BAB 3
ANALISIS DAN PERANCANGAN SISTEM

3.1 Analisis Sistem

Analisis sistem adalah penguraian dari suatu informasi yang utuh kedalam bagian-bagian komponennya dengan maksud untuk mengidentifikasi dan mengevaluasi permasalahan, kesempatan, hambatan yang terjadi dan kebutuhan yang diharapkan sehingga dapat diusulkan perbaikan. Analisis Sistem ini bertujuan untuk menjelaskan permasalahan yang muncul pada saat pembangunan sistem, hal ini bertujuan untuk membantu ketika proses perancangan sistem berlangsung. Tugas utama analisis sistem dalam tahap ini adalah menemukan kelemahan-kelemahan dari sistem yang berjalan sehingga dapat diusulkan perbaikannya. Dalam analisis sistem ini meliputi beberapa bagian, yaitu :

1. Analisis Masalah
2. Analisis Penerepan *Data Mining*
3. Analisis Spesifikasi Kebutuhan Perangkat Lunak
4. Analisis Penerapan Kebutuhan Fungsional
5. Analisis Penerapan Kebutuhan Non Fungsional

3.1.1 Analisis Masalah

Analisis masalah yang dilakukan adalah melihat dan mengidentifikasi permasalahan atau kendala di dalam penelitian yang dilakukan. Masalah dapat diartikan sebagai suatu pernyataan yang menyatakan tentang situasi yang memerlukan pemecahan. Perlu ada analisis masalah melalui rumusan masalah yang sudah ditentukan sebelumnya. Berdasarkan hasil pengamatan, dapat disimpulkan bahwa permasalahan yang ada di PT. PATISA CHAKRA MANDIRI yaitu pada pemaketan penjualan saat ini dan penyusunan barang di dalam paket kemasan yang akan ditawarkan kepada konsumen dan belum adanya informasi penentuan paket barang yang sesuai dengan minat konsumen. Pemaketan barang dilakukan pada *event* besar seperti HUT Kota Bandung, HUT TNI, Bazzar dan *event* besar lainnya.

3.1.2 Analisis Penerapan *Data Mining*

Metode pembangunan perangkat *data mining* yang digunakan dalam penelitian ini adalah *Cross-Industry Standard Process for Data Mining* (*CRISP-DM*).

3.1.2.1 Pemahaman Bisnis

Tahapan pemahaman bisnis berfokus pada pemahaman tujuan kebutuhan berdasarkan penilaian bisnis. Kemudian pemahaman tersebut diubah menjadi sebuah rencana awal *data mining* yang dirancang untuk mencapai tujuan.

1. Tujuan Bisnis

   Menghasilkan keuntungan dengan menjual produk dalam jumlah banyak dengan memanfaatkan penjualan paket produk.

2. Menilai situasi

   Menilai situasi merupakan tahapan untuk menjelaskan sumber daya yang tersedia di PT. PATISA CHAKRA MANDIRI. Berikut merupakan sumber daya yang ada pada PT. PATISA CHAKRA MANDIRI:

   a. Inventory of resource

   1. Sumber Daya Perangkat Keras
      a) Terdapat alat penunjang transaksi yaitu *printer thermal* dan *cashdrawer* (laci kasir).
      b) Komputer.
      c) *Printer inject*.
   2. Sumber Daya Data
      a) Sumber daya data yang ada di PT. PATISA CHAKRA MANDIRI meliputi data transaksi, data stok barang, data pegawai, dan data permakloon.
      b) Sumber daya data di PT. PATISA ini bersifat internal yaitu tersimpan di database PT. PATISA CHAKRA MANDIRI.
   3. Sumber Daya Personil
      a) Pegawai kasir
      b) Pegawai gudang
      c) Pegawai produksi

d) Pegawai lab & QC (*Quality Control*)
e) Pegawai teknisi
f) Pegawai *accounting*
g) Pegawai personalia

b. Requirements, assumption, and constraints

a. Kebutuhan (*Requirements*)

Data yang digunakan dalam penelitian ini adalah data transaksi periode bulan Januari-Maret 2015.

b. Asumsi (*Assumption*)

Data transaksi yang digunakan merupakan hasil *export* dari kumpulan nota di PT. PATISA CHAKRA MANDIRI dengan format *Microsoft Excel*(.xlsx)

c. Batasan (*Constrains*)

a) Hasil dari *data mining* berupa *rule* sebagai acuan untuk membuat paket barang.
b) Hasil dari *generate rule* akan dilakukan *filtering rule*, yaitu menyeleksi beberapa *rule* yang tidak sesuai dengan batasan kriteria bisnis dari PT. PATISA CHAKRA MANDIRI dalam membentuk paket barang.
c) Batasan pembentukan paket berupa penentuan harga minimal dan maksimal yang dapat ditentukan oleh pengguna.

3. Penentuan sasaran *data mining*

Mengetahui keterkaitan barang yang sering dibeli oleh konsumen sehingga dapat dijadikan landasan untuk merekomendasikan barang mana saja yang dapat dijadikan paket barang.

3.1.2.2 Pemahaman Data

Tahapan pemahamn data dimulai dengan pengumpulan data awal dan hasil kegiatan dalam rangka untuk membiasakan diri dengan data untuk mengidentifikasi masalah data, untuk menentukan wawasan pertama ke data atau mendeteksi subset menarik untuk membentukan hipotesis untuk informasi yang tersembunyi.

1. Pengumpulan Data Awal

   Data yang digunakan dalam penelitian ini yakni data transaksi penjualan barang di PT. PATISA CHAKRA MANDIRI. Data yang digunakan untuk proses analisis berupa file Excel dengan format *.xlsx.

2. Mendeskripsikan Data

   Tahap kedua dalam pemahaman data adalah mendeskripsikan data dengan tujuan memahami data yang didapatkan dari hasil pengumpulan data awal. Berikut merupakan penjelasan dari masing-masing atribut yang terdapat pada tabel transaksi yang dapat dilihat pada tabel 3.1 di bawah ini.

Tabel 3. 1 Struktur Data Transaksi

DOKUMEN	KETERANGAN		
Detail data penjualan	Deskripsi :	Data ini berisi mengenai pemesanan langsung dari beberapa agen pada tiga bulan di PT. PATISA CHAKRA MANDIRI	
	Fungsi :	Untuk mengetahui produk mana saja yang paling banyak dipesan pada bulan tersebut	
	Format :	Microsoft Excel (.xlsx)	
	Atribut :	NoNota	Kode pembelian
		Tanggal	Waktu pemesanan
		Kode Barang	Kode dari barang yang dipesan
		Nama Barang	Nama dari barang
		Qty	Jumlah barang yang dibeli
		Kms	Bentuk barang
		Harga	Harga satuan barang

3. Verifikasi Kualitas Data

Verifikasi kualitas data merupakan tahap di mana dilakukan pengecekan apakah data tersebut memiliki *noise* atau tidaknya. *Noise* disini diartikan data transaksi yang hanya memiliki 1 barang dalam sekali pembelian, maka dari itu untuk menghilangkan *noise* pada data transaksi akan dilakukan proses pembersihan data pada tahap selanjunya dalam tahapan *data preparation.*

3.1.2.3 Persiapan Data

Persiapan data adalah hal yang harus dilakukan dalam proses *data mining*, karena tidak semua data atau atribut data dalam data digunakan pada data *mining*. Proses ini dilakukan agar data yang akan digunakan sesuai dengan kebutuhan. Adapun tahapan-tahapan *preprocessing* data dalam penelitian ini adalah sebagai berikut :

1. Ekstraksi Data

   Ekstraksi data dapat diartikan sebagai proses pengambilan data dari sumber data dalam rangka untuk melanjutkan proses pengolahan data ketingkat selanjutnya ataupun untuk menyimpan data hasil *ekstrak* tersebut. Dalam penelitian ini, data yang berasal dari *flat file* berformat *microsoft excel* di-*ekstrak*, kemudian disimpan kedalam sebuah *database* agar memudahkan dalam proses pengolahan data. Data transaksi penjualan terdapat dalam table D-1 dalam lampiran D.

Gambar 3.1 Sumber Data Transaksi PT. PATISA CHAKRA MANDIRI

Dalam Bentuk *Microsoft Excel*

No Nota	Tanggal	Kode Barang	Nama Barang	Qty	Kms	Harga
Mak001	1-Jan-15	AG240	AGUATIC CUP 240 ml	1,000	DUS	1,000
Mak002	2-Jan-15	AM220	AMIDIS CUP 220 ml	700	DUS	700
Mak002	2-Jan-15	AR500	ARVIN BOTOL 500 ml	700	PCS	2,000
Mak003	3-Jan-15	165600	165 BOTOL 600 ml	2,500	DUS	2,500
Mak003	3-Jan-15	AL1500	ALTO BOTOL 1500 ml	2,000	DUS	3,000
Mak003	3-Jan-15	AG240	AGUATIC CUP 240 ml	1,000	DUS	1,000
Mak003	3-Jan-15	AM220	AMIDIS CUP 220 ml	700	DUS	700
Mak004	4-Jan-15	GP19L	GALON PATISA 19 ltr	6,500	PCS	6,500
Mak005	5-Jan-15	GP19L	GALON PATISA 19 ltr	6,500	PCS	6,500
Mak005	5-Jan-15	AGU240	AGUARIA CUP 240 ml	1,000	DUS	1,000
Mak005	5-Jan-15	165600	165 BOTOL 600 ml	2,500	DUS	2,500
Mak005	5-Jan-15	AL1500	ALTO BOTOL 1500 ml	10	DUS	3,000
Mak005	5-Jan-15	AR500	ARVIN BOTOL 500 ml	1,500	PCS	2,000
Mak006	6-Jan-15	GP19L	GALON PATISA 19 ltr	2,000	PCS	6,500
Mak007	7-Jan-15	GP19L	GALON PATISA 19 ltr	950	PCS	6,500
Mak007	7-Jan-15	AR600	ARVIN BOTOL 600 ml	2,500	DUS	2,500
Mak007	7-Jan-15	AL1500	ALTO BOTOL 1500 ml	9	DUS	3,000
Mak007	7-Jan-15	AGU220	AGUARIA CUP 220 ml	1,000	DUS	700
Mak008	8-Jan-15	AM500	AMIDIS BOTOL 500 ml	100	DUS	2,000
Mak008	8-Jan-15	AM1500	AMIDIS BOTOL 1500 ml	5	DUS	3,000

2. Pemilihan Atribut (*Atribut Selection*)

   Pemilihan atribut adalah proses pemilihan atribut data yang digunakan, sehingga data tersebut dapat diproses sesuai dengan kebutuhan data mining. dalam tahap ini atribut yang digunakan adalah NoNota dan Kode Barang. NoNota merupakan identitas dari setiap transaksi, sedangkan Kode Barang digunakan untuk mengetahui barang apa saja yang dibeli oleh pelanggan. Hasil pemilihan Atribut dapat dilihat pada Lampiran D-2.

   Contoh:

Tabel 3. 2 pemilihan atribut

No Nota	Kode Barang
Mak001	AG240
Mak002	AM220
Mak002	AR500
Mak003	165600
Mak003	AL1500
Mak003	AG240
Mak003	AM220
Mak004	GP19L
Mak005	GP19L
Mak005	AGU240
Mak005	165600
Mak005	AL1500
Mak005	AR500
Mak006	GP19L
Mak007	GP19L
Mak007	AR600
Mak007	AL1500
Mak007	AGU220

3. Pembersihan data (*Data Cleaning*)

Pada tahap pembersihan data, hasil pemilihan atribut pada tabel D-2 akan dibersihkan dari NoNota yang hanya memiliki kode barang tunggal. Artinya pada satu transaksi konsumen hanya membeli satu jenis produk air mineral dalam kemasan saja. Data transaksi yang memiliki Kode Barang tunggal ini tidak memiliki hubungan asosiasi dengan Kode Barang lain yang sudah dibeli. Hasil pembersihan data dapat dilihat pada tabel D-3 dalam lampiran D. Contoh:

Tabel 3. 3 *Cleaning*

No Nota	Kode Barang
Mak001	AG240
Mak002	AM220
Mak002	AR500
Mak003	165600
Mak003	AL1500
Mak003	AG240
Mak003	AM220
Mak004	GP19L
Mak005	GP19L
Mak005	AGU240
Mak005	165600
Mak005	AL1500
Mak005	AR500
Mak006	GP19L
Mak007	GP19L
Mak007	AR600
Mak007	AL1500
Mak007	AGU220
Mak008	AM500
Mak008	AM1500
Mak008	AGU220
Mak008	GP19L
Mak008	AR500
Mak009	GP19L
Mak009	AR500
Mak010	165600
Mak011	165600
Mak012	AMGL19L
Mak013	AMGL19L



No Nota	Kode Barang
Mak002	AM220
Mak002	AR500
Mak003	165600
Mak003	AL1500
Mak003	AG240
Mak003	AM220
Mak005	GP19L
Mak005	AGU240
Mak005	165600
Mak005	AL1500
Mak005	AR500
Mak007	GP19L
Mak007	AR600
Mak007	AL1500
Mak007	AGU220
Mak008	AM500
Mak008	AM1500
Mak008	AGU220
Mak008	GP19L
Mak008	AR500
Mak009	GP19L
Mak009	AR500
Mak014	AR500
Mak014	GP19L
Mak014	1651500
Mak014	AR600
Mak014	AM240
Mak014	AGU220

3.1.2.4 Pemodelan

Pemodelan merupakan tahap untuk membuat model desain sistem yang akan dibangun. Data yang digunakan untuk pengolahan data ini adalah data terakhir yang sudah dilakukan proses pembersihan data pada lampiran D tabel D-3. Dalam tahap ini akan dipilih metode yang cocok untuk menyelesaikan masalah di PT. PATISA CHAKRA MANDIRI dalam pembuatan paket air minum dalam kemasan. Metode yang akan digunakan adalah metode *associaiton rule* dengan algoritma *fp-growth*. Metode dasar *association rule* dibagi menjadi dua tahap yaitu:

1. ***Frequent Itemset.***

Pada tahap awal dalam pemodelan adalah menentukan *minimum support*. *Minimum support* adalah suatu ukuran yang menunjukan seberapa besar tingkat suatu kombinasi suatu barang pada pada sebuah data transaksi. Untuk melihat banyaknya variasi data yang diperoleh, maka pada kasus ini nilai *minimum support* yang akan digunakan yaitu 7% dari 200 sampel data transaksi yang digunakan yaitu sebesar $2{,}85 \approx 3$, yang berarti barang yang memiliki jumlah *frequent* kurang dari 3 akan dieliminasi dalam data transaksi. Langkah pertama yang dilakukan untuk menghitung nilai *support* dari sebuah barang adalah dengan menghitung kemunculan tiap barang yang sudah disusun secara *ascending*. Jumlah kemunculan setiap barang dapat dilihat pada tabel 3.4.

Tabel 3.4 *Frequent Item*

Kode Barang	Support count	Priority
GP19L	29	1
AR500	26	2
165600	20	3
AL1500	16	4
AG240	14	5
AM220	13	6
AGU240	11	7
AM240	9	8
AR220	5	9
AMGL19L	5	10
AGU220	3	11
FU600	2	12
AR220	2	13
AG220	2	14
FU500	2	15
AM600	1	16
FU240	1	17
AR600	1	18
AR1500	1	19
1651500	1	20

Setelah mengetahui nilai *frequent* dari semua *item*, langkah kedua adalah mengeliminasi *item* yang memiliki *frequent* kurang dari nilai *minimum support* yang telah ditentukan sebelumnya, dalam kasus ini nilai *minimum support* yang telah ditentukan adalah sebesar 3. Dimana *item* yang kurang dari 3 akan dihilangkan dan *item* yang memiliki *frequent* lebih atau sama dengan 3 akan akan digunakan untuk ke proses berikutnya. Berikut adalah data yang sudah memenuhi nilai *minimum support* terdapat pada tabel 3.5.

Tabel 3. 5 *Item* yang memenuhi *Minimum support*

Kode Barang	Support count
GP19L	29
AR500	26
165600	20
AL1500	16
AG240	14
AM220	13
AGU240	11
AM240	9
AR220	5
AMGL19L	5
AGU220	3

Setelah mendapatkan data yang memenuhi nilai *minimum support* seperti pada tabel 3.3 diatas, selanjutnya mencari *Frequency of Occurrence*, urutkan tabel 3.3 berdasarkan *frequent* terbesar sampai dengan terkecil, kemudian berikan nilai *priority* dimulai dari data *frequent* terbesar sampai dengan terkecil. Jika terdapat kode barang dengan *frequent* yang sama maka yang didahulukan adalah kode barang yang pertama kali muncul dalam transaksi. Berikut ini adalah hasil pengurutan *frequent item* berdasarkan proritas terdapat pada tabel 3.6.

Tabel 3. 6 *Frequency of Occurence*

Kode Barang	Support count	PRIORITY
GP19L	29	1
AR500	26	2
165600	20	3
AL1500	16	4
AG240	14	5
AM220	13	6
AGU240	11	7
AM240	9	8
AR220	5	9
AMGL19L	5	10
AGU220	3	11

Setelah semua *item* diketahui nilai *priority* nya maka data transaksi yang mengandung *item* yang memenuhi *minimum support* akan diurutkan berdasarkan nilai *priority*. Pada tahap ini *item* yang memiliki *frequent* tertinggi atau nilai *priority* tertinggi akan didahulukan penulisannya. Hasil dari pengurutan data berdasarkan *priority* terdapat pada tabel 3.7.

Tabel 3. 7 Hasil Pengurutan data Berdasarkan *Priority*

No Nota	Kode Barang
Mak002	AR500
Mak002	AM220
Mak003	165600
Mak003	AL1500
Mak003	AG240
Mak003	AM220
Mak005	GP19L
Mak005	AR500
Mak005	165600
Mak005	AL1500
Mak005	AGU240
Mak007	165600
Mak007	AL1500
Mak007	AG240
Mak007	AM220
Mak008	GP19L
Mak008	AR500
Mak008	165600
Mak008	AL1500
Mak008	AGU240
Mak009	GP19L
Mak009	AR500

Setelah data sudah terurut berdasarkan *priority*, langkah selanjutnya membuat *fp- tree*. *FP-tree* merupakan struktur penyimpanan data yang dimampatkan. *FP-tree* dibangun dengan memetakan setiap data transaksi ke dalam setiap lintasan tertentu dalam *FP-tree*. Karena dalam setiap transaksi yang dipetakan, mungkin ada transaksi yang memiliki *item* yang sama, maka lintasannya memungkinkan untuk saling menimpa. Semakin banyak data transaksi yang memiliki item yang sama, maka proses pemampatan dengan struktur data *FP-tree* semakin efektif. Kelebihan dari *FP-tree* adalah hanya memerlukan dua kali pemindaian data transaksi yang terbukti sangat efisien. Misal I= {a1, a2, …, an} adalah kumpulan dari *item*. Dan basis data transaksi DB = {T1, T2, …, Tn}, di mana Ti (i € [1..n]) adalah sekumpulan transaksi yang mengandung *item* di I. Sedangkan *support* adalah penghitung (*counter*) frekuensi kemunculan transaksi yang mengandung suatu pola. Suatu pola dikatakan sering muncul (*frequent pattern*) apabila *support* dari pola tersebut tidak kurang dari suatu konstanta ξ (batas ambang *minimum support*) yang telah didefinisikan sebelumnya. Permasalahan mencari pola *frequent* dengan batas ambang *minimum support count* ξ inilah yang dicoba untuk dipecahkan oleh *FP-Growth* dengan bantuan Struktur *FP-tree*. Berikut adalah tahap-tahap dalam pembuatan *fp-tree* yaitu :

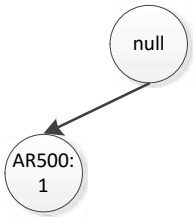
a. Buat *Tree* dengan *Root* yang tidak memiliki anak.
   Contoh :



Gambar 3.1 *Tree* dengan *root* tidak memiliki anak

b. Jika *Root* tidak memiliki anak, maka data yang muncul pertama kali dijadikan sebagai anak pertama dari *Root* disertai dengan *support count*= 1.

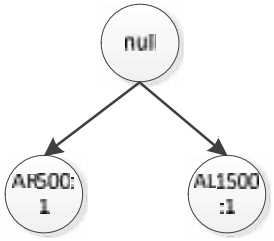
Contoh:



Gambar 3.1 *Tree* dengan *root* tidak memiliki anak disertai *support count*

c. Jika *Root* sudah memiliki anak, tetapi data yang muncul berbeda dengan anak dari *Root*, maka data tersebut merupakan anak selanjutnya dari *Root* tersebut dan *support count* nya bertambah 1.

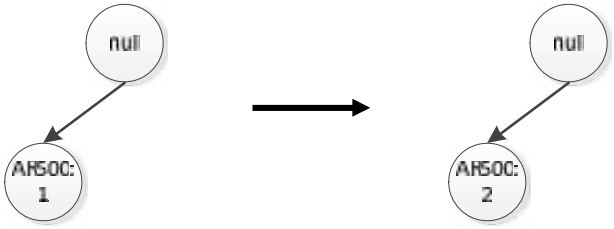
Contoh:



Gambar 3.1 *Root* dengan berbeda anak

d. Jika *Root* sudah mempunyai anak, dan data yang muncul sama dengan anak tersebut maka *support count* nya bertambah 1.

