



BAB II

TINJAUAN PUSTAKA

2.1 Teori Judul

2.1.1 Pengertian Aplikasi

Pengertian aplikasi menurut Jogiyanto (dalam Suhartini, 2017:74) adalah penerapan, menyimpan sesuatu hal, data, permasalahan, pekerjaan ke dalam suatu sarana atau media yang dapat digunakan untuk diterapkan menjadi sebuah bentuk yang baru.

Sedangkan menurut Syifani dan Dores (2018:23) menyatakan bahwa aplikasi berasal dari kata *application* yang artinya penerapan, lamaran, dan penggunaan. Secara istilah aplikasi adalah program siap pakai yang dibuat untuk melaksanakan suatu fungsi bagi pengguna aplikasi dan dapat digunakan untuk sasaran yang dituju.

Selanjutnya menurut Kamus Besar Bahasa Indonesia, “Aplikasi adalah penerapan dari rancang sistem untuk mengolah data yang menggunakan aturan atau ketentuan bahasa pemrograman tertentu. Aplikasi adalah suatu program computer yang dibuat untuk mengerjakan dan melaksanakan tugas khusus dari *user* (pengguna)”.

Dari ketiga penjelasan di atas dapat disimpulkan bahwa aplikasi merupakan penerapan dan penggunaan dari sebuah rancangan sistem yang digunakan untuk mengolah data dan melaksanakan fungsi atau tugas tertentu yang diberikan oleh *user* atau pengguna aplikasi.

2.1.2 Pengertian Klasifikasi

Menurut Novandya dan Oktria (2017:99) mengemukakan bahwa klasifikasi adalah suatu proses menemukan sekumpulan pola atau fungsi yang mendeskripsikan serta memisahkan kelas data yang satu dengan yang lainnya untuk menyatakan objek tersebut masuk pada kategori tertentu yang sudah ditetapkan.



Sedangkan menurut Kamus Besar Bahasa Indonesia versi *Online* menyatakan bahwa klasifikasi merupakan penyusunan bersistem dalam kelompok atau golongan menurut kaidah atau standar yang ditetapkan.

Dari penjelasan terkait klasifikasi di atas, dapat disimpulkan bahwa definisi dari klasifikasi adalah sebuah proses untuk menemukan suatu pola atau fungsi kemudian mengumpulkan pola atau kelas data tersebut berdasarkan kategori secara berkelompok menurut standar yang telah ditetapkan.

Dalam klasifikasi terdapat target variable kategori. Metode-metode / model-model yang telah dikembangkan oleh periset untuk menyelesaikan kasus klasifikasi adalah sebagai berikut.

- a. pohon keputusan
- b. pengklasifikasi bayes / *naïve bayes*
- c. jaringan saraf tiruan
- d. analisis statistic
- e. algoritma genetic
- f. *rough sets*
- g. pengklasifikasi *k-nearest neighbour*
- h. metode berbasis aturan
- i. *memory based reasoning*
- j. *support vector machine*

2.1.3 Pengertian Kartu Kredit

Kartu kredit merupakan sebuah alat pembayaran yang berfungsi sebagai pengganti uang tunai, dimana alat ini dapat dipergunakan oleh konsumen untuk ditukarkan dengan berbagai barang/jasa yang dibelinya di tempat-tempat yang bisa menerima pembayaran dengan menggunakan kartu kredit (*merchant*).

Hal ini juga sesuai dengan pengertian kartu kredit yang tercantum di dalam pasal 1 angka 4 Peraturan Bank Indonesia Nomor 7/52/PBI/2005 yang kemudian diubah dalam Peraturan Bank Indonesia Nomor 10/8/PBI/2008 tentang Penyelenggara Kegiatan Alat Pembayaran dengan Menggunakan Kartu yaitu



“kartu kredit adalah alat pembayaran dengan menggunakan kartu yang dapat digunakan untuk melakukan pembayaran atas kewajiban yang timbul dari suatu kegiatan ekonomi, termasuk transaksi pembelian atau untuk melakukan penarikan tunai, dimana kewajiban pembayaran pemegang kartu dipenuhi terlebih dahulu oleh *acquirer* atau penerbit, dan pemegang kartu berkewajiban melakukan pelunasan kewajiban pembayaran tersebut pada waktu yang disepakati, baik secara sekaligus (*charge card*) atau secara angsuran”.

