Alo. Dok 7109

D3
6872
Sim
p

RANCANG BANGUN SISTEM INFORMASI PENGADAAN BAHAN *MAINTENANCE, REPAIR,* DAN *OPERATION* (MRO) PADA DIVISI *MAINTENANCE* MENGGUNAKAN LARAVEL 5.4.28 DAN MARIADB 10.1.31 PADA PT CIKARANG PERKASA MANUFACTURING

SUMBANGAN ALUMNI

TUGAS AKHIR

Untuk Memenuhi Salah Satu Syarat Program Sarjana Terapan

Program Studi Sistem Informasi Industri Otomotif pada Politeknik STMI Jakarta

OLEH

Gabriella Victoria Simarmata

1314069

DATA BUKU PERPUSTAKAAN
Tgl Terima | 27/10/22
No Induk Buku | 1651/SIM/55/10/22

POLITEKNIK STMI JAKARTA

KEMENTERIAN PERINDUSTRIAN REPUBLIK INDONESIA

JAKARTA

2019

Dipindai dengan CamScanner

TANDA PERSETUJUAN DOSEN PEMBIMBING

JUDUL TUGAS AKHIR

RANCANG BANGUN SISTEM INFORMASI PENGADAAN BAHAN *MAINTENANCE*, *REPAIR* DAN *OPERATION* (MRO) PADA DIVISI *MAINTENANCE* MENGGUNAKAN LARAVEL 5.2.31 DAN MARIADB 10.1.31 PADA PT. CIKARANG PERKASA MANUFACTURING

DISUSUN OLEH

NAMA : GABRIELLA V. SIMARMATA (1314069)

PROGRAM STUDI : SISTEM INFORMASI INDUSTRI OTOMOTIF

Telah diperiksa dan disetujui untuk memenuhi salah satu syarat program Sarjana Terapan Program Studi Sistem Informasi Industri Otomotif Politeknik STMI Jakarta

Jakarta, 06 Agustus 2019



Noveriza Yuliasari, S.Si, M.T.

NIP. 197811212009012003

Dipindai dengan CamScanner

POLITEKNIK STMI KEMENTERIAN PERINDUSTRIAN REPUBLIK INDONESIA

TANDA PERSETUJUAN DOSEN PEMBIMBING

JUDUL TUGAS AKHIR :

RANCANG BANGUN SISTEM INFORMASI PENGADAAN BAHAN *MAINTENANCE*, *REPAIR*, DAN *OPERATION* (MRO) PADA DIVISI *MAINTENANCE* MENGGUNAKAN LARAVEL 5.4.28 DAN MARIA DB 10.1.31 PADA PT CIKARANG PERKASA MANUFACTURING

Disusun Oleh:

Nama	: Gabriella Victoria Simarmata
Nim	: 1314069
Program Studi	: Sistem Informasi Industri Otomotif
Tanggal Seminar I	: 14 Agustus 2019
Tanggal Seminar II	: 05 September 2019
Tanggal Sidang	: 20 September 2019
Tanggal Lulus	: 20 September 2019

Telah diperiksa dan disetujui untuk diajukan dan dipertahankan dalam Ujian Tugas Akhir Politeknik STMI Jakarta Kementrian Perindustrian R.I.

Jakarta, 23 September 2019

Dosen Pembimbing,

a.n

Noveriza Yuliasari, S.Si.M.T.
NIP. 19781121 200901 2003

Dipindai dengan CamScanner

POLITEKNIK STMI JAKARTA
KEMENTERIAN PERINDUSTRIAN RI
LEMBAR PENGESAHAN

UDUL TUGAS AKHIR:

RANCANG BANGUN SISTEM INFORMASI PENGADAAN BAHAN *MAINTENANCE*, *REPAIR*, DAN *OPERATION* (MRO) PADA DIVISI *MAINTENANCE* MENGGUNAKAN LARAVEL 5.4.28 DAN MARIADB 10.1.31 PADA PT CIKARANG PERKASA MANUFACTURING

isusun Oleh:

ama : Gabriella Victoria Simarmata

im : 1314069

ogram Studi : Sistem Informasi Industri Otomotif

lah diuji oleh Tim Penguji Sidang Tugas Akhir Program Studi Sistem Informasi dustri Otomotif Politeknik STMI Jakarta Kementrian Perindustrian R.I. pada i Jumat, 20 September 2019.

Jakarta, 23 September 2019

Dosen Pembimbing

a.n

Noveriza Yuliasari, S.Si, M.T.
NIP.19781121 200901 2003

Ketua Penguji

Lucky Heriyanto, M.TI
NIP.19790820 200502 1002

Dosen Penguji

ifi Lailasari H, S.Kom, M.Kes.
NIP.19731016 200502 2001

Dosen Penguji

Drs. Jacob Saragih, M.M.
NIP. 19540428 198603 1002

Dipindai dengan CamScanner

Kementerian Perindustrian
REPUBLIK INDONESIA

BADAN PENGEMBANGAN SUMBER DAYA MANUSIA INDUSTRI
POLITEKNIK STMI JAKARTA
Jl. Letjen Suprapto No. 26 Cempaka Putih, Jakarta 10510
Telp: (021) 42886064 Fax: (021) 42888206
www.stmi.ac.id

LEMBAR BIMBINGAN TUGAS AKHIR

Nama : Gabriella Victoria
NIM : 1314069
Judul Tugas Akhir : Rancang Bangun Sistem Informasi Pengadaan Bahan *Maintenance, Repair,* dan *Operation* (MRO) pada Divisi *Maintenance* Menggunakan Laravel 5.4.28 dan MariaDB 10.1.31 pada PT. Cikarang Perkasa Manufacturing
Pembimbing : Noveriza Yuliasari, S.Si, M.T.

Tanggal	Keterangan	Paraf
02/04/2019	Pertemuan awal, penyerahan Bab I, II	
10/04/2019	Revisi Bab I, II, penyerahan Bab III	
23/04/2019	Revisi Bab I, II, III	
04/05/2019	Revisi Bab I, II, III, penyerahan Bab IV	
14/05/2019	Penyerahan Bab V	
27/06/2019	Revisi Bab V	
19/07/2019	Revisi Bab V	
23/07/2019	Revisi Bab V	
25/07/2019	Revisi Bab V	
31/07/2019	Revisi Bab V	
01/08/2019	Revisi Bab V	

Mengetahui,
Ketua Program Studi
Sistem Informasi Industri Otomotif

Noveriza Yuliasari, S.Si, M.T
NIP. 197811212009012003

Pembimbing

Noveriza Yuliasari, S.Si, M.T
NIP. 197811212009012003

Dipindai dengan CamScanner

LEMBAR PERNYATAAN KEASLIAN

Saya yang bertanda tangan di bawah ini:

Nama : Gabriella Victoria Simarmata

NIM : 1314069

Berstatus sebagai mahasiswa Program Studi Sistem Informasi Industri Otomotif di Politeknik STMI Jakarta Kementerian Perindustrian RI. Dengan ini menyatakan bahwa hasil karya Tugas Akhir yang saya buat dengan judul:

"RANCANG BANGUN SISTEM INFORMASI PENGADAAN BAHAN MAINTENANCE, REPAIR, DAN OPERATION (MRO) PADA DIVISI MAINTENANCE MENGGUNAKAN LARAVEL 5.4.28 DAN MARIADB 10.1.31 PADA PT CIKARANG PERKASA MANUFACTURING"

- **Dibuat** dan diselesaikan sendiri, dengan menggunakan literatur hasil kuliah, *survey* lapangan, dibantu oleh dosen pembimbing, serta buku-buku maupun jurnal-jurnal ilmiah yang menjadi bahan acuan yang tertera dalam referensi karya Tugas Akhir ini.
- **Bukan** merupakan hasil duplikasi karya tulis yang sudah dipublikasikan atau yang pernah dipakai sebelumnya untuk mendapatkan gelar sarjana di Universitas/Perguruan Tinggi lain, kecuali yang telah disebutkan sumbernya dan dicantumkan pada referensi karya Tugas Akhir ini.
- **Bukan** merupakan karya tulis hasil terjemahan dari kumpulan buku atau jurnal acuan yang tertera dalam referensi karya Tugas Akhir ini.

Jika terbukti saya tidak memenuhi apa yang telah saya nyatakan di atas, maka saya bersedia menerima sanksi atas apa yang telah saya lakukan sesuai dengan peraturan yang berlaku.

Jakarta, 06 Agustus 2019



Gabriella Victoria Simarmata

Dipindai dengan CamScanner

ABSTRAK

PT Cikarang Perkasa Manufacturing merupakan perusahaan manufaktur komponen otomotif yang mempunyai peran memenuhi kebutuhan konsumen akan *spare-part* motor dan mobil dengan bahan baku *Aluminium Ingot ADC 12*. Dalam mengerjakan aktivitas pemeliharaan mesin-mesin produksi yang ada *Maintenance Staff* akan menghubungi *warehouse* untuk mengetahui jumlah *stock* yang ada, dan *warehouse* akan melakukan pengecekan secara langsung karena tidak terdokumentasinya bahan MRO yang masuk dan keluar. Penggunaan Ms. Excel saat proses pemasukan data sehingga faktor *human error* tidak dapat dihindari pada. Pencarian data di tumpukan file untuk memastikan jumlah bahan MRO yang datang sesuai dengan jumlah bahan MRO yang dipesan pada *purchase order*. Untuk mengatasi masalah tersebut dibutuhkan pengembangan sistem informasi yang dapat menangani masalah pengolahan data pengadaan bahan MRO. Metode pengembangan sistem yang digunakan pada penelitian ini adalah metode *waterfall*. Dalam merancang sistem informasi pengadaan bahan MRO menggunakan *Unified Modelling Language* (UML). Pemodelan data dan implementasi menggunakan *Class Diagram*, Kamus Data, dan *Deployment Diagram*. Desain sistem menggunakan *Windows Navigation Diagram* (WND). Sistem informasi pengadaan bahan MRO ini menggunakan *framework* Laravel 5.4.28 dan Maria DB 10.1.31 sebagai *database*. Sistem Pengadaan Bahan MRO ini akan memudahkan *user* dalam pengajuan perencanaan pembelian bahan MRO, pembuatan *purchase request*, *purchase order*, dan distribusi data antar divisi. Disarankan untuk menggunakan sistem informasi pengadaan bahan MRO ini untuk mendukung proses pengadaan bahan MRO di PT Cikarang Perkasa Manufacturing.

Kata kunci: Sistem Informasi, Pengadaan Bahan MRO, Metode *Waterfall*, *Unified Modelling Language* (UML), Laravel, MariaDB.

Dipindai dengan CamScanner

KATA PENGANTAR

Puji syukur serta terima kasih kepada Tuhan Yang Maha Esa atas berkat dan rahmat-Nya penulis dapat menyelesaikan Tugas Akhir dengan judul **"RANCANG BANGUN SISTEM INFORMASI PENGADAAN BAHAN *MAINTENANCE*, *REPAIR* DAN *OPERATION* (MRO)PADA DIVISI *MAINTENANCE*MENGGUNAKAN LARAVEL 5.2.31 DAN MARIADB 10.1.31 PADA PT. CIKARANG PERKASA MANUFACTURING"**. Tugas Akhir ini merupakan salah satu syarat untuk menyelesaikan pendidikan program Sarjana Terapan jurusan Sistem Informasi Industri Otomotif pada Politeknik STMI Jakarta.

Penulis menyadari bahwa Tugas Akhir ini tidak terwujud tanpa adanya bantuan, bimbingan, dukungan, doa, dan kasih sayang yang tulus dari berbagai pihak. Penulis juga ingin menyampaikan rasa terima kasih yang sebesar-besarnya kepada semua pihak yang telah membantu dalam penyelesaian laporan ini. Ucapan terima kasih penulis sampaikan terutama kepada:

1. Papa, Mama, Michael, Philip yang selalu mendukung dari awal dan senantiasa memberikan kasih sayang, dan doa untuk keberhasilan Penulis.
2. Bapak DR. Mustofa, ST. MT selaku Ketua Politeknik STMI Jakarta.
3. Bapak Drs. Jacob Saragih, MM, selaku Ketua Jurusan Sistem Informasi Industri Otomotif.
4. Ibu Noveriza Yuliasari, S.Si, M.T selaku Dosen Pembimbing. Terima kasih untuk bantuan, dorongan, bimbingan, motivasi, dan waktu yang telah diluangkan untuk saya dalam proses penulisan laporan PKL ini.
5. Bapak Chiko bagian *Purchasing*, selaku pembimbing saya di PT. Cikarang Perkasa Manufacturing. Terima kasih telah memberikan saya arahan, bimbingan, dan banyak pelajaran yang tidak saya dapatkan ketika masa kuliah.
6. Seluruh Dosen Politeknik STMI Jakarta yang telah memberikan ilmunya guna menambah pengetahuan dan pengalaman yang sangat berharga.

Dipindai dengan CamScanner

7. Teman-teman mahasiswa/i Politeknik STMI Jakarta Program Studi Sistem Informasi Industri Otomotif terutama SA02 dan PMK atas kebersamaan dan motivasinya selama ini.
8. Melani, teman seperjuangan pencarian tempat PKL dan Safira yang telah banyak membantu dan memberi masukan untuk kelancaran PKL.
9. Semua pihak yang turut mendukung dan mendoakan kelancaran Tugas Akhir dan pembuatan Tugas Akhir yang tidak dapat Penulis sebutkan satu persatu.

Semoga Tuhan membalas kebaikan dan ketulusan semua pihak yang telah membantu menyelesaikan laporan ini dengan melimpahkan kasih karunia-Nya. Penulis menyadari bahwa penulisan Tugas Akhir ini masih jauh dari kesempurnaan, untuk itu Penulis mohon kritik dan saran yang bersifat membangun demi kesempurnaan penulisan di masa mendatang. Besar harapan Penulis bahwa laporan ini dapat memberikan manfaat bagi kita semua.

Jakarta, 06 Agustus 2019

Penulis

Dipindai dengan CamScanner

DAFTAR ISI

Halaman

Halaman Judul i
Lembar Persetujuan Dosen Pembimbing ii
Lembar Bimbingan Tugas Akhir iii
Lembar Pernyataan Keaslian iv
Abstrak v
Kata Pengantar vi
Daftar Isi viii
Daftar Gambar xii
Daftar Tabel xv
BAB I PENDAHULUAN 1
1.1. Latar Belakang 1
1.2. Pokok Permasalahan 2
1.3. Tujuan Penelitian 3
1.4. Batasan Masalah 3
1.5. Manfaat Tugas Akhir 4
1.6. Sistematika Penulisan 4
BAB II LANDASAN TEORI 7
2.1. Konsep Dasar Sistem Informasi 7
2.1.1. Pengertian Sistem 7
2.1.2. Pengertian Informasi 11
2.1.3. Pengertian Sistem Informasi 12
2.2. Pengertian *Procurement* 15
2.2.1. Proses *Procurement* 15
2.2.2. Kategori *Aktivitas Procurement* 16
2.2.3. Prinsip *Procurement* 17
2.3. Pengertian Pemeliharaan (*Maintenance*) 17
2.4. Jenis Pemeliharaan 18

Dipindai dengan CamScanner

2.4. Pemeliharaan Peralatan Produksi *Mold/Dies* 19
2.5. *Systems Development Life Cycle* (SDLC) 21
2.6. *Waterfall Development* 22
2.7. *System Requirements* 23
2.7.1. *Functional Requirement* 24
2.7.2. *Non Functional Requirement* 24
2.8. *Flowmap* 25
2.9. *Unified Modelling Language* (UML) 26
2.9.1. *Use Case Diagram* 28
2.9.2. *Activity Diagram* 30
2.9.3. *Sequence Diagram* 31
2.9.4. *Class Diagram* 33
2.9.5. *Deployment Diagram* 34
2.10. *Database* 35
2.11. *Entity Relationship Diagram* (ERD) 35
2.12. *Physical Data Model* 37
2.13. Kamus Data 39
2.14. *Window Navigation Diagram* (WND) 39
2.15. PHP 40
2.16. *Framework* Laravel 41
2.17. Sublime Text 44
2.18. *Composer* 44
2.19. MariaDB 44
2.20. *Black Box Testing* 45
BAB III METODOLOGI PENELITIAN 47
3.1. Metodologi Penelitian 47
3.2. Jenis dan Sumber Data 47
3.3. Metode Pengumpulan Data 48
3.4. Kerangka Penelitian 49
BAB IV PENGUMPULAN DAN PENGOLAHAN DATA 54
4.1. Struktur Organisasi Perusahaan 54

Dipindai dengan CamScanner

4.1.1. Tugas dan Wewenang Divisi 55
4.2. Dokumen Pengadaan Bahan MRO 57
4.2.1. Dokumen Masukan 58
4.2.2. Dokumen Keluaran 63
4.3. Analisis Proses Pengadaan Bahan MRO yang Sedang Berjalan .66
4.3.1. *Flowmap* Proses Pengadaan Bahan MRO yang Sedang Berjalan 67
BAB V ANALISIS DAN PEMBAHASAN 69
5.1. Analisis Kebutuhan Sistem 69
5.1.1. *Functional Requirement* 70
5.1.2. *Non-Functional Requirement* 71
5.2. Prosedur Sistem Informasi Usulan Pengadaan Bahan MRO 71
5.3. Analisis Sistem Usulan 73
5.3.1. *Use Case Diagram* 73
5.3.2. *Activity Diagram* 79
5.3.3. *Sequence Diagram* 89
5.3.4. *Class Diagram* 97
5.3.5. *Deployment Diagram* 98
5.4. Pemodelan Data Sistem Usulan 98
5.4.1. *Entity Relationship Diagram* (ERD) 98
5.4.2. *Physical Data Model* 99
5.4.3. Kamus Data 100
5.5. Perancangan Sistem 104
5.5.1. Windows Navigation Diagram 104
5.5.2. Perancangan Antarmuka 105
5.6. Implementasi Sistem *Software* dan *Hardware* 112
5.7. Pengujian Sistem 113
BAB VI KESIMPULAN DAN SARAN 114
6.1. Kesimpulan 114
6.2. Saran 114
DAFTAR PUSTAKA 115

Dipindai dengan CamScanner

DAFTAR LAMPIRAN **L-1**

LAMPIRAN A L-1

LAMPIRAN B L-8

Dipindai dengan CamScanner

DAFTAR GAMBAR

Gambar II.1. Karakteristik Sistem 10
Gambar II.2. Siklus Informasi 12
Gambar II.3. Komponen Sistem Informasi 15
Gambar II.4. Pengembangan dengan Metode *Waterfall* 23
Gambar II.5. *Use Case Diagram* 28
Gambar II. 6. Kardinalitas *One to One* 37
Gambar II. 7. Kardinalitas *One to Many* 37
Gambar II. 8. Kardinalitas *Many to Many* 37
Gambar II.9. *Windows Navigation Diagram* 40
Gambar II.10. Konsep MVC 42
Gambar III.1. *Flowchart* Kerangka Penelitian 53
Gambar IV.1. Struktur Organisasi PT Cikarang Perkasa Manufacturing 55
Gambar IV.2. *Request Form Planning* 58
Gambar IV.3. *Purchase Requisition* 60
Gambar IV.4. *Purchase Order* 62
Gambar IV.5. *Purchase Order Report* 64
Gambar IV.6. *Warehouse Report* 65
Gambar IV.7. Aliran Analisis Proses Pengadaan Bahan-Bahan MRO 68
Gambar V.1. *Flowmap* Sistem Usulan Pengadaan Bahan MRO di PT CPM 73
Gambar V.2. *Use Case Diagram* Sistem Usulan 74
Gambar V.3. *Activity Diagram Login* ke Sistem 80
Gambar V.4. *Activity Diagram* Mengelola Data Master Bahan MRO 81
Gambar V.5. *Activity Diagram* Mengelola Data Master *Supplier* 82
Gambar V.6. *Activity Diagram* Membuat *Request Form Planning* 83
Gambar V.7. *Activity Diagram* Menyetujui *Request Form Planning* 84
Gambar V.8. *Activity Diagram* Membuat *Purchase Requisition* 85
Gambar V.9. *Activity Diagram* Menyetujui *Purchase Requisition* 85
Gambar V.10. *Activity Diagram* Membuat *Purchase Order* 86

Dipindai dengan CamScanner

Gambar V.11. *Activity Diagram* Mencetak Laporan *Purchase Requisition*.....87
Gambar V.12. *Activity Diagram* Mencetak *Purchase Order* 87
Gambar V.13. *Activity Diagram* Mencetak Laporan *Purchase Order* 88
Gambar V.14. *Activity Diagram* Mengelola Data *Warehouse Stock*.....89
Gambar V.15. *Sequence Diagram* Proses *Login* 90
Gambar V.16. *Sequence Diagram* Mengelola Data Master Bahan MRO.....90
Gambar V.17. *Sequence Diagram* Mengelola Data Master *Supplier*.....91
Gambar V.18. *Sequence Diagram* Membuat *Request Form Planning*.....92
Gambar V.19. *Sequence Diagram* Menyetujui *Request Form Planning*.....92
Gambar V.20. *Sequence Diagram* Membuat *Purchase Requisition*.....93
Gambar V.21. *Sequence Diagram* Menyetujui *Purchase Requisition* 93
Gambar V.22. *Sequence Diagram* Membuat *Purchase Order*.....94
Gambar V.23. *Sequence Diagram* Mencetak *Purchase Order*.....94
Gambar V.24. *Sequence Diagram* Mencetak Laporan *Purchase Order* 95
Gambar V.25. *Sequence Diagram* Mencetak Laporan *Purchase Requisition*.....95
Gambar V.26. *Sequence Diagram* Mengelola *Warehouse Stock* 96
Gambar V.27. *Class Diagram* Sistem Usulan Pengadaan Bahan MRO.....97
Gambar V.28. *Deployment Diagram* Sistem Usulan.....98
Gambar V.29. ERD Sistem Usulan Pengadaan Bahan MRO.....99
Gambar V.30. *Physical Data Model* Sistem Usulan Pengadaan Bahan MRO....100
Gambar V.31. WND Sistem Usulan Pengadaan Bahan MRO 105
Gambar V.32. Tampilan *Form Login* 106
Gambar V.33. Tampilan *Form Data Master* Bahan MRO.....106
Gambar V.34. Tampilan *Form Edit Data Master* Bahan MRO 107
Gambar V.35. Tampilan *New Data* Bahan MRO *Form* 107
Gambar V.36. Tampilan *Form Data Master Supplier*.....108
Gambar V.37. Tampilan *Data Supplier Edit Form* 108
Gambar V.38. Tampilan *Request Form Planning* 109
Gambar V.39. Tampilan *New Request Form Planning*.....109
Gambar V.40. Tampilan *Purchase Requisition*.....110
Gambar V.41. Tampilan *Warehouse* 110

Dipindai dengan CamScanner

Gambar V.42. Tampilan *Warehouse Edit Form* 111
Gambar V.43. Tampilan *Purchase Requisition Form* 111
Gambar V.44. Tampilan *Purchase Order Form* 112

Dipindai dengan CamScanner

DAFTAR TABEL

Tabel II.1. Simbol-Simbol *Flowmap* 25
Tabel II.2. Simbol-Simbol *Use Case Diagram* 29
Tabel II.3. Simbol-simbol *Activity Diagram* 30
Tabel II.4. Simbol-simbol *Sequence Diagram* 32
Tabel II.5. *Class Diagram Multiply* 33
Tabel II.6. Simbol-Simbol *Class Diagram* 33
Tabel II.7. Simbol-Simbol *Deployment Diagram* 34
Tabel II.8. Simbol-Simbol ERD 36
Tabel II.9. Simbol-Simbol pada *Physical Data Model* 38
Tabel II.10. Jenis Data pada MariaDB 45
Tabel V.1. Analisis Kebutuhan Sistem 75
Tabel V.2. Definisi Aktor Sistem Usulan 81
Tabel V.3. Definisi *Use Case* 81
Tabel V.4. Skenario *Use Case Login* 82
Tabel V.5. Skenario *Use Case* Mengelola Data Master Bahan MRO 82
Tabel V.6. Skenario *Use Case* Membuat *Request Form Planning* 82
Tabel V.7. Skenario *Use Case* Menyetujui *Request Form Planning* 83
Tabel V.8. *Use Case* Membuat *Purchase Requisition* 83
Tabel V.9. *Use Case* Menyetujui *Purchase Requisition* 83
Tabel V.10. Skenario *Use Case* Membuat *Purchase Order* 84
Tabel V.11. Skenario *Use Case* Mencetak PO 84
Tabel V.12. Skenario *Use Case* Membuat PR *Report* 84
Tabel V.13. Skenario *Use Case* Membuat PO *Report* 85
Tabel V.14. Skenario *Use Case* Mengelola Data Master *Supplier* 85
Tabel V.15. Struktur Data Tabel *Users* 107
Tabel V.16. Struktur Data Tabel Bahan MRO 107
Tabel V.17. Struktur Data Tabel *Supplier* 107
Tabel V.18. Struktur Data Tabel RFP 108

Dipindai dengan CamScanner

Tabel V.19. Struktur Data Tabel PR 108
Tabel V.20. Struktur Data Tabel Detail PR 109
Tabel V.21. Struktur Data Tabel PO 109
Tabel V.23. Struktur Data Tabel *warehouse* 110

Dipindai dengan CamScanner

MILIK PERPUSTAKAAN STMI
Membaca : Ibadah, Mengambil : Dosa

BAB I
PENDAHULUAN

1.1. Latar Belakang

Dalam kehidupan sekarang ini apalagi era persaingan bisnis yang sangat ketat yang disebabkan oleh globalisasi, suatu pembekalan informasi yang cepat, akurat, dan dapat dipercaya merupakan tuntutan utama. Apabila suatu perusahaan ingin mengadakan perubahan untuk meningkatkan kinerja kerjanya maka tuntutan utamanya adalah memberikan informasi yang akurat, dan cepat dengan memperbaiki sistem informasi yang telah ada.

Peranan sistem informasi juga telah mengotomatisasi proses pengolahan dan penyampaian informasi yang terjadi pada perusahaan bidang manufaktur otomotif, terutama pada proses pengadaan bahan-bahan *maintenance*, *repair*, dan *operation* (MRO). Kecepatan pengolahan dan penyampaian informasi yang terjalin antara departemen terkait pengadaan bahan MRO memudahkan proses pemeliharaan mesin-mesin produksi sehingga meningkatkan keunggulan infrastruktur mesin produksi. Kesalahan atau gangguan yang terjadi pada proses pemeliharaan mesin produksi tentu saja akan mengakibatkan kerusakan mesin-mesin produksi baik dalam waktu dekat ataupun waktu lama. Resiko kerusakan ini dapat menghambat proses produksi tentunya.

PT Cikarang Perkasa Manufacturing (CPM) yang bergerak di bidang manufaktur otomotif mempunyai peran untuk memenuhi kebutuhan konsumen dalam produksi *spare-part* motor dan mobil dengan bahan baku *Aluminium Ingot ADC 12*. Produk yang dihasilkan diantaranya adalah *cover machine, body and cover, car wiper holder, manifold, bracket,* danlampu penerangan jalan. Kemampuan produksi PT. Cikarang Perkasa Manufacturing yang baik diimbangi dengan konsistennya Departemen *Maintenance* dalam mengerjakan aktivitas pemeliharaan mesin-mesin produksi yang ada. Hal ini menjadikan Departemen *Maintenance* memegang peranan penting dalam proses produksi.

Dipindai dengan CamScanner

Pengolahan informasi yang digunakan pada PT Cikarang Perkasa Manufacturing saat ini menggunakan Ms. Excel mulai dari proses *input* data hingga *output* dalam bentuk laporan. Tidak adanya dokumentasi bahan MRO yang masuk dan keluar sehingga *warehouse* melakukan pengecekan *stock* secara langsung setiap saat *Maintenance Staff* bertanya pada *warehouse* jumlah bahan MRO yang tersisa untuk mengajukan *Request Form Planning* (RFP) apabila bahan MRO di *warehouse* kurang dari *limit stock. Purchase Request* yang diajukan *Maintenance* yang telah disetujui *Finance*, akan dibuatkan *Purchase Order* oleh *Purchasing* menggunakan Ms. Excel sehingga mengakibatkan ketidaksesuaian *Purchase Order* dan *Purchase Request* karena memungkinkan kesalahan memasukkan data. Bertambahnya waktu yang dibutuhkan untuk membuat *Purchase Order* karena *Purchasing* mencari *Purchase Request* pada tumpukan file terlebih dahulu.

Dalam membantu masalah yang dihadapi PT CPM dapat diusulkan untuk merancang sistem informasi pengadaan bahan MRO yang mencakup kegiatan-kegiatan yang menunjang proses pengadaan bahan MRO sehingga distribusi informasi dan penyimpanan data dapat terekam dalam sistem. Untuk itu, maka penulis menyusun Tugas Akhir dengan judul "Rancang Bangun Sistem Informasi Pengadaan Bahan-Bahan *Maintenance*, *Repair*, dan *Operation* (MRO) Pada PT Cikarang Perkasa Manufacturing Berbasis Web Menggunakan *Framework* Laravel dan MariaDB 10.1.31".

1.2. Pokok Permasalahan

Permasalahan yang terjadi pada PT. CPM adalah sebagai berikut:

1. Data *Purchase Order* yang tidak sesuai dengan *Purchase Request* disebabkan oleh kesalahan *Purchasing* dalam memasukkan data *Purchase Request*.
2. Tidak adanya dokumentasi data bahan MRO yang masuk dan keluar sehingga mengakibatkan *warehouse* melakukan pengecekan secara langsung setiap kali dibutuhkan pengambilan keputusan pengajuan RFP oleh *Maintenance Staff*.
3. Pencarian dokumen *Purchase Request* pada tumpukan file mengakibatkan bertambahnya waktu pembuatan *Purchase Order*.

Dipindai dengan CamScanner

1.3. Tujuan Penelitian

Adapun tujuan dari penelitian Tugas Akhir ini untuk menganalisis, merancang, dan membangun sistem informasi pengadaan bahan MRO yang mampu:

1. Memberikan data *Purchase Order* yang akurat dengan data *Purchase Request*.
2. Menyediakan sistem yang terkomputerasi dengan menggunakan *database* sebagai media penyimpanan agar bahan MRO yang masuk dan keluar dapat terdokumentasi dan distribusi informasi antara *warehouse*, *maintenance*, dan *purchasing* berjalan lancar.

1.4. Batasan Masalah

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

1. Penelitian dilakukan pada Departemen *Purchasing* PT Cikarang Perkasa Manufacturing selama 3 bulan mulai dari 15 Oktober 2018 sampai dengan 15 Januari 2019.
2. Penelitian ini hanya mengenai analisis sistem informasi pengadaan bahan-bahan *maintenance*, *repair*, dan *operation* di PT Cikarang Perkasa Manufacturing dari mulai pengecekan *stock* bahan-bahan *maintenance*, *repair*, dan *operation*, pembuatan RFP, pembuatan *Purchase Request*, pembuatan *Purchase Order*, dan proses penerimaan bahan-bahan oleh *warehouse*.
3. Penelitian ini tidak mengamati proses pembayaran dan penawaran bahan-bahan *maintenance*, *repair*, dan *operation* seperti *invoice*, faktur, dan *quotation*.

Dipindai dengan CamScanner

1.5. Manfaat Tugas Akhir

Adapun manfaat yang dapat diambil dari penelitian pada Tugas Akhir ini adalah sebagai berikut:

1. Bagi Perusahaan

   Hasil penelitian ini dapat diimplementasikan pada perusahaan guna membantu perusahaan untuk meningkatkan efisiensi dan efektivitas proses pengadaan bahan-bahan MRO.

2. Bagi Mahasiswa
   a. Memberikan kemampuan dalam mengaplikasikan teori secara jelas terhadap masalah yang dihadapi.
   b. Memberikan pembelajaran untuk menerapkan ilmu yang diperoleh selama perkuliahan serta menambah wawasan dan pengetahuan penulis, khususnya dalam analisis dan perancangan sistem informasi.

3. Bagi Pihak Lain

   Hasil penelitian ini diharapkan dapat memberikan sumbangan pemikiran dan sebagai referensi bagi peneliti lain yang melakukan penelitian serupa.

1.6. Sistematika Penulisan

Tugas Akhir ini disusun secara sistematis guna memberikan gambaran yang jelas mengenai isi dan pembahasan yang ada didalamnya. Adapun sistematika penulisan Tugas Akhir ini diuraikan dalam enam bab yaitu:

BAB I PENDAHULUAN

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

BAB II LANDASAN TEORI

Pada bab ini membahas tentang berbagai teori yang diperoleh dari buku-buku literatur, jurnal, *electronic book (e-book)* ataupun berbagai macam referensi yang berkaitan dengan tema yang diambil. Teori-teori yang dipaparkan pada laporan ini adalah

Dipindai dengan CamScanner

seputar pengertian rancang bangun, pengertian sistem informasi, pengertian *procurement*, pengertian pemeliharaan, jenis pemeliharaan, pemeliharaan peralatan *mold/dies*, *System Development Life Cycle*(SDLC), *waterfall development*, *system requirements*, *flowmap*, *Unified Modeling Language* (UML), *Windows Navigation Diagram* (WND), PHP, *Framework*, MVC, Laravel, Sublime Text, *Composer*, MariaDB,dan *Black box testing*.

BAB III METODOLOGI PENELITIAN

Pada bab ini menjelaskan metode ilmiah dalam mencari dan menjelaskan kerangka pemecahan masalah yang menguraikan tahap-tahap untuk mengatasi permasalahan yang telah diidentifikasi.

BAB IV PENGUMPULAN DAN PENGOLAHAN DATA

Pada bab ini menguraikan hasil pengumpulan dan pengolahan data dalam penelitian yang dilakukan, berupa tentang PT. Cikarang Perkasa Manufacturing, alur sistem pengadaan bahan-bahan MRO, dokumen-dokumen yang terlibat dalam pengadaan bahan-bahan MRO, serta laporan yang ada dalam sistem pengadaan bahan-bahan MRO.

BAB V ANALISIS DAN PEMBAHASAN

Bab ini berisi tentang uraian tahap pengembangan sistem, yakni mulai dari tahapan analisis dan perancangan berorientasi objek dengan *tools Unified Modelling Language* (UML) yaitu *Use Case Diagram, Activity Diagram, Sequence Diagram, Class Diagram,* Kamus Data, *Deployment Diagram, Windows Navigation Diagram* (WND), *design interface*, dan spesifikasi kebutuhan *software* dan *hardware* untuk implementasi sistem yang dibangun.

Dipindai dengan CamScanner

BAB VI KESIMPULAN DAN SARAN

Bab ini berisi kesimpulan secara keseluruhan yang diperoleh dari hasil penelitian yang dilakukan, serta saran-saran dalam penerapan sistem pengadaan *indirect procurement* bahan-bahan MRO untuk perusahaan dan pengembangan selanjutnya.

Dipindai dengan CamScanner

MILIK PERPUSTAKAAN STMI
Membaca : Ibadah, Mengambil : Dosa

BAB II

LANDASAN TEORI

2.1. Konsep Dasar Sistem Informasi

Pada era globalisasi dan informasi saat ini, peranan sistem informasi telah menjadi salah satu syarat mutlak atau kunci keberhasilan yang harus dipahami dan diterapkan oleh seluruh organisasi, baik swasta maupun publik. Hampir seluruh aktivitas organisasi, seperti aktivitas pemerintahan, usaha/bisnis, perbankan, pelayanan kepada pelanggan atau konsumen, pengambilan keputusan, formulasi strategi kompetitif organisasi, dan berbagai ragam aktivitas lainnya tidak dapat terlepas dari sistem informasi. Berikut ini akan dijelaskan lebih lanjut pengertian sistem, informasi, dan sistem informasi.

2.1.1. Pengertian Sistem

Menurut Gerald dikutip oleh Ladjamudin (2013) dalam bukunya bahwa dalam mendefinisikan sistem terdapat dua kelompok pendekatan sistem, yaitu sistem yang lebih menekankan pada prosedur dan elemennya. Kelompok Gerald menekankan sistem pada prosedur. Prosedur didefinisikan sebagai suatu urut-urutan yang tepat dari tahapan-tahapan instruksi yang menerangkan apa yang harus dikerjakan, siapa yang mengerjakan, kapan dikerjakan dan bagaimana mengerjakannya. Kelompok kedua adalah Davis, dalam buku Ladjamudin, kelompok Davis mendefinisikan sistem sebagai bagian-bagian yang saling berkaitan yang beroperasi bersama untuk mencapai beberapa sasaran atau maksud. Berikut definisi sistem menurut beberapa ahli:

1. Sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan kegiatan atau untuk melakukan sasaran tertentu (Hutahean, 2014).
2. Sistem adalah kumpulan dari sub sistem, bagian, ataupun komponen apapun baik dalam bentuk fisik maupun non fisik yang saling berhubungan satu sama

Dipindai dengan CamScanner

lain dan bekerja sama secara harmonis untuk mencapai satu tujuan tertentu (Susanto, 2013).

3. Sistem adalah sekelompok elemen-elemen yang terintegrasi dengan tujuan yang sama untuk mencapai suatu tujuan (Yakub, 2012).
4. Sistem adalah jaringan prosedur yang dibuat menurut pola yang terpadu untuk melaksanakan kegiatan-kegiatan pokok perusahaan sedangkan prosedur adalah suatu urutan kegiatan klerikal, biasanya melibatkan beberapa orang dalam satu departemen atau lebih yang dibuat untuk menjamin penanganan secara seragam transaksi perusahaan yang terjadi secara berulang-ulang (Mulyadi, 2010).
5. Sistem adalah kumpulan elemen yang saling berinteraksi dalam suatu kesatuan untuk menjalankan suatu proses pencapaian suatu tujuan utama (Sutarman, 2009).

2.1.1.1. Karakteristik Sistem

Menurut Sutabri (2012), model umum sebuah sistem adalah *input*, proses, dan *output*. Hal ini merupakan konsep sebuah sistem yang sangat sederhana sebab sebuah sistem dapat mempunyai beberapa masukan dan keluaran. Selain itu, sebuah sistem memiliki karakteristik atau sifat-sifat tertentu, yang mencirikan bahwa hal tersebut bisa dikatakan sebagai suatu sistem. Karakteristik untuk dapat dikatakan sebagai suatu sistem adalah sebagai berikut:

1. Komponen Sistem (*Components*)

   Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, yang bekerja sama membentuk satu kesatuan. Komponen-komponen sistem tersebut dapat berupa suatu bentuk subsistem. Setiap subsistem memiliki sifat-sifat dari sistem yang menjalankan suatu fungsi tertentu dan mempengaruhi proses sistem secara keseluruhan. Suatu sistem dapat mempunyai sistem yang lebih besar, yang disebut dengan supra sistem.
2. Batasan Sistem (*Boundary*)

   Ruang lingkup sistem merupakan daerah yang membatasi antara sistem dengan sistem lainnya atau sistem dengan lingkungan luarnya. Batasan sistem

Dipindai dengan CamScanner

ini memungkinkan suatu sistem dipandang sebagai satu kesatuan yang tidak dapat dipisah-pisahkan.

4. Lingkungan Luar Sistem (*Environments*)

   Bentuk apapun yang ada di luar ruang lingkup atau batasan sistem yang mempengaruhi operasi sistem tersebut disebut dengan lingkungan luar sistem. Lingkungan luar sistem ini dapat menguntungkan dan dapat juga merugikan sistem tersebut. Lingkungan luar yang menguntungkan merupakan energi bagi sistem tersebut, yang dengan demikian lingkungan luar tersebut harus selalu dijaga dan dipelihara. Sedangkan lingkungan luar yang merugikan harus dikendalikan karena kalau tidak maka akan mengganggu kelangsungan hidup sistem tersebut.

5. Penghubung Sistem (*Interface*)

   Sebagai media yang menghubungkan sistem dengan subsistem yang lain disebut dengan penghubung sistem atau *interface*. Penghubung ini memungkinkan sumber-sumber daya mengalir dari satu subsistem ke subsistem yang lain. Keluaran suatu subsistem akan menjadi masukan untuk subsistem yang lain dengan melewati penghubung. Dengan demikian terjadi suatu *integrate system* yang membentuk satu kesatuan.

6. Masukan Sistem (*Input*)

   Energi yang dimasukkan kedalam sistem disebut masukan sistem, yang dapat berupa pemeliharaan (*maintenance input*) dan sinyal (*signal input*). Sebagai contoh, di dalam suatu unit sistem komputer, "program" adalah *maintenance input* yang digunakan untuk mengoperasikan komputer sementara "data" adalah *signal input* yang akan diolah menjadi informasi.

7. Keluaran Sistem (*Output*)

   Hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna. Keluaran ini merupakan masukan bagi subsistem lain. Seperti contoh sistem informasi, keluaran yang dihasilkan adalah informasi, yang mana informasi ini dapat digunakan sebagai masukan untuk pengambilan keputusan atau hal-hal lain yang merupakan input bagi subsistem lainnya.

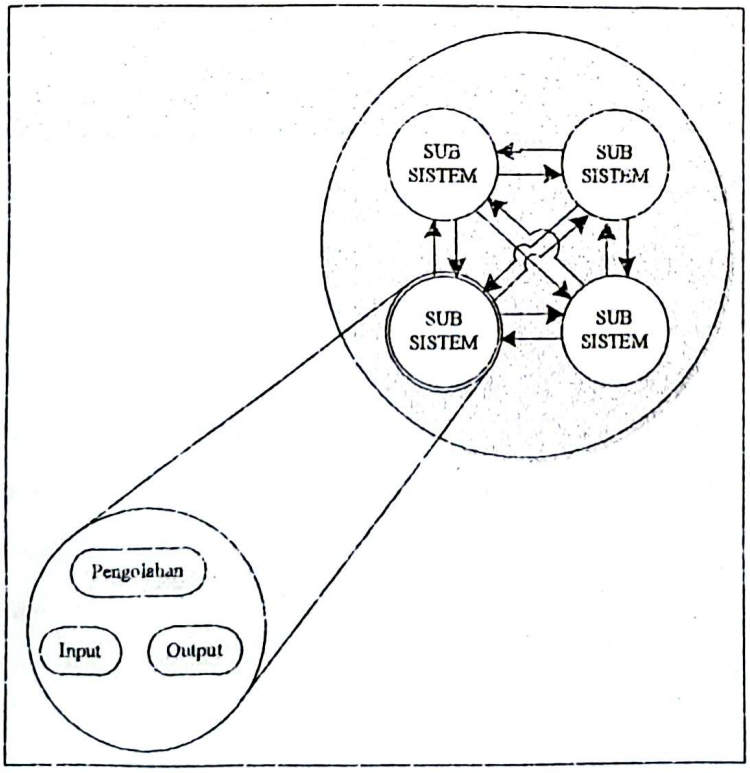
Dipindai dengan CamScanner

8. Pengolah Sistem (Proses)

   Suatu sistem dapat mempunyai suatu proses yang akan mengubah masukan menjadi keluaran. Sebagai contoh, sistem akuntansi. sistem ini akan mengolah data transaksi menjadi laporan-laporan yang dibutuhkan oleh pihak manajemen.