Contoh:

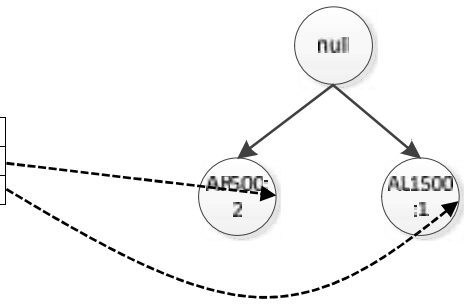


Gambar 3.1 *Root* memiliki anak dengan data yang sama

e. Jika data yang muncul sudah ada dalam notasi anak, maka *support count* nya bertambah 1 .

Contoh:

Kode barang	Support count
AR500	2
AL1500	1



Gambar 3.2 Data yang sudah ada dalam notasi anak

Hasil pembentukan *fp-tree* tiap transaksi pada data transaksi ini dapat dilihat pada table D-4 dalam lampiran D.

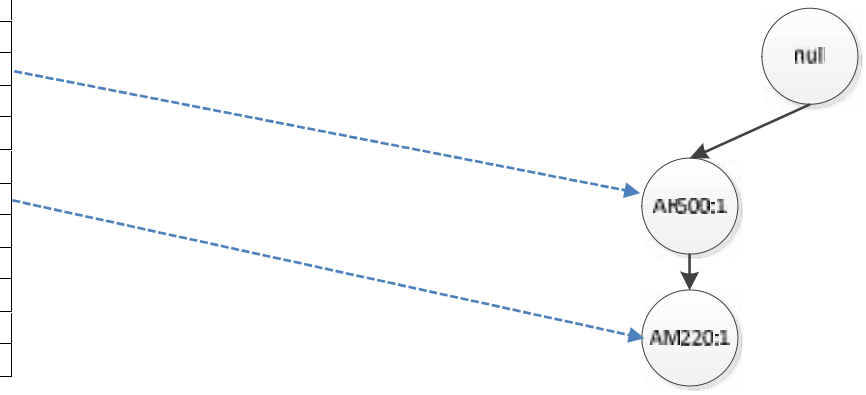
1. Setelah tahap pembangunan fp-tree dari sekumpulan data transaksi dilakukan, maka akan diterapkan Algoritma *FP-Growth* untuk mencari *frequent patterns generated* yang signifikan. Berikut langkah-langkah utama Algoritma *FP-Growth*:
   a) Tahapan Pembangkitan *Conditional Pattern Base*

      Pembangkitan *Conditional Pattern Base* didapatkan melalui hasil *fp-tree* seluruhnya dengan mencari *support count* terkecil sesuai dengan hasil pengurutan *priority* yang telah dibuat sebelumnya.
   b) Tahap Pembangkitan *Conditional FP-Tree*

      Pada tahap ini, *support count* dari setiap item pada Tabel *conditional pattern base* diatas dijumlahkan, lalu setiap item yang memiliki jumlah *Support Count* lebih besar atau sama dengan *Minimum Support* akan dibangkitkan dengan *Conditional FP-Tree*.
   c) Tahap Pencarian *frequent patterns generated* apabila *conditional FP-Tree* merupakan lintasan tunggal (*single path*), maka didapatkan *frequent patterns generated* dengan melakukan kombinasi item untuk setiap *conditional FP-Tree*. Jika bukan lintasan tunggal, maka dilakukan pembangkitan secara rekursif.

Berikut merupakan pembentukan *fp-tree* untuk noNota MAK002 dengan barang yang dipesan adalah AR500, AM220 :

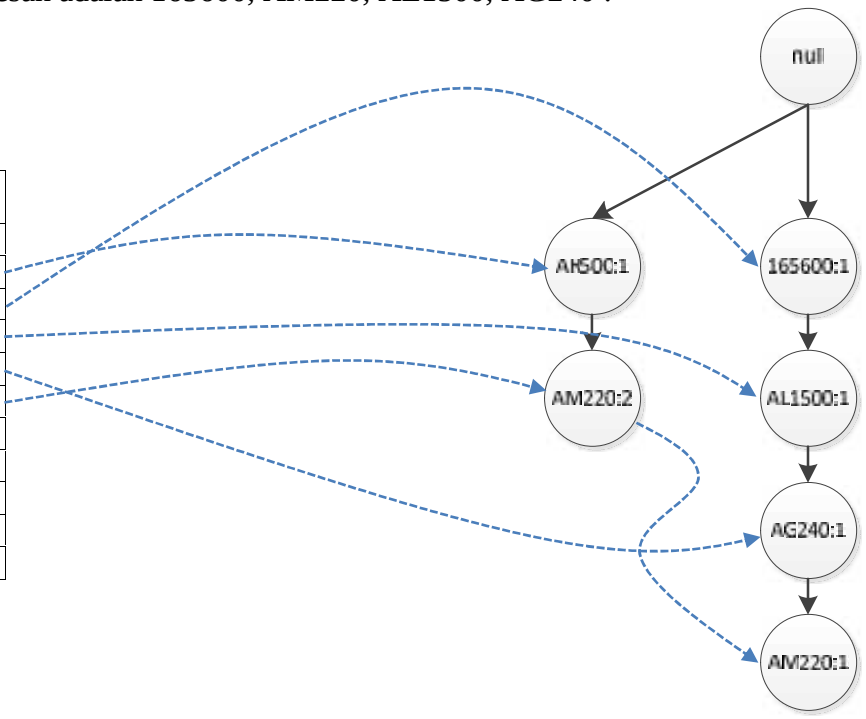
Kode barang	frekuensi	Node link
GP19L		
AR500	1	•
165600		
AL1500		
AG240		
AM220	1	•
AGU240		
AM240		
AR240		
AMGL19L		
AGU220		



Gambar 3. 3 Hasil pembentukan *FP-Tree* noNota MAK002

Berikut merupakan pembentukan *fp-tree* untuk noNota MAK003 dengan barang yang dipesan adalah 165600, AM220, AL1500, AG240 :

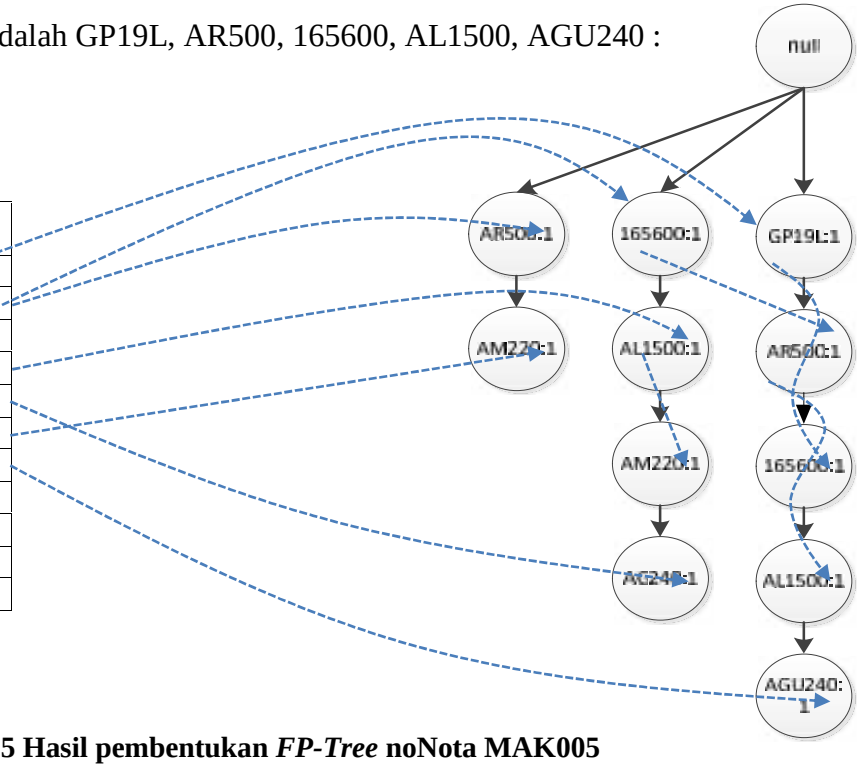
Kode barang	frekuensi	Node link
GP19L		
AR500	1	•
165600	1	•
AL1500	1	•
AG240	1	•
AM220	3	•
AGU240		
AM240		
AR240		
AMGL19L		
AGU220		



Gambar 3. 4 Hasil pembentukan *FP-Tree* noNota MAK003

Berikut merupakan pembentukan *fp-tree* untuk noNota MAK005 dengan barang yang dipesan adalah GP19L, AR500, 165600, AL1500, AGU240 :

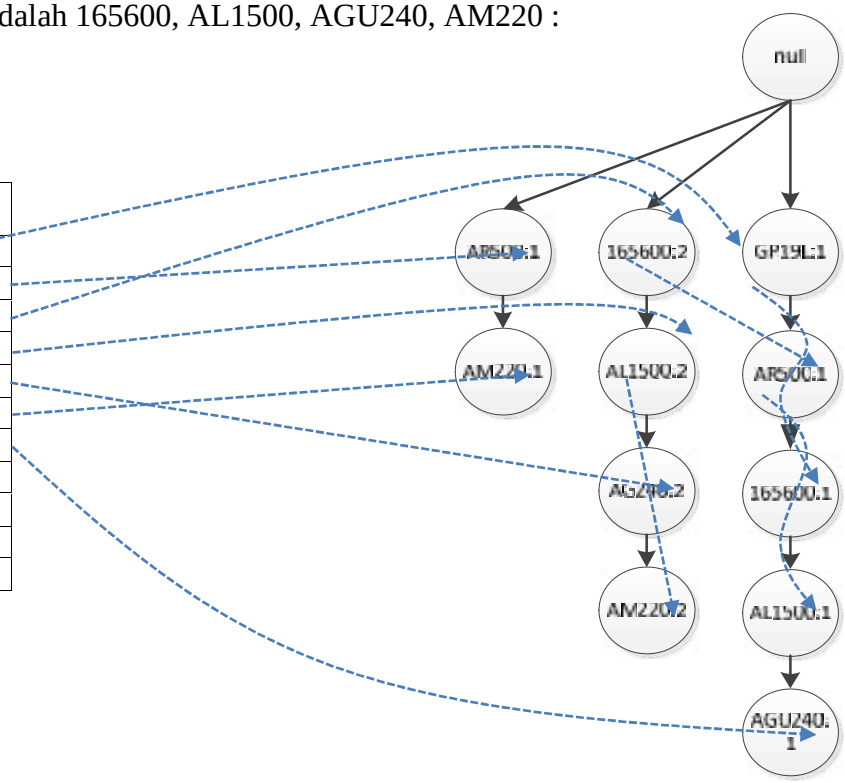
Kode barang	frekuensi	Node link
GP19L	1	•
AR500	2	•
165600	2	•
AL1500	2	•
AG240	1	•
AM220	2	•
AGU240	1	•
AM240		
AR240		
AMGL19L		
AGU220		



Gambar 3. 5 Hasil pembentukan *FP-Tree* noNota MAK005

Berikut merupakan pembentukan *fp-tree* untuk noNota MAK007 dengan barang yang dipesan adalah 165600, AL1500, AGU240, AM220 :

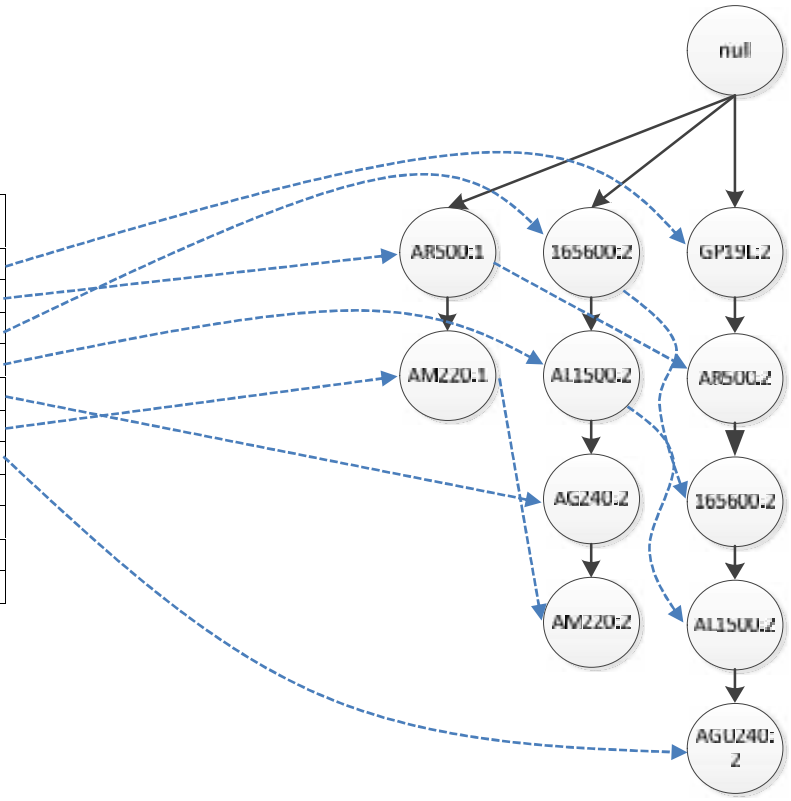
Kode barang	frekuensi	Node link
GP19L	1	•
AR500	2	•
165600	3	•
AL1500	3	•
AG240	2	•
AM220	3	•
AGU240	1	•
AM240		
AR240		
AMGL19L		
AGU220		



Gambar 3. 6 Hasil pembentukan *FP-Tree* noNota MAK007

Berikut merupakan pembentukan *fp-tree* untuk noNota MAK008 dengan barang yang dipesan adalah GP19L, AR500, 165600, AL1500, AGU240 :

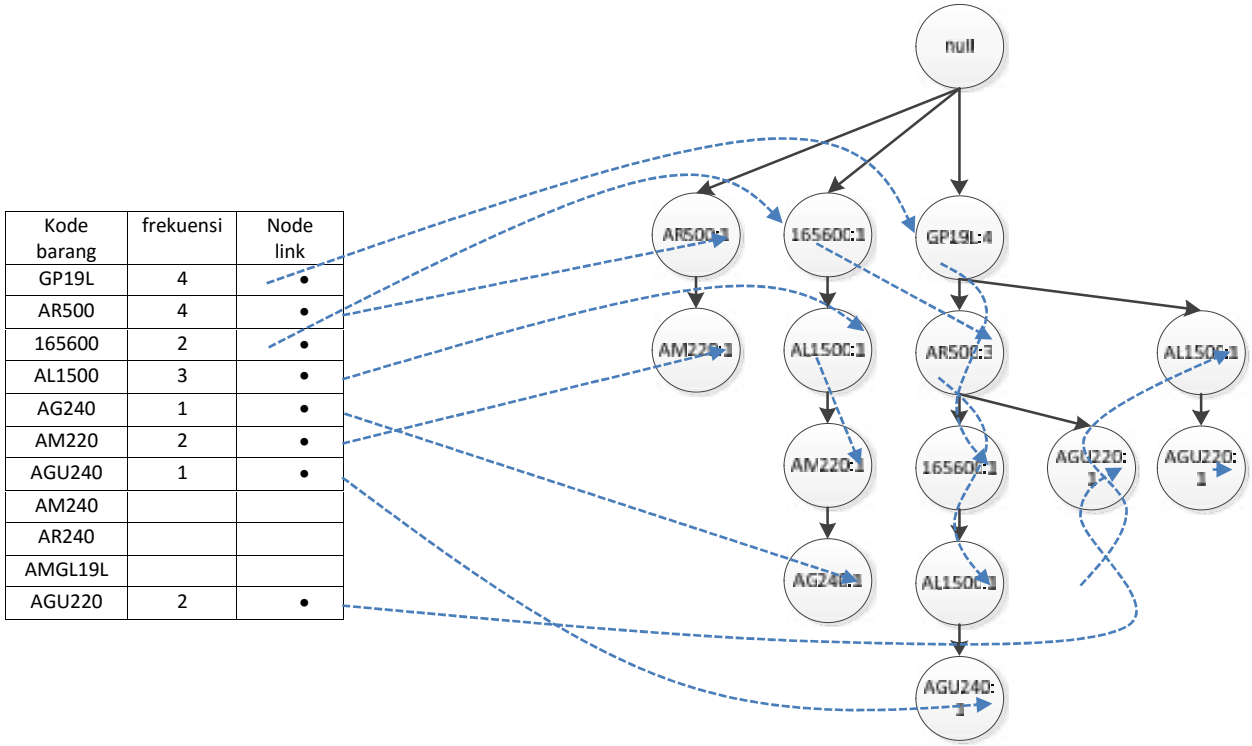
Kode barang	frekuensi	Node link
GP19L	2	•
AR500	3	•
165600	4	•
AL1500	4	•
AG240	2	•
AM220	3	•
AGU240	2	•
AM240		
AR240		
AMGL19L		
AGU220		



Gambar 3. 7 Hasil pembentukan FP-Tree noNota MAK008

Berikut merupakan pembentukan *fp-tree* untuk noNota MAK009 dengan barang yang dipesan adalah GP19L, AR500 :

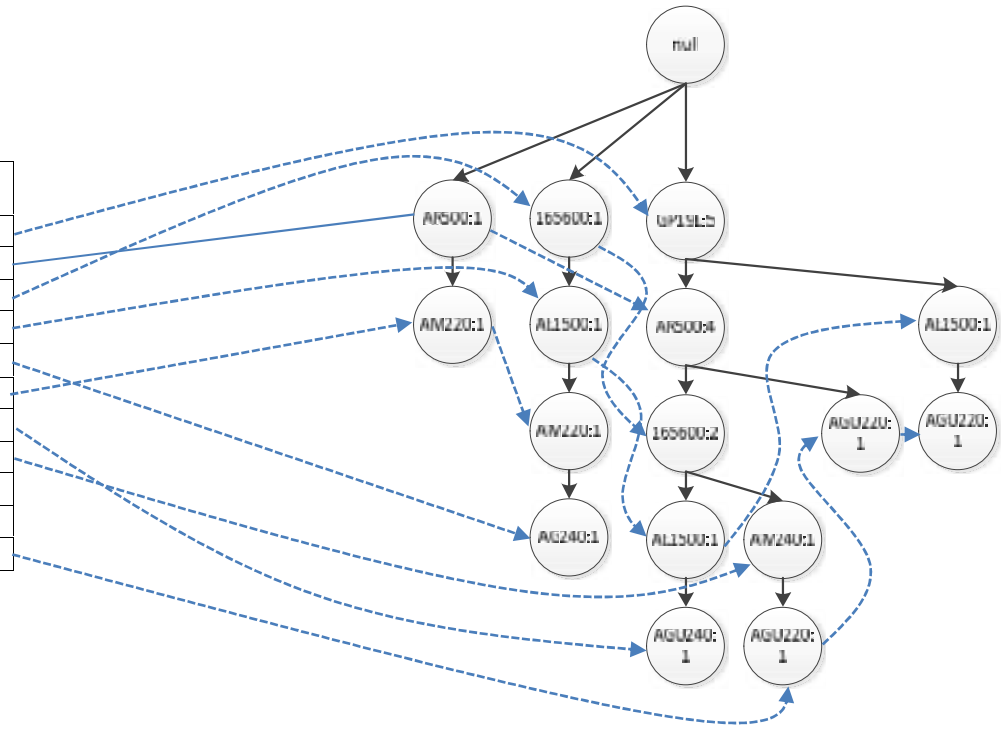
Kode barang	frekuensi	Node link
GP19L	4	•
AR500	4	•
165600	2	•
AL1500	3	•
AG240	1	•
AM220	2	•
AGU240	1	•
AM240		
AR240		
AMGL19L		
AGU220	2	•



Gambar 3. 8 Hasil pembentukan *FP-Tree* noNota MAK009

Berikut merupakan pembentukan *fp-tree* untuk noNota MAK014 dengan barang yang dipesan adalah GP19L, AR500, 165600, AM240, AGU220 :

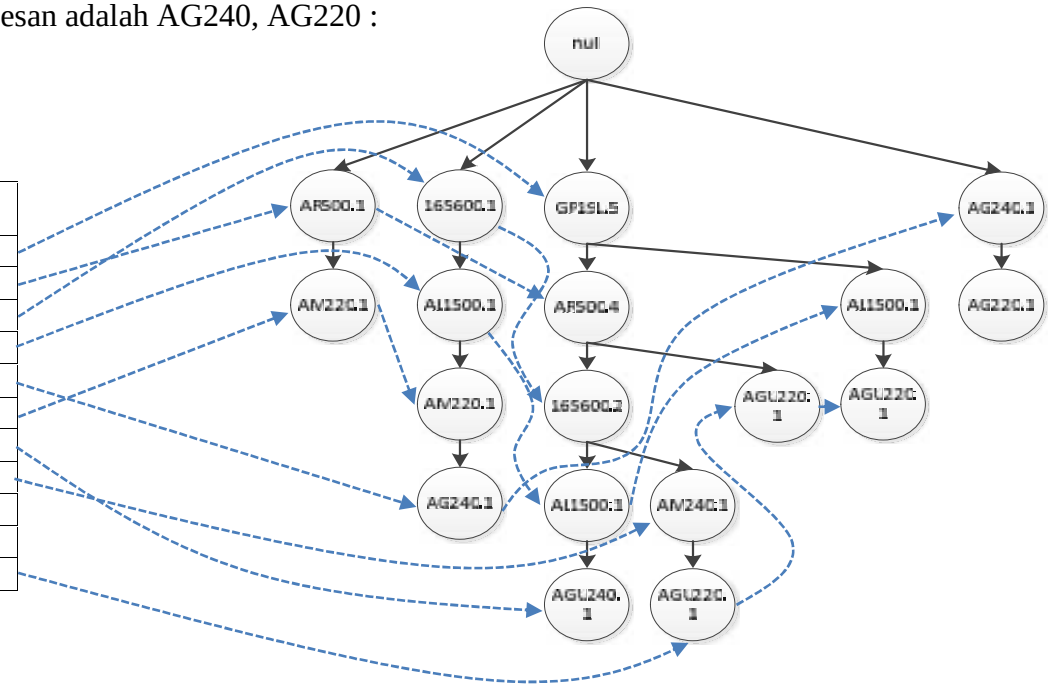
Kode barang	frekuensi	Node link
GP19L	5	•
AR500	5	•
165600	3	•
AL1500	3	•
AG240	1	•
AM220	2	•
AGU240	1	•
AM240	1	•
AR240		
AMGL19L		
AGU220	3	•



