

BAB II

LANDASAN TEORI

2.1 Penelitian Terdahulu

Berikut ini merupakan beberapa hasil penelitian terdahulu yang terkait dan menjadi bahan acuan dan referensi penelitian ini :

1. Penelitian oleh Sigit Kurniawan, Windu Gata dan Hari Wiyana (2018)
Dalam penelitian ini terdapat permasalahan pada tempat penelitian yaitu belum terorganisirnya produk-produk yang paling banyak terjual dan keterhubungan antar produk didalam toko, selain itu masih banyak produk-produk yang diproduksi namun proses penjualannya belum maksimal. Metode atau algoritma yang digunakan dalam penelitian ini adalah algoritma *FP-Growth* untuk melakukan pencarian pola yang muncul berdasarkan *database* yang ada. (Kurniawan et al., 2018b)
2. Penelitian oleh Muhammad Tabrani dan Eni Pudjiarti (2017). Dalam penelitian ini terdapat masalah yaitu tempat penelitian masih menggunakan program yang belum sepenuhnya terintegrasi, yang berarti pencatatan dan pengolahan data hanya dapat diakses oleh satu departemen, yaitu admin gudang. Dan pelaporan dari kantor cabang ke kantor pusat dilakukan dengan format Excel sehingga tidak efisien. Metode yang digunakan dalam penelitian ini adalah Metode Waterfall untuk membuat sistem yang terintegrasi tersebut. (Tabrani & Pudjiarti, 2017)

3. Penelitian oleh Dwi Astuti, Ade Rahmat Iskandar dan Atik Febrianti (2019). Permasalahan yang terjadi dalam penelitian ini adalah dikarenakan UMKM mengalami penurunan sebesar 0,003% menyebabkan ekonomi Indonesia bertumbuh dengan sangat lambat. Hal ini berdampak pada pertumbuhan ekonomi negara. Menurut LPPi dan Bank Indonesia, terdapat berbagai kendala yang menyebabkan penurunan persentase jumlah UMKM. Salah satu di antaranya adalah kurangnya strategi pemasaran yang efektif bagi para pelaku UMKM untuk menjual produk mereka. Metode *CRISP-DM* dipilih untuk penelitian ini, dengan *K-Means Clustering* sebagai algoritma yang diterapkan. (Astuti et al., 2019)
4. Penelitian oleh Msy Aulia Hasanah, Sopian Soim dan Ade Silvia Handayani (2021). Permasalahan dalam penelitian ini adalah dikarenakan curah hujan yang tinggi yang dapat memengaruhi volume aliran debit air sehingga menyebabkan aliran sungai menjadi kelebihan kapasitas dan menjadi penyebab banjir. Akibatnya, ada kebutuhan akan teknologi dan informasi yang berperan dalam pengelolaan data sebagai landasan untuk pengambilan keputusan. Dataset yang digunakan merupakan dataset historis curah hujan yang diperoleh dari BMKG.. Penelitian ini memanfaatkan Decision Tree sebagai metode dan menggunakan CART sebagai algoritma perhitungannya, bertujuan agar dapat memprediksi curah hujan yang berpotensi menimbulkan banjir. (Hasanah et al., 2021)

5. Penelitian oleh Rizal Amegia Saputra, Sri Wasianti dan Rusli Nugraha (2020). Permasalahan yang terjadi pada penelitian ini adalah penempatan barang di toko Berkah Baru selama ini tidak tertata dengan baik dan data transaksi yang ada tidak digunakan dan dimanfaatkan dengan maksimal dan hanya dijadikan arsip atau data mati saja. Data tersebut dapat dimanfaatkan untuk mengembangkan strategi pemasaran, dengan menggunakan data penjualan sebagai acuan untuk merekomendasikan penempatan barang agar pelanggan lebih mudah mengambil dan menambahkan barang yang ingin dibeli, sehingga dapat membantu meningkatkan penjualan. Metode yang digunakan dalam penelitian ini adalah Algoritma Apriori, yang berfungsi untuk menemukan pola hubungan antar produk pada data transaksi atau penjualan. (Saputra et al., 2020).
6. Penelitian oleh Srinadh, V (2022) dengan judul penelitian *Evaluation of Apriori, FP-Growth and Eclat association rule mining algorithms*. Di dalam penelitian ini menjelaskan bahwa terdapat beberapa algoritma didalam aturan asosiasi dalam mencari pola yang berhubungan dalam suatu *dataset*. Algoritma-algoritma tersebut memiliki pendekatan atau cara kerja yang berbeda untuk mendapatkan hasil pola yang dicari. Untuk itu, peneliti tersebut membandingkan 3 algoritma yang berbeda dalam menentukan suatu pola hubungan atau aturan asosiasi yang ada dengan menjadikan waktu eksekusi dan penggunaan memori tiap algoritma yang digunakan sebagai acuan perbandingan antar algoritma tersebut. Algoritma yang digunakan dalam penelitian ini adalah

algoritma Apriori , FP-Growth dan juga Eclat untuk mencari aturan asosiasi yang ditentukan. (Srinadh, 2022)