9. Sasaran Sistem (*Objective*)

   Suatu sistem memiliki tujuan dan sasaran yang pasti dan bersifat *deterministic*. Kalau suatu sistem tidak memiliki sasaran, maka operasi sistem tidak ada gunanya. Suatu sistem dikatakan berhasil bila mengenal sasaran atau tujuan yang telah direncanakan.



Gambar II.1. Karakteristik Sistem
(Sumber: Sutabri, 2012)

2.1.2. Pengertian Informasi

Menurut Rusdiana (2014), informasi adalah suatu data yang sudah diolah atau diproses sehingga menjadi suatu bentuk yang memiliki arti bagi penerima informasi yang memiliki nilai bermanfaat. Sesuatu yang diolah oleh suatu sistem menjadi sumber dari infomasi yaitu data, ketika menguraikan informasi harus

Dipindai dengan CamScanner

dikaitkan dengan pengertian data. Data adalah kenyataan yang menggambarkan suatu kejadian-kejadian dan kesatuan nyata. Kejadian-kejadian *(event)* adalah sesuatu yang terjadi pada saat tertentu. Berikut adalah pengertian Informasi menurut beberapa ahli:

1. Informasi adalah data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi penerimanya, yang dimaksud dengan informasi adalah data yang diolah sehingga dapat dijadikan dasar-dasar untuk mengambil keputusan yang tepat (Hutahean, 2014).
2. Informasi adalah data yang telah diklasifikasi atau diolah dan diinterprestasikan untuk digunakan dalam proses pengambilan keputusan (Sutabri, 2012).

2.1.2.1. Pengelompokkan Informasi

Informasi merupakan proses lebih lanjut dari data yang sudah memiliki nilai tambah. Sutabri dalam bukunya membagi informasi menjadi 3 kelompok yaitu:

1. Informasi Strategis

   Informasi ini digunakan untuk mengambil keputusan jangka panjang, yang mencakup informasi eksternal, rencana perluasan perusahaan, dan sebagainya.
2. Informasi Taktis

   Informasi ini dibutuhkan untuk mengambil keputusan jangka menengah, seperti informasi tren penjualan yang dapat dimanfaatkan untuk menyusun rencana penjualan.
3. Informasi Teknis

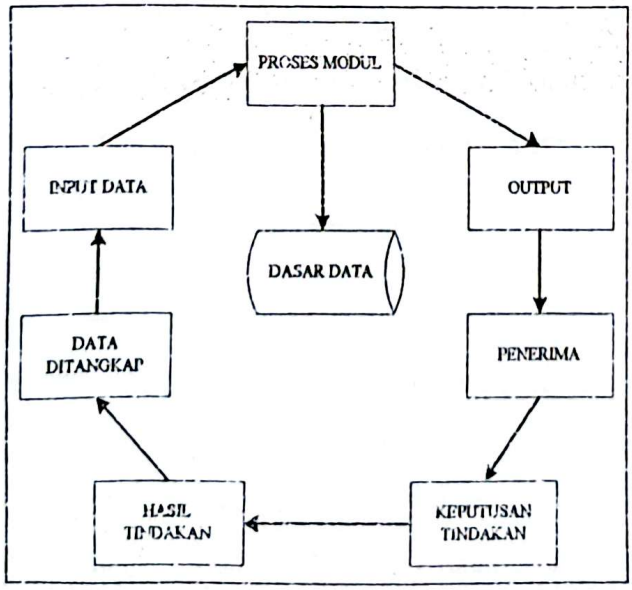
   Informasi ini dibutuhkan untuk keperluan operasional sehari-hari, seperti informasi persediaan stok, retur penjualan dan laporan kas harian.

Dipindai dengan CamScanner

2.1.2.2. Siklus Informasi

Siklus informasi adalah gambaran secara umum mengenai proses terhadap data sehingga menjadi informasi yang bermanfaat bagi pengguna. Informasi yang menghasilkan informasi berikutnya. Demikian seterusnya proses pengolahan data menjadi informasi.

Menurut Sutabri (2012), data diolah melalui suatu model informasi. *Receiver* akan menerima informasi tersebut untuk membuat suatu keputusan dan melakukan tindakan yang akan mengakibatkan munculnya sejumlah data lagi. Data tersebut akan ditangkap sebagai *input*, diproses kembali lewat suatu model, dan seterusnya sehingga membentuk suatu siklus. Siklus inilah yang disebut Siklus Informasi *(Information Cycle)*. Berikut adalah gambar siklus informasi:



Gambar II.2. Siklus Informasi
(Sumber: Sutabri, 2012)

2.1.3. Pengertian Sistem Informasi

Menyangkut pemahaman tentang pengertian sistem informasi ini, dalam bukunya Mulyanto (2009) yang berjudul Sistem Informasi Konsep dan Aplikasi mengutip beberapa pendapat para ahli, diantaranya:

1. Menurut James Alter, sistem informasi adalah "Kombinasi antar prosedur kerja, informasi, orang dan teknologi informasi yang diorganisasikan untuk mencapai tujuan dalam sebuah organisasi".

Dipindai dengan CamScanner

2. Menurut Bodnar dan Hopwood, sistem informasi adalah "Kumpulan perangkat keras dan perangkat lunak yang dirancang untuk mentransformasikan data ke dalam bentuk informasi yang berguna".
3. Menurut Gelinas, Oram dan Wiggins, sistem informasi adalah " Suatu sistem buatan manusia yang secara umum terdiri atas sekumpulan komponen berbasis komputer dan manual yang dibuat untuk menghimpun, menyimpan, dan mengelola data serta menyediakan informasi keluaran kepada pemakai ".
4. Menurut Turban, McLean dan Waterbe, sistem informasi adalah "Sistem yang mengumpulkan, memproses, menyimpan, menganalisis, dan menyebarkan informasi untuk tujuan spesifik".
5. Menurut Joseph Wilkinson, sistem informasi adalah "Kerangka kerja yang mengkoordinasikan sumber daya (manusia, komputer) untuk mengubah masukan (*input*) menjadi keluaran (informasi), guna mencapai sasaran-sasaran perusahaan".

2.1.3.1. 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, 2012).

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 dimasukkan, 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.

Dipindai dengan CamScanner

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 bagian utama, yaitu teknisi *(brainware)*, perangkat lunak *(software)*, dan perangkat keras *(hardware)*.

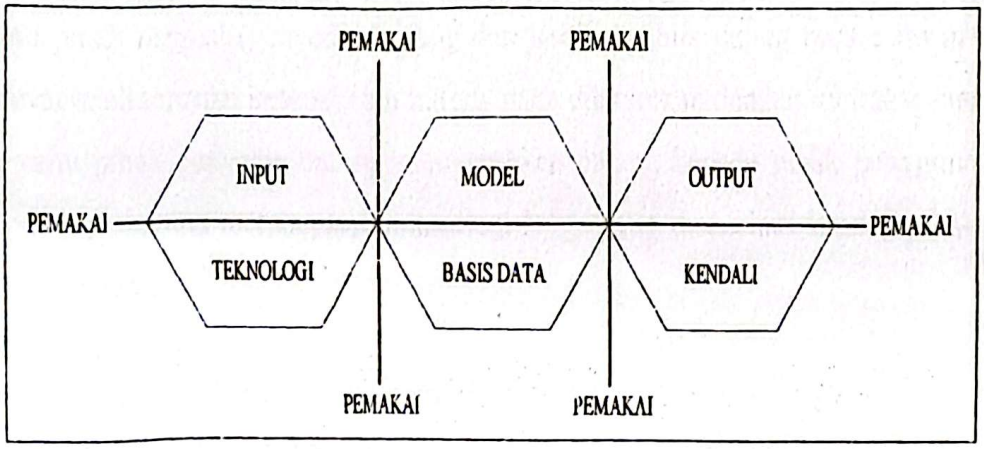
5. Blok Basis Data *(Database Block)*

   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.

Dipindai dengan CamScanner



Gambar II.3. Komponen Sistem Informasi
(Sumber: Sutabri, 2012)

2.2. Pengertian *Procurement*

Procurement dalam bahasa Indonesia memiliki arti pengadaan dimana secara umum tahapan ini melibatkan banyak proses. Proses yang terlibat pada tahap *procurement* adalah perencanaan strategis, penentuan kebutuhan, sistem persetujuan atas permintaan, pemilihan vendor, pembelian, penerimaan maupun penilaian *supplier performance.*

Definisi lain mengenai pengadaan barang dan jasa yaitu seperti yang diungkapkan Marbun (2010) dimana pengadaan barang dan jasa adalah upaya mendapatkan barang dan jasa yang diinginkan yang dilakukan atas dasar pemikiran yang logis dan sistematis (*the system of thought*), mengikuti norma dan etika yang berlaku, berdasarkan metode dan proses pengadaan yang baku.

2.2.1. Proses *Procurement*

Proses pengadaan merupakan akuisisi pada sebuah perusahaan. Menurut Sutedi (2012), proses pengadaan barang dan jasa dimulai dari adanya transaksi pembelian/penjualan barang di pasar secara langsung (tunai) yang kemudian berkembang ke arah pembelian berjangka waktu pembayaran dengan membuat dokumen pertanggung jawaban antar pembeli dan penjual dan pada akhirnya melalui pengadaan melalui proses pelelangan.

Cara atau metode yang digunakan dalam proses pengadaan barang dan jasa adalah cara tawar-menawar secara langsung antara pihak pembeli (pengguna)

Dipindai dengan CamScanner

dengan pihak penjual (penyedia barang dan jasa). Apabila dalam proses tawar-menawar telah tercapai kesepakatan harga, maka dilanjutkan dengan transaksi jual beli, yaitu pihak penyedia barang menyerahkan barang kepada pihak pengguna dan pihak pengguna membayar berdasarkan harga yang disepakati kepada pihak penyedia barang.

2.2.2. Kategori *Aktivitas Procurement*

Menurut Neef (2001), aktifitas pengadaan dapat dibedakan menjadi dua poin utama yaitu pengadaan tidak langsung dan pengadaan langsung. Pengadaan langsung (*indirect procurement*) mencerminkan proses pengadaan untuk komoditi yang tidak menghasilkan barang jadi. Pengadaan ini meliputi dua jenis barang yakni barang-barang keperluan kantor sering disebut *Operating Resource Management* (ORM) dan barang-barang keperluan pemeliharaan sering disebut *Maintenance, Repair, and Operation* (MRO). Kedua jenis barang ini memiliki nilai beli dari yang harga per unitnya rendah seperti perlengkapan hingga yang harga per unitnya cukup tinggi seperti jasa konsultasi dan peralatan berat.

Jenis pengadaan langsung meliputi komoditi yang berkaitan secara langsung untuk menghasilkan produk jadi. Biasanya nilai dari bahan baku ini sangat besar dan memiliki proporsi yang cukup siginifikan dalam perhitungan harga pokok produk per unitnya. Contoh pengadaan langsung antara lain pengadaan gas alam untuk pabrikasi pupuk, pengadaan besi untuk pengolahan baja, dan lainnya. Pengadaan MRO terdiri dari urutan peristiwa-peristiwa berikut:

1. Pencarian katalog
2. Membandingkan barang pencarian katalog
3. Membuat permintaan
4. Menyetujui permintaan
5. Memesan barang
6. Memenuhi *order*
7. Membayar pemasok

Dipindai dengan CamScanner

2.2.3. Prinsip *Procurement*

Menurut Neef (2001) pentingnya pengadaan pada sebuah perusahaan dapat ditinjau dari berbagai aspek, diantaranya yaitu:

1. Aspek keuangan, pada proses pengadaan perlu menggunakan biaya yang paling rendah.
2. Aspek barang, pada aspek ini perlu diperhatikan pengadaan barang dengan cara baik dan benar.
3. Aspek tempat, dalam tahap pengadaan juga diperlukan perhatian khusus bagaimana cara penyimpanan barang tersebut agar barang yang dibutuhkan oleh pengguna tidak mudah rusak.

Penerapan prinsip pengadaan barang dan jasa diperlukan untuk meningkatkan kualitas proses pengadaan barang dan jasa suatu perusahaan. Berikut beberapa pendapat para ahli:

1. Dalam pelaksanaan kegiatan pengadaan sejak perencanaan harus menerapkan prinsip pengadaan barang dan jasa berdasarkan prinsip efisien, efektif, kompetitif, transparan dan bertangungjawab (Siahaya, 2012)
2. Prinsip pengadaan adalah tata nilai utama yang harus dipenuhi dalam setiap proses pengadaan barang/jasa pemerintah. Tata nilai ini mencangkup keseluruhan proses (Samsul, 2011).

2.3. Pengertian Pemeliharaan (*Maintenance*)

Ansori (2013) di dalam bukunya mendefinisikan perawatan atau *maintenance* sebagai konsepsi dari semua aktivitas yang diperlukan untuk menjaga atau mempertahankan kualitas fasilitas/mesin agar dapat berfungsi dengan baik seperti kondisi awal. Aktivitas perawatan sangat diperlukan karena:

1. Setiap peralatan punya umur penggunaan (*useful life*). Suatu saat dapat mengalami kegagalan dan kerusakan.
2. Kita dapat mengetahui dengan tepat kapan peralatan mengalami kerusakan.
3. Manusia selalu berusaha untuk meningkatkan umur penggunaan dengan melakukan perawatan (*maintenance*).

Dipindai dengan CamScanner

Yang menjadi musuh utama bagian perawatan adalah *breakdown*, *deterioration*, dan konsekuensi dari semua tipe kejadian yang tidak terencana. *Maintenance* sangat berperan penting dalam kegiatan produksi dari suatu perusahaan yang menyangkut kelancaran dan kemacetan produksi, volume produksi serta agar produk dapat diproduksi dan diterima konsumen tepat pada waktunya dan menjaga agar tidak ada sumber daya yang menganggur karena kerusakan (*downtime*) pada mesin sewaktu proses produksi sehingga dapat meminimalkan biaya kehilangan produksi.

2.4. Jenis Pemeliharaan

Menurut Ansori (2013) kegiatan pemeliharaan dapat dibedakan menjadi 3 jenis yaitu *corrective maintenance* (*breakdown maintenance*), *preventive maintenance*, dan *total productive maintenance* (perawatan keseluruhan). Berikut ini adalah pengertian dari jenis-jenis pemeliharaan.

1. *Corrective Maintenance*

   Corrective maintenance merupakan kegiatan perawatan yang dilakukan setelah mesin atau fasilitas produksi mengalami ganggugan atau kerusakan sehingga tidak dapat berfungsi dengan baik. Aktivitas *corrective maintenance* sering disebut aktivitas perbaikan. *Corrective maintenance* biasanya tidak dapat kita rencanakan terlebih dahulu karena kita hanya bisa memperbaikinya setelah terjadi kerusakan, bahkan terkadang perbaikan tersebut bisa tertunda atau terlambat.

2. *Preventive Maintenance*

   Preventive maintenance adalah pemeliharaan yang dilakukan terjadwal, umumnya secara periodik, dimana sejumlah tugas pemeliharaan seperti inspeksi, perbaikan, penggantian, pembersihan, pelumasan, dan penyesuaian dilaksanakan. Dengan adanya *preventive maintenance* diharapkan semua mesin yang ada akan terjamin kelancaran proses kerjanya sehingga tidak ada yang terhambat dalam proses produksinya dan bisa selalu dalam keadaan optimal.

Dipindai dengan CamScanner

3. *Total Preventive Maintenance*

Secara teoritis, total biaya pemeliharaan dapat digambarkan bahwa biaya pemeliharaan korektif akan berbanding terbalik dengan pemeliharaan. Pemeliharaan secara produktivitas dapat dilakukan dengan jalan berikut:

a. Mendesain mesin atau peralatan yang memiliki reabilitas tinggi, mudah dioperasikan dan mudah dipelihara.
b. Analisa biaya investasi untuk mesin atau peralatan dengan pelayanan (*service*) dari pemasok dan biaya-biaya pemeliharaannya.
c. Mengembangkan perencanaan pemeliharaan *preventif* yang dapat dimanfaatkan secara praktis oleh operator, bagian pemeliharaan, dan teknisi.
d. Melatih pekerja untuk mengoperasikan mesin atau peralatan, termasuk cara memeliharanya.

2.4. Pemeliharaan Peralatan Produksi *Mold/Dies*

Berdasarkan *Plan Book Mold Dies* oleh PT Cikarang Perkasa Manufacturing (2017), tujuan dari pemeliharaan peralatan *mold/dies* adalah untuk menjaga kualitas pada tingkat yang tepat untuk memenuhi apa yang dibutuhkan oleh *mold* itu sendiri terutama spesifikasi produk yang dihasilkan dengan meningkatkan fungsi dan keakuratan *mold* sehingga aktivitas produksi dapat berjalan stabil. Ada 3 proses kerja yang dilakukan dalam pemeliharaan *mold/dies* yaitu:

1. Pemeliharaan harian untuk pencegahan:
   a. Melakukan pembersihan terhadap *mold* terutama permukaan yang bersentuhan dengan mesin dan permukaan yang bergerak dan tetap pada *mold* sehingga kondisi *mold* selalu bersih sebelum digunakan produksi.
   b. Sesaat sebelum digunakan produksi, lakukan *pre-heating* pada *mold*. *Pre-heating* adalah proses pemanasan awal terhadap *die casting mold* (250°-350° C) yang dilakukan sebelum prodksi dengan cara menggunakan alat pemanas listrik (*electric heater*), gas LPG dengan peralatan pipanya atau dengan menginjeksikan alumunium cair bertekanan rendah ke dalam

Dipindai dengan CamScanner

mold, sehingga perbedaan suhu antara *mold* dan alumunium cair saat digunakan produksi tidak terlalu besar.

c. Selama proses produksi berlangsung, lakukan *spraying* pada posisi yang benar sehingga *die lubricant* dapat menempel pada *mold* (*cavity*) dengan baik dan dapat mencapai posisi-posisi yang sulit dijangkau. *Die lubricant* merupakan cairan untuk pelapisan permukaan pada *mold* yang digunakan pada saat produksi, dengan cara menyemprotkannya pada permukaan *mold* (*cavity*). *Spraying* merupakan proses menyemprotkan *die lubricant* yang dicampur dengan air dengan bantuan tekanan udara dan alat penyemprot (*spray gun*).

2. Pemeliharaan berkala untuk pencegahan:

a. Buat jadwal pemeliharaan *mold* dengan menggunakan "Jadwal Pemeliharaan *Mold* (*Tempering & Polishing*)" sesuai dengan rencana produksi harian pada setiap bulan produksi yang diperoleh dari *section PPC*.

b. Lakukan pemeliharaan *mold* dengan proses *polishing & tempering* sesuai dengan jadwal pemeliharaan *mold* yang telah ditetapkan, dan bila perlu dilakukan penggantian suku cadang *mold*. *Polishing* merupakan proses menghilangkan sisa alumunium dan lapisan endapan hitam *die lubricant* yang menempel pada *mold* (*cavity*) yang terbentuk akibat siklus temperatur dan waktu yang cepat dan berulang-ulang.

c. Lakukan *polishing* terhadap *mold* yang telah dipakai produksi, maksimum 5000 *shots*. Hal ini tergantung pada fungsi dan tingkat keakuratan *mold* tersebut pada saat itu.

d. Pada saat yang sama (bila perlu) lakukan pembersihan terhadap pipa dan selang pendingin dari terak sehingga air pendingin dapat mengalir ke dalam *mold* (*cavity*) tanpa mendapatkan hambatan.

e. Lakukan penggantian terhadap komponen *mold* yang diprediksi akan berpengaruh buruk terhadap kualitas produk hasil produksi.

f. Catat aktivitas *polishing* dan penggantian suku cadang *mold* setiap bulan produksi.

Dipindai dengan CamScanner

g. Untuk memantau kualitas *mold* dan kronologis pemeliharaan *mold*, catat semua aktifitas kejadian pemeliharaan *mold* dari *mold* baru dipakai sampai habis usia pakainya dengan menggunakan "Daftar Riwayat *Mold*"

h. Sesuai dengan rencana produksi pada setiap bulan produksi, sangat dimungkinkan untuk melakukan penggantian *mold* sesuai rencana produksi harian per modelnya. Untuk maksud tersebut, bagian pemeliharaan menerima permintaan penggantian *mold* ("Permintaan Penggantian *Mold*") dari bagian pengguna. Berdasarkan permintaan tersebut, lakukan penggantian *mold*.

3. Perbaikan (*Corrective Maintenance*):

a. Bila didalam proses produksi terjadi kerusakan tiba-tiba (darurat) terhadap *mold* (*cavity*) yang menyebabkan barang hasil produksi tidak standar atau terhambatnya proses produksi maka bagian pemeliharaan menerima permintaan perbaikan *mold*.

b. Lakukan pemeriksaan dan analisa apabila terjadi penyimpangan spesifikasi terhadap *air vent*, *over flow*, *gate overflow*, *main gate*, dan *runner* yang diprediksi akan menyebabkan barang hasil produksi tidak standar atau bahkan akan menimbulkan kerusakan lebih berat pada mold, maka lakukan perbaikan dengan cara pengelasan, permesinan, dan penghalusan.

2.5. *Systems Development Life Cycle* (SDLC)

Menurut Dennis, Wixom, & Tegarden (2010), *Systems Development Life Cycle* (SDLC) atau siklus hidup pengembangan sistem adalah proses pemahaman bagaimana sistem informasi dapat mendukung kebutuhan bisnis dengan merancang sebuah sistem, membangunnya, dan memberikan kepada *user*. *Systems Development Life Cycle* (SDLC) membangun sistem informasi dengan 4 fase fundamental, yaitu:

Dipindai dengan CamScanner

1. *Planning Phase*

   Planning phase adalah proses fundamental untuk memahami mengapa sebuah sistem informasi harus dibangun dan menentukan bagaimana suatu tim proyek akan membangunnya.

2. *Analysis Phase*

   Analysis phase menjawab pertanyaan mengenai apa yang dikerjakan sistem, dimana, dan kapan digunakan. Pada fase ini, tim proyek melakukan investigasi terhadap kondisi sistem yang ada, pengembangan kesempatan, dan pengembangan konsep pada sistem yang baru.

3. *Design Phase*

   Design phase menentukan bagaimana sistem akan dioperasikan dalam *hardware*, *software*, dan infrastuktur jaringan yang akan ditempatkan, tampilan *form*, laporan yang akan digunakan, spesifikasi program, *database*, dan dokumen yang akan dibutuhkan.

4. *Implementation Phase*

   Implementation phase adalah fase terakhir dalam SLDC, fase ini membahas bagaimana sistem dibangun atau yang dibeli, dalam kasus *package software design*. Fase ini yang menjadi fokus utama dari sebagian besar proses pengembangan sistem informasi karena fase ini merupakan fase yang terpanjang dan termahal untuk setiap bagian pengembangan.

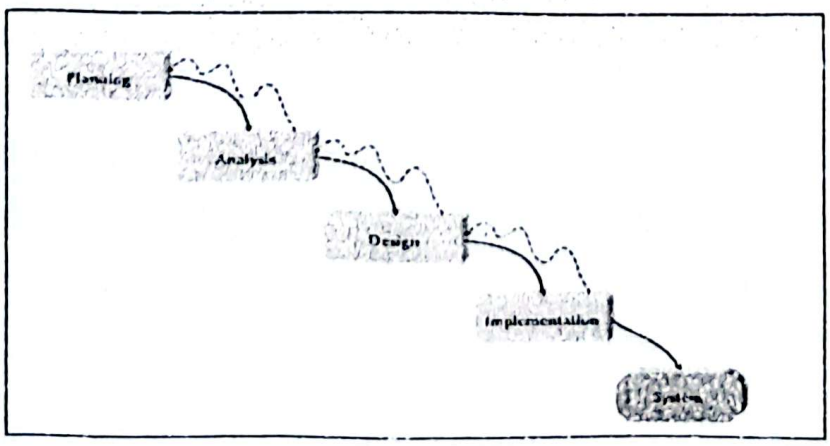
2.6. *Waterfall Development*

Metode pengembangan *waterfall* pertama kali di kemukakan oleh Royce pada tahun 1970. Menurut Dennis, Wixom, & Tegarden (2010), pada penggunaan metode pengembangan *waterfall*, seorang analis dan *user* memproses pengembangan secara bertahap dari satu fase ke fase berikutnya. Setiap fase biasanya berlangsung cukup lama dan setiap fase yang dilewati akan dipresentasikan kepada sponsor untuk mendapatkan persetujuan. Jika sponsor belum menyetujui suatu fase maka pengembangan sistem tidak dapat dilanjutkan ke fase berikutnya. Metodologi ini menyerupai air terjun atau *waterfall* karena

Dipindai dengan CamScanner

air terjun, walaupun dalam SDLC memungkinkan untuk kembali ke fase sebelumnya, namun hal ini akan sulit dilakukan dalam metode *Waterfall.*

Pengembangan dengan metode *waterfall* ini juga mengenalkan penggunaan model formal atau teknik penyusunan rencana untuk menjelaskan proses bisnis sederhana dan data-data pendukung. Keuntungan yang didapat dari pengembangan dengan metode *waterfall* yaitu, dapat mengidentifikasi kebutuhan sistem jauh sebelum proses pemrograman berlangsung sehingga meminimalisir perubahan yang dapat terjadi pada kebutuhan sistem saat proyek berjalan. Kelemahan dari metode *Waterfall* adalah desain harus benar-benar ditentukan sebelum pemrograman dimulai dan lamanya proses yang dibutuhkan untuk menyelesaikan sistem. Berikut penggambaran metode *Waterfall* menurut Dennis, Wixom, & Tegarden (2010) adalah sebagai berikut:



Gambar II.4. Pengembangan dengan Metode *Waterfall*
(Sumber: Dennis, 2010)

2.7. *System Requirements*

Menurut Satzinger, Jackson & Burd, (2012), *system requirements* adalah aktivitas yang harus dilakukan oleh sistem atau *support* atau *constraints* yang sistem harus hadapi. Semua informasi yang dibutuhkan oleh sistem sangatlah penting untuk dicatat, baik itu informasi yang mendeskripsikan *functional requirement* maupun *non-functional requirement. Functional requirement* adalah aktifitas yang harus dilakukan oleh sistem, sedangkan *non-functional requirement* adalah karakter sistem selain aktifitas yang harus di lakukan.

Dipindai dengan CamScanner

2.7.1. *Functional Requirement*

Merupakan penjelasan tentang layanan yang perlu disediakan oleh sistem, bagaimana sistem menerima dan mengolah masukan, dan bagaimana sistem mengatasi situasi-situasi tertentu. *Functional Requirement* menggambarkan *system requirement* secara detil seperti *input*, *output*, dan pengecualian yang berlaku. Contoh *Functional Requirement* pada sistem informasi perpustakaan:

1. Sistem dapat melakukan *input* pendataan buku.
2. Sistem dapat melakukan transaksi peminjaman.
3. Sistem dapat melakukan transaksi pengembalian.
4. Sistem dapat menghitung denda apabila peminjam terlambat mengembalikan buku.
5. Sistem dapat menampilkan laporan bulanan peminjaman dan pengembalian buku.

2.7.2. *Non Functional Requirement*

Secara umum berisi batasan-batasan pada pelayanan atau fungsi yang disediakan oleh sistem. Termasuk di dalamnya adalah batasan waktu, batasan proses pembangunan, standar-standar tertentu. Karena berkaitan dengan kebutuhan sistem secara keseluruhan, maka kegagalan memenuhi kebutuhan jenis ini berakibat pada sistem secara keseluruhan. Contoh *Non Functional Requirement* pada sistem informasi persediaan bahan baku:

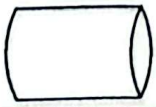
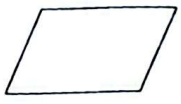
1. Melakukan *input* data bahan baku dengan memasukan data seperti *part number*, *part name*, satuan dan *supplier*.
2. Pencarian data bahan baku dapat dicari dengan kata kunci antara lain *part number*, *part name*, harga, *supplier*, min, masuk, keluar dan stok tersedia.
3. Data yang ditampilkan dari data ketersediaan stok bahan baku antara lain *part number*, *part name*, harga, *supplier*, min, masuk, keluar, stok tersedia, *safety stock* dan rop.

Dipindai dengan CamScanner

2.8. *Flowmap*

Flowmap adalah campuran peta dan *flowchart* yang menunjukan pergerakan benda dari suatu lokasi ke lokasi lain, seperti jumlah orang dalam migrasi, jumlah barang yang diperdagangkan, atau jumlah paket dalam jaringan. *Flowmap* menolong analis dan *programmer* untuk memecahkan masalah kedalam segmen yang lebih kecil dan menolong dalam menganalisis alternatif lain dalam pengoperasian (Jogiyanto, 2010). Simbol-simbol yang digunakan dalam *flowmap* dapat dilihat pada tabel II.1 berikut:

Tabel II.1. Simbol-Simbol *Flowmap*

No	Gambar	Nama	Keterangan
1		Mulai/Akhir	Menunjukkan dimulainya/akhir dari sebuah proses
2		Proses	Menunjukkan kegiatan proses dari operasi program komputer
3		Kegiatan Manual	Menunjukkan kegiatan manual
4		Garis Alir	Menunjukkan arus dari proses
5		Keputusan	Keputusan dalam suatu program
6		Dokumen	Menunjukkan dokumen *input/output* baik untuk proses manual, mekanik, atau komputer
7		*Database*	Melambangkan basis data atau *database*
8		Simbol arsip	Menunjukkan pengarsipan file tanpa menggunakan komputer
9		Simbol penghubung	Menunjukkan penghubung ke halaman yang masih sama atau ke halaman lain
10		*Input-Output*	Simbol yang meyatakan proses *input* dan *output* tanpa tergantung dengan jenis peralatannya.

(Sumber Jogiyanto, 2010)

Dipindai dengan CamScanner

2.9. *Unified Modelling Language* (UML)

Menurut Whitten & Bentley (2007), UML memberikan 9 diagram yang dikelompokkan ke dalam 5 perspektif berbeda dari model menjadi sebuah sistem. Ini seperti sekumpulan *blueprint* yang digunakan untuk membangun sebuah rumah.Dimana ada sekumpulan *blueprints* yang digunakan oleh si pembangun dengan perspektif penjajakan, listrik, pemanas, *air conditioning*, dan lainnya, setiap UML diagram menunjukkan kepada tim pengembang dengan perspektif yang berbeda-beda dari sistem informasi. Jenis-jenis diagram UML dan tujuannya dijelaskan sebagai berikut:

1. *Use-Case Model Diagrams*

   Secara kasar menggambarkan interaksi antara sistem dan sistem eksternal dan *users*. Dengan kata lain, secara kasar mendeskripsikan siapa yang akan menggunakan sistem dan dengan cara apa *user* akan berinteraksi dengan sistem. Narasi *use case* digunakan untuk menambahkan deskripsi tertulis urutan dari langkah-langkah setiap interaksi.

2. *Static Structure Diagrams*

   UML memberikan 2 diagram untuk model struktur statis sebuah sistem informasi. Dua diagram tersebut adalah sebagai berikut:

   a. *Class Diagram* menggambarkan struktur objek sistem. Menampilkan kelas objek dari sistem yang tersusun berikut dan juga hubungannya antara kelas objek.

   b. *Object Diagram*hampir sama dengan *class diagram* tapi selain menggambarkan kelas objek, juga memodelkan objek kasus yang sebenarnya, menunjukkan nilai-nilai atribut-atribut sebuah instansi.

3. *Interaction Diagrams* memodelkan sebuah interaksi, terdiri atas sekumpulan objek-objek, hubungan-hubungannya, dan pesan yang dikirim diantara mereka. Diagram ini memodelkan perilaku dinamis dari sistem, dan UML memiliki 2 jenis diagram untuk tujuan ini. Dua jenis diagram tersebut adalah sebagai berikut:

   a. *Sequence Diagram* menggambarkan secara kasar bagaimana objek-objek

Dipindai dengan CamScanner

eksekusi *use case* atau sebuah operasi. Mereka mengilustrasikan bagaimana pesan-pesan dikirimkan dan diterima antara objek-objek dan dalam urutan apa.

b. *Collaboration Diagram* sama dengan *sequence diagram*, tapi mereka tidak fokus pada waktu atau urutan dari pesan-pesan tersebut. Malah sebaliknya, mereka menampilkan interaksi (atau kolaborasi) antara objek-objek dalam bentuk jaringan. Baik *sequence diagram* dan *collaboration diagram* bersifat *isomorphic*, maksudnya dapat saling berubah bentuk satu sama lain.

4. *State Diagrams*

State diagrams juga memodelkan perilaku dinamis sebuah sistem. UML memiliki diagram untuk memodelkan perilaku kompleks dari sebuah bagian kecil objek (*statechart diagram*) dan sebuah diagram untuk memodelkan perilaku dari sebuah *use case* atau *method*. Yang termasuk dalam *state diagrams* ini adalah:

a. *Statechart diagram* biasanya digunakan untuk memodelkan perilaku dinamis dari sebuah bagian kecil objek. Diagram ini mengilustrasikan sebuah daur hidup objek.

b. *Activity diagrams* biasanya digunakan untuk menggambarkan secara kasar alur *sequential* dari aktivitas-aktivitas baik itu proses bisnis atau sebuah *use case*. Diagram ini juga bisa digunakan untuk memodelkan tindakan-tindakan yang akan dikerjakan ketika operasi dieksekusi, seperti hasil akhir dari beberapa tindakan.

5. *Implementation Diagrams*

Implementation diagrams juga memodelkan struktur dari sistem informasi. Diagram yang termasuk dalam kategori ini adalah:

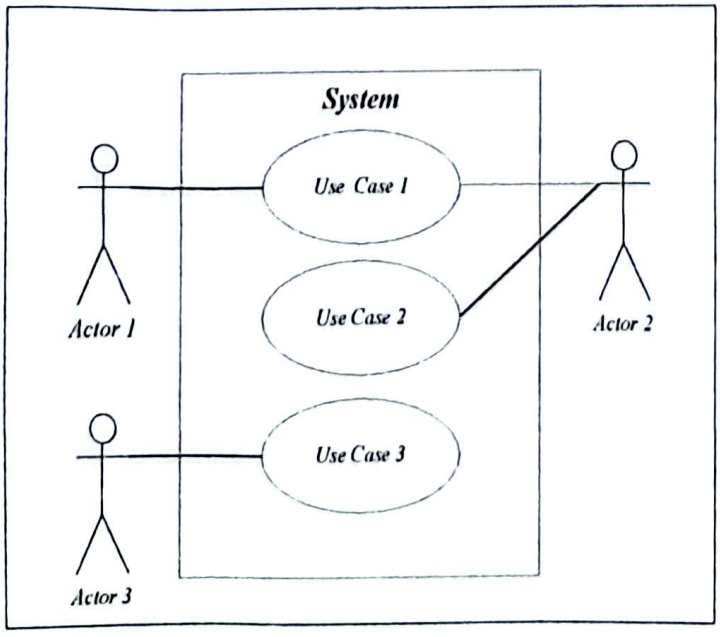
a. *Component diagrams* biasanya digunakan untuk menggambarkan secara kasar organisasi dan ketergantungan dari komponen-komponen perangkat lunak sistem. Diagram ini juga dapat digunakan untuk menunjukkan bagaimana *programming code* dibagi menjadi modul-modul (atau

Dipindai dengan CamScanner

b. *Deployment diagram* mendeskripsikan arsitektur fisik dalam hal ini "*nodes*" untuk perangkat keras dan perangkat lunak sistem. Diagram ini menggambarkan konfigurasi dari *run time* komponen-komponen perangkat lunak sistem, *processors*, dan perangkat yang dapat digunakan untuk menjalankan arsitektur sistem tersebut.

2.9.1. *Use Case Diagram*

Menurut Whitten & Bentley (2007), *use case diagram* digunakan untuk memberikan gambaran umum mengenai keseluruhan fungsionalitas suatu sistem. Sebuah *use case* merepresentasikan sebuah interaksi antara *actor* dengan sistem. *Use case* terkumpul dalam suatu ruang lingkup yang disebut *boundary*. Semua yang berada dalam cakupan kotak mewakili fungsi yang ada dalam lingkup sebuah sistem.



Gambar II.5. ***Use Case Diagram***
(Sumber: Whitten & Bentley, 2007)

Dipindai dengan CamScanner

Notasi-notasi yang digunakan dalam perancangan *use case* diagram yaitu seperti pada tabel II.2. berikut ini:

Tabel II.2. Simbol-Simbol *Use Case Diagram*

No	Gambar	Nama	Keterangan
1		*Actor/Role*	• Beseorang atau sistem yang mendapatkan keuntungan dari sistem. • Digambarkan sebagai gambar *stick* gambar orang (*default*) atau jika bukan seorang aktor manusia. • Dapat digabungkan dengan *actors* lain menggunakan spesialisasi/asosiasi *superclass*, ditandai dengan lubang pada ujung panah. • Diletakkan di luar batas sistem.
2		*Use Case*	• Merepresentasikan bagian utama dari sistem secara fungsional. • Diletakkan di dalam *system boundary*. • Dilabelkan dengan frasa kata kerja deskriptif.
3		*Subject Boundary*	• Nama *subject* berada didalam atau diatas. • Menggambarkan area *subject*, misalnya sebuah sistem atau individu proses bisnis
4		*Association Relationship*	• Menghubungkan suatu *actor* dengan *use case*
5		*Include Relationship*	• Menggambarkan penambahan fungsi dari 1 *use case*yang berhubungan dengan *use case* lainnya • Arah panah dari *use case* asal menuju *use case* tambahan.
6		*Extend Relationship*	• Menggambarkan perluasan *use case* yang menambahkan tingkah laku baru. • Arah panah dari *use case* tambahan menuju *use case* asal.
7		*Generalization Relationship*	• Menggambarkan *use case* yang terspesialisasi terhubung dengan *use case* umum lainnya • Arah panah dari *use case* yang terspesialisasi menuju *use case* umum.

(Sumber: Dennis, Wixom, & Tegarden, 2010

Dipindai dengan CamScanner

2.9.2. *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 diagram alir data, diagram aktivitas termasuk notasi yang membahas pemodelan paralel, kegiatan bersamaan dan proses (Dennis, Wixom, & Tegarden, 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:

1. Rancangan proses bisnis dimana setiap urutan aktivitas yang digambarkan merupakan proses bisnis sistem yang didefinisikan.
2. Urutan atau pengelompokkan tampilan dari sistem/*user interface* dimana setiap aktivitas dianggap memiliki sebuah rancangan antarmuka tampilan.
3. Rancangan pengujian dimana setiap aktivitas dianggap memerlukan sebuah pengujian yang perlu didefinisikan kasus ujinya.
4. Rancangan menu yang akan ditampilkan pada perangkat lunak.

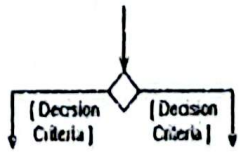
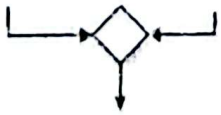
Berikut adalah simbol-simbol yang ada pada *Activity Diagram* seperti pada tabel II.3. dibawah ini:

Tabel II.3. Simbol-simbol *Activity Diagram*

No.	Nama elemen	Fungsi	Notasi
1.	*Action*	Untuk menggambarkan perilaku yang sederhana dan bersifat *non- decomposable.*	*Action*
2.	*Activity*	Untuk mewakili kumpulan aksi (*action*).	*Activity*
3.	*Object Node*	Untuk mewakili objek yang terhubung dengan kumpulan *object flow.*	*Class Name*
4.	*Control Flow*	Menunjukkan rangkaian pelaksanaan.	⟶

Dipindai dengan CamScanner

Tabel II.3. Simbol-simbol *Activity Diagram* (lanjutan)

No.	Nama elemen	Fungsi	Notasi
5.	*Object Flow*	Menunjukkan aliran sebuah objek dari sebuah aktivitas atau aksi ke aktivitas atau aksi lainnya.	
6.	*Initial Node*	Menandakan awal dari kumpulan aksi atau aktivitas.	
7.	*Final-Activity Node*	Untuk menghentikan seluruh *control flows atau object flows* pada sebuah aktivitas (atau aksi).	
8.	*Final-Flow Node*	Untuk menghentikan *control flow* atau *object flow* tertentu.	
9.	*Decision Node*	Untuk mewakili suatu kondisi pengujian yang bertujuan untuk memastikan bahwa *control flow* atau *object flow* hanya menuju ke satu arah.	
10.	*Merge Node*	Untuk menyatukan kembali *decision path* yang dibuat dengan menggunakan *decision node*.	

(Sumber: Dennis, Wixom, & Tegarden, 2010)

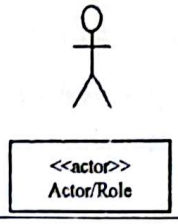
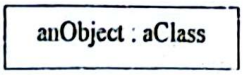
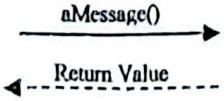
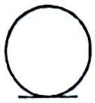
2.9.3. *Sequence Diagram*

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 menunjukan urutan eksplisit pesan yang lewat di antara objek dalam interaksi didefinisikan. Karena urutan diagram menekankan pemesanan berbasis waktu kegiatan yang

Dipindai dengan CamScanner

real-time dan kompleks menggunakan kasus (Dennis, Wixom, & Tegarden, 2010). Berikut simbol yang ada pada *Sequence Diagram* pada tabel II.4. berikut:

Tabel II.4. Simbol-simbol *Sequence Diagram*

No	Simbol	Nama	Keterangan
1.		Aktor	Orang atau sistem yang berasal dari manfaat dan eksternal ke sistem yang berpartisipasi secara berurutan dengan mengirim dan atau menerima pesan
2.		Objek	Berpartisipasi secara berurutan dengan mengirim dan atau menerima pesan yang ditempatkan diatas diagram.
3.		Garis Hidup/*lifeline*	Menyatakan kehidupan suatu objek.
4.		*Execution Occurrence*	Menyatakan objek dalam keadaan aktif dan berinteraksi pesan.
5.		*Message*	Pesan yang menggambarkan komunikasi yang terjadi antar objek.
6.		*Boundary*	Terletak antara sistem dengan dunia disekelilingnya. Semua form, laporan, antarmuka ke perangkat keras seperti printer atau scanner dan antarmuka ke sistem lainnya termasuk dalam kategori
7.		*Control*	Berhubungan dengan fungsionalitas seperti pemanfaatan sumber daya, pemrosesan distribusi, atau penanganan kesalahan
8.		*Entity*	Digunakan untuk menangani informasi yang mungkin akan disimpan permanen

(Sumber: Fowler, 2005)

Dipindai dengan CamScanner

2.9.4. *Class Diagram*

Menurut Whitten & Bentley (2007), *class diagram* adalah gambaran mengenai *object-oriented* atau struktur yang menyusun sebuah sistem dan penggambaran hubungan antar kelas-kelas yang ada. Sebuah *class diagram* disusun oleh tiga bagian, yaitu:

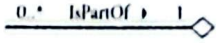
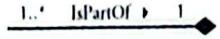
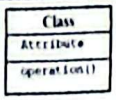
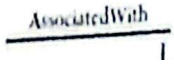
1. Nama *objek* yang digunakan untuk membedakan antara satu kelas dengan kelas lainnya.
2. Atribut kelas yang merepresentasikan karakter dari sebuah objek.
3. *Behavior* yang merupakan operasi-operasi yang digunakan.

