

## **BAB II**

### **LANDASAN TEORI**

#### **2.1 Penelitian Terkait**

Dalam penyusunan skripsi ini, penulis sedikit banyak terinspirasi dan mereferensi dari penelitian-penelitian sebelumnya yang berkaitan dengan latar belakang masalah pada skripsi ini. Berikut ini penelitian terdahulu yang berhubungan dengan skripsi ini antara lain :

1. Impelentasi *Naive Bayes Classification Method To Predict Graduate Time Of IBI Darmajaya Scholar.*(Artaye, 2015)
2. Sistem Informasi Kerusakan Laptop Menggunakan Metode *Naive Bayes.*(Pramudia & Nugroho, 2017)
3. Penerapan Metode *Naive Bayes* Untuk Menganalisa Kerusakan *Automatic Transmission* Pada Mobil.(Luh & Pivin, n.d.)
4. Perbandingan Kinerja *Algoritme C4.5* dan *Naive Bayes* Mengklasifikasi Penyakit Diabetes.(Marcos & Utomo, 2015)

#### **2.2 Rancang Bangun**

Menurut (Ladjamudin 2013, 2017) menjelaskan bahwa perancangan adalah kegiatan yang memiliki tujuan untuk mendesain sistem baru yang dapat menyelesaikan masalah-masalah yang dihadapi perusahaan yang diperoleh daari pemilihan alternatif sistem yang terbaik.

dapat disimpulkan bahwa rancang bangun merupakan kegiatan menerjemahkan hasil analisa ke dalam bentuk paket perangkat lunak kemudian menciptakan sistem tersebut ataupun memperbaiki sistem yang sudah ada.

#### **2.3 Perancangan**

Menurut (M. Shalahuddin dan Rosa A.S, 2016) Desain atau perancangan adalah merupakan upaya untuk mengonstuksi sebuah sistem yang memberikan upaya kepuasan (mungkin informal) akan spesifikasi kebutuhan fungsional, memenuhi target, memenuhi kebutuhan secara implisit atau eksplisit dari segi performansi

maupun pengguna sumber daya, kepuasan batasan pada proses desain dari segi biaya, waktu, dan perangkat.

## 2.4 Perangkat Lunak

Perangkat lunak atau peranti lunak (*software*) adalah istilah khusus untuk data yang diformat, dan disimpan secara digital, termasuk program komputer, dokumentasinya, dan berbagai informasi yang bisa dibaca, dan ditulis oleh komputer. Dengan kata lain, bagian sistem komputer yang tidak berwujud. Istilah ini menonjolkan perbedaan dengan perangkat keras komputer.

Pembuatan perangkat lunak itu sendiri memerlukan “bahasa pemrograman” yang ditulis oleh seorang pemrogram untuk selanjutnya dikompilasi dengan aplikasi kompilator sehingga menjadi kode yang bisa dikenali oleh mesin hardware. Perangkat lunak seperti Windows atau Linux bisa disebut sebagai nyawa dari komputer, di mana tanpa diinstal sistem operasi tersebut maka komputer tidak dapat dijalankan. Perangkat lunak sistem operasi biasanya tersimpan di partisi C sehingga saat komputer mengalami masalah, partisi C yang terkena imbas. Perangkat lunak lain juga tersimpan di partisi C yang apabila komponen tidak lagi berjalan maksimal dapat diuninstal dan diinstal ulang. Data-data lain lebih baik disimpan di partisi D sehingga saat software terkena virus atau tidak berjalan normal kembali maka dapat diinstal ulang dan data penting dapat diselamatkan.

Di bawah ini ada beberapa contoh macam perangkat lunak, yaitu:

- a) Perangkat lunak aplikasi (*application software*) seperti pengolah kata, lembar tabel hitung, pemutar media, dan paket aplikasi perkantoran seperti OpenOffice.org.
- b) Sistem operasi (*operating system*) misalnya Linux dan Windows.
- c) Perkakas pengembangan perangkat lunak (*software development tool*) seperti Kompilator untuk bahasa pemrograman tingkat tinggi seperti Pascal dan bahasa pemrograman tingkat rendah yaitu bahasa rakitan.
- d) Pengendali perangkat keras (*device driver*) yaitu penghubung antara perangkat perangkat keras pembantu, dan komputer adalah *software* yang

banyak dipakai di swalayan, dan juga sekolah, yaitu penggunaan *barcode scanner* pada aplikasi database lainnya.

## 2.5 Diagnosa

Secara umum, Pengertian Diagnosa ialah istilah kedokteran yang berarti proses dalam menemukan penyebab pokok dari masalah-masalah organisasi yang dipergunakan. Sedangkan dalam pengertian dan perspektif lebih luas, diagnose tersebut diartikan ialah sebagai sesuatu prinsip kolaboratif antara tim manajemen dengan konsultan PO untuk dapat menemukan informasi, menganalisa, serta juga menentukan tindakan intervensi.

Diagnosa Merupakan suatu pendekatan sistematis terhadap suatu pemahaman dan juga gambaran kondisi terkini organisasi yang merinci pada hakekat permasalahan serta juga identifikasi faktor-faktor penyebab yang memberikan dasar untuk pilih strategi perubahan serta juga teknik yang paling tepat.

Orientasi masalah dalam diagnose pembahasan ini berfungsi dalam menemukan serta juga memecahkan masalah sebenarnya yang dihadapi sekolah untuk mendapatkan solusi tercepatnya. Sedangkan untuk orientasi kemajuan diagnosa ini hanya memikirkan perbaikan serta juga kemajuan systemnya.

