

**RANCANG BANGUN SISTEM INFORMASI PELAPORAN
KERUSAKAN *PART* SAAT PROSES PRODUKSI DI BAGIAN
TRIMMING 3 MENGGUNAKAN CODEIGNITER 3.1.7 DAN
MYSQL 5.5.32 PADA PT KRAMA YUDHA RATU MOTOR**

TUGAS AKHIR

Diajukan Untuk Memenuhi Salah Satu Syarat Penyelesaian Program Sarjana
Terapan Program Studi Sistem Informasi Industri Otomotif
Pada Politeknik STMI Jakarta

OLEH

NURUL ISTIQOMAH

1314089



**POLITEKNIK STMI JAKARTA
KEMENTERIAN PERINDUSTRIAN REPUBLIK INDONESIA
JAKARTA
2018**

ABSTRAK

PT Krama Yudha Ratu Motor merupakan sebuah perusahaan yang bergerak dibidang industri otomotif, produk-produk yang dihasilkan atau ditawarkan pun beragam **salah satunya adalah alat transportasi yaitu berupa mobil angkutan niaga** seperti kendaraan Colt T120 SS atau biasa dikenal dengan CJM dimana kendaraan ini diproduksi pada bagian *Trimming 3*. Dalam melakukan pencatatan data masih manual dan semi komputerisasi dengan menggunakan aplikasi sederhana seperti *Microsoft Excel* untuk mengolah data produksi dan juga untuk menghasilkan laporan produksi. Proses pengolahan data pada sistem informasi pelaporan kerusakan part di bagian *Trimming 3* masih belum terintegrasi antar bagian dimana ketika terjadi kerusakan part saat proses produksi berlangsung harus mendatangi bagian *Part Control* untuk memberikan *line damage* agar part yang rusak tadi diganti. Rancang bangun sistem informasi pelaporan kerusakan *part* saat proses produksi sangat diperlukan dalam perusahaan karena sistem informasi yang diusulkan akan mempermudah dan mengintegrasikan proses pengolahan data sehingga akan membantu antar bagian dalam sistem untuk saling bertukar informasi dan mengambil keputusan dengan cepat. Pengembangan sistem informasi produksi ini menggunakan metode prototipe evolusioner. Pemodelan sistem menggunakan *unified modeling language* (UML), pemodelan data menggunakan *entity relational diagram* (ERD) dan perancangan sistem menggunakan *windows navigation diagram* (WND) dan perancangan antarmuka. Pembuatan aplikasi sistem informasi produksi ini menggunakan bahasa pemrograman PHP dengan *framework Codeigniter 3.1.7* dan menggunakan *database MySQL 5.5.32*. Dengan adanya sistem informasi pelaporan kerusakan *part* ini diharapkan dapat membantu bagian *Trimming 3* dalam mengolah data dan mempermudah untuk membuat laporan yang berkaitan dengan kerusakan *part* saat proses produksi dengan menggunakan sistem yang sudah terintegrasi.

Kata Kunci : rancang bangun, sistem informasi, pelaporan, produksi, kerusakan *part*, prototipe evolusioner, *Codeigniter 3.1.7*, *MySQL 5.5.32*.

BAB I

PENDAHULUAN

1.1. Latar Belakang

Seiring dengan perkembangan jaman yang disertai perkembangan ilmu pengetahuan dan teknologi yang tumbuh demikian pesat di segala bidang, khususnya di bidang teknologi informasi yang cepat dan mendorong perusahaan untuk saling berlomba meningkatkan mutu dan kualitas baik produk, pelayanan, sumber daya manusia, dan sumber daya mesin. Teknologi informasi menjadi sarana atau acuan yang sangat penting untuk mendukung perkembangan sistem informasi pada perusahaan, terkhusus bagi perusahaan yang baru membangun sebuah sistem dalam pengolahan data yang besar maupun kecil agar data yang dikelola menjadi lebih efisien dan efektif.

Perkembangan teknologi informasi dan komunikasi membawa perubahan dalam berbagai sektor, termasuk juga sektor industri. Pada saat ini banyak perusahaan yang menggunakan teknologi informasi sebagai sarana utama untuk menunjang dan meningkatkan layanan dan operasional, salah satunya adalah dengan digunakannya sistem informasi dalam kegiatan proses produksi dan pendataan data yang sistematis. Pengolahan data secara terkomputerisasi berupa sistem informasi sangat diperlukan bagi setiap perusahaan yang berada dalam lingkup industri untuk membuat suatu kemajuan dalam pelaksanaan produksi serta sebagai sarana pengambilan keputusan di perusahaan tersebut.

PT Krama Yudha Ratu Motor merupakan sebuah perusahaan yang bergerak dibidang industri otomotif, produk-produk yang dihasilkan atau ditawarkan pun beragam **salah satunya adalah alat transportasi yaitu berupa mobil angkutan niaga** seperti kendaraan Colt T120 SS atau biasa dikenal dengan CJM dimana kendaraan ini diproduksi pada bagian *trimming* 3 yaitu proses yang dilakukan setelah proses *welding* dan *painting* yang merupakan proses pemasangan atau perakitan *spare part* atau komponen-komponen

kendaraan contohnya seperti pemasangan aki, pemasangan kabel-kabel dalam mobil yang dilakukan secara manual.

Perkembangan teknologi menyebabkan adanya pergeseran penggunaan sistem informasi. Sistem informasi mulai beralih dari sistem manual menjadi sistem terkomputerisasi. Sistem informasi yang terkomputerisasi ini berguna untuk meng-input dan mengolah data secara cepat dan akurat, memudahkan akses komunikasi, dan mengotomatisasi suatu proses bisnis. Pada bagian *Trimming 3* dalam melakukan pencatatan data masih manual dan semi komputerisasi dengan menggunakan aplikasi sederhana seperti *Microsoft Excel* dalam mengolah data produksi dan juga untuk menghasilkan laporan produksi. Dimana dalam pengolahan datanya masih ada beberapa proses yang masih belum terintegrasi dengan informasi yang ada salah satunya yaitu pada saat melaporkan *part* yang rusak pada saat proses produksi dengan mengisi manual kartu *line damage* untuk diberikan secara langsung kepada bagian *Part Control* agar *part* yang rusak diganti dengan *part* yang baru. Selain itu, kurangnya tempat penyimpanan dokumen membuat para pekerja dibagian tersebut sering menyimpan dokumen di lokasi yang tidak semestinya sehingga sering terjadi kesulitan mencari dokumen apabila dibutuhkan dan juga beresiko terjadi kehilangan dokumen yang disimpan.

Untuk mengatasi masalah tersebut, diperlukan aplikasi untuk membantu perusahaan dalam mengatasi masalah tersebut. Adapun judul Tugas Akhir ini adalah “RANCANG BANGUN SISTEM INFORMASI PELAPORAN KERUSAKAN *PART* SAAT PROSES PRODUKSI DI BAGIAN *TRIMMING 3* MENGGUNAKAN CODEIGNITER 3.1.7 DAN MYSQL 5.5.32 PADA PT KRAMA YUDHA RATU MOTOR”

1.2. Pokok Permasalahan

Permasalahan yang terjadi pada PT Krama Yudha Ratu Motor adalah sebagai berikut:

1. Proses pengolahan data pada sistem informasi pelaporan kerusakan *part* di bagian *Trimming 3* pada PT Krama Yudha Ratu Motor belum terintegrasi antar bagian dimana ketika terjadi kerusakan *part* saat proses produksi

berlangsung bagian *Trimming* 3 harus mendatangi bagian *Part Control* secara langsung untuk memberikan *line damage*.

2. Penyimpanan dokumen di sembarang tempat yang disebabkan kurangnya tempat penyimpanan dokumen sehingga berisiko terjadinya kehilangan data, kesulitan mencari data dan kerusakan data.

1.3. Tujuan Penelitian

Adapun tujuan dari Tugas Akhir ini adalah merancang dan membangun sistem informasi produksi sebagai berikut:

1. Membantu bagian *Trimming* 3 di PT Krama Yudha Ratu Motor dalam mengolah data dan mempermudah untuk membuat laporan yang berkaitan dengan kerusakan *part* dan hasil produksi dengan menggunakan aplikasi yang sudah terintegrasi.
2. Menyediakan fasilitas untuk penyimpanan data yang berkaitan dengan produksi di bagian *Trimming* 3 dengan menggunakan *database* agar data yang sudah diinput disimpan dalam satu sistem penyimpanan sehingga mencegah terjadinya kehilangan data, kesulitan mencari data dan kerusakan data.

1.4. Batasan Masalah

Agar dalam penulisan Tugas Akhir ini lebih fokus dan lebih terarah, maka perlu diadakan batasan-batasan sebagai berikut:

1. Penelitian dilakukan di PT Krama Yudha Ratu Motor di Bagian *Trimming* 3 selama 1 bulan mulai dari tanggal 02 Oktober 2017 sampai tanggal 31 Oktober 2017.
2. Ruang lingkup yang dianalisis hanya mencakup pelaporan mengenai kerusakan *part* pada saat proses produksi dan hasil produksi di bagian *Trimming* 3.

1.5. Manfaat Tugas Akhir

Adapun manfaat yang diharapkan dari hasil pembuatan Tugas Akhir ini adalah sebagai berikut:

1. Bagi perusahaan
Hasil penelitian ini agar dapat diimplementasikan di perusahaan untuk membantu kinerja perusahaan dalam melaporkan kerusakan *part* serta pada saat proses produksi di bagian *Trimming* 3.
2. Bagi mahasiswa
 - a. Memberikan kemampuan dalam mengaplikasikan teori secara jelas terhadap masalah yang diamati.
 - b. Memberikan wawasan dan pengalaman kepada mahasiswa dalam menganalisis suatu sistem dan diharapkan dapat memberikan suatu solusi permasalahan.
3. Bagi pihak lain
Hasil penelitian ini diharapkan dapat memberikan sumbangan pemikiran dan sebagai referensi bagi peneliti lain yang ingin melakukan penelitian serupa.

1.6. Sistematika Penulisan

Agar lebih mempermudah perumusan dan pemecahan masalah yang akan dibahas pada penelitian Tugas Akhir ini, maka diuraikan tahapan-tahapan dalam penyusunan laporan ini sebagai berikut:

BAB I PENDAHULUAN

Pada bab ini memuat latar belakang, pokok permasalahan, tujuan penelitian, batasan masalah, manfaat tugas akhir, dan sistematika penulisan tugas akhir.

BAB II LANDASAN TEORI

Bab ini menjelaskan tentang berbagai teori yang di rangkum dari buku-buku ataupun berbagai macam referensi yang berkaitan dengan tema yang diambil. Teori-teori yang dipaparkan pada

laporan ini adalah seputar pengertian sistem, analisis sistem produksi, *Unified Modelling Language* (UML) sebagai alat bantu untuk membuat rancangan konseptual, teori-teori lain yang berhubungan dengan perancangan sistem, dan *Codeigniter* 3.1.7, MySQL 5.5.32.

BAB III METODOLOGI PENELITIAN

Pada bab ini membahas tentang langkah-langkah yang akan dilakukan dalam perumusan dan pemecahan masalah, jenis dan sumber data, metode pengumpulan data, serta metodologi pengembangan sistem yang digunakan

BAB IV PENGUMPULAN DAN PENGOLAHAN DATA

Pada bab ini akan membahas mengenai data-data yang telah diperoleh berdasarkan hasil pengamatan selama melaksanakan Praktik Kerja Lapangan di PT Krama Yudha Ratu Motor beserta pengumpulan data berupa proses bisnis dan dokumen yang terlibat pada sistem yang sedang berjalan di PT Krama Yudha Ratu Motor.

BAB V ANALISIS DAN PEMBAHASAN

Bab ini berisi analisis rinci dari pengolahan data, yakni mulai dari analisis dan perancangan sistem yang meliputi identifikasi kebutuhan user, pemodelan sistem menggunakan *tools* UML, pemodelan data, perancangan antar muka, dan implementasi sistem informasi pelaporan kerusakan *part* di *Trimming* 3 .

BAB VI KESIMPULAN DAN SARAN

Bab ini berisi kesimpulan secara keseluruhan yang diperoleh dari hasil analisis dan perancangan sistem serta saran-saran untuk penerapan dan pengembangan untuk penelitian selanjutnya pada sistem yang bersangkutan.

BAB II

LANDASAN TEORI

2.1. Pengertian Rancang Bangun

Menurut Jogiyanto (2005) rancang bangun adalah tahap setelah analisis dari siklus pengembangan sistem yang merupakan pendefinisian dari kebutuhan-kebutuhan fungsional, serta menggambarkan bagaimana suatu sistem dibentuk yang dapat berupa penggambaran, perencanaan dan pembuatan sketsa atau pengaturan dari beberapa elemen yang terpisah ke dalam satu kesatuan yang utuh dan berfungsi, termasuk menyangkut menkonfigurasi dari komponen-komponen perangkat keras dan perangkat lunak dari suatu sistem.

2.2. Konsep Dasar Sistem

Sistem dapat didefinisikan sebagai suatu kesatuan yang terdiri dari dua atau lebih komponen atau subsistem yang berinteraksi untuk mencapai suatu tujuan. Berikut definisi sistem menurut para ahli dalam Rochaety et al. (2013):

1. Sistem adalah seperangkat unsur yang saling berhubungan dan saling mempengaruhi dalam satu lingkungan tertentu (Ludwig, 2007).
2. Sistem adalah sekumpulan elemen yang saling berhubungan untuk mencapai suatu tujuan (Rapoport, 2004).
3. Sistem adalah kumpulan dari elemen-elemen yang berinteraksi untuk mencapai suatu tujuan tertentu. Sistem ini menggambarkan suatu kejadian-kejadian dan kesatuan yang nyata adalah objek nyata, seperti tempat, benda, dan orang-orang yang benar adanya (Jogianto, 2005).

2.2.1. Karakteristik Sistem

Dalam Hutahaean (2014), sistem dikatakan baik apabila memiliki karakteristik sebagai berikut:

1. Komponen

Suatu sistem terdiri dari sejumlah komponen-komponen yang saling berinteraksi, yang artinya saling bekerja sama membentuk satu kesatuan. Komponen sistem terdiri dari komponen yang berupa subsistem atau bagian-bagian dari sistem.

2. Batasan sistem (*boundary*)

Batasan sistem merupakan daerah yang membatasi suatu sistem dengan sistem yang lain atau dengan lingkungan luarnya. Batasan sistem ini memungkinkan suatu sistem dipandang sebagai suatu kesatuan. Batasan suatu sistem menunjukkan ruang lingkup (*scope*) dari sistem tersebut.

3. Lingkungan luar sistem (*environment*)

Lingkungan luar sistem adalah diluar batas dari sistem yang mempengaruhi operasi sistem. Lingkungan dapat bersifat menguntungkan yang harus tetap dijaga dan yang merugikan yang harus dikendalikan, kalau tidak akan mengganggu kelangsungan hidup dari sistem.

4. Penghubung sistem (*interface*)

Penghubung sistem merupakan media penghubung antara satu subsistem dengan subsistem lainnya. Melalui penghubung ini memungkinkan sumber-sumber daya mengalir dari subsistem ke subsistem lain. Keluaran (*output*) dari subsistem akan menjadi masukan (*input*) untuk subsistem lain melalui penghubung.

5. Masukan sistem (*input*)

Masukan adalah energi yang dimasukkan kedalam sistem, yang dapat berupa perawatan (*maintenance input*), dan masukan sinyal (*signal input*). *Maintenance input* adalah energi yang dimasukkan agar sistem dapat beroperasi. *Signal input* adalah energy yang diproses untuk didapatkan keluaran. Contoh dalam sistem komputer program adalah *maintenance input* sedangkan data adalah *signal input* untuk diolah menjadi informasi.

6. Pengolah sistem

Suatu sistem menjadi bagian pengolah yang akan merubah masukan

menjadi keluaran. Sistem produksi akan mengolah bahan baku menjadi bahan jadi.

7. Keluaran sistem (*output*)

Keluaran sistem adalah hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna dan sisa pembuangan. Misalnya, komputer menghasilkan panas yang merupakan sisa pembuangan, sedangkan informasi adalah keluaran yang dibutuhkan.

2.2.2. Klasifikasi Sistem

Menurut Rochaety et al. (2013), sistem dapat diklasifikasikan dalam beberapa sudut pandang diantaranya sebagai berikut ini:

1. Sistem diklasifikasikan sebagai:
 - a. Sistem abstrak (*abstract system*)
Sistem yang berupa pemikiran atau ide-ide yang tidak tampak secara fisik. Misalnya sistem teologika, yaitu sistem yang berupa pemikiran-pemikiran hubungan antara manusia dengan Tuhan.
 - b. Sistem fisik (*physical system*)
Sistem yang ada secara fisik. Misalnya sistem computer, sistem produksi dan lain sebagainya.
2. Sistem diklasifikasikan sebagai:
 - a. Sistem alamiah (*natural system*)
Sistem yang terjadi melalui proses alam, tidak buatan manusia. Misalnya sistem perputaran bumi.
 - b. Sistem buatan manusia (*human made system*)
Sistem yang melibatkan interaksi antara manusia dengan mesin yang disebut dengan *human-machine system* atau ada yang menyebut dengan *man-machine system*.
3. Sistem diklasifikasikan sebagai:
 - a. Sistem tertentu (*deterministic system*)
Sistem tertentu beroperasi dengan tingkah laku yang sudah dapat

diprediksi. Interaksi diantara bagian-bagiannya dapat dideteksi dengan pasti, sehingga keluaran dari sistem dapat diramalkan.

b. Sistem tidak tertentu (*probabilistic system*)

Sistem tidak tertentu adalah sistem yang kondisi masa depannya tidak dapat diprediksi karena mengandung unsur probabilitas.

4. Sistem diklasifikasikan sebagai:

a. Sistem tertutup (*close-loop system*)

Sistem tertutup merupakan sistem yang tidak berhubungan dengan lingkungan luarnya. Sistem ini bekerja secara otomatis tanpa adanya turut campur tangan dari pihak luar. Secara teoritis sistem tertutup ini ada, tetapi kenyataannya tidak ada sistem yang benar-benar tertutup, yang ada hanyalah *relatively closed system* (secara relatif tertutup, tidak benar-benar tertutup).

b. Sistem terbuka (*open-loop system*)

Sistem terbuka adalah sistem yang dipengaruhi oleh lingkungan luarnya. Sistem ini menerima masukan dan menghasilkan keluaran untuk lingkungan luar atau subsistem yang lainnya. Karena sistem terbuka dipengaruhi oleh lingkungannya, maka suatu sistem harus memiliki sistem pengendalian yang baik.

2.2.3. Tujuan Sistem

Tujuan sistem merupakan target atau sasaran akhir yang ingin dicapai oleh suatu sistem. Agar supaya target tersebut bisa tercapai, maka target atau sasaran tersebut harus diketahui terlebih dahulu ciri-ciri atau kriterianya. Upaya mencapai suatu sasaran tanpa mengetahui ciri-ciri atau kriteria dari sasaran tersebut kemungkinan besar sasaran tersebut tidak akan pernah tercapai. Ciri-ciri atau kriteria dapat juga digunakan sebagai tolak ukur dalam menilai suatu keberhasilan suatu sistem dan menjadi dasar dilakukannya suatu pengendalian (Susanto, 2013).

2.3. Pengertian Informasi

Menurut Susanto (2013), informasi adalah hasil dari pengolahan data, akan tetapi tidak semua hasil dari pengolahan tersebut bisa menjadi informasi, hasil pengolahan data yang tidak memberikan makna atau arti serta tidak bermanfaat bagi seseorang bukanlah merupakan informasi bagi orang tersebut. Dari definisi diatas dapat diketahui bahwa informasi adalah suatu pengolahan data yang diproses sehingga mempunyai arti dan bermanfaat bagi seseorang.

2.4. Konsep Dasar Sistem Informasi

Menurut Marakas dan O'Brien (2017), sistem informasi adalah kombinasi dari orang-orang, perangkat keras, perangkat lunak, jaringan komunikasi, sumber daya data, dan kebijakan serta prosedur dalam menyimpan, mendapatkan kembali, mengubah, dan menyebarkan informasi dalam suatu organisasi. Orang-orang bergantung pada sistem informasi modern untuk berkomunikasi satu sama lain dengan menggunakan berbagai macam alat (perangkat keras), instruksi dan prosedur pemrosesan informasi (perangkat lunak), saluran komunikasi (jaringan), dan penyimpanan data (sumber daya data). Walaupun sistem informasi pada saat ini biasanya berhubungan dengan komputer, kita telah menggunakan sistem informasi sejak awal peradaban. Bahkan saat ini pun kita selalu menggunakan sistem informasi yang tidak ada hubungannya dengan komputer. Contohnya adalah seperti tas tempat buku, perencanaan harian, buku catatan, dan folder dokumen yang merupakan bagian dari sistem informasi yang dirancang untuk membantu kita menyusun masukan yang diterima melalui selebaran, materi pelajaran, presentasi, dan diskusi. Alat-alat tersebut juga membantu kita memproses masukan menjadi keluaran yang bermanfaat seperti pekerjaan rumah dan nilai ujian yang bagus.

2.4.1. Komponen Sistem Informasi

Sistem informasi terdiri dari komponen-komponen yang disebut dengan istilah blok bangunan (*building block*), yang terdiri dari blok masukan, blok model, blok keluaran, blok teknologi, blok basis data, dan blok kendali. Sebagai

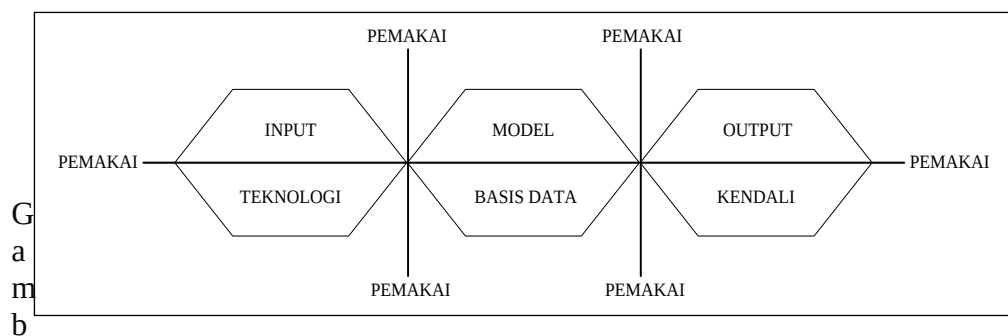
suatu sistem, keenam blok tersebut saling berinteraksi satu dengan yang lain membentuk satu kesatuan untuk mencapai sasaran (Sutabri, 2014).

1. Blok masukan (*input block*)
Input mewakili data yang masuk ke dalam sistem informasi. Yang dimaksud dengan *input* di sini termasuk metode dan media untuk menangkap data yang akan dimasukan, yang dapat berupa dokumen-dokumen dasar.
2. Blok model (*model block*)
 Blok ini terdiri dari kombinasi prosedur, logika, dan model matematika yang akan memanipulasi data *input* dan data yang tersimpan di basis data dengan cara yang sudah tertentu untuk menghasilkan keluaran yang diinginkan.
3. Blok keluaran (*output block*)
 Produk dari sistem informasi adalah keluaran yang merupakan informasi yang berkualitas dan dokumentasi yang berguna untuk semua tingkatan manajemen serta semua pemakai sistem.
4. Blok teknologi (*technology block*)
 Teknologi merupakan *tool box* dalam sistem informasi. Teknologi digunakan untuk menerima *input*, menjalankan model, menyimpan dan mengakses data, menghasilkan dan mengirimkan keluaran dan membantu pengendalian sistem secara keseluruhan. Teknologi terdiri dari 3 (tiga) bagian utama, yaitu teknisi (*brainware*), perangkat lunak (*software*), dan perangkat keras (*hardware*).
5. Blok basis data (*database block*)
 Basis data merupakan kumpulan data yang saling berkaitan dan berhubungan satu dengan yang lainnya, tersimpan di perangkat keras komputer dan perangkat lunak digunakan untuk memanipulasinya. Data perlu disimpan dalam basis data untuk keperluan penyediaan informasi lebih lanjut. Data di dalam basis data perlu diorganisasikan sedemikian rupa supaya informasi yang dihasilkan berkualitas. Organisasi basis data

yang baik juga berguna untuk efisiensi kapasitas penyimpanannya. Basis data diakses atau dimanipulasi dengan menggunakan perangkat lunak paket yang disebut dengan DBMS (*Database Management System*).

6. Blok kendali (*control block*)

Banyak hal dapat merusak sistem informasi, seperti bencana alam, api, temperatur, air, debu, kecurangan-kecurangan, kegagalan pada sistem itu sendiri, sabotase, dan sebagainya. Beberapa pengendalian perlu dirancang dan diterapkan untuk meyakinkan bahwa hal-hal yang dapat merusak sistem dicegah dan bila terlanjur terjadi maka kesalahan-kesalahan dapat dengan cepat diatasi.



ar II.1 Komponen Sistem Informasi
(Sumber: Sutabri, 2014)

2.5. Pengertian Pelaporan

Menurut Siagina (2013), pelaporan merupakan catatan yg memberikan informasi tentang kegiatan tertentu dan hasilnya disampaikan ke pihak yang berwenang atau berkaitan dengan kegiatan tertentu.

Reporting (pelaporan) merupakan salah satu fungsi manajemen berupa penyampaian perkembangan atau hasil kegiatan atau pemberian keterangan mengenai segala hal yang bertalian dengan tugas dan fungsi-fungsi kepada pejabat yang lebih tinggi. baik secara lisan maupun tertulis sehingga dalam penerimaan laporan dapat memperoleh gambaran bagaimana pelaksanaan tugas orang yang memberi laporan. (Gullick 2004)

Berdasarkan pendapat tersebut, dapat disimpulkan bahwa pelaporan adalah suatu bentuk penyampaian informasi yang didukung oleh data yang lengkap

sesuai dengan fakta sehingga informasi yang diberikan dapat dipercaya serta mudah dipahami. Dalam penyampaian, pelaporan dapat bersifat lisan maupun tertulis.

2.5.1. Fungsi Pelaporan

Adapun beberapa fungsi laporan menurut Locker dikutip oleh Sukoco (2007) yaitu:

1. Sebagai sarana komunikasi vertikal
Melalui laporan, pihak bawahan dapat menginformasikan berbagai kegiatan dan masukan berupa ide atau gagasan terhadap sesuatu permasalahan. Sedangkan pola pimpinan dapat memperoleh berbagai data dan informasi yang kemudian diolah, dikembangkan dan digunakan sebagai pengambilan keputusan serta perencanaan selanjutnya.
2. Sebagai alat pertanggungjawaban
Laporan merupakan manifestasi dari bentuk komunikasi vertikal dari atas ke bawah. Sebagai bentuk pertanggungjawaban terhadap pertanggungjawaban dan wewenang yang telah diberikan oleh atasan.
3. Memberikan informasi penting
Laporan harus berisi informasi faktual dan pemikiran-pemikiran yang rasional, argumentatif, serta objektif sebagai tanggapan terhadap fenomena faktual tersebut.
4. Sebagai bahan untuk pengambilan keputusan
Laporan dapat digunakan sebagai sumber pertimbangan untuk pengambilan kebijakan atau keputusan bagi unit kerja dan organisasi secara keseluruhan.

2.6. Pengertian Produksi

Produksi yang dalam bahasa Inggris disebut *production* ialah suatu kegiatan mengenai pembuatan produk baik berwujud fisik (*tangible products*) maupun berwujud jasa (*intangible products*). Pengertian di atas menjelaskan bahwa

produksi adalah proses yang berkenaan dengan pengubahan (*conversion*) asupan (*input*) menjadi barang atau jasa (Sinulingga, 2013).

Istilah *production* berasal dari bahasa Latin *producere* yang bermakna *to lead forward* yaitu membimbing ke depan. Saat ini produksi merupakan salah satu fungsi dasar yang paling penting dimasyarakat industri modern dan telah dipandang sebagai sebuah aktivitas budaya. Karena itu, suatu idea yang dikemukakan juga tidak jarang dipandang sebagai sebuah *intangible product*

Produksi merupakan salah satu fungsi yang paling mendasar dan penting dari kegiatan manusia dalam masyarakat industri modern dan sekarang dilihat sebagai aktivitas budaya.

Pada saat kebutuhan manusia masih sedikit dan sederhana, kegiatan produksi dan konsumsi seringkali dilakukan sendiri, yaitu seseorang memproduksi untuk memenuhi kebutuhannya sendiri. Namun, seiring dengan semakin beragamnya kebutuhan dan keterbatasan sumber daya, sehingga seseorang tidak dapat lagi memproduksi sendiri barang dan jasa yang dibutuhkannya, sehingga membutuhkan pihak lain untuk memproduksi apa yang menjadi kebutuhannya tersebut. Secara teknis produksi dapat diartikan sebagai suatu proses mentransformasi *input* menjadi *output*.

2.6.1. Fungsi Produksi

Menurut Hakim dan Yudha (2008), konsep fungsi produksi dapat digunakan untuk mengungkapkan hubungan fisik antara masukan (*input*) dengan keluaran (*output*) untuk suatu macam produk, fungsi produk menunjukkan *output* atau jumlah hasil produksi maksimum yang dapat dihasilkan per satuan waktu dengan menggunakan berbagai kombinasi sumber-sumber daya yang dipakai dalam berproduksi. Ada tiga fungsi utama dari kegiatan-kegiatan produksi yang dapat kita identifikasi yaitu:

1. Proses produksi, yaitu metode dan teknik yang digunakan dalam mengolah bahan baku menjadi produk.
2. Perencanaan produksi, yaitu merupakan tindakan antisipasi di masa mendatang sesuai dengan periode waktu yang direncanakan.

3. Pengendalian produksi, yaitu tindakan yang menjamin bahwa semua kegiatan yang dilaksanakan dalam perencanaan telah dilakukan sesuai dengan target yang telah ditetapkan.

2.6.2. Faktor-Faktor Produksi

Faktor produksi adalah sumber daya yang digunakan dalam sebuah proses produksi barang dan jasa. Pada awalnya, faktor produksi dibagi menjadi empat kelompok. Namun pada perkembangannya, faktor sumber daya alam diperluas cakupannya menjadi seluruh benda *tangible*, baik langsung dari alam maupun tidak, yang digunakan oleh perusahaan, dan kemudian disebut sebagai faktor fisik (*physical resources*). Selain itu, beberapa ahli juga menganggap sumber daya informasi sebagai sebuah faktor produksi mengingat semakin pentingnya peran informasi ditotalkan saat ini (Ricky, 2006). Secara total, saat ini ada lima hal yang dianggap sebagai faktor produksi, yaitu tenaga kerja (*labor*), modal (*capital*), sumber daya fisik (*physical resources*), kewirausahaan (*entrepreneurship*), dan sumber daya informasi (*information resources*). Berikut penjelasan dari faktor-faktor produksi:

1. Tenaga Kerja

Tenaga kerja merupakan faktor produksi yang secara langsung maupun tidak langsung menjalankan kegiatan produksi. Faktor produksi tenaga kerja juga dikategorikan sebagai faktor produksi asli. Dalam faktor produksi tenaga kerja, terkandung unsur pikiran, serta kemampuan yang dimiliki oleh tenaga kerja.

2. Modal

Yang dimaksud dengan modal barang atau peralatan yang dapat digunakan untuk melakukan proses produksi. Modal dapat digolongkan berdasarkan sumbernya, bentuknya, berdasarkan pemilikan, serta berdasarkan sifatnya. Berdasarkan sumbernya, modal dapat dibagi menjadi dua, yaitu modal sendiri dan modal asing. Modal sendiri adalah modal yang berasal dari dalam perusahaan sendiri, misalnya setoran dari pemilik perusahaan.

Sementara itu, modal asing adalah modal yang bersumber dari luar perusahaan, misalnya modal yang berupa pinjaman.

3. Sumber Daya Fisik

Faktor produksi fisik ialah semua kekayaan yang terdapat di barang mentah lainnya yang dapat digunakan dalam proses produksi. Faktor yang termasuk di dalamnya adalah tanah, air, dan bahan mentah lainnya.

4. Kewirausahaan

Faktor kewirausahaan adalah keahlian atau keterampilan yang digunakan seseorang dalam mengatur faktor-faktor produk.

5. Sumber Daya Informasi

Sumber daya informasi adalah seluruh data yang dibutuhkan perusahaan untuk menjalankan bisnisnya. Data ini bisa berupa ramalan kondisi pasar, pengetahuan yang dimiliki oleh karyawan, dan data ekonomi lainnya.

