

**PERANCANGAN SISTEM INFORMASI
MANAJEMEN PERAWATAN BERBASIS
*COMPUTERIZED MAINTENANCE
MANAGEMENT SYSTEM (CMMS) DAN
GROUP TECHNOLOGY*
(Studi Kasus: PT. Gunawan Dianjaya Steel, Tbk)**

SKRIPSI

KONSENTRASI MANAJEMEN SISTEM INDUSTRI

**Diajukan untuk memenuhi persyaratan
memperoleh gelar Sarjana Teknik**



Disusun oleh :

**REZA MAULANA
NIM 115060700111045-67**

**KEMENTERIAN RISET TEKNOLOGI DAN PENDIDIKAN TINGGI
UNIVERSITAS BRAWIJAYA
FAKULTAS TEKNIK
MALANG
2015**

LEMBAR PERSETUJUAN

PERANCANGAN SISTEM INFORMASI MANAJEMEN PERAWATAN BERBASIS *COMPUTERIZED MAINTENANCE MANAGEMENT SYSTEM (CMMS) DAN GROUP TECHNOLOGY*

(Studi Kasus: PT. Gunawan Dianjaya Steel, Tbk)

SKRIPSI

KONSENTRASI MANAJEMEN SISTEM INDUSTRI

Diajukan untuk memenuhi persyaratan
memperoleh gelar Sarjana Teknik



Disusun oleh :

REZA MAULANA
NIM 115060700111045-67

Telah diperiksa dan disetujui oleh :

Dosen Pembimbing I

Ir. Purnomo Budi Santoso, M.Sc., Ph.D.
NIP. 19530113 198303 1 003

Dosen Pembimbing II

Zefry Darmawan, ST., MT.
NIP. 830206 06 1 1 0248

LEMBAR PENGESAHAN

PERANCANGAN SISTEM INFORMASI MANAJEMEN PERAWATAN BERBASIS *COMPUTERIZED* *MAINTENANCE MANAGEMENT SYSTEM (CMMS) DAN* *GROUP TECHNOLOGY*

(Studi Kasus: PT. Gunawan Dianjaya Steel, Tbk)

SKRIPSI

KONSENTRASI MANAJEMEN SISTEM INDUSTRI

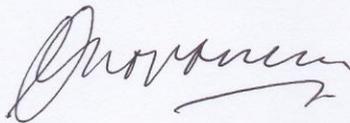
Diajukan untuk memenuhi persyaratan
memperoleh gelar Sarjana Teknik

Disusun oleh :

REZA MAULANA
NIM 115060700111045-67

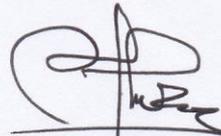
Skripsi ini telah diuji dan dinyatakan lulus pada
Tanggal 13 Agustus 2015

Penguji Skripsi 1



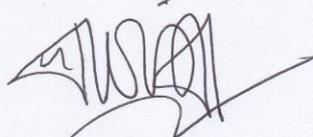
Oyong Novareza, ST., MT. Ph.D.
NIP. 19741115 200604 1 002

Penguji Skripsi 2



Ir. Mochamad Choiri, MT.
NIP. 19540104 198602 1 001

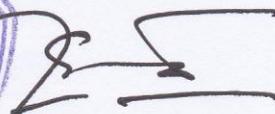
Penguji Skripsi 3



Ceria Farela Mada Tantrika, ST., MT.
NIP. 19840426 200812 2 002

Mengetahui,

Ketua Jurusan Teknik Industri



Ishardita Pambudi T., ST., MT., Ph.D.
NIP. 19730819 199903 1 002

PERNYATAAN ORISINALITAS SKRIPSI

Saya menyatakan dengan sebenar-benarnya bahwa sepanjang pengetahuan saya, di dalam naskah skripsi ini tidak terdapat karya ilmiah yang pernah diajukan oleh orang lain untuk memperoleh gelar akademik di suatu perguruan tinggi, dan tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan orang lain, kecuali yang secara tertulis dikutip dalam naskah ini dan disebutkan dalam sumber kutipan dan daftar pustaka.

Apabila ternyata didalam naskah skripsi ini dapat dibuktikan unsur-unsur plagiasi, saya bersedia skripsi ini digugurkan dan gelar akademik yang telah saya peroleh (SARJANA TEKNIK) dibatalkan serta diproses sesuai dengan peraturan perundang-undangan yang berlaku (UU No. 20 Tahun 2003, Pasal 25 ayat 2 dan Pasal 70).

Malang, Agustus 2015
Mahasiswa



Reza Maulana
NIM. 115060700111045

KATA PENGANTAR

Puji syukur kehadiran Tuhan Yang Maha Esa karena atas limpahan rahmat, taufik, dan hidayah-Nya penulis dapat menyelesaikan skripsi dengan judul **“Perancangan Sistem Informasi Manajemen Perawatan Berbasis *Computerized Maintenance Management System (CMMS)* dan *Group Technology*”**. Penyusunan laporan skripsi ini dilakukan sebagai persyaratan untuk memperoleh gelar Sarjana Teknik pada Jurusan Teknik Industri Fakultas Teknik Universitas Brawijaya. Laporan skripsi ini tentu tidak akan mencapai hasil yang maksimal tanpa adanya bantuan yang diberikan oleh berbagai pihak, maka dari itu penulis ingin mengucapkan banyak terima kasih kepada:

1. Bapak Ishardita Pambudi Tama, ST., MT., Ph.D. selaku Ketua Jurusan Teknik Industri Universitas Brawijaya Malang.
2. Bapak Ir. Purnomo Budi Santoso, M.Sc., Ph.D selaku Dosen Pembimbing I sekaligus Kepala Laboratorium Pemrograman Komputer yang telah dengan sabar membantu penulis dan memberikan banyak saran yang bermanfaat.
3. Bapak Zefry Darmawan, ST., MT. selaku Dosen Pembimbing II yang telah dengan sabar membantu penulis dan memberikan banyak saran yang bermanfaat.
4. Bapak Remba Yanuar Efranto, ST., MT. selaku Ketua Kelompok Dosen Keahlian (KKDK) Konsentrasi Manajemen Sistem Industri atas masukan dan arahnya dalam penentuan topik untuk skripsi penulis.
5. Ibu Ceria Farela Mada Tantrika, ST., MT. selaku Dosen Pembimbing Akademik yang dengan sabar selalu bersedia membimbing dan memberikan arahan akademik.
6. Seluruh Dosen Pengajar di Jurusan Teknik Industri Universitas Brawijaya yang telah memberikan ilmu yang bermanfaat bagi penulis.
7. Bapak Mugiarto, dan Bapak Suyanto, serta seluruh karyawan PT. Gunawan Dianjaya Steel, Tbk yang telah membantu penulis dalam penyelesaian skripsi ini.
8. Karyawan Jurusan Teknik Industri Universitas Brawijaya, Mas Reza, dan Mbak Ifa yang telah banyak membantu selama masa perkuliahan di Jurusan Teknik Industri ini.
9. Keluargaku tercinta; Bapak Bagus Ari Basuki, Mama Ratna Indria Setyowati, Mbak Icha, dan Mas Yogi yang merupakan sumber motivasi utama penulis dalam menjalani kehidupan. Terima kasih atas segala dorongan, nasihat, kehangatan, dan

doa yang selalu diberikan untuk penulis sehingga penulis dapat menyelesaikan studi di Universitas Brawijaya.

10. Sahabat-sahabatku di kemahasiswaan Jurusan Teknik Industri; Johny, Fanani, Mulfi, dan Sadek yang memberikan kebahagiaan dan keceriaan, serta setia menemani dan menyemangati penulis baik selama penyelesaian skripsi maupun selama 4 tahun di Jurusan Teknik Industri Universitas Brawijaya.
11. Adinda Sekar Arum Perwitasari, atas waktu yang diberikan untuk penulis selama ini. Terima kasih atas segala bantuan dan keceriaan yang diberikan, serta doa yang dipanjatkan untuk penulis dalam penyelesaian skripsi ini.
12. Pengurus HMTI FT-UB 2014/2015; Fabrito, Tita, Ela, Triya, dan Aini atas dukungan, keceriaan, dan semangatnya selama ini.
13. Asisten Industrial Computer Laboratory; Youngky, Helmi, Adi, Esa, Firda, Gagas, Fiqar, Della, Fitri, dan Lintang, atas motivasi serta keceriaan yang diberikan selama pengerjaan skripsi
14. Astari Ramadawika Prasindi yang telah memfasilitasi penulis dalam pelaksanaan penelitian skripsi ini.
15. Keluarga Besar Teknik Industri 2011 yang telah memberi banyak pelajaran dan inspirasi bagi penulis.
16. Pihak-pihak yang tidak dapat saya sebutkan satu-persatu pada kata pengantar ini.

Penulis menyadari bahwa masih banyak kekurangan dalam skripsi ini. Oleh karena itu, saran dan kritik yang membangun sangat diharapkan oleh penulis sebagai bentuk pembelajaran dalam penyusunan karya serupa ke depannya. Semoga skripsi ini dapat memberikan manfaat kepada pembaca dan semoga penulis dapat selalu melaksanakan tanggung jawab sebagai Sarjana Teknik dan memberikan sumbangsih lebih terhadap ilmu pengetahuan serta kesejahteraan masyarakat.

Malang, Agustus 2015

Penulis



DAFTAR ISI

| | |
|---|------|
| KATA PENGANTAR | i |
| DAFTAR ISI | iii |
| DAFTAR TABEL | vi |
| DAFTAR GAMBAR | viii |
| DAFTAR LAMPIRAN | x |
| RINGKASAN | xi |
| BAB I PENDAHULUAN | 1 |
| 1.1 Latar Belakang | 1 |
| 1.2 Identifikasi Masalah | 5 |
| 1.3 Rumusan Masalah | 5 |
| 1.4 Batasan Masalah..... | 6 |
| 1.6 Tujuan Penelitian..... | 6 |
| 1.7 Manfaat Penelitian..... | 6 |
| BAB II TINJAUAN PUSTAKA | 8 |
| 2.1 Penelitian Terdahulu..... | 8 |
| 2.2 Perawatan | 10 |
| 2.2.1 Pengertian dan Tujuan Perawatan | 10 |
| 2.2.2 Manajemen Perawatan | 11 |
| 2.2.3 Perawatan Preventif..... | 12 |
| 2.2.4 <i>Corrective Maintenance</i> | 12 |
| 2.2.5 <i>Computerized Maintenance Management System</i> | 13 |
| 2.2.5.1 Modul <i>Computerized Maintenance Management System</i> | 13 |
| 2.2.6 <i>Overall Equipment Effectiveness</i> | 15 |
| 2.3 Sistem Informasi..... | 16 |
| 2.3.1 Pengertian Sistem..... | 16 |
| 2.3.2 Pengertian Informasi | 17 |
| 2.3.2.1 Kualitas Informasi..... | 17 |
| 2.3.2 Pengertian Sistem Informasi | 17 |
| 2.4 Sistem Database | 18 |
| 2.4.1 Komponen Sistem Database..... | 18 |
| 2.4.2 Pengertian Data | 19 |

| | |
|---|----|
| 2.4.3 Hierarki Data | 19 |
| 2.4.4 Tujuan Sistem Database | 20 |
| 2.4.5 <i>Entity Relational Diagram</i> (ERD)..... | 21 |
| 2.4.6 Normalisasi..... | 23 |
| 2.5 <i>Group Technology</i> | 23 |
| 2.5.1 Metode Klasifikasi <i>Group Technology</i> | 24 |
| 2.6 <i>Software Prototyping</i> | 26 |
| 2.7 <i>Visual Basic for Application</i> (VBA)..... | 30 |
| 2.8 <i>Microsoft Access</i> | 30 |
| BAB III METODOLOGI PENELITIAN | 32 |
| 3.1 Metode Penelitian..... | 32 |
| 3.2 Tempat dan Waktu Penelitian | 32 |
| 3.3 Data yang Digunakan | 33 |
| 3.4 Metode Pengumpulan Data | 33 |
| 3.5 Langkah Penelitian | 32 |
| 3.6 Diagram Alir Penelitian..... | 35 |
| BAB IV PENGUMPULAN DATA DAN ANALISIS SISTEM | 37 |
| 4.1 Gambaran Umum Perusahaan | 37 |
| 4.1.1 Profil PT. Gunawan Dianjaya Steel..... | 37 |
| 4.1.2 Visi dan Misi Perusahaan | 38 |
| 4.1.3 Struktur Organisasi..... | 38 |
| 4.1.4 Sistem <i>Maintenance</i> PT. Gunawan Dianjaya Steel | 40 |
| 4.1.4 Kebutuhan Data..... | 42 |
| 4.2 Pengembangan <i>Prototype</i> | 43 |
| 4.3 Analisis Sistem | 46 |
| 4.3.1 <i>Requirement Modelling</i> | 47 |
| 4.3.2 <i>Data and Process Modelling</i> | 50 |
| 4.3.2.1 <i>Data Modelling</i> | 50 |
| 4.3.2.2 <i>Process Modelling</i> | 58 |
| 4.3.3 <i>Development Strategies</i> | 59 |
| BAB V PERANCANGAN SISTEM | 61 |
| 5.1 Desain Sistem..... | 61 |
| 5.1.1 Konsep <i>Group Technology</i> dalam Desain <i>Database</i> | 61 |
| 5.1.1.1 Struktur Sistem dan Parameter Kodefikasi | 65 |

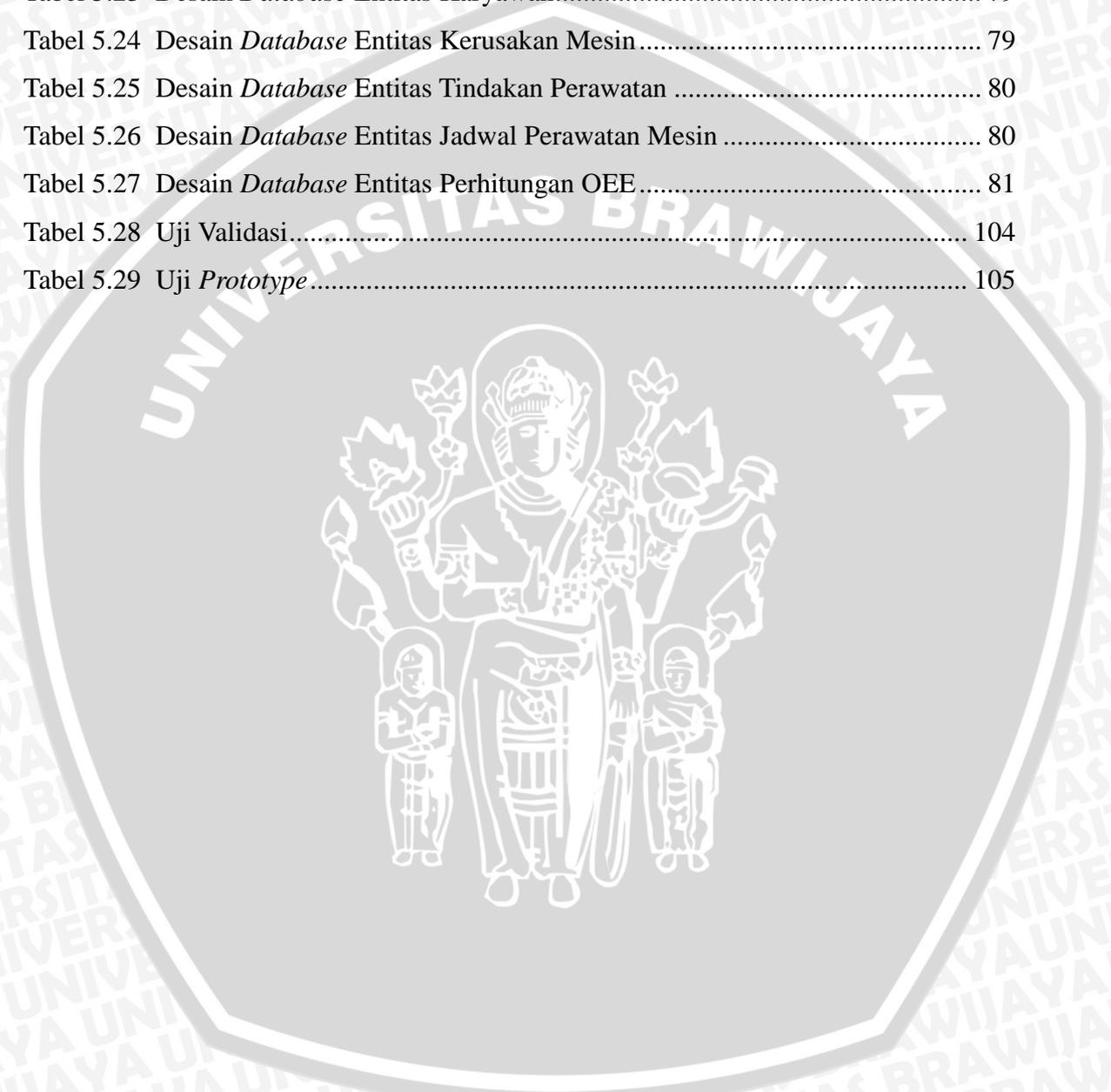


| | |
|--|-----|
| 5.1.1.2 Kegunaan <i>Group Technology</i> dalam Sistem Informasi | |
| Manajemen Perawatan Mesin | 66 |
| 5.1.2 Konsep <i>Computerized Maintenance Management System</i> dalam | |
| Perancangan Sistem Informasi | 67 |
| 5.1.3 Pengukuran <i>Overall Equipment Effectiveness</i> (OEE)..... | 68 |
| 5.1.3.1 <i>Availability Rate</i> | 68 |
| 5.1.3.2 <i>Performance Rate</i> | 69 |
| 5.1.3.3 <i>Quality Rate</i> | 70 |
| 5.1.3.4 <i>Overall Equipment Effectiveness</i> | 71 |
| 5.1.4 Desain <i>Database</i> | 72 |
| 5.1.4.1 Desain <i>Database</i> Logis | 73 |
| 5.1.4.2 Desain <i>Database</i> Fisik | 76 |
| 5.1.5 Desain <i>User Interface</i> | 81 |
| 5.1.5.1 Bagan Hierarki Menu <i>User Interface</i> | 81 |
| 5.1.5.2 Desain <i>Interface Form</i> | 84 |
| 5.1.5.3 Desain <i>Report</i> | 88 |
| 5.1.6 Desain <i>Algoritma</i> | 89 |
| 5.2 Implementasi | 91 |
| 5.2.1 Implementasi <i>Database</i> | 91 |
| 5.2.2 Implementasi <i>User Interface</i> | 94 |
| 5.2.3 Implementasi Modul Program..... | 98 |
| 5.3 Pengujian | 100 |
| 5.3.1 Verifikasi | 100 |
| 5.3.2 Validasi | 104 |
| 5.3.3 Uji Prototype | 105 |
| 5.4 Prototype Sistem Informasi Manajemen Perawatan Berbasis | |
| <i>Computerized Maintenance Management System</i> dan <i>Group Technology</i> | 106 |
| BAB VI PENUTUP | 108 |
| 6.1 Kesimpulan..... | 108 |
| 6.2 Saran..... | 109 |
| DAFTAR PUSTAKA | 110 |
| LAMPIRAN | 112 |

DAFTAR TABEL

| | | |
|------------|---|----|
| Tabel 1.1 | Tanggal Perencanaan dan Tanggal Aktual Tindakan Perawatan Mesin..... | 2 |
| Tabel 1.2 | Analisis PIECES Terhadap Sistem Lama di Departemen <i>Maintenance</i> PT. GDS | 3 |
| Tabel 2.1 | Perbandingan Penelitian Pendahuluan..... | 10 |
| Tabel 2.2 | Kardinalitas Minimum dan Maksimum Pada Relasi Dua Entitas dalam ERD..... | 23 |
| Tabel 2.3 | Kode Atribut <i>Inventory</i> | 25 |
| Tabel 4.1 | Sistem <i>Requirement Checklist</i> untuk Manajer dan SPV | 48 |
| Tabel 4.2 | Sistem <i>Requirement Checklist</i> untuk Teknisi..... | 49 |
| Tabel 4.3 | Identifikasi <i>Input</i> dan <i>Output</i> DFD | 51 |
| Tabel 4.4 | Logika Proses Bisnis dan Aturan Pada Departemen <i>Maintenance</i> | 58 |
| Tabel 4.5 | Kebutuhan Minimum <i>Software</i> | 59 |
| Tabel 4.6 | Kebutuhan Minimum <i>Hardware</i> | 60 |
| Tabel 5.1 | Kriteria Pengelompokan <i>Sparepart</i> | 64 |
| Tabel 5.2 | Parameter Kodefikasi..... | 66 |
| Tabel 5.3 | Kebutuhan Sistem dan Modul CMMS Terkait | 67 |
| Tabel 5.4 | Data dan Hasil Perhitungan <i>Availability Rate</i> Mesin <i>Hot Rolling Mill</i> | 69 |
| Tabel 5.5 | Data dan Hasil Perhitungan <i>Availability Rate</i> Mesin <i>Hot Leveler</i> | 69 |
| Tabel 5.6 | Data dan Hasil Perhitungan <i>Availability Rate</i> Mesin <i>Reheating Furnace</i> ... | 69 |
| Tabel 5.7 | Data dan Hasil Perhitungan <i>Performance Rate</i> Mesin <i>Hot Rolling Mill</i> | 70 |
| Tabel 5.8 | Data dan Hasil Perhitungan <i>Performance Rate</i> Mesin <i>Hot Leveler</i> | 70 |
| Tabel 5.9 | Data dan Hasil Perhitungan <i>Performance Rate</i> Mesin <i>Reheating Furnace</i> | 70 |
| Tabel 5.10 | Data dan Hasil Perhitungan <i>Quality Rate</i> Mesin <i>Hot Rolling Mill</i> | 71 |
| Tabel 5.11 | Data dan Hasil Perhitungan <i>Quality Rate</i> Mesin <i>Hot Leveler</i> | 71 |
| Tabel 5.12 | Data dan Hasil Perhitungan <i>Quality</i> Mesin <i>Reheating Furnace</i> | 71 |
| Tabel 5.13 | Hasil Perhitungan <i>Overall Equipment Effectiveness</i> Mesin <i>Hot Rolling</i> <i>Mill</i> | 72 |
| Tabel 5.14 | Hasil Perhitungan <i>Overall Equipment Effectiveness</i> Mesin <i>Hot Leveler</i> ... | 72 |
| Tabel 5.15 | Hasil Perhitungan <i>Overall Equipment Effectiveness</i> Mesin <i>Reheating</i> <i>Furnace</i> | 72 |
| Tabel 5.16 | Daftar Entitas dan Atribut ERD | 73 |
| Tabel 5.17 | Identifikasi Relasi | 74 |

| | |
|--|-----|
| Tabel 5.18 Desain <i>Database</i> Entitas Departemen | 77 |
| Tabel 5.19 Desain <i>Database</i> Entitas Jenis Mesin | 77 |
| Tabel 5.20 Desain <i>Database</i> Entitas Mesin | 77 |
| Tabel 5.21 Desain <i>Database</i> Entitas Komponen | 78 |
| Tabel 5.22 Desain <i>Database</i> Entitas <i>Sparepart</i> | 78 |
| Tabel 5.23 Desain <i>Database</i> Entitas Karyawan | 79 |
| Tabel 5.24 Desain <i>Database</i> Entitas Kerusakan Mesin | 79 |
| Tabel 5.25 Desain <i>Database</i> Entitas Tindakan Perawatan | 80 |
| Tabel 5.26 Desain <i>Database</i> Entitas Jadwal Perawatan Mesin | 80 |
| Tabel 5.27 Desain <i>Database</i> Entitas Perhitungan OEE | 81 |
| Tabel 5.28 Uji Validasi | 104 |
| Tabel 5.29 Uji <i>Prototype</i> | 105 |



DAFTAR GAMBAR

| | | |
|-------------|--|----|
| Gambar 2.1 | Hierarki Data | 19 |
| Gambar 2.2 | Lambang Entitas..... | 21 |
| Gambar 2.3 | Lambang <i>Relationship</i> | 22 |
| Gambar 2.4 | Hierarki Klasifikasi <i>Sparepart</i> | 25 |
| Gambar 2.5 | Sistem Kode <i>Hybrid</i> | 25 |
| Gambar 2.6 | Contoh Sistem Kode <i>Opitz</i> | 26 |
| Gambar 2.7 | Model Proses Pengembangan <i>Prototype</i> | 27 |
| Gambar 3.1 | Diagram Alir Penelitian..... | 36 |
| Gambar 4.1 | Kantor Perusahaan GDS | 37 |
| Gambar 4.2 | Struktur Organisasi GDS..... | 39 |
| Gambar 4.3 | Struktur Organisasi Departemen <i>Maintenance</i> GDS | 39 |
| Gambar 4.4 | Bagan Alir Analitis Sistem Penugasan <i>Planned Maintenance</i> | 41 |
| Gambar 4.5 | Bagan Alir Analitis Sistem Penugasan <i>Unplanned Maintenance</i> | 42 |
| Gambar 4.6 | Diagram Alir Perancangan <i>Prototype</i> Sistem Informasi | 45 |
| Gambar 4.7 | <i>Context Diagram</i> Sistem Informasi Manajemen Perawatan Mesin | 52 |
| Gambar 4.8 | <i>Hierarchy Chart</i> DFD | 53 |
| Gambar 4.9 | DFD Level 0 Sistem Informasi Manajemen Perawatan Mesin..... | 54 |
| Gambar 4.10 | DFD Level 1 Proses 1 Sistem Pengolahan Data Awal | 55 |
| Gambar 4.11 | DFD Level 1 Proses 1 Sistem Pengukuran OEE | 56 |
| Gambar 4.12 | DFD Level 1 Proses 1 Sistem Penugasan dan Penjadwalan Perawatan . | 57 |
| Gambar 5.1 | Konsep <i>Group Technology</i> dengan Sistem Kodefikasi Hirarki..... | 62 |
| Gambar 5.2 | Konsep <i>Group Technology</i> dengan Sistem Kodefikasi <i>Hybrid</i> | 62 |
| Gambar 5.3 | Contoh Kodefikasi Mesin, Komponen, dan <i>Spareparts</i> | 66 |
| Gambar 5.4 | Diagram E-R dengan Atribut | 75 |
| Gambar 5.5 | Desain Hierarki Menu Sistem Informasi Manajemen Perawatan Berbasis CMMS dan <i>Group Technology</i> | 82 |
| Gambar 5.6 | Desain Hierarki Menu <i>Work Order</i> | 82 |
| Gambar 5.7 | Desain Hierarki Menu Manajemen Mesin | 83 |
| Gambar 5.8 | Desain Hierarki Menu <i>Master Data</i> | 84 |
| Gambar 5.9 | Desain <i>Form Main Menu</i> | 85 |
| Gambar 5.10 | Desain <i>Form Work Order</i> | 86 |
| Gambar 5.11 | Desain <i>Form</i> Manajemen Mesin | 87 |

| | | |
|-------------|--|-----|
| Gambar 5.12 | Desain <i>Form Master Data</i> | 88 |
| Gambar 5.13 | <i>Pseudocode</i> Proses Menambah, Menyimpan, Memperbarui, dan Menghapus Data | 90 |
| Gambar 5.14 | <i>Pseudocode</i> Proses Perhitungan OEE | 91 |
| Gambar 5.15 | Tampilan <i>Table Design</i> Pada <i>Microsoft Access 2013</i> | 92 |
| Gambar 5.16 | <i>Printscreen</i> Tabel <i>Asset</i> Pada <i>Microsoft Access 2013</i> | 92 |
| Gambar 5.17 | <i>Printscreen</i> Tabel <i>Sparepart</i> Pada <i>Microsoft Access 2013</i> | 93 |
| Gambar 5.18 | <i>Printscreen</i> Tabel <i>Work Order</i> Pada <i>Microsoft Access 2013</i> | 93 |
| Gambar 5.19 | <i>Printscreen</i> Relasi Antar Tabel Entitas..... | 93 |
| Gambar 5.20 | <i>Printscreen</i> Menu Login | 94 |
| Gambar 5.21 | <i>Printscreen</i> Main Menu | 95 |
| Gambar 5.22 | <i>Printscreen</i> Form <i>Maintenance</i> | 96 |
| Gambar 5.23 | <i>Printscreen</i> Form Manajemen Aset | 96 |
| Gambar 5.24 | <i>Printscreen</i> Form <i>Master List</i> | 97 |
| Gambar 5.25 | <i>Print Preview Report</i> dari <i>Sparepart</i> | 97 |
| Gambar 5.26 | <i>Print Preview Report</i> dari <i>Work Order</i> | 98 |
| Gambar 5.27 | <i>Print Preview Report</i> Perhitungan OEE..... | 98 |
| Gambar 5.28 | <i>Syntax</i> Proses Memasukkan dan Menyimpan Data Mesin | 99 |
| Gambar 5.29 | <i>Syntax</i> Proses Perhitungan OEE..... | 100 |
| Gambar 5.30 | Memilih Periode <i>Report Summary Work Order</i> | 101 |
| Gambar 5.31 | <i>Print Preview</i> Hasil <i>Report Summary Work Order</i> | 102 |
| Gambar 5.32 | Jumlah <i>Sparepart Serfo Amplifier</i> | 102 |
| Gambar 5.33 | Perhitungan OEE..... | 104 |



DAFTAR LAMPIRAN

| | | |
|------------|--------------------------|-----|
| Lampiran 1 | <i>Manual Book</i> | 112 |
|------------|--------------------------|-----|



RINGKASAN

REZA MAULANA, Jurusan Teknik Industri, Fakultas Teknik Universitas Brawijaya, 2015, *Perancangan Sistem Informasi Manajemen Perawatan Berbasis Computerized Maintenance Management System dan Group Technology (Studi Kasus: PT. Gunawan Dianjaya Steel, Surabaya)*, Dosen Pembimbing: Purnomo Budi Santoso dan Zefry Darmawan.

PT. Gunawan Dianjaya Steel (GDS) merupakan salah satu perusahaan yang bergerak dalam bidang manufaktur yang memproduksi plat baja. Meningkatnya rutinitas produksi setiap mesin seiring dengan usaha untuk memenuhi kebutuhan konsumen, mengharuskan departemen *maintenance* GDS melakukan perawatan mesin untuk menjaga agar performansi mesinnya tetap stabil. Selama ini, sistem manajemen perawatan mesin yang ada masih menggunakan sistem pencatatan secara manual. Pada kondisi tersebut, memungkinkan adanya data yang hilang sehingga tindakan perawatan mesin yang dilakukan sering tidak sesuai dengan rencana yang dijadwalkan. Permasalahan yang terjadi lainnya adalah pencarian *sparepart* ketika dibutuhkan memerlukan waktu lama karena belum ada sistem pengelompokan *sparepart* yang dapat memudahkan pencarian informasi *sparepart* dari suatu komponen mesin. Selain itu, belum ada pengukuran peformansi mesin yang menyebabkan tidak sesuainya tindakan perawatan yang dilakukan terhadap kondisi mesin berakibat pada kecepatan mesin yang belum optimal dan sering terjadi *downtime*. Oleh karena itu, diperlukan perancangan aplikasi perangkat lunak berbasis *Computerized Maintenance Management System* (CMMS) serta *database* berbasis kodefikasi mesin dengan *group technology* sebagai sarana pendukung manajemen perawatan mesin pada departemen *maintenance*.

Dalam penelitian ini, metode yang digunakan dalam perancangan sistem informasi ini adalah *software prototyping*. Langkah awal dalam perancangan *prototype* ini adalah menetapkan tujuan *prototype* dengan mengidentifikasi masalah dalam sistem yang sedang berjalan menggunakan analisis PIECES. Langkah selanjutnya adalah mendefinisikan fungsi *prototype* dengan membuat model kebutuhan sistem, model aliran data, model logika proses bisnis, serta pengembangan strategi. Kemudian, langkah selanjutnya adalah desain yang meliputi *database* logis, *database* fisik, *user interface*, dan algoritma. Pada tahap ini juga dirancang sistem kodefikasi *group technology* untuk mengelompokkan mesin, komponen dan *sparepart* serta merancang konsep perhitungan performansi mesin menggunakan *Overall Equipment Effectiveness*. Setelah itu, desain yang telah dibuat diimplementasikan pada *Microsoft Access 2013*. Langkah terakhir yang dilakukan adalah evaluasi *prototype* dengan uji verifikasi, uji validasi, dan uji *prototype*.

Hasil dari penelitian ini adalah *prototype* sistem informasi manajemen perawatan dengan *database* berbasis kodefikasi menggunakan *group technology* yang memiliki beberapa fitur dari modul CMMS. Modul CMMS yang digunakan pada perancangan ini yaitu *work order*, *asset management*, *employee*, dan *supplier/ vendor*. Sistem informasi ini juga dapat memberikan *report* data secara keseluruhan dan mempermudah pencarian data yang dibutuhkan. Selain itu, *prototype* ini juga dapat melakukan pengukuran peformansi mesin dengan menggunakan *Overall Equipment Effectiveness*.

Kata Kunci: Sistem Informasi Manajemen Perawatan, *Computerized Maintenance Management System*, *Group Technology*, *Prototyping*, Sistem *Database*, Perancangan Sistem, *Overall Equipment Effectiveness*.

SUMMARY

REZA MAULANA, Department of Industrial Engineering, Faculty of Engineering, University of Brawijaya, 2015, *Maintenance Management Information System Design Based on Computerized Maintenance Management System and Group Technology (A Case Study in PT. Gunawan Dianjaya Steel Tbk.)*, Academic Supervisors: Purnomo Budi Santoso and Zefry Darmawan.

PT. Gunawan Dianjaya Steel (GDS) is a company which engages in manufacturing steel plate products. In order to meet consumer requirements, GDS should increase its production rate, which affects to its machine performance. Therefore, GDS requires maintenance activities to make the machine performance remain stable. During machine maintenance, recording activities are often not carried out in accordance with the scheduled plan because recording is still done manually on paper without utilizing software to simplify this process. Another problem which occurs is the more time needed during search of spare because there is no grouping system of spare parts to facilitate the information search about spare parts of a machine component. In addition, there has been no performance measurement which causes the incompatibility of maintenance actions performed and machine conditions resulting in the machine speed being not optimal and downtime often happening. Therefore, it is necessary to design application based software by using the concept of Computerized Maintenance Management System and codification-based database machine with group technology as a maintenance management supporting tools of maintenance department.

The method used to design information systems in this study was software prototyping. The initial step was setting a goal by identifying problems in a running system using PIECES analysis. The next step was defining the function of prototype by creating system requirements model, data flow model, logic of business process model, and strategy development. Then, designing phase was done which includes logical database, physical database, user interface, and algorithms design. At this stage, group technology codification system was designed for grouping machines, components, and spare parts as well as machine performance calculation concept by using Overall Equipment Effectiveness. Once the design had been made, Microsoft Access 2013 was used to implement this design. The final step was conducting evaluations which were done by verification test, validation test, and prototype test.

Results from this study was the prototype of maintenance management information system whose database was codification-based using group technology which had several features of CMMS module. CMMS module used in this study were work order, asset management, employee, and supplier/ vendor. This information system could provide data report as a whole and facilitate necessary data search. In addition, this prototype could also perform performance measurements on machine using Overall Equipment Effectiveness.

Keywords: Maintenance Management Information System, Computerized Maintenance Management System, Group Technology, Prototyping, Database System, System Design, Overall Equipment Effectiveness.

BAB I PENDAHULUAN

Pada bab pendahuluan ini akan dijelaskan mengenai latar belakang dilakukannya penelitian, pengidentifikasian masalah yang ada, rumusan masalah, batasan masalah, asumsi, tujuan serta manfaat penelitian.

1.1 LATAR BELAKANG

Kebutuhan baja di Indonesia pada 3 tahun terakhir mengalami peningkatan hingga 6 juta ton baja/tahun. Sedangkan berdasarkan data Kemenperin tahun 2014, perusahaan Indonesia hanya mampu memproduksi 4 juta ton per tahun untuk memenuhi kebutuhan tersebut. Hal ini mengakibatkan para konsumen baja di Indonesia akhirnya mendatangkan baja dari luar negeri untuk memenuhi kebutuhan mereka.

Melihat kenyataan tersebut, maka perusahaan baja dalam negeri harus meningkatkan kapasitas produksinya agar dapat memenuhi permintaan pasar domestik yang ada. Industri baja di Indonesia harus lebih meningkatkan lagi efektivitas dan efisiensi kerjanya. Peningkatan efektivitas dan efisiensi tersebut tidak lepas dari peranan mesin produksi yang merupakan faktor produksi untuk mengkonversi bahan baku menjadi bahan setengah jadi atau barang jadi pada perusahaan. Pentingnya peranan mesin produksi mengharuskan suatu perusahaan menjaga performansi mesin produksi yang dimilikinya agar selalu optimal dengan cara perawatan mesin (Fachrurrozi, 2002:4).

Perawatan adalah suatu kombinasi dari setiap tindakan yang dilakukan untuk menjaga suatu barang dalam, atau untuk memperbaiki sampai suatu kondisi yang bisa diterima (Corder, 1988:4). Perawatan mesin merupakan suatu permasalahan yang masih tergolong rumit karena di dalamnya terdapat beberapa faktor yang saling berkaitan, diantaranya seperti tenaga kerja/karyawan, mesin-mesin, penjadwalan, *sparepart*, serta jenis perawatan. Berkaitan dengan peningkatan kapasitas produksi, maka rutinitas produksi setiap mesin juga akan bertambah. Hal ini dapat menyebabkan turunnya performansi mesin dan mesin mudah rusak apabila tidak dilakukan perawatan mesin secara rutin.

PT. Gunawan Dianjaya Steel, Tbk adalah perusahaan yang bergerak di bidang manufaktur pembuatan pelat baja. *Raw material* yang digunakan untuk pembuatan pelat baja yaitu slab. Ada 4 jenis slab yang digunakan untuk membuat pelat baja dengan spesifikasi yang berbeda-beda, yaitu *Mild Steel*, *High Steel*, *Boiler A-516*, dan

Shipbuilding Steel. Pelat baja yang diproduksi PT. GDS memiliki spesifikasi yang dapat bermacam-macam tergantung dari permintaan pelanggan.

Di PT. GDS terdapat 3 jenis pengelompokan perawatan mesin, yaitu mekanik, mesin elektrik, dan *water treatment*. Mekanik terdiri dari perawatan mesin-mesin yang memiliki peran utama dalam produksi plat baja seperti mesin *Hot Rolling Mill*, *Reheating Furnace*, dan *Hot Leveler*. Elektrik merupakan perawatan mesin-mesin yang menunjang kegiatan produksi seperti mesin *crane*, mesin *cutting*, dan mesin las. Sedangkan *water treatment* adalah perawatan mesin yang menunjang produksi dalam hal pelumasan dan pendinginan. Departemen *maintenance* PT. GDS melakukan perawatan pada 3 pengelompokan perawatan mesin tersebut.

Selama ini, sistem manajemen perawatan preventif mesin mekanik PT. GDS sering tidak sesuai dengan rencana yang dijadwalkan. Berikut data perbedaan tanggal perencanaan dan tanggal aktual tindakan perawatan mesin yang dilakukan periode Oktober - November 2014 pada Tabel 1.1.

Tabel 1.1 Tanggal Perencanaan dan Tanggal Aktual Tindakan Perawatan Mesin

| Tanggal Perencanaan | Tanggal Aktual | Nama Tindakan | Nama Mesin |
|---------------------|------------------|--|--------------------------|
| 19 Oktober 2014 | 27 Oktober 2014 | Bongkar / turunkan <i>back up roll mill</i> atas dan bawah | <i>Back Up Roll Mill</i> |
| 6 November 2014 | 14 November 2014 | Pengecekan baut – baut <i>Screw Down Mill</i> Barat & Timur | <i>Screw Down Mil</i> |
| 17 November 2014 | 19 November 2014 | Perbaiki <i>bock back up roll hot leveller</i> atas yang tidak bisa putar. | <i>Hot Leveller</i> |
| 17 November 2014 | 21 November 2014 | Las <i>saport box work roll hot leveller</i> sebelah barat putus | <i>Hot Leveller</i> |

Sumber: PT. Gunawan Dianjaya Steel, Tbk

Berdasarkan data tindakan perawatan yang diambil pada bulan September – Oktober 2014, dapat dilihat bahwa selama ini tindakan perawatan mesin yang direncanakan oleh departemen *maintenance* PT. GDS mengalami keterlambatan dalam pelaksanaannya. Keterlambatan tersebut terjadi karena departemen *maintenance* PT. GDS tidak mempunyai sistem penyajian informasi yang baik dan sistematis. Data mesin, data *sparepart*, *history* kerusakan mesin, dan jadwal perawatan mesin yang saling terkait tidak terkomputerisasi dengan baik. Selain itu, pengingat untuk tindakan perawatan yang belum dilakukan juga masih belum tersedia. Keterlambatan perawatan mesin mengakibatkan *shutdown* secara tiba-tiba pada saat produksi sehingga proses produksi berhenti. Hal itu merugikan PT. GDS di karenakan target produksi akan tidak terpenuhi.

Ketersediaan *sparepart* mesin juga sering menjadi masalah departemen *maintenance* PT. GDS yaitu sering tidak tersedia ketika dibutuhkan dan pencarian data

sparepart membutuhkan waktu yang lama. Dalam hal ini, sistem manajemen *inventory* *sparepart* yang ada di PT. GDS masih belum terkomputerasi dengan baik. Sistem pencatatan masih dilakukan secara manual di atas kertas baik masuk maupun keluar sehingga informasi stok *sparepart* tidak akurat dan berakibat pada tidak adanya ketersediaan. Permasalahan lainnya adalah belum adanya kodefikasi pada mesin, komponen dan *sparepart* yang terintegrasi menyebabkan perlunya waktu yang lama dalam pencarian data *sparepart* yang akan digunakan.

Agar dapat dilihat secara ringkas dan jelas, maka digunakan analisa PIECES (*Performance-Information-Economy-Control-Efficiency-Service*) untuk mengetahui kelemahan-kelemahan sistem yang ada di PT. GDS. Berikut analisa PIECES pada sistem di departemen *maintenance* PT. GDS yang dapat dilihat pada Tabel 1.2.

Tabel 1.2 Analisa PIECES Terhadap Sistem Lama di Departemen *Maintenance* PT. GDS

| Jenis Analisis | Kelemahan Sistem Lama |
|--------------------|--|
| <i>Performance</i> | Pengumpulan informasi data untuk tindakan perawatan mesin memerlukan waktu lama. |
| | Sulitnya menelusuri informasi yang dibutuhkan oleh SPV tentang data semua mesin, jadwal perawatan mesin dan kerusakan mesin serta tindakan harian. |
| | Belum adanya pengukuran peformansi dan efektivitas mesin. |
| <i>Information</i> | Sistem informasi tidak mudah diakses oleh SPV dan karyawan <i>maintenance</i> karena pencarian data mesin, karyawan, <i>sparepart</i> maupun jadwal perawatan mesin masih manual dan lisan serta tidak terorganisir. |
| | Tidak ada informasi tindakan harian yang harus dikerjakan teknis. |
| | Kemungkinan kehilangan data serta kesalahan pencatatan akibat kelalalian atau <i>human error</i> tinggi. |
| <i>Economy</i> | Biaya operasional serta waktu yang dibutuhkan besar karena terdapat biaya administrasi untuk pembuatan dokumen dengan kertas dan penyimpanan secara manual. |
| <i>Control</i> | Keamanan data tidak terjamin dikarenakan data disimpan secara manual sehingga rentan akan hilang. |
| | Belum adanya sistem yang mengatur data sehingga kemungkinan besar dapat terjadinya kesalahan informasi tentang pengambilan keputusan tindakan perawatan mesin. |
| <i>Efficiency</i> | Belum memanfaatkan teknologi yang ada, data masih dicatat secara manual dan penyimpanan data secara manual membutuhkan tempat penyimpanan. |
| | Waktu pencarian <i>sparepart</i> lama yang berakibat memperlambat kinerja proses. |
| <i>Services</i> | Pelayanan akan kebutuhan informasi dari beberapa data berlangsung lama karena data belum terintegrasi menjadi satu sehingga sistem tidak fleksibel. |

Berdasarkan Tabel 1.2 diketahui bahwa sistem lama pada departemen *maintenance* PT. GDS masih bersifat manual dimana kompleksitas data belum dapat diolah secara baik. *Accessibility* data pada sistem lama masih sulit dilakukan begitu pula dengan permintaan laporan untuk pihak manajerial secara cepat, tepat, dan akurat karena data

belum terorganisasi dengan baik. Selain itu, belum ada klasifikasi pengelompokan *sparepart* pada departemen *maintenance* menyebabkan pencarian data *sparepart* membutuhkan waktu yang lama.

Tingkat performansi serta *availability* mesin juga sering menjadi masalah berupa kecepatan mesin *Roll Table* yang belum optimal. Kecepatan mesin tersebut sering kali berubah dan tidak sesuai dengan *setting* yang sudah dilakukan oleh operator. Bahkan mesin sering mengalami *downtime* akibat dari kerusakan yang terjadi pada komponen mesin. Hal ini terjadi karena tidak dilakukannya perawatan preventif yang sesuai dengan kondisi mesin tersebut. Oleh karena itu, diperlukan pengukuran performansi mesin untuk mengukur kinerja peralatan, utilitas, serta kemampuan untuk menghasilkan produk yang berkualitas agar kinerja peralatan, utilitas, serta kemampuan untuk menghasilkan produk yang berkualitas untuk setiap mesin yang ada di perusahaan tetap termonitoring dengan baik dan dapat ditingkatkan efektivitasnya supaya produksi tidak terganggu.

Kondisi yang terjadi pada departemen *maintenance* PT. GDS ini kemudian mendasari perlunya perancangan sistem informasi manajemen perawatan mesin untuk departemen *maintenance*. Sistem informasi manajemen perawatan mesin ini akan dibangun dengan didukung sistem *database* yang baik. Dengan adanya sistem informasi manajemen perawatan mesin, maka aliran data manajemen perawatan mesin dapat diakses kapanpun, lebih cepat, murah, serta mampu menurunkan biaya atas kebutuhan penyimpanan informasi. Pemanfaatan sistem informasi dalam bidang *maintenance* yang saat ini mengalami perkembangan sangat cepat di perusahaan – perusahaan besar tingkat dunia dikenal dengan nama *Computerized Maintenance Management Systems* (CMMS). CMMS adalah aplikasi perangkat lunak yang mengatur *database* berisi informasi aktivitas pemeliharaan dalam suatu perusahaan (Terry Wireman, 1994:23).

