

**KLASIFIKASI PENDIVISIAN ORGANISASI PANKER
MENGUNAKAN METODE NAIVE BAYES**

Tugas Akhir

diajukan untuk memenuhi sebagian persyaratan mencapai
gelar Ahli Madya pada jenjang Diploma III
Jurusan Teknik Informatika



Oleh :

KHAIRUL ANWAR

NIM. 1603044

**TEKNIK INFORMATIKA
POLITEKNIK NEGERI INDRAMAYU**

2019

HALAMAN PENGESAHAN

Tugas Akhir ini ajukan oleh :

Nama : Khairul Anwar
NIM : 1603044
Program Studi : Teknik Informatika
Judul : Klasifikasi Pendivisian Organisasi Panker
Menggunakan Metode Naive Bayes
Pembimbing : 1. Eka Ismantohadi., S.Kom., M.Eng
2. Muhammad Anis Al Hilmi, S.Si.,M.T

Telah berhasil dipertahankan dihadapan dewan penguji pada tanggal 2019 dan diterima sebagai bagian persyaratan yang diperlukan untuk memperoleh gelar Ahli Madya Program Studi Teknik Informatika Jurusan Teknik Informatika Politeknik Negeri Indramayu

DEWAN PENGUJI

Nama	Jabatan	Tandatangan	Tanggal
1.	Ketua Penguji
2.	Sekretaris Penguji
3.	Anggota

Indramayu, 2019
Ketua Jurusan Teknik Informatika

Iryanto, S.Si., M.Si
NIK. 16039001

PERNYATAAN KEASLIAN

Saya menyatakan dengan sesungguhnya bahwa Tugas Akhir ini adalah hasil karya saya sendiri, dan sepanjang pengetahuan saya tidak terdapat karya atau pendapat yang pernah ditulis atau dipublikasikan oleh orang lain, kecuali yang secara tertulis disebutkan sumbernya dalam naskah dan dalam daftar pustaka.

Indramayu, 17 Agustus 2019
Yang menyatakan,

Khairul Anwar
NIM.1603044

ABSTRAK

Pecinta Alam Negeri 1 Kertasemaya (PANKER) merupakan organisasi ekstrakurikuler pecinta alam yang mempunyai hubungan koordinatif dengan Organisasi Intra Sekolah (OSIS) yang berada di Sekolah Menengah Atas Negeri 1 Sukagumiwang. Panker merupakan organisasi kepecinta alaman yang memiliki 3 divisi dalam hal ilmu kepecinta alaman yaitu :

1. Survival
2. Search And Rescue (SAR).
3. Rock Climbing(RC).

Dalam pembagian divisi pada organisasi panker masih menggunakan cara pembagian divisi secara manual dan sistem tunjuk, Hal ini berpotensi mengurangi bakat yang dimiliki oleh calon pengurus untuk menyalurkan ilmu secara efektif kepada generasi penerus selanjutnya, hal ini akan berakibat pada penurunan kualitas pengajaran dibidang materi dikarenakan cara pengajaran kurang begitu baik dikarenakan pengaruh pembagian divisi pada awal pembagian. Berdasarkan analisis tersebut maka dibuatlah aplikasi klasifikasi pendivisian yang berjudul “Klasifikasi Pendivisian Organisasi PANKER menggunakan metode naive bayes” agar memudahkan anggota pengurus untuk menentukan penempatan divisi yang cocok untuk calon pengurus dengan melihat aspek nilai terbaik dengan metode perhitungan naive bayes. Sistem yang dibangun menggunakan bahasa pemrograman PHP, database PhpMyadmin dengan server xampp.

Hasil yang diperoleh yaitu sebuah aplikasi klasifikasi yang dapat digunakan oleh dewan kerja atau pengurus organisasi pecinta alam negeri 1 kertasemaya (PANKER) Sukagumiwang 1 Public High School to classify prospective administrators and determine the division recommendations that are suitable for them.

Kata kunci : Naive Bayes, Survival , SAR, RC, *waterfall*.

MOTTO

“Kalau Ingin Tetap Hidup Jangan Lupa Bernapas”

KATA PENGANTAR

Assalamu'alaikum Warahmatullahi Wabarakatuh.

Puji syukur penulis panjatkan kehadirat Allah Swt yang telah memberikan rahmat dan hidayah-Nya sehingga penulis dapat menyelesaikan Tugas Akhir yang berjudul “KLASIFIKASI PENDIVISAN ORGANISASI PANKER MENGGUNAKAN METODE NAIVE BAYES” sesuai dengan waktu yang telah ditentukan. Sholawat serta salam penulis curahkan kepada Baginda Nabi Muhammad Saw yang telah memberikan cahaya dalam islam sehingga tidak lagi berada di zaman jahiliyyah.

Tugas akhir ini merupakan salah satu tugas yang wajib ditempuh oleh mahasiswa tingkat akhir. Penulisan laporan tugas akhir ini dibuat sebagai persyaratan utama untuk dapat dinyatakan lulus sebagai Ahli Madya Diploma 3.

Penulis dapat menyelesaikan tugas akhir ini tidak lain karena adanya dukungan dan doa dari berbagai pihak yang tidak ternilai besarnya. Oleh karena itu penulis menyampaikan terimakasih kepada:

1. Allah Swt yang telah memberikan kemudahan dan kelancaran dalam setiap proses pengerjaanya.
2. Kedua orangtua penulis (Bapak Dulkayi dan IbuWastinih) yang telah medoakan dan memotivasi agar penulis terus berusaha dan tidak menyerah dalam keadaan apapun untuk menyelesaikan laporan ini dengan tepat waktu.
3. Direktur Politeknik Negeri Indramayu Bapak Casiman Sukardi, ST.M.T
4. Bapak Iryanto selalu Ketua Jurusan Teknik Informatika Politeknik Negeri Indramayu
5. Bapak Eka Ismantohadi,S.Kom.,M.Eng. selaku pembimbing I.
6. Bapak Muh. Anis Al Hilmi,S.Pd.,M.Kom. selaku pembimbing II
7. Seluruh Dosen Teknik Informatika Polindra yang mensuport serta memberikan doa yang terbaik untuk mahasiswanya.
8. Teman-teman Kontrakan Club yang telah membantu dalam proses pembuatan Tugas Akhir dalam bentuk doa, materil maupun keuangan.
9. Teman-teman Teknik Informatika khususnya kelas D3TI.3B yang telah membantu dalam proses pembuatan Tugas Akhir.

Penulis telah berusaha semaksimal mungkin untuk menyelesaikan laporan tugas akhir ini dengan baik dan benar, tetapi manusia tempatnya salah dan lupa karena kesempurnaan hanya milik Allah SWT semata. Apabila terdapat kesalahan dalam penulisan tugas akhir ini penulis minta maaf yang sebesar-besarnya dan sangat menerima kritik dan saran yang diberikan pembaca untuk kedepannya dijadikan sebagai bekal yang lebih matang lagi. Penulis berharap semoga laporan ini dapat bermanfaat bagi Almamater dan rekan mahasiswa lainnya.

Wassalamu'alaikum Warahmatullahi Wabarakatuh.