Istilah diagnosa seringkali kita dengar dalam istilah medis. Dikemukakan oleh Thorndike dan Hagen dalam Suherman (2011), diagnosa tersebut dapat diartikan sebagai :

- a) Upaya atau juga proses dalam menemukan kelemahan atau penyakit (*weakness, disease*) apa yang dialami seseorang dengan melalui pengujian serta juga studi yang seksama mengenai gejala-gejalanya (*symptoms*).
- b) Studi yang seksama terhadap fakta mengenai suatu hal untuk dapat menemukan karakteristik atau juga kesalahan-kesalahan dan sebagainya yang esensial.
- c) Keputusan yang dicapai setelah dilakukan suatu studi yang seksama dari segala gejala-gejala atau fakta tentang suatu hal.

## 2.6 Vhd Unbk

Vhd Unbk adalah aplikasi ujian nasional yang dibuat menjadi *virtual harddisk (VHD)* yang bisa dijalankan dengan aplikasi mesin virtual. Aplikasi yang digunakan untuk menjalankan vhd unbk yaitu *oracle virtualbox*. Karena *oracle virtualbox* dianggap paling ringan oleh puspendik. *VirtualBox* adalah aplikasi virtualisasi untuk hardware x64, yang dapat digunakan pada server dan desktop. Selama ini VirtualBox menjadi pilihan aplikasi yang digunakan dalam UNBK. VirtualBox merupakan aplikasi *open source* yang *free* dengan lisensi GNU *General Public License (GPL)* versi 2, dapat berjalan di berbagai sistem operasi host (*Windows, Linux, Macintosh, dan Solaris*), serta dikembangkan oleh Oracle dan komunitas *open source*.

Untuk UNBK, sistem operasi host yang direkomendasikan adalah: Windows Server/Windows 8/Windows 7 /Linux Ubuntu 14.04 (64 bit). Sistem operasi host ini nantinya akan menjalankan sistem operasi *quest* yang digunakan dalam UNBK, yaitu Windows Server 2012. Distribusi *virtual machine* UNBK ini menggunakan *VHD (Virtual Hard Disk)*. Prinsipnya bahwa dalam 1 sistem operasi host dapat dibuat beberapa *virtual machine (VM)* untuk sistem operasi *guest*. Namun, untuk sekolah yang melaksanakan UNBK SMP dan SMA di laboratorium komputer yang sama dari tim Puspendik mengharuskan menggunakan server yang berbeda atau PC yang sama namun harddisk berbeda, perkiraan saya berkaitan dengan identifikasi *ID server* berdasarkan Hardisk yang digunakan.

## 2.7 Naive Bayes

*Algoritma Naive Bayes* merupakan sebuah metoda klasifikasi menggunakan metode probabilitas dan statistik yg dikemukakan oleh ilmuwan Inggris Thomas Bayes. *Algoritma Naive Bayes* memprediksi peluang di masa depan berdasarkan pengalaman di masa sebelumnya sehingga dikenal sebagai Teorema Bayes. Ciri utama dari *Naive Bayes Classifier* ini adalah asumsi yg sangat kuat akan independensi dari masing-masing kondisi / kejadian.

*Naive Bayes Classifier* bekerja sangat baik dibanding dengan model *classifier* lainnya. Hal ini dibuktikan pada jurnal (Xhemali, Daniela, Chris J. Hinde, 2009)

mengatakan bahwa “*Naïve Bayes Classifier* memiliki tingkat akurasi yg lebih baik dibanding model *classifier* lainnya”.

Keuntungan penggunaan adalah bahwa metoda ini hanya membutuhkan jumlah data pelatihan (*training data*) yang kecil untuk menentukan estimasi parameter yg diperlukan dalam proses pengklasifikasian. Karena yg diasumsikan sebagai variabel independent, maka hanya *varians* dari suatu variabel dalam sebuah kelas yang dibutuhkan untuk menentukan klasifikasi, bukan keseluruhan dari matriks kovarians.

Tahapan dari proses *Algoritma Naive Bayes* adalah:

- a. Menghitung jumlah kelas / label.
- b. Menghitung Jumlah Kasus Per Kelas
- c. Kalikan Semua Variable Kelas
- d. Bandingkan Hasil Per Kelas

*Bayesian classification* adalah pengklasifikasian statistik yang dapat digunakan untuk memprediksi probabilitas keanggotaan suatu class. Bayesian 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. (Kusrini, 2009). Perhitungan *Naive Bayes* yang digunakan adalah menghitung  $P(a_i|v_j)$  dengan rumus (Suanda, Wijaya, Indra & Rohadi, 2017):