Gambar 3. 9 Hasil pembentukan *FP-Tree* noNota MAK014

Berikut merupakan pembentukan *fp-tree* untuk noNota MAK016 dengan barang yang dipesan adalah AG240, AG220 :

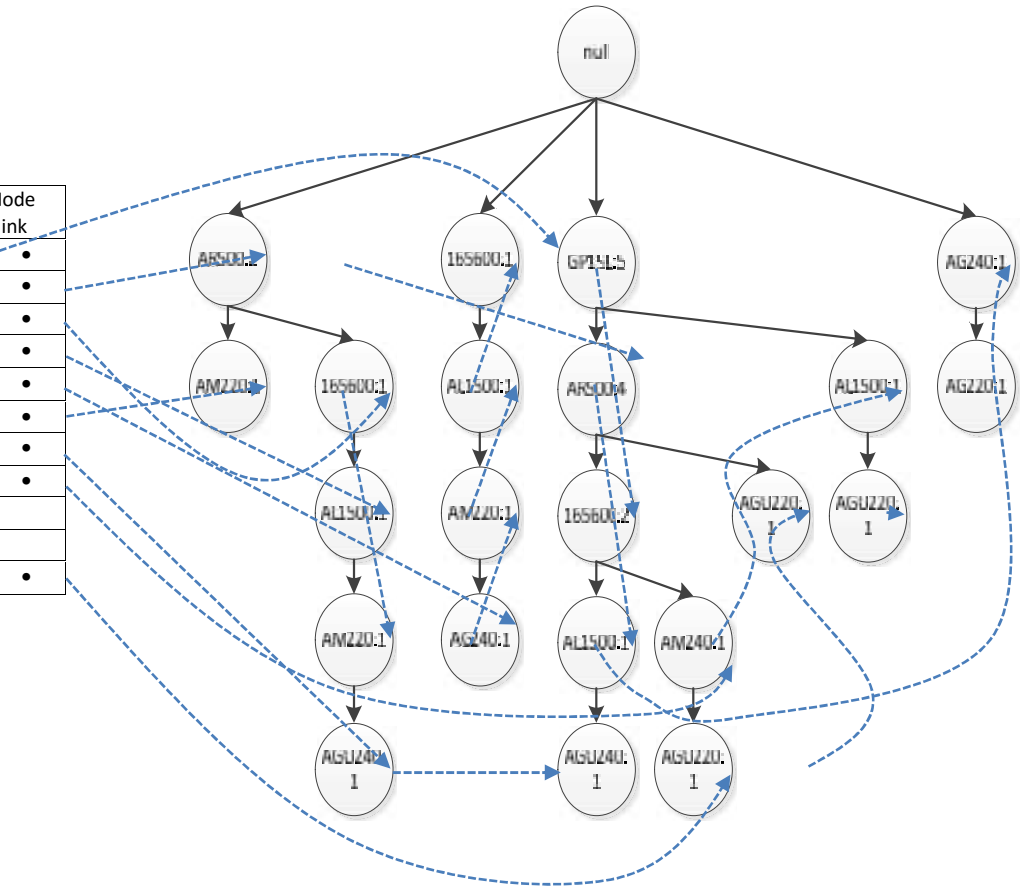
Kode barang	frekuensi	Node link
GP19L	5	•
AR500	6	•
165600	3	•
AL1500	3	•
AG240	2	•
AM220	2	•
AGU240	1	•
AM240	1	•
AR240		
AMGL19L		
AGU220	3	•



Gambar 3. 10 Hasil pembentukan *FP-Tree* noNota MAK016

Berikut merupakan pembentukan *fp-tree* untuk noNota MAK017 dengan barang yang dipesan adalah AR500, 165600, AL1500, AM220, AGU240 :

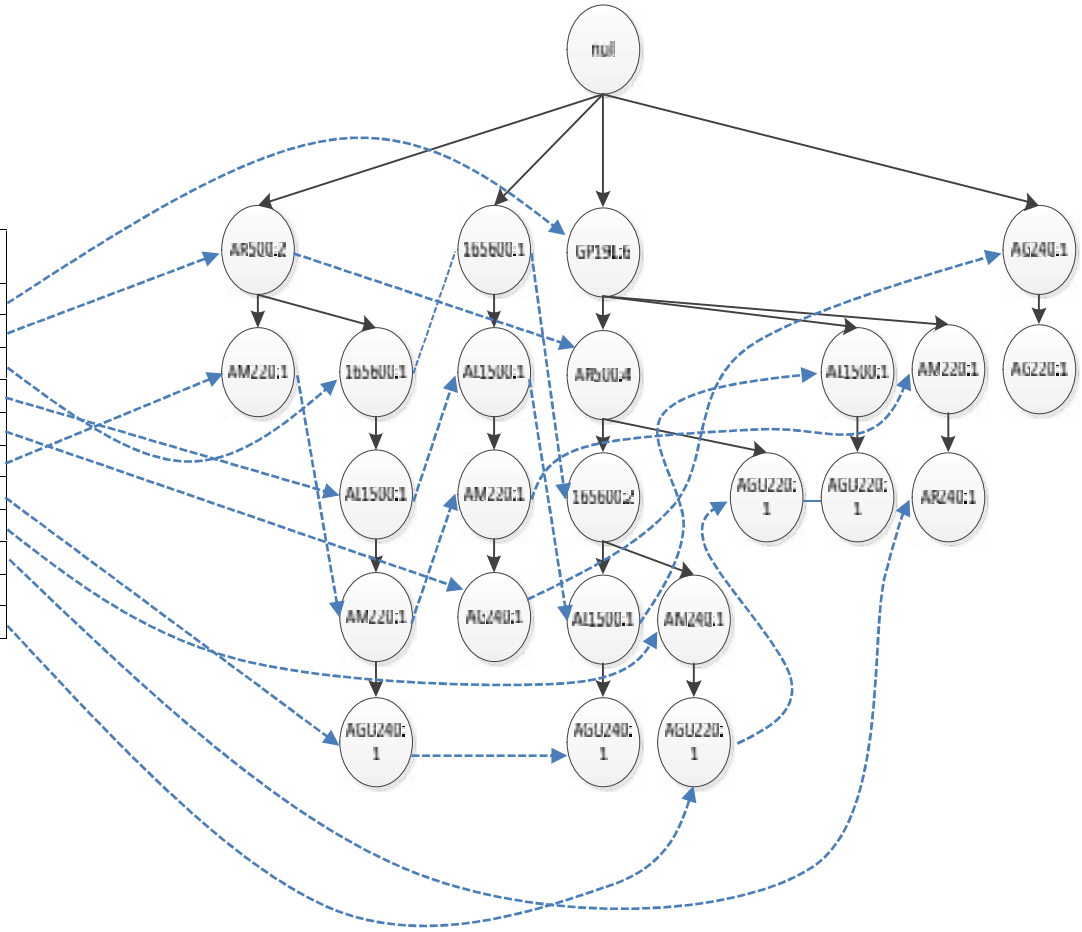
Kode barang	frekuensi	Node link
GP19L	5	•
AR500	6	•
165600	4	•
AL1500	4	•
AG240	2	•
AM220	2	•
AGU240	2	•
AM240	1	•
AR240		
AMGL19L		
AGU220	3	•



Gambar 3. 11 Hasil pembentukan *FP-Tree* noNota MAK017

Berikut merupakan pembentukan *fp-tree* untuk noNota MAK020 dengan barang yang dipesan adalah GP19L, AM220, AR240 :

Kode barang	frekuensi	Node link
GP19L	6	•
AR500	6	•
165600	4	•
AL1500	4	•
AG240	2	•
AM220	4	•
AGU240	2	•
AM240	1	•
AR240	1	•
AMGL19L		
AGU220	3	•



Gambar 3. 12 Hasil pembentukan *FP-Tree* noNota MAK020

a. Tahap pembangkitan *conditional pattern base*

Pembangkitan *conditional pattern base* didapatkan melalui hasil *FP-Tree* seluruhya, dengan mencari *support count* terkecil sesuai dengan hasil pengurutan *priority*. Dan telah didapatkan yaitu : AGU220, AMGL19L, AR220, AM240, AGU240, AM220, AG240, AL1500, 165600, AR500. Berikut adalah hasil tabel dari *conditional pattern base*:

Tabel 3. 8 Hasil perhitungan *Conditional Pattern Base*

Items	***Conditional Pattern Base***
AGU220	• GP19L, AL1500: 1 • GP19L, AR500, 165600, AM240: 1 • GP19L, AR500, AL1500, AG240: 1 • 165600, AL1500, AG240: 1
AMGL19L	• GP19L, 165600, AL1500: 1 • GP19L, 165600, AL1500, AGU240: 1 • 165600: 1 • GP19L: 1 • GP19L, AGU240: 1
AR220	• GP19L, AG240: 1
AM240	• 165600: 1 • AM220: 1 • GP19L, AR500, 165600:1 • GP19L, AR500: 2 • AR500: 1 • GP19L: 1
AGU240	• AR500: 1 • 165600:1 • GP19L: 1 • GP19L, AM220: 1 • GP19L, AR500, 165600, AL1500: 6 • GP19L, 165600, AL1500: 1
AM220	• AR500: 3 • AG240: 1 • GP19L: 3 • 165600, AL1500, AG240: 2 • AR500, AL1500, AG240: 2 • GP19L, 165600, AL1500: 1 • AR500, 165600, AL1500, AG240: 1
AG240	• GP19L: 1

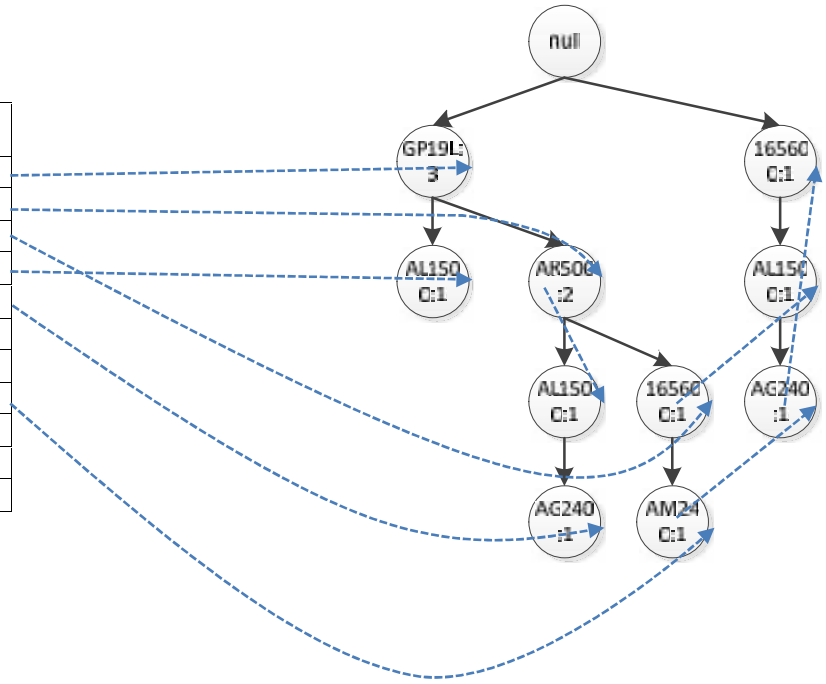
Items	*Conditional Pattern Base*
	• AR500: 1 • AR500, AL1500: 1 • 165600, AL1500: 3 • GP19L, AR500: 1 • AR500, 165600, AL1500: 1 • GP19L, AR500, 165600:1 • GP19L, AR500, AL1500:1
AL1500	• AR500, 165600: 1 • GP19L, AR500: 1 • GP19L, 165600: 3 • AR500: 1 • 165600: 3 • GP19L: 1
165600	• AR500: 2 • GP19L: 3 • GP19L, AR500: 10
AR500	• GP19L: 16

b. Tahap pembangkitan *conditional FP-Tree*

Pada tahap ini, *support count* dari setiap *item* pada setiap *conditional pattern base* dijumlahkan, lalu setiap *item* yang memiliki jumlah *support count* lebih besar atau sama dengan *minimum support* akan dibangkitkan dengan *Conditional FP-Tree.*

1. *Conditional FP-Tree* untuk kode barang AGU220 {GP19L:3}{AL1500:3}

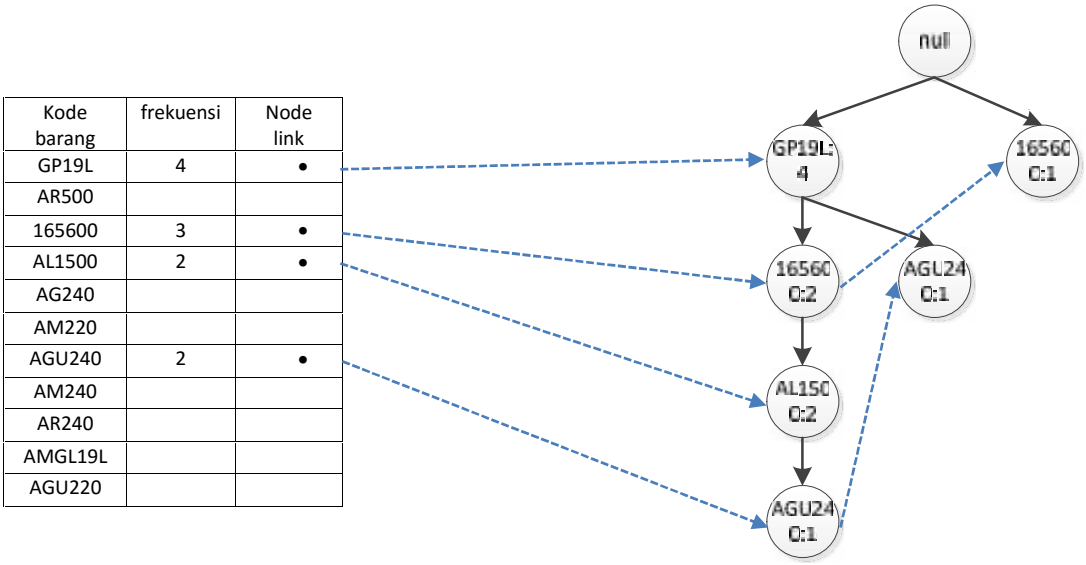
Kode barang	frekuensi	Node link
GP19L	3	•
AR500	2	•
165600	1	•
AL1500	3	•
AG240	1	•
AM220		
AGU240		
AM240	1	•
AR240		
AMGL19L		
AGU220		



Gambar 3.13 Lintasan yang mengandung kode barang AGU220

2. *Conditional FP-Tree* untuk kode barang AMGL19L {GP19L:4}{165600:3}

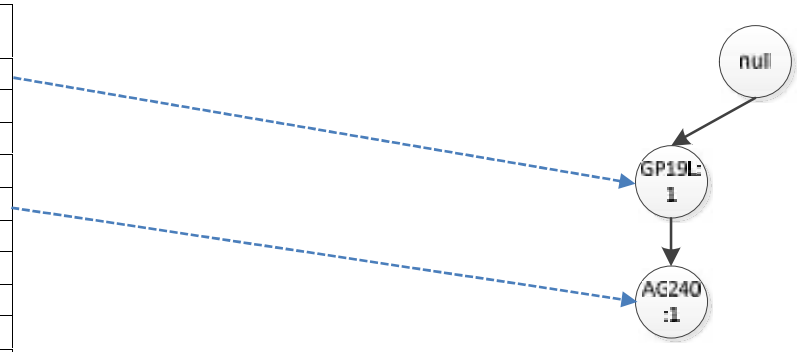
Kode barang	frekuensi	Node link
GP19L	4	•
AR500		
165600	3	•
AL1500	2	•
AG240		
AM220		
AGU240	2	•
AM240		
AR240		
AMGL19L		
AGU220		



Gambar 3.14 Lintasan yang mengandung kode barang AMGL19L

3. *Conditional FP-Tree* untuk kode barang AR220 yaitu null karena walapun *support*nya memenuhi nilai *support*, tetapi pada proses pembangkitan tidak didapatkan *support count* yang memenuhi nilai *minimum support*

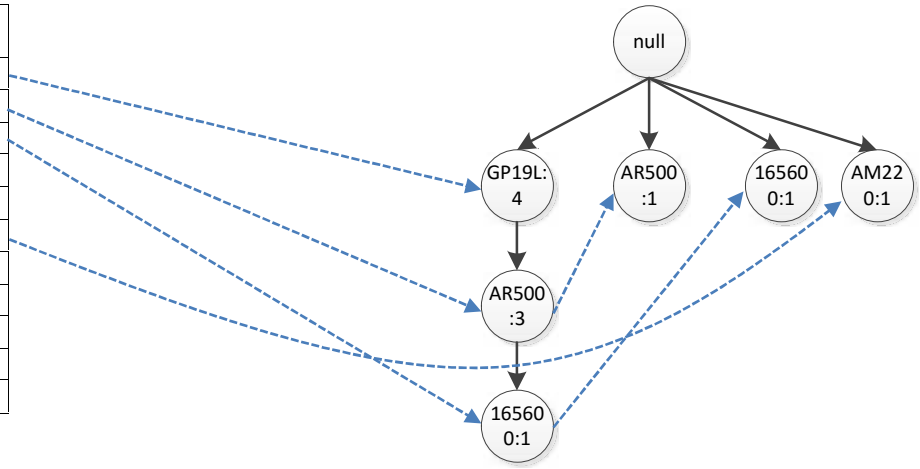
Kode barang	frekuensi	Node link
GP19L	1	•
AR500		
165600		
AL1500		
AG240	1	•
AM220		
AGU240		
AM240		
AR240		
AMGL19L		
AGU220		



Gambar 3.15 Lintasan yang mengandung kode barang AG220

4. *Conditional FP-Tree* untuk kode barang AM240 {GP19L:4}{AR500:4}

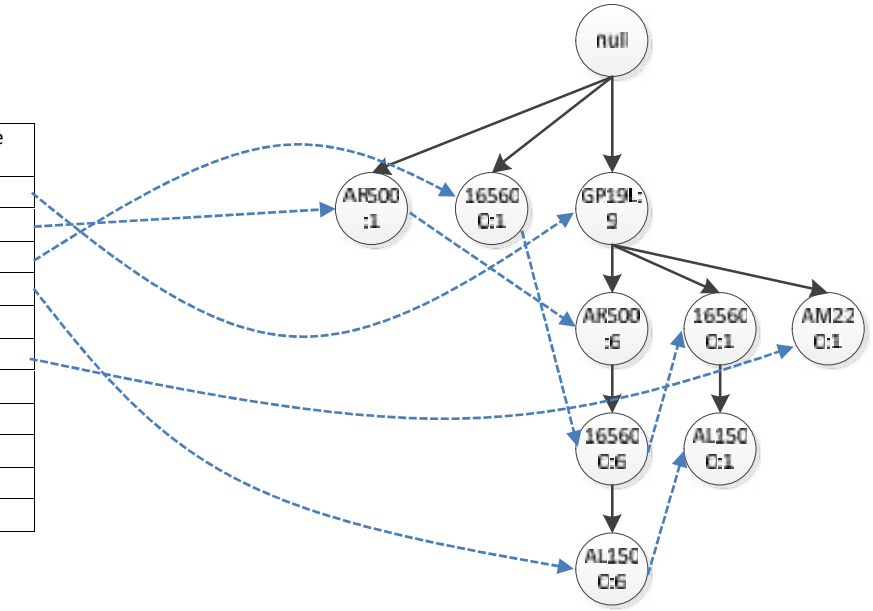
Kode barang	frekuensi	Node link
GP19L	4	•
AR500	4	•
165600	2	•
AL1500		
AG240		
AM220	1	•
AGU240		
AM240		
AR240		
AMGL19L		
AGU220		