2.2 Data Mining

Data Mining, yang juga dikenal sebagai *Knowledge Discovery in Database* (KDD), adalah aktivitas yang melibatkan pengumpulan data dan penggunaan data historis untuk menemukan pengetahuan, informasi, keteraturan, pola, atau hubungan. Tujuannya adalah sebagai alternatif dalam pengambilan keputusan atau untuk meningkatkan pengambilan keputusan di masa depan. (Bulolo, Efori.2020)

Data Mining adalah salah satu bidang ilmu yang berkembang pesat, dan perkembangan ini dipengaruhi oleh sejumlah faktor, di antaranya adalah:

1. Semakin banyak orang yang menyadari betapa pentingnya data.
2. Penggunaan hasil olahan data dalam beragam bidang, seperti bisnis, mengalami peningkatan yang signifikan.
3. Peningkatan dalam aksesibilitas internet.
4. Proses pengembangan kumpulan data berlangsung dengan sangat cepat.
5. Berkembangnya *hardware* dan *software* yang ada.
6. Perangkat penyimpanan yang semakin besar dengan harga yang lebih bersahabat.
7. Inovasi yang berlangsung dengan cepat di dunia komputasi komputer..

2.2.1 Metode pada Data Mining

Menurut Anjar Wanto (2020), Ada bermacam tugas atau metode yang ada pada oleh *Data Mining* untuk melakukan pemecahan masalah dan pencarian informasi baru, di antaranya :

1. Klustering (*Clustering*)

Umumnya dipakai untuk mengelompokkan atau mengidentifikasi data berdasarkan atribut yang telah ditentukan. Contoh algoritma: *K-Means, K-Medoids, dan lain-lain.*

2. Metode Klasifikasi (*Classification*)

Digunakan jika ingin mengidentifikasi suatu fungsi atau model yang dapat membedakan antara kategori atau konsep suatu data, yang bertujuan untuk menilai kelas dari suatu objek yang belum memiliki label.

Contoh Algoritma : *K-Nearest Neighbor (KNN), C4.5*

3. Metode Asosiasi (*Association*)

Berfungsi untuk mengatasi jika ada masalah bisnis tertentu dengan cara menganalisis dataset tabel penjualan dan mengidentifikasi Item apa saja yang sering dibeli pelanggan dalam satu waktu. Sebagai contoh, jika seseorang ingin membeli sikat gigi, biasanya mereka juga ingin membeli pasta gigi.

Contoh Algoritma : *ECLAT, Frequent Pattern Growth (FP-Growth), Apriori* dan lain sebagainya.

4. Metode Estimasi (*Estimation*)

Metode ini digunakan ketika kita ingin menilai atau menerka suatu nilai yang belum ada.

Contoh Algoritma : *Confidence Interval Estimations, Regresi Linier, dsb.*

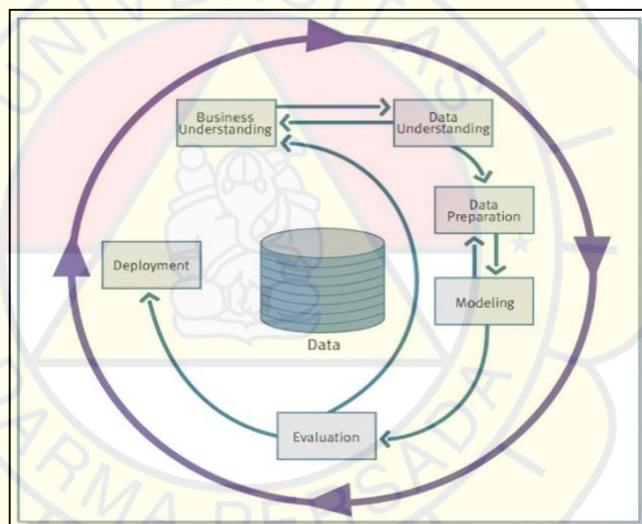
5. Metode Prediksi (*Predictions*)

Metode ini dipakai untuk menerka atau meramalkan sebuah peristiwa yang mungkin akan terjadi dimasa depan dengan dataset historis yang dimiliki.