$$P(a_i|v_j) = \frac{nc + m.p}{n + m}$$

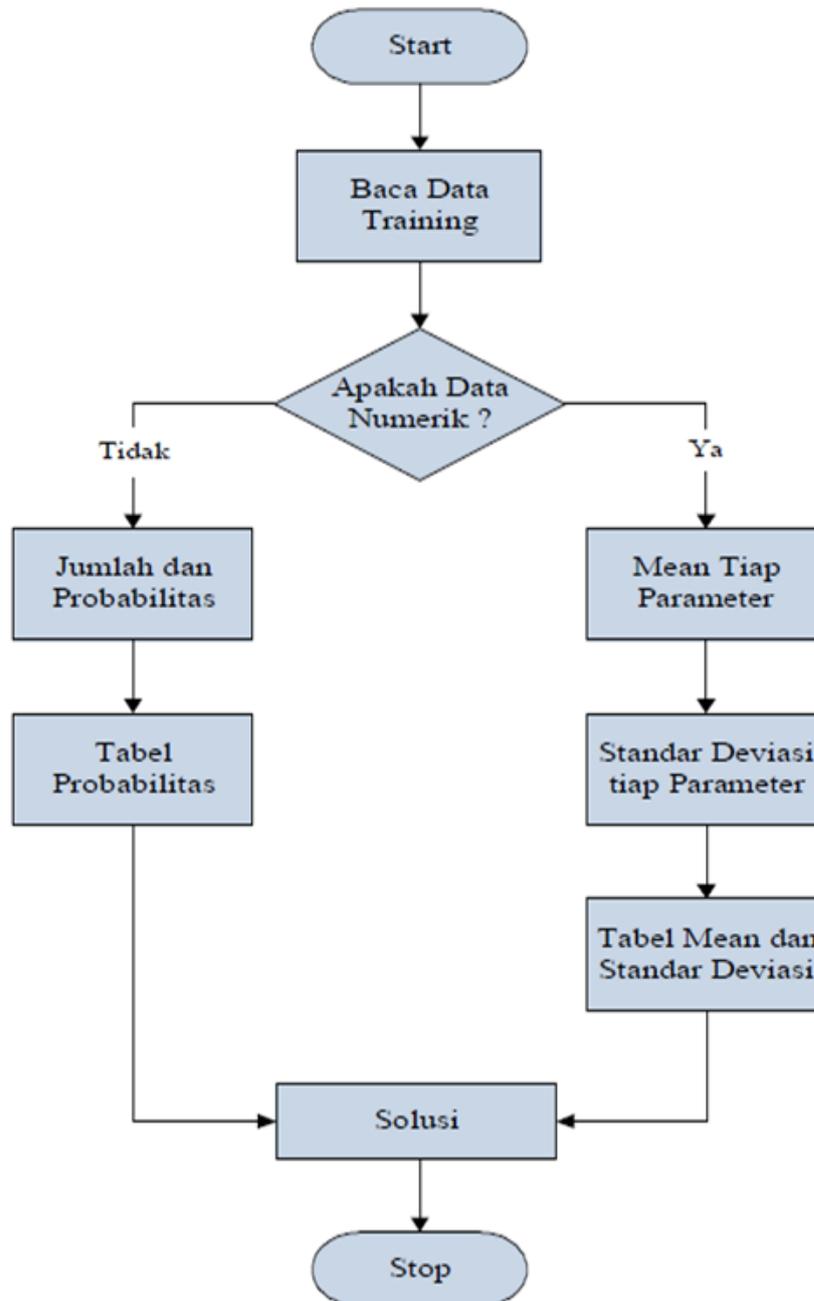
Dimana:

- nc : nilai data *record* pada data *training* yang  $v = v_j$  dan  $a = a_i$
- p : 1/banyaknya jenis kelas/kerusakan
- m : jumlah parameter/gejala
- n : nilai data *record* pada data *training* yang  $v = v_j$  / tiap kelas

Persamaan diselesaikan melalui perhitungan sebagai berikut:

1. Menentukan nilai  $n_c$  untuk setiap kelas
2. Menghitung nilai  $P(a_i|v_j)$  dan menghitung nilai  $P(v_j)$  dimana  
$$P(a_i|v_j) = \frac{nc+m.p}{n+m}$$
 dan  $P(a_i|v_j) = \frac{n}{m}$
3. Menghitung nilai  $P(a_i|v_j) \cdot P(v_j)$  untuk tiap  $v$
4. Menentukan hasil klasifikasi yaitu  $v$  yang memiliki hasil perkalian terbesar

Alur dari metode *Naive Bayes* dapat dilihat pada gambar 2 sebagai berikut:



**Gambar 2.1** Alur Metode Naive Bayes

Adapun keterangan dari gambar 2 di atas sebagai berikut:

1. Baca data training
2. Hitung Jumlah dan probabilitas, namun apabila data numerik maka:
  - a. Cari nilai mean dan standar deviasi dari masing-masing parameter yang merupakan data numerik.

Adapun persamaan yang digunakan untuk menghitung nilai rata – rata hitung (mean) dapat dilihat sebagai berikut :

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

atau

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

di mana :

$\mu$  : rata – rata hitung (mean)

$x_i$  : nilai sample ke -i

$n$  : jumlah sampel

Dan persamaan untuk menghitung nilai simpangan baku (*standar deviasi*) dapat dilihat sebagai berikut:

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

di mana :

$\sigma$  : standar deviasi

$x_i$  : nilai x ke -i

$\mu$  : rata-rata hitung

$n$  : jumlah sampel

- b. Cari nilai probabilistik dengan cara menghitung jumlah data yang sesuai dari kategori yang sama dibagi dengan jumlah data pada kategori tersebut.
3. Mendapatkan nilai dalam tabel mean, standard deviasi dan probabilitas.
4. Solusi kemudian dihasilkan.

## 2.8 Klasifikasi

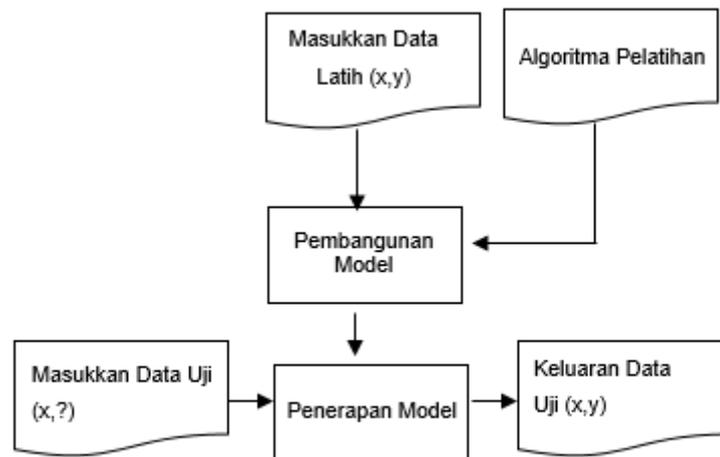
### 2.8.1 Konsep Klasifikasi

Klasifikasi merupakan suatu pekerjaan menilai objek data untuk memasukkannya ke dalam kelas tertentu dari sejumlah kelas yang tersedia. Dalam klasifikasi ada dua pekerjaan utama yang dilakukan, yaitu : pertama, Pembangunan model sebagai *prototype* untuk disimpan sebagai memori dan kedua, Penggunaan model tersebut untuk melakukan pengenalan/ klasifikasi/ prediksi pada suatu objek data lain agar diketahui di kelas mana objek data tersebut dalam model yang mudah disimpan.

