

**APLIKASI ANALISA KERANJANG BELANJA DENGAN METODE  
APRIORI UNTUK REKOMENDASI TREND PENJUALAN  
PADA ARTPEDIASHOP**

**SKRIPSI**

**Diajukan sebagai salah satu syarat untuk memperoleh kelulusan  
Jenjang Strata Satu (S1)  
Pada program Studi Sistem Informasi**

Oleh :

Anita Rahayu  
351761005



**SEKOLAH TINGGI MANAJEMEN INFORMATIKA DAN KOMPUTER  
INDONESIA MANDIRI  
BANDUNG  
2020**

## **LEMBAR PENGESAHAN**

### **APLIKASI ANALISA KERANJANG BELANJA DENGAN METODE APRIORI UNTUK REKOMENDASI TREND PENJUALAN PADA ARTPEDIASHOP**

Oleh  
Anita Rahayu  
351761005

Skripsi ini telah diterima dan disahkan untuk memenuhi persyaratan pencapaian  
gelar  
SARJANA KOMPUTER

Pada  
PROGRAM STUDI SISTEM INFORMASI  
SEKOLAH TINGGI MANAJEMEN INFORMATIKA DAN KOMPUTER  
INDONESIA MANDIRI

Bandung, Oktober 2020  
Disahkan oleh

**Ketua Program Studi,**

**Dosen Pembimbing,**

**Moch Ali Ramdhani, S.T., M.Kom.**  
NIDN: 0403097701

**Dr. Pahlawan Sagala**  
NIDN: 0422065802

## LEMBAR PERSETUJUAN REVISI

### APLIKASI ANALISA KERANJANG BELANJA DENGAN METODE APRIORI UNTUK REKOMENDASI TREND PENJUALAN PADA ARTPEDIASHOP

Oleh  
Anita Rahayu  
351761005

Telah melakukan sidang skripsi dan melakukan revisi sesuai dengan perubahan dan perbaikan yang diminta pada saat sidang skripsi

Bandung, Oktober 2020  
Menyetujui

No	Nama Dosen	Keterangan	Tanda Tangan
1	Dr. Pahlawan Sagala	Pembimbing	
2	Hendra Gunawan, S.T.,M.Kom	Penguji 1	
3	Yudhi W Arthana R., S.T.,M.Kom	Penguji 2	

Mengetahui,  
Ketua Program Studi Sistem Informasi

Moch. Ali Ramdhani S.T., M.Kom.  
NIDN. 0403097701

## LEMBAR PERNYATAAN

Dengan ini saya menyatakan bahwa :

- 1) Naskah Skripsi ini adalah asli dan belum pernah di ajukan untuk mendapatkan gelar akademik, baik di Sekolah Tinggi Manajemen Informatika dan Komputer Indonesia Mandiri maupun perguruan tinggi lainnya.
- 2) Laporan Skripsi ini murni merupakan karya penelitian saya sendiri dan tidak menjiplak karya pihak lain. Dalam hal ada bantuan atau arahan dari pihak lain maka telah saya sebutkan identitasnya dan jenis bantuannya di dalam lembar ucapan terima kasih.
- 3) Seandainya ada karya pihak lain yang ternyata memiliki kemiripan dengan karya saya ini, maka hal ini adalah diluar pengetahuan saya dan terjadi tanpa kesengajaan dari pihak saya.

Pernyataan ini penulis buat dengan sesungguhnya dan apabila kemudian hari terbukti adanya kebohongan dalam pernyataan ini, maka saya bersedia menerima sanksi akademik sesuai norma yang berlaku di Sekolah Tinggi Manajemen Informatika dan Komputer Indonesia Mandiri.

Bandung, Oktober 2020

Anita Rahayu  
351761005

## **ABSTRAK**

### **APLIKASI ANALISA KERANJANG BELANJA DENGAN METODE APRIORI UNTUK REKOMENDASI TREND PENJUALAN PADA ARTPEDIASHOP**

Oleh :  
Anita Rahayu  
351761005

Artpedia adalah salah satu perusahaan retail yang bergerak di bidang alat tulis kantor dan alat kesenian. Akan tetapi untuk mengikuti trend penjualan yang sedang ramai saat itu data penjualan yang ada belum di manfaatkan dengan baik padahal data tersebut mengandung informasi yang dapat di proses untuk keperluan yang lebih luas. Salah satunya untuk menemukan hubungan khusus antar produk yang di beli bersamaan. Berdasarkan hubungan tersebut, dimungkinkan melakukan promosi barang dengan pola keterkaitan barang yang ada dalam penjualan. Oleh karena itu, penulis mengambil skripsi berjudul “Aplikasi Analisa Keranjang Belanja Dengan Metode Apriori Untuk Rekomendasi Trend Penjualan Pada Artpediashop”. Metode yang diguankan dalam penerapan aplikasi ini dengan Data mining merupakan proses analisa untuk menentukan suatu pola dari kumpulan data yang mampu menganalisa data yang besar menjadi informasi berupa pola yang mempunyai arti bagi pendukung keputusan. Pendekatan yang digunakan dalam penelitian ini adalah algoritma apriori merupakan salah satu teknik *assosiation rule* yang digunakan untuk analisa keranjang belanja. Dengan menggunakan teknik *assosiation rule* menghasilkan suatu aturan asosiasi antar produk-produk yang di jual secara bersamaan dalam suatu transaksi. Hasil yang di capai dalam penelitian ini mendapatkan pola penjualan pembelian suatu produk dengan itemset 1,2 dan 3 item produk serta memberikan aturan analisa asosiainya.

Kata kunci : Data mining, Algoritma Apriori, Aturan Asosiasi, penjualan alat tulis kantor, produk bundling

## **ABSTRACT**

### **APPLICATION MARKET BASKET ANALYSIS USE APRIORI METHOD FOR RECOMMENDATIONS ON SALES TRENDS AT ARTPEDIASHOP**

By:  
Anita Rahayu  
351761005

*Artpedia is a retail company that is engaged in office stationery and art supplies. However, to follow the sales trend which was busy at that time, the existing sales data had not been used properly, even though the data contained information that could be processed for broader purposes. One of them is to find a special relationship between products that are purchased together. Based on this relationship, it is possible to promote goods with a pattern of related goods in sales. Therefore, the author takes a thesis entitled "Shopping Cart Analysis Application Using the Priori Method for Recommendations on Sales Trends at Artpediashop". The method used in implementing this application with data mining is an analysis process to determine a pattern from a data set that is able to analyze large data into information in the form of patterns that have meaning for decision support. The approach used in this study is the a priori algorithm, which is one of the association rule techniques used for shopping cart analysis. Using the association rule technique produces an association rule between products that are sold simultaneously in a transaction. The results achieved in this study were to obtain a sales pattern for purchasing a product with itemset 1,2 and 3 product items and providing rules for the analysis of the association.*

*Keyword : Data mining, Algoritma Apriori, Assosation rule, Sale ATK, bundling products*

## KATA PENGANTAR

Bismillahirrahmanirahim, Dengan Memanjatkan puji dan syukur kehadirat Allah SWT atas segala limpahan Rahmat dan karunia-Nya, Penulis dapat menyelesaikan laporan skripsi yang dilaksanakan di Artpedia yang berjudul “APLIKASI ANALISA KERANJANG BELANJA DENGAN METODE APRIORI UNTUK REKOMENDASI TREND PENJUALAN PADA ARTPEDIASHOP”. Laporan Skripsi ini bertujuan untuk memenuhi persyaratan Strata Satu di STMIK-Indonesia Mandiri Bandung, serta sebagai syarat untuk menyelesaikan skripsi untuk memperoleh kelulusan.

Dengan segala kerendahan hati penulis menyadari bahwa skripsi ini masih jauh dari kesempurnaan karena keterbatasan pengetahuan dan kemampuan serta pengalaman yang penulis miliki. Untuk itu dengan penulis mengharapkan kritik dan saran yang membangun untuk kemajuan penulis dikemudian hari. Penulis berharap skripsi ini bisa menjadi salah satu sumber yang dapat bermanfaat.

Bandung, Oktober 2020

Anita Rahayu  
351761005

## UCAPAN TERIMAKASIH

Pada Kesempatan ini penulis menyampaikan terima kasih kepada dosen pembimbing, Rekan-rekan, serta semua pihak yang ikut serta membantu penulis dalam menyusun laporan ini :

- 1) Bapak Dr Pahlawan Sagala. Selaku dosen pembimbing, atas bimbingan, saran dan pengarahannya sehingga pelaksanaan dan penulisan laporan skripsi ini dapat diselesaikan dengan baik.
- 2) Bapak Dr.Chairuddin, Ir., M.M., M.T . Selaku Ketua STMIK- IM Bandung.
- 3) Bapak Moch Ali Ramdhani S.T M.Kom, selaku Ketua Program Studi Sistem Informasi di STMIK-IM
- 4) Bapak dan Ibu dosen yang telah memberikan penulis ilmu yang bermanfaat selama melakukan kegiatan belajar mengajar di STMIK-IM Bandung.
- 5) Kedua Orang Tua Penulis. Ibu dan bapak yang selalu mendukung dan memberi semangat untuk kelancaran Skripsi
- 6) Rekan-rekan di STMIK-IM Bandung dan di Artpedia. Yang telah memberikan dukungan dan semangat serta selalu mendoakan penulis dalam penyusunan laporan kerja praktek.
- 7) Artpedia yang telah mengizinkan penulis untuk melakukan penelitian guna kelancaran penyusunan skripsi.



- 8) Andika Dyah Iriyanti , Nida Muffidah , Virna Des fernanda Tika Bahari dan Nadiadin Kulsum . Sahabat yang selalu setia mendukung dan memberi semangat selama masa perkuliahan hingga lulus.
- 9) Seluruh pihak-pihak yang tidak dapat penulis sebutkan satu persatu yang telah membantu penulis baik secara langsung maupun tidak langsung yang memberikan semangat kepada penulis dalam menyelesaikan Skripsi ini.

Penulis menyadari masih terdapat banyak kekurangan yang terdapat dalam laporan ini. Semoga skripsi ini dapat bernilai Ibadah disisi Allah SWT. Penulis juga berharap, Semoga Laporan Skripsi ini dapat memberikan wawasan serta manfaat untuk pembaca.

Bandung, Oktober 2020

Anita Rahayu  
351761005

## DAFTAR ISI

LEMBAR PENGESAHAN .....	ii
LEMBAR PERSETUJUAN REVISI.....	iii
LEMBAR PERNYATAAN.....	iv
ABSTRAK.....	v
ABSTRAC .....	vi
KATA PENGANTAR.....	vii
UCAPAN TERIMAKASIH.....	viii
DAFTAR ISI.....	x
DAFTAR TABEL .....	xiii
DAFTAR GAMBAR.....	xiv
BAB I PENDAHULUAN .....	16
1.1 Latar Belakang .....	16
1.2 Identifikasi Masalah .....	18
1.3 Tujuan Penulisan .....	18
1.4 Batasan Masalah.....	18
1.5 Metode Penelitian .....	19
1.5.1 Metode Pengumpulan Data.....	19
1.5.2 Metode Pengembangan Sistem .....	20
6.1 Sistematika Penulisan.....	21
BAB II LANDASAN TEORI.....	23
2.1 Konsep Dasar Sistem .....	23
2.1.1 Pengertian Sistem.....	23
2.1.2 Komponen dari Sistem Informasi.....	23
2.1.3 Karakteristik Sistem .....	26
2.2 Aplikasi .....	28
2.2.1 Pengertian Web.....	29
2.2.2 Sejarah Website .....	29
2.2.3 Perkembangan Web.....	30
2.3 PHP (Hypertext Preprocessor).....	34

2.3.1	Sejarah PHP .....	35
2.3.2	Fungsi PHP.....	36
2.3.3	Syntax PHP .....	36
2.4	Mysql.....	38
2.5	Data Mining .....	40
2.5.1	Fungsi Data Mining.....	41
2.5.2	Algoritma Apriori ( <i>Market Basket Analysis</i> ) .....	43
2.6	Diagram UML .....	49
2.6.1	<i>Use Case</i> diagram .....	49
2.6.2	<i>Activity</i> Diagram .....	50
2.6.3	<i>Class</i> Diagram .....	52
2.6.4	<i>Sequence</i> Diagram.....	53
<b>BAB III ANALISIS MASALAH DAN PERANCANGAN PROGRAM .....</b>		<b>41</b>
3.1	Pengumpulan Data .....	41
3.1.1	Studi Observasi .....	41
3.1.2	Wawancara.....	41
3.1.3	Studi Dokumen.....	41
3.2	Metode Pengembangan Sistem .....	42
3.2.1	Analisa Masalah .....	42
3.2.2	Analisa sitem yang sedang berjalan.....	43
3.2.3	Sistem yang di sarankan .....	44
3.3	Perancangan Sistem.....	47
3.3.1	<i>Desain Sistem dengan Unified Modeling Language (UML)</i> .....	47
3.3.2	Desain Database Sistem .....	66
3.3.3	Desain Interface Sistem.....	70
<b>BAB IV UJI COBA DAN IMPLEMENTASI SISTEM .....</b>		<b>74</b>
4.1	Implementasi.....	74
4.1.1	Implementasi Perangkat Keras .....	74
4.1.2	Implementasi Perangkat Lunak .....	74
4.1.3	Implementasi database.....	74
4.1.3	Implementasi Antarmuka .....	80
4.2	Uji Coba .....	83
<b>BAB V PENUTUP.....</b>		<b>89</b>
5.2	Kesimpulan .....	89

<b>4.2</b>	<b>Saran</b> .....	89
	<b>DAFTAR PUSTAKA</b> .....	90
	<b>LAMPIRAN</b> .....	92

## DAFTAR TABEL

TABEL: 2.1 Simbol <i>Usecase</i> Diagram .....	49
TABEL: 2.2 Simbol <i>Activty</i> Diagram .....	51
TABEL: 2.3 Simbol <i>Class</i> Diagram .....	52
TABEL: 2.4 Simbol <i>Sequence</i> Diagram .....	53
TABEL: 3.1 Tabel Definisi Aktor .....	48
TABEL: 3.2 Tabel Definisi <i>Use Case</i> .....	48
TABEL: 3.3 Skenario <i>Use ase Login</i> .....	49
TABEL: 3.4 Skenario <i>Use Case</i> Data Transaksi.....	50
TABEL: 3.5 Skenario <i>Use Case</i> Proses Apriori .....	50
TABEL: 3.6 Skenario <i>Use Case</i> Hasil .....	51
TABEL: 3.7 Skenario <i>Use Case Logout</i> .....	52
TABEL: 3.8 Perancangan Database Tabel User .....	68
TABEL: 3.9 Tabel Perancangan Database Data Transaksi.....	68
TABEL: 3.10 Perancangan Database <i>process_log</i> .....	68
TABEL: 3.11 Tabel perancangan database <i>itemset1</i> .....	69
TABEL: 3.12 Tabel perancangan database <i>itemset2</i> .....	69
TABEL: 3.13 Tabel Perancangan database <i>itemset3</i> .....	69
TABEL: 3.14 Tabel Perancangan database <i>confidence</i> .....	70
TABEL: 4.1 Tabel Pengujian Login .....	84
TABEL: 4.2 Tabel Pengujian Halaman Utama .....	85
TABEL: 4.3 Tabel Pengujian Data Transaksi .....	85
TABEL: 4.4 Tabel Pengujian Proses Apriori .....	86
TABEL: 4.5 Tabel Pengujian Hasil .....	87
TABEL: 4.6 Tabel Pengujian <i>Logout</i> .....	88

## DAFTAR GAMBAR

GAMBAR: 1.1 Model Prototype.....	20
GAMBAR: 3.1 Flowchart sistem yang sedang berjalan.....	44
GAMBAR: 3.2 <i>Use Case Diagram</i> .....	49
GAMBAR: 3.3 <i>Activity Diagram Login</i> .....	53
GAMBAR: 3.4 <i>Activity Diagram Logout</i> .....	53
GAMBAR: 3.5 <i>Activity Diagram Data Transaksi (Choose file)</i> .....	54
GAMBAR: 3.6 <i>Activity Diagram Data Transaksi (upload)</i> .....	55
GAMBAR: 3.7 <i>Activity Diagram Data Transaksi (Delete)</i> .....	56
GAMBAR: 3.8 <i>Activity Diagram Proses Apriori (Memilih Tanggal)</i> .....	57
GAMBAR: 3.9 <i>Activity Diagram Min Support &amp; Confidence</i> .....	58
GAMBAR: 3.10 <i>Activity Diagram Hasil (View Rule)</i> .....	59
GAMBAR: 3.11 <i>Activity Diagram Hasil (Print)</i> .....	60
GAMBAR: 3.12 <i>Sequence diagram login</i> .....	61
GAMBAR: 3.13 <i>Sequence Diagram logout</i> .....	61
GAMBAR: 3.14 <i>Sequence diagram memilih file</i> .....	62
GAMBAR: 3.15 <i>Sequence diagram upload</i> .....	62
GAMBAR: 3.16 <i>Sequence diagram delete</i> .....	63
GAMBAR: 3.17 <i>Sequence diagram memilih tanggal</i> .....	63
GAMBAR: 3.18 <i>Sequence diagram memilih proses</i> .....	64
GAMBAR: 3.19 <i>Sequence diagram view rule</i> .....	64
GAMBAR: 3.20 <i>Sequence diagram print</i> .....	65
GAMBAR: 3.21 <i>Class Diagram</i> .....	66
GAMBAR: 3.22 ERD Aplikasi Analisa Keranjang Belanja .....	67
GAMBAR: 3.23 Desain Halaman Login .....	71
GAMBAR: 3.24 Desain Halaman Utama .....	71
GAMBAR: 3.25 Desain Halaman Data Transaksi .....	72
GAMBAR: 3.26 Desain Halaman Proses Apriori .....	72
GAMBAR: 3.27 Desain Halaman Hasil .....	73
GAMBAR: 4.1 Entitas Pada database .....	75

GAMBAR: 4.2 Tabel User .....	75
GAMBAR: 4.3 Tabel data transaksi .....	76
GAMBAR: 4.4 Tabel process_log .....	76
GAMBAR: 4.5 Tabel itemset1 .....	77
GAMBAR: 4.6 Tabel itemset2 .....	78
GAMBAR: 4.7 Tabel itemset3 .....	78
GAMBAR: 4.8 Tabel Confidence .....	80
GAMBAR: 4.9 Halaman Login .....	81
GAMBAR: 4.10 Halaman Utama .....	81
GAMBAR: 4.11 Halaman Data Transaksi .....	82
GAMBAR: 4.12 Halaman proses apriori .....	82
GAMBAR: 4.13 Memilih periode tanggal mulai dan selesai .....	83
GAMBAR: 4.14 Halaman Hasil .....	83

# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang

Ketatnya Persaingan dalam dunia bisnis khususnya dalam industri penjualan, membuat para pelaku usaha harus pintar dalam menganalisis pasar dan mencari peluang agar dapat bertahan di bidangnya. Perubahan kondisi dari dalam dan luar perusahaan dapat mempengaruhi pola beli konsumen sangat penting untuk bisa di hadapi, Pengambilan keputusan bisnis yang cepat dan tepat adalah solusi agar perusahaan dapat beradaptasi pada perubahan-perubahan yang terjadi. Salah satu faktor yang harus di analisis adalah pola penjualan supaya kebutuhan pelanggan terpenuhi, untuk mendukung hal tersebut data penjualan dapat dimanfaatkan untuk menganalisa pasar dan kebutuhan pelanggan

Artpedia adalah sebuah perusahaan yg bergerak di bidang penjualan alat tulis kantor dan alat kesenian, Dengan adanya kegiatan penjualan setiap hari, data semakin lama akan semakin bertambah banyak. Data tersebut tidak hanya berfungsi sebagai arsip bagi perusahaan, data tersebut dapat dimanfaatkan dan diolah menjadi informasi yang berguna untuk peningkatan penjualan dan promosi produk. *Data mining* adalah salah satu ilmu yang dapat diterapkan dalam kasus seperti ini. proses mencari pola atau informasi menarik dalam data terpilih dengan menggunakan teknik atau metode tertentu. Salah satu metode yang seringkali digunakan dalam teknologi *data mining* adalah metode asosiasi atau *association rule mining*. Di dalam bidang usaha *retail* metode *association rule mining* ini lebih dikenal dengan



istilah analisa keranjang belanja (*market basket analysis*). Dalam teknik data mining tersebut terdapat beberapa algoritma untuk menyelesaikannya. Beberapa algoritma tersebut antara lain algoritma *hash-based*, algoritma 10 apriori, dan algoritma FP-Growth. Sedangkan dalam penelitian ini akan menggunakan algoritma apriori. data transaksi penjualan dapat diolah kembali sehingga bisa menghasilkan suatu pola. (marsela dan veronica : 2004) Algoritma apriori adalah algoritma *market basket analysis* yang digunakan untuk menghasilkan *association rule*, dengan pola “*if then* ”. *Market basket analysis* merupakan salah satu teknik dari *data mining* yang mempelajari tentang perilaku kebiasaan konsumen dalam membeli barang secara bersamaan dalam satu waktu. Teknik tersebut bisa diterapkan dalam data yang sangat besar seperti data transaksi penjualan.

Sehingga penulis membuat penelitian yang berjudul “**APLIKASI ANALISA KERANJANG BELANJA DENGAN METODE APRIORI UNTUK REKOMENDASI TREND PENJUALAN PADA ARTPEDIASHOP**”

. Dibuatkan suatu sistem analisa pembelian produk untuk mengolah data transaksi penjualan hingga menghasilkan informasi mengenai pola pembelian konsumen dengan bantuan metode algoritma apriori. algoritma ini digunakan untuk membentuk *frequent itemsets* yang nantinya akan dijadikan sebagai acuan untuk merumuskan aturan-aturan asosiasi yang dihasilkan oleh model *market basket analysis* untuk menentukan promosi-promosi apa yang perlu di terapkan manajemen untuk membantu pihak manajemen dalam membuat keputusan bisnis.

## 1.2 Identifikasi Masalah

Sesuai dengan latar belakang yang telah diuraikan sebelumnya, maka identifikasi masalah dari penelitian ini adalah:

1. Bagaimana memanfaatkan data penjualan agar menjadi sesuatu yang bermanfaat ?
2. Bagaimana cara agar memudahkan proses analisa data penjualan ?
3. Bagaimana cara meningkatkan penjualan dan menentukan promosi produk ?