Terdapat berbagai jenis kartu kredit khususnya pada bank mandiri. Setiap jenis kartu kredit yang diterima nasabah memiliki *limit* yang bervariasi. Nilai *limit* kartu kredit tersebut ditentukan berdasarkan besaran penghasilan bulanan nasabah. Semakin tinggi penghasilannya maka limit yang diberikan oleh pihak bank juga semakin tinggi. Adapun jenis-jenis kartu kredit tersebut pada bank Mandiri antara lain Mandiri Signature, Mandiri Platinum, Mandiri Gold, Mandiri Everyday Card, Mandiri Hypermart Card, Mandiri Skyz Card, Mandiri Pertamina Card, Mandiri Golf Signature, Mandiri Golf Platinum, Mandiri Golf Gold, dan Mandiri World Elite.

2.1.4 Metode *Naïve Bayes*

Patil dan Sherekar (dalam Cahya, 2019), “*Naïve Bayes* merupakan sebuah pengklasifikasian *probabilistic* sederhana yang menghitung sekumpulan probabilitas dengan menjumlahkan frekuensi dan kombinasi nilai dari dataset yang diberikan. Algoritma menggunakan teorema Bayes dan mengasumsikan semua atribut independen atau tidak saling ketergantungan yang diberikan oleh nilai pada variable kelas.”

Sedangkan menurut Rosandy (dalam Hakam, dkk., 2020:21) menyatakan bahwa *Bayesian Classification* adalah pengklasifikasian statistic yang digunakan untuk memprediksi probabilitas keanggotaan suatu kelas. *Bayes classification* didasarkan pada teorema Bayes yang memiliki kemampuan klasifikasi serupa dengan *decision tree* dan *neural network*. *Bayesian classification* terbukti



memiliki akurasi dan kecepatan yang tinggi saat diaplikasikan ke dalam *database* dengan data yang besar.

Adapun teorema Bayes dirumuskan sebagai berikut (Cahya, 2019):

$$P(C|X) = \frac{P(X|C)}{P(X)} \cdot P(C)$$

Keterangan :

- X : Data dengan *class* yang belum diketahui
- C : Hipotesis data merupakan suatu *class* spesifik
- $P(C|X)$: Probabilitas hipotesis C berdasar kondisi X (*posteriori probabilitas*)
- $P(H)$: Probabilitas hipotesis C (*prior probabilitas*)
- $P(X|C)$: Probabilitas X berdasarkan kondisi pada hipotesis C
- $P(X)$: Probabilitas X

Teorema Bayes sering pula dikembangkan mengingat berlakunya hukum probabilitas total, sehingga menjadi seperti berikut.

$$P(C|X) = \frac{P(X|C)}{\sum_{i=1}^n P(C_i|X)} \cdot P(C)$$

Keterangan :

- i : 1,2,3, ... , n jumlah data hipotesis (*prior probabilitas*)
- Dimana : $C_1 \cup C_2 \cup C_3 \dots \cup C_n = S$
- S : Probabilitas toatal C

Untuk menjelaskan Teorema *Naïve Bayes*, perlu diketahui bahwa proses klasifikasi memerlukan sejumlah petunjuk untuk menentukan kelas apa yang cocok bagi sampel yang dianalisis tersebut. Karena itu, Teorema Bayes di atas disesuaikan sebagai berikut.



$$P(C|X_1, \dots, X_n) = \frac{P(X_1, \dots, X_n|C)}{P(X_1, \dots, X_n)} \cdot P(C)$$

Dimana variable C merepresentasikan kelas, sementara variable $X_1 \dots X_n$ merepresentasikan karakteristik petunjuk yang dibutuhkan untuk melakukan klasifikasi. Maka rumus tersebut menjelaskan bahwa peluang masuknya sampel karakteristik tertentu dalam kelas c (*posterior*) adalah peluang munculnya kelas c (sebelum masuknya sampel tersebut, seringkali disebut *prior*), dikali dengan peluang kemunculan karakteristik-karakteristik sampel pada kelas c (disebut juga *likelihood*), dibagi dengan peluang kemunculan karakteristik-karakteristik sampel secara global (disebut juga *evidence*). Karena itu, rumus di atas dapat pula ditulis secara sederhana sebagai berikut.

$$\text{Posterior} = \frac{\text{prior} \times \text{likelihood}}{\text{evidence}}$$

