BAB 3
PEMBAHASAN

3.1 Analisis Sistem

Analisis merupakan suatu tahapan pemahaman terhadap sistem atau aplikasi yang sedang berjalan maupun yang akan dibuat. Tahapan analisis bertujuan untuk mengetahui mekanisme atau prosedur kerja dari proses yang sedang berjalan maupun yang akan dibuat.

3.1.1 Analisis Masalah

Analisis masalah adalah pengidentifikasian terhadap kendala yang terjadi dalam suatu penelitian yang sedang dilakukan. Masalah sendiri dapat diartikan sebagai sesuatu yang memerlukan suatu jalan keluar atau penyelesaian. Berdasarkan hasil pengamatan, Primer Koperasi Kartika Baja memiliki permasalahan dalam penentuan paket hemat sembako dan kebutuhan harian yang memiliki harga yang menarik minat pembeli.

3.1.2 Business Understanding

Pemahaman bisnis ini merupakan tahap pertama dalam CRISP-DM. Di dalamnya terdapat beberapa tahap lagi yang meliputi:

1. Penentuan tujuan bisnis

   Primer Koperasi Kartika Baja dalam proses bisnisnya memiliki tujuan bisnis yaitu dapat mensejahterakan dan memenuhi setiap kebutuhan para pembelinya.
2. Menilai situasi

   Pada tahapan ini, dapat dilihat apakah situasi di Primer Koperasi Kartika Baja memungkinkan untuk menentukan paket sembako dan kebutuhan harian menggunakan aplikasi yang telah dibuat. Berikut ini adalah spesifikasi perangkat keras yang ada di Primkop Kartika Baja, dapat dilihat pada tabel 3.1:

Tabel 3.1 Spesifikasi Perangkat Keras

No	Perangkat Keras	Spesifikasi
1	Monitor	LCD 14”
2	Processor	Intel Dual Core 2.93 GHz
3	RAM	1 GB
4	Harddisk	250 GB
5	VGA	512 MB

3. Penentuan sasaran *data mining*

   Tujuan dari penetapan *data mining* dalam penentuan paket sembako dan kebutuhan harian adalah membantu KOMURNIKKOP dalam menentukan paket yang dapat menarik minat pembeli.

3.1.3 Data Understanding

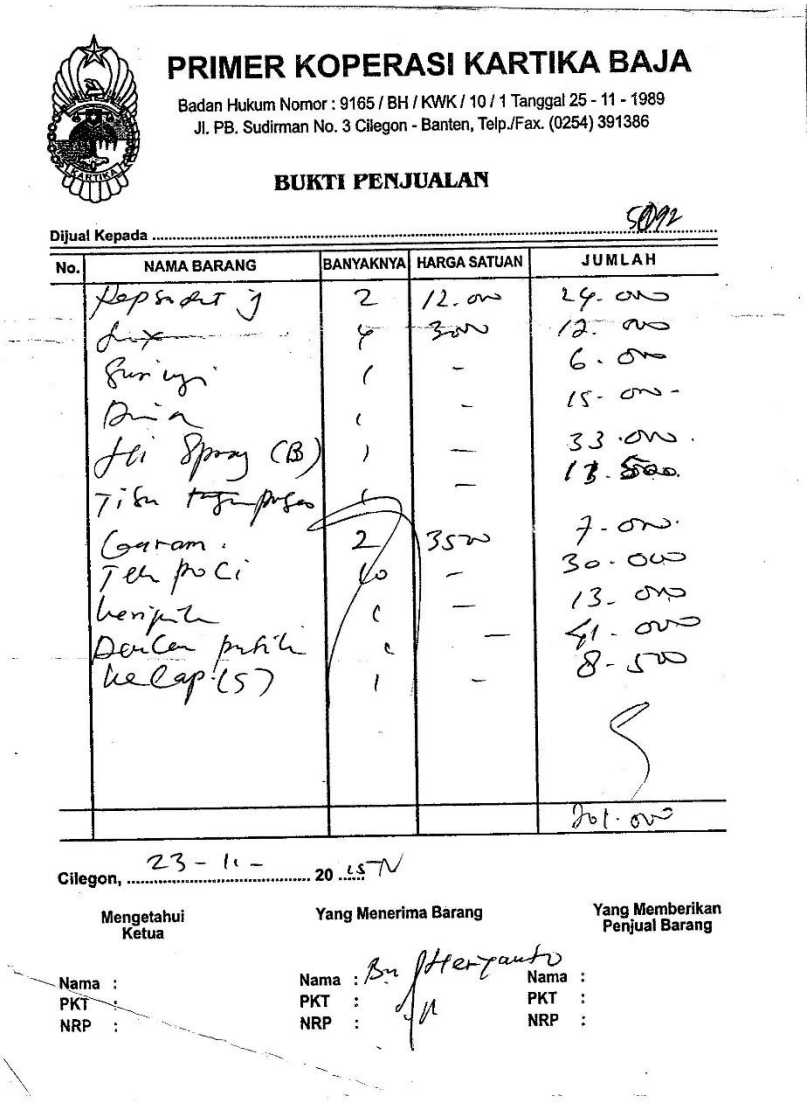
Setelah melakukan tahap *business understanding,* selanjutnya akan ada tahap *data understanding.* Untuk memahami data, ada beberapa tahapan yang harus dilakukan, diantaranya sebagai berikut:

1. Mengumpulkan data awal

   Data dalam penelitian ini adalah *sample* data transaksi penjualan di Primkop Kartika Baja yang berasal dari nota-nota transaksi penjualan yang dimasukan ke dalam *file* berbentuk *Microsoft excel* (.xlsx) yang nantinya akan dilakukan proses *mining* pada program. Paket akan dibentuk per bulan dan data yang digunakan adalah data pada periode 2 bulan terakhir.

2. Menjelaskan data

   Data transaksi awal diambil dari nota penjualan yang terdapat di Primkop Kartika Baja. Berikut adalah contoh nota penjualan sebelum dipindahkan dalam *file Microsoft Excel* (.xlsx) dapat dilihat pada gambar 3.1:



PRIMER KOPERASI KARTIKA BAJA

Badan Hukum Nomor : 9165 / BH / KWK / 10 / 1 Tanggal 25 - 11 - 1989
Jl. PB. Sudirman No. 3 Cilegon - Banten, Telp./Fax. (0254) 391386

BUKTI PENJUALAN

Dijual Kepada .....

No.	NAMA BARANG	BANYAKNYA	HARGA SATUAN	JUMLAH
	Kopi Sachet J	2	12.000	24.000
	[illegible]	4	3000	12.000
	Gula	1	-	6.000
	[illegible]	1	-	15.000
	Teh Sprey (B)	1	-	33.000
	Tisu [illegible]	1	-	13.500
	Garam	2	3500	7.000
	Teh Poci	10	-	30.000
	[illegible]	1	-	13.000
	Deterjen [illegible]	1	-	41.000
	Kelapa (S)	1	-	8.500
				201.000

Cilegon, 23 - 11 - 20 15

Mengetahui Ketua

Nama :
PKT :
NRP :

Yang Menerima Barang

Nama : Bu Heryanto
PKT :
NRP :

Yang Memberikan Penjual Barang

Nama :
PKT :
NRP :

Gambar 3. 1 Contoh Nota Penjualan

Lalu data dari nota penjualan dipindahkan ke dalam *file Microsoft Excel* (.xlsx) seperti pada tabel 3.2:

Tabel 3. 2 Data transaksi dari nota

No. Transaksi	Nama Barang	Qty	Harga Satuan	Jumlah
5001	MACKEREL	1	11000	11000
5001	BERAS	1	56000	56000
5001	INDOMIE GORENG	10	2400	24000
5001	BIMOLI 1000ML	1	14000	14000
5002	BERAS	1	56000	56000
5002	BIMOLI 1000ML	1	14000	14000
5003	TISU NICE 120 SCHT	1	5952	5952

No. Transaksi	Nama Barang	Qty	Harga Satuan	Jumlah
5004	BERAS	1	56000	56000
5004	GULA PUTIH	3	16000	48000
5004	SEGITIGA BIRU	2	9500	19000
5005	MACKEREL	2	11000	22000
5005	BERAS	1	56000	56000
5005	BIMOLI 1000ML	1	14000	14000
5006	MACKEREL	1	11000	11000
5006	GULA PUTIH	5	16000	80000
5007	BERAS	1	56000	56000
5007	GULA PUTIH	5	16000	80000
5007	WHITE KOFFEE	10	1350	13500
5008	MACKEREL	1	11000	11000
5008	GULA PUTIH	2	16000	32000
5008	BIMOLI 1000ML	2	14000	28000
5009	MACKEREL	1	11000	11000
5009	BERAS	1	56000	56000
5009	GULA PUTIH	1	16000	16000
5009	INDOMIE GORENG	3	2400	7200
5010	MACKEREL	1	11000	11000
5010	BERAS	1	56000	56000
5010	GULA PUTIH	2	16000	32000
5010	SEGITIGA BIRU	3	9500	28500
5011	JAKET NKRI	1	145500	145500
5011	INDOMIE AYAM BAWANG	2	2200	4400
5012	PAKET 1	1	23691	23691
5013	MARLBORO PUTIH	1	18350	18350
5014	SARUNG LORENG	1	45500	45500
5014	SABUK	1	35500	35500
5014	DJI SAM SOE	1	13150	13150
5015	BIMOLI 1000ML	1	14000	14000
5016	BARET	1	90500	90500
5017	TOPI SILIWANGI	1	35500	35500
5018	BERAS	1	56000	56000
5018	MACKEREL	1	11000	11000
5018	BIMOLI 1000ML	2	14000	28000
5019	BERAS	1	56000	56000
5019	BARET	1	90500	90500
5019	KAOS ABU-ABU	1	23500	23500
5020	GULA PUTIH	3	16000	48000

Untuk pemaketan sendiri, dibutuhkan kode barang dan jenis untuk melihat apa yang akan dibutuhkan dalam tahap selanjutnya. Untuk kode barang dan jenis dapat dilihat dari tabel stok barang, contoh dari tabel stok barang dapat dilihat pada tabel 3.3:

Tabel 3. 3 Contoh data stok barang

Kode Barang	Nama Barang	Jenis	Stok	Satuan	Harga Pokok	Harga Jual	Jumlah
RK-1	MARLBORO MERAH	ROKOK	27.00	PCS	18,100.00	18,600.00	488,700.00
RK-2	MARLB0RO PUTIH	ROKOK	23.00	PCS	17,850.00	18,350.00	410,550.00
RK-3	JARUM SUPER 12	ROKOK	-1.00	PCS	12,875.00	13,375.00	12,875.00
RK-4	DUNHIL	ROKOK	2.00	PCS	17,800.00	18,300.00	35,600.00
RK-5	GUDANG GARAM FILTER	ROKOK	111.00	PCS	113,350.00	113,850.00	12,581,850.00
RK-6	SURYA 16	ROKOK	19.00	PCS	17,400.00	17,900.00	330,600.00
RK-7	GUDANG GARAM SIGNATUR	ROKOK	90.00	PCS	11,600.00	12,100.00	1,044,000.00
RK-8	MAGNUM	ROKOK	36.00	PCS	11,800.00	12,300.00	424,800.00
RK-9	DJI SAM SOE	ROKOK	128.00	PCS	12,650.00	13,150.00	1,619,200.00
RK-10	SAMPURNA KRETEK	ROKOK	36.00	PCS	9,300.00	9,800.00	334,800.00
SM-11	ABC KECAP BOTOL	SEMBAKO	17.00	PCS	7,000.00	7,500.00	119,000.00
SM-12	ABC MACKEREL	SEMBAKO	1.00	PCS	21,000.00	21,500.00	21,000.00
SM-13	ABC REFIL	SEMBAKO	3.00	PCS	17,000.00	17,500.00	51,000.00
SM-14	ABC SAOS BOTOL BELING	SEMBAKO	17.00	PCS	7,000.00	7,500.00	119,000.00
SM-15	ABC SAOS CABAY	SEMBAKO	0.00	PCS	6,000.00	6,500.00	0.00
SM-16	ABC SAOS KECIL	SEMBAKO	2.00	PCS	5,000.00	5,500.00	10,000.00
SM-17	ABC SARDEN SAOS	SEMBAKO	10.00	PCS	16,500.00	17,000.00	165,000.00
MN-18	ADEM SARI	MNM	1.00	PCS	4,600.00	5,100.00	4,600.00
MN-19	ADES	MNM	4.00	PCS	1,791.00	2,291.00	7,164.00
MK-20	AHH	MKN	1.00	PCS	2,000.00	2,500.00	2,000.00
SM-21	AIR WICK	SEMBAKO	0.00	PCS	9,000.00	9,500.00	0.00
…….	…….	…….	…….	…….	…….	…….	…….
SM-456	STELLA MATIC REFF FANTASY	SEMBAKO	1.00	PCS	25,391.00	25,891.00	25,391.00

Selanjutnya, kode barang dan jenis dimasukan dan disatukan ke dalam *file* data transaksi yang telah tersedia, berikut hasil setelah kode barang dan jenis yang telah dimasukan dapat dilihat pada tabel 3.4:

Tabel 3. 4 Data Transaksi

No. Transaksi	Nama Barang	Kode Barang	Jenis	Qty	Harga Satuan	Jumlah
5001	MACKEREL	SM-227	SEMBAKO	1	11000	11000
5001	BERAS	SM-55	SEMBAKO	1	56000	56000
5001	INDOMIE GORENG	MK-161	MKN	10	2400	24000
5001	BIMOLI 1000ML	SM-57	SEMBAKO	1	14000	14000
5002	BERAS	SM-55	SEMBAKO	1	56000	56000
5002	BIMOLI 1000ML	SM-57	SEMBAKO	1	14000	14000
5003	TISU NICE 120 SCHT	SM-411	SEMBAKO	1	5952	5952
5004	BERAS	SM-55	SEMBAKO	1	56000	56000
5004	GULA PUTIH	SM-141	SEMBAKO	3	16000	48000

No. Transaksi	Nama Barang	Kode Barang	Jenis	Qty	Harga Satuan	Jumlah
5004	SEGITIGA BIRU	SM-336	SEMBAKO	2	9500	19000
5005	MACKEREL	SM-227	SEMBAKO	2	11000	22000
5005	BERAS	SM-55	SEMBAKO	1	56000	56000
5005	BIMOLI 1000ML	SM-57	SEMBAKO	1	14000	14000
5006	MACKEREL	SM-227	SEMBAKO	1	11000	11000
5006	GULA PUTIH	SM-141	SEMBAKO	5	16000	80000
5007	BERAS	SM-55	SEMBAKO	1	56000	56000
5007	GULA PUTIH	SM-141	SEMBAKO	5	16000	80000
5007	WHITE KOFFEE	MN-439	MNM	10	1350	13500
5008	MACKEREL	SM-227	SEMBAKO	1	11000	11000
5008	GULA PUTIH	SM-141	SEMBAKO	2	16000	32000
5008	BIMOLI 1000ML	SM-57	SEMBAKO	2	14000	28000
5009	MACKEREL	SM-227	SEMBAKO	1	11000	11000
5009	BERAS	SM-55	SEMBAKO	1	56000	56000
5009	GULA PUTIH	SM-141	SEMBAKO	1	16000	16000
5009	INDOMIE GORENG	MK-161	MKN	3	2400	7200
5010	MACKEREL	SM-227	SEMBAKO	1	11000	11000
5010	BERAS	SM-55	SEMBAKO	1	56000	56000
5010	GULA PUTIH	SM-141	SEMBAKO	2	16000	32000
5010	SEGITIGA BIRU	SM-336	SEMBAKO	3	9500	28500
5011	JAKET NKRI	SN-177	SANDANG	1	145500	145500
5011	INDOMIE AYAM BAWANG	MK-160	MKN	2	2200	4400
5012	PAKET 1	PK-1	PAKET	1	23691	23691
5013	MARLBORO PUTIH	RK-2	ROKOK	1	18350	18350
5014	SARUNG LORENG	SN-334	SANDANG	1	45500	45500
5014	SABUK	SN-325	SANDANG	1	35500	35500
5014	DJI SAM SOE	RK-9	ROKOK	1	13150	13150
5015	BIMOLI 1000ML	SM-57	SEMBAKO	1	14000	14000
5016	BARET	SN-44	SANDANG	1	90500	90500
5017	TOPI SILIWANGI	SN-415	SANDANG	1	35500	35500
5018	BERAS	SM-55	SEMBAKO	1	56000	56000
5018	MACKEREL	SM-227	SEMBAKO	1	11000	11000
5018	BIMOLI 1000ML	SM-57	SEMBAKO	2	14000	28000
5019	BERAS	SM-55	SEMBAKO	1	56000	56000
5019	BARET	SN-44	SANDANG	1	90500	90500
5019	KAOS ABU-ABU	SN-181	SANDANG	1	23500	23500
5020	GULA PUTIH	SM-141	SEMBAKO	3	16000	48000

Berikut ini adalah penjelasan masing-masing atribut yang terdapat pada tabel hasil pergabungan antara nota dan data stok barang yang telah disatukan dalam *file Micosoft Excel* (.xlsx), dapat dilihat pada tabel 3.5:

Tabel 3.5 Struktur Data Transaksi

Keterangan		
Fungsi	Untuk mengetahui barang yang sering dibeli oleh pembeli	
Format	*Microsoft Excel* (.xlsx)	
Atribut	No_transaksi	Urutan penjualan
	Nama_barang	Nama barang yang dibeli
	Kode_barang	Kode tiap-tiap barang
	Jenis	Jenis tiap-tiap barang
	Qty	Jumlah barang yang dibeli
	Harga_satuan	Harga satuan masing-masing barang
	Jumlah	Jumlah keseluruhan harga barang

3.1.4 Data Preparation

Dalam proses *data mining* ini, tidak semua atribut dalam data transaksi penjualan akan digunakan. Maka sebelum diproses, ada beberapa tahapan yang harus dilakukan agar data siap digunakan untuk masuk ke dalam pemrosesan. Adapun beberapa tahap tersebut antara lain:

1. *Cleaning data*

   Dalam tahap pembersihan data ini, data yang memiliki jenis selain SEMBAKO, MKN, MNM, dan ROKOK dan yang mengandung paket akan dihapus serta data transaksi yang hanya memiliki data tunggal juga akan dihapus sebab data tunggal tidak perlu digunakan dalam proses selanjutnya karena dapat dilihat bahwa data tunggal tidak memiliki hubungan atau keterkaitan dengan antar barang lainnya. Dalam tahap ini, dari 46 *record*, terhapus menjadi 38 *record.* Data yang terdiri dari No.Transaksi 5011, 5012, 5014, 5016, 5017, dan 5019 mengandung jenis barang selain SEMBAKO, MKN, MNM, dan ROKOK dan mengandung paket, maka data tersebut akan dihapuskan, seperti tabel 3.6:

