**BAB IV**

**PEMBAHASAN**

**4.1. Aktivitas selama PKL**

Aktivitas yang dilakukan selama Praktek Kerja Lapangan pada PT Astra International tbk Auto 2000 yaitu :

1. Melakukan penginputan nama mekanik yang akan melakukan *service*.
2. Melakukan *start* pada sistem saat mekanik mulai memberikan PKB sebagai tanda bahwa pekerjaan sedang dilakukan.
3. Melakukan *stop* pada sistem saat mekanik mulai memberikan PKB sebagai tanda bahwa mekanik telah selesai melakukan *service*.
4. Melakukan pencatatan terhadap nama-nama mekanik beserta mobil yang di-*service.*
5. Memberikan jumlah jam yang didapat masing-masing mekanik sesuai jenis pekerjaan serta perhitungan total jam yang didapat mekanik selama 1 hari.
6. Melakukan perhitungan jumlah mobil yang di-*service* ke dalam buku harian.
7. Berperan aktif dalam kegiatan yang terdapat pada bagian Pembagi Tugas Mekanik.

**4.2. Analisis Sistem yang sedang berjalan**

Sistem yang saat ini sedang berjalan pada PT Astra International tbk Auto 2000 cabang Raden intan masih memiliki kelemahan yaitu di bagian pembagi tugas mekanik terdapat 1 komputer yang menggunakan *Microsoft Excel* dalam pemberian jam kerja. Karena pemberian jam kerja di-*input* menggunakan *Ms. Excel*, maka dikhawatirkan akan terdapat kesalahan pemberian jumlah jam kerja yang harus dilakukan secara teliti pada masing-masing mekanik..

Berikut ini sistem yang sedang berjalan pada PT Astra Internationa tbk Auto 2000 cabang Raden Intan.

1. Dimulai dari bagian *Service Advisor* yang menerima *customer* dan memeriksa buku servis atau buku *warranty* sebagai penyetujuan atau tidaknya *claim* dari *customer.*
2. *Service advisor* membuatkan Perintah Kerja Bengkel 3 rangkap berdasarkan data *customer* dan data kendaraan berdasarkan buku servis yanag dibawa oleh *customer,* rangkap 1 diberikan kepada *customer* yang nantinya untuk pengambilan kendaraan, rangkap 3 untuk bagian *foreman* dan mekanik dan rangkap 2 bagian *Service Advisor* untuk diarsipkan sementara berdasarkan nomor urut.
3. Bagian mekanik menerima rangkap 3 lalu mendiagnosa kendaraan. Setelah itu diserahkan kepada PTM yang digunakan untuk membagi pekerjaan mekanik dan kendaraan sesuai jam yang tertera pada rangkap 3 PKB tersebut dan melakukan *start* pada sistem sebagai tanda bahwa Mekanik bersangkutan sedang melakukan perawatan.
4. Setelah dilakukan pencatatan pada bagian PTM, mekanik menyerahkan rangkap 3 kepada *foreman* untuk melakukan perawatan kendaraan dan mengambil *Part* di bagian *Sparepart*.
5. Bagian *Sparepart* melakukan input pembebanan part pada PKB dan membuatkan Bukti Pengeluaran Suku Cadang rangkap 2 berdasarkan Perintah Kerja Bengkel yang diterima dari *Foreman* atau mekanik, rangkap 1 diarsipkan berdasarkan tanggal dan rangkap 2 diberikan kepada *Foreman* atau mekanik beserta *Part*.
6. *Foreman* dan mekanik lalu melakukan pemasangan *Part* dan melakukan *Chek* kendaraan. Perintah Kerja Bengkel (PKB) serta Bukti Pengeluaran Suku Cadang (BPSC) diserahkan kembali ke bagian PTM untuk dapat dilakukan *stop* pada sistem.
7. Kemudian *Foreman* melakukan *Test Drive* dan *Final Chek.*
8. Setelah selesai Perintah Kerja Bengkel (PKB) serta Bukti Pengeluaran Suku Cadang (BPSC) diserahkan *Service Advisor.*
9. *Service Advisor* menyatukan PKB rangkap 2 dan rangkap 3 serta Bukti Pengeluaran Suku Cadang dan mengubah status PKB menjadi TECO lalu diberikan ke bagian *Billing Service.*
10. *Billing Service* menerima PKB rangkap 2 dan 3 serta BPSC, lalu membuatkan Surat Izin Keluar Kendaraan (SIKK) 2 rangkap. Rangkap 1 diberikan kepada bagian *Service Advisor* beserta PKB rangkap 3 lalu PKB rangkap 2, BPSC dan SIKK diarsipkan berdasarkan tanggal, lalu merubah status TECO menjadi UNTECO.
11. *Service Advisor* memberikan SIKK kepada *Customer* sebagai surat keluar kendaraan dan PKB*.*
    * 1. **Bagan Alir Dokumen (BAD)**

35

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Bagian *Service Advisor*** | **Bagian Formen dan Mekanik** | **Bagian PTM (Pembagi Tugas Mekanik)** | **Bagian *Spare Part*** | **Bagian *Billing service*** |
|  |  |  |  |  |

**Gambar 4.1** Bagan Alir Dokumen pada

PT Astra Internationa tbk Auto 2000 cabang Raden Intan

Keterangan :

* PKB : Perintah Kerja Bengkel
* BPSC : Bukti Pengeluaran Suku Cadang
* SIKK : Surat Izin Keluar Kendaraan

37

* 1. **Analisis Sistem**

Terdapat beberapa analisis sistem yang dapat dipakai, dalam penelitian ini penulis menggunakan analisis sistem PIECES. Adapun penjelasan dan analisis PIECES adalah sebagai berikut :

* + 1. **Analisis PIECES**

Untuk mengidentifikasi masalah, haruslah dilakukan analisis terhadap kinerja, informasi, ekonomi, keamanan aplikasi, efisiensi dan pelayanan kepada pegawai. Analisis yang digunakan dalam perancangan pemilihan kualitas bibit karet terbaik ini menggunakan metode PIECES (*Performance*, *Information*, *Economy*, *Control*, *Efficiency* dan *Security*). Dari hasil analisis ini akan diperoleh berbagai usulan untuk membantu perancangan sistem yang lebih baik.

1. **Analisis Kinerja (*Performance*)**

Kinerja dapat diukur dari banyaknya volume pekerjaan yang dilakukan dalam satuan waktu.

**Tabel 4.1 Analisis Kinerja**

|  |  |
| --- | --- |
| No | Kinerja Sistem Lama |
| 1 | Waktu yang dibutuhkan untuk melakukan penginputan mekanik beserta jam dan pekerjaannya kurang lebih 2-4 menit dan bisa lebih lama dari itu ketika jumlah *flate rate* |

1. **Analisis Informasi (*Information*)**

Dalam penyajian informasi seperti laporan pada akhir bulan sering terjadi keterlambatan, karena harus melakukan peng-*edit*-an terlebih dahulu terhadap data, karena tidak terhubung satu sama lain.

