

BAB II

LANDASAN TEORI

2.1 Teori Pengolahan Data

2.1.1 Data

Data dalam bidang komputasi atau komputer didefinisikan sebagai suatu fakta dalam bentuk karakter, simbol, kualitas, yang dioperasi atau diproses dengan berbagai algoritma oleh komputer. Hasil dari proses tersebut disimpan, ditransmisikan, lalu dilakukan perubahan dalam berbagai bentuk, seperti *binary digital*, dan sinyal listrik (Arhami & Nasir, 2020). Data yaitu suatu istilah majemuk yang berarti fakta atau bagian dari fakta yang mengandung arti yang dihubungkan dengan kenyataan, simbol-simbol, gambar-gambar, angka-angka, huruf atau simbol yang menunjukkan suatu ide, objek, kondisi, atau situasi (Ayu & Permatasari, 2018).

2.1.2 Variabel

Variabel diartikan sebagai suatu karakteristik yang dapat mengasumsikan lebih dari satu himpunan nilai dalam ukuran numerik yang diberikan. Variabel juga dapat berupa suatu nama yang diberikan untuk suatu observasi, pengukuran, atau perhitungan dari sekumpulan data yang telah diketahui (Arhami & Nasir, 2020).

Atribut adalah sifat, properti, atau karakteristik objek yang nilainya bisa bermacam-macam dari satu objek dengan objek lainnya, maupun dari satu waktu ke waktu yang lainnya. Atribut juga sering disebut variabel, *field*, fitur, atau dimensi (Nizar, 2016). Variabel atau atribut dapat dibedakan dalam dua tipe (Arhami & Nasir, 2020) sebagai berikut :

1. Variabel kualitatif, yaitu variabel dengan karakteristik yang dihasilkan tidak dalam bentuk angka, tetapi dalam bentuk kategori. Contoh variabel kualitatif adalah agama, jenis kelamin, tipe kendaraan, warna mata, dan lain lain.

2. Variabel kuantitatif, yaitu variabel yang disajikan dalam bentuk angka atau hasil akhirnya diperoleh dari suatu pengukuran fisik atau perhitungan. Contohnya umur, daya tahan baterai mobil, banyaknya anak, kecepatan kendaraan dan lain lain.

2.1.3 Informasi

Informasi dapat didefinisikan sebagai data yang telah direkam, diklasifikasikan, diorganisir, direalisasikan, dan kemudian dibuat kerangka penafsiran atau interpretasi terhadap makna yang dikandung sehingga menghasilkan suatu kesimpulan yang tepat (Arhami & Nasir, 2020).

Informasi adalah data yang telah diklasifikasi atau diolah atau interpretasi untuk digunakan dalam proses pengambilan keputusan. Sistem pengolahan informasi mengolah data menjadi informasi atau tepatnya pengolah data dari bentuk tidak berguna menjadi berguna bagi penerimanya (Sutabri, 2016 dalam Ayu dan Permatasari, 2018). Nilai informasi berhubungan dengan keputusan. Bila tidak ada pilihan atau keputusan, maka informasi menjadi tidak diperlukan (Ayu & Permatasari, 2018).

2.1.4 Data Mining

Data mining adalah suatu proses menganalisis pola data yang tersembunyi menurut berbagai perspektif sehingga menjadi informasi yang berguna. Informasi yang dihasilkan dapat memfasilitasi penggunaannya seperti pengambilan keputusan bisnis, perkiraan cuaca, maupun informasi lainnya. *Data mining* menggunakan analisis matematika untuk mendapat atau menemukan pola dan kecenderungan dari kumpulan data. Secara umum pola ini sulit ditemukan dengan eksplorasi data secara biasa/tradisional, hal ini disebabkan oleh terlalu rumitnya hubungan antar data maupun disebabkan oleh data yang diolah sangat besar (Arhami & Nasir, 2020).

