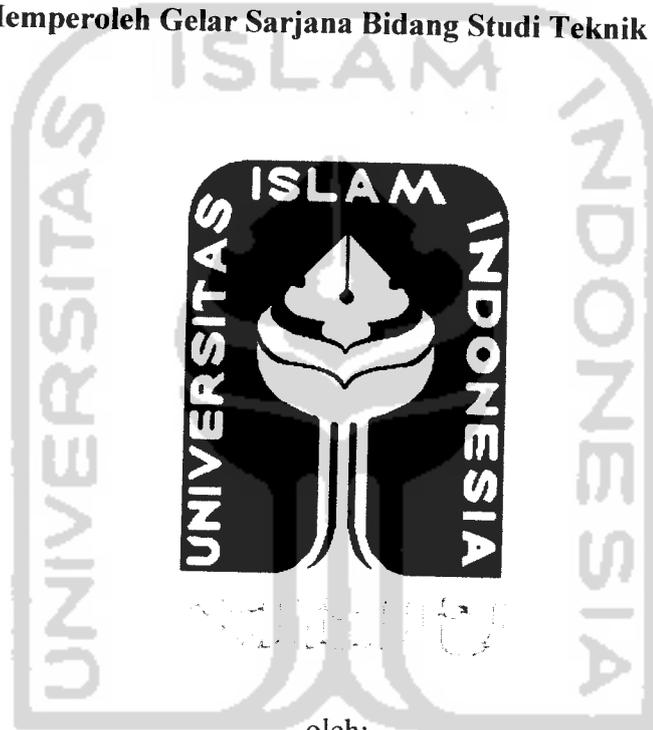


**ASSOCIATION RULES ANTAR KOMODITI PADA PUSAT
PERBELANJAAN DENGAN MENGGUNAKAN ALGORITMA
APRIORI**

TUGAS AKHIR

**Diajukan sebagai Salah Satu Syarat
untuk Memperoleh Gelar Sarjana Bidang Studi Teknik Informatika**



oleh:

Nama : Muhammad Wisynu Kurniawan
No. Mahasiswa : 01523094

**JURUSAN TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INDUSTRI
UNIVERSITAS ISLAM INDONESIA
YOGYAKARTA**

2007

LEMBAR PERNYATAAN KEASLIAN

HASIL TUGAS AKHIR

Saya yang bertanda tangan di bawah ini,

Nama : Muhammad Wisynu Kurniawan

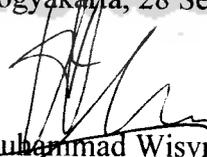
No. Mahasiswa : 01523094

Menyatakan bahwa seluruh komponen dan isi dalam Laporan Tugas Akhir ini adalah hasil karya saya sendiri.

Apabila di kemudian hari terbukti bahwa ada beberapa bagian dari karya ini adalah bukan hasil karya saya sendiri, maka saya siap menanggung resiko dan konsekuensi apapun.

Demikian pernyataan ini saya buat, semoga dapat dipergunakan sebagaimana mestinya.

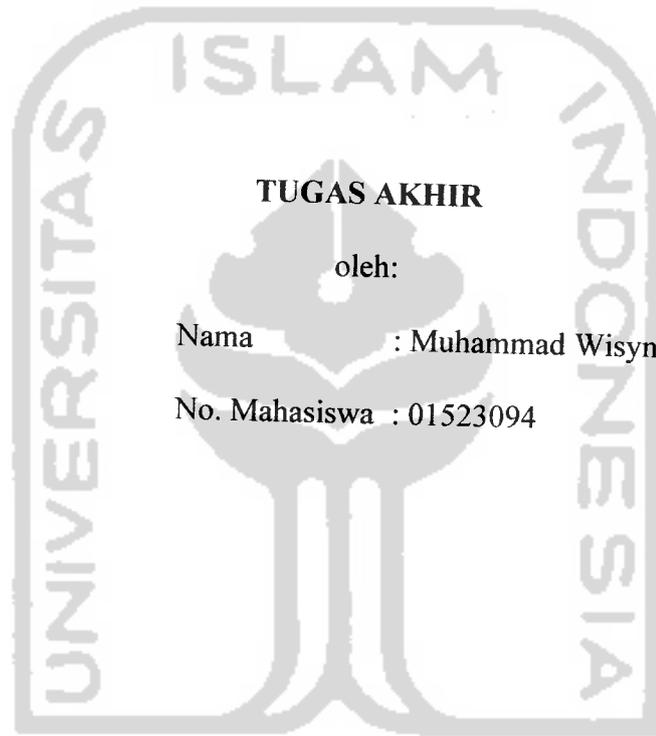
Yogyakarta, 28 September 2007


(Muhammad Wisynu Kurniawan)

LEMBAR PENGESAHAN PEMBIMBING

**ASSOCIATION RULES ANTAR KOMODITI PADA PUSAT
PERBELANJAAN DENGAN MUNGUNAKAN ALGORITMA**

APRIORI



TUGAS AKHIR

oleh:

Nama : Muhammad Wisynu Kurniawan

No. Mahasiswa : 01523094

Yogyakarta, 28 September 2007

Pembimbing,

A handwritten signature in black ink, appearing to read 'Taufiq Hidayat', is written over a horizontal line.

Taufiq Hidayat, ST, MCS.

LEMBAR PENGESAHAN PENGUJI

**ASSOCIATION RULES ANTAR KOMODITI PADA PUSAT
PERBELANJAAN DENGAN MUNGUNAKAN ALGORITMA
APRIORI**

TUGAS AKHIR

oleh:

Nama : Muhammad Wisynu K.
No. Mahasiswa : 01523094

Telah Dipertahankan di Depan Sidang Penguji sebagai Salah Satu Syarat
untuk Memperoleh Gelar Sarjana Jurusan Teknik Informatika
Fakultas Teknologi Industri Universitas Islam Indonesia

Yogyakarta, 6 Nopember 2007

Tim Penguji

Taufiq Hidayat, ST, MCS.
Ketua



Nur Wijayaning R, S. Kom.
Anggota I



Hendrik, ST.
Anggota II



Mengetahui,
Ketua Tim Penguji Teknik Informatika
(Muhammad Wisynu K., S.Si., M.Kom)

HALAMAN PERSEMBAHAN



MOTTO

أَقْرَأْ بِأَسْمِ رَبِّكَ الَّذِي خَلَقَ ﴿١﴾

“Bacalah dengan (menyebut) nama Tuhanmu yang Menciptakan,” (QS. Al-Alaq:1)

وَأَسْتَعِينُوا بِالصَّبْرِ وَالصَّلَاةِ وَإِنَّهَا لَكَبِيرَةٌ إِلَّا عَلَى الْخَاشِعِينَ ﴿٤٥﴾

“Jadikanlah sabar dan shalat sebagai penolongmu. dan Sesungguhnya yang demikian itu sungguh berat, kecuali bagi orang-orang yang khusyu,”(QS. Al-baqarah:45)

وَلَا تُفْسِدُوا فِي الْأَرْضِ بَعْدَ إِصْلَاحِهَا وَادْعُوهُ خَوْفًا وَطَمَعًا إِنَّ رَحْمَتَ اللَّهِ

قَرِيبٌ مِّنَ الْمُحْسِنِينَ ﴿٥٦﴾

“Dan janganlah kamu membuat kerusakan di muka bumi, sesudah (Allah) memperbaikinya dan berdoalah kepada-Nya dengan rasa takut (Tidak akan diterima) dan harapan (akan dikabulkan). Sesungguhnya rahmat Allah amat dekat kepada orang-orang yang berbuat baik.”(QS. Al-A’raf:56)

KATA PENGANTAR

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

Assalamu'alaikum wa rahmatullaahi wa barakaatuh

Segala puji dan syukur hanyalah milik Allah Subhanahu Wa Ta'ala yang Maha Pengasih dan Maha Penyayang, yang telah menjadikan manusia beriman dan berilmu setelah sebelumnya berada dalam kondisi yang lemah dan diliputi kebodohan. Atas izin dan kehendak-Nya, tugas akhir berjudul "ASSOCIATION RULES ANTAR KOMODITI PADA PUSAT PERBELANJAAN MENGGUNAKAN ALGORITMA APRIORI" ini akhirnya dapat terselesaikan.

Tugas Akhir ini merupakan salah satu syarat untuk memperoleh gelar sarjana Teknik Informatika di Universitas Islam Indonesia.

Penulis dalam kesempatan ini mengucapkan terima kasih kepada :

1. Papa, mama, mbak Ida ,mas Agus serta keluarga tercinta lainnya yang telah banyak memberikan do'a dan semangat.
2. Maya yang selalu mendukung dan menemani penulis dalam mengerjakan tugas akhir ini.
3. Bapak Taufik Hidayat, ST. MCS. selaku dosen pembimbing tugas akhir. Terima kasih atas segala kesabaran, bantuan, dukungan, pengetahuan dan segala kemudahan yang diberikan pada penulis.
4. Bapak Fathul Wahid, ST, M.Sc. selaku Dekan Fakultas Teknologi Industri.

5. Bapak Yudi Prayudi, S.Si, M.Kom. selaku Ketua Jurusan Teknik Informatika.
6. Sahabat-sahabatku eks-kontrakan Banteng, alan dan ana, mas pandu dan mbak devine, mas gotri dan mbak fitri, mas ajie, mas yudi, edwin, davi.
7. Teman-teman “BINER” diyah “Bo**dhe” ayu ratnaningsih, kurniawan “bowl” arsita, syamsul adinugroho, fitiyan “blak” zamzami, rindang “sikil 4” rahdityo, chairul “ilur” sabar, isnaendera “uus” yusminanda, kurnia “ozamah” iriawan, saiful “robot” munif, wildan ammar, ryan akbar serta teman-teman lainnya yang tidak dapat disebutkan satu persatu, penulis tidak akan melupakan semua pengalaman yang telah dan akan kita jalani.....”**SATU UNTUK SEMUA, SEMUA UNTUK SATU!!!**”
8. Teman – temanku di Purbalingga.
9. Serta pihak-pihak yang penulis tidak dapat sebutkan satu persatu.

Dalam penyelesaian Tugas Akhir ini penulis menyadari bahwa masih banyak terdapat kelemahan dan kekurangan. Oleh karena itu penulis mengharapkan kritik dan saran yang bersifat membangun agar pada masa mendatang menjadi lebih baik.

Akhir kata, Semoga Tugas Akhir ini dapat berguna bagi para penuntut ilmu, para praktisi, dan seluruh masyarakat IT untuk tujuan kemaslahatan dan kepentingan bersama.

Amin

Wassalamu'alaikum wa rahmatullaahi wa barakaatuh

Yogyakarta, 28 September 2007

Penulis



SARI

Pada pusat-pusat perbelanjaan saat ini, dapat dipastikan menyimpan semua data penjualan atau transaksi yang terjadi pada media digital dan data tersebut menjadi hal yang sangat krusial. Dengan adanya data penjualan, pembuatan laporan keuangan, stok barang, mencari tahu barang yang paling banyak dibeli akan menjadi sangat mudah dan efisien. Data tersebut juga akan mengalami pertumbuhan yang sangat cepat, seiring dengan banyaknya transaksi yang terjadi. Di samping banyaknya manfaat dari data penjualan, muncul satu pertanyaan yang menarik, apakah ada suatu pola di dalam data tersebut yang sebelumnya tidak diketahui. Data mining adalah ekstraksi informasi yang sebelumnya tidak diketahui, dan informasi yang ditemukan bermanfaat bagi pemilik data. Pada penelitian ini akan dilakukan suatu proses data mining dengan teknik *association rules* (aturan asosiasi) menggunakan algoritma apriori. Aturan-aturan yang didapat akan disajikan dalam bentuk “if-then” atau “jika-maka”, misalnya jika membeli pengharum ruangan maka membeli juga sabun mandi. Karena awalnya berasal dari studi tentang database transaksi pelanggan untuk menentukan kebiasaan suatu produk dibeli bersama produk apa, maka aturan asosiasi juga sering dinamakan *market basket analysis*.

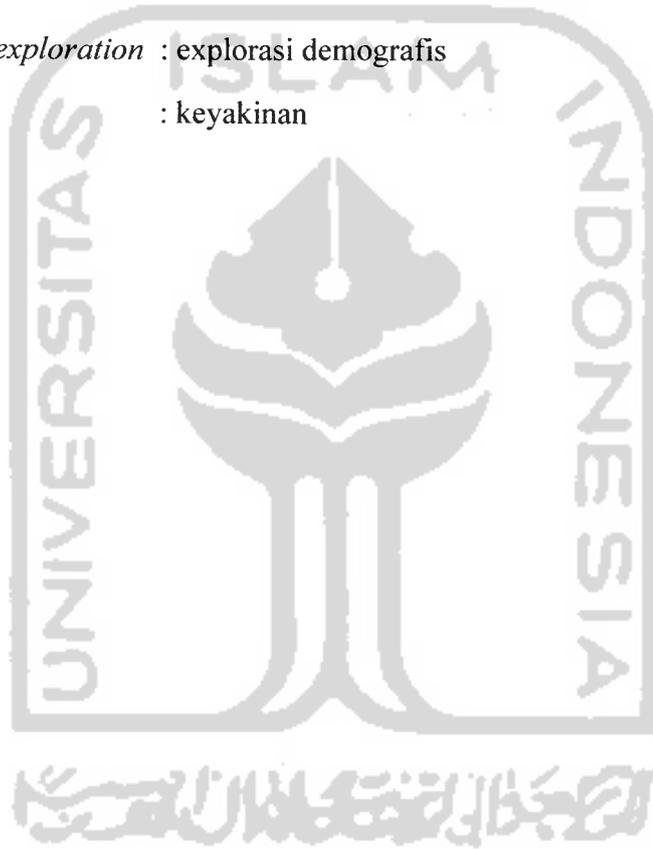
Algoritma apriori menggunakan *support* dan *confidence* dalam pencarian aturan. *Support* adalah berapa banyak transaksi yang mengandung suatu itemset tertentu. *Confidence* mengukur seberapa besar ketergantungan suatu *item* dengan item yang lainnya. *Confidence* digunakan sebagai perangkingan dari aturan asosiasi.

Dari hasil pengujian yang dilakukan dengan menggunakan input dari beberapa *dataset* transaksi yang berbeda ditemukan aturan asosiasi antar item atau barang, salah satu di antaranya adalah jika membeli Tissue Tessa, Fanta 1lt, Attack 500gr, Pepsodent 100gr maka juga membeli Djarum Super 12 dengan *support* sebesar 3 dan *confidence* sebesar 1. Dari aturan yang ditemukan dapat digunakan pemilik data khususnya pusat perbelanjaan untuk pengambilan keputusan yang menguntungkan. Contohnya peletakan barang-barang, diskon, barang-barang apa saja yang diminati oleh konsumen dengan pekerjaan tertentu dan lain sebagainya.

Kata kunci: *Data mining*, Aturan asosiasi, *Market basket analysis*, Algoritma apriori, *support*, *confidence*.

TAKARIR

<i>Association rule</i>	: aturan asosiasi
<i>Machine learning</i>	: pembelajaran mesin
<i>Pattern recognition</i>	: pengenalan pola
<i>Constraint</i>	: batasan
<i>demographic exploration</i>	: explorasi demografis
<i>confidence</i>	: keyakinan



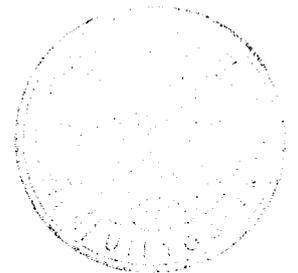
DAFTAR ISI

JUDUL.....	i
LEMBAR PERNYATAAN KEASLIAN.....	ii
LEMBAR PENGESAHAN PEMBIMBING.....	iii
LEMBAR PENGESAHAN PENGUJI.....	iv
HALAMAN PERSEMBAHAN.....	v
HALAMAN MOTTO.....	vi
KATA PENGANTAR.....	vii
SARI.....	x
TAKARIR.....	xi
DAFTAR ISI.....	xii
DAFTAR GAMBAR.....	xv
DAFTAR TABEL.....	xvii
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang Masalah.....	1
1.2 Rumusan Masalah.....	2
1.3 Batasan Masalah.....	3
1.4 Tujuan Penelitian.....	3
1.5 Manfaat Penelitian.....	4
1.6 Metodologi Penelitian.....	4
1.7 Sistematika Penulisan.....	5
BAB II DASAR TEORI.....	7
2.1 Dasar Data mining.....	7
2.1.1 Definisi Data Mining.....	7
2.1.2 Bidang Ilmu Yang Mempengaruhi Data Mining.....	8
2.1.3 Market Basket Analysis.....	9
2.1.4 Teknik-teknik Data Mining.....	10
2.2 Association Rules Mining.....	10
2.2.1 Aturan dari Association Rules Mining.....	11
2.2.1.1 <i>Itemset</i>	11
2.2.1.2 <i>Support</i>	12
2.2.1.3 <i>Confidence</i>	12
2.2.1.4 <i>Rule</i>	13
2.2.2 Algoritma Apriori.....	14
BAB III METODOLOGI.....	24
3.1.1 Metode Analisis.....	24

3.1.2 Hasil Analisis	24
3.1.3 Masukan Sistem	24
3.1.4 Keluaran Sistem	26
3.1.5 Kebutuhan Proses.....	26
3.1.6 Kebutuhan Perangkat Lunak	27
3.1.7 Kebutuhan Perangkat Keras.....	27
3.1.8 Antar Muka Sistem	28
3.2.1 Metode Perancangan	28
3.2.2 Hasil Perancangan.....	28
3.2.3 Data Flow Diagram.....	29
3.2.3.1 DFD Level 0	29
3.2.3.2 DFD Level 1	30
3.2.3.3 DFD Level 2 proses 1 (Proses Koneksi).....	31
3.2.3.4 DFD level 2 Proses 2 (Proses Input Query).....	32
3.2.3.5 DFD level 2 proses 3 (Proses Pembentukan Frequent Itemset)...	33
3.2.3.6 DFD Level 2 proses 4 (Proses Pembentukan Aturan Asosiasi)...	34
3.2.4 Perancangan Basis Data	36
3.2.5 Relasi Antar Tabel dalam Database	39
3.2.6 Perancangan Antarmuka	40
3.2.6.1 Rancangan Antarmuka Koneksi Database.....	40
3.2.6.2 Rancangan Antarmuka Pengambilan Data dari Database	41
3.2.6.3 Rancangan Antarmuka Proses <i>Association Rule</i>	42
3.3.1 implementasi Sistem	44
3.3.2 Implementasi Basis Data.....	44
3.3.3 Komponen Pengembangan Antarmuka	45
3.3.3.1 Antarmuka Koneksi Database.....	45
3.3.3.2 Antarmuka <i>Query</i> Database	46
3.3.3.3. Antarmuka Proses <i>Association Rule</i>	49
BAB IV HASIL DAN PEMBAHASAN	56
4.1 Pengujian Sistem.....	57
4.1.1 Pengujian Normal	57
4.1.1.1 Pengujian menggunakan data pada Tabel 2.1	59
4.1.1.2 Pengujian menggunakan keseluruhan data transaksi	60
4.1.1.3 Pengujian menggunakan <i>dataset</i> transaksi berdasar jenis pekerjaan “mahasiswa”	62
4.1.1.4 Pengujian menggunakan <i>dataset</i> transaksi berdasarkan jenis pekerjaan “karyawan”	65
4.1.2 Pengujian Tidak Normal.....	68
4.1.2.1 Proses Koneksi Database.....	68
4.1.2.2 Proses Input Query	68
4.1.2.3 Proses Pembentukan Aturan Asosiasi	69
4.1.3 Analisis Hasil Pengujian.....	70

