

BAB V

DESAIN, IMPLEMENTASI DAN PENGUJIAN SISTEM

Pada bagian ini akan dijelaskan mengenai tahapan dalam perancangan dan pengembangan sistem mulai dari *design*, *implementation* sampai dengan *testing*. Bagian ini merupakan bagian penting dalam suatu proses perancangan sistem yang diharapkan dapat menghasilkan suatu perancangan sistem informasi manajemen pemeliharaan mesin-mesin produksi yang sesuai dengan kebutuhan *user* atau pengguna nantinya.

5.1. Desain (*Design*)

Tahapan desain ini merupakan proses pengidentifikasian komponen-komponen sistem informasi yang akan didesain secara rinci/detail atau dapat juga disebut dengan tahapan dimana spesifikasi sistem secara lengkap dibuat berdasarkan kebutuhan yang telah direkomendasikan pada tahap sebelumnya. Tujuan dari tahapan desain ini yaitu untuk memberikan gambaran secara umum kepada *user* tentang sistem yang baru. Proses desain ini berfokus pada struktur data, arsitektur perangkat lunak, representasi *interface*, dan detail alortima prosedural.

5.1.1. Desain *Database*

Pada bagian ini akan dijelaskan mengenai desain *database* yang akan terlibat atau digunakan dalam sistem informasi manajemen pemeliharaan mesin produksi. Desain *database* ini terdiri dari desain *database* logis dan desain *database* fisik.

5.1.1.1. Desain *Database* Logis

Perancangan desain *database* logis adalah proses pemetaan model konseptual menjadi struktur logika basis data, tahap ini dilakukan agar sesuai dengan model DBMS yang digunakan. Hal ini sesuai dengan penggunaan Microsoft SQL Server 2005 sebagai basis data karena merupakan perangkat lunak sistem manajemen basis data relasional. Model relasional yang terbentuk adalah sebagai berikut, dimana kata-kata yang digarisbawahi merupakan *primary key*.

1. **Mesin** (KODE_MESIN, NAMA_MESIN, MERK, THN_BUAT, PERSHNN_BUAT)
2. **Spareparts** (KODE_SP, NAMA_SP, STOCK, SUPPLIER, ALMT_SUPP, TLP_SUPP)
3. **Karyawan** (NIK, NAMA, ALAMAT, TELP, PEND_TRKHIR, KEAHLIAN)

4. **Kerusakan Mesin** (ID_RUSAK, KODE_MESIN, KODE_SP, NIK, WAKTU_RUSAK, WAKTU_BAIK, KODE_MAINT)

5. **Jenis Maintenance** (KODE_MAINT, NAMA_MAINT, JENIS_MAINT)

5.1.1.2. Normalisasi

Normalisasi merupakan teknik yang digunakan untuk mengidentifikasi hubungan, dimulai dari menguji hubungan, yaitu *functional dependencies* antara atribut atau digunakan untuk memvalidasi model data. Serangkaian aturan diberlakukan pada basis data logis untuk meningkatkannya.

Syarat-syarat pada tahapan normalisasi yang ada (dari tahap 1NF sampai dengan 3NF) sebagai berikut:

1. Tahap 1NF : merupakan sebuah relasi dimana setiap baris dan kolom berisikan satu dan hanya satu nilai (tidak ada *repeating value*).
2. Tahap 2NF : merupakan sebuah relasi dalam 1NF dan setiap atribut *non-primary key* bersifat *fully functionally dependent* pada *primary key* (semua atribut bergantung pada *primary key*).
3. 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 penggantian karena data atau hubungan sudah memenuhi persyaratan-persyaratan tersebut (data atau hubungan sudah normal).

5.1.1.3. Desain Database Fisik

Setelah melalui tahap perancangan desain logis dan normalisasi data, tahap selanjutnya merancang *database* fisik. Tujuan utama tahap ini adalah untuk mengimplementasikan basis data sebagai suatu himpunan *record*, *file*, *index* atau struktur data lainnya. Rancangan yang dilakukan meliputi komponen tabel, *data type*, ukuran *field* dan keterangan *key*. Rincian tabel-tabel tersebut sebagai berikut:

Tabel 5.1 Entitas Mesin

<i>Field Name</i>	<i>Type</i>	<i>Key</i>	<i>Description</i>
KODE_MESIN	Nvarchar (8)	PK	Kode untuk mesin
NAMA_MESIN	Varchar(50)		Nama dari mesin
MERK	Varchar(20)		Merk dari mesin
THN_BUAT	Int(5)		Tahun pembuatan mesin
PERSHNN_BUAT	Varchar(50)		Perusahaan pembuat mesin

Tabel 5.2 Entitas Spareparts

<i>Field Name</i>	<i>Type</i>	<i>Key</i>	<i>Description</i>
KODE_SP	Nvarchar(8)	PK	Kode dari <i>spareparts</i>
NAMA_SP	Varchar(30)		Nama dari <i>sparepart</i>
STOCK	Int(2)		Persediaan untuk <i>spareparts</i>
SUPPLIER	Varchar(30)		Nama <i>supplier spareparts</i>
ALMT_SUPP	Varchar(50)		Alamat dari <i>supplier spareparts</i>
TLP_SUPP	Nvarchar(25)		Telepon dari <i>supplier spareparts</i>

Tabel 5.3 Entitas Karyawan

<i>Field Name</i>	<i>Type</i>	<i>Key</i>	<i>Description</i>
NIK	Nvarchar(8)	PK	Nomor induk karyawan
NAMA	Varchar(30)		Nama karyawan
ALAMAT	Varchar(50)		Alamat karyawan
TELP	Nvarchar(25)		Telepon karyawan
PEND_TRKHIR	Varchar(10)		Pendidikan terakhir dari karyawan
KEAHLIAN	Varchar(max)		Keahlian dari karyawan

Tabel 5.4 Entitas Kerusakan Mesin

<i>Field Name</i>	<i>Type</i>	<i>Key</i>	<i>Description</i>
ID_RUSAK	Nvarchar (8)	PK	Kode untuk kerusakan mesin
KODE_MESIN	Nvarchar (8)	FK	Kode mesin dari entitas mesin
KODE_SP	Nvarchar (8)	FK	Kode <i>spareparts</i> dari entitas <i>spareparts</i>
NIK	Nvarchar (8)	FK	Nomor induk karyawan dari entitas karyawan

