BAB 3
ANALISIS DAN PERANCANGAN SISTEM

3.1 Komunikasi

Dalam membangun suatu aplikasi, komunikasi adalah tahapan pertama penguraian dari apa yang didapat pada penelitian untuk mengidentifikasikan dan mengevaluasi permasalahan, kesempatan, hambatan yang terjadi dan kebutuhan yang diharapkan sehingga dapat tercapainya tujuan suatu sistem yang akan dibangun.

Perancangan dan implementasi sistem injeksi sepeda motor dengan berbasiskan IoT merupakan sistem yang akan dibangun pada penelitian ini, sistem perancangan dan implementasi yang akan dibuat ini meliputi pemberitahuan kondisi kendaraan bila ada malfunngsi dengan cara mengirimkan pesan kepada pengguna, lalu pengecekan dan monitoring sistem injeksi sepeda motor yang dihubungkan dengan internet sehingga pengguna dapat melihat kondisi melalui browser.

3.1.1 Analisis Masalah

Analisa masalah dengan kasus yang diteliti yaitu bagaimana membuat ECU (*engine control unit*) dapat menyimpan semua data yang dihasilkan dari sensor serta kegagalan system injeksi ke webserver agar pemilik sepeda motor tersebut dapat memantau keadaan kendaraannya sendiri kapanpun dan dimanapun selama terkoneksi dengan internet. Serta untuk mempermudah teknisi juga dalam memperbaiki kendaraan yang akan di perbaikinya, karena berdasarkan hasil wawancara yang dilakukan dengan Bapak Eli Mulyana selaku *service advisor* bengkel yamaha star motor, beliau sering menjumpai konsumen yang mengeluh namun tidak dapat menjelaskan keluhannya (kronologi) tersebut secara jelas kepada teknisi. Dan ini menjadi kesulitan bagi para teknisi pada saat pengecekan kendaraannya melalui alat diagnosa yang tersedia di bengkel karena ECU pabrikan tidak mempunyai data loging perihal bilamana terjadi kegagalan system di bagian injeksi elektronik nya, ECU pabrikan hanya menampilkan kode

kesalahan dan kedipan lampu saja pada speedometer kendaraan, sehingga pada saat di periksa menggunakan alat diagnosa yang terdapat di bengkel resmi hanya tersimpan kode dan bagian kerusakan nya saja tanpa mengetahui bagaimana kondisi pertamakali masalah itu timbul.

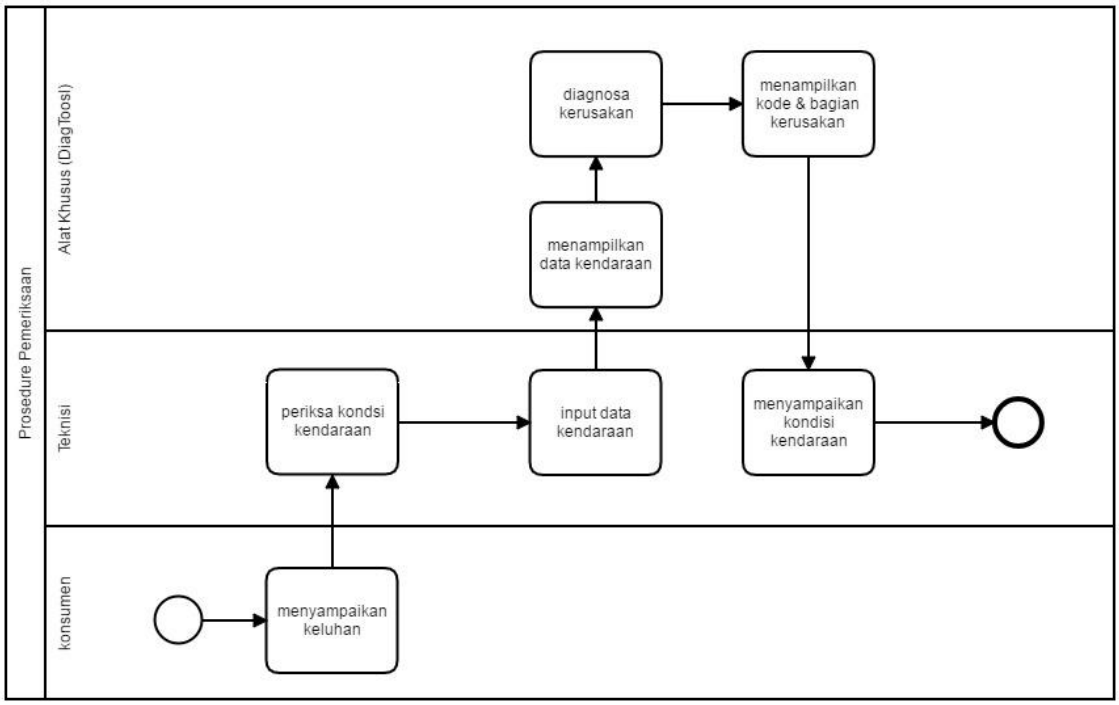
3.1.2 Analisis sistem yang sedang berjalan

Analisa ini bertujuan untuk mempelajari serta mengevaluasi sistem pemeriksaan kendaraan yang sedang berjalan dan mengetahui seperti apa sistem pemeriksaan kendaraan yang sedang berjalan saat ini di bengkel yamaha star motor.

3.1.2.1. Prosedur pemeriksaan kendaraan

Berikut dibawah ini adalah standar operasional prosedur pemeriksaan kendaraan yang sedang berjalan di bengkel yamaha star motor.

1) Konsumen datang membawa kendaraan lalu menyampaikan keluhannya terhadap teknisi.
2) Teknisi memeriksa kondisi kendaraan konsumen lalu memasukan data kendaraan tersebut untuk di diagnosa menggunakan alat diagtools yamaha yang sebelumnya telah di koneksikan terlebih dahulu.
3) Diagtools menerima inputan dari teknisi dan menampilkan data kendaraan, selanjutnya diagtools memindai ECU kendaraan dan menampilkannya pada layar monitor beserta diagnosa kerusakannya.
4) Teknisi menyampaikan kondisi kendaraan yang telah di diagnosa terhadap konsumen.



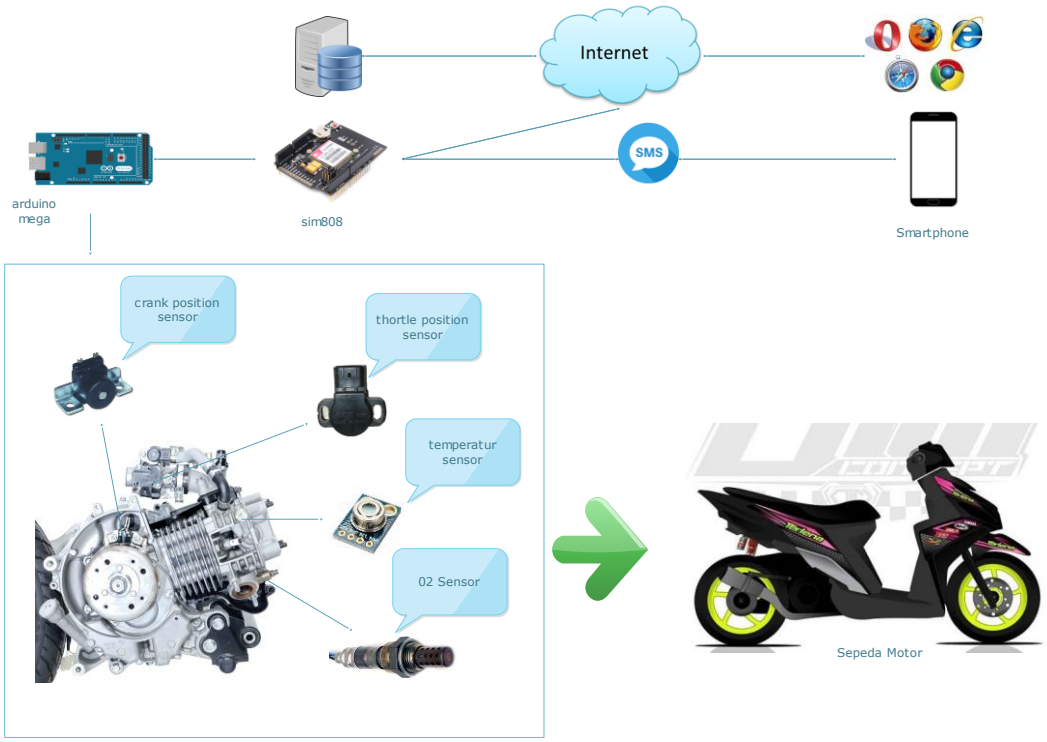
Gambar 3 1 Prosedur Pemeriksaan Kendaraan

3.1 Perencanaan cepat

Pada tahapan ini dilakukan perancangan sistem yang akan dibangun dalam bentuk *ptorotype*, dan disesuaikan dengan sistem yang telah didefinisikan sebelumnya dari permasalahan yang dihadapi oleh teknisi maupun pengguna sepeda motor.

3.2.1 Analisis arsitektur sistem

Analisi sarsitektur sistem merupakan sebuah proses untuk mendeskripsikan fisik sistem yang akan dibangun dan juga berbagai komponen pendukungnya. Berikut dibawah ini adalah gambaran arsitektur sistem yang akan dibangun



Gambar 3 2 Arsitektur Sistem

Berikut dibawah ini adalah penjelasan mengenai arsitektur sistem ECU berbasiskan IoT yang memiliki 2 buah alur kerja pengiriman data pada gambar diatas ini :

Proses untuk memonitoring ECU melalui browser :

a) Sensor yang terdapat pada mesin sepeda motor yang diantaranya adalah sensor *thortle position*, *temperature sensor*, *crankshaft position*, dan O2 sensor akan memberikan data input mengenai kondisi mesin kepada mikrokontroller (arduino) untuk diolah datanya agar sepeda motor dapat berjalan dengan baik.

b) Mikrokontroller (arduino) di hubungkan dengan modul SIM808 yang telah terhubung dengan internet untuk mengirim data input sensor ke *webservice* yang berada dalam *webserver* .
c) *Webservice* menyimpan data yang dikirim oleh modul SIM808 ke dalam *database.*
d) Pengguna mengakses browser dan memasukan alamat *website* pemonitoringan, *webservice* mengambil data dari database lalu menampilkan data tersebut pada *browser* pengguna.

Proses pemberitahuan masalah melalui SMS.

a) Sensor tidak dapat mengrimkan data kepada mikrokontroller (arduino).
b) Mikrokontroller (arduino) membaca sensor yang tidak mengirimkan data itu sebagai malfungsi dan mengrimkan data kode kerusakan ke modul SIM808.
c) SIM808 yang telah di inputkan nomor pengguna mengirimkan data kerusakan berupa SMS kepada *smartphone* pengguna sesuai dengan yang diperintahkan mikrokontroller (arduino).

3.2.2 Analisis komunikasi data

Komunikasi data merupakan suatu hal yang sangat penting, karena tanpa adanya komunikasi data, komunikasi yang dibangun tidak akan berjalan dengan baik, komunikasi data yang digunakan pada perancangan dan implementasi ECU ini menggunakan SIM808 dengan mikrokontroller arduino mega dengan perintah AT+*command* sehingga modul dapat mengirimkan data berupa SMS dan data-data lain seperti sensor ke database melalui *protocol* HTTP(*Hyper Text Transfer Protocol*) dan metode yang digunakan dalam pengiriman paket data tersebut adalah menggunakan metode GET dan POST.

3.2.3 Analisis kebutuhan non fungsional

Analisis kebutuhan non fungsional merupakan analisis yang dibutuhkan untuk menentukan spesifikasi kebutuhan sistem, sepesifikasi ini juga merupakan meliputi elemen atau komponen-komponen yang diperlukan untuk sistem yang akan dubangun dan di implementasikan. Analisis kebutuhan ini juga menentukan spesifikasi masukan yang diperlukan sistem, keluaran yang dihasilkan oleh sistem dan proses yang dibutuhkan untuk mengolah masukan sehingga diperoleh keluaran yang di inginkan. Berikut dibawah ini adalah spesifikasi kebutuhan perangkat lunak non fungsional dalam penelitian ini.

Tabel 3 1 Spesifikasi Kebutuhan Non Fungsional

Nomor	Keterangan Non Fungsional
SKPL-NF-001	Sistem mampu diakses melalui jaringan internet
SKPL-NF-002	Sistem monitoring dapat dijalankan di tipe perangkat apapun yang memiliki *browser*
SKPL-NF-003	Sistem mampu mengirimkan data melalui jaringan GPRS dan GSM
SKPL-NF-004	Sistem mampu mengirimkan pesan singkat (SMS)
SKPL-NF-005	Sistem dapat membaca semua sensor yang ada pada mesin sepeda motor
SKPL-NF-006	Sistem dapat mengatur dan menentukan semprotan bahan bakar dan pengapian pada mesin sepeda motor.

3.2.3.1. Analisa kebutuhan pengguna

Analisa kebutuhan pengguna dimaksudkan untuk mengetahui siapa pengguna yang terlibat menjalankan sistem, hal ini untuk mengetahui siapa pengguna yang terlibat dalam menjalankan sistem yang akan dibangun.

Berdasarkan hasil pengmatan, pengguna yang terlibat dalam sistem yang sedang berjalan adalah sebagai berikut :

1) Pengguna aplikasi diagtools yang sedang berjalan ini adalah teknisi.