BAB V KESIMPULAN DAN SARAN	73
5.1 Kesimpulan	73
5.2 Saran	73

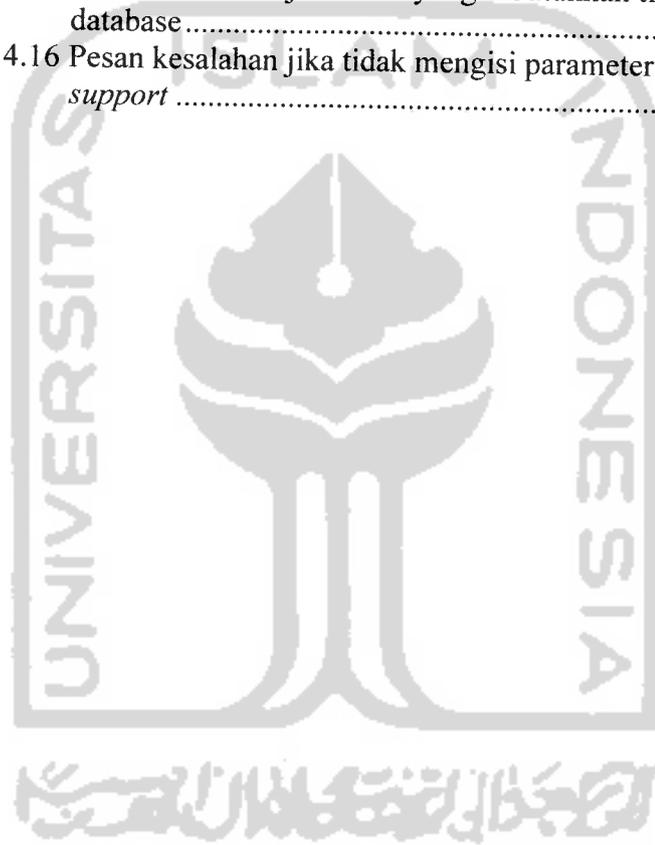
DAFTAR PUSTAKA
LAMPIRAN



DAFTAR GAMBAR

Gambar 2.1 Bidang ilmu yang mempengaruhi data mining	8
Gambar 2.2 <i>Pseudo Code</i> Algoritma Apriori	15
Gambar 2.3 <i>Pseudo Code</i> Pembentukan Kandidat Itemset	16
Gambar 3.1 DFD Level 0.....	29
Gambar 3.2 DFD Level 1.....	30
Gambar 3.3 DFD level 2 Proses 1 (Proses Koneksi).....	31
Gambar 3.4 DFD level 2 Proses 2 (Proses Input Query).....	32
Gambar 3.5 DFD level 2 proses 3(Proses Pembentukan Frequent Itemset).....	33
Gambar 3.6 DFD Level 2 proses 4 (Proses Pembentukan Assosiation Rule)	35
Gambar 3.7 Relasi Antar Tabel dalam Database	39
Gambar 3.8 Antarmuka Proses Koneksi	40
Gambar 3.9 <i>Tabsheet Dataset</i>	41
Gambar 3.10 <i>Form log</i>	42
Gambar 3.11 <i>Tabsheet</i> Proses Asosiasi	43
Gambar 3.12 Pembuatan Basis Data menggunakan Navicat 8 for MySQL	44
Gambar 3.13 Antarmuka Koneksi Database.....	45
Gambar 3.14 Prosedur untuk melakukan koneksi ke database.....	46
Gambar 3.15 Antarmuka Query Database <i>tabsheet</i> Editor <i>Query</i>	47
Gambar 3.16 Antarmuka Query Database <i>tabsheet</i> Hasil <i>Query</i>	47
Gambar 3.17 Prosedure untuk menjalankan <i>query</i> dan menampilkan jumlah data dan jumlah atribut.....	48
Gambar 3.18 Panel parameter <i>apriori</i>	49
Gambar 3.19 <i>Tabsheet Frequent Itemset</i>	50
Gambar 3.20 <i>Tabsheet</i> Aturan Asosiasi.....	50
Gambar 3.21 <i>Form Log</i>	51
Gambar 3.22 prosedur untuk proses <i>assosiation rule</i> , tampil <i>form log</i> , tampil <i>frequent itemset</i> pada <i>tabsheet frequent itemset</i> dan tampil aturan pada <i>tabsheet</i> aturan asosiasi	51
Gambar 4.1 Input Parameter Koneksi.....	58
Gambar 4.2 <i>Frequent itemset</i> yang diperoleh dari data pada Tabel 2.1	59
Gambar 4.3 Aturan asosiasi yang diperoleh dari data pada Tabel 2.1	60
Gambar 4.4 <i>Script Query</i> pengujian menggunakan keseluruhan data transaksi	60
Gambar 4.5 <i>frequent itemset</i> pengujian menggunakan keseluruhan data transaksi	61
Gambar 4.6 Beberapa aturan asosiasi yang diperoleh dari pengujian menggunakan keseluruhan data transaksi	62
Gambar 4.7 <i>Script query</i> pengujian menggunakan <i>dataset</i> transaksi berdasarkan jenis pekerjaan “mahasiswa”	63
Gambar 4.8 <i>frequent itemset</i> pengujian menggunakan <i>dataset</i>	

transaksi berdasarkan jenis pekerjaan “mahasiswa”	64
Gambar 4.9 Aturan asosiasi yang diperoleh dari pengujian menggunakan <i>dataset</i> transaksi berdasarkan jenis pekerjaan “mahasiswa”	65
Gambar 4.10 <i>script query</i> pengujian menggunakan <i>dataset</i> transaksi berdasarkan jenis pekerjaan “karyawan”	65
Gambar 4.11 <i>frequent itemset</i> pengujian menggunakan <i>dataset</i> transaksi berdasarkan jenis pekerjaan “karyawan”	66
Gambar 4.12 Aturan asosiasi yang diperoleh dari pengujian menggunakan <i>dataset</i> transaksi berdasarkan jenis pekerjaan “karyawan”	67
Gambar 4.13 Pesan kesalahan jika tidak mengisi nama database.....	68
Gambar 4.14 Pesan kesalahan jika tidak ada <i>query</i> yang akan diproses	68
Gambar 4.15 Pesan kesalahan jika tabel yang dibutuhkan tidak ada dalam database.....	69
Gambar 4.16 Pesan kesalahan jika tidak mengisi parameter batas bawah <i>support</i>	69



DAFTAR TABEL

Tabel 2.1 Tabel Hasil Transaksi	17
Tabel 2.2 Kandidat One Item Set.....	17
Tabel 2.3 Frequent Itemset 1 (L)	18
Tabel 2.4 Kandidat Two Item Set	19
Tabel 2.5 Kandidat Two Item Set Dan Support.....	19
Tabel 2.6 Frequent Item Set 2 (L).....	20
Tabel 2.7 Hasil Gabung L * L Sebelum Pruning.....	21
Tabel 2.8 Kandidat Three Itemset.....	21
Tabel 2.9 Kandidat Three Itemset Dan Support	22
Tabel 2.10 Frequent Itemset 3 (L)	22
Tabel 3.1 Struktur tabel transaksi	36
Tabel 3.2 Struktur tabel transaksi_detail.....	36
Tabel 3.3 Struktur tabel barang.....	37
Tabel 3.4 Struktur tabel harga.....	37
Tabel 3.5 Struktur tabel member.....	37
Tabel 3.6 Struktur tabel tmp_itemsets	38
Tabel 3.7 Struktur tabel tmp_aturan	38
Tabel 4.1 Detail data database super_2400_fixTabel	58

BAB I

PENDAHULUAN

1.1 Latar Belakang Masalah

Teknologi Informasi selalu mengalami kemajuan yang sangat pesat sehingga mendorong organisasi maupun perusahaan yang memanfaatkannya mencari data yang terbaru dan terakurat. Produsen perangkat lunak juga berlomba-lomba untuk memberikan kemudahan-kemudahan dan keuntungan dengan penyimpanan data atau informasi dalam bentuk *digital* yang mempunyai ukuran yang sangat besar. Sehingga data atau informasi yang dapat disimpan menjadi sangat banyak dan juga menyebabkan data tersebut menjadi sangat kompleks.

Ini terjadi pada semua bidang, mulai dari data yang sederhana tetapi sangat besar (seperti database transaksi supermarket, penggunaan kartu kredit, sampai ke data statistik pemerintahan) hingga pada data-data yang kompleks (seperti database molekul-molekul, peta geologi, rekam medis kesehatan). Perkembangan internet juga memberikan andil besar dalam akumulasi penyimpanan data ini.

Dikarenakan data yang ada sangat banyak dan kompleks maka pengambilan informasi dari data tersebut agar bisa bermanfaat, menjadi lebih kompleks dan rumit dan sering menyebabkan data tersebut dalam kondisi yang disebut "*rich of data but poor of information*" karena data yang terkumpul tidak

dapat digunakan untuk aplikasi yang berguna. Tidak jarang kumpulan data tersebut dibiarkan begitu saja seakan-akan menjadi sebuah "kuburan data" (*data tombs*).

Hal inilah yang melatarbelakangi lahirnya *data mining*. *Data mining* digunakan untuk menemukan pola-pola di dalam data dan bahkan dapat menyimpulkan aturan dari data tersebut. Sebagai contoh, seorang manajer pemasaran bertanya "apakah penjualan sabun cuci berhubungan dengan penjualan pembersih lantai?". dengan bersumber dari data transaksi penjualan, data mining dapat menemukan pola-pola dan asosiasi antara data tersebut dan dapat menemukan fakta yang ada pada data tersebut bahkan juga fakta yang tersembunyi bahkan yang tidak diperhatikan.

Dengan mengetahui pola-pola yang ada di dalam data, aturan-aturan yang dihasilkan akan bermanfaat dalam pengambilan keputusan yang menguntungkan.

1.2 Rumusan Masalah

Dari latar belakang yang dikemukakan di atas, akan dibangun suatu aplikasi data mining yang dapat menemukan asosiasi atau hubungan antara barang satu dengan barang lainnya menggunakan salah satu teknik *data mining* yaitu *association rules*, menggunakan algoritma *apriori*.

1.3 Batasan Masalah

Dalam pengerjaan tugas akhir ini untuk mengatasi permasalahan yang ada, maka penyusun membatasi permasalahan sebagai berikut :

1. Teknik *data mining* yang digunakan adalah *association rules* dengan bentuk *market basket analysis*.
2. Algoritma yang digunakan dalam membangun aplikasi data mining ini adalah dengan menggunakan algoritma *apriori*.
3. Sistem menerima input yang berupa data sintetis transaksi penjualan sebuah supermarket.

1.4 Tujuan Penelitian

Tujuan dari penelitian ini adalah untuk menerapkan salah satu teori *data mining* yang ada yaitu aturan asosiasi, serta menggunakan algoritma *apriori*. Aplikasi *data mining* yang akan dikembangkan berfungsi menemukan aturan-aturan atau pola-pola tersembunyi yang dapat disimpulkan dari data yang dimiliki sehingga dihasilkan informasi yang berguna bagi perusahaan dan organisasi khususnya pusat perbelanjaan. Contohnya untuk menentukan peletakan barang-barang, diskon, barang-barang apa saja yang diminati oleh konsumen dengan pekerjaan tertentu dan lain sebagainya.

1.5 Manfaat Penelitian

Manfaat dari penelitian ini adalah untuk membantu perusahaan dan organisasi khususnya pusat perbelanjaan yang membutuhkan pemrosesan terhadap data transaksi penjualan yang terletak di database yang mereka miliki dan menemukan informasi terpendam yang nantinya akan menghasilkan aturan-aturan yang akan bermanfaat terhadap perusahaan.

1.6 Metodologi Penelitian

Metode penelitian yang digunakan dalam penyusunan tugas akhir ini adalah :

1. Studi literatur

Pengumpulan informasi, pengetahuan serta teori-teori mengenai *Knowledge discovery in database*(KDD), *data mining*, aturan asosiasi, algoritma *apriori* yang didapat dari buku-buku, artikel, jurnal, dan website yang berhubungan dengan penulisan tugas akhir ini.

2. Analisis Kebutuhan

Pengumpulan data yang dibutuhkan dalam pengembangan aplikasi didapat dari website yang berhubungan dengan penelitian.

3. Perancangan dan implementasi

Pengembangan sistem dilaksanakan setelah data yang berhubungan dengan penelitian ini terkumpul secara lengkap. Kemudian dilaksanakan proses perancangan sistem yang meliputi desain struktur data, arsitektur

sistem, representasi interface dan algoritma program, perancangan komponen dan perancangan antarmuka untuk pemakai.

Setelah proses di atas selesai maka sistem akan diimplementasikan serta akan dilakukan uji coba terhadap sistem.

4. Pengujian Kinerja

Sistem yang telah berhasil diimplementasikan akan dianalisis, analisis yang dilakukan akan lebih ditekankan pada *association rules mining*.

1.7 Sistematika Penulisan

Untuk memberi gambaran menyeluruh mengenai masalah yang akan dibahas, maka digunakan sistematika penulisan tugas akhir.

Sistematika penulisan tugas akhir ini adalah sebagai berikut :

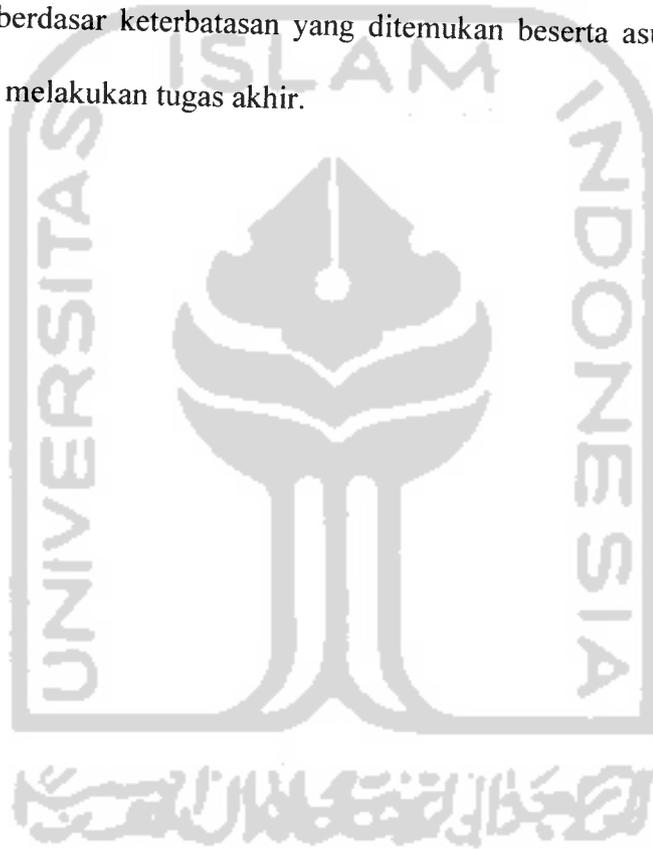
BAB I Pendahuluan. Bab ini berisi latar belakang permasalahan yang terjadi pada objek yang diambil, rumusan masalah, tujuan penelitian, manfaat penelitian, metodologi penelitian dan sistematika penulisan.

BAB II Landasan teori. Bab ini memuat gambaran umum tentang *data mining*, aturan asosiasi, *market basket analysis* dan algoritma *apriori*.

BAB III Metodologi. Menganalisis metode yang digunakan serta hasilnya (masukan sistem, kebutuhan proses, keluaran sistem, kebutuhan perangkat lunak dan perangkat keras, kebutuhan fungsi, antarmuka sistem), metode perancangan perangkat lunak yang akan dipakai dan hasil perancangan perangkat lunak dan batasan perangkat lunak dan dokumentasi implementasi perangkat lunak.

BAB VI Hasil dan pembahasan. Bab ini memuat dokumentasi hasil pengujian terhadap perangkat lunak yang dibandingkan kebenaran dan kesesuaiannya dengan kebutuhan perangkat lunak yang dituliskan dalam bagian sebelumnya yang kemudian dianalisis.

BAB V Simpulan dan saran. Bab ini memuat kesimpulan-kesimpulan dari proses pengembangan perangkat lunak dan saran-saran yang perlu diperhatikan berdasar keterbatasan yang ditemukan beserta asumsi-asumsi yang dibuat selama melakukan tugas akhir.



BAB II

DASAR TEORI

2.1 Dasar Data mining

2.1.1 Definisi Data Mining

Data mining adalah eksplorasi dan analisis terhadap sejumlah data dengan tujuan untuk menemukan pola dan aturan yang sangat penting.

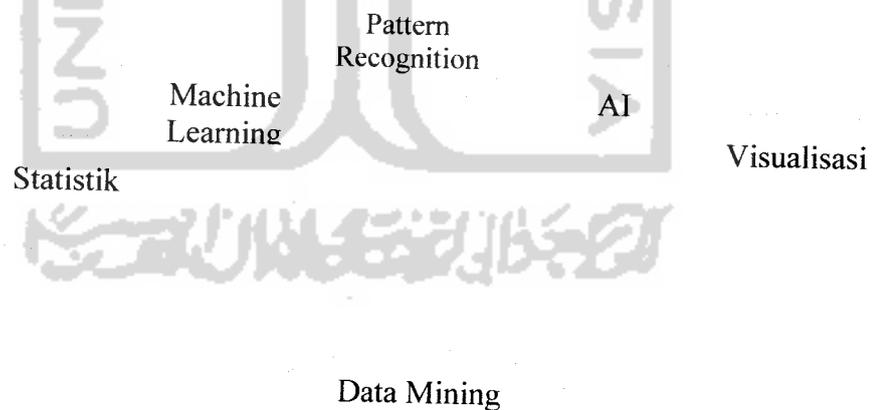
Beberapa pemahaman dari sekian banyak pengertian *data mining* adalah :

1. *Data mining* merupakan serangkaian proses untuk menggali nilai tambah dari suatu kumpulan data berupa pengetahuan yang selama ini tidak diketahui secara manual [PRA03].
2. *Data mining* adalah proses untuk menemukan korelasi-korelasi baru yang ber arti, pola-pola dan kecenderungan-kecenderungan dengan menyaring data yang sangat banyak yang disimpan dalam tempat penyimpanan menggunakan teknik pengenalan pola seperti juga statistik dan teknik-teknik matematika [LAR05].
3. *Data mining* seperti yang diketahui adalah penjabaran secara implisit, yang sebelumnya tidak diketahui, yang sangat mungkin berguna sebagai informasi dari data. Ini mencakup beberapa pendekatan teknik yang berbeda, seperti pengelompokan (*clustering*), peringkasan data, memahami aturan klasifikasi, mencari depedensi jaringan, memahami perubahan-perubahan, dan juga mendeteksi kelainan [WIT05].

Dari beberapa pengertian di atas dapat disimpulkan bahwa *data mining* adalah analisis terhadap dataset untuk menemukan hubungan dan menyimpulkan data dengan cara yang jelas, dimana hasilnya dapat dimengerti dan berguna terhadap pemilik data.

2.1.2 Bidang Ilmu Yang Mempengaruhi Data Mining

Data mining merupakan penerapan bidang ilmu interdisipliner, statistik, *database*, *machine learning*, *pattern recognition*, kecerdasan buatan, dan visualisasi. Masing-masing memiliki peranan penting di dalam *data mining* yang saling berhubungan. Sehingga sangat sulit untuk memberikan batasan yang jelas di antara masing-masing disiplin ilmu tersebut dengan *data mining*.



