BAB 3

ANALISIS DAN PERANCANGAN SISTEM

3.1 Analisis Sistem

Analisis Sistem dapat didefinisikan sebagai penguraian dari suatu sistem informasi yang utuh ke dalam bagian-bagian komponennya dengan maksud untukmengidentifikasikan dan mengevaluasi permasalahan-permasalahan, kesempatan-kesempatan, hambatan-hambatanyang terjadi dan kebutuhan-kebutuhan yang diharapkan sehingga dapat diusulkan perbaikan-perbaikan.Atau secara lebih mudahnya, analisis sistem adalah penelitian atas sistem yang telah ada dengan tujuan untukmerancang sistem yang baru atau diperbarui. Tahap analisis sistem ini merupakan tahap yang sangat kritis dan sangatpenting, karena kesalahan di dalam tahap ini akan menyebabkan juga kesalahan di tahap selanjutnya.Tugas utama analis sistem dalam tahap ini adalah menemukan kelemahan-kelemahan dari sistem yang berjalansehingga dapat diusulkan perbaikannya.

3.1.1 Analisis Masalah

Saham jenis *common stock* dikatakan memiliki harga yang cenderung fluktuatif.Harga saham tersebut dipengaruhi oleh banyak factor, dimana menurut analisis fundamental terdiri dari tiga macam, yaitu faktor ekonomi, faktor industry, dan faktor perusahaan.

Permasalahan didalam sistem yang berjalan di BEI adalah aplikasi hanya menyediakan fasilitas pembelian dan penjualan saham saja, padahal kebutuhan

investor sebagai *end user dari* aplikasi di BEI ini membutuhkan analisa untuk menentukan penjualan atau pembelian pada aplikasi karena setiap keputusan yang dibuat oleh investor mengandung resiko yang tinggi.

Para Pelaku Ekonomi didunia sahammendapatkankesulitan tersendiri dalam menganalisa faktor-faktor tersebut, terlebih apabila disaat jumlah data yang dianalisa sangat besar membuat seringkali menemukan kesulitan dalam proses analisa. Padahal analisa yang tepat akan menghasilkan keputusan yang tepat sehingga dapat memaksimalkan keuntungan yang didapat dari saham. Permasalahan ini adalah Dalam menganalisa data masih dilakukan berdasarkan insting penganalisa dan menggunakan manual sehingga dalam menganalisa data dalam jumlah besar membuat tidak efisiensi dan efektif bahkan sering tidak tepat dan tidak sesuai harapan para investor.

3.1.2 Analisis *Non*Fungsional dan Kebutuhan *Non* Fungsional

Analisis *non* fungsional dilakukan untuk menghasilkan spesifikasi kebutuhan *non* fungsional. Spesifikasi kebutuhan *non* fungsional adalah spesifikasi yang rinci tentang hal-hal yang akan dilakukan sistem ketika diimplementasikan. Analisis kebutuhan ini diperlukan untuk menentukan keluaran yang akan dihasilkan sistem, masukan yang diperlukan sistem, lingkup proses yang digunakan untuk mengolah menjadi keluaran, *volume* data yang akan ditangani sistem, jumlah pemakai dan kategori pemakai, serta kontrol terhadap sistem.

3.1.2.1. Analisis Perangkat Keras

Berdasarkan hasil penelitian user yang ada baik investor atau pun fund manager di BEI memiliki spesifikasi Komputer/ Laptop dengan spesifikasi diatas standart yang dibutuhkan oleh aplikasi ini, adapun spesifikasi perangkat keras yang dibutuhkan untuk menjalankan aplikasi ini adalah sebagai berikut :

a. *Processor* : Intel Pentium 4
b. *Harddisk* : 80 Gb
c. *Memory* : 512MB
d. *Monitor* : 14”
e. *Printer* : Color
f. *ROM* : DVD RW
g. *Keyboard* : *Standard*
h. *Mouse* : *Optical Mouse*

3.1.2.2. Analisis Perangkat Lunak

Kebutuhan akan perangkat lunak pendukung sangatlah penting bagi lahirnya perangkat lunak yang sedang dirancang. Adapun kebutuhan spesifikasi perangkat lunak pendukung aplikasi yang akan dibuat antara lain :

a. Sistem Operasi Windows XP
b. *Database* menggunakan MySQL
c. *Tools* implementasi perancangan sistem menggunakan Delphi 7
d. Wamp Server

3.1.2.3. Analisis *User* (Pengguna)

Analisis pengguna dimaksudkan untuk mengetahui siapa saja dan seperti apa karakteristik yang dimiliki oleh orang yang dapat menggunakan aplikasi yang akan dibuat. Untuk lebih jelas mengenai karakteristik pengguna, dapat dilihat pada tabel 3.1.

Tabel 3.1 Karakteristik Pengguna

Pengguna	Tanggung Jawab	Hak Akses	Tingkat Pendidikan	Tingkat Keterampilan	Pengalaman
User	Mengoperasikan aplikasi sesuai dengan kebutuhan akan data output dari aplikasi	Melakukan kegiatan yang berhubungan dengan isi tampilan yang ada pada program aplikasi	Minimal SMA	Bisa mengikuti petunjuk yang ada pada sistem	Minimal mampu mengoperasikan komputer

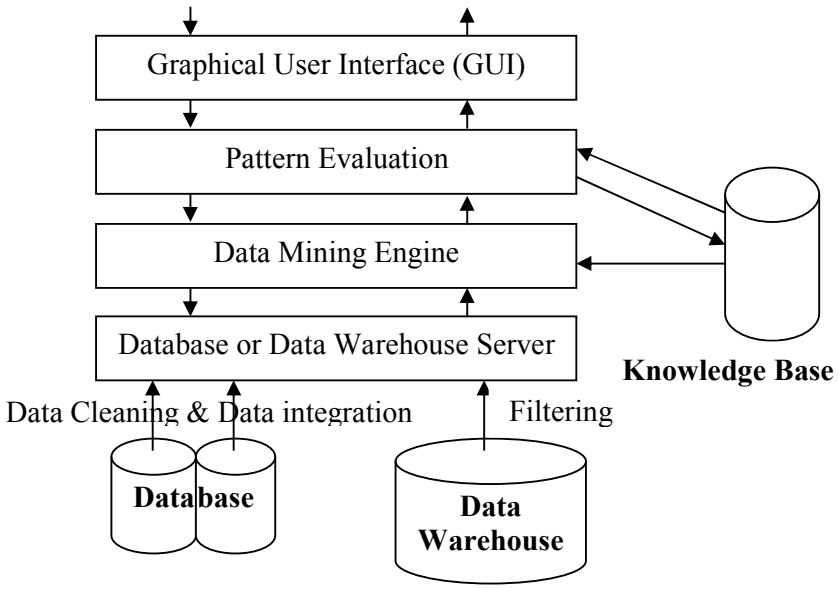
Dengan kondisi yang terdapat diatas, dapat diambil kesimpulan bahwa aplikasi yang akan dibuat dapat dijalankan olehsiapapun dengan latar belakang pendidikan minimal SMA yang terpenting adalah mampu mengoperasikan Komputer dan mampu mengikuti petunjuk dari aplikasi.User yang menjalankan aplikasi ini adalah orang-orang yang menjadi target end user dari aplikasi ini baik fund manager, broker perusahan securitas ataupun individual trader.

3.1.3 Analisa Data dan Metode

3.1.3.1 Analisa Data dan Metode menggunakan DT(decision Tree)

Pada Sub Bab ini akan berbicara tentang analisis data sampai dengan data dihasilkan, baik data apa yang dipakai, proses *cleaning dan integrasi data*serta tahapan mining data menggunakan algoritma DT, adapun acuan proses

analisis data mengacu kepada arsitektur data mining yang ditunjukan pada gambar 3.1 dibawah ini :



Gambar 3.1 Tahapan pada proses *Data Mining*

3.1.3.2 Proses Data Cleaning dan Integrasi Data

Adapun dalam proses perubahan data dari Excel yang berdasarkan pemahaman struktur data file excel berkategori *unspeanded*, untuk dapat dibaca dan diaplikasikan sehingga menghasilkan sebuah database dan memiliki struktur dara seperti ditunjukan seperti pada tabel 3.3, proses analisa data dapat diekpresikan dalam sebuah nilai dari format berbeda, secara instant berikut analisa yang dapat digunakan :

1. *Range Data Set* – Jenis Type data dari atribut dengan melihat isi dari field (Number (Integer/ real) date/ time, binary atau string/ text)
   a. *String* – Type data untuk field yang berisikan huruf
   b. *Numerik* – Type data untuk field yang berisikan angka

c. *Boolean* – Type data untuk field yang berisikan data hanya 2 type (ya/ tidak, benar/ salah, 1/0 dll)

d. *Date/ time* – untuk type data berisikan waktu

2. *Length data* - Jumlah dari character atau satuan apa dalam nilai matematis(string- char or varchar atau number longint, integer, real dan lainnya)

a. *String* – dengan kategori 2

1) Char – untuk field yang berisikan karakter satu huruf

2) Varchar - untuk field yang berisikan sederetan karakter yang membentuk satu kesatuan.

b. *Number* – Type data untuk berisikan angka

1) Integer– untuk field yang berisikan angka bilangan bulat dengan panjang data diantara -32768 s/d 32767

cat: jenis integer memiliki banyak type tergantung dari panjang dari bilangan bulat seperti small integer, Long integer, int64 dll)

2) Real Number - untuk field yang berisikan angka memiliki angka dibelakang koma atau ditulis dengan ($a \times 10^b$).

Didalam sebeum menganalisa sesuai aturan diatas dengan melihat isi data dan panjang dari data ditunjukan pada tabel 3.2 adalah tabel berisikan *field-field* pada data yang ada di excel berisikan 11 *field* untuk dianalisa baik type data dan Length data.

Tabel 3.2 *field* pada data STOCK_TRADES

Field
Code
Date
Prev
High
Low
Close
Change
Value
Volume
Frek
Index

Adapun analisa dari penentuan *Type data* dan *Length data* dari setiap *field* yang ada pada data pada file excel adalah sebagai berikut :

1. *Field '*CODE'
   a. *Range Data Set,*Untuk type data *field '*CODE' memiliki isi data berupa huruf (co: ISAT/ TLKM), hal ini sudah dapat dikategorikan kepada *typical* data string, dengan melihat konsistensi data berupa huruf dan berjumlah 4 maka dipilih type data 'CHAR'
   b. *Length data,* Untuk *length* data dari *field '*CODE' memiliki isi data berupa huruf (co: ISAT/ TLKM), hal ini sudah dapat dikategorikan kepada *typical* data string, dengan melihat konsistensi data berupa huruf dan berjumlah 4 maka *length* data sudah data ditentukan panjang data 4.
2. *Field '*DATE'.
   a. *Range Data Set,*Untuk type data *field 'DATE*' memiliki isi data berupa angka (co: 20040102/ 20040119), walaupun berupa angka bukan berarti type data *integer* atau *real* akan tetapi apabila diurai angka didalam isi pada *field* '*DATE*' memiliki informasi sebagai

berikut; 4 angka pertama menunjukan informasi tahun, 2 angka berikutnya menunjukan informasi bulan diperkuat dengan isi data 01-12, kemudian angka berikutnya menunjukan informasi tanggal diperkuat dengan data angka berisi data dari 01-31, maka untuk *field* 'DATE' dapat dipilih type data DATE.

b. *Length data,* Untuk *length* data dari *field 'DATE*' Karena type data sudah ditentukan date maka untuk *field* 'DATE', *length* data tidak perlu ditentukan.

3. *Field* 'PREV'
   a. *Range Data Set,*Untuk type data *field 'PREV*' memiliki isi data berupa angka (co: 15000/ 16300), hal ini sudah dapat dikategorikan kepada *typical* data number, dengan melihat konsistensi data berupa angka bilangan bulat maka dipilih type data *Integer*
   b. *Length data,* Untuk *length* data dari *field 'PREV*' memiliki panjang data dalam panjang data diantara -32768 s/d 32767 maka dapat ditentukan masuk dalam kategori *integer* dengan type data *Smallint*.
4. *Field* 'HIGH'
   a. *Range Data Set,*Untuk type data *field 'HIGH*' memiliki isi data berupa angka (co: 15200/ 15800), hal ini sudah dapat dikategorikan kepada *typical* data number, dengan melihat konsistensi data berupa angka bilangan bulat maka dipilih type data *Integer*
   b. *Length data,* Untuk *length* data dari *field 'HIGH*' memiliki panjang data dalam panjang data diantara -32768 s/d 32767 maka dapat

ditentukan masuk dalam kategori *integer* dengan type data *Smallint*.

c.

5. *Field* 'LOW'
   a. *Range Data Set,*Untuk type data *field 'LOW*' memiliki isi data berupa angka (co: 15000/ 15250), hal ini sudah dapat dikategorikan kepada *typical* data number, dengan melihat konsistensi data berupa angka bilangan bulat maka dipilih type data *Integer*
   b. *Length data,* Untuk *length* data dari *field 'LOW*' memiliki panjang data dalam panjang data diantara -32768 s/d 32767 maka dapat ditentukan masuk dalam kategori *integer* dengan type data *Smallint*.
6. *Field* 'CLOSE'
   a. *Range Data Set,*Untuk type data *field 'CLOSE*' memiliki isi data berupa angka (co: 15200/ 15800), hal ini sudah dapat dikategorikan kepada *typical* data number, dengan melihat konsistensi data berupa angka bilangan bulat maka dipilih type data *Integer*
   b. *Length data,* Untuk *length* data dari *field 'CLOSE*' memiliki panjang data dalam panjang data diantara -32768 s/d 32767 maka dapat ditentukan masuk dalam kategori *integer* dengan type data *Smallint*.
7. *Field* 'CHANGE'
   a. *Range Data Set,*Untuk type data *field 'CHANGE*' memiliki isi data berupa angka (co: 15200/ 15800), hal ini sudah dapat dikategorikan

kepada *typical* data number, dengan melihat konsistensi data berupa angka bilangan bulat maka dipilih type data *Integer*

b. *Length data,* Untuk *length* data dari *field 'CHANGE*' memiliki panjang data dalam panjang data diantara -32768 s/d 32767 maka dapat ditentukan masuk dalam kategori *integer* dengan type data *Smallint*.

8. *Field* 'VALUE'

a. *Range Data Set,*Untuk type data *field 'VALUE*' memiliki isi data berupa angka (co: 14409324544), hal ini sudah dapat dikategorikan kepada *typical* data number, dengan melihat data berupa angka bilangan bulat yang memiliki nominal diatas dalam satuan milyar dengan karakteristik data tidak dapat diperdiksikan maka dipilih type data *real.*

b. *Length data,* Untuk *length* data dari *field 'VALUE*' memiliki panjang data untuk field yang berisikan angka memiliki angka dibelakang koma atau ditulis dengan ($a \times 10^b$) dengan tidak dapat diprediksikan maka ditentukan masuk dalam kategori *Real* dengan type data *Double*.

9. *Field* 'VOLUME'

a. *Range Data Set,*Untuk type data *field 'VOLUME*' memiliki isi data berupa angka (co: 3261500, 953000), hal ini sudah dapat dikategorikan kepada *typical* data number, dengan melihat data berupa angka bilangan bulat yang memiliki nominal diatas dalam satuan puluhan ribu dengan karakteristik data tidak dapat diperdiksikan maka dipilih type data *real.*

b. *Length data,* Untuk *length* data dari *field 'VOLUME*' memiliki panjang data untuk field yang berisikan angka memiliki angka dibelakang koma atau ditulis dengan (a x 10^b) dengan tidak dapat diprediksikan maka ditentukan masuk dalam kategori *Real* dengan type data *Double*.

10. *Field* 'FREK',

a. *Range Data Set,*Untuk type data *field 'FREK*' memiliki isi data berupa angka (co: 167/ 24), hal ini sudah dapat dikategorikan kepada *typical* data number, dengan melihat konsistensi data berupa angka bilangan bulat maka dipilih type data *Integer*

b. *Length data,* Untuk *length* data dari *field 'FREK*' memiliki panjang data dalam panjang data diantara -32768 s/d 32767 maka dapat ditentukan masuk dalam kategori *integer* dengan type data *Smallint*.

11. *Field* 'INDEX'

a. *Range Data Set,*Untuk type data *field 'INDEX*' memiliki isi data berupa angka (co: 272.857, 232.857), hal ini sudah dapat dikategorikan kepada *typical* data number, dengan melihat data berupa angka dengan memiliki angka dibelakang koma maka dipilih type data *real.*

b. *Length data,* Untuk *length* data dari *field 'VOLUME*' memiliki panjang data untuk field yang berisikan angka memiliki angka dibelakang koma atau ditulis dengan (a x 10^b) maka ditentukan masuk dalam kategori *Real/ Float*.