Nilai *evidence* selalu tetap untuk setiap kelas pada satu sampel. Nilai dari *posterior* tersebut nantinya akan dibandingkan dengan nilai-nilai *posterior* kelas lainnya untuk menentukan ke kelas apa suatu sampel akan diklasifikasikan. Penjabaran lebih lanjut rumus Bayes tersebut dilakukan dengan menjabarkan (C, X_1, \dots, X_n) menggunakan aturan perkalian sebagai berikut.

$$\begin{aligned} P(C|X_1, \dots, X_n) &= P(C) \cdot P(X_1, \dots, X_n|C) \\ &= P(C) \cdot P(X_1|C) \cdot P(X_2, \dots, X_n|C, X_1) \\ &= P(C) \cdot P(X_1|C) \cdot P(X_2|C, X_1) \cdot P(X_3, \dots, X_n|C, X_1, X_2) \\ &= P(C) \cdot P(X_1|C) \cdot P(X_2|C, X_1) \cdot P(X_3|C, X_1, X_2) \dots P(X_n|C, X_1, X_2, X_3, \dots, X_{n-1}) \end{aligned}$$

Dapat dilihat bahwa hasil penjabaran tersebut menyebabkan semakin banyak dan semakin kompleksnya faktor-faktor syarat yang mempengaruhi nilai probabilitas, yang hampir mustahil untuk dianalisa satu persatu. Akibatnya,



perhitungan tersebut menjadi sulit untuk dilakukan. Disinilah digunakan asumsi independensi yang sangat tinggi (*naif*), bahwa masing-masing petunjuk (X_1, \dots, X_n) saling bebas (*independen*) satu sama lain. Dengan asumsi tersebut, maka berlaku suatu keamanan sebagai berikut.

$$P(X_i|X_j) = \frac{P(F_i \cap F_j)}{P(F_j)} = \frac{P(F_i) \cdot P(F_j)}{P(F_j)} = P(F_i)$$

Untuk $i \neq j$, sehingga

$$P(X_i|C, X_j) = P(X_i|C)$$

Atau dapat tuliskan dalam notasi

$$P(C|X_1, X_2, X_3, \dots, X_n) = P(C) \prod_{i=1}^n P(X_i|C)$$

Yang dapat dijabarkan sebagai berikut

$$P(C|X) = P(X_1|C) \cdot P(X_2|C) \cdot P(X_3|C) \dots P(X_n|C) \cdot P(C)$$

Persamaan di atas merupakan model dari teorema *Naïve Bayes* yang selanjutnya akan digunakan dalam proses klasifikasi. Untuk klasifikasi dengan data kontinyu digunakan rumus Densitas Gauss, sebagai berikut.

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

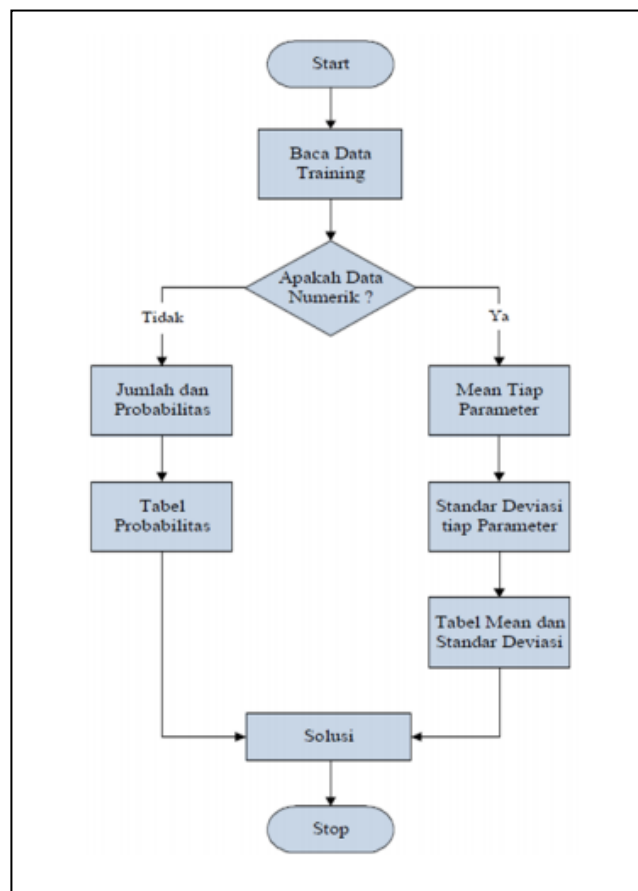
Keterangan:

