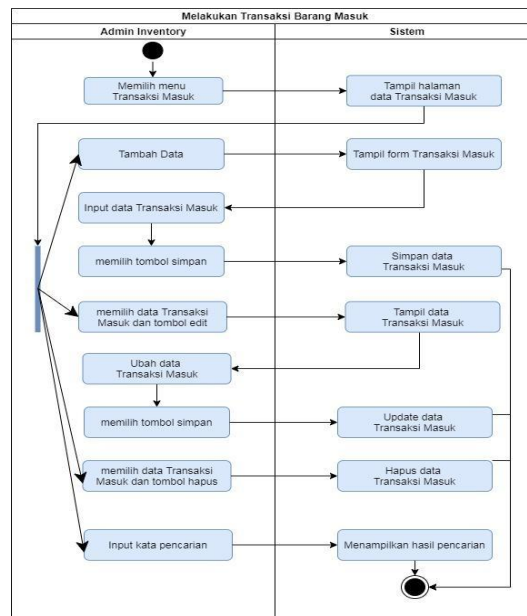
9. Activity Diagram Melakukan Verifikasi



Gambar 3.19 Activity Diagram Melakukan Verifikasi

Gambar 3.19 adalah *activity diagram* untuk proses melakukan verifikasi. Pada diagram ini aktor *warehouse manager* melakukan kegiatan verifikasi data *release order* dan *request production*.

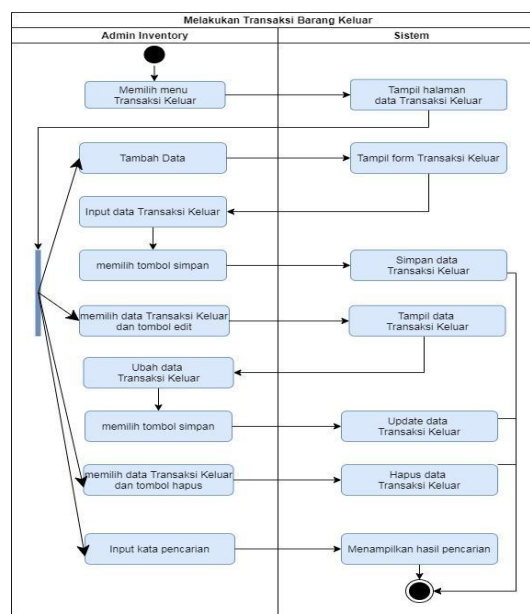
10. Activity Diagram Melakukan Transaksi Masuk



Gambar 3.20 Activity Diagram Melakukan Transaksi Masuk

Gambar 3.20 adalah *activity diagram* untuk proses melakukan transaksi masuk. Pada diagram ini aktor *admin inventory* melakukan kegiatan menambah, mengubah, menghapus dan mencari data transaksi masuk.

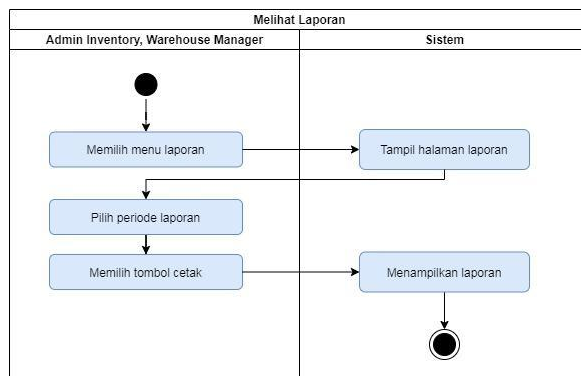
11. Activity Diagram Melakukan Transaksi Keluar



Gambar 3.21 Activity Diagram Melakukan Transaksi Keluar

Gambar 3.21 adalah *activity diagram* untuk proses melakukan transaksi keluar. Pada diagram ini aktor *admin inventory* melakukan kegiatan menambah, mengubah, menghapus dan mencari data transaksi keluar.

12. Activity Diagram Melihat Laporan

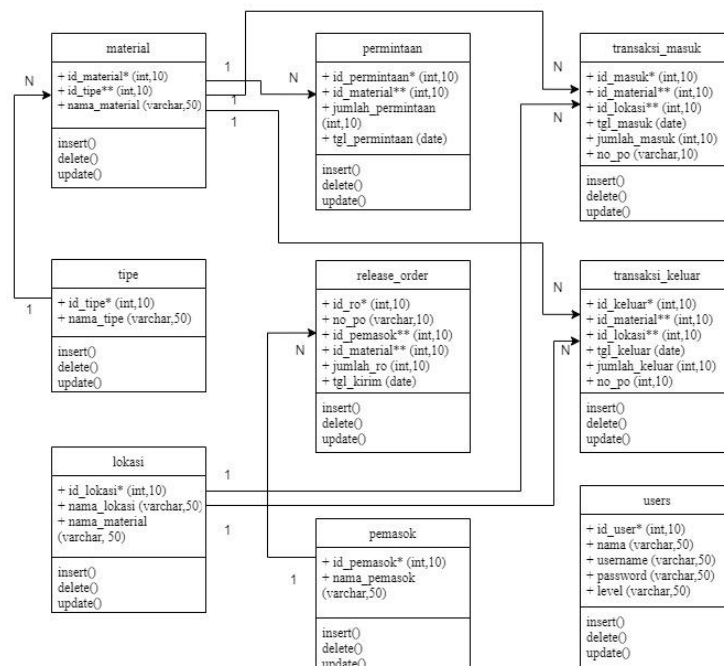


Gambar 3.22 Activity Diagram Melihat Laporan

Gambar 3.22 adalah *activity diagram* untuk proses melihat laporan. Pada diagram ini aktor *admin inventory* dan *warehouse manager* melihat laporan yang nantinya bisa di cetak.

3.6.3 Class Diagram

Class Diagram atau Diagram kelas menggambarkan sebuah struktur sistem dengan mendefinisikan kelas-kelas yang akan dibangun. Properti, metode, atau operasi merupakan bagian dari kelas.

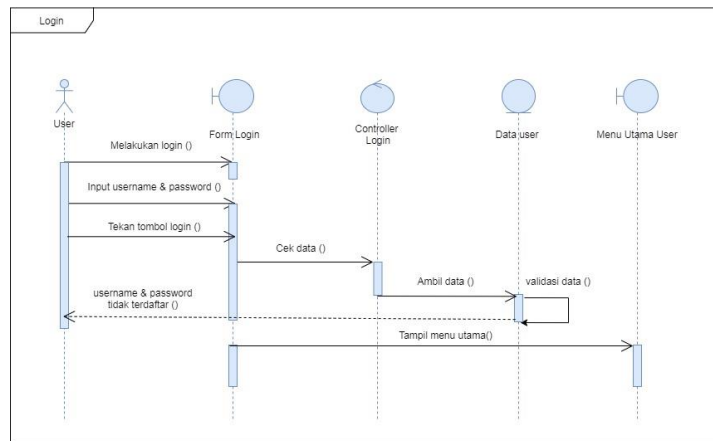


Gambar 3.23 Class Diagram

Gambar 3.10 adalah *class diagram* untuk aplikasi sistem informasi logistik. *Class diagram* diatas terdiri dari 9 tabel. Terdiri dari tabel user, pemasok, lokasi, tipe, material, permintaan, transaksi masuk, trnsaksi keluar dan release order.

3.6.4 Sequence Diagram

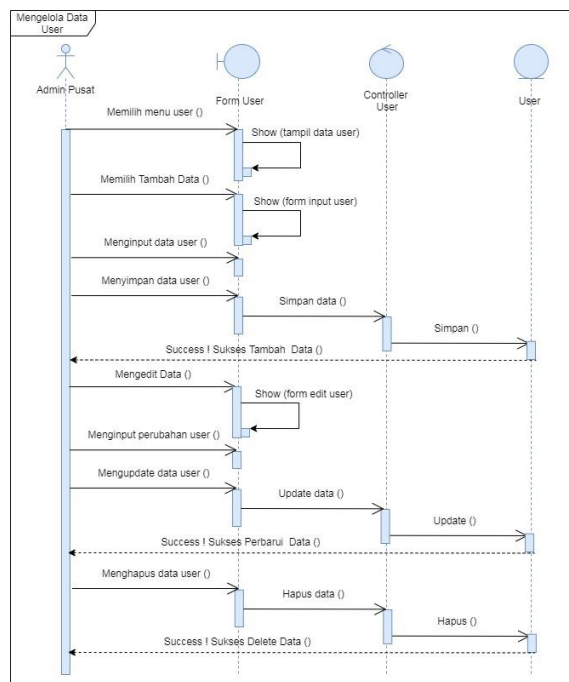
1. Sequence Diagram Login



Gambar 3.24 Sequence Diagram Login

Gambar 3.24 adalah *sequence diagram* untuk proses login. Gambar ini menggambarkan proses *user* memasuki sistem dengan memasukkan nama pengguna dan kata sandi.

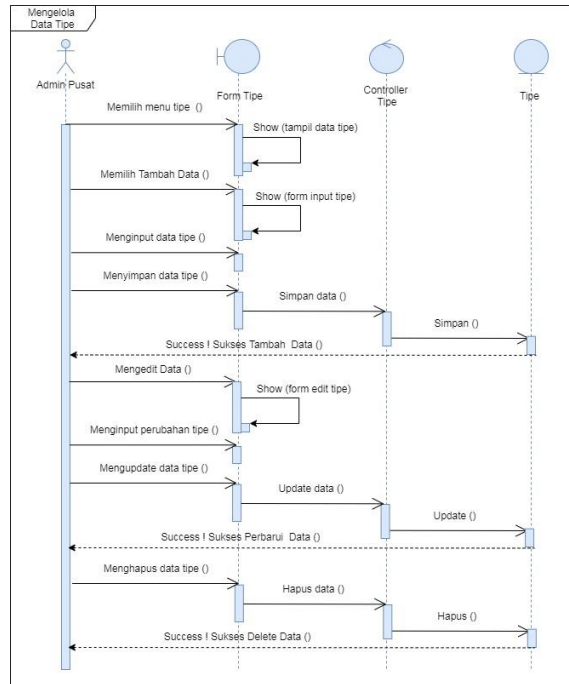
2. Sequence Diagram Mengelola Data User



Gambar 3.25 Sequence Diagram Mengelola Data User

Gambar 3.25 adalah *sequence diagram* untuk proses mengelola data *user*. Gambar ini menggambarkan proses *user* admin pusat dalam mengelola data *user* seperti menambah, mengubah dan menghapus data *user*.

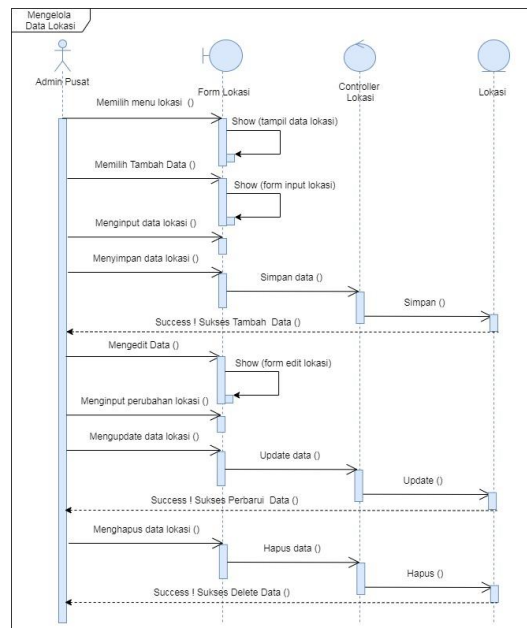
3. Sequence Diagram Mengelola Data Tipe



Gambar 3.26 Sequence Diagram Mengelola Data Tipe

Gambar 3.26 adalah *sequence diagram* untuk proses mengelola data tipe. Gambar ini menggambarkan proses *user* admin pusat dalam mengelola data tipe seperti menambah, mengubah dan menghapus data tipe.

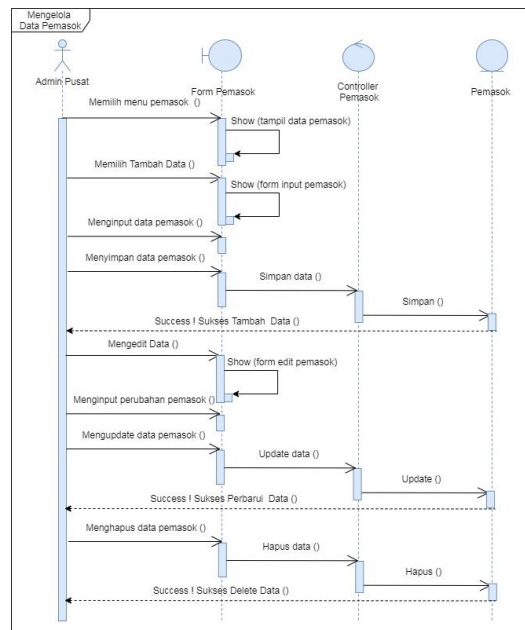
4. Sequence Diagram Mengelola Data Lokasi



Gambar 3.27 Sequence Diagram Mengelola Data Lokasi

Gambar 3.27 adalah *sequence diagram* untuk proses mengelola data lokasi. Gambar ini menggambarkan proses *user* admin pusat dalam mengelola data lokasi seperti menambah, mengubah dan menghapus data lokasi.

