

BAB III

ANALISA DAN DESAIN SISTEM

III.1. Analisis Sistem Yang Berjalan

Kendaraan merupakan salah satu kebutuhan primer pada saat ini, buktinya banyak masyarakat menggunakan kereta sebagai alat transportasi untuk digunakan sehari – hari. Oleh karena itu banyak masyarakat membeli kereta untuk keperluan primernya. Pada zaman sekarang ini banyak jenis dan merk kereta di Indonesia, seperti Honda, Yamaha, Suzuki, dan lain sebagainya. Namun jenis kereta Yamaha banyak terjual setiap tahunnya kepada masyarakat umum. Alasan dari mereka karena Yamaha memiliki ketahanan mesin yang kuat dan kecepatan yang di idamkan pecinta kereta. Namun bagi mereka yang ingin membeli kereta, mereka sulit menemukan lokasi penjualan kereta Yamaha. Terkadang mereka harus bertanya kepada orang-orang lain untuk menemukan lokasi penjualan kereta Yamaha. Dengan alasan itu masyarakat membutuhkan sebuah sistem yang dapat membantu masyarakat menemukan lokasi penjualan kereta Yamaha.

III.2. Penerapan Metode

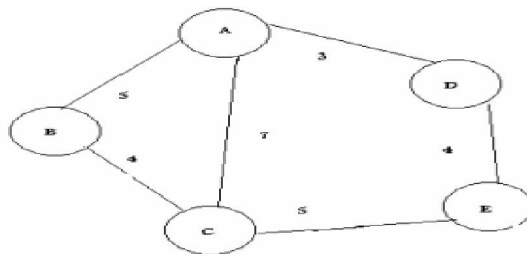
Setelah melihat permasalahan diatas maka penulis mencoba untuk merancang suatu aplikasi penemuan rute terpendek titik lokasi penjualan kereta

yamaha yang lebih baik sehingga dapat mencari titik lokasi terpendek dengan tepat. Dengan menggunakan metode *ant colony*, masalah *route* lokasi penjualan kereta dapat teratasi. Dimana *Ant Colony* diadopsi dari perilaku koloni semut yang dikenal sebagai sistem semut. Koloni semut dapat menemukan rute terpendek antara sarang dan sumber makanan.(Budi Triandi, *CSRID Journal*, Vo. 4, No. 2. Juni 2012, Hal.73-80). Untuk mendapatkan *route* terpendek pada sistem ini, terdapat tahap yang dilalui yaitu :

1. Masukkan titik awal dan titik tujuan yang berupa fasilitas umum
2. Inisialisasi parameter berupa α , β , ρ , q_0 , dan m , dimana nilai parameter ini telah ditentukan dan merupakan nilai terbaik.
3. Proses menghitung dengan menggunakan algoritma semut. Untuk memulai perhitungan tentukan beberapa siklus, dimana siklus pertama dilakukan proses pencarian dengan beberapa jumlah semut lalu dimulai dengan menghitung nilai probabilitas. Dimulai dengan siklus pertama dan semut pertama lalu kemudian perhitungan probabilitas dan nilai random, jika kota tujuan dicapai maka dihitung jarak jalan yang dilalui jika tidak maka kembali menghitung nilai probabilitas. Perhitungan akan terus terjadi hingga semut mencapai jumlah maksimal. Jumlah maksimal yang telah ditentukan misalnya sebanyak 5 semut. Setelah mencapai jumlah maksimal semut maka perbaharui intensitas jarak feromon. Selanjutnya perhitungan diulang lagi hingga mencapai siklus maksimum sebanyak 5 kali.

4. Jika jumlah semut dan siklus maksimal telah terpenuhi maka akan tampil jarak dan rute terpendek.
5. Jika tidak ditemukan jarak dan *rute* terpendek maka terjadi kondisi terbalik atau disebut *inverting condition*, dimana mengubah titik awal dan titik tujuan menjadi sebaliknya.
6. Tampil Jarak dan *rute* terpendek. (Deisy Novita Sari dan Septya Maharani, Vol. IX, No. 1, February 2014, ISSN : 1978-0087)

Berikut contoh kasus 1 Pencarian jalur terpendek dengan jarak :



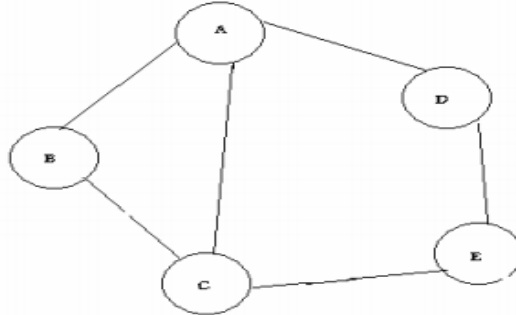
Gambar III.1. Contoh kasus jalur Terpendek dengan Jarak

Gambar diatas merupakan jenis kasus dengan mengetahui jarak antar titik.

Tabel III.1. Tabel Jarak Antar Titik

	A	B	C	D	E
A	0	5	7	3	
B	5	0	4		
C	7	4	0		5
D	3			0	4
E			5	4	0

Berikut contoh kasus 2 Pencarian jalur terpendek tanpa jarak:



Gambar III.2. Contoh kasus jalur Terpendek Tanpa Jarak

Gambar diatas merupakan jenis kasus dengan mengetahui titik koordinatnya saja.

