

BAB II

TINJAUAN PUSTAKA

II.1. Penelitian Terkait

Adapun penelitian terkait yang akan digunakan sebagai sumber acuan yang relevan dan terkini yaitu:

Berdasarkan penelitian yang dilakukan oleh Erma Delima Sikumbang (2018). Hasil penelitian Data mining dan algoritma apriori sangat berguna untuk mengetahui hubungan frekuensi penjualan sepatu yang paling diminati oleh konsumen, sehingga dapat dijadikan sebagai informasi yang sangat berharga dalam pengambil keputusan untuk mempersiapkan stok jenis sepatu apa saja yang diperlukan dikemudian hari, Algoritma Apriori membantu mengembangkan strategi penjualan sepatu, Berdasarkan dari hasil penelitian, sepatu yang paling diminati adalah *New Balance* (91,67 %), *Adidas* (75 %) , *Geox* (50 %), *Nike* (41.67 %) dan *Palladium* (41.67 %).

Berdasarkan penelitian yang dilakukan oleh Mateus Paga Tana, dkk; (2018). Dari hasil penelitian yang dilakukan dapat diketahui hasil yaitu : Hubungan-hubungan keterkaitan barang yang satu dengan barang yang lainnya. Dan dari hubungan-hubungan keterkaitan tersebut digunakan untuk mengatur penempatan barang. Pengaturan penempatan barang dapat diketahui melalui nilai *support* dan nilai *confidence*. Barang-barang yang memiliki nilai support tinggi posisi penempatannya ditempatkan diawal/ujung karena barang-barang tersebut merupakan barang-barang yang paling sering dibeli oleh pembeli. Sedangkan

barang-barang yang memiliki nilai *confidence* tinggi diletakkan bersebelahan karena dengan tingginya nilai *confidence* antar kedua barang atau lebih memiliki kesempatan dibeli secara bersamaan yang tinggi. Penerapan Algoritma Apriori pada teknik Data Mining sangat efisien dan dapat mempercepat proses pembentukan kecenderungan pola kombinasi *itemset* hasil penjualan Produk-produk barang di Toko OASE, yaitu dengan *support* dan *confidence* tertinggi adalah Rokok, kopi *Snack*, mie goreng, dan waper Nabati. Barang-barang tersebut di letakan berdekatan di *etalase* barang.

Berdasarkan penelitian yang dilakukan oleh Sheih Al Syahdan, Anita Sindar; 2018. Berdasarkan perhitungan asosiatif *rule* apriori dapat diketahui produk apa saja yang laku terjual, sehingga perusahaan dapat menyusun strategi penjualan untuk meningkatkan penjualan dengan membuat paket-paket produk yang berisi kombinasi dari produk-produk yang diperoleh dari asosiasi item.

Berdasarkan penelitian yang dilakukan oleh Ade Fitria Lestari, M. Hafiz; (2020). Hasil didapatkan berdasarkan perhitungan manual dan menggunakan *software Rapid Miner* memiliki hasil yang sama yaitu 90 % begitu pula dengan hasil yang didapat sesuai dengan hasil penjualan. Algoritma apriori mempunyai pengaruh positif dalam laporan penjualan *Barbar Warehouse* karena pemilik dapat mengetahui produk yang perlu diperbanyak atau dikurangi stoknya guna meningkatkan *omset* penjualan.

Berdasarkan penelitian yang dilakukan oleh Golda TM Napitupulu, dkk; (2019). Berdasarkan pembahasan diatas, maka dapat diperoleh kesimpulan bahwa penelitian algoritma apriori dapat membantu mengembangkan strategi pemasaran

untuk memasarkan produk lain dengan menganalisa kelebihan dari nilai jual produk yang paling laris terjual. Hal tersebut dapat dilihat dari hasil penelitian yang menunjukkan bahwa penjualan produk pipa paling banyak terjual pada CV. Gaskindo Sentosa, dengan melihat produk yang memenuhi minimal *support* dan minimal *confidence*, produk yang banyak terjual tersebut adalah *Stud Anchor*, *Red Union*, *Stud Bolt*, *Union Tee*. Dari aturan asosiasi final yang diketahui jika membeli *Union Tee* maka akan membeli *Red Union* dengan *support* 50% dan *confidence* 87,5%. Jika membeli *Red Union* maka akan membeli *Union Tee* dengan *support* 50% dan *confidence* 77,78%. Jika membeli *Stud Anchor*, *Red Union* maka akan membeli *Union Tee* dengan *support* 33,33% dan *confidence* 83,33%. Jika membeli *Stud Anchor*, *Union Tee* maka akan membeli *Red Union* dengan *support* 33,33% dan *confidence* 83,33%.