<i>Field Name</i>	<i>Type</i>	<i>Key</i>	<i>Description</i>
WAKTU_BAIK	Int(8)		Lama waktu yang dialami mesin pada saat mesin tersebut berjalan dengan baik
WAKTU_RUSAK	Int(8)		Lama waktu yang dialami mesin pada saat mesin tersebut rusak
KODE_MAINT	Nvarchar (8)	FK	Kode <i>maintenance</i> dari entitas jenis <i>maintenance</i>

Tabel 5.5 Entitas Jenis *Maintenance*

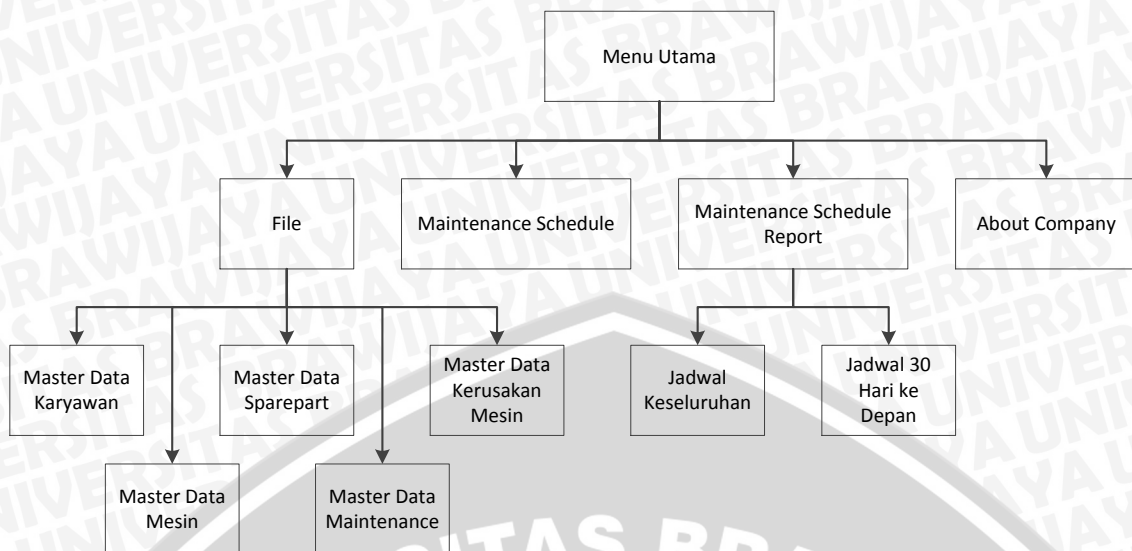
<i>Field Name</i>	<i>Type</i>	<i>Key</i>	<i>Description</i>
KODE_MAINT	Nvarchar(8)	PK	Kode dari <i>maintenance</i>
NAMA_MAINT	Varchar(25)		Nama dari <i>maintenance</i>
JENIS_MAINT	Varchar(max)		Jenis dari <i>maintenance</i> yang akan dilakukan

5.1.2. Desain *User Interface*

Setelah melakukan desain *database* berupa *database* logis, normalisasi dan *database* fisik, langkah selanjutnya yang dilakukan adalah membuat desain *user interface* (desain antarmuka). Desain *user interface* ini bertujuan untuk membuat rancangan dari tampilan sistem yang nantinya akan berinteraksi langsung dengan *user* (pengguna). Desain ini merupakan desain sistem dialog yang dapat diartikan dan diimplementasikan sehingga pengguna atau pemakai dapat berkomunikasi dengan sistem yang dirancang.

5.1.2.1. Desain Menu

Pada bagian ini akan dijelaskan mengenai desain menu dari sistem informasi manajemen pemeliharaan mesin produksi yang dibuat. Desain ini digambarkan dalam bentuk hirarki untuk memudahkan desain *user interface* dari sistem nantinya. Desain menu sistem informasi manajemen pemeliharaan mesin produksi memiliki empat (4) buah menu utama yaitu menu *File*, menu *Maintenance Schedule*, *Maintenance Schedule Report* dan menu *About Company*. Tampilan menu utama dari sistem informasi manajemen pemeliharaan mesin produksi dapat digambarkan melalui hirarki sebagai berikut:



Gambar 5.1 Hirarki menu Sistem Informasi Manajemen Pemeliharaan Mesin Produksi.

Gambar 5.1 menunjukkan bahwa *user* (kepala Departemen *Process Control & Engineering*) nantinya dapat memiliki kewenangan dalam mengakses sistem informasi tersebut. Namun, kewenangan tersebut tidak termasuk memasukkan, menghapus, mengubah dan memperbarui data. Proses-proses tersebut (memasukkan, menghapus, mengubah dan memperbarui data) hanya dapat dilakukan oleh administrator Departemen *Process Control & Engineering*. Menu *Maintenance Schedule* dapat dipilih untuk mendapatkan informasi mengenai penjadwalan *maintenance*, penanganan kerusakan mesin dan penggantian *spareparts* yang dibutuhkan. Menu *Maintenance Schedule Report* merupakan menu yang di dalamnya terdapat sub menu untuk meminta laporan dari penjadwalan *maintenance* di PT. Sekar Laut Tbk secara keseluruhan dan dalam kurun waktu 30 hari ke depan.

5.1.2.2. Desain Form

Desain *form* dibuat untuk merancang tampilan antar muka antara *user* dengan komputer agar bersifat lebih komunikatif. Dalam Sistem Informasi Manajemen Pemeliharaan Mesin Produksi pada PT. Sekar Laut Tbk. yang akan dibuat, terdapat beberapa macam *form* di dalamnya yaitu *form* Master Data Karyawan, Master Data Mesin, Master Data *Sparepart*, Master Data *Maintenance*, Master Data Kerusakan Mesin, *About Company* dan *form Maintenance Schedule*.