## 1.3 Tujuan Penulisan

Sesuai dengan identifikasi masalah yang telah di uraikan sebelumnya, maka tujuan penelitian ini adalah:

1. Menyediakan aplikasi analisa keranjang belanja dengan aplikasi berbasis web.
2. Membuat aplikasi analisa keranjang belanja dengan memanfaatkan Algoritma Apriori untuk menentukan itemset1. Itemset2 dan itemset3.
3. Menyediakan hasil analisa untuk membantu manajer dalam meningkatkan penjualan dan menentukan produk promosi.

## 1.4 Batasan Masalah

Agar penelitian ini mencapai sasaran dan tujuan yang di harapkan maka penelitian ini diberikan batasan masalah :

- 1 Aplikasi yang dibuat hanya dapat di akses perusahaan dan karyawan.
- 2 Aplikasi berisi data keterkaitan antar produk dari data penjualan
- 3 Aplikasi digunakan untuk penjualan barang-barang Alat tulis kantor dan Kesenian.

## **1.5 Metode Penelitian**

### **1.5.1 Metode Pengumpulan Data**

Adapun teknik pengumpulan data yang dilakukan penulis dalam penelitian yang dilakukan, antara lain :

#### **1 Observasi**

Observasi adalah metode pengumpulan data yang kompleks karena melibatkan berbagai faktor dalam pelaksanaannya. Metode pengumpulan data observasi tidak hanya mengukur sikap dari responden, namun juga dapat digunakan untuk merekam berbagai fenomena yang terjadi. Teknik pengumpulan data observasi cocok digunakan untuk penelitian yang bertujuan untuk mempelajari perilaku manusia, proses kerja, dan gejala-gejala alam.

#### **2 Wawancara**

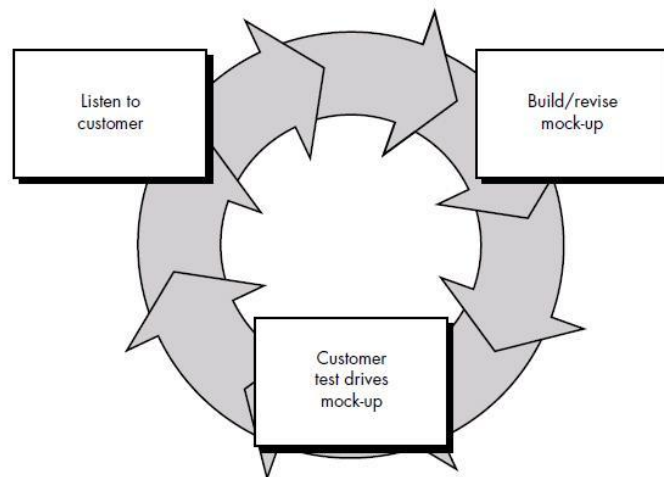
Wawancara adalah teknik pengumpulan data yang dilakukan melalui tatap muka dan tanya jawab langsung antara peneliti dan narasumber. Seiring perkembangan teknologi, metode wawancara dapat pula dilakukan melalui media-media tertentu, misalnya telepon, *email*, atau *skype*.

### 3 Studi Dokumen

Studi dokumen adalah metode pengumpulan data yang tidak ditujukan langsung kepada subjek penelitian. Studi dokumen adalah jenis pengumpulan data yang meneliti berbagai macam dokumen yang berguna untuk bahan analisis.

#### 1.5.2 Metode Pengembangan Sistem

Metode pengembangan sistem menggunakan Metode *Prototype*, yaitu :



**GAMBAR 1.1** Model Prototype menurut Roger S. Pressman, Ph.D.

- 1 **Pengumpulan Kebutuhan.** *Developer* dan klien bertemu dan menentukan tujuan umum , kebutuhan yang diketahui dan gambaran bagian - bagian yang akan dibutuhkan berikutnya . Detail kebutuhan mungkin tidak dibicarakan disini , pada awal pengumpulan kebutuhan.
- 2 **Perancangan.** Perancangan dilakukan cepat dan rancangan mewakili aspek software yang diketahui dan rancangan ini menjadi dasar pembuatan *prototype*.

- 3 **Evaluasi Prototype.** Klien mengevaluasi *prototype* yang dibuat dan dipergunakan untuk memperjelas kebutuhan *software*.

## 6.1 Sistematika Penulisan

Sistematika penulisan yang dilakukan penulis dalam pembuatan proposal penelitian ini terdiri dari beberapa bab antara lain :

### **BAB I                    PENDAHULUAN**

Pada bab ini membahas penjelasan mengenai latar belakang, identifikasi masalah, tujuan penelitian, batasan masalah, metode penelitian, dan sistematika penulisan yang dilakukan oleh penulis.

### **BAB II                    LANDASAN TEORI**

Pada bab ini berisikan tentang landasan teori yang diperoleh oleh penulis dari berbagai sumber yang ada, yang dapat mendukung terhadap analisa yang dilakukan oleh penulis, serta terdapat definisi mengenai aplikasi yang berkaitan.

### **BAB III                    ANALISIS MASALAH DAN PERANCANGAN PROGRAM**

Pada bab ini berisi tentang perancangan program berdasarkan dengan analisa permasalahan yang dilakukan oleh penulis sesuai dengan metode yang dipilih.

**BAB IV UJI COBA DAN IMPLEMENTASI SISTEM**

Pada bab ini berisikan tentang ujicoba aplikasi yang telah dirancang untuk mengetahui hasil yang di dapatkan sesuai dengan apa yang di harapkan

**BAB V PENUTUP**

Bab ini berisi kesimpulan dan saran yang diperoleh dari hasil penelitian skripsi yang telah dibuat. Kesimpulan didapatkan dari penelitian yang telah dilakukan sampai pembuatan laporan serta saran agar aplikasi dikembangkan menjadi lebih baik lagi untuk pengembangan selanjutnya.

## **BAB II**

### **LANDASAN TEORI**

#### **2.1 Konsep Dasar Sistem**

##### **2.1.1 Pengertian Sistem**

Sistem Informasi (SI) adalah kombinasi dari teknologi informasi dan aktivitas orang yang menggunakan teknologi itu untuk mendukung operasi dan manajemen. Dalam arti yang sangat luas, istilah sistem informasi yang sering digunakan merujuk kepada interaksi antara orang, proses algoritmik, data, dan teknologi. Dalam pengertian ini, istilah ini digunakan untuk merujuk tidak hanya pada penggunaan organisasi teknologi informasi dan komunikasi (TIK), tetapi juga untuk cara di mana orang berinteraksi dengan teknologi ini dalam mendukung proses bisnis.

Menurut (Sutabri, 2004) “Sistem informasi merupakan sistem dalam organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang mendukung fungsi manajerial organisasi dalam kegiatan organisasi untuk dapat menyediakan kepada pihak luar tertentu dengan laporan yang diperlukan.”

##### **2.1.2 Komponen dari Sistem Informasi**

komponen SI dalam prosedur yang berkaitan dengan manual prosedur dan prosedur berbasis komputer dan standar untuk mengolah data menjadi informasi yang berguna. Prosedur adalah urutan langkah-langkah yang diambil untuk menyelesaikan satu atau lebih kegiatan pengolahan informasi. Memproses

informasi ini dapat dilakukan oleh pengguna, atau kombinasi dari pengguna dan staf IT.

Sistem informasi terdiri dari komputer, instruksi, fakta yang tersimpan, manusia dan prosedur sehingga dapat dikategorikan dalam empat bagian:

1. Sistem Informasi Manajemen
2. Sistem Pendukung Keputusan
3. Sistem Informasi Eksekutif
4. Sistem Pemrosesan Transaksi

Data Mining Merupakan Salah satu Proses untuk menambang informasi yang berguna dari basis data. Dalam kegiatan transaksi di minimarket ataupun toko pemilik harus mengerti apa yang diinginkan konsumen dalam memberikan kenyamanan berbelanja, terutama kemudahan dalam pemilihan barang yang disesuaikan dengan tata letak. Penelitian ini Melakukan analisis pola data dengan menerapkan algoritma apriori yang ditunjukkan untuk frekuensi keranjang belanja.",

Menurut (Sutabri, 2004), Komponen-komponen sistem informasi yang dikenal dengan istilah blok bangunan (*building block*) yaitu :

1. Blok Masukan (*input block*)

Input mewakili data yang masuk kedalam sistem informasi, termasuk metode dan media untuk memperoleh data yang dimasukkan, yang dapat berupa dokumen dasar.

2. Blok Model (*model block*)

Terdiri dari kombinasi prosedur, logika dan model matematik yang akan memanipulasi atau mentransformasi data masukan dan data yang



tersimpan dalam basis data untuk menghasilkan keluaran yang diinginkan.

3. Blok Keluaran (*output block*)

Produk dari sistem informasi adalah keluaran berupa informasi yang berkualitas.

4. Blok Teknologi (*technology block*)

Teknologi merupakan kotak alat (*tool-box*) dalam sistem informasi. Teknologi digunakan untuk menerima *input*, menjalankan model, menyimpan dan mengakses data, menghasilkan dan mengirimkan keluaran dan membantu pengendalian dari sistem secara keseluruhan. Teknologi terdiri dari tiga bagian utama yaitu teknisi (*brainware*), perangkat lunak (*software*), dan perangkat keras (*hardware*).

5. Blok Basis Data (*database block*)

Merupakan kumpulan data yang saling berhubungan yang diorganisasikan sedemikian rupa agar dapat diakses dengan mudah dan cepat.

6. Blok Kendali (*control block*)

Pengendalian perlu dirancang dan diterapkan untuk meyakinkan bahwa hal-hal yang dapat merusak sistem dapat dicegah atau terlanjur terjadi kesalahan dapat langsung diatasi.

### 2.1.3 Karakteristik Sistem

Menurut (Sutabri, 2004) suatu sistem mempunyai karakteristik atau sifat-sifat tertentu, yaitu mempunyai komponen-komponen, batas sistem, lingkungan luar sistem, penghubung, masukan, keluaran, pengolah, dan sasaran atau tujuan.

#### 1. Komponen Sistem (*Components*)

Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi yang artinya saling bekerjasama membentuk suatu kesatuan. Komponen-komponen sistem atau elemen-elemen sistem dapat berupa suatu subsistem atau bagian-bagian dari sistem. Setiap subsistem mempunyai karakteristik dari sistem yang menjalankan suatu fungsi tertentu dan mempengaruhi proses sistem secara keseluruhan.

#### 2. Batas Sistem (*Boundary*)

Ruang lingkup sistem merupakan daerah yang membatasi antara sistem dengan sistem lainnya atau sistem dengan lingkungan luarnya. Batasan sistem ini memungkinkan suatu sistem dipandang sebagai satu kesatuan yang tidak dapat dipisah-pisahkan.

#### 3. Lingkungan Luar Sistem (*Environment*)

Lingkungan luar dari suatu sistem adalah apapun diluar batas sistem yang mempengaruhi operasi sistem. Lingkungan luar sistem dapat bersifat menguntungkan dan juga merugikan. Lingkungan luar yang menguntungkan merupakan energi dari sistem dan dengan demikian harus dijaga dan dipelihara. Sedangkan lingkungan luar yang merugikan harus

ditahan dan dikendalikan, jika tidak maka akan mengganggu kelangsungan hidup dari sistem.

#### 4. Penghubung Sistem (*Interface*)

Merupakan suatu media yang menghubungkan antara satu subsistem dengan subsistem lainnya. Melalui penghubung ini memungkinkan sumber-sumber daya mengalir dari satu subsistem ke subsistem lainnya. Keluaran dari satu subsistem akan menjadi masukan untuk subsistem lainnya melalui penghubung. Dengan penghubung satu sub sistem dapat berintegrasi dengan sub sistem yang lainnya membentuk satu kesatuan.

#### 5. Masukan Sistem (*Input*)

Masukan (*input*) adalah energi yang dimasukkan ke dalam sistem yang berupa masukan perawatan (*maintenance input*) agar sistem tersebut dapat beroperasi dan masukan sinyal yang diproses untuk menghasilkan keluaran. Sebagai contoh di dalam sistem komputer, program adalah *maintenance input* yang digunakan untuk mengoperasikan komputer dan data adalah signal *input* untuk diolah menjadi informasi.

#### 6. Keluaran Sistem (*Output*)

Merupakan hasil dari energi yang telah diproses dan dapat diklasifikasikan antara lain menjadi output yang langsung digunakan, *output* yang bisa dijadikan *input* untuk subsistem yang lain atau kepada supra sistem serta *output* yang bagi sistem tidak digunakan lagi (sisa pembuangan) tapi bagian sistem lain sangat berguna.

#### 7. Pengolah Sistem (*Process*)

Suatu sistem dapat mempunyai bagian pengolah atau sistem itu sendiri sebagai pengolahnya. Pengolah yang akan merupakan masukan menjadi keluaran. Sistem akuntansi akan mengolah data-data transaksi menjadi laporan-laporan keuangan dan laporan-laporan yang dibutuhkan oleh manajemen.

#### 8. Sasaran dan Tujuan

Suatu sistem dikatakan berhasil jika mengenai sasaran atau tujuannya, kalau sistem tidak mempunyai sasaran maka sistem tidak akan ada. Sasaran sangat berpengaruh pada masukan dan keluaran yang dihasilkan

## 2.2 Aplikasi

Dalam ilmu komputer pengertian aplikasi adalah suatu perangkat lunak (*software*) atau program komputer yang beroperasi pada sistem tertentu yang diciptakan dan dikembangkan untuk melakukan perintah tertentu. Adapun Menurut (Ali zaky, Smitdev Community, 2019) , pengertian aplikasi adalah suatu komponen yang bermanfaat sebagai media untuk mengoperasikan pengolahan data atau kegiatan lainnya.

Dalam pengembangannya aplikasi dapat dikategorikan dalam tiga kelompok, diantaranya :

#### 1. Aplikasi Desktop

Aplikasi yang hanya dijalankan di perangkat PC komputer atau laptop yang mampu beroperasi secara *offline* maupun *online*.

#### 2. Aplikasi Web

Aplikasi yang dijalankan menggunakan komputer dan koneksi internet.

### 3. Aplikasi Mobile

Aplikasi yang di jalankan di perangkat mobile seperti gawai ,*smartphone* dll.

#### 2.2.1 Pengertian Web

(Ali Zaki, Smitdev Community, 2019) mendefinisikan “Website adalah beberapa kumpulan dari halaman web yang terdapat pada satu domain, yang terdiri dari 2 atau lebih halaman web.”

Web merupakan suatu kebutuhan bagi masyarakat modern sekarang ini, baik itu digunakan untuk melakukan transaksi, penyebaran informasi, maupun pencarian informasi.

Web atau situs dapat diartikan sebagai kumpulan halaman yang menampilkan data teks, data gambar diam atau gerak, data animasi, suara, video dan gabungan dari semuanya, baik bersifat statis maupun dinamis yang membentuk suatu rangkaian bangunan yang saling terkait dimana masing-masing dihubungkan dengan jaringan-jaringan halaman (*hyperlink*). Web terdapat dua jenis, yaitu :

1. Web Statis, yaitu isi informasi websitenya tetap, jarang berubah, dan isi informasinya satu arah.
2. Web Dinamis, yaitu isi informasi websitenya selalu berubah-ubah dan isi informasinya interaktif dua arah.

#### 2.2.2 Sejarah Website

Pada tahun 1989 Tim Berners-Lee, seorang programmer komputer berkebangsaan Inggris yang bekerja pada European Pysics Laboratory (CERN) di

Genewa, Swiss, melakukan sesuatu yang berbeda dari apa yang telah dilakukan sebelumnya. Dia mengkombinasikan hypermedia dengan sumber-sumber informasi Internet yang sangat luas. Sebelum ada Web, Anda dapat melakukan banyak hal di internet, tetapi tidak satupun dapat dilakukan dengan mudah.

Solusi Berners-Lee adalah teknologi *hypertext* untuk membentuk sebuah dokumen Web. Tidak seperti buku atau kebanyakan database, Web memiliki banyak kemungkinan informasi ini dijalin, disembunyikan dengan suatu antar muka *hypertext* berbasis karakter. Dengan Web, seorang ahli fisika dapat dengan mudah melompat dari suatu artikel pada teori partikel di sebuah mesin lokal ke suatu kamus istilah-istilah fisika nuklir pada suatu sistem yang jauhnya ribuan mil.

Dokumen Web harus ditulis dalam suatu format khusus yang memungkinkan hypertext saling terjalin untuk bekerja. Format ini adalah *Hypertext Markup Language* (HTML). HTML merupakan bagian dari *Standard Generalized Markup Language* (SGML). SGML merupakan standar dari Internasional *Standards Organization* (ISO), untuk mendefinikasikan format pada dokumen teks. Meskipun SGI ditunjukan untuk *desktop publishing*, Berners-Lee dan rekan-rekannya mengambil kemampuan hyperlink untuk membentuk dasar dokumen Web yang pertama.

### **2.2.3 Perkembangan Web**

Web sedemikian berkembangnya sehingga para ahli telah memberikan penomoran untuk mengklasifikasikan generasi teknologi Web yang digunakan, yaitu:

### 1. **Web 1.0 (1990–2000)**

Web1.0 adalah generasi pertama dari website di internet. Dalam web 1.0 para pengunjung dapat mencari dan *browsing* data yang ada di dalam web, namun pengunjung tidak memiliki akses untuk memasukkan data atau informasi ke dalam situs. Pada masa web 1.0, layanan internet masih berkisar diantara static website yang saling dihubungkan dengan *hyperlink*. Website pada kala itu, hanya menyampaikan informasi satu arah dalam bentuk brosur online. Pada tahun 1998 Google berdiri dan internet semakin mudah untuk digunakan. Pada masa itu, website dengan format portal berita dan *took online* (example: Amazon.com) menjadi website yang paling diminati oleh para investor.

### 2. **Web 2.0 (2001–2010)**

Pada Web 2.0 pengembangan dimulai. Pada era ini, pengunjung dapat melakukan interaksi dengan diatur oleh system yang ada pada web. Web 2.0 sendiri merupakan istilah yang dicetuskan pada tahun 2003 oleh O'reilly Media lalu kemudia dipopulerkan pada konfrensi web 2.0 pada tahun 2004. Web 2.0 memiliki beberapa ciri khas, yaitu: *share, collaborate* dan *exploit*. Di dalam era web 2.0 penggunaan web untuk berbagi pertemanan, kolaborasi menjadi sesuatu yang penting.

Web 2.0 adalah revolusi bisnis di industri komputer yang disebabkan oleh penggunaan internet sebagai *platform*, dan merupakan suatu percobaan untuk memahami berbagai aturan untuk mencapai keberhasilan pada platform baru tersebut. Salah satu aturan terutama adalah: Membangun

aplikasi yang mengeksploitasi efek jaringan untuk mendapatkan lebih banyak lagi pengguna aplikasi tersebut.

Berbagai layanan berbasis web seperti jejaring sosial, wiki dan folksonomies (misalnya: “flickr.com”, “del.icio.us”) merupakan teknologi Web 2.0 yang menambah interaktifitas di antara para pengguna Web. Pada umumnya, Website yang dibangun dengan menggunakan teknologi Web 2.0 memiliki fitur-fitur sebagai berikut:

- CSS (*Cascading Style Sheets*)
- Aplikasi Rich Internet atau berbasis Ajax
- Markup XHTML
- Sindikasi dan agregasi data menggunakan RSS/Atom
- URL yang *valid*
- Folksonomies
- Aplikasi wiki pada sebagian atau seluruh Website
- XML Web-Service API

### 3. **Web 3.0 (2005-now)**

Pada masa ini, lambat laun kebiasaan dan kebutuhan para pengunjung dunia maya semakin bertambah dan selalu berubah. Hal ini juga sejalan dengan semakin cepatnya akses internet *broadband* dan teknologi komputer yang semakin canggih.

Walaupun masih dalam perdebatan di kalangan analis dan peneliti, istilah Web 3.0 tetap berpotensi menjadi generasi teknologi di dunia Internet.



Saat ini, definisi untuk Web 3.0 sangat beragam mulai dari pengaksesan broadband secara mobile sampai kepada layanan Web berisikan perangkat lunak bersifat *on-demand*. Namun, menurut John Markoff, Web 3.0 adalah sekumpulan teknologi yang menawarkan cara baru yang efisien dalam membantu komputer mengorganisasi dan menarik kesimpulan dari data online. Berdasarkan definisi yang dikemukakan tersebut, maka pada dasarnya Semantic Web memiliki tujuan yang sama karena Semantic Web memiliki isi Web yang tidak dapat hanya diekspresikan di dalam bahasa alami yang dimengerti manusia, tetapi juga di dalam bentuk yang dapat dimengerti, diinterpretasi dan digunakan oleh perangkat lunak (*software agents*). Melalui Semantic Web inilah, berbagai perangkat lunak akan mampu mencari, membagi, dan mengintegrasikan informasi dengan cara yang lebih mudah.

Web 3.0 terdiri dari: Web semantic, Format mikro, Pencarian dalam bahasa pengguna, Penyimpanan data dalam jumlah besar, Pembelajaran lewat mesin, Agen rekomendasi yang merujuk pada kecerdasan buatan Web.

Sekarang perkembangan web sangat beragam baik disisi tampilanya maupun di sisi bahasa pemrograman yang digunakan. Sehingga setiap orang akan lebih mudah tentunya dalam membuat web dan mempergunakanya. Karena tentu saja kembali ke fungsi teknologi sendiri, yaitu untuk memudahkan manusia dalam melakukna sesuatu hal.

Berdasarkan definisi yang dikemukakan tersebut, maka pada dasarnya Semantic Web memiliki tujuan yang sama karena Semantic Web memiliki isi Web yang tidak dapat hanya diekpresikan di dalam bahasa alami yang dimengerti manusia, tetapi juga di dalam bentuk yang dapat dimengerti, diinterpretasi dan digunakan oleh perangkat lunak (*software agents*). Melalui Semantic Web inilah, berbagai perangkat lunak akan mampu mencari, membagi, dan mengintegrasikan informasi dengan cara yang lebih mudah

### 2.3 PHP (Hypertext Preprocessor)

Menurut (Arief, 2011c:43) “PHP adalah Bahasa *server-side –scripting* yang menyatu dengan HTML untuk membuat halaman web yang dinamis.” Karena PHP merupakan *server-side-scripting* maka sintaks dan perintah-perintah PHP akan dieksekusi di server kemudian hasilnya akan dikirimkan ke browser dengan format HTML.