Tabel III.2. Tabel Jarak Antar Titik

Titik Awal	Nama Titik	Titik Koordinat	
		Lat (x)	Long (y)
A	Showroom 1	3,6120014	98,6713624
B	Showroom 2	3,6810883	98,6901869
C	Showroom 3	3,6888385	98,6578553
D	Showroom 4	3,5984784	98,6668801
E	Showroom 5	3,6764884	98,6539673

Berdasarkan tabel III.2 diatas pencarian jalur terpendek dengan mengetahui titik koordinatnya maka dapat dilakukan penentuan dij yaitu jarak antar titik , dilakukan perhitungan atau rumus berdasarkan persamaan berikut ini :

$$d_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$$

$$d_{ij} = \sqrt{(3,610014 - 3,6810883)^2 + (98,671364 - 98,6901869)^2}$$

$$= \sqrt{-0,0710743^2 + -0,0188229^2}$$

$$= \sqrt{0,00505155612049 + 0,00035430156441}$$

$$= \sqrt{0,0054058576849}$$

$$= 0,07352453797814713231875358519156 * 111.319$$

$$= 8,1846780431893606225913303499393 \text{ km}$$

Selanjutnya dilakukan hal yang sama untuk setiap jalur, setelah dilakukan perhitungan akan terlihat seperti yang ada pada tabel III.3. berikut :

Tabel III.3. Tabel Total Jarak Antar Titik

Titik Asal	Nama Titik	Titik Koordinat				Jarak (dij)
		Lat (x1)	Long (y1)	Lat (x2)	Long (y2)	
A=>B	Showroom 1	3,610014	98,671364	3,6810883	98,671364	8,18
B=>C	Showroom 2	3,6810883	98,6901869	3,6888385	98,6578553	12,245
C=>D	Showroom 3	3,6888385	98,6578553	3,5984784	98,6668801	54,5463
D=>E	Showroom 4	3,5984784	98,6668801	3,6764884	98,6539673	8,794201

Kemudian akan ditentukan total jarak minimum dari semut dari masing-masing siklus.

$$L_{\min NC} = \text{Min} ((d_{ij1}), (d_{ij2}), (d_{ij3}), (d_{ij4}))$$

$$L_{\min NC} = \text{Min} ((8,18), (12,245), (54,5463), (8,794201))$$

$$L_{\min NC} = 8,18 \text{ km}$$

III.3. Desain Sistem

Berdasarkan hasil analisa sistem pada penelitian ini adapun desain sistem secara global untuk penggambaran model sistem secara garis besar.

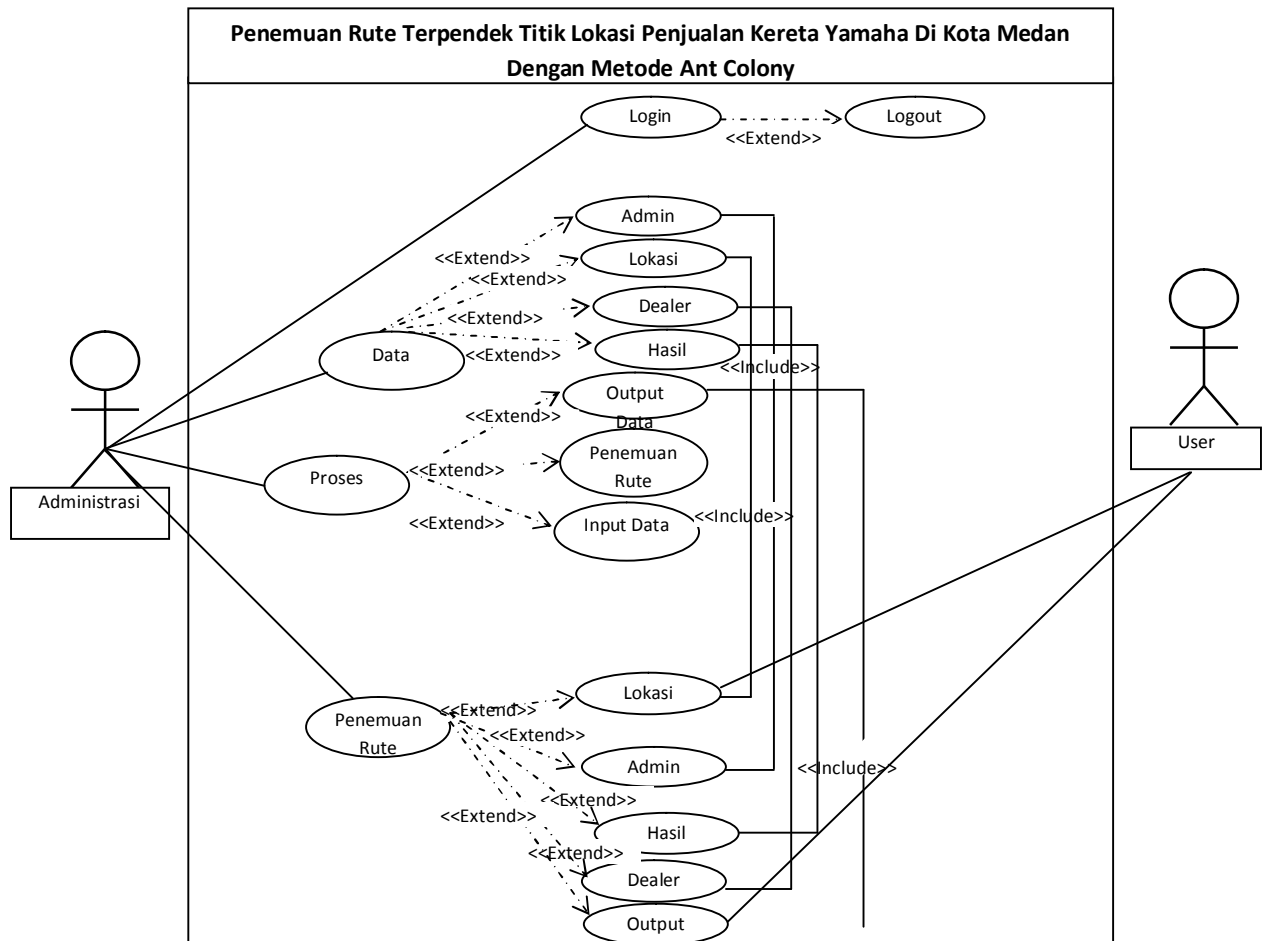
III.3.1. Desain Sistem Secara Global

Desain sistem atau perancangan sistem adalah proses pengembangan spesifikasi baru berdasarkan hasil rekomendasi analisis sistem. Dalam tahap perancangan, diharuskan merancang spesifikasi yang dibutuhkan.

Bentuk rancangan sistem yang penulis buat menggunakan beberapa bentuk diagram dari *UML (Unified Modeling Language)* yaitu *Use Case Diagram, Class Diagram, Sequence Diagram* dan *Activity Diagram*.

