BAB III
ANALISIS DAN PERANCANGAN SISTEM

III.1 Analisis Sistem

Analisis sistem adalah penguraian dari suatu sistem informasi yang utuh kedalam bagian-bagian komponennya dengan maksud untuk mengidentifikasi dan mengevaluasi permasalahan, kesempatan, hambatan yang terjadi dan kebutuhan yang diharapkan sehingga dapat diusulkan perbaikan. Tahap analisis sistem merupakan tahapan yang sangat kritis dan penting, karena kesalahan di dalam tahap ini akan menyebabkan juga kesalahan di tahap selanjutnya. Tugas utama analisis sistem dalam tahap ini adalah menemukan kelemahan-kelemahan dari sistem yang berjalan sehingga dapat diusulkan perbaikannya. Dalam analisa sistem ini meliputi beberapa bagian, yaitu :

1. Analisis Masalah
2. Analisis *Data Mining*
3. Analisis Kebutuhan *Non-Fungsional*
4. Analisis Kebutuhan *Fungsional*

III.2 Analisis Masalah

Adapun analisis masalah di Kalvin Sock Production adalah pemrosesan data membutuhkan waktu yang cukup lama, hal tersebut disebabkan karena banyaknya data transaksi penjualan dan algoritma yang telah diterapkan pada aplikasi sebelumnya yaitu algoritma apriori harus melakukan *scan database* setiap kali iterasinya dalam menemukan jenis kaos kaki yang sering dipesan oleh para konsumennya.

III.3 Analisis Data Mining

Berdasarkan tahapan-tahapan yang terdapat dalam CRIPS-DM, maka dalam penelitian *data mining* di Kalvin Socks Production ini terdapat beberapa tahapan sebagai berikut.

III.3.1 Pemahaman Bisnis

Tahapan pemahaman bisnis merupakan tahapan pertama yang dilakukan dalam kerangka kerja CRISP-DM. dalam tahapan bisnis ini terdapat beberapa tahapan lainnya, yaitu :

1. Identifikasi Tujuan Bisnis

   Dalam proses bisnisnya Kalvin Socks Production memasarkan berbagai jenis kaos kakinya langsung kepada konsumen dan kepada agen *retail*. Untuk membuat strategi bisnis perusahaan, pihak Kalvin Socks Production harus menganalisa data penjualan. Tujuan dari menganalisa data penjualan yakni untuk mengetahui kecenderungan pemilihan jenis kaos kaki yang sering dibeli dengan cara mengetahui pola pemesanan dari konsumen agar jenis kaos kaki yang diproduksi tepat sasaran.

2. Penentuan Sasaran *Data Mining*

   Tujuan *data mining* untuk perusahaan adalah menggali pengetahuan (*discovering knowledge*) tentang pola (*pattern*) jenis kaos kaki apa saja yang sering dibeli oleh konsumen sehingga diketahui jenis kaos kaki mana saja yang sering dipesan komsumen.

III.3.2 Pemahaman Data

Tahapan pemahaman data merupakan tahapan kedua yang dilakukan setelah tahapan bisnis dalam kerangka kerja CRISP-DM. dalam tahap pemahaman data ini terdapat beberapa langkah diantaranya:

a. Pengumpulan data awal

   Data yang digunakan dalam penelitian ini yakni data transaksi penjualan produk kaos kaki pada bulan Agusuts 2015. Data yang digunakan untuk proses analisis berupa file Excel dengan format *.csv atau *.xlsx.

Data yang digunakan untuk proses perhitungan secara manual dengan mengambil sebanyak 178 transaksi untuk dilakukan pengolahan ke tahap selanjutnya. Adapun detail informasi mengenai data transaksi yang digunakan dapat dilihat dalam table III.1 dibawah ini.

Tabel III. 1 Struktur Data Transaksi Penjualan

DOKUMENT	**KETERANGAN**		
Detail Data Transaksi Penjualan	Deskripsi	Data ini berisi mengenai transaksi penjualan pada bulan Agustus 2015 di Kalvin Socks Production	
	Fungsi	Untuk mengatahui jenis kaos kaki apa saja yang sering dipesan oleh konsumen pada bulan Agustus	
	Format	Microsoft excel (.xlsx)	
	Atribut	Tanggal	Waktu transaksi
		NoNota	Nomor komsumen pada saat transaksi
		kodeBarang	Kode dari jenis kaos kaki
		NamaBarang	Kode dari nama kaos kaki
		Jumlah	Jumlah pemesanan kaos kaki

Berikut adalah contoh data transaksi penjualan pada bulan Agustus yang ada di Kalvin Socks Production yang akan digunakan untuk proses perhitungan manual dengan metode *association rule* dan menggunakan algoritma *Improved Apriori*, dimana atribut-atribut yang terdapat dalam data penjualan tersebut antara lain Tanggal, NoNota, kodeBarang, NamaBarang, Jumlah. Untuk lebih jelasnya, dapat dilihat pada lampiran tabel D-1 di lembar lampiran.

III.3.3 Preprocessing Data

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

1. Ekstraksi data

   Ekstraksi data dapat diartikan sebagai proses pengambilan data dari sumber data dalam rangka untuk melanjutkan proses pengolahan data ke tingkat selanjutnya ataupun untuk menyimpan data hasil ekstrak tersebut.

Dalam penelitian ini, data yang berasal dari *flat file* berupa *microsoft excel* (.xlsx) diekstrak, kemudian disimpan ke dalam sebuah *database* agar memudahkan dalam proses pengolahan data.

2. Pemilihan atribut (*artibut selection)*

   Pemilihan atribut adalah proses pemilihan atribut data yang akan digunakan, sehingga data tersebut dapat diproses sesuai dengan kebutuhan *data mining*. Dalam penelitian ini, atribut yang akan digunakan yakni NoNota dan kodeBarang. Pada NoNota berguna untuk membedakan antar konsumen sedangkan kodebarang berguna untuk mengetahui jenis kaos kaki apa saja yang dibeli oleh konsumen dalam satu transaksi. Tahapan pemilihan atribut ini sama seperti penelitian sebelumnya[1]. Data pemilihan atribut dapat dilihat pada tabel D-2 dalam lampiran D.

3. Pembersihan data (*data cleaning*)

   Pada tahap pembersihan data, hasil pemilihan atribut pada tabel D-2 akan dibersihkan dari NoNota yang hanya memiliki Kodebarang tunggal. Artinya pada satu transaksi konsumen hanya membeli satu jenis kaos kaki saja. Data transaksi yang memiliki kodeBarang tunggal ini tidak memiliki hubungan asosiasi dengan kodeBarang lain yang sudah dibeli. Tahapan pemilihan atribut ini sama seperti penelitian sebelumnya[1]. Hasil pembersihan data dapat dilihat pada tabel D-3 dalam lampiran D.

III.3.4 Pemodelan

Pemodelan merupakan tahap pembuatan model yang akan digunakan dalam proses *data mining*. Tahap ini dilakukan dengan menggunakan metode *association rule* yang akan dibagi ke dalam dua tahap yaitu :

a. *Frequent Itemset*

Dalam tahapan mencari *frequent itemset* akan menggunakan algoritma *improved apriori*. Langkah-langkah proses pengerjaan algoritma *improved apriori* dalam penelitian ini adalah sebagai berikut :

1. Asumsi nilai *minimum support* yang akan digunakan adalah 7% dari total transakasi yaitu 10,5 ≈ 10. Nilai tersebut diambil agar mendapatkan 3 kombinasi *item*.
2. Nilai *minimum confidence* digunakan untuk menyakinkan kuatnya hubungan antara *item* yang satu dengan yang lainnya. Asumsi nilai *minimum confidence* yang akan digunakan dalam perhitungan manual sebesar 60%.
3. Data yang sudah melewati tahap *preprocessing* kemudian diubah kedalam bentuk *matrix,* kemudian isi bagian baris dengan NoNota dan isi bagian kolom dengan kodeBarang. Jika pada suatu NoNota terdapat kodeBarang maka masukan *value* 1 pada kolom kodeBarang di NoNota tersebut, sedangkan jika tidak terdapat kodeBarang pada suatu NoNota maka masukan *value* 0 pada kolom kodeBarang di NoNota tersebut. Hasil *transformasi* ke dalam bentuk *matrix* dapat dilihat pada tabel D-4 dalam lampiran D.
4. Data yang sudah dalam bentuk *matrix* kemudian dihitung untuk mencari *frequent item*-nya, Caranya yaitu jumlahkan tiap kodeBarang pada bagian kolom untuk mencari nilai *support count* dari setiap kodeBarangnya. Hasilnya dapat dilihat pada tabel D-5 dalam lampiran D.
5. Dari data tabel D-5 dalam lampiran D dapat diketahui *kandidat frequent 1-itemset*. Jumlahkan semua nilai yang bernilai 1 pada setiap kolom dan barisnya. Jumlah nilai pada kolom adalah *Support Count* sedangkan jumlah nilai pada baris adalah *Count.* Hasilnya dapat dilihat pada tabel D-6 dalam lampiran D.
6. Dari tabel D-6 kemudian cari *frequent 1-itemset*-nya. Hapus kodeBarang yang jumlah nilai *Support Count*-nya kurang dari *minimum support* dan hapus NoNota yang jumlah nilai *Count*-nya $\leq$ 1. Hasilnya dapat dilihat pada tabel D-7 dalam lampiran D.
7. Setelah *frequent 1-itemset* didapatkan maka lakukan *cross product* untuk mendapatkan 2 kombinasi *item,* lalu setelah melakukan *cross product* hitung nilai *support*-nya menggunakan operan *AND*. Setelah didapatkan nilai

support-nya lalu jumlahkan tiap baris dan kolomnya. Hasilnya dapat dilihat pada tabel D-8

8. Dari tabel D-8 kemudian cari *frequent 2-itemset*-nya. Hapus kodeBarang yang jumlah nilai *Support Count*-nya kurang dari *minimum support* dan hapus NoNota yang jumlah nilai *Count*-nya $\leq$ 2. Hasilnya dapat dilihat pada tabel D-9 dalam lampiran D.
9. Setelah *frequent 2-itemset* didapatkan maka lakukan *cross product* lagi untuk mendapatkan 3 kombinasi *item,* lalu setelah melakukan *cross product* hitung nilai *support*-nya menggunakan operan *AND*. Setelah didapatkan nilai *support*-nya lalu jumlahkan tiap baris dan kolomnya. Hasilnya dapat dilihat pada tabel D-10
10. Dari tabel D-10 kemudian cari *frequent 3-itemset*-nya. Hapus kodeBarang yang jumlah nilai *Support Count*-nya kurang dari *minimum support* dan hapus NoNota yang jumlah nilai *Count*-nya $\leq$ 3. Hasilnya dapat dilihat pada tabel D-11 dalam lampiran D.
11. Setelah *frequent 3-itemset* didapatkan maka lakukan *cross product* lagi untuk mendapatkan 4 kombinasi *item,* lalu setelah melakukan *cross product* hitung nilai *support*-nya menggunakan operan *AND*. Setelah didapatkan nilai *support*-nya lalu jumlahkan tiap baris dan kolomnya. Hasilnya dapat dilihat pada tabel D-12
12. Dari tabel D-12 kemudian cari *frequent 4-itemset*-nya. Hapus kodeBarang yang jumlah nilai *Support Count*-nya kurang dari *minimum support* dan hapus NoNota yang jumlah nilai *Count*-nya $\leq$ 4. Hasilnya dapat dilihat pada tabel D-13 dalam lampiran D.
13. Dikarenakan pada data tabel D-13 dalam lampiran D kosong atau tidak memiliki data yang memenuhi nilai *minimum support,* maka proses untuk pembentukan *frequent itemset* selanjutnya terhenti.

b. *Generate Rule*

Setelah mencari dan mendaptkan *frequent itemset-nya*, lalu dibentuk aturan asosiasi untuk mencari *rule* yang memenuhi nilai *minimum confidence*. Berikut adalah tahapannya:

1. Karena hanya mendapatkan *frequent itemset*-nya sampai 3 maka lakukan pembentukan rule pada {B17,B18,B19}, {B18,B19,B20}, {B14,B18}, {B14,B19}, {B15,B18}, {B16,B18}, {B17,B18}, {B17,B19}, {B17,B20}, {B18,B19}, {B18,B20}, {B18,B29}, {B19,B20}, {B20,B29} kemudian melakukan perhitungan nilai *confidence*-nya. Hasilnya dapat dilihat pada tabel III.2

Tabel III. 2 Pembentukan *Rule*

Kombinasi item	Confidence %
B17 ^ B18 → B19	50
B17 ^ B19 → B18	91
B18 ^ B19 → B17	62.5
B17 → B18 ^ B19	37.03
B18→ B17 ^ B19	35.71
B19 → B17 ^ B18	28.57
B18 ^ B19 → B20	68.75
B18 ^ B20 → B19	61.11
B19 ^ B20 → B18	73.33
B18 → B19 ^ B20	28.94
B19→ B18 ^ B20	31.42
B20 → B18 ^ B19	32.35
B14 → B18	41.66666667
B14 → B19	50
B15 → B18	68.75
B16 → B18	57.89473684
B17 → B18	74.07407407
B17 → B19	31.42857143

Kombinasi item	Confidence %
B17 → B20	40.74074074
B18 → B19	42.10526316
B18 → B20	47.36842105
B18 → B29	26.31578947
B19 → B20	42.85714286
B20 → B29	32.35294118
B18 → B14	26.31578947
B19 → B14	34.28571429
B18 → B15	28.94736842
B18 → B16	28.94736842
B18 → B17	52.63157895
B19 → B17	31.42857143
B20 → B17	32.35294118
B19 → B18	45.71428571
B20 → B18	52.94117647
B29 → B18	37.03703704
B20 → B19	44.11764706
B29,B20	40.74074074

2. Dari tabel III.3 kemudian *rule* diseleksi. *Rule* yang nilai *confidence*-nya kurang dari nilai minimum *confidence* dihilangkan. Hasilnya dapat dilihat pada tabel III.3.

