

BAB II

LANDASAN TEORI

2.1 *Online Shop*

Berikut ini adalah beberapa pengertian *Online Shop* atau *e-commerce* dari beberapa sudut pandang :

1. Dalam sudut pandang komunikasi, *Online Shop* adalah sebuah *transfer* informasi baik itu informasi produk, layanan ataupun pembayaran melalui jaringan suatu komputer atau sarana elektronik lainnya.
2. Dalam sudut pandang proses bisnis, *Online Shop* adalah suatu teknologi aplikasi yang mengarah ke otomatisasi aliran kerja suatu perusahaan dan transaksi.
3. Dalam sudut pandang layanan, *Online Shop* adalah suatu alat yang dapat memenuhi ambisi suatu perusahaan, Pembeli dan manajemen dalam hal memotong biaya layanan dalam meningkatkan mutu barang dan kecepatan dalam suatu pelayanan..
4. Dalam sudut pandang *online*, *Online Shop* berkaitan dalam suatu daya kapasitas transaksi produk dan informasi di *internet*.

(Ravi Kalakota dan Andrew Whinston, 2018)

2.2 Rekomendasi

Rekomendasi merupakan suatu program yang mencoba memprediksikan suatu item baik itu berupa buku, lagu, film dan lain – lain berdasarkan suatu informasi yang diperoleh dari penggunaanya. (Albertus Bayu Aji Priyono, 2016)

2.3 Produk

Produk merupakan suatu hal yang dapat ditawarkan konsumen baik itu berupa benda berwujud atau pun jasa yang dapat diterima dengan baik oleh pembeli yang dapat memuaskan kebutuhan konsumen. (Riyono dan Gigih Erlik Budiharja, 2016)

2.4 Data Mining

Pengertian singkat dari *Data Mining* merupakan suatu intisari dari suatu informasi ataupun pola penting dari suatu data yang terdapat dalam suatu *database* berukuran besar. Selain itu, dalam sudut pandang jurnal ilmiah, *Data Mining* sering disebut juga sebagai *Knowledge Discovery in Database*. (Amri Mutio Siregar dan Adam Puspabhuana, 2017)

2.5 Market Basket Analysis

Market Basket Analysis merupakan suatu cara yang bekerja dalam menemukan dan menentukan prediksi pola hubungan antara produk, misalnya untuk menemukan kebiasaan Pembeli dalam membeli suatu produk dimana biasanya produk A dapat dibeli bersamaan dengan produk B. Menganalisis suatu keranjang belanja Pembeli dari data transaksi penjualan dapat mempermudah dalam mengetahui kebiasaan Pembeli dalam membeli suatu produk dalam waktu bersamaan. (Elisa, 2018)

2.6 Association Rules

Association Rules Mining atau sering disebut juga dengan Analisis Asosiasi merupakan salah satu metode *Data Mining* yang sering diterapkan untuk mendapatkan aturan asosiasi dari kombinasi atau relasi suatu produk. Dari analisis tersebut didapatkan hasil bahwa suatu pola yang mendefinisikan suatu hubungan antara fitur pada suatu data secara lebih efisien dan menerangkan bentuk aturan implikasi

Dalam hasil akhir analisis asosiasi menjelaskan mengenai nilai *Support* dan nilai *Confidence* dari setiap relasi 2 *itemsets*. Untuk mencari nilai *Support* sebuah *item* (A) dengan cara :

$$\text{Support } A = \frac{\text{Jumlah Transaksi Mengandung } A}{\text{Total Transaksi}} \times 100$$

Sedangkan untuk mencari nilai *Support itemset* (A, B) dengan cara :

$$\text{Support } A, B = \frac{\text{Jumlah Transaksi Mengandung } A \text{ dan } B}{\text{Total Transaksi}} \times 100$$

Sementara, untuk mencari nilai *Confidence* didapatkan dengan cara :

$$\text{Confidence } A, B = \frac{\text{Jumlah Transaksi Mengandung } A \text{ dan } B}{\text{Total Transaksi } A} \times 100$$

(Feresia Panjaitan, Ade Surahman, Tri Rosmalasari, 2020)

2.7 Algoritma

Dalam merancang suatu program aplikasi tentunya pembuat harus menentukan terlebih dahulu mengenai *input* dan *output* yang diterima oleh aplikasi tersebut nantinya. Dari *input* tersebut nantinya akan diolah menjadi *output* yang diinginkan dan dapat mempermudah pengguna aplikasi dalam

berinteraksi dengan aplikasi tersebut. Alur proses yang sistematis tentang jalanya suatu aplikasi dari awal hingga akhir disebut dengan algoritma.

