

BAB II

LANDASAN TEORI

2.1. Data Mining

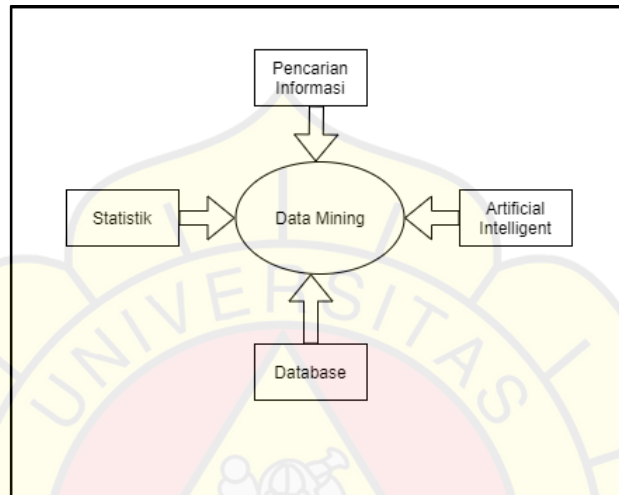
Menurut (Efori Buulolo, 2020) dalam e-book yang berjudul “Data mining untuk Perguruan Tinggi”. Data Mining atau kadang disebut juga *Knowledge Discovery in Database* (KDD) merupakan aktivitas yang berkaitan dengan pengumpulan data, pemakaian data historis untuk menemukan pengetahuan, informasi, keteraturan, pola atau hubungan dalam data yang berukuran besar. Output dalam data mining dapat dipergunakan sebagai alternatif dalam pengambilan keputusan atau untuk memperbaiki keputusan di masa yang akan datang.

Data mining salah satu bidang ilmu yang perkembangan sangat pesat, perkembangan tersebut disebabkan oleh beberapa faktor, antara lain :

1. . Semakin tingginya kesadaran akan pentingnya data
2. Semakin tingginya pemanfaat output dari hasil pengolahan data dalam berbagai bidang contohnya bidang bisnis.
3. Perkembangan kumpulan data yang begitu cepat.
4. Peningkatan akses internet baik melalui navigasi web ataupun melalui smartphone
5. Perkembangan hardware dan software khususnya yang berhubungan dengan data mining.
6. Perkembangan yang begitu cepat dalam bidang komputasi computer

7. Media penyimpanan yang semakin besar dengan harga yang semakin terjangkau.

Data mining bukanlah bidang ilmu yang berdiri sendiri, tetapi sangat berkaitan dengan bidang ilmu lain seperti database, statistic, pencarian informasi, dan artificial intelligent

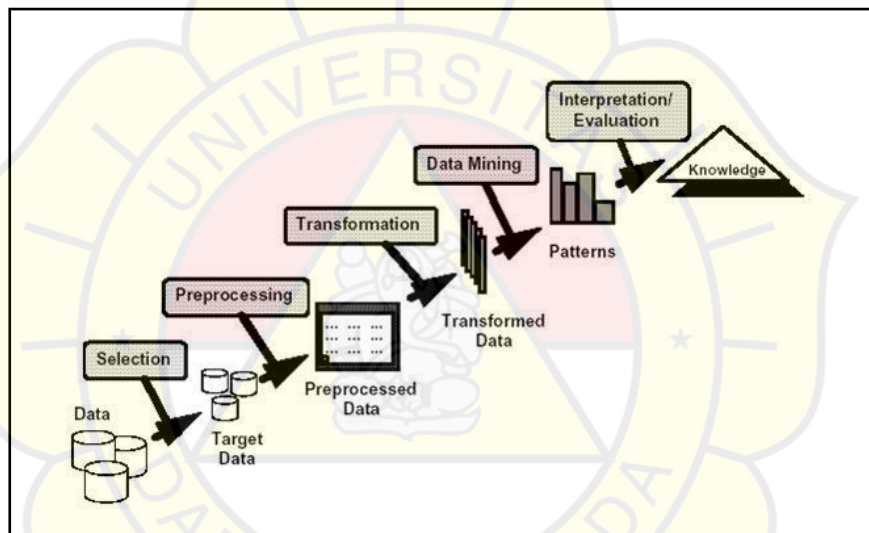


Gambar 2. 1Bidang Ilmu Data Mining

1. *Database – Data mining* : Kumpulan data yang digunakan dalam *data mining* bersumber salah satunya dari *database*. Data yang digali/dicari informasinya dipisahkan dari data operasional yang di *database*.
2. *Statistic – Data mining* : Dalam pengambilan keputusan, *statistic* membutuhkan data mulai dari pengumpulan data, pengambilan sampel data dan probabilitas. *Data mining* dalam penentuan sampel data, menganalisa, dan mempresentasikan output menggunakan teknik *statistic*.
3. *Pencarian Informasi – Data mining* : Pencarian informasi merupakan salahsatu kegiatan dalam proses *data mining* yang meliputi interpretasi, analisis dan penyimpanan data.