Data mining merupakan sebuah istilah yang digunakan dalam menguraikan informasi baru yang bermanfaat dari hasil proses pengolahan menggunakan teknik statistik, matematika, kecerdasan buatan dan pembelajaran mesin terhadap data

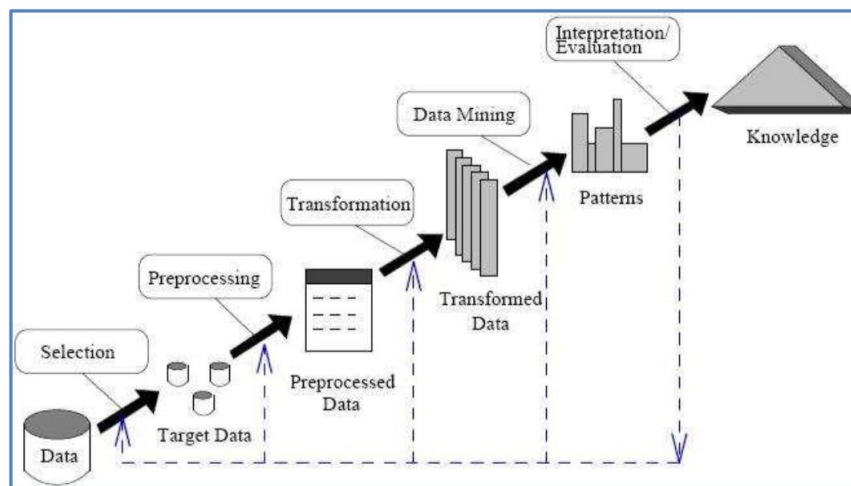
berjumlah besar. *Data mining* terbagi menjadi beberapa kelompok berdasarkan tugas yang dapat dilakukan, yaitu *classification*, *association*, *clustering*, *prediction*, *estimation* dan *description* (Kusrini & Emha Taufiq, 2009 dalam Takdirillah, 2020).

Secara sederhana *data mining* digambarkan sebagai suatu pola atau model atau kaidah atau pengetahuan yang dihasilkan dari *data mining*, sebagai berikut (Arhami dan Nasir, 2020).



Gambar 2.1 Model Pengetahuan Menggunakan Output *Data Mining*

2.1.5 Proses *Knowledge Discovery in Database (KDD)*



Gambar 2.2 Proses *Knowledge Discovery in Databases (KDD)*

(Karisto dan Winda, 2018)

Proses dari KDD dijabarkan sebagai berikut (Arhami dan Nasir, 2020).

1. Identifikasi Tujuan, merupakan tahap awal yang harus dilakukan dalam memahami domain atau daerah apa yang akan diambil dan tujuan apa yang ingin dicapai (target harus jelas).
2. Seleksi Data, dilakukan untuk melihat relevansi data yang berasal dari *database* dan disesuaikan dengan pengetahuan dari ahli yang terkait agar dicapai hasil yang sesuai dan tingkat keandalannya juga bagus.

3. *Preprocessing data*, tahap ini meliputi operasi untuk pembersihan data, seperti penanganan adanya gangguan terhadap data, serta penanganan data yang tidak konsisten, integrasi data dapat menggabungkan data yang sumber datanya berbeda, transformasi data dan reduksi data, termasuk pemilihan dan ekstraksi fitur.
4. *Data mining*, merupakan proses penting dimana metode tersebut digunakan untuk mengekstraksi pola data yang *valid*. Langkah ini termasuk pilihan strategi DM yang paling cocok, seperti klasifikasi, regresi, pengelompokan atau asosiasi.
5. Interpretasi atau evaluasi, merupakan tahapan untuk memperkirakan, mengidentifikasi, dan menafsirkan pola yang benar benar penting dan mewakili pengetahuan yang berbasis pada ukuran-ukuran derajat kepentingan.
6. Pengetahuan, merupakan tahapan terakhir yang dapat melibatkan pengetahuan secara langsung, kemudian menggabungkan pengetahuan kedalam sistem lain untuk proses lebih lanjut. Visualisasi dan representasi pengetahuan digunakan untuk menyampaikan dan melaporkan pengetahuan yang ditambang kepada pengguna.

2.1.6 Algoritma *Apriori*

Analisis dari kaidah asosiasi bertujuan untuk mendapatkan sekumpulan *item* atau *itemset* yang muncul secara bersamaan dalam kumpulan transaksi. Suatu *itemset* terdiri dari *i*-buah *item* yang disebut dengan *i-itemset*. Persentase transaksi atau kombinasi yang terdiri dari satu *itemset* disebut dengan *support itemset* dan *confidence* (nilai kepastian) merupakan rasio kuatnya hubungan antar item dalam kaidah asosiasi. *Support* dan *confidence* tersebut merupakan dua kriteria dasar dalam kaidah asosiasi (Arhami & Nasir, 2020).