Untuk melakukan pengukuran terhadap kinerja peralatan, utilitas, dan kemampuan menghasilkan produk yang berkualitas, maka perlu dilakukan pengukuran terhadap nilai *Overall Equipment Effectiveness* (OEE) dari mesin tersebut. Nilai OEE nantinya menjadi pedoman untuk perusahaan dalam melakukan perbaikan maupun *improvement*. Pengukuran nilai OEE sendiri pada dasarnya melibatkan tiga faktor penting, yaitu *availability rate*, *performance rate*, dan *quality rate*.

Salah satu prinsip pengelompokan yang bisa dikembangkan dalam mendukung perancangan *database* sistem manajemen informasi ini adalah *Group Technology* (GT). Dengan GT, mesin dan *sparepart* yang ada akan diklasifikasikan dan dikodekan menurut kriteria dan parameter yang ditentukan. Dari klasifikasi kode yang telah dibuat inilah, GT

akan berperan sebagai pendukung pembuatan *database* sistem informasi manajemen perawatan mesin sehingga nantinya informasi mengenai data *sparepart* dapat langsung ditelusuri berdasarkan kode yang telah dirancang.

Dengan mempertimbangkan semua faktor di atas, maka dirasa perlu dilakukan penelitian mengenai perancangan sistem informasi manajemen perawatan berbasis *Computerized Maintenance Management System (CMMS)* pada pengembangan sistem informasi manajemen perawatan mesin pada departemen *maintenance* PT. GDS agar dapat melakukan pengolahan data serta manajemen informasi berbentuk *database* yang terintegrasi dengan *Group Technology* dalam rangka meningkatkan efektifitas dan efisiensi dalam manajemen perawatan mesin.

1.2 IDENTIFIKASI MASALAH

Berdasarkan latar belakang yang telah disampaikan, permasalahan yang terdapat di departemen *maintenance* PT. GDS antara lain:

1. Penerapan sistem informasi manajemen perawatan seperti data *history* mesin, komponen mesin, data karyawan serta *sparepart* yang masih secara manual menyebabkan *accessibility* data dan penyajian data menjadi tidak efisien.
2. Sering terjadi keterlambatan jadwal perawatan mesin dan tidak akuratnya informasi ketersediaan jumlah *sparepart* yang kurang dari batas minimal.
3. Belum adanya pengukuran peformansi mesin secara *real-time* sehingga perusahaan kesulitan dalam menentukan kapan harus dilakukan *improvement* untuk memaksimalkan efektivitas mesin.
4. Proses pencarian informasi mengenai *sparepart* oleh teknisi membutuhkan waktu yang lama sehingga memperlambat kinerja proses manajemen perawatan.

1.3 RUMUSAN MASALAH

Berdasarkan latar belakang dan identifikasi masalah yang telah dipaparkan, maka masalah yang terjadi dapat dirumuskan sebagai berikut:

1. Bagaimana desain *database* serta melakukan pengelompokan *sparepart* berbasis *Computerized Maintenance Management System* dan *Group Technology* sebagai sarana pendukung manajemen perawatan pada departemen *maintenance*?
2. Bagaimana rancangan *prototype* sistem informasi perawatan mesin berbasis *Computerized Maintenance Management System* sebagai penunjang kegiatan manajemen perawatan mesin serta untuk pengukuran peformansi mesin?

3. Bagaimana hasil dari uji coba terhadap *prototype* sistem manajemen perawatan mesin yang dirancang dengan uji verifikasi, validasi dan uji *prototype*?

1.4 BATASAN MASALAH

Agar pembahasan pada penelitian supaya lebih terarah, maka batasan-batasan yang terdapat pada penelitian sebagai berikut:

1. Penelitian dilaksanakan di divisi mekanik dalam departemen *maintenance*.
2. Perancangan sistem informasi manajemen perawatan dibatasi hingga level *prototype* dengan menggunakan *software database*.
3. Hanya membahas sistem informasi mengenai kegiatan di departemen *maintenance*.
4. Modul *Computerized Maintenance Management System* dibatasi menggunakan modul *Work Order, Asset Management, Inventory Control*, serta *Labor/Employee*.

1.5 TUJUAN PENELITIAN

Adapun tujuan dari perancangan sistem informasi manajemen perawatan mesin yang dibuat antara lain:

1. Mendesain *database* dan mengelompokkan *sparepart* berbasis *Computerized Maintenance Management System* dan *Group Technology* yang dapat dimanfaatkan sebagai sarana pendukung manajemen perawatan mesin pada departemen *maintenance*.
2. Merancang *prototype* sistem informasi perawatan mesin menggunakan *software database* yang di dalamnya mencakup data mengenai beberapa faktor penunjang kegiatan manajemen perawatan mesin.
3. Melakukan uji coba terhadap *prototype* sistem manajemen perawatan mesin yang dirancang dengan uji verifikasi, validasi dan uji *prototype*.

1.6 MANFAAT PENELITIAN

Adapun manfaat yang diharapkan dalam penelitian ini ada sebagai berikut:

1. Bagi akademika Jurusan Teknik Industri Universitas Brawijaya
Memberikan karya tulis berupa pengalaman dalam penerapan dan pengembangan ilmu pengetahuan yang diperoleh di perkuliahan mengenai Sistem Informasi Manajemen, *Computerized Maintenance Management System*, Manajemen Perawatan, serta *Group Technology* pada Jurusan Teknik Industri.

2. Bagi Perusahaan

Memberikan suatu aplikasi *prototype* berupa sistem informasi manajemen perawatan mesin pada departemen *maintenance* agar dapat melakukan pengolahan data serta manajemen informasi berbentuk *database* yang terintegrasi dengan *Group Technology* dalam rangka meningkatkan efektifitas dan efisiensi dalam manajemen perawatan mesin.



BAB II TINJAUAN PUSTAKA

Pada bagian tinjauan pustaka akan diuraikan berbagai teori atau referensi yang terkait dan menunjang permasalahan yang akan diteliti. Referensi tersebut berkaitan dengan sistem informasi, konsep perancangan sistem, konsep pengembangan *prototype* dan referensi lainnya. Bab ini bertujuan untuk mendukung permasalahan yang akan diteliti serta mendukung hasil penelitian.

2.1 PENELITIAN TERDAHULU

Ada beberapa penelitian sebelumnya yang berkaitan dengan perancangan sistem informasi manajemen perawatan mesin. Berikut ini merupakan penjelasan secara singkat beberapa penelitian tersebut:

1. Endy (2011) melakukan penelitian di PT. Hadi Baru Medan. Permasalahan yang dihadapi yaitu informasi jadwal kegiatan perawatan mesin masih belum maksimal, dimana semua sistem masih dilakukan secara manual. Sistem informasi perawatan dikembangkan dengan pembuatan *database* untuk keperluan pengelolaan prosedur pembuatan jadwal dalam memperbaharui *reliability* dan penyusunan jadwal perawatan mesin. Sistem informasi ini dirancang dengan metode SDLC (*System Development Life Cycle*). Hasil rancangan sistem informasi perawatan berupa bentuk *software* yang berfungsi menyimpan rekaman kegiatan perawatan dan pembuatan jadwal perawatan berdasarkan hasil analisis *reliability*. Hasil dari pengujian diperoleh bahwa aplikasi system informasi mampu menghasilkan jadwal perawatan dan informasi yang diperlukan dalam pelaksanaan kegiatan perawatan.
2. Yudistira (2012) melakukan penelitian di PT. Sekar Laut Tbk. di Sidoarjo. Permasalahan yang terjadi diakibatkan besarnya kapasitas produksi sehingga kebutuhan pemeliharaan mesin-mesin produksi juga semakin tinggi. Perusahaan tidak melakukan dokumentasi data-data mengenai mesin, karyawan serta *spareparts*. Selain itu, belum ada tindakan pemeliharaan pencegahan karena selama ini perawatan hanya dilakukan ketika mesin mengalami kerusakan. Metode yang digunakan untuk mengembangkan sistem informasi *maintenance* menggunakan metode SDLC dan *software* yang digunakan untuk membuat *prototype* adalah VB.Net. Hasil dari penelitian ini menunjukkan bahwa sistem informasi yang dirancang telah mampu melakukan proses penyimpanan beberapa data dan dapat

menentukan penjadwalan *maintenance* mesin secara keseluruhan hingga 30 hari ke depan. Proses yang terkomputerisasi ini sangat bermanfaat untuk mengurangi hilangnya data mengenai pemeliharaan mesin sehingga dapat membantu perusahaan dalam melakukan pemeliharaan mesin.

3. Yudistiro (2012) menemukan masalah dalam hal perencanaan *line* produksi, perencana proses *assembly* di perusahaan dimana pada awalnya membutuhkan banyak waktu untuk merencanakan proses produksi suatu komponen. Metode yang digunakan adalah *group technology* untuk pengelompokan *assembly part* yang diaplikasikan dalam sistem informasi. Pengelompokan ini kemudian dikodekan, sehingga setiap *part* memiliki kode yang mewakili *family part*. Tujuan dari kodefikasi adalah untuk memudahkan *assembly planner* serta *bagian gudang* dalam merencanakan *assembly line*, serta mengidentifikasi lokasi simpan *part* dan *family part*. Hasil dari penelitian ini adalah sebuah *prototype* program *database* sistem informasi yang menggunakan *software Microsoft Acces 2007*.
4. Melladya (2013) melakukan penelitian di departemen *maintenance* PT. APW, dimana peneliti merancang suatu sistem informasi perawatan berbasis *group technology*. Sistem informasi yang di rancang hanya mencakup mengenai data *sparepart* yang dibantu oleh *Group Technology* dalam merancang *database*.

Penelitian ini akan mengembangkan sistem informasi manajemen perawatan mesin dengan memanfaatkan kodefikasi berbasis *group technology* dalam membangun *database*. Penelitian ini dilakukan di departemen *maintenance* PT. GDS, dimana terdapat permasalahan mengenai sistem informasi manajemen yang belum efektif dan efisien yang menyebabkan banyak terjadi ketidaksesuaian jadwal perawatan mesin yang dijadwalkan dengan pelaksanaannya. Serta di perusahaan tersebut belum melakukan pengukuran peformansi mesin secara *real-time* sehingga kesulitan kapan perlu dilakukan *improvement* terhadap mesin. Selain itu, masalah lainnya yang ditemukan adalah belum adanya sistem pengelompokan *sparepart* untuk mempermudah dan mempercepat pencarian informasi.

Sistem informasi manajemen perawatan mesin yang dirancang menggunakan metode pengembangan *prototype* dengan menggunakan aplikasi konsep *group technology* pada pengembangan *database*. Perancangan sistem ini menghasilkan sistem informasi yang dapat memberikan peringatan terhadap jadwal perawatan mesin dan peringatan untuk jumlah *sparepart* yang kurang dari batas minimal. Selain itu, sistem informasi manajemen perawatan mesin ini dapat menyajikan laporan yang dibutuhkan oleh pihak

SPV yang berfungsi untuk menentukan keputusan dan kebijakan dalam tindakan perawatan selanjutnya. Tabel 2.1 merupakan tabel perbandingan penelitian yang relevan dengan penelitian yang akan dilakukan.

Tabel 2.1 Perbandingan Penelitian Pendahuluan

| Peneliti | Metode | Software | Hasil Penelitian |
|-------------------------|--|-------------------------------|--|
| Endy (2011) | <i>Reliability Centered Maintenance</i> | <i>VB.Net</i> | <i>Software</i> yang menyimpan rekaman kegiatan dan jadwal <i>maintenance</i> . |
| Yudistira (2012) | <i>Preventive Maintenance</i> | <i>VB.Net with SQL Server</i> | Membuat data mesin, karyawan, <i>sparepart</i> , jenis <i>maintenance</i> , kerusakan mesin dan menentukan jadwal 30 hari ke depan. |
| Yudistiro (2012) | <i>Group Technology</i> | <i>Ms. Access</i> | Sebuah <i>prototype</i> program <i>database</i> sistem informasi dengan pengelompokkan <i>part</i> sehingga setiap <i>part</i> memiliki kode yang mewakili <i>family part</i> . |
| Melladya (2014) | <i>Group Technology</i> | <i>Ms. Access</i> | Mellady (2013), penelitian ini dilakukan di departemen <i>maintenance</i> PT. APW, dimana peneliti merancang suatu sistem informasi perawatan berbasis <i>group technology</i> . Sistem informasi yang di rancang hanya mencakup mengenai data <i>sparepart</i> yang dibantu oleh <i>Group Technology</i> dalam merancang <i>database</i> . |
| Maulana (2015) | <i>Group Technology, Overall Equipment Effectiveness</i> | <i>Ms. Access</i> | <i>Prototype</i> sistem informasi manajemen perawatan mesin lebih baik daripada sistem lama karena dapat memberikan laporan tentang jadwal perawatan mesin yang dibutuhkan SPV dan melakukan kontrol terhadap jadwal perawatan. Selain itu, sistem informasi ini berbasis CMMS yang umumnya digunakan sebagai alat kontrol terhadap aset perusahaan seperti mesin dan spare part selama tiga dekade terakhir telah berkembang pesat dari sebatas penyedia informasi aset perusahaan dan fungsi <i>preventive maintenance</i> , menjadi sistem informasi <i>maintenance</i> perusahaan. Pengkodean mesin, komponen dan <i>sparepart</i> dengan sistem kodefikasi <i>hybrid</i> yang dihubungkan dengan <i>database</i> dapat mempermudah pencarian informasi <i>sparepart</i> . |

2.2 PERAWATAN

2.2.1 Pengertian dan Tujuan Perawatan

Perawatan atau pemeliharaan adalah suatu kombinasi dari berbagai tindakan yang dilakukan untuk menjaga atau memperbaikinya sampai suatu kondisi yang bisa diterima (Asyari, 2007:2). Menurut definisi, istilah “Pemeliharaan” pada kenyataannya menunjuk kepada fungsi pemeliharaan secara keseluruhan yang bisa dibayangkan, dan sebagai hasilnya kata tersebut dengan longgar digunakan dalam industri untuk menunjuk setiap pekerjaan yang dikerjakan oleh pekerja bagian pemeliharaan (Corder, 1988:2).

Tujuan perawatan menurut Corder (1988:3) yang utama dapat didefinisikan dengan jelas sebagai berikut:

1. Untuk memperpanjang usia kegunaan aset, yaitu setiap bagian dari suatu tempat kerja, bangunan dan isinya. Hal ini terutama penting di negara berkembang karena kurangnya sumber daya modal untuk penggantian. Di negara maju kadang-kadang lebih menguntungkan mengganti daripada memelihara.
2. Untuk menjamin ketersediaan optimum peralatan yang dipasang untuk produksi (atau jasa) dan mendapatkan laba investasi maksimum yang mungkin.
3. Untuk menjamin kesiapan operasional dari seluruh peralatan yang diperlukan dalam keadaan darurat setiap waktu, misalnya: unit cadangan, unit pemadam kebakaran dan penyelamat, dan sebagainya untuk menjamin keselamatan orang yang menggunakan sarana perawatan.

2.2.2 Manajemen Perawatan

Manajemen perawatan didefinisikan sebagai suatu bagian dari pekerjaan perawatan dan merupakan faktor atau elemen yang sangat penting di dalam pengaturan keseluruhan kerja yang melibatkan sumber daya yang ada, seperti tenaga kerja, mesin, material/sparepart, dan sebagainya. Dalam pelaksanaannya, manajemen perawatan ini meliputi hal-hal seperti perencanaan, pelaksanaan, pengendalian, dan penyempurnaan.

Dalam pengertian yang sempit, manajemen perawatan merupakan pengendalian proses pemeliharaan atau disebut sebagai *maintenance control*. *Maintenance control* dapat berupa pemeriksaan peralatan, penyesuaian, perbaikan dan *overhaul*, baik yang sudah dijadwalkan maupun berdasarkan kondisi peralatan, serta pekerjaan penyempurnaan.

Dalam pengertian yang luas, manajemen perawatan mempunyai fungsi yang lebih luas disebabkan oleh kompleksnya pekerjaan pemeliharaan. Manajemen perawatan dalam hal ini mencakup *engineering* yang di dalamnya termasuk pekerjaan teknis untuk menyempurnakan proses pemeliharaan. *Maintenance engineering* terdiri dari *maintenance administrative engineering, management data and process, maintenance planning and control, material management, investment control and work management*, serta *technical and design*.

2.2.3 Perawatan Preventif

Tindakan perawatan ada bermacam-macam bentuk, namun pada penelitian ini perawatan yang akan dibahas yaitu mengenai sistem perawatan preventif. Perawatan preventif adalah pekerjaan perawatan yang bertujuan untuk mencegah terjadinya kerusakan, atau cara perawatan yang direncanakan untuk pencegahan (Supandi, 1990). Ruang lingkup tindakan perawatan preventif meliputi inspeksi, perbaikan kecil, serta pelumasan dan penyetelan. Tipe pemeriksaan dan perawatan preventif dibuat dengan mempertimbangkan ketersediaan tenaga kerja, *sparepart*, bahan untuk perbaikan, dan faktor lainnya. Keuntungan melakukan pemeriksaan dan perbaikan secara periode pada mesin-mesin adalah dapat diramalkannya total perbaikan pada seluruh peralatan di suatu pabrik oleh para insinyur perawatan. Selanjutnya, kesalahan atau kerusakan dapat diramalkan lebih awal dengan melihat fenomena kenaikan getaran mesin, kenaikan suhu, suara, perbaikan dan lainnya. Dalam hal ini, perbaikan dapat dilakukan segera sebelum terjadi kerusakan yang lebih fatal.

Dalam prakteknya, perawatan preventif yang dilakukan oleh suatu perusahaan dapat dibedakan menjadi perawatan rutin dan perawatan periodik. Perawatan rutin merupakan aktivitas pemeliharaan dan perawatan yang dilakukan secara rutin (setiap hari), misalnya pembersihan peralatan pelumasan oli, pengecekan isi bahan bakar, dan lain sebagainya, sedangkan perawatan periodik merupakan aktivitas pemeliharaan dan perawatan yang dilakukan secara periodik atau dalam jangka waktu tertentu, misalnya setiap 100 jam kerja mesin, lalu meningkat setiap 500 jam sekali, dan seterusnya.

2.2.4 Corrective Maintenance

Corrective Maintenance adalah suatu kegiatan perawatan yang terjadwal dilakukan untuk memperbaiki dan meningkatkan kondisi mesin sehingga mencapai standar yang telah ditetapkan pada mesin tersebut (Corder, 1988:4). Kegiatan pemeliharaan dan perawatan *corrective maintenance* dilakukan untuk memperbaiki masalah-masalah khusus yang sebelumnya telah teridentifikasi dalam sebuah sistem.

Pada umumnya, *Corrective Maintenance* bukanlah aktivitas perawatan yang terjadwal, karena dilakukan setelah sebuah komponen mengalami kerusakan dan bertujuan untuk mengembalikan kehandalan sebuah komponen atau sistem ke kondisi semula. *Corrective maintenance*, dikenal sebagai *breakdown* atau *run to failure maintenance*. Pemeliharaan hanya dilakukan setelah peralatan atau mesin rusak. Bila

strategi pemeliharaan ini digunakan sebagai strategi utama akan menimbulkan dampak tingginya kegiatan pemeliharaan yang tidak direncanakan dan inventori part pengganti.

2.2.5 Computerized Maintenance Management System

Computerized Maintenance Management System (CMMS) adalah aplikasi perangkat lunak yang mengatur *database* berisi informasi aktivitas pemeliharaan dalam suatu perusahaan. Informasi ini digunakan untuk membantu para karyawan dalam melaksanakan tugas pemeliharaan aset agar lebih efektif, seperti misalnya menentukan mesin mana yang sedang memerlukan perawatan dan gudang mana saja yg menyediakan *spareparts* yang akan digunakan dalam proses perawatan tersebut. (Wireman, 1994:24)

Aplikasi CMMS digunakan oleh perusahaan yang harus melakukan perawatan dan pemeliharaan terhadap peralatan aset dan properti. Paket CMMS juga bisa menampilkan *output maintenance* yang sudah dilakukan sebelumnya. Perangkat lunak CMMS ini juga bisa berupa aplikasi web, dengan menggunakan *server* dari luar maupun internal perusahaan yang hanya bisa diakses dalam lingkup intranet perusahaan.

2.2.5.1 Modul Computerized Maintenance Management System

Modul yang umum terdapat pada perangkat lunak CMMS (Bagadia, 2008:6) adalah sebagai berikut:

1. *Work Orders*

Penjadwalan kerja, alokasi personel, material yang dibutuhkan, biaya yang dibutuhkan dan catatan informasi yang relevan dengan proses *involved*, dan saran untuk melakukan tindak lanjut dalam menanganinya. Dalam *Work Order* terdapat juga sub-komponennya, yaitu:

- a. *Planned Maintenance*

- 1) *Inspections*

Melakukan inspeksi terhadap kondisi peralatan, aset dan properti yang dimiliki.

- 2) *Preventive Maintenance (PM)*

Menjadwalkan pekerjaan untuk *preventive maintenance*, termasuk di dalamnya berupa instruksi langkah demi langkah atau *check lists*, daftar material yang dibutuhkan dan detail lain yang masih berhubungan dengan proses *maintenance* yang dilakukan sebelumnya.

b. *Unplanned Maintenance*

1) *Machine Breakdown*

Terjadinya kerusakan pada peralatan, detail dari perbaikan yang sudah dan akan dilakukan.

2) *Corrective Maintenance*

Melakukan pengecekan terhadap peralatan ketika keandalan dari peralatan tersebut dirasa sudah menurun dari kondisi normal.

2. *Asset Management*

Data mengenai peralatan dan properti meliputi aktivitas *maintenance*, spesifikasi, tanggal pembelian, *warranty information*, *service contracts*, *service history*, *spare-parts* dan informasi lain yang dibutuhkan oleh manajemen dan pekerja *maintenance*.

3. *Inventory Control*

Manajemen untuk mengatur *spare parts*, peralatan dan material lain termasuk juga *reserved materials* untuk *job* tertentu, gudang lokasi dari material tersebut, menentukan kapan material dibeli, catatan pembelian dan catatan pengambilan material dari gudang.

4. *Labor/Employee*

Manajemen untuk mengatur data karyawan yang bekerja dalam departemen *maintenance*. Modul ini akan terintegrasi dengan modul lain sebagai subjek pelaksana.

5. *Safety Requirements*

Manajemen untuk mengatur perijinan dan dokumen yang diperlukan untuk proses *safety requirements*.

6. *Vendor*

Modul ini memberikan informasi terhadap *supplier* yang dimiliki untuk mensuplai peralatan dan *sparepart* maupun *service* yang diberikan.

7. *Purchasing*

Purchasing modul bertujuan untuk membuat dan memproses permintaan pembelian serta membuat *Plan Order* (PO). Dengan modul ini, *user* dapat melakukan maupun mengecek mengenai order yang sudah dilakukan.

8. *Budgeting Module*

Modul ini meberikan laporan keuangan departemen *maintenance* mengenai biaya apa saja yang sudah dikeluarkan untuk memenuhi kebutuhan *maintenance* perusahaan.

2.2.6 Overall Equipment Effectiveness

Overall Equipment Effectiveness (OEE) merupakan suatu metode pengukuran yang dipergunakan dalam implementasi *Total Productive Maintenance* (TPM) dengan mempertimbangkan kombinasi dari operasi pemeliharaan, manajemen peralatan, serta ketersediaan sumber daya (Chan et al., 2003: 24). Dalam hal ini, OEE bertujuan untuk mengukur keefektifan penggunaan mesin atau peralatan beserta tingkat efisiensinya (Purwono, 2013: 71).

Klasifikasi kerugian (*losses*) peralatan diklasifikasi menjadi enam kategori utama (Nakajima, 1988: 28), yaitu:

- 1) *Breakdown losses* adalah *losses* yang disebabkan oleh adanya kegagalan (*failure*) yang kronis ataupun sporadik. Kegagalan sporadik terjadi ketika tiba-tiba terjadi beberapa perubahan misal seperti metode kerja yang membutuhkan penyesuaian kembali sehingga perlu kembali seperti awal.
- 2) *Set-up and adjustment losses* adalah *losses* waktu yang hilang saat menyelesaikan produksi item sebelumnya untuk kemudian berganti memproduksi item baru.
- 3) *Minor/idling stoppage losses* adalah *losses* yang terjadi karena adanya produksi yang terputus oleh kesalahan kerja sementara (*temporary malfunction*) atau ketika mesin dalam keadaan menganggur (*idle*).
- 4) *Reduced speed losses* adalah *losses* yang terjadi karena kecepatan mesin menurun
- 5) *Defect/rework losses* adalah *losses* yang disebabkan oleh adanya cacat ataupun rework terhadap produk yang dihasilkan, serta hilangnya sejumlah waktu tertentu akibat proses perbaikan terhadap produk cacat menjadi produk dengan kondisi yang lebih baik (*excellent product*).
- 6) *Start-up losses* adalah waktu atau volume yang hilang karena terjadinya beberapa hal seperti *start-up* setelah periode perbaikan, *start-up* setelah pemberhentian operasi yang cukup lama, *start-up* setelah waktu libur, dan *start-up* setelah waktu istirahat.

Untuk *breakdown losses* dan *set-up and adjustment losses* didefinisikan sebagai *time losses* dan dipergunakan untuk menentukan *availability* dari peralatan. Nilai *availability* dihitung dengan rumus (Stephens, 2004):

$$\text{Equipment availability rate} = \frac{\text{loading time} - \text{downtime}}{\text{loading time}} \quad (2-1)$$

Minor/idling stoppage losses dan *reduce losses* dipergunakan untuk mengukur *performance rate* dari peralatan. *Performance rate* dapat dihitung dengan rumus (Stephens, 2004):

$$\text{Performance rate} = \frac{\text{process amount} \times \text{actual time}}{\text{process time}} \quad (2-2)$$

Sedangkan *defect/rework losses* dan *start-up losses* berkaitan dengan *quality rate* atau *quality efficiency*. *Rate Of Quality Product* dapat dihitung dengan rumus (Stephens, 2004):

$$\text{Quality Rate} = \frac{\text{process amount} - \text{defect amount}}{\text{process amount}} \quad (2-3)$$

Penentuan nilai OEE mengombinasikan ketiga persamaan tersebut sehingga diperoleh persamaan:

$$\text{OEE} = \text{Equipment Availability Rate} \times \text{Performance Rate} \times \text{Quality Rate} \quad (2-4)$$

Nilai untuk masing-masing dari *availability*, *peformance efficiency*, dan *quality rate* yang ideal berturut-turut adalah 90%, 95%, dan 99%. Sedangkan untuk nilai dari OEE berdasarkan TPM adalah sebesar 85% (Ahuja and Khamba, 2008: 21).

2.3 SISTEM INFORMASI

2.3.1 Pengertian Sistem

Terdapat dua pendekatan di dalam mendefinisikan sistem, yaitu yang menekankan pada prosedur dan yang menekankan pada komponen atau elemennya. Pendekatan sistem yang lebih menekankan pada prosedur mendefinisikan sistem sebagai suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran tertentu. Sedangkan pendekatan sistem yang lebih menekankan pada pada komponen atau elemen mendefinisikan sistem sebagai kumpulan dari elemen-elemen yang berinteraksi untuk mencapai suatu tujuan tertentu.

Menurut Kusrini, M. Kom (2007:11) sistem adalah tatanan yang terdiri atas sejumlah komponen fungsional (dengan tugas/fungsi khusus) yang saling berhubungan dan secara bersama-sama bertujuan untuk memenuhi suatu proses/pekerjaan tertentu. Richardus Eko Indrajit (2001:2) dalam bukunya juga menjelaskan sistem adalah kumpulan dari komponen-komponen yang memiliki unsur atau keterikatan antara satu dengan yang lainnya. Sedangkan Menurut Fathansyah (2002:9) sistem adalah sebuah tatanan (keterpaduan) yang terdiri atas sejumlah komponen fungsional (tugas khusus) yang saling berhubungan dan secara bersama-sama bertujuan untuk memenuhi suatu

proses. Dari beberapa pengertian sistem diatas, dapat disimpulkan bahwa sistem adalah kumpulan elemen atau kumpulan prosedur yang saling berhubungan satu dengan yang lainnya untuk mencapai tujuan tertentu.

2.3.2 Pengertian Informasi

Menurut (Kusrini and Koniyo,2007:7), informasi adalah data yang sudah diolah menjadi sebuah bentuk yang berarti bagi pengguna, yang bermanfaat dalam pengambilan keputusan saat ini atau mendukung sistem informasi. Menurut (Amsyah,2003:5) Informasi adalah bahan yang dihasilkan dari pengolahan data. Pengertian informasi menurut Gordon B. Davis(2005:3) dalam bukunya yang berjudul *Management Information System*, informasi adalah data yang sudah diproses menjadi bentuk yang berguna bagi pemakai, dan mempunyai nilai pikir yang nyata bagi pembuatan keputusan pada saat sedang berjalan atau untuk prospek masa depan.

2.3.2.1 Kualitas Informasi

Menurut Raymond McLeod (2005), ada empat dimensi dasar informasi yang memberikan kontribusi pada nilai informasi. Keempat dimensi itu adalah:

1. Akurat (*Accurate*)
Informasi harus bebas dari kesalahan-kesalahan dan tidak bisa atau menyesatkan.
2. Tepat Waktu (*Timeslines*)
Informasi harus up to date, datang pada penerima tidak boleh terlambat.
3. Relevan (*Relevance*)
Informasi tersebut mempunyai manfaat untuk pemakainya.
4. Kelengkapan
Informasi dapat dikatakan lengkap ketika informasi tersebut mempunyai jumlah yang tepat dan menunjang semua area di mana keputusan akan dibuat.

2.3.3 Pengertian Sistem Informasi

Menurut Robert A. Leitch dan K. Roscoe Davis dalam buku Jogiyanto H.M. (2005 :11) pengertian sistem informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi, bersifat manajerial dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan. Menurut Alter dalam buku Abdul Kadir (2005

: 546) sistem informasi adalah kombinasi antar prosedur kerja, informasi, orang dan teknologi yang diorganisasikan untuk mencapai tujuan dalam sebuah organisasi.

2.4 SISTEM DATABASE

Database merupakan kumpulan data yang terhubung dan disimpan secara bersama pada suatu media dengan cara-cara tertentu sehingga mudah untuk digunakan maupun ditampilkan kembali, serta dapat digunakan oleh satu atau lebih program aplikasi secara optimal, data disimpan sedemikian sehingga penambahan, pengambilan serta modifikasi data dapat dilakukan dengan mudah dan dikontrol dengan baik (Utami & Sutrisno, 2005: 1).

2.4.1 Komponen Sistem Database

Berikut ini adalah komponen penting dalam Sistem *Database* (Utami & Sutrisno, 2005: 1):

1. Data

Ditinjau dari jumlah pemakai, data dikelompokkan menjadi dua macam, yaitu:

- a. *Data single user*, merupakan suatu sistem *database* yang hanya dapat dioperasikan oleh satu pemakai.
- b. *Data multi user*, merupakan satu atau lebih pemakai dapat beroperasi secara bersamaan ke dalam *database*, sehingga data yang ada dalam *database* dapat digunakan bersama dan saling terintegrasi.

2. Hardware

Merupakan piranti keras yang dibutuhkan oleh manajemen *database*, yang biasanya merupakan mesin atau alat yang standar.

3. Software

Merupakan piranti lunak yang menjadi penghubung antara *database* dengan pemakai yang disebut sebagai *Database Management System* (DBMS) atau *database* manajer. Tugas dari DBMS ini adalah melakukan semua kebutuhan akses data.

4. User

User atau pemakai *database* dapat dikelompokkan menjadi tiga, yaitu:

- a. *Programmer* aplikasi, yaitu orang yang bertanggung jawab untuk menulis program aplikasi.

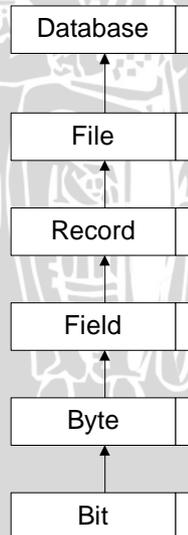
- b. *End user*, yaitu orang yang menggunakan data dalam *database* untuk kebutuhan tugas atau fungsinya.
- c. Administrator *database*, yaitu orang yang bertanggung jawab pada keseluruhan sistem *database*.

2.4.2 Pengertian Data

Menurut kamus besar bahasa Indonesia, data merupakan istilah yang berasal dari kata “*datum*” yang berarti fakta atau bahan-bahan keterangan kumpulan kejadian yang diangkat dari suatu kenyataan. Data dapat berupa angka-angka, huruf, atau simbol-simbol khusus atau gabungan darinya. Data mentah masih belum bisa bercerita banyak, sehingga perlu diolah lebih lanjut. Pengolahan data merupakan manipulasi dari data ke dalam bentuk yang lebih berguna dan lebih berarti berupa suatu informasi (McLeod, 2004: 167).

2.4.3 Hierarki Data

Berdasarkan tingkat kompleksitas nilai data, tingkatan data dapat disusun ke dalam sebuah hirarki, mulai dari yang paling sederhana hingga paling kompleks yang dapat digambarkan dengan Gambar 2.1.



Gambar 2.1 Hierarki Data

Berdasarkan Gambar 2.1, pengorganisasian data dapat dibagi menjadi enam tingkatan, yaitu (Santoso,2005):

1. *Bit* adalah suatu sistem angka biner yang terdiri atas dua macam nilai saja, yaitu 0 dan 1. Sistem angka biner merupakan dasar-dasar yang dapat digunakan untuk komunikas antara manusia dan mesin (komputer) yang merupakan sekumpulan

komponen elektronik dan hanya dapat membedakan dua keadaan saja (*on* dan *off*). Jadi, bit adalah unit terkecil pembentuk data.

2. *Byte* adalah bagian terkecil yang dapat dialamatkan dalam memori. *Byte* merupakan sekumpulan *bit* yang secara konvensional terdiri atas delapan *bit*. Satu *byte* digunakan untuk mengkodekan satu buah karakter dalam memori. Contoh: Kode ASCII untuk J ialah 10101010. Jadi *byte* adalah kumpulan *bit* yang membentuk suatu karakter (huruf, angka atau tanda). Dengan kombinasi 8 *bit*, dapat diperoleh 256 karakter (= 2 dipangkat 8).
3. *Field* atau kolom adalah unit terkecil yang disebut data. *Field* merupakan sekumpulan *byte* yang mempunyai makna. Contoh: Joni yang merupakan *field* nama. Jadi, *field* ibarat kumpulan karakter yang membentuk suatu kata.
4. *Record* atau baris adalah kumpulan item yang secara *logic* saling berhubungan. Setiap *record* dapat dikenali oleh sesuatu yang mengenalinya, yaitu *field* kunci. Jadi, *record* ibarat kumpulan data yang membentuk satu kalimat yang berarti.
5. *File* atau tabel adalah kumpulan *record* yang sejenis dan secara logis berhubungan. Pembuatan dan pemeliharaan *file* adalah faktor yang sangat penting dalam sistem informasi manajemen yang memakai komputer. Jadi, tabel ibarat kumpulan baris/*record* yang membentuk satu tabel.
6. *Database* merupakan kumpulan *file-file* yang berhubungan secara logis dan digunakan secara rutin pada operasi-operasi sistem informasi manajemen. Semua *database* umumnya berisi elemen-elemen data yang disusun ke dalam *file-file* yang diorganisasikan berdasarkan sebuah skema atau struktur tertentu, tersimpan di *hardware* komputer dengan *software* untuk melakukan manipulasi data untuk kegunaan tertentu. Jadi, suatu *database* adalah menunjukkan suatu kumpulan tabel yang dipakai dalam suatu lingkup perusahaan atau instansi untuk tujuan tertentu.

2.4.4 Tujuan Sistem *Database*

Tujuan utama dari sistem *database* adalah praktis dan efisien. Secara umum, tujuan dari sistem *database* adalah sebagai berikut:

1. Isolasi data, menempatkan tiap data pada tempatnya masing-masing.
2. Multi *user*, saat perusahaan mengambil pendekatan berorientasi masalah, pertama didefinisikan kemudian pengambilan keputusan untuk penyelesaian masalah tersebut dan untuk pengambilan keputusan diperlukan informasi.

Sedangkan tujuan dari manajemen sistem *database* adalah sebagai berikut:

1. Menyediakan tempat penyimpanan massal untuk data yang relevan.
2. Memudahkan pemakai dalam mengakses data.
3. Memungkinkan respon yang segera atas permintaan data dari pemakai.
4. Melakukan modifikasi terakhir dengan segera pada *database*.
5. Memungkinkan perkembangan lebih lanjut dalam sistem *database*.
6. Meminimasi duplikasi dan redudansi dalam penyimpanan data.
7. Memungkinkan secara serentak dan bersamaan beberapa pemakaian *database* yang berarti juga meningkatkan kebebasan data sehingga berguna untuk beberapa program.
8. Melindungi data dari gangguan kerusakan atau pemakaian oleh orang yang tidak terotorisasi.

2.4.5 Entity Relational Diagram (ERD)

Menurut Kusri (2005: 21), perancangan basis data dengan menggunakan model entity relationship adalah dengan menggunakan Entity Relationship Diagram (ERD). Terdapat tiga notasi dasar yang bekerja pada model E-R yaitu *entity sets*, *relationship sets*, dan *attributes*. ERD adalah hasil akhir dari proses analisis terhadap sistem yang ditinjau yang dilakukan oleh seorang analisis sistem (Heryanto, 2004;12). Model E-R merupakan rincian yang merupakan representasi logika dari data pada suatu organisasi atau area bisnis tertentu. Model E-R terdiri dari beberapa komponen dasar yaitu sebagai berikut: (Kusri,2007).

1. Entitas

Entitas adalah sesuatu atau objek di dunia nyata yang dapat dibedakan dengan sesuatu atau objek yang lainnya. Contohnya, setiap karyawan dalam suatu kantor adalah suatu entitas. Setiap divisi dalam suatu kantor merupakan suatu entitas. Dapat dikatakan bahwa entitas bisa bersifat konseptual/abstrak atau nyata hadir di dunia nyata.



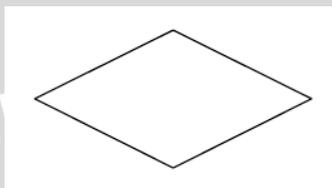
Gambar 2.2 Lambang Entitas

2. Hubungan antar relasi (*Relationship*)

Relationship adalah hubungan diantara beberapa entiti. Relationship set adalah sekumpulan relasi yang mempunyai tipe yang sama. Relationship set

digambar dengan diamond seperti tampak pada Gambar 2.3. Misalnya, entitas mahasiswa memiliki hubungan tertentu dengan entitas matakuliah (mahasiswa mengambil matakuliah). Pada penggambaran model E-R, relasi adalah perekat yang menghubungkan suatu entitas dengan entitas yang lainnya. Tipe-tipe relasi tersebut adalah:

- a. Relasi satu ke satu (*one-to-one*)
- b. Relasi satu ke banyak (*one-to-many*)
- c. Relasi banyak ke banyak (*many-to-many*)



Gambar 2.3 Lambang *Relationship*

3. Kardinalitas/Derajat Relasi

Kardinalitas relasi menunjukkan jumlah maksimum entitas yang dapat berelasi dengan entitas pada himpunan entitas yang lain. Sebagai contoh: entitas-entitas pada himpunan entitas mahasiswa dapat berelasi dengan satu entitas, banyak entitas atau tidak satupun entitas dari himpunan entitas kuliah. Tabel 2.2 menjelaskan macam-macam relasi yang ada dalam kardinalitas, berikut penjelasan dari masing-masing relasi yang ada Kusrini(2007:24).

- a. Hubungan banyak ke satu (*Many to One*) satu entity dalam A dihubungkan dengan maksimum satu entity B. satu entity dalam B hubungkan dengan sejumlah entity dalam A.
- b. Hubungan satu ke satu (*One to One*) yaitu satu entity dalam A dihubungkan dengan maksimum satu entity
- c. Hubungan satu ke banyak (*One to Many*) yaitu satu entity dalam A dihubungkan dengan sejumlah entity dalam entity dalam B dihubungkan dengan maksimum satu entity dalam A.
- d. Hubungan banyak ke banyak (*Many to Many*) satu entity dalam A dihubungkan dengan sejumlah entity dalam entity dalam B dihubungkan dengan sejumlah entity dalam A.

Berikut merupakan Tabel 2.2 yang menjelaskan kriteria masing-masing kardinalitas pada ERD yang telah dijelaskan.

Tabel 2.2 Kardinalitas Minimum dan Maksimum Pada Relasi Dua Entitas dalam ERD

| No. | Notasi | Derajat Relasi Maks-Min | Nama Relasi |
|-----|---|-------------------------|---------------------|
| 1. |  | (N , 1) | <i>Many to One</i> |
| 2. |  | (1 , 1) | <i>One to One</i> |
| 3. |  | (1 , N) | <i>One to Many</i> |
| 4. |  | (N , N) | <i>Many to Many</i> |

Sumber: Lio (2010)

2.4.6 Normalisasi

Hal yang sangat penting dalam melakukan perancangan *database* adalah merancang *database* yang kompak dan efisien dalam penggunaan ruang penyimpanan, kecepatan pengambilan data, dan mudah dalam memanipulasi data. Untuk mencapai tujuan tersebut, maka diperlukan suatu struktur data yang sederhana dan stabil. Proses untuk mengubah suatu bentuk struktur data menjadi lebih sederhana dan stabil disebut dengan normalisasi. Normalisasi didefinisikan sebagai suatu teknik yang menstrukturkan atau mendekomposisi atau memecah data menggunakan cara-cara tertentu untuk mencegah timbulnya permasalahan pengolahan data dalam basis data (Munif, 2013: 101). Berikut ini diberikan gambaran tentang aturan tersebut secara ringkas.

1. *First normal form* (1NF).

Menghapus semua elemen yang berulang, sehingga hanya terdapat satu nilai untuk setiap perpotongan baris dan kolom pada tabel.

2. *Second normal form* (2NF)

Setiap atribut bukan kunci (*non key*) yang terdapat pada relasi harus tergantung pada *primary key*. Apabila tidak, maka harus dibuat table baru.

3. *Third normal form* (3NF)

Jika terdapat kesalingtergantungan antar dua atribut yang bukan merupakan atribut kunci, maka harus dibuat tabel baru. Dengan kata lain, setiap relasi (tabel) hanya memuat satu kepentingan.

2.5 GROUP TECHNOLOGY

Group Technology adalah suatu filosofi dalam manufaktur yang mengidentifikasi keserupaan komponen-komponen dan mengelompokkannya bersama dengan mengambil keuntungan dari keserupaan dalam desain dan manufaktur. Konsep dasar dari *Group Technology* adalah menyederhanakan dan standardisasi proses (Burbidge, 1993:5).

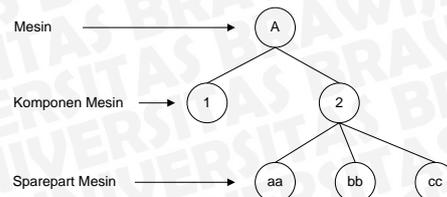
Sedangkan Ranson (1972) mendefinisikan *group technology* sebagai bentuk pengaturan secara logis dari semua aspek operasi yang bertujuan untuk mendapatkan keuntungan dari jumlah produksi yang tinggi dan jenis produk yang beragam.

2.5.1 Metode Klasifikasi *Group Technology*

Metode klasifikasi pada *group technology* terdiri dari tiga macam, yaitu berdasarkan bentuk *part*, proses, dan pengkodean. Dalam klasifikasi berdasar bentuk, *part* dikelompokkan berdasarkan keserupaan bentuk geometrik. Klasifikasi yang kedua adalah klasifikasi berdasarkan proses dimana *part* dikelompokkan berdasar pada keserupaan proses yang dialami. Klasifikasi yang terakhir adalah klasifikasi dengan pengkodean. Menurut Hyer (1985:23), sistem klasifikasi dan kodefikasi dengan pengkodean berdasar pada atribut yang ada pada *part*, kemudian dilakukan pengkodean. Kode-kode tersebut mengandung informasi yang ada pada *part* yang bersangkutan. Kode dapat berupa huruf, angka atau gabungan keduanya. Namun demikian, kode *part* tidak sama dengan *part number* yang biasa terdapat pada *part*. Berikut merupakan 3 macam tipe klasifikasi kode dalam *group technology* (Kusiak,1990):

1. *Monocode* (Struktur Hirarki)

Pada struktur hirarki ini masing-masing *digit* memperkuat informasi dari *digit* sebelumnya. Hal ini mengindikasikan bahwa deretan kode pada setiap *digit* yang ada bergantung pada karakter *digit* sebelumnya. Tiap-tiap *digit* (posisi) pada *code* direpresentasikan dalam *fitur/subsection*. *Digit* pertama mewakili seluruh *group*, *digit* selanjutnya mewakili kelompok *sub-section* dan seterusnya. Struktur hirarki ini mengijinkan beberapa bagian kode digunakan untuk informasi yang berbeda. Namun bagaimanapun juga sistem kodifikasi berdasarkan struktur sangatlah susah dan rumit untuk diimplementasikan untuk *group* dengan *fitur/section* yang banyak. Pada penelitian ini, kodefikasi dengan struktur hirarki diterapkan pada *section* komponen dan *sparepart*, dimana mesin merupakan perwakilan dari seluruh *group* yang terdiri dari komponen dan *sparepart*, komponen mesin bergantung pada salah satu mesin, dan *sparepart* bergantung pada komponen. Komponen dan *sparepart* pada setiap mesin berbeda-beda, inilah yang menyebabkan kodefikasi setiap mesin, komponen dan *sparepart*-nya juga berbeda. Misalnya, mesin A mempunyai komponen 1, 2, dan 3 dan pada komponen 1 terdapat *sparepart* aa,bb, dan cc. Hal ini dapat diilustrasikan pada Gambar 2.4.