2.6.3. Klasifikasi Produksi

Merancang sistem produksi secara optimal dan memutuskan prosedur operasi yang optimal sesuai dengan karakteristik unik reproduksi masing-masing perusahaan, yaitu jenis produksi organisasi dibawah pertimbangan. Klasifikasi produksi secara umum dapat dilihat sebagai berikut (Sinulingga, 2013):

1. Produksi untuk Order/untuk Stok Persediaan

Produksi untuk pemesanan adalah produksi barang berdasarkan pesanan pelanggan. Produksi untuk stok adalah pengisian ulang dilakukan sebelum menerima pesanan pelanggan, produk ini kemudian disimpan sebagai persediaan dan dikirim sebagai pesanan diterima. Dalam produksi untuk penambahan stok, spesifikasi produk ditetapkan sebelum penerimaan pesanan dengan kepastian yang memadai.

2. Produksi berdasarkan kerja/terus-menerus

Klasifikasi produksi ini terkait dengan penjualan yang diharapkan atau volume produksi atau jumlah produk yang diminta per periode waktu,

katakanlah per bulan, per musim, per tahun. Jika volumenya sangat rendah, produksi akan dilakukan pada permintaan diharapkan atau basis bergerak lambat (produksi *job shop*), jika permintaan diharapkan akan sangat besar, produksi akan secara terus menerus (*continuous/mass production*) diantara dua hal ini, *intermittent (batch) production* berdasarkan lot atau batch yang tepat. Produksi terus menerus dan produksi massal dapat diterapkan dimana berbagai produk dibuat. Misalnya, dibaris mobil perakitan model campuran yang berbeda dapat diproduksi dalam suatu system tipe aliran yang disebut *mixed production*.

3. Produksi Part/Proses Diskrit

Ini merupakan klasifikasi produksi yang didasarkan pada sifat produk. Jika produk jadi terdiri dari beberapa bagian atau komponen diskrit, ini disebut bagian produksi didiskrit. Sebuah fitur dari jenis produksi adalah bahwa produk diskrit dapat dibongkar dan dipasang kembali bagiannya. Contohnya produksi mobil. Di sisi lain, dimana komponen atau bahan tidak dapat langsung diidentifikasi dalam sebuah produk jadi dan produk tidak dapat dibongkar. Contohnya adalah baja, kain, produk kimia, dan sebagainya.

2.6.4. Klasifikasi Posisi Produk

Menurut Sinulingga (2013) ada empat tipe produksi dalam lingkungan *manufacturing* yaitu:

a. *Engineering to order*

Pelanggan menyediakan spesifikasi dari produk yang diinginkannya dan berdasarkan spesifikasi tersebut perusahaan membuat desain, menyediakan bahan, membuat komponen, merakit, menguji kinerja produk kepada pelanggan. Kegiatan produksi dilakukan apabila pelanggan telah datang mengajukan *order*, *Bills of materials*, gambar-gambar teknik dan perintah kerja dipersiapkan perusahaan secara terpisah untuk setiap order.

b. *Make to order*

Pelanggan menyediakan spesifikasi dan desain produk. Berdasarkan desain tersebut perusahaan menyediakan bahan, pembuatan *part* dan komponen, merakit dan mengirimkan produk kepada pelanggan. Sama seperti *engineering to order* kegiatan produksi dilakukan apabila pelanggan telah mengajukan permintaan.

c. *Assembly to order*

Perusahaan menyediakan sejumlah model dasar dari produk tetapi dilengkapi dengan berbagai alternatif dan variasi yang diperkirakan akan memperkaya pilihan bagi pelanggan. Pelanggan melakukan pemilihan terhadap model, variasi dan tipe produk yang diinginkannya dari alternatif yang tersedia. Kegiatan produksi dilakukan untuk membuat komponen-komponen standar dengan semua variasinya dan perakitan produk akhir dilakukan setelah pelanggan mengajukan permintaan.

d. *Make to stock*

Pelanggan tidak mempunyai kesempatan untuk memilih sesuai dengan selera mereka tetapi membeli langsung produk yang sudah jadi dari persediaan. Kegiatan produksi dilakukan untuk mengisi persediaan yang jumlahnya dinyatakan dalam jadwal induk produksi.

2.7. Proses Perakitan (*Trimming*)

Proses *trimming* yaitu proses yang dilakukan setelah proses *welding* dan *painting* yang merupakan proses pemasangan atau perakitan *spare part* atau komponen-komponen kendaraan contohnya seperti pemasangan aki, pemasangan kabel-kabel dalam mobil yang dilakukan secara manual.

Perakitan berfungsi untuk memasang dan merakit komponen-komponen kendaraan. Pekerjaan perakitan dimulai bila obyek sudah siap untuk dipasang dan berakhir bila obyek tersebut telah bergabung secara sempurna. Perakitan juga dapat diartikan penggabungan antara bagian yang satu terhadap bagian yang lain atau pasangannya. Pada prinsipnya perakitan dalam proses produksi terdiri dari pasangan semua bagian-bagian komponen menjadi satu produk, proses

pengencangan, proses inspeksi dan pengujian fungsional (suhdi.wordpress.com, 2009).

2.7.1. Macam dan Jenis Perakitan

Ada beberapa macam jenis perakitan yang sering digunakan didunia industri, hal ini tergantung pada pekerjaan yang akan dilakukan. Biasanya faktor bentuk dan jumlah produk yang akan dihasilkan sangat menentukan. Pada umumnya ada dua macam jenis perakitan yaitu manual dan otomatis berikut penjelasannya (suhdi.wordpress.com, 2009):

1. Perakitan manual

Perakitan yang sebagian besar proses dikerjakan secara konvensional atau menggunakan tenaga manusia dengan peralatan yang sederhana tanpa alat-alat bantu yang spesifik atau khusus.

2. Perakitan otomatis

Perakitan yang dikerjakan dengan sistem otomatis seperti otomatisasi, elektronik, mekanik, gabungan mekanik dan elektronik (mekatronik), dan membutuhkan alat bantu yang khusus.

Sedangkan untuk jenis perakitan dapat dibedakan menurut jenis produk yang akan dilakukan perakitan yaitu:

1. Jenis perakitan produk tunggal

Perakitan dengan produk hanya satu jenis saja.

2. Jenis perakitan produk seri

Bila perakitan dilakukan dalam jumlah massal dalam bentuk dan ukuran yang sama. Contohnya proses perakitan produk elektronik, perakitan mobil, perakitan motor dan lain-lain.

2.8. Line Damage

Line Damage merupakan surat yang digunakan untuk meminta komponen atau *part* kepada bagian *Part Control* jika bagian produksi termasuk *Trimming* 3

yang mengalami kekurangan *part* yang akan digunakan karena mengalami kerusakan atau cacat saat diinstal oleh operator.

2.9. Pengembangan Sistem

Pengembangan sistem dapat berarti menyusun suatu sistem yang baru untuk menggantikan sistem yang lama secara keseluruhan atau memperbaiki sistem yang telah ada. Sistem yang lama perlu diperbaiki oleh karena beberapa hal, yaitu (Sutabri, 2014):

3. Munculnya masalah pada sistem yang lama.
4. Untuk meraih kesempatan.
5. Adanya instruksi.

Proses perancangan atau pengembangan sistem informasi, mulai dari konsep sampai dengan implementasinya disebut dengan istilah *Software Development Life Cycle* atau sering disebut juga *System Development Life Cycle* (SDLC) adalah suatu proses mengembangkan atau mengubah suatu sistem perangkat lunak dengan menggunakan model-model dan metodologi yang digunakan orang untuk mengembangkan sistem-sistem perangkat lunak sebelumnya berdasarkan *best practice* atau cara-cara yang sudah teruji baik (Rosa dan Shalahuddin, 2014).

Menurut Rosa dan Shalahuddin (2014), tahapan-tahapan yang ada pada SDLC secara global adalah sebagai berikut:

1. Inisiasi (*initiation*)
Tahap ini biasanya ditandai dengan pembuatan proposal proyek perangkat lunak.
2. Pengembangan konsep sistem (*system concept development*)
Mendefinisikan lingkup konsep termasuk dokumen lingkup sistem analisis manfaat biaya, manajemen rencana dan pembelajaran kemudahan sistem.
3. Perencanaan (*planning*)
Mengembangkan rencana manajemen proyek dan dokumen perencanaan lainnya. Menyediakan dasar untuk mendapatkan sumber daya (*resources*) yang dibutuhkan untuk memperoleh solusi.

4. Analisis kebutuhan (*requirements analysis*)
Menganalisis kebutuhan pemakai sistem perangkat lunak (*user*) dan mengembangkan kebutuhan *user*. Membuat dokumen kebutuhan fungsional.
5. Desain (*design*)
Mentransformasikan kebutuhan detail menjadi kebutuhan yang sudah lengkap, dokumen desain sistem fokus pada bagaimana dapat memenuhi fungsi-fungsi yang dibutuhkan.
6. Pengembangan (*development*)
Mengonversi desain ke sistem informasi yang lengkap termasuk bagaimana memperoleh dan melakukan instalasi lingkungan sistem yang dibutuhkan, membuat basis data dan mempersiapkan prosedur kasus pengujian, mempersiapkan berkas atau *file* pengujian, pengkodean, pengkompilasian, memperbaiki dan membersihkan program serta peninjauan pengujian.
7. Integrasi dan pengujian (*integration and test*)
Mendemonstrasikan sistem perangkat lunak bahwa telah memenuhi kebutuhan yang dispesifikasikan pada dokumen kebutuhan fungsional. Dengan diarahkan oleh staf penjamin kualitas (*quality assurance*) dan *user* sehingga menghasilkan laporan analisis pengujian.
8. Implementasi (*implementation*)
Termasuk pada persiapan implementasi, implementasi perangkat lunak pada lingkungan produksi (lingkungan luar *user*) dan menjalankan resolusi dari permasalahan yang teridentifikasi dari fase integrasi dan pengujian.
9. Operasi dan pemeliharaan (*operations and maintenance*)
Mendeskrripsikan pekerjaan untuk mengoperasikan dan memelihara sistem informasi pada lingkungan produksi (lingkungan pada *user*), termasuk implementasi akhir dan masuk pada proses peninjauan.
10. Disposisi (*disposition*)
Mendeskrripsikan aktifitas akhir dari pengembangan sistem dan membangun data yang sebenarnya sesuai dengan aktifitas *user*.

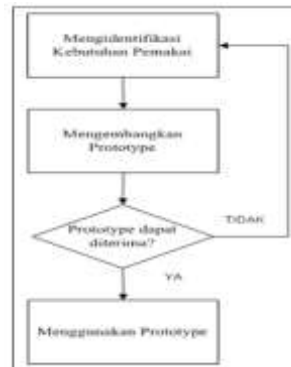
2.10. *Prototyping*

Prototyping adalah proses *iterative* dalam pengembangan sistem dimana kebutuhan diubah ke dalam sistem yang bekerja (*working system*) yang secara terus menerus diperbaiki melalui kerja sama antara pengguna dan analis. *Prototype* juga bisa dibangun melalui beberapa *tool* pengembangan untuk menyederhanakan proses. *Prototyping* merupakan bentuk dari *Rapid Application Development* (RAD).

Menurut McLeod dan Schell (2011) *prototype* adalah suatu versi sistem potensial yang disediakan bagi pengembang dan calon pengguna yang dapat memberikan gambaran bagaimana kira-kira sistem tersebut akan berfungsi bila telah disusun dalam bentuk yang lengkap. Proses dalam memproduksi suatu *prototype* disebut *prototyping*. Tujuannya adalah menghasilkan *prototype* secepat mungkin dan memperoleh umpan balik dari pengguna yang akan memungkinkan prototipe untuk ditingkatkan sampai sistem dianggap sempurna. Adapun jenis *prototype* terdapat dua jenis *prototype* yaitu *prototype* evolusioner (*evolutionary prototype*) dan *prototype requirement* (*requirement prototype*) (McLeod dan Schell 2011).

1. *Evolutionary Prototype*

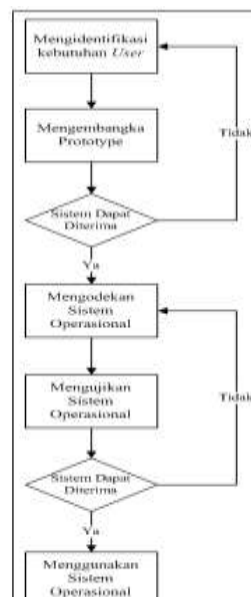
Evolutionary prototype yaitu *prototype* yang secara terus menerus dikembangkan hingga *prototype* tersebut memenuhi fungsi dan prosedur yang dibutuhkan oleh sistem. Pada pendekatan evolusioner, suatu *prototype* dibangun berdasarkan pada kebutuhan dan pemahaman secara umum. *Prototype* kemudian Ada empat langkah yang diambil dalam mengembangkan suatu *evolutionary prototype* yaitu identifikasi kebutuhan pengguna, mengembangkan prototipe, menentukan *prototype* dapat diterima atau tidak, dan penggunaan *prototype*.



Gambar II.2 Pengembangan *Evolutionary Prototype*
(Sumber: McLeod, 2011)

2. *Requirement Prototype*

Requirement prototype adalah suatu pengembangan untuk menentukan kebutuhan fungsional dari sistem baru pada saat para pengguna tidak mampu mengungkapkan dengan tepat apa yang mereka butuhkan. Saat kebutuhan telah ditentukan *requirement prototype* dapat mulai dikerjakan dan proyek siap untuk mengembangkan suatu sistem yang baru.



Gambar II.3 Pengembangan *Requirement Prototype*
(Sumber: McLeod, 2011)

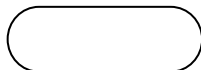
2.11. *Flowmap*

Menurut Jogiyanto (2005), *Flowmap* atau bagan alir adalah bagan yang menunjukkan aliran di dalam program atau prosedur sistem secara logika. *Flowmap* ini berfungsi untuk memodelkan masukan, keluaran, proses maupun transaksi dengan menggunakan simbol-simbol tertentu. Pembuatan *flowmap* ini harus dapat memudahkan bagi pemakai dalam memahami alur dari sistem atau transaksi. Adapun pedoman-pedoman dalam pembuatan *flowmap* adalah sebagai berikut:

1. *Flowmap* sebaiknya digambarkan dari atas ke bawah dan mulai dari bagian kiri dari suatu halaman.
2. Kegiatan di dalam *flowmap* harus ditunjukkan dengan jelas.
3. Harus ditunjukkan dari mana kegiatan akan dimulai dan dimana akan berakhir.
4. Masing-masing kegiatan didalam *flowmap* sebaiknya digunakan suatu kata yang mewakili suatu pekerjaan.
5. Masing-masing kegiatan didalam *flowmap* harus dalam urutan yang semestinya.
6. Kegiatan yang terpotong dan akan disambung ditempat lain harus ditunjukkan dengan jelas menggunakan simbol penghubung.
7. Gunakan simbol-simbol *flowmap* yang standar.

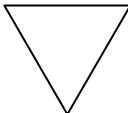
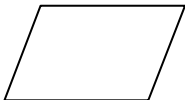
Adapun simbol-simbol yang sering digunakan dalam flowmap dapat dilihat pada Tabel II.I berikut ini:

Tabel II.I Simbol-simbol *Flowmap*

Simbol	Nama	Fungsi
	Mulai/Akhir	Menunjukkan dimulai/akhir dari sebuah proses.

	Dokumen	Simbol yang menyatakan <i>input</i> berasal dari dokumen dalam bentuk kertas atau <i>output</i> dicetak dikertas.
---	---------	---

Tabel II.I Simbol-simbol *Flowmap* (lanjutan)

Simbol	Nama	Fungsi
	Dokumen Rangkap	Menunjukkan multi dokumen.
	Manual Process	Simbol untuk menunjukkan pengolahan yang tidak dilakukan oleh komputer/pekerjaan manual.
	Proses	Proses pengolahan data.
	Arsip	Menunjukkan pengarsipan data.
	Garis Aliran	Arus dari suatu proses.
	Input/Output	Simbol <i>input/output</i> digunakan untuk mewakili data <i>input/output</i> .

(Sumber: Jogianto, 2005)

2.12. *Unified Modeling Language (UML)*

Unified Modelling Language (UML) adalah suatu bahasa yang digunakan untuk menentukan, memvisualisasikan, membangun, dan mendokumentasikan suatu sistem informasi. UML dikembangkan sebagai suatu alat untuk analisis dan desain berorientasi objek oleh Grady Booch, Jim Rumbaugh, dan Ivar Jacobson. Tujuan dari UML adalah untuk menyediakan kosa kata yang umum dari istilah-istilah berbasis objek dan teknik yang cukup banyak untuk memodelkan proyek pengembangan sistem dari analisis ke desain. Menurut Dennis (2010), diagram-diagram yang ada dalam UML terbagi menjadi dua bagian utama yaitu, *Structure diagram* dan *Behavior diagram*. *Structure diagram* biasanya digunakan untuk mempresentasikan data dan hubungan statik yang ada di dalam sebuah sistem informasi. Sedangkan *Behavior diagram* menyediakan para analis dengan sebuah gambaran hubungan yang dinamis antara instansi atau objek yang merepresentasikan sistem informasi bisnis. Versi 2.0 ini sudah diterima oleh *Object Management Group (OMG)* pada tahun 2003. Versi UML ini mendefinisikan suatu set dari 14 teknik untuk pemodelan sistem. Berikut adalah tabel dari jenis diagram yang ada dan fungsi-fungsinya:

Tabel II.2 Jenis Diagram Pada UML dan Fungsinya

Nama Diagram	Fungsi
<i>Structure Diagrams</i>	
<i>Class Diagram</i>	Menggambarkan hubungan antara pemodelan-pemodelan <i>class</i> di dalam sistem.
<i>Object Diagram</i>	Menggambarkan hubungan antara pemodelan-pemodelan objek di dalam sistem.
<i>Package Diagram</i>	Mengelompokkan elemen UML lainnya menjadi satu untuk membentuk level konstruksi yang lebih tinggi.

<i>Deployment Diagram</i>	Menampilkan arsitektur fisik dari suatu sistem. Bisa juga digunakan untuk menunjukkan komponen <i>software</i> sedang dijadikan arsitektur fisik suatu sistem.
<i>Component Diagram</i>	Menggambarkan hubungan fisik diantara komponen-komponen <i>software</i> .
<i>Composite Diagram</i>	Menggambarkan struktur internal dari suatu <i>class</i> dan hubungan diantara bagian-bagian dari suatu <i>class</i> .

Tabel II.2 Jenis Diagram Pada UML dan Fungsinya (lanjutan)

Nama Diagram	Fungsi
	<i>Behavior Diagram</i>
<i>Activity Diagram</i>	Menggambarkan proses bisnis masing-masing <i>class</i> .
<i>Sequence Diagram</i>	Memodelkan kebiasaan dari objek-objek dalam suatu <i>use case</i> . Fokus dalam aktifitas berdasarkan urutan waktu.
<i>Communication Diagram</i>	Memodelkan kebiasaan dari objek-objek dalam suatu <i>use case</i> . Fokus dalam komunikasi antara satu set dari kolaborasi objek dari suatu objek.
<i>Interaction Overview Diagram</i>	Menggambarkan suatu ikhtisar alur dari kontrol suatu proses.
<i>Timing Diagram</i>	Menggambarkan interaksi yang terjadi diantara suatu set dari objek-objek dan perubahan keadaan selama perjalanan waktu.
<i>Behavioral State Machine Diagram</i>	Memeriksa kebiasaan dari suatu <i>class</i> .
<i>Protocol State Machine Diagram</i>	Menggambarkan ketergantungan diantara perbedaan-perbedaan <i>interface</i> dari suatu <i>class</i> .
<i>Use Case Diagram</i>	Menangkap kebutuhan bisnis untuk sistem dan untuk menggambarkan interaksi diantara sistem dan lingkungannya.

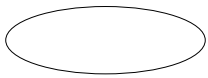
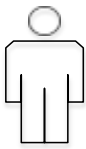
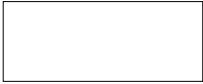
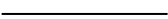
(Sumber : Dennis, 2010)

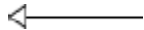
2.12.1. Use Case Diagram

Diagram *use case* dibuat untuk menggambarkan fungsionalitas yang diharapkan dari sebuah sistem. Suatu landasan pola pikir yang ditekankan dalam diagram *use case* ini adalah “apa” yang dapat diperbuat oleh sistem, dan

bukan “bagaimana” sistem melakukannya. Diagram *use case* menggambarkan kebutuhan sistem dari sudut pandang *user*, yaitu proses yang dilakukan oleh sistem dalam melayani *user* yang berinteraksi dengan sistem tersebut. Diagram ini sangat tepat untuk menggambarkan hubungan yang terjadi antara sistem dengan *user*nya, dapat dilihat pada tabel II.3 sebagai berikut:

Tabel II.3 Simbol-simbol *Use Case Diagram*

No.	Simbol	Deskripsi
1.	<i>Use Case</i> 	– Merepresentasikan bagian utama dari sistem secara fungsional. – Diletakan didalam <i>system boundary</i>. – Dilabelkan dengan frasa kata kerja deskriptif.
2.	<i>Actor</i> 	– Seseorang atau sistem yang mendapatkan keuntungan dari sistem. – Dilabelkan dengan peran. – Bisa diasosiasikan dengan aktor lainnya berdasarkan spesialisasi. – Diletakan diluar batas sistem.
3.	<i>Boundary</i> 	– Nama dari sistem terdapat didalam atau diatas. – Merepresentasikan ruang lingkup dari sistem
4.	<i>Association</i> 	– Menghubungkan suatu aktor dengan <i>use case</i> dengan apa interaksinya.
5.	<i>Extend</i> 	– Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan itu.

6.	<i>Generalization</i> 	– Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya.
7.	<i>Include</i> 	– Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsinya atau sebagai syarat dijalankan <i>use case</i> ini.

(Sumber : Dennis, 2010)

2.12.2. Activity Diagram

Digunakan untuk model perilaku dalam independen proses bisnis benda. Dalam banyak hal, diagram aktivitas dapat dipandang sebagai diagram aliran data yang canggih yang digunakan dalam hubungannya dengan analisis terstruktur. Namun, tidak seperti aliran data diagram, diagram aktivitas termasuk notasi yang membahas pemodelan paralel, kegiatan bersamaan dan proses. (Dennis, 2010)

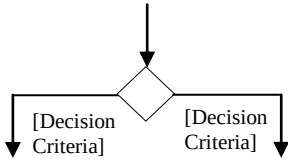
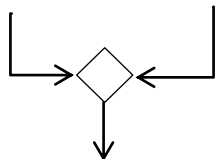
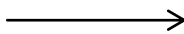
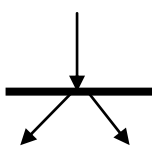
Yang perlu diperhatikan disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem. Diagram aktivitas juga banyak digunakan untuk mendefinisikan hal-hal berikut:

- Rancangan proses bisnis dimana setiap urutan aktivitas yang digambarkan merupakan proses bisnis sistem yang didefinisikan
- Urutan atau pengelompokan tampilan dari sistem atau *user interface* dimana setiap aktivitas dianggap memiliki sebuah rancangan antarmuka tampilan
- Rancangan pengujian dimana setiap aktivitas dianggap memerlukan sebuah pengujian yang perlu didefinisikan kasus ujinya
- Rancangan menu yang akan ditampilkan pada perangkat lunak

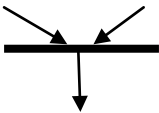
Berikut adalah simbol-simbol yang ada pada *Activity Diagram*:

Tabel II.4 Simbol-Simbol *Activity Diagram*

No.	Simbol	Deskripsi
1.	<i>Initial Node</i> 	Merupakan tanda awal dari sebuah aktifitas.
2.	<i>Activity</i> 	Merupakan sebuah gambaran aktifitas yang terjadi
3.	<i>Decision Node</i>	<i>Decision node</i> digunakan untuk merepresentasikan suatu alur logika yang

		timbul dari sekumpulan / urutan aktivitas pada suatu proses bisnis. Alur logika ini merupakan pilihan atas jalur aktivitas yang bernilai "True" dan "False", dan hanya salah satu dari jalur tersebut yang akan dipilih sesuai dengan syarat/kriteria pada <i>Decision node</i> yang telah ditentukan
4.	<i>Merge Node</i> 	<i>Merge node</i> digunakan untuk menyatukan / menutup alur logika yang sebelumnya dibentuk oleh <i>Decision node</i>
5.	<i>Final-Activity Node</i> 	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.
6.	<i>Swimlane</i> 	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi.
7.	<i>Control flow</i> 	Menunjukkan urutan eksekusi
8.	<i>Fork</i> 	Fork node digunakan untuk membagi (<i>split</i>) sekumpulan aktivitas dalam suatu proses bisnis, dimana sekumpulan aktivitas tersebut dapat berjalan secara paralel dalam satu kondisi waktu yang sama.

Tabel II.4 Simbol-Simbol *Activity Diagram* (lanjutan)

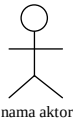
No.	Simbol	Deskripsi
9.	<i>Join</i> 	Join node digunakan untuk menyatukan / menutup aktivitas yang berjalan paralel yang sebelumnya dibentuk oleh Fork node

(Sumber: Dennis, 2010)

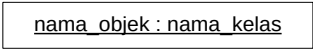
2.12.3. *Sequence Diagram*

Sequence diagram adalah salah satu dari dua jenis diagram interaksi. Mereka menggambarkan benda-benda yang berpartisipasi dalam kasus penggunaan dan pesan yang melewati antara mereka dari waktu ke waktu untuk satu *use case*. Sebuah diagram *sequence* adalah model dinamis yang menunjukkan urutan eksplisit pesan yang lewat di antara objek dalam interaksi didefinisikan. Karena urutan diagram menekankan pemesanan berbasis waktu kegiatan yang terjadi di antara set benda, mereka sangat membantu untuk memahami spesifikasi *real-time* dan kompleks menggunakan kasus (Dennis, 2010). Berikut simbol yang ada pada *Sequence Diagram*:

Tabel II.5 Simbol-Simbol *Sequence Diagram*

No.	Simbol	Deskripsi
1.	Aktor 	Orang atau sistem yang berasal dari manfaat dan eksternal ke sistem yang berpartisipasi secara berurutan dengan mengirim dan atau menerima pesan
2.	Garis Hidup/<i>lifeline</i> 	Menyatakan kehidupan suatu objek.

Tabel II.5 Simbol-Simbol *Sequence Diagram* (lanjutan)

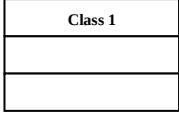
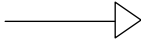
No.	Simbol	Deskripsi
3.	Objek 	Berpartisipasi secara berurutan dengan mengirim dan atau menerima pesan yang ditempatkan diatas diagram.
4.	<i>Execcution Occurence</i> 	Menyatakan objek dalam keadaan aktif dan berinteraksi pesan.
5.	<i>Message aMessage()</i> 	Pesan yang menggambarkan komunikasi yang terjadi antar objek.
6.	<i>Object Destruction</i> 	Orang atau sistem yang berasal dari manfaat dan eksternal ke sistem yang berpartisipasi secara berurutan dengan mengirim dan atau menerima pesan

(Sumber: Dennis, 2010)

2.12.4. Class Diagram

Class diagram merupakan salah satu diagram dalam konsep *Unified Modelling Language* (UML) yang menjelaskan mengenai berbagai jenis objek yang terdapat dalam sistem beserta beberapa hubungan antar objek tersebut. Sebuah *Class* merupakan deskripsi dari sekumpulan objek yang memiliki properti (*attribute*), operasi (*method*), relasi (*association*), dan tingkah laku (*behavior*) yang sama. Sebuah *Class* menggambarkan keadaan (atribut/properti) suatu sistem, sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut (metoda /fungsi). (Dennis, 2010). Berikut adalah simbol-simbol class diagram:

Tabel II.6 Simbol-Simbol *Class Diagram*

No.	Simbol	Deskripsi
1.	<i>Class</i> 	<i>Class</i> digunakan sebagai <i>template</i> dari kumpulan objek.
2.	<i>Association</i> 	Asosiasi (<i>Association</i>) yaitu hubungan statis antar class yang direpresentasikan dengan kata kerja (<i>Verb</i>). Umumnya menggambarkan class yang memiliki atribut berupa class lain, atau class yang harus mengetahui eksistensi class lain. Panah <i>navigability</i> menunjukkan arah <i>query</i> antar <i>class</i> .
3.	<i>Generalization</i> 	Merupakan sebuah <i>taxonomic relationship</i> antara <i>class</i> yang lebih umum dengan <i>class</i> yang lebih khusus
4.	<i>Aggregation</i> 	Menggambarkan suatu <i>class</i> terdiri dari <i>class</i> lain atau suatu <i>class</i> adalah bagian dari <i>class</i> lain.
5.	<i>Multiplicity</i> 1 0..1 (0..*) 2..4 1..* 2, 4..6, 8 *	<i>Multiplicity</i> merupakan gambaran dari sejumlah objek (<i>instance</i>) pada suatu <i>class</i> yang terlibat dalam asosiasi dengan sejumlah objek pada <i>class</i> yang lainnya

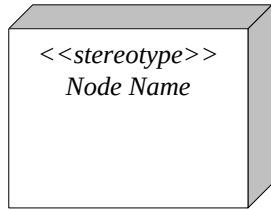
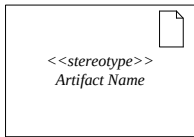
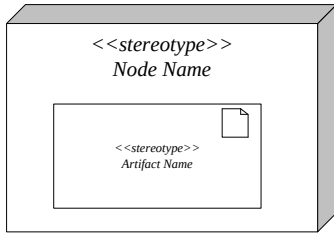
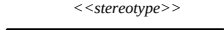
(Sumber: Dennis, 2010)

2.12.5. *Deployment Diagram*

Deployment diagram digunakan untuk mewakili hubungan diantara komponen-komponen perangkat keras dalam infrastuktur fisik sebuah sistem informasi. Misalnya, ketika merancang sistem informasi yang akan terdistribusi menggunakan *wide area network*, sebuah *deployment diagram* dapat digunakan untuk menunjukkan hubungan komunikasi antara node yang berbeda dalam jaringan. *Deployment diagram* juga dapat digunakan untuk mewakili komponen-komponen perangkat lunak dan bagaimana mereka dikerahkan dalam arsitektur

atau infrastruktur dari suatu sistem informasi. Berikut adalah simbol-simbol *deployment diagram* menurut Dennis 2010:

Tabel II.7 Simbol-Simbol *Deployment Diagram*

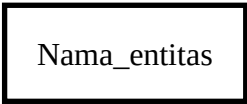
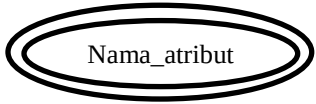
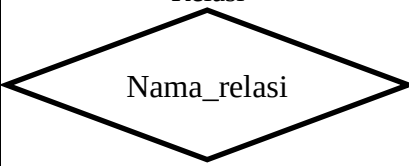
Nama Simbol	Deskripsi	Simbol
<i>Node</i>	Sumber daya komputasi seperti komputer klien, <i>server</i> atau perangkat jaringan individu.	
<i>Artifact</i>	Spesifikasi sebuah <i>software</i> atau <i>database</i> , contoh: <i>database</i> atau tabel atau <i>layer</i> .	
<i>Node with a deployed artifact</i>	Menggambarkan sebuah artefak yang ditempatkan pada node fisik. Mendukung pemodelan distribusi perangkat lunak melalui jaringan.	
<i>Communication path</i>	Merupakan asosiasi antara dua <i>node</i> . Memungkinkan <i>node</i> untuk bertukar pesan.	