Indramayu, 16 Juli 2019

Penulis

DAFTAR ISI

HALAMAN PENGESAHAN.....	ii
PERNYATAAN KEASLIAN.....	iii
ABSTRAK	iv
MOTTO.....	v
KATA PENGANTAR	vi
DAFTAR ISI.....	viii
DAFTAR TABEL.....	xii
DAFTAR GAMBAR	xiii
DAFTAR LAMPIRAN.....	xv
BAB 1 PENDAHULUAN	1
1.1 Latar Belakang Masalah.....	1
1.2 Rumusan Masalah	1
1.3 Batasan Masalah.....	1
1.4 Tujuan Penelitian.....	2
1.5 Manfaat Penelitian.....	2
1.6 Sisitematis Penulisan Laporan.....	2
BAB II LANDASAN TEORI	4
2.1 Naive Bayes.....	4
2.2 Teorema Naive Bayes.....	4
2.3 XAMPP	8
2.4 PhpMyadmin	8
2.5 PHP.....	9
2.6 Pengenalan PHP	9
2.7 Basis Data.....	10

2.8	Website	11
2.9	HTML(<i>Hyper Text Markup Language</i>).....	12
2.9.1	Definnisi HTML(<i>Hyper Text Markup Language</i>)	12
2.9.2	Pengenalan dasar HTML (<i>Hyper Text Markup Language</i>).....	12
2.10	MySql.....	14
2.11	UML(<i>Unfied Modeling Language</i>).....	15
2.12	Laravel	16
2.13	Composer	18
2.14	Black Box Testing	18
2.15	Bootstrap.....	19
2.16	Google Chrome.....	19
2.17	Javascript	19
BAB III METODOLOGI PENELITIAN.....		21
3.1	Metodologi Penelitian	21
3.2	Metode Pengumpulan Data	22
3.2.1	Wawancara.....	22
3.2.2	Observasi	23
3.2.3	Studi Pustaka.....	23
3.3	Analisis Kebutuhan Sistem	23
3.3.1	Kebutuhan Hardware	23
3.3.2	Kebutuhan Software	24
3.4	Perancangan Sistem.....	24
3.4.1	Use Case Diagram	24
3.4.2	Activity Diagram	25
3.4.3	Flowchart Aplikasi.....	29
3.4.4	Perancangan Awal Antar Muka.....	29

A.	Rancangan Halaman Awal	30
B.	Rancangan Halaman Login	30
C.	Rancangan Halaman Tampil Data Training	31
D.	Rancangan Halaman Tampil Data Testing.....	31
E.	Rancangan View Data Tesing	32
F.	Rancangan Halaman Tampil Export PDF.....	33
3.4.5	Perancangan Database	33
A.	Tabel Users.....	33
B.	Tabel training.....	34
C.	Tabel testing	35
BAB	IV HASIL DAN PEMBAHASAN.....	36
4.1	Hasil Penelitian.....	36
4.2	Hasil Pustaka	36
4.3	Implementasi Desain Antar Muka.....	36
4.3.1	Implementasi Desain Antar Muka	36
4.3.2	Implementasi Hasil Tampilan Intro Aplikasi.....	36
4.3.3	Implementasi Hasil Tampilan Login	37
4.3.4	Implementasi Hasil Tampilan Data Training.....	37
4.3.5	Implementasi Hasil Tampilan Data Testing	38
4.3.6	Implementasi Form Testing Klasifikasi.....	38
4.3.7	Implementasi Hasil Tampilan Modal View Testing.....	39
4.3.8	Implementasi Hasil Tampilan Export Pdf	40
4.4	Implementasi Database.....	41
4.4.1	Tabel users	41
4.4.2	Tabel Training.....	41
4.4.3	Tabel Testing	42

4.5	Tahap Pengujian Sistem	42
4.5.1	User Melakukan Login	42
4.5.2	User Memilih Menu Training	44
4.5.3	User Mengimport Data Training	44
4.5.4	User Memilih Menu Testing	45
4.5.5	User Melakukan Klasifikasi Calon Pengurus	45
4.5.6	User Melihat Data Testing	46
4.5.7	User Melihat Hasil Data Testing	47
4.5.8	User Mengexport Pdf Hasil Testing	47
4.6	Hasil Pengujian <i>Black-Box</i>	48
4.7	Kelebihan dan Kekurangan	49
BAB V PENUTUP		50
5.1	Kesimpulan	50
5.2	Saran	50
DAFTAR PUSTAKA		51
LAMPIRAN		53

DAFTAR TABEL

Tabel 3.1 Kebutuhan Hardware	23
Tabel 3.2 Kebutuhan Software.....	24
Tabel 3.3 Tabel users	34
Tabel 3.4 Tabel training	34
Tabel 3.5 Tabel testing	35

DAFTAR GAMBAR

Gambar 2.1 Logo Xampp.....	8
Gambar 2.2 Logo PhpMyAdmin.....	8
Gambar 2.3 Penjelasan Simbol Entitas	11
Gambar 2.4 Penulisan Element HTML Yang Tumpang Tindih.....	13
Gambar 2.5 Penulisan Element HTML Yang Normal.....	13
Gambar 3.1 Desain Penelitian.....	21
Gambar 3.2 Use Case.....	25
Gambar 3.3 Activity Diagram Login User.....	25
Gambar 3.4 Activity Diagram Import Data Training.....	26
Gambar 3.5 Activity Diagram Klasifikasi Testing	27
Gambar 3.6 Activity Export Pdf	28
Gambar 3.7 Gambar Flowcart.....	29
Gambar 3.8 Rancangan Halaman Awal	30
Gambar 3.9 Rancangan Halaman Login	30
Gambar 3.10 Rancangan Halaman Tampil Data Training.....	31
Gambar 3.11 Rancangan Halaman Tampil Data Testing.....	32
Gambar 3.12 Rancangan Halaman Tampil Data.....	32
Gambar 3.13 Rancangan Halaman Tampil Data Export Pdf Per-id	33
Gambar 4.1. Tampilan Halaman Intro	37
Gambar 4.2 Tampilan Halaman Login.....	37
Gambar 4.3 Tampilan Tabel Training.....	38
Gambar 4.4 Tampilan Halaman Testing	38
Gambar 4.5 Tampilan Form Testing.....	39

Gambar 4.6 Tampilan Tabel Testing	39
Gambar 4.7 Tampilan Modal View Testing	40
Gambar 4.8 Tampilan Tabel Testing	40
Gambar 4.9 Tampilan Export Pdf	40
Gambar 4.10 Gambar Tabel user	41
Gambar 4.11 Gambar Tabel training	41
Gambar 4.12 Gambar Tabel testing	42
Gambar 4.13 Tampilan Halaman Login.....	42
Gambar 4.14 Gambar Setelah Berhasil Melakukan Login	43
Gambar 4.15 Ketika user salah memasukan username / Password	43
Gambar 4.16 Tampilan Menu dan sub-menu.....	44
Gambar 4.17 Gambar Import Data Training.....	44
Gambar 4.18 Gambar Pilih Data Excel.....	44
Gambar 4.19 Gambar Status Berhasil Import	45
Gambar 4.20 Tampilan Menu dan Sub-menu	46
Gambar 4.21 Form Klasifikasi Calon Pengurus	46
Gambar 4.22 Tabel Testing.....	46
Gambar 4.23 View Modal Testing.....	47
Gambar 4.24 Tampilan Export Testing.....	47

DAFTAR LAMPIRAN

Lampiran 1 Hasil Pengumpulan Data Training

Lampiran 2 Kode Program

Lampiran 3 Biodata Penulis

BAB 1

PENDAHULUAN

1.1 Latar Belakang Masalah

Didalam sebuah organisasi pendivisian sangatlah penting, karena tujuan pendivisian untuk mengelompokkan masing-masing pengurus pada kemampuan, tugas, dan kurikulum dari organisasi hal ini sebab hal ini yang akan berpengaruh kepada kualitas organisasi dari penempatan anggota pengurus pada divisi yang akan berpengaruh pada kurikulum yang akan disampaikan kepada calon pengurus yang akan menjadi penerus menggantikan pengurus yang lama.

