BAB 3
ANALISIS DAN PERANCANGAN

3.1 Analisis Sistem

Analisis sistem (*System Analysis*) dapat didefinisikan sebagai penguraian dari suatu sistem informasi yang utuh ke dalam bagian-bagian komponennya dengan maksud untuk mengidentifikasikan dan mengevaluasi permasalahan-permasalahan, kesempatan-kesempatan, hambatan-hambatan yang terjadi dan kebutuhan-kebutuhan yang diharapkan sehingga dapat diusulkan perbaikan-perbaikannya. Dalam analisa sistem ini meluputi beberapa bagian, yaitu :

1. Analisis Masalah
2. Analisis Prosedur Penentuan Paket
3. Analisis Crisp-DM
4. Analisis Non Fungsional dan Kebutuhan Non Fungsional
5. Analisis Kebutuhan Fungsional

3.1.1 Analisis Masalah

Berdasarkan hasil pengamatan, dapat disimpulkan bahwa permasalahan yang ada di toko Kartika Kosmetik yaitu belum adanya informasi penentuan paket produk kosmetik yang sesuai dengan minat konsumen yang akan ditawarkan menjelang hari-hari besar tertentu, seperti hari raya Idul Fitri, hari raya Idul Adha, hari raya kemerdekaan, hari raya Natal, tahun baru dan hari besar lainnya.

3.1.2 Prosedur Penentuan Paket Produk Kosmetik

Berdasarkan observasi ke toko Kartika Kosmetik bahwa prosedur atau alur pembentukan paket produk kosmetik adalah sebagai berikut:

1. Bagian keuangan membuat laporan penjualan selama 1 bulan periode sebelum hari raya, periode tersebut diambil karena pada saat itu biasanya konsumen berbelanja untuk kebutuhan hari raya.
2. Bagian keuangan memberikan laporan penjualan tersebut pada bagian gudang untuk pengecekan stok barang.

3. Bagian gudang mengecek stok barang dan membuat laporan barang dari yang paling laku sampai yang tidak laku dan dari stok barang yang paling banyak hingga stok yang paling sedikit. Laporan stok tersebut diberikan pada pihak manager.
4. Manager kemudian membuat laporan paket produk secara manual dan acak, didasarkan dari laporan data stok. Produk yang paling laris akan dipasangkan dengan produk yang kurang laris dan produk yang stoknya masih banyak.
5. Laporan paket produk tersebut kemudian diberikan pada pemimpin untuk mendapatkan persetujuan.
6. Jika pemimpin tidak memberikan persetujuan, data dikembalikan pada pihak manager untuk diubah. Jika pemimpin memberikan persetujuan, data paket produk diberikan pada bagian gudang untuk dikemas.
7. Bagian gudang mengemas produk untuk dijadikan paket, jika sudah selesai paket tersebut akan diberikan pada karyawan untuk ditawarkan ke konsumen.
8. Bagian karyawan menawarkan produk pada konsumen selama 2 minggu sebelum lebaran sampai 2 minggu setelah lebaran.

Dari alur pemaketan produk diatas memiliki kelebihan dan kekuraangan. Salahsatu contoh dimisalkan dalam satu paket terdiri dari 3 produk yang dipasangkan. 3 produk tersebut terdiri dari produk yang terlaris, sedang dan tidak laris. Untuk kelebihan dari pembuatan paket tersebut yaitu produk yang kurang laris bisa saja ikut terjual karena dipasangkan pada produk yang laris. Namun untuk kekurangannya adalah belum tentu paket tersebut dapat terjual karena isi dari paket produk tersebut kurang tepat bila dilihat dari minat konsumen dalam membeli produk yang berbeda-beda. Bahkan bisa jadi produk yang laris yang sudah dipaketkan tersebut bisa malah menjadi tidak laris, karena beberapa konsumen tidak membutuhkan pasangan produk dari produk yang laris tersebut. Hal itu bisa membuat pihak kartika malah menjadi merug. Maka dari itu dibutuhkan data mining untuk merekomendasikan paket produk yang sesuai dengan minat konsumen.

3.1.3 Analisis Crisp-DM

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

3.1.3.1 Business Understanding

Pemahaman Bisnis atau disebut dengan *Business Understanding* merupakan tahapan pertama yang dilakukan dalam kerangka kerja CRISP-DM. Dalam tahapan bisnis ini terbagi menjadi dua bagian, yaitu:

a. Identifikasi Tujuan Bisnis

Tujuan Bisnis dari toko Kartika Kosmetik yaitu memasarkan produk secara langsung untuk memenuhi permintaan konsumen.

b. Penentuan Sasaran Data *Mining*

Tujuan dari penerapan *Data Mining* ini adalah untuk mengetahui pasangan produk yang sering dibeli oleh konsumen atau pola pembelian konsumen yang sering terjadi yang akan dijadikan dasar oleh pihak Kartika Kosmetik dalam penentuan pembuatan paket produknya.

3.1.3.2 Data Understanding

Sumber data yang didapat dalam penelitian ini merupakan data transaksi penjualan yang terjadi di toko Kartika Kosmetik periode 1 bulan sebelum menjelang hari raya Idul Fitri (contoh kasus diambil untuk hari raya Idul Fitri) yaitu periode bulan Juni 2015. Adapun detail informasi mengenai data transaksi penjualan yang digunakan dapat dilihat pada tabel 3.1 di bawah ini.

Tabel 3.1 Struktur Data Transaksi Penjualan

Dokumen	Keterangan		
Detail Data Transaksi	Deskripsi	Data ini berisi mengenai data transaksi yang ada di toko Kartika Kosmetik	
	Format	Microsoft Excel (.xlsx)	
	Atribut	No	Nomor urut

		No Faktur	Nomor struk
		Tanggal	Tanggal pencetakan struk
		Kode Barang	Kode barang yang dibeli
		Nama Barang	Nama barang yang dibeli
		Varian	Jenis dari barang yang dibeli
		Harga Barang	Harga barang yang dibeli
		Qty	Jumlah barang yang dibeli
		Jumlah	Harga Barang dikalikan dengan Qty
		Total Harga	Total harga dari barang yang dibeli

3.1.3.3 Data Preparation

Persiapan Data merupakan tahap dimana akan dilakukan pemilihan tabel dan *field* yang akan digunakan dalam proses *mining.* Persiapan data dilakukan dengan sebutan *Preprocessing Data. Preprocessing* merupakan hal yang harus dilakukan dalam proses *data mining*, karena tidak semua data atau atribut data dalam data digunakan dalam proses data *mining*. Proses ini dilakukan agar data yang digunakan sesuai dengan kebutuhan. Adapun tahapan-tahapan *preprocessing* data dalam penelitian ini adalah sebagai berikut:

1. Ekstrasi Data

   Ekstraksi data dapat diartikan sebagai proses pengambilan data dari sumber data dalam rangka untuk melanjutkan proses pengolahan data ke tingkat selanjutnya ataupun untuk menyimpan data hasil ekstrak tersebut. Dalam penelitian ini, data yang berasal dari *flat file* berformat *microsoft excel* (.xlsx) di ekstrak kedalam format *sql*, kemudian disimpan kedalam sebuah *database* agar memudahkan dalam proses pengolahan data. Berikut merupakan contoh sampel data transaksi 1 bulan periode bulan Juni sebanyak 44 transaksi yang terdapat di toko Kartika Kosmetik, dimana atribut-atribut yang terdapat dalam data tersebut antara lain No, No Faktur, Tanggal, Kode Barang, Nama Barang, Varian, Harga Barang, Qty, Jumlah, dan Total Harga. Data transaksi dapat dilihat pada tabel D-1 pada Lampiran D.

2. Pemilihan Atribut

   Proses pemilihan atribut atau *selection data* adalah proses dimana atribut data akan dipilih dan diproses sesuai dengan kebutuhan *data mining*. Sebelum melakukan proses *cleaning* atau pembersihan data akan lebih efisien jika melakukan proses *selection* atau pemilihan atribut ini terlebih dahulu. Karena dari data transaksi Kartika Kosmetik yang sebanyak ribuan record ini memiliki 10 atribut sedangkan yang dibutuhkan untuk *data mining* hanya 2 atribut. Dengan menyeleksi atribut yang tidak dibutuhkan akan memudahkan pada proses pembersihan data nanti, sehingga pada saat pembersihan data tidak akan memperberat memori karena program harus membaca atribut yang tidak perlu. Dalam penelitian ini, 2 atribut yang akan digunakan adalah atribut No Faktur dan Nama Barang. Kedua atribut ini digunakan untuk memenuhi tujuan awal dimana akan dicari pola pembelian konsumen berdasarkan produk yang dibeli. Seperti atribut No Faktur digunakan untuk membedakan satu transaksi dengan transaksi lainnya, dan atribut Nama Barang digunakan untuk mengetahui barang apa saja yang dibeli dalam satu transaksi. No Faktur dengan tipe data varchar dirubah ke integer dengan di-*trim* sehingga menjadi 6 angka dari belakang. Hasil pemilihan atribut dapat dilihat pada tabel D-2 pada lampiran D.

3. Pembersihan Data

   Proses pembersihan data atau *cleaning data* adalah proses menghilangkan data tidak relevan atau inkosisten dan proses menghilangkan *noise*. *Noise* disini yaitu data transaksi yang hanya memiliki 1 produk dalam sekali pembelian. Dalam data transaksi ini akan dilakukan pengeleminasian terhadap transaksi yang memiliki jumlah produk kurang dari 2 produk dalam satu kali transaksinya, karena syarat ini diperlukan dalam *Association Rules* untuk melihat keterhubungan antar 2 produk atau lebih. Setelah melakukan proses pembersihan data, dari data transaksi yang awalnya sebanyak 44 transaksi di *cleaning*

menjadi 30 transaksi. Hasil pembersihan data dapat dilihat pada tabel D-3 pada lampiran D.

4. Penyiapan Data Awal

   Setelah semua proses pembersihan data berhasil dilakukan dan data transaksi telah sesuai dengan kebutuhan yang diperlukan dalam proses *data mining*, maka data transaksi sudah dapat digunakan untuk proses selanjutnya dalam sistem *data mining*. Data pada tabel D-3 pada lampiran D adalah data yang akan digunakan untuk proses mining.

3.1.3.4 Modeling

Penelitian ini bertujuan untuk mendapatkan informasi mengenai pola pembelian konsumen yang nantinya akan digunakan oleh pihak Kartika Kosmetik sebagai dasar pengambilan keputusan untuk menentukan produk apa saja yang bisa dijadikan sebuah paket penjualan produk yang nantinya akan ditawarkan ke konsumen.

Dengan menerapkan *Data Mining*, data transaksi akan diolah dengan aturan asosiasi atau metode *Association Rules* untuk menemukan pola-pola pembelian produk yang sering dibeli oleh konsumen. Metodologi dasar *Association Rule* terbagi menjadi dua tahap, yaitu Analisa pola frekuensi tinggi dan Pembentukan Aturan Assosiatif. Algoritma yang digunakan dalam penelitian ini adalah algoritma *CT-Pro.* Langkah-langkah algoritma *CT-PRO* adalah sebagai berikut:

1. Menghitung Frekuensi Kemunculan Tiap *Item*

Dengan menggunakan tabel D-3, masing-masing *item* dihitung frekuensi kemunculan berdasarkan *item*nya. Hasil dari penghitungan kemunculan *item* dapat dilihat pada tabel 3.2 dibawah ini.

Tabel 3.2 Hasi Perhitungan Kemunculan Produk

Nama Barang	*Support Count*
Wardah Lig Day Cream Step 1	13
Ponds Wb Ff	12

Nama Barang	*Support Count*
Minyak Kayu Putih	8
Wardah Facial Scrub	8

Nama Barang	*Support Count*
Wardah Lig Day Cream Step 2	12
Ponds Wb Lightenig Cream	11
Inez Ppc	11
Kapas Sariayu	11
Casablanca Spray	10
Rexona Wmn Roll On Power Dry	10
Elips Hair Mask	10
Ponds Fw Day Cream	9
Natur E Hbl	9
Eye Bulu	9
Inez Lip	8

Nama Barang	*Support Count*
Mascara Candy	7
Elips Hair Vit	6
Rexona Women Whitening	6
Vaseline Hlt Wht Uv Light	4
Pixy Ref. Twc	4
Garnier Light Comp Crm	3
Clean N Clear F Wash	2
Vitalis	2
Citra Hbl Lst Wht Uv	1
Ms. Ratu Masker Bengk	1

2. Membangun *Global Item Tabel*

Berikut merupakan langkah-langkah untuk membuat *global item tabel* :

a. Langkah pertama adalah menentukan *minimum support* untuk melihat batasan terendah munculnya *item*. Untuk *minimum support* yang diambil pada penelitian ini adalah sebesar 7, maka batasan produk yang muncul harus sebanyak >= 7 kali.

b. Pada tabel 3.2, masing-masing *item* diseleksi berdasarkan *minimum support count* yang telah ditentukan sehingga terbentuk *global item*. Untuk setiap *item* diberikan *id global item* yaitu penomoran secara *ascending* dari frekuensi terbesar ke terkecil. Berikut adalah tabel *global item* yang dapat dilihat pada tabel 3.3.

Tabel 3.3 *Global Item Tabel*

Nama Barang	*Support Count*	*Id Global Item*
Wardah Lig Day Cream Step 1	13	1
Ponds Wb Ff	12	2
Wardah Lig Day Cream Step 2	12	3
Ponds Wb Lightenig Cream	11	4
Inez Ppc	11	5
Kapas Sariayu	11	6
Casablanca Spray	10	7
Rexona Wmn Roll On Power Dry	10	8
Elips Hair Mask	10	9
Ponds Fw Day Cream	9	10
Natur E Hbl	9	11

Nama Barang	*Support Count*	*Id Global Item*
Eye Bulu	9	12
Inez Lip	8	13
Minyak Kayu Putih	8	14
Wardah Facial Scrub	8	15
Mascara Candy	7	16

Pada tabel D-3 untuk *Item* yang *frequent*-nya tidak memenuhi batas *minimum support* dieleminasi dan hasilnya dapat dilihat pada tabel 3.4.

Tabel 3.4 *Item* yang *Frequent*

Nomor Faktur	Nama Barang
600001	Casablanca Spray
600001	Eye Bulu
600001	Mascara Candy
600001	Ponds Fw Day Cream
600001	Ponds Wb Ff
600001	Ponds Wb Lightenig Cream
600001	Wardah Facial Scrub
600001	Wardah Lig Day Cream Step 1
600001	Wardah Lig Day Cream Step 2
600002	Inez Ppc
600002	Kapas Sariayu
600002	Natur E Hbl
600002	Rexona Wmn Roll On Power Dry
600002	Wardah Facial Scrub
600002	Wardah Lig Day Cream Step 1
600002	Wardah Lig Day Cream Step 2
600003	Elips Hair Mask
600003	Inez Lip
600003	Kapas Sariayu
600003	Rexona Wmn Roll On Power Dry
600003	Wardah Facial Scrub
600003	Wardah Lig Day Cream Step 1
600003	Wardah Lig Day Cream Step 2
600004	Elips Hair Mask
600004	Inez Lip
600004	Inez Ppc
600004	Minyak Kayu Putih
600004	Rexona Wmn Roll On Power Dry
600004	Wardah Lig Day Cream Step 1
600004	Wardah Lig Day Cream Step 2
600005	Casablanca Spray
600005	Kapas Sariayu
600005	Mascara Candy
600005	Wardah Facial Scrub

Nomor Faktur	Nama Barang
600021	Eye Bulu
600021	Inez Lip
600021	Mascara Candy
600021	Minyak Kayu Putih
600021	Ponds Wb Ff
600021	Ponds Wb Lightenig Cream
600024	Elips Hair Mask
600025	Elips Hair Mask
600025	Natur E Hbl
600027	Casablanca Spray
600027	Kapas Sariayu
600027	Mascara Candy
600027	Minyak Kayu Putih
600027	Ponds Fw Day Cream
600028	Inez Ppc
600028	Kapas Sariayu
600028	Natur E Hbl
600028	Rexona Wmn Roll On Power Dry
600028	Wardah Facial Scrub
600028	Wardah Lig Day Cream Step 1
600028	Wardah Lig Day Cream Step 2
600029	Inez Lip
600029	Ponds Fw Day Cream
600029	Ponds Wb Ff
600029	Wardah Lig Day Cream Step 1
600029	Wardah Lig Day Cream Step 2
600031	Eye Bulu
600031	Kapas Sariayu
600031	Minyak Kayu Putih
600031	Natur E Hbl
600031	Ponds Wb Ff
600031	Ponds Wb Lightenig Cream
600033	Casablanca Spray
600033	Inez Lip

Nomor Faktur	Nama Barang
600005	Wardah Lig Day Cream Step 1
600005	Wardah Lig Day Cream Step 2
600006	Elips Hair Mask
600006	Inez Ppc
600006	Kapas Sariayu
600006	Minyak Kayu Putih
600006	Natur E Hbl
600006	Rexona Wmn Roll On Power Dry
600007	Inez Ppc
600007	Kapas Sariayu
600007	Ponds Wb Ff
600007	Ponds Wb Lightenig Cream
600007	Wardah Lig Day Cream Step 1
600008	Wardah Lig Day Cream Step 1
600008	Wardah Lig Day Cream Step 2
600009	Ponds Wb Ff
600009	Ponds Wb Lightenig Cream
600011	Casablanca Spray
600011	Eye Bulu
600011	Inez Lip
600011	Kapas Sariayu
600011	Mascara Candy
600011	Ponds Wb Ff
600011	Ponds Wb Lightenig Cream
600012	Casablanca Spray
600012	Eye Bulu
600012	Natur E Hbl
600012	Ponds Fw Day Cream
600012	Ponds Wb Ff
600012	Ponds Wb Lightenig Cream
600012	Rexona Wmn Roll On Power Dry
600013	Elips Hair Mask
600013	Kapas Sariayu
600013	Wardah Lig Day Cream Step 1
600013	Wardah Lig Day Cream Step 2
600014	Wardah Facial Scrub
600014	Wardah Lig Day Cream Step 1
600014	Wardah Lig Day Cream Step 2
600016	Ponds Fw Day Cream
600016	Ponds Wb Ff
600016	Ponds Wb Lightenig Cream
600017	Inez Lip
600017	Inez Ppc
600021	Casablanca Spray
600021	Elips Hair Mask

Nomor Faktur	Nama Barang
600033	Inez Ppc
600033	Ponds Fw Day Cream
600033	Ponds Wb Lightenig Cream
600033	Rexona Wmn Roll On Power Dry
600035	Elips Hair Mask
600035	Eye Bulu
600035	Kapas Sariayu
600035	Mascara Candy
600035	Minyak Kayu Putih
600035	Ponds Fw Day Cream
600035	Rexona Wmn Roll On Power Dry
600036	Wardah Facial Scrub
600037	Casablanca Spray
600037	Elips Hair Mask
600037	Mascara Candy
600037	Minyak Kayu Putih
600037	Ponds Fw Day Cream
600037	Ponds Wb Ff
600037	Ponds Wb Lightenig Cream
600039	Elips Hair Mask
600039	Inez Ppc
600039	Natur E Hbl
600039	Wardah Lig Day Cream Step 1
600039	Wardah Lig Day Cream Step 2
600041	Casablanca Spray
600041	Eye Bulu
600041	Inez Ppc
600041	Ponds Wb Ff
600041	Ponds Wb Lightenig Cream
600041	Rexona Wmn Roll On Power Dry
600043	Casablanca Spray
600043	Eye Bulu
600043	Inez Ppc
600043	Natur E Hbl
600043	Ponds Fw Day Cream
600043	Ponds Wb Ff
600043	Rexona Wmn Roll On Power Dry
600044	Eye Bulu
600044	Inez Lip
600044	Inez Ppc
600044	Minyak Kayu Putih
600044	Natur E Hbl
600044	Wardah Facial Scrub
600044	Wardah Lig Day Cream Step 1
600044	Wardah Lig Day Cream Step 2

3. *Mapping* Data

Mapping yaitu memetakan data transaksi dari tabel 3.4 terhadap *id global item* pada tabel 3.3, dimana nama barang pada tabel 3.4 digantikan oleh *id global item* sesuai denganyang ada pada tabel 3.3. Data transaksi kemudian diurutkan dari *id global item* terkecil ke terbesar dari setiap nomor fakturnya. Berikut adalah hasil *mapping* data transaksi tabel 3.4 yang dapat dilihat pada tabel 3.5 di bawah ini.