- P : Peluang
- X_i : Atribut ke-i
- x_i : Nilai atribut ke-i
- Y : Kelas yang dicari



- y_i : Sub-kelas yang dicari
 μ : *Mean*, menyatakan rata-rata dari seluruh atribut
 σ : Deviasi standar, menyatakan varian dari seluruh atribut

Adapun alur dari metode *Naïve Bayes* dapat dilihat pada gambar berikut (Cahya, 2019):



Gambar 2.1 Alur Metode *Naïve Bayes*

Adapun keterangan dari gambar di atas yaitu sebagai berikut.

Membaca data *training*

1. Menghitung jumlah dan probabilitas, namun jika data numeric maka :



- a. Menghitung nilai *mean* dan standar deviasi dari masing-masing parameter yang merupakan numeric. Adapun persamaan untuk mencari nilai rata-rata hitung (*mean*) adalah seperti dalam persamaan berikut ini:

$$\mu = \frac{\sum_{i=1}^n x_i}{n}$$

Atau

$$\mu = \frac{x_1 + x_2 + x_3 + \dots + x_n}{n}$$

Keterangan :

- μ : Nilai rata-rata hitung (*mean*)
 x_i : Nilai x ke- i
 n : Jumlah sampel

Sedangkan persamaan untuk menghitung nilai simpangan baku (standar deviasi) dirumuskan sebagai berikut.

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

Keterangan :

- σ : Standar deviasi
 x_i : Nilai x ke-i
 μ : Nilai rata-rata hitung (*mean*)
 n : Jumlah sampel

- b. Menghitung nilai probabilitik dengan cara menghitung jumlah data yang sesuai dari kategori yang sama dibagi dengan jumlah data pada kategori tersebut.
2. Mendapatkan nilai dalam tabel *mean*, standar deviasi dan probabilitas.
 3. Menghasilkan solusi.



2.2 Teori Program

2.2.1 MySQL (*My Structured Query Language*)

Menurut Yudhanto dan Prasetyo (2018:13) mengatakan bahwa MySQL adalah *database* yang paling favorit dimana program ini berjalan sebagai *server* yang menyediakan *multi-user*, mengakses ke sejumlah *database* baik *multithread* maupun *multi-user*.

Sedangkan menurut Mundzir (2018:217) dalam buku *Buku Sakti Pemrograman Web Seri PHP*, “MySQL adalah manajemen *database* SQL yang sifatnya *open source* (terbuka) dan paling banyak digunakan saat ini. Sistem *database* MySQL mampu mendukung beberapa fitur seperti *multithreaded*, *multi-user*, dan *SQL database management system* (DBMS).”

MySQL adalah sebuah *database management system* (DBMS) populer yang memiliki fungsi sebagai *relational database management system* (RDBMS). Selain itu MySQL *software* merupakan suatu aplikasi yang sifatnya *open source* serta *server* basis data MySQL memiliki kinerja sangat cepat, *reliable*, dan mudah untuk digunakan serta bekerja dengan arsitektur *client server* atau *embedded systems*.

Adapun kelebihan yang dimiliki oleh MySQL adalah sebagai berikut (Yudhanto dan Prasetyo, 2018:13):

- a. *Free*, stabil, dan tangguh.
- b. Fleksibel dengan berbagai pemrograman.
- c. *Security* atau keamanan yang baik.
- d. Kemudahan dalam *management database*.
- e. Mendukung transaksi.
- f. Perkembangan *software* cukup cepat.

2.2.2 PHP (*Hypertext Preprocessor*)

Menurut MADCOMS (2016) “PHP (*Hypertext Preprocessor*) adalah bahasa *script* yang dapat ditanamkan atau disisipkan ke dalam HTML. PHP banyak dipakai untuk membuat program situs web dinamis”.



Sedangkan menurut Yudhanto dan Prasetyo (2018:7) mengemukakan bahwa PHP adalah bahasa pemrograman *script server side* yang sengaja dirancang lebih cenderung untuk membuat dan mengembangkan web. Bahasa pemrograman ini dirancang untuk para pengembang web agar dapat menciptakan suatu halaman web yang bersifat dinamis.

