BAB III

ANALISIS DAN PERANCANGAN

3.1 Analisis Sistem

Analisis sistem dapat didefinisikan sebagai penguraian dari suatu sistem informasi yang utuh ke dalam bagian-bagian komponennya dengan maksud untuk mengidentifikasikan dan mengevaluasi permasalahan-permasalahan, kesempatan-kesempatan, hambatan-hambatan yang terjadi dan kebutuhan-kebutuhan yang diharapkan sehingga dapat diusulkan perbaikan-perbaikannya.

3.1.1 Analisis Masalah

PD. Wijaya Abadi Bandung adalah perusahaan yang bergerak di bidang jual beli sepeda motor bermerek Honda. Dalam kegiatan operaisonalnya, PD. Wijaya Abadi juga mendistribusikan sepeda motor terhadap konsumen perorangan maupun perusahaan.

Dalam suatu hari, di perusahaan ini bisa terjadi banyak transaksi, terutama pada proses distribusi penjualan sepeda motor. Karena jumlahnya yang sangat banyak, data sulit untuk dianalisa. Dari data yang sangat banyak tersebut perusahaan ingin adanya pengolahan data lebih lanjut untuk menemukan informasi/pengetahuan baru yang berguna sebagai pembantu dalam mengambil keputusan, dalam hal ini yaitu mendistribusikan sepeda motor ke perusahaan target distribusinya. Informasi yang dibutuhkan adalah mengklasifikasikan keterkaitan suatu perusahaan target pendistribusiannya dengan atribut tipe motor

dan kode warna pada data distribusi penjualan sepeda motor yang tersimpan dalam suatu *database* Wijaya Abadi, sehingga perusahaan memiliki bekal pengetahuan dan aturan dalam pendistribusiannya.

3.1.2 Analisis Data

Database PD.Wijaya Abadi terbagi ke dalam beberapa tabel, secara garis besar tabel-tabel tersebut adalah tabel Warna, tabel TipeMotor, tabel Perusahaan dan sebagai target data dalam penerapan teknik *data mining* adalah perusahaan target pendistribusian yang tersimpan dalam tabel DistribusiMotor. Setiap variabel/atribut pengambilan keputusan mewakili satu dimensi tertentu, dalam hal ini dimensi yang dilibatkan adalah DsitribusiMotor. Struktur tabel Distribusi Motor dapat ditunjukan pada Tabel 3.1.

Tabel 3. 1 Struktur Tabel DistribusiMotor

Tabel DistribusiMotor
TIPE_MOTOR
KODE_WARNA
TGL_DISTRIBUSI
NO_DISTRIBUSI
TGL_PENJUALAN
NAMA_PERUSAHAAN

Data yang tidak lengkap dan dan inkonsisten umumnya terjadi hampir pada setiap *database*, data yang tidak lengkap dapat disebabkan oleh berbagai macam sebab, seperti atribut dengan data yang salah. Demikian pula dengan *database* PD. Wijaya Abadi, ada sebagian atribut yang tidak terlalu diperlukan sehingga proses *Data Preprocessing* perlu dilakukan sehingga *database* sesuai dengan ketentuan yang diperlukan oleh sistem.

Data preprocessing merupakan hal yang penting dalam proses *data mining*, hal-hal yang termasuk di dalamnya adalah:

a. *Data Selection*

Sebelum masuk ke proses *Data Preprocessing,* yang harus dilakukan lebih awal adalah pemilihan data (*data selection*). Pada *database* PD.Wijaya Abadi, data yang akan diproses untuk di-*mining* yaitu data DistribusiMotor. Data DistribusiMotor ini nantinya akan menjadi Data Kasus dalam proses operasional *data mining*. Dari data yang ada, kolom yang diambil sebagai atribut/variabel keputusan adalah kolom ***nama_perusahaan***, sedangkan kolom yang diambil sebagai variabel penentu dalam pembentukan pohon keputusan adalah kolom:

1. **tipe_motor**
2. **kode_warna**

Pemilihan variabel-variabel tersebut dengan pertimbangan bahwa jumlah nilai variabelnya tidak banyak dan tidak unik sehingga diharapkan suatu motor masuk dalam satu klasifikasi nilai variabel tersebut cukup banyak.

b. *Data Preprocessing / Data Cleaning*

Data *cleaning* diterapkan untuk menambahkan isi atribut yang hilang atau kosong, dan merubah data yang tidak konsisten. Tahapan dalam *data cleaning* yang diterapkan pada tabel DistribusiMotor yang akan dijadikan *input* bagi proses *data mining* itu sendiri adalah sebagai berikut:

1. Mengabaikan tupel, dilakukan jika *value* dari suatu tupel hilang atau tidak ada. Metode ini sangatlah tidak efektif apabila terdapat banyak atribut dengan tupel-tupel kosong.
2. Menambahkan isi terhadap atribut yang kosong tersebut secara manual, namun pendekatan ini sangatlah memakan waktu dan tidak efektif bila diterapkan pada data yang sangat besar.

c. *Data Transformation*

Dalam proses ini, data ditransformasikan ke dalam bentuk yang sesuai untuk proses *data mining*.

d. *Data Reduction*

Reduksi data dilakukan dengan menghilangkan atribut-atribut yang tidak diperlukan sehingga ukuran dari *database* menjadi kecil dan hanya menyertakan atribut-atribut yang diperlukan dalam proses *data mining*. Hal ini dikarenakan proses *data mining* akan lebih efisien terhadap data yang lebih kecil. Dari proses *data reduction* tersebut maka dibuat tabel dengan atribut-atribut yang siap untuk proses klasifikasi, dalam hal ini tabel KASUS seperti pada Tabel 3.2.

Tabel 3. 2 Struktur Tabel KASUS

Tabel KASUS
TIPE_MOTOR
KODE_WARNA
NAMA_PERUSAHAAN

Dari proses *data preprocessing* di atas dihasilkan tabel yang siap untuk proses klasifikasi, dengan atribut dan nilai atribut yang ada pada Tabel 3.3.

Tabel 3. 3 Hasil Proses *Data Preprocessing*

TIPE_MOTOR	KODE_WARNA	NAMA_PERUSAHAAN
GL160D2	AH	GELORA FAJAR PERKASA PT
GL160CW2	AR	GEMILANG MAKMUR ABADI CV
GL200R1	BB	
NC110CW1	BE	
NC110CW2A/T	BG	
NC11A2CA/T	BK	
NC11BICA/T	BL	
NC11B2C	BP	
NC11B3CA/T	BR	
NC11B3DA/T	BS	
NF11A1CM/T	BZ	
NF11A1CAM/T	GR	
NF11B1CA	GS	
NF11BICM/T	HA	
NF11BIDM/T	HB	
NF11B1D1M/T	HJ	
NF11B1C1M/T	HS	
NF125TR1	HZ	
NF125TR2	LS	
NF125TRF1	MH	
NF125TRF2	MS	
NF125TD1	NK	
NF125TD2	OR	
CS12A1RR	OU	
	PH	
	PJ	
	PM	
	PP	
	PK	
	PS	
	RD	
	RS	
	SB	
	SH	
	SV	
	UO	
	VS	
	WG	
	WH	
	WP	

Atribut **tipe_motor** biasanya berisi nilai berupa jenis-jenis motor Honda. Nilai dari atribut **tipe_motor** di atas dapat dijelaskan pada Tabel 3.4.

Tabel 3. 4 Keterangan Atribut Tipe Motor

TIPE_MOTOR	Nama Motor
GL160D2	MEGA PRO D CW
GL160CW2	MEGA PRO SPOKE
GL200R1	TIGER CW
NC110CW1	VARIO CW
NC110CW2A/T	NEW VARIO CW
NC11A2CA/T	VARIO TECHNO
NC11BICA/T	NEW BEAT
NC11B2C	NEW BEAT CW
NC11B3CA/T	NEW BEAT SPOKE
NC11B3DA/T	BEAT
NF11A1CM/T	BLADE REPSOL
NF11A1CAM/T	BLADE
NF11B1CA	ABSOLUTE REVO
NF11BICM/T	ABSOLUTE REVO DELUXE
NF11BIDM/T	NEW ABSOLUTE REVO
NF11B1D1M/T	ABSOLUTE REVO CW
NF11B1C1M/T	ABSOLUTE REVO SPOKE
NF125TR1	SUPRA X 125 D
NF125TR2	SUPRA X 125 DD
NF125TRF1	SUPRA X 125 CW
NF125TRF2	SUPRA X 125R CW
NF125TD1	SUPRA X 125 INJ
NF125TD2	SUPRA X 125R INJ
CS12A1RR	CITY SPORT ONE

Sedangkan untuk atribut **kode_warna** ini berisi kombinasi dua warna. Nilai dari atribut **kode_warna** pada Tabel 3.3 dapat dijelaskan pada Tabel 3.5.