Kaidah asosiasi atau *association rule* adalah teknik data mining untuk menemukan aturan asosiatif antara suatu kombinasi item dengan tujuan menemukan *frequent itemsets* yang terdapat pada sekumpulan data. Analisis apriori didefinisikan suatu

proses untuk menemukan semua aturan apriori yang memenuhi syarat *minimum* untuk *support* dan *confidence* (Purba & Ginting, 2018)

Pada tahap awal dalam algoritma apriori adalah mencari *frequent itemset* dengan menganalisis pola frekuensi tinggi yaitu dengan mencari kombinasi yang memenuhi syarat *minimum* dari nilai *support* dalam basis data atau *dataset* (Takdirillah, 2020).

Kaidah $X \Rightarrow Y$ dalam transaksi DB dengan *confidence* c jika $c\%$ transaksi dalam DB berisi X dan juga berisi Y . Selain itu, kaidah $X \Rightarrow Y$ memiliki *support* dalam transaksi DB jika $s\%$ transaksi dalam DB merupakan $X \cup Y$. Kaidah yang memiliki tingkat *confidence* dan nilai *support* yang tinggi dapat dikatakan bahwa kaidah tersebut merupakan kaidah aturan (*rule*) yang kuat (Arhami & Nasir, 2020).

Tahapan untuk perhitungan dalam kaidah asosiasi dapat dituliskan seperti berikut:

1. Untuk mencari nilai *support* sebuah item, digunakan persamaan berikut.

$$\text{Support (X)} = \frac{\text{jumlah transaksi mengandung X}}{\text{jumlah transaksi keseluruhan}}$$

2. Untuk mencari nilai *support* dari dua itemset digunakan persamaan berikut.

$$\text{Support (A} \Rightarrow \text{B)} = \frac{\text{jumlah transaksi mengandung A dan B}}{\text{jumlah transaksi keseluruhan}}$$

3. Untuk mencari nilai *support* dari dua *itemset* digunakan persamaan.

$$\text{Confidence} = \frac{\text{jumlah transaksi mengandung Anteseden dan Konsekuen}}{\text{jumlah transaksi mengandung Anteseden}}$$

Anteseden merupakan sebab yang menjadikan item *konsekuen*. Sedangkan *konsekuen* adalah sebuah akibat atau kesimpulan yaitu item yang akan dibeli setelah membeli *anteseden*. Jika didapatkan aturan asosiasi $A \Rightarrow B$, maka A sebagai *anteseden*, dan B sebagai *konsekuen*.

Sehingga rumus perhitungan *confidence* diatas dapat disederhanakan menjadi :

$$\text{Confidence (A} \Rightarrow \text{B)} = \frac{\text{jumlah transaksi mengandung A dan B}}{\text{jumlah transaksi mengandung A}}$$

Tahapan berikutnya yaitu pengujian aturan asosiasi yaitu pengujian dengan nilai *lift*. *Lift Ratio* adalah suatu ukuran untuk mengetahui kekuatan aturan asosiasi (*association rule*) yang telah terbentuk dari nilai support dan confidence pada setiap pola yang dihasilkan. Pengujian akurasi *lift ratio* biasa digunakan pada perhitungan *unsupervised learning* (Nengsih, 2019)

Pengujian akurasi nilai *lift ratio* dituliskan dengan persamaan sebagai berikut

$$Lift(X \Rightarrow Y) = \frac{Support\ X\ dan\ Y}{Support\ X * Support\ Y} \quad (4)$$