Contoh aplikasi yang sering ditemui adalah pengklasifikasian jenis hewan, yang mempunyai sejumlah atribut. Dengan atribut tersebut, jika ada hewan baru, kelas hewannya bisa langsung diketahui. Contoh lain adalah bagaimana melakukan diagnosis penyakit kulit kanker melanoma (*Amaliyah et al, 2011*), yaitu dengan melakukan pembangunan model berdasarkan data latih yang ada, kemudian menggunakan model tersebut untuk mengidentifikasi penyakit pasien baru sehingga diketahui apakah pasien tersebut menderita kanker atau tidak.

### 2.8.2 Model Klasifikasi

Model dalam klasifikasi mempunyai arti yang sama dengan kotak hitam, dimana ada suatu model yang menerima masukan, kemudian mampu melakukan pemikiran terhadap masukan tersebut dan memberikan jawaban sebagai keluaran dari hasil pemikirannya. Kerangka kerja (*framework*) klasifikasi ditunjukkan pada gambar 2.2. pada gambar tersebut disediakan sejumlah data latih (x,y) untuk digunakan sebagai data pembangunan model. Model tersebut kemudian dipakai untuk memprediksi kelas dari data uji (x,y) sehingga diketahui kelas y yang sesungguhnya.



**Gambar 2.2** Proses Klasifikasi

Model yang sudah dibangun pada saat pelatihan kemudian dapat digunakan untuk memprediksi label kelas baru yang belum diketahui. Dalam pembangunan model selama proses pelatihan tersebut diperlukan suatu algoritma untuk membangunnya, yang disebut algoritma pelatihan (*learning algorithm*). Ada banyak algoritma pelatihan yang sudah dikembangkan oleh para peneliti, seperti *K-Nearest Neighbor*, *Artificial Neural Network*, *Support Vector Machine* dan sebagainya. Setiap algoritma mempunyai kelebihan dan kekurangan, tetapi semua algoritma berprinsip sama, yaitu melakukan suatu pelatihan sehingga di akhir pelatihan, model dapat memetakan (memprediksi) setiap vektor masukan ke label kelas keluaran dengan benar.

Dalam pengembangan aplikasi ini, metodologi yang digunakan adalah *Expert System Development Life Cycle*. Metode ini memiliki enam tahap siklus pengembangan, yaitu fase inialisasi kasus, fase analisis dan desain sistem, fase prototype dasar kasus, fase pengembangan sistem, fase implementasi sistem, fase implementasi tahap lanjut.

*Expert System Development Life Cycle* dipilih dengan alasan model pengembangan ini menggunakan tahapan yang dapat merepresentasikan kebutuhan pada sistem identifikasi kesalahan penggunaan yang terjadi pada aplikasi unbk dengan metode *NB (Naive Bayes)*.

### 2.8.3 *Inisialisasi Kasus*

Dalam tahapan ini ditentukan permasalahan yaitu kesalahan penggunaan yang terjadi pada aplikasi unbk atau *diagnosa error*. Berangkat dari data hasil observasi kemudian dilakukan pengkajian dan pembatasan masalah yang akan diimplementasikan ke dalam sebuah aplikasi. Masalah yang diidentifikasi dicari solusi serta fasilitas yang akan dikembangkan.

Tahapan yang dilakukan antara lain :

- a. Aquisi pengetahuan identifikasi kesalahan penggunaan yang terjadi pada aplikasi unbk, bertujuan untuk menentukan acuan dalam menentukan pola berfikir sistem yang akan dibuat.
- b. Objek observation merupakan hasil klasifikasi beberapa indikator bentuk atau yang didapatkan dalam observasi.

### 2.8.4 *Analisis Data Sistem*

Tahapan ini merupakan tahapan dimana *Knowledge Engginer* dan menentukan konsep identifikasi kesalahan penggunaan yang terjadi pada aplikasi unbk yang akan dikembangkan menjadi sistem. Tahapan konseptualisasi yang dilakukan dalam penelitian ini adalah menguraikan komponen apa saja yang dapat dijadikan faktor dari sebuah kesalahan yang muncul pada aplikasi VHD UNBK.

Dalam tahapan ini di lakukan tahapan sebagai berikut :

#### 1. *Mekanisme Inferensi*

Proses penggabungan banyak aturan berdasarkan data yang tersedia, disebut inferensi, komponen yang melakukan inferensi dalam sistem disebut mesin inferensi. Dua pendekatan populer untuk menarik kesimpulan adalah *forward chaining* dan *backward chaining* (Turban, Aronson, & Llang, 2003)

#### 2. *Forward Chaining*

Runut maju adalah aturan-aturan diuji satu demi satu dalam urutan tertentu (*data driven*). Selain teknik penalaran runut maju, diperlukan juga teknik penelusuran data dalam bentuk network atau jaringan yang terdiri dari node – node berbentuk tree atau pohon. Ada tiga teknik yang digunakan dalam proses

penelusuran data, yaitu *Depth First Search*, *Breadth First Search*, dan *Best First Search* (Andi, 2003).

