

BAB II

TINJAUAN PUSTAKA

2.1 Sistem Informasi

2.1.1 Definisi Sistem

Sistem adalah sekelompok komponen dan elemen yang bersatu sebagai satu kesatuan untuk mencapai tujuan tertentu. Sistem berasal dari bahasa Latin (*systema*) dan bahasa Yunani (*systema*), yang merupakan suatu kesatuan yang terdiri dari komponen-komponen atau unsur-unsur yang dihubungkan bersama untuk memperlancar suatu informasi, materi, dan energi untuk mencapai suatu tujuan. Istilah ini sering digunakan untuk menggambarkan sekumpulan entitas yang saling berinteraksi, yang sering kali menjadi model matematika yang dapat dibangun. (Simangunsong et al., 2018).

Oleh karena itu, sistem merupakan kumpulan dari beberapa bagian yang saling berhubungan dan bekerja sama membentuk suatu kesatuan untuk mencapai tujuan dari sistem tersebut. Tujuan suatu sistem adalah untuk mencapai tujuan dan sasaran dalam ruang lingkup yang cukup sempit.

2.1.2 Definisi Informasi

Informasi merupakan suatu bentuk yang diolah menjadi suatu hal yang lebih berguna dan bermakna bagi orang yang menerimanya. Informasi adalah data yang diolah menjadi format yang bermakna bagi penerimanya dan berguna untuk pengambilan keputusan dan pengumpulan data. Dapat kita simpulkan bahwa pengertian informasi adalah data yang telah dimanipulasi atau diolah menjadi suatu bentuk yang bermakna bagi penerimanya, sehingga dapat dijadikan sebagai dasar pengambilan keputusan (Agustin 2018).

Informasi adalah data yang telah diklasifikasikan, diproses, atau diinterpretasikan untuk digunakan dalam proses pengambilan keputusan. Informasi dalam suatu organisasi sangatlah penting karena suatu organisasi tanpa informasi akan menjadi tidak berfungsi dan kurang sinergi. Informasi yang berkualitas harus bersifat akurat, terkini dan relevan, yang dimaksud adalah (Hakim, 2019) :

- a. Akurat berarti informasi tersebut akurat dan tidak boleh menyesatkan.
- b. Tepat waktu berarti informasi tidak terlambat sampai kepada penerimanya. Informasi yang lama tidak lagi mempunyai nilai. Karena informasi merupakan dasar dalam pengambilan keputusan.
- c. Relevan berarti informasi tersebut bermanfaat bagi pemakai. Relevansi informasi berbeda-beda pada setiap orang.

2.1.3 Sistem Informasi

Sistem informasi adalah kumpulan subsistem yang terintegrasi dan bekerja sama untuk memecahkan masalah tertentu dengan mengolah data menggunakan alat yang disebut komputer untuk memberikan nilai dan bantuan kepada pengguna. (Setiyanto et al., 2019).

Menurut Laudon, sistem informasi adalah komponen-komponen yang saling berhubungan yang bekerja sama untuk mengumpulkan, memproses, menyimpan, dan menyebarkan informasi untuk mendukung pengambilan keputusan, koordinasi, dan pengendalian, serta untuk memantau aktivitas dalam suatu organisasi. Sistem informasi merupakan kumpulan sub-sub sistem fisik dan non fisik yang saling berhubungan dan bekerja sama secara harmonis untuk mencapai tujuan yaitu mengolah data menjadi informasi yang berguna. (Handayani et al., 2022).

2.2 Implementasi

Implementasi adalah penerapan suatu ide, konsep, kebijakan, atau inovasi yang diungkapkan dalam bentuk tindakan praktis sehingga menimbulkan dampak berupa perubahan pengetahuan, keterampilan, nilai atau sikap. Implementasi juga dapat dipahami sebagai tindakan atau pelaksanaan suatu rencana yang dipersiapkan secara matang dan detail jauh sebelum implementasi dilaksanakan. Tujuan dari implementasi adalah :

1. Membuat desain sistem selama melaksanakan penelitian dan menganalisis sistem.
2. Menguji dan mendokumentasikan prosedur dan program yang diperlukan.
3. Menyelesaikan desain sistem yang sudah disetujui.
4. Melakukan pertimbangan untuk membangun sistem yang memenuhi kebutuhan pengguna (Irawan et al., 2019).

Dari pengertian implementasi di atas dapat dikatakan bahwa implementasi adalah suatu tindakan yang dilakukan seseorang untuk menyelesaikan suatu permasalahan. Karena pada dasarnya setiap rencana yang dibuat mempunyai tujuan yang ingin dicapai, maka pelaksanaan meliputi berbagai tindakan yang dilakukan untuk melaksanakan dan mewujudkan rencana yang dibuat untuk mencapai tujuan dari rencana yang sudah direncanakan sebelumnya.

2.3 Data Mining

Teknik dan metode khusus digunakan dalam *data mining* untuk mengungkap suatu pola dan juga informasi yang menarik dalam data tertentu. Teknik, metode, dan algoritma penambangan data sangat bervariasi. Pemilihan metode atau algoritma yang tepat sangat bergantung pada tujuan dan juga proses KDD secara keseluruhan. Idealnya data yang digunakan merupakan model yang baik dan cukup untuk data penelitian. Semakin besar jumlah data dan semakin rendah jumlah kesalahannya, maka semakin efektif model tersebut digunakan sebagai tolak ukur (Nofriansyah, 2014:10).

Data mining merupakan proses penggalian informasi dan pola yang bermanfaat dari data yang sangat besar. Data mining mencakup pengumpulan data, ekstraksi data, analisis data, dan statistik data. Data mining juga dikenal sebagai *Knowledge discovery*, *Knowledge extraction*, *data pattern analysis*, *information harvesting*, dan lain-lain.

Mengekstraksi pola dan informasi berharga dari sejumlah besar data merupakan suatu fokus dari sistem *data mining*. *Data mining* meliputi pengumpulan data, ekstraksi data, analisis data, dan statistik data. *Data mining* juga dikenal sebagai penemuan sebuah pengetahuan, ekstraksi pengetahuan, analisis pola data, dan juga pengumpulan informasi.