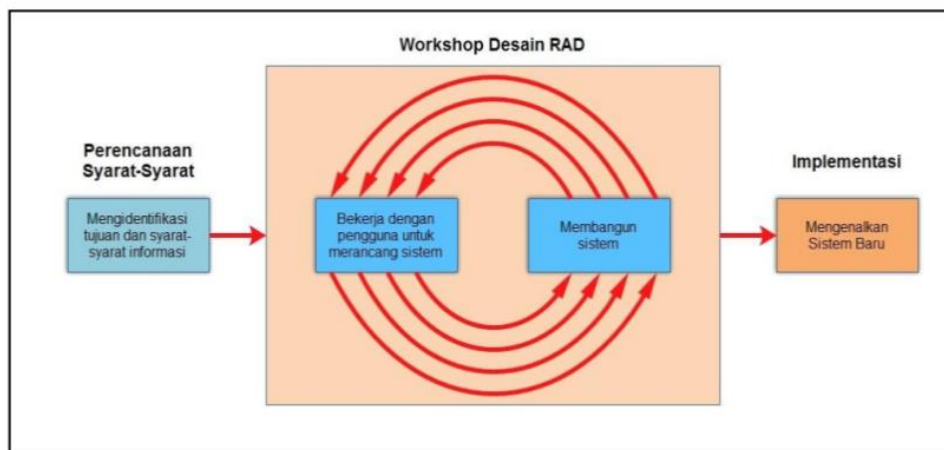
Adapun langkah-langkah dari algoritma apriori untuk mencari pola asosiasi pada data adalah sebagai berikut (Ulmer dan David, 2002).

1. Set $k=1$ (menunjuk pada *itemset* ke-1)
2. Hitung semua k -*itemset* (*itemset* yang mempunyai k *itemset*), untuk mendapat kandidat 1 -*itemset*.
3. Hitung support dari semua calon *itemset*, kemudian sortir *itemset* berdasarkan perhitungan minimum *support*, untuk mendapat *frequent itemset*.
4. Gabungkan semua *itemset* berukuran k untuk menghasilkan calon *itemset* $k+1$ atau kandidat k -*itemset*.
5. Set $k=k+1$
6. Ulangi langkah 2-5 hingga tidak ada *itemset* lebih besar yang dapat dibentuk.

2.2 Teori Pengembangan Sistem

2.2.1 Rapid Application Development (RAD)

Rapid Application Development (RAD) adalah proses model perangkat lunak inkremental yang menekankan siklus pengembangan yang singkat, yaitu dimana perkembangan pesat dapat dicapai dengan menggunakan pendekatan konstruksi berbasis komponen. Jika tiap-tiap kebutuhan dan batasan ruang lingkup proyek telah diketahui dengan baik, maka proses RAD dapat memungkinkan tim pengembang untuk menciptakan sebuah “sistem yang berfungsi penuh” dalam jangka waktu yang sangat singkat (Roger, S. Pressman, 2012).



(Sumber : Kendall, 2017)

Gambar 2.3 Proses dalam metode RAD

Berdasarkan tahapan diatas, terdapat tiga fase dalam RAD yang melibatkan penganalisis dan pengguna dalam tahap penilaian, perancangan, dan penerapan. Adapun ketiga fase tersebut adalah *requirements planning* yaitu perencanaan syarat-syarat), *RAD design workshop* (*workshop* desain RAD), dan *implementation* yaitu tahap implementasi (Kendall, 2010).

2.2.2 Data Flow Diagram (DFD)

Data Flow Diagram (DFD) adalah suatu model logika data atau proses yang dibuat lebih mendetail dibanding diagram konteks yang biasa digunakan. Data Flow Diagram dipresentasikan dalam bentuk lambang dan simbol tertentu yang menunjukkan aliran data, penyimpanan data, maupun proses yang dilakukan sistem (Harjanto et al., 2018).

Pada model *Data Flow Diagram* terdapat tiga tingkatan, yaitu:

1. Diagram Konteks

Diagram konteks adalah diagram yang terdiri dari suatu proses dan diagram yang menggambarkan ruang lingkup suatu sistem.

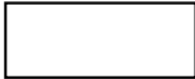
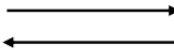
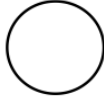
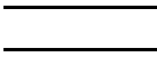
2. Diagram Nol/Zero

Diagram nol adalah diagram yang menggambarkan proses yang ada di dalam diagram konteks, yang penjabarannya lebih terperinci.

3. Diagram Rinci (Level Diagram)

Diagram rinci adalah diagram yang menguraikan proses apa yang ada dalam diagram zero atau diagram level di atasnya.