Tabel 3. 6 *Cleaning* jenis dan paket

No. Transaksi	Nama Barang	Kode Barang	Jenis	Qty	Harga Satuan	Jumlah
5001	MACKEREL	SM-227	SEMBAKO	1	11000	11000
5001	BERAS	SM-55	SEMBAKO	1	56000	56000
5001	INDOMIE GORENG	MK-161	MKN	10	2400	24000
5001	BIMOLI 1000ML	SM-57	SEMBAKO	1	14000	14000
5002	BERAS	SM-55	SEMBAKO	1	56000	56000
5002	BIMOLI 1000ML	SM-57	SEMBAKO	1	14000	14000
5003	TISU NICE 120 SCHT	SM-411	SEMBAKO	1	5952	5952
5004	BERAS	SM-55	SEMBAKO	1	56000	56000
5004	GULA PUTIH	SM-141	SEMBAKO	3	16000	48000
5004	SEGITIGA BIRU	SM-336	SEMBAKO	2	9500	19000
5005	MACKEREL	SM-227	SEMBAKO	2	11000	22000
5005	BERAS	SM-55	SEMBAKO	1	56000	56000
5005	BIMOLI 1000ML	SM-57	SEMBAKO	1	14000	14000
5006	MACKEREL	SM-227	SEMBAKO	1	11000	22000
5006	GULA PUTIH	SM-141	SEMBAKO	5	16000	80000
5007	BERAS	SM-55	SEMBAKO	1	56000	56000
5007	GULA PUTIH	SM-141	SEMBAKO	5	16000	80000
5007	WHITE KOFFEE	MN-439	MNM	10	1350	13500
5008	MACKEREL	SM-227	SEMBAKO	1	11000	11000
5008	GULA PUTIH	SM-141	SEMBAKO	2	16000	32000
5008	BIMOLI 1000ML	SM-57	SEMBAKO	2	14000	28000
5009	MACKEREL	SM-227	SEMBAKO	1	11000	11000
5009	BERAS	SM-55	SEMBAKO	1	56000	56000
5009	GULA PUTIH	SM-141	SEMBAKO	1	16000	16000
5009	INDOMIE GORENG	MK-161	MKN	3	2400	7200
5010	MACKEREL	SM-227	SEMBAKO	1	11000	11000
5010	BERAS	SM-55	SEMBAKO	1	56000	56000
5010	GULA PUTIH	SM-141	SEMBAKO	2	16000	32000
5010	SEGITIGA BIRU	SM-336	SEMBAKO	3	9500	28500
5011	INDOMIE AYAM BAWANG	MK-160	MKN	2	2200	4400
5013	MARLBORO PUTIH	RK-2	ROKOK	1	18350	18350
5014	DJI SAM SOE	RK-9	ROKOK	1	13150	13150
5015	BIMOLI 1000ML	SM-57	SEMBAKO	1	14000	14000
5018	BERAS	SM-55	SEMBAKO	1	56000	56000
5018	MACKEREL	SM-227	SEMBAKO	1	11000	11000
5018	BIMOLI 1000ML	SM-57	SEMBAKO	2	14000	28000
5019	BERAS	SM-55	SEMBAKO	1	56000	56000
5020	GULA PUTIH	SM-141	SEMBAKO	3	16000	48000

Lalu tahap kedua yaitu menghapus data tunggal yang terdapat dalam transaksi, seperti yang dapat dilihat di tabel 3.7 nomor transaksi 5003, 5011, 5013, 5014, 5015, 5019, dan 5020 dihapus datanya karena tidak memenuhi syarat. Dari 38 *record*, terdapat 31 *record* yang tersisa. Data yang telah di-*cleaning* dapat dilihat pada tabel 3.7:

Tabel 3.7 *Cleaning* data tunggal

No. Transaksi	Nama Barang	Kode Barang	Jenis	Qty	Harga Satuan	Jumlah
5001	MACKEREL	SM-227	SEMBAKO	1	11000	11000
5001	BERAS	SM-55	SEMBAKO	1	56000	56000
5001	INDOMIE GORENG	MK-161	MKN	10	2400	24000
5001	BIMOLI 1000ML	SM-57	SEMBAKO	1	14000	14000
5002	BERAS	SM-55	SEMBAKO	1	56000	56000
5002	BIMOLI 1000ML	SM-57	SEMBAKO	1	14000	14000
5004	BERAS	SM-55	SEMBAKO	1	56000	56000
5004	GULA PUTIH	SM-141	SEMBAKO	3	16000	48000
5004	SEGITIGA BIRU	SM-336	SEMBAKO	2	9500	19000
5005	MACKEREL	SM-227	SEMBAKO	2	11000	22000
5005	BERAS	SM-55	SEMBAKO	1	56000	56000
5005	BIMOLI 1000ML	SM-57	SEMBAKO	1	14000	14000
5006	MACKEREL	SM-227	SEMBAKO	1	11000	11000
5006	GULA PUTIH	SM-141	SEMBAKO	5	16000	80000
5007	BERAS	SM-55	SEMBAKO	1	56000	56000
5007	GULA PUTIH	SM-141	SEMBAKO	5	16000	80000
5007	WHITE KOFFEE	MN-439	MNM	10	1350	13500
5008	MACKEREL	SM-227	SEMBAKO	1	11000	11000
5008	GULA PUTIH	SM-141	SEMBAKO	2	16000	32000
5008	BIMOLI 1000ML	SM-57	SEMBAKO	2	14000	28000
5009	MACKEREL	SM-227	SEMBAKO	1	11000	11000
5009	BERAS	SM-55	SEMBAKO	1	56000	56000
5009	GULA PUTIH	SM-141	SEMBAKO	1	16000	16000
5009	INDOMIE GORENG	MK-161	MKN	3	2400	7200
5010	MACKEREL	SM-227	SEMBAKO	1	11000	11000
5010	BERAS	SM-55	SEMBAKO	1	56000	56000
5010	GULA PUTIH	SM-141	SEMBAKO	2	16000	32000
5010	SEGITIGA BIRU	SM-336	SEMBAKO	3	9500	28500
5018	BERAS	SM-55	SEMBAKO	1	56000	56000
5018	MACKEREL	SM-227	SEMBAKO	1	11000	11000
5018	BIMOLI 1000ML	SM-57	SEMBAKO	2	14000	28000

2. Pemilihan Atribut

   Dalam proses *data mining* nanti, atribut yang akan dipakai hanya no_transaksi dan kode_barang. No_transaksi di sini berfungsi untuk menunjukkan urutan transaksi sekaligus sebagai pembeda antar transaksi. Sedangkan kode_barang berfungsi sebagai identitas masing-masing barang yang telah dibeli oleh pembeli. Atribut yang telah dipilih dapat dilihat pada tabel 3.8:

Tabel 3.8 Atribut yang dipilih

No. Transaksi	Kode Barang
5001	SM-227
5001	SM-55
5001	MK-161
5001	SM-57
5002	SM-55
5002	SM-57
5004	SM-55
5004	SM-141
5004	SM-336
5005	SM-227
5005	SM-55
5005	SM-57
5006	SM-227
5006	SM-141
5007	SM-55
5007	SM-141
5007	MN-439
5008	SM-227
5008	SM-141
5008	SM-57
5009	SM-227
5009	SM-55
5009	SM-141
5009	MK-161
5010	SM-227
5010	SM-55
5010	SM-141
5010	SM-336
5018	SM-55
5018	SM-227

No. Transaksi	Kode Barang
5018	SM-57

3.1.5 Modelling

Modelling merupakan tahap untuk membuat desain sistem yang akan dibangun. Data yang akan digunakan dalam pengolahan data adalah data yang telah melalui *data preparation* yaitu data yang ada dalam Tabel 3.8. Di dalam tahapan ini akan dilakukan penyelesaian masalah pada pemaketan produk di Primkop Kartika Baja dengan metode *association rule* dan menggunakan algoritma *FP-Growth*.

Metode *association rule* dibagi menjadi dua tahapan yaitu tahap mencari pola frekuensi yang berfungsi untuk mencari frekuensi setiap barang dan pembentukan atau perhitungan asosiasi untuk mencari pola *item* yang memenuhi nilai *minimum support.*

Dalam proses ini, untuk melihat banyaknya variasi data yang diperoleh, nilai *minimum support* yang akan digunakan adalah yang bernilai 2. Jika hasil dari *minimum support* bilangan *decimal*, maka harus dibulatkan ke atas. Hasil dari perhitungan *support count* bisa dilihat dalam Tabel 3.9 berikut:

Tabel 3.9 Support Count

Kode_Barang	Support Count
SM-227	7
SM-55	8
MK-161	2
SM-57	5
SM-141	6
SM-336	2
MN-439	1

Setelah menghitung frekuensi setiap *item* pada seluruh transaksi, tahapan selanjutnya yang harus dilakukan adalah sebagai berikut:

1. Menghapus data yang tidak memenuhi *minimum support* yang telah ditentukan yaitu 2, hasilnya bisa dilihat pada Tabel 3.10 berikut:

Tabel 3.10 Barang yang memenuhi nilai minimum support

Kode_Barang	Support Count
SM-227	7
SM-55	8
MK-161	2
SM-57	5
SM-141	6
SM-336	2

2. Setelah itu, berikan nilai prioritas untuk setiap barang dari yang memiliki nilai terbesar sampai ke terkecil yang dapat dilihat pada Tabel 3.11:

Tabel 3.11 Pemberian Nilai Priority

Kode_Barang	Support Count	Priority
SM-227	7	2
SM-55	8	1
MK-161	2	5
SM-57	5	4
SM-141	6	3
SM-336	2	6

3. Urutkan kode barang sesuai nilai *priority* dari urutan pertama sampai urutan terakhir. Hasilnya dapat dilihat pada Tabel 3.12:

Tabel 3.12 Pengurutan sesuai nilai priority

Kode_Barang	Support Count	Priority
SM-55	8	1
SM-227	7	2
SM-141	6	3
SM-57	5	4
MK-161	2	5
SM-336	2	6

4. Data transaksi yang mengandung barang yang telah memenuhi *minimum support* dan diketahui nilai *priority*-nya akan diurutkan berdasarkan nilai *priority*-nya, dapat dilihat pada tabel 3.13.

Tabel 3.13 Data yang memenuhi minimum support dan diketahui priority-nya

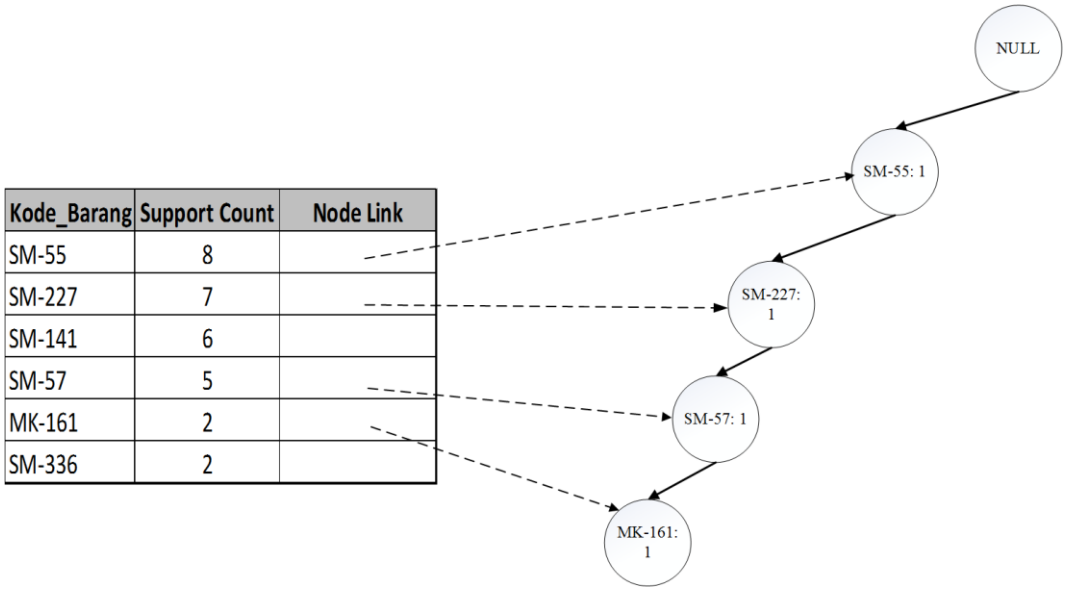
No. Transaksi	Kode Barang
5001	SM-55
5001	SM-227
5001	SM-57
5001	MK-161
5002	SM-55
5002	SM-57
5004	SM-55
5004	SM-141
5004	SM-336
5005	SM-55
5005	SM-227
5005	SM-57
5006	SM-227
5006	SM-141
5007	SM-55
5007	SM-141
5008	SM-227
5008	SM-141
5008	SM-57
5009	SM-55
5009	SM-227
5009	SM-141
5009	MK-161
5010	SM-55
5010	SM-227
5010	SM-141
5010	SM-336
5018	SM-55
5018	SM-227
5018	SM-57

5. Tahap selanjutnya adalah pembuatan *FP-Tree*. Dalam pembuatan *FP-Tree* ada beberapa langkah yaitu:
   a. Buatlah *Tree* dengan *root* bernilai *null.*
   b. Data yang muncul pertama akan menjadi anak pertama dari *root* disertai dengan *support count* = 1.

c. Jika *root* sudah memiliki anak pertama, dan ada data baru yang nilainya tidak sama dengan anak pertama, maka root akan membuat cabang untuk anak yang baru dan *support count*-nya = 1.
d. Jika *root* sudah memiliki anak dan data selanjutnya bernilai sama dengan anak tersebut, maka *support count*-nya bertambah 1.
e. Jika data yag muncul sudah ada dalam notasi anak, maka *support count*-nya bertambah 1.

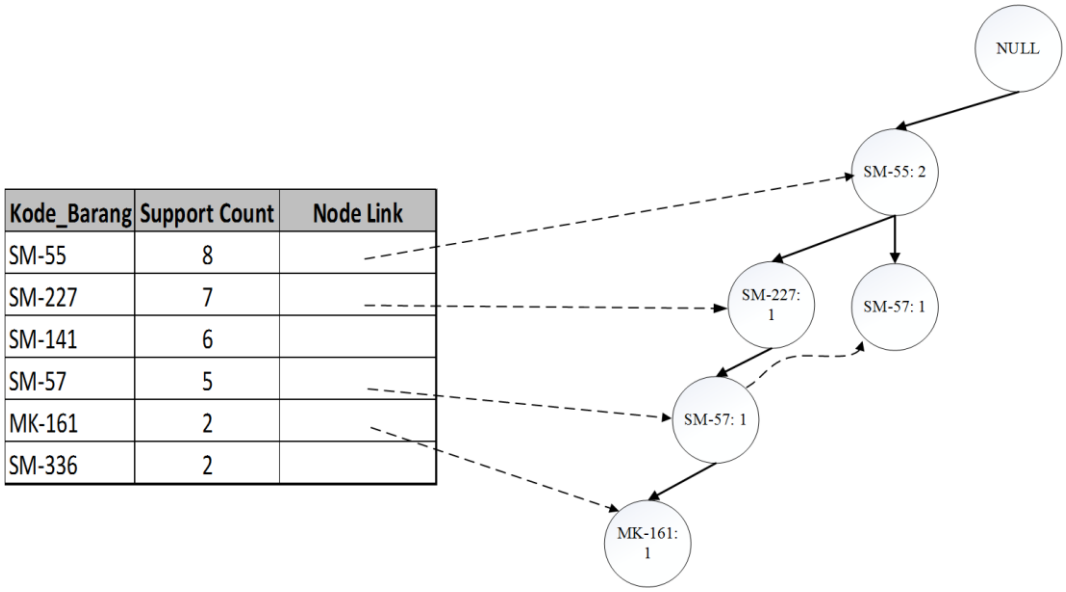
Berikut ini adalah pembuatan *FP-Tree* data transaksi penjualan Primkop Kartika Baja:

1) *FP-Tree* untuk No_Transaksi 5001: SM-55, SM-227, SM-57, MK-161. *FP-Tree* yang dihasilkan bisa dilihat pada gambar 3.2:



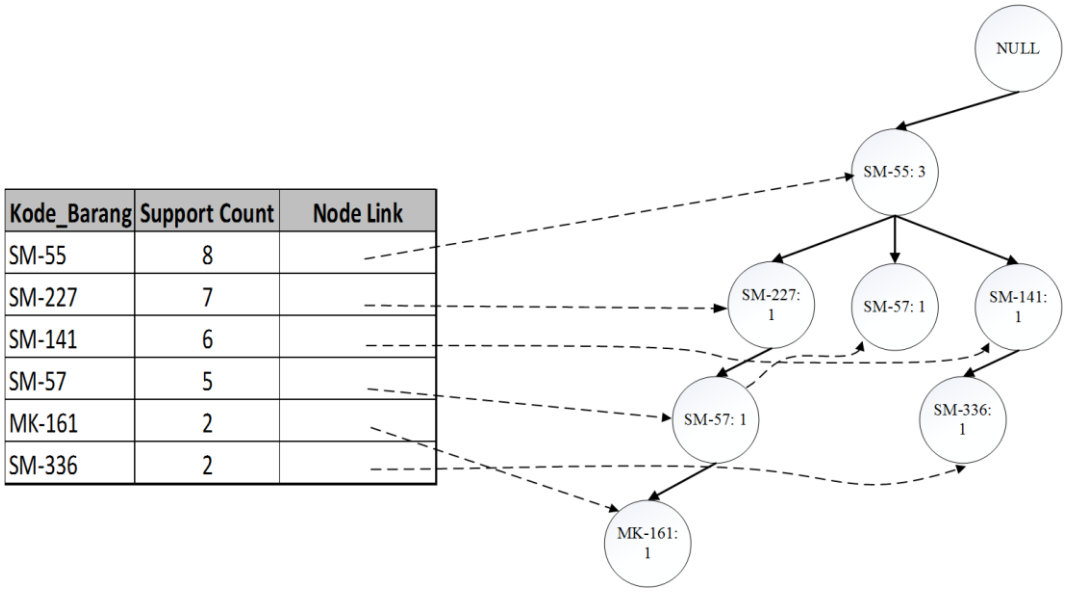
Gambar 3. 2 FP-Tree No_Transaksi 5001

2) *FP-Tree* untuk No_Transaksi 5002: SM-55, SM-57. *FP-Tree* yang dihasilkan bisa dilihat pada gambar 3.3:



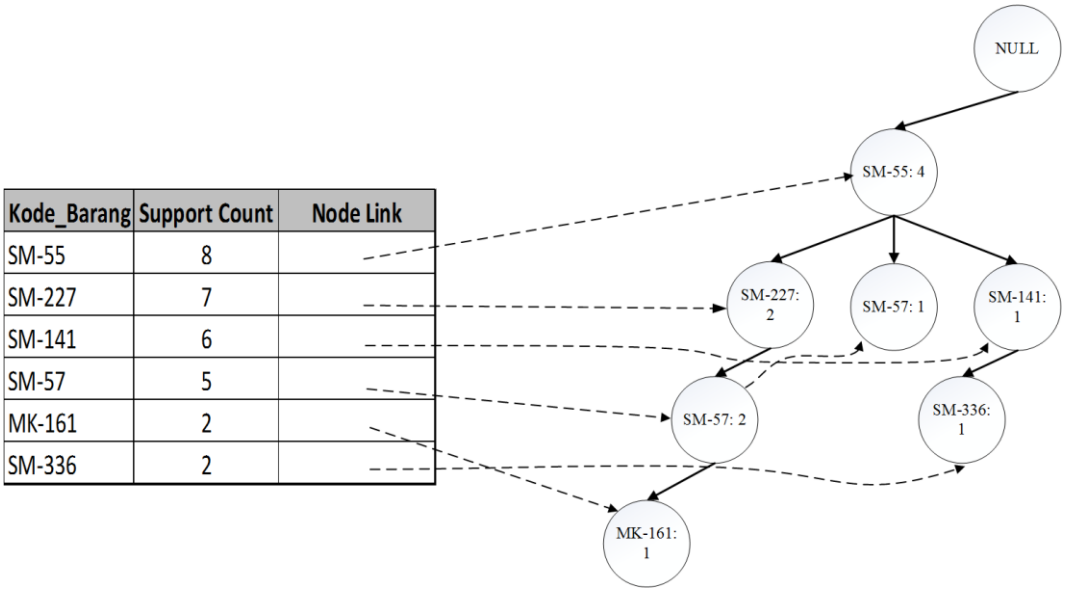
Gambar 3. 3 FP-Tree No_Transaksi 5002

3) *FP-Tree* untuk No_Transaksi 5004: SM-55, SM-141, SM-336. *FP-Tree* yang dihasilkan bisa dilihat pada gambar 3.4:



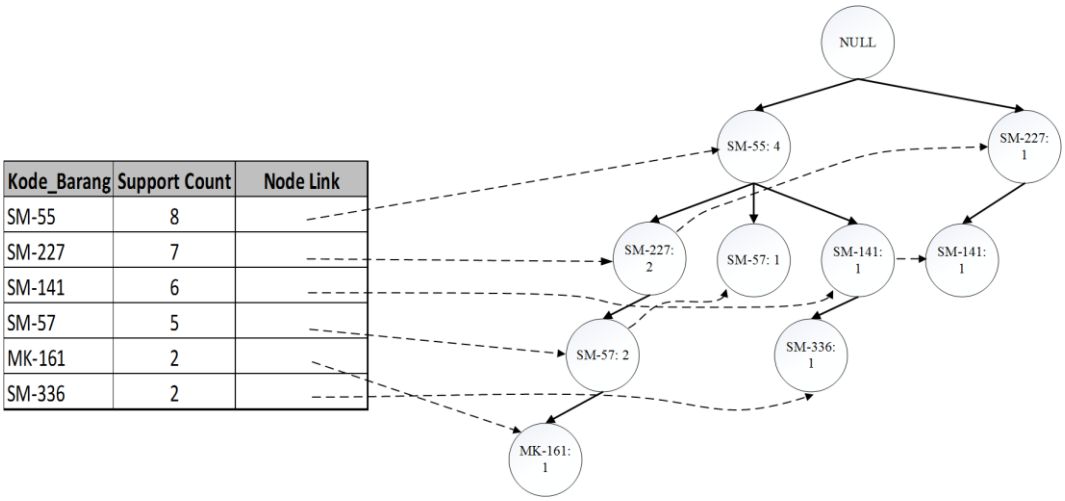
Gambar 3. 4 FP-Tree No_Transaksi 5004

4) *FP-Tree* untuk No_Transaksi 5005: SM-55, SM-227, SM-57. *FP-Tree* yang dihasilkan bisa dilihat pada gambar 3.5:



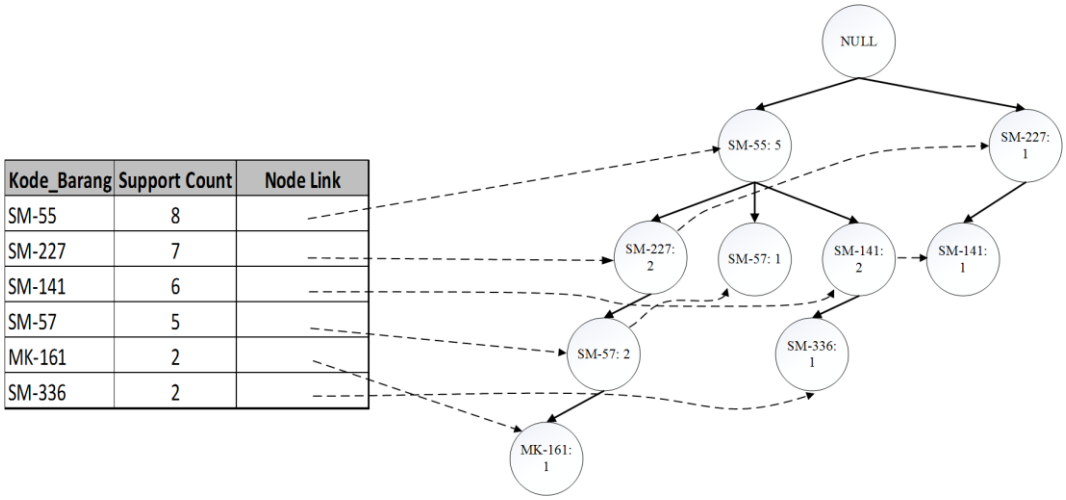
Gambar 3. 5 FP-Tree No_Transaksi 5005

5) *FP-Tree* untuk No_Transaksi 5006: SM-227, SM-141 *FP-Tree* yang dihasilkan bisa dilihat pada gambar 3.6:



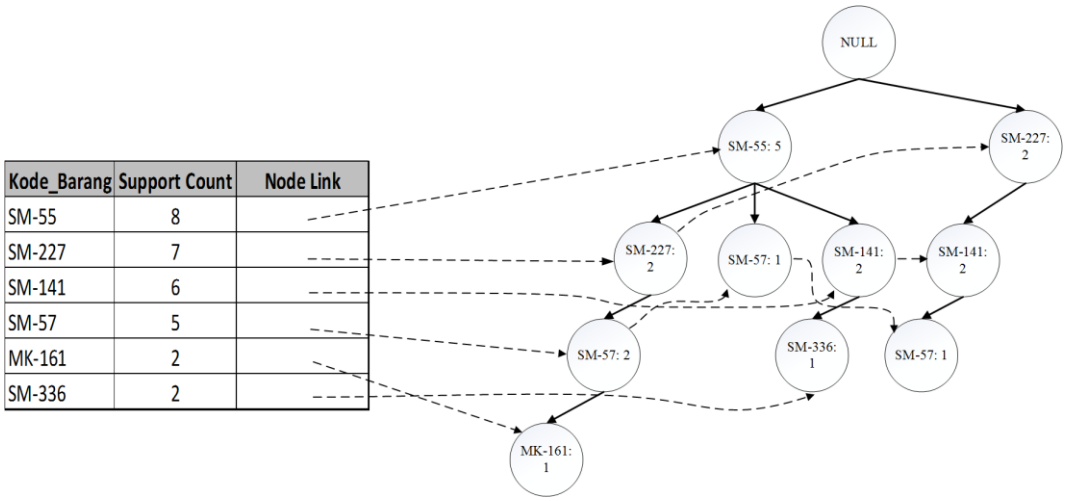
Gambar 3. 6 FP-Tree No_Transaksi 5006

6) *FP-Tree* untuk No_Transaksi 5007: SM-55, SM-141. *FP-Tree* yang dihasilkan bisa dilihat pada gambar 3.7:



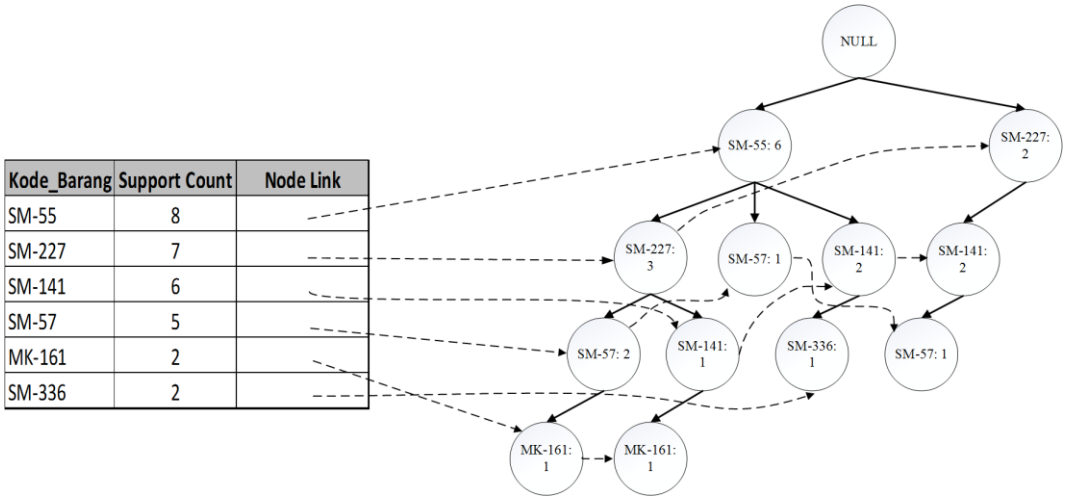
Gambar 3. 7 FP-Tree No_Transaksi 5007

7) *FP-Tree* untuk No_Transaksi 5008: SM-227, SM-141, SM-57. *FP-Tree* yang dihasilkan bisa dilihat pada gambar 3.8:



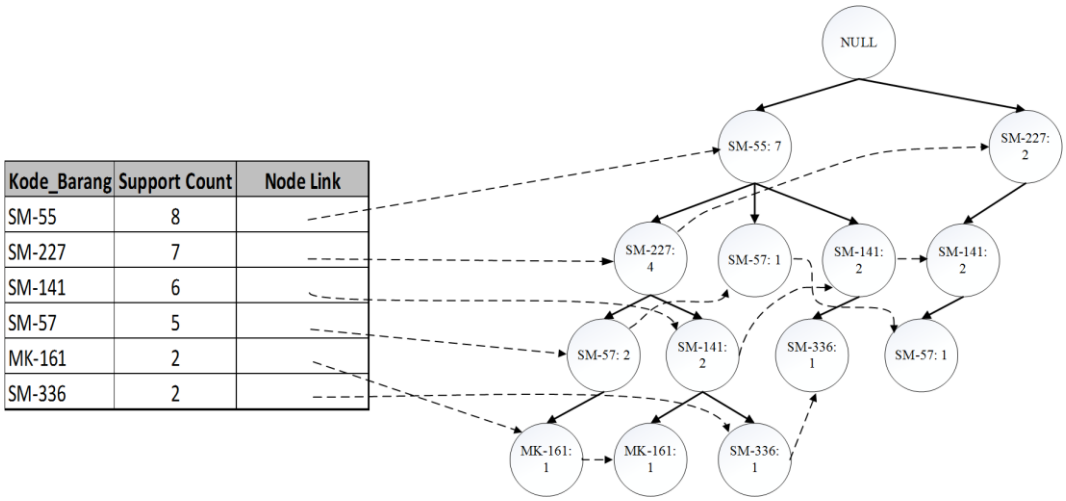
Gambar 3. 8 FP-Tree No_Transaksi 5008

8) *FP-Tree* untuk No_Transaksi 5009: SM-55, SM-227, SM-141, MK-161. *FP-Tree* yang dihasilkan bisa dilihat pada gambar 3.9:



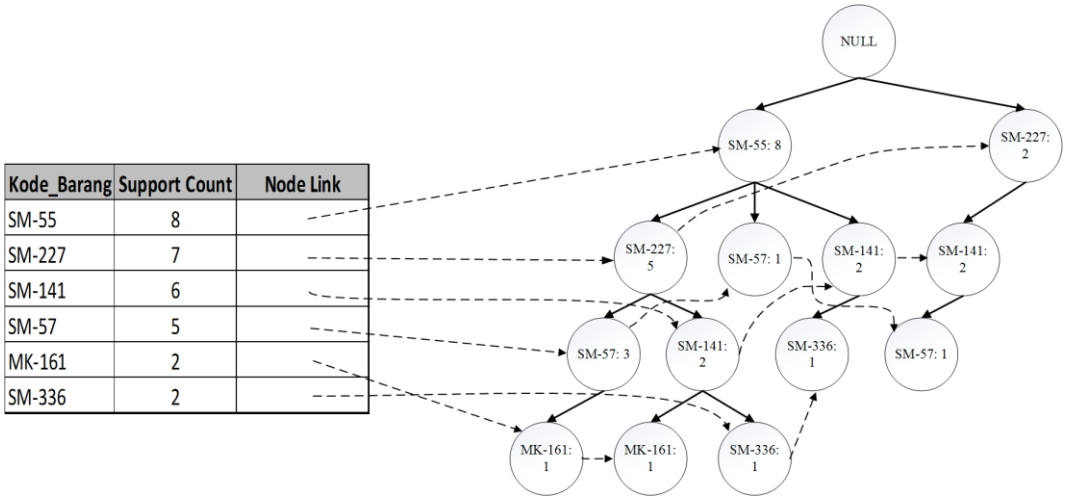
Gambar 3. 9 FP-Tree No_Transaksi 5009

9) *FP-Tree* untuk No_Transaksi 5010: SM-55, SM-227, SM-141, SM-336. *FP-Tree* yang dihasilkan bisa dilihat pada gambar 3.10:



Gambar 3. 10 FP-Tree No_Transaksi 5010

10) *FP-Tree* untuk No_Transaksi 5018: SM-55, SM-227, SM-57. *FP-Tree* yang dihasilkan bisa dilihat pada gambar 3.11:



Gambar 3. 11 FP-Tree No_Transaksi 5018

Setelah membuat *FP-Tree* data transaksi yang ada, maka akan diterapkan algoritma *FP-Growth* untuk mencari *frequent itemset* yang signifikan. Berikut adalah langkah-langkah utama dalam algoritma *FP-Growth*:

a. Tahap pembangkitan *conditional pattern base*

Pembangkitan ini didapatkan melalui hasil *FP-Tree* seluruhnya, dengan mencari *support count* terkecil sesuai dengan pengurutan *priority* dan telah didapat yaitu: SM-336, MK-161, SM-57, MK-141, SM-227, SM-55. Hasil *conditional pattern base* dapat dilihat pada Tabel 3.14:

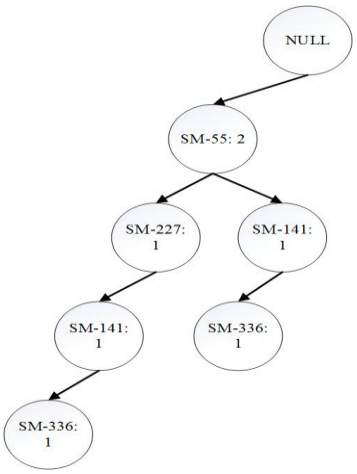
Tabel 3. 14 Conditional Pattern Base

Kode_Barang	Conditional Pattern Base
SM-336	{SM-55, SM-227,SM-141: 1}, {SM-55, SM-141:1}
MK-161	{SM-55, SM-227, SM-57: 1}, {SM-55, SM-227, SM-141:1}
SM-57	{SM-55, SM-227: 3}, {SM-55: 1}, {SM-227, SM-141: 1}
SM-141	{SM-55, SM-227: 2}, {SM-55: 2}, {SM-227: 2}
SM-227	{SM-55: 3}

b. Tahap pembangkitan *conditional FP-Tree*

Pada tahap ini, akan dibuat *tree* untuk masing-masing kode barang, lalu setiap kode barang yang *support count*-nya lebih atau sama dengan *minimum support* akan dibangkitkan dengan *conditional FP-Tree.*

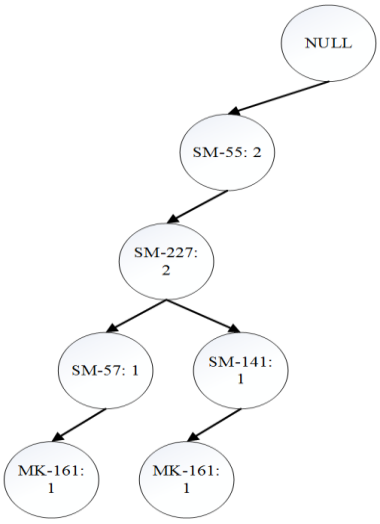
1. *Conditional FP-Tree* untuk kode barang SM-336: {SM-55: 2}



Gambar 3. 12 Conditional FP-Tree SM-336

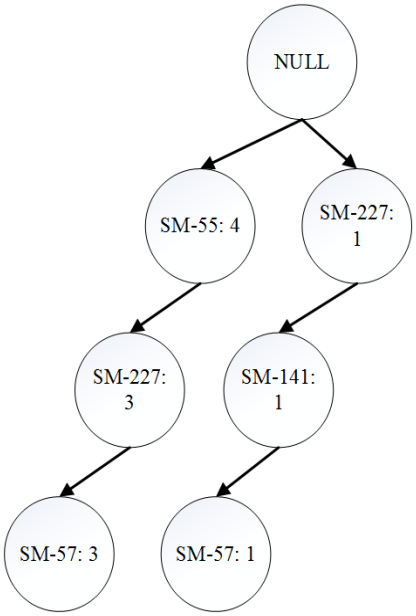
Setelah kode barang SM-336 dibentuk *tree*, maka hanya kode barang SM-55 yang memenuhi *minimum support.*