Organisasi PANKER didalam pendivisian masih menggunakan sistem manual yang masih ada sedikit kesalahan dikarenakan *human error* dalam penempatan anggota divisi tanpa melihat kemampuan secara spesifikasi dari calon pengurus yang akan menjadi pengurus.

Berdasarkan uraian diatas maka penulis akan membuat suatu sistem yaitu “Klasifikasi Pendivisian Pada Organisasi Panker Menggunakan Metode Naive Bayes”. Dengan adanya sistem ini penulis harap dapat membantu mengklasifikasikan pendivisian pada organisasi panker sehingga lebih selektif dalam penempatan dalam penempatan pengurus.

1.2 Rumusan Masalah

Berdasarkan latar belakang di atas, maka beberapa lingkup permasalahan yang akan dibahas dalam penelitian Tugas Akhir ini diantaranya :

1. Bagaimana hasil dari pembutaan website klasifikasi pendivisian pada organisasi PANKER menggunakan metode naive bayes ?
2. Bagaimana pengaruh perubahan sistem pendivisian pada organisasi PANKER menggunakan sistem komputerisasi untuk rekomendasi pendivisian ?
3. Bagaimana cara mengetahui hasil dari pemrosesan pendivisian tersebut ?

1.3 Batasan Masalah

Agar penelitian ini dapat terarah dan permasalahan yang dihadapi tidak terlalu luas serta sesuai dengan judul penelitan Tugas Akhir yang diajukan maka

ditetapkan batasan-batasan terhadap masalah yang diteliti. Adapun beberapa hal yang membatasi dalam penelitian Tugas Akhir ini, antara lain adalah :

1. Aplikasi Klasifikasi Pendivisian Pada Organisasi Panker yang dibangun hanya untuk pengurus organisasi PANKER.
2. Sistem ini dibangun menggunakan teknologi *web-based* dengan tools PHP, MYSQL, APACHE dan *software* pendukung lainnya.
3. Perancangan sistem menggunakan metode naive bayes.

1.4 Tujuan Penelitian

Tujuan diadakannya penelitian ini, diantaranya adalah sebagai berikut.

1. Menghasilkan rekomendasi divisi yang akan diberikan untuk calon pengurus selanjutnya.
2. Membantu pengurus untuk mengklasifikasikan tiap tiap calon pengurus.
3. Membuat aplikasi klasifikasi pendivisian pada organisasi panker dengan metode naive bayes.

1.5 Manfaat Penelitian

Dengan adanya penelitian ini diharapkan dapat memberikan manfaat yaitu:

1. Memudahkan pengurus untuk menentukan divisi yang direkomendasikan.
2. Memberikan kemudahan dalam melihat hasil rekomendasi.
3. Dapat mempersingkat waktu dalam penentuan klasifikasi.

1.6 Sistematis Penulisan Laporan

Untuk memenuhi kaedah penulisan laporan dan pemahaman yang lebih sistematis, penulisan laporan tugas akhir ini tersusun dalam lima bab sesuai dengan panduan penulisan tugas akhir yang dikeluarkan oleh Politeknik Negeri Indramayu dengan penjelasan sebagai berikut :

BAB I PENDAHULUAN

Merupakan pendahuluan laporan penelitian tugas akhir yang mengemukakan Latar Belakang Masalah, Rumusan Masalah, Batasan Masalah, Manfaat Penelitian dan Sistematika Penulisan Laporan

BAB II LANDASAN TEORI

Bagian ini berisikan tentang landasan teori yang berkaitan dengan proses penelitian tugas akhir seperti metode yang digunakan, Web, Framework Laravel, dan *Database*.

BAB III METODOLOGI PENELITIAN

Bagian ini menjelaskan tentang alur sistem dari pembuatan aplikasi. Terdapat flowchart, use case, dan class diagram yang menggambarkan alur dari pengguna aplikasi serta rancangan database.

BAB IV HASIL DAN PEMBAHASAN

Pada bab ini dibahas hasil-hasil dari tahapan analisis, tahapan implementasi dari penelitian yang telah dilakukan.

BAB V PENUTUP

Bagian terakhir yang berisikan tentang kesimpulan dari penelitian dan terdapat saran-saran berdasarkan penelitian tugas akhir ini.

BAB II

LANDASAN TEORI

2.1 Naive Bayes

Naïve Bayes merupakan sebuah pengklasifikasian probabilistik 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 variabel kelas (Patil and Sherekar 2013).

Definisi lain mengatakan Naïve Bayes merupakan pengklasifikasian dengan metode probabilitas dan statistik yang dikemukakan oleh ilmuwan Inggris Thomas Bayes, yaitu memprediksi peluang di masa depan berdasarkan pengalaman di masa sebelumnya (Bustami 2013).

Naïve Bayes didasarkan pada asumsi penyederhanaan bahwa nilai atribut secara kondisional saling bebas jika diberikan nilai output. Dengan kata lain, diberikan nilai output, probabilitas mengamati secara bersama adalah produk dari probabilitas individu (Ridwan 2013). Keuntungan penggunaan Naive Bayes adalah bahwa metode ini hanya membutuhkan jumlah data pelatihan (Training Data) yang kecil untuk menentukan estimasi parameter yang diperlukan dalam proses pengklasifikasian. Naive Bayes sering bekerja jauh lebih baik dalam kebanyakan situasi dunia nyata yang kompleks dari pada yang diharapkan (Pattekari and Parveen 2012).

Naive Bayes Classifier dinilai bekerja sangat baik dibanding dengan model classifier lainnya, yaitu Naïve Bayes Classifier memiliki tingkat akurasi yg lebih baik dibanding model classifier lainnya (Xhemali 2009).

2.2 Teorema Naive Bayes

Teorema Bayes yang menjadi dasar dari metoda tersebut. Pada Teorema Bayes, bila terdapat dua kejadian yang terpisah (misalkan X dan H), maka Teorema Bayes dirumuskan sebagai berikut (Bustami 2013).

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

X : Data dengan class yang belum diketahui

H : Hipotesis data merupakan suatu class spesifik

$P(H|X)$: Probabilitas hipotesis H berdasar kondisi X (posteriori probabilitas)

$P(H)$: Probabilitas hipotesis H (prior probabilitas)

$P(X|H)$: Probabilitas X berdasarkan kondisi pada hipotesis H

$P(X)$: Probabilitas X

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

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

Keterangan

- $i : 1, 2, 3, \dots, n$ jumlah data Hipotesis (*prior probabilitas*)
- dimana : $H_1 \cup H_2 \cup H_3 \dots \cup H_n = S$
- S : Probabilitas total H

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:

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

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

Keterangan

- $i : 1, 2, 3, \dots, n$ jumlah data Hipotesis (*prior probabilitas*)
- dimana : $H_1 \cup H_2 \cup H_3 \dots \cup H_n = S$

- **S** : Probabilitas total **H**

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:

$$P(C|F_1, \dots, F_n) = \frac{P(F_1, \dots, F_n|C)}{P(F_1, \dots, F_n)} \cdot P(C)$$

Di mana Variabel **C** merepresentasikan kelas, sementara variabel $F_1 \dots F_n$ 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:

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

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, F_1, \dots, F_n) menggunakan aturan perkalian sebagai berikut:

$$\begin{aligned} P(C|F_1, \dots, F_n) &= P(C) \cdot P(F_1, \dots, F_n|C) \\ &= P(C) \cdot P(F_1|C) \cdot P(F_2, \dots, F_n|C, F_1) \\ &= P(C) \cdot P(F_1|C) \cdot P(F_2|C, F_1) \cdot P(F_3, \dots, F_n|C, F_1, F_2) \\ &= P(C) \cdot P(F_1|C) \cdot P(F_2|C, F_1) \cdot P(F_3|C, F_1, F_2) \dots P(F_n|C, F_1, F_2, F_3, \dots, F_{n-1}) \end{aligned}$$

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 (F_1, F_2, \dots, F_n) saling bebas (*independen*) satu sama lain. Dengan asumsi tersebut, maka berlaku suatu kesamaan sebagai berikut:

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

Untuk $i \neq j$, sehingga

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

Atau dapat dituliskan dalam notasi

$$P(c|F_1, F_2, F_3, \dots, F_n) = P(C) \prod_{i=1}^n P(F_i|C)$$

yang dapat dijabarkan sebagai berikut

$$P(c|F) = P(f_1|c) \cdot P(f_2|c) \cdot P(f_3|c) \dots P(f_n|c) \cdot P(c)$$

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 :

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

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

2.3 XAMPP

XAMPP adalah *software* aplikasi pengembang yang digunakan untuk pengembangan *website* berbasis PHP dan juga sebagai server untuk local dalam pembuatan database dengan MySql.

XAMPP memiliki kelebihan untuk bisa berperan sebagai server web Apache dalam melakukan simulasi pengembangan web. Tool pengembangan web ini mendukung teknologi web populer seperti PHP, MySql dan Perl. (Tumanggor Ronaldi. 2015)



Gambar 2.1 Logo Xampp

2.4 PhpMyadmin

PhpMyadmin adalah sebuah aplikasi open source yang berfungsi untuk memudahkan manajemen MySQL. Dengan menggunakan software ini dapat dilakukan pembuatan database, dengan melakukan fungsi – fungsi membuat tabel, insert, hapus dan update data. Dengan GUI (*Graphical User Interface*) terasa lebih mudah, tanpa perlu mengetikkan perintah SQL secara manual. Phpmyadmin dapat di download di www.phpmyadmin.net. (Risnandar, 2015)



Gambar 2.1 Logo PhpMyAdmin

2.5 PHP

PHP berasal dari kata “*Hypertext Preprocessor*”, yaitu bahasa universal untuk penanganan pembuatan dan pengembangan sebuah situs web dan bisa digunakan bersamaan dengan HTML.

PHP sebagai sekumpulan skrip atau bahasa program memiliki fungsi utama yaitu mampu mengumpulkan dan mengevaluasi hasil survei atau bentuk apapun ke server database dan tahap selanjutnya akan menciptakan efek beruntun. Efek beruntun ini merupakan tindakan dari skrip lain yang akan melakukan komunikasi dengan database, mengumpulkan dan mengelompokkan informasi, kemudian menampilkannya pada saat ada tamu website memerlukannya menampilkan informasi sesuai permintaan user.

2.6 Pengenalan PHP

Sebuah halaman web yang ditulis menggunakan bahasa pemrograman PHP adalah sebagai berikut.

```
<?php
echo “Hallo dunia”;
?>
```

Penulisan dan pemanggilan fungsi dalam bahasa pemrograman PHP adalah sebagai berikut.

```
<?php
function penjumlahan($i1,$i2)
{
    return $i1+$i2;
}
echo penjumlahan(1,2);
#output : 3
?>
```

Di dalam bahasa pemrograman PHP terdapat 3 jenis sintaks sebagai komentar pada kode yaitu tanda blok `/* */` komentar 2 baris, `//` serta tanda pagar `#` digunakan untuk komentar satu baris. Komentar bertujuan untuk meninggalkan catatan pada kode PHP dan tidak akan diterjemahkan ke program. Contoh penulisan komentar dalam program PHP adalah sebagai berikut.


```
<?php
/*
Di sini dapat ditulis komentar,
Dapat digunakan untuk komentar banyak baris
*/
//digunakan untuk komentar satu baris
# digunakan untuk komentar satu baris
?>
```

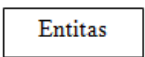

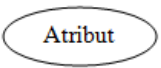

2.7 Basis Data

Basis data (database) adalah kumpulan data yang disimpan secara sistematis di dalam komputer yang dapat diolah atau dimanipulasi menggunakan perangkat lunak (program aplikasi) untuk menghasilkan informasi. Pendefinisian basis data meliputi spesifikasi berupa tipe data, struktur data dan juga batasan-batasan pada data yang akan disimpan. Basis data merupakan aspek yang sangat penting dalam sistem informasi karena berfungsi sebagai gudang penyimpanan data yang akan diolah lebih lanjut. Basis data menjadi penting karena dapat mengorganisasi data, menghindari duplikasi data, menghindari hubungan antar data yang tidak jelas dan juga update yang rumit

Menurut Pakereng & Wahyono (2004), basis data merupakan kumpulan data yang dipakai ada dalam suatu lingkup tertentu, misalkan instansi, perusahaan, dan lain-lain atau kasus tertentu. Menurut Pakereng & Wahyono (2004), sebuah konsep database memiliki beberapa hal sebagai berikut.

1. Entitas

Entitas merupakan tempat informasi direkam, dapat berupa orang, tempat, kejadian dan lain-lain. Sebagai contoh dalam kasus Administrasi Siswa misalnya, maka terdapat entity siswa, matakuliah, guru, pembayaran.

Notasi	Keterangan
	Entitas adalah suatu objek yang dapat diidentifikasi dalam lingkungan pemakai.
	Relasi menunjukkan adanya hubungan di antara sejumlah entitas yang berbeda.
	Atribut berfungsi mendeskripsikan karakter entitas (atribut yang berfungsi sebagai <i>key</i> diberi garis bawah).
	Garis sebagai penghubung antara relasi dan entitas atau relasi dan entitas dengan atribut.

Gambar 2.3 Penjelasan simbol entitas

2. Atribut

Atribut dapat juga disebut sebagai data elemen, data field, atau data item yang digunakan untuk menerangkan suatu entitas dan mempunyai harga tertentu, misalnya atribut dari entitas siswa diterangkan oleh nama, tanggal lahir, alamat.

3. Data value

Data value merupakan suatu informasi atau data aktual yang disimpan pada tiap data, elemen, atau atribut. Atribut nama pegawai menunjukkan tempat dimana informasi nama karyawan disimpan, nilai datanya misalnya adalah Anjang, Arif, Suryo, dan lain-lain yang merupakan isi data nama pegawai tersebut.

4. File/Table

Merupakan kumpulan record sejenis yang mempunyai panjang elemen yang sama, atribut yang sama, namun berbeda nilai datanya.

5. Record/Tuple

Merupakan kumpulan elemen-elemen yang saling berkaitan menginformasikan tentang suatu entitas secara lengkap. Satu record mewakili satu data atau informasi.

2.8 Website

Secara terminologi *website* adalah kumpulan dari halaman-halaman situs, yang biasanya terangkum dalam sebuah domain atau subdomain, yang tempatnya berada di dalam *World Wide Web* (WWW) di Internet. WWW terdiri dari seluruh situs web yang tersedia kepada publik. Halaman-halaman sebuah situs web (*web page*) diakses dari sebuah URL yang menjadi “akar” (*root*), yang disebut homepage

(halaman induk; sering diterjemahkan menjadi “beranda”, “halaman muka”), URL ini mengatur *web page* untuk menjadi sebuah hirarki, meskipun hyperlink-hyperlink yang ada di halaman tersebut mengatur para pembaca dan memberitahu mereka susunan keseluruhan dan bagaimana arus informasi ini berjalan.