Tabel 3.5 Tabel *Mapping*

Nomor Faktur	*Items*
600001	1
600001	2
600001	3
600001	4
600001	7
600001	10
600001	12
600001	15
600001	16
600002	1
600002	3
600002	5
600002	6
600002	8
600002	11
600002	15
600003	1
600003	3
600003	6
600003	8
600003	9
600003	13
600003	15
600004	1
600004	3
600004	5
600004	8
600004	9

Nomor Faktur	*Items*
600006	11
600006	14
600007	1
600007	2
600007	4
600007	5
600007	6
600008	1
600008	3
600009	2
600009	4
600011	2
600011	4
600011	6
600011	7
600011	12
600011	13
600011	16
600012	2
600012	4
600012	7
600012	8
600012	10
600012	11
600012	12
600013	1
600013	3
600013	6

Nomor Faktur	*Items*
600021	9
600021	12
600021	13
600021	14
600021	16
600024	9
600025	9
600025	11
600027	6
600027	7
600027	10
600027	14
600027	16
600028	1
600028	3
600028	5
600028	6
600028	8
600028	11
600028	15
600029	1
600029	2
600029	3
600029	10
600029	13
600031	2
600031	4
600031	6

Nomor Faktur	*Items*
600035	9
600035	10
600035	12
600035	14
600035	16
600036	15
600037	2
600037	4
600037	7
600037	9
600037	10
600037	14
600037	16
600039	1
600039	3
600039	5
600039	9
600039	11
600041	2
600041	4
600041	5
600041	7
600041	8
600041	12
600043	2
600043	5
600043	7
600043	8

Nomor Faktur	*Items*
600004	13
600004	14
600005	1
600005	3
600005	6
600005	7
600005	15
600005	16
600006	5
600006	6
600006	8
600006	9

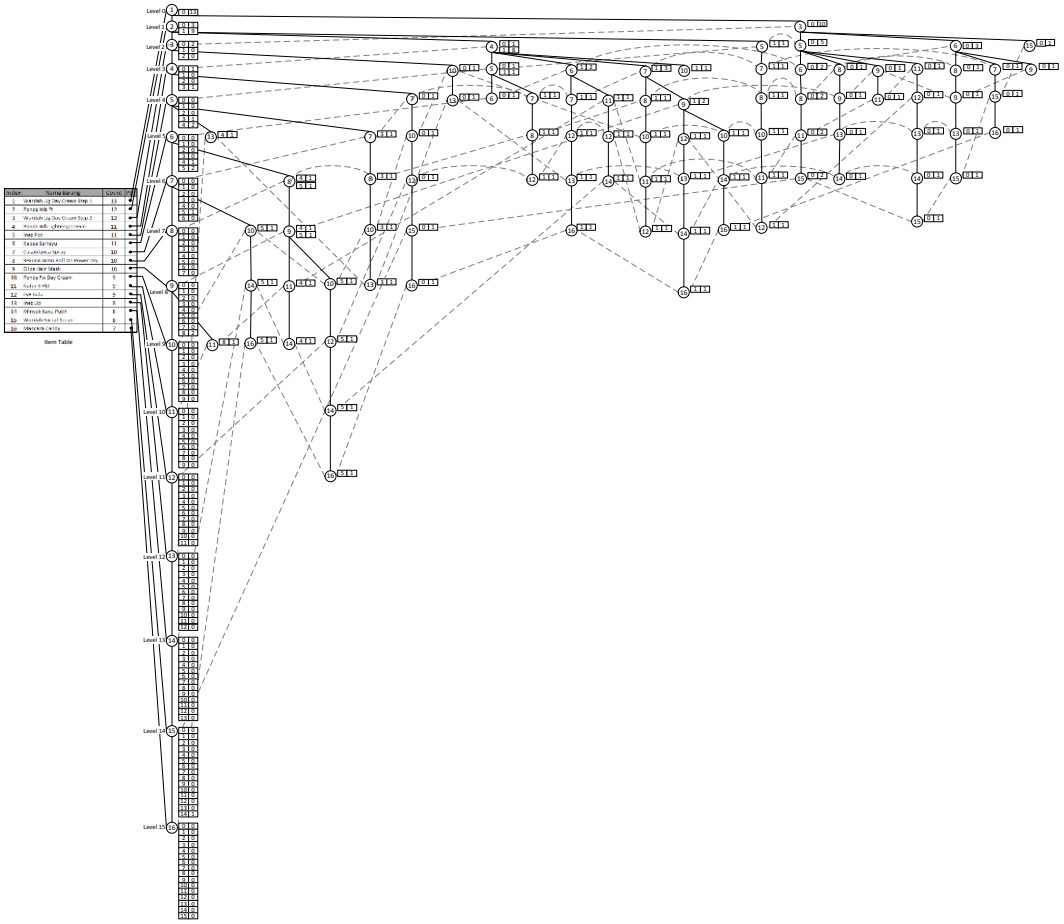
Nomor Faktur	*Items*
600013	9
600014	1
600014	3
600014	15
600016	2
600016	4
600016	10
600017	5
600017	13
600021	2
600021	4
600021	7

Nomor Faktur	*Items*
600031	11
600031	12
600031	14
600033	4
600033	5
600033	7
600033	8
600033	10
600033	13
600035	6
600035	8

Nomor Faktur	*Items*
600043	10
600043	11
600043	12
600044	1
600044	3
600044	5
600044	11
600044	12
600044	13
600044	14
600044	15

4. Membuat CFP-Tree

Setelah proses *mapping* kemudian proses pembentukan *CFP-Tree.* Untuk proses pembentukan *CPF-Tree* dapat dilihat pada lampiran D. Berikut adalah *Global CFP-Tree* yang terbentuk dari seluruh transaksi, yang diacu pada gambar 3.1



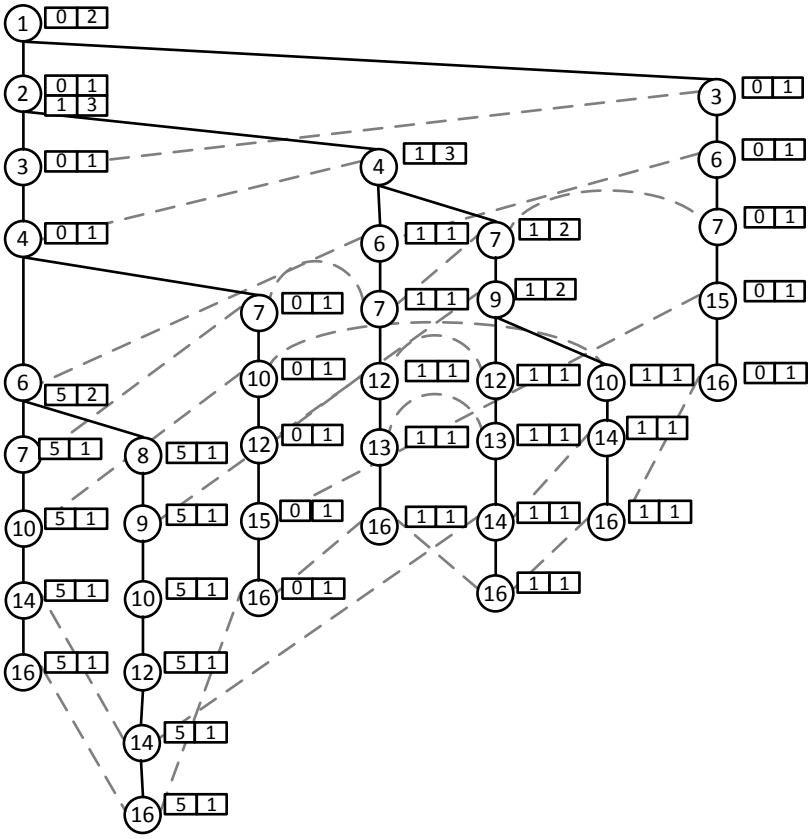
Gambar 3.1 ***Global CFP-Tree***

5. Mencari *Frequent Itemset*

Setelah tahap pembangunan CFP-*Tree* dari sekumpulan data transaksi, akan diterapkan algoritma *CT-Pro* untuk mencari *frequent itemset* yang signifikan. Berikut adalah langkah-langkah dari algoritma CT-Pro :

1. Lakukan pencarian *node* yang berkaitan dengan *item* dimana pencarian dimulai dari *global item* dengan *support count* terkecil sampai *global item* dengan *support count* terbesar karena *CT-Pro* bekerja *Bottom-Up* pada *Global CFP-Tree*. Dalam pencarian *local Frequent Pattern Tree* berisikan *prefix path* (lintasan *prefix*) dan *suffix pattern* (pola akhiran). Untuk setiap pencarian *suffix* dilakukan dengan mencari lintasan *prefix* dari *suffix* yang dicari.
2. Dari semua *node* yang ditemukan untuk setiap *item* inilah yang disebut dengan *Local Frequent item* dan digunakan untuk membuat *Local item tabel* yang pembuatannya dilakukan berdasarkan jumalah minimun *support* yang telah ditentukan.
3. Selanjutnya dibuat *Local CFP-Tree* berdasarkan *Local Item Tabel* yang terbentuk.
4. Dari *Local CFP-Tree* yang ada, kita dapat membuat *frequent pattern*nya. Dari hasil *frequent pattern* ini akan dilakukan *mining* dengan rumus *confidence.*

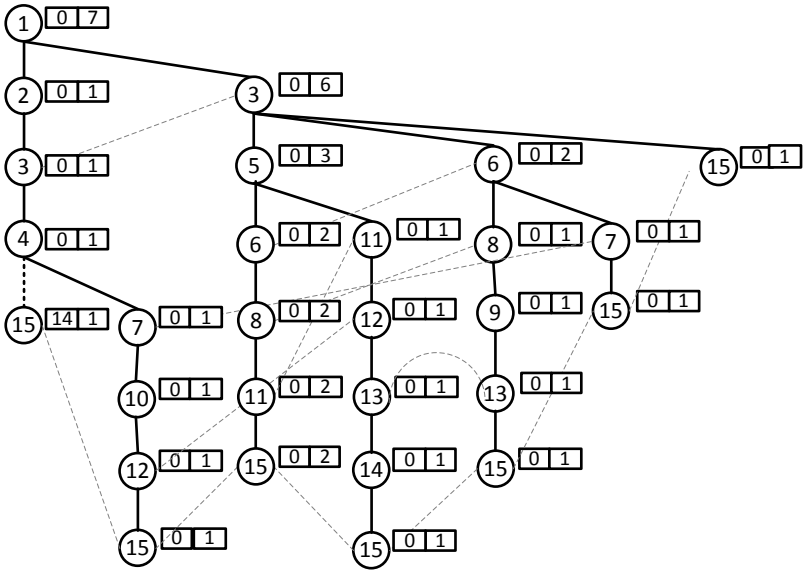
- Untuk pencarian *localFrequentPatternTree* untuk *item* 16. Cari jalur (*node*) pada *Global CFP-Tree* yang hanya berakhiran di *item* 16, untuk jelasnya dapat dilihat pada gambar berikut :



Gambar 3.2 *CFP-Tree* berakhiran *item* 16

Dari gambar 3.2 diatas didapat jumlah *support count* dari setiap *node* yaitu 1(2), 2(4), 3(2), 4(4), 6(4), 7(6), 8(1), 9(3), 10(4), 12(4), 13(2), 14(4) dan 15(2), karena *minimum suport*nya adalah 7, maka untuk *node* yang tidak memenuhi nilai *minmum support* tidak dimasukan dalam *local item tabel*. Pada *node* diatas tidak ada yang memenuhi nilai *minimum support*, maka tidak terbentuk *local item tabel* sehingga untuk *Local CFP-Tree* dan *Frequent Pattern*nya juga tidak akan terbentuk.

- Untuk pencarian *localFrequentPatternTree* untuk *item* 15. Cari jalur (*node*) pada *Global CFP-Tree* yang hanya berakhiran di *item* 15, untuk jelasnya dapat dilihat pada gambar berikut :



Gambar 3.3 *CFP-Tree* berakhiran *item* 15

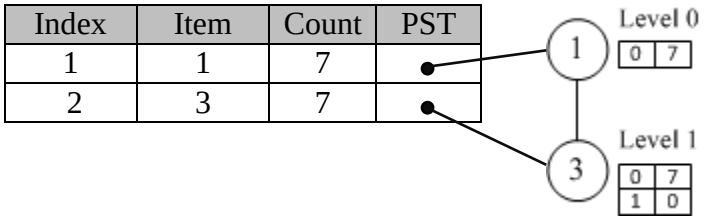
Dari gambar 3.3 diatas didapat jumlah *support count* dari setiap *node* yaitu 1(7), 2(1), 3(7), 4(1), 5(3), 6(4), 7(2), 8(3), 9(1), 10(1), 11(3), 12(2), 13(2) dan 14(1). Karena *minimum support*nya adalah 7, maka untuk *node* 2(1), 4(1), 5(3), 6(4), 7(2), 8(3), 9(1), 10(1), 11(3), 12(2), 13(2) dan 14(1) tidak memenuhi nilai *minmum support*, maka tidak dimasukan dalam *local item tabel*. Berikut merupakan *local item table* yang terbentuk :

Tabel 3.6 *Local item tabel*

Index	Item	Count
1	1	7
2	3	7

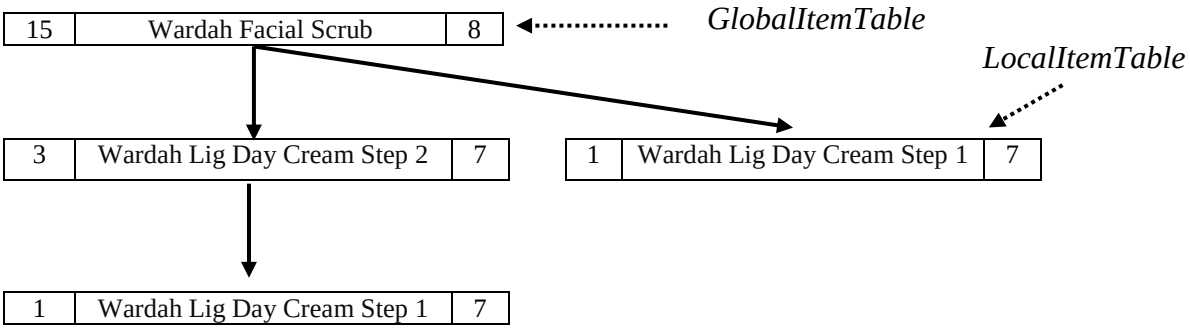
Dari *Local Item* tabel yang didapatkan pada tabel 3.6 dibuat *Local CFP-Tree*. Aturan pembentukan *Local CFP-Tree* sama dengan pembentukan *Global CFP-Tree*, yang membedakan adalah pada *Global CFP-Tree* yang digunakan dalam pembentukan tree-nya adalah *Global Item tabel* yang terbentuk dari *Global Item tabel data* sedangkan pada *Local CFP-Tree* yang digunakan dalam

pembentukan *tree*-nya adalah *Local Item tabel* yang terbentuk dari *Local Frequent item*. Berikut adalah *Local CFP-Tree* dari *item id* 15 :



Gambar 3.4 *local CFP Tree* untuk *item* 15

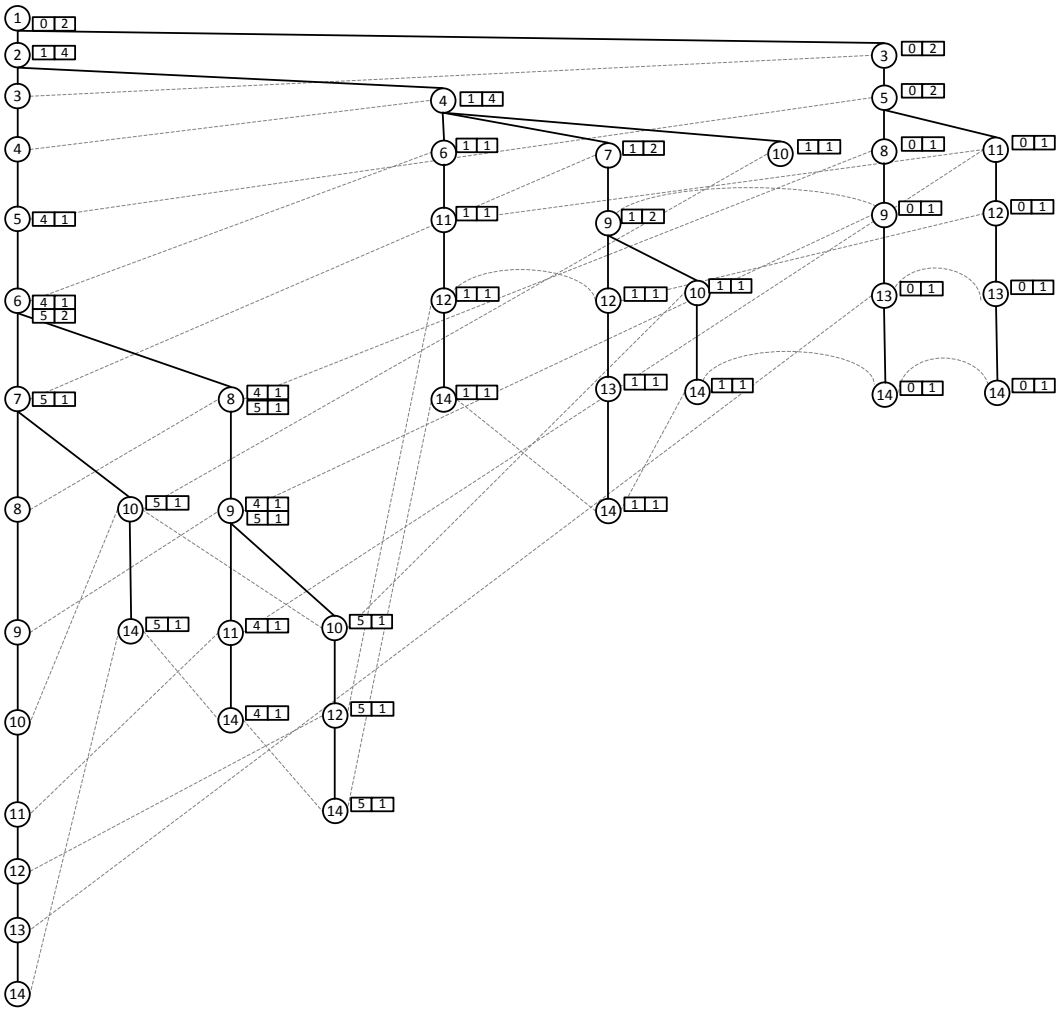
Kemudian membuat *local CFP-tree projection* untuk *item* 15, dapat dilihat pada gambar berikut :



Gambar 3.5 *local CFP-tree projection* 15

Dari *local CFP-tree projection* maka didapatkan *item frequent* untuk index 15 dengan nama barang Wardah Facial Scrub adalah sebagai berikut: (15,1:7), (15,3:7), dan (15,3,1:7).