(Winda Swastika, 2018)

2.7.1 Algoritma FP-Growth

Frequent Pattern Growth atau sering kita kenal dengan Algoritma FG-Growth adalah salah satu cara dalam analisa asosiasi. Langkah awal dalam metode ini adalah dengan menghitung kumpulan *itemset* dengan jumlah kemunculan *itemset* tersebut didalam suatu *dataset*. Selanjutnya, dari data tersebut akan dibuatkan struktur *tree*. Struktur *tree* merupakan suatu cara pengambilan data yang menyerupai bentuk dari sebuah pohon, yang didalamnya terdapat serangkaian *node* yang saling berhubungan. Setiap *node* bisa terdiri dari 0 atau lebih *node* yang terhubung dengan sebuah *vector*. Apabila terdapat *item* yang kurang dari batas maksimum, maka *item* tersebut akan dikeluarkan dari *dataset*.

Dalam proses Algoritma FP-Growth digunakan suatu struktur data berbentuk pohon yang disebut dengan *FP-Tree*. Dalam mencari suatu *itemset* yang *frequent*. Algoritma FP-Growth menggunakan tiga langkah ini sebagai berikut :

1. *Conditional Pattern Base*

Conditional Pattern Base merupakan suatu tahapan yang didapatkan dari *FP-Tree* yang dimana telah dibangun sebelumnya.

2. *Conditional FP-Tree*

Jumlah *Support* dari setiap produk yang terdapat pada setiap *Conditional Pattern Base* dijumlahkan. Kemudian, produk yang memiliki nilai jumlah

Support yang sama atau lebih besar akan dibangkitkan dengan *Conditional FP-Tree*.

3. *Frequent Itemset*

Jika *Conditional FP-Tree* termasuk satu lintasan tunggal, maka akan memperoleh *Frequent Itemset* dengan cara melakukan relasi produk dalam setiap *Conditional FP-Tree*. Apabila tidak termasuk satu lintasan tunggal, harus melakukan pembangkitan FP-Growth secara rekursif atau memanggil dirinya sendiri.

Dalam hasil akhir FP-Growth menjelaskan mengenai nilai *Support* dan nilai *Confidence* dari setiap relasi 2 *itemsets*. Untuk mencari nilai *Support itemset* (A, B) dengan cara :

$$\text{Support } A, B = \frac{\text{Jumlah Transaksi Mengandung } A \text{ dan } B}{\text{Total Transaksi}} \times 100$$

Sementara, untuk mencari nilai *Confidence* didapatkan dengan cara :

$$\text{Confidence } A, B = \frac{\text{Jumlah Transaksi Mengandung } A \text{ dan } B}{\text{Total Transaksi } A} \times 100$$

(Indah Werdiningsih, Barry Nuqoba, Muhammadun, 2020)

2.7.2 **Algoritma Apriori**

Dalam *Data Mining* Algoritma Apriori merupakan salah satu metode yang paling terkenal dalam mencari pola kemunculan data. Biasanya algoritma ini digunakan untuk mencari kebiasaan pola pembelian Pembeli pada setiap transaksi. Pentingnya suatu asosiasi dapat diketahui dengan *Support* dan *Confidence*, dimana nilai *Support* merupakan *persentase* suatu relasi produk

tersebut dalam *database*, sedangkan nilai *Confidence* merupakan kuatnya suatu relasi antar produk didalam aturan asosiasi.

Misalkan sebagai contoh kasus disebuah minimarket ada seorang Pembeli yang membeli sebuah kopi bersamaan dengan membeli sebuah cemilan. Dari informasi tersebut, pemilik toko dapat mengetahui pola kebiasaan dari pembeli tersebut dan mengatur penempatan rak kedua barang tersebut menjadi lebih berdekatan.