II.2. Landasan Teori

II.2.1. Data Mining

Proses ekstraksi informasi dari kumpulan data melalui penggunaan algoritma dan teknik yang melibatkan bidang ilmu statistik, mesin pembelajaran dan sistem manajemen *database*. Data mining digunakan untuk ekstraksi informasi penting yang tersembunyi dari dataset yang besar. Adanya data mining akan didapatkan suatu permata berupa pengetahuan didalam kumpulan data-data yang banyak jumlahnya (Ade Fitria Lestari, M. Hafiz; 2020).

II.2.2. Metode Apriori

Algoritma apriori merupakan salah satu jenis aturan asosiasi pada data mining yang menerangkan asosiasi terhadap beberapa atribut sering disebut affinity analysis atau market basket analysis. Analisis asosiasi atau *association rule* mining adalah cara data mining untuk menemukan aturan suatu kombinasi item. Pada tahap analisis asosiasi, para peneliti tertarik untuk mendapatkan algoritma yang efisien adalah analisis pola frekuensi tinggi (*frequent pattern mining*). Tingkat kepentingan sebuah asosiasi dapat diketahui dengan dua tolak ukur, yaitu *support* dan *confidence*. *Support* (nilai penunjang) merupakan persentase kombinasi *item* tersebut dalam *database*, sementara *confidence* (nilai kepastian) merupakan kuatnya relasi antar-*item* dalam aturan asosiasi (Fauziah Nur Harahap, dkk; 2020).

II.2.3. Penjualan

Penjualan adalah proses dimana sang penjual memuaskan segala kebutuhan dan keinginan pembeli agar dicapai manfaatnya bagi yang penjual maupun sang pembeli yang berkelanjutan dan yang menguntungkan bagi kedua belah pihak. Penjualan juga hasil yang dicapai sebagai imbalan jasa – jasa yang diselenggarakan yang dilakukannya perniagaan transaksi dunia usaha (Reza Fahlevi Ahmad, Novrini Hasti; 2018).

II.2.4. PHP

PHP adalah bahasa yang dirancang secara khusus untuk penggunaan pada *Web*. *PHP* adalah *tool* untuk pembuatan halaman web dinamis. 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 *FI (Form Interpreted)*, yang wujudnya berupa sekumpulan *script* yang digunakan untuk mengolah data *form* dari *web*. Saat ini *PHP* adalah singkatan dari *PHP: Hypertext Preprocessor*, sebuah kepanjangan rekursif, yakni permainan kata dimana kepanjangannya terdiri dari singkatan itu sendiri: *PHP: Hypertext Preprocessor* (Ahmad Lutfi, 2017 ; 105).

II.2.5. MySQL

MySQL adalah salah satu aplikasi *DBMS (Database Management System)* yang sudah sangat banyak digunakan oleh para pemrogram aplikasi *web*. Dalam sistem *database* tak relasional, semua informasi disimpan pada satu bidang luas, yang kadangkala data di dalamnya sangat sulit dan melelahkan untuk diakses. Tetapi *MySQL* merupakan sebuah sistem database relasional, sehingga dapat mengelompokkan informasi ke dalam tabel-tabel atau grup-grup informasi yang berkaitan. Setiap tabel memuat bidang-bidang yang terpisah, yang mempresentasikan setiap *bit* informasi. *MySQL* menggunakan indeks untuk mempercepat proses pencarian terhadap baris informasi tertentu. *MySQL* memerlukan sedikitnya satu indeks pada tiap tabel. Biasanya akan

menggunakan suatu *primary key* atau pengenal unik untuk membantu penjejukan data (Ahmad Lutfi, 2017; 106).

II.2.6. Database

Database atau biasa disebut basis data merupakan kumpulan data yang saling berhubungan. Data tersebut biasanya terdapat dalam tabel-tabel yang saling berhubungan satu sama lain, dengan menggunakan *field*/kolom pada tiap tabel yang ada” (Agus Prayitno dan Yulia Safitri, 2015; 2).