(Sumber: Dennis, 2010)

2.13. *Entity Relationship Diagram (ERD)*

Menurut Rosa dan Shalahuddin, 2015, ERD adalah pemodelan awal basis data yang dikembangkan berdasarkan teori himpunan dalam bidang matematika. ERD digunakan untuk pemodelan basis data relasional. (Rosa dan Shalahuddin, 2015) menjelaskan simbol-simbol yang digunakan dalam ERD, yaitu:

Tabel II.8 Simbol-simbol *Entity Relationship Diagram (ERD)*

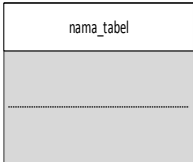
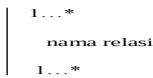
No.	Simbol	Deskripsi
1.	Entitas 	Entitas merupakan data inti yang akan disimpan
2.	Atribut 	Field atau kolom data yang butuh disimpan dalam suatu entitas
3.	Atribut kunci primer 	Field atau kolom data yang butuh disimpan dalam suatu entitas dan digunakan sebagai kunci akses record yang diinginkan.
4.	Atribut multinilai / <i>multivalve</i> 	Field atau kolom data yang butuh disimpan dalam suatu entitas yang dapat memiliki nilai lebih dari satu
5.	Relasi 	Relasi yang menghubungkan antar entitas, biasanya diawali dengan kata kerja
6.	Asosiasi 	Penghubung antara relasi dan entitas

(Sumber: Rosa dan Shalahuddin, 2015)

2.14. *Conceptual Data Model (CDM)*

Menurut Rosa dan Shalahuddin, 2016, CDM (*Conceptual Data Model*) atau model konsep data adalah konsep yang berkaitan dengan pandangan pemakai terhadap data yang disimpan dalam basis data. CDM dibuat sudah dalam bentuk tabel-tabel tanpa tipe data yang menggambarkan relasi antar tabel untuk keperluan implementasi ke basis data. CDM merupakan hasil penjabaran lebih lanjut dari ERD. Berikut adalah simbol-simbol yang ada pada CDM:

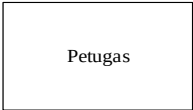
Tabel II.9 Simbol-simbol *Conceptual Data Model (CDM)*

Simbol	Deskripsi
Entitas/tabel 	Entitas atau tabel yang menyimpan data dalam basis data
Relasi 	Relasi antar tabel yang terdiri atas nama relasi dan <i>multiplicity</i>

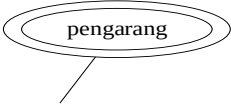
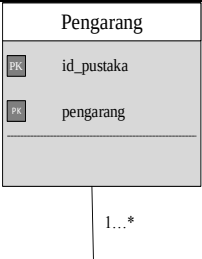
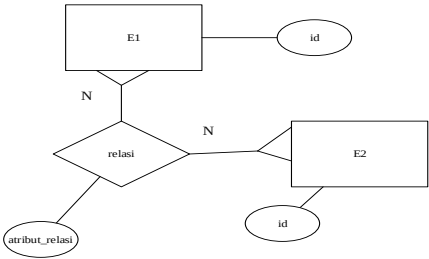
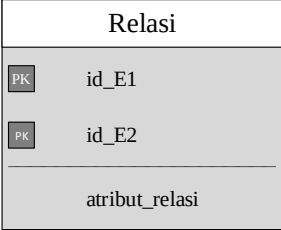
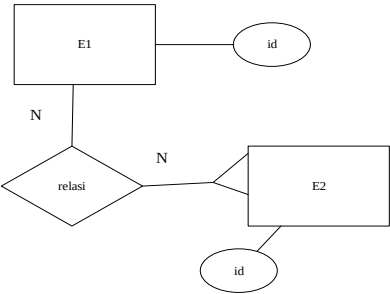
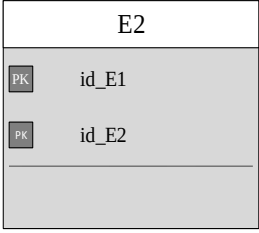
(Sumber: Rosa dan Shalahuddin, 2016)

Aturan untuk mengubah ERD menjadi CDM secara umum adalah sebagai berikut:

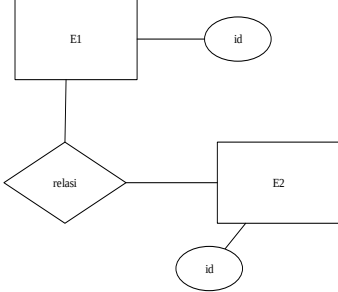
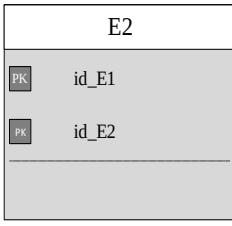
Tabel II.10 Simbol-simbol Aturan *Conceptual Data Model (CDM)*

ERD	CDM
 Entitas	 Menjadi sebuah tabel tersendiri

Tabel II.10 Simbol-simbol Aturan *Conceptual Data Model* (CDM) (Lanjutan)

ERD	CDM
 Atribut <i>multivalue</i>	 Menjadi sebuah tabel tersendiri dengan kunci primer (primary key) adalah kunci primer pada entitas dan memiliki atribut dengan nama seperti pada atribut entitas
 Relasi dengan kardinalitas <i>many to many</i>	 Menjadi sebuah tabel tersendiri dengan kunci primer adalah atribut yang menjadi kunci primer di kedua entitas yang direlasikannya.
 Relasi dengan kardinalitas <i>one to many</i>	 Kunci primer entitas yang memiliki hubungan <i>one</i> akan dijadikan kunci primer di entitas yang memiliki hubungan <i>many</i> dengan kata lain, relasi tidak menjadi tabel sendiri.

Tabel II.10 Simbol-simbol Aturan *Conceptual Data Model* (CDM) (Lanjutan)

ERD	CDM
 Relasi dengan kardinalitas <i>one to one</i>	 Kunci primer salah satu entitas akan dijadikan kunci asing (<i>foreign key</i>) pada tabel yang lain dan kunci asing itu dijadikan kunci primer juga, dengan kata lain, relasi tidak menjadi tabel sendiri

(Sumber: Rosa dan Shalahuddin, 2016)

2.15. Kamus Data

Kamus data adalah kumpulan daftar elemen data yang mengalir pada sistem perangkat lunak sehingga masukan (*input*) dan keluaran (*output*) dapat dipahami secara umum (memiliki standar penulisan). Kamus data dalam implementasi program dapat menjadi parameter masukan atau keluaran dari sebuah fungsi atau prosedur. Kamus data biasanya berisi (Rosa dan Shalahuddin, 2014):

1. Nama, nama dari data.
2. Digunakan pada, merupakan proses-proses yang terkait data.
3. Deskripsi, merupakan deskripsi data.
4. Informasi tambahan, seperti tipe data, nilai data, batas nilai data, dan komponen yang membentuk data.

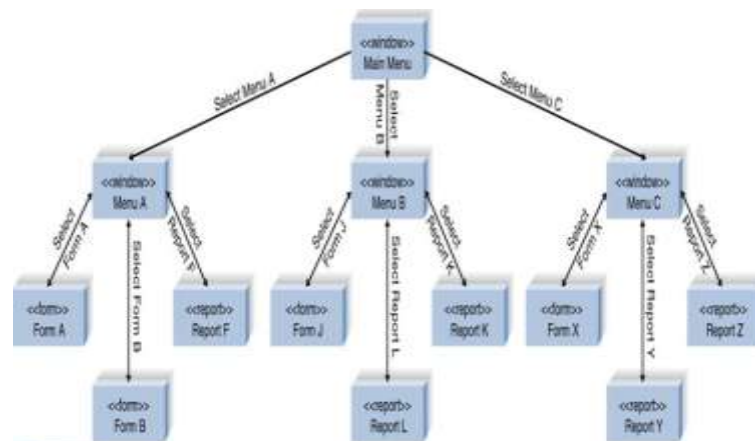
2.16. *Windows Navigation Diagram* (WND)

Windows Navigation Diagram (WND) Menurut Dennis (2010):

1. Menunjukkan bagaimana semua layar, formulir dan laporan terkait.
2. Menunjukkan bagaimana pengguna bergerak dari satu ke yang lain.
3. Seperti diagram keadaan untuk *user interface*

- Kotak mewakili komponen
- Panah mewakili transisi
- Stereotip menunjukkan tipe antarmuka

Contoh Diagram WND menurut Dennis dapat dilihat seperti gambar dibawah ini:



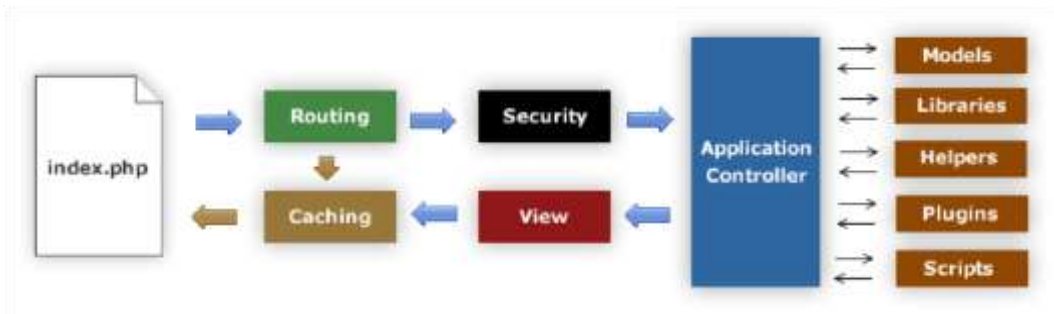
Gambar II.4 Contoh *Windows Navigation Diagram*
(Sumber: Dennis, 2010)

2.17. Codeigniter

Menurut Septian (2011), codeigniter adalah aplikasi yang berupa *open source* yang berupa *framework* dengan model MVC (Model, View, Controller) untuk membangun *website* dinamis dengan cepat dan mudah. CodeIgniter bertujuan untuk menghasilkan *framework* yang dapat digunakan untuk pengembangan proyek pembuatan situs web. CodeIgniter memiliki desain dan struktur file yang sederhana, didukung dengan dokumentasi yang lengkap sehingga *framework* lebih mudah dipelajari.

CodeIgniter ini memungkinkan para pengembang untuk menggunakan *framework* secara parsial atau secara keseluruhan. Artinya bahwa CodeIgniter masih memberi kebebasan kepada para pengembang untuk menulis bagian-bagian kode tertentu di dalam aplikasi menggunakan cara konvensional atau dengan *syntax* umum didalam PHP, tidak harus menggunakan aturan penulisan kode di

codeigniter. Adapun alur dari aplikasi yang ditulis menggunakan CodeIgniter seperti pada gambar II.4.



Gambar II.5 Alur Aplikasi CodeIgniter

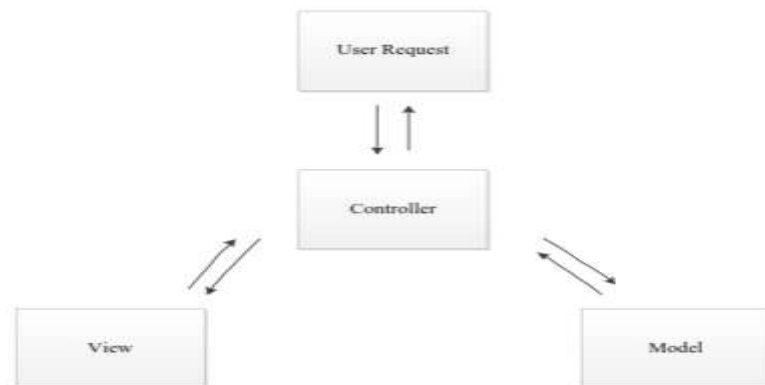
(Sumber: [http:// idcloudhost.com/wp-content/uploads/2017/08/IDCloudHost.pdf](http://idcloudhost.com/wp-content/uploads/2017/08/IDCloudHost.pdf))

Berikut penjelasan dari alur aplikasi dari CodeIgniter yaitu:

1. File `index.php`, berfungsi sebagai *front controller*, menginisialisasi *resource* utama yang dibutuhkan untuk menjalankan CodeIgniter.
2. *Router*, memeriksa *HTTP request* untuk menentukan apa yang harus dilakukan.
3. Jika file cache ada, dikirim langsung ke *browser*, melewati eksekusi sistem normal.
4. Keamanan, sebelum *controller* aplikasi dimuat, *HTTP request* dan setiap data pengguna yang *submit* disaring terlebih dahulu untuk keamanan.
5. *Controller*, memuat model, *library* utama, *helper*, dan setiap *resource* lainnya yang diperlukan untuk memproses permintaan khusus.
6. *View*, proses *render* kemudian dikirim ke web *browser* agar dapat dilihat. Jika *caching* diaktifkan, *view* dicache terlebih dahulu sehingga pada permintaan berikutnya dapat dilayani.

2.18. Model MVC

CodeIgniter adalah *framework* PHP yang dibuat berdasarkan kaidah model- *view-controller*. Dengan MVC, maka memungkinkan pemisahan antara *layer application-logic* dan *presentation*. MVC adalah konsep dasar yang harus diketahui sebelum mengenal CodeIgniter. MVC (Model View Controller) merupakan suatu konsep yang cukup populer dalam pembangunan aplikasi web, berawal pada bahasa pemrograman Small Talk, yang memisahkan bisnis logic (alur pikir), data logic (penyimpanan data) dan presentation logic (antarmuka aplikasi) atau secara sederhana adalah memisahkan antara desain, data dan proses.



Gambar II.6 Model-View-Controller
(Sumber: Hakim, 2010)

Gambar diatas menerangkan bahwa ketika datang sebuah *user request*, maka akan ditangani oleh *controller*, kemudian *controller* akan memanggil model jika memang diperlukan operasi *database*. Hasil dari *query* oleh *model* kemudian akan dikembalikan ke *controller*. Selanjutnya *controller* akan memanggil *view* yang tepat dan mengkombinasikannya dengan hasil *query model*. Hasil akhir dari operasi ini akan ditampilkan di *browser*.

Dalam konteks codeigniter dan aplikasi berbasis web, maka penerapan konsep MVC mengakibatkan kode program dapat dibagi menjadi tiga kategori, yaitu:

1. Model, Kode program (berupa OOP *class*) yang digunakan untuk memanipulasi *database*.
2. View, Berupa *template* html/xml atau php untuk menampilkan data pada *browser*
3. *Controller*, Kode program (berupa OOP *class*) yang digunakan untuk mengontrol aliran aplikasi (sebagai pengontrol *model* dan *View*)

2.19. MySQL

Menurut Anhar (2010) MySQL adalah sebuah perangkat lunak sistem manajemen basis data (DBMS) dari sekian banyak DBMS seperti Oracle, MSSQL, Postgre SQL, dan lain-lain. MySQL merupakan DBMS yang multithread, multi-user yang bersifat gratis dibawah GNU *General Public Licence* (GPL). MySQL memiliki beberapa kelebihan antara lain:

1. Portabilitas

MySQL dapat berjalan stabil pada berbagai sistem operasi seperti Windows, Linux, FreeBSD, Mac Os X Server, Solaris, Amiga, dan masih banyak lagi.

2. *Open Source*

MySQL didistribusikan secara *open source*, dibawah lisensi GPL sehingga dapat digunakan secara cuma-cuma.

3. *Multi User*

MySQL dapat digunakan oleh beberapa *user* dalam waktu yang bersamaan tanpa mengalami masalah atau konflik.

4. *Performance tuning*

MySQL memiliki kecepatan yang menakjubkan dalam menangani *query* sederhana, dengan kata lain dapat memproses lebih banyak SQL per satuan waktu.

5. Jenis Kolom

MySQL memiliki tipe kolom yang sangat kompleks, seperti *signed* atau *unsigned integer*, *float*, *double*, *char*, *text*, *date*, *timestamp*, dan lain-lain.

6. Perintah dan Fungsi

MySQL memiliki operator dan fungsi secara penuh yang mendukung perintah *select* dan *where* dalam perintah (*query*).

7. Keamanan

MySQL memiliki beberapa lapisan sekuritas seperti level *subnetmask*, nama *host*, dan izin akses *user* dengan sistem perizinan yang mendetail serta sandi terenkripsi.

8. Konektivitas

MySQL dapat melakukan koneksi dengan klien menggunakan protokol TCP/IP, Unix socket (UNIX), atau Named Pipes (NT).

9. Lokalisasi

MySQL dapat mendeteksi pesan kesalahan pada klien dengan menggunakan lebih dari dua puluh bahasa. Meskipun demikian, bahasa Indonesia belum termasuk di dalamnya.

10. Antar Muka

MySQL memiliki *interface* (antar muka) terhadap berbagai aplikasi dan bahasa pemrograman dengan menggunakan fungsi API (*Application Programming Interface*).

11. Klien dan Peralatan

MySQL dilengkapi dengan berbagai peralatan (*tools*) yang dapat digunakan untuk administrasi basis data, dan pada setiap peralatan yang ada disertakan petunjuk *online*.

2.19.1. Tipe Data MySQL

MySQL mendukung banyak tipe data yang dapat disimpan pada sebuah kolom. Penjelasan singkat kegunaan masing-masing jenis data akan sangat bermanfaat dalam memilih suatu jenis data yang dipakai dalam merancang *table*. Beberapa jenis data yang tersedia pada MySQL dapat dilihat pada Tabel II.7 (Sutaji, 2012).

Tabel II.11 Tipe Data pada MySQL

Jenis Data	Keterangan
CHAR	Sebuah <i>string</i> dengan panjang tetap. Sisa Jumlah karakter yang belum terisi akan diisi dengan spasi, akan tetapi spasi ini dibuang jika data dipanggil. Jangkauan nilai M adalah 1-255 karakter
VARCHAR	<i>String</i> dengan panjang berupa variabel. M bisa mencapai 65535
DATE	Data berupa tanggal. Format tanggal dalam bentuk ‘YYYY-MM-DD’
TIME	Data berupa waktu. Format waktu dalam bentuk ‘HH:MM:SS’
TINYINT	Bilangan antara -128 sampai dengan +127
SMALLINT	Bilangan antara -32768 sampai dengan +32767
INT	Bilangan antara -2147683648 sampai dengan +2147683647
FLOAT	Bilangan <i>floating point</i> yang kecil (presisi tunggal). Jangkauan nilai yang diizinkan adalah -3.402823466E+38 s/d -1.175494351e-38,0 dan 1.175494351E-38 s/d 3.402823466E+38
DOUBLE	Bilangan <i>floating point</i> dengan ukuran normal (presisi ganda). Jangkauan nilai yang diizinkan adalah -1.7976931348623157E+308 s/d -2.225073858507201E-308,0 dan 2.225073858507201E-308 s/d 1.7976931348623157E+308
ENUM	Sebuah <i>enumeration</i> . Sebuah obyek string yang hanya boleh memiliki satu nilai, yang terambil dari ‘value1’, ‘value2’, ..., NULL atau nilai spesial “”error. Sebuah enum dapat menampung 65535 pilihan nilai
TEXT, BLOB	Sebuah TEXT atau BLOB dengan panjang karakter maksimum 65535 karakter

(Sumber: Sutaji, 2012)

2.20. XAMPP

XAMPP adalah sebuah *software* yang berfungsi untuk menjalankan *website* berbasis PHP dan menggunakan pengolahan data MySQL di komputer lokal. XAMPP berperan sebagai *web server* pada komputer. XAMPP juga dapat disebut sebagai sebuah *CPanel server virtual*, yang dapat membantu melakukan *preview* sehingga dapat memodifikasi *website* tanpa harus *online* atau terakses internet (Kadir, 2014).

Fungsi lainnya adalah sebagai server yang berdiri sendiri (*localhost*), yang terdiri atas program Apache HTTP Server, MySQL *database*, dan penerjemah bahasa yang ditulis dengan bahasa pemrograman PHP dan Perl. Nama XAMPP merupakan singkatan dari X, Apache, MySQL, PHP dan Perl. Program ini tersedia dalam GNU *General Public License* dan bebas, merupakan web server yang mudah digunakan yang dapat melayani tampilan halaman web yang dinamis.

MySQL mendukung banyak tipe data yang dapat disimpan pada sebuah kolom. Penjelasan singkat kegunaan masing-masing jenis data akan sangat bermanfaat dalam memilih suatu jenis data yang dipakai dalam merancang tabel.

BAB III

METODOLOGI PENELITIAN

3.1. Metodologi Penelitian

Menurut Achmadi dan Narbuko (2016), metodologi penelitian merupakan ilmu yang mempelajari cara-cara melakukan pengamatan dengan pemikiran yang tepat secara terpadu melalui tahapan-tahapan yang disusun secara ilmiah untuk mencari, menyusun serta menganalisis dan menyimpulkan data-data, sehingga dapat dipergunakan untuk menemukan, mengembangkan dan menguji kebenaran suatu pengetahuan. Metode penelitian ini terdiri dari beberapa tahapan, yaitu tahapan pengumpulan data atau informasi dan pengembangan sistem. Tahap pengumpulan data dan informasi dilakukan dengan cara observasi langsung di tempat penelitian, wawancara dengan pengguna sistem yang diamati serta studi kepustakaan. Pada tahap pengembangan sistem, pengembangan sistem yang digunakan adalah model prototipe.

3.2. Jenis dan Sumber Data

Data yang diperlukan dalam penelitian ini ada dua jenis, yaitu data primer dan data sekunder. Sumber dari data-data ini berasal dari tempat yang diamati pada praktik kerja lapangan di PT Krama Yudha Ratu Motor diantaranya:

1. Data Primer

Data Primer adalah data yang diperoleh secara langsung dari PT Krama Yudha Ratu Motor, dimana pengumpulan data atau informasi dilakukan langsung dari objek yang diteliti. Data-data tersebut adalah data yang digunakan dalam proses produksi diantaranya struktur organisasi, analisis sistem yang telah berjalan, proses bisnis sistem saat ini dan yang akan diusulkan, dan kebutuhan pengguna sistem.

2. Data Sekunder

Data sekunder adalah data yang diperoleh dari data yang tersedia dan telah terlebih dahulu dikumpulkan dan dilaporkan oleh orang lain, buku-buku dan kajian ilmiah dari berbagai sumber yang berkaitan dengan penelitian. Data tersebut adalah data umum perusahaan, profil perusahaan, dan struktur organisasi perusahaan.

3.3. Metode Pengumpulan Data

Menurut Sugiyono (2013) teknik pengumpulan data merupakan langkah yang paling strategis dalam penelitian, karena tujuan utama dari penelitian adalah mendapatkan data. Data yang diperoleh dari responden secara langsung yang dikumpulkan melalui *survey* lapangan dengan menggunakan teknik pengumpulan data. Teknik pengumpulan data dalam rangka pengumpulan informasi mengenai objek penelitian ini, yaitu:

1. Observasi

Observasi yaitu melakukan pengamatan langsung terhadap sistem yang sedang berjalan di Bagian *Trimming* 3 pada PT Krama Yudha Ratu Motor. Hasil dari pengamatan yang dilakukan menjadi landasan penulis dalam melakukan pengembangan sistem yang akan dibuat.

2. Wawancara

Metode wawancara adalah metode pengumpulan data melalui tanya jawab secara langsung kepada pihak-pihak yang terkait dengan objek penelitian. Pihak yang diwawancarai adalah Staf Bagian *Trimming* 3 dengan meminta keterangan yang lebih jelas kepada pembimbing atau karyawan perusahaan tentang bagaimana proses produksi kendaraan tipe Colt T120 SS, bagaimana jika ada part yang rusak saat proses produksi berjalan dan bila perlu meminta bahan-bahan atau data yang diperlukan untuk kelengkapan laporan.

3. Studi Pustaka

Studi pustaka yaitu metode pengumpulan data dengan melakukan pencarian referensi yang berkaitan dengan judul dan permasalahan, sehingga dapat menunjang dalam penulisan tugas akhir ini, seperti teori tentang Sistem

Informasi, UML, CodeIgniter, *database* MySQL, dari berbagai referensi, baik itu referensi yang didapat dari buku, jurnal, maupun dari media internet yang berhubungan dengan masalah yang sedang dianalisis.

3.4. Metode Pengembangan Sistem

Metodologi pengembangan sistem merupakan kerangka formal dalam mengimplementasikan konsep SDLC untuk mengembangkan atau mengubah suatu sistem perangkat lunak dengan menggunakan model-model dan metodologi yang digunakan orang untuk mengembangkan sistem-sistem perangkat lunak sebelumnya berdasarkan *best practice* atau cara-cara yang sudah teruji baik (Rosa dan Shalahuddin, 2015).

Pada penelitian ini untuk mengatasi masalah yang ada pada sistem, diputuskan untuk membuat pengembangan sistem. Dalam pengembangan sistem ini digunakan metode *evolutionary prototype*. Tahapan-tahapan dalam *evolutionary prototype* adalah sebagai berikut:

1. Identifikasi kebutuhan pengguna
Pengembang melakukan diskusi dengan Bagian *Trimming* 3 tentang kebutuhan sistem yang digunakan.
2. Mengembangkan *prototype*
Pengembang membuat *prototype* dari sistem yang telah dijelaskan oleh pihak terkait.
3. Menyesuaikan dan evaluasi *prototype* dengan keinginan *user*
Pengembang menanyakan Bagian *Trimming* 3 tentang *prototype* yang sudah dibuat, apakah sesuai atau tidak dengan kebutuhan sistem.
4. Menggunakan *prototype*
Sistem mulai dikembangkan dengan *prototype* yang sudah dibuat.

3.5 Kerangka Penelitian

Dalam penelitian yang dibahas dalam tugas akhir ini, dilakukan langkah-langkah atau tahapan dalam penelitian yang sesuai dengan metodologi penelitian

yang digunakan dalam penelitian tersebut. Penjelasan langkah-langkah atau tahapan dalam penelitian tersebut adalah sebagai berikut:

1. Studi pendahuluan

Yang dilakukan yaitu dengan mengetahui terlebih dahulu proses yang berjalan pada PT Krama Yudha Ratu Motor. Studi pendahuluan ini dimaksudkan untuk dapat mengetahui gambaran yang jelas mengenai kondisi dan situasi di bagian *Trimming* 3, serta untuk mengetahui masalah yang sedang dihadapi. Langkah-langkah yang dilakukan dalam studi pendahuluan yang bertujuan untuk mendapatkan informasi adalah dengan cara observasi dan melakukan wawancara terhadap pihak-pihak terkait yang berada di Bagian *Trimming* 3 dan bagian lainnya yang terlibat dalam proses produksi.

2. Identifikasi Masalah

Pokok dari permasalahan yang menjadi objek penelitian ini adalah proses pengolahan data produksi yang masih manual dan semi komputerisasi dengan menggunakan aplikasi sederhana seperti *Microsoft Excel*. Untuk proses pengolahan data pada sistem informasi pelaporan kerusakan part di bagian *Trimming* 3 masih belum terintegrasi antar bagian dimana ketika terjadi kerusakan part saat proses produksi berlangsung harus mendatangi bagian *Part Control* untuk memberikan line damage agar part yang rusak tadi diganti, dan belum ada media penyimpanan data seperti *database*.

3. Studi Literatur

Penulis melakukan studi literatur yang lebih mendalam dengan tujuan untuk mendapatkan landasan teori yang dapat mendukung pemecahan masalah yang sedang diteliti. Teori yang didapatkan merupakan langkah awal peneliti agar lebih memahami permasalahan yang sedang diteliti dengan benar sesuai dengan kerangka berpikir ilmiah.

4. Analisis kebutuhan *user*

Melakukan analisis kembali terhadap data yang telah didapat pada penelitian sebelumnya, sehingga dapat diketahui kebutuhan *user* terhadap aplikasi yang akan dibuat.

5. Pembuatan *Prototype*

Merancang *prototype* sesuai dengan analisis dan perancangan yang dilakukan.

- a. Pemodelan sistem dengan *use case diagram*, *activity diagram*, *sequence diagram*, *class diagram* dan *Deployment diagram*.
- b. Pemodelan data dengan menggunakan *Entity Relationship Diagram* dan Kamus Data.
- c. Perancangan sistem dengan menggunakan WND (*Windows Navigation Diagram*) dan perancangan antar muka.
- d. Pembuatan *prototype* dengan menggunakan *CodeIgniter* 3.1.7 dan MySQL 5.5.32.

6. Menyesuaikan *Prototype* dengan Keinginan *User*

Pada tahap ini, *prototype* yang diusulkan akan diperlihatkan dan diuji coba terlebih dahulu kepada *user* apakah sudah sesuai dengan keinginan *user* atau belum. Jika belum maka dilakukan analisis kembali, tetapi apabila sesuai maka tahap selanjutnya adalah implementasi penggunaan *prototype* tersebut.

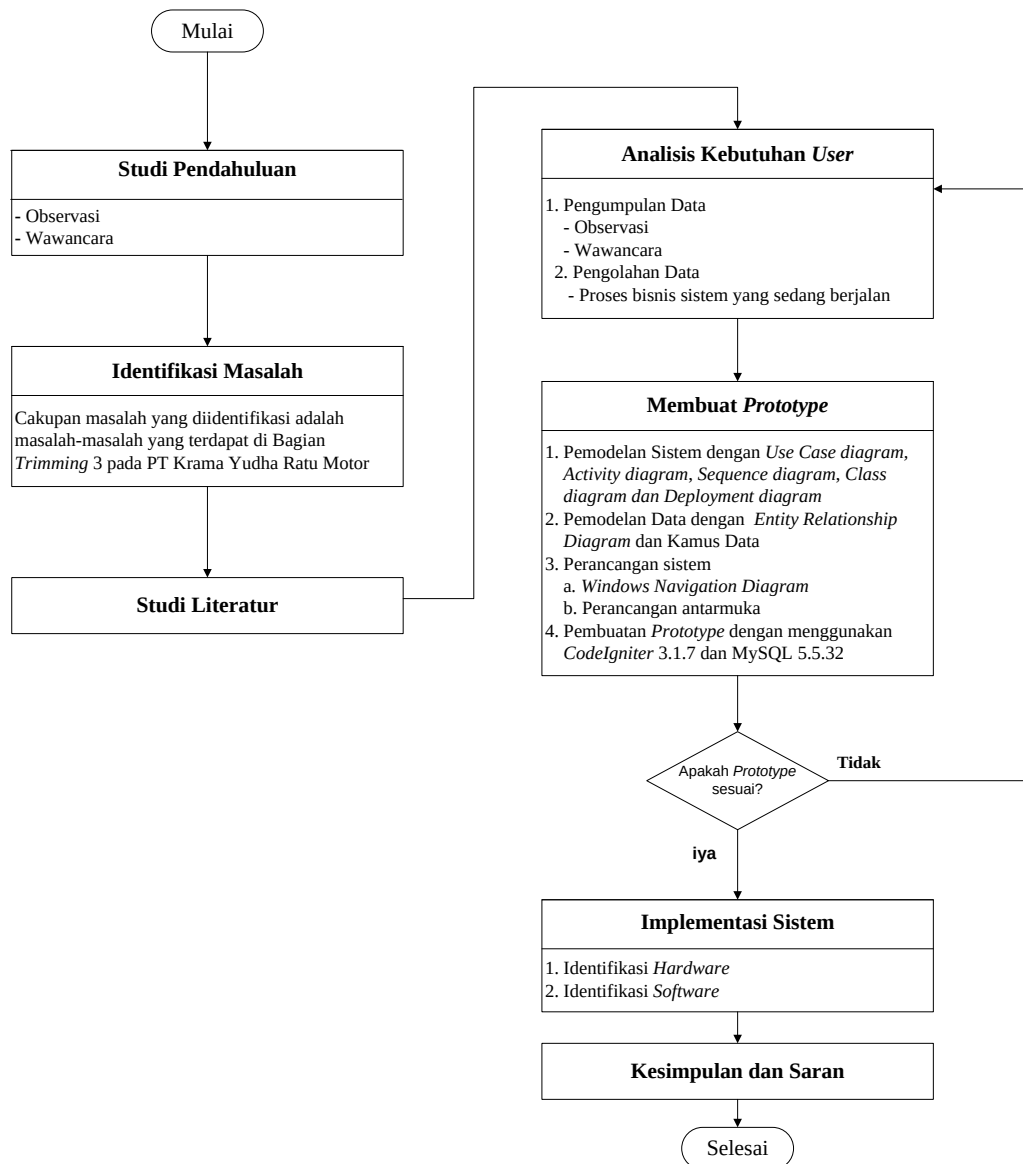
7. Implementasi Sistem

Sebelum melakukan implementasi sistem usulan, ada 2 (dua) hal yang harus dipenuhi yaitu perangkat keras yang akan digunakan seperti laptop atau *Personal Computer* (PC), dan perangkat lunak seperti *CodeIgniter* dan *MySQL Service*.

8. Kesimpulan dan Saran

Kesimpulan digunakan untuk membandingkan hasil penelitian atau pengembangan sistem dengan sistem sebelumnya. Sedangkan saran berisi masukan untuk pengembangan sistem selanjutnya.

Kerangka penelitian dibuat dalam bentuk *flowchart* yang menggambarkan tahap-tahap kegiatan mulai dari awal hingga akhir dapat dilihat pada Gambar III.1.



Gambar III.1 Kerangka Penelitian
(Sumber: Pengolahan Data, 2018)

BAB IV

PENGUMPULAN DAN PENGOLAHAN DATA

4.1. Sejarah Singkat Perusahaan

PT Krama Yudha Ratu Motor (KRM) didirikan pada tanggal 1 Juni 1973 sebagai perusahaan swasta yang merupakan sebuah perusahaan yang merakit berbagai macam kendaraan niaga Mitsubishi di Indonesia. PT KRM ini merupakan dari bagian Krama Yudha Mitsubishi Group (KYMG). Awal berdirinya KYMG adalah akibat dari banyaknya kendaraan bermotor dari eropa yang diimpor ke Indonesia untuk mengurangi pengimporan kendaraan tersebut maka para pengusaha melakukan pertemuan dan sepakat mendirikan suatu perakitan kendaraan bermotor di Indonesia dengan menggunakan lisensi dari Mitsubishi motor *corporation* Jepang.

PT Krama Yudha Ratu Motor mengacu pada ISO 9001:2008 dan ISO 14001:2004 dengan subjek utama perakitan / *assembling* kendaraan bermotor roda 4 dan atau lebih melakukan pengendalian sistem manajemen mutu seperti penerimaan barang, penyimpanan, distribusi ke line produksi untuk selanjutnya melakukan perakitan pengelasan di bagian *welding*, pengecatan di bagian *painting* dan perakitan / pemasangan *parts* di bagian *trimming*.