PHP adalah singkatan dari "*PHP: Hypertext Preprocessor*", yaitu bahasa pemrograman yang digunakan secara luas untuk penanganan pembuatan dan pengembangan sebuah situs web dan bisa digunakan bersamaan dengan CSS dan HTML. PHP diciptakan oleh Dr. Leonardo Bernart pertama kali tahun 1994. Pada awalnya PHP adalah singkatan dari "*Personal Home Page Tools*". Selanjutnya diganti menjadi FI ("*Forms Interpreter*"). Sejak versi 3.0, nama bahasa ini diubah menjadi "*PHP: Hypertext Preprocessor*" dengan singkatannya "PHP".

### 2.3.1 Sejarah PHP

Pada awalnya PHP merupakan kependekan dari Personal Home Page (Situs personal). PHP pertama kali dibuat oleh Rasmus Lerdorf pada tahun 1995. Pada waktu itu PHP masih bernama *Form Interpreted* (FI), yang wujudnya berupa sekumpulan skrip yang digunakan untuk mengolah data formulir dari web.

Selanjutnya Rasmus merilis kode sumber tersebut untuk umum dan menamakannya PHP/FI. Dengan perilsan kode sumber ini menjadi sumber terbuka, maka banyak pemrogram yang tertarik untuk ikut mengembangkan PHP.

Pada November 1997, dirilis PHP/FI 2.0. Pada rilis ini, *interpreter* PHP sudah diimplementasikan dalam program C. Dalam rilis ini disertakan juga modul-modul ekstensi yang meningkatkan kemampuan PHP/FI secara signifikan.

Pada tahun 1997, sebuah perusahaan bernama Zend menulis ulang interpreter PHP menjadi lebih bersih, lebih baik, dan lebih cepat. Kemudian pada Juni 1998, perusahaan tersebut merilis interpreter baru untuk PHP dan meresmikan rilis tersebut sebagai PHP 3.0 dan singkatan PHP diubah menjadi akronim berulang PHP: *Hypertext Preprocessing*.

Pada pertengahan tahun 1999, Zend merilis interpreter PHP baru dan rilis tersebut dikenal dengan PHP 4.0. PHP 4.0 adalah versi PHP yang paling banyak dipakai pada awal abad ke-21. Versi ini banyak dipakai disebabkan kemampuannya untuk membangun aplikasi web kompleks tetapi tetap memiliki kecepatan dan stabilitas yang tinggi.

Pada Juni 2004, Zend merilis PHP 5.0. Dalam versi ini, inti dari interpreter PHP mengalami perubahan besar. Versi ini juga memasukkan model pemrograman

berorientasi objek ke dalam PHP untuk menjawab perkembangan bahasa pemrograman ke arah paradigma berorientasi objek. Server web bawaan ditambahkan pada versi 5.4 untuk mempermudah pengembang menjalankan kode PHP tanpa menginstall *software server*.

Versi terbaru dan stabil dari bahasa pemrograman PHP saat ini adalah versi 7.0.16 dan 7.1.2 yang resmi dirilis pada tanggal 17 Februari 2017.

### **2.3.2 Fungsi PHP**

Fungsi PHP adalah membuat atau mengembangkan situs web statis atau situs web dinamis atau aplikasi Web. Walaupun sebenarnya bukan hanya PHP bahasa pemrograman yang bisa digunakan untuk memuat website.

PHP digunakan karena untuk membuat website dinamis bisa digunakan untuk menyimpan data ke dalam database, membuat halaman yang dapat berubah-ubah sesuai dengan input *user*, memproses *form*, dll.

Dalam membuat file PHP dapat digabung menggunakan tag html, Dan ketika tanpa menggunakan tag html apa pun disebut file PHP Murni. Server menginterpretasikan kode PHP dan mengeluarkan hasilnya sebagai kode HTML ke browser web. Agar server mengidentifikasi kode PHP dari kode HTML, kita harus selalu menyertakan kode PHP dalam tag PHP.

### **2.3.3 Syntax PHP**

Pengertian Syntax PHP adalah aturan penulisan agar mampu dimengerti dengan benar oleh *compiler* saat membaca bahasa pemrograman. Dalam penulisan PHP yang benar diawali dengan “<?php” dan diakhiri dengan “?>”. Dan di dalam File PHP juga dapat berisi tag seperti HTML dan skrip sisi klien seperti JavaScript.

## Sintaksis Dasar PHP

### 1. Pembatas

PHP hanya mengeksekusi kode yang ditulis dalam pembatas sebagaimana ditentukan oleh dasar sintaks PHP. Apapun di luar pembatas tidak diproses oleh PHP (meskipun teks PHP ini masih mengendalikan struktur yang dijelaskan dalam kode PHP. Pembatas yang paling umum adalah "<?php" untuk membuka dan ">" Untuk menutup kode PHP. Tujuan dari pembatas ini adalah untuk memisahkan kode PHP dari kode di luar PHP, seperti HTML, JavaScript.

### 2. Variabel

Variabel diawali dengan simbol dolar \$. Pada versi php PHP 5 diperkenalkan jenis isyarat yang memungkinkan fungsi untuk memaksa mereka menjadi parameter objek dari *class* tertentu, *array*, atau fungsi. Namun, jenis petunjuk tidak dapat digunakan dengan jenis skalar seperti angka atau *string*. Contoh variabel dapat ditulis sebagai \$nama\_variabel.

Penulisan fungsi, penamaan kelas, nama variabel adalah peka akan huruf besar (Kapital) dan huruf kecil . Kedua kutip ganda "" dari string memberikan kemampuan untuk interpolasi nilai variabel ke dalam string PHP. PHP menerjemahkan baris sebagai spasi, dan pernyataan harus diakhiri dengan titik koma ;.

### 3. Komentar

PHP memiliki 3 jenis sintaks sebagai komentar pada kode yaitu tanda blok / \* \* / , komentar 2 baris // Serta tanda pagar # digunakan untuk komentar satu

baris. Komentar bertujuan untuk meninggalkan catatan pada kode PHP dan tidak akan diterjemahkan ke program.

#### 4. Fungsi

Ratusan fungsi yang disediakan oleh PHP serta ribuan lainnya yang tersedia melalui berbagai ekstensi tambahan. fungsi-fungsi ini didokumentasikan dalam dokumentasi PHP. Namun, dalam berbagai tingkat pengembangan, kini memiliki berbagai konvensi penamaan.

### 2.4 Mysql

Menurut (Arief, 2011) MySQL (My Structure Query Language) adalah “salah satu jenis database server yang sangat terkenal dan banyak digunakan untuk membangun aplikasi web yang menggunakan database sebagai sumber dan pengelolaan datanya”. Mysql bersifat open source dan menggunakan SQL (*Structured Query Language*). MySQL biasa dijalankan diberbagai *platform* misalnya windows Linux, dan lain sebagainya.

Menurut (Abdul Kadir, 2008) mengatakan bahwa, “MySQL (baca: mai-se-kyu-el) merupakan sebuah *software* yang tergolong sebuah DBMS (*Database Management System*) yang bersifat *Open Source*. *Open Source* menyatakan bahwa suatu software ini dilengkapi dengan *suatu source code executable*-nya atau sebuah kode yang dapat dijalankan dengan secara langsung yang terdapat dalam system operasi, dan bisa diperoleh dengan cara *men-download* (mengunduh) di internet secara gratis.”

Menurut (Abdul Kadir, 2008) mengatakan bahwa, “Sebagai *software* DBMS, MySQL memiliki sejumlah fitur seperti yang dijelaskan dibawah ini.

1. Multiplatform

MySQL merupakan *platform* yang tersedia pada beberapa *platform* (Windows, Linux, Unix dan lain-lain)

2. Andal, cepat, dan mudah digunakan

MySQL tergolong sebagai sebuah *database server* (server yang melayani permintaan terhadap database) yang sangat andal, dan dapat menangani database yang besar dengan kecepatan sangat tinggi, mendukung banyak sekali fungsi untuk mengakses sebuah database, dan sekaligus mudah untuk digunakan. Berbagai *tool* pendukung juga tersedia (walaupun dibuat oleh pihak lain). Perlu diketahui, MySQL dapat menangani sebuah *table* yang berukuran dalam terabyte (1 terabyte = 1024 gigabyte). Namun, ukuran yang sesungguhnya sangat bergantung pada Batasan *system* operasi. Sebagai contoh, pada *system* Solaris 9/10, Batasan ukuran file sebesar 16 terabyte.

3. Jaminan keamanan akses

MySQL mendukung pengamanan database dengan berbagai kriteria pengaksesan. Sebagai gambaran, dimungkinkan untuk mengatur user tertentu agar bisa mengakses data yang bersifat rahasia (misalnya gaji pegawai), sedangkan user lain tidak boleh. MySQL juga mendukung konektivitas ke berbagai *software*. Sebagai contoh, dengan menggunakan ODBC (*Open Database Connectivity*), data-base yang ditangani MySQL dapat diakses melalui program yang dibuat dengan Visual Basic. MySQL juga mendukung suatu program klien yang berbasis Java untuk melakukan

komunikasi dengan database MySQL melalui JDBC (*Java Database Connectivity*). MySQL juga bisa diakses melalui aplikasi berbasis Web; misalnya dengan menggunakan PHP.

#### 4. Dukungan SQL

Seperti tersirat dalam Namanya, MySQL mendukung perintah SQL (*Structured Query Language*). Sebagaimana diketahui, SQL merupakan standar dalam pengaksesan database relasional. Pengetahuan akan SQL akan memudahkan siapa pun untuk menggunakan MySQL.”

### 2.5 Data Mining

Menurut (Larose, 2006) “*data mining* adalah proses menemukan sesuatu yang bermakna oleh suatu korelasi baru, pola dan juga tren yang terdapat dengan cara memilah-milah data yang berukuran besar, dimana data tersebut disimpan dalam *repository*, menggunakan teknologi sosialisasi pola serta statistik dan teknik matematika.”

Pengertian penambangan data atau “*data mining*” merupakan analisis pengamatan *database* dalam menemukan hubungan yang tidak terduga serta juga untuk meringkas data dengan cara atau sebuah metode baru yang dapat dimengerti yang berguna pada pemilik data.

Penggalian data juga dapat diartikan sebagai suatu proses ekstraksi sebuah informasi yang berguna serta juga potensial dari sekumpulan data yang secara *implisit* terdapat didalam suatu kumpulan data.



Menurut (Larose, 2005) “ada beberapa istilah lain yang mempunyai makna yang sama dengan apa itu *data mining*, yaitu *Knowledge Discovery in Databases* (KDD), *Knowledge Extraction* atau ekstraksi pengetahuan, *Data/Pattern Analysis*, *Business Intelligence* dan *Data Archeology* serta *Data Dredging*.”

Kemampuan penggalian data dalam mencari informasi bisnis yang berharga dari basis data yang besar, dapat dianalogikan dengan penambangan logam mulia dari lahan sumbernya.

Oleh karena itulah, teknologi ini biasa digunakan untuk:

- **Prediksi trend dan sifat-sifat bisnis;** Dimana *data mining* disini mengotomatisasi proses pencarian informasi memprediksi di dalam basis data yang besar.
- **Penemuan pola-pola;** Disini *data mining* menyapu basis data, lalu mengidentifikasi pola-pola yang sebelumnya tersembunyi di dalam sapuan tersebut.
- **Membuat keputusan penting;** *Data mining* berguna untuk membuat sebuah keputusan yang penting atau kritis, terutama didalam strategi.

### 2.5.1 Fungsi Data Mining

*Data mining* memiliki banyak sekali fungsi, Untuk fungsi utamanya sendiri yaitu ada dua; Yaitu fungsi *descriptive* dan fungsi *predictive*. Berikut Fungsi-fungsi Data mining:

#### 1. *Descriptive*

fungsi deskripsi dalam data mining adalah sebuah fungsi untuk memahami lebih jauh tentang data yang diamati. Dengan melakukan

sebuah proses diharap bisa mengetahui perilaku dari sebuah data tersebut. Data tersebut itulah yang nantinya dapat digunakan untuk mengetahui karakteristik dari data yang dimaksud.

Dengan menggunakan Fungsi *descriptive Data mining*, Maka nantinya bisa menemukan pola tertentu yang tersembunyi dalam sebuah data. Dengan kata lain jika pola yang berulang dan bernilai itulah karakteristik sebuah data bisa diketahui.

## 2. *Predictive*

Fungsi prediksi merupakan sebuah fungsi bagaimana sebuah proses nantinya akan menemukan pola tertentu dari suatu data. Pola-pola tersebut dapat diketahui dari berbagai *variabel-variabel* yang ada pada data.

## 3. *Multidimensional concept description*

Karakterisasi dan diskriminasi, Atau berfungsi untuk Menggeneralisasikan, meringkas, dan membedakan karakteristik data, dll.

## 4. *Frequent patterns, association, correlation*

## 5. *Classification and prediction*

Membangun model (fungsi) yang menggambarkan dan membedakan kelas atau konsep untuk prediksi masa depan. Misalnya, Mengklasifikasikan negara berdasarkan (iklim), atau mengklasifikasikan mobil berdasarkan (jarak tempuh gas)

## 6. *Cluster analysis*

Membuat data grup untuk membentuk kelas baru. Misalnya, Memaksimalkan kesamaan intra-kelas & meminimalkan kesamaan antar kelas

#### 7. *Outlier analysis*

Objek data yang tidak sesuai dengan perilaku umum dari data, Berguna dalam deteksi penipuan, analisis peristiwa langka.

#### 8. *Trend and evolution analysis*

Trend dan penyimpangan: misalnya Analisis regresi atau *Mining*  
Penambangan pola berurutan: misalnya, Kamera digital, atau Analisis periodisitas dan Analisis berbasis kesamaan.

#### 9. *Other pattern-directed or statistical analyses*

### 2.5.2 Algoritma Apriori (*Market Basket Analysis*)

*Association rule mining* adalah suatu prosedur untuk mencari hubungan antar *item* dalam suatu *data set* yang ditentukan. (Han, Kamber, 2001). Analisis asosiasi memberikan informasi dalam bentuk hubungan “*if-then*” atau “jika-maka”. Informasi ini dihitung dari data yang sifatnya probabilistik. Istilah *antecedent* untuk mewakili bagian “jika”. Istilah *consequent* untuk mewakili bagian “maka”. Dalam analisis asosiasi, *antecedent* dan *consequent* adalah sekelompok item yang tidak punya hubungan secara bersama.

Dalam menentukan suatu *association rule*, terdapat suatu *interestingness measure* (ukuran ketertarikan) yang didapatkan dari hasil pengolahan data dengan perhitungan tertentu. Umumnya ada dua ukuran, yaitu:

1. *Support* (nilai penunjang) adalah rasio antara jumlah transaksi yang memuat *antecedent* dan *consequent* dengan jumlah transaksi. suatu ukuran yang menunjukkan seberapa besar tingkat dominasi suatu *item/itemset* dari keseluruhan transaksi. Ukuran ini menentukan apakah suatu *item/itemset* layak untuk dicari *confidence*-nya (misal, dari keseluruhan transaksi yang ada, seberapa besar tingkat dominasi yang menunjukkan bahwa *item* A dan B dibeli bersamaan).

$$S = \frac{\Sigma (Ta + Tc)}{\Sigma(T)}$$

Keterangan :

$S$  = *Support*

$\Sigma (Ta + Tc)$  = Jumlah transaksi yang mengandung *antecedent* dan *consequent*

$\Sigma(T)$  = Jumlah transaksi

2. *Confidence* (nilai kepastian) adalah rasio antara jumlah transaksi yang meliputi semua item dalam *antecedent* dan *consequent* dengan jumlah transaksi yang meliputi semua item dalam antecedent. suatu ukuran yang menunjukkan hubungan antar 2 *item* secara *conditional* (misal, seberapa sering *item* B dibeli **jika** orang membeli *item* A).

$$C = \frac{\Sigma (Ta+Tc)}{\Sigma(Ta)}$$

Keterangan :

$C$  = Confidence

$\Sigma (Ta + Tc)$  = Jumlah transaksi yang mengandung *antecedent* dan *consequent*

$\Sigma(Ta)$  = Jumlah transaksi yang mengandung *antecedent*

“Apriori is an influential algorithm for mining frequent itemsets for Boolean association rules.” (Han, Kamber, 2001).

Apriori adalah suatu algoritma yang sudah sangat dikenal dalam melakukan pencarian *frequent itemset* dengan *association rule*. Sesuai dengan namanya, algoritma ini menggunakan *knowledge* mengenai *frequent itemset* yang telah diketahui sebelumnya, untuk memproses informasi selanjutnya. Algoritma inilah yang biasanya dipakai dalam proses *data mining* untuk *market basket analysis*.

Algoritma apriori banyak digunakan pada data transaksi atau biasa disebut market basket, misalnya sebuah swalayan memiliki market basket, dengan adanya algoritma apriori, pemilik swalayan dapat mengetahui pola pembelian seorang konsumen, jika seorang konsumen membeli item A , B, punya kemungkinan 50% dia akan membeli item C, pola ini sangat signifikan dengan adanya data transaksi selama ini.

Association rule umumnya mengambil bentuk IF-THEN yang menggabungkan beberapa items menjadi satu, misalnya:

*IF A and B THEN C*

Secara teori, beberapa hal yang digunakan untuk mengukur apakah sekumpulan items (an item set) sering muncul bersamaan atau tidak, adalah *support of an item set*, *confidence of an association rule*, dan beberapa *rule selection methods*.

- *Support of an item set* adalah *persentase* dari semua transaksi yang terjadi yang mengandung item set tersebut. Adapun rumus yang digunakan untuk menghitung adalah sebagai berikut:

$$\text{Support}(S) = (|U|/|T|) * 100\%$$

U adalah himpunan transaksi yang mengandung itemset S

|U| adalah jumlah *element* dalam U

|T| adalah jumlah transaksi yang terjadi

- *Confidence of an association rule* adalah perbandingan antara nilai *support* dari himpunan items yang terdapat di dalam *rule* dan nilai *support* dari himpunan items yang mendahuluinya. Sebagai contoh: apabila terdapat *rule* “Jika A dan B maka C”, *confidence of association rule* tersebut adalah:

$$\text{Confidence}(R) = (\text{Support}\{A,B,C\}/\text{Support}\{A,B\}) * 100\%$$

Penting tidaknya suatu aturan asosiatif dapat diketahui dengan dua parameter, *support* (nilai penunjang) yaitu *persentase* kombinasi item tersebut dalam *database* dan *confidence* (nilai kepastian) yaitu kuatnya hubungan antar item dalam aturan asosiatif.

Aturan asosiatif biasanya dinyatakan dalam bentuk :  $\{roti, mentega\} \rightarrow \{susu\}$  (*support* = 40%, *confidence* = 50%)

Yang artinya : “Seorang konsumen yang membeli roti dan mentega punya kemungkinan 50% untuk juga membeli susu. Aturan ini cukup signifikan karena mewakili 40% dari catatan transaksi selama ini.”

Analisis asosiasi didefinisikan suatu proses untuk menemukan semua aturan asosiatif yang memenuhi syarat minimum untuk *support* (*minimum support*) dan syarat minimum untuk *confidence* (*minimum confidence*).

Tetapi di lain pihak Apriori memiliki kelemahan karena harus melakukan *scan database* setiap kali iterasi, sehingga waktu yang diperlukan bertambah dengan makin banyak iterasi. Masalah ini yang dipecahkan oleh algoritma-algoritma baru seperti FP-growth.

*Association rule* meliputi dua tahap: (Ulmer, David, 2002) :

1. Analisa pola frekuensi tinggi

Mencari kombinasi yang paling sering terjadi dari suatu *itemset*. Tahap ini mencari kombinasi item yang memenuhi syarat minimum dari nilai *support* dalam *database*. Nilai *support* sebuah item diperoleh dengan rumus berikut:

$$\text{Support (A)} = \frac{\text{Jumlah Transaksi mengandung A}}{\text{Total Transaksi}}$$

sedangkan nilai *support* dari 2 item diperoleh dari rumus berikut:

$$\text{Support (A} \cap \text{B)} = \frac{\text{Jumlah Transaksi mengandung A}}{\text{Total Transaksi}}$$

## Total Transaksi

## 2. Pembentukan aturan asosiatif

Mendefinisikan *Condition* dan *Result* (untuk *conditional association rule*).

Setelah semua pola frekuensi tinggi ditemukan, barulah dicari aturan asosiatif yang memenuhi syarat minimum untuk *confidence* dengan menghitung *confidence* aturan asosiatif A B Nilai *confidence* dari aturan A B diperoleh dari rumus berikut:

$$\text{Confidence} = P(B | A) = \frac{\text{Jumlah Transaksi mengandung A dan B}}{\text{Jumlah Transaksi mengandung A}}$$

Algoritma apriori memakai pendekatan *iterative (level-wise search)*, dimana *k-itemset* dipakai untuk menyelidiki *(k+1)-itemset*. Langkah-langkah dari algoritma ini adalah sebagai berikut: (Ulmer, David, 2002).

1. Set  $k=1$  (menunjuk pada *itemset* ke-1).
2. Hitung semua *k-itemset* (*itemset* yang mempunyai  $k$  item).
3. Hitung *support* dari semua calon *itemset* – *filter itemset* tersebut berdasarkan perhitungan *minimum support*.
4. Gabungkan semua *k-sized itemset* untuk menghasilkan calon *itemset*  $k+1$ .
5. Set  $k=k+1$ .



6. Ulangi langkah 3-5 sampai tidak ada *itemset* yang lebih besar yang dapat dibentuk.
7. Buat *final set* dari *itemset* dengan menciptakan suatu *union* dari semua *k-itemset*.

## 2.6 Diagram UML

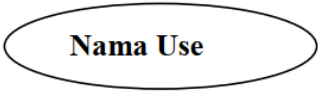
Menurut (Rosa dan Shalahuddin, 2014), UML adalah sebuah standarisasi bahasa pemodelan untuk pembangunan perangkat lunak yang dibangun dengan menggunakan teknik pemrograman berorientasi objek.