Dalam hasil akhir Algoritma Apriori menjelaskan mengenai nilai *Support* dan nilai *Confidence* dari setiap relasi 2 *itemsets*. Untuk mencari nilai *Support* sebuah *item* (A) dengan cara :

$$\text{Support } A = \frac{\text{Jumlah Transaksi Mengandung } A}{\text{Total Transaksi}} \times 100$$

Sedangkan untuk mencari nilai *Support itemset* (A, B) dengan cara :

$$\text{Support } A, B = \frac{\text{Jumlah Transaksi Mengandung } A \text{ dan } B}{\text{Total Transaksi}} \times 100$$

Sementara, untuk mencari nilai *Confidence* didapatkan dengan cara :

$$\text{Confidence } A, B = \frac{\text{Jumlah Transaksi Mengandung } A \text{ dan } B}{\text{Total Transaksi } A} \times 100$$

(Efori Buulolo, 2020)

2.8 *Internet dan Aplikasi Web*

2.8.1 *Internet*

Pada dasarnya *internet* merupakan suatu jaringan dunia yang dapat menghubungkan setiap komputer diseluruh dunia. Dengan adanya *internet*, sebuah komputer dapat memperoleh data dari komputer lain, baik itu dalam jarak yang cukup dekat ataupun jarak yang mencangkup antar negara bahkan benua.

(Priyanto Hidayatulloh dan Jauhari Khairul Kawistara, 2014)

2.8.2 Website

Website merupakan suatu sekumpulan halaman – halaman yang didalamnya terdapat informasi berupa *text* ataupun media baik itu animasi, gambar, suara, video ataupun gabungan dari semuanya yang berbasis data digital yang dapat diakses melalui koneksi *internet*. Sehingga, halaman tersebut dapat dilihat oleh orang banyak diseluruh dunia. Halaman *website* tersebut dibuat dengan menggunakan bahasa pemrograman standar seperti HTML, CSS, JAVA SCRIPT dan PHP yang nantinya akan diterjemahkan oleh *web browser* untuk menampilkan informasi.

(Rohi Andulloh, 2018)

2.9 Tools Aplikasi Web

2.9.1 HTML

Hypertext Markup Language atau yang sering kita kenal dengan singkatan HTML merupakan bahasa pemrograman standar berupa tag – tag penyusun setiap elemen dari sebuah *web* yang dikelola penggunanya oleh *World Wide Web Consortium* (W3C). HTML dapat diletakan sesuai keinginan dalam mengatur setiap elemen *web*. Aplikasi *text* editor yang paling standar untuk *skrip* HTML adalah Notepad yang merupakan bawaan dari komputer. Selain Notepad kita bisa menggunakan *text* editor lainnya seperti Visual Studio Code, Sublime, Notepad ++ dan masih banyak yang lainnya. *File* HTML biasanya disimpan dengan format *.html*. (Rohi Andulloh, 2018)

2.9.2 CSS

Cascading Style Sheet atau yang sering kita kenal dengan singkatan CSS merupakan bahasa pemrograman yang bertugas untuk mengatur *style* atau gaya tampilan elemen pada HTML seperti yang kita inginkan. Banyak orang yang beranggapan bahwa CSS bukan termasuk kedalam bahasa pemrograman, karena bentuk struktur yang terlalu sederhana. CSS berkerja dengan cara memilih element HTML yang disiapkan untuk kemudian nantinya akan diberikan properti *style* atau gaya tampilan sesuai yang kita inginkan. Didalam css terbagi menjadi 3 bagian yaitu pertama untuk memilih elemen untuk menentukan aturan, kedua *property* ialah aturan yang diberikan dan yang terakhir adalah sebagai nilai dari suatu aturan. (Rohi Andulloh, 2018)

2.9.3 PHP

PHP : Hypertext Preprocessor atau PHP merupakan bahasa pemrograman yang disisipkan kedalam *skrip* HTML dan berkerja bersama *server*. Tujuan utama dari bahasa pemrograman ini adalah untuk membantu perancang aplikasi dalam mengembangkan *web* supaya menjadi lebih cepat dan dinamis saat digunakan.

Sebelum menggunakan PHP, terlebih dahulu harus mempersiapkan *Software* sebagai berikut :

1. *Server Web* : Apache, IIS, Personal *Web Server* dan lain – lain.
2. *Server PHP*.
3. *Server Database* : MYSQL, Interbase, MS SQL dan lain – lain.