Tabel 3. 5 Keterangan Atribut Kode Warna

KODE_WARNA	Nama Warna
AH	ABU HITAM
AR	ABU RED
BB	BLACK
BE	BLACK E
BG	BLACK GREY
BK	BLACK K
BL	BLUE
BP	BIRU PUTIH

KODE_WARNA	Nama Warna
BR	BLACK RED
BS	BLACK SILVER
BZ	BLACK Z
GR	GREY RED
GS	GREY SILVER
HA	HITAM ABU
HB	HITAM BIRU
HJ	HITAM MERAH
HS	HITAM SILVER
HZ	HITAM Z
LS	LIGHT SILVER
MH	MERAH HITAM
MS	MERAH SILVER
NK	NILA KUNING
OR	ORANGE
OU	ORANGE UNGU
PH	PUTIH HITAM
PJ	PUTIH JINGGA
PM	PUTIH MERAH
PP	PUTIH PINK
PK	PUTIH KUNING
PS	PUTIH SILVER
RD	RED
RS	RED SILVER
SB	SILVER BIRU
SH	SILVER HITAM
SV	SILVER VIOLET
UO	U ORANGE
VS	VIOLET SILVER
WG	WHITE GREEN
WH	WHITE
WP	WHITE PINK

3.1.3 Analisis Metode *Decision Tree*

Dari proses *Data Preprocessing* yang telah dilakukan maka dihasilkan tabel yang siap untuk proses klasifikasi, seperti pada Tabel 3.6.

Tabel 3. 6 Tabel KASUS

No	tipe_motor	kode_warna	nama_perusahaan
1	GL200R1	BE	GELORA FAJAR PERKASA PT
2	NC110CW1	WG	GELORA FAJAR PERKASA PT
3	NC11BICA/T	NK	GELORA FAJAR PERKASA PT

No	tipe_motor	kode_warna	nama_perusahaan
4	NC11BICA/T	NK	GELORA FAJAR PERKASA PT
5	NC11BICA/T	BK	GELORA FAJAR PERKASA PT
6	NF125TR1	HJ	GELORA FAJAR PERKASA PT
7	NF125TR1	BG	GELORA FAJAR PERKASA PT
8	NF125TR1	BG	GELORA FAJAR PERKASA PT
9	NF125TR1	HJ	GELORA FAJAR PERKASA PT
10	NF11BICM/T	RD	GELORA FAJAR PERKASA PT
11	NF11BICM/T	BK	GELORA FAJAR PERKASA PT
12	NF11BICM/T	BK	GELORA FAJAR PERKASA PT
13	NF11BIDM/T	BB	GELORA FAJAR PERKASA PT
14	NF11BIDM/T	BB	GELORA FAJAR PERKASA PT
15	NF11BIDM/T	BS	GELORA FAJAR PERKASA PT
16	NF11BIDM/T	BS	GELORA FAJAR PERKASA PT
17	NF11BIDM/T	BR	GELORA FAJAR PERKASA PT
18	NF11BIDM/T	BR	GELORA FAJAR PERKASA PT
19	NF11BICM/T	RD	GELORA FAJAR PERKASA PT
20	NF11BICM/T	RD	GEMILANG MAKMUR ABADI CV
21	NC110CW1	WP	GEMILANG MAKMUR ABADI CV
22	NC110CW1	BK	GEMILANG MAKMUR ABADI CV
23	NC11BICA/T	NK	GEMILANG MAKMUR ABADI CV
24	NF11BICM/T	BK	GEMILANG MAKMUR ABADI CV
25	NF11BIDM/T	BB	GEMILANG MAKMUR ABADI CV
26	NF11BIDM/T	BR	GEMILANG MAKMUR ABADI CV

Tabel KASUS adalah kumpulan data yang akan diproses, data di atas merupakan 26 data sampel yang diambil secara acak dari tabel DistribusiMotor. Atribut tujuan/keputusannya yaitu ***nama_perusahaan***, dimana memiliki dua isi yang berbeda yaitu GELORA FAJAR PERKASA PT dan GEMILANG MAKMUR ABADI CV yang menggambarkan perusahaan yang menjadi target pendistribusiannya. Kita asumsikan bahwa kelas C_1= GELORA FAJAR PERKASA PT dan C_2= GEMILANG MAKMUR ABADI CV, terdapat 19 sampel data dengan kelas C_1 dan 7 sampel data dengan kelas C_2.

Proses klasifikasi dimulai dengan mengecek isi dari atribut kelas data (**nama_perusahaan**) karena isi atribut tersebut berbeda maka digunakan fungsi *entropy* untuk mendapatkan informasi tertinggi (*highest information gain*) dengan perhitungan sebagai berikut:

$I(s_1,s_2)$ = $I(19,7) = -\frac{19}{26}log_2\frac{19}{26} - \frac{7}{26}log_2\frac{7}{26} = 0.84$

1. Atribut tipe_motor

   a. GL200R1

   $I(s_{11},s_{12})$ = $I(1,0) = -\frac{1}{1}log_2\frac{1}{1} - \frac{0}{1}log_2\frac{0}{1} = 0$

   b. NC110CW1

   $I(s_{21},s_{22})$ = $I(1,2) = -\frac{1}{3}log_2\frac{1}{3} - \frac{2}{3}log_2\frac{2}{3} = 0.91$

   c. NC11BICA/T

   $I(s_{31},s_{32})$ = $I(3,1) = -\frac{3}{4}log_2\frac{3}{4} - \frac{1}{4}log_2\frac{1}{4} = 0.81$

   d. NF125TR1

   $I(s_{41},s_{42})$ = $I(4,0) = -\frac{4}{4}log_2\frac{4}{4} - \frac{0}{4}log_2\frac{0}{4} = 0$

   e. NF11BICM/T

   $I(s_{51},s_{52})$ = $I(4,2) = -\frac{4}{6}log_2\frac{4}{6} - \frac{2}{6}log_2\frac{2}{6} = 0.92$

   f. NF11BIDM/T

   $I(s_{61},s_{62})$ = $I(6,2) = -\frac{6}{8}log_2\frac{6}{8} - \frac{2}{8}log_2\frac{2}{8} = 0.81$

E(tipe_motor) = $\frac{1}{26}I(s_{11},s_{12}) + \frac{3}{26}I(s_{21},s_{22}) + \frac{4}{26}I(s_{31},s_{32}) + \frac{4}{26}I(s_{41},s_{42})$

$+\frac{6}{26}I(s_{61},s_{62})\frac{8}{26}I(s_{61},s_{62})$

$$= \frac{1}{26}0 + \frac{3}{26}0.91 + \frac{4}{26}0.81 + \frac{4}{26}0 + \frac{6}{26}0.92\frac{8}{26}0.81$$

$$= 0.691$$

Gain(tipe_motor) = $I(s_1,s_2)$ - *E*(tipe_motor)

= 0.84 – 0.691 = 0.15

Nilai *entropy* untuk atribut **tipe_motor** yang dihasilkan adalah 0.691, yang berarti bahwa tingkat keberagaman untuk nilai dari atirbut **tipe_motor** adalah 0.691. Sedangkan untuk nilai *gain* atau nilai yang efektif untuk mengklasifikasikan atribut **tipe_motor** sebesar 0.15.

2. Atribut kode_warna

a. BE

$I(s_{11},s_{12}) = I(1,0) = 0$

b. WG

$I(s_{21},s_{22}) = I(1,0) = 0$

c. NK

$$I(s_{31},s_{32}) = I(2,1) = -\frac{2}{3}log_2\frac{2}{3} - \frac{1}{3}log_2\frac{1}{3} = 0.92$$

d. BK

$$I(s_{41},s_{42}) = I(3,2) = -\frac{3}{5}log_2\frac{3}{5} - \frac{2}{5}log_2\frac{2}{5} = 0.97$$

e. BS

$I(s_{51},s_{52}) = I(2,0) = 0$

f. BR

$$I(s_{61},s_{62}) = I(2,1) = -\frac{2}{3}log_2\frac{2}{3} - \frac{1}{3}log_2\frac{1}{3} = 0.92$$

g. HJ

$I(s_{71},s_{72}) = I(2,0) = 0$

h. BG

$I(s_{81},s_{82}) = I(2,0) = 0$

i. RD

$I(s_{91},s_{92}) = I(2,1) = -\frac{2}{3}log_2\frac{2}{3} - \frac{1}{3}log_2\frac{1}{3} - 0.92$

j. BB

$I(s_{101},s_{102}) = I(2,1) = -\frac{2}{3}log_2\frac{2}{3} - \frac{1}{3}log_2\frac{1}{3} = 0.92$

k. WP

$I(s_{111},s_{112}) = I(0,1) = 0$

E(kode_warna)

$= 0 + 0 + \frac{3}{26}0.92 + \frac{5}{26}0.97 + 0 + \frac{3}{26}0.92 + 0 + 0 + \frac{3}{26}0.92 + \frac{3}{26}0.92 + 0$