Data mining juga merupakan proses logis untuk menemukan informasi yang berguna. Setelah ditemukan informasi dan pola dapat digunakan untuk alat pendukung dalam pengambilan keputusan dalam mengembangkan bisnis. Alat data mining dapat memberikan jawaban untuk berbagai pertanyaan yang terkait dengan bisnis dan terlalu sulit untuk diselesaikan. Data mining juga dapat digunakan untuk meramalkan tren masa depan yang memungkinkan pebisnis membuat keputusan

yang efektif, proaktif, dan dinamis. Data-data yang diolah dengan menggunakan teknik data mining juga mampu menghasilkan pengetahuan yang sesuai dengan harapan (Arhami et al., 2020).

Data mining juga merupakan proses logis untuk menemukan informasi yang berguna. Setelah informasi dan pola ditemukan, maka dapat digunakan sebagai alat pendukung keputusan dalam pengembangan bisnis. Sistem *Data mining* memberikan jawaban atas berbagai pertanyaan terkait bisnis yang terlalu sulit untuk dipecahkan. Data mining juga dapat digunakan untuk memprediksi *trend* masa depan, sehingga memungkinkan pebisnis mengambil keputusan yang efektif, proaktif, dan dinamis.

2.4 Market Base Analysis

Market base analysis adalah metode atau teknik yang umum digunakan dan paling berguna dalam lingkungan pemasaran. Tujuan *Market base analysis* adalah untuk menentukan produk mana yang dibeli pelanggan pada saat yang bersamaan. Nama metode ini berasal dari kebiasaan pelanggan memasukkan barang ke dalam keranjang belanja atau daftar belanjaan (*shopping cart*). Mengetahui produk mana yang perlu dibeli pada saat yang sama sangat berguna bagi pengecer dan bisnis lainnya. Keuntungan menggunakan teknik *Market base analysis* adalah tidak hanya memungkinkan dalam memahami produk mana yang dibeli pada saat yang sama, namun juga memungkinkan dalam menggunakan informasi yang dapat dihasilkan untuk memesan ulang dua atau lebih produk pada saat yang bersamaan. Metode ini juga membantu manajer puncak melihat data pembelian pelanggan untuk melihat pelanggan mana yang merupakan pelanggan tetap atau pelanggan mana yang paling banyak melakukan pembelian (Gemala et al., 2018).

Ide yang mendasari *Market base analysis* adalah menghubungkan keputusan pembelian dengan konsumen. Misalnya, ketika pelanggan berbelanja di supermarket, mereka jarang membeli satu produk, namun biasanya seluruh keranjang belanja berisi produk dari kategori produk yang berbeda. Dengan menggunakan informasi keranjang belanja, pelanggan tidak hanya dapat mengekstrak kategori produk dan produk yang kemungkinan akan dibeli secara bersamaan, namun juga dapat menganalisis data untuk mengetahui produk dan

kategori produk apa saja yang tersedia. Dengan pengetahuan ini, manajer dapat mengambil keputusan yang baik yang bertujuan untuk mempengaruhi perilaku pembelian, seperti menstimulasi permintaan secara keseluruhan, mempromosikan kategori produk tertentu, atau membuat penawaran promosi untuk menjual produk yang kemungkinan besar akan meningkat (Solnet et al., 2019).

2.5 *Algoritma Apriori*

Algoritma apriori adalah algoritma yang menggunakan teknik *association rule* untuk menemukan *frequent itemset* yang sering digunakan. Untuk menemukan *association rule* dalam kumpulan data, pertama-tama yang perlu dicari yaitu kumpulan item yang paling sering digunakan. *Frequent itemset* adalah sekelompok item yang sering muncul secara bersamaan. Ada dua tolak ukur yang bersifat penting atau tidaknya suatu hubungan: dukungan dan kepercayaan. Support adalah nilai dukungan atau persentase kombinasi elemen-elemen dalam database, dan *Confidence* yang merupakan nilai kepastian atau kekuatan hubungan antar elemen dalam aturan asosiasi. Proses utama dari *algoritma apriori* adalah mendapatkan *itemset* yang paling sering digunakan (Riszky et al., 2019).

Algoritma apriori merupakan salah satu cabang *data mining* yang bertujuan untuk mengidentifikasi hubungan antara berbagai elemen yang disimpan dalam database yang cukup besar. Bentuk umum dari algoritma *apriori* seperti pada persamaan 1.

$$A_1 \dots A_n \rightarrow B \quad (1)$$

Tidak ada batasan jumlah produk yang dibeli oleh pelanggan, namun pelanggan yang membeli produk A kini memiliki peluang besar untuk membeli produk B juga. Bentuk lain yang lebih kompleks dari algoritma *apriori* seperti pada persamaan 2.

$$A, C \rightarrow B, D \quad (2)$$

Pelanggan yang membeli produk A dan C kemungkinan besar juga akan membeli produk B dan D. Salah satu masalah dalam menemukan algoritma *apriori* adalah banyaknya cara untuk menemukan aturan yang belum tentu aturan yang bersifat baik atau dapat diandalkan. Pentingnya aturan *association rule* dapat ditentukan oleh dua parameter yaitu *support* dan *confidence*. Support adalah

persentase kombinasi item dalam *database*, dan *Confidence* adalah kekuatan hubungan antar item dalam aturan asosiasi. (Setiawan et al., 2020).

Analisis asosiasi didefinisikan sebagai proses menemukan semua aturan terkait yang memenuhi persyaratan minimum *support* (minimum *support*) dan *confidence* (minimum *confidence*). Suatu aturan dikatakan *interesting rule* jika $support \geq \text{minimum support}$ dan $confidence \geq \text{minimum confidence}$. Fase ini akan mencari kombinasi item dalam database yang memenuhi persyaratan nilai dukungan minimum (C.Pradeepkumar et al., 2019).

Penting tidaknya suatu *association rule* dapat diketahui dengan dua parameter yaitu:

a. *Support*

Ukuran tingkat dominasi suatu item atau sekumpulan item dalam keseluruhan transaksi. Ukuran ini akan menguji *confidence* (misalnya dengan mempertimbangkan semua data transaksi yang ada, seberapa besar kemungkinan barang A dan barang B dibeli secara bersamaan).

b. *Confidence*

Dengan menggunakan nilai *confidence* yang telah ditentukan, penghitungan dilakukan menggunakan algoritma *apriori*. Algoritma ini membuat akan aturan asosiasi menggunakan pola *if-then*. Algoritma ini akan membantu mengurangi jumlah pengulangan pencarian yang telah dilakukan.