2. *Conditional FP-Tree* untuk kode barang MK-161: {SM-55:2, SM-227:2}



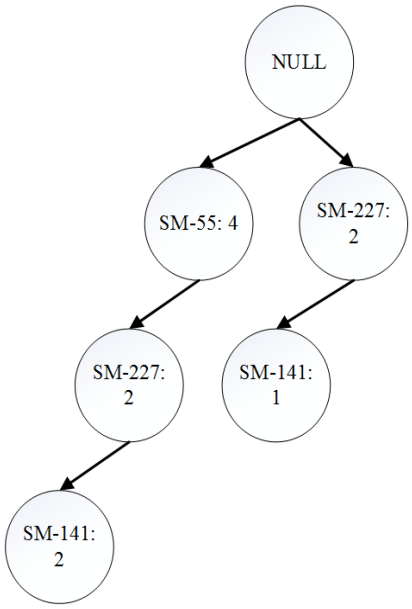
Gambar 3. 13 Conditional FP-Tree MK-161

3. *Conditional FP-Tree* untuk kode barang SM-57: {SM-55:4, SM-227:3}



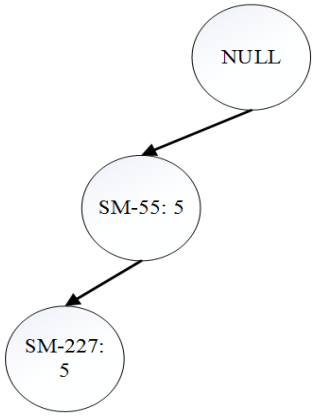
Gambar 3. 14 Conditional FP-Tree SM-57

4. *Conditional FP-Tree* untuk kode barang SM-141: {SM-55:4, SM-227:2} {SM-227:2}



Gambar 3. 15 Conditional FP-Tree SM-141

5. *Conditional FP-Tree* untuk kode barang SM-227: {SM-55:5}



Gambar 3. 16 Conditional FP-Tree SM-227

Setelah mendapatkan hasil *conditional FP-Tree* dari setiap *frequent itemset.*,dapat dilihat hasilnya pada tabel 3.15:

Tabel 3. 15 Conditional FP-Tree

Kode_Barang	Conditional FP-Tree
SM-336	{SM-55: 2}
MK-161	{SM-55:2, SM-227:2}
SM-57	{SM-55:4, SM-227:3}
SM-141	{SM-55:4, SM-227:2} {SM-227:2}
SM-227	{SM-55:5}

c. Tahap pembangkitan *frequent pattern generate*

Pada tahap ini, akan menentukan *frequent pattern generate* dari *conditional FP-Tree* yang telah ada. Hasil setiap *conditional FP-Tree* dapat dilihat pada tabel 3.16:

Tabel 3. 16 Frequent Pattern Generate

Kode_Barang	Frequent Pattern Generate
SM-336	{SM-336, SM-55:2}
MK-161	{MK-161, SM-55:2}, {MK-161, SM-227:2}, {MK-161, SM-55, SM-227:2}
SM-57	{SM-57, SM-55:4}, {SM-57, SM-227:3}, {SM-57, SM-55, SM-227:3}
SM-141	{SM-141. SM-55:4}, {SM-141, SM-227:4}, {SM-141, SM-55, SM-227:2}
SM-227	{SM-227, SM-55:5}

Dari hasil *frequent itemset* yang didapat pada tabel 3.16 di atas, akan di *generate* untuk mendapatkan *rule* dengan *minimum support* 2 dan *minimum confidence* 60% dengan masing-masing *itemset* dikombinasikan dengan *itemset* lain. Dengan menggunakan rumus yang ada pada bagian bab 2 untuk memperoleh nilai *confidence.* Berikut hasil perhitungan *minimum confidence*, dapat dilihat pada tabel 3.17:

Tabel 3. 17 Pembentukan Rule

Rule	*Minimum Confidence*
SM-336 → SM-55	(2/2)*100% = 100%
MK-161 → SM-55	(2/2)*100% = 100%
MK-161 → SM-227	(2/2)*100% = 100%
MK-161 → SM-55, SM-227	(2/2)*100% = 100%
SM-57 → SM-55	(4/5)*100% = 80%
SM-57 → SM-227	(3/5)*100% = 60%
SM-57 → SM-55, SM-227	(3/5)*100% = 60%
SM-141 → SM-55	(4/6)*100% = 66%
SM-141 → SM-227	(4/6)*100% = 66%
SM-141 → SM-55, SM-227	(2/6)*100% = 33%
SM-227 → SM-55	(5/7)*100% = 71%

-

Berikut adalah *rules* yang memenuhi *minimum confidence*, dapat dilihat pada tabel 3.18:

Tabel 3. 18 *Rule* yang memenuhi *minimum confidence*

Rule	*Minimum Confidence*
SM-336 → SM-55	(2/2)*100% = 100%
MK-161 → SM-55	(2/2)*100% = 100%
MK-161 → SM-227	(2/2)*100% = 100%
MK-161 → SM-55, SM-227	(2/2)*100% = 100%
SM-57 → SM-55	(4/5)*100% = 80%
SM-57 → SM-227	(3/5)*100% = 60%
SM-57 → SM-55, SM-227	(3/5)*100% = 60%
SM-141 → SM-55	(4/6)*100% = 67%
SM-141 → SM-227	(4/6)*100% = 67%

Rule	*Minimum Confidence*
SM-227 → SM-55	(5/7)*100% = 71%

Berdasarkan tabel stok barang yag dapat dilihat pada Tabel 3.3, maka kode barang yang didapat harus dikonversi menjadi nama barang seperti yang ada pada tabel 3.19:

Tabel 3. 19 Tabel Konversi

Kode_Barang	Nama_Barang	Harga
SM-336	SEGITIGA BIRU	Rp. 9.500
MK-161	INDOMIE GORENG	Rp. 2.400
SM-227	MACKEREL	Rp. 11.000
SM-55	BERAS	Rp. 56.000
SM-57	BIMOLI 1000ML	Rp. 14.000
SM-141	GULA PUTIH	Rp. 16.000

Sehingga didapatkan hasil *rules* pemaketan produk seperti yang terdapat pada tabel 3.20:

Tabel 3. 20 Hasil *rules* untuk pemaketan produk

Isi Rule	Harga
SEGITIGA BIRU (1 PCS) → BERAS (1 KARUNG)	Rp. 65.500
INDOMIE GORENG (1 PCS) → BERAS (1 KARUNG)	Rp. 58.400
INDOMIE GORENG (1 PCS) → MACKEREL (1 PCS)	Rp. 13.400
INDOMIE GORENG (1 PCS) → BERAS (1 KARUNG), MACKEREL (1 PCS)	Rp. 69.400
BIMOLI 200ML (1 PCS) → BERAS (1 KARUNG)	Rp. 70.000
BIMOLI 200ML (1 PCS) → MACKEREL (1 PCS)	Rp. 25.000
BIMOLI 200ML (1 PCS) → BERAS (1 KARUNG), MACKEREL (1 PCS)	Rp. 81.000

Isi Rule	Harga
GULA PUTIH (1 KG) → BERAS (1 KARUNG)	Rp. 72.000
GULA PUTIH (1 KG) → MACKEREL (1 PCS)	Rp. 27.000
MACKEREL (1 PCS) → BERAS (1 KARUNG)	Rp. 67.000

Setelah mendapatkan hasil *rules*, maka langkah selajutnya adalah membuat pemaketan produk dari *rules* yang telah didapatkan. Pihak Primkop Kartika Baja sendiri memberi beberapa batasan untuk pembuatan paket, diantaranya:

1. Jumlah barang yang dipaketkan dalam satu paket adalah tiga barang.
2. Jumlah setiap barang dalam suatu paket hanya berjumlah satu.
3. Jumlah harga setiap paket tidak lebih dari Rp. 100.000,- dan tidak kurang dari Rp. 50.000,-
4. Paket yang akan dibuat adalah dua paket.
5. Pergantian paket dilakukan setiap dalam periode per bulan.

Dari batasan yang ada, pemaketan yang terbentuk dapat dilihat di tabel 3.21:

Tabel 3. 21 Paket yang terbentuk

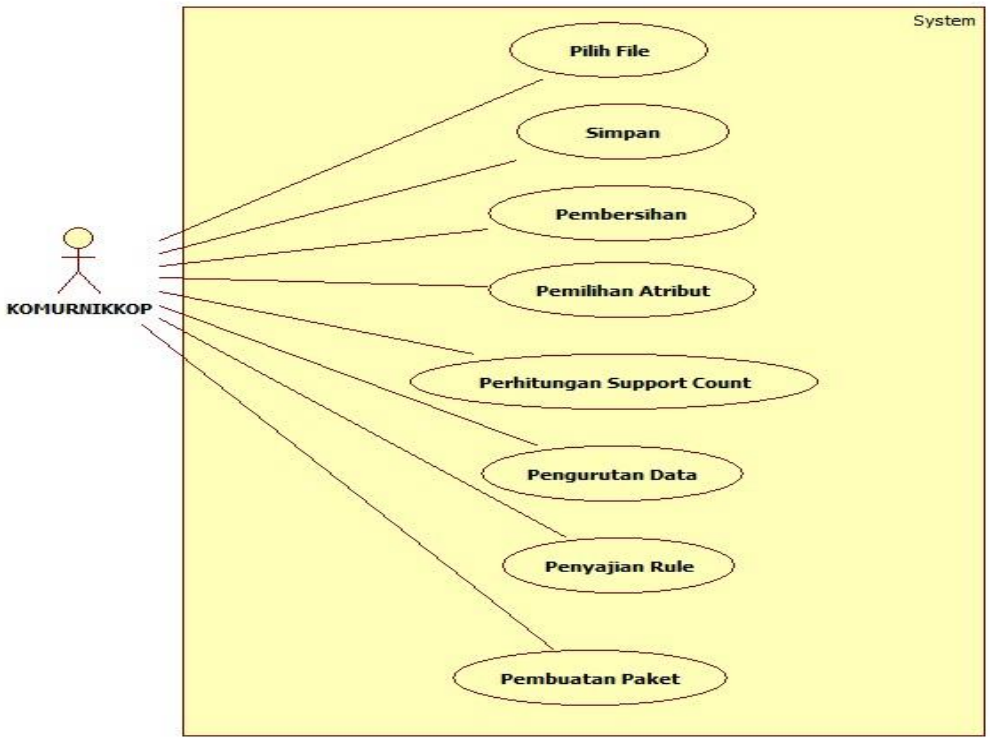
Paket	Isi Paket
PAKET 1	1. INDOMIE GORENG (1 PCS) 2. BERAS (1 KARUNG) 3. MACKEREL (1 PCS) Total Harga Rp. 69.400
PAKET 2	1. BIMOLI 200ML (1 PCS) 2. BERAS (1 KARUNG) 3. MACKEREL (1 PCS) Total Harga Rp. 81.000

3.2 Analisis Kebutuhan Fungsional

Pada sub bab ini, akan diuraikan analisis kebutuhan fungsional yang bertujuan untuk pembangunan perangkat lunak. Aplikasi yanga akan dibangun berorientasi objek dengan menggunakan pemodelan UML. Pemodelan yang akan digunakan untuk memodelkan terdiri dari diagram *use case, activity diagram, sequence diagram,* dan *class diagram.*

3.2.1 Diagram *Use Case*

Diagram *use case* pemodelan untuk kelakuan sistem yang dibuat. Untuk *use case* di sistem ini, dapat dilihat pada gambar 3.17 berikut:



Gambar 3. 17 Use case diagram

Penjelasan definisi user dapat dilihat pada tabel 3.22 dan deskripsi *use case* pada tabel 3.23:

Tabel 3. 22 Definisi *user*

No	User	Deskripsi
1	KOMURNIKKOP	Aktor ini memiliki wewenang untuk melakukan seluruh fungsional yang ada pada sistem.

Tabel 3. 23 Deskripsi *use case*

No	Use Case	Deskripsi
1	Pilih *File*	Berisi proses yang berkaitan dengan pemilihan file yang akan di-*import*
2	Simpan	Berisi proses penyimpanan data ke dalam *database*
3	Pembersihan	Berisi proses bisnis yang berkaitan dengan pembersihan pada data latih.

No	Use Case	Deskripsi
4	Pemilihan Atribut	Berisi proses bisnis yang berkaitan dengan pemilihan atribut pada data latih.
5	Perhitungan *Support Count*	Berisi proses bisnis yang berkaitan dengan perhitungan *support count* pada data latih.
6	Pengurutan Data	Berisi proses bisnis yang berkaitan dengan pengurutan pada data latih.
7	Penyajian *rule*	Berisi proses bisnis yang berkaitan dengan penyajian rule hasil perhitungan algoritma *FP-Growth* pada data latih.
8	Pembuatan Paket	Berisi proses bisnis yang berkaitan dengan pembentukan paket hasil perhitungan algoritma *FP-Growth* pada data latih.

3.2.1.1 Skenario *Use Case*

Skenario *use case* berguna untuk menjelaskan setiap proses yang terdapat *diagram use case* dari gambar 3.17. Berikut adalah skenario *use case* yang akan dibuat:

1. Skenario *use case* pilih *file*

Tabel 3. 24 Skenario *use case* pilih *file*

Nama Use Case	Pilih file	
Use Case Terkait	-	
Goal	Memilih dan menampilkan file dari file excel	
Preconditions	File Excel harus mengikuti format yang telah disediakan dan sudah berisikan data, serta lokasi file excel sudah didefinisikan oleh user	
Successful End Condition	Data yang berasal dari file excel tersimpan pada tabel yang disediakan	
Failed End Condition	Data yang berasal dari file excel tidak tersimpan pada tabel yang disediakan	
Aktor	KOMURNIKKOP	
Trigger	User meminta sistem untuk meng-import data	
Main Flow	Langkah	Aksi
	1	User memasukan atau mencari lokasi file Excel yang akan diekstrak.
	2	User meminta sistem untuk meng-import data dari file Excel tersebut.
	3	Sistem membuka file Excel yang sudah dipilih user.
	4	Sistem melakukan validasi format file
	5	Sistem melakukan pembacaan data yang ada pada file Excel.
	6	Sistem menampilkan file Excel yang telah dibaca.
Extension	Langkah	Aksi Percabangan
	4.1	Sistem menampilkan pesan bahwa file yang dimasukan tidak berformat .xlsx

2. Skenario *use case* Simpan

Tabel 3. 25 Skenario *use case* simpan

Nama Use Case	Simpan
Use Case Terkait	-
Goal	Menyimpan file dari file excel ke dalam database

Preconditions	File Excel harus mengikuti format yang telah disediakan dan sudah berisikan data	
Successful End Condition	Data yang berasal dari file excel tersimpan dalam database	
Failed End Condition	Data yang berasal dari file excel tidak tersimpan pada database	
Aktor	KOMURNIKKOP	
Trigger	User meminta sistem untuk menyimpan data	
Main Flow	Langkah	Aksi
	1	User meminta sistem menyimpan file Excel yang telah dibaca ke dalam database
	2	Sistem melakukan validasi atribut file excel tersebut
	3	Sistem menyimpan data kedalam database
Extension	Langkah	Aksi Percabangan
	3.1	Sistem tidak menyimpan data ke dalam database dan menampilkan pesan data gagal disimpan

3. Skenario *use case* pembersihan

Tabel 3. 26 Skenario *use case* pembersihan

Nama Use Case	Pembersihan	
Use Case Terkait	-	
Goal	Membersihkan data yang ada pada tabel	
Preconditions	Sistem sudah memiliki data hasil import	
Successful End Condition	Tabel dibersihkan dan ditampilkan pada sistem	
Failed End Condition	Tabel tidak dibersihkan dan tidak ditampilkan pada sistem	
Aktor	KOMURNIKKOP	
Trigger	User meminta sistem untuk membersihkan data	
Main Flow	Langkah	Aksi
	1	User meminta sistem untuk membersihkan data
	2	Sistem melakukan pembersihan data
	3	Sistem menampilkan hasil pembersihan data
Extension	Langkah	Aksi Percabangan
	-	-

4. Skenario *use case* pemilihan atribut

Tabel 3. 27 Skenario *use case* pemilihan atribut

Nama Use Case	Pemilihan atribut	
Use Case Terkait	-	
Goal	Memilih atribut yang ada pada tabel agar sesuai dengan atribut yang telah ditentukan oleh user	
Preconditions	Sistem sudah memiliki data hasil pembersihan	
Successful End Condition	Tabel dipilih sesuai atribut yang diperlukan user dan ditampilkan pada sistem	
Failed End Condition	Tabel tidak diseleksi sesuai pilihan user dan tidak ditampilkan pada sistem	
Aktor	KOMURNIKKOP	
Trigger	User meminta sistem untuk memilihkan atribut	
Main Flow	Langkah	Aksi
	1	User meminta sistem melakukan pemilihan atribut

	2	Sistem melakukan pemilihan atribut
	3	Sistem menampilkan hasil pemilihan atribut
Extension	Langkah	Aksi Percabangan
	-	-

5. Skenario *use case* perhitungan *support count*

Tabel 3. 28 Skenario *use case* perhitungan *support count*

Nama Use Case	Perhitungan support count	
Use Case Terkait	-	
Goal	Menghitung support count masing-masing kode barang	
Preconditions	Sistem sudah memiliki data hasil pemilihan atribut	
Successful End Condition	Data dalam data latih dihitung support count-nya dan ditampilkan pada sistem	
Failed End Condition	Sistem tidak menampilkan data yang telah dihitung support count	
Aktor	KOMURNIKKOP	
Trigger	User meminta sistem untuk menghitung support count	
Main Flow	Langkah	Aksi
	1	User meminta sistem untuk menghitung support count
	2	Sistem melakukan perhitungan support count
	3	Sistem menampilkan perhitungan support count
Extension	Langkah	Aksi Percabangan
	-	-

6. Skenario *use case* pengurutan data

Tabel 3. 29 Skenario *use case* pengurutan data

Nama Use Case	Pengurutan data	
Use Case Terkait	-	
Goal	Mengurutkan data sesuai dengan minimum support yang telah ditentukan	
Preconditions	Sistem sudah memiliki data hasil peritungan support count	
Successful End Condition	Data dalam data latih diurutkan dan ditampilkan pada sistem	
Failed End Condition	Sistem tidak menampilkan data yang telah diurutkan	
Aktor	KOMURNIKKOP	
Trigger	User meminta sistem untuk mengurutkan data	
Main Flow	Langkah	Aksi
	1	User memasukan nilai minimum support
	2	User meminta sistem untuk mengurutkan data
	3	Sistem melakukan pengurutan data
	4	Sistem menampilkan hasil pengurutan data
Extension	Langkah	Aksi Percabangan
	-	-