Tabel III. 3 *Rule* yang memenuhi nilai *minimum confidence*

Kombinasi item	Confidence %
B17 ∧ B19 → B18	91
B18 ∧ B19 → B17	62.5
B18 ∧ B19 → B20	68.75
B18 ∧ B20 → B19	61.11
B19 ∧ B20 → B18	73.33
B17 → B18	74.07407407
B15 → B18	68.75

3. Rule yang telah memenuhi nilai *minimum confidence* kemudian disesuaikan dengan tabel barang. Sehingga kodebarang yang terdapat pada *rule* sesuai dengan nama barangnya dan dapat dijadikan informasi. Tabel barang dapat dilihat pada tabel III.4

Tabel III. 4 Tabel barang

kodeBarang	namaBarang
B1	ankle nylon pj
B2	ankle nylon tg
B3	ankle nylon pdk
B4	arygle nylon pj
B5	arygle nylon tg
B6	arygle nylon pdk
B7	bed nylon pj
B8	bed nylon tg
B9	bed nylon pdk
B10	slipper nylon pj
B11	slipper nylon tg
B12	slipper nylon pdk
B13	sd h kaoxin
B14	smp h kaoxin
B15	sma h kaoxin
B16	sd hp kaoxin
B17	smp hp kaoxin
B18	sma hp kaoxin

kodeBarang	namaBarang
B27	sma pramuka katun
B28	bayi lurus kids
B29	bayi lipat kids
B30	bayi lipat/lurus anti slip qino
B31	children
B32	strawberry
B33	jempol
B34	handset
B35	jempol telapak (shofa) nylon pj
B36	jempol telapak (shofa) nylon tg
B37	jempol telapak (shofa) nylon pdk
B38	jempol telapak (muslim) polys pj
B39	jempol telapak (muslim) polys tg
B40	jempol telapak (muslim) polys pdk
B41	long mashika p/h/krem
B42	paskibra vinsaya
B43	jempol polos nylon m
B44	mansocks

kodeBarang	namaBarang
B19	sd p kaoxin
B20	smp p kaoxin
B21	sma p kaoxin
B22	mk sd h polos
B23	mk sd hp polos
B24	mk sd p polos
B25	sd pramuka katun
B26	smp pramuka katun

kodeBarang	namaBarang
B45	sport motif / catur polys
B46	sport comp
B47	sport fe
B48	sport long rib vjm
B49	bola dws
B50	bola tg
B51	mk ladys kids
B52	ladys qino
B53	dws thermal

4. Informasi yang dihasilkan berupa kombinasi *item* dan nama jenis kaos kaki berdasarkan kodeBarangnya sesuai pada tabel III.4. Hasilnya dapat dilihat pada tabel III.5

Tabel III. 5 Informasi yang dihasilkan dari *rule*

Kombinasi item	Informasi produk
B17 ∧ B19 → B18	Smp hp kaoxin ∧ Sd p kaoxin → Sma hp kaoxin
B18 ∧ B19 → B17	Sma hp kaoxin ∧ Sd p kaoxin → Smp hp kaoxin
B18 ∧ B19 → B20	Sma hp kaoxin ∧ Sd p kaoxin → Smp p kaoxin
B18 ∧ B20 → B19	Sma hp kaoxin ∧ Smp p kaoxin → Sd p kaoxin
B19 ∧ B20 → B18	Sd p kaoxin ∧ Smp p kaoxin → Sma hp kaoxin
B17 → B18	Smp hp kaoxin → Sma hp kaoxin
B15 → B18	Sma h kaoxin → Sma hp kaoxin

5. Dari tabel informasi yang dihasilkan dari *rule* maka dijadikan informasi dalam merekomendasikan jenis kaos kaki apa saja yang akan diproduksi lebih dibulan selanjutnya oleh pihak KALVIN SOCKS PRODUCTION akan tetapi tidak berikut dengan jumlahnya karena yang akan menentukan jumlah atau banyaknya adalah pihak KALVIN SOCKS PRODUCTION yang menentukannya. Berikut adalah informasi rekomendasi jenis kaos kaki yang akan diproduksi lebih dibulan selanjutnya :

Smp h kaoxin
Smp hp kaoxin
Sma hp kaoxin
Sd p kaoxin
Smp p kaoxin

III.4 Analisis Kebutuhan Non-Fungsional

Analisis kebutuhan *non* fungsional dilakukan untuk mengetahui spesifikasi kebutuhan sistem. Spesifikasi kebutuhan melibatkan analisis perangkat keras (*hardware*), analisis perangkat lunak (*software*), dan analisis pengguna (*user*).

III.4.1 Analisis Kebutuhan Perangkat Keras

Perangkat keras pendukung dalam pembangunan perangkat lunak ini adalah sebagai berikut :

1. Analisis spesifikasi kebutuhan perangkat keras pada sistem yang sedang berjalan di Kalvin Socks Production adalah :
   a. *Processor* : Intel Core Duo
   b. *Harddisk* : 250 GB
   c. *Memory* : 2GB
   d. *Monitor* : 14”
   e. *Keyboard* : *standard*
   f. *Mouse* : *optical*
2. Analisis spesifikasi *minimum* kebutuhan perangkat keras pada sistem yang akan dibangun membutuhkan :
   a. Processor berkecepatan minimal 2Ghz
   b. *Hardisk* 1GB untuk penyimpanan data
   c. *Memory* 1 GB
   d. Monitor
   e. *Keyboard*
   f. *Mouse*
3. Evaluasi kebutuhan perangkat keras

   Dari data spesifikasi perangkat keras yang dimiliki oleh Kalvin Socks Production seperti diatas, perangkat keras yang telah dimiliki sudah cukup untuk dapat menjalankan aplikasi pengolahan data penjualan yang akan dibuat.

III.4.2 Analisis Kebutuhan Perangkat Lunak

Kebutuhan perangkat lunak pendukung sangatlah penting bagi pembangunan perangkat lunak yang sedang dirancang. Perangkat lunak pendukung dalam pembangunan perangkat lunak ini adalah sebagai berikut :

1. Analisis kebutuhan perangkat lunak pada sistem yang sedang berjalan di Kalvin Socks Production adalah :
   a. Sistem operasi Windows 7
   b. *Microsoft office*
2. Analisis spesifikasi kebutuhan perangkat lunak untuk menjalankan sistem yang akan dibangun adalah :
   a. Sistem Operasi Windows 7
   b. XAMPP sebagai *database server*
   c. Visual studio 2010
3. Evaluasi kebutuhan perangkat lunak
   Berdasarkan analisis spesifikasi perangkat lunak yang ada di Kalvin Socks Production, spesifikasi perangkat lunak yang akan digunakan sudah memadai

III.4.3 Analisis *User* (Pengguna)

Analisis *user* dibuat untuk mengetahui siapa saja dan seperti apa saja karakteristik yang dimiliki oleh orang yang dapat menggunakan aplikasi yang akan dibuat. Untuk lebih jelas mengenai karakteristik pengguna, dapat dilihat pada tabel III.6

Tabel III. 6 Karakteristik Pengguna

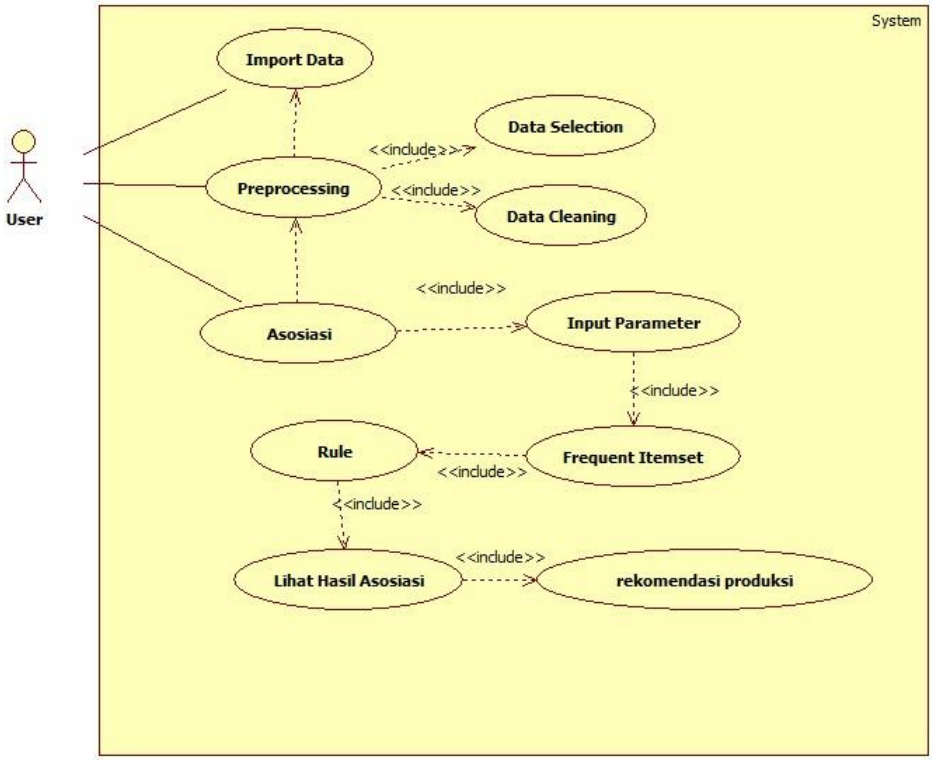
Pengguna	Tanggung jawab	Hak akses	Tingkat Pendidikan	Tingkat Keterampilan	Pengalaman
User	mengoperasikan aplikasi sesuai dengan tugas yang diberikan	Melakukan kegiatan yang berhubungan dengan isi tampilan yang ada pada aplikasi	Minimal Sekolah Menengah Atas (SMA)	Memahami konten yang tersedia pda program aplikasi	Minimal mampu mengoperasikan komputer

III.5 Analisis Kebutuhan Fungsional

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

III.5.1 Diagram Use Case

Use case atau diagram *use case* merupakan pemodelan untuk kelakuan (behavior) sistem yang akan dibuat. Diagram *use case* yang terdapat pada sistem yang akan dibangun terdiri dari satu *user* dan sepuuluh *use case*. Diagram *use case* dapat dilihat pada gambar III.1 Dan penjelasan *user* dan *use case* dapat dilihat pada Tabel III.7 dan Tabel III.8



Gambar III. 1 *Diagram Use Case* Pada Sistem Data Mining di Kalvin Socks Production

Tabel III. 7 Definisi *User*

User	**Deskripsi**
User	Bertugas untuk melakukan *import* data, menginputkan nilai *minimum support* dan *minimum confidence*.

Tabel III. 8 Deskripsi *Use Case*

No	**Use Case**	**Deskripsi**
1	*Import* Data	Sistem mengharuskan *user* yang belum melakukan *import* data untuk melakukan *import* data terlebih dahulu lalu menyimpan data data tersebut kedalam *database*
2	*Preprocessing*	Data yang sudah berada pada *database* kemudian dilakukan proses data *selection* dan data *cleaning*
3	*Data Selection*	Sistem melakukan pemilihan atribut NoNota dan kodeBarang jika *user* sudah melakukan *import* data
4	*Data Cleaning*	Sistem melakukan pembersihan data pada NoNota yang mengandung kodeBarang tunggal jika *user* sudah melakukan *import* data dan data *selection*
5	Proses Asosiasi	Sistem melakukan proses pencarian *frequent itemset*, *rule*, melihat hasil asosiasi dan menampilkan informasi. Sedangkan *user* menginputkan nilai minimum *support* dan *confidence* sebagai parameter.
6	*Input* parameter	*User* menginputkan nilai minimum *support* dan minimum *confidence* sebagai parameter
7	*Frequent itemset*	Sistem melakukan pencarian kombinasi kaos kaki sesuai dengan nilai *support* yang di*input*kan oleh *user*
8	*Rule*	Sistem membuat *rule* sesuai kombinasi kaos kaki yang telah didapatkan pada *frequent itemset*. Lalu *rule* diseksi sesuai nilai minimum *confidence* yang di*input*kan oleh *user*
9	Lihat hasil asosiasi	*Sistem* menampilkan kombinasi jenis kaos kaki yang memenuhi nilai minimum *support* dan minimum *confidence*.
10	Rekomendasi produksi	Menampilkan informasi jenis kaos kaki apa saja yang akan diproduksi lebih dibulan selanjutnya

III.3.2.1 Skenario *Use Case*

Skenario *use case* menjelaskan skenario dari setiap proses yang digambarkan pada diagram *use case*. Berikut ini skenario *use case* dari gambar III.1.

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. Skenario *use case import* data dapat dilihat pada tabel III. 9 Requirement A.1 dan tabel III.10. Skenario Use Case Import Data.