Berdasarkan penelitian dan hasil analisa yang telah dilakukan, data transaksi saham yang disimpan dalam *database* pada tabel dengan nama STOCK_TRADES,dimana tabel tersebut terdiri dari 11 *field.* Berikut struktur table STOCK_TRADES bisa dilihat pada table 3.3.

Tabel 3.3 Struktur Tabel STOCK_TRADES

Field	Type	Length
Code	Char	4
Date	Date	
Prev	Smallint	
High	Smallint	
Low	Smallint	
Close	Smallint	
Change	Smallint	
Value	Double	
Volume	Double	
Frek	Smallint	
Index	Float	

Dibawah ini pada tabel 3.4 merupakan tabel sample data yang akan dimasukan pada data algoritma Decision Tree, dengan proses dicleaning dan diproses untuk menjadi data *preparation* untuk siap di mining. Adapun data setelah di cleaning lengkap dengan sample datanya ditampilkan pada tabel 3.6 dan Data yang siap di mining atau data preparation ditunjukan tabel 3.8 dengan setiap data yang dijadikan parameter target atau class harus berupa bilangan diskrit.

Tabel 3.4 *sample data Records* tabel STOCK_TRADES

Code	Date	Prev	High	Low	Close	Change	Value	Volume	Frek	Index
ISAT	20040102	15,000	15,200	15,000	15,200	200	14,409,324,544	953,000	125	217.143
ISAT	20040105	15,200	15,800	15,250	15,800	600	50,799,173,632	3,261,500	358	225.714
ISAT	20040106	15,800	16,650	15,800	16,400	600	79,629,852,672	4,882,000	415	234.286
ISAT	20040107	16,400	16,300	15,900	16,300	-100	44,128,423,936	2,739,500	224	232.857
ISAT	20040108	16,300	16,500	16,300	16,500	200	11,426,450,432	694,000	86	235.714
ISAT	20040109	16,500	16,650	16,300	16,600	100	35,859,001,344	2,177,000	143	237.143
ISAT	20040112	16,600	16,650	16,200	16,400	-200	39,365,873,664	2,386,500	191	234.286
ISAT	20040113	16,400	16,550	16,300	16,500	100	47,630,573,568	2,889,000	131	235.714
ISAT	20040114	16,500	16,750	16,500	16,750	250	30,773,925,888	1,800,000	145	239.286
ISAT	20040115	16,750	16,800	16,500	16,550	-200	27,656,900,608	1,666,000	163	236.429
ISAT	20040116	16,550	16,550	16,450	16,500	-50	18,832,900,096	1,141,000	98	235.714
ISAT	20040119	16,500	16,550	16,200	16,300	-200	17,104,050,176	1,049,500	173	232.857
ISAT	20040120	16,300	16,450	16,250	16,450	150	15,028,074,496	922,000	174	235
ISAT	20040121	16,450	16,800	16,500	16,800	350	63,596,875,776	999,999	303	240
ISAT	20040123	16,800	16,900	16,700	16,700	-100	14,866,950,144	1,000,000	70	238.571
ISAT	20040126	16,700	16,700	16,500	16,600	-100	41,066,000,384	2,481,000	108	237.143
ISAT	20040127	16,600	16,800	16,550	16,750	150	51,822,874,624	3,113,000	225	239.286
ISAT	20040128	16,750	16,650	16,300	16,400	-350	18,416,875,520	1,120,500	85	234.286
ISAT	20040129	16,400	16,050	15,950	16,050	-350	13,155,999,744	822,000	144	229.286
ISAT	20040130	16,050	16,150	15,850	16,150	100	32,614,500,352	2,036,000	112	230.714
ISAT	20040203	16,150	16,150	15,700	15,700	-450	36,405,899,264	2,302,000	220	224.286
ISAT	20040204	15,700	15,800	15,700	15,700	0	15,792,575,488	1,005,000	70	224.286
ISAT	20040205	15,700	16,100	15,500	16,000	300	33,378,050,048	2,095,500	215	228.571
ISAT	20040206	16,000	17,350	16,200	17,350	1,350	64,184,201,216	3,743,000	438	247.857
ISAT	20040209	17,350	18,300	17,400	18,200	850	64,273,174,528	3,559,000	467	260
ISAT	20040210	18,200	18,300	17,950	18,150	-50	19,170,549,760	1,056,500	132	259.286
ISAT	20040211	18,150	18,750	18,250	18,600	450	46,383,276,032	2,491,500	315	265.714
ISAT	20040212	18,600	19,250	18,700	19,100	500	59,828,473,856	2,333,987	371	272.857
ISAT	20040213	19,100	19,100	18,750	18,900	-200	29,922,224,128	1,585,000	240	270
SAT	20040216	18,900	18,900	18,650	18,750	-150	23,145,400,320	1,236,500	167	267.857
ISAT	20040217	18,750	18,700	18,650	18,650	-100	29,912,313,314	608,000	94	266.429

3.1.3.3 Data Preparation

3.1.3.3.1 Perubahan Data tahap I

Data di Proses dengan melewati Proses Cleaning dengan menghilangkan beberapa *field* yang diantaranya Code, Prev, Close dan Index yang nantinya tidak dipakai didalam proses Mining dengan menggunakan Metode Classification Teknik Decision Tree, disebabkan alasan sebagai berikut:

1. *Field '*Code' memiliki Nilai sama pada setiap recordnya
2. *Field* 'Prev', 'Close' dan 'Index' memiliki kesamaan perubahan dengan parameter 'Change' yang merupakan Parameter penentu dari Teknik decision tree dalam proses Data Mining, sehingga tidak dijadikan acuan sebagai parameter penentu node.

   Perubahaan sturktur tabel dari struktur tabel sebelum dicleaning yang ditunjukan tabel 3.3, adapun setelah dicleaning ditunjukan struktur tabelnya pada tabel 3.5.

Tabel 3.5 Struktur Tabel STOCK_TRADES

Field	Type	Length
Date	Date	
High	Smallint	
Low	Smallint	
Change	Smallint	
Value	Double	
Volume	Double	
Frek	Smallint	

Adapun perubahan data lengkap dengan *sample record* ditunjukan oleh tabel 3.6 sebelum data diproses kembali untuk disiapkan menjadi data preparation.

Tabel 3.6 30 *sample data Records* tabel STOCK_TRADES

Date	High	Low	Change	Value	Volume	Frek
20040102	15,200	15,000	200	14,409,324,544	953,000	125
20040105	15,800	15,250	600	50,799,173,632	3,261,500	358
20040106	16,650	15,800	600	79,629,852,672	4,882,000	415
20040107	16,300	15,900	-100	44,128,423,936	2,739,500	224
20040108	16,500	16,300	200	11,426,450,432	694,000	86
20040109	16,650	16,300	100	35,859,001,344	2,177,000	143
20040112	16,650	16,200	-200	39,365,873,664	2,386,500	191
20040113	16,550	16,300	100	47,630,573,568	2,889,000	131
20040114	16,750	16,500	250	30,773,925,888	1,800,000	145
20040115	16,800	16,500	-200	27,656,900,608	1,666,000	163
20040116	16,550	16,450	-50	18,832,900,096	1,141,000	98
20040119	16,550	16,200	-200	17,104,050,176	1,049,500	173
20040120	16,450	16,250	150	15,028,074,496	922,000	174
20040121	16,800	16,500	350	63,596,875,776	999,999	303
20040123	16,900	16,700	-100	14,866,950,144	1,000,000	70
20040126	16,700	16,500	-100	41,066,000,384	2,481,000	108
20040127	16,800	16,550	150	51,822,874,624	3,113,000	225
20040128	16,650	16,300	-350	18,416,875,520	1,120,500	85
20040129	16,050	15,950	-350	13,155,999,744	822,000	144
20040130	16,150	15,850	100	32,614,500,352	2,036,000	112
20040203	16,150	15,700	-450	36,405,899,264	2,302,000	220
20040204	15,800	15,700	0	15,792,575,488	1,005,000	70
20040205	16,100	15,500	300	33,378,050,048	2,095,500	215
20040206	17,350	16,200	1,350	64,184,201,216	3,743,000	438
20040209	18,300	17,400	850	64,273,174,528	3,559,000	467
20040210	18,300	17,950	-50	19,170,549,760	1,056,500	132
20040211	18,750	18,250	450	46,383,276,032	2,491,500	315
20040212	19,250	18,700	500	59,828,473,856	2,333,987	371
20040213	19,100	18,750	-200	29,922,224,128	1,585,000	240
20040216	18,900	18,650	-150	23,145,400,320	1,236,500	167
20040217	18,700	18,650	-100	29,912,313,314	608,000	94

3.1.3.3.2 Perubahan Data Tahap II

Didalam Teknik *Classification* keluaran dari setiap data yang dijadikan parameter target harus berupa bilangan diskrit. Dari data *sample* dengan isi *field* seperti ditunjukan pada tabel 3.5 dipilih *field* parameter sebagai penentu field yang dijadikan input algoritma adalah sebagai berikut :

1. Change sebagai penentu *field Class*
2. High sebagai penentu *field High (c)*

3. Low sebagai penentu *field Low (c)*
4. Value sebagai penentu *field Value (c)*
5. Volume sebagai penentu *field Volume (c)*
6. Frek sebagai penentu *field Frek (c)*

Adapun Field 'CLASS' sebagai penentu dengan mengidentifikasi field 'CHANGE' menjadi diskrit berikut asumsi perubahannya :

1. CLASS 'TURUN' dengan asumsi nilai parameter pada *field Change* berisi nilai *angka* (-)
2. CLASS 'SAMA; dengan asumsi nilai parameter pada *field change* berisi angka (0)
3. CLASS 'NAIK' dengan asumsi nilai parameter pada *field* berisi *angka* (+)

Adapun dalam menentukan data cleaning, *logical* proses yang dipakai adalah menentukan perubahan data kontinyu ke diskrit dengan logika perubahan

1. Pada *field* target berisi 'NAIK' apabila nilai record bernilai lebih besar dari nilai record sebelumnya
2. Pada *field* target berisi 'SAMA' apabila nilai record bernilai sama dari nilai record sebelumnya
3. Pada *field* target berisi CLASS 'TURUN' apabila nilai record dinyatakan lebih kecil dari nilai record sebelumnya.

Adapun perubahan setelah dicleaning dan diproses menjadi data preparation untuk selanjutnya siap dimining menggunakan metode *Classification* dengan teknik *Decision tree* lengkap dengan sample datanya dengan penerapan logika proses diatas ditunjukan pada tabel 3.7.

Tabel 3.7 30 *sample data* tabel setelah Data diproses menjadi Data preparation tahap 2

Date	High	Low	Change	Value	Volume	Frek	Class	High(c)	Low (c)	Value(C)	Volume(C)	Frek(C)
20040319	3,850	3,525	300	111,409,799,168	29,770,000	1,128	Naik	Naik	Naik	Naik	Naik	Naik
20040323	3,875	3,650	25	58,351,714,304	15,455,000	546	Naik	Naik	Naik	Turun	Turun	Turun
20040324	3,850	3,750	-75	22,750,699,520	6,001,500	320	Turun	Turun	Naik	Turun	Turun	Turun
20040325	3,775	3,700	-25	25,901,400,064	6,963,500	233	Turun	Turun	Turun	Naik	Naik	Turun
20040326	3,750	3,650	-50	10,179,862,528	2,762,500	205	Turun	Turun	Turun	Turun	Turun	Turun
20040329	3,700	3,675	0	1,631,837,440	443,000	215	Sama	Turun	Naik	Turun	Turun	Naik
20040330	3,775	3,700	100	12,158,087,168	3,237,500	245	Naik	Naik	Naik	Naik	Naik	Naik
20040331	3,875	3,800	75	48,793,276,416	12,688,500	499	Naik	Naik	Naik	Naik	Naik	Naik
20040401	3,850	3,800	-25	24,841,775,104	6,491,500	192	Turun	Turun	Sama	Turun	Turun	Turun
20040402	3,850	3,750	-75	44,418,011,136	11,767,000	365	Turun	Sama	Turun	Naik	Naik	Naik
20040406	3,925	3,800	175	72,068,841,472	18,635,000	732	Naik	Naik	Naik	Naik	Naik	Naik
20040407	4,250	3,925	325	121,989,079,040	29,735,000	989	Naik	Naik	Naik	Naik	Naik	Naik
20040408	4,350	4,150	50	88,768,798,720	20,691,500	620	Naik	Naik	Naik	Turun	Turun	Turun
20040412	4,275	4,150	-100	28,152,475,648	6,700,000	292	Turun	Turun	Sama	Turun	Turun	Turun
20040413	4,250	4,125	-50	48,986,898,432	11,691,000	327	Turun	Turun	Turun	Naik	Naik	Naik
20040414	4,150	4,050	-50	43,592,974,336	10,660,500	470	Turun	Turun	Turun	Turun	Turun	Naik
20040415	4,125	4,050	-25	20,422,025,216	4,996,000	237	Turun	Turun	Sama	Turun	Turun	Turun
20040416	4,125	4,075	50	13,162,274,816	3,206,500	187	Naik	Sama	Naik	Turun	Turun	Turun
20040419	4,275	4,125	125	77,511,434,240	18,293,000	466	Naik	Naik	Naik	Naik	Naik	Naik
20040420	4,325	4,225	25	91,812,487,168	21,438,000	571	Naik	Naik	Naik	Naik	Naik	Naik
20040421	4,650	4,250	250	159,741,067,264	35,589,500	1,063	Naik	Naik	Naik	Naik	Naik	Naik
20040422	4,525	4,375	-100	40,316,088,320	9,098,000	410	Turun	Turun	Naik	Turun	Turun	Turun
20040423	4,450	4,325	-25	30,922,637,312	7,033,500	336	Turun	Turun	Turun	Turun	Turun	Turun
20040426	4,450	4,350	50	31,518,300,160	7,177,500	217	Naik	Sama	Naik	Naik	Naik	Turun
20040427	4,525	4,425	25	31,108,249,600	6,948,000	302	Naik	Naik	Naik	Turun	Turun	Naik
20040428	4,475	4,400	-50	15,495,100,416	3,505,500	179	Turun	Turun	Turun	Turun	Turun	Turun
20040429	4,400	4,150	-250	107,094,409,216	25,329,500	1,255	Turun	Turun	Turun	Naik	Naik	Naik
20040430	4,150	3,950	-200	113,598,963,712	28,365,500	1,604	Turun	Turun	Turun	Naik	Naik	Naik
20040504	4,150	4,025	150	95,086,551,040	23,172,500	956	Naik	Sama	Naik	Turun	Turun	Turun
20040505	4,150	4,000	-100	36,140,974,080	8,925,000	487	Turun	Sama	Turun	Turun	Turun	Turun

3.1.3.4 Data Mining Engine

3.1.3.4.1 Proses Pembentukan Pohon Keputusan

Berikut ini adalah contoh pembentukan pohon keputusan pada node pertama dengan menggunakan algorita C 4.5 terhadap tabel berisikan 30 *sampel data record* yang diambil dari database Sistem rekapitulasi penjualan saham di BEI, setelah melalui proses penyeleksian parameter. Model Pohon keputusan yang akan dibangun diasumsikan setelah diproses cleaning semua parameter digunakan dengan parameter 'Class' sebagai parameter *Class*-nya. Parameter yang digunakan memiliki pengaruh yang cukup tinggi dalam perubahan dan pergerakan saham. Adapun pemilihan parameter dalam pembentukan pohon keputusan ditenntukan dari semua data masukan seperti ditunjukan pada tabel 3.3 ada beberapa *field* dengan yang dipilih setelah melewati data Cleaning dan data preparation, berikut alasan penentuan parameter tersebut :

a. Field yang tidak dipilih baik dari awal sebelum diproses, adalah sebagai berikut :

   1) *Field* 'Code', *Field* ini dihilangkan pada saat proses cleaning dikarenakan isi dari semua *recordnya* sama.

   2) *Field* 'Date', *Field* ini tidak dihilangkan akan tetapi tidak dipilih juga dalam pemilihan parameter karena *Field* ini berisikan *content* unik pada setiap *record.*

   3) *Field* 'Prev', *Field* ini dihilangkan pada saat proses cleaning dikarenakan *field* ini dalam perubahannya memiliki

identifikasi sama dengan *field* ’Change’ yang merupakan *field* penentu parameter class yang menjadi parameter target.

4) *Field* ‘Close’, *Field* ini dihilangkan pada saat proses cleaning sama halnya dengan *field ‘prev’* dikarenakan *field* ini dalam perubahannya memiliki identifikasi sama dengan *field* ’Change’ yang merupakan *field* penentu parameter class yang menjadi parameter target.