Selanjutnya menurut Jubilee (2017:1) mengatakan bahwa PHP merupakan bahasa pemrograman yang digunakan untuk membuat *website* dinamis dan interaktif. Dinamis artinya, *website* tersebut bisa berubah-ubah tampilan dan kontennya sesuai kondisi tertentu. Sedangkan interaktif artinya, PHP dapat memberi *feedback* bagi *user* (misalnya menampilkan hasil pencarian produk).

Dari ketiga pendapat di atas dapat disimpulkan bahwa PHP merupakan bahasa pemrograman *server side* yang digunakan untuk membuat *website* yang sifatnya dinamis.

2.2.3 Laragon

Laragon adalah *universal development environment* yang portabel, terisolasi, cepat, ringan, dan mudah digunakan untuk bahasa pemrograman PHP, Node.js, Python, Java, Go, dan Ruby. Aplikasi ini sama halnya seperti XAMPP, namun didesain untuk kebutuhan *developer PHP* yang menggunakan *framework* Laravel. Laravel lebih berfokus pada kinerja, dirancang dengan kesederhanaan, fleksibilitas, dan kebebasan. Sehingga sangat bagus untuk membangun dan mengelola *modern web applications*.

2.2.4 Sublime Text

Sublime Text Editor adalah *text editor* yang digunakan untuk mempermudah pekerjaan *programmer*. selain itu juga *sublime* digunakan untuk banyak sekali bahasa pemrograman dan bahasa *markup* serta mendukung penambahan *plugin*. *Sublime* dibangun dengan menggunakan *python*.

Sublime Text memiliki banyak kelebihan diantaranya:



1. *Multiple Selection*, mempunyai fungsi untuk melakukan perubahan pada sebuah kode dalam waktu yang sama dan dalam baris yang berbeda.
2. *Command Pallette*, berguna untuk mengakses file *shortcut* dengan mudah dengan menekan CTRL+SHIFT+P.
3. *Distraction free mode*, fitur ini sangat dibutuhkan oleh pengguna yang sedang fokus dalam pekerjaan, yaitu dapat merubah tampilan layar menjadi penuh dengan menekan SHIFT+F11.
4. *Find in project*, fungsi ini memberikan kemudahan dalam mencari dan memilih file dalam *project*, dengan menekan SHIFT+P.
5. *Multi platform*, *Sublime Text* sudah tersedia dalam berbagai *platform* sistem operasi seperti Windows, Linux, Mac os.

2.3 Teori Khusus

2.3.1 *Unified Modeling Language (UML)*

Menurut Avison dan Fitzgerald dalam buku *Metodologi Pengembangan Sistem Informasi* (Sarosa, 2017:143) mengatakan bahwa UML adalah notasi pemodelan yang banyak digunakan dalam metodologi berorientasi objek. Tujuan dari UML yaitu sebagai berikut.

- a. Bagi pengguna, tersedia bahasa pemodelan visual yang siap digunakan dan ekspresif sehingga dapat digunakan bersama dalam pemodelan suatu sistem informasi.
- b. Memberikan mekanisme pemodelan yang dapat diperluas dan dikhususkan untuk memperluas konsep dasarnya.
- c. Tidak tergantung pada bahasa pemrograman tertentu maupun proses pengembangan (metodologi) tertentu.
- d. Memberikan dasar pemahaman secara formal terhadap bahasa pemodelan.
- e. Mendorong pertumbuhan alat-alat bantu berorientasi objek.
- f. Mendukung konsep pengembangan sistem pada tingkatan tinggi (tidak terlalu teknis).
- g. Mengintegrasikan praktik dan metodologi terbaik.

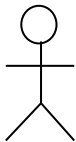
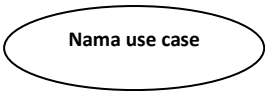

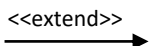


Terdapat beberapa diagram UML yaitu sebagai berikut (Rosa dan Shalahuddin (2018:155)).

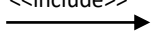

a. *Use Case Diagram*

Menurut Rosa dan Shalahuddin (2018:155) mengatakan bahwa *use case diagram* digunakan untuk memodelkan kelakuan (*behavior*) dari sistem informasi yang akan dibuat. *Use case* akan mendeskripsikan sebuah interaksi yang terjadi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Atau dapat dikatakan bahwa *use case* digunakan untuk mengetahui fungsi-fungsi yang terdapat di dalam sistem dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut.