Tabel III. 9 *Requirement* A.1

Requirement A.1
Sistem menyediakan fasilitas *import data* untuk melakukan *import* data terlebih dahulu sebelum melakukan *preprocessing* dan proses asosiasi

Tabel III. 10 *Skenario Use Case Import Data*

Use case Name	***Import Data***	
Related Requirements	Requirement *A.1*	
Goal In Context	*Import* data transaksi penjualan dan menyimpannya ke dalam *database*	
Precondition	*User* menyiapkan data transaksi	
Successful End Condition	Data transaksi berhasil disimpan ke dalam *database*	
Failed End Condition	Data transaksi tidak berhasil disimpan ke dalam *database*	
Actors	*User*	
Triger	*User* memasukan data transaksi	
Included Cases	-	
Main Flow	Step	Action
	1	*User* memasukan data transaksi penjualan
	2	Sistem melakukan proses validasi format file terhadap masukan
	3	Sistem menampilkan data transaksi
	4	*User* melakukan *request* kepada sistem untuk menyimpan data transaksi kedalam *database*
	5	Sistem melakukan validasi terdapat data
	6	Sistem menyimpan data transaksi kedalam *database*

Extension	Step	Branching Action
	3.1	Menampilkan notifikasi kesalahan terhadap masukan
	6.1	Menampilkan notifikasi kesalahan pada data

2. Skenario *Use Case Preprocesing*

Skenario *use case preprocessing* menjelaskan langkah-langkah *user* terhadap sistem untuk melakukan proses data *selection* dan data *cleaning*. Skenario *use case preprocesing* dapat dilihat pada table III.11 *Requirement A.2* dan tabel III.12 *Skenario Use Case Preprocesing.*

Tabel III. 11 *Requirement* A.2

Requirement A.2
Sistem menyediakan fasilitas *preprocesing* untuk melakukan data *selection* dan data *cleaning*

Tabel III. 12 Skenario *Use Case Preprocesing*

Use case Name	***Preprocesing***	
Related Requirements	Requirement *A.1*, Requirement *A.2*	
Goal In Context	Sistem melakukan proses data *selection* dan data *cleaning*	
Precondition	Data transaksi telah disimpan didalam *database*	
Successful End Condition	Sistem berhasil melakukan proses data *selection* dan data *cleaning*	
Failed End Condition	Sistem tidak berhasil melakukan proses data *selection* dan data *cleaning*	
Actors	*User*	
Triger	*User* melakukan *request* proses data *selection* dan data *cleaning* kepada sistem	
Included Cases	Data *selection* dan data *cleaning*	
Main Flow	Step	Action
	1	User melakakukan *request* data *selection*
	2	Sistem melakukan proses data *selection*
	3	Sistem menampilkan data hasil data *selection*
	4	User melakakukan *request* data *cleaning*
	5	Sistem melakukan proses data *cleaning*
	6	Sistem menampilkan data hasil data *cleaning*
Extension	Step	Branching Action
	-	-

3. Scenario *Use Case Data Selection*

Skenario *use case data selection* menggambarkan langkah-langkah user terhadap sistem melakukan pemilihan atribut terhadap data transaksi yang telah melewati proses *import data*. Skenario *use case data selection* dapat dilihat pada tabel III.13.

Tabel III. 13 Skenario *Use Case* Data *Selection*

Use case Name	***Data selection***	
Related Requirements	Requirement *A.1,* Requirement *A.2*	
Goal In Context	Sistem melakukan pemilihan atribut NoNota dan kodeBarang	
Precondition	Data transaksi telah disimpan didalam *database*	
Successful End Condition	Sistem berhasil memilih atribut NoNota dan kodeBarang	
Failed End Condition	Sistem tidak berhasil memilih atribut NoNota dan kodeBarang	
Actors	*User*	
Triger	*User* melakukan *request* data *selection* kepada sistem	
Included Cases	-	
Main Flow	Step	Action
	1	User melakukan *request* data *selection* kepada sistem
	2	Sistem melakukan pemilihan atribut NoNota dan kodeBarang
	3	Sistem menyimpan hasil pemilihan atribut kedalam *database*
	4	Sistem menampilkan data hasil pemilihan atribut
Extension	Step	Branching Action
	-	-

4. Skenario *Use Case Data Cleaning*

Skenario *use case data cleaning* menggambarkan langkah-langkah *user* terhadap sistem melakukan pembersihan terhadap NoNota yang hanya mempunya 1 kodeBarang dan telah melewati proses *selection*. Skenario *use case data cleaning* dapat dilihat pada tabel III.14.

Tabel III. 14 Skenario *Use Case* Data *Cleaning*

Use case Name	***Data Cleaning***	
Related Requirements	Requirement *A.2*	
Goal In Context	Sistem melakukan pemberishan pada atribut NoNota yang hanya mempunyai 1 kodeBarang	
Precondition	Sudah melalu proses data *selection*	
Successful End Condition	Sistem berhasil melakukan pemberishan pada atribut NoNota yang hanya mempunyai 1 kodeBarang	
Failed End Condition	Sistem tidak berhasil melakukan pemberishan pada atribut NoNota yang hanya mempunyai 1 kodeBarang	
Actors	*User*	
Triger	*User* melakukan *request* data *cleaning* kepada sistem	
Included Cases	-	
Main Flow	Step	Action
	1	User melakukan *request* data *cleaning* kepada sistem
	2	Sistem melakukan pemberishan pada atribut NoNota yang hanya mempunyai 1 kodeBarang
	3	Sistem menyimpan hasil *cleaning* kedalam *database*
	4	Sistem menampilkan data hasil proses data cleaning
Extension	Step	Branching Action
	-	-

5. Skenario Use Case Proses Asosiasi

Skenario *use case* proses *assosiasi* menggambarkan langkah-langkah aksi *user* terhadap sistem dalam melakukan proses *frequent itemset*, *rule*, dan melihat hasil asosiasi. Skenario *use case* proses asosiasi dapat dilihat pada tabel III.15 Requirement *A.3 dan* tabel III.skenario 16 Use Case *Proses Asosiasi*

Tabel III. 15 *Requirement* A.3

Requirement A.3
Sistem menyediakan fasilitas input parameter untuk mengisi nilai minimum *support* dan minimum *confidence*.

Tabel III. 16 Skenario *Use Case* Proses Asosiasi

Use case Name	**Proses Asosiasi**	
Related Requirements	Requirement *A.1,* Requirement *A.2,* Requirement *A.3*	
Goal In Context	Sistem mendapatkan *rule* lalu menampilkan *rule* tersebut	
Precondition	telah melakukan proses *import* data dan *preprocessing*	
Successful End Condition	Sistem berhasil mendapatkan *rule* lalu menampilkan *rule* tersebut	
Failed End Condition	Sistem tidak berhasil mendapatkan *rule* untuk dijadikan informasi	
Actors	*User*	
Triger	*User* memasukan nilai *minimum support* dan *minimum confidence*	
Included Cases	*Input* parameter, *Frequent itemset*, *rule*, lihat hasil asosiasi, dan rekomendasi produksi	
Main Flow	Step	Action
	1	*User input* nilai minimum *support*
	2	*User input* nilai minimum *confidence*
	3	Sistem melakukan proses pencarian *frequent itemset* sesuai nilai *minimum support*
	4	Sistem melakukan proses seleksi *rule* sesuai nilai minimum *confidence*
	5	Sistem melakukan proses lihat hasil asosiasi
	6	Sistem melakukan proses menampilkan informasi
Extension	Step	Branching Action
	1.1	Sistem menampilkan notifikasi nilai minimum *support* yang diinputkan tidak sesuai
	2.1	Sistem menampilkan notifikasi nilai minimum *confidence* yang diinputkan tidak sesuai
	3.1	Sistem tidak mendapatkan *frequent itemset*
	4.1	Sistem tidak mendapatkan *rule*
	5.1	Sistem tidak menampilkan hasil
	6.1	Sistem tidak menampilkan informasi

6. Skenario *Use Case Input* Parameter

Skenario *use case input* parameter menggambarkan langkah-langkah aksi *user* terhdap sistem dalam melakukan *input* nilai *minimum support* sebagai parameter dalam pencarian *frequent itemset*. Skenario *use case minimum support* dapat dilihat pada tabel III.17

Tabel III. 17 Skenario *Use Case Minimum Support*

Use case Name	**Minimum support**	
Related Requirements	Requirement *A.3*	
Goal In Context	Sistem mendapatkan nilai minimum *support* dan minimum *confidence*	
Precondition	*User* menginputkan nilai minimum *support* dan minimum *confidence*	
Successful End Condition	Sistem berhasil mendapatkan nilai minimum *support* dan minimum *confidence*	
Failed End Condition	Sistem tidak berhasil mendapatkan nilai minimum *support* dan minimum *confidence*	
Actors	*User*	
Triger	*User* menginputkan nilai minimum *support* dan minimum *confidence*	
Included Cases	*Frequent itemset*	
Main Flow	Step	Action
	1	Sistem mencari nilai rekomendasi *minimum support*
	2	Sistem menampilkan rekomendasi nilai *minimum support*
	3	User dapat menggunakan rekomendasi nilai *minimum support*
	4	*User* dapat mengubah nilai *minimum support* dan dapat menginputkan nilai *minimum support*
	5	Validasi inputan
	6	Sistem mendapatkan nilai *minimum support*
	7	*User* menginputkan nilai minimum *confidence*
	8	Sistem melakukan *validasi* inputan nilai *minimum confidence*
	9	Sistem mendapatkan nilai minimum *confidence*

Extension	Step	Branching Action
	6.1	Sistem menampilkan notifikasi nilai minimum *support* yang diinputkan tidak sesuai
	9.1	Sistem menampilkan notifikasi nilai minimum *confidence* yang diinputkan tidak sesuai

7. Scenario *Use Case Frequent Itemset*

Skenario *use case frequent itemset* menggambarkan langkah-langkah aksi *user* terhadap sistem dalam melakukan proses pencarian *frequent itemset*. Skenario *use case frequent itemset* dapat dilihat pada tabel III.18

Tabel III. 18 Skenario *Use Case Frequent Itemset*

Use case Name	***Frequent itemset***	
Related Requirements	Requirement *A.3*	
Goal In Context	Sistem melakukan pencarian *frequent itemset*	
Precondition	Telah melakukan proses *input* parameter	
Successful End Condition	Sistem berhasil melakukan pencarian *frequent itemset*	
Failed End Condition	Sistem tidak berhasil melakukan pencarian *frequent itemset*	
Actors	*User*	
Triger	*User* menginputkan nilai minimum *support* dan minimum *confidence*	
Included Cases	*rule*	
Main Flow	Step	Action
	1	*User* melakukan proses *input* parameter
	2	Sistem mengambil data hasil proses cleaning dari *database*
	3	Sistem membuat matrix 2 dimensi
	4	Sistem melakukan *transformasi* data yang bermula dari *database* ke dalam bentuk *matrix*
	5	data dalam bentuk *matrix*
	6	Sistem mencari k-*frequent* n-*itemset*
	7	Melakukan *cross* produk
	8	mencari nilai *support* menggunakan operan AND

Main Flow	Step	Action
	9	menghitung nilai *support count* dari setiap kolom dan *count* setiap barisnya
	10	mengeleminasi setiap kolom yang jumlah *support count*nya < minimum *support* dan menghapus setiap baris yang jumlah barisnya <= n
	11	Sistem melakukan cek *matrix*
	12	Sistem mendapatkan *frequent* n-*itemset*
	13	Proses pencarian *frequent itemset* selesai
Extension	Step	Branching Action
	1.1	Sistem menampilkan notifikasi nilai minimum *support* yang diinputkan tidak sesuai
	12.1	Ulangi step 6 sampai step 11 jika *matrix not null*
	12.2	Tidak mendapatkan *frequent n-itemset*

8. Skenario *Use Case Rule*

Skenario *use case rule* menggambarkan langkah-langkah aksi *user* untuk melakukan proses pembuatan *rule*. Skenario *use case rule* dapat dilihat pada tabel III.19

Tabel III. 19 *Skenario Use Case Rule*

Use case Name	***Rule***	
Related Requirements	Requirement *A.3*	
Goal In Context	Sistem mendapatkan *rule*	
Precondition	Sudah melewati proses *frequent itemset*	
Successful End Condition	Sistem berhasil mendapatkan *rule*	
Failed End Condition	Sistem tidak berhasil mendapatkan *rule*	
Actors	*User*	
Triger	*User* menginputkan nilai minimum *confidence*	
Included Cases	Lihat hasil asosiasi	
Main Flow	Step	Action
	1	*User* melakukan proses input nilai minimum *confidence*
	2	Sistem mengambil data hasil pencarian *frequent itemset*
	3	Sistem melakukan *generate rule*

Main Flow	Step	Action
	4	Sistem memilah *rule* sesuai dengan minimum *confidence*
	5	Sistem mendapatkan *rule* yang memenuhi nilai minimum *confidence*
Extension	Step	Branching Action
	3.1	Sistem gagal melakukan generate *rule* karena tidak terdapat *frequent itemset*
	5.1	Sistem gagal mendapatkan *rule*

9. Skenario *Use Case* Lihat Hasil Asosiasi

Skenario *use case lihat hasil asosiasi* menggambarkan langkah-langkah aksi sistem dalam menampilkan *rule*. Skenario *use case* lihat hasil asosiasi dapat dilihat pada tabel III.20

Tabel III. 20 Skenario *Use Case* Lihat Hasil Asosiasi

Use case Name	**Proses Asosiasi**	
Related Requirements	Requirement *A.3*	
Goal In Context	Sistem menampilkan *rule*	
Precondition	Sudah melakukan pencarian *frequent itemset* dan *rule*	
Successful End Condition	Sistem berhasil menampilkan *rule*	
Failed End Condition	Sistem tidak berhasil menampilkan *rule*	
Actors	*User*	
Triger	*User* menginputkan nilai minimum *support* dan minimum *confidence*	
Included Cases	Informasi	
Main Flow	Step	Action
	1	*rule* yang memenuhi nilai minimum *confidence*
	2	menampilkan *rule*
Extension	Step	Branching Action
	2.1	Sistem tidak menampilkan rule