4. *Artificial Interlligent : Data mining* : Salah satu cabang ilmu dari *Artificial Intelligent* adalah *machine learning*. *Machine learning* merupakan disiplin ilmu yang penting dalam *data mining* dimana system computer belajar dari *training* data yang digunakan.

Data Mining mempunyai beberapa model proses yang digunakan untuk mengarahkan pelaksanaan data mining, model proses yang biasa digunakan adalah *Knowledge Discovery Databases (KDD)*, *CRISP-DM* dan *SEMMA*.



Gambar 2. 2 Tahapan KDD

Berikut adalah penjelasan dari setiap tahapan pada KDD :

a. *Domain Understanding and KDD Goals.*

Tujuan ditentukan dari sudut pandang user dan digunakan untuk mengembangkan dan pemahaman tentang domain aplikasi dan pengetahuansebelumnya.

b. *Selection and Additions*

Tahap kedua berfokus pada penentuan data target dan subset dari data sampel atau variabel.

c. *Preprocessing: Data Cleaning etc*

Pembersihan dan *preprocessing* data merupakan operasi dasar untuk menyelesaikan data yang konsisten tanpa *noisy*.

d. *Transformation*

Transformasi data dari satu bentuk ke bentuk lainnya sehingga data diimplementasikan dengan mudah.

e. *Data Mining (Chosing the Suitable Data Mining Task)*

Memilih metode *data mining* yang sesuai berdasarkan tujuan tertentu yang telah didefinisikan pada tahap pertama, contoh dari metode *data mining* adalah *classification*, *regression*, *clustering* dan *summarization*.

f. *Data Mining (Chosing the Suitable Data Mining Algorithm)*

Memilih algoritma yang tepat untuk pencarian pola-pola data, algoritma yang dipilih berdasarkan kecocokan kriteria dengan metode *data mining*.

g. *Data Mining (Implying Data Mining Algorithm)*

Pada tahap ini algoritma yang telah dipilih diimplementasikan.

h. Evaluation and Interpretation

Tahap ini berfokus pada interpretasi dan evaluasi yang mencakup pemeriksaan apakah pola atau informasi yang ditemukan bertentangan dengan hipotesa yang ada sebelumnya.

i. Discovered Knowledge

Penggunaan pengetahuan yang ditemukan dari proses KDD, dimana memutuskan apa yang akan dilakukan dengan pengetahuan dihasilkan.

2.2 Teknik Data Mining

Menurut Vulandari (2017) teknik yang digunakan dalam *data mining* erat kaitannya dengan “penemuan” (*discovery*) dan “pembelajaran” (*learning*) yang terbagi dalam tiga metode utama pembelajaran yaitu:

1. Supervised Learning

Supervised Learning adalah teknik yang paling sering digunakan. Teknik ini sama dengan “*programming by example*”. Teknik ini melibatkan fase pelatihan dimana data pelatihan historis yang karakter-karakternya dipetakan ke hasil yang telah diketahui diolah dalam algoritma data mining. Proses ini melatih algoritma untuk mengenali variabel-variabel dan nilai-nilai kunci yang nantinya akan digunakan sebagai dasar dalam membuat perkiraan-perkiraan ketika diberikan data baru.

2. *Unsupervised Learning*

Teknik pembelajaran ini tidak melibatkan fase pelatihan seperti yang terdapat pada *supervised learning*. Teknik ini bergantung pada penggunaan algoritma yang mendeteksi semua pola seperti *associations* yang muncul dari kriteria penting yang spesifik dari data masukan. Pendekatan ini mengarah pada pembuatan banyak aturan (*rules*) yang mengkarakterisasikan penemuan seperti *associations*, *clusters* dan *segments*. Aturan-aturan ini kemudian dianalisis untuk menemukan hal-hal penting.

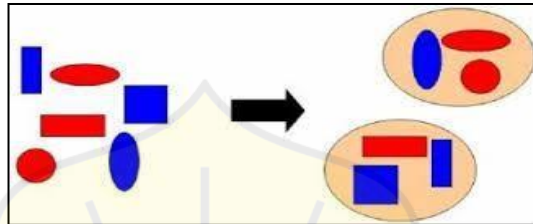
3. *Reinforcement Learning*

Teknik pembelajaran ini jarang digunakan dibandingkan dengan dua teknik lainnya, namun memiliki penerapan-penerapan yang terus dioptimalkan dari waktu ke waktu dan memiliki kontrol adaptif. *Reinforcement learning* sangat tepat digunakan untuk menyelesaikan masalah-masalah yang sulit serta bergantung pada waktu.