Tidak perlu menginstal satu persatu, karena didalam aplikasi Xampp ataupun Appserv sudah tersedia menjadi satu paket aplikasi yang siap digunakan. (Rohi Andulloh, 2018)

2.9.4 JavaScript

JavaScript ialah bahasa pemrograman *web* yang berkerja disamping klien. Oleh karena itu JavaScript bisa dijalankan walaupun hanya dengan menggunakan *browser web* saja. Ketika JavaScript dijalankan maka akan beraneka ragam perintah tertentu yang terjadi dihalaman *web*. Baik itu perintah yang dilakukan klien ataupun secara otomatis dilakukan *web*. (Rohi Andulloh, 2018)

2.9.5 JQuery

JQuery adalah salah satu perpustakaan dari bahasa pemrograman JavaScript. JQuery biasanya berisikan kumpulan fungsi JavaScript yang sudah siap digunakan, sehingga memudahkan klien dalam membuat *script* JavaScript. JQuery dapat mengubah bahasa pemrograman JavaScript yang panjang menjadi hanya satu baris kode saja. Sebelum menggunakan JQuery, alangkah lebih baiknya jika memiliki pengetahuan mengenai HTML, JavaScript dan CSS terlebih dahulu. (Rohi Andulloh, 2018)

2.9.6 Bootstrap

Bootstrap merupakan *framework* dari bahasa pemrograman CSS yang paling populer. Tampilan *design web* dengan menggunakan bootstrap dapat menjadi lebih responsif dan mudah dibuka diberbagai jenis ukuran dengan *design* yang

menarik. Selain itu dengan bootstrap dapat membuat *script* CSS menjadi lebih mudah dan cepat. Bootstrap sudah *Support* diberbagai jenis *browser* baik itu *desktop* atau pun *mobile*. (Rohi Andulloh, 2018)

2.9.7 XAMPP

XAMPP (“X (Windows / Linux) Apache MYSQL PHP dan Perl”) ialah suatu aplikasi yang terdiri dari paket *server web* dan *database*, dikalangan klien *web* yang paling populer adalah menggunakan *web server* PHP dan *database* MYSQL. Xampp bisa digunakan pada operating sistem windows, linux dan macos. XAMPP sangat mudah digunakan untuk membantu pengembang dalam membuat aplikasi *web*. Pembaruan aplikasi yang rutin membuat XAMPP sangat bagus untuk dipilih. (Betha Sidik, 2014)

2.10 Database dan Mysql

2.10.1 Database

Berikut ini adalah beberapa pengertian basis data atau *database* :

1. Suatu sekumpulan data yang saling terhubung sedemikian rupa, sehingga dapat digunakan dengan mudah.
2. Suatu sekumpulan data yang terhubung dan disimpan secara bersamaan tanpa melalui pengulangan.
3. Suatu sekumpulan *file*, tabel, atau arsip yang saling terhubung dan disimpan didalam suatu penyimpanan eletronik. (Uus Rusmawan, 2019)

2.10.2 MySQL

MySQL merupakan suatu *Software DBMS* yang *multiuser, multithread* dengan instalasi aplikasi 6 jutaan diseluruh dunia. MySQL AB selaku *Software company* membuat MySQL untuk *Support* diberbagai jenis perangkat dan dibagikan secara gratis, akan tetapi walaupun terdapat MySQL yang gratis MySQL AB tetap membuat MySQL untuk dijual secara lisensi komersial untuk kasus – kasus tertentu apabila klien tidak suka dengan menggunakan GPL yang terdapat pada *Software* gratisnya. (Uus Rusmawan, 2019)

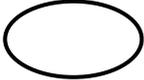
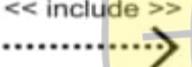
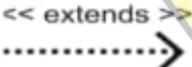
2.11 UML

2.11.1 Use Case Diagram

Use Case Diagram merupakan suatu teknik yang digunakan untuk merekam fungsional dalam sebuah sistem. *Use case* menggambarkan suatu hubungan antara pengguna dengan sistem, dengan memberikan suatu narasi mengenai bagaimana suatu sistem tersebut akan digunakan nantinya oleh pengguna. Dalam *Use Case Diagram* terdapat dua hal penting yaitu skenario dan aktor. Skenario ialah suatu rangkaian tahapan – tahapan yang menggambarkan suatu hubungan antara pengguna dengan aplikasi tersebut, sedangkan aktor ialah gambaran dari pengguna itu sendiri. (Uus Rusmawan, 2019)