2.9 HTML(*Hyper Text Markup Language*)

HTML adalah singkatan dari *HyperText Markup Language*. HTML merupakan file teks yang ditulis menggunakan aturan-aturan kode tertentu untuk kemudian disajikan ke user melalui suatu aplikasi *web browser*. Setiap informasi yang tampil di web selalu dibuat menggunakan kode HTML. Oleh karena itu, dokumen HTML sering disebut juga sebagai *web page* (halaman web). Untuk membuat dokumen HTML, kita tidak tergantung pada aplikasi tertentu, karena dokumen HTML dapat dibuat menggunakan aplikasi Text Editor apapun, bisa Notepad (untuk lingkungan MS Windows), Emacs atau Vi Editor (untuk lingkungan Linux), dan sebagainya. (Raharjo, 2016).

Penjelasan-penjelasan lengkap tentang HTML akan di bahas pada sub bab berikut.

2.9.1 Definnisi HTML(*Hyper Text Markup Language*)

HTML adalah singkatan dari *HyperText Markup Language*. HTML merupakan file teks yang ditulis menggunakan aturan-aturan kode tertentu untuk kemudian disajikan ke user melalui suatu aplikasi *web browser*. Setiap informasi yang tampil di web selalu dibuat menggunakan kode HTML.

2.9.2 Pengenalan dasar HTML (*Hyper Text Markup Language*)

Di dalam *script* HTML terdapat *tag*, elemen, dan atribut. Dokumen HTML disimpan dalam format teks reguler dan mengandung tag-tag yang memerintahkan web browser untuk mengeksekusi perintah-perintah yang dispesifikasikan.

1. Tag HTML

Tag adalah teks khusus (markup) berupa dua karakter “” (tanpa tanda kutip) yang merupakan cara untuk memberitahu *web browser* bagaimana suatu teks ditampilkan.

2. Elemen HTML

Menurut Rahardi (2014) Elemen adalah isi dari *tag* yang berada di antara *tag* pembuka dan *tag* penutup. Elemen terdiri atas tiga bagian, yaitu tag pembuka, isi, dan *tag* penutup. Dalam penulisan elemen HTML harus diperhatikan jangan sampai saling tumpang tindih .seperti pada Gambar 2.4 dan penulisan elemen HTML yang benar (tidak tumpang tindih) terdapat pada Gambar 2.5.



Gambar 2.4 Penulisan Elemen HTML yang Tumpang Tindih.



Gambar 2.5 Penulisan Elemen HTML yang Normal.

Pada Gambar 2.1 disebut penulisan elemen HTML yang tumpang tindih karena barisan elemen yang pertama, pasangan akhirnya adalah. penulisan *script* yang seperti itu akan mengakibatkan *script* tidak berfungsi.

3. Atribut HTML

Atribut adalah informasi tambahan yang diberikan kepada tag. Informasi ini bisa berupa instruksi untuk warna dari *text*, besar huruf dari *text*, dan lain sebagainya. Setiap atribut juga memiliki pasangan nama dan nilai (*value*), dan ditulis dengan *name="value"*. Value diapit tanda kutip, boleh tanda kutip satu (') atau dua (").

Penulisan atribut harus berada di antara tag pembuka dan penutup, contoh untuk membuat halaman *web* menjadi warna hitam dan tulisan menjadi warna kuning adalah sebagai berikut.

```
<html>
<head>
<title>Judul Dokumen HTML</title>
</head>
<body>
<body bgcolor="black" text="yellow">
  Isi Dokumen (Belajar HTML)
</body>
</html>
```

2.10 MySql

MySQL merupakan sebuah perangkat lunak atau software sistem manajemen basis data SQL atau DBMS Multithread dan multi user. MySQL sebenarnya merupakan turunan dari salah satu konsep utama dalam database untuk pemilihan atau seleksi dan pemasukan data yang memungkinkan pengoperasian data dikerjakan secara mudah dan otomatis. MySQL diciptakan oleh Michael "Monty" Widenius pada tahun 1979, seorang programmer komputer asal Swedia yang mengembangkan sebuah sistem database sederhana yang dinamakan UNIREG yang menggunakan koneksi low-level ISAM database engine dengan indexing.

a. Kelebihan MySQL

Adapun kelebihan MySQL dalam penggunaannya dalam database adalah:

- Free atau gratis sehingga MySQL dapat dengan mudah untuk mendapatkannya.
- MySQL stabil dan tangguh dalam pengoperasiannya
- MySQL mempunyai sistem keamanan yang cukup baik
- Sangat mendukung transaksi dan mempunyai banyak dukungan dari komunitas
- Sangat fleksibel dengan berbagai macam program
- Perkembangan dari MySQL sangat cepat

b. Kelemahan MySQL

Selain kelebihan yang disampaikan diatas, ada beberapa kekurangan yang dimiliki oleh mySQL, diantaranya:

- Kurang mendukung koneksi bahasa pemrograman seperti Visual basic atau biasa kita kenal dengan sebutan VB, Foxpro, Delphi dan lain-lain sebab koneksi ini menyebabkan field yang dibaca harus sesuai dengan koneksi dari bahasa pemrograman visual tersebut.

Data yang dapat ditangani belum besar dan belum mendukung widening function.

2.11 UML(*Unfied Modeling Language*)

Menurut Nugroho (2009) UML (Unifeid Modeling Language) adalah bahasa pemodelan untuk sistem atau perangkat lunak yang berparadigma (berorientasi objek). Pemodelan (modeling) sesungguhnya digunakan untuk penyederhanaan permasalahan-permasalahan yang kompleks sedemikian rupa sehingga lebih mudah dipelajari dan dipahami.

UML bukan hanya sekadar diagram, tetapi juga menceritakan konteksnya. Ketika pelanggan memesan sesuatu dari sistem, bagaimana transaksinya? bagaimana sistem mengatasi error yang terjadi? bagaimana keamanan terhadap sistem yang kita buat? dan sebagainya dapat dijawab dengan UML. Beberapa diagram dalam UML yaitu :

1. Diagram Kelas (Class Diagram), bersifat statis. Diagram ini memperlihatkan himpunan kelas-kelas, antarmuka - antarmuka, kolaborasi-kolaborasi, serta relasi-relasi. Diagram ini umum dijumpai pada pemodelan sistem berorientasi objek. Meskipun bersifat statis, sering pula diagram kelas memuat kelas-kelas aktif.
2. Diagram Paket (Package Diagram), bersifat statis. Diagram ini memperlihatkan kumpulan kelas-kelas, merupakan bagian dari diagram komponen.
3. Diagram Use Case, bersifat statis. Diagram ini memperlihatkan himpunan use case dan aktor-aktor (suatu jenis khusus dari kelas). Diagram ini terutama sangat penting untuk mengorganisasi dan memodelkan perilaku suatu sistem yang dibutuhkan serta diharapkan pengguna.