Adapun metodologi dasar analisis asosiasi adalah sebagai berikut :

c. Analisis Pola Frekuensi Tinggi

Pada tahapan ini akan mencari sebuah kombinasi item yang memenuhi syarat minimum dari nilai *support* dalam database yang sudah ada. Nilai *support* sebuah item diperoleh dengan rumus berikut.

$$Support A = \frac{\text{Jumlah transaksi yang mengandung A}}{\text{Total transaksi}} \quad (3)$$

Sedangkan nilai *support* dari 2 item diperoleh dari rumus berikut.

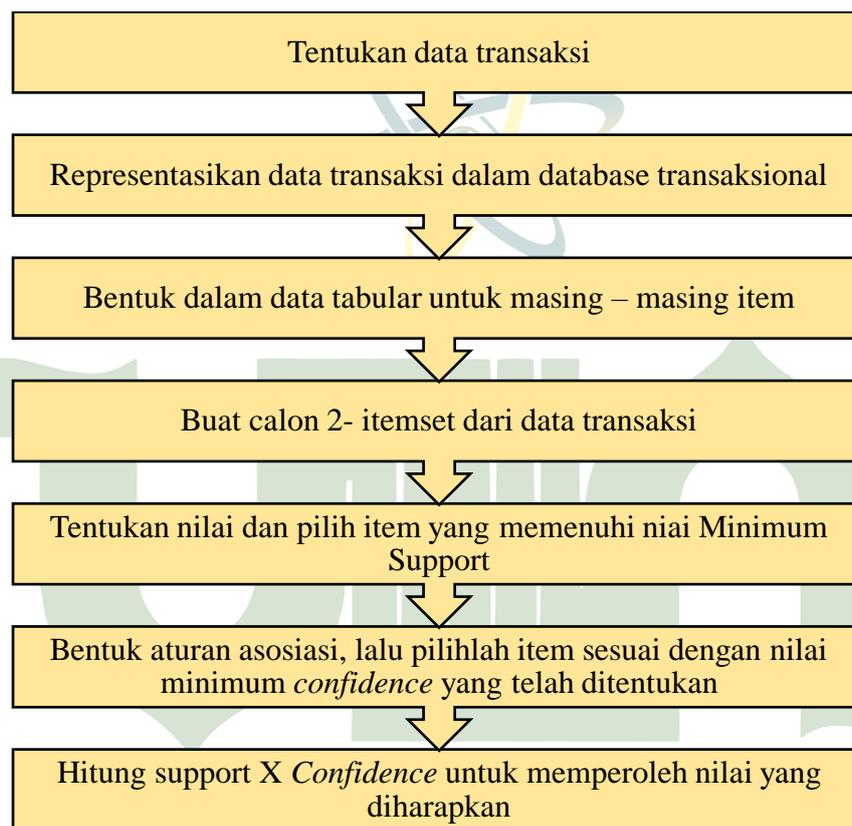
$$Support (A, B) = \frac{\sum \text{Transaksi mengandung A dan B}}{\text{Total transaksi}} \quad (4)$$

d. Pembentukan Aturan Asosiasi

Setelah semua pola frekuensi yang tinggi ditemukan, kemudian dicari aturan yang bersifat asosiatif dan juga memenuhi syarat minimum untuk *confidence* dengan menghitung *confidence* aturan asosiatif $A \rightarrow B$ dengan rumus berikut :

$$Confidence (A, B) = \frac{\sum \text{Transaksi mengandung A dan B}}{\sum \text{Transaksi mengandung A}} \quad (5)$$

Sebelum membahas mengenai analisis data, terlebih dahulu dipaparkan sebuah diagram dalam penyelesaian masalah menggunakan algoritma *Apriori* untuk memudahkan pemahaman dan alur proses yang akan dibahas (Ikhwan, 2018).



Gambar 2.1 Alur Proses Penyelesaian Algoritma *Apriori* (Ikhwan, 2018)

2.6 Website

Menurut *Afief*, website adalah suatu halaman web yang mempunyai domain/URL (*Uniform Resource Locator*) yang dipublikasikan di jaringan Internet dan dapat diakses oleh semua pengguna Internet dengan mengunjungi website tersebut dapat dijelaskan bahwa itu adalah himpunan. Dalam teknologi *World Wide Web* (WWW), halaman *website* biasanya berupa dokumen yang ditulis dalam

format *Hyper Text Markup Language* (HTML) yang dapat diakses melalui HTTP. HTTP adalah sebuah protokol yang dapat mengirimkan berbagai informasi untuk ditampilkan dari server suatu situs web kepada pengguna di web (Nofyat et al., 2018).

2.7 Bahasa Pemrograman

2.7.1 Pemrograman

Menurut Saragih (2018), bahasa pemrograman yang sering juga disebut dengan bahasa komputer atau bahasa pemrograman komputer adalah instruksi standar untuk mengendalikan komputer. Bahasa pemrograman ini adalah seperangkat aturan sintaksis dan semantik yang digunakan untuk mendefinisikan program komputer. Bahasa ini memungkinkan pemrogram untuk menjelaskan dengan tepat data apa yang akan diproses oleh komputer, bagaimana data tersebut akan disimpan/ditransmisikan, dan langkah apa yang akan diambil dalam situasi berbeda. Bahasa pemrograman adalah instruksi standar untuk menginstruksikan komputer untuk melakukan fungsi tertentu.

2.7.2 PHP

PHP adalah bahasa pemrograman *script server-side* yang dirancang untuk pengembangan web. PHP disebut sebagai bahasa pemrograman sisi server karena diproses di komputer server. PHP (*Hypertext Preprocessor*) merupakan bahasa scripting yang dapat digunakan untuk membuat program website dinamis. Berbeda dengan bahasa pemrograman *client-side* seperti *JavaScript* yang diproses di web browser (Irawan et al., 2018).