Tabel 3.1 Simbol *Use Case Diagram*

| No | Simbol | Keterangan |
|----|--|---|
| 1. | Aktor  | Orang, proses, atau system lain yang berinteraksi dengan system informasi yang akan dibuat di luar system informasi yang akan dibuat itu sendiri. Actor biasanya dinyatakan menggunakan kata benda di awal fase nama actor. |
| 2. | <i>Use case</i>  | Fungsionalitas yang disediakan system sebagai unit-unit yang saling bertukar pesan antar unit atau actor, biasanya <i>use case</i> dinyatakan menggunakan kata kerja di awal frase nama <i>use case</i> |
| 3. | Asosiasi  | Komunikasi antar actor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan actor |
| 4. | <i>Extend</i>  | 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 itu. Biasanya <i>use case</i> tambahan memiliki nama depan yang sama dengan <i>use case</i> yang ditambahkan |

Lanjutan Tabel 3.1 Simbol *Use Case Diagram*


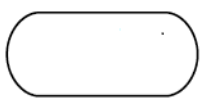

| No | Simbol | Keterangan |
|----|--|---|
| 5. | <i>Include</i> <<include>>  | Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsinya atau sebagai syarat dijalankan <i>use case</i> ini |
| 6. | Generalisasi  | Hubungan generalisasi dan spesialisasi (umum – khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya. Arah panah mengarah pada <i>use case</i> yang menjadi generalisasinya (umum) |

(Sumber : Rosa dan Shalahuddin, 2018:156)



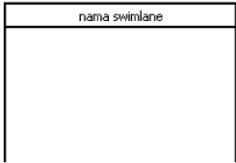
b. *Activity Diagram*

Activity Diagram menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah system atau proses bisnis atau menu yang ada pada perangkat lunak. Diagram aktivitas menggambarkan aktivitas system bukan apa yang dilakukan actor, jadi aktivitas yang dapat dilakukan oleh system (Rosa dan Shalahuddin, 2018:161).

Tabel 3.2 Simbol *Activity Diagram*

| No | Simbol | Keterangan |
|----|--|--|
| 1. | Status Awal  | Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal |
| 2. | Aktivitas  | Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja |
| 3. | Percabangan  | Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu |

Lanjutan Tabel 3.2 Simbol *Activity Diagram*


| No | Simbol | Keterangan |
|----|--|---|
| 4. | Penggabungan  | Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu |
| 5. | Status akhir  | Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir |
| 6. | <i>Swimlane</i>  | Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi |

(Sumber : Rosa dan Shalahuddin, 2018:162)


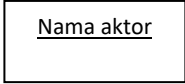

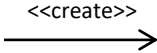
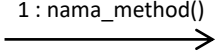
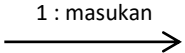
c. *Sequence Diagram*

Menurut Rosa dan Shalahuddin (2018:165) mengatakan bahwa *sequence diagram* menggambarkan kelakuan objek yang ada pada *use case* dengan mendeskripsikan waktu hidup objek dan pesan yang dikirim dan diterima antar objek tersebut. Maka dari itu dalam menggambarkan diagram sekuen, perlu diketahui terlebih dahulu objek-objek yang terlibat dalam sebuah *use case* beserta metode-metode yang dimiliki oleh setiap kelas yang diinstansiasi menjadi objek tersebut. Selain itu membuat diagram sekuen juga dibutuhkan untuk melihat skenario yang ada pada *use case*.

Tabel 3.3 Simbol *Sequence Diagram*

| No | Simbol | Deskripsi |
|----|--|-----------------------------------|
| 1. | Garis hidup / <i>lifeline</i>  | Menyatakan kehidupan suatu objek. |

Lanjutan Tabel 3.3 Simbol *Sequence Diagram*

| No | Simbol | Deskripsi |
|----|--|---|
| 2. | <p>Aktor</p>  <p>Atau</p>  | Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang, biasanya dinyatakan menggunakan kata benda di awal frase nama aktor. |
| 3. | <p>Waktu aktif</p>  | Menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan di dalamnya. |
| 4. | <p>Pesan tipe <i>create</i></p>  | Menyatakan suatu objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat. |
| 5. | <p>Pesan tipe <i>call</i></p>  | Menyatakan suatu objek memanggil operasi/metode yang ada pada objek lain atau dirinya sendiri, arah panah mengarah pada objek yang memiliki operasi/metode. |
| 6. | <p>Pesan tipe <i>send</i></p>  | Menyatakan bahwa suatu objek mengirimkan data/masukan/infor-masi ke objek lainnya, arah panah mengarah pada objek yang dikirim. |