$= 0.61$

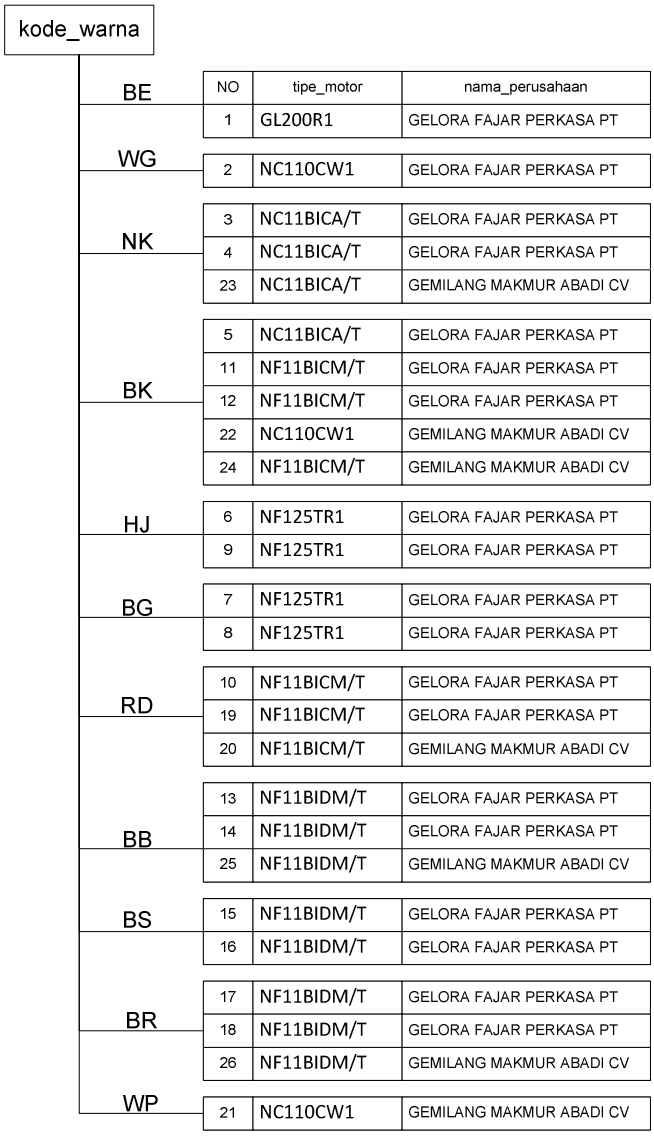
G(kode_warna) = (s_1,s_2) - E(kode_warna)

$= 0.84 - 0.61 = 0.23$

Nilai *entropy* untuk atribut **tipe_motor** yang dihasilkan adalah 0.61, yang berarti bahwa tingkat keberagaman untuk nilai dari atirbut **tipe_motor** adalah 0.691. Sedangkan untuk nilai *gain* atau nilai yang efektif untuk mengklasifikasikan atribut **tipe_motor** sebesar 0.23.

Dari perhitungan diatas tersebut didapat bahwa atribut **kode_warna** mempunyai informasi tertinggi (*gain*) yaitu 0.23 dibandingkan dengan atribut **tipe_motor**, dengan demikian atribut **kode_warna** dipilih sebagai atribut yang

akan diuji. Dimana sebuah node dibuat dengan label **kode_warna** dan cabang dari node tersebut tumbuh berdasarkan isi atribut **kode_warna** dan data dipartisi berdasarkan isi atribut tersebut. Pohon keputusan yang terbentuk dapat ditunjukan pada Gambar 3.1.



	NO	tipe_motor	nama_perusahaan
BE	1	GL200R1	GELORA FAJAR PERKASA PT
WG	2	NC110CW1	GELORA FAJAR PERKASA PT
NK	3	NC11BICA/T	GELORA FAJAR PERKASA PT
	4	NC11BICA/T	GELORA FAJAR PERKASA PT
	23	NC11BICA/T	GEMILANG MAKMUR ABADI CV
BK	5	NC11BICA/T	GELORA FAJAR PERKASA PT
	11	NF11BICM/T	GELORA FAJAR PERKASA PT
	12	NF11BICM/T	GELORA FAJAR PERKASA PT
	22	NC110CW1	GEMILANG MAKMUR ABADI CV
	24	NF11BICM/T	GEMILANG MAKMUR ABADI CV
HJ	6	NF125TR1	GELORA FAJAR PERKASA PT
	9	NF125TR1	GELORA FAJAR PERKASA PT
BG	7	NF125TR1	GELORA FAJAR PERKASA PT
	8	NF125TR1	GELORA FAJAR PERKASA PT
RD	10	NF11BICM/T	GELORA FAJAR PERKASA PT
	19	NF11BICM/T	GELORA FAJAR PERKASA PT
	20	NF11BICM/T	GEMILANG MAKMUR ABADI CV
BB	13	NF11BIDM/T	GELORA FAJAR PERKASA PT
	14	NF11BIDM/T	GELORA FAJAR PERKASA PT
	25	NF11BIDM/T	GEMILANG MAKMUR ABADI CV
BS	15	NF11BIDM/T	GELORA FAJAR PERKASA PT
	16	NF11BIDM/T	GELORA FAJAR PERKASA PT
BR	17	NF11BIDM/T	GELORA FAJAR PERKASA PT
	18	NF11BIDM/T	GELORA FAJAR PERKASA PT
	26	NF11BIDM/T	GEMILANG MAKMUR ABADI CV
WP	21	NC110CW1	GEMILANG MAKMUR ABADI CV

Gambar 3. 1 Proses Klasifikasi Tahap-1

Untuk isi atribut **kode_warna** BE, WG, HJ, BG, BS, dan WP memiliki sampel kelas data yang sama, maka sudah dapat diambil sebagai keputusan seperti digambarkan pada Gambar 3.2.



Gambar 3. 2 Proses Klasifikasi Tahap-2

Dari proses klasifikasi di atas dapat diperoleh informasi bahwa isi atribut **kode_warna** BE, WG, HJ, BG, dan BS ini biasa didistribusikan ke perusahaan GELORA FAJAR PERKASA PT dan **kode_warna** WP ini biasa didstribusikan ke GEMILANG MAKMUR ABADI CV.

Oleh karena atribut penentu yang digunakan hanya dua atribut, yaitu **kode_warna** dan **tipe_motor** maka proses klasifikasi berhenti pada tahap-2, karena sudah tidak atribut lain yang akan diklasifikasikan. Sehingga untuk **kode_warna** NK, BK, RD, BB, and BR tidak terklasifikasi.

3.1.4 Analisis Kebutuhan Non Fungsional

Analisis non fungsional adalah sebuah langkah dimana seorang pembangun perangkat lunak menganalisis sumber daya yang akan menggunakan perangkat lunak yang dibangun.

Analisis non fungsional tidak hanya menganalisis siapa saja yang akan menggunakan aplikasi tetapi juga menganalisis perangkat keras dan perangkat lunak yang dimiliki oleh pemesan, sehingga dapat ditentukan kompabilitas aplikasi yang dibangun terhadap sumber daya yang ada. Setelah melakukan analisis non fungsional, maka dilanjutkan ke langkah berikutnya yaitu menentukan kebutuhan non fungsional sistem yang akan dibangun untuk disesuaikan dengan fakta yang ada.

Apabila terjadi ketidakcocokan antara fakta dan kebutuhan maka perlu adanya penyesuaian fakta terhadap kebutuhan yang ada. Apabila kebutuhan tidak

dipenuhi maka sistem yang dibangun tidak akan berjalan baik sesuai yang diharapkan.

Analisis non fungsional dan kebutuhan non fungsional yang dilakukan dibagi dalam tiga tahap, yaitu:

1. Kebutuhan pengguna (*user*).
2. Kebutuhan perangkat keras (*hardware*).
3. Kebutuhan perangkat lunak (*software*).

3.1.4.1 Analisis Kebutuhan Pengguna (*User*)

Manajer PD. Wijaya Abadi adalah orang biasa menganalisis data-data transaksi, salah satunya transaksi distribusi penjualan sepeda motor yang tersimpan pada *database* perusahaan. Hal tersebut berhubungan dengan aplikasi yang akan dibangun, bahwa pengguna (*user*) aplikasi *data mining* ini adalah Manajer PD. Wijaya Abadi Bandung, dimana keterlibatannya dalam menggunakan aplikasi ini harus mengacu pada spesifikasi pengguna sebagai berikut:

1. Terbiasa menggunakan aplikasi yang ada di system operasi Windows
2. Memiliki pengetahuan mengenai *database*
3. Mengetahui atribut yang dianggap kuat untuk dilibatkan dalam proses *data mining*

Spesifikasi pengguna aplikasi dapat pula dijelaskan pada Tabel 3.7.

Tabel 3. 7 Spesifikasi Pengguna (*User*)

Pengguna	Hak Akses	Pendidikan	Tingkat Keterampilan	Pengalaman
Manajer Perusahaan	Melakukan aksi proses *data mining*	Minimal Diploma 3	Bisa mengikuti petunjuk yang ada pada sistem	Pelatihan pengguna

Dengan demikian, Manajer PD. Wijaya Abadi sudah termasuk pada kualifikasi kebutuhan pengguna aplikasi *data mining* ini.

3.1.4.2 Analisis Kebutuhan Perangkat Keras (*Hardware*)

Dari hasil pengamatan di perusahaan, penggunaan komputer oleh manajer PD. Wijaya Abadi berjumlah satu unit komputer. Berdasarkan analisis yang telah dilakukan, dapat diketahui spesifikasi perangkat keras yang digunakan di PD. Wijaya Abadi saat ini adalah sebagai berikut:

a. *Processor* 2 GHz

b. *Harddisk* 80 GB

c. RAM 1 GB

d. Monitor 17 inch dengan resolusi minimal 1024 X 768 *pixels*

Kebutuhan *hardware* yang disarankan untuk menjalankan aplikasi ini adalah sebagai berikut :

a. *Processor* 1,5 GHz

b. *Harddisk* 20 GB

c. RAM 512 MB

d. Monitor yang mendukung kualitas warna 16 bit dengan resolusi 800 X 600 *pixels*

Berdasakan spesifikasi yang telah ada, secara keseluruhan kebutuhan perangkat keras untuk aplikasi ini telah terpenuhi.