5) *Field* ‘Index’, *Field* ini dihilangkan pada saat proses cleaning samah halnya dengan *field ‘prev’* dan *field ‘close’* dikarenakan *field* ini dalam perubahannya memiliki identifikasi sama dengan *field* ’Change’ yang merupakan *field* penentu parameter class yang menjadi parameter target.

b. *Field* yang dipilih sebagai acuan pembentuk node, adalah sebagai berikut :

1) *Field ‘*Change*’, Field* ini dipilih karena mengidentifikasi Class dari setiap *recordnya* apakah ‘Naik’, ‘Turun’ ataupun ‘Sama’.

2) *Field ‘*High*’, Field* ini dipilih karena mengidentifikasi Harga tertinggi, dimana mengidentifikasi perubahan tertinggi harga saham pada hari atau record tersebut.

3) *Field ‘*Low*’, Field* ini dipilih karena mengidentifikasi Harga terendah, dimana mengidentifikasi perubahan terendah harga saham pada hari atau record tersebut.

4) *Field* 'Value', *Field* ini dipilih karena mengidentifikasi nilai transaksi dalam rupiah dikalikan dengan jumlah saham pada saat hari atau record tersebut.

5) *Field* 'Volume', *Field* ini dipilih karena mengidentifikasi nilai transaksi dalam rupiah pada saat hari atau record tersebut.

6) *Field* 'Frek', *Field* ini dipilih karena mengidentifikasi jumlah transaksi yang terjadi pada hari atau record tersebut.

Adapun pemilihan untuk pemilihan root maka dilakukan perhitungan gain terlebih dahulu seperti yang tertera pada rumus 1

$$Gain(S,A) = Entropy(S) - \sum_{i=1}^{n} \frac{|S_i|}{|S|} * Entropy\ (S_i) \ldots\ldots\ldots\ldots\ldots (1)$$

Dengan

S : Himpunan Kasus

A : Atribut

N : Jumlah Partisi A

$|S_i|$: Jumlah kasus pada partisi ke I

$|S|$: Jumlah Kasus dalam S

Sedangkan untuk perhitungan nilai Entropy dapat dilihat pada rumus 2

$$Entropy(S) = \sum_{i=1}^{n} - pi * log2pi \ldots\ldots\ldots\ldots\ldots\ldots\ldots (2)$$

Adapun dalam pembentukan Node Langkah- langkah dan aturan seperti dibawag ini :

a. Hal pertama yang akan dilakukan adalah menghitung jumlah kasus secara keseluruhan, kemudian jumlah kasus untuk class NAIK, jumlah

kasus untuk class TURUN, jumlah kasus untuk SAMA, entrophy dari semua kasus dan mengitung Entropy sesuai dengan parameter yang dipilih berdasarkan parameter yang akan digunakan.

b. Setelah nilai gain didapat untuk setiap parameternya, maka parameter dengan nilai gain tetinggi akan dijadikan root pada model keputusan.

c. Setelah itu maka akan dilakukan perhitungan *gain* untuk setiap parameter yang masih tersisa.

d. Apabila dalam kasus perhitungan nilai gain tertinggi lebih dari satu akan dipilih nilai gain yang masuk pertama kali.

3.1.3.4.2 Perhitungan 30 Sample Record

Berikut langkah-langkah pembentukan pohon keputusan dari 30 sample data record sampai membentuk sebuah pohon keputusan:

1. Perhitungan untuk Node 1
   a. Menentukan Entropy(S). Untuk mencari *entropy* himpunan kasus maka dilakukan perhitungan dengan menggunakan rumus 2 sebagai berikut :

$$Entropy(S) = \sum_{i=1}^{n} - pi * log2pi \ldots\ldots\ldots\ldots\ldots\ldots\ldots (2)$$

$$Entropy(S) = \left(\frac{15}{30} * log_2\left(\frac{15}{30}\right)\right) + \left(\frac{1}{30} * log_2\left(\frac{1}{30}\right)\right) + \left(\frac{14}{30} * log_2\left(\frac{14}{30}\right)\right)$$

$$Entropy(S) = 1.1767$$

   b. Menentukan root. Parameter yang dijadikan *root* pada pohon kepurusan adalah parameter yang memiliki gain tertinggi, berikut

perhitungan gain untuk setiap parameternya dengan menggunakan rumus 1.

1) Parameter High

$$Gain(S, High) = Entropy(S) - \sum_{i=1}^{n} \frac{|S_i|}{|S|} * Entropy\ (S_i) \ldots (1)$$

$$Gain(S, High) = 1.1767 - \left(\left(\frac{|15|}{|30|} * 0.5665\right) + \left(\frac{|4|}{|30|} * 0.0000\right) + \left(\frac{|10|}{|30|} * 1.0958\right)\right)$$

$$Gain(S, High) = 0{,}4916$$

2) Parameter Low

$$Gain(S, Low) = Entropy(S) - \sum_{i=1}^{n} \frac{|S_i|}{|S|} * Entropy\ (S_i) \ldots (1)$$

$$Gain(S, Low) = 1.1767 - \left(\left(\frac{|16|}{|30|} * 0.8113\right) + \left(\frac{|4|}{|30|} * 1.5000\right) + \left(\frac{|10|}{|30|} * 0.7219\right)\right)$$

$$Gain(S, Low) = 0{,}3034$$

3) Parameter Value

$$Gain(S, \text{Value}) = Entropy(S) - \sum_{i=1}^{n} \frac{|S_i|}{|S|} * Entropy\ (S_i) \ldots (1)$$

$$Gain(S, \text{Value}) = 1.1767 - \left(\left(\frac{|16|}{|30|} * 0.8113\right) + \left(\frac{|14|}{|30|} * 1.0949\right)\right)$$

$$Gain(S, \text{Value}) = 0{,}2330$$

4) Parameter Volume

$$Gain(S, Volume) = Entropy(S) - \sum_{i=1}^{n} \frac{|S_i|}{|S|} * Entropy\ (S_i) \ldots (1)$$

$$Gain(S, Volume) = 1.1767 - \left(\left(\frac{|14|}{|30|} * 0.8631\right) + \left(\frac{|16|}{|30|} * 1.1982\right)\right)$$

$$Gain(S, Volume) = 0{,}1349$$

5) Parameter Frek

$$Gain(S, Frek) = Entropy(S) - \sum_{i=1}^{n} \frac{|S_i|}{|S|} * Entropy\ (S_i) \ldots (1)$$

$$Gain(S, Frek) = 1.1767 - \left(\left(\frac{|18|}{|30|} * 0.9183\right) + \left(\frac{|12|}{|30|} * 1.1887\right)\right)$$

$$Gain(S, Frek) = 0{,}1502$$

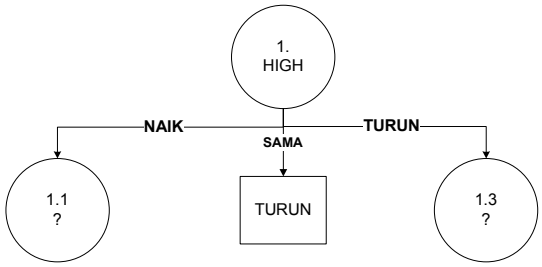
Dari Hasil perhitungan dapat diketahui bahwa parameter High memiliki nilai gain tertinggi sebesar 0,4916, maka parameter ini dapat menjadi root node pertama dari model keputusan untuk hasil perhitungan lengkapnya dapat dilihat pada tabel 3.7.

Ada tiga data yang dimiliki oleh parameter High yakni ‘Naik’, ‘sama’ dan Turun’. Dari kedua data parameter tersebut, kondisi ‘sama’ saja yang terklasifikasi class ‘TURUN’ sedangkan kondisi ‘Naik’ dan ‘turun’ memiliki persepsi yang belum terklasifikasi dari hasil Node pertama sehingga kedua cabang node pertama dihitung dengan memisahkan data acuan parameter volume baik ‘naik’ atau turun menghasilkan tabel yang berbeda.

Tabel 3.8 Perhitungan Node 1

NODE	Parameter	Nama Parameter	Jumlah Data	Naik	Sama	Turun	Entropy	Gain
1		Total	30	15	1	14	1.1767	
	High							0.4916
		Naik	15	13	0	2	0.5665	
		Sama	4	0	0	4	0.0000	
		Turun	11	2	1	8	1.0958	
	Low							0.3034
		Naik	16	12	0	4	0.8113	
		Sama	4	1	1	2	1.5000	
		Turun	10	2	0	8	0.7219	
	Value							0.2330
		Naik	16	12	0	4	0.8113	
		Turun	14	3	1	10	1.0949	
	Volume							0.1349
		Naik	14	10	0	4	0.8631	
		Turun	16	5	1	10	1.1982	
	Frek							0.1502
		Naik	18	12	0	6	0.9183	
		Turun	12	3	1	8	1.1887	

Hasil perhitungan dari Node pertama ini dapat digambarkan sebagai pohon keputusan seperti gambar 3.2



Gambar 3.2 Pohon keputusan Hasil perhitungan Node 1

2. Perhitungan untuk Node 1.1

   a. Menentukan Entropy (High=’NAIK’). Untuk mencari *entropy* himpunan kasus dengan nilai parameter High=’NAIK’ maka dilakukan perhitungan dengan menggunakan rumus 2 sebagai berikut :

$$Entropy(S) = \left(\frac{13}{15} * log_2\left(\frac{13}{15}\right)\right) + \left(\frac{0}{14} * log_2\left(\frac{0}{14}\right)\right) + \left(\frac{2}{14} * log_2\left(\frac{2}{14}\right)\right)$$

$$Entropy(S) = 0.5665$$

   b. Menentukan node cabang untuk parameter High dengan data ’NAIK’ Parameter yang dijadikan *node* cabang adalah parameter yang memiliki gain tertinggi, berikut perhitungan gain untuk setiap parameternya dengan menggunakan rumus 1.

      1) Parameter Low

$$Gain(S, Low) = Entropy(S) - \sum_{i=1}^{n} \frac{|S_i|}{|S|} * Entropy\ (S_i)$$

$$Gain(S, Low) = 0.5665 - \left(\left(\frac{|11|}{|15|} * 0.3145\right) + \left(\frac{|2|}{|15|} * 1.0000\right) + \left(\frac{|2|}{|15|} * 0.0000\right)\right)$$

$$Gain(S, Low) = 0{,}2025$$

2) Parameter Value

$$Gain(S, \text{Value}) = Entropy(S) - \sum_{i=1}^{n} \frac{|S_i|}{|S|} * Entropy\ (S_i)$$

$$Gain(S, \text{Value}) = 0.5665 - \left(\left(\frac{|11|}{|15|} * 0.0000\right) + \left(\frac{|4|}{|15|} * 1.0000\right)\right)$$

$$Gain(S, \text{Value}) = 0{,}2998$$

3) Parameter Volume

$$Gain(S, Volume) = Entropy(S) - \sum_{i=1}^{n} \frac{|S_i|}{|S|} * Entropy\ (S_i)$$

$$Gain(S, Volume) = 0.5665 - \left(\left(\frac{|10|}{|15|} * 0.4690\right) + \left(\frac{|5|}{|15|} * 0.7219\right)\right)$$

$$Gain(S, Volume) = 0{,}0132$$

4) Parameter Frek

$$Gain(S, Frek) = Entropy(S) - \sum_{i=1}^{n} \frac{|S_i|}{|S|} * Entropy\ (S_i)$$

$$Gain(S, Frek) = 0.5665 - \left(\left(\frac{|12|}{|15|} * 0.4138\right) + \left(\frac{|3|}{|15|} * 0.9183\right)\right)$$

$$Gain(S, Frek) = 0{,}0518$$

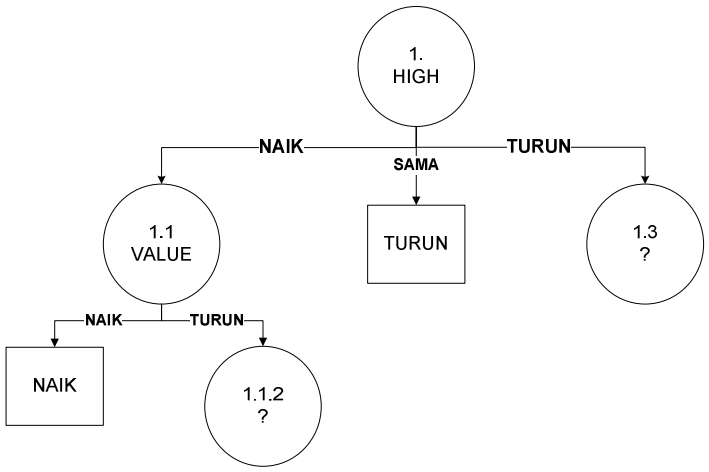
Dari Hasil perhitungan dapat diketahui bahwa parameter value memiliki nilai gain tertinggi sebesar 0,2998 maka parameter ini dapat menjadi node cabang dari parameter High dengan data record ‘Naik’. Ada 2 data *record* pada node ini hanya 1 yang terklasifikasi, dengan data pada parameter Value yang berisi ‘Naik’ menyatakan *class NAIK* dan data parameter ‘Turun’ belum terklasifikasi *class*

sehingga perlu dihitung kembali untuk node 1.1.2. Adapun hasil perhitungan lengkapnya dapat dilihat pada tabel 3.8.

Tabel 3.9 Perhitungan Node 1.1

NODE	Parameter	Nama Parameter	Jumlah Data	Naik	Sama	Turun	Entropy	Gain
1.1		High	15	13	0	2	0.5665	
	Low							0.2025
		Naik	11	11	0	1	0.3145	
		Sama	2	1	0	1	1.0000	
		Turun	2	2	0	0	0.0000	
	Value							0.2998
		Naik	11	11	0	0	0.0000	
		Turun	4	2	0	2	1.0000	
	Volume							0.0132
		Naik	10	9	0	1	0.4690	
		Turun	5	4	0	1	0.7219	
	Frek							0.0518
		Naik	12	11	0	1	0.4138	
		Turun	3	2	0	1	0.9183	

Adapun Pohon keputusan setelah perhitungan node 1.1 sampai tahap ini dapat dilihat pada gambar 3.3.



Gambar 3.3 Pohon keputusan Hasil perhitungan node 1.1

3. Perhitungan untuk Node 1.3

   a. Menentukan Entropy (High=’Turun’). Untuk mencari *entropy* himpunan kasus dengan nilai parameter High=’Turun’ maka dilakukan perhitungan dengan menggunakan rumus 2 sebagai berikut :

$$Entropy(S) = \left(\frac{2}{11} * log_2\left(\frac{2}{11}\right)\right) + \left(\frac{1}{11} * log_2\left(\frac{1}{11}\right)\right) + \left(\frac{8}{11} * log_2\left(\frac{8}{11}\right)\right)$$

$$Entropy(S) = 1.0958$$

   b. Menentukan node cabang untuk parameter High dengan data ’Turun’ Parameter yang dijadikan *node* cabang adalah parameter yang memiliki gain tertinggi, berikut perhitungan gain untuk setiap parameternya dengan menggunakan rumus 1.

      1) Parameter Low

$$Gain(S, Low) = Entropy(S) - \sum_{i=1}^{n} \frac{|S_i|}{|S|} * Entropy\ (S_i)$$

$$Gain(S, Low) = 1.1887 - \left(\left(\frac{|4|}{|11|} * 1.0000\right) + \left(\frac{|2|}{|11|} * 1.0000\right) + \left(\frac{|5|}{|11|} * 0.0000\right)\right)$$

$$Gain(S, Low) = 0{,}5503$$

      2) Parameter Value

$$Gain(S, \text{Value}) = Entropy(S) - \sum_{i=1}^{n} \frac{|S_i|}{|S|} * Entropy\ (S_i)$$

$$Gain(S, \text{Value}) = 1.1887 - \left(\left(\frac{|3|}{|11|} * 0.9183\right) + \left(\frac{|8|}{|11|} * 1.0613\right)\right)$$

$$Gain(S, \text{Value}) = 0{,}0735$$

      3) Parameter Volume

$$Gain(S, Volume) = Entropy(S) - \sum_{i=1}^{n} \frac{|S_i|}{|S|} * Entropy\ (S_i)$$

$$Gain(S, Volume) = 1.1887 - \left(\left(\frac{|2|}{|11|} * 1.0000\right) + \left(\frac{|9|}{|11|} * 0.9864\right)\right)$$

$$Gain(S, Volume) = 0.1069$$

4) Parameter Frek

$$Gain(S, Frek) = Entropy(S) - \sum_{i=1}^{n} \frac{|S_i|}{|S|} * Entropy\ (S_i)$$

$$Gain(S, Frek) = 1.1887 - \left(\left(\frac{|3|}{|12|} * 0.9183\right) + \left(\frac{|9|}{|12|} * 1.2244\right)\right)$$

$$Gain(S, Frek) = 0{,}0735$$

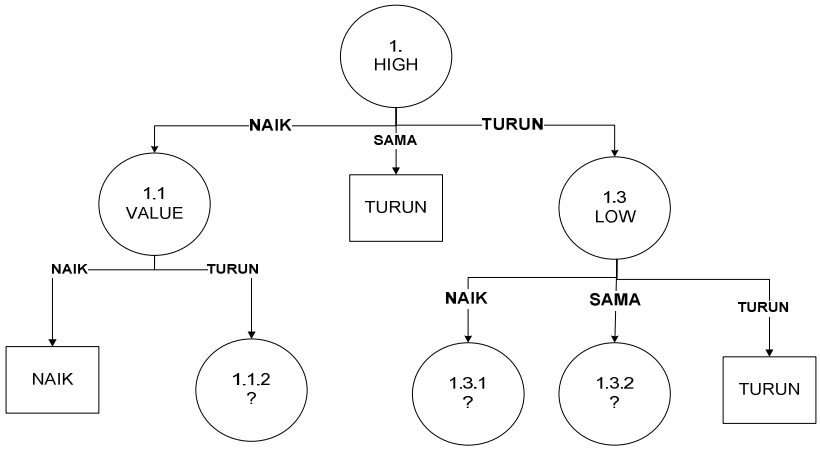
Dari Hasil perhitungan dapat diketahui bahwa parameter Low memiliki nilai gain tertinggi sebesar 0,5503, maka parameter ini dapat menjadi node cabang dari parameter High =’Turun’.