1. *Form Maintenance Schedule*

Form Maintenance Schedule ini merupakan *form* yang berisi tentang informasi penjadwalan pemeliharaan mesin-mesin produksi yang terdapat pada perusahaan. *Form* ini memberikan informasi penjadwalan dan pelaporan mengenai penggantian dan ketersediaan sparepart serta penanganan mesin yang mengalami kerusakan. Tampilan desain untuk *form Maintenance Schedule* dalam sistem informasi manajemen pemeliharaan mesin dapat dilihat pada Gambar 5.2.



Gambar 5.2 Tampilan desain *form Maintenance Schedule*.

Keterangan gambar:

- Nomor 1 merupakan judul dari *form Maintenance Schedule*.
- Nomor 2 berisi pilihan untuk melihat hasil penjadwalan berdasarkan waktu (bulan dan tahun) atau kode mesin.
- Nomor 3 berisi pilihan untuk melihat hasil penjadwalan berdasarkan waktu (bulan dan tahun).
- Nomor 4 berisi pilihan untuk melihat hasil penjadwalan berdasarkan kode mesin dan juga menampilkan nama mesin yang dipilih.
- Nomor 5 merupakan hasil dari penjadwalan.
- Nomor 6 berisi tombol-tombol untuk menjalankan perintah-perintah yang telah dipilih dalam *form Maintenance Schedule*.

2. *Form Master Data Karyawan*

Form Master Data Karyawan merupakan *form* master data berisi informasi-informasi tentang karyawan perusahaan yang di dalamnya terdapat perintah untuk memasukkan, menghapus, mengubah dan memperbarui data. Rancangan *form Master Data Karyawan* dapat dilihat pada Lampiran 1.

3. *Form Master Data Mesin*

Form Master Data Mesin merupakan *form* master data berisi informasi-informasi tentang mesin-mesin produksi perusahaan yang di dalamnya terdapat perintah untuk memasukkan, menghapus, mengubah dan memperbarui data. Rancangan *form Master Data Mesin* dapat dilihat pada Lampiran 2.

4. *Form Master Data Sparepart*

Form Master Data Sparepart merupakan *form* master data berisi informasi-informasi tentang *sparepart* atau komponen-komponen yang dimiliki oleh perusahaan untuk persediaan jika terjadi kerusakan yang membutuhkan penggantian *sparepart* atau komponen yang di dalamnya terdapat perintah untuk memasukkan, menghapus, mengubah dan memperbarui data. Rancangan *form Master Data Sparepart* dapat dilihat pada Lampiran 3.

5. *Form Master Data Maintenance*

Form Master Data Maintenance merupakan *form* master data berisi informasi-informasi tentang jenis-jenis *maintenance* dalam perusahaan untuk dijadikan acuan *maintenance* apabila terjadi kerusakan yang di dalamnya terdapat perintah untuk memasukkan, menghapus, mengubah dan memperbarui data. Rancangan *form Master Data Maintenance* dapat dilihat pada Lampiran 4.

6. *Form Master Data Kerusakan Mesin*

Form Master Data Kerusakan Mesin merupakan *form* master data berisi informasi-informasi tentang kerusakan mesin yang sering terjadi dalam perusahaan yang di dalamnya terdapat perintah untuk memasukkan, menghapus, mengubah dan memperbarui data. Rancangan *form Master Data Kerusakan Mesin* dapat dilihat pada Lampiran 5.

7. *Form Jadwal Maintenance 30 Hari Ke Depan*

Form Jadwal Maintenance 30 Hari Ke Depan merupakan *form report* yang berisi tentang jadwal *maintenance* yang akan terjadi dalam waktu 30 hari berikutnya dihitung mulai dari tanggal sekarang (tanggal pada saat *user* menjalankan *form* ini).

8. *Form Jadwal Maintenance* Keseluruhan

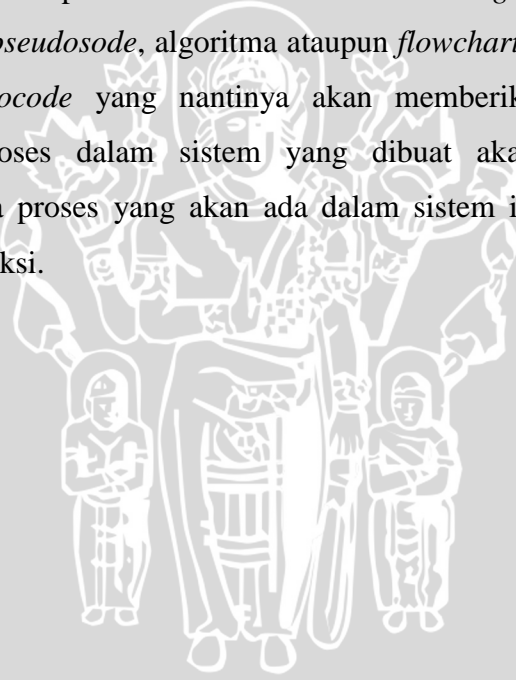
Form Jadwal Maintenance Keseluruhan merupakan *form report* yang berisi tentang jadwal *maintenance* keseluruhan yang akan terjadi yang telah terjadwalkan oleh sistem informasi berdasarkan selisih antar waktu kerusakan untuk masing-masing mesin produksi yang terdapat pada perusahaan.

9. *Form About Company*

Form About Company merupakan form yang berisikan informasi-informasi perusahaan. Mulai dari sejarah sampai identitas/kontak dari perusahaan itu sendiri. Rancangan *form About Company* dapat dilihat pada Lampiran 6.