Contoh Algoritma : K-Nearest Neighbor (KNN), Decision Tree dan lain-lain.

2.2.2 Tahapan Data Mining dengan CRISP-DM

Pada penelitian ini, penulis menggunakan kaidah *Cross Industry Standard for Data Mining* (CRISP-DM) untuk melakukan proses pengembangan data yang ada.



Gambar 2.1 Tahapan Dalam CRISP-DM

Penjelasan tahapan-tahapan tersebut adalah sebagai berikut (Hasanah et al., 2021b) :

1. *Business Understanding* (Pemahaman tentang Bisnis)

Tahap ini mencakup beberapa kegiatan, seperti memahami kebutuhan dan tujuan dari perspektif bisnis, mendefinisikan masalah dalam konteks data mining berdasarkan pengetahuan

yang ada, serta merancang cara untuk mencapai tujuan yang telah ditetapkan.

2. *Data Understanding* (Pemahaman Data)

Pada tahap ini dimulai dari mengunmpulkan data sebagai bahan acuan, mendeskripsikan data, dan mengevaluasi kualitas data.

3. *Data Preparation* (Persiapan Data)

Pada tahap ini, yang bertujuan untuk membangun dataset akhir dari data mentah, beberapa aktivitas yang akan dilakukan meliputi *Data Cleaning* (pembersihan data), *Data Selection* (pemilihan data), penentuan atribut dan *record*, serta *Data Transformation* (transformasi data) untuk disiapkan sebagai *input* dalam tahap pemodelan.

4. *Modelling* (Pemodelan)

Di tahap ini, Machine Learning terlibat secara langsung dalam penentuan teknik *Data Mining* yang akan digunakan, alat dan algoritma yang digunakan dalam *Data Mining*. Pada penelitian ini menggunakan model *Association Rules* dengan algoritma yang digunakan adalah algoritma *FP-Growth* dan algoritma *ECLAT*.

5. *Evalutaion* (Evaluasi)

Pada tahap ini dilakukan interpretasi terhadap hasil dari *data mining* yang telah dihasilkan dalam proses pemodelan pada tahap sebelumnya. Evaluasi dilakukan dengan tujuan agar model yang

ditentukan telah sesuai dengan tujuan yang telah dibuat pada tahap awal.

6. *Deployment* (Penyebaran)

Tahapan ini dilakukan dengan pembuatan sistem informasi berbasis website, *report* (laporan) serta jurnal memakai model yang telah dihasilkan.

2.2.3 *Market Basket Analysis*

Metode analisis Perilaku konsumen yang spesifik dari kelompok tertentu. Sumber data dapat berasal dari transaksi kartu lotre, kartu kredit, kupon diskon, dan keluhan pelanggan. Metode ini biasanya digunakan sebagai langkah awal untuk mencari pengetahuan dari data transaksi ketika pola spesifik yang dicari belum diketahui. Kebutuhan akan market basket analysis muncul dari keakuratan dan manfaat yang dihasilkan dalam bentuk aturan asosiasi (association rules). (Junaidi, 2019)

2.2.4 *Association Rules*

Suatu teknik atau metode dalam *data mining* yang bisa dipakai untuk menemukan suatu aturan baru didalam kombinasi item pada dataset. Salah satu tahap pada metode ini yang cukup menarik banyak peneliti untuk mengembangkan metode dan Untuk meningkatkan efisiensi, algoritma yang digunakan adalah analisis pola dengan frekuensi tinggi (frequent pattern mining). Penting atau tidaknya teknik asosiasi, bisa diukur melalui dua indikator, yang pertama adalah *support* dan yang kedua adalah *confidence*. Nilai penunjang atau *Support* merujuk pada persentase dari kombinasi itemset pada basis data, sementara *confidence* atau nilai

kepastian, menunjukkan kekuatan korelasi antar tiap item dalam dataset.(Badrul, 2016)

Menurut (Buulolo, Efori.2020), Association Rule adalah suatu teknik didalam data mining yang digunakan untuk membentuk asosiasi antara item pada kombinasi itemsets. Aturan ini biasanya berbentuk If... then... (jika... maka...). Proses pembuatan aturan asosiasi merupakan kelanjutan dari algoritma Apriori, FP-Growth, dan ECLAT, di mana pembentukan aturan dimulai dari kombinasi dua itemsets yang memenuhi kriteria support minimum, yang berarti hanya kombinasi yang memenuhi support minimum yang akan dibuatkan aturannya.