### 3. *Depth First Search*

Teknik penelusuran dari node ke node bergerak menurun ke tingkat dalam yang berurutan.

### 4. **Model NB** (*Naive Bayes*)

*Naive Bayes Classifier* disebut juga *Bayesian Classification* merupakan metode pengklasifikasian statistik yang dapat digunakan untuk memprediksi probabilitas keanggotaan dari suatu *class*.

## 2.8.5 **Prototipe Dasar Kasus**

*Prototyping* sistem meliputi representasi pengetahuan yang ditangkap dengan sebuah cara yang memungkinkan inferensi dan kreasi cepat dari komponen utama pada sistem pada basis elementer (Turban, 2005).

Pengetahuan yang diperoleh dari sumber yang berpengalaman atau sekumpulan data harus direpresentasikan dalam format yang dipahami oleh manusia dan dapat dieksekusi pada komputer. Terdapat banyak metode yang berbeda untuk representasi, yang paling populer adalah aturan produksi .

Teknik representasi pengetahuan yang dilakukan pada tahap ini adalah dengan menggunakan kaidah produksi, yaitu membuat kaidah produksi berupa aturan (*rule*) yang berupa *IF* (kondisi) *THEN* (aksi) dimana kondisi merupakan bagian dari awal yang mengekspresikan situasi (pernyataan berawal *IF*) dan aksi merupakan bagian yang menyatakan suatu tindakan tertentu yang diharapkan jika suatu situasi bernilai benar (pernyataan berawalan *THEN*).

## 2.8.6 **Pengembangan Sistem**

Setelah informasi komponen-komponen kesalahan penggunaan aplikasi unbk dan kesimpulannya diformulasikan secara lengkap, kemudian diimplementasikan dengan membuat perancangan sistem yang akan dibangun. Pengembangan sistem ini terdiri atas perancangan sistem, analisis dan desain sistem, implementasi dan implementasi tahap lanjut.

Dalam pengembangan sistem ini dilakukan beberapa tahapan, yakni:

a. **Perancangan Sistem**

Dalam perancangan sistem ini dilakukan beberapa tahapan yaitu merancang flowchart, data flowchart diagram dan state transition diagram untuk aplikasi ini menggunakan tools Microsoft office visio 2003.

b. **Analisis dan Desain Sistem**

Dalam analisis dan desain sistem ini dilakukan beberapa tahapan yaitu merancang database, entity relationship diagram, *merancang Graphic User Interface* identifikasi kesalahan yang terjadi pada aplikasi unbk.

c. **Implementasi Sistem**

*System implentation* atau implementasi sistem merupakan konstruksi, instalasi, pengujian, dan pengiriman sistem kedalam produksi.

Terdapat dua fase dalam implementasi sebuah sistem, yakni fase konstruksi dan implementasi. Fase konstruksi merupakan kegiatan pengembangan, instalasi, dan pengujian terhadap komponen sistem, sedangkan fase implemenatasi merupakan kegiatan instalasi dan pengiriman sistem keseluruhan ke dalam produksi.

Dalam fase konstruksi, pemograman yang digunakan adalah PHP dengan menggunakan database MySQL serta mekanisme inferensi yang digunakan yaitu dengan pendekatan *forward chaining*.

Dalam fase implementasi, dilakukan pengujian sistem dengan menggunakan kotak hitam (*black box testing*) atau pengujian fungsional dimana pengujian prilakunya dengan mempelajari *input* dan *output* yang berkaitan. Pengujian aplikasi ini dilakukan dua penguji yaitu pengujian mandiri yang dilakukan oleh penulis dan *black box testing* yang dilakukan oleh Ichsan Tito Subur, S.Kom, selaku helpdesk propinsi yang menguasai tentang penggunaan aplikasi unbk.

### 2.8.3 Pengukuran Kinerja Klasifikasi

Sebuah sistem yang melakukan klasifikasi diharapkan dapat melakukan klasifikasi semua set data dengan benar, tetapi tidak dapat dipungkiri bahwa kinerja suatu sistem tidak bisa 100% benar sehingga sebuah sistem klasifikasi juga harus diukur kinerjanya. Umumnya, pengukuran kinerja klasifikasi dilakukan dengan matriks konfusi (*confusion matrix*).

Matriks konfusi merupakan tabel pencatat hasil kerja klasifikasi. Kuantitas matriks konfusi dapat diringkus menjadi dua nilai, yaitu akurasi dan laju *error*. Dengan mengetahui jumlah data yang diklasifikasikan secara benar, kita dapat mengetahui akurasi hasil prediksi dan dengan mengetahui jumlah data yang diklasifikasikan secara salah, kita dapat mengetahui laju *error* dari prediksi yang dilakukan. Dua kuantitas ini digunakan sebagai matrik kinerja klasifikasi.

Semua algoritma klasifikasi berusaha membentuk model yang mempunyai akurasi tinggi atau (laju *error* yang rendah). Umumnya, model yang dibangun memprediksi dengan benar pada semua data yang menjadi data latihnya, tetapi ketika model berhadapan dengan data uji, barulah kinerja model dari sebuah algoritma klasifikasi ditentukan.

## 2.9 Flowchart

*Flowchart* adalah bagan-bagan yang mempunyai arus yang menggambarkan langkah-langkah penyelesaian suatu masalah. *Flowchart* merupakan cara penyajian dari suatu algoritma (Ladjamudin, 2005 : 263).

Ada dua macam flowchart yang menggambarkan proses dengan komputer, yaitu :

### 1. *System Flowchart*

Bagan yang memperlihatkan urutan proses dalam system dengan menunjukkan alat media input, output serta jenis media penyimpanan dalam proses pengolahan data.