Gambar 2.1 Bidang ilmu yang mempengaruhi *data mining*

Hal yang perlu diperhatikan adalah bahwa *data mining* berbeda dengan *query tools*. *Query* dan *data mining* merupakan dua hal yang saling melengkapi.

Keberadaan *data mining* bukan untuk menggantikan *query*, tetapi menambahkan beberapa tambahan yang berarti. Jika menggunakan *query* sederhana maka informasi yang dapat diakses sekitar 80% dari data yang ada di dalam *database*, sedangkan 20% lagi akan menjadi informasi tersembunyi yang memerlukan teknik-teknik khusus dalam mengaksesnya yaitu dengan *data mining* [ADR96].

2.1.3 Market Basket Analysis

Market basket analysis adalah salah satu bentuk pengaplikasian aturan asosiasi yang biasa digunakan dalam marketing. Tujuan umum dari *market basket analysis* adalah untuk menemukan item apa saja yang berada dalam transaksi para konsumen.

Istilah *market basket analysis* ini sendiri muncul dari ide ketika pelanggan memasukkan item dalam keranjang belanja (*market basket*) selama transaksi. Mengetahui item apa yang dibeli oleh para konsumen sebagai suatu kelompok (*itemset*) adalah informasi yang berharga terhadap penjualan suatu perusahaan. Suatu toko dapat menggunakan informasi ini untuk menempatkan produk-produk yang “sering” dibeli secara bersama-sama dalam suatu area yang berdekatan atau area yang sama, sedangkan untuk penjualan lewat internet dapat menggunakannya untuk menentukan gambaran katalog, desain formulir pemesanan barang serta isi setiap halaman web. Dan penjual dapat langsung menggunakan hasil *market basket analysis* untuk menentukan item baru apa yang dapat ditawarkan secara langsung kepada pelanggan utama mereka.

2.1.4 Teknik-teknik Data Mining

Terdapat beberapa teknik *data mining* yang dapat digunakan dalam pemecahan masalah. Meskipun demikian pada dasarnya teknik *data mining* dapat dibedakan atas 2 bagian berdasarkan prosesnya :

1. *Supervised Learning*

Dalam *supervised learning*, output dari pembelajaran yang diharapkan telah diketahui sebelumnya. Teknik-teknik yang termasuk ke dalam bagian ini adalah klasifikasi (*classification*), regresi (*regression*), dan lainnya.

2. *Unsupervised Learning*

Berbeda dengan *supervised learning*, dalam *unsupervised learning* tidak dapat ditentukan output seperti apa yang diharapkan dari pembelajaran yang dilakukan. Teknik-teknik yang termasuk ke dalam bagian ini adalah *clustering*, *decision tree*, aturan asosiasi (*association rule*), dan lainnya.

2.2 Association Rules Mining

Association rules mining berfungsi untuk menemukan asosiasi antar variabel, korelasi atau suatu struktur di antara item atau objek-objek didalam *database* transaksi, *database* relasional, maupun pada penyimpanan informasi lainnya.

Ide awal dari *asosociation rules mining* berasal dari aplikasi yang berhubungan dengan *retail (market basket) data*. Ilustrasi dari *association rules mining* adalah sebagai berikut, terdapat sebuah *database* transaksi supermarket,

dimana terdapat kemungkinan ditemukannya pola “10 persen dari pembeli membeli bir dan popok untuk bayi secara bersamaan,” dan contoh untuk dataset analisis weblog, pola yang mungkin adalah “jika seseorang mengunjungi website CNN, terdapat kemungkinan sebesar 60% orang tersebut juga mengunjungi website ABC News pada bulan yang sama.” Pada dua kasus tadi, pola yang ditemukan berpotensi menghasilkan potongan informasi yang menarik dan dibutuhkan oleh perusahaan yang terkait.

2.2.1 Aturan dari Association Rules Mining

Sebuah aturan asosiasi adalah implikasi dari $X \Rightarrow Y$, dimana $X \subset I$, $Y \subset I$, dan $X \cap Y = \phi$. Aturan $X \Rightarrow Y$ berada di dalam himpunan transaksi D dengan kepercayaan (*confidence*) c , jika $c\%$ dari transaksi dalam D yang ada X terdapat juga Y . Aturan $X \Rightarrow Y$ memiliki dukungan (*support*) s di dalam set transaksi D , jika $s\%$ transaksi dalam D terdapat $X \cup Y$. I adalah *itemset*. [TAN06].

2.2.1.1 Itemset

Itemset adalah kumpulan dari item-item. Setiap item adalah merupakan sebuah nilai atribut. Contoh dalam *market basket*, sebuah itemset berisi kumpulan dari produk-pruduk, seperti sikat gigi, pasta gigi, *mouthwash*. Jika dalam *customer demographic exploration*, sebuah itemset berisi kumpulan dari nilai-nilai atribut seperti $\{\text{gender} = \text{'perempuan'}, \text{pendidikan} = \text{'sarjana'}\}$.

Setiap itemset mempunyai ukuran, yaitu jumlah dari item yang terdapat dalam itemset. Ukuran dari *itemset* {sikat gigi, pasta gigi, mouthwash} adalah 3.

Frequent itemsets adalah itemset-itemset yang sering ditemukan dalam dataset dan memiliki nilai support lebih atau sama besar dengan nilai minimum *support* yang ditentukan.[TAN05]

2.2.1.2 Support

Dimisalkan suatu aturan $R : X \Rightarrow Y$. *Support itemset* (I) adalah jumlah transaksi T yang didalamnya terdapat *itemset*. Rumus untuk mendefinisikan *support* [ARE04] :

$$\text{Support}(X \Rightarrow Y) = \text{support}(\{X\} \cup \{Y\}) \quad (2.1)$$

2.2.1.3 Confidence

Confidence mengukur seberapa besar ketergantungan suatu *item* dengan item yang lainnya. Rumus dalam mendefinisikan *confidence* [ARE04] :

$$\text{Confidence}(X \Rightarrow Y) = \frac{\text{support}(\{X\} \cup \{Y\})}{\text{support}(\{X\})} \quad (2.2)$$

2.2.1.4 Rule

Sebuah aturan (*rule*) biasanya terdiri dari dua bagian, yaitu kondisi (*antecedent*) dan hasil (*consequent*). Dan biasanya disajikan dalam pernyataan sebagai berikut :

Jika **kondisi** maka **hasil**

Bentuk invers dari aturan tersebut adalah dengan menukar hasil dengan kondisi, yang berarti mencari peluang jika kondisi yang dimiliki adalah hasil yang ingin dilihat, dan hasil yang ada dijadikan sebagai kondisi. Sebagai contoh mengenai perhitungan dan bagaimana aturan ini dapat dimengerti adalah dengan menggunakan data transaksi, misalkan dari contoh data penjualan bir dan popok, yaitu terdapat :

500.000 transaksi

20.000 transaksi terdapat popok

30.000 transaksi terdapat bir

10.000 transaksi terdapat keduanya (bir dan popok)

Aturan yang ingin dilihat: jika dibeli {popok} maka pembeli juga akan membeli {bir}. Dalam contoh ini bir dan popok terdapat bersamaan sebesar 2% ($10.000 / 500.000$), maka *support* aturan tersebut adalah sebesar 2%. Karena dari 20.000 transaksi terdapat popok dan 10.000 di antaranya terdapat bir jika orang tersebut membeli popok, maka ketergantungan ada bir jika orang membeli popok adalah 50% ($10.000 / 20.000$), yang berarti *confidence* dari aturan tersebut adalah 50%.

Invers dari aturan tersebut adalah :

Jika dibeli {bir} maka pembeli juga membeli {popok}, yang memiliki *confidence* sebesar 33% (10.000 / 30.000), dan *support* yang aturannya sama dengan aturan awalnya. Jadi *support* tidak bergantung dari pada arah aturan implikasinya, melainkan hanya bergantung dari *itemset* dari aturan tersebut.

2.2.2 Algoritma Apriori

Algoritma *apriori* merupakan algoritma pertama dan sering digunakan untuk menemukan aturan asosiasi di dalam aplikasi *data mining* dengan teknik aturan asosiasi.

Tujuan dari *algoritma apriori* adalah untuk menemukan aturan (*rule*) yang memenuhi minimum *support* yang telah ditetapkan sebelumnya dan memenuhi nilai *confidence* yang disyaratkan. Tugas ini secara teoritis akan terlihat kompleks, karena akan ditemukan *itemset* yang berbentuk fungsi eksponensial dari variabel yang ada, sehingga dataset nya akan sangat besar. Tetapi di dalam dataset dunia nyata, sebagian kasus yang ada relatif ditemukan hanya beberapa *itemset*, sebagai contoh kebanyakan konsumen hanya akan membeli sebagian kecil *item* dari keseluruhan *item* yang ada di supermarket.

Algoritma *apriori* yang digunakan dalam menemukan aturan asosiasi dapat dibagi menjadi dua fase, yang pertama untuk menemukan *frequent itemsets* dan kemudian membentuk aturan (*rule*) dari *frequent itemsets* yang ditemukan. Potongan *pseudo code* dari algoritma *apriori* adalah seperti dibawah ini :

```

Ck: Candidate itemset of size k
Lk : frequent itemset of size k

L1 = {frequent items};

for (k = 1; Lk !=∅; k++) do begin
    Ck+1 = candidates generated from Lk;
    for each transaction t in database do
        increment the count of all candidates in Ck+1 that
        are contained in t
    Lk+1 = candidates in Ck+1 with min_support
    end
return  $\cup_k L_k$ ;

```

Gambar 2.2 *Pseudo Code* Algoritma Apriori

Dari *pseudo code* di atas dapat dilihat algoritma *apriori* melakukan *generate* untuk membentuk kandidat-kandidat *itemsets*. Terdapat dua langkah untuk membentuk kandidat-kandidat *itemsets*:

- Langkah 1 : *self-joining* L_k

$$L_k * L_k = \{X \cup Y \text{ where } X, Y \in L_k, |X \cap Y| = k - 1\}$$

- Langkah 2 : *pruning* (jika terdapat *item sets* yang tidak *frequent*, maka *item set* tersebut tidak akan dibentuk atau dihapus).

Contoh pembentukan kandidat itemset:

Self-joining : $L_3 = \{abc, abd, acd, ace, bcd\}$

gabung $L_3 * L_3$ dan akan didapat *itemset* $\{a,b,c,d\}$ dari abc dan abd, $\{a,b,c,e\}$ dari abc dan ace, $\{a,c,d,e\}$ dari acd dan ade, $\{a,b,c,d,e\}$ dari abd dan ace.

Pruning :

$\{a,b,c,e\}$ dihapus karena abe, bce tidak ada pada L_3 , $\{a,c,d,e\}$ dihapus karena ade tidak ada pada L_3 , $\{a,b,c,d,e\}$ dihapus karena ade, bce, abe tidak ada pada L_3 .

```

langkah 1: self-joining  $L_{k-1}$ 
    insert into  $C_k$ 
    select  $p.item1, p.item2, \dots, p.item_{k-1}, q.item_{k-1}$ 
    from  $L_{k-1} p, L_{k-1} q$ 
    where  $p.item1=q.item1, \dots, p.item_{k-2}=q.item_{k-2},$ 
     $p.item_{k-1} < q.item_{k-1}$ 

langkah 2: pruning
     $\forall$  itemsets  $c$  in  $C_k$  do
     $\forall$   $(k-1)$ -subsets  $s$  of  $c$  do
    if ( $s$  is not in  $L_{k-1}$ ) then delete  $c$  from  $C_k$ 
  
```

Gambar 2.3 Pseudo Code Pembentukan Kandidat Itemset

Ilustrasi dari cara kerja algoritma apriori akan lebih mudah dimengerti dengan menampilkan data transaksi pada sebuah tabel. Terdapat sebuah tabel data hasil transaksi, seperti yang ditunjukkan pada Tabel 2.1. [AGR94].

Tabel 2.1 Tabel Hasil Transaksi

id	Items
10	A, C, D
20	B, C, E
30	A, B, C, E
40	B, E

Pada Tabel 2.1 di atas merupakan data hasil transaksi, kemudian dari Tabel 2.1 tersebut diproses untuk mencari kandidat *one itemset* (C_1) apa saja yang terdapat didalam tabel transaksi. Akan dicari aturan yang memiliki minimum support = 2.

Pembentukan kandidat *one itemset* (C_1) pada proses iterasi pertama ini sederhana yaitu seluruh item tunggal dari tabel 2.1 merupakan kandidat dari *one itemset*. Kemudian dilakukan proses cari nilai support masing-masing kandidat. Kandidat *one itemset* yang ditemukan, ditunjukkan pada Tabel 2.2.

Tabel 2.2 Kandidat One Item Set

Itemset	suport
{A}	2
{B}	3
{C}	3
{D}	1
{E}	3

Dari Tabel 2.2 diperoleh 5 kandidat *one itemset*., Setelah didapatkan support maka hapus *one itemset* yang nilai *supportnya* tidak memenuhi minimum *support* yang telah disyaratkan. Setelah *one item set* yang tidak memenuhi minimum *support* dihapus, didapatkanlah *frequent item set* pertama (L_1). *item set* pertama (L_1) ditunjukkan pada Tabel 2.3.

Tabel 2.3 Frequent Itemset 1 (L_1)

Itemset	support
{A}	2
{B}	3
{C}	3
{E}	3

Proses selanjutnya adalah membentuk kandidat *two itemset* (C_2) dari *frequent itemset* (L_1) pada Tabel 2.3. Proses pembentukan kandidat *two itemset* dilakukan algoritma *apriori* melalui dua tahap yaitu gabung dan pruning.

Proses penggabungan (*self-joining*) dengan menggabungkan $L_1 * L_1$ [KAN03] Pada $k = 1$, operasi yang dilakukan adalah penggabungan secara sederhana antara L_1 . Hasil dari penggabungan didapatkan 6 kandidat *two itemset*.

Kemudian pruning hasil penggabungan, dengan cara cek kandidat *two itemset* (C_2) terhadap *frequent itemset* 1 (L_1). Apabila pada kandidat *itemset* terdapat *itemset* yang tidak frequent atau tidak terdapat pada Tabel 2.3, maka hapus kandidat *itemset* tersebut.

Pada Tabel 2.4 semua kandidat itemsetnya (C_2) *frequent* dengan L_1 pada Tabel 2.3 sehingga tidak ada kandidat *itemset* 1 yang dihilangkan. Didapatlah hasil seperti yang ditunjukkan pada Tabel 2.4.

Tabel 2.4 Kandidat Two Item Set

Itemset
{A, B}
{A, C}
{A, E}
{B, C}
{B, E}
{C, E}

Dari Tabel 2.4 dilakukan pencarian nilai support dari kandidat *two itemset*, seperti ditunjukkan pada Tabel 2.5

Tabel 2.5 Kandidat Two Item Set Dan Support

Itemset	suport
{A, B}	1
{A, C}	2
{A, E}	1
{B, C}	2
{B, E}	3
{C, E}	2

Untuk mendapatkan *frequent itemset 2* (L_2), kandidat *two itemset* (C_2) diproses dengan menghapus kandidat *two itemset* yang tidak memenuhi minimum support. Hasil dari penghapusan kandidat *two itemset* yang tidak memenuhi minimum support akan membentuk *frequent itemset 2* (L_2). *Frequent itemset 2* (L_2) ditunjukkan pada Tabel 2.6

Tabel 2.6 Frequent Item Set 2 (L_2)

Itemset	suport
{A, C}	2
{B, C}	2
{B, E}	3
{C, E}	2

Proses dilanjutkan dengan mencari kandidat *three itemset* (C_3) dari Tabel 2.6. Pembentukan kandidat *three itemset* menggunakan $L_2 * L_2$. Proses penggabungan dilakukan terhadap *itemset* dengan item pertama yang sama terlebih dahulu, pada contoh ini adalah {B,C} dan {B,E}, dan didapat kandidat {B,C,E}. Kemudian gabung {A,C} dan {C,E} dan didapat kandidat {A,C,E}, dan kandidat sisa juga digabung dengan kandidat yang lain.

Tabel 2.7 Hasil Gabung $L_2 * L_2$ Sebelum Pruning

Itemset
{B, C, E}
{A, B, C}
{A, C, E}

Proses selanjutnya adalah lakukan pruning pada Tabel 2.7, seperti pada tahap pembentukan kandidat *two itemset* sebelumnya. Dari hasil pruning didapat bahwa kandidat {A,C,E} tidak *frequent* karena {A,E} tidak terdapat pada Tabel 2.6. Pada kandidat {A,B,C} juga tidak *frequent* karena {A,B} tidak terdapat pada Tabel 2.6. Kemudian kandidat yang tidak *frequent* tadi akan dihapus atau tidak dibentuk menjadi kandidat *three itemset*. Dan didapatlah {B,C,E} sebagai satu-satunya kandidat *three itemset*, karena seluruh anggotanya terdapat pada Tabel 2.6. Kandidat *three itemset* ditunjukkan pada tabel 2.8.

Tabel 2.8 Kandidat Three Itemset

Itemset
{B, C, E}

Kemudian cari nilai support dari {B,C,E}, dengan melakukan *scan* terhadap database dalam hal ini pada Tabel 2.1.

Tabel 2.9 Kandidat Three Itemset Dan Support

Itemset	support
{B, C, E}	2

Proses pembentukan *frequent item set* 3 (L_3) dengan cara menghapus kandidat *three itemset* yang tidak memenuhi minimum *support*, kemudian didapatlah {B,C,E} sebagai *frequent itemset* 3 (L_3) pada Tabel 2.10. Pada contoh data transaksi di atas, tidak dapat dibentuk kandidat *four itemset* lagi dari L_3 maka proses perulangan algoritma *apriori* berhenti.

Tabel 2.10 Frequent Itemset 3 (L_3)

Itemset	support
{B, C, E}	2

Dari *frequent itemsets* yang dihasilkan kemudian akan dicari kandidat aturan yang dapat disimpulkan dari masing-masing *frequent itemsets*. Pencarian aturan berdasarkan *frequent itemset*, akan didapat kemungkinan $2^k - 2$ aturan yang akan ditemukan, dengan tidak memasukkan $\phi \rightarrow L$ dan $L \rightarrow \phi$ [TAN06]. Sebagai contoh dari kemungkinan aturan yang dapat disimpulkan dari *frequent itemset* atau *large item set* {B, C, E}, yaitu $2^3 - 2$ adalah 6 aturan, nilai support dari aturan ini hanya berlaku pada kondisi AND saja.

$\{B, C\} \Rightarrow \{E\}$	2/2 , sehingga <i>confidence</i> 1
$\{B\} \Rightarrow \{C, E\}$	2/3, sehingga <i>confidence</i> 0.66
$\{C\} \Rightarrow \{B, E\}$	2/3, sehingga <i>confidence</i> 0.66
$\{B, E\} \Rightarrow \{C\}$	2/3, sehingga <i>confidence</i> 0.66
$\{C, E\} \Rightarrow \{B\}$	2/2, sehingga <i>confidence</i> 1
$\{E\} \Rightarrow \{B, C\}$	2/3, sehingga <i>confidence</i> 0.66

Setelah kandidat aturan disimpulkan, aturan yang tidak memenuhi minimum *confidence* dihilangkan. Dianggap *confidence* yang ingin dicari adalah bernilai 1, sehingga yang dihasilkan adalah aturan yang telah memenuhi syarat nilai *confidence* yang diinginkan, yaitu:

$$\{B, C\} \Rightarrow \{E\} \text{ confidence } 1$$

$$\{C, E\} \Rightarrow \{B\} \text{ confidence } 1$$

BAB III

METODOLOGI

3.1.1 Metode Analisis

Metode analisis merupakan langkah penting dalam perancangan perangkat lunak. Langkah ini sangat mempengaruhi perancangan yang dibuat beserta implementasinya.