Tabel 2.1 Simbol DFD




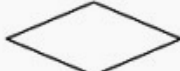


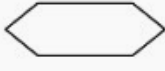



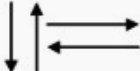
Nama	Simbol	Keterangan
<i>External Entity</i>		Simbol ini merupakan kesatuan di lingkungan luar sistem yang dapat berupa orang, organisasi atau sistem lainnya yang berada di lingkungan luarnya yang akan memberikan input atau menerima output dari sistem.
<i>Processing</i>		Simbol ini digunakan untuk menggambarkan aliran data dari satu proses ke proses lainnya.
<i>Data Flow</i>		Simbol ini digunakan untuk mentransformasikan data secara umum.
<i>Data Store</i>		Simbol ini digunakan untuk menyimpan data seperti: suatu file, suatu arsip, suatu kotak, suatu tabel dan suatu agenda

2.2.3 Diagram Alir (*Flowchart*)

Diagram Alir (*Flowchart*) merupakan bagan (*chart*) yang menunjukkan alir (*flow*) di dalam program atau prosedur sistem secara logika. Bagan alir digunakan terutama untuk alat bantu komunikasi dan untuk dokumentasi (Supardi, 2013).

Berikut simbol-simbol dari diagram alir atau *flowchart* :

Tabel 2.2 Simbol *Flowchart*








No.	Simbol	Nama	Fungsi
1		<i>Terminal</i>	Menyatakan permulaan atau akhir suatu program
2		<i>Input / Output</i>	Menyatakan proses input atau output tanpa tergantung jenis peralatannya
3		<i>Process</i>	Menyatakan suatu tindakan (proses) yang dilakukan oleh komputer
4		<i>Decision</i>	Menunjukkan suatu kondisi tertentu yang akan menghasilkan dua kemungkinan jawaban: ya / tidak
5		<i>Connector</i>	Menyatakan sambungan dari proses ke proses lainnya dalam halaman yang sama
6		<i>Offline Connector</i>	Menyatakan sambungan dari proses ke proses lainnya dalam halaman yang berbeda
7		<i>Predefined Process</i>	Menyatakan penyediaan tempat penyimpanan suatu pengolahan untuk memberi harga awal
8		<i>Punched Card</i>	Menyatakan input berasal dari kartu atau output ditulis ke kartu
9		<i>Punch Tape</i>	
10		<i>Document</i>	Mencetak keluaran dalam bentuk dokumen (melalui printer)
11		<i>Flow</i>	Menyatakan jalannya arus suatu proses

2.2.4 Entity Relationship Diagram (ERD)

Entity Relationship Diagram (ERD) adalah suatu model untuk menjelaskan hubungan antar data dalam basis data berdasarkan objek-objek data yang mempunyai hubungan antar relasi (Harjanto et al., 2018). ERD bisa juga digunakan untuk menunjukkan aturan-aturan bisnis yang ada pada system informasi yang akan dibangun (Hamdani & Ziveria, 2017).

Berikut simbol dari ER Diagram (Hamdani & Ziveria, 2017) :

Tabel 2.3 Simbol ERD

Gambar	Nama	Keterangan
	Entity	Objek diluar sistem yang berkomunikasi dengan sistem sebagai sumber maupun pemakai informasi
	Relationship line	Garis penghubung yang di gunakan pada diagram
	Option symbol	Di gunakan untuk relasi fungsional
	One Symbol	Di gunakan pada relasi menunjukkan 1(One)
	Many	Di gunakan pada relasi untuk menunjukkan banyak(many)
	Composite Entity	Menggambarkan sebuah entitas yang komposit
	Weak Entity	Entitas yang kemunculannya tergantung dari entitas lain yang lebih kuat

2.2.5 Pengujian Kotak Hitam (*Black Box Testing*)

Salah satu metode pengujian yang berfokus pada spesifikasi fungsionalitas dari perangkat lunak disebut *Black Box Testing* (Vikasari, 2018) Pengujian ini memberikan gambaran atas sekumpulan kondisi masukan dan melakukan pengujian pada uraian fungsional program. Untuk menguji kesalahan yang tidak dapat dicakup oleh *White Box Testing*, maka solusi lainnya dapat menggunakan *Black Box Testing* (Mustaqbal et al., 2015).

