

## **BAB II**

### **LANDASAN TEORI**

#### **2.1 Program Studi**

Program studi adalah kesatuan rencana belajar sebagai pedoman penyelenggaraan pendidikan akademik, profesional, dan atau profesi yang di selenggarakan atas dasar suatu kurikulum serta di tujukan agar mahasiswa dapat menguasai pengetahuan, keterampilan dan sikap, sesuai dengan sasaran kurikulum. Program penjurusan merupakan proses penempatan dan pemilihan program studi para siswa.

Penjurusan merupakan suatu proses yang akan menentukan keberhasilan para siswa, baik pada waktu belajar di SMA maupun setelah di perguruan tinggi. Maka diperlukan suatu bimbingan khusus yaitu bimbingan penjurusan. Di dalam penjurusan ini terdapat kaitan yang erat antara bimbingan penjurusan dengan bimbingan karir yaitu merupakan suatu proses yang bebas, meluas, dan berurutan. Para pembimbing diharapkan dapat mengarahkan siswa untuk memilih program jurusan yang sangat sesuai dengan diri siswa. Para pembimbing diharapkan pula memperhatikan ciri-ciri kepribadian siswa dan pengaruh lingkungan terhadap diri siswa yang bersangkutan kepribadian yang dimaksud adalah intelegensi, bakat, minat. Sedangkan faktor lingkungan adalah peran orang tersebut dan pendidikan. Pada faktor pendidikan meliputi aspek prestasi akademik, program pilihan jurusan, keadaan kelas, dan lain sebagainya. Kesesuaian karir ataupun program permintaan merupakan hal yang tak kalah penting dalam penempatan siswa di suatu program dalam teori Tipologi karir mengenai

prilaku vokasional berpendapat bahwa dalam membangun ketekaitan atau kecocokan antara tipe kepribadian individu dalam karir pemilihan sangat penting. Karena ini dari pemilihan dan penyesuaian karir merupakan gambaran kepribadian seseorang. Sehingga minat tertntu menjadi suatu ciri kepribadian yang berupa ekspresi diri dalam bidang karir, bidang studi akademik, dan hobi. Bahwa terdapat sejumlah lingkungan kerja yang memikat beberapa kepribadian. Jika lingkungan kerja itu sesuai dengan kepribadian orang yang memilihnya, ini bisa membuatnya berhasil dalam meniti karir. Pada intinya sebuah keberhasilan serta motivasi dalam sebuah aktivitas itu dipengaruhi atas kesesuaian kepribadian terhadap suatu karir ataupun pilihan dalam program permintaan pada SMA.

## 2.2 Data Mining

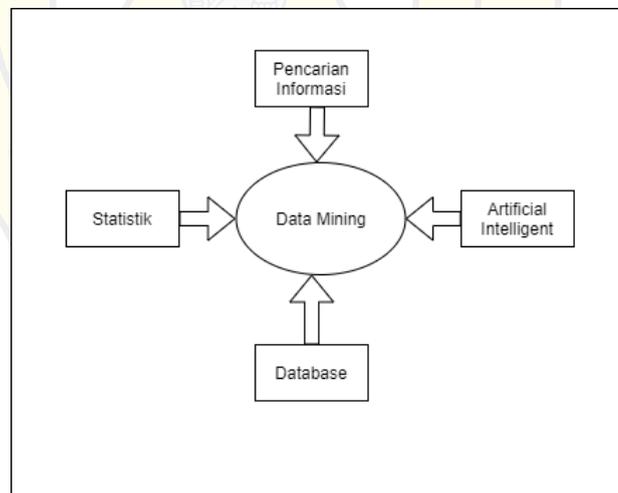
Menurut (Marisa 2014) *Data Mining* atau kadang disebut juga *Knowledge Discovery in Database (KDD)* merupakan aktivitas yang berkaitan dengan pengumpulan data, pemakaian data historis untuk menemukan pengetahuan, informasi, keteraturan, pola atau hubungan dalam data yang berukuran besar. *Output* dalam *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 factor, antara lain :

1. Semakin tingginya kesadaran akan pentingnya data
2. Semakin tingginya pemanfaat *output* dari hasil pengolahan datadalam 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 komputasicomputer.
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*.