III.3.1.1 Use Case Diagram

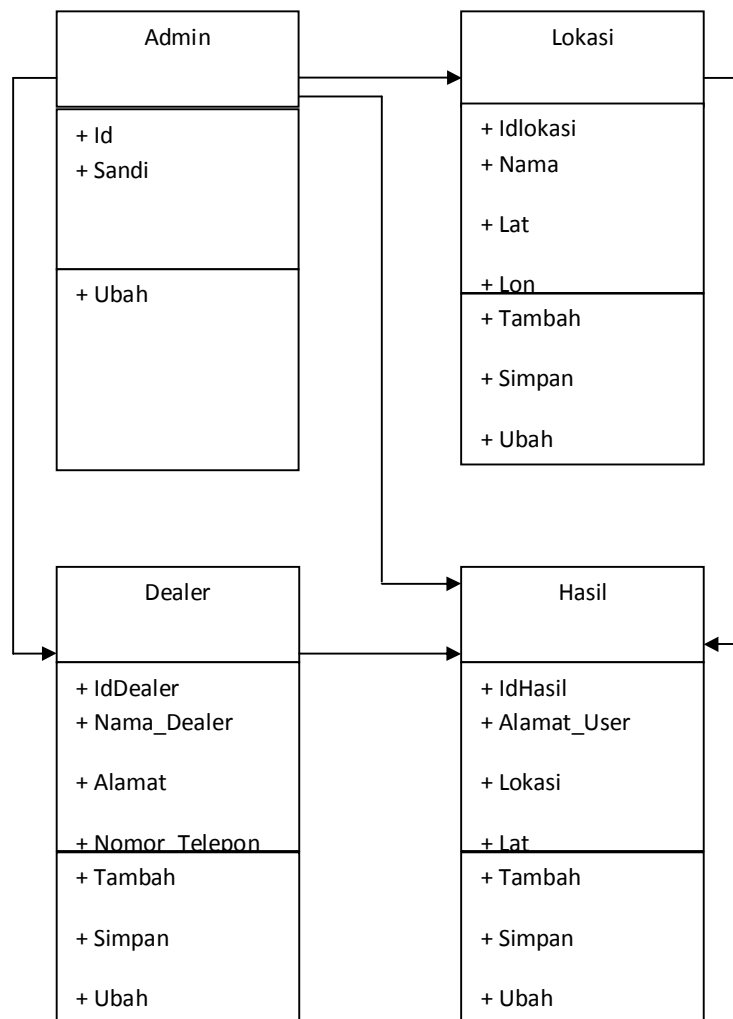
Perancangan dimulai dari identifikasi aktor dan bagaimana hubungan antara aktor dan *use case* didalam sistem. Perancangan *Use Case* Diagram dapat dilihat pada gambar III.3 berikut :



Gambar III.3 Use Case Penemuan Rute Terpendek Titik Lokasi Penjualan Kereta Yamaha Di Kota Medan Dengan Metode *Ant Colony*

III.3.1.2. Class Diagram

Class Diagram adalah sebuah spesifikasi yang akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek. Class menggambarkan keadaan (atribut/properti) suatu sistem. Rancangan kelas-kelas yang akan digunakan pada sistem yang akan dirancang dapat dilihat pada gambar III.4 :



Gambar III.4 Class Diagram Penemuan Rute Terpendek Titik Lokasi Penjualan Kereta Yamaha Di Kota Medan Dengan Metode *Ant Colony*

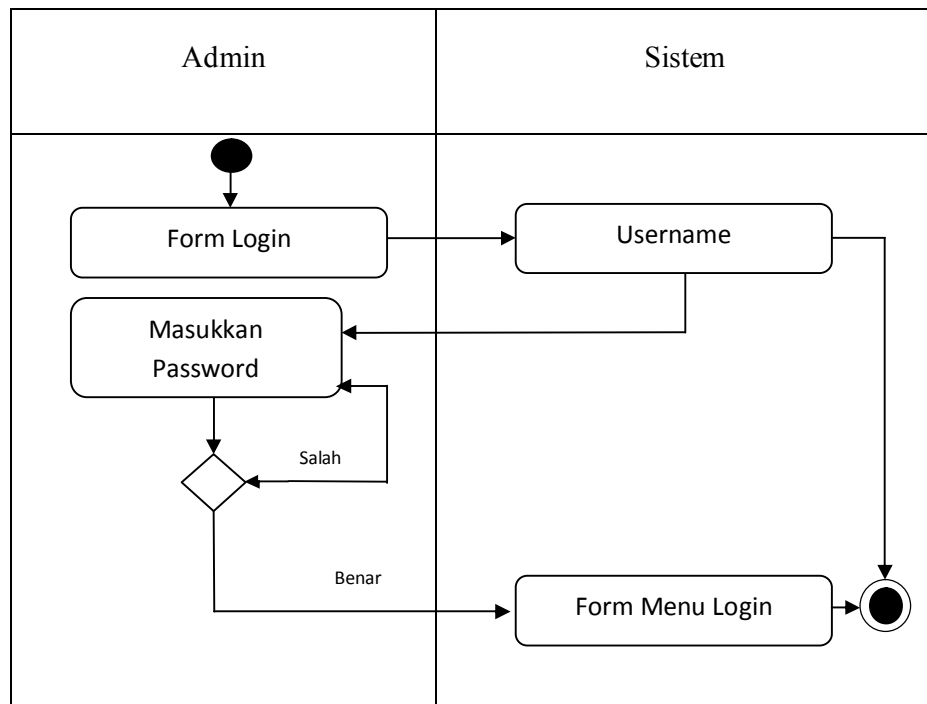
III.3.1.3. Activity Diagram Login

Pada proses ini kita akan membuat alur dari system yang dirancang yaitu *activity diagram*. Berikut adalah *activity diagram* sistem yang dirancang.