PT KRM terletak di Jl. Raya Bekasi Km. 21-22 Jakarta Timur bersama-sama dengan dua buah perusahaan Krama Yudha Grup lainnya yaitu PT Mitsubishi Krama Yudha Manufacturing I dan II (MKM) dan PT Krama Yudha Tiga Berlian Motors (KTB). PT. KRM menempati lahan seluas 343,354m² dengan luas bangunan pabrik 41,950m² dan bangunan gudang seluas 68,330 m² dan bangunan kantor seluas 30,420 m².

4.2. Profil Perusahaan

Nama Perusahaan	: PT Krama Yudha Ratu Motor
Tanggal berdiri	: 01 Juni 1973
Jumlah karyawan	: 1,063 (April 2017)

Alamat	: Jl. Raya Bekasi Km. 21-22 Rawa Terate Cakung, Jakarta - Indonesia
Telp	: 021-4602905
Fax	: 021-4602904
Jumlah Karyawan	: 1,063 (April 2017)
Jam Kerja	
Senin s/d Kamis	: Jam 07.10 s/d 16.20 WIB
• Istirahat 1	: 10.00-10.10
• Istirahat 2	: 11.35-12.25
• Istirahat 3	: 14.00-14.10
Jumat	: Jam 07.10 s/d 16.20 WIB
• Istirahat 1	: 10.00-10.10
• Istirahat 2	: 11.40-13.00
• Istirahat 3	: 15.00-15.10

4.3. Visi dan Misi Perusahaan

Guna meningkatkan kualitas dan mutu produknya, PT Krama Yudha Ratu Motor memiliki visi dan misi sebagai berikut:

Visi :

Menjadi Perusahaan perakitan terunggul dan sanggup bersaing ditingkat regional maupun global.

Misi :

- Menjadi Perusahaan yang cukup memuaskan *Shareholder* dan kesejahteraan yang baik bagi seluruh karyawan.
- Membangun sumber daya manusia yang handal dan sanggup mengantisipasi perkembangan regional maupun global.
- Komit akan industri yang ramah lingkungan

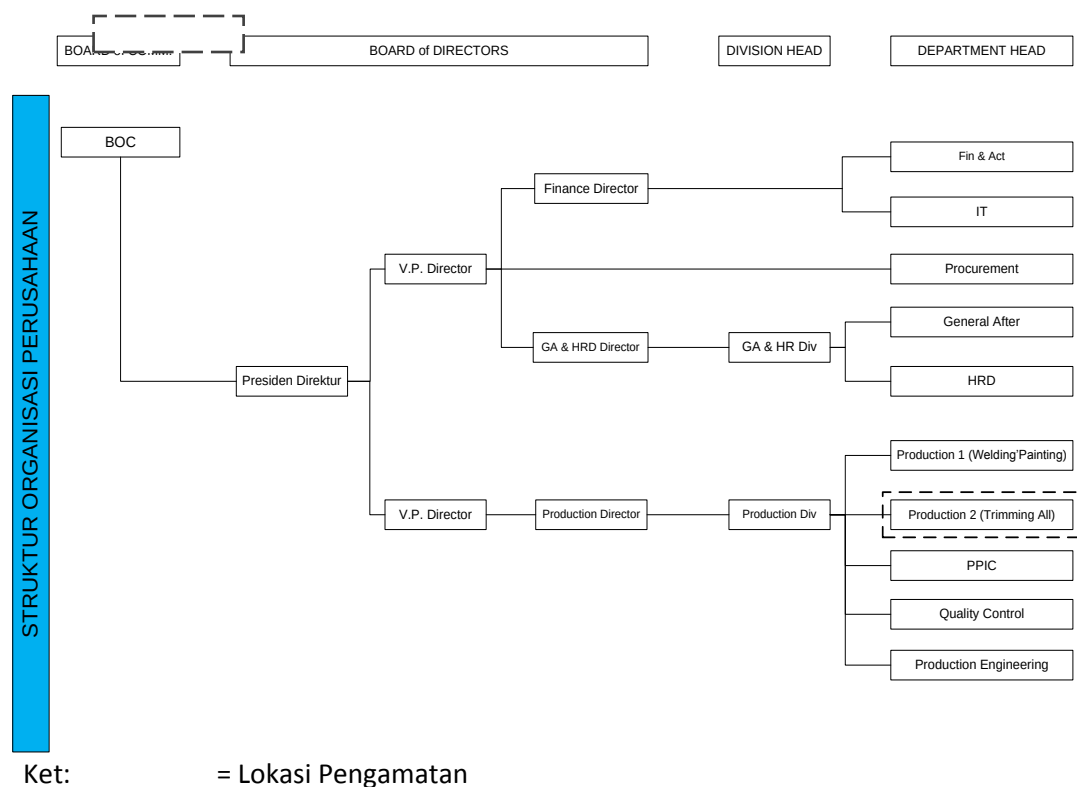
4.4. Struktur Organisasi Perusahaan

Dalam suatu perusahaan baik itu perusahaan kecil maupun besar, struktur organisasi sangatlah penting, karena struktur organisasi merupakan alur *job description*

dalam pelaksanaan kerja yang baik dan terarah, serta dapat diketahui batas tanggung jawab dari suatu pekerjaan.

Untuk menunjang kegiatan perusahaan dan untuk mempermudah pengawasan dalam kegiatan produksi maka dibentuklah suatu struktur organisasi dari PT Krama Yudha Ratu Motor yang akan memudahkan dalam melakukan koordinasi dan hubungan yang dapat mempengaruhi efektifitas dan efisiensi suatu perusahaan didalam mencapai tujuan yang telah ditetapkan sebelumnya.

Berikut ini merupakan struktur organisasi dari PT Krama Yudha Ratu Motor secara umum.

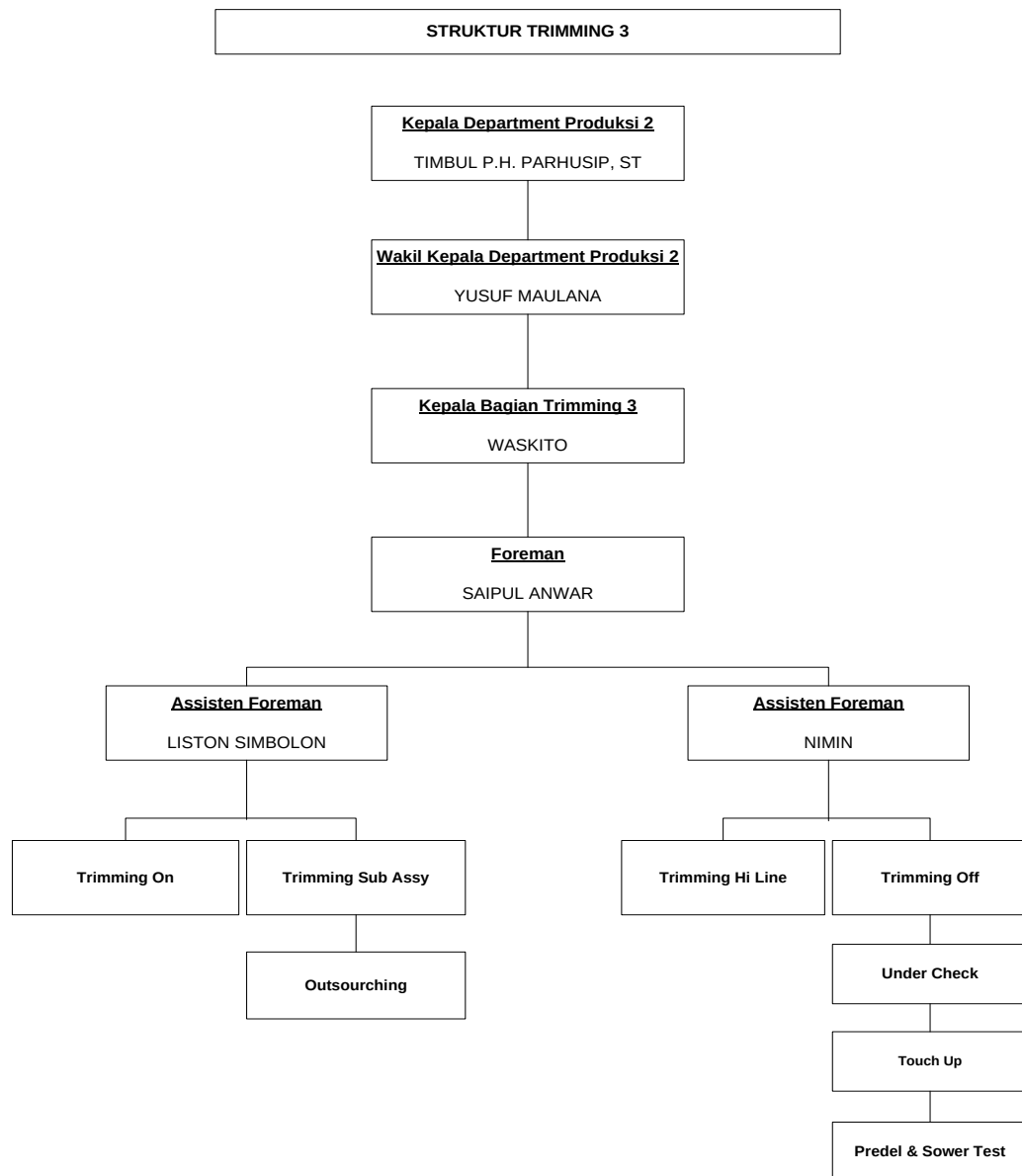


Gambar IV.1 Struktur Organisasi PT Krama Yudha Ratu Motor

(Sumber : PT Krama Yudha Ratu Motor, 2017)

4.5. Struktur Organisasi *Trimming* 3

Adapun struktur organisasi *trimming 3* PT Krama Yudha Ratu Motor sebagai berikut:



Gambar IV.2 Struktur Organisasi *Trimming 3*
(Sumber : PT Krama Yudha Ratu Motor, 2017)

Tugas dan tanggung jawab dari masing-masing bagian terutama yang digaris bawahhi dalam struktur organisasi *trimming 3* pada kendaraan tipe CJM adalah sebagai berikut:

1. Kepala Department Produksi

Mengkoordinir, mengendalikan proses produksi yang sesuai dengan kualitas dan jumlah yang telah ditentukan, terpeliharanya lingkungan kerja serta keselamatan dan kesehatan kerja.

Tugas dan tanggung jawab :

- a. Membuat rencana kerja dan melaksanakan pengawasan serta pengendalian pada bagian *Welding*, *Painting* dan *Trimming*.
- b. Memberdayakan dan memonitor kegiatan proses produksi dengan menggunakan tenaga kerja yang cukup sesuai dengan kondisi alat dan mesin agar mencapai produktivitas sesuai standard.
- c. Mengevaluasi dan mengembangkan Sumber Daya Manusia (SDM) secara terpadu untuk mencapai sasaran yang telah ditentukan.
- d. Menjalankan, memelihara, meningkatkan sistem manajemen perusahaan.
- e. Melakukan identifikasi, mengelola dan memonitoring lingkungan ditempat kerja masing – masing.
- f. Pengendalian atau mencegah kecelakaan kerja dengan cara : Eliminasi, Substitusi, Rekayasa Teknik, Administrasi dan APD.

2. Wakil Kepala Departement Produksi

Membantu / mewakili kepala departemen produksi dalam melaksanakan pengawasan serta pengendalian proses produksi untuk menghasilkan produk sesuai dengan kualitas dan jumlah yang telah di tentukan, terpeliharanya lingkungan kerja serta keselamatan dan kesehatan kerja.

Tugas dan tanggung jawab:

- a. Membantu / mewakili kepala departemen membuat rencana dan melaksanakan pengawasan serta pengendalian pada bagian *Welding*, *Painting* dan *Trimming*.

- b. Membantu / mewakili kepala departemen memberdayakan dan memonitor kegiatan proses produksi dengan menggunakan tenaga kerja yang sesuai kondisi alat dan mesin agar mencapai produktivitas sesuai standard.
- c. Membantu / mewakili kepala departemen mengevaluasi dan mengembangkan Sumber Daya Manusia (SDM) secara terpadu untuk mencapai sasaran yang telah ditetapkan.
- d. Melaksanakan tugas kepala bagian, bila kepala bagian mendapat tugas keluar perusahaan/berhalangan.
- e. Menjalankan, memelihara, meningkatkan sistem manajemen perusahaan.
- f. Melakukan identifikasi, mengelolah dan memonitoring lingkungan ditempat kerja masing – masing.
- g. Pengendalian atau mencegah kecelakaan kerja dengan cara : Eliminasi, Subtitusi, Rekayasa Teknik, Administrasi dan APD.

3. Kepala Bagian *Trimming* 3

Mengkoordinasikan, mengendalikan proses produksi kendaraan CJM di *trimming* 3 dan *pre – delivery* sesuai standard yang berlaku untuk menghasilkan produk sesuai mutu dan jumlah yang telah ditentukan, terpeliharanya lingkungan kerja serta keselamatan dan kesehatan kerja.

Tugas dan tanggung jawab:

- a. Mengendalikan *Line Damage*.
- b. Memberdayakan dan memonitor kegiatan proses produksi dengan menggunakan tenaga kerja yang cukup sesuai kondisi alat dan mesin agar mencapai produktivitas sesuai standard.
- c. Memelihara lingkungan kerja, alat-alat, mesin-mesin dengan melaksanakan program K3, 5R dan GKM.
- d. Pengendalian atau mencegah kecelakaan kerja dengan cara: Eliminasi, Subtitusi, Rekayasa Teknik, Administrasi dan APD.
- e. Menjalankan, memelihara, meningkatkan sistem manajemen perusahaan.
- f. Melakukan identifikasi, mengelolah dan memonitoring lingkungan

ditempat kerja masing – masing.

4. *Foreman*

Melakukan pengawasan, koordinasi dan pengendalian proses produksi kendaraan CJM di *trimming* 3 sehingga dapat di *delivery* sesuai dengan standard, terpeliharanya lingkungan kerja serta keselamatan dan kesehatan kerja

Tugas dan tanggung jawab:

- a. Merealisasikan *schedule* produksi yang dibuat PPC sesuai standard mutu dan jumlah.
- b. Mengevaluasi, mengarahkan dan membina karyawan sehingga menghasilkan karyawan yang bertanggung jawab, disiplin dan berprestasi guna tercapainya sasaran mutu.
- c. Melakukan pengukuran *cycle time* sehingga tercapai *line balance* / pembagian kerja yang merata.
- d. Memelihara lingkungan kerja, alat-alat dan mesin-mesin dengan cara melaksanakan program K3, 5R dan GKM.
- e. Pengendalian atau mencegah kecelakaan kerja dengan cara : Eliminasi, Substitusi, Rekayasa Teknik, Administrasi dan APD.
- f. Melakukan absensi bawahan
- g. Menjalankan, memelihara, meningkatkan sistem manajemen perusahaan.
- h. Melakukan identifikasi, mengelola dan memonitoring lingkungan ditempat kerja masing – masing.

5. *Assisten Foreman*

Membantu/mewakili *foreman* pengawasan, koordinasi dan pengendalian proses kendaraan CJM di *trimming* 3 sehingga dapat di *delivery* sesuai dengan standard, terpeliharanya lingkungan kerja serta keselamatan dan kesehatan kerja.

Tugas dan tanggung jawab:

- a. Membantu/mewakilkkan merealisasikan *schedule* produksi yang dibuat PPC sesuai standard mutu dan jumlah.
- b. Membantu/mewakilkkan mengevaluasi, mengarahkan dan membina karyawan sehingga menghasilkan karyawan yang bertanggung jawab, disiplin dan berprestasi guna tercapainya sasaran mutu.
- c. Membantu/mewakilkkan melakukan pengukuran *cycle time* sehingga tercapai *line balance* / pembagian kerja yang merata.
- d. Membantu/mewakilkkan memelihara lingkungan kerja, alat-alat dan mesin-mesin dengan cara melaksanakan program K3, 5R dan GKM.
- e. Membantu/mewakilkkan pengendalian atau mencegah kecelakaan kerja dengan cara : Eliminasi, Substitusi, Rekayasa Teknik, Administrasi dan APD.
- f. Membantu/mewakilkkan melakukan absensi bawahan.
- g. Membantu/mewakilkkan menjalankan, memelihara, meningkatkan sistem manajemen perusahaan.
- h. Membantu/mewakilkkan melakukan identifikasi, mengolah dan memonitoring lingkungan ditempat kerja masing - masing.

4.6. Kebijakan Perusahaan

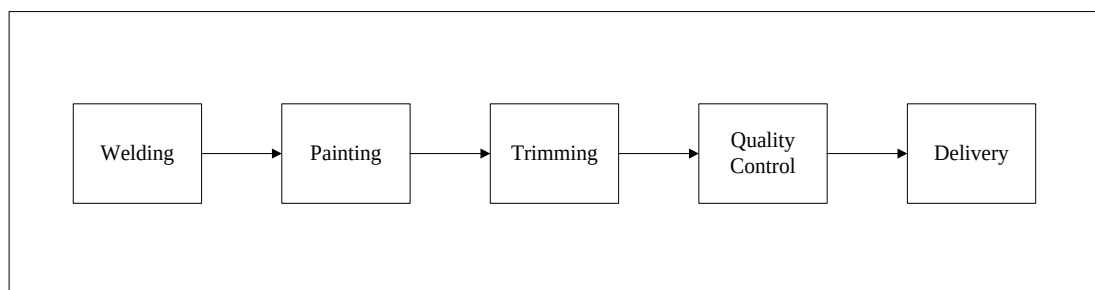
PT Krama Yudha Ratu Motor menerapkan Sistem Manajemen Keselamatan Dan Kesehatan Kerja (SMK3) dalam kegiatan selama bekerja sebagai berikut :

1. Mengelola Perusahaan yang aman dan semakin berkembang dengan menerapkan prinsip **QCDSM** (*Quality, Cost, Delivery, Safety, Morale*) dengan melakukan peningkatan kualitas, penghematan biaya, pengiriman tepat waktu, mengusahakan nihil kecelakaan, peningkatan sumber daya manusia, mencegah gangguan kesehatan dan penyakit akibat kerja serta mencegah kebakaran di pabrik.
2. Senantiasa melakukan pengelolaan lingkungan dengan mencegah pencemaran atau dampak lingkungan dan penghematan sumber daya alam dan atau energi.
3. Menyediakan induksi dan pembelajaran kepada karyawan dan mitra kerja.

4. Mematuhi peraturan dan persyaratan lain yang terkait dengan keselamatan dan kesehatan kerja serta lingkungan hidup.
5. Senantiasa melakukan peningkatan serta perbaikan berkesinambungan.
6. Memastikan setiap karyawan dan atau mitra kerja PT KRM memahami, bertanggung jawab dan mempertanggung jawabkan untuk peduli pada lingkungan dan K3 dalam aktifitasnya.

4.7. Alur Produksi Perusahaan Secara Umum

Alur proses produksi perakitan kendaraan pada PT KRM terdiri dari beberapa bagian, dimana setiap bagian memiliki keterhubungan antar satu dengan yang lainnya.



Gambar IV.4 Proses Produksi Secara Umum
(Sumber: PT Krama Yudha Ratu Motor, 2017)

a. *Welding*

PT Krama Yudha Ratu Motor memproduksi sendiri *cabin* setiap produk mobil yang dihasilkan dengan melakukan pengelasan setiap komponen/*part* pembentuk *cabin*. Dalam proses *welding* terdapat beberapa *station* yang kemudian akan di *metal finish* dan dicek kualitas *cabin* untuk dikirim ke proses selanjutnya yaitu proses *Painting*.

b. *Painting*

Tahapan selanjutnya setelah proses *welding* adalah proses *painting*. Pada proses *painting* dilakukan tahapan pengecatan terhadap *cabin* dan *rear body*.

c. *Trimming*

Pada tahap ini dilakukan perakitan semua *part* pembentuk mobil sesuai dengan model yang sudah ada. *Cabin* dan *part* dirakit membentuk mobil dan dilakukan *testing* untuk mengecek kelayakan mobil. Mobil yang lulus uji coba kemudian

langsung disimpan ke gudang KTB dan akan di distribusikan ke *customer*. Pada bagian trimming terbagi beberapa bagian, yaitu:

1. *Trimming 1*

Pada bagian *trimming 1* dilakukan perakitan jenis TD, *cabin* SL dan *rear body* SL.

2. *Trimming 2*

Pada bagian *trimming 2* dilakukan perakitan jenis TD/Fuso.

3. *Trimming 3*

Pada bagian *trimming 3* dilakukan perakitan jenis CJM.

4. *Trimming 4*

Pada bagian ini dilakukan perakitan jenis ZC.

d. *Quality Control*

Pada tahapan ini dilakukan tes pada kendaraan, apakah ada cacat atau tidak.

e. *Delivery* ke Pihak PT Krama Yudha Tiga Berlian

Setelah semua proses selesai dilaksanakan, maka selanjutnya kendaraan dikirim ke PT Krama Yudha Tiga Berlian untuk dijual kepada customer diseluruh Indonesia.

4.8. Komponen (*Part*)

Komponen atau *part* merupakan suatu bahan yang digunakan untuk merakit mobil tipe Colt T120 SS. Berikut terdapat daftar dari beberapa *part* yang digunakan.

Tabel IV.1 Daftar *Part*

No.	Nama <i>Part</i>	No. <i>Part</i>
1.	BOLT REAR BODY MOUNTING	09112 - 10011
2.	CLAMP ROOM LAMP WIRE	09407 - 05104
3.	Belt. Assy Front	84900-77501L
4.	BOX COMP HEAD LAMP HOUSING	73650 - 85200
5.	BUCKLE ASSY CENTER	84901 - 77501L

6.	CUSHION AIR CLEANER	MB 169307
7.	RESERVE TANK ASSY	17930 - 77500
8.	CABLE CAMB PARKING BRAKE	54410 - 77500
9.	ANTENNA SET	KM 003334
10.	HARNESS CABIN	8510B350 M
11.	CLIP BAND HARNESS	09407 - 14408
12.	BRAKE PIPE	51380-77500
13.	BOLT PARKING CABLE	02162 - 0610A
14.	MOTOR LINK	38100 - 77500
15.	HAND BRAKE	54100 - 7750
16.	GROMET PIPA VACUM	51751 - 85000
17.	BOLT PARKING BRK	01550-08203
18.	CLAMP.SPEEDOMETER CABLE	09408-00069
19.	PEDAL ASSY.ACCEL	49400-77500
20.	COVER.ACCEL PEDAL	49451-70D00
21.	STOPPER BOLT	09521-09002

Tabel IV.1 Daftar *Part* (lanjutan)

No.	Nama <i>Part</i>	No. <i>Part</i>
22.	MASTER VACUM ASSY	55000-77500
23.	SUNVISOR ASSY	84802-77500
24.	HOLDER SUNVISOR	84851-60800
25.	BOLT FUSE BOX	02162 - 0512A
26.	BOLT WASHER ASSY	MS 240143
27.	CENTER PANEL ASSY	MB 680465
28.	RADIO CD/MP3 ASSY	KM 003339

29.	BOX SPEAKER ASSY RH	KM 003289
30.	GRIP PILLAR	76411 – 85410
31.	RESERVOIR ASSY	51200 – 77501
32.	SCREW BRACKET RADIATOR	09122 – 76001
33.	RADIATOR ASSY	17700 – 77500
34.	AIR CLEANER	MD 161188Y
35.	NUT FRONT DOOR INSIDE HANDLE	09148-06007
36.	Regulator	82960-82000
37.	Mirror assy door Rh	MB 671955
38.	View mirror	02142-06123
39.	CLAMP PARKING CABLE	09308-11008
40.	WASHER,R/SHK ABS UPR	08321-0110
41.	NUT,R/SHK ABS UPR	08310-00103
42.	CLAMP PACKING BRAKE CABLE NO 1	71510-77500
43.	PIPE COMB BRACKET VACUM MASTER	14281 - 79010
44.	CLAMP SPEEDOMETER CABLE	45341-77500
45.	PIPE VALVE TO REAR BRK HO	450309-77500
46.	PIPE COMP FUEL	08314-40123
47.	WASHER, TENSION ROD	45313-77500
48.	BUSHING	45341-77500
49.	ROD SET	450309-77500
50.	PIN, BALL JOINT STUD SPLIT	04111-30208
51.	NUT, BALL JOINT STUD CASTLE	08314-40123
52.	CLAMP CLUTCH CABLE	48500-77500

Tabel IV.1 Daftar *Part* (lanjutan)

No.	Nama <i>Part</i>	No. <i>Part</i>
53	Fender deck Rr Lh TB116	41786-52011
54.	LAMP BACK UP	35910-7750
55.	TIRE ASSY	5.50-13-8PR

(Sumber: PT Krama Yudha Ratu Motor, 2017)

4.9. Proses Produksi *Trimming 3*

Proses *trimming* berfungsi untuk memasang dan merakit komponen-komponen kendaraan (produk). Pada proses *Trimming 3* terdapat 15 stasiun untuk menjadi suatu produk atau barang jadi, dimana setiap stasiun memiliki jumlah mekanik yang berbeda-beda tergantung tingkat kerumitan yang ada pada setiap stasiun. Pada *trimming 3* memiliki 3 *foreman* dimana 1 *foreman* sebagai ketuanya dan 2 nya lagi sebagai asisten dari *foreman* tersebut. Setiap stasiun memiliki SOP yang berbeda beda pula tentunya, di *layout trimming 3* ini bahwa ada 3 proses yang akan dilalui untuk sampai pada tahap *quality control* yaitu:

1. *Trimming On*

Dimana pada proses ini cabin baru turun dari proses *welding* dan *painting* dan pada tipe T120 SS atau CJM (*Car Join Mitsubishi*) ini kap dan bak mobil sudah menyatu. Terdapat 5 *man power* dalam melakukan proses ini. Berikut salah satu proses di *Trimming On*.



Gambar IV.4 Proses di *station 1*
(Sumber: PT Krama Yudha Ratu Motor, 2017)

2. *Hi Line*

Pada proses ini terdapat 3 *man power* yang akan melakukan proses pemasangan *engine*, transmisi dan roda-roda pada mobil itu di pasang dan yang lebih penting tingkat keamanan pun lebih ditingkatkan pada fase ini seperti penambahan APD atau *special tools* untuk menunjang pemasangan pada part yang akan dipasang. Berikut salah satu proses yang ada di Hi Line.



(Sumber: PT Krama Yudha Ratu Motor, 2017)

3. *Trimming Off*

Proses pemasangan yang berhubungan dengan *part electrical* seperti pemasangan aki mobil dan lampu-lampu pada mobil yang dilakukan oleh 7 *man power*. Berikut proses yang ada di *Trimming Off*.



Gambar IV.6 Proses di *station 15*
(Sumber: PT Krama Yudha Ratu Motor, 2017)

4.10. Hasil Produksi *Trimming 3*

Mitsubishi Colt T120 SS atau CJM merupakan salah satu produk yang dihasilkan oleh PT Krama Yudha Ratu Motor. Dipercaya sejak tahun 1972, Mitsubishi Colt T120 SS terus melakukan perubahan guna memenuhi kebutuhan kendaraan komersial di Indonesia. Kendaraan tipe ini di produksi pada bagian *trimming 3*, dimana pada proses produksi di bagian *trimming* ini dilakukan perakitan terhadap mobil tersebut. Berikut tampilan untuk mobil tipe CJM:



Gambar IV.7 Mobil
trimming 3
(Sumber: PT Krama

tipe CJM pada
Yudha Ratu Motor, 2017)

4.11. Prosedur Sistem yang Berjalan di Bagian *Trimming* 3

Bagian *Trimming* 3 merupakan bagian yang memiliki tugas untuk melakukan produksi dalam merakit komponen atau *parts* khusus kendaraan tipe Colt T120 SS atau lebih sering disebut dengan CJM. Dalam melakukan kegiatan produksi bagian *Trimming* 3 sudah melakukan pencatatan informasi dengan menggunakan bantuan aplikasi berbasis *office* atau *Microsoft Excel* dimana belum adanya media penyimpanan *database*

4.12. Prosedur Dokumen yang Digunakan

Produksi ialah suatu kegiatan mengenai pembuatan produk baik berwujud fisik maupun berwujud jasa untuk menambah nilai guna. Pada bagian *Trimming* 3 dalam memproduksi suatu produk yaitu dengan melakukan perakitan komponen atau *part* pada kendaraan tipe Colt T120 SS yang lebih sering disebut dengan CJM. Dalam melakukan produksi tersebut memerlukan dokumen-dokumen yang terkait dengan tujuan untuk mengatur dalam kegiatan proses produksi agar dapat berjalan dengan lancar. Adapun dokumen yang terlibat dalam sistem informasi pelaporan kerusakan part diantaranya adalah:

1. Rencana Produksi Harian

Rencana produksi pada PT Krama Yudha Ratu Motor berisikan jadwal dan jumlah suatu produk harus selesai di produksi sesuai jadwal yang

[illegible]

Gambar IV.10 Laporan Produksi Harian
(Sumber : PT Krama Yudha Ratu Motor, 2017)

Keterangan:

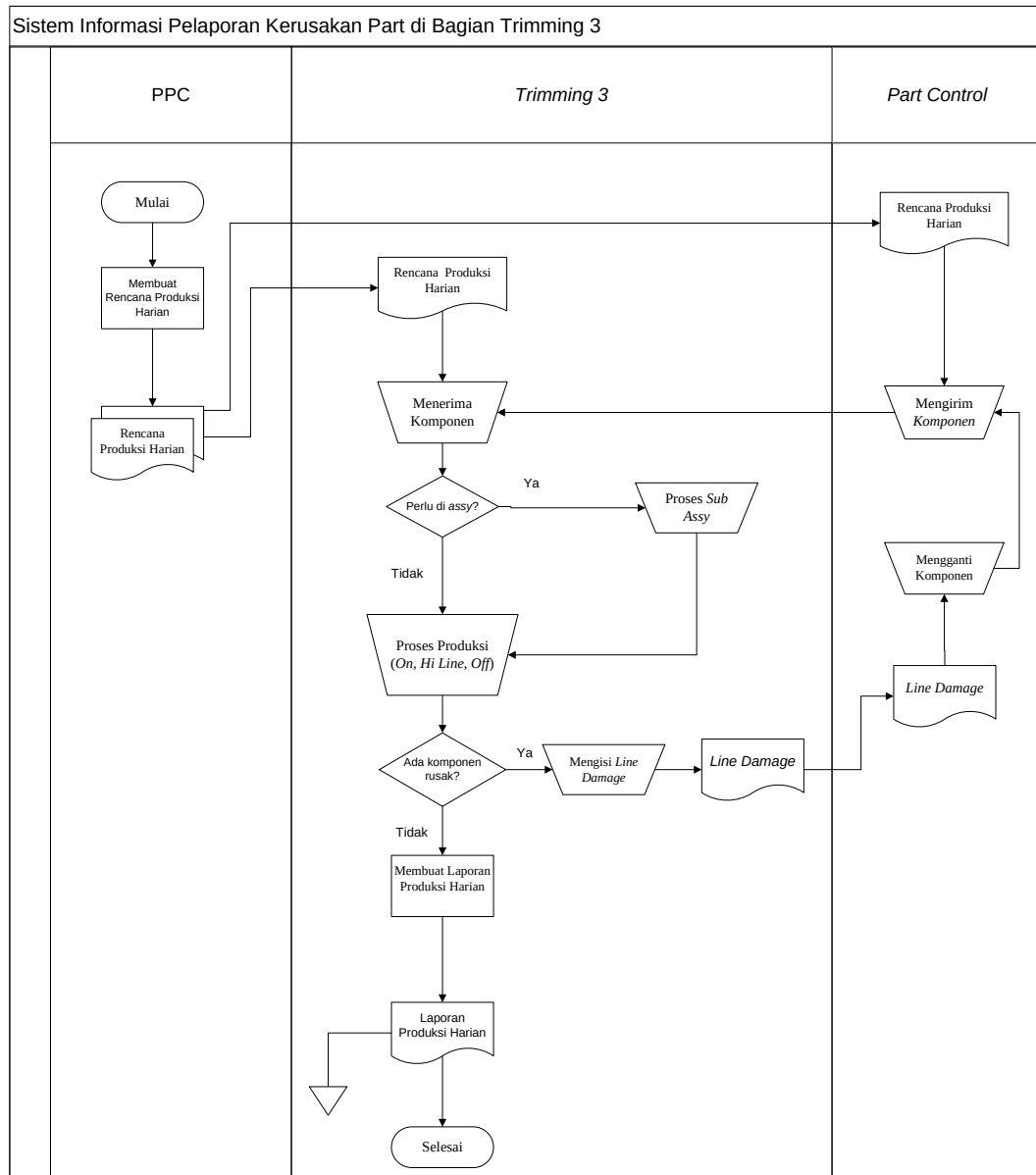
- a. Type : tipe-tipe mobil Colt T120 SS (CJM)
- b. Rencana : Rencana produksi yang telah direncanakan
- c. Hasil : Hasil dari produksi
- d. Total : total dari rencana dan hasil yang diproduksi

4.13. Prosedur Sistem Informasi yang Berjalan

Prosedur sistem informasi pelaporan kerusakan parti pada PT Krama Yudha Ratu Motor terdiri dari beberapa tahapan yaitu:

1. Bagian *Production Planning and Control* (PPC) memberikan rencana produksi harian yang telah dibuat kepada seluruh bagian produksi termasuk *Trimming 3* dan juga *Part Control*.
2. Bagian *Part Control* menerima rencana produksi harian tersebut kemudian komponen akan disusun dan siap dikirim ke bagian *Trimming 3*.
3. Bagian *Trimming 3* menerima rencana produksi harian kemudian menerima komponen yang dikirimkan oleh bagian *Part Control*.
4. Apakah komponen tersebut perlu di Assy? Jika Iya maka komponen tersebut akan di bawa ke *Sub Assy* baru kemudian akan dikirim ke *line* produksi yang dibutuhkan dalam proses produksi (*Trimming On*, *Trimming Hi Line*, dan *Trimming Off*) untuk perakitan.
5. Jika komponen tersebut tidak perlu di Assy maka akan langsung dikirimkan ke *line* proses produksi (*Trimming On*, *Trimming Hi Line*, dan *Trimming Off*).
6. Pada saat produksi (*Trimming On*, *Trimming Hi Line*, dan *Trimming Off*) , apakah terjadi masalah kerusakan komponen? jika terjadi kerusakan pada komponen yang dilakukan oleh operator saat pemasangan/instalasi komponen saat dirakit maka mengisi *line damage* dan akan dikirimkan ke bagian *Part Control*
7. Setelah menerima *line damage* bagian *Part Control* akan mengganti *part* tersebut dan mengirimkan kembali kebagian *Trimming 3*.
8. Jika proses produksi lancar tidak ada komponen yang rusak maka akan membuat laporan produksi harian kemudian dicetak dan diarsipkan.