Menurut Huda,Miftahul.(2019), *Association Rule Mining* adalah prosedur yang bertujuan untuk menemukan pola umum, korelasi, asosiasi atau struktur data-set yang diperoleh dari berbagai jenis basis data seperti basis data relatif, basis data transaksi, dan berbagai jenis basis data lain yang diperoleh dari tempat penyimpanan data. Association Rule Mining pada tingkat dasar menerapkan konsep dasar *Machine Learning* untuk menganalisis pola suatu data atau kejadian pada basis data.

Rumus untuk menentukan *Support* adalah :

$$Support (A,B) = \frac{\sum Transaksi Mengandung A dan B}{\sum Total Transaksi}$$

Sementara rumus untuk menentukan *Confidence* adalah :

$$Confidence (A,B) = \frac{\sum Transaksi Mengandung A dan B}{\sum Total Transaksi mengandung A}$$

2.2.5 Algoritma FP-Growth

Sebagai pengembangan dari Apriori, algoritma ini bertugas mengidentifikasi itemset yang paling sering muncul (Frequent Itemset). Algoritma ini memakai konsep pembuatan tree, yang dikenal sebagai FP-Tree. Pada saat frequent itemset dicari, algoritma ini tidak menggunakan *metode generate candidate* seperti yang dilakukan oleh algoritma Apriori, sehingga dapat memberikan hasil yang lebih maksimal. (Amelia & Utomo, 2019)

Seperti yang sudah dijelaskan diatas, FP-Growth berbeda jika dibandingkan dengan Apriori. Algoritma ini adalah cara lain yang efektif untuk menemukan *frequent itemset*, artinya adalah kumpulan data yang sering muncul pada sebuah dataset. Selain itu, FP-Growth menggunakan konsep pembangunan tree, yang dikenal sebagai FP-Tree, untuk menemukan itemset, alih-alih memakai metode generate candidate seperti yang diterapkan pada Apriori.

Metode FP-Growth ini dapat dibagi lagi menjadi 3 tahapan, yaitu (Puspitasari et al., 2020):

1. *Conditional Pattern Base*

Ini adalah *sub-database* yang berisi lintasan prefix (*prefix path*) dan pola akhiran (*suffix pattern*). Tahap ini dihasilkan oleh FP-Tree yang telah dibentuk sebelumnya.

2. *Conditional FP-Tree*

Pada langkah ini, *support count* tiap *itemset* di tiap *conditional pattern base* akan dijumlahkan, setelah itu tiap *itemset* yang

memiliki jumlah *support count* \geq *minimum support count* akan dihasilkan menggunakan *Conditional FP-Tree*.

3. Pencarian *Frequent Itemset*

Tahap ini dilakukan ketika *Conditional FP-Tree* merupakan lintasan tunggal (*single path*), sehingga *frequent itemset* diperoleh dengan mengombinasikan item untuk setiap *Conditional FP-Tree*. Jika tidak, FP-Growth akan dibangkitkan secara rekursif.

2.2.6 *Algoritma Eclat*

Menurut Huda, Miftahul. (2019), Equivalence Class Transformation atau disingkat ECLAT adalah metode sederhana yang digunakan untuk mencari item dengan kemunculan paling banyak dengan melakukan pencarian menggunakan metode *depth-first search* pada basis data. Pada metode ini, item yang paling sering muncul lebih didahulukan daripada item yang jarang muncul, tanpa mempertimbangkan urutan kemunculannya. *Depth-First Search* (DFS) adalah algoritma yang dimulai dari simpul akar dan mengeksplorasi setiap cabang sejauh mungkin sebelum melakukan *backtracking*. Contoh penggunaan algoritma ECLAT termasuk menentukan pola keterkaitan antara subjek buku dan program studi mahasiswa, serta menganalisis pola pinjaman berdasarkan buku yang sering dipinjam secara bersamaan.

Eclat mengubah dataset menjadi vertikal dari yang awalnya horizontal dengan membuat daftar transaksi (*tid-list*) untuk setiap itemset. *Tid-list* (TID) untuk Itemset diperoleh dari urutan ID transaksi yang

mencakup itemset tersebut, di mana setiap ID transaksi merepresentasikan kumpulan item yang dibeli bersama-sama dalam satu transaksi.(Subianto et al., 2018)

Tabel 2.1 Contoh Dataset yang masih Horizontal (Viswanath,K,dkk.2017.)

TID (Sequence ID)	Itemsets
1	E1. E3
2	E1. E4
3	E2, E3,E4
4	E2,E4

Tabel 2.1 menampilkan dataset yang masih dalam bentuk horizontal per-transaksi membeli item apa saja.