Gambar 2.4 Hierarki Klasifikasi *Sparepart*
Sumber: Chang (1991)

2. *Polycode* (Kode Atribut)

Konsep kode Atribut *chain-structure (polycode)* yang digunakan pada penelitian ini mempunyai arti bahwa masing-masing *digit* kode yang digunakan tidak tergantung pada karakter deretan kode sebelum *digit* kode ini sehingga dapat mengakomodasi setiap perubahan. Pada konsep kode ini, masing-masing bagian dalam kode mempunyai posisi yang spesifik. Struktur kodefikasi ini sangat mudah diterapkan tetapi jumlah *digit* yang besar memerlukan perwakilan karakteristik dari suatu *section*. Contoh konsep kode atribut yang diterapkan pada kodefikasi mesin di penelitian ini adalah untuk menentukan kode letak *sparepart*, selain itu untuk menentukan kode dari merk mesin, nama mesin, dan tahun. Hal ini dapat dijelaskan pada Tabel 2.3.

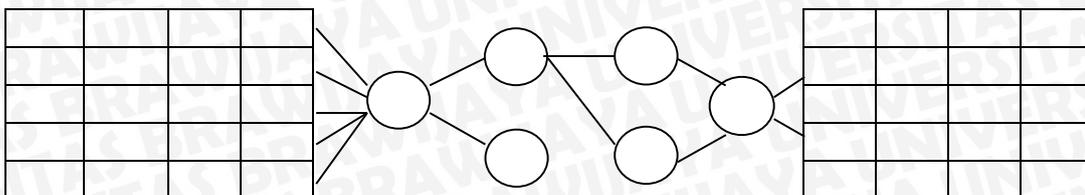
Tabel 2.3 Kode Atribut *Inventory*

| Digit | Lokasi | 1 | 2 |
|-------|--------|----------|----------|
| 1 | Area | Lemari A | Lemari B |
| 2 | Rak | Rak 1 | Rak 2 |
| ... | ... | | |

Sumber: Houtzeel dan Schilperoort (1976)

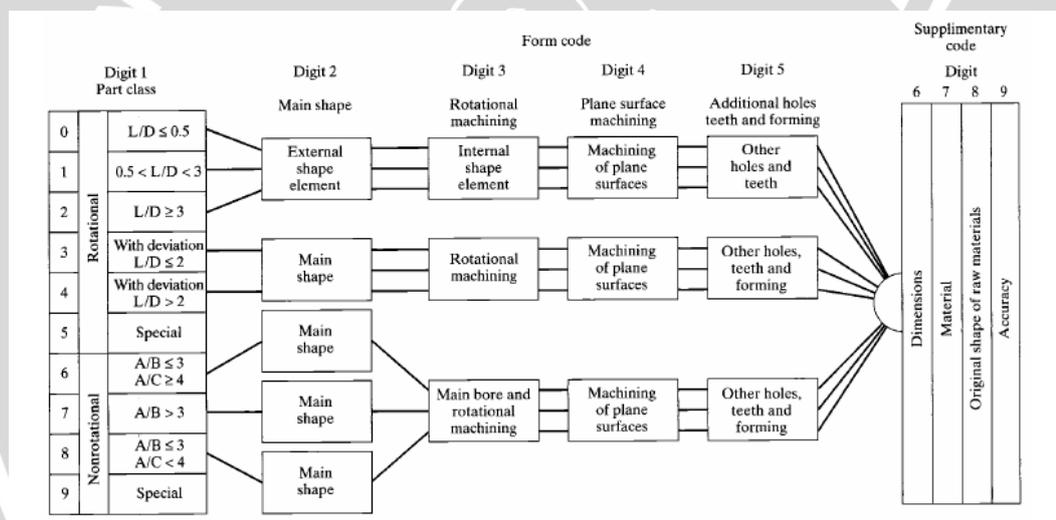
3. *Hybrid* (Penggabungan *monocode* dan *polycode*)

Pada penelitian ini, pengkodean yang digunakan adalah sistem pengkodean *hybrid*, yang merupakan penggabungan sistem pengkodean *monocode* dan *polycode* dengan memanfaatkan keuntungan dari setiap karakteristik sistem kodefikasi *monocode* dan *polycode*. Sistem pengkodean *hybrid* ini banyak digunakan pada sistem kodefikasi mesin dan *part* karena sistem kodefikasi ini dianggap praktis dengan memberikan karakteristik *part* yang dijelaskan dengan *polycode* dan memberikan hirarki sebuah *part* dari induknya yang dijelaskan dengan *monocode*. Berikut ilustrasi sistem kodifikasi *hybrid* yang dapat dilihat pada Gambar 2.5.



Gambar 2.5 Sistem Kode *Hybrid*
Sumber: Opitz (1970)

Pada Gambar 2.5, tabel di awal menggambarkan *polycode* yang menggambarkan karakteristik suatu *part* dan dilanjutkan dengan *monocode* yang merupakan hirarki *part* sebelumnya hingga bagian terkecil dari *part* dan dilanjutkan kembali dengan *polycode* yang merupakan atribut atau karakteristik dari sub *part* yang lebih kecil dari sebelumnya. Kodefikasi juga diadopsi oleh Opitz(1970) dalam pemberian kodefikasi *sparepart*. Sistem klasifikasi Opitz secara umum menggunakan 9 digit kode dengan 4 digit kode sebagai tambahan. Untuk 9 digit pertama menjelaskan tentang desain *sparepart* dan data produksi dengan rincian 5 digit pertama disebut *form code* yang mendeskripsikan desain utama *attribute* dari *sparepart* tersebut. 4 digit selanjutnya adalah *supplementary code* yang mengidentifikasi *attribute* yang digunakan pada proses manufaktur. Serta 4 digit kode tambahan disebut *secondary code* yang mengidentitaskan tipe operasi produksi dan urutan. Berikut ini adalah contoh dari kodefikasi Opitz.



Gambar 2.6 Contoh Sistem Kode Opitz
Sumber: Opitz (1970)

2.6 SOFTWARE PROTOTYPING

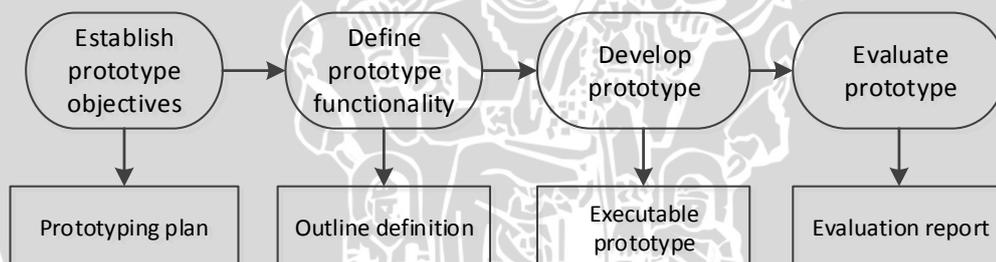
Menurut Kendall (2003:6), *prototyping* adalah suatu teknik pengumpulan data yang sangat berguna melengkapi siklus hidup pengembangan sistem tradisional. Saat penganalisis sistem menggunakan *prototyping*, mereka berusaha mencari reaksi, saran – saran, inovasi, rencana revisi pengguna untuk membuat peningkatan terhadap *protoype* sekaligus memodifikasi rencana sistem dengan biaya dan gangguan maksimum.

Prototyping memiliki beberapa arti berbeda, dimana empat diantaranya paling banyak digunakan. Definisi *protoype* pertama adalah penyusunan *protoype patch*. Definisi kedua adalah *protoype* operasional yang digunakan untuk menguji fitur – fitur

desain tertentu. Konsepsi ketiga adalah menciptakan *prototype* pertama yang benar – benar operasional. Sedangkan jenis *prototype* yang keempat adalah *prototype* fitur – fitur terpilih yang memiliki beberapa fitur – fitur sistem inti. Empat petunjuk utama untuk mengembangkan suatu *prototype* adalah:

1. Bekerja menurut modul – modul dapat dikendalikan
2. Membangun *prototype* dengan cepat
3. Memodifikasi *prototype*
4. Menekankan *interface* pengguna.

Sommerville (2011:14) menjelaskan bahwa terdapat empat tahapan dalam pengembangan *software* menggunakan *prototyping*. Tahapan pertama yaitu menetapkan tujuan *prototype* dengan melakukan identifikasi masalah pada sistem yang akan dibuat *prototype*-nya. Tahapan kedua yaitu mendefinisikan fungsi *prototype* sehingga sesuai dengan kebutuhan *user*. Tahapan ketiga yaitu mengembangkan *prototype* dengan merancang desain logis hingga implementasi konsep yang telah dibuat. Tahapan terakhir yaitu evaluasi *prototype* dengan membandingkan hasil *prototype* dan kebutuhan *user*. Tahapan-tahapan pengembangan tersebut dapat dilihat pada Gambar 2.7.



Gambar 2.7 Model Proses Pengembangan *Prototype*
Sumber: Sommerville (2011)

Berdasarkan Gambar 2.7 model proses pengembangan *prototype* yang lebih rinci dapat dijelaskan sebagai berikut (Sommerville, 2011:14):

1. Menetapkan tujuan *prototype*
 - a. Mengidentifikasi masalah dalam sistem yang sedang berjalan dengan menggunakan analisis PIECES (*Performance-Information-Economy-Control-Efficiency-Services*)
 - b. Menetapkan batasan-batasan (*constraints*) atau ruang lingkup *prototype* yang akan dirancang.
 - c. Menetapkan tujuan dan manfaat dari *prototype*.
2. Mendefinisikan fungsi *prototype*
 - a. Membuat model kebutuhan sistem (*Requirement Modelling*)

Suatu kebutuhan sistem dinyatakan dalam suatu *checklist* yang disebut SRC (*System Requirement Checklist*). SRC adalah fitur-fitur atau karakteristik yang harus ada dalam sistem informasi untuk memenuhi kebutuhan bisnis (*business requirements*) dan yang dapat diterima pengguna. Model kebutuhan sistem ini digambarkan ke dalam lima kategori umum yaitu *output*, *input*, *process*, *performance*, dan *control*.

b. Membuat model data (*Data Modelling*)

Pada pembuatan model data atau *data modelling*, sistem yang akan dirancang digambarkan dengan aliran data dan informasi yang dihasilkan dari dan untuk *entity* dalam sistem dengan *Data Flow Diagram* agar nampak jelas.

c. Membuat model proses (*Process Modelling*)

Process Modelling merupakan penggambaran logika atau aturan bisnis yang dapat dinyatakan dengan *flowchart* atau tabel aturan. Model proses nantinya akan berguna dalam membangun algoritma program komputer.

d. Pengembangan Strategi (*Development Strategies*)

Development Strategies merupakan tahap untuk menggambarkan kegiatan yang tersisa dalam tahap analisis sistem. Tahap ini menjelaskan transisi dari analisa sistem ke sistem desain, *prototyping*, dan pedoman desain sistem diakhiri dengan bagaimana pengembangan perangkat lunak yang dirancang.

3. Mengembangkan *prototype*

a. Langkah Desain

1) Desain *Database Logis*

Konsep model data yang telah terbentuk pada tahap *data modelling* dibawa ke dalam bentuk logis. Desain logis terdiri dari daftar entitas untuk pembuatan tabel dan normalisasi tabel.

2) Desain *Database Fisik*

Desain fisik merupakan aktualisasi dari desain logis. Disini, *entity* sudah berubah menjadi *table* dengan rancangan bentuk fisik.

3) Desain *User Interface* (UI)

Desain *user interface* bertujuan untuk membuat rancangan dari tampilan sistem yang nantinya akan berinteraksi langsung dengan *user* (pengguna). Desain *user interface* harus siap digunakan dan sesuai dengan kebutuhan pengguna.

4) Desain Algoritma

Desain algoritma merupakan desain yang bertujuan untuk merancang tahapan proses apa saja yang harus dilakukan sehingga *input*, *user interface*, dan *database* menghasilkan *output* yang diharapkan dan dapat ditampilkan, algoritma dapat dinyatakan dengan *flowchart* ataupun *pseudocode*.

- b. Implementasi, tahapan ini merupakan tahapan pembuatan aplikasi pada tingkatan *prototype* dari spesifikasi dan konsep desain yang dirancang dengan melakukan pengembangan *database*, *module* dan *user interface*.

4. Mengevaluasi *prototype*

Tahapan ini merupakan tahapan pengujian terhadap program aplikasi yang telah dibuat. Pengujian program ditinjau dari tiga hal, yaitu verifikasi, validasi, dan uji *prototype*.

- a. Verifikasi, merupakan pengujian apakah *prototype* berjalan sesuai dengan yang telah direncanakan. Uji Verifikasi ini meliputi pengujian *hierarki menu*, *form*, *report* dan pengujian ketelitian.
- b. Validasi, merupakan pengujian apakah fungsi *prototype* yang dirancang telah merepresentasikan kebutuhan *user* yang meliputi lima kategori umum; *output*, *input*, proses, *performance*, dan *control* (SRC). Uji *prototype* bertujuan untuk mengetahui apakah *prototype* dapat mengatasi masalah dan kelemahan sistem lama.

Menurut Wetherbe (1994:15), mengidentifikasi dan mengevaluasi permasalahan menggunakan *software prototyping* dapat menggunakan PIECES (*Performance, Information, Economy, Control, Efficiency* dan *Services*).

1. *Performance* (kinerja) merupakan suatu kemampuan sistem dalam menyelesaikan tugas dengan cepat sehingga sasaran dapat segera tercapai. Kinerja diukur melalui jumlah produksi (*throughput*) dan waktu yang digunakan untuk menyesuaikan perpindahan pekerjaan (*response time*).
2. *Information* (informasi) merupakan suatu hal yang penting karena dengan adanya informasi pihak manajemen dapat melakukan pengambilan keputusan dan pengguna dapat melakukan langkah selanjutnya. Apabila kemampuan sistem baik, maka pengguna akan mendapatkan informasi yang akurat, tepat waktu, dan relevan sesuai dengan yang diharapkan.

3. *Economy* (ekonomi) merupakan penilaian sistem atas penghematan dan keuntungan yang akan didapatkan dari sistem yang dikembangkan. Peningkatan terhadap kebutuhan ekonomis mempengaruhi pengendalian biaya dan peningkatan manfaat.
4. *Control* (kendali) dipasang untuk meningkatkan kinerja sistem, mencegah, atau mendeteksi kesalahan sistem, menjamin keamanan data, informasi dan kebutuhan.
5. *Efficiency* (efisiensi) berkenaan dengan bagaimana menghasilkan *output* sebanyak-banyaknya dengan *input* seminimal mungkin. Sistem dapat dikatakan tidak efisien bila banyak waktu yang terbuang pada aktivitas sumber daya manusia, mesin dan komputer, peng-*input*-an data yang berlebihan, pemrosesan data yang berlebihan, atau informasi yang dihasilkan berlebihan.
6. *Service* (pelayanan) berkenaan dengan penilaian dari suatu sistem yang dilihat pula dari kriteria-kriteria, seperti keakuratan dan konsistensi produk yang dihasilkan sistem, kemudahan sistem untuk dipelajari dan digunakan, atau fleksibilitas.

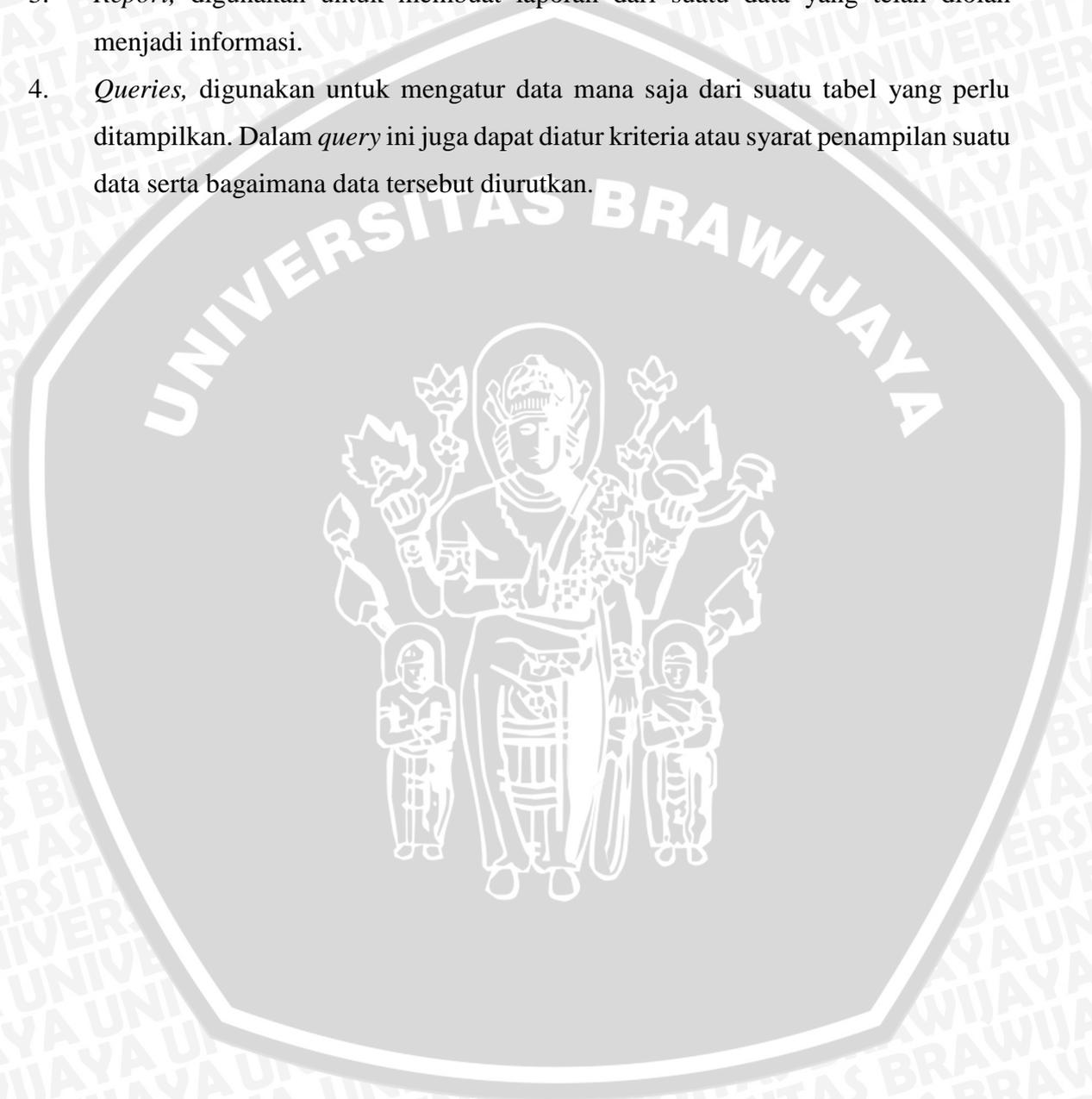
2.7 VISUAL BASIC FOR APPLICATION (VBA)

VBA menurut MacDonald (2010:4) merupakan bahasa pemrograman untuk *Microsoft Office* serta aplikasi-aplikasi yang terkait. Dalam VBA juga menggunakan *macro* untuk membuat objek dalam aplikasi sistem yang koheren. Penggunaan *macro* dapat membantu menyelesaikan tugas – tugas kecil dalam aplikasi *Microsoft*. VBA digunakan karena beberapa alasan, diantaranya adalah VBA dapat membuat aplikasi lebih mudah dipertahankan, dapat membuat fungsi sesuai keinginan, dapat membuat atau memanipulasi objek, dapat melakukan sistem berbasis pada tindakan, dan dapat memanipulasi *record* pada suatu waktu.

2.8 MICROSOFT ACCESS

Microsoft Access merupakan program manajemen *database* yang relatif mudah untuk dipelajari karena dalam *Microsoft Access* diberi banyak sekali fasilitas yang membantu pemakai (Utami & Sukrisno,2005:31). Selain itu, terdapat juga *Wizard* disediakan oleh *Access* untuk merancang *tabel*, *query*, *form* dan *report* dengan cepat dan mudah. Keuntungan utama dari penggunaan *Microsoft Access* yaitu kemampuan integrasinya yang optimal dengan aplikasi *software* lainnya . Keunggulan ini disebabkan oleh suatu sarana penunjang yaitu *Visual Basic for Application*. Berikut ini merupakan beberapa istilah yang digunakan dalam *Microsoft Access*, yaitu:

1. *Table*, merupakan komponen utama yang paling penting dari sebuah *database* serta merupakan objek pertama yang harus dibuat. Tabel dapat berjumlah satu atau lebih sesuai dengan kebutuhan.
2. *Form*, suatu komponen dalam *Microsoft Access* yang digunakan untuk memasukan data ke dalam tabel.
3. *Report*, digunakan untuk membuat laporan dari suatu data yang telah diolah menjadi informasi.
4. *Queries*, digunakan untuk mengatur data mana saja dari suatu tabel yang perlu ditampilkan. Dalam *query* ini juga dapat diatur kriteria atau syarat penampilan suatu data serta bagaimana data tersebut diurutkan.



BAB III METODELOGI PENELITIAN

Pada bagian metodologi penelitian akan menjelaskan mengenai pendekatan, metode, teknik dan langkah-langkah terstruktur dalam melakukan penelitian mulai dari pengumpulan data serta cara analisis data yang sapat membantu pendiskripsian masalah sampai mendapatkan penyelesaian atas masalah yang diteliti. Dengan adanya metodologi penelitian, penyusunan skripsi akan memiliki alur yang terarah dan sistematis.

3.1 METODE PENELITIAN

Metode yang digunakan dalam penelitian ini adalah *software prototyping*. Metode ini sangat baik digunakan untuk menyelesaikan masalah kesalahpahaman antara user dan analis yang timbul akibat user tidak mampu mendefinisikan secara jelas kebutuhannya (Mulyanto, 2009). *Prototyping* adalah pengembangan yang cepat dan pengujian terhadap model kerja (prototipe) dari aplikasi baru melalui proses interaksi dan berulang-ulang yang biasa digunakan ahli sistem informasi dan ahli bisnis.

Prototyping disebut juga desain aplikasi cepat (*rapid application design/RAD*) karena menyederhanakan dan mempercepat desain sistem (O'Brien, 2005). *Prototyping* merupakan metodologi pengembangan *software* yang menitikberatkan pada pendekatan dari sisi desain, fungsi serta *user interface*. Pengembangan *prototype* dimulai dengan tahap mendefinisikan spesifikasi, fungsi, desain dan bagaimana *software* bekerja dan fokus pada *user interface*. Setelah itu akan ditetapkan tujuan umum, kebutuhan yang diketahui dan gambaran bagian-bagian yang akan dibutuhkan. Detail dari kebutuhan tersebut dikumpulkan serta diberikan suatu gambaran dengan *prototype*. Dari proses tersebut akan diketahui detail-detail yang harus dikembangkan atau ditambahkan pada *prototype*, atau menghapus detail-detail yang tidak diperlukan oleh pengguna.

3.2 TEMPAT DAN WAKTU PENELITIAN

Penelitian untuk perancangan sistem informasi manajemen perawatan ini mulai dilaksanakan pada bulan November 2014 – Agustus 2015 untuk menganalisis serta merancang suatu sistem informasi manajemen perawatan di PT. Gunawan Dianjaya Steel Tbk. yang terletak di Jalan Margomulyo 29A, Greges, Asemrowo, Surabaya, Jawa Timur sebagai objeknya.

3.3 DATA YANG DIGUNAKAN

Dalam penelitian ini data yang akan digunakan antara lain adalah:

1. Profil singkat dan gambaran umum PT. Gunawan Dianjaya Steel, Tbk
2. Struktur organisasi departemen *maintenance* di PT. Gunawan Dianjaya Steel, Tbk
3. Sistem perawatan mesin produksi yang telah diterapkan PT. Gunawan Dianjaya Steel, Tbk
4. Data peformansi mesin
5. Data *history* kerusakan mesin
6. Data jadwal perawatan mesin produksi
7. Data karyawan
8. Data *inventory sparepart*
9. Data mesin, komponen mesin dan *sparepart*

3.4 METODE PENGUMPULAN DATA

Pengumpulan data dilakukan untuk memperoleh informasi yang dilakukan dalam rangka tujuan dari penelitian yang dilakukan ini tercapai dengan pencatatan-pencatatan hal-hal atau keterangan-keterangan atau karakteristik-karakteristik sebagian atau seluruh elemen populasi yang akan menunjang serta mendukung penelitian. Adapun metode pengumpulan data yang digunakan pada penelitian ini adalah:

1. Dokumentasi, yaitu cara pengumpulan data dengan mengambil data-data perusahaan berupa laporan-laporan, catatan-catatan, atau arsip-arsip yang sudah ada.
2. Studi Pustaka, sebagai salah satu usaha yang dilakukan oleh peneliti untuk menghimpun informasi yang relevan dengan topik atau masalah yang akan atau sedang diteliti. Studi pustaka dengan melakukan pengumpulan data-data yang bersumber dari buku-buku literatur, catatan-catatan maupun bahan kuliah yang berhubungan dengan masalah yang akan dibahas. Dengan studi pustaka maka diperoleh penjelasan secara teori mengenai permasalahan utama dalam penelitian.

3.5 LANGKAH PENELITIAN

Langkah penelitian merupakan suatu gambaran sistematika penulisan yang akan dijadikan acuan dalam melaksanakan penelitian agar terarah. Langkah-langkah penelitian ini dapat dijelaskan sebagai berikut:

1. Studi lapangan

Tahap awal yang dilakukan untuk memulai penelitian ini adalah dengan melakukan pengamatan secara langsung ke lapangan untuk mengumpulkan informasi yang ada di departemen *maintenance* PT. Gunawan Dianjaya Steel, Tbk. Pengamatan ini dilakukan untuk mengumpulkan informasi seperti kondisi sistem perawatan departemen *maintenance* PT. Gunawan Dianjaya Steel, Tbk, data mesin, komponen, *inventory sparepart*, data karyawan serta data jadwal perawatan mesin yang ada di departemen *maintenance* PT. Gunawan Dianjaya Steel, Tbk.

2. Studi Literatur

Hasil dari tahap studi lapangan perlu didukung oleh studi pustaka dengan melakukan pengumpulan teori yang berhubungan dengan permasalahan yang akan diteliti yang dapat dijadikan referensi untuk mendukung penelitian ini. Sumber pustaka ini dapat diperoleh dari buku, laporan penelitian, jurnal dan internet.

3. Identifikasi Masalah

Berdasarkan studi pendahuluan akan dapat diidentifikasi masalah-masalah yang sedang terjadi pada departemen *maintenance* PT. Gunawan Dianjaya Steel, terutama dalam bidang perawatan mesin dan sistem informasi yang ada.

4. Perumusan Masalah

Setelah dilakukan identifikasi masalah, maka tahap selanjutnya adalah merumuskan masalah sesuai dengan kondisi nyata di PT. Gunawan Dianjaya Steel.

5. Tujuan Perancangan

Tujuan perancangan sistem informasi manajemen perawatan adalah untuk membantu departemen *maintenance* agar dapat lebih efisien dalam mengelola data penjadwalan perawatan mesin dan data *sparepart*.

6. Pengumpulan Data

Pengumpulan data adalah proses atau kegiatan yang dilakukan dalam penelitian untuk mendapatkan informasi yang dibutuhkan di lokasi penelitian yang mendukung kegiatan penelitian. Data yang dibutuhkan dalam penelitian ini antara lain:

- a. Data umum mengenai PT. Gunawan Dianjaya Steel dan departemen *maintenance* PT. Gunawan Dianjaya Steel.
- b. *User requirement*, yang berisi harapan serta atribut/karakter sistem yang dibutuhkan oleh departemen *maintenance* PT. Gunawan Dianjaya Steel yang

nantinya akan menggunakan sistem informasi perawatan mesin yang dirancang.

c. Data historis mesin merupakan data perawatan mesin yang telah dilakukan selama ini, data mesin, data komponen mesin, data karyawan, data jadwal perawatan mesin serta data *inventory sparepart*.

7. Kodenisasi mesin, komponen, serta *sparepart* menggunakan *Group Technology*
Tahap ini melakukan kodenisasi mesin, komponen dan *sparepart* sesuai dengan data yang telah di ambil pada data sebelumnya.

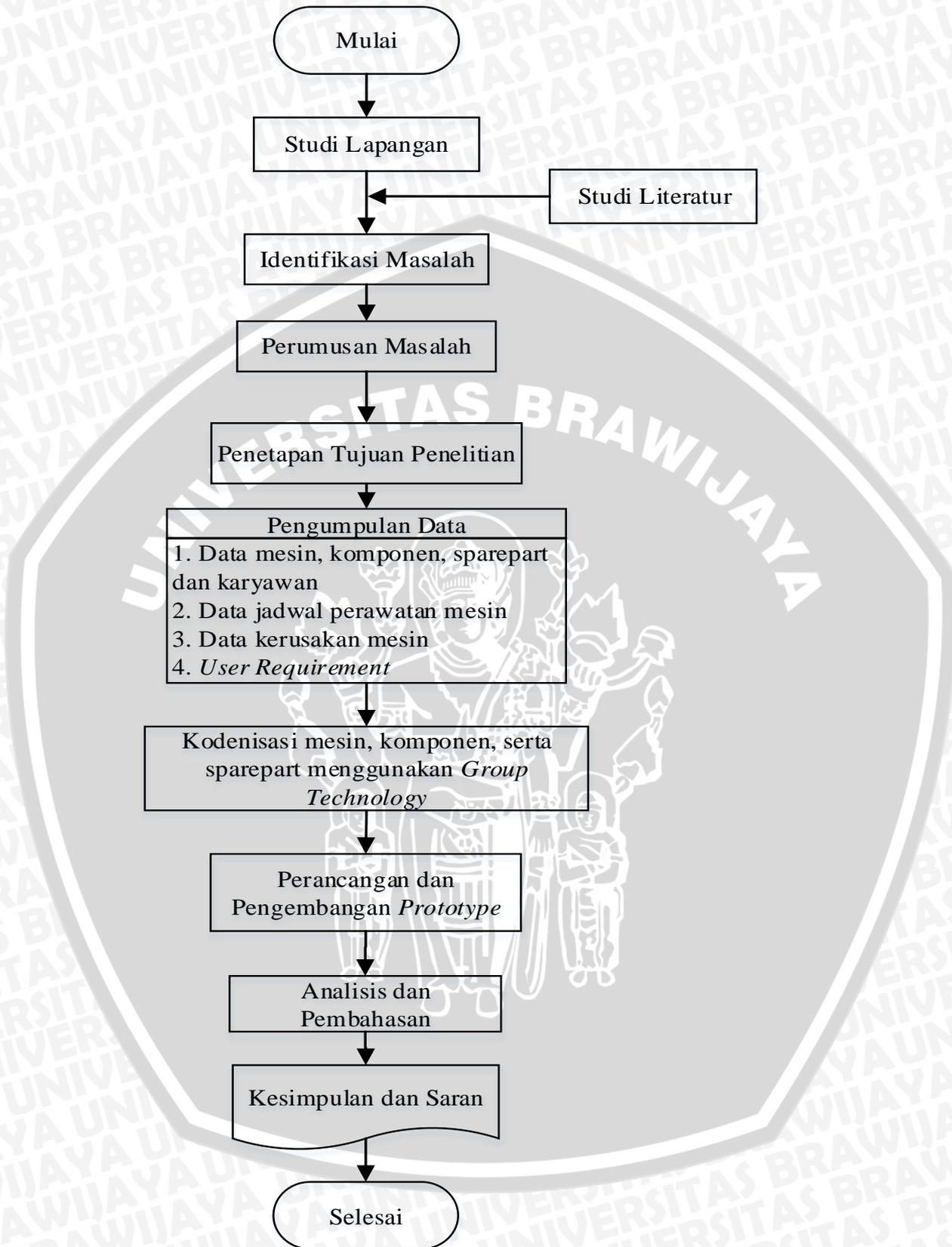
8. Perancangan Sistem dan Pengembangan *Prototype*
Perancangan sistem ini dimulai dengan perencanaan sistem untuk memahami mengapa sebuah sistem harus dibangun, kemudian melakukan analisa terhadap sistem dengan mengidentifikasi semua *entity* yang akan terlibat. Setelah analisa sistem, kemudian dibuatlah rancangan sistem atau konsep dasar yang nantinya akan dikembangkan menjadi sebuah sistem baru. Setelah sistem baru terbuat, akan dilakukan pengujian dengan melakukan uji verifikasi, uji validasi dan terakhir uji *prototype*.

8. Pembahasan
Tahap ini merupakan pembahasan untuk sistem yang telah dirancang apakah telah mencapai tujuan dan menyelesaikan masalah yang ada pada PT. Gunawan Dianjaya Steel.

9. Penarikan Kesimpulan Saran
Pada tahap ini adalah penutup dari keseluruhan langkah penelitian. Kesimpulan berisikan mengenai hasil-hasil analisa dan manfaat yang didapat setelah melakukan penelitian. Saran digunakan sebagai tindak lanjut dari penelitian diharapkan dapat memberi manfaat untuk PT. Gunawan Dianjaya Steel dalam pengembangan Sistem Informasi Manajemen Perawatan mendatang.

3.6 DIAGRAM ALIR PENELITIAN

Tahapan penelitian yang sudah dijelaskan pada sub bab sebelumnya dapat dilihat lebih jelas secara sistematis dengan diagram alir penelitian pada Gambar 3.1.



Gambar 3.1 Diagram Alir Penelitian

BAB IV

PENGUMPULAN DATA DAN ANALISIS SISTEM

Pada bab ini akan dijelaskan mengenai pengumpulan data perusahaan dan langkah analisis. Pengumpulan data di penelitian ini akan lebih mengarah pada pengumpulan data yang didapatkan dari hasil wawancara dengan manajer dan *supervisor* departemen *maintenance* PT. Gunawan Dianjaya Steel Tbk.

4.1 GAMBARAN UMUM PERUSAHAAN

Dalam gambaran umum perusahaan ini akan dijelaskan mengenai profil dari PT. Gunawan Dianjaya Steel Tbk. (PT. GDS), tujuan perusahaan, struktur organisasi departemen *maintenance* perusahaan, dan bagan alir analisis sistem perawatan mesin yang diterapkan pada departemen *maintenance* PT. GDS.

4.1.1 Profil PT. Gunawan Dianjaya Steel

PT. Gunawan Dianjaya Steel (GDS) didirikan pada tahun 1989 bertempat di Margomulyo 29A, Greges-Asemworo, Surabaya. Perusahaan ini terletak di atas lahan seluas kurang lebih 20 hektar dan memulai kegiatan produksi komersial sejak akhir tahun 1993. Berikut pada Gambar 4.1 merupakan kantor perusahaan GDS.



Gambar 4.1 Kantor Perusahaan GDS
Sumber: PT. Gunawan Dianjaya Steel, Tbk

GDS merupakan perusahaan manufaktur di Indonesia yang bertaraf internasional dengan menetapkan 60% produk yang dibuat adalah untuk memenuhi kebutuhan ekspor dan 40% sisanya untuk memenuhi kebutuhan domestik. Perusahaan ini memproduksi *hot roller carbon steel plates*. Plat baja PT. GDS diproduksi dengan spesifikasi internasional dan tersedia dalam berbagai jenis baja, seperti jenis baja umum untuk penggunaan umum, jenis baja struktur yang dapat dilas, jenis baja struktur bangunan, baja untuk

pembangunan kapal, baja boiler dan *pressure vessel*, serta baja struktural berkualitas *intermediate temperature high-strength low alloy*.

Untuk bahan baku dari pelat baja ini, PT. GDS menggunakan baja impor yang biasa disebut dengan *slab*. Kandungan dari *slab* tersebut terdiri dari carbon, silica, chromate, phosphor, sulfur, total aluminium, nikel, dan carbon ekuivalen. Jenis *slab* terbagi menjadi 4 jenis, yaitu *ship building quality*, *mild steel*, *high strength* dan boiler A-516.

4.1.2 Visi dan Misi Perusahaan

Berikut adalah visi dan misi PT. Gunawan Dianjaya Steel, Tbk.

1. Visi:

“Menjadi perusahaan terkemuka di Indonesia dalam industry baja, menjadi pilihan utama klien untuk pelat-pelat baja berkualitas tinggi, dan menjadi tolak ukur bagi para kompetitor.”

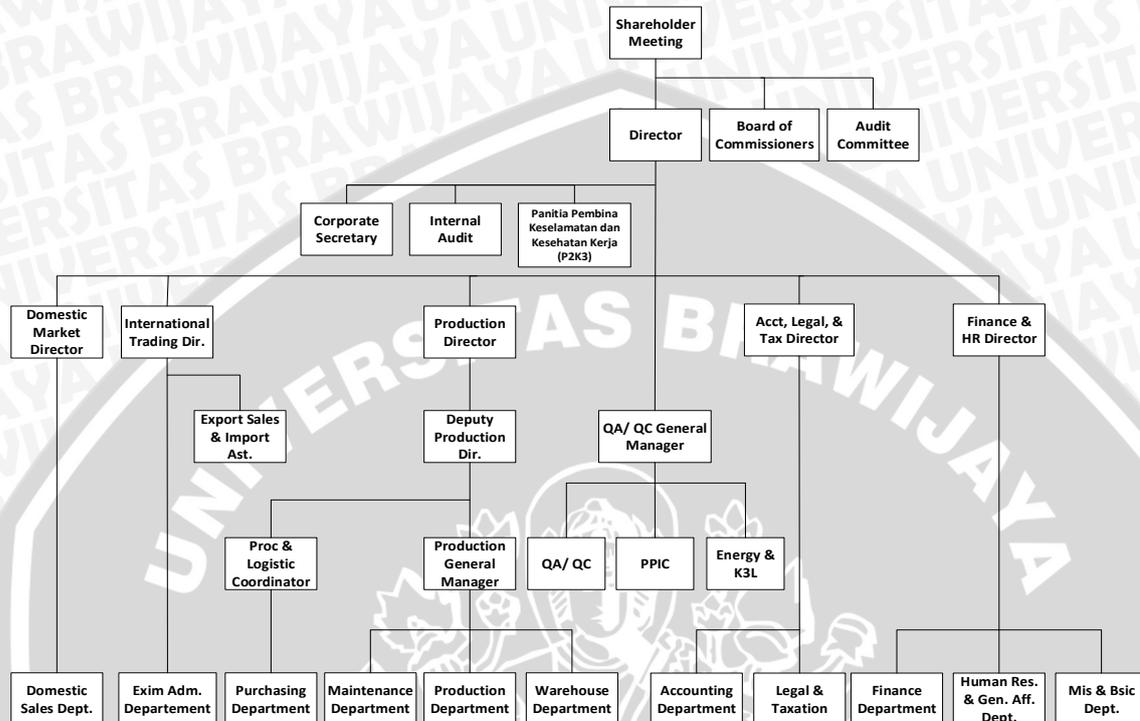
2. Misi:

- Menyediakan pelat baja berkualitas tinggi dengan konsisten dan pelayanan yang terbaik untuk memenuhi kepuasan pelanggan
- Membuat *brand image* sebagai perusahaan yang terbaik di bidang produksi plat baja.
- Beroperasi secara efisien untuk dapat memberikan *profit* bagi pemegang saham dan kesempatan berkembang bagi seluruh karyawan,
- Pengembangan sumber daya manusia, kerja sama yang baik dengan relasi, dan penerapan teknologi maju guna mencapai tujuan perusahaan.

4.1.3 Struktur Organisasi

Struktur organisasi umum PT. GDS pada tahun 2014 dapat dilihat pada gambar 4.2. Pimpinan tertinggi di PT. GDS dipegang oleh seorang direktur utama yang membawahi lima direktur dan satu *general manager*. Direktur yang pertama yaitu direktur *Domestic Sales*, bertanggungjawab terhadap Departemen *Domestic Sales* dan kegiatan yang berhubungan dengan penjualan produk di dalam negeri. Direktur kedua yaitu direktur *International Trading* yang bertanggungjawab pada Departemen *Export-Import Administration* serta kegiatan teknis ekspor-impor. Kemudian ketiga yaitu direktur Produksi yang membawahi *general manager* Produksi dan koordinator Pengadaan dan Logistik. Keempat adalah direktur *Accounting*, Legal, dan *Tax* yang

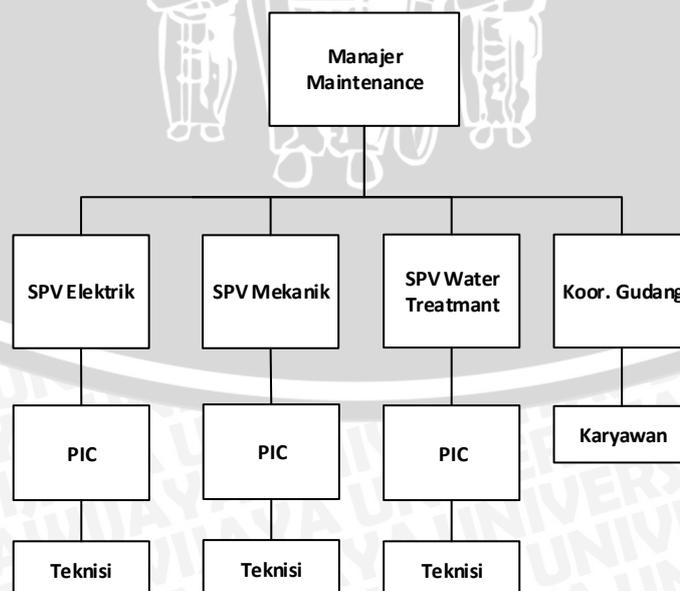
membawahi Departemen *Accounting* dan Departemen *Legal & Tax*. Direktur terakhir yaitu direktur *Finance* dan *Human Resource*. Serta satu general manager yang dibawah langsung oleh direktur utama yaitu *General Manager Quality Control* yang membawahi departemen *Quality Control*, *PPIC*, dan *Energy & K3L*.



Gambar 4.2 Struktur Organisasi GDS

Sumber: PT. Gunawan Dianjaya Steel, Tbk

Penelitian ini dilakukan di departemen *maintenance*, tepatnya di divisi mekanik. Berikut struktur organisasi departemen *maintenance* dapat dilihat pada Gambar 4.3.



Gambar 4.3 Struktur Organisasi Departemen *Maintenance* GDS

Sumber: PT. Gunawan Dianjaya Steel, Tbk

Departemen *maintenance* dipimpin oleh seorang manager yang membawahi 4 divisi yang dipimpin oleh 3 orang yaitu SPV Elektrik, SPV Mekanik, SPV *Water Treatment* serta seorang Koordinator Gudang. SPV Elektrik bertanggung jawab terhadap segala kegiatan perawatan kelistrikan di dalam PT. GDS. Kemudian SPV Mekanik bertanggung terhadap keseluruhan kegiatan perawatan pada mesin yang berhubungan langsung pada kegiatan produksi. SPV *Water Treatment* bertanggung jawab pada kegiatan perawatan mengenai *cooling* sistem serta pelumasan.

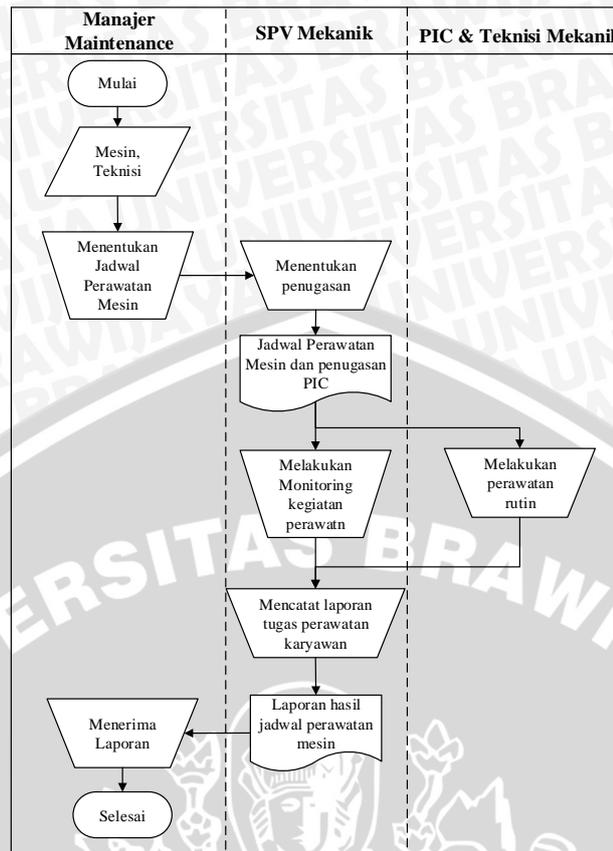
Di bawah SPV mekanik terdapat PIC (*Person in Charge*) yaitu orang yang bertanggung jawab pada setiap kegiatan perawatan. Seorang PIC akan mendapatkan tugas dari SPV dan membuat laporan mengenai kegiatan perawatan yang dia lakukan kepada SPV. Dalam melakukan kegiatan perawatan mesin, PIC dibantu oleh beberapa teknisi mekanik. SPV yang akan menunjuk seseorang yang menjadi PIC serta yang menjadi teknisi pada saat pemberian tugas perawatan.

4.1.4 Sistem Maintenance PT. Gunawan Dianjaya Steel

Sistem manajemen perawatan di departemen *maintenance* PT. GDS dibagi menjadi 2 metode penugasan perawatan mesin. Berikut ini akan dijelaskan bagaimana sistem yang sekarang sedang berjalan pada departemen *maintenance* PT. GDS.