**Tabel 4.2 Analisis informasi**

|  |  |
| --- | --- |
| No | Informasi Sistem Yang Lama |
|  | Apabila seorang pegawai yang kurang paham dan membutuhkan data maka harus mencari dalam *file* yang lain sehingga informasi tidak tepat waktu. |

1. **Analisis Ekonomi (*Economy*)**

Persoalan ekonomi berkaitan dengan masalah biaya dan manfaat, Pengembangan sistem diharapkan dapat memberikan banyak manfaat dan biaya yang terjadi bisa diminimalisasi sekecil mungkin.

**Tabel 4.3 Analisis Ekonomi**

|  |  |
| --- | --- |
| No | Ekonomi Sistem Yang Lama |
| 1. | Karena tersimpan dalam *file* di dokumen, jika *file* mengalami kerusakan atau kehilangan maka proses cetak mengeluarkan biaya tambahan seperti perbaikan komputer. |

1. **Analisis Pengendalian (*Control)***

Kontrol atau pengendalian dalam sebuah sistem sangat diperlukan yaitu untuk menghindari dan mendeteksi secara dini terhadap kesalahan-kesalahan yang terjadi serta untuk menjamin keamanan data atau informasi. Dengan adanya pengendalian maka tugas-tugas atau kinerja yang mengalami gangguan bisa diatasi.

**Tabel 4.4 Analisis pengendalian**

|  |  |
| --- | --- |
| No | Pengendalian Sistem Yang Lama |
| 1.  2. | Tidak ada proteksi terhadap data sehingga kehilangan data sering terjadi.  Pengendalian terhadap data relatif sulit karena penyimpanan data masih dalam bentuk *file* *Ms. Excel* dan siapa saja dapat mengakses. |

1. **Analisis Efisiensi (*Efficiency*)**

Efisiensi erat hubungannya dengan input. Tidak dapat dipungkiri kalau tingkat ketelitian dan kemampuan manusia cukup terbatas sedangkan pengolahan data dan laporan masih mengandalkan pada kemampuan manusia. Hal ini dapat mengakibatkan kesalahan.

**Tabel 4.5 Analisis efisiensi**

|  |  |
| --- | --- |
| No | Efisiensi Sistem Yang Lama |
| 1. | Penyampaian informasi terhadap hasil kerja mekanik kurang tepat guna karena memerlukan sangat ketelitian ketika melakukan penginputan. |

1. **Analisis Pelayanan (*Service*)**

Pelayanan dari segi informasi mempunyai sasaran baik bagi pihak yang membutuhkan. Koordinasi yang belum teratur mengenai pengolahan data, penyimpanan laporan serta dokumentasi akan menurunkan kualitas pelayanan yang pada akhirnya akan mengakibatkan kerja dari bagian tersebut menjadi terganggu sehingga pelayanan perlu ditingkatkan.

**Tabel 4.6 Analisis Pelayanan**

|  |  |
| --- | --- |
| No | Pelayanan Sistem Yang Lama |
| 1. | Pemberian jam yang diberikan masih kurang memuaskan bagi mekanik karena proses penginputan data masih lambat dan dengan hati-hati karena tidak terdapat *database*. |

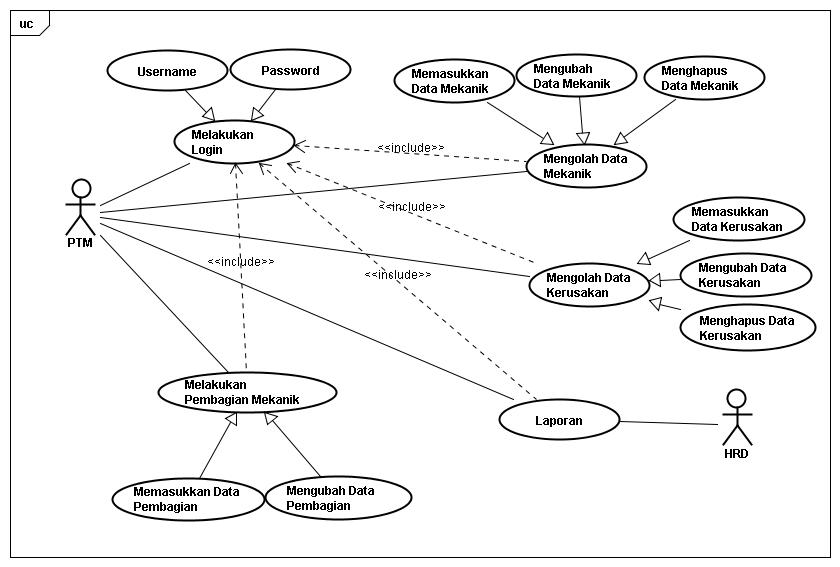
* 1. **Usulan Desain Pengembangan Sistem**

Berdasarkan uraian diatas dapat disimpulkan masih terdapat kelemahan sistem dalam proses penginputan jam mekanik yang berhubungan dengan jenis perawatan memerlukan ketelitian dan harus menggunakan waktu sebaik mungkin karena berhubungan dengan waktu total system (*Lead Time*).

Disini dapat diusulkan penginputan jam pembagian mekanik dengan menggunakan sebuah aplikasi pemrograman yang dapat memudahkan bagian Pembagi Tugas Mekanik untuk melakukan pembagian tugas serta pencarian data ketika terjadi *return*. Adapun rancangan yang baru adalah sebagai berikut :