UNIVERSITAS ISLAM NEGERI
SUMATERA UTARA MEDAN



Gambar 2.2 Logo PHP

Sumber <https://www.google.com>

2.7.3 Database

Basis data atau database adalah kumpulan informasi yang disimpan secara sistematis dalam perangkat keras (komputer) dan disusun menjadi satu kesatuan yang utuh sehingga dapat diolah dengan menggunakan perangkat lunak. Dalam sistem ini, data dikumpulkan ke dalam database (Samsudin et al., 2019).

Database adalah sistem file komputer yang menggunakan metode organisasi tertentu yang dirancang untuk mempercepat pembaruan setiap *record* dan pembaruan simultan dari *record* terkait, dan untuk menyederhanakan dan mempercepat akses ke semua *record* melalui program aplikasi. Tidak hanya dapat mengakses semua *record* melalui program aplikasi, tetapi juga dapat dengan cepat mengakses data tersimpan yang perlu digunakan bersama untuk membuat laporan berkala secara khusus (Wijoyo et al., 2021)

2.7.4 XAMPP

2.7.5 Menurut Wikipedia (2017), pengertian XAMPP adalah perangkat lunak bebas yang mendukung banyak sistem operasi dan merupakan kompilasi dari beberapa program. XAMPP bertindak sebagai server mandiri (*localhost*) dan terdiri dari program server HTTP Apache, database MySQL, dan penerjemah bahasa yang ditulis dalam bahasa pemrograman PHP dan Perl. Nama XAMPP merupakan singkatan dari X (salah satu dari empat sistem operasi), Apache, MySQL, PHP, dan Perl. Program ini dilisensikan di bawah GNU *General Public License* secara bebas dan gratis. Server web yang mudah digunakan yang dapat menampilkan halaman web dinamis. XAMPP dikembangkan oleh tim proyek bernama Apache Friends, yang terdiri dari Tim Inti (*Core Team*), Tim Pengembang (*Development Team*) & Tim Dukungan (*Support Team*) (Safitri, 2018).

Berikut merupakan logo dari XAMPP :



Gambar 2.3 Logo XAMPP

Sumber (<https://www.apachefriends.org/>)

2.7.6 MySQL

MySQL sebenarnya berasal dari SQL (*Structure Query Language*), yang telah lama menjadi salah satu konsep database terpenting. SQL merupakan sebuah konsep operasi *database* yang khusus ditujukan untuk menyeleksi atau pemilihan dan memasukkan data, sehingga memungkinkan pengoperasian data dengan mudah diotomatisasi (Wijoyo et al., 2021).



Gambar 2.4 Logo MySQL
Sumber <https://www.google.com>

2.7.7 HTML

Hypertext Markup Language (HTML) adalah bahasa eksplorasi halaman untuk membuat dokumen *hypertext* atau *hypermedia*. HTML berisi kode kontrol di berbagai kemungkinan lokasi dalam dokumen yang memungkinkan Anda menjalin hubungan (*hyperlink*) dengan bagian lain dari dokumen atau dengan dokumen lain di *World Wide Web*. Tag HTML tidak peka terhadap huruf besar-kecil. Oleh karena itu, pengguna dapat menggunakan salah satu atau keduanya untuk mendapatkan keluaran yang sama. HTML berasal dari GSML (*Standard Generalized Markup Language*), dan notasinya yang disederhanakan. HTML dapat dibaca oleh banyak platform berbeda. HTML juga merupakan bahasa pemrograman yang fleksibel, memungkinkan Anda menyisipkan dan menggabungkan bahasa pemrograman lain seperti PHP, ASP, JSP, dan JavaScript. Jika terjadi kesalahan saat membuat HTML, browser tidak menampilkan kesalahan sintaksis; HTML berkembang seiring berkembangnya browser (Nofyat et al., 2018)



Gambar 2.5 Logo HTML *Hypertext Markup Language*

Sumber (<https://id.wikipedia.org/wiki/HTML>)

2.7.8 CSS

CSS adalah singkatan dari *Cascading Style Sheet* dan merupakan bahasa untuk merender halaman web. Ini termasuk warna, *tata letak*, *font*, dll. CSS memungkinkan pengembang web membuat halaman web yang beradaptasi dengan berbagai ukuran layar. CSS biasanya dibuat terpisah dari halaman HTML. CSS dapat dimasukkan ke dalam halaman HTML. Hal ini dimaksudkan untuk membantu menata halaman HTML dengan desain yang sama (Jannah et al., 2019).



Gambar 2.6 Logo CSS *Cascading Style Sheet*

Sumber (<https://en.wikipedia.org/wiki/CSS>)

2.8 *Unified Modeling Language* (UML)

Unified Modeling Language (UML) adalah bahasa spesifikasi standar yang digunakan untuk mendokumentasikan, menentukan, dan membuat perangkat lunak. UML adalah metodologi untuk mengembangkan sistem berorientasi objek dan alat untuk mendukung pengembangan sistem. *Unified Modeling Language* (UML) adalah bahasa berbasis grafik atau gambar untuk memvisualisasikan, menentukan,