10. Skenario *Use Case* Rekomendasi Produksi

Skenario *use case* rekomendasi produksi menggambarkan langkah-langkah aksi sistem dalam menampilkan rekomendasi jenis-jenis kaos kaki apa saja yang akan di produksi lebih dibulan selanjutnya. Skenario *use case* informasi dapat dilihat pada tabel III.21

Tabel III. 21 Skenario *Use Case* Rekomendasi Produksi

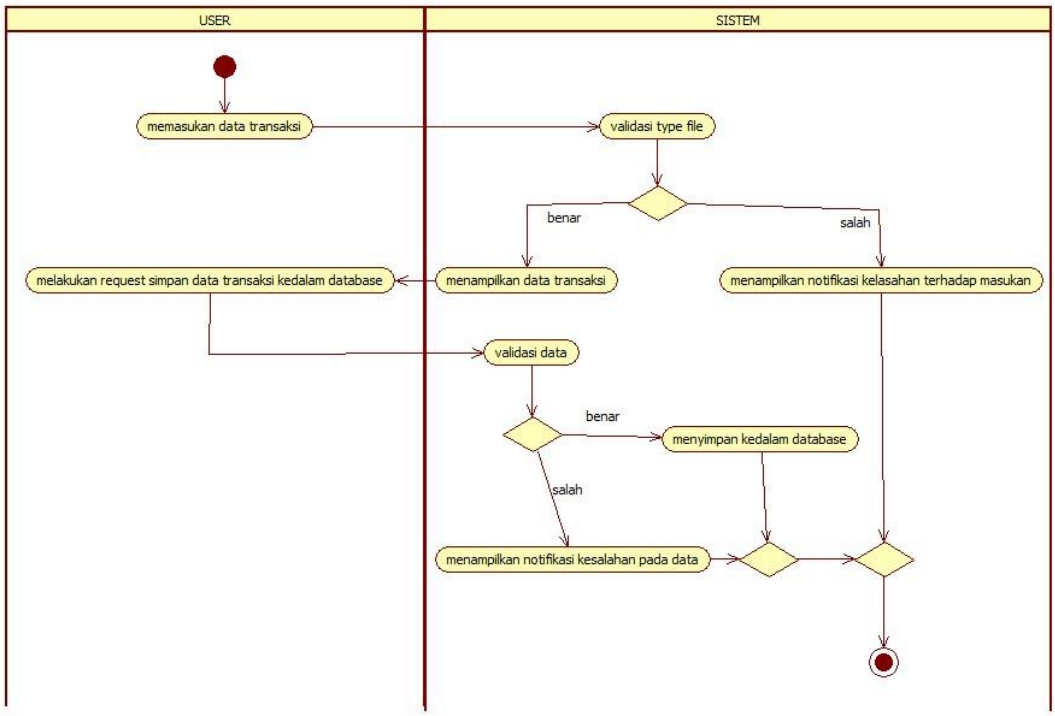
Use case Name	**Rekomendasi Produksi**	
Related Requirements	Requirement *A.3*	
Goal In Context	Sistem menampilkan *informasi* jenis-jenis kaos kaki yang akan di produksi lebih dibulan selanjutnya	
Precondition	Sudah melakukan pencarian *frequent itemset* dan *rule*	
Successful End Condition	Sistem berhasil menampilkan *informasi* jenis-jenis kaos kaki yang akan di produksi lebih dibulan selanjutnya	
Failed End Condition	Sistem tidak berhasil menampilkan *informasi* jenis-jenis kaos kaki yang akan di produksi lebih dibulan selanjutnya	
Actors	*User*	
Triger	*User* menginputkan nilai minimum *support* dan minimum *confidence*	
Included Cases	-	
Main Flow	Step	Action
	1	*rule* yang memenuhi nilai minimum *confidence*
	2	rule yang diperoleh dijadikan informasi
	3	menampilkan informasi jenis-jenis kaos kaki yang akan di produksi lebih dibulan selanjutnya
Extension	Step	Branching Action
	3.1	Sistem tidak menampilkan informasi

III.5.2 Activity Diagram

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

1. *Activity Diagram Import Data*

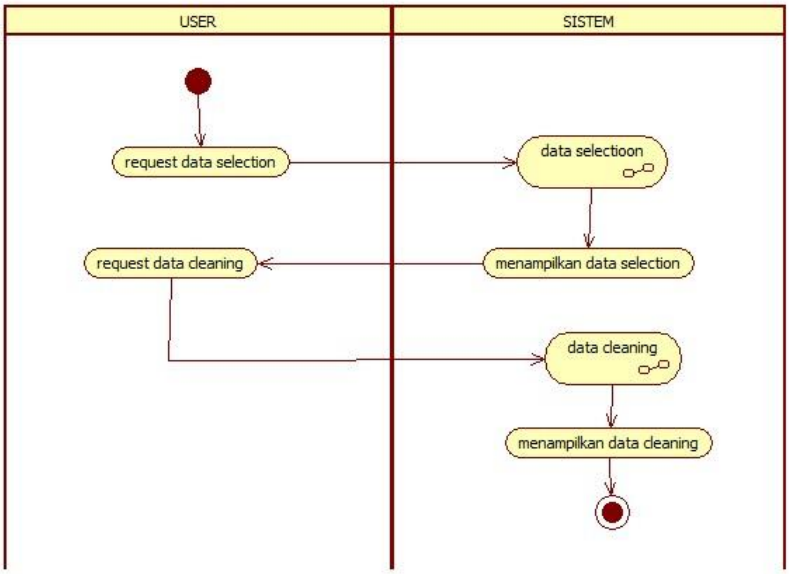
   Activity diagram import data menjelaskan aktivitas *user* dalam melakukan proses *import* data. *Activity diagram import* data dapat dilihat pada gambar III.2



Gambar III. 2 *Activity Diagram Import Data*

2. *Activity Diagram Preprocesing*

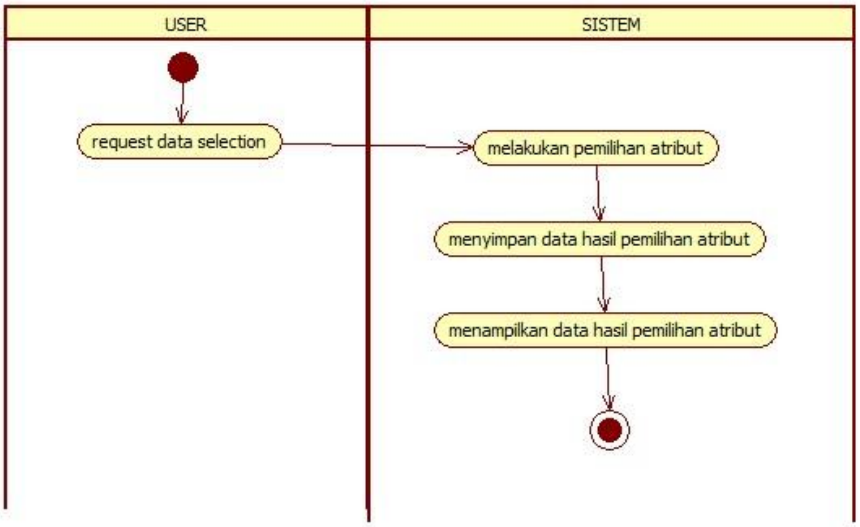
   Activity diagram preprocessing menjelaskan aktivitas sistem dalam melakukan proses *data selection dan data cleanning*. *Activity diagram preprocessing* dapat dilihat pada gambar III.3



Gambar III. 3 *Activity Diagram Preprocessing*

3. Activity Diagram Data Selection

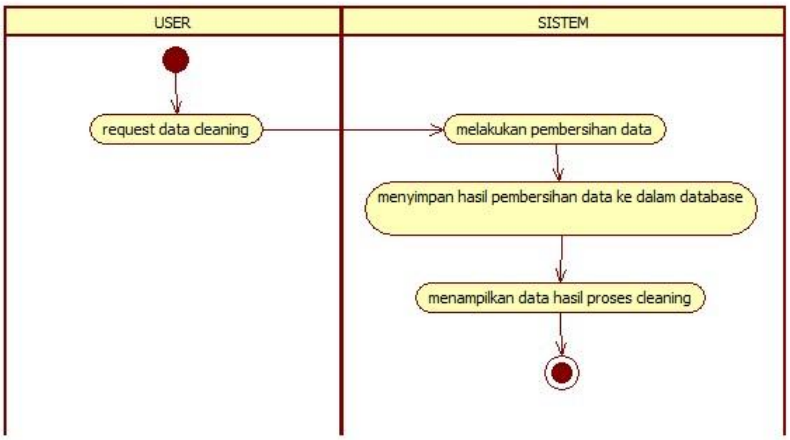
Activity diagram data selection menjelaskan aktivitas sistem dalam melakukan proses *data selection*. *Activity diagram data selection* dapat dilihat pada gambar III.4



Gambar III. 4 *Activity Diagram Data Selection*

4. *Activity Diagram Data Cleaning*

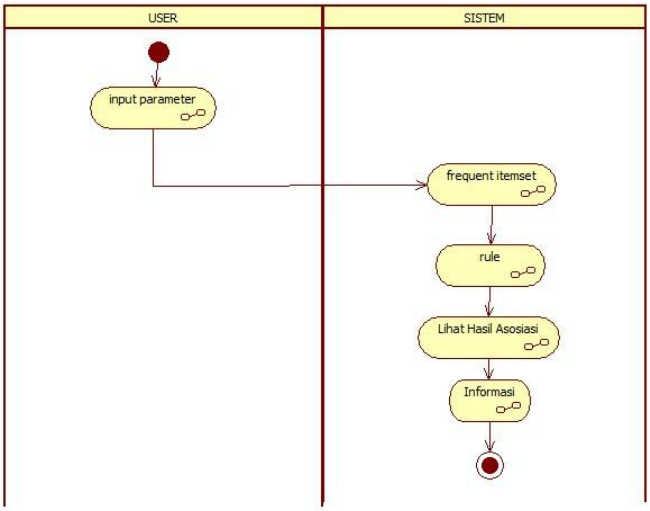
Activity diagram data cleaning menjelaskan aktivitas sistem dalam melakukan proses *data cleaning*. *Activity diagram data cleaning* dapat dilihat pada gambar III.5



Gambar III. 5 *Activity Diagram Data Cleaning*

5. *Activity Diagram* Proses Asosiasi

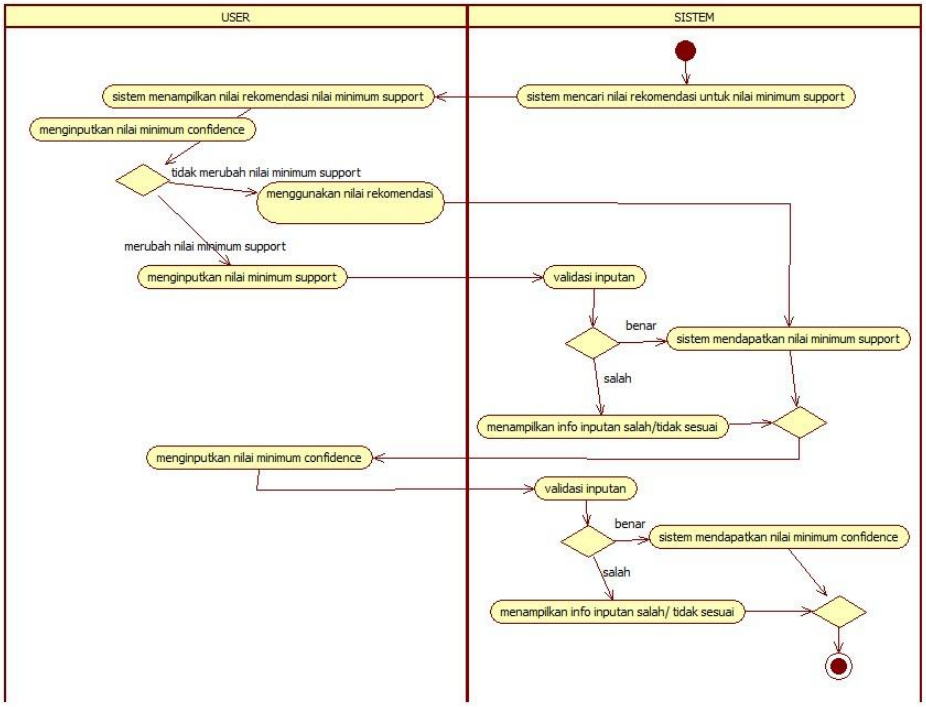
Activity diagram proses asosiasi menjelaskan aktivitas *user* dan sistem dalam melakukan proses pencarian *frequent itemset* , *rule* dan melihat hasil asosiasi. *Activity diagram proses asosiasi* dapat dilihat pada gambar III.6