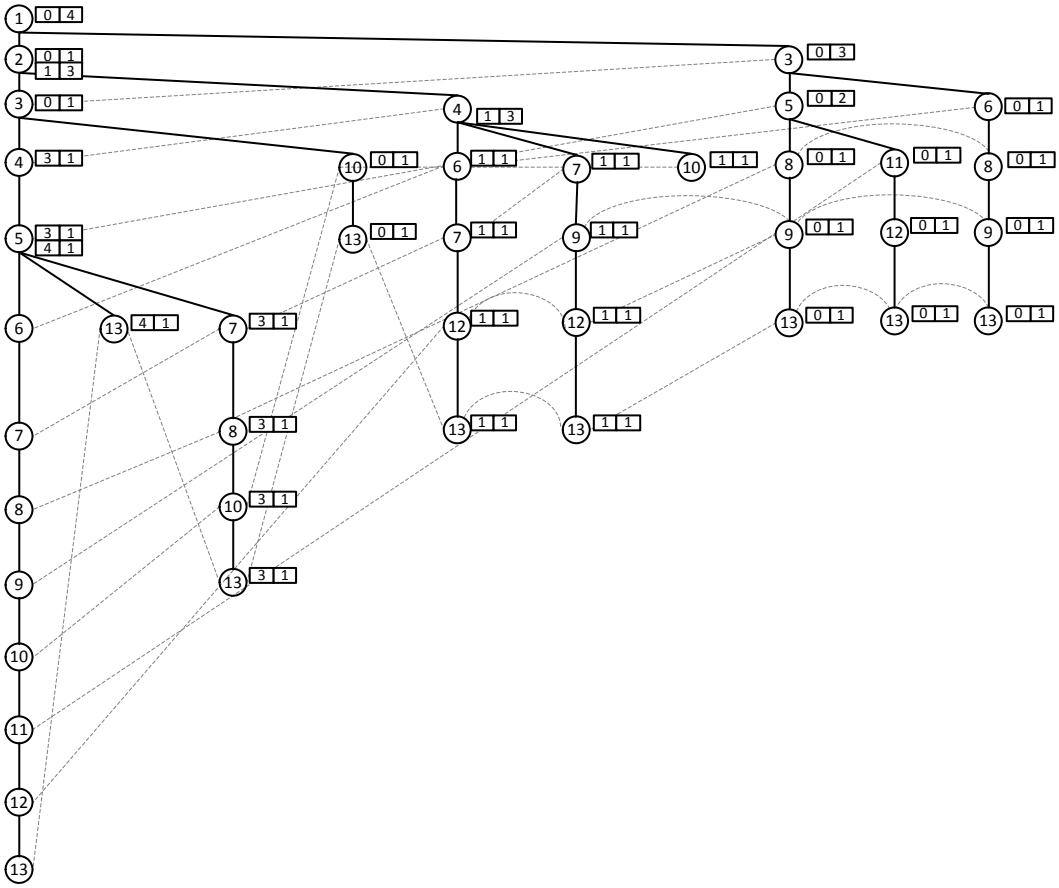
- Untuk pencarian *localFrequentPatternTree* untuk *item* 14. Cari jalur (*node*) pada *Global CFP-Tree* yang hanya berakhiran di *item* 14, untuk jelasnya dapat dilihat pada gambar berikut :



Gambar 3.6 *CFP-Tree* berakhiran *item* 14

Dari gambar 3.6 diatas didapat jumlah *support count* dari setiap *node* yaitu 1(2), 2(4), 3(2), 4(4), 5(6), 6(4), 7(3), 8(3), 9(5), 10(3), 11(3), 12(4) dan 13(3), karena *minimum suport*nya adalah 7, maka untuk *node* yang tidak memenuhi nilai *minmum support* tidak dimasukan dalam *local item tabel*. Pada *node* diatas tidak ada yang memenuhi nilai *minimum support*, maka tidak terbentuk *local item tabel* sehingga untuk *Local CFP-Tree* dan *Frequent Pattern*nya juga tidak akan terbentuk.

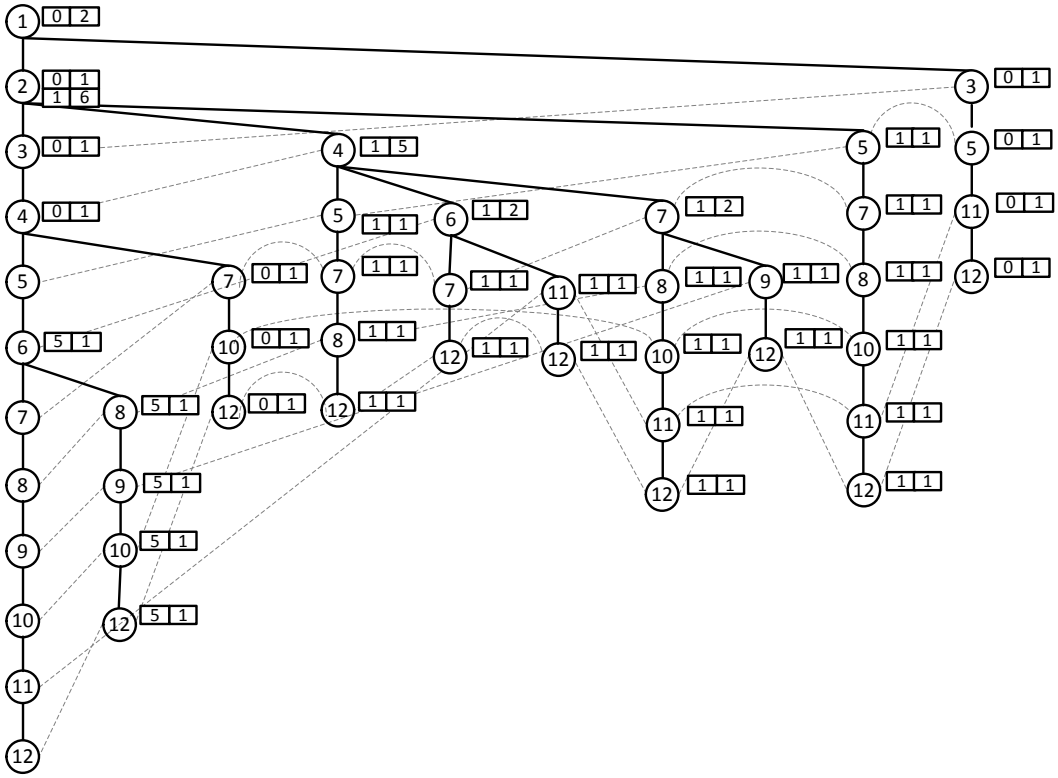
- Untuk pencarian *localFrequentPatternTree* untuk *item* 13. Cari jalur (*node*) pada *Global CFP-Tree* yang hanya berakhiran di *item* 13, untuk jelasnya dapat dilihat pada gambar berikut :



Gambar 3.7 *CFP-Tree* berakhiran *item* 13

Dari gambar 3.7 diatas didapat jumlah *support count* dari setiap *node* yaitu 1(4), 2(4), 3(4), 4(4), 5(4), 6(4), 7(3), 8(3), 9(5), 10(3), 11(3), dan 12(4) karena *minimum suport*nya adalah 7, maka untuk *node* yang tidak memenuhi nilai *minmum support* tidak dimasukan dalam *local item tabel*. Pada *node* diatas tidak ada yang memenuhi nilai *minimum support*, maka tidak terbentuk *local item tabel* sehingga untuk *Local CFP-Tree* dan *Frequent Pattern*nya juga tidak akan terbentuk.

- Untuk pencarian *localFrequentPatternTree* untuk *item* 12. Cari jalur (*node*) pada *Global CFP-Tree* yang hanya berakhiran di *item* 12, untuk jelasnya dapat dilihat pada gambar berikut :



Gambar 3.8 *CFP-Tree* berakhiran *item* 12

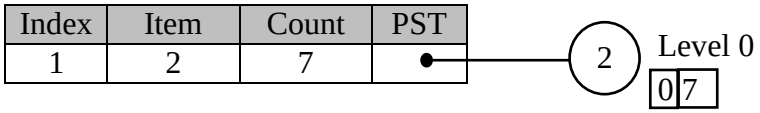
Dari gambar 3.8 diatas didapat jumlah *support count* dari setiap *node* yaitu 1(2), 2(7), 3(2), 4(6), 5(2), 6(3), 7(6), 8(4), 9(2), 10(4), dan 11(3) karena *minimum suport*nya adalah 7, maka untuk *node* yang tidak memenuhi nilai *minmum support* tidak dimasukan dalam *local item tabel*. Berikut merupakan *local item table* yang terbentuk :

Tabel 3.7 *Local item tabel*

Index	Item	Count
1	2	7

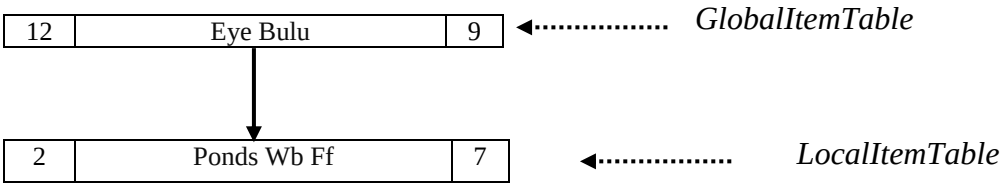
Dari *Local Item* tabel yang didapatkan pada tabel 3.7 dibuat *Local CFP-Tree*. Aturan pembentukan *Local CFP-Tree* sama dengan pembentukan *Global CFP-Tree*, yang membedakan adalah pada *Global CFP-Tree* yang digunakan dalam pembentukan tree-nya adalah *Global Item tabel* yang terbentuk dari *Global*

Item tabel data sedangkan pada *Local CFP-Tree* yang digunakan dalam pembentukan *tree*-nya adalah *Local Item tabel* yang terbentuk dari *Local Frequent item*. Berikut adalah *Local CFP-Tree* dari *item id* 12 :



Gambar 3.9 *local CFP Tree* untuk *item* 12

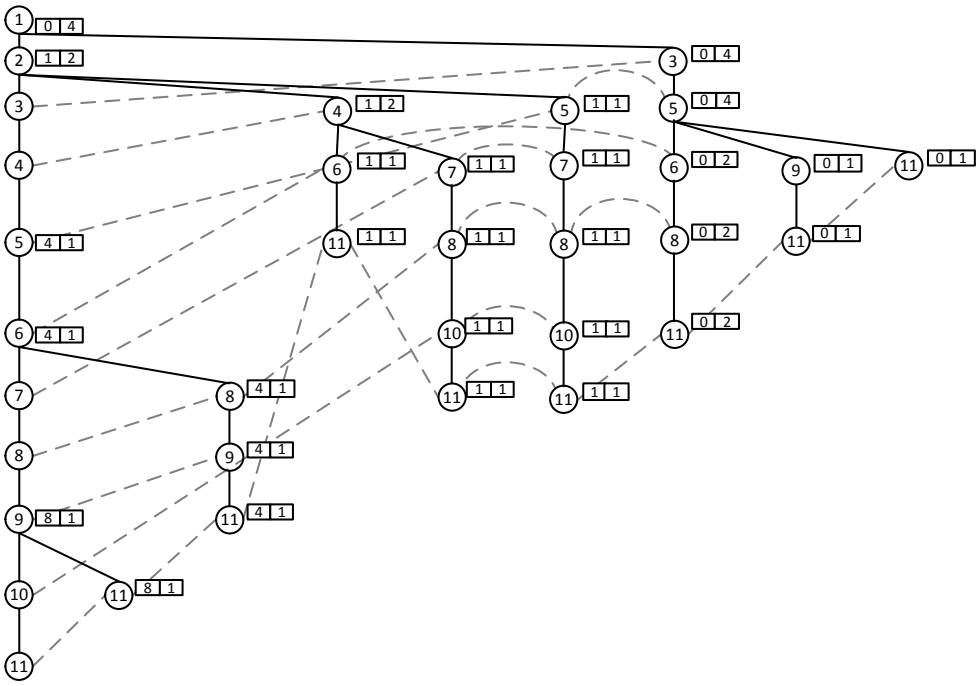
Kemudian membuat *local CFP-tree projection* untuk *item* 12, dapat dilihat pada gambar berikut :



Gambar 3.10 *local CFP-tree projection* 12

Dari *local CFP-tree projection* maka didapatkan *item frequent* untuk index 12 dengan nama barang Eye Bulu adalah sebagai berikut: (12,2:7).

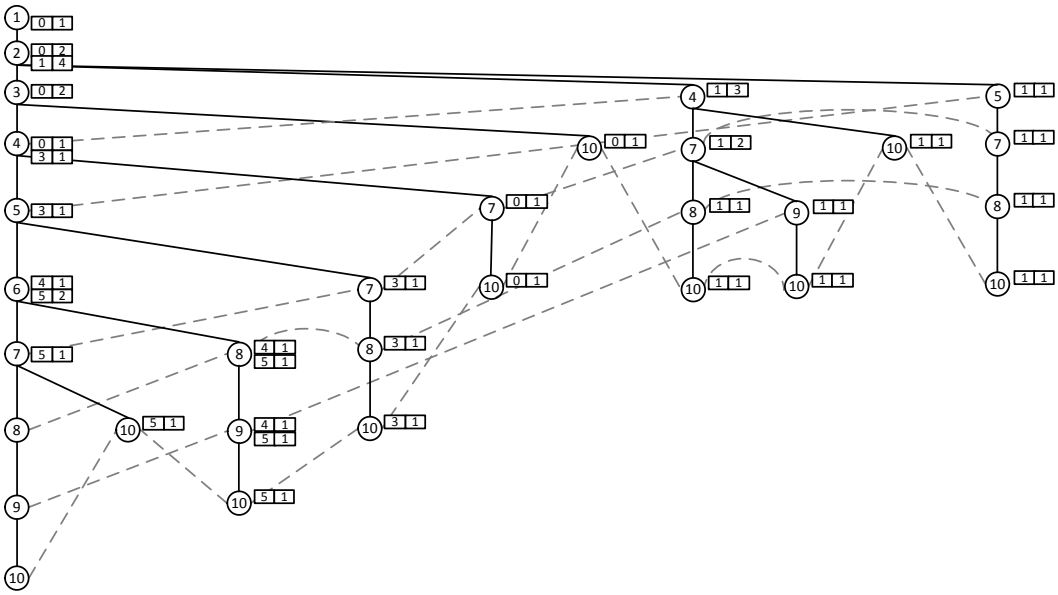
- Untuk pencarian *localFrequentPatternTree* untuk *item* 11. Cari jalur (*node*) pada *Global CFP-Tree* yang hanya berakhiran di *item* 11, untuk jelasnya dapat dilihat pada gambar berikut :



Gambar 3.11 *CFP-Tree* berakhiran *item* 11

Dari gambar 3.11 diatas didapat jumlah *support count* dari setiap *node* yaitu 1(4), 2(2), 3(4), 4(2), 5(6), 6(4), 7(2), 8(5), 9(3), dan 10(2) karena *minimum suport*nya adalah 7, maka untuk *node* yang tidak memenuhi nilai *minmum support* tidak dimasukan dalam *local item tabel*. Pada *node* diatas tidak ada yang memenuhi nilai *minimum support*, maka tidak terbentuk *local item tabel* sehingga untuk *Local CFP-Tree* dan *Frequent Pattern*nya juga tidak akan terbentuk.

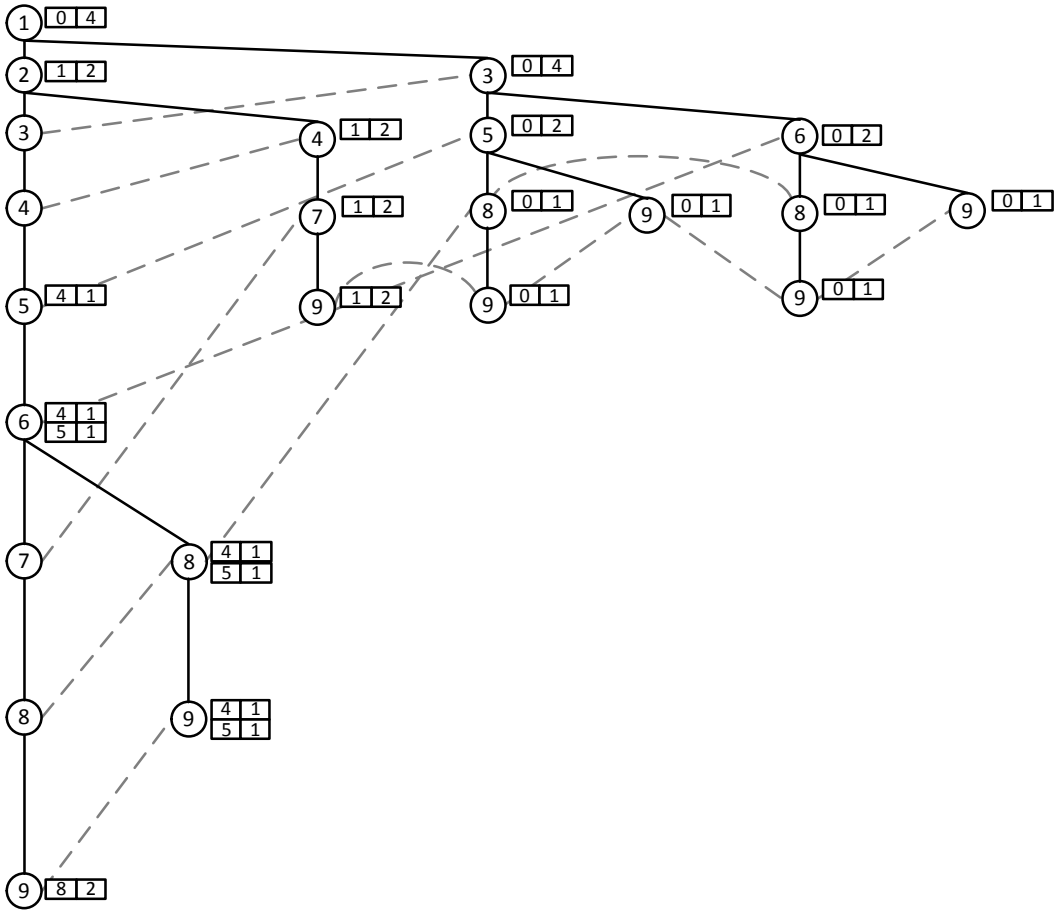
- Untuk pencarian *localFrequentPatternTree* untuk *item* 10. Cari jalur (*node*) pada *Global CFP-Tree* yang hanya berakhiran di *item* 10, untuk jelasnya dapat dilihat pada gambar berikut :



Gambar 3.12 *CFP-Tree* berakhiran *item* 10

Dari gambar 3.12 diatas didapat jumlah *support count* dari setiap *node* yaitu 1(1), 2(6), 3(2), 4(5), 5(2), 6(3), 7(6), 8(4), dan 9(3) karena *minimum suport*nya adalah 7, maka untuk *node* yang tidak memenuhi nilai *minmum support* tidak dimasukan dalam *local item tabel*. Pada *node* diatas tidak ada yang memenuhi nilai *minimum support*, maka tidak terbentuk *local item tabel* sehingga untuk *Local CFP-Tree* dan *Frequent Pattern*nya juga tidak akan terbentuk.

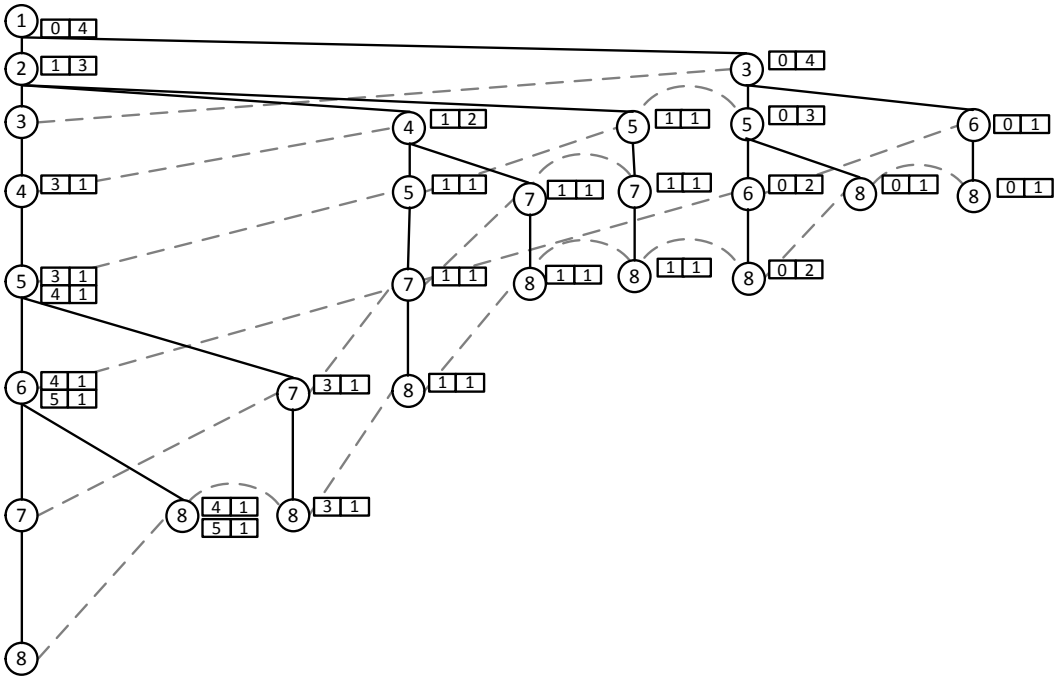
- Untuk pencarian *localFrequentPatternTree* untuk *item* 9. Cari jalur (*node*) pada *Global CFP-Tree* yang hanya berakhiran di *item* 9, untuk jelasnya dapat dilihat pada gambar berikut :



Gambar 3.13 *CFP-Tree* berakhiran *item* 9

Dari gambar 3.13 diatas didapat jumlah *support count* dari setiap *node* yaitu 1(4), 2(2), 3(4), 4(2), 5(3), 6(4), 7(2), dan 8(4) karena *minimum suport*nya adalah 7, maka untuk *node* yang tidak memenuhi nilai *minmum support* tidak dimasukan dalam *local item tabel*. Pada *node* diatas tidak ada yang memenuhi nilai *minimum support*, maka tidak terbentuk *local item tabel* sehingga untuk *Local CFP-Tree* dan *Frequent Pattern*nya juga tidak akan terbentuk.