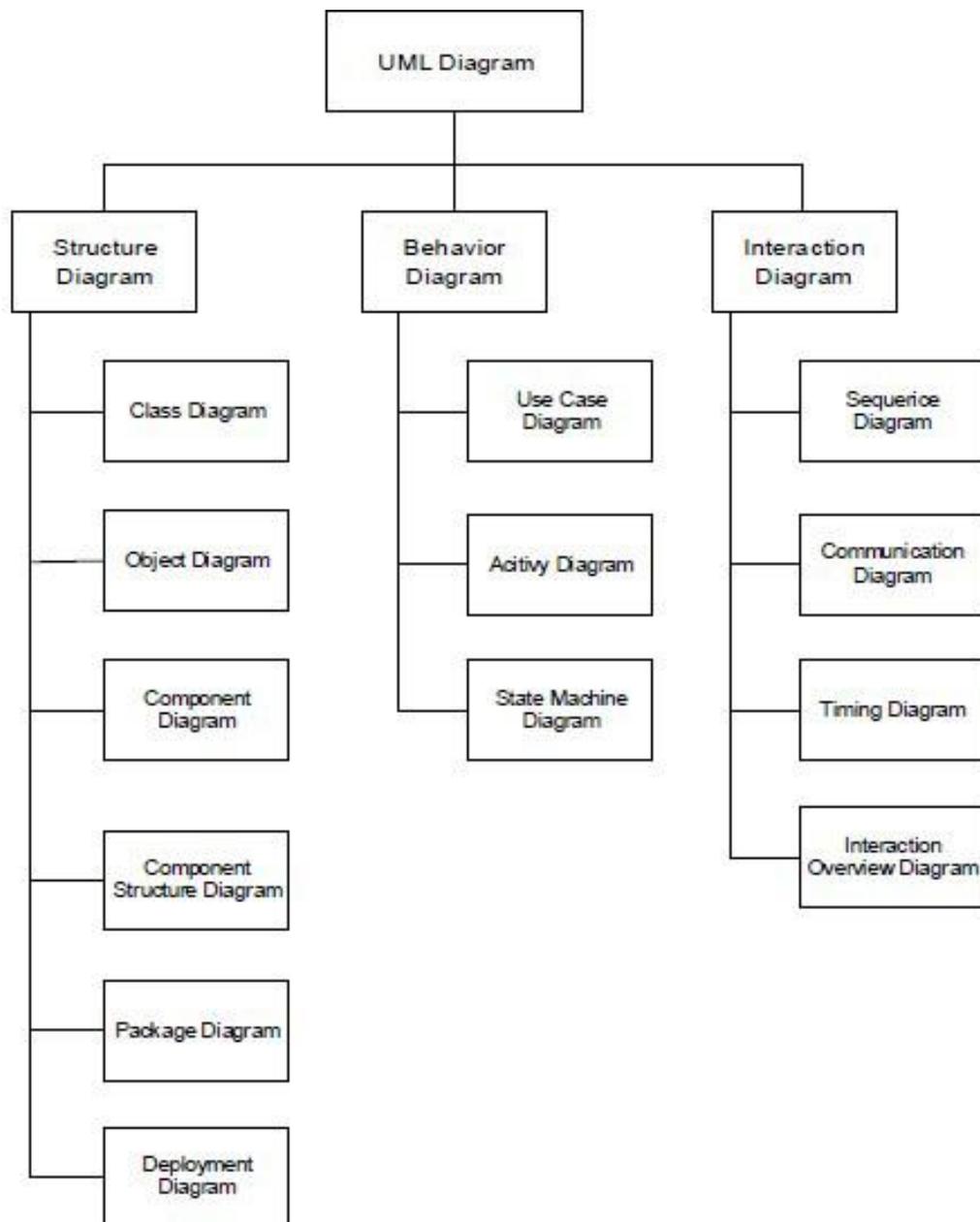
membuat, dan mendokumentasikan sistem pengembangan perangkat lunak berbasis OO (berorientasi objek). UML sendiri juga menyediakan standar untuk membuat cetak biru sistem yang berisi konsep proses bisnis, pembuatan kelas dalam bahasa pemrograman tertentu, skema database, dan komponen yang diperlukan untuk sistem perangkat lunak (Suendri, 2018).

Unified Modeling Language (UML) adalah bahasa pemodelan perangkat lunak yang distandarisi sebagai media pembuatan cetak biru perangkat lunak (Pressman). UML memungkinkan pengguna dapat memvisualisasikan, menentukan, mendesain, dan juga mendokumentasikan beberapa bagian sistem dalam perangkat lunak. Dengan kata lain, sama seperti arsitek membuat dokumen cetak biru yang digunakan perusahaan konstruksi untuk membangun gedung, arsitek perangkat lunak membuat diagram UML yang membantu pemrogram/pengembang membuat perangkat lunak (Jannah et al., 2019).



UNIVERSITAS ISLAM NEGERI
SUMATERA UTARA MEDAN

Berikut ini adalah contoh *Unified Modeling Language* (UML) :



Gambar 2.7 Diagram UML (*Unified Modeling Language*)
(Suendri, 2018)

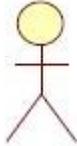
Diagram *Unified Modelling Language* (UML) antara lain sebagai berikut:

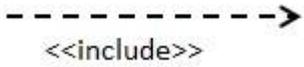
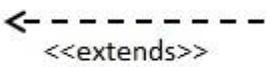
1. *Use Case Diagram*

Use case merupakan sekumpulan kelompok atau deskripsi yang saling terkait dan membentuk suatu sistem yang teratur yang dijalankan atau dipantau oleh sebuah aktor. *Use case* membantu merancang perilaku objek dan benda dalam model pengguna dan dapat dicapai melalui kolaborasi (Samsudin, 2018).

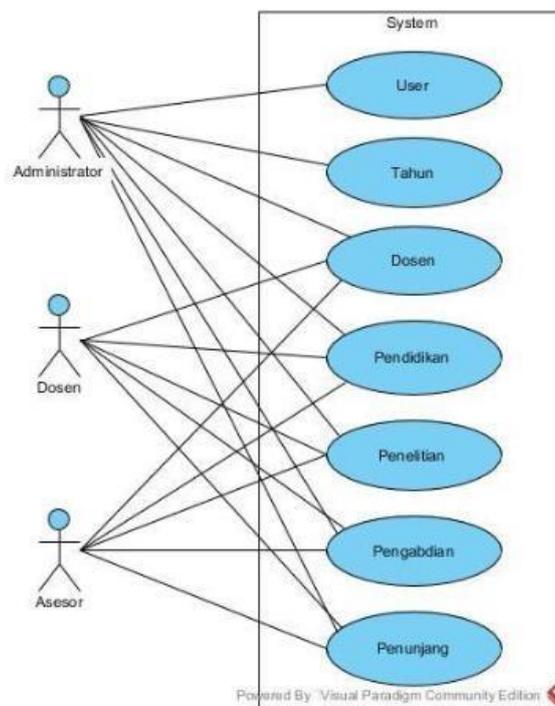
Simbol-simbol yang digunakan dalam *Use Case Diagram* yaitu :

Tabel 2.1 Simbol-Simbol *Use Case Diagram*
(Suendri, 2018)