Berikut ini adalah penggambaran prosedur sistem informasi pelaporan kerusakan part dengan menggunakan *flowmap* dapat dilihat pada Gambar IV.11.



Gambar IV.11 Flowmap sistem informasi pelaporan kerusakan part yang berjalan

(Sumber : Pengolahan data, 2018)

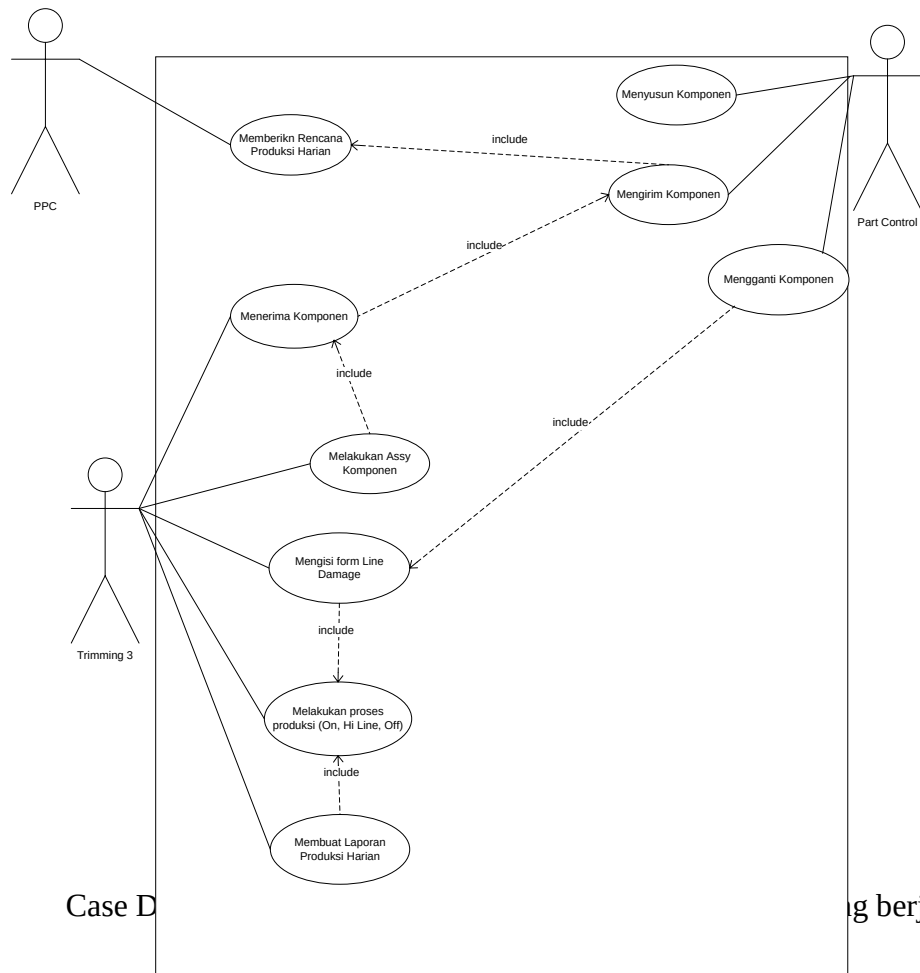
4.13 Pemodelan Sistem yang Berjalan dengan *Use Case*

Diagram *use case* menggambarkan kebutuhan sistem dari sudut pandang *user*, yaitu proses yang dilakukan oleh sistem dalam melayani *user* yang berinteraksi dengan sistem tersebut. Dalam sistem informasi pelaporan kerusakan part ini, terdapat tiga aktor. Antara lain adalah Bagian *Production Planning and*

Control (PPC), Bagian *Trimming* 3, dan Bagian *Part Control* . Gambar IV.12

No	Aktor	Deskripsi
----	-------	-----------

menjelaskan tentang penggambaran proses produksi pada sistem informasi pelaporan kerusakan part yang berjalan dalam *use case diagram*.



Ga
mb
ar
IV.
12
Use

g berjalan

Penjelasan *use case diagram* sistem informasi pelaporan kerusakan part dapat dijelaskan sebagai berikut:

1. Definisi Aktor

Pendefinisian aktor pada *use case diagram* Sistem Informasi Pelaporan kerusakan part dapat dilihat pada Tabel IV.1 dibawah ini

Tabel IV.2 Definisi Aktor

1	Bagian PPC	Membuat dan memberikan dokumen
No.	Use Case	Deskripsi
2	Bagian <i>Trimming 3</i>	Melakukan proses produksi, membuat <i>line damage</i> membuat laporan harian produksi.
3	Bagian <i>Part Control</i>	Mengirim komponen ke bagian produksi termasuk <i>Trimming 3</i> , mengganti komponen yang rusak akibat kesalahan operator saat instal,

(Sumber: Pengolahan data, 2018)

2. Definisi *Use Case Diagram*

Pendefinisian *use case* pada *use case diagram* sistem informasi pelaporan kerusakan part di bagian *Trimming 3* pada PT Krama Yudha Ratu Motor dapat dilihat pada Tabel IV.2 berikut:

Tabel IV.3 Definisi *Use Case* Sistem Informasi Pelaporan kerusakan *part*

No.	Use Case	Deskripsi
1.	Memberi Rencana Produksi Harian	Rencana produksi harian dibuat dan dibagikan oleh bagian PPC
2.	Mengirim Komponen	Bagian <i>Part Control</i> mengirimkan komponen atau <i>part</i> ke bagian <i>Trimming 3</i> untuk digunakan dalam proses produksi perakitan.

Tabel IV.3 Definisi *Use Case* Sistem Informasi Pelaporan kerusakan *part*
(lanjutan)

4.		Bagian <i>Trimming</i> 3 menerima komponen atau <i>part</i> yang telah dikirim untuk keperluan produksi.
5.	Meng <i>assy</i> Komponen	Bagian <i>Trimming</i> 3 melakukan <i>assy</i> terhadap komponen tertentu sebelum di bawa ke bagian perakitan.
6.	Melakukan Proses Produksi	Proses untuk menghasilkan suatu produk yang dilakukan oleh bagian <i>Trimming</i> 3 meliputi <i>Trimming On</i> , <i>Trimming Hi Line</i> , <i>Trimming Off</i> .
7.	Membuat Line Damage	Proses yang akan dilakukan jika bagian <i>Trimming</i> 3 mengalami kerusakan komponen saat proses instal oleh operator .
8.	Mengganti <i>Part</i>	Bagian <i>Part Control</i> akan mengganti <i>part</i> yang rusak atau cacat dengan mengirimkan lagi yang baru ke bagian <i>Trimming</i> 3.
9.	Membuat Laporan Produksi Harian	Bagian <i>Trimming</i> 3 membuat laporan produksi harian kemudian akan diarsip.

(Sumber: Pengolahan data, 2018)

BAB V

ANALISIS DAN PEMBAHASAN

5.1. Analisis Kebutuhan *Functional* Sistem

Berdasarkan hasil pengolahan data dan analisis sistem, maka dibutuhkan perancangan aplikasi sistem informasi pelaporan kerusakan *part* pada Bagian *Trimming* 3. Berikut adalah daftar kebutuhan *Functional* sistem untuk aplikasi sistem informasi pelaporan kerusakan *part* pada Bagian *Trimming* 3.

Tabel V.1 Kebutuhan *Functional* Sistem

Masalah	Kebutuhan User	Kebutuhan <i>Functional</i> Sistem	Uraian
Belum adanya media pengolahan data yang terhubung ke basisdata.	Pengolahan data yang mudah dan cepat.	Pengelolaan Data Master.	Data master diantaranya adalah data <i>user</i> , data <i>part</i> dan data produk. Proses yang dapat dilakukan pada pengelolaan data tersebut adalah: - Dapat menambah, mengubah, mencari dan menghapus data tersebut.
		Pengelolaan Data Transaksi.	Data Transaksi diantaranya adalah data rencana produksi, <i>Line damage</i> , dan hasil produksi - Dapat menambah, mengubah, mencari dan menghapus data tersebut.

Tabel V.1 Kebutuhan *Functional* Sistem (lanjutan)

Masalah	Kebutuhan User	Kebutuhan <i>Functional</i> Sistem	Uraian
Pembuatan line damage yang masih manual dengan mengisi lembaran line damage secara manual.	Membuat <i>line damage</i> di sistem yang sudah terkomputerisasi.	Adanya menu <i>line damage</i> yang sudah terhubung dengan database.	Proses yang dapat dilakukan pada pengelolaan data tersebut adalah: - Dapat menambah, mengubah, mencari dan menghapus data tersebut.
Setiap bagian belum terintegrasi.	Pengiriman <i>Line Damage</i> tanpa harus datang langsung ke bagian lain.	Pengiriman <i>Line Damage</i> .	Proses yang dapat dilakukan pada pengiriman <i>line damage</i> yaitu: - Dapat melihat data <i>line damage</i> .
Dalam mencetak laporan kerusakan part masih belum terintegrasi dengan data yang akan dicetak	Mencetak laporan secara mudah	Mencetak Laporan kerusakan <i>part</i> .	Proses yang dapat dilakukan pada mencetak laporan kerusakan <i>part</i> .

(Sumber: Hasil Analisis, 2018)

5.2. Prosedur Sistem Informasi Pelaporan Kerusakan *Part*

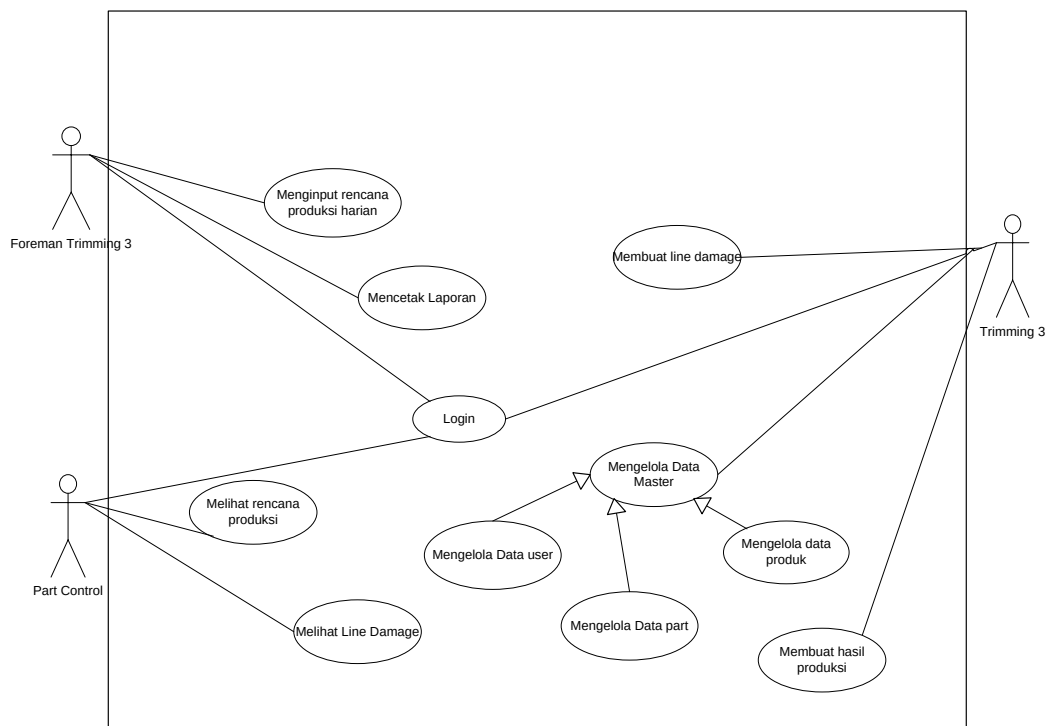
Prosedur sistem informasi pelaporan kerusakan *part* di Bagian *Trimming 3* yang diusulkan melalui tahapan sebagai berikut:

1. *Foreman* di *Trimming 3* menerima rencana produksi harian lalu menginputkan rencana produksi harian tersebut kedalam sistem dan akan disimpan ke dalam database produksi
2. Bagian *Part Control* melihat data rencana produksi harian yang telah diinputkan oleh *foreman Trimming 3*
3. Bagian *part control* akan mengirim komponen/*part* yang sesuai dengan jumlah yang ada di rencana produksi yang nantinya akan dikirim ke bagian *Trimming 3*.
4. *Trimming 3* menerima komponen dari *part control* setelah itu apakah komponen tersebut perlu di Assy? Jika Iya maka komponen tersebut akan di bawa ke *Sub Assy* baru kemudian akan dikirim ke *line* produksi yang dibutuhkan dalam proses produksi (*Trimming On*, *Trimming Hi Line*, dan *Trimming Off*) untuk perakitan.
5. Jika komponen tersebut tidak perlu di Assy maka akan langsung dikirimkan ke *line* proses produksi (*Trimming On*, *Trimming Hi Line*, dan *Trimming Off*).
6. Pada saat produksi (*Trimming On*, *Trimming Hi Line*, dan *Trimming Off*) , apakah terjadi masalah kerusakan komponen? jika terjadi kerusakan pada komponen yang dilakukan oleh operator saat pemasangan/instalasi komponen saat dirakit maka membuat *line damage* dan akan disimpan kedalam *database*.
7. Bagian *part control* melihat data *line damage* kemudian akan mengganti komponen yang telah rusak dan dikirim ke bagian *Trimming 3*.
8. Dan jika pada saat proses produksi lancar maka *Trimming 3* akan membuat data hasil produksi yang akan masuk ke dalam *database* produksi.
9. *Foreman Trimming 3* akan mencetak laporan hasil produksi dan kerusakan *part* yang ada.

pembuatan kamus data. Tahapan analisis ini akan memberikan gambaran mengenai aliran informasi dan data pada sistem informasi yang akan dibangun.

5.3.1. Use Case Diagram

Use Case Diagram mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Rancangan *use case diagram* sistem informasi pelaporan kerusakan *part* yang diusulkan dapat dilihat pada Gambar V.2.



Gambar V.2 *Use Case Diagram* Usulan Sistem Informasi Pelaporan Kerusakan *Part* di Bagian *Trimming 3*
(Sumber: Hasil Analisis, 2018)

Perancangan *use case diagram* perancangan sistem informasi pelaporan kerusakan *part* dapat dijelaskan sebagai berikut:

1. Definisi Aktor

Pendefinisian aktor pada *use case diagram* usulan perancangan sistem informasi pelaporan kerusakan *part* di Bagian *Trimming 3* dapat dilihat pada Tabel V.2 berikut:

Tabel V.2 Definisi Aktor *Use Case Diagram* Sistem Usulan

No.	Aktor	Deskripsi
-----	-------	-----------

1.	Bagian <i>Foreman Trimming 3</i>	Bagian <i>Foreman Trimming 3</i> adalah aktor yang menginput data rencana produksi harian dan mencetak laporan.
2.	Bagian <i>Trimming 3</i>	Bagian <i>Trimming 3</i> adalah aktor mengelola data master, menginput <i>line damage</i> dan membuat hasil produksi.
3.	<i>Part Control</i>	Bagian <i>part control</i> adalah bagian yang akan menerima <i>line damage</i> dari bagian <i>Trimming 3</i> dan melihat rencana produksi

(Sumber: Hasil Analisis, 2018)

2. Definisi Use Case

Pendefinisian *use case* pada sistem informasi pelaporan kerusakan *part* di bagian *Trimming 3* usulan dapat dilihat pada Tabel V.3:

Tabel V.3 Definisi *Use Case Diagram* Sistem Usulan

No.	Use Case	Deskripsi
1.	<i>Login</i>	Proses untuk melakukan <i>login</i> pada aplikasi.
2.	Mengelola data master	Proses generalisasi yang meliputi tiga buah proses pengolahan data master, yaitu data <i>user</i> , <i>parts</i> , operator.
3.	Mengelola data <i>user</i>	Proses mengelola data <i>user</i> yang terdiri dari tambah, ubah, dan hapus pada data pengguna.
4.	Mengelola data <i>part</i>	Proses mengelola data <i>part</i> yang terdiri dari tambah, ubah, dan hapus pada data komponen.
5.	Mengelola data produk	Proses mengelola data produk yang terdiri dari tambah, ubah, dan hapus pada data produk.
6.	Menginput data rencana produksi	Proses menginput data rencana produksi yang terdiri dari tambah, ubah, dan hapus pada data rencana produksi.

Tabel V.3 Definisi *Use Case Diagram* Sistem Usulan (lanjutan)

No.	Use Case	Deskripsi
-----	----------	-----------

7.	Melihat data rencana produksi	proses melihat data rencana produksi sebelum melakukan produksi.
8.	Mengelola data <i>line damage</i>	Proses mengelola data <i>line damage</i> yang terdiri dari tambah, ubah, dan hapus pada data rencana line damage
9.	Melihat data <i>line damage</i>	proses melihat data rencana produksi sebelum melakukan mengganti <i>parts</i> /komponen yang rusak
10.	Membuat hasil produksi	Proses mengelola data hasil produksi yang terdiri dari tambah, ubah, dan hapus pada data produk.
11.	Mencetak Laporan	Proses mencetak laporan

(Sumber: Hasil Analisis, 2018)

3. Skenario *Use Case*

Skenario jalannya masing-masing *use case* usulan pada sistem informasi pelaporan kerusakan *part* dapat dilihat pada poin berikut:

1. *Use Case Login*

Berikut adalah skenario *use case login* yang terdapat pada tabel V.4.

Tabel V.4 Definisi *Use Case Login*

Nama <i>Use Case</i>	<i>Login</i>
<i>Primary Actor</i>	Bagian <i>Foreman Trimming 3</i> , Bagian <i>Trimming 3</i> , Bagian <i>Part control</i>
<i>Use Case Description</i>	<i>Use case</i> ini menggambarkan <i>login</i> sesuai dengan hak akses dari user. Dalam hal ini hak aksesnya sesuai Divisi atau bagian.
<i>Relationship</i>	<i>Association</i> : Bagian <i>Foreman Trimming 3</i> , Bagian <i>Trimming 3</i> , Bagian <i>Part control</i>

Tabel V.4 Definisi *Use Case Diagram Login* (lanjutan)

Nama <i>Use Case</i>	<i>Login</i>
----------------------	--------------

<i>Normal Flow of Events</i>	1. <i>User</i> membuka aplikasi. 2. Sistem menampilkan <i>form Login</i>. 3. <i>User</i> mengisi <i>username</i> dan <i>password</i> pada <i>form login</i>. 4. Sistem melakukan validasi <i>login</i>. 5. Jika <i>username</i> dan <i>password</i> valid, maka muncul halaman utama. 6. Jika <i>username</i> dan <i>password</i> salah (tidak valid), maka menampilkan <i>form login</i> kembali.
------------------------------	---

(Sumber: Hasil Analisis, 2018)

2. Use Case Mengelola Data Master User

Berikut adalah skenario *use case* mengelola data master *user* yang terdapat pada tabel V.5.

Tabel V.5 Definisi Use Case Mengelola Data Master *user*

Nama Use Case	Mengelola Data Master <i>user</i>
<i>Primary Actor</i>	Bagian <i>Trimming 3</i>
<i>Use Case Description</i>	<i>Use Case</i> ini adalah proses untuk melakukan pengolahan data master diantaranya adalah data pengguna
<i>Relationship</i>	<i>Association: Trimming 3</i>
<i>Normal Flow of Events</i>	1. Bagian <i>Trimming 3</i> masuk ke menu tampilan utama. 2. Memilih menu data master. 3. <i>User</i> memilih submenu data <i>user</i>. 4. Sistem menampilkan data <i>user</i>. 5. <i>User</i> dapat melakukan proses tambah, ubah, cari dan hapus data pengguna ke <i>database</i>.

Tabel V.5 Definisi Use Case Mengelola Data Master *user* (lanjutan)

Nama Use Case	Mengelola Data Master user
Normal Flow of Events	1. Bagian <i>Trimming 3</i> masuk ke menu tampilan utama. 2. Memilih menu data master. 3. User memilih submenu data user. 4. Sistem menampilkan data user. 5. User dapat melakukan proses tambah, ubah, cari dan hapus data pengguna ke <i>database</i>. 6. Jika <i>Trimming 3</i> memilih aksi tambah data, sistem akan menampilkan <i>form</i> tambah data, kemudian user memasukkan data user lalu mengklik simpan dan data masuk ke dalam <i>database</i> lalu sistem menampilkan data user. 7. Jika <i>Trimming 3</i> memilih aksi ubah data, sistem akan menampilkan form ubah data, user memasukkan data yang ingin di ubah, lalu mengklik simpan kemudian data masuk ke dalam <i>database</i> lalu sistem menampilkan data user. 8. Jika <i>Trimming 3</i> memilih aksi hapus data, user memilih data user yang ingin dihapus lalu sistem akan menghapus data. 9. Jika <i>Trimming 3</i> memilih aksi cari data, user memasukkan kode atau nama produk lalu sistem akan menampilkan data yang dicari.

(Sumber: Hasil Analisis, 2018)

3. Use Case Mengelola Data Master Part

Berikut adalah skenario *use case* mengelola data master *part* yang terdapat pada tabel V.6.

Tabel V.6 Definisi Use Case Mengelola Data Master Part

Nama Use Case	Mengelola Data Master Part
Primary Actor	Bagian <i>Trimming 3</i>
Use Case Description	<i>Use Case</i> ini adalah proses untuk melakukan pengolahan data master diantaranya adalah data part.
Relationship	Association: <i>Trimming 3</i>
Normal Flow of Events	1. Bagian <i>Trimming 3</i> masuk ke menu tampilan utama. 2. Memilih menu data master. 3. User memilih submenu data <i>part</i> 4. Sistem menampilkan data <i>part</i>. 5. User dapat melakukan proses tambah, ubah, cari dan hapus data pengguna ke <i>database</i>. 6. Jika <i>Trimming 3</i> memilih aksi tambah data, sistem akan menampilkan <i>form</i> tambah data, kemudian <i>user</i> memasukkan data <i>part</i> lalu mengklik simpan dan data masuk ke dalam <i>database</i> lalu sistem menampilkan data <i>part</i>. 7. Jika <i>Trimming 3</i> memilih aksi ubah data, sistem akan menampilkan <i>form</i> ubah data, <i>user</i> memasukkan data yang ingin di ubah, lalu mengklik simpan kemudian data masuk ke dalam <i>database</i> lalu sistem menampilkan data <i>part</i>. 8. Jika <i>Trimming 3</i> memilih aksi hapus data, <i>user</i> memilih data <i>part</i> yang ingin dihapus lalu sistem akan menghapus data. 9. Jika <i>Trimming 3</i> memilih aksi cari data, <i>user</i> memasukkan kode atau nama produk lalu sistem akan menampilkan data yang dicari.

(Sumber: Hasil Analisis, 2018)

4. Use Case Mengelola Data Master Produk

Berikut adalah skenario *use case* mengelola data master produk yang terdapat pada tabel V.7.

Tabel V.7 Definisi *Use Case* Mengelola Data Master Produk

Nama Use Case	Mengelola Data Master Produk
Primary Actor	Bagian <i>Trimming</i> 3
Use Case Description	<i>Use Case</i> ini adalah proses untuk melakukan pengolahan data master diantaranya adalah data produk.
Relationship	Association: <i>Trimming</i> 3
Normal Flow of Events	1. Bagian <i>Trimming</i> 3 masuk ke menu tampilan utama. 2. Memilih menu data master. 3. User memilih submenu data produk 4. Sistem menampilkan data produk. 5. User dapat melakukan proses tambah, ubah, cari dan hapus data pengguna ke <i>database</i>. 6. Jika <i>Trimming</i> 3 memilih aksi tambah data, sistem akan menampilkan <i>form</i> tambah data, kemudian <i>user</i> memasukkan data <i>part</i> lalu mengklik simpan dan data masuk ke dalam <i>database</i> lalu sistem menampilkan data <i>part</i>. 7. Jika <i>Trimming</i> 3 memilih aksi ubah data, sistem akan menampilkan <i>form</i> ubah data, <i>user</i> memasukkan data yang ingin di ubah, lalu mengklik simpan kemudian data masuk ke dalam <i>database</i> lalu sistem menampilkan data <i>part</i>. 8. Jika <i>Trimming</i> 3 memilih aksi hapus data, <i>user</i> memilih data <i>part</i> yang ingin dihapus lalu sistem akan menghapus data. 9. Jika <i>Trimming</i> 3 memilih aksi cari data, <i>user</i> memasukkan kode atau nama produk lalu sistem akan menampilkan data yang dicari.

(Sumber: Hasil Analisis, 2018)

5. Use Case Menginput Rencana Produksi

Berikut adalah skenario *use case* menginput rencana produksi yang terdapat pada tabel V.8.

Tabel V.8 Definisi *Use Case* Menginput Rencana Produksi

Nama Use Case	Menginput Rencana Produksi
<i>Primary Actor</i>	<i>Foreman Trimming 3</i>
<i>Use Case Description</i>	<i>Use case ini menginputkan rencana harian produksi yang kemudian diberikan kepada Bagian Trimming 3 untuk melakukan produksi.</i>
<i>Relationship</i>	<i>Association: Bagian Trimming 3 dan Part Control</i>
<i>Normal Flow of Events</i>	1. <i>Foreman Trimming 3</i> masuk ke menu tampilan utama 2. <i>Foreman Trimming 3</i> memilih menu transaksi 3. User memilih sub menu rencana produksi lalu memilih tambah data 4. Sistem akan menampilkan <i>form</i> rencana produksi 5. User akan melakukan dan mengisi <i>form</i> rencana produksi dan disimpan ke <i>database</i> Sistem akan menampilkan data rencana produksi yang dibuat

(Sumber: Hasil Analisis, 2018)

6. *Use Case* Melihat Rencana Produksi

Berikut adalah skenario *use case* melihat rencana produksi yang terdapat pada tabel V.9.

Tabel V.9 Definisi *Use Case* Melihat Rencana Produksi

Nama Use Case	Melihat Rencana Produksi
<i>Primary Actor</i>	Bagian <i>Part Control</i> (PC)
<i>Use Case Description</i>	<i>Use case ini menampilkan rencana produksi</i>
<i>Relationship</i>	<i>Association: Bagian Foreman Trimming 3</i>

Tabel V.9 Definisi *Use Case* Melihat Rencana Produksi (lanjutan)

Nama <i>Use Case</i>	Melihat Rencana Produksi
<i>Normal Flow of Events</i>	1. Bagian PC masuk ke menu tampilan utama 2. Bagian PC memilih menu rencana produksi 3. <i>User</i> memilih submenu rencana produksi 4. Sistem akan menampilkan data rencana produksi

(Sumber: Hasil Analisis, 2018)

7. *Use Case* Membuat *Line Damage*

Berikut adalah skenario *use case* membuat *line damage* yang terdapat pada tabel V.10.

Tabel V.10 Definisi *Use Case* Membuat *Line Damage*

Nama <i>Use Case</i>	Membuat <i>Line Damage</i>
<i>Primary Actor</i>	Bagian <i>Trimming 3</i>
<i>Use Case Description</i>	<i>Use case</i> ini untuk membuat <i>line damage</i> atau jalur kerusakan pada <i>part</i>
<i>Relationship</i>	<i>Association</i> : Bagian PC
<i>Normal Flow of Events</i>	1. Bagian <i>Trimming 3</i> masuk ke menu tampilan utama 2. <i>User</i> memilih menu transaksi 3. <i>User</i> membuka submenu <i>line damage</i> 4. Sistem akan menampilkan <i>form line damage</i> 5. <i>User</i> akan menginputkan ke <i>form line damage</i> 6. <i>User</i> dapat melakukan proses tambah, ubah, cari dan hapus data <i>line damage</i> ke <i>database</i>.

(Sumber: Hasil Analisis, 2018)

8. *Use Case* Melihat *Line Damage*

Berikut adalah skenario *use case* melihat *line damage* yang terdapat pada tabel V.11.

Tabel V.11 Definisi *Use Case* Melihat *Line Damage*

Nama Use Case	Melihat Line Damage
<i>Primary Actor</i>	Bagian <i>Part Control</i>
<i>Use Case Description</i>	<i>Use case</i> ini untuk melihat <i>line damage</i> yang sudah dikirim oleh Bagian <i>Trimming 3</i> , sehingga nantinya bagian <i>part control</i> akan langsung mengganti dan mengirimkan <i>part</i> yang baru
<i>Relationship</i>	<i>Association</i> : Bagian <i>Trimming 3</i>
<i>Normal Flow of Events</i>	1. Bagian <i>part control</i> masuk ke menu tampilan utama 2. <i>User</i> memilih menu <i>line damage</i> Sistem akan menampilkan data <i>line damage</i>

(Sumber: Hasil Analisis, 2018)

9. *Use Case* Membuat Hasil Produksi

Berikut adalah skenario *use case* membuat hasil produksi yang terdapat pada tabel V.12.

Tabel V.12 Definisi *Use Case* Membuat Hasil Produksi

Nama Use Case	Membuat Hasil Produksi
<i>Primary Actor</i>	<i>Trimming 3</i>
<i>Use Case Description</i>	<i>Use case</i> ini membuat hasil produksi
<i>Relationship</i>	<i>Association</i> : Bagian <i>ForemanTrimming 3</i>
<i>Normal Flow of Events</i>	1. <i>Trimming 3</i> masuk ke menu tampilan utama 2. <i>Trimming 3</i> memilih menu transaksi 3. <i>User</i> memilih sub menu hasil produksi lalu memilih tambah data 4. Sistem akan menampilkan <i>form</i> hasil produksi 5. <i>User</i> akan mengisi <i>form</i> hasil produksi dan disimpan ke <i>database</i> 6. Sistem akan menampilkan data hasil produksi yang dibuat

(Sumber: Hasil Analisis, 2018)

10. Use Case Mencetak Laporan

Berikut adalah skenario *use case* mencetak laporan seperti laporan kerusakan *part* dan hasil produksi yang terdapat pada tabel V.13.