Metode analisis yang digunakan adalah metode analisis dengan pendekatan terstruktur, sehingga hasil analisis dari sistem yang dikembangkan menghasilkan sistem yang strukturnya dapat didefinisikan dengan baik dan jelas.

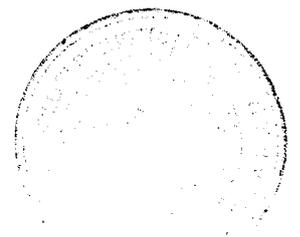
3.1.2 Hasil Analisis

Berdasarkan analisis yang telah dilakukan maka dapat diketahui apa saja yang menjadi masukan sistem, keluaran sistem, antarmuka yang diinginkan dan fungsi atau metode yang digunakan oleh sistem, sehingga sistem yang dibuat nantinya sesuai dengan yang diharapkan.

3.1.3 Masukan Sistem

Parameter yang digunakan sebagai masukan sistem adalah sebagai berikut:

1. Parameter koneksi, yaitu :
 - a. *Host* atau IP dari *server*.
 - b. *Port server*.



c. Nama user.

d. Password.

e. Database .

2. Input *dataset* yang berupa data transaksi penjualan, yaitu:

a. Data transaksi : berisi data transaksi yang dilakukan.

b. Data transaksi_detail : berisi detail barang yang dibeli dari tiap transaksi.

c. Data barang : berisi id dan nama barang.

d. Data member : berisi data member yang melakukan transaksi, digunakan jika ingin melakukan pembatasan data yang akan di proses. Misalnya, data yang akan diproses hanya transaksi yang dilakukan oleh *member* yang mempunyai pekerjaan mahasiswa.

e. Data harga : berisi harga dari barang, dapat digunakan untuk melakukan pembatasan data yang akan diproses.

3. Parameter apriori :

a. Batas bawah *support*.

Jumlah minimum banyaknya *itemset* terdapat pada transaksi agar *itemset* tersebut dapat menjadi *frequent itemset*.

b. Minimal *confidence*.

Jumlah minimal *Confidence* (seberapa besar ketergantungan suatu item dengan item yang lainnya) dari aturan, sehingga aturan yang memenuhi akan ditampilkan pada *user*.

3.1.4 Keluaran Sistem

Keluaran sistem dari perangkat lunak yang dibuat berupa aturan-aturan asosiasi yang diperoleh setelah melakukan proses terhadap data transaksi penjualan dengan menggunakan algoritma *apriori*.

3.1.5 Kebutuhan Proses

Dalam sistem ini, dibutuhkan proses-proses yang harus dilakukan yaitu :

1. Proses koneksi, pada proses ini akan dibangun sebuah koneksi untuk menghubungkan *database* dengan sistem.
2. Input *dataset* yang berupa data transaksi penjualan.
3. Pembentukan *frequent itemset*.

Dalam proses ini data akan diproses sehingga akan diperoleh hanya item-item yang sering ditemukan dalam *dataset*. Dalam pembentukan *frequent itemset* ini terdapat juga beberapa proses, yaitu:

- a. Proses pembentukan kandidat, dalam proses ini akan diperoleh *itemset* dengan melakukan penggabungan item-item yang terdapat dalam *frequent itemset* sebelumnya dan melakukan penghitungan *support* dari item-item yang diperoleh.
- b. Proses pruning, dalam proses ini *itemset* yang diperoleh pada pembentukan kandidat *itemset* akan dicek apakah *itemset* tersebut sering ditemukan pada *frequent itemset* sebelumnya.

c. Proses validitas item, pada proses ini kandidat *itemset* yang diperoleh akan di cek apakah sudah sesuai dengan batas bawah *support*.

4. Pembentukan *association rule*.

Pada proses ini akan dilakukan proses pada *frequent itemset* terakhir atau yang tertinggi yang diperoleh, sehingga akan diperoleh aturan-aturan. Aturan-aturan yang diperoleh kemudian di cek apakah sesuai dengan minimal *confidence* yang diinginkan atau tidak, jika tidak maka tidak ditampilkan kepada *user*.

3.1.6 Kebutuhan Perangkat Lunak

Perangkat lunak yang dibutuhkan untuk mendukung dalam pembuatan atau pengoperasian program aplikasi ini, sebagai berikut :

1. Sistem operasi berbasis Microsoft windows XP Profesional SP2.
2. Kakas pemrograman Borland Developer Studio 2006.
3. MySQL Server 5.0 sebagai *database server*.
4. Navicat 8 for MySQL sebagai MySQL gui, dalam membuat table-tabel yang diperlukan.

3.1.7 Kebutuhan Perangkat Keras

Kebutuhan minimal yang digunakan untuk menjalankan sistem adalah :

1. *Processor* : AMD Athlon™ 64 X2 Dual Core atau yang setingkat.
2. *Hardisk* : 1GB.
3. *RAM* : 512MB.

3.1.8 Antarmuka Sistem

Antarmuka yang dikembangkan pada perangkat lunak ini berbasis grafis dengan menggunakan *icon-icon* yang *user friendly* dengan harapan akan memudahkan pengguna dalam menggunakannya.

3.2.1 Metode Perancangan

Metode perancangan yang digunakan dalam mengembangkan perangkat lunak ini yaitu perancangan sistem berupa metode perancangan berarah aliran data dengan menggunakan alat-alat pengembangan sistem berupa DFD (*Data Flow Diagram*).

3.2.2 Hasil Perancangan

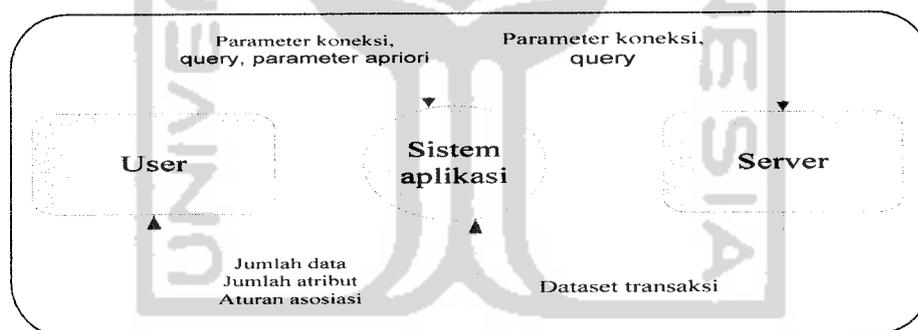
Berdasarkan hasil analisa permasalahan pada alur kerja metode *assosiation rule* menggunakan algoritma *apriori*, dapat diketahui masukan sistem, keluaran sistem dan proses-proses dalam sistem. Hasil perancangan ini dikelompokkan menjadi beberapa bagian yaitu Data Flow Diagram, Perancangan Basis Data, Perancangan Antarmuka Sistem.

3.2.3 Data Flow Diagram

Hasil pengamatan dari sistem diwujudkan dalam bentuk *data flow diagram* (DFD). *Data flow diagram* merupakan dokumen yang menggambarkan aliran data antar entitas yang ada dalam sistem.

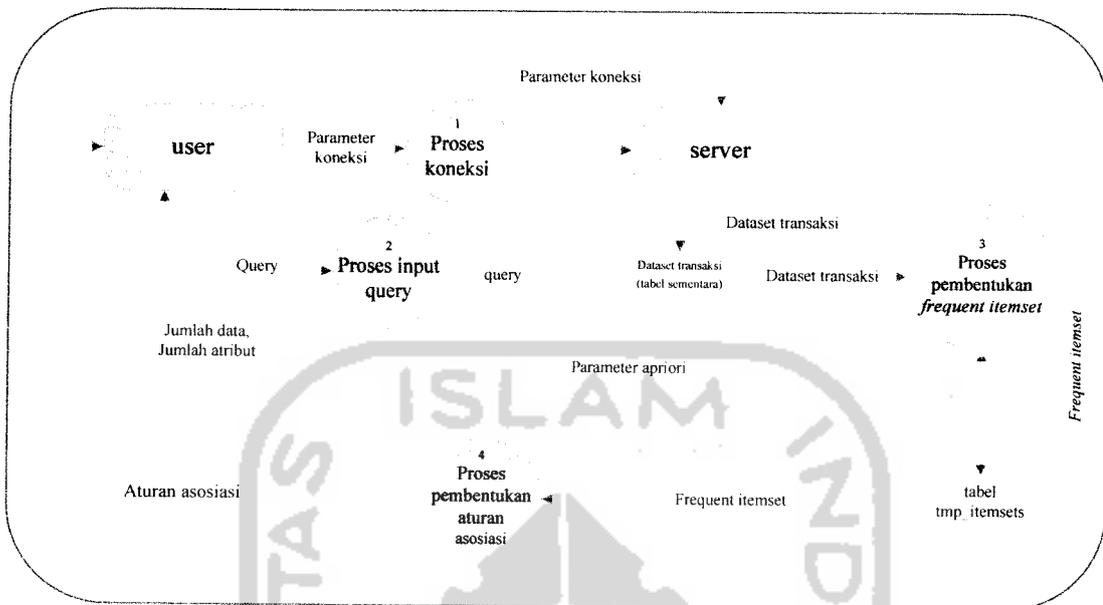
3.2.3.1 DFD level 0

Pada DFD level 0, yang pertama dilakukan adalah melakukan koneksi ke *database* dengan menginputkan parameter koneksi, kemudian akan dilakukan *query* terhadap *database* sehingga menghasilkan *dataset* transaksi yang akan diinputkan ke sistem. Proses dilanjutkan dengan menginputkan nilai parameter *apriori*, kemudian akan dilakukan pemrosesan terhadap *dataset* transaksi yang telah diinputkan ke sistem dengan menggunakan algoritma *apriori* yang akan menghasilkan aturan-aturan yang berguna dalam pengambilan keputusan. DFD level 0 ditunjukkan pada Gambar 3.1.



Gambar 3.1 DFD Level 0

3.2.3.2 DFD Level 1



Gambar 3.2 DFD Level 1

Proses utama terdiri dari empat proses, yaitu proses koneksi, proses input *query*, proses pembentukan *itemset* dan proses pembentukan aturan asosiasi.

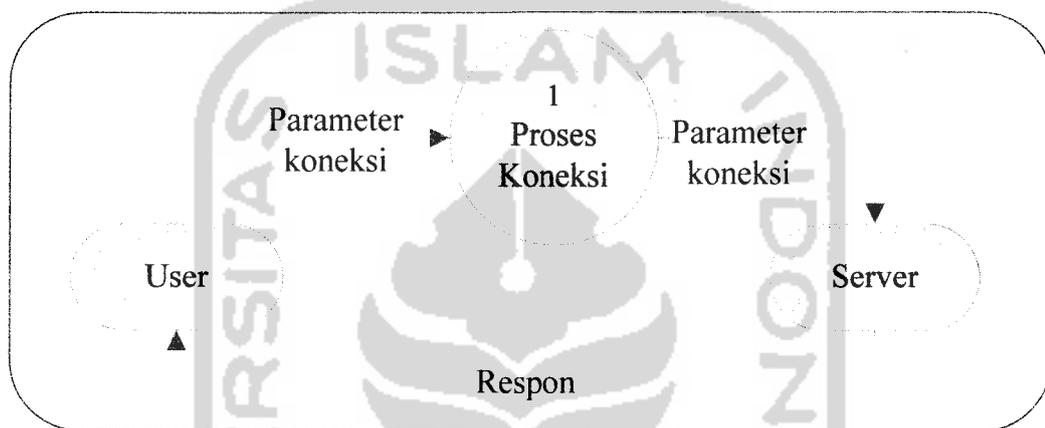
Pada proses koneksi akan dibangun sebuah koneksi untuk menghubungkan *database* dengan sistem. Koneksi dilakukan dengan memasukkan parameter yang diperlukan, yaitu *database*, *hostname*, *username*, *port* dan *password*.

Proses input *query* akan menghasilkan *dataset* transaksi dengan melakukan *query* ke *database*.

Proses pembentukan *frequent itemset* akan mencari item-item yang sering muncul dalam *dataset* transaksi.

Proses pembentukan aturan asosiasi akan menjalankan algoritma *apriori* terhadap *frequent itemset* yang terakhir dengan menggunakan batasan yang berupa batas bawah *support*, minimum *confidence*, sehingga diperoleh hasil berupa aturan-aturan asosiasi.

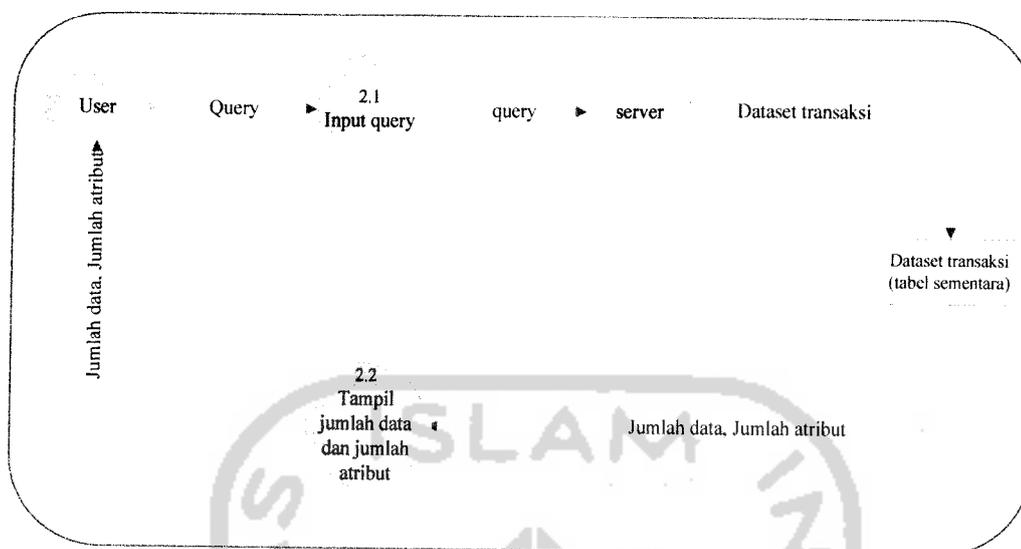
3.2.3.3 DFD Level 2 proses 1 (Proses Koneksi)



Gambar 3.3 DFD level 2 Proses 1 (Proses Koneksi)

Pada proses ini, *user* memasukkan parameter koneksi yang kemudian akan diteruskan pada server, dan server akan memberikan respon apakah koneksi ke *database* berhasil atau tidak, jika berhasil maka *database* akan terhubung dengan sistem dan siap untuk digunakan pada proses selanjutnya.

3.2.3.4 DFD level 2 Proses 2 (Proses Input Query)



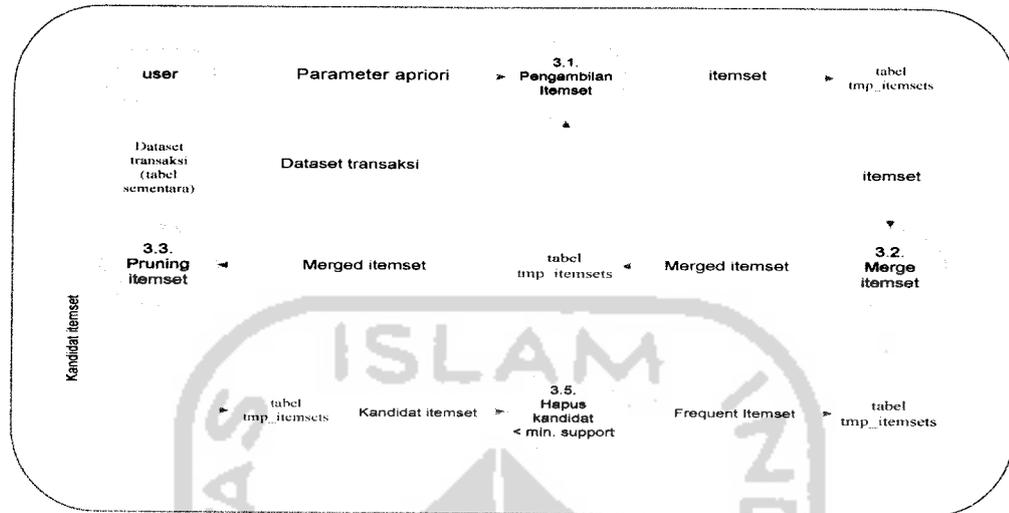
Gambar 3.4 DFD level 2 Proses 2 (Proses Input Query)

Pada DFD level 2 Proses 2 yaitu Proses Input *Query*, dijelaskan proses bagaimana memasukkan input berupa *query*, yang akan menghasilkan *dataset* transaksi, seperti yang ditunjukkan pada Gambar 3.4 di atas.

Untuk mengambil data yang terdapat di dalam tabel *database* sehingga diperoleh *dataset* transaksi, dilakukan *query* dengan menggunakan bahasa SQL.

Hasil dari *query* terhadap *database* yang akan diproses oleh sistem, ditampilkan banyaknya jumlah atribut dan jumlah data dari *dataset* transaksi yang akan diproses. Selanjutnya akan menuju proses pembentukan *frequent itemset* yang akan menghasilkan *frequent itemset*, yang menjadi input proses pembentukan aturan asosiasi, yang akan mencari aturan yang dapat disimpulkan dari input *database*.

3.2.3.5 DFD level 2 proses 3 (Proses Pembentukan *Frequent Itemset*)



Gambar 3.5 DFD level 2 proses 3(Proses Pembentukan *Frequent Itemset*)

Pada Gambar 3.5 dijelaskan proses pembentukan *frequent itemset*, menggunakan algoritma *apriori*. Parameter *apriori* yang sudah diinputkan kemudian diproses dan dilanjutkan ke pencarian kandidat *itemset* dari *dataset* transaksi. Pada kandidat *itemset* pertama semua *itemset* tunggal pada *dataset* transaksi dapat menjadi kandidat *itemset*, kemudian kandidat *itemset* yang memenuhi minimum *support* akan menjadi *frequent itemset* pertama.