Aktivitas yang dilakukan untuk melakukan login admin dapat dilihat seperti pada gambar III.5 berikut :

1. Activity Diagram Form Input Login

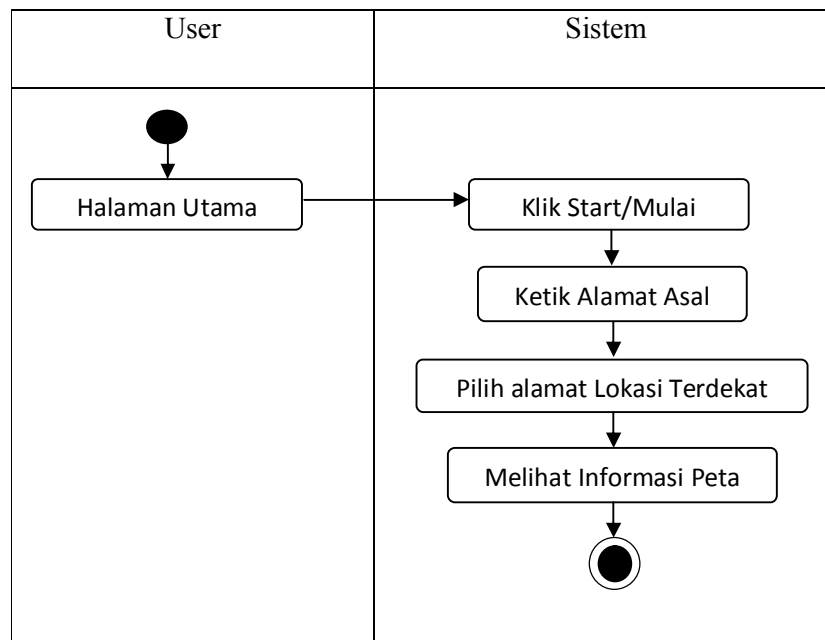
Activity diagram form input login dapat dilihat seperti pada gambar III.5 berikut :



Gambar III.5. Activity Diagram Login

2. Activity Diagram Melihat Peta

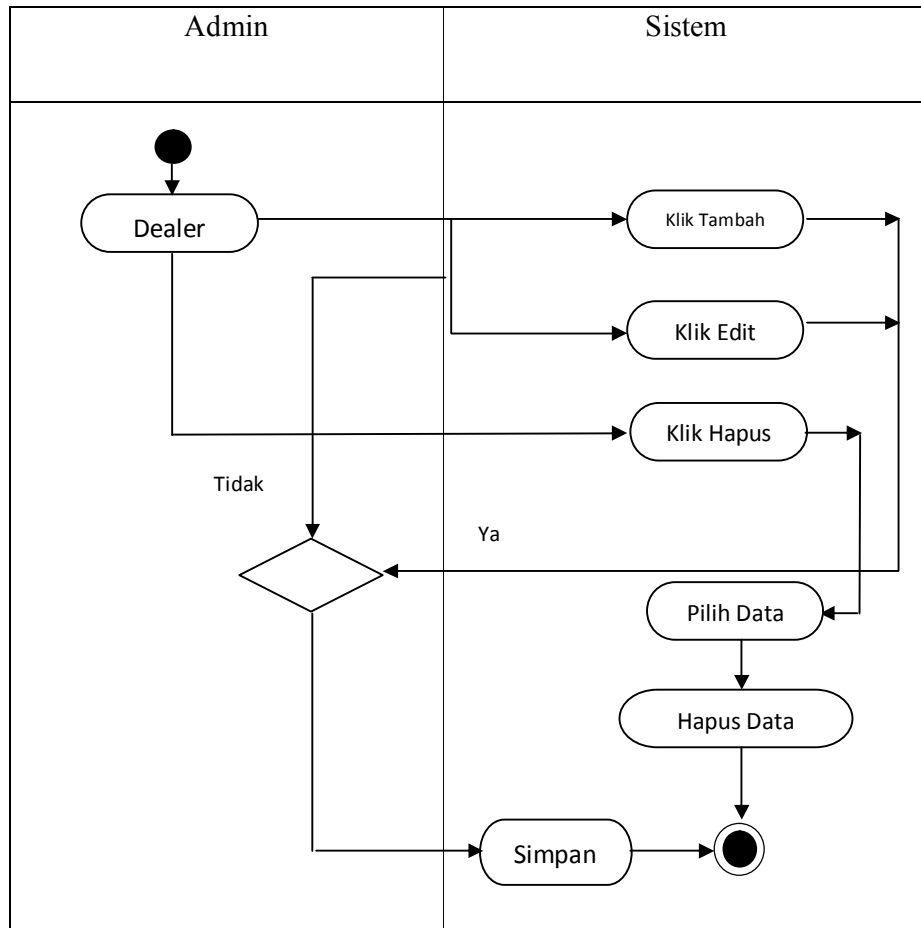
Aktivitas melihat peta diterangkan dalam langkah-langkah di awali dengan *start* atau klik mulai pada tampilan yang disediakan, kemudian ketik alamat asal lalu mencari lokasi yang terdekat, selanjutnya menekan tombol rute dan yang terakhir melihat informasi yang disajikan dalam peta yang ditunjukkan pada gambar III.6 :