3.1.4.3 Analisis Kebutuhan Perangkat Lunak (*Software*)

Perangkat lunak merupakan sarana pendukung lainnya bagi pembangunan aplikasi *data mining* ini. Adapun perangkat lunak yang terpasang di komputer manajer PD. Wijaya Abadi saat ini secara umum adalah sebagai berikut:

a. Sistem Operasi Windows XP
b. Microsoft Office

Sedangkan perangkat lunak yang disarankan untuk menjalankan aplikasi ini adalah sebagai berikut:

a. Sistem Operasi

   Untuk sistem operasi windows disarankan, karena sistem operasi ini banyak dikenal oleh *user* awam dan lebih mudah untuk dipelajari.

b. Microsoft Office
c. *Database Management System*

   Database Management System (DBMS) adalah sistem untuk mengelola basisdata yang digunakan. Untuk aplikasi *data mining* ini digunakan Borland Interbase sebagai DBMS.

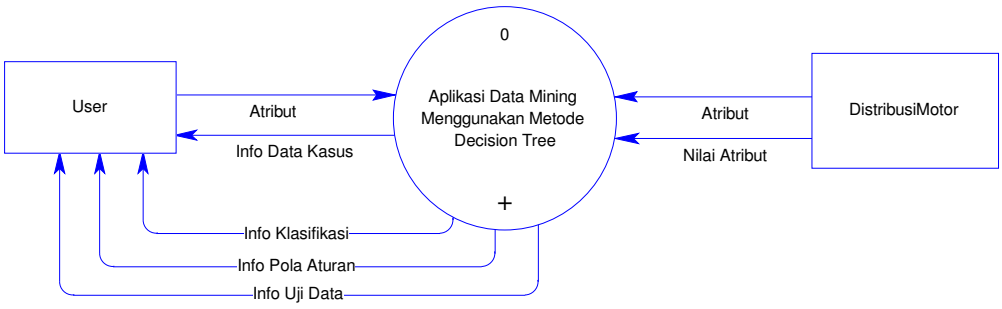
Dengan demikian, perusahaan disarankan menginstal DBMS yang sesuai dengan kebutuhan untuk menjalankan aplikasi *data mining* ini.

3.1.5 Analisis Kebutuhan Fungsional

Setelah melakukan analisis kebutuhan non fungsional maka dilanjutkan ke langkah berikutnya yaitu analisis kebutuhan fungsional. Dalam langkah ini dilakukan penentuan entitas-entitas baik entitas internal maupun entitas eksternal, data yang mengalir, serta prosedur-prosedur yang bisa dilakukan oleh masing-masing entitas.

3.1.5.1 Diagram Konteks

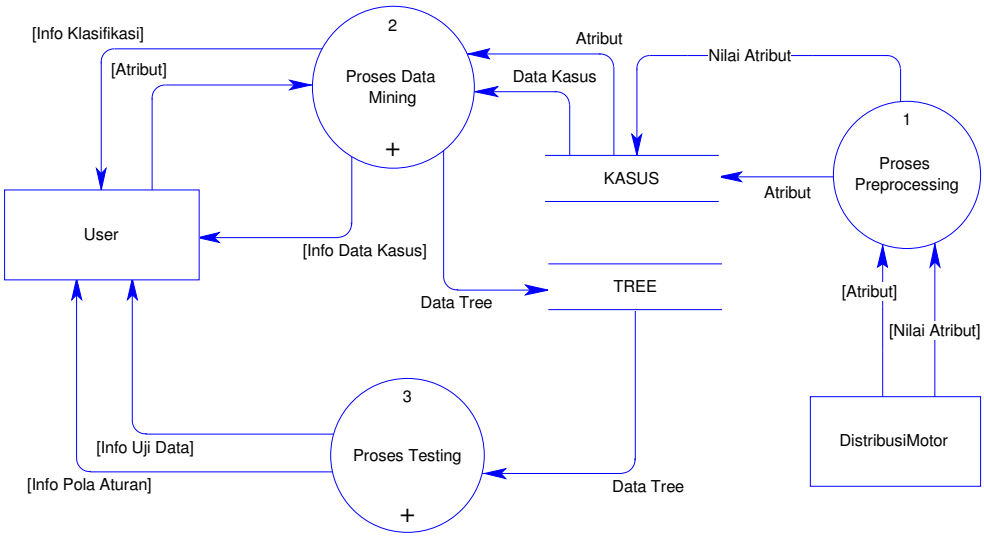
Diagram konteks merupakan salah satu alat bantu dalam melakukan analisis terstruktur. Diagram konteks ini menggambarkan suatu sistem secara garis besarnya atau keseluruhannya saja. Dalam Diagram konteks juga digambarkan entitas eksternal yang merupakan *brainware* yang menghasilkan data yang akan diolah oleh sistem maupun tujuan dari informasi yang dihasilkan oleh sistem. Dikarenakan tidak cukupnya penggambaran dalam satu halaman maka diagram konteks dibagi menjadi dua bagian (dalam kenyataannya tetap dilihat sebagai satu diagram konteks). Adapun diagram konteks yang diusulkan ditunjukan oleh Gambar 3.3.



Gambar 3. 3 Diagram Konteks Aplikasi *Data Mining*

3.1.5.2 Diagram Alir Data Level 1

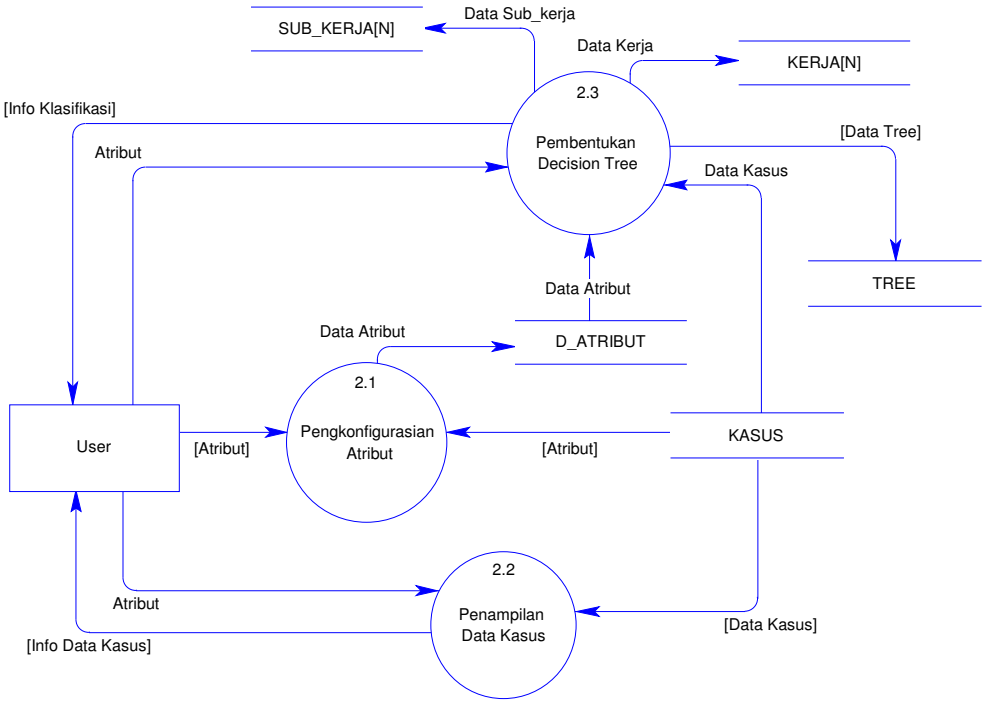
Diagram alir data merupakan sebuah representasi dari suatu sistem yang menggambarkan bagian-bagian dari sistem tersebut beserta keterkaitan antara bagian-bagian yang ada. Dari diagram alir data ini seseorang bisa mengetahui sumber dari informasi di dalam sistem maupun tujuan dari masukan yang berasal dari entitas eksternal. Adapun diagram alir data level 1 dari aplikasi *Data Mining* ditunjukan oleh Gambar 3.4.



Gambar 3. 4 DFD Level 1 Aplikasi *Data Mining*

3.1.5.3 Diagram Alir Data Level 2 Proses 2 Proses Data Mining

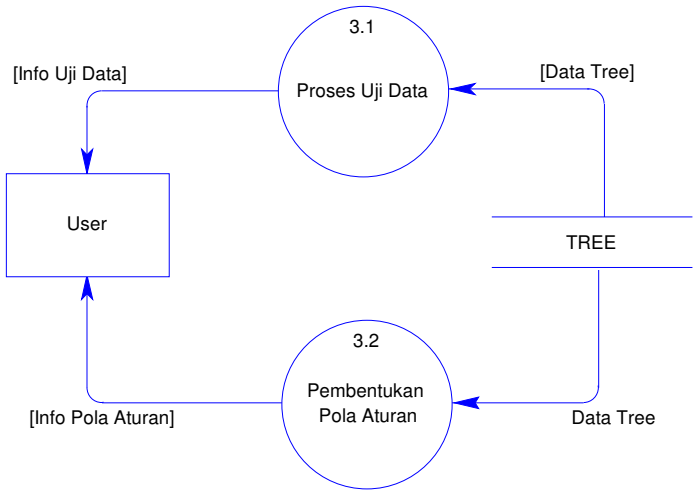
Gambar 3.5 menunjukan diagram alir data pada proses Data Mining.