5.1.3. Desain Proses

Desain proses akan menerjemahkan syarat kebutuhan ke sebuah perancangan perangkat lunak yang dapat diperkirakan sebelum dibuat *coding*. Desain proses dapat dijabarkan dalam bentuk *pseudocode*, algoritma ataupun *flowchart*. Dalam hal ini, yang digunakan adalah *pseudocode* yang nantinya akan memberikan gambaran suatu algoritma bagaimana proses dalam sistem yang dibuat akan berjalan. Berikut *pseudocode* dari beberapa proses yang akan ada dalam sistem informasi manajemen pemeliharaan mesin produksi.




```

// tambah, simpan dan hapus data SIM pemeliharaan mesin produksi
//menambah data atau memasukkan data baru
    Mengambil data ← SELECT FROM mesin
    Posisi rekam data ← baris tabel mesin – 1
    Membuat kode mesin ← “M-000” & value (Mid(kolom Kode Mesin, 4,
        Len(kolom Kode Mesin) - 1)) + 1

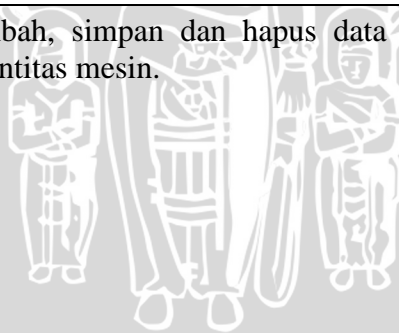
//menyimpan data
    Menyimpan data ← INSERT INTO mesin (KODE_MESIN,
        NAMA_MESIN, MERK, THN_BUAT,
        PERSHNN_BUAT)

    IF hasil baris penyimpanan > 0 THEN
        Data berhasil disimpan
    END IF
    Memperbarui data ← UPDATE mesin SET (KODE_MESIN,
        NAMA_MESIN, MERK, THN_BUAT,
        PERSHNN_BUAT)

//menghapus data
    IF len(nama mesin) = 0 THEN
        Muncul peringatan bahwa tidak ada data yang dihapus
    ELSE
        Muncul peringatan data dihapus atau tidak
        IF data batal dihapus THEN
            Keluar
        ELSE
            DELETE FROM mesin WHERE KODE_MESIN
        END IF
    END IF

```

Gambar 5.3 Pseudocode tambah, simpan dan hapus data SIM pemeliharaan mesin produksi pada entitas mesin.



```
//pencarian data SIM pemeliharaan mesin produksi
IF kata kunci pencarian = " " THEN
    Muncul peringatan untuk memasukkan kata kunci
ELSEIF kata kunci tidak sesuai THEN
    Muncul peringatan untuk memasukkan kata kunci yang sesuai
END IF
IF memilih berdasarkan kode mesin THEN
    SELECT KODE_MESIN as [Kode Mesin], NAMA_MESIN as
    [Nama Mesin], MERK as [Merk], THN_BUAT as [Tahun Produksi],
    PERSHNN_BUAT as [Perusahaan (Produsen)] FROM mesin
    WHERE KODE_MESIN=' ' & kata kunci & " ' ", koneksi
    tersambung.
ELSEIF memilih berdasarkan nama mesin THEN
    SELECT KODE_MESIN as [Kode Mesin], NAMA_MESIN as
    [Nama Mesin], MERK as [Merk], THN_BUAT as [Tahun Produksi],
    PERSHNN_BUAT as [Perusahaan (Produsen)] FROM mesin
    WHERE NAMA_MESIN like ' ' & kata kunci & " %' ", koneksi
    tersambung.
END IF
```

Gambar 5.4 *Pseudocode* proses pencarian data SIM pemeliharaan mesin produksi pada entitas mesin.

```
//penjadwalan pemeliharaan mesin-mesin produksi
//inisialisasi variabel
    Waktu rusak adalah waktu dimana mesin mengalami kerusakan
    Waktu baik adalah waktu dimana mesin mulai dapat bekerja seperti semula

//penentuan waktu maintenance per mesin
    wakturusak = input waktu kerusakan atau umur mesin (MTTF)
    MTTR = lama perbaikan
    For i As Integer = 1 To 31
        tanggalrusak = DateAdd("n", wakturusak, #9/24/2012#)
        INSERT INTO reportmesintanggal VALUES (" " & tanggalrusak & " ',' " &
        Nama Mesin & " ',' " & Nama Sparepart & " ',' " & Nama Karyawan & " _
        ',' " & Nama Maintenance & " ')"

        wakturusak = wakturusak + MTTF + MTTR

    SELECT Tanggal_Rusak FROM reportmesintanggal ORDER BY_
    Tanggal_Rusak ASC

    Next i
    Data hasil pengulangan dimasukkan ke tabel reportmesintanggal
```

Gambar 5.5 *Pseudocode* penjadwalan pemeliharaan mesin-mesin produksi.

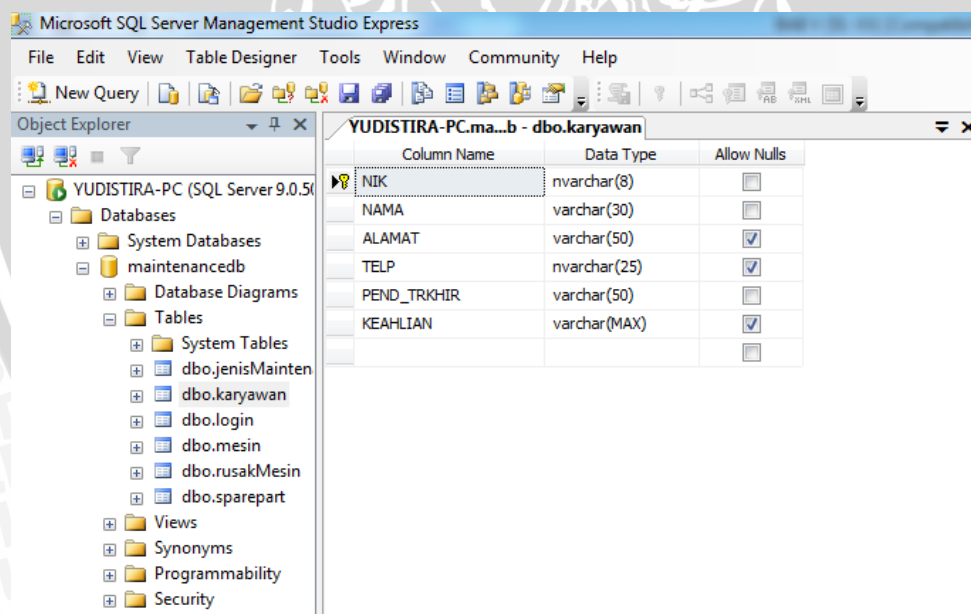
5.2. Implementasi (*Implementation*)

Tahap implementasi merupakan tahap dimana penerapan semua hasil desain pada tahap sebelumnya dibuat ke dalam bentuk program aplikasi berbasis komputer. Pada perancangan sistem informasi manajemen pemeliharaan mesin ini, implementasi program dilakukan dengan melakukan pembuatan basis data yang dalam hal ini digunakan pengimplementasian *pseudocode* ke dalam kode program dan pembuatan *user interface* berdasarkan rancangan menu yang sudah dibuat sebelumnya.