Untuk menggunakan sistem yang akan dibangun terdapat beberapa karakteristik yang harus dimiliki oleh pengguna yaitu dapat dilihat pada tabel di bawah ini

Tabel 3 2 Karteristik Pengguna

Pengguna	Hak Akses	Tingkat keterampilan
Teknisi	• Monitoring kendaraan • Mengubah data pelanggan • Mengelola data kendaraan	• Dapat menggoperasikan komputer • Biasa menggunalan internet
Pelanggan	• Monitoring kendaraannya	• Memahami dan tebiasa dengan internet

3.2.3.2. Analisis kebutuhan perangkat keras

Perangkat keras atau *hardware* merupakan salah satu yang sangat penting dalam pembuatan dan pembangunan sistem, sistem terbeut tidak akan dapat berjalan dengan baik apabila tidak di dukung dengan komponen *hardware* yang baik dan memadai.

Kebutuhan perangkat keras yang dibutuhkan meliputi kebutuhan perangkat keras yang digunakan pada sepeda motor dan kebutuhan perangkat keras minimum pada komputer/laptop serta *smartphone* yang digunakan oleh pengguna yang nantinya akan digunakan sebagai pemonitoring data kondisi kendaraan dan pemberitahuan bila ada kesalahan kpada sistem sensor injeksi sepeda motor.

Berikut dibawah ini adalah spesifikasi kebutuhan perngkat yang digunakan dalam penelitian ini.

Tabel 3 3 Spesifikasi Sistem ECU berbasis IoT

Hardware	Detail
Arduino Mega 2560	ATmega 2560
	SIM808 shield

Modul GSM	GPRS
	GPS
	GSM
Sensor	Thorlte Position (Original Yamaha)
	Temperatur (ML90614)
	Crank Position (Original Yamaha)
	O2 Sensor (Original Yamaha)
	Tegangan (voltage sensor/ACS712)
DC to DC Converter	LM2956

Berikut adalah sepesifikasi minimal perangkat *smarthpone* dan komputer untuk dapat memantau sistem ECU berbasis IoT ini.

Tabel 3 4 spesifikasi minimal komputer

Jenis	Spesifikasi
Browser	Mozila, Internet explore, Chrome
RAM	1 GB
Hardisk	250 GB
System Operasi	Windows , Linux, MAC.

Tabel 3 5 Spesifikasi minimal *Smartphone*

Jenis	Spesifikasi
Browser	Mozila, Internet explore, Chrome
RAM	512 MB
Memory internal	500 MB
System Operasi	Android, IOS, Windows Phone.

Berdasarkan pada spesifikasi yang ada diatas maka untuk sistem yang akan dibangun sudah memenuhi kriteria yang diperlukan untuk mendukung pengopreasian sistem.

3.2.3.3. Analisis kebutuhan perangkat lunak

Perangkat lunak (*software*) merupakan sebuah komponen penting pendamping perangkat keras (*hardware*). Secara umum *software* merupakan sekumpulan data-data elektronik yang tersimpan dan diatur oleh komputer yang berupa program atau instruksi untuk menjalankan dan mengeksekusi suatu perintah. Perangkat lunak berfungsi sebagai sarana interaksi antara pengguna dan perangkat keras. Berikut tabel dibawa ini adalah spesifikasi perangkat lunak untuk membangun sistem ECU berbasis IoT ini

Tabel 3 6 Spesifikasi pembangunan perangkat lunak aplikasi

o	Perangkat Lunak	Spesifikasi
	Sistem Operasi	Microsoft windows 7 64 bit
	Tools Utama	Arduino IDE, Notepad ++
	Tools Pendukung	Xampp

3.2 Pemodelan Perencanaan cepat

Pada tahap ini dilakukan pemodelan *protoype* sistem yang disesuaikan dengan perancangan sistem yang akan dibuat .

3.3.1 Analisis kebutuhan fungsional

Analisis kebutuhan fungsional menggambarkan proses kegiatan yang akan diterapkan dalam sistem dan menjelaskan kebutuhan yang diperlukan agar sistem dapat belajar dengan baik serta sesuai dengan kebutuhan sistem. Analisis yang dilakukan dimodelkan dengan menggunakan UML (*Unified Modeling Language*).

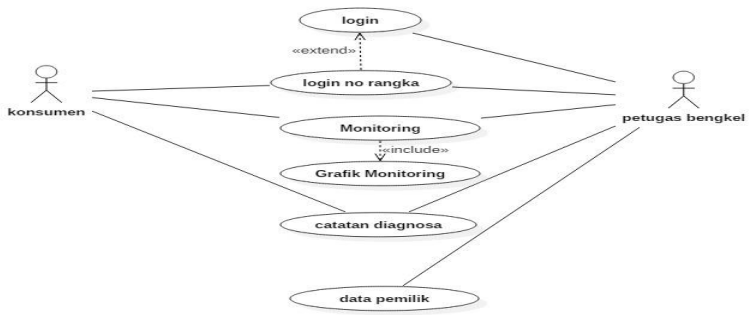
Tahapan pemodelan dalam analisis tersebut antara lain mengidentifikasi aktor, pembuatan *Use Case Diagram*, *Use Case Scenario*, *Activity Diagram*, *Sequence Diagram*, dan *Class Diagram*.

Tabel 3 7 Spesifikasi kebutuhan fungsional

Nomor	Keterangan Fungsional
SKPL-F-001	Sistem menyediakan fasilitas login untuk pengguna
SKPL-F-002	Sistem menyediakan fasilitas login untuk petugas bengkel (teknisi)
SKPL-F-003	Sistem dapat memonitoring data kendaraan
SKPL-F-004	Sistem dapat memonitoring dengan menampilkan grafik monitoring
SKPL-F-005	Sistem dapat menampilkan catatn diagonsa kendaraan
SKPL-F-006	Sistem dapat mengubah data pemilik kendaraan

3.3.1.1. Use case diagram

Use case atau *use case diagram* adalah merupakan pemodelan untuk kelakuan (*behaviour*) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antar satu aktor atau lebih dengan sistem informasi yang akan dibangun, use case digunakan untuk mengetahui fungsi apa saja yang ada didalam sistem informasi dan siapa saja yang berhak menggunakna fungsi itu.



Gambar 3 3 *Use Case* Diagram Sistem Informasi ECU Berbasis IoT

1. **Definisi aktor**

Definisi aktor berfungsi untuk menjelaskan aktor yang terdapat pada *Use Case Diagram* . berikut adalah definisi aktor yang ada pada sistem ECU berbasis IoT ini.

Tabel 3 8. Definisi Aktor

o	Aktor	Dekripsi
	Konsumen	Pengguna dapat memantau kondisi kendaraannya melalui website yang diakses melalui browser lalu memasukan nomor frame agar bisa masuk ke halaman utama pemantauan.
	Petugas Bengkel (Teknisi)	Petugas bengkel dapat melihat kondisi sepeda motor konsumennya dengan mengakses website melalui browser lalu memasukan kode bengkel dan password untuk dapat mauk ke halaman utama pemantauan khusus untuk petugas bengkel (teknisi).

2. **Definsi *Uses Case Diagram***

Definisi *Use case diagram* digunakan untuk menjelaskan fungsi dari *use case* yang terdapat pada *use case diagram* . untuk definsi diagram nya dapat dilihat pada tabel di bawah ini.

Tabel 3 9 Definisi *Use Case Diagram*

o	*Use Case*	Deskripsi
	Login No Rangka	konsumen melakukan login untuk masuk ke halaman utama aplikasi
	Login	Petugas bengkel (teknisi) memasukan kode bengkel dan password untuk dapat masuk pada menu utama aplikasi.

	Monitoring	pengguna dapat memantau kondisi sistem injeksi kendaraannya.
	Grafik Monitoring	pengguna dapat memantau kondisi sistem injeksi kendaraannya dengan tampilan grafik monitoring
	Catatan diagnosa	Pengguna dapat melihat catatan diagnosa dan kegagalan sistem yang pernah terjadi pada sistem injeksi kendaraan
	Data pemilik	Petugas bengkel (teknisi) dapat merubah data pemegang/pemilik kendaraan sepeda motor.

3. Use case scenario

Use scase scenario bertujuan untuk mendeskripsikan urutan langkah-langkah dalam proses yang dilakukan aktor terhadap sistem maupun yang dilakukan sistem terhadap aktor.

1) *Use scenario login* no rangka.

Tabel 3 10 *Use case scenario login* no rangka

Scenario Login no rangka	
Nama *Use Case*	Login no rangka
Nama Aktor	Konsumen, petugas bengkel
Deskripsi	Menampilkan halaman login dengan memasukan no rangka sepeda motor
Kondisi awal	Pengguna di hadapkan pada tampilan login bertuliskan masukan no rangka .
Aksi Aktor	**Respon Sistem**
***Scenario* Normal**	
1. Pengguna menuliskan alamat di brwoser.	
	2. Menampilkan halaman untuk login
3. Pengguna memasukan no rangka	
	4. Sistem memvalidasi no rangka, apakah sudah tervalidasi dan regsitrasi
	5. Sistem menemukan no rangka

	6. Sistem menampilkan halaman utama
	7. Tidak di temukan
	8. Pesan *error*
Kondisi Akhir	Pengguna masuk ke halaman utama aplikasi

2) Use case scenario login

Tabel 3 11 *Usecase scenario login*

Scenario Login	
Nama *Use Case*	Login
Nama Aktor	Petugas bengkel
Deskripsi	Menampilkan halaman login khusus untuk petugas bengkel
Kondisi awal	Petugas berada pada tampilan login yang bertuliskan masukan kode bengkel dan password
Aksi Aktor	**Respon Sistem**
***Scenario* Normal**	
1. Pengguna menuliskan alamat di brwoser.	
	2. Menampilkan halaman login
3. Petugas memasukan kode bengkel dan password	
	4. Sistem memvalidasi kode bengkel dan password
	5. Validasi
	6. Menampilkan halaman utama untuk petugas bengkel
	7. Tidak valid
	8. Pesan *error*
Kondisi Akhir	Petugas bengkel masuk pada halaman utama aplikasi

3) *Use case scenario* monitoring

Tabel 3 12 Tabel u*se case scenario* monitoring

Scenario Monitoring	
Nama *Use Case*	Monitoring
Nama Aktor	Konsumen, petugas bengkel

Deskripsi	Menampilkan informasi kondisi sepeda motor.
Kondisi awal	pengguna disajikan pilihan pada menu utama.
Aksi Aktor	**Respon Sistem**
***Scenario* Normal**	
1. Pengguna menekan tombol monitoring pada menu utama	
	2. Sistem mengambil data terkahir yang di kirim oleh SIM808 pada database
	3. Sistem menampilkan informasi kondisi sepeda motor
Kondisi Akhir	Menampilkan informasi kondisi sepeda motor.

4) Use case grafik monitoring

Tabel 3 13 *scenario use case grafik monitoring*

Scenario Grafik Monitoring	
Nama *Use Case*	Grafik monitoring
Nama Aktor	Konsumen, petugas bengkel
Deskripsi	Menampilkan informasi kondisi sepeda motor dengan tampilan grafik monitoring
Kondisi awal	pengguna disajikan pilihan pada menu utama.
Aksi Aktor	**Respon Sistem**
***Scenario* Normal**	
1. Pengguna menekan tombol monitoring pada menu utama	
	2. Sistem mengambil data terkahir yang di kirim oleh SIM808 pada database
	3. Sistem menampilkan informasi kondisi sepeda motor
4. Menekan tombol grafik	
	5. Sistem menampilkan informasi kondisi sepeda motor dala bentuk grafik
Kondisi Akhir	Menampilkan informasi kondisi sepeda motor dalam bentuk grafik.

5) *Use case* catatan diagnosa

Tabel 3 *14 use case scenario* catatan diagnosa

Scenario catatan diagnosa	
Nama *Use Case*	Catatan diagnosa
Nama Aktor	Konsumen, petugas bengkel
Deskripsi	Menampilkan informasi diagnosa kerusakan apa saja yang pernah terjadi pada sistem injeksi sepeda motor.
Kondisi awal	
Aksi Aktor	**Respon Sistem**
***Scenario* Normal**	
1. Pengguna memilih tombol catatan diagnosa pada menu utama	
	2. Sistem mengambil data diagnosa dari database
	3. Sistem menampilkan daftar catatan diagnosa sepeda motor
Kondisi Akhir	Tampilan catatan diagnosa sepeda motor

6) *Use case* data pemilik

Tabel 3 15 *scenario use case* data pemilik

Scenario data pemilik	
Nama *Use Case*	Edit data pemilik
Nama Aktor	Petugas bengkel
Deskripsi	Untuk mengubah data pemilik sepeda motor
Kondisi awal	pengguna disajikan pilihan pada menu utama.
Aksi Aktor	**Respon Sistem**
***Scenario* Normal**	
1. Petugas memilih menu konsumen	
	2. Sistem menampilkan data konsumen
3. Petugas memilih tombol edit	
	4. Sistem menampilkan form edit data
5. Petugas memasukan data konsumen lalu menekan tombol	