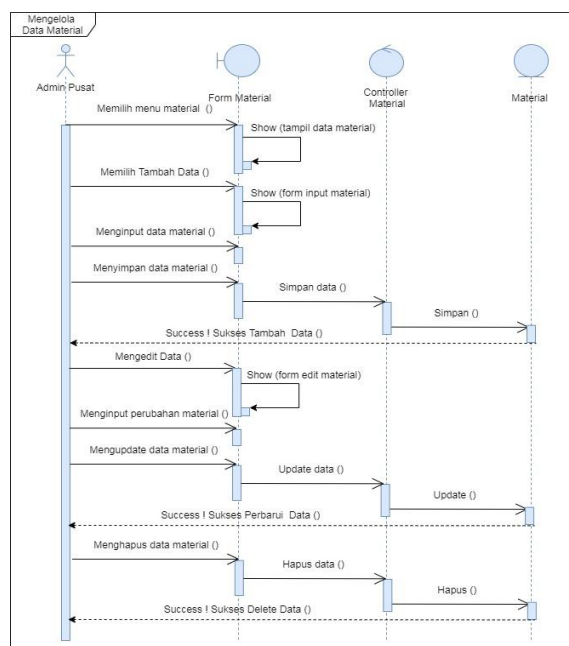
5. Sequence Diagram Mengelola Data Pemasok



Gambar 3.28 Sequence Diagram Mengelola Data Pemasok

Gambar 3.28 adalah *sequence diagram* untuk proses mengelola data pemasok. Gambar ini menggambarkan proses *user* admin pusat dalam mengelola data pemasok seperti menambah, mengubah dan menghapus data pemasok.

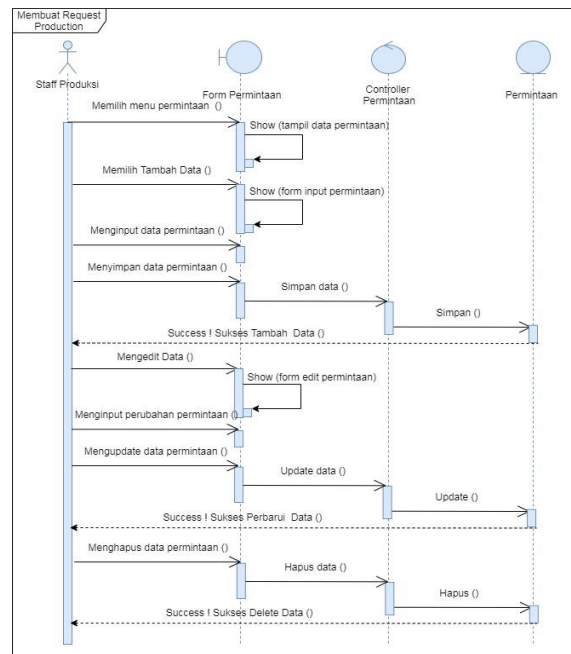
6. Sequence Diagram Mengelola Data Material



Gambar 3.29 Sequence Diagram Mengelola Data Material

Gambar 3.29 adalah *sequence diagram* untuk proses mengelola data material. Gambar ini menggambarkan proses *user* admin pusat dalam mengelola data material seperti menambah, mengubah dan menghapus data material.

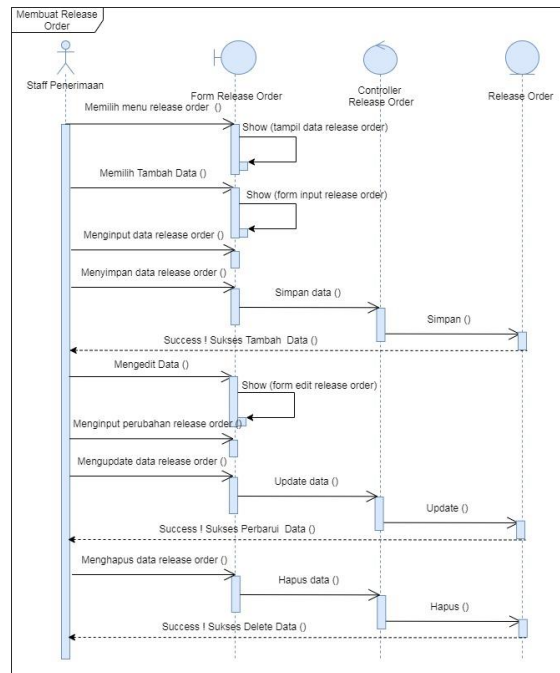
7. Sequence Diagram Membuat Request Production



Gambar 3.30 Sequence Diagram Membuat Request Production

Gambar 3.30 adalah *sequence diagram* untuk proses mengelola data *request production*. Gambar ini menggambarkan proses *user* staf produksi untuk mengelola data *request production* seperti menambah, mengubah dan menghapus data *request production*.

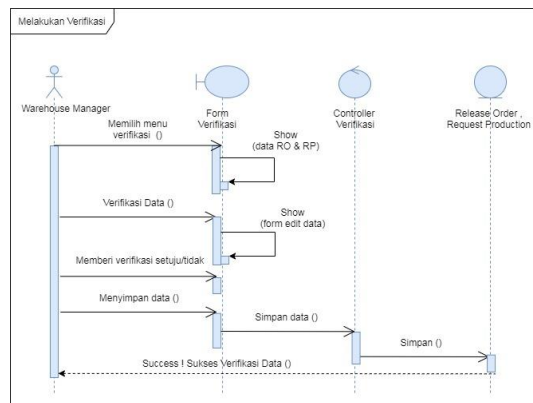
8. Sequence Diagram Membuat Release Order



Gambar 3.31 Sequence Diagram Membuat Release Order

Gambar 3.31 adalah *sequence diagram* untuk proses mengelola data *release order*. Pada diagram ini menggambarkan proses *user* staf penerimaan untuk mengelola data *release order* seperti menambah, mengubah dan menghapus data *release order*.

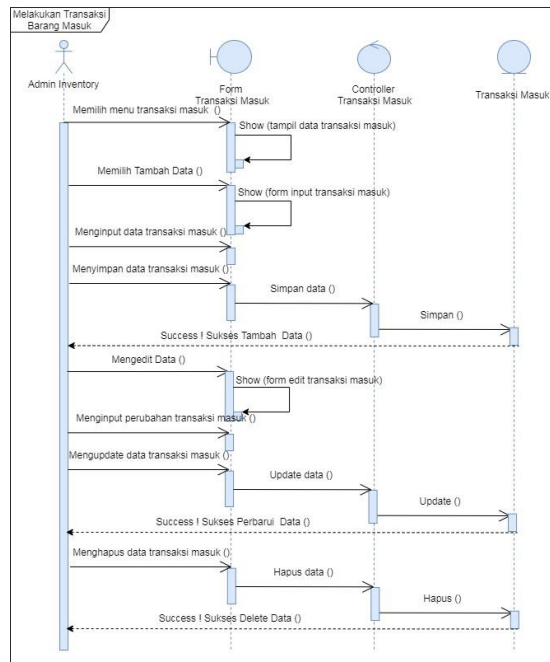
9. Sequence Diagram Melakukan Verifikasi



Gambar 3.32 Sequence Diagram Melakukan Verifikasi

Gambar 3.32 adalah *sequence diagram* untuk proses mengelola data *verifikasi*. Gambar ini menggambarkan proses *user warehouse manager* pusat untuk mengelola data *release order* dan *request production* dengan melakukan verifikasi.

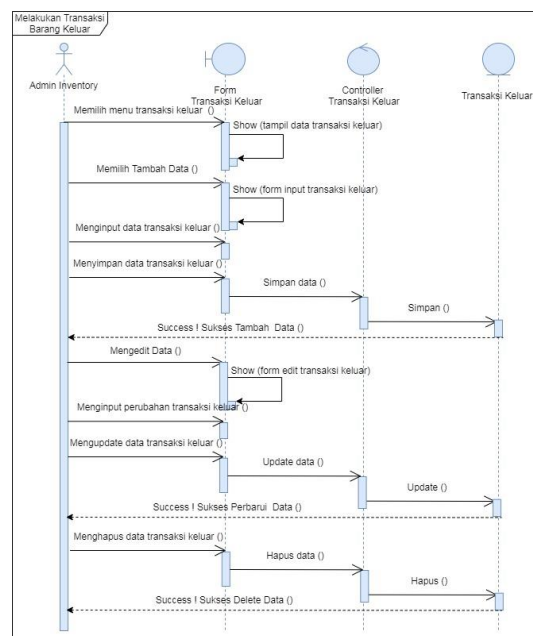
10. Sequence Diagram Melakukan Transaksi Masuk



Gambar 3.33 Sequence Diagram Melakukan Transaksi Masuk

Gambar 3.33 adalah *sequence diagram* untuk proses mengelola data transaksi masuk. Gambar ini menggambarkan proses *user admin inventory* untuk mengelola data transaksi masuk seperti menambah, mengubah dan menghapus data transaksi masuk.

11. Sequence Diagram Melakukan Transaksi Keluar

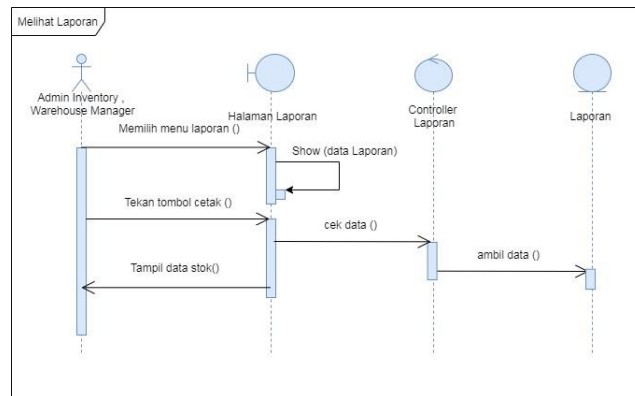


Gambar 3.34 Sequence Diagram Melakukan Transaksi Keluar

Gambar 3.34 adalah *sequence diagram* untuk proses mengelola data transaksi keluar. Gambar ini menggambarkan proses *user admin inventory* untuk

mengelola data transaksi keluar seperti menambah, mengubah dan menghapus data transaksi keluar.

12. Sequence Diagram Melihat Laporan

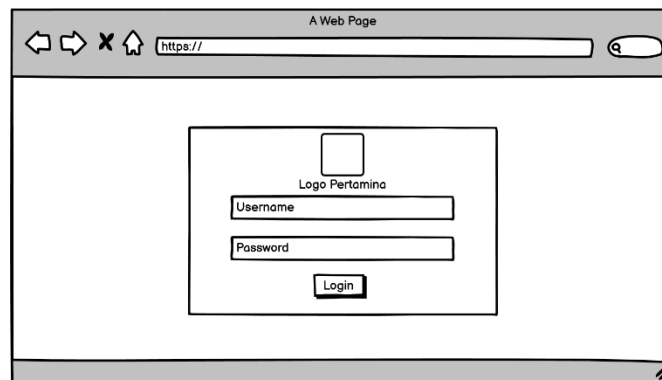


Gambar 3.35 Sequence Diagram Melihat Laporan

Gambar 3.35 adalah *sequence diagram* untuk proses melihat laporan. Gambar ini menggambarkan proses *user admin inventory* dan *warehouse manager* untuk melihat laporan seperti laporan transaksi dan laporan stok.

3.6.5 User Interface

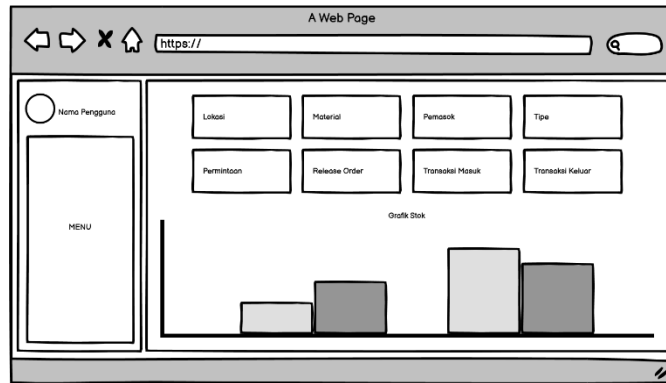
1. Interface Halaman Login



Gambar 3.36 Interface Halaman Login

Gambar 3.36 adalah *interface* rancangan halaman *login*. Menampilkan inputan *username* dan *password* dan logo perusahaan di bagian atas. Kemudian terdapat tombol login.

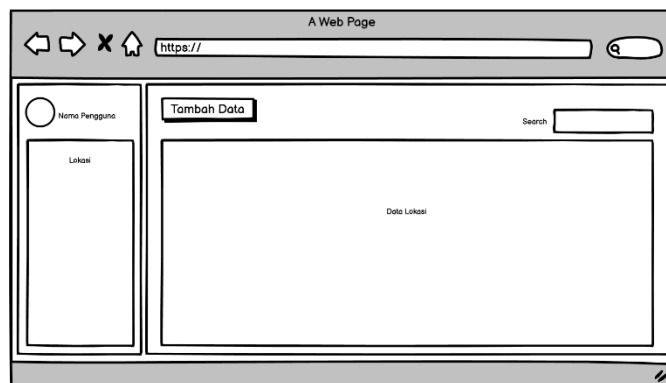
2. Interface Halaman Dashboard



Gambar 3.37 Interface Halaman Dashboard

Gambar 3.37 adalah *interface* rancangan halaman *dashboard*. Menampilkan data jumlah rekapan tabel lokasi material, stok hingga transaksi.