Gambar	Keterangan
	<p><i>Use Case</i> menggambarkan sebuah fungsionalitas yang disediakan oleh sistem sebagai macam unit yang bertukar</p>
	<p>Aktor adalah abstraksi dari seseorang atau sistem lain yang mengaktifkan fungsionalitas dalam sistem target. Mengidentifikasi aktor memerlukan penentuan pembagian kerja dan tugas yang terkait dengan peran mereka dalam konteks sistem sasaran. Manusia dan sistem dapat memainkan banyak peran. Perhatikan bahwa aktor berinteraksi dengan <i>use case</i>, namun tidak memiliki kendali atas <i>use case</i> tersebut.</p>
	<p>Asosiasi antara <i>actor</i> dan <i>Use Case</i>, dapat digambarkan melalui garis tanpa panah yang mengindikasikan bahwa ada siapa atau apa yang meminta interaksi secara langsung dan bukan mengindikasikan sebuah data.</p>
	<p>Asosiasi antara aktor dan <i>Use Case</i> yang menggunakan tanda panah terbuka digunakan untuk mengindikasikan bila terdapat sebuah aktor</p>

	yang berinteraksi secara pasif dengan sistem yang tersedia.
	<i>Include</i> merupakan tanda yang terdapat di dalam <i>Use Case</i> lain (<i>required</i>) atau pemanggilan <i>Use Case</i> oleh <i>Use Case</i> lain, contohnya adalah sebuah pemanggilan fungsi program.
	<i>Extend</i> merupakan suatu perluasan dari <i>Use Case</i> lain yang jika kondisi atau syaratnya terpenuhi.

Berikut ini adalah contoh *Use Case Diagram* :



Gambar 2.8 *Use Case Diagram*
(Suendri, 2018)

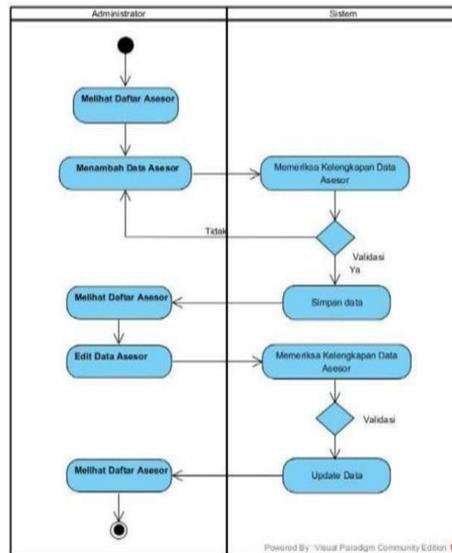
2. Activity Diagram

Activity Diagram adalah sebuah diagram dengan keadaan khusus dimana sebagian besar keadaan adalah tindakan dan sebagian besar transisi dipicu oleh penyelesaian *state* sebelumnya (*internal processing*). Oleh karena itu, daripada menggambarkan secara akurat bagaimana cara kerja internal suatu sistem (dan interaksi antar subsistem), *Activity Diagram* menggambarkan proses dan jalur aktivitas secara umum dari tingkat atas. *Activity Diagram* menggambarkan berbagai aliran aktivitas dalam sistem yang dirancang, bagaimana setiap aliran dimulai, keputusan apa yang dibuat, dan bagaimana berakhir. *Activity Diagram* pada dasarnya serupa strukturnya dengan diagram alur dan diagram alur dalam desain sistem terstruktur. *Activity Diagram* juga dapat mewakili proses paralel yang dapat terjadi dalam beberapa eksekusi. *Activity Diagram* memiliki beberapa simbol seperti pada table 2 dibawah ini (Samsudin, 2018)

Tabel 2. 2 Simbol *Activity Diagram*
(Triase, 2018)

Simbol	Nama	Keterangan
	<i>Start</i>	Menunjukkan awal dari proses
	<i>Stop</i>	Menyatakan akhir dari proses
	<i>Decision</i>	Digunakan untuk mewakili keadaan suatu proses
	<i>Action</i>	Merupakan tindakan yang dilakukan dalam arsitektur sistem.

Dibawah ini merupakan contoh *Activity Diagram* :



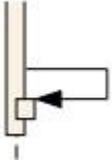
Gambar 2.9 *Activity Diagram*
(Samsudin, 2019)

3. *Sequence diagram*

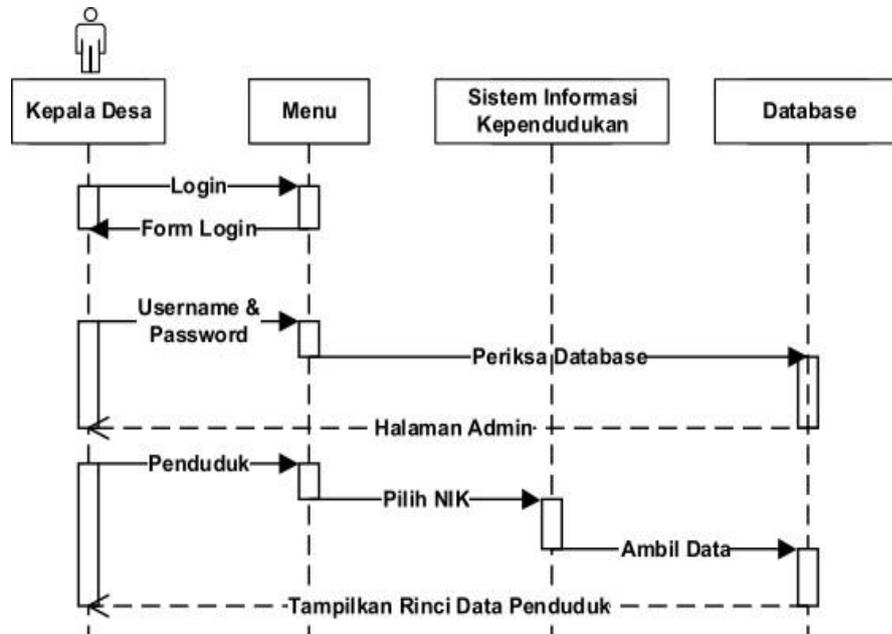
Sequence diagram menggambarkan perilaku objek dalam *use case* dengan memperkirakan masa hidup objek dan pesan yang dikirim dan diterima antar objek (Julianto & Setiawan, 2019).