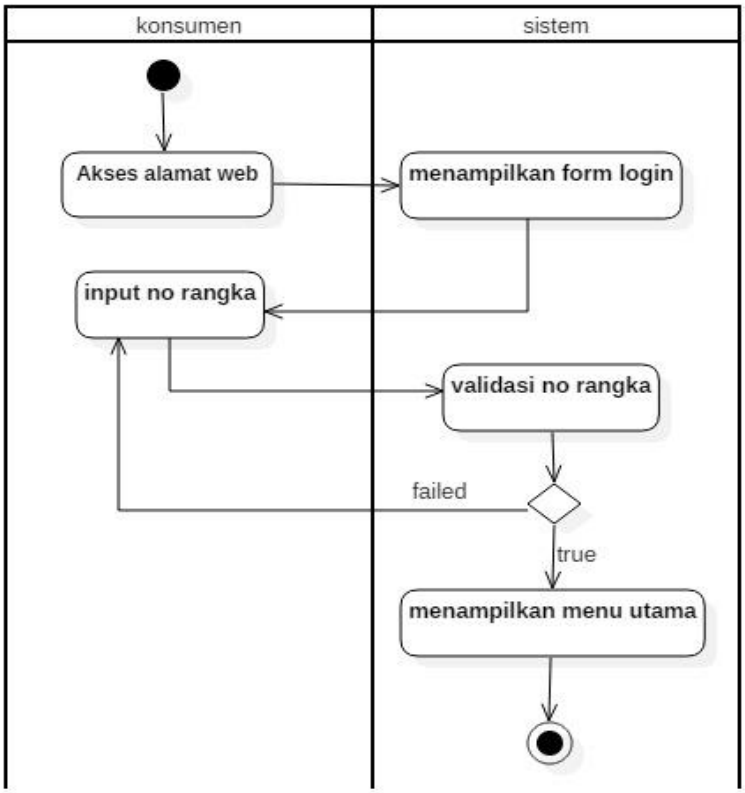
edit	
	6. Sistem menyimpan data konsumen pada database
Kondisi Akhir	Menampilkan data pemilik kendaraan yang sudah di *edit*

3.3.1.2. *Activity diagram*

Activity diagram menggambarkan proses pelayanan dan urutan aktivitas dalam sebuah proses, untuk memperlihatkan urutan aktivitas proses, *activity diagram* sangat bermanfaat untuk memahami proses dari sistem secara keseluruhan. *Activity diagram* dibuat berdasarkan sebuah atau beberapa *Use case* pada *Use case* diagram

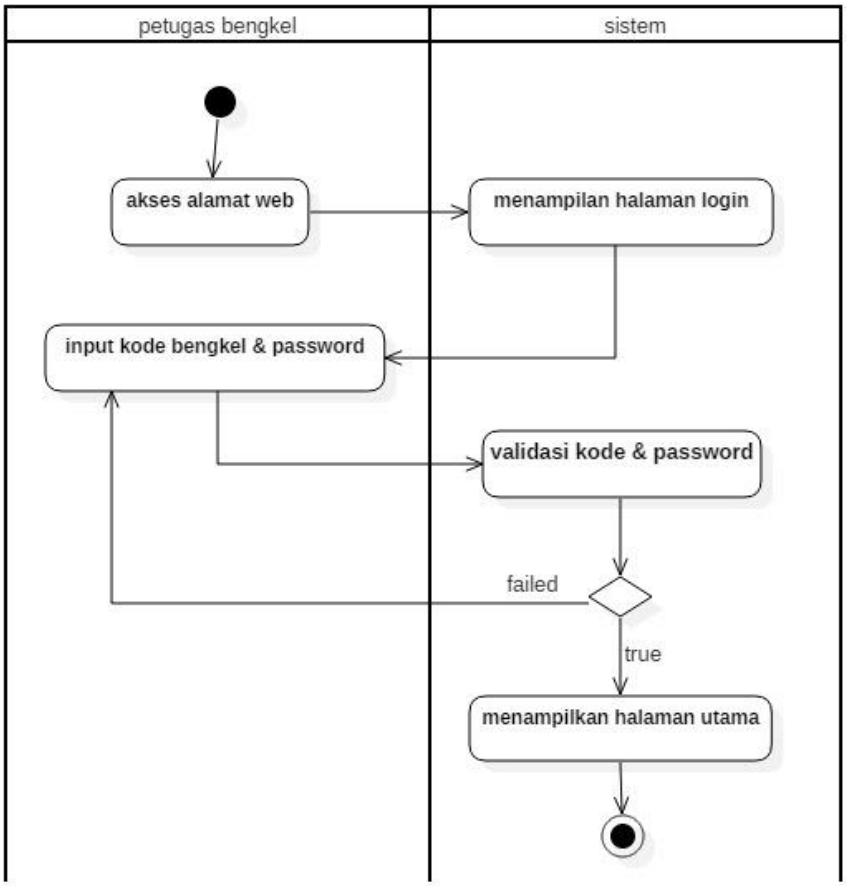
1. *Activity diagram login* konsumen

   Berikut adalah *activity diagram login* pada pengguna sistem dibawah ini :



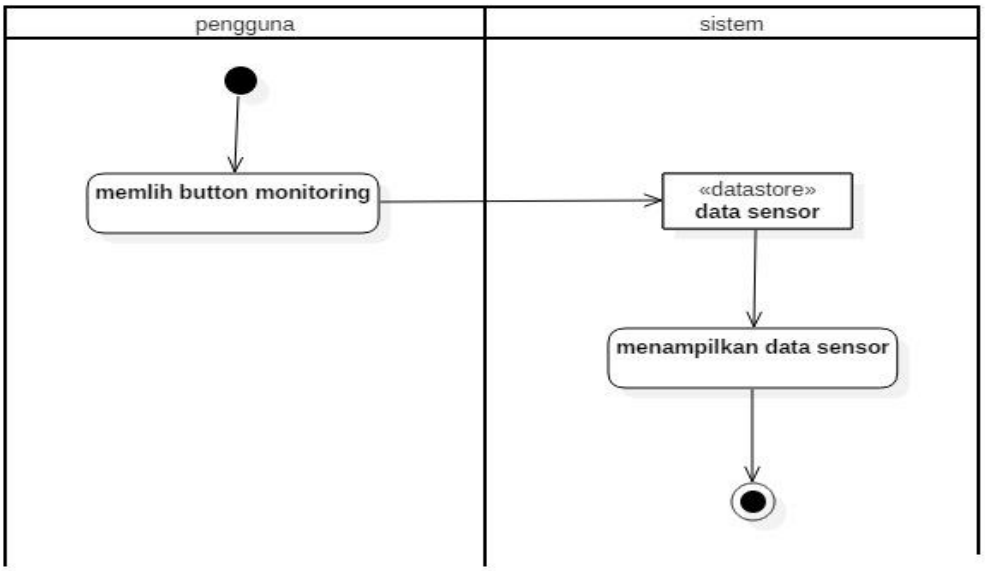
Gambar 3 4 *Activity Diagram login* pelanggan

2. *Activity diagram login* petugas bengkel



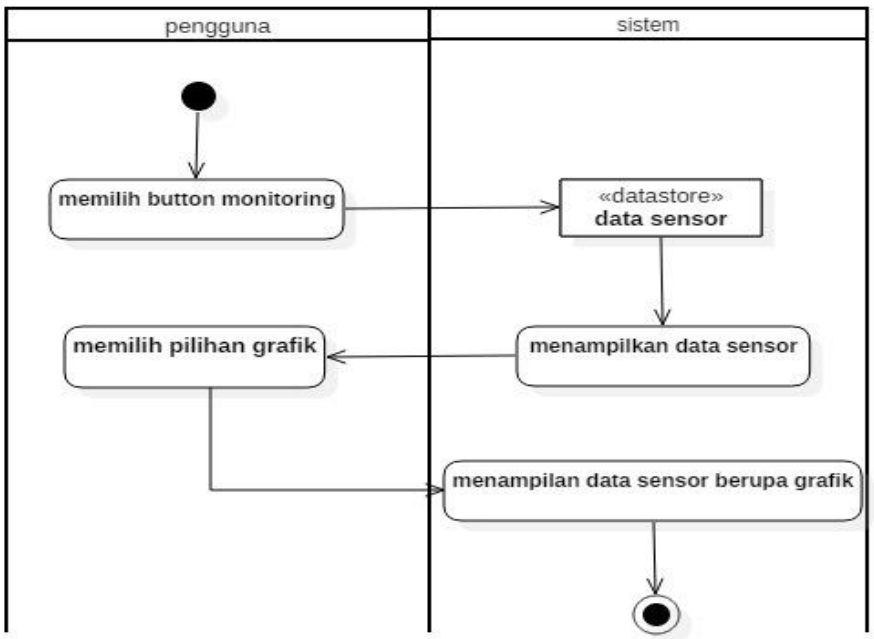
Gambar 3 5 *Activity diagram* login petugas bengkel

3. *Activity diagram* monitoring



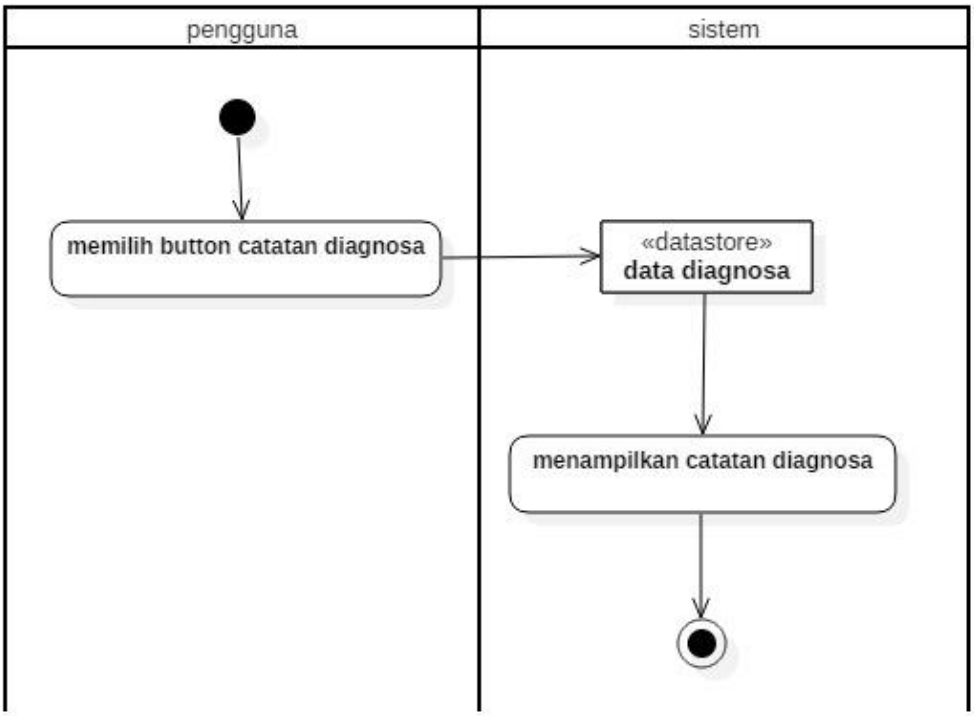
Gambar 3 6 *Activity diagram* monitoring

4. *Activity diagram* grafik monitoring



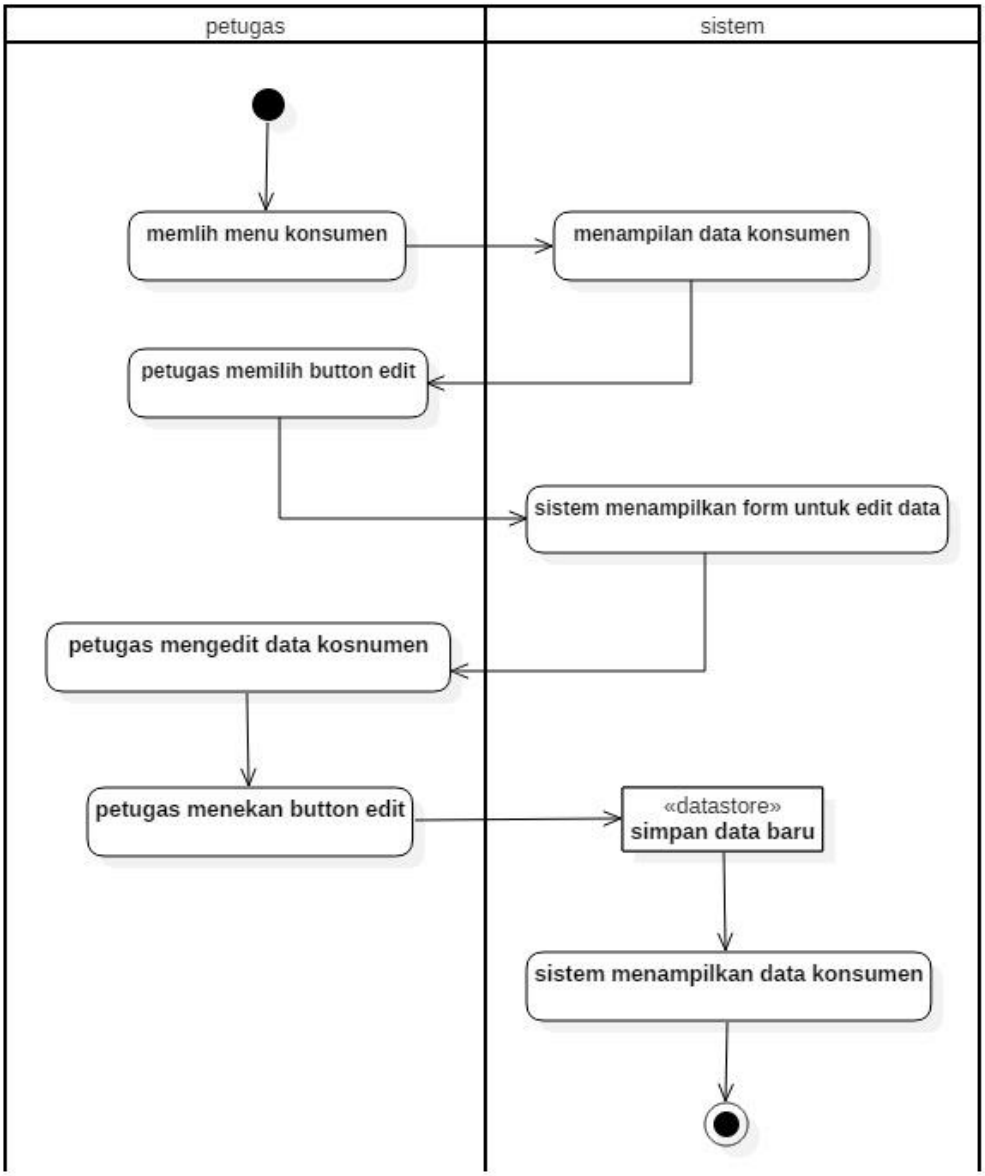
Gambar 3 7 A*ctivity diagram* grafik monitoring