Dari *frequent itemset* pertama yang ditemukan, kemudian sistem melakukan penggabungan *frequent itemset* untuk mendapatkan kandidat *itemset* selanjutnya. Setelah digabung, akan dihitung nilai *support* dari masing-masing kandidat *itemset* tadi, pada *dataset* transaksi untuk mencari berapa banyak kandidat *itemset* tersebut ditemukan pada tiap transaksi. Kemudian akan dilakukan *pruning* dengan cara melakukan pengecekan kandidat *itemset* yang

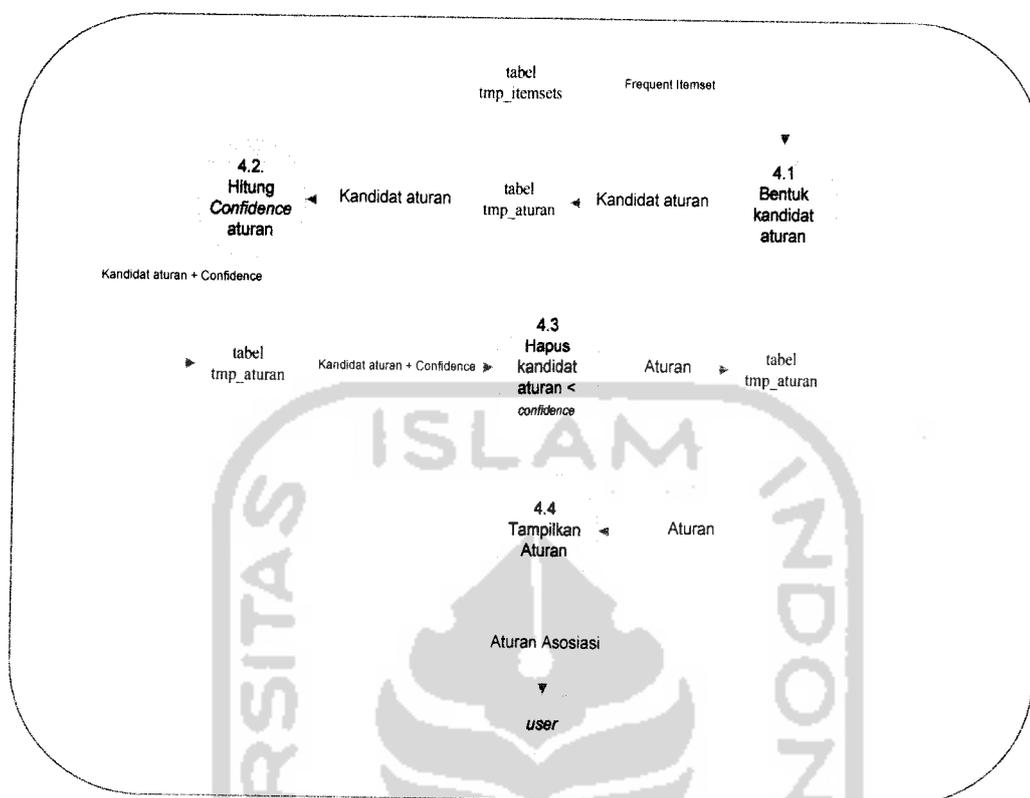
diperoleh dari hasil penggabungan, dengan *frequent itemset* sebelumnya. Apabila terdapat *itemset* yang tidak *frequent* (*itemset* pada kandidat tidak ditemukan pada *frequent itemset* sebelumnya) maka kandidat *itemset* tadi tidak akan dibentuk menjadi kandidat.

Proses diteruskan pada hapus *itemset*, pada proses ini akan dihilangkan kandidat *itemset* yang *supportnya* kurang dari minimum *support*.

Itemset yang diperoleh setelah proses hapus *itemset* di atas dilakukan, adalah *frequent itemset*. Dari *frequent itemset* tersebut akan kembali dibentuk kandidat *itemset* dengan anggota n item dan seterusnya, dengan melakukan proses yang sama pada penjelasan di atas, hingga diperoleh *frequent itemset* yang merupakan *frequent itemset* terakhir.

3.2.3.6 DFD Level 2 proses 4 (Proses Pembentukan Aturan Asosiasi)

Pembentukan aturan asosiasi dijelaskan pada DFD level 2 proses 4 yang ditunjukkan pada Gambar 3.6. *Frequent itemsets* yang terakhir diproses untuk pembentukan aturan (*rule*). Proses pertama adalah pembentukan kandidat aturan berdasarkan *frequent itemsets* terakhir. Dari kandidat aturan tersebut kemudian dihitung nilai *confidence* masing-masing. Setelah diperoleh nilai *confidence* dari kandidat aturan tersebut, dilakukanlah proses menghilangkan kandidat aturan yang tidak memenuhi nilai *confidence* yang telah diinputkan oleh *user*. Hasil dari proses ini akan menjadi aturan asosiasi yang memenuhi kriteria minimum *confidence*. Aturan asosiasi yang memenuhi minimum *confidence* kemudian ditampilkan kepada *user*.



Gambar 3.6 DFD Level 2 proses 4 (Proses Pembentukan *Association Rule*)

3.2.4 Perancangan Basis Data

Basis data berfungsi sebagai tempat penyimpanan data sebelum diolah. Di dalam menyimpan data, *database* memerlukan tabel sebagai tempat penyimpanan. *Database* ini terdiri dari 4 tabel, yaitu tabel transaksi, tabel *item_transaksi*, tabel barang, tabel member.

1. Tabel transaksi

Digunakan untuk mencatat transaksi yang dilakukan member.

Tabel 3.1 Struktur tabel transaksi

No.	Nama Field	Tipe Data	Keterangan
1	idtrans	Int (11)	kode dari transaksi yang dilakukan oleh member, dan sebagai primary key tabel.
2	tanggal	Date	Tanggal transaksi dilakukan
3	idmember	Varchar(11)	Kode member

2. Tabel transaksi_detail

Digunakan untuk menyimpan detail item transaksi.

Tabel 3.2 Struktur tabel transaksi_detail

No.	Nama Field	Tipe Data	Keterangan
1.	Idtrans	Int(11)	Kode transaksi
2.	idbrg	Varchar(11)	Kode dari barang yang dibeli
3.	unit	Int(11)	Jumlah barang yang dibeli

3. Tabel barang

Digunakan untuk menyimpan data barang yang dimiliki.

Tabel 3.3 Struktur tabel barang

No.	Nama Field	Tipe Data	Keterangan
1	idbrg	Varchar(11)	Kode dari barang yang dimiliki, dan sebagai primary key tabel.
2	Namabrg	Varchar(20)	Nama dari barang
3	Hargabrg	Decimal(20,2)	Harga dari barang

4. Tabel member

Digunakan untuk menyimpan data *member*

Tabel 3.5 Struktur tabel member

No.	Nama Field	Tipe Data	Keterangan
1.	Idmember	Varchar(11)	Kode anggota yang terdaftar, dan sebagai primary key tabel.
2.	Nama	Varchar(20)	Nama anggota
3.	Sex	Char(1)	Jenis
4.	Usia	Int(2)	Usia anggota
5.	Kerja	Varchar(20)	Jenis pekerjaan anggota

5. Tabel tmp_itemsets

Digunakan untuk menyimpan data yang diproses menjadi *frequent itemsets*.

Tabel 3.6 Struktur tabel tmp_itemsets

No.	Nama Field	Tipe Data	Keterangan
1.	kandidat	int(11)	Nama kandidat itemsets.
2.	items	Varchar(255)	Items yang ditemukan.
3.	support	int(11)	Besarnya support dari items
4.	valid	char(1)	Berisi Y jika frequent dan memenuhi batasan support, N jika tidak.
5.	hasil	Varchar(255)	Items yang akan di gabungkan
6.	Hasil2	Varchar(255)	Items yang akan di gabungkan

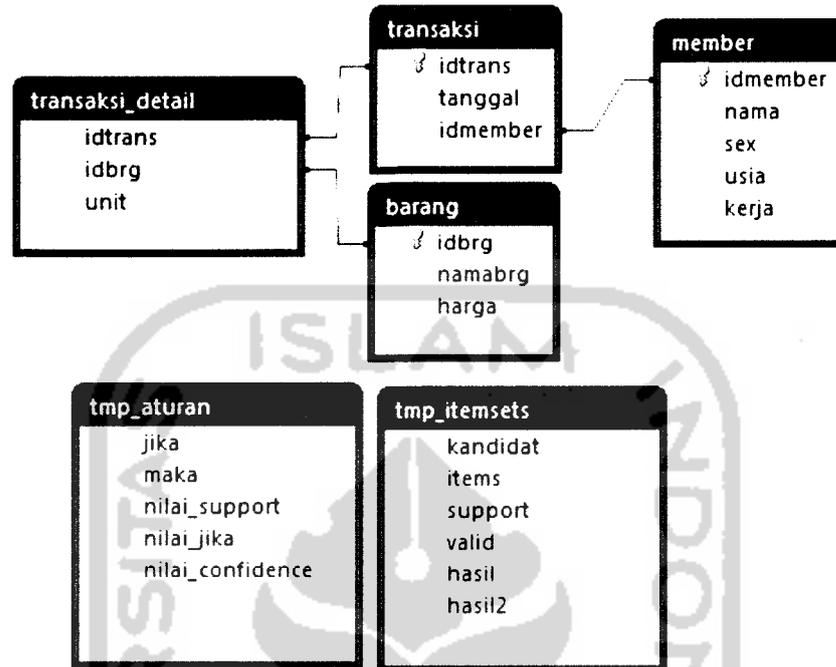
6. Tabel tmp_aturan

Digunakan untuk menyimpan data yang diproses menjadi aturan.

Tabel 3.7 Struktur tabel tmp_aturan

No.	Nama Field	Tipe Data	Keterangan
1.	jika	varchar(255)	Nilai dari <i>antecedent</i>
2.	maka	Varchar(255)	Nilai dari <i>consequent</i>
3.	Nilai_support	decimal (15)	Nilai support dari <i>frequent itemset</i> yang akan di bentuk aturannya
4.	Nilai_jika	decimal (15)	Nilai support dari <i>antecedent</i>
5.	Nilai_confidence	decimal (15)	Nilai confidence dari aturan

3.2.5 Relasi Antar Tabel dalam Database



Gambar 3.7 Relasi Antar Tabel dalam Database

3.2.6 Perancangan Antarmuka

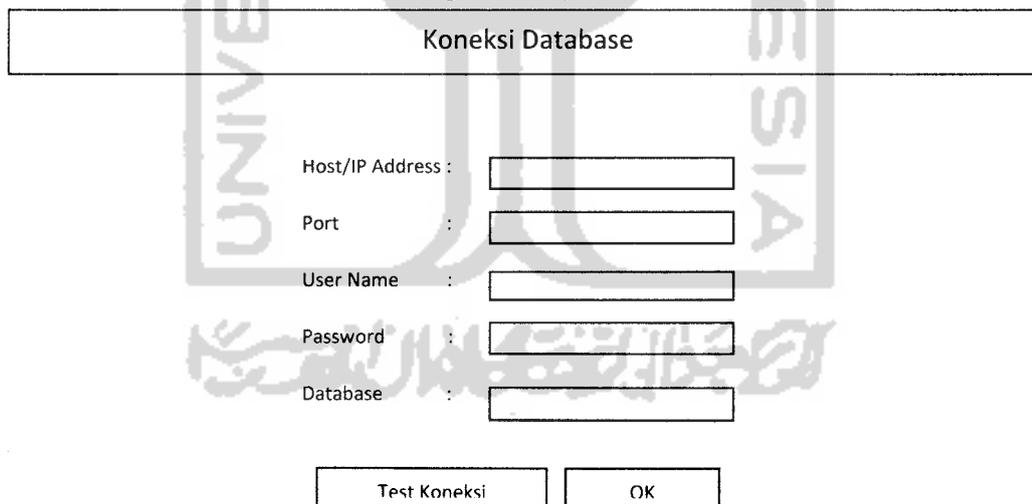
Rancangan antarmuka dari perangkat lunak ini menggunakan perancangan model grafis (visual). Terdapat sebuah form utama, panel menu dan *page control* yang berisi *tabsheet* koneksi, *tabsheet dataset*, *tabsheet* proses asosiasi.

3.2.6.1 Rancangan Antarmuka Koneksi Database

Pada Antarmuka Koneksi Database ditampilkan *tabsheet* koneksi yang digunakan untuk input koneksi ke *database server*. Parameter-parameter input koneksi adalah input *hostname*, *port*, *username*, *password* dan *database*.

Assosiation Rule Antar Komoditi pada Pusat Perbelanjaan dengan Menggunakan

Algoritma Apriori



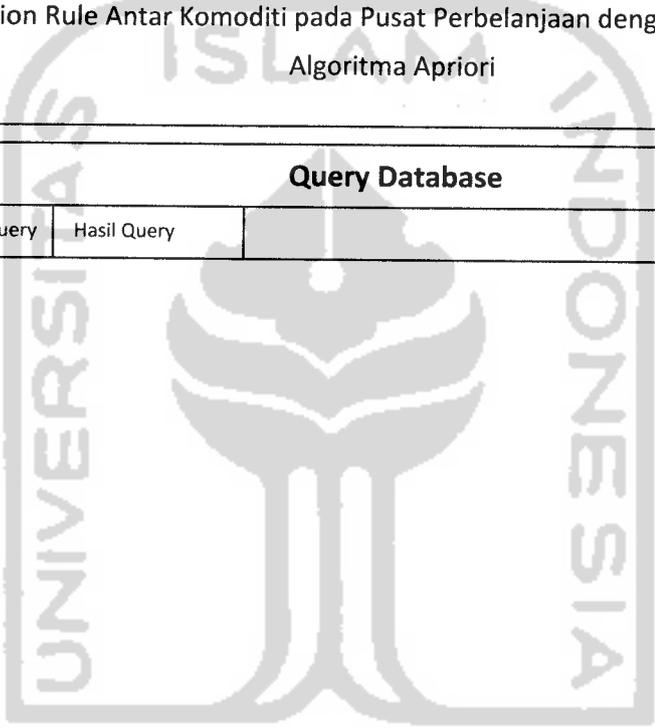
The image shows a dialog box titled "Koneksi Database" with a title bar. Below the title bar, there are five input fields with labels: "Host/IP Address", "Port", "User Name", "Password", and "Database". At the bottom of the dialog box, there are two buttons: "Test Koneksi" and "OK".

Gambar 3.8 Antarmuka Proses Koneksi

3.2.6.2 Rancangan Antarmuka Pengambilan Data dari Database

Pada *tabsheet dataset* seperti yang ditampilkan pada gambar 3.9, terdapat dua *tabsheet* yaitu *tabsheet editor query* dan *tabsheet hasil query*. Berfungsi untuk mengambil data dari dari *database* dengan menggunakan *query* SQL dan ditampilkan pada *dbgrid* di *tabsheet* hasil *query*.

Assosiation Rule Antar Komoditi pada Pusat Perbelanjaan dengan Menggunakan Algoritma Apriori

Query Database	
Editor Query	Hasil Query
	
<input type="button" value="Buka Script SQL"/> <input type="button" value="Simpan Script SQL"/> <input type="button" value="Hapus Tesxt"/> <input type="button" value="Jalankan"/>	

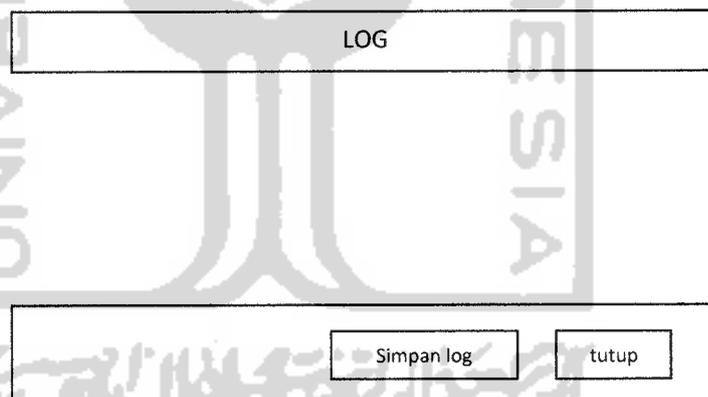
Gambar 3.9 *Tabsheet Dataset*

3.2.6.3 Rancangan Antarmuka Proses *Association Rule*

Pada antarmuka proses asosiasi, proses utama dari *association rule* dilakukan. Dengan memberikan input parameter-parameter yang berupa batas bawah *support*, minimum *confidence* dan jumlah aturan.

Antarmuka proses asosiasi memiliki dua *tabsheet* dan satu form yaitu :

1. *Tabsheet frequent itemset*, menampilkan *frequent itemset* yang ditemukan.
2. *Tabsheet* aturan asosiasi, menampilkan aturan-aturan yang berhasil ditemukan.
3. *Form Log*, menampilkan proses yang dilakukan dalam pencarian aturan asosiasi.



The image shows a screenshot of a software interface. At the top, there is a large rectangular box with the word "LOG" centered inside it. Below this box, there are two smaller rectangular buttons. The left button is labeled "Simpan log" and the right button is labeled "tutup". The background of the interface is light gray and features a large, faint watermark of the logo of Universitas Islam Indonesia, which includes a stylized tree and the text "UNIVERSITAS ISLAM INDONESIA".

Gambar 310 *form log*

Association Rule Antar Komoditi pada Pusat Perbelanjaan dengan Menggunakan
Algoritma Apriori

Frequent Itemset	Aturan Asosiasi	Log

Parameter dan Proses

Batas bawah suport :

Minimum confidence :

Jumlah aturan :

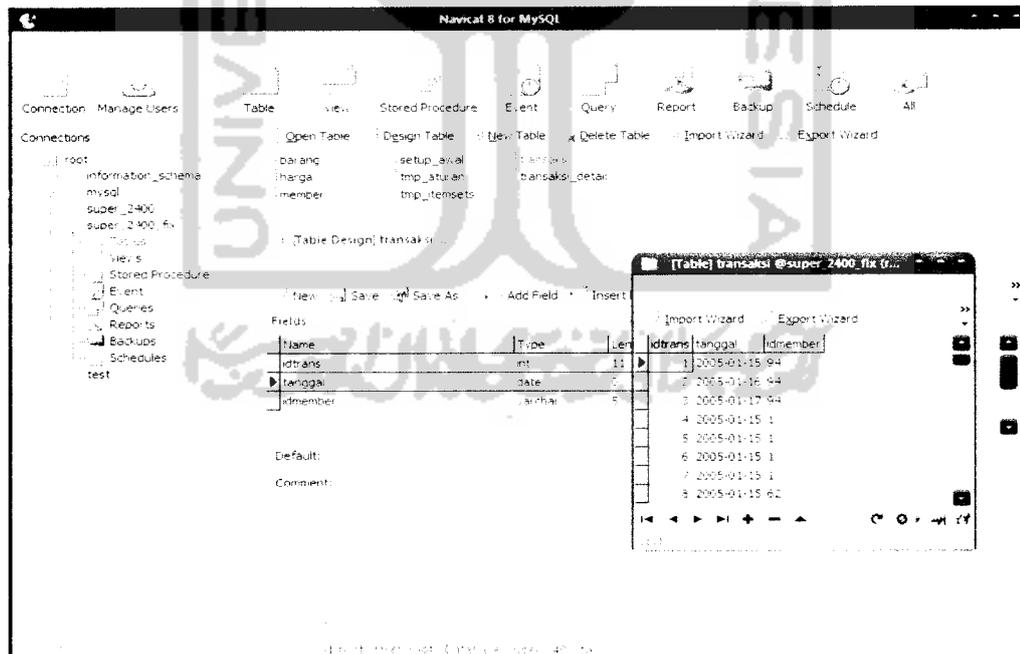
Gambar 3.11 *Tabsheet* Proses Asosiasi

3.3.1 Implementasi Sistem

Implementasi sistem merupakan tahap dimana sistem mampu diaplikasikan dalam keadaan yang sesungguhnya. Dari implementasi ini akan diketahui apakah sistem yang dibuat dapat berjalan dengan baik atau tidak dan menghasilkan output yang sesuai dengan perancangan yang ada.

3.3.2 Implementasi Basis Data

Proses pembangunan basis data menggunakan tools Navicat 8 for MySQL. Proses dimulai dengan membuat *database* baru. Kemudian dilanjutkan dengan membuat tabel-tabel sebagai sarana penyimpanan data dalam aplikasi. Berikut tampilan proses pembuatan *database* dan tabel.