2.3 Teknik Clustering

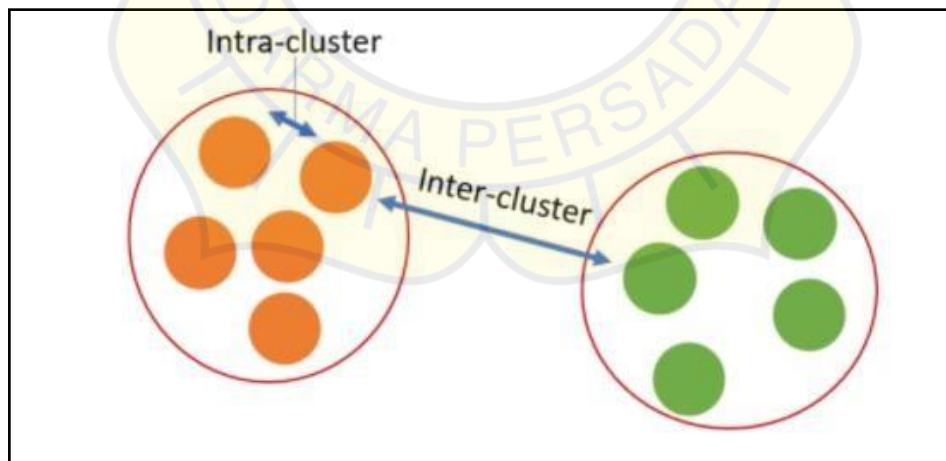
Teknik *Clustering* atau pengklusteran merupakan teknik yang membagi data menjadi kelompok-kelompok (Kusuma and Andriani, 2020). Pembagian kelompok data tersebut berdasarkan kriteria tertentu.

Tujuan dari teknik *clustering* adalah mengelompokkan data yang mempunyai kriteria sama ke dalam satu wilayah atau kelompok yang sama dan memisahkan data yang mempunyai perbedaan kriteria. Data akan dikelompokkan dalam satu kelompok atau *cluster* mempunyai tingkat kemiripan yang maksimum dan setiap kelompok mempunyai tingkat kemiripan yang minimum.



Gambar 2. 3Penggambaran Teknik Clustering

Pada gambar 2.4 Mengilustrasikan data yang dikelompokkan berdasarkan kriteria bentuknya. *Cluster* yang baik ketika jarak antara objek data yang ada pada *cluster* yang sama (*intra-cluster*) dapat diminimalkan dan jarak antara objek yang ada pada *cluster* yang berbeda (*inter-cluster*) dapat dimaksimalkan.



Gambar 2. 4Perbedaan intra-cluster dengan inter-cluster

Berikut merupakan beberapa contoh implementasi teknik *clustering* dalam dunia nyata (Aggarwal, 2016):

1. *Data summarization* atau peringkasan data. *Data mining* berkaitan dengan meringkas informasi dari data sehingga proses *clustering* sering kali merupakan langkah awal dari banyak algoritma *data mining*.
2. *Customer segmentation* atau segmentasi konsumen yang sering kali digunakan untuk menganalisis perilaku umum dari kelompok konsumen yang serupa. Sebagai contoh pola perilaku pemilihan produk suatu konsumen yang ada pada suatu kelompok dapat digunakan untuk merekomendasikan produk tersebut kepada konsumen lain yang masih dalam satu kelompok.
3. *Social network analysis* atau analisis jaringan sosial yang digunakan untuk menganalisis perilaku manusia yang diperoleh dari menganalisis dinamika yang terjadi pada komunitas. Jaringan sosial merupakan simpul yang telah user kluster secara erat melalui hubungan yang saling terkait yang biasanya kelompok teman yang sama atau komunitas.
4. Berhubungan dengan masalah *data mining* yang lain.

Berikut merupakan metode yang umum digunakan pada teknik *clustering* antara lain:

5. *Partitioning method* merupakan teknik yang digunakan untuk mengelompokkan data sesuai dengan jumlah cluster atau kelompok yang ditentukan. Dalam hal ini jumlah kelompok ditentukan di awal. Algoritma yang populer digunakan pada *partitioning method* adalah K-means, K-medoids dan CLARA (*Clustering Large Application*).
6. *Hierarchical method* atau metode hierarki merupakan alternatif dari *partitioning method* dan tidak memerlukan menentukan jumlah cluster atau kelompok di awal. Terdapat dua pendekatan yaitu *Divisive* dan *Agglomerative*. Pendekatan *divisive* yang merupakan pendekatan *top-down*. Semua objek data dikumpulkan dalam satu *cluster* dan selanjutnya dipecah menjadi *cluster* yang lebih kecil menggunakan iterasi berulang hingga kondisi selesai tercapai. Sedangkan pendekatan *agglomerative* merupakan pendekatan *bottom-up* di mana setiap kelompok data terpisah sejak awal dan digabungkan hingga kondisi selesai tercapai.

2.4 Teknik Klasifikasi

Menurut (Dito Putro Utomo & Mesran, 2020) dalam jurnal Media Informatika Budidarma dengan judul “Analisis Komparasi Metode Klasifikasi Data Mining dan Reduksi Atribut Pada Data Set Penyakit Jantung”.