II.2.7. Unified Modeling Language (UML)

Hasil pemodelan pada *OOAD* terdokumentasikan dalam bentuk *Unified Modeling Language* (UML). UML adalah satu alat bantu yang sangat handal didunia pengembangan sistem yang berorientasi objek. Hal ini disebabkan karena *UML* menyediakan bahasa pemodelan *visual* yang memungkinkan bagi pengembang sistem untuk membuat cetak biru atas visi mereka dalam membentuk yang baku, mudah dimengerti serta dilengkapi dengan mekanisme yang efektif untuk berbagi (*sharing*) dan mengkomunikasikan rancangan mereka dengan yang lain (Munawar ; 2018 : 49).

Alat bantu yang digunakan dalam perancangan berorientasi objek berbasis *UML* adalah sebagai berikut :

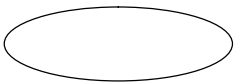
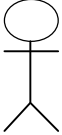


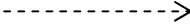
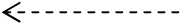
1. *Use case* Diagram

Use case diagram merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use case* mendeskripsikan tipikal interaksi antara (pengguna) sebuah *system* dengan sistemnya sendiri melalui sebuah cerita

bagaimana sebuah sistem yang dipakai (Munawar ; 2018 : 89).

Untuk menganalisis sistem yang berjalan, pada penelitian ini digunakan *UML (Unified Modeling Language)* untuk menggambarkan prosedur dan proses yang berjalan saat ini, adapun *tools* beserta keterangan *Use Case Diagram* yang digunakan dapat dijelaskan sebagai berikut:

Tabel II.1. Simbol *Use Case Diagram*

Gambar	Keterangan
	<p><i>Use Case</i> menggambarkan fungsionalitas yang disediakan sistem sebagai unit-unit yang bertukar pesan antar unit dengan aktor, biasanya dinyatakan dengan menggunakan kata kerja di awal nama <i>Use Case</i>.</p>
	<p>Aktor adalah <i>abstraction</i> dari orang atau sistem yang lain yang mengaktifkan fungsi dari target sistem. Untuk mengidentifikasi aktor, harus ditentukan pembagian tenaga kerja dan tugas-tugas yang berkaitan dengan peran pada konteks target sistem. Orang atau sistem bisa muncul dalam beberapa peran. Perlu dicatat bahwa aktor berinteraksi dengan <i>Use Case</i>, tetapi tidak memiliki <i>control</i> terhadap <i>Use Case</i>.</p>
	<p>Asosiasi antara aktor dan <i>Use Case</i>, digambarkan dengan garis tanpa panah yang mengindikasikan siapa atau apa yang meminta interaksi secara langsung dan bukannya mengindikasikan aliran data.</p>
	<p>Asosiasi antara aktor dan <i>Use Case</i> yang menggunakan panah terbuka untuk mengindikasikan bila aktor berinteraksi secara pasif dengan sistem.</p>
	<p><i>Include</i>, merupakan di dalam <i>Use Case</i> lain (<i>required</i>) atau pemanggilan <i>Use Case</i> oleh <i>Use Case</i> lain, contohnya adalah pemanggilan sebuah fungsi program.</p>
	<p><i>Extend</i>, merupakan perluasan dari <i>Use Case</i> lain jika kondisi atau syarat terpenuhi.</p>

(Sumber : Munawar ; 2018 : 93)

2. *Class Diagram* (Diagram Kelas)

Merupakan hubungan antar kelas dan penjelasan detail tiap-tiap kelas di dalam model desain dari suatu sistem, juga memperlihatkan aturan-aturan dan tanggung jawab entitas yang menentukan perilaku sistem.

Class diagram merupakan diagram statis dari suatu aplikasi. *Class Diagram* tidak hanya digunakan untuk memvisualisasikan, menggambarkan, dan mendokumentasikan berbagai aspek sistem tetapi juga untuk membangun kodeeksekusi (*executable code*) dari aplikasi perangkat lunak (Munawar ; 2018 : 101).

Untuk menganalisis sistem yang berjalan, pada penelitian ini digunakan *UML (Unified Modeling Language)* untuk menggambarkan prosedur dan proses yang berjalan saat ini, adapun *tools* beserta penjelasan *Class Diagram* yang digunakan dapat dijelaskan sebagai berikut:

Tabel II.2. Simbol *Class Diagram*

<i>Multiplicity</i>	Penjelasan
1	Satu dan hanya satu
0..*	Boleh tidak ada atau 1 atau lebih
1..*	1 atau lebih
0..1	Boleh tidak ada, maksimal 1
n..n	Batasan antara. Contoh 2..4 mempunyai arti minimal 2 maksimum 4