Pengujian *Black Box Testing* biasa digunakan untuk mendeteksi permasalahan berikut (Mustaqbal et al., 2015) :

1. Fungsi yang salah atau hilang.
2. Kesalahan pada interface.
3. Kesalahan struktur data dan basis data.
4. Kesalahan fungsi.
5. Kesalahan deklarasi dan terminasi.

Pengujian Black Box memiliki beberapa teknik, diantaranya *Equivalence Partitioning*, *Boundary Value Analysis*, *Robustness Testing*, *Behavior Testing*, dan *Cause-Effect Relationship Testing* (Safitri & Pramudita, 2018).

2.2.6 Android

Android adalah sistem operasi untuk perangkat *mobile* berbasis *Linux* yang awalnya dikembangkan oleh *Android Inc.* *Android* terdiri dari tiga bagian yaitu *operation system*, *middleware* dan *application*. *Android* adalah istilah dalam bahasa Inggris yang berarti robot yang menyerupai manusia. Pada tahun 2005, *Google* secara resmi telah membeli *Android*, sehingga pengembangan *Android* sepenuhnya berada di tangan *Google*. Dalam proses pengembangan sistem operasi *android*, dibentuklah organisasi *Open Handset Alliance*. *Google* kemudian merilis *software* berbasis *open source* untuk pengguna *android*, sehingga pengguna maupun pengembang aplikasi dapat berkontribusi untuk mengembangkan *android* (Firly, 2020).

2.2.7 Kotlin

Kotlin adalah bahasa pemrograman berbasis *Java Virtual Machine (JVM)*. *Kotlin* merupakan bahasa pemrograman yang pragmatis untuk *android* yang mengkombinasikan *object oriented* dan bahasa fungsional. *Kotlin* juga merupakan bahasa pemrograman *interoperabilitas*, dimana membuat bahasa ini dapat digabungkan dalam satu project dengan bahasa pemrograman *Java*.

Bahasa pemrograman ini juga dapat digunakan untuk pengembangan aplikasi berbasis *desktop*, *web* dan *backend* (Sibarani et al., 2018). *Kotlin* awalnya dikembangkan oleh *JetBrains*, perusahaan dibalik *IntelliJ IDEA*. Setelah melalui banyak perkembangan, *JetBrains* merilis *kotlin* secara *open source* dan kini perkembangannya semakin maju. *Google* mendukung penuh *kotlin* untuk pengembang aplikasi *android* (Aulia, 2019).

2.2.8 Android Studio

Android studio adalah IDE (*Integrated Development Environment*) resmi untuk pengembangan *android* dan bersifat *open source* atau gratis. Peluncuran *android Studio* ini diumumkan oleh *google* pada 16 mei 2013 pada *event google I/O conference* untuk tahun 2013. Sejak saat itu, *android studio* menggantikan *eclipse* sebagai IDE resmi untuk mengembangkan aplikasi *android* (Karman et al., 2019).

2.2.9 Java

Java merupakan bahasa pemrograman tingkat tinggi yang dapat diterapkan pada banyak *platform*. Bahasa pemrograman *java* mempunyai ciri sebagai bahasa yang sederhana, arsitektur netral berorientasi obyek, mempunyai kinerja yang tinggi, mempunyai *multithreaded*, kuat, dinamis dan aman (Febrianny Ulfha & Amin, 2020). *Java* mempunyai kemampuan dapat berjalan di banyak perangkat yang berbeda, seperti *Microsoft Windows*, *Linux*, *Solaris OS* dan *Mac OS*. Platform *java* mempunyai dua komponen yaitu *Java Virtual Machine* dan *Java Application Programming Interface* (API) (Febrianny Ulfha & Amin, 2020).