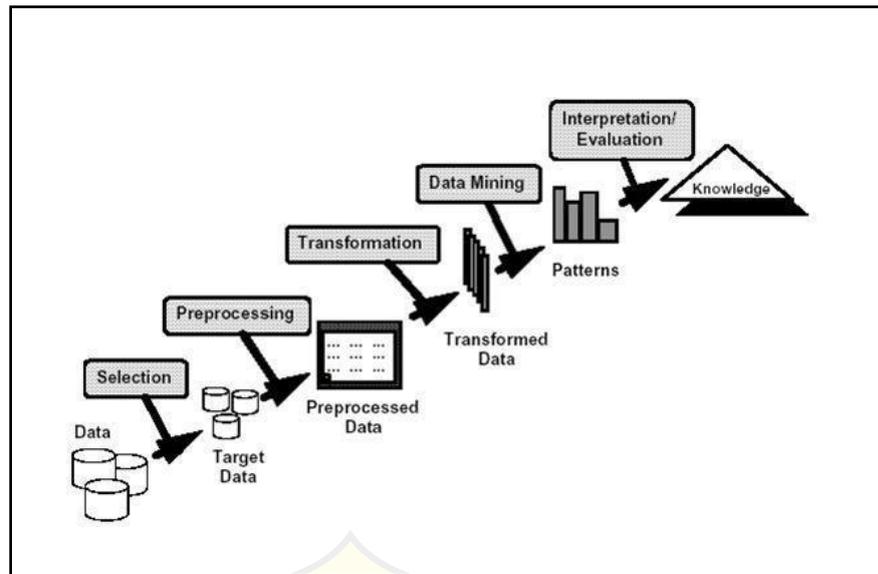


Gambar 2. 1 Bidang Data Mining

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 pengetahuan sebelumnya.

- 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 (Choosing 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 (Choosing 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 (Employing 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.

### 2.3 Kegunaan Data Mining

Menurut (Marisa 2014) Secara garis besar kegunaan data mining dibagi menjadi dua yaitu : deskriptif dan prediktif. Secara deskriptif artinya data mining berarti menemukan pola-pola yang digunakan untuk

menjelaskan karakteristik data. Sedangkan secara prediktif data mining berarti dapat digunakan untuk menemukan model-model pengetahuan yang digunakan untuk melakukan prediksi.

Data mining berdasarkan fungsionalitasnya dapat dikelompokkan menjadi enam bagian yaitu :

1. Klasifikasi (*classification*)

Diterapkan pada data baru untuk mengelompokkan jenis objek.

Klasifikasi termasuk pada model *supervised*. Pada persoalan klasifikasi kita memiliki sampel data dan memprediksi beberapa class yang ada berdasarkan sampel yang ada. Hanya satu atribut di antara banyak atribut yang disebut dengan atribut predator. Klasifikasi ini juga umum, digunakan untuk pemodelan bisnis dan lainnya. Misal klasifikasi menentukan penyakit tertentu atau menentukan customer berdasarkan model pembayarannya.

2. Klastering (*Clustering*)

Berbeda dengan klasifikasi, klastering termasuk model *unsupervised*. Klastering mengelompokkan data yang tidak diketahui labelnya. Klastering yang diorganisasi ke dalam struktur hierarkikal akan mendefinisikan taksonomi dari data. Penerapan metode klastering yang tepat akan menghasilkan clustering yang berkualitas. Suatu *cluster* dikarakterisasi oleh *centroid*, atribut histogram dan *clustering model hierarchial tree*.

3. Regresi (*Regression*)

Merupakan suatu fungsi yang digunakan untuk memodelkan data untuk meminimalkan hasil kesalahan prediksi. Umumnya regresi dilakukan dengan data yang bersifat *time series*.

#### 4. *Association Rule*

Merupakan pemodelan kebergantungan. Fungsi asosiasi ini biasanya kita kenal dengan istilah “market basket analysis” yang merupakan fungsi untuk menemukan relasi atau korelasi antara himpunan item-item. Aturan asosiasi diartikan pada basket data yang digunakan untuk keperluan promosi, desain katalog untuk meningkatkan penjualan. Contoh penerapan asosiasi adalah Ketika *customer* membeli pampers maka ada kemungkinan membeli bir.

#### 5. *Anomaly Detection*

Mengidentifikasi data yang tidak umum. Bisa berupa outlier, perubahan deviasi/bias yang penting dan perlu investigasi lebih lanjut.