Gambar 3.16 Lintasan yang mengandung kode barang AM240

5. *Conditional FP-Tree* untuk kode barang AGU240 {GP19L:9}{165600:8}{AL1500:7}

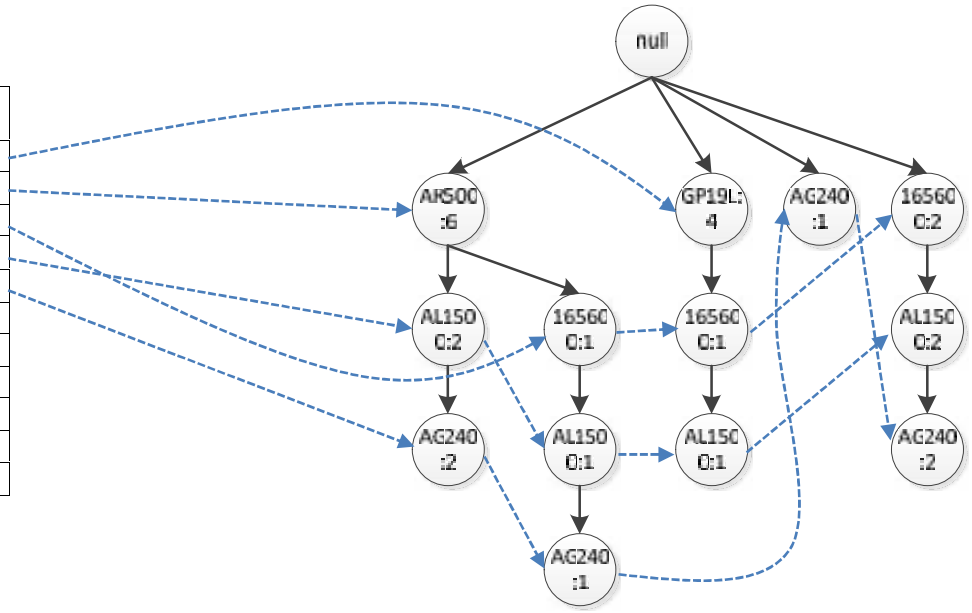
Kode barang	frekuensi	Node link
GP19L	9	•
AR500	7	•
165600	8	•
AL1500	7	•
AG240		
AM220	1	•
AGU240		
AM240		
AR240		
AMGL19L		
AGU220		



Gambar 3.17 Lintasan yang mengandung kode barang AGU240

6. *Conditional FP-Tree* untuk kode barang AM220 {AR500:6}
   { AG240:6}{AL1500:6}{165600:4}

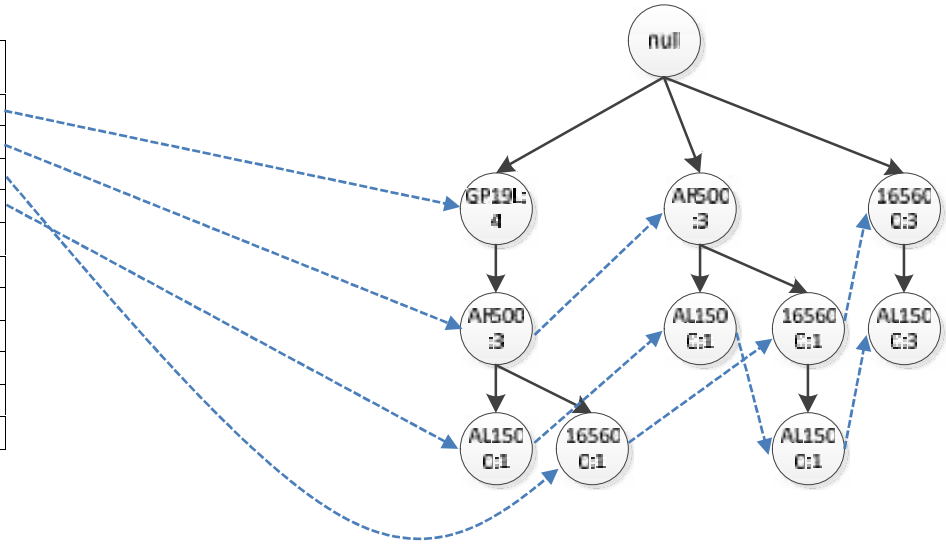
Kode barang	frekuensi	Node link
GP19L	4	•
AR500	6	•
165600	4	•
AL1500	6	•
AG240	6	•
AM220		
AGU240		
AM240		
AR240		
AMGL19L		
AGU220		



Gambar 3.18 Lintasan yang mengandung kode barang AM220

7. *Conditional FP-Tree* untuk kode barang AG240 {GP19L:4}
   { AR500:6}{165600:5}{AL1500:6}

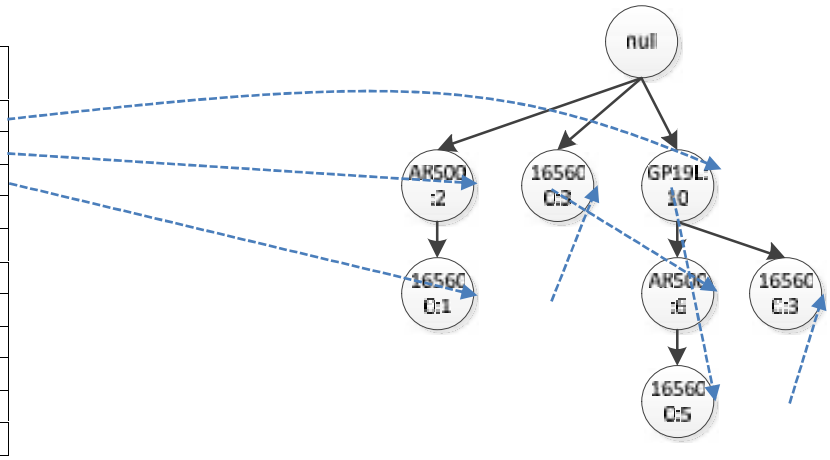
Kode barang	frekuensi	Node link
GP19L	4	•
AR500	6	•
165600	5	•
AL1500	6	•
AG240		
AM220		
AGU240		
AM240		
AR240		
AMGL19L		
AGU220		



Gambar 3.19 Lintasan yang mengandung kode barang AG240

8. *Conditional FP-Tree* untuk kode barang AL1500 {GP19L:10}
   { AR500:8}{165600:12}

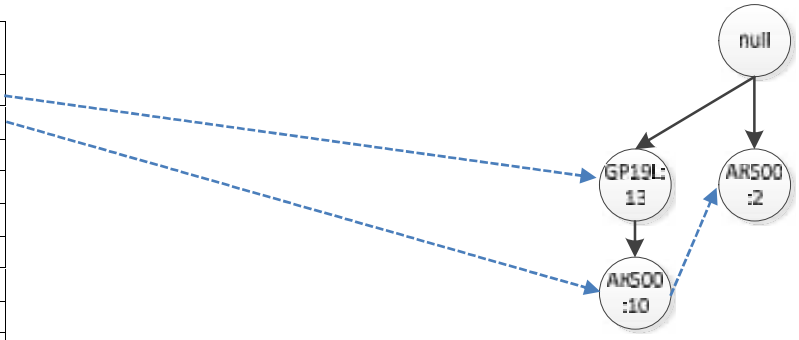
Kode barang	frekuensi	Node link
GP19L	10	•
AR500	8	•
165600	12	•
AL1500		
AG240		
AM220		
AGU240		
AM240		
AR240		
AMGL19L		
AGU220		



Gambar 3.20 Lintasan yang mengandung kode barang AL1500

9. *Conditional FP-Tree* untuk kode barang 165600 {GP19L:13}
   { AR500:12}

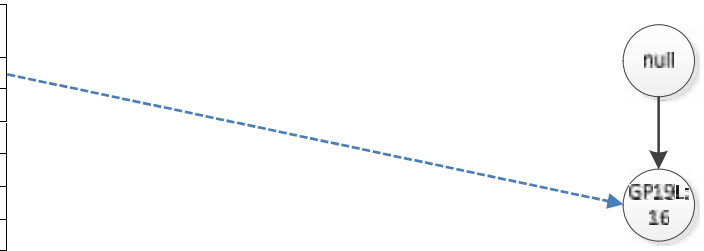
Kode barang	frekuensi	Node link
GP19L	13	•
AR500	12	•
165600		
AL1500		
AG240		
AM220		
AGU240		
AM240		
AR240		
AMGL19L		
AGU220		



Gambar 3.21 Lintasan yang mengandung kode barang AG240

10. *Conditional FP-Tree* untuk kode barang AR500 {GP19L:16}

Kode barang	frekuensi	Node link
GP19L	16	•
AR500		
165600		
AL1500		
AG240		
AM220		
AGU240		
AM240		
AR240		
AMGL19L		
AGU220		



Gambar 3.22 Lintasan yang mengandung kode barang AL1500

Setelah mendapatkan hasil *conditional fp-tree*, ditemukan beberapa *frequent patterns generated* yang keluar seperti pada tabel 3.17:

Tabel 3. 9 Hasil Perhitungan *Conditional FP-Tree*

Item	***Conditional FP Tree***
AGU220	{GP19L, AL1500:3}
AMGL19L	{GP19L:4}{165600:3}
AM240	{GP19L, AR500:4}
AGU240	{GP19L:9}{ AR500:7}{165600:8}{AL1500:7}
AM220	{GP19L, 165600:4}{ AR500, AG240, AL1500:6}
AG240	{GP19L:4}{ AR500, AL1500:6}{165600:5}
AL1500	{GP19L:10}{ AR500:8}{165600:12}
165600	{GP19L:13}{AR500:12}
AR500	{GP19L:16}

c. Tahap pembentukan *Frequent Patterns Generated*

Pembentukan *Frequent Patterns Generated* untuk setiap item berdasarkan *conditional FP-Tree* yang diperoleh. Berikut adalah hasil dari *frequent patterns generated:*

Tabel 3. 10 Tabel *Frequent Patterns Generated*

Item	***Frequent Patterns Generated***
AGU220	{AGU220,GP19L:3}{AGU220, AL1500:3}
AMGL19L	{AMGL19L, GP19L:4},{AMGL19L, 165600:3}
AM240	{AM240,GP19L:4},{AM240, AR500:4}
AGU240	{AGU240,GP19L:9},{AGU240, AR500:7},{AGU240,165600:8},{AGU240,AL1500:7}
AM220	{AM220,GP19L:4},{AM220, AR500:6},{AM220,AL1500:5}{AM220, 165600:4}{AM220, AG240:6}
AG240	{AG240,GP19L:4},{AG240, AR500:6},{AG240, 165600:5}{AG240, AL1500:6}
AL1500	{AL1500,GP19L:10},{AL1500, AR500:8}{AL1500, 165600:12}
165600	{ 165600, GP19L:13}{165600, AR500:12}
AR500	{AR500, GP19L:16}

2. *Generate Rule*.

Dari hasil *Frequent Patterns Generated* di atas, kemudian tahapan selanjutnya adalah pembentukan rule. Berikut adalah tabel dari pembentukan *generated rule*:

Tabel 3. 11 Tabel pembentukan *generated rule*

Rule	***Minimum Confidence***
AGU220 → GP19L	1/3*100= 33%
GP19L → AGU220	1/29*100= 3%
AGU220→ AL1500	2/3*100= 66%
AL1500 → AGU220	2/16*100= 12%
AMGL19L → GP19L	3/5*100= 60%
GP19L → AMGL19L	3/29*100= 10%
AMGL19L → 165600	3/5*100= 60%
165600 → AMGL19L	3/20*100= 15%
AMGL19L→ GP19L, 165600	2/5*100= 40%
AM240 → GP19L	5/9*100= 55%
GP19L → AM240	5/29*100= 17%

Rule	*Minimum Confidence*
AM240 → AR500	4/9*100= 44%
AR500 → AM240	4/26*100= 15%
AM240 → GP19L, AR500	5/9*100= 55%
AGU240 → GP19L	9/11*100= 81%
GP19L → AGU240	9/29*100= 31%
AGU240 → AR500	7/11*100= 63%
AGU240 → GP19L, AR500	5/11*100= 45%
AM220 → GP19L	4/13*100= 30%
GP19L → AM220	4/29*100= 13%
AM220 → 165600	4/13*100= 30%
165600 → AM220	4/20*100= 20%
AM220 → AR500	4/13*100= 30%
AR500 → AM220	4/26*100= 15%
AM220 → AG240	4/13*100= 30%
AG240 → AM220	4/14*100= 28%
AM220 → AL1500	4/13*100= 30%
AL1500 → AM220	4/16*100= 25%
AG240 → GP19L	4/14*100= 29%
GP19L → AG240	4/29*100= 14%
AG240 → AR500	5/14*100= 36%
AR500 → AG240	5/26*100= 14%
AG240 → AL1500	5/14*100= 38%
AL1500 → AG240	5/16*100= 31%
AG240 → 165600	5/14*100= 38%
165600 → AG240	5/20*100= 25%
AL1500 → GP19L	11/16*100= 68%
GP19L → AL1500	11/29*100= 38%
AL1500 → AR500	9/16*100= 56%
AR500 → AL1500	9/26*100= 35%
AL1500 → 165600	14/16*100= 88%
165600 → AL1500	14/20*100= 70%
165600 → GP19L	12/20*100= 60%
GP19L → 165600	12/29*100= 41%
165600 → AR500	8/20*100= 40%
AR500 → 165600	8/26*100= 31%
165600 → GP19L, AR500	11/20*100= 55%

Rule	*Minimum Confidence*
AR500 → GP19L	17/26*100= 65%
GP19L → AR500	17/29*100= 59%

a. Setelah diperoleh nilai *confidence* untuk setiap *rule*, maka tahap selanjutnya adalah menentukan *rule* dari *frequent patterns generated*. Untuk setiap pasangan yang tidak memenuhi syarat nilai *confidence* >= 60% maka akan dihilangkan. Hasil dari pembentukan *rule* dalam penentuan barang-barang dalam bentuk awal *confidence*. Dapat dilihat pada tabel berikut.

Tabel 3. 12 *Rules* yang memenuhi *minimum confidence*

Rule	*Minimum Confidence*
AGU220→ AL1500	2/3*100= 66%
AMGL19L → GP19L	3/5*100= 60%
AMGL19L → 165600	3/5*100= 60%
AGU240 → GP19L	9/11*100= 81%
AGU240 → AR500	7/11*100= 63%
AL1500 → GP19L	11/16*100= 68%
AL1500 → 165600	14/16*100= 88%
165600 → AL1500	14/20*100= 70%
165600 → GP19L	12/20*100= 60%
AR500 → GP19L	17/26*100= 65%

Setelah mendapatkan *rule* yang memenuhi *minimum confidence*, maka didapatkan hasil pemaketan *rule* seperti yang terdapat pada tabel 3.22, karena *rule* yang dihasilkan diatas masih berupa kode barang, maka dilakukan tahap konversi terhadap nama barang agar dapat diketahui nama barang apa saja yang masuk kedalam isi paket maka, kita membutuhkan tabel nama barang yang dapat dilihat pada tabel 3.21.

Tabel 3. 13 Tabel nama barang

KODE BARANG	NAMA BARANG	KMS	HARGA
165600	165 botol 600 ml	DUS	2,500
AL1500	Alto botol 1500 ml	DUS	3,000
AM220	Amidis cup 220 ml	DUS	700
AM240	Amidis cup 240 ml	DUS	1,000
AGU220	Aguatic cup 220 ml	DUS	700
AGU240	Aguatic cup 240 ml	DUS	1,000
AMGL19L	Amidis galon 19l	PCS	5,000
GP19L	Galon patisa 19l	PCS	6,500
AR500	Arvin botol 500 ml	DUS	2,000
AR600	Arvin botol 600 ml	DUS	2,500
FU500	Fujitso botol 500 ml	DUS	2,000
AG220	Aguaria cup 220 ml	DUS	700
AG240	Aguaria cup 240 ml	DUS	1,000
FU330	Fujitso botol 330 ml	DUS	1,500
1651500	165 botol 1500 ml	DUS	3,000
AR240	Arvin cup 240 ml	DUS	1,000
AR220	Arvin cup 220 ml	DUS	700

b. Tahapan selanjutnya yaitu menampilkan informasi paket yang dihasilkan dari *rules* yang terbentuk sebelumnya kedalam bentuk yang mudah dimengerti.

Tabel 3. 14 Hasil *rules*

Rule	Paket barang
AGU220→ AL1500	Jika Aguatic cup 220ml dibeli, maka Alto botol 1500ml dibeli dengan confidence 66%
AMGL19L → GP19L	Jika Amidis gallon 19L dibeli, maka Galon patisa 19L dibeli dengan confidence 60%
AMGL19L → 165600	Jika Amidis gallon 19L dibeli, maka 165 600ml dibeli dengan confidence 60%
AGU240 → GP19L	Jika Aguatic cup 240ml dibeli, maka Galon patisa 19L dibeli dengan confidence 81%
AGU240 → AR500	Jika Aguatic cup240ml dibeli, maka Arvin500 dibeli dengan confidence 63%
AL1500 → GP19L	Jika Alto botol 1500ml dibeli, maka Galon patisa 19L dibeli dengan confidence 68%
AL1500 → 165600	Jika Alto botol 1500ml dibeli, maka 165 600ml

Rule	Paket barang
	dibeli dengan confidence 88%
165600 → AL1500	Jika 165 botol 600ml dibeli, maka Alto botol 1500ml dibeli dengan confidence 70%
165600 → GP19L	Jika 165 botol 600ml dibeli, maka Galon patisa 19L dibeli dengan confidence 60%
AR500 → GP19L	Jika Arvin botol 500ml dibeli, maka Galon patisa 19L dibeli dengan confidence 65%

Dalam penentuan kombinasi barang yang akan dijual dalam bentuk paket pihak PT. PATISA CHAKRA MANDIRI memiliki batasan khusus:

1. Harga per paket didapatkan dari pengurangan ke ribuan atau puluhribuan terdekat. Misalkan jumlah harga dari paket 3 sebesar Rp. 27.000,. maka akan dikurangi menjadi Rp. 25.000,.
2. Harga per paket minimal sebesar Rp. 20.000,. dan maksimal Rp. 75.000,.
3. Jumlah barang yang dapat dipaketkan yaitu sebanyak 2 sampai 5 barang.
4. Jumlah paket yang tersedia dalam 1 per 3 bulan yaitu sebanyak 3 sampai 5 paket

Dari batasan yang dimiliki oleh PT. PATISA CHAKRA MANDIRI maka didapatkan rekomendasi paket sebagai berikut:

Tabel 3. 15 Informasi Paket

PAKET	PAKET BARANG
PAKET 1	• ARVIN BOTOL 500 ml • GALON PATISA 19L • Rekomendasi harga per paket Rp. 43.000,.
PAKET 2	• ALTO BOTOL 1500 ml • GALON PATISA 19L • Rekomendasi harga per paket Rp. 53.000,.
PAKET 3	• ARVIN CUP 240ml • GALON PATISA 19L • Rekomendasi harga per paket Rp. 29.000,.

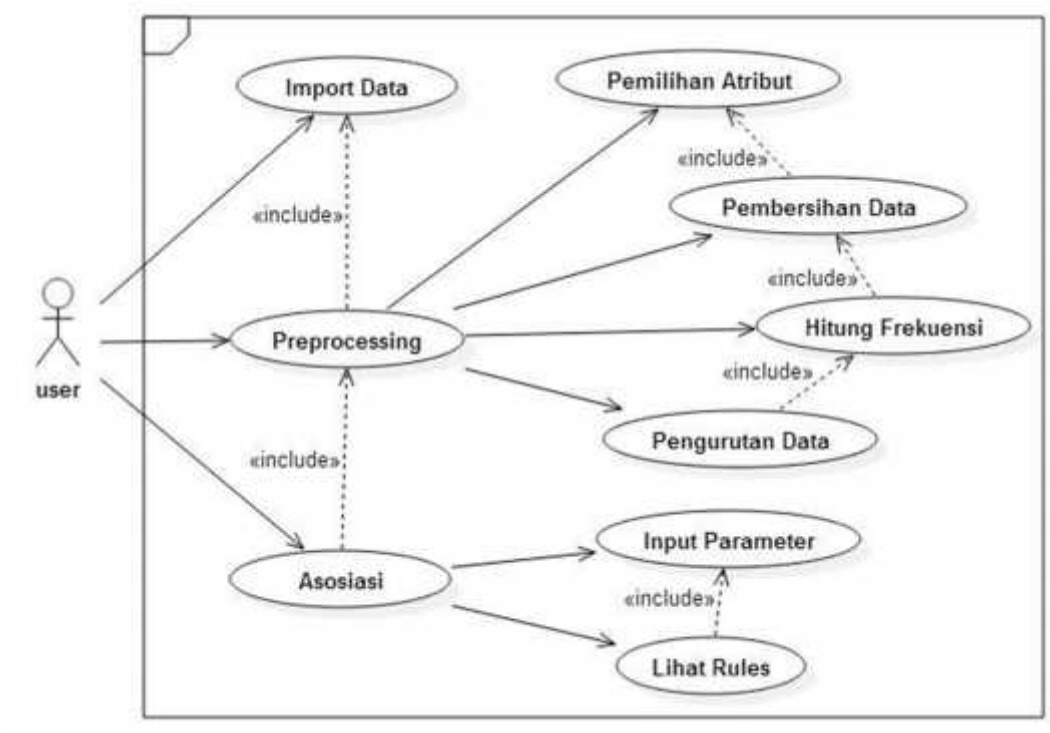
Dari informasi diatas dapat dilihat harga paket yang tertera hanya merupakan rekomendasi harga paket berdasarkan harga jual, untuk harga minimum yaitu pada paket 3 dengan barang Arvin cup 240 ml dan Galon patisa 19 l dengan harga Rp. 29.000,. sedangkan untuk harga maksimum terdapat pada paket 2 dengan barang Alto botol 1500 ml dikombinasikan dengan Galon patisa 19 l dengan harga Rp. 53.000,. Untuk menentukan harga paket akan dikembalikan pada kebutuhan pihak PT. PATISA CHAKRA MANDIRI.