* 1. ***Use Case Diagram***

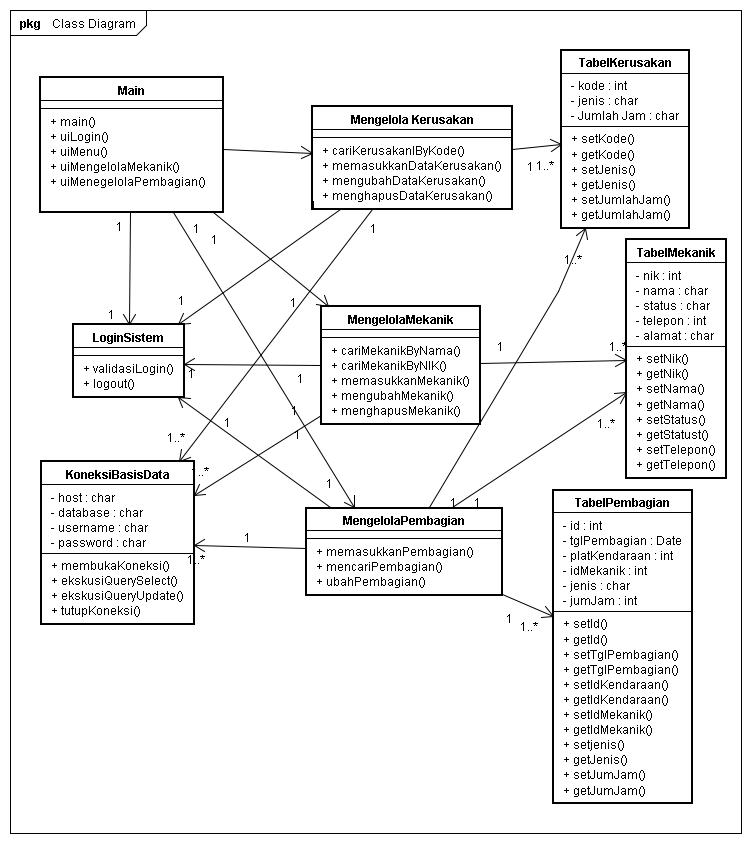
*Use case diagram* menggambarkan fungsionalitas yang diharapkan dari sebuah sistem. Sebuah *use case* merepresentasikan sebuah interaksi antara aktor dengan sistem. Dapat dilihat pada gambar 4.2 terdapat 2 aktor yaitu PTM sebagai pemakai system dan HRD sebagai pimpinan yang mendapatkan laporan hasil pencatatan yang dilakukan oleh PTM. PTM dapat melakukan login, mengolah data mekanik serta melakukan pembagian jam mekanik.



**Gambar 4.2 *Use Case Diagram***

* 1. ***Class Diagram***

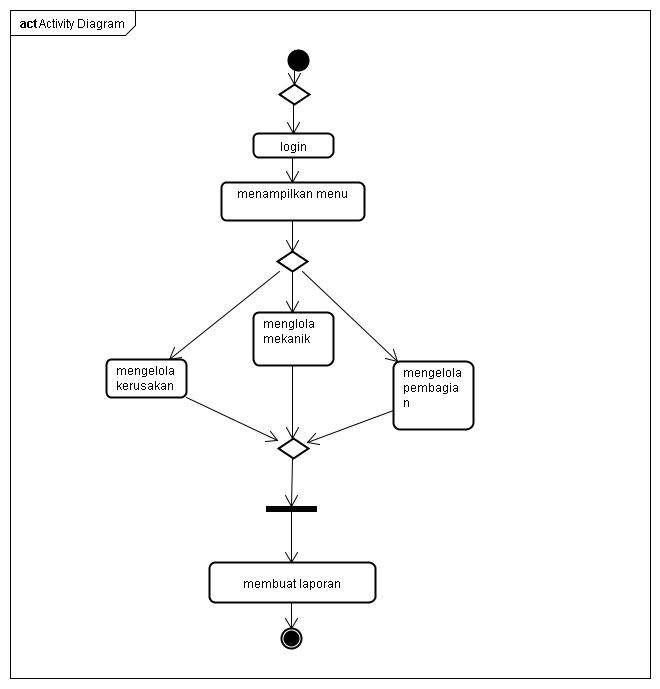
*Class diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Terdapat beberapa class pada class diagram yang terdapat pada gambar 4.3 yaitu *class main* yang berfungsi sebagai kelas yang menangani tampilan dan kelas main sendiri, *login* sistem untuk pendefinisian *use case login*, begitu juga dengan mengelola kerusakan, mengelola mekanik, mengelola pembagian, koneksi basis data untuk koneksi basis data dan melakukan *query* serta 3 tabel yang digunakan sebagai tabel untuk penggunaan *database* yaitu tabel kerusakan, tabel mekanik, dan tabel pembagian.



**Gambar 4.3 *Class Diagram***

* 1. ***Activity Diagram***

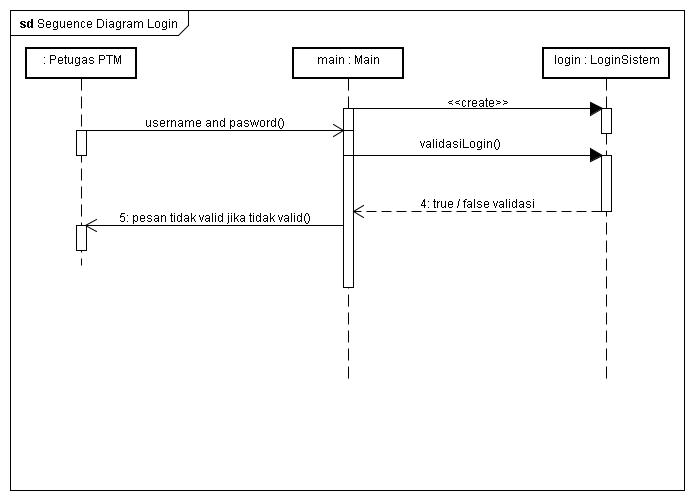
*Activity* Diagram ini menggambarkan rangkaian aliran dari aktivitas, digunakan untuk mendeskripsikan aktifitas yang dibentuk dalam suatu operasi sehingga dapat juga digunakan untuk aktifitas lainnya seperti *use case* atau interaksi.



**Gambar 4.4 *Activity Diagram***

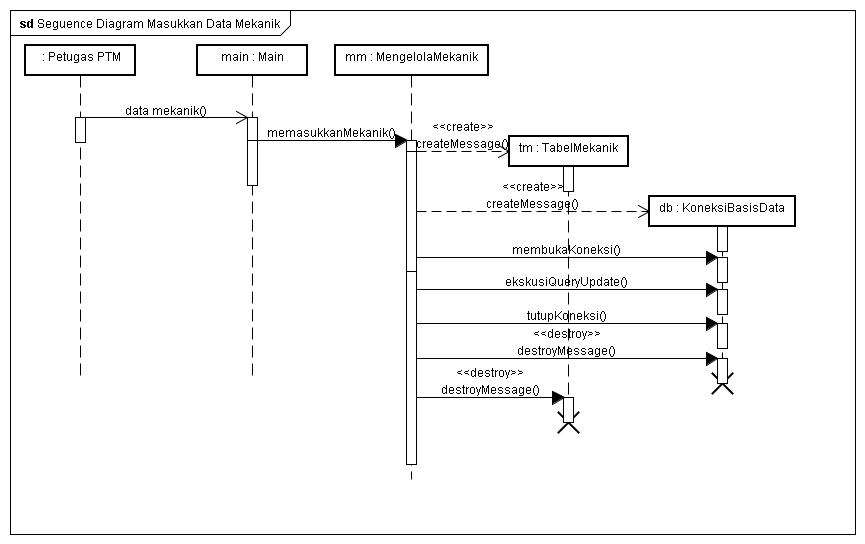
* 1. ***Sequence Diagram***

*Sequence diagram* menggambarkan interaksi antar objek di dalam dan di sekitar sistem berupa *message* yang digambarkan terhadap waktu. Pada bagian ini, diagram terbagi menjadi beberapa bagian sesuai interaksi yang terjadi dalam sistem.