Tabel V.13 Definisi *Use Case* Melihat Mencetak Laporan

Nama Use Case	Mencetak Laporan Produksi
Primary Actor	Bagian Foreman Trimming 3
Use Case Description	Use case ini dilakukan untuk mencetak laporan Trimming 3
Relationship	Association: Foreman Trimming 3
Normal Flow Event	1. User memilih menu laporan 2. User memilih laporan yang akan dicetak 3. Sistem menampilkan form laporan yang akan dicetak 4. User memilih cetak 5. Sistem akan mencetak laporan

(Sumber: Hasil Analisis, 2018)

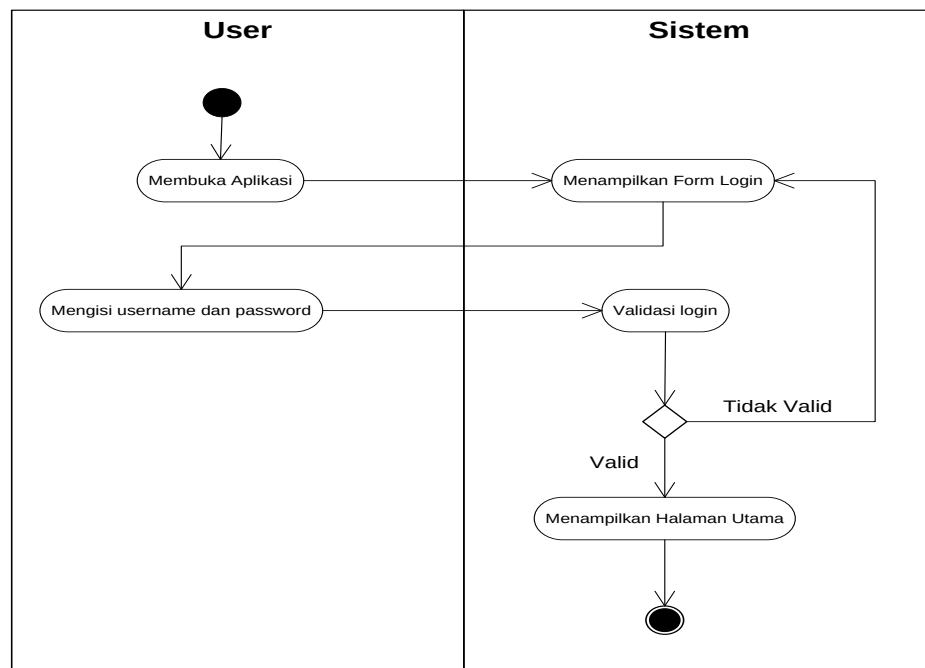
5.3.2 Activity Diagram

Activity diagram digunakan untuk menggambarkan kegiatan-kegiatan yang ada di dalam sistem. Agar memahami sistem yang akan dibuat, maka perlu dibuatkan *activity diagram* tentang sistem informasi pelaporan kerusakan *part* pada Bagian Trimming 3.

1. Activity Diagram Login

Activity diagram login ini menjelaskan alur aktivitas yang dilakukan oleh user, yaitu Foreman Trimming 3, Trimming 3 dan Part Control untuk dapat masuk ke dalam sistem informasi produksi. *Activity diagram* dapat dilihat pada Gambar V.3.

Ga
mba
r
V.3
Acti
vity
Dia
gra
m



Login

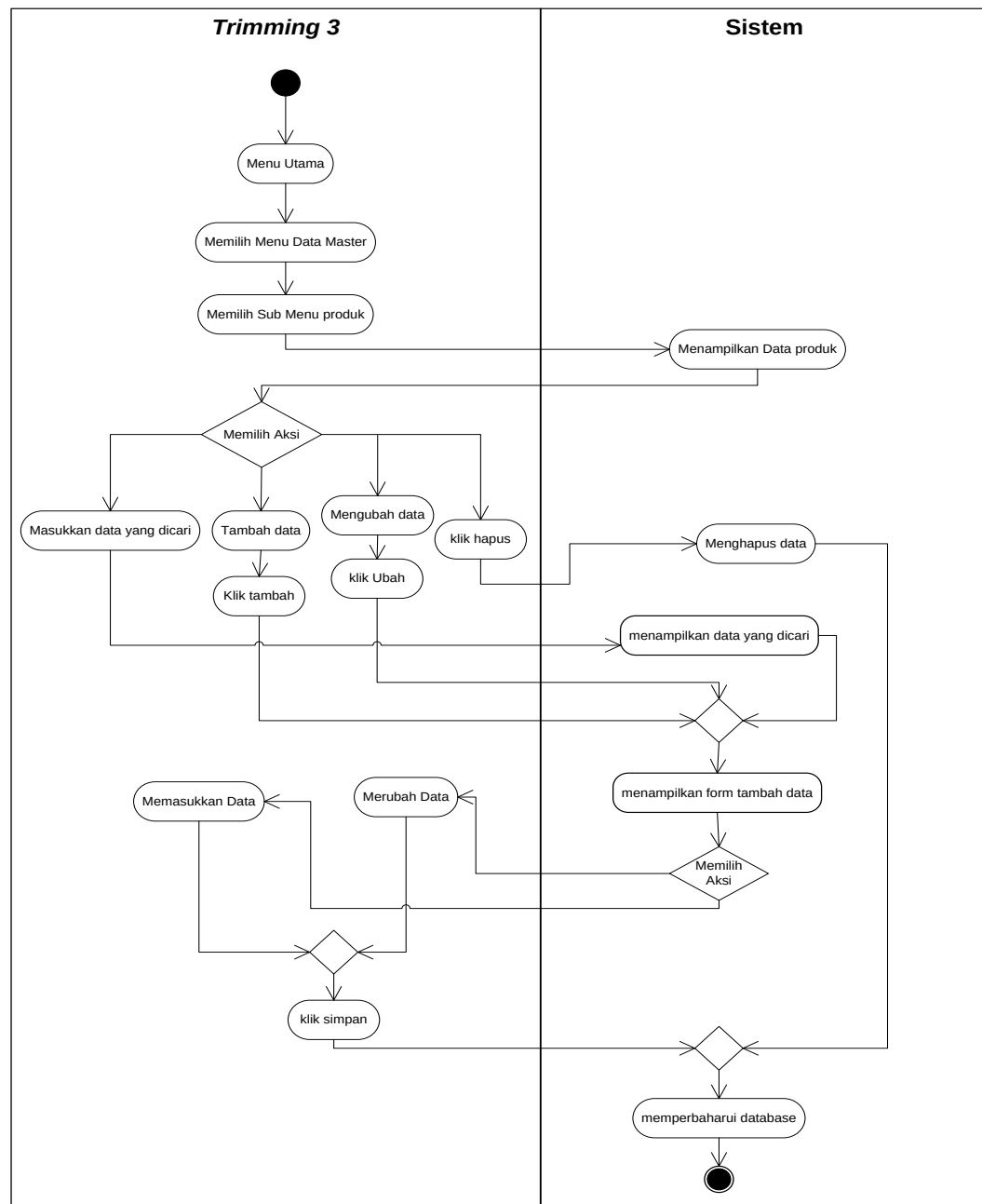
(Sumber: Hasil Analisis, 2018)

2. Activity Diagram Mengelola Data User

Activity diagram mengelola data pengguna berikut ini menunjukkan fungsi-fungsi yang dapat dilakukan pada aplikasi pengolahan data. *Activity diagram* data user yang diusulkan dapat dilihat pada Gambar V.4.

4. Activity Diagram Mengelola Data Produk

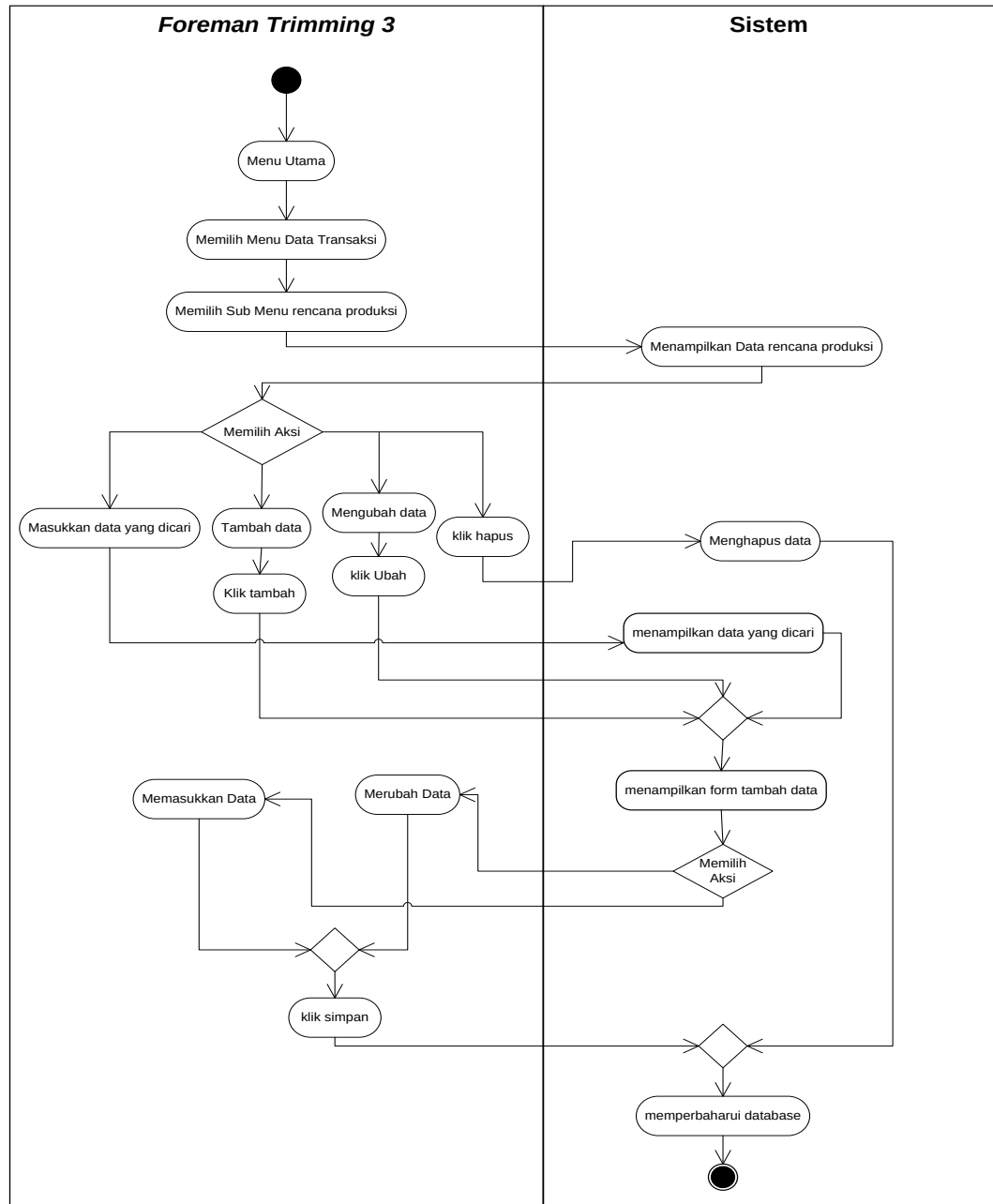
Activity diagram mengelola data *part* berikut ini menunjukkan fungsi-fungsi yang dapat dilakukan pada aplikasi pengolahan data. *Activity diagram* data produk yang diusulkan dapat dilihat pada Gambar V.6.



Gambar V.6 Activity Diagram Mengelola Data Produk
(Sumber: Hasil Analisis, 2018)

5. Activity Diagram Menginput Rencana Produksi

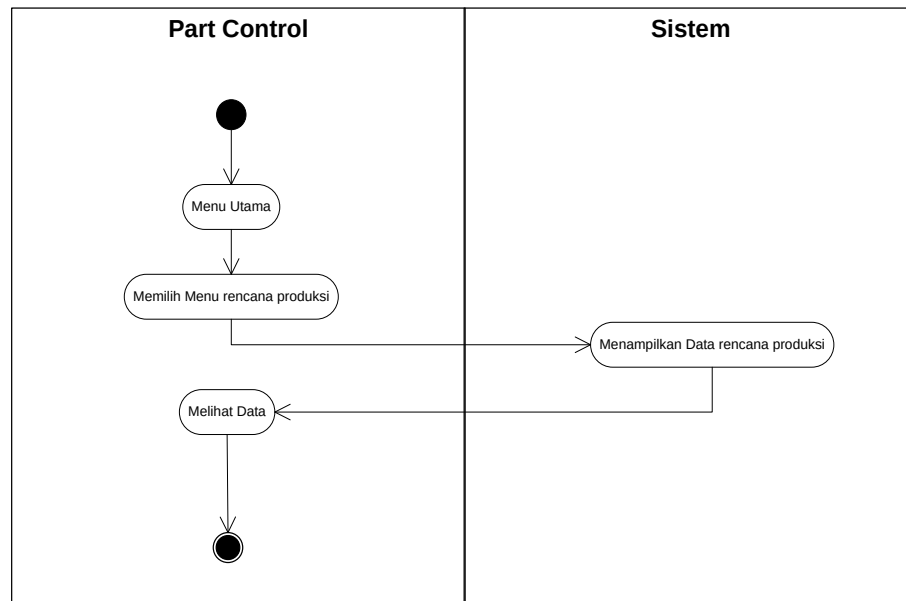
Activity diagram membuat rencana produksi yang dibuat oleh bagian Foreman Trimming 3 berikut ini menunjukkan fungsi-fungsi yang dapat dilakukan pada aplikasi pengolahan data. Activity diagram data rencana produksi yang diusulkan dapat dilihat pada Gambar V.7.



Gambar V.7 *Activity Diagram* membuat rencana produksi
(Sumber: Hasil Analisis, 2018)

6. *Activity Diagram* Melihat Rencana Produksi

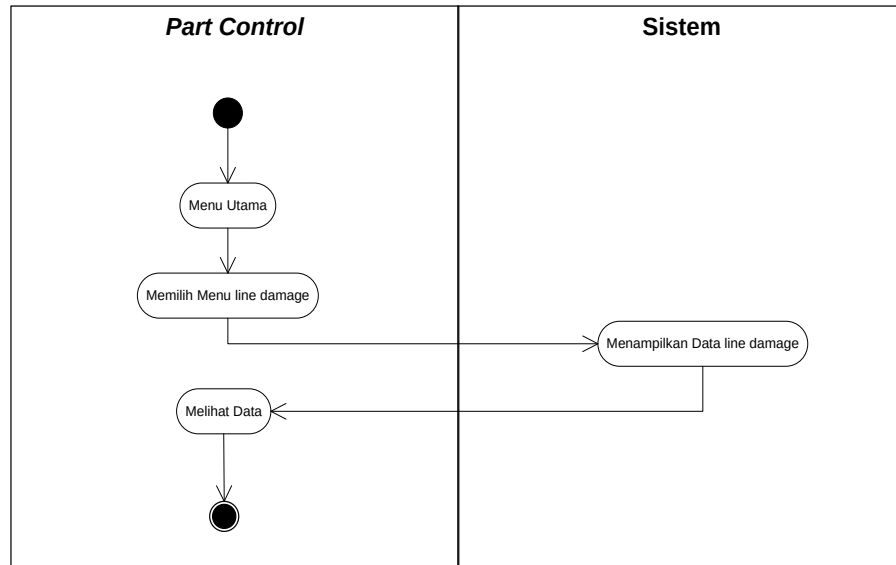
Activity diagram melihat data rencana produksi berikut ini menunjukkan fungsi-fungsi yang dapat dilakukan dalam melihat data rencana produksi. *Activity diagram* melihat data rencana produksi yang diusulkan dapat dilihat pada Gambar V.8.



Gambar V.8 Activity Diagram melihat rencana produksi oleh *Part Control*
(Sumber: Hasil Analisis, 2018)

7. Activity Diagram Membuat Line Damage

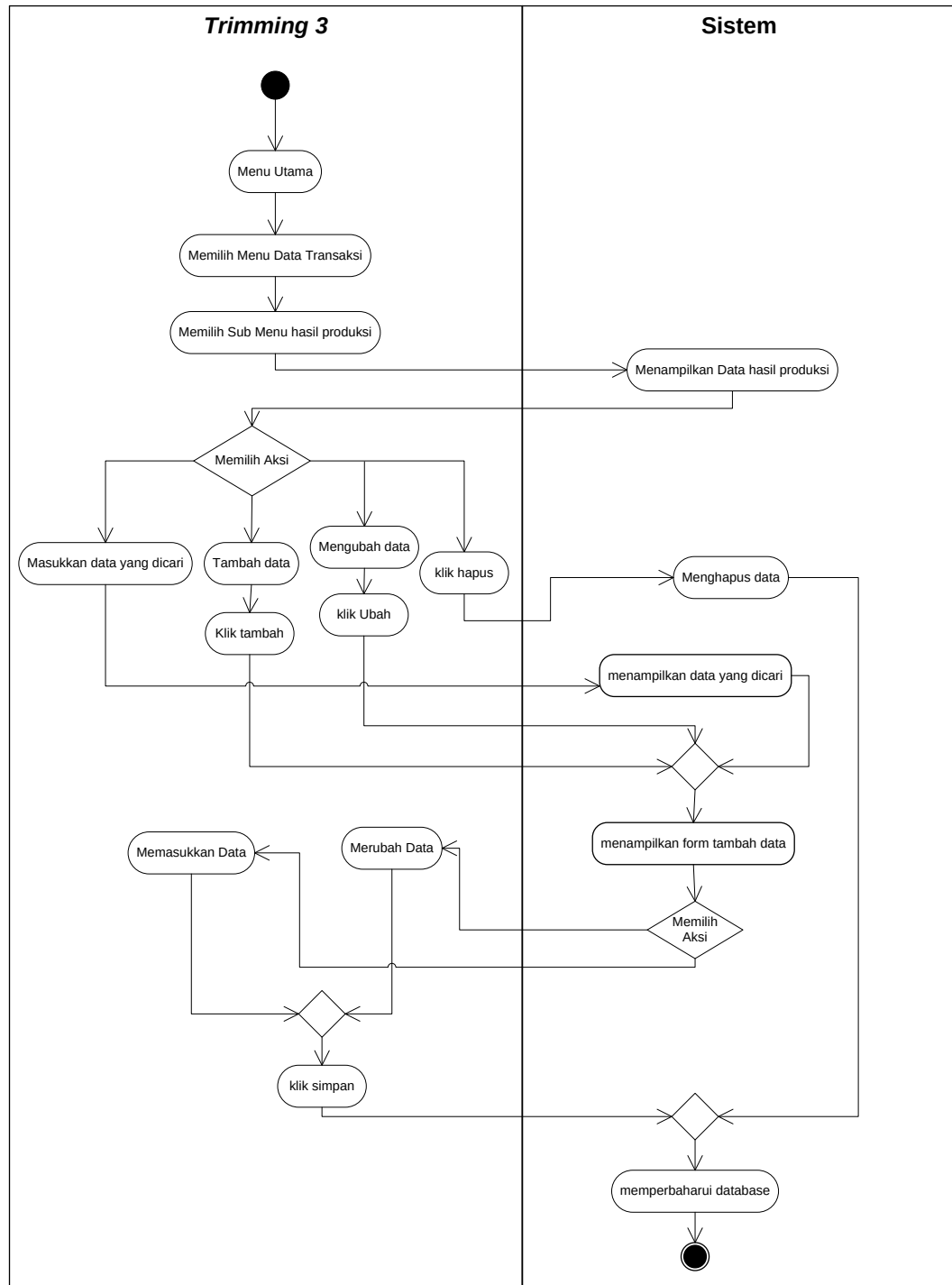
Activity diagram membuat *line damage* berikut ini menunjukkan fungsi-fungsi yang dapat dilakukan dalam membuat *line damage*. Activity diagram membuat *line damage* yang diusulkan dapat dilihat pada Gambar V.9.



Gambar V.10 Activity Diagram melihat line damage
(Sumber: Hasil Analisis, 2018)

9. Activity Diagram Membuat Hasil Produksi

Activity diagram membuat hasil produksi yang dibuat oleh bagian *Trimming* 3 berikut ini menunjukkan fungsi-fungsi yang dapat dilakukan pada aplikasi pengolahan data. Activity diagram data pengguna yang diusulkan dapat dilihat pada Gambar V.11.

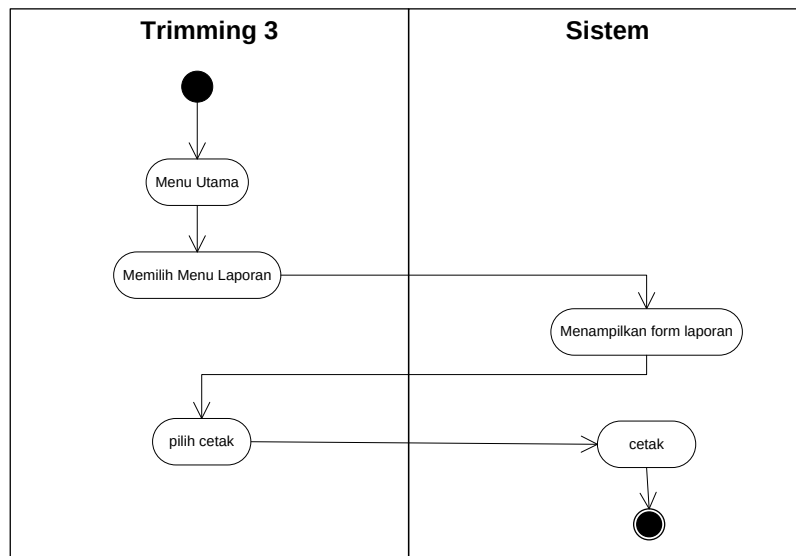


Gambar V.11 Activity Diagram membuat hasil produksi
(Sumber: Hasil Analisis, 2018)

10. Mencetak Laporan hasil produksi

Activity diagram membuat laporan hasil produksi berikut ini menunjukkan fungsi-fungsi yang dapat dilakukan dalam membuat laporan produksi.

Activity diagram mencetak laporan produksi yang diusulkan dapat dilihat pada Gambar V.12.



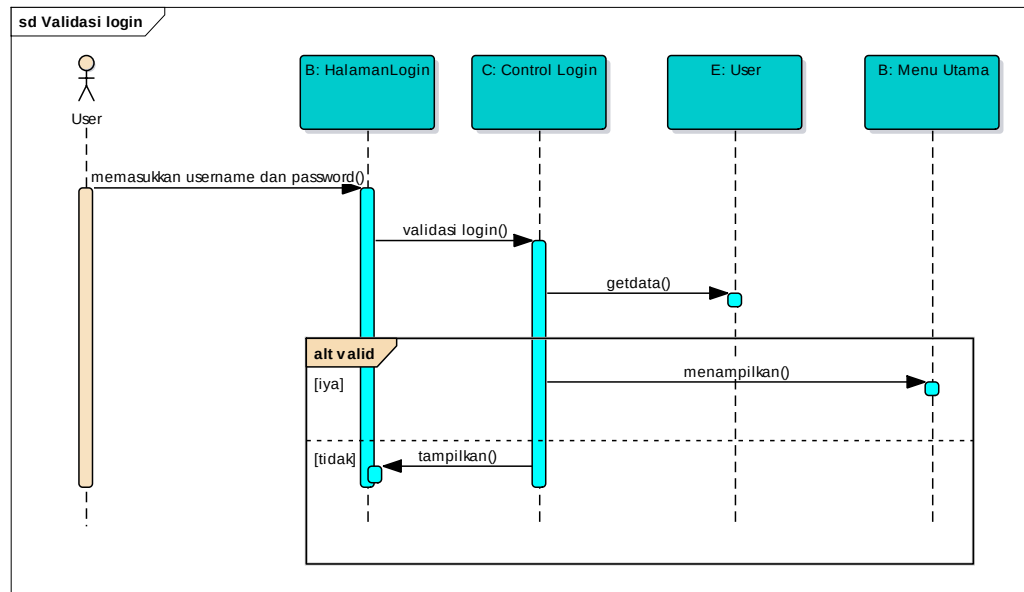
Gambar V.12 *Activity Diagram* mencetak laporan hasil produksi
(Sumber: Hasil Analisis, 2018)

5.3.3 Sequence Diagram

Sequence diagram menggambarkan interaksi yang terjadi pada suatu objek *use case diagram* ketika melakukan suatu proses tertentu, dimana urutan proses ketika melakukan suatu proses tertentu dapat diketahui dengan melihat gambaran pada diagram. Hubungan yang ada pada gambar di bawah ini adalah proses yang dilakukan oleh sistem ketika melakukan proses yang sesuai dengan suatu objek *use case diagram* pada sistem informasi pelaporan kerusakan part *trimming 3*.

1. Sequence Diagram Proses Login

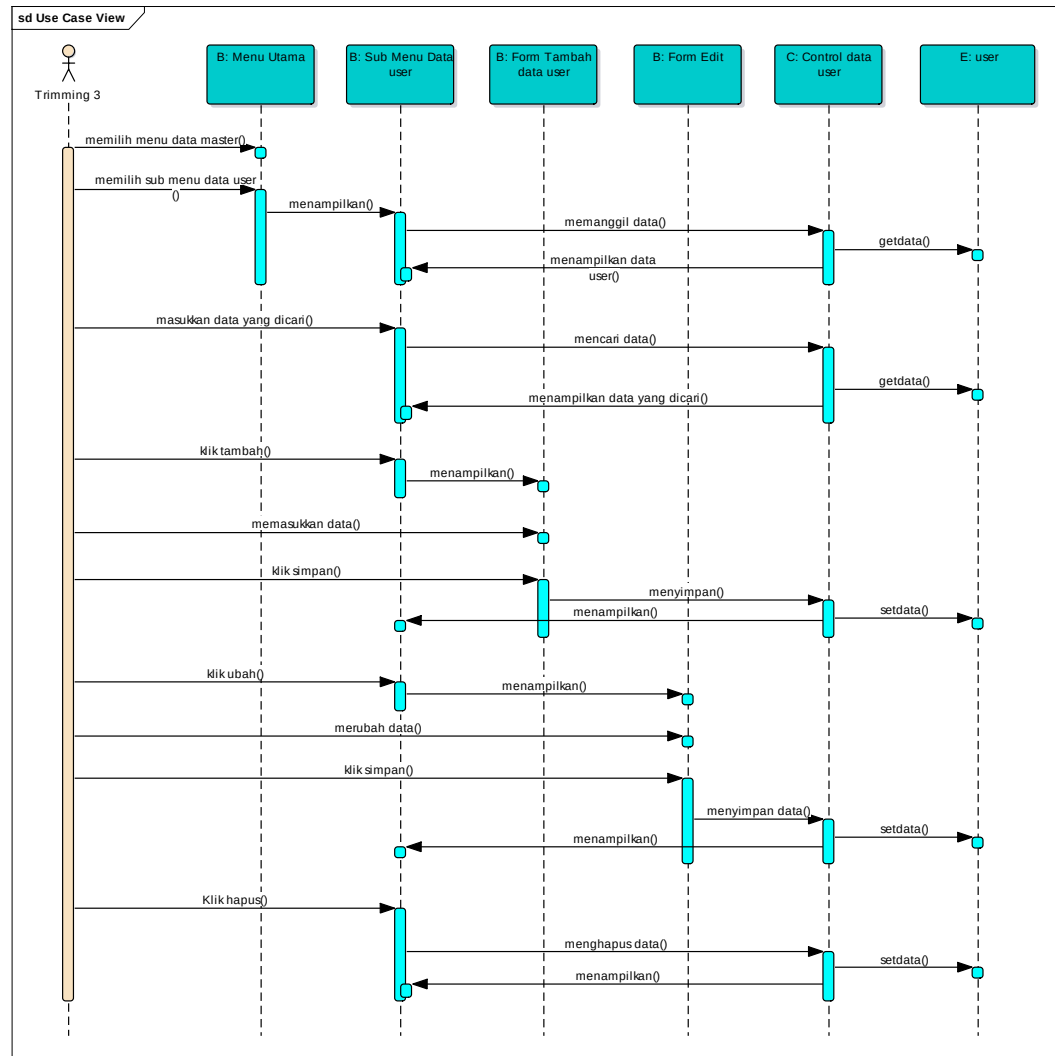
Sequence diagram login menjelaskan sebuah *sequence diagram* dalam proses *login*. Proses ini dilakukan oleh karyawan yang mempunyai hak akses sebelum masuk ke sistem. Adapun *sequence diagram* dari *use case login* dapat dilihat pada Gambar V.13.



Gambar V.13 *Sequence Diagram Login*
(Sumber: Hasil Analisis, 2018)

1. *Sequence Diagram Mengelola Data User*

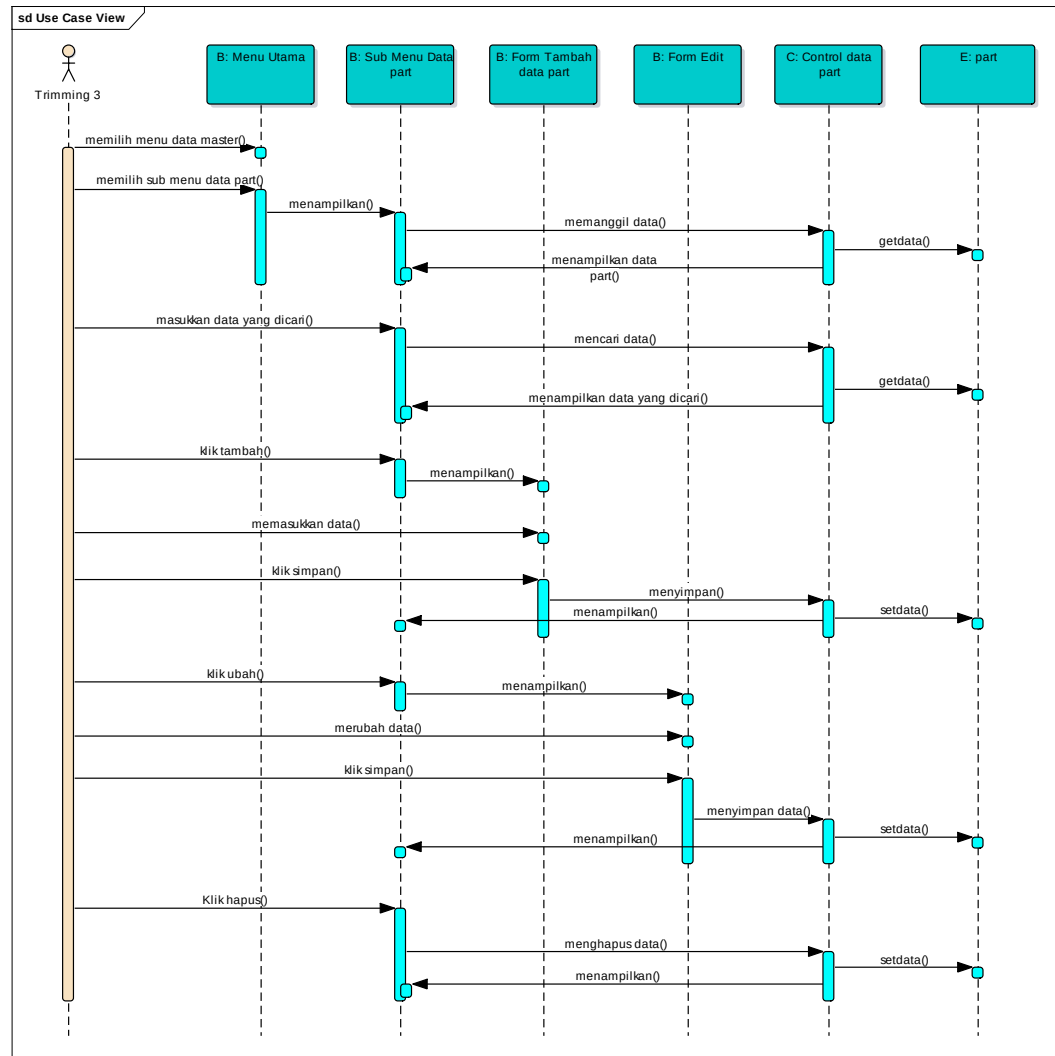
Sequence diagram mengelola data *user* menjelaskan sebuah *sequence diagram* dalam transaksi menambah, menghapus dan mengubah data *user*. Adapun *sequence diagram* dari *use case* mengelola data *user* dapat dilihat pada Gambar V.14.