Ada tiga data yang dimiliki oleh parameter low yakni ‘Naik’, ‘sama’ dan Turun’. Dari kedua data parameter tersebut, kondisi ‘Turun’ terklasifikasi class ‘TURUN’ sedangkan kondisi ‘Naik’ dan ‘sama’ yang belum terklasifikasi dari hasil Node pertama sehingga kedua cabang node pertama dihitung dengan memisahkan data acuan parameter low baik ‘naik’ atau ‘sama’ menghasilkan tabel yang berbeda. Adapun hasil perhitungan dari Node 1.3 dapat terlihat pada tabel 3.10

Tabel 3.10 Perhitungan Node 1.3

NODE	Parameter	Nama Parameter	Jumlah Data	Naik	Sama	Turun	Entropy	Gain
1.3		Total	11	2	1	8	1.0958	
	Low							0.5503
		Naik	4	2	0	2	1.0000	
		Sama	2	0	1	1	1.0000	
		Turun	5	0	0	5	0.0000	
	Value							0.0735
		Naik	3	1	0	2	0.9183	
		Turun	8	1	1	6	1.0613	
	Volume							0.1069
		Naik	2	1	0	1	1.0000	
		Turun	9	1	1	7	0.9864	
	Frek							0.0735
		Naik	3	1	0	2	0.9183	
		Turun	8	1	1	6	1.0613	

Adapun Pohon keputusan setelah perhitungan node 1.1 sampai tahap ini dapat dilihat pada gambar 3.3.



Gambar 3.4 Pohon Hasil perhitungan keputusan node 1.3

4. Perhitungan untuk Node 1.1.2

   a. Menentukan Entropy (Value=’Turun’). Untuk mencari *entropy* himpunan kasus dengan nilai parameter Value=’Turun’ maka

dilakukan perhitungan dengan menggunakan rumus 2 sebagai berikut :

$$Entropy(S) = \left(\frac{2}{4} * log_2\left(\frac{2}{4}\right)\right) + \left(\frac{0}{4} * log_2\left(\frac{0}{4}\right)\right) + \left(\frac{2}{4} * log_2\left(\frac{2}{4}\right)\right)$$

$$Entropy(S) = 1.0000$$

b. Menentukan node cabang untuk parameter Value dengan data ’Turun’ Parameter yang dijadikan *node* cabang adalah parameter yang memiliki gain tertinggi, berikut perhitungan gain untuk setiap parameternya dengan menggunakan rumus 1.

1) Parameter Low

$$Gain(S, Low) = Entropy(S) - \sum_{i=1}^{n} \frac{|S_i|}{|S|} * Entropy\ (S_i)$$

$$Gain(S, Low) = 1.0000 - \left(\left(\frac{|3|}{|4|} * 0.9183\right) + \left(\frac{|1|}{|4|} * 0.0000\right) + \left(\frac{|0|}{|4|} * 0.0000\right)\right)$$

$$Gain(S, Low) = 0{,}3113$$

2) Parameter Volume

$$Gain(S, Volume) = Entropy(S) - \sum_{i=1}^{n} \frac{|S_i|}{|S|} * Entropy\ (S_i)$$

$$Gain(S, Volume) = 1.0000 - \left(\left(\frac{|1|}{|4|} * 0.0000\right) + \left(\frac{|3|}{|4|} * 0.9183\right)\right)$$

$$Gain(S, Volume) = 0{,}3113$$

3) Parameter Frek

$$Gain(S, Frek) = Entropy(S) - \sum_{i=1}^{n} \frac{|S_i|}{|S|} * Entropy\ (S_i)$$

$$Gain(S, Frek) = 1.0000 - \left(\left(\frac{|2|}{|4|} * 0.5000\right) + \left(\frac{|2|}{|4|} * 1.0000\right)\right)$$

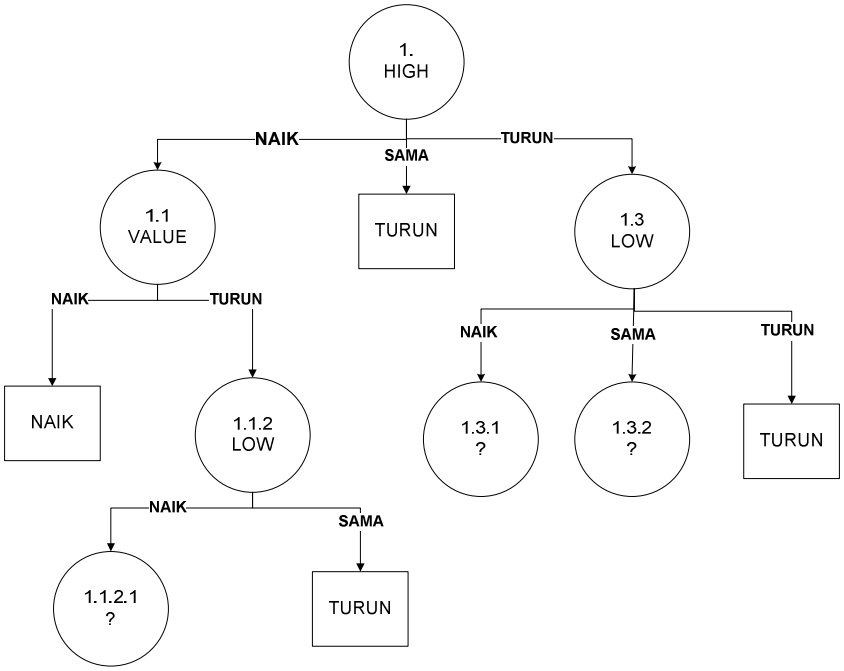
$$Gain(S, Frek) = 0{,}2500$$

Dari Hasil perhitungan dapat diketahui bahwa parameter Low dan volume memiliki nilai gain tertinggi sebesar 0,3113 akan tetapi parameter low kecenderung lebih pertama dan variable penentunya lebih maka parameter ini dapat menjadi node cabang dari parameter Value dengan data record 'turun'. Ada 2 data *record* pada node ini hanya 1 yang terklasifikasi, dengan data pada parameter Low yang berisi 'Sama' menyatakan *class Turun* sedangkan data parameter *'NAIK'* belum terklasifikasi sehingga perlu dihitung kembali untuk node 1.1.2.1. Adapun hasil perhitungan lengkapnya dapat dilihat pada tabel 3.11.

Tabel 3.11 Perhitungan Node 1.1.2

NODE	Parameter	Nama Parameter	Jumlah Data	Naik	Sama	Turun	Entropy	Gain
1.1.2		Value	4	2	0	2	1.0000	
	Low							0.3113
		Naik	3	2	0	1	0.9183	
		Sama	1	0	0	1	0.0000	
		Turun	0	0	0	0	0.0000	
	Volume							0.3113
		Naik	1	0	0	1	0.0000	
		Turun	3	2	0	1	0.9183	
	Frek							0.2500
		Naik	2	1	0	1	0.5000	
		Turun	2	1	0	1	1.0000	

Adapun Pohon keputusan setelah perhitungan node 1.1.2 sampai tahap ini dapat dilihat pada gambar 3.3.



Gambar 3.5 Pohon Hasil perhitungan keputusan node 1.1.2

5. Perhitungan Untuk Node 1.3.1

   a. Menentukan Entropy (Low=’Naik’). Untuk mencari *entropy* himpunan kasus dengan nilai parameter Low=’Naik’ maka dilakukan perhitungan dengan menggunakan rumus 2 sebagai berikut :

$$Entropy(S) = \left(\frac{2}{4} * log_2\left(\frac{2}{4}\right)\right) + \left(\frac{0}{4} * log_2\left(\frac{0}{4}\right)\right) + \left(\frac{2}{4} * log_2\left(\frac{2}{4}\right)\right)$$

$$Entropy(S) = 1.0000$$

   b. Menentukan node cabang untuk parameter Low dengan data ’Naik’ Parameter yang dijadikan *node* cabang adalah parameter yang memiliki gain tertinggi, berikut perhitungan gain untuk setiap parameternya dengan menggunakan rumus 1.

1) Parameter Value

$$Gain(S, \text{Value}) = Entropy(S) - \sum_{i=1}^{n} \frac{|S_i|}{|S|} * Entropy\ (S_i)$$

$$Gain(S, \text{Value}) = 1.0000 - \left(\left(\frac{|1|}{|4|} * 0.0000\right) + \left(\frac{|3|}{|4|} * 0.9183\right)\right)$$

$$Gain(S, \text{Value}) = 0{,}3113$$

2) Parameter Volume

$$Gain(S, Volume) = Entropy(S) - \sum_{i=1}^{n} \frac{|S_i|}{|S|} * Entropy\ (S_i)$$

$$Gain(S, Volume) = 1.0000 - \left(\left(\frac{|1|}{|4|} * 0.0000\right) + \left(\frac{|3|}{|4|} * 0.9183\right)\right)$$

$$Gain(S, Volume) = 0{,}3113$$

3) Parameter Frek

$$Gain(S, Frek) = Entropy(S) - \sum_{i=1}^{n} \frac{|S_i|}{|S|} * Entropy\ (S_i)$$

$$Gain(S, Frek) = 1.0000 - \left(\left(\frac{|1|}{|4|} * 0.0000\right) + \left(\frac{|3|}{|4|} * 0.9183\right)\right)$$

$$Gain(S, Frek) = 0{,}3113$$

Dari Hasil perhitungan dapat diketahui bahwa parameter Value memiliki nilai gain tertinggi sebesar 0,3113 walaupun nilai gainnya sama dengan nilai frek akan tetapi karena data yang berbeda dengan yang lain adalah frek, yang dipilih adalah parameter frek, maka parameter ini dapat menjadi node cabang dari parameter low =’Naik’.

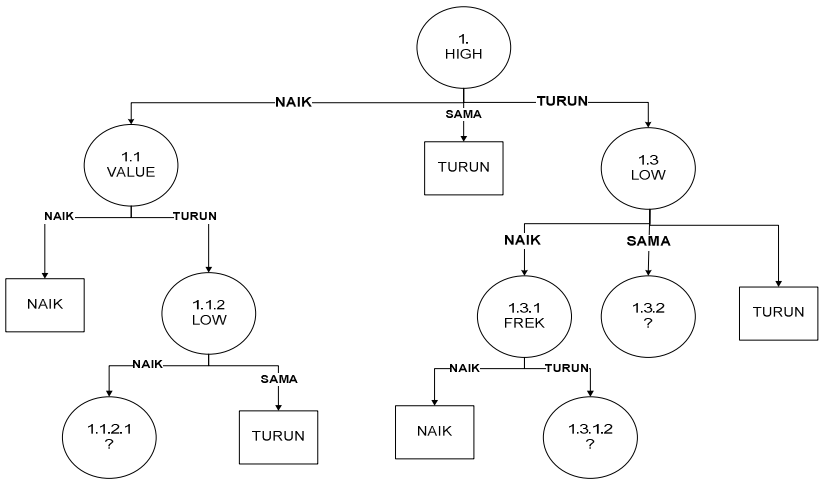
Ada dua data yang dimiliki oleh parameter value yakni ‘Naik’, dan Turun’. Dari kedua data parameter tersebut, kondisi ‘naik’ yang

telah terklasifikasi class ‘NAIK’ sedangkan kondisi ‘Turun’ yang belum terklasifikasi sehingga perlu dihitung kembali untuk node 1.3.1.2. Adapun hasil perhitungan dari Node 1.3.1 dapat terlihat pada tabel 3.12

Tabel 3.12 Perhitungan Node 1.3.1

NODE	Parameter	Nama Parameter	Jumlah Data	Naik	Sama	Turun	Entropy	Gain
1.3.1		Low	4	2	0	2	1.0000	
	Value							0.3113
		Naik	1	1	0	0	0.0000	
		Turun	3	1	0	2	0.9183	
	Volume							0.3113
		Naik	1	1	0	0	0.0000	
		Turun	3	1	0	2	0.9183	
	Frek							0.3113
		Naik	1	1	0	0	0.0000	
		Turun	3	1	0	2	0.9183	

Adapun Pohon keputusan setelah perhitungan node 1.3.1 sampai tahap ini dapat dilihat pada gambar 3.6.



Gambar 3.6 Pohon keputusan node 1.3.1

6. Perhitungan Untuk Node 1.3.2

a. Menentukan Entropy (Low=’Sama’). Untuk mencari *entropy* himpunan kasus dengan nilai parameter low=’Sama’ maka dilakukan perhitungan dengan menggunakan rumus 2 sebagai berikut :

$$Entropy(S) = \left(\frac{0}{2} * log_2\left(\frac{0}{2}\right)\right) + \left(\frac{1}{2} * log_2\left(\frac{1}{2}\right)\right) + \left(\frac{1}{2} * log_2\left(\frac{1}{2}\right)\right)$$

$$Entropy(S) = 1.0000$$

b. Menentukan node cabang untuk parameter Low dengan data ’Sama’ Parameter yang dijadikan *node* cabang adalah parameter yang memiliki gain tertinggi, berikut perhitungan gain untuk setiap parameternya dengan menggunakan rumus 1.

1) Parameter Value

$$Gain(S, \text{Value}) = Entropy(S) - \sum_{i=1}^{n} \frac{|S_i|}{|S|} * Entropy\ (S_i)$$

$$Gain(S, \text{Value}) = 1.0000 - \left(\left(\frac{|1|}{|2|} * 0.0000\right) + \left(\frac{|1|}{|2|} * 0.0000\right)\right)$$

$$Gain(S, \text{Value}) = 1.0000$$

2) Parameter Volume

$$Gain(S, Volume) = Entropy(S) - \sum_{i=1}^{n} \frac{|S_i|}{|S|} * Entropy\ (S_i)$$

$$Gain(S, Volume) = 1.0000 - \left(\left(\frac{|0|}{|2|} * 0.0000\right) + \left(\frac{|2|}{|2|} * 1.0000\right)\right)$$

$$Gain(S, Volume) = 0.0000$$

3) Parameter Frek

$$Gain(S, Frek) = Entropy(S) - \sum_{i=1}^{n} \frac{|S_i|}{|S|} * Entropy\ (S_i)$$

$$Gain(S, Frek) = 1.0000 - \left(\left(\frac{|0|}{|2|} * 0.0000\right) + \left(\frac{|2|}{|2|} * 1.0000\right)\right)$$

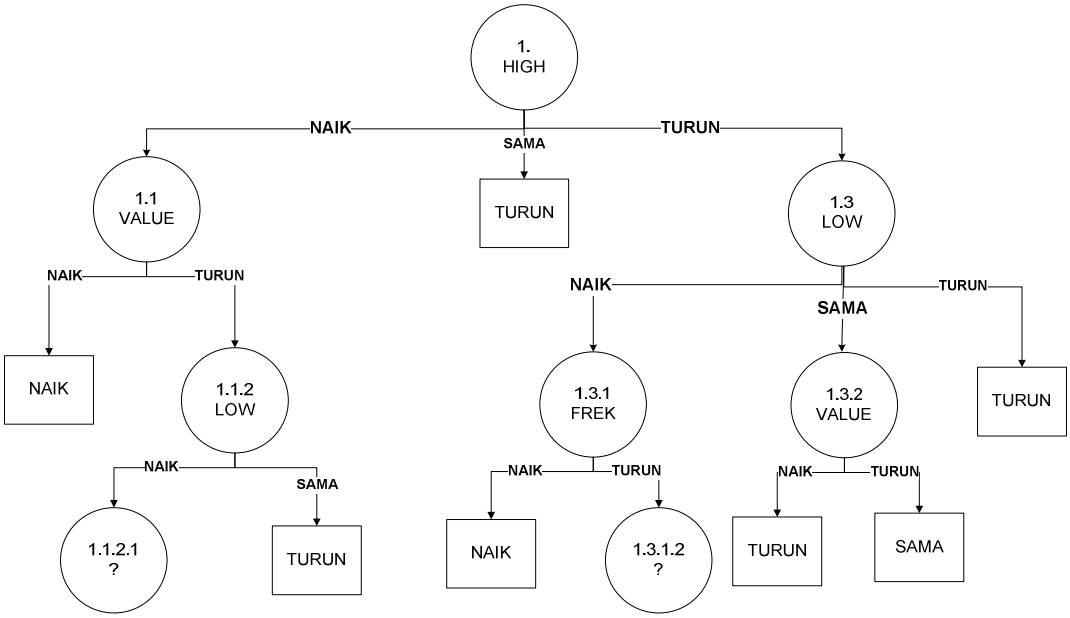
$$Gain(S, Frek) = 0.0000$$

Dari Hasil perhitungan dapat diketahui bahwa parameter Value memiliki nilai gain tertinggi sebesar 1,0000, maka parameter ini dapat menjadi node cabang dari parameter low =’Sama’. Ada dua data yang dimiliki oleh parameter Value yakni ‘Naik’, dan Turun’. Dari kedua data parameter tersebut sudah terklasifikasi semua, kondisi ‘naik’ sudah terklasifikasi dengan class ‘TURUN’ dan kondisi ‘turun terklasifikasi dengan Class ‘SAMA’. Adapun hasil perhitungan dari Node 1.3 dapat terlihat pada tabel 3.13

Tabel 3.13 Perhitungan Node 1.3.2

NODE	Parameter	Nama Parameter	Jumlah Data	Naik	Sama	Turun	Entropy	Gain
1.3.2		Low	2	0	1	1	1.0000	
	Value							1.0000
		Naik	1	0	0	1	0.0000	
		Turun	1	0	1	0	0.0000	
	Volume							0.0000
		Naik	0	0	0	0	0.0000	
		Turun	2	0	1	1	1.0000	
	Frek							0.0000
		Naik	0	0	0	0	0.0000	
		Turun	2	0	1	1	1.0000	

Adapun Pohon keputusan setelah perhitungan node 1.3.2 sampai tahap ini dapat dilihat pada gambar 3.7.