1. Sistem Penugasan *Planned Maintenance*.

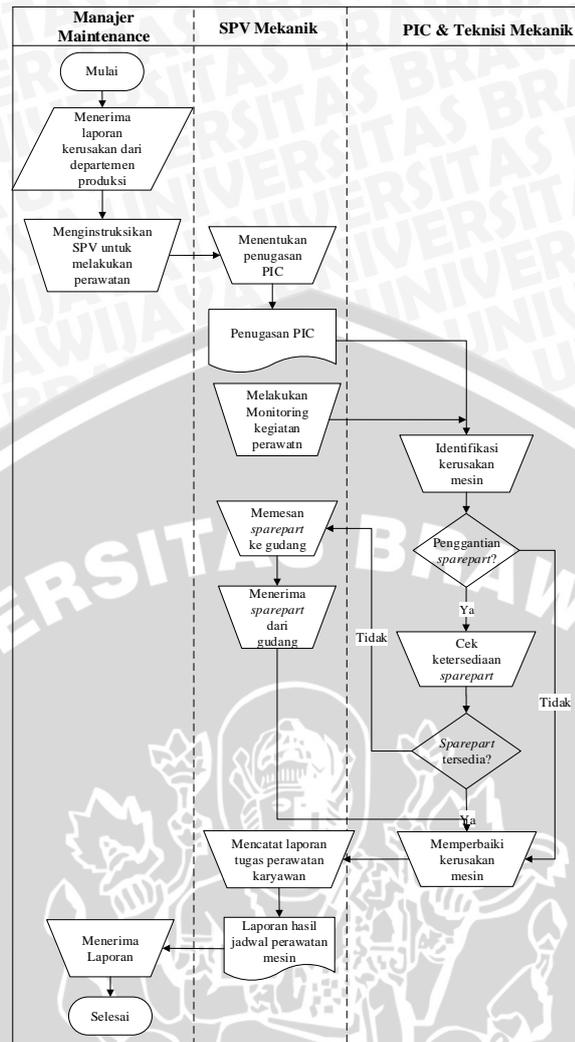
Sistem penugasan ini menggunakan sistem perawatan preventive yang dilakukan setiap hari oleh seluruh teknisi departemen *maintenance*. Selain itu, sistem ini juga dilakukan pada saat mesin mati ketika tidak ada jadwal produksi. Perawatan mesin dilakukan setiap hari berdasarkan instruksi secara lisan dari SPV melalui PIC yang ditunjuk SPV. SPV menentukan jadwal perawatan berdasarkan kebutuhan masing-masing mesin. Pembuatan jadwal perawatan dilakukan pada satu minggu sebelum periode perawatan terlaksana. Apabila perawatan selesai dilakukan, PIC memberikan laporan lisan mengenai tindakan perawatan yang telah dilakukan, dan kemudian SPV mencatat kegiatan perawatan tersebut pada selembar kertas sebagai tanda bahwa perawatan telah dilakukan serta mencatat hasil dari kegiatan perawatan yang telah dilakukan. Pada sore hari sekitar pukul 15.30, SPV membawa lembar kegiatan perawatan tersebut pada manajer *maintenance* untuk dilaporkan kegiatan *planned maintenance* pada hari tersebut kemudian menyimpannya pada *document keeper* di ruang *maintenance*. Berikut ini adalah bagan alir dari sistem penugasan *Planned Maintenance*.



Gambar 4.4 Bagan Alir Analitis Sistem Penugasan *Planned Maintenance*

2. Sistem Penugasan *Unplanned Maintenance*.

Pada sistem penugasan *Unplanned Maintenance* ini, kegiatan perawatan dilakukan pada saat terjadi kerusakan mesin. Operator produksi melaporkan bila mesin mengalami kerusakan kepada teknisi mekanik serta manajer produksi. Setelah manajer produksi memutuskan untuk memberhentikan proses produksi, laporan diteruskan kepada manajer *maintenance*. Manajer *maintenance* memerintahkan SPV Mekanik untuk melakukan tindakan perbaikan. SPV menugaskan PIC beserta teknisi untuk melakukan identifikasi kerusakan mesin yang terjadi. Setelah melakukan identifikasi masalah, jika ditemukan kerusakan yang membutuhkan pergantian *sparepart*, maka teknisi akan memesan *sparepart* melalui SPV ke koordinator gudang. Setelah itu, pihak gudang akan mengirim *sparepart* kepada SPV kemudian memberinya pada PIC. Jika kegiatan perbaikan telah selesai, maka PIC melapor kepada SPV untuk dicatat di kertas untuk kemudian di simpan di *document keeper*. Sistem penugasan *unplanned maintenance* ini dapat digambarkan dengan bagan alir pada Gambar 4.5.



Gambar 4.5 Bagan Alir Analitis Sistem Penugasan *Unplanned Maintenance*

4.1.5 Kebutuhan Data

Dalam melakukan perancangan *prototype* sistem informasi manajemen perawatan ini, dibutuhkan beberapa data yang ada di departemen maintenance PT. GDS. Berikut ini merupakan penjelasan mengenai data yang dibutuhkan untuk menunjang perancangan *prototype* ini.

1. Data Mesin, di departemen *maintenance* PT. GDS, divisi mekanik menangi 29 mesin yang beroperasi di 5 area proses produksi. Pada penelitian ini, akan dibatasi pada 6 mesin yang akan dijadikan objek penelitian. Mesin-mesin tersebut adalah *Roll Table A1 - A2*, *Roll Table 2Q*, *Roll Table 4Q*, *Hot Rolling Mill*, *Hot Leveler*, dan *Reheating Furnace*. Data mesin yang dibutuhkan antara lain, nama mesin, lokasi area mesin, kategori mesin, *Person in Charge*(PIC) mesin, model mesin, nomer seri mesin, fungsi mesin, kontraktor/*supplier* mesin serta keterangan garansi mesin.

2. Data Komponen, merupakan *breakdown* dari mesin. Komponen yang digunakan dalam *prototype* ini adalah bagian-bagian mesin yang terdiri dari beberapa *sparepart*. Data yang dibutuhkan adalah, nama komponen, nama mesin, model komponen, nomer seri komponen, fungsi komponen, kontraktor/*supplier* komponen, serta keterangan garansi komponen,.
3. Data *Sparepart*, merupakan *breakdown* dari komponen. *Sparepart* yang digunakan dalam *prototype* ini adalah bagian terkecil dari mesin yang menjadi suku cadang di departemen maintenance PT. GDS. Data yang dibutuhkan adalah, nama *sparepart*, nama *sparepart*, model *sparepart*, nomer seri *sparepart*, fungsi *sparepart*, kontraktor/*supplier sparepart*, serta *stock inventory sparepart*.
4. Data Karyawan, data yang dibutuhkan adalah, nama karyawan, jabatan karyawan, *password* karyawan, nomer telepon karyawan, alamat, serta *e-mail*.
5. Data *Work Order*, merupakan data penjadwalan dan perawatan mesin yang sudah dan akan berlangsung di PT. GDS. Data yang dibutuhkan adalah, nama pembuat dan penerima *Work Order*, identitas mesin yang di perbaiki, masalah mesin yang terjadi, tindakan perbaikan, tanggal pembuatan dan penugasan, jenis tindakan, serta perbaikan yang dilakukan.

4.2 PENGEMBANGAN *PROTOTYPE*

Berikut ini adalah tahapan dalam pengembangan *prototype* Sistem Informasi Manajemen Perawatan yang akan dirancang. Diagram alir pengembangan *prototype* dapat dilihat pada Gambar 4.6, yang selanjutnya akan dijelaskan.

1. Menetapkan tujuan *prototype*
 - a. Melakukan identifikasi masalah dalam sistem manajemen perawatan mesin yang sedang berjalan di PT. Gunawan Dianjaya Steel.
 - b. Menetapkan batasan-batasan atau ruang lingkup *prototype* dari sistem manajemen perawatan yang akan dirancang.
 - c. Menetapkan tujuan serta manfaat dari *prototype*
Penetapan tujuan serta manfaat dari *prototype* telah dilakukan pada bab I.
2. Mendefinisikan fungsi *prototype*
 - a. *Requirement modelling*
SRC yang dibuat nantinya akan mewakili *user requirement* dari manajer *maintenance*, SPV dan teknisi departemen *maintenance*.
 - b. *Data modelling*

Pada tahap ini, sistem manajemen perawatan akan digambarkan dengan aliran data serta informasi yang dihasilkan dari dan untuk *entity* dalam sistem dengan DFD.

c. *Process modelling*

Menggambarkan logika atau aturan bisnis dalam manajemen perawatan di departemen *maintenance* PT. Gunawan Dianjaya Steel yang dinyatakan dengan tabel aturan.

d. *Development Strategies*

Dalam langkah ini, akan ditentukan kebutuhan minimum *software* maupun *hardawre*, level pembuatan sistem, kebutuhan minimal sistem operasi serta kebutuhan minimal pengguna.

3. Mengembangkan *prototype*

a. Langkah Desain

1) Desain *Database* Logis

Dalam tahap ini akan dibuat *list entity*, kardinalitas serta ERD yang telah dirancang pada tahap *data modelling*.

2) Desain *Database* Fisik

Desain fisik merupakan aktualisasi dari desain logis. DPada langkah ini *entity* sudah berubah menjadi *table* dengan rancangan bentuk fisik.

3) Desain *User Interface* (UI)

Desain *User Interface* meliputi *hierarki menu*, *form* dan *report*.

b. Implementasi, membuat aplikasi yang dirancang dengan melakukan pengembangan *database*, *module* dan *user interface* menggunakan VBA *Microsoft Access* 2013.

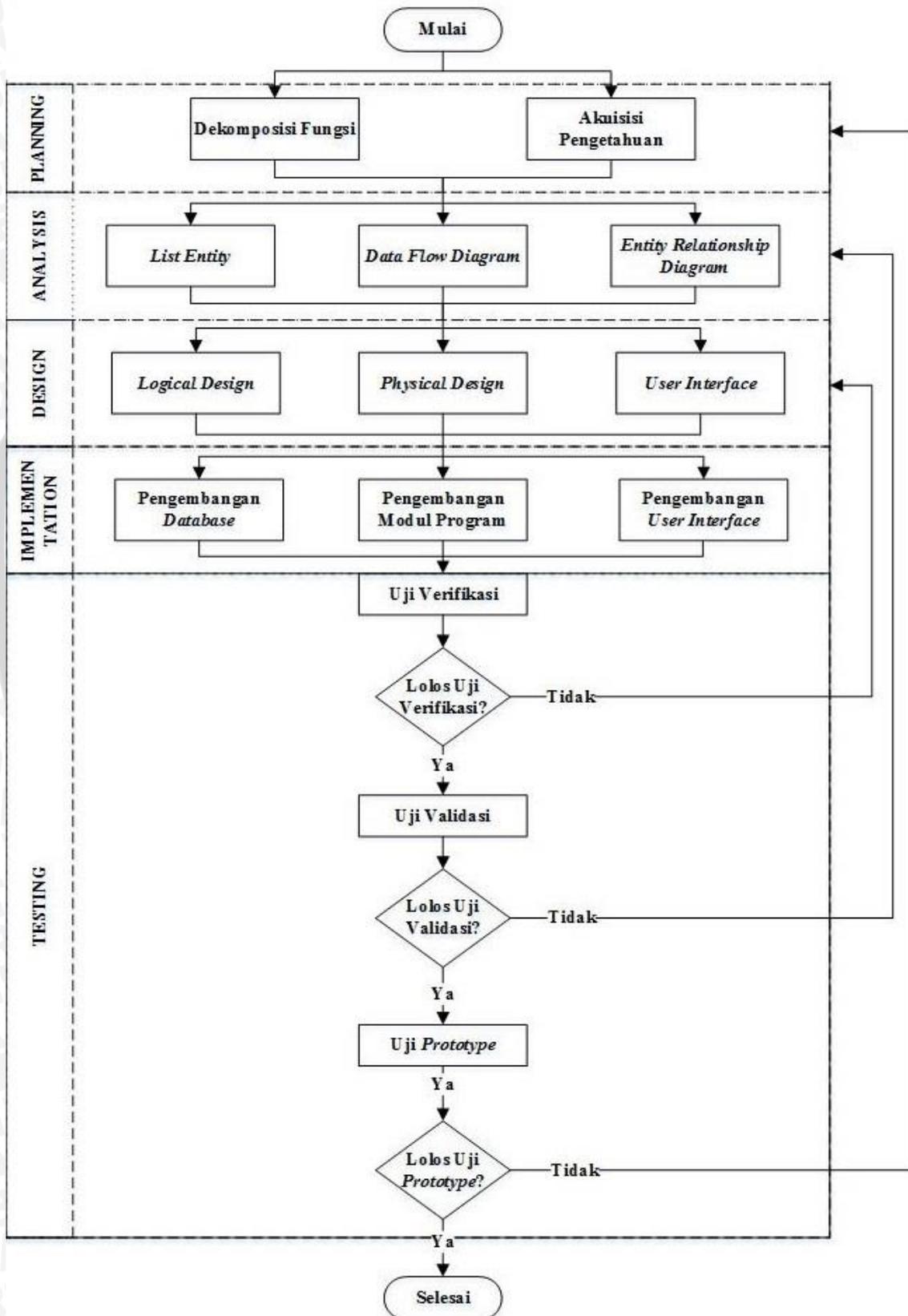
4. Mengevaluasi *prototype*

Pada langkah ini dilakukan pengujian pada program aplikasi yang telah dirancang. Pengujian program ini ditinjau dari tiga segi, yaitu verifikasi, validasi serta uji *prototype*.

a. Verifikasi, dengan membandingkan antara desain *database* fisik, desain *user interface* dan desain alortima yang dirancang dengan implementasi.

b. Validasi, dengan membandingkan antara kebutuhan sistem yang telah terpenuhi dengan SRC.

- c. Uji *prototype* dengan membandingkan *prototype* sistem baru apakah dapat mengatasi kelemahan sistem lama yang dipaparkan sebagai hasil analisis PIECES.



Gambar 4.6 Diagram Alir Perancangan *Prototype* Sistem Informasi

4.3 ANALISIS SISTEM

Pada diagram alir pengembangan *prototype*, langkah awal yang dilakukan adalah menetapkan tujuan *prototype* yang sudah dilakukan pada bab I. Langkah selanjutnya dari proses pengembangan ini adalah mendefinisikan fungsi *prototype* dengan mengalisis data yang ada untuk diterjemahkan menjadi kebutuhan sistem. Terdapat tiga fase yang ada pada tahap analisis sistem yaitu membuat model kebutuhan sistem (*requirement modelling*), model data (*data modelling*) dan model proses (*process modelling*). Sebelum memasuki tahap analisis sistem, berikut ini merupakan analisis data dari sistem manajemen perawatan yang sedang berjalan pada PT. GDS.

Pada sistem manajemen perawatan yang sekarang sedang berjalan pada PT. GDS yang telah dijelaskan masih memiliki beberapa kelemahan. Kelemahan sistem yang diterapkan oleh PT. GDS telah dijelaskan pada Tabel 1.2 dengan melakukan analisa PIECES pada kelemahan sistem lama. Berdasarkan bagan alir pada Gambar 4.4 dan Gambar 4.5, SPV melakukan pencatatan laporan kegiatan perawatan dan mengelola laporan itu sendiri. Seharusnya SPV yang merupakan pihak manajerial yang seharusnya menerima sebuah laporan. Selain itu, sistem manual yang diterapkan berdampak pada seringnya terjadi keterlambatan jadwal perawatan mesin akibat dari SPV yang kesulitan dalam mengontrol tindakan perawatan mesin yang belum ataupun yang sudah dilakukan oleh teknisi.

Dalam kegiatan *planned maintenance*, SPV tidak memiliki pedoman khusus dalam melaksanakan kegiatan perawatan. Hanya menggunakan urutan perawatan *preventive* dari mesin satu ke mesin lainnya secara bergantian. Dalam melakukan *preventive maintenance* seharusnya juga memperhatikan kondisi mesin. Tetapi dikarenakan di departemen *maintenance* tidak melakukan pengukuran peformansi mesin, SPV hanya melakukan pengecekan visual melalui pendengaran maupun penglihatan serta untuk menentukan mesin tersebut memerlukan perawatan *preventive* atau tidak. Dampaknya adalah beberapa mesin menjadi sering mengalami *downtime* serta terjadi *speed losses* di beberapa mesin.

Pada sistem penugasan *unplanned maintenance*, ketika mesin mengalami kerusakan dan membutuhkan pergantian *sparepart*, teknisi harus mencari *sparepart* yang dibutuhkan tanpa mengetahui lokasi *sparepart* serta stock *sparepart*. Apabila *sparepart* yang dibutuhkan tidak ada, maka teknisi melaporkan kondisi tersebut ke SPV agar SPV dapat melakukan pemesanan terhadap gudang pusat. Hal ini yang menyebabkan waktu penyelesaian perbaikan kerusakan mesin menjadi lebih lama. Setelah kegiatan perbaikan

selesai, teknisi melaporkan secara lisan mengenai kegiatan perawatan yang dilakukan kepada SPV untuk dicatat, aliran dan pengolahan data seperti ini tidak efisien.

Melihat kondisi sistem manajemen perawatan seperti yang telah dijelaskan, maka dibutuhkan sistem informasi manajemen perawatan mesin yang mampu memudahkan teknisi, SPV, maupun manajer *maintenance* dalam proses pelaporan, penentuan jadwal dan penugasan kegiatan *maintenance*, serta dalam pencarian *sparepart*. Dalam dunia perawatan, dikenal *Computerized Maintenance Management System (CMMS)* yang merupakan aplikasi perangkat lunak yang mengatur *database* berisi informasi aktivitas pemeliharaan dalam suatu perusahaan. Selain itu, diperlukan pengukuran peformansi mesin untuk memberikan informasi mengenai peformansi mesin. Informasi-informasi ini digunakan untuk membantu para manajer hingga teknisi dalam melaksanakan tugas pemeliharaan aset agar lebih efektif, seperti misalnya menentukan mesin mana yang sedang memerlukan perawatan dan gudang mana saja yang menyediakan *sparepart* yang akan digunakan dalam proses perawatan tersebut.

4.3.1 Requirement Modelling

Tahap *requirement modeling* ini merupakan suatu langkah yang bertujuan untuk mengidentifikasi dan menjelaskan seluruh persyaratan sistem. Persyaratan sistem merupakan karakteristik atau fitur harus ada dalam sistem informasi dalam rangka memenuhi kebutuhan *user* atau pengguna. Kebutuhan tersebut kemudian dinyatakan dalam suatu *checklist* yang disebut SRC (*System Requirement Checklist*). SRC adalah fitur-fitur atau karakteristik yang harus ada dalam sistem informasi untuk memenuhi kebutuhan bisnis (*business requirements*) dan yang dapat diterima pengguna. Nantinya SRC akan berperan sebagai acuan dalam mengukur keberhasilan sistem informasi yang dibangun. Model kebutuhan sistem ini digambarkan ke dalam lima kategori umum yaitu *output*, *input*, *process*, *performance*, dan *control*.

Sistem informasi yang akan dirancang diperuntukkan untuk Teknisi, SPV, dan manajer *maintenance*. SRC yang dibuat adalah SRC dari masing-masing pengguna aplikasi sistem informasi manajemen perawatan mesin ini yaitu SRC dari sisi teknisi, SRC dari sisi SPV, dan SRC dari sisi manajer *maintenance*. Berikut pada Tabel 4.2 merupakan SRC dari SPV serta manajer *maintenance*.

Tabel 4.1 Sistem *Requirement Checklist* untuk Manajer dan SPV

| Kategori | Penjabaran |
|--------------------|--|
| Input | <p>Berikut ini adalah penjabaran dari kategori <i>input</i> yang dibutuhkan oleh manajer <i>maintenance</i> dalam sistem informasi yang akan dirancang.</p> <ol style="list-style-type: none"> 1. Data mesin: Nama mesin, merk mesin, tahun dan fungsi 2. Data Komponen: Nama komponen mesin, merk dan tahun 3. Data <i>Sparepart</i>: Nama <i>sparepart</i>, merk, tahun, <i>stock sparepart</i>, jumlah minimum, letak <i>sparepart</i>, jumlah <i>sparepart</i> yang masuk dan keluar 4. Data Karyawan: Nama karyawan, alamat, nomor telepon, tanggal lahir, , institusi |
| | <p>SPV Mekanik dapat memasukkan data yang berhubungan dengan penugasan karyawan untuk melakukan tindakan <i>maintenance</i>, antara lain:</p> <ol style="list-style-type: none"> 1. Jadwal perawatan mesin <i>unplanned & planned maintenance</i> : Tanggal perawatan, komponen mesin, nama karyawan yang bertugas, nama tindakan perawatan, keterangan. 2. Data mesin: Nama mesin, merk mesin, tahun dan fungsi 3. Data Komponen: Nama komponen mesin, merk dan tahun 4. Data <i>Sparepart</i>: Nama <i>sparepart</i>, merk, tahun, <i>stock sparepart</i>, jumlah minimum, letak <i>sparepart</i>, jumlah <i>sparepart</i> yang masuk dan keluar 5. Data performansi mesin : Down time, breakdown, jumlah produk defect, jumlah produk yang dihasilkan, kapasitas produksi. |
| Output | <p>Sistem dapat memberikan laporan yang dibutuhkan, antara lain:</p> <ol style="list-style-type: none"> 1. Laporan harian: laporan perawatan mesin yang harus dan telah dilakukan serta laporan <i>inventory sparepart</i>. 2. Laporan rutin bulanan: <i>summary</i> jadwal perawatan mesin dan <i>summary inventory sparepart</i> 3. Laporan 1 periode produksi : laporan mengenai peformansi mesin yang akan dihitung setiap periode produksi |
| | <p>Sistem dapat memberikan <i>form hardcopy</i> untuk laporan segala kegiatan <i>planned</i> dan <i>unplanned maintenance</i>.</p> |
| | <p>Sistem dapat memberikan <i>alert</i> mengenai jadwal perawatan mesin harian yang harus dilakukan.</p> |
| | <p>Sistem dapat memberikan informasi performansi mesin untuk rekomendasi prioritas tindakan perawatan</p> |
| Process | <p>Sistem dapat memonitoring seluruh kegiatan perawatan di departemen <i>maintenance</i> mulai dari status pengerjaan perawatan, status <i>sparepart</i>, histori mesin hingga kondisi performansi mesin,</p> |
| | <p>Sistem dapat memberikan informasi penugasan perawatan mesin yang harus dilakukan oleh teknisi <i>maintenance</i>.</p> |
| | <p>Sistem dapat meberikan informasi tindakan perawatan yang harus dilakukan pada setiap waktu secara realtime.</p> |
| Performance | <p>Sistem dapat dioperasikan setiap saat tidak terbatas oleh waktu.</p> |
| | <p>Sistem memungkinkan dalam melakukan kontrol terhadap jadwal perawatan mesin dan stok <i>sparepart</i>.</p> <p>Sistem dapat memberi informasi mengenai kegiatan perawatan yang harus dilakukan pada waktu itu.</p> |
| Control | <p>Manajer <i>maintenance</i> dan SPV Mekanik akan di beri hak akses <i>login</i> sesuai dengan kebutuhan dari masing-masing.</p> |

Dari Tabel 4.1, kebutuhan sistem yang paling dibutuhkan oleh manajer *maintenance* adalah sistem ini dapat memonitoring seluruh kegiatan perawatan di departemen *maintenance* mulai dari status pengerjaan perawatan, status *sparepart*, hingga kondisi peformansi mesin. Sedang kebutuhan sistem yang dibutuhkan oleh SPV adalah dapat memasukkan data yang berhubungan dengan penugasan teknisi untuk melakukan tindakan *maintenance* serta dapat memberikan informasi tindakan perawatan yang harus dilakukan pada setiap waktu secara *realtime* agar tidak terjadi keterlambatan. Manajer *maintenance* dan SPV juga memerlukan sistem yang dapat dioperasikan setiap saat tidak terbatas oleh waktu dan dapat melakukan kontrol terhadap jadwal perawatan mesin dan stok *sparepart*. Sedangkan SRC kebutuhan dari teknisi *maintenance* adalah sebagai berikut.

Tabel 4.2 Sistem Requirement Checklist untuk Teknisi

| Kategori | Penjabaran |
|--------------------|--|
| Input | Teknisi dapat memasukkan data: <ol style="list-style-type: none"> 1. Data mesin: Nama mesin, merk mesin, tahun dan fungsi 2. Data Komponen: Nama komponen mesin, merk dan tahun 3. Data <i>Sparepart</i>: Nama <i>sparepart</i>, merk, tahun, <i>stock sparepart</i>, jumlah minimum, letak <i>sparepart</i>, jumlah <i>sparepart</i> yang masuk dan keluar |
| Output | Sistem dapat memberikan laporan pencarian dadakan tentang jadwal perawatan mesin yang harus dilakukan oleh teknisi setiap hari, data jumlah <i>sparepart</i> , letak <i>sparepart</i> dan <i>history</i> mesin. |
| Process | Sistem dapat memberikan informasi penugasan perawatan mesin yang harus dilakukan oleh teknisi <i>maintenance</i> . |
| | Sistem dapat meberikan informasi tindakan perawatan yang harus dilakukan pada setiap waktu secara <i>realtime</i> . |
| | Sistem dapat mencari data letak <i>sparepart</i> , <i>inventory sparepart</i> dan <i>history</i> mesin. |
| Performance | Sistem dapat selalu <i>support</i> pada karyawan <i>maintenance</i> . |
| | Sistem dapat beroperasi 7 hari seminggu, 365 hari setahun. |
| | Sitem dapat menampilkan jadwal perawatan yang harus dilakukan setiap hari. |
| | Sistem dapat memberi peringatan mengenai kegiatan perawatan yang harus dilakukan pada waktu itu. |
| | Sistem dapat menampilkan data teknisi lain, jumlah <i>sparepart</i> , letak <i>sparepart</i> dan <i>history</i> mesin yang dicari. |
| Control | Teknisi diberi <i>username</i> dan password multiuser yang dapat digunakan oleh seluruh teknisi. |

Berdasarkan Tabel 4.2 di atas, kebutuhan sistem yang paling dibutuhkan oleh karyawan *maintenance* adalah dapat informasi penugasan serta tindakan perawatan mesin yang harus dilakukan oleh karyawan *maintenance*. Selain itu, untuk menunjang kegiatan perawatan mesin, karyawan juga membutuhkan informasi mengenai letak *sparepart*, *inventory sparepart* dan *history* mesin. Sistem diharapkan mampu selalu mensupport

karyawan dengan beroperasi 7 hari seminggu, 365 hari setahun. Sistem *requirement checklist* yang telah dirumuskan di Tabel 4.2 dan Tabel 4.3 selanjutnya akan digunakan untuk menentukan spesifikasi sistem yang akan dibuat. Berikut ini adalah spesifikasi kebutuhan sistem informasi manajemen perawatan mesin secara keseluruhan:

1. Sistem dapat mengontrol tindakan perawatan yang dilakukan dengan memberikan peringatan terhadap tindakan perawatan yang harus dilakukan.
2. Sistem dapat diakses oleh manajer, SPV, dan karyawan dengan hak akses yang berbeda sesuai dengan kebutuhan dengan memasukkan *username* dan *password*. Manajer dan SPV mempunyai hak untuk mengubah data dan menambah data, sedangkan karyawan tidak dapat mengubah data dan menambahkan data.
3. Sistem dapat melakukan kontrol terhadap penugasan perawatan serta *inventory sparepart*.
4. Sistem dapat memonitoring seluruh kegiatan perawatan di departemen *maintenance* mulai dari status pengerjaan perawatan, status *sparepart*, hingga kondisi peformansi mesin.
5. Sistem dapat menginformasikan tindakan perawatan yang harus dilakukan oleh teknisi setiap hari dan memberikan informasi *inventory sparepart*, lokasi *sparepart* dan *history* perawatan mesin yang dibutuhkan oleh teknisi.

4.3.2 Data and Process Modelling

Pada tahap ini analisis sistem akan melakukan pengembangan model grafik untuk menunjukkan bagaimana sistem dalam mengubah data menjadi informasi dengan data modelling. Sedangkan logika proses bisnis yang terdapat pada departemen *maintenance* PT. GDS digambarkan menggunakan flowcart sistem dalam process modelling. Produk akhir dari pemodelan data dan proses adalah model logis yang akan mendukung operasi bisnis dan dapat memenuhi kebutuhan pengguna.

4.3.2.1 Data Modelling

Pada tahap ini, sistem manajemen perawatan mesin akan digambarkan melalui *Data Flow Diagram* (DFD) yang digunakan untuk menunjukkan logika dari kebutuhan-kebutuhan sistem yaitu proses yang dibutuhkan sistem dan pertukaran data dalam sistem. Langkah-langkah dalam pembuatan DFD adalah sebagai berikut:

1. Identifikasi kesatuan luar (*external entities*) yang terkait, dalam perancangan sistem ini kesatuan luar yang terakit adalah:

- a. Manajer *maintenance*
 - b. SPV Mekanik
 - c. Teknisi *maintenance*
2. Identifikasi semua *input* output dari kesatuan luar yang terkait dalam sistem ini.

Berikut ini adalah *input-output* yang terkait dalam alur sistem:

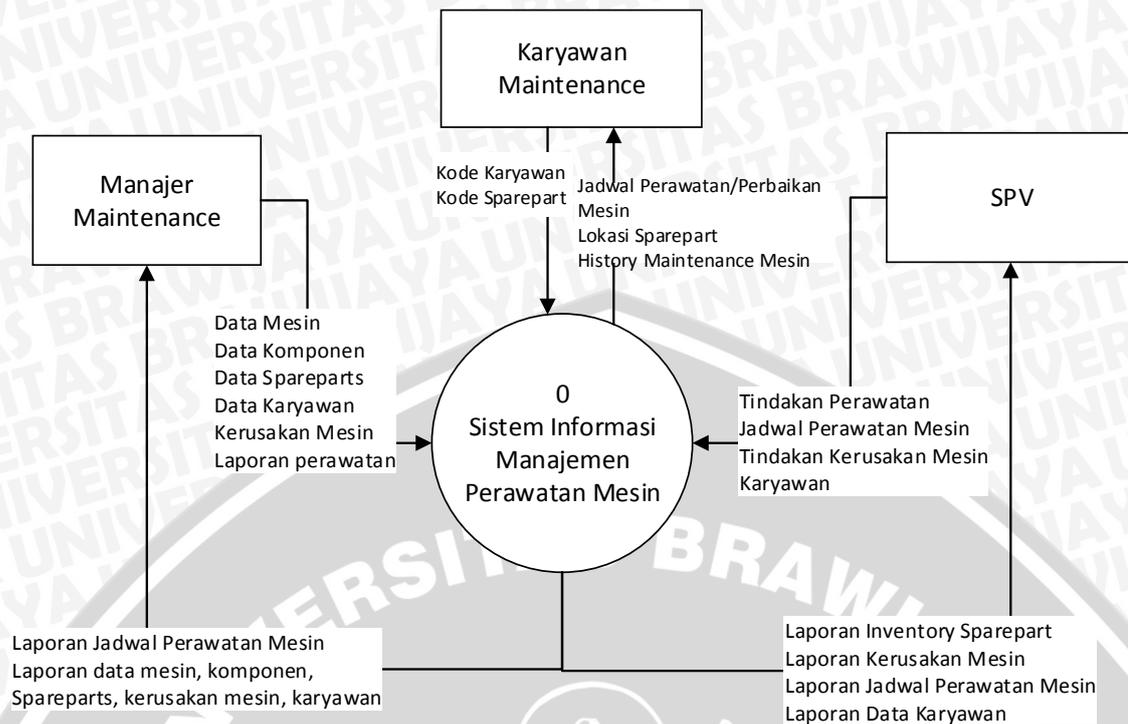
Tabel 4.3 Identifikasi *Input* dan *Output* DFD

| Kesatuan Luar | <i>Input</i> | <i>Output</i> |
|-----------------------------|--|---|
| Manajer <i>Maintenance</i> | Data mesin, data komponen, data <i>sparepart</i> , data kerusakan mesin, data karyawan, laporan perawatan mesin oleh karyawan, dan laporan perbaikan mesin karyawan. | Laporan data mesin, komponen, <i>sparepart</i> , karyawan, kerusakan mesin, laporan data karyawan dan laporan jadwal perawatan mesin. |
| SPV Mekanik | Data tindakan perawatan mesin, data jadwal perawatan mesin dan data penugasan untuk kerusakan mesin | Laporan jadwal perawatan mesin, laporan kerusakan mesin dan laporan <i>inventory sparepart</i> . |
| Karyawan <i>maintenance</i> | Kode karyawan, nama mesin dan kode <i>sparepart</i> . | Jadwal perawatan mesin, penugasan perbaikan kerusakan mesin, <i>history</i> perawatan dan kerusakan mesin serta lokasi <i>sparepart</i> . |

Berdasarkan Tabel 4.3, dapat diketahui bahwa terdapat 3 entitas yang terlibat dalam sistem informasi manajemen perawatan. Ketiga entitas itu adalah manajer *maintenance*, SPV, dan teknisi. *Input* dan *output* pada ketiga entitas tersebut berbeda sesuai dengan kebutuhan sistem yang telah dirancang.

3. *Context Diagram*

Context Diagram merupakan diagram yang merepresentasikan proses ruang lingkup sistem dengan menunjukkan proses yang akan terjadi pada sistem. *Context Diagram* adalah level teratas dari aliran data dalam sistem yang dikembangkan yang menggambarkan hubungan *input-output* antara sistem dengan dunia luarnya (kesatuan luar). Berikut *Context Diagram* dari sistem informasi manajemen perawatan mesin pada Gambar 4.7.



Gambar 4.7 Context Diagram Sistem Informasi Manajemen Perawatan Mesin

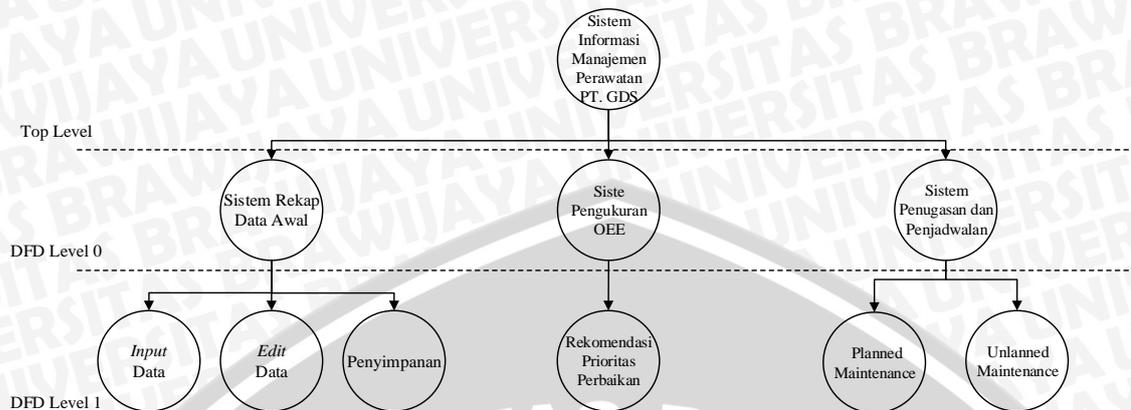
Berdasarkan Gambar 4.7, proses pada *Context Diagram* sistem informasi manajemen perawatan adalah sebagai berikut.

- 1) Manajer *maintenance* dapat memasukkan data mesin, data komponen, data *sparepart*, serta data karyawan. Manajer *maintenance* juga dapat melihat status kegiatan *maintenance* yang telah atau akan dilakukan serta data histori mesin.
- 2) SPV dapat memasukkan data penugasan karyawan, jadwal *planned maintenance*, jadwal *unplanned maintenance*, serta jadwal kegiatan *planned maintenance*. SPV juga dapat memasukkan kondisi performansi mesin untuk setiap periode produksi. Setelah itu sistem informasi mengolah data kondisi tersebut, SPV dapat melihat sejauh mana performansi mesin tersebut.
- 3) Sistem Informasi akan mengolah data perawatan yang telah dilakukan oleh SPV untuk nantinya dapat menjadi informasi bagi manajer *maintenance* maupun teknisi.
- 4) Teknisi mampu mendapatkan informasi mengenai penugasan kegiatan perawatan yang harus dilakukan, data mengenai mesin, komponen, dan *sparepart*.

4. Hierarchy Chart

Hierarchy chart ini bertujuan untuk mempersiapkan penggambaran *Data Flow Diagram* ke level yang lebih awal. *Hierarchy chart* digambarkan lebih detail pada DFD

level 0 dan DFD level 1 nanti. Berikut ini adalah *hierarchy chart* dari sistem informasi yang dirancang.

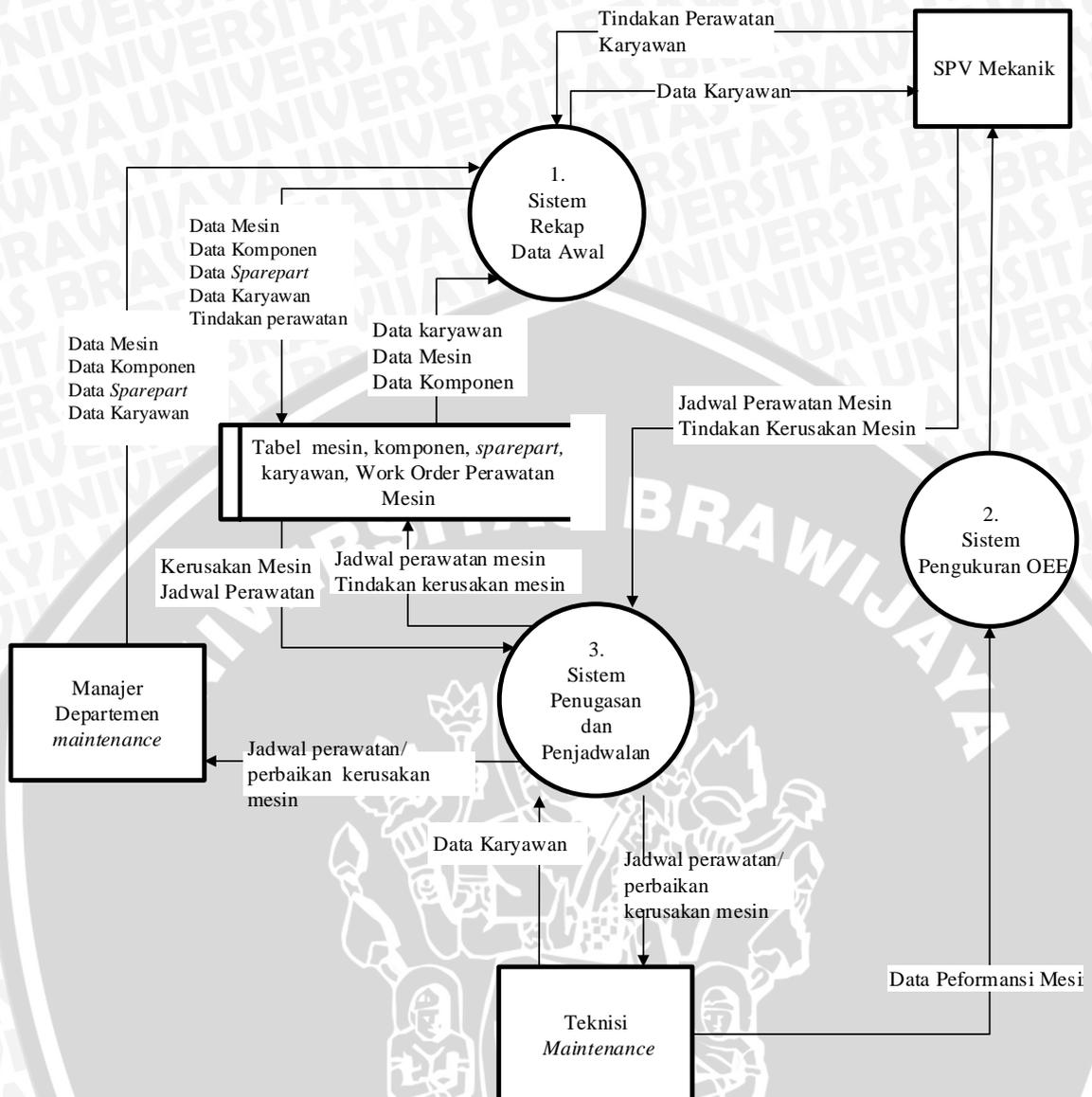


Gambar 4.8 Hierarchy Chart DFD

Berdasarkan Gambar 4.8, proses yang telah diidentifikasi pada *Context Diagram* akan dibagi menjadi 3 proses utama pada DFD level 0. Ketiga proses tersebut adalah sistem rekap data awal, serta sistem penugasan dan penjadwalan. Nantinya pada DFD level 0, kedua proses utama akan dibagi lagi menjadi proses yang lebih detail. Untuk sistem pengolahan data awal, proses akan terbagi menjadi *input* data, edit data, serta penyimpanan. Sistem pengukuran OEE nantinya memberikan rekomendasi prioritas perbaikan mesin pada *Planned Maintenance*. Sedangkan untuk sistem penugasan dan penjadwalan terdapat penugasan kegiatan *unplanned maintenance* serta *planned maintenance*.

5. DFD Level 0

Pada DFD Level 0, akan ditunjukkan proses internal yang menyusun proses utama dalam *Context Diagram* sekaligus bagaimana informasi berpindah dari satu proses ke proses yang lainnya. Dari *Context Diagram* yang sudah dibahas sebelumnya terdapat 3 proses utama, yaitu sistem pengolahan data awal, sistem pengukuran OEE, serta sistem penugasan dan penjadwalan perawatan. Proses yang terjadi pada DFD level 0 sistem informasi manajemen perawatan akan dijelaskan pada gambar berikut ini.



Gambar 4.9 DFD Level 0 Sistem Informasi Manajemen Perawatan Mesin

Berdasar Gambar 4.9, terdapat 3 proses utama yang ada yaitu sistem pengolah data awal, sistem penugasan dan penjadwalan perawatan serta sistem pengukuran OEE.

Proses-proses pada DFD level 0 tersebut dapat dijelaskan sebagai berikut:

a. Proses 1. Sistem Rekap Data Awal

Pada proses ini, manajer akan memasukkan data-data awal seperti data mesin, komponen, *sparepart*, kerusakan mesin, dan data karyawan yang nanti tersimpan dalam tabel. Sehingga nantinya data tersebut diolah oleh sistem informasi agar terintegrasi ketika dibutuhkan SPV untuk membuat penugasan perawatan.

b. Proses 2. Sistem Perhitungan OEE

Proses ini mengolah data dari teknisi mengenai data peformansi mesin hingga menghasilkan nilai OEE sebagai rekomendasi prioritas perbaikan. Dari nilai OEE itu, SPV akan mendapat pertimbangan dalam rangka menentukan tindakan *planned maintenance*.

c. Proses 3. Sistem Penugasan dan Penjadwalan Perawatan

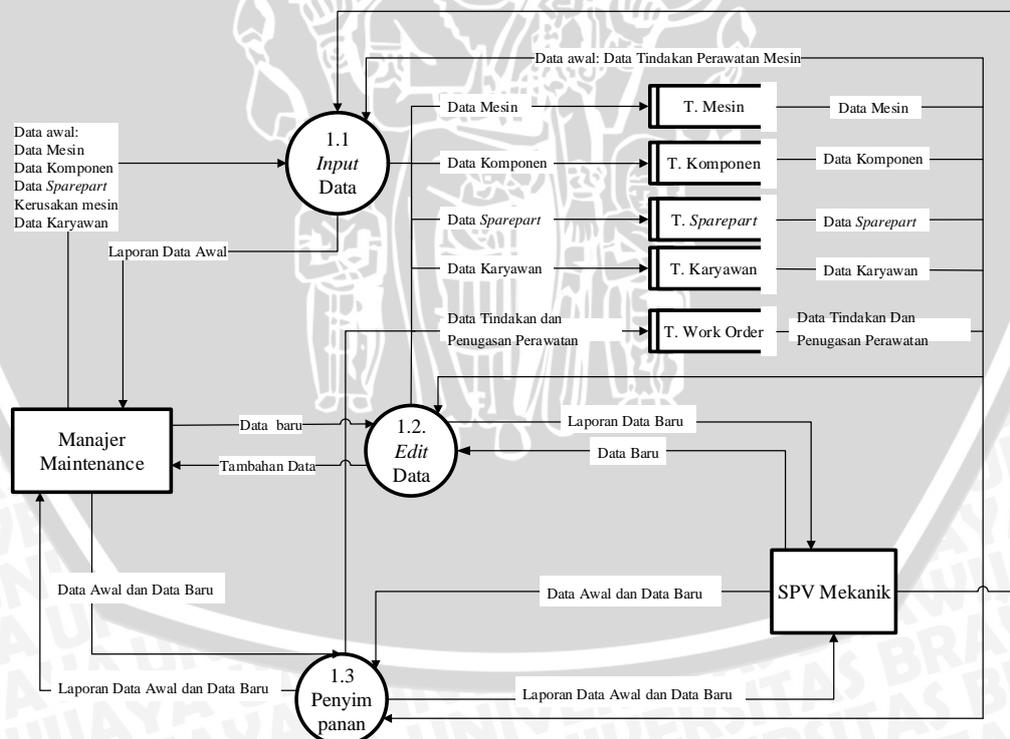
Pada proses ketiga ini, SPV akan memasukan data jadwal perawatan mesin beserta penugasan yang akan di simpan pada tabel perawatan mesin. Teknisi selaku PIC penugasan tersebut dapat melihat penugasan yang diberikan kepadanya beserta teknisi yang membantu.

6. DFD Level 1

Pada DFD Level 1 ini, akan digambarkan lebih detail mengenai masing-masing proses.

a. Proses 1. Sistem Rekap Data Awal

Pada level 1 ini, proses 1 yang terdapat pada DFD level 0 akan dipecah menjadi 3 bagian proses yaitu *input* data, edit data serta penyimpanan. Untuk lebih jelasnya dapat dilihat pada Gambar 4.10.