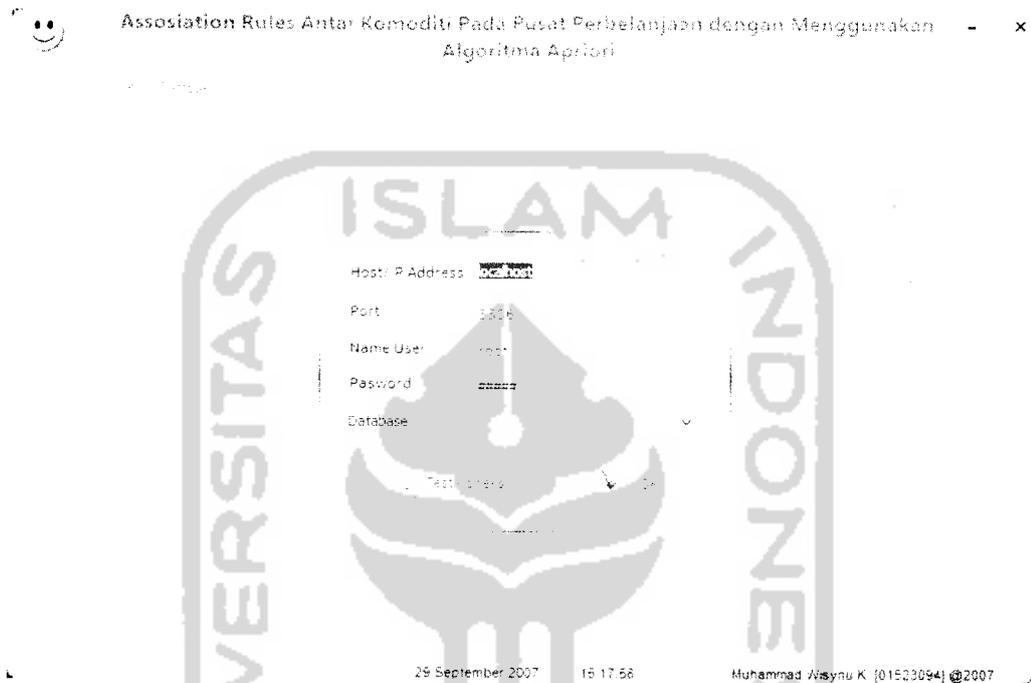


Gambar 3.12 Pembuatan Basis Data menggunakan Navicat 8 for MySQL

3.3.3 Komponen pengembangan antarmuka

3.3.3.1 Antarmuka Koneksi *Database*

Antarmuka koneksi *database* digunakan untuk melakukan proses koneksi ke *database* dengan memasukkan input berupa parameter koneksi.



Gambar 3.13 Antarmuka Koneksi *Database*

Prosedur untuk melakukan koneksi ke *database* ditunjukkan pada gambar 3.14.

```

procedure prep_connect; //isi properti MyConect
begin
  DM.MyConect.Server := frmmain.ed_konek_host.Text;
  DM.MyConect.Port := StrToInt(frmmain.ed_konek_port.Text);
  DM.MyConect.Username := frmmain.ed_konek_user.Text;
  DM.MyConect.Password := frmmain.ed_konek_pwd.Text;
  DM.MyConect.Database := frmmain.ed_konek_db.Text;
end;
...
...
DM.MyConect.Disconnect;
prep_connect;
try

```

```

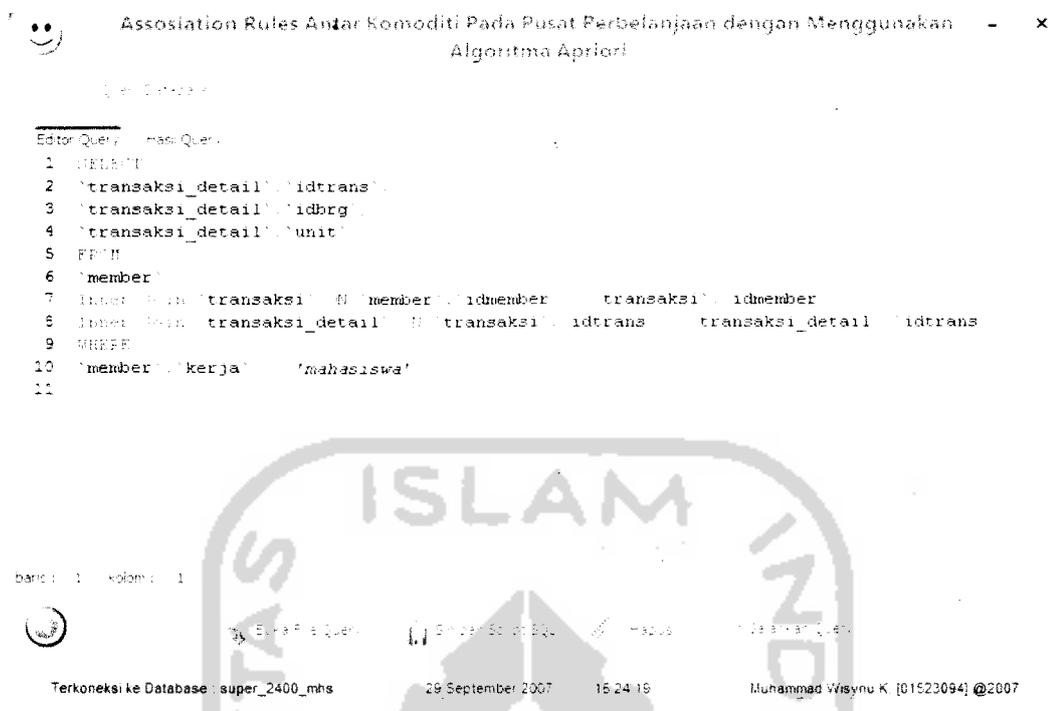
DM.MyConect.Connect; //lakukan koneksi ke database
...
...
except
on E: EMyError do begin
    MessageDlg('Koneksi Ke Database Gagal!!' + #13 + E.Message,
mtError, [mbOk], 0);
...
...

```

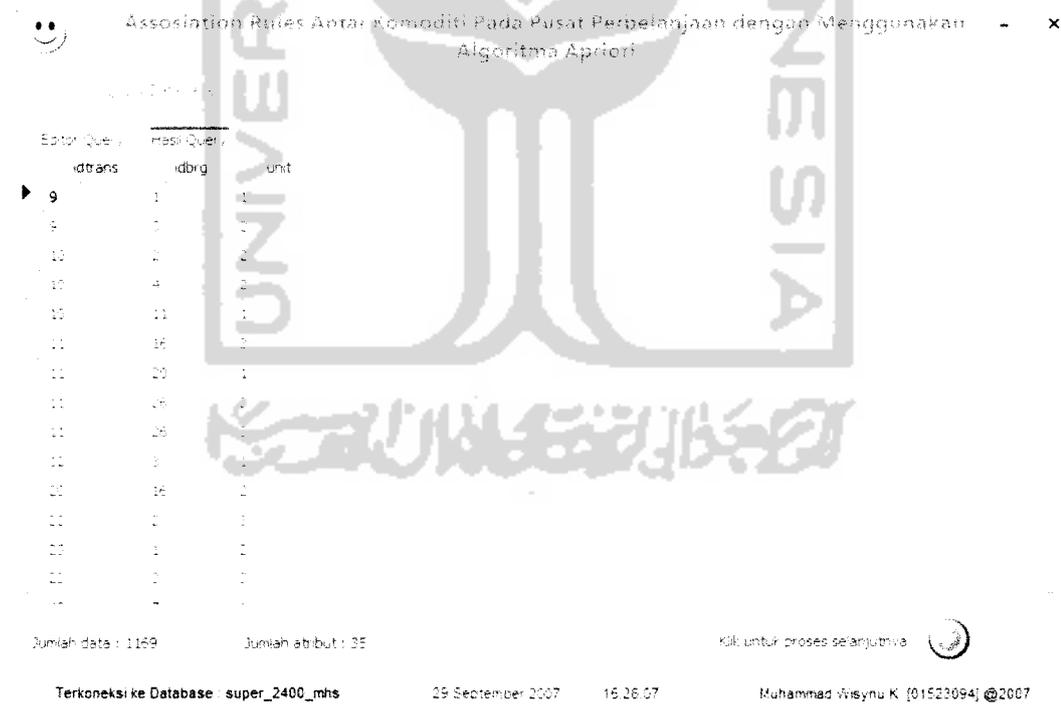
Gambar 3.14 Prosedur untuk melakukan koneksi ke *database*

3.3.3.2 Antarmuka *Query Database*

Antarmuka ini digunakan untuk melakukan proses input *query* dengan memasukkan atau menuliskan *query* menggunakan bahasa *SQL* pada *tabsheet* editor *query* (seperti ditampilkan pada gambar 3.15) kemudian dengan klik tombol “Jalankan Query”, akan ditampilkan *tabsheet* hasil *query* (gambar 3.16) yang berisi tampilan *dataset* transaksi, jumlah data dan jumlah atribut.



Gambar 3.15 Antarmuka Query Database *tabsheet* Editor Query



Gambar 3.16 Antarmuka Query Database *tabsheet* Hasil Query

```

procedure Tfrmmain.btn_gry_runClick(Sender: TObject);
var len: Integer;
begin
    DM.Script_query.SQL.Clear;

    try
        DM.Script_query.SQL.Text := mmo_gry.Lines.Text;
        DM.Script_query.Connection := DM.MyConect;
        DM.Script_query.Execute;

        DM.gry_query.SQL.Clear;
        DM.gry_query.SQL.Add('select count(*) from(select idbrg
from (' + mmo_gry.Lines.Text + ')as xx ' +
        'group by idbrg)as yy');
        DM.gry_query.Open;
        DBGrid_obat.DataSource := DM.Ds_gry;
        tab_gryresult.Show;

        lb_jmldata.Caption := 'Jumlah data : ' +
        IntToStr(DM.Script_query.recordcount);

        lb_jmlatribut.Caption := 'Jumlah atribut : ' +
        IntToStr(DM.gry_query.FieldByName('count(*)').AsInteger);
        DM.gry_query.SQL.Clear;

    except
        on E: EMyError do begin
            MessageDlg(E.Message, mtError, [mbOk], 0);
        end;
    end;
end;

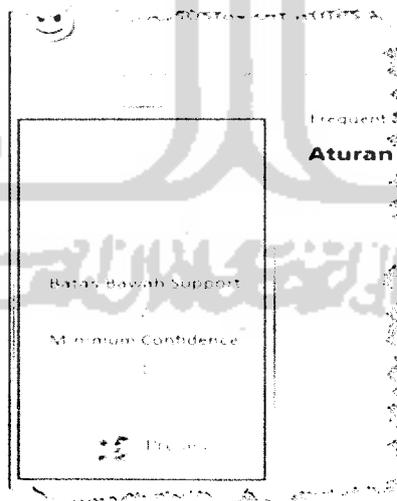
```

Gambar 3.17 Prosedure untuk menjalankan *query* dan menampilkan jumlah data dan jumlah atribut

3.3.3.3 Antarmuka Proses *Assosiation Rule*

Tampilan antarmuka proses *assosiation rule* terbagi menjadi 4 yaitu :

1. Panel parameter *apriori*. Panel ini digunakan untuk memasukkan input parameter *apriori* (gambar 3.18).
2. *Tabsheet frequent itemset*. Menampilkan *frequent itemset* yang diperoleh. Ditampilkan setelah proses pencarian *assosiation rule* selesai (gambar 3.19).
3. *Tabsheet aturan asosiasi*. Menampilkan aturan-aturan yang diperoleh. Ditampilkan setelah proses pencarian *assosiation rule* selesai (gambar 3.20).
4. *Form log*. Menampilkan proses yang terjadi saat proses *assosiation rule* berlangsung (gambar 3.21).



Gambar 3.18 Panel parameter *apriori*

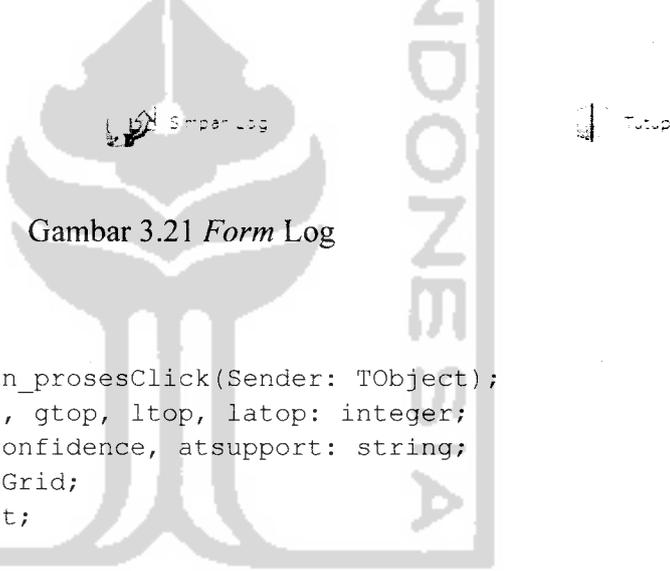
LOG

- x

```

=====
Proses pruning!
=====
Tidak ada item yang di pruning pada pembentukan kandidat itemset 2
=====
Frequent itemset 2 telah terbentuk
=====
Pembentukan kandidat itemset 3
=====
Proses pruning!
=====
items 123 tidak frequent karena items 12 tidak ada pada frequent itemset sebelumnya!
items 135 tidak frequent karena items 15 tidak ada pada frequent itemset sebelumnya!
=====
Frequent itemset 3 telah terbentuk
=====
=====

```



Tutup

Gambar 3.21 *Form Log*

```

procedure Tfrmmain.btn_prosesClick(Sender: TObject);
var i, count, z, c, l, gtop, ltop, latop: integer;
    atjika, atmaka, atconfidence, atsupport: string;
    strgrid: TAdvStringGrid;
    list: TAdvStringList;
    lbl: TLabel;
    startDT, endDT: TDateTime;
    mm: Extended;
begin
    if ed_bbsupport.Text = '' then
        begin
            MessageDlg('Masukan Batas Bawah Support', mtInformation,
            [mbOk], 1);
            ed_bbsupport.SetFocus;
            exit;
        end;
    if ed_basupport.Text = '' then
        begin
            MessageDlg('Masukan Batas Atas Support', mtInformation,
            [mbOk], 1);
            ed_basupport.SetFocus;

```

```

        exit;
    end;
    if ed_minconf.Text = '' then
    begin
        MessageDlg('Masukan Minimum Confidence', mtInformation,
[mbOk], 1);
        ed_minconf.SetFocus;
        exit;
    end;

    DM.SqlExec('DROP TABLE IF EXISTS temp_transaksi_detail');
    DM.SqlExec('CREATE TABLE `temp_transaksi_detail` (`idtrans`
int(11) NOT NULL, ' +
        '`idbrg` varchar(11) NOT NULL, `unit` varchar(5));');
    DM.SqlExec('insert into
temp_transaksi_detail(idtrans,idbrg,unit) ' +
        '(' + mmo_qry.Lines.Text + ')');
    frmmain.Hide;
    frmlog.Show;
    Application.ProcessMessages;
    frmlog.mlog.Clear;
    startDT := Now;
    frmlog.mlog.Lines.Add('mulai : ' + (DateTimetoStr(startDT)));
    Application.ProcessMessages;
    i := 1;
    frmlog.AdvWiiProgressBar1.Visible := True;
    Application.ProcessMessages;
    frmlog.lbl_wait.Visible := True;
    Application.ProcessMessages;
    frmlog.lbl_wait.Scrolling := True;
    Application.ProcessMessages;
    //ambil variable awal dari db
    ambil_var_awal;
        Application.ProcessMessages;
    //pengelompokan barang
    pembentukan_kandidat_awal(i);
        Application.ProcessMessages;
    //mencari kandidat yang memenuhi persyaratan
    cek_valid(i);
        Application.ProcessMessages;
    repeat
    begin
        i := i + 1;
        //memasukkan kandidat item ke 2 sd n
        pembentukan_kandidat_ke_n(i);
        Application.ProcessMessages;
        with dm.qry3 do

```

```

begin
    Close;
    sql.Clear;
    SQL.Add('select count(*) count from tmp_itemsets where
valid='Y' and ' +
        'kandidat=' + QuotedStr(IntToStr(i)));
    Open;
end;
count := DM.qry3.FieldByName('count').AsInteger;
//proses pruning
prunning(i);
Application.ProcessMessages;
    //cek valid
cek_valid(i);

end;

until count < 1;

buat_aturan(i);
frmlog.mlog.Lines.Add('');

frmlog.mlog.Lines.Add('=====');
frmlog.mlog.Lines.Add('Aturan yang ditemukan');
frmlog.mlog.Lines.Add('');

latop := 0;
l := 1;

for c := sb_aturan.ControlCount - 1 downto 0 do
begin
    sb_aturan.Controls[c].Destroy;
end;
while not DM.qry4.Eof do
begin
    atjika := DM.qry4.FieldByName('str_jika').AsString;
    atmaka := DM.qry4.FieldByName('str_maka').AsString;
    atsupport := DM.qry4.FieldByName('nilai_support').AsString;
    atconfidence :=
DM.qry4.FieldByName('nilai_confidence').AsString;

    frmlog.mlog.Lines.Add('Jika ' + atjika + ' ==> ' + atmaka +
' [support : ' + atsupport + ' confidence : ' + atconfidence +
' ]');
    frmlog.mlog.Lines.Add('');

    lbl := TLabel.Create(sb_aturan);
    lbl.Parent := sb_aturan;
    lbl.Left := 30;

```

```

    lbl.Top := latop;
    lbl.Font.Color := clBlack;
    lbl.Font.Name := 'calibri';
    lbl.Font.Size := 9;
    lbl.Transparent := True;
    lbl.Caption := ' ' + IntToStr(l) + '. jika membeli ' + atjika
+ ' maka mungkin membeli juga ' +
    ' ' + atmaka + ' [support : ' + atsupport + ' , ' +
    'confidence : ' + atconfidence + ']';
    latop := latop + 30;
    l := l + 1;
    dm.qry4.Next;
end;

frmlog.mlog.Lines.Add('-----');
endDT := Now;
frmlog.mlog.Lines.Add('selesai : ' + DateTimeToStr(endDT));
mm := ((endDT - StartDT) * 24 * 60);
frmlog.mlog.Lines.Add('Lama waktu proses : ' +
Float2EStr(mm) + ' menit');

DM.SqlExec('DROP TABLE IF EXISTS temp_transaksi_detail');
//////////
for c := sb_freq.ControlCount - 1 downto 0 do
begin
    sb_freq.Controls[c].Destroy;
end;

gtop := 35;
ltop := 18;

for z := 1 to (i - 1) do
begin
    lbl := TLabel.Create(sb_freq);
    lbl.Parent := sb_freq;
    lbl.Left := 217;
    lbl.Top := ltop;
    lbl.Font.Color := clBlack;
    lbl.Transparent := True;
    lbl.Caption := 'Frequent Itemset : ' + IntToStr(z);
    strgrid := TAdvStringGrid.Create(sb_freq);
    strgrid.Parent := sb_freq;
    strgrid.Height := 156;
    strgrid.Width := 264;
    strgrid.Left := 164;
    strgrid.Top := gtop;
    strgrid.ColCount := 3;

```

```

list := TAdvStringList.Create;
list.Add('');
list.Add('Items');
list.Add('Support');
strgrid.ColumnHeaders := list;
strgrid.FixedColWidth := 2;
strgrid.ColWidths[1] := 177;
with dm.qrySelect1 do
begin
  Close;
  SQL.Clear;
  SQL.Add('select items,support from tmp_itemsets WHERE
kandidat=' + QuotedStr(IntToStr(z)) + ' and valid='Y'');
  Open;
  First;
  l := 1;
  while not Eof do
  begin
    atjika := DM.qrySelect1.FieldByName('items').AsString;
    atmaka := DM.qrySelect1.FieldByName('support').AsString;

    strgrid.Cells[1, l] := atjika;
    strgrid.Cells[2, l] := atmaka;
    l := l + 1;
    strgrid.RowCount := RecordCount + 1;
  Next;
  end;
end;

gtop := gtop + strgrid.Height + 30;
ltop := ltop + strgrid.Height + 30;
end;

frmlog.AdvWiiProgressBar1.Visible := False;
frmlog.lbl_wait.Visible := False;
frmlog.lbl_wait.Scrolling := False;
frmlog.btn_log_savelog.Enabled := True;
frmlog.btn_log_close.Enabled := True;
end;

```

Gambar 3.22 prosedur untuk proses *association rule*, tampil *form log*, tampil *frequent itemset* pada *tabsheet frequent itemset* dan tampil aturan pada *tabsheet aturan asosiasi*

BAB IV

HASIL DAN PEMBAHASAN

Sebelum program diterapkan terlebih dahulu program harus bebas dari kesalahan. Untuk itu program harus diuji untuk menemukan kesalahan-kesalahan yang mungkin dapat terjadi.