Tabel 2.2 Dataset yang telah diubah ke bentuk Vertical 1 (Viswanath,K,dkk.2017)

Itemsets	TID list
E1	{1,2}
E2	{3,4}
E3	{3}
E4	{2,3,4}

Tabel 2.2 menampilkan *dataset* yang telah diubah kedalam bentuk *vertical* untuk setiap 1 *itemset* yang ada pada transaksi.

Tabel 2.3 Dataset yang telah diubah ke bentuk Vertical 2 (Viswanath,K,dkk.2017)

Itemsets	TID list
----------	----------

{E1, E2}	-
{E1, E3}	{3}
{E1, E4}	{2}
{E2, E4}	{3,4}

Tabel 2.3 menampilkan *dataset* yang telah diubah kedalam bentuk vertical untuk tiap 2 *itemset*. Atau bisa dibilang juga pada tabel ini menampilkan transaksi yang berisikan 2 kombinasi *item*.

Tabel 2.4 *Dataset yang telah diubah ke bentuk Vertical 3 (Viswanath,K,dkk.2017)*

Itemsets	TID list
{E2, E3, E4}	{3}

Tabel 2.4 menampilkan *dataset* yang telah diubah kedalam bentuk vertical untuk tiap 3 *itemset*. Atau bisa dibilang juga pada tabel ini menampilkan transaksi yang berisikan 3 kombinasi *item*.

2.3 Pembuatan Website

2.3.1 HTML

“HTML adalah singkatan dari Hyper Text Markup Language, adalah salah satu bahasa pemrograman yang terstruktur dan dirancang untuk merancang halaman *website* yang dapat digunakan dan ditampilkan melalui *web browser*.”(Setiawan,Didik.2017:16)

“HTML, atau Hyper Text Markup Language, adalah bahasa yang dikenali oleh web browser untuk menampilkan berbagai informasi, termasuk teks, gambar, animasi, dan video.” (Tabrani & Pudjiarti, 2017)

“ HTML (*Hyper Text Markup Language*) berperan sebagai bahasa yang mendeskripsikan tata letak halaman web dan memungkinkan publikasi dokumen secara online. Sebuah pernyataan dalam HTML disebut tag, yang dinyatakan dalam kurung siku (<>) dan tag harus ditulis secara berpasangan untuk menandai dokumen atau bagian dari dokumen tersebut. Setiap tag memiliki pasangan pembuka dan penutup, dengan tag penutup menggunakan garis miring (/) di depan namanya.” (Pahlevi et al., 2018)

Berdasarkan beberapa definisi dan pengertian tersebut, dapat disimpulkan bahwa HTML adalah bahasa pemrograman dasar yang dipakai untuk pembuatan situs web.

2.3.2 CSS (Cascading Style Sheets)

“CSS, singkatan dari Cascading Style Sheets, adalah kode pemrograman yang digunakan untuk menghias dan mengatur tampilan atau tata letak halaman web agar lebih menarik.”(Setiawan,Didik.2017:166)

“CSS adalah sebuah bahasa pemrograman yang bisa digunakan untuk mendesain tampilan website. Dalam CSS, penanda yang dikenal sebagai ID dan class digunakan untuk mengatur elemen-elemen. CSS mampu mengubah font, warna, dan format teks, serta mengatur tata letak, lebar, tinggi, dan warna elemen. Selain itu, CSS juga dapat mengubah tampilan formulir, membuat halaman web responsif, dan melakukan banyak hal lainnya.”(Limbong,T.2021:51-52)

Berdasarkan dari pernyataan diatas, dapat disimpulkan bahwa CSS merupakan salah satu bahasa pemrograman yang berfungsi untuk memperindah tampilan *website* agar lebih menarik untuk dilihat.

2.3.3 PHP

“PHP, yang merupakan sebuah singkatan dari "Hypertext Preprocessor," PHP adalah salah satu dari banyaknya bahasa pemrograman yang serbaguna yang bisa dipakai untuk membuat, mengembangkan, dan menangani situs web. PHP bisa digunakan beriringan dengan HTML, dan sekarang, tidak sedikit digunakan untuk membangun program situs web yang dinamis.”(Mundzir,MF.2018:3)

“PHP Merupakan bahasa pemrograman yang digunakan untuk menerjemahkan baris kode program menjadi kode mesin yang dapat dipahami oleh komputer. Bahasa ini bersifat server-side dan dapat disisipkan ke dalam HTML.”(Supono dan Putratama,V.2018:3)

“PHP adalah sebuah bahasa pemrograman yang dirancang untuk dijalankan di sebuah halaman website, biasanya digunakan untuk memproses informasi yang ada di internet. Dalam pengertian lain, PHP memiliki kepanjangan yaitu Hypertext Preprocessor, yang merupakan bahasa pemrograman open source atau gratis yang mana dapat dijalankan server-side. PHP berfungsi sebagai skrip yang terintegrasi dengan HTML dan dijalankan di server.”(Pahlevi et al., 2018)

Dari pengertian yang sudah dijelaskan, dapat diambil kesimpulan bahwa PHP adalah salah satu Bahasa pemrograman yang bersifat *Open-Source* atau bisa diakses siapapun yang berfungsi untuk membuat *website*

dinamis dan juga memiliki sifat *server-side* yang bisa dimasukkan ke dalam HTML.