Gambar 4.10 DFD Level 1 Proses 1 Sistem Pengolahan Data Awal

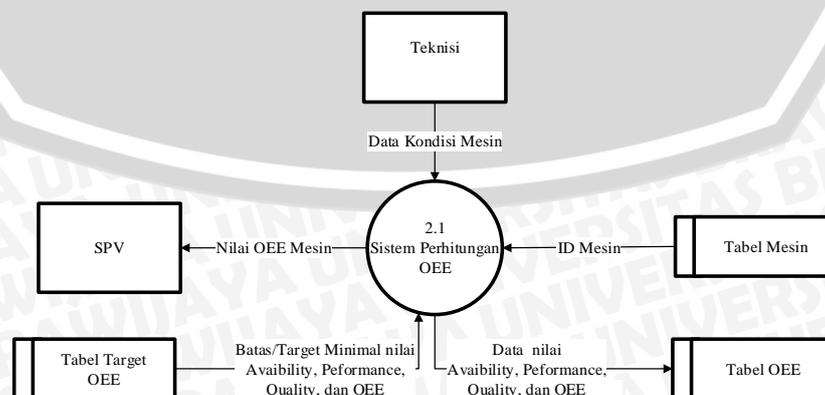
Berdasarkan Gambar 4.10, pada proses 1 sistem rekap data awal menggambarkan aliran informasi dalam pengolahan data pada sistem informasi

manajemen perawatan. Terdapat proses *input* data, edit data dan penyimpanan yang terjadi dalam DFD level 1 proses 1. Proses-proses tersebut dapat dijelaskan sebagai berikut:

- 1) *Input Data*, merupakan proses dimana manajer dan SPV dapat memasukan data awal ke dalam sistem. Kemudian sistem akan menyimpan data tersebut ke dalam masing-masing tabel. Nantinya data-data yang diolah oleh sistem informasi manajemen perawatan tersebut dapat di akses oleh manajer, SPV, maupun teknisi departemen *maintenance* sehingga mampu memberikan informasi yang dibutuhkan.
- 2) *Edit Data*, pada proses ini manajer maupun SPV dapat merubah data awal yang telah dimasukan bila ada perubahan. Selanjutnya perubahan data tersebut akan tersimpan di masing-masing tabel. Nantinya data-data tersebut diolah oleh sistem informasi manajemen perawatan tersebut dapat di akses oleh manajer, SPV, maupun teknisi departemen *maintenance* sehingga mampu memberikan informasi yang dibutuhkan.
- 3) *Penyimpanan*, merupakan proses dimana data awal dan data baru yang dimasukkan baik oleh manajer maupun SPV dan disimpan, sistem kemudian menyimpan data-data ini pada tabel masing-masing. Selanjutnya sistem mengolah data tersebut untuk ditampilkan dalam bentuk informasi kepada manajer, SPV maupun teknisi.

b. Proses 2. Sistem Pengukuran OEE

Pada proses ini, terdapat proses utama yakni perhitungan peformansi mesin menggunakan *Overall Equipment Effectiveness* (OEE) dalam memberikan rekomendasi prioritas pada saat *planned maintenance*. Berikut DFD level 1 Proses 2 akan digambarkan pada Gambar 4.11.

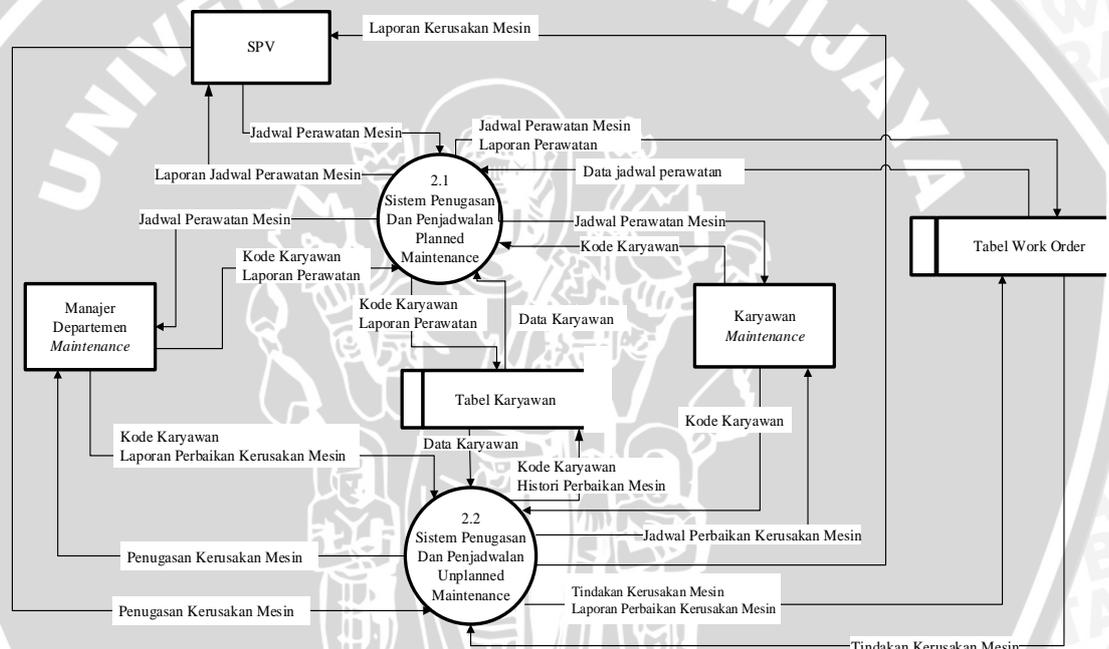


Gambar 4.11 DFD Level 1 Proses 1 Sistem Pengukuran OEE

Proses ini bertujuan untuk mendapatkan rekomendasi prioritas perawatan mesin bagi SPV. Untuk mendapatkan nilai OEE tersebut, dibutuhkan *input* data kondisi mesin dari Teknisi selaku PIC mesin tersebut. Kemudian data yang dimasukan akan di olah oleh Sistem Perhitungan OEE hingga menghasilkan output berupa nilai OEE. Dari nilai tersebut akan dibandingkan dengan batas target minimal nilai yang ditentukan oleh perusahaan sebelumnya.

c. Proses 3. Sistem Penugasan dan Penjadwalan Perawatan

Pada level ini, proses 3 yang ada pada DFD level 0 sebelumnya, dipecah menjadi 2 proses yaitu sistem Penugasan & Perawatan Mesin *Planned Maintenance* serta Penugasan & Perawatan Mesin *Unplanned Maintenance*. Gambar berikut merupakan DFD level 1 dari proses 2 ini.



Gambar 4.12 DFD Level 1 Proses 1 Sistem Penugasan dan Penjadwalan Perawatan

Berdasarkan Gambar 4.12, sistem penugasan dan penjadwalan dibagi menjadi 2 yaitu sistem penugasan dan penjadwalan *planned maintenance* serta sistem penugasan dan penjadwalan *unplanned maintenance*. kedua sistem tersebut dapat dijelaskan sebagai berikut:

- 1) Sistem Penugasan Dan Penjadwalan *Planned Maintenance*, SPV akan memasukan data berupa penugasan preventive *maintenance* ke dalam sistem informasi untuk kemudian disimpan pada tabel perawatan mesin. Selanjutnya, teknisi selaku PIC dapat mengakses jadwal serta penugasan tersebut dengan mencari berdasarkan nama atau kode karyawan. Setelah

kegiatan perawatan selesai, PIC melakukan update terhadap data kegiatan perawatan tersebut. Sehingga SPV dapat menerima informasi mengenai status kegiatan perawatan tersebut.

- 2) Sistem Penugasan Dan Penjadwalan *Unplanned Maintenance*, SPV akan memasukan data berupa penugasan breakdown *maintenance* dimana penugasan dilakukan ketika ada kerusakan pada mesin. Teknisi dapat mengakses penugasan tersebut dengan mencari berdasarkan data teknisi. Setelah kegiatan perawatan itu selesai, teknisi yang bertugas sebagai PIC update terhadap data kegiatan perawatan tersebut. Sehingga SPV dapat menerima informasi mengenai status kegiatan perawatan tersebut.

4.3.2.2 Process Modelling

Process modelling akan mendeskripsikan rincian fungsional, dan merupakan satu set spesifik logika bisnis serta langkah-langkah pengolahan. Logika proses bisnis merupakan langkah untuk menggambarkan proses bisnis atau aturan bisnis yang berlaku pada departemen *maintenance* PT. GDS. Tabel berikut ini akan menjelaskan logika proses bisnis serta aturan yang ada pada departemen *maintenance* PT. GDS dalam menjalankan sistem informasi manajemen perawatan mesin.

Tabel 4.4 Logika Proses Bisnis dan Aturan Pada Departemen *Maintenance*

| Proses | Aturan Proses |
|--|--|
| Penugasan & Perawatan Mesin <i>Planned Maintenance</i> | <ol style="list-style-type: none"> 1. Jadwal perawatan mesin ditentukan 1 minggu sebelum periode perawatan oleh SPV dengan melihat laporan perawatan mesin yang telah dilakukan karyawan dalam 1 minggu sebelumnya serta perhitungan performansi mesin. 2. Jadwal perawatan mesin dilakukan secara rutin sesuai dengan jadwal yang ditentukan. 3. SPV akan menunjuk PIC serta teknisi yang untuk melakukan kegiatan perawatan tersebut. 4. Teknisi departemen <i>maintenance</i> harus melakukan perawatan mesin sesuai jadwal. 5. PIC yang ditugaskan harus melaporkan tugas setelah menyelesaikan tindakan perawatan mesin melalui sistem informasi perawatan yang dirancang. |
| Penugasan & Perawatan Mesin <i>Unplanned Maintenance</i> | <ol style="list-style-type: none"> 1. Penugasan kegiatan perawatan dilakukan oleh SPV. 2. SPV akan menunjuk PIC serta teknisi yang untuk melakukan kegiatan perawatan tersebut. 3. PIC yang ditugaskan harus melaporkan tugas setelah menyelesaikan tindakan perawatan mesin melalui sistem informasi perawatan yang dirancang. |
| <i>Inventory Sparepart</i> | <ol style="list-style-type: none"> 1. Jumlah stok <i>sparepart</i> harus melebihi jumlah minimal sehingga ketersediaan <i>sparepart</i> selalu ada . 2. SPV melakukan pemesanan <i>sparepart</i> ke bagian gudang ketika jumlah stok <i>sparepart</i> kurang dari jumlah minimal. 3. SPV melakukan pencatatan pada sistem berupa data setiap ada <i>sparepart</i> yang masuk dan keluar. |

Berdasarkan Tabel 4.4, dapat diketahui logika proses bisnis yang dipakai untuk melakukan perawatan pada mesin. Pada sistem informasi ini terdapat 3 proses yaitu penugasan & perawatan mesin *planned maintenance*, penugasan & perawatan mesin *unplanned maintenance*, serta *inventory sparepart*. Pada penugasan & perawatan mesin *planned maintenance*, setiap mesin akan dijadwalkan penugasan perawatannya dua minggu sebelum periode perawatan itu terlaksana. Untuk *unplanned maintenance*, penugasan perawatan dilakukan ketika ada mesin rusak. SPV akan memberikan penugasan kepada PIC serta teknisi yang membantu. Dan yang terakhir adalah *inventory sparepart*, ketersediaan *sparepart* di departemen *maintenance* harus selalu ada agar ketika *sparepart* dibutuhkan sewaktu-waktu, *sparepart* selalu tersedia. Ketersediaan *sparepart* dapat dikontrol dengan memastikan bahwa jumlah stock harus selalu ada.

4.3.3 Development Strategies

Development strategies akan mencakup segala kebutuhan sistem baru yang meliputi kebutuhan software serta kebutuhan hardware. Berikut ini adalah strategi pengembangan sistem informasi manajemen perawatan yang disesuaikan dengan kondisi dari PT. Gunawan Dianjaya Steel.

1. Level pengembangan *prototype*

Sistem informasi ini merupakan aplikasi yang menggunakan jaringan Local Area Network (LAN) dengan minimal membutuhkan 3 buah computer yang saling berhubungan sehingga sistem informasi ini dapat di akses dengan mudah.

2. *Software Requirement*

Kebutuhan perangkat lunak atau *software* untuk menjalankan aplikasi sistem informasi ini akan dijelaskan pada Tabel 4.5.

Tabel 4.5 Kebutuhan Minimum *Software*

| Unsur | Pilihan | Alasan |
|--------------------------|---|--|
| <i>Operating Sistem</i> | Windows 7 | Populer dan umum digunakan |
| <i>Software Database</i> | Microsoft Access 2013 | 1. <i>Visual programming</i> 2. <i>Event driven programming</i> 3. Menyatu dengan <i>Microsoft Office</i> yang notabene terinstall pada semua komputer |
| Bahasa Pemrograman | <i>Visual Basic for Application (VBA)</i> | 1. Populer dan mudah dipelajari 2. Menyatu dengan <i>Microsoft Office</i> |

3. *Hardware Requirement*

Untuk *hardware requirement*, dapat dilihat pada Tabel 4.6.

Tabel 4.6 Kebutuhan Minimum *Hardware*

| Unsur | Pilihan | Alasan |
|---------------|---|------------------------------|
| <i>Server</i> | Intel Pentium Processor B925C (4M Cache, 2.00 GHz) atau lebih tinggi. | Dapat menjalankan Windows 7 |
| | RAM 1 GB atau lebih tinggi. | |
| | Kapasitas hard disk minimal 80 GB. | |
| <i>Input</i> | <i>Monitor, mouse dan keyboard</i> | Sebagai peralatan antar muka |

4. Dokumentasi

Dokumentasi yang dilakukan dalam pengembangan sistem ini antara lain menggunakan *source code* program untuk memudahkan pengembangan program di kemudian hari serta manual book untuk memudahkan penguasaan kompetensi.

5. Ketrampilan Operator

Operator yang menjalankan aplikasi sistem informasi perawatan ini minimal sudah terbiasa dengan lingkungan windows serta *Microsoft Office*. Selain itu operator diharapkan sudah memiliki kemampuan dasar komputer mengingat aplikasi ini cukup mudah dijalankan.

BAB V PERANCANGAN SISTEM

Pada bagian bab perancangan sistem ini akan dijelaskan tahapan-tahapan dalam perancangan dan pengembangan sistem yaitu yang dimulai dari desain, implementasi hingga pengujian sistem. Bab ini merupakan bagian penting dalam proses perancangan sistem yang diharapkan agar dapat menghasilkan sistem informasi perawatan yang sesuai dengan kebutuhan *user*.

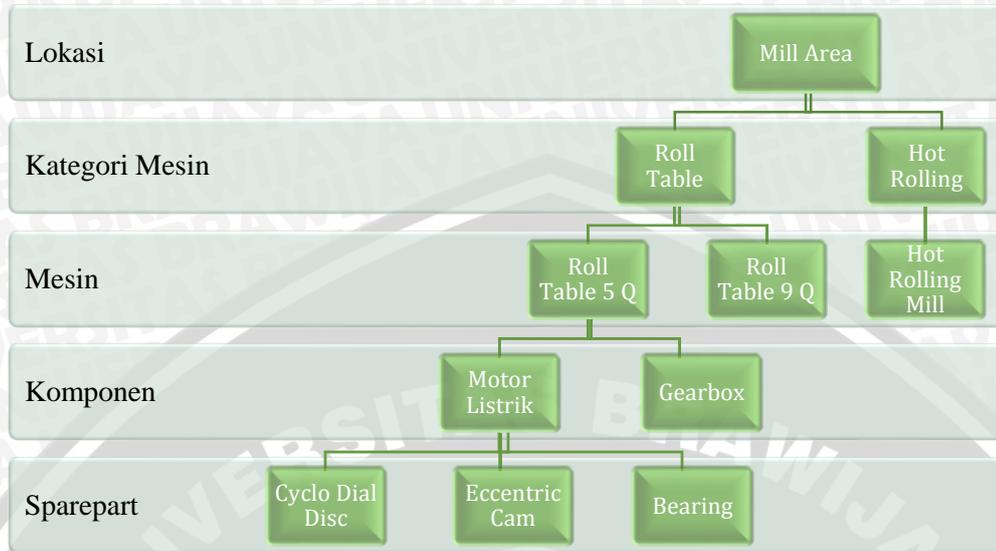
5.1 DESAIN SISTEM

Tujuan dari tahap ini adalah untuk menciptakan model fisik dari sistem yang memenuhi persyaratan desain yang ditetapkan selama fase analisis sistem pada bab sebelumnya. Desain sistem ini merupakan sebuah model logis dari sistem yang dikembangkan dari tahap analisa sebelumnya. Langkah yang ada pada tahap ini meliputi desain *database*, *user interface*, dan desain algoritma. *Deliverable* pada fase ini adalah desain sistem spesifikasi. Dalam desain sistem *database* ini memakai prinsip *group technology* (GT), desain fitur *prototype* yang digunakan memanfaatkan modul dari *Computerized Maintenance Management System*, serta konsep *Overall Equipment Effectiveness* (OEE) untuk mengukur performansi serta efektifitas mesin.

5.1.1 Konsep *Group technology* dalam Desain *Database*

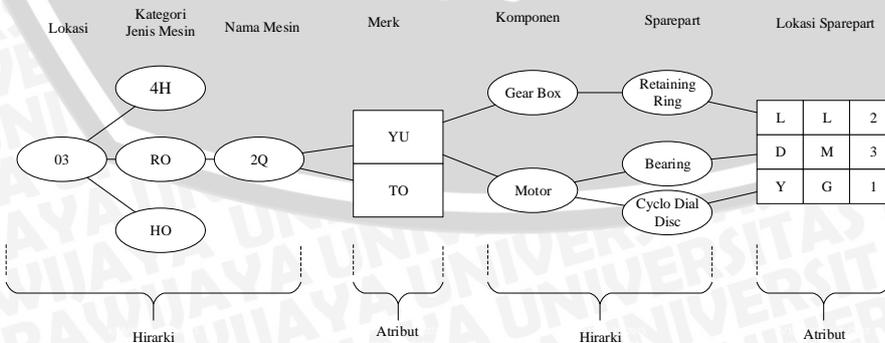
Dalam penelitian ini, subsistem model akan dikembangkan menggunakan konsep GT yang secara filosofi GT untuk mengelompokkan *part* ataupun desain sebuah benda dengan cara melakukan pengelompokan sesuai kesamaan berupa nama mesin serta komponen melalui kodifikasi. Hal ini juga berlaku untuk pengelompokan mesin, komponen, dan *sparepart* yang terdapat pada PT. Gunawan Dianjaya Steel Tbk. Kodifikasi mesin, komponen serta *sparepart* yang akan diterapkan pada PT. GDS adalah sistem kodifikasi *hybrid* yang diadaptasi Opitz yang telah dijelaskan pada bab sebelumnya. Jenis pengkodean *hybrid* ini dipilih karena pada pengaplikasiannya dibutuhkan pengkodean yang hierarkis untuk family *sparepart* dan pengkodean *attribute* untuk family *sparepart* serta pengkodean atribut untuk menentukan lokasi penyimpanan *sparepart*. Pada sistem kodifikasi hirarki di awal, mesin akan dikelompokkan berdasarkan lokasi mesin berada serta jenis mesin. Kode untuk komponen ditentukan dengan mengelompokkan berdasarkan nama mesin yang sama. Sedangkan untuk *sparepart*

dikelompokkan berdasarkan nama mesin serta komponen. Berikut ini adalah gambaran dari konsep GT dengan sistem kodefikasi hirarki yang akan dibuat.



Gambar 5.1 Konsep *Group Technology* dengan Sistem Kodefikasi Hirarki

Sistem kodefikasi selanjutnya adalah sistem kodefikasi atribut yang berbeda dengan sistem kodefikasi hirarki, untuk, bagian-bagian kode pada sistem kodefikasi atribut tidak bergantung pada kode sebelumnya. Terdapat dua kode yang menggunakan kode atribut ini, yaitu kode mesin serta kode *sparepart*. Untuk kode mesin, mesin dikodekan dengan identitas masing-masing mesin berupa nama mesin, serta merk mesin yang beroperasi di PT.GDS Tbk. Sedangkan untuk kode *sparepart*, kodefikasi atribut akan digunakan dalam melakukan pengkodean lokasi letak *sparepart* disimpan. Berikut ini adalah gambaran konsep *group technology* dengan sistem kodefikasi hybrid pada kode mesin, kode komponen mesin, serta kode *sparepart* di PT. GDS Tbk yang diadaptasi dari Opitz pada Gambar 5.2.



Gambar 5.2 Konsep *Group Technology* dengan Sistem Kodefikasi Hybrid

Keterangan:

03 : Slab Area

4H : 4-High Roughing & Finishing Mill

RO : Roll Table

HO : Hot Leveler

2Q : Roll Table 2Q

YU : YuangFang

D : Gudang Dalam

L : Gudang Luar

Y : Halaman

L : Lemari

G : Ground

Berdasarkan Gambar 5.2, lokasi *sparepart* dapat dikelompokkan berdasarkan nama mesin atau *sparepart* dapat dikelompokkan berdasarkan dari jenisnya, namun kode *sparepart* sudah mencakup kode lokasi serta letak *sparepart* agar mudah dicari ketika *sparepart* dibutuhkan. Hasil dari GT tersebut kemudian ditampilkan dalam bentuk *group* yang dapat memberikan rekomendasi tata letak *sparepart* kepada departemen *maintenance* di PT. GDS Tbk.

Saat ini, departemen *maintenance* PT. GDS memiliki beberapa tempat lokasi penyimpanan *sparepart*. Lokasi-lokasi tersebut adalah gudang dalam, gudang luar, serta halaman. Pada masing-masing lokasi tersebut terdapat tempat penyimpanan seperti rak, meja serta lemari. *Sparepart* yang ada pada departemen *maintenance* akan dikelompokkan berdasarkan beberapa kategori yang menentukan dimana lokasi dan letak *sparepart* disimpan. Kategori pengelompokan *sparepart* (Indrajit, 2003:8) yang digunakan untuk membuat parameter kodefikasi antara lain:

1. Menurut harga, yaitu:

a. Memiliki harga tinggi (*High Value Items*)

Sparepart ini biasanya berjumlah sekitar hanya 10% dari jumlah total persediaan, namun jumlah nilainya mewakili sekitar 70% dari seluruh nilai total persediaan dan oleh sebab itu memerlukan tingkat pengawasan yang sangat tinggi.

b. Memiliki harga menengah (*Medium Value Items*)

Sparepart ini biasanya berjumlah kira- kira 20% dari jumlah total persediaan, dan jumlah nilainya juga sekitar 20% dari jumlah total nilai persediaan, sehingga memerlukan tingkat pengawasan yang cukup saja.

c. Memiliki harga rendah (*Low Value Items*)

Berlawanan dengan *sparepart* yang memiliki harga tinggi, jenis *sparepart* ini biasanya berjumlah kira - kira 70% dari total seluruh persediaan, namun nilai harganya hanya mewakili 10% saja dari total nilai barang persediaan, sehingga hanya memerlukan tingkat pengawasan rendah.

2. Menurut cara penggunaannya, yaitu:
 - a. *Rotable*
Sparepart yang dapat dirotasikan antar mesin, dapat diperbaiki, serta harganya relatif paling mahal dibandingkan dengan *sparepart* lain.
 - b. *Repairable*
Sparepart yang memiliki sifat hampir sama dengan *rotable*, namun harganya masih lebih murah dari *rotable*.
 - c. *Consumable*
Sparepart yang habis pakai atau sekali pakai setelah mengalami kerusakan sehingga harus diganti dengan *sparepart* yang baru.
3. Menurut frekuensi penggunaan, yaitu:
 - a. Pemakaian maupun pergerakan cepat (*Fast Moving Items*)
Sparepart yang sangat sering sekali digunakan dalam pengoperasian mesin. Umumnya *sparepart* ini bersifat *periodical* dan *consumable*. Permintaannya dapat mencapai 3-4 kali dalam sebulan.
 - b. Pemakaian maupun pergerakan sedang (*Medium Moving Items*)
Sparepart yang sering digunakan dalam pengoperasian mesin namun tidak sesering *fast moving*, permintaannya mencapai 1-2 kali dalam sebulan. Pada umumnya *sparepart* ini *consumable* dan digunakan dalam *minor repair*.
 - c. Pemakaian maupun pergerakan lambat (*Slow Moving Items*)
Sparepart yang jarang digunakan dalam pengoperasian mesin yang mempunyai permintaan 1 kali dalam 3 bulan. Umumnya digunakan pada *minor* dan *major repair*.

Dari kategori pengelompokan *sparepart* menurut harga, cara penggunaannya, serta frekuensi penggunaannya, berikut merupakan rancangan dari tata letak penyimpanan *sparepart* departemen *maintenance* PT. GDS.

Tabel 5.1 Kriteria Pengelompokan *Sparepart*

| Lokasi Letak | Gudang Dalam (D) | Gudang Luar (L) | Halaman (Y) |
|-----------------|--|--|--|
| Lemari (L) | 1. <i>Slow moving</i> 2. <i>Consumable</i> 3. <i>High value items</i> | 1. <i>Medium moving</i> 2. <i>Consumable</i> 3. <i>Medium value items</i> | 1. <i>Medium moving</i> 2. <i>Consumable</i> |
| Rak (R) | 1. <i>Slow moving</i> 2. <i>High value items</i> 3. <i>Repairable</i> dan <i>Consumable</i> | 1. <i>Medium moving</i> 2. <i>Repairable</i> dan <i>Consumable</i> 3. <i>Medium value items</i> | 1. <i>Fast moving</i> 2. <i>Consumable</i> 3. <i>Sparepart</i> seperti Work Roll, Backup Roll |

Tabel 5.1 Kriteria Pengelompokan *Sparepart* (Lanjutan)

| Lokasi Letak | Gudang Dalam (D) | Gudang Luar (L) | Halaman (Y) |
|-----------------|------------------|-----------------|--|
| Meja (M) | - | - | 1. <i>Fast moving</i> 2. <i>Consumable</i> 3. <i>Low Value Items</i> |
| Tanah (G) | - | - | 1. <i>Fast moving</i> 2. <i>Consumable</i> 3. <i>Low value items</i> |

Berdasarkan Tabel 5.1, gudang dalam menyimpan *sparepart* mempunyai nilai tinggi dan *slow moving* untuk mengurangi resiko hilang. Sedangkan pada halaman yang berada di depan ruang teknisi departemen *maintenance*, *sparepart* yang disimpan memiliki karakteristik *medium moving*, *medium value items* dan *repairable*. Sedangkan gudang luar yang terletak disebelah ruang teknisi, digunakan untuk menyimpan *sparepart* dengan karakteristik *low value items*, *consumable* dan *fast moving*. Klasifikasi lokasi dan letak *sparepart* ini selanjutnya akan menjadi parameter dalam penentuan kodefikasi *sparepart*.

5.1.1.1 Struktur Sistem dan Parameter Kodefikasi

Klasifikasi dan kodefikasi GT yang digunakan dalam perancangan desain *database* pada penelitian ini adalah kodefikasi *hybird* yang menggabungkan metode *monocode* serta *polycode*. Sistem pengkodean yang dirancang nantinya juga dapat diterapkan di perusahaan lain karena kode yang dipakai adalah kode yang bersifat universal. Sebelum membuat tabel data maka diperlukan suatu parameter yang digunakan dalam pengkodean yang menggunakan konsep pengkodean *hybrid*. Berikut parameter yang digunakan pada pengkodean mesin, komponen dan *sparepart*:

1. Kode Lokasi Mesin (*digit 1-2*)
Kode lokasi mesin merupakan kode untuk menunjukkan dimana letak mesin berada pada proses produksil. Misalnya “01” untuk area proses “Slabs Area”, “02” untuk area proses “Furnaces Heat Area” dan seterusnya.
2. Kode Jenis Mesin (*digit 3-4*)
Kode jenis mesin merupakan jenis masing-masing mesin, misalnya “RO” merupakan jenis mesin *Rolling Table*.
3. Kode Nama mesin (*digit 5-6*)
Kode nama mesin merupakan kode yang menunjukkan nama mesin yang diberikan oleh perusahaan.
4. Kode Merk Mesin (*digit 7-8*)

Merk Mesin berisi tentang merk setiap mesin, misalnya “SU” yang berarti merk Sumitomo, “TO” untuk Toyota, “MI” untuk Mitsubishi dan seterusnya.

5. Kode *Hierarchical* Komponen mesin (digit 9-10) dan *Sparepart* (digit 11-12)
Kode *Hierarchical Part* berisi tentang :
 - a. Section Komponen (Level 1) : (Motor Listrik) (digit 9-10)
 - b. *Sub Section*, nama *sparepart* (Level 2) : (Cyclo Dial Disc, Bearing, dll) (digit 11-12)
6. Kode *Attribute Inventory* (digit ke 13-16), kode *Attribute Inventory* berisi tentang:
 1. Lokasi Area (Gudang Dalam, Gudang Luar, Halaman)
 2. Lokasi Rak (Lemari, Ground, Rak)
 3. Nomor (Rak A; baris 1, 2, 3, dst)

Penjelasan mengenai pemberian kode pada mesin, komponen dan *sparepart* dapat dilihat pada Tabel 5.2.

Tabel 5.2 Parameter Kodefikasi

| Parameter | Entity | Atribut | Kode | Keterangan |
|----------------------------|--------------------------|-------------------|-----------|-------------|
| Data Umum Mesin | Lokasi Area Proses Mesin | Slabs Area | 01 | Digit 1-2 |
| | | Milling Area | 03 | |
| | Kategori Jenis Mesin | Roll Table | RO | Digit 3-4 |
| | | Reheating Furnace | RE | |
| | Nama Mesin | Roll Table A1 | R1 | Digit 5-6 |
| Merk Mesin | Yuangfang | YU | Digit 7-8 | |
| | Toyota | TO | | |
| Hirarki Komponen | Komponen | Motor Listrik | MO | Digit 9-10 |
| | | Gearbox | GE | |
| | <i>Spareparts</i> | Bearing | BE | Digit 11-12 |
| <i>Attribute Inventori</i> | Lokasi Area | Gudang Dalam | D | Digit 13 |
| | | Halaman | Y | |
| | Letak | Rak Meja | M | Digit 14 |
| | | Lemari | L | |
| | Nomor | Rak C baris 2 | C2 | Digit 15-16 |

Berikut contoh pemberian kode pada mesin, komponen dan *spareparts*:

| | | | | | | | | | | | | | | | | | |
|-------|---|---|---|---|---|---|---|----------|---|---|------------|---|---|---|---|---|---|
| 0 | 3 | R | O | R | 1 | Y | U | - | M | O | - | B | E | D | M | C | 2 |
| Mesin | | | | | | | | Komponen | | | Spareparts | | | | | | |

Gambar 5.3 Contoh Kodefikasi Mesin, Komponen dan *Spareparts*

5.1.1.2 Kegunaan *Group Technology* dalam Sistem Informasi Manajemen Perawatan Mesin

Dalam sistem informasi manajemen perawatan mesin ini, kodefikasi GT berfungsi sebagai tempat untuk penyimpanan nilai atau suatu data. Sebagai contoh, suatu kode mesin menyimpan suatu data *history* perawatan mesin dan kerusakan mesin yang pernah dialami oleh mesin tersebut. Data *history* kerusakan mesin yang terjadi dan laporan

perbaikan kerusakan mesin yang telah diselesaikan oleh teknisi yang ada dalam *database* akan diolah dan direlasikan dengan data yang lain seperti data mesin, data *sparepart* dan data karyawan sehingga dapat menghasilkan sebuah laporan kerusakan mesin untuk SPV.

Dengan menggunakan kodefikasi GT ini juga karyawan akan dapat mencari informasi mengenai *history* kerusakan dari mesin berdasarkan laporan perawatan mesin yang telah terjadi sebagai referensi untuk menyelesaikan tindakan perawatan yang sedang dilakukan. Selain itu, kegunaan kodefikasi ini juga untuk *sparepart* yang telah dijelaskan pada sub bab sebelumnya akan mempermudah dalam pencarian informasi data *sparepart*.

5.1.2 Konsep *Computerized Maintenance Management System* Dalam Perancangan Sistem Informasi

Dari penelitian ini, sistem informasi yang akan dikembangkan berbasis *Computerized Maintenance Management System* yang merupakan aplikasi perangkat lunak yang mengatur *database* berisi informasi aktivitas pemeliharaan dalam suatu perusahaan. Informasi ini digunakan untuk membantu para karyawan dalam melaksanakan tugas pemeliharaan aset agar lebih efektif, seperti misalnya menentukan mesin mana yang sedang memerlukan perawatan dan gudang mana saja yg menyediakan *spare parts* yang akan digunakan dalam proses perawatan tersebut. (Wireman, 1994:24).

Dari analisis sistem yang telah dilakukan pada bab sebelumnya, kebutuhan sistem informasi manajemen perawatan dapat terakomodasi dari modul-modul yang ada di *Computerized Maintenance Management System* (CMMS). Berikut ini adalah kebutuhan sistem serta modul dari CMMS yang dapat menjadi solusi dari kebutuhan mesin tersebut.

Tabel 5.3 Kebutuhan Sistem dan Modul CMMS Terkait

| No. | Kebutuhan Sistem | Modul CMMS Terkait |
|-----|--|--|
| 1. | Sistem dapat mengontrol tindakan perawatan yang dilakukan dengan memberikan peringatan terhadap tindakan perawatan yang harus dilakukan. | <i>Work order</i> |
| 2. | Sistem dapat diakses oleh manajer, SPV, dan karyawan dengan hak akses yang berbeda sesuai dengan kebutuhan dengan memasukkan <i>username</i> dan <i>password</i> . Administrator dan SPV mempunyai hak untuk mengubah data dan menambah data, sedangkan karyawan tidak dapat mengubah data dan menambahkan data. | <i>Employee & Asset Management,</i> |
| 3. | Sistem dapat melakukan kontrol terhadap penugasan perawatan serta <i>inventory sparepart</i> . | <i>Work order, Asset Management, & Inventory Control</i> |

Tabel 5.3 Kebutuhan Sistem dan Modul CMMS Terkait (Lanjutan)

| No. | Kebutuhan Sistem | Modul CMMS Terkait |
|-----|--|--|
| 4. | Sistem dapat memonitoring seluruh kegiatan perawatan di departemen <i>maintenance</i> mulai dari status pengerjaan perawatan, status <i>sparepart</i> , hingga kondisi performansi mesin. | <i>Work order</i> |
| 5. | Sistem dapat menginformasikan tindakan perawatan yang harus dilakukan oleh teknisi setiap hari dan memberikan informasi <i>inventory sparepart</i> , lokasi <i>sparepart</i> dan <i>history</i> perawatan mesin yang dibutuhkan oleh teknisi | <i>Work order, Asset Management, & Inventory Control</i> |

Dari Tabel 5.3 di atas dapat diketahui bahwa modul CMMS yang dibutuhkan adalah *Work order, Employee, Asset Management, serta Inventory Control*. Modul-modul CMMS tersebut yang nantinya akan diaplikasikan pada Sistem Informasi Manajemen Perawatan Mesin PT. GDS ini.

5.1.3 Pengukuran *Overall Equipment Effectiveness* (OEE)

Pada sistem informasi manajemen perawatan ini akan memanfaatkan *Overall Equipment Effectiveness* (OEE) seperti yang sudah dijelaskan pada bab sebelumnya sebagai alat pengukur performansi dan efektifitas mesin. OEE sendiri dipengaruhi oleh tiga factor yaitu *availability, performance, dan quality*. Berikut ini adalah contoh perhitungan OEE di PT. GDS selama 3 periode.

5.1.3.1 *Availability Rate*

Data yang dibutuhkan untuk menghitung *availability rate* antara lain adalah data jam kerja dan data *downtime* mesin. Pada PT. GDS apabila sebuah mesin mengalami kerusakan dan harus berhenti bekerja, maka mesin yang lain juga berhenti bekerja karena proses produksi saling terkait satu dengan lainnya. Hal ini disebabkan *output* dari suatu mesin merupakan *input* dari proses setelahnya. Apabila satu mesin mati maka tidak ada *output* dari mesin tersebut sehingga tidak ada bahan baku untuk mesin selanjutnya. Tetapi, mesin selanjutnya tetap berjalan menyelesaikan proses produk yang terakhir kali di proses oleh mesin sebelumnya sebelum mati. Contoh perhitungan *availability rate* mesin *Hot Rolling Mill* pada periode produksi 1 mengacu pada persamaan (2-1) dengan data jam kerja dan downtime mesin pada table 5.4 adalah sebagai berikut:

$$\text{Availability rate} = \frac{\text{loading time} - \text{downtime}}{\text{loading time}} \times 100\%$$

$$\text{Availability rate} = \frac{19140 - 1020}{19140} \times 100\% = 94,67 \%$$

Adapun hasil perhitungan *availability rate* mesin *Hot Rolling Mill* dapat dilihat pada Tabel 5.4.

Tabel 5.4 Data dan Hasil Perhitungan *Availability Rate* Mesin *Hot Rolling Mill*

| Periode | Loading Time (menit) | Downtime (menit) | Operating time (menit) | Availability Rate |
|---------|----------------------|------------------|------------------------|-------------------|
| 1 | 19140 | 1020 | 18120 | 94,67% |
| 2 | 14760 | 900 | 13860 | 93,90% |
| 3 | 25020 | 1440 | 23580 | 94,24% |

Sumber: PT. Gunawan Dianjaya Steel (2015)

Hasil perhitungan *availability rate* mesin *Hot Leveler* dapat dilihat pada Tabel 5.5.

Tabel 5.5 Data dan Hasil Perhitungan *Availability Rate* Mesin *Hot Leveler*

| Periode | Loading Time (Jam) | Downtime (Jam) | Operating time (Jam) | Availability Rate |
|---------|--------------------|----------------|----------------------|-------------------|
| 1 | 19140 | 900 | 18240 | 95,29% |
| 2 | 14760 | 780 | 13980 | 94,71% |
| 3 | 25020 | 1260 | 23760 | 94,96% |

Sumber: PT. Gunawan Dianjaya Steel (2015)

Hasil perhitungan *availability rate* mesin *Reheating Furnace* dapat dilihat pada Tabel 5.6.

Tabel 5.6 Data dan Hasil Perhitungan *Availability Rate* Mesin *Reheating Furnace*

| Periode | Loading Time (Jam) | Downtime (Jam) | Operating time (Jam) | Availability Rate |
|---------|--------------------|----------------|----------------------|-------------------|
| 1 | 19140 | 1200 | 17940 | 93,73% |
| 2 | 14760 | 1080 | 13680 | 92,68% |
| 3 | 25020 | 1440 | 23580 | 94,24% |

Sumber: PT. Gunawan Dianjaya Steel (2015)

Availability rate merupakan rasio yang menggambarkan pemanfaatan waktu yang tersedia untuk kegiatan operasi mesin (Stephens, 2014). Dari Tabel 5.4 hingga Tabel 5.7 dapat diketahui bahwa nilai *availability rate* untuk mesin *Hot Rolling Mill* berurutan dari periode produksi 1 hingga 3 adalah 96,47%, 93,90% dan 94,24%. Untuk mesin *Hot Leveler* adalah 95,29%, 94,71%, dan 94,96%. Sedangkan untuk mesin *Reheating Furnace* adalah 93,73%, 92,68%, dan 94,24%.

5.1.3.2 Performance Rate

Data yang dibutuhkan dalam perhitungan *Performance Rate* adalah jumlah bahan baku yang diolah, jumlah ideal produk yang dihasilkan per jamnya dan waktu operasi. Ideal produk yang dihasilkan per jamnya oleh masing-masing mesin berbeda karena lama waktu proses yang berbeda pada tiap mesin. Contoh perhitungan *performance rate* mesin *Hot Rolling Mill* pada periode pertama mengacu pada persamaan (2-2) adalah sebagai berikut:

$$\text{Performance rate} = \frac{2811 \times 5,76}{18120} \times 100\% = 89,36\%$$

Adapun hasil perhitungan *performance rate* mesin *Hot Rolling Mill* dapat dilihat pada Tabel 5.7.

Tabel 5.7 Data dan Hasil Perhitungan *Performance Rate* Mesin *Hot Rolling Mill*

| Periode | Operating time (Menit) | Processed Amount (buah) | Ideal Cycle Time (Menit/buah) | Performance Rate |
|---------|------------------------|-------------------------|-------------------------------|------------------|
| 1 | 18120 | 2811 | 5.76 | 89.36% |
| 2 | 13860 | 2036 | 5.76 | 84.61% |
| 3 | 23580 | 3641 | 5.76 | 88.94% |

Sumber: PT. Gunawan Dianjaya Steel (2015)

$$Performance\ rate = \frac{2825 \times 5,76}{18240} \times 100\% = 89,36\%$$

Hasil perhitungan *performance rate* mesin *Hot Leveler* dapat dilihat pada Tabel 5.8.

Tabel 5.8 Data dan Hasil Perhitungan *Performance Rate* Mesin *Hot Leveler*

| Periode | Operating time (Menit) | Processed Amount (buah) | Ideal Cycle Time (Menit/buah) | Performance Rate |
|---------|------------------------|-------------------------|-------------------------------|------------------|
| 1 | 18240 | 2825 | 5 | 77.44 |
| 2 | 13980 | 2056 | 5 | 73.53 |
| 3 | 23760 | 3653 | 5 | 76.87 |

Sumber: PT. Gunawan Dianjaya Steel (2015)

$$Performance\ rate = \frac{615120 \times 0,025455}{17940} \times 100\% = 89,36\%$$

Hasil perhitungan *performance rate* mesin *Reheating Furnace* dapat dilihat pada Tabel 5.9

Tabel 5.9 Data dan Hasil Perhitungan *Performance Rate* Mesin *Reheating Furnace*

| Periode | Operating time (Menit) | Processed Amount (Kg) | Ideal Cycle Time (Menit/Kg) | Performance Rate |
|---------|------------------------|-----------------------|-----------------------------|------------------|
| 1 | 17940 | 615120 | 0.025455 | 87.28% |
| 2 | 13680 | 440220 | 0.025455 | 81.91% |
| 3 | 23580 | 792880 | 0.025455 | 85.59% |

Sumber: PT. Gunawan Dianjaya Steel (2015)

Performance rate adalah rasio yang menggambarkan kemampuan suatu mesin/peralatan dalam menghasilkan suatu produk/barang (Stephens,2014). Dari Tabel 5.7 hingga Tabel 5.9 dapat diketahui bahwa nilai *performance rate* untuk mesin *Hot Rolling Mill* berurutan dari periode produksi 1 hingga 3 adalah 89,36%, 84,61% dan 88,94%. Untuk mesin *Hot Leveler* adalah 77,44%, 73,53%, dan 76,87% Sedangkan untuk mesin *Reheating Furnace* adalah 87,28%, 81,91%, dan 85,59%.

5.1.3.3 Quality Rate

Data yang dibutuhkan dalam perhitungan ini adalah data produksi dan data produk *defect*. Contoh perhitungan *rate of quality* mesin *Hot Rolling Mill* pada periode 1 mengacu pada persamaan (2-3) adalah sebagai berikut:

$$\text{Rate of quality product} = \frac{2811-256}{2811} \times 100\% = 90,89\%$$

Adapun hasil perhitungan *rate of quality* mesin *Hot Rolling Mill* dapat dilihat pada Tabel 5.10

Tabel 5.10 Data dan Hasil Perhitungan *Quality Rate* Mesin *Hot Rolling Mill*

| Periode | <i>Processed Amount</i> (buah) | <i>Defect Amount</i> (buah) | <i>Rate of Quality</i> |
|---------|--------------------------------|-----------------------------|------------------------|
| 1 | 2811 | 256 | 90.89% |
| 2 | 2036 | 230 | 88.70% |
| 3 | 3641 | 342 | 90.61% |

Sumber: PT. Gunawan Dianjaya Steel (2015)

Hasil perhitungan *quality rate* mesin *hot leveler* dapat dilihat pada Tabel 5.11.

Tabel 5.11 Data dan Hasil Perhitungan *Quality Rate* Mesin *Hot Leveler*

| Periode | <i>Processed Amount</i> (buah) | <i>Defect Amount</i> (buah) | <i>Rate of Quality</i> |
|---------|--------------------------------|-----------------------------|------------------------|
| 1 | 2825 | 231 | 91.82 |
| 2 | 2056 | 228 | 88.91 |
| 3 | 3653 | 312 | 91.46 |

Sumber: PT. Gunawan Dianjaya Steel (2015)

Hasil perhitungan *quality rate* mesin *Reheating Furnace* dapat dilihat pada Tabel 5.12

Tabel 5.12 Data dan Hasil Perhitungan *Quality Rate* Mesin *Reheating Furnace*

| Bulan | <i>Processed Amount</i> (Kg) | <i>Defect Amount</i> (buah) | <i>Rate of Quality</i> |
|-------|------------------------------|-----------------------------|------------------------|
| 1 | 615120 | 6142 | 99.00 |
| 2 | 440220 | 4321 | 99.02 |
| 3 | 792880 | 9233 | 98.84 |

Sumber: PT. Gunawan Dianjaya Steel (2015)

Rate of quality adalah rasio mesin dalam menghasilkan suatu produk sesuai dengan standar yang telah ditetapkan (Stephens,2014). Dari Tabel 4.12 hingga Tabel 4.15 dapat diketahui bahwa nilai *rate of quality* untuk mesin *Hot Rolling Mill* berurutan dari periode produksi 1 hingga 3 adalah 90,89%, 88,70% dan 90,61%. Untuk mesin *Hot Leveler* adalah 91,82%, 88,91%, dan 91,46% Sedangkan untuk mesin *Reheating Furnace* adalah 99%, 99,02 dan 98,84%.

5.1.3.4 Overall Equipment Effectiveness (OEE)

Pada tahap ini dilakukan perhitungan nilai OEE untuk mengetahui efektivitas secara total dari kinerja tiap-tiap peralatan dalam melakukan suatu pekerjaan yang sudah direncanakan, diukur dari data aktual terkait dengan *availability rate*, *performance rate*, dan *rate of quality* yang masing-masing dapat dilihat pada Tabel 5.4 hingga 5.12. Contoh

perhitungan OEE mesin *Hot Rolling Mill* pada periode 1 mengacu pada persamaan (2-4) adalah sebagai berikut:

$$OEE = 94,67\% \times 89,36\% \times 90,89\% = 76,89\%$$

Adapun hasil perhitungan OEE mesin *Hot Rolling Mill* dapat dilihat pada Tabel 5.13.

Tabel 5.13 Hasil Perhitungan *Overall Equipment Effectiveness* Mesin *Hot Rolling Mill*

| Periode | Availability Rate | Performance Rate | Rate of Quality | OEE |
|---------|-------------------|------------------|-----------------|--------|
| 1 | 94.67 | 89.36 | 90.89 | 76,89% |
| 2 | 93.90 | 84.61 | 88.70 | 70,47% |
| 3 | 94.24 | 88.94 | 90.61 | 75,94% |

Sumber: PT. Gunawan Dianjaya Steel (2015)

Hasil perhitungan OEE mesin *Hot Leveler* dapat dilihat pada Tabel 5.14.