5.2.1. Pengembangan *Database*

Pengembangan *database* untuk prototipe sistem informasi manajemen pemeliharaan mesin ini dibuat untuk menunjukkan relasi antar tabel yang digunakan dalam sistem informasi manajemen pemeliharaan mesin yang dibuat. Pengerjaan *database* sistem informasi ini dilakukan pada Microsoft SQL Server 2005, mulai dari pembuatan mulai dari pembuatan tabel sampai dengan relasi antar tabel entitas. Berikut tampilan salah satu pengembangan *database* untuk salah satu tabel entitas, yaitu entitas karyawan.

1. Pembuatan tabel



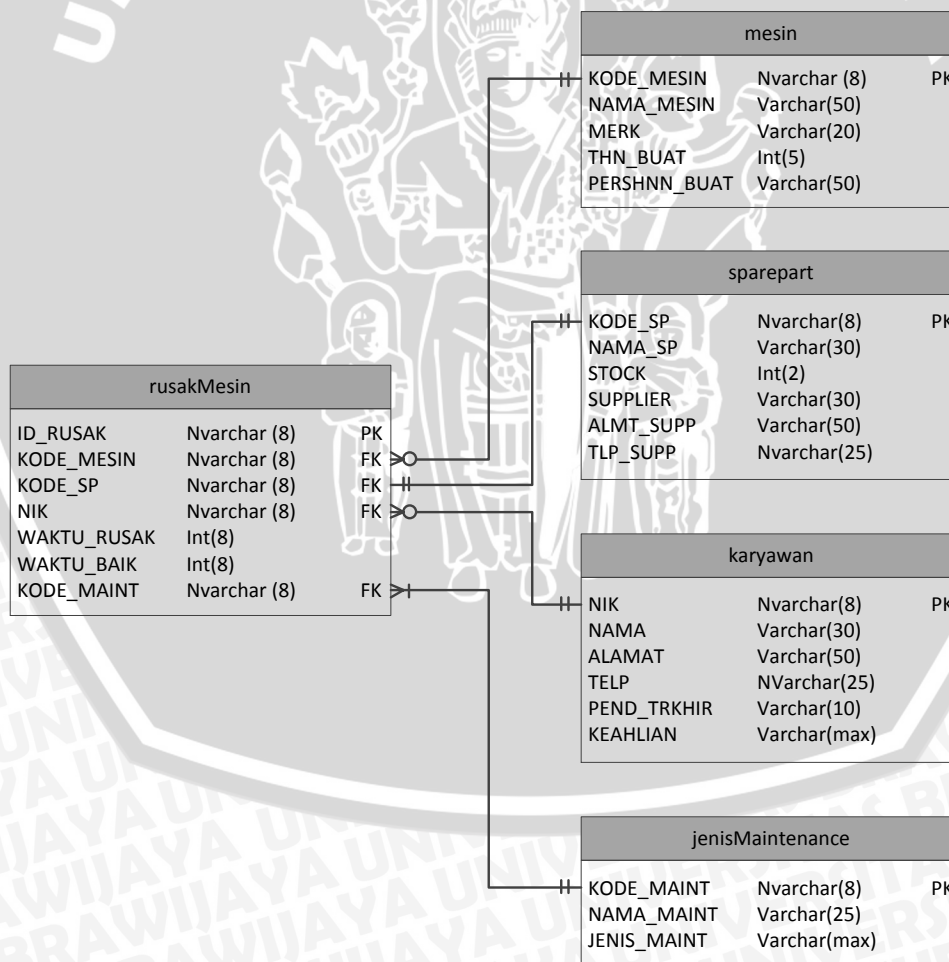
Gambar 5.6 Tampilan pembuatan tabel entitas pada Microsoft SQL Server 2005.

2. Pengisian tabel entitas

NIK	NAMA	ALAMAT	TELP	PEND_TRKHIR	KEAHLIAN
KR-0001	Bambang Wirat...	Jln. Kemiri 10 Bu...	-	SMA / STM	Perawatan mesi...
KR-0002	Choiron	Jln. Jenggolo 24...	-	SMA / STM	Memperbaiki me...
KR-0003	Junarto	Jln. Ahmad Yani ...	-	SMA / STM	ahli dalam memp...
KR-0004	Solikhin	Jln. Suwafi Timu...	08155573451	SMA / STM	perawatan mesi...
KR-0005	Agus Salim	Jln. Jenggolo Gg...	-	SMA / STM	perawatan untu...
* NULL	NULL	NULL	NULL	NULL	NULL

Gambar 5.7 Tampilan pengisian tabel entitas pada Microsoft SQL Server 2005.

3. Relasi antar tabel entitas



Gambar 5.8 Tampilan relasi antar tabel entitas dalam SIM pemeliharaan mesin.

5.2.2. Pengembangan Modul Program

Dalam merealisasikan aplikasi sistem informasi manajemen pemeliharaan mesin-mesin produksi di PT. Sekar Laut Tbk. secara keseluruhan, diperlukan pembuatan *listing program* yang diterapkan pada Microsoft Visual Studio 2008 dengan bahasa Basic. *Listing program* yang akan disampaikan bukan merupakan *listing program* secara keseluruhan dari aplikasi. Namun, hanya beberapa bagian yang dianggap penting saja yang disampaikan dalam bagian ini.

```

Sub BuatKodeMesin()
    KoneksiDatabase()
    Dim ambilTengah As String
    Dim recTerakhir As String
    Dim kodeMesin As Integer
    Try
        koneksi.Open()
        dtSet = New DataSet
        sql = "select*from mesin"
        dtAdapter.SelectCommand = New SqlCommand(sql, koneksi)
        dtAdapter.Fill(dtSet, "mesin")
        PosisiRecord = dtSet.Tables("mesin").Rows.Count - 1
        recTerakhir = dtSet.Tables("mesin").Rows(PosisiRecord)_
            ("KODE_MESIN").ToString()

        ambilTengah = Mid(recTerakhir, 4, Len(recTerakhir) - 1)
        kodeMesin = Val(ambilTengah)
        kodeMesin = kodeMesin + 1
        txtKode.Text = "M-000" & kodeMesin
        koneksi.Close()
    Catch ex As Exception
        MessageBox.Show(ex.Message)
    End Try
End Sub