5. *Activity diagram* catatan diagnosa



Gambar 3 8 Activity diagram catatan diagnosa

6. *Activity diagram* data pemilik

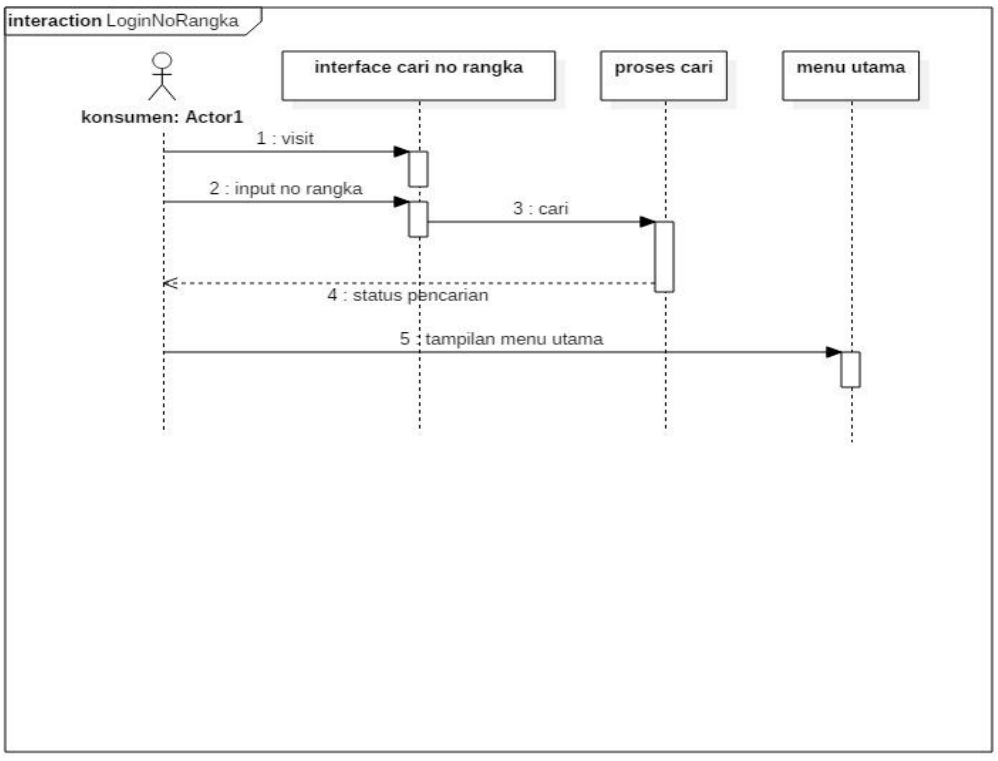


Gambar 3 9 *Activity diagram* edit konsumen

3.3.1.3. *Sequence diagram*

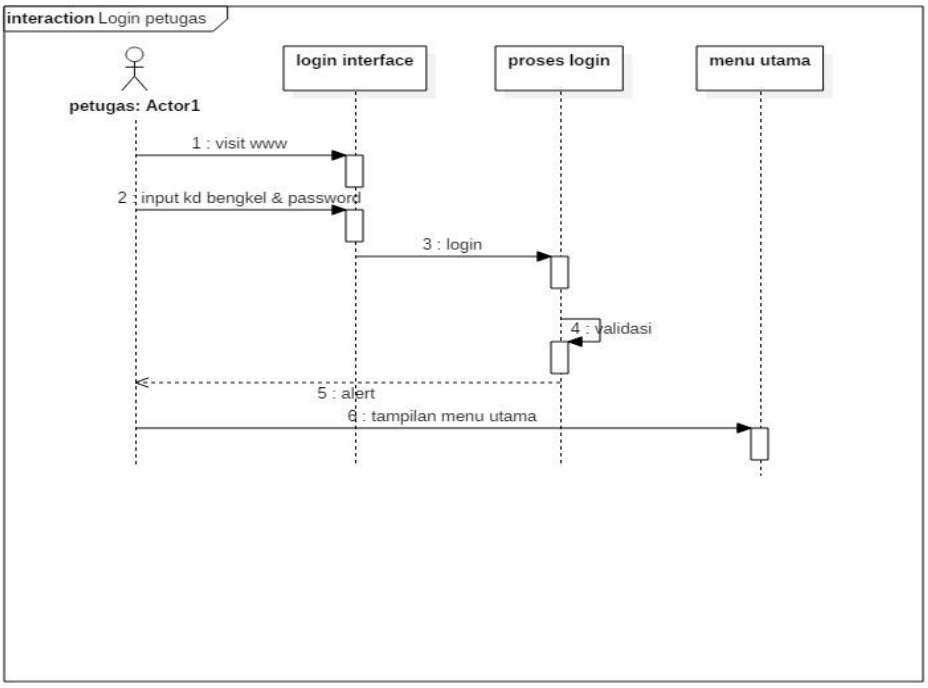
Merupakan bentuk dari *interaction diagram*, menggambarkan interaksi antar objek di dalam dan di sekitar sistem berupa *message* yang di gambarkan terhadap waktu. Masing–masing objek termasuk *actor* memiliki *line vertical message* digambarkan garis berpanah dari satu objek ke objek lainnya, dan *activation bar* menunjukan lamanya eksekusi sebuah proses. Sebuah *Sequence diagram*, secara khusus menjabarkan *behavior* sebuah skenario tunggal. Diagram tersebut menunjukkan sejumlah objek contoh dan pesan – pesan yang melewati objek–objek ini di dalam *Use case*. Berikut ini adalah *Sequence diagram* dari sistem yang akan dibangun.

1. *Sequence diagram* login no rangka



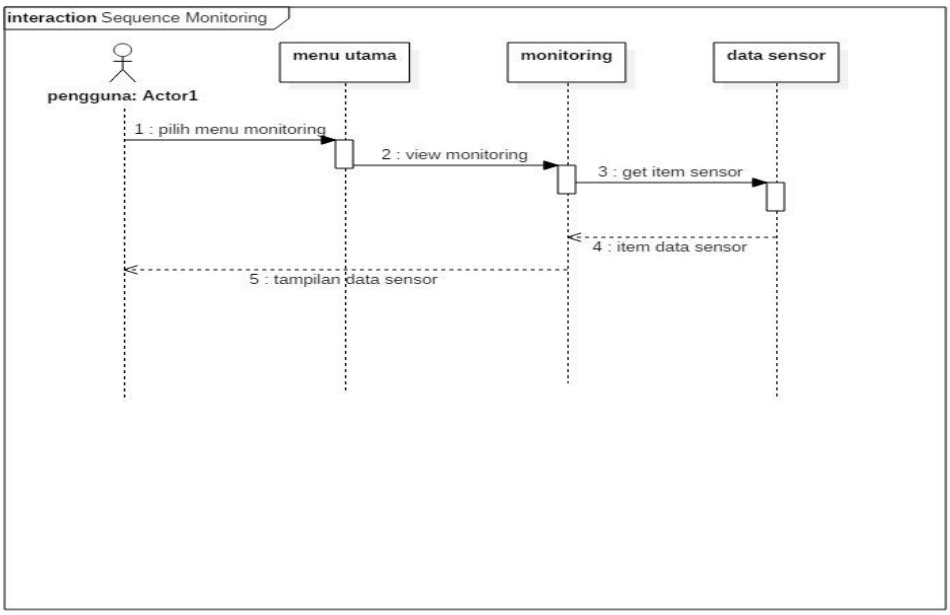
Gambar 3 10 *sequence diagram* login no rangka.

2. Sequence diagram Petugas bengkel



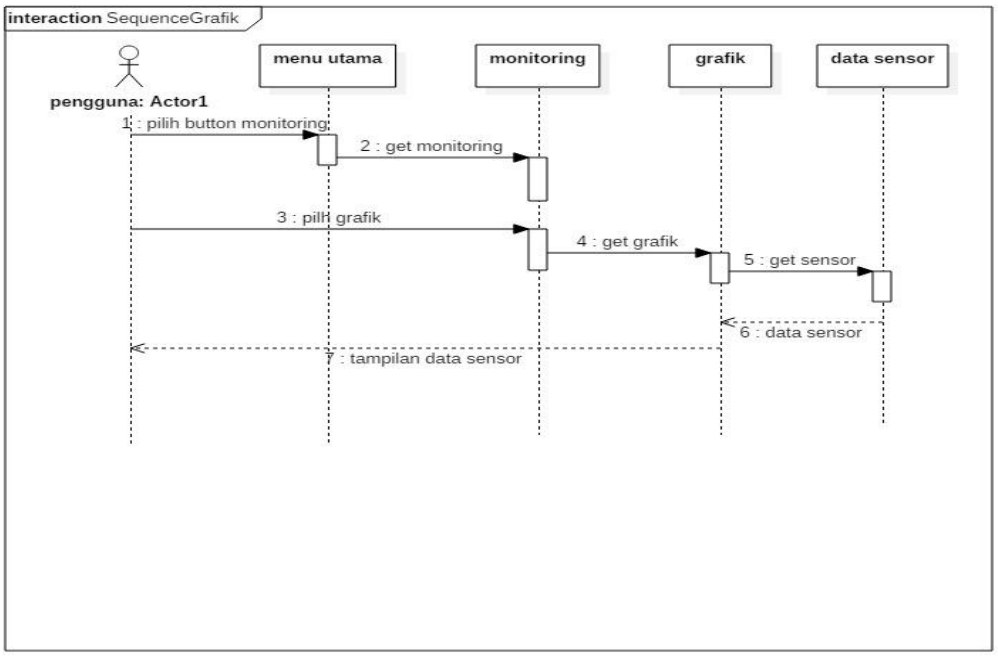
Gambar 3 11 *Sequence diagram* login petugas

3. Sequence diagram monitoring



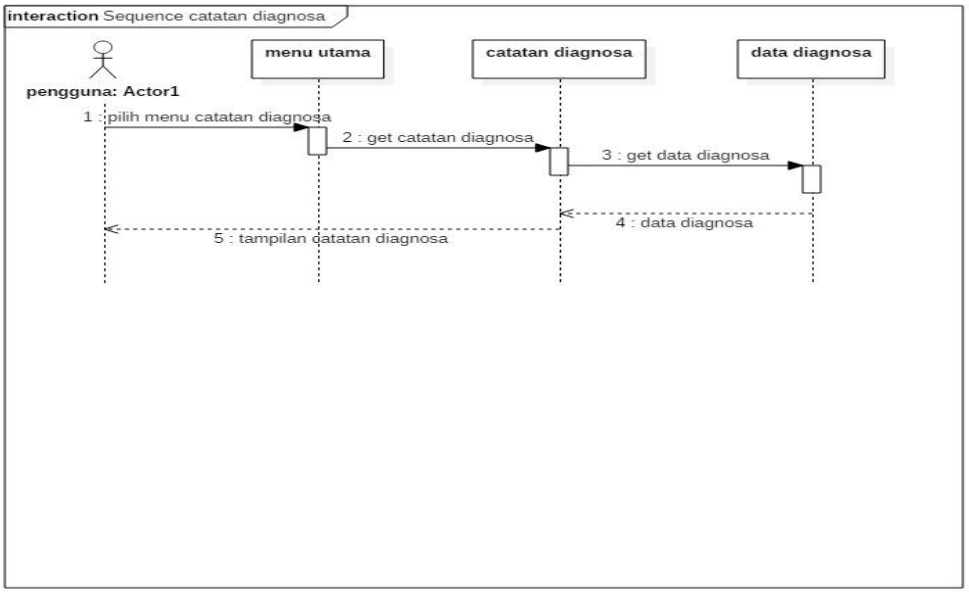
Gambar 3 12 *Sequence diagram* monitoring

4. Sequemce diagram grafik monitoring



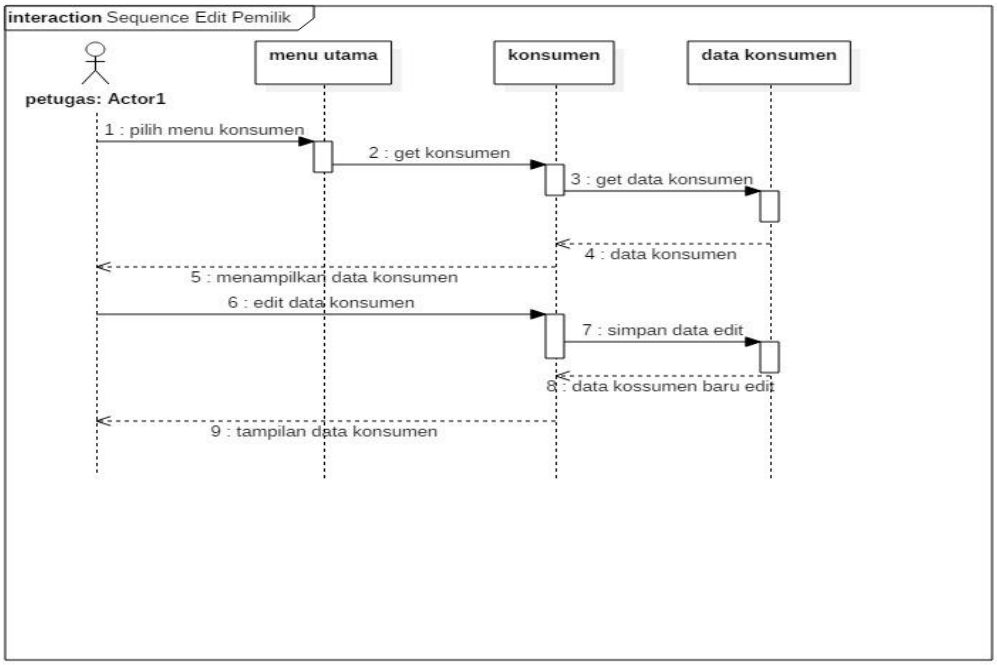
Gambar 3 13 *Sequence diagram* grafik monitoring