Gambar 3. 5 DFD Level 2 Proses *Data Mining*

3.1.5.4 Diagram Alir Data Level 2 Proses 3 Proses Testing

Gambar 3.6 menunjukan diagram alir data pada proses Testing.



Gambar 3. 6 DFD Level 2 Proses Testing

3.1.5.5 Spesifikasi Proses

Proses-proses yang terdapat pada Diagram Alir Data akan dijelaskan lebih terperinci pada spesifikasi proses, sebagaimana diperlihatkan pada Tabel 3.8.

Tabel 3. 8 Spesifikasi Proses

No.	Proses	Keterangan
1	No. Proses	1
	Nama Proses	Proses *Preprocessing*
	Sumber	DistribusiMotor
	Input	Atribut, Nilai Atribut
	Output	Atribut, Nilai Atribut
	Deskripsi	Melakukan proses *cleaning* terhadap atribut yang akan digunakan untuk proses *data mining*
	Logika Proses	Dalam proses *Preprocessing* meliputi proses: 1. Data Selection 2. Data Cleaning 3. Data Transformation 4. Data Reduction
2	No. Proses	2
	Nama Proses	Proses Data Mining

	Sumber	User, Tabel KASUS
	Input	Aksi, Atribut, Data Kasus
	Output	*Decision Tree*
	Deskripsi	Melakukan penggalian data (*data mining*) dengan melakukan proses klasifikasi
	Logika Proses	1. *User* memilih menu Data Mining 2. *User* memilih submenu
3	No. Proses	3
	Nama Proses	Proses Testing
	Sumber	User, Tabel TREE
	Input	Data Testing, Data Tree
	Output	Pola Aturan
	Deskripsi	Melakukan proses pengujian terhadap data terhadap pola aturan yang sudah terbentuk
	Logika Proses	1. *User* memilih menu Testing 2. *User* memilih submenu
4	No. Proses	2.1
	Nama Proses	Pengkonfigurasian Atribut
	Sumber	User, Tabel KASUS
	Input	Aksi, Atribut
	Output	Data Atribut
	Deskripsi	Melakukan pengaturan terhadap atribut yang akan digunakan sebagai atribut penentu dan atribut tujuan

	Logika Proses	`procedure TFrmSettingAtribut.BtnSimpanClick(Sender: TObject);`
		`  var i : integer;`
		`  is_salah : boolean;`
		`begin`
		`  if EdtAtribut.Text = '' then`
		`  begin`
		`    MessageDlg('Pilih nama atributnya!', mtInformation, [mbOK],0);`
		`    CbAktif.SetFocus;`
		`    Exit;`
		`  end;`
		`  with DM.Q1 do`
		`  begin`
		`      SQL.Text :=`
		`        ' update d_atribut set '+`
		`        ' is_aktif = :is_aktif '+`
		`        ' where nama_atribut = :namalama ';`
		`      ParamByName('namalama').AsString := LVAtribut.Selected.Caption;`
		`    if CbAktif.Checked = true then ParamByName('is_aktif').AsString := 'Y'`
		`    else ParamByName('is_aktif').AsString := 'T';`
		`    try`
		`      ExecSQL;`
		`    except`
		`    end;`
		
		`    try`
		`      IF DM.IBT.Active = False THEN`
		`        DM.IBT.Active := True;`
		`      DM.IBT.Commit;`

<table>
<tr><td></td><td></td><td><pre>
    except
    end;
  end;
  //buat tabel kasus
  tampil;
end;
</pre></td></tr>
<tr><td rowspan="6">5</td><td>No. Proses</td><td>2.2</td></tr>
<tr><td>Nama Proses</td><td>Penampilan Data Kasus</td></tr>
<tr><td>Sumber</td><td>Tabel KASUS</td></tr>
<tr><td>Input</td><td>Data Kasus</td></tr>
<tr><td>Output</td><td>Data Kasus</td></tr>
<tr><td>Deskripsi</td><td>Menampilkan item data dari data kasus yang akan diproses data mining sesuai atribut yang telah dikonfigurasi</td></tr>
<tr><td>Logika Proses</td><td><pre>
procedure TFrmKasus.FormShow(Sender: TObject);
var i : integer;
begin
  DBGKasus.Columns.Clear;
  DM.IBTable1.TableName := 'Kasus';
  DM.IBTable1.Active := false;
  DM.IBTable1.Active := true;
  DM.DS.DataSet := DM.IBTable1;
  DBGKasus.DataSource := DM.DS;
  //ambil jumlah atribut aktif
  with DM.Q1 do
  begin
    Close;
    SQL.Text := 'SELECT NAMA_ATRIBUT FROM
D_ATRIBUT WHERE IS_AKTIF = ''Y''';
    Open;
    i := 0;
    if IsEmpty then
    begin
      Close;
      SQL.Text := 'DELETE FROM kasus';
      ExecSQL;
    end;
    while not Eof do
    begin
      DBGKasus.Columns.Add;
      DBGKasus.Columns[i].FieldName :=
Fields[0].AsString;
      DBGKasus.Columns[i].Width := 120;
      Next;
      inc(i);
    end;
  end;
end;
</pre></td></tr>
<tr><td rowspan="6">6</td><td>No. Proses</td><td>2.3</td></tr>
<tr><td>Nama Proses</td><td>Pembentukan Decision Tree</td></tr>
<tr><td>Sumber</td><td>Tabel KASUS, Tabel D_ATRIBUT</td></tr>
<tr><td>Input</td><td>Data Kasus, Data Atribut</td></tr>
<tr><td>Output</td><td>Decision Tree</td></tr>
<tr><td>Deskripsi</td><td>Melakukan penggalian data (data mining) dengan melakukan proses klasifikasi berupa pohon keputusan (decision tree)</td></tr>
<tr><td>Logika Proses</td><td>Algorithm: Generate_decision_tree.
Narative : Generate a decision tree from the given training data.</td></tr>
</table>

		Input: The training samples, *samples*, represented by discrete-valued attribute; the
		set of candidate attributes, *attribute-list*.
		Output: A decision tree.
		Method:
		1. create a node *N*;
		2. if *samples* are all of the same class, C **then**
		3. return *N* as a leaf node labeled with the class C;
		4. if *attribute-list* is empty **then**
		5. return *N* as a leaf node labeled with the most common class in *samples*;//majority voting
		6. select *test-attribute*, the attribute among *attribute-list* with the highest information gain;
		7. label node N with *test-attribute*;
		8. **for each** known value *ai* of *test-attribute;*
		9. grow a branch from node N for the condition *test-attribute = ai;*
		10. let *si* be the set of samples in *samples* for which *test-attribute = ai;* // a partition
		11. if *si* is empty **then**
		12. attach a leaf labeled with the most common class in *samples*;
		13. **else** attach the node returned by Generate_decision_tree (*si, attribute-listtest-attribute);*

7	No. Proses	3.1
	Nama Proses	Proses Uji Data
	Sumber	User, Tabel TREE
	Input	Data Testing, Data Tree
	Output	Info Testing
	Deskripsi	Melakukan pengujian data terhadap pola aturan yang sudah terbentuk

	Logika Proses	`begin`
		`  if BtnNext.Caption = '&Mulai' then`
		`  begin`
		`  LVAtribut.Items.Clear;`
		`  with DM.Q1 do`
		`  begin`
		`  //cari induk`
		`  Close;`
		`  SQL.Text := 'select id_node, node from tree where induk is null';`
		`  Open;`
		`  EdtAtribut.Text := Fields[1].AsString;`
		`  id_node := Fields[0].AsInteger;`
		`  Cari_Nilai(id_node);`
		`  BtnNext.Enabled := true;`
		`  LblHasil.Caption := '';`
		`  BtnNext.Caption := '&Selanjutnya'`
		`  end;`
		`  end else`
		`  begin`
		`  with DM.Q1, daftar do`
		`  begin`
		`  daftar := LVAtribut.Items.Add;`
		`  Caption := EdtAtribut.Text;`

		SubItems.Add(CmbNilai.Items[CmbNilai.Itemindex]);
		close;
		SQL.Text := 'select id_node, node, is_atribut '+
		' from tree '+
		' where induk = :induk '+
		' and nilai = :nilai '+
		' ORDER BY ID_NODE ';
		ParamByName('induk').AsInteger := id_node;
		ParamByName('nilai').AsString := Nilai_Combo[CmbNilai.itemindex];
		Open;
		if Fields[2].AsString = 'Y' then
		begin
		EdtAtribut.Text := Fields[1].AsString;
		id_node := Fields[0].AsInteger;
		Cari_Nilai(id_node);
		end else
		begin
		EdtAtribut.Text := '';
		CmbNilai.Items.Clear;
		CmbNilai.Text := '';
		LblHasil.Caption := Fields[1].AsString;
		BtnNext.Caption := '&Mulai';
		end;
		end;
		end;

8	No. Proses	3.2
	Nama Proses	Pembentukan Pola Aturan
	Sumber	Tabel TREE
	Input	Data Tree
	Output	Pola Aturan
	Deskripsi	Menampilkan pola aturan yang sudah terklasifikasi