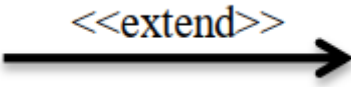

### 2.6.1 Use Case diagram

*Use Case Diagram* Menurut (Rosa dan Shalahuddin, 2014:155), “*Use case* atau diagram *use case* merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat”. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat.

Berikut adalah simbol-simbol yang ada pada diagram *use case* (Rosa dan Shalahuddin, 2014:156) :

**TABEL: 2.1 Simbol Usecase Diagram**

Simbol	Deskripsi
<p><i>Use case</i></p> 	<p>Fungsionalitas yang di sediakan sistem sebagai unit-unit yang saling bertukar peran antara unit dana aktor, biasanya dinyatakan dengan menggunakan kata kerja di awal <i>frase</i> nama <i>usecase</i></p>
Aktor / <i>Actor</i>	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi

	<p>yang akan dibuat diluar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang: biasanya dinyatakan menggunakan kata benda diawal <i>frase</i> nama aktor</p>
<p>Asosiasi / <i>Assosiation</i></p> 	<p>Komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor</p>
<p>Ekstensi / <i>extend</i></p> 	<p>Relasi <i>use case</i> tambahan kesebuah <i>use case</i> dinamakan <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan itu; mirip dengan prinsip <i>intherince</i> pada pemrograman berorientasi objek; biasanya <i>use case</i> tambahan memiliki nama depan yang sama dengan <i>use case</i> yang ditambahkan.</p>
<p>Generaliasi / <i>Generalization</i></p> 	<p>Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya.</p>






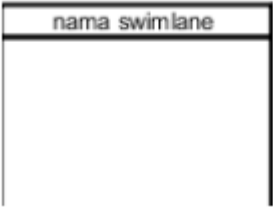
## 2.6.2 Activity Diagram

*Activity* Diagram Menurut (Rosa dan Shalahuddin, 2014:161) “Diagram aktivitas atau *activity* diagram menggambarkan *workflow* (aliran kerja) atau

aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak”. *Activity Diagram* menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem.

Berikut adalah simbol-simbol yang ada pada diagram aktivitas (Rosa dan Shalahuddin, 2014:162):

**TABEL: 2.2 Simbol Activity Diagram**

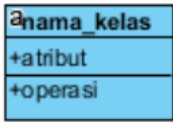



Simbol	Deskripsi
Status Awal 	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal
Aktivitas 	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja
Percabangan / decision 	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu
Penggabungan / Join 	Asosiasi penggabungan dimana lebih dari satu aktivitas di gabungkan menjadi satu
Status Akhir 	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir
Swimlane 	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi



### 2.6.3 Class Diagram

*Class Diagram* Menurut (Rosa dan Shalahuddin, 2014:146) menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi.

Berikut adalah simbol-simbol yang ada pada diagram Kelas (Rosa dan Shalahuddin, 2014:146) :

**TABEL: 2.3 Simbol Class Diagram**

Simbol	Deskripsi
Kelas 	Kelas pada struktur sistem
Antarmuka / <i>Interface</i>  Nama_interface	Sama dengan konsep <i>interface</i> dalam pemrograman berorientasi objek
Asosiasi / <i>Assosiation</i> 	Relasi antar kelas dengan makna umum, asosisai biasanya juga disertasi dengan <i>multiplicity</i>
Generasilasi 	Relasi antar kelas dengan makna <i>generalisasi-spesialisasi</i> (umum khusus)

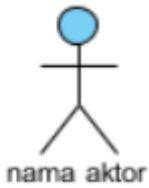
Kepbergantungan / Dependency 	Relasi antar kelas dengan makna kebergantungan antarkelas
Agregasi / Aggregation 	Relasi antar kelas dengan makna semua bagian ( <i>Whole-part</i> )

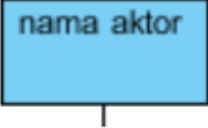

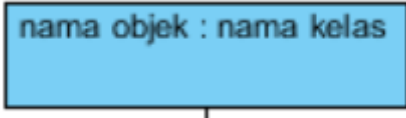
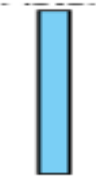

#### 2.6.4 Sequence Diagram



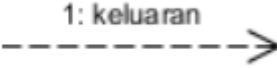
Diagram sekuen menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek. Untuk menggambarkan diagram sekuen maka harus diketahui objek objek yang terlibat dalam sebuah *use case* beserta metode-metode yang dimiliki kelas yang diinstansiasi menjadi objek itu.

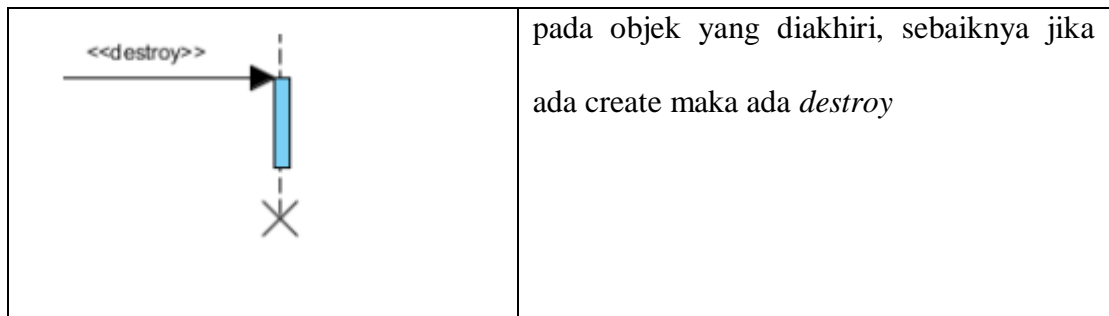
Berikut adalah simbol-simbol yang ada pada diagram sekuen (Rosa dan Shalahuddin,2014:165) :

**TABEL: 2.4 Simbol Sequence Diagram**

Simbol	Deskripsi
Aktor / Actor 	Orang, proses atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat siluar sistem informasi itu sendiri, jadi walaupun simbol dari aktor adalah orang, tapi aktor belum tentu

<p>Atau</p> 	<p>merupakan orang, biasanya dinyatakan menggunakan kata benda diawal frase nama aktor</p>
<p>Garis hidup / <i>Lifeline</i></p> 	<p>Menyatakan kehidupan suatu objek</p>
<p>Objek</p> 	<p>Menyatakan objek yang berinteraksi pesan</p>
<p>Waktu aktif</p> 	<p>Menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang di lakukan di dalamnya.</p>
<p>Pesan tipe <i>create</i></p>  <p>&lt;&lt;create&gt;&gt;</p>	<p>Menyatakan suatu objek membuat objek yang lain, arah panah mengarah pada objek yaang dibuat</p>

<p>Pesan tipe <i>Call</i></p> 	<p>Menyatakan suatu objek memanggil operasi / metode yang ada pada objek lain atau dirinya sendiri.</p> <p>Arah panah mengarah pada objek yang memiliki operasi / metode, karena ini memanggil operasi / metode maka operasi / metode yang dipanggil harus ada pada diagram kelas sesuai dengan kelas objek yang berinteraksi</p>
<p>Pesan tipe <i>send</i></p> 	<p>Menyatakan bahwa suatu objek mengirimkan data / masukan / informasi ke objek lainnya, arah panah mengarah pada objek yang dikirim</p>
<p>Pesan tipe <i>return</i></p> 	<p>Menyatakan suatu objek yang telah menjalankan suatu operasi atau metode menghasilkan suatu kembalian ke objek tertentu, arah panah mengarah pada objek yang menerima kembalian</p>
<p>Pesan tipe <i>destroy</i></p>	<p>Menyatakan suatu objek mengakhiri hidup objek yang lain, arah panah mengarah</p>





## **BAB III**

### **ANALISIS MASALAH DAN PERANCANGAN PROGRAM**

#### **3.1 Pengumpulan Data**

##### **3.1.1 Studi Observasi**

Berdasarkan hasil observasi yang di lakukan di Artpedia. Admin optimasi yang bertugas untuk menentukan produk unggulan dan membuat paket kombo hemat serta paket *bundlling* masih di lakukan secara manual dimana admin perlu *men-download* data penjualan satu-persatu di setiap *market place* dan menganalisa hasil data dengan bantuan microsoft excel saja. Dimana kegiatan tersebut perlu mengecek konfersi produk di lihat, di masukan keranjang dan pesanan di buat. Masalahnya setiap bulan memiliki pola pembelian yang berbeda sehingga perlu di lakukan analisa secara berkala.

##### **3.1.2 Wawancara**

Berdasarkan Hasil wawancara yang di lakukan kepada beberapa pegawai artpedia yang berkaitan dengan pembuatan analisa ini diantaranya kepada *owner* selaku pemilik artpediashop , *purchasing* sebagai pengadaan barang yang akan menyediakan barang yang di *request* di aplikasi dan admin optimasi yang akan menggunakan secara langsung aplikasi tersebut.

##### **3.1.3 Studi Dokumen**

Studi dokumen yang di lakukan diantaranya menyiapkan dokumen pendukung seperti data penjualan di setiap *market place* dan mempelajari jurnal atau referensi mengenai penerapan *alogirtma apriori* dalam menentukan *item set* dan produk unggulannya.

### 3.2 Metode Pengembangan Sistem

Sesuai dengan permasalahan yang dihadapi dan tujuan yang di harapkan maka penulis menggunakan metode prototype dalam pengembangan sistem ini dimana di buatkan sebuah aplikasi analisa keranjang belanja untuk memudahkan analisa trend penjualan di artpedia. Dengan demikian pengolahan data di artpedia dapat dilakukan secara mudah cepat dan akurat.

Dalam menganalisa sistem di Artpedia ini terdapat 3 tahapan sitem yaitu :

1. **Pengumpulan Kebutuhan** , kebutuhan disini adalah analisa kebutuhan perangkat keras , perangkat lunak dan *brainware* seret data-data yang di perlukan.
2. **Perancangan**, dalam tahap ini dilakukan perancangan sistem yaitu tampilan *user interface* , dan pengumpulan kebutuhan pengolahan data , penerapan *coding*.
3. **Evaluasi Prototype**, tahapan ini mengevaluasi dari progres yang dilakukan di setiap perancangannya, bila ada kekurangan yang perlu di perbaiki atau tambahan yang ingin di sempurnakan.

#### 3.2.1 Analisa Masalah

Analisa masalah merupakan kegiatan menganalisa masalah yang terjadi pada prosedur yang sedang berjalan.

1. Identifikasi Masalah

Berdasarkan dengan apa yang telah diteliti dan di amati oleh penulis di Artpedia, terdapat beberapa masalah diantaranya penentuan produk unggulan dan item set masih di lakukan masih dilakukan secara manual dengan rumus

excel sederhana saja, dan bagaimana membuat aplikasi yang dapat membantu pengolahan data yang lebih efisien dan akurat.

## 2. Kelemahan prosedur terdahulu

Berdasarkan analisis prosedur terdahulu di artpedia terdapat kelemahan diantaranya dalam pengolahan data penjualan admin optimasi dan *purchasing* melakukan pengolahan data masing-masing sehingga memerlukan waktu yang lama, dan memiliki standarisasi masing-masing.

## 3. Evaluasi prosedur terdahulu

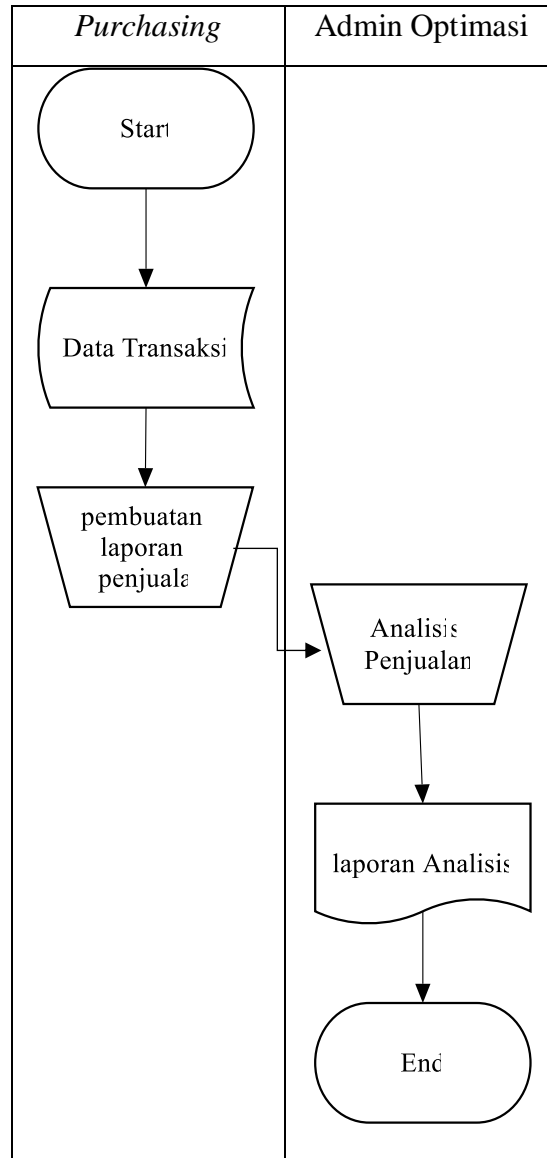
Berdasarkan analisis prosedur terdahulu dan menjabarkan permasalahan pada prosedur terdahulu maka dapat ditarik kesimpulan bahwa untuk memperbaiki sistem kerja perlu dilakukan evaluasi pada prosedur terdahulu. Adapun evaluasi prosedur terdahulu adalah diperlukan suatu perangkat lunak untuk menangani sistem pengolahan data penjualan yang cepat, akurat dan memberikan informasi item set serta produk unggulan dari penjualan perbulan.

### 3.2.2 Analisa sitem yang sedang berjalan

Dari hasil pengamatan penulis berikut adalah prosedur yang berjalan pada saat ini:

1. Admin optimasi mengolah data penjualan *online (market place)* setiap bulan dan menentukan produk unggulannya.
2. Admin *purchasing* mengolah data penjualan *Market place* setiap bulan, data yang di ambil *purcharing* adalah gabungan penjualan penjualan toko (*offline*) dan penjualan *online (market place)* yaitu *ipos*. Lalu menentukan produk *booster* yang perlu di kuatkan stoknya.

3. Admin *Purchasing* melihat data penjualan online dari admin optimasi
4. Admin *Purchasing* menentukan produk secara manual.



GAMBAR 3.1 *Flowchart* sistem yang sedang berjalan

### 3.2.3 Sistem yang di sarankan

Adanya masalah diatas membuat penulis menyarankan kepada perusahaan untuk penerapan analisa keranjang belanja sehingga dibuatkan aplikasi analisa keranjang belanja agar memudahkan pekerjaan karyawan menjadi lebih cepat,

sebelum membuat sistem informasi terlebih dahulu membuat analisa untuk memudahkan pembuatan sistem aplikasi analisa tersebut. Analisis ini di bagi dua yaitu , Analisa *Non-Fungsional* dan Analisa Fungsional.

### 3.2.3.1 Analisis Non Fungsional

Adapun kebutuhan secara fisik yang dibutuhkan agar dapat menunjang kelancaran dalam penggunaan aplikasi analisa keranjang belanja. Ada 3 kebutuhan, yaitu :

#### 1. Kebutuhan *Hardware*

Hardware atau perangkat keras yang dibutuhkan untuk menunjang kelancaran dalam penggunaan sistem yaitu PC atau laptop.

Berikut minimal spesifikasi detail mengenai perangkat yang di butuhkan :

- 1)Komputer dengan prosesor pentium *dual core*
- 2)Memory Komputer minimal 4GB atau lebih
- 3)Resolusi VGA minimum 800 x 600
- 4)Modem / Perangkat Internet

#### 2. Kebutuhan *Software*

*Software* atau perangkat lunak yang dibutuhkan untuk menunjang kelancaran sistem , antara lain :

- 1)Windows 7, windows 8/8.1 , Windows 10.
- 2)Browser
- 3)Web Server/ XAMPP

#### 3. Kebutuhan *Brainware*

*Brainware* atau pengguna dalam sistem ini hanya 1 level , yaitu :

- Admin , Tugas admin dalam sistem dapat memasukan data tarnsaksi, melakukan proses perhitungan dan melihat hasilnya (laporan).

### 3.2.3.2 Analisis Fungsional

Analisa fungsional ini menggambarkan kebutuhan yang di jalankan oleh sistem sesuai fungsi yang ada. Berikut adalah gambaran mengenai fungsi yang ada pada sistem :

#### 1. Data Transaksi

Transaksi penjualan setelah di gabungkan dengan sistem. Data penjualan yang sudah ada perlu di inputkan kembali kedalam sistem agar dapat di lakukan perhitungan. Berikut adalah langkah-langkah dalam memasukan data transaksi ke sistem :

- 1) Admin memasukan data penjualan dengan memilih menu data transaksi di sistem.
- 2) Admin memilih file data penjualan dan mengupload nya di sistem.
- 3) Sistem akan mengirimkan data penjualan yang di pilih oleh admin ke dalam database.
- 4) Admin dapat melihat tampilan data penjualan yg di inputkan di menu data transaksi.

#### 2. Perhitungan Algoritma Apriori

Dengan adanya sistem perhitungan tidak perlu dilakukan secara manual sehingga tidak perlu khawatir adanya kekeliruan penentuan produk yang di

lakukan oleh admin. Berikut Adalah langkah-langkah penilaian setelah diadakan sistem :

- 1) Admin melihat data transaksi yang telah di masukan di menu data transaksi.
- 2) Admin memilih menu proses apriori dan menentukan tanggal data transaksi penjualan.
- 3) Admin memasukan *minimal support* dan *Minimal confidence*
- 4) Sistem akan melakukan proses perhitungan dengan menggunakan metode apriori.
- 5) Admin memperoleh hasil perhitungan yang di lakukan oleh sistem.
- 6) Admin dapat melihat *rule* dan *itemset* yang terbentuk di menu hasil.

### **3.3 Perancangan Sistem**

#### **3.3.1 Desain Sistem dengan Unified Modeling Language (UML)**

Perancangan yang dilakukan oleh penulis terhadap alur sistem yang di buat akan di implementasikan menggunakan UML yang terdiri dari use case diagram, activity diagram, sequence diagram dan class diagram.

##### **3.3.1.1 Use Case Diagram**

Pada *use case* diagram ini menjelaskan interaksi antara sistem dengan aktor (admin) yang terlibat dengan sistem.

## a. Definisi Aktor

**TABEL : 3.1 Tabel Definisi Aktor**

NO	Aktor	Deskripsi
1	Admin	Mengelola Laporan data penjualan dan menghitung apriori.

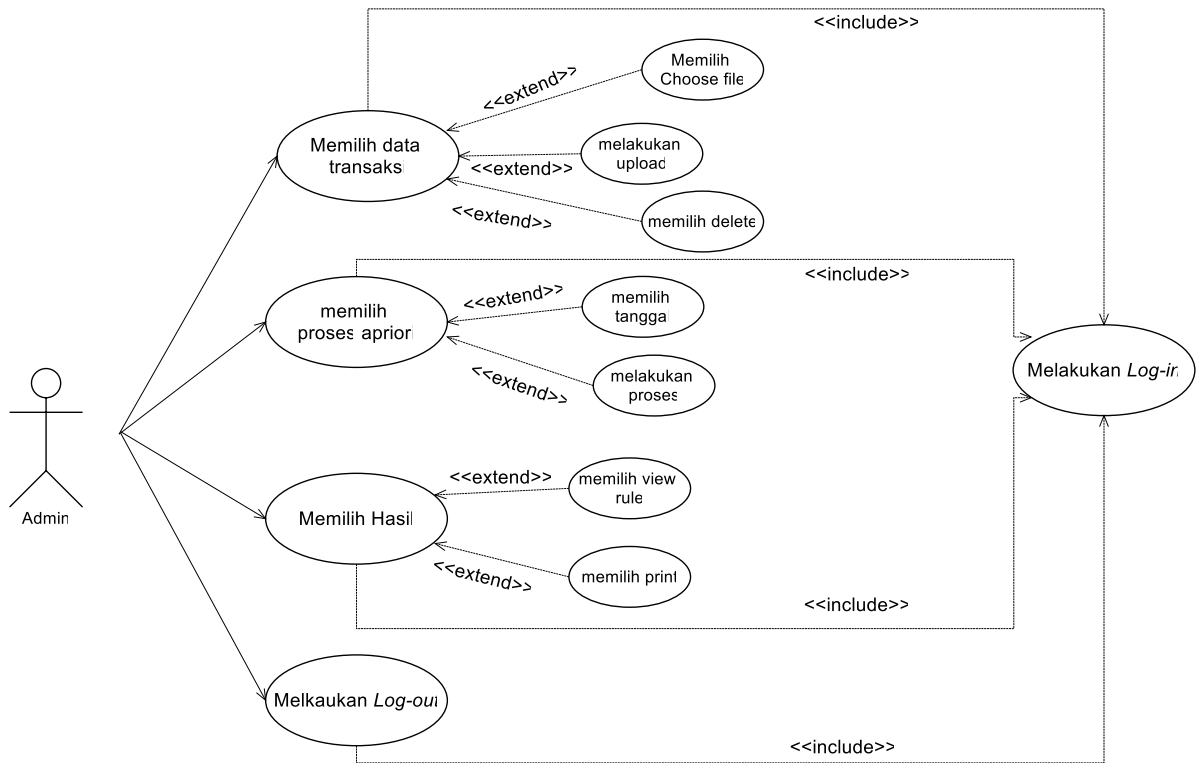
## b. Definisi Use Case

**TABEL : 3.2 Tabel Definisi Use Case**

NO	Aktor	Deskripsi
1.	Melakukan <i>Login</i>	Proses Masuknya <i>User</i> kedalam Sistem
2	Memilih Data Transaksi	Proses untuk memasukan data transaksi
3	Memilih Proses Apriori	Proses Menghitung data transaksi dengan metode apriori
4	Melihat Hasil	Proses melihat perolehan hasil perhitungan dan melakukan <i>print</i> data
5	Melakukan <i>Logout</i>	Proses Keluarnya <i>User</i> dari Sistem



c. Gambar Usecase Diagram



GAMBAR 3.2 Use Case Diagram

d.Skenario usecase

TABEL : 3.3 Skenario Use Case Login

Use Case		: Login	
Aktor		: Admin	
Deskripsi		: Aktor melakukan login untuk menggunakan sistem	
	Aktor		Sistem
1.	Menjalankan aplikasi		
		2.	Menampilkan halaman login
3.	Mengisi form login		
4.	Menekan tombol login		