Tabel II.5. ***Class Diagram Multiply***

Multiplicity	Arti
1	Tepat satu
0..1	Nol atau satu
0..*	Nol atau banyak
1..*	Satu atau banyak
7..9	Angka spesifik

(Sumber: Whitten & Bentley, 2007)

Tabel II.6. Simbol-Simbol ***Class Diagram***

No	Simbol	Nama	Keterangan
1		*Generalization*	Merupakan sebuah *taxonomic relationship* antara *class* yang lebih umum dengan *class* yang lebih khusus.
2		*Aggregation*	Merupakan *logical* bagian dari *relationship* antara *multiple classes* atau sebuah *class*
3		*Composition*	Merupakan *physical* bagian dari *relationship* antara *multiple classes* atau sebuah *class*
4		*Class*	Kelas pada struktur *system*.
5		*Association*	Asosiasi yang menghubungkan *class* dengan *class Multiplycity*.

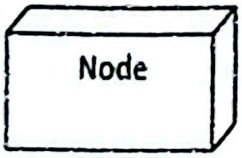
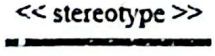
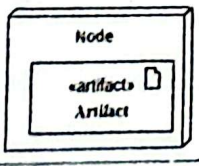
Dipindai dengan CamScanner

2.9.5. *Deployment Diagram*

Deployment diagram digunakan untuk mewakili hubungan antara komponen-komponen *hardware* yang digunakan dalam infrastruktur fisik sistem informasi. Misalnya, ketika mengarang suatu sistem informasi terdistribusi yang akan menggunakan jaringan luas, *deployment diagram* dapat digunakan untuk menunjukkan hubungan komunikasi antara *node* yang berbeda dalam jaringan. *Deployment diagram* juga dapat digunakan untuk mewakili komponen-komponen *software* dan cara *software* ditempatkan pada arsitektur fisik atau infrastruktur sistem informasi.

Dalam hal ini, *deployment diagram* juga dapat digunakan untuk mewakili komponen-komponen *software* dan cara *software* ditempatkan pada arsitektur fisik atau infrastruktur sistem informasi. Simbol-simbol *Deployment Diagram* dapat dilihat pada Tabel II.7. berikut:

Tabel II.7. Simbol-Simbol *Deployment Diagram*

No	Simbol	Nama	Deskripsi
1		*Node*	Untuk menggambarkan sumber daya komputasi dalam sebuah sistem (misalnya, computer klien, *server*, jaringan yang terpisah, atau individu perangkat jaringan).
2		*Artifact*	Untuk menggambarkan spesifikasi dari *software* atau *database*, misalnya *file* sumber, tabel *database*, *executable file*.
3		*Communication Path*	Untuk menggambarkan hubungan antara dua *node* untuk bertukar pesan.
4		*Node with a Deployed Artifact*	Untuk menggambarkan *artifact* yang ditempatkan pada *node* fisik. Mendukung pemodelan distribusi perangkat lunak melalui jaringan.

(Sumber: Dennis, Wixom, & Tegarden, 2010)

Dipindai dengan CamScanner

2.10. *Database*

Menurut Atzeni, Ceri, Paraboschi, & Torlone (2003), *database* adalah sekumpulan data yang digunakan untuk merepresentasikan informasi yang diinginkan dan diimplementasikan kedalam sistem. Tujuan utama pengelolaan data dalam *database* adalah agar kita dapat memperoleh data yang kita cari dengan mudah dan cepat. Pemanfaatan *database* dilakukan untuk memenuhi sejumlah tujuan seperti berikut ini:

1. Kecepatan dan kemudahan (*speed*)
2. Efisiensi ruang penyimpanan (*space*)
3. Keakuratan (*accuracy*)
4. Ketersediaan (*availability*)
5. Kelengkapan (*completeness*)
6. Keamanan (*security*)
7. Kebersamaan pemakaian (*sharability*)

Dalam penggunaannya, *database* memiliki beberapa keuntungan yaitu:

1. Mengurangi kesalahan yang disebabkan oleh faktor manusia. Tugas mekanis lebih baik dilaksanakan oleh mesin.
2. Komputer dapat mengambil dan mengubah data lebih cepat dari manusia.
3. Akurat dan informasi terbaru selalu tersedia setiap saat.
4. Menghemat ruangan karena tidak perlu menyediakan ruangan penyimpanan kertas *file* yang sangat banyak.

2.11. *Entity Relationship Diagram* (ERD)

Menurut Whitten & Bentley (2007), ERD adalah permodelan data yang menggunakan beberapa notasi untuk menggambarkan data yang berhubungan dengan *entity* dan hubungan (*relationships*) yang dideskripsikan oleh data tersebut. Menurut pendapat Satzinger, Burd, & Jackson (2012), ERD adalah diagram yang berisi data *entity* dan hubungannya. Sedangkan data *entities* adalah istilah yang digunakan dalam ERD untuk mendeskripsikan sekumpulan benda atau benda secara individu. Notasi-notasi yang digunakan dalam perancangan ERD yaitu:

Dipindai dengan CamScanner

1. *Data entities* digambarkan dengan persegi panjang.
2. *Relationship* digambarkan garis yang menghubungkan antar *data entities.*
3. *Cardinality,* menggambarkan jumlah kejadian minimum dan maksimum dari suatu *entity* untuk sebuah kejadian dari *entity* yang berhubungan.

Tabel II.8. Simbol-Simbol ERD

Simbol	Nama	Deskripsi
	*Strong Entity*	Karakteristik atau tipe entitas yang tidak bergantung pada entitas lain.
	*Attribute*	Properti atau karakteristik tipe entitas.
	*Multivalued Attribute*	Karakteristik tipe entitas yang dapat memiliki nilai lebih dari satu.
	*Relationship*	Hubungan antar entitas.
	*Weak Entity*	Entitas yang bergantung pada entitas lain.
	*Identifying Entity*	Entitas yang merupakan induk dari *weak entity.*
	*Associative Entity*	Hanya bisa terdapat antara 2 entitas dan menggambarkan hubungan diantara 2 entitas tersebut.
	*Derived Attribute*	Suatu atribut yang dihasilkan dari atribut yang lain.

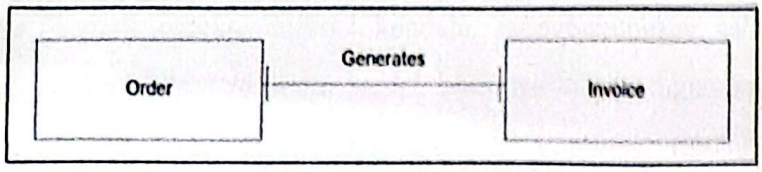
(Sumber: Jeffrey A. Hoffer, 2010)

Relasi adalah hubungan beberapa entitas sedangkan kardinalitas menyatakan jumlah himpunan relasi antar entitas. Berikut ini merupakan macam-macam dari kardinalitas (Brady dan Loonam, 2010) yaitu:

1. *One-to-One*

   Menyatakan hubungan sebuah entitas A dengan entitas B paling banyak satu jenis hubungan.

Dipindai dengan CamScanner



Gambar II. 6. Kardinalitas *One to One*
(Sumber: Brady dan Loonam, 2010)

2. *One-to-Many*

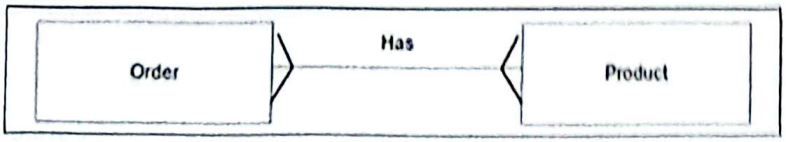
Menyatakan hubungan sebuah entitas A dengan entitas B sebanyak lebih dari satu jenis hubungan.



Gambar II. 7. Kardinalitas *One to Many*
(Sumber: Brady dan Loonam, 2010)

3. *Many-to-Many*

Menyatakan sebuah hubungan entitas A dan entitas B memiliki lebih dari satu jenis hubungan begitu pula sebaliknya.



Gambar II. 8. Kardinalitas *Many to Many*
(Sumber: Brady dan Loonam, 2010)

2.12. *Physical Data Model*

Menurut Connolly & Begg (2010), *physical data model* menjelaskan bagaimana data disimpan dalam komputer, menampilkan informasi seperti struktur dokumen, permintaan dokumen, dan jalur akses dokumen.Sebuah *physical data model* (alias desain *database*) adalah representasi dari desain data yang memperhitungkan fasilitas dan kendala sistem *database* yang diberikan manajemen. Dalam siklus hidup proyek itu biasanya berasal dari model data logis, meskipun mungkin *reverse-engineered* dari implementasi *database* yang diberikan. Sebuah *physical data model* lengkap akan mencakup semua artefak *database* yang diperlukan untuk membuat hubungan antara tabel atau mencapai

Dipindai dengan CamScanner

tujuan kinerja, seperti indeks, definisi kendala, menghubungkan tabel, tabel dipartisi atau *cluster*. *Physical data model* biasanya dapat digunakan untuk menghitung perkiraan penyimpanan dan mungkin termasuk rincian alokasi penyimpanan khusus untuk sistem *database* tertentu. Model data fisik adalah model yang menjelaskan cara komputer memandang data, bahwa data tersimpan pada lokasi fisik sebagai file-file yang terpisah. Model data fisik terbagi menjadi 2 yaitu :

1. Penyimpanan berurutan yaitu organisasi atau penyusunan data di suatu medium penyimpanan yang terdiri dari satu *record* mengikuti satu *record* lain dalam suatu urutan tertentu. Misalnya, *record* pegawai disusun dalam urutan nomor pegawai. Saat penyimpanan berurutan digunakan, data pertama harus diproses pertama, data kedua diproses kedua, dan seterusnya sampai akhir file itu ditemukan. Contoh media penyimpanan ini adalah pita magnetik (*magnetic tape*).
2. Penyimpanan akses langsung yaitu suatu cara mengorganisasikan data yang memungkinkan *record-record* ditulis dan dibaca tanpa pencarian secara berurutan. Unit perangkat keras yang memungkinkan hal ini disebut *Direct Access Storage Device* (DASD). DASD memiliki mekanisme membaca dan menulis yang dapat diarahkan ke lokasi manapun dalam media penyimpanan. Yang paling populer adalah piringan magnetik (*magnetic disc*).

Tabel II.9. Simbol-Simbol pada *Physical Data Model*

Simbol	Deskripsi
Tabel nama_tabel	Tabel yang menyimpan data dalam basis data
Relasi id_tbl1=id_fk_tbl2	Relasi antar tabel yang terdiri dari persamaan antara *primary key* (kunciprimer) tabel yang diacu dengan kunci yang menjadi referensi acuan ditabel lain

(Sumber: Connolly & Begg, 2010)

Dipindai dengan CamScanner

2.13. Kamus Data

Menurut Jogiyanto (2010), dalam buku Analisis dan Desain Sistem Informasi, menjelaskan bahwa kamus data merupakan katalog fakta tentang data dan kebutuhan-kebutuhan informasi dari suatu sistem informasi dalam kamus data harus memuat hal-hal berikut:

1. Nama Arus Data

   Nama arus data dicatat pada kamus data, sehingga mereka yang membaca diagram alir data memerlukan penjelasan lebih lanjut tentang suatu arus data tertentu.

2. Alias

   Untuk menyatakan nama lain dari *element* atau data *store* yang sebenarnya sama dengan data *element* atau data *store* yang telah ada.

3. Bentuk Data

   Dipergunakan untuk mengelompokkan kamus data ke dalam kegunaannya sewaktu perancangan sistem.

4. Arus Data

   Menunjukkan dari mana data mengalir dan kemana data akan menuju, keterangan arus data ini perlu dicatat di kamus data supaya memudahkan mencari arus data ini di diagram alir data.

5. Struktur Data

   Struktur data menunjukkan harus data yang dicatat di kamus data terdiri dari *item - item* data apa saja.

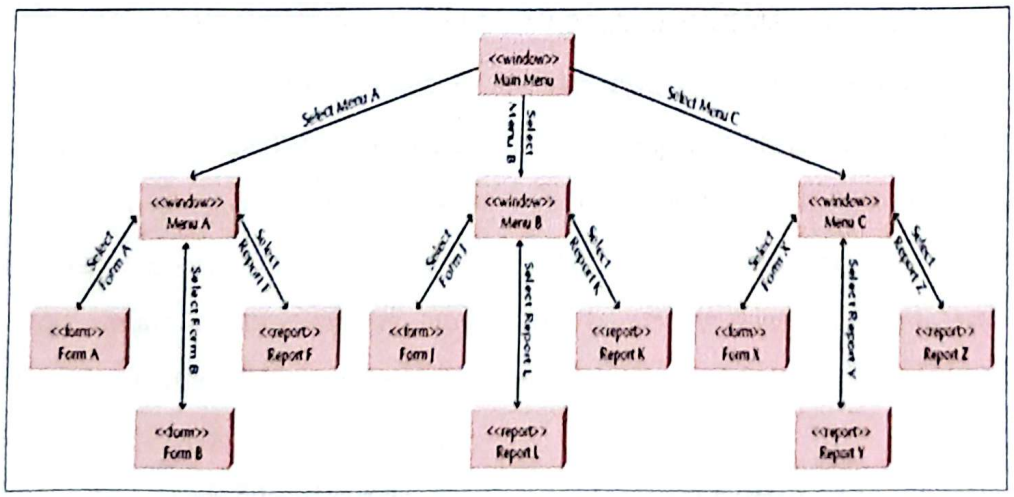
2.14. *Window Navigation Diagram* (WND)

Menurut Dennis, Wixom, & Tegarden (2010), desain struktur navigasi mendefinisikan komponen dasar antarmuka dan bagaimana mereka bekerja sama untuk menyediakan fungsionalitas kepada pengguna. *Windows Navigation Diagram* (WND) digunakan untuk menunjukkan bagaimana semua layar, bentuk dan laporan yang digunakan oleh sistem terkait dan bagaimana pengguna berpindah dari satu ke lainnya. Kebanyakan sistem memiliki beberapa WND's, satu untuk setiap bagian utama dari sistem.

Dipindai dengan CamScanner

WND sangat mirip dengan *state machine*, karena keduanya memodelkan perubahan status mesin. Keadaannya biasanya membuat model perubahan dari suatu objek, sedangkan keadaan model perubahan lainnya yaitu di antarmuka pengguna. Di WND setiap status antarmuka pengguna direpresentasikan sebagai kotak. Kotak biasanya sesuai dengan komponen antarmuka pengguna, seperti jendela, *form*, tombol, atau laporan.

Struktur navigasi dasar antarmuka mengikuti struktur dasar dari proses bisnis itu sendiri, sebagaimana didefinisikan dalam kasus penggunaan dan model perilaku. Analis memulai dengan kasus-kasus penggunaan *essential* dan mengembangkan aliran mendasar dari kontrol sistem ketika bergerak dari objek ke objek.



Gambar II.9. ***Windows Navigation Diagram***
(Sumber: Dennis, Wixom, & Tegarden, 2010)

2.15. PHP

Menurut Saputra (2011), PHP atau PHP *Hypertext Preprocessor* merupakan suatu bahasa pemrograman yang difungsikan untuk membangun suatu *website* dinamis. PHP menyatu dengan kode HTML, maksudnya adalah beda kondisi. HTML digunakan sebagai pembangun atau pondasi dari kerangka *layout web*, sedangkan PHP difungsikan sebagai prosesnya sehingga dengan adanya PHP tersebut, web akan sangat mudah di lakukan *maintenance*. PHP juga merupakan sebuah bahasa pemograman *open source* yang berbasis *server side*. PHP mampu berintegrasi dengan berbagai macam jenis *database*, salah satunya dengan MySQL.

Dipindai dengan CamScanner

2.16. *Framework* Laravel

Menurut Pressman & Bruce (2015), *framework* adalah kerangka kode yang dapat disempurnakan dengan *classes* yang spesifik atau dengan fungsi yang telah dirancang untuk mengatasi masalah yang dihadapi. Dapat disimpulkan bahwa *framework* biasanya bersifat *object-oriented* dan merupakan suatu desain sistem yang dapat digunakan kembali. Tujuannya untuk mengurangi pembuatan kembali kode yang sama sehingga *programmer* dapat lebih fokus mengerjakan bagian lainnya.

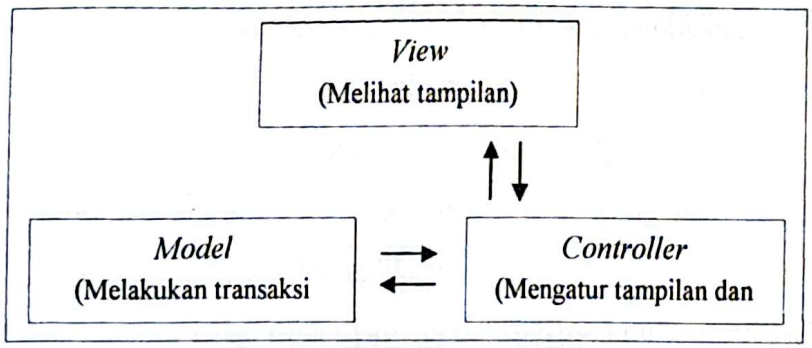
Laravel adalah salah satu *framework* MVC yang ditulis menggunakan bahasa pemrogaman PHP dibuat oleh Taylor Otwell pada tahun 2011. MVC merupakan singkatan dari *Model-View-Controller*. Menurut Subagia (2018), MVC merupakan metode untuk membuat sebuah aplikasi dengan memisahkan antara data (*Model*) dari tampilan (*View*) dan cara bagaimana memprosesnya (*Controller*). Dengan kata lain, konsep MVC merupakan suatu metode dalam pemrogaman dengan memisahkan komponen utama yang membangun aplikasi, yaitu memanipulasi data, *user interface*, dan juga bagian yang mengontrol aplikasi. Dalam implementasinya, tidak sedikit *framework* dalam aplikasi *website* berbasis arsitektur MVC, misalnya CodeIgniter, Laravel, CakePHP, Yii, Phalcon, Zend, PHPixie, Symfony, dan sebagainya.

Penjelasan lebih lengkap mengenai 3 komponen utama dalam konsep MVC adalah sebagai berikut:

1. Model, merupakan bagian yang berhubungan langsung dengan *database* untuk memanipulasi data (*Insert*, *Update*, *Delete* maupun *Search*), menangani validasi dari bagian *Controller*, namun tidak dapat langsung berhubungan dengan bagian *View*.
2. *View*, merupakan bagian yang berisi skrip untuk menerima dan mempresentasikan data kepada *user*. Bagian ini biasanya berupa template HTML yang penampilannya diatur oleh *Controller*. *View* berfungsi untuk menerima dan mempresentasikan data kepada *user*. Bagian ini tidak memiliki akses langsung terhadap bagian model.

Dipindai dengan CamScanner

3. *Controller*, merupakaan bagian yang mengatur hubungan antara bagian model dan bagian *View*. *Controller* berfungsi untuk menerima *request* dan data dari *user* kemudian menentukan apa yang akan diproses oleh aplikasi.



Gambar II.10. Konsep MVC
(Sumber: Subagia, 2018)

Pada gambar diatas, ketika *user* meminta (*request*) atau mengirim data melalui URL website, maka *Controller* yang akan menangani request tersebut. Jika dalam prosesnya membutuhkan data dari database, maka *Controller*akan memberikan tugas tersebut kepada *Model*. *Model* akan langsung berhubungan dengan *Database* untuk melakukan transaksi data (memasukkan, mengambil, memperbaharui, atau menghapus) sesuai permintaan dari *user*. Selanjutnya, data yang diterima dari model akan diserahkan pada *view* oleh *controller* untuk ditampilkan pada halaman *website* melalui *browser* dalam bentuk *file* HTML.

Dibandingkan dengan cara konvensional, pemrograman dengan teknik MVC memiliki beberapa keuntungan diantaranya sebagai berikut:

1. Operasi data hanya dilakukan pada area yang ditentukan (*Model*) tidak tersebar ke seluruh aplikasi, sehingga memudahkan dalam proses pemeliharaan aplikasi.
2. Memudahkan penggabungan divisi desain dalam *team development*.
3. Komponen *View* dapat didesain tanpa harus memperhatikan bagian lain secara berlebihan.

Laravel telah dirancang untuk meningkatkan kualitas perangkat lunak dengan mengurangi biaya, baik biaya pengembangan awal dan biaya pemeliharaan, serta memberikan sintaks ekspresif yang jelas dan setiap fungsi ini yang akan menghemat waktu pengerjaan. Untuk dapat menggunakan Laravel

Dipindai dengan CamScanner

diperlukan *software* pendukung seperti *text editor*, *web server*, dan *composer*. Pada folder hasil *install* Laravel terdapat beberapa folder penting untuk diketahui yaitu:

1. Folder *app* merupakan folder yang berisi semua skrip aplikasi.
2. Folder *bootstrap* merupakan folder yang berisi file app.php yang mengendalikan *framework* dan file autoload.php yang mengkonfigurasi *autoloading*. Folder ini juga berisi folder *cache* yang berisi file-file *cache* untuk meningkatkan kecepatan aplikasi.
3. Folder *config* merupakan folder yang berisi file-file konfigurasi aplikasi.
4. Folder *database* merupakan folder yang berisi *database migration* dan *seeds*.
5. Folder *public* merupakan folder yang berisi file index.php yang merupakan file utama sebagai pintu masuk semua *request* pada aplikasi. Folder ini juga tempat semua *asset* aplikasi seperti gambar, file Javascript, dan CSS.
6. Folder *resource* merupakan folder yang berisi file-file *asset* yang belum dikompilasi seperti file LESS, SASS, atau Javascript. Folder ini juga tempat semua file bahasa.
7. Folder *routes* merupakan folder yang berisi semua *route* yang didefinisikan pada aplikasi.
8. Folder *storage* merupakan folder yang berisi file-file yang dibuat oleh *framework*. Folder ini berisi 3 folder, yaitu *app*, *framework*, dan *logs*. Folder *app* digunakan untuk menyimpan file yang dibuat oleh aplikasi. Folder *framework* digunakan untuk menyimpan file yang dibuat oleh *framework*. Sedangkan folder *logs* digunakan untuk menyimpan file *log*.
9. Folder *tests* merupakan folder yang berisi file-file *dependency* yang diperoleh melalui *Composer*.

Dipindai dengan CamScanner

2.17. Sublime Text

Sublime *text editor* pertama kali dirilis pada tanggal 18 Januari 2009. Sublime Text adalah aplikasi *editor* untuk kode dan teks yang dapat berjalan di berbagai *platform operating system* dengan menggunakan teknologi *phyton* API (Haughee, 2013). Aplikasi ini tercipta karena terinspirasi dari aplikasi Vim. Aplikasi ini sangatlah fleksibel dan *powerfull.* Sublime Text bukanlah aplikasi *open source* yang artinya aplikasi ini membutuhkan lisensi yang harus dibeli.

2.18. *Composer*

Composer digunakan untuk memudahkan instalasi Laravel dan mendapatkan *library* yang dibutuhkan dari internet. Jadi, sebelum *install* Laravel wajib *install* aplikasi ini terlebih dahulu. Aplikasi ini dapat diunduh dari *website* getcomposer.org.Saat instalasi *composer*, pastikan dapat mendeteksi letak PHP.Biasanya terletak pada folder xampp/php (Abdulloh, 2017).

2.19. MariaDB

MariaDB adalah sebuah implementasi dari sistem manajemen basis data relasional (RDBMS) yang didistribusikan secara gratis dibawah lisensi GPL (*General Public License*). Setiap pengguna dapat secara bebas menggunakan MariaDB, namun dengan batasan perangkat lunak tersebut tidak boleh dijadikan produk turunan yang bersifat komersial.

MariaDB mendukung banyak tipe data yang dapat disimpan pada sebuah kolom. Terdapat tiga kategori tipe data yang didukung oleh MySQL, yaitu tipe data numerik, string, serta penanggallan dan waktu. Sebuah data yang akan disimpan harus sesuai dengan tipe data yang bersangkutan (Komputer, 2015). MariaDB merupakan versi pengembangan dari MySQL oleh karena itu untuk saat ini tipe-tipe datanya sama dengan tipe-tipe data MySQL. Penjelasan singkat kegunaan masing-masing jenis data akan sangat bermanfaat dalam memilih suatu jenis data yang dipakai dalam merancang *table*. Adapun jenis data yang tersedia pada MariaDB dapat dilihat pada TabelII.10.

Dipindai dengan CamScanner

Tabel II.10. Jenis Data pada MariaDB

Jenis Data	Keterangan
CHAR	Sebuah *string* 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	*String* 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 *floating point* 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 *floating point* 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 *enumeration*. Sebuah obyek string yang hanya boleh memiliki satu nilai, yang terambil dari 'value1','value2,'...,NULLatau 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. *Black Box Testing*

Pengujian *black box* berfokus pada persyaratan fungsional perangkat lunak. Dengan demikian, pengujian *black box* memungkinkan merekayasa perangkat lunak mendapatkan serangkaian kondisi *input* yang sepenuhnya menggunakan semua persyaratan fungsional untuk suatu program. Pengujian *black box* bukan merupakan alternatif dari teknik *white box*, tetapi merupakan pendekatan komplementer yang kemungkinan besar mampu mengungkap kelas kesalahan dari pada metode *white box* (Pressman & Bruce, 2015). Pengujian *black box* berusaha menemukan kesalahan dengan kategori sebagai berikut:

1. Fungsi-fungsi yang tidak benar atau hilang.
2. Kesalahan *interface*
3. Kesalahan dalam struktur data atau akses *database eksternal*

Dipindai dengan CamScanner

4. Kesalahan kinerja
5. Inisialisasi dan kesalahan terminasi

Tidak seperti *white box*, yang dilakukan pada saat awal proses pengujian, pengujian *black box* cenderung diaplikasikan selama tahap akhir pengujian. Karena pengujian *black box* memperhatikan struktur kontrol, maka perhatian berfokus pada domain informasi(Pressman & Bruce, 2015).

Dipindai dengan CamScanner

MILIK PERPUSTAKAAN STMI
Membaca : Ibadah, Mengambil : Ilmu

BAB III
METODOLOGI PENELITIAN

3.1. Metodologi Penelitian

Menurut Suderyono (2017), metode penelitian pada dasarnya merupakan cara ilmiah untuk mendapatkan data dengan tujuan dan kegunaan tertentu. Cara ilmiah mempunyai karakteristik rasional, empiris, dan sistematis. Rasional berarti penelitian dilakukan dengan cara-cara masuk akal dan terjangkau penalaran atau logika manusia. Empiris berarti penelitian dilakukan berdasarkan fakta-fakta di lapangan yang dapat diuji oleh orang lain atau pihak lain. Kemudian, sistematis berarti penelitian merupakan proses tertentu yang logis.

Penelitian dimulai dengan memunculkan permasalahan, mencari jawaban permasalahan dengan mengkaji literatur untuk membuat hipotesis, mengumpulkan data dari lapangan, menganalisis data dengan teknik yang relevan, lalu pada akhirnya membuat kesimpulan atau temuan. Penelitian dapat dikatakan sebagai usaha penyelidikan yang sistematis dan terorganisasi. Alasannya adalah penelitian merupakan suatu proses yang dilakukan melalui tahapan-tahapan tertentu untuk mencapai tujuan penelitian. Tahapannya dimulai dari adanya permasalahan, mengumpulkan fakta-fakta, menganalisis data dengan teknik tertentu, hingga membuat kesimpulan. Kesimpulan merupakan pengetahuan yang diharapkan mampu memecahkan masalah atau menjawab pertanyaan. Tahap penelitian (proses penelitian) dilakukan dengan menggunakan cara atau prosedur tertentu yang diatur dengan baik. Metodologi penelitian adalah ilmu yang mengkaji ketentuan atau aturan mengenai metode yang digunakan dalam penelitian.

3.2. Jenis dan Sumber Data

Menurut Sugiyono (2009), pengumpulan data dapat menggunakan sumber primer dan sumber sekunder.

1. Sumber Primer

   Sumber primer adalah sumber data yang langsung memberikan data kepada pengumpul data. Data yang menjadi sumber primer adalah data yang

Dipindai dengan CamScanner

diperoleh melalui pengamatan langsung terhadap sistem yang sedang berjalan dan wawancara dengan *department maintenance*, *warehouse*, dan *purchasing* sebagai sumber informasinya. Dalam penelitian ini data tersebut berupa proses bisnis, prosedur pemeliharaan *mold/dies*, pembuatan PR dan PO, prosedur pembuatan *warehouse stock report*, prosedur pembuatan *purchasing report*, dan prosedur pembelian bahan-bahan MRO pada PT Cikarang Perkasa Manufacturing.

2. Sumber Sekunder

   Sumber sekunder merupakan yang tidak langsung memberikan data kepada pengumpul data, misalnya lewat orang lain atau lewat dokumen. Dalam penelitian ini data tersebut diperoleh dari perusahaan dan modul yang disediakan oleh Departemen *Maintenance*. Data tersebut adalah data dokumen *warehouse stock report* dan dokumen *purchasing report*.

3.3. Metode Pengumpulan Data

Menurut Sugiyono (2009), teknik pengumpulan data merupakan langkah yang paling strategis dalam penelitian, karena tujuan utama dari penelitian adalah mendapatkan data. Penelitian ini menggunakan beberapa teknik pengumpulan data, yaitu:

1. Studi lapangan

   Studi lapangan adalah usaha melakukan pengumpulan data secara langsung pada objek yang dilakukan dengan cara sebagai berikut:

   a. Pengamatan, tahap ini dilakukan secara langsung di bagian Departemen *Maintenance*, Departemen *Purchasing*, dan Departemen *Warehouse* dengan mengamati sistem informasi pengadaan bahan-bahan MRO pada Departemen *Maintenance*, mulai dari pengajuan pembelian bahan MRO, pembelian bahan MRO, dan kedatangan bahan MRO. Melalui teknik ini, data yang dibutuhkan diamati, dikumpulkan dan diolah sebagai bahan dalam penelitian.

   b. Wawancara, yaitu mencari data yang dibutuhkan secara langsung dengan memberikan pertanyaan spontan tentang proses bisnis pengadaan bahan-

Dipindai dengan CamScanner

bahan MRO yang diperlukan pada penyusunan tugas akhir ini. Wawancara ini dilakukan kepada pegawai di bagian *Purchasing* yaitu Bapak Chiko, pegawai di bagian *Warehouse* yaitu Bapak Taqhim, dan pegawai di bagian *Maintenance* yaitu Bapak Ndandung dan Bapak Parlan.

2. Studi Literatur

Tahap ini merupakan tahap awal pengumpulan data dari sumber-sumber seperti buku yang berhubungan *indirect procurement* bahan MRO, analisis dan desain sistem informasi menggunakan UML, pemrograman web dengan PHP, MySQL, sistem informasi manajemen, pengembangan perangkat lunak, dan juga mencari data tambahan yang diperlukan seperti konsep pemrograman PHP, beberapa contoh *coding* program melalui media *internet.*

3.4. Kerangka Penelitian

Dalam penelitian yang dibahas dalam Tugas Akhir ini, dilakukan langkah-langkah atau tahapan dalam penelitian yang ada pada Gambar III.1 sesuai dengan metodologi penelitian yang digunakan dalam penelitian tersebut. Penjelasan langkah-langkah atau tahapan dalam penelitian tersebut adalah sebagai berikut:

1. Melakukan Perencanaan (*Planning*)

Pada tahap ini penulis melakukan perencanaan dalam membangun sistem dengan beberapa cara seperti:

a. Studi Pendahuluan

Pada tahap ini penulis melakukan studi pendahuluan dengan beberapa cara seperti:

1) Melakukan studi pustaka dengan mempelajari buku referensi, jurnal-jurnal yang serupa dengan topik yang diambil, dan menelaah dari berbagai laporan terkait seperti, struktur organisasi, proses pembelian yang dilakukan di perusahaan, dan proses bisnis yang digunakan di PT Cikarang Perkasa Manufacturing.

Dipindai dengan CamScanner

2) Melakukan proses permagangan selama tiga bulan di PT Cikarang Perkasa Manufacturing pada Departemen *Purchasing* dengan mengumpulkan data yang dibutuhkan untuk melakukan analisis.

b. Identifikasi Masalah

Pada tahap ini penulis melakukan identifikasi masalah dengan beberapa cara seperti:

1) Observasi selama tiga bulan dengan melakukan pengumpulan data di PT Cikarang Perkasa Manufacturing pada proses persediaan bahan-bahan MRO pada Departemen *Purchasing*.

2) Wawancara pada departemen terkait yaitu *warehouse*, *maintenance*, dan *purchasing* yang terlibat dalam proses persediaan bahan MRO.

3) Melakukan analisis dari hasil observasi dan wawancara yang telah dilakukan pada departemen *maintenance*, *warehouse*, dan *purchasing* dalam bentuk transkrip wawancara dan hasil observasi.

c. Identifikasi Solusi

Pada tahap ini bertujuan mengetahui maksud dan tujuan dari pembuatan laporan ini, pada departemen terkait. Dengan penjabaran antara lain:

1) Mengidentifikasi maksud dan tujuan dari melakukan penelitian di departemen *Purchasing* dengan maksud mencari masalah apa yang ada di PT Cikarang Perkasa Manufacturing pada proses pembelian bahan-bahan MRO.

2) Mengidentifikasi batasan masalah yang ada pada PT Cikarang Perkasa Manufacturing yaitu pada pendataan penerimaan bahan MRO, pengolahan data pengeluaran bahan MRO dan laporan ketersediaan bahan MRO.

3) Mengembangkan sistem dengan menggunakan metode *waterfall*, dengan alasan metode ini melakukan pendekatan secara sistematis dan berurut mulai dari *level* kebutuhan sistem lalu menuju ke tahap analisis, desain, *coding*, *testing* dan *maintenance*.

Dipindai dengan CamScanner

2. Melakukan Analisis (*Analysis*)

Pada tahap ini penulis melakukan identifikasi kebutuhan sistem dengan beberapa cara seperti:

a. Analisis sistem yang berjalan yaitu, melakukan analisis dokumen *input* dan *output* yang berjalan dan menganalisis proses bisnis yang berjalan di PT Cikarang Perkasa Manufacturing.

b. Analisis sistem usulan yaitu, mengidentifikasi kebutuhan pengembangan sistem informasi yang ada di PT Cikarang Perkasa Manufacturing terutama departemen *Maintenance*, *Warehouse*, dan *Purchasing*.

c. Membuat *flowmap* usulan untuk mengetahui alur proses bisnis dari sistem yang digunakan.

3. Melakukan Desain (*Design*)

Pada tahap ini penulis melakukan perancangan sistem dimana dalam pembuatan sistem menggunakan dua proses yaitu:

a. Menggunakan *use case diagram* yang bertujuan untuk menggambarkan proses sistem yaitu kebutuhan sistem dari sudut pandang *user*, menggunakan *activity diagram* yang bertujuan menggambarkan proses bisnis *modelling* untuk memperlihatkan urutan aktifitas proses bisnis, menggunakan *sequence diagram* yang bertujuan untuk menggambarkan skenario atau rangkaian langkah-langkah yang dilakukan sebagai respon dari suatu kejadian untuk menghasilkan *output*, menggunakan *class diagram* bertujuan menggambarkan struktur, hubungan dan deskripsi antar *class*.

b. Menggunakan *Entity Relationship Diagram* (ERD) yang bertujuan untuk memodelkan suatu data dan dapat dilakukan juga untuk perencanaan *database* serta hubungan data dalam pembuatan kamus data dengan tujuan memvalidasi diagram alir data untuk hal kelengkapan dan keakuratan data.

c. Menggunakan *Windows Navigation* (WND) untuk desain *interface*.

4. Melakukan Implementasi (*Implementation*)

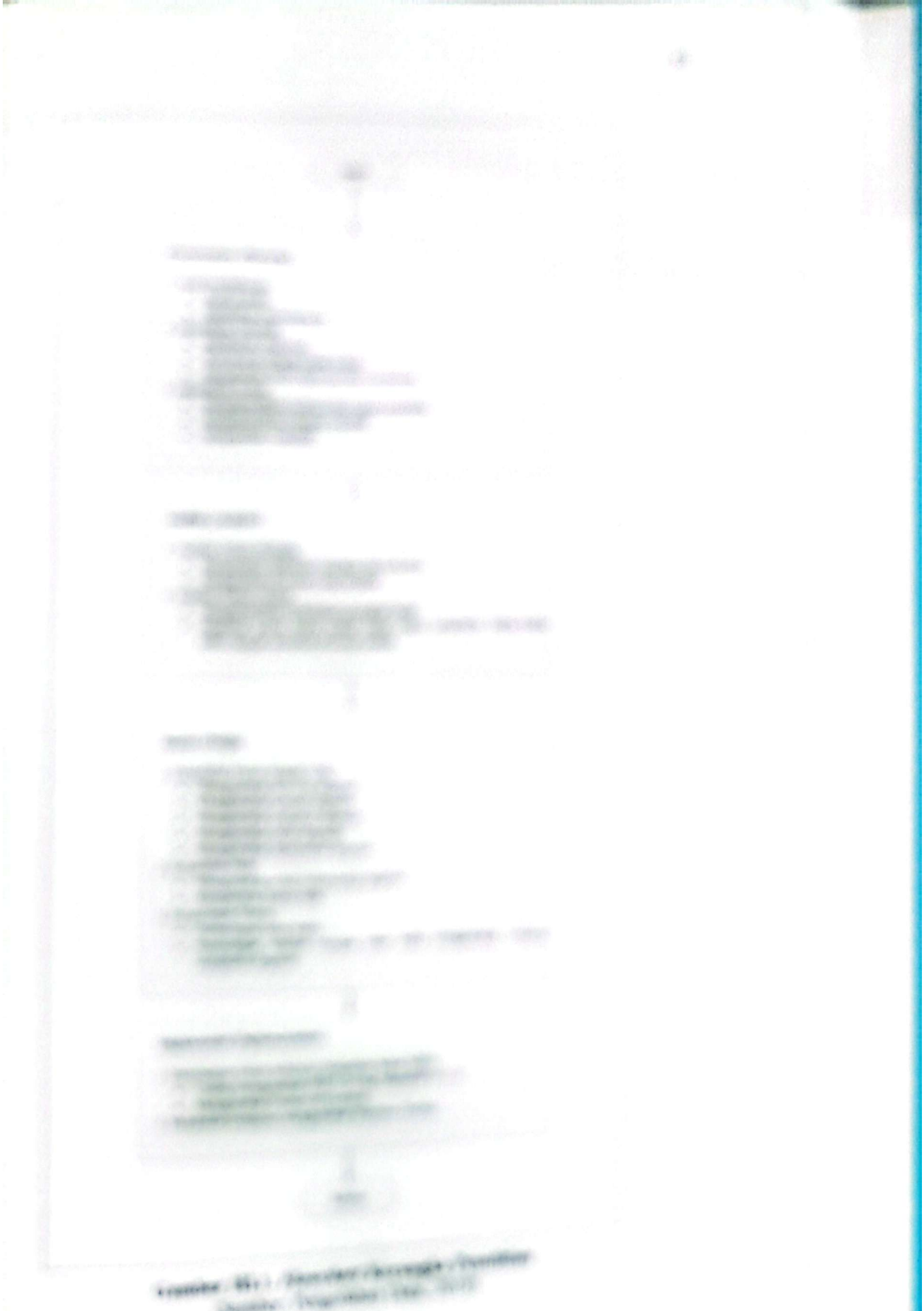
Pada tahap implementasi perangkat lunak yang di lakukan yaitu:

Dipindai dengan CamScanner

a. *Coding* pada aplikasi yang akan dibuat sesuai dengan kebutuhan yang ada dengan menggunakan Laravel 5.4.28 dan MariaDB 10.1.31 dalam menuliskan kode program dan menggunakan *framework* Laravel dengan bantuan Sublime Text Editor untuk pembuatan aplikasi sistem yang dibutuhkan. Selain itu menggunakan juga XAMPP, XAMPP merupakan sebuah *server* yang berdiri sendiri (*localhost*) yang bertujuan untuk mempermudah dalam mengoperasikan sistem operasi MariaDB 10.1.31 dan PHP 7.2.4.
b. Melakukan *testing* menggunakan *blackbox testing* bertujuan untuk pengujian yang dilakukan hanya mengamati hasil eksekusi melalui data uji dan memeriksa fungsional dari perangkat lunak. Analogi dari *blackbox testing* merupakan proses *testing* yang hanya dilakukan dari tampak luarnya saja tanpa tau apa yang ada didalamnya.
c. Membuat dokumentasi dalam bentuk laporan pengembangan Sistem Informasi Pengadaan Bahan-Bahan MRO Di PT Cikarang Perkasa Manufacturing Berbasis Web.