Kesalahan yang mungkin terjadi dapat dikualifikasikan dalam tiga bentuk kesalahan, yakni sebagai berikut :

1. Kesalahan bahasa (*language errors*) atau disebut juga dengan kesalahan penulisan (*syntax error*) adalah kesalahan di dalam penulisan *source* program yang tidak sesuai dengan yang telah diisyaratkan.
2. Kesalahan sewaktu proses (*run-time errors*) adalah kesalahan yang terjadi sewaktu program dieksekusi.
3. Kesalahan logika (*logical errors*) adalah kesalahan dari logika program yang dibuat. Kesalahan seperti ini sulit ditemukan karena tidak ada pemberitahuan mengenai kesalahan, dan akan tetap diperoleh hasil dari proses program, tetapi hasilnya salah.

4.1. Pengujian Sistem

Pengujian sistem dilakukan untuk memeriksa sistem yang telah diimplementasikan. Tujuan utama dari pengujian ini adalah untuk memastikan bahwa elemen-elemen atau komponen-komponen dari sistem telah berfungsi sesuai dengan yang diharapkan. Pengujian perlu dilakukan untuk mencari kesalahan-kesalahan atau kelemahan-kelemahan yang mungkin masih terjadi. Pengujian sistem termasuk juga analisis program secara menyeluruh untuk melihat apakah suatu program dapat berjalan dengan baik dan dapat memproses data dengan baik.

Pengujian yang dilakukan pada sistem ini dilakukan dengan dua cara yaitu pengujian normal dan tidak normal. Maksud dari pengujian normal adalah memberikan input yang sesuai dengan isian yang dibutuhkan untuk membuktikan bahwa proses berjalan dengan benar. Sedangkan pengujian secara tidak normal yaitu memberikan masukan yang salah, sehingga diketahui apakah sistem dapat menangani kesalahan yang ada dan memberitahukannya kepada *user*.

4.1.1. Pengujian Normal

Pengujian ini berfungsi untuk melihat respon yang dikeluarkan program apabila sistem ini berjalan sesuai dengan aturan.

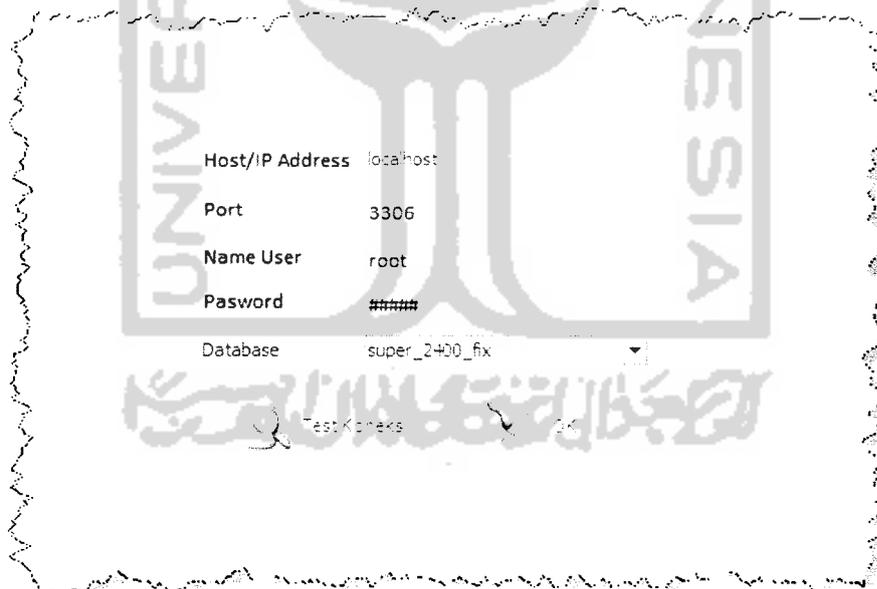
Pada pengujian aplikasi ini, akan digunakan beberapa *query* yang berbeda dalam pengkategorian data transaksi yang ada. *Database* yang akan dicari aturan asosiasinya adalah *database* dengan nama “super_2400_fix”, dengan struktur basis data seperti yang telah dijelaskan pada perancangan basis data,

dengan detail data seperti pada tabel 4.1. Sedangkan server yang digunakan berada pada *localhost* dengan menggunakan port “3306”, username “root” dan password “12345”.

Tabel 4.1 Detail data *database* super_2400_fixTabel

tabel	Banyak data
barang	35
harga	35
member	200
transaksi	2400
transaksi_detail	4038

Pada proses koneksi *database* dimasukkan input parameter koneksi seperti pada gambar 4.1.



Host/IP Address localhost
Port 3306
Name User root
Password #####
Database super_2400_fix
Test Koneksi

Gambar 4.1 Input Parameter Koneksi

4.1.1.1. Pengujian menggunakan data pada Tabel 2.1

Pada pengujian ini akan dicari aturan-aturan dari data yang ada pada ilustrasi algoritma apriori, data tersebut dapat dilihat pada Tabel 2.1, dengan jumlah transaksi sebanyak 4. Dan menggunakan *script query* seperti pada Gambar 4.4. Pengujian ini menggunakan *database* yang berbeda dengan yang digunakan pada pengujian lainnya, yaitu *database* “apriori_sample”. Parameter apriori yang di inputkan, yaitu :

1. Batas bawah *support* : 2.
2. Minimum *confidence* : 1.

Setelah *query* dijalankan maka akan diperoleh jumlah data sebanyak 12 dan jumlah atribut sebanyak 5. *Frequent itemset* yang diperoleh, dapat dilihat pada Gambar 4.2, yaitu *frequent itemset* 1,2 dan 3.

Aturan asosiasi yang diperoleh sebanyak 2 (gambar 4.3). Lama waktu proses pencarian aturan asosiasi adalah 0,0258 menit.

Frequent Itemset : 1	
Items	Support
1	2
2	3
3	3
5	3

Frequent Itemset : 2	
Items	Support
1,3	2
2,3	3
2,5	3
3,5	3

Frequent Itemset : 3	
Items	Support
2,3,5	2

Gambar 4.2 *Frequent itemset* yang diperoleh dari data pada Tabel 2.1

Jika membeli Tissue Paseo,Pepsodent 100gr maka membeli juga Pampers Bayi [support : 2
confidence : 1]

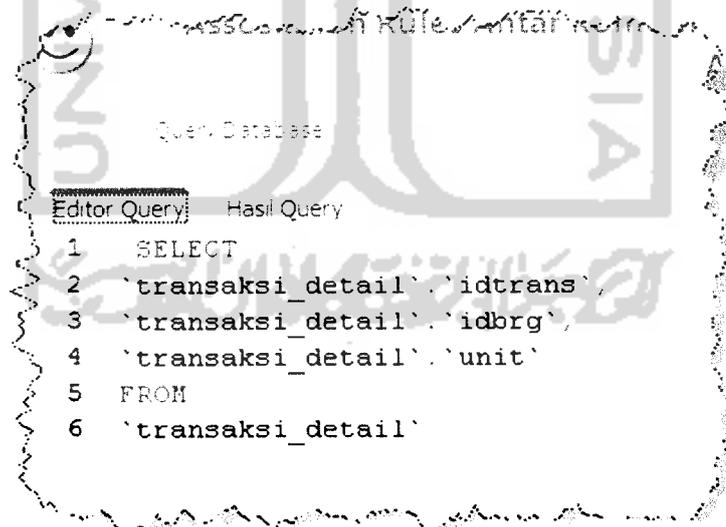
Jika membeli Pepsodent 100gr,Pampers Bayi maka membeli juga Tissue Paseo [support : 2
confidence : 1]

Gambar 4.3 Aturan asosiasi yang diperoleh dari data pada Tabel 2.1

4.1.1.2. Pengujian menggunakan keseluruhan data transaksi

Pada pengujian ini akan dicari aturan-aturan dari keseluruhan data transaksi yang terdapat pada *database*, dengan menggunakan *script query* seperti pada Gambar 4.4. Parameter apriori yang di inputkan, yaitu :

1. Batas bawah *support* : 2.
2. Minimum *confidence* : 1.



Gambar 4.4 *Script Query* pengujian menggunakan keseluruhan data transaksi

Dari *query* di atas diperoleh jumlah data sebanyak 4038 dan jumlah atribut sebanyak 35. *Frequent itemset* yang diperoleh, dapat dilihat pada gambar 4.5, yaitu *frequent itemset* 1, 2, 3, 4, 5, 6.

Aturan asosiasi yang diperoleh sebanyak 145, beberapa aturan asosiasi dapat dilihat pada Gambar 4.6. Lama waktu proses pencarian aturan asosiasi 28,4661 menit.

Frequent Itemset : 1	
Items	Support
1	320
10	14
11	74
12	26
13	318
...	...
Frequent Itemset : 2	
Items	Support
1,11	20
1,12	2
1,13	16
1,15	2
1,16	36
...	...
Frequent Itemset : 3	
Items	Support
1,11,16	2
1,11,17	6
1,11,13	2
1,11,2	2
1,11,23	1
...	...
Frequent Itemset : 4	
Items	Support
1,11,16,17	2
1,11,16,3	2
1,11,17,3	4
1,11,17,31	2
1,11,17,33	2
...	...
Frequent Itemset : 5	
Items	Support
1,11,16,17,3	2
1,11,17,3,31	2
1,11,17,3,33	2
1,11,17,31,33	2
1,11,13,26,3	3
...	...
Frequent Itemset : 6	
Items	Support
1,11,17,3,31,33	2
1,16,13,26,33,34	2
13,14,16,20,25,23	2
13,16,2,25,27,4	2
16,13,2,26,31,5	2

Gambar 4.5 *frequent itemset* pengujian menggunakan keseluruhan data transaksi

.....

 Jika membeli Tissue Tessa, Bir Bintang Botol 750ml, Pepsodent 100gr, Wall's Vinetta 1000ml, Vaseline Complete Care 120ml maka membeli juga fanta kaleng 500ml [support : 2 confidence : 1]

Jika membeli fanta kaleng 500ml, Pepsodent 100gr, Wall's Vinetta 1000ml, Vaseline Complete Care 120ml maka membeli juga Tissue Tessa, Bir Bintang Botol 750ml [support : 2 confidence : 1]

Jika membeli Tissue Tessa, fanta kaleng 500ml, Pepsodent 100gr, Wall's Vinetta 1000ml, Vaseline Complete Care 120ml maka membeli juga Bir Bintang Botol 750ml [support : 2 confidence : 1]

.....

Gambar 4.6 Beberapa aturan asosiasi yang diperoleh dari pengujian menggunakan keseluruhan data transaksi.

4.1.1.3. Pengujian menggunakan *dataset* transaksi berdasarkan jenis pekerjaan “mahasiswa”

Pada pengujian ini, *dataset* transaksi yang digunakan berdasarkan kepada jenis pekerjaan yang terdapat pada tabel “member” yaitu “mahasiswa”. Untuk mendefinisikan bahwa data transaksi yang akan diproses hanya transaksi yang dilakukan oleh “member” dengan pekerjaan “mahasiswa”, digunakan *query* yang sedikit berbeda dari pengujian sebelumnya. *Script query* yang diinputkan ditunjukkan pada Gambar 4.7. parameter apriori yang di inputkan, yaitu :

1. Batas bawah *support* : 2.
2. Minimum *confidence* : 1.

```

1 SELECT
2   transaksi_detail.idtrans
3   transaksi_detail.idbrg
4   transaksi_detail.unit
5 FROM
6   member
7   inner join transaksi on member.idmember = transaksi.idmember
8   inner join transaksi_detail on transaksi.idtrans = transaksi_detail
9   idtrans
10 WHERE
11   member.kerja = 'mahasiswa'
12

```

Gambar 4.7 Script query pengujian menggunakan *dataset* transaksi berdasarkan jenis pekerjaan “mahasiswa”

Dari query di atas diperoleh jumlah data sebanyak 1169 dan jumlah atribut sebanyak 35. Frequent itemset yang diperoleh, dapat dilihat pada gambar 4.8, yaitu *frequent itemset* 1, 2, 3, 4, 5.

Aturan asosiasi yang diperoleh sebanyak 2 (gambar 4.9). Lama waktu proses pencarian aturan asosiasi adalah 1,3964 menit.

Frequent Itemset : 1	
Items	Support
1	103
10	5
11	19
12	7
13	84
--	--

Frequent Itemset : 2	
Items	Support
1,11	6
1,13	3
1,16	13
1,17	7
1,18	4
--	--

Frequent Itemset : 3	
Items	Support
1,11,28	2
1,11,3	4
1,13,25	2
1,16,18	3
1,16,20	4
--	--

Frequent Itemset : 4	
Items	Support
1,16,20,26	5
1,16,20,3	3
1,16,26,3	4
1,2,3,4	4
1,20,26,3	3
--	--

Frequent Itemset : 5	
Items	Support
1,16,20,26,3	3

Gambar 4.8 *frequent itemset* pengujian menggunakan *dataset* transaksi berdasarkan jenis pekerjaan “mahasiswa”

Jika membeli Tissue Tessa,Fanta 1lt,Attack 500gr,Pepsodent 100gr maka membeli juga Djarum Super 12 [support : 3 confidence : 1]

Jika membeli Tissue Tessa,Attack 500gr,Djarum Super 12,Pepsodent 100gr maka membeli juga Fanta 1lt [support : 3 confidence : 1]

Gambar 4.9 Aturan asosiasi yang diperoleh dari pengujian menggunakan *dataset* transaksi berdasarkan jenis pekerjaan “mahasiswa”

4.1.1.4. Pengujian menggunakan *dataset* transaksi berdasarkan jenis pekerjaan “karyawan”

Pada pengujian ini, *dataset* transaksi yang digunakan berdasarkan kepada jenis pekerjaan yang terdapat pada tabel “member” yaitu “karyawan”. *Script query* yang diinputkan ditunjukkan pada Gambar 4.10. Parameter apriori yang diinputkan, yaitu :

1. Batas bawah *support* : 2.
2. Minimum *confidence* : 1.

```

1 SELECT
2 transaksi_detail idtrans
3 transaksi_detail idbrg
4 transaksi_detail unit
5 FROM
6 member
7 JOIN transaksi ON member idmember = transaksi idmember
8 JOIN transaksi_detail ON transaksi idtrans = transaksi_detail idtrans
9
10
11 member kerja 'Karyawan'
12

```

Gambar 4.10 *script query* pengujian menggunakan *dataset* transaksi berdasarkan jenis pekerjaan “karyawan”

Dari *query* di atas diperoleh jumlah data sebanyak 908 dan jumlah atribut sebanyak 35. *Frequent itemset* yang diperoleh, dapat dilihat pada gambar 4.11, yaitu *frequent itemset* 1, 2, 3, 4, 5.

Aturan asosiasi yang diperoleh sebanyak 17 (gambar 4.12). Lama waktu proses pencarian aturan asosiasi adalah 1,6213 menit.

Frequent Itemset : 1	
Items	Support
1	61
10	4
11	9
12	6
13	66
	--
Frequent Itemset : 2	
Items	Support
1,13	4
1,16	7
1,17	2
1,2	3
1,20	2
	-
Frequent Itemset : 3	
Items	Support
1,13,27	2
1,13,4	2
1,16,3	3
1,17,5	2
1,2,7	2
	-
Frequent Itemset : 4	
Items	Support
13,15,25,27	2
13,16,25,27	2
15,21,25,27	4
13,23,25,27	2
14,16,20,22	2
	-
Frequent Itemset : 5	
Items	Support
14,16,20,22,25	2

Gambar 4.11 *frequent itemset* pengujian menggunakan *dataset* transaksi berdasarkan jenis pekerjaan “karyawan”

Jika membeli Gillette Biasa biru,Fanta 1lt maka membeli juga Attack 500gr,Lifebuoy Deepclean isi ulang 250ml,Gatsby Body Lotion Fresh 100ml [support : 2 confidence : 1]

Jika membeli Gillette Biasa biru,Attack 500gr maka membeli juga Fanta 1lt,Lifebuoy Deepclean isi ulang 250ml,Gatsby Body Lotion Fresh 100ml [support : 2 confidence : 1]

Jika membeli Gillette Biasa biru,Fanta 1lt,Attack 500gr maka membeli juga Lifebuoy Deepclean isi ulang 250ml,Gatsby Body Lotion Fresh 100ml [support : 2 confidence : 1]

Jika membeli Fanta 1lt,Lifebuoy Deepclean isi ulang 250ml maka membeli juga Gillette Biasa biru,Attack 500gr,Gatsby Body Lotion Fresh 100ml [support : 2 confidence : 1]

Jika membeli Gillette Biasa biru,Fanta 1lt,Lifebuoy Deepclean isi ulang 250ml maka membeli juga Attack 500gr,Gatsby Body Lotion Fresh 100ml [support : 2 confidence : 1]

Jika membeli Attack 500gr,Lifebuoy Deepclean isi ulang 250ml maka membeli juga Gillette Biasa biru,Fanta 1lt,Gatsby Body Lotion Fresh 100ml [support : 2 confidence : 1]

Jika membeli Gillette Biasa biru,Attack 500gr,Lifebuoy Deepclean isi ulang 250ml maka membeli juga Fanta 1lt,Gatsby Body Lotion Fresh 100ml [support : 2 confidence : 1]

Jika membeli Fanta 1lt,Attack 500gr,Lifebuoy Deepclean isi ulang 250ml maka membeli juga Gillette Biasa biru,Gatsby Body Lotion Fresh 100ml [support : 2 confidence : 1]

Jika membeli Gillette Biasa biru,Fanta 1lt,Attack 500gr,Lifebuoy Deepclean isi ulang 250ml maka membeli juga Gatsby Body Lotion Fresh 100ml [support : 2 confidence : 1]

Jika membeli Gillette Biasa biru,Fanta 1lt,Gatsby Body Lotion Fresh 100ml maka membeli juga Attack 500gr,Lifebuoy Deepclean isi ulang 250ml [support : 2 confidence : 1]

Jika membeli Gillette Biasa biru,Attack 500gr,Gatsby Body Lotion Fresh 100ml maka membeli juga Fanta 1lt,Lifebuoy Deepclean isi ulang 250ml [support : 2 confidence : 1]

Jika membeli Gillette Biasa biru,Fanta 1lt,Attack 500gr,Gatsby Body Lotion Fresh 100ml maka membeli juga Lifebuoy Deepclean isi ulang 250ml [support : 2 confidence : 1]

Jika membeli Fanta 1lt,Lifebuoy Deepclean isi ulang 250ml,Gatsby Body Lotion Fresh 100ml maka membeli juga Gillette Biasa biru,Attack 500gr [support : 2 confidence : 1]

Jika membeli Gillette Biasa biru,Fanta 1lt,Lifebuoy Deepclean isi ulang 250ml,Gatsby Body Lotion Fresh 100ml maka membeli juga Attack 500gr [support : 2 confidence : 1]

Jika membeli Attack 500gr,Lifebuoy Deepclean isi ulang 250ml,Gatsby Body Lotion Fresh 100ml maka membeli juga Gillette Biasa biru,Fanta 1lt [support : 2 confidence : 1]

Jika membeli Gillette Biasa biru,Attack 500gr,Lifebuoy Deepclean isi ulang 250ml,Gatsby Body Lotion Fresh 100ml maka membeli juga Fanta 1lt [support : 2 confidence : 1]

Jika membeli Fanta 1lt,Attack 500gr,Lifebuoy Deepclean isi ulang 250ml,Gatsby Body Lotion Fresh 100ml maka membeli juga Gillette Biasa biru [support : 2 confidence : 1]

Gambar 4.12 Aturan asosiasi yang diperoleh dari pengujian menggunakan *dataset* transaksi berdasarkan jenis pekerjaan “karyawan”

4.1.2. Pengujian Tidak Normal

Dalam tahap pengujian tidak normal ini, akan dilakukan suatu proses pengisian atau masukan yang salah dan diharapkan sistem dapat menangani kesalahan dan dapat menampilkan suatu pesan kesalahan.