```

Gambar 5.9 *Syntax* untuk menambah data pada entitas mesin.

```

Sub SearchRecord()
    If txtCari.Text = "" Then
        MsgBox("Isikan kata kunci pencarian!!!", , "Searching_
Record")
        txtCari.Focus()
        Exit Sub
    End If
    Try
        If rbKode_Mesin.Checked = True Then
            dtAdapter = New SqlDataAdapter("select KODE_MESIN_
as [Kode Mesin], NAMA_MESIN as [Nama Mesin], MERK as [Merk],_
THN_BUAT as [Tahun Produksi], PERSHNN_BUAT as [Perusahaan_
(Produsen)] from mesin where KODE_MESIN='" & txtCari.Text & "'",_
koneksi)
        Else
            dtAdapter = New SqlDataAdapter("select KODE_MESIN_
as [Kode Mesin], NAMA_MESIN as [Nama Mesin], MERK as [Merk],_
THN_BUAT as [Tahun Produksi], PERSHNN_BUAT as [Perusahaan_
(Produsen)] from mesin where NAMA_MESIN like '" & txtCari.Text &_
"%'", koneksi)
        End If
        Dim tbMesin As New DataTable
        tbMesin.Clear()
        dtAdapter.Fill(tbMesin)
        If tbMesin.Rows.Count = Nothing Then
            MsgBox("Data tidak ditemukan, isikan input yang_
tepat....!!!", MsgBoxStyle.Information, "Perhatian")
            txtCari.Clear()
            txtCari.Focus()
        End If
        dgvMesin.DataSource = tbMesin
    Catch ex As Exception
        MsgBox(ex.Message, MsgBoxStyle.Information, "Perhatian")
    End Try
End Sub

```

Gambar 5.10 *Syntax* untuk mencari data pada SIM entitas mesin.



```

wakturusak = Val(txtMTTF.Text)

For i As Integer = 1 To 31

    tanggalrusak = DateAdd("n", wakturusak, #9/24/2012#)

    sql2 = "insert into reportmesintanggal values ('" & _
    tanggalrusak & "','" & cbKodeMesin.Text & "','" & _
    cbKodeSparepart.Text & "','" & cbIDKaryawan.Text & _
    "','" & cbKodeMaint.Text & "')"
    cmd3 = New SqlCommand(sql2, koneksi)
    cmd3.Connection = koneksi
    cmd3.ExecuteNonQuery()

    wakturusak = wakturusak + Val(txtMTTF.Text) + _
    Val(txtMTTR.Text)

    sql3 = "select Tanggal_Rusak from reportmesintanggal_
    order by Tanggal_Rusak asc"
    cmd4 = New SqlCommand(sql3, koneksi)
    cmd4.Connection = koneksi
    cmd4.ExecuteNonQuery()

Next i

dtAdapter.Fill(dtSet, "reportmesintanggal")

```

Gambar 5.11 *Syntax* untuk penjadwalan pemeliharaan mesin produksi.

5.2.3. Pengembangan *User Interface*

Dalam pembahasan sebelumnya pada tahap desain atau perancangan, pengembangan *user interface* ini dirancang dan dikembangkan untuk menjadi saran bagi *user* agar dapat berinteraksi dengan sistem yang dibuat. Berikut ini adalah tampilan dari *form* penjadwalan *maintenance* dari sistem informasi manajemen pemeliharaan mesin-mesin produksi, sedangkan untuk tampilan yang lain dapat dilihat pada lampiran.

Form penjadwalan *maintenance* yang dibuat berisikan tentang tabel jadwal *maintenance* mesin-mesin produksi perusahaan. Jadwal *maintenance* ini dapat dilihat dengan memasukkan atau memilih data berdasarkan nama mesin-mesin produksi atau waktu (bulan dan tahun) *maintenance* yang diinginkan.

Tanggal Maintenance	Nama Mesin	Nama Sparepart	Nama Karyawan	Jenis Maintenance
09/30/2012 12:24 PM	Mesin Steamer (Mesin Pengukus)	Rantai RS-60	Agus Salim	Steamer
10/07/2012 3:13 AM	Mesin Steamer (Mesin Pengukus)	Rantai RS-60	Agus Salim	Steamer
10/13/2012 6:02 PM	Mesin Steamer (Mesin Pengukus)	Rantai RS-60	Agus Salim	Steamer
10/20/2012 8:51 AM	Mesin Steamer (Mesin Pengukus)	Rantai RS-60	Agus Salim	Steamer
10/26/2012 11:40 PM	Mesin Steamer (Mesin Pengukus)	Rantai RS-60	Agus Salim	Steamer
11/02/2012 2:29 PM	Mesin Steamer (Mesin Pengukus)	Rantai RS-60	Agus Salim	Steamer
11/09/2012 5:18 AM	Mesin Steamer (Mesin Pengukus)	Rantai RS-60	Agus Salim	Steamer
11/15/2012 8:07 PM	Mesin Steamer (Mesin Pengukus)	Rantai RS-60	Agus Salim	Steamer

Gambar 5.12 Form penjadwalan maintenance.

5.3. Pengujian (Testing)

Tahap pengujian sistem adalah tahap untuk menerapkan semua desain sistem yang telah dibuat ke dalam bentuk program aplikasi berbasis komputer. Pada tahap pengujian aplikasi sistem informasi manajemen pemeliharaan mesin produksi ini menggunakan tiga macam pengujian, yaitu uji verifikasi, uji validasi dan uji prototipe.