- Untuk pencarian *localFrequentPatternTree* untuk *item* 8. Cari jalur (*node*) pada *Global CFP-Tree* yang hanya berakhiran di *item* 8, untuk jelasnya dapat dilihat pada gambar berikut :



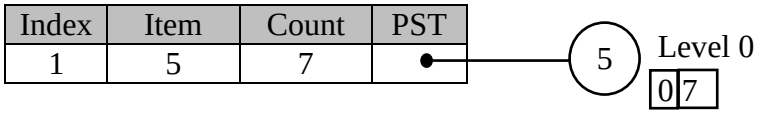
Gambar 3.14 *CFP-Tree* berakhiran *item* 8

Dari gambar 3.14 diatas didapat jumlah *support count* dari setiap *node* yaitu 1(4), 2(2), 3(4), 4(3), 5(7), 6(5), dan 7(4) karena *minimum suport*nya adalah 7, maka untuk *node* yang tidak memenuhi nilai *minmum support* tidak dimasukan dalam *local item tabel*. Berikut merupakan *local item table* yang terbentuk :

Tabel 3.8 *Local item tabel*

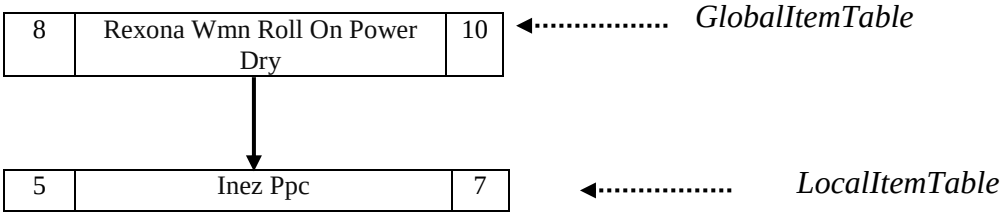
Index	Item	Count
1	5	7

Dari *Local Item* tabel yang didapatkan pada tabel 3.8 dibuat *Local CFP-Tree*. Aturan pembentukan *Local CFP-Tree* sama dengan pembentukan *Global CFP-Tree*, yang membedakan adalah pada *Global CFP-Tree* yang digunakan dalam pembentukan tree-nya adalah *Global Item tabel* yang terbentuk dari *Global Item tabel data* sedangkan pada *Local CFP-Tree* yang digunakan dalam pembentukan *tree*-nya adalah *Local Item tabel* yang terbentuk dari *Local Frequent item*. Berikut adalah *Local CFP-Tree* dari *item id* 8 :



Gambar 3.15 *local CFP Tree* untuk *item* 12

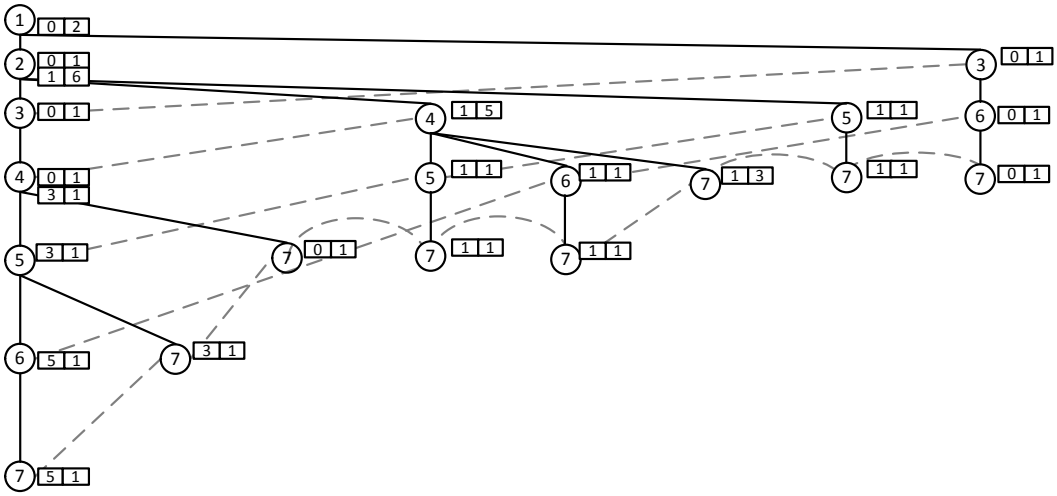
Kemudian membuat *local CFP-tree projection* untuk *item* 12, dapat dilihat pada gambar berikut :



Gambar 3.16 *local CFP-tree projection* 8

Dari *local CFP-tree projection* maka didapatkan *item frequent* untuk index 8 dengan nama barang Rexona Wmn Roll On Power Dry adalah sebagai berikut: (8,5:7).

- Untuk pencarian *localFrequentPatternTree* untuk *item* 7. Cari jalur (*node*) pada *Global CFP-Tree* yang hanya berakhiran di *item* 7, untuk jelasnya dapat dilihat pada gambar berikut :



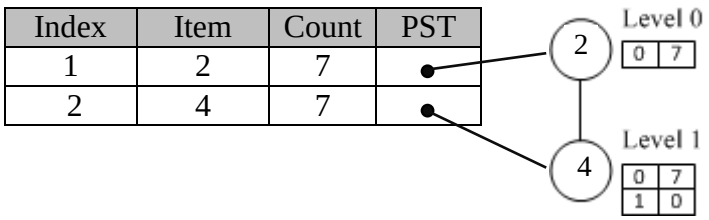
Gambar 3.17 *CFP-Tree* berakhiran *item* 7

Dari gambar 3.17 diatas didapat jumlah *support count* dari setiap *node* yaitu 1(2), 2(7), 3(2), 4(7), 5(3), dan 6(3) karena *minimum suport*nya adalah 7, maka untuk *node* yang tidak memenuhi nilai *minmum support* tidak dimasukan dalam *local item tabel*. Berikut merupakan *local item table* yang terbentuk :

Tabel 3.9 *Local item tabel*

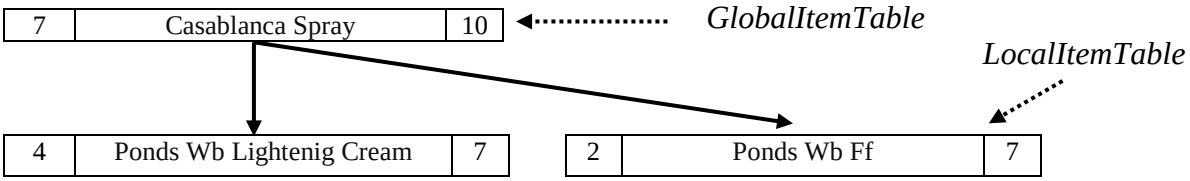
Index	Item	Count
1	2	7
2	4	7

Dari *Local Item* tabel yang didapatkan pada tabel 3.9 dibuat *Local CFP-Tree*. Aturan pembentukan *Local CFP-Tree* sama dengan pembentukan *Global CFP-Tree*, yang membedakan adalah pada *Global CFP-Tree* yang digunakan dalam pembentukan tree-nya adalah *Global Item tabel* yang terbentuk dari *Global Item tabel data* sedangkan pada *Local CFP-Tree* yang digunakan dalam pembentukan *tree*-nya adalah *Local Item tabel* yang terbentuk dari *Local Frequent item*. Berikut adalah *Local CFP-Tree* dari *item id* 7 :



Gambar 3.18 *local CFP Tree* untuk *item* 7

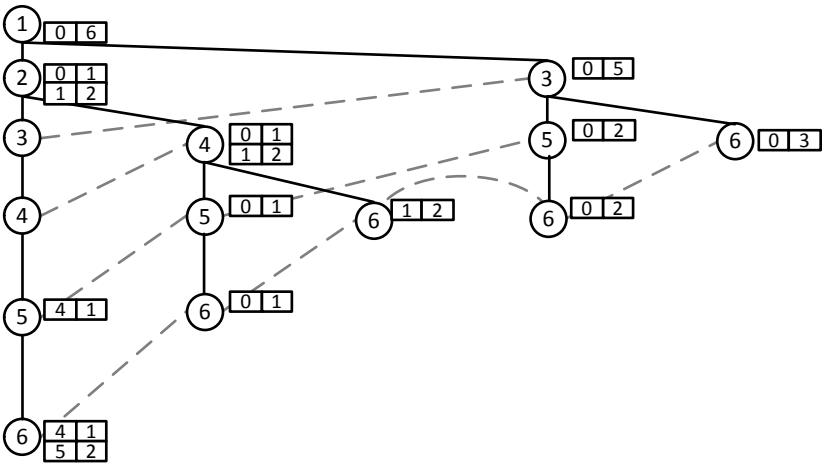
Kemudian membuat *local CFP-tree projection* untuk *item* 7, dapat dilihat pada gambar berikut :



Gambar 3.19 *local CFP-tree projection* 7

Dari *local CFP-tree projection* maka didapatkan *item frequent* untuk index 7 dengan nama barang Casablanca Spray adalah sebagai berikut: (7,2:7) dan (7,4:7).

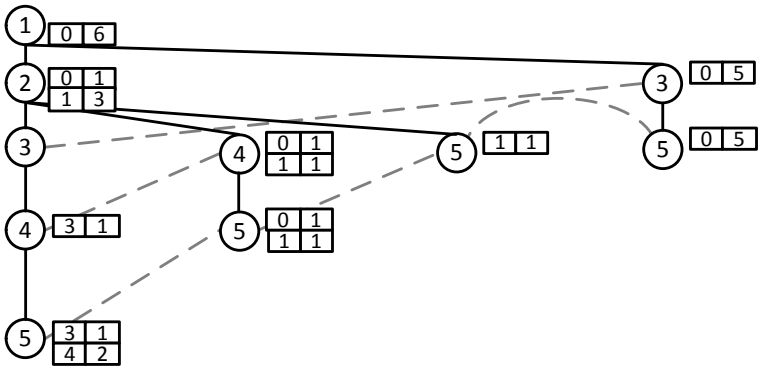
- Untuk pencarian *localFrequentPatternTree* untuk *item* 6. Cari jalur (*node*) pada *Global CFP-Tree* yang hanya berakhiran di *item* 6, untuk jelasnya dapat dilihat pada gambar berikut :



Gambar 3.20 *CFP-Tree* berakhiran *item* 6

Dari gambar 3.20 diatas didapat jumlah *support count* dari setiap *node* yaitu 1(6), 2(3), 3(5), 4(3), dan 5(4) karena *minimum suport*nya adalah 7, maka untuk *node* yang tidak memenuhi nilai *minmum support* tidak dimasukan dalam *local item tabel*. Pada *node* diatas tidak ada yang memenuhi nilai *minimum support*, maka tidak terbentuk *local item tabel* sehingga untuk *Local CFP-Tree* dan *Frequent Pattern*nya juga tidak akan terbentuk.

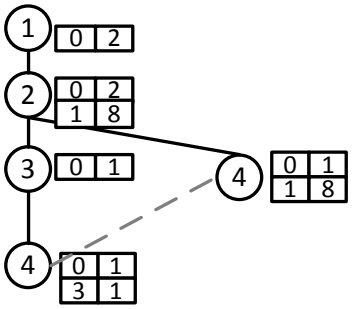
- Untuk pencarian *localFrequentPatternTree* untuk *item* 5. Cari jalur (*node*) pada *Global CFP-Tree* yang hanya berakhiran di *item* 5, untuk jelasnya dapat dilihat pada gambar berikut :



Gambar 3.21 *CFP-Tree* berakhiran *item* 5

Dari gambar 3.21 diatas didapat jumlah *support count* dari setiap *node* yaitu 1(6), 2(3), 3(5), dan 4(4) karena *minimum suport*nya adalah 7, maka untuk *node* yang tidak memenuhi nilai *minmum support* tidak dimasukan dalam *local item tabel*. Pada *node* diatas tidak ada yang memenuhi nilai *minimum support*, maka tidak terbentuk *local item tabel* sehingga untuk *Local CFP-Tree* dan *Frequent Pattern*nya juga tidak akan terbentuk.

- Untuk pencarian *localFrequentPatternTree* untuk *item* 4. Cari jalur (*node*) pada *Global CFP-Tree* yang hanya berakhiran di *item* 4, untuk jelasnya dapat dilihat pada gambar berikut :



Gambar 3.22*CFP-Tree* berakhiran *item* 4

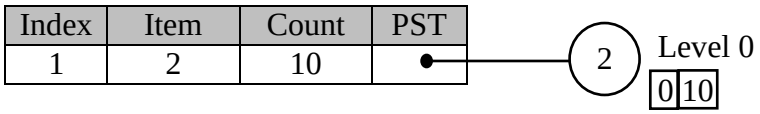
Dari gambar 3.22 diatas didapat jumlah *support count* dari setiap *node* yaitu 1(2), 2(10), dan 3(1), karena *minimum suport*nya adalah 7, maka untuk *node* yang tidak memenuhi nilai *minmum support* tidak dimasukan dalam *local item tabel*. Berikut merupakan node yang memenuhi nilai *minimum support* yang diacu pada tabel 3.10.

Tabel 3.10 *Local item tabel* untuk *item* 4

Index	Item	Count
1	2	10

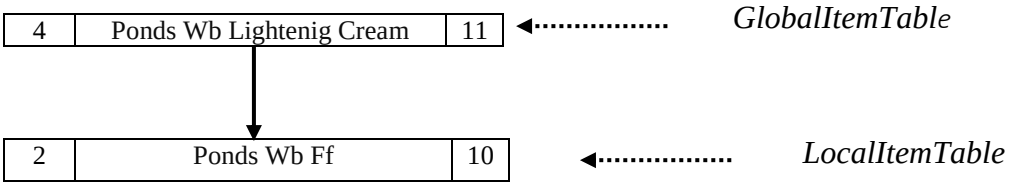
Dari *Local Item* tabel yang didapatkan pada tabel 3.10 dibuat *Local CFP-Tree*. Aturan pembentukan *Local CFP-Tree* sama dengan pembentukan *Global CFP-Tree*, yang membedakan adalah pada *Global CFP-Tree* yang digunakan dalam pembentukan tree-nya adalah *Global Item tabel* yang terbentuk dari *Global Item tabel data* sedangkan pada *Local CFP-Tree* yang digunakan dalam

pembentukan *tree*-nya adalah *Local Item tabel* yang terbentuk dari *Local Frequent item*. Berikut adalah *Local CFP-Tree* dari *item id* 4 :



Gambar 3.23 *local CFP Tree* untuk *item* 4

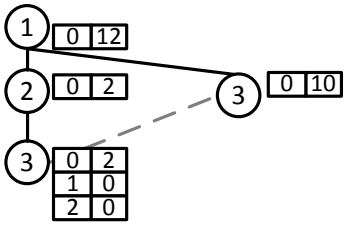
Kemudian membuat *local CFP-tree projection* untuk *item* 4, dapat dilihat pada gambar berikut :



Gambar 3.24 *local CFP-tree projection* 4

Dari *local CFP-tree projection* maka didapatkan *item frequent* untuk index 4 dengan nama barang Ponds Wb Lightenig Cream adalah sebagai berikut: (4,2:10).

- Untuk pencarian *localFrequentPatternTree* untuk *item* 3. Cari jalur (*node*) pada *Global CFP-Tree* yang hanya berakhiran di *item* 3, untuk jelasnya dapat dilihat pada gambar berikut :



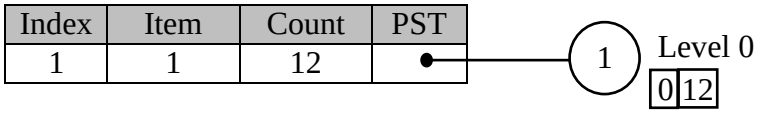
Gambar 3.25 *CFP-Tree* berakhiran *item* 3

Dari gambar 3.25 diatas didapat jumlah *support count* dari setiap *node* yaitu 1(12) dan 2(2) karena *minimum suport*nya adalah 7, maka untuk *node* yang tidak memenuhi nilai *minmum support* tidak dimasukan dalam *local item tabel*. Berikut merupakan node yang memenuhi nilai *minimum support* yang diacu pada tabel 3.11.

Tabel 3.11 *Local item tabel* untuk *item* 3

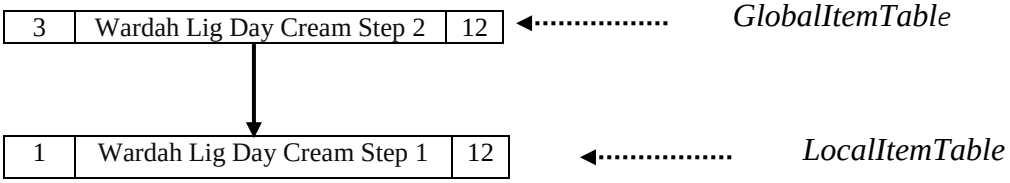
Index	Item	Count
1	1	12

Dari *Local Item* tabel yang didapatkan pada tabel 3.11 dibuat *Local CFP-Tree*. Aturan pembentukan *Local CFP-Tree* sama dengan pembentukan *Global CFP-Tree*, yang membedakan adalah pada *Global CFP-Tree* yang digunakan dalam pembentukan tree-nya adalah *Global Item tabel* yang terbentuk dari *Global Item tabel data* sedangkan pada *Local CFP-Tree* yang digunakan dalam pembentukan *tree*-nya adalah *Local Item tabel* yang terbentuk dari *Local Frequent item*. Berikut adalah *Local CFP-Tree* dari *item id* 3 :



Gambar 3.26 *local CFP Tree* untuk *item* 3

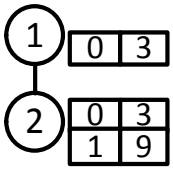
Kemudian membuat *local CFP-tree projection* untuk *item* 3, dapat dilihat pada gambar berikut :



Gambar 3.27 *local CFP-tree projection* 3

Dari *local CFP-tree projection* maka didapatkan *item frequent* untuk index 3 dengan nama barang Wardah Lig Day Cream Step 2 adalah sebagai berikut: (3,1:12).

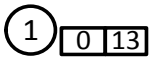
- Untuk pencarian *localFrequentPatternTree* untuk *item* 2. Cari jalur (*node*) pada *Global CFP-Tree* yang hanya berakhiran di *item* 2, untuk jelasnya dapat dilihat pada gambar berikut :



Gambar 3.28 *CFP-Tree* berakhiran *item* 2

Dari gambar 3.28 diatas didapat jumlah *support count* dari setiap *node* yaitu 1(3), karena *minimum suport*nya adalah 7, maka untuk *node* yang tidak memenuhi nilai *minmum support* tidak dimasukan dalam *local item tabel*. Pada *node* diatas tidak ada yang memenuhi nilai *minimum support*, maka tidak terbentuk *local item tabel* sehingga untuk *Local CFP-Tree* dan *Frequent Pattern*nya juga tidak akan terbentuk.

- Untuk pencarian *localFrequentPatternTree* untuk *item* 1. Cari jalur (*node*) pada *Global CFP-Tree* yang hanya berakhiran di *item* 1, untuk jelasnya dapat dilihat pada gambar berikut :



Gambar 3.29 *CFP-Tree* berakhiran *item* 1

Dari gambar 3.29 diatas tidak ada *node* yang ditemukan, maka tidak terbentuk *local item tabel* sehingga untuk *Local CFP-Tree* dan *Frequent Pattern*nya juga tidak akan terbentuk.

Dari *frequent pattern* yang didapat, akan di-*generate* untuk mendapatkan *rule* dengan *minimum confidence* = 50%. Dihitung dengan persamaan 2-4 sebagai berikut :

$$\text{Confidence} = (A \rightarrow B) = \frac{\text{Jumlah Transaksi Yang Mengandung A dan B}}{\text{Jumlah Transaksi Yang Mengandung A}}$$

(Persamaan 2-4)

Berikut adalah hasil generate rule diacu pada tabel 3.8.