2.3 Penelitian Sebelumnya

Tabel 2.4 Penelitian Sebelumnya

NO	Judul Penelitian	Nama Penulis	Hasil Penelitian	Penerapan Dalam Penelitian Yang Dilakukan Saat Ini
1	Penerapan Data Mining Menggunakan Algoritma Apriori Terhadap Data Transaksi Sebagai Pendukung Informasi Strategi Penjualan.	Robby Takdirillah (2020).	Pola dari <i>Association Rule</i> pada data transaksi penjualan sembako dan minimarket dengan menggunakan <i>tools Orange</i> .	<ul style="list-style-type: none"> - Penerapan rumus perhitungan association rule - Teori dan pendapat peneliti yang digunakan pada tinjauan pustaka. - Nilai minimal transaksi pada penelitian adalah 53 transaksi dari 884 transaksi atau <i>support</i> 6%. - Nilai minimal <i>confidence</i> pada penelitian adalah 70%
2	Implementasi Algoritma Apriori Untuk Mencari Relasi Pada Transaksi Pembelian Alat-Alat Kesehatan (Studi Kasus: Rs. Estomihi).	Berta Rohana Br. Purba dan Guidio L Ginting (2018).	Pola dari <i>Association Rule</i> pada data transaksi penjualan alat kesehatan, yang diterapkan dalam aplikasi berbasis desktop.	<ul style="list-style-type: none"> - Penerapan format tampilan tabel untuk menampilkan hasil perhitungan algoritma apriori yang telah didapat. - Teori dan pendapat peneliti yang digunakan pada tinjauan pustaka. - Nilai minimal item pada penelitian adalah 3 dari 38 transaksi atau <i>support</i> 7%. - Nilai minimal <i>confidence</i> pada penelitian adalah 50%
3	Aplikasi Market Basket Analysis Dengan Metode Association Rule Untuk Menemukan Perilaku Konsumen Melalui Data transaksi.	Binti Umayah (2015).	Pola dari <i>Association Rule</i> pada data transaksi penjualan sembako dalam aplikasi berbasis desktop.	<ul style="list-style-type: none"> - Penerapan kaidah asosiasi dan algoritma apriori dalam logika komputer. - Teori dan pendapat peneliti yang digunakan pada tinjauan pustaka. - Nilai minimal transaksi pada penelitian adalah 15 transaksi dari atau <i>support</i> 1%. Nilai minimal <i>confidence</i> pada penelitian adalah 25%.

4	Analisis Pola Pembelian Konsumen Pada Transaksi Penjualan Menggunakan Algoritma Apriori.	Irsyad Djamaludin dan Agus Nursikuwagus (2017).	Pola dari <i>Association Rule</i> pada data transaksi penjualan bolu dan kue basah dalam aplikasi berbasis <i>website</i> .	<ul style="list-style-type: none"> - Teori dan pendapat peneliti yang digunakan pada tinjauan pustaka. - Nilai minimal transaksi pada penelitian adalah 7 transaksi dari 245 transaksi atau support 3%. - Nilai minimal <i>confidence</i> pada penelitian adalah 40%.
5	Implementasi Data Mining Untuk Mengetahui Pola Pembelian Obat Menggunakan Algoritma Apriori.	Nadya Febrianny Ulfha dan Ruhul Amin (2020).	Pola dari <i>Association Rule</i> pada data transaksi penjualan obat-obatan kesehatan.	<ul style="list-style-type: none"> - Teori dan pendapat peneliti yang digunakan pada tinjauan pustaka. - Nilai minimal transaksi pada penelitian adalah 36 transaksi dari 92 transaksi atau support 40%. - Nilai minimal <i>confidence</i> pada penelitian adalah 70%
6	Perancangan Sistem Informasi Pengolahan Data Praktek Kerja Lapangan (PKL) Pada Devisi Humas Pt. Pegadaian.	Fitri Ayu dan Nia Permatasari (2018).	Suatu sistem informasi pengolahan data Praktek Kerja Lapangan (PKL) berbasis <i>website</i> .	<ul style="list-style-type: none"> - Teori dan pendapat peneliti yang digunakan pada tinjauan pustaka.
7	Penerapan Data Mining Untuk Menghasilkan Pola Pembelian Roti Menggunakan Algoritma Apriori.	Nurul Hidayati, Handoyo Widi Nugroho dan Nurjoko (2021).	Suatu Pola dari <i>Association Rule</i> pada data transaksi penjualan roti dengan perhitungan <i>manual</i> dan perhitungan <i>Rapid Miner</i> .	<ul style="list-style-type: none"> - Teori dan pendapat peneliti yang digunakan pada tinjauan pustaka. - Nilai minimal transaksi pada penelitian adalah 8 transaksi dari 25 transaksi atau support 30% - Nilai minimal <i>confidence</i> pada penelitian adalah 60%.