Berikut adalah Gambar III.1 *flowchart* kerangka penelitian untuk menyelesaikan masalah dalam Tugas Akhir ini:

Dipindai dengan CamScanner



MILIK PERPUSTAKAAN STMI
Membaca : Ibadah, Mengambil : Dosa

BAB IV

PENGUMPULAN DAN PENGOLAHAN DATA

4.1. Latar Belakang Perusahaan

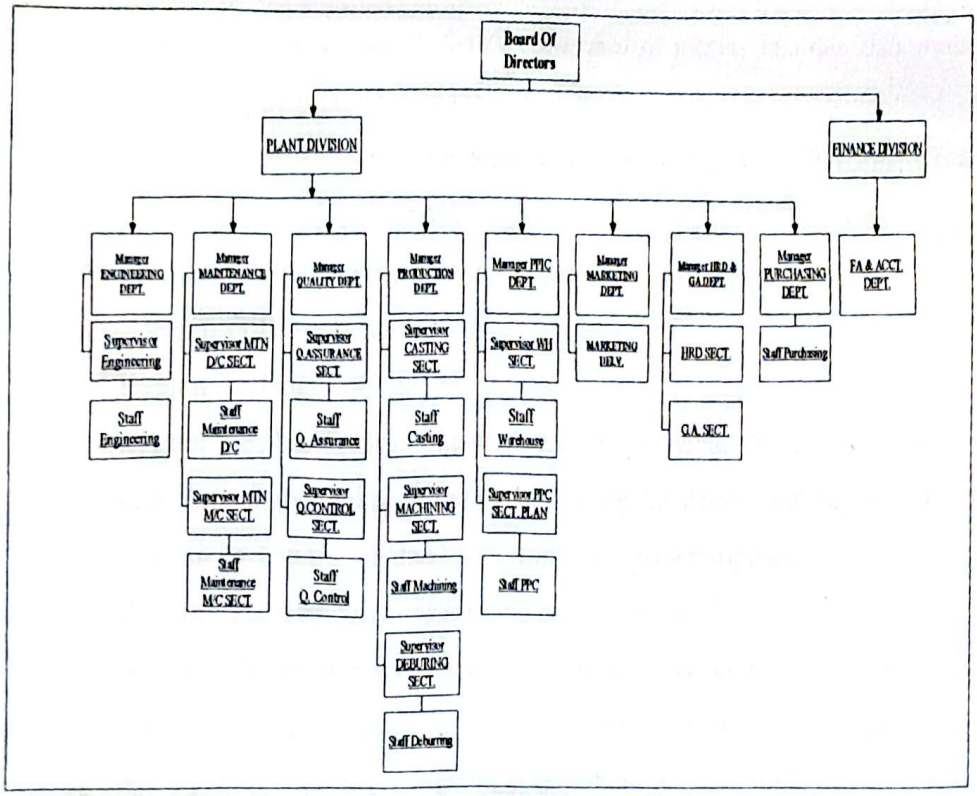
PT Cikarang Perkasa Manufacturing (CPM) didirikan sejak tahun 1996, bermula dari sebuah ide untuk memanfaatkan peluang yang ada dimana industri otomotif di Indonesia mulai berkembang. PT Cikarang Perkasa Manufacturing merupakan perusahaan manufaktur komponen otomotif yang bergerak di bidang *Die Casting & Machining* khusus *Aluminium ingot ADC 12*. Didirikan sejak tahun 1996, PT Cikarang Perkasa Manufacturing telah memproduksi sekitar 100 jenis komponen otomotif. Produk komponen otomotif yang dihasilkan di antaranya adalah *cover machine, body and cover* dan *car wiper holder* yang dikirim untuk beberapa perusahaan otomotif besar.

Perusahaan otomotif mulai melihat peluang untuk membuka manufaktur perakitan kendaraan seiring dengan pesatnya pertumbuhan bisnis otomotif di Indonesia. Hal ini menyebabkan peningkatan permintaan produk komponen dari konsumen PT Cikarang Perkasa Manufacturing. Struktur kepengurusan PT Cikarang Perkasa Manufacturing sangat mempengaruhi jalur bisnis penyebaran produknya ke perusahaan perusahaan konsumen di Indonesia.

Konsumen PT Cikarang Perkasa Manufacturing terdiri dari beberapa perusahaan otomotif besar yang produknya dikirim ke berbagai daerah termasuk produk ekspor. Berikut adalah konsumen utama PT CPM:

1. PT Suzuki Indomobil Motor
2. PT ASMO Indonesia
3. PT Mesin Isuzu Indonesia
4. PT Tjokro Bersaudara Komponenindo
5. PT NAKAKIN Indonesia
6. PT Yanmar Diesel Indonesia
7. PT General Motor Indonesia
8. PT Mikuni Indonesia

Dipindai dengan CamScanner



Gambar IV.1. Struktur Organisasi PT Cikarang Perkasa Manufacturing
(Sumber: PT Cikarang Perkasa Manufacturing, 2018)

4.1.1. Tugas dan Wewenang Divisi

Dengan adanya struktur organisasi perusahaan yang berjalan dengan baik maka berikut saya uraikan tugas serta wewenang dari setiap divisi.

1. *Maintenance*

   Departemen *Maintenance* bertugas untuk melakukan pemeliharaan terhadap mesin dan memperbaiki setiap *problem* yang terjadi pada mesin di lapangan. Berikut adalah tugas dari Departemen *Maintenance* di PT Cikarang Perkasa Manufacturing:

   a. Melakukan pengecekkan harian atas mesin, fasilitas, *mold* dan *tool* sesuai dengan prosedur.
   b. Melaksanakan pencegahan berdasarkan evaluasi bulanan, triwulan/semester dan tahunan sehingga fasilitas, mesin, dan *mold* dalam kondisi baik dan siap pakai.
   c. Melakukan pembaruandata mesin, fasilitas, *mold* dan *jig* setiap ada perubahan maupun perbaikan.

Dipindai dengan CamScanner

d. Menganalisa dan memperbaiki *abnormality* mesin, fasilitas dan *mold* termasuk potensial kecelakaan.

e. Memberikan usulan dan ide perbaikan dalam hal QCDSM *(Quality, Cost, Delivery, Safety, and Morale)*.

2. PPIC *(Production Planning Inventory Control)*

Departemen PPIC sendiri terdiri dari:

a. Bagian Gudang

Bagian ini bertugas mengawasi masuknya bahan baku, *spare parts*, keperluan habis-pakai, barang pendukung ke dalam gudang, membuat laporan barang masuk dan menyerahkan kepada pimpinan.

b. Bagian PPC *(Production Planning and Control)*

Bagian PPC bertugas sebagai pembuat rencana produksi dan membuat target produksi per harinya, serta membuat laporan ketersediaan barang di gudang, bertugas membuat laporan barang *finish good* (FG) berdasarkan stok, mengatur peletakan barang *FG*, mengawasi barang FG yang keluar dan mengatur proses menyiapkan barang dan mengirim barang yang dipesan oleh konsumen.

3. *Finance*

Bagian *Finance* bertugas untuk membayarkan tagihan kantor seperti listrik, telepon dan juga tagihan pembayaran kepada pemasok. Biasanya laporan pembayaran ini masuk melalui *accounting* untuk didata terlebih dahulu kemudian baru dibayar oleh *finance*, jadi terdapat *internal control* yang menjamin uang perusahaan digunakan semestinya dan pegawai *finance* juga bertugas untuk membuat *invoice* berdasarkan laporan dari bagian penjualan.

4. *Purchasing*

Bagian *purchasing* bertugas untuk membeli barang ke pemasok serta bertanggung jawab terhadap barang yang dibeli, dan membuat laporan kepada pimpinan. Berikut adalah tugas-tugas departemen *purchasing* PT Cikarang Perkasa Manufacturing:

Dipindai dengan CamScanner

a. Melakukan pembelian terhadap barang (bahan baku, *spare parts*, keperluan habis-pakai, barang pendukung) dan jasa sesuai dengan permintaan pembelian.
b. Mengontrol kedatangan barang dan penerimaan barang sesuai dengan persyaratan pembelian.
c. Melakukan seleksi dan evaluasi terhadap performa pemasok yang nantinya dijadikan acuan penilaian.
d. Melakukan *monitoring* terhadap pencapaian performa proses pembelian dan pengendalian pemasok.
e. Menyusun sasaran proses departemen pembelian, memastikan penyusunan dan pelaksanaan rencana.
f. Memeriksa seluruh permintaan pembelian *(Purchase Requisiton)* yang dikeluarkan oleh masing-masing bagian dan berkoordinasi dengan *user* dari bagian terkait.

4.2. Dokumen Pengadaan Bahan MRO

Dokumen yang digunakan dalam proses pengadaan bahan MRO pada Departemen *Maintenance* di PT Cikarang Perkasa Manufacturing diantaranya adalah:

1. *Request form planning*
2. *Purchase requisition*
3. *Purchase order*
4. *Purchase report computerized*
5. *Warehouse stock report computerized*

Pada analisis dokumen berikut ini akan dijelaskan dokumen apa saja yang digunakan dalam proses pengadaan bahan MRO di departemen *Maintenance* pada PT Cikarang Perkasa Manufacturing terdiri dari dokumen masukan dan keluaran.

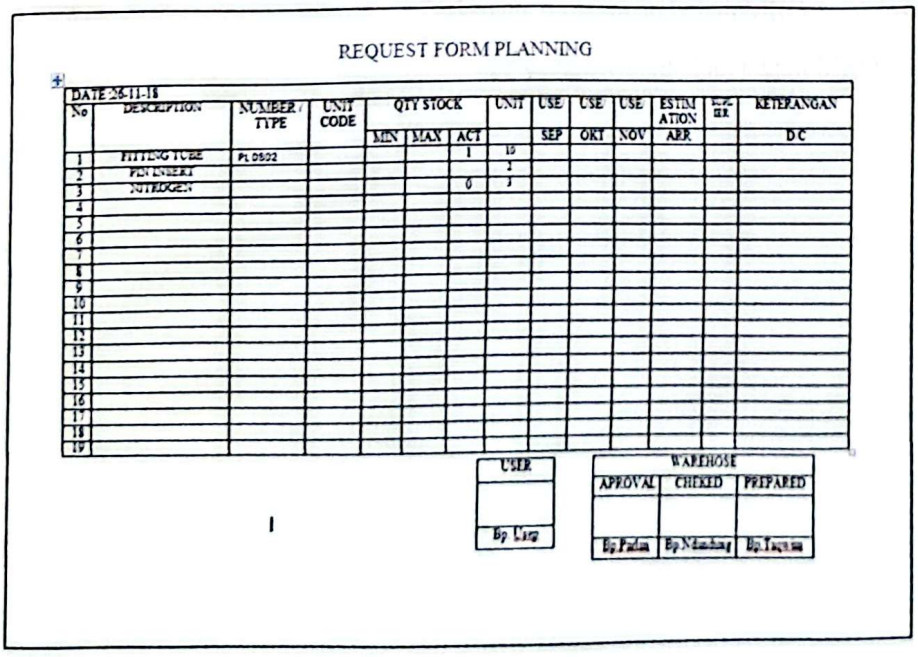
Dipindai dengan CamScanner

4.2.1. Dokumen Masukan

Dokumen masukan berfungsi untuk menjelaskan mengenai semua dokumen *input* yang digunakan pada proses pengadaan bahan-bahan MRO. Berikut adalah bentuk dokumen masukan yang ada di PT Cikarang Perkasa Manufacturing:

1. *Request Form Planning*

   Request Form Planning adalah dokumen yang berfungsi untuk pengajuan perencanaan pengadaan barang dari *Warehouse* diajukan kepada *user* yaitu Departemen *Maintenance.* Sebelum dokumen ini diajukan, *warehouse* akan melakukan pengecekan *stock* dan *limit stock* bahan-bahan MRO pada *warehouse stock report computerized.*


REQUEST FORM PLANNING

DATE 26-11-18

No	DESCRIPTION	NUMBER / TYPE	UNIT CODE	QTY STOCK			UNIT	USE	USE	USE	ESTIMATION		KETERANGAN
				MIN	MAX	ACT		SEP	OKT	NOV	ARR		DC
1	FITTING TUBE	PL 0502				1	15						
2	PEN ENERJI						2						
3	NITROGEN					0	3						
4													
5													
6													
7													
8													
9													
10													
11													
12													
13													
14													
15													
16													
17													
18													
19													

USER

Bp Ucup

WAREHOSE

APPROVAL	CHEKED	PREPARED
Bp Fadlan	Bp Nandang	Bp Tasman

Gambar IV.2. ***Request Form Planning***
(Sumber: Hasil Pengamatan, 2018)

Keterangan:

a. Asal : Departemen *Warehouse*
b. Tujuan : Departemen *Maintenance*
c. Frekuensi : Dibutuhkan ketika bahan MRO sudah habis
d. Kolom-kolom :
   1) No :Nomor urut bahan MRO
   2) *Description* :Nama bahan MRO

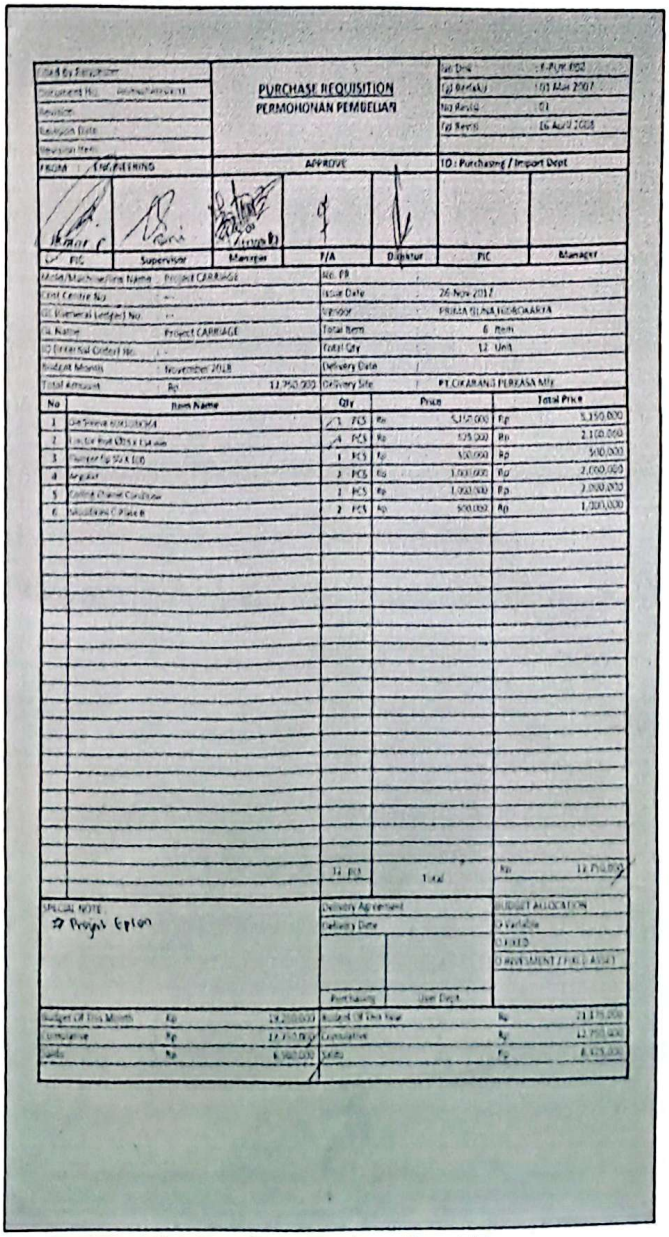
Dipindai dengan CamScanner

3) *Number/Type* :Tipe bahan MRO
4) Unit/Code :Nomor PO
5) *Qty Stock* :
   - *Min* :Jumlah minimum bahan MRO yang seharusnya ada di *warehouse*
   - *Max* :Jumlah maksimum bahan MRO yang seharusnya ada di *warehouse*
   - *Act* :Jumlah bahan MRO saat ini di *warehouse*
6) *Unit* :Jumlah bahan MRO yang di minta
7) *Use* :Perkiraan jumlah penggunaan bahan MRO selama 3 bulan kedepan
8) *Estimation* :Perkiraan kedatangan bahan MRO
9) *Supplier* :*Supplier* yang menyediakan bahan MRO

Dipindai dengan CamScanner

2. *Purchase Requisition*

Purchase Requisition (PR) adalah dokumen yang berfungsi untuk pengajuan pembelian barang dari *User Department*. PRakan diproses oleh Departemen *Purchasing* setelah mendapatkan persetujuan berupa tanda tangan dari pihak*General Manager* dan *Finance.*



PURCHASE REQUISITION
PERMOHONAN PEMBELIAN

No	Item Name	Qty	Price	Total Price
1		3 PCS	Rp	Rp
2		4 PCS	Rp	Rp
3		1 PCS	Rp	Rp
4		2 PCS	Rp	Rp
5		1 PCS	Rp	Rp
6		2 PCS	Rp	Rp

Gambar IV.3. *Purchase Requisition*
(Sumber: Hasil Pengamatan, 2018)

Keterangan:

a. Asal : Departemen *Maintenance*

b. Tujuan : Departemen *Finance, Purchasing*

Dipindai dengan CamScanner

c. Frekuensi : Dibutuhkan untuk pengajuaan pembelian

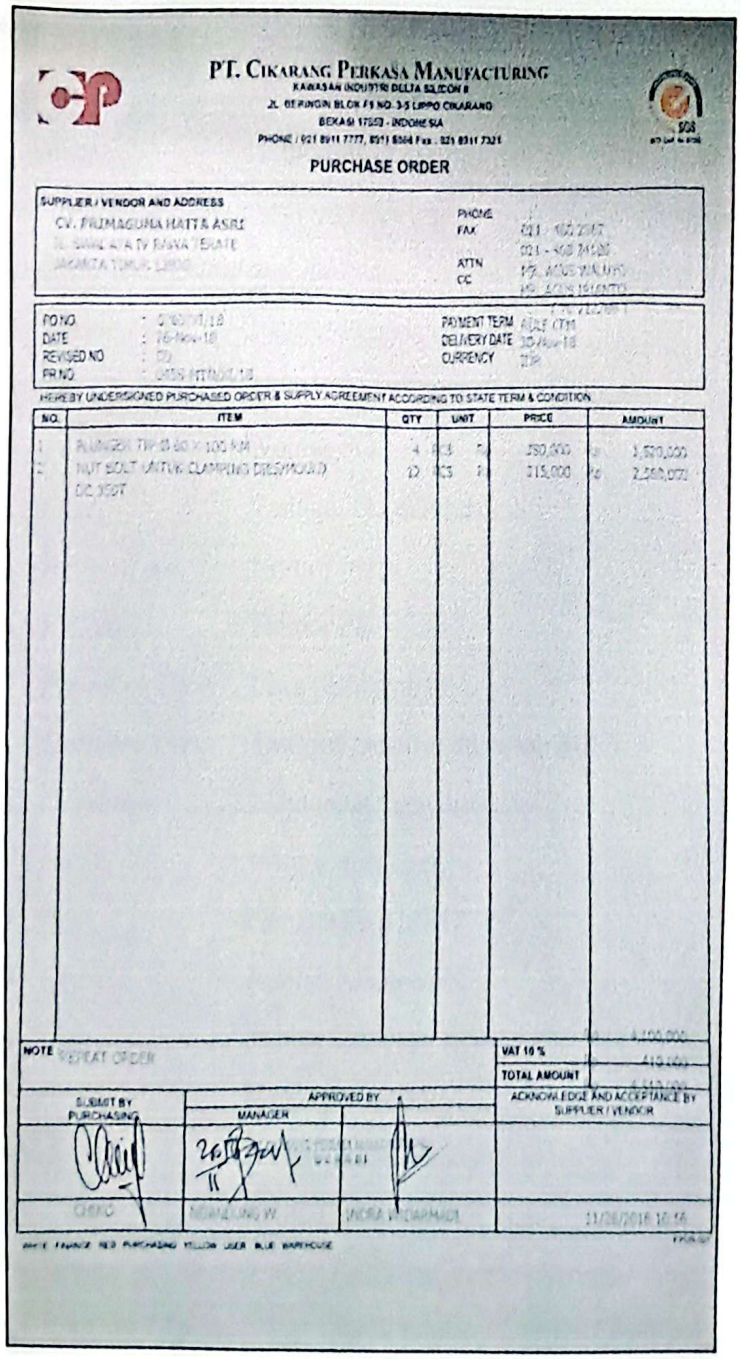
d. Kolom-kolom :

1) No PR :Nomor *Purchase Requisition*
2) *Issue Date* :Tanggal pengajuan PR
3) *Vendor* :Pemasok bahan MRO
4) Total *Item* :Jumlah jenis *item* bahan MRO
5) *Total Qty* :Jumlah seluruh bahan MRO
6) *Delivery Date* :Tanggal pengiriman bahan MRO
7) *Delivery Site* :Tanggal kedatangan bahan MRO
8) *No* :Nomor urut bahan MRO yang dipesan
9) *Item Name* :Namabahan MRO
10) *Qty* :Jumlah bahan MRO
11) *Price* :Harga bahan MRO/satuan
12) *Total Price* :Harga bahan MRO/satuan dikali *Qty*
13) Total :Total dari *Total Price*
14) *Budget of this month* :*Budget* pengeluaran bulan ini
15) *Cumulative* :Total pengeluaran bulan ini
16) Saldo :Sisa *budget* bulan ini
17) *Budget of this year* :*Budget* pengeluaran tahun ini
18) *Cumulative* :Total pengeluaran tahun ini
19) Saldo :Sisa *budget* tahun ini

3. *Purchase Order*

Purchase Order (PO) merupakan *Purchase Requisition* yang telah disetujui oleh *Finance* dan GM. PO berisi produk yang telah dibeli, berapa *quantity* yang diminta dan *total amount* dari pembelian tersebut. PO dibuat oleh *Purchasing* dan dikirim ke *vendor* sebagai bukti pembelian dari pihak perusahaan.

Dipindai dengan CamScanner



PT. CIKARANG PERKASA MANUFACTURING
KAWASAN INDUSTRI DELTA SILICON II
JL. GERINGIN BLOK F5 NO. 3-5 LIPPO CIKARANG
BEKASI 17550 - INDONESIA
PHONE / 021 8911 7777, 8911 8366 Fax. 021 8911 7321

PURCHASE ORDER

SUPPLIER / VENDOR AND ADDRESS
CV. PRIMAGUNA HATTA ASRI
JL. RAYA TV BARAT BLOK
JAKARTA TIMUR, DKI

PHONE
FAX
ATTN
CC

PO NO
DATE
REVISED NO
PR NO

PAYMENT TERM
DELIVERY DATE
CURRENCY

HEREBY UNDERSIGNED PURCHASED ORDER & SUPPLY AGREEMENT ACCORDING TO STATE TERM & CONDITION

NO.	ITEM	QTY	UNIT	PRICE	AMOUNT
1	RUNNER TIP Ø 60 X 105 MM	4	PCS	350,000	1,400,000
2	NUT BOLT UNTUK CLAMPING DIES(MOLD) DC 350T	12	PCS	315,000	3,780,000

NOTE: REPEAT ORDER

VAT 10 %
TOTAL AMOUNT

SUBMIT BY PURCHASING	APPROVED BY MANAGER		ACKNOWLEDGE AND ACCEPTANCE BY SUPPLIER / VENDOR
CHRIS			

WHITE : FINANCE RED : PURCHASING YELLOW : USER BLUE : WAREHOUSE

Gambar IV.4. *Purchase Order*
(Sumber: Hasil Pengamatan, 2018)

Purchase order ada 4 rangkap ditujukan kepada *vendor, warehouse, finance*, dan satu disimpan oleh bagian *purchasing*. Ketika barang datang, *warehouse* perlu memastikan barang yang datang sesuai dengan jumlah *quantity* pada *purchase order* dan surat jalan dari *vendor*. Begitu pula *purchasing* akan memastikan barang yang datang sesuai dengan *purchase order*.

Keterangan:

Dipindai dengan CamScanner

a. Asal : Departemen *Purchasing*

b. Tujuan : *Vendor*

c. Frekuensi : Dibutuhkan ketika membeli bahan MRO

d. Kolom-kolom :

1) *Supplier/Vendor and Address* :Nama *Vendor*
2) *Phone Fax* :Nomor telepon*fax vendor*
3) CC :Nama *vendor* yang dapat dihubungi
4) *PO No* :Nomor PO
5) *Date* :Tanggal PO dikeluarkan
6) *Revised No* :Nomor revisi
7) *PR No* :Nomor PR
8) *Payment Term* :Cara pembayaran
9) *Delivery Date* :Tanggal pengiriman bahan MRO
10) *Currency* :Mata uang yang dipakai
11) *No* :Nomor urut *item*
12) *Item* :Nama bahan MRO
13) *Qty* :Jumlah bahan MRO
14) *Unit* :Satuan unit yang dipakai
15) *Price* :Harga per satuan
16) *Amount* :Jumlah harga per *item*
17) *Total Amount* :Jumlah harga seluruh item
18) *Submit by Purchasing* :Tanda tangan *Purchasing*
19) *Approved by* :Tanda tangan *Manager Maintenance*, GM
20) *Acknowledge and Acceptance by Supplier/Vendor* :Tanda tangan *supplier*

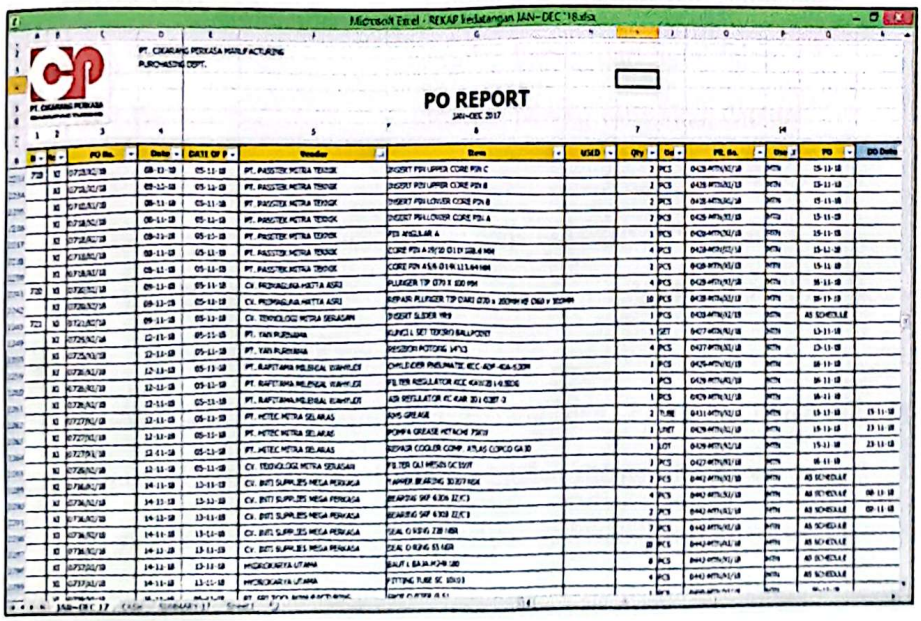
4.2.2. Dokumen Keluaran

Dokumen masukan berfungsi untuk menjelaskan mengenai semua dokumen *output* yang digunakan pada proses pengadaan bahan-bahan MRO. Berikut adalah bentuk dokumen keluaran yang ada di PT Cikarang Perkasa Manufacturing:

Dipindai dengan CamScanner

1. *Purchase Report Computerized*

Rekap dokumen *Purchase Order* disimpan dalam dokumen *PO Report* dimana Departemen *Purchasing* dapat dengan mudah melacak PO yang sudah *closed* dan masih *Open*, maksudnya *closed PO* berarti barang yang dipesan ke pihak *vendor* telah diterima sesuai dengan permintaan oleh pihak CPM sedangkan PO yang masih berstatus *open* berarti barang belum diterima.


PT. CHAROENG POKPHAND MANUFACTURING
PURCHASING DEPT.

PO REPORT

No	PO No.	Date	DATE OF PR	Vendor	Item	USED	Qty	Unit	PR No.	Unit 2	PO	DO Date
729	10	08-11-18	05-11-18	PT. PASSTEK MITRA TEKNIK	INSERT PIN UPPER CORE PIN C		2	PCS	0428-MTK/XI/18	MTK	15-11-18	
	10	08-11-18	05-11-18	PT. PASSTEK MITRA TEKNIK	INSERT PIN UPPER CORE PIN B		2	PCS	0428-MTK/XI/18	MTK	15-11-18	
	10	08-11-18	05-11-18	PT. PASSTEK MITRA TEKNIK	INSERT PIN LOWER CORE PIN B		2	PCS	0428-MTK/XI/18	MTK	15-11-18	
	10	08-11-18	05-11-18	PT. PASSTEK MITRA TEKNIK	INSERT PIN LOWER CORE PIN A		2	PCS	0428-MTK/XI/18	MTK	15-11-18	
	10	08-11-18	05-11-18	PT. PASSTEK MITRA TEKNIK	PIN ANGULAR A		1	PCS	0428-MTK/XI/18	MTK	15-11-18	
	10	08-11-18	05-11-18	PT. PASSTEK MITRA TEKNIK	CORE PIN A 19/10 D19 L18.4 HSS		4	PCS	0428-MTK/XI/18	MTK	15-11-18	
	10	08-11-18	05-11-18	PT. PASSTEK MITRA TEKNIK	CORE PIN A 6/8 D14 L13.64 HSS		1	PCS	0428-MTK/XI/18	MTK	15-11-18	
730	10	09-11-18	05-11-18	CV. INDRAGUNA MITRA ASRI	PLUNGER TIP D70 X 100 HSS		4	PCS	0428-MTK/XI/18	MTK	16-11-18	
	10	09-11-18	05-11-18	CV. INDRAGUNA MITRA ASRI	REPAIR PLUNGER TIP D.ME D70 x 100HSS HT CNGI x 100HSS		10	PCS	0428-MTK/XI/18	MTK	19-11-18	
731	10	09-11-18	05-11-18	CV. TEKNOLOGI MITRA SEJAHTERA	INSERT SLIDER 1899		1	PCS	0428-MTK/XI/18	MTK	AS SCHEDULE	
	10	12-11-18	05-11-18	PT. YAN PURNAMA	KUNCI L SET TEKIRO BALLPOINT		1	SET	0427-MTK/XI/18	MTK	13-11-18	
	10	12-11-18	05-11-18	PT. YAN PURNAMA	PISTON MOTORE 1HT13		4	PCS	0427-MTK/XI/18	MTK	13-11-18	
	10	12-11-18	05-11-18	PT. RAPITAMA MILENIAL EXAMPLES	CYLINDER PNEUMATIC ISO 6DF-40A-630H		1	PCS	0429-MTK/XI/18	MTK	16-11-18	
	10	12-11-18	05-11-18	PT. RAPITAMA MILENIAL EXAMPLES	FR TFR REGULATOR ACC KAH3E1-14-3EDG		1	PCS	0429-MTK/XI/18	MTK	16-11-18	
	10	12-11-18	05-11-18	PT. RAPITAMA MILENIAL EXAMPLES	AIR REGULATOR RC KAH 3E1-02BT-2		1	PCS	0429-MTK/XI/18	MTK	16-11-18	
	10	12-11-18	05-11-18	PT. HITEC MITRA SELARAS	KMS GREASE		2	TUBE	0431-MTK/XI/18	MTK	13-11-18	13-11-18
	10	12-11-18	05-11-18	PT. HITEC MITRA SELARAS	POMPA GREASE HITACHI 75KG		1	UNIT	0429-MTK/XI/18	MTK	13-11-18	23-11-18
	10	12-11-18	05-11-18	PT. HITEC MITRA SELARAS	REPAIR COOLER COMP. ATLAS COPCO GA30		1	LOT	0429-MTK/XI/18	MTK	15-11-18	23-11-18
	10	12-11-18	05-11-18	CV. TEKNOLOGI MITRA SEJAHTERA	PE TER GLI MESIN SC150W		1	PCS	0427-MTK/XI/18	MTK	16-11-18	
	10	14-11-18	13-11-18	CV. INTI SUPPLIES MEGA PERKASA	* ANKER BEARING 30307 HSA		2	PCS	0442-MTK/XI/18	MTK	AS SCHEDULE	
	10	14-11-18	13-11-18	CV. INTI SUPPLIES MEGA PERKASA	BEARING SKF 6306 2Z/C3		4	PCS	0442-MTK/XI/18	MTK	AS SCHEDULE	08-11-18
	10	14-11-18	13-11-18	CV. INTI SUPPLIES MEGA PERKASA	BEARING SKF 6303 2Z/C3		2	PCS	0442-MTK/XI/18	MTK	AS SCHEDULE	09-11-18
	10	14-11-18	13-11-18	CV. INTI SUPPLIES MEGA PERKASA	SEAL O RING 238 NBR		7	PCS	0442-MTK/XI/18	MTK	AS SCHEDULE	
	10	14-11-18	13-11-18	CV. INTI SUPPLIES MEGA PERKASA	SEAL O RING 85 NBR		10	PCS	0442-MTK/XI/18	MTK	AS SCHEDULE	
	10	14-11-18	13-11-18	CV. INTI SUPPLIES MEGA PERKASA	SEAL O RING 85 NBR		6	PCS	0442-MTK/XI/18	MTK	AS SCHEDULE	
	10	14-11-18	13-11-18	MITRAGUNA UTAMA	SEAL I 65 H 90 HB 180		4	PCS	0442-MTK/XI/18	MTK	AS SCHEDULE	
	10	14-11-18	13-11-18	MITRAGUNA UTAMA	FITTING TUBE SC 10X10		4	PCS	0442-MTK/XI/18	MTK	AS SCHEDULE	

Gambar IV.5. *Purchase Order Report*
(Sumber: Hasil Pengamatan, 2018)

Keterangan:

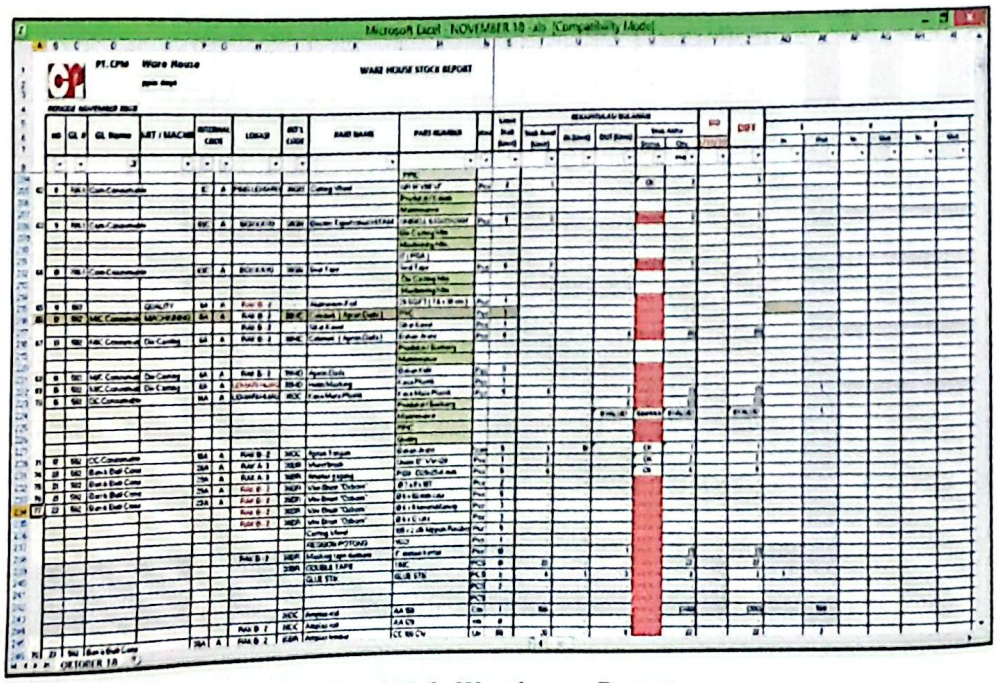
a. Asal : Departemen *Purchasing*
b. Tujuan : Departemen *Finance*
c. Frekuensi : Setiap satu bulan sekali diakhir bulan
d. Kolom-kolom :
   1) No :Nomor urut PO
   2) *Month* : Bulan pengajuan PO
   3) *Date* : Tanggal pengajuan PO
   4) *Date of PR* : Tanggal pengajuan PR
   5) *Vendor* : *Vendor* bahan MRO
   6) *Item* : Nama *item*
   7) *Used* : Tujuan penggunaan

Dipindai dengan CamScanner

8) *Qty* : Jumlah *item*
9) *Unit* : Satuan unit yang digunakan
10) *PR No.* : Nomor PR
11) *User* : Departemen yang menggunakan
12) *PD* : Tanggal kedatangan barang
13) *DO Date* : Tanggal *order* barang dari Surat Jalan
14) *DO No* : Nomor *order* barang dari Surat Jalan
15) *Qty* : Jumlah barang dari Surat Jalan
16) *OS* : Jumlah barang yang akan datang
17) *Status*
    - *Open* :*Item* pada PO telah datang semua
    - *Close* :*Item* pada PO masih ada yang belum datang

2. *Warehouse Stock Report Computerized*

Rekap dokumen Slip Permintaan Barang dan *Purchase Order* dimana Departemen *Warehouse* dapat dengan mudah mengetahui *stock* yang masih ada atau sudah memasuki *limit stock*. Apabila barang sudah *limit stock*, maka Departemen *Warehouse* akan mengajukan *Request Form Planning* kepada *user*.



Gambar IV.6. *Warehouse Report*
(Sumber: Hasil Pengamatan, 2018)

Dipindai dengan CamScanner

Keterangan:

a. Asal : Departemen *Warehouse*

b. Tujuan : Semua Departemen

c. Frekuensi : Dibutuhkan sebagai laporan ketersediaan

d. Kolom-kolom :

1) No :Nomor urut bahan MRO

2) *GL #* :Kode bahan MRO

3) *GL Name* :Jenis penggunaan

4) Unit/Code :Nomor PO

5) *Machine* :Digunakan pada mesin

6) *Internal Code* :Kode internal *warehouse*

7) Lokasi :Lokasi *item* di *warehouse*

8) *Part Name* :Nama *part*

9) *Part Number* :Nomor *part*

10) Satuan :Jenis satuan yang digunakan

11) *Limit Stock* :Batas minimum *stock* di *warehouse*

12) Rekapitulasi Bulanan

- *Stock Awal* :Jumlah awal
- *In* :Jumlah barang yang datang
- *Out* :Jumlah barang yang keluar
- *Stock* akhir
  - Status :OK atau *Order*(ajukan pemesanan)
  - *Qty* :Jumlah *stock* akhir per bulan

4.3. Analisis Proses Pengadaan Bahan MRO yang Sedang Berjalan

Flowmap menjelaskan urutan-urutan dari prosedur-prosedur yang ada di dalam sistem yang berada pada PT Cikarang Perkasa Manufacturing. Bagan alir menunjukkan apa yang dikerjakan di sistem, menunjukkan arus dari pekerjaan secara keseluruhan dari sistem termasuk dokumen-dokumennya yang terkait dan digunakan untuk alat bantu komunikasi dan untuk dokumentasi. Bagan alir ini digunakan terutama untuk alat bantu komunikasi dan untukdokumentasi.Analisis

Dipindai dengan CamScanner

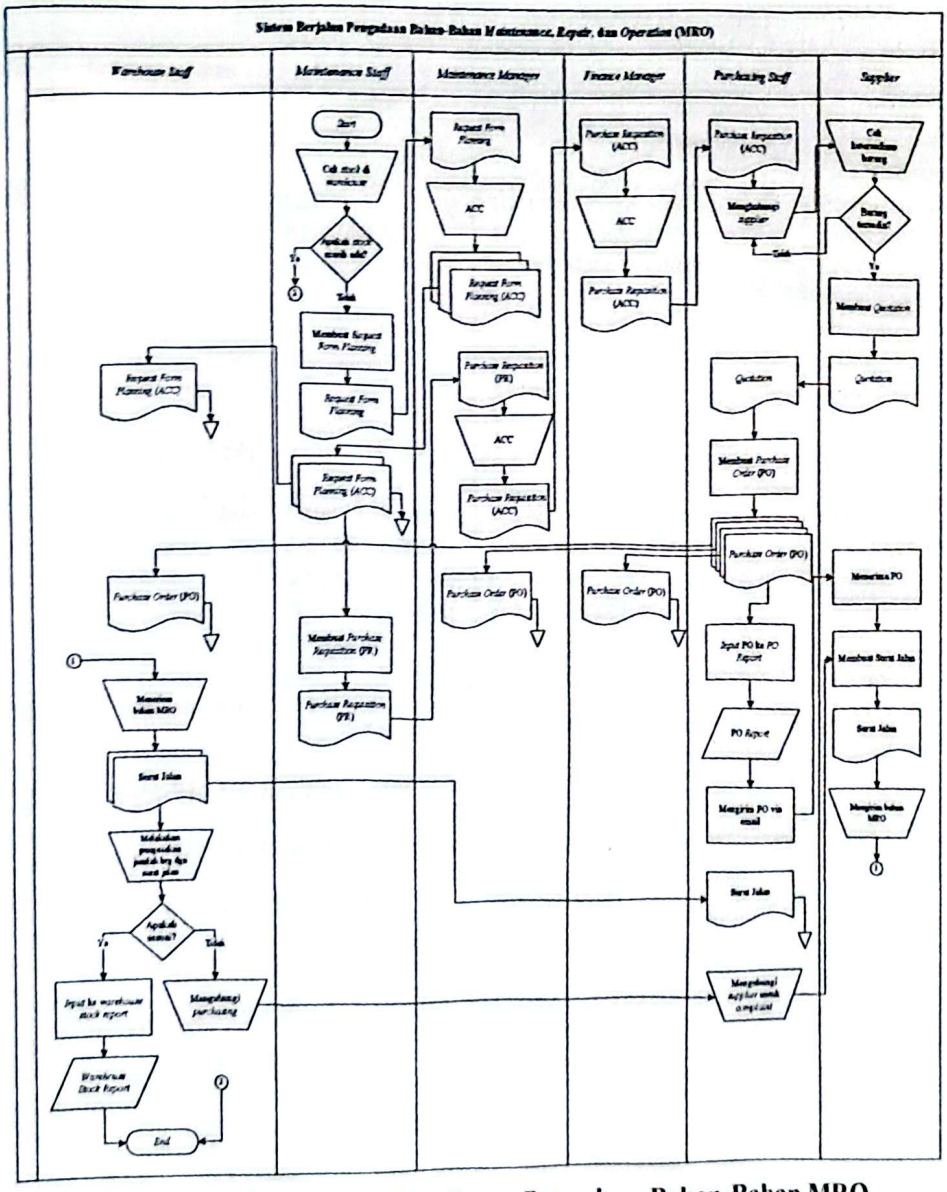
sistem yang sedang berjalan digambarkan dengan *flowmap* yang bertujuan untuk memudahkan pemahaman terhadap sistem tersebut dan menyederhanakan rangkaian proses atau prosedur dengan menggambarkannya secara ringkas, jelas, dan logis.

4.3.1. *Flowmap* Proses Pengadaan Bahan MRO yang Sedang Berjalan

Analisis sistem yang sedang berjalan digambarkan dengan *flowchart* dengan tujuan untuk memudahkan pemahaman terhadap sistem tersebut dan menyederhanakan rangkaian proses atau prosedur dengan menyajikannya secara ringkas, jelas, dan logis. Pada proses pengadaan bahan-bahan MRO, Departemen *Purchasing* memiliki peran besar untuk memastikan bahwa proses produksi berjalan dengan lancar tanpa ada gangguan potensi adanya *out of stock* dan *blank material.* Sistem yang berjalan adalah sebagai berikut:

1. *Maintenance Staff* mengajukan *Request Form Planning* kepada *Maintenance Manager*.
2. *Maintenance Manager* menyetujui RFP.
3. *Maintenance Manager* membuat PR menggunakan Ms. Excel.
4. *Maintenance Manager* mengajukan PR kepada *Finance Manager*, dan *Purchasing Manager.*
5. *Purchasing Manager* mengecek ketersediaan barang yang tercantum pada PR ke *supplier* melalui telepon dan email.
6. Jika barang tersedia, *supplier* mengirimkan *quotation* melalui email dan fax kepada *Purchasing Manager.*
7. *Purchasing Manager* membuat PO menggunakan Ms. Excel dan diajukan kepada *supplier* melalui email.
8. *Supplier* mengirim barang beserta surat jalan.
9. *Warehouse Staff* menerima barang dan surat jalan.