Klasifikasi merupakan suatu pekerjaan menilai objek data untuk memasukkannya kedalam kelas tertentu dari jumlah kelas yang tersedia. Klasifikasi melakukan pembangunan model berdasarkan data latih yang ada, kemudian menggunakan model tersebut untuk mengklasifikasikan pada data yang baru. Klasifikasi dapat didefinisikan sebagai pekerjaan yang melakukan pelatihan/pembelajaran terhadap fungsi target yang memetakan setiap set atribut (fitur) ke satu jumlah label kelas yang tersedia. Sebuah sistem yang melakukan klasifikasi diharapkan dapat melakukan klasifikasi semua data set dengan benar, tetapi tidak dapat dipungkiri bahwa kinerja sistem tidak bisa 100% benar sehingga sebuah sistem klasifikasi juga harus diukur kinerjanya. Umumnya, pengukuran kinerja klasifikasi dilakukan dengan matriks konfusi.

2.5 Pemrograman Aplikasi

2.5.1 Web

Menurut Agus Hariyanto (2015), Web dapat diartikan sebagai kumpulan halaman yang menampilkan informasi data teks, data gambar, data animasi, suara, video dan gabungan dari semuanya, baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling terkait, dimana masing-masing dihubungkan dengan jaringan-jaringan halaman (*hyperlink*).

Menurut Rohi Abdulloh (2015), Web adalah sekumpulan halaman yang terdiri dari beberapa halaman yang berisi informasi dalam bentuk data digital baik berupa text, gambar, video, audio, dan animasi lainnya yang disediakan melalui jalur koneksi internet.

Menurut (Mara Destiningrum, Qadhli Jafar Adrian, 2017) Web adalah Sebuah *software* yang berfungsi untuk menampilkan dokumen - dokumen pada suatu web yang membuat pengguna dapat mengakses internet melalui *software*

2.5.2 HTML

“*HyperText Markup Language* (HTML) adalah sebuah bahasa *markup* yang digunakan untuk membuat sebuah halaman web, menampilkan berbagai informasi di dalam sebuah Penjelajah web Internet dan *formatting hypertext* yang sederhana ditulis kedalam berkas format ASCII agar dapat menghasilkan tampilan wujud yang terintegrasi. Dengan kata lain, berkas yang dibuat dalam perangkat lunak pengolah kata dan disimpan kedalam format ASCII normal sehingga menjadi *home page* dengan perintah perintah HTML. Bermula dari sebuah bahasa yang sebelumnya banyak digunakan di dunia penerbitan dan percetakan yang disebut dengan SGML(*Standard Generalized Markup Language*), HTML adalah sebuah standar yang digunakan secara luas untuk menampilkan halaman web. HTML saat ini merupakan standar Internet yang didefinisikan dan dikendalikan penggunaannya oleh *World Wide Web Consortium*(W3C). HTML dibuat oleh kolaborasi Caillau TIM dengan Berners-lee robert ketika mereka bekerja di CERN pada tahun 1989 (CERN adalah lembaga penelitian fisika energi tinggi di Jenewa)” (Ahmad dan Syarif, 2016).

2.5.3 CSS

CSS (*Cascading Style Sheet*) merupakan suatu bahasa program web yang digunakan untuk mengendalikan dan membangun berbagai komponen dalam web yang digunakan untuk mengendalikan dan membangun berbagai komponen dalam web sehingga tampilan web akan lebih rapi, terstruktur dan seragam.

CSS saat ini dikembangkan oleh *World Wide Web Consortium* (W3C) dan menjadi bahasa standar dalam pembuatan web. CSS difungsikan sebagai penopang atau pendukung, dan pelengkap dari file html yang berperan dalam penataan kerangka dan *layout*.

CSS mampu di berbagai platform, maksudnya dapat dijalankan pada berbagai macam sistem operasi dan web browser. Secara umum, yang dilakukan oleh CSS adalah pengaturan layout, kerangka, teks, gambar, warna, tabel, spasi, dan lain sebagainya. Kita akan mengetahui masing-masing pengaturan tersebut pada pembahasan selanjutnya. (M.Taofik Chulkamdi, SulisPurnomo, 2016).

2.5.4 JavaScript

JavaScript merupakan bahasa *scripting* yang pada awalnya dikembangkan oleh Netscape. Dengan menggunakan javascript, dapat dibuat tampilan web yang lebih interaktif lagi. Agar user dapat menjalankan javascript, alat yang dibutuhkan hanyalah *browser* yang kemampuan javascript-nya telah diaktifkan. Penulisan javascript dapat dipadukan pada halaman HTML, dan darisini dapat dilihat bahwa javascript merupakan bahasa pemrograman yang berjalan di sisi *client*. Karena proses dari javascript tergantung proses dari *browser* yang digunakan oleh *user*. Contoh penggunaan JavaScript adalah sebagai berikut:

```
<html>
<body>

<script language="javascript">

document.write("Ini dihasilkan oleh javascript");

</script>

</body>

</html>
```

2.5.5 Bootstrap

Bootstrap adalah framework yang kuat menyediakan set kelas CSS dan fungsi JavaScript untuk memudahkan proses pembangunan antarmuka halaman web. Mengaktifkan fitur *design* responsif dukungan untuk menampilkan desktop maupun mobile. Situs dikembangkan dapat bekerja dengan baik pada desktop maupun mobile. Developer tidak harus bekerja dengan CSS untuk membuat website terlihat menarik atau mendukung prinsip desain *responsive*, kecuali diperlukan . Twitter Bootstrap dapat diunduh secara gratis di website resminya.