5. Sequence diagram catatan diagnosa



Gambar 3 14 *Sequence diagram* catatan diagnosa

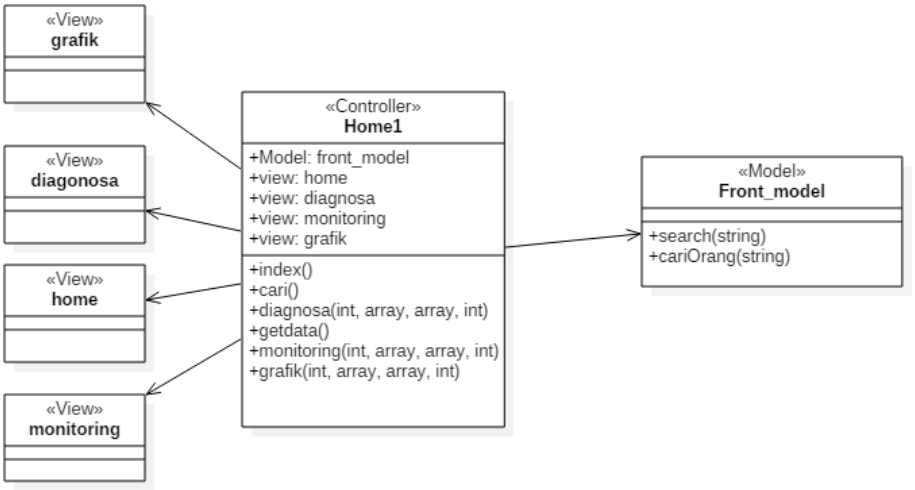
6. Sequence diagram data pemilik



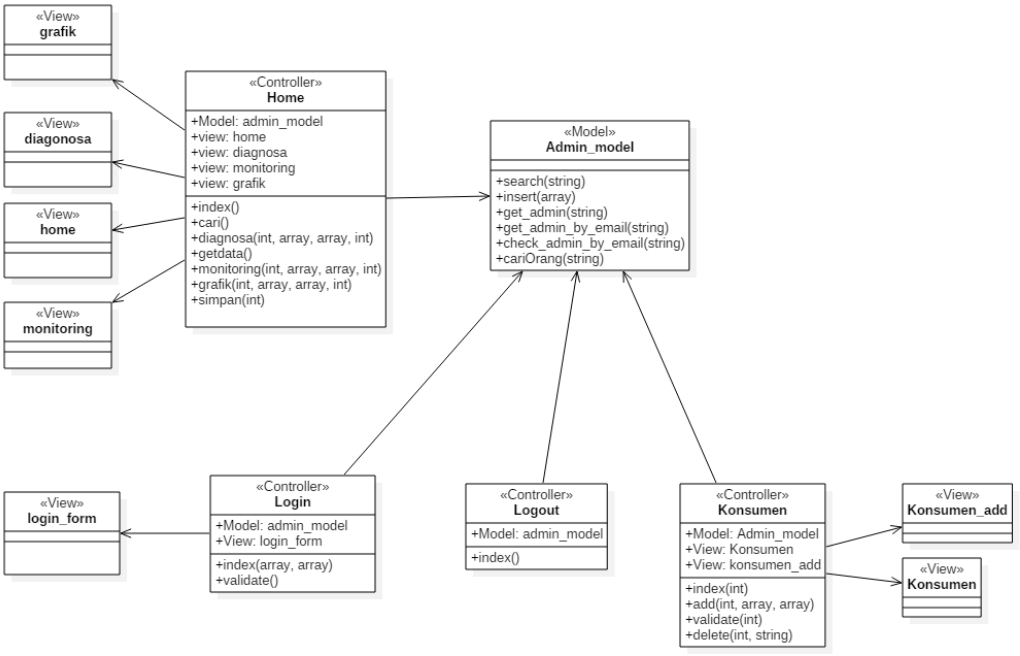
Gambar 3 15 *Sequence diagram* edit data pemilik

3.3.1.4. Class diagram

Berikut dibawah ini adalah gambar dari *class diagram* web untuk mengontrol keadaan suatu mesin sepeda motor.



Gambar 3 16 *Diagram class* bagian *front end*



Gambar 3 17 *Diagram class* Admin

3.3.2 Perancangan basis data

Perancangan basis data adalah tahapan untuk memetakan model konseptual ke model basis data yang akan dipakai. Perancangan basis data terbagi menjadi dua yaitu skema relasi dan perancangan struktur tabel.

3.3.2.1. Skema relasi

Skema relasi adalah rangkaian hubungan antara beberapa tabel pada sistem basis data. Penjelasan rangkaian basis data pada sistem ini dapat dilihat pada gambar di bawah ini .

3.3.2.2. Struktur tabel

Struktut tabel merupakan penjelasan dari tabel – tabel yang ada pada basis data. Struktur tabel dari perancangan dan implementasi ECU (engine control unit) berbasis IoT ini dapat dilihat sebagai berikut .

1) Tabel ini merupakan perancangan tabel User/konsumen pada sistem yang akan dibangun, berikut di bawah ini adalah penjelasan struktur tabel tb_user.

Tabel 3 16 tb_user

No	Field	Type	size	Kunci	Keterangan
1	no_rangka	varchar	150	Primary key	Not null
2	type	varchar	25		Not null
3	nm_pemilik	varchar	60		null
4	tlp	varchar	25		null

2) Tabel ini merupakan perancangan tabel tb_logdiag untuk loging dari sensor yang dikirim oleh modul GSM yang terdapat pada arduino, berikut di bawah ini adalah penjelasan struktur tabel tb_logdiag.

Tabel 3 17 tb_logdiag

No	Field	Type	Size	Kunci	Keterangan
1	id_diag	int	1	Primary key	not null
2	id_sensor	int	1		not null
3	waktu	date			not null
4	temperatur	int	1		not null
5	rpm	int	1		not null
6	tps	int	1		not null
7	volt_battere	int	1		not null
8	O2	int	1		not null
9	kd_kesalahan	int	1		not null

3) Tabel ini merupakan perancangan tabel kd_kesalahan pada sistem yang akan dibangun, berikut di bawah ini adalah penjelasan struktur tabel tb_kesalahan.

Tabel 3 18 tb_kesalahan

No	Field	Type	Size	Kunci	Keterangan
1	id	int		Primary key	Not null

2	Kd_kesalahan	varchar	25		Not null
3	gejala	varchar	500		Not null
4	penyebab	varchar	500		Not null

4) Tabel ini merupakan perancangan tabel tb_historidiagnosa pada sistem yang akan dibangun, berikut di bawah ini adalah penjelasan struktur tabel tb_historidiagnosa

Tabel 3 19 tb_historidiagnosa

No	Field	Type	Size	Kunci	Keterangan
1	id	int	11	Primary key	Not null
2	Id_konsumen	varchar	50		Not null
3	Id_diag	int	11		Not null
4	Id_sensor	int	11		Not null
5	waktu	date			Not null
6	temperatur	int	11		Not null
7	rpm	int	11		Not null
8	tps	int	11		Not null
9	Volt_battere	int	11		Not null
10	O2	int	11		Not null
11	Kd_kesalahan	varchar	25		Not null
12	diagnosa	varchat	500		Not null

5) Tabel ini merupakan perancangan tabel tb_bengkel pada sistem yang akan dibangun, berikut di bawah ini adalah penjelasan struktur tabel tb_bengkel

Tabel 3 20 tb_bengkel

No	Field	Type	Size	Kunci	Keterangan
1	id	int	11	Primry key	Not null
2	Kode_bengkel	varchar	25		Not null
3	password	varchar	50		Not null
4	nm_bengkel	varchar	25		Not null
5	almt_bengkel	varchar	25		Not null
6	tlp	varchar	25		Not null
7	image	varchar	100		Not null

6) Tabel ini merupakan perancangan tabel tb_admin_backend pada sistem yang akan dibangun, berikut di bawah ini adalah penjelasan struktur tabel tb_admin_backend.

Tabel 3 21 tb_admin_backend

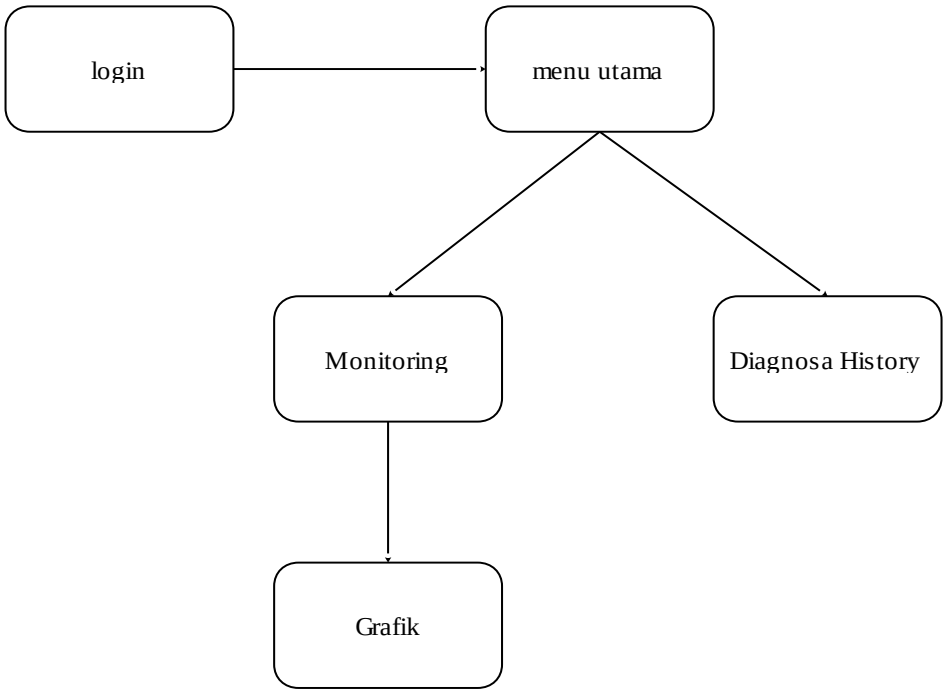
No	Field	Type	ize	Kunci	Keterangan
1	id	int	11	Primary key	Not null
2	Kode_bengkel	varchar	5		Not null
3	password	varchar	0		Not null
4	nm_bengkel	varchar	5		Not null
5	Almt_bengkel	varchar	5		Not null
6	tlp	varchar	5		Not null
7	image	varchar	00		Not null

3.3.3 Perancangan sistem

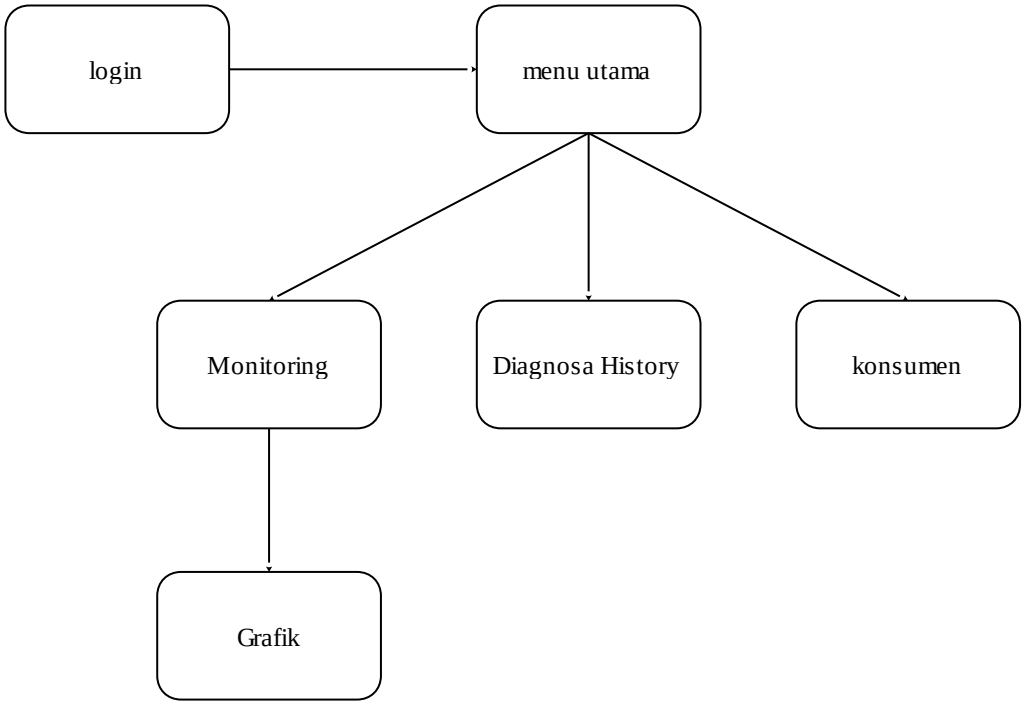
Perancangan sistem menentukan bagaimana suatu sistem akan menyelesaikan apa yang mesti diselesaikan. Tahap ini menyangkut mengkonfigurasikan dari komponen-komponen perangkat lunak dan perangkat keras dari suatu sistem sehingga sistem akan memuaskan rancang bangun yang telah ditetapkan pada tahap akhir analisa sistem. Perancangan sistem yang akan dibangun meliputi perancangan struktur menu, perancangan antarmuka, jaringan semantik, dan perancangan *prototype*.