Dipindai dengan CamScanner



Gambar IV.7. Aliran Analisis Proses Pengadaan Bahan-Bahan MRO
(Sumber: Hasil analisis, 2018)

Dipindai dengan CamScanner

Tabel V.1. Analisis Kebutuhan Sistem (lanjutan)

No.	Permasalahan	Kebutuhan *User*	Solusi	Kebutuhan Sistem	
				Functional Requirement	***Non-Functional Requirement***
3.	Karyawan *Purchasing* mengalami kesusahan pencarian *Purchase Order* dikarenakan banyaknya tumpukan file yang ada.	Sistem yang memiliki *database* sebagai media penyimpanan seluruh data	Merancang *database* bersifat *data oriented* dan memenuhi kebutuhan sistem	- Membuat *Purchase Order*	- Pencarian data *Purchase Order* dapat dilakukan dengan kata kunci, No PR, *Issue Date* PR, No PO.
4.	Keamanan data yang lemah.	Sistem yang aman digunakan sesuai dengan hak aksesnya masing-masing	Membangun suatu sistem informasi yang dapat mengelola hak akses	- Melakukan *login*	- *Login* dengan memasukkann *email* dan *password*

(Sumber: Hasil Analisis, 2019)

5.1.1. *Functional Requirement*

Functional Requirement dari sistem informasi pengadaan bahan-bahan MRO yang diusulkan antara lain:

1. *Warehouse Staff*, *Maintenance Staff*, *Maintenance Manager*, *Finance Manager* dan *Purchasing Manager* melakukan *login*.
2. *Maintenance Staff* meng-*input Request Form Planning*.
3. *Maintenance Manager* meng-*update* status RFP.
4. *Maintenance Staff* melakukan *input* data *Purchase Requisition*.
5. *Maintenance Manager* meng-*update* status PR.
6. *Finance Manager* meng-*update* status PR yang telah disetujui *Maintenance Manager*.
7. *Purchasing Manager* melakukan pencarian data *Purhase Requisition*.
8. *Purchasing Manager* melakukan *input* data *Purchase Order*.
9. *Warehouse Staff* melakukan *update stock* bahan MRO yang masuk dan keluar.
10. *Maintenance Manager download* file laporan PR.
11. *Purchasing Manager download* file PO dan laporan PO.

Dipindai dengan CamScanner

5.1.2. *Non-Functional Requirement*

Non Functional Requirement dari sistem informasi pengadaan bahan-bahan MRO yang diusulkan antara lain:

1. *Login* dengan memasukkan *username* dan *password*.
2. Melakukan *input* data RFP dan menampilkan data RFP seperti No RFP, *Supplier*, *Part Name*, *min qty*, *max qty*, *act qty*, *qty request*, *month*, dan *use*.
3. Melakukan *update status* RFP dengan memilih *status Approved*
4. Melakukan *input* data PR dan menampilkan data PR seperti No PR, No RFP, *supplier*, *part name*, *qty*, *price*, *issue date*, dan *delivery date*.
5. Melakukan *update status* PR dengan memilih *status Approved*
6. Melakukan pencarian data PR dan menampilkan hasil pencarian dengan memasukkan kata kunci seperti No PR, dan *Issue Date*.
7. Melakukan *input* data PO seperti No PR, No PO, *supplier*, *part name*, *qty*, *price*, *issue date*, dan *delivery date*.
8. Melakukan *download file* laporan PR dengan memasukkan data *Start Date* dan *End Date*.
9. Melakukan *download file* PO terpilih.
10. Melakukan *download file* laporan PO dengan memasukkan data *Start Date* dan *End Date*.
11. Melakukan *update stock* bahan MRO dan menampilkan *warehouse stock* dengan *input data part name*, *qty*, *stock*, *limit stock*, *location*, dan *status*.

5.2. Prosedur Sistem Informasi Usulan Pengadaan Bahan MRO

Prosedur sistem informasi pengadaan bahan-bahan MRO yang diusulkan sebagai bentuk perbaikan sistem informasi di PT Cikarang Perkasa Manufacturing melalui tahapan-tahapan sebagai berikut:

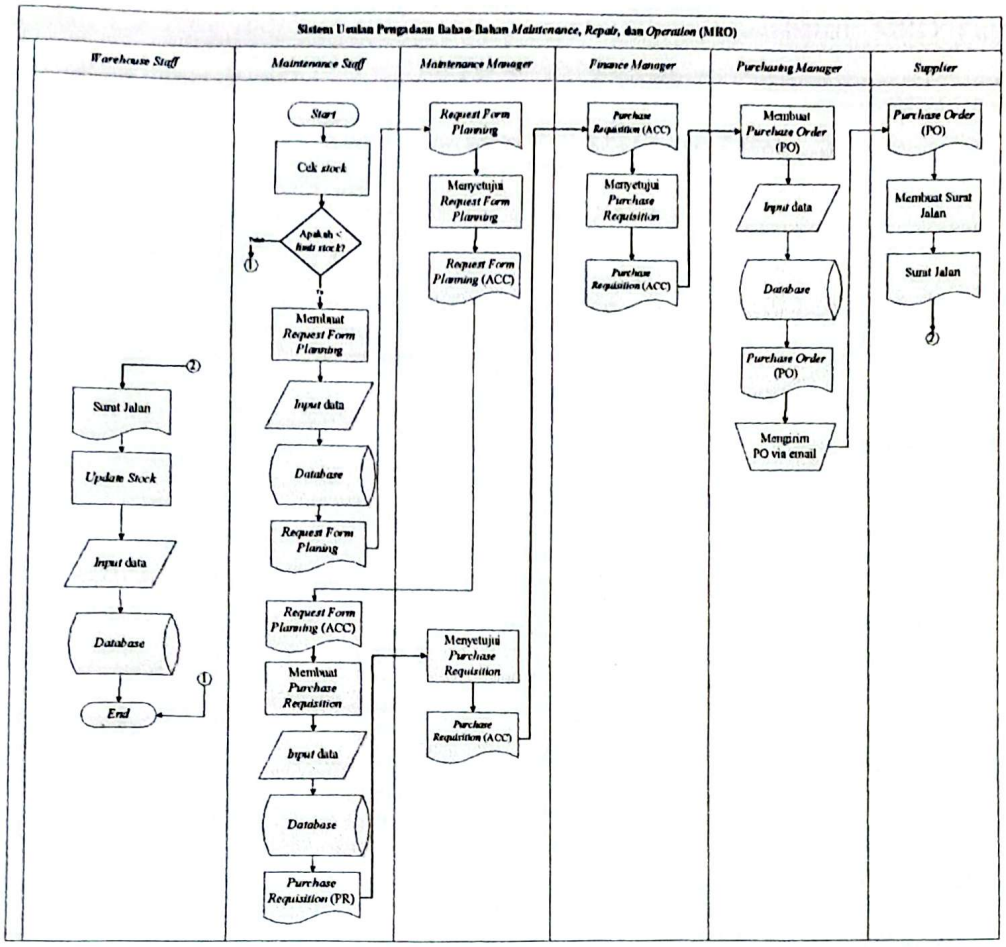
1. *Maintenance Staff* membuat *Request Form Planning* untuk merencanakan penggunaan bahan MRO kepada *Maintenance Manager*.
2. *Maintenance Manager* menyetujui RFP.
3. *Maintenance Staff* memproses pengajuan RFP menjadi PR.
4. *Maintenance Manager* menyetujui PR.

Dipindai dengan CamScanner

5. *Finance Manager* menyetujui PR setelah disetujui oleh *Maintenance Manager*.
6. *Purchasing Manager* mencari PR dan membuat *Purchase Order*.
7. *Purchase Order* kemudian dikirimkan kepada *supplier* melalui email oleh *Purchasing Manager*.
8. *Supplier* mengirimkan bahan MRO yang dipesan beserta surat jalan.
9. *Warehouse Staff* menerima bahan MRO yang dikirim *supplier* dan memeriksa kesesuaian bahan MRO yang datang dengan surat jalan.
10. *Warehouse Staff update stock* bahan MRO di *warehouse stock*.
11. *Maintenance manager download file* Laporan PR.
12. *Purchasing Manager download file* PO dan Laporan PO.

Seperti yang dapat dilihat pada Gambar V.1. adalah *Flowmap* Sistem Informasi Usulan Pengadaan Bahan-Bahan MRO di PT Cikarang Perkasa Manufacturing sebagai berikut:

Dipindai dengan CamScanner



Gambar V.1. *Flowmap* Sistem Usulan Pengadaan Bahan MRO di PT CPM
(Sumber: Hasil Analisis, 2019)

5.3. Analisis Sistem Usulan

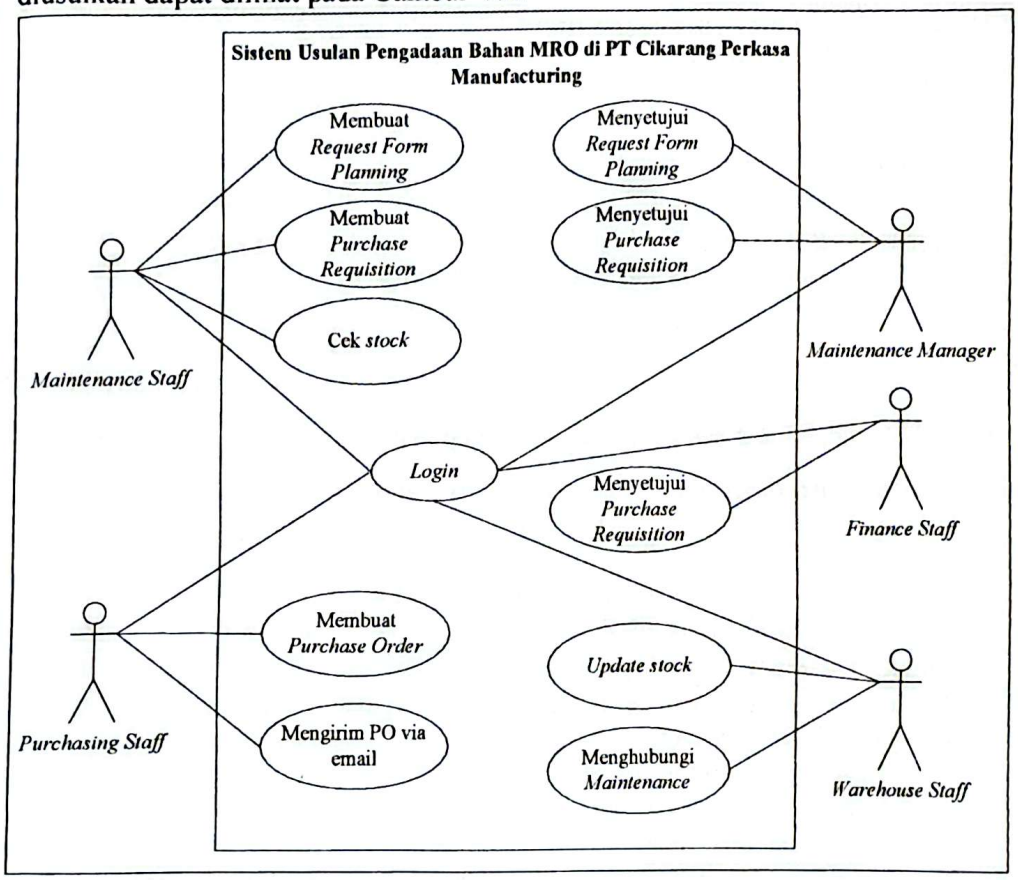
Pada tahap ini dilakukan analisis sistem dengan menggunakan UML (*Unified Modelling Language*) di antaranya adalah*Use Case Diagram*, *Activity Diagram*, *Sequence Diagram*, *Class Diagram*, dan *Deployment Diagram*.

5.3.1. *Use Case Diagram*

Use Case diagram berfungsi untuk mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem yang akan dibuat. Dengan kata lain *use case diagram* juga digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem dan siapa saja yang berhak menggunakan fungsi-fungsi itu. Berikut

Dipindai dengan CamScanner

adalah *use case diagram* sistem informasi pengadaan bahan-bahan MRO yang diusulkan dapat dilihat pada Gambar V.2.



Gambar V.2. *Use Case Diagram* Sistem Usulan
(Sumber: Hasil Analisis, 2019)

Penjelasan *Use Case Diagram* Sistem Informasi Pengadaan Bahan MRO pada PT Cikarang Perkasa Manufacturing diatas, dapat dijelaskan sebagai berikut:

1. Definisi Aktor

   Pendefinisian aktor pada *use case* Sistem Informasi Pengadaan Bahan MRO pada PT Cikarang Perkasa Manufacturing dapat dilihat pada Tabel V.2.

Dipindai dengan CamScanner

Tabel V.2. Definisi Aktor Sistem Usulan

No	Aktor	Definisi
1.	*Warehouse Staff*	Mendeskripsikan bagaimana *actor* mengecek jumlah bahan MRO dan *update stock* bahan MRO.
2.	*Maintenance Staff*	Mendeskripsikan bagaimana *actor* membuat RFP dan PR.
3.	*Maintenance Manager*	Mendeskripsikan bagaimana *actor* menyetujui RFP dan PR
4.	*Finance Manager*	Mendeskripsikan bagaimana *actor* menyetujui PR
5.	*Purchasing Manager*	Mendeskripsikan bagaimana *actor* membuat PO, dan mengirim PO ke *Supplier.*

(Sumber: Hasil Analisis, 2019)

2. Definisi *Use Case*

Definisi *Use Case* yang terdapat pada Sistem Pengadaan Bahan MRO PT. Cikarang Perkasa Manufacturing akan dijelaskan lebih rinci pada tabel berikut ini berikut ini:

Tabel V.3. Definisi *Use Case*

No	*Use Case*	Definisi *Use Case*
1.	*Login*	Mendeskripsikan bagaimana *actor* masuk sistem.
2.	Mengelola data master Bahan MRO	Mendefinisikan bagaimana *actor* menambahkan, memperbaharui, dan menghapus data bahan MRO.
3.	Membuat RFP	Mendefinisikan bagaimana *actor* membuat RFP
4.	Menyetujui RFP	Mendefinisikan bagaimana *actor* menyetujui RFP
5.	Membuat PR	Mendefinisikan bagaimana *actor* membuat PR
6.	Menyetujui PR	Mendefinisikan bagaimana *actor* menyetujui PR
7.	Membuat PO	Mendefinisikan bagaimana *actor* membuat PO
8.	Membuat *Warehouse Stock*	Mendefinisikan bagaimana *actor* mengelola *warehouse stock*
9.	Mengelola data master *Supplier*	Mendefinisikan bagaimana *actor* menambahkan data *supplier*, memperbaharui, dan menghapus.
10.	Mencetak PO	Mendefinisikan bagaimana *actor* mencetak PO
11.	Mencetak PO *Report*	Mendefinisikan bagaimana *actor* mencetak PO *Report*
12.	Mencetak PR *Report*	Mendefinisikan bagaimana *actor* mencetak PR *Report*

(Sumber: Hasil Analisis, 2019)

3. Skenario *Use Case*

Skenario *Use Case* yang terdapat pada Sistem Pengadaan Bahan MRO PT. Cikarang Perkasa Manufacturing akan dijelaskan lebih rinci dibawah ini:

Dipindai dengan CamScanner

Tabel V.4. Skenario *Use Case Login*

Use Case Name	*Login* ke Sistem Pengadaan Bahan MRO PT CPM	
Actor	*Warehouse, Maintenance, Finance, Purchasing*	
Brief Description	*Use case* ini mendeskripsikan bagaimana *actor login*	
Trigerring Event	-	
Flow of Events		
No	*Actor*	*System*
1.	*Actor* membuka *login form*	Sistem menampilkan *login form*
2.	*Actor* memasukkan *username* dan *password*	Sistem mengecek *username* dan *password*
3.	*Actor* masuk ke halaman utama	Jika sistem memvalidasi *username* dan *password* sesuai
4.	*Actor* kembali ke *login form*	Jika sistem memvalidasi *username* dan *password* tidak sesuai

(Sumber: Hasil Analisis, 2019)

Tabel V.5. Skenario *Use Case* Mengelola Data Master Bahan MRO

Use Case Name	Mengelola data master bahan MRO	
Actor	*Maintenance Staff, Maintenance Manager*	
Brief Description	*Use case* ini mendeskripsikan bagaimana *actor* mengelola data master bahan MRO	
Trigerring Event	-	
Flow of Events		
No	*Actor*	*System*
1.	*Actor* membuka menu Data Master Bahan MRO	Sistem menampilkan Data Master BahanMRO
2.	*Actor* klik New BahanMRO	Sistem menampilkan *Form New* BahanMRO
3.	*Actor* memasukkan data dan klik *Submit*	Sistem menyimpan ke *database*
4.	*Actor* klik *Delete*	Sistem menghapus data di *database*
5.	*Actor* klik *Edit*	Sistem menampilkan *Form Edit* BahanMRO

(Sumber: Hasil Analisis, 2019)

Tabel V.6. Skenario *Use Case* Membuat *Request Form Planning*

Use Case Name	Membuat *Request Form Planning*	
Actor	*Maintenance Staff*	
Brief Description	*Maintenance Staff* mengajukan RFP	
Trigerring Event	*Maintenance* membutuhkan bahan MRO	
Flow of Events		
No	*Actor*	*System*
1.	*Actor* mengklik menu *Request Form Planning*	Sistem menampilkan *Request Form Planning*
2.	*Actor* memasukkan data	Sistem menyimpan dalam *database*
3.	*Actor* klik *Create*	Sistem menyimpan dalam *database*

(Sumber: Hasil Analisis, 2019)

Dipindai dengan CamScanner

Tabel V.7. Skenario *Use Case* Menyetujui *Request Form Planning*

Use Case Name	Menyetujui RFP	
Actor	*Maintenance Manager*	
Brief Description	*Maintenance Manager* menyetujui RFP	
Trigerring Event	*Maintenance Staff* mengajukan RFP	
Flow of Events		
No	*Actor*	*System*
1.	*Actor* mengklik menu *Request Form Planning*	Sistem menampilkan Data *Request Form Planning*
2.	*Actor* klik *Approved*	Sistem menyimpan dalam *database*

(Sumber: Hasil Analisis, 2019)

Tabel V.8. *Use Case* Membuat *Purchase Requisition*

Use Case Name	Membuat PR	
Actor	*Maintenance Staff*	
Brief Description	*Maintenance* mengisi form PR sesuai data RFP	
Trigerring Event	*Maintenance Manager* menyetujui RFP	
Flow of Events		
No	*Actor*	*System*
1.	*Actor* mengklik RFP	Sistem menampilkan data RFP
2.	*Actor* mengklik tombol PR	Sistem menampilkan *Create* PR form
3.	*Actor* memasukkan data PR dan klik *Create*	Sistem menyimpan dalam *database*

(Sumber: Hasil Analisis, 2019)

Tabel V.9. *Use Case* Menyetujui *Purchase Requisition*

Use Case Name	Menyetujui PR	
Actor	*Maintenance Manager, Finance Manager*	
Brief Description	*Maintenance Manager, Finance Manager* menyetujui PR	
Trigerring Event	*Maintenance Manager* menerima PR *Maintenance Manager* menyetujui PR dan *Finance Manager* menerima PR	
Flow of Events		
No	*Actor*	*System*
1.	*Actor* mengklik PR	Sistem menampilkan data PR
2.	*Actor* klik *Approved*	Sistem menyimpan dalam *database*

(Sumber: Hasil Analisis, 2019)

Dipindai dengan CamScanner

Tabel V.10. Skenario *Use Case* Membuat *Purchase Order*

Use Case Name	Membuat *Purchase Order*	
Actor	*Purchasing Manager*	
Brief Description	*Purchasing* mengisi PO *form* sesuai data PR	
Trigerring Event	Adanya PR dari *Maintenance Manager*	
Flow of Events		
No	*Actor*	*System*
1.	*Actor* mengklik data PR	Sistem menampilkan data PR
2.	*Actor* mencari no PR di *field search PR*	Sistem menampilkan no PR yang dicari
3.	*Actor* mengklik tombol PO	Sistem menampilkan *Create* PO *form*
4.	*Actor* memasukkan data dan klik *Create*	Sistem menyimpan dalam database

(Sumber: Hasil Analisis, 2019)

Tabel V.11. Skenario *Use Case* Mencetak PO

Use Case Name	Mencetak PO	
Actor	*Purchasing Manager*	
Brief Description	*Maintenance* mencetak PO	
Trigerring Event	Pemesanan ke *Supplier*	
Flow of Events		
No	*Actor*	*System*
1.	*Actor* mengklik menu PO	Sistem menampilkan *Listing* PR
2.	*Actor* mengklik *print*	Sistem menghasilkan *output* po.pdf

(Sumber: Hasil Analisis, 2019)

Tabel V.12. Skenario *Use Case* Membuat PR *Report*

Use Case Name	Membuat PR *Report*	
Actor	*Maintenance Manager*	
Brief Description	*Maintenance* membuat PR *Report*	
Trigerring Event	*Monthly Report*	
Flow of Events		
No	*Actor*	*System*
1.	*Actor* mengklik menu PR	Sistem menampilkan *Listing* PR
2.	*Actor* memasukkan tanggal *start* dan tanggal *end* dan klik *import report*	Sistem menghasilkan *output* prreport.pdf

(Sumber: Hasil Analisis, 2019)

Dipindai dengan CamScanner

Tabel V.13. Skenario *Use Case* Membuat PO *Report*

Use Case Name	Membuat PO *Report*	
Actor	*Purchasing Manager*	
Brief Description	*Purchasing* membuat PO *Report*	
Trigerring Event	*Monthly Report*	
Flow of Events		
No	*Actor*	*System*
1.	*Actor* mengklik menu PO	Sistem menampilkan *Listing* PO
2.	*Actor* memasukkan tanggal *start* dan tanggal *end* dan klik *import report*	Sistem menghasilkan *output* poreport.pdf

(Sumber: Hasil Analisis, 2019)

Tabel V.14. Skenario *Use Case* Mengelola Data Master *Supplier*

Use Case Name	Mengelola Data Master *Supplier*	
Actor	*Purchasing Manager*	
Brief Description	*Purchasing* menambahkan, mengubah, dan menghapus *Supplier*	
Trigerring Event	*Purchasing* memiliki data *Supplier* baru	
Flow of Events		
No	*Actor*	*System*
1.	*Actor* membuka halaman utama	Sistem menampilkan halaman utama
2.	*Actor* mengklik menu Data Master *Supplier*	Sistem menampilkan data master *Supplier*
3.	*Actor* memasukkan data *Supplier* baru	Sistem menyimpan dalam *database*
4.	*Actor* mengubah data *Supplier* lama	Sistem menyimpan dalam *database*
5.	*Actor* menghapus data *Supplier*	Sistem menghapus dalam *database*

(Sumber: Hasil Analisis, 2019)

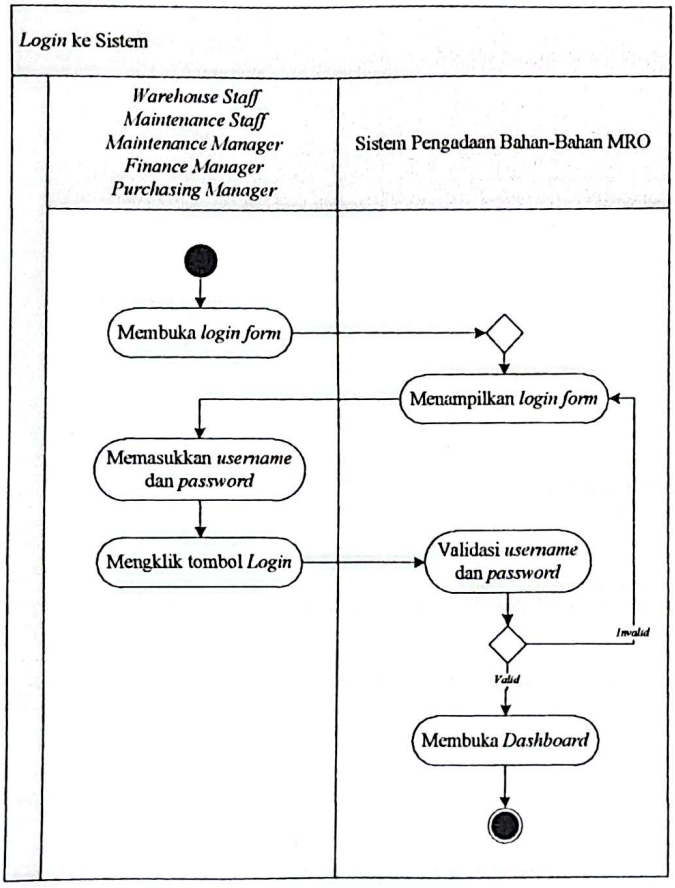
5.3.2. *Activity Diagram*

Activity Diagram merupakan sebuah diagram yang menggambarkan urutan aktivitas proses bisnis pada suatu sistem, sebagai bantuan dalam proses memahami proses bisnis yang berjalan pada sistem informasi pengadaan bahan MRO. Berikut merupakan *activity diagram* usulan dari sistem informasi pengadaan bahan MRO di PT Cikarang Perkasa Manufacturing:

Dipindai dengan CamScanner

1. *Activity Diagram Login* ke Sistem Pengadaan Bahan MRO

   Activity diagram dibawah ini menjelaskan bagaimana *user* yang ada di sistem seperti *Warehouse Staff*, *Maintenance Staff*, *Maintenance Manager*, *Finance Manager* dan *Purchasing Manager login* ke dalam sistem pengadaan bahan MRO.

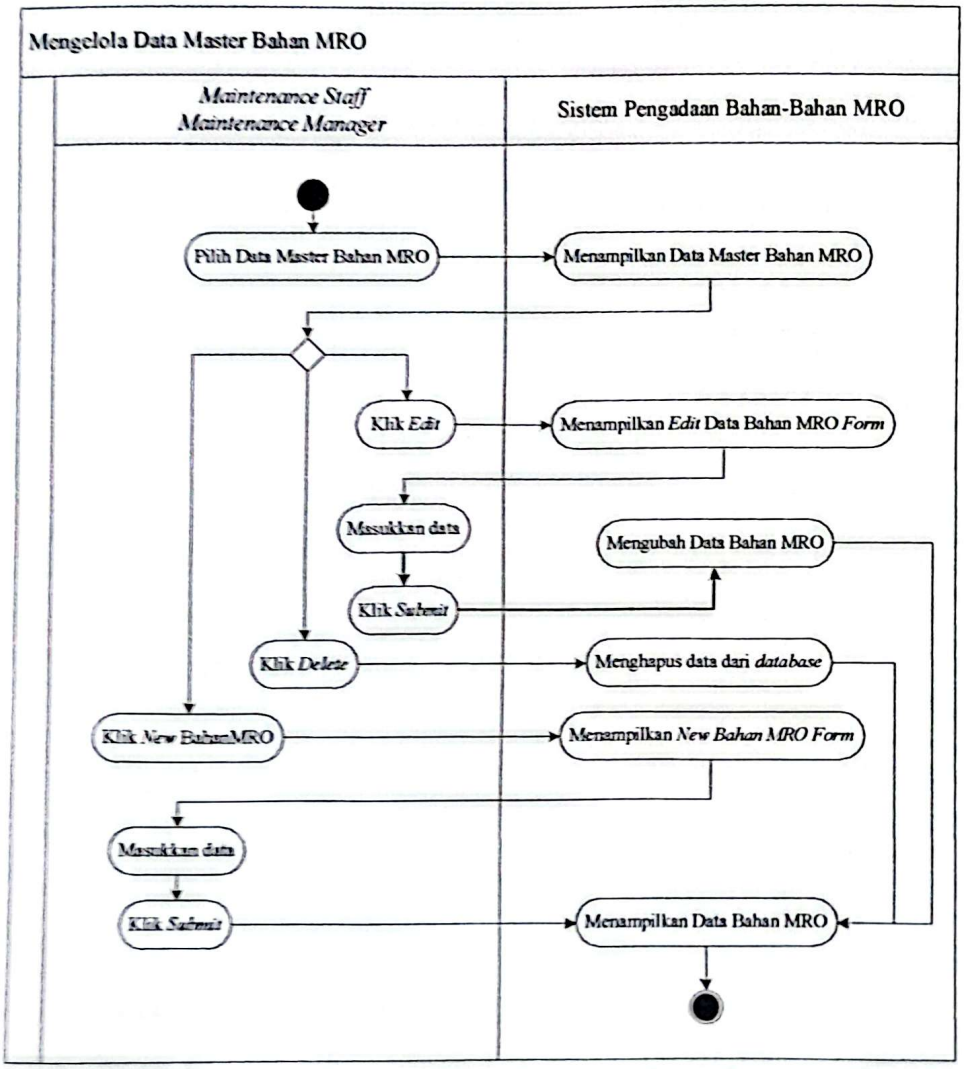


Gambar V.3. ***Activity Diagram Login*** **ke Sistem**
(Sumber: Hasil Analisis, 2019)

Dipindai dengan CamScanner

2. *Activity Diagram* Mengelola Data Master Bahan MRO

Activity diagram ini menjelaskan bagaimana *Maintenance* mengelola bahan MRO yang ada seperti melakukan *input* data bahan MRO baru, mengubah data bahan MRO, menghapus data bahan MRO.

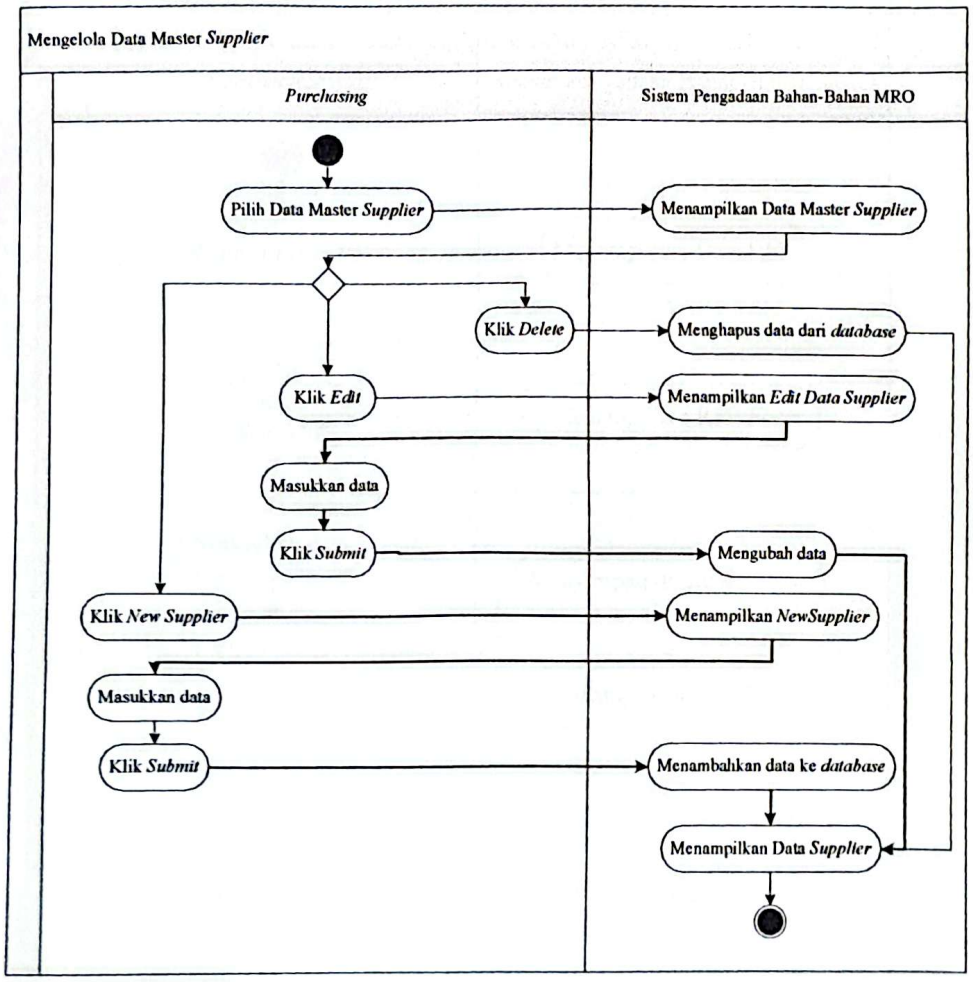


Gambar V.4. *Activity Diagram* Mengelola Data Master Bahan MRO
(Sumber: Hasil Analisis, 2019)

Dipindai dengan CamScanner

Activity Diagram Mengelola Data Master *Supplier*

Activity diagram ini menjelaskan bagaimana *Purchasing* mengelola data master *supplier* yang ada seperti menambahkan *supplier* baru, mengubah data *supplier*, dan menghapus data *supplier*.

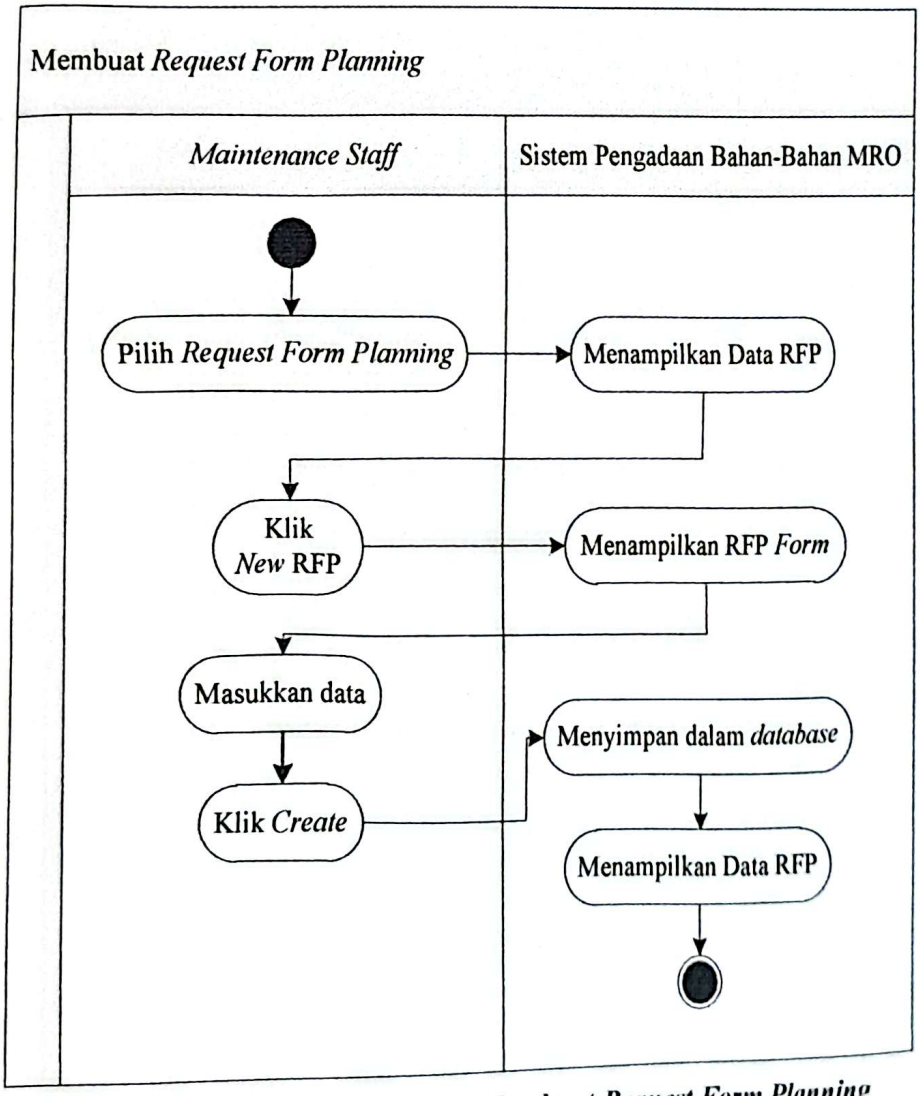


Gambar V.5. *Activity Diagram* Mengelola Data Master *Supplier*
(Sumber: Hasil Analisis, 2019)

Dipindai dengan CamScanner

4. *Activity Diagram* Membuat *Request Form Planning*

Activity diagram ini menjelaskan bagaimana *Maintenance Staff* mengajukan *Request Form Planning*.

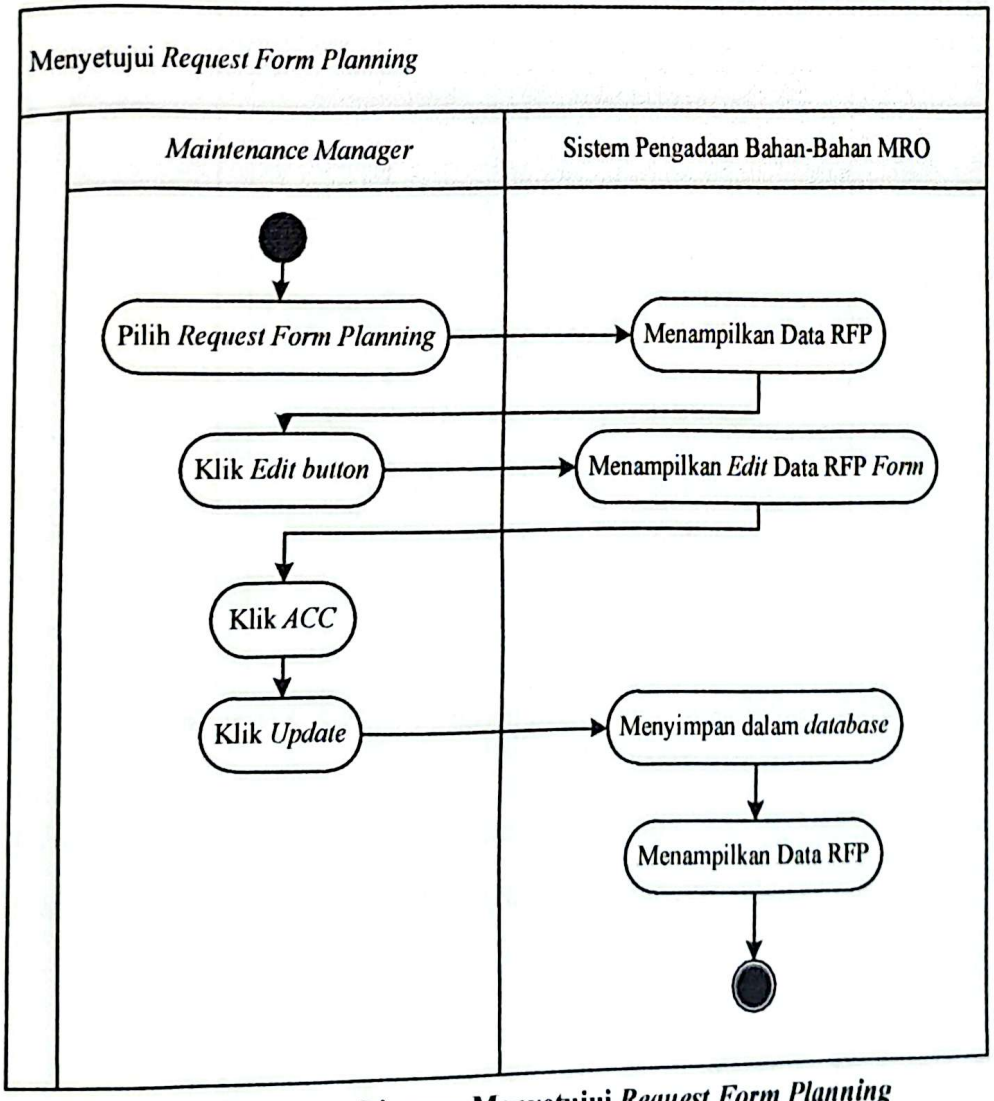


Gambar V.6. ***Activity Diagram*** **Membuat** ***Request Form Planning***
(Sumber: Hasil Analisis, 2019)

Dipindai dengan CamScanner

5. *Activity Diagram* Menyetujui *Request Form Planning*

Activity diagram ini menjelaskan bagaimana *Maintenance Manager* menyetujui RFP.

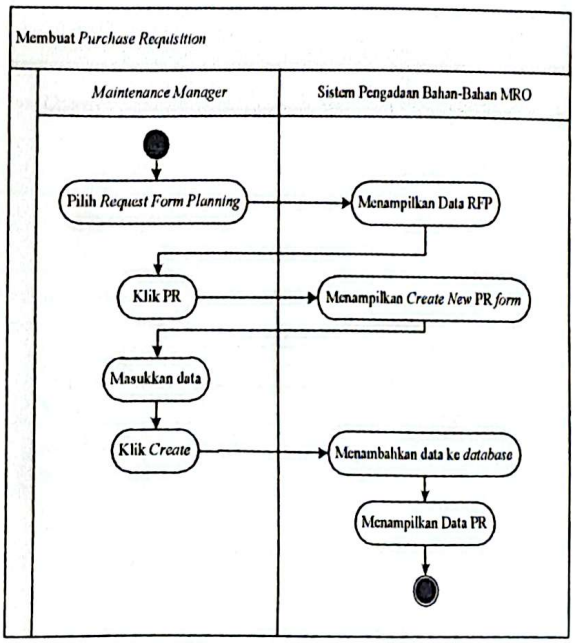


Gambar V.7. *Activity Diagram* Menyetujui *Request Form Planning*
(Sumber: Hasil Analisis, 2019)

Dipindai dengan CamScanner

6. *Activity Diagram* Membuat *Purchase Requisition*

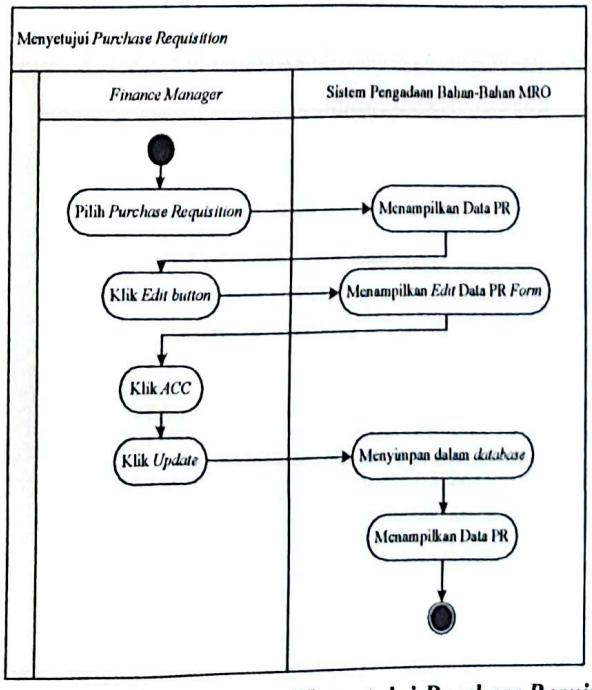
Activity diagram ini menjelaskan bagaimana *Maintenance Staff* membuat PR berdasarkan RFP yang sudah disetujui.