3.1.3 Analisis Kebutuhan Fungsional

Analisis kebutuhan fungsional bertujuan untuk perancangan terhadap aplikasi yang akan dibangun. Aplikasi yang akan dibangun menggunakan pendekatan berorientasi objek dengan menggunakan pemodelan UML. Pemodelan yang akan digunakan untuk memodelkan terdiri dari diagram *usecase*, *activity diagram*, *sequence diagram* dan *class diagram*.

3.1.3.1 Diagram *Use Case*

Use case atau diagram *use case* merupakan pemodelan untuk kelakuan (behavior) sistem yang akan dibuat. Diagram *use case* yang terdapat pada sistem yang akan dibangun terdiri dari satu *user* dan 9 *use case*. Adapun diagram *use case* dari sistem yang akan dibuat dapat dilihat di gambar 3.23.



Gambar 3. 23 Diagram *Use Case* Pada Sistem *Data Mining*

Tabel 3. 15 Tabel Definisi *User*

User	**Deskripsi**
User	*User* bertugas untuk melakukan *import* data, memasukkan *Minimum Support* dan *Minimum Confidence*

No	Menu	Deskripsi	Nama File
1	Utama	Tampilan Menu Utama pada sistem	HomeForm.cs
2	Import Data	Digunakan untuk mengimport data transaksi yang dilakukan oleh user	PersiapanForm.cs
3	Preprocessing Data	Menu pada program yang digunakan *user* untuk melakukan proses pemilihan atribut sesuai dengan informasi aturan asosiasi yang akan dihasilkan, membersihkan data hasil pemilihan atribut, hitung kemunculan tiap item dan pengurutan data berdasarkan *priority* untuk dilanjutkan ke proses asosiasi	PreprocessingForm.cs

No	Menu	Deskripsi	Nama File
4	Asosiasi	Digunakan untuk menganalisis data berdasarkan hasil *preprocessing* dengan menginputkan nilai *Minimum Support* dan *Minimum Confidence* serta menampilkan hasil akhir dari proses asosiasi yang telah dianalisis.	AsosiasiForm.cs
5	Batasan	Digunakan untuk memberi batasan pada paket yang sudah terbentuk yang telah disesuaikan dengan kebutuhan pengguna	BatasanForm.cs

Tabel 3. 16 Deskripsi *Use Case*

3.1.3.2 Skenario *Use Case*

Skenario *use case* menjelaskan skenario dari setiap proses yang digambarkan pada diagram *use case*. Berikut ini skenario *use case* dari gambar 3.16

1. **Skenario *Use Case Import Data***

   Skenario *use case import* data menggambarkan langkah – langkah aksi user terhadap sistem untuk melakukan *import* data transaksi penjualan yang akan disimpan ke dalam *database*.

Tabel 3. 17 *Requirement* A.1

Requirement A.1
Sistem menyediakan menu *import data* untuk melakukan *import* data transaksi kedalam *database* sebelum melakukan *preprocessing* dan asosiasi

Tabel 3. 18 Skenario *Use Case Import Data*

Use case Name	***Import Data***
Related Requirements	Requirement *A.1*
Goal In Context	*Import* data transaksi penjualan dan menyimpannya ke dalam *database*
Precondition	*User* menyiapkan data transaksi
Successful End Condition	Data transaksi yang di *import* oleh *user* berhasil disimpan ke dalam *database*
Failed End Condition	Data transaksi gagal disimpan ke dalam *database*

Actors	*User*	
Triger	*User* memasukan data transaksi	
Included Cases	-	
Main Flow	Step	Action
	1	*User* melakukan *request import* data
	2	*User* memasukkan data transaksi
	3	Sistem melakukan *generate file*
	4	Sistem menampilkan data di *gridview*
	5	Data disimpan ke *database*
	6	Sistem menampilkan pesan data tersimpan
Extension	Step	Branching Action
	6.1	Menampilkan pesan kesalahan dalam proses *import* data

2. Skenario *Use Case Preprocesing*

Skenario *use case preprocessing* menjelaskan langkah-langkah untuk melakukan proses Pemilihan Atribut, Pembersihan Data, Hitung Frekuensi dan Pengurutan Data.

Tabel 3. 19 *Requirement* A.2

Requirement A.2
Sistem menyediakan menu *preprocesing* untuk melakukan pemilihan atribut, pembersihan data, hitung frekuensi dan pengurutan data.

Tabel 3. 20 Skenario *Use Case Preprocesing*

Use case Name	***Preprocesing***
Related Requirements	Requirement *A.1,* Requirement *A.2*
Goal In Context	Sistem melakukan proses pemilihan atribut, pembersihan data, hitung frekuensi dan pengurutan data
Precondition	Data transaksi sudah tersimpan didalam *database*
Successful End Condition	Sistem berhasil melakukan proses pemilihan atribut, pembersihan data, hitung frekuensi dan pengurutan data
Failed End Condition	Sistem gagal untuk melakukan proses pemilihan atribut, pembersihan data, hitung frekuensi dan pengurutan data
Actors	*User*
Triger	*User* melakukan *request* proses pemilihan atribut, pembersihan data, hitung frekuensi dan pengurutan

	data kepada sistem	
Included Cases	Pemilihan Atribut, Pembersihan Data, Hitung Frekuensi dan Pengurutan Data	
Main Flow	Step	Action
	1	*User* melakukan *request preprocessing*
	2	*User* melakukan *request* Pemilihan Atribut
	3	Sistem melakukan proses pemilihan atribut
	4	*User* melakukan *request* pembersihan data
	5	Sistem melakukan proses pembersihan data
	6	*User* melakukan *request* hitung frekuensi
	7	Sistem melakukan proses hitung frekuensi
	8	*User* melakukan *request* pengurutan data
	9	Sistem melakukan proses pengurutan data
Extension	Step	Branching Action
	-	-

3. Skenario *Use Case* Pemilihan Atribut

Skenario *Use Case* Pemilihan Atribut menggambarkan langkah-langkah melakukan pemilihan atribut terhadap data transaksi yang telah melewati proses *import data*.

Tabel 3. 21 Skenario *Use Case* Pemilihan Atribut

Use case Name	***Pemilihan Atribut***	
Related Requirements	Requirement *A.1*, Requirement *A.2*	
Goal In Context	Sistem melakukan pemilihan atribut noNota dan Kode Barang	
Precondition	Data transaksi tersimpan didalam *database*	
Successful End Condition	Sistem berhasil memilih atribut noNota dan Kode Barang	
Failed End Condition	Sistem gagal melakukan Pemilihan Atribut noNota dan Kode Barang	
Actors	*User*	
Triger	*User* melakukan *request* Pemilihan Atribut kepada system	
Included Cases	-	
Main Flow	Step	Action
	1	*User request preprocessing*
	2	*User* melakukan *request* pemilihan atribut
	3	Sistem menampilkan hasil pemilihan atribut
	4	Sistem menampilkan pesan berhasil melakukan proses pemilihan atribut
Extension	Step	Branching Action

	4.1	Sistem manampilkan pesan gagal dalam melakukan proses pemilihan atribut

4. Skenario *Use Case* Pembersihan Data

Skenario *use case* pembersihan data menggambarkan langkah-langkah melakukan pembersihan terhadap noNota dan Kode Barang yang memiliki *item* tunggal dan telah melewati proses Pemilihan Atribut.

Tabel 3. 22 Skenario *Use Case* Pembersihan Data

Use case Name	***Pembersihan Data***	
Related Requirements	Requirement *A.1,* Requirement *A.2*	
Goal In Context	Sistem melakukan pembersihan pada atribut noNota dan Kode Barang yang memiliki item tunggal.	
Precondition	Data Transaksi sudah melalui proses Pemilihan Atribut	
Successful End Condition	Sistem berhasil melakukan pembersihan pada atribut noNota dan Kode Barang yang memiliki item tunggal	
Failed End Condition	Sistem gagal dalam melakukan pembersihan pada atribut noNota dan Kode Barang yang memiliki item tunggal	
Actors	*User*	
Triger	*User* melakukan *request* Pembersihan Data pada system	
Included Cases	-	
Main Flow	Step	Action
	1	*User* melakukan *request* pembersihan data
	2	Sistem menghapus data item tunggal
	3	Sistem menampilkan pesan berhasil melakukan proses pembersihan data
	4	Sistem menyimpan data hasil pembersihan data ke *database*
	5	Sistem menampilkan hasil proses pembersihan data
Extension	Step	Branching Action
	5.1	Sistem menampilkan pesan gagal menampilkan proses pembersihan data

5. Skenario *Use Case* Hitung Frekuensi

Skenario *Use Case* Hitung Frekuensi menggambarkan langkah-langkah melakukan perhitungan jumlah kemunculan tiap *item* dalam data transaksi dan telah melewati proses Pembersihan Data.

Tabel 3. 23 Skenario *Use Case* Hitung Frekuensi

Use case Name	**Hitung Frekuensi**	
Related Requirements	Requirement *A.1,* Requirement *A.2*	
Goal In Context	Sistem melakukan perhitungan kemunculan tiap item pada data transaksi .	
Precondition	Data Transaksi sudah melalui proses Pembersihan Data	
Successful End Condition	Sistem berhasil melakukan perhitungan jumlah kemunculan tiap item pada data transaksi	
Failed End Condition	Sistem gagal dalam melakukan perhitungan jumlah kemunculan tiap item pada data transaksi	
Actors	*User*	
Triger	*User* melakukan *request* Hitung Frekuensi pada system	
Included Cases	-	
Main Flow	Step	Action
	1	*User* melakukan *request* hitung frekuensi
	2	Sistem melakukan proses hitung frekuensi tiap item
	3	Sistem menampilkan pesan berhasil melakukan proses hitung frekuensi
	4	Sistem menampilkan hasil proses hitung frekuensi
Extension	Step	Branching Action
	4.1	Sistem menampilkan pesan gagal melakukan proses hitung frekuensi

6. Skenario *Use Case* Pengurutan Data

Skenario *Use Case* Pengurutan Data menggambarkan langkah-langkah melakukan pengurutan data berdasarkan *priority* dan telah melewati proses Hitung Frekuensi .

Tabel 3. 24 Skenario *Use Case* Pengurutan Data

Use case Name	**Pengurutan Data**
Related Requirements	Requirement *A.1,* Requirement *A.2*
Goal In Context	Sistem melakukan pengurutan data berdasarkan *priority* .
Precondition	Data Transaksi sudah melalui proses Hitung Frekuensi
Successful End Condition	Sistem berhasil melakukan pengurutan data berdasarkan *priority*
Failed End Condition	Sistem gagal dalam melakukan pengurutan data berdasarkan *priority*

Actors	*User*	
Triger	*User* melakukan *request* Pengurutan Data pada system	
Included Cases	-	
Main Flow	Step	Action
	1	User melakukan request pengurutan data
	2	Sistem melakukan proses pengurutan data berdasarkan *priority*
	3	Sistem menampilkan pesan berhasil melakukan proses pengurutan data
	4	Sistem menampilkan hasil pengurutan data
Extension	Step	Branching Action
	4.1	Sistem menampilkan pesan gagal melakukan proses pengurutan data

7. Skenario *Use Case* Asosiasi

Skenario *Use Case* Assosiasi menggambarkan langkah-langkah aksi *user* untuk melakukan proses *Input* Parameter dan *Rules.*

Tabel 3. 25 *Requirement* A.3

Reqirement A.3
Sistem menyediakan menu asosiasi untuk melakukan input parameter berupa *Minimum Support* dan *Minimum Confidence*, Eliminasi untuk mengeliminasi item apasaja yang tidak memenuhi nilai parameter yang diinputkan dan *Rules* untuk melihat *rule* apa saja yang dihasilkan.

Tabel 3. 26 Skenario *Use Case* Asosiasi

Use case Name	**Asosiasi**
Related Requirements	Requirement *A.1*, Requirement *A.2*, Requirement *A.3*
Goal In Context	Sistem mendapatkan *rule* yang akan ditampilkan dalam bentuk informasi paket air dalam kemasan
Precondition	Data sudah melalui proses *preprocessing*
Successful End Condition	Sistem berhasil mendapatkan *rule* yang akan ditampilkan dalam bentuk informasi paket air dalam kemasan
Failed End Condition	Sistem gagal mendapatkan *rule* yang akan ditampilkan dalam bentuk informasi paket air dalam kemasan
Actors	*User*
Triger	*User* menginputkan nilai parameter berupa nilai *Minimum Support* dan *Minimum Confidence*
Included Cases	Input Parameter, Rules

Main Flow	Step	Action
	1	*User* melakukan *request* asosiasi
	2	*User* memasukkan nilai parameter
	3	User melakukan *request* hasil analisa
	4	Sistem melakukan *validasi* inputan user
	5	Sistem melakukan pembentukan *rule*
	6	Sistem menampilkan hasil *rule*
Extension	Step	Branching Action
	4.1	Sistem menampilkan pesan nilai *Minimum Support* yang dimasukkan tidak sesuai
	6.1	Sistem menampilkan pesan bahwa *rule* tidak ditemukan

8. Skenario *Use Case Input* Parameter

Skenario *use case Input* Parameter menggambarkan langkah-langkah aksi *user* untuk melakukan *input* nilai *Minimum Support* dan *Minimum Confidence* sebagai pendukung dalam pencarian *frequent itemset*.

Tabel 3. 27 Skenario *Use Case Input* Parameter

Use case Name	**Input Parameter**	
Related Requirements	Requirement *A.3*	
Goal In Context	Sistem mendapatkan nilai inputan *Minimum Support* dan *Minimum Confidence* dari *user*	
Precondition	*User* memasukkan nilai *Minimum Support* dan *Minimum Confidence*	
Successful End Condition	Sistem berhasil mendapatkan nilai *Minimum Support* dan *Minimum Confidence* sesuai dengan nilai yang dimasukkan *user*	
Failed End Condition	Sistem gagal mendapatkan nilai *Minimum Support* dan *Minimum Confidence* dari *user*	
Actors	*User*	
Triger	*User* memasukkan nilai *Minimum Support* dan *Minimum Confidence* yang sesuai	
Main Flow	Step	Action
	1	*User* melakukan *request* asosiasi
	2	User memasukkan nilai *minimum support* dan *minimum confidence*
	3	Sistem melakukan *validasi* inputan user
	4	Sistem menampilkan nilai *minimum support* dan *minimum confidence*
Extension	Step	Branching Action
	3.1	Sistem menampilkan pesan kesalahan nilai *Minimum Support* dan *Minimum Confidence* yang dimasukkan tidak sesuai

9. Skenario *Use Case Rules*

Skenario *Use Case Rules* menggambarkan langkah-langkah aksi *user* untuk melakukan proses *Rules*.

Tabel 3. 28 Skenario *Use Case Rules*

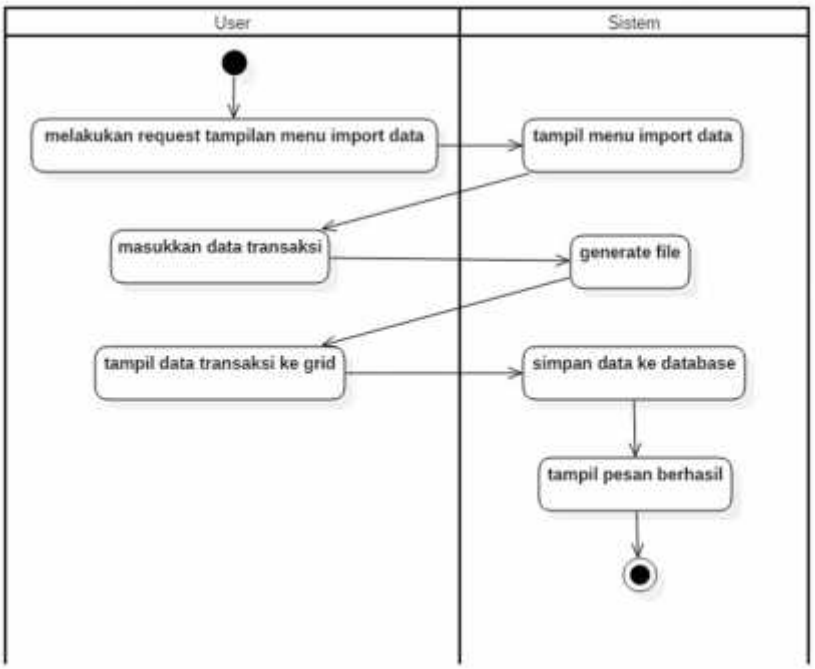
Use case Name	**Rules**	
Related Requirements	Requirement *A.1*, Requirement *A.2*, Requirement *A.3*	
Goal In Context	Sistem menampilkan informasi paket air dalam kemasan hasil dari pencarian *frequent itemset* dan pembentukan *rule*	
Precondition	User sudah melakukan memasukkan nilai *Minimum Support* dan *Minimum Confidence*	
Successful End Condition	Sistem berhasil menampilkan informasi paket air dalam kemasan hasil dari pencarian *frequent itemset* dan pembentukan *rule*	
Failed End Condition	Sistem gagal menampilkan informasi paket air dalam kemasan hasil dari pencarian *frequent itemset* dan pembentukan *rule*	
Actors	*User*	
Triger	*User* memasukkan nilai *Minimum Support* dan *Minimum Confidence*	
Main Flow	Step	Action
	1	Sistem melakukan *request* data *confidence item*
	2	Sistem melakukan proses pembuatan *Rules*
	3	Sistem melakukan seleksi terhadap *item* yang memenuhi nilai parameter
	4	Sistem menyimpan hasil seleksi *item* ke *database*
	5	Sistem menampilkan hasil *rules*, nilai support dan confidence tiap item
	6	User melihat *rules* yang terbentuk
Extension	Step	Branching Action
	5.1	Sistem menampilkan pesan bahwa *rule* tidak ditemukan

3.1.4 *Activity* Diagaram

Activity Diagram menggambarkan rangkaian aliran dari aktivitas dari setiap proses. Aktivitas diagram dari setiap proses dapat dilihat pada gambar-gambar berikut :

1. *Activity* Diagram *Import Data*

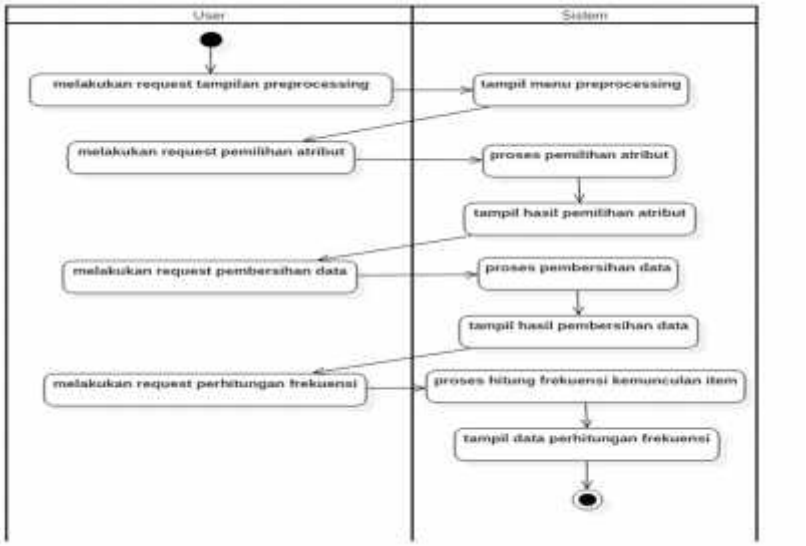
   Activity diagram *import data* menjelaskan aktivitas *user* dalam melakukan proses *import data*. *Activity* diagram *import data* dapat dilihat pada gambar 3.24



Gambar 3. 24 *Activity* diagram *import data*

2. *Activity* Diagram *Preprocessing*

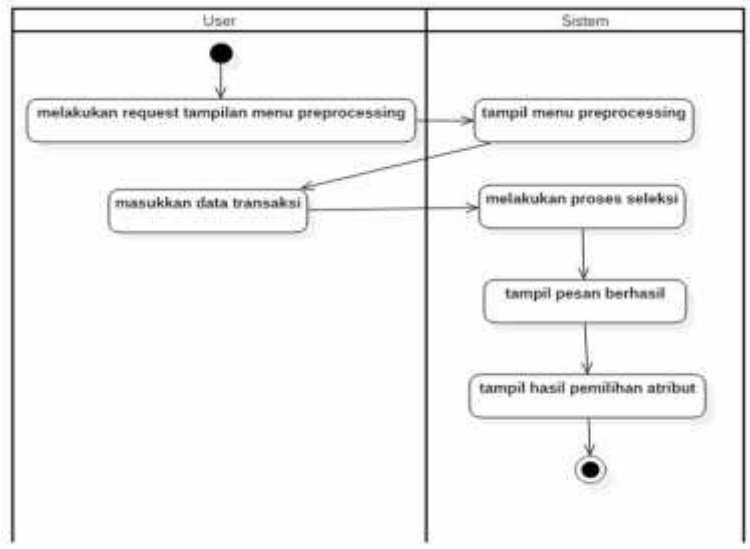
Activity diagram *preprocessing* menjelaskan aktivitas sistem dalam melakukan proses Pemilihan Atribut dan Pembersihan Data. *Activity* diagram *preprocessing* dapat dilihat pada gambar 3.25