3.3.3.1. Perancangan struktur menu

Proses perancangan struktur menu untuk sistem ini ada dua yaitu bagian konsumen dan petugas bengkel (admin) adalah sebagai berikut :



Gambar 3 18 Struktur menu bagian konsumen



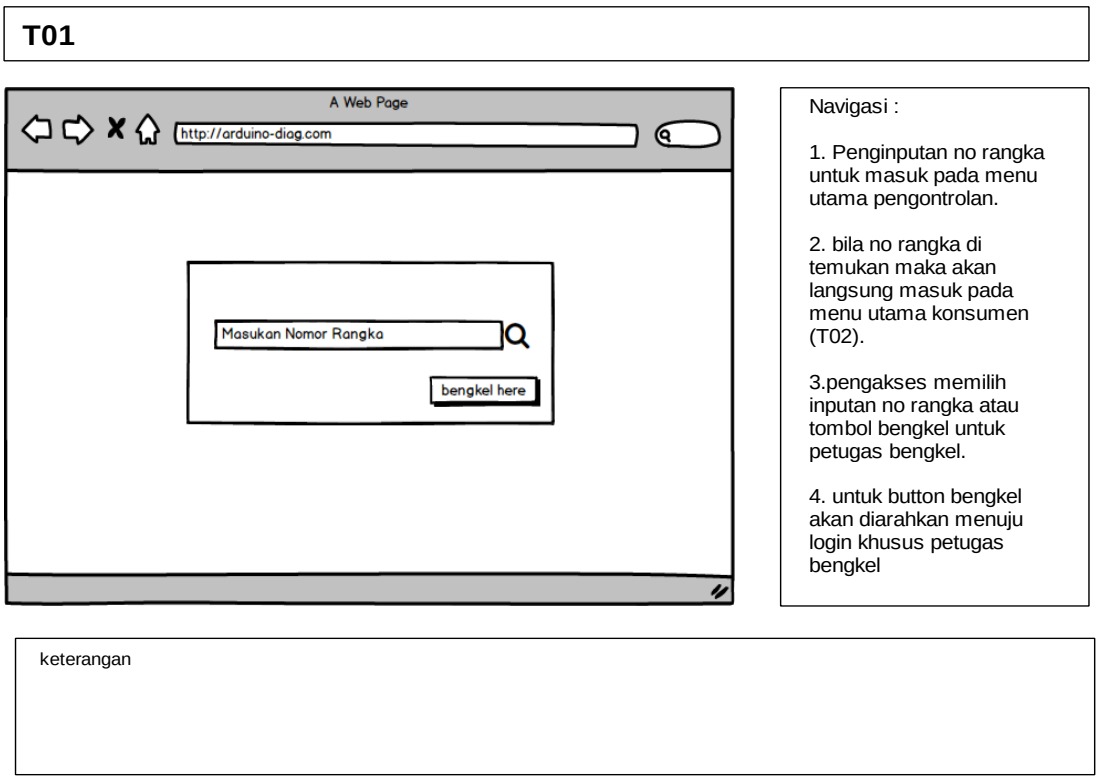
Gambar 3 19 Struktur menu bagian petugas bengkel

3.3.3.2. Perancangan antar muka

Perancangan antarmuka merupakan suatu deskripsi *layout* antarmuka sistem yang akan dibuat. Dengan adanya perancangan antar muka diharapkan baik pengembang aplikasi maupun pengguna dapat menyatukan pandangan mengenai *layout* tampilan aplikasi yang dibuat. Perancangan antarmuka dilakukan untuk menggambarkan lebih detil mengenai desain tampilan dan navigasi menu yang terdapat didalamnya. Berikut ini adalah perancangan antarmuka dan navigasinya.

1) Perancangan antar muka kosunmen
   a) Perancangan antar muka login

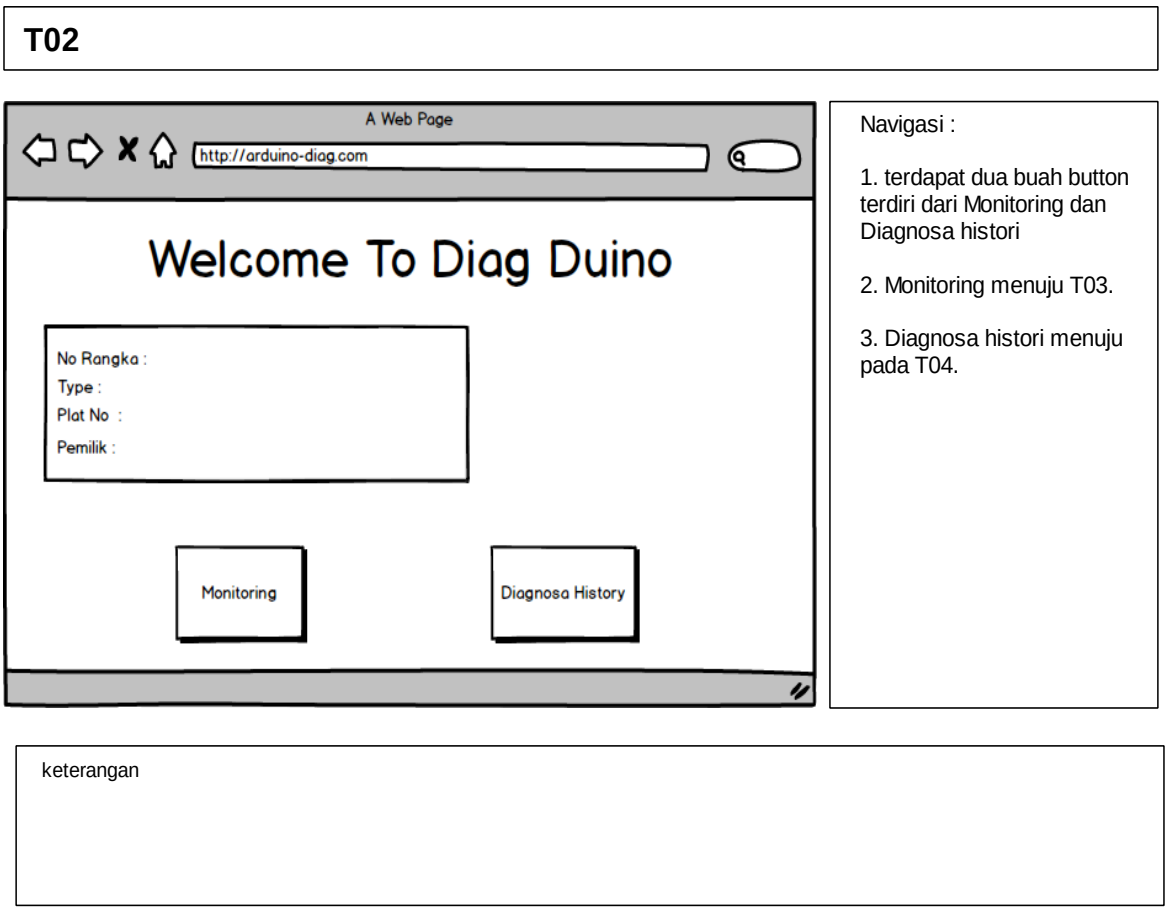
      Perancangan ini menampilkan halaman login pada aplikasi web yang merupakan awal pengguna untuk dapat masuk pada menu utama lalu mengontrol kondisi keadaan sebuah sepeda motor.



Gambar 3 20 Antarmuka login konsumen

b) Perancangan antarmuka menu utama.

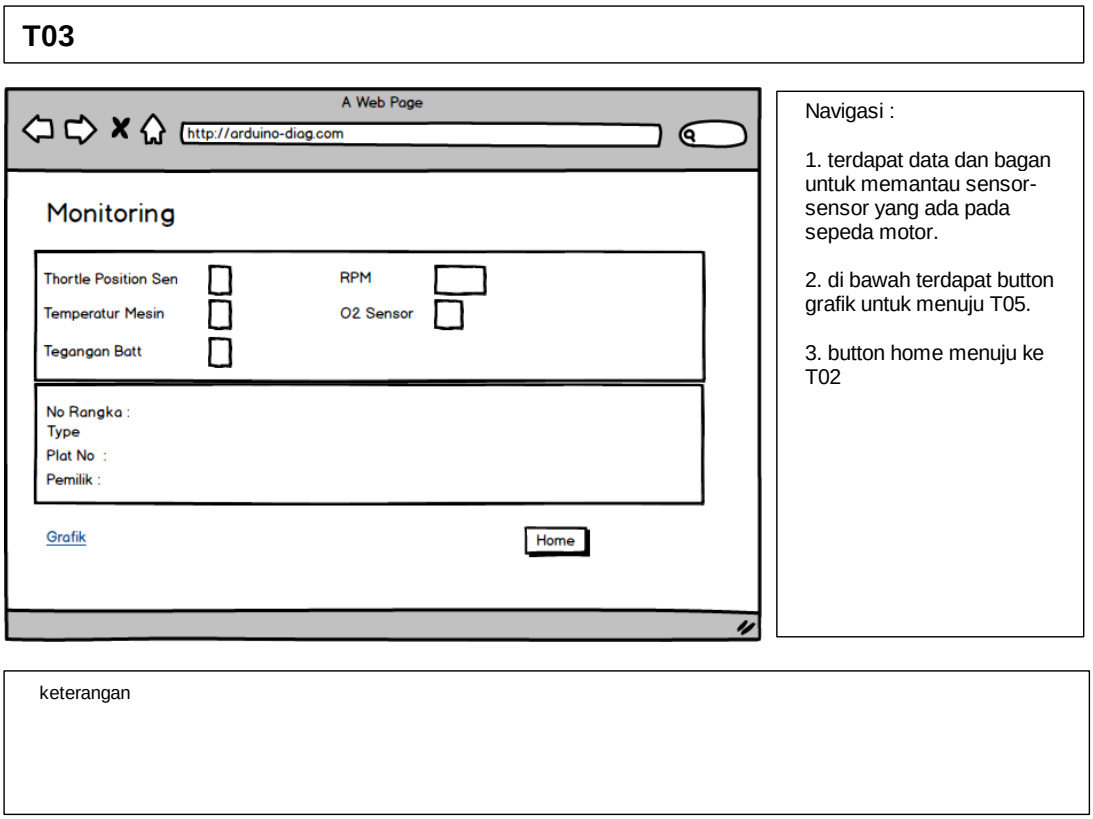
Halaman ini adalah tampilan menu utama yang didalamnya terdapat pilihan menu monitoring dan catatan diagnosa sepeda motor.



Gambar 3 21 Antarmuka Menu utama

c) Perancangan antarmuka menu monitoring.

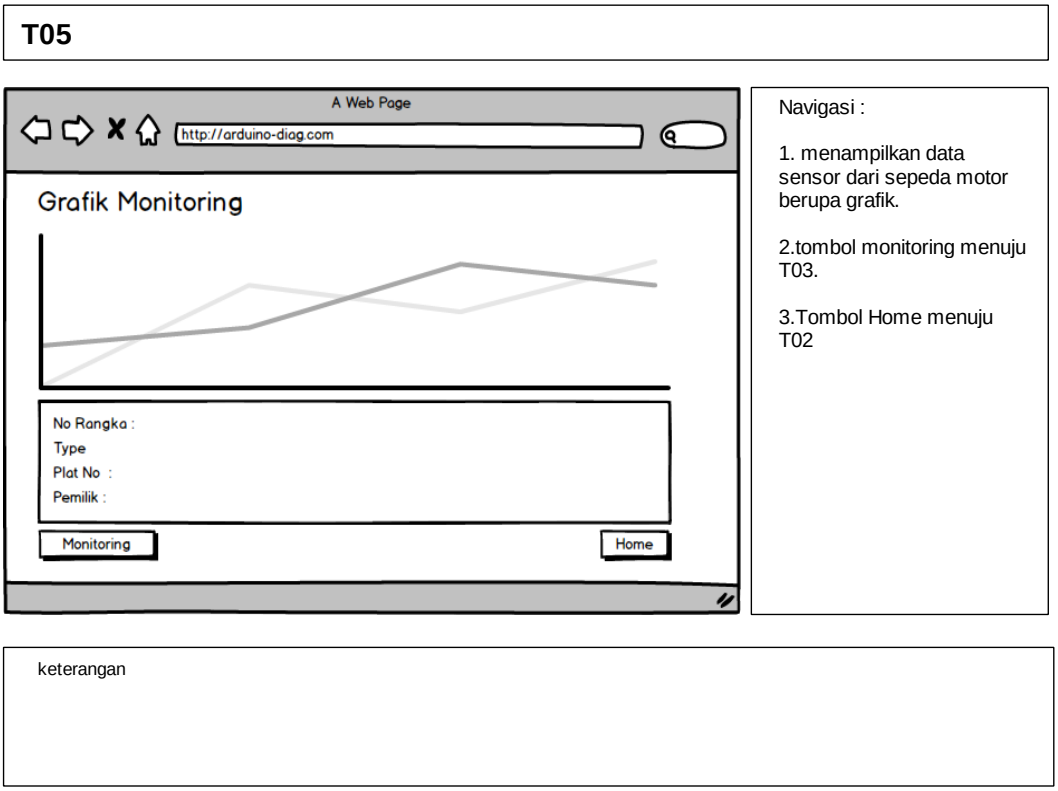
Halaman ini menampilkan data sensor yang terdapat pada sepeda motor untuk mngetahui kondsi sepedda motor.



Gambar 3 22 Antarmuka Monitoring

d) Perancangan antarmuka menu grafik.