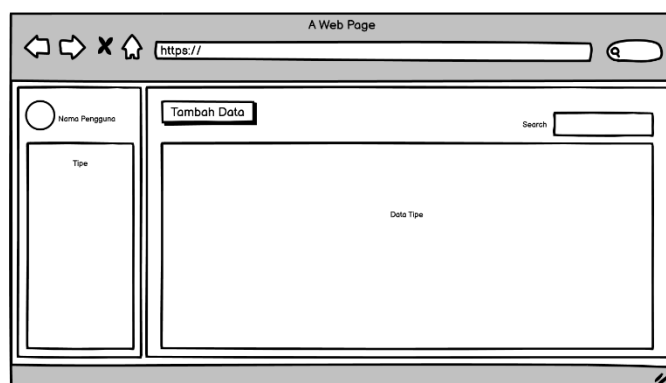
3. *Interface* Halaman Lokasi



Gambar 3.38 Interface Halaman Lokasi

Gambar 3.38 adalah *interface* rancangan halaman lokasi. Menampilkan data lokasi dan tombol tambah data. Terdapat juga kotak pencarian.

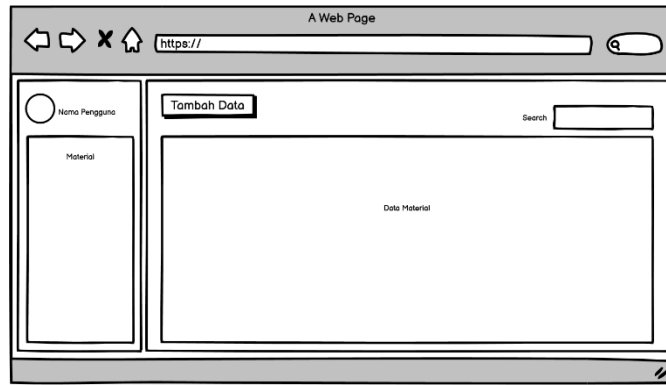
4. *Interface* Halaman Tipe



Gambar 3.39 Interface Halaman Tipe

Gambar 3.39 adalah *interface* rancangan halaman tipe. Menampilkan data tipe dan tombol tambah data. Terdapat juga kotak pencarian.

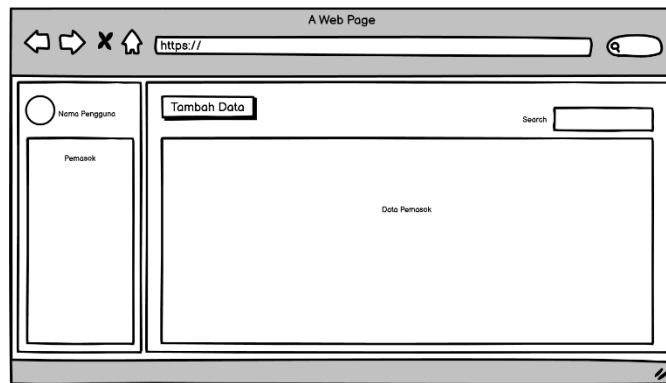
5. *Interface* Halaman Material



Gambar 3.40 Interface Halaman Material

Gambar 3.40 adalah *interface* rancangan halaman material. Menampilkan data material dan tombol tambah data. Terdapat juga kotak pencarian.

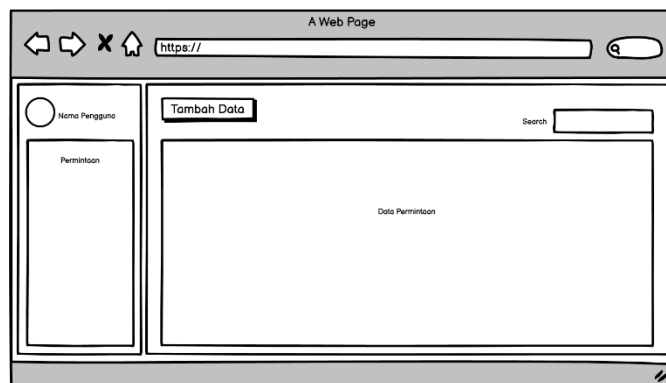
6. *Interface* Halaman Pemasok



Gambar 3.41 Interface Halaman Pemasok

Gambar 3.41 adalah *interface* rancangan halaman pemasok. Menampilkan data pemasok dan tombol tambah data. Terdapat juga kotak pencarian.

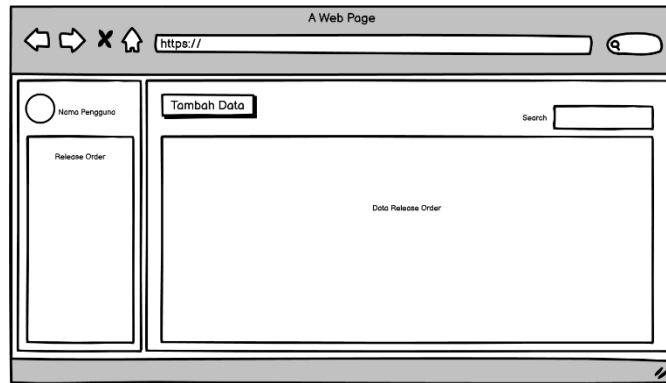
7. *Interface* Halaman Permintaan



Gambar 3.42 Interface Halaman Permintaan

Gambar 3.42 adalah *interface* rancangan halaman permintaan. Menampilkan data permintaan dan tombol tambah data. Terdapat juga kotak pencarian.

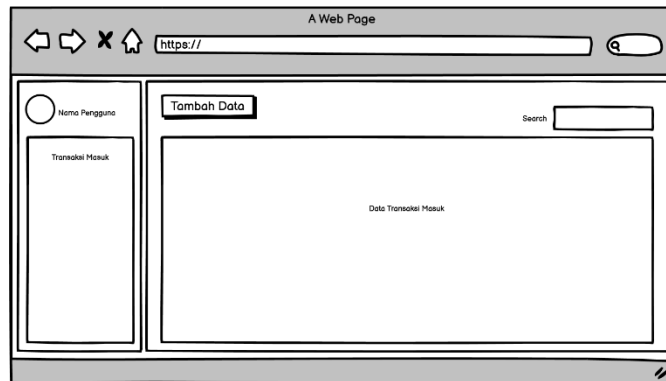
8. *Interface* Halaman Release Order



Gambar 3.43 Interface Halaman Release Order

Gambar 3.43 adalah *interface* rancangan halaman *release order*. Menampilkan data *release order* dan tombol tambah data. Terdapat juga kotak pencarian.

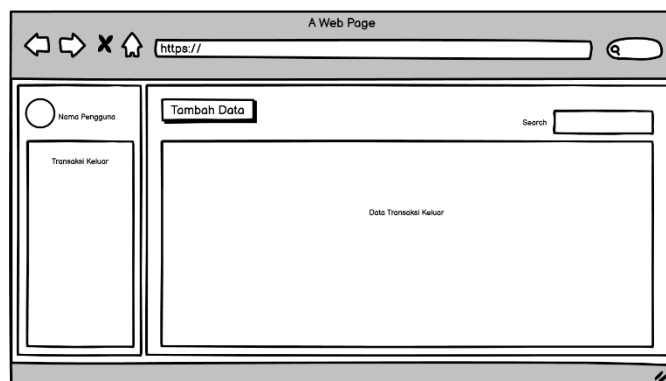
9. Interface Halaman Transaksi Masuk



Gambar 3.44 Interface Halaman Transaksi Masuk

Gambar 3.44 adalah *interface* rancangan halaman transaksi masuk. Menampilkan data transaksi masuk dan tombol tambah data. Terdapat juga kotak pencarian.

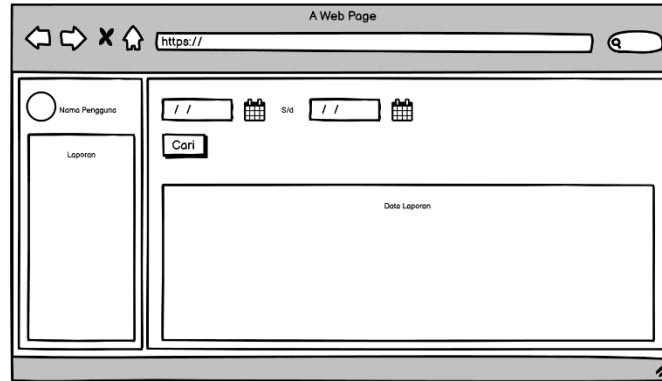
10. Interface Halaman Transaksi Keluar



Gambar 3.45 Interface Halaman Transaksi Keluar

Gambar 3.45 adalah *interface* rancangan halaman transaksi keluar. Menampilkan data transaksi keluar dan tombol tambah data. Terdapat juga kotak pencarian.

11. Interface Halaman Laporan



Gambar 3.46 Interface Halaman Laporan

Gambar 3.46 adalah *interface* rancangan halaman laporan. Menampilkan tanggal atau periode waktu yang digunakan untuk menyeleksi laporan berdasarkan waktu.

3.7 Testing Aplikasi

Pada tahap ini akan dilakukan pengujian terhadap sistem yang telah dibangun, pada kasus ini digunakan 2 metode pengujian yaitu *Black Box* dan pengujian UAT (*User Acceptance Test*).

3.7.1 Pengujian *Black Box*

Pengujian *Black Box* merupakan metode *testing* atau pengujian yang dilakukan dengan mengoperasikan atau mengeksekusi modul atau unit, kemudian mengamati apakah hasil dari modul tersebut memenuhi proses atau hasil yang diinginkan atau diharapkan.

1. Pengujian *Form Login*

Tabel 4.1 Tabel Pengujian *Form Login*

No	Skenario Pengujian	Hasil yang diharapkan
1	Ketika <i>field username</i> dan <i>field password</i> tidak di isi kemudian pencet tombol " <i>Log in</i> "	Muncul pesan peringatan " <i>Username dan Password Harus Diisi</i> "
2	Ketika <i>field username</i> di isi dan <i>field password</i>	Muncul pesan peringatan " <i>Password Harus Diisi</i> "

	dikosongkan lalu klik tombol “ <i>Log in</i> ”	
3	Ketika field username dikosongkan dan field password diisi lalu klik tombol “ <i>Log in</i> ”	Muncul pesan peringatan “ <i>Username</i> Harus Diisi”
4	Ketika <i>username</i> dan <i>password</i> tidak diisi dengan benar kemudian klik tombol “ <i>Log in</i> ”	Muncul pesan peringatan “ <i>Username</i> atau <i>Password</i> tidak Sesuai”
5	Ketika <i>username</i> dan <i>password</i> diisi dengan benar lalu klik tombol “ <i>Log in</i> ”	Muncul pesan “Selamat Datang (nama <i>user</i>)”

2. Pengujian *Form* Lokasi

Tabel 4.2 Tabel Pengujian *Form* Lokasi

No	Skenario Pengujian	Hasil yang diharapkan
1	Kondisi tambah lokasi semua <i>field</i> kosong, lalu tekan tombol simpan	Muncul pesan peringatan “ <i>The</i> nama lokasi <i>field is required</i> ” dan “ <i>The</i> deskripsi <i>field is required</i> ”
2	Kondisi semua <i>field</i> terisi, lalu tekan tombol simpan	Muncul pesan “Sukses Tambah Data”
3	Kondisi ketika klik tombol edit dan <i>field</i> data lokasi diisi atau diubah, lalu klik tombol simpan	Muncul pesan “Sukses Perbarui Data”

4	Ketika ingin menghapus data lokasi, pilih data yang ingin dihapus, kemudian klik tombol hapus	Muncul pesan “Apakah yakin? Ingin menghapus data ini...” ketika ingin menghapus data yang dipilih, lalu muncul pesan “Sukses <i>Delete</i> Data” saat memencet tombol hapus
5	Ketika ingin mencari data lokasi lalu klik tombol cari dan mengisi kata pencarian.	Muncul hasil pencarian data lokasi

3. Pengujian *Form* Tipe Produk

Tabel 4.3 Tabel Pengujian *Form* Tipe Produk

No	Skenario Pengujian	Hasil yang diharapkan
1	Kondisi tambah tipe produk semua <i>field</i> kosong, lalu tekan tombol simpan	Muncul pesan peringatan “ <i>The</i> nama tipe <i>field is required</i> ”
2	Kondisi semua <i>field</i> terisi, lalu tekan tombol simpan	Muncul pesan “Sukses Tambah Data”
3	Kondisi ketika klik tombol edit dan <i>field</i> data tipe produk diisi atau diubah, lalu klik tombol simpan	Muncul pesan “Sukses Perbarui Data”
4	Ketika ingin menghapus data tipe produk, pilih data yang ingin dihapus, lalu klik tombol hapus	Muncul pesan “Apakah yakin? Ingin menghapus data ini...” ketika ingin menghapus data yang dipilih, lalu muncul pesan “Sukses <i>Delete</i> Data” saat memencet tombol hapus

5	Ketika ingin mencari data tipe produk lalu klik tombol cari dan mengisi kata pencarian.	Muncul hasil pencarian data tipe produk
---	---	---

4. Pengujian *Form Material*

Tabel 4.4 Tabel Pengujian *Form Material*

No	Skenario Pengujian	Hasil yang diharapkan
1	Kondisi tambah material semua <i>field</i> kosong, lalu tekan tombol simpan	Muncul pesan peringatan “ <i>The nama material field is required</i> ”, “ <i>The id tipe field is required</i> ” dan “ <i>The stok minimum field is required</i> ”
2	Kondisi semua <i>field</i> terisi, lalu tekan tombol simpan	Muncul pesan “Sukses Tambah Data”
3	Kondisi ketika klik tombol edit dan <i>field</i> data material diisi atau diubah, lalu klik tombol simpan	Muncul pesan “Sukses Perbarui Data”
4	Ketika ingin menghapus data material, pilih data yang ingin dihapus, lalu klik tombol hapus	Muncul pesan “Apakah yakin? Ingin menghapus data ini...” ketika ingin menghapus data yang dipilih, lalu muncul pesan “Sukses <i>Delete</i> Data” saat memencet tombol hapus
5	Ketika ingin mencari data material lalu klik tombol cari dan mengisi kata pencarian.	Muncul hasil pencarian data material