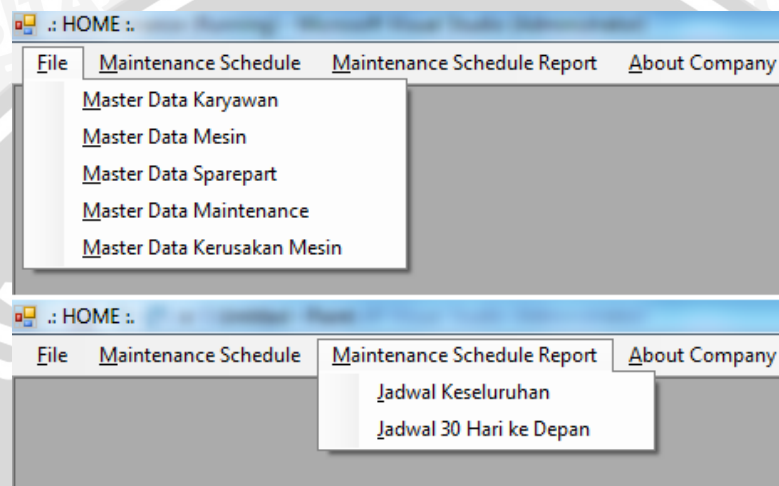
5.3.1. Uji Verifikasi

Uji verifikasi dilakukan dengan tujuan untuk mengetahui apakah penerjemahan model konseptual menjadi aplikasi sistem informasi manajemen pemeliharaan mesin-mesin produksi telah dilakukan secara benar atau tidak. Uji verifikasi dilakukan dengan membandingkan desain *database* dengan *pseudocode* yang telah dibuat, juga dengan membandingkan antara *listing program* dengan *pseudocode* yang telah dibuat. Proses perbandingan ini dilakukan dengan tujuan untuk mengetahui kesinambungan antara desain *database*, *pseudocode* dan *coding (listing program)* saat program dijalankan. Untuk uji verifikasi ini, proses yang akan diuji adalah mulai proses mengubah data (menambah, menyimpan, memperbarui, dan menghapus data), proses pencarian data sampai proses penentuan penjadwalan pemeliharaan mesin-mesin produksi. Dan dilakukan juga membandingkan antara menu utama sistem informasi pemeliharaan mesin produksi dengan rancangan hierarki menu utama yang telah dibuat sebelumnya.

Proses membandingkan antara desain *pseudocode* dengan *listing program* dimulai dari *pseudocode* yang ditunjukkan Gambar 5.3 sampai Gambar 5.5 diimplementasikan dengan *listing program* yang ditunjukkan Gambar 5.9 sampai

Gambar 5.11. Apabila *listing program* dari proses mengubah data (menambah, menyimpan, memperbarui, dan menghapus data), proses pencarian data serta perhitungan dan penjadwalan mesin pada program sistem informasi manajemen pemeliharaan mesin produksi telah sesuai dengan *pseudocode* yang telah dibuat maka dapat disimpulkan bahwa telah terjadi kesesuaian antara *listing program* dengan *pseudocode* saat program sistem informasi manajemen pemeliharaan dijalankan.

Sedangkan untuk rancangan hierarki menu seperti yang ditunjukkan pada Gambar 5.1 diimplementasikan oleh Gambar 5.13 berikut.



Gambar 5.13 Menu utama sistem informasi manajemen pemeliharaan mesin produksi.

Pada Gambar 5.13 terlihat bahwa terdapat 4 menu utama pada sistem informasi ini, yaitu *File*, *Maintenance Schedule*, *Maintenance Schedule Report* dan *About Company*. Pada menu *File* terdapat 5 (lima) sub menu master data. Sedangkan untuk menu *Maintenance Schedule Report* terdapat 2 (dua) sub menu jadwal keseluruhan dan jadwal 30 hari ke depan. Hal ini menunjukkan bahwa menu utama pada program sistem informasi manajemen pemeliharaan mesin telah sesuai dengan rancangan hierarki menu utama yang telah dibuat.

Berdasarkan penjelasan kondisi-kondisi tersebut, maka dapat disimpulkan bahwa prototipe hasil perancangan sistem informasi manajemen pemeliharaan mesin produksi yang dibuat telah terealisasi dan sesuai dengan perancangan konseptual yang telah dilakukan sebelumnya.

5.3.2. Uji Validasi

Uji validasi secara umum bertujuan untuk menguji apakah sistem *database* yang telah dibuat mampu memberikan informasi yang diharapkan oleh *user*, sehingga dapat berfungsi sepenuhnya sebagai sistem *database* dan dapat membantu *user* mengenai

penjadwalan pemeliharaan mesin-mesin produksi yang berdasarkan lama waktu mesin tersebut beroperasi dan lama perbaikan ketika mesin tersebut mengalami kerusakan. Untuk melakukan uji validasi, perhitungan manual akan dilakukan terhadap salah satu mesin produksi yang bernama mesin Steamer (Mesin Pengukus).

Berikut perhitungan penentuan penjadwalan *maintenance* dari mesin Steamer (Mesin Pengukus).

Nama	: Mesin Steamer (Mesin Pengukus)
Lama Beroperasi (MTTF)	: 9384 menit
Lama Perbaikan (MTTR)	: 145 menit
(terhitung mulai tanggal 24 September 2012)	
Waktu Maintenance ke-1	= MTTF = 9384 menit = 6 hari 12 jam 24 menit = 30 September 2012 pukul 12:24
Waktu Maintenance ke-2	= MTTF + MTTR + MTTF = 9384 + 145 + 9384 = 18913 menit = 13 hari 3 jam 13 menit = 7 Oktober 2012 pukul 3:13
Waktu Maintenance ke-3	= MTTF + MTTR + MTTF + MTTR + MTTF = 9384 + 145 + 9384 + 145 + 9384 = 28442 menit = 19 hari 18 jam 2 menit = 13 Oktober 2012 pukul 18:02
Dan seterusnya.	

Gambar 5.14 Perhitungan manual penentuan penjadwalan *maintenance* mesin.