Gambar III. 6 *Activity Diagram Proses Asosiasi*

6. *Activity* Diagram *Input* Parameter

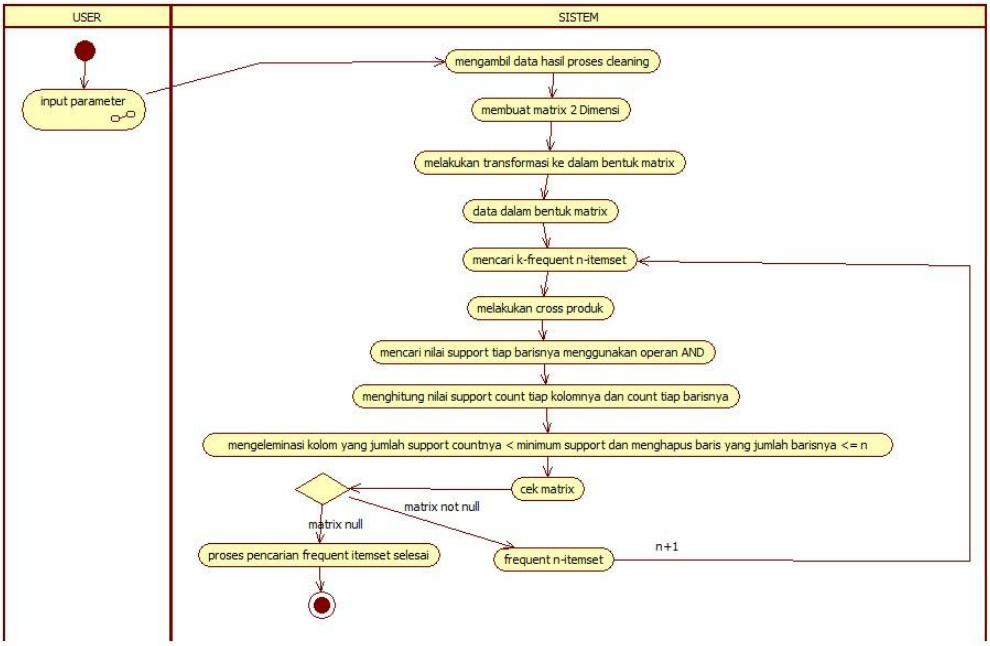
Activity diagram input parameter menjelaskan aktivitas *user* dan sistem dalam melakukan proses input nilai *minimum support dan minimum confidence*. *Activity diagram minimum support dan minimum confidence* dapat dilihat pada gambar III.7



Gambar III. 7 *Activity Diagram Input* Parameter

7. Activity Diagram Frequent Itemset

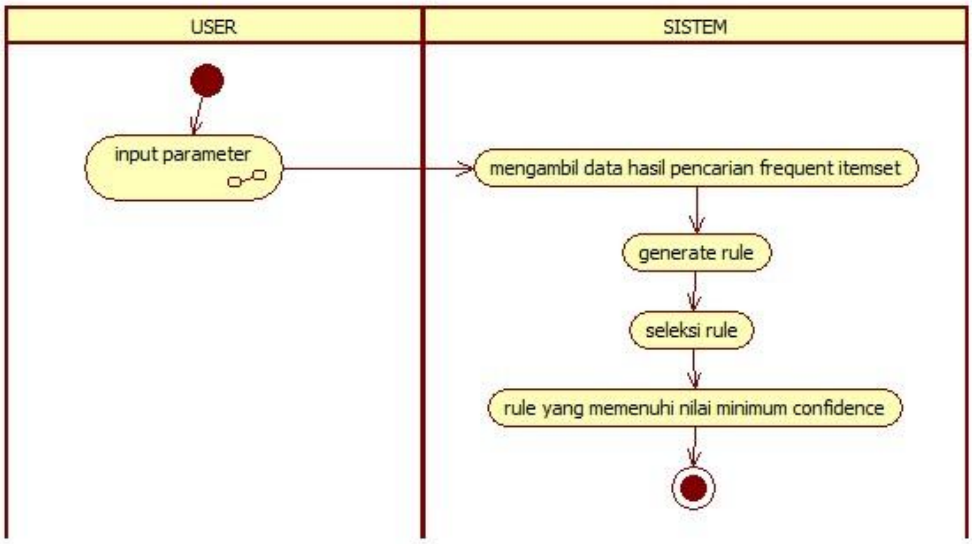
Activity diagram frequent itemset menjelaskan aktivitas *user* dan sistem dalam melakukan proses pencarian *frequent itemset*. *Activity diagram frequent itemset* dapat dilihat pada gambar III.8



Gambar III. 8 ***Activity Diagram Frequent Itemset***

8. Activity Diagram Rule

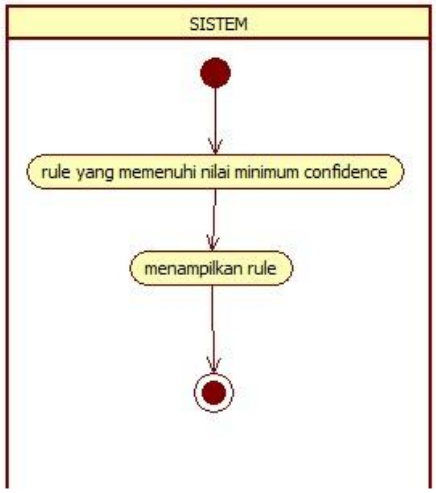
Activity diagram rule menjelaskan aktivitas *user* dan sistem dalam melakukan proses *generate rule*. *Activity diagram rule* dapat dilihat pada gambar III.9



Gambar III. 9 ***Activity Diagram Rule***

9. *Activity Diagram* Lihat Hasil Asosiasi

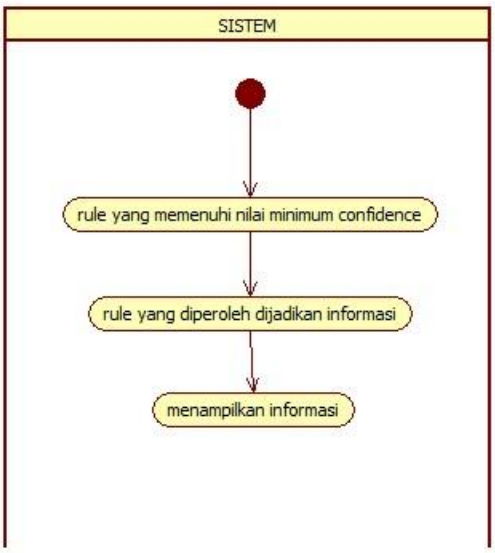
Activity diagram lihat hasil asosiasi menjelaskan aktivitas sistem dalam menampilkan hasil seleksi rule sesuai nilai minimum *confidence* yang diinputkan oleh *user*. *Activity diagram* lihat hasil asosiasi dapat dilihat pada gambar III.10



Gambar III. 10 ***Activity Diagram Lihat Hasil Asosiasi***

10. *Activity Diagram* Rekomendasi Produksi

Activity diagram rekomendasi produksi menjelaskan aktivitas sistem dalam menampilkan informasi jenis-jenis kaos kaki yang akan diproduksi lebih dibulan selanjutnya sesuai dengan *rule* yang didapatkan. *Activity diagram* informasi dapat dilihat pada gambar III.11



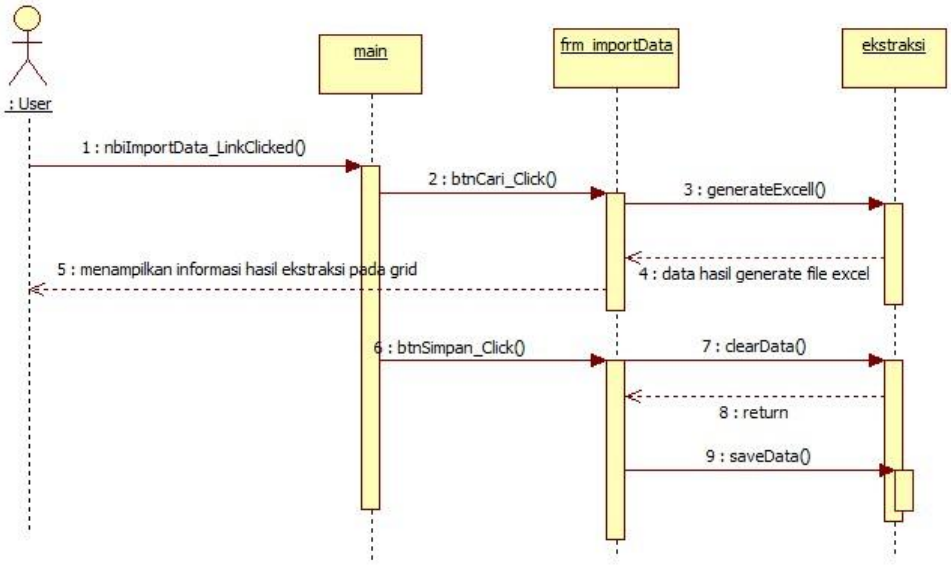
Gambar III. 11 *Activity Diagram* Rekomendasi Produksi

III.5.3 Sequense Diagram

Sequence diagram digunakan untuk menggambarkan arus pekerjaan, pesan yang disampaikan dan bagaimana elemen-elemen di dalamnya bekerja sama dari waktu ke waktu untuk mencapai tujuan hasil. Berikut adalah *sequence diagram* pada aplikasi *data mining* yang sedang dibangun pada Gambar III. 12 sampai dengan Gambar III. 21.

1. *Sequence Diagram Import Data*

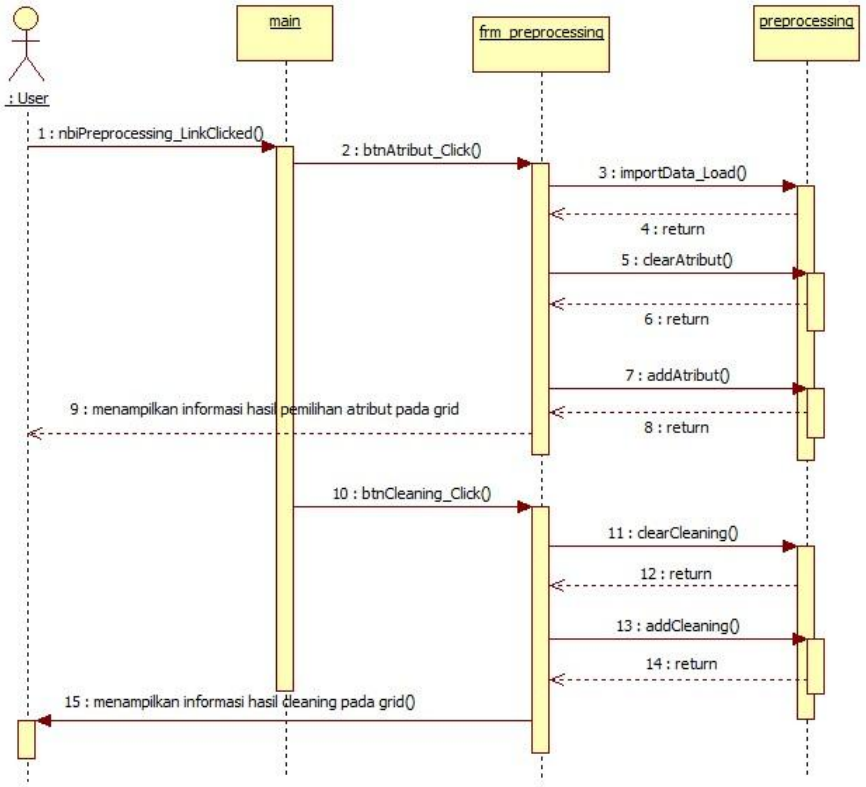
   Sequence diagram import data dapat dilihat pada gambar III.12.



Gambar III. 12 *Sequence Diagram Import Data*

2. *Sequence Diagram Preprocessing*

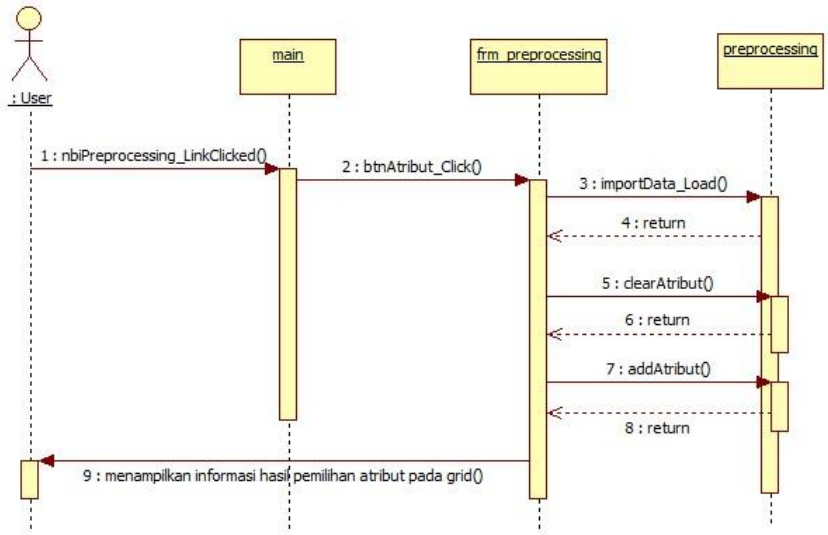
Sequence diagram preprocesing dapat dilihat pada gambar III.13