Tabel 3.12 Hasil *Generate Rule*

Frequent Itemset	*Confidence*
15 → 1	(7/8) * 100% = 87,5%
15 → 3	(7/8) * 100% = 87,5%
15 → 3,1	(7/8) * 100% = 87,5%
12 → 2	(7/9) * 100% = 77,78%
8 → 5	(7/10) * 100% = 70%
7 → 2	(7/10) * 100% = 70%
7 → 4	(7/10) * 100% = 70%
4 → 2	(10/11) * 100% = 90,90%
3 → 1	(12/12) * 100% = 100%

Setelah perhitungan *confidence* diketahui maka selanjutnya dilakukan pengecekan, jika tidak memenuhi *minimum confidence* produk tersebut akan dihapus untuk menemukan *rule*. Setiap *id global item* akan dikembalikan sesuai dengan nama barang dari masing-masing *id* agar pembacaan *rule* lebih jelas. Berikut hasil penentuan paket kosmetik hemat dalam bentuk awal :

Tabel 3.13 Hasil Paket Kosmetik dalam Bentuk Awal

Rules	Paket Kosmetik	*Confidence*
15 → 1	Wardah Facial Scrub → Wardah Lig Day Cream Step 1	87,5%
15 → 3	Wardah Facial Scrub → Wardah Lig Day Cream Step 2	87,5%
15 → 3,1	Wardah Facial Scrub → Wardah Lig Day Cream Step 2, Wardah Lig Day Cream Step 1	87,5%
12 → 2	Eye Bulu → Ponds Wb Ff	77,78%
8 → 5	Rexona Wmn Roll On Power Dry → Inez Ppc	70%
7 → 2	Casablanca Spray → Ponds Wb Ff	70%
7 → 4	Casablanca Spray → Ponds Wb Lightenig Cream	70%
4 → 2	Ponds Wb Lightenig Cream → Ponds Wb Ff	90,91%
3 → 1	Wardah Lig Day Cream Step 2 → Wardah Lig Day Cream Step 1	100%

Dalam penentuan kombinasi produk yang akan dijual dalam bentuk paket, pihak Kartika Kosmetik tidak memiliki batasan khusus, sehingga dari rule yang terbentuk dapat digunakan sebagai pertimbangan dalam menentukan paket produk yang akan dijual. Dari hasil pengolahan data ini kita dapat mengambil sebuah keputusan mana produk yang cocok untuk dijadikan sebuah paket penjualan produk kosmetik. Sebagai contoh, bila kita ingin membuat sebuah paket penjualan kosmetik yang berisikan dua buah produk, maka kita bisa menggunakan paket Ponds Wb Lightenig Cream dan Ponds Wb Ff dengan nilai confidence 100% sebagai priduk-produk yang akan dijadikan paket penjualan. Namun jika ingin membuat paket penjualan kosmetik dengan jumlah produk lebih dari dua maka data transaksi yang digunakan bisa ditambahkan atau nilai *support* harus diturunkan agar *rules* yang didapatkan menghasilkan lebih dari dua buah kombinasi produk sehingga paket dapat dibentuk dengan jumlah item lebih banyak.

Berikut merupakan paket produk yang terbentuk :

Tabel 3.14 Hasil Paket Kosmetik yang terbentuk

Paket	Paket Produk
Paket 1	Wardah Facial Scrub dan Wardah Lig Day Cream Step 1
Paket 2	Wardah Facial Scrub dan Wardah Lig Day Cream Step 2
Paket 3	Wardah Facial Scrub, Wardah Lig Day Cream Step 2 dan Wardah Lig Day Cream Step 1
Paket 4	Eye Bulu dan Ponds Wb Ff
Paket 5	Rexona Wmn Roll On Power Dry dan Inez Ppc
Paket 6	Casablanca Spray dan Ponds Wb Ff
Paket 7	Casablanca Spray dan Ponds Wb Lightenig Cream
Paket 8	Ponds Wb Lightenig Cream dan Ponds Wb Ff
Paket 9	Wardah Lig Day Cream Step 2 dan Wardah Lig Day Cream Step 1

3.1.4 Analisis Non Fungsional dan Kebutuhan Non Fungsional

Analisis non fungsional dan kebutuhan non fungsional dibagi menjadi beberapa tahapan analisis, yaitu analisis pengguna, analisis perangkat lunak dan analisis perangkat keras.

3.1.4.1 Analisis Pengguna

User atau pengguna yang akan mengoperasikan aplikasi ini merupakan manager yang bertindak sebagai pengolah bisnis. Analisis Pengguna dapat dilihat pada tabel 3.10.

Tabel 3.15 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.

3.1.4.2 Analisis Perangkat Lunak

Analisis kebutuhan perangkat lunak dalam pembangunan aplikasi digunakan untuk mengetahui dan mengoptimalkan implementasi dari aplikasi yang akan dibangun. Berikut merupakan Analisis terhadap perangkat lunak di bagian manager toko Kartika Kosmetik.

1. Analisis spesifikasi perangkat lunak pada sistem yang sedang berjalan di bagian manager toko Kartika Kosmetik adalah:
   a. Sistem Operasi Windows 7
   b. Microsoft Office 2010
2. Analisis spesifikasi kebutuhan perangkat lunak untuk menjalankan sistem 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 bagian manager toko Kartika Kosmetik, spesifikasi perangkat lunak yang digunakan masih kurang. Untuk mengatasi hal tersebut akan di install MySQL dan juga Visual Studio pada komputer yang digunakan.

3.1.4.3 Analisis Perangkat Keras

Analisis kebutuhan perangkat keras digunakan untuk memudahkan proses perancangan dan implementasi. Berikut merupakan Analisis terhadap perangkat keras di bagian manager toko Kartika Kosmetik.

1. Analisis spesifikasi perangkat keras pada system yang sedang berjalan di bagian manager toko Kartika Kosmetik adalah :
   a. *Processor* : Intel Core i3-550 3.2 Ghz
   b. *Hard disk* : 500 GB
   c. *Memory* : 4 GB
2. Analisis spesifikasi kebutuhan perangkat keras untuk menjalankan sistem yang akan dibangun adalah :
   a. *Processor* : berkecepatan minimal 2 GHz
   b. *Hard disk* 500 MB untuk menyimpan file
   c. *Memory* : 1 GB
3. Evaluasi kebutuhan perangkat lunak

   Berdasarkan analisis spesifikasi perangkat keras yang ada di bagian manager toko Kartika Kosmetik, perangkat keras yang digunakan sudah cukup untuk dapat menjalankan aplikasi Data *Mining*.

3.1.5 Analisis Kebutuhan Fungsional

Analisis kebutuhan fungsional adalah analisis terhadap kebutuhan secara fungsional baik dalam aliran data ataupun informasi. Analisis kebutuhan data fungsional yang digunakan dalam sistem ini digambarkan dengan analisis Pemrograman Berorientasi Objek dengan menggunakan konsep UML (*Unified Modeling Languages*).

3.1.5.1 Diagram *Use Case*

Use case atau diagram *use case* merupakan pemodelan untuk kelakuan (behavior) sistem yang akan dibuat. Berikut adalah tabel definisi *user* yang diacu pada tabel 3.11 dan tabel deskripsi *use case* yang diacu pada tabel 3.12.

3.16 Tabel Definisi *User*

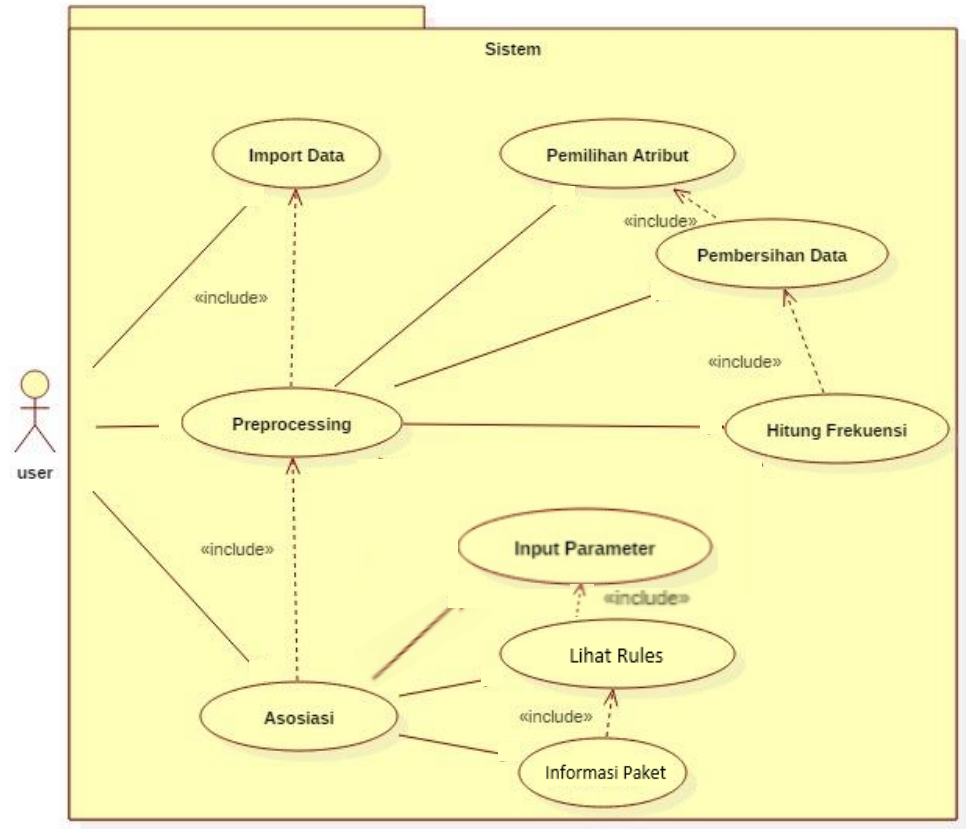
User	**Deskripsi**
User	*User* bertugas untuk melakukan *import* data, memasukkan *Minimum Support* dan *Minimum Confidence*

3.17 Tabel Deskripsi Use Case

No	***Use Case***	**Deskripsi**
1	*Import* Data	Import data harus dilakukan user terlebih dahulu, data yang diimport merupakan data transaksi yang selanjutnya data tersebut akan disimpan kedalam database
2	*Preprocessing*	Data transaksi yang sudah tersimpan dalam database kemudian akan dilakukan proses pemilihan atribut, pembersihan data dan hitung frekuensi
3	Pemilihan Atribut	Sistem melakukan pemilihan atribut Nomor Faktur dan Nama Barang, proses ini hanya bisa dilakukan jika user sudah melakukan import data
4	Pembersihan Data	Sistem melakukan pembersihan data pada Nomor Faktur yang mengandung Nama Barang tunggal, proses ini hanya bisa dilakukan jika user sudah melakukan *import* data dan pemilihan

No	*Use Case*	Deskripsi
		atribut
5	Hitung *frekuensi*	Sistem melakukan proses perhitungan kemunculan tiap item pada data transaksi, proses ini hanya bias dilakukan jika user sudah melakukan import data, pemilian atribut dan pembersihan data
6	Asosiasi	U*ser* menginputkan nilai parameter yaitu nilai *Minimum Support* dan *Minimum Confidence*, kemudian sistem melakukan pencarian *frequent itemset*, melakukan proses eliminasi, menghasilkan *rule* dan informasi rekomendasi paket.
7	Input Parameter	*User* memasukkan nilai *Minimum Support* dan *Minimum Confidence* sebagai parameter dalam menentukan *rule.*
8	Lihat *rules*	Sistem menampilkan hasil dari seleksi *rule.*
9	Lihat Informasi Paket	Sistem menampilkan informasi berupa produk kosmetik apa saja yang bisa dipasangkan dalam bentuk paket.

Berikut merupakan gambar diagram *use case* dari sistem yang akan dibuat, yang diacu pada gambar 3.30.



Gambar 3.30. Diagram *Use Case*

3.1.5.1.1 Skenario *Use Case*

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.

Requirement A.1

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

Skenario Use Case Import Data

Use case Name	***Import Data***	
Related Requirements	Requirement *A.1*	
Goal In Context	*Import* data transaksi kartika kosmetik 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* memasukan data transaksi kartika kosmetik
	2	Sistem memvalidasi data transaksi kartika kosmetik
	3	Sistem menampilkan data transaksi kartika kosmetik
	4	*User* melakukan *request* kepada sistem untuk menyimpan data transaksi kedalam *database*

	5	Sistem menyimpan data transaksi kedalam *database*
Extension	Step	Branching Action
	2.1	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, dan Hitung Frekuensi.

Requirement A.2

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

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, dan hitung frekuensi
Precondition	Data transaksi sudah tersimpan didalam *database*
Successful End Condition	Sistem berhasil melakukan proses pemilihan atribut, pembersihan data, dan hitung frekuensi
Failed End Condition	Sistem gagal untuk melakukan proses pemilihan atribut, pembersihan data dan hitung frekuensi
Actors	*User*
Triger	*User* melakukan *request* proses pemilihan atribut, pembersihan data, dan hitung frekuensi
Included Cases	Pemilihan Atribut, Pembersihan Data, dan Hitung Frekuensi

Main Flow	Step	Action
	1	User melakakukan *request* Pemilihan Atribut, Pembersihan Data, dan Hitung Frekuensi kepada sistem
	2	Sistem melakukan proses Pemilihan Atribut
	3	Sistem menyimpan hasil proses Pemilihan Atribut kedalam *database*
	4	Sistem menampilkan hasil proses Pemilihan Atribut
	5	Sistem melakukan proses Pembersihan Data
	6	Sistem menyimpan hasil proses Pembersihan Data kedalam *database*
	7	Sistem menampilkan hasil proses Pembersihan Data
	8	Sistem melakukan proses Hitung Frekuensi tiap item pada data transaksi
	9	Sistem menyimpan hasil proses Hitung Frekuensi kedalam *database*
	10	Sistem menampilkan hasil proses Hitung Frekuensi
Extension	Step	Branching Action
	2.1	Sistem menampilkan pesan kesalahan dalam proses pemilihan atribut
	5.1	Sistem menampilkan pesan kesalahan dalam proses pembersihan data
	8.1	Sistem menampilkan pesan kesalahan dalam proses hitung frekuensi

3. Scenario *Use Case Pemilihan Atribut*

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

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 Nomor Faktur dan Nama Barang	
Precondition	Data transaksi tersimpan didalam *database*	
Successful End Condition	Sistem berhasil memilih atribut Nomor Faktur dan Nama Barang	
Failed End Condition	Sistem gagal melakukan Pemilihan Atribut Nomor Faktur dan Nama Barang	
Actors	*User*	
Triger	*User* melakukan *request* Pemilihan Atribut kepada system	
Included Cases	-	
Main Flow	Step	Action
	1	User melakukan *request* Pemilihan Atribut kepada sistem
	2	Sistem melakukan pemilihan atribut Nomor Faktur dan Nama Barang
	3	Sistem menyimpan hasil pemilihan atribut kedalam *database*
Extension	Step	Branching Action
	2.1	Sistem menampilkan pesan kesalahan saat proses pemilihan atribut

4. Skenario *Use Case Pembersihan Data*

Skenario *use case Pembersihan Data* menggambarkan langkah-langkah melakukan pembersihan terhadap Nomor Faktur dan Nama Barang yang memiliki *item* tunggal dan telah melewati proses Pemilihan Atribut.

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 Nomor Faktur dan Nama Barang yang memiliki item tunggal.	
Precondition	Data Transaksi sudah melalui proses Pemilihan Atribut	
Successful End Condition	Sistem berhasil melakukan pembersihan pada atribut Nomor Faktur dan Nama Barang yang memiliki item tunggal	
Failed End Condition	Sistem gagal dalam melakukan pembersihan pada atribut Nomor Faktur dan Nama Barang yang memiliki item tunggal	
Actors	*User*	
Triger	*User* melakukan *request* Pembersihan Data pada sistem	
Included Cases	-	
Main Flow	Step	Action
	1	User melakukan *request* pembersihan data kepada sistem
	2	Sistem melakukan pembersihan pada atribut Nama Faktur dan Nama Barang yang memiliki item tunggal.
	3	Sistem menyimpan hasil Pembersihan Data kedalam *database*

Extension	Step	Branching Action
	2.1	Sistem menampilkan pesan kesalahan saat 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.

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 sistem	
Included Cases	-	
Main Flow	Step	Action
	1	User melakukan *request* hitung frekuensi kepada sistem
	2	Sistem melakukan proses hitung frekuensi tiap item pada data transaksi
	3	Sistem menyimpan hasil Hitung Frekuensi kedalam *database*

Extension	Step	Branching Action
	2.1	Sistem menampilkan pesan kesalahan saat proses menghitung frekuensi

6. Skenario Use Case Proses Asosiasi

Skenario *use case*proses *assosiasi* menggambarkan langkah-langkah aksi *user* untuk melakukan proses Input Parameter, Lihat *Rules*, dan Lihat Informasi Paket

Requirement A.3

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

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 kemudian akan ditampilkan dalam bentuk informasi paket produk
Precondition	Data sudah melalui proses *preprocessing*
Successful End Condition	Sistem berhasil mendapatkan *rule* yang kemudian akan ditampilkan dalam bentuk informasi paket produk
Failed End Condition	Sistem gagal mendapatkan *rule* yang akan ditampilkan dalam bentuk informasi paket produk
Actors	*User*
Triger	*User* menginputkan nilai parameter berupa nilai *Minimum Support* dan *Minimum Confidence*

Included Cases	Input Parameter, Lihat *Rules*, Lihat Informasi Paket	
Main Flow	Step	Action
	1	*User* menginputkan nilai *Minimum Support*
	2	*User* menginputkan nilai *Minimum Confidence*
	3	Sistem melakukan proses pencarian *frequent itemset* sesuai nilai *Minimum Support* dengan *Algoritma CT-Pro*
	4	Sistem mendapatkan *frequent itemset*
	5	Sistem melakukan *generate rule*
	6	Sistem melakukan Eliminasi terhadap item yang kurang dari nilai *Minimum Confidence*
	7	Sistem mendapatkan hasil *rule* yang sesuai dengan *minimum confidence*
	8	Sistem menampilkan hasil *rule* dan menampilkan hasil informasi paket produk
Extension	Step	Branching Action
	4.1	Sistem tidak mendapatkan *frequent itemset* karena nilai *Minimum Support* yang dimasukkan oleh *user* terlalu tinggi
	5.1	Sistem gagal melakukan *generate rule*, karna tidak ada *frequent itemset* yang ditemukan
	7.1	Sistem tidak mendapatkan *rule* karena nilai *Minimum Confidence* yang dimasukkan terlalu tinggi
	8.1	Sistem gagal menampilkan hasil dari *generate rule* karena nilai *support* / *confidence*nya terlalu tinggi

7. Scenario *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*.

Skenario Use Case Input Parameter

Use case Name	**Input Parameter**	
Related Requirements	Requirement *A.1*, Requirement *A.2*, 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* menginputkan nilai *Minimum Support*
	2	*User* menginputkan nilai *Minimum Confidence*
	3	Sistem mendapatkan nilai *Minimum Support* dan nilai *Minimum Confidence*
Extension	Step	Branching Action
	-	-

8. Skenario *Use Case* Lihat Rules

Skenario *use case Lihat Rules* menggambarkan langkah-langkah aksi *user* untuk melakukan proses lihat rules.

Skenario Use Case Lihat Rules

Use case Name	**Lihat Rules**	
Related Requirements	Requirement *A.1*, Requirement *A.2*, Requirement *A.3*	
Goal In Context	Sistem menampilkan hasil dari pencarian *frequent itemset* berupa pembentukan *rule*	
Precondition	User sudah memasukkan nilai *Minimum Support* dan *Minimum Confidence*	
Successful End Condition	Sistem berhasil menampilkan hasil dari pencarian *frequent itemset* berupa pembentukan *rule*	
Failed End Condition	Sistem gagal hasil dari pencarian *frequent itemset* berupa pembentukan *rule*	
Actors	*User*	
Triger	*User* memasukkan nilai *Minimum Support* dan *Minimum Confidence*	
Main Flow	Step	Action
	1	*User* memasukkan nilai *Minimum Support*
	2	*User* memasukkan nilai *Minimum Confidence*
	3	Sistem mendapatkan nilai *Minimum Support*
	4	Sistem mendapatkan nilai *Minimum Confidence*
	5	Sistem melakukan pencarian *frequent itemset* sesuai dengan nilai *Minimum Support* dengan *Algoritma CT-Pro*
	6	Sistem mendapatkan *frequent itemset*
	7	Sistem melakukan *generate rule*
	8	Sistem mengeliminasi *rule* yang dihasilkan sesuai dengan nilai *Minimum Confidence*

	9	Sistem mendapatkan *rule*
	10	Sistem menampilkan informasi tentang *rule* yang telah didapatkan
Extension	Step	Branching Action
	3.1	Sistem gagal mendapatkan nilai *Minimum Support*
	4.1	Sistem gagal mendapatkan nilai *Minimum Confidence*
	6.1	Sistem gagal mendapatkan *frequent itemset* karena nilai *Minimum Support* yang dimasukkan oleh *user* terlalu tinggi
	7.1	Sistem gagal melakukan *generate rule* karena tidak mendapatkan *frequent itemset*
	9.1	Sistem gagal mendapatkan *rule* karena nilai *Minimum Confidence* yang dimasukkan oleh user terlalu tinggi
	10.1	Sistem tidak dapat menampilkan hasil dari *generate rule* karena nilai *Minimum Support* atau *Minimum Confidence* terlalu tinggi

9. Skenario *Use Case* Lihat Informasi Pembentukan Paket

Skenario *use case* Lihat Informasi Pembentukan Paket menggambarkan langkah-langkah aksi *user* untuk melakukan proses lihat informasi pembentukan paket.