4. Diagram Interaksi dan Sequence (Urutan), bersifat dinamis. Dinamis urutan adalah diagram interaksi yang menekankan pada pengiriman pesan dalam suatu waktu tertentu.
5. Diagram Komunikasi (*Communication Diagram*), bersifat dinamis. Diagram sebagai pengganti diagram kolaborasi UML 1.4 yang menekankan organisasi struktural dari objek-objek yang menerima serta mengirim pesan.
6. Diagram Statechart (*Statechart Diagram*), bersifat dinamis. Diagram status memperlihatkan keadaan-keadaan pada sistem, memuat status (*state*), transisi, kejadian serta aktivitas. Diagram ini terutama penting untuk memperlihatkan sifat dinamis dari antarmuka (*inteface*), kelas, kolaborasi, dan terutama pada pemodelan sistem-sistem yang reaktif.
7. Diagram Aktivitas (*Activity Diagram*), bersifat dinamis. Diagram aktivitas adalah tipe khusus dari diagram status yang memperlihatkan aliran dari suatu aktivitas ke aktivitas lainnya dalam suatu sistem. Diagram ini terutama penting dalam suatu sistem serta pemodelan fungsi-fungsi suatu sistem dan memberi tekanan pada aliran kendali antarobjek.
8. Diagram Komponen (*Component Diagram*), bersifat statis. Diagram komponen ini memperlihatkan organisasi serta ketergantungan sistem/perangkat lunak pada komponen-komponen yang telah ada sebelumnya. Diagram ini berhubungan dengan diagram kelas dimana komponen secara tipikal dipetakan ke dalam satu atau lebih kelas-kelas, antarmuka-antarmuka serta kolaborasi-kolaborasi.
9. *Diagram Deployment (Deployment Diagram)*, bersifat statis. Diagram ini memperlihatkan konfigurasi saat aplikasi dijalankan (*run-time*). Memuat simpul-simpul beserta komponen-komponen yang ada di dalamnya. Diagram deployment berhubungan erat dengan diagram komponen dimana diagram ini memuat satu atau lebih komponen-komponen. Diagram ini sangat berguna saat aplikasi kita berlaku sebagai aplikasi yang dijalankan pada banyak mesin (*distributed computing*).

2.12 Laravel

Laravel adalah sebuah framework PHP yang dirilis dibawah lisensi MIT, dibangun dengan konsep MVC (model view controller). Laravel adalah

pengembangan website berbasis MVP yang ditulis dalam PHP yang dirancang untuk meningkatkan kualitas perangkat lunak dengan mengurangi biaya pengembangan awal dan biaya pemeliharaan, dan untuk meningkatkan pengalaman bekerja dengan aplikasi dengan menyediakan sintaks yang ekspresif, jelas dan menghemat waktu.

MVC adalah sebuah pendekatan perangkat lunak yang memisahkan aplikasi logika dari presentasi. MVC memisahkan aplikasi berdasarkan komponen-komponen aplikasi, seperti : manipulasi data, controller, dan user interface.

1. Model, Model mewakili struktur data. Biasanya model berisi fungsi-fungsi yang membantu seseorang dalam pengelolaan basis data seperti memasukkan data ke basis data, pembaruan data dan lain-lain.
2. View, View adalah bagian yang mengatur tampilan ke pengguna. Bisa dikatakan berupa halaman web.
3. Controller, Controller merupakan bagian yang menjembatani model dan view.

Beberapa fitur yang terdapat di Laravel :

- Bundles, yaitu sebuah fitur dengan sistem pengemasan modular dan tersedia beragam di aplikasi.
- Eloquent ORM, merupakan penerapan PHP lanjutan menyediakan metode internal dari pola “active record” yang mengatasi masalah pada hubungan objek database.
- Application Logic, merupakan bagian dari aplikasi, menggunakan controller atau bagian Route.
- Reverse Routing, mendefinisikan relasi atau hubungan antara Link dan Route.
- Restful controllers, memisahkan logika dalam melayani HTTP GET and POST.
- Class Auto Loading, menyediakan loading otomatis untuk class PHP.
- View Composer, adalah kode unit logikal yang dapat dieksekusi ketika view sedang loading.
- IoC Container, memungkinkan obyek baru dihasilkan dengan pembalikan controller.
- Migration, menyediakan sistem kontrol untuk skema database.
- Unit Testing, banyak tes untuk mendeteksi dan mencegah regresi.
- Automatic Pagination, menyederhanakan tugas dari penerapan halaman.

Menurut Hanry HAM, S.Kom., M.Eng pengembangan Web Menggunakan framework Laravel mempunyai kelebihan sebagai berikut:

- Waktu yang dibutuhkan untuk mengembangkan projek website dengan menggunakan *framework* ini menjadi lebih cepat.
- Dapat meningkatkan pengunjung webiste karena teknologi *framework* ini dapat digunakan di segala *browser* dan berbagai perangkat dengan baik.

Laravel dilengkapi dengan utilitas pemrograman untuk membantu proses pengembangan aplikasi web dan juga moderasi dengan cara terbaik. Ini dikemas dengan Modular Packaging System (MPS) dengan pengaturan ketergantungan yang lengkap.

2.13 Composer

Composer adalah *Dependecy Manager* yang ada di dalam pemrograman PHP. Dependensi adalah bergantung, atau bisa di sebut banyak *file* yang bergantung dalam *file – file* yang lain, dimana satu program bergantung pada program yang lainnya (Arif. 2018).

Composer merupakan utilitas yang dapat sangat membantu kita untuk mengelola proyek terhadap sebuah *library* atau *helper / plugin*. Masa depan akan berlaku seperti ini, dimana tidak mendownload direct dari situs resminya, melainkan menggunakan composer, pernah pakai linux, tentu tidak kesulitan menggunakan composer (Febiyan. 2014).

2.14 Black Box Testing

Black Box Testing Seperti yang dikutip oleh Sinaga (2015) dalam jurnalnya menurut Efendi (2011) menjelaskan bahwa “Pengujian Black Box adalah pengujian yang dilakukan terhadap interface tertentu untuk menguji bahwa fungsi-fungsi interface tersebut bekerja dengan baik”. Black-Box Testing bukanlah solusi alternatif dari White-Box Testing tapi lebih merupakan pelengkap untuk menguji hal-hal yang tidak dicakup oleh WhiteBox Testing (Ariani, 2009). Black-Box Testing cenderung untuk menemukan hal-hal berikut (Ariani, 2009).

1. Fungsi yang tidak benar atau tidak ada.
2. Kesalahan antarmuka (interface errors).
3. Kesalahan pada struktur data dan akses basis data.
4. Kesalahan performansi (performance errors).

5. Kesalahan inisialisasi dan terminasi.

Pengujian didesain untuk menjawab pertanyaan-pertanyaan berikut:

1. Bagaimana fungsi-fungsi diuji agar dapat dinyatakan valid?
2. Input seperti apa yang dapat menjadi bahan kasus uji yang baik?
3. Apakah sistem sensitif pada input-input tertentu?
4. Bagaimana sekumpulan data dapat diisolasi?
5. Berapa banyak rata-rata data dan jumlah data yang dapat ditangani sistem?

Efek apa yang dapat membuat kombinasi data ditangani spesifik pada operasi sistem?

2.15 Bootstrap

Bootstrap merupakan framework untuk membangun desain web secara responsif. Artinya, tampilan web yang dibuat oleh bootstrap akan menyesuaikan ukuran layar dari browser yang kita gunakan baik di desktop, tablet ataupun mobile *device*. Fitur ini bisa diaktifkan ataupun dinon-aktifkan sesuai dengan keinginan kita sendiri. Sehingga, kita bisa membuat web untuk tampilan desktop saja dan apabila dirender oleh mobile browser maka tampilan dari web yang kita buat tidak bisa beradaptasi sesuai layar. Dengan bootstrap kita juga bisa membangun web dinamis ataupun statis (Fadul, 2018).