7. Skenario *use case* penyajian *rule*

Tabel 3. 30 Skenario *use case* penyajian *rule*

Nama Use Case	Penyajian Rule	
Use Case Terkait	Analisis Pemaketan Produk	
Goal	Menampilkan rule yang dihasilkan dari pengurutan data latih	
Preconditions	Sistem sudah memiliki data hasil analisis pemaketan produk	
Successful End Condition	Data latih yang telah dianalisis ditampilkan pada sistem	
Failed End Condition	Sistem tidak menampilkan data yang telah dianalisis	
Aktor	KOMURNIKKOP	
Trigger	User meminta sistem untuk menganalisis data	
Main Flow	Langkah	Aksi
	1	User memasukan nilai minimum confidence
	2	User meminta sistem untuk menyajikan rule sesuai minimum confidence
	3	Sistem menampilkan hasil rule sesuai minimum confidence yang telah dimasukan
Extension	Langkah	Aksi Percabangan
	-	-

8. Skenario *use case* pembuatan paket

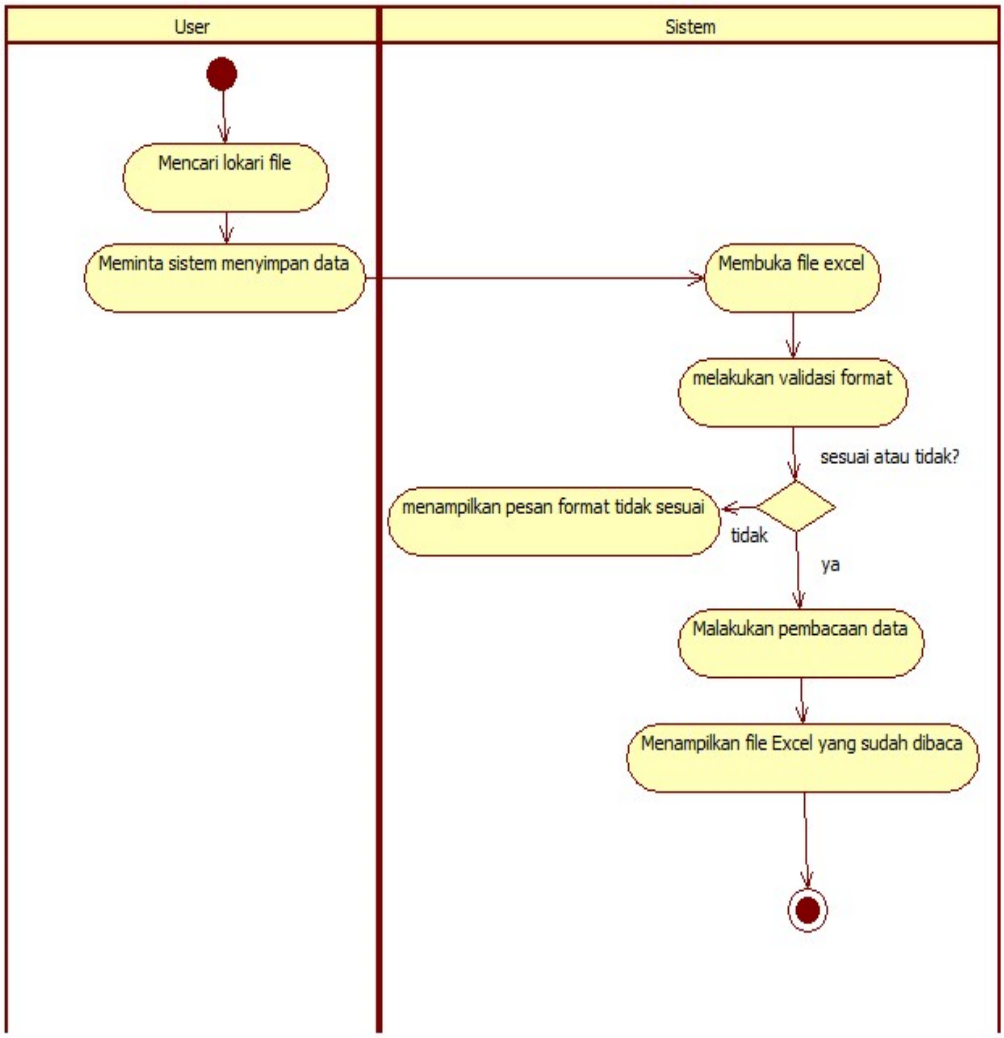
Tabel 3. 31 Skenario *use case* pembuatan paket

Nama Use Case	Pembuatan Paket	
Use Case Terkait	-	
Goal	Menampilkan paket yang telah dibuat	
Preconditions	Sistem sudah memiliki data hasil penyajian rule	
Successful End Condition	Pemaketan produk ditampilkan pada sistem	
Failed End Condition	Sistem tidak menampilkan pemaketan produk	
Aktor	KOMURNIKKOP	
Trigger	User meminta sistem untuk membuat paket	
Main Flow	Langkah	Aksi
	1	User memasukkan harga minimum dan maksimum
	2	User meminta sistem untuk membuat paket
	3	Sistem melakukan pembuatan paket
	4	Sistem menampilkan paket yang telah dibuat
Extension	Langkah	Aksi Percabangan
	-	-

3.2.2 *Activity Diagram*

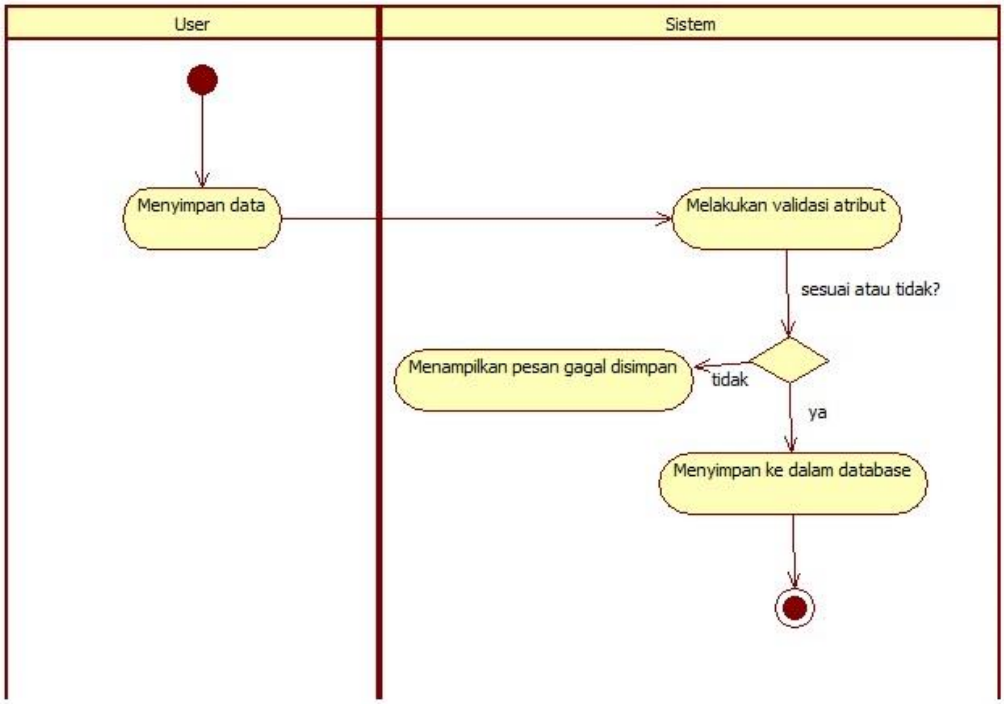
Activity Diagram dibuat untuk menggambarkan alur aktivitas yang ada pada sebuah *use case. Activity diagram* yang ada pada sistem ini adalah sebagai berikut:

1 *Activity diagram* pilih data



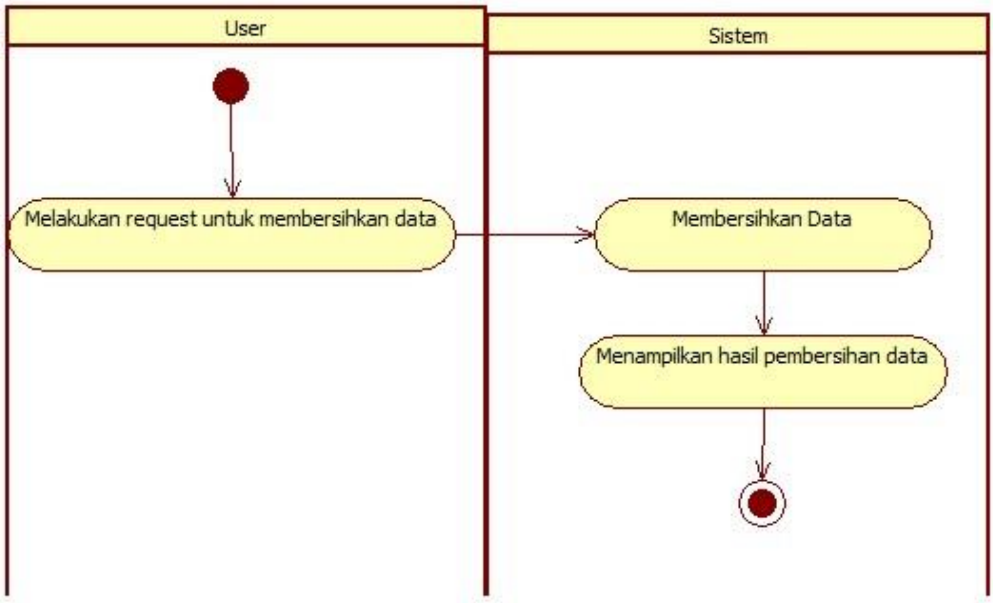
Gambar 3. 18 *Activity diagram* pilih data

2 *Activity diagram* simpan



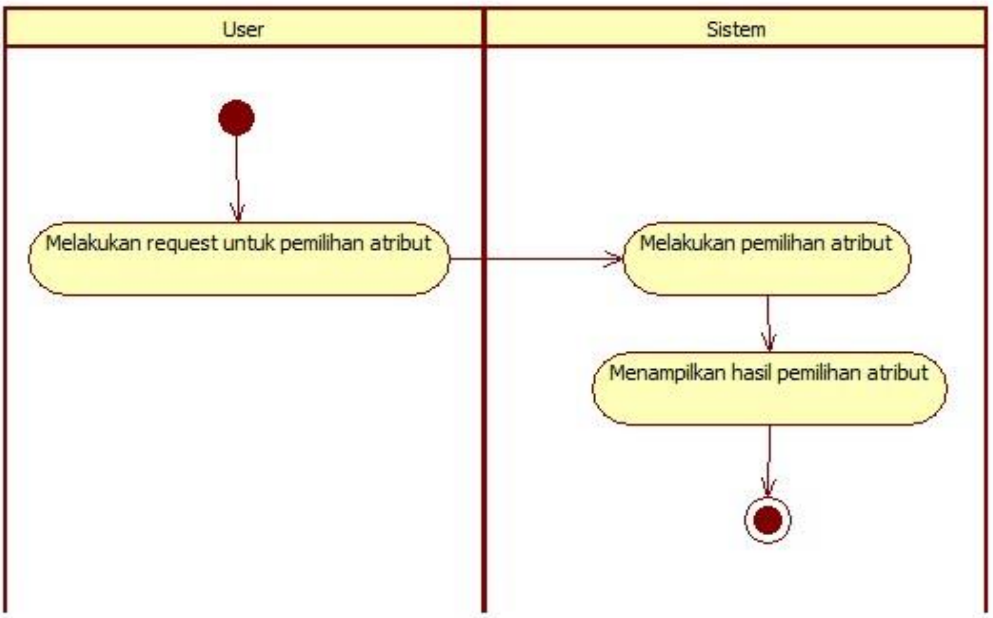
Gambar 3. 19 *Activity diagram* simpan

3 *Activity diagram* pembersihan



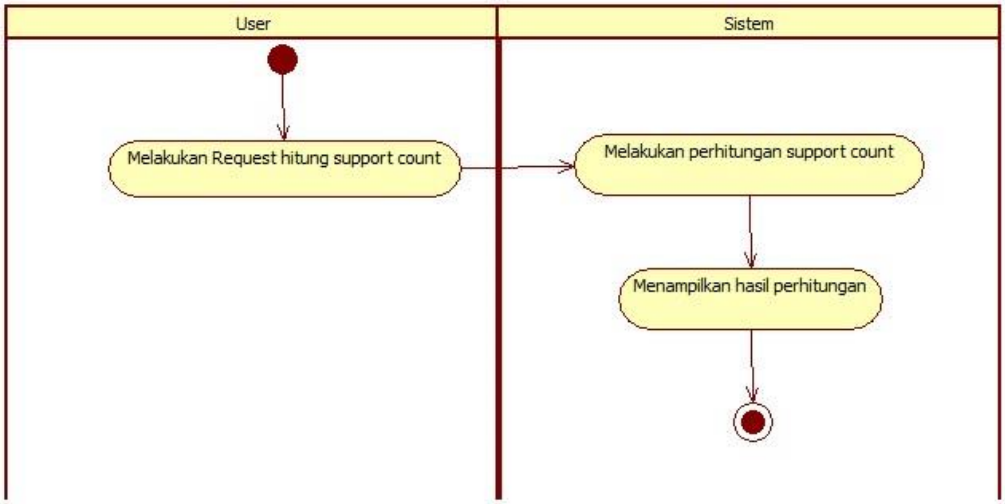
Gambar 3. 20 *Activity diagram* pembersihan

4 *Activity diagram* pemilihan atribut



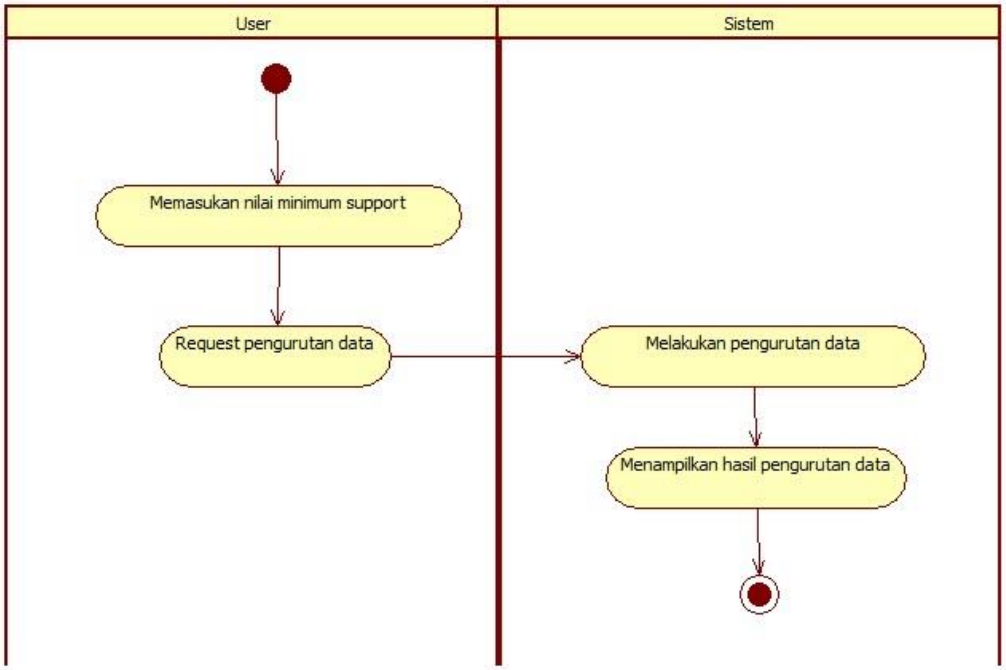
Gambar 3. 21 *Activity diagram* pemilihan atribut

5 *Activity diagram* perhitungan *support count*



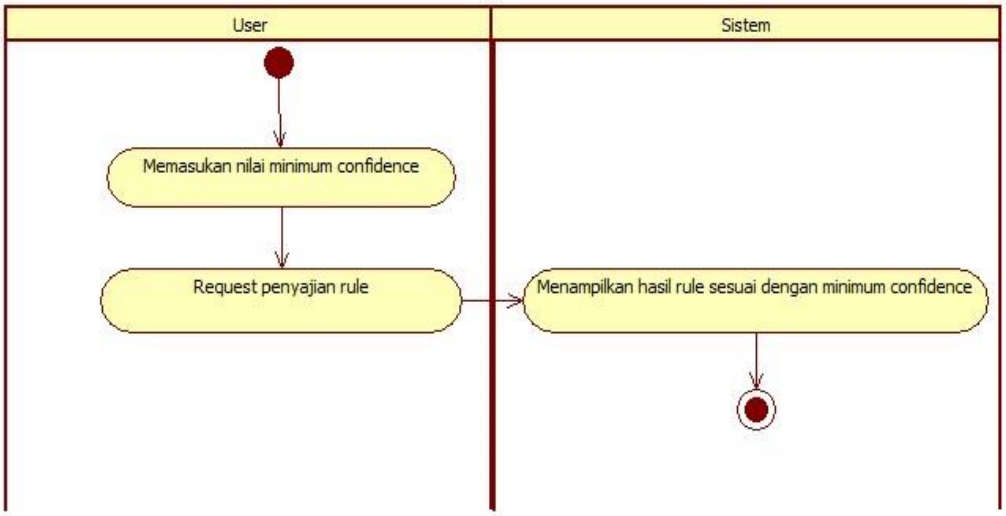
Gambar 3. 22 *Activity diagram* perhitungan *support count*