5. Pengujian *Form* Pemasok

Tabel 4. 5 Tabel Pengujian *Form* Pemasok

No	Skenario Pengujian	Hasil yang diharapkan
1	Kondisi tambah pemasok semua <i>field</i> kosong, lalu tekan tombol simpan	Muncul pesan peringatan “ <i>The nama pemasok field is required</i> ”
2	Kondisi semua <i>field</i> terisi, lalu tekan tombol simpan	Muncul pesan “Sukses Tambah Data”
3	Kondisi ketika klik tombol edit dan <i>field</i> data pemasok diisi atau diubah, lalu klik tombol simpan	Muncul pesan “Sukses Perbarui Data”
4	Ketika ingin menghapus data pemasok, pilih data yang ingin dihapus, lalu klik tombol hapus	Muncul pesan “Apakah yakin? Ingin menghapus data ini...” ketika ingin menghapus data yang dipilih, lalu muncul pesan “Sukses <i>Delete</i> Data” saat memencet tombol hapus
5	Ketika ingin mencari data pemasok lalu klik tombol cari dan mengisi kata pencarian.	Muncul hasil pencarian data pemasok

6. Pengujian *Form* Permintaan

Tabel 4.6 Tabel Pengujian *Form* Permintaan

No	Skenario Pengujian	Hasil yang diharapkan
----	--------------------	-----------------------

1	Kondisi tambah permintaan semua <i>field</i> kosong, lalu tekan tombol simpan	Muncul pesan peringatan “ <i>The id material field is required</i> ”, “ <i>The jumlah permintaan field is required</i> ” dan “ <i>The tgl permintaan field is required</i> ”
2	Kondisi semua <i>field</i> terisi, lalu tekan tombol simpan	Muncul pesan “Sukses Tambah Data”
3	Kondisi ketika klik tombol edit dan <i>field</i> data permintaan diisi atau diubah, lalu klik tombol simpan	Muncul pesan “Sukses Perbarui Data”
4	Ketika ingin menghapus data permintaan, pilih data yang ingin dihapus, lalu klik tombol hapus	Muncul pesan “Apakah yakin? Ingin menghapus data ini...” ketika ingin menghapus data yang dipilih, lalu muncul pesan “Sukses <i>Delete</i> Data” saat memencet tombol hapus
5	Ketika ingin mencari data permintaan lalu klik tombol cari dan mengisi kata pencarian.	Muncul hasil pencarian data permintaan

7. Pengujian *Form Release Order*

Tabel 4.7 Tabel Pengujian *Form Release Order*

No	Skenario Pengujian	Hasil yang diharapkan
1	Kondisi tambah release order semua <i>field</i> kosong, lalu tekan tombol simpan	Muncul pesan peringatan “ <i>The no po field is required</i> ”, “ <i>The id pemasok field is required</i> ”, “ <i>The id material field is required</i> ”, “ <i>The jumlah ro field is required</i> ” dan “ <i>The tgl kirim field is required</i> ”

2	Kondisi semua <i>field</i> terisi, lalu tekan tombol simpan	Muncul pesan “Sukses Tambah Data”
3	Kondisi ketika klik tombol edit dan <i>field</i> data <i>release order</i> diisi atau diubah, lalu klik tombol simpan	Muncul pesan “Sukses Perbarui Data”
4	Ketika ingin menghapus data <i>release order</i> , pilih data yang ingin dihapus, kemudian klik tombol hapus	Muncul pesan “Apakah yakin? Ingin menghapus data ini...” ketika ingin menghapus data yang dipilih, lalu muncul pesan “Sukses <i>Delete</i> Data” saat memencet tombol hapus
5	Ketika ingin mencari data <i>release order</i> lalu klik tombol cari dan mengisi kata pencarian.	Muncul hasil pencarian data <i>release order</i>

8. Pengujian *Form* Transaksi Masuk

Tabel 4.8 Tabel Pengujian *Form* Transaksi Masuk

No	Skenario Pengujian	Hasil yang diharapkan
1	Kondisi tambah data semua <i>field</i> kosong, lalu tekan tombol simpan	Muncul pesan peringatan “ <i>The tgl masuk field is required</i> ”, “ <i>The id material field is required</i> ”, “ <i>The jumlah masuk field is required</i> ”, “ <i>The no po field is required</i> ” dan “ <i>The id lokasi field is required</i> ”
2	Kondisi semua <i>field</i> terisi, lalu tekan tombol simpan	Muncul pesan “Sukses Tambah Data”

3	Kondisi ketika klik tombol edit dan <i>field</i> data transaksi masuk diisi atau diubah, lalu klik tombol simpan	Muncul pesan “Sukses Perbarui Data”
4	Ketika ingin menghapus data transaksi masuk, pilih data yang ingin dihapus, kemudian klik tombol hapus	Muncul pesan “Apakah yakin? Ingin menghapus data ini...” ketika ingin menghapus data yang dipilih, lalu muncul pesan “Sukses <i>Delete</i> Data” saat memencet tombol hapus
5	Ketika ingin mencari data transaksi masuk lalu klik tombol cari dan mengisi kata pencarian.	Muncul hasil pencarian data user

9. Pengujian *Form* Transaksi Keluar

Tabel 4.9 Tabel Pengujian *Form* Transaksi Keluar

No	Skenario Pengujian	Hasil yang diharapkan
1	Kondisi tambah data semua field kosong, lalu tekan tombol simpan	Muncul pesan peringatan “ <i>The tgl keluar field is required</i> ”, “ <i>The id material field is required</i> ”, “ <i>The jumlah keluar field is required</i> ”, “ <i>The no po field is required</i> ” dan “ <i>The id lokasi field is required</i> ”
2	Kondisi semua field terisi, lalu tekan tombol simpan	Muncul pesan “Sukses Tambah Data”
3	Kondisi ketika klik tombol edit dan <i>field</i> data transaksi keluar diisi atau diubah, lalu klik tombol simpan	Muncul pesan “Sukses Perbarui Data”

4	Ketika ingin menghapus data transaksi keluar, pilih data yang ingin dihapus, kemudian klik tombol hapus	Muncul pesan “Apakah yakin? Ingin menghapus data ini...” ketika ingin menghapus data yang dipilih, lalu muncul pesan “Sukses <i>Delete Data</i> ” saat memencet tombol hapus
5	Ketika ingin mencari data transaksi keluar lalu klik tombol cari dan mengisi kata pencarian.	Muncul hasil pencarian data transaksi keluar

10. Pengujian *Form User*

Tabel 4.10 Tabel Pengujian *Form User*

No	Skenario Pengujian	Hasil yang diharapkan
1	Kondisi tambah user semua field kosong, lalu tekan tombol simpan	Muncul pesan peringatan “ <i>The nama user field is required</i> ”, “ <i>The username field is required</i> ” dan “ <i>The password field is required</i> ”
2	Kondisi semua field terisi, lalu tekan tombol simpan	Muncul pesan “Sukses Tambah Data”
3	Kondisi ketika klik tombol edit dan field data user diisi atau diubah, lalu klik tombol simpan	Muncul pesan “Sukses Perbarui Data”
4	Ketika ingin menghapus data user, pilih data yang ingin dihapus, kemudian klik tombol hapus	Muncul pesan “Apakah yakin? Ingin menghapus data ini...” ketika ingin menghapus data yang dipilih, lalu muncul pesan “Sukses <i>Delete Data</i> ” saat memencet tombol hapus
5	Ketika ingin mencari data user lalu klik tombol cari dan mengisi kata	Muncul hasil pencarian data user

	pencarian.	
--	------------	--

11. Pengujian *Form* Laporan

Tabel 4.11Tabel Pengujian *Form* Laporan

No	Skenario Pengujian	Hasil yang diharapkan
1	Kondisi periode tanggal semua <i>field</i> kosong, lalu tekan tombol cari	Muncul pesan peringatan “ <i>Please fill out this field</i> ” atau tidak ada pencarian yang di proses.

3.7.2 Pengujian UAT (*User Acceptance Test*)

Metode UAT digunakan pada tahap pengujian atau *testing* untuk memperoleh respon dari responden (*user*) terhadap sistem yang akan diimplementasikan dengan menggunakan skala *likert*, yang biasanya digunakan dalam penelitian dalam bentuk survei dan mengajukan pertanyaan kepada responden (*user*). Tabel dibawah merupakan level yang bisa dipilih sebagai jawaban atas pertanyaan-pertanyaan yang diberikan:

Jawaban	Bobot
A. Sangat Sesuai	5
B. Sesuai	4
C. Kurang Sesuai	3
D. Tidak Sesuai	2
E. Sangat Tidak Sesuai	1

Tabel 3.7 Bobot Nilai Jawaban

BAB IV

HASIL DAN PEMBAHASAN

4.1 Implementasi

Implementasi mengacu pada perilaku atau aktivitas yang cermat, detail (matang), dan tahap lanjutan dari desain sistem yang akan dilakukan. Termasuk desain program untuk menjadi salah satu operasi terbaik sesuai dengan kebutuhan, adalah dengan mengimplementasikan desain antarmuka sebagai bentuk halaman muka, dan berbagai aplikasi yang akan digunakan di masa depan.

4.1.1 Implementasi Antar Muka Sistem

Implementasi antar muka sistem ialah sebuah tampilan dari website sistem informasi logistik yang sudah di buat sebelumnya dan nantinya di implementasi antar muka ini akan dijelaskan tentang fitur-fitur yang terdapat pada sistem informasi logistik yang telah dibuat.

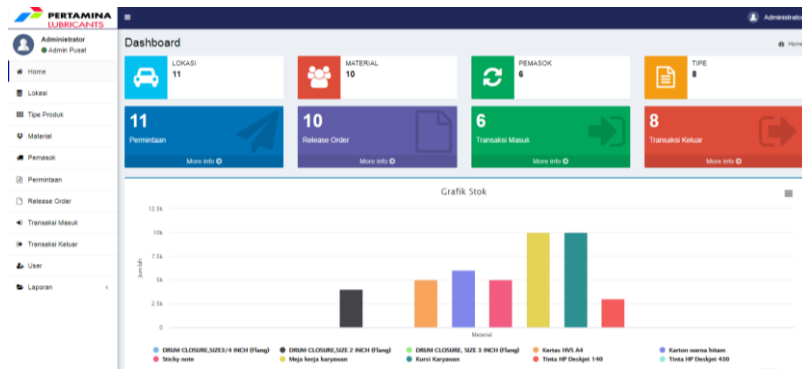
1. Tampilan Halaman *Login*



Gambar 4.1 Tampilan Halaman *Login*

Gambar 4.1 adalah halaman pertama ketika pengguna membuka aplikasi ini. Tampilan utamanya berupa *form login*, jika ingin memasuki sistem maka harus menginput *username* dan *password* terlebih dahulu dengan akun tiap-tiap *user* yang telah dibikin oleh admin pusat.

2. Tampilan Halaman *Dashboard*



Gambar 4.2 Tampilan Halaman *Dashboard*

Gambar 4.2 adalah halaman *dashboard* ketika *user* masuk ke dalam aplikasi ini. Tampilan menu nantinya akan menyesuaikan dari jenis *user* yang masuk. Misalnya jenis *user* admin pusat akan menampilkan sesuai otoritasnya saja, dan begitu juga dengan jenis *user* lainnya berdasarkan *use case* diagramnya. Halaman *dashboard* juga menyediakan informasi jumlah stok barang yang tersedia saat ini dengan menampilkan grafik dari stok barang yang ada saat ini.

3. Tampilan Halaman Lokasi

No	Nama Lokasi	Deskripsi	Aksi
1	A1.01	Rak A Lantai 1 nomor 01	Tambah Hapus
2	A1.02	Rak A Lantai 1 nomor 02	Tambah Hapus
3	A1.03	Rak A Lantai 1 nomor 03	Tambah Hapus
4	A1.04	Rak A Lantai 1 nomor 04	Tambah Hapus
5	A1.05	Rak A Lantai 1 nomor 05	Tambah Hapus
6	B1.01	Rak B Lantai 1 nomor 01	Tambah Hapus
7	B1.02	Rak B Lantai 1 nomor 02	Tambah Hapus
8	B1.03	Rak B Lantai 1 nomor 03	Tambah Hapus
9	B1.04	Rak B Lantai 1 nomor 04	Tambah Hapus

Gambar 4.3 Tampilan Halaman Lokasi

Gambar 4.3 adalah halaman lokasi. Admin pusat mengoperasikan menu lokasi untuk mencatat dan menampilkan data lokasi penempatan barang berada.

4. Tampilan Halaman Tipe Produk

No	Nama Tipe	Aksi
1	Drum	Edit Delete
2	Karton	Edit Delete
3	Sticker & Label	Edit Delete
4	Meja	Edit Delete
5	Kursi	Edit Delete
6	Kertas A4	Edit Delete
7	Tinta Printer Warna Hitam	Edit Delete
8	Tinta Printer Warna Wami	Edit Delete

Gambar 4.4 Tampilan Halaman Tipe Produk

Gambar 4.4 adalah halaman tipe produk. Admin pusat mengoperasikan menu tipe produk untuk mencatat dan menampilkan tipe-tipe dari produk apa saja yang terdapat pada PT. Pertamina Lubricants.