2.16 Google Chrome

Google Chrome adalah sebuah aplikasi peramban yang digunakan untuk menjelajah dunia maya seperti halnya Firefox, Opera ataupun Microsoft Edge. Jika Firefox dikembangkan oleh Mozilla, Google Chrome dibuat dan dirancang oleh Google, perusahaan internet terbesar di dunia yang juga mempunyai Android. Proyek *open source* yang digunakan oleh Google disebut Chromium, menggunakan mesin *rendering* Webkit sampai dengan versi 27 dan dirancang untuk bekerja dengan kecepatan di atas rata-rata namun tetap ringan dijalankan di perangkat desktop dan mobile (Winarso. 2016).

2.17 Javascript

Javascript adalah sebuah bahasa pemrograman web yang bersifat *Client Side*. *Client Side Programming Language* adalah tipe bahasa pemrograman yang

permrosesannya dilakukan oleh client yang dimaksud untuk merujuk kepada web browser seperti Google Chrome, Mozilla dan lain sebagainya.

Javascript adalah Bahasa pemrograman *Client Side* berbeda dengan bahasa pemrograman *Server Side* seperti PHP, C, Java dan bahasa *back end* lainnya yang seluruh kodenya akan berjalan pada sisi *server* dan tidak dapat muncul pada layar browser (*Inspect Element*).

Untuk menjalankan Javascript hanya membutuhkan aplikasi *text editor* sebagai contoh sublime, atom, dreamweaver dan web browser. Javascript memiliki fitur high level *programming language, client side* dan berorientasi objek (Rendra H., 2018).

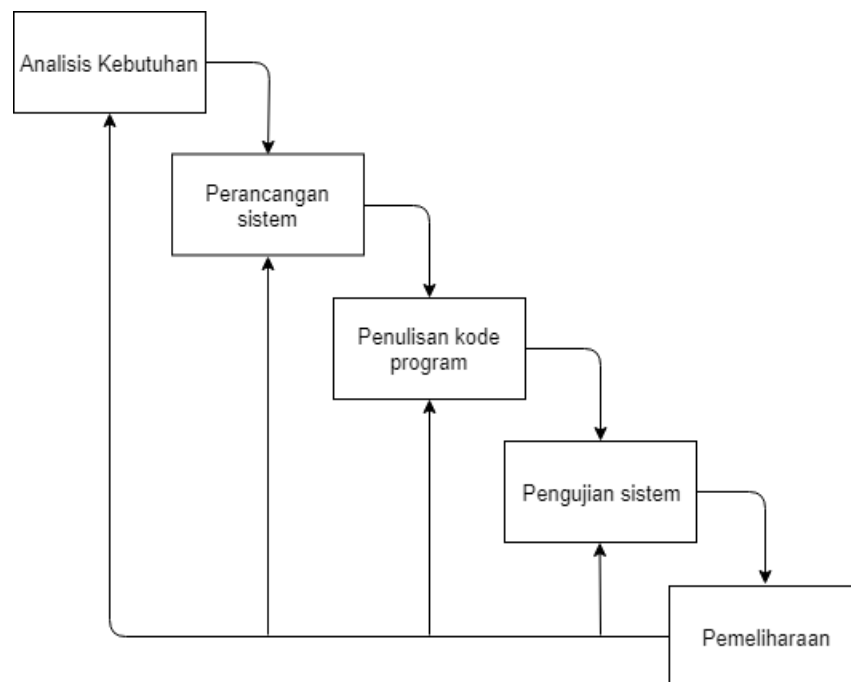
BAB III

METODOLOGI PENELITIAN

3.1 Metodologi Penelitian

Metode yang penulis gunakan dalam penelitian ini adalah metode *waterfall*. *Waterfall* apabila diartikan secara *literature* berarti air terjun. Namun demikian, bagi ilmu komputer dan juga teknologi informasi, *waterfall* merupakan salah satu jenis metode yang digunakan dalam melakukan sebuah pengembangan sistem (Dini S.Kom.,).

Langkah – langkah pembuatan dapat ditunjukkan pada Gambar berikut:



Gambar 3.1 Desain Penelitian

1. Analisi Kebutuhan

Sistem yang digunakan untuk mengklasifikasikan divisi organisasi PANKER pada Sekolah Mengah Atas Negeri 1 Sukagumiwang masih belum terkomputerisasi secara khusus untuk mengkalsifikasikannya. Sistem yang diusulkan kali ini adalah sistem yang terkomputerisasi berbasis website, sehingga semakin mudah menggunakannya karena data yang dikelola dapat terstruktur dengan baik sehingga dapat mempermudah semua pihak yang mengelola klasifikasi pendivisian pada organisai tersebut.

2. Perancangan Sistem

Tahapan berikutnya adalah pembuatan desain dari sebuah sistem. Pada tahapan ini penulis menggunakan UML (*Unified Modelling Language*), *flowchart*, dan perancangan antarmuka (*mockup*) untuk memberikan gambaran mengenai alur sistem. Selain itu penulis menggunakan use case diagram yang digunakan selama proses analisa untuk menangkap requirement sistem dan untuk memahami bagaimana sistem bekerja

3. Penulisan Kode Program

Untuk menjalankan desain sistem yang sudah dibuat, maka desain sistem tersebut diterjemahkan ke dalam kode dan juga *script*, sehingga nantinya desain dari sistem tersebut bisa berjalan dengan lancar dan juga baik.

4. Pengujian Sistem

Pengujian sistem dilakukan dengan melakukan setiap proses untuk menguji fungsionalitas sistem dan mengamati kemungkinan kesalahan yang terjadi pada setiap proses. Pengujian ini dilakukan secara *blackbox* yaitu pengujian dilakukan dengan hanya memperhatikan masukan ke sistem dan keluaran sistem.

5. Pemeliharaan

Tahap akhir dari model *Waterfall* adalah **pemeliharaan (*Maintenance*)**. *Software* yang sudah jadi, dijalankan serta dilakukan pemeliharaan / *Maintenance*. Pemeliharaan termasuk dalam memperbaiki kesalahan-kesalahan yang tidak ditemukan pada langkah sebelumnya.

3.2 Metode Pengumpulan Data

3.2.1 Wawancara

Penulis melakukan wawancara khususnya beberapa sumber dari bagian Kesekretariatan, anggota divisi, dan ketua umum dari organisasi PANKER Sman 1 Sukagumiwang untuk mengetahui informasi-informasi serta data data yang akan di pakai dalam pemrosesa data pada naive bayes, data yang dikumpulkan sebagai berikut.

a) Data Training.

Data yang digunakan dalam penelitian yaitu data dari organisasi pecinta alam

Sekolah Menengah Atas Negeri 1 Sukagumiwang (PANKER) di Kecamatan Kertasemaya sebagai data training. Data tersebut diperoleh dari Dewan pengurus organisasi pecinta alam PAKER..

b) Data Testing.

Data testing merupakan data yang sedang atau akan terjadi dan digunakan untuk menguji pola yang telah didapatkan dari data training. Data testing yang digunakan adalah data calon pengurus yang akan menjadi dewan kerja organisasi PANNKER.

3.2.2 Observasi

Penulis melakukan pengamatan dari data – data yang didapatkan yang kemudian penulis menyimpulkan ke dalam proses yang terstruktur untuk di implementasikan ke dalam sistem aplikasi berbasis web.