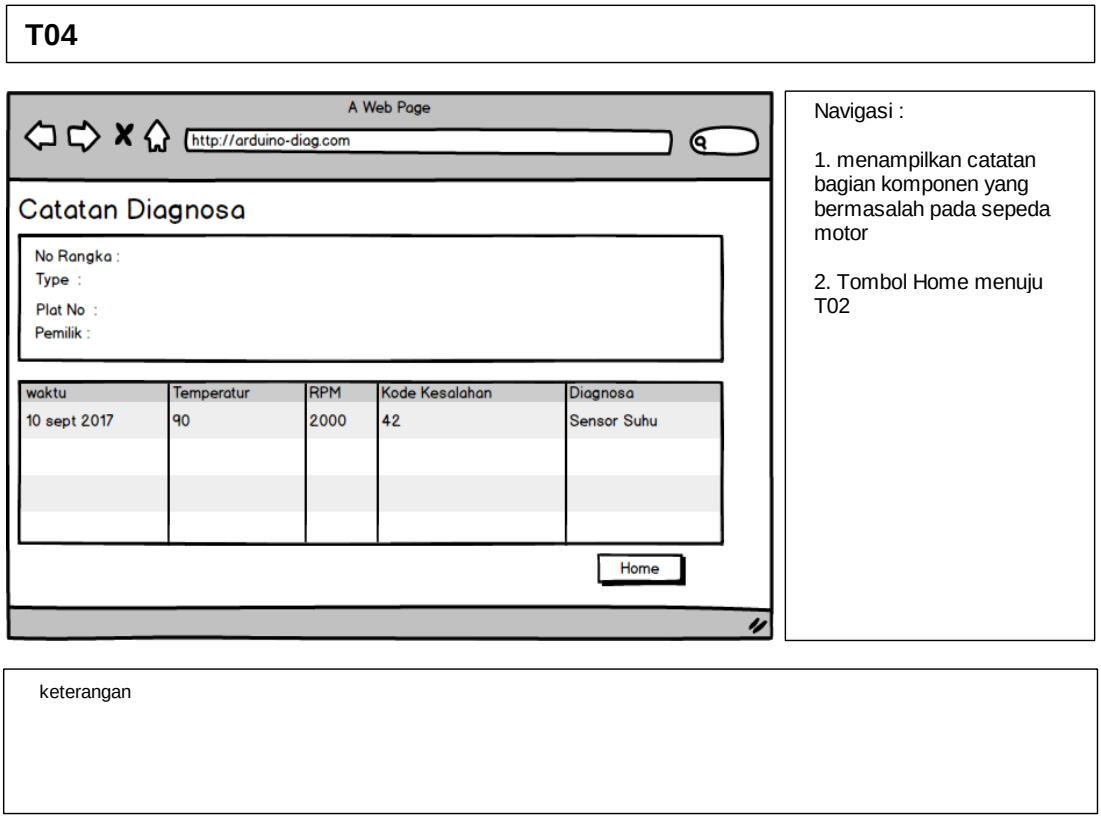
Halaman ini menampilkan data sensor yang terdapat pada sepeda motor dengan menu tampilan grafik.



Gambar 3 23 Antarmuka menu grafik monitoring

e) Perancangan antarmuka diagnosa history

Halaman ini menampilkan suatu masalah yang terjadi pada sepeda motor untuk dijadikan refrensi dalam hal perbaikan sepeda motor.

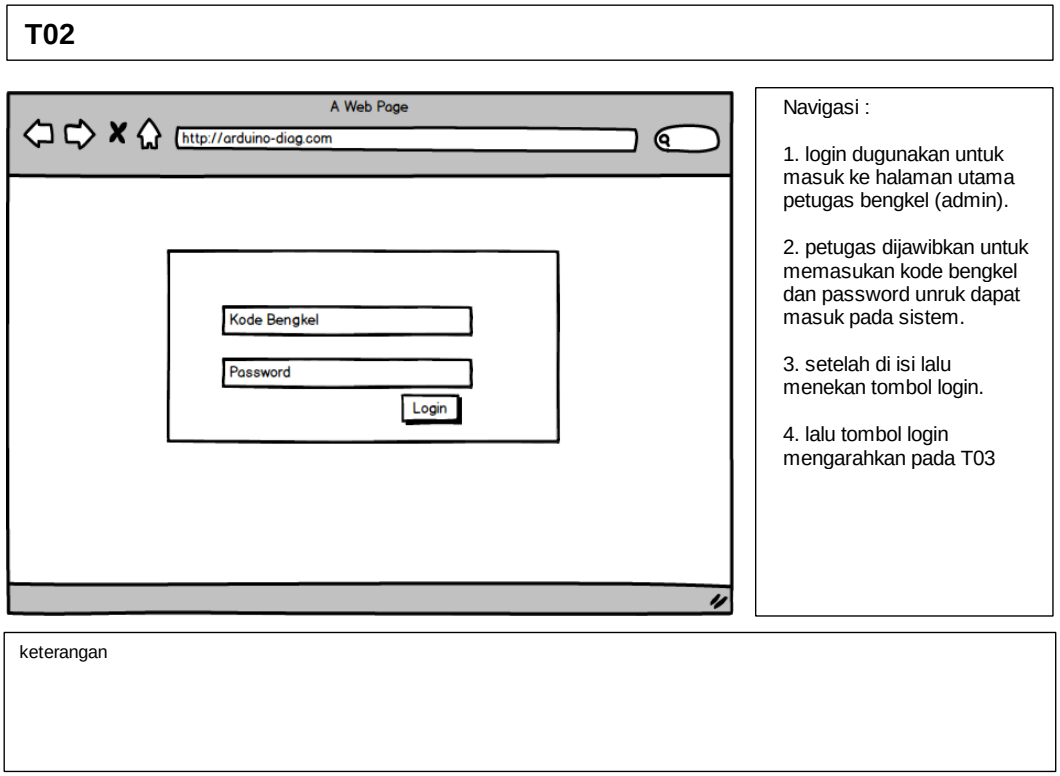


Gambar 3 24 Antarmuka Catatan diagnosa

2) Perancangan antamuka petugas bengkel

a) Perancangan antarmuka login

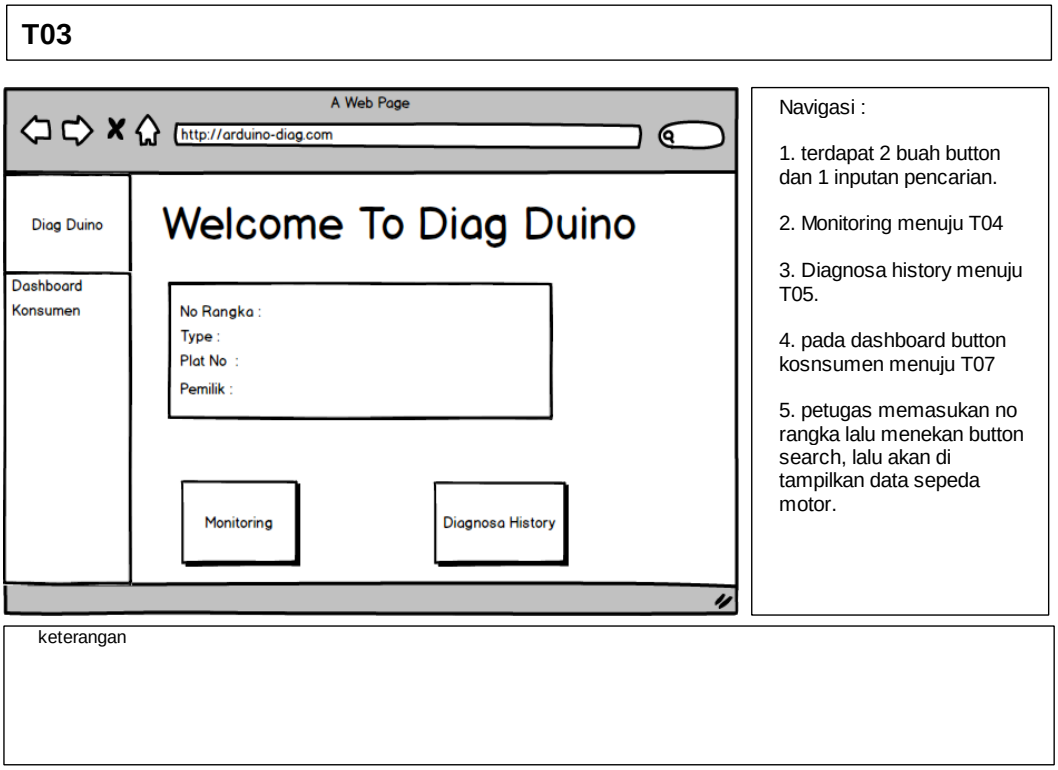
Perancangan ini akan menampilkan halaman login pada aplikasi *web* yang merupakan awal aplikasi dijalankan untuk petugas bengkel.



Gambar 3 25 Antarmuka login petugas bengkel

b) Perancangan antarmuka menu utama

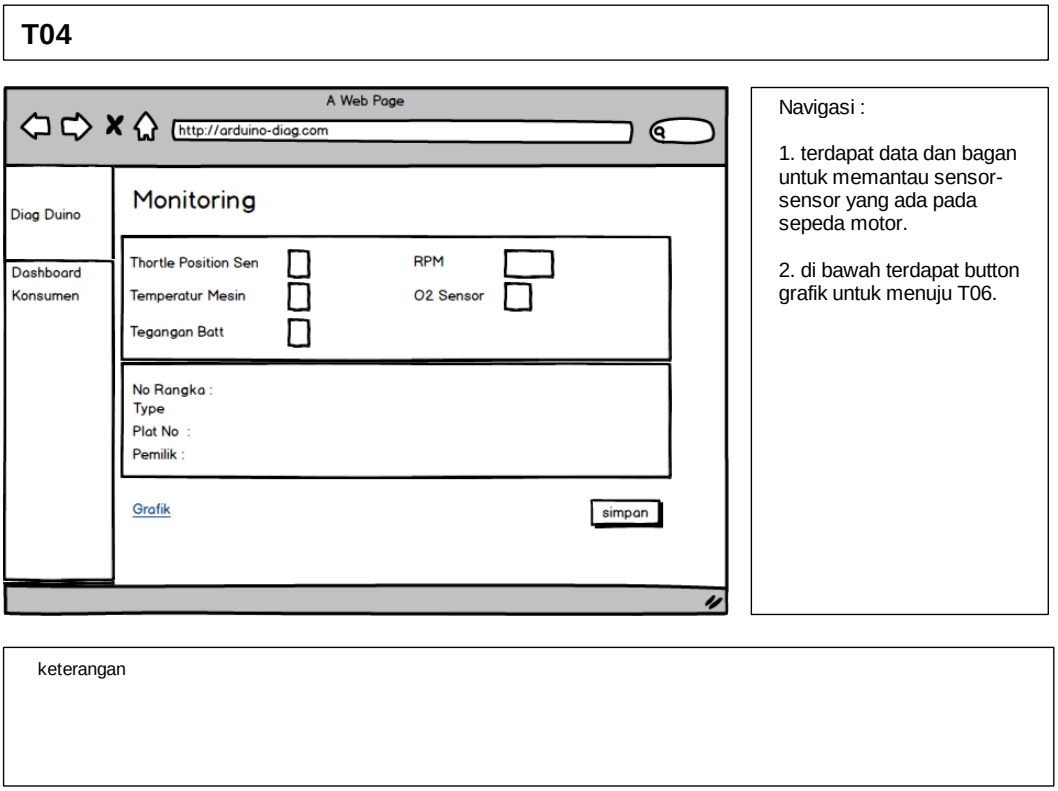
Halaman ini adalah tampilan menu utama yang didalamnya terdapat pilihan menu monitoring dan catatan diagnosa sepeda motor.



Gambar 3 26 Antarmuka menu utama petugas bengkel

c) Perancangan antarmuka monitoring

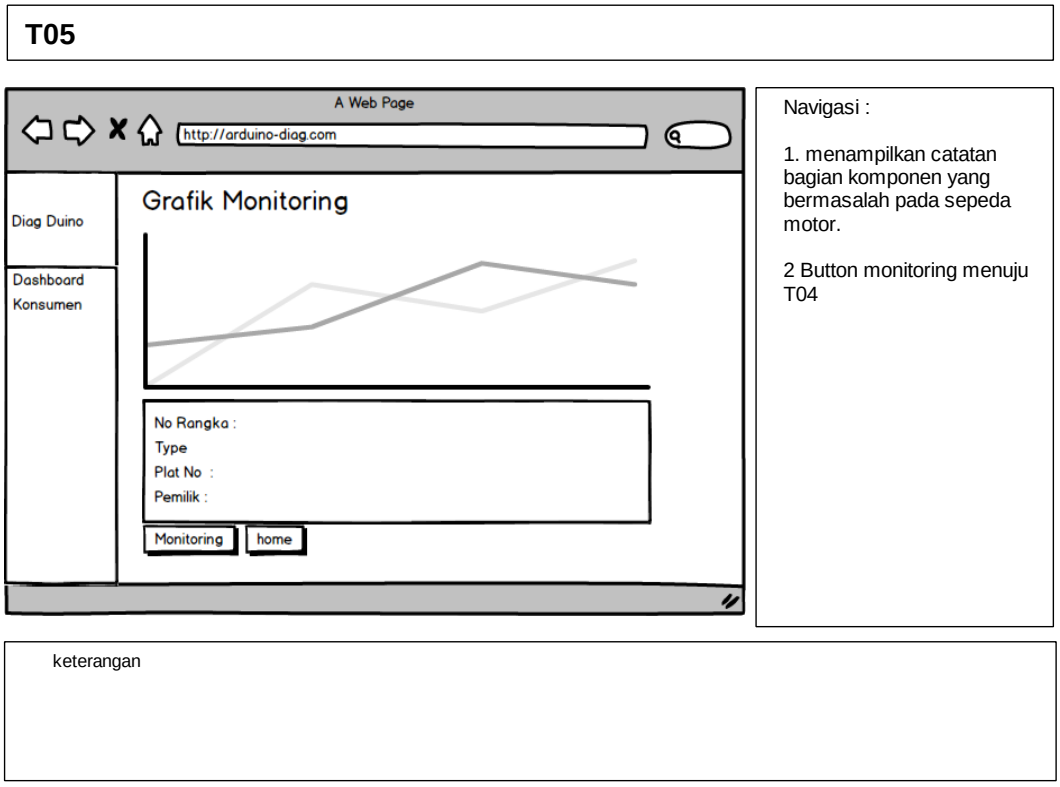
Halaman ini menampilkan data sensor yang terdapat pada sepeda motor untuk mngetahui kondsi sepedda motor.