4.1.2.1. Proses Koneksi Database

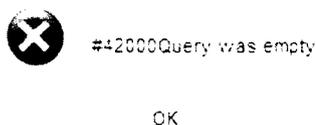
Pada proses ini jika tidak mengisi salah satu parameter yang dibutuhkan maka akan muncul pesan kesalahan contohnya pada gambar 4.13, pesan kesalahan akan muncul jika *user* tidak mengisi nama *database*.



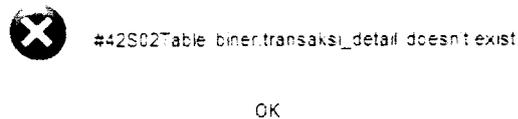
Gambar 4.13 Pesan kesalahan jika tidak mengisi nama *database*

4.1.2.2. Proses Input Query

Jika tidak ada *query* yang akan diproses maka akan muncul pesan kesalahan seperti pada gambar 4.14 dan juga jika tidak ada tabel yang dibutuhkan dalam *database* maka akan muncul pesan kesalahan seperti pada gambar 4.15.



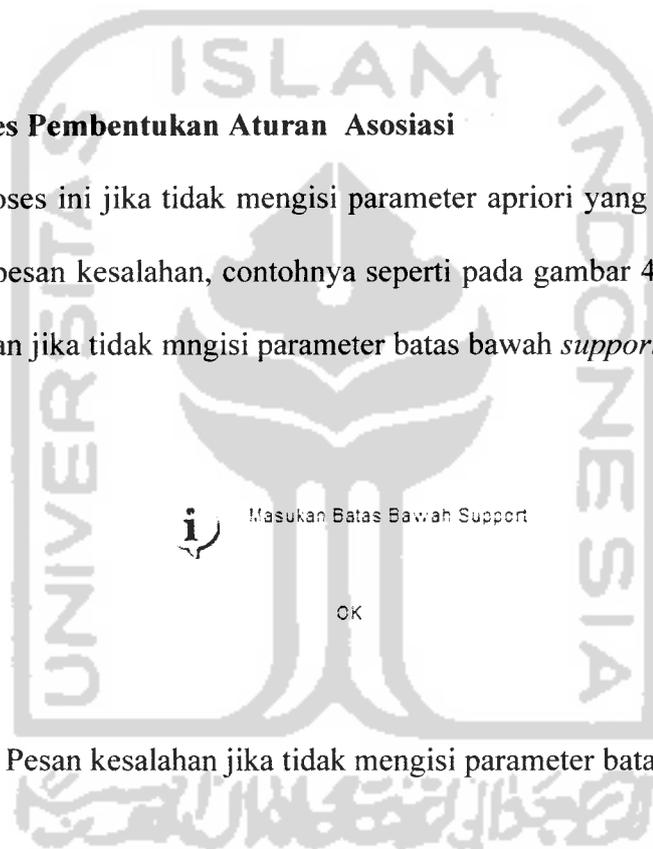
Gambar 4.14 Pesan kesalahan jika tidak ada *query* yang akan diproses



Gambar 4.15 Pesan kesalahan jika tabel yang dibutuhkan tidak ada dalam *database*

4.1.2.3. Proses Pembentukan Aturan Asosiasi

Pada proses ini jika tidak mengisi parameter apriori yang dibutuhkan maka akan muncul pesan kesalahan, contohnya seperti pada gambar 4.16, akan muncul pesan kesalahan jika tidak mengisi parameter batas bawah *support*.



Gambar 4.16 Pesan kesalahan jika tidak mengisi parameter batas bawah *support*

4.1.3. Analisis Hasil Pengujian

Dari hasil pengujian yang telah dilakukan dapat dilihat bahwa aplikasi yang telah dibuat telah berhasil mendapatkan aturan-aturan asosiasi dari *dataset* transaksi yang telah di inputkan (menggunakan *script query* pada proses input *query*). Salah satu aturan yang didapat adalah “Jika membeli Pepsodent 100gr,Pempers Bayi maka membeli juga Tisu Paseo [support : 2 confidence : 1]”. *support* : 2 artinya adalah dari keseluruhan transaksi yang berjumlah 4 transaksi, ada 2 transaksi yang di dalam nya terdapat item Pepsodent 100gr,Pempers Bayi dan Tisu Paseo. *confidence* : 1 artinya adalah nilai peluang item Tisu Paseo dibeli jika membeli item Pepsodent 100gr dan Pempers Bayi secara bersamaan adalah 1 atau bisa juga dikatakan dari dua transaksi yang didalamnya terdapat item Pepsodent 100gr dan Pempers Bayi(didapat dengan menghitung *support* { Pepsodent 100gr,Pempers Bayi }), terdapat juga item Tisu Paseo. Nilai *confidence* didapat dari :

$$\frac{\text{support} \{ \text{Pepsodent 100gr, Pempers Bayi, Tisu Paseo} \}}{\text{support} \{ \text{Pepsodent 100gr, Pempers Bayi} \}} = \frac{2}{2}$$

Untuk membuktikan bahwa aturan-aturan asosiasi yang dihasilkan oleh aplikasi ini adalah valid, maka dibandingkan pengujian menggunakan data pada Tabel 2.1 dengan ilustrasi algoritma apriori pada BAB II yang juga menggunakan data pada Tabel 2.1, dapat dilihat bahwa aturan yang ditemukan sama, begitu pula *frequent itemset* dan item-item yang di pruning juga sama, jumlah aturan-aturan yang diperoleh juga sesuai dengan ketentuan $2^k - 2$ (k adalah ukuran dari *itemset*) seperti yang telah dijelaskan pada BAB II, pada pengujian ini juga diperoleh

aturan sebanyak 6 dan karena syarat minimal *confidence* nya 1 maka aturan yang memenuhi syarat dan kemudian yang ditampilkan, sebanyak 2 aturan.

Aturan-aturan yang dihasilkan juga tidak bertentangan dengan *frequent itemset* tertinggi yang ada, seperti pada pengujian menggunakan *dataset* transaksi berdasarkan jenis pekerjaan “mahasiswa”, aturan yang memenuhi *minimal confidence* adalah :

1. Jika membeli Tissu Tessa,Fanta 1lt,Attack 500gr,Pepsodent 100gr maka membeli juga Djarum Super 12 [support : 3 confidence : 1]
2. Jika membeli Tissu Tessa,Attack 500gr,Djarum Super 12,Pepsodent 100gr maka membeli juga Fanta 1lt [support : 3 confidence : 1]

Dua aturan di atas disusun dari *frequent itemset* tertinggi yaitu *frequent itemset* 5 yang memiliki item 1, 16, 20, 26 dan 3 yang merupakan id barang dari Tissu Tessa, Fanta 1lt, Attack 500gr, Djarum Super 12 dan Pepsodent 100gr.

Aturan-aturan yang dihasilkan tidak bertentangan dengan *frequent itemset* tertinggi karena item-item yang terdapat pada aturan juga terdapat pada *frequent itemset* tertinggi.

Dari hasil pengujian dapat dilihat juga, bahwa jumlah data dalam *dataset* tidak mempengaruhi banyaknya aturan yang ditemukan dan juga lama waktu proses pencarian aturan asosiasi, ini bisa dilihat dengan membandingkan hasil pengujian menggunakan *dataset* transaksi berdasarkan jenis pekerjaan “mahasiswa” dan menggunakan *dataset* transaksi berdasarkan jenis pekerjaan “karyawan”.

Pada pengujian menggunakan *dataset* transaksi berdasarkan jenis pekerjaan “mahasiswa” dengan jumlah data dalam *dataset* sebanyak 1169 menghasilkan 2 aturan yang memenuhi minimum *confidence*, dengan lama proses 1,3964 menit, sedangkan dalam pengujian menggunakan *dataset* transaksi berdasarkan jenis pekerjaan “karyawan” dengan jumlah data dalam *dataset* yang lebih sedikit, yaitu 908, menghasilkan 17 aturan yang memenuhi minimum *confidence* dengan lama waktu proses yang lebih lama, yaitu 1,6213 menit. Hal ini terjadi karena dipengaruhi proses pembentukan *frequent itemset*, proses *pruning* dan juga karena adanya batasan – batasan yang telah diberikan, dalam hal ini yaitu parameter *apriori*.

Dalam pencarian *association rule* batasan-batasan (*constraints*) sangat diperlukan, karena jika mencari semua aturan yang ada dalam *database*, maka akan diperoleh aturan yang sangat banyak tetapi tidak terfokus [HAN00] (contohnya pada pengujian menggunakan keseluruhan data transaksi).

Dengan melihat hal ini, pengujian menggunakan *dataset* transaksi berdasarkan jenis pekerjaan “mahasiswa” dan pengujian menggunakan *dataset* transaksi berdasarkan jenis pekerjaan “karyawan”, lebih realistis dan terfokus karena dalam pembuatan *dataset* yang akan diproses (pada proses input *query*) telah diberikan batasan, pada pengujian menggunakan *dataset* transaksi berdasarkan jenis pekerjaan “mahasiswa” *dataset* yang terbentuk adalah hanya dari *member* yang mempunyai pekerjaan “mahasiswa” sedangkan pada pengujian menggunakan *dataset* transaksi berdasarkan jenis pekerjaan “karyawan” hanya *member* yang mempunyai pekerjaan “karyawan”.

BAB V

KESIMPULAN DAN SARAN

5.1. Kesimpulan

Dari pembuatan sistem ini, dapat diambil kesimpulan bahwa :

1. Konsep *data mining* menggunakan teknik aturan asosiasi dengan algoritma *apriori* berhasil diterapkan.
2. Sistem aplikasi tidak tergantung terhadap database, sehingga dapat digunakan untuk database yang lain dengan menggunakan query yang tepat.
3. Sistem aplikasi dapat menemukan *frequent itemset* apa saja yang terdapat di dalam database penjualan yang diproses.
4. Sistem aplikasi dapat digunakan untuk mencari pola-pola tersembunyi atau aturan-aturan asosiasi, yang sebelumnya tidak diketahui dari database penjualan.
5. Dalam pencarian *association rule* batasan-batasan (*constraints*) sangat diperlukan, agar aturan yang dihasilkan terfokus.

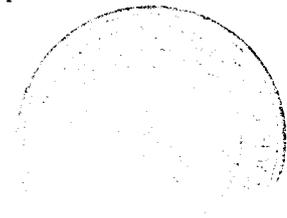
5.2. Saran

Beberapa saran untuk pengembangan dan penelitian selanjutnya sebagai berikut :

1. Pengukuran kepercayaan atau kekuatan aturan asosiasi tidak hanya menggunakan nilai *confidence* saja, tetapi ditambah dengan nilai *Lift*.
2. Sistem yang akan datang dapat menggunakan algoritma yang berbeda dari *apriori* untuk menemukan aturan asosiasi, yang dapat meminimalkan waktu pembentukan *frequent itemset*, sehingga waktu keseluruhan untuk mendapatkan aturan asosiasi menjadi se-minimal mungkin.
3. Pada pengembangan lebih lanjut dari sistem yang sudah dibuat, dapat dipakai data transaksi yang riil, sehingga dengan menggunakan metode-metode tertentu, aturan yang ditemukan dapat diteliti lebih lanjut apakah aturan yang ditemukan sesuai dengan kenyataannya atau tidak dan kenapa jika membeli barang A maka juga membeli barang B.

DAFTAR PUSTAKA

- [ADR96] Adriaans P., and Zantige D., Data Mining, Harlow : Addison – Wesley, 1996
- [AGR94] Agrawal R. and Srikant R., Fast Algorithm for Mining Association Rules, VLDB Conference, 1994
- [ARE04] Aref, Walid, Introduction to Data Mining, <<http://www.cs.purdue.edu/homes/cliffon/cs490d/Assoc.ppt>>, diakses terakhir pada 20-10-2006
- [BER04] Berry, Michael J. A. and Gordon Linoff, Data mining techniques : for marketing, sales, and customer relationship management 2nd ed, Indianapolis : Wiley Publishing, Inc., 2004
- [DUN02] Dunham, Margaret H. Data Mining, Introductory and Advanced Topics (Companion slides part II), Prentice Hall, 2002.
- [HAN00] Han , Jiawei and Kamber , Micheline, Data Mining: Concepts and Techniques, San Francisco : Morgan Kaufmann Publishers, 2000
- [HAN01] Hand, David, Mannila, Heikki and Smyth Padhraic, Principles Of Data Mining, Massachusetts : The MIT Press, 2001
- [KAN03] Kantardzic, Mehmed, Data Mining: Concepts, Models, Methods, and Algorithms, New Jersey : John Wiley and Sons, 2003
- [LAR05] Larose, Daniel T. Discovering knowledge in data : an introduction to data mining, New Jersey : John Wiley and Sons, 2005
- [PRA03] Pramudiono I., Pengantar Data Mining: Menambang Permata Pengetahuan di Gunung Permata, Kuliah Umum Ilmukomputer.com, 2003
- [SUC03] Sucahyo Y., Data Mining Menggali Informasi Yang Terpendam, Kuliah Umum Ilmukomputer.com, 2003



- [TAN05] Tang, ZhaoHui and MacLennan, Jamie, Data Mining with SQL Server 2005, Indianapolis : Wiley Publishing, Inc.,2005
- [TAN06] Tan P., Steinbach M., and Kumar V., Association Analysis: Basic Concepts and Algorithms, <<http://www-users.cs.umn.edu/~kumar/dmbook/index.php>>, diakses terakhir 22-10-2006
- [WIT05] Witten , Ian H. and Eibe Frank., DATA MINING : Practical Machine Learning Tools and Techniques, San Francisco : Morgan Kaufmann, 2005
- [YE03] Ye, Nong.. The handbook of data mining / edited by Nong Ye, London : Lawrence Erlbaum Associates, 2003





LAMPIRAN

Cara pemakaian

The image shows a sequence of four screenshots from a software application, illustrating the process of connecting to a database and running a query. The application window title is "Association Rules Antar Komoditi Pada Pasar Perkebunan dengan Menggunakan Algoritma Apriori".

1. Isi Parameter Koneksi
The first screenshot shows the "Koneksi Database" dialog box. The fields are filled with the following information:
Host/P Address: localhost
Port: 3306
Name User: root
Paswörd: [masked]
Database: [dropdown menu]
An "OK" button is highlighted with a callout box.

2. Klik "OK" untuk koneksi ke Database
The second screenshot shows the "Test koneksi" button, which is highlighted with a callout box. The status bar at the bottom indicates "Belum terkoneksi ke database" (Not connected to database).

3. Isi Query pada editor
The third screenshot shows the "Editor Query" window. The status bar at the bottom indicates "Terkoneksi ke database : apriori_sample" (Connected to database: apriori_sample). The editor area contains the text "Kembali ke tampilan proses koneksi" (Return to connection process display). Below this, there are four callout boxes with instructions:
- "klik untuk Buka file query yang telah disimpan" (click to open saved query file)
- "klik untuk simpan query yang telah ditulis pada editor" (click to save query written in editor)
- "hapus query yang telah ditulis pada editor" (delete query written in editor)
- "4. klik untuk menjalankan query" (4. click to run query)

4. klik untuk menjalankan query
The fourth screenshot shows the "Eksekusi Query" button, which is highlighted with a callout box. The status bar at the bottom indicates "Terkoneksi ke Database : apriori_sample" (Connected to Database: apriori_sample).

Query Database

Editor Query	Hasil Query																																							
	<table border="1"> <thead> <tr> <th>idtrans</th> <th>idbrg</th> <th>unrt</th> </tr> </thead> <tbody> <tr><td>10</td><td>1</td><td>5</td></tr> <tr><td>10</td><td>3</td><td>5</td></tr> <tr><td>10</td><td>4</td><td>5</td></tr> <tr><td>20</td><td>2</td><td>5</td></tr> <tr><td>20</td><td>3</td><td>5</td></tr> <tr><td>20</td><td>5</td><td>5</td></tr> <tr><td>30</td><td>1</td><td>5</td></tr> <tr><td>30</td><td>2</td><td>5</td></tr> <tr><td>30</td><td>3</td><td>5</td></tr> <tr><td>30</td><td>5</td><td>5</td></tr> <tr><td>40</td><td>2</td><td>5</td></tr> <tr><td>40</td><td>5</td><td>5</td></tr> </tbody> </table>	idtrans	idbrg	unrt	10	1	5	10	3	5	10	4	5	20	2	5	20	3	5	20	5	5	30	1	5	30	2	5	30	3	5	30	5	5	40	2	5	40	5	5
idtrans	idbrg	unrt																																						
10	1	5																																						
10	3	5																																						
10	4	5																																						
20	2	5																																						
20	3	5																																						
20	5	5																																						
30	1	5																																						
30	2	5																																						
30	3	5																																						
30	5	5																																						
40	2	5																																						
40	5	5																																						

Tampilan data yang akan diproses beserta keterangan jumlah data dan jumlah atribut

Jumlah data : 12 Jumlah atribut : 5

5. Klik untuk melanjutkan ke proses pencarian Association Rule

Klik untuk proses selanjutnya

Terkoneksi ke Database : apriori_sample 30 September 2007 18:06:24 Muhammad Wisnu K. (01523094) ©2007

Association Rules Antar Komoditi Pada Pusat Perbelanjaan dengan Menggunakan Algoritma Apriori

Proses Association Rule

Frequent Itemset Aturan Asosiasi

6. Isi parameter apriori memenuhi min

Batas Atas Support	3
Batas Bawah Support	2
Minimum Confidence	1

7. Klik untuk mencari Association Rule dari data yang telah diinputkan

Proses

Association Rules Antar Komoditi Pada Pusat Perbelanjaan
Algoritma Ar
Setelah proses pencarian Association Rule selesai, aturan yang ditemukan akan ditampilkan

Frequent Itemset Aturan Asosiasi

Aturan yang memenuhi minimal confidence :

1. jika membeli Tissue Paseo, Pepsodent 100gr maka mungkin membeli juga Pampers Bayi (support : 2, confid
2. jika membeli Pepsodent 100gr, Pampers Bayi maka mungkin membeli juga Tissue Paseo (support : 2, confid

Klik untuk menyimpan aturan-aturan yang didapat