5. Tampilan Halaman Material

No	Nama Material	Tipe Produk	Stok Minimum	Aksi
1	DRUM CLOSURE, SIZE 3/4 INCH (Flang)	Drum	100	Edit Delete
2	DRUM CLOSURE, SIZE 2 INCH (Flang)	Drum	100	Edit Delete
3	DRUM CLOSURE, SIZE 3 INCH (Flang)	Drum	100	Edit Delete
4	Kertas HVS A4	Kertas A4	5000	Edit Delete
5	Karton warna hitam	Karton	500	Edit Delete
6	Sticky note	Sticker & Label	1000	Edit Delete
7	Meja kerja karyawan	Meja	50	Edit Delete
8	Kursi Karyawan	Kursi	100	Edit Delete
9	Tinta HP Deskjet 140	Tinta Printer Warna Hitam	500	Edit Delete
10	Tinta HP Deskjet 150	Tinta Printer Warna Hitam	500	Edit Delete

Gambar 4.5 Tampilan Halaman Material

Gambar 4.5 adalah halaman material. Admin pusat dan staf produksi mengoperasikan menu material untuk menampilkan informasi data material dan juga stok minimum dari material tersebut.

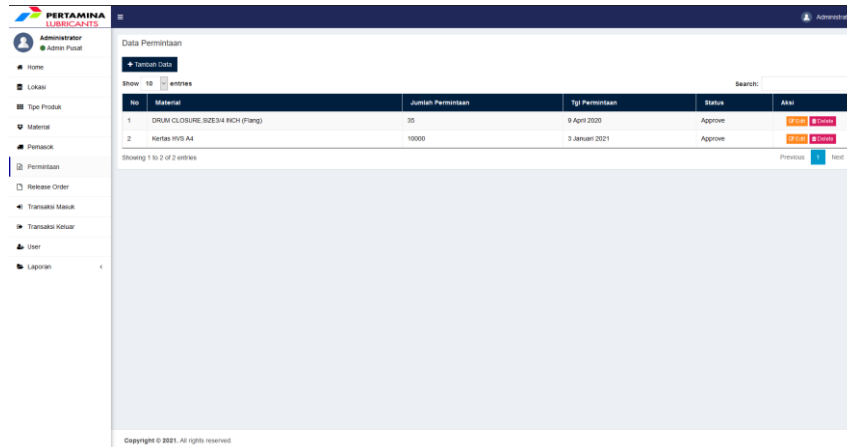
6. Tampilan Halaman Pemasok

No	Nama Pemasok	Aksi
1	PT. PRIMASINDO	Edit Delete
2	PT. ABADI PLASTIK	Edit Delete
3	PT. PURABARUTAMA	Edit Delete
4	PT. INDOJAYA	Edit Delete
5	PT. SEJAHTERAINDO	Edit Delete
6	PT. MERAKPUTIH	Edit Delete

Gambar 4.6 Tampilan Halaman Pemasok

Gambar 4.6 adalah halaman pemasok. Admin pusat mengoperasikan menu pemasok untuk menampilkan data perusahaan yang memasok material yang berkaitan dengan masuknya barang material.

7. Tampilan Halaman Permintaan

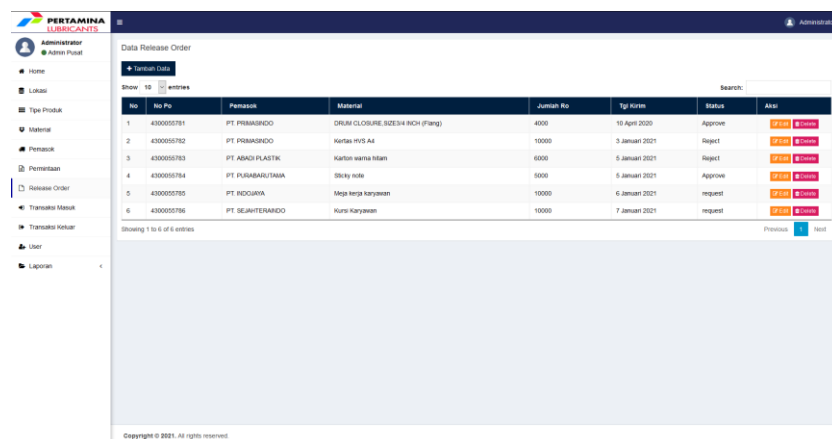


No	Material	Jumlah Permintaan	Tgl Permintaan	Status	Aksi
1	DRUM CLOSURE, SIZE 3/4 INCH (Pang)	35	9 April 2020	Approve	Detail Cancel
2	Kertas HVS A4	10000	3 Januari 2021	Approve	Detail Cancel

Gambar 4.7 Tampilan Halaman Permintaan

Gambar 4.7 adalah halaman permintaan. Staf produksi mengoperasikan menu permintaan jika ingin melakukan permintaan barang atau *request production*, pada halaman ini juga terdapat status barang apakah di *approve* atau di *reject* oleh *warehouse manager*, maka perlu dilakukan verifikasi terhadap barang yang akan masuk terlebih dahulu oleh *warehouse manager*.

8. Tampilan Halaman Release Order



No	No PO	Pemasok	Material	Jumlah RO	Tgl Krmn	Status	Aksi
1	4300005781	PT PRIMASINDO	DRUM CLOSURE, SIZE 3/4 INCH (Pang)	4000	10 April 2020	Approve	Detail Cancel
2	4300005782	PT PRIMASINDO	Kertas HVS A4	10000	3 Januari 2021	Reject	Detail Cancel
3	4300005783	PT ABADI PLASTIK	Karton warna hitam	6000	5 Januari 2021	Reject	Detail Cancel
4	4300005784	PT PURABASULTAMA	Sticky note	6000	5 Januari 2021	Approve	Detail Cancel
5	4300005785	PT REDJURA	Meja kerja karyawan	10000	6 Januari 2021	request	Detail Cancel
6	4300005786	PT SEJANTENARIDO	Kursi Karyawan	10000	7 Januari 2021	request	Detail Cancel

Gambar 4.8 Tampilan Halaman Release Order

Gambar 4.8 adalah halaman *release order*. Staf penerimaan mengoperasikan menu *release order* untuk melakukan pendataan terhadap barang yang akan diterima, pada halaman ini juga terdapat status barang apakah di *approve* atau di *reject* oleh

warehouse manager, maka perlu dilakukan verifikasi terhadap barang yang akan masuk terlebih dahulu oleh *warehouse manager*.

9. Tampilan Halaman Transaksi Masuk

No	Tgl Masuk	Material	Jumlah Masuk	No Po	Lokasi	Aksi
1	10 April 2020	DRUM CLOSURE,SIZE 2 INCH (Flang)	4000	4300055781	A1.01	[Edit] [Delete]
2	4 Januari 2021	Kertas HVS A4	10000	4300055782	A1.02	[Edit] [Delete]
3	6 Januari 2021	Karton warna hitam	6000	4300055783	A1.03	[Edit] [Delete]
4	6 Januari 2021	Sticky note	5000	4300055784	A1.04	[Edit] [Delete]
5	7 Januari 2021	Meja kerja karyawan	10000	4300055785	A1.05	[Edit] [Delete]
6	8 Januari 2021	Kursi Karyawan	10000	4300055786	A1.05	[Edit] [Delete]
7	9 Januari 2021	Tinta HP Deskjet 140	3000	4300055787	A1.03	[Edit] [Delete]

Gambar 4.9 Tampilan Halaman Transaksi Masuk

Gambar 4.9 adalah halaman transaksi masuk. Admin *inventory* mengoperasikan menu transaksi masuk ketika ada transaksi barang yang akan masuk. Halaman ini juga mencatat data material serta jumlah material yang masuk dan secara otomatis akan menambahkan kedalam stok material.

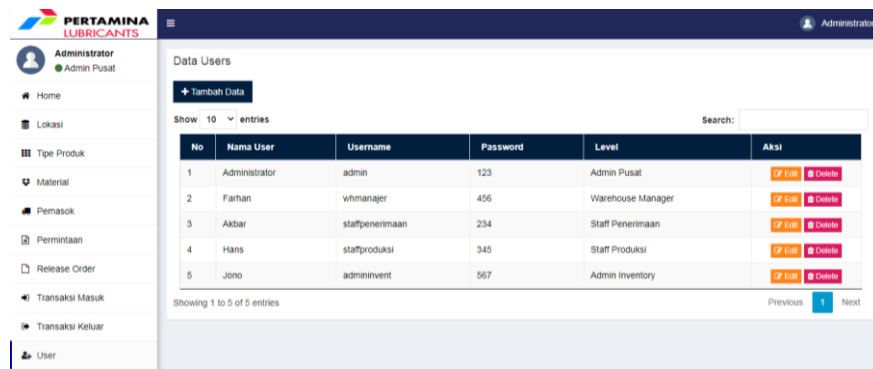
10. Tampilan Halaman Transaksi Keluar

No	Tgl Keluar	Material	Jumlah Keluar	No Po	Lokasi	Aksi
1	9 April 2020	DRUM CLOSURE,SIZE3/4 INCH (Flang)	90	4300055781	B1.05	[Edit] [Delete]
2	5 Januari 2021	Kertas HVS A4	5000	4300055782	B1.01	[Edit] [Delete]

Gambar 4.10 Tampilan Halaman Transaksi Keluar

Gambar 4.10 adalah halaman transaksi keluar. Admin *inventory* mengoperasikan menu transaksi keluar ketika ada transaksi barang keluar. Pada halaman ini mencatat material serta jumlah material yang masuk dan secara otomatis akan mengurangi jumlah stok material.

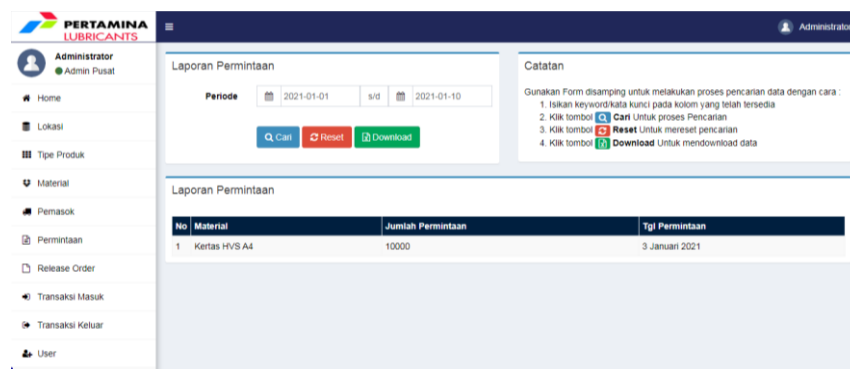
11. Tampilan Halaman Tambah User



Gambar 4.11 Tampilan Halaman User

Gambar 4.11 adalah halaman *user*. Admin pusat mengoperasikan menu *user* untuk mengelola data *user*. Mulai dari menambah data, mengedit serta menghapus data *user*.

12. Tampilan Halaman Laporan



Gambar 4.12 Tampilan Halaman Laporan

Gambar 4.12 merupakan halaman laporan. Pada halaman ini digunakan untuk melihat laporan yang ada. Seperti laporan permintaan, transaksi barang masuk dan keluar dan laporan *release order*. Pada halaman ini menampilkan pilihan periode laporan yang akan dilihat sesuai kebutuhan. Menu laporan hanya bisa digunakan oleh *warehouse manager* dan *admin inventory*

13. Tampilan Halaman Laporan Stok

No	Material	Stock Min	Jumlah Masuk	Jumlah Keluar	Stock
1	DRUM CLOSURE SIZE 3/4 INCH (Piang)	100			90
2	DRUM CLOSURE SIZE 2 INCH (Piang)	100			4000
3	DRUM CLOSURE SIZE 3 INCH (Piang)	100			0
4	Kertas HVS A4	5000	10000	5000	5000
5	Kertas warna Hitam	500	6000		6000
6	Sticky note	1000	5000		5000
7	Mega kerja karyawan	50	10000		10000
8	Kursi Karyawan	100	10000		10000
9	Titik HP Desktop 140	500	3000		3000
10	Titik HP Desktop 430	200			0

Gambar 4.13 Tampilan Halaman Laporan Stok

Gambar 4.13 merupakan halaman laporan stok. Menampilkan data material dan jumlah stok saat ini, juga merupakan alarm berupa baris yang ditandai dengan warna pastel jika terdapat stok yang kosong atau sudah dibawah angka minimum stok seharusnya. Menu laporan stok hanya bisa digunakan oleh *warehouse manager* dan admin *inventory*.

4.2 Hasil Pengujian Black Box

Pada tahapan ini, peneliti menguji sistem yang sudah dibuat, pada kasus ini pengujian akan melakukan pengujian menggunakan metode *Black Box*. Berikut adalah hasil pengujian *Black Box* yang telah dilakukan.