yakni www.getbootstrap.com, setelah itu tinggal memanggil file CSS Bootstrap pada file project website yang akan menggunakan Bootstrap. Begituselesai memanggil Bootstrap, maka secara otomatis akan mengubah tampilan website tanpa harus melakukan pengetikan sintak-sintak CSS seperti biasa dilakukan” (Alfan dan Bamban, 2016). Bootstrap merupakan framework yang disediakan sediakan oleh twitter untuk membangun sebuah website. Untuk dapat menggunakan CSS serta fungsi java script yang terkandung pada

bootstrap, terdapat template HTML yang disediakan oleh bootstrap yang bisa didapatkan melalui website resmi bootstrap . Cara menggunakannya adalah dengan menyalin kode-kode dari template tersebut, kemudian edit kode-kode tersebut sesuai dengan kebutuhan. Sebagai batasan, ada beberapa kode yang tidak boleh diubah agar tetap selalu dapat menggunakannya, seperti misalnya pemanggilan link yang terdapat pada tag head.

2.5.6 PHP

PHP atau (*hypertext preprocessor*), merupakan bahasa pemrograman pada sisi server yang memperbolehkan programmer menyisipkan perintah-perintah perangkat lunak web server akan dieksekusi sebelum perintah itu dikirim oleh halaman ke browser yang memintanya. Sesuai dengan fungsinya php berjalan di sisi server maka php adalah bahasa pemrograman yang digunakan untuk membangun teknologi web application. Oleh karena itu, salah satu tool yang harus tersedia sebelum memulai pemrograman PHP adalah server (Jubilee, 2017). Dengan menggunakan bahasa pemrograman PHP sebuah website dapat menjadi website dinamis. Tidak seperti HTML yang bertipe statis. PHP dirancang untuk dapat bekerja sama dengan database server dan dibuat sedemikian rupa sehingga pembuatan dokumen HTML yang dapat mengakses database menjadi begitu mudah. Tujuan dari bahasa scripting ini adalah untuk membuat aplikasi di mana aplikasi tersebut yang dibangun oleh PHP pada umumnya akan memberikan hasil pada web browser, tetapi prosesnya secara keseluruhan dijalankan di *server*.

2.5.7 Database

Menurut Mukhamad Masrur (2016), Database adalah sekumpulan file data yang satu sama lainnya saling berhubungan yang diorganisasi sedemikian rupa sehingga memudahkan untuk mendapatkan dan memproses data tersebut. Lingkungan sistem database menekankan pada data yang tidak tergantung (*independent*) pada aplikasi yang akan menggunakan data tersebut.

Penggunaan database pada komputer dilakukan dengan menggunakan table-table. Pada tabel-tabel tersebut masih dikelompokkan lagi menjadi beberapa bagian untuk membedakan data yang satu dengan data yang lain. Pada sebuah tabel database harus memiliki setau kategori data yang digunakan sebagai kunci untuk membedakan data-data yang ada didalam satu tabel. Data kunci tersebut tidak boleh sama antara satu data dengan data lainnya. Data kunci sering disebut dengan *Primary Key*.

2.5.8 mysql

MySQL adalah suatu sistem manajemen database (*Relational Database Management System*) bersifat terbuka (*Open Source*). Maksudnya terbuka adalah mysql boleh di-download oleh siapa saja baik versi kode program aslinyamaupun versi binernya dan bisa digunakan secara gratis. Sejarah Mysql yang merupakan hasil buah pikiran dari Michael "Monty" Widenius, David Axmark dan Allan Larson dimulai tahun 1995. mereka bertiga mendirikan perusahaan bernama Mysql di Swedia. (Lukman,2015:148). Dengan berbagai keunggulan yang dimiliki, membuat software database ini banyak digunakan oleh praktisi untuk membangun suatu project. Adanya fasilitas API (*Application Programming Interface*) yang dimiliki oleh Mysql, memungkinkan bermacam

2.6 Algoritma Sistem

2.6.1 Algoritma Arima

Moving Average sering juga disebut model rata-rata bergerak terpadu autoregresif. Model moving average (MA) pertama kali digunakan oleh Slutsky pada tahun 1917. Akan tetapi Wold lah pada tahun 1938 yang menghasilkan dasar-dasar teoritis dari proses kombinasi ARMA. Wold membentuk model MA yang dikembangkan pada tiga arah identifikasi efisien dan prosedur penaksiran (untuk proses AR dan MA). Sedangkan menurut pendapat Victor E. McGee defines dari Moving Average adalah terdapat dua arti berbeda untuk istilah ini, yang pertama untuk deret berkala kita dapat definisikan rata-rata bergerak ordo K sebagai nilai rata-rata dari K pengamatan terakhir. Kedua dalam pemodelan MA adalah “rata-rata bergerak” dan berarti bahwa nilai deret berkala pada waktu dipengaruhi oleh unsur kesalahan pada saat ini dan mungkin unsur kesalahan terbobot pada masa lalu. Model Ma merupakan model yang menggambarkan bahwa variabel tersebut dipengaruhi oleh nilai residual waktu yang lampau. Dengan demikian model MA pada prinsipnya sama dengan model regresi tetapi yang bertindak sebagai variabel bebas adalah variabel residual periode sebelumnya. Model umum dari Moving Average (MA) adalah sebagai berikut :

$$Y_t = c + e_t + \theta_1 e_{t-1} + \theta_2 e_{t-2} + \dots + \theta_{k-1} e_{t-k+1} + \theta_k e_{t-k}$$