#### 6. *Summarization*

Menyediakan representasi data yang lebih sederhana meliputi pelaporan, visualisasi data yang dipergunakan untuk menunjang informasi dan penguatan keputusan.

### **2.4 Teknik Klasifikasi**

Menurut (Hidayanti, Kurniawan, and Afriyudi 2020) Klasifikasi merupakan suatu pekerjaan menilai objek data untuk memasukkannya ke dalam 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 Teknik Data Mining

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-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 6 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 tekniknya, 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.

## 2.6 Pemrograman Aplikasi

### 2.6.1. Web

Menurut , 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 Abdullah (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 yang terkoneksi dengan internet.

### **2.6.2. 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 ke dalam berkas format ASCII agar dapat menghasilkan tampilan wujud yang terintegrasi. Dengan kata lain, berkas yang dibuat dalam perangkat lunak pengolah kata dan disimpan ke dalam 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 dan Berners-lee Robert ketika mereka bekerja di CERN pada tahun 1989 (CERN adalah lembaga penelitian fisika energi tinggi

diJenewa)” (Supriyanti, Kusriani, and Amborowati 2016).

### **2.6.3. 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. (Hidayanti, Kurniawan, and Afriyudi 2020)

### **2.6.4. 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 dari sini dapat dilihat bahwa javascript merupakan bahasa pemrograman yang berjalan di sisi client. Karena proses dari javascript tergantung proses dari browser yang digunakan oleh user. JavaScript pada penelitian ini digunakan untuk perancangan kuis. Contoh penggunaan JavaScript adalah sebagai berikut:

```
<html>  
  
<body>  
  
<script language="javascript">  
document.write("Ini dihasilkan oleh  
javascript");  
</script>  
  
</body>  
  
</html>
```

#### **2.6.5. 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](http://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”(Kadafi 2018). Bootsap 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 bootsrap yang bisa didapatkan melalui website resmi bootstrap . Cara meggunakannya adalah dengan mencopy paste kodingan dari template tersebut, kemudian edit kode- kode tersebut sesuai dengan kebutuhan. Sebagai batasan, ada beberapa kode yang tidak boleh diubah agar tetap selalu dapat meggunakannya, sepertimisalnya pemanggilan link yang terdapat pada tag head “ ”.

#### **2.6.6. 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 me-request-nya. Sesuai dengan fungsinya phpberjalan 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 (Suwirmayanti 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.

#### **2.6.7. Database**

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 kuncising disebut dengan Primary Key.

#### **2.6.8. MySQL**

MySql adalah suatu sistem manajemen database (Relasional Database Management System) bersifat terbuka (Open Source). Maksudnya terbuka adalah mysql boleh di-download oleh siapa saja baik versi kode program aslinya maupun 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. 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-macam aplikasi Komputer yang ditulis dengan berbagai bahasa pemrograman dapat mengakses basis data MySQL.

### **2.7 Algoritma Sistem**

#### **2.7.1 Algoritma Naïve Bayes**

Algoritma *naive bayes* merupakan pengklasifikasian dengan metode probabilitas dan statistik yang ditemukan oleh ilmuwan Inggris *Thomas Bayes*, yaitu memprediksi peluang di masa depan berdasarkan pengalaman di masa sebelumnya sehingga dikenal

sebagai *Teorema Bayes*.

Teorema tersebut dikombinasikan dengan Naive dimana diasumsikan kondisi antar atribut saling bebas. Klasifikasi Naive Bayes diasumsikan bahwa ada atau tidak ciri tertentu dari sebuah kelas tidak ada hubungannya dengan ciri dari kelas lainnya

$$P(H | X) = \frac{P(X | H)P(H)}{P(X)}$$

Dimana :

X = data dengan class yang belum diketahui

H = hipotesis data X, merupakan suatu class spesifik

$P(H|X)$  = probabilitas hipotesis H berdasarkan kondisi X  
(posteriori probability)

$P(H)$  = probabilitas hipotesis H (prior probability)

$P(X|H)$  = probabilitas X berdasarkan kondisi pada hipotesis

$P(X)$  = probabilitas dari X

Dalam penerapan metode *Naive Bayes*, diperlukan beberapa petunjuk dalam proses klasifikasi untuk menentukan kelas yang sesuai dengan data sampel yang dianalisa. Metode *Naive Bayes* dapat dinyatakan dalam persamaan berikut [7]:

$$P(C|XF_1 \dots F_n) = \frac{P(C)P(F_1 \dots F_n|C)}{P(F_1 \dots F_n)} \quad (2)$$