	Logika Proses	var noaturan, level : integer;
		daftar : TListItem;
		QTemp : TIBQuery;
		procedure Tampil (id_node : integer; nilai : string);
		begin
		inc (level);
		try
		with TIBQuery.Create(self) do
		begin
		Name := 'TIQ'+ IntToStr(Level);
		Database := DM.DB;
		Transaction := DM.IBT;
		end;
		except
		end;
		QTemp := FindComponent('TIQ'+ IntToStr(Level)) as TIBQuery;
		with daftar do
		begin
		QTemp.Close;
		QTemp.SQL.Text := 'SELECT node, induk, nilai '+
		' FROM TREE '+
		' WHERE ID_NODE = :id '+
		' ORDER BY ID_NODE';
		QTemp.ParamByName('id').AsInteger := id_node;

		QTemp.Open;
		Tampil(QTemp.Fields[1].AsInteger,
		QTemp.Fields[2].AsString);
		dec (level);
		QTemp := FindComponent('TIQ'+
		IntToStr(Level)) as TIBQuery;
		daftar := LVAturan.Items.Add;
		Caption := '';
		(FindComponent('TIQ'+ IntToStr(Level)) as
		TIBQuery).Free;
		end;
		end;
		
		begin
		LVAturan.Items.Clear;
		noaturan := 1;
		//cari induk
		with DM.Q1, daftar do
		begin
		Close;
		SQL.Text := 'SELECT ID_NODE, NODE, INDUK,
		nilai '+
		' FROM TREE '+
		' WHERE IS_ATRIBUT = ''T''' +
		' ORDER BY ID_NODE';
		Open;
		while not Eof do
		begin
		//tampilkan dalam listview
		daftar := LVAturan.Items.Add;
		Caption:= '';
		SubItems.Add('');
		level := 0;
		if Fields[2].AsString <> '' then
		Tampil(Fields[2].AsInteger,
		Fields[3].AsString);
		daftar := LVAturan.Items.Add;
		Caption := '';
		SubItems.Add('MAKA ' + Fields[1].AsString);
		Application.ProcessMessages;
		Inc(noaturan);
		Next;
		end;
		end;

3.1.5.6 Kamus Data

Kamus data dapat mendefinisikan dengan lengkap yang mengalir diantara proses, menyimpan data, dan entitas. Data yang mengalir tersebut dapat berupa masukan untuk sistem atau hasil dari proses sistem. Kamus data dibuat berdasarkan arus data yang mengalir pada Konteks Diagram dan Data Alir Diagram, sebagaimana diperlihatkan pada Tabel 3.9.

Tabel 3. 9 Kamus Data

Nama Aliran data	Data Atribut
Where used/ how used	*User* - proses 1 *User* - proses 1.1 *User* - proses 1.3 Proses 1 - file D_ATRIBUT Proses 1.1 - file D_ATRIBUT Proses 1.3 - file D_ATRIBUT
Deskripsi	Berisi data berupa nama_atribut yang diperoleh dari data kasus
Struktur data	nama_atribut+is_aktif+is_hasil+ket
nama_atribut is_aktif is_hasil ket	[A-Z|a-z|0-9] [A-Z|a-z|0-9] [A-Z|a-z|0-9] [A-Z|a-z|0-9]
Nama Aliran data	Data Kasus
Where used/ how used	*User* - proses 1 *User* - proses 1.1 *User* - proses 1.2 *User* - proses 1.3 Proses 1 - *file* KASUS Proses 1.1 - *file* KASUS Proses 1.2 - *file* KASUS Proses 1.3 - *file* KASUS
Deskripsi	Berisi data hasil *preprocessing* dari data distribusi motor
Struktur data	nomor_mesin+tipe_motor+kode_warna+tgl_distribusi+no_distribusi+tgl_penjualan+nama_perusahaan
nomor_mesin tipe_motor kode_warna tgl_distribusi no_distribusi tgl_penjualan nama_perusahaan	[A-Z|a-z|0-9] [A-Z|a-z|0-9] [A-Z|a-z|0-9] [0-9] [A-Z|a-z|0-9] [0-9] [A-Z|a-z|0-9]
Nama Aliran data	Data Sub_kerja
Where used/ how used	*User* - proses 1 *User* - proses 1.3 Proses 1 - *file* SUB_KERJA Proses 1.3 - *file* SUB_KERJA
Deskripsi	Berisi data sub_kerja yang merupakan tempat penyimpanan hasil perhitungan *entropy*
Struktur data	nama_atribut+nilai+entropy+result1+result2+jml_kasus
nama_atribut nilai entropy	[A-Z|a-z|0-9] [A-Z|a-z|0-9] [0-9]

result1 result2 jml_kasus	[A-Z|a-z|0-9] [A-Z|a-z|0-9] [0-9]
Nama Aliran data	Data Kerja
Where used/ how used	*User* - proses 1 *User* - proses 1.3 Proses 1 - *file* KERJA Proses 1.3 - *file* KERJA
Deskripsi	Berisi data kerja yang merupakan tempat penyimpanan hasil perhitungan *gain*
Struktur data	nama_atribut+gain
nama_atribut gain	[A-Z|a-z|0-9] [0-9]
Nama Aliran data	Data Tree
Where used/ how used	*User* - proses 1 *User* - proses 1.3 *User* - proses 2 *User* - proses 2.1 *User* - proses 2.2 Proses 1 - file TREE Proses 1.3 - file TREE Proses 2 - file TREE Proses 2.1 - file TREE Proses 2.2 - file TREE
Deskripsi	Berisi data kerja yang merupakan tempat penyimpanan hasil perhitungan gain
Struktur data	id_node+node+nilai+induk+is_atribut
id_node node nilai induk is_atribut	[0-9] [A-Z|a-z|0-9] [A-Z|a-z|0-9] [A-Z|a-z|0-9] [A-Z|a-z|0-9]

3.2 Perancangan Sistem

Perancangan merupakan bagian dari metodologi pembangunan suatu perangkat lunak yang harus dilakukan setelah melalui tahapan analisis. Pada bagian ini akan dijelaskan perancangan sistem yang dimaksudkan untuk menggambarkan perbedaan antara sistem yang sedang berjalan dengan sistem

yang diusulkan. Perancangan sistem ini menggunakan pendekatan sistem terkomputerisasi.

3.2.1 Perancangan Data

Perancangan data atau lebih dikenal dengan perancangan basis data yaitu menciptakan atau merancang data yang terhubung dan disimpan secara bersama-sama. Untuk menggambarkannya digunakanlah struktur *file*.

3.2.1.1 Struktur Tabel

Pada proses operasi yang dilakukan aplikasi *data mining* ini, dibutuhkan suatu *database* untuk penymipanan data hasil proses. Adapun struktur tabel digunakan untuk menggambarkan tabel-tabel beserta field-field yang digunakan secara lebih detail, dimana tipe data dan ukuran suatu field serta kunci-kunci (*primary key* atau *foreign key*) disajikan.

Tabel 3.10 memperlihatkan tabel D_ATRIBUT sebagai tempat penyimpanan data distribusi motor.

Tabel 3. 10 Tabel D_ATRIBUT

Nama Field	Tipe	Panjang	Kunci
nama_atribut	varchar	30	*Primary Key*
is_aktif	char	1	
is_hasil	char	1	
ket	varchar	255	

Tabel 3.11 memperlihatkan tabel KASUS sebagai tempat penyimpanan data distribusi motor.

Tabel 3. 11 Tabel KASUS

Nama Field	Tipe	Panjang	Kunci
nomor_mesin	varchar	13	*Primary Key*
tipe_motor	integer	15	
kode_warna	integer	2	
nama_perusahaan	integer	5	

Tabel 3.12 memperlihatkan tabel SUB_KERJA sebagai tempat penyimpanan data hasil perhitungan *entropy*.

Tabel 3. 12 Tabel SUB_KERJA

Nama Field	Tipe	Panjang	Kunci
nama_atribut	varchar	30	
nilai	varchar	255	
entropy	numeric	15,2	
result1	varchar	30	
result1	varchar	30	
jml_kasus	integer		

Tabel 3.13 memperlihatkan tabel KERJA sebagai tempat penyimpanan data hasil perhitungan *gain*.

Tabel 3. 13 Tabel KERJA

Nama Field	Tipe	Panjang	Kunci
nama_atribut	varchar	30	
gain	float	15,2	

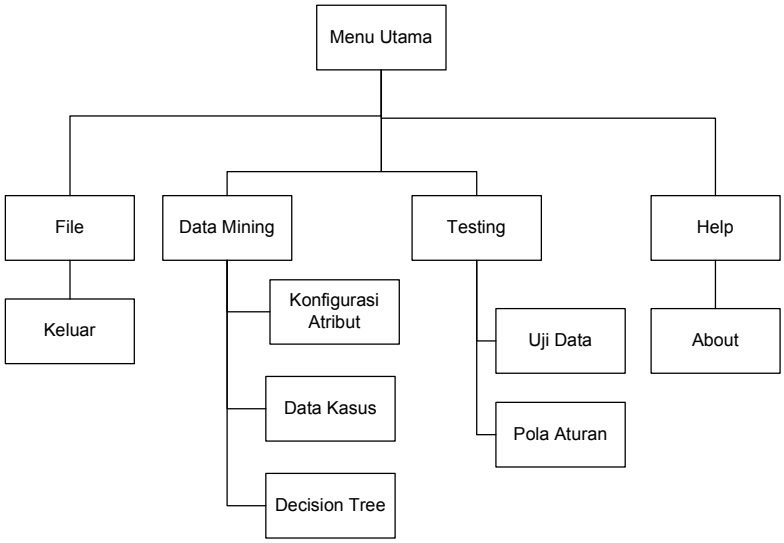
Tabel 3.14 memperlihatkan tabel TREE sebagai tempat penyimpanan data hasil perhitungan *gain*.