Berikut ini adalah beberapa simbol *Use Case Diagram* beserta keteranganya terdapat :

Tabel 2.1. Tabel Deskripsi Simbol *Use Case Diagram* (Uus Rusmawan, 2019)

Simbol	Nama	Keterangan
	<i>Actor</i>	Seseorang yang berinteraksi dengan suatu sistem yang sedang dibuat atau dijalankan.
	<i>Use Case</i>	Mendefinisikan mengenai seseorang dalam menggunakan sistem tersebut.
	Relasi Asosiasi	Suatu garis penghubung yang digunakan untuk menunjukkan relasi atau interaksi antara dengan <i>use case</i> .
	Relasi <i>Include</i>	Memungkinkan <i>use case</i> dalam menggunakan suatu fungsional yang telah disediakan oleh <i>use case</i> .
	Relasi <i>Extend</i>	Memungkinkan <i>use case</i> secara optimal dalam menggunakan suatu fungsional yang telah disediakan oleh <i>use case</i> .

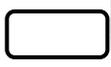
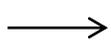
2.11.2 Activity Diagram

Activity Diagram merupakan suatu rangkaian aktivitas yang menunjukkan suatu alur kerja dari awal kejadian hingga akhir keputusan dengan merinci didalam perkembangan suatu proses atau peristiwa yang berada didalam suatu aktivitas tersebut. *Activity diagram* memiliki bentuk tertentu didalam komponennya yang dihubungkan dengan suatu tanda panah. Panah mengarah

sesuai tahapan – tahapan aktivitas dari awal kejadian hingga akhir keputusan.
(Uus Rusmawan, 2019)

Berikut ini adalah beberapa simbol *Activity Diagram* beserta keterangannya terdapat :

Tabel 2.2. Tabel Deskripsi Simbol *Activity Diagram* (Uus Rusmawan, 2019)

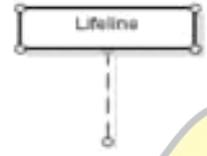
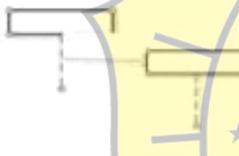
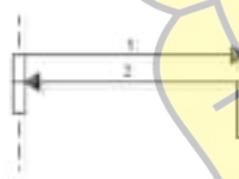
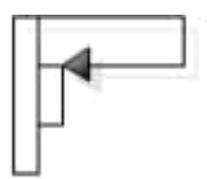
Simbol	Nama	Keterangan
	<i>Start State</i>	Merupakan bentuk suatu titik awal dalam suatu aktivitas.
	<i>End State</i>	Merupakan bentuk suatu titik akhir keputusan dari aktivitas tersebut.
	<i>Activity</i>	Menggambarkan suatu aktivitas yang dilakukan <i>actor</i> tersebut.
	<i>Decision</i>	Menggambarkan untuk menentukan keputusan dalam suatu aktivitas.
	<i>Interaction</i>	Menggambarkan jalur penghubung antar aktivitas yang mengarah.

2.11.3 Sequence Diagram

Sequence Diagram merupakan suatu susunan mengenai urutan waktu tertentu. *Sequence Diagram* dapat menggambarkan satu tugas dari beberapa tugas didalam suatu *use case* secara lebih *detail*. (Uus Rusmawan, 2019)

Berikut ini adalah beberapa simbol *Sequence Diagram* beserta keteranganya terdapat :

Tabel 2.3. Tabel Deskripsi Simbol *Sequence Diagram* (Uus Rusmawan, 2019)

Simbol	Nama	Keterangan
	<i>Actor</i>	Menggambarkan pengguna yang akan melakukan aktivitas tersebut.
	<i>Lifeline</i>	Menggambarkan suatu objek didalam suatu sistem atau komponennya.
	<i>Create Message</i>	Membuat suatu pesan sederhana antar elemen dan melakukan interaksi antar suatu objek.
	<i>Synchronous Message</i>	Pesan yang mengaktifkan suatu proses hingga sampai, baru bisa melakukan pengiriman sebuah pesan baru..
	<i>Message to self</i>	Menggambarkan sebuah hasil kembalian dari suatu operasi dan berjalan kepada objek itu sendiri..