		5.	Pengecekan data , jika gagal menampilkan pesan “user tidak ditemukan”, dan jika berhasil menampilkan menu berikutnya.
		6.	Menampilkan menu halaman utama

**TABEL : 3.4 Skenario *Usecase* Data Transaksi**

Use Case		: Data Transaksi	
Aktor		: Admin	
Deskripsi		: Aktor memilih Data Transaksi	
Aktor		Sistem	
1.	Menjalankan aplikasi		
		2.	Menampilkan halaman utama
3.	Menekan tombol data transaksi		
		4.	Menampilkan menu data transaksi
5.	Menekan tombol <i>Choose file</i>		
		6.	Menampilkan <i>windows explorer</i>
7.	Memilih File dan Menekan <i>open</i>		
		8	Menampilkan Data terpilih
8.	Menekan tombol <i>upload</i>		
		9.	Menyimpan data
		10.	Data berhasil disimpan
11.	<i>delete</i>		
		12.	Menampilan jumlah data 0
		13.	Data Berhasil Dihapus

**TABEL : 3.5 Skenario Use Case Proses Apriori**

Use Case		: Proses Apriori	
Aktor		: Admin	
Deskripsi		: Aktor memilih Proses apriori	
Aktor		Sistem	
1.	Menjalankan aplikasi		
		2.	Menampilkan halaman utama
3.	Menekan tombol proses apriori		
		4	Menampilkan halaman proses apriori
5	Memilih tanggal awal dan akhir		
6	Menekan tombol <i>apply</i>		
7	Memasukan <i>Min Support</i> dan <i>confidence</i>		
8	Menekan tombol proses		
		9	Melakukan perhitungan data
		10	Menampilkan hasil perhitungan
		11	Selesai

**TABEL : 3.6 Skenario Use Case Hasil**

Use Case		: Hasil	
Aktor		: Admin	
Deskripsi		: Aktor memilih Hasil	
Aktor		Sistem	
1.	Menjalankan aplikasi		
		2.	Menampilkan halaman utama
3.	Menekan tombol data hasil		
		4	Menampilkan tabel hasil
5	Menekan tombol <i>view</i>		
		6	Menampilkan rumus perhitungan (kombinasi produk)
7	Menekan tombol <i>print</i>		
		8	Menampilkan halaman print

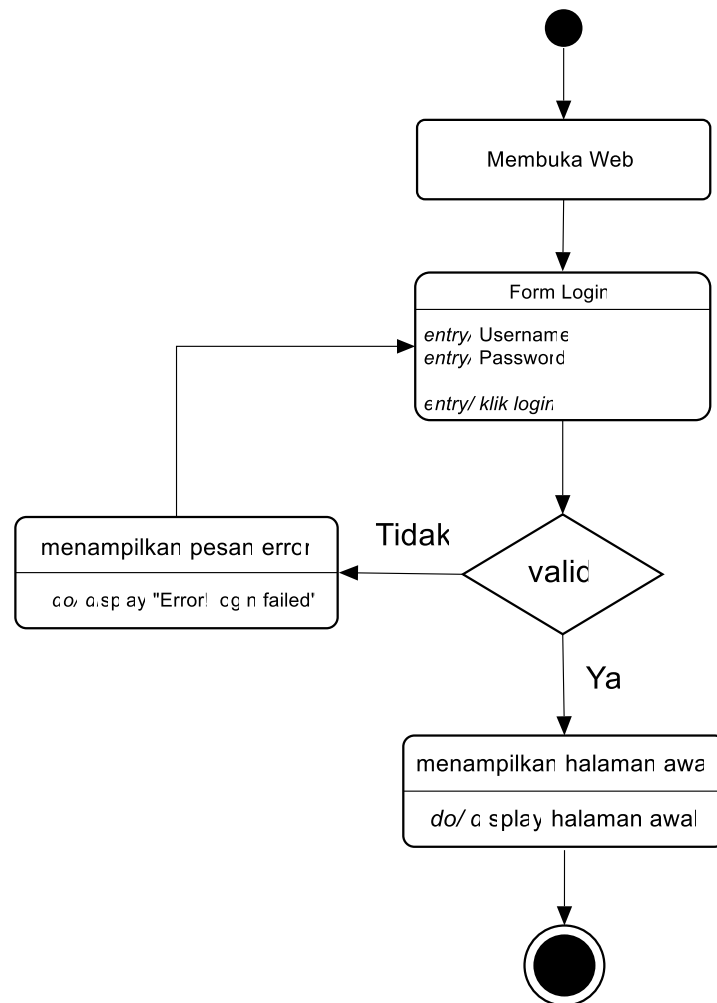
**TABEL : 3.7 Skenario Use Case Logout**

Use Case		: Logout	
Aktor		: Admin	
Deskripsi		: Aktor melakukan logout untuk keluar dari sistem	
Aktor		Sistem	
1.	Menjalankan aplikasi		
		2.	Menampilkan halaman
3	Menekan tombol <i>logout</i>		
		4	Keluar dari sistem , menampilkan halaman <i>login</i>
		5.	Selesai

### 3.3.1.2 Activity Diagram

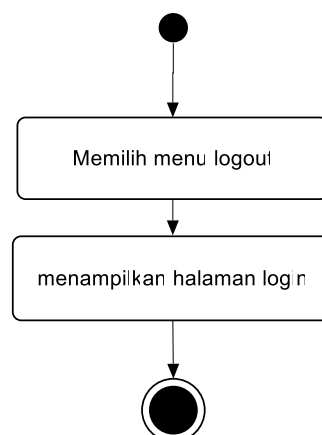
Pada Aktifitas diagram ini menjelaskan aliran kerja dari aktifitas yang dilakukan pada aplikasi.

1. *Activity Diagram* pada saat admin melakukan proses *login*



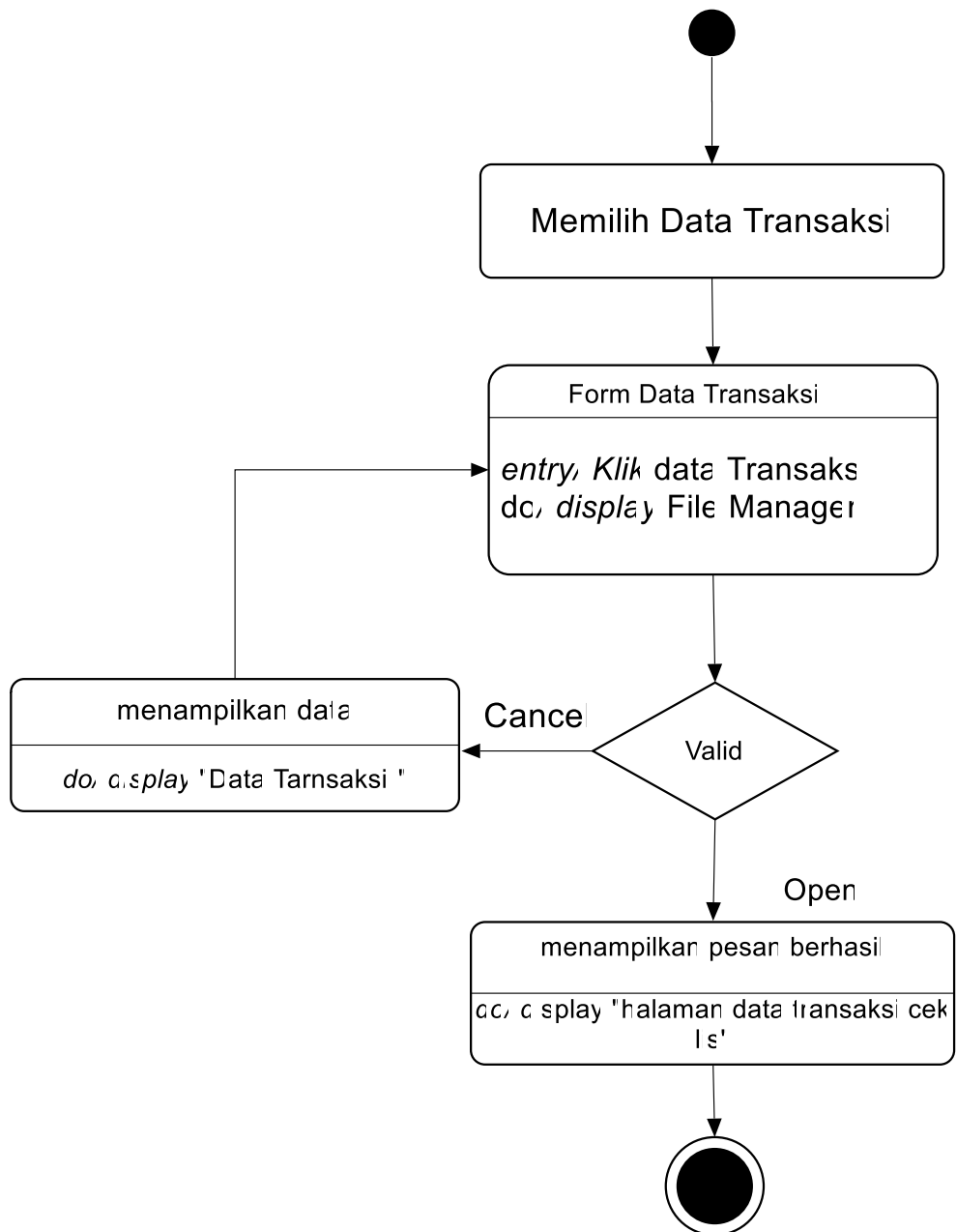
GAMBAR: 3.3 Activity Diagram Login

2. Activity Diagram pada saat admin melakukan proses *logout*



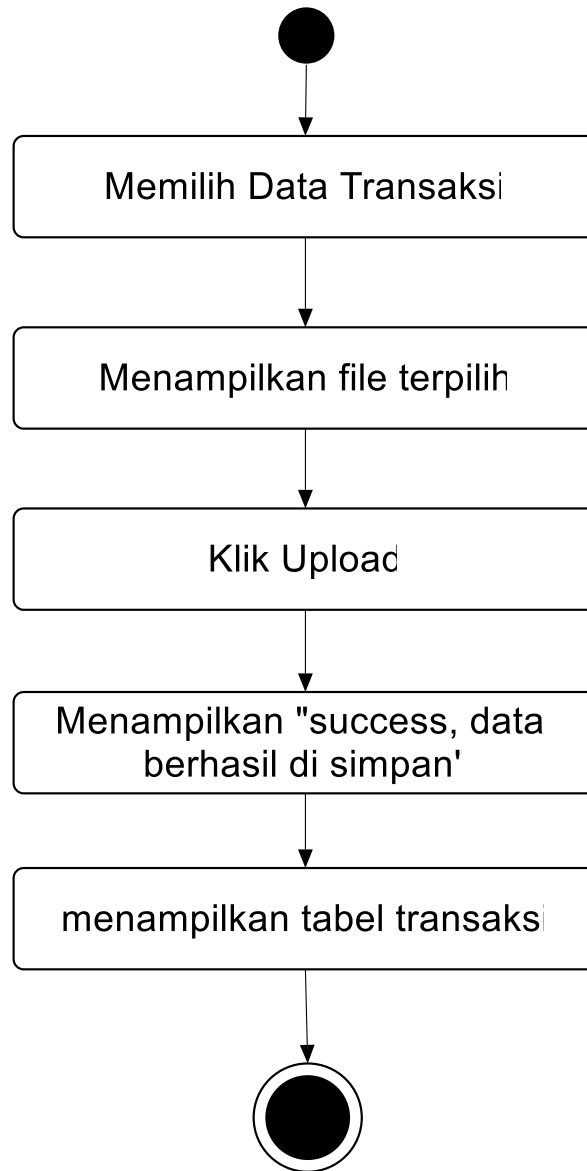
GAMBAR: 3.4 Activity diagram Logout

3. *Activity Diagram* pada saat admin memilih halaman data transaksi (*Choose File*)



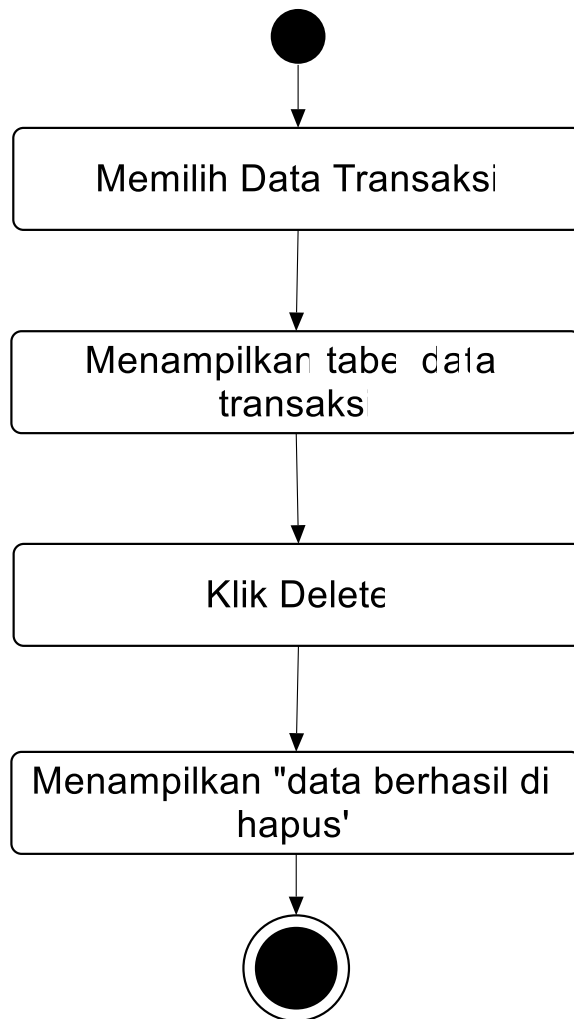
GAMBAR: 3.5 *Activity Diagram* Data Transaksi (*Choose File*)

4. *Activity Diagram* pada saat admin memilih halaman data transaksi (*upload*)



GAMBAR: 3.6 *Activity Diagram* Data Transaksi (*upload*)

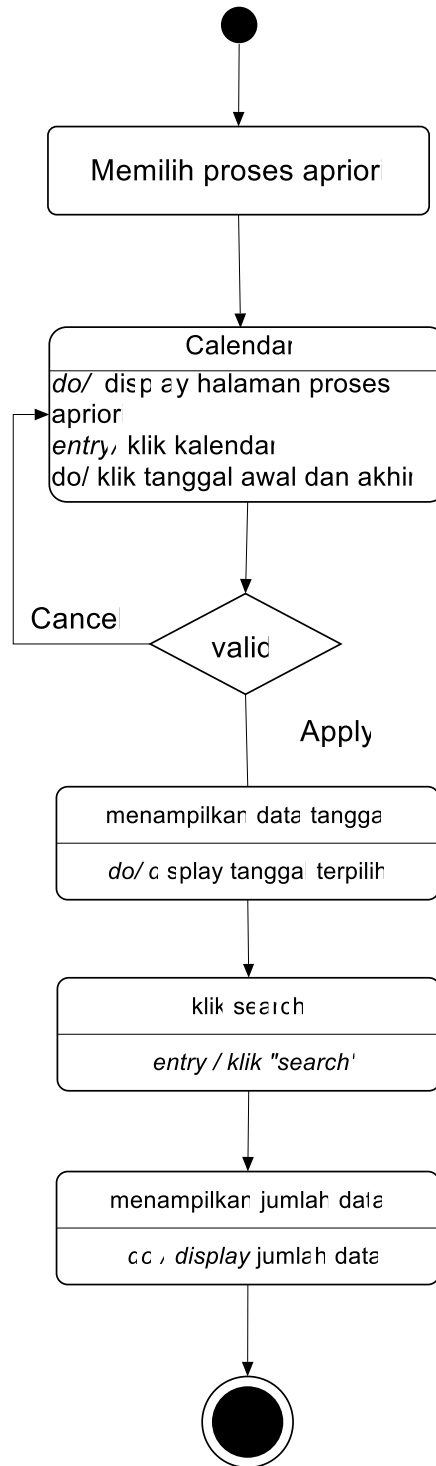
5. *Activity Diagram* pada saat admin memilih halaman data transaksi (*delete*)



GAMBAR: 3.7 *Activity Diagram* Data Transaksi (*Delete*)

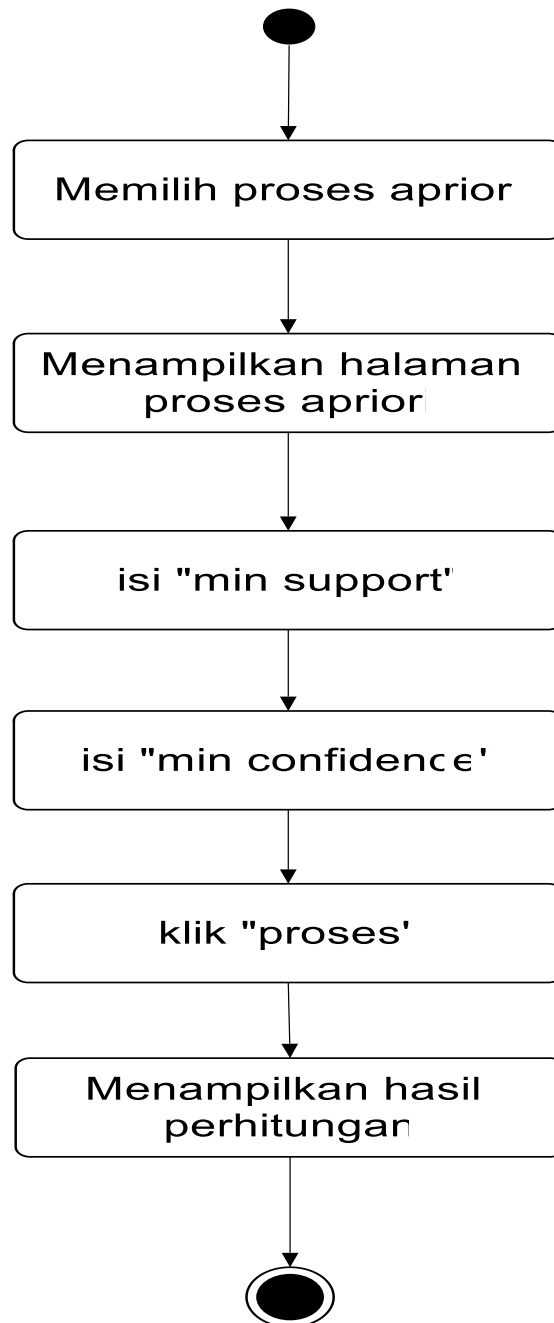


6. *Activity Diagram* pada saat admin memilih halaman Proses Apriori (Memilih Tanggal)



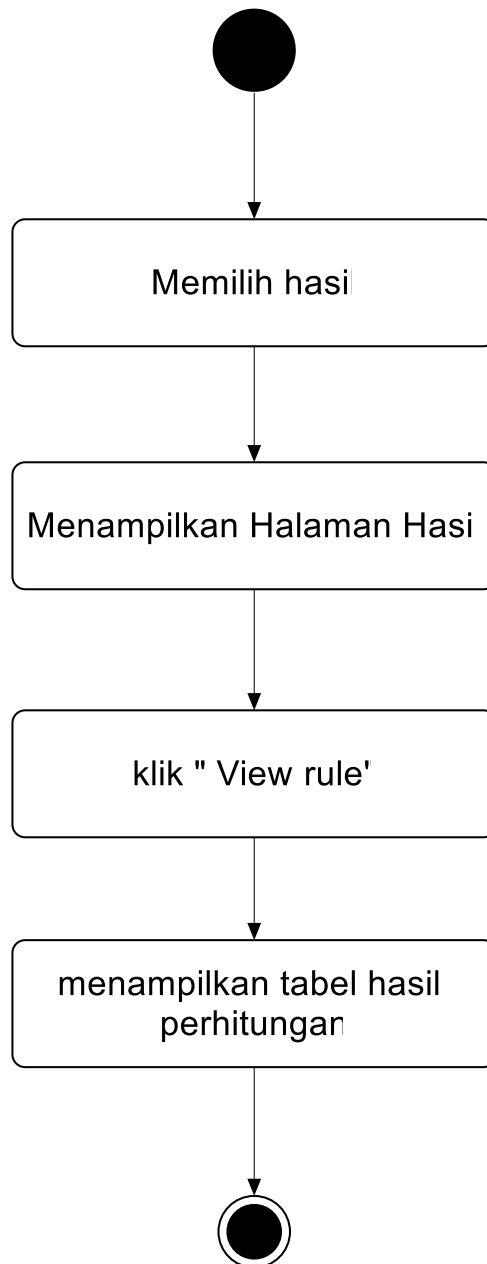
GAMBAR 3.8 *Activity Diagram* Proses Apriori (Memilih Tanggal)

7. *Activity Diagram* pada saat admin memilih Proses Apriori Mengisi *Min Support & min Confidende*