Skenario Use Case Lihat Informasi Pembentukan Paket

Use case Name	**Lihat Rules**
Related Requirements	Requirement *A.1*, Requirement *A.2*, Requirement *A.3*
Goal In Context	Sistem menampilkan informasi paket produk hasil pembentukan *rule*
Precondition	User sudah melakukan memasukkan nilai *Minimum*

	Support dan *Minimum Confidence*	
Successful End Condition	Sistem berhasil menampilkan informasi paket produk hasil pembentukan *rule*	
Failed End Condition	Sistem gagal menampilkan informasi paket produk hasil pembentukan *rule*	
Actors	*User*	
Triger	*User* memasukkan nilai *Minimum Support* dan *Minimum Confidence*	
Main Flow	Step	Action
	1	*User* memasukkan nilai *Minimum Support*
	2	*User* memasukkan nilai *Minimum Confidence*
	3	Sistem mendapatkan nilai *Minimum Support*
	4	Sistem mendapatkan nilai *Minimum Confidence*
	5	Sistem melakukan pencarian *frequent itemset* sesuai dengan nilai *Minimum Support* dengan *Algoritma CT-Pro*
	6	Sistem mendapatkan *frequent itemset*
	7	Sistem melakukan *generate rule*
	8	Sistem mengeliminasi *rule* yang dihasilkan sesuai dengan nilai *Minimum Confidence*
	9	Sistem mendapatkan *rule*
	10	Sistem menampilkan informasi tentang *rule* yang telah didapatkan
	11	Sistem menampilkan produk apa saja yang bisa dijual dalam bentuk paket
Extension	Step	Branching Action
	3.1	Sistem gagal mendapatkan nilai *Minimum Support*
	4.1	Sistem gagal mendapatkan nilai *Minimum*

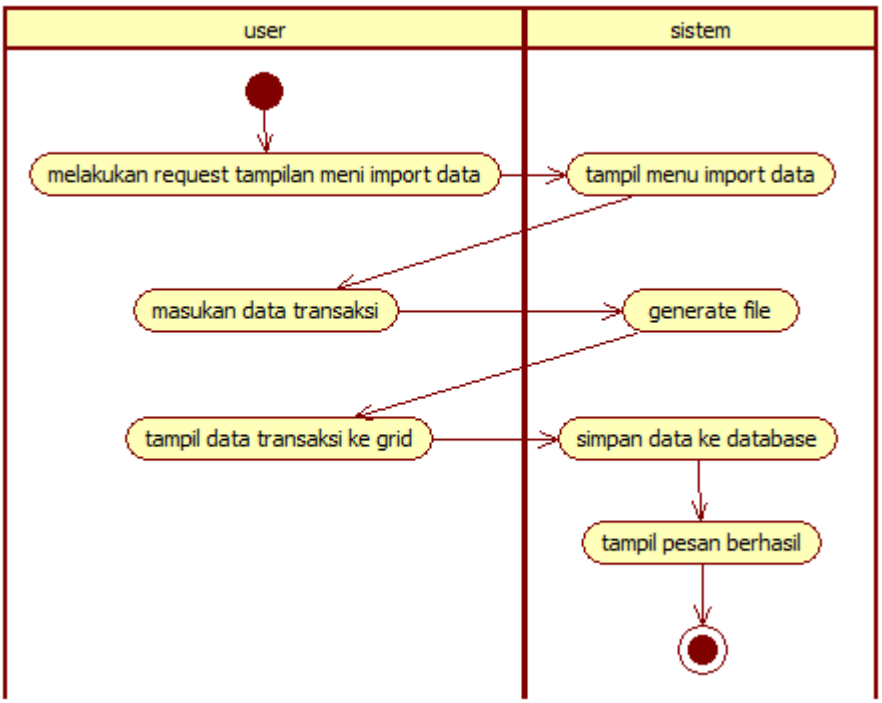
		Confidence
	6.1	Sistem gagal mendapatkan *frequent itemset* karena nilai *Minimum Support* yang dimasukkan oleh *user* terlalu tinggi
	7.1	Sistem gagal melakukan *generate rule* karena tidak mendapatkan *frequent itemset*
	9.1	Sistem gagal mendapatkan *rule* karena nilai *Minimum Confidence* yang dimasukkan oleh user terlalu tinggi
	10.1	Sistem tidak dapat menampilkan hasil dari *generate rule* karena nilai *Minimum Support* atau *Minimum Confidence* terlalu tinggi
	11.1	Sistem tidak memiliki informasi produk yang bisa dijual dalam bentuk paket

3.1.5.2 Diagram Activity

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

1. *Activity Diagram* Import Data

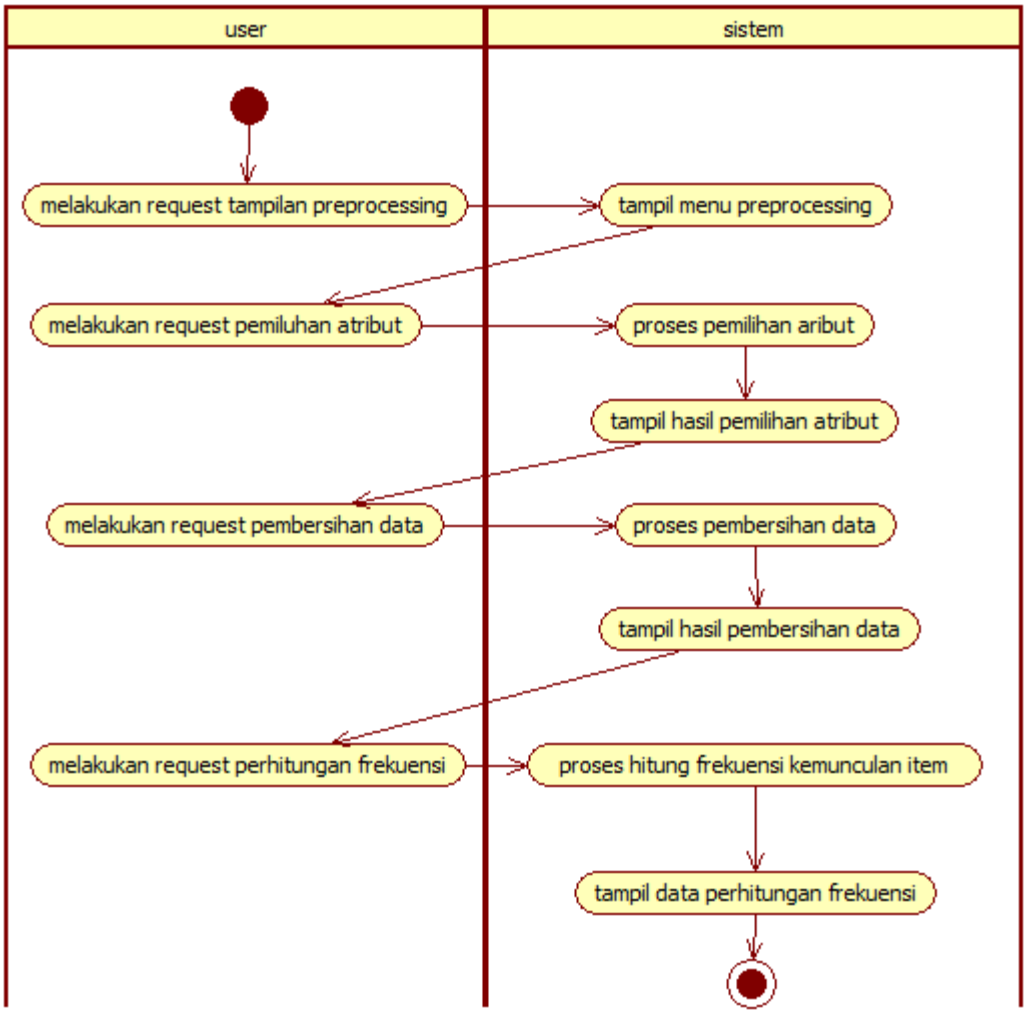
   Activity diagram import data menjelaskan aktivitas aktor dalam melakukan proses import data. Penjelasan activity *diagram* import data diacu pada Gambar 3.31.



Gambar 3.31. *Activity Diagram* Import Data

2. *Activity Diagram* Preprocesing

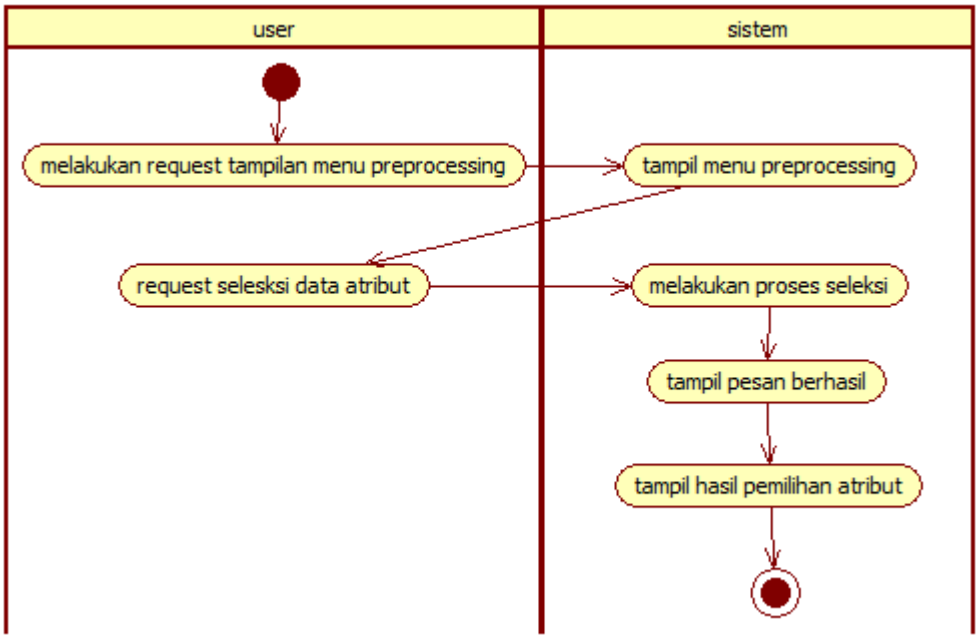
Activity diagram preprocessing menjelaskan aktivitas sistem dalam melakukan proses Pemilihan Atribut dan Pembersihan Data. *Activity diagram* preprocessing dapat dilihat pada gambar 3.32.



Gambar 3.32. *Activity Diagram* Preprocessing

3. *Activity Diagram* Pemilihan Atribut

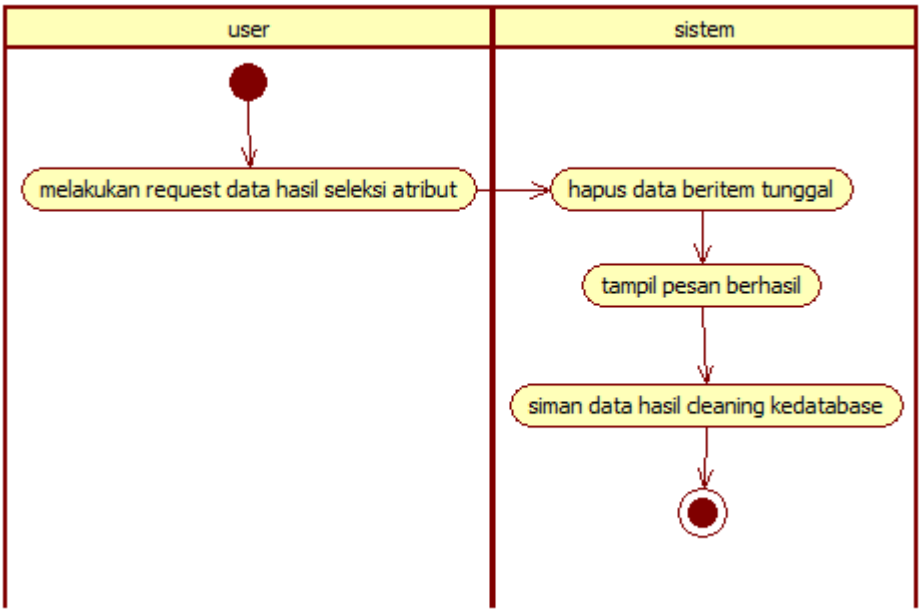
Activity diagram Pemilihan Atribut menjelaskan aktivitas sistem dalam melakukan proses data Pemilihan Atribut . *Activity diagram* data Pemilihan Atribut dapat dilihat pada gambar 3.33.



Gambar 3.33. *Activity Diagram* Pemilihan Atribut

4. *Activity Diagram* Pembersihan Data

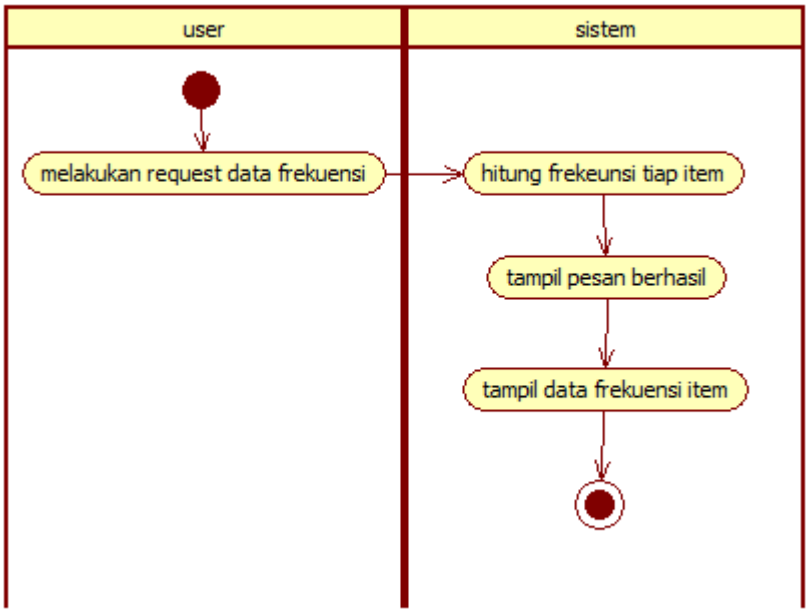
Activity diagram data Pembersihan Data menjelaskan aktivitas sistem dalam melakukan proses data Pembersihan Data. *Activity diagram* data Pembersihan Data dapat dilihat pada gambar 3.34.



Gambar 3.34. *Activity Diagram* Pembersihan Data

5. *Activity Diagram* Hitung Frekuensi

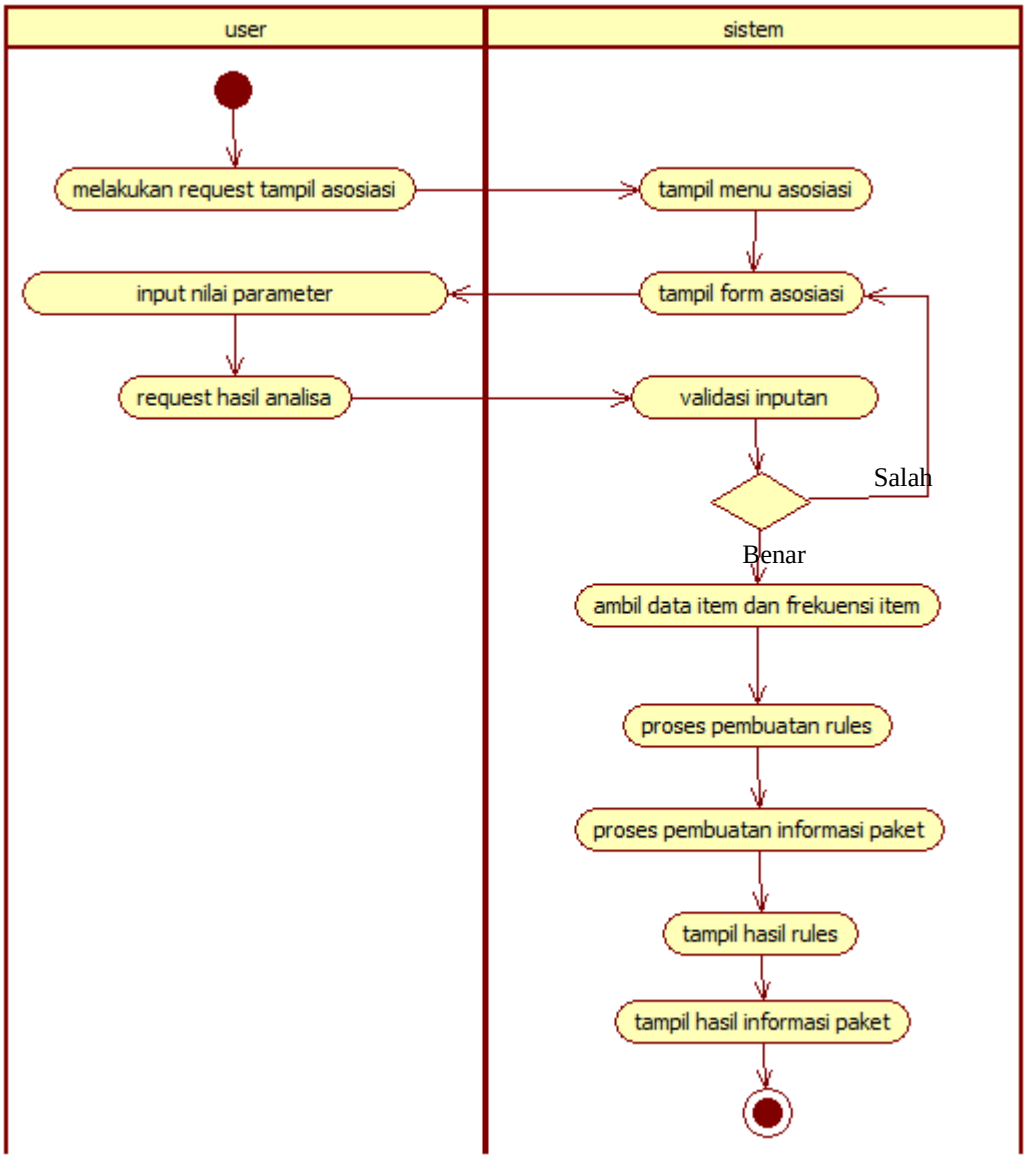
Activity diagram Hitung Frekuensi menjelaskan aktivitas user dan sistem dalam melakukan proses hitung frekuensi. *Activity diagram* Hitung Frekuensi dapat dilihat pada gambar 3.35.



Gambar 3.35. *Activity Diagram* Hitung Frekuensi

5. *Activity Diagram* Asosiasi

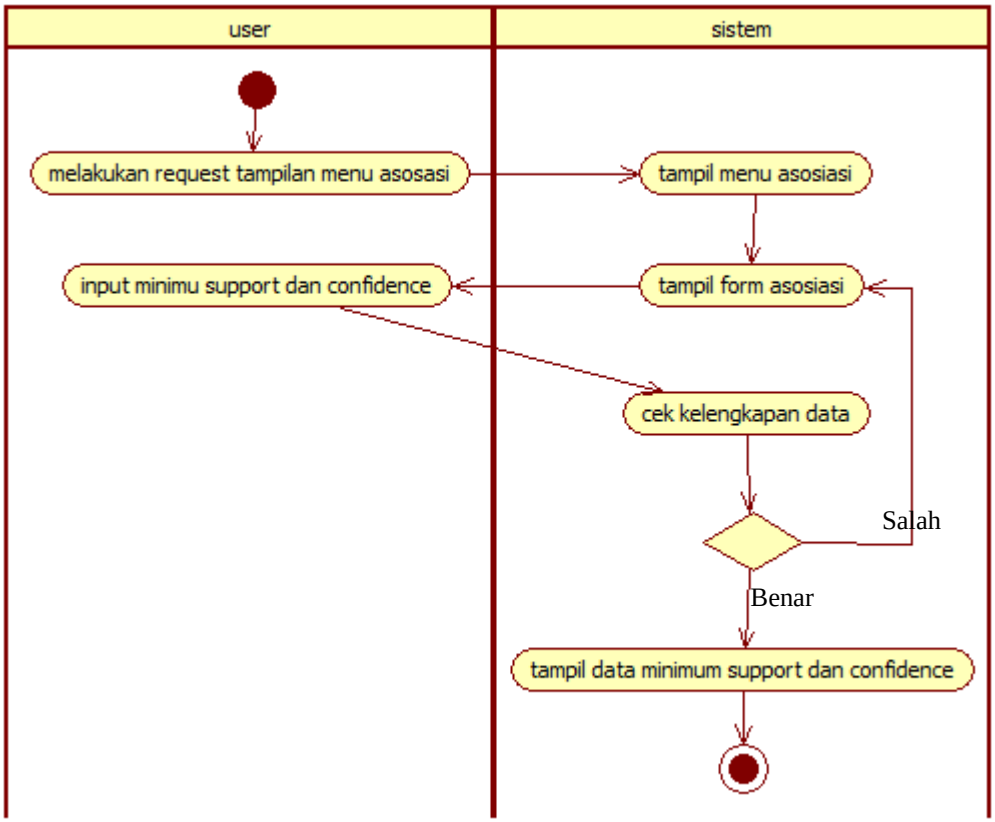
Activity diagram asosiasi menjelaskan aktivitas user dan sistem dalam melakukan proses input parameter dan melihat rule. *Activity diagram* Asosiasi dapat dilihat pada gambar 3.36.