Gambar V.14 *Sequence Diagram* Mengolah data User
(Sumber: Hasil Analisis, 2018)

3. *Sequence Diagram Mengelola Data Part*

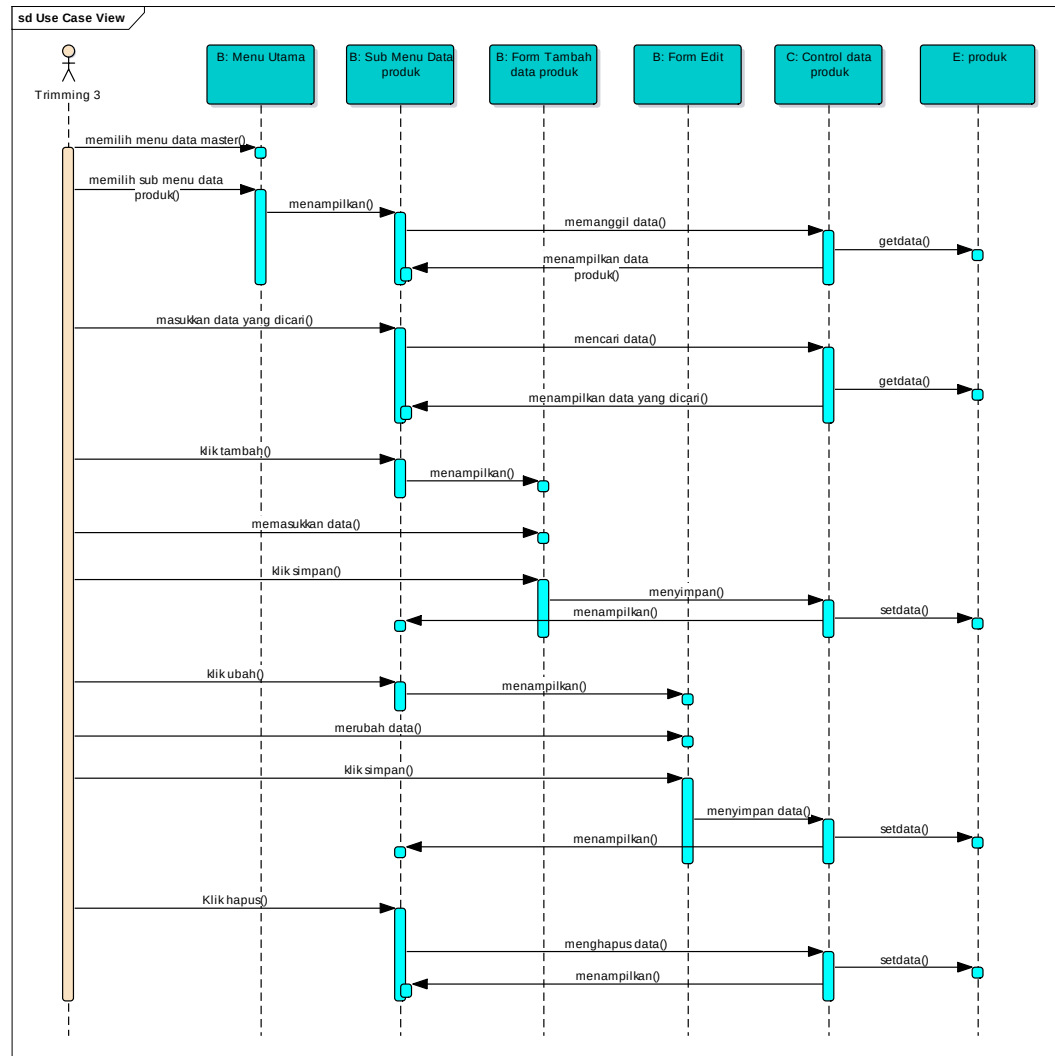
Sequence diagram mengelola data *part* menjelaskan sebuah *sequence diagram* dalam transaksi menambah, menghapus dan mengubah data *part*. Adapun *sequence diagram* dari *use case* mengelola data *part* dapat dilihat pada Gambar V.15.



Gambar V.15 Sequence Diagram Mengolah Data Part
(Sumber: Hasil Analisis, 2018)

4. Sequence Diagram Mengolah Data Produk

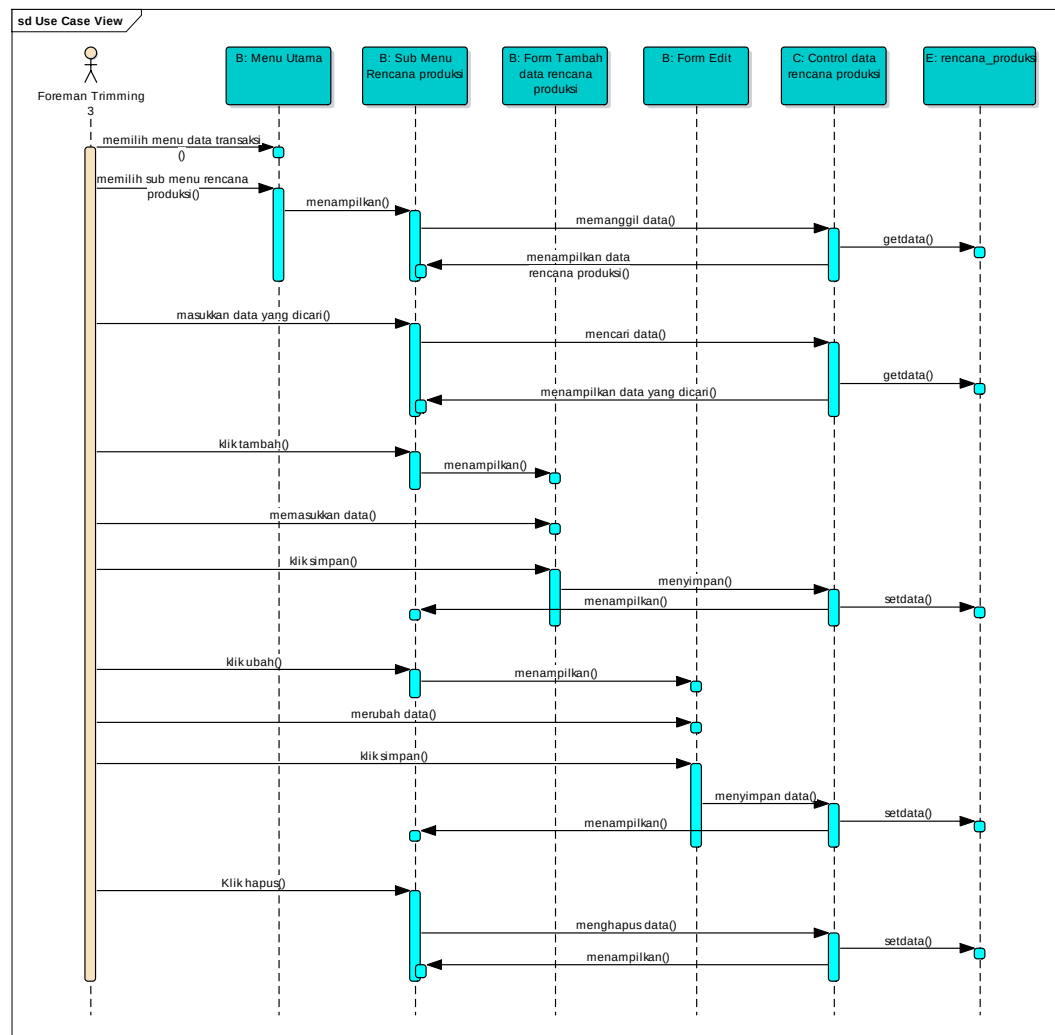
Sequence diagram mengelola data produk menjelaskan sebuah sequence diagram dalam transaksi menambah, menghapus dan mengubah data produk Adapun sequence diagram dari use case mengelola data produk dapat dilihat pada Gambar V.16.



Gambar V.16 Sequence Diagram Mengolah Data Produk
(Sumber: Hasil Analisis, 2018)

5. Sequence Diagram Menginput Rencana Produksi

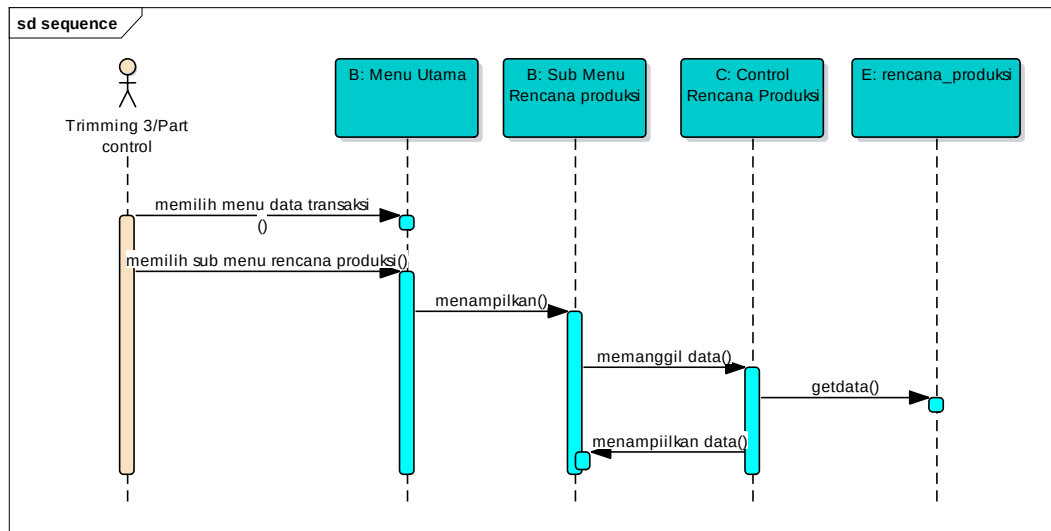
Sequence diagram perencanaan produksi menjelaskan sebuah sequence diagram dalam proses menginputkan rencana produksi. Adapun sequence diagram dari use case rencana produksi dapat dilihat pada Gambar V.17.



Gambar V.17 *Sequence Diagram* Menginput Rencana Produksi
(Sumber: Hasil Analisis, 2018)

6. *Sequence Diagram* Melihat Rencana Produksi

Sequence diagram perencanaan produksi menjelaskan sebuah *sequence diagram* dalam bagian *Trimming* 3 dan *part control* melihat rencana produksi. Adapun *sequence diagram* dari *use case* rencana produksi dapat dilihat pada Gambar V.18.

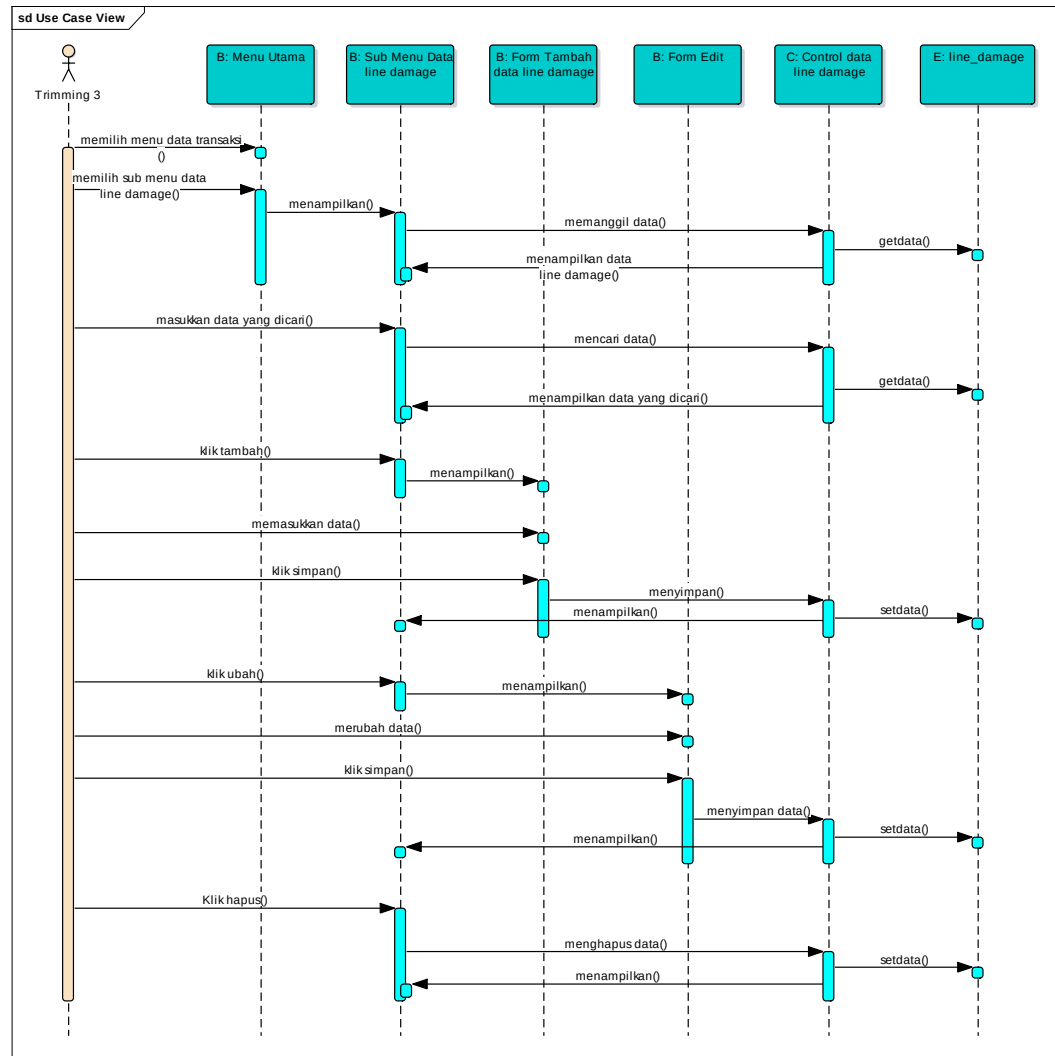


Gambar V.18 *Sequence Diagram* Melihat Data Rencana Produksi oleh *Part Control*

(Sumber: Hasil Analisis, 2018)

7. *Sequence Diagram* Membuat *Line Damage*

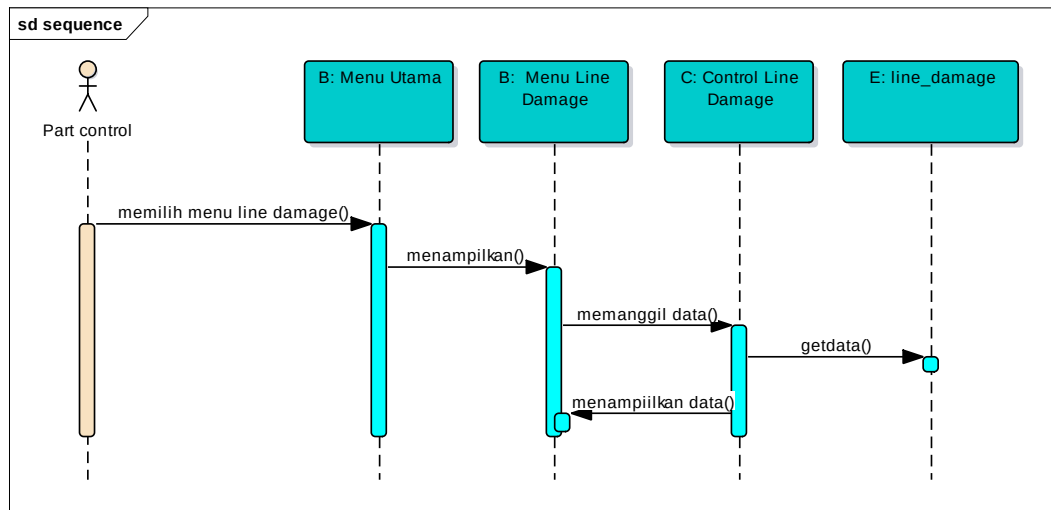
Sequence diagram membuat *line damage* menjelaskan sebuah *sequence diagram* dalam proses membuatt *line damage*. Adapun *sequence diagram* dari *use case* membuatt *line damage* dapat dilihat pada Gambar V.19.



Gambar V.19 Sequence Diagram Membuat Data line damage
(Sumber: Hasil Analisis, 2018)

8. Sequence Diagram Melihat Line Damage

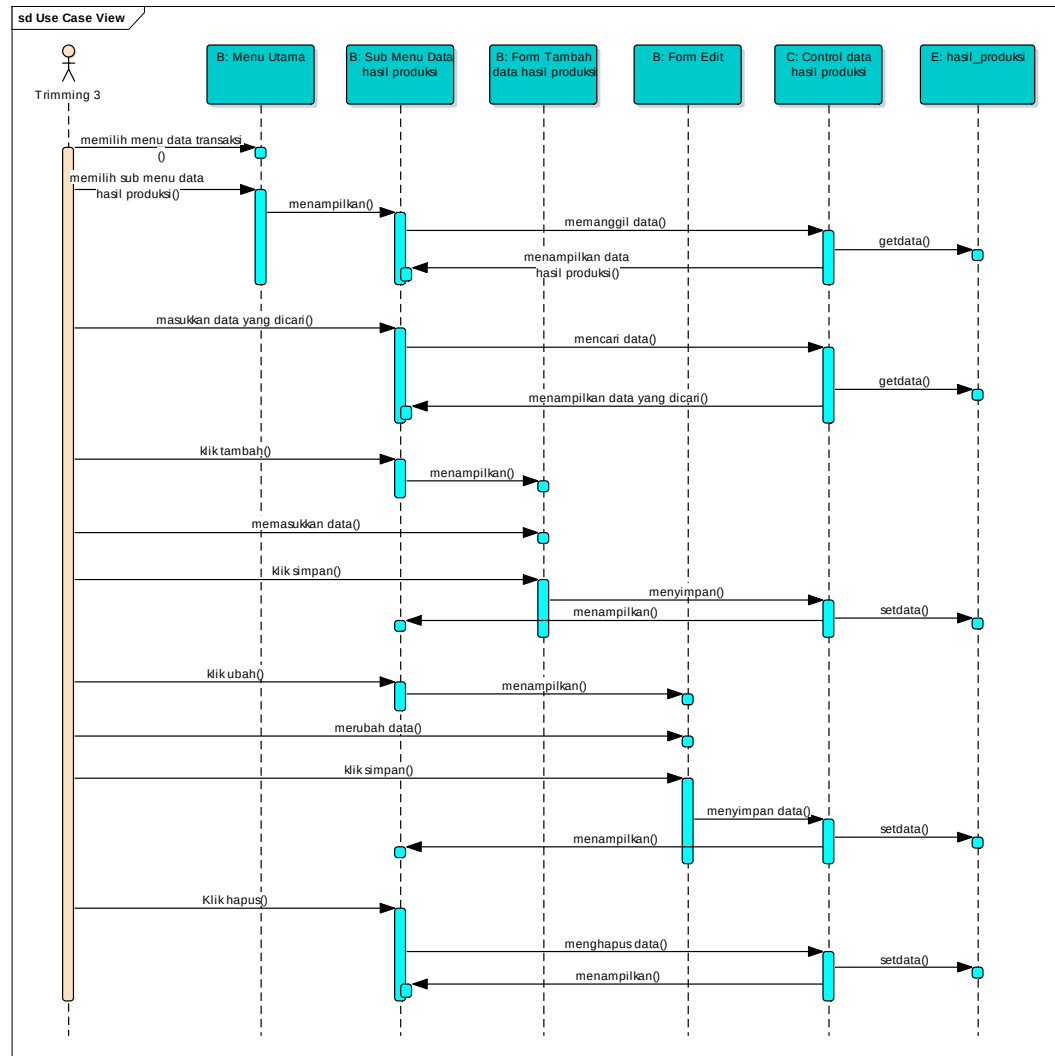
Sequence diagram melihat line damage menjelaskan sebuah sequence diagram dalam melihat line damage. Adapun sequence diagram dari use case sistem informasi produksi dapat dilihat pada Gambar V.20.



Gambar V.20 *Sequence Diagram* Melihat Data *line damage*
(Sumber: Hasil Analisis, 2018)

9. *Sequence Diagram* Membuat Hasil Produksi

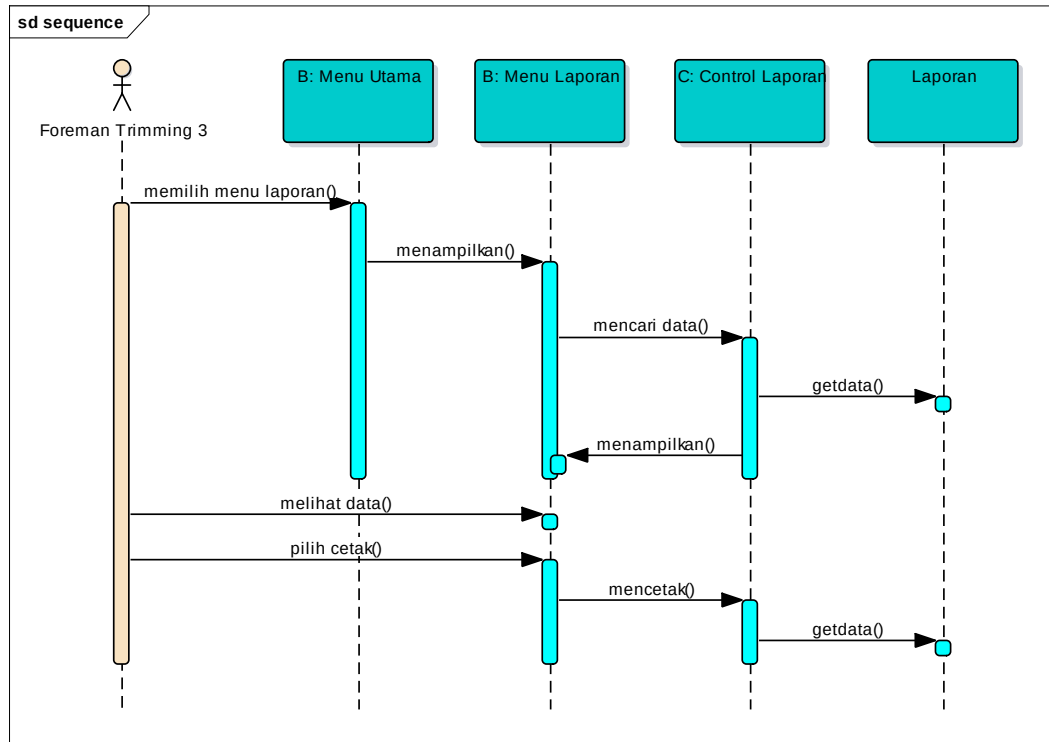
Sequence diagram membuat hasil produksi menjelaskan sebuah *sequence diagram* dalam membuat hasil produksi. Adapun *sequence diagram* dari *use case* sistem informasi produksi dapat dilihat pada Gambar V.21.



Gambar V.21 *Sequence Diagram* Membuat hasil produksi
(Sumber: Hasil Analisis, 2018)

10. *Sequence Diagram* Mencetak Laporan Hasil Produksi

Sequence diagram mencetak laporan hasil produksi menjelaskan sebuah *sequence diagram* dalam proses mencetak laporan. Adapun *sequence diagram* dari *use case* mencetak laporan hasil produksi dapat dilihat pada Gambar V.22.

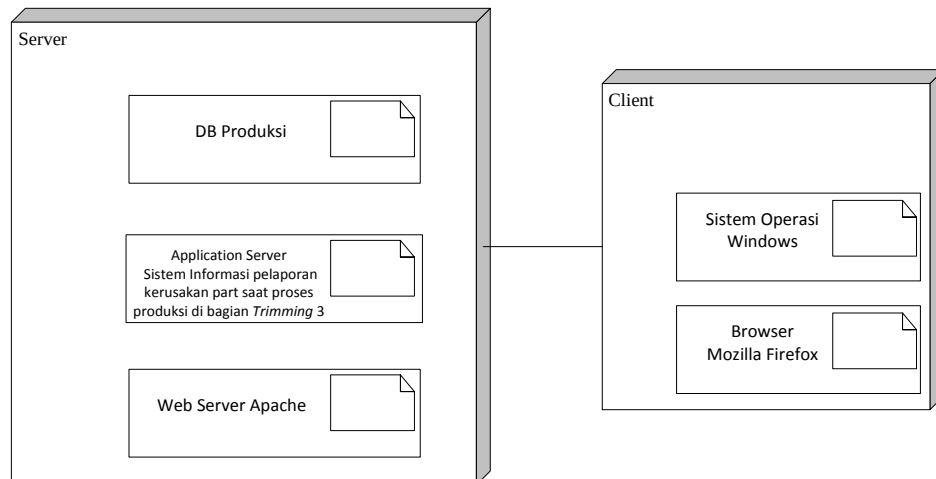


Gambar V.22 *Sequence Diagram* Mencetak Laporan kerusakan *part* dan hasil produksi

(Sumber: Hasil Analisis, 2018)

5.3.4. *Class Diagram*

Class diagram menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem *class diagram* memperlihatkan hubungan antar kelas dan penjelasan *detail* tiap-tiap kelas di dalam model desain dari suatu sistem. *Class diagram* sistem informasi pelaporan kerusakan *part* di bagian *Trimming 3* yang diusulkan dapat dilihat pada Gambar V.23.



Gambar V.24 *Deployment Diagram* Sistem Informasi Pelaporan kerusakan *part*

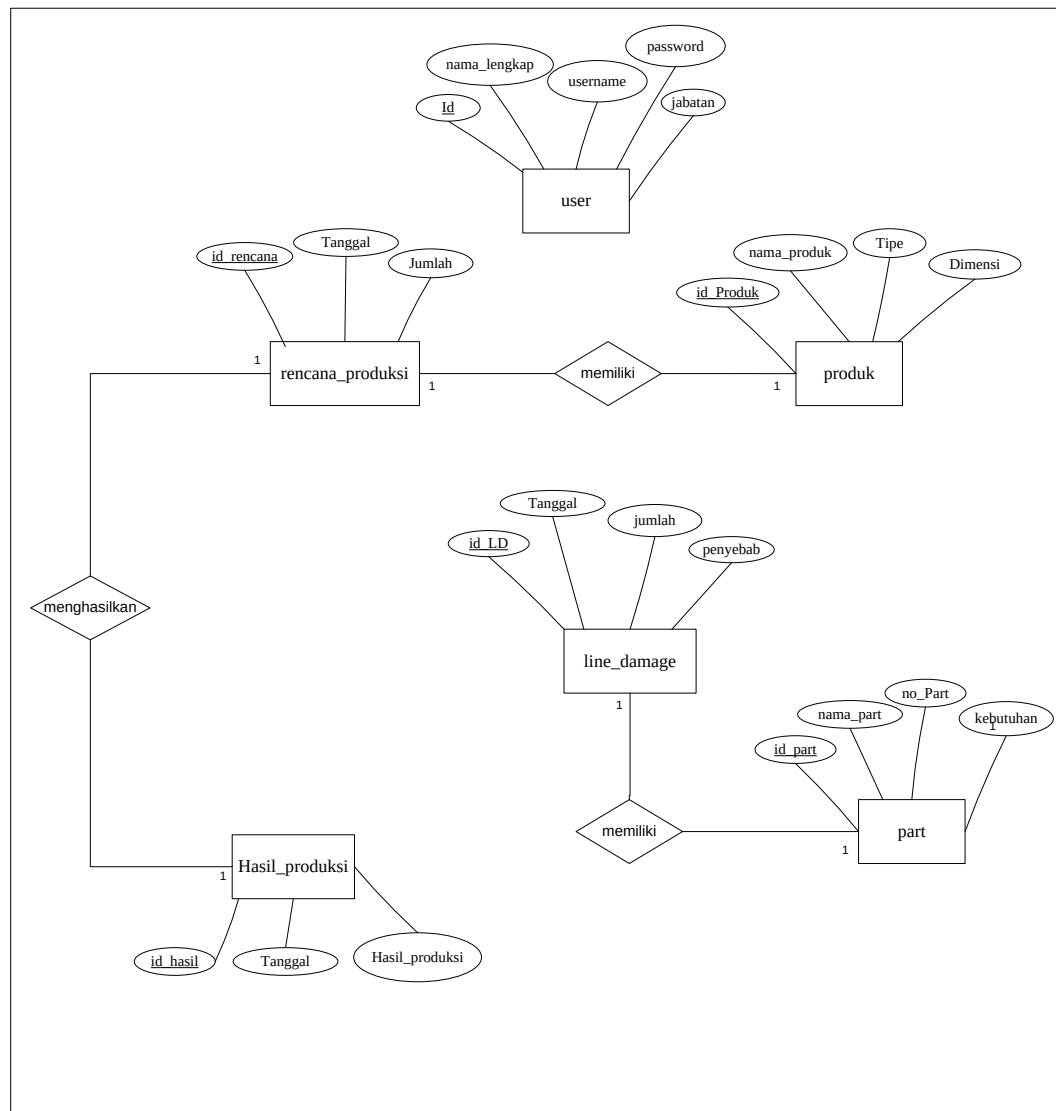
(Sumber: Hasil Analisis, 2018)

5.4. Pemodelan Data Sistem Usulan

Pemodelan data pada sistem informasi produksi pelaporan kerusakan *part* di Bagian *Trimming 3* pada PT Krama Yudha Ratu Motor menggunakan dua cara yaitu *Entity Relationship Diagram* (ERD) yang digunakan untuk menjelaskan hubungan antar data dalam basis data dan kamus data yang digunakan untuk menjelaskan isi dari *database* yang digunakan dalam sistem usulan.

5.4.1. *Entity Relationship Diagram* (ERD)

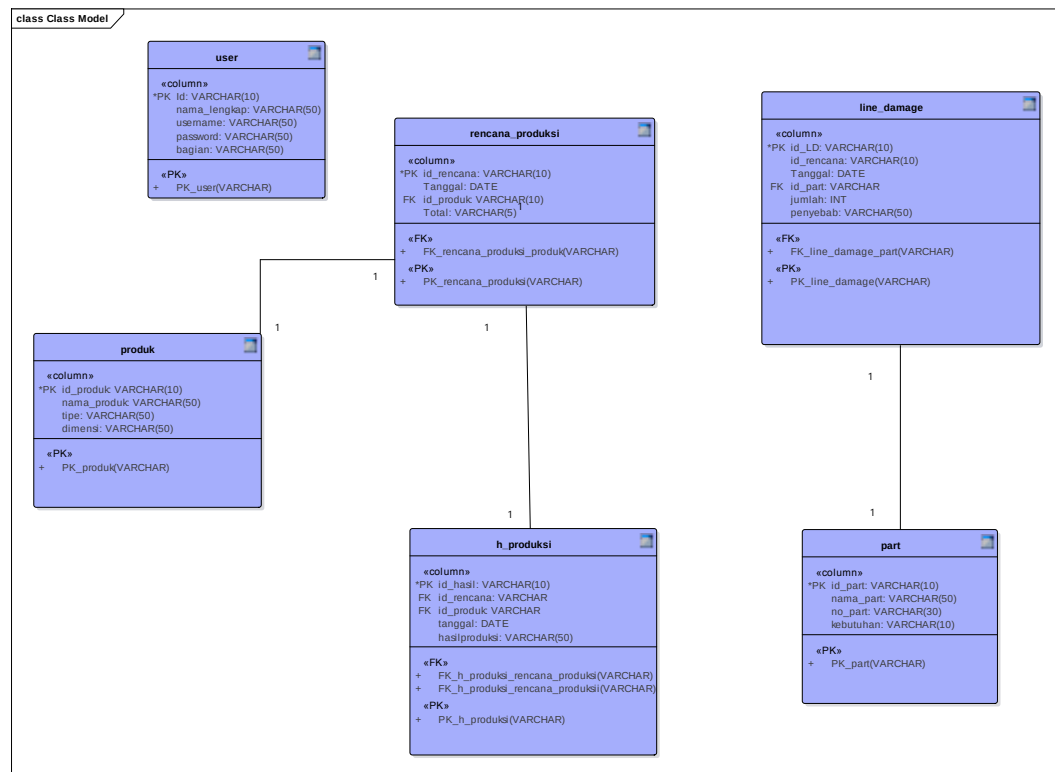
ERD merupakan suatu model untuk menjelaskan hubungan antar data dalam basis data berdasarkan objek-objek dasar data yang mempunyai hubungan antar relasi. ERD untuk memodelkan struktur data dan hubungan antar data, untuk menggambarkan digunakan beberapa notasi dan simbol. ERD sistem informasi produksi yang diusulkan dapat dilihat pada Gambar V.25.



Gambar V.25 *Entity Relationship Diagram* Usulan
(Sumber: Hasil Analisis, 2018)

5.4.1 *Conceptual Data Model (CDM)*

Conceptual Data Model digunakan untuk menggambarkan atau merepresentasikan seluruh muatan informasi yang dikandung oleh basis data, berikut merupakan *Conceptual Data Model* dari sistem usulan:



Gambar V.26 CDM Sistem Informasi Pelaporan kerusakan *part*
(Sumber: Hasil Analisis, 2018)

5.4.3. Kamus Data

Kamus data adalah suatu daftar data elemen yang terorganisir dengan definisi yang tetap dan sesuai dengan sistem, sehingga *user* dan analisis sistem mempunyai pengertian yang sama tentang *input*, *output* dan komponen *data store*. Pada tahap perancangan sistem, kamus data digunakan untuk merancang input, laporan dan basis data. Berikut adalah kamus data sistem informasi produksi.