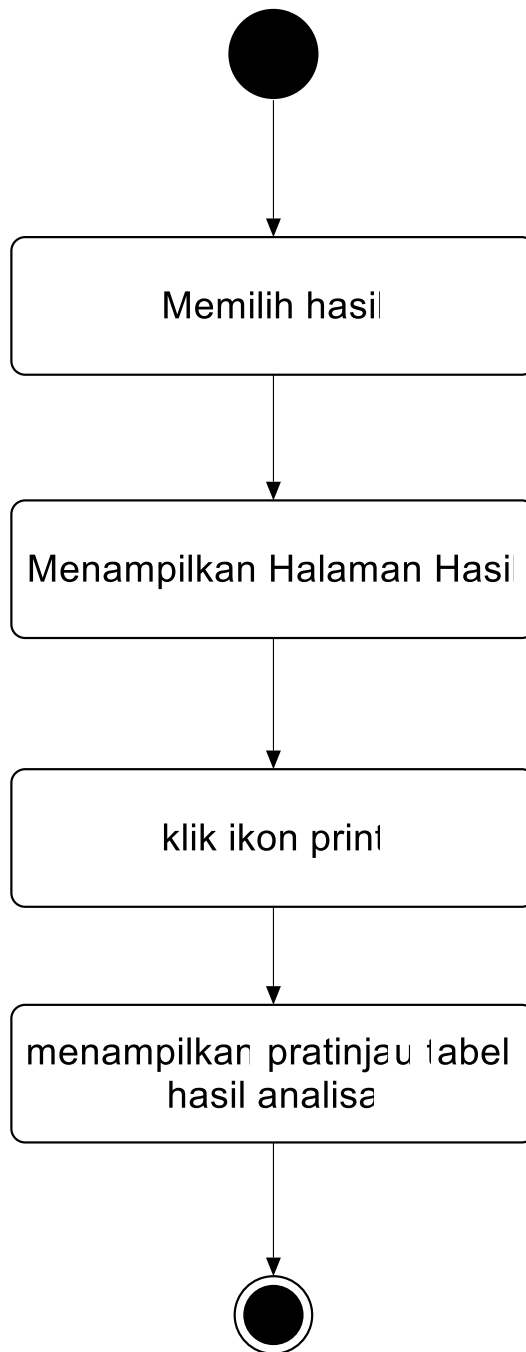


GAMBAR 3.9 *Activity Diagram* Proses Apriori (Mengisi *Min Support & min Confidende*)

8. Activity Diagram pada saat admin memilih halaman hasil (*view rule*)



GAMBAR 3.10 *Activity Diagram Hasil (View Rule)*

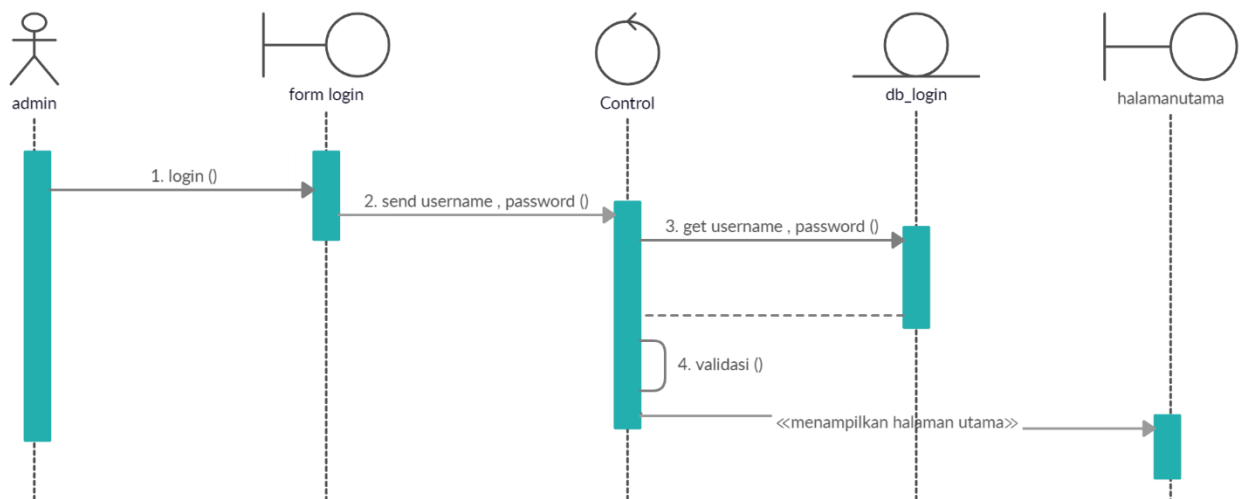
9. Activity Diagram pada saat admin memilih halaman hasil (*print*)

GAMBAR 3.11 Activity Diagram Hasil (Print)

### 3.3.1.3 Sequence Diagram

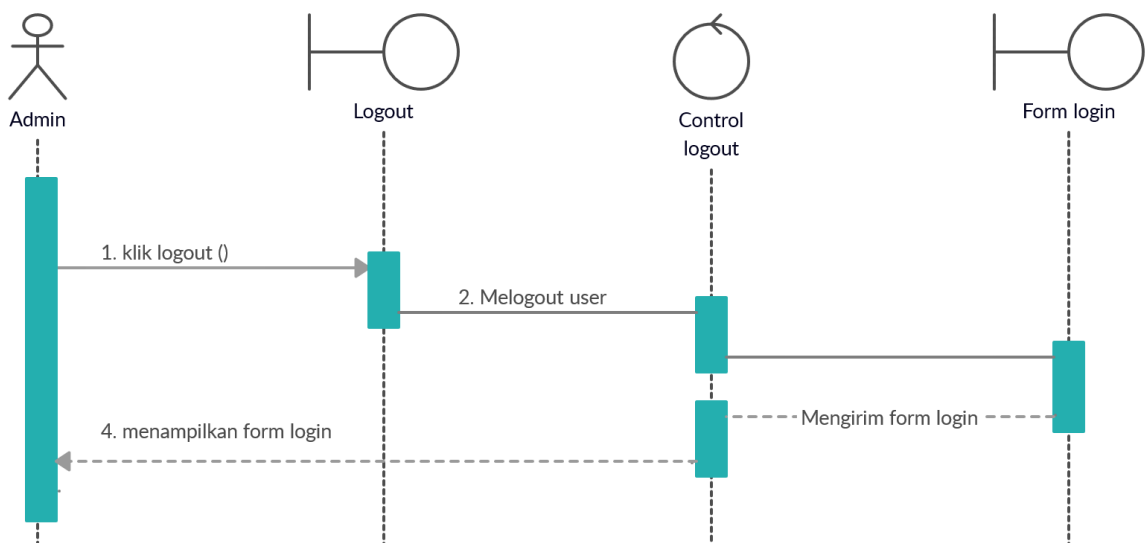
Pada Diagram ini menggambarkan waktu aktif/urutan waktu interaksi antara objek pada sistem.

#### 1. Sequence Diagram Log-in



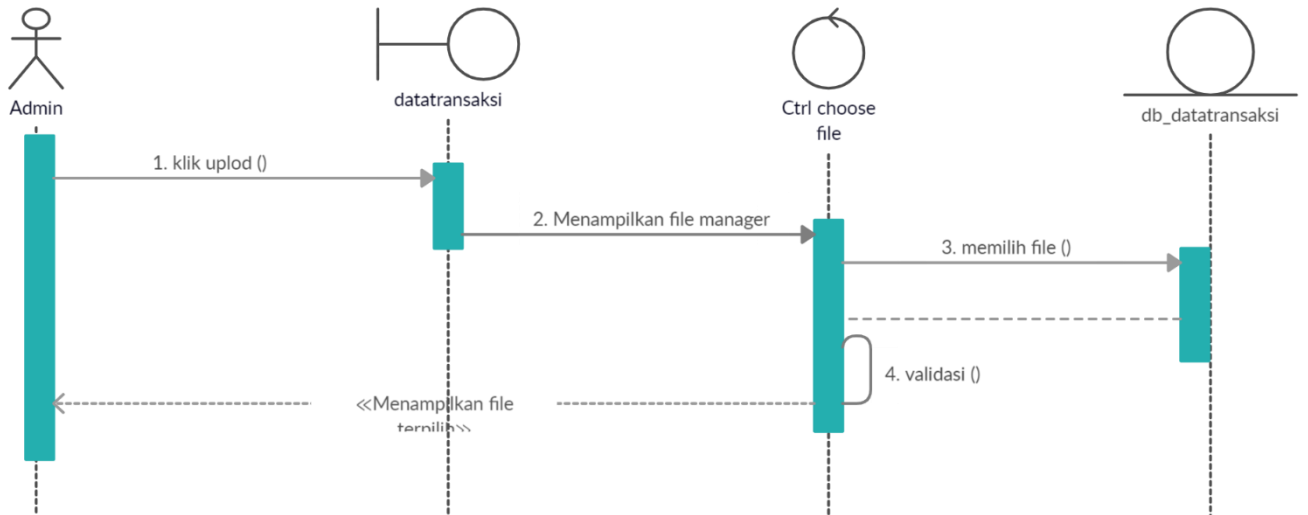
GAMBAR: 3.12 Sequence diagram login

#### 2. Sequence Diagram logout



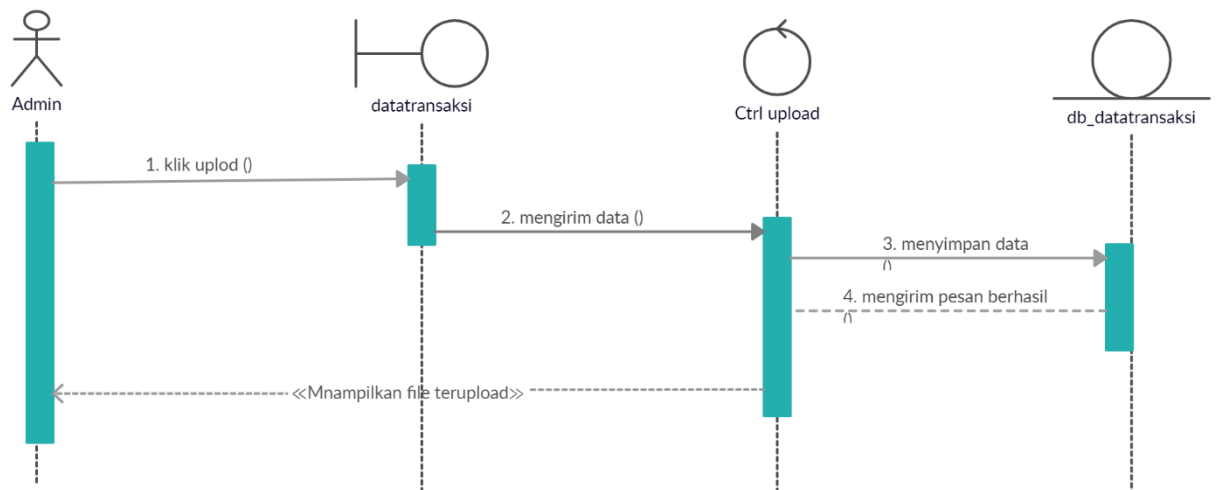
GAMBAR : 3.13 Sequence Diagram logout

### 3. Sequence diagram Data transaksi (*choose file*)



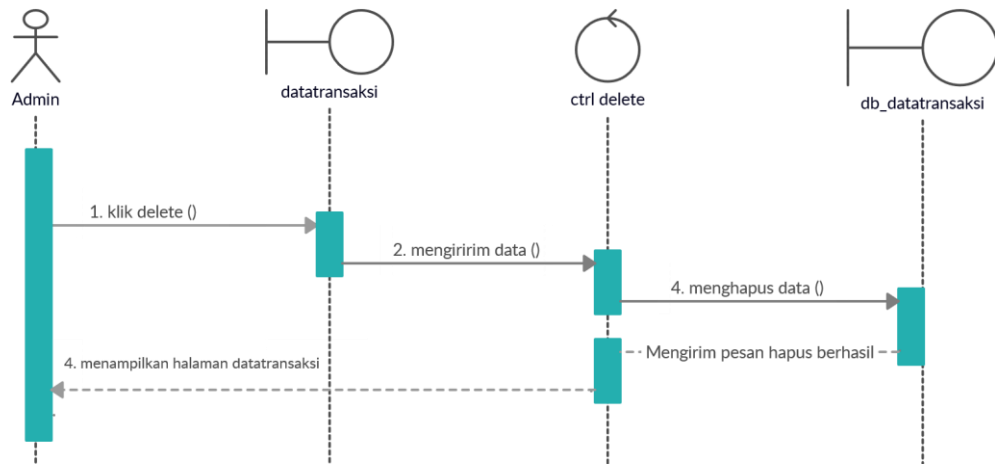
GAMBAR: 3.14 Sequence diagram memilih file

### 4. Sequence diagram data transaksi *upload* file



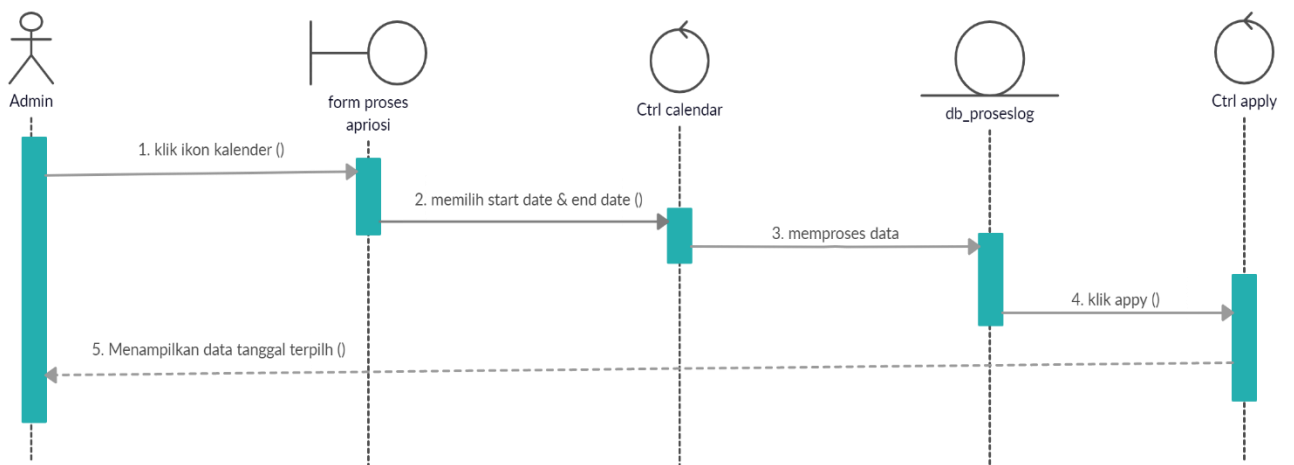
GAMBAR: 3.15 Sequence diagram upload

### 5. Sequence Diagram data transaksi delete



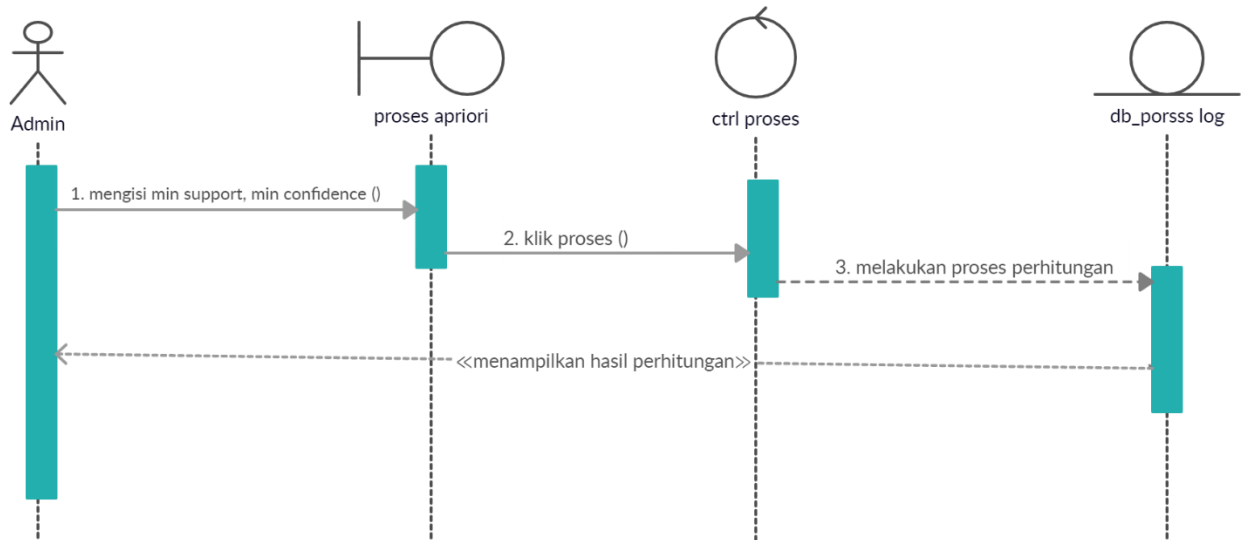
GAMBAR: 3.16 Sequence diagram delete

### 6. Sequence diagram proses apriori memilih tanggal



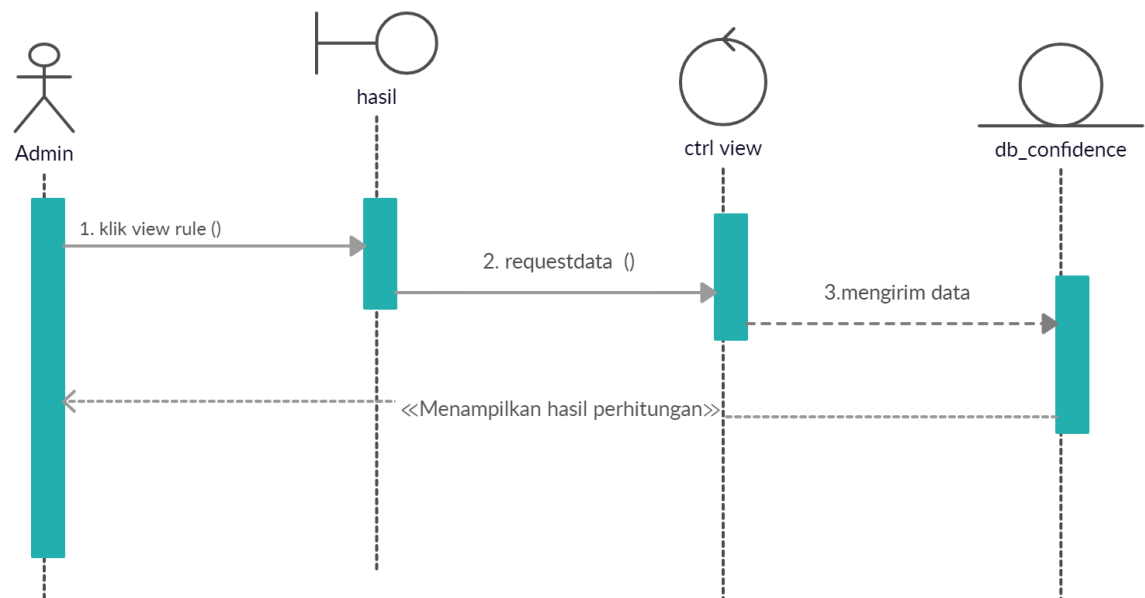
GAMBAR: 3.17 Sequence diagram memilih tanggal

### 7. Sequence diagram memilih proses



GAMBAR: 3.18 Sequence diagram memilih proses

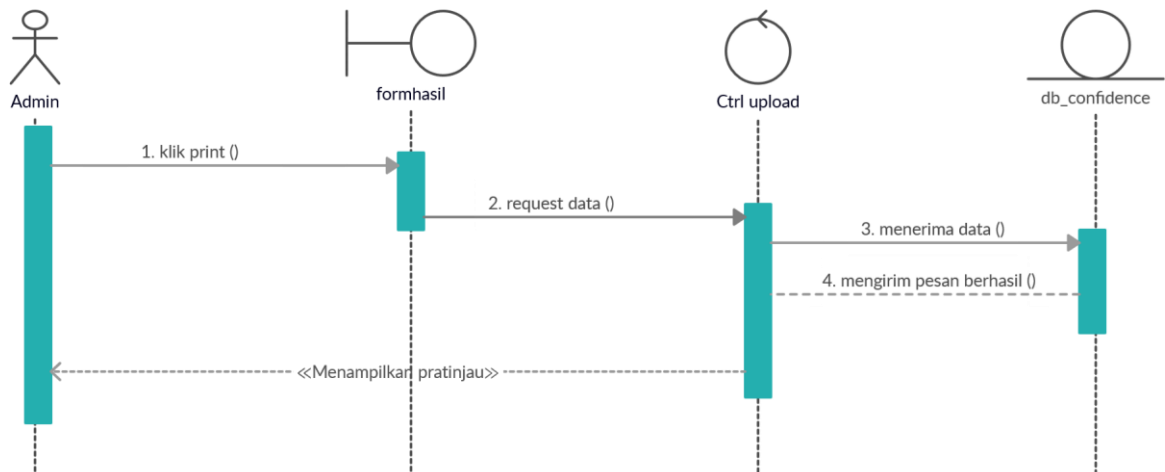
### 8. Sequence diagram view rule



GAMBAR: 3.19 Sequence diagram view rule



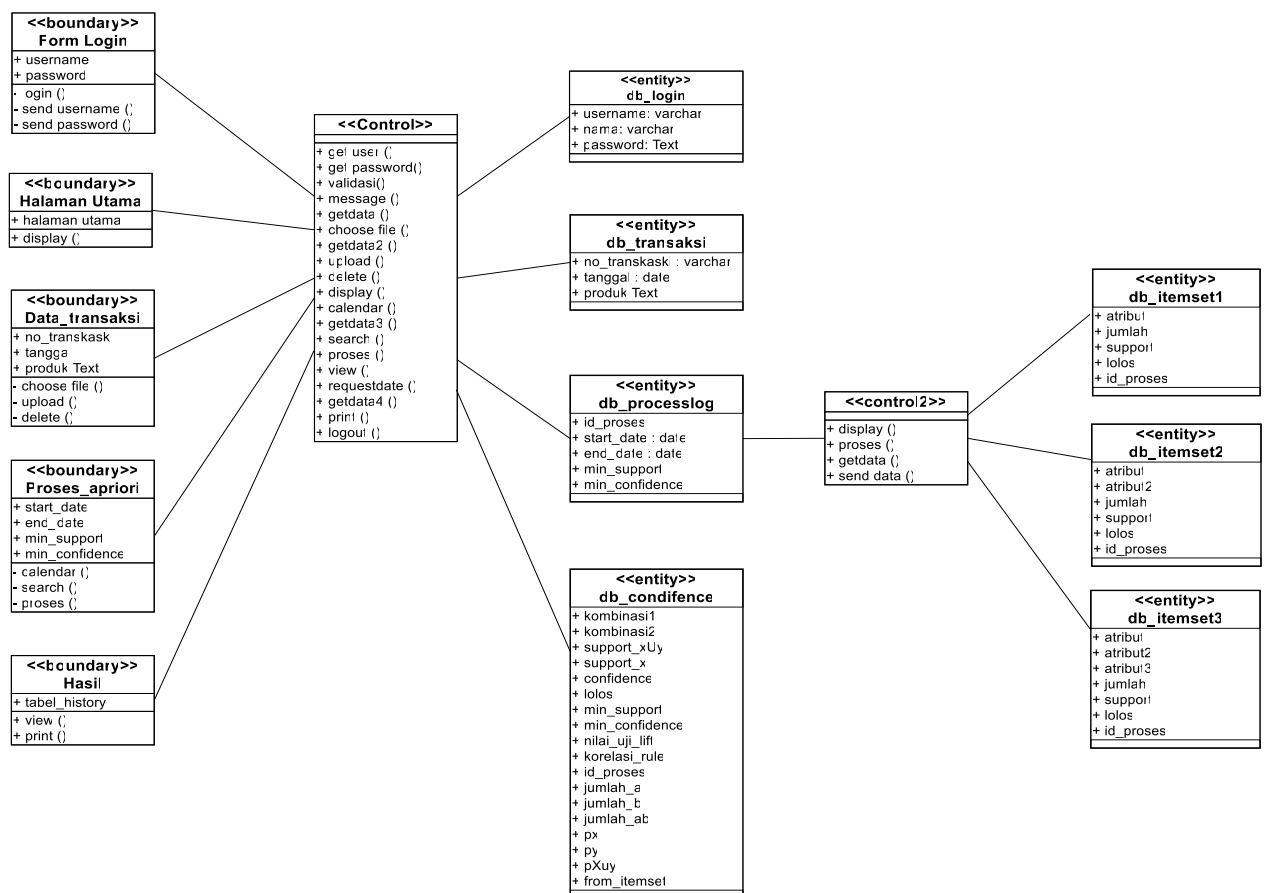
### 9. Sequence diagram print



GAMBAR: 3.20 Sequence diagram print

### 3.3.1.4 Class Diagram

Pada *Class* diagram menggambarkan struktur dan deskripsi kelas , objek dan hubungan satu sama lain.