Tabel 3. 14 Tabel TREE

Nama Field	Tipe	Panjang	Kunci
id_node	integer		
node	varchar	30	
nilai	varchar	30	
induk	varchar	30	
is_atribut	char	1	

3.2.2 Arsitektur Menu

Berikut ini adalah perancangan struktur menu yang akan diterapkan pada program dan dirancang seefisien mungkin untuk mengurangi kompleksitas sistem. Struktur menu yang digunakan yaitu struktur menu web yang dapat dilihat pada Gambar 3.7.



Gambar 3. 7 Struktur Menu Aplikasi *Data Mining*

3.2.3 Perancangan Antarmuka

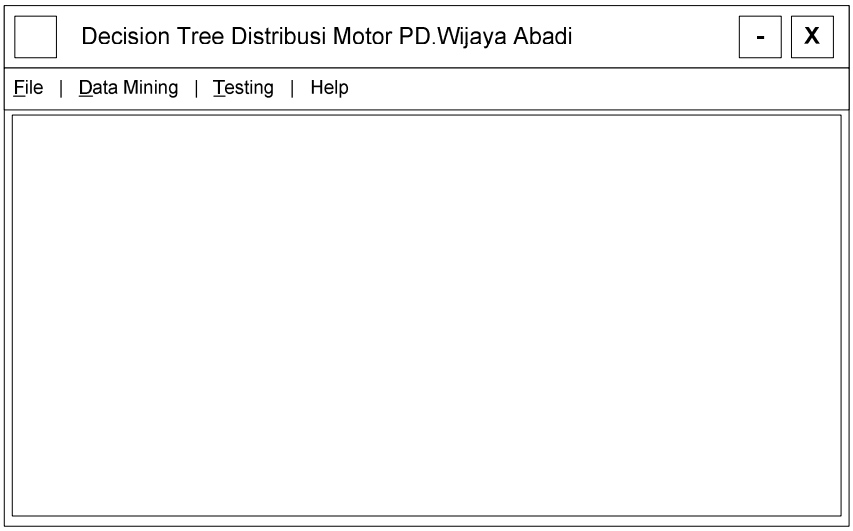
Perancangan antar muka merupakan suatu bentuk tampilan dari program yang akan dibuat untuk kebutuhan *interface* dengan *user.* Spesifikasi antarmuka terdiri dari perancangan tampilan menu, tampilan form, tampilan pesan, keluaran dan jaringan semantik.

3.2.3.1 Perancangan Form

Dalam perancangan *form* di sini dilakukan perancangan baik terhadap *form input* dan *form output*. Adapun *form-form* yang terdapat dalam Aplikasi *Data Mining* ini adalah sebagai berikut:

3.2.3.1.1 Perancangan *Form* Menu Utama

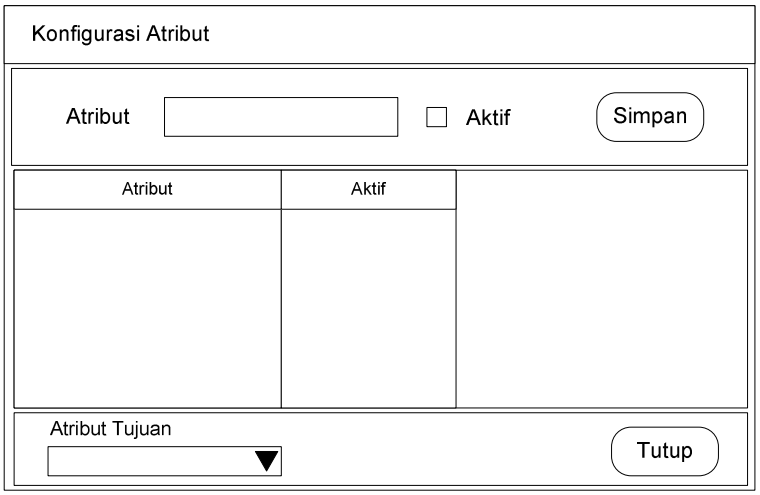
Gambar 3.8 memperlihatkan perancangan halaman utama aplikasi.



Gambar 3. 8 *Form* Menu Utama Aplikasi *Data Mining*

3.2.3.1.1 Perancangan *Form* Konfigurasi Atribut

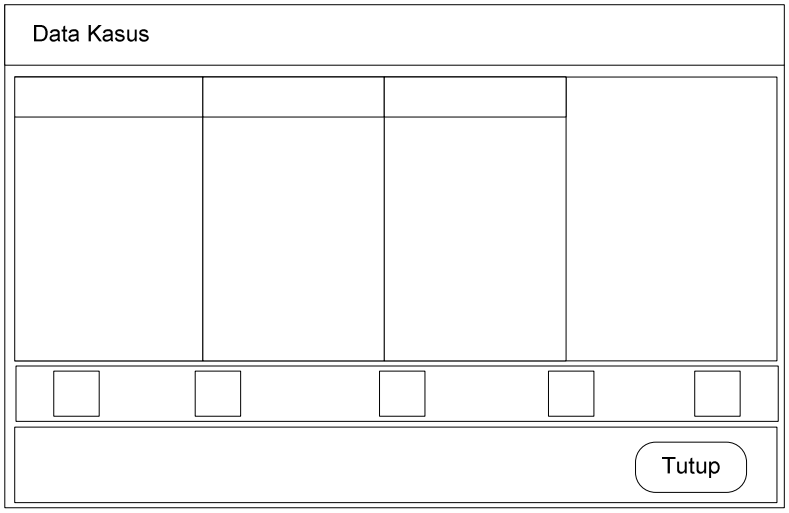
Gambar 3.9 memperlihatkan perancangan *form* untuk pengaturan atribut yang akan digunakan dalam proses *data mining*.



Gambar 3. 9 *Form* Konfigurasi Atribut

3.2.3.1.2 Perancangan *Form* Data Kasus

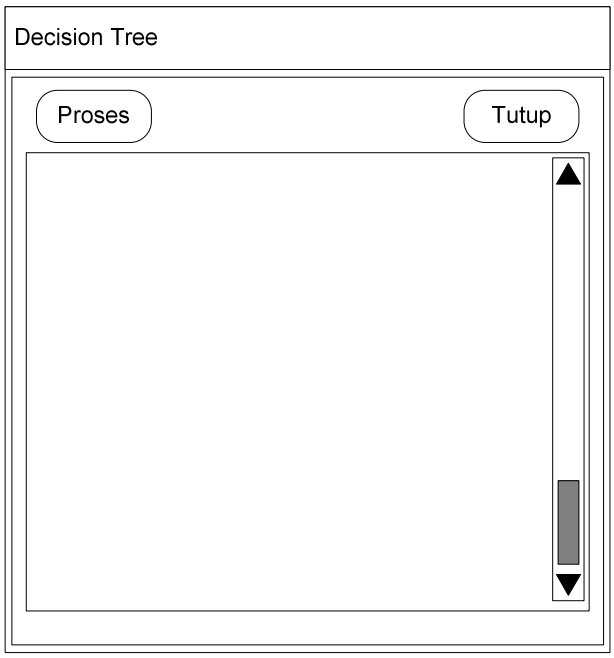
Gambar 3.10 memperlihatkan perancangan *form* yang hanya menampilkan sekumpulan data yang siap untuk di-mining.



Gambar 3. 10 *Form* Data Kasus

3.2.3.1.3 Perancangan *Form Decision Tree*

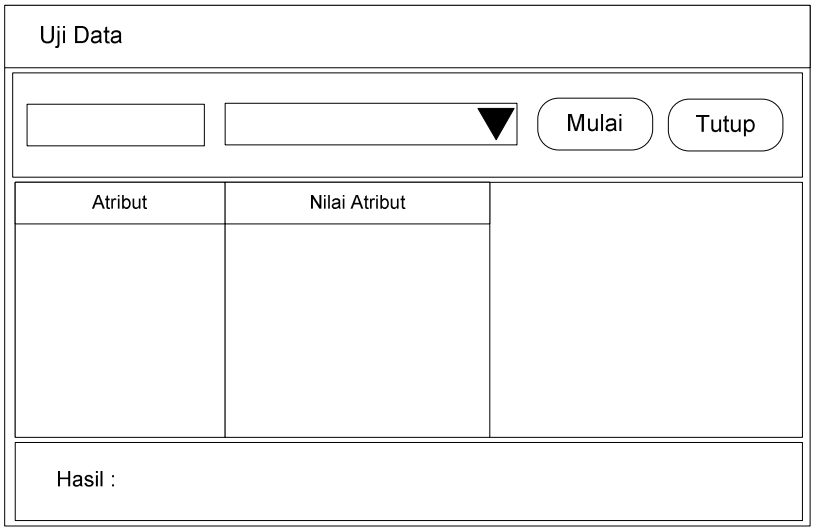
Gambar 3.11 memperlihatkan perancangan *form* yang akan memproses data hingga menghasilkan informasi klasifikasi dalam bentuk pohon keputusan (*decision tree*).