Gambar 3.36. *Activity Diagram* Asosiasi

6. *Activity Diagram* Input Parameter

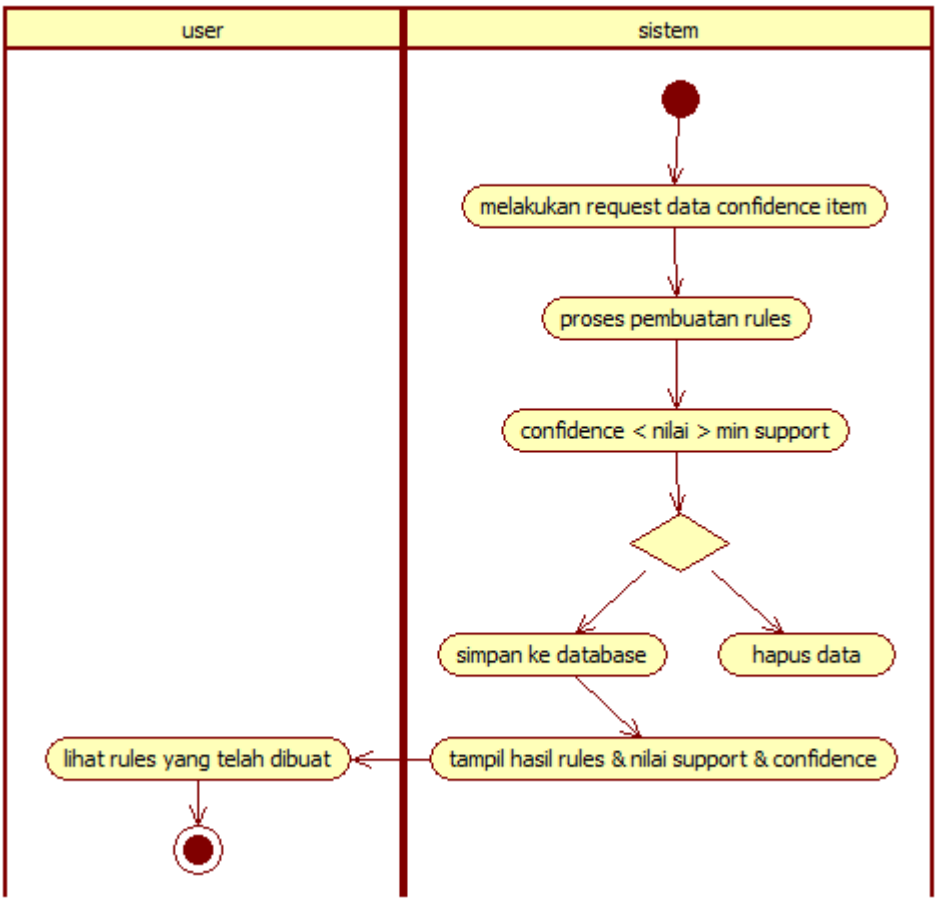
Activity diagram input parameter menjelaskan aktivitas *user* dan sistem dalam melakukan proses memasukkan nilai *Minimum Support* dan *Minimum Confidence*. *Activity diagram* Input Parameter dapat dilihat pada gambar 3.37.



Gambar 3.37. *Activity Diagram* Input Parameter

7. *Activity Diagram* Lihat Hasil *Rules*

Activity diagram lihat hasil *rule* 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 3.38.



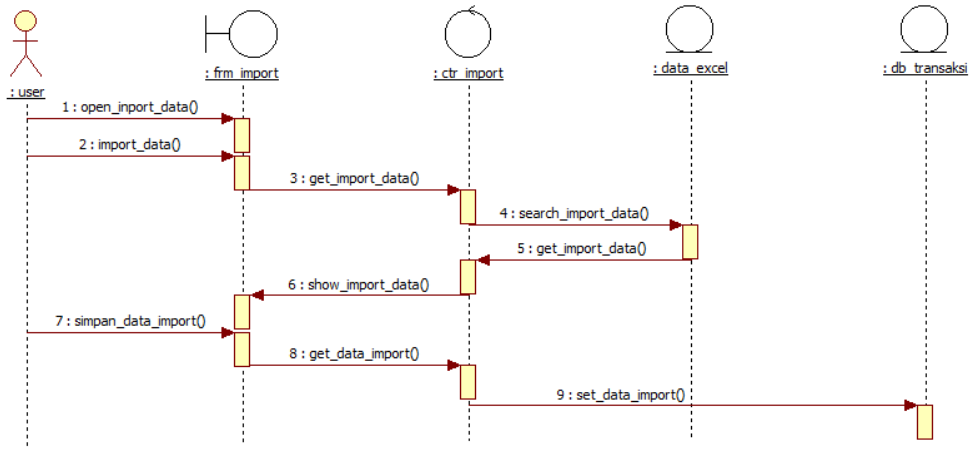
Gambar 3.38. *Activity Diagram* Lihat Hasil Rules

3.1.5.3 Diagram Sequence

Diagram sequensial menggambarkan interaksi objek pada *use case* dengan mendeksripsikan waktu hidup objek dan pesan yang dikirimkan dan di terima antar objek. Berikut adalah diagram squensial dari sistem yang akan dibangun :

1. *Sequence Diagram* Import Data

   Untuk *sequence diagram* pada proses import data dapat dilihat pada gambar 3.39.

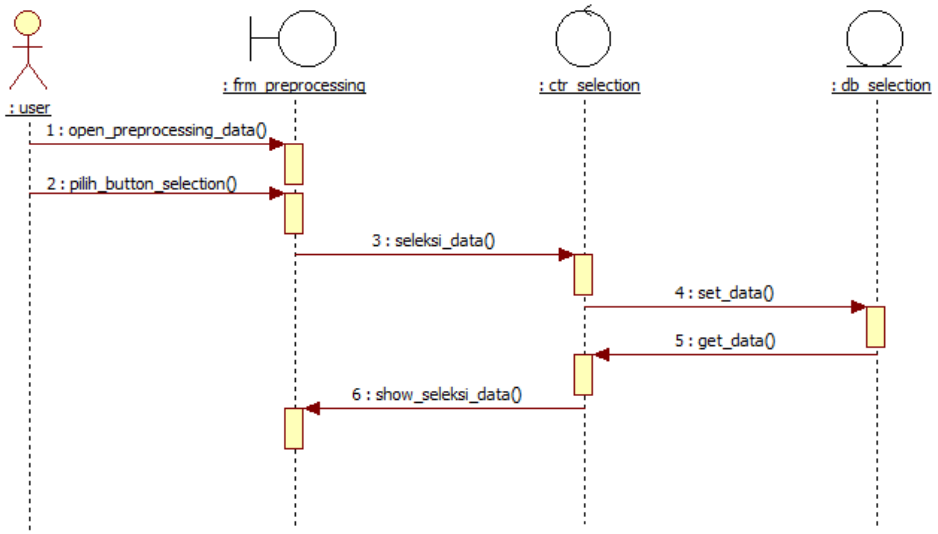


Gambar 3.39. *Sequence Diagram* Import Data

2. *Sequence Diagram Preprocessing* :

a. *Sequence Diagram* Proses Seleksi Data

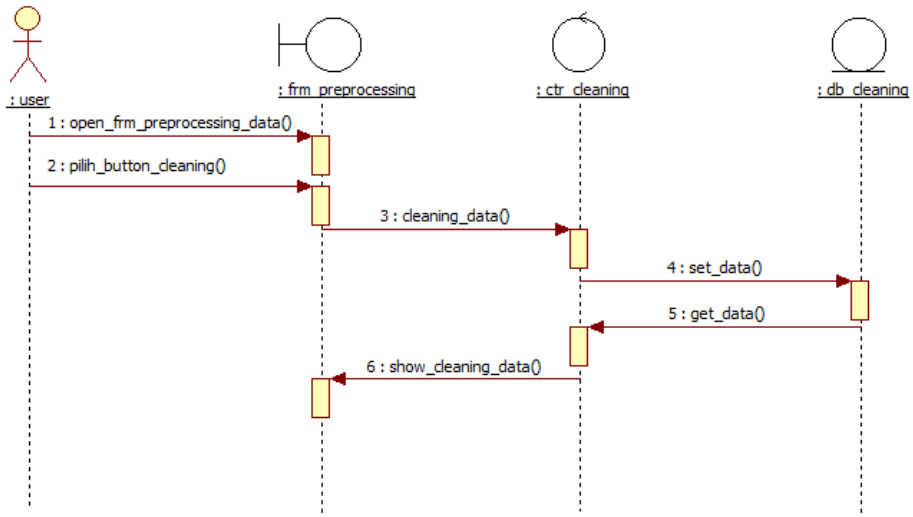
   Untuk *sequence diagram* pada proses seleksi data dapat dilihat pada gambar 3.40.



Gambar 3.40. *Sequence Diagram* Seleksi Data

b. *Sequence Diagram* Proses Pembersihan Data

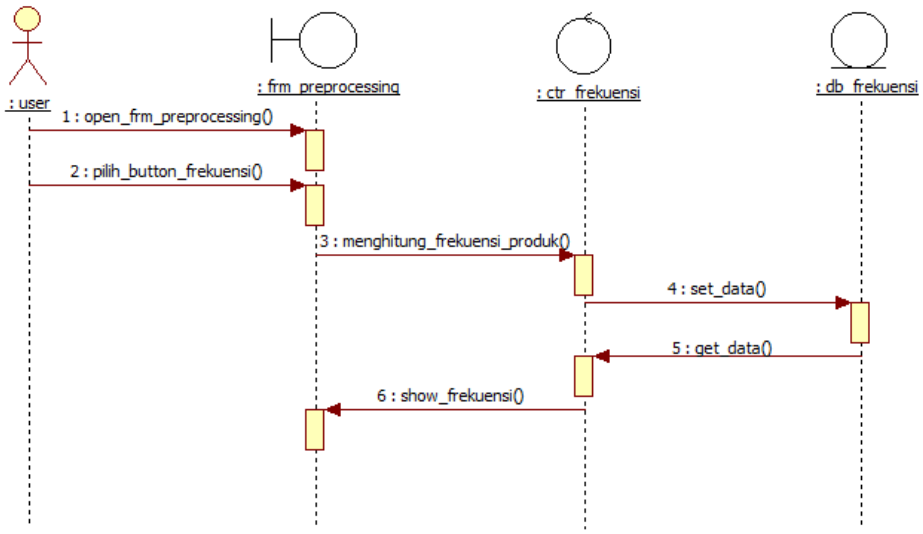
Untuk *sequence diagram* pada proses pembersihan data dapat dilihat pada gambar 3.41.



Gambar 3.41. *Sequence Diagram* Pembersihan Data

c. *Sequence Diagram* Proses Hitung Frekuensi

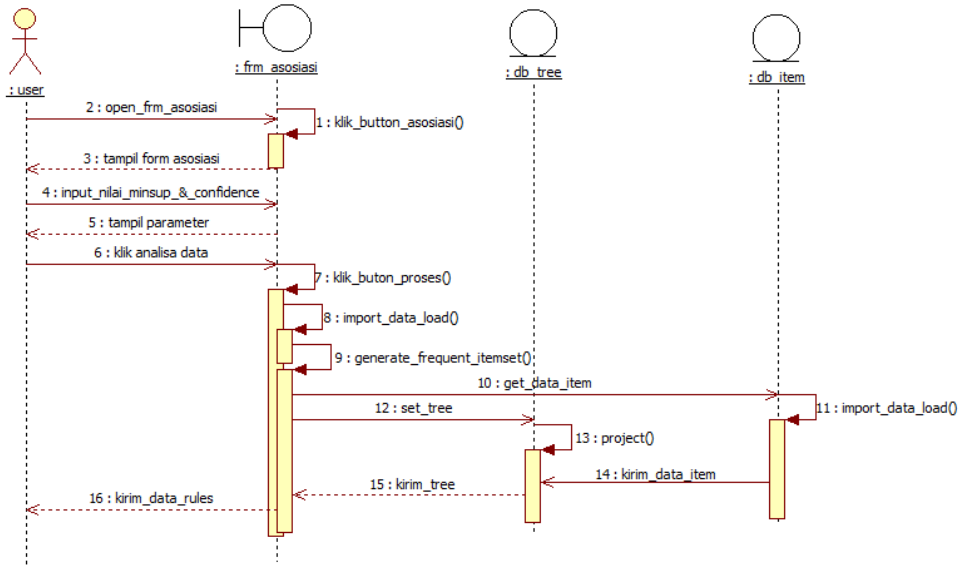
Untuk *sequence diagram* pada proses kemunculan produk dapat dilihat pada gambar 3.42.



Gambar 3.42. *Sequence Diagram* Menghitung Frekuensi

3. *Sequence Diagram* Proses Asosiasi

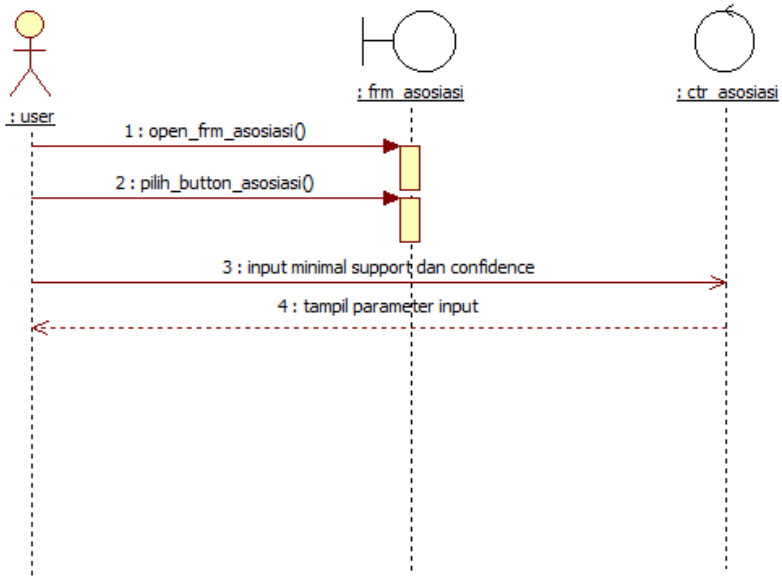
Untuk *sequence diagram* pada proses asoasiasi dapat dilihat pada gambar 3.43.



Gambar 3.43. *Sequence Diagram* Proses Asosiasi

4. *Sequence Diagram Input Parameter*

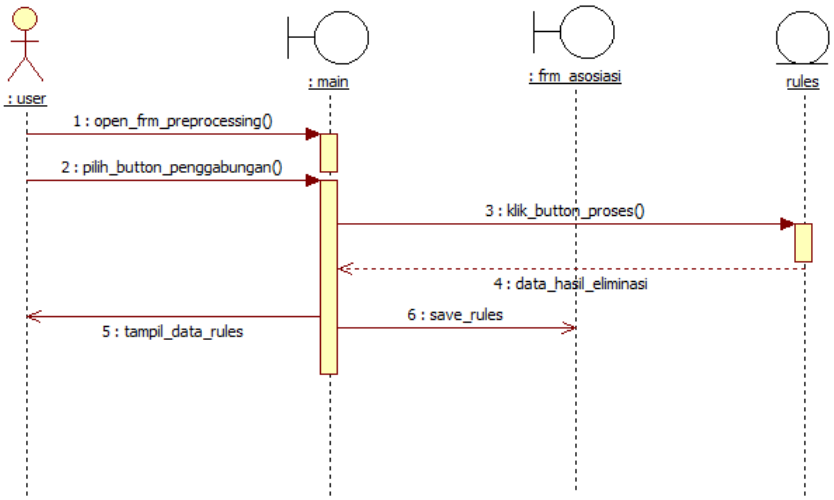
   Untuk *sequence diagram* pada proses input parameter dapat dilihat pada gambar 3.44.



Gambar 3.44. *Sequence Diagram* Input Parameter

5. *Sequence Diagram* Lihat *Rules*

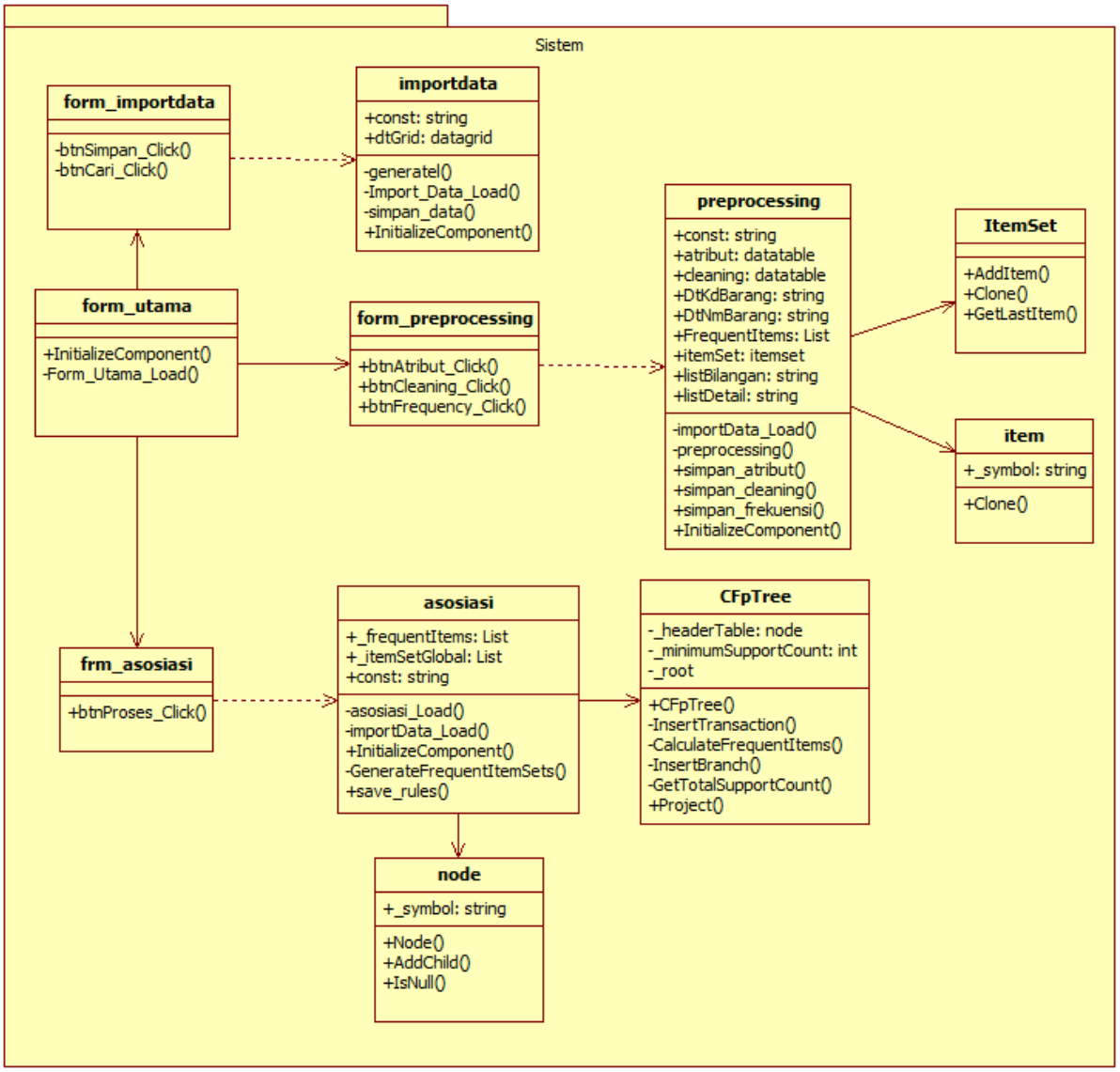
   Untuk *sequence diagram* pada proses lihat rules dapat dilihat pada gambar 3.45.