Gambar III.6. Activity Diagram Melihat Peta

3. Activity Diagram Form Input Lokasi

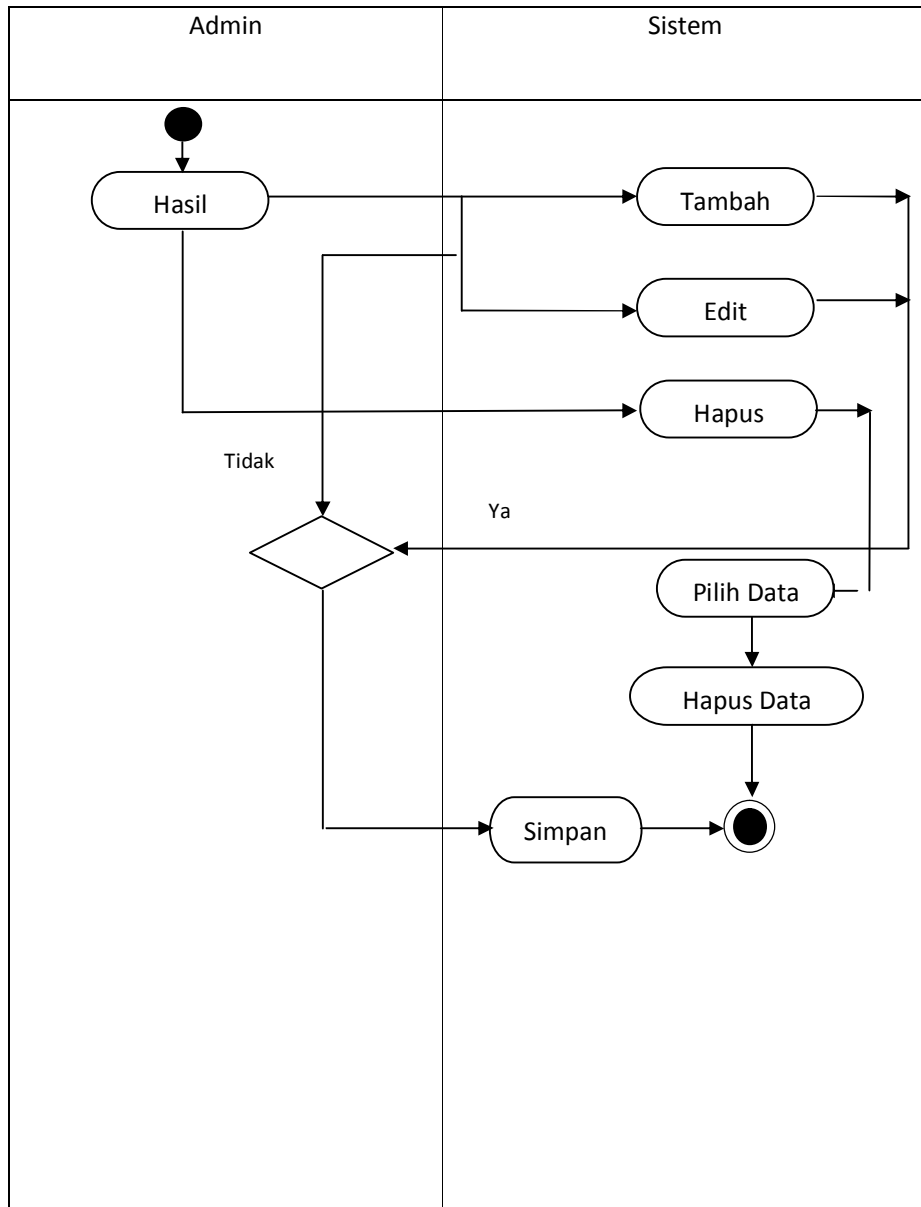
Activity diagram form Input Lokasi dapat dilihat seperti pada gambar III.7 berikut :



Gambar III.8. Activity Diagram Form Input Dealer

4. *Activity Diagram Form Input Hasil*

Activity diagram form Input Hasil dapat dilihat seperti pada gambar III.9 berikut :



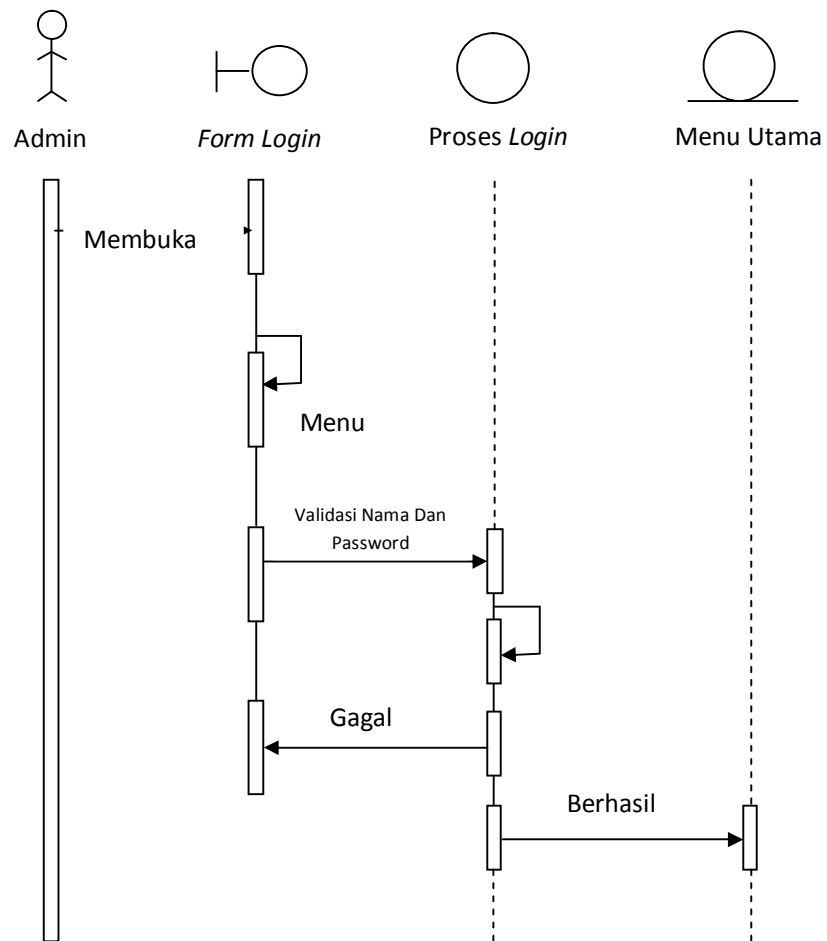
Gambar III.9. Activity Diagram Form Input Metode

III.3.1.4. Sequence Diagram

Rangkaian kegiatan pada setiap terjadi *event* sistem digambarkan pada *sequence* diagram berikut:

1. Sequence Diagram Login

Serangkaian kerja melakukan login admin dapat terlihat seperti pada gambar III.10 berikut :