* + - 1. *Sequence Diagram Login*

**Gambar 4.5 *Sequence Diagram Login***

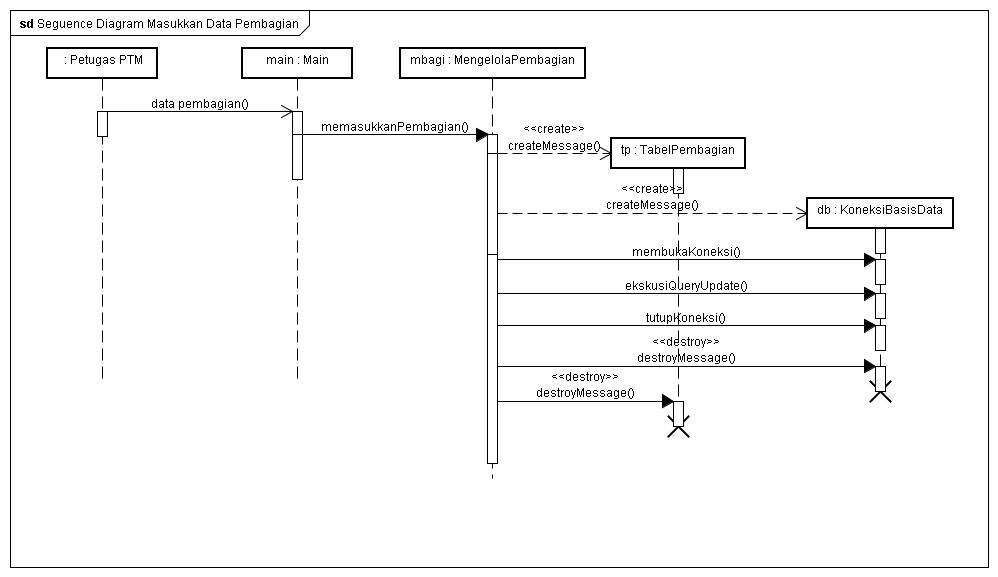
Pada gambar 4.5 kita dapat melihat proses melakukan *login* yang dimulai ketika petugas PTM memasukkan *username* dan *password* kemudian kelas main akan meneruskan pesan ke *login* sistem untuk melakukan validasi. Apabila data yang dimasukkan benar maka pesan akan diteruskan ke menu selanjutnya, tetapi apabila tidak maka akan dikembalikan lagi ke menu *login.*

* + - 1. *Sequence Diagram* Memasukkan Data Mekanik

**Gambar 4.6 *Sequence Diagram* Memasukkan Data Mekanik**

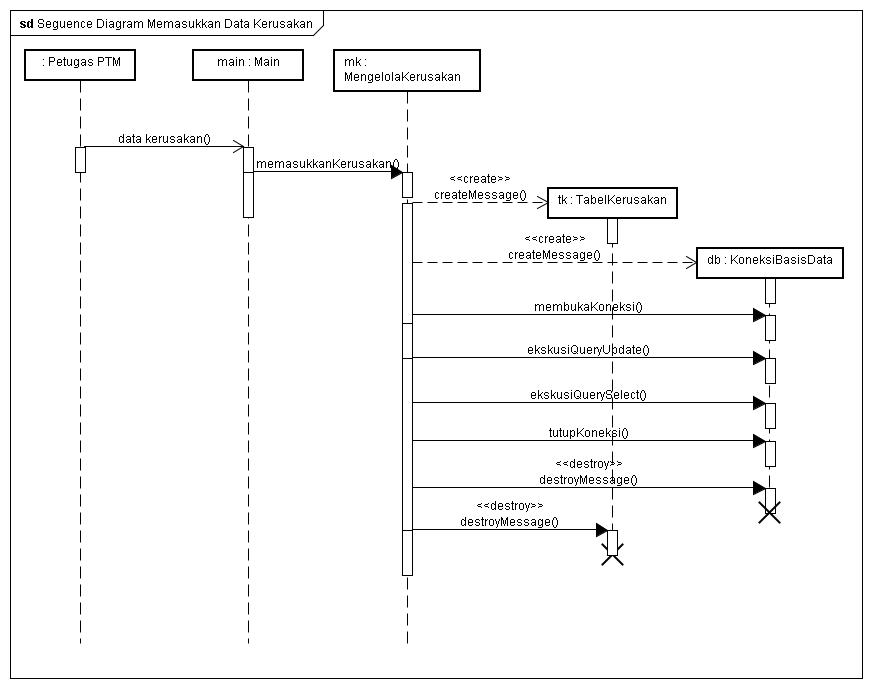
Pada gambar 4.6 kita dapat melihat proses memasukkan data mekanik yang dimulai ketika petugas PTM memasukkan data mekanik kemudian kelas main akan meneruskan pesan ke kelas mengelola mekanik sistem untuk membuat pesan yang akan dikirim ke table mekanik dan melakukan *query* setelah terhubung dengan koneksi basis data.

1. *Sequence Diagram* Memasukkan Data Pembagian

**

**Gambar 4.7 *Sequence Diagram* Memasukkan Data Pembagian**

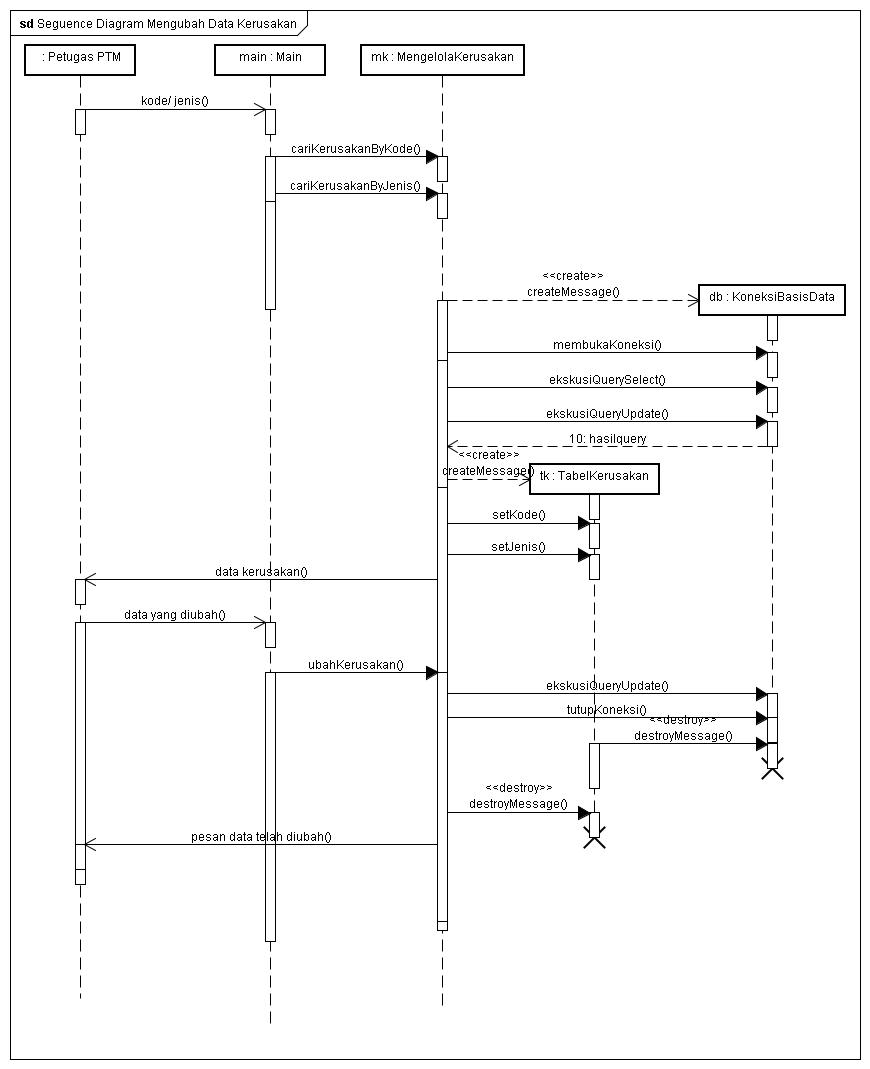
Pada gambar 4.7 kita dapat melihat proses memasukkan data pembagian yang dimulai ketika petugas PTM memasukkan data pembagian kemudian kelas main akan meneruskan pesan ke kelas mengelola pembagian untuk membuat pesan yang akan dikirim ke table mekanik dan melakukan *query* setelah terhubung dengan koneksi basis data.