Lanjutan Tabel 3.3 Simbol *Sequence Diagram*

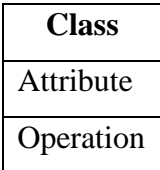

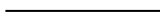
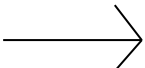
| No | Simbol | Deskripsi |
|----|---|---|
| 7. | Pesan tipe keluaran 1 : keluaran -----> | Menyatakan bahwa suatu objek yang telah menjalankan suatu operasi atau metode menghasilkan suatu kembalian ke objek tertentu, arah panah mengarah pada objek yang menerima kembalian. |

(Sumber : Rosa dan Shalahuddin, 2018:165)

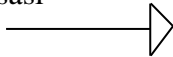
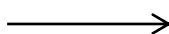
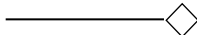
d. Class Diagram

Menurut Rosa dan Shalahuddin (2018:141) mengemukakan bahwa diagram kelas atau *class diagram* menggambarkan struktur sistem dari pendefinisian kelas-kelas yang akan digunakan untuk membangun sebuah sistem sistem. Setiap kelas memiliki atribut dan *method*. Atribut merupakan variable-variabel yang dimiliki oleh suatu kelas. Sedangkan operasi atau *method* merupakan fungsi-fungsi yang dimiliki oleh suatu kelas.

Tabel 3.4 Simbol *Class Diagram*

| No | Simbol | Deskripsi |
|----|---|---|
| 1. | Kelas  | Kelas pada struktur sistem. |
| 2. | Antarmuka / <i>interface</i>  | Sama dengan konsep <i>interface</i> dalam pemrograman berorientasi obyek. |
| 3. | Asosiasi  | Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i> . |
| 4. | Asosiasi berarah  | Asosiasi antarkelas dengan makna kelas yang satu digunakan oleh kelas lain, asosiasi biasanya juga disertain dengan <i>multiplicity</i> . |

Lanjutan Tabel 3.4 Simbol *Class Diagram*

| No | Simbol | Deskripsi |
|----|---|--|
| 5. | Generalisasi  | Asosiasi antar kelas dengan makna generalisasi spesialisasi (umum – khusus). |
| 6. | Kebergantungan / <i>dependency</i>  | Relasi antar kelas dengan makna kebergantungan antarkelas. |
| 7. | Agregasi  | Relasi antar kelas dengan makna semua bagian (<i>whole-part</i>). |

(Sumber : Rosa dan Shalahuddin, 2018:146)

2.4 Referensi Penelitian Sebelumnya

Menurut penelitian yang berjudul **Pengembangan Sistem Pendukung Keputusan Kredit Konsumtif dengan Metode *Naïve Bayes* (Studi Kasus : PT. Bank Pembangunan Daerah (BPD) Bali Cabang Tabanan)** tahun 2016 oleh I Gede Mahendra Darmawiguma, I Made Agus Wirawan, dan Luh Md Dwi Kusumayanti. Penilaian kelayakan kredit nasabah didasarkan pada 5 (lima) kriteria yaitu *character* (watak), *capacity* (kesanggupan melunasi), *capital* (modal), *condition* (kondisi), dan *collateral* (agunan). Selanjutnya untuk penentuan layak atau tidaknya nasabah menerima kredit ialah menggunakan metode *Naïve Bayes*. Metode ini memanfaatkan metode probabilitas dan statistic untuk memprediksi probabilitas di masa depan berdasarkan pengalaman di masa sebelumnya. Adapun proses yang terdapat pada sistem, yaitu memanipulasi data nasabah, pengajuan kredit, analisis kelayakan kredit, validasi kredit, pembuatan PK dan pencairan kredit. Hasil menunjukkan bahwa rata-rata akurasi sistem sebesar 99.33% dari 150 data sample yang digunakan. Artinya sistem mampu memberikan rekomendasi keputusan kelayakan kredit nasabah dengan baik dan mampu menghasilkan keputusan sesuai dengan kasus nyatanya. Sedangkan hasil uji respon pengguna setelah menggunakan sistem sebesar 91% dengan kategori sangat



baik.