Gambar V.8. ***Activity Diagram*** **Membuat** ***Purchase Requisition***
(Sumber: Hasil Analisis, 2019)

7. *Activity Diagram* Menyetujui *Purchase Requisition*

Activity diagram ini menjelaskan bagaimana *actor* menyetujui PR.

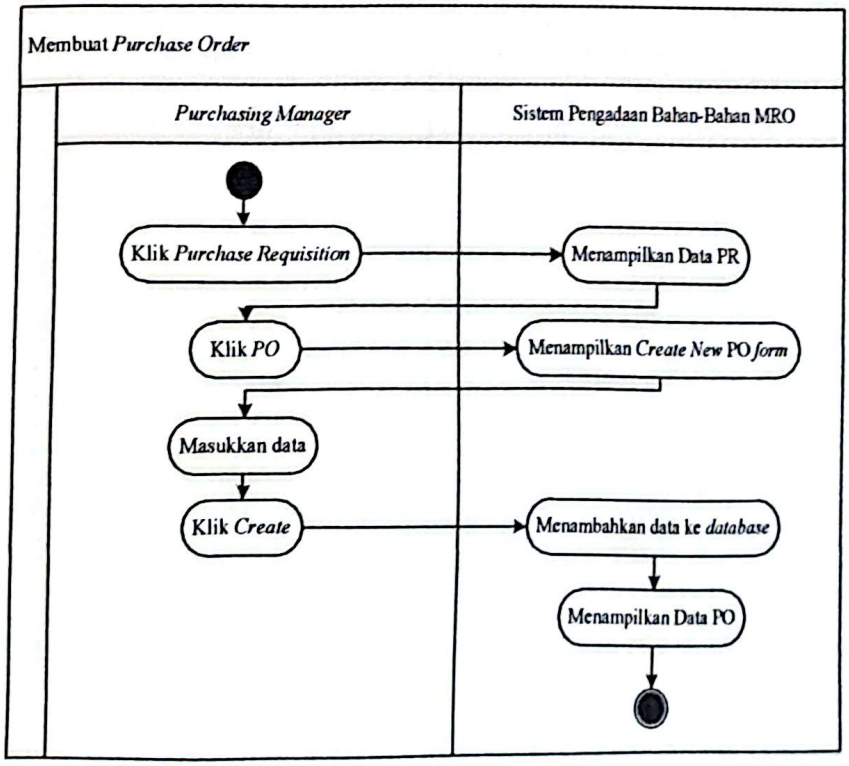


Gambar V.9. ***Activity Diagram*** **Menyetujui** ***Purchase Requisition***
(Sumber: Hasil Analisis, 2019)

Dipindai dengan CamScanner

8. *Activity Diagram* Membuat *Purchase Order*

Activity diagram ini menjelaskan bagaimana *Purchasing Manager* membuat *Purchase Order* berdasarkan *Purchase Requisition* yang nantinya akan dikirim kepada *supplier.*

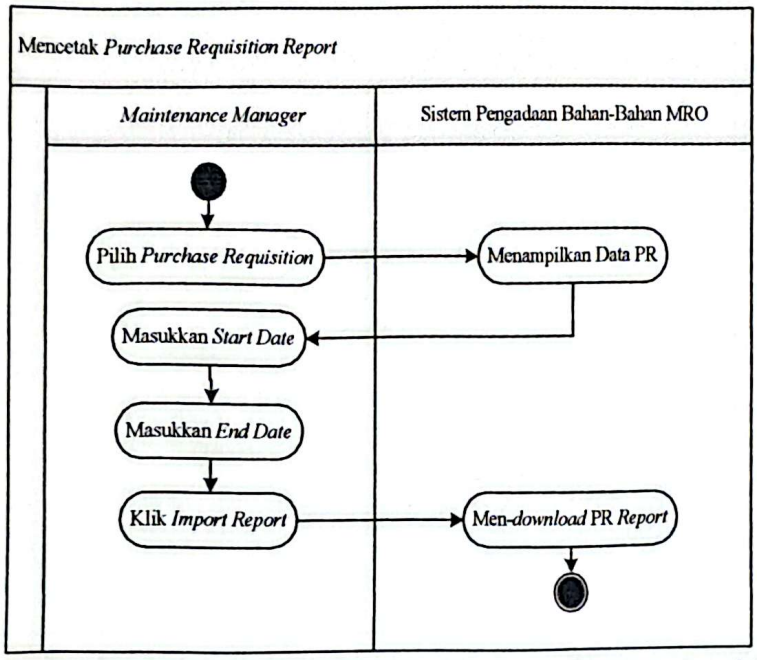


Gambar V.10. ***Activity Diagram*** **Membuat** ***Purchase Order***
(Sumber: Hasil Analisis, 2019)

Dipindai dengan CamScanner

9. *Activity Diagram* Mencetak Laporan PR

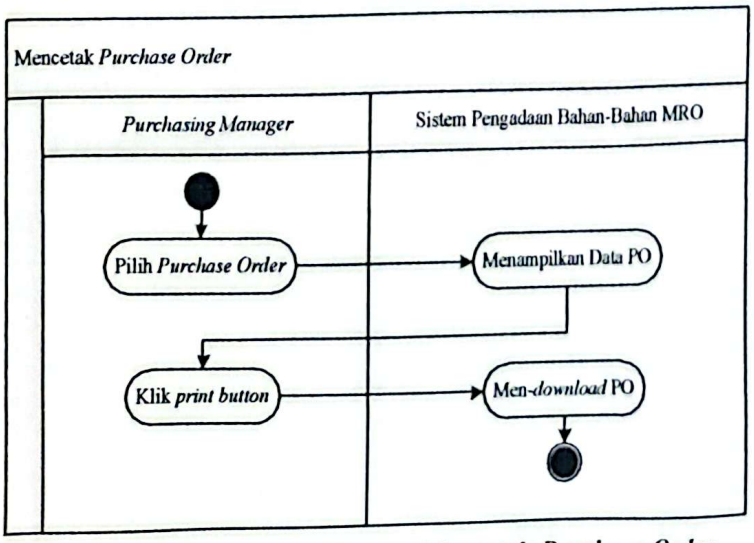
Activity diagram ini menjelaskan bagaimana *Maintenance Manager* mencetak laporan PR berdasarkan periode waktu yang dapat ditentukan.



Gambar V.11. ***Activity Diagram*** **Mencetak Laporan** ***Purchase Requisition***
(Sumber: Hasil Analisis, 2019)

10. *Activity Diagram* Mencetak PO

Activity diagram ini menjelaskan bagaimana *Purchasing Manager* mencetak *Purchase Order* untuk dikirim via email kepada *supplier*.

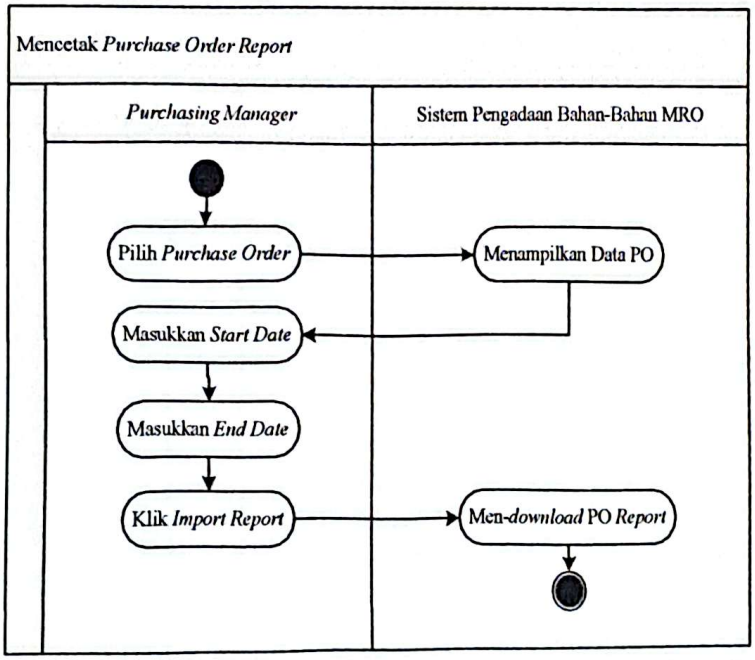


Gambar V.12. ***Activity Diagram*** **Mencetak** ***Purchase Order***
(Sumber: Hasil Analisis, 2019)

Dipindai dengan CamScanner

11. *Activity Diagram* Mencetak Laporan PO

Activity diagram ini menjelaskan bagaimana *Purchasing Manager* mencetak Laporan *Purchase Order* per periode waktu yang dapat ditentukan.

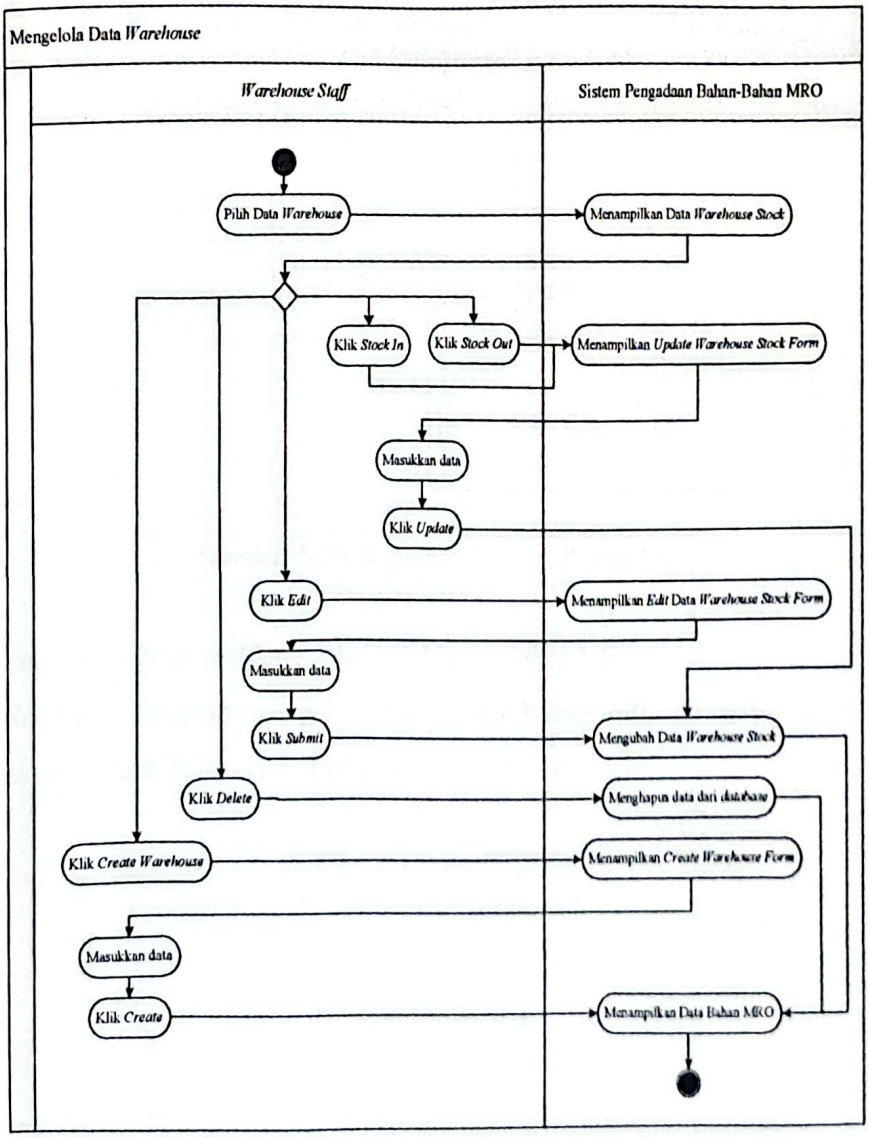


Gambar V.13. ***Activity Diagram*** **Mencetak Laporan** ***Purchase Order***
(Sumber: Hasil Analisis, 2019)

Dipindai dengan CamScanner

12. *Activity Diagram* Mengelola Data *Warehouse*

Activity diagram ini menjelaskan bagaimana *Warehouse Staff* mengelola data bahan MRO yang masuk dan keluar *warehouse.*



Gambar V.14. *Activity Diagram* Mengelola Data *Warehouse Stock*
(Sumber: Hasil Analisis, 2019)

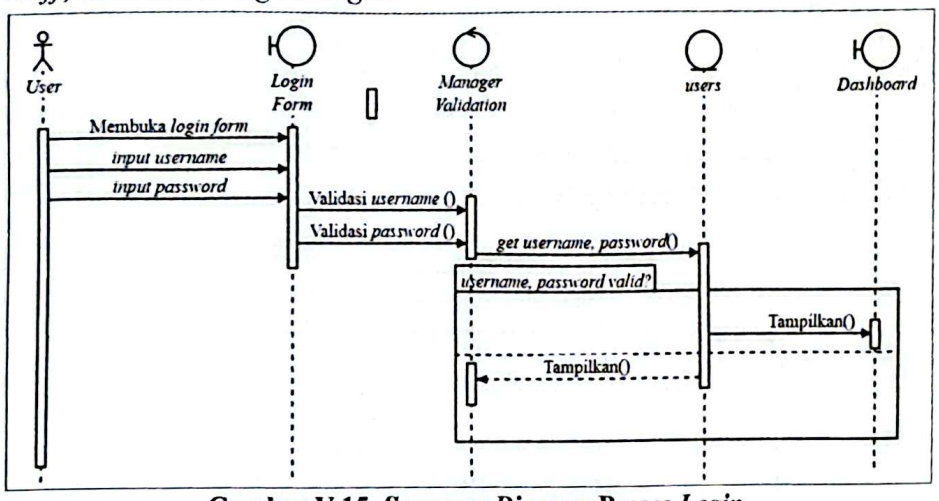
5.3.3. *Sequence Diagram*

Untuk menggambarkan bagaimana aliran aktivitas dan interaksi antar objek dalam sistem yang dirancang dan yang digunakan maka menggunakan *sequence diagram.* Berikut ini merupakan *sequence diagram* untuk sistem pengadaan bahan-bahan MRO.

Dipindai dengan CamScanner

1. *Sequence Diagram* Proses *Login*

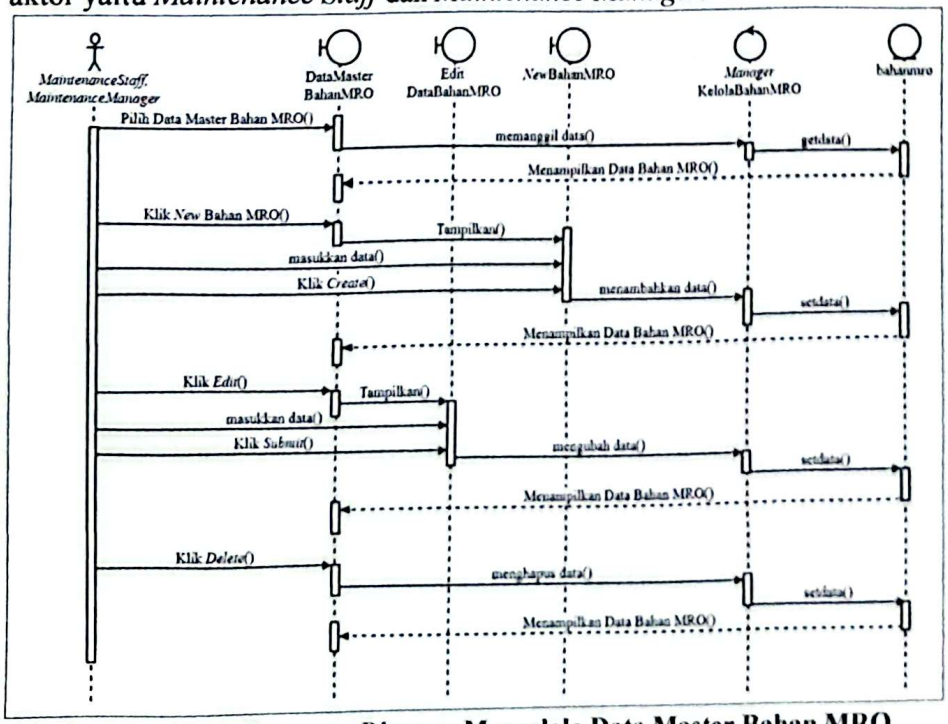
   Sequence Diagram proses *login* dibawah ini dikerjakan oleh 5 *user* yaitu *Maintenance Staff*, *Maintenance Manager*, *Finance Manager*, *Warehouse Staff*, dan *Purchasing Manager*.



Gambar V.15. *Sequence Diagram* Proses *Login*
(Sumber: Hasil Analisis, 2019)

2. *Sequence Diagram* Mengelola Data Master Bahan MRO

   Sequence Diagram mengelola data master bahan MRO dikerjakan oleh 2 aktor yaitu *Maintenance Staff* dan *Maintenance Manager*.

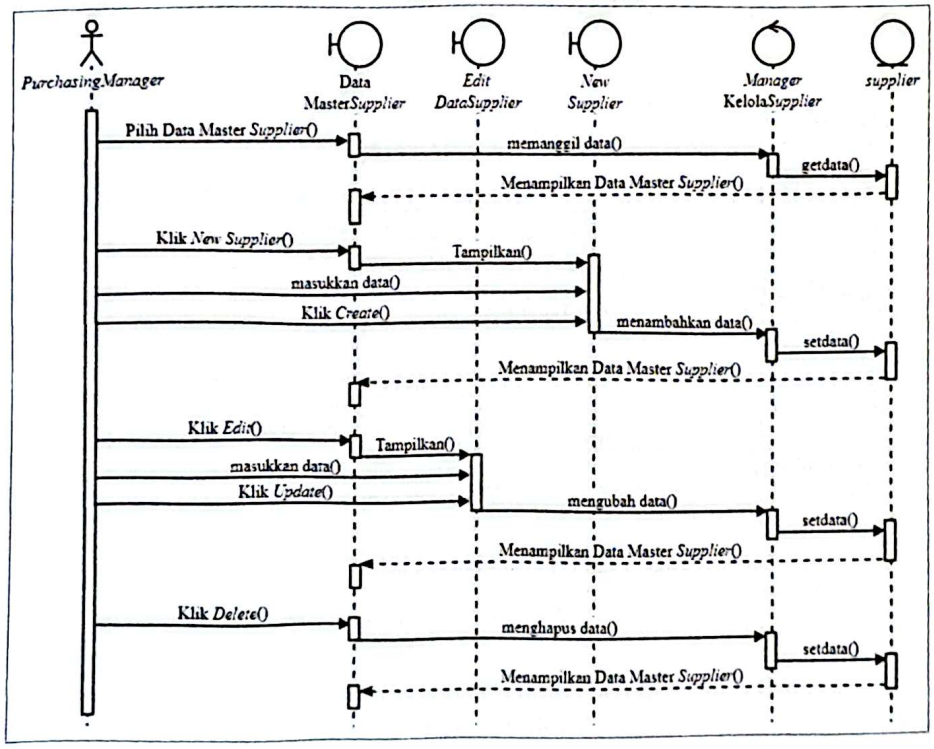


Gambar V.16. *Sequence Diagram* Mengelola Data Master Bahan MRO
(Sumber: Hasil Analisis, 2019)

Dipindai dengan CamScanner

3. *Sequence Diagram* Mengelola Data Master *Supplier*

Sequence Diagram mengelola data master *supplier* dikerjakan oleh 1 aktor yaitu *Purchasing Manager*. *Boundary* yang terlibat adalah DataMasterSupplier, EditDataSupplier, dan NewSupplier.

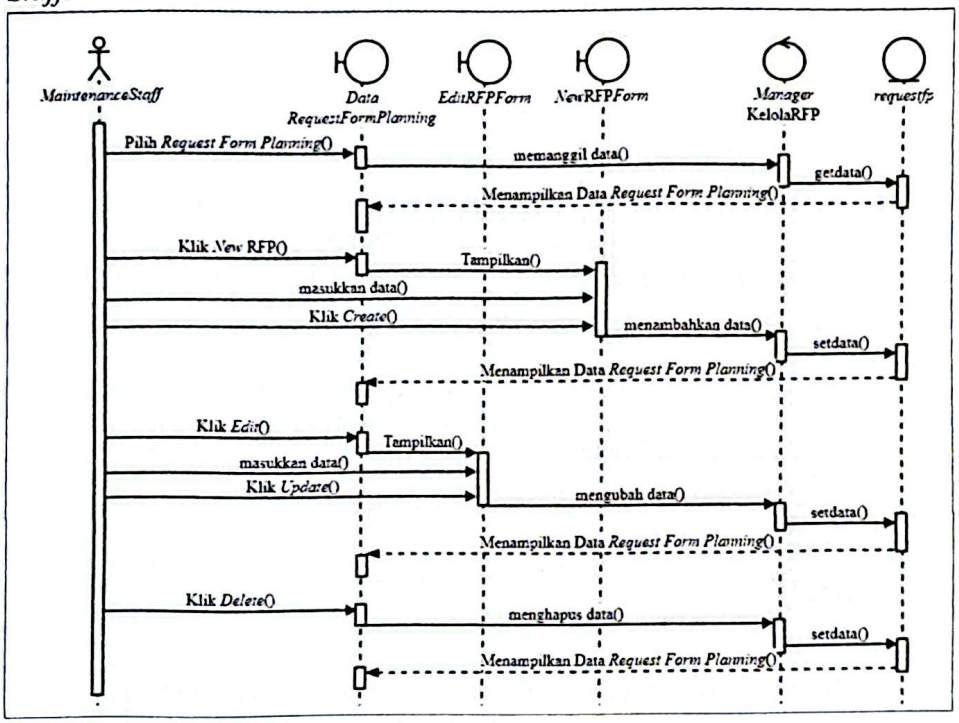


Gambar V.17. *Sequence Diagram* Mengelola Data Master *Supplier*
(Sumber: Hasil Analisis, 2019)

Dipindai dengan CamScanner

4. *Sequence Diagram* Membuat *Request Form Planning*

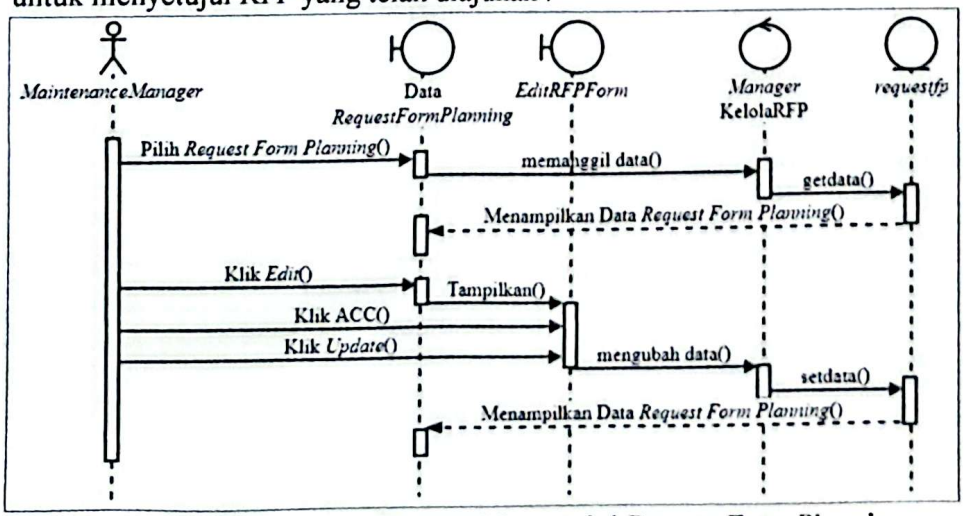
Sequence diagram membuat RFP dikerjakan oleh 1 aktor yaitu *Maintenance Staff.*



Gambar V.18. ***Sequence Diagram*** **Membuat** ***Request Form Planning***
(Sumber: Hasil Analisis, 2019)

5. *Sequence Diagram* Menyetujui *Request Form Planning*

Sequence diagram ini dikerjakan oleh 1 *actor* yaitu *Maintenance Manager* untuk menyetujui RFP yang telah diajukan .

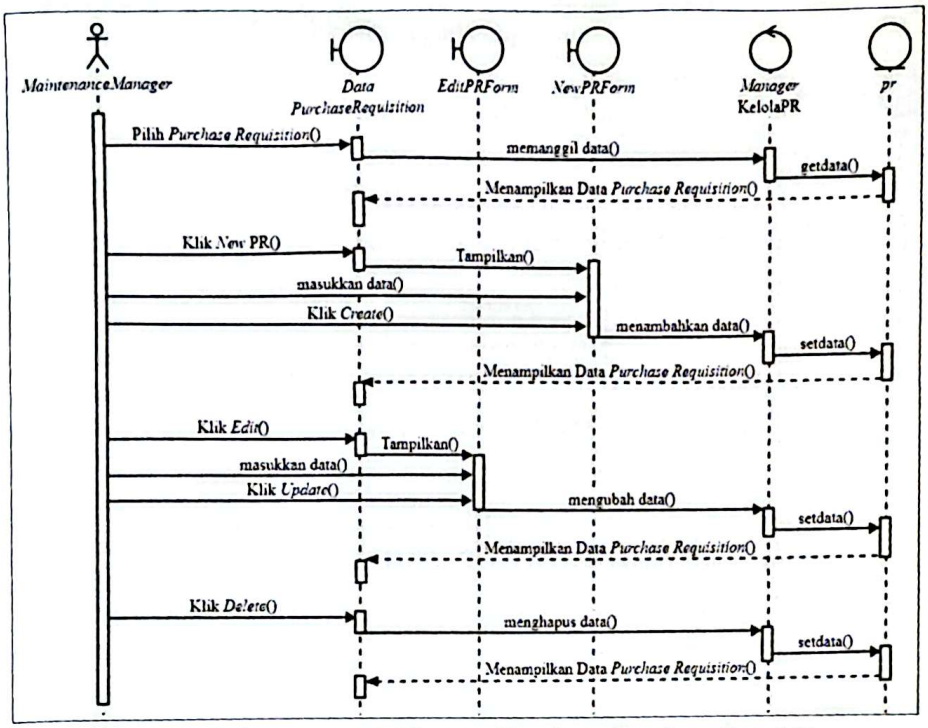


Gambar V.19. ***Sequence Diagram*** **Menyetujui** ***Request Form Planning***
(Sumber: Hasil Analisis, 2019)

Dipindai dengan CamScanner

6. *Sequence Diagram* Membuat *Purchase Requisition*

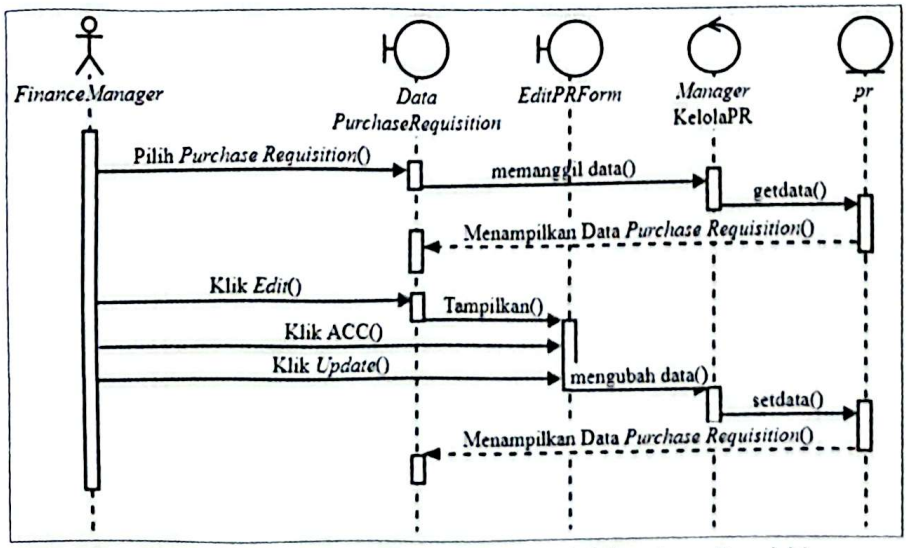
Sequence diagram ini dikerjakan oleh 1 aktor yaitu *Maintenance Manager*.



Gambar V.20. *Sequence Diagram* Membuat *Purchase Requisition*
(Sumber: Hasil Analisis, 2019)

7. *Sequence Diagram* Menyetujui *Purchase Requisition*

Sequence diagram ini terdiri dari *actor Finance Manager* untuk menyetujui PR yang telah dibuat *Maintenance Manager*.

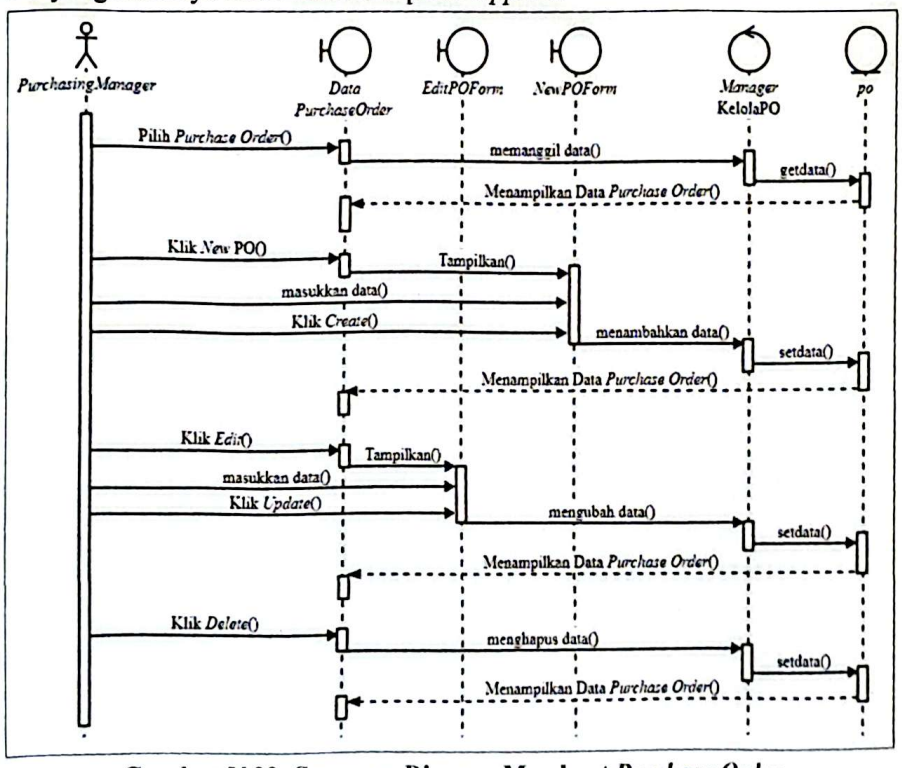


Gambar V.21. *Sequence Diagram* Menyetujui *Purchase Requisition*
(Sumber: Hasil Analisis, 2019)

Dipindai dengan CamScanner

8. *Sequence Diagram* Membuat *Purchase Order*

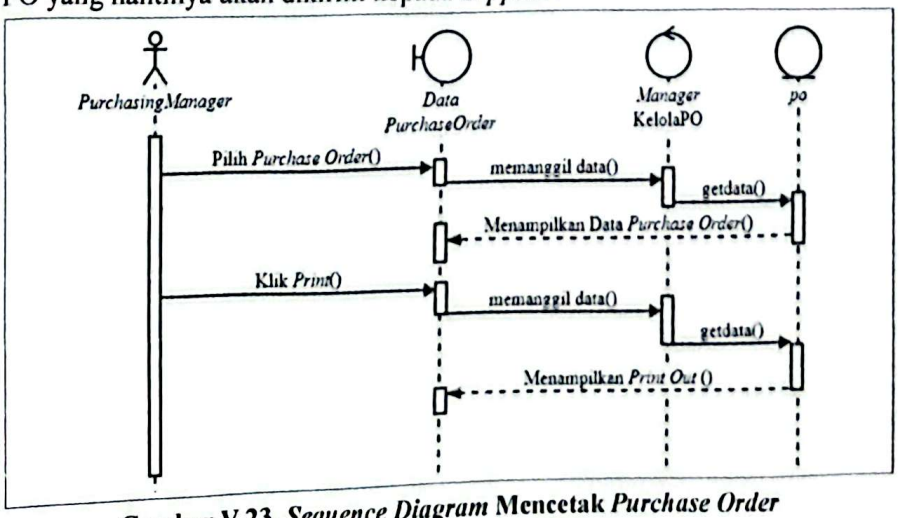
Sequence diagram ini dikerjakan oleh *Purchasing Manager* untuk membuat PO yang nantinya akan dikirim kepada *Supplier*.



Gambar V.22. *Sequence Diagram* Membuat *Purchase Order*
(Sumber: Hasil Analisis, 2019)

9. *Sequence Diagram* Mencetak *Purchase Order*

Sequence diagram ini dikerjakan oleh *Purchasing Manager* untuk mencetak PO yang nantinya akan dikirim kepada *Supplier*.

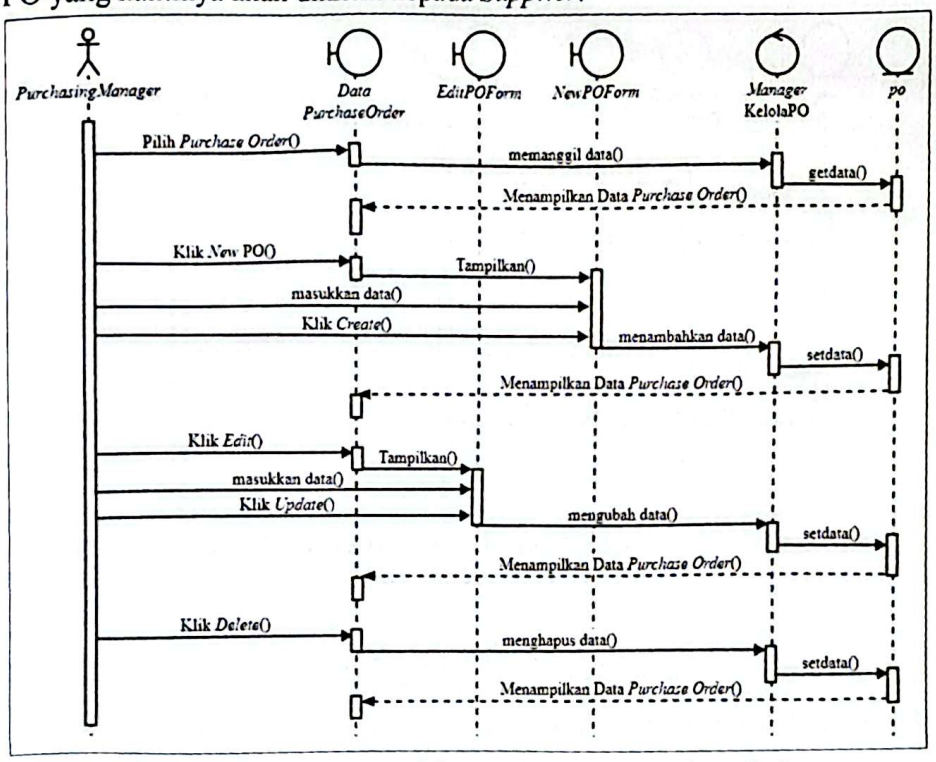


Gambar V.23. *Sequence Diagram* Mencetak *Purchase Order*
(Sumber: Hasil Analisis, 2019)

Dipindai dengan CamScanner

8. *Sequence Diagram* Membuat *Purchase Order*

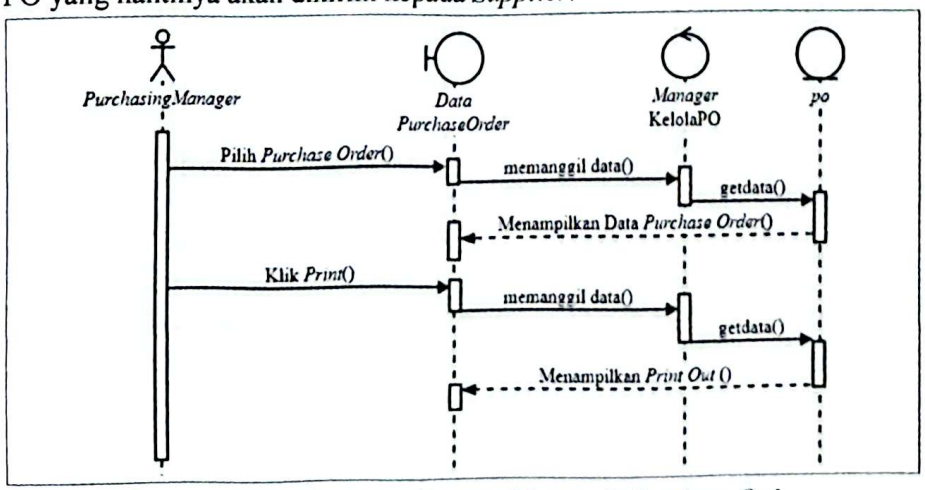
Sequence diagram ini dikerjakan oleh *Purchasing Manager* untuk membuat PO yang nantinya akan dikirim kepada *Supplier*.



Gambar V.22. *Sequence Diagram* Membuat *Purchase Order*
(Sumber: Hasil Analisis, 2019)

9. *Sequence Diagram* Mencetak *Purchase Order*

Sequence diagram ini dikerjakan oleh *Purchasing Manager* untuk mencetak PO yang nantinya akan dikirim kepada *Supplier*.

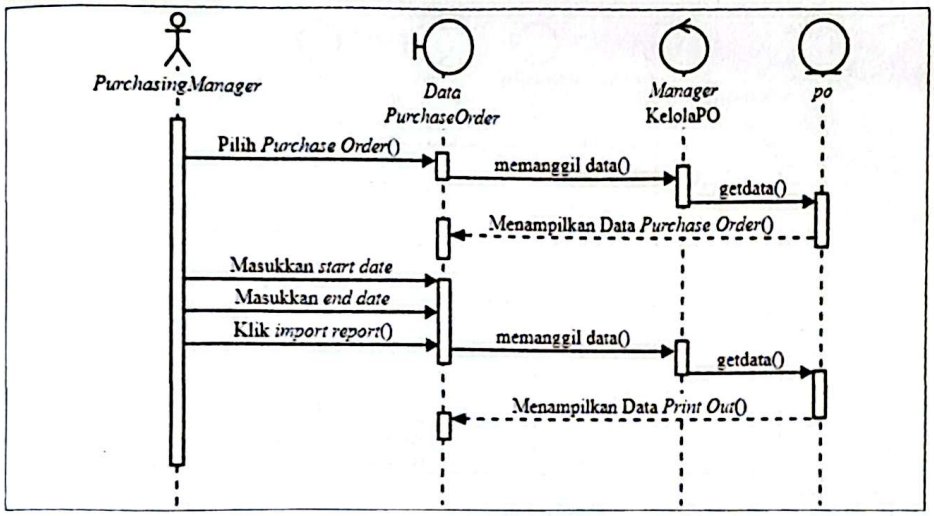


Gambar V.23. *Sequence Diagram* Mencetak *Purchase Order*
(Sumber: Hasil Analisis, 2019)

Dipindai dengan CamScanner

10. *Sequence Diagram* Mencetak Laporan *Purchase Order*

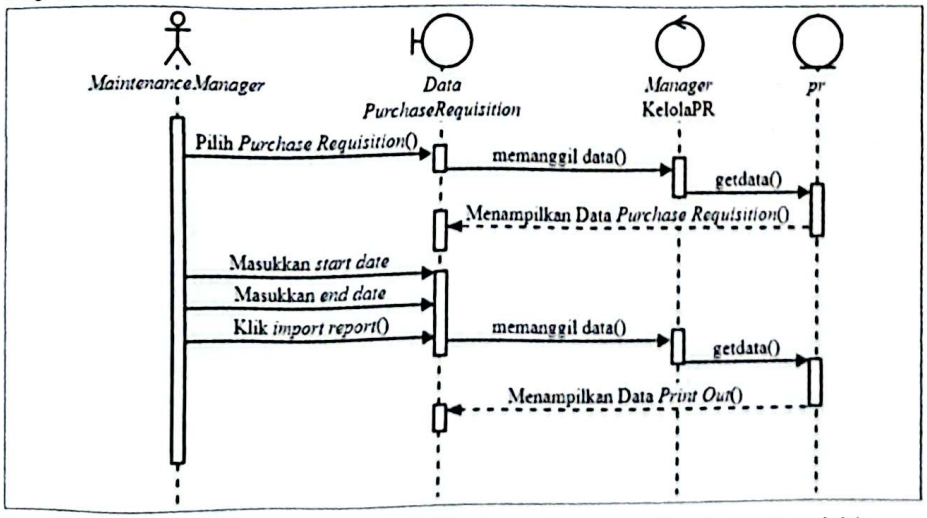
Sequence diagram ini dikerjakan oleh *Purchasing Manager* untuk mencetak Laporan PO.



Gambar V.24. *Sequence Diagram* Mencetak Laporan *Purchase Order*
(Sumber: Hasil Analisis, 2019)

11. *Sequence Diagram* Mencetak Laporan *Purchase Requisition*

Sequence diagram ini dikerjakan oleh *Maintenance Manager* untuk mencetak Laporan PR.