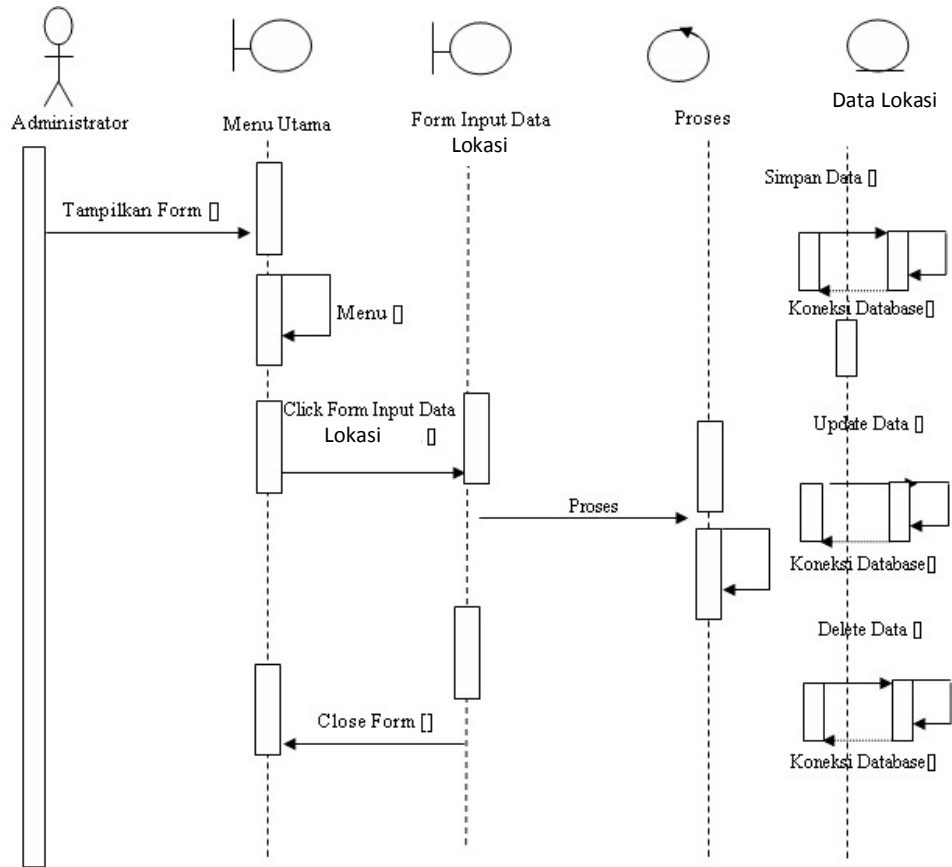


Gambar III.10. Sequence Diagram Login

2. Sequence Diagram Lokasi

Sequence diagram data Lokasi dapat dilihat seperti pada gambar III.11.

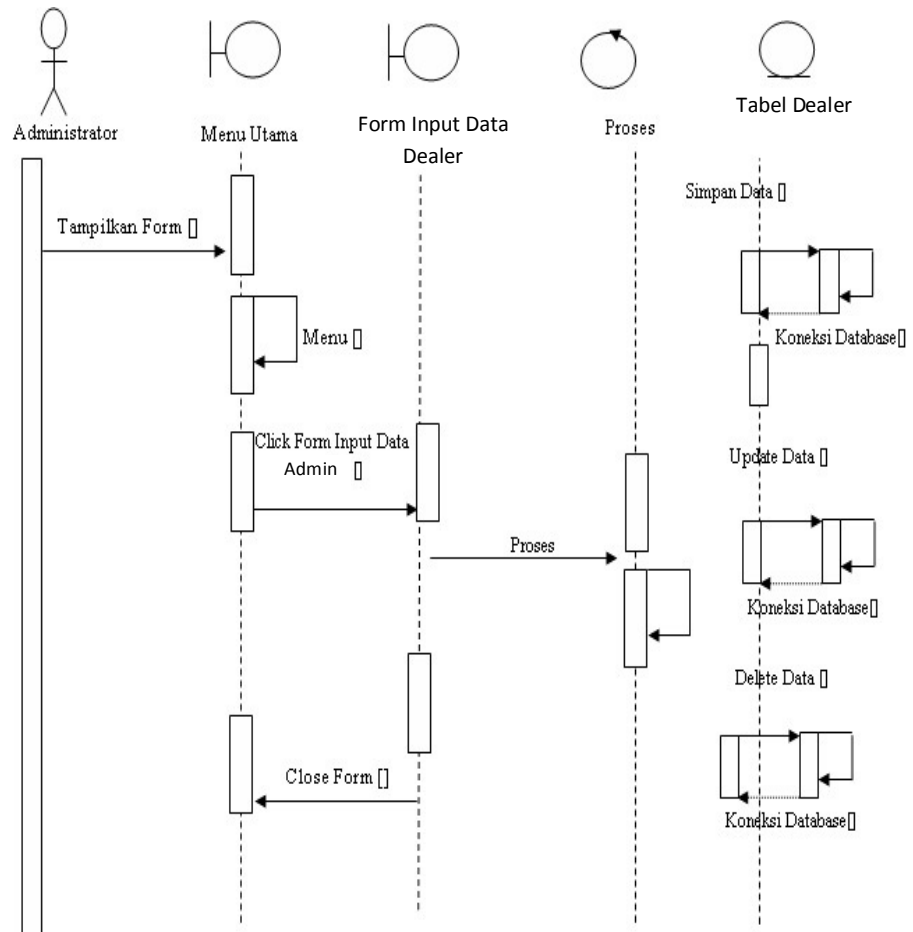
berikut :



3. Sequence Diagram Dealer

Sequence diagram data Dealer dapat dilihat seperti pada gambar III.12.

berikut :