Gambar 3. 25 *Activity* diagram *preprocessing*

3. *Activity* Diagram Pemilihan Atribut

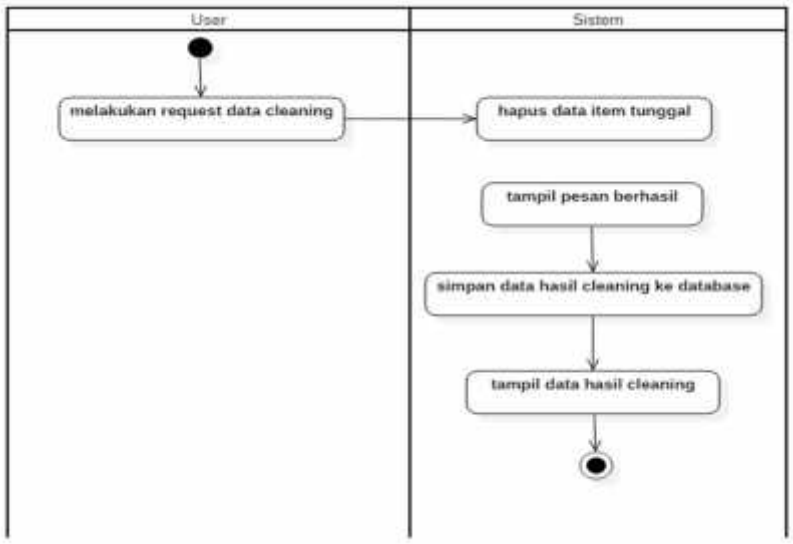
Activity diagram Pemilihan Atribut menjelaskan aktivitas sistem dalam melakukan proses data Pemilihan Atribut . *Activity* diagram data Pemilihan Atribut dapat dilihat pada gambar 3.26



Gambar 3. 26 *Activity* diagram pemilihan atribut

4. *Activity* Diagram Pembersihan Data

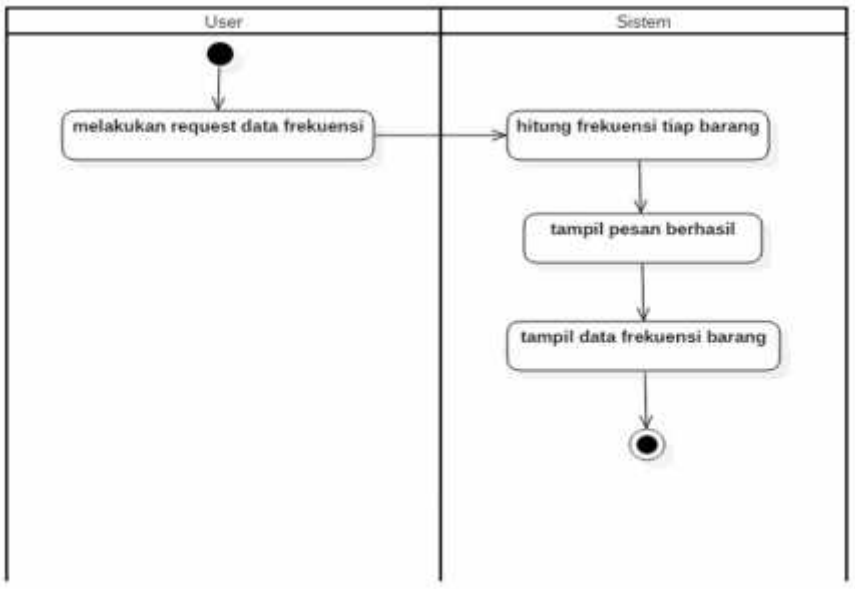
Activity diagram data Pembersihan Data menjelaskan aktivitas sistem dalam melakukan proses data Pembersihan Data. *Activity* diagram data Pembersihan Data dapat dilihat pada gambar 3.27



Gambar 3. 27 *Activity* diagram Pembersihan Data

5. *Activity* Diagram Hitung Frekuensi

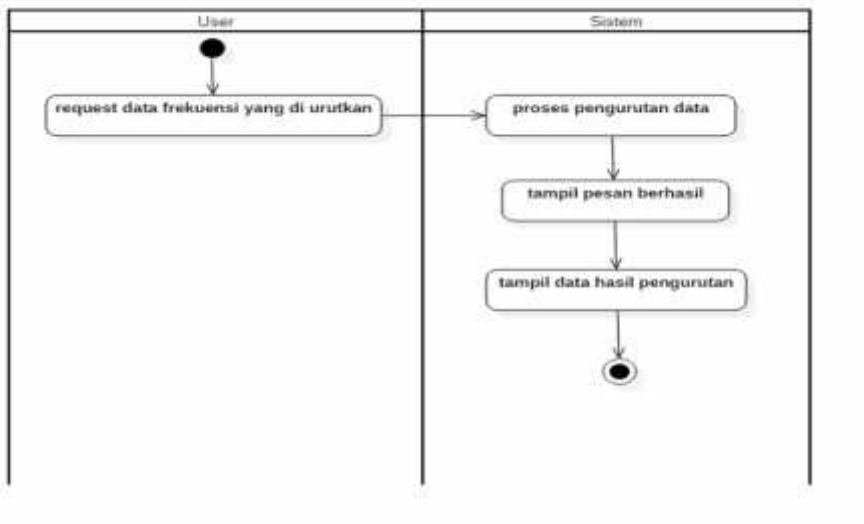
Activity diagram Hitung Frekuensi menjelaskan aktivitas *user* dan sistem dalam melakukan proses hitung frekuensi. *Activity* diagram Hitung Frekuensi dapat dilihat pada gambar 3.28



Gambar 3. 28 *Activity* diagram Hitung Frekuensi

6. *Activity* Diagram Pengurutan Data

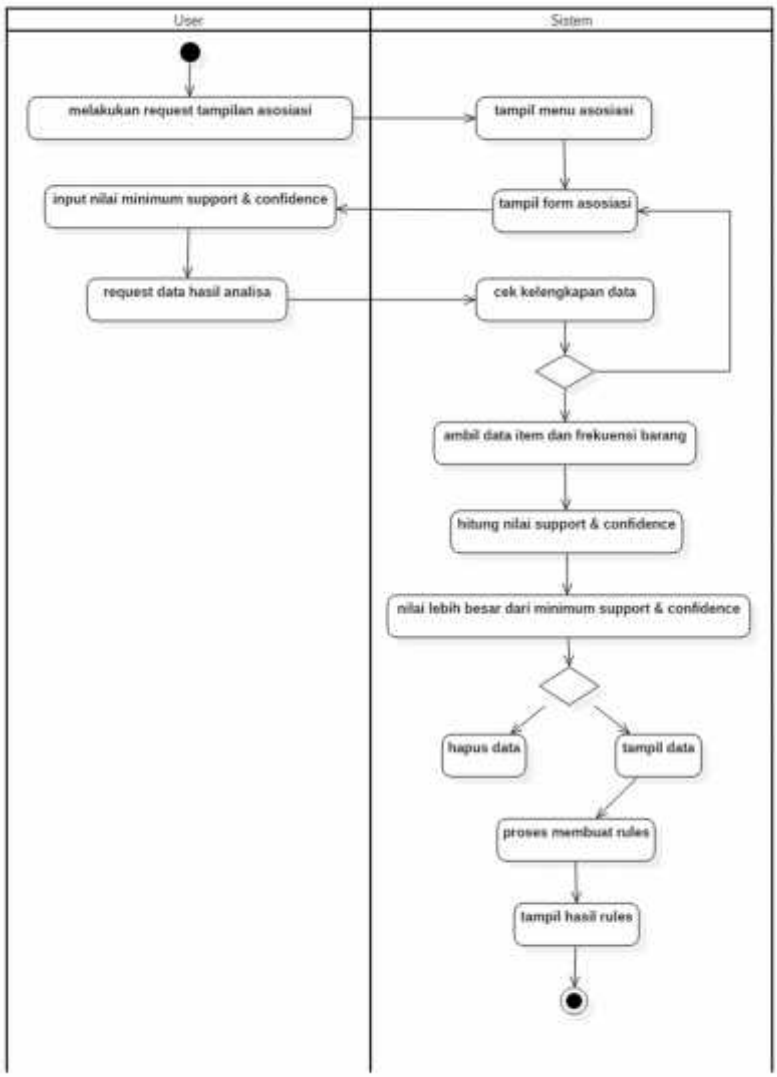
Activity diagram Pengurutan Data menjelaskan aktivitas *user* dan sistem dalam melakukan proses Pengurutan Data. *Activity* diagram Pengurutan Data dapat dilihat pada gambar 3.29



Gambar 3. 29 *Activity* diagram Pengurutan Data

7. *Activity* Diagram Asosiasi

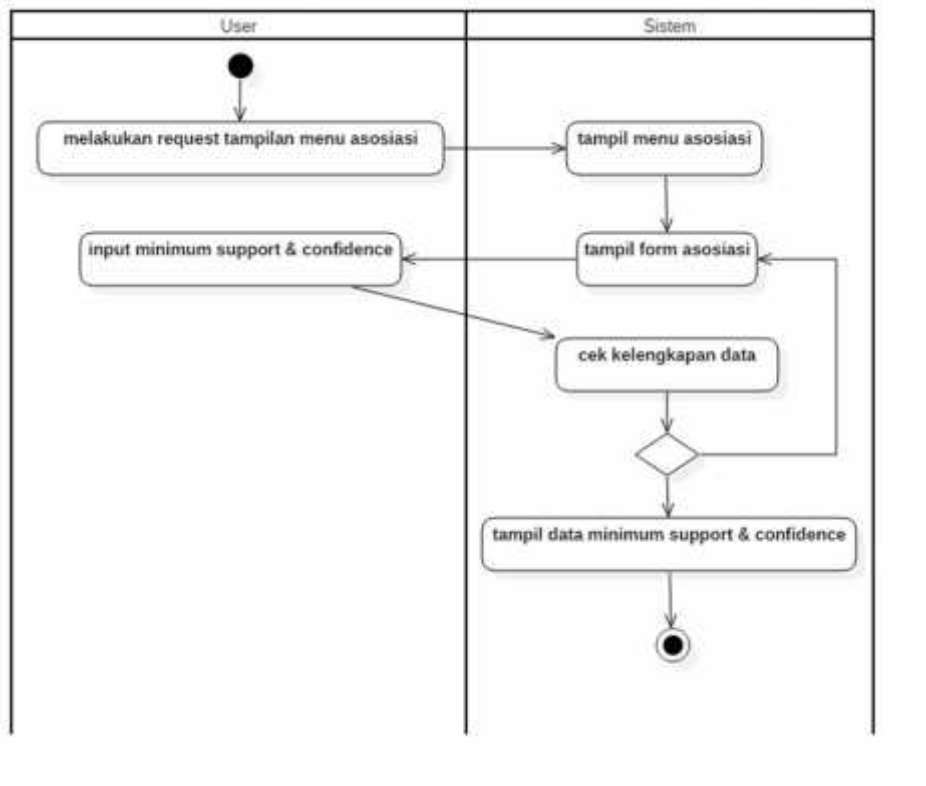
Activity diagram asosiasi menjelaskan aktivitas *user* dan sistem dalam melakukan proses *input* parameter dan melihat *rule*. *Activity* diagram Asosiasi dapat dilihat pada gambar 3.30



Gambar 3. 30 *Activity* diagram asosiasi

8. *Activity* Diagram *Input* Parameter

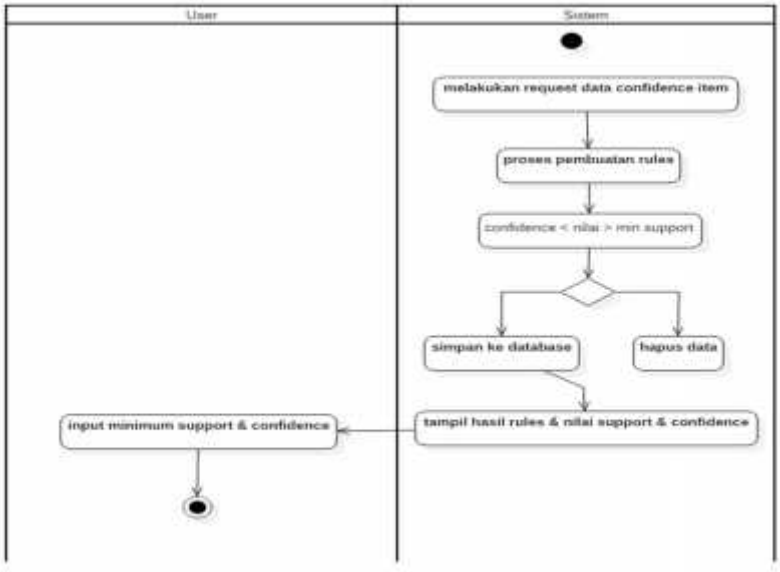
Activity diagram *input* parameter menjelaskan aktivitas *user* dan sistem dalam melakukan proses memasukkan nilai *Minimum Support* dan *Minimum Confidence*. *Activity* diagram *Input* Parameter dapat dilihat pada gambar 3.31



Gambar 3. 31 *Activity* diagram input parameter

9. *Activity* Diagram Lihat *Rules*

Activity diagram lihat *rules* menjelaskan aktivitas *user* dan sistem dalam menampilkan hasil seleksi *rule* sesuai nilai *Minimum confidence* yang dimasukkan oleh *user*. *Activity* diagram lihat hasil asosiasi dapat dilihat pada gambar 3.32

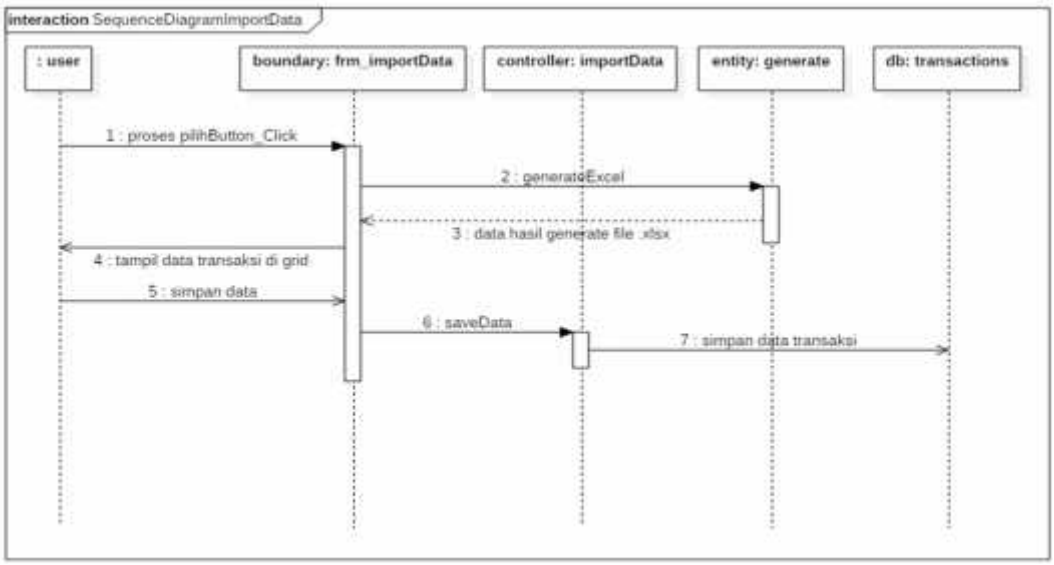


Gambar 3. 32 *Activity* diagram lihat *rules*

3.1.5 *Sequence* Diagram

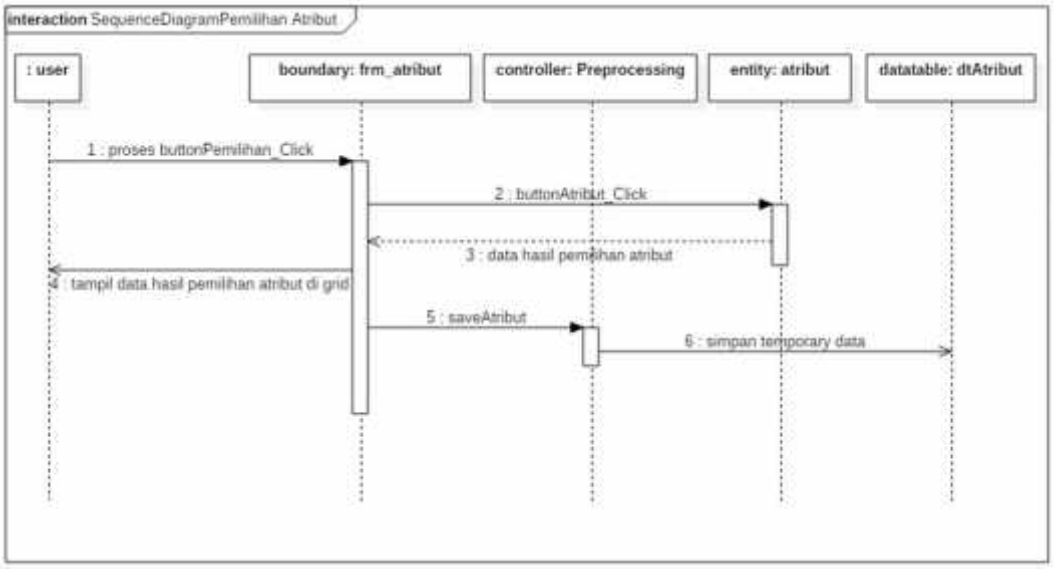
Sequence diagram digunakan untuk menggambarkan arus pekerjaan, pesan yang disampaikan dan bagaimana elemen-elemen di dalamnya bekerja sama dari waktu ke waktu untuk mencapai tujuan hasil. Berikut adalah *sequence* diagram pada aplikasi *data mining* yang sedang dibangun pada Gambar 3.39 sampai dengan Gambar 3.33

1. *Sequence* Diagram *Import Data*



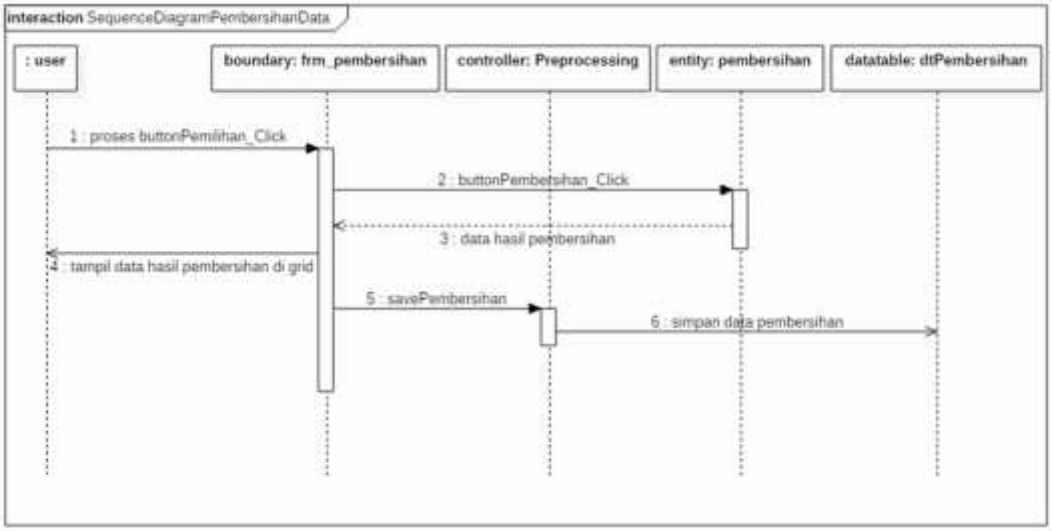
Gambar 3. 33 *Sequence* Diagram *Import Data*

2. *Sequence* Diagram *Preprocessing*
   a. *Sequence* Diagram Data Pemilihan Atribut