1. Pengujian Form Login

Tabel 4. 1 Tabel Pengujian Form Login

No	Skenario Pengujian	Hasil yang diharapkan	Kesimpulan
1	Ketika <i>field username</i> dan <i>field password</i> tidak di isi kemudian pencet tombol “Log in”	Muncul pesan peringatan “Username dan Password Harus Diisi”	Valid
2	Ketika <i>field username</i> di isi dan <i>field password</i> dikosongkan lalu klik tombol “Log in”	Muncul pesan peringatan “Password Harus Diisi”	Valid
3	Ketika <i>field username</i> dikosongkan dan <i>field password</i> di isi lalu klik	Muncul pesan peringatan “Username Harus Diisi”	Valid

	tombol “ <i>Log in</i> ”		
4	Ketika <i>username</i> dan <i>password</i> tidak diisi dengan benar kemudian klik tombol “ <i>Log in</i> ”	Muncul pesan peringatan “ <i>Username</i> atau <i>Password</i> tidak Sesuai”	Valid
5	Ketika <i>username</i> dan <i>password</i> di isi benar lalu klik tombol “ <i>Log in</i> ”	Muncul pesan “Selamat Datang (nama <i>user</i>)”	Valid

2. Pengujian *Form* Lokasi

Tabel 4. 2 Tabel Pengujian *Form* Lokasi

No	Skenario Pengujian	Hasil yang diharapkan	Kesimpulan
1	Kondisi tambah lokasi semua <i>field</i> kosong, lalu tekan tombol simpan	Muncul pesan peringatan “ <i>The</i> nama lokasi <i>field is required</i> ” dan “ <i>The</i> deskripsi <i>field is required</i> ”	Valid
2	Kondisi semua <i>field</i> terisi, lalu tekan tombol simpan	Muncul pesan “Sukses Tambah Data”	Valid
3	Kondisi ketika klik tombol edit dan <i>field</i> data lokasi diisi atau diubah, lalu klik tombol simpan	Muncul pesan “Sukses Perbarui Data”	Valid
4	Ketika ingin menghapus data lokasi, pilih data yang ingin dihapus, kemudian klik tombol hapus	Muncul pesan “Apakah yakin? Ingin menghapus data ini...” ketika ingin menghapus data yang dipilih, lalu muncul pesan “Sukses <i>Delete</i> Data” saat memencet tombol hapus	Valid
5	Ketika ingin mencari data lokasi lalu klik tombol cari dan mengisi kata	Muncul hasil pencarian data lokasi	Valid

	pencarian.		
--	------------	--	--

3. Pengujian *Form* Tipe Produk

Tabel 4. 3 Tabel Pengujian *Form* Tipe Produk

No	Skenario Pengujian	Hasil yang diharapkan	Kesimpulan
1	Kondisi tambah tipe produk semua <i>field</i> kosong, lalu tekan tombol simpan	Muncul pesan peringatan “ <i>The nama tipe field is required</i> ”	Valid
2	Kondisi semua <i>field</i> terisi, lalu tekan tombol simpan	Muncul pesan “Sukses Tambah Data”	Valid
3	Kondisi ketika klik tombol edit dan <i>field</i> data tipe produk diisi atau diubah, lalu klik tombol simpan	Muncul pesan “Sukses Perbarui Data”	Valid
4	Ketika ingin menghapus data tipe produk, pilih data yang ingin dihapus, kemudian klik tombol hapus	Muncul pesan “Apakah yakin? Ingin menghapus data ini...” ketika ingin menghapus data yang dipilih, lalu muncul pesan “Sukses <i>Delete</i> Data” saat memencet tombol hapus	Valid
5	Ketika ingin mencari data tipe produk lalu klik tombol cari dan mengisi kata pencarian.	Muncul hasil pencarian data tipe produk	Valid

4. Pengujian *Form* Material

Tabel 4. 4 Tabel Pengujian *Form* Material

No	Skenario Pengujian	Hasil yang diharapkan	Kesimpulan
1	Kondisi tambah material semua <i>field</i> kosong, lalu tekan tombol simpan	Muncul pesan peringatan “ <i>The nama material field is required</i> ”, “ <i>The id tipe field is required</i> ” dan “ <i>The stok minimum field is required</i> ”	Valid
2	Kondisi semua <i>field</i> terisi, lalu tekan tombol simpan	Muncul pesan “Sukses Tambah Data”	Valid
3	Kondisi ketika klik tombol edit dan <i>field</i> data material diisi atau diubah, lalu klik tombol simpan	Muncul pesan “Sukses Perbarui Data”	Valid
4	Ketika ingin menghapus data material, pilih data yang ingin dihapus, kemudian klik tombol hapus	Muncul pesan “Apakah yakin? Ingin menghapus data ini...” ketika ingin menghapus data yang dipilih, lalu muncul pesan “Sukses <i>Delete</i> Data” saat memencet tombol hapus	Valid
5	Ketika ingin mencari data material lalu klik tombol cari dan mengisi kata pencarian.	Muncul hasil pencarian data material	Valid

5. Pengujian *Form* Pemasok

Tabel 4. 5 Tabel Pengujian *Form* Pemasok

No	Skenario Pengujian	Hasil yang diharapkan	Kesimpulan
1	Kondisi tambah pemasok semua <i>field</i> kosong, lalu tekan tombol simpan	Muncul pesan peringatan “ <i>The nama pemasok field is required</i> ”	Valid

2	Kondisi semua <i>field</i> terisi, lalu tekan tombol simpan	Muncul pesan “Sukses Tambah Data”	Valid
3	Kondisi ketika klik tombol edit dan <i>field</i> data pemasok diisi atau diubah, lalu klik tombol simpan	Muncul pesan “Sukses Perbarui Data”	Valid
4	Ketika ingin menghapus data pemasok, pilih data yang ingin dihapus, kemudian klik tombol hapus	Muncul pesan “Apakah yakin? Ingin menghapus data ini...” ketika ingin menghapus data yang dipilih, lalu muncul pesan “Sukses <i>Delete</i> Data” saat memencet tombol hapus	Valid
5	Ketika ingin mencari data pemasok lalu klik tombol cari dan mengisi kata pencarian.	Muncul hasil pencarian data pemasok	Valid

6. Pengujian *Form* Permintaan

Tabel 4. 6 Tabel Pengujian *Form* Permintaan

No	Skenario Pengujian	Hasil yang diharapkan	Kesimpulan
1	Kondisi tambah permintaan semua <i>field</i> kosong, lalu tekan tombol simpan	Muncul pesan peringatan “ <i>The id material field is required</i> ”, “ <i>The jumlah permintaan field is required</i> ” dan “ <i>The tgl permintaan field is required</i> ”	Valid
2	Kondisi semua <i>field</i> terisi, lalu tekan tombol simpan	Muncul pesan “Sukses Tambah Data”	Valid

3	Kondisi ketika klik tombol edit dan <i>field</i> data permintaan diisi atau diubah, lalu klik tombol simpan	Muncul pesan “Sukses Perbarui Data”	Valid
4	Ketika ingin menghapus data permintaan, pilih data yang ingin dihapus, kemudiani klik tombol hapus	Muncul pesan “Apakah yakin? Ingin menghapus data ini...” ketika ingin menghapus data yang dipilih, lalu muncul pesan “Sukses <i>Delete</i> Data” saat memencet tombol hapus	Valid
5	Ketika ingin mencari data permintaan lalu klik tombol cari dan mengisi kata pencarian.	Muncul hasil pencarian data permintaan	Valid

7. Pengujian Form Release Order

Tabel 4. 7 Tabel Pengujian Form Release Order

No	Skenario Pengujian	Hasil yang diharapkan	Kesimpulan
1	Kondisi tambah release order semua <i>field</i> kosong, lalu tekan tombol simpan	Muncul pesan peringatan “ <i>The no po field is required</i> ”, “ <i>The id pemasok field is required</i> ”, “ <i>The id material field is required</i> ”, “ <i>The jumlah ro field is required</i> ” dan “ <i>The tgl kirim field is required</i> ”	Valid
2	Kondisi semua <i>field</i> terisi, lalu tekan tombol simpan	Muncul pesan “Sukses Tambah Data”	Valid
3	Kondisi ketika klik tombol edit dan <i>field</i> data <i>release order</i> diisi atau diubah, lalu klik tombol	Muncul pesan “Sukses Perbarui Data”	Valid

	simpan		
4	Ketika ingin menghapus data <i>release order</i> , pilih data yang ingin dihapus, kemudian klik tombol hapus	Muncul pesan “Apakah yakin? Ingin menghapus data ini...” ketika ingin menghapus data yang dipilih, lalu muncul pesan “Sukses <i>Delete Data</i> ” saat memencet tombol hapus	Valid
5	Ketika ingin mencari data <i>release order</i> lalu klik tombol cari dan mengisi kata pencarian.	Muncul hasil pencarian data <i>release order</i>	Valid

8. Pengujian *Form Transaksi Masuk*

Tabel 4. 8 Tabel Pengujian *Form Transaksi Masuk*

No	Skenario Pengujian	Hasil yang diharapkan	Kesimpulan
1	Kondisi tambah data semua <i>field</i> kosong, lalu tekan tombol simpan	Muncul pesan peringatan “ <i>The tgl masuk field is required</i> ”, “ <i>The id material field is required</i> ”, “ <i>The jumlah masuk field is required</i> ”, “ <i>The no po field is required</i> ” dan “ <i>The id lokasi field is required</i> ”	Valid
2	Kondisi semua <i>field</i> terisi, lalu tekan tombol simpan	Muncul pesan “Sukses Tambah Data”	Valid
3	Kondisi ketika klik tombol edit dan <i>field</i> data transaksi masuk diisi atau diubah, lalu klik tombol simpan	Muncul pesan “Sukses Perbarui Data”	Valid

4	Ketika ingin menghapus data transaksi masuk, pilih data yang ingin dihapus, kemudian klik tombol hapus	Muncul pesan “Apakah yakin? Ingin menghapus data ini...” ketika ingin menghapus data yang dipilih, lalu muncul pesan “Sukses <i>Delete Data</i> ” saat memencet tombol hapus	Valid
5	Ketika ingin mencari data transaksi masuk lalu klik tombol cari dan mengisi kata pencarian.	Muncul hasil pencarian data transaksi masuk	Valid

9. Pengujian *Form Transaksi Keluar*

Tabel 4. 9 Tabel Pengujian *Form Transaksi Keluar*

No	Skenario Pengujian	Hasil yang diharapkan	Kesimpulan
1	Kondisi tambah data semua <i>field</i> kosong, lalu tekan tombol simpan	Muncul pesan peringatan “ <i>The tgl keluar field is required</i> ”, “ <i>The id material field is required</i> ”, “ <i>The jumlah keluar field is required</i> ”, “ <i>The no po field is required</i> ” dan “ <i>The id lokasi field is required</i> ”	Valid
2	Kondisi semua <i>field</i> terisi, lalu tekan tombol simpan	Muncul pesan “Sukses Tambah Data”	Valid
3	Kondisi ketika klik tombol edit dan <i>field</i> data transaksi keluar diisi atau diubah, lalu klik tombol simpan	Muncul pesan “Sukses Perbarui Data”	Valid
4	Ketika ingin menghapus data transaksi keluar, pilih data yang ingin dihapus, kemudian klik tombol hapus	Muncul pesan “Apakah yakin? Ingin menghapus data ini...” ketika ingin menghapus data yang dipilih, lalu muncul pesan “Sukses <i>Delete Data</i> ” saat memencet tombol hapus	Valid

5	Ketika ingin mencari data transaksi keluar lalu klik tombol cari dan mengisi kata pencarian.	Muncul hasil pencarian data transaksi keluar	Valid
---	--	--	-------

10. Pengujian *Form User*

Tabel 4. 10 Tabel Pengujian *Form User*

No	Skenario Pengujian	Hasil yang diharapkan	Kesimpulan
1	Kondisi tambah <i>user</i> semua <i>field</i> kosong, lalu tekan tombol simpan	Muncul pesan peringatan “ <i>The nama user field is required</i> ”, “ <i>The username field is required</i> ” dan “ <i>The password field is required</i> ”	Valid
2	Kondisi semua <i>field</i> terisi, lalu tekan tombol simpan	Muncul pesan “Sukses Tambah Data”	Valid
3	Kondisi ketika klik tombol edit dan <i>field</i> data user diisi atau diubah, lalu klik tombol Simpan	Muncul pesan “Sukses Perbarui Data”	Valid
4	Ketika ingin menghapus data <i>user</i> , pilih data yang ingin dihapus, kemudian klik tombol hapus	Muncul pesan “Apakah yakin? Ingin menghapus data ini...” ketika ingin menghapus data yang dipilih, lalu muncul pesan “Sukses <i>Delete</i> Data” saat memencet tombol hapus	Valid
5	Ketika ingin mencari data <i>user</i> lalu klik tombol cari dan mengisi kata pencarian.	Muncul hasil pencarian data <i>user</i>	Valid

11. Pengujian *Form Laporan*

Tabel 4. 11 Tabel Pengujian *Form* Laporan

No	Skenario Pengujian	Hasil yang diharapkan	Kesimpulan
1	Kondisi periode tanggal semua <i>field</i> kosong, lalu tekan tombol cari	Muncul pesan peringatan “ <i>Please fill out this field</i> ” atau tidak ada pencarian yang di proses.	Valid

4.3 Hasil Pengujian UAT (*User Acceptance Test*)

Pengujian UAT (*User Acceptance Testing*) ialah proses pengujian bagi *user* untuk menghasilkan dokumen yang membuktikan apakah sistem yang sedang dikembangkan dapat diterima oleh *user*, ketika hasil pengujian dianggap memenuhi kebutuhan *user* maka aplikasi dapat diterapkan. Tes UAT dilaksanakan dengan mengajukan total 25 pertanyaan terhadap orang yang sudah menggunakan sistem informasi logistik yang bertindak sebagai *user*, pengujian ini melibatkan 20 orang yang pernah menggunakan aplikasi ini dan beberapa diantaranya berperan melakukan *testing* secara langsung ke aplikasinya melalui *team viewers*.

Hasil UAT terbagi dalam 5 kategori yaitu SS (Sangat Sesuai), S (Sesuai), KS (Kurang Sesuai), TS (Tidak Sesuai) dan TJ (Tidak Jawab). Soal angket dijelaskan dalam lampiran formulir tes UAT beserta dengan dokumentasi dari *user* yang terlibat secara langsung dalam melakukan *testing* aplikasi. Berikut adalah *test case* dan hasil dari UAT yang telah dilakukan.