2.3.4 JavaScript

“*JavaScript* adalah salah satu teknologi utama di World Wide Web, bersama dengan HTML dan CSS. Bahasa pemrograman ini berfungsi untuk membuat halaman web yang interaktif dan menjadi komponen penting dalam aplikasi web. Pada awalnya, JavaScript hanya diterapkan sebagai *client-side* di dalam peramban web.” (Limbong,T.2021:69)

2.4 Database

“*Database* adalah kumpulan data yang terorganisir dan saling terkait, dirancang agar dapat diakses dengan cepat dan mudah di masa mendatang. Data tersebut disimpan dalam format file, tabel, atau arsip di media penyimpanan elektronik, sehingga memudahkan pengaturan, pemilahan, pengelompokan, dan organisasi data sesuai dengan tujuan yang diinginkan. Ada delapan operasi dasar dalam database, yaitu *Create database, Drop database, Create table, Drop table, Insert, Read, Update, dan Delete.*” (Rachmadi, Tri.2020)

“*Database* adalah sekumpulan data yang saling berhubungan dan tersimpan di satu lokasi, digunakan oleh sistem aplikasi yang dikelola secara terpusat. Data tersebut memiliki nilai penting bagi pemiliknya.”(Setiawan,Didik.2017:9)

Sebuah *Database* menyimpan informasi berbentuk tabel, di mana setiap tabel terdiri dari baris dan kolom. Data yang disimpan dapat berupa angka atau fakta. Tabel tersebut memiliki informasi banyak hal, misalnya informasi terkait barang atau dosen. Tiap baris dalam tabel yang menyimpan data disebut dengan

record, sedangkan kolom yang berisi karakteristik yang sama untuk semua baris disebut fields

- a) **Fields** merupakan sekumpulan karakter yang menyusun sebuah makna. Sebagai contoh, jika ada *field* seperti NomorBrg atau NamaBrg, maka data yang dimasukkan ke dalam *field* tersebut harus relevan dengan nomor atau nama barang. Definisi lain dari *field* adalah kolom dalam tabel yang digunakan untuk memasukkan nama-nama (data) yang akan diisi ke dalam *field* tersebut.
- b) **Record** adalah sekumpulan field yang menyeluruh, biasanya diukur dalam satuan baris. Sebuah tabel terdiri dari beberapa record dan field. (Rachmadi, Tri.2020)

2.4.1 MySql

“MySQL adalah jenis database yang sering digunakan untuk mengembangkan aplikasi web dinamis. Termasuk dalam RDBMS (Relational Database Management System), MySQL mendukung bahasa pemrograman PHP dan memiliki query sederhana menggunakan SQL (Structured Query Language).”(Pahlevi et al., 2018)

2.5 Pemodelan Sistem dengan UML

Menurut Sukamto & Salahudin pada (2018), UML merupakan bahasa visual yang berfungsi untuk memodelkan dan mengkomunikasikan informasi tentang sistem menggunakan diagram dan teks pendukung, dan penggunaannya terutama difokuskan pada pemodelan.(Muharni,Sita.2021:52)


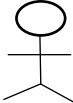


UML merupakan sebuah bahasa yang digunakan untuk merinci, memvisualisasikan, mengembangkan, dan menyimpan artefak dari sistem

perangkat lunak. Di dalam artefak tersebut, terdapat model, deskripsi, dan perangkat lunak yang dihasilkan selama tahap pengembangan perangkat lunak, yang juga meliputi pemodelan bisnis dan sistem di luar perangkat lunak. Selain itu, UML berfungsi sebagai bahasa pemodelan yang mengadopsi konsep-konsep orientasi objek. (Destriana,Rachmat,dkk.2021:1-2)

2.5.1 UseCase Diagram

Sebagai model pemodelan, use case menggambarkan perilaku dari sistem informasi yang akan dibangun. Ini menjelaskan interaksi antara satu atau lebih aktor dan sistem tersebut, serta berfungsi untuk mengidentifikasi fungsi yang tersedia dan pengguna yang berhak mengaksesnya. Simbol-simbol yang terdapat dalam diagram use case termasuk aktor, use case, hubungan komunikasi, dan generalisasi. (Rosa,AS dan Shalahuddin,M.2016:155) :