Tabel 5.14 Hasil Perhitungan *Overall Equipment Effectiveness* Mesin *Hot Leveler*

| Periode | Availability Rate | Performance Rate | Rate of Quality | OEE |
|---------|-------------------|------------------|-----------------|--------|
| 1 | 95.30 | 77.44 | 91.82 | 67,73% |
| 2 | 94.72 | 73.53 | 88.91 | 61,92% |
| 3 | 94.96 | 76.87 | 91.46 | 66,76% |

Sumber: PT. Gunawan Dianjaya Steel (2015)

Hasil perhitungan OEE mesin *Reheating Furnace* dapat dilihat pada Tabel 5.15.

Tabel 5.15 Hasil Perhitungan *Overall Equipment Effectiveness* Mesin *Reheating Furnace*

| Periode | Availability Rate | Performance Rate | Rate of Quality | OEE |
|---------|-------------------|------------------|-----------------|--------|
| 1 | 93.73 | 87.28 | 99.00 | 88,98% |
| 2 | 92.68 | 81.91 | 99.02 | 75,17% |
| 3 | 94.24 | 85.59 | 98.84 | 66,76% |

Sumber: PT. Gunawan Dianjaya Steel (2015)

Dari Tabel 5.13 hingga Tabel 5.15 dapat diketahui bahwa nilai OEE untuk mesin *Hot Rolling Mill* berurutan dari periode produksi 1 hingga 3 adalah 76,89%, 70,47% dan 75,94%. Untuk mesin *Hot Leveler* adalah 77,73%, 61,92%, dan 66,76% Sedangkan untuk mesin *Reheating Furnace* adalah 88,98%, 75,17%, dan 66,76%.

5.1.4 Desain Database

Untuk tahap desain *database*, yang perlu dilakukan oleh analis adalah mengidentifikasi terlebih dahulu *file-file* yang diperlukan oleh sistem informasi. Langkah-langkah untuk mendesain *database* adalah dengan desain *database* logis, normalisasi dan desain *database* fisik.

5.1.4.1 Desain Database Logis

Desain *database* logis dari sistem informasi akan lebih menjelaskan kepada *user* bagaimana fungsi-fungsi di sistem informasi secara logika dapat bekerja. Desain *database* logis akan digambar dengan menggunakan ERD. ERD menggambarkan entitas serta atribut yang terlibat di dalam sistem. Berikut ini adalah langkah-langkah pembuatan ERD:

1. List Entity

Dalam membuat daftar entitas, langkah pertama yang dilakukan adalah identifikasi entitas, pada tahap ini dilakukan identifikasi terhadap entitas yang terlibat. Setiap entitas tersebut merupakan calon dari tabel yang akan dibuat. Dalam perancangan sistem informasi manajemen perawatan ini, entitas dan atribut yang akan dibuat antara lain:

Tabel 5.16 Daftar Entitas dan Atribut ERD

| Entitas | Atribut |
|----------------------------------|---|
| Lokasi | LocationNo , LocationName |
| Jenis Mesin | AssetCatCode , AssetCategory |
| Asset | AssetNumber , AssetDescription, LocationNo, AssetCatCode, Status, Critically, EmployeeNo, ModelNumber, Fungsi, SerialNumber, Vendor, Contractor, Warranty, DateAcquired, DateDisposed, DepartementNo. |
| Komponen | KomponenID , NamaKomponen, MesinID, Status, ModelNumber, Fungsi, SerialNumber, Vendor, Contractor, Warranty, DateAcquired, DateDisposed, |
| Sparepart | SparepartID , MesinID, KomponenID, NamaSparepart, Status, ModelNumber, Fungsi, SerialNumber, Vendor, Contractor, Warranty, Stok, Lokasi, Letak, No. |
| Karyawan/ Employee | EmployeeNo , Name, Positions, Password, DepartementID, WorkPhone, Handphone, Email, HousePhone, Address, OfficeLocation |
| Checklist Preventive Maintenance | ChecklistNo , ChecklistName, Task |
| Jadwal Preventive Maintenance | PMNo , PMName, AssetNo, Lokasi, ChecklistNo, PeriodDays, SatuanFrekuensiWaktu, FrekuensiPM, StartDate, TargetStartDate, TargetComplete |
| Work order | WorkOrderNo , RequestorNo, EmployeeNo, AssetNo, AssetDesc, LocationNo, ProblemDescription, DateReceived, EstDateStart, EstDateEnd, ActDateEnd, DateRequired, DateHandover, ActionTaken, FailureCause, WorkType, CauseDescription, WOstatus, PMNo, TaskNo |
| OEE | IDOEE , IDMesin, NamaMesin, Performance, Avaibility, Quality, OEE, Periode |

Entitas yang terlibat dari sistem informasi manajemen perawatan merupakan data utama yang diintegrasikan yang merupakan nama tabel dalam *database* yang akan dijelaskan pada sub bab selanjutnya. Atribut merupakan karakteristik dalam entitas merupakan nama *field* dalam *database*, sebagai contoh entitas mesin memiliki AssetNumber, AssetDescription, LocationNo, AssetCatCode, Status, EmployeeID ModelNumber, Fungsi, SerialNumber, Vendor, Contractor, Warranty, DateAcquired, DateDisposed, DepartementNo.

2. Entity-Relational Diagram (ERD)

Tujuan dari dibuatnya ERD adalah untuk menunjukkan objek/entitas data dan *relationship* atau hubungan yang ada pada objek/entitas tertentu. *Relationship* dalam *database* sistem manajemen perawatan mesin sangat penting karena pada tahap ini semua entitas diintegrasikan menjadi satu sehingga dapat menghasilkan informasi yang dibutuhkan oleh *user*. Untuk menggambarkan diagram E-R, maka yang perlu dilakukan selanjutnya adalah mengidentifikasi kardinalitas atau jenis relasi untuk masing-masing entitas yang terlibat seperti pada Tabel 5.17.

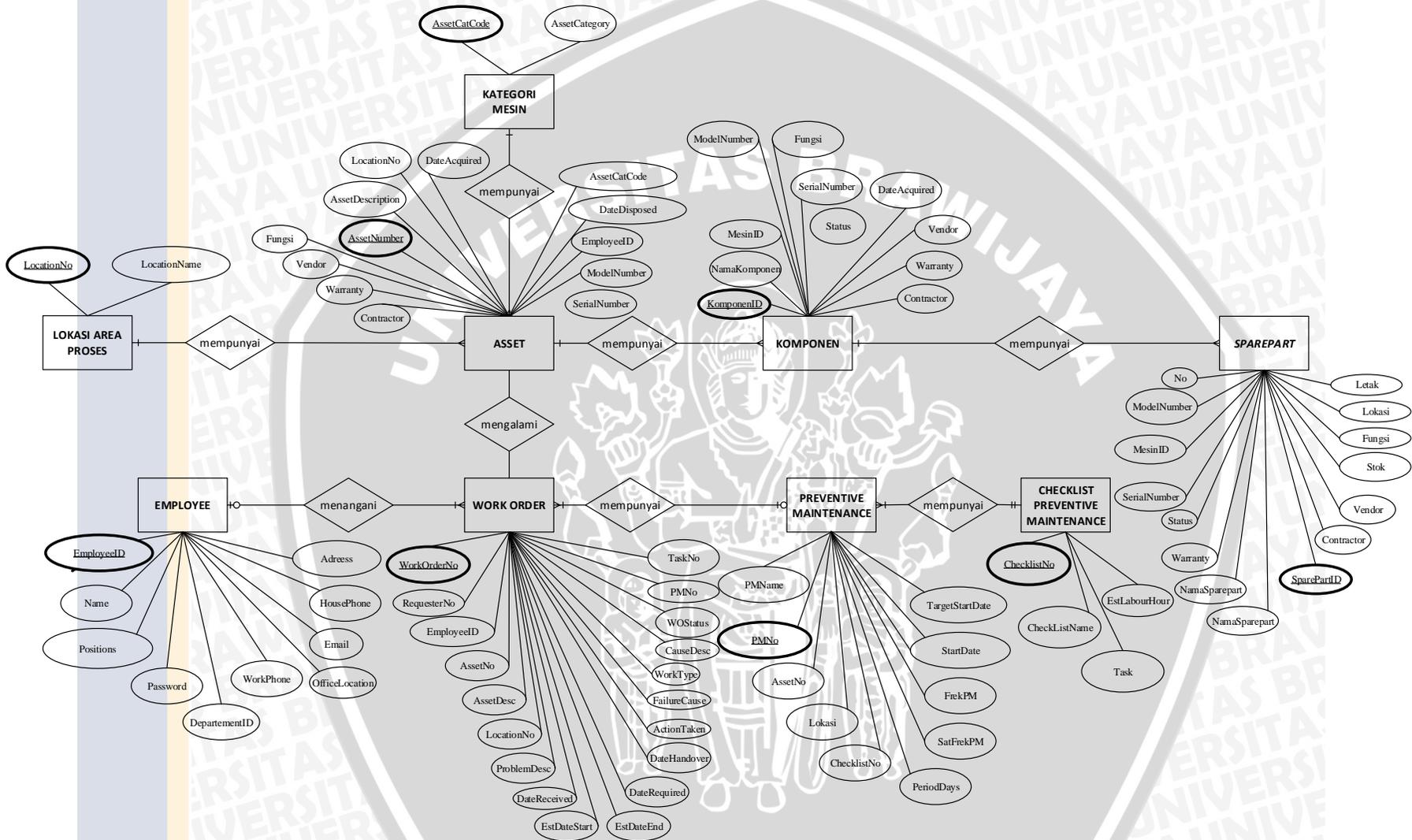
Tabel 5.17 Identifikasi Relasi

| Entitas | Relasi | Entitas | Derajat Relasi Maks-Min |
|-------------------------------|-----------|-------------------------------|-------------------------|
| Lokasi | mempunyai | Kategori Mesin | (0, N) |
| Jenis Mesin | mempunyai | Mesin | (1, N) |
| Mesin | mempunyai | Komponen | (1, N) |
| Mesin | mengalami | <i>Planned Maintenance</i> | (0, N) |
| Mesin | mengalami | <i>Work order</i> | (0, N) |
| Mesin | mempunyai | OEE | (0, N) |
| Komponen | mempunyai | <i>Sparepart</i> | (1, N) |
| Karyawan | menangani | <i>Work order</i> | (0, N) |
| ChecklistPM | mempunyai | <i>Preventive Maintenance</i> | (1, N) |
| <i>Preventive Maintenance</i> | mempunyai | <i>Work order</i> | (1, 1) |
| <i>Work order</i> | ditangani | Karyawan | (1, N) |
| <i>Work order</i> | mempunyai | <i>Planned Maintenance</i> | (0, 1) |

Berdasarkan identifikasi jenis relasi yang dilakukan, kardinalitas atau jenis relasi yang terjadi terhadap masing-masing entitas dapat digambarkan pada Gambar ERD 5.4 disertai dengan atribut-atribut yang menyertai masing-masing entitas.

Berdasarkan Gambar 5.4 dan Tabel 5.17, hubungan antar masing-masing entitas yang terlibat dapat dijelaskan sebagai berikut:

- a. Satu Lokasi mempunyai 0 atau banyak mesin.
- b. Satu kategori mesin mempunyai 1 atau lebih mesin.
- c. Satu mesin terdiri atas banyak komponen.
- d. Satu mesin dapat mengalami 0 atau lebih *Planned Maintenance*.
- e. Satu mesin dapat mengalami 0 atau lebih *Work order*.
- f. Satu mesin dapat mempunyai 0 atau lebih OEE.
- g. Satu komponen dalam mesin terdiri atas banyak *sparepart*.
- h. Satu orang karyawan dapat menangani 0 atau lebih *Work order*.
- i. Satu *Check List* PM mempunyai 0 atau lebih *Planned Maintenance*
- j. Satu *Planned Maintenance* mempunyai 1 *Work order*.
- k. Satu *Work order* ditangani oleh satu orang atau lebih karyawan.
- l. Satu *Work order* dapat mempunyai 0 atau 1 *Planned Maintenance*.



Gambar 5.4 Diagram E-R dengan Atribut

3. Normalisasi

Normalisasi, yaitu struktur tabel yang telah terbentuk dinormalisasi menggunakan tahapan dan aturan yang telah dijelaskan pada subbab sebelumnya. Syarat-syarat pada tahapan normalisasi yang ada (dari tahap 1NF sampai dengan 3NF) sebagai berikut:

- a. Tahap 1NF, merupakan sebuah relasi dimana setiap baris dan kolom berisikan satu dan hanya satu nilai (tidak ada *repeating value*) yang berinteraksi di antara setiap baris pada suatu tabel dan setiap atribut harus mempunyai nilai data yang *atomic*.
- b. Tahap 2NF, merupakan sebuah bentuk data yang telah memenuhi kriteria bentuk 1NF dan setiap atribut *non-primary key* bersifat *fully functionally dependent* pada *primary key* (semua atribut bergantung pada *primary key*).
- c. Tahap 3NF, merupakan sebuah relasi dalam 1NF dan 2NF dan dimana tidak terdapat atribut *non-primary key* yang bersifat *transitively dependent* pada *primary key* (tidak boleh ada *field-field*) yang *non-primary key* saling bergantung).

Berdasarkan syarat-syarat yang terdapat pada tahapan normalisasi (1NF-3NF), maka data atau hubungan yang dibuat sejak awal tidak perlu dilakukan normalisasi tabel karena data atau hubungan sudah memenuhi persyaratan-persyaratan tersebut (data atau hubungan sudah normal).

5.1.4.2 Desain Database Fisik

Desain fisik merupakan aktualisasi dari desain *database* logis yang sangat bergantung dengan *software* yang dipakai. Oleh karena itu pada tahap desain *database* fisik ini diawali dengan pemilihan *software* yang akan dipakai terlebih dahulu yaitu *Microsoft Access 2013*. Setelah dilakukan pemilihan maka dirancanglah struktur tabel sesuai dengan kebutuhan untuk melakukan penyimpanan data. Pada tahap ini, *entity* sudah berubah menjadi tabel sesuai dengan format tabel pada *Microsoft Access 2013*. Berdasarkan ERD dan normalisasi tabel yang telah dibuat sebelumnya maka selanjutnya adalah membuat rancangan tabel *database* sebagai berikut:

1. Entitas Lokasi

Entitas lokasi merupakan tabel data nama dan kode lokasi area proses yang ada di PT. GDS untuk menentukan lokasi mesin. Tabel 5.18 merupakan desain spesifikasi fisik untuk entitas lokasi.

Tabel 5.18 Desain *Database* Entitas Departemen

| <i>Field</i> | <i>Data Type</i> | <i>Field Size</i> | <i>Note</i> | <i>Key</i> |
|--------------|------------------|-------------------|-------------|------------------|
| LocationNo | Short Text | 4 | ID Lokasi | Primary Key (PK) |
| LocationName | Short Text | 4 | NamaLokasi | |

Pada entitas *departemen* terdapat dua atribut yaitu LocationNo yang merupakan kode dari lokasi yang ada di PT. GDS dan LocationName merupakan nama lokasi yang ada di PT. GDS. *Data type* merupakan tipe data atribut pada entitas sedangkan *field size* merupakan panjang karakter atribut sesuai dengan tipe data yang digunakan.

2. Entitas Kategori Mesin

Entitas kategori mesin merupakan tabel data nama dan kode kategori mesin yang ada di PT. GDS untuk menentukan jenis dari suatu mesin. Tabel 5.19 merupakan desain spesifikasi fisik untuk entitas mesin.

Tabel 5.19 Desain *Database* Entitas Jenis Mesin

| <i>Field</i> | <i>Data Type</i> | <i>Field Size</i> | <i>Note</i> | <i>Key</i> |
|---------------|------------------|-------------------|---------------------|------------|
| AssetCatCode | Short Text | 4 | ID Kategori Mesin | PK |
| AssetCategory | Short Text | 100 | Nama Kategori Mesin | |

Pada entitas kategori mesin terdapat dua atribut yaitu AssetCatCode yang merupakan 2 karakter huruf dari nama kategori mesin dan AssetCategory yang merupakan nama jenis mesin.

3. Entitas Asset

Entitas asset merupakan tabel data yang berisikan seluruh data mesin mekanik yang ada pada PT. GDS. Tabel 5.20 merupakan desain spesifikasi fisik untuk entitas mesin.

Tabel 5.20 Desain *Database* Entitas Mesin

| <i>Field</i> | <i>Data Type</i> | <i>Field Size</i> | <i>Note</i> | <i>Key</i> |
|------------------|------------------|-------------------|---------------------|------------------|
| AssetNumber | Short Text | 20 | ID Mesin | PK |
| AssetDescription | Short Text | 100 | Nama Mesin | Foreign Key (FK) |
| LocationNo | Short Text | 255 | ID Lokasi | |
| AssetCatCode | Short Text | 20 | Kategori Mesin | FK |
| Status | Short Text | 30 | Status Mesin | FK |
| EmployeeNo | Short Text | 30 | ID Employee PIC | FK |
| ModelNumber | Short Text | 30 | Series Mesin | |
| Fungsi | Short Text | 255 | Fungsi | |
| SerialNumber | Long Text | 100 | Serial Number Mesin | |
| Vendor | Long Text | 20 | ID Vendor | FK |
| Contractor | Long Text | 20 | ID Contractor | FK |
| Warranty | Date/Time | | | |
| DateAcquired | Date/Time | | | |
| DateDisposed | Date/Time | | | |
| DepartementNo | Short Text | 30 | ID Departemen | FK |

Pada entitas mesin terdapat informasi mengenai suatu mesin, kategori mesin dan lokasi mesin. Dalam entitas ini informasi lengkap suatu mesin seperti nama mesin, lokasi,

kategori mesin, status, model, seri, fungsi, vendor/supplier, serta garansi mesin telah terdata. Kode mesin pada entitas ini akan otomatis terbentuk ketika *user* memasukkan data detail suatu mesin.

4. Entitas Komponen

Entitas komponen merupakan tabel yang berisi informasi semua komponen yang berada dalam setiap mesin. Pada entitas komponen, setiap komponen mempunyai kode komponen yang langsung terbentuk secara otomatis ketika *user* memasukkan data komponen. Tabel 5.21 merupakan desain spesifikasi fisik untuk entitas komponen.

Tabel 5.21 Desain Database Entitas Komponen

| <i>Field</i> | <i>Data Type</i> | <i>Field Size</i> | <i>Note</i> | <i>Key</i> |
|-------------------|-------------------|-------------------|------------------------|------------------|
| KomponenID | <i>Short Text</i> | 20 | ID Komponen | PK |
| NamaKomponen | <i>Short Text</i> | 100 | Nama Komponen | Foreign Key (FK) |
| MesinID | <i>Short Text</i> | 255 | ID Mesin | |
| Status | <i>Short Text</i> | 30 | Status Komponen | FK |
| ModelNumber | <i>Short Text</i> | 30 | Series Komponen | |
| EmployeeNo | <i>Short Text</i> | 30 | ID Employee PIC | |
| Fungsi | <i>Short Text</i> | 255 | Fungsi Komponen | |
| SerialNumber | <i>Long Text</i> | 100 | Serial Number Komponen | |
| Vendor | <i>Long Text</i> | 20 | ID Vendor | FK |
| Contractor | <i>Long Text</i> | 20 | ID Contractor | FK |
| Warranty | <i>Date/Time</i> | | | |
| DateAcquired | <i>Date/Time</i> | | | |
| DateDisposed | <i>Date/Time</i> | | | |

5. Entitas Sparepart

Entitas *sparepart* merupakan tabel data *sparepart* mesin yang ada pada departemen *maintenance* PT. APW yang dihubungkan dengan kode mesin dan kode *sparepart*. Pada entitas *sparepart*, informasi seperti jumlah *sparepart* dan lokasi *sparepart* tercatat disini. Tabel 5.22 merupakan desain spesifikasi fisik untuk entitas *sparepart*.

Tabel 5.22 Desain Database Entitas Sparepart

| <i>Field</i> | <i>Data Type</i> | <i>Field Size</i> | <i>Note</i> | <i>Key</i> |
|--------------------|-------------------|---------------------|-------------------------|------------------|
| SparepartID | <i>Short Text</i> | 20 | ID Sparepart | PK |
| NamaSparepart | <i>Short Text</i> | 100 | Nama Sparepart | Foreign Key (FK) |
| EmployeeNo | <i>Short Text</i> | 255 | ID Employee PIC | FK |
| MesinID | <i>Short Text</i> | 20 | ID Mesin | FK |
| KomponenID | <i>Short Text</i> | 20 | ID Komponen | FK |
| Status | <i>Number</i> | 2 | Status | |
| Fungsi | <i>Number</i> | <i>Long Integer</i> | Fungsi | |
| SerialNumber | <i>Long Text</i> | 100 | Serial Number Sparepart | |
| Vendor | <i>Long Text</i> | 20 | ID Vendor | FK |
| Contractor | <i>Long Text</i> | 20 | ID Contractor | FK |
| Warranty | <i>Date/Time</i> | | | |

Tabel 5.22 Desain Database Entitas Sparepart (Lanjutan)

| <i>Field</i> | <i>Data Type</i> | <i>Field Size</i> | <i>Note</i> | <i>Key</i> |
|--------------|------------------|-------------------|--------------------|------------|
| DateAcquired | Date/Time | | | |
| DateDisposed | Date/Time | | | |
| Stock | Number | 5 | Jumlah | |
| Lokasi | Short Text | 100 | Lokasi Sparepart | FK |
| Letak | Short Text | 100 | Letak Sparepart | FK |
| NoLetak | Number | 5 | No Letak Sparepart | FK |

6. Entitas *Employee*

Entitas *employee* merupakan data yang berisi informasi karyawan *maintenance* meliputi informasi pribadi setiap karyawan pada departemen *maintenance*. Tabel 5.23 merupakan desain spesifikasi fisik untuk entitas karyawan.

Tabel 5.23 Desain Database Entitas Karyawan

| <i>Field</i> | <i>Data Type</i> | <i>Field Size</i> | <i>Note</i> | <i>Key</i> |
|-------------------|------------------|-------------------|----------------------------------|------------|
| EmployeeNo | Short Text | 10 | ID Karyawan | PK |
| Name | Short Text | 20 | Nama Karyawan | |
| Positions | Short Text | 20 | Jabatan (SPV, Foreman, karyawan) | |
| Password | Short Text | 20 | Password Login Karyawan | |
| DepartementID | Short Text | 2 | Lokasi Departement | |
| WorkPhone | Short Text | 20 | No. Telp. Kantor | |
| Handphone | Number | Long Integer | No. Handphone | |
| Email | Date/Time | 20 | Email Karyawan | |
| HousePhone | Number | Long Integer | No. Telp. Rumah | |
| Address | Short Text | 100 | Alamat Rumah | |
| OfficeLocation | Short Text | 100 | Alamat Kantor | |

Pada entitas *employee* ini, juga terdapat attribute *password*. *Password* akan dikombinasi dengan *id* karyawan untuk akses login yang membatasi akses masing-masing karyawan sesuai dengan jabatan di dalam sistem informasi manajemen perawatan ini.

7. Entitas Checklist *Preventive Maintenance*

Entitas *checklist preventive maintenance* merupakan tabel kategori jenis kegiatan perawatan *preventive maintenance*. Tabel 5.24 merupakan desain spesifikasi fisik untuk entitas kerusakan mesin.

Tabel 5.24 Desain Database Entitas Kerusakan Mesin

| <i>Field</i> | <i>Data Type</i> | <i>Field Size</i> | <i>Note</i> | <i>Key</i> |
|---------------|------------------|-------------------|----------------------|------------|
| ChecklistNo | Short Text | 20 | Kode Kerusakan Mesin | PK |
| ChecklistName | Short Text | 30 | Kode Mesin | |
| Task | Short Text | 255 | Kode Komponen | |

Pada entitas ini, akan dihubungkan dengan entitas jadwal *preventive maintenance*. Nantinya kegiatan *preventive maintenance* akan menjadi salah satu bagian dari *work order*.

8. Entitas Jadwal *Preventive Maintenance*

Entitas jadwal *preventive maintenance* merupakan tabel data informasi tindakan *preventive maintenance* secara periodik yang harus dilakukan untuk setiap mesin. Tabel 5.25 ini merupakan desain spesifikasi fisik untuk entitas tindakan perawatan.

Tabel 5.25 Desain Database Entitas Tindakan Perawatan

| <i>Field</i> | <i>Data Type</i> | <i>Field Size</i> | <i>Note</i> | <i>Key</i> |
|----------------------|-------------------|-------------------|------------------------------------|------------|
| PMNo | <i>Short Text</i> | 10 | ID <i>Preventive Maintenance</i> | PK |
| PMName | <i>Short Text</i> | 2 | Nama <i>Preventive Maintenance</i> | FK |
| AssetNo | <i>Short Text</i> | 20 | ID Mesin | FK |
| Lokasi | <i>Short Text</i> | 20 | ID Lokasi | FK |
| CheckListNo | <i>Number</i> | 10 | ID Checklist PM | FK |
| PeriodDays | <i>Number</i> | 2 | | |
| SatuanFrekuensiWaktu | <i>Number</i> | 2 | | |
| FrekuensiPM | <i>Number</i> | 2 | | |
| StartDate | <i>Date/Time</i> | | | |
| TargetStartDate | <i>Date/Time</i> | | | |
| TargetComplete | <i>Date/Time</i> | | | |

9. Entitas *Work order*

Entitas *work order* merupakan tabel data yang berisi informasi jadwal penugasan perawatan mesin yang dijdwalkan oleh SPV. Tabel 5.26 merupakan desain spesifikasi fisik untuk entitas *work order*.

Tabel 5.26 Desain Database Entitas Jadwal Perawatan Mesin

| <i>Field</i> | <i>Data Type</i> | <i>Field Size</i> | <i>Note</i> | <i>Key</i> |
|--------------------|-------------------|-------------------|--|------------|
| WorkOrderNo | <i>Short Text</i> | 20 | ID <i>Work order</i> | PK |
| RequestorNo | <i>Short Text</i> | 2 | ID Pembuat WO | FK |
| EmployeeNo | <i>Short Text</i> | 10 | ID Penerima WO | FK |
| AssetNo | <i>Short Text</i> | 20 | ID Mesin | FK |
| AssetDesc | <i>Short Text</i> | 50 | Nama Mesin | FK |
| LocationNo | <i>Short Text</i> | 20 | ID Lokasi Mesin | FK |
| ProblemDescription | <i>Short Text</i> | 255 | Penjelasan Masalah WO | |
| DateReceived | <i>Date/Time</i> | | Tanggal WO di buat | |
| EstDateStart | <i>Date/Time</i> | | Estimasi Tanggal WO dimulai | |
| EstDateEnd | <i>Date/Time</i> | | Estimasi Tanggal WO selesai | |
| ActDateEnd | <i>Date/Time</i> | | Tanggal WO dimulai | |
| DateRequired | <i>Date/Time</i> | | Tanggal WO selesai | |
| ActionTaken | <i>Short Text</i> | 255 | Perbaikan yang Dilakukan | |
| FailureCause | <i>Short Text</i> | 255 | Penyebab Kerusakan | |
| WorkType | <i>Number</i> | 2 | Tipe Pekerjaan | FK |
| CauseDescription | <i>Short Text</i> | 255 | Permasalahan | |
| WOStatus | <i>Number</i> | 2 | Status WO | FK |
| PMNo | <i>Number</i> | 10 | ID <i>Preventive Maintenance</i> | FK |
| TaskNo | <i>Number</i> | 10 | ID Pekerjaan <i>Preventive Maintenance</i> | FK |

Pada entitas jadwal *work order*, SPV memasukkan data jadwal perawatan yang ia rencanakan, informasi pada tabel ini antara lain nama mesin yang akan dikenai jadwal

perawatan, nama karyawan sebagai PIC yang akan mengerjakan jadwal perawatan, prioritas tindakan perawatan dan tanggal perawatan mesin seharusnya dilakukan.

10. Entitas Perhitungan OEE

Entitas perhitungan merupakan tabel data *history* perhitungan OEE mesin yang pernah dilakukan oleh teknisi atau PIC mesin. Tabel 5.27 merupakan desain spesifikasi fisik untuk entitas kerusakan mesin.

Tabel 5.27 Desain Database Entitas Perhitungan OEE

| <i>Field</i> | <i>Data Type</i> | <i>Field Size</i> | <i>Note</i> | <i>Key</i> |
|--------------------|-------------------|-------------------|--------------------------|------------|
| IDOEE | <i>Short Text</i> | 20 | ID OEE | PK |
| IDMesin | <i>Short Text</i> | 20 | Kode Mesin | PK |
| NamaMesin | <i>Short Text</i> | 30 | Kode Komponen | |
| Periode | <i>Number</i> | 2 | Periode Perhitungan OEE | |
| <i>Performance</i> | <i>Number</i> | 10 | Nilai <i>Performance</i> | |
| <i>Avaibility</i> | <i>Number</i> | 10 | Nilai <i>Avaibility</i> | |
| <i>Quality</i> | <i>Number</i> | 10 | Nilai <i>Quality</i> | |
| OEE | <i>Number</i> | 10 | Nilai OEE | |

Pada entitas perhitungan OEE, setiap kode OEE memiliki data perhitungan variable OEE yang pernah dilakukan, pada entitas ini dilengkapi atribut ID Mesin yang dapat menunjukkan mesin yang dilakukan perhitungan. Periode untuk menunjukan periode perhitungan yang dilakukan. Serta atribut berupa *performance*, *avaibility*, *quality*, serta nilai OEE.

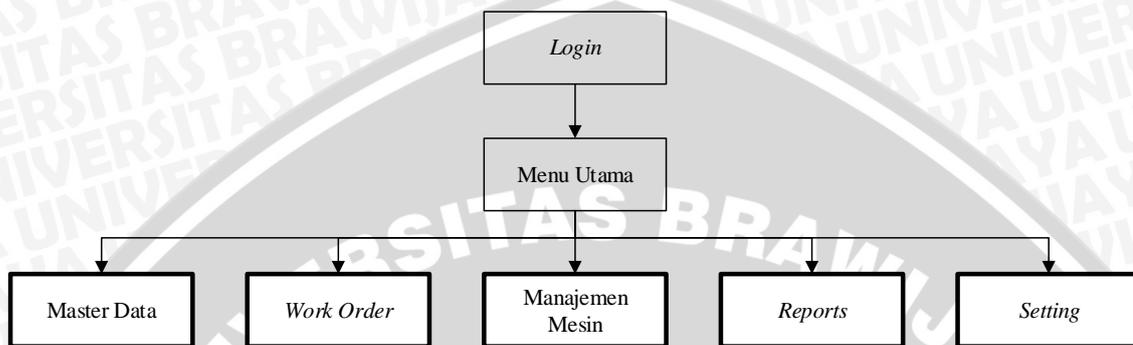
5.1.5 Desain User Interface

Setelah tahap desain *database* berupa *database* logis, normalisasi dan *database* fisik, tahap selanjutnya adalah membuat desain *user interface* (desain antar muka). Tujuan membuat desain *user interface* adalah membuat rancangan dari tampilan sistem yang nantinya akan berinteraksi langsung dengan *user*. Desain *user interface* harus siap digunakan serta harus sesuai dengan kebutuhan pengguna karena desain ini merupakan desain sistem dialog yang dapat diartikan dan diimplementasikan sehingga pengguna dapat berkomunikasi dengan sistem yang dirancang. Desain *user interface* meliputi *hierarki menu*, *form* dan *report*.

5.1.5.1 Bagan Hierarki Menu User Interface

Hierarki menu yang dirancang untuk sistem informasi manajemen perawatan memiliki tujuan untuk memudahkan *user* dalam menggunakan aplikasi. Menu yang dimunculkan di awal pada sistem informasi manajemen perawatan adalah menu utama

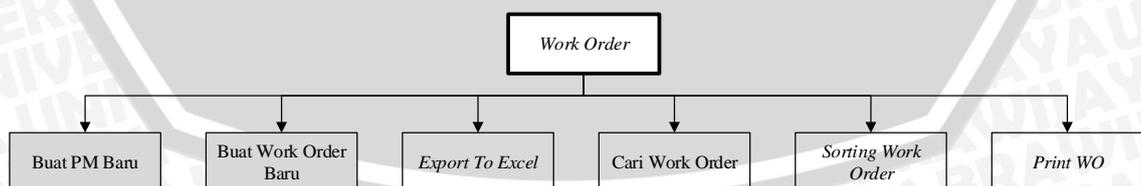
yang berisi *form login*. *Form login* pada menu utama merupakan menu awal yang akan membedakan pengguna dari aplikasi sistem informasi manajemen perawatan sesuai dengan jabatan karena *input* yang diberikan dari *user* pada *form login* akan menentukan hak akses atau permission dalam mengakses modul di dalam sistem informasi perawatan ini sesuai dengan kebutuhan masing-masing *user* pada Tabel 4.1. Gambar 5.5 merupakan desain bagan hierarki menu dalam sistem informasi manajemen perawatan mesin.



Gambar 5.5 Desain Hierarki Menu Sistem Informasi Manajemen Perawatan Berbasis CMMS dan *Group Technology*

Dari Gambar 5.5 di atas dapat dilihat bahwa setelah *user* melakukan *login* terdapat menu utama yang saling berhubungan dalam satu form Menu Utama. Ketiga *user* akan menggunakan menu yang sama hanya hak akses antar *user* yang berbeda. Manajer mengakses semua menu dikarenakan tugas serta tanggung jawab manajer pada keseluruhan sistem yang ada di departemen *maintenance*. SPV dapat mengakses semua menu kecuali menu master data karena dalam job description di departemen *maintenance*, SPV hanya berhak mengelolah penjadwalan dan penugasan perawatan, kondisi mesin, serta laporan kepada manajer. Sedangkan teknisi tidak mendapatkan hak akses terhadap menu yang dapat menambah data seperti yang ditandai pada gambar di atas. Berikut hirarki menu yang terdapat pada masing-masing *form*:

1. *Form Maintenance*

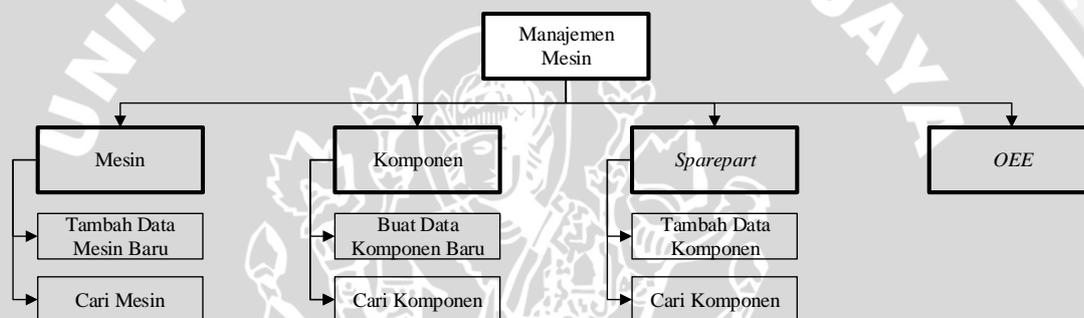


Gambar 5.6 Desain Hierarki Menu *Work Order*

Dari Gambar 5.6 dapat dilihat bahwa *user* dapat memilih menu yang berada di dalam form *maintenance*. Berikut ini merupakan penjelasan untuk masing-masing menu di form *maintenance*.

- a. Buat PM Baru, merupakan menu untuk menambah jadwal *preventive maintenance* secara periodik.
- b. Buat *Work order* Baru, merupakan menu untuk membuat penjadwalan serta penugasan perawatan mesin.
- c. Cari *Work order*, merupakan menu untuk mencari *work order* dengan memasukan data mengenai *work order* tersebut seperti, PIC yang bertugas, type *work order* dan lainnya.
- d. Sorting *Work order*, merupakan menu untuk mengurutkan *work order* berdasarkan tanggal pembuatan, nomer *work order*, atau type *work order*.
- e. Print WO, merupakan menu untuk mencetak detail *work order* sesuai dengan nomer yang dipilih.

2. Form Manajemen Aset



Gambar 5.7 Desain Hierarki Menu Manajemen Mesin

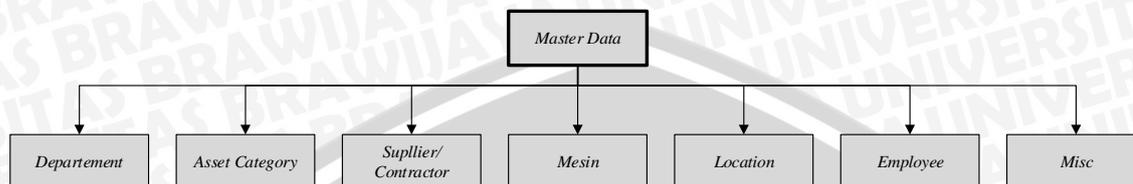
Berdasarkan Gambar 5.7 dapat dilihat bahwa *user* dapat memilih menu yang berada di dalam form manajemen aset. Berikut ini merupakan penjelasan untuk masing-masing menu di form Manajemen Aset.

- a. Mesin, merupakan menu untuk memajemen mesin yang ada di PT. GDS. Di dalam form mesin terdapat menu Tambah Data Mesin untuk menambah data mengenai mesin baru serta menu Cari Mesin untuk melakukan pencarian data mesin dengan memasukan identitas mesin tersebut.
- b. Komponen, merupakan menu untuk memajemen komponen berdasarkan mesin yang ada di PT. GDS. Di dalam form komponen terdapat menu Tambah Data Komponen untuk menambah data mengenai komponen baru serta menu Cari Komponen untuk melakukan pencarian data mesin dengan memasukan identitas mesin tersebut.
- c. *Sparepart*, merupakan menu untuk memajemen *sparepart* mesin yang ada di PT. GDS. Di dalam form *sparepart* terdapat menu Tambah Data Komponen untuk menambah data mengenai *sparepart* baru serta menu Cari *Sparepart*

untuk melakukan pencarian data mesin dengan memasukkan identitas mesin tersebut.

- d. OEE, merupakan menu untuk melakukan perhitungan OEE terhadap mesin yang berada di PT. GDS.

3. *Form Master Data*



Gambar 5.8 Desain Hierarki Menu *Master Data*

Berdasarkan Gambar 5.8 di atas, dapat dilihat bahwa *user* dapat memilih menu menambah data yang berada di dalam form Master Data. Berikut ini merupakan data yang bisa ditambah di form Master Data.

- a. *Departemen*
- b. *Asset Category*
- c. *Supplier/Contractor*
- d. *Mesin/Asset*
- e. *Lokasi*
- f. *Karyawan*

5.1.5.2 *Desain Interface Form*

Desain *interface form* dibuat untuk merancang tampilan antar muka antara *user* dengan komputer agar lebih komunikatif dan informatif. Dalam desain *user interface form* ini, tata letak menu dan warna harus diperhatikan agar *user* tidak mengalami kesulitan ketika menggunakan aplikasi sistem informasi manajemen perawatan mesin ini. Warna yang digunakan pada *form* aplikasi ini hampir senada, yaitu menggunakan warna biru muda, biru tua dan putih. Warna-warna tersebut dipilih dengan pertimbangan menyesuaikan dengan warna identitas PT. Gunawan Dianjaya Steel. Selain itu, warna-warna tersebut merupakan kombinasi warna yang netral pada mata. Ukuran *font* dan ukuran *form* juga disesuaikan agar tidak terlalu kecil sehingga tetap mudah dibaca oleh *user*. Berikut *form* yang akan dirancang sesuai dengan hierarki menu pada Gambar 5.5.

1. *Form Main Menu*

Berikut merupakan desain *form Main Menu* yang merupakan rancangan visual dari hirarki menu *form Main Menu* yang digambarkan pada Gambar 5.6 beserta menu yang ada pada *form Main Menu* pada Gambar 5.9.

The image shows a wireframe for a 'Form Main Menu'. At the top, there is a box labeled 'LOGO PERUSAHAAN' on the left and 'Login Sebagai : Manajer' on the right. Below this is a vertical column of six buttons numbered 1 through 6. To the right of these buttons are two large rectangular areas, labeled 7 and 8, which represent the main content of the menu.

Gambar 5.9 Desain *Form Main Menu*

Keterangan:

- | | |
|--|---------------|
| 1. <i>Maintenance</i> | 8. Visual OEE |
| 2. Manajemen Aset | |
| 3. Master Data | |
| 4. Report | |
| 5. Setting | |
| 6. <i>Exit</i> | |
| 7. Penugasan dan Penjadwalan Perawatan (<i>Work order</i>) | |

2. *Form Work Order*

Berikut merupakan desain *form maintenance* yang merupakan rancangan visual dari hirarki menu *form maintenance* pada Gambar 5.6 beserta menu yang ada pada *form maintenance*.

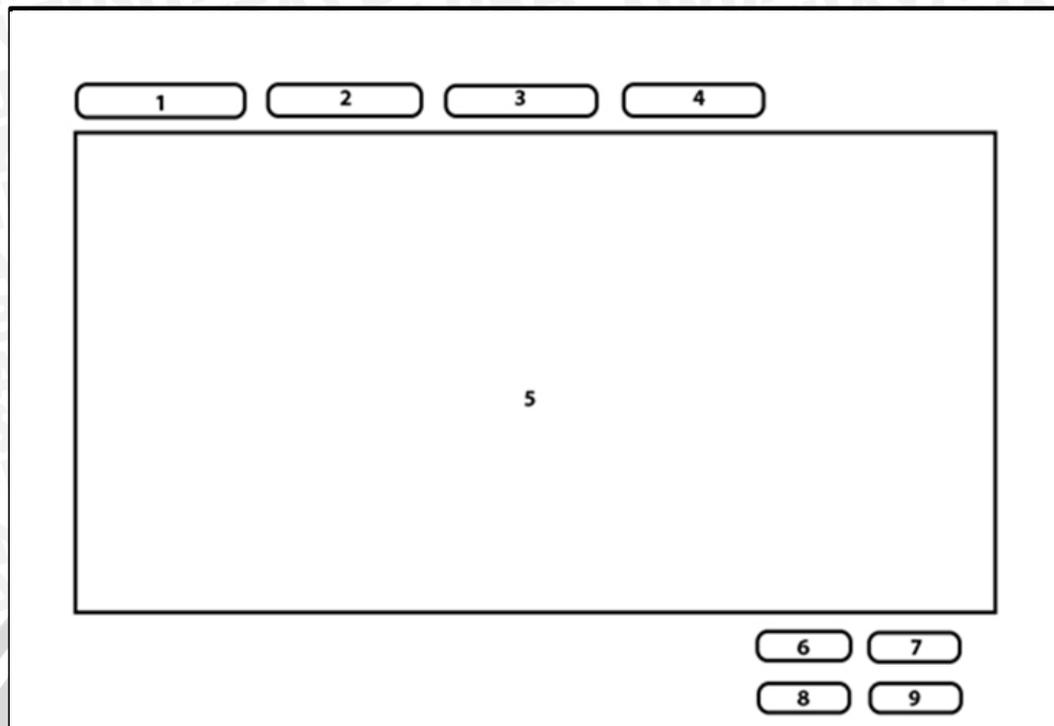
The diagram shows a rectangular form layout. At the top is a large empty box labeled '1'. Below it is a horizontal row of six buttons labeled '7', '6', '5', '4', '3', and '2' from left to right. Below this row is another horizontal row of two buttons labeled '8' and '9'. At the bottom is a large empty box labeled '10'.

Gambar 5.10 Desain *Form Work Order*

Keterangan:

- | | |
|--|---------------------------------------|
| 1. Penugasan dan Penjadwalan Perawatan (<i>Work order</i>) | 7. PM Baru |
| 2. Close | 8. Cari <i>Work order</i> |
| 3. Export To Excel | 9. <i>Sorting Work order</i> |
| 4. Lihat Semua | 10. Parameter Cari dan <i>Sorting</i> |
| 5. <i>Print WO</i> | |
| 6. <i>Work order Baru</i> | |
3. *Form* Manajemen Mesin

Berikut merupakan desain *form* Manajemen Mesin yang merupakan rancangan visual dari hirarki menu *form* Manajemen Mesin pada Gambar 5.7 beserta menu yang ada pada *form* Manajemen Mesin.



Gambar 5.11 Desain *Form* Manajemen Mesin

Keterangan:

- | | |
|---------------------|-----------------------|
| 1. <i>Mesin</i> | 6. <i>Search Aset</i> |
| 2. <i>Komponen</i> | 7. <i>Aset Baru</i> |
| 3. <i>Sparepart</i> | 8. <i>View All</i> |
| 4. <i>OEE</i> | 9. <i>Close</i> |
| 5. <i>List Aset</i> | |
4. *Form Master Data*

Berikut ini merupakan desain *form* master data yang merupakan rancangan visual dari hirarki menu *form* master data pada Gambar 5.8 beserta menu yang ada pada *form* master data.

The diagram shows a form layout with a header row containing eight numbered tabs (1 through 8). Below the tabs are two large rectangular boxes, labeled 9 and 10, representing the main data entry areas.

Gambar 5.12 Desain Form Master Data

Keterangan:

- | | |
|------------------------|-------------|
| 1. Departement | 5. Assets |
| 2. Failure Code | 6. Location |
| 3. Asset Category | 7. Employee |
| 4. Supplier/Contractor | 8. Misc |

5.1.5.3 Desain Report

Sistem informasi manajemen perawatan mesin dapat menghasilkan *report* untuk SPV Mekanik maupun Manajer. *Report* untuk SPV dan Manajer merupakan *report* yang diperlukan sebagai pertimbangan dalam mengambil kebijakan-kebijakan di departemen *maintenance*. Berikut desain *report* pada sistem informasi manajemen perawatan mesin.

1. *Report Inventory Sparepart*

Report ini berisi data *spareparts* yang dikelompokkan berdasarkan jenis mesin serta komponen yang berisi informasi mengenai persediaan *spareparts* yang ada di departemen *maintenance*. Pada *report inventory sparepart*, SPV maupun manajer akan mengetahui jumlah stok tiap *sparepart* sehingga SPV dapat menentukan keputusan untuk memesan *sparepart* yang *low-stock* agar stok *sparepart* tetap dalam kondisi tersedia.