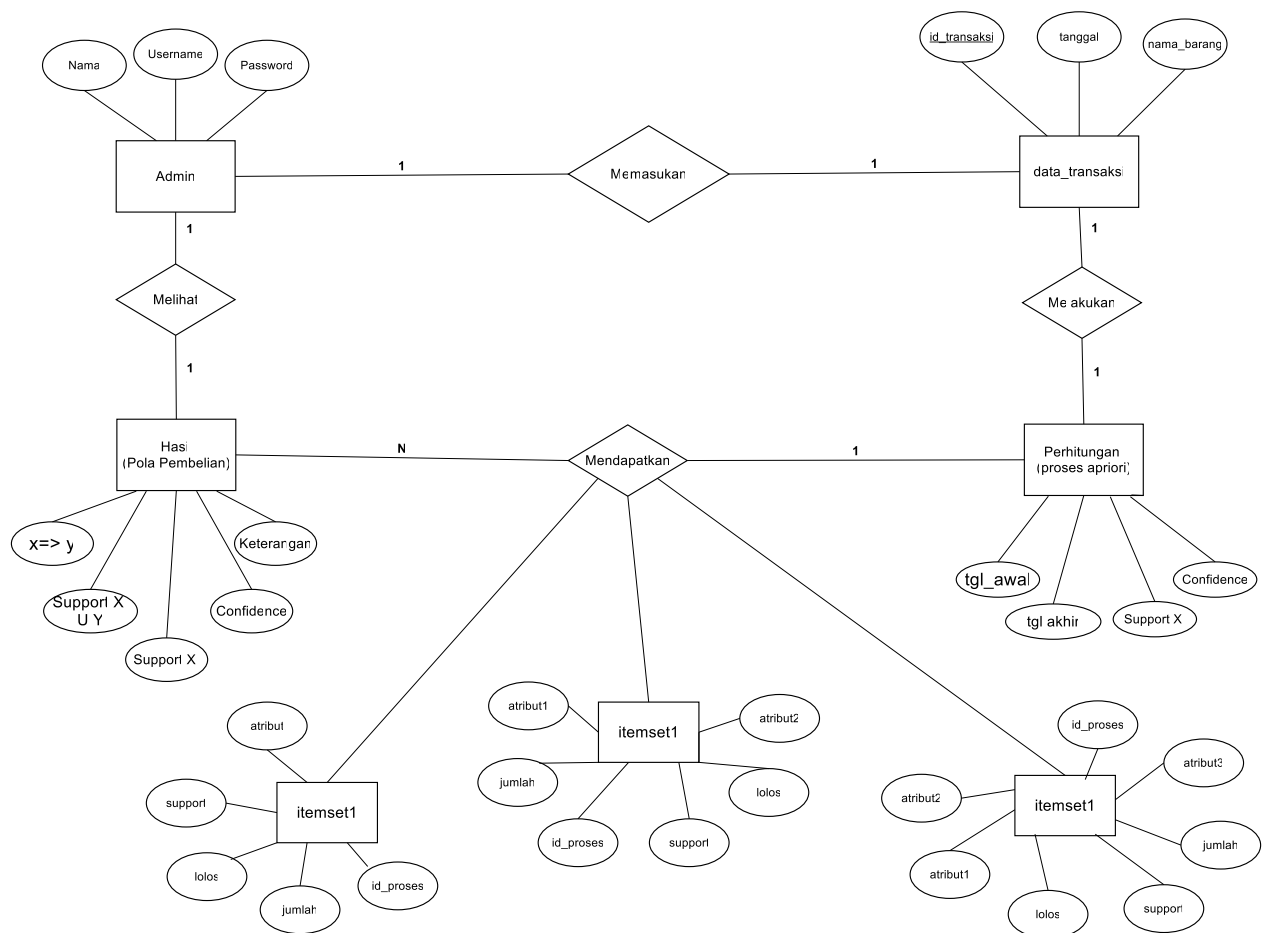
GAMBAR: 3.21 Class Diagram

### 3.3.2 Desain Database Sistem

Dalam penelitian ini perancangan *database* nya akan membagi menjadi 2 bagian yaitu perancangan database dengan menggunakan *Entity Relationship Diagram* (ERD) dan juga perancangan database dengan menggunakan struktur tabel. Berikut adalah perancangan database sistem.

### 3.3.2.1 Entity Relationship Diagram (ERD)

Berikut *Entity Relationship Diagram* yang di gunakan dalam perancangan sistem ini :



GAMABAR: 3.22 ERD Aplikasi Analisa Keranjang Belanja

### 3.3.2.2 Struktur Tabel

Berikut adalah struktur tabel yang digunakan dalam pembuatan sistem aplikasi analisa keranjang belanja.

#### 1. Tabel User

**TABEL : 3.9 Perancangan Database Tabel *User***

Field	Tipe data
<b>username</b>	varchar(200)
<b>nama</b>	varchar(200)
<b>password</b>	Text

#### 2. Tabel Data Transaksi

**TABEL : 3.10 Perancangan Database Data Transaksi**

Field	Tipe Data
<b>Id</b>	Int(11)
<b><i>Transaction</i></b>	<i>Date</i>
<b>Produk</b>	<i>Text</i>

#### 3. Tabel *process\_log*

**TABEL : 3.11 Perancangan Database *process\_log***

Field	Tipe data
<b>id</b>	int(11)
<b><i>start_date</i></b>	date
<b><i>end_date</i></b>	date
<b><i>min_support</i></b>	<i>Double</i>
<b><i>min_confidence</i></b>	<i>Double</i>

4. Tabel *itemset1***TABEL : 3.12** Tabel perancangan database *itemset1*

<i>Field</i>	<b>Tipe data</b>
<i>Atribut</i>	varchar(200)
<i>jumlah</i>	int(11)
<i>support</i>	Double
<i>Lolos</i>	tinyint(4)
<i>Id_process</i>	Int(11)

5. Tabel *itemset2***TABEL : 3.13** Tabel perancangan database *itemset2*

<i>Field</i>	<b>Tipe data</b>
<b>atribut1</b>	varchar(200)
<b>atribut2</b>	varchar(200)
<b>jumlah</b>	int(11)
<b>support</b>	Double
<b>Lolos</b>	tinyint(4)
<b>Id_process</b>	Int(11)

6. Tabel *itemset3***TABEL : 3.14** Tabel Perancangan database *itemset3*

<i>Field</i>	<b>Tipe data</b>
<i>atribut1</i>	varchar(200)
<i>atribut2</i>	varchar(200)
<i>atribut2</i>	varchar(200)
<i>jumlah</i>	int(11)
<i>support</i>	double
<i>Lolos</i>	tinyint(4)
<i>Id_process</i>	Int(11)

## 7. Tabel confidence

**TABEL : 3.15 Tabel Perancangan database confidence**

Field	Tipe data
<b>kombinasi1</b>	Varchar (255)
<b>kombinasi2</b>	Varchar(255)
<b>support_xUy</b>	Double
<b>support_x</b>	Double
<b>confidence</b>	Double
<b>lolos</b>	Tinyint(4)
<b>min_support</b>	Double
<b>min_confidence</b>	Double
<b>nilai_uji_lift</b>	Double
<b>korelasi_rule</b>	Varchar(100)
<b>id_process</b>	Int(11)
<b>jumlah_a</b>	Int(11)
<b>jumlah_b</b>	Int(11)
<b>jumlah_ab</b>	Int(11)
<b>px</b>	double
<b>Py</b>	double
<b>Pxuy</b>	double
<b>from_itemset</b>	Int(11)

### 3.3.3 Desain Interface Sistem

Sebelum membuat program dibuatkan sebuah rancangan tampilan sistem , untuk selanjutnya di kembangan menjadi halaman sistem sesuai kebutuhan. Berikut adalah desain tampilan sistem yang akan di buat.

### 1. Desain Halaman *Login*

The wireframe shows a rectangular page layout. On the left side, there is a square box labeled "logo". On the right side, there are two stacked input fields: the top one is labeled "username" and the bottom one is labeled "password". Below the "password" field is a button labeled "login".

GAMBAR: 3.23 Desain Halaman *Login*

### 2. Desain Menu Halaman utama

The wireframe shows a page layout with a vertical menu on the left and a main content area on the right. The menu consists of five items: "Halaman Utama", "Data Transaksi", "Proses Apri ori", "Hasil", and "Logout". The "Halaman Utama" item is currently selected, indicated by a horizontal line underneath it. The main content area is currently empty.

GAMBAR: 3.24 Desain Halaman Utama

### 3. Desain menu Data Transaksi

Halaman Utama		Data Transaksi							
Data Transaksi		<input type="text"/> <input type="button" value="Choose file"/>							
Proses Apriori		<input type="button" value="Upload"/> <input type="button" value="Delete"/>							
Hasil		<table border="1"> <thead> <tr> <th>ID transaks</th> <th>Tanggal</th> <th>Produk</th> </tr> </thead> <tbody> <tr> <td> </td> <td> </td> <td> </td> </tr> </tbody> </table>		ID transaks	Tanggal	Produk			
ID transaks	Tanggal	Produk							
Logout									

GAMBAR: 3.25 Desain Halaman Data Transaksi

### 4. Desain menu Proses Apriori

Halaman Utama		Proses Apriori	
Data Transaksi		Tanggal <input type="text"/> <input type="button" value="Search"/>	
Proses Apriori		Min Support <input type="text"/>	
Hasil		Min Confidence <input type="text"/>	
Logout		<input type="button" value="Proses"/>	

GAMBAR: 3.26 Desain Halaman Proses Apriori



## 5. Desain menu Hasil

Halaman Utama	Hasil						
Data Transaksi							
Proses Apri ori	NC	Star Date	End Date	Min Support	Min Confidence		Print
Hasil						View Rule	PDF
Logout							

GAMBAR: 3.27 Desain Halaman Hasil

## **BAB IV**

### **UJI COBA DAN IMPLEMENTASI SISTEM**

#### **4.1 Implementasi**

##### **4.1.1 Implementasi Perangkat Keras**

Spesifikasi yang di sarankan dalam penerapan aplikasi sesuai dengan apa yang telah di jabarkan di bab 3, sebagai berikut :

1. Komputer dengan prosesor pentium dual core
2. Memory Komputer minimal 4GB atau lebih
3. Resolusi VGA minimum 800 x 600
4. Modem / Perangkat Internet

##### **4.1.2 Implementasi Perangkat Lunak**

Perangkat lunak yang digunakan untuk menunjang penerapan aplikasi ini, sebagai berikut :

1. Windows 7, windows 8/8.1 , Windows 10.
2. *Browser*
3. XAMPP

##### **4.1.3 Implementasi database**

Sesuai dengan yang telah dijelaskan pada bab 3 mengenai database yang digunakan pada sistem yang digunakan, berikut adalah implementasi dari database yang digunakan pada aplikasi.

1. Entitas-entitas yang terdapat pada database yang digunakan

Table	Action
<b>confidence</b>	★ Browse Structure Search Insert Empty
<b>itemset1</b>	★ Browse Structure Search Insert Empty
<b>itemset2</b>	★ Browse Structure Search Insert Empty
<b>itemset3</b>	★ Browse Structure Search Insert Empty
<b>process_log</b>	★ Browse Structure Search Insert Empty
<b>transaksi</b>	★ Browse Structure Search Insert Empty
<b>users</b>	★ Browse Structure Search Insert Empty

GAMBAR: 4.1 Entitas Pada database

2. Script SQL membuat entitas user

```
CREATE TABLE `users` (
  `username` varchar(200) DEFAULT NULL,
  `nama` varchar(200) DEFAULT NULL,
  `password` text
);
```

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	<b>id</b> 🔑	int(11)			No	None		AUTO_INCREMENT
2	<b>username</b>	varchar(200)	latin1_swedish_ci		Yes	NULL		
3	<b>nama</b>	varchar(200)	latin1_swedish_ci		Yes	NULL		
4	<b>password</b>	text	latin1_swedish_ci		Yes	NULL		


GAMBAR: 4.2 Tabel User

3. Script SQL membuat entitas data transaksi

```
CREATE TABLE `transaksi` (
  `id` int(11) NOT NULL,
  `transaction_date` date DEFAULT NULL,
```

``produk` text`

) ENGINE=InnoDB DEFAULT CHARSET=latin1;

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	<b>id</b> 	int(11)			No	None		AUTO_INCREMENT
2	<b>transaction_date</b>	date			Yes	NULL		
3	<b>produk</b>	text	latin1_swedish_ci		Yes	NULL		

GAMBAR: 4.3 Tabel data transaksi

4. Script SQL membuat entitas process\_log

CREATE TABLE `process\_log` (

``id` int(11) NOT NULL,`


``start_date` date DEFAULT NULL,`

``end_date` date DEFAULT NULL,`

``min_support` double DEFAULT NULL,`

``min_confidence` double DEFAULT NULL`

) ENGINE=InnoDB DEFAULT CHARSET=latin1;

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	<b>id</b> 	int(11)			No	None		AUTO_INCREMENT
2	<b>start_date</b>	date			Yes	NULL		
3	<b>end_date</b>	date			Yes	NULL		
4	<b>min_support</b>	double			Yes	NULL		
5	<b>min_confidence</b>	double			Yes	NULL		

GAMBAR: 4.4 Tabel process\_log

5. Script SQL membuat entitas itemset1

CREATE TABLE `itemset1` (

``atribut` varchar(200) DEFAULT NULL,`

``jumlah` int(11) DEFAULT NULL,`

```

`support` double DEFAULT NULL,
`lolos` tinyint(4) DEFAULT NULL,
`id_process` int(11) NOT NULL DEFAULT '0'
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

```

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	<b>atribut</b>	varchar(200)	latin1_swedish_ci		Yes	NULL		
2	<b>jumlah</b>	int(11)			Yes	NULL		
3	<b>support</b>	double			Yes	NULL		
4	<b>lolos</b>	tinyint(4)			Yes	NULL		
5	<b>id_process</b>	int(11)			No	0		

GAMBAR: 4.5 Tabel itemset1

6. Script SQL entitas itemset2

```

CREATE TABLE `itemset2` (
  `atribut1` varchar(200) DEFAULT NULL,
  `atribut2` varchar(200) DEFAULT NULL,
  `jumlah` int(11) DEFAULT NULL,
  `support` double DEFAULT NULL,
  `lolos` tinyint(4) DEFAULT NULL,
  `id_process` int(11) NOT NULL DEFAULT '0'
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

```

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	<b>atribut1</b>	varchar(200)	latin1_swedish_ci		Yes	NULL		
2	<b>atribut2</b>	varchar(200)	latin1_swedish_ci		Yes	NULL		
3	<b>jumlah</b>	int(11)			Yes	NULL		
4	<b>support</b>	double			Yes	NULL		
5	<b>lolos</b>	tinyint(4)			Yes	NULL		
6	<b>id_process</b>	int(11)			No	0		

GAMBAR: 4.6 Tabel itemset2

## 7. Script SQL membuat entitas itemset3

```
CREATE TABLE `itemset3` (
  `atribut1` varchar(200) DEFAULT NULL,
  `atribut2` varchar(200) DEFAULT NULL,
  `atribut3` varchar(200) DEFAULT NULL,
  `jumlah` int(11) DEFAULT NULL,
  `support` double DEFAULT NULL,
  `lolos` tinyint(4) DEFAULT NULL,
  `id_process` int(11) NOT NULL DEFAULT '0'
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	<b>atribut1</b>	varchar(200)	latin1_swedish_ci		Yes	NULL		
2	<b>atribut2</b>	varchar(200)	latin1_swedish_ci		Yes	NULL		
3	<b>atribut3</b>	varchar(200)	latin1_swedish_ci		Yes	NULL		
4	<b>jumlah</b>	int(11)			Yes	NULL		
5	<b>support</b>	double			Yes	NULL		
6	<b>lolos</b>	tinyint(4)			Yes	NULL		
7	<b>id_process</b>	int(11)			No	0		

GAMBAR: 4.7 Tabel itemset3

## 8. Script SQL membuat entitas confidence