Di mana:

Y_t = Nilai variabel dependen pada waktu t

e = Konstanta

e_t = Nilai residual pada period ke t

m_1, m_2, m_3 = Koefisien atau parameter dari model Moving Average

$e_{t-1}, e_{t-2}, e_{t-q}$ = Nilai residual sebelumnya (lag)

Besarnya orde model moving average tergantung pada jumlah lag residual yang dimasukkan ke dalam model persamaan. Untuk lebih jelasnya, lihat model berikut ini:

$$Y_t = c + e_t - m_1 e_{t-1}$$

Model tersebut merupakan model MA orde 1 atau ARIMA (0,0,1).

$$Y_t = c + e_t - m_1 e_{t-1} - m_2 e_{t-2}$$

Model tersebut merupakan model MA orde 2, atau ARIMA (0,0,2).

2.6.2 Tahap Identifikasi Model

Fungsi autokorelasi dan fungsi autokorelasi parsial dapat digunakan untuk mengetahui cirri, pola data dan jenis data, sehingga fungsi autokorelasi dan fungsi autokorelasi parsial dapat memenuhi maksud untuk mengidentifikasi suatu model tentative atau model sementara yang dapat disesuaikan dengan data. Atau dengan kata lain fungsi autokorelasi dan autokorelasi parsial yang melebihi batas interval penerimaan dapat digunakan untuk mengidentifikasi model ARIMA Box-Jenkins.

2.6.3 Fungsi Autokorelasi

Koefisien autokorelasi adalah menyatakan hubungan atau asosiasi antara nilai-nilai variabel Y_t dengan variabel Y_{t-k} . Menurut Pindyck dan Rubinfeld (1981) secara matematis rumus untuk koefisien autokorelasi dapat dituliskan dengan rumus seperti persamaan sebagai berikut : Koefisien autokorelasi dilambangkan dengan r_k , sementara koefisien autokorelasi dilambangkan dengan r_k^2 . Untuk menghitung autokorelasi parsial harus dihitung terlebih dahulu koefisien autokorelasinya. Koefisien autokorelasi dapat dihitung dengan menggunakan rumus berikut:

$$r_k = \frac{\sum_{t=k}^{n-k} (Y_t - \bar{Y})(Y_{t-k} - \bar{Y})}{\sum_{t=1}^n (Y_t - \bar{Y})^2}$$

Di mana:

- r_k = Koefisien autokorelasi tingkat k
- Y_t = Nilai observasi pada waktu t
- Y_{t-k} = Nilai observasi pada k periode sebelumnya (t-k)
- \bar{Y} = Nilai rata-rata serial data
- $y = Y_t - \bar{Y}$

Koefisien autokorelasi perlu diuji untuk menentukan apakah secara statistik nilainya berbeda secara signifikan dari nol atau tidak. Nilai Standat Error (SE) dari nilai r_k adalah:

$$SE_{r_k} = \left(\frac{1}{\sqrt{n}} \right)$$
$$-Z \left(\frac{1}{\sqrt{n}} \right) \leq r_k \leq Z \left(\frac{1}{\sqrt{n}} \right)$$
$$-1,96 \left(\frac{1}{\sqrt{n}} \right) \leq r_k \leq 1,96$$

2.7 Pemodelan UML

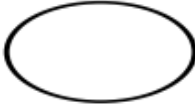





“*Unified Modeling Language (UML)* bukanlah suatu proses melainkan bahasa pemodelan secara grafis untuk menspesifikasikan, memvisualisasikan, membangun, dan mendokumentasikan seluruh artifak sistem perangkat lunak. Penggunaan model ini bertujuan untuk mengidentifikasi bagian-bagian yang termasuk dalam lingkup sistem dibahas dan bagaimana hubungan antara sistem suatu dengan subsistem maupun sistem lain di luarnya.” *Unified Modeling Language (UML)*” adalah sebuah bahasa yang berdasarkan grafik atau gambar untuk memvisualisasi, menspesifikasikan dari sebuah sistem pengembangan software berbasis object oriented.” Dari Pengertian diatas penulis menyimpulkan bahwa *Unified Modeling Language (UML)* merupakan bahasa pemodelan yang berbentuk grafis yang digunakan untuk memvisualisasi, menspesifikasikan suatu sistem perangkat lunak (Yuhanar, 2018).