## 2. *Program Flowchart*

Bagan yang memperlihatkan urutan instruksi yang digambarkan dengan symbol tertentu untuk memecahkan masalah dalam suatu program (Ladjamudin, 2005: 263).

### 2.10 *DFD (Data Flow Diagram)*

*DFD (Data Flow Diagram)* adalah model dari sistem untuk menggambarkan pembagian sistem ke modul yang lebih kecil. DFD dapat memudahkan pemakai (user) yang kurang menguasai bidang komputer untuk mengerti sistem yang akan dikerjakan, urutannya sebagai berikut:

#### 1. *Diagram Konteks (Context Diagram)*

Diagram konteks adalah diagram yang terdiri dari suatu proses dan menggambarkan ruang lingkup suatu sistem (Ladjamudin, 2005 : 64).

#### 2. *Diagram Zero (Overview Diagram)*

Diagram zero adalah diagram yang menggambarkan proses dari data flow diagram.

#### 3. *Diagram Rinci (Level Diagram)*

Diagram rinci adalah diagram yang menguraikan proses yang ada dalam diagram zero.

### 2.11 *Tools Pengembangan Sistem*

Menurut (Pressman, 1997) ada tiga alasan dalam memakai rancangan untuk membuat suatu sistem :

1. Agar dapat terfokus pada bagian sistem yang penting.
2. Agar dapat terfokus pada bagian yang akan mengalami perubahan-perubahan dan koreksi, serta dokumentasi.
3. Agar dapat mengerti akan lingkungan pemakai, sehingga sistem tersebut lebih baik.

### 2.11.1 *State Transition Diagram (STD)*

*State transition* diagram merupakan suatu diagram yang menggambarkan bagaimana state dihubungkan dengan state yang lain pada satu waktu. *State Transition Diagram* menggambarkan suatu state yang mempunyai kondisi dimana dapat menyebabkan perubahan satu state ke state yang lain (Hoffer, George, dan Valacich, 1996:364). *State Transition Diagram* pada dasarnya merupakan sebuah diagram yang terdiri dari state dan transisi atau perpindahan state. Transisi atau perpindahan state terdiri dari kondisi dan aksi. Transisi diantara kedua keadaan pada umumnya disebabkan oleh suatu kondisi. Kondisi adalah suatu kejadian yang dapat diketahui oleh sistem. Sedangkan aksi adalah tindakan yang dilakukan oleh sistem apabila terjadi perubahan state atau merupakan reaksi dari sistem.

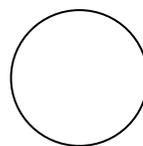


**Gambar 2.3** Contoh Perubahan State

Adapun komponen atau simbol yang digunakan dalam diagram ini adalah:

- a. Modul

Menggunakan simbol lingkaran kecil (Gambar 2.12 yang mewakili modul yang dipanggil apabila terjadi suatu tindakan.



**Gambar 2.4** Notasi Modul

b. Tampilan kondisi (*state*)

Merupakan layer yang ditampilkan menurut keadaan atau atribut, untuk memenuhi suatu tindakan pada waktu tertentu yang mewakili suatu bentuk keberadaan atau kondisi tertentu, menggunakan simbol kotak (Gambar 2.13).



**Gambar 2.5** Notasi Tampilan

c. Tindakan (*state transition*)

Menggunakan symbol anak panah (Gambar 2.9) disertai keterangan tindakan yang dilakukan.



**Gambar 2.6** Notasi Tindakan

## 2.12 Alat Perancangan Database

### 2.12.1 Entity Relationship Diagram (ERD)

*Entity Relationship Diagram* pertama kali diperkenalkan oleh Peter Chen (1976) (dalam Kowal, 1992:104) adalah sebuah ilustrasi entitas secara grafik (juga dikenal sebagai objek), atribut, dan relasi (*relationship*) yang ada diantara mereka.

Dalam Fathansyah (1999:70) disebutkan bahwa model *Entity Relationship Diagram* yang berisi komponen-komponen himpunan entitas dan relasi masing-masing dilengkapi dengan atribut-atribut yang merepresentasikan seluruh fakta dari “dunia nyata” yang kita tinjau, dapat digambarkan dengan lebih sistematis dengan menggunakan *Entity Relationship Diagram* (ERD).

Komponen-komponen yang ada dalam *ERD*, adalah:

### 1. Entitas

Merupakan representasi dari sebuah objek tempat dimana sistem menyimpan data. Simbol untuk entitas digambarkan dengan empat persegi panjang.

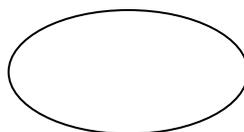
### 2. Himpunan Relasi

Himpunan relasi adalah perantara yang menghubungkan antara dua atau lebih entitas. Ada empat jenis relasi, yaitu :

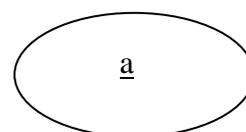
- a. *Relasi one to one* : relasi yang ada ketika kita menghubungkan tepat 1 data pada suatu tabel dengan 1 data lainnya pada tabel lain.
- b. *Relasi one to many* : relasi yang terbentuk ketika kita menghubungkan secara tepat 1 data pada suatu tabel dengan 1 atau lebih data pada tabel lainnya.
- c. *Relasi many to one* : relasi yang terbentuk ketika kita menghubungkan secara tepat 1 atau lebih data pada suatu tabel dengan 1 data pada tabel lainnya.
- d. *Relasi many to many* : relasi yang terbentuk ketika kita menghubungkan secara tepat 1 atau lebih data pada suatu tabel dengan 1 atau lebih data pada tabel lainnya.