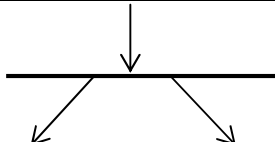
(*Sumber : Munawar ; 2018 : 101*)

3. Diagram Aktivitas (*Activity Diagram*)

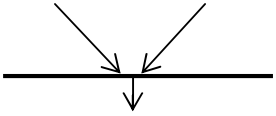
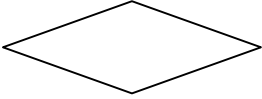

Activity Diagram bagian penting dari *UML* yang menggambarkan aspek dinamis dari sistem. Logika prosedural, proses bisnis dan aliran kerja suatu bisnis bisa dengan mudah dideskripsikan dalam *activity diagram* (Munawar ; 2018 : 137).

Untuk menganalisis sistem yang berjalan, pada penelitian ini digunakan *UML* (*Unified Modeling Language*) untuk menggambarkan prosedur dan proses yang berjalan saat ini, adapun *tools* beserta keterangan *Activity Diagram* yang digunakan dapat dijelaskan sebagai berikut:

Tabel II.3. Simbol Diagram Aktivitas

Simbol	Keterangan
	<i>Start point</i> , diletakkan pada pojok kiri atas dan merupakan awal aktifitas.
	<i>End point</i> , akhir aktifitas.
	<i>Activites</i> , menggambarkan suatu proses/kegiatan bisnis.
	<i>Fork</i> (Percabangan), digunakan untuk menunjukkan kegiatan yang dilakukan secara parallel atau untuk menggabungkan dua kegiatan pararel menjadi satu.

Tabel II.3. Simbol Diagram Aktivitas (Lanjutan)

Simbol	
	<p><i>Join</i> (penggabungan) atau <i>merge</i>, digunakan untuk menunjukkan adanya dekomposisi.</p>
	<p><i>Decision Points</i>, menggambarkan pilihan untuk pengambilan keputusan, <i>true, false</i>.</p>
	<p><i>Swimlane</i>, pembagian <i>activity</i> diagram untuk menunjukkan siapa melakukan apa.</p>

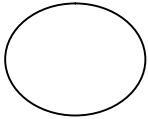
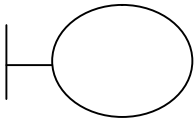
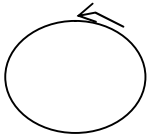
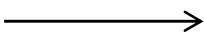
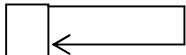


(Sumber : Munawar ; 2018 : 137)

4. Diagram Urutan (*Sequence Diagram*)

Sequence diagram adalah salah satu *interaction diagram*. Karena *sequence diagram* mengacu kepada obyek, maka sbelum membuat diagram ini class diagram sudah harus teridentifikasi (Munawar ; 2018 : 186).

Untuk menganalisis sistem yang berjalan, pada penelitian ini digunakan *UML (Unified Modeling Language)* untuk menggambarkan prosedur dan proses yang berjalan saat ini, adapun *tools* beserta keterangan *Sequence Diagram* yang digunakan dapat dijelaskan sebagai berikut:

Tabel II.4. Simbol Diagram Urutan

Gambar	Keterangan
	<p><i>EntityClass</i>, merupakan bagian dari sistem yang berisi kumpulan kelas berupa entitas-entitas yang membentuk gambaran awal sistem dan menjadi landasan untuk menyusun basis data.</p>
	<p><i>Boundary Class</i>, berisi kumpulan kelas yang menjadi <i>interface</i> atau interaksi antara satu atau lebih aktor dengan sistem, seperti tampilan <i>formentry</i> dan <i>form</i> cetak.</p>
	<p><i>Control class</i>, suatu objek yang berisi logika aplikasi yang tidak memiliki tanggung jawab kepada entitas, contohnya adalah kalkulasi dan aturan bisnis yang melibatkan berbagai objek.</p>
	<p><i>Message</i>, simbol mengirim pesan antar <i>class</i>.</p>
	<p><i>Recursive</i>, menggambarkan pengiriman pesan yang dikirim untuk dirinya sendiri.</p>
	<p><i>Activation</i>, mewakili sebuah eksekusi operasi dari objek, panjang kotak ini berbanding lurus dengan durasi aktivitas sebuah operasi.</p>
	<p><i>Lifeline</i>, garis titik-titik yang terhubung dengan objek, sepanjang <i>lifeline</i> terdapat <i>activation</i>.</p>

(Sumber : Munawar ; 2018 : 186)