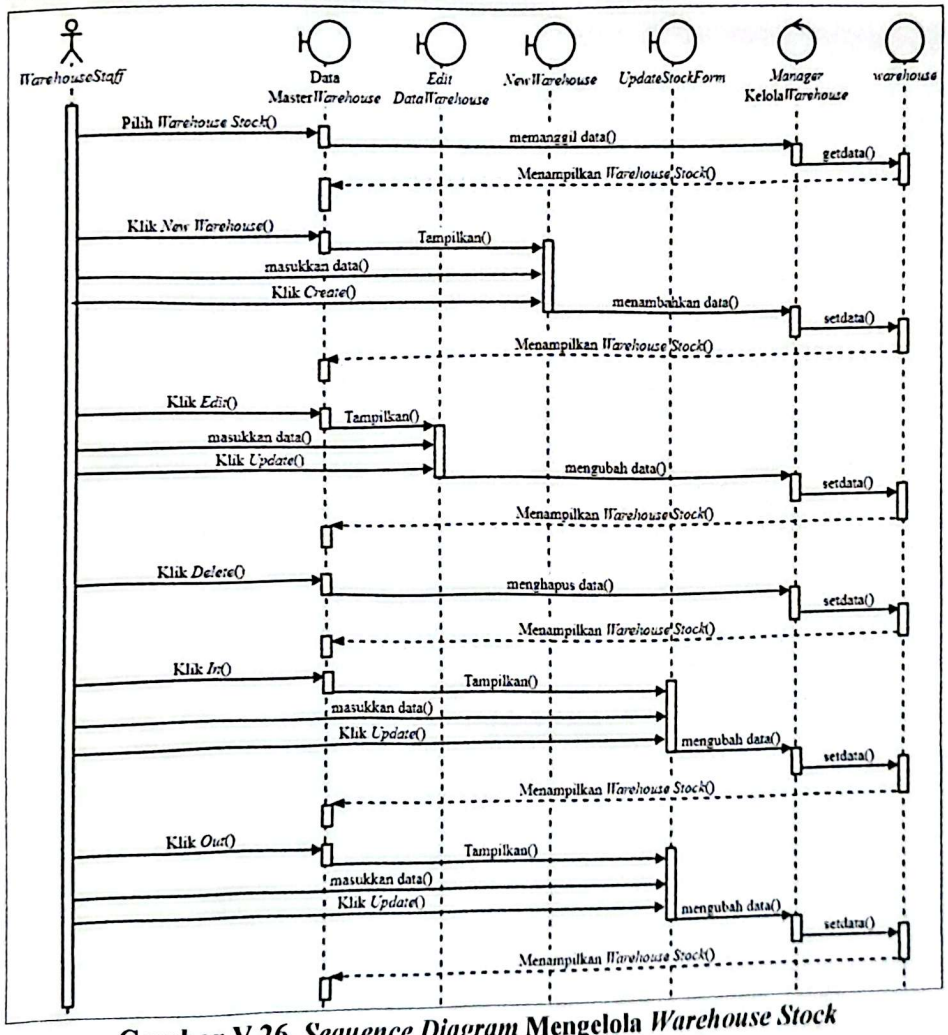


Gambar V.25. *Sequence Diagram* Mencetak Laporan *Purchase Requisition*
(Sumber: Hasil Analisis, 2019)

Dipindai dengan CamScanner

12. *Sequence Diagram* Mengelola *Warehouse Stock*

Sequence diagram ini dikerjakan oleh *Warehouse Staff* untuk mengelola bahan MRO yang masuk dan keluar.

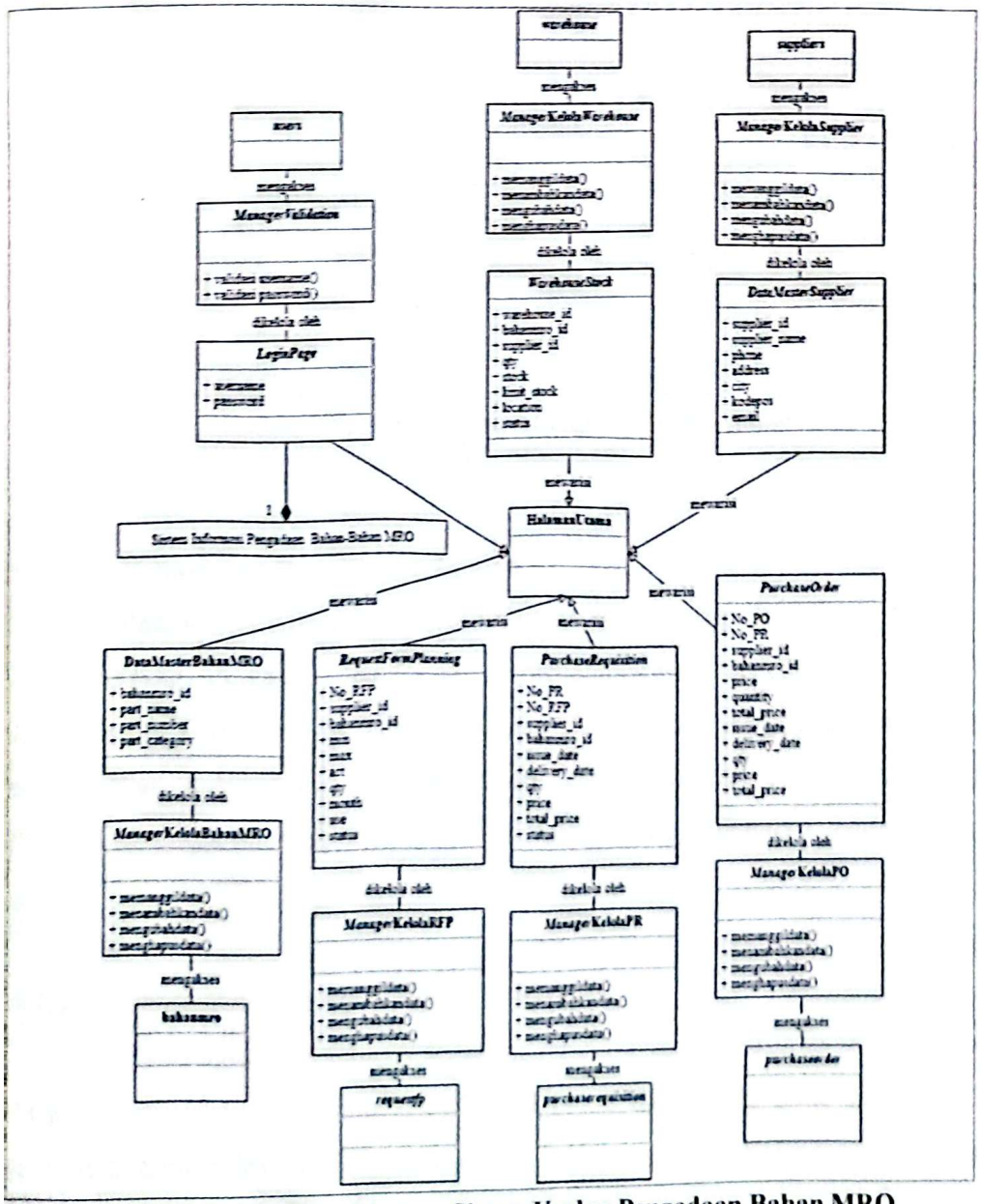


Gambar V.26. *Sequence Diagram* Mengelola *Warehouse Stock*
(Sumber: Hasil Analisis, 2019)

Dipindai dengan CamScanner

5.3.4. *Class Diagram*

Class Diagram pada usulan sistem informasi persediaan bahan MRO di PT Cikarang Perkasa Manufacturing digunakan untuk 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, berikut merupakan *class diagram* yang diusulkan dapat dilihat pada Gambar V.27. berikut:

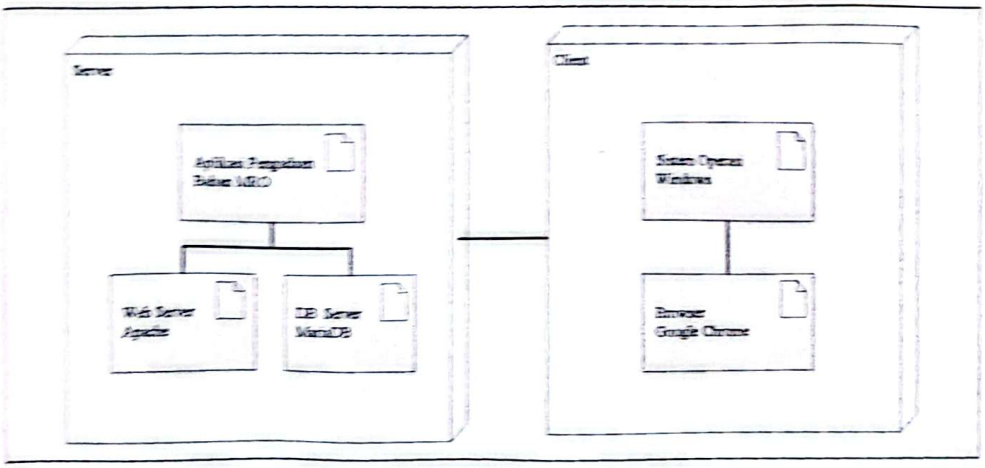


Gambar V.27. *Class Diagram* Sistem Usulan Pengadaan Bahan MRO
(Sumber: Hasil Analisis, 2019)

Dipindai dengan CamScanner

5.3.5. *Deployment Diagram*

Deployment diagram yang digambarkan dibawah ini menggambarkan arsitektur fisik dari sistem pengadaan bahan MRO seperti bagaimana komponen dalam infrastruktur sistem, letak *server web* dan *server database*, *browser* yang digunakan. *Deployment diagram* Sistem Usulan Pengadaan Bahan-Bahan MRO dapat dilihat pada Gambar V.28.



Gambar V.28. *Deployment Diagram* Sistem Usulan
Sumber: Hasil Analisis, (2019)

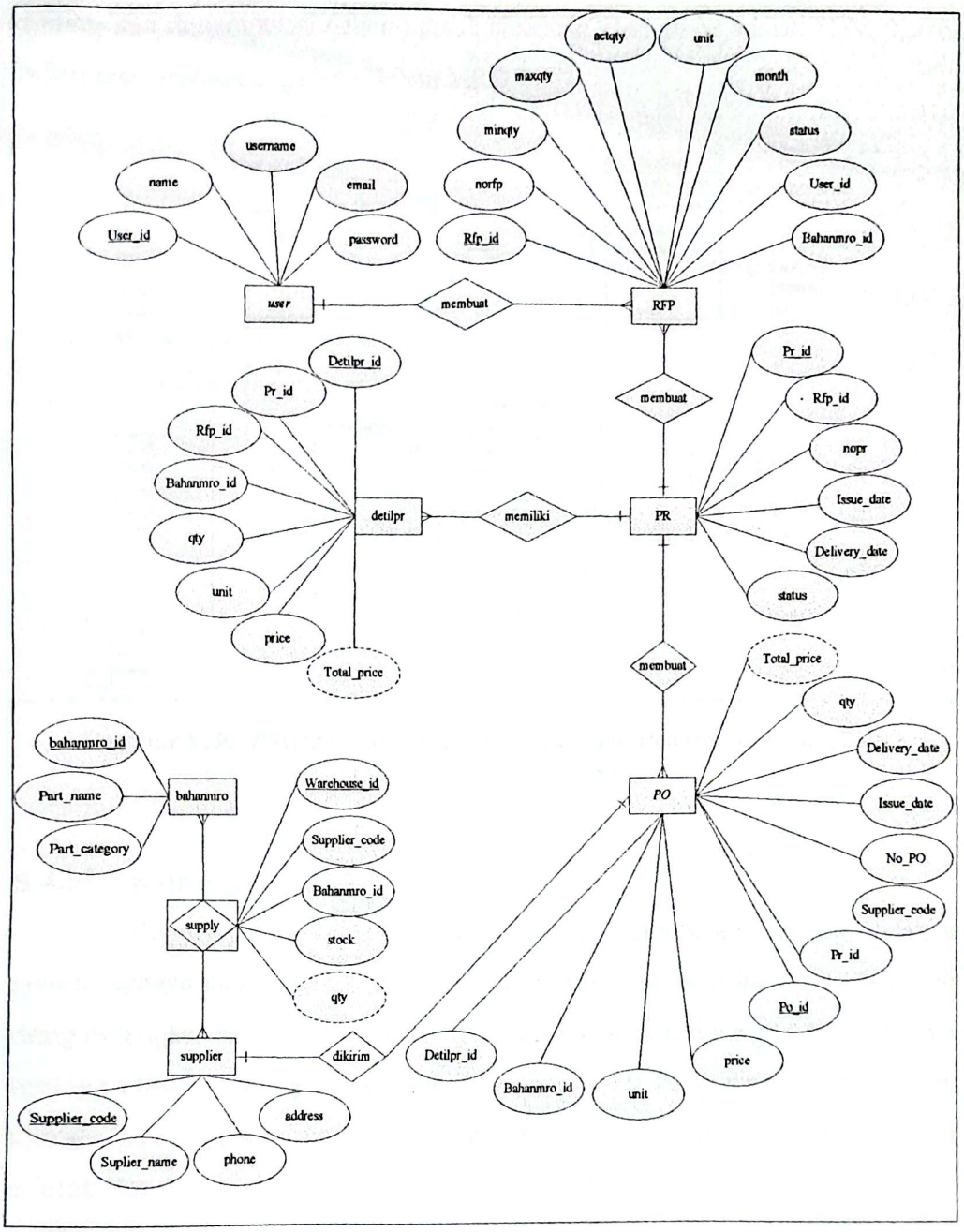
5.4. Pemodelan Data Sistem Usulan

Pemodelan data pada sistem informasi usulan pengadaan bahan MRO di PT Cikarang Perkasa Manufacturing menggunakan dua cara yaitu *Entity Relationship Diagram* (ERD) dan *Physical Data Model* 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 digunakan sebagai suatu model untuk menjelaskan hubungan antar data dalam basis data berdasarkan objek-objek dasar data yang mempunyai hubungan antar relasi dalam sistem usulan. ERD digambarkan dengan simbol dan notasi yang berbeda, berikut merupakan ERD dari sistem informasi pengadaan bahan MRO dapat dilihat pada gambar berikut ini:

Dipindai dengan CamScanner



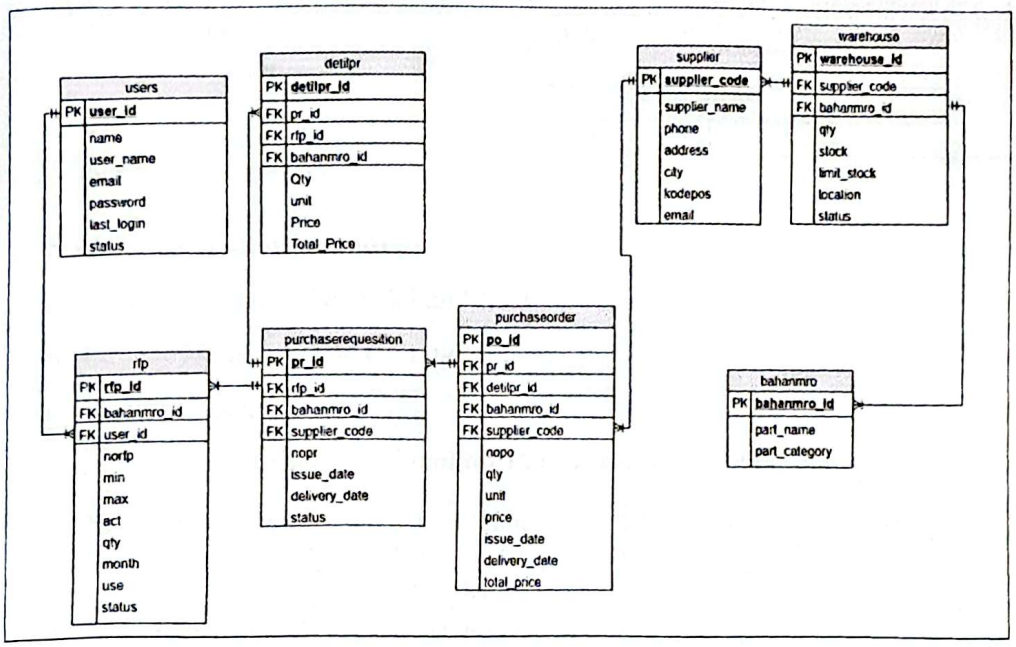
Gambar V.29. ERD Sistem Usulan Pengadaan Bahan MRO
(Sumber: Hasil Analisis, 2019)

5.4.2. *Physical Data Model*

Physical Data Model akan menggambarkan data yang akan digunakan dalam sistem informasi dalam suatu proses bisnis. PDM ini akan melengkapi data model ERD, dimana PDM memberikan gambar bagaimana data akan disimpan,

Dipindai dengan CamScanner

dibuat, dan dimanipulasi dalam sebuah *database*. Berikut ini adalah PDM Sistem Informasi Usulan Pengadaan Bahan MRO.



Gambar V.30. *Physical Data Model* Sistem Usulan Pengadaan Bahan MRO
(Sumber: Hasil Analisis, 2019)

5.4.3. Kamus Data

Kamus menggambarkan sebuah katalog data yang terdapat didalam sistem dengan maksud untuk mendefinisikan data yang mengalir di dalam sistem dengan lengkap sehingga user dan analis 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 persediaan bahan MRO:

1. Spesifikasi tabel *user*
   a. Nama arus data : *user*
   b. Bentuk Data : File data *master*
   c. Struktur Data :

Dipindai dengan CamScanner

Tabel V.15. Struktur Data Tabel *Users*

No.	Nama Elemen	Alias	Tipe Data	Panjang	Keterangan
1	user_id		Int	10	*Primary key*
2	Name		Varchar	20	
3	user_name		Varchar	10	
4	Email		Varchar	20	
5	Password		Varchar	8	

(Sumber: Hasil Analisis, 2019)

2. Spesifikasi tabel bahanmro
   a. Nama arus data : Data Bahan MRO
   b. Bentuk Data : File data *master*
   c. Struktur Data :

Tabel V.16. Struktur Data Tabel Bahan MRO

No.	Nama Elemen	Tipe Data	Panjang	Keterangan
1	Bahanmro_id	Int	10	*Primary key*
2	Part_name	Varchar	20	
3	Part_category	Varchar	20	

(Sumber: Hasil Analisis, 2019)

3. Spesifikasi tabel *supplier*
   a. Nama arus data : Data *Supplier*
   b. Bentuk Data : File data *master*
   c. Struktur Data :

Tabel V.17. Struktur Data Tabel *Supplier*

No.	Nama Elemen	Tipe Data	Panjang	Keterangan
1	supplier_code	Int	10	*Primary key*
2	supplier_name	Varchar	20	
3	Phone	Varchar	15	
4	Address	Varchar	20	
5	City	Varchar	15	
6	Kodepos	Varchar	6	
7	Email	Varchar	20	

(Sumber: Hasil Analisis, 2019)

Dipindai dengan CamScanner

4. Spesifikasi tabel RFP

   a. Nama Arus Data : Data RFP

   b. Bentuk Data : File data transaksi

   c. Struktur Data

Tabel V.18. Struktur Data Tabel RFP

No.	Nama Elemen	Tipe Data	Panjang	Keterangan
1	Rfp_id	Int	10	*Primary key*
2	Norfp	Char	5	
3	user_id	Int	10	*Foreign key*
4	Bahanmro_id	Int	10	*Foreign key*
5	Min	Int	20	
6	Max	Int	20	
7	Act	Int	20	
8	Qty	Int	20	
9	Month	Char	15	
10	Use	Int	20	
11	Status	Tinyint	1	

(Sumber: Hasil Analisis, 2019)

5. Spesifikasi tabel PR

   a. Nama Arus Data : *Purchase Requisition*

   b. Bentuk Data : File data transaksi

   c. Struktur Data

Tabel V.19. Struktur Data Tabel PR

No.	Nama Elemen	Tipe Data	Panjang	Keterangan
1	Pr_id	Int	10	*Primary key*
2	Rfp_id	Int	10	*Foreign key*
3	Nopr	Char	5	
4	issue_date	Date		
5	delivery_date	Date		
6	Status	Tinyint	1	

(Sumber: Hasil Analisis, 2019)

6. Spesifikasi tabel Detail PR

   a. Nama Arus Data : Detail *Purchase Requisition*

   b. Bentuk Data : File data transaksi

   c. Struktur Data

Dipindai dengan CamScanner

Tabel V.20. Struktur Data Tabel Detail PR

No.	Nama Elemen	Tipe Data	Panjang	Keterangan
1	detailpr_id	Int	10	*Primary key*
2	Pr_id	Int	10	*Foreign key*
3	Rfp_id	Int	10	*Foreign key*
4	Bahanmro_id	Int	10	*Foreign key*
5	Qty	Int	10	
6	Unit	Char	5	
7	Price	Decimal	15, 2	
8	Total_price	Decimal	15, 2	

(Sumber: Hasil Analisis, 2019)

)) Spesifikasi tabel PO

a. Nama Arus Data : *Purchase Order*

b. Bentuk Data : File data transaksi

c. Struktur Data

Tabel V.21. Struktur Data Tabel PO

No.	Nama Elemen	Tipe Data	Panjang	Keterangan
1	Po_id	Int	10	*Primary key*
2	Pr_id	Int	10	*Foreign key*
3	Detailpr_id	Int	10	*Foreign key*
4	Bahanmro_id	Int	10	*Foreign key*
5	Supplier_code	int	10	*Foreign key*
6	Nopo	Char	5	
7	Qty	int	10	
8	unit	char	5	
9	Issue_date	Date		
10	Delivery_date	Date		
11	price	Decimal	15,2	
12	Total_price	Decimal	15,2	

(Sumber: Hasil Analisis, 2019)

Dipindai dengan CamScanner

11) Spesifikasi tabel *warehouse*

a. Nama Arus Data : *Warehouse Stock*

b. Bentuk Data : File data transaksi

c. Struktur Data

Tabel V.22. Struktur Data Tabel *warehouse*

No.	Nama Elemen	Tipe Data	Panjang	Keterangan
1	Warehouse_id	Int	10	*Primary key*
2	Bahanmro_id	Int	10	*Foreign key*
3	Supplier_code	Int	10	*Foreign key*
4	Qty	Int	20	
5	Stock	int	20	
6	Limit_stock	Int	20	
7	Location	Varchar	5	
8	Status	Tinyint	1	

(Sumber: Hasil Analisis, 2019)

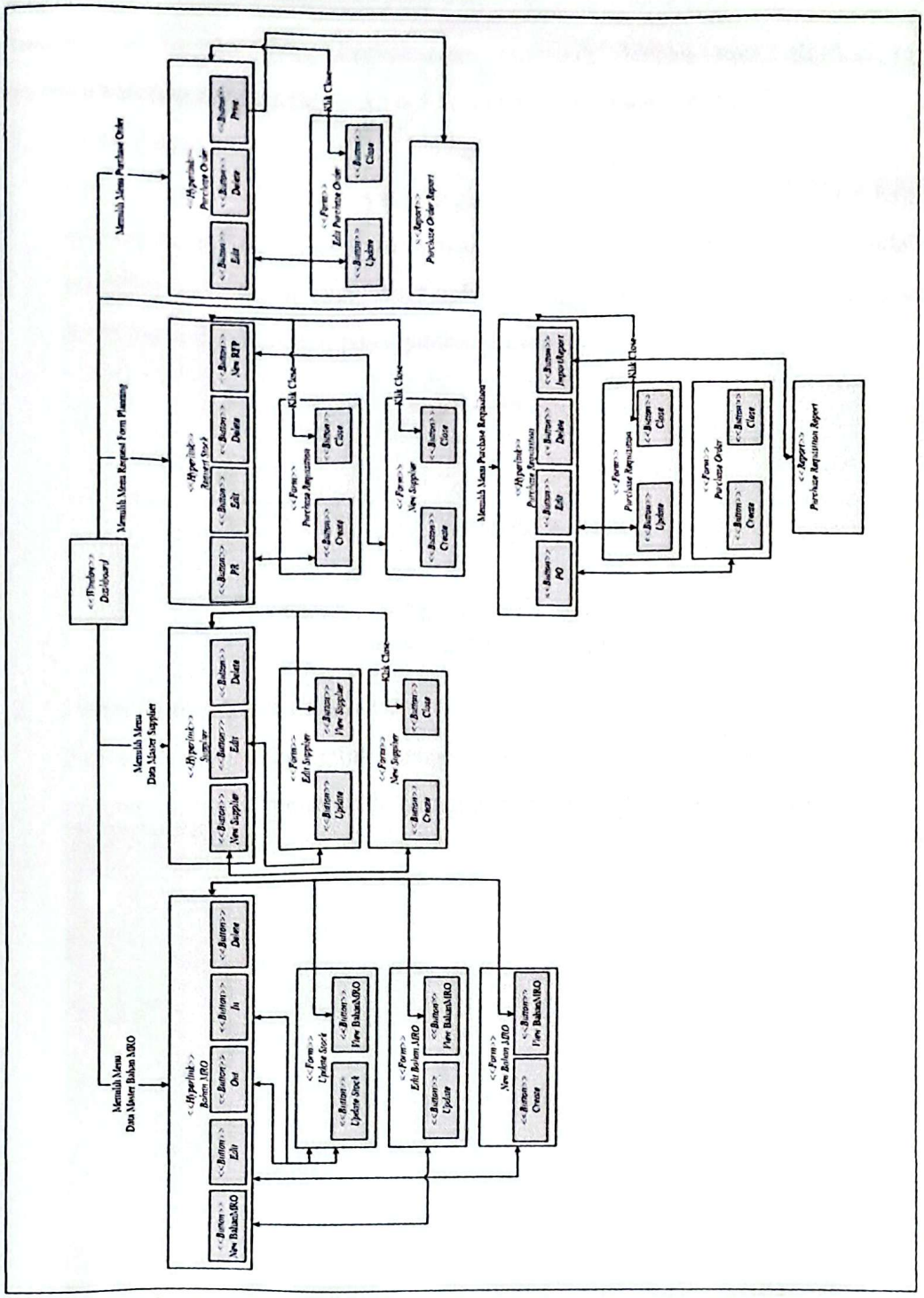
5.5. Perancangan Sistem

Perancangan sistem pada sistem usulan digunakan sebagai ilustrasi yang menggambarkan bagaimana aplikasi yang dibuat berjalan dan juga menggambarkan navigasi dari aplikasi tersebut dengan *tools Windows Navigation Diagram* (WND) serta menggambarkan tampilan antar muka dari aplikasi.

5.5.1. Windows Navigation Diagram

Windows Navigation Diagram pada sistem usulan digunakan untuk menunjukkan bagaimana navigasi dari halaman-halaman yang terdapat di dalam aplikasi. Di WND setiap status antarmuka pengguna direpresentasikan sebagai kotak. Kotak biasanya sesuai dengan komponen antarmuka pengguna, seperti jendela, *form*, tombol, atau laporan, berikut merupakan *Windows Navigation Diagram* sistem pengadaan bahan MRO dapat dilihat pada gambar V.31.:

Dipindai dengan CamScanner



Gambar V.31. WND Sistem Usulan Pengadaan Bahan MRO
(Sumber: Hasil Analisis, 2019)

5.5.2. Perancangan Antarmuka

Perancangan antarmuka sistem usulan pengadaan bahan MRO menggambarkan antarmuka aplikasi yang hendak dibuat, antarmuka ini

Dipindai dengan CamScanner

merupakan bagian yang berhubungan langsung dengan *user*. Berikut ini merupakan rancangan antarmuka aplikasi pengadaan bahan MRO:

1. *Form Login*

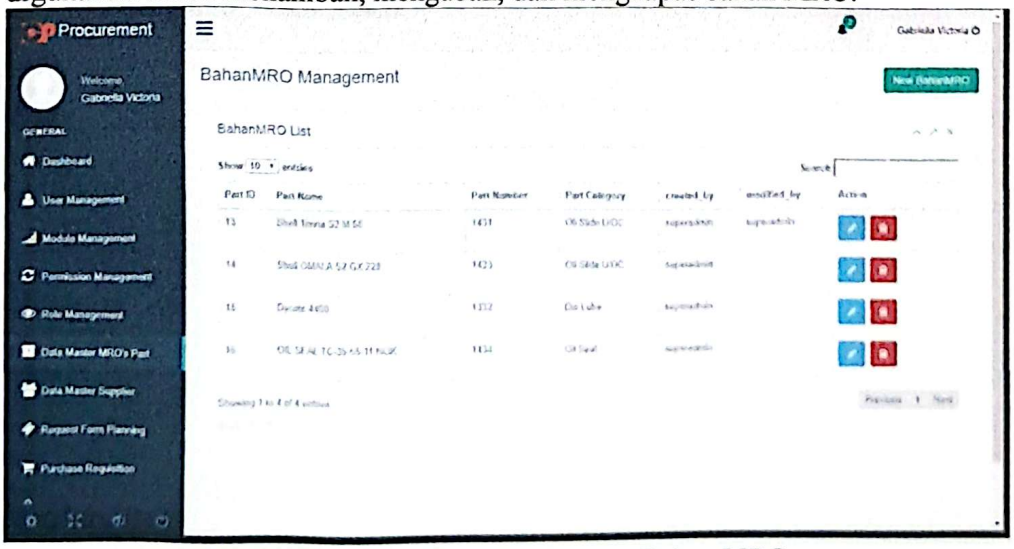
   Form login adalah *form* yang digunakan untuk membatasi hak akses bagi *user*untuk melihat dan berinteraksi dengan data. Hanya *user* yang sudah terdaftar yang dapat mengakses aplikasi pengadaan bahan MRO. Rancangan *form login* dapat dilihat pada gambar dibawah ini:



Gambar V.32. Tampilan *Form Login*
(Sumber: Hasil Analisis, 2019)

2. *Form Data Master Bahan MRO*

   Form ini digunakan untuk mengelola data bahan MRO yang ada. *Form* ini digunakan untuk menambah, mengubah, dan menghapus bahan MRO.

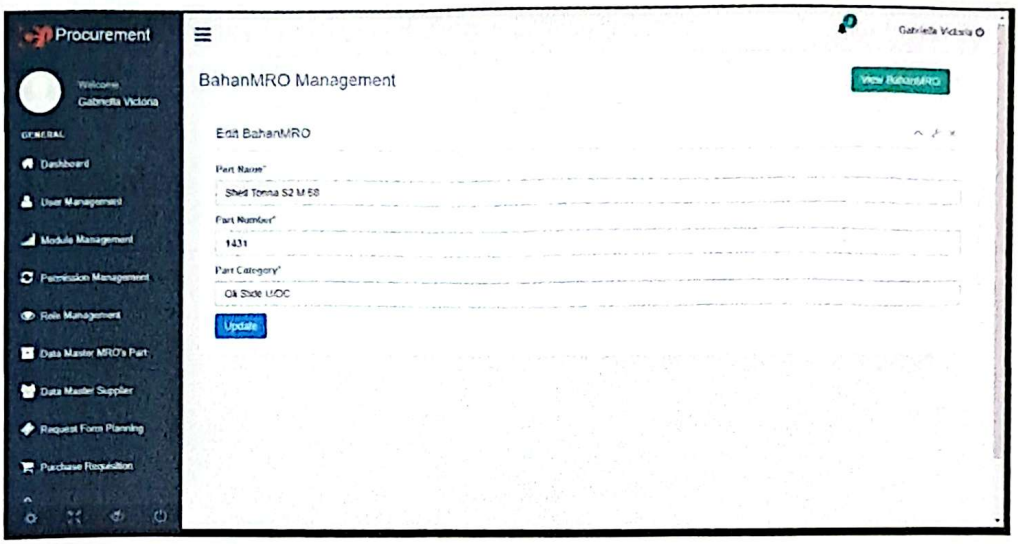


Gambar V.33. Tampilan *Form Data Master* Bahan MRO
(Sumber: Hasil Analisis, 2019)

Dipindai dengan CamScanner

3. *Data* Bahan MRO *Edit Form*

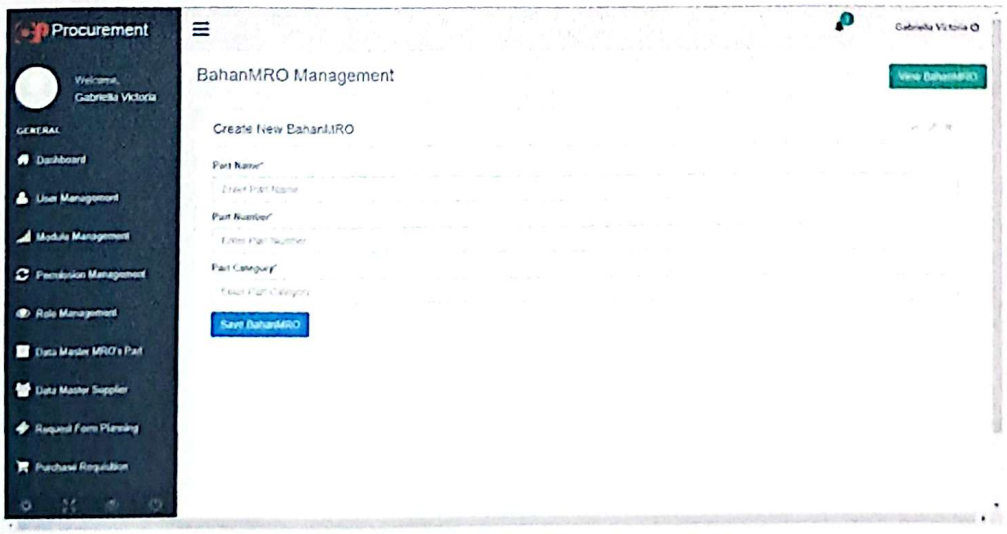
Form dibawah ini merupakan *form edit* bahan MRO, digunakan untuk mengubah data bahan MRO yang sudah ada.



Gambar V.34. Tampilan *Form Edit Data Master* Bahan MRO
(Sumber: Hasil Analisis, 2019)

4. *New Data* Bahan MRO *Form*

Form dibawah ini merupakan *form new* bahan MRO, digunakan untuk menambah data bahan MRO.

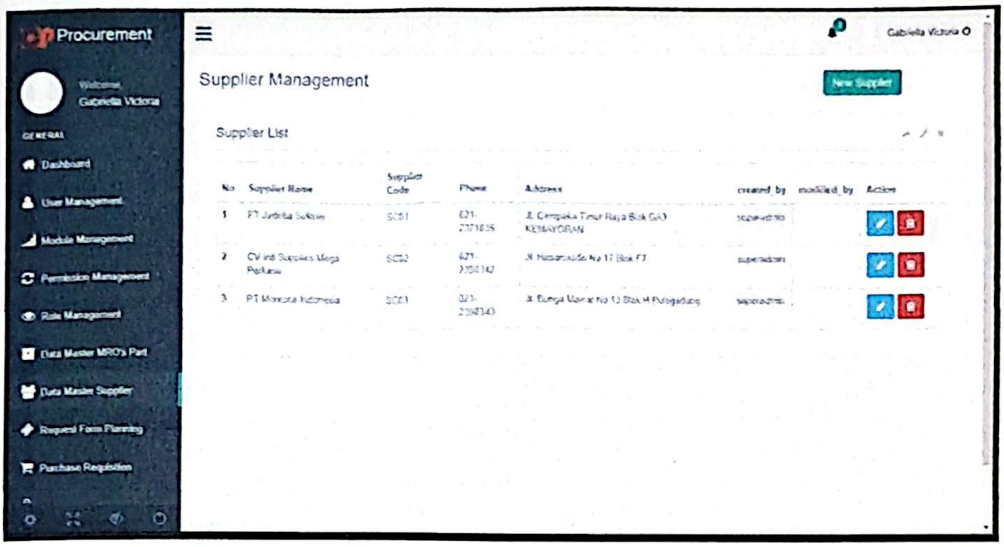


Gambar V.35. Tampilan *New Data* Bahan MRO *Form*
(Sumber: Hasil Analisis, 2019)

Dipindai dengan CamScanner

5. *Data Master Supplier*

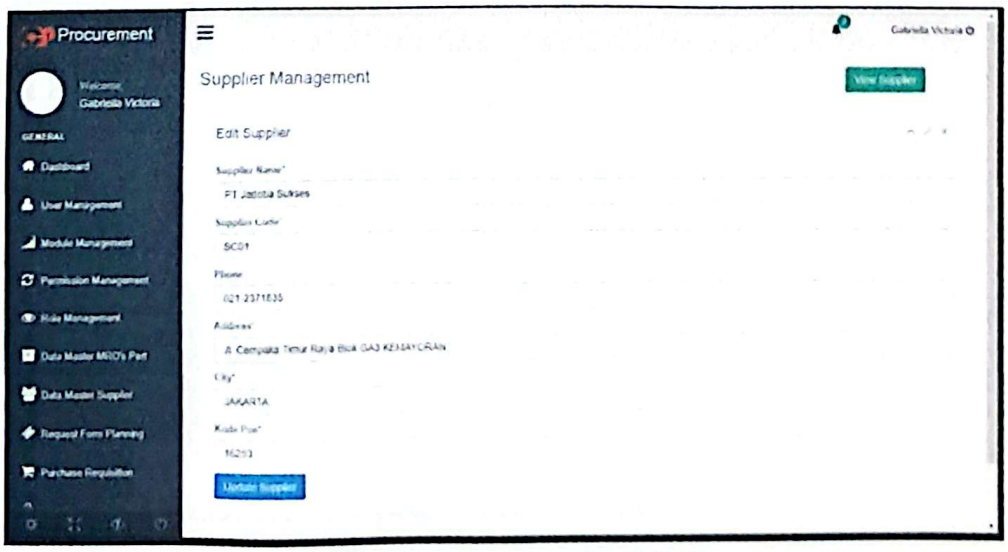
Form ini digunakan untuk mengelola data *supplier* yang ada. Kegiatan pengelolaan yang dapat dikerjakan seperti menambah *supplier* baru, mengubah data *supplier*, dan menghapus data *supplier*.



Gambar V.36. Tampilan *Form Data Master Supplier*
(Sumber: Hasil Analisis, 2019)

6. *Data Supplier Edit Form*

Form dibawah ini digunakan untuk mengubah data *supplier*. Data *supplier* yang baru akan menggantikan data *supplier* yang lama. Data yang lama akan terhapus.

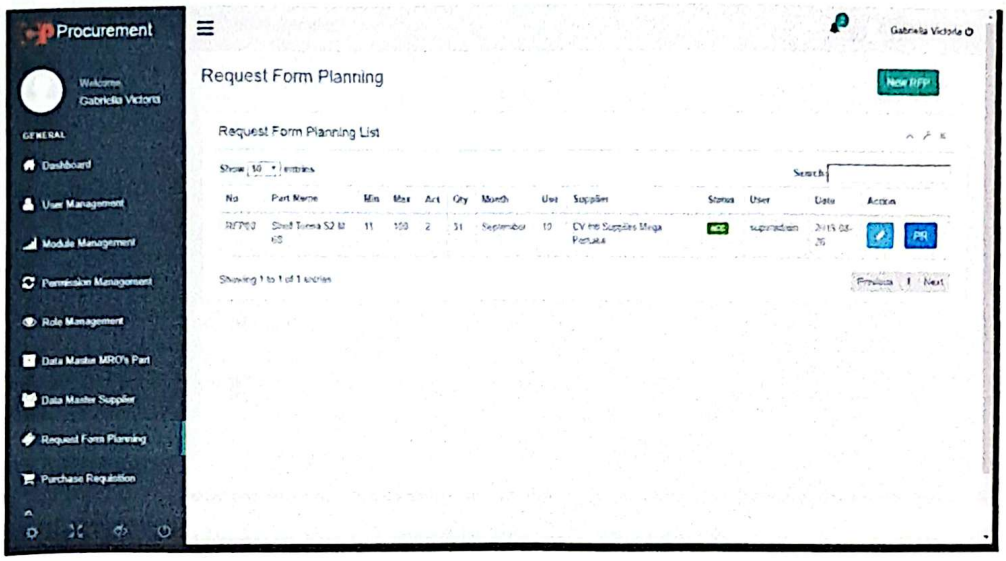


Gambar V.37. Tampilan *Data Supplier Edit Form*
(Sumber: Hasil Analisis, 2019)

Dipindai dengan CamScanner

7. *Request Form Planning*

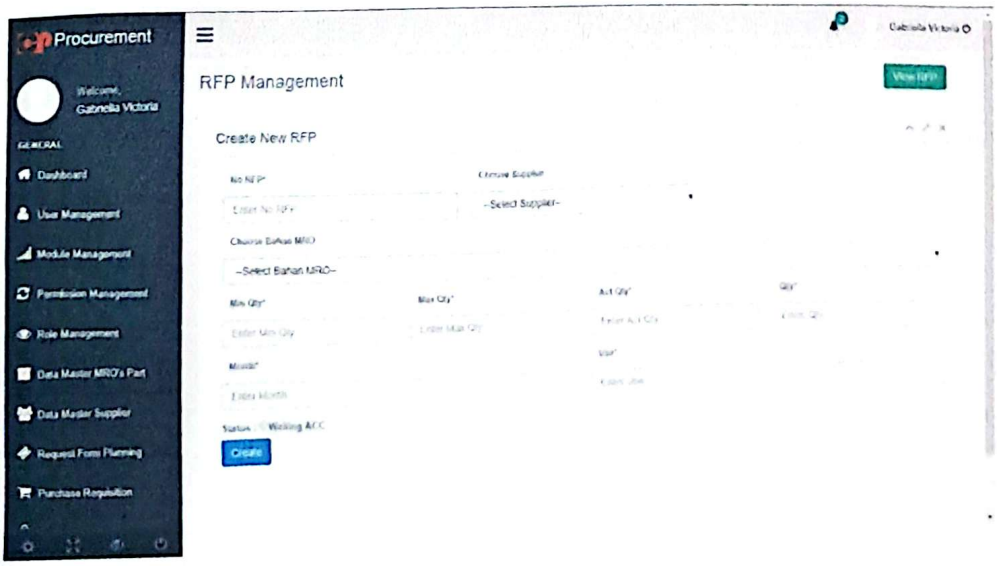
Form ini digunakan ketika *Maintenance Staff* ingin merencanakan pemesanan bahan MRO. *Maintenance Staff* akan menyerahkannya kepada *Maintenance Manager* untuk disetujui.



Gambar V.38. Tampilan *Request Form Planning*
(Sumber: Hasil Analisis, 2019)

8. *New Request Form Planning*

Form ini digunakan ketika *Maintenance Staff* mau membuat RFP.