### 3. Atribut

Atribut adalah data yang dihubungkan dengan entitas. Atribut berguna untuk menjelaskan, mengidentifikasi, dan mengekspresikan hubungan antar entitas. Simbol atribut digambarkan dengan ellips.



**Gambar 2.7** Simbol Atribut



**Gambar 2.8** Simbol Atribut  
a sebagai *key*

## 2.13 Alat Pengembangan Aplikasi

### 2.13.1 PHP

#### 1. Sejarah PHP

PHP pertama kali dibuat musin gugur tahun 1994 oleh Rasmus Lerdoff, awalnya digunakan pada websitenya untuk mencatat siapa saja yang berkunjung dan melihat biodatanya. Versi pertama yang di-release tersedia pada awal tahun 1995, dikenal sebagai tool Personal Home Page, yang terdiri atas engine parser yang sangat sederhana yang hanya mengerti beberapa makro khusus dan sejumlah utilitas yang sering digunakan pada halaman-halaman web, seperti buku tamu, counter pengunjung, dan lainnya. Parser diprogram ulang pada pertengahan 1995 dan diberi nama PHP/F1 versi 2.0. F1 berasal dari paket Rasmus lainnya yang ditulis untuk menginterpretasikan data dari form, yang kemudian dikombinasikan dengan tool Personal Home Page dan ditambahkan dukungan untuk database mSQL (mini SQL).

Tahun 1995 ini dianggap sebagai tahun kelahiran dari PHP/F1 yang kemudian membuat pertumbuhan aplikasi web yang pesat, dan banyak orang kemudian berkontribusi mengembangkan PHP/F1. Sulit untuk mendapatkan statistik yang tepat untuk memperikan penggunaan PHP/F1, tetapi diperkirakan pada akhir 1996 telah digunakan oleh sedikitnya 15000 web site di seluruh dunia. Dan pertengahan 1997 mencapai 50000 situs.

Pada pertengahan 1997 ini juga terjadi perubahan pengembangan PHP. Pengembangan dilakukan oleh tim yang terorganisasi bukan oleh Rasmus sendiri saja lagi. Parser dikembangkan oleh Zeev Suraski dan Andi Gutmans yang kemudian menjadi dasar untuk versi 3, dan banyak utilitas tambahan yang deprogram untuk menambah kemampuan dari versi 2. Versi terakhir (PHP 4) menggunakan *engine script Zend* untuk lebih meningkatkan kinerja (performance) dan mempunyai dukungan yang banyak berupa ekstensi dan fungsi dari berbagai library pihak ketiga (*third party*), dan berjalan seolah modul asli (*native*) dari berbagai server web yang populer.

Sejak tahun 2001 PHP3 dan PHP4 disertakan pada sejumlah produk server web komersial seperti server web komersial seperti seperti server *web StrongHold RedHat*. Perkiraan konservatif yang didapat dari angka yang diberikan oleh Netcraft yang

diekstrapolasi, pengguna PHP sekitar 5.100.000 sedikit lebih banyak dari server web yang menggunakan Microsoft IIS (5.03 juta) di Internet. Berdasarkan data terakhir dari Netcraft Maret 2002 pemakai PHP telah mencapai hamper 9 juta nama domain.

### **2.13.2 *Sublime Text***

*Sublime Text* adalah aplikasi editor untuk kode dan teks yang dapat berjalan diberbagai *platform operating system* dengan menggunakan teknologi *Phyton* API. Terciptanya aplikasi ini terinspirasi dari aplikasi Vim, Aplikasi ini sangatlah fleksibel dan *powerfull*. Fungsionalitas dari aplikasi ini dapat dikembangkan dengan menggunakan *sublime-packages*. *Sublime Text* bukanlah aplikasi *opensource* dan juga aplikasi yang dapat digunakan dan didapatkan secara gratis, akan tetapi beberapa fitur pengembangan fungsionalitas (*packages*) dari aplikasi ini merupakan hasil dari temuan dan mendapat dukungan penuh dari komunitas serta memiliki linsensi aplikasi gratis.

*Sublime Text* mendukung berbagai bahasa pemrograman dan mampu menyajikan fitur *syntax highlight* hampir di semua bahasa pemrograman yang didukung ataupun dikembangkan oleh komunitas seperti; *C, C++, C#, CSS, D, Dylan, Erlang, HTML, Groovy, Haskell, Java, JavaScript, LaTeX, Lisp, Lua, Markdown, MATLAB, OCaml, Perl, PHP, Python, R, Ruby, SQL, TCL, Textile and XML*. Biasanya bagi bahasa pemrograman yang didukung ataupun belum terdukung secara default dapat lebih dimaksimalkan atau didukung dengan menggunakan *add-ons* yang bisa didownload sesuai kebutuhan user.

### **2.13.3 *XAMPP***

*XAMPP* merupakan aplikasi server yang menggabungkan beberapa aplikasi *server* yang biasa digunakan di *web server*. Berikut beberapa komponen-komponen yang terdapat pada *XAMPP*, yaitu *Apache (web server), MySQL (database server), Filezila FTP server, Mercury Mail (mail server), phpMyAdmin (web-based interface MySQL)*.

## 2.13.4 MySQL

### 1. Mengenal MySQL

Menurut Nugroho (2013:26) MySQL adalah *software* atau program *Database Server*. Sedangkan SQL adalah bahasa pemrogramannya, bahasa permintaan (*query*) dalam *database server* termasuk dalam MySQL itu sendiri. SQL juga dipakai dalam *software database server* lain, seperti *SQL Server*, *Oracle*, *PostgreSQL* dan lainnya.

MySQL adalah sebuah program database server yang mampu menerima dan mengirimkan datanya dengan sangat cepat, *multi user* serta menggunakan perintah standar *SQL (Structured Query Language)*.