Dari persamaan diatas, variabel C menjelaskan kelas, sedangkan variabel  $F_1 \dots F_n$  menggambarkan kondisi yang

dibutuhkan untuk proses klasifikasi data. Rumus pada persamaan tersebut menjelaskan bahwa probabilitas masuknya sampel karakteristik tertentu ke dalam kelas  $C$  (*Posterior*) merupakan probabilitas munculnya kelas  $C$ , dikalikan dengan probabilitas munculnya karakteristik sampel pada kelas  $C$  (*likelihood*), dibagi dengan probabilitas munculnya karakteristik sampel secara keseluruhan. Sehingga rumus pada persamaan 2 diatas, dapat dituliskan dalam bentuk yang lebih sederhana sebagai berikut [7]:

$$Posterior = (prior \times likelihood) / evidence \quad (3)$$

Nilai dari *Evidence* tetap pada setiap kelas untuk satu sampel data tertentu. Untuk menentukan ke kelas apa sampel tertentu akan diklasifikasikan, nilai pada *posterior* akan dibandingkan dengan nilai *posterior* pada kelas yang lainnya. Penjabaran dari rumus *Bayes* menggunakan aturan perkalian sebagai berikut ( $C|F1, \dots$ ):

$$\begin{aligned} (C|F1, \dots, &= P(C)P(F1, \dots, Fn|C) \\ &= (C) (F1|C) (F2, \dots, |C, F1) \\ &= (C) (F1|C) (F2|C, F1) P(F3, \dots, Fn|C, F1, F2) \\ &= (C) P(F1|C)P(F2|C, F1) P(F3|C, F1, F2) P(F4, \dots, Fn|C, F1, F2, F3) \\ &= P(C) P(F1|C)P(F2|C, F1) P(F3|C, F1, F2) \dots \\ &P(Fn|C, F1, F2, F3, \dots, Fn-1)(5) \end{aligned}$$

Dari rumus yang dikemukakan pada persamaan 5 diatas, dapat dijelaskan bahwa faktor yang mempengaruhi nilai probabilitas semakin banyak dan beragam, dan tidak mudah dilakukan analisa satu persatu. Sehingga perhitungan probabilitas pada persamaan tersebut, sangat sulit dilakukan. Digunakan asumsi independen yang sangat tinggi (naif), masing-masing faktor ( $F1, F2 \dots Fn$ ) tidak saling terikat satu sama lain (*independen*). Berdasarkan asumsi tersebut, maka persamaan 5 dapat diberlakukan menjadi kesamaan berikut:

$$\begin{aligned}
 P(F_i|F_j) &= \frac{P(F_i \cap F_j)}{P(F_j)} = \frac{P(F_i) P(F_j)}{P(F_j)} & (6) \\
 &= P(F_i)
 \end{aligned}$$

Dengan

$i \neq j$ , dan

$$P(F_i|C, F_j) = P(F_i|C)$$

Persamaan 6 di atas adalah model teori *Naive Bayes* yang akan digunakan untuk proses klasifikasi dalam data mining. Sedangkan klasifikasi untuk data kontinyu dapat digunakan rumus *Densitas Gauss* [6]

$$P(X_i = x_i | Y = y_j) = \frac{1}{\sqrt{2\pi}\sigma_{ij}} e^{-\frac{(x_i - \mu_{ij})^2}{2\sigma_{ij}^2}} \quad (7)$$

Keterangan:

$P$  : Peluang

$X_i$ : Atribut ke  $i$

$x_i$ : Nilai

atribut ke  $i$

$Y$  : Kelas yang

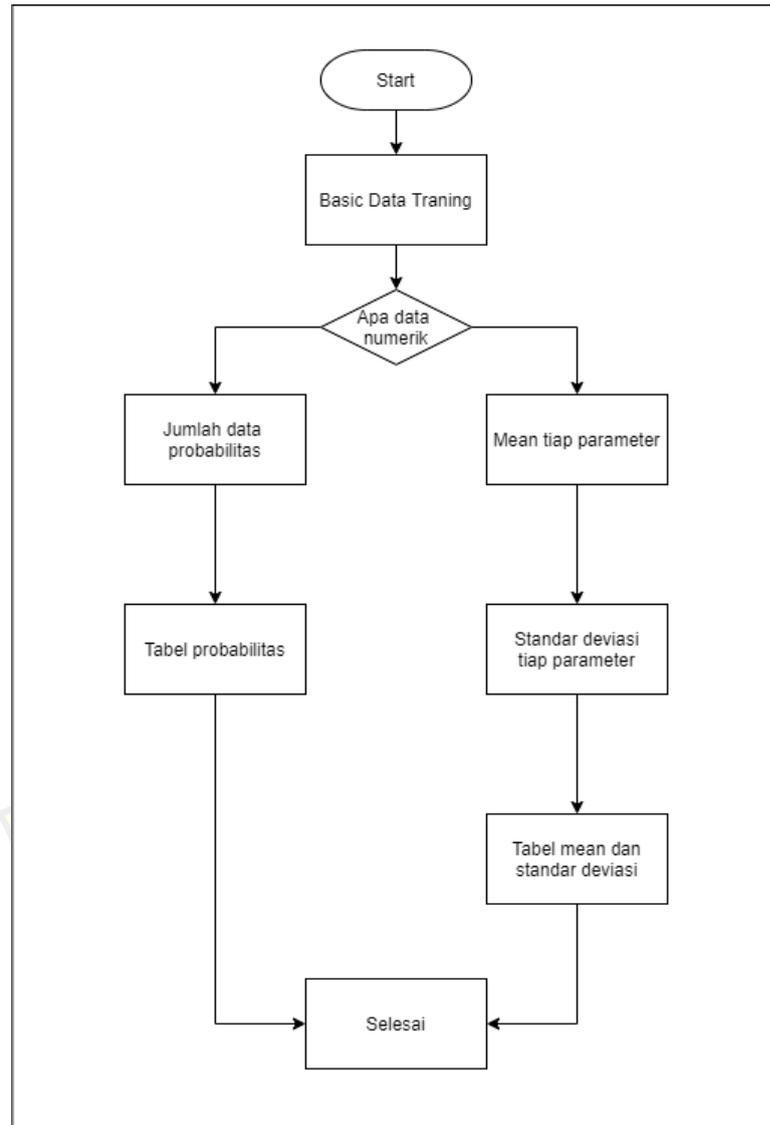
dicari

$Y_i$ : Sub kelas  $Y$  yang dicari

$M$ : *mean*, menyatakan rata – rata dari seluruh atribut

$\sigma$  : Standar deviasi , menyatakan varian dari seluruh atribut.

Dalam proses klasifikasi data, alur penerapan metode *Naive Bayes* dapat dijelaskan melalui bagan berikut.



Gambar 2. 3 Alur Algoritma Naive Bayes

Dari Gambar 1 dapat dijelaskan bahwa alur kerja metode *Naive Bayes* adalah sebagai berikut:

1. Mempersiapkan *data training*
2. Menghitung Jumlah dan probabilitas, jika data bertipe numerik maka:
  - a. Mencari nilai *mean* dan standar *deviasi* dari parameter yang bertipe data *numeric*, dengan persamaan yang digunakan untuk menghitung nilai *mean* adalah sebagai berikut:

$$\mu = \frac{\sum_{i=1}^n x_i}{n} \quad (8)$$

Atau

$$\mu = \frac{(x_1 + x_2 + x_3 \dots + x_n)}{n} \quad (9)$$

dengan:

$\mu$  : nilai rata-rata (*mean*)

$x_i$  : nilai *sample* ke - $i$

$n$  : jumlah sampel yang digunakan

persamaan yang dapat digunakan untuk menghitung nilai standar deviasi adalah sebagai berikut:

$$\sigma = \frac{\sqrt{\sum_{i=1}^n (x_i - \mu)^2}}{n - 1}$$

(10)

Dengan keterangan:

$\sigma$ : standar deviasi

$x_i$  : nilai  $x$  ke - $i$

$\mu$  : nilai rata-rata

$n$  : jumlah sampel yang digunakan

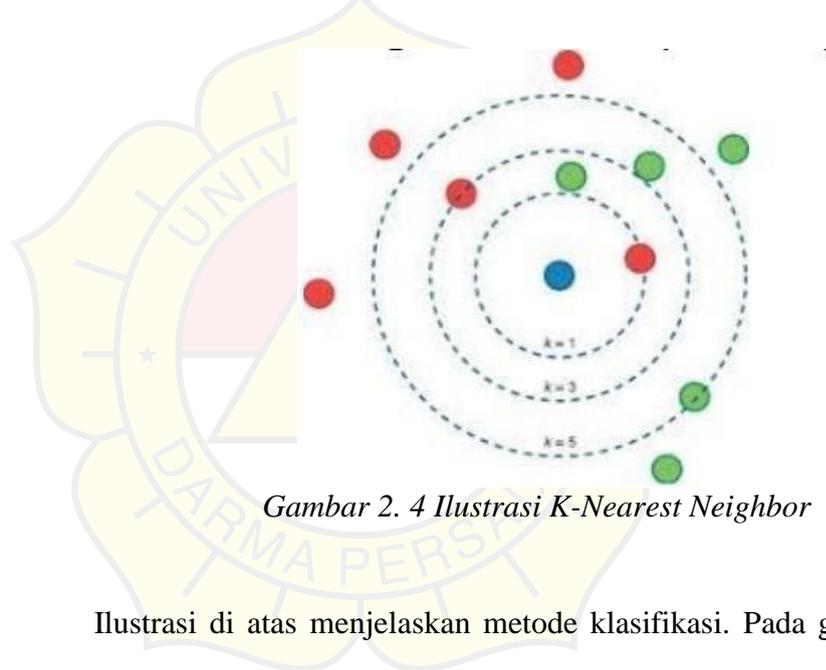
b. Menghasilkan nilai rata-rata, standar deviasi dan probabilitas.

3. jika data yang diolah tidak bertipe numerik maka langkah berikutnya adalah mencari nilai *probabilistic*, jumlah data sesuai dari kategori yang sama dihitung kemudian dibagi dengan jumlah data pada kategori tersebut.

4. Menghasilkan solusi.

### 2.7.2 Algoritma K-NN

Metode *K-Nearest Neighbor* (K-NN) merupakan salah satu dalam top 10 metode data mining yang paling banyak digunakan. Metode ini melakukan klasifikasi berdasarkan kemiripan suatu data dengan data yang lain. Prinsip sederhana metode ini adalah “Jika suatu hewan berjalan seperti bebek, bersuarakwek-kwek seperti bebek, dan penampilannya seperti bebek, maka hewan itu mungkin bebek (Anshori, Putri, and Tibyani 2018).



Gambar 2. 4 Ilustrasi *K-Nearest Neighbor*

Ilustrasi di atas menjelaskan metode klasifikasi. Pada gambar ini, titik biru akan dijadikan objek diprediksi kelasnya. Untuk  $k = 1$  maka kemungkinan objek masuk ke kelas merah. Selanjutnya  $k = 3$ , objek diprediksi masuk kelas merah, dengan perhitungan 2-1 lebih banyak atas kelas hijau. Untuk  $k = 5$ , maka akan diprediksi masuk kelas hijau dengan perhitungan 3-2

lebih banyak dari kelas merah. *K-Nearest Neighbor* digunakan untuk mengklasifikasi data yang tidak dilabeli. Karakteristik data

didapatkan dari training set dan test set(Zhang, 2016). Langkah-langkah klasifikasi data menggunakan K-Nearest Neighbor adalah sebagai berikut :

1. Tentukan nilai K
2. Hitung jarak antara data baru ke setiap label data
3. Tentukan k labeled data yang mempunyai jarak yang paling minimal
4. Klasifikasikan data baru ke dalam label data yang mayoritas K-NN dipilih berdasarkan metrik jarak.

Pada penelitian ini, data teks yang telah didapatkan akan diubah menjadi vektor. Dalam perhitungan jarak antar vektor, metode yang sering digunakan adalah cosine similarity. Metode ini mudah untuk dijelaskan pada data yang bersifat sparse seperti text data. Persamaan dari cosine similarity adalah:

$$\text{Scosine}(x, y) = \frac{x \cdot y}{\|x\| \|y\|}$$

Di mana  $\|x\| = \sqrt{\sum_{i=1}^l x_i^2}$  dan  $\|y\| = \sqrt{\sum_{i=1}^l y_i^2}$  adalah panjang dari vektor .

## 2.8 Pemodelan UML

“*Unified Modeling Language* (UML) bukanlah suatu proses melainkan bahasa pemodelan secara grafis untuk menspesifikasikan, memvisualisasikan, membangun, dan mendokumentasikan seluruh artefak 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 yangdigunakan untuk memvisualisasi, menspesifikasikan suatu sistem perangkat lunak.