2. *Report Work order*

Report ini berisi jadwal serta penugasan perawatan mesin selama suatu periode untuk SPV maupun Manajer sebagai bentuk laporan periode. *Report* ini merupakan laporan yang menunjukkan tindakan perawatan apa saja yang sudah dilakukan dan

yang belum dilakukan dalam periode tertentu. Dalam *report* ini, dapat diperoleh informasi seputar jadwal perawatan mesin yang mencakup tanggal dan jam perawatan preventif yang harus dilakukan, jenis tindakan perawatan, mesin yang dikenai tindakan perawatan, nama PIC serta teknisi yang melakukan tindakan perawatan, *checklist* yang merupakan penanda tindakan perawatan mana yang sudah atau belum dilakukan dan keterangan yang berisi kondisi-kondisi khusus mengenai perawatan mesin. Dengan *report* ini, SPV maupun manajer *maintenance* dapat mengetahui *work order* yang mengalami keterlambatan untuk digunakan sebagai bahan evaluasi tindakan perawatan selanjutnya.

3. *Report* OEE

Report OEE ini berisi data perhitungan OEE yang telah terjadi selama periode tertentu. Pada *report* ini akan muncul hasil dari perhitungan OEE untuk mesin-mesin yang dilakukan perhitungan. Report ini tersaji dalam bentuk tabel yang dilekngkapi oleh identitas mesin, periode pelaksanaan perhitungan, *avaibility*, *performance*, serta *quality*. Dari report ini, manajer maupun SPV akan mendapatkan rekomendasi dari nilai OEE untuk menentukan prioritas perawatan mesin pada kegiatan *planned maintenance* serta membantu dalam menentukan tindakan perawatan yang tepat.

5.1.6 Desain Algoritma

Desain Algoritma memiliki tujuan untuk merancang tahapan proses apa saja yang harus dilakukan sehingga *input*, *database*, dan *database* menghasilkan *output* yang diharapkan dan dapat ditampilkan, algoritma dapat dinyatakan dengan *flowchart* ataupun *pseudocode*. Pada penelitian ini, yang digunakan adalah *pseudocode* yang nantinya akan memberikan gambaran suatu algoritma bagaimana proses dalam sistem yang akan dibuat berjalan. Berikut *pseudocode* dari beberapa proses penting yang ada dalam sistem informasi manajemen perawatan mesin ini, yaitu *pseudocode* menambah data, menyimpan data, dan algoritma perhitungan OEE mesin dan *pseudocode report*.

1. *Pseudocode* untuk menambah dan menyimpan data

Proses penambahan data dan penyimpanan data pada sistem informasi manajemen perawatan mesin digunakan pada data mesin, komponen, *sparepart*, karyawan, kerusakan mesin, dan *workorder*. *Pseudocode* yang akan ditampilkan berikut ini, merupakan *pseudocode* dari proses penambahan, penyimpanan dan penghapusan

data mesin. Gambar 5.13 merupakan *pseudocode* dalam proses menambah, menyimpan, dan memperbaiki data.

```
//tambah dan simpan data
//menambah data atau memasukkan data baru
    Mengambil data ← Pilih data dari [Asset]
    Posisi rekam data ← baris tabel mesin – 1
    Memilih lokasi mesin
    Memilih kategori mesin
    Memasukkan merk, model/kapasitas, series, tahun dan fungsi mesin
    Membuat kode mesin ← Kode Lokasi + Kode Kategori Mesin + Nama Mesin

//menyimpan data
    Menyimpan data ← Memasukkan (AssetNumber, AssetDescription, LocationNo,
    AssetCatCode, Status, Critically, EmployeeNo,
    ModelNumber, Fungsi, SerialNumber, Vendor, Contractor,
    Warranty, DateAcquired, DateDisposed.) ke dalam record
    [Assets]
    Memperbarui data ← UPDATE data pada [Asset] sesuai dengan atribut yang
    diperbarui (LocationNo, AssetCatCode, Status, Critically,
    EmployeeNo, ModelNumber, Fungsi, SerialNumber,
    Vendor, Contractor, Warranty, DateAcquired,
    DateDisposed.)
```

Gambar 5.13 *Pseudocode* Proses Menambah, Menyimpan, Memperbarui, dan Menghapus Data

2. *Pseudocode* perhitungan OEE

Dalam perhitungan OEE akan menghasilkan nilai OEE yang menjadi rekomendasi prioritas perbaikan dalam *Preventive Maintenance*. Prosesnya meliputi pengumpulan data kondisi mesin, perhitungan variable-variable yang mempengaruhi OEE (*Performance*, *Quality*, dan *Availability*). Berikut ini adalah algoritma perhitungan OEE dalam bentuk *pseudocode*.

```
//Perhitungan OEE

Memilih mesin
Memilih "AssetNo" sebagai ID Mesin, "AssetDesc" sebagai Nama Mesin, dari
[Assets]

//Memasukan Data Kondisi Mesin
Memasukan data periode perhitungan, jumlah produksi, durasi produksi,
kecepatan produksi, setup time, scheduled downtime, unscheduled downtime,
dan defect

Avaibility =  $\frac{(\text{Durasi Produksi} - \text{Scheduled Downtime} - \text{Unscheduled Downtime})}{\text{Durasi Produksi} - \text{Scheduled Downtime}} \times 100\%$ 

Quality =  $\frac{(\text{Jumlah Produksi} - \text{Defect})}{\text{Jumlah Produksi}} \times 100\%$ 

Performance =  $\frac{(\text{Kecepatan Produksi} \times \text{Jumlah Produksi})}{\text{Durasi Produksi} - \text{Setup Time} - \text{Scheduled Downtime} - \text{Unscheduled Downtime}} \times 100\%$ 

OEE =  $\left( \frac{\text{Avaibility}}{100\%} \times \frac{\text{Performance}}{100\%} \times \frac{\text{Quality}}{100\%} \right) \times 100\%$ 

//Menyimpan Data OEE
Save Data OEE, hasil perhitungan akan disimpan dengan bantuan hubungan
antara tabel [Assets] dengan [OEE]
```

Gambar 5.14 Pseudocode Proses Perhitungan OEE

5.2 IMPLEMENTASI

Langkah implementasi adalah membuat aplikasi pada tingkatan *prototype* dari spesifikasi dan konsep desain yang dirancang dengan melakukan pengembangan *database*, *module* dan *user interface* menggunakan VBA with Microsoft Access. Tahap implementasi sistem terdiri dari pengembangan aplikasi, pengujian, instalasi, serta evaluasi sistem baru. Selama pengembangan aplikasi, analis menentukan strategi desain keseluruhan dan bekerja dengan programmer untuk menyelesaikan desain, *coding*, serta pengujian. Implementasi yang ditampilkan adalah berupa tabel, *form*, *report* dan *main menu*.

5.2.1 Implementasi Database

Pengembangan *database* untuk *prototype* sistem informasi manajemen perawatan ini dibuat untuk menunjukkan relasi antar tabel yang digunakan. Perancangan *database* sistem informasi manajemen perawatan mesin ini dilakukan menggunakan Microsoft Access 2013, mulai dari pembuatan tabel dan penentuan relasi. Implementasi *database* dengan menggunakan Microsoft Access 2013 dapat dilakukan dengan cara sebagai berikut.

1. Buka Microsoft Access 2013 dan pilih *Blank desktop database*.

2. Selanjutnya pilih *Create* kemudian pilih *Table Design*
3. Isi *field name* dengan judul kolom yang dan *data type* dengan tipe data pada kolom tersebut.
4. Pilih *Save* dan beri nama tabel.
5. *Database* sudah terbentuk secara otomatis dengan tabel yang dibuat.

| Field Name | Data Type | Description (Optional) |
|-------------------------|------------|------------------------|
| AssetDescription | Short Text | |
| AssetNumber | Short Text | |
| EmployeeNo | Short Text | |
| LocationNo | Short Text | |
| AssetCatCode | Short Text | |
| Status | Number | |
| ModelNumber | Short Text | |
| SerialNumber | Short Text | |
| DateAcquired | Date/Time | |
| PurchasePrice | Number | |
| CurrentValue | Number | |
| Comments | Long Text | |
| Warranty/Contract | Number | |
| Warranty/ContractExpiry | Date/Time | |
| Contractor | Short Text | |
| Vendor | Short Text | |
| DepartmentNo | Short Text | |
| FailureCauselD | Number | |
| Criticality | Short Text | |
| DataDisposed | Date/Time | |
| NotesToTech | Long Text | |
| Manufacturer | Short Text | |

| Field Properties | |
|------------------|-----|
| Field Size | 50 |
| Format | |
| Input Mask | |
| Caption | |
| Default Value | |
| Validation Rule | |
| Validation Text | |
| Required | Yes |

Gambar 5.15 Tampilan *Table Design* Pada *Microsoft Access 2013*

Gambar 5.15 merupakan implementasi dari entitas *assets* yang telah dirancang pada Tabel 5.8. *Field name* yang diisikan harus sesuai dengan *field name* pada desain begitu pula dengan *data type*. Untuk setiap entitas seperti entitas Lokasi, Karyawan dan lainnya cara pembuatan tabel/entitas sama seperti pembuatan entitas *asset* diatas. Setelah entitas tabel yang telah ditentukan dibuat, langkah berikutnya adalah membuat atribut dari tabel tersebut dan pengisian *database* berdasarkan *group technology* yang telah dirancang di awal bab kemudian membuat relasi antar tabel entitas.

1. Tabel Mesin

Berikut ini merupakan implementasi tabel *database* untuk entitas *asset/mesin* setelah diisikan data mesin yang ada pada PT. GDS.

| AssetDescription | AssetNumber | EmployeeNo | LocationNo | AssetC. | Status | ModelNumber | Department | NotesToTech | SerialN | Date | Purc | Current | Comments | Warranty |
|-------------------|-------------|------------|------------|---------|--------|----------------------------------|------------|--|---------|------|------|---------|----------|----------|
| REHEATING FURNACE | 02SECEKU | 001 | 02 | SE | 1 | Walking Beam Type 250ton/Hr | ME | Slab Reheating Furnace of # 1 Plate Mill | | | 0 | 0 | | |
| HOT LEVELLER | 03HRERTA | | 03 | HR | 1 | 3.5 Meters Hot Plate Leveler | ME | | | | 0 | 0 | | |
| ROLLING MILL | 03MDLLZH | | 03 | MD | 1 | 4-High Roughing & Finishing Mill | ME | 4-High Roughing & Finishing Mill | | | 0 | 0 | | |
| ROLL TABLE 2Q | 03RE2QLO | | 03 | RE | 1 | | ME | | | | 0 | 0 | | |
| ROLL TABLE 2Q | 03RE2QLO | | 03 | RE | 1 | | ME | | | | 0 | 0 | | |
| ROLL TABLE A1 | 03REA1LO | | 03 | RE | 1 | | ME | | | | 0 | 0 | | |
| ROLL TABLE A2 | 03REA2LO | | 03 | RE | 1 | | ME | | | | 0 | 0 | | |

Gambar 5.16 Printscreen Tabel *Asset* Pada *Microsoft Access 2013*

Implementasi tabel mesin pada Gambar 5.16 berdasarkan dengan desain *database* fisik entitas mesin pada Tabel 5.8, terdapat atribut seperti *AssetNo*, *AssetDescription*, *LocationNo*, serta atribut lain.

2. Tabel *Sparepart*

Berikut Gambar 5.17 yang merupakan implementasi tabel *database* untuk entitas *sparepart* setelah diisi data *sparepart*.

| NamaSpareq | SparepartID | EmployeeID | EmployeeNo | LocationNo | KomponenID | NotesToTech | Stock | JumlahMin | Lokasi | Letak | NoLetak | Manufacture | Status | ModelNumit | SerialNumb | DateAcq |
|---------------|--------------|------------|------------|------------|------------|--------------|-------|-----------|-------------|-------|---------|-------------|--------|------------|------------|---------|
| BEARING 22326 | 02SECEKUBBEI | | | 02SECEKU | 02SECEKUBL | | 2 | 1 | Gudang Luar | Rak | A2 | | | | | |
| GASKET HOUSI | 02SECEKUPOGI | | | 02SECEKU | 02SECEKUPO | | 1 | 1 | Gudang Luar | Rak | A2 | | | | | |
| HOUSING TA19 | 02SECEKUPOHF | | | 02SECEKU | 02SECEKUPO | HOUSING dari | 0 | 1 | Gudang Luar | Rak | A1 | Vicker | | | | |
| SHAFT MAIN T7 | 02SECEKUPOSH | | | 02SECEKU | 02SECEKUPO | untuk meneru | 2 | 1 | Gudang Luar | Rak | A1 | Vickers | | | | |
| | | | | | | | 0 | 0 | | | | | | | | |

Gambar 5.17 Printscreen Tabel *Sparepart* Pada *Microsoft Access 2013*

Implementasi tabel *sparepart* pada Gambar 5.17 berdasarkan desain *database* fisik entitas *sparepart* pada Tabel 5.10. Atribut NamaLokasi dan Letak pada entitas *sparepart* akan dihubungkan dengan tabel lokasi serta tabel letak yang menjelaskan kode-kode tersebut

3. Tabel Jadwal *Work order*

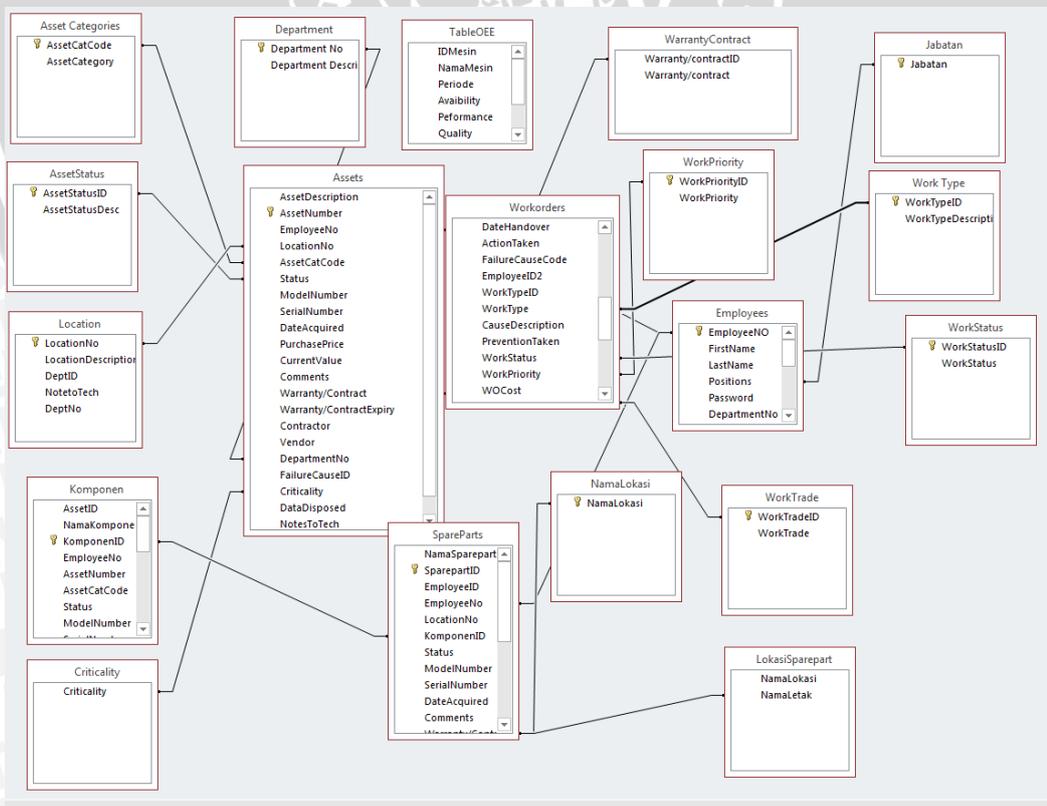
Berikut ini merupakan implementasi tabel *work order* pada Gambar 5.18 berdasarkan desain *database* fisik entitas jadwal perawatan mesin pada Tabel 5.14.

| WorkorderID | WorkorderNo | RequestorN | EmployeeNo | AssetNo | AssetDesc | Warranty/Ct | Warrant/Ct | LocationNo | ProblemDes | DateReceive | EstDateStart | EstDateEnd | ActDateStart | ActDateEnd | DateRequire | DateHando |
|--------------|-------------|------------|------------|----------|--------------|-------------|------------|------------|--------------------------------|-------------|--------------|------------|--------------|------------|-------------|-----------|
| 85 000000001 | 002 | | | 03MDLLZH | ROLLING MILL | | | 03 | * Gerinda Line 015 8:59:12 AM | 6/23/2015 | 6/24/2015 | 6/24/2015 | 6/24/2015 | 6/21/2015 | | |
| 85 000000002 | 002 | | | 03HRERTA | HOT LEVELLER | | | 03 | * Ajust Paju Bc 015 9:02:50 AM | | | | | | | |

Gambar 5.18 Printscreen Tabel *Work Order* Pada *Microsoft Access 2013*

4. Relasi antar tabel entitas

Berikut ini Gambar 5.19 merupakan relasi antar tabel entitas pada sistem informasi manajemen perawatan berbasis CMMS ini.



Gambar 5.19 Printscreen Relasi Antar Tabel Entitas

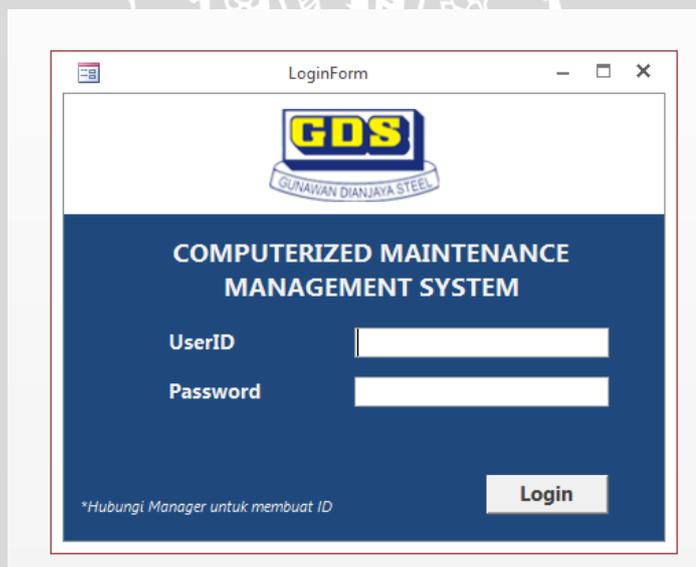
Relasi antar tabel entitas yang dibuat pada *Relationship Microsoft Access 2013* pada Gambar 5.19 merupakan implementasi dari relasi yang digambarkan oleh ERD. Setiap entitas telah berelasi sesuai dengan entitas lain yang telah dijelaskan pada Tabel 5.4 dan Gambar 5.4.

5.2.2 Implementasi *User Interface*

Implementasi *user interface* akan menggunakan *Microsoft Access 2013* yang ditujukan supaya pengguna/*user* lebih mudah untuk mengakses sistem informasi manajemen perawatan berbasis CMMS dan *group technology* ini. Implementasi ini didasarkan atas analisa dan desain yang telah dibuat pada tahapan sebelumnya. Implementasi *user interface* sistem informasi manajemen perawatan mesin sebagai berikut:

1. Login

Berikut merupakan implementasi menu login berdasarkan desain hirarki menu utama pada Gambar 5.5 yang ditunjukkan oleh Gambar 5.20.



Gambar 5.20 *Printscreen* Menu Login

Menu Login merupakan menu yang muncul sebelum SPV, administrator dan karyawan menggunakan aplikasi. Pada menu ini, manajer, SPV dan teknisi harus memasukkan *username* dan *password* yang tersedia seperti pada Gambar 5.20 untuk masuk ke form Main Menu.

2. *Form*

Form yang muncul sesaat setelah Manajer, SPV, maupun teknisi *Sign In/Log In* bergantung pada *username* dan *password* yang dimasukkan. *Form* yang akan

muncul adalah Menu Utama, yang didalamnya terdapat menu-menu yang diadaptasi dari *Computerized Maintenance Management System*. Berikut implementasi masing-masing form.

a. *Menu Utama*

Berikut Gambar 5.21 merupakan implementasi desain *form* Menu Utama pada Gambar 5.9.

Computerized Maintenance Management System PT. GDS Tbk

Thursday, August 06, 2015
12:04:52 AM

GDS PT GUNAWAN DIANJAYA STEEL, Tbk.

Login Sebagai : Manajer

PENUGASAN DAN PENJADWALAN PERAWATAN

| Work Order # | Work Type | Nama Teknisi | Tanggal Penugasan | DateRequired | Nama Mesin |
|--------------|------------|------------------|--------------------|--------------|-------------------|
| 0000000012 | Manual Pre | Budi Wicaksono | 7/9/2015 9:56:51 P | 7/13/2015 | HOT LEVELLER |
| 0000000011 | Manual Pre | Maryanto | 7/9/2015 9:54:21 P | 7/10/2015 | ROLLING MILL |
| 0000000010 | Manual Pre | Margiono | 7/9/2015 9:50:26 P | 7/10/2015 | ROLL TABLE 2Q |
| 0000000008 | Breakdown | Suparman | 7/9/2015 9:00:42 P | 7/7/2015 | ROLLING MILL |
| 0000000007 | Breakdown | Mochamad Ibrahim | 7/9/2015 8:57:59 P | 7/11/2015 | ROLL TABLE 5Q |
| 0000000006 | Breakdown | Maryanto | 7/9/2015 8:43:11 P | 7/13/2015 | ROLL TABLE 2Q |
| 0000000005 | Corrective | Inwan S | 7/9/2015 8:18:16 P | 7/13/2015 | ROLL TABLE A1 |
| 0000000004 | Breakdown | Mochamad Ibrahim | 7/9/2015 8:13:45 P | 7/11/2015 | REHEATING FURNACE |
| 0000000003 | Corrective | Margiono | 7/9/2015 8:08:42 P | 7/11/2015 | REHEATING FURNACE |
| 0000000002 | Corrective | Mochamad Ibrahim | 6/22/2015 9:02:50 | 6/22/2015 | HOT LEVELLER |

REPORTS

Laporan Departemen Maintenance PT. GDS

SETTING

Pengaturan User Account dan Data Diri

VISUAL OEE

| IDMesin | Nama Mesin | Periode | Availability | Performance | Quality | OEE | Target |
|----------|--------------|---------|--------------|-------------|-------------|-------------|--------|
| 03MDLLZH | ROLLING MILL | 1 | 92.65625 | 77.97976391 | 95.64032697 | 69.103125 | 80 |
| 03MDLLZH | ROLLING MILL | 2 | 100 | 80.42553191 | 92.85714285 | 74.68085106 | 80 |
| 03MDLLZH | ROLLING MILL | 3 | 100 | 84.2212355 | 90.993111 | 73.44226382 | 80 |

EXIT

Cek Tindakan Hari Ini

Urutkan Data Berdasarkan Mesin

Urutkan Data Berdasarkan Periode

Copyright © 2015 Reza Maulana @ IEBU - PT. Gunawan Dianjaya Steel Tbk

Gambar 5.21 *Printscreen Main Menu*

b. *Form Work Order*

Berikut Gambar 5.22 merupakan implementasi desain *form* *Work Order* pada Gambar 5.10.

Work Order List

| Work Order # | Work Type | Work Status | AssetNo | Problem Description | Nama Teknisi | Action |
|--------------|-------------------|-------------------|----------|--------------------------------------|------------------|--------|
| 000000012 | Manual Preventive | Open | 03HRERTA | Pengecekan paju Back Up Roll Ho | Budi Wicaksono | |
| 000000011 | Manual Preventive | Open | 03MDLLZH | Pengecekan baut – baut Screw Dov | Maryanto | |
| 000000010 | Manual Preventive | Open | 03RE2QPA | Pengecekan Gear box dan Roll Tal | Margiono | |
| 000000008 | Breakdown | Open | 03MDLLZH | * Ganti Work Roll Mill Atas dan Ba | Suparman | |
| 000000007 | Breakdown | Open | 03RE5QPA | * Ganti Penggerak Gear V.Z 26 Roll | Mochamad Ibrahim | |
| 000000006 | Breakdown | Open | 03RE2QPA | - Ganti Roller Bearing 224 yg Prot | Maryanto | |
| 000000005 | Corrective | Open | 03REA1PA | * Las tutup motor Roll Table A1 & | Irwan S | |
| 000000004 | Breakdown | Open | 02SECEKU | * Ganti Bearing 22326 As Blower. | Mochamad Ibrahim | |
| 000000003 | Corrective | Open | 02SECEKU | * Ganti Hydroulis Extractor Furnac | Margiono | |
| 000000002 | Corrective | Waiting for Parts | 03HRERTA | * Adjust Paju Box Back Up Roll Hot l | Mochamad Ibrahim | |

Total Data **11** **PM Baru** **WorkOrder Baru** **Print WO** **Lihat Semua** **Export to Excel** **Close**

Cari Work Order | Sorting Work Order | Tutorial Form Work Order

Work Type: Masalah:

Work Status: Lokasi:

Work Trade: PM No.:

Workorder No: PM Task No.:

Nama PIC: Received Date: To: M:

Mesin: Est Start Date: To: M:

Prioritas WO: Act.Close Date: To: M:

Failure Code: PM Target Start Date: To: M:

Find => **Cari** **Reset**

Gambar 5.22 Printscreen Form Maintenance

c. *Form Manajemen Mesin*

Berikut Gambar 5.23 merupakan implementasi desain form Manajemen Mesin pada Gambar 5.11.

MANAJEMEN ASET

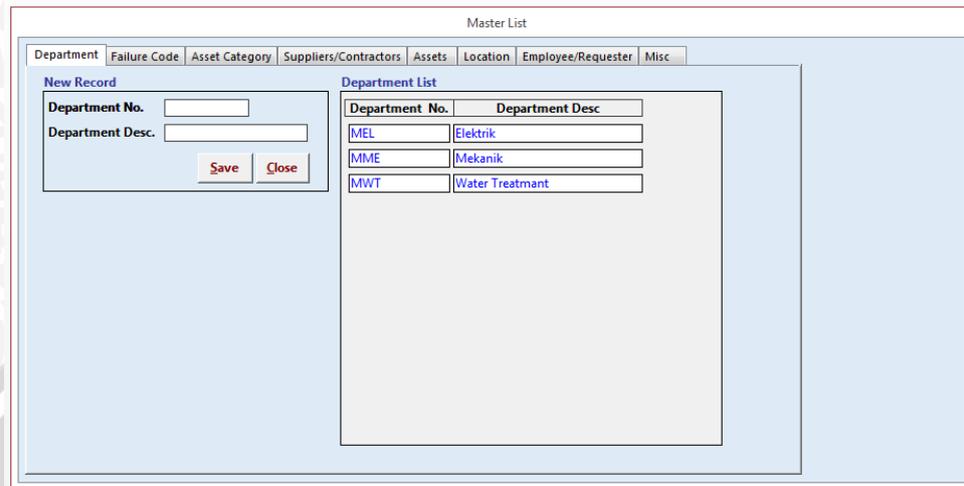
| MESIN | KOMPONEN | SPAREPART | OEE MESIN |
|------------|-------------------|--------------|-----------|
| ID Mesin # | Nama Mesin | Status Mesin | |
| 02SECEKU | REHEATING FURNACE | Active | |
| 03REA1PA | ROLL TABLE A1 | Active | |
| 03REA2PA | ROLL TABLE A2 | Active | |
| 03RE2QPA | ROLL TABLE 2Q | Active | |
| 03RE9QPA | ROLL TABLE 9Q | Active | |
| 03MDLLZH | ROLLING MILL | Active | |
| 03HRERTA | HOT LEVELLER | Active | |
| 03RE4QPA | ROLL TABLE 4Q | Active | |
| 03RE5QPA | ROLL TABLE 5Q | Active | |
| 04CEDAL | COOLING BED | Active | |
| 03RE6QPA | ROLL TABLE 6Q | Active | |
| 03RE7QPA | ROLL TABLE 7Q | Active | |

ID Mesin
 Nama Mesin
 Status Mesin

Gambar 5.23 Printscreen Form Manajemen Aset

d. *Form Master List*

Berikut Gambar 5.24 merupakan implementasi desain *form Master List* pada Gambar 5.11.



| Department No. | Department Desc. |
|----------------|------------------|
| MEL | Elektrik |
| MME | Mekanik |
| MWT | Water Treatment |

Gambar 5.24 Printscreen Form Master List

3. *Report*

Report merupakan laporan informasi data yang dibutuhkan oleh *user* baik manajer, SPV maupun teknisi. Pada sistem informasi manajemen perawatan mesin terdapat 3 *report* utama yaitu *report sparepart*, *report* perhitungan OEE serta *report* jadwal perawatan mesin. Berikut ini adalah implentasi dari *report* pada sistem informasi yang dirancang.

a. *Report Spaprepart*

Berikut Gambar 5.25 yang merupakan implementasi *report sparepart*.



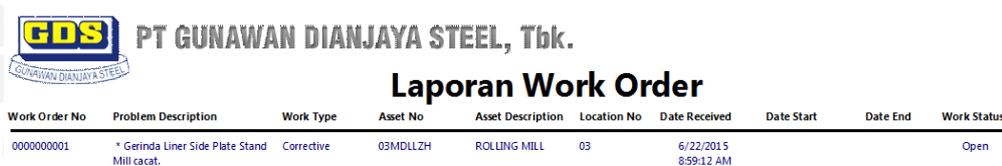
| Nama Mesin | Komponen | Kode Sparepart | Nama Sparepart | Merk | Jumlah Min | Stock | Satuan | Lokasi | Letak | Nomor |
|-------------------|-------------------------|------------------|-----------------------|------|------------|-------|--------|-------------|-------|-------|
| REHEATING FURNACE | POMPA BAHAN BAKAR A & B | 02SECEKUPOGALRA2 | GASKET HOUSING MFE 19 | | 1 | 1 | unit | Gudang Luar | Rak | A2 |
| | | 02SECEKUPHOLRA1 | HOUSING TAL1919 | | 1 | 0 | unit | Gudang Luar | Rak | A1 |
| | | 02SECEKUPOSH1RA1 | SHAFT MAIN TAL1919 | | 1 | 2 | unit | Gudang Luar | Rak | A1 |

Gambar 5.25 Print Preview Report dari Sparepart

Berdasarkan gambar di atas, pada *report sparepart* dapat diketahui informasi mengenai *inventory sparepart* dalam setiap mesin. *User* dapat menggunakan *report inventory sparepart* untuk melihat informasi *sparepart* mesin yang dimiliki PT. GDS.

b. *Report Work order*

Berikut Gambar 5.26 merupakan implementasi *report* dari *Work order*.



| Work Order No | Problem Description | Work Type | Asset No | Asset Description | Location No | Date Received | Date Start | Date End | Work Status |
|---------------|--|------------|----------|-------------------|-------------|----------------------|------------|----------|-------------|
| 000000001 | * Gerinda Liner Side Plate Stand Mill cacat. | Corrective | 03MDLLZH | ROLLING MILL | 03 | 6/22/2015 8:59:12 AM | | | Open |

Gambar 5.26 Print Preview Report dari *Work Order*

Berdasarkan Gambar 5.26, dengan *report work order* ini SPV maupun manajer dapat melihat *work order* yang telah dilakukan maupun belum dilakukan.

c. *Report* Perhitungan OEE

Berikut Gambar 5.27 adalah implementasi *report* perhitungan OEE.



| Nama Mesin | ID Mesin | ID OEE | Periode OEE | Availability | Performance | Quality | OEE | Goal | |
|--------------|----------|----------|-------------|--------------|-------------|-------------|-----------|-----------|----|
| ROLLING MILL | 03MDLLZH | 03MDLLZH | | 1 | 92.65625 | 77.97976391 | 95.640327 | 69.103125 | 80 |

Gambar 5.27 Print Preview Report Perhitungan OEE

Berdasarkan Gambar 5.27, dengan *report* perhitunga OEE SPV maupun manajer dapat melihat mesin yang mengalami penurunan performansi maupun mesin yang tidak mencapai target OEE yang ditetapkan. Dengan begitu, SPV maupun manajer akan mendapatkan rekomendasi prioritas perawatan untuk kegiatan *preventive maintenance* selanjutnya.

5.2.3 Implementasi Modul Program

Berikut ini merupakan contoh beberapa penggalan *source code* sesuai dengan desain algoritma proses yang telah disusun sebelumnya menggunakan VBA *with Microsoft Access* 2013.

1. Memasukkan dan menyimpan data

Berikut Gambar 5.28 merupakan implementasi modul program berdasarkan desain algoritma yang telah disusun untuk menyimpan data, contoh *syntax* berikut adalah *syntax* memasukkan dan menyimpan data mesin.

```

Private Sub Save_Click() `menyimpan data mesin
Dim Msg, Style, title, Help, Ctxt, response, MyString
Msg = "Do you want to save this record ?"
Style = vbYesNo + vbInformation + vbDefaultButton2
title = "Confirm Save"
IDMesin.Value = Left(Description, 2) + Right(Description, 2)
ManfID.Value = Left(Manf, 2)
AssetNo.Value = LocationNo.Value + AssetCategory + IDMesin.Value +
ManfID.Value `pembuatan kode mesin

If IsNull(LocationNo) Or Trim(LocationNo) = "" Then
MsgBox "Please provide data to the field", vbInformation
DoCmd.GoToControl "LocationNo"
ElseIf IsNull(departNo) Or Trim(departNo) = "" Then
MsgBox "Please provide data to the field", vbInformation
DoCmd.GoToControl "departNo"
Else
response = MsgBox(Msg, Style, title)
If response = vbYes Then
Set dbcW = CurrentDb()
Set rsasset = dbcW.OpenRecordset("Assets")
With rsasset
.AddNew `menyimpan data pada tabel mesin
! [AssetNumber] = Me.AssetNo
! [AssetDescription] = Me.Description
! [LocationNo] = Me.LocationNo
! [EmployeeNo] = Me.employeno
! [DepartmentNo] = Me.departNo
! [AssetCatCode] = Me.AssetCategory.Column(0)
! [Status] = Me.Status.Column(0)
! [Criticality] = Me.Criticality
! [Warranty/Contract] = Me.WarrantyContract.Column(0)
! [Warranty/ContractExpiry] = Me.WarrContExpiry
! [Contractor] = Me.Contractor.Column(0)
! [Comments] = Me.Asset_Notes
! [Manufacturer] = Me.Manf
! [ModelNumber] = Me.Model Number
! [SerialNumber] = Me.Serial Number
! [Vendor] = Me.Vendor
! [PurchasePrice] = Me.PurchasePrice
! [CurrentValue] = Me.CurrentValue
! [DateAcquired] = Me.DateAcquired
! [DataDisposed] = Me.DataDisposed
! [NotesToTech] = Me.NotesToTech
.Update
.Close
End With
DoCmd.Close
DoCmd.OpenForm "NewAssetForm", , , stLinkCriteria
Else
DoCmd.GoToControl "AssetNo"
End If
End If

End Sub

```

Gambar 5.28 Syntax Proses Memasukkan dan Menyimpan Data Mesin

2. Perhitungan OEE

Berikut ini merupakan implementasi modul program berdasarkan desain algoritma yang telah disusun untuk perhitungan nilai OEE Mesin.

```

Option Compare Database

Private Sub Command58_Click() 'Hitung OEE

AR.Value = ((Produksi.value - SD.Value - UD.Value) / (Produksi.Value) -
(SD.Value)) * 100 'hitung avaibility
QR.Value = ((Produk.Value - Defect.Value) / Produk.Value) * 100 'hitung Quality
PR.Value = ((KecProduksi.Value * Produk.Value) / (Durasi.Value - Setup.Value
- SD.Value - UD.Value)) * 100 'hitung performance
OEE.Value = ((AR.value / 100) * (QR.Value / 100) * (PR.Value / 100))*100
'hitung OEE

End Sub

Private Sub Save_Click() 'simpan data pada tabel

Dim Msg, Style, title, Help, Ctxt, response, MyString
Dim rsworder As DAO.Recordset
Msg = "Do you want to save the record ?"
Style = vbYesNo + vbInformation + vbDefaultButton2
title = "Confirm Save"
response = MsgBox(Msg, Style, title)
If response = vbYes Then
    Set dbCW = CurrentDb()
    Set rsworder = dbCW.OpenRecordset("TableOEE")
    With rsworder
        .AddNew
        ![IDMesin] = Me.IDMesin
        ![Periode] = Me.ContName
        ![Quality] = Me.QR
        ![Avaibility] = Me.AR
        ![Performance] = Me.PR
        ![OEE] = Me.OEE
        ![IDOEE] = Me.IDOEE

        .Update
        .Close
    End With
Else
    'DoCmd.GoToControl "WorkorderNo"
End If

End Sub

```

Gambar 5.29 Syntax Proses Perhitungan OEE

5.3 PENGUJIAN

Tahapan terakhir dari perancangan ini setelah sistem sudah menjadi *prototype* adalah pengujian. Langkah ini adalah langkah yang penting karena berguna untuk melihat apakah *prototype* yang telah dibuat sudah sesuai dengan harapan atau tidak. Tahap pengujian ini ditinjau dari tiga segi, yaitu uji verifikasi, uji validasi dan uji *prototype* yang masing-masing terdapat tujuan yang saling terhubung.

5.3.1 Verifikasi

Verifikasi adalah tahap pengujian yang akan menguji apakah *prototype* yang telah dirancang sudah berjalan sesuai dengan rencana. Uji verifikasi akan dilakukan dengan cara membandingkan antara desain *database*, *user interface*, modul program pada tahap desain dengan implementasi dan ketelitian program aplikasi. Selain itu, verifikasi juga

akan dilakukan dengan membandingkan hasil perhitungan OEE secara manual dengan perhitungan OEE menggunakan sistem informasi yang telah dirancang.

Perbandingan model *database* yang telah dirancang pada Tabel 5.18 sampai 5.27 dengan implementasi pengembangan *database* pada Gambar 5.16 sampai 5.19 sudah sesuai. Untuk perbandingan *pseudocode* yang telah dibuat dengan *listing program* dapat ditunjukkan dengan *pseudocode* pada Gambar 5.13 hingga 5.14, sedangkan untuk implementasi *listing program* dapat dilihat pada Gambar 5.27 sampai Gambar 5.28. Perbandingan untuk rancangan hierarki menu seperti yang ditunjukkan pada Gambar 5.5 diimplementasikan oleh Gambar 5.20. Berdasarkan dari Gambar 5.20, implementasi hierarki menu telah sesuai dengan desain hierarki menu yang dirancang yaitu *user* akan memasukkan *username* dan *password* untuk masuk ke dalam masing-masing *form* sesuai dengan *Sign In/Log In* sebelum dapat mengakses aplikasi sistem informasi. Begitu juga hierarki menu yang terdapat pada *form Maintenance*, Asset Management, Master Data serta Report, rancangan hierarki menu dan *user interface* pada Gambar 5.6 hingga 5.11 telah diinterpretasikan dengan Gambar 5.21 hingga 5.23.

Untuk pengujian desain *report* dilakukan dengan cara melihat apakah periode *report* yang dihasilkan sesuai dengan periode yang dipilih. Berikut ini adalah hasil uji verifikasi *report. Summary report* mengenai *work order* yang diinginkan oleh *user* pada Gambar 5.29 berikut akan ditampilkan *report* pada Gambar 5.30.

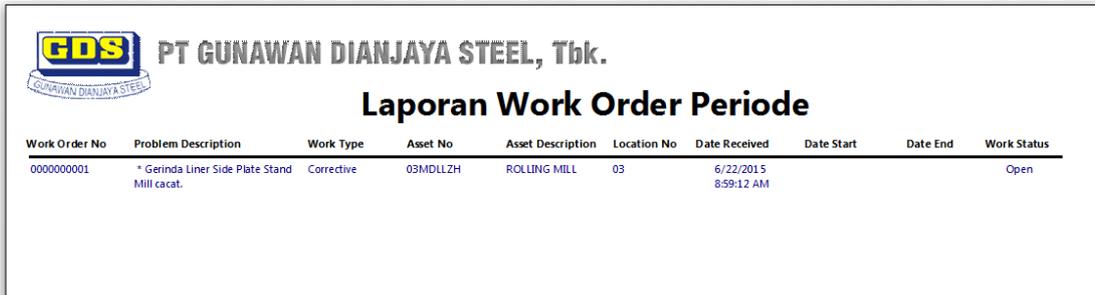
The screenshot shows a web application window titled "Reports" with a sub-header "REPORT DEPARTEMEN MAINTENANCE PT. GDS Tbk". Below the header, there is a prompt: "Pilih jenis laporan yang dibutuhkan!".

There are two main sections for report selection:

- Laporan Rekap Data:** This section contains six buttons: "Rekap Data Mesin", "Rekap Data Detail Mesin", "Rekap Data Sparepart", "Chart Work Order Status", "Rekap Data List Work Order", and "Rekap Data Work Order".
- Laporan Spesifik:** This section is divided into two parts. The top part has input fields for "Dari" (6/1/2015), "Hingga" (6/25/2015), and "Nama Mesin" (with a dropdown arrow). To the right are two buttons: "Laporan Periodik WO Mesin Tertentu" and "Laporan Periodik WO Semua Mesin". The bottom part has a "Nama Mesin" dropdown and two buttons: "Laporan OEE Mesin Tertentu" and "Laporan OEE Semua Mesin".

A "Close" button is located at the bottom right of the form.

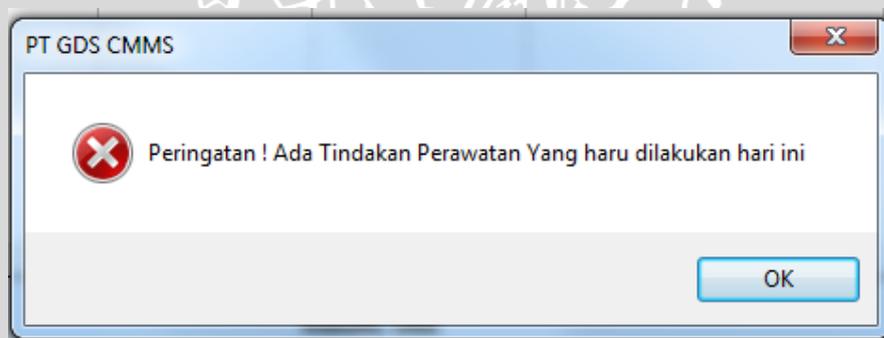
Gambar 5.30 Memilih Periode Report Summary Work Order



| Work Order No | Problem Description | Work Type | Asset No | Asset Description | Location No | Date Received | Date Start | Date End | Work Status |
|---------------|--|------------|----------|-------------------|-------------|----------------------|------------|----------|-------------|
| 000000001 | * Gerinda Liner Side Plate Stand Mill cacat. | Corrective | 03MDLLZH | ROLLING MILL | 03 | 6/22/2015 8:59:12 AM | | | Open |

Gambar 5.31 Print Preview Hasil Report Summary Work Order

Berdasarkan Gambar 5.31, dapat diketahui bahwa hasil periode *report* yang ingin ditampilkan sudah sesuai yang diinginkan oleh *user*. Hal ini membuktikan bahwa *summary* sudah sesuai dan sistem telah dapat menghasilkan *report* yang diinginkan oleh *user*. Pengujian selanjutnya adalah menguji ketelitian terhadap alert atau peringatan mengenai *work order* yang harus dilakukan pada hari itu. Dari Gambar 5.31, terdapat 3 *work order* yang harus dilakukan pada hari tersebut, kemudian sistem akan memberi peringatan yang ditunjukkan pada Gambar 5.32 mengenai *work order* yang harus dilakukan pada hari tersebut. Hal ini menunjukkan bahwa sistem sudah memberikan peringatan yang sesuai dengan data yang ada dan dapat memberikan informasi yang tepat.



Gambar 5.32 Peringatan Tindakan Perawatan

Berikutnya adalah pengujian verifikasi terhadap perhitungan nilai OEE yang dihasilkan oleh sistem informasi terhadap nilai perhitungan yang dilakukan secara manual. Sebagai contoh adalah perhitungan OEE terhadap mesin Hot Rolling Mill pada periode pertama tahun 2015. Berikut ini adalah data mengenai kondisi mesin Hot Rolling Mill pada periode pertama tahun 2015.

| | |
|-------------------------------|------------|
| Periode Produksi | 1 |
| Durasi produksi (menit) | 5760 menit |
| Jumlah produk yang diproduksi | 734 unit |

| | |
|----------------------|---------------------|
| Kecepatan produksi | 5,67 menit / produk |
| Setup time | 0 |
| Scheduled Downtime | 0 |
| Unscheduled Downtime | 423 menit |
| Defect | 32 unit |

Berikut adalah hasil perhitungan manual untuk mesin Rolling Mill pada periode 1:

$$\text{Availability rate} = \frac{\text{loading time} - \text{downtime}}{\text{loading time}} \times 100 \%$$

$$\begin{aligned} \text{Availability rate} &= \frac{5760 - 423}{5760} \times 100\% \\ &= \frac{5337}{5760} \times 100\% \\ &= 92,65 \% \end{aligned}$$

$$\text{Performance rate} = \frac{\text{process amount} \times \text{actual time}}{\text{process time}} \times 100 \%$$

$$\text{Performance rate} = \frac{734 \times 5,67}{5337} \times 100 \%$$

$$\text{Performance rate} = \frac{4161,78}{5337} \times 100 \%$$

$$\text{Performance rate} = 77,98 \%$$

$$\text{Quality Rate} = \frac{\text{process amount} - \text{defect amount}}{\text{process amount}}$$

$$\text{Quality Rate} = \frac{734 - 32}{734} \times 100\%$$

$$\text{Quality Rate} = \frac{702}{734} \times 100\%$$

$$\text{Quality Rate} = 95,64 \%$$

Sedangkan untuk perhitungan menggunakan aplikasi sistem informasi manajemen perawatan dapat ditunjukkan pada Gambar 5.33 Hal ini menunjukkan bahwa sistem perhitungan OEE sudah sesuai dengan hasil perhitungan dengan manual.

Gambar 5.33 Perhitungan OEE

5.3.2 Validasi

Pada tahap validasi ini, akan diuji kemampuan mempresentasikan tujuan awal dan memenuhi kebutuhan sehingga apakah program yang dibuat sudah sesuai dengan fungsi yang diharapkan. Uji Validasi dilakukan dengan menguji coba aplikasi sistem informasi ini kepada seluruh *user*, yaitu manajer, SPV, dan teknisi.