6 *Activity diagram* pengurutan data



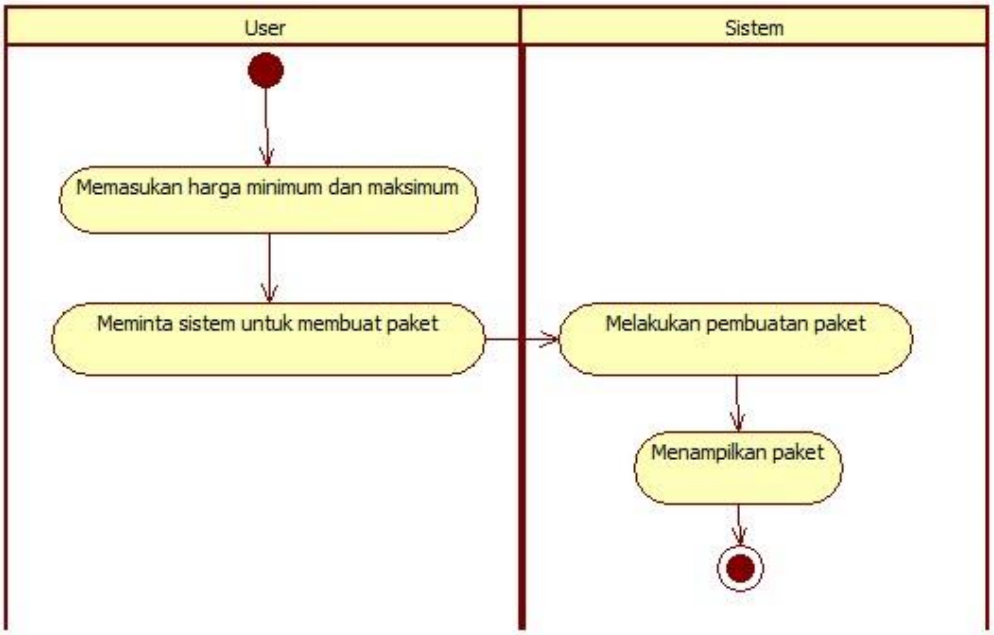
Gambar 3. 23 *Activity diagram* pengurutan data

7 *Activity diagram* penyajian *rule*



Gambar 3. 24 *Activity diagram* penyajian rule

8 *Activity diagram* pembuatan paket

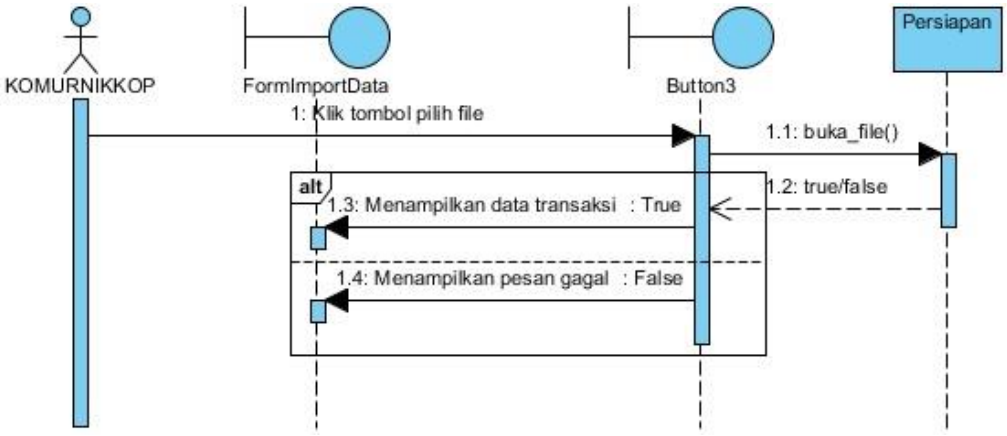


Gambar 3. 25 *Activity diagram* pembuatan paket

3.2.3 *Sequence Diagram*

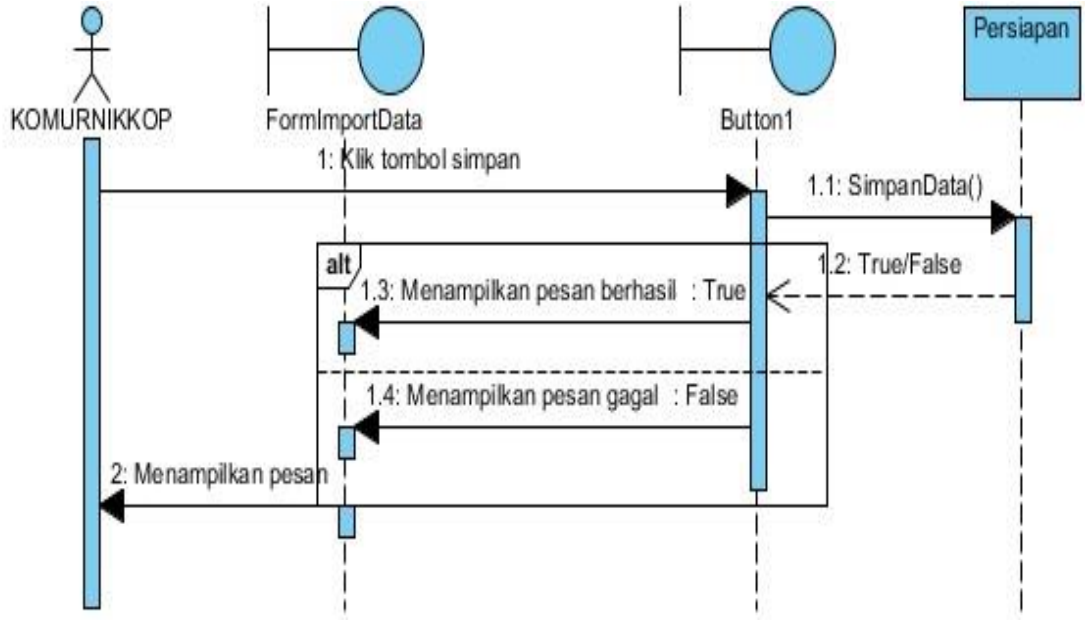
Sequence diagram berguna untuk menggambarkan arus pekerjaan, pesan yang disampaikan dan bagaiman elemen-elemen bekerja sama untuk mencapai tujuan hasil. Berikut adalah *sequence diagram* pada aplikasi yang sedang dibangun:

1. *Sequence Diagram* Pilih Data



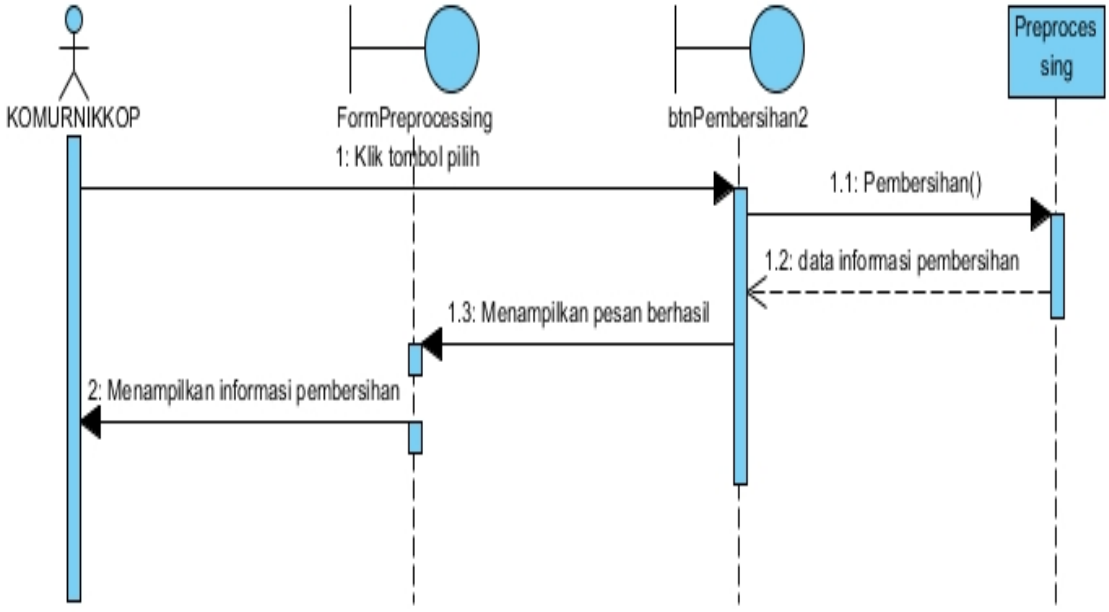
Gambar 3. 26 *Sequence Diagram* Pilih Data

2. *Sequence Diagram* Simpan



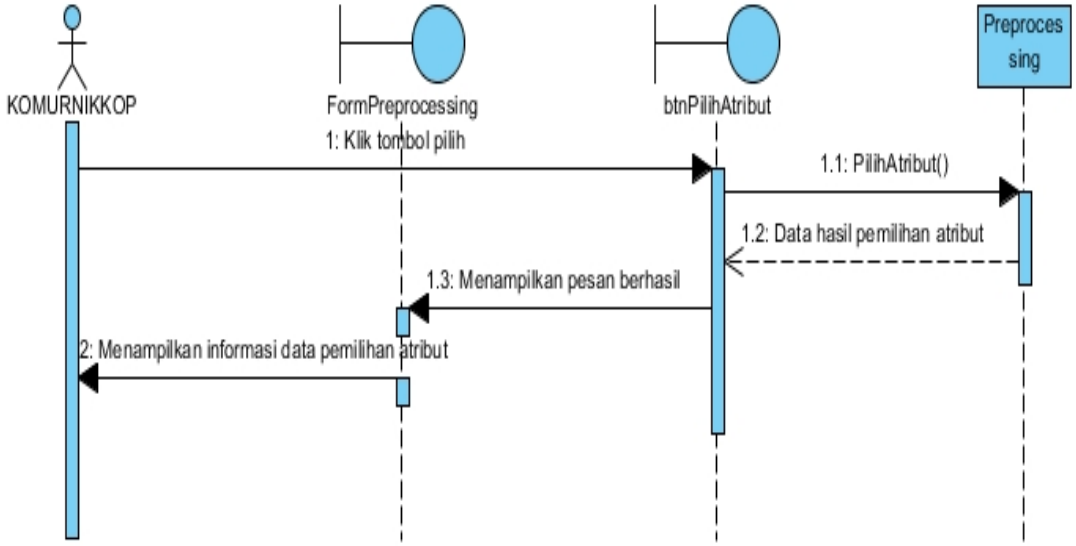
Gambar 3. 27 *Sequence Diagram* Simpan

3. *Sequence Diagram* Pembersihan



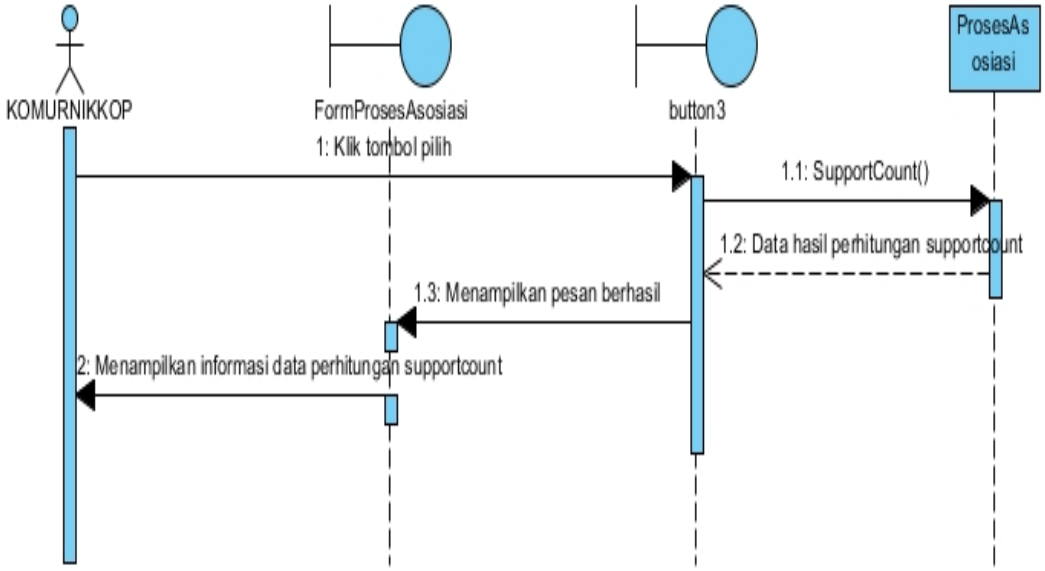
Gambar 3. 28 *Sequence Diagram* Pembersihan

4. *Sequence Diagram* Pemilihan Atribut



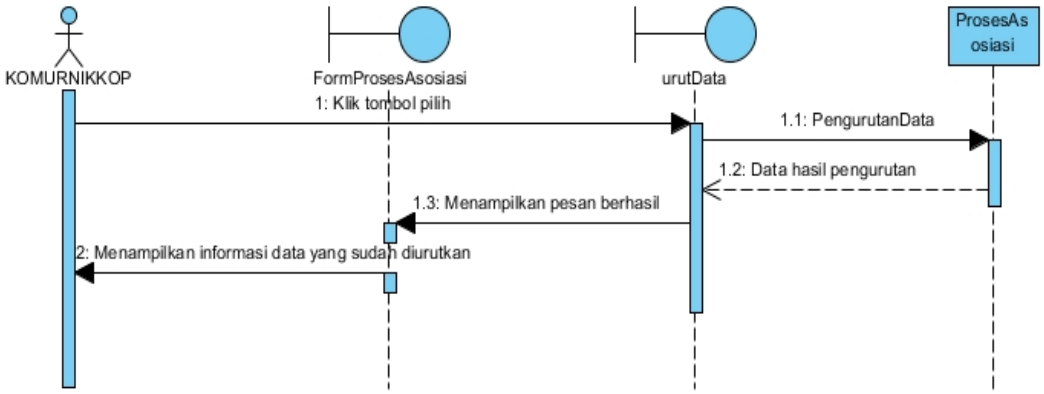
Gambar 3. 29 *Sequence Diagram* Pemilihan Atribut

5. *Sequence Diagram* Perhitungan *Support Count*



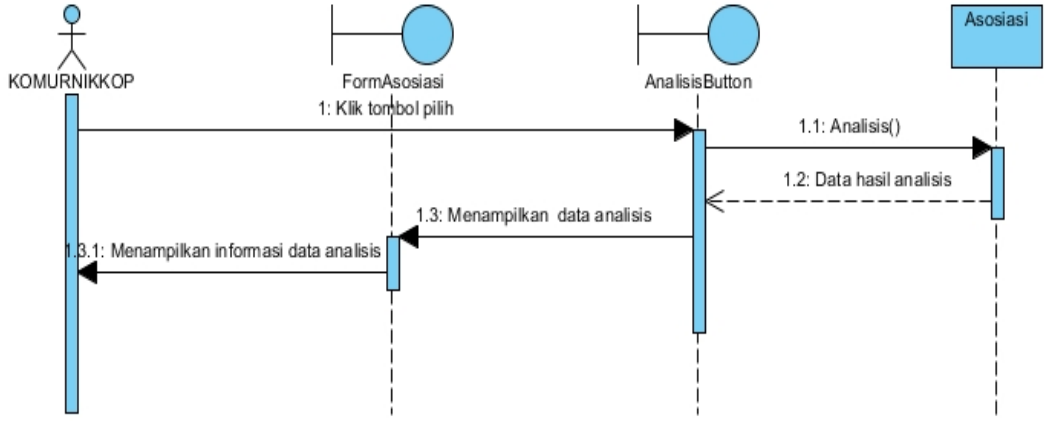
Gambar 3. 30 *Sequence Diagram* Perhitungan Support Count

6. *Sequence Diagram* Pengurutan Data



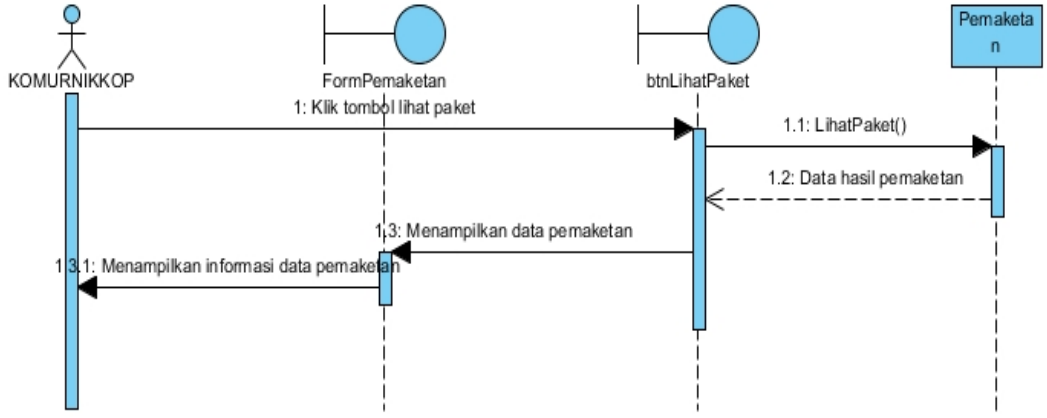
Gambar 3. 31 *Sequence Diagram* Pengurutan Data

7. *Sequence Diagram* Penyajian Rule



Gambar 3. 32 *Sequence Diagram* Penyajian Rule

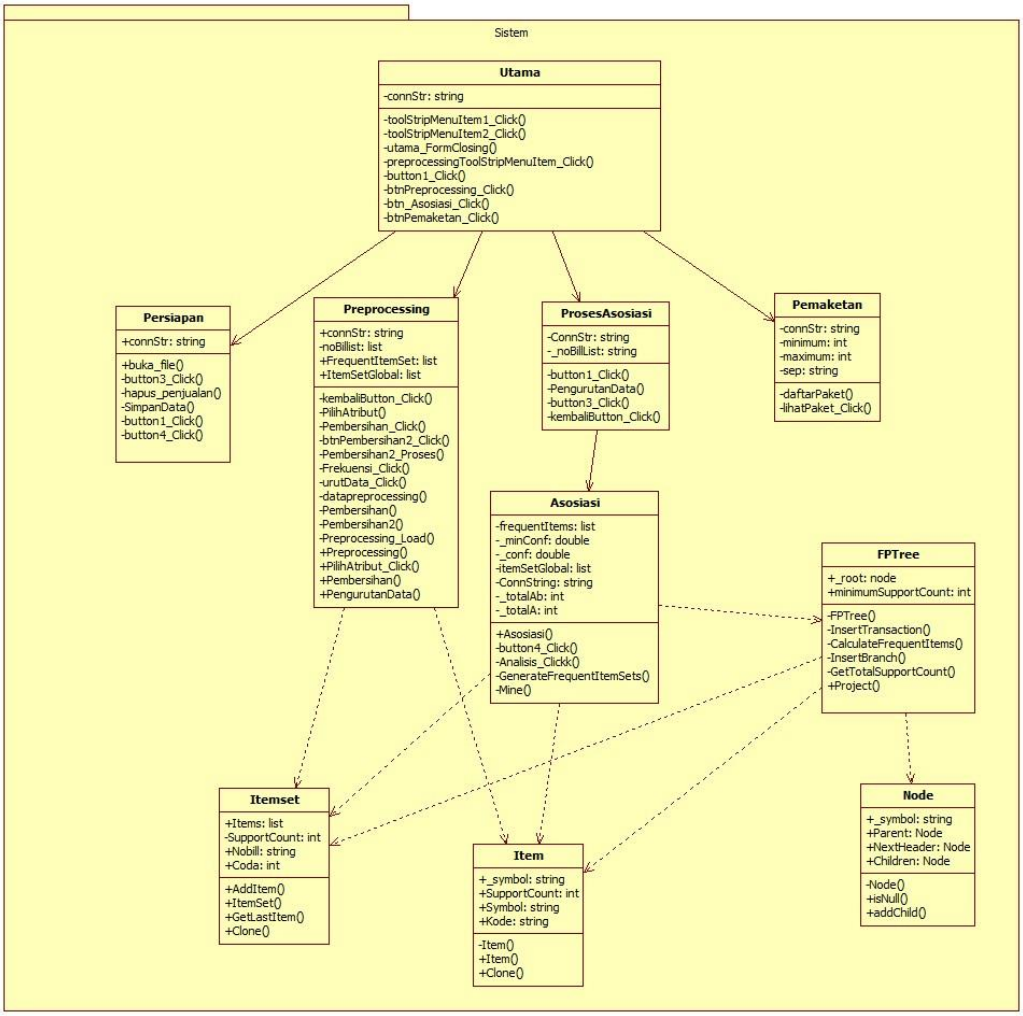
8. *Sequence Diagram* Pembuatan Paket



Gambar 3. 33 *Sequence Diagram* Pembuatan Paket

3.2.4 *Class Diagram*

Berikut adalah *class diagram* dari sistem yang akan dibangun. Dapat dilihat seperti Gambar 3.34 berikut ini:



Gambar 3. 34 Class Diagram

3.3 Analisis Kebutuhan Non Fungsional

Pada sub bab ini, akan diuraikan analisis kebutuhan non fungsional yang terdiri dari analisis kebutuhan perangkat keras, analisis kebutuhan perangkat lunak, dan analisis kebutuhan pengguna.