1. *Test Case Login*

Skenario	Langkah Tes	Hasil yang Diharapkan	Hasil Aktual
Login	1. Membuka aplikasi 2. Mengisi <i>username</i> dan <i>password</i> 3. Menekan tombol <i>login</i>	Menampilkan pesan “Selamat Datang (nama <i>user</i>)” kemudian masuk ke dalam aplikasi	Berhasil menampilkan pesan dan masuk ke dalam aplikasi

2. *Test Case Mengelola Data Lokasi*

Skenario	Langkah Tes	Hasil yang Diharapkan	Hasil Aktual
Menambah data lokasi	1. Membuka menu lokasi 2. Menekan tombol tambah data 3. Input data (nama lokasi dan deskripsi) 4. Menekan tombol <i>save</i>	Menampilkan pesan “Sukses Tambah Data” kemudian data tersimpan ke <i>database</i> dan ditampilkan pada tabel lokasi	Berhasil menampilkan pesan data lokasi berhasil ditambah dan tersimpan di <i>database</i> serta menampilkan datanya pada tabel lokasi.
Mengedit data lokasi	1. Membuka menu lokasi 2. Memilih data lokasi 3. Menekan tombol edit data 4. Input data perubahan 5. Menekan tombol <i>update</i>	Menampilkan pesan “Sukses Perbarui Data” kemudian data ter- <i>update</i> di <i>database</i> dan ditampilkan pada tabel lokasi	Berhasil menampilkan pesan data lokasi berhasil diperbarui dan tersimpan di <i>database</i> serta menampilkan data yang telah di- <i>update</i> pada tabel lokasi.
Menghapus data lokasi	1. Membuka menu lokasi 2. Memilih data lokasi 3. Menekan tombol <i>delete</i> data 4. Menekan tombol “ya saya yakin” (hapus data)	Menampilkan pesan “Apakah yakin? Ingin menghapus data ini...” ketika ingin menghapus data yang dipilih, lalu muncul pesan “Sukses <i>Delete</i> Data” saat menekan tombol hapus kemudian data	Berhasil menampilkan pesan <i>delete</i> dan <i>delete</i> data lokasi di <i>database</i> dan di tabel lokasi.

		terhapus dari <i>database</i>	
Mencari data lokasi	1. Membuka menu lokasi 2. Input kata pencarian	Menampilkan data hasil pencarian	Berhasil menampilkan hasil pencarian

3. Test Case Mengelola Data Tipe Produk

Skenario	Langkah Tes	Hasil yang Diharapkan	Hasil Aktual
Menambah data tipe produk	1. Membuka menu tipe produk 2. Menekan tombol tambah data 3. Input data (nama tipe produk) 4. Menekan tombol <i>save</i>	Menampilkan pesan “Sukses Tambah Data” kemudian data tersimpan ke <i>database</i> dan ditampilkan pada tabel tipe produk	Berhasil menampilkan pesan data tipe produk berhasil ditambah dan tersimpan di <i>database</i> serta menampilkan datanya pada tabel tipe produk.
Mengedit data tipe produk	1. Membuka menu tipe produk 2. Memilih data tipe produk 3. Menekan tombol edit data 4. Input data perubahan 5. Menekan tombol <i>update</i>	Menampilkan pesan “Sukses Perbarui Data” kemudian data ter- <i>update</i> di <i>database</i> dan ditampilkan pada tabel tipe produk	Berhasil menampilkan pesan data tipe produk berhasil diperbarui dan tersimpan di <i>database</i> serta menampilkan data yang telah di- <i>update</i> pada tabel tipe produk.
Menghapus data tipe produk	1. Membuka menu tipe produk	Menampilkan pesan “Apakah yakin? Ingin menghapus	Berhasil menampilkan pesan <i>delete</i> dan

	2. Memilih data tipe produk 3. Menekan tombol <i>delete</i> data 4. Menekan tombol “ya saya yakin” (hapus data)	data ini...” ketika ingin menghapus data yang dipilih, lalu muncul pesan “Sukses <i>Delete Data</i> ” saat menekan tombol hapus kemudian data terhapus dari <i>database</i>	<i>delete</i> data tipe produk di <i>database</i> dan di tabel tipe produk.
Mencari data tipe produk	1. Membuka menu tipe produk 2. Input kata pencarian	Menampilkan data hasil pencarian	Berhasil menampilkan hasil pencarian

4. Test Case Mengelola Data Material

Skenario	Langkah Tes	Hasil yang Diharapkan	Hasil Aktual
Menambah data material	1. Membuka menu material 2. Menekan tombol tambah data 3. Input data (nama material, tipe produk dan stok minimum) 4. Menekan tombol <i>save</i>	Menampilkan pesan “Sukses Tambah Data” kemudian data tersimpan ke <i>database</i> dan ditampilkan pada tabel data material	Berhasil menampilkan pesan data material berhasil ditambah dan tersimpan di <i>database</i> serta menampilkan datanya pada tabel data material.
Mengedit data material	1. Membuka menu material 2. Memilih data material 3. Menekan tombol edit data 4. Input data perubahan	Menampilkan pesan “Sukses Perbarui Data” kemudian data ter- <i>update</i> di <i>database</i> dan	Berhasil menampilkan pesan data material berhasil diperbarui dan

	5. Menekan tombol <i>update</i>	ditampilkan pada tabel data material	tersimpan di <i>database</i> serta menampilkan data yang telah di- <i>update</i> pada tabel data material.
Menghapus data material	1. Membuka menu material 2. Memilih data material 3. Menekan tombol <i>delete</i> data 4. Menekan tombol “ya saya yakin” (hapus data)	Menampilkan pesan “Apakah yakin? Ingin menghapus data ini...” ketika ingin menghapus data yang dipilih, lalu muncul pesan “Sukses <i>Delete</i> Data” saat menekan tombol hapus kemudian data terhapus dari <i>database</i>	Berhasil menampilkan pesan <i>delete</i> dan <i>delete</i> data material di <i>database</i> dan di tabel data material
Mencari data material	1. Membuka menu material 2. Input kata pencarian	Menampilkan data hasil pencarian	Berhasil menampilkan hasil pencarian

5. Test Case Mengelola Data Pemasok

Skenario	Langkah Tes	Hasil yang Diharapkan	Hasil Aktual
Menambah data pemasok	1. Membuka menu pemasok 2. Menekan tombol tambah data 3. Input data (nama pemasok)	Menampilkan pesan “Sukses Tambah Data” kemudian data tersimpan ke <i>database</i> dan	Berhasil menampilkan pesan data pemasok berhasil ditambah dan tersimpan di

	4. Menekan tombol <i>save</i>	ditampilkan pada tabel data pemasok	<i>database</i> serta menampilkan datanya pada tabel data pemasok
Mengedit data pemasok	1. Membuka menu pemasok 2. Memilih data pemasok 3. Menekan tombol edit data 4. Input data perubahan 5. Menekan tombol <i>update</i>	Menampilkan pesan “Sukses Perbarui Data” kemudian data ter- <i>update</i> di <i>database</i> dan ditampilkan pada tabel data pemasok	Berhasil menampilkan pesan data pemasok berhasil diperbarui dan tersimpan di <i>database</i> serta menampilkan data yang telah di- <i>update</i> pada tabel data pemasok
Menghapus data pemasok	1. Membuka menu pemasok 2. Memilih data pemasok 3. Menekan tombol <i>delete</i> data 4. Menekan tombol “ya saya yakin” (hapus data)	Menampilkan pesan “Apakah yakin? Ingin menghapus data ini...” ketika ingin menghapus data yang dipilih, lalu muncul pesan “Sukses <i>Delete</i> Data” saat menekan tombol hapus kemudian data terhapus dari <i>database</i>	Berhasil menampilkan pesan <i>delete</i> dan <i>delete</i> data pemasok di <i>database</i> dan di tabel data pemasok
Mencari data pemasok	1. Membuka menu pemasok 2. Input kata pencarian	Menampilkan data hasil pencarian	Berhasil menampilkan hasil pencarian

6. Test Case Mengelola Data Permintaan

Skenario	Langkah Tes	Hasil yang Diharapkan	Hasil Aktual
Menambah data permintaan	<ol style="list-style-type: none"> 1. Membuka menu permintaan 2. Menekan tombol tambah data 3. Input data (nama material, jumlah permintaan dan tgl permintaan) 4. Menekan tombol <i>save</i> 	Menampilkan pesan “Sukses Tambah Data” kemudian data tersimpan ke <i>database</i> dan ditampilkan pada tabel data permintaan	Berhasil menampilkan pesan data permintaan berhasil ditambah dan tersimpan di <i>database</i> serta menampilkan datanya pada tabel data permintaan.
Mengedit data permintaan	<ol style="list-style-type: none"> 1. Membuka menu permintaan 2. Memilih data permintaan 3. Menekan tombol edit data 4. Input data perubahan 5. Menekan tombol <i>update</i> 	Menampilkan pesan “Sukses Perbarui Data” kemudian data ter- <i>update</i> di <i>database</i> dan ditampilkan pada tabel data permintaan	Berhasil menampilkan pesan data permintaan berhasil diperbarui dan tersimpan di <i>database</i> serta menampilkan data yang telah di- <i>update</i> pada tabel data permintaan
Menghapus data permintaan	<ol style="list-style-type: none"> 1. Membuka menu permintaan 2. Memilih data permintaan 3. Menekan tombol <i>delete</i> data 4. Menekan tombol “ya saya yakin” (hapus data) 	Menampilkan pesan “Apakah yakin? Ingin menghapus data ini...” ketika ingin menghapus data yang dipilih, lalu muncul pesan	Berhasil menampilkan pesan <i>delete</i> dan <i>delete</i> data permintaan di <i>database</i> dan di

		“Sukses <i>Delete</i> Data” saat menekan tombol hapus kemudian data terhapus dari <i>database</i>	tabel data permintaan.
Mencari data permintaan	1. Membuka menu permintaan 2. Input kata pencarian	Menampilkan data hasil pencarian	Berhasil menampilkan hasil pencarian

7. Test Case Mengelola Release Order

Skenario	Langkah Tes	Hasil yang Diharapkan	Hasil Aktual
Menambah data <i>release order</i>	1. Membuka menu <i>release order</i> 2. Menekan tombol tambah data 3. Input data (No PO, nama pemasok, material, jumlah RO, tanggal kirim) 4. Menekan tombol <i>save</i>	Menampilkan pesan “Sukses Tambah Data” kemudian data tersimpan ke <i>database</i> dan ditampilkan pada tabel <i>release order</i>	Berhasil menampilkan pesan data <i>release order</i> berhasil ditambah dan tersimpan di <i>database</i> serta menampilkan datanya pada tabel <i>release order</i> .
Mengedit data <i>release order</i>	1. Membuka menu <i>release order</i> 2. Memilih data <i>release order</i> 3. Menekan tombol edit data 4. Input data perubahan 5. Menekan tombol <i>update</i>	Menampilkan pesan “Sukses Perbarui Data” kemudian data ter- <i>update</i> di <i>database</i> dan ditampilkan pada tabel <i>release order</i>	Berhasil menampilkan pesan data <i>release order</i> berhasil diperbarui dan tersimpan di <i>database</i> serta menampilkan

			data yang telah di- <i>update</i> pada tabel <i>release order</i> .
Menghapus data <i>release order</i>	1. Membuka menu <i>release order</i> 2. Memilih data <i>release order</i> 3. Menekan tombol <i>delete</i> data 4. Menekan tombol “ya saya yakin” (hapus data)	Menampilkan pesan “Apakah yakin? Ingin menghapus data ini...” ketika ingin menghapus data yang dipilih, lalu muncul pesan “Sukses <i>Delete Data</i> ” saat menekan tombol hapus kemudian data terhapus dari <i>database</i>	Berhasil menampilkan pesan <i>delete</i> dan <i>delete</i> data <i>release order</i> di <i>database</i> dan di tabel <i>release order</i> .
Mencari data <i>release order</i>	1. Membuka menu <i>release order</i> 2. Input kata pencarian	Menampilkan data hasil pencarian	Berhasil menampilkan hasil pencarian

8. Test Case Mengelola Transaksi Masuk

Skenario	Langkah Tes	Hasil yang Diharapkan	Hasil Aktual
Menambah data transaksi masuk	1. Membuka menu transaksi masuk 2. Menekan tombol tambah data 3. Input data (tanggal masuk, material, jumlah masuk, No PO, nama lokasi) 4. Menekan tombol <i>save</i>	Menampilkan pesan “Sukses Tambah Data” kemudian data tersimpan ke <i>database</i> dan ditampilkan pada tabel transaksi masuk	Berhasil menampilkan pesan data transaksi masuk berhasil ditambah dan tersimpan di <i>database</i> serta menampilkan datanya pada

			tabel transaksi masuk.
Mengedit data transaksi masuk	<ol style="list-style-type: none"> 1. Membuka menu transaksi masuk 2. Memilih data transaksi masuk 3. Menekan tombol edit data 4. Input data perubahan 5. Menekan tombol <i>update</i> 	Menampilkan pesan “Sukses Perbarui Data” kemudian data ter- <i>update</i> di <i>database</i> dan ditampilkan pada tabel transaksi masuk	Berhasil menampilkan pesan data transaksi masuk berhasil diperbarui dan tersimpan di <i>database</i> serta menampilkan data yang telah di- <i>update</i> pada tabel transaksi masuk.
Menghapus data transaksi masuk	<ol style="list-style-type: none"> 1. Membuka menu transaksi masuk 2. Memilih data transaksi masuk 3. Menekan tombol <i>delete</i> data 4. Menekan tombol “ya saya yakin” (hapus data) 	Menampilkan pesan “Apakah yakin? Ingin menghapus data ini...” ketika ingin menghapus data yang dipilih, lalu muncul pesan “Sukses <i>Delete</i> Data” saat menekan tombol hapus kemudian data terhapus dari <i>database</i>	Berhasil menampilkan pesan <i>delete</i> dan <i>delete</i> data transaksi masuk di <i>database</i> dan di tabel transaksi masuk.
Mencari data transaksi masuk	<ol style="list-style-type: none"> 1. Membuka menu transaksi masuk 2. Input kata pencarian 	Menampilkan data hasil pencarian	Berhasil menampilkan hasil pencarian