Gambar 3. 11 *Form Decision Tree*

3.2.3.1.4 Perancangan *Form* Uji Data

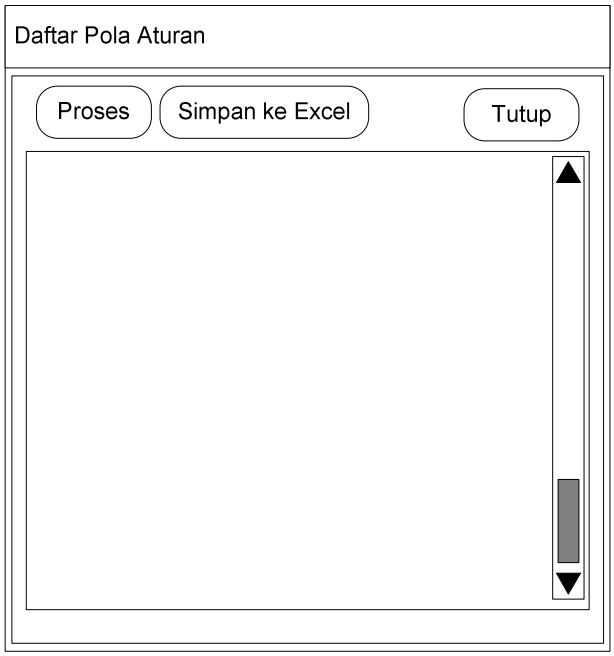
Gambar 3.12 memperlihatkan perancangan *form* sebagai media untuk menguji nilai atribut (data) yang sudah terklasifikasi.



Gambar 3. 12 *Form* Uji Data

3.2.3.1.5 Perancangan *Form* Pola Aturan

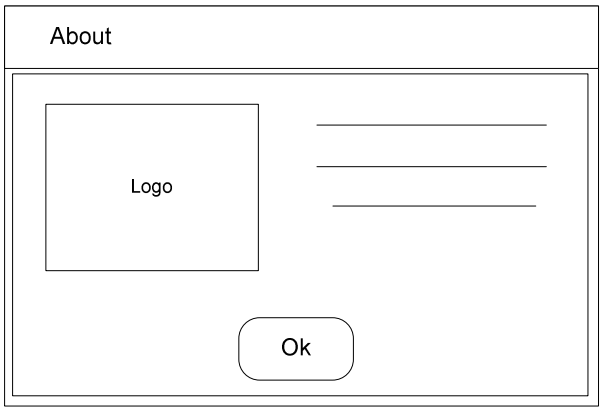
Gambar 3.13 memperlihatkan perancangan *form* untuk melakukan proses menmpilkan daftar pola aturan pendistribusian sepeda motor.



Gambar 3. 13 *Form* Pola Aturan

3.2.3.1.6 Peracangan *Form About*

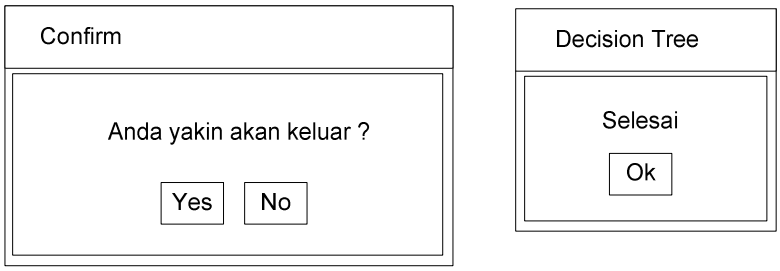
Gambar 3.14 memperlihatkan perancangan *form* yang hanya menampilkan sedikit informasi mengenai aplikasi.



Gambar 3. 14 *Form About*

3.2.3.1.7 Perancangan Pesan

Pada gambar 3.5 di bawah ini merupakan perancangan pesan yang terdapat pada Aplikasi *Data Mining*.



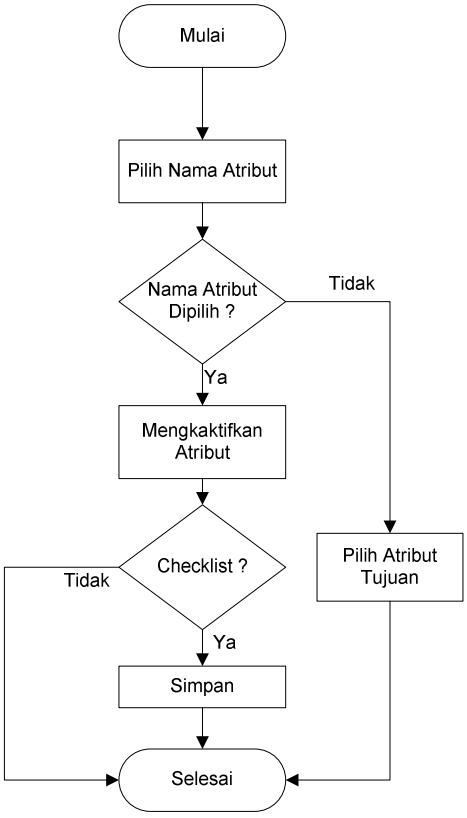
Gambar 3. 15 *Form* Pesan

3.2.4 Perancangan Prosedural

Sebagai langkah terakhir dalam perancangan yaitu merancang prosedural yang akan diimplementasikan ke dalam sistem. Prosedural ini akan digunakan

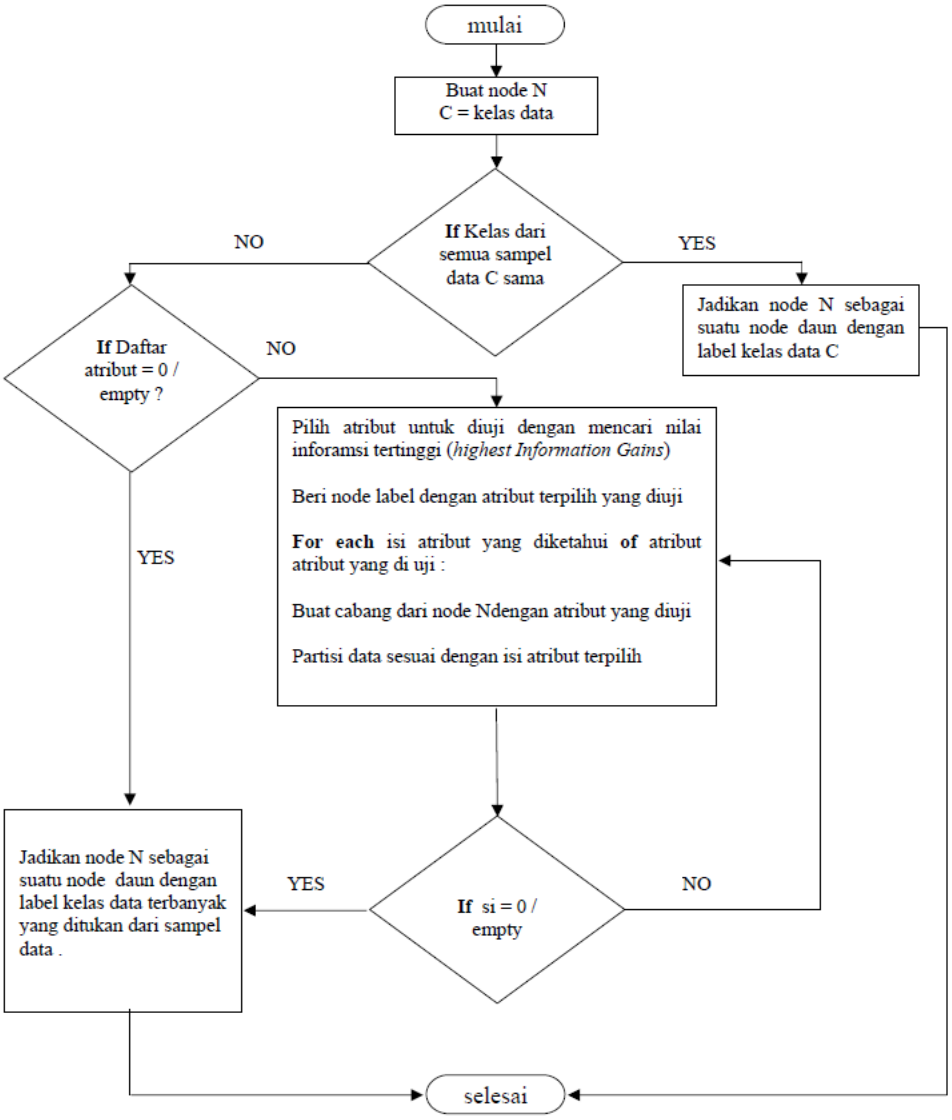
sebagai algoritma dasar dalam mengkodekan prosedur yang ada. Adapun perancangan prosedural aplikasi *Data Mining* adalah sebagai berikut:

3.2.4.1 Prosedural Konfigurasi Atribut



Gambar 3. 16 Diagram Alir Prosedur Konfigurasi Atribut

3.2.4.2 Prosedural *Decision Tree*



Gambar 3. 17 Diagram Alir Prosedur *Decision tree*