1. *Sequence Diagram* Memasukkan Data Kerusakan

**Gambar 4.8 *Sequence Diagram* Memasukkan Data Kerusakan**

Pada gambar 4.8 kita dapat melihat proses memasukkan data kerusakan yang dimulai ketika petugas PTM memasukkan data kerusakan, kemudian kelas main akan meneruskan pesan ke kelas mengelola kerusakan untuk membuat pesan yang akan dikirim ke table mekanik dan melakukan *query* setelah terhubung dengan koneksi basis data.

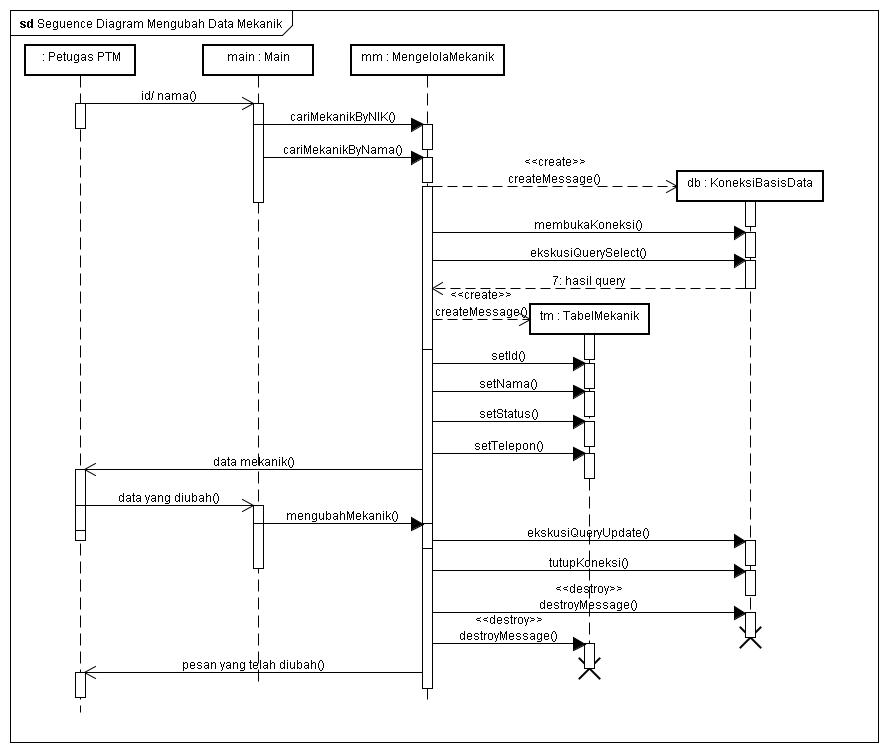
1. *Sequence Diagram* Mengubah Data Kerusakan

**

**Gambar 4.9 *Sequence Diagram* Mengubah Data Kerusakan**

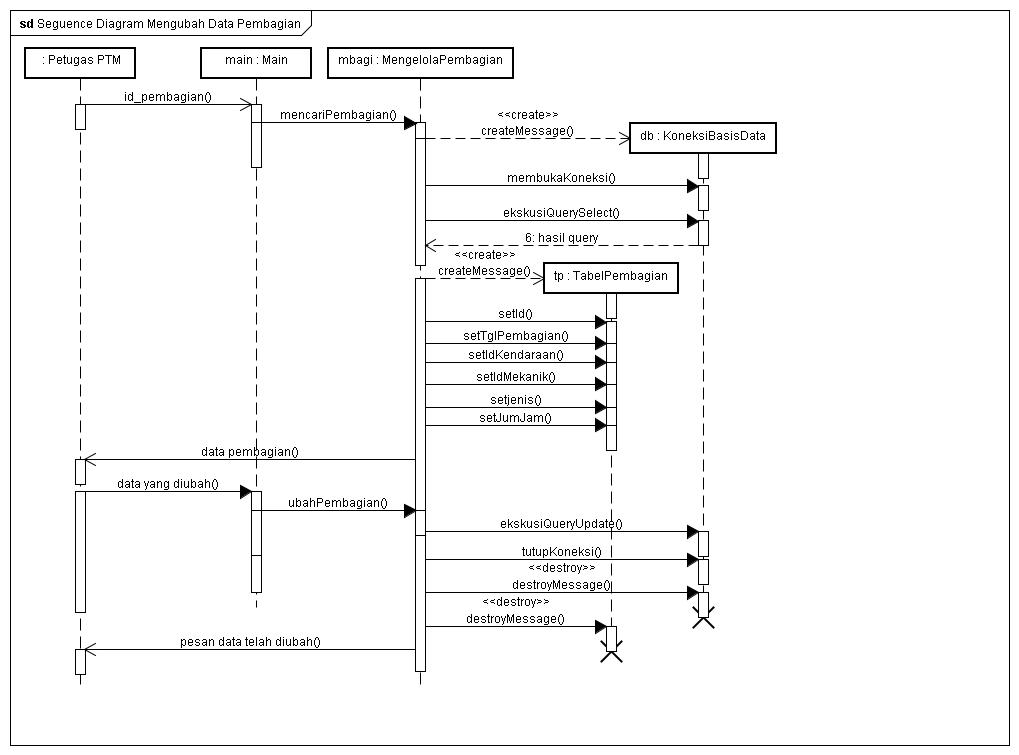
Pada gambar 4.9 kita dapat melihat proses mengubah data kerusakan yang akan dilakukan saat PTM memasukkan data yang akan dilakukan perubahan. Kemudian kelas main akan mencari data berupa kode/jenis kerusakan. Setelah itu kelas main akan meneruskan pesan ke kelas mengelola kerusakan untuk membuat pesan yang akan dikirim ke table mekanik dan melakukan *query* setelah terhubung dengan koneksi basis data. Setelah terhubung dengan tabel kerusakan pesan akan diteruskan ke PTM untuk dapat melakukan perubahan data kerusakan yang selanjutnya akan dikembalikan lagi ke table kerusakan untuk dilakukan perubahan.