Gambar 3.45. *Sequence Diagram* Proses Lihat Rules

3.1.5.4 Diagram Class

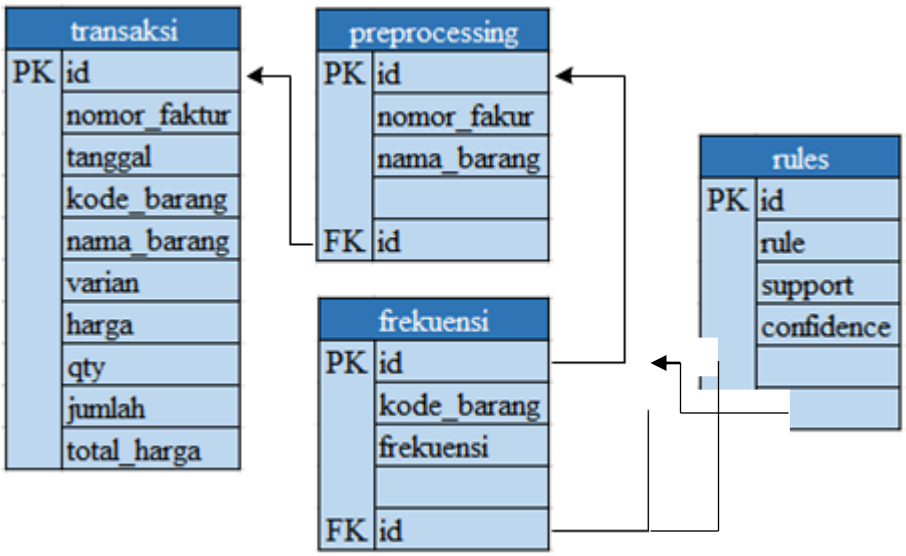
Diagram kelas atau *class diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan di buat untuk membangun sistem. Diagram Kelas dalam aplikasi yang akan dibangun ini digambarkan pada gambar 3.46.



Gambar 3.46. ***Class Diagram***

3.1.5.5 Skema Relasi

Berikut adalah skema relasi dari system yang akan dibangun, yang diacu pada gambar 3.47.



Gambar 3.47. *Skema Relasi*

3.2 Perancangan Sistem

Tahap perancangan sistem merupakan tahap untuk melakukan perencanaan rancangan terhadap sistem yang akan dibangun, mulai dari perancangan basis data, perancangan antarmuka perangkat lunak, perancangan struktur menu, perancangan pesan, jaringan semantik dan perancangan prosedural. Dimana semua perancangan yang dibuat tersebut harus saling berhubungan satu sama lain agar dapat menghasil suatu sistem yang baik.

3.2.1 Perancangan Kelas

Berikut adalah perancangan kelas yang ada di *Class Diagram* :

Tabel 3.18. Struktur Kelas *form_utama*

Class : form_utama		
Nama	Jenis	Type
InitializeComponent()	Method	Public
Form_Utama_Load()	Method	Private

Tabel 3.19. Struktur Kelas *form_importdata*

Class : form_importdata		
Nama	Jenis	Type
btnCari_Click()	Method	Private
btnSimpan_Click()	Method	Private

Tabel 3.20. Struktur Kelas *importdata*

Class : importdata		
Nama	Jenis	Type
const: string	Atribut	Public
dtGrid: datagrid	Atribut	Public
generate ()	Method	Private
importData_Load()	Method	Private
simpandata()	Method	Private

InitializeComponent()	Method	Public

Tabel 3.21. Struktur Kelas *form_preprocessing*

Class : form_preprocessing		
Nama	Jenis	Type
btnAtribut_Click()	Method	Public
btnCleaning_Click()	Method	Public
btnFrequency_Click()	Method	Public

Tabel 3.22. Struktur Kelas *preprocessing*

Class : preprocessing		
Nama	Jenis	Type
conns: string	Atribut	Public
atribut: datatable	Atribut	Public
cleaning: datatable	Atribut	Public
DtKdFaktur: string	Atribut	Public
DtNmBarang: string	Atribut	Public
FrequentItems: List	Atribut	Public
itemSet: itemset	Atribut	Public
listBilangan: string	Atribut	Public
listDetail: string	Atribut	Public
importData_Load()	Method	Private
preprocessing()	Method	Private
simpan_atribut()	Method	Public
simpan_cleaning()	Method	Public
simpan _frekuensi()	Method	Public
InitializeComponent()	Method	Public

Tabel 3.23. Struktur Kelas *ItemSet*

Class : ItemSet		
Nama	Jenis	Type
AddItem()	Method	Public
Clone()	Method	Public
GetLastItem()	Method	Public

Tabel 3.24. Struktur Kelas *Item*

Class : Item		
Nama	Jenis	Type
_symbol: string	Atribut	Public
Clone()	Method	Public

Tabel 3.25. Struktur Kelas *frm_asosiasi*

Class : frm_asosiasi		
Nama	Jenis	Type
btnProses_Click()	Method	Public

Tabel 3.26. Struktur Kelas *asosiasi*

Class : asosiasi		
Nama	Jenis	Type
_frequentItems: List	Atribut	Public
_itemSetGlobal: List	Atribut	Public
ConnString: string	Atribut	Public
asosiasi_Load()	Method	Private
importData_Load()	Method	Private
InitializeComponent()	Method	Public
GenerateFrequentItemSets()	Method	Private
save_rules()	Method	Public

Tabel 3.27. Struktur Kelas *node*

Class : node		
Nama	Jenis	Type
_symbol: string	Atribut	Public
AddChild()	Method	Public
IsNull()	Method	Public
Node()	Method	Public

Tabel 3.28. Struktur Kelas *CFPTree*

Class : CFpTree		
Nama	Jenis	Type
_headerTable: node	Atribut	Private
_minimumSupportCount: int	Atribut	Private
_root	Atribut	Private
CFpTree()	Atribut	Public
InsertTransaction()	Method	Private

CalculateFrequentItems()	Method	Private
InsertBranch()	Method	Private
GetTotalSupportCount()	Method	Private
Project()	Method	Public

3.2.2 Perancangan Data

3.2.2.1 Perancangan Basis Data

Perancangan Basis Data adalah menciptakan atau merancang kumpulan data yang terhubung dan tersimpan secara bersama-sama. Tabel yang akan digunakan dalam program ini antara lain:

1. Data Transaksi
2. Data Preprocessing
3. Data Frekuensi
4. Data Rules

3.2.2.2 Struktur Tabel

Struktur tabel merupakan gambaran *detail* tabel yang berisi *field*, tipe data, panjang data, dan keterangan lainnya. Adapun tabel-tabel yang digunakan dalam perangkat lunak yang dibangun yang diacu dari tabel 3.15 sampai tabel 3.20.

1. Tabel transaksi

   Tabel transaksi merupakan tabel yang digunakan untuk menyimpan data transaksi hasil import dari file excel.

Tabel 3.29. Data Awal

Nama Field	Tipe Data	Panjang	Keterangan
id	Int	11	Primari Key
nomor_faktur	Varchar	20	
tanggal	Varchar	20	
kode_barang	Varchar	20	
nama_barang	Varchar	50	
varian	Varchar	20	
harga	Int	11	

qty	Int	11	
jumlah	Int	11	
total_harga	Int	11	

2. Tabel preprocessing

Tabel selection adalah tabel yang digunakan untuk menyimpan data hasil *selection data*.

Tabel 3.30. Data Preprocessing

Nama Field	Tipe Data	Panjang	Keterangan
id	Int	11	Primari Key
nomor_fakur	Varchar	20	
nama_barang	Varchar	50	
id	Int	11	Foreign Key dari tabel transaksi

3. Tabel frekuensi

Tabel frekuensi adalah tabel yang digunakan untuk menyimpan data hasil penghitungan kemunculuan banyaknya masing-masing jenis produk.

Tabel 3.31. Data Frekuensi

Nama Field	Tipe Data	Panjang	Keterangan
id	Int	11	Primari Key
kode_barang	Varchar	20	
frekuensi	int	5	
id	Int	11	Foreign Key dari tabel preprocessing

4. Tabel rules

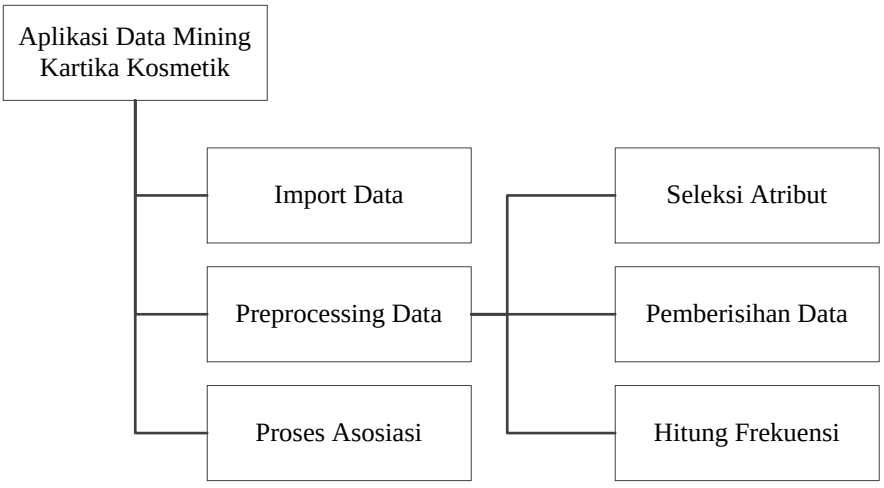
Tabel rules adalah tabel yang digunakan untuk menyimpan hasil pencarian *Frequent Itemset.*

Tabel 3.32. Data Hasil Pembentukan Rules

Nama Field	Tipe Data	Panjang	Keterangan
id	Int	11	Primari Key
rule	Text		
support	Int	10	
confidence	Float		
id	Int	11	Foreign Key dari tabel frekuensi

3.2.2 Perancangan Struktur Menu

Dalam pembangunan perangkat lunak diperlukan sebuah struktur menu untuk menunjukkan keterkaitan dan tingkatan antara menu utama dengan sub menu, sehingga lebih mudah dipahami. Berikut adalah struktur menu dari perangkat lunak yang akan dibangun yang diacu pada gambar 3.48.



Gambar 3.48. Struktur Menu

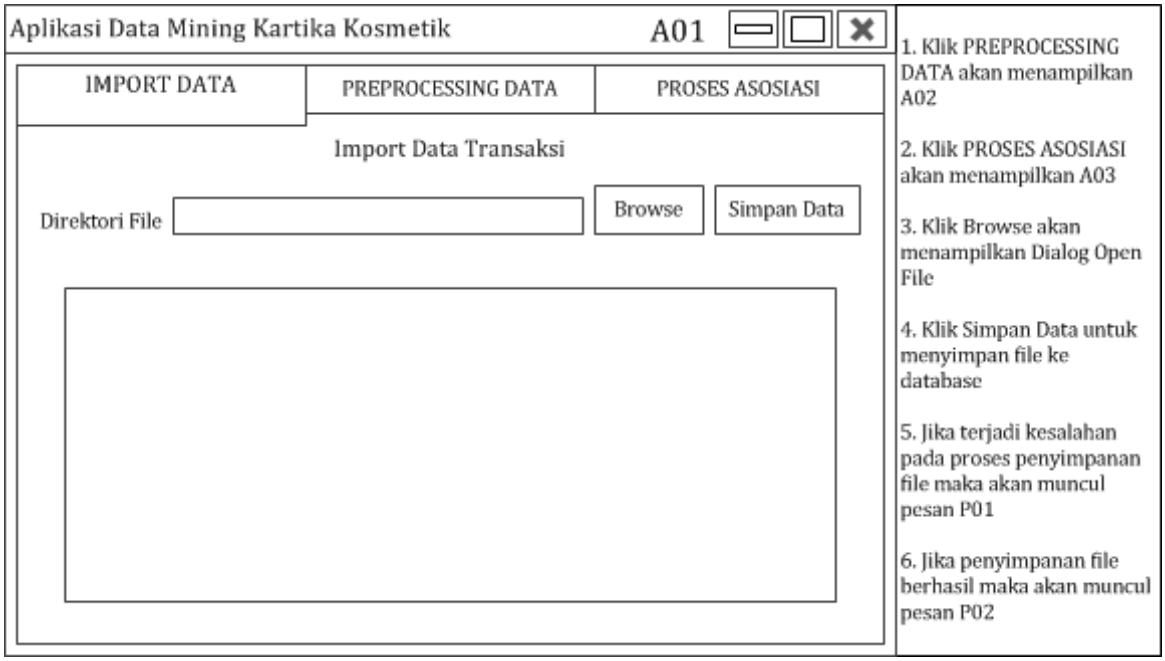
3.2.3 Perancangan Antar Muka

Perancangan antarmuka atau *interface* merupakan tampilan dari sebuah program atau aplikasi yang berperan sebagai media komunikasi yang digunakan sebagai sarana dialog antara program dengan *user*. Sistem yang

akan dibangun diharapkan menyediakan *interface* yang sesuai dengan kebutuhan dan karakteristik *user*. Perancangan *interface* untuk aplikasi Data *Mining* Kartika Kosmetik adalah sebagai berikut:

1. Perancangan Antar Muka Import Data

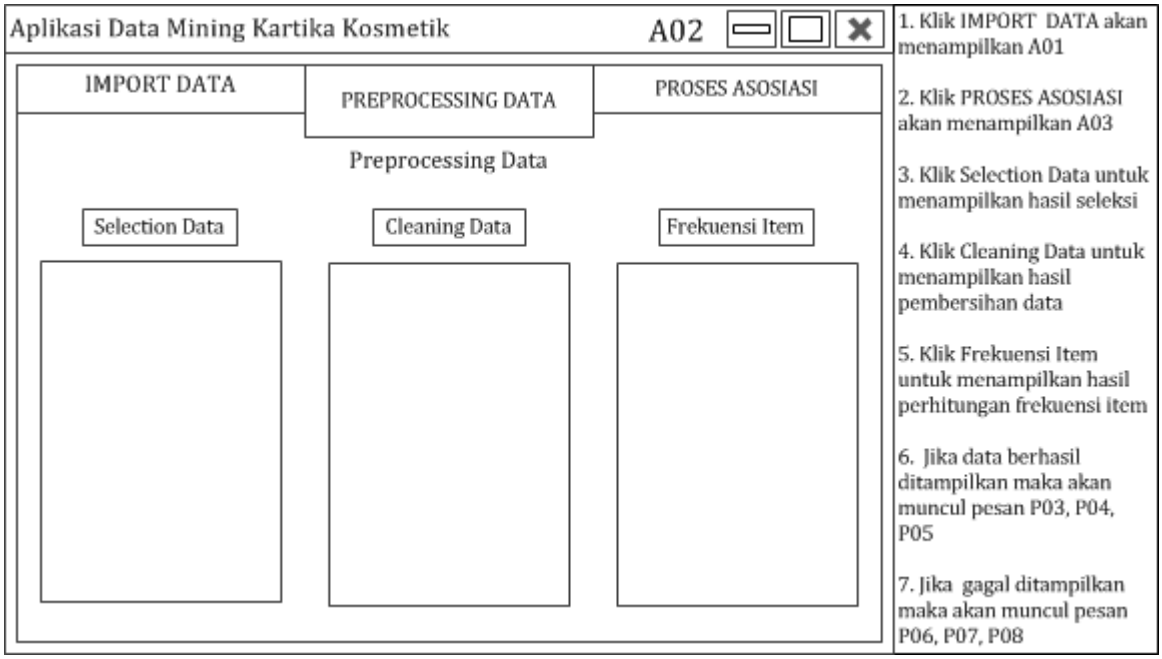
   Perancangan Antar Muka Import Data dapat dilihat pada gambar 3.49.



Gambar 3.49. Antar Muka Import Data

2. Perancangan Antar Muka Preprocessing Data

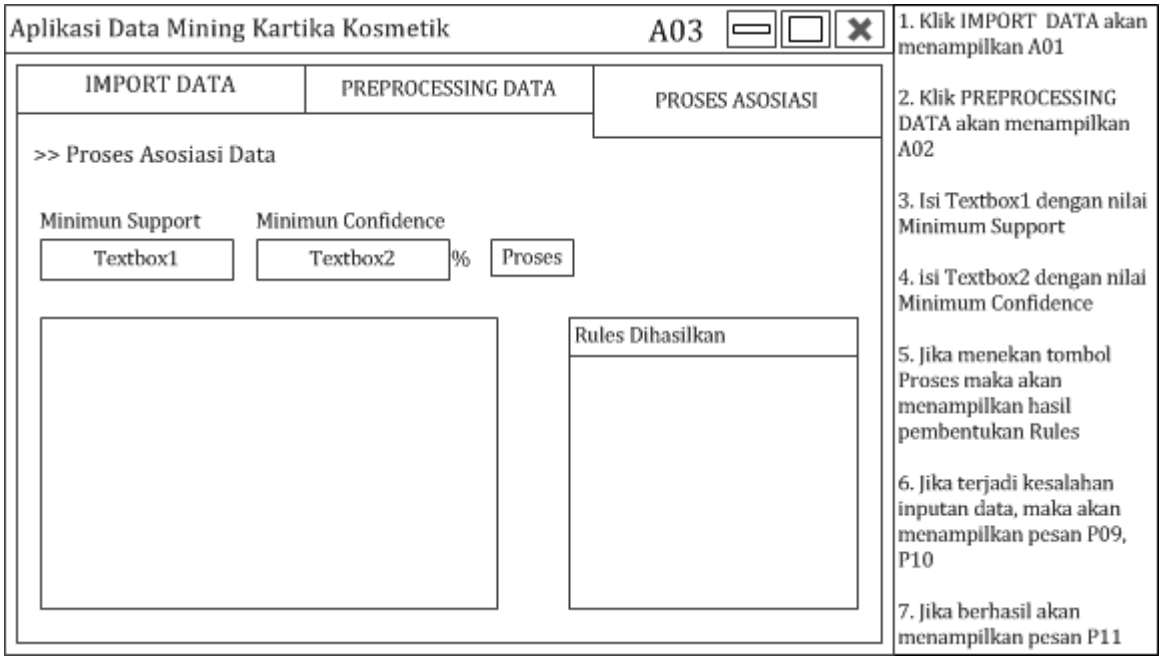
   Perancangan Antar Muka Preprocessing Data dapat dilihat pada gambar 3.50.



Gambar 3.50. Antar Muka Preprocessing Data

3. Perancangan Antar Muka Proses Asosiasi

Perancangan Antar Muka Proses Asosiasi dapat dilihat pada gambar 3.51



Gambar 3.51. Antar Muka Proses Asosiasi

3.2.4 Perancangan Pesan

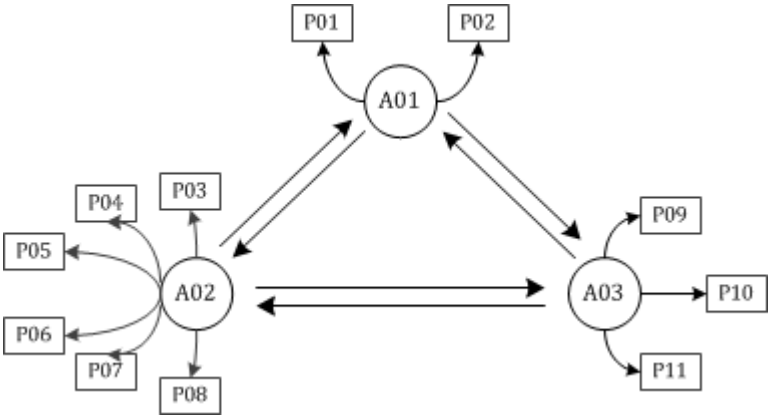
Adapun perancangan pesan dapat dilihat pada gambar 3.52 berikut ini.



Gambar 3.52 Perancangan Pesan

3.2.5 Perancangan Semantik

Perancangan Semantik dapat dilihat pada gambar 3.53.



Gambar 3.53. Perancangan Semantik