Tabel 2.5 Komponen *UseCase* Diagram



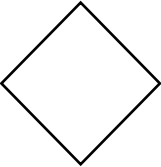

Simbol	Deskripsi
<i>Usecase</i> 	Aktivitas yang dilakukan oleh aktor dalam suatu sistem.
Aktor / <i>Actor</i> 	Entitas yang berhubungan dengan sistem yang telah dirancang.
Asosiasi / <i>Association</i> 	Interaksi antara <i>usecase</i> dan juga aktor yang terlibat dalam sistem..
Generalisasi / <i>Generalization</i> 	Relasi antar <i>usecase</i> yang menunjukkan spesialisasi dan generalisasi, di mana suatu

	use case merupakan versi yang lebih umum dibandingkan yang lainnya.
--	---

2.5.2 Activity Diagram

Diagram aktivitas (*activity diagram*) memvisualkan alur kerja (*workflow*) atau kegiatan yang terjadi pada sebuah sistem, proses bisnis, atau fitur pada perangkat lunak yang mendukung fungsi-fungsi tertentu di dalam aplikasi. Hal yang perlu diperhatikan di sini adalah bahwa diagram aktivitas menggambarkan kegiatan dalam sistem. Di bawah ini merupakan simbol-simbol yang terdapat dalam diagram aktivitas. (Rosa,AS dan Shalahuddin,M.2016:162):

Tabel 2.6 Komponen Activity Diagram

Simbol	Deskripsi
Status Awal 	Sebuah simbol status untuk mengawali sebuah diagram aktifitas.
Aktivitas 	Aktivitas yang dilakukan oleh sistem biasanya digambarkan dengan tindakan tertentu, yang umumnya dimulai dengan kata kerja.
Percabangan / <i>Decision</i> 	Percabangan terjadi saat ada beberapa pilihan aktivitas yang dapat dipilih dalam alur proses.
Penggabungan / <i>Join</i> 	

	Ini digunakan ketika ada banyak aktivitas yang terjadi dalam satu alur, dan digunakan untuk menggabungkan alur aktivitas tersebut.
<p>Status akhir</p> 	Status akhir dalam aktivitas sistem menunjukkan titik di mana proses berakhir. Setiap diagram aktivitas akan memiliki satu status akhir yang menandakan penyelesaian alur kerja.
<p>Swimlane</p> 	Pemecahan organisasi bisnis menunjukkan pembagian tanggung jawab atas aktivitas tertentu, biasanya digambarkan dengan swimlane dalam diagram aktivitas.


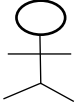


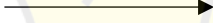
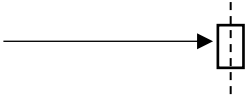
2.5.3 Sequence Diagram

Diagram sekuen mencerminkan tindakan suatu objek dalam *usecase* dengan menunjukkan siklus hidup objek serta pesan yang disampaikan dan dapat diterima antar objek. Berikut adalah simbol-simbol yang terdapat pada diagram sekuen.

(Rosa,AS dan Shalahuddin,M.2016:162): :

Tabel 2.7 Komponen *Sequence Diagram*

Simbol	Deskripsi
--------	-----------

<p>Garis hidup / <i>lifeline</i></p> 	<p>Menandakan siklus hidup sebuah objek.</p>
<p>Actor / Aktor</p> 	<p>Entitas, proses, atau sistem lain yang melakukan interaksi dengan sistem yang telah dibuat, berada pada luar sistem tersebut.</p>
<p>Objek</p> <div style="border: 1px solid black; padding: 5px; width: fit-content; margin: 10px auto;"> <p>Nama objek : nama kelas</p> </div>	<p>Merupakan objek yang melakukan interaksi dengan pesan.</p>
<p>Waktu aktif</p> 	<p>Menandakan suatu objek itu sedang mengirim atau menerima pesan.</p>
<p>Create / Membuat</p> <p style="text-align: center;"><<create>></p> 	<p>Menjelaskan bahwa sebuah objek membuat objek lain, dengan arah panah yang menunjukkan objek yang dibuat.</p>
<p>Call / Memanggil</p> <p style="text-align: center;">1::nama metode()</p> 	<p>Menjelaskan bahwa suatu objek memanggil sebuah operasi atau metode yang terdapat di objek lain atau pada dirinya sendiri.</p>
<p>Destroy</p> <p style="text-align: center;"><<Destroy>></p> 	<p>Mengartikan sebuah objek mengakhiri hidup objek lain.</p>