### **2.8.1. 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
<p><i>Use Case</i></p> 	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 <u>use case</u>
<p><i>Actor</i></p> 	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
<p>Asosiasi</p> 	Komunikasi antar aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor
<p><i>Extensi</i></p> 	Relasi <i>use case</i> tambahkan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan
<p>Generalisasi</p> 	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
<p>Include</p> 	Relasi <i>use case</i> tambahkan 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

### 2.8.2. 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” (Heriyanto 2018). Simbol-simbol yang digunakan dalam activity diagram sebagai berikut :

Tabel 2. 2 Simbol Activity Diagram (Heriyanto 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

### 2.8.3. 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” (Heriyanto 2018).

Tabel 2. 3 Simbol Sequence Diagram (Heriyanto 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 lifeline mengindikasikan sebuah objek yang akan melakukan sebuah aksi	
<i>Message</i>	Digambarkan dengan anak panah horizontal antara activation. Message 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	

#### 2.8.4. Deployment Diagram

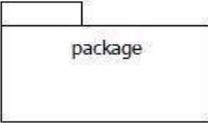
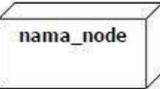
*Deployment Diagram* menunjukkan konfigurasi komponen dalam proses eksekusi aplikasi”. *Deployment Diagram* dapat digunakan untuk menggambarkan hal-hal sebagai berikut:

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 (Heriyanto 2018) *Deployment Diagram* digunakan untuk menggambarkan detail bagaimana komponen disusun di infrastruktur sistem.

Dapat disimpulkan bahwa *Deployment Diagram* merupakan diagram yang digunakan untuk menggambarkan konfigurasi pada sistem.

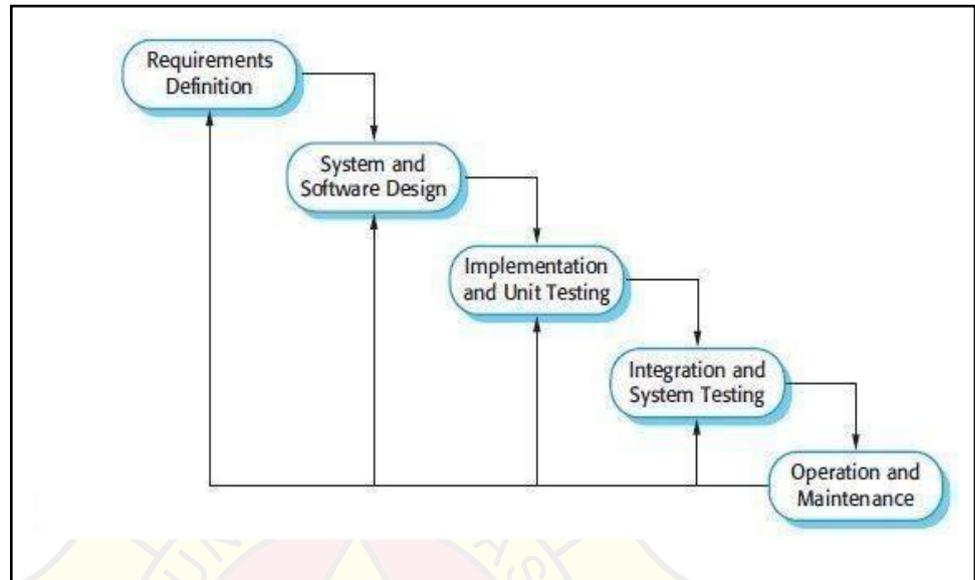
Tabel 2. 4 Simbol *Deployment Diagram* (Heriyanto 2018)

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>

## 2.9 Metodologi 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), design 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. Shalahudin, 2016)

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

### 2.9.1. 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 analis 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.

### **2.9.2. Design System (Design Sistem)**

Proses design akan menterjemahkan syarat kebutuhan 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.

### **2.9.3. Coding & Testing (Penulisan Sinkode Program/Implementation)**

Coding merupakan penerjemahan design dalam bahasa yang bisa dikenali oleh komputer. Dilakukan oleh programmer yang akan menterjemahkan 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 terhadap system tersebut dan kemudian bisa diperbaiki.

### **2.9.4. 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 digunakan oleh Pemeliharaan (Operation & Maintenance) Perangkat lunak yang susah 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.