Gambar 3.7 Pohon keputusan node 1.3.2

7. Perhitungan Untuk Node 1.1.2.1

   a. Menentukan Entropy (Low=’Naik’). Untuk mencari *entropy* himpunan kasus dengan nilai parameter Low=’Naik’ maka dilakukan perhitungan dengan menggunakan rumus 2 sebagai berikut :

$$Entropy(S) = \left(\frac{2}{3} * log_2\left(\frac{2}{3}\right)\right) + \left(\frac{0}{3} * log_2\left(\frac{0}{3}\right)\right) + \left(\frac{1}{3} * log_2\left(\frac{1}{3}\right)\right)$$

$$Entropy(S) = 0.9183$$

   b. Menentukan node cabang untuk parameter Low dengan data ’Naik’ Parameter yang dijadikan *node* cabang adalah parameter yang memiliki gain tertinggi, berikut perhitungan gain untuk setiap parameternya dengan menggunakan rumus 1.

1) Parameter Volume

$$Gain(S, Volume) = Entropy(S) - \sum_{i=1}^{n} \frac{|S_i|}{|S|} * Entropy\ (S_i)$$

$$Gain(S, Volume) = 0.9183 - \left(\left(\frac{|1|}{|3|} * 0.0000\right) + \left(\frac{|2|}{|3|} * 0.0000\right)\right)$$

$$Gain(S, Volume) = 0{,}2516$$

2) Parameter Frek

$$Gain(S, Frek) = Entropy(S) - \sum_{i=1}^{n} \frac{|S_i|}{|S|} * Entropy\ (S_i)$$

$$Gain(S, Frek) = 0.9183 - \left(\left(\frac{|1|}{|3|} * 0.0000\right) + \left(\frac{|2|}{|3|} * 1.0000\right)\right)$$

$$Gain(S, Frek) = 0{,}2516$$

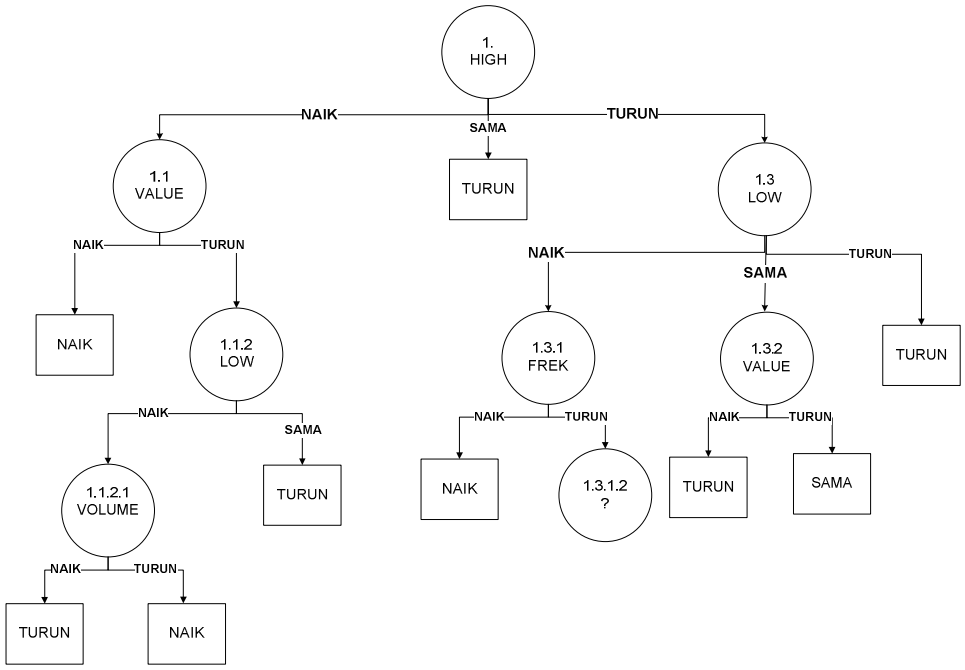
Dari Hasil perhitungan dapat diketahui bahwa Nilai parameter sama nilai gain tertinggi sebesar 0,2516 akan tetapi karena parameter Value data masukan terlebih dahulu sehingga parameter value yang dipilih maka parameter ini dapat menjadi node cabang dari parameter Low dengan data record 'Naik'.

Ada 2 data *record* pada node ini hanya 1 yang terklasifikasi, dengan data pada parameter Value yang berisi 'Naik' menyatakan *class NAIK* sedangkan data parameter *'Turun'* belum terklasifikasi sehingga perlu dihitung kembali untuk node 1.1.2.1.2. Adapun hasil perhitungan lengkapnya dapat dilihat pada tabel 3.14.

Tabel 3.14 Perhitungan Node 1.1.2.1

NODE	Parameter	Nama Parameter	Jumlah Data	Naik	Sama	Turun	Entropy	Gain
1.1.2.1		Low	3	2	0	1	0,9183	
	Volume							0,2516
		Naik	1	1	0	0	0,0000	
		Turun	2	1	0	1	1,0000	
	Frek							0,2516
		Naik	1	1	0	0	0,0000	
		Turun	2	1	0	1	1,0000	

Adapun Pohon keputusan setelah perhitungan node 1.1.2.1 sampai tahap ini dapat dilihat pada gambar 3.8.



Gambar 3.8 Pohon keputusan node 1.1.2.1

8. Perhitungan Untuk Node 1.3.1.2

   a. Menentukan Entropy (Frek=’Turun’). Untuk mencari *entropy* himpunan kasus dengan nilai parameter Value=’Turun’ maka

dilakukan perhitungan dengan menggunakan rumus 2 sebagai berikut :

$$Entropy(S) = \left(\frac{1}{3} * log_2\left(\frac{1}{3}\right)\right) + \left(\frac{0}{3} * log_2\left(\frac{0}{3}\right)\right) + \left(\frac{2}{3} * log_2\left(\frac{2}{3}\right)\right)$$

$$Entropy(S) = 0.9183$$

b. Menentukan node cabang untuk parameter Value dengan data ’Turun’ Parameter yang dijadikan *node* cabang adalah parameter yang memiliki gain tertinggi, berikut perhitungan gain untuk setiap parameternya dengan menggunakan rumus 1.

1) Parameter Value

$$Gain(S, value) = Entropy(S) - \sum_{i=1}^{n} \frac{|S_i|}{|S|} * Entropy\ (S_i)$$

$$Gain(S, Volume) = 0.9183 - \left(\left(\frac{|1|}{|3|} * 0.0000\right) + \left(\frac{|2|}{|3|} * 1.0000\right)\right)$$

$$Gain(S, Volume) = 0.9183$$

2) Parameter volume

$$Gain(S, volume) = Entropy(S) - \sum_{i=1}^{n} \frac{|S_i|}{|S|} * Entropy\ (S_i)$$

$$Gain(S, Frek) = 0.9183 - \left(\left(\frac{|1|}{|3|} * 0.0000\right) + \left(\frac{|2|}{|3|} * 0.0000\right)\right)$$

$$Gain(S, Frek) = 0{,}9183$$

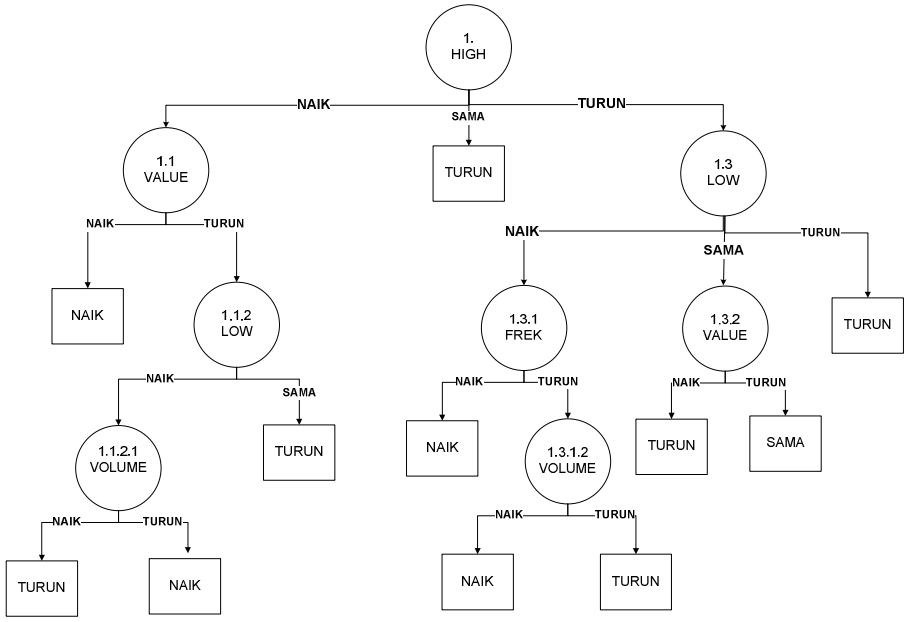
Dari Hasil perhitungan dapat diketahui bahwa parameter Value memiliki nilai gain tertinggi yang sama dengan volume sebesar 0,9183, maka parameter ini dapat menjadi node cabang dari parameter Frek =’Turun’. Ada dua data yang dimiliki oleh parameter value yakni ‘Naik’, dan Turun’. Dari kedua data

parameter tersebut, kondisi ‘naik’ yang telah terklasifikasi dengan class ‘NAIK’ sedangkan kondisi ‘Turun’ telah dengan class ‘TURUN’. Adapun hasil perhitungan dari Node 1.3.1.2 dapat terlihat pada tabel 3.15

Tabel 3.15 Perhitungan Node 1.3.1.2

NODE	Parameter	Nama Parameter	Jumlah Data	Naik	Sama	Turun	Entropy	Gain
1.3.1.2		Frek	3	1	0	2	0.9183	
	Value							0.9183
		Naik	1	1	0	0	0.0000	
		Turun	2	0	0	2	0.0000	
	Volume							0.9183
		Naik	1	1	0	0	0.0000	
		Turun	2	0	0	2	0.0000	

Adapun Pohon keputusan setelah perhitungan node 1.1 sampai tahap ini dapat dilihat pada gambar 3.9.



Gambar 3.9 Pohon keputusan node 1.3.1.2 (UTUH)

Setelah pohon keputusan secara utuh maka kita bias membuat aturan prediksi berdasarkan hasil pembentukan pohon keputusan tersebut. Adapun rules aturan prediksi yang bias didapat dari pembentukan pohon keputusan pada contoh kasus yang telah dibuat sebelumnya adalah sebagai berikut :

1. Jika HIGH NAIK
   VALUE NAIK
   Diprediksi Saham NAIK
2. Jika HIGH NAIK
   VALUE TURUN
   LOW NAIK
   Diprediksi Saham TURUN
3. Jika HIGH NAIK
   VALUE TURUN
   LOW NAIK
   VOLUME NAIK
   Diprediksi Saham NAIK
4. Jika HIGH NAIK
   VALUE TURUN
   LOW NAIK
   VOLUME TURUN
   Diprediksi Saham NAIK
5. Jika HIGH SAMA
   Diprediksi Saham TURUN
6. Jika HIGH TURUN
   LOW NAIK
   FREK NAIK
   Diprediksi Saham NAIK
7. Jika HIGH TURUN

LOW NAIK
FREK TURUN
VALUE NAIK
Diprediksi Saham NAIK

8. Jika HIGH TURUN
   LOW NAIK
   FREK TURUN
   VALUE TURUN
   Diprediksi Saham TURUN
9. Jika HIGH TURUN
   LOW SAMA
   VALUE NAIK
   Diprediksi Saham TURUN
10. Jika HIGH TURUN
    LOW SAMA
    VALUE TURUN
    Diprediksi Saham SAMA
11. Jika HIGH TURUN
    LOW TURUN
    Diprediksi Saham TURUN

Setelah kondisi ini maka kecenderungan prediksi saham akan dapat terlihat dari data masukan yang menjadi nilai acuan masukan dengan kecenderungan bahwa dari data testing yang menjadi masukan selanjutnya dapat menentukan fluktuasi tren pada pembentukan harga dihari berikutnya.

Dalam perhitungan Akurasi dari prediksi tidak dibahas langsung dalam penelitian ini baik dari aplikasi atau pun skala perhitungan didalam dokumentasi penelitian, akan tetapi dari rule yang terbentuk di coba ke 30 sample record

setelahnya atau data diluar aturan dengan nilai akurasi dihitung dari hasil yang didapat dalam simulasi prediksi fluktuasi saham pada aplikasi, berikut simulasi dan perhitungan dalam dokunentasinya :

1. Sample data

Dari sample data ini dimasukan sesuai perubahan yang terjadi dengan menggunakan aturan dalam aplikasi dari hasil dari Data Mining aplikasi

Tabel 3.16 30 *sample data Records* kedua tabel STOCK_TRADES

Code	Date	Prev	High	Low	Close	Change	Value	Volume	Frek	Index
ISAT	20040318	3,500	3,575	3,500	3,500	0	26,127,486,976	7,410,000	391	250
ISAT	20040319	3,500	3,850	3,525	3,800	300	111,409,799,168	29,770,000	1,128	271.429
ISAT	20040323	3,800	3,875	3,650	3,825	25	58,351,714,304	15,455,000	546	273.214
ISAT	20040324	3,825	3,850	3,750	3,750	-75	22,750,699,520	6,001,500	320	267.857
ISAT	20040325	3,750	3,775	3,700	3,725	-25	25,901,400,064	6,963,500	233	266.071
ISAT	20040326	3,725	3,750	3,650	3,675	-50	10,179,862,528	2,762,500	205	262.5
ISAT	20040329	3,675	3,700	3,675	3,675	0	1,631,837,440	443,000	215	262.5
ISAT	20040330	3,675	3,775	3,700	3,775	100	12,158,087,168	3,237,500	245	269.643
ISAT	20040331	3,775	3,875	3,800	3,850	75	48,793,276,416	12,688,500	499	275
ISAT	20040401	3,850	3,850	3,800	3,825	-25	24,841,775,104	6,491,500	192	273.214
ISAT	20040402	3,825	3,850	3,750	3,750	-75	44,418,011,136	11,767,000	365	267.857
ISAT	20040406	3,750	3,925	3,800	3,925	175	72,068,841,472	18,635,000	732	280.357
ISAT	20040407	3,925	4,250	3,925	4,250	325	121,989,079,040	29,735,000	989	303.571
ISAT	20040408	4,250	4,350	4,150	4,300	50	88,768,798,720	20,691,500	620	307.143
ISAT	20040412	4,300	4,275	4,150	4,200	-100	28,152,475,648	6,700,000	292	300
ISAT	20040413	4,200	4,250	4,125	4,150	-50	48,986,898,432	11,691,000	327	296.429
ISAT	20040414	4,150	4,150	4,050	4,100	-50	43,592,974,336	10,660,500	470	292.857
ISAT	20040415	4,100	4,125	4,050	4,075	-25	20,422,025,216	4,996,000	237	291.071
ISAT	20040416	4,075	4,125	4,075	4,125	50	13,162,274,816	3,206,500	187	294.643
ISAT	20040419	4,125	4,275	4,125	4,250	125	77,511,434,240	18,293,000	466	303.571
ISAT	20040420	4,250	4,325	4,225	4,275	25	91,812,487,168	21,438,000	571	305.357
ISAT	20040421	4,275	4,650	4,250	4,525	250	159,741,067,264	35,589,500	1,063	323.214
ISAT	20040422	4,525	4,525	4,375	4,425	-100	40,316,088,320	9,098,000	410	316.071
ISAT	20040423	4,425	4,450	4,325	4,400	-25	30,922,637,312	7,033,500	336	314.286
ISAT	20040426	4,400	4,450	4,350	4,450	50	31,518,300,160	7,177,500	217	317.857
ISAT	20040427	4,450	4,525	4,425	4,475	25	31,108,249,600	6,948,000	302	319.643
ISAT	20040428	4,475	4,475	4,400	4,425	-50	15,495,100,416	3,505,500	179	316.071
ISAT	20040429	4,425	4,400	4,150	4,175	-250	107,094,409,216	25,329,500	1,255	298.214
ISAT	20040430	4,175	4,150	3,950	3,975	-200	113,598,963,712	28,365,500	1,604	283.929
ISAT	20040504	3,975	4,150	4,025	4,125	150	95,086,551,040	23,172,500	956	294.643
ISAT	20040505	4,125	4,150	4,000	4,025	-100	36,140,974,080	8,925,000	487	287.5