1. Spesifikasi Tabel Data *User*

Nama Tabel : user

Fungsi : Untuk memberikan hak akses pada pengguna

Tipe : File data master

Tabel V.14 Tabel *User*

No	Nama Elemen	Akronim	Tipe	Panjang	Keterangan
1.	ID User	id	<i>varchar</i>	10	<i>Primary Key</i>
2.	Nama Lengkap	nama_lengkap	<i>varchar</i>	50	<i>Not Null</i>
3.	<i>User Name</i>	username	<i>varchar</i>	50	<i>Not Null</i>
4.	<i>Password</i>	password	<i>varchar</i>	20	<i>Not Null</i>
5.	Bagian	bagian	<i>varcahr</i>	50	<i>Not Null</i>

(Sumber: Hasil Analisis, 2018)

2. Spesifikasi Tabel Produk

Nama Tabel : produk

Fungsi : Untuk menyimpan data produk

Tipe : File data master

Tabel V.15 Tabel Produk

No	Nama Elemen	Akronim	Tipe	Panjang	Keterangan
1.	Id Produk	id_produk	<i>varhar</i>	10	<i>Primary Key</i>
2.	Nama Produk	nama_produk	<i>varchar</i>	50	<i>Not Null</i>
3.	Tipe	tipe	<i>varchar</i>	50	<i>Not Null</i>
4.	Dimensi	dimensi	<i>Int</i>	20	<i>Not Null</i>

(Sumber: Hasil Analisis, 2018)

3. Spesifikasi Tabel *Part*

Nama Tabel : part

Fungsi : Untuk menyimpan data *part*

Tipe : File data master

Tabel V.16 Tabel *Part*

No	Nama Elemen	Akronim	Tipe	Panjang	Keterangan
1.	Id_part	id_part	varchar	10	Primary Key
2.	Nama.Part	nama_part	varchar	10	Not Null
3.	No. Part	no_part	varchar	50	Not Null
4.	Kebutuhan	kebutuhan	varchar	10	Not Null

(Sumber: Hasil Analisis, 2018)

4. Spesifikasi Tabel Rencana Produksi

Nama Tabel : rencana_produksi

Fungsi : Untuk menyimpan data Rencana Produksi

Tipe : File Transaksi

Tabel V.17 Tabel Rencana Produksi

No	Nama Elemen	Akronim	Tipe	Panjang	Keterangan
1.	ID Rencana	id_rencana	varchar	10	Primary Key
2.	Tanggal	Tanggal	Date		Not Null
3.	ID Produk	id_produk	varchar	10	Foreign Key
4.	Total	Total	varchar	5	Not Null

(Sumber: Hasil Analisis, 2018)

5. Spesifikasi Tabel *Line Damage*

Nama Tabel : line_damage

Fungsi : Untuk menyimpan data *line damage*

Tipe : File Transaksi

Tabel V.18 Tabel Data *Line Damage*

No	Nama Elemen	Akronim	Tipe	Panjang	Keterangan
1.	ID LD	id_LD	varchar	10	Primary Key
2.	Tanggal	Tanggal	Date		Not Null
3.	Id.Part	id_part	varchar	10	Foreign Key

Tabel V.18 Tabel Data *Line Damage* (lanjutan)

No	Nama Elemen	Akronim	Tipe	Panjang	Keterangan
4.	Jumlah	jumlah	<i>int</i>	20	<i>Not Null</i>
5.	Jenis Kerusakan	Jenis_kerusakan	<i>varchar</i>	50	<i>Not Null</i>
6.	Penyebab	penyebab	<i>varchar</i>	100	<i>Not Null</i>

(Sumber: Hasil Analisis, 2018)

6. Spesifikasi Tabel Hasil Produksi

Nama Tabel : hasil_produksi

Fungsi : Untuk menyimpan data hasil produksi

Tipe : File Transaksi

Tabel V.19 Tabel Hasil Produksi

No	Nama Elemen	Akronim	Tipe	Panjang	Keterangan
1.	ID Hasil	id_hp	<i>varchar</i>	10	<i>Primary Key</i>
2.	ID Rencana	id_rencana	<i>varchar</i>	10	<i>Foreign Key</i>
3.	ID Produk	Id_produk	<i>varchar</i>	10	<i>Foreign Key</i>
4.	Tanggal	Tanggal	<i>Date</i>		<i>Not Null</i>
5.	Hasil produksi	jumlah	<i>varchar</i>	5	<i>Not Null</i>

(Sumber: Hasil Analisis, 2018)

5.5 Analisis Desain Program

Tahap ini merupakan tahap kedua dalam metodologi prototipe evolusioner, yaitu tahap membuat sebuah prototipe dari program atau aplikasi. Dimulai dengan analisis desain usulan meliputi pembuatan struktur menu program dan *interface* program sampai dengan program dapat dijalankan.

5.5.1. WND (*Windows Navigation Diagram*)

Windows Navigation Diagram (WND) digunakan untuk mendokumentasikan sebuah struktur yang menggambarkan hubungan antar fungsi dalam program secara hirarkis. Diagram ini memuat semua modul judul yang ada dalam sistem beserta nama dan nomornya. Diagram ini menunjukkan *window-window* tersebut. Sebuah window dapat digambarkan sebagai sebuah *state*. *State*

ini memiliki nama dan berisi gambar *miniature* window. Transisi antar *state* dipicu oleh ditekannya sebuah tombol yang menghubungkan dua *window*.

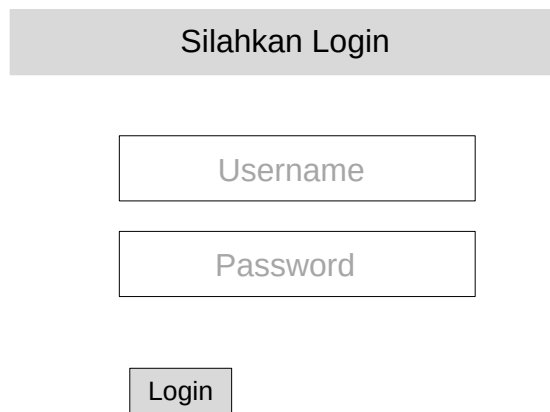
Perancangan WND aplikasi usulan sistem informasi pelaporan kerusakan *part* saat proses produksi di bagian *Trimming* 3 dapat dilihat pada Gambar V.25

5.5.2. Perancangan *Interface* Sistem

Rancangan *interface* dari program pelaporan produksi ini adalah sebagai berikut:

1. *Form Login*

Form login adakah *form* yang digunakan untuk masuk ke dalam program aplikasi. Untuk masuk ke dalam aplikasi, *user* harus memasukkan *username* dan *password* yang benar. Rancangan *form login* dapat dilihat pada Gambar V.28.



The image shows a login window with a light gray title bar containing the text "Silahkan Login". Below the title bar, there are two white input fields with gray borders. The first field is labeled "Username" and the second field is labeled "Password". Below these fields is a small gray button with the text "Login" in white.

Gambar V.28 *Interface Login*
(Sumber: Hasil Analisis, 2018)

2. *Form Menu Utama*

Form menu utama adalah tampilan awal pada aplikasi yang berisi enam menu. Menu tersebut adalah halaman utama, data master, transaksi dan laporan. Rancangan menu utama dapat dilihat pada Gambar V.29 berikut:

Gambar V.29 *Interface Form Menu Utama*
(Sumber: Hasil Analisis, 2018)

3. *Form Data Master User*

Form data master user adalah form yang digunakan untuk mengelola data user. Rancangan *interface* dari form data user dapat dilihat pada Gambar V.30 berikut:

Gambar V.30 *Interface Form Data Master User*
(Sumber: Hasil Analisis, 2017)

Pada *interface form* data master *user* terdapat kolom search yang berfungsi untuk mencari data yang dibutuhkan dan tiga tombol, yaitu tambah, ubah dan hapus. Jika tombol tambah atau ubah ditekan maka akan tampil *form input user*. Sedangkan tombol hapus digunakan untuk menghapus data *user*. *Interface form input user* dapat dilihat pada Gambar V.31 berikut:

Gambar V.31 *Interface Form Input Data User*
(Sumber: Hasil Analisis, 2018)

4. *Form Data Master Part*

Form data master *part* adalah *form* yang digunakan untuk mengelola data *part*. Rancangan *interface* dari *form* data *part* dapat dilihat pada Gambar V.32 berikut:

Gambar V.32 *Interface Form Data Master Part*
(Sumber: Hasil Analisis, 2018)

Pada *interface form data master part* terdapat tiga tombol, yaitu tambah, ubah dan hapus serta terdapat kolom cari untuk mencari data yang dibutuhkan . Jika tombol tambah atau ubah ditekan maka akan tampil *form input part* Sedangkan tombol hapus digunakan untuk menghapus data *part* yang dipilih. *Interface form input part* dapat dilihat pada Gambar V.33 berikut:

Gambar V.33 *Interface Form Data Input Master Part*
(Sumber: Hasil Analisis, 2018)

5. *Form Data Master Produk*

Form data master produk adalah *form* yang digunakan untuk mengelola data produk. Rancangan *interface* dari *form data produk* dapat dilihat pada Gambar V.34 berikut:

Gambar V.36 *Interface Form Data Transaksi Rencana Produksi*
(Sumber: Hasil Analisis, 2018)

Pada *interface form* data transaksi rencana produksi terdapat tiga tombol, yaitu tambah, ubah dan hapus serta terdapat kolom cari untuk mencari data yang dibutuhkan. Jika tombol tambah atau ubah ditekan maka akan tampil *form input* rencana produksi. Sedangkan tombol hapus digunakan untuk menghapus data produk yang dipilih. *Interface form input* rencana produksi dapat dilihat pada Gambar V.37 berikut:

Gambar V.37 *Interface Form Input Data Transaksi Rencana Produksi*
(Sumber: Hasil Analisis, 2018)

7. *Form Data Transaksi Line Damage*

Form data transaksi *line damage* adalah *form* yang digunakan untuk menginputkan data *part* rusak yang diakibatkan karena kesalahan operator saat proses pemasangan atau perakitan untuk minta diganti oleh bagian *part control*. Rancangan *interface* dari *form* data *line damage* dapat dilihat pada Gambar V.38 berikut:

Gambar V.38 *Interface Form Data Transaksi Line Damage*

(Sumber: Hasil Analisis, 2018)

Pada *interface form* data transaksi *line damage* terdapat tiga tombol, yaitu tambah, ubah dan hapus serta terdapat kolom cari untuk mencari data yang dibutuhkan. Jika tombol tambah atau ubah ditekan maka akan tampil *form input line damage*. Sedangkan tombol hapus digunakan untuk menghapus data produk yang dipilih. *Interface form input line damage* dapat dilihat pada Gambar V.39 berikut:

The screenshot shows a web application interface for PT KRM. On the left is a sidebar menu with the following items: Menu Utama, Data Master, Data Transaksi (highlighted with a dropdown arrow), Rencana Produksi, Line Damage, Hasil Produksi, Rencana Produksi, Line Damage, Laporan, and Grafik. The main content area is titled 'Form Input Line Damage' and contains the following input fields: ID LD, Tanggal, ID Part, Jumlah, Jenis Kerusakan, and Penyebab. A 'Simpan' button is located at the bottom left of the form area. A 'Logout' button is visible in the top right corner of the header.

Gambar V.39 *Interface Form Input Data Transaksi Line Damage*

(Sumber: Hasil Analisis, 2018)

8. *Form Data Transaksi Hasil Produksi*

Form data transaksi adalah *form* yang digunakan untuk mengelola data hasil produksi. Rancangan *interface* dari *form* data hasil produksi dapat dilihat pada Gambar V.40 berikut:

PT KRM Logout

Form Hasil Produksi

Records per page: 10 Search:

No	ID Hasil	ID Rencana	Tanggal	Hasil Produksi	Total	ID LD	
							Tambah Ubah Hapus

Gambar V.40 *Interface Form Data Transaksi Hasil Produksi*
(Sumber: Hasil Analisis, 2018)

Pada *interface form* data transaksi rencana produksi terdapat tiga tombol, yaitu tambah, ubah dan hapus serta terdapat kolom cari untuk mencari data yang dibutuhkan. Jika tombol tambah atau ubah ditekan maka akan tampil *form input* hasil produksi. Sedangkan tombol hapus digunakan untuk menghapus data produksi yang dipilih. *Interface form input* hasil produksi dapat dilihat pada Gambar V.41 berikut:

PT KRM Logout

Form Input Hasil Produksi

ID Hasil:

ID Rencana:

Tanggal:

HasilProduksi:

Simpan

Gambar V.41 *Interface Form Input Data Transaksi Hasil Produksi*
(Sumber: Hasil Analisis, 2018)

9. *Form* Lihat Data Rencana Produksi

Form data transaksi adalah *form* yang digunakan untuk melihat rencana produksi harian. Rancangan *interface* dari *form* lihat rencana produksi dapat dilihat pada Gambar V.42 berikut:

PT KRM			
			Logout
Menu Utama			
Data Master			
Data Transaksi			
Rencana Produksi			
Line Damage			
Laporan			
Grafik			

Rencana Produksi Harian			
ID Rencana	Tipe Produk	Tanggal	Jumlah

Gambar V.42 *Interface Form Data Lihat Rencana produksi*
(Sumber: Hasil Analisis, 2018)

10. *Form Lihat Data Line Damage*

Form data transaksi adalah form yang digunakan untuk melihat data line damage. Rancangan interface dari form lihat line damage dapat dilihat pada Gambar V.43 berikut:

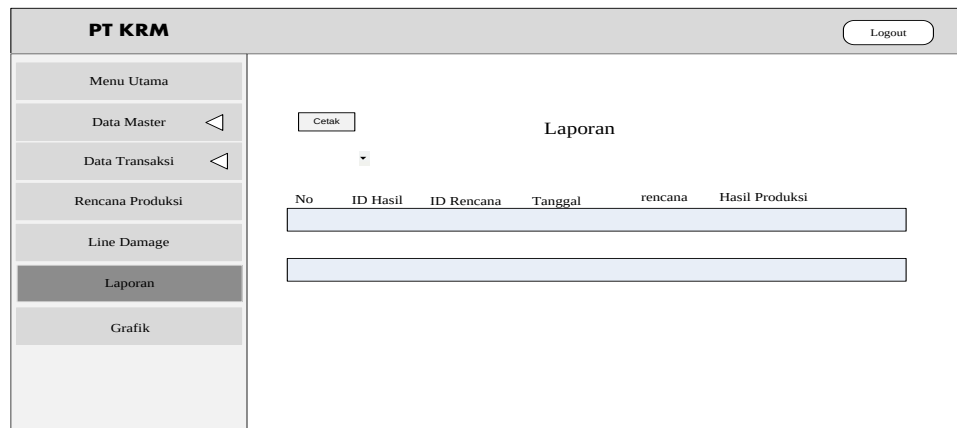
PT KRM						
						Logout
Menu Utama						
Data Master						
Data Transaksi						
Rencana Produksi						
Line Damage						
Laporan						
Grafik						

Line Damage						
ID LD	Tanggal	Nama Part	No Part	Jumlah	Jenis Kerusakan	Penyebab

Gambar V.43 *Interface Form Data Lihat Line Damage*
(Sumber: Hasil Analisis, 2018)

11. *Form Data Laporan Produksi*

Form data transaksi adalah form yang digunakan untuk melihat dan mencetak laporan produksi harian. Rancangan interface dari form laporan produksi dapat dilihat pada Gambar V.44 berikut:



Gambar V.44 *Interface* cetak Laporan Hasil Produksi
(Sumber: Hasil Analisis, 2018)

5.5.2. Implementasi Sistem

Untuk dapat menggunakan aplikasi, tentunya diperlukan suatu spesifikasi perangkat keras (*hardware*) ataupun perangkat lunak (*software*) yang mendukung agar aplikasi dapat berjalan dengan baik. Adapun spesifikasinya sebagai berikut:

1. Kebutuhan *Hardware*

- a. *Processor* : Intel Core 2 Duo.
- b. *RAM* : Minimal 2 GB.
- c. *Hard Disk* : 200 GB
- d. *Mouse, Keyboard, Monitor* sebagai peralatan antarmuka

2. Kebutuhan *Software*

- a. *Sistem Operasi* : *Microsoft Windows 7 Ultimate*
- b. *Web Server* : *Apache*
- c. *Database Server* : *MySQL 5.5.32*
- d. *Bahasa Pemrograman* : *PHP*
- e. *Framework* : *Codeigniter 3.1.7*
- f. *Web Browser* : *Mozilla Firefox*

BAB IV

KESIMPULAN DAN SARAN

6.1. Kesimpulan

Berdasarkan hasil pengumpulan, pengolahan dan analisis data yang telah dilakukan dalam penelitian ini dapat disimpulkan bahwa:

1. Dengan membuat sistem informasi pelaporan kerusakan *part* di bagian *Trimming* 3 berbasis *website*, diharapkan dapat membantu mengolah data serta membuat laporan kerusakan *part* pada proses produksi secara cepat dan tepat .
2. Adanya *database* sebagai media penyimpanan sehingga mengurangi resiko kehilangan data.
3. Untuk melakukan pengiriman *line damage* tidak harus mendatangi antar bagian secara langsung karena sistem informasi ini sudah terintegrasi antar bagian.

6.2. Saran

Saran yang diberikan untuk pengembangan sistem informasi produksi ini selanjutnya adalah sebagai berikut:

1. Untuk penerapan sistem yang baru terhadap sistem lama, sebaiknya dilakukan secara bertahap dan diperlukan sosialisasi penggunaannya kepada bagian yang terkait.
2. Untuk penelitian selanjutnya diharapkan dapat membahas pengadaan *part* atau komponen di bagian *Trimming* 3.

DAFTAR PUSTAKA

- Abdul Kadir. 2014. *Pengenalan Sistem Informasi Edisi Revisi*. Yogyakarta: Andi.
- Achmadi, Abu dan Narbuko Cholid. 2016. *Metodologi Penelitian*. Jakarta: Bumi Aksara.
- Ahyari, Agus. 2011. *Perencanaan dan Pengendalian Produksi*. Yogyakarta: BPFE.
- Anhar. 2010. *PHP & MySql Secara Otodidak*. Jakarta: PT TransMedia.
- Arman Hakim & Yudha Prasetyawan. 2008. *Perencanaan & Pengendalian Produksi*. Yogyakarta: Graha Ilmu.
- Assauri, Sofjan. 2011. *Manajemen Produksi dan Operasi*. Jakarta: LPFEUI.
- Azhar Susanto. 2013. *Sistem Informasi Akuntansi*. Bandung: Lingga Jaya.
- Dennis, A. et al. 2010. *System Analysis and Design with UML An Object Oriented Approach*, 3th Edition. Asia: John Wiley.
- Departemen Pendidikan Indonesia (2008). *Kamus Besar Bahasa Indonesia*. Jakarta: Balai Pustaka.
- Elbert, Ronald J. & Ricky W. Griffin. 2006. *Bisnis*. Alih bahasa Rd. Soemarnagara, Jakarta: Erlangga.
- Gaol, L, Jimmy. 2008. *Sistem Informasi Manajemen Pemahaman dan Aplikasi*. Jakarta: PT Grasindo.
- Gullick, Luther. 2004. *Papers on the Science of Administration*. New York: Columbia University
- Hutaeon, Jeperson. 2014. *Konsep Sistem Informasi*. (Edisi 1. Cetakan ke-1) Yogyakarta: Deepublish.
- Ishak. 2010. *Manajemen Produksi dan Operasi*. Yogyakarta: Graha Ilmu.
- Jogianto, H. M. 2005. *Sistem Informasi Strategik Untuk Keunggulan Kompetitif*. Yogyakarta: Andi Offset.

- Makinuddin dan Tri Hadiyanto. 2006. *Analisis Sosial: Bersaksi Dalam Advokasi Irigasi*. Bandung: Yayasan AKATIGA.
- McLeod, Jr., Raymond; Schell, George P. 2011. *Sistem Informasi Manajemen* (Terjemahan). Jakarta: Salemba Empat.
- Nur, Rusdi. 2017. *Pengantar Sistem Manufaktur*. (Edisi 1. Cetak ke-1) Yogyakarta: Deepublish.
- O'Brien, James dan George M. Marakas. 2017. *Pengantar Sistem Informasi*. Jakarta: Salemba Empat.
- Ristono, Agus. 2011. *Pemodelan Sistem*. (Edisi 1. Cetak ke-1) Yogyakarta: Graha Ilmu.
- Rochaety, Eti., dkk. 2013. *Sistem Informasi Manajemen*. Edisi ke 2. Jakarta: Mitra Wacana Media.
- Shalahuddin, M., Rosa A. S. 2013. *Rekayasa Perangkat Lunak Terstruktur Dan Berorientasi Objek*. Bandung: Informatika.
- Shalahuddin, M., Rosa A. S. 2014. *Rekayasa Perangkat Lunak Struktur Dan Berorientasi Objek*. Bandung: Informatika.
- Shalahuddin, M., Rosa A. S. 2015. *Rekayasa Perangkat Lunak Struktur Dan Berorientasi Objek*. Bandung: Informatika.
- Shalahuddin, M., Rosa A. S. 2016. *Rekayasa Perangkat Lunak Struktur Dan Berorientasi Objek*. Bandung: Informatika.
- Saludin. 2016. *Rekayasa Sistem Manufaktur*. Jakarta: Mitra Wacana Media.
- Siagina, S. 2013. *Filsafat Administrasi Edisi Revisi*. Jakarta: PT Bumi Aksara.
- Sinulingga, Sukaria. 2013. *Perencanaan dan Pengendalian Produksi*. (Edisi 1. Cetak ke-2) Yogyakarta: Graha Ilmu.
- Sukoco, Badri. 2007. *Manajemen Administrasi Perkantoran Modern*. Jakarta: Erlangga.
- Sutabri, Tata. 2014. *Analisis Sistem Informasi*. Yogyakarta: CV Andi Offset.
- Sutaji, Deni. (2012). *Sistem Inventory Mini Market dengan PHP & JQUERY* Yogyakarta: Lokomedia.

Septian, Gungun. 2011. *Trik Pintar Menguasai Codeigniter*. Jakarta: PT Elex Media Komputindo.

Situs Internet

Suhdi. 2009. *Teori Dasar Perakitan*. suhdi.wordpress.com/2009/01/31/teori-dasar-perakitan. Diakses 07 Januari 2017.

LAMPIRAN A**TAMPILAN PROGRAM****1. Login****2. Menu Utama****3. Data Master user**

PT KRM

Form Data User

12 records per page

Search

No	Nama Lengkap	Username	Role	Status
1	Admin	admin	Admin (Super User)	Active
2	Admin (Super User)	admin	Admin (Super User)	Active
3	Admin (Super User)	admin	Admin (Super User)	Active

Showing 1 to 3 of 3 entries

Previous 1 2 Next

4. Data Master part

PT KRM

Form Data Part

12 records per page

Search

No	Part	Nama Part	Qty Part	Status
1	Part	Part 1 (Super User)	1000	Active
2	Part	Part 2 (Super User)	1000	Active
3	Part	Part 3 (Super User)	1000	Active
4	Part	Part 4 (Super User)	1000	Active
5	Part	Part 5 (Super User)	1000	Active

Showing 1 to 5 of 5 entries

Previous 1 2 Next

5. Data Master produk

PT KRM

Form Data Produk

12 records per page

Search

No	Produk	Nama Produk	Qty	Status
1	Produk	Produk 1 (Super User)	1000	Active
2	Produk	Produk 2 (Super User)	1000	Active

Showing 1 to 2 of 2 entries

Previous 1 2 Next

6. Data Transaksi Rencana Produksi

PT KRM

Form Rencana Produksi

10 records per page

No	ID Rencana	Tanggal	ID Produk	Total	Tipe Rencana
1	0001	2019-03-15	0001	10	Normal
2	0002	2019-03-15	0002	10	Normal

Showing 1 to 2 of 2 records

Buttons: [Previous] [Next]

7. Lihat Data Rencana Produksi

PT KRM

Menampilkan informasi rencana produksi

No	ID Rencana	Tanggal	Tipe Rencana	Total
1	0001	2019-03-15	Normal	10

LAMPIRAN B

KODE PROGRAM

1. Login

➔ View

```
<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
  <meta charset="utf-8" />
  <meta name="viewport" content="width=device-width, initial-scale=1.0" />
  <title>PT Krama Yudha Ratu Motor</title>
  <!-- BOOTSTRAP STYLES-->
  <link href="<?php echo base_url();?>assets/css/bootstrap.css" rel="stylesheet"
/>
  <!-- FONTAWESOME STYLES-->
  <link href="<?php echo base_url();?>assets/css/font-awesome.css"
rel="stylesheet" />
  <!-- CUSTOM STYLES-->
  <link href="<?php echo base_url();?>assets/css/custom.css" rel="stylesheet" />
  <!-- GOOGLE FONTS-->
  <link href='http://fonts.googleapis.com/css?family=Open+Sans' rel='stylesheet'
type='text/css' />
  <link href="<?php echo base_url();?>assets2/csslogin/style.css" rel="stylesheet"
/>

</head>
<body>
  <div class="container">
    <div class="row text-center ">
      <div class="col-md-12">
        <br /><br />

        <br />
      </div>
    </div>
  </div>
  <div class="row ">
```

```

        <div class="col-md-4 col-md-offset-4 col-sm-6 col-sm-offset-3 col-xs-
10 col-xs-offset-1">
            <div class="panel panel-default">
                <div class="panel-heading">
                    <strong><center> <h2>Silahkan Login</h2></center></strong>
                </div>
                <div class="panel-body">
                    <?php
                        echo form_open('auth/login');
                    ?>
                    <br />
                    <div class="form-group input-group">
                        <span class="input-group-addon"><i class="fa fa-tag"
></i></span>
                            <input type="text" name="username" class="form-
control" placeholder="username " />
                        </div>
                        <div class="form-group input-group">
                            <span class="input-group-addon"><i class="fa fa-
lock" ></i></span>
                                <input type="password" name="password"
class="form-control" placeholder="password" />
                            </div>
                            <button class="btn btn-primary" type="submit"
name="submit">Login</button>
                        </form>
                    </div>
                </div>
            </div>
        </div>
    </div>

```

```

<!-- SCRIPTS -AT THE BOTOM TO REDUCE THE LOAD TIME-->
<!-- JQUERY SCRIPTS -->
<script src="<?php echo base_url();?>assets/js/jquery-1.10.2.js"></script>
<!-- BOOTSTRAP SCRIPTS -->
<script src="<?php echo base_url();?>assets/js/bootstrap.min.js"></script>
<!-- METISMENU SCRIPTS -->
<script src="<?php echo base_url();?>assets/js/jquery.metisMenu.js"></script>
<!-- CUSTOM SCRIPTS -->
<script src="<?php echo base_url();?>assets/js/custom.js"></script>

```

```
</body>
</html>
```

➔ Model

```
<?php
class M_user extends CI_Model{
    function login($username,$password)
    {
        $chek= $this->db-
>get_where('user',array('username'=>$username,'password'=>$password));

        // untuk check data username dan password ada atau tidak
        if ($chek->num_rows()>0)
        {
            return 1;
        }
        else {
            return 0;
        }
    }

    function tampilkan_data(){
        //untuk menampilkan data
        return $this->db->get('user');
    }

    function simpan($data){

        $this->db->insert('user', $data);
    }

    function hapus($id){
        $this->db->where('id', $id);
        $this->db->delete('user');
    }

    function get_data($id){
        $vans = array('id' => $id);
        return $this->db->get_where('user', $vans);
    }

    function update($data, $id){
        $this->db->where('id', $id);
        $this->db->update('user', $data);
    }
}
```

➔ Controller

```
<?php
```

```

class Auth extends CI_controller{

    function __construct(){
        parent::__construct();
        $this->load->model('M_user');
    }

    function login()
    {
        if(isset($_POST['submit'])){
            $username=$this->input->post('username');
            $password=$this->input->post('password');
            $hasil=$this->M_user->login($username,$password);
            $jabatan      =      $this->db-
>get_where('user',array('username'=>$username))->row_array();
            if($hasil == 1){
                $this->session-
>set_userdata(array('status_login'=>'sukses'));
                redirect('Home');
            }else{
                redirect('auth/login');
            }
        } else{
            $this->load->view('f_login');
        }

    }

    function logout(){

        $this->session->sess_destroy();
    }
}

```

```

        redirect('auth/login');
    }
}

```

2. Menu Utama

```

<!DOCTYPE html>
<head>
    <meta charset="utf-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>PT Krama Yudha Ratu Motor</title>
    <!-- BOOTSTRAP STYLES-->
    <link href="<?php echo base_url();?>assets/css/bootstrap.css" rel="stylesheet" />
    <!-- FONTAWESOME STYLES-->
    <link href="<?php echo base_url();?>assets/css/font-awesome.css" rel="stylesheet" />
    <!-- MORRIS CHART STYLES-->

    <!-- CUSTOM STYLES-->
    <link href="<?php echo base_url();?>assets/css/custom.css" rel="stylesheet" />
    <!-- GOOGLE FONTS-->
    <link href='http://fonts.googleapis.com/css?family=Open+Sans' rel='stylesheet' type='text/css' />
    <!-- TABLE STYLES-->
    <link href="<?php echo base_url();?>assets/js/dataTables/dataTables.bootstrap.css" rel="stylesheet" />
</head>

<body>
    <div id="wrapper">
        <nav class="navbar navbar-default navbar-cls-top " role="navigation" style="margin-bottom: 0">
            <div class="navbar-header">
                <button type="button" class="navbar-toggle" data-toggle="collapse" data-target=".sidebar-collapse">
                    <span class="sr-only">Toggle navigation</span>
                    <span class="icon-bar"></span>
                    <span class="icon-bar"></span>
                    <span class="icon-bar"></span>
                </button>
            </div>
        </nav>
    </div>

```

```

        </button>
        <a class="navbar-brand" href="index.html">PT KRM</a>
    </div>
    <div style="color: white;
padding: 15px 50px 5px 50px;
float: right;
font-size: 17px;"> <a href="<?php echo site_url() ?>/auth/logout"><i
class="btn btn-danger square-btn-adjust"></i>Logout</a> </div>
</nav>
<!-- /. NAV TOP -->
    <nav class="navbar-default navbar-side" role="navigation">
    <div class="sidebar-collapse">
        <ul class="nav" id="main-menu">

            <li>
                <a href="http://localhost/produksi1/index.php/Home"><i
class="fa fa-dashboard fa-2x"></i>Menu Utama</a>
            </li>
            <li>
                <a href="#"><i class="fa fa-briefcase fa-2x"></i> Data
Master<span class="fa arrow"></span></a>
                <ul class="nav nav-second-level">
                    <li>
                        <a href="http://localhost/produksi1/index.php/user"><i
class="fa fa-fw fa-users"></i>Data User </a>
                    </li>
                    <li>
                        <a
href="http://localhost/produksi1/index.php/data_part"><i class="fa fa-fw fa-
edit"></i>Data Part</a>
                    </li>
                    <li>
                        <a
href="http://localhost/produksi1/index.php/data_produk"><i class="fa fa-fw
fa-edit"></i>Data Produk</a>
                    </li>
                </ul>
            </li>
            <li>
                <a href="#"><i class="fa fa-laptop fa-2x"></i> Data
Transaksi<span class="fa arrow"></span></a>
                <ul class="nav nav-second-level">

```

```

        <li>
            <a
href="http://localhost/produksi1/index.php/Rencana_Produksi"><i class="fa
fa-file fa-2x"></i> Rencana Produksi</a>
        </li>
        <li>
            <a
href="http://localhost/produksi1/index.php/data_line_damage"><i class="fa
fa-file fa-2x"></i> Line Damage</a>
        </li>
        <li>
            <a
href="http://localhost/produksi1/index.php/data_line_damage"><i class="fa
fa-file fa-2x"></i> Hasil Produksi</a>
        </li>
    </ul>
</li>
</li>
    <li>
        <a
href="http://localhost/produksi1/index.php/tampilan"><i class="fa fa-
calendar fa-2x"></i>Rencana Produksi </a>
    </li>
    <li>
        <a
href="http://localhost/produksi1/index.php/pelamar"><i class="fa fa-file fa-
2x"></i>Line Damage </a>
    </li>
    <li>
        <a
href="http://localhost/produksi1/index.php/pelamar"><i class="fa fa-file fa-
2x"></i>Laporan </a>
    </li>
    <li>
        <a
href="http://localhost/produksi1/index.php/control_grafik"><i class="fa fa-
bar-chart-o"></i>Grafik</a>
    </li>
</ul>
</li>
</div>

</nav>

```

```

<!-- /. NAV SIDE -->
<div id="page-wrapper" >
    <div id="page-inner">

        <!-- /. ROW -->
        <hr />
        <?php echo $contents; ?>

    </div>

</div>

<!-- /. PAGE INNER -->
</div>
<!-- /. PAGE WRAPPER -->

<!-- /. WRAPPER -->
<!-- SCRIPTS -AT THE BOTTOM TO REDUCE THE LOAD TIME-->
<!-- JQUERY SCRIPTS -->
<script src="<?php echo base_url();?>assets/js/jquery-1.10.2.js"></script>
<!-- BOOTSTRAP SCRIPTS -->
<script src="<?php echo base_url();?>assets/js/bootstrap.min.js"></script>
<!-- METISMENU SCRIPTS -->
<script src="<?php echo base_url();?>assets/js/jquery.metisMenu.js"></script>
<!-- DATA TABLE SCRIPTS -->
<script src="<?php echo base_url();?>assets/js/dataTables/jquery.dataTables.js"></script>
<script src="<?php echo base_url();?>assets/js/dataTables/dataTables.bootstrap.js"></script>
<script>
    $(document).ready(function () {
        $('#dataTables-example').dataTable();
    });
</script>
<!-- CUSTOM SCRIPTS -->
<script src="<?php echo base_url();?>assets/js/custom.js"></script>

</body>
</html>

```