MySQL memiliki dua bentuk lisensi, yaitu *FreeSoftware* dan *Shareware*. MySQL yang biasa kita gunakan adalah MySQL *FreeSoftware* yang berada di bawah Lisensi *GNU/GPL (General Public License)*.

MySQL merupakan sebuah database server yang *free*, artinya kita bebas menggunakan database ini untuk keperluan pribadi atau usaha tanpa harus membeli atau membayar lisensinya. MySQL pertama kali dirintis oleh seorang programmer database bernama Michael Widenus.

Selain untuk database server, MySQL juga merupakan program yang dapat mengakses suatu database MySQL yang berposisi sebagai server. Pada saat itu berarti program kita berposisi sebagai client. Jadi MySQL adalah sebuah database yang dapat digunakan baik sebagai Client maupun Server.

Database MySQL, biasa kita baca mai-es-ki-el atau bisa juga mai-se-kuel, merupakan suatu perangkat lunak database yang berbentuk database relasional atau dalam bahasa basisdata sering kita sebut dengan *Relation Database Management System (RDBMS)* yang menggunakan suatu bahasa permintaan bernama SQL.

Sebagai pengguna awal database mungkin anda merasa bingung dengan kedua istilah antara *MySQL* dengan *SQL*. Nah, yang perlu anda pahami adalah bahwa kedua istilah tersebut sangat berbeda artinya. *MySQL* adalah sebuah program database, sedangkan *SQL* adalah bahasa perintah (*Query*) dalam program *MySQL* (Nugroho, 2005 : 1).

## 2. Kelebihan MySQL

MySQL adalah sebuah *database server*, dapat juga berperan sebagai *client* sehingga sering disebut *database client/server*, yang pen source dengan kemampuan dapat berjalan baik di OS (Operating Sistem) manapun, dengan Platform Windows maupun linux. Selain itu database ini memiliki beberapa kelebihan disbanding database lain, diantaranya adalah:

- a) MySQL sebagai Database Management System (DBMS)
- b) MySQL sebagai Relation Database Management System (RDBMS)
- c) MySQL adalah sebuah Software database yang OpenSource, artinya program ini bersifat free atau bebas digunakan oleh siapa saja tanpa harus membeli dan membayar lisensi kepada pembuatnya.
- d) MySQL merupakan sebuah database server, jadi dengan menggunakan database ini anda dapat menghubungkannya ke media internet sehingga dapat diakses dari jauh.
- e) MySQL merupakan sebuah client. Selain menjadi server yang melayani permintaan, MySQL juga dapat melakukan query yang mengakses database pada Server. Jadi MySQL dapat juga berperan sebagai Client.
- f) MySQL mampu menerima query yang bertumpuk dalam satu permintaan atau yang disebut Multi-Threading.
- g) MySQL merupakan sebuah database yang mampu mnyimpan data berkapasitas besar hingga berukuran Gigabite sekalipun.
- h) MySQL database menggunakan enkripsi password. Jadi database ini cukup aman karena memiliki password untuk mengaksesnya.
- i) MySQL merupakan Server database yang multi user, artinya database ini tidak hanya digunakan oleh sepihak orang akan tetapi merupakan database yang dapat digunakan oleh banyak pengguna.

Dengan beberapa kelebihan yang dimiliki di atas, MySQL menjadi sebuah program database yang sangat populer digunakan. Pada umumnya MySQL digunakan sebagai database yang diakses melalui web.

## DAFTAR PUSTAKA

- Artaye, K. (2015). *International Conference On Information Technology And Business ISSN 2460-7223 IMPLEMENTATION OF NAÏVE BAYES CLASSIFICATION METHOD TO PREDICT GRADUATION TIME OF IBI DARMAJAYA SCHOLAR Z. A . Pagar Alam Street No . 93 Bandar Lampung.* (August), 284–290.
- Ladjamudin 2013. (2017). Analisis Laporan Penjualan Menggunakan Website Dalam Kaitannya Terhadap Persediaan Barang Dagang Pada Cv . Pacific Accessories & Car Audio. *Pengolahan Data.*
- Luh, N., & Pivin, G. (n.d.). *Penerapan Metode Naive Bayes Untuk Menganalisa Kerusakan Automatic Transmission Pada Mobil.* 41–51.
- M. Shalahuddin dan Rosa A.S. (2016). Rekayasa Perangkat Lunak dan Berorientasi Objek. *SemanTIK.* <https://doi.org/10.1249/MSS.0b013e3181946lc2>
- Marcos, H., & Utomo, H. (2015). Perbandingan Kinerja Algoritme C.45 Dan Naive Bayes Mengklasifikasi Penyakit Diabete. *Jurnal Informatika Darmajaya,* 15(2), 141–148. <https://doi.org/10.30873/ji.v15i2.596>
- Pramudia, H., & Nugroho, A. (2017). Sistem Informasi Kerusakan Laptop Menggunakan Metode Naïve Bayes. *Jurnal Teknologi Elektro, Universitas Mercu Buana,* 8(3), 206–214.
- Pressman, R. (1997). Breathing New Life into the Waterfall Model. *IEEE Software.* <https://doi.org/10.1109/52.605938>
- Suanda, Luis; Wijaya, Indra Dharma; Rohadi, E. (2017). *Jurnal Program Studi Teknik Informatika Politeknik Negeri Malang.*
- Turban, E., Aronson, J., & Llang, T. (2003). Decision Support Systems and Intelligent Systems. In *Decision Support Systems and Intelligent Systems.*

Xhemali, Daniela, Chris J. Hinde, and R. G. S. (2009). *Naive Bayes vs. decision trees vs. neural networks in the classification of training web pages.*