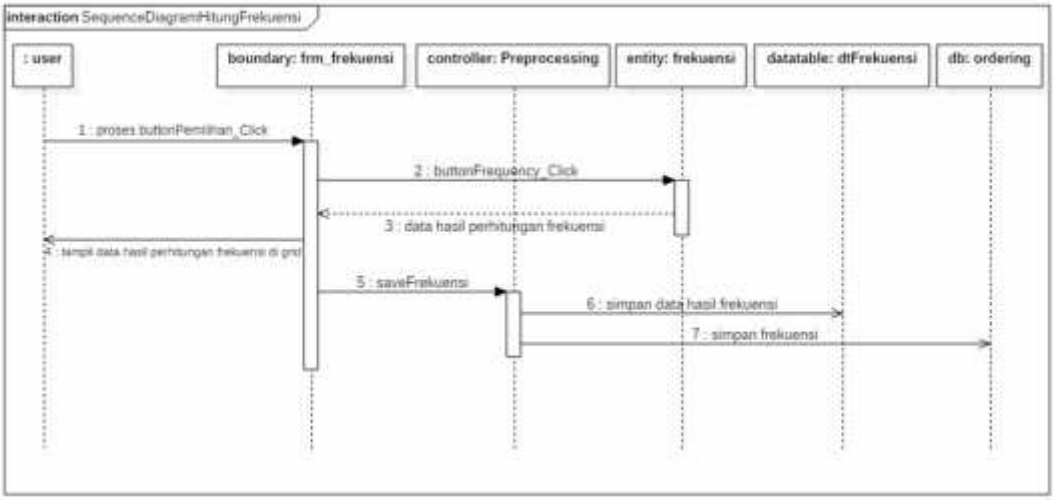
Gambar 3. 34 *Sequence* Diagram Data Pemilihan Atribut

b. *Sequence* Diagram Pembersihan Data



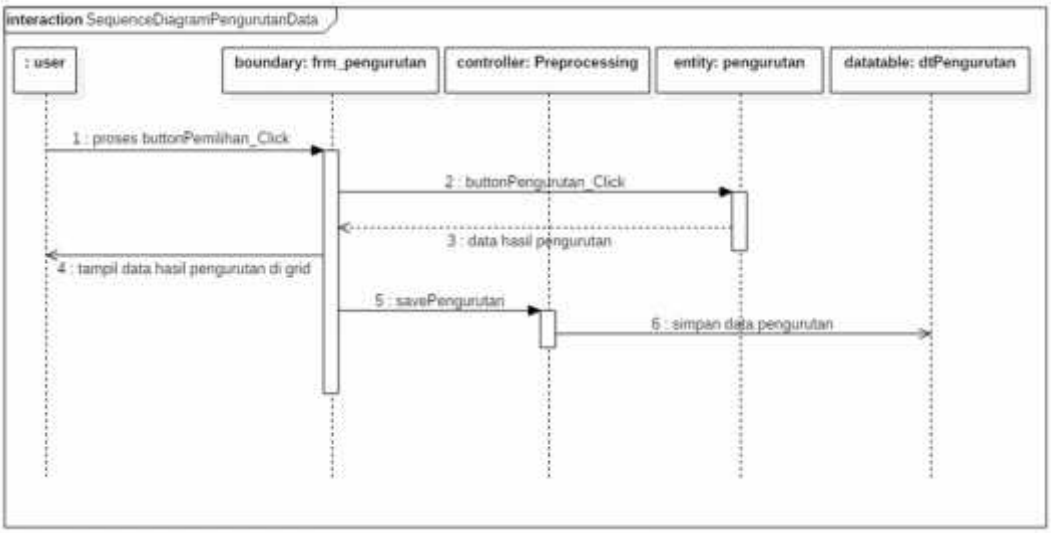
Gambar 3. 35 *Sequence* Diagram Pembersihan Data

c. *Sequence* Diagram Hitung Frekuensi



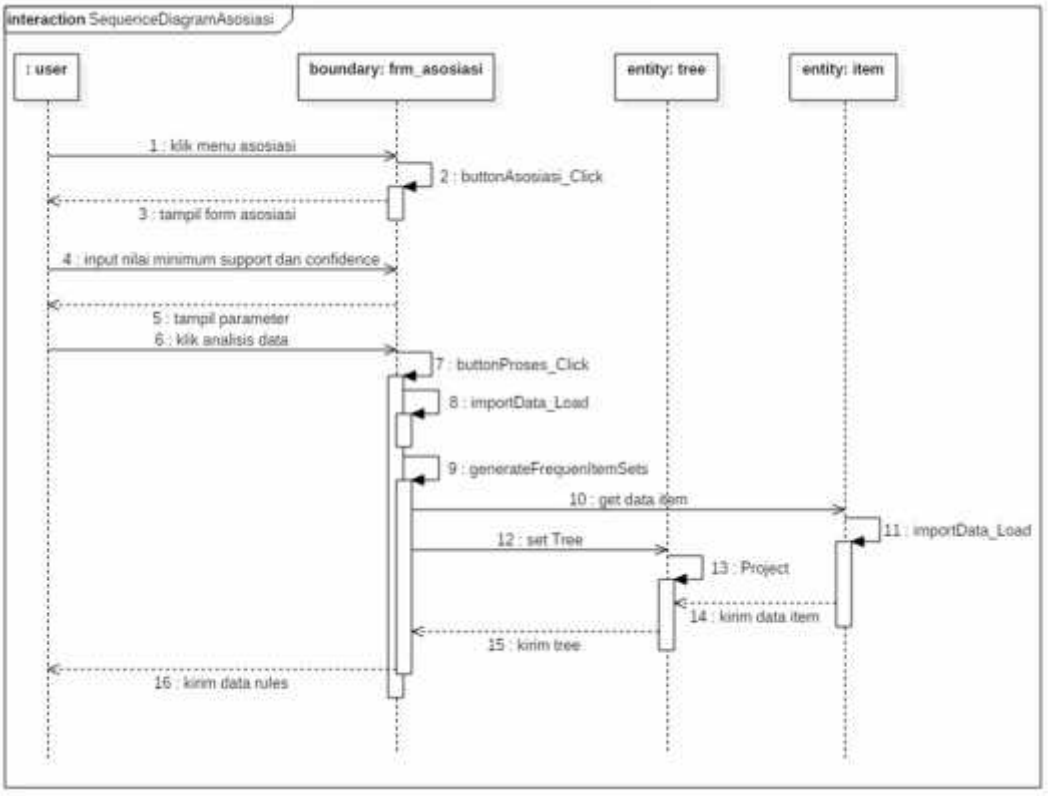
Gambar 3. 36 *Sequence* Diagram Hitung Frekuensi

d. *Sequence* Diagram Pengurutan Data



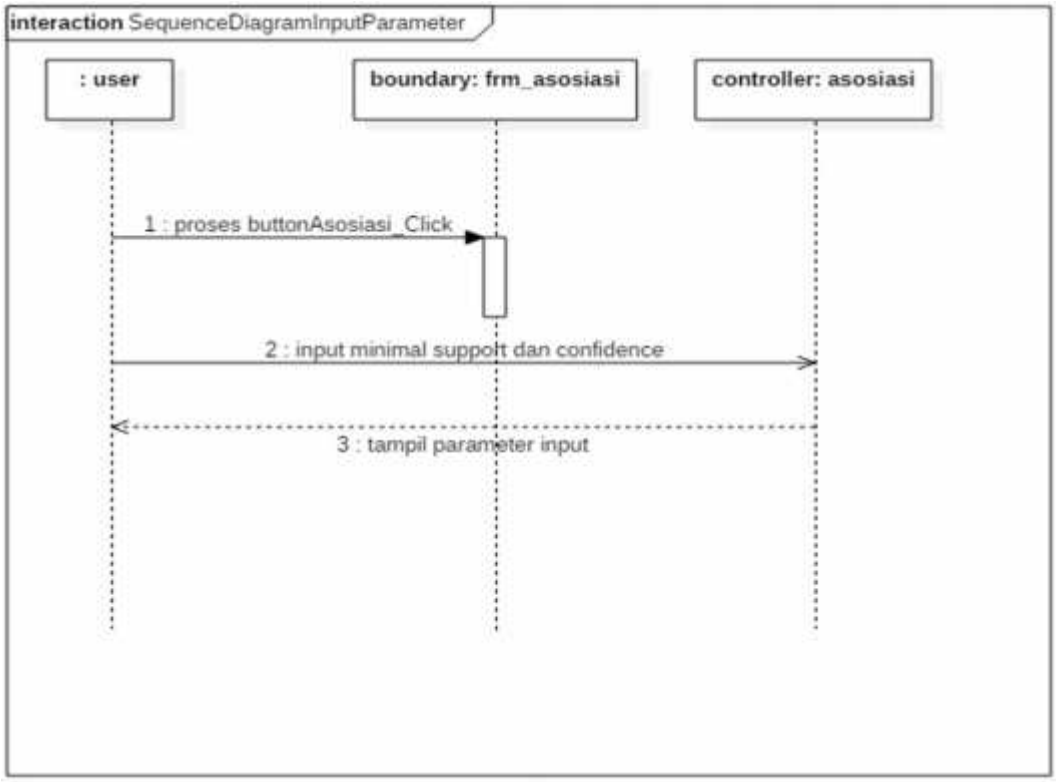
Gambar 3. 37 *Sequence* Diagram Pengurutan Data

3. *Sequence* Diagram Asosiasi



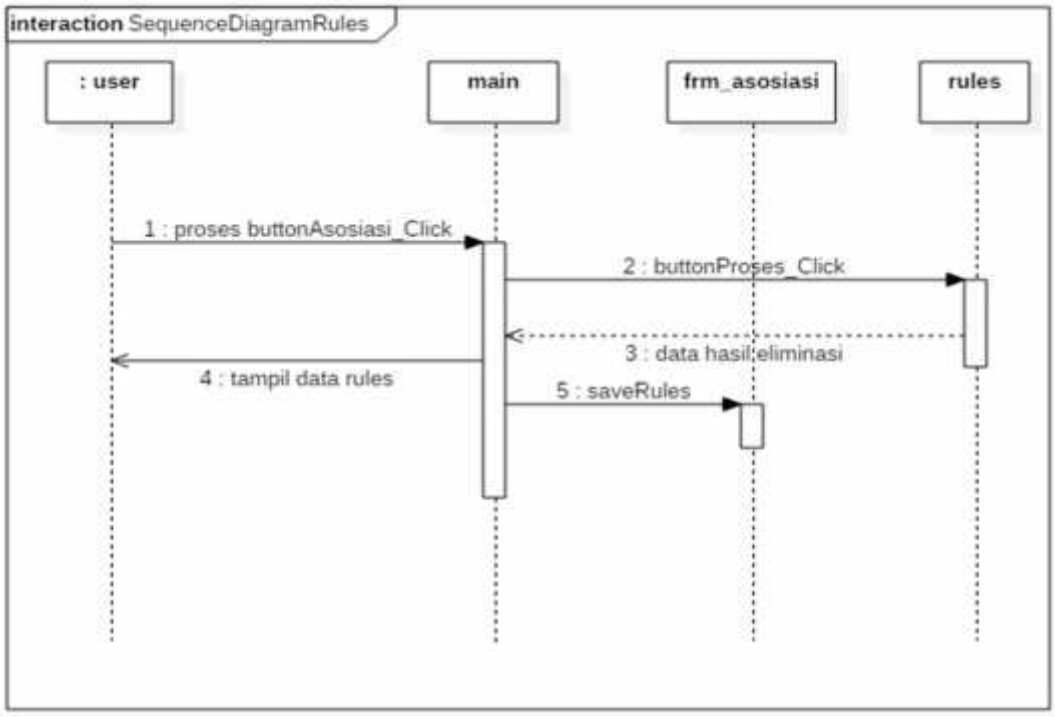
Gambar 3. 38 *Sequence* Diagram Asosiasi

4. *Sequence* Diagram Input Parameter



Gambar 3. 39 *Sequence* Diagram Input Parameter

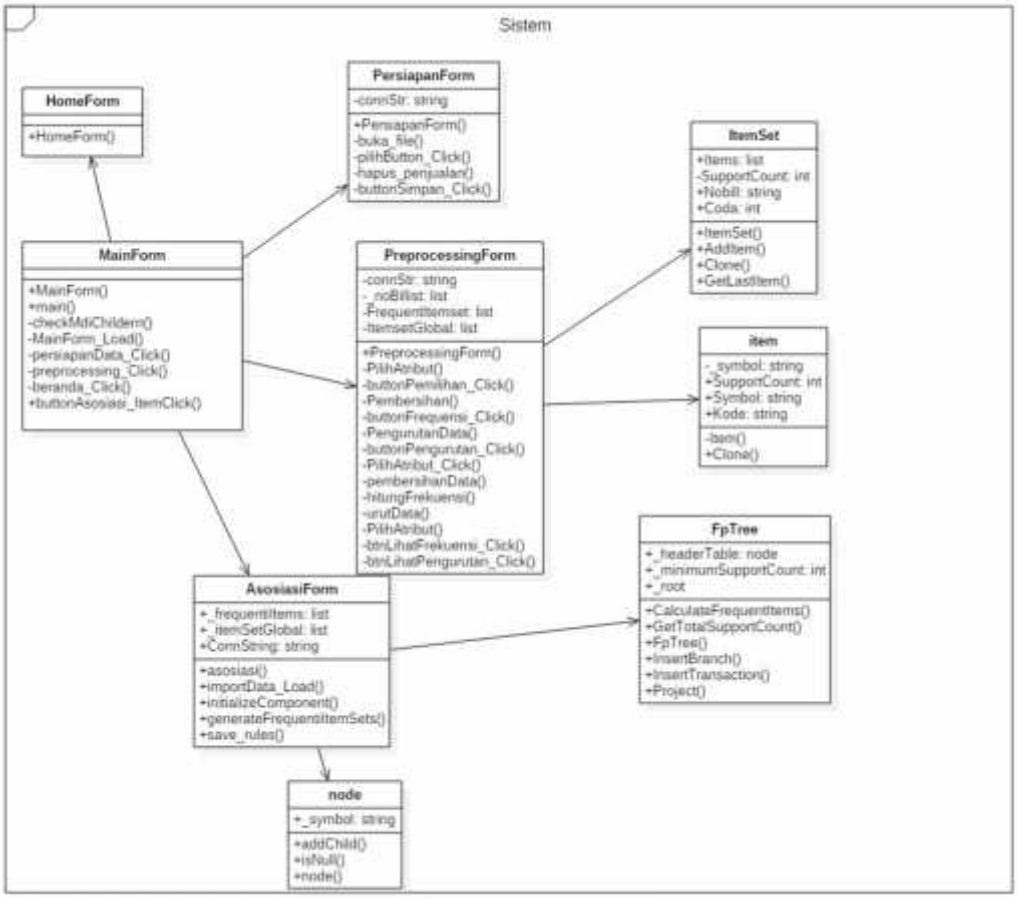
5. *Sequence* Diagram *Rules*



Gambar 3. 40 *Sequence* Diagram *Rules*

3.1.6 *Class* Diagram

Berikut adalah *class diagram* dari sistem yang akan dibangun. Hasilnya dapat pada gambar 3.41



Gambar 3. 41 *Class Diagram*

3.1.7 Analisis Kebutuhan Non Fungsional

Analisis kebutuhan non fungsional pada penelitian ini terdiri dari analisis spesifikasi kebutuhan perangkat keras / *hardware*, analisis kebutuhan perangkat lunak / *software*, dan analisis kebutuhan pengguna / *user*.

3.1.7.1 Analisis Kebutuhan Perangkat Keras

Analisis spesifikasi kebutuhan perangkat keras bertujuan untuk mengetahui spesifikasi perangkat keras yang digunakan. Berdasarkan pengamatan yang dilakukan di PT. PATISA CHAKRA MANDIRI, spesifikasi perangkat keras yang digunakan oleh PT. PATISA CHAKRA MANDIRI adalah sebagai berikut:

Tabel 3. 29 Spesifikasi Perangkat Keras yang Sedang Berjalan

No	Perangkat Keras	Spesifikasi
1	RAM	1 GB
2	Harddisk	160 GB
3	Monitor	LCD 14''
4	Processor	Intel Core 2 Duo E7500

Adapun spesifikasi perangkat keras yang dibutuhkan untuk mengimplementasikan sistem yang akan dibangun adalah sebagai berikut:

Tabel 3. 30 Spesifikasi Perangkat Keras yang dibutuhkan

No	Perangkat Keras	Spesifikasi
1	RAM	512MB
2	Harddisk	160 GB
3	Monitor	LCD 14''
4	Processor	Intel Dual Core 2.93 GHz

Melihat spesifikasi perangkat keras yang digunakan di PT. PATISA CHAKRA MANDIRI seperti pada tabel 3.27 dan tabel spesifikasi perangkat keras yang dibutuhkan seperti pada tabel 3.28 maka dapat disimpulkan bahwa semua perangkat keras yang dibutuhkan telah memadai untuk menjalankan sistem.

3.1.7.2 Analisis Kebutuhan Perangkat Lunak

Analisis kebutuhan perangkat lunak dalam pembangunan aplikasi digunakan untuk mengetahui dan mengoptimalkan implementasi dari aplikasi yang akan dibangun.

1. Analisis spesifikasi perangkat lunak pada sistem yang sedang berjalan di PT. PATISA CHAKRA MANDIRI adalah:
   a. Sistem Operasi Windows 7
   b. Microsoft Office 2013
2. Analisis spesifikasi kebutuhan perangkat lunak untuk menjalankan system yang akan dibangun adalah:
   a. Sistem Operasi Windows 7
   b. MySQL sebagai database server
   c. Microsoft Visual Studio 2010
3. Evaluasi kebutuhan perangkat lunak
   Berdasarkan analisis spesifikasi perangkat lunak yang ada di PT. PATISA CHAKRA MANDIRI, spesifikasi yang digunakan masih kurang. Untuk mengatasi hal tersebut akan di install MySQL dan juga Visual Studio pada computer yang digunakan.

3.1.7.3 Analisis Kebutuhan Pengguna

User atau pengguna yang akan mengoperasikan aplikasi ini merupakan manager yang bertindak sebagai pengolah bisnis. Analisis Pengguna dapat dilihat pada tabel 3.28

Tabel 3. 31 Analisis pengguna

User	Akses	Keahlian
Manager	Manager dapat mengolah data transaksi penjualan dan memperoleh informasi dari aplikasi *mining*.	Dapat mengoperasikan aplikasi dengan baik seperti penginputan data transaksi, penghapusan data transaksi, dan juga dapat mengolah informasi yang didapat.

3.2 Perancangan Sistem

Perancangan sistem berkenaan dengan pembangunan perangkat lunak setelah melaui tahap analisis sistem yang akan dibangun. Beberapa aspek dari perancangan sistem dalam penelitian ini meliputi perancangan *class*, perancangan struktur menu, perancangan antarmuka, perancangan pesan dan jaringan semantik.

3.2.1 Perancangan *Class*

Berikut ini adalah perancangan class yang digunakan pada sistem yang akan dibangun

Tabel 3.29 Struktur *class HomeForm*

Nama *class* : HomeForm		
Nama	**Jenis**	***Type***
HomeForm()	Method	Public

Tabel 3.30Struktur *class MainForm*

Nama *class* : MainForm		
Nama	**Jenis**	***Type***
MainForm()	Method	Public
Main()	Method	Public
buttonAsosiasi_ItemClick()	Method	Public
checkMdiChildern()	Method	Private
MainForm_Load()	Method	Private
persiapanData_Click()	Method	Private
preprocessing_Click()	Method	Private
beranda_Click()	Method	Private

Tabel 3.31 Struktur *class* PersiapanForm

Nama *class* : PersiapanForm		
Nama	**Jenis**	***Type***
connStr: string	Atribut	Private
PersiapanForm()	Method	Public
buka_file()	Method	Private

Nama *class* : PersiapanForm		
Nama	**Jenis**	***Type***
pilihButton_Click()	Method	Private
hapus_penjualan()	Method	Private
buttonSimpan_Click()	Method	Private

Tabel 3.32 Struktur *class PreprocessingForm*

Nama *class* : PreprocessingForm		
Nama	**Jenis**	***Type***
connStr: string	Atribut	Private
_noBillist: list	Atribut	Private
FrequentItemset: list	Atribut	Private
ItemsetGlobal: list	Atribut	Private
PreprocessingForm()	Method	Public
PilihAtribut_Click()	Method	Private
buttonPemilihan_Click()	Method	Private
Pembersihan()	Method	Private
buttonFrequensi_Click()	Method	Private
PengurutanData()	Mehod	Private
buttonPengurutan_Click()	Method	Private
PilihAtribut()	Method	Private
pembersihanData()	Method	Private
hitungFrekuensi()	Method	Private
urutData()	Method	Private
PilihAtribut()	Mehod	Private
btnLihatFrekuensi_Click()	Method	Private
btnLihatPengurutan_Click()	Method	Private

Tabel 3.33 Struktur *class ItemSet*

Nama *class* : ItemSet		
Nama	**Jenis**	***Type***
Items: list	Atribut	Public
Nobill: string	Atribut	Public
Coda: int	Atribut	Public
SupportCount: int	Atribut	Private
ItemSet()	Method	Public
AddItem()	Method	Public
Clone()	Method	Public
GetLastItem()	Method	Public

Tabel 3.34 Struktur *class item*

Nama *class* : item		
Nama	**Jenis**	***Type***
SupportCount: int	Atribut	Public
Symbol: string	Atribut	Public
Kode: string	Atribut	Public
_symbol: string	Atribut	Private
clone()	Method	Public
item()	Method	Private

Tabel 3.35 Struktur *class* AsosiasiForm

Nama *class* : AsosiasiForm		
Nama	**Jenis**	***Type***
_frequentItems: list	Atribut	Public
_itemSetGlobal: list	Atribut	Public
ConnString: string	Atribut	Public
asosiasi()	Method	Public
importData_Load()	Method	Public
initializeComponent()	Method	Public
generateFrequentItemSet()	Method	Public
save_rules()	Method	Public

Tabel 3.36 Struktur *class FpTree*

Nama *class* : FpTree		
Nama	**Jenis**	***Type***
headerTable: node	Atribut	Public
_minimumSupportCount: int	Atribut	Public
_root	Atribut	Public
CalculateFrequentItems()	Method	Public
GetTotalSupportCount()	Method	Public
FpTree()	Method	Public
InsertBranch()	Method	Public
InsertTransaction()	Method	Public
Project()	Method	Public

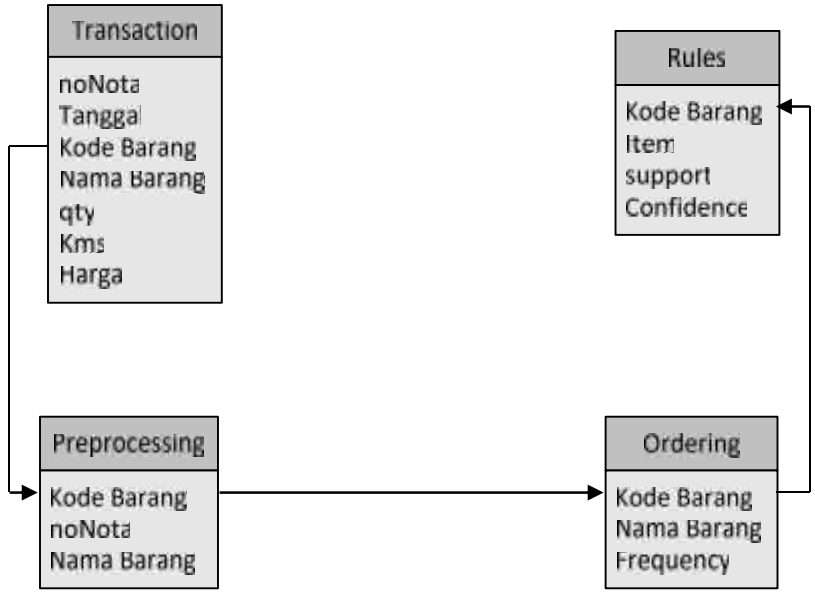
Tabel 3.37 Struktur *class node*

Nama *class* : node		
Nama	**Jenis**	***Type***
_symbol: string	Atribut	Public
addChild()	Method	Public

Nama *class* : node		
Nama	**Jenis**	***Type***
isNull()	Method	Public
node()	Method	Public

3.2.2 Skema Relasi

Skema relasi merupakan rangkaian hubungan antara dua tabel atau lebih pada sistem basis data. Berikut ini adalah skema relasi pada sistem pembentukan paket barang di PT. PATISA CHAKRA MANDIRI:



Gambar 3. 42 Skema relasi

3.2.3 Struktur Tabel

Berikut ini adalah struktur dari tabel-tabel yang digunakan pada sistem yang akan dibangun:

1. Tabel *Transaction*

Tabel *Transaction* adalah tabel yang digunakan untuk menyimpan data transaksi.