Tabel 5.28 Uji Validasi

| Pengguna | Kebutuhan yang Terpenuhi |
|----------|---|
| Manajer | Dapat mengelolah data mengenai mesin, komponen, <i>sparepart</i> serta karyawan di dalam sistem informasi yang dirancang. |
| | Manajer dapat memonitoring seluruh kegiatan perawatan di <i>departemen maintenance</i> . |
| | Manajer dapat memperoleh <i>report</i> hasil rekap data mesin, komponen, <i>sparepart</i> , karyawan, <i>work order</i> serta perhitungan OEE mesin. |
| SPV | SPV dapat melakukan penugasan serta penjadwalan perawatan mesin kepada teknisi dan memonitoring seluruh kegiatan perawatan di <i>departemen maintenance</i> . |
| | Sistem mampu memberikan informasi terhadap stock inventory <i>sparepart</i> . |
| | Sistem mampu memberikan laporan rutin harian dan <i>summary</i> kepada SPV berupa laporan <i>inventory sparepart</i> , dan laporan <i>work order</i> . |
| Teknisi | Sistem dapat menampilkan penugasan dan penjadwalan yang harus dilakukan teknisi dalam bentuk <i>work order</i> setiap hari. |
| | Sistem mampu menyajikan data lokasi dan jumlah <i>sparepart</i> serta data <i>history</i> mesin yang dibutuhkan karyawan. |

Dari hasil uji validasi Tabel 5.28, *prototype* sistem informasi yang telah dirancang sudah dapat memenuhi kebutuhan sistem yang dibutuhkan oleh manajer, SPV maupun teknisi. *Prototype* sistem informasi manajemen perawatan ini dapat memberikan laporan

yang dibutuhkan oleh Manajer maupun SPV. Kebutuhan *user* dalam mengelola data-data untuk manajemen perawatan mesin sudah terpenuhi dengan adanya *database* yang mengintegrasikan seluruh data dan mengolah data sehingga dapat menyajikan informasi yang dibutuhkan. Sedangkan untuk teknisi, *prototype* sistem informasi manajemen perawatan ini sudah memenuhi kebutuhan yang diperlukan yaitu teknisi dapat melihat jadwal penugasan untuk masing-masing teknisi setiap harinya. Selain itu, sistem sudah dilengkapi sistem keamanan untuk masing-masing *user* dengan adanya *username* dan *password* yang berbeda untuk membatasi hak akses sesuai dengan jabatannya.

5.3.3 Uji Prototype

Pada tahap uji *prototype* ini bertujuan untuk mengetahui apakah *prototype* dapat mengatasi masalah serta kelemahan sistem sebelumnya yang telah ditulis pada Bab 1. Berikut pada Tabel 5.29 adalah hasil dari uji *prototype*.

Tabel 5.29 Uji Prototype

| Jenis Analisis | Sistem Lama | Sistem Baru |
|----------------|--|---|
| Performance | Pengumpulan informasi data untuk tindakan perawatan mesin memerlukan waktu lama. | Sistem baru sudah menggunakan <i>database</i> dalam pengolahan data yang berbasis <i>group technology</i> untuk mengelompokkan mesin, komponen serta <i>sparepart</i> sehingga waktu untuk pengumpulan data lebih rendah. |
| | Sulitnya menelusuri informasi yang dibutuhkan oleh SPV tentang data semua mesin, jadwal perawatan mesin dan kerusakan mesin serta tindakan harian. | Data yang dibutuhkan oleh manajer maupun SPV sudah terintegrasi dengan baik sehingga mudah dalam menelusuri informasi yang dibutuhkan. |
| | Belum adanya pengukuran peformansi dan efektivitas mesin. | Tersedia sistem pengukuran OEE yang di dalamnya terdapat penguruan peformansi dan efektivitas mesin. |
| Information | Sistem informasi tidak mudah diakses oleh SPV dan karyawan <i>maintenance</i> karena pencarian data mesin, karyawan, <i>sparepart</i> maupun jadwal perawatan mesin masih manual dan lisan serta tidak terorganisir. | Sistem informasi dapat diakses dengan mudah oleh SPV maupun teknisi <i>maintenance</i> serta dapat mencari informasi mengenai mesin, karyawan, <i>sparepart</i> serta jadwal perawat secara terorganisir. |
| | Data yang tersimpan belum terintegrasi antara data yang satu dengan data lainnya berakibat pada kemungkinan terjadi redundansi data. | Data yang tersimpan sudah terintegrasi antara satu dengan yang lain melalui <i>Entity Relationship Diagram</i> (ERD) |
| | Kemungkinan kehilangan data serta kesalahan pencatatan akibat kelalalian atau <i>human error</i> tinggi. | Proses pencatatan sudah terkomputerisasi dengan baik serta dilengkapi dengan menu drop-down yang mengurangi <i>human error</i> ketika memasukan data. |
| Economy | Biaya operasional dan waktu yang dibutuhkan besar karena terdapat biaya administrasi untuk pembuatan dokumen dengan kertas dan penyimpanan secara manual. | Biaya operasional dan waktu yang dibutuhkan menurun karena data atau informasi tidak harus dicetak. |

Tabel 5.29 Uji *Prototype* (Lanjutan)

| Jenis Analisis | Sistem Lama | Sistem Baru |
|-------------------|--|---|
| <i>Control</i> | Keamanan data tidak terjamin dikarenakan data disimpan secara manual sehingga rentan akan hilang. | Keamanan data terjamin karena disimpan di server secara otomatis sehingga kemungkinan data hilang berkurang |
| | Belum adanya sistem yang mengatur data sehingga kemungkinan besar dapat terjadinya kesalahan informasi tentang pengambilan keputusan tindakan perawatan mesin. | Sistem informasi yang baru dapat mengatur data serta dapat menyediakan laporan/report semua data agar dapat dimanfaatkan <i>user</i> untuk mengambil keputusan berdasarkan data yang ada. |
| <i>Efficiency</i> | Belum memanfaatkan teknologi yang ada, data masih dicatat secara manual dan penyimpanan data secara manual membutuhkan tempat penyimpanan. | Sistem informasi ini sudah memanfaatkan komputer serta software terbaru untuk melakukan pencatatan serta penyimpanan data. |
| | Waktu pencarian <i>sparepart</i> lama yang berakibat memperlambat kinerja proses. | <i>Database sparepart</i> sistem memanfaatkan <i>GT</i> dalam pengelompokan data <i>sparepart</i> . Proses pencarian data tersebut sudah terotomasi sehingga tidak perlu mencari manual. |
| <i>Services</i> | Pelayanan akan kebutuhan informasi dari beberapa data berlangsung lama karena data belum terintegrasi menjadi satu sehingga sistem tidak fleksibel. | Pelayanan akan kebutuhan informasi semua data yang dibutuhkan lebih cepat karena data sudah terintegrasi menjadi satu sehingga sistem lebih fleksibel. |

Berdasarkan Tabel 5.17 di atas, dapat diketahui bahwa sistem informasi yang telah dirancang sudah dapat memperbaiki semua kelemahan sistem lama dari segi *performance, information, economy, control, efficiency* dan *service* yang di sajikan dalam tabel PIECES. Sistem informasi manajemen perawatan juga dapat melakukan pencarian data yang sudah terotomasi dengan cepat karena mesin, komponen dan *sparepart* sudah dikelompokkan dan dikodekan menggunakan *group technology* dalam *database*. Dengan menggunakan *database*, data-data yang ada di departemen *maintenance* dapat terorganisir dengan baik serta tidak diperlukan lagi pencatatan dokumen secara manual dan penyimpanan dokumen sehingga biaya operasional dan waktu dapat diminimalisir. Sistem informasi manajemen perawatan juga dapat memberikan informasi kepada SPV berupa *report* data secara keseluruhan sehingga SPV maupun Manajer nantinya dapat dengan mudah membuat kebijakan-kebijakan di departemen *maintenance* mendatang.

5.4 Prototype Sistem Informasi Manajemen Perawatan Berbasis *Computerized Maintenance Management System* dan *Group Technology*

Prototype sistem informasi yang telah dirancang telah mengintegrasikan beberapa data yang kompleks dalam sistem manajemen perawatan mesin berbasis CMMS di PT.GDS seperti data mesin, komponen, *sparepart*, jadwal perawatan mesin, dan teknisi

dengan menggunakan *database*. perancangan *database* menggunakan konsep GT untuk mengklasifikasikan serta mengodekan mesin, komponen dan *sparepart* agar proses pencarian data lebih mudah. Dengan memanfaatkan modul dari CMMS serta mengintegrasikan semua data yang berhubungan dengan proses manajemen perawatan di departemen maintenance dapat membantu *user* pengguna sistem informasi. Di tingkat manajerial yaitu manajer dan SPV, sistem informasi ini mampu memberikan laporan yang dibutuhkan oleh manajer maupun SPV yaitu laporan harian dan laporan rutin tentang jadwal perawatan mesin dan *inventory sparepart* yang dapat membantu manajer maupun SPV dalam menentukan kebijakan serta tindakan yang berhubungan dengan perawatan mesin ke depannya. Selain itu, sistem informasi manajemen perawatan ini dapat membantu manajer dan SPV untuk mengukur performansi mesin menggunakan OEE yang berguna untuk menentukan kegiatan perawatan yang tepat. SPV juga dapat mengontrol tindakan perawatan mesin yang dilakukan oleh teknisi agar tidak terjadi keterlambatan.

Sistem informasi dapat membantu manajer maupun SPV maupun manajer untuk mengelola data yang berhubungan dengan manajemen secara mudah. Data mesin, komponen, *sparepart*, karyawan, history perawatan mesin serta jadwal perawatan mesin tidak lagi dicatat secara manual di kertas maupun buku, melainkan dimasukkan ke dalam sistem informasi manajemen perawatan berbasis CMMS ini. Data-data tersebut kemudian direlasikan dan diolah sehingga sistem dapat menyajikan informasi yang dibutuhkan oleh *user* baik itu teknisi, maupun SPV serta manajer itu sendiri. Untuk teknisi, sistem informasi manajemen perawatan ini dapat memberikan jadwal perawatan mesin yang harus dikerjakan oleh teknisi setiap hari. Teknisi juga dapat melakukan pencarian informasi *sparepart* dengan mudah yaitu memasukkan kode maupun nama atau kode mesin dari *sparepart* tersebut. Dari pengujian yang telah dilakukan, seluruh fitur yang ada pada sistem informasi manajemen perawatan ini dapat digunakan sesuai fungsinya. Selain itu, sistem informasi manajemen perawatan juga dapat memenuhi semua kebutuhan dari *user* dan dapat memperbaiki sistem lama. Sistem informasi manajemen perawatan mesin merupakan sistem yang efektif dan efisien dalam membantu PT. GDS pada proses manajemen perawatan mesin.

BAB VI PENUTUP

Pada bagian penutup ini berisi kesimpulan yang diuraikan dalam pembahasan dan saran yang disampaikan berdasarkan pengalaman dan pertimbangan terkait dengan penelitian yang telah dilakukan.

6.1 KESIMPULAN

Berdasarkan penelitian yang telah dilakukan, dapat ditarik beberapa kesimpulan antara lain:

1. Desain *database* sistem informasi manajemen perawatan berbasis *Computerized Maintenance Management System (CMMS)* dan *Group Technology (GT)* telah dirancang dengan menggunakan sistem kodefikasi *hybrid*. Di dalam kodefikasi *hybrid* terdapat kodefikasi hierarki mesin serta kodefikasi atribut identitas mesin dan kodefikasi lokasi maupun letak *sparepart*.
2. Rancangan *prototype* sistem informasi manajemen perawatan *prototype* menggunakan *software Microsoft Access 2013* dengan implementasi dari desain yang dibuat serta memenuhi kebutuhan sistem. Berikut ini adalah implementasi desain serta pemenuhan kebutuhan sistem.
 - a. Implementasi *user interface* dengan melakukan perancangan *form* sesuai dengan desain *user interface*. Pada sistem informasi manajemen perawatan terdapat 1 *form* menu utama yang mencakup 4 *form* yaitu *form maintenance*, *form manajemen aset*, *master data*, serta *reports*.
 - b. Implementasi modul program yang dilakukan dengan cara membuat *source code* menggunakan bahasa *Visual Basic for Application (VBA)*. *Source code* yang dibuat disesuaikan dari desain algoritma yang telah dirancang.
 - c. Implementasi modul *Computerized Maintenance Management System (CMMS)* yaitu modul *Work Order*, *Employee*, *Asset Management*, dan *Inventory Control* ke dalam modul sistem informasi berdasarkan spesifikasi sistem yang dirumuskan dari *System Requirement Checklist (SRC)*.
3. Pengujian *prototype* sistem informasi perawatan yang dirancang untuk PT. Gunawan Dianjaya Steel dilakukan menggunakan 3 tahap uji yaitu uji verifikasi, uji validasi, dan uji *prototype*. Dalam uji verifikasi, implementasi *database* telah sesuai dengan desain *database*, *listing program* yang dibuat telah sesuai dengan

pseudocode, serta *user interface* sesuai dengan desain rancangan hierarki menu. Untuk tahap uji validasi, sistem informasi manajemen perawatan yang dirancang dapat memenuhi semua kebutuhan sistem sesuai dengan spesifikasi pada SRC. Sedangkan pada tahap uji validasi, sistem informasi manajemen perawatan dapat memperbaiki sistem manajemen perawatan yang lama. *Prototype* sistem informasi yang dirancang dapat memberikan informasi-informasi yang berkaitan dengan kegiatan manajemen perawatan mesin seperti penjadwalan dan penugasan perawatan. Selain itu, kodefikasi mesin, komponen dan *sparepart* yang dirancang dalam pembuatan *database* dapat mempermudah pencarian informasi yang dibutuhkan.

6.2 SARAN

Dari penelitian yang telah dilakukan, terdapat beberapa saran yang dapat diberikan untuk mendukung pengembangan *prototype* sistem informasi manajemen perawatan mesin ini lebih lanjut, antara lain:

1. Diharapkan adaya penambahan fitur-fitur lain serta pengolahan data dalam pengembangan *prototype* selanjutnya yang dapat menunjang kegiatan perawatan mesin di departemen *maintenance* agar lebih baik.
2. Diharapkan adanya pengembangan *prototype* yang lebih luas agar *prototype* dapat digunakan secara *universal* pada departemen *maintenance* perusahaan lain.

DAFTAR PUSTAKA

- Agarwal, M., Ali K. Kamrani & Hamid R.P. 1994. *An Autommated coding and Classification System with Supporting database for Effective Design of Manufacturing Systems*. Journal of Intelligent Manufacturing, 5, 235-249.
- Asyari, Daryus. 2007. *Manajemen Pemeliharaan Mesin*. Jakarta: Jurusan Teknik Mesin, Fakultas Teknik Universitas Drama Persada.
- Bagadia, Kris. 2008. *Computerized Maintenance Management System Made Easy*. New York: McGraw-Hill Companies, Inc
- Billo, R.E., Ruker, R. & Shunk, D.L. 1987. *Integration of a group technology classification and coding system with an engineering database*. Journal of Manufacturing Systems, 6(1), 37-45.
- Burbridge, J.L. 1993. *The Introduction of Group Technology*. New York: Proceeding of an International Seminar On Group Technology
- Chan, F.T.S., Lau, H.C.W., Ip, R.W.L., Chan, H.K., and Kong, S. 2003. *Implementation of Total Productive Maintenance : A Case Study*. International Journal Production Economics 95 (2005), 71-94.
- Chang, T., Wysk, R.A. & Wang, H. 1991. *Computer Aided Manufacturing*. International series in Industrial and System Engineering. New Jersey: Prentice Hall
- Corder, Anthony. 1988. *Teknik Manajemen Pemeliharaan*. Jakarta: Erlangga.
- Cox, Joyce & Joan Lambert. 2013. *Step by Step:Microsoft Access 2013*. Washington: Microsoft Press
- Davis, Gordon B. 1998. *Sistem Informasi Manajemen*. Jakarta: Penerbit PT Pustaka Binaman Pressindo
- Endy. 2011. *Perencanaan Perawatan Mesin-mesin Produksi Menggunakan Metode RCM (Reliability Centered Maintenance) di PT. Tjita Rimba Djaja*. Medan: Skripsi USU
- Fachrurrozi. 2002. *Studi Manajemen Pemeliharaan Mesin-Mesin Produksi di Industri Pengolahan Kayu PT. Inhutani Administratur Industri Bekasi, Jawa Barat*. Bogor: Skripsi IPB
- Houtzeel, A. & Schilperoort, B.A. 1976. *A Chain-Structured Part Classification System (Miclass) and Group Technology*. Proceedings of the 13th annual Meeting and Technical Conference. Cincinnati, Ohio.
- Jogiyanto. 2005. *Analisis dan Desain Sistem Informasi: Pendekatan terstruktur Teori dan Praktik Aplikasi Bisnis*. Yogyakarta: Andi Offset.

- Kusiak. 1990. *Intelligent Manufacturing System*. Englewood Cliffs. New Jersey : Prentice Hall Inc.
- Laudon, Kenneth C. & Jane P. Laudon. 2011. *Management Information Systems, 12th edition*. New Jersey: Pearson
- MacDonald, Matthew. 2010. *Access 2010: The Missing Manual*. United States of America: O'Reilly Media Inc.
- McLeod, Raymond. 2004. *Management Information System*. New Jersey: Pearson Prentice Hall, Inc.
- Munif, Abdul, 2013. *Basis Data*. Jakarta. Kementerian Pendidikan & Kebudayaan.
- Nakajima, S 1988. *Introduction to Total Productive Maintenance (TPM)*. MA, Productivity Press, Cambridge.
- Opitz, H. 1970. *A Classification System to Describe Workpieces*. New York: Pergamon Press
- Purwono, B. S. A. & Rahbini, Suyanta, Mashudi, I., & Patma, T.S., *Buku Ajar Manajemen Pemeliharaan*. Malang: NN Press.
- Santoso, Budi. 2012. *Hierarki Data (Data Hierarchy)*., Universitas Brawijaya : UB Blog. <http://pbsabn.lecture.ub.ac.id/2012/05/hierarki-data-data-hierarchy/> . (di akses tanggal 24 Januari 2015)
- Sommerville, Ian. 2011. *Software Engineering: Ninth Edition*. United States of America: Addison-Wesley
- Utami, Erma & Sukrisno. 2005. *Konsep Dasar Pengolahan dan Pemrograman Database dengan SQL Server, Ms. Access, dan Ms. Visual Basic*. Yogyakarta : ANDI Yogyakarta
- Whitten, Jeffery L. & Lonnie D. Bentley. 2007. *Systems Analysis and Design Method, Seventh Edition*. New York: McGraw-Hill Companies, Inc.
- Wireman, Terry. 1994. *Computerized Maintenance Management System*. New York: Industrial Press.
- Yudistira, Yanuar G. 2012. *Perancangan Sistem Informasi Manajemen Pemeliharaan dalam Upaya Mencegah Kerusakan Mesin-Mesin Produksi pada PT. Sekar Laut Tbk. Sidoarjo*. Malang: Skripsi UB
- Yudistiro, Danang. 2012. *Rekayasa Sistem Informasi Inventori yang Cerdas Berbasis Group Technology untuk Memudahkan Assembly Kendaraan*. Malang: Skripsi UB

UNIVERSITAS BRAWIJAYA

MANUAL BOOK

COMPUTERIZED MAINTENANCE MANAGEMENT SYSTEM

PT. GUNAWAN DIANJAYA STEEL

Hardware Requirement

Aplikasi *Computerized Maintenance Management System* ini membutuhkan dukungan perangkat keras dengan spesifikasi minimal sebagai berikut :

1. CPU dengan prosesor Intel Pentium Processor B925C (4M Cache, 2.00 GHz) atau lebih tinggi
2. RAM 1 GB atau lebih tinggi
3. Kapasitas hard disk minimal 100 MB
4. *Mouse* sebagai peralatan antarmuka
5. *Monitor* sebagai peralatan antarmuka
6. *Keyboard* sebagai peralatan antarmuka

Software Requirement

Kebutuhan perangkat lunak yang mendukung untuk aplikasi *Computerized Maintenance Management System* adalah sebagai berikut:

1. *Windows XP, Windows 7, Windows 8, Windows 8.1, Windows 10*, atau *Windows Server 2010* sebagai *Operating System*
2. *Microsoft Acces 2013 32 bit* atau lebih tinggi sebagai *RDBMS Server*

Database

Ada beberapa database yang ada di CMMS PT. GDS ini. Berikut ini adalah keterangan dari database tersebut:

3. Untuk menambah data mesin, komponen, *sparepart*, karyawan, serta Work Order hanya SPV dan manajer memiliki akses
4. Untuk merubah data mesin, komponen, *sparepart*, karyawan, serta Work Order dapat dilakukan oleh semua *user*

Aktivasi Aplikasi Sistem Informasi Manajemen Perawatan Mesin

1. Buka CD CMMS PT. GDS kemudian buka “CMMS.exe” untuk menginstall aplikasi ke komputer
2. Setelah itu di desktop terdapat aplikasi “CMMS PT. GDS . Buka aplikasi tersebut, kemudian akan muncul tampilan untuk menghubungkan aplikasi dengan database. Klik

icon dokumen berwarna kuning, kemudian cari folder tempat dimana aplikasi di install. (Normalnya di : C:\Program Files\CMMS)

- Pilih database “DatabaseGDS.mdb”, klik **Connect**

Gambar 1. *Computerized Maintenance Management System*

Menu Utama dan Sign In

Aplikasi *Computerized Maintenance Management System* mempunyai Login form untuk login user agar nantinya user dapat mengakses Form Menu Utama. Terdapat 4 form yang ada di dalam menu *Computerized Maintenance Management System*.

- Form Master Data* merupakan form yang digunakan untuk mengolah data administrasi PT. GDS seperti data *supplier*, karyawan, kategori mesin, dsb.
- Form Work Order* merupakan form yang digunakan untuk memajemen kegiatan perawatan mesin yang dilakukan di PT. GDS.
- Form Manajemen Mesin* untuk melakukan pengolahan data mesin, komponen, sparepart serta pengukuran nilai OEE.
- Form Reports* untuk menyajikan laporan.

Untuk pertama kali masuk ke dalam aplikasi ini, berikut daftar *username* dan *password* yang dapat digunakan. **Ubah password** anda ketika sudah masuk ke dalam *form* anda.

| <i>Username</i> | Jabatan | <i>Password</i> |
|----------------------|----------------|-----------------|
| Nomor Induk Karyawan | Manager | 1234 |
| | SPV | 1234 |
| | Teknisi | 1234 |

Sign In:

1. Isikan *username* sesuai dengan nomor induk masing-masing.
2. Isikan *password*.
3. Klik **SUBMIT**.

Gambar 2. Tampilan menu utama dan *sign in*

Merubah Password

1. *Sign In* terlebih dahulu menggunakan masing-masing username
2. Pilih menu **SETTING**.
3. Isikan *password* baru pada **Password**
4. Klik **Close** untuk menyimpan *+password* baru.

Gambar 3. *Setting password* baru

Form Menu Utama

Computerized Maintenance Management System PT. GDS Tbk

Thursday, July 30, 2015
1:51:36 AM

GDS PT GUNAWAN DIANJAYA STEEL, Tbk.

Login Sebagai : Manajer

PENUGASAN DAN PENJADWALAN PERAWATAN

| Work Order # | Work Type | Nama Teknisi | Tanggal Penugasan | DateRequired | Nama Mesin |
|--------------|------------|------------------|--------------------|--------------|-------------------|
| 0000000011 | Manual Pre | Maryanto | 7/9/2015 9:54:21 P | 7/10/2015 | ROLLING MILL |
| 0000000010 | Manual Pre | Margiono | 7/9/2015 9:50:26 P | 7/10/2015 | ROLL TABLE 2Q |
| 0000000008 | Breakdown | Suparman | 7/9/2015 9:00:42 P | 7/7/2015 | ROLLING MILL |
| 0000000007 | Breakdown | Mochamad Ibrahim | 7/9/2015 8:57:59 P | 7/11/2015 | ROLL TABLE 5Q |
| 0000000006 | Breakdown | Maryanto | 7/9/2015 8:43:11 P | 7/13/2015 | ROLL TABLE 2Q |
| 0000000005 | Corrective | Inwan S | 7/9/2015 8:18:16 P | 7/13/2015 | ROLL TABLE A1 |
| 0000000004 | Breakdown | Mochamad Ibrahim | 7/9/2015 8:13:45 P | 7/11/2015 | REHEATING FURNACE |
| 0000000003 | Corrective | Margiono | 7/9/2015 8:08:42 P | 7/11/2015 | REHEATING FURNACE |
| 0000000002 | Corrective | Mochamad Ibrahim | 6/22/2015 9:02:50 | 6/22/2015 | HOT LEVELLER |

REPORTS
Laporan Departemen Maintenance PT. GDS

SETTING
Pengaturan User Account dan Data Diri

EXIT

VISUAL OEE

| IDMesin | Nama Mesin | Periode | Availibility | Performance | Quality | OEE | Target |
|----------|--------------|---------|--------------|-------------|-------------|-------------|--------|
| 03MDLLZH | ROLLING MILL | 1 | 92.65625 | 77.97976391 | 95.64032697 | 69.103125 | 80 |
| 03MDLLZH | ROLLING MILL | 2 | 100 | 80.42553191 | 92.85714285 | 74.68085106 | 80 |
| 03MDLLZH | ROLLING MILL | 3 | 100 | 84.2212355 | 90.993111 | 73.44226382 | 80 |

Cek Tindakan Hari Ini

Urutkan Data Berdasarkan Mesin

Urutkan Data Berdasarkan Periode

Copyright © 2015 Reza Maulana @ IEBU - PT. Gunawan Dianjaya Steel Tbk

Gambar 4. Tampilan *form* untuk Menu Utama

1. **Master Data** merupakan *form* yang digunakan untuk mengolah data administrasi PT. GDS seperti data *supplier*, karyawan, kategori mesin, dsb..
2. **Work Order** merupakan *form* yang digunakan untuk memajemen kegiatan perawatan mesin yang dilakukan di PT. GDS.
3. **Manajemen Mesin** untuk melakukan pengolahan data mesin, komponen, sparepart serta pengukuran nilai OEE
4. **Report** untuk melihat laporan keseluruhan berupa laporan *inventory sparepart*, laporan kerusakan mesin yang terjadi dan laporan jadwal perawatan mesin.
5. **Setting** untuk mengatur data pribadi beserta password akses user terkait.
6. **Tampilan Penugasan dan Penjadwalan Perawatan** untuk melihat jadwal kegiatan perawatan mesin yang belum dilakukan
7. **Tampilan Visual OEE** untuk melihat data historis nilai OEE yang telah dihitung.
8. **Sort Data OEE** untuk mengurutkan data di visual OEE berdasarkan periode maupun mesin.
9. **Exit** untuk keluar dari aplikasi *Computerized Maintenance Management System*.

10. **Cek Tindakan Hari Ini** untuk melihat jadwal *maintenance* yang harus dilaksanakan pada hari itu.

Form Master Data

The screenshot shows a web application window titled "Master List". It has a menu bar with options: Department, Failure Code, Asset Category, Suppliers/Contractors, Assets, Location, Employee/Requester, and Misc. The main content area is divided into two sections:

- New Record:** Contains two input fields: "Department No." and "Department Desc.". Below these fields are two buttons: "Save" and "Close".
- Department List:** Contains a table with the following data:

| Department No. | Department Desc |
|----------------|------------------|
| ELK | Elektrik |
| ME | Mekanik |
| WT | Water Threatment |

Gambar 5. Tampilan *form* Master Data

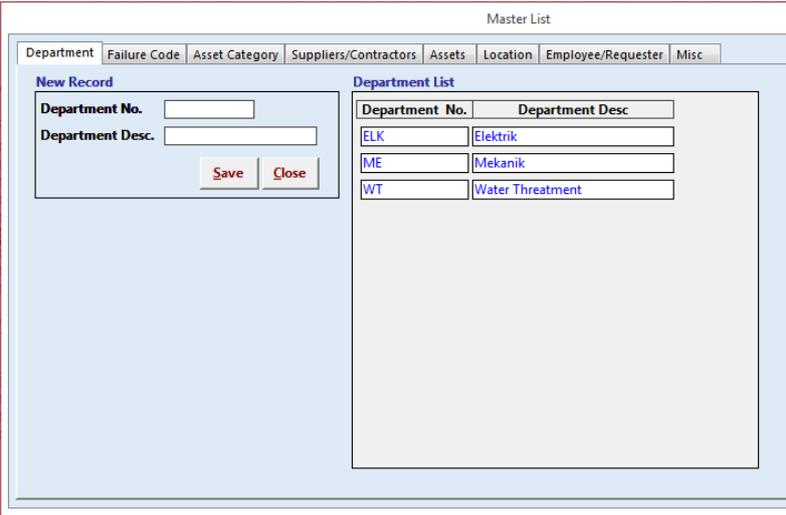
Tambah Data Departemen

This screenshot is identical to the one above, showing the "Master List" form with the "New Record" and "Department List" sections.

1. Isi Kode Departemen di **Department No.**
2. Isi deskripsi departemen di **Department Desc.**
3. Klik **SAVE** untuk menyimpan data.
4. Klik **Close** untuk keluar.

Gambar 5. Tampilan *form* Master Data

Tambah Failure Code

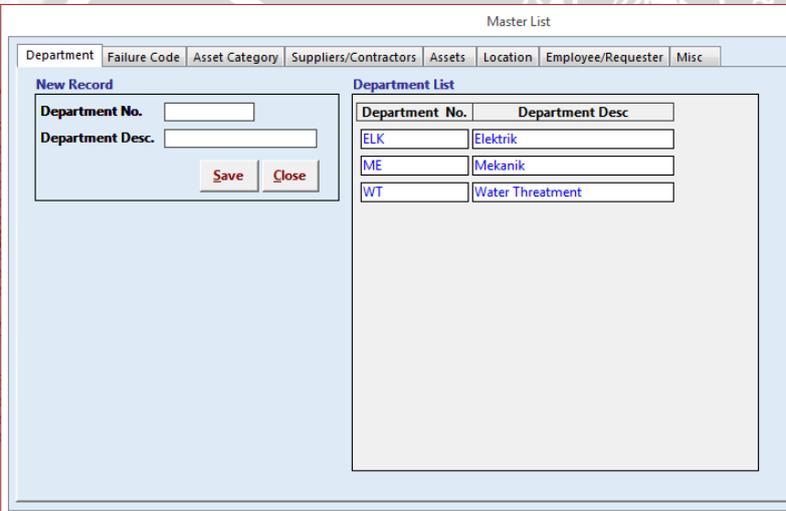


| Department No. | Department Desc |
|----------------|------------------|
| ELK | Elektrik |
| ME | Mekanik |
| WT | Water Threatment |

1. Isi Kode Kerusakan di **Failure Code**.
2. Isi deskripsi kerusakan di **Failure Desc**.
3. Klik **SAVE** untuk menyimpan data.
4. Klik **Close** untuk keluar.

Gambar 5. Tampilan form Master Data

Tambah Kategori Mesin



| Department No. | Department Desc |
|----------------|------------------|
| ELK | Elektrik |
| ME | Mekanik |
| WT | Water Threatment |

1. Isi Kode Kategori Mesin di **ID Kategori Mesin**
2. Isi Nama Kategori di **Kategori Mesin**
3. Klik **SAVE** untuk menyimpan data.
4. Klik **Close** untuk keluar.

Gambar 5. Tampilan form Master Data

Tambah Data Supplier dan Kontraktor

The screenshot shows a software interface titled 'Master List'. It has several tabs: 'Department', 'Failure Code', 'Asset Category', 'Suppliers/Contractors', 'Assets', 'Location', 'Employee/Requester', and 'Misc'. The 'Suppliers/Contractors' tab is active. The form is split into two panes. The left pane, 'New Record', contains input fields for: Supplier No, Supplier Name, Contact Person, Designation, Address, City, State/Province, Post/Zip code, Country, Telephone No., Fax No., and Services. The right pane, 'Suppliers/Contractors List', displays a table with columns 'Supplier No' and 'Supplier Name'. The table contains the following data:

| Supplier No | Supplier Name |
|--------------|------------------------------|
| AsiaBear | PT. Asian Bearindo Sejahtera |
| Casey | Casey Equipment |
| Dongkuk | Dongkuk Steel Mill Co., Ltd. |
| MBIndustrial | M B Industrial Equipment Inc |

At the bottom of the form are 'Save' and 'Close' buttons.

Gambar 5. Tampilan form Master Data

1. Isi Kode Supplier di **Supplier No.**
2. Isi Nama Supplier di **Supplier Name.**
3. Isi Nama Contact Person di **Contact Person**
4. Isi Deskripsi supplier di **Description.**
5. Isi alamat supplier di **Address, City, State/Province, Zip Code, dan Country.**
6. Isi nomer telepon contact person dan fax di **Telephone No. dan Fax No.**
7. Isi **Service** dengan servis apa saja yang diberikan supplier
8. Klik **Close** untuk keluar.

Form Work Order

The screenshot shows a software interface titled 'Work Order List'. It displays a table with the following columns: Work Order #, Work Type, Work Status, AssetNo, Problem Description, Nama Teknisi, and Action. The table contains the following data:

| Work Order # | Work Type | Work Status | AssetNo | Problem Description | Nama Teknisi | Action |
|--------------|-------------------|-------------------|----------|-------------------------------------|------------------|--------|
| 0000000012 | Manual Preventive | Open | 03HRERTA | Pengecekan paju Back Up Roll Ho | Budi Wicaksono | |
| 0000000011 | Manual Preventive | Open | 03MDLLZH | Pengecekan baut - baut Screw Dov | Maryanto | |
| 0000000010 | Manual Preventive | Open | 03RE2QA | Pengecekan Gear box dan Roll Tal | Margiono | |
| 0000000008 | Breakdown | Open | 03MDLLZH | * Ganti Work Roll Mill Atas dan Ba | Suparman | |
| 0000000007 | Breakdown | Open | 03RE5QA | * Ganti Penggerak Gear V.Z 26 Roll | Mochamad Ibrahim | |
| 0000000006 | Breakdown | Open | 03RE2QA | - Las tutup motor Roll Table A1 & | Maryanto | |
| 0000000005 | Corrective | Open | 03RE1PA | * Ganti Bearing 22326 As Blower. | Mochamad Ibrahim | |
| 0000000004 | Breakdown | Open | 02SECEKU | * Ganti Hydraulic Extractor Furnace | Margiono | |
| 0000000003 | Corrective | Open | 02SECEKU | * Ajust Paju Box Back Up Roll Hot I | Mochamad Ibrahim | |
| 0000000002 | Corrective | Waiting for Parts | 03HRERTA | | | |

Below the table are buttons: 'Total Data' (11), 'PM Baru', 'WorkOrder Baru', 'Print WO', 'Lihat Semua', 'Export to Excel', and 'Close'. Below these buttons is a search filter section with the following fields:

Cari Work Order | Sorting Work Order | Tutorial Form Work Order

Work Type: [] Masalah [X!]
 Work Status: [] Lokasi [X!]
 Work Trade: [] PM No. [X!]
 Workorder No: [] PM Task No. [X!]
 Nama PIC: [] Received Date: [] To: [] M [X!]
 Mesin: [] Est Start Date: [] To: [] M [X!]
 Prioritas WO: [] Act.Close Date: [] To: [] M [X!]
 Failure Code: [] PM Target Start Date: [] To: [] M [X!]

Find => [] Cari Reset

Gambar 7. Form Work Order

Tambah Work Order Baru

1. Pilih **Jenis Mesin**, **Nama Mesin** dan **Tindakan** yang akan dijadwalkan.
2. Isikan **Tanggal Perawatan** yang dijadwalkan.
3. Pilih **Prioritas** perawatan mesin.
4. Pilih **Karyawan**, apabila nama karyawan yang dimaksud tidak ada maka klik tanda (+!) untuk menambah data karyawan baru.
5. Klik **SAVE** untuk menyimpan data jadwal perawatan mesin baru.

Gambar 9. Tambah Work Order baru

Form Manajemen Mesin

| MESIN | KOMPONEN | SPAREPART | OEE MESIN |
|------------|-------------------|--------------|-----------|
| ID Mesin # | Nama Mesin | Status Mesin | |
| 025CCEKU | REHEATING FURNACE | Active | |
| 03RE41PA | ROLL TABLE A1 | Active | |
| 03RE42PA | ROLL TABLE A2 | Active | |
| 03RE2QPA | ROLL TABLE 2Q | Active | |
| 03RE3QPA | ROLL TABLE 3Q | Active | |
| 03MDLZH | ROLLING MILL | Active | |
| 03HRERTA | HOT LEVELLER | Active | |
| 03RE4QPA | ROLL TABLE 4Q | Active | |
| 03RE5QPA | ROLL TABLE 5Q | Active | |
| 04CDEDAL | COOLING BED | Active | |
| 03RE6QPA | ROLL TABLE 6Q | Active | |
| 03RE7QPA | ROLL TABLE 7Q | Active | |

Gambar 10. Form Manajemen Mesin

Menambah Mesin Baru

Gambar 11. Tambah Mesin Baru

1. Isikan informasi mesin meliputi Nama mesin, lokasi mesin, Asset Category, Status Mesin, Kondisi, Departemen, Nama PIC Mesin, dan seterusnya.
2. Untuk menambah Supplier, Contractor, maupun nama PIC Mesin, klik tanda panah kebawah di dekat kolom.
3. Apabila data sudah di isi, klik **Save**

Menghitung OEE

Gambar 12. Hitung nilai OEE

1. Pilih Mesin yang akan dihitung nilai OEE.
2. Isikan informasi berupa periode, produk yang di produksi, durasi produksi, kecepatan produksi, setup time, schedule downtime, dan defect serta target OEE
3. Apabila sudah, klik **Hitung**.
4. Apabila data ingin disimpan, klik **Save**

Report

Pada menu *report* masih ada sub menu lagi karena sistem informasi manajemen perawatan mesin menghasilkan report mesin, komponen, sparepart, karyawan, kerusakan mesin dan jadwal perawatan mesin yang dibutuhkan.

Gambar 13. sub menu *report*

Report Work Order Periode

Gambar 14. Periode *report* yang dibutuhkan

1. Pilih tanggal dimulainya *report* kerusakan mesin yang dibutuhkan pada **From**.
2. Pilih tanggal batas *report* kerusakan mesin pada **To**.
3. Klik **View Report** untuk melihat *report* kerusakan mesin.



PT GUNAWAN DIANJAYA STEEL, Tbk.

Laporan Work Order Periode

| Work Order No | Problem Description | Work Type | Asset No | Asset Description | Location No | Date Received | Date Start | Date End | Work Status |
|---------------|--|-------------------|----------|-------------------|-------------|------------------------|------------|----------|-------------|
| 000000007 | * Ganti Penggerak Gear V.Z 26 Roll Table 5Q | Breakdown | 03RE5QPA | ROLL TABLE 5Q | 03 | 7/9/2015 8:57:59 PM | | | Open |
| 000000008 | * Ganti Work Roll Mill Atas dan Bawah | Breakdown | 03MDLLZH | ROLLING MILL | 03 | 7/9/2015 9:00:42 PM | | | Open |
| 000000009 | * Pasang / Ganti Back Up Roll Mill Atas dan Bawah. | Breakdown | 03MDLLZH | ROLLING MILL | 03 | 7/9/2015 9:15:37 PM | 7/7/2015 | 7/7/2015 | Closed |
| 000000010 | Pengecekan Gear box dan Roll Table 2Q | Manual Preventive | 03RE2QPA | ROLL TABLE 2Q | 03 | 7/9/2015 9:50:26 PM | | | Open |
| 000000011 | Pengecekan baut – baut Srew Down Mill Barat & Timur (coupling, gear box dll) | Manual Preventive | 03MDLLZH | ROLLING MILL | 03 | 7/9/2015 9:54:21 PM | | | Open |
| 000000012 | Pengecekan paju Back Up Roll Hot Leveller Atas dan Bawah | Manual Preventive | 03HRERTA | HOT LEVELLER | 03 | 7/9/2015 9:56:51 PM | | | Open |
| 000000013 | Pembersihan scale di | Manual | 02SECEKU | REHEATING | 02 | 7/9/2015 | | | Cancelled |

Gambar 15. Tampilan *report* kerusakan mesin

Report OEE Mesin

Gambar 16. Periode *report* yang dibutuhkan

1. Pilih tanggal dimulainya *report* jadwal perawatan mesin yang dibutuhkan pada **From**.
2. Pilih tanggal batas *report* jadwal perawatan mesin pada **To**.
3. Klik **View Report** untuk melihat *report* jadwal perawatan mesin.

| Nama Mesin | ID Mesin | ID OEE | Periode OEE | Availability | Performance | Quality | OEE | Goal | Nomor |
|--------------|----------|-----------|-------------|--------------|--------------|-----------|-------------|------|-------|
| ROLLING MILL | 03MDLLZH | 03MDLLZH1 | 1 | 92.65625 | 77.979763912 | 95.640327 | 69.103125 | 80 | |
| | | 03MDLLZH2 | 2 | 100 | 80.425531915 | 92.857143 | 74.68085106 | 80 | |
| | | 03MDLLZH3 | 3 | 100 | 84.2212355 | 90.993111 | 73.44226383 | 80 | |

-Keterangan-
: OEE di bawah Goal

Page 1 of 1

Gambar 17. Tampilan *report* OEE Mesin

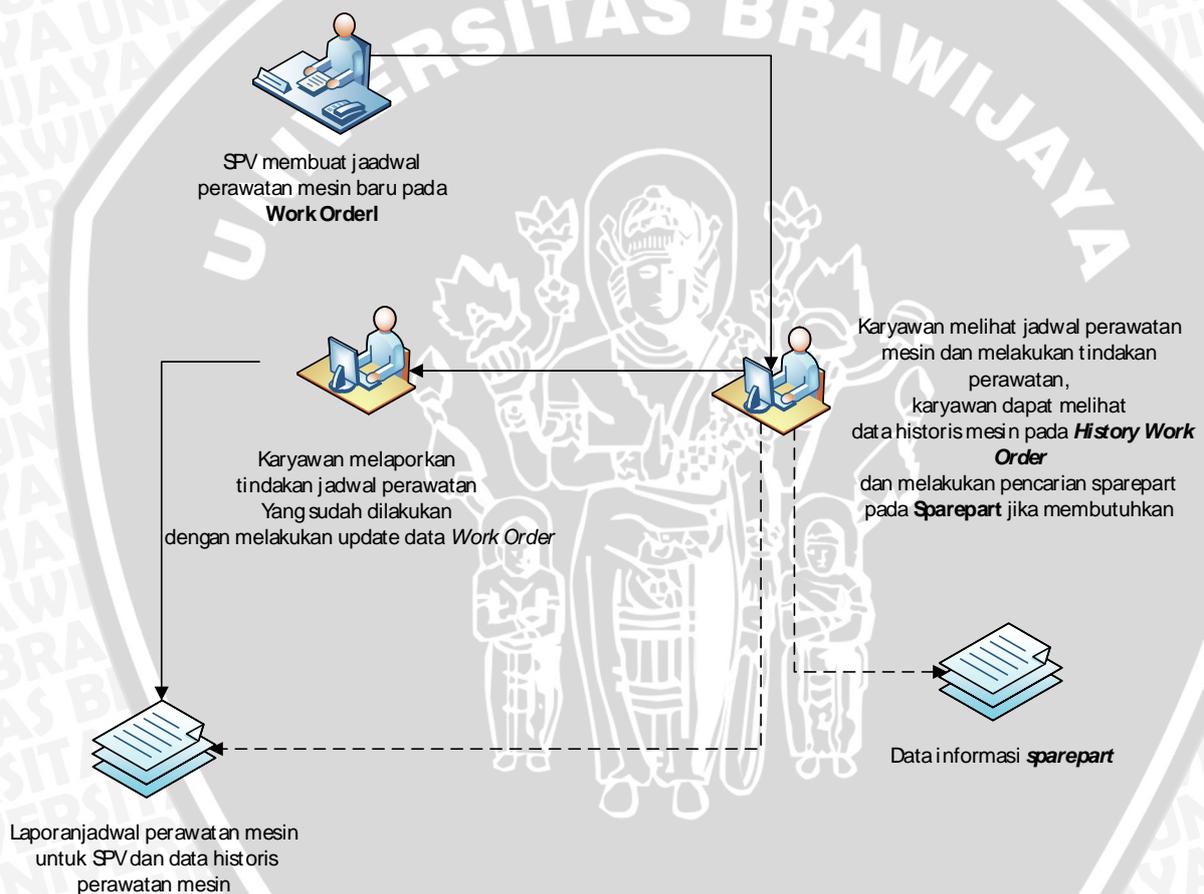
Alur Sistem Penugasan Jadwal Perawatan Mesin Baru

Gambar 32. Menunjukkan alur sistem pembuatan jadwal perawatan mesin baru hingga SPV menerima laporan.

1. SPV membuat jadwal perawatan mesin baru pada *form Work Order*.
2. Jadwal penugasan perawatan mesin baru yang telah dibuat masuk ke dalam *database*.
3. Teknisi jadwal kegiatan perawatan dari SPV setiap harinya pada *form Work Order*. Untuk mendapatkan *alert* mengenai kegiatan perawatan yang dilakukan pada hari itu, teknisi dapat memilih **Cek Tindakan Perawatan Hari ini**
4. Jika teknisi membutuhkan informasi data *sparepart*, maka teknisi dapat memilih **Manajemen Mesin > Sparepart >** mengisi data sparepart berupa kode/nama/komponen/mesin. Jika teknisi membutuhkan informasi data historis mesin,

maka karyawan dapat memilih **Mesin** > Pilih mesin mana yang ingin dilihat > **History Work Order**.

5. Teknisi melakukan laporan kepada SPV jika sudah selesai melakukan tindakan perawatan mesin dengan cara melakukan *update* data *Work Order* dan melaporkan jika terdapat pemakaian *sparepart* dengan cara melakukan *update* data *sparepart* dengan memilih **Pakai Sparepart**.
6. Laporan *Work Order* mesin yang sudah masuk ke dalam *database* akan dilolah sehingga menghasilkan laporan *Work Order* mesin untuk SPV yang dapat diakses oleh SPV pada **Report** dan menjadi data historis mesin yang baru.



Gambar 18. Alur Sistem Penugasan dan Penjadwal Perawatan Mesin