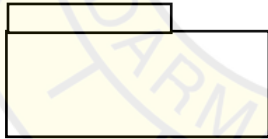
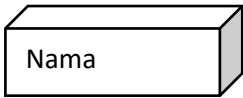
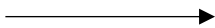
2.5.4 Deployment Diagram

Diagram ini menggambarkan Proses eksekusi aplikasi dipresentasikan melalui penggambaran konfigurasi komponen-komponennya. Hal-hal berikut dapat dimodelkan dengan menggunakan diagram deployment :

- a) Sistem tertanam yang menunjukkan desain perangkat, node, dan perangkat keras.
- b) Sistem berbasis client/server
- c) Pengembangan ulang aplikasi
- d) Sistem yang terdistribusi sepenuhnya.

Berikut adalah simbol-simbol yang terdapat pada diagram ini (Rosa,AS dan Shalahuddin,M.2016:162) :

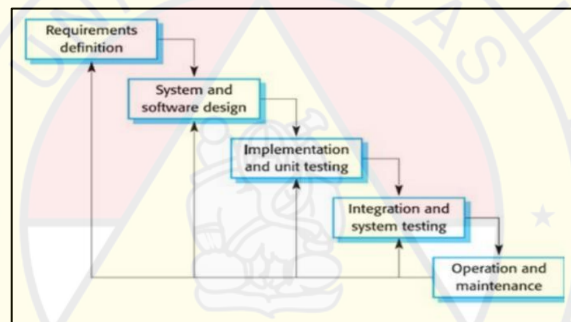
Tabel 2.8 Komponen *Deployment Diagram*

Simbol	Deskripsi
<p><i>Package</i></p> 	<p><i>Package</i> adalah sebuah wadah yang berisi satu atau lebih node, yang digunakan untuk mengelompokkan elemen-elemen terkait dalam satu unit untuk tujuan organisasi dan manajemen dalam diagram deployment.</p>
<p><i>Node</i></p> 	<p>Umumnya merujuk pada perangkat keras dan perangkat lunak yang bukan hasil pengembangan internal. (<i>software</i>).</p>
<p>Ketergantungan / <i>dependency</i></p> 	<p>Ketergantungan antar node, ditunjukkan dengan arah panah yang mengarah ke node yang digunakan.</p>
<p><i>Link</i></p>	<p>Hubungan antara <i>node</i> yang digunakan.</p>

--	--

2.6 Metode Pengembangan Sistem

Dalam pengembangan sistem ini, penulis mengadopsi metodologi *Waterfall*. **Metodologi *Waterfall***, atau model SDLC air terjun, dikenal juga sebagai model sekuensial linier atau alur hidup klasik. Pendekatan ini mengatur pengembangan perangkat lunak secara berurutan, dimulai dari analisis, desain, pengkodean, pengujian, hingga tahap pendukung atau pemeliharaan). (Tabrani & Pudjiarti, 2017)



Gambar 2.2 Metode Waterfall

Penjelasan dari urutan gambar di atas adalah sebagai berikut :

2.6.1 *Requirements analysis and definition*

Layanan sistem, batasan, dan tujuan ditentukan melalui diskusi dengan *user*, yang nantinya bisa diartikan secara lengkap untuk digunakan sebagai acuan spesifikasi sistem yang dibuat.

2.6.2 *System and software design*

Tahapan ini bertujuan untuk membagi kebutuhan perangkat lunak dan perangkat keras dengan membuat rancangan keseluruhan sistem. Proses pembuatan perangkat lunak mencakup identifikasi

dan penggambaran tentang abstraksi dari sistem dasar, serta interaksi antar perangkat lunak yang terlibat.

2.6.3 Implementation and unit testing

Pada tahap ini, perancangan perangkat lunak diwujudkan dalam bentuk serangkaian program atau unit program. Pengujian dilakukan untuk memastikan bahwa setiap unit memenuhi spesifikasi yang telah ditetapkan.

2.6.4 Integration and system testing

Unit-unit program yang terpisah digabung dan diuji sebagai satu sistem lengkap untuk memastikan kesesuaiannya dengan kebutuhan perangkat lunak. Setelah pengujian selesai, perangkat lunak dapat diserahkan kepada pelanggan.

2.6.5 Operation and maintenance

Tahapan ini biasanya, meskipun tidak selalu, merupakan yang paling panjang. Pada fase ini, sistem dipasang dan digunakan secara nyata. Pemeliharaan mencakup perbaikan kesalahan yang mungkin terlewat pada tahapan sebelumnya, peningkatan implementasi unit sistem, serta penyesuaian layanan sistem sesuai dengan kebutuhan baru.