Gambar 3 27 Antarmuka menu monitoring

d) Perancangan antarmuka menu grafik

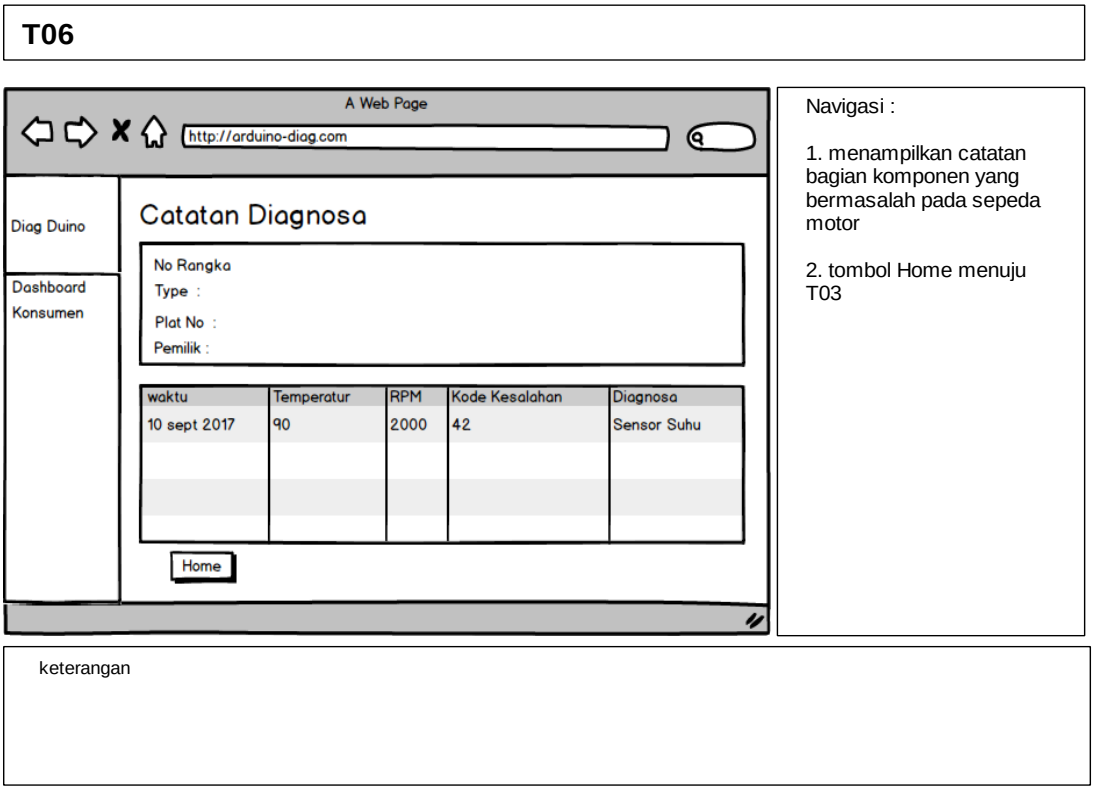
Halaman ini menampilkan data sensor yang terdapat pada sepeda motor dengan menu tampilan grafik.



Gambar 3 28 Antarmuka menu grafik monitoring petugas

e) Perancangan antarmuka diagnosa history

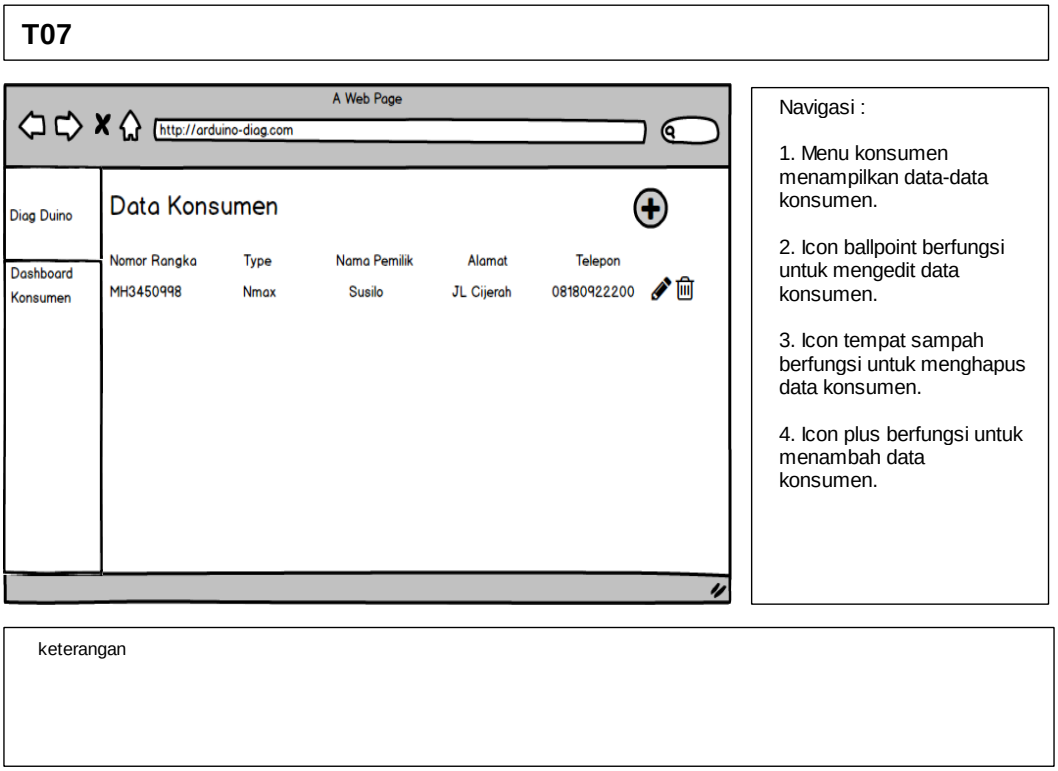
Halaman ini menampilkan suatu masalah yang terjadi pada sepeda motor untuk dijadikan refrensi dalam hal perbaikan sepeda motor.



Gambar 3 29 Antarmuka catatan diagnosa

f) Perancangan antarmuka data pemilik/konsumen

Halaman ini menampilkan data kendaraan dan pemilik sepeda motor.



Gambar 3 30 Antarmuka menu data pemilik

3.3.3.3. Perancangan pesan

Perancangan pesan dilakukan untuk menyesuaikan informasi kepada pengguna terhadap status kegagalan dari aktifitas yang dilakukan, seperti melakukan gagal login dan menghapus data.

Tabel 3 22 Perancangan pesan

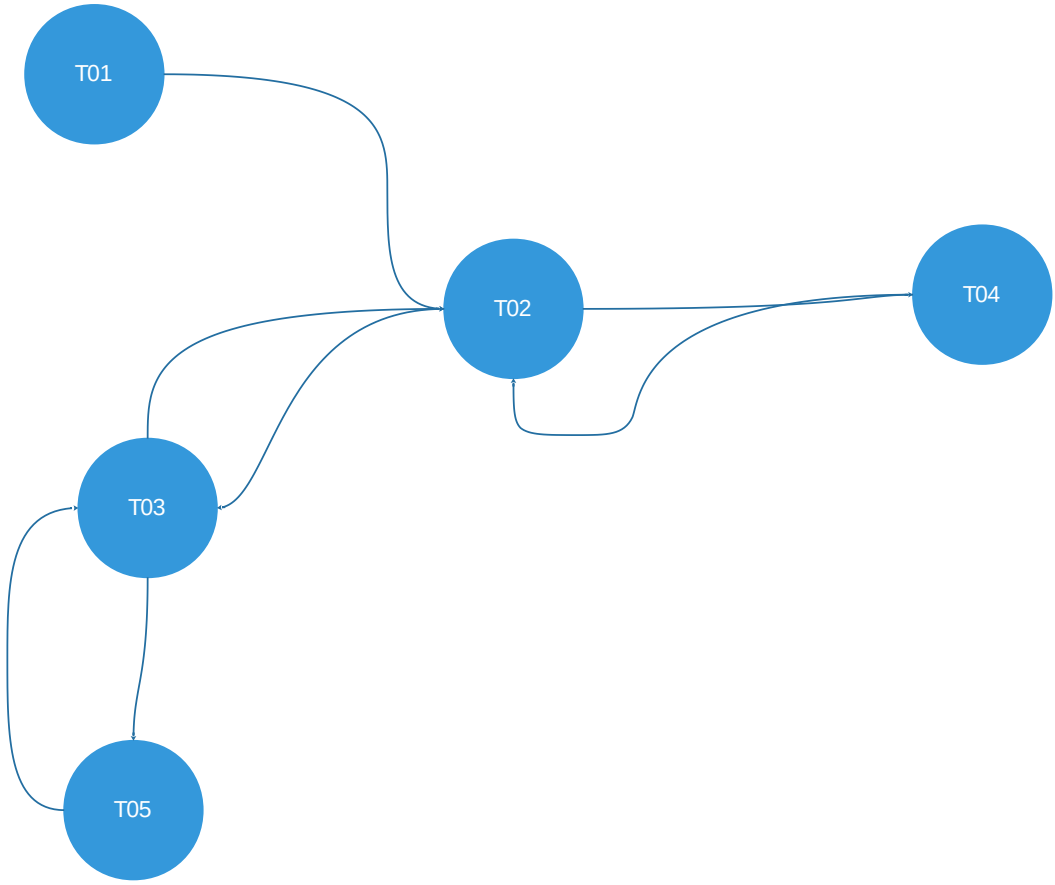
Kode Pesan	Pesan	Digunakan pada tampilan
M01	Wrong detail login	T02 (petugas)
M02	Are you sure ?	T07 (petugas)
M03	Nomor rangka wajib di isi	T03 (Petugas)
M04	Nomor Belum terdaftar	T03 (petugas)
M05	Maaf data kendaraan yang anda	T01 (konsumen)

	masukan belom terdaftar	

3.3.3.4. Jaringan semantik

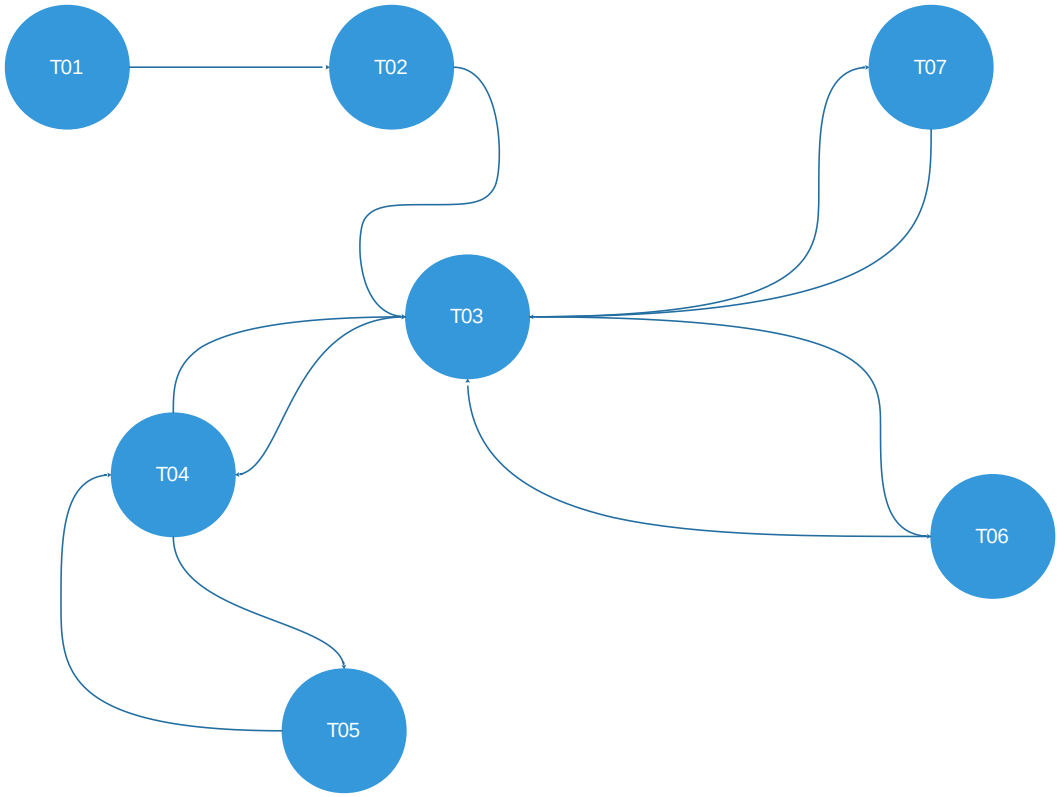
Jaringan semantik merupakan gambaran mengenai hubungan navigasi menu dari satu halaman ke halaman lainnya. Adapun jaringan semantik yang menggambarkan kontrol sistem yng akan dibangun adalah sebagai berikut :

a) Rancangan semantik konsumen



Gambar 3 31 Jaringan semantik konsumen

b) Perancangan jaringan semantik petugas



Gambar 3 32 Jaringan semantik petugas bengkel