Gambar III. 13 *Sequence Diagram Preprocessing*

a Sequence diagram Data Selection

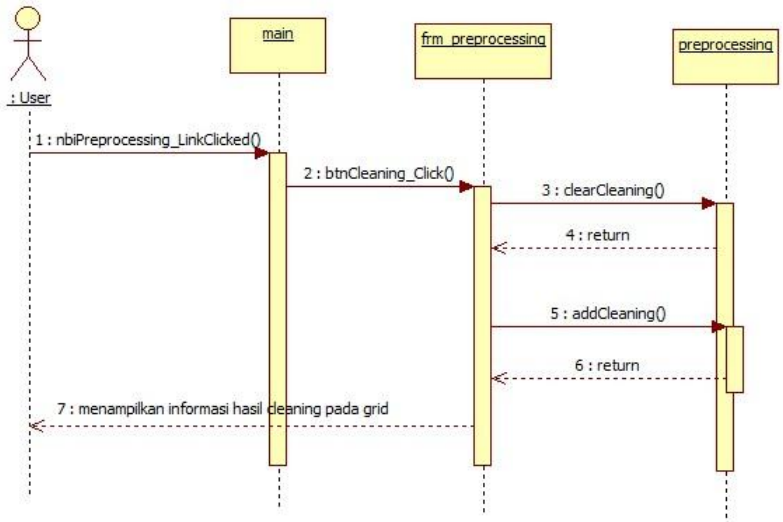
Sequence diagram data selection dapat dilihat pada gambar III.14



Gambar III. 14 *Sequence Diagram Data Selection*

b Sequence diagram Data Cleaning

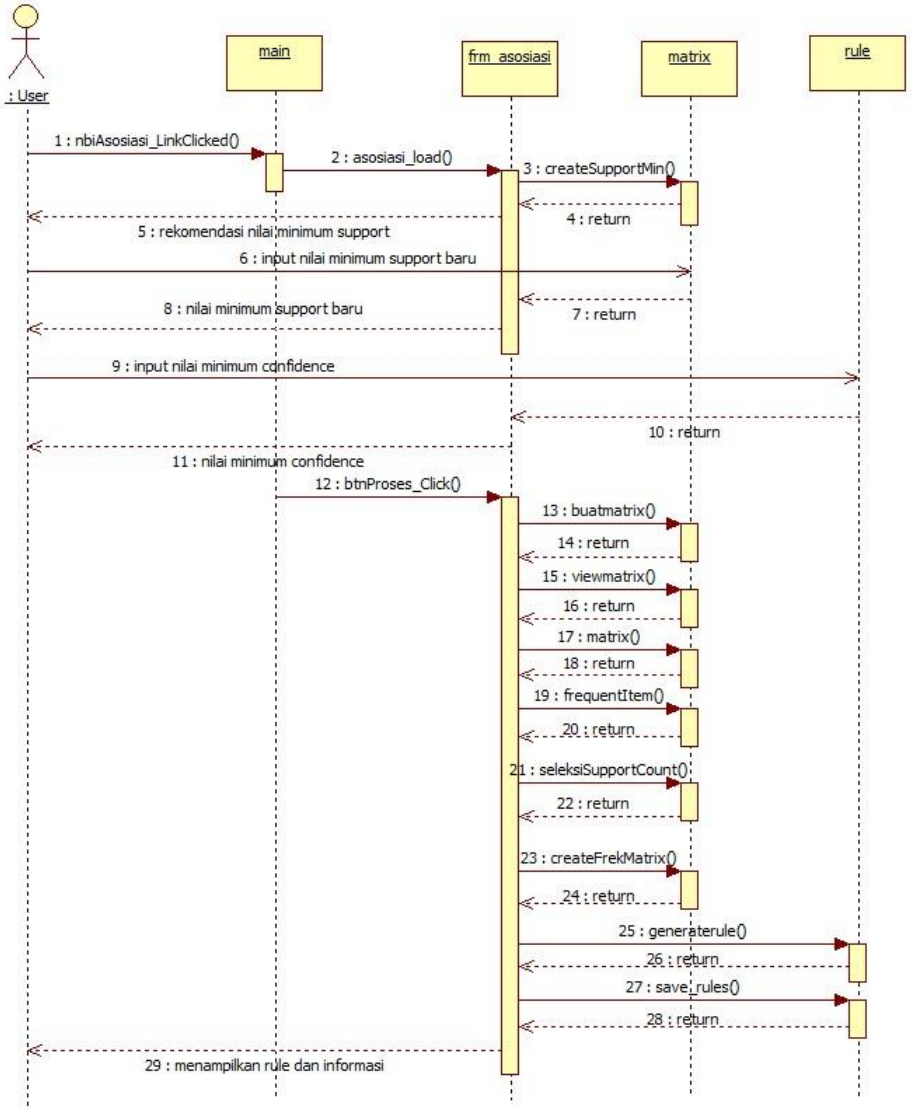
Sequence diagram data cleaning dapat dilihat pada gambar III.15



Gambar III. 15 *Sequence Digram Data Cleaning*

3. *Sequence Diagram* Asosiasi

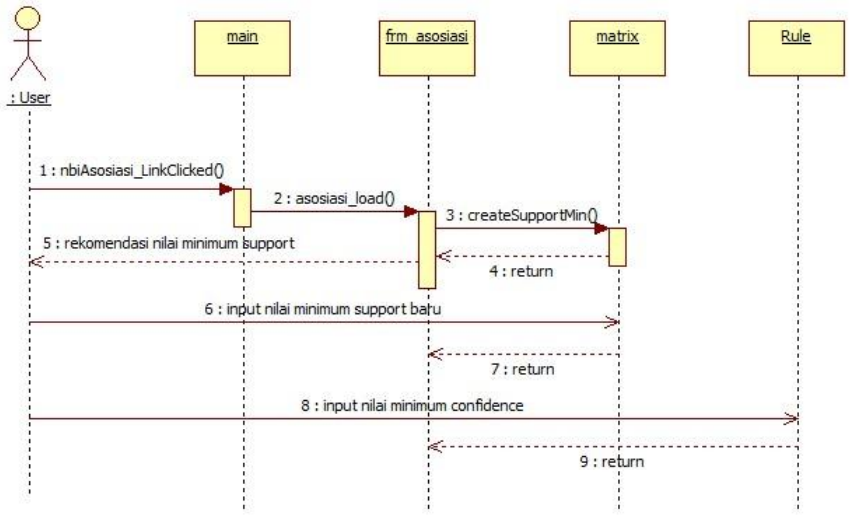
Sequence diagram Asosiasi dapat dilihat pada gambar III.16



Gambar III. 16 *Sequense Diagram Asosiasi*

a Sequence Diagram Input parameter

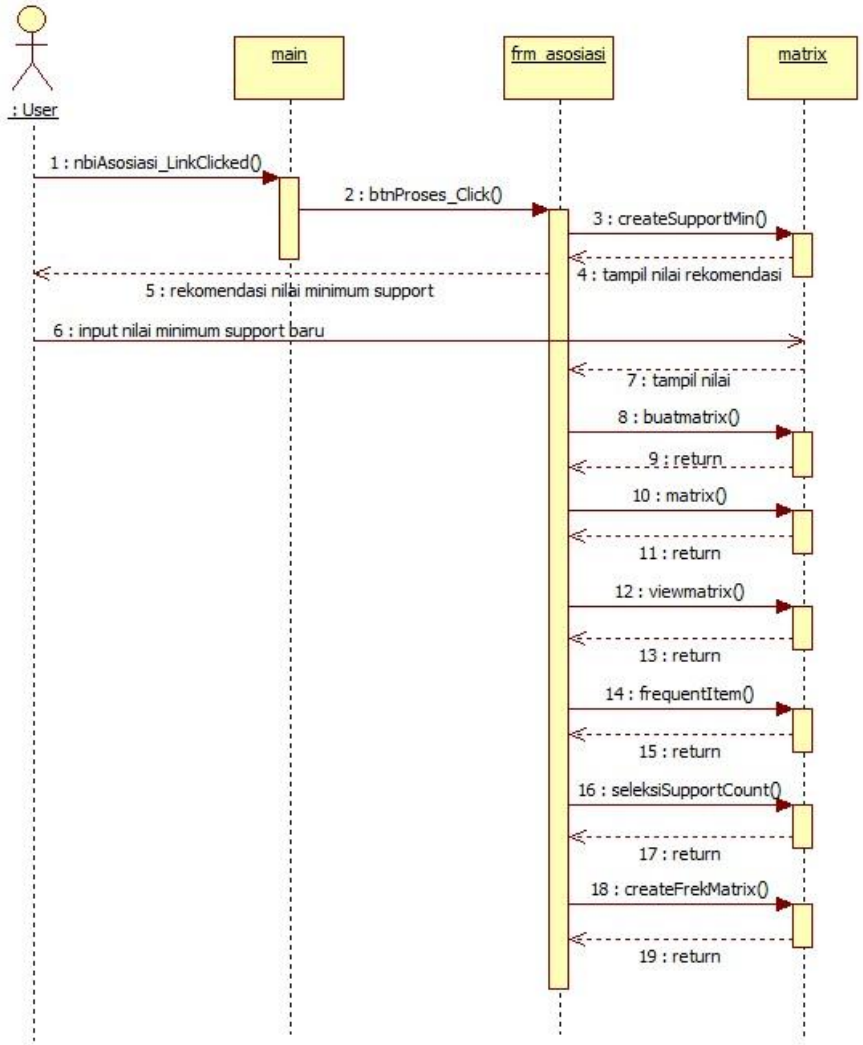
Sequence diagram input parameter dapat dilihat pada gambar III.17.



Gambar III. 17 *Sequence Diagram Input* Parameter

b Sequence Diagram Frequent Itemset

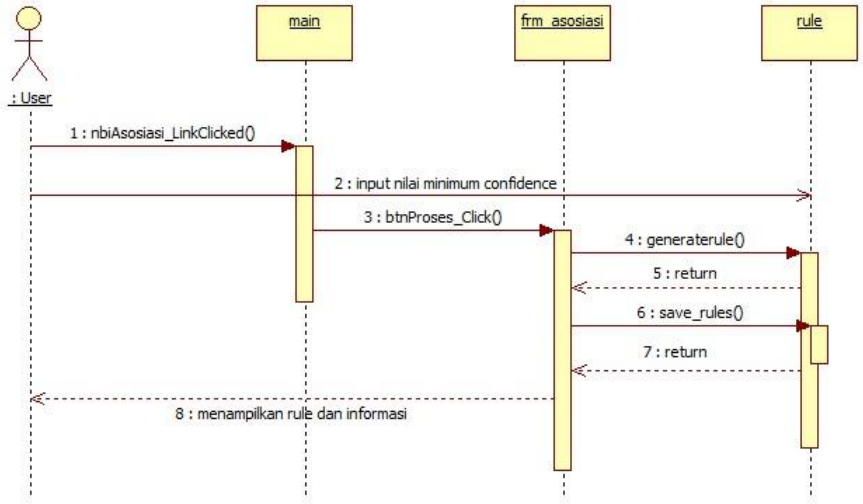
Sequence diagram frequent itemset dapat dilihat pada gambar III.18



Gambar III. 18 *Sequence Frequent Itemset*

c *Sequence Diagram Rule*

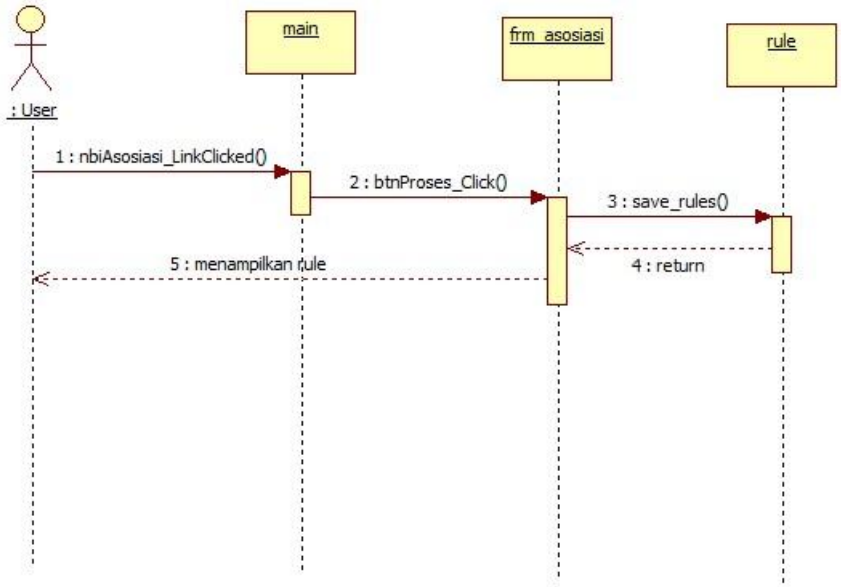
Sequence diagram rule dapat dilihat pada gambar III.19.