9. Test Case Mengelola Transaksi Keluar

Skenario	Langkah Tes	Hasil yang Diharapkan	Hasil Aktual
Menambah data transaksi keluar	<ol style="list-style-type: none"> 1. Membuka menu transaksi keluar 2. Menekan tombol tambah data 3. Input data (tanggal keluar, material, jumlah keluar, No PO, nama lokasi) 4. Menekan tombol <i>save</i> 	Menampilkan pesan “Sukses Tambah Data” kemudian data tersimpan ke <i>database</i> dan ditampilkan pada tabel transaksi keluar	Berhasil menampilkan pesan data transaksi keluar berhasil ditambah dan tersimpan di <i>database</i> serta menampilkan datanya pada tabel transaksi keluar.
Mengedit data transaksi keluar	<ol style="list-style-type: none"> 1. Membuka menu transaksi keluar 2. Memilih data transaksi keluar 3. Menekan tombol edit data 4. Input data perubahan 5. Menekan tombol <i>update</i> 	Menampilkan pesan “Sukses Perbarui Data” kemudian data ter- <i>update</i> di <i>database</i> dan ditampilkan pada tabel transaksi keluar	Berhasil menampilkan pesan data transaksi keluar berhasil diperbarui dan tersimpan di <i>database</i> serta menampilkan data yang telah di- <i>update</i> pada tabel transaksi keluar.
Menghapus data transaksi keluar	<ol style="list-style-type: none"> 1. Membuka menu transaksi keluar 2. Memilih data transaksi keluar 3. Menekan tombol <i>delete</i> data 4. Menekan tombol “ya saya yakin” (hapus data) 	Menampilkan pesan “Sukses Perbarui Data” kemudian data ter- <i>update</i> di <i>database</i> dan ditampilkan pada tabel transaksi keluar	Berhasil menampilkan pesan data transaksi keluar berhasil diperbarui dan tersimpan di <i>database</i> serta

			menampilkan data yang telah di- <i>update</i> pada tabel transaksi keluar.
Mencari data transaksi keluar	1. Membuka menu transaksi keluar 2. Input kata pencarian	Menampilkan data hasil pencarian	Berhasil menampilkan hasil pencarian

10. Test Case Mengelola Data User

Skenario	Langkah Tes	Hasil yang Diharapkan	Hasil Aktual
Menambah data <i>user</i>	1. Membuka menu <i>user</i> 2. Menekan tombol tambah data 3. Input data (nama <i>user</i> , <i>username</i> , password dan level) 4. Menekan tombol <i>save</i>	Menampilkan pesan “Sukses Tambah Data” kemudian data tersimpan ke <i>database</i> dan ditampilkan pada tabel data <i>user</i>	Berhasil menampilkan pesan data <i>user</i> berhasil ditambah dan tersimpan di <i>database</i> serta menampilkan datanya pada tabel data <i>user</i> .
Mengedit data <i>user</i>	1. Membuka menu <i>user</i> 2. Memilih data <i>user</i> 3. Menekan tombol edit data 4. Input data perubahan 5. Menekan tombol <i>update</i>	Menampilkan pesan “Sukses Perbarui Data” kemudian data ter- <i>update</i> di <i>database</i> dan ditampilkan pada tabel data <i>user</i>	Berhasil menampilkan pesan data <i>user</i> berhasil diperbarui dan tersimpan di <i>database</i> serta menampilkan data yang telah di- <i>update</i> pada tabel data <i>user</i> .

Menghapus data <i>user</i>	<ol style="list-style-type: none"> 1. Membuka menu <i>user</i> 2. Memilih data lokasi 3. Menekan tombol <i>delete</i> data 4. Menekan tombol “ya saya yakin” (hapus data) 	Menampilkan pesan “Apakah yakin? Ingin menghapus data ini...” ketika ingin menghapus data yang dipilih, lalu muncul pesan “Sukses <i>Delete</i> Data” saat menekan tombol hapus kemudian data terhapus dari <i>database</i>	Berhasil menampilkan pesan <i>delete</i> dan <i>delete</i> data <i>user</i> di <i>database</i> dan di tabel data <i>user</i> .
Mencari data <i>user</i>	<ol style="list-style-type: none"> 1. Membuka menu <i>user</i> 2. Input kata pencarian 	Menampilkan data hasil pencarian	Berhasil menampilkan hasil pencarian

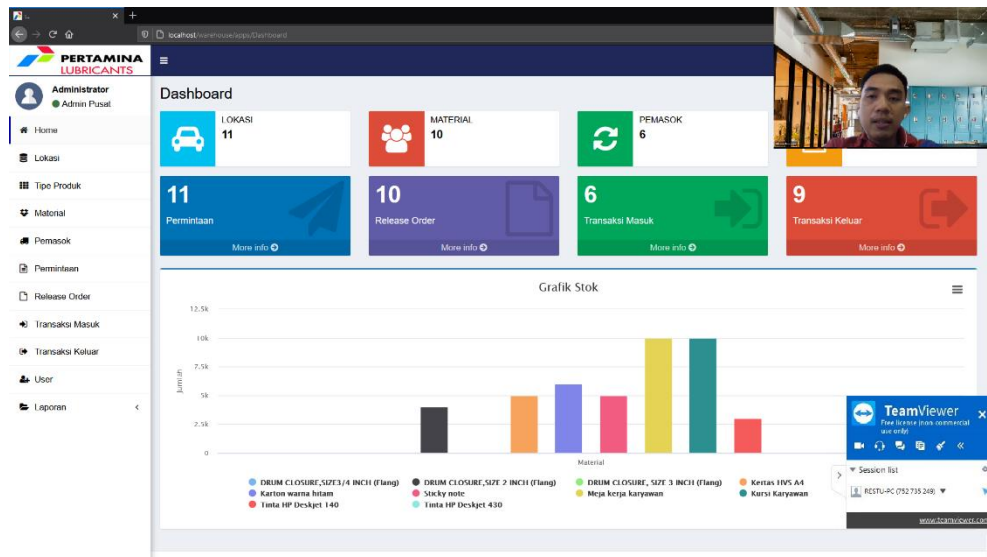
11. Test Case Membuat Laporan

Skenario	Langkah Tes	Hasil yang Diharapkan	Hasil Aktual
Mencetak laporan permintaan	<ol style="list-style-type: none"> 1. Membuka menu laporan permintaan 2. Input periode tanggal 3. Menekan tombol cari 4. Menekan tombol <i>download</i> laporan yang dicari 	Menampilkan data laporan permintaan sesuai periode tanggal yang dipilih, kemudian ada notifikasi dari sistem untuk <i>user</i> menyimpan laporan permintaan dalam format <i>Microsoft Excel</i> ketika user menekan tombol <i>download</i>	Berhasil menampilkan data laporan permintaan sesuai periode tanggal yang dipilih dan laporan berhasil diunduh dalam format <i>Microsoft Excel</i> .

Mencetak laporan <i>release order</i>	<ol style="list-style-type: none"> 1. Membuka menu laporan <i>release order</i> 2. Input periode tanggal 3. Menekan tombol cari 	Menampilkan data laporan <i>release order</i> sesuai periode yang dipilih, kemudian ada notifikasi dari sistem untuk <i>user</i> menyimpan laporan <i>release order</i> dalam format <i>Microsoft Excel</i> ketika user menekan tombol <i>download</i>	Berhasil menampilkan data laporan <i>release order</i> sesuai periode yang dipilih dan laporan berhasil diunduh dalam format <i>Microsoft Excel</i> .
Mencetak laporan transaksi masuk	<ol style="list-style-type: none"> 1. Membuka menu laporan transaksi masuk 2. Input periode tanggal 3. Menekan tombol cari 	Menampilkan data laporan transaksi masuk sesuai periode yang dipilih, kemudian ada notifikasi dari sistem untuk <i>user</i> menyimpan laporan transaksi masuk dalam format <i>Microsoft Excel</i> ketika user menekan tombol <i>download</i>	Berhasil menampilkan data laporan transaksi masuk sesuai periode yang dipilih dan laporan berhasil diunduh dalam format <i>Microsoft Excel</i> .
Mencetak laporan transaksi keluar	<ol style="list-style-type: none"> 1. Membuka menu laporan transaksi keluar 2. Input periode tanggal 3. Menekan tombol cari 	Menampilkan data laporan transaksi keluar sesuai periode yang dipilih, kemudian ada notifikasi dari sistem untuk <i>user</i> menyimpan laporan	Berhasil menampilkan data laporan transaksi keluar sesuai periode yang dipilih dan laporan berhasil diunduh dalam

		transaksi keluar dalam format <i>Microsoft Excel</i> ketika user menekan tombol <i>download</i>	format <i>Microsoft Excel</i> .
Mencetak laporan stok	1. Membuka menu laporan stok 2. Input periode tanggal 3. Menekan tombol cari	Menampilkan data laporan stok sesuai periode yang dipilih, kemudian ada notifikasi dari sistem untuk <i>user</i> menyimpan laporan stok dalam format <i>Microsoft Excel</i> ketika user menekan tombol <i>download</i>	Berhasil menampilkan data laporan stok sesuai periode yang dipilih dan laporan berhasil diunduh dalam format <i>Microsoft Excel</i> .

1. Kesimpulan yang diambil dari pengujian pada *user* yang menggunakan sistem informasi logistik dengan menggunakan metode UAT adalah 39% *user* mengatakan setuju, kemudian 52% *user* sangat setuju, dan 9% *user* mengatakan tidak setuju untuk aspek desain telah dirancang sesuai dengan keinginan. *User* juga mengatakan bahwa proses sistem informasi logistik sudah sesuai dengan kebutuhannya, 71% sangat setuju, 4% tidak setuju, dan 26% setuju. Terkait fungsi sistem, sebanyak 86% *user* mengatakan sangat setuju dan 14% pengguna setuju bahwa aplikasi ini telah memenuhi harapan mereka.
2. Dari pengujian UAT pada pengguna sistem informasi logistik secara keseluruhan dapat disimpulkan bahwa 68% pengguna sangat setuju terhadap sistem yang dibuat dan yang lainnya menyatakan setuju sebanyak 27% dan kurang setuju sebanyak 5%.



Gambar 4.14 Dokumentasi Testing Dengan *User* PT. Pertamina Lubricants

Gambar 4.14 adalah *screenshot* dari salah satu orang yang berperan sebagai *user* pada PT. Pertamina Lubricants yang sedang melakukan *testing* terhadap aplikasi sistem informasi logistik yang telah dibuat.

Setelah melakukan pengujian sistem dengan metode UAT (*User Acceptance Test*) yang dilakukan oleh orang-orang yang berperan sebagai *user* dari sistem informasi logistik pada PT. Pertamina Lubricants didapatkan masukan dari *user* sebagai berikut :

1. Tambahkan keterangan pada verifikasi yang dilakukan oleh *warehouse manager*.
2. Harusnya transaksi masuk *link* dengan *release order*.
3. Informasi stok pada laporan stok tidak boleh sampai minus.

Pelaksanaan *testing* UAT ini dilaksanakan pada tanggal 21 Januari 2021 dengan menggunakan *Microsoft Teams* dan juga aplikasi *remote* yaitu *Team Viewer*.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil pembahasan dari penelitian dan pengujian sistem informasi logistik dengan metode pengujian *Black Box* dan UAT (*User Acceptance Test*), maka dapat disimpulkan bahwa sistem yang dibangun berbasis web ini dapat membantu pengelolaan logistik PT. Pertamina Lubricants dengan cepat tanpa harus mengecek ke gudang. Sistem yang dibangun juga dapat memenuhi dan memberikan informasi mengenai logistik yang akurat secara *real time* serta dapat menampilkan grafik stok sehingga bisa memudahkan user yang berkaitan dengan logistik di PT. Pertamina Lubricants. Sistem ini dapat mengelola barang masuk dan keluar, dimana ketika melakukan transaksi masuk maka stok material akan bertambah otomatis begitu juga dengan transaksi keluar maka stok akan berkurang secara otomatis sehingga pemindahan material dalam pengelolaan logistik pada PT. Pertamina Lubricants menjadi lebih terakomodir. Pembuatan laporan langsung otomatis dari data tersebut, selain itu juga laporan dapat di download dengan format *Microsoft Excel* sehingga dapat memudahkan dalam pembuatan laporan.

Hal ini tentu memudahkan admin dan *user* aplikasi sistem informasi pada PT. Pertamina Lubricants dalam hal mencatat dan mengontrol barang serta mempercepat waktu proses pendataan stok material masuk dan keluar di PT. Pertamina Lubricants.

5.2 Saran

Sistem yang telah dibuat masih mempunyai beberapa kekurangan, sehingga terdapat beberapa hal yang bisa dikembangkan untuk penelitian selanjutnya yaitu:

- a. Isi dari situs web harus diperbarui secara berkala oleh administrator sesuai dengan situasi perkembangan teknologi, sehingga dapat memberikan informasi dan pelatihan aktual bagi administrator yang ditunjuk untuk mengelola sistem informasi logistik ini.
- b. Di harapkan ke depannya sistem informasi logistik ini dapat lebih dikembangkan lagi dengan menggunakan basis *mobile*.