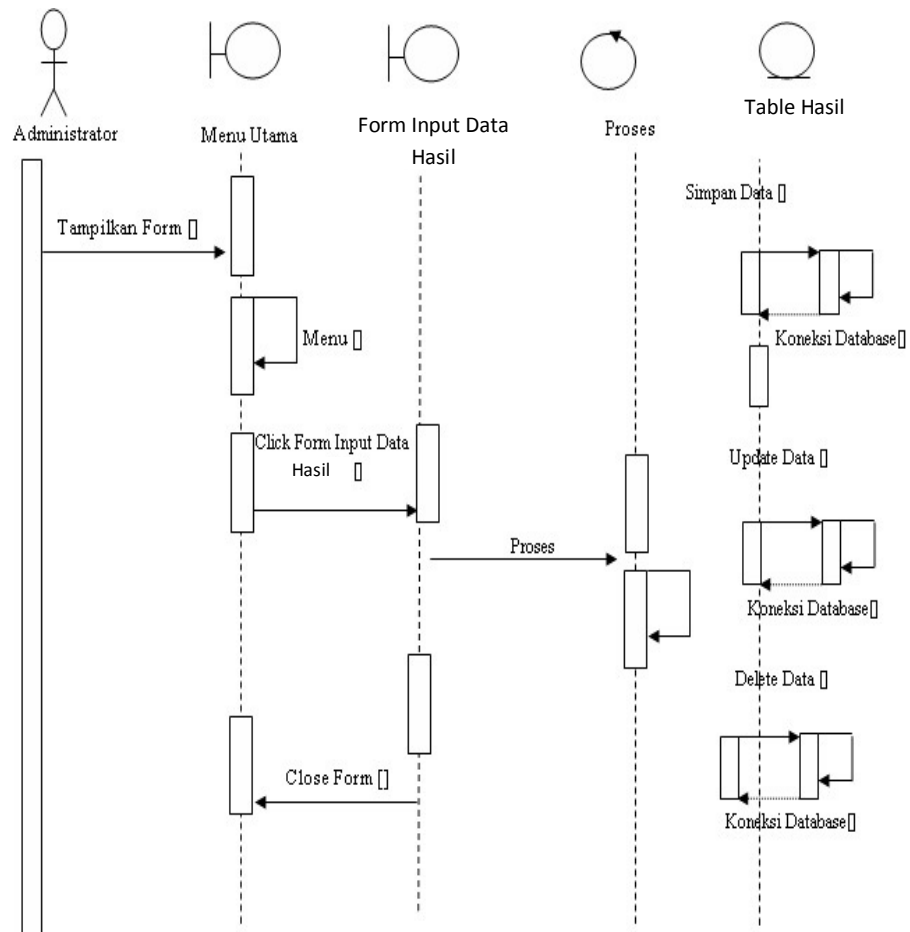


Gambar III.12. Sequence Diagram Form Dealer

4. Sequence Diagram Hasil

Sequence diagram data Hasil dapat dilihat seperti pada gambar III.13.

berikut :



Gambar III.13. Sequence Diagram Form Hasil

III.4. Desain Database

Desain *database* terdiri dari tahap merancang kamus data, melakukan normalisasi tabel, merancang desain tabel, dan desain user interface.

III.4.1 Kamus Data (*Data Dictionary*)

Kamus data merupakan suatu daftar terorganisasi tentang komposisi elemen data, aliran data dan data store yang digunakan. Pengisian kamus data dilakukan setiap hari selama proses pengembangan berlangsung, ketika diketahui adanya data atau saat diperlukan penambahan data item ke dalam sistem. Berikut kamus data dari sistem penemuan rute terpendek penjualan sepeda motor :

1. Login = Id + Sandi
2. Lokasi = Idlokasi + nama + lat + lon + icon + alamat
3. Dealer = Iddealer + nama_dealer + alamat + nomor_telepon
4. Metode = Idmetode + asal + tujuan + hasil

III.4.2 Normalisasi

Tahap normalisasi ini bertujuan untuk menghilangkan masalah berupa ketidak konsistenan apabila dilakukannya proses manipulasi data seperti penghapusan, perubahan dan penambahan data sehingga data tidak ambigu.

1. Bentuk Normal Pertama (1NF)

Bentuk normal pertama dari data produk merupakan bentuk tidak normal yang atribut kosongnya diisi sesuai dengan atribut induk dari *record*-nya, bentuk ini dapat dilihat pada tabel III.4 di berikut ini:

Tabel III.4 Data Penemuan Rute Terpendek 1NF

Idlokasi	Nama	Lat	Lon	Icon	Alamat
01	Shoowroom1	3.588387	98.73178	Icon.png	Jl. Sutomo
0	Shoowroom	3.57833	98.9837	Icon.png	Jl. Antasari

2. Bentuk Normal Kedua (NF)

Bentuk normal kedua dari data penemuan rute terpendek merupakan bentuk normal pertama, dimana telah dilakukan pemisahan data sehingga tidak adanya ketergantungan parsial. Setiap data memiliki kunci primer untuk membuat relasi antar data, bentuk ini dapat dilihat pada tabel III.5 dan tabel III.6 berikut ini:

Tabel III.5 Data Penemuan Rute Terpendek NF

Idlokasi	Nama	Lat	Lon
01	Shoowroom1	3.588387	98.73178
0	Shoowroom	3.57833	98.9837

Tabel III.6 Data Penemuan Rute Terpendek NF

Idlokasi	Lat	Lon
01	3.588387	98.73178
0	3.57833	98.9837

III.4.3. Desain Tabel

Setelah melakukan tahap normalisasi, maka tahap selanjutnya yang dikerjakan yaitu merancang struktur tabel pada basis data sistem yang akan dibuat, berikut ini merupakan rancangan struktur tabel tersebut :

1. Struktur Tabel Login

Tabel login digunakan untuk menyimpan data Login selengkapnya mengenai struktur tabel ini dapat dilihat pada tabel III.7 di bawah ini:

Nama Database : Rute

Nama Tabel : Login

Primary Key : Id

Foreign Key :-

Tabel III.7 Tabel Login

Nama Field	Tipe Data	Ukuran	Keterangan
Id	Int	-	Id Pencarian
Sandi	Varchar	100	Sandi Admin

2. Struktur Tabel Lokasi

Tabel Lokasi digunakan untuk menyimpan data Lokasi, selengkapnya mengenai struktur tabel ini dapat dilihat pada tabel III.8 di bawah ini:

Nama Database : Rute

Nama Tabel : Lokasi

Primary Key : Id

Foreign Key : -

Tabel III.8 Tabel Lokasi

Nama Field	Tipe Data	Ukuran	Keterangan
Idlokasi	Int	-	Id Pencarian
Nama	Varchar	50	Nama Showroom
Lat	Varchar	50	Nilai Berupa Angka
Lon	Varchar	50	Nilai Berupa Angka
Icon	Varchar	50	Gambar
Alamat	Varchar	50	Lokasi