Gambar III. 19 *Sequence Diagram Rule*

d *Sequence Diagram* Lihat Hasil Asosiasi

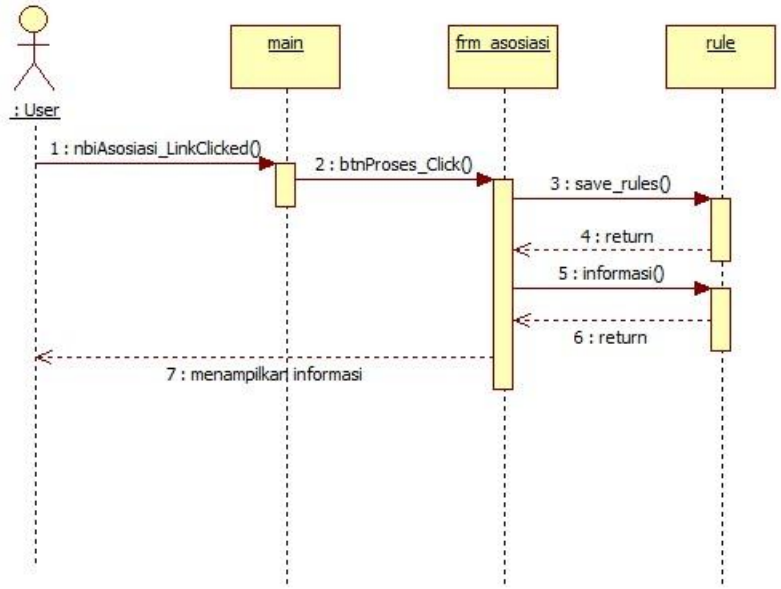
Sequence diagram lihat hasil asosiasi dapat dilihat pada gambar III.20



Gambar III. 20 *Sequence Diagram* Lihat Hasil Asosiasi

e *Sequence Diagram* Rekomendasi Produksi

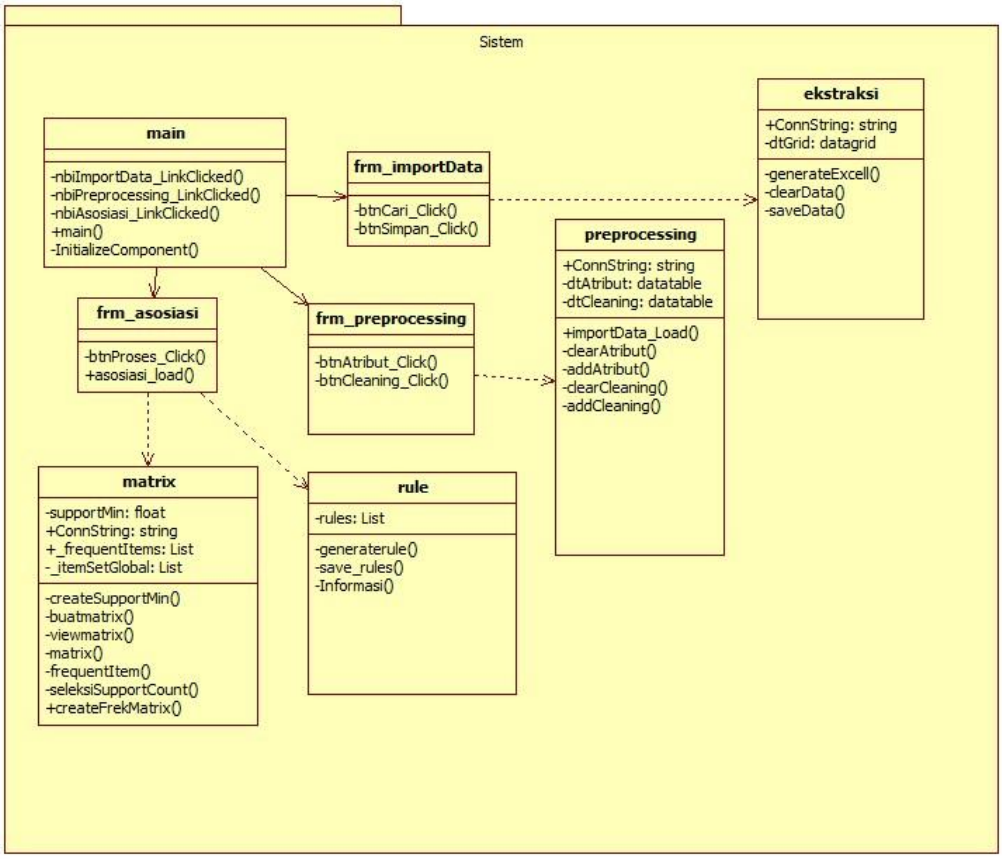
Sequence diagram Rekomendasi produksidapat dilihat pada gambar III.21



Gambar III. 21 *Sequence Diagram* Rekomendasi Produksi

III.5.4 Class Diagram

Berikut adalah *class diagram* dari sistem yang akan dibangun. Hasilnya dapat pada gambar III.22



Gambar III. 22 *Class Diagram*

III.6 Perancangan Sistem

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

III.6.1 Perancangan *Class*

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

Tabel III. 22 *Struktur Class* Main

Nama Class : Main		
Nama	Jenis	Type
main()	Method	Public
InitializeComponent()	Method	Public
nbiImportData_LinkClicked()	Method	Private
nbiPreprocessing_LinkClicked()	Method	Private
nbiAsosiasi_LinkClicked()	Method	Private

Tabel III. 23 *Struktur Class* frm_importData

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

Tabel III. 24 *Struktur Class* frm_preprocessing

Nama Class : frm_preprocessing		
Nama	Jenis	Type
btnAtribut_Click()	Method	Private
btnCleaning_Click()	Method	Private

Tabel III. 25 *Struktur Class* frm_asosiasi

Nama Class : frm_asosiasi		
Nama	Jenis	Type
btnProses_Click()	Method	Private
asosiasi_load()	Method	Private

Tabel III. 26 *Struktur Class Ekstraksi*

Nama Class : ekstraksi		
Nama	Jenis	Type
generateExcell()	Method	Private
clearData()	Method	Private
saveData()	Method	Private
ConnString: string	Atribut	Private
dtGrid: datagrid	Atribut	Private

Tabel III. 27 *Struktur Class Preprocessing*

Nama Class : preprocessing		
Nama	Jenis	Type
importData_Load()	Method	Private
clearAtribut()	Method	Private
addAtribut()	Method	Private
clearCleaning()	Method	Private
addCleaning()	Method	Private
ConnString: string	Atribut	Private
dtAtribut: datatable	Atribut	Private
dtCleaning: datatable	Atribut	Private

Tabel III. 28 *Struktur Class Matrix*

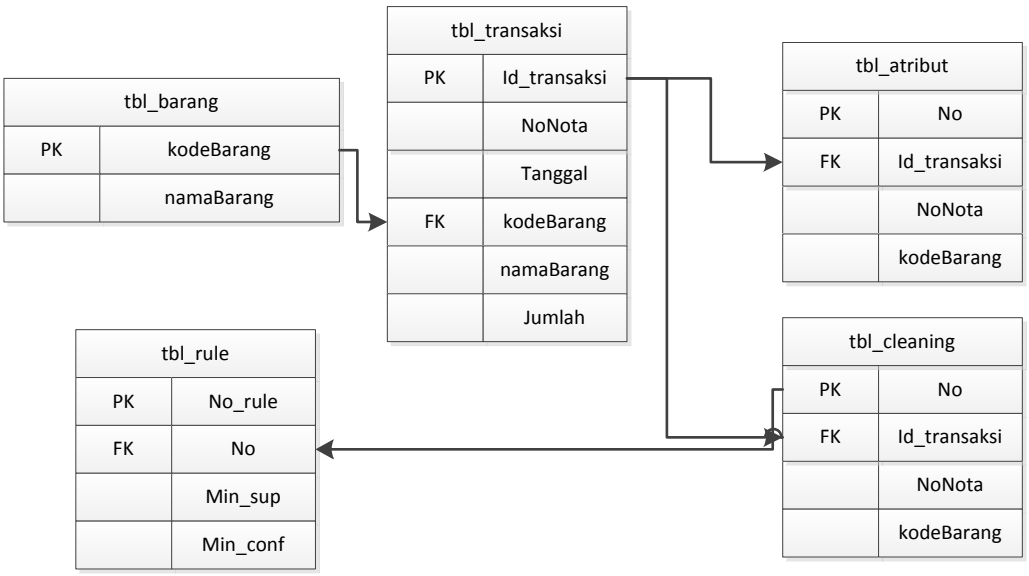
Nama Class : matrix		
Nama	Jenis	Type
createSupportMin()	Method	Private
buatmatrix()	Method	Private
viewmatrix()	Method	Private
matrix()	Method	Private
frequentItem()	Method	Private
seleksiSupportCount()	Method	Private
createFrekMatrix()	Method	Private
supportMin: float	Atribut	Private
ConnString: string	Atribut	Private
_frequentItems: List	Atribut	Private
_itemSetGlobal: List	Atribut	Private

Tabel III. 29 *Struktur Class Rule*

Nama Class : rule		
Nama	Jenis	Type
generaterule()	Method	Private
save_rules()	Method	Private
Informasi()	Method	Private
rules: List	Atribut	Private

III.6.2 Skema Relasi

Skema relasi merupakan rangkaian hubungan antara dua tabel atau lebih pada sistem basis data. Berikut ini adalah skema relasi pada sistem data *mining* di Kalvin Socks Production. Gambar Skema relasi dapat dilihat pada gambar III.23



Gambar III. 23 *Sekma Relasi*

III.6.3 Struktur Tabel

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

Tabel III. 30 Struktur Tabel tbl_barang

Nama Tabel : tbl_barang			
Nama Field	Type	Kunci	Keterangan
kodeBarang	Varchar(6)	Primary Key	Not Null
namaBarang	Varchar(50)		Not Null

Tabel III. 31 Struktur Tabel tbl_transaksi

Nama Tabel : tbl_transaksi			
Nama Field	Type	Kunci	Keterangan
Id_transaksi	Int (11)	Primary Key	Not Null
NoNota	Varchar(6)		Not Null
tanggal	Varchar(10)		Not Null
kodeBarang	Varchar(6)	Foreign Key references tbl_barang (kodeBarang)	Not Null
namaBarang	Varchar(50)		Not Null
jumlah	Int(11)		Not Null

Tabel III. 32 Struktur Tabel tbl_atribut

Nama Tabel : tbl_atribut			
Nama Field	Type	Kunci	Keterangan
No	Int(11)	Primary Key	Not Null
Id_transaksi	Int(11)	Foreign Key references tbl_transaksi (Id_transaksi)	Not Null
NoNota	Varchar(6)		Not Null
kodeBarang	Varchar(6)		Not Null

Tabel III. 33 Struktur Tabel tbl_cleaning

Nama Tabel : tbl_cleaning			
Nama Field	Type	Kunci	Keterangan
No	Int(11)	Primary Key	Not Null
Id_transaksi	Int(11)	Foreign Key references tbl_transaksi (Id_transaksi)	Not Null
NoNota	Varchar(6)		Not Null
kodeBarang	Varchar(6)		Not Null

Tabel III. 34 Struktur Tabel tbl_*rule*

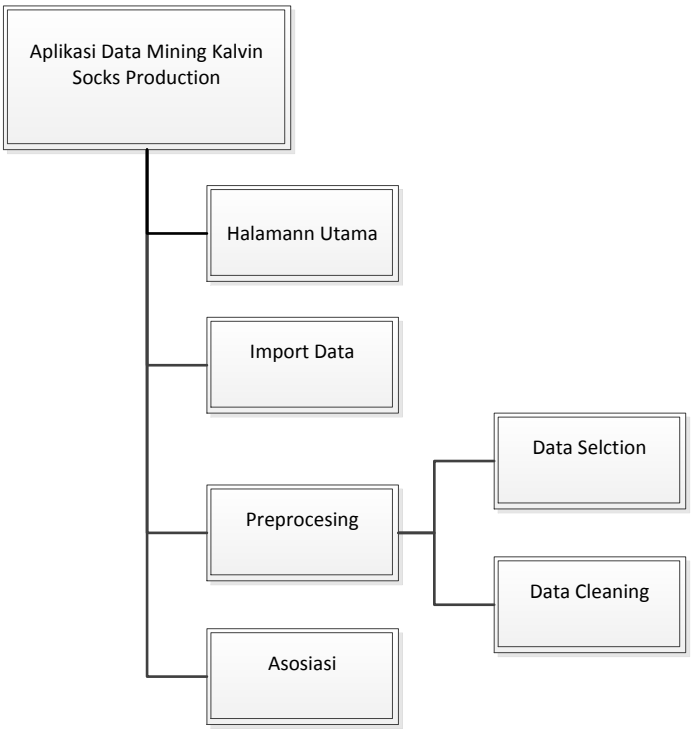
Nama Tabel : Rule			
Nama Field	Type	Kunci	Keterangan
No_rule	int(11)	Primary Key	Not Null
No	Int(11)	Foreign Key references tbl_cleaning (No)	Not Null
Support	Float		Not Null
confidence	Float		Not Null

III.6.4 Perancangan Arsitekur

Perancangan arsitektur merupakan perancangan yang dbuat sebelum perangkat lunak dibangun. Perancangan Arsitektur ini terdiri dari perancangan struktur menu, perancangan antar muka, dan jaringan semantik.