1. *Sequence Diagram* Mengubah Data Mekanik



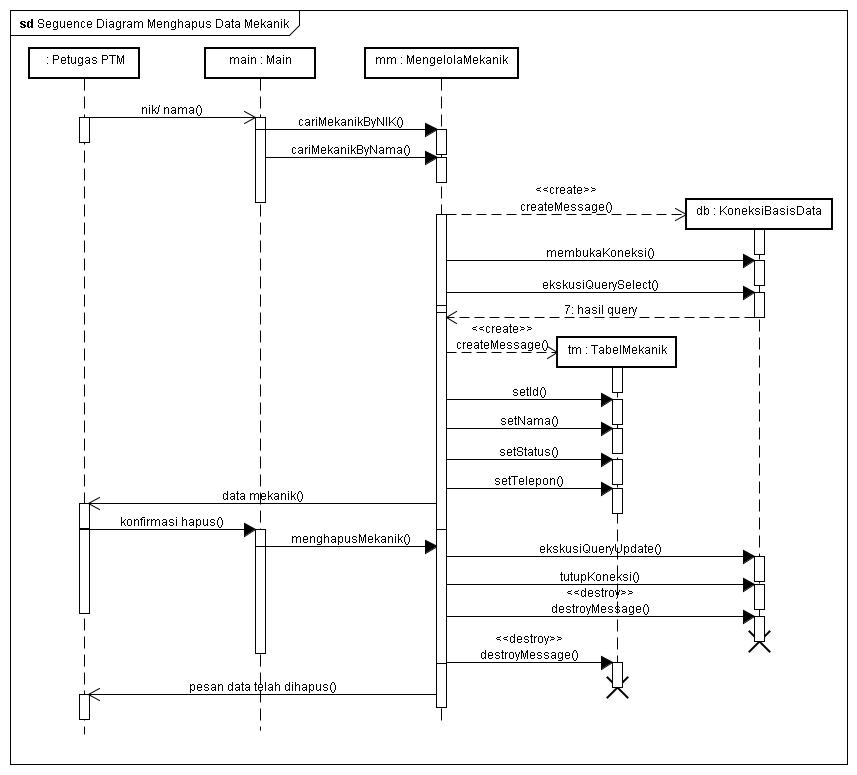
**Gambar 4.10 *Sequence Diagram* Mengubah Data Mekanik**

Pada gambar 4.10 kita dapat melihat proses mengubah data mekanik yang akan dilakukan saat PTM memasukkan data yang akan dilakukan perubahan. Kemudian kelas main akan mencari data berupa id/nama mekanik. Setelah itu kelas main akan meneruskan pesan ke kelas mengelola mekanik untuk membuat pesan yang akan dikirim ke tabel mekanik dan melakukan *query* setelah terhubung dengan koneksi basis data. Setelah terhubung dengan tabel mekanik pesan akan diteruskan ke PTM untuk dapat melakukan perubahan data yang selanjutnya akan dikembalikan lagi ke table kerusakan untuk dilakukan perubahan.

1. *Sequence Diagram* Mengubah Data Pembagian

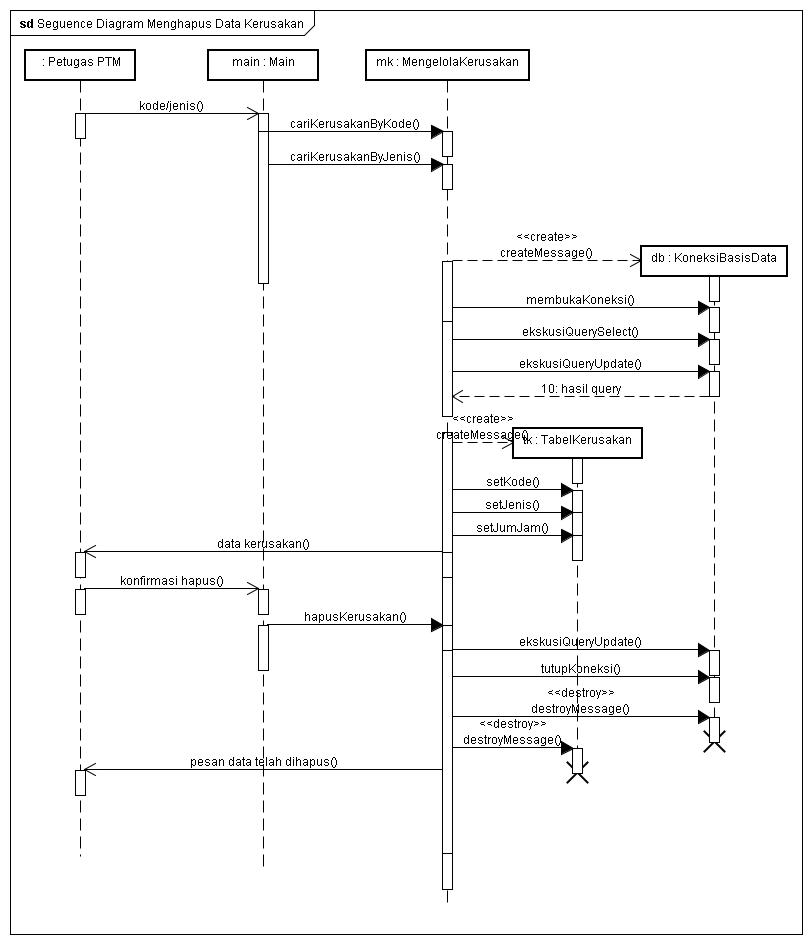
**Gambar 4.11 *Sequence Diagram* Mengubah Data Pembagian**

Pada gambar 4.11 kita dapat melihat proses mengubah data pembagian yang akan dilakukan saat PTM memasukkan data yang akan dilakukan perubahan. Kemudian kelas main akan mencari data berupa id pembagian. Setelah itu kelas main akan meneruskan pesan ke kelas mengelola pembagian untuk membuat pesan yang akan dikirim ke tabel pembagian dan melakukan *query* setelah terhubung dengan koneksi basis data. Setelah terhubung dengan tabel pembagian pesan akan diteruskan ke PTM untuk dapat melakukan perubahan data yang selanjutnya akan dikembalikan lagi ke tabel pembagian untuk dilakukan perubahan.