Berdasarkan perhitungan yang telah dilakukan, didapatkan bahwa waktu *maintenance* mesin pertama, kedua sampai ketiga adalah 30 September 2012 pukul 12:24, 7 Oktober 2012 pukul 3:13, dan 13 Oktober 2012 pukul 18:02. Hasil perhitungan tersebut dibandingkan dengan hasil penjadwalan yang dilakukan oleh program aplikasi sistem informasi manajemen pemeliharaan mesin, sehingga hasil perhitungan dapat dilihat pada Gambar 5.15 berikut.

Form Informasi Maintenance Mesin

| Kunci Pencarian |

Waktu Maintenance

Mesin Produksi

Waktu Maintenance

October / 2012

Mesin Produksi

Mesin Steamer (Mesin Pengukus)

Tabel Jadwal Maintenance

Tanggal Maintenance	Nama Mesin	Nama Sparepart	Nama Karyawan	Jenis Maintenance
09/30/2012 12:24 PM	Mesin Steamer (Mesin Pengukus)	Rantai RS-60	Agus Salim	Steamer
10/07/2012 3:13 AM	Mesin Steamer (Mesin Pengukus)	Rantai RS-60	Agus Salim	Steamer
10/13/2012 6:02 PM	Mesin Steamer (Mesin Pengukus)	Rantai RS-60	Agus Salim	Steamer
10/20/2012 8:51 AM	Mesin Steamer (Mesin Pengukus)	Rantai RS-60	Agus Salim	Steamer
10/26/2012 11:40 PM	Mesin Steamer (Mesin Pengukus)	Rantai RS-60	Agus Salim	Steamer
11/02/2012 2:29 PM	Mesin Steamer (Mesin Pengukus)	Rantai RS-60	Agus Salim	Steamer
11/09/2012 5:18 AM	Mesin Steamer (Mesin Pengukus)	Rantai RS-60	Agus Salim	Steamer
11/15/2012 8:07 PM	Mesin Steamer (Mesin Pengukus)	Rantai RS-60	Agus Salim	Steamer

View Data RESET EXIT

Gambar 5.15 Hasil penjadwalan *maintenance* mesin Steamer.

Berdasarkan hasil penjadwalan *maintenance* mesin Steamer pada Gambar 5.15, dapat disimpulkan bahwa hasil pengujian dikatakan valid. Hal ini dikarenakan oleh hasil perhitungan manual maupun menggunakan program aplikasi menghasilkan nilai dan informasi yang sama, sehingga dapat dikatakan pula bahwa sistem informasi manajemen pemeliharaan mesin produksi yang dibuat telah mampu merepresentasikan tujuan perancangan yaitu untuk merancang sebuah sistem informasi manajemen pemeliharaan mesin berbasis komputer yang dapat memberi informasi-informasi kepada pihak Departemen *Process Control & Engineering* PT. Sekar Laut Tbk. dalam mengelola sistem pemeliharaan mesin-mesin produksi.

5.3.3. Uji Prototipe

Pengujian prototipe dilakukan dengan tujuan untuk mengetahui apakah prototipe yang dibuat telah memenuhi kebutuhan pengguna atau belum. Pada pengujian ini juga dideskripsikan mengenai kelebihan dari sistem yang baru jika dibandingkan dengan sistem lama atau yang telah ada sebelumnya. PT. Sekar Laut Tbk. sebelumnya tidak memiliki sistem informasi mengenai pemeliharaan mesin-mesin produksi sehingga dalam menjadwalkan *maintenance* mesin sendiri pun dilakukan secara manual, baik perhitungan penentuan jadwal maupun penulisan jadwal *maintenance* mesin.

Perbandingan antara proses manual dengan proses yang dilakukan dengan prototipe sistem informasi manajemen pemeliharaan mesin dapat dilihat pada tabel berikut.

Tabel 5.6 Perbandingan Performa Antara Sistem Lama Dan Sistem Baru

Unsur	Sistem Lama	Sistem Baru
Kecepatan	Pengerjaan perhitungan penentuan penjadwalan <i>maintenance</i> mesin produksi dan penulisan jadwal <i>maintenance</i> pun masih dilakukan secara manual. Sehingga dibutuhkan waktu sekitar 10-15 menit sebelum memberikan informasi mengenai penjadwalan <i>maintenance</i> .	Data-data master sistem informasi ini terkumpul dalam satu <i>database</i> sistem informasi manajemen pemeliharaan mesin, sehingga penentuan jadwal dan menampilkan jadwal <i>maintenance</i> serta menginformasikan data-data lain dapat langsung diketahui. Proses dengan sistem baru membutuhkan waktu sekitar 5-10 detik.
Fleksibilitas	Tidak fleksibel, karena sistem lama masih dilakukan secara manual dan data-data pendukung <i>maintenance</i> masih sering kurang lengkap dan belum terintegrasi jadi satu sistem.	Lebih fleksibel, karena data yang dibutuhkan untuk menentukan penjadwalan <i>maintenance</i> mesin dan memberikan informasi lainnya telah terintegrasi ke dalam satu sistem informasi manajemen pemeliharaan mesin produksi.
Ketelitian	Rentannya terjadi kesalahan dalam perhitungan mulai dari menentukan waktu kerusakan sampai penentuan jadwal <i>maintenance</i> mesin produksi. Dan dikarenakan masih manual sehingga belum mampu memprediksi atau menjadwalkan <i>maintenance</i> untuk jangka waktu yang lebih lama.	Tingkat terjadinya kesalahan lebih kecil karena sistem informasi ini mampu melakukan kalkulasi (perhitungan) secara otomatis sehingga lebih cepat dan akurat dalam memberikan informasi penjadwalan <i>maintenance</i> dan informasi-informasi lainnya yang berkaitan dengan <i>maintenance</i> mesin.

Berdasarkan penjelasan tersebut, terbukti bahwa sistem informasi manajemen pemeliharaan yang dibuat mampu mempercepat proses penentuan jadwal *maintenance* mesin dan pemberian informasi-informasi lainnya mengenai *maintenance* mesin kepada pihak manajemen Departemen *Process Control & Engineering* PT. Sekar Laut Tbk. Hal lain juga sistem informasi ini didukung dengan adanya *user interface* yang bersifat *user friendly* karena mudah digunakan dan dipahami oleh *user*. Dengan beberapa kelebihan tersebut, diharapkan prototipe ini mampu memenuhi kebutuhan *user*.