2. Hasil Uji Coba

Dari hasil uji coba menggunakan salah satu form didalam aplikasi untuk memprediksi fluktuasi harga saham dihasilkan perhitungan sebagai berikut :

a. Hasil Prediksi Yang salah ada 4 record, maka persentasi error dari 30 sample data uji coba aplikasi adalah sebagai berikut :

$$\text{Error} = \frac{Jumlah\ data\ error}{Jumlah\ seluruh\ data} X\ 100\ \%$$

$$\text{Error} = \frac{4}{30} X\ 100\ \%$$

$$\text{Error} = 13.34\ \%$$

b. Hasil Prediksi yang memiliki akurasi benar sesuai dengan perubahan saham ada 26 record, maka persentasi Akurasi dari 30 sample data uji coba aplikasi adalah sebagai berikut :

$$\text{Akurasi} = \frac{Jumlah\ data\ tepat}{Jumlah\ seluruh\ data} X\ 100\ \%$$

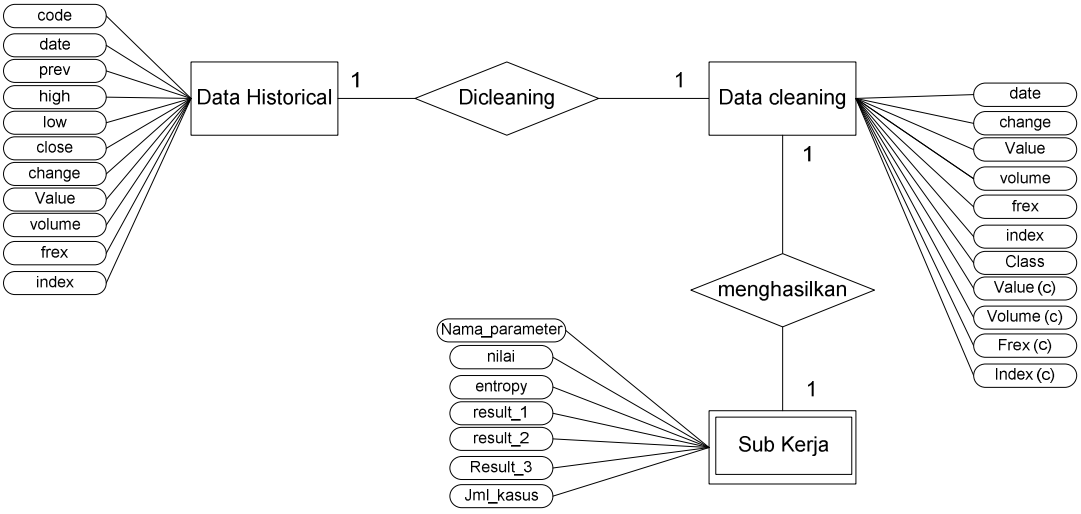
$$\text{Akurasi} = \frac{26}{30} X\ 100\ \%$$

$$\text{Akurasi} = 86{,}67\ \%$$

Dari Hasil uji coba terhadap aplikasi memiliki ketepatan dalam memprediksi dari 30 sample data yang diuji coba diluar sample data pembentuk aturan, memiliki akurasi 86,67 % dan memiliki persentasi error 13,34 %

3.1.4 Analisis Basis Data

Struktur logika dari basis data dapat digambarkan dalam sebuah grafik dengan menggunakan *Entity Relationship Diagram* (ERD). ERD merupakan hubungan antara entitas yang digunakan dalam sistem untuk menggambarkan hubungan antara entitas atau struktur data dan relasi antar file. Komponen utama pembentukan ERD yaitu *Entity* (entitas) dan *Relation* (relasi) sehingga dalam hal ini ERD merupakan komponen-komponen himpunan entitas dan himpunan relasi yang dideskripsikan lebih jauh melalui sejumlah atribut-atribut (*property*) yang menggambarkan seluruh fakta dari sistem yang ditinjau. Adapun ERD dari aplikasi yang akan dibuat adalah sebagai berikut :



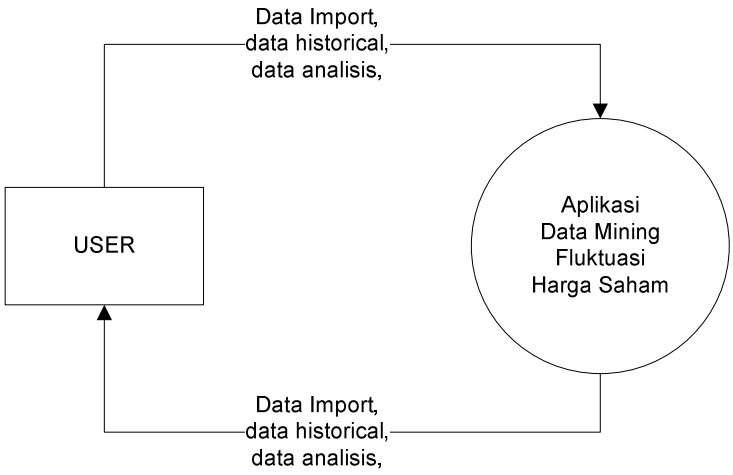
Gambar 3.10 *Entity Relationship Diagram* (ERD)

3.1.5 Analisis Kebutuhan Fungsional

3.1.5.1 Diagram Konteks

Diagram konteks atau disebut juga dengan model sistem *fundamental* merepresentasikan seluruh elemen sistem sebagai sebuah *bubble* tunggal dengan data *input, output* yang ditunjukkan oleh anak panah yang masuk dan keluar secara berurutan.

Berdasarkan hasil analisis, maka diagram konteks dari aplikasi yang akan dibangun dapat dilihat pada gambar dibawah ini :



Gambar 3.11 Diagram Konteks

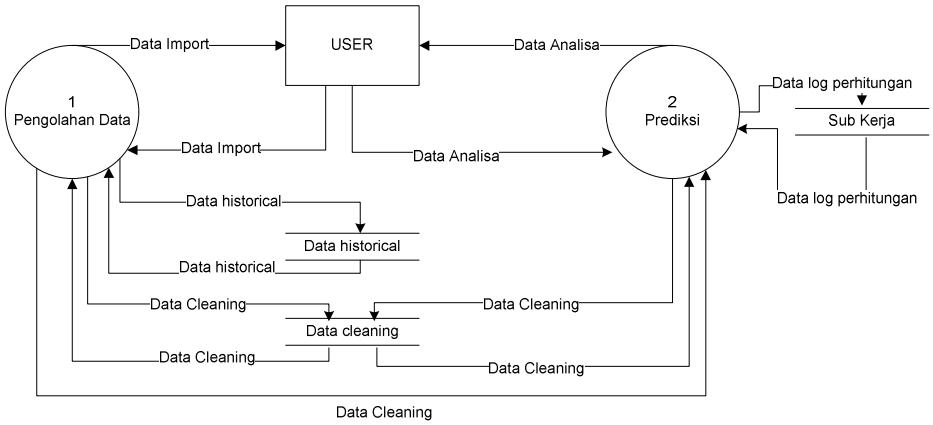
3.1.5.2 *Data Flow Diagram* (DFD)

Data *flow* diagram merupakan model dari sistem untuk menggambarkan pembagian sistem ke model yang lebih kecil. Salah satu keuntungan menggunakan DFD adalah memudahkan pemakai yang kurang menguasai bidang komputer dapat mengerti sistem yang akan dijalankan.

Berikut *data flow diagram* untuk aplikasi yang akan dibuat :

3.1.5.2.1 DFD Level 1

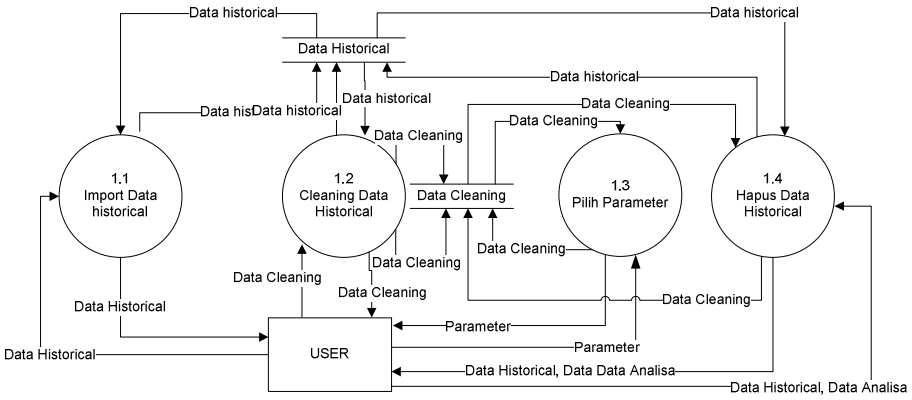
Proses ini diakses oleh user yang merupakan investor yang telah memiliki keygen dari aplikasi, dimana proses yang terjadi adalah proses login user kemudian proses pengolahan data master, serta proses analisis.



Gambar 3.12 DFD Level 1

3.1.5.2.2 DFD Level 2 Proses Pengolahan Data

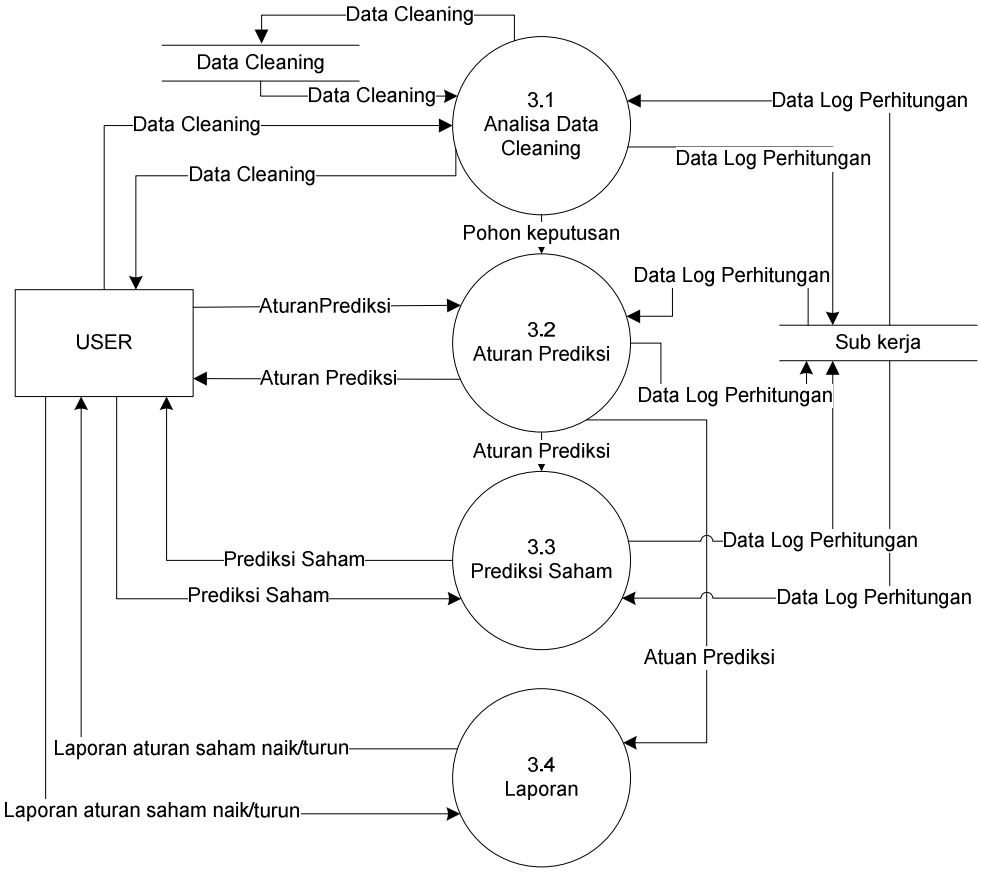
Proses ini diakses oleh user yang merupakan investor yang telah memiliki keygen dari aplikasi, dimana proses yang terjadi adalah proses import data dan hapus data dari hasil keluaran data T_Stocktrades yang telah di cleaning.



Gambar 3.13 DFD Level 2 Proses pengolahan Data

3.1.5.2.3 DFD Level 2 Proses *Prediksi*

Proses ini diakses oleh user yang merupakan investor yang telah memiliki keygen dari aplikasi, dimana proses yang terjadi adalah proses menganalisis dengan berproses dengan hasil keluaran berupa data.



Gambar 3.14 DFD Level 2 Proses Prediksi

3.1.5.3 Spesifikasi Proses

Spesifikasi proses digunakan untuk menggambarkan proses model aliran yang terdapat pada DFD. Spesifikasi proses dari DFD yang telah dibuat dapat dijelaskan pada tabel berikut :

Tabel 3.17 Spesifikasi Proses

	Proses	Keterangan
1	No Proses	1
	Nama Proses	Pengolahan Data
	Source (Sumber)	*Mengimport* Data, Mencleaning Data, Menentukan Parameter, dan Menghapus Data
	Input	Data Historical
	Output	Info Historical,Info cleaning
	Destination (Tujuan)	User
	Logika Proses	1 Status program dijalankan akan tampil menu utama 2. User Mengklik tombol pengolahan data yang berisikan proses *import*, proses *cleaning, pilih parameter* dan proses hapus data 3. User memilih file dengan menggunakan fasilitas icon 'browse' untuk memilih dan mencari file 4. User melakukan *import* data pada form pengolahan data setelah data dipilih pada icon browser 5. User tidak dapat mengimport data jika tidak ada data yang dipilih 6. User dapat mengcleaning data didalam tampilan satu form dengan pengimport data 7. User dapat mengolah data historical menjadi data cleaning siap analisis setelah mengimport data terlebih dahulu 8. Sistem akan menampilkan tombol pilih parameter setelah proses cleaning selesai 9. User melakukan pemilihan parameter dengan mengklik tombol parameter setelah proses cleaning 10. system akan menampilkan file form pilih parameter. 11. Setelah selesai pilih parameter system akan menutup form pilih parameter kemudian kembali ke form pengolahan data dan melanjutkan langkah selanjutnya, baik akan menghapus atau menganalisi dengan kembali ke menu utama 11. User tidak bisa menghapus data apabila user belum mengimport data atau mengcleaning data

2	No Proses	2
	Nama Proses	Prediksi
	Source (Sumber)	Menganalisis Data, View Hasil Analisis
	Input	Data Cleaning, Parameter
	Output	Data analisis
	Destination (Tujuan)	User
	Logika Proses	1. Status User telah mengimport dan mengcleaning data di pengolahan data' 2. Status parameter telah terpilih proses analisa akan dapat menganalisis, apabila parameter belum dipilih akan meminta pilih parameter terlebih dahulu 3. User memproses data setelah parameter dipilih dengan menggunakan algoritma C 4.5 (Algortitma DT dalam penerapan Metode Classification). 4. Status data setelah dianalisa user akan mendapatkan info tren dari hasil keluaran node penentu arah perubahan fluktuasi saham 5. User dapat mengetahui tren fluktuasi harga saham dari aturan yang keluar setelah data diproses.
3	No Proses	2.1
	Nama Proses	import Data historical
	Source (Sumber)	User
	Input	Data Historical berupa file Excel
	Output	Data historical
	Destination (Tujuan)	User
	Logika Proses	1. User memilih menu Import di Menu Utama 2. Aplikasi merespon pilihan User dengan menampilkan form import data yang berada di form pengolahan data 3. User mencari lokasi file menggunakan aplikasi dengan fasilitas icon browser, 4. Setelah File dipilih berupa File *.xls atau Excel user mengklik proses import untuk mengimport data 5. Selesai Proses Import data User mendapat tampilan dari aplikasi berupa tabel berisikan data yang telah diimport berupa *fìeld* yang masih belum siap dianalisa 6. Selesai proses import data icon 'proses' untuk mengcleaning data.