1. *Sequence Diagram* Menghapus Data Mekanik

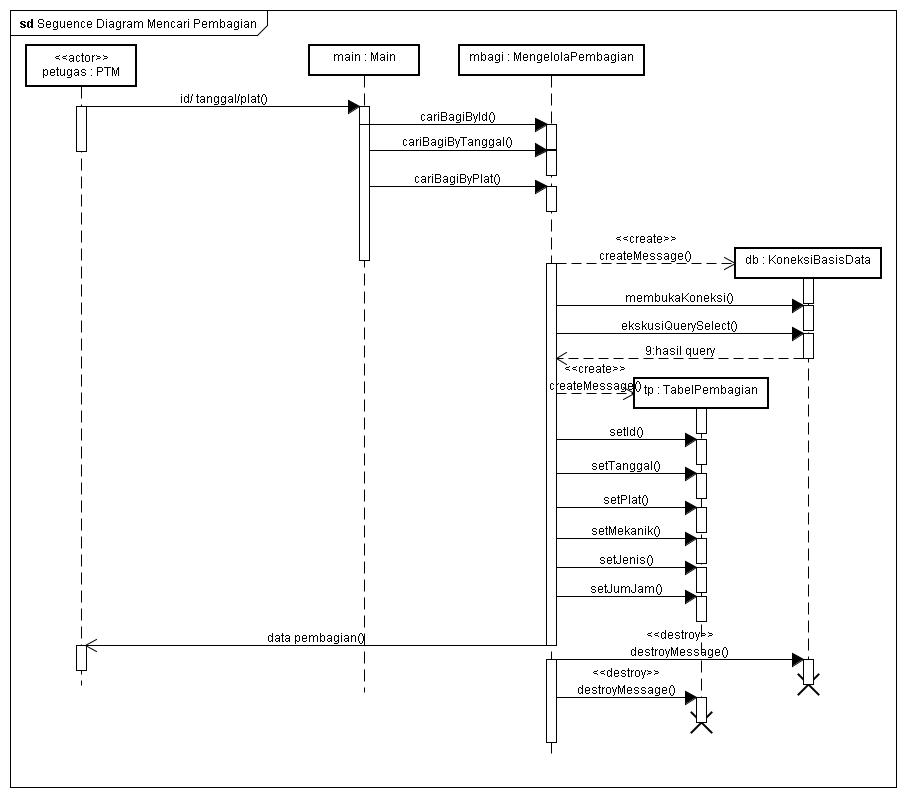
**Gambar 4.12 *Sequence Diagram* Menghapus Data Mekanik**

Pada gambar 4.12 kita dapat melihat proses menghapus data mekanik yang akan dilakukan saat PTM memasukkan data yang akan dihapus. Kemudian kelas main akan mencari data berupa id/nama mekanik. Setelah itu kelas main akan meneruskan pesan ke kelas mengelola mekanik untuk membuat pesan yang akan dikirim ke tabel mekanik dan melakukan *query* setelah terhubung dengan koneksi basis data. Setelah terhubung dengan tabel mekanik pesan akan diteruskan ke PTM untuk dapat melakukan penghapusan data yang selanjutnya akan dikembalikan lagi ke table mekanik untuk dilakukan perubahan.

1. *Sequence Diagram* Menghapus Data Kerusakan

**Gambar 4.13 *Sequence Diagram Login* Menghapus Data Kerusakan**

Pada gambar 4.13 kita dapat melihat proses menghapus data kerusakan yang akan dilakukan saat PTM memasukkan data yang akan dihapus. Kemudian kelas main akan mencari data berupa kode/jenis kerusakan. Setelah itu kelas main akan meneruskan pesan ke kelas mengelola kerusakan untuk membuat pesan yang akan dikirim ke tabel kerusakan dan melakukan *query* setelah terhubung dengan koneksi basis data. Setelah terhubung dengan tabel mekanik pesan akan diteruskan ke PTM untuk dapat melakukan penghapusan data yang selanjutnya akan dikembalikan lagi ke table kerusakan untuk dilakukan perubahan.

1. *Sequence Diagram* Mencari Pembagian

**Gambar 4.14 *Sequence Diagram* Mencari Pembagian**

Pada gambar 4.7 kita dapat melihat proses mencari data pembagian yang dimulai ketika petugas PTM memasukkan data pembagian berupa id pembagian jam. Kemudian kelas main akan meneruskan pesan ke kelas mengelola pembagian untuk membuat pesan yang akan dikirim ke tabel pembagian dan melakukan *query* setelah terhubung dengan koneksi basis data. Apabila data ditemukan maka data akan dimunculkan melalui *class main*.

* 1. **Rancangan Desain *Form***
     1. **Tampilan *Form Login***

*From Login* digunakan untuk masuk ke Menu Utama dengan memasukkan *Username* dan *Password* dengan benar, setelah itu pilih tombol *Login* untuk dapat masuk ke Menu Utama. Apabila terjadi kesalahan dalam penginputan *Username* dan *Password* maka akan muncul pesan *username, password* terjadi kesalahan, silahkan inputkan kembali , pilih tombol *Cancel* jika tidak ingin login dan pilih tombol keluar untuk keluar dari program aplikasi. Tampilan rancangan *form* menu *login* dapat kita lihat pada gambar 4.15.



**Gambar 4.15 Tampilan *Form* Menu *Login***

* + 1. **Tampilan *Form* Menu Utama**

Tampilan *Form* Menu Utama berisi menu-menu pilihan yang berkaitan dengan seluruh aplikasi penggunaan program. Terdapat beberapa menu pada menu utama seperti Menu *File* yang berisi *Form* Data Mekanik dan *Form* Data Kerusakan, Menu Prosesberisi *Input* Pembagian Jam Mekanik, pada menu Laporan terdapat Cetak Laporan, dan pada menu Akun terdapat *Login* baru dan *Form* Ubah *Password.* Tampilan rancangan form menu utama dapat kita lihat pada gambar 4.16.



**Gambar 4.16 Tampilan *Form* Menu Utama**

* + 1. **Tampilan *Form* Data Mekanik**

Data Mekanikdigunakan untuk menginputkan mekanikyang akan memperbaiki kendaraan yang menjadi data untuk acuan untuk dilakukan pembagian data mekanik. Tampilan rancangan *form* data mekanik dapat kita lihat pada gambar 4.17.



**Gambar 4.17 Tampilan *Form* Data Mekanik**

* + 1. **Tampilan *Form* Data Kerusakan**

*Form* ini digunakan untuk menginputkan data kerusakan dimulai dari Kode Kerusakan, Jenis Kerusakan, dan jumlah Flate Rate(Jam). Tampilan rancangan *form* data kerusakan dapat kita lihat pada gambar 4.18.