2.7.1 Use Case Diagram

Diagram *use case* merupakan pemodelan untuk kelakuan (behavior) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Secara kasar, use case digunakan untuk mengetahui fungsi apa saja yang ada di dalam suatu sistem dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut.”(Yunahar, 2018).

Berikut adalah simbol simbol yang ada pada diagram use case:





Tabel 2. 1 Simbol *Use Case Diagram* (Yunahar, 2018)

Simbol	Deskripsi
	Fungsional yang disediakan sistem sebagai unit-unit yang saling bertukar antar unit atau aktor biasanya dinyatakan dengan menggunakan kata kerja di awal <i>frase</i> nama <i>use case</i>
	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, biasanya dinyatakan menggunakan kata benda di awal <i>frase</i> nama aktor
	Komunikasi antar aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor
	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan
	Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari yang lainnya
	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsinya atau sebagai syarat dijalankan <i>use case</i> ini

2.7.2 Activity Diagram

Activity Diagram menggambarkan work flow (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis. Yang perlu diperhatikan disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas dapat dilakukan oleh sistem” (Yunahar, 2018). Simbol-simbol yang digunakan dalam activity diagram sebagai berikut :

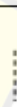


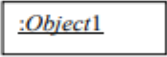

Tabel 2. 2 Simbol *Activity Diagram* (Yunahar, 2018)

Simbol	Deskripsi
	Simbol <i>start</i> untuk menyatakan awal dari suatu proses
	Simbol <i>stop</i> untuk menyatakan akhir dari suatu proses
	Simbol <i>decision</i> digunakan untuk menyatakan kondisi dari suatu proses
	Simbol <i>action</i> menyatakan aksi yang dilakukan dalam suatu arsitektur sistem

2.7.3 Sequence Diagram

“*Sequence Diagram* adalah tool yang sangat populer dalam pengembangan sistem informasi secara object-oriented untuk menampilkan interaksi antar objek. Berdasarkan definisi tersebut, dapat disimpulkan bahwa *Sequence Diagram* adalah tool yang digunakan dalam pengembangan system” (Yunahar, 2018).

Tabel 2. 3 Simbol *Sequence Diagram* (Yunahar, 2018)

Nama Komponen	Keterangan	Simbol
<i>Lifeline</i>	Mengindikasikan keberadaan sebuah objek dalam basis waktu. Notasi untuk <i>lifeline</i> adalah garis putus-putus vertikal yang ditarik dari sebuah objek	
<i>Activation</i>	Dinotasikan sebagai sebuah kotak segi empat digambar pada sebuah <i>lifeline</i> mengindikasikan sebuah objek yang akan melakukan sebuah aksi	
<i>Message</i>	Digambarkan dengan anak panah horizontal antara <i>activation</i> . <i>Message</i> mengindikasikan komunikasi antara objek-objek	
<i>Object</i>	Merupakan <i>instance</i> dari sebuah class dan dituliskan tersusun secara horizontal.	
<i>Actor</i>	<i>Actor</i> juga dapat berkomunikasi	

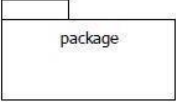
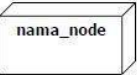


2.7.4 Deployment Diagram

Menurut Sukamto dan Shalahudin (2015) mengemukakan bahwa “*Deployment Diagram* menunjukkan konfigurasi komponen dalam proses eksekusi aplikasi”. *Deployment Diagram* dapat digunakan untuk menggambarkan hal-hal sebagai berikut:

1. Sistem tambahan (*embedded system*) yang menggambarkan rancangan *device, node dan hardware*.
2. Sistem *client* atau server.
3. Sistem terdistribusi murni.
4. Rekayasa ulang aplikasi.

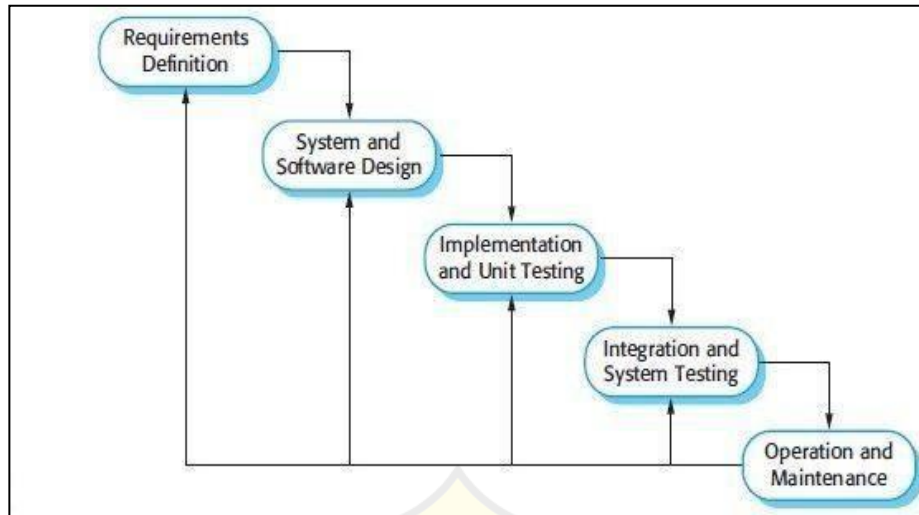
Menurut Hendini (2016) *Deployment Diagram* digunakan untuk menggambarkan detail bagaimana komponen disusun di infrastruktur system. Dapat disimpulkan bahwa *Deployment Diagram* merupakan diagram yang digunakan untuk menggambarkan konfigurasi pada sistem.