Penelitian dengan judul **Metode Naïve Bayes untuk Penentuan Penerima Beasiswa Bidikmisi Universitas Mulawarman** tahun 2016 oleh Diasrina Dahri, Fahrul Agus, dan Dyna Marisa Khairina, penentuan penerima beasiswa menggunakan beberapa kriteria meliputi pekerjaan orang tua, penghasilan orang tua, jumlah tanggungan, daya listrik (*watt*), dan nilai Ujian Nasional (UN). Kelayakan calon penerima beasiswa Bidikmisi ditentukan dengan menerapkan metode *Naïve Bayes* dimana metode ini dipilih karena mampu mempelajari data kasus sebelumnya yang digunakan sebagai data uji. Penerapan metode *Naïve Bayes* dalam sistem ini diketahui kelas yang Tidak Layak sebanyak 114 data dan kelas Layak sebanyak 154 data dari 278 data yang digunakan. Setelah dilakukan pengujian, didapatkan bahwa penelitian ini menghasilkan sebuah aplikasi sistem pendukung keputusan dengan tingkat akurasi sebesar 85,56%.

Berdasarkan penelitian yang telah dilakukan oleh Haditsah Annur pada tahun 2018 dengan judul **Klasifikasi Masyarakat Miskin Menggunakan Metode Naïve Bayes**, penentuan klasifikasi berdasarkan data penduduk miskin yang diperoleh dari Kec. Tibawa dengan menggunakan teknik *data mining* metode *Naïve Bayes*. Atribut yang digunakan berupa umur, pendidikan, pekerjaan, penghasilan, tanggungan, dan status (kawin/belum kawin). Penelitian ini menghasilkan bahwa sistem klasifikasi masyarakat miskin di wilayah pemerintahan Kec. Tibawa Kab. Gorontalo dapat direkayasa dan berdasarkan hasil pengujian *confusion matrix* dengan teknik *split* penelitian diperoleh tingkat akurasi sebesar 73% atau termasuk dalam kategori *Good*. Sementara nilai *precision* sebesar 92% dan *recall* sebesar 86%.

Berdasarkan penelitian yang dilakukan oleh Andri Firmansyah, Feri Ramadhani, dan Edri Fauzan dengan judul **Sistem Pendukung Keputusan Penentuan Status Karyawan Menggunakan Metode Naïve Bayes** pada tahun 2020. Sistem ini mengimplementasikan metode *Naïve Bayes* yang



digunakan untuk mempercepat dan mempermudah pengambilan keputusan dalam menentukan status karyawan magang menjadi karyawan kontrak pada PT. Emsonic Indonesia. Dalam penilaian dan referensi penentuan keputusannya berdasarkan kriteria-kriteria karyawan sebagai data *training* (absensi, kompetensi, penilaian *leader*, pelanggaran/sanksi), dan menghitung data tersebut untuk mendapatkan nilai probabilitasnya sehingga dengan nilai probabilitas tersebut dapat memprediksi hasil keputusan apakah seorang karyawan layak atau tidak diubah statusnya. Hasil yang didapat dengan *Naïve Bayes* menunjukkan hasil data sample X (MCN016021) memiliki kelas yang berstatus “OK” karena nilai dari status OK lebih besar bila dibandingkan dengan nilai status gagal yaitu 0,03095. Sedangkan tingkat akurasi sistem yang dihasilkan sebesar 94% dengan kesalahan hanya 6 data dari 105 data sample yang digunakan.

Menurut penelitian dengan judul **Penerapan Metode *Naïve Bayes* untuk Klasifikasi Pelanggan** oleh Hakam Febtadianrano Putro, Retno Tri Vulandari, dan Wawan Laksito Yuly Saptomo pada tahun 2020, atribut yang digunakan dalam proses klasifikasi pelanggan yaitu jumlah pembelian, interval waktu, dan lokasi. Sistem ini digunakan untuk melihat dan mengklasifikasikan pelanggan mana yang berpotensi atau tidak berpotensi dari banyaknya data transaksi yang dilakukan. Jumlah data latih dan data uji yang digunakan untuk klasifikasi berjumlah 75 data latih yang kemudian diproses menggunakan metode *Naïve Bayes* dan 25 data uji pelanggan. Hasil dari sistem klasifikasi ini adalah 23 reaksi benar dan 2 reaksi salah. Berdasarkan hasil yang didapat dengan menggunakan metode pengujian *confusion matrix* didapatkan nilai akurasi mencapai 92%, nilai presisi mencapai 100%, dan nilai *recall* mencapai 91%.