3. Struktur Tabel Dealer

Tabel Dealer digunakan untuk menyimpan data Dealer, selengkapnya mengenai struktur tabel ini dapat dilihat pada tabel III.9 di bawah ini:

Nama Database : Rute

Nama Tabel : Dealer

Primary Key : Id

Foreign Key : -

Tabel III.9 Tabel Dealer

Nama Field	Tipe Data	Ukuran	Keterangan
IdDealer	Int	-	Id Pencarian
Nama_Dealer	Varchar	50	Nama Dealer
Alamat	Varchar	50	Alamat
Nomor_Telepon	Varchar	50	Nomor Telepon

4. Struktur Tabel Hasil

Tabel Hasil digunakan untuk menyimpan data hasil metode, selengkapnya mengenai struktur tabel ini dapat dilihat pada tabel III.10 di bawah ini:

Nama Database : Rute

Nama Tabel : Hasil

Primary Key : Id

Foreign Key : -

Tabel III.10 Tabel Metode

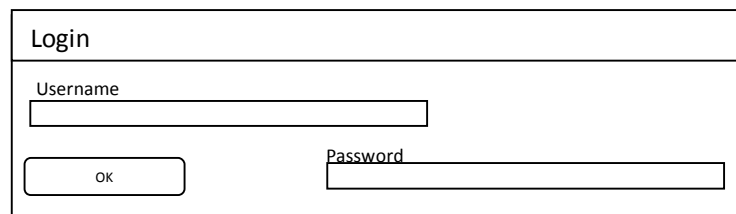
Nama Field	Tipe Data	Ukuran	Keterangan
Id	Int	-	Id Pencarian
Alamat_User	Varchar	50	Titik User
Lokasi	Varchar	50	Lokasi Showroom
Lat	Varchar	50	Koordinat X
Long	Varchar	50	Koordinat Y

III.4.4. Desain *User Interface*

Perancangan *Interface* yang dirancang adalah sebagai berikut :

1. Perancangan *Form Login*

Perancangan *form login* berfungsi untuk verifikasi pengguna yang berhak menggunakan sistem. Adapun rancangan form login dapat dilihat pada gambar III.14. sebagai berikut :



The diagram shows a rectangular box representing a login form. At the top left corner, the word "Login" is written. Below it, the label "Username" is positioned above a horizontal input field. To the right of the "Username" field, the label "Password" is positioned above another horizontal input field. Below the "Username" input field, there is a button labeled "OK".

Gambar III.14. Rancangan *Input Form Login*

2. Rancangan *Form Lokasi*

Rancangan *Form Lokasi* berfungsi untuk menambah, mengubah dan menghapus data Lokasi. Adapun rancangan *form Lokasi* dapat dilihat pada gambar III.15. sebagai berikut :

PENEMUAN RUTE TERPENDEK TITI LOKASI PENJUALAN KERETA YAMAHA	
	Lokasi Deale Hasil Keluar
	r
Id	<input type="text"/>
Nama :	<input type="text"/>
Lat	<input type="text"/>
Lon	<input type="text"/>
Alamat	<input type="text"/>
	<input type="button" value="Tambah"/> <input type="button" value="Simpan"/> <input type="button" value="Edit"/> <input type="button" value="Cari"/> <input type="button" value="Hapus"/>

3. Rancangan *Form Dealer*

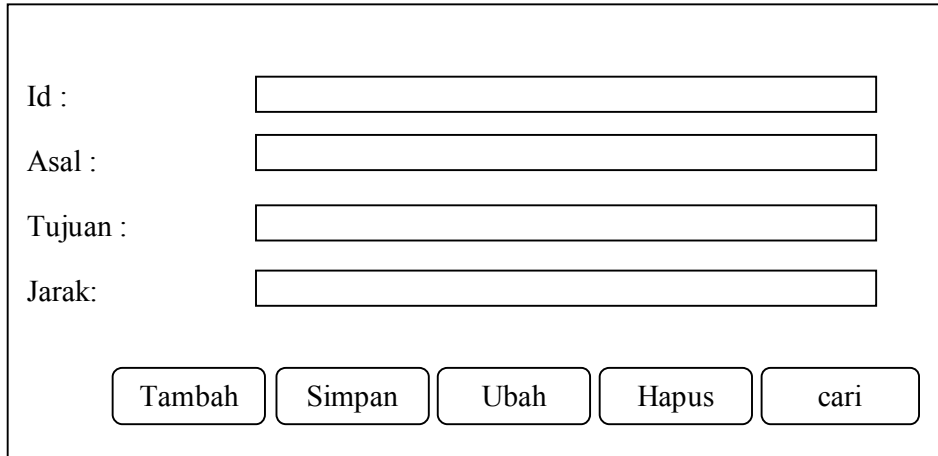
Rancangan *Form Dealer* berfungsi untuk menambah, mengubah dan menghapus data Dealer. Adapun rancangan *form Dealer* dapat dilihat pada gambar III.16. sebagai berikut :

Id :	<input type="text"/>
Nama Dealer :	<input type="text"/>
Alamat :	<input type="text"/>
Nomor Telepon :	<input type="text"/>
	<input type="button" value="Tambah"/> <input type="button" value="Simpan"/> <input type="button" value="Ubah"/> <input type="button" value="Hapus"/> <input type="button" value="cari"/>

Gambar III.16. Rancangan *Form Dealer*

4. Rancangan *Form* Hasil

Rancangan *Form* Hasil berfungsi untuk menambah, mengubah dan menghapus data Hasil. Adapun rancangan *form* Hasil dapat dilihat pada gambar III.17. sebagai berikut :



The image shows a form layout for 'Form Hasil'. It consists of four text input fields stacked vertically, each preceded by a label: 'Id :', 'Asal :', 'Tujuan :', and 'Jarak:'. Below these fields is a horizontal row of five buttons: 'Tambah', 'Simpan', 'Ubah', 'Hapus', and 'cari'. The entire form is enclosed in a rectangular border.

Gambar III.17. Rancangan *Form* Hasil