4	No Proses	2.2
	Nama Proses	Cleaning Data
	Source (Sumber)	User
	Input	Data Historical
	Output	Data Cleaning
	Destination (Tujuan)	User
	Logika Proses	1. Status user telah mengimport akan muncul icon ‘proses’ di form pengolahan data 2. User memilih icon ‘proses’ untuk melakukan proses Cleaning data 3. Aplikasi akan merespon klik icon ‘proses’ dengan melakukan proses cleaning data 4. Aplikasi akan melakukan proses cleaning dengan langkah-langkah sebagai berikut : a. Inisialisasi data pada tabel data historical dengan mengecheck keberadaan data pada tabel tersebut. b. Proses Selanjutnya system akan menghitung jumlah record c. Sistem akan menghapus *field* yang tidak bisa dijadikan parameter akan dihapus kolomnya baik data didalamnya samapi tersisa hanya field yang dapat dijadikan parameter penentu node. 5. Setelah selesai proses cleaning selesai akan muncul informasi bahwa data telah dicleaning dan muncul info data hasil cleaning pada tabel dibawah data tabel historical
5	No Proses	2.3
	Nama Proses	Pilih Parameter
	Source (Sumber)	User
	Input	Data Cleaning
	Output	Pilih Parameter Penentu
	Destination (Tujuan)	User
	Logika Proses	1. Status user telah mengimport akan muncul icon ‘pilih parameter’ di form pengolahan data 2. User Apabila mengklik pilih parameter akan muncul form pilih parameter 3. User akan ditampilkan 2 *field* sebelah kiri semua field dari data historical akan tetapi hanya dapat dipilih hanya field ’HIGH’,’LOW’,’VALUE’,VOLUME’,’FREK’, SELAIN ITU tidak bisa dipilih kecuali change akan otomatis terpilih 4. Setelah memilih dengan memindahkan field ke field sebelah keanan akan muncul tombol

		save untuk mensave pilihan parameter sekaligus akan kembali kem form pengolahan data
6	No Proses	2.4
	Nama Proses	Hapus Data
	Source (Sumber)	User
	Input	Data historical, data cleaning
	Output	Data historical, data cleaning
	Destination (Tujuan)	User
	Logika Proses	1. Status user telah mengimport data atau mengcleaning data, akan muncul icon hapus di form pengolahan data 2. User memilih icon 'hapus' untuk melakukan proses hapus data 3. Aplikasi akan merespon klik icon 'hapus' dengan memberikan notifikasi untuk meyakinkan kembali akan melakukan proses import data 4. aplikasi akan merespon apabila notifkasi telah dijawab ya dengan menghapus atau menghilangkan data baik data yang sudah atau sebelum dicleaning 5. Setelah proses penghapusan selesai aplikasi akan kembali seperti semula sebelum diimport data dengan posisi icon dan icon proses cleaning akan 'deaktif' 6. User hanya bisa menghapus data apabila oser telah mengimport data atau mengcleaning
7	No Proses	3.1
	Nama Proses	Analisis Data
	Source (Sumber)	User
	Input	Data Historical
	Output	Data analisis
	Destination (Tujuan)	User
	Logika Proses	1. User menekan Icon Analisis untuk memproses data historical menjadi data analisis 2. Aplikasi akan menganailisis data dan menampilkan aturan keluaran dari pelajaran algoritma dengan langkah-langkah sebagai berikut : a. Sistem akan menginisialisasi data b. Sistem Menghitung Nilai Gain untuk Seluruh parameter c. Nilai Gain yang tertinggi akan dijadikan root dari pohon keputusan

		d. Selanjutnya akan dilakukan perhitungan yang sama untuk seluruh parameter yang masih belum terklasifikasi sampai setiap parameter terklasifikasi pada class target
8	No Proses	3.2
	Nama Proses	Mementukan Aturan Prediksi
	Source (Sumber)	User
	Input	Data Analisis
	Output	Data Analisis
	Destination (Tujuan)	User
	Logika Proses	1. Status Data Telah Dianalisis 2. User akan melihat hasil analisis dengan muncul aturan yang keluar atau pohon keputusan setelah menganalisa dengan algoritma C.45 3. Sistem akan memberikan keluaran aturan dari factor penentu yang keluar dari perhitungan gain dan entropy setelah diproses algoritma c.45
9	No Proses	3.3
	Nama Proses	Prediksi Fluktuasu Harga Saham
	Source (Sumber)	User
	Input	Data Analisis
	Output	Data Analisis
	Destination (Tujuan)	User
	Logika Proses	1. Status Aturan telah dibuat 2. User memilih content parameter di form Prediksi Prameter 3. Sistem akan memberikan keluaran berupa parameter baru yang perlu dipilih kembali sama sampai menghasilkan kesimpulan fluktuasi harga saham
10	No Proses	3.4
	Nama Proses	Laporan
	Source (Sumber)	User
	Input	Data Cleaning, Aturan Prediksi
	Output	Laporan Aturan Prediksi
	Destination (Tujuan)	User
	Logika Proses	1. Status Aturan Prediksi Telah diproses 2. User memilih icon Laporan 3. Sistem akan memberikan keluaran berupa laporan aturan saham naik atau saham turun

3.1.5.4 Kamus Data

Kamus data berfungsi untuk menjelaskan semua data yang digunakan di dalam sistem. Berikut adalah kamus data untuk aplikasi yang akan dibuat :

Tabel 3.18 Kamus Data Historical

Code	[A..Z]
Date	[0..9]
prev	[0..9]
High	[0..9]
Low	[0..9]
Close	[0..9]
change	[0..9]
Value	[0..9]
Volume	[0..9]
Frex	[0..9]
Index	[0..9]

Tabel 3.19 Kamus Data Cleaning

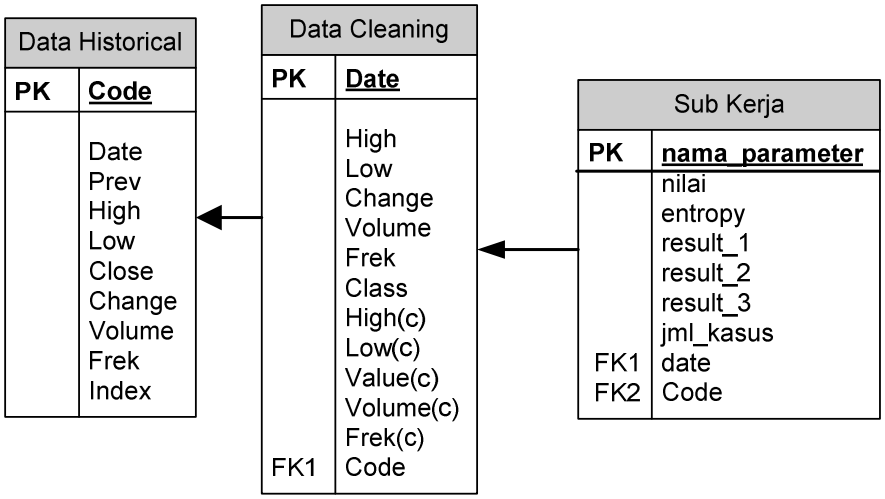
Date	
High	[0..9]
Low	[0..9]
Change	[0..9]
Value	[0..9]
Volume	[0..9]
Frex	[0..9]
Highc	[a..z | A..Z]
Lowc	[a..z | A..Z]
Class	[a..z | A..Z]
Valuec	[a..z | A..Z]
Volumec	[a..z | A..Z]
Frexc	[a..z | A..Z]
Indexc	[a..z | A..Z]

Tabel 3.20 Kamus Data Sub Kerja

Nama_parameter	[a..z | A..Z]
Nilai	[a..z | A..Z]
Entropy	[0..9]
Result_1	[0..9]
Result_2	[0..9]
Result_3	[0..9]
Jml_kasus	[0..9]

3.1.5.5 Skema Relasi

Proses relasi antar *file* merupakan gabungan antar *file* yang memiliki *primarykey* (kunci utama) yang sama, sehingga *file-file* tersebut menjadi satu kesatuan yang dihubungkan oleh *field* (atribut) kunci tersebut. Pada proses ini elemen-elemen data di kekompokan menjadi satu *file database* beserta entitas dan hubungannya. Skema relasi dari aplikasi *data mining* yang akan dibuat adalah sebagai berikut :



Gambar 3.15 Skema Relasi

3.1.5.6 Struktur Tabel

Berikut adalah tabel-tabel yang terdapat dalam basis data yang digunakan dalam aplikasi yang akan dibangun.

Tabel 3.21 Struktur Tabel data historical

Nama Field	Type	Length	Kata kunci	Keterangan
Code	*Varchar*	4	Primary Key	Not Null
Date	*Date*			Not Null
Prev	*Smallint*			Not Null
High	*Smallint*			Not Null
Low	*Smallint*			Not Null
Close	*Smallint*			Not Null
Change	*Smallint*			Not Null
Volume	*Double*			Not Null
Value	*Double*			Not Null
Frex	*Smallint*			Not Null
Index	*Float*			Not Null

Tabel 3.22 Struktur Tabel data Cleaning

Nama Field	Type	Length	Kata kunci	Keterangan
Date	*Date*		Primary Key	Not Null
High	*Int*			Not Null
Low	*Int*			Not Null
Change	*Int*			Not Null
Value	*Int*			Not Null
Volume	*Int*			Not Null
Frex	*Int*			Not Null
Index	*Float*			Not Null
Class	*Varchar*	5		Not Null
Highc	*Varchar*	5		Not Null
Lowc	*Varchar*	5		Not Null

Valuec	*Varchar*	5		Not Null
Volumec	*Varchar*	5		Not Null
Frexc	*Varchar*	5		Not Null

Tabel 3.23 Struktur Tabel data Sub Kerja

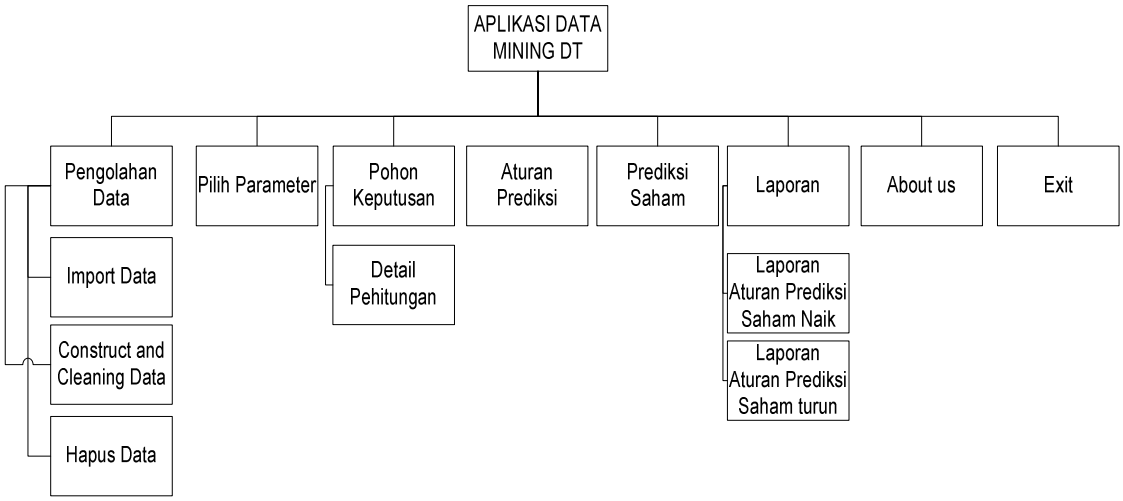
Nama Field	Type	Length	Kata kunci	Keterangan
Nama_parameter	*Varchar*	10	Primary Key	Not Null
Nilai	*Varchar*	5		Not Null
Entropy	*Float*			Not Null
Result_1	*Varchar*	3		Not Null
Result_2	*Varchar*	3		Not Null
Result_3	*Varchar*	3		Not Null
Jml_Kasus	*Int*			Not Null

3.2 Perancangan Arsitektur

Perancangan arsitektur merupakan perancangan yang dibuat sebelum program aplikasi dibuat. Dengan perancangan arsitektur akan mempermudah dalam proses pembangunan aplikasi *data mining* ini. Perancangan perangkat lunak ini mengikuti konsep yang telah dilakukan pada tahap analisis sistem.Perancangan arsitektur ini terdiri dari perancangan struktur menu, perancangan antar muka, dan jaringan semantik.

3.2.1 Perancangan Struktur Menu

Dalam perancangan sebuah aplikasi dibutuhkan struktur menu yang berisikan menu dan submenu yang berfungsi untuk memudahkan *user* dalam menggunakan aplikasi tersebut. Berikut ini digambarkan mengenai stuktur menu dalam aplikasi ini :



Gambar 3.16 Perancangan Struktur Menu

3.2.2 Perancangan Antarmuka

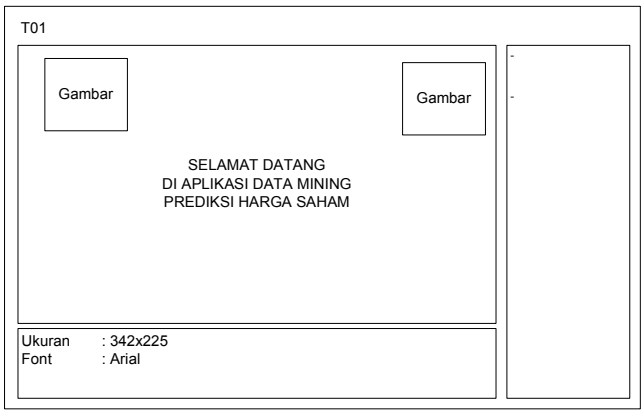
Dalam membuat aplikasi dibutuhkan tampilan yang sangat menarik, dimana dalam perancangan antar muka ini akan dituangkan beberapa imajinasi dalam merancang tampilan program dan tampilan pesan yang nantinya akan dipakai untuk membuat aplikasi sesungguhnya. Perancangan antar muka ini terdiri dari perancangan tampilan program, dan perancangan tampilan pesan.

3.2.2.1 Perancangan Tampilan Program

Perancangan tampilan program dari aplikasi data mining yang akan dibuat adalah sebagai berikut :

3.2.2.1.1 Tampilan Awal Aplikasi

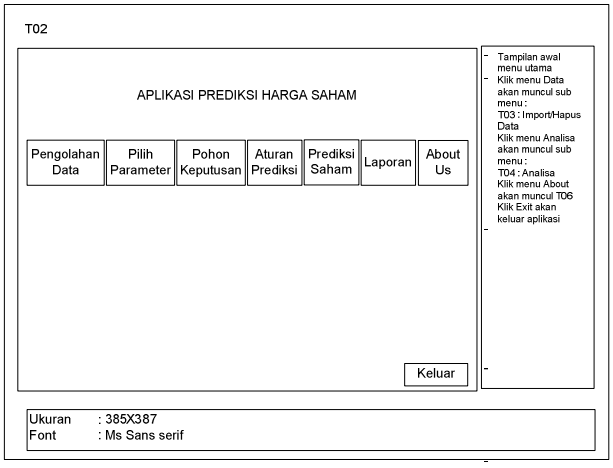
Berikut ini adalah gambar perancangan tampilan awal dari aplikasi yang akan dibuat ketika *user* menjalankan program.



Gambar 3.17 Tampilan Splash Screen (tampilan awal Aplikasi)

3.2.2.1.2 Tampilan Menu Utama

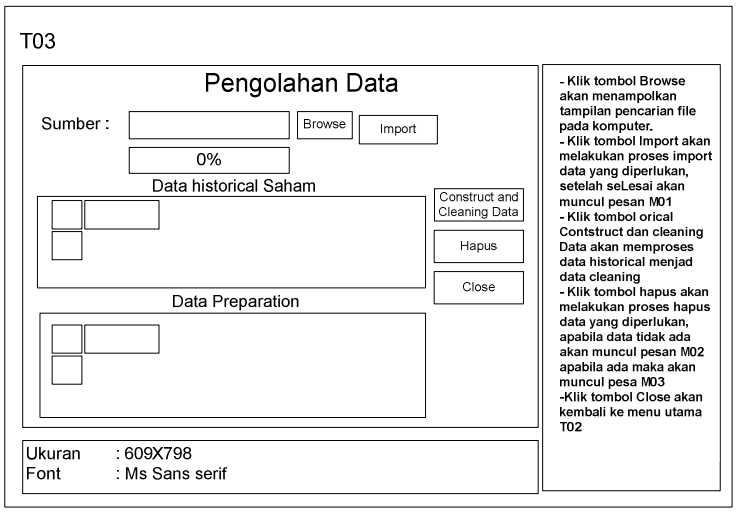
Berikut ini adalah gambar perancangan menu utama yang akan muncul.