III.6.5 Perancangan Struktur Menu

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



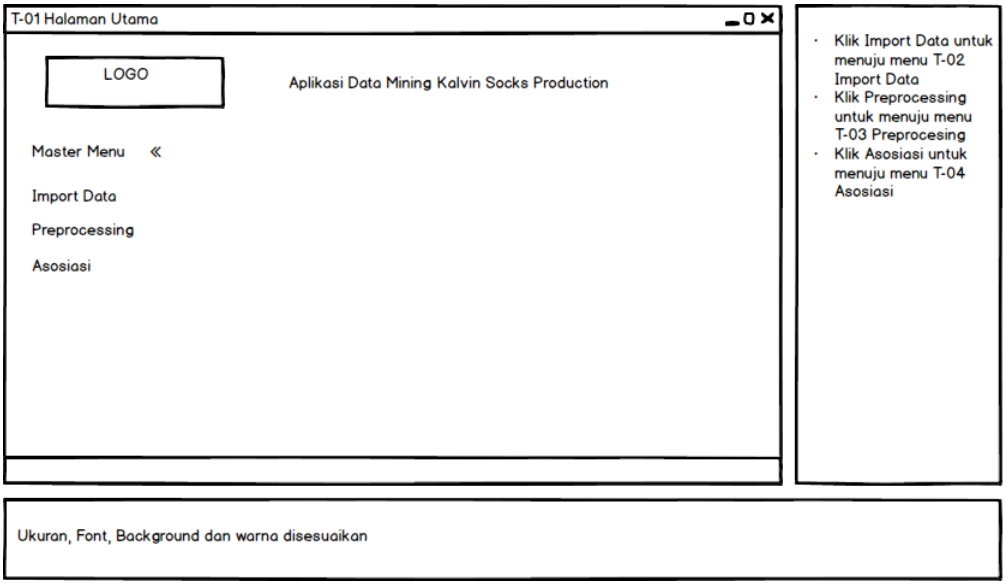
Gambar III. 24 Struktur Menu

III.6.6 Perancangan Antarmuka

Perancangan tampilan perangkat lunak bertujuan untuk memberikan gambaran-gambaran tenatang perangkat lunak yang akan dibangun, sehingga akan mempermudah dalam mengimplementasikan perangkat lunak. berikut merupakan perancangan antarmuka dari perangkat lunak yang akan dibangun :

a. Tampilan halaman Utama

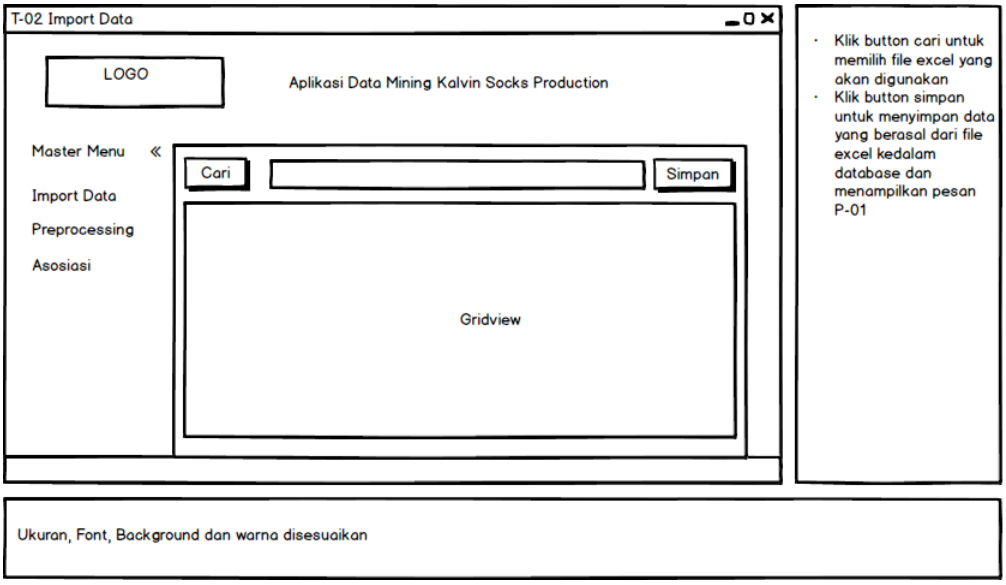
Berikut adalah tampilan halam utama yang akan dibangun pada sistem



Gambar III. 25 Tampilan Halaman Utama(T-01)

b. Tampilan *Import* Data

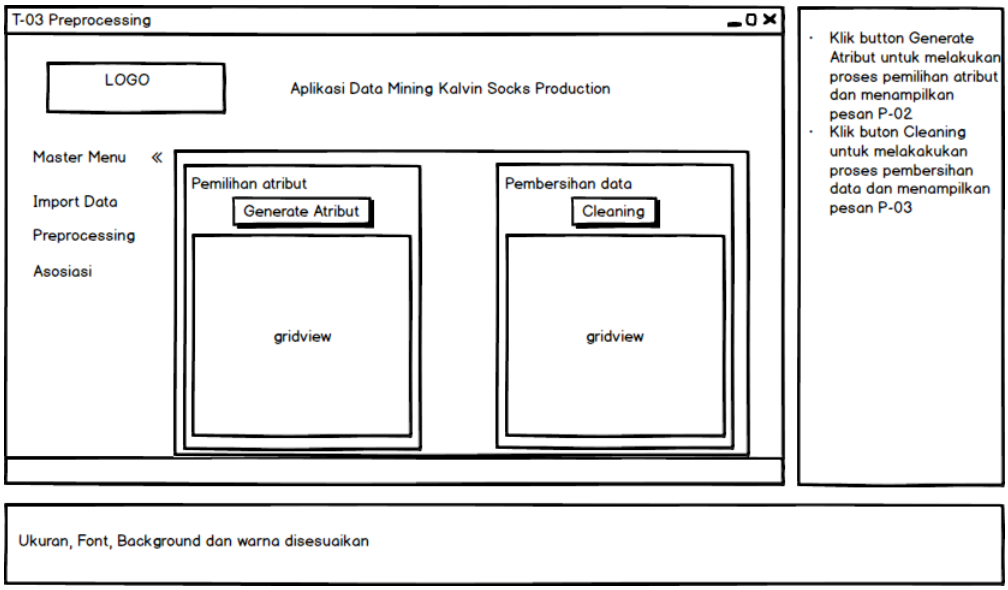
Berikut adalah tampilan *import* data yang akan dibangun pada sistem.



Gambar III. 26 Tampilan Import Data(T-02)

c. Tampilan *Preprocesing*

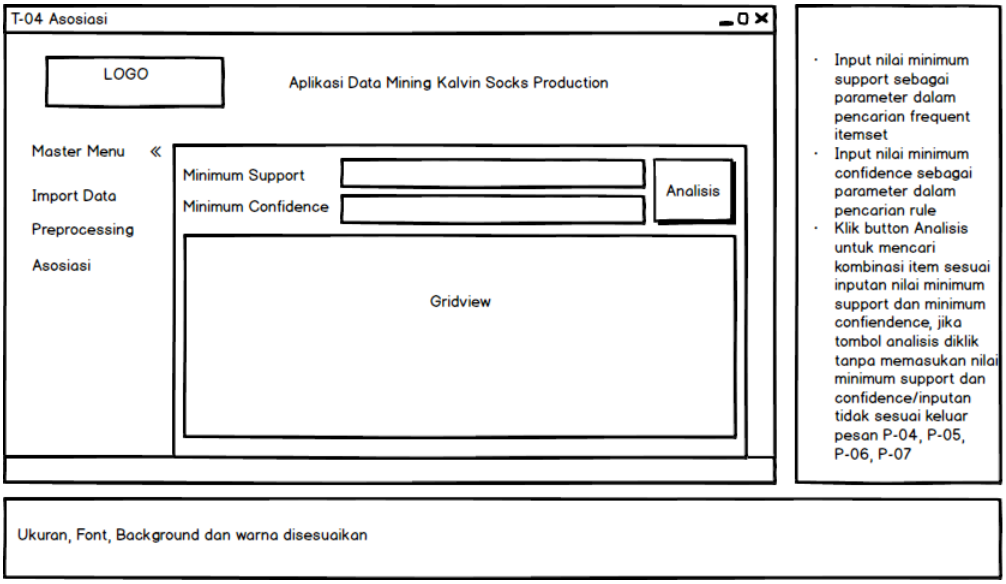
Berikut adalah tampilan *preprocessing* yang akan dibangun pada sistem.



Gambar III. 27 Tampilan *Preprocessing* (T-03)

d. Tampilan Asosiasi

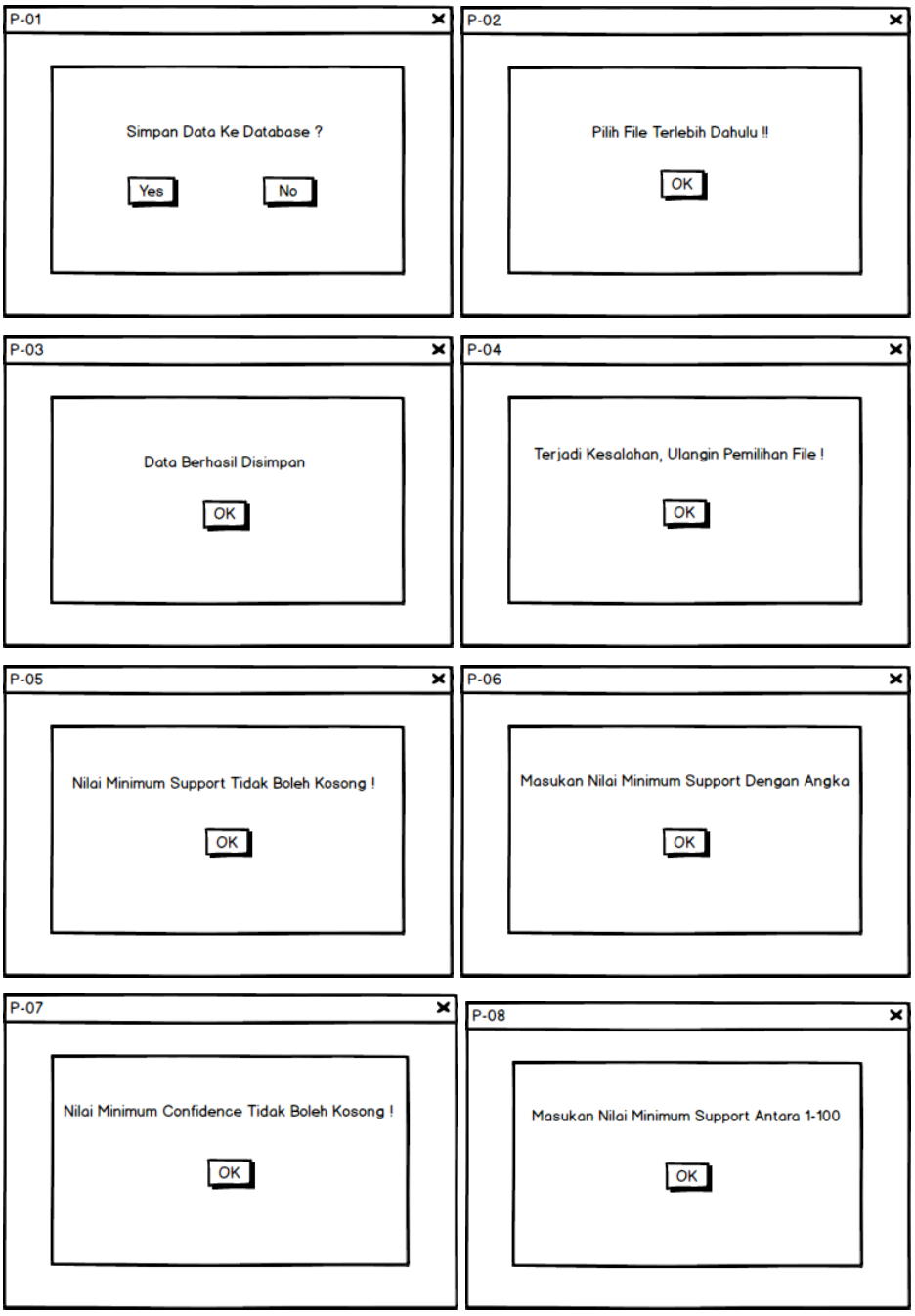
Berikut adalah tampilan asosiasi yang akan dibangun pada sistem.



Gambar III. 28 Tampilan *Asosiasi* (T-04)

III.6.7 Perancangan Tampilan Pesan

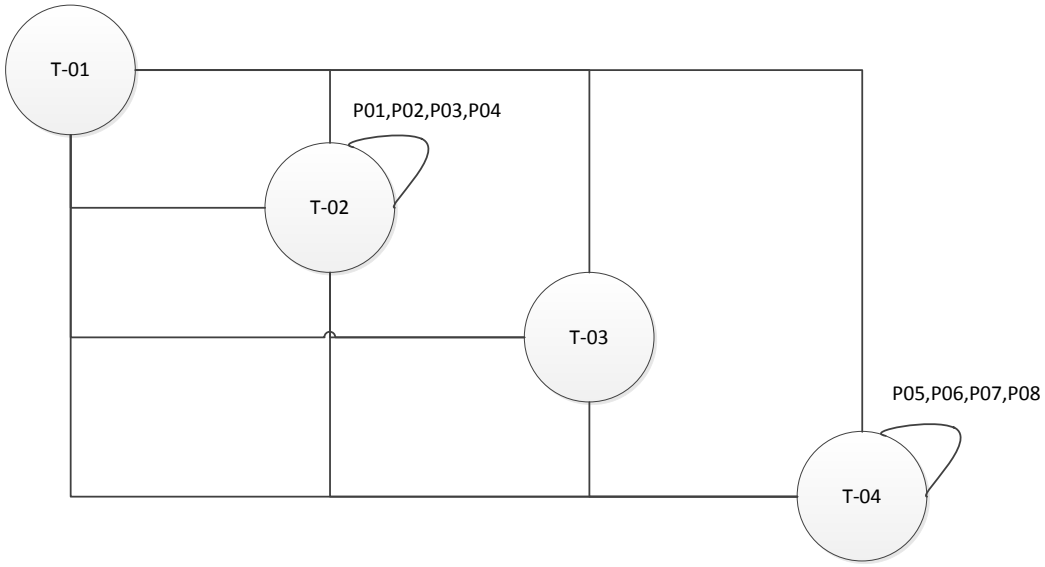
Berikut ini adalah perancangan pesan dari aplikasi yang akan dibangun.



Gambar III. 29 Perancangan Pesan

III.6.8 Jaringan Semantik

Berikut ini adalah jaringan sematik dari perangkat lunak yang akan dibuat.



Gambar III. 30 Jaringan Semantik