Tabel 2.3 Simbol-Simbol *Sequence diagram*
(Alda, 2021)

Gambar	Keterangan
	<i>Entity Class</i> , merupakan bagian dari suatu sistem yang berisi kumpulan kelas-kelas berupa entitas yang membentuk diagram awal sistem dan dijadikan sebagai dasar penyusunan database.
	<i>Boundary Class</i> , Berisi kumpulan kelas yang bertindak sebagai <i>interfaces</i> atau interaksi antara satu atau lebih aktor dan sistem, seperti menampilkan formulir pendaftaran atau formulir cetak.

	<p><i>Control class</i>, Objek yang berisi logika aplikasi yang tidak bertanggung jawab atas entitas. Contohnya termasuk penghitungan dan aturan bisnis yang mencakup banyak objek.</p>
	<p><i>Message</i>, merupakan simbol mengirim pesan antar <i>class</i>.</p>
	<p><i>Recursive</i>, menggambarkan bagaimana cara pengiriman pesan yang dikirim untuk dirinya sendiri.</p>
	<p><i>Activation</i>, yang mewakili eksekusi operasi dari sebuah objek, berbentuk panjang kotak dan berbanding lurus dengan durasi dari aktivasi dengan operasi.</p>
	<p><i>Lifeline</i>, merupakan sebuah garis titik-titik yang terhubung dengan suatu objek, sepanjang <i>lifeline</i> yang terdapat <i>activation</i>.</p>

Berikut ini merupakan bagian dari contoh *Sequence diagram*:



Gambar 2.10 *Sequence diagram*
(Alda, 2021)

4. *Class diagram*

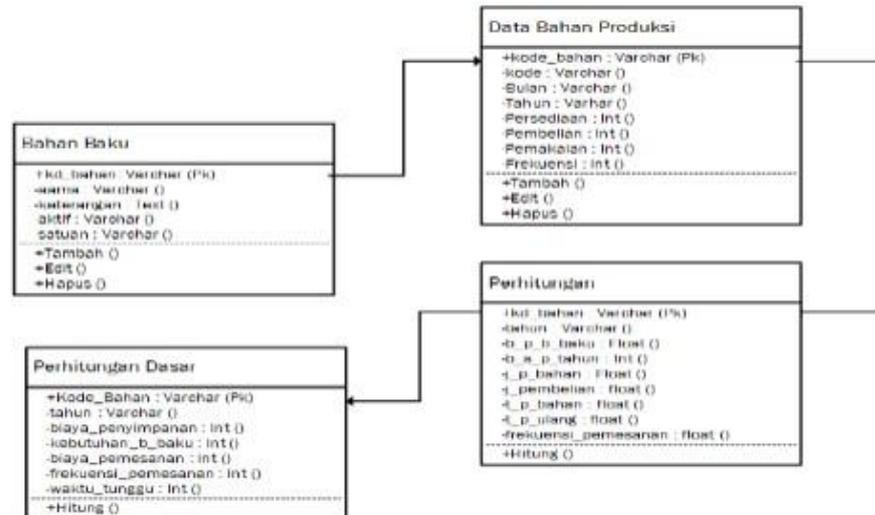
Class adalah spesifikasi yang membuat objek di awal dan merupakan inti dari pengembangan dan desain berorientasi objek. Sebuah *Class* menggambarkan keadaan atribut sistem dan pada saat yang sama menyediakan layanan untuk memanipulasi keadaan, metode, atau fungsi ini. *Class* dapat menjadi implementasi dari sebuah *Interface*, sebuah kelas abstrak yang hanya memiliki metode. *Interface* tidak dapat dipakai secara langsung, tetapi harus diimplementasikan di *Class* terlebih dahulu (Dewi et al. 2018).

Class diagram adalah jenis diagram UML yang digunakan untuk menggambarkan koneksi database. *Class diagram* menunjukkan ada atau tidaknya kelas dan hubungan dalam desain logis suatu sistem. Semua proses yang dijalankan oleh peserta aplikasi ditentukan menggunakan diagram kelas. *Class diagram* adalah diagram yang menunjukkan dan menjelaskan hubungan antara beberapa tabel dalam database dan tabel yang digunakan dalam sistem (Samsudin, 2018).

Tabel 2.4 Simbol-Simbol *Class diagram*
(Maiyendra, 2019)

Simbol	Keterangan
<p data-bbox="432 495 628 528"><i>Generalization</i></p> 	<p data-bbox="758 443 1369 636">Hubungan antara objek anak (<i>descendent</i>) berbagi perilaku dengan struktur data dari objek yang sudah ada di atasnya objek induk (<i>ancestor</i>).</p>
<p data-bbox="416 663 644 696"><i>Nary Association</i></p> 	<p data-bbox="758 663 1369 748">Merupakan suatu upaya untuk menghindari asosiasi yang lebih dari 2 objek.</p>
<p data-bbox="491 826 571 860"><i>Class</i></p> 	<p data-bbox="758 826 1369 965">Merupakan impunan dari berbagai objek yang berbagi atribut dengan operasi yang bersifat sama.</p>
<p data-bbox="437 1016 625 1050"><i>Collaboration</i></p> 	<p data-bbox="758 999 1369 1191">Deskripsi dari urutan sebuah aksi yang ditampilkan melalui sebuah sistem yang menghasilkan suatu hasil yang bernilai terukur bagi suatu <i>actor</i>.</p>
<p data-bbox="453 1220 609 1254"><i>Realization</i></p> 	<p data-bbox="758 1220 1369 1305">Operasi yang benar dilakukan oleh suatu objek.</p>
<p data-bbox="448 1386 614 1420"><i>Dependency</i></p> 	<p data-bbox="758 1332 1369 1585">Hubungan dimana sebuah perubahan terjadi pada suatu elemen yang mandiri (<i>independent</i>) dan akan mempegaruhi elemen yang telah bergantung padanya yaitu sebuah elemen yang bersifat tidak mandiri.</p>
<p data-bbox="448 1624 614 1657"><i>Association</i></p> 	<p data-bbox="758 1624 1369 1709">Apa yang dapat menghubungkan antara objek satu dengan objek lainnya.</p>

Berikut ini merupakan contoh *Class diagram* :



Gambar 2.11 *Class diagram*
(Irwan et al. 2020)

2.9 Penelitian Terdahulu

Penelitian-penelitian sebelumnya merupakan sebuah penelitian yang telah dilakukan sebelumnya dan selanjutnya akan digunakan oleh penulis sebagai acuan/referensi untuk melakukan penelitian. Tabel berikut ini akan menunjukkan bagaimana penelitian sebelumnya yang berkaitan dengan judul penelitian oleh penulis.