3.3.1 Analisis Kebutuhan Perangkat Keras

Untuk menjalankan sistem pemaketan sembako dan kebutuhan harian, maka diperlukan perangkat keras yang dapat mendukung proses kerja dari sistem itu sendiri. Pada dasarnya, aplikasi ini berbasis *desktop* yang berarti hanya bisa dijalankan di PC. Berikut adalah spesifikasinya dapat dilihat pada Tabel 3.32:

Tabel 3. 32 Kebutuhan Perangkat Keras

No	Spesifikasi	Keterangan
1	RAM 1GB	Untuk penyimpanan data dan menyelesaikan sebuah perintah
2	Harddisk 250 GB	Untuk tempat penyimpanan data
3	Monitor LCD 14”	Untuk menampilkan tampilan sistem
4	Processor Intel Dual Core 2.93 GHz	Untuk mengubah sinyal digital dari komputer

3.3.2 Analisis Kebutuhan Perangkat Lunak

Perangkat lunak atau *software* merupakan hal yang terpenting dalam mendukung kinerja sebuah sistem. Perangkat lunak harus digunakan dalam sebuah sistem merupakan perintah-perintah yang diberikan kepada perangkat keras agar dapat saling berinteraksi diantara keduanya. Perangkat lunak yang dibutuhkan untuk menjalankan aplikasi sistem pemaketan sembako dan kebutuhan harian dapat dilihat pada Tabel 3.33:

Tabel 3. 33 Analisis Kebutuhan Perangkat Lunak

No	Spesifikasi	Keterangan
1	Visual Studio 2012	Untuk membuat kode
2	Php MyAdmin	Untuk membuat database
3	XAMPP	Untuk koneksi ke database

3.3.3 Analisis Kebutuhan Pengguna

Berikut ini dapat dilihat pada Tabel 3.34 analisis kebutuhan pengguna untuk aplikasi sistem pemaketan sembako dan kebutuhan harian

Tabel 3. 34 Analisis Kebutuhan Pengguna

No	Pengguna Sistem	Hak Akses	Tingkat Keterampilan yang Harus Dimiliki	Pengalaman yang Harus Dimiliki	Jenis Pelatihan yang Akan Diberikan
1	KOMURNIKKOP	Memasukan data dan mengecek pemaketan	Mengerti serta dapat menggunakan komputer	Mengoperasikan komputer	Cara menggunakan aplikasi

3.4 Perancangan Sistem

Perancangan adalah sebuah proses yang menghasilkan sebuah model yang memberi kemudahan kepada penggunanya atau *user*.

3.4.1 Perancangan *Class*

Berikut adalah hasil perancangan *class* pada sistem pemaketan di Primer Koperasi Kartika Baja Cilegon:

Tabel 3. 35 Perancangan *Class* Utama

Nama *class*: Utama **Jenis *class: views***		
Nama	**Jenis**	***Type***
connStr	atribut	string
toolStripMenuItem1_Click	method	Private
toolStripMenuItem2_Click	method	Private
utama_FormClosing	method	Private
preprocessingToolStripMenuItem_Click	method	Private
button1_Click	method	Private
btnPreprocessing_Click	method	Private
btn_Asosiasi_Click	method	Private
btnPemaketan_Click	method	Private

Tabel 3. 36 Perancangan *Class* Persiapan

Nama *class*: Persiapan **Jenis *class: views***		
Nama	**Jenis**	***Type***
connStr	atribut	string
buka_file	method	Public
button3_Click	method	Private
hapus_penjualan	method	Private
SimpanData	method	Private
button1_Click	method	Private
button4_Click	method	Private

Tabel 3. 37 Perancangan *Class Preprocessing*

Nama *class*: Preprocessing **Jenis *class: views***		
Nama	**Jenis**	***Type***
connStr	atribut	string
noBillist	atribut	list
FrequentItemSet	atribut	list
ItemSetGlobal	atribut	list
kembaliButton_Click	method	Private
PilihAtribut_Click	method	Private
Pembersihan_Click	method	Private
btnPembersihan2_Click	method	Private
Pembersihan2_Proses	method	Private
Frekuensi_Click	method	Private
urutData_Click	method	Private
datapreprocessing	method	Private
Pembersihan	method	Privae
Pembersihan2	method	Private
Prepocessing_Load	method	Private
Preprocessing	method	Public
PilihAtribut_Click	method	Public
Pembersihan	method	Public
PengurutanData	method	Public

Tabel 3. 38 Perancangan *Class Item*

Nama *class*: Item **Jenis *class: views***		
Nama	**Jenis**	***Type***
_symbol	atribut	string
SupportCount	atribut	int
Symbol	atribut	string
Kode	atribut	string
Item	method	Public
Clone	method	Public

Tabel 3. 39 Perancangan *Class Itemset*

Nama *class*: Itemset **Jenis *class: views***		
Nama	**Jenis**	***Type***
Items	atribut	list
SupportCount	atribut	int
Nobill	atribut	string
Coda	atribut	int
AddItem	method	Public
ItemSet	method	Public
GetLastItem	method	Public
Clone	method	Public

Tabel 3. 40 Perancangan *Class* Proses Asosiasi

Nama *class*: ProsesAsosiasi **Jenis *class: views***		
Nama	**Jenis**	***Type***
ConnStr	atribut	string
_noBillList	atribut	string
button1_Click	method	Private
PengurutanData	method	Private
button3_Click	method	Private
kembaliButton_Click	method	Private

Tabel 3. 41 Perancangan *Class* Asosiasi

Nama *class*: Asosiasi **Jenis *class: views***		
Nama	**Jenis**	***Type***
frequentItem	atribut	list
_minConf	atribut	double
_conf	atribut	double
ItemSetGlobal	atribut	list
ConnString	atribut	string
_totalAb	atribut	int
_totalA	atribut	int
Asosiasi	method	Public
button4_Click	method	Private
Analisis_Click	method	Private
GenerateFrequentItemset	method	Private
Mine	method	Private

Tabel 3. 42 Perancangan *Class Node*

Nama *class*: Node **Jenis *class: views***		
Nama	**Jenis**	***Type***
_symbol	atribut	string
Parent	atribut	node
NextHeader	atribut	node

Nama *class*: Node **Jenis *class: views***		
Nama	**Jenis**	***Type***
Childern	atribut	node
node	method	Public
isNull	method	Public
addChild	method	Public

Tabel 3. 43 Perancangan *Class FPTree*

Nama *class*: FPTree **Jenis *class: views***		
Nama	**Jenis**	***Type***
_root	atribut	node
minimumSupportCout	atribut	int
FPTree	method	Private
InsertTransaction	method	Private
CalculateFrequetItems	method	Private
InsertBranch	method	Private
GetTotalSupportCount	method	Private
Project	method	Public

Tabel 3. 44 Perancangan *Class* Pemaketan

Nama *class*: Pemaketan **Jenis *class: views***		
Nama	**Jenis**	***Type***
connStr	atribut	string
minimum	atribut	int
maximum	atribut	int
sep	atribut	string
daftarPaket	method	Private
lihatPaket_Click	method	Private

3.4.2 Perancangan Data

Berikut adalah hasil perancangan data pada sistem pemaketan di Primer Koperasi Kartika Baja Cilegon:

3.4.2.1 Perancangan Basis Data

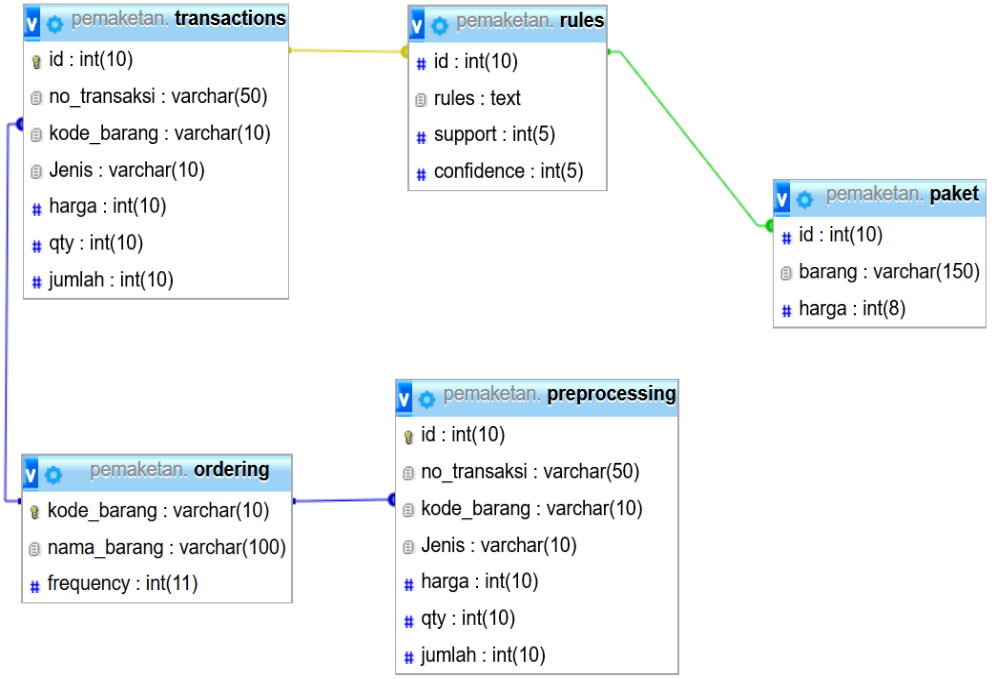
Perancangan basis data adalah merancang kumpulan data yang terhubung dan tersimpan secara bersama-sama. Tabel yang digunakan dalam program ini antara lain:

1. *Ordering*
2. *Preprocessing*

3. *Rules*
4. *Transactions*

3.4.2.2 Skema Relasi

Skema relasi merupakan rangkaian hubungan antara dua tabel atau lebih pada sistem basis data. Berikut adalah skema relasi dari kasus pembangunan aplikasi sistem pemaketan sembako dan kebutuhan harian di Primkop Kartika Baja Cilegon, dapat dilihat pada Gambar 3.35:



Gambar 3. 35 Skema Relasi

3.4.2.3 Struktur Tabel

Struktur tabel dari kasus sistem pemaketan di Primer Koperasi Kartika Baja Cilegon adalah sebagai berikut:

1. ordering

   Nama file: ordering.sql

   Tempat Penyimpanan: *Harddisk*

Tabel 3. 45 Struktur Tabel *Ordering*

Nama Field	Tipe Data	Panjang	Kunci	Keterangan
kode_barang	varchar	10	Primary Key	
nama_barang	varchar	100		
frequency	int	11		

2. preprocessing

   Nama file: preprocessing.sql

   Tempat Penyimpanan: *Harddisk*

Tabel 3. 46 Struktur Tabel *Preprocessing*

Nama Field	Tipe Data	Panjang	Kunci	Keterangan
id	int	10	Primary Key	
no_transaksi	varchar	50		
kode_barang	varchar	10	Foreign Key references ordering (‘kode_barang’)	
Jenis	varchar	10		
harga	int	10		
qty	int	10		
jumlah	int	10		

3. transactions

   Nama file: transactions.sql

   Tempat Penyimpanan: *Harddisk*

Tabel 3. 47 Struktur Tabel *transactions*

Nama Field	Tipe Data	Panjang	Kunci	Keterangan
id	int	10	Primary Key	
no_transaksi	varchar	50		
kode_barang	varchar	10	Foreign Key references ordering (‘kode_barang’)	
Jenis	varchar	10		

Nama Field	Tipe Data	Panjang	Kunci	Keterangan
harga	int	10		
qty	int	10		
jumlah	int	10		

4. rules

   Nama file: rules.sql

   Tempat Penyimpanan: *Harddisk*

Tabel 3. 48 Struktur Tabel *Rules*

Nama Field	Tipe Data	Panjang	Kunci	Keterangan
id	int	10	Primary Key	
rules	text			
support	int	5		
cofidence	int	5		

5. paket

   Nama file: paket.sql

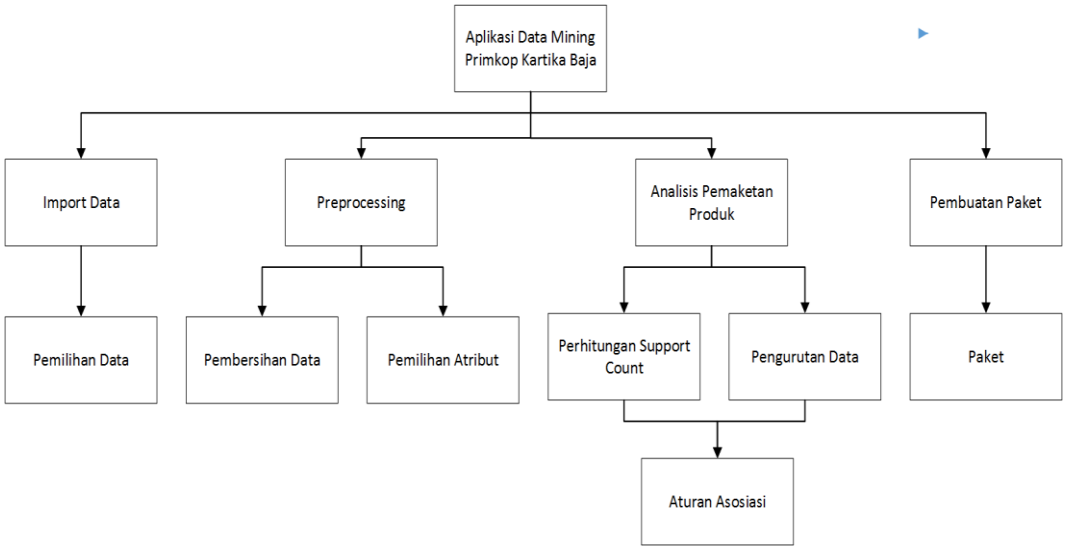
   Tempat Penyimpanan: Harddisk

Tabel 3. 49 Struktur Tabel Paket

Nama Field	Tipe Data	Panjang	Kunci	Keterangan
id_paket	int	10	Primary Key	
barang	varchar	150		
harga	int	8		

3.4.3 Perancangan Struktur Menu

Berikut ini adalah arsitektur struktur menu pada sistem pemaketan sembako dan kebutuhan harian Primkop Kartika Baja, dapat dilihat pada gambar 3.36:

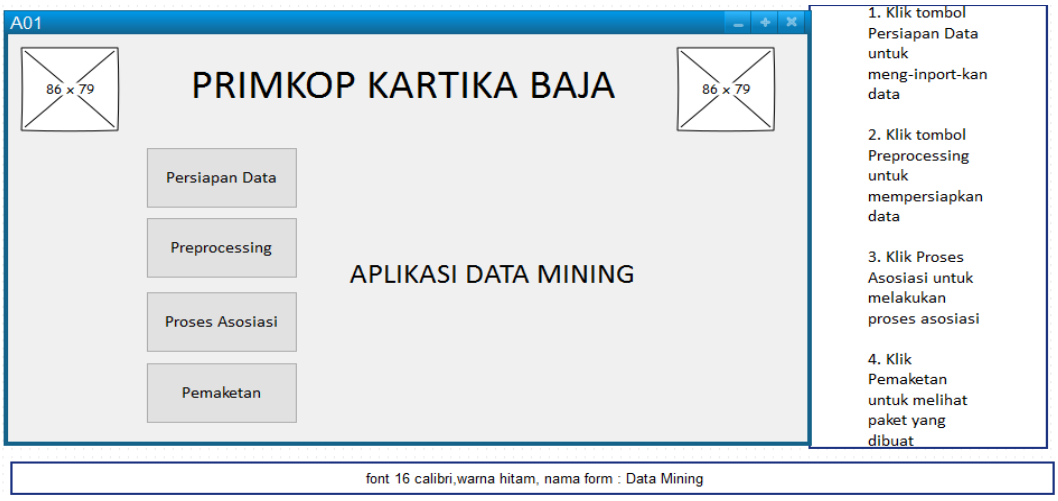


Gambar 3. 36 Perancangan Struktur Menu

3.4.4 Perancangan Antar Muka

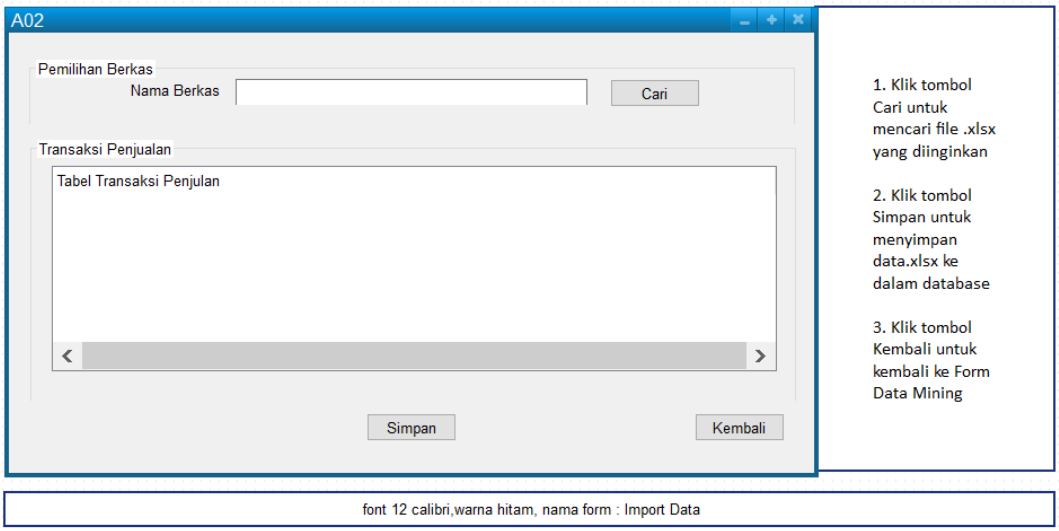
Berikut ini adalah antarmuka pada sistem pemaketan sembako dan kebutuhan harian Primkop Kartika Baja:

1. Halaman ini adalah halaman utama yang dapat memilih persiapan data, *preprocessing*, proses asosiasi, dan pemaketan. Dapat dilihat pada gambar 3.37:



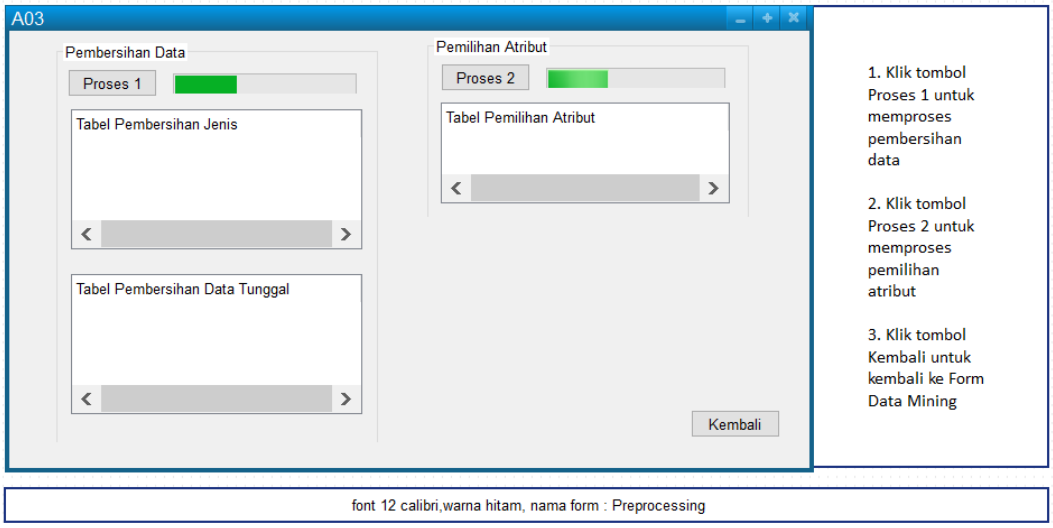
Gambar 3. 37 Halaman Utama Sistem

2. Klik cari untuk mencari file .xlsx yang diinginkan, lalu klik simpan untuk disimpan ke *database*. Dapat dilihat tampilannya pada gambar 3.38:



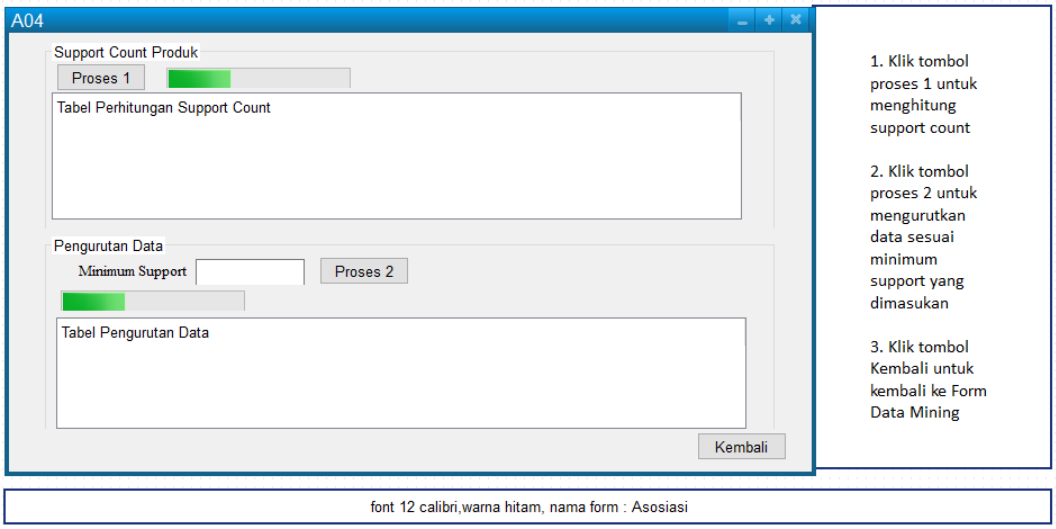
Gambar 3. 38 Import Data

3. Klik proses untuk memproses pembersihan data dan pemilihan atribut. Dapat dilihat pada gambar 3.39 :



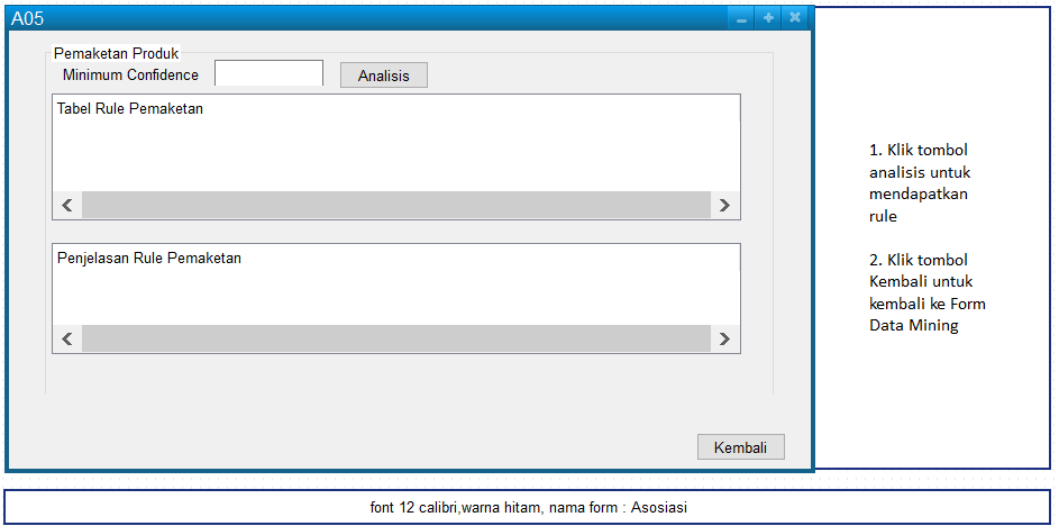
Gambar 3. 39 Preprocessing

4. Klik tombol proses di bagian *support count*, lalu isi *minimum support* dan pilih tombol proses di bagian pengurutan data. Dapat dilihat pada gambar 3.40:



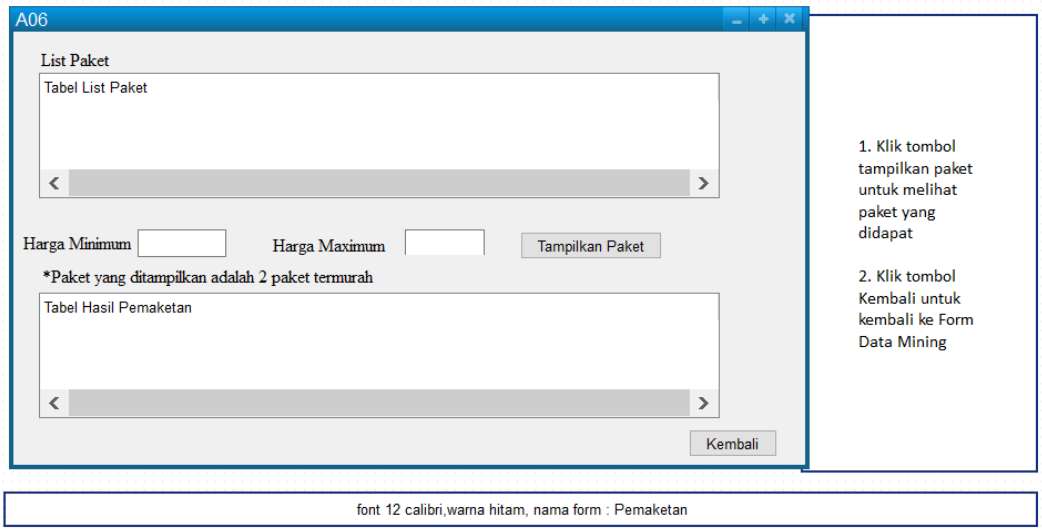
Gambar 3. 40 Proses Asosiasi

5. Isi *minimum confidence* lalu klik tombol analisis untuk memulai menganalisis data. Dapat dilihat pada gambar 3.41:



Gambar 3. 41 Asosiasi

6. Klik tombol tampilkan paket untuk menampilkan paket yang terbentuk. Dapat dilihat pada gambar 3.42:

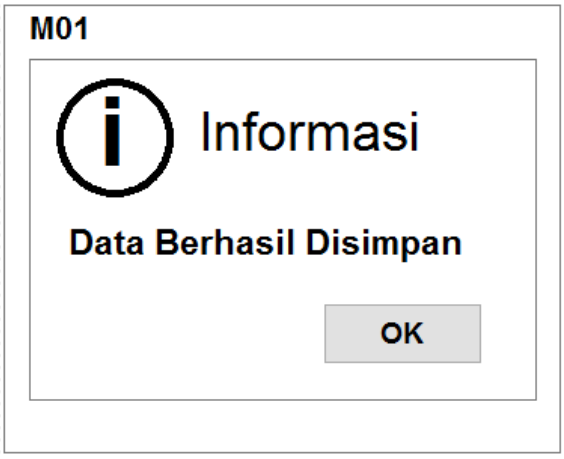


Gambar 3. 42 Pemaketan

3.4.5 Perancangan Pesan

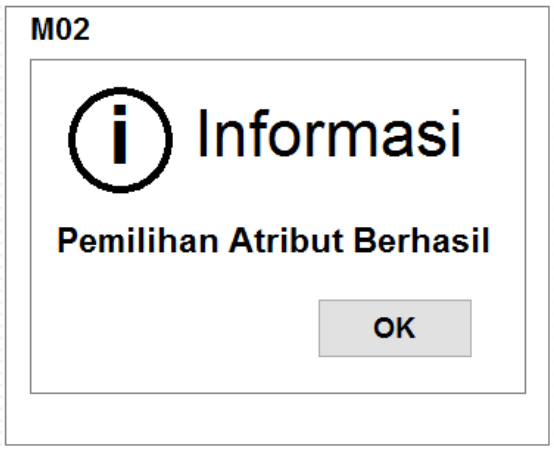
Berikut adalah perancangan pesan pada sistem pemaketan sembako dan kebutuhan harian di Primkop Kartika Baja Cilegon. Pada aplikasi ini, terdapat 5 tampilan pesan diantaranya sebagai berikut:

1. Pesan apabila *user* berhasil melakukan prosen penyimpanan data ke *database*, dapat dilihat pada gambar 3.43:



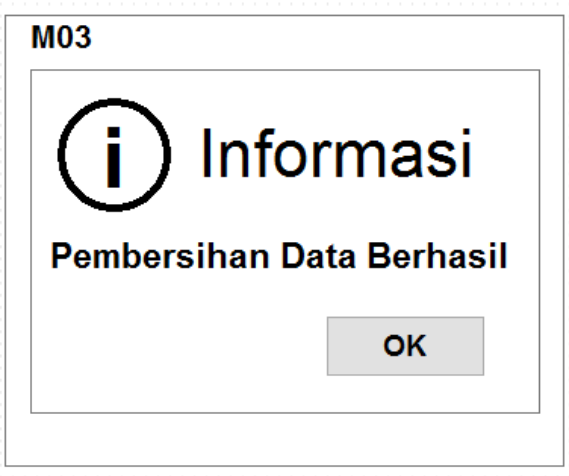
Gambar 3. 43 Pesan Data Berhasil Disimpan

2. Pesan apabila *user* berhasil melakukan proses pemilihan atribut, dapat dilihat pada gambar 3.44:



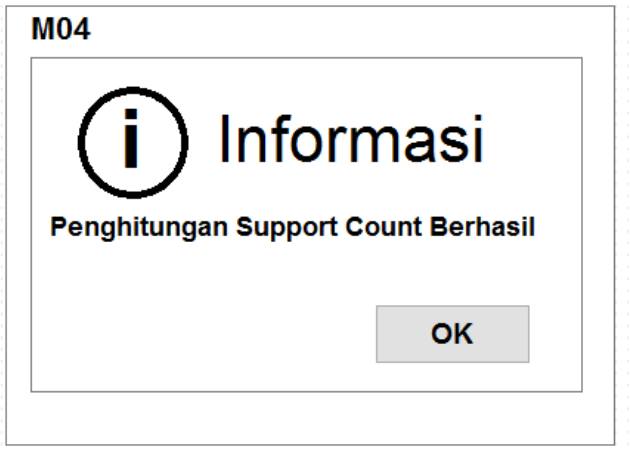
Gambar 3. 44 Atribut Berhasil Dipilih

3. Pesan apabila *user* berhasil melakukan proses pembersihan pada data transaksi tunggal, dapat dilihat pada gambar 3.45:



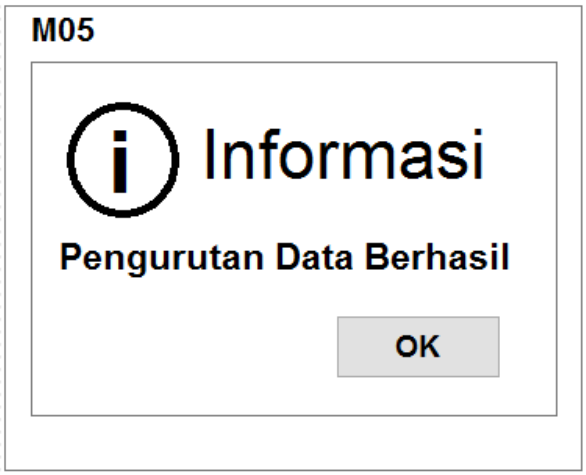
Gambar 3. 45 Data Transaksi Tunggal Berhasil Dibersihkan

4. Pesan apabila *user* berhasil melakukan proses perhitungan *support count* setiap kode barang, dapat dilihat pada gambar 3.46:



Gambar 3. 46 Perhitungan *Support Count* Berhasil

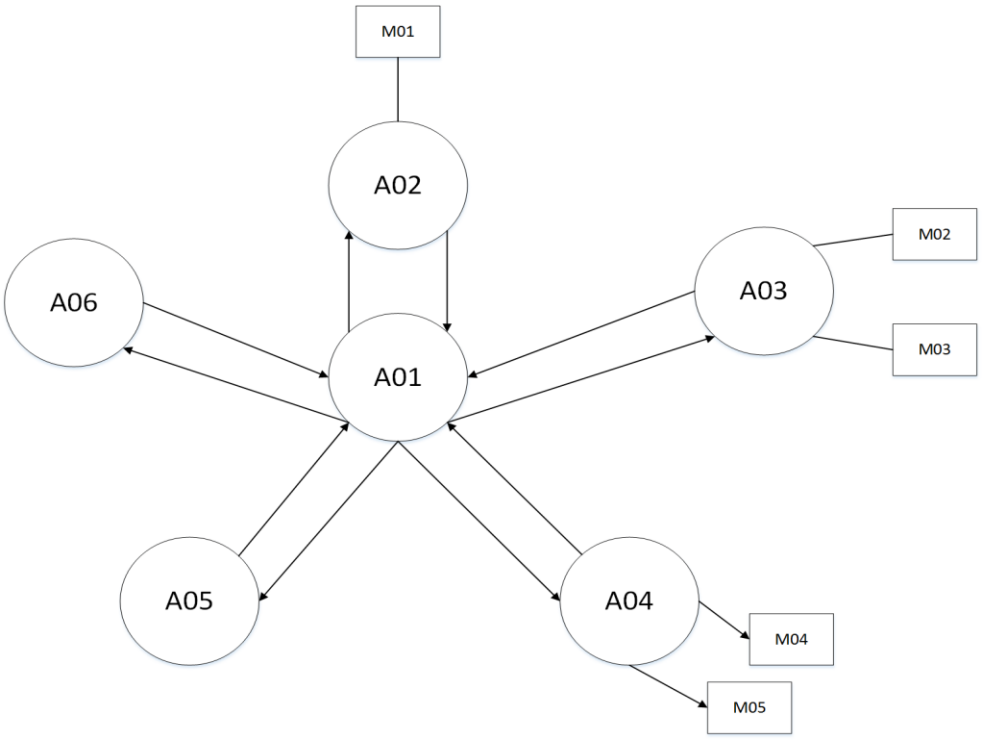
5. Pesan apabila *user* berhasil melakukan proses pengurutkan data, dapat dilihat pada gambar 3.47:



Gambar 3. 47 Data Berhasil Diurutkan

3.4.6 Jaringan Semantik

Jaringan semantik dibuat bertujuan untuk mempresentasikan hubungan antara komponen antar muka pada sistem yang akan dibangun. Dapat dilihat seperti gambar 3.48 berikut:



Gambar 3. 48 Jaringan Semantik