```

CREATE TABLE `confidence` (
  `kombinasi1` varchar(255) DEFAULT NULL,
  `kombinasi2` varchar(255) DEFAULT NULL,
  `support_xUy` double DEFAULT NULL,
  `support_x` double DEFAULT NULL,
  `confidence` double DEFAULT NULL,
  `lolos` tinyint(4) DEFAULT NULL,
  `min_support` double DEFAULT NULL,
  `min_confidence` double DEFAULT NULL,
  `nilai_uji_lift` double DEFAULT NULL,
  `korelasi_rule` varchar(100) DEFAULT NULL,
  `id_process` int(11) NOT NULL DEFAULT '0',
  `jumlah_a` int(11) DEFAULT NULL,
  `jumlah_b` int(11) DEFAULT NULL,
  `jumlah_ab` int(11) DEFAULT NULL,
  `px` double DEFAULT NULL,
  `py` double DEFAULT NULL,
  `pxuy` double DEFAULT NULL,
  `from_itemset` int(11) DEFAULT NULL COMMENT 'dari itemset 2/3'
) ENGINE=MyISAM DEFAULT CHARSET=latin1;

```

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	<b>kombinasi1</b>	varchar(255)	latin1_swedish_ci		Yes	NULL		
2	<b>kombinasi2</b>	varchar(255)	latin1_swedish_ci		Yes	NULL		
3	<b>support_xUy</b>	double			Yes	NULL		
4	<b>support_x</b>	double			Yes	NULL		
5	<b>confidence</b>	double			Yes	NULL		
6	<b>lolos</b>	tinyint(4)			Yes	NULL		
7	<b>min_support</b>	double			Yes	NULL		
8	<b>min_confidence</b>	double			Yes	NULL		
9	<b>nilai_uji_lift</b>	double			Yes	NULL		
10	<b>korelasi_rule</b>	varchar(100)	latin1_swedish_ci		Yes	NULL		
11	<b>id_process</b>	int(11)			No	0		
12	<b>jumlah_a</b>	int(11)			Yes	NULL		
13	<b>jumlah_b</b>	int(11)			Yes	NULL		
14	<b>jumlah_ab</b>	int(11)			Yes	NULL		
15	<b>px</b>	double			Yes	NULL		
16	<b>py</b>	double			Yes	NULL		
17	<b>pxuy</b>	double			Yes	NULL		
18	<b>from_itemset</b>	int(11)			Yes	NULL	dari itemset 2/3	

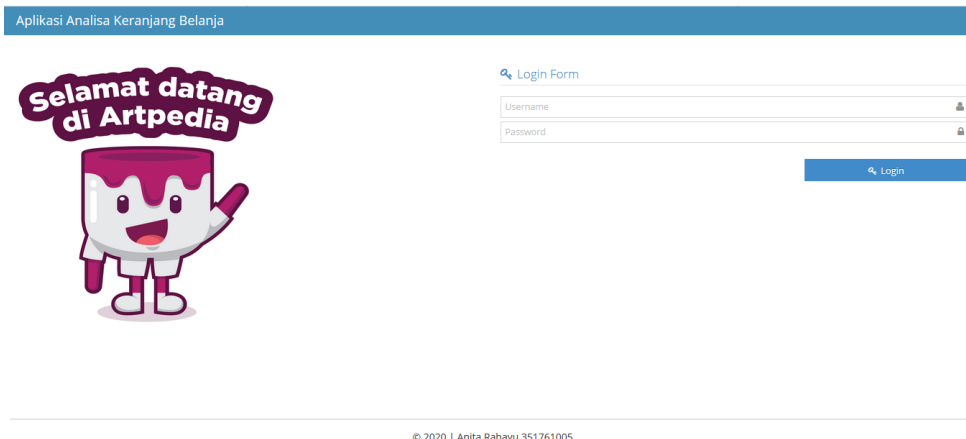
GAMBAR: 4.8 Tabel Confidence

### 4.1.3 Implementasi Antarmuka

Adapun implementasi dari aplikasi yang telah di bangun menggunakan PHP sesuai dengan rancangan yang telah di gambarkan pada bab3. Berikut tampilan aplikasi yang telah di buat.

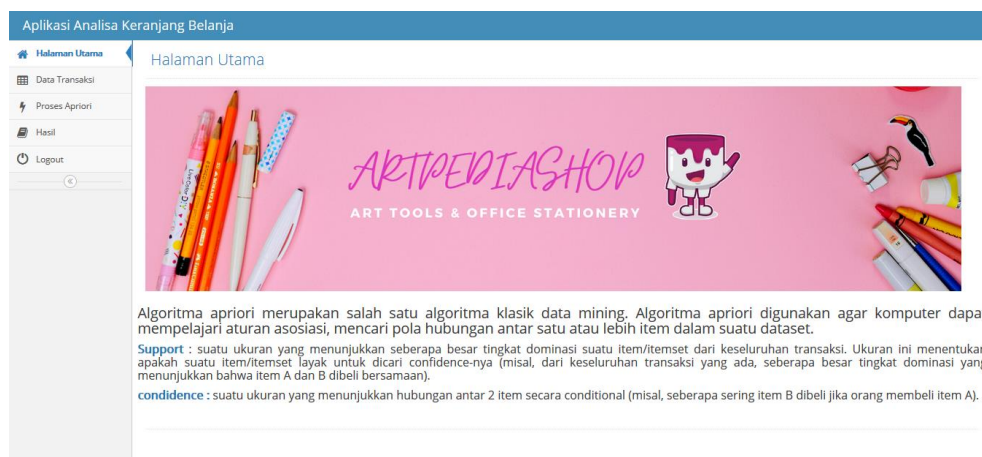
1. Menu yang di tampilkan oleh sistem pada saat *user* akan melakukan *login* dan mencoba untuk mengakses sistem.





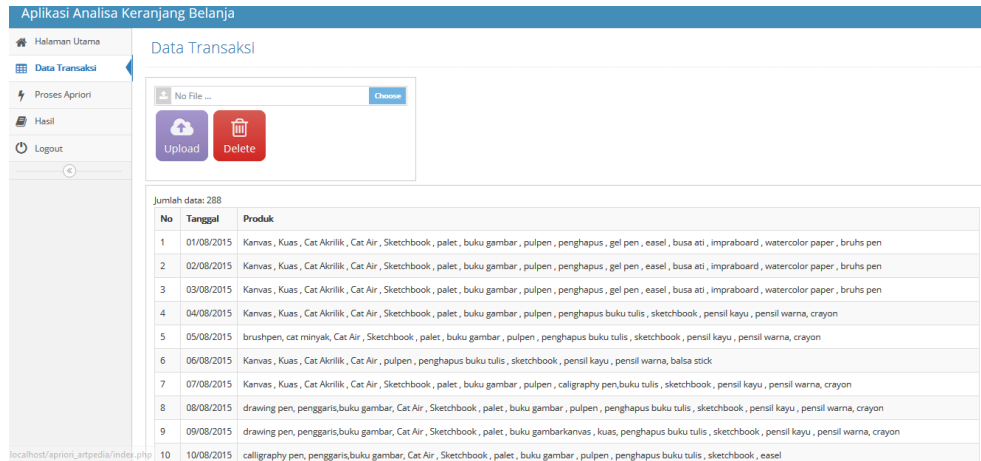
GAMBAR 4.9 Halaman *Login*

2. Setelah *user* berhasil melakukan *login*, sistem akan menampilkan halaman utama. Untuk Admin mempunyai kuasa penuh terhadap akses aplikasi.



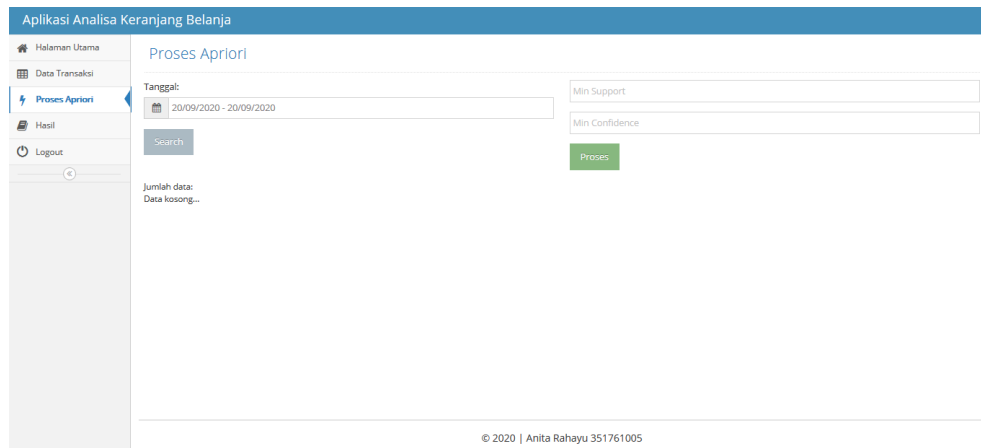
GAMBAR:4.10 Halaman Utama

3. Pada halaman data transaksi admin dapat mengupload data transaksi dan menghapusnya.

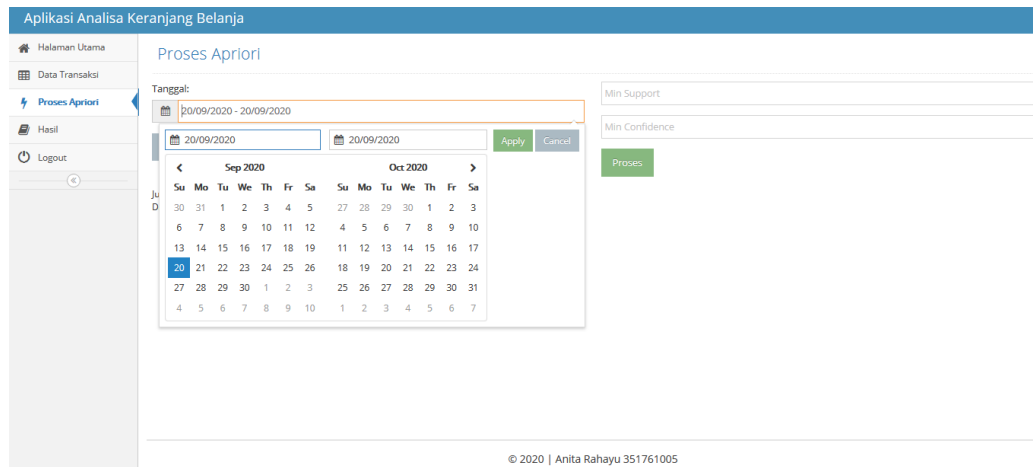


GAMBAR: 4.11 Halaman Data Transaksi

4. Pada halaman proses apriori, data transaksi yang telah di upload sebelumnya dapat di lakukan perhitungan dengan memilih tanggal mulai dan selesai, lalu mengisi *min support* dan *min confidence* lalu mengklik tombol proses.



GAMBAR: 4.12 Halaman proses apriori



GAMBAR: 4.13 Memilih *periode* tanggal mulai dan selesai

5. Pada halaman hasil dapat melihat *history* hasil perhitungan yang telah di lakukan sebelumnya. Juga terdapat fitur “*view rule*” dan “*print*”.

No	Start Date	End Date	Min Support	Min Confidence		Pdf
1	13/08/2010	13/08/2017	20	35	<a href="#">View rule</a>	
2	13/03/2015	13/10/2015	4	1	<a href="#">View rule</a>	
3	13/03/2014	13/03/2020	20	20	<a href="#">View rule</a>	
4	01/08/2015	31/08/2015	10	20	<a href="#">View rule</a>	
5	01/09/2015	30/09/2015	20	35	<a href="#">View rule</a>	
6	02/08/2017	17/08/2017	25	30	<a href="#">View rule</a>	



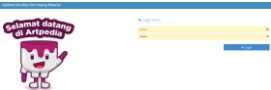

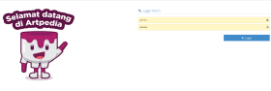

GAMBAR: 4.14 Halaman Hasil

## 4.2 Uji Coba

Metode uji coba yang digunakan adalah *Black Box Testing*. Metode *Black Box* ini menguji fungsi-fungsi yang ada pada aplikasi secara keseluruhan. Berikut adalah hasil dari pengujian yang dilakukan terhadap aplikasi ini:



1. Hasil pengujian pada menu login

TABEL: 4.1 Tabel Pengujian Login

No.	Skenario Pengujian	Yang Diharapkan	Output	Kesimpulan
1.	Form username dan password tidak diisi. <b>Test Case :</b> 	Sistem memberitahukan bahwa gagal login, muncul pesan error!	Sesuai yang diharapkan. <b>Test Result :</b> 	Valid
2.	Menggunakan <i>username</i> yang benar dan <i>password</i> yang salah atau sebaliknya. <b>Test Case :</b> 	Sistem memberitahukan bahwa <i>error!</i> , <i>login failed</i> .	Sesuai yang diharapkan. <b>Test Result :</b> 	Valid
3.	Menggunakan <i>username</i> & <i>password</i> yang benar. <b>Test Case :</b> 	Login berhasil dan sistem akan menampilkan halaman utama aplikasi.	Sesuai yang diharapkan. <b>Test Result :</b> 	Valid

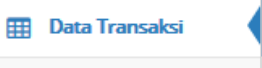
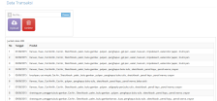
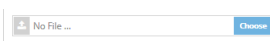
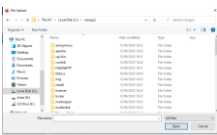
## 2. Hasil Pengujian pada menu Halaman Utama



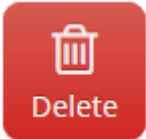

TABEL : 4.2 Tabel Pengujian Halaman Utama

No.	Skenario Pengujian	Yang Diharapkan	Output	Kesimpulan
1.	Mengklik Menu Halaman utama <b>Test Case :</b> 	Sistem menampilkan halaman utama	Sesuai yang diharapkan. <i>Test Result :</i> 	Valid

## 3. Hasil Pengujian Pada menu Data transaksi

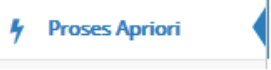

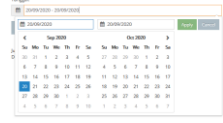
TABEL : 4.3 Tabel Pengujian Data Transaksi


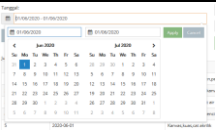
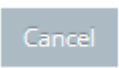
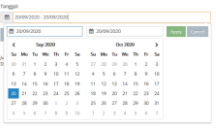

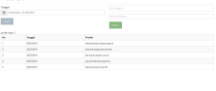


No.	Skenario Pengujian	Yang Diharapkan	Output	Kesimpulan
1.	Mengklik Menu Data Transaksi <b>Test Case :</b> 	Sistem menampilkan halaman data transaksi	Sesuai yang diharapkan. <i>Test Result :</i> 	Valid
2.	Mengklik ikon “Choose” pada data transaksi 	Sistem menampilkan windows explore	Sesuai yang diharapkan. <i>Test Result :</i> 	Valid
3.	Mengklik button “upload” pada data transaksi.	Sistem menampilkan pesan bahwa	Sesuai yang diharapkan. <i>Test Result :</i>	Valid

		“data berhasil di simpan”		
4.	Mengklik button “Delete” pada data transaksi. 	Sistem menampilkan pesan “ are you sure” pilih yes /no dan pesan bahwa data berhasil di hapus	Sesuai yang diharapkan. <i>Test Result :</i> 	Valid

#### 4. Hasil Pengujian Pada menu Proses Apriori


TABEL: 4.4 Tabel Pengujian Proses Apriori

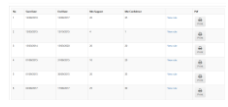
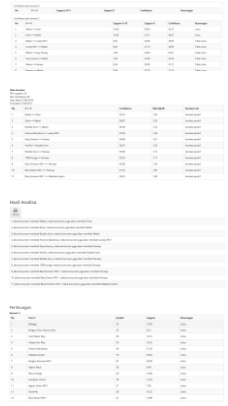
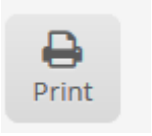

No.	Skenario Pengujian	Yang Diharapkan	Output	Kesimpulan
1.	Mengklik Menu Proses Apriori <i>Test Case :</i> 	Sistem menampilkan halaman proses apriori	Sesuai yang diharapkan. <i>Test Result :</i> 	Valid
2.	Mengklik ikon “kalender” pada proses apriori	Sistem menampilkan kalender	Sesuai yang diharapkan. <i>Test Result :</i> 	Valid
3.	Mengklik button “apply” pada proses apriori pilih tanggal	Sistem menampilkan tanggal terpilih	Sesuai yang diharapkan. <i>Test Result :</i>	Valid

				
4.	Mengklik button “cancel” pada proses apriori pilih tanggal 	Sistem menampilkan ulang kalender	Sesuai yang diharapkan. <i>Test Result :</i> 	Valid
5.	Mengklik button “search” pada proses apriori pilih tanggal 	Sistem menampilkan data transaksi tanggal perilih	Sesuai yang diharapkan. <i>Test Result :</i> 	Valid
6.	Mengklik button “proses” pada proses apriori 	Sistem menampilkan hasil perhitungan	Sesuai yang diharapkan. <i>Test Result :</i> 	Valid

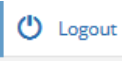

#### 5. Hasil Pengujian Pada menu Hasil

TABEL: 4.5 Tabel Pengujian Hasil

No.	Skenario Pengujian	Yang Diharapkan	Output	Kesimpulan
1.	Mengklik Menu Hasil <b>Test Case :</b> 	Sistem menampilkan halaman hasil	Sesuai yang diharapkan. <i>Test Result :</i>	Valid

				
2.	Mengklik “ <i>view rule</i> ” pada hasil	Sistem menampilkan tabel <i>itemset3</i> , <i>itemset2</i> , <i>rule asosiasi</i> , hasil analisa dan perhitungan	Sesuai yang diharapkan. <i>Test Result :</i> 	Valid
3.	Mengklik button “ <i>print</i> ” pada hasil 	Sistem menampilkan pratinjau pdf data yg akan di cetak	Sesuai yang diharapkan. <i>Test Result :</i> 	Valid

6. Hasil Pengujian Pada menu *Logout*TABEL : 4.6 Tabel Pengujian *Logout*

No.	Skenario Pengujian	Yang Diharapkan	Output	Kesimpulan
1.	Mengklik Menu <i>logout</i> <b>Test Case :</b> 	Sistem menampilkan halaman login	Sesuai yang diharapkan. <i>Test Result :</i> 	Valid



## **BAB V PENUTUP**

### **5.2 Kesimpulan**

Berdasarkan penelitian yang telah dilakukan penulis mengenai aplikasi analisa keranjang belanja ini, dapat dikemukakan beberapa kesimpulan dari penelitian yang dilakukan sebagai berikut :

1. Aplikasi analisa keranjang belanja ini dapat membantu dalam pengambilan keputusan untuk menentukan produk promosi.
2. Dapat menentukan hubungan antar produk *itemset3* , *itemset2* dan cara perhitungannya.

### **4.2 Saran**

Berdasarkan pemaparan yang telah di jelaskan di bab sebelumnya , aplikasi ini masih memiliki beberapa kekurangan dan dapat dikembangkan agar menjadi lebih baik lagi, Adapun saran penulis, antara lain :

1. Dapat dikembangkan dengan menabahkan level *login* agar ada pemisahan hak akses.
2. Penambahan fitur filter dalam persentase hasil analisa, untuk mengurutkan hasil confidence .

## DAFTAR PUSTAKA

- Arief, 2011. Pengertian MySQL (My Structure Query Language) [WWW Document].
- A.S, Rosa dan Shalahuddin, M. 2014. Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek. Bandung: Informatika.
- Badrul, M., n.d. ALGORITMA ASOSIASI DENGAN ALGORITMA APRIORI UNTUK ANALISA DATA PENJUALAN 9.
- Kadir.A., 2002. Pengenalan Sistem Informasi. Penerbit Andi. Yogyakarta
- Kadir, A., 2008. Tuntunan Praktis Belajar Database Menggunakan MySQL, C.V Andi Offset. Yogyakarta.
- Nursikuwagus, A., Hartono, T., 2016. IMPLEMENTASI ALGORITMA APRIORI UNTUK ANALISIS PENJUALAN DENGAN BERBASIS WEB. Simet 7, 701. <https://doi.org/10.24176/simet.v7i2.784>
- Smitdev Community, A.Z., 2019. Pengertian Aplikasi: Arti, Fungsi, Klasifikasi, dan Contoh Aplikasi [WWW Document]. URL <https://www.maxmanroe.com/vid/teknologi/pengertian-aplikasi.html>
- Sutabri, T., 2004. . Pengertian Sistem | Definisi, Unsur-Unsur, Jenis-Jenis, dan Elemennya. URL <https://www.zonareferensi.com/pengertian-sistem/>
- Sutabri, Tata. S.Kom,MM. (2004). Analisa Sistem Informasi. Edisi Pertama. Yogyakarta: Andi.

- Qomariah, S., & Rangan, A. Y. (2015). Penerapan Metode C4. 5 dan Naïve Bayes untuk Klasifikasi Tipe Belajar Siswa di SMKN 4 Samarinda. *Prosiding Senatkom, 1*.
- Udariansyah, D., & Kom, M. (2016). Sistem Informasi Penjualan Barang Pada CV. Sinar Musi Group Palembang Berbasis WEB Menerapkan Metode Single Moving Average. *Jurnal Informanika, 2(2)*, 12-22.
- Wibowo, R. (2020). *APLIKASI PEMESANAN LAPANGAN FUTSAL DI TELAGA 1 FUTSAL BERBASIS ANDROID* (Doctoral dissertation, STMIK AKAKOM YOGYAKARTA).
- Wijaya, Krisna Nata. 2017. "Analisa Pola Frekuensi Keranjang Belanja dengan Algoritma Apriori." 3(1):4.
- Yulita, Marsela dan Veronica S. Moertini. Analisis Keranjang Pasar dengan Algoritma Hash-Based pada transaksi penjualan di Apotik. Bandung: Jurnal Integral Majalah Ilmiah Matematika dan Ilmu Pengetahuan Alam, Vol 9, No 3, Jurusan Ilmu Komputer Universitas Katolik Parahyangan, 2004

## LAMPIRAN

```
<?php
function reset_temporary($db_object){
    $sql1 = "TRUNCATE itemset1";
    $db_object->db_query($sql1);

    $sql2 = "TRUNCATE itemset2";
    $db_object->db_query($sql2);

    $sql3 = "TRUNCATE itemset3";
    $db_object->db_query($sql3);

    $sql4 = "TRUNCATE confidence";
    $db_object->db_query($sql4);
}

function reset_hitungan($db_object, $id_process){
    $condition = array("id_process"=>$id_process);
    $db_object->delete_record("itemset1", $condition);

    $db_object->delete_record("itemset2", $condition);

    $db_object->delete_record("itemset3", $condition);
```

```

$db_object->delete_record("confidence", $condition);
}

function is_exist_variasi_itemset($array_item1, $array_item2, $item1, $item2) {
    $bool1 = array_keys(array_map('strtoupper', $array_item1),
    strtoupper($item1));
    $bool2 = array_keys(array_map('strtoupper', $array_item2),
    strtoupper($item2));
    $bool3 = array_keys(array_map('strtoupper', $array_item2),
    strtoupper($item1));
    $bool4 = array_keys(array_map('strtoupper', $array_item1),
    strtoupper($item2));

    foreach ($bool1 as $key => $value) {
        $aa = array_search($value, $bool2);
        if(is_numeric($aa)){
            return true;
        }
    }

    foreach ($bool3 as $key => $value) {
        $aa = array_search($value, $bool4);
        if(is_numeric($aa)){
            return true;
        }
    }

    return false;
}

```

```
}

```

```
function mining_process($db_object, $min_support, $min_confidence,
$start_date, $end_date, $id_process){
    $sql_trans = "SELECT * FROM transaksi
        WHERE transaction_date BETWEEN '$start_date' AND '$end_date' ";
    $result_trans = $db_object->db_query($sql_trans);
    $dataTransaksi = $item_list = array();
    $jumlah_transaksi = $db_object->db_num_rows($result_trans);
    $min_support_relative = ($min_support/$jumlah_transaksi)*100;
    $x=0;
    while($myrow = $db_object->db_fetch_array($result_trans)){
        $dataTransaksi[$x]['tanggal'] = $myrow['transaction_date'];
        $item_produk = $myrow['produk'].",";
        $item_produk = str_replace(" ", ",", $item_produk);
        $item_produk = str_replace(" ", ",", $item_produk);
        $item_produk = str_replace(" ", ",", $item_produk);
        $item_produk = str_replace(" ", ",", $item_produk);
        $item_produk = str_replace(" ", ",", $item_produk);
        $item_produk = str_replace(" ", ",", $item_produk);
        $item_produk = str_replace(" ", ",", $item_produk);
        $item_produk = str_replace(" ", ",", $item_produk);
        $dataTransaksi[$x]['produk'] = $item_produk;
        $produk = explode(",", $myrow['produk']);
        foreach ($produk as $key => $value_produk) {
            if(!in_array(strtoupper($value_produk), array_map('strtoupper',
$item_list))){

```

```

        if(!empty($value_produk)){
            $item_list[] = $value_produk;
        }
    }
}
$x++;
}

echo "<br><strong>Itemset 1:</strong><br>";
echo "<table class='table table-bordered table-striped table-hover'>
    <tr>
        <th>No</th>
        <th>Item</th>
        <th>Jumlah</th>
        <th>Support</th>
        <th>Keterangan</th>
    </tr>";
$itemset1 = $jumlahItemset1 = $supportItemset1 = $valueIn = array();
$x=1;
foreach ($item_list as $key => $item) {
    $jumlah = jumlah_itemset1($dataTransaksi, $item);
    $support = ($jumlah/$jumlah_transaksi) * 100;
    $lolos = ($support>=$min_support_relative)?"1":"0";
    $valueIn[] = ("'$item','$jumlah','$support','$lolos','$id_process')";
    if($lolos){
        $itemset1[] = $item;
        $jumlahItemset1[] = $jumlah;
    }
}

```

```

        $supportItemset1[] = $support;
    }
    echo "<tr>";
    echo "<td>" . $x . "</td>";
    echo "<td>" . $item . "</td>";
    echo "<td>" . $jumlah . "</td>";
    echo "<td>" . price_format($support) . "</td>";
    echo "<td>" . (($lolos==1)?"Lolos":"Tidak Lolos") . "</td>";
    echo "</tr>";
    $x++;
}
echo "</table>";

$value_insert = implode(",", $valueIn);
$sql_insert_itemset1 = "INSERT INTO itemset1 (atribut, jumlah, support,
lolos, id_process) "
    . " VALUES ".$value_insert;
$db_object->db_query($sql_insert_itemset1);

```