Tabel 2.5 Tabel Penelitian Terdahulu

No	Judul	Hasil Penelitian	Kelebihan	Kekurangan
1	<p>“<i>Market base analysis</i> dengan Algoritma <i>Apriori</i> pada <i>Ecommerce</i> Toko Busana Muslim Trendy (Aji Setiawan, Rizka Mulyanti, 2020)”</p>	<p>Penelitian ini mencoba menerapkan <i>data mining</i> dengan menggunakan algoritma yang dapat memprediksi jenis produk yang akan dibeli konsumen. Pada penelitian ini, mengembangkan sistem yang menggunakan pendekatan <i>market base analysis</i> (MBA) untuk menentukan produk mana yang akan dibeli konsumen berdasarkan analisis daftar transaksi pelanggan. Metodologi perancangan sistem yang digunakan untuk membuat aplikasi ini didasarkan pada pendekatan model <i>Waterfall</i>.</p>	<p>Jurnal tersebut mendeskripsikan <i>step by step</i> bagaimana cara perhitungan algoritma <i>apriori</i> sehingga dapat mudah dipahami oleh peneliti selanjutnya dan penulisan jurnal ini juga sesuai dengan kaidah penulisan jurnal.</p>	<p>Penelitian ini tidak dilengkapi dengan pengimplementasian sistem ke dalam bahasa pemrograman sehingga informasi yang ditampilkan tidak bersifat detail.</p>

Lanjutan Tabel 2.5 Tabel Penelitian Terdahulu

2	<p>“Data Mining Menggunakan Algoritma <i>Apriori</i> untuk Rekomendasi Produk bagi Pelanggan (Ariefana Ria Riszky, Mujiono Sadikin, 2019)”</p>	<p>Penelitian ini mengkaji bagaimana penerapan dari teknik <i>association rule</i> dan algoritma <i>apriori</i> pada dataset transaksi belanja untuk membantu membentuk kandidat kombinasi antar item produk dalam tindakan rekomendasi produk kepada pelanggan. Algoritma <i>apriori</i> dapat membuat <i>association rule</i> sebagai acuan/referensi promosi produk dan pendukung keputusan dalam memberikan rekomendasi produk kepada konsumen berdasarkan dukungan dan kepercayaan minimum yang ditetapkan.</p>	<p>Pembahasan metode pada jurnal ini sangat jelas, sehingga hanya membaca abstraknya saja pembaca dapat mengetahui bagaimana hasil dari penelitian tersebut.</p>	<p>Jurnal ini masih menggunakan aplikasi lama dalam penyelesaian data mining, yang mana aplikasi tersebut tidak menampilkan data yang jelas sehingga pembaca sulit untuk melihat hasil dari penelitian tersebut.</p>
---	--	--	--	--

Lanjutan Tabel 2.5 Tabel Penelitian Terdahulu

3	<p>“<i>Market base analysis</i> Pada Mini Market Ayu Dengan Algoritma <i>Apriori</i> (Erlin Elisa, 2018)”</p>	<p><i>Market base analysis</i> dengan algoritma <i>Apriori</i> merupakan teknik <i>data mining</i> yang mencari pola terkait berdasarkan perilaku pembelian konsumen dan bertujuan untuk mempelajari produk mana yang dibeli dalam waktu bersamaan. Hasil penelitian menunjukkan nilai <i>support</i> dan <i>confidence</i> tertinggi terdapat pada minyak dan susu dengan nilai dukungan sebesar 42,85%. dan tingkat kepercayaan sebesar 85,71%.</p>	<p>Metode dan desain penelitian ini dibuat secara lengkap serta dijelaskan secara detail membuat pembaca mudah memahami isi dari jurnal tersebut. Penyajian dalam bentuk data juga dijelaskan secara detail.</p>	<p>Pada jurnal ini masih menggunakan perhitungan secara manual dan belum mengimplementasikan kedalam bahasa pemrograman.</p>
---	---	---	--	--

Lanjutan Tabel 2.5 Tabel Penelitian Terdahulu

4	“ <i>Market base analysis</i> pada Barang Minimarket dimasa Pandemi Covid-19 (Sasa Ani Arnomo, 2019)”	Metode <i>market basket analys</i> dapat digunakan untuk menganalisis bagaimana keterkaitan antara produk yang dibeli konsumen dan juga untuk mengetahui bagaimana kebiasaan konsumen dalam membeli barang yang tersedia. Penelitian ini juga berhasil menemukan bagaimana kombinasi antara produk yang memiliki keterkaitan dan memiliki nilai <i>support</i> dan <i>confidence</i> dengan tolak ukur yang tertinggi.	Dalam jurnal tersebut terdapat variabel-variabel yang bersifat relevan dengan fenomena yang ada. Kemudian pembahasan yang rinci dan data yang lengkap membuat pembaca mudah mengerti isi dari jurnal tersebut.	Jurnal ini belum menjelaskan bagaimana secara detail tentang penelitian serta metode yang digunakan dalam penelitian ini. Pada jurnal ini juga belum mengimplementasikan bahasa pemrograman seperti php atau python.
---	---	--	--	--

Terdapat beberapa perbedaan dan perkembangan antara penelitian ini dengan penelitian-penelitian sebelumnya pada penelitian yang dilakukan oleh penulis. Pertama, yaitu perbedaan di lokasi penelitian. Tentunya setiap lokasi mempunyai karakteristik yang berbeda dengan penelitian-penelitian sebelumnya. Kedua, sistem yang dibangun pada penelitian ini akan dibangun menggunakan website yang nantinya akan menampilkan hasil penelitian ini, sehingga informasi yang ditampilkan akan lebih jelas dan detail. Penelitian ini juga menggunakan teknik *market base analysis*. *Market base analysis* adalah sebuah teknik yang akan menentukan hubungan antara produk yang dibeli pada waktu yang bersamaan.

Sehingga kita dapat mengetahui pola pembelian konsumen dan mengkombinasikan barang yang sering dibeli dan dapat menentukan strategi promosi yang tepat untuk perusahaan.



UNIVERSITAS ISLAM NEGERI
SUMATERA UTARA MEDAN