Gambar 3.18 Tampilan Menu Utama

3.2.2.1.3 Tampilan Menu Pengolahan Data

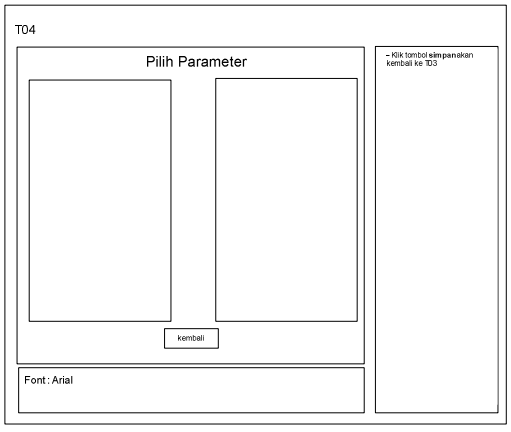
Berikut ini adalah gambar perancangan sub menu user/operator yang terdapat pada menu utama.



Gambar 3.19 Tampilan Pengolahan Data

3.2.2.1.4 Tampilan Pilih Parameter

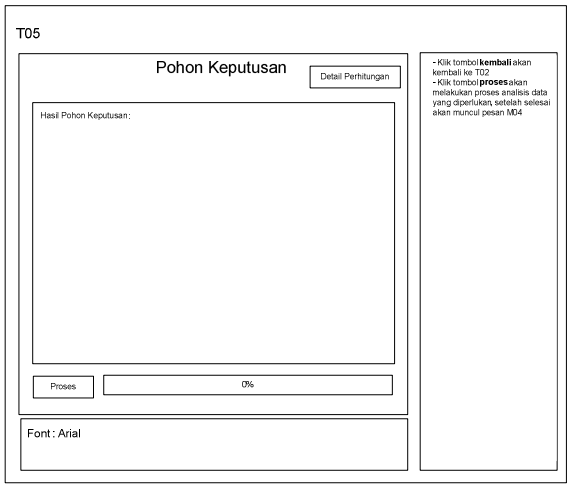
Berikut ini adalah gambar perancangan Pilih Parameter *DT* yang terdapat pada menu utama.



Gambar 3.20 Tampilan Pilih Parameter

3.2.2.1.5 Tampilan Analisa Pohon Keputusan

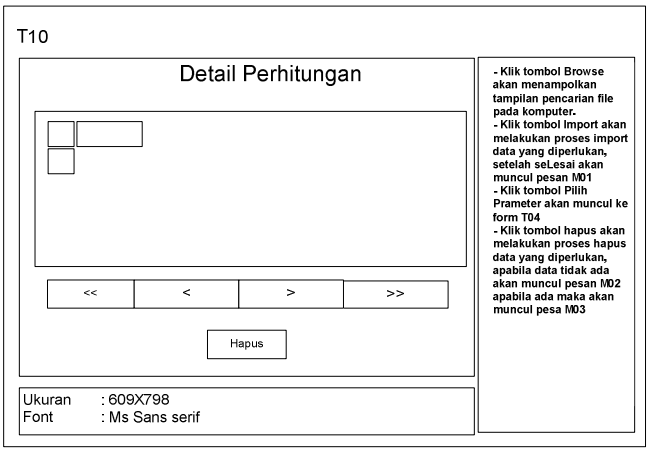
Berikut ini adalah gambar perancangan analisa Pohon Keputusan *DT* yang terdapat pada menu utama.



Gambar 3.21 Tampilan Analisa Pohon Keputusan

3.2.2.1.6 Tampilan Detail Perhitungan

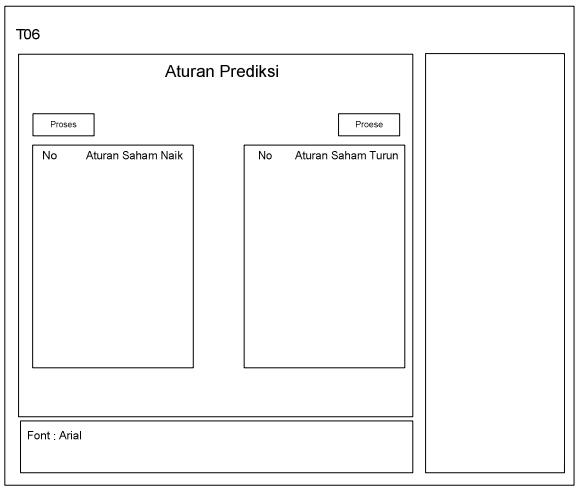
Berikut ini adalah gambar perancangan Detail Perhitungan *DT* yang terdapat pada sub menu Pohon keputusan.



Gambar 3.22 Tampilan Detail Perhitungan

3.2.2.1.7 Tampilan Aturan Prediksi

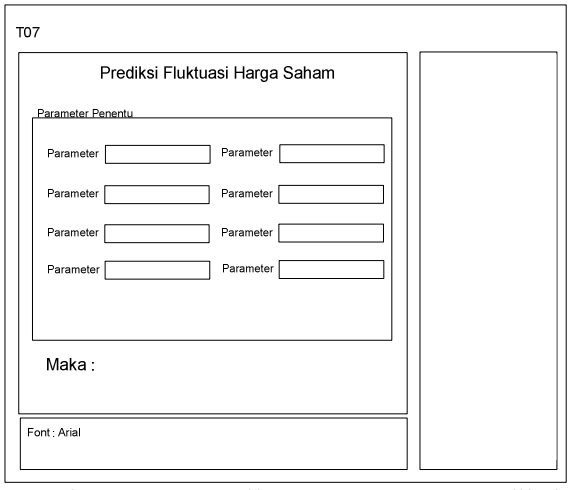
Berikut ini adalah gambar perancangan Aturan Prediksi analisa *DT* yang terdapat pada menu Utama.



Gambar 3.23 Tampilan Form Aturan Prediksi

3.2.2.1.8 Tampilan Prediksi Fluktuasi Saham

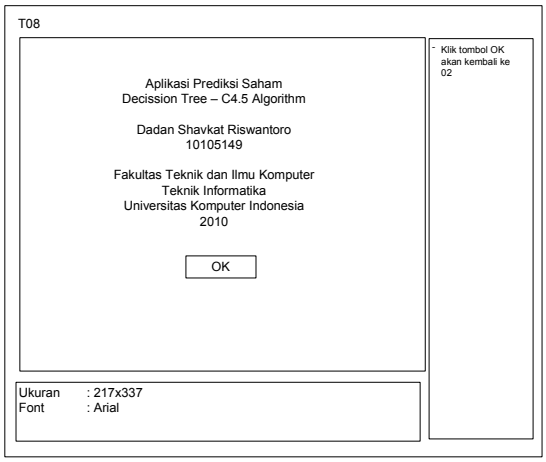
Berikut ini adalah gambar perancangan Prediksi Saham *DT* yang terdapat pada menu Utama.



Gambar 3.24 Tampilan Form Aturan Prediksi

3.2.2.1.9 Tampilan *About*

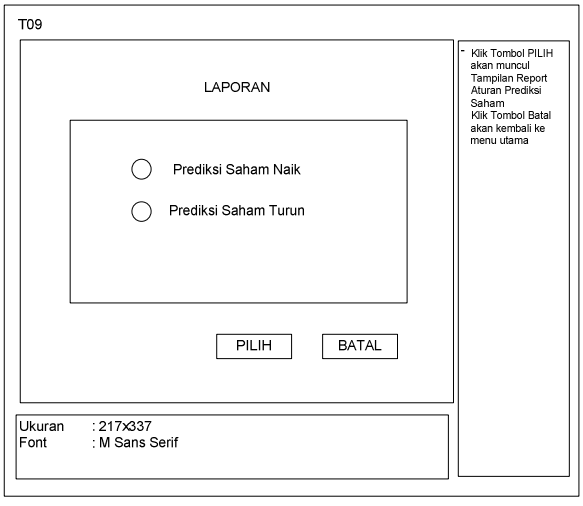
Berikut ini adalah gambar perancangan menu *about us* pada menu utama.



Gambar 3.25 Tampilan Menu *About us*

3.2.2.1.10 Tampilan Sub Menu Laporan

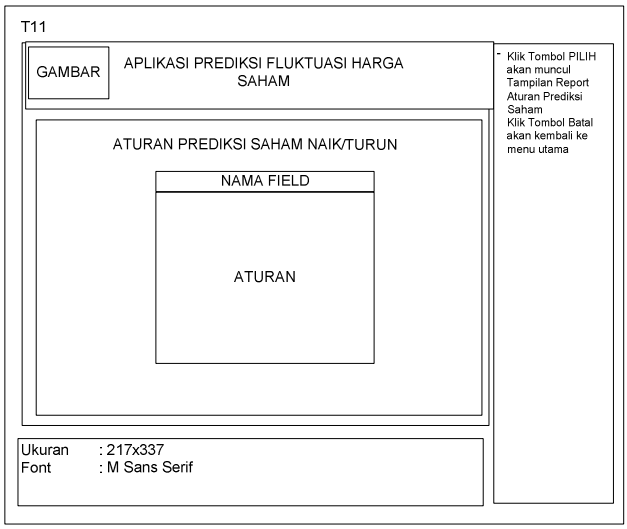
Berikut ini adalah gambar perancangan menu Laporan pada menu utama.



Gambar 3.26 Tampilan Menu Laporan

3.2.2.1.11 Tampilan Laporan

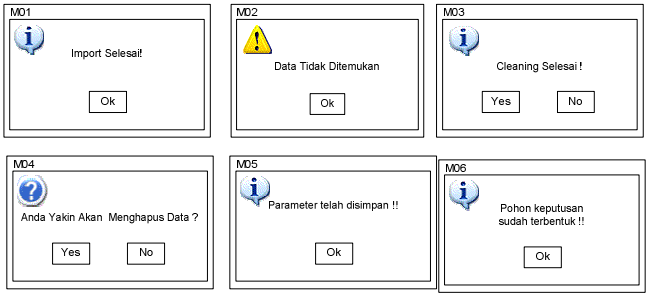
Berikut ini adalah gambar perancangan Desain Laporan pada sub menuLaporan .



Gambar 3.27 Tampilan Laporan

3.2.2.2 Perancangan Pesan

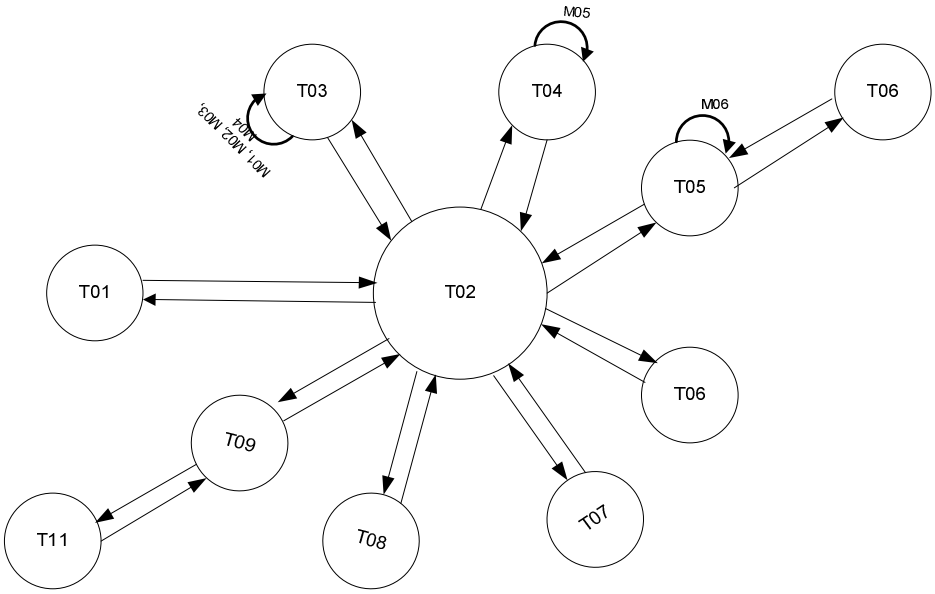
Rancangan pesan dari aplikasi *data mining* Prediksi fluktuasi harga saham yang akan dibuat adalah sebagai berikut :



Gambar 3.28 Tampilan Perancangan Pesan

3.2.3 Jaringan Semantik

Berikut adalah jaringan semantik dari aplikasi yang akan dibuat untuk menunjukkan keterhubungan antar objek perancangan.

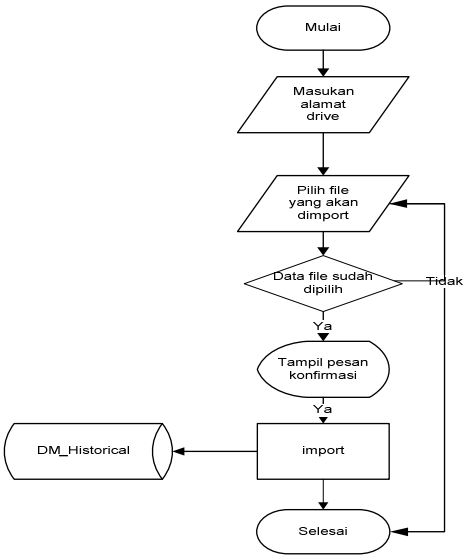


Gambar 3.29 Jaringan Semantik

3.2.4 Perancangan Prosedural

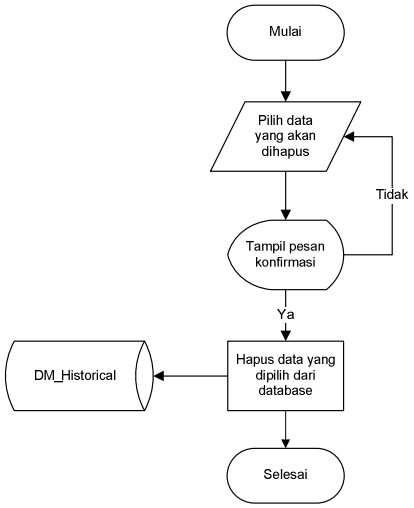
Perancangan prosedural merupakan perancangan yang dilakukan untuk menetapkan detail algoritma yang akan dinyatakan ke dalam suatu program. Adapun perancangan prosedural untuk aplikasi yang akan dibangun sebagai berikut :

1. *Flowchart* import data, merupakan prosedur yang terjadi ketika *user* mengimport data untuk dimasukkan kedalam *database*



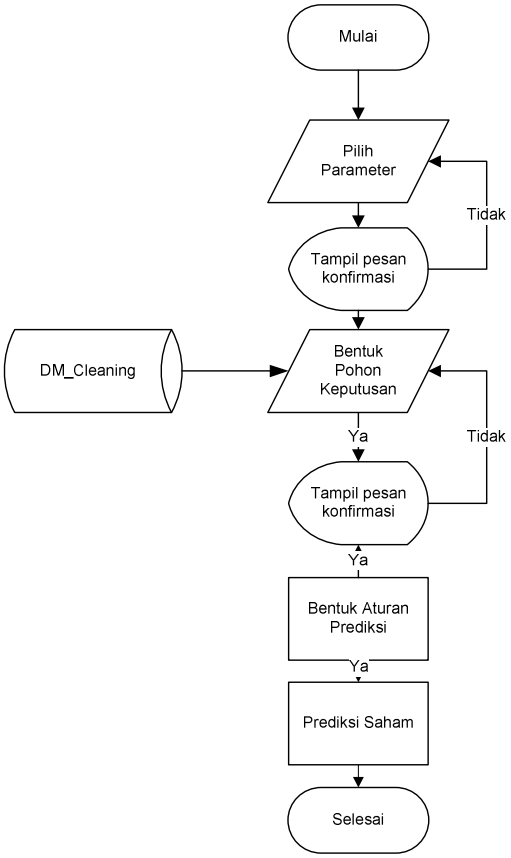
Gambar 3.30 Perancangan Prosedural *Flowchart* import data

2. *Flowchart* hapus data, merupakan prosedur yang terjadi ketika *user* menghapus data



Gambar 3.31 Perancangan Prosedural *Flowchart* hapus data

3. *Flowchart Proses Data*, merupakan prosedur yang terjadi ketika *user* Proses Data untuk sampai menjadi aturan prediksi



Gambar 3.32 Perancangan Prosedural *Flowchart* proses data