Tabel 2. 4 Simbol *Deployment Diagram* (Rosa dan M.Shalahuddin, 2016)

Simbol	Deskripsi
Package 	package merupakan sebuah bungkusan dari satu atau lebih <i>node</i>
Node 	biasanya mengacu pada perangkat keras (<i>hardware</i>), perangkat lunak yang tidak dibuat sendiri (<i>software</i>), jika di dalam <i>node</i> disertakan komponen untuk mengkonsistenkan rancangan maka komponen yang diikutsertakan harus sesuai dengan komponen yang telah didefinisikan sebelumnya pada diagram komponen
Kebergantungan / <i>dependency</i> 	Kebergantungan antar <i>node</i> , arah panah mengarah pada <i>node</i> yang dipakai
Link 	relasi antar <i>node</i>

2.8 Metode Pengembangan Sistem

Metode Waterfall adalah suatu proses pengembangan perangkat lunak berurutan, di mana kemajuan dipandang sebagai terus mengalir ke bawah (seperti air terjun) melewati fase-fase perencanaan, pemodelan, implementasi (konstruksi), dan pengujian. Dalam pengembangannya metode waterfall memiliki beberapa tahapan yang runtut: *requirement* (analisis kebutuhan), desain sistem (*system design*), *Coding & Testing*, Penerapan Program, pemeliharaan. (Chrisantus Trisianto. 2018. Jurnal Teknologi Informasi : Penggunaan Metode Waterfall Untuk Pengembangan Sistem Monitoring Dan Evaluasi Pembangunan Pedesaan).



Gambar 2. 5 Metodologi Waterfall (Rosa A. S dan M. Shalahuddin, 2016)

Adapun penjelasan urutan dari tahapan-tahapan yang dimiliki metodologi waterfall adalah sebagai berikut:

2.8.1 Requirement (Analisa Kebutuhan)

Dalam langkah ini merupakan analisa terhadap kebutuhan sistem. Pengumpulan data dalam tahap ini bisa melakukan sebuah penelitian, wawancara atau *study* literatur. Seseorang sistem analisis akan menggali informasi sebanyak-banyaknya dari user sehingga akan tercipta sebuah sistem komputer yang bisa melakukan tugas-tugas yang diinginkan oleh *user* tersebut. Tahapan ini akan menghasilkan dokumen *user requirement* atau bisa dikatakan sebagai data yang berhubungan dengan keinginan *user* dalam pembuatan sistem. Dokumen inilah yang akan menjadi acuan *system* analisis untuk menterjemahkan kedalam bahasa pemrograman.

2.8.2 Design System (Design Sistem)

Proses *design* akan menterjemahkan syarat kebutuhan ke sebuah perancangan perangkat lunak yang dapat diperkirakan sebelum dibuat koding. Proses ini berfokus pada struktur data, arsitektur perangkat lunak, representasi *interface*, dan detail (algoritma) prosedural. Tahapan ini akan menghasilkan dokumen yang disebut *software requirement*.

Dokumen inilah yang akan digunakan programmer untuk melakukan aktivitas pembuatan sistemnya.

2.8.3 Coding & Testing (Penulisan Sinkode Program/Implementation)

Coding merupakan penerjemahan *design* dalam bahasa yang bisa dikenali oleh komputer. Dilakukan oleh programmer yang akan menerjemahkan transaksi yang diminta oleh user. Tahapan inilah yang merupakan tahapan secara nyata dalam mengerjakan suatu sistem. Dalam artian penggunaan *computer* akan dimaksimalkan dalam tahapan ini. Setelah pengkodean selesai maka akan dilakukan testing terhadap sistem yang telah dibuat tadi. Tujuan *testing* adalah menemukan kesalahan-kesalahan terhadap *system* tersebut dan kemudian bisa diperbaiki.

2.8.4 Penerapan/Pengujian Program (Integration & Testing)

Tahapan ini bisa dikatakan final dalam pembuatan sebuah sistem. Setelah melakukan analisa, design dan pengkodean maka sistem yang sudah jadi akan digunakan oleh *user*.

2.8.5 Pemeliharaan (Operation & Maintenance)

Perangkat lunak yang sudah disampaikan kepada pelanggan pasti akan mengalami perubahan. Perubahan tersebut bisa karena mengalami kesalahan karena perangkat lunak harus menyesuaikan dengan lingkungan (peripheral atau sistem operasi baru) baru, atau karena pelanggan membutuhkan perkembangan fungsional.