Tabel 3. 41 Tabel *transactions*

Nama Field	Tipe Data	Panjang	Keterangan
No	Int	10	
noNota	Varchar	50	Primary Key
Tanggal	Int	10	
Kode_barang	Varchar	10	
Nama_barang	Varchar	255	
Qty	Int	10	
Kms	Varchar	10	
Harga	Int	10	

2. Tabel *Preprocessing*

Tabel *Preprocessing* digunakan untuk menyimpan data hasil pemilihan atribut dan pembersihan data.

Tabel 3. 42 Tabel *preprocessing*

Nama Field	Tipe Data	Panjang	Keterangan
kode_barang	Int	10	FOREIGN KEY (kode_barang) REFERENCES `transaction`(kode_barang)
noNota	Varchar	50	
nama_barang	Varchar	255	

3. Tabel *Ordering*

Tabel *Ordering* digunakan untuk menyimpan data barang yang dibeli.

Tabel 3. 43 Tabel *Ordering*

Nama Field	Tipe Data	Panjang	Keterangan
kode_barang	int	10	FOREIGN KEY (kode_barang) REFERENCES `transaction`(kode_barang)
nama_barang	varchar	255	
frequency	int	10	

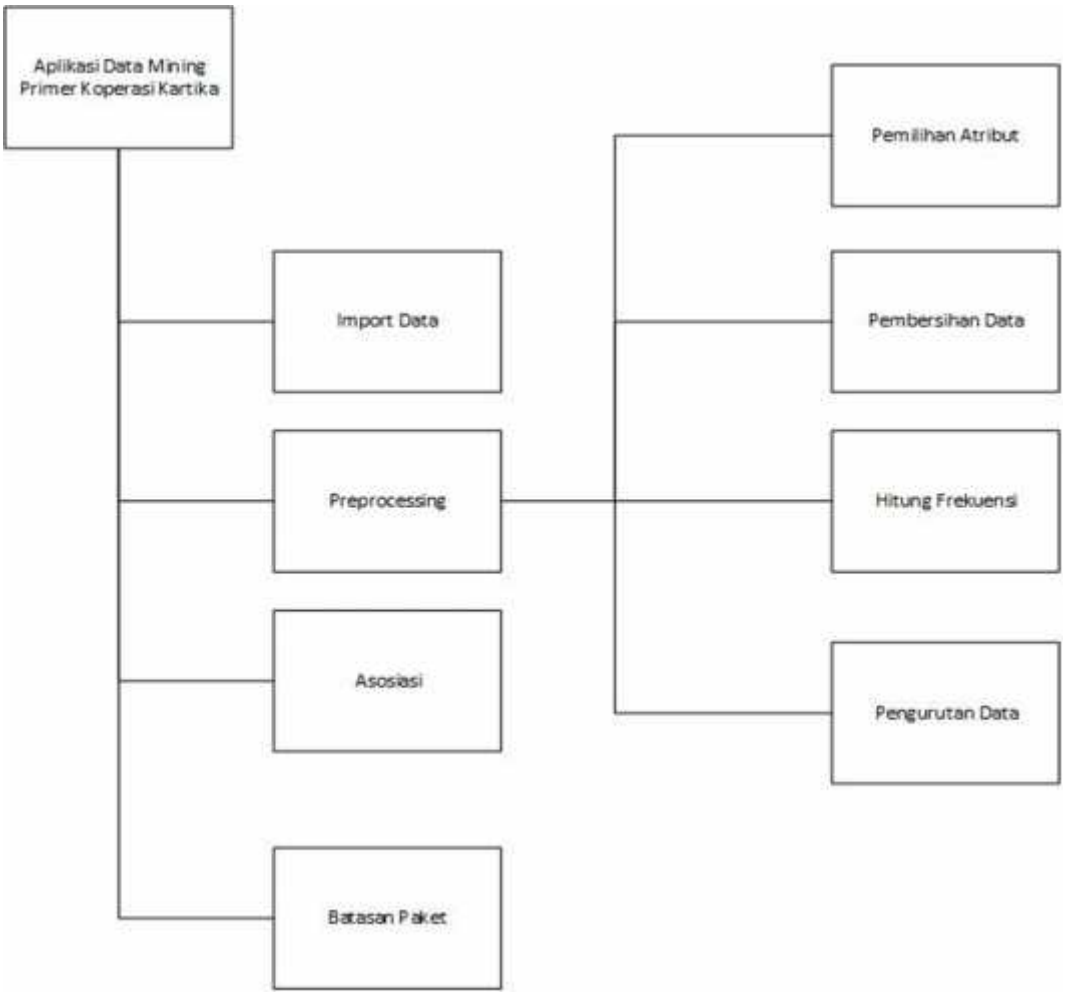
4. Tabel *Rules*

Tabel *Rules* digunakan untuk menyimpan hasil rule

Tabel 3. 44 Tabel *Rules*

Nama Field	Tipe Data	Panjang	Keterangan
Kode_barang	Int	10	FOREIGN KEY (kode_barang) REFERENCES `transaction`(kode_barang)
Item	Varchar	255	
Sup	Double		
Conf	Double		

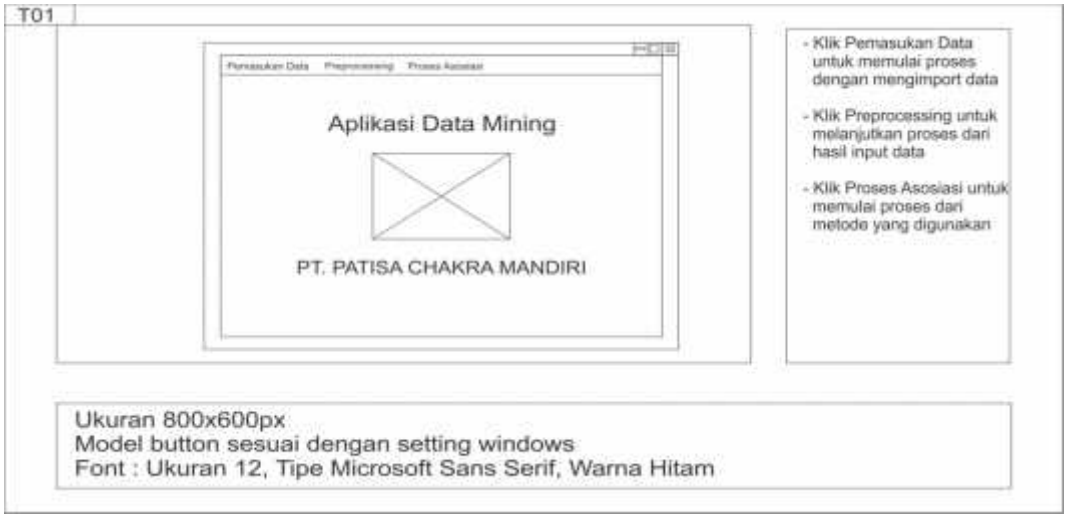
3.2.4 Perancangan Struktur Menu



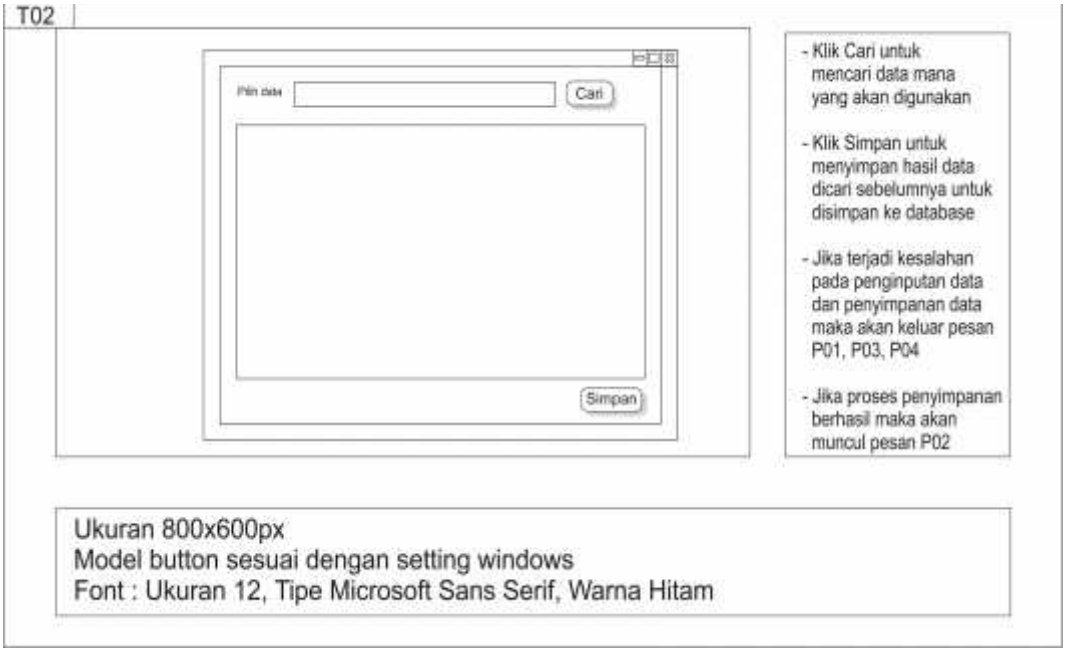
Gambar 3. 43 Struktur menu

3.2.5 Perancangan Antar Muka

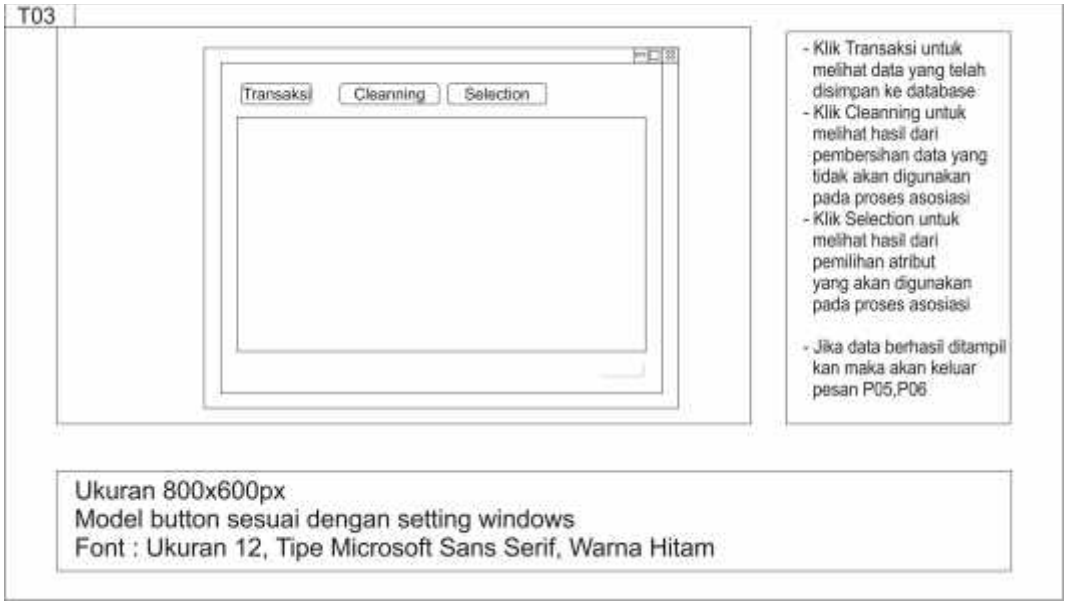
Perancangan antar muka berfungsi untuk memberi gambaran tentang perangkat lunak yang akan dibangun, supaya lebih memudahkan dalam pembuatan perangkat lunak nantinya. Adapun perancangan antar muka untuk perangkat lunak yang akan dibangun akan dijelaskan berikut ini:



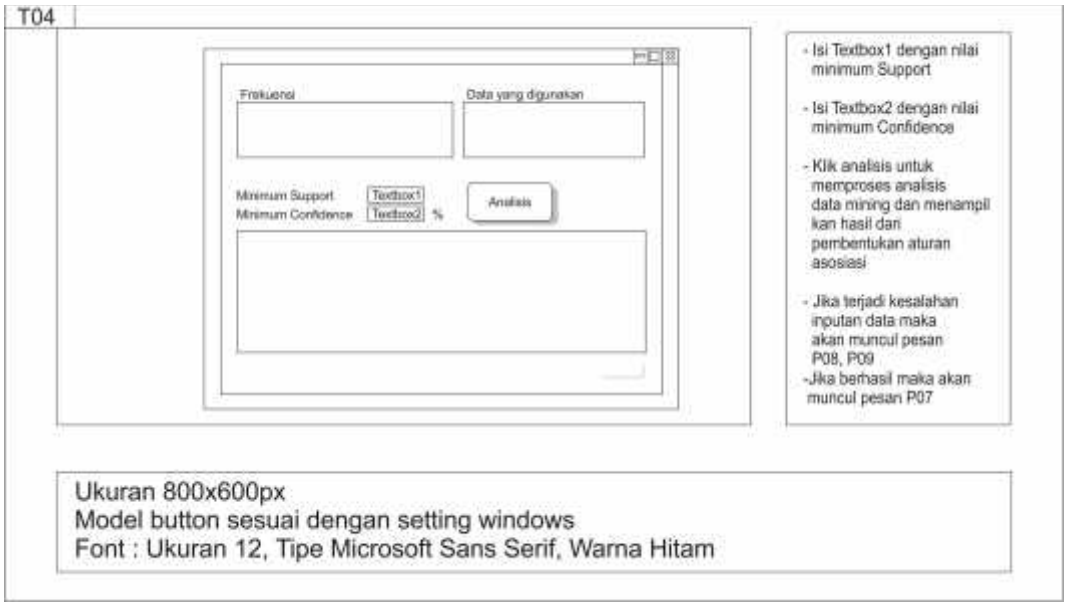
Gambar 3. 44 Perancangan antar muka menu utama



Gambar 3. 45 Perancangan antarmuka *data import*



Gambar 3. 46 Perancangan antarmuka *preprocessing*

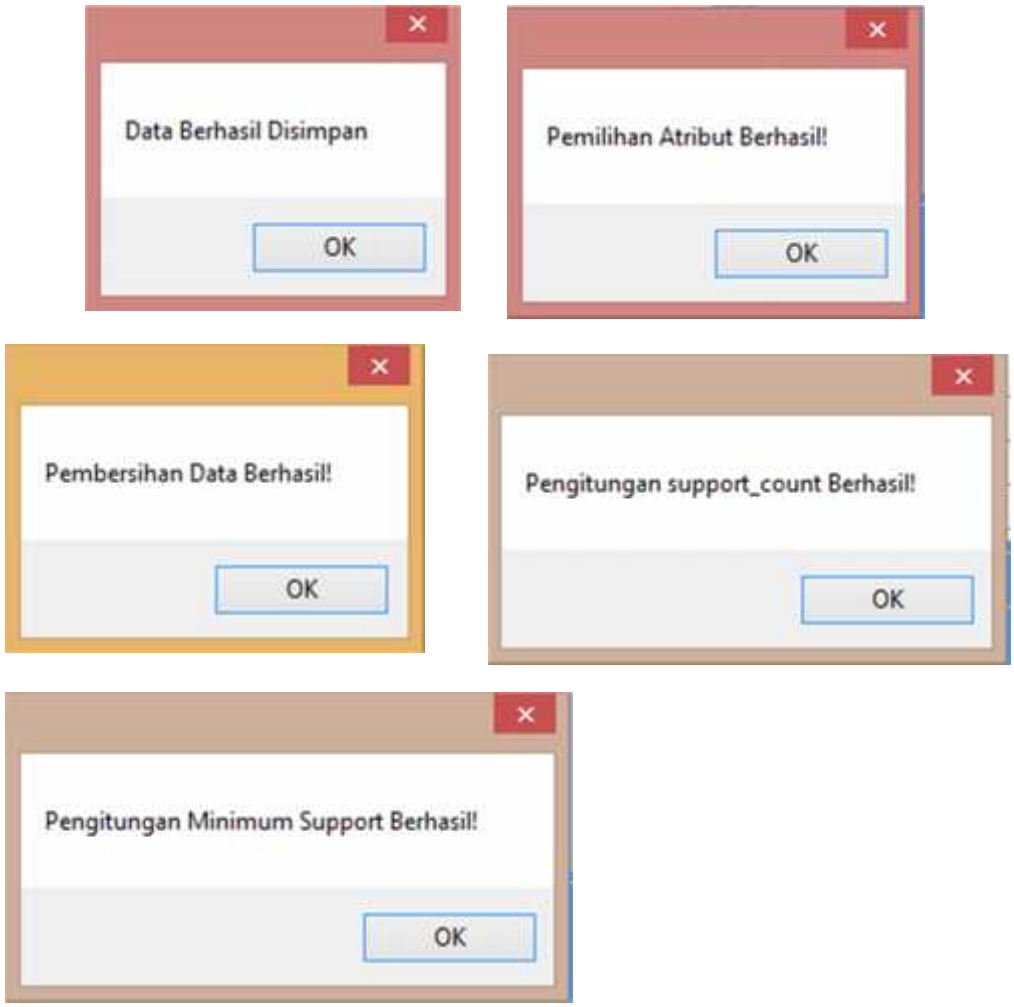


Gambar 3. 47 Perancangan antar muka asosiasi

3.2.6 Perancangan Pesan

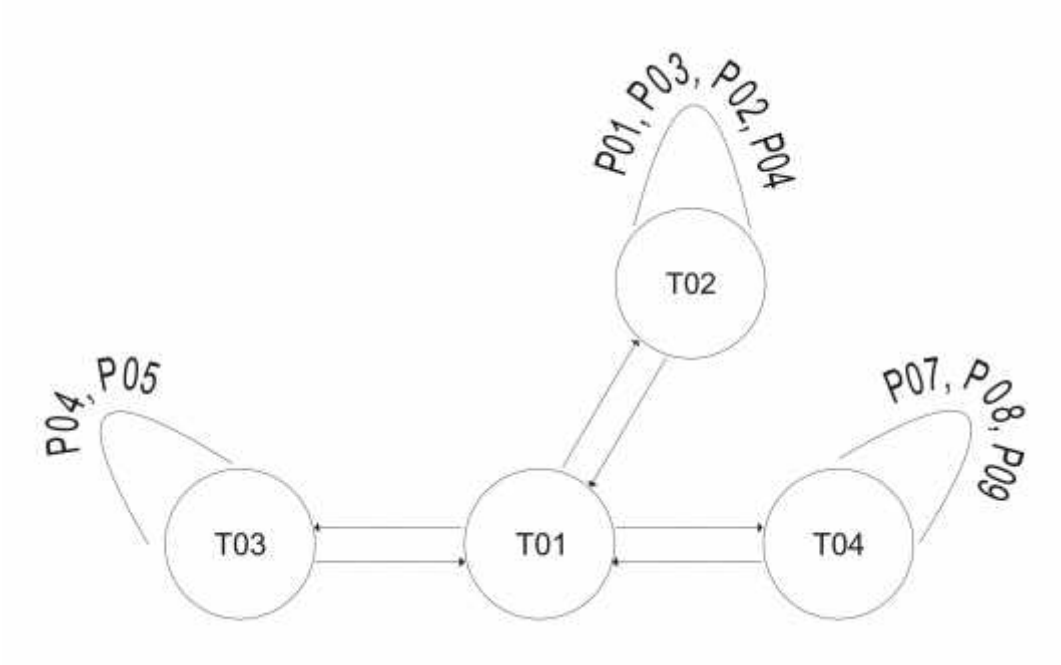
Adapun perancangan pesan dapat dilihat pada tabel:

Gambar 3. 48 Tabel Pesan



3.2.7 Jaringan Semantik

Jaringan semantik bertujuan untuk merepresentasikan pemetaan data untuk mencegah duplikasi data. Adapun jaringan semantik yang menghubungkan antarmuka pada aplikasi yang akan dibangun dapat dilihat pada gambar berikut ini:



Gambar 3. 49 Perancangan Jaringan Semantik