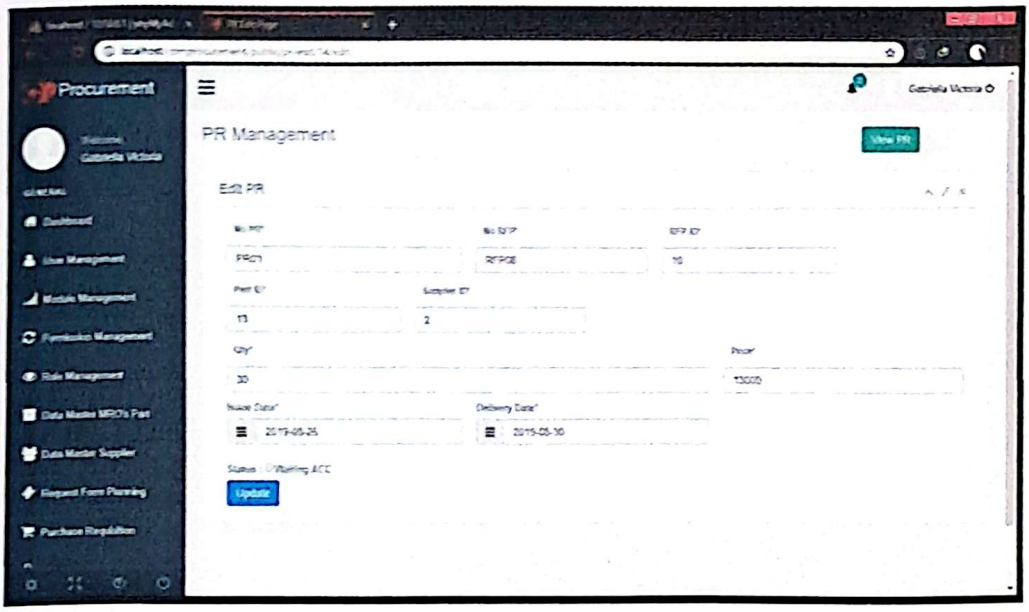


Gambar V.39. Tampilan *New Request Form Planning*
(Sumber: Hasil Analisis, 2019)

Dipindai dengan CamScanner

9. *Purchase Requisition Edit Form*

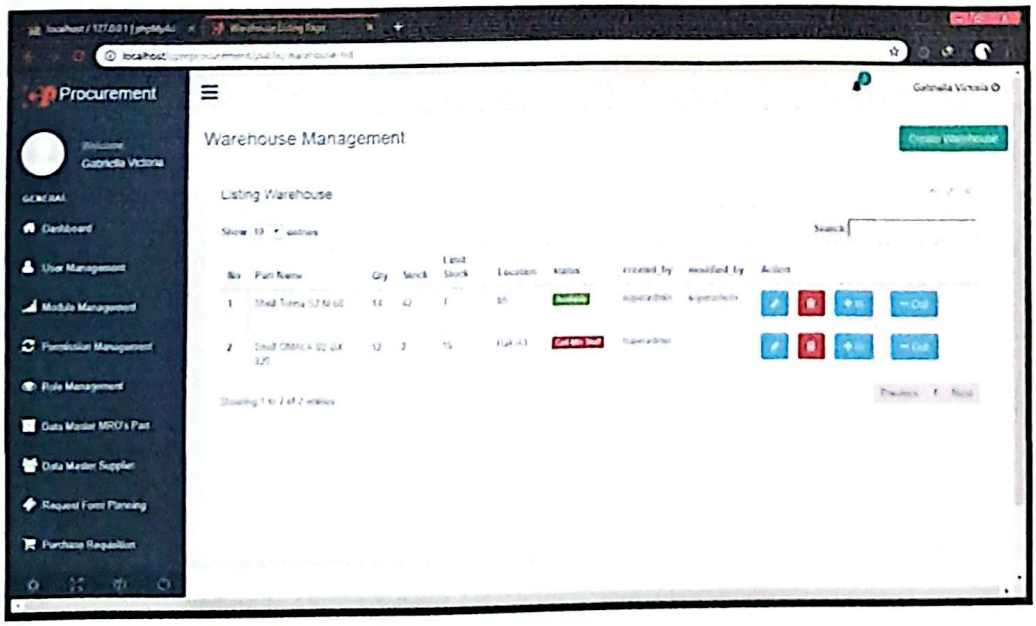
Form ini digunakan untuk mengubah data PR.



Gambar V.40. Tampilan *Purchase Requisition*
(Sumber: Hasil Analisis, 2019)

10. *Warehouse Stock*

Form ini digunakan untuk mendokumentasikan data bahan MRO yang masuk dan keluar oleh *Warehouse Staff.*

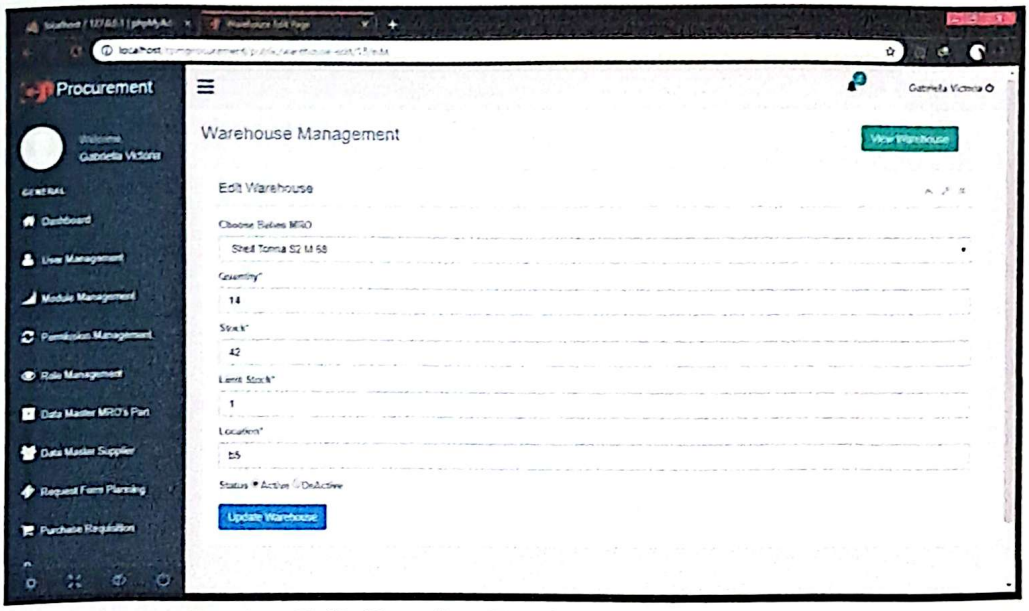


Gambar V.41. Tampilan *Warehouse*
(Sumber: Hasil Analisis, 2019)

Dipindai dengan CamScanner

11. *Warehouse Edit Form*

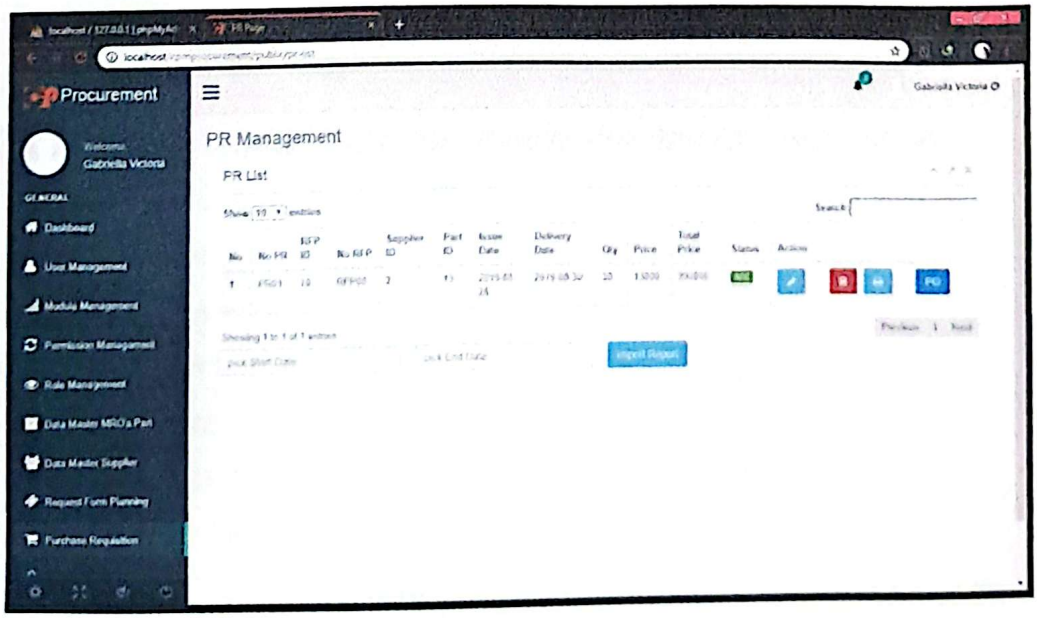
Form ini digunakan untuk mengubah data *Warehouse*.



Gambar V.42. Tampilan *Warehouse Edit Form*
(Sumber: Hasil Analisis, 2019)

12. *Purchase Requisition Form*

Form ini digunakan untuk memesan bahan MRO yang telah diajukan RFP.

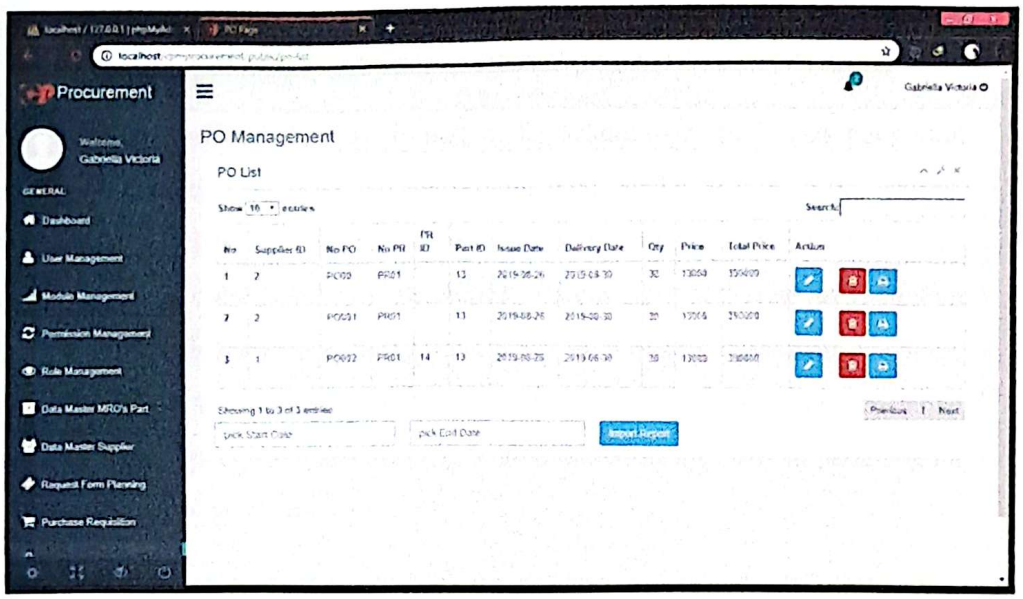


Gambar V.43. Tampilan *Purchase Requisition Form*
(Sumber: Hasil Analisis, 2019)

Dipindai dengan CamScanner

13. *Purchase Order Form*

Form ini dibuat oleh *Purchasing* untuk memesan bahan MRO kepada *supplier*.



Gambar V.44. Tampilan *Purchase Order Form*
(Sumber: Hasil Analisis, 2019)

5.6. Implementasi Sistem *Software* dan *Hardware*

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. Analisis kebutuhan *software*
   a. Sistem operasi : Minimal *Microsoft Windows* 7
   b. *Database server* : MariaDB versi 10.1.31.
   c. Bahasa pemrograman : PHP.
   d. *Web Server* : *Apache.*
   e. *Web Browser* : *Mozilla Firefox, Google Chrome*
2. Analisis kebutuhan *hardware*
   a. *Processor* : Minimal *processor pentium* IV.
   b. RAM : Minimal RAM 2 GB.
   c. *Harddisk* : Minimal 100 GB.

Dipindai dengan CamScanner

MILIK PERPUSTAKAAN STMI
Membaca : Ibadah, Mengambil : Dosa

BAB VI
KESIMPULAN DAN SARAN

6.1. Kesimpulan

Berdasarkan hasil rancang bangun sistem yang dilakukan mengenai sistem informasi pengadaan bahan MRO di PT Cikarang Perkasa Manufacturing dapat disimpulkan beberapa hal yaitu:

1. Dengan dibuatnya sistem informasi pengadaan bahan MRO *user* dapat mengetahui *stock* bahan MRO yang masuk dan keluar sehingga tidak diperlukan pengecekan langsung jumlah bahan MRO setiap kali informasi *warehouse stock* dibutuhkan.
2. Dengan adanya sistem informasi pengadaan bahan MRO data *Purchase Requisition* akurat dengan data *Purchase Order* sehingga tidak ditemukan perbedaan data bahan MRO antara *Purchase Requisition* dan *Purchase Order*.
3. Dengan adanya sistem informasi pengadaan bahan MRO distribusi data antara *warehouse*, *maintenance*, dan *purchasing* disimpan didalam *database* tanpa perlu pencarian data di tumpukan file.

6.2. Saran

Saran yang dapat diberikan untuk pengembangan sistem informasi pengadaan bahan MRO ini selanjutnya adalah diharapkan sistem ini dapat di implementasikan pada proses pengadaan bahan MRO di bagian *maintenance*, *warehouse*, dan *purchasing* pada PT Cikarang Perkasa Manufacturing.

Dipindai dengan CamScanner

DAFTAR PUSTAKA

Abdulloh, R. (2017). *Membuat Aplikasi Point of Sale dengan Laravel dan Ajax.* Jakarta: PT Elex Media Komputindo.

Ansori, N., & Mustajib, M. I. (2013). *Sistem Perawatan Terpadu.* Yogyakarta: Graha Ilmu.

Atzeni, P., Ceri, S., Paraboschi, S., & Torlone, R. (2003). *Database Systems.* Australia: Mc Graw Hill.

Connolly, T. M., & Begg, C. E. (2010). *Database Systems.* Boston: Pearson.

Dennis, A., Wixom, B. H., & Tegarden, D. (2010). *Systems Analysis and Design with UML.* United States: John Wiley & Sons, Inc.

Dhillon, B. (2006). *Maintainability, Maintenance, and Reliability for Engineers.* New York: Taylor and Francis Group.

Fowler, M. (2005). *UML Distilled 3th E, Panduan Singkat Bahasa Pemodelan Objek Standar.* Yogyakarta: Andi.

Haughee, E. (2013). *Sublime Text Starter.* Birmingham: PACKT Publishing.

Hutahean, J. (2014). Konsep Sistem Informasi. Yogyakarta: Deepublish.

Indonesia-Investments. (2017, Juli 12). *Industri Manufaktur Otomotif Indonesia.* Retrieved Oktober 24, 2018, from Indonesia-Investments: https://www.indonesia-investments.com/id/bisnis/industri-sektor/otomotif/item6047?

Jogiyanto. (2005). *Sistem Teknologi Informasi.* Yogyakarta: Andi.

Jogiyanto. (2010). *Analisis dan Desain Sistem Informasi.* Yogyakarta: Andi.

Kidd, A. (2005). *Definition of Procurement.* Australia: CIPS Australia.

Komarudin. (2001). Ensiklopedia Manajemen . Jakarta: Bumi Aksara.

Komputer, W. (2015). *Webmaster Menguasai CSS.* Yogyakarta: CV Andi Offset.

Ladjamudin, A.-B. b. (2013). *Analisis dan Desain Sistem Informasi.* Yogyakarta: Graha Ilmu.

Dipindai dengan CamScanner

Laudon, K. C., & Laudon, J. P. (2007). *Management Information System.* Jakarta: Salemba Empat.

Manufacturing, P. C. (2017). *Plan Book Mold Dies.* Bekasi: PT. Cikarang Perkasa Manufacturing.

Marbun, R. (2010). *Tanya Jawab Seputar Tata Cara Pengadaan Barang/Jasa Pemerintah.* Jakarta: Visi Media.

Mulyadi. (2010). Sistem Akuntansi. Jakarta: Salemba Empat.

Mulyanto, A. (2009). *Sistem Informasi Konsep dan Aplikasi.* Yogyakarta: Pustaka Pelajar.

Mustakini. (2009). *Sistem Informasi Teknologi.* Yogyakarta: Andi.

Neef, D. (2001). *E-Procurement: From Strategy to Implementation.* Upper Sadle River: Prentice-Hall.

Palmius, J. (2005). *Defining the Information Part of Information System.* Retrieved 12 15, 2018, from www.diva-portal.org/smash/get/diva2.../FULLTEXT01.pdf.

Pressman, R. S., & Bruce, M. R. (2015). *Software Engineering A Practitioner's Approach.* Singapore: Mc Graw Hill.

Riyanto. (2010). *Sistem Informasi Penjualan dengan PHP dan MySQL.* Yogyakarta: Gava Media.

Rusdiana, & Irfan, M. (2014). *Sistem Informasi Manajemen.* Bandung: Pustaka Setia.

Samsul, R. (2011). *Pengadaan Barang dan Jasa Pemerintah.* Jakarta: Visi Media.

Saputra, A. (2011). *Trik dan Solusi Jitu Pemrogaman PHP.* Jakarta: PT. Elex Media Komputindo.

Satzinger, J. W., Burd, S. D., & Jackson, R. B. (2012). *Introduction to System Analysis and Design.* USA: Course Technology, Cengage Learning.

Schach, S. R. (2007). *Object-Oriented & Classical Software Engineering.* New York: Mc. Graw Hill.

Siahaya, W. (2012). *Manajemen Pengadaan Procurement Management.* Bandung: Alfabeta.

Dipindai dengan CamScanner

DAFTAR LAMPIRAN

LAMPIRAN A

RfpController.php (Controller)

```
<?php
namespace App\Http\Controllers\backend;
use Illuminate\Http\Request;
use App\Http\Controllers\Controller;
use App\Models\Requestfp;
use App\Models\Bahanmros;
use App\Models\Supplier;
use Illuminate\Support\Facades\Auth;
use Session;
use DB;

class RfpController extends Controller
{
   public function index()
   {
      $this->checkpermission('rfp-list');
      $requestfps =DB::table('requestfps')
            ->join('bahanmros', 'requestfps.bahanmros_id', '=', 'bahanmros.id')
            ->join('suppliers', 'suppliers.id', '=', 'requestfps.supplier_id')
            ->select('requestfps.id','requestfps.norfps','bahanmros.partname','bahanmros.partnumber',
'suppliers.suppliername', 'suppliers.suppliercode', 'requestfps.minquantity',
'requestfps.maxquantity', 'requestfps.actquantity', 'requestfps.qty', 'requestfps.month',
'requestfps.use', 'requestfps.status', 'requestfps.created_by', 'requestfps.created_at')
            ->get();
      return view('backend.rfp.list', compact('requestfps', 'bahanmros','suppliers'));
   }

   public function create()
   {
      $this->checkpermission('rfp-create');
      $bahanmros = Bahanmros::all();
      $suppliers = Supplier::all();
      return view('backend.rfp.create', compact('bahanmros','suppliers'));
   }

   public function store(Request $request)
   {$this->validate($request, [
         'norfps' => 'required',
         'bahanmros_id' => 'required',]);
      $message = Requestfp::create([
```


Dipindai dengan CamScanner

```
            'norfps' => $request->norfps,
            'bahanmros_id' => $request->bahanmros_id,
            'minquantity' => $request->minquantity,
            'maxquantity' => $request->maxquantity,
            'actquantity' => $request->actquantity,
            'qty' => $request->qty,
            'month' => $request->month,
            'use' => $request->use,
            'supplier_id' => $request->supplier_id,
            'status' => $request->status,
            'created_by' => Auth::user()->username,
            'created_at' => date('Y-m-d H:i:s'),
        ]);
        if ($message) {return redirect()->route('rfp.list')->with('success_message', 'successfully
created');} else {return redirect()->route('rfp.create')->with('error_message', 'Failed To create');
        }
    }

    public function edit($id)
    {$this->checkpermission('rfp-edit');
        $requestfps = Requestfps::find($id);
        $bahanmros = Bahanmros::all();
        $suppliers = Supplier::all();
        return view('backend.rfp.edit', compact('requestfps', 'bahanmros','suppliers'));
    }

    public function update(Request $request)
    {DB::table('requestfps')->where('id',$request->id)->update([
        'norfps' => $request->norfps,
        'bahanmros_id' => $request->bahanmros_id,
        'minquantity' => $request->minquantity,
        'maxquantity' => $request->maxquantity,
        'actquantity' => $request->actquantity,
        'qty' => $request->qty,
        'month' => $request->month,
        'use' => $request->use,
        'supplier_id' => $request->supplier_id,
        'status' => $request->status,
        'modified_by' => Auth::user()->username,
        'updated_at' => date('Y-m-d H:i:s')
        ]);
            return redirect()->route('rfp.list')->with('success_message', 'successfully updated');
    }

    public function destroy($id)
    {$check = $this->checkpermission('rfp-delete');
        if ($check) { $this->checkpermission('rfp-delete');
        } else {
```


Dipindai dengan CamScanner

```
        $requestfix = Requestfix::find($id);
        $message = $requestfix->delete();
        if ($message) {return redirect()->route('rfp_list')->with('success_message', 'Successfully
Deleted');
        } else {return redirect()->route('rfp_update')->with('error_message', 'Failed to Delete');
        }
    }
}
```

Requestfix.php (Model)

```
<?php
namespace App\Models;
use Illuminate\Database\Eloquent\Model;
class Requestfix extends Model
{ protected $fillable = ['rmp', 'requestfix_id', 'notes', 'balanmos_id', 'minquantity', 'maxquantity',
'acquantity', 'qty', 'month', 'use', 'supplier_id', 'status', 'created_by', 'modified_by', 'created_at',
'updated_at'];}
```

create.blade.php (View)

```
@extends('backend.layouts.master')
@section('title')
   New RFP Page
@endsection
@section('css')
@endsection
<!-- page content -->
@section('content')
   <div class="right_col" role="main">
      <div class="">
         <div class="page-title">
            <div class="title_left">
               <h3>RFP Management </h3>
            </div>
            <div class="title_right">
               <div class="col-md-5 col-sm-5 col-xs-12 form-group pull-right top_search">
                  <div class="col-md-5 col-sm-5 col-xs-12 form-group top_search" style="padding-
left: 15%px;">
                     <div class="input-group">
                        <a href="{{route('rfp_list')}}" class="btn btn-success">View RFP</a>
                     </div>
                  </div>
               </div>
            </div>
         </div>
         <div class="clearfix"></div>
         @if(Session::has('success_message'))
            <div class="alert alert-success">
               {{ Session::get('success_message') }}
```


Dipindai dengan CamScanner

```
                    <label class="col-md-5 input-sm control-label" for="supplier_id">Choose
Supplier</label>
                    <select class="form-control" id="supplier_id" name="supplier_id">
                      <option value="">--Select Supplier--</option>
                      @foreach($suppliers as $m)
                        <option value="{{$m->id}}">{{$m->suppliername}}</option>
                      @endforeach
                    </select>
                    <span class="error"><b>
                      @if($errors->has('supplier_id'))
                          {{$errors->first('supplier_id')}}
                        @endif</b>
                    </span>
                  </div>

                  <div class="col-md-12 form-group">
                    <label class="col-md-12 input-sm control-label"
for="bahanmros_id">Choose Bahan MRO</label>
                    <select class="form-control" id="bahanmros_id" name="bahanmros_id">
                      <option value="">--Select Bahan MRO--</option>
                      @foreach($bahanmros as $m)
                        <option value="{{$m->id}}">{{$m->partname}}</option>
                      @endforeach
                    </select>
                    <span class="error"><b>
                      @if($errors->has('bahanmros_id'))
                          {{$errors->first('bahanmros_id')}}
                        @endif</b>
                    </span>
                  </div>

                  <div class="col-md-3 form-group">
                    <label class="col-md-4 input-sm control-label" for="minquantity">Min
Qty*</label>
                    <input type="text" class="form-control" id="minquantity"
name="minquantity" placeholder="Enter Min Qty">
                    <div class="col-md-3">
                    <span class="error"><b>
                        @if($errors->has('minquantity'))
                          {{$errors->first('minquantity')}}
                        @endif</b>
                    </span>
                    </div>
                  </div>

                  <div class="col-md-3 form-group">
                    <label class="col-md-4 input-sm control-label" for="maxquantity">Max
Qty*</label>
```


Dipindai dengan CamScanner

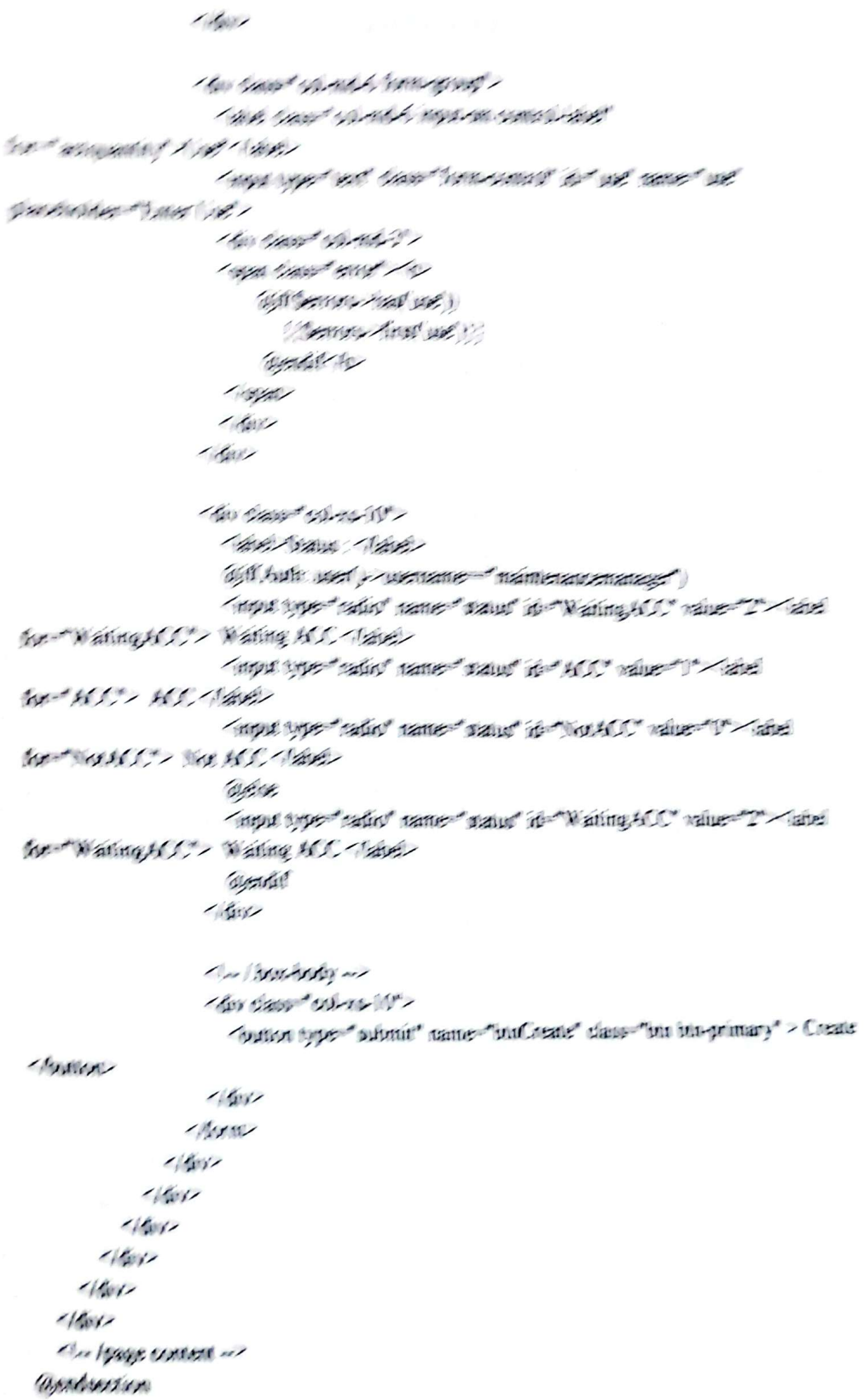
```
                    <input type="text" class="form-control" id="maxquantity"
name="maxquantity" placeholder="Enter Max Qty">
                    <div class="col-md-3">
                    <span class="error"><b>
                        @if($errors->has('maxquantity'))
                            {{$errors->first('maxquantity')}}
                        @endif</b>
                    </span>
                    </div>
                  </div>

                  <div class="col-md-3 form-group">
                    <label class="col-md-4 input-sm control-label" for="actquantity">Act
Qty*</label>
                    <input type="text" class="form-control" id="actquantity"
name="actquantity" placeholder="Enter Act Qty">
                    <div class="col-md-3">
                    <span class="error"><b>
                        @if($errors->has('actquantity'))
                            {{$errors->first('actquantity')}}
                        @endif</b>
                    </span>
                    </div>
                  </div>

                  <div class="col-md-3 form-group">
                    <label class="col-md-2 input-sm control-label" for="qty">Qty*</label>
                    <input type="number" class="form-control" id="qty" name="qty"
placeholder="Enter Qty">
                    <span class="error"><b>
                        @if($errors->has('qty'))
                            {{$errors->first('qty')}}
                        @endif</b>
                    </span>
                  </div>

                  <div class="col-md-6 form-group">
                    <label class="col-md-6 input-sm control-label"
for="month">Month*</label>
                    <input type="text" class="form-control" id="month" name="month"
placeholder="Enter Month">
                    <div class="col-md-6">
                    <span class="error"><b>
                        @if($errors->has('month'))
                            {{$errors->first('month')}}
                        @endif</b>
                    </span>
                    </div>
```


Dipindai dengan CamScanner



Dipindai dengan CamScanner

LAMPIRAN B

1. *Functional Test Use Case* melakukan *Login*

Berikut merupakan *functional test use case* melakukan *login* yang dapat dilihat sebagai berikut:

Test case ID : Login 001

Function : Operasi validasi saat melakukan *login*

Data Assumption :Fungsi operasi validasi saat melakukan login sudah berjalan dengan baik

Deskripsi : Melakukan *login* ke dalam sistem dengan menguji kesalahan*username* dan *password*

Tabel 1 *Test Case* melakukan *Login*

Test ID	***Test Case Name***	***Description***	***Expected Result***	***Actual Result***
001	Validasi *Login*	*Username* dan *password* tidak diisi dan klik tombol *Login*	Gagal masuk dan kembali ke *Login Page*	Gagal masuk dan kembali ke *Login Page*
002	Validasi *Login*	Mengisi *username* salah dan *password* benar atau sebaliknya	Muncul pesan "*Invalid username or password*"	Muncul pesan "*Invalid username or password*"
003	Validasi *Login*	Mengisi *username* benar dan *password* benar	Masuk ke *dashboard*	Masuk ke *dashboard*

(Sumber: Hasil Analisis, 2019)

2. *Functional Test Use Case* mengakses Menu Data Master Bahan MRO

Berikut merupakan *functional test use case* mengakses menu data master bahan MRO yang dapat dilihat sebagai berikut:

Test case ID : MasterBahanMRO 001

Function : Operasi menambah data bahan MRO
Operasi menghapus data bahan MRO
Operasi mengubah data bahan MRO

Data Assumption : Fungsi yang ada pada menu data master bahan MRO sudah berjalan dengan baik

Deskripsi : Mengakses seluruh *function* yang ada pada menu data master bahan MRO

Dipindai dengan CamScanner

Tabel 2 Test Case mengakses Menu Data Master Bahan (MB)

Test ID	Test Case Name	Description	Expected Result	Actual Result
MB	Tambah Bahan (MB)	- Klik tombol tambah bahan (MB)	Masuk ke form Create New Bahan (MB)	Masuk ke form Create New Bahan (MB)
		- Klik tombol menambahkan data dan mengisi button save	Tampil message "The field is required"	Tampil message "The field is required"
		- Klik menambahkan data pada semua field dan mengisi button save	Sistem menyimpan data dalam database dan muncul message "Successfully created"	Sistem menyimpan data dalam database dan muncul message "Successfully created"
MB	Hapus Bahan (MB)	- Menekan button delete dan menekan button OK pada confirm message	Muncul pesan "are you sure want to delete" dan sistem menghapus data dari database dan muncul confirm message "successfully deleted"	Muncul pesan "are you sure want to delete" dan sistem menghapus data dari database dan muncul confirm message "successfully deleted"
		- Menekan button delete dan menekan button cancel pada confirm message	Muncul pesan "are you sure want to delete" dan sistem kembali menampilkan BahanMB() list	Muncul pesan "are you sure want to delete" dan sistem kembali menampilkan BahanMB() list
MB	Edit Bahan (MB)	- Klik tombol edit	Masuk ke Form Edit BahanMB()	Masuk ke Form Edit BahanMB()
		- Klik mengubah data dan mengisi button submit	Sistem menyimpan data dalam database dan muncul message "Successfully updated"	Sistem menyimpan data dalam database dan muncul message "Successfully updated"

(Sumber: Hasil Analisa, 2019)

3. Functional Test Use Case mengakses Menu Data Warehouse Stock

Berikut merupakan functional test use case mengakses menu data warehouse yang dapat dilihat sebagai berikut:

Test case ID : MasterWarehouse (01)

Function : Operasi menambah data warehouse

Operasi menghapus data warehouse

Operasi mengubah data warehouse

Operasi stock in data warehouse

Operasi stock out data warehouse

Dipindai dengan CamScanner

Data Assumption : Fungsi yang ada pada menu data master warehouse sudah berjalan dengan baik

Deskripsi : Mengakses seluruh *function* yang ada pada menu data master warehouse

Tabel 3 *Test Case* mengakses Menu Data Warehouse Stock

Test ID	*Test Case Name*	*Description*	*Expected Result*	*Actual Result*
001	Tambah Data Warehouse	- *Actor* klik *button New* Warehouse	Masuk ke *Form Create New Warehouse*	Masuk ke *Form Create New Warehouse*
		- *Actor* tidak memasukkan data dan mengklik *button save*	Tampil message "*The field is required*"	Tampil message "*The field is required*"
		- *Actor* memasukkan data pada seluruh *field* dan mengklik *button save*	Sistem menyimpan data dalam database dan muncul message "*Successfully created*"	Sistem menyimpan data dalam database dan muncul message "*Successfully created*"
002	Hapus Data Warehouse	- Menekan *button delete* dan menekan *button* OK pada *confirm message*	Muncul pesan "*are you sure want to delete*" dan sistem menghapus data dari *database* dan muncul *confirm message* "*successfully deleted*"	Muncul pesan "*are you sure want to delete*" dan sistem menghapus data dari *database* dan muncul *confirm message* "*successfully deleted*"
		- Menekan *button delete*	Muncul pesan "*are you sure want to delete*"	Muncul pesan "*are you sure want to delete*"
003	Edit Data Warehouse	- *Actor* klik *button edit*	Masuk ke *Form Edit* Warehouse	Masuk ke *Form Edit* Warehouse
		- *Actor* mengubah data dan mengklik *button submit*	Sistem menyimpan data dalam database dan muncul message "*Successfully updated*"	Sistem menyimpan data dalam database dan muncul message "*Successfully updated*"
004	*Stock In* Bahan MRO	- *Actor* klik *button In*	Masuk ke *Update Stock Form*	Masuk ke *Update Stock Form*
		- *Actor* memasukkan data pada *fieldstock in* dan klik *button update*	Jumlah pada field *Qty* dan *Stock* bertambah dan muncul *message* "*Successfully updated your stock*"	Jumlah pada field *Qty* dan *Stock* bertambah dan muncul *message* "*Successfully updated your stock*"
005	*Stock Out* bahan MRO	- *Actor* klik *button Out*	Masuk ke *Update Stock Form*	Masuk ke *Update Stock Form*
		- *Actor* memasukkan data pada *field* yang ada dan klik *update*	Sistem menyimpan data ke *database* dan muncul message "*Successfully updated your stock*"	Sistem menyimpan data ke *database* dan muncul message "*Successfully updated your stock*"

(Sumber: Hasil Analisis, 2019)

Dipindai dengan CamScanner

unctional Test Use Case mengakses menu Request Form Planning

erikut merupakan functional test use case mengakses menu Request Form Planning yang dapat dilihat sebagai berikut:

Test case ID : *RFP001*

Function : Operasi membuat *Purchase Requisition*

Operasi membuat *Request Form Planning*

Operasi mengubah data *Request Form Planning*

Operasi menghapus data *Request Form Planning*

Data Assumption : Fungsi yang ada pada menu *Request Form Planning* sudah berjalan dengan baik

Deskripsi : Mengakses seluruh *function* yang ada pada menu *Request Form Planning*

Tabel 4 *Test Case* mengakses Menu *Request Form Planning*

Test ID	*Test Case Name*	*Description*	*Expected Result*	*Actual Result*
001	*Make PR*	- *Actor* klik *Make PR*	Masuk ke *New Purchase Requisition Form* dan mengambil nilai dari *field* yang ada pada *RFP*	Masuk ke *New Purchase Requisition Form* dan mengambil nilai dari *field* yang ada pada *RFP*
		- *Actor* klik button *Create* pada *Purchase Requisition Form*	Tampil *message* "*Successfully created PR*" dan sistem menyimpan dalam *database*	Tampil *message* "*Successfully created PR*" dan sistem menyimpan dalam *database*
002	Edit *RFP*	- Menekan *button edit*	Masuk ke *Form Edit RFP*	Masuk ke *Form Edit RFP*
		- Mengubah data *quantity* dan klik *button Submit*	Muncul *message* "*Successfully updated RFP*" dan sistem menyimpan dalam *database*	Muncul *message* "*Successfully updated RFP*" dan sistem menyimpan dalam *database*
003	*Delete RFP*	- *Actor* klik *button delete*	Muncul *confirm message* "*are you sure want to cancel your RFP?*"	Muncul *confirm message* "*are you sure want to cancel your RFP?*"

(Sumber: Hasil Analisis, 2019)

Dipindai dengan CamScanner

5. *Functional Test Use Case* mengakses menu *Purchase Requisition*

Berikut merupakan *functional test use case* mengakses menu data master bahan MROyang dapat dilihat sebagai berikut:

Test case ID : PR001

Function : Operasi membuat *Purchase Order*
Operasi mengubah *Purchase Requisition*
Operasi menghapus *Purchase Requisition*

Data Assumption : Fungsi yang ada pada menu *Purchase Requisition* sudah berjalan dengan baik

Deskripsi : Mengakses seluruh *function* yang ada pada menu *Purchase Requisition*

Tabel 5*Test Case* mengakses Menu *Purchase Requisition*

Test ID	*Test Case Name*	*Description*	*Expected Result*	*Actual Result*
001	*Make PO*	- *Actor* klik *button Make PO*	Masuk ke *Form Purchase Order* dan mengambil nilai dari *field* yang ada pada *Purchase Requisition*	Masuk ke *Form Purchase Order* dan mengambil nilai dari *field* yang ada pada *Purchase Requisition*
002	*Delete PR*	- Menekan *buttondelete* danmenekan *button* OK pada *confirm message*	Muncul pesan "*are you sure want to delete*" dan sistem menghapus data dari *database* dan muncul *confirm message* "*successfully deleted*"	Muncul pesan "*are you sure want to delete*" dan sistem menghapus data dari *database* dan muncul *confirm message* "*successfully deleted*"
		- Menekan *button delete* dan menekan *button cancel* pada *confirm message*	Muncul pesan "*are you sure want to delete*" dan sistem kembali menampilkan *Purchase Requisition List*	Muncul pesan "*are you sure want to delete*" dan sistem kembali menampilkan *Purchase Requisition List*
003	*Edit PR*	- *Actor* klik *button edit*	Masuk ke *Form Edit Purchase Requisition*	Masuk ke *Form Edit Purchase Requisition*
		- *Actor* mengubah data dan mengklik *button submit*	Sistem menyimpan data dalam *database* dan muncul message "*Successfully updated*"	Sistem menyimpan data dalam *database* dan muncul message "*Successfully updated*"

(Sumber: Hasil Analisis, 2019)

Dipindai dengan CamScanner

6. *Functional Test Use Case* membuat *Purchase Order*

Berikut merupakan *functional test use case* mengakses menu *Purchase Order*yang dapat dilihat sebagai berikut:

Test case ID : PO 001

Function : Operasi mencetak *Purchase Order*
Operasi menghapus *Purchase Order*
Operasi mengubah *Purchase Order*

Data Assumption : Fungsi yang ada pada menu *Purchase Order* sudah berjalan dengan baik

Deskripsi : Mengakses seluruh *function* yang ada pada menu *Purchase Order*

Tabel 6 *Test Case* mengakses Menu *Purchase Order*

Test ID	*Test Case Name*	*Description*	*Expected Result*	*Actual Result*
001	*Print PO*	- *Actor* klik *button Print PO*	Sistem mencetak PO dan muncul *message "Successfully printed Purchase Order"*	Sistem mencetak PO dan muncul *message "Successfully printed Purchase Order"*
002	*DeletePO*	- Menekan *button delete* dan menekan *button* OK pada *confirm message*	Muncul pesan *"are you sure want to delete"* dan sistem menghapus data dari *database* dan muncul *confirm message "successfully deleted"*	Muncul pesan *"are you sure want to delete"* dan sistem menghapus data dari *database* dan muncul *confirm message "successfully deleted"*
		- Menekan *button delete* dan menekan *button cancel* pada *confirm message*	Muncul pesan *"are you sure want to delete"* dan sistem kembali menampilkan *Purchase Order List*	Muncul pesan *"are you sure want to delete"* dan sistem kembali menampilkan *Purchase Order List*
003	*Edit PO*	- *Actor* klik *button edit*	Masuk ke *Form Edit Purchase Order*	Masuk ke *Form Edit Purchase Order*
		- *Actor* mengubah data dan mengklik *button submit*	Sistem menyimpan data dalam *database* dan muncul message *"Successfully updated"*	Sistem menyimpan data dalam *database* dan muncul message *"Successfully updated"*

(Sumber: Hasil Analisis, 2019)

Dipindai dengan CamScanner

7. *Functional Test Use Case* mengelola Data Master *Supplier*

Berikut merupakan *functional test use case* mengakses menu data master bahan MROyang dapat dilihat sebagai berikut:

Test case ID : MasterSupplier 001

Function : Operasi menambah data bahan MRO
Operasi menghapus data bahan MRO
Operasi mengubah data bahan MRO

Data Assumption : Fungsi yang ada pada menu data master *supplier* sudah berjalan dengan baik

Deskripsi : Mengakses seluruh *function* yang ada pada menu data master *supplier*

Tabel 8 *Test Case* mengakses Menu Data Master *Supplier*

Test ID	*Test Case Name*	*Description*	*Expected Result*	*Actual Result*
001	Tambah *Supplier*	- *Actor* klik *button New Supplier*	Masuk ke *Form Create New* Bahan *Supplier*	Masuk ke *Form Create New* Bahan *Supplier*
		- *Actor* tidak memasukkan data dan mengklik *button save*	Tampil message "*The field is required*"	Tampil message "*The field is required*"
		- *Actor* memasukkan data pada seluruh *field* dan mengklik *button save*	Sistem menyimpan data dalam database dan muncul message "*Successfully created*"	Sistem menyimpan data dalam database dan muncul message "*Successfully created*"
002	Hapus *Supplier*	- Menekan *button delete* dan menekan *button* OK pada *confirm message*	Muncul pesan "*are you sure want to delete*", sistem menghapus data dari *database* dan muncul *confirm message* "*successfully deleted*"	Muncul pesan "*are you sure want to delete*", sistem menghapus data dari *database* dan muncul *confirm message* "*successfully deleted*"
		- Menekan *button delete* dan menekan *button cancel*	Muncul pesan "*are you sure want to delete*"	Muncul pesan "*are you sure want to delete*"
003	*Edit Supplier*	- *Actor* klik *edit*	Masuk ke *Form Edit*	Masuk ke *Form Edit*
		- *Actor* mengubah data dan mengklik *button submit*	Sistem menyimpan data dalam database dan muncul message "*Successfully updated*"	Sistem menyimpan data dalam database dan muncul message "*Successfully updated*"

(Sumber: Hasil Analisis, 2019)

Dipindai dengan CamScanner