**Gambar 4.18 Tampilan *Form* Data Kerusakan**

* + 1. **Tampilan *Form* Data Pembagian Jam Mekanik**

*Form* Data Pembagian Jam Mekanikdigunakan untuk melakukan pembagian jam dan pembagian tugas pada mekanik, Tampilan rancangan form data pembagian jam mekanik dapat kita lihat pada gambar 4.19.



**Gambar 4.19 Tampilan *Form* Data Pembagian Jam Mekanik**

* + 1. **Tampilan *Form* Cetak**

*Form* ini digunakan untuk mencetak laporan pembagian jam. Laporan pembagian tugas mekanik dicetak berdasarkan tanggal, tampilan rancangan *form* cetak laporan dapat kita lihat pada gambar 4.20.



**Gambar 4.20 Tampilan *Form* Cetak**

* + 1. **Tampilan *Form User* Baru**

*Form* ini digunakan untuk melakukan penambahan user, ini berguna ketika petugas PTMdigantikan dengan petugas lain ketika ada pe-*Rolling*-an*,* Tampilan rancangan form *user* dapat kita lihat pada gambar 4.21 berikut :



**Gambar 4.21 Tampilan *Form User* Baru**

* + 1. **Tampilan *Form* Ubah *User***

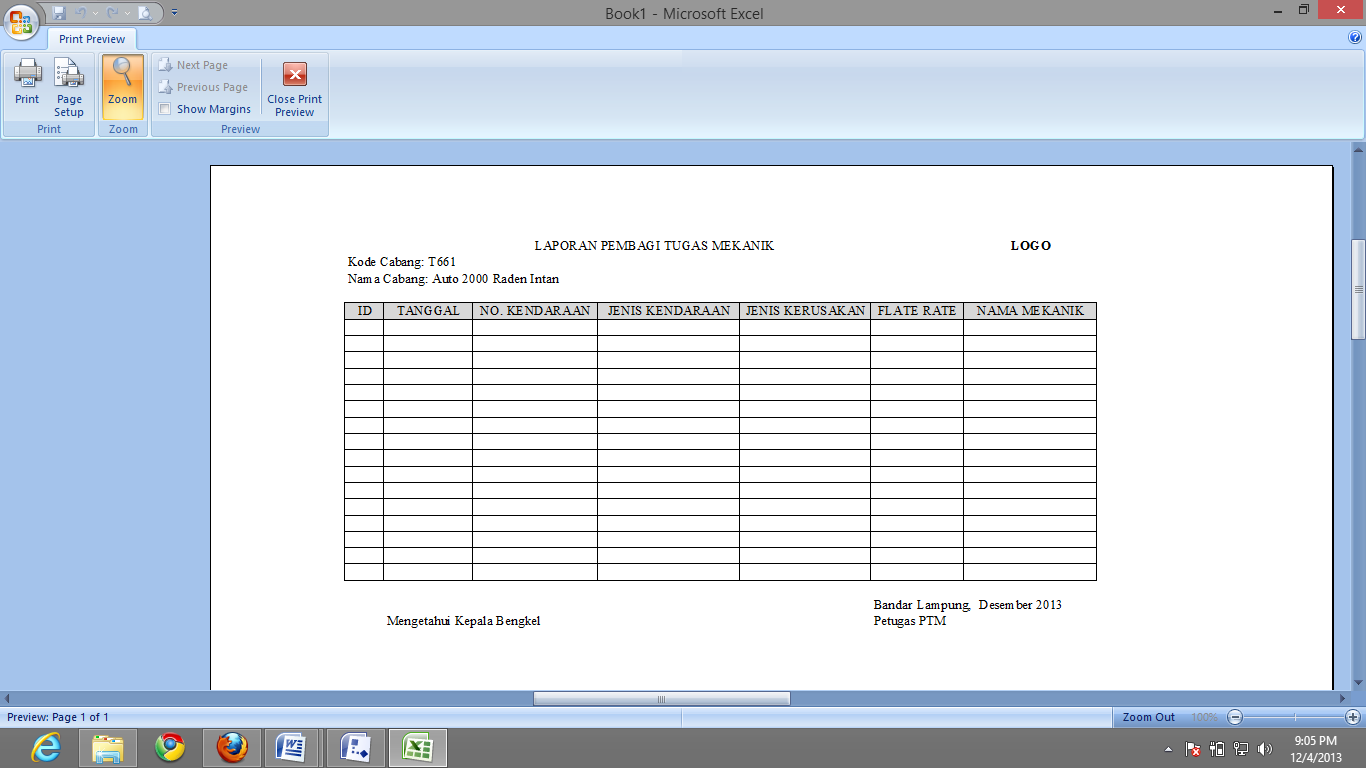
*Form* ini digunakan untuk melakukan perubahan pada *user* maupun sandinya, ketika *user* sudah diketahui oleh umum, maka perubahan *user* dapat dilakukan secara berkala, Tampilan rancangan *form* ubah *user* dapat kita lihat pada gambar 4.22 berikut:



**Gambar 4.22 Tampilan *Form* Ubah *User***

* + 1. **Tampilan Cetak Laporan Pembagi Tugas Mekanik**

Laporan pembagi tugas mekanikdigunakan sebagai laporan kepada HRD. Tampilan laporan pembagi tugas mekanikdapat kita lihat pada gambar 4.23:



**Gambar 4.7 Tampilan Laporan Pembagi Tugas Mekanik**

* 1. **Penjadwalan**

Jadwal penelitian yang meliputi persiapan pelaksanaan dan penyusunan laporan penelitian disajikan dalam bentuk Gann-Chhart pada table 4.7:

**Tabel 4.7 Penjadwalan**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| No | Kegiatan | Agustus | | | | September | | | | Oktober | | | | November | | | | Ket |
| 1 | 2 | 3 | 4 | 1 | 2 | 3 | 4 | 1 | 2 | 3 | 4 | 1 | 2 | 3 | 4 |
| 1 | Perencanaan |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | a. Observasi(Pengamatan) |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | b. Wawancara |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | c. Dokumentasi |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | d. Pengumpulan Data |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 2 | Analisis |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | Membuat Bagan Alir Dokumen |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 3 | Desain |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | a. Membuat UML |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | b. Membuat Rancangan Form |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

Penjelasan penjadwalan yaitu :

1. Perencanaan

Perencanaan dilakukan pada saat melakukan kegiatan PKL yaitu terdiri dari pengumpulan data yang dibutuhkan, wawancara terhadap pihak yang terkait dan dokumentasi yaitu megumpulkan dokumen pendukung.

1. Analisis

Tahap ini membuat Bagan Alir Dokumen (BAD) yaitu membuat sistem yang berjalan dengan menggunakan *Flowchart* dan melakukan Analisis PIECES*.*

1. Desain

Tahap ini membuat UML dan rancangan *form* tampilan sebagai pengembangan sistem.