3.2.3 Studi Pustaka

Penulis mencari bahan – bahan atau referensi yang dapat digunakan sebagai referensi penulis dalam pembuatan sistem aplikasi yang dapat digunakan oleh pengurus organisasi PANKER di Sekolah Menengah Atas Negeri 1 Sukagumiwang.

3.3 Analisis Kebutuhan Sistem

Dalam pembuatan aplikasi klasifikasi ini terdapat kebutuhan baik dalam bentuk *hardware* maupun *software*. Adapun kebutuhan-kebutuhan tersebut dijelaskan pada sub bab selanjutnya.

3.3.1 Kebutuhan Hardware

Adapun kebutuhan *hardware* untuk membuat klasifikasi ini terdapat pada tabel berikut,

Tabel 3.1 Kebutuhan *Hardware*

Jenis <i>Harware</i>	Kebutuhan <i>Hardware</i>
Processor	Intel Core i3
RAM	2 GB atau lebih
<i>Hardisk</i>	180 GB atau lebih

3.3.2 Kebutuhan Software

Adapun kebutuhan *software* untuk membuat klasifikasi pendivisian ini terdapat pada berikut,

Tabel 3.2 Kebutuhan *Software*

Jenis <i>Software</i>	Kebutuhan <i>Software</i>
Sistem Operasi	Windows 7 Profesional
Desain Aplikasi	CorelDraw X8
Bahasa <i>Scripting</i>	PHP 7, HTML, CSS, Javascript, JQuery, Framework Laravel
Software Pengolah	Visual Studio Code Text Editor, XAMPP, Navicat Premium
Web Browser	Google Chrome, Mozilla Firefox, Microsoft Edge, dan lain - lain

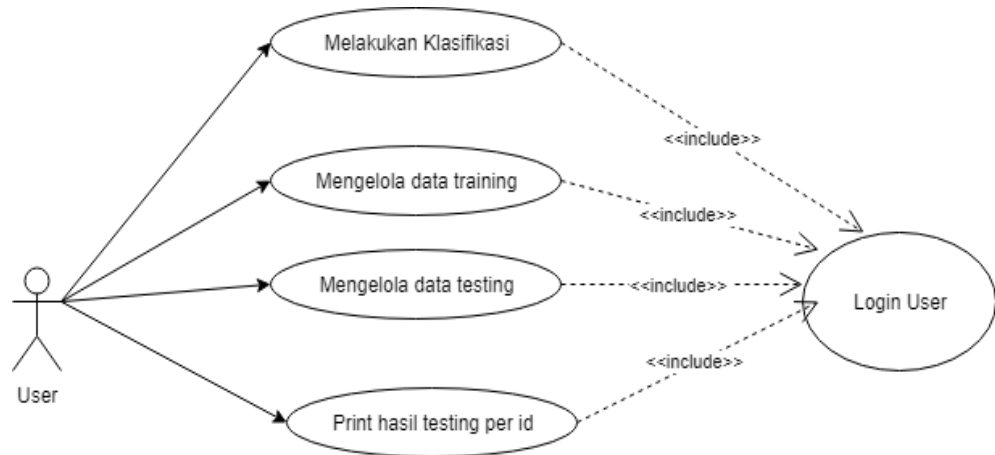
3.4 Perancangan Sistem

Perancangan sistem ini menggunakan UML (*Unified Modelling Language*). Pada perancangan ini akan digambarkan secara garis besar mengenai aplikasi Klasifikasi Pendivisian Organisasi Panker Menggunakan Metode Naive Bayes Menggunakan Framework Laravel yang akan dijelaskan pada sub bab selanjutnya.

3.4.1 Use Case Diagram

Pada use case diagram yang dibuat oleh penulis dalam pembuatan aplikasi klasifikasi pendivisian organisasi panker menggunakan metode naive bayes ini memiliki satu aktor yaitu user. Aktor user ini menginputkan data training yang akan di jadikan proses sampel untuk naive bayes, aktor user ini juga menginputkan data testing yang akan diproses sehingga menghasilkan output akhir yang menentukan rekomendasi calon pengurus masuk pada divisi yang direkomendasikan oleh sistem. User ini juga bisa melakukan print pdf dari hasil proses klasifikasi.

Berikut adalah use case diagram dari aplikasi Klasifikasi Pendivisian Organisasi Panker Menggunakan Metode Naive Bayes terdapat pada Gambar 3.2



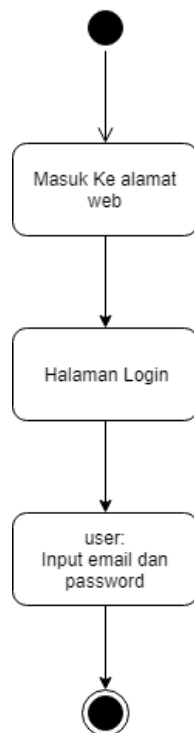
Gambar 3.2 Use Case

3.4.2 Activity Diagram

Activity diagram merupakan gambaran aktifitas dalam sistem yang sedang dirancang, bagaimana masing – masing alir dirancang dan bagaimana itu berakhir.

1. Activity Diagram Login User

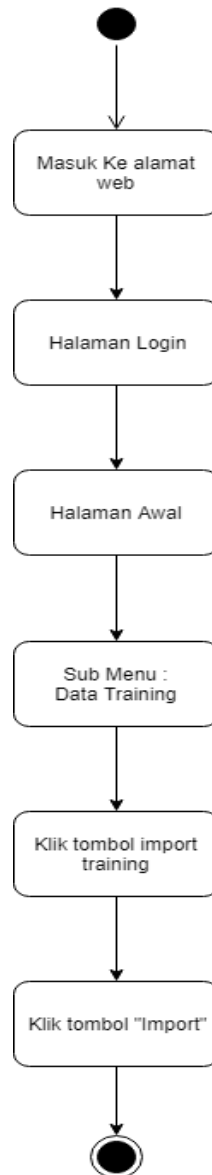
User harus melakukan login untuk dapat menggunakan fitur – fitur testing dan training yang ada pada sistem. Rancangan activity diagram login user terdapat pada Gambar 3.3.



Gambar 3.3 Activity Diagram Login User

2. Activity Diagram Import Data Training

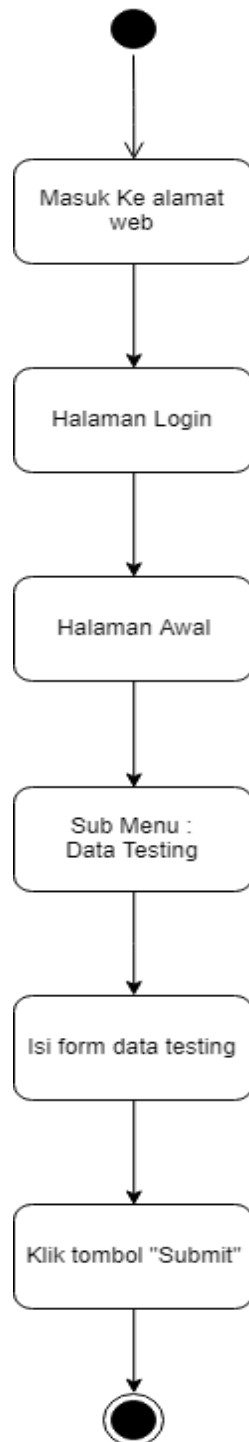
User Mengimport data – data training yang akan di olah oleh sistem untuk mencari probabilitas dari sistem klasifikasi sehingga mendapatkan hasil training dari pengolahan data testing oleh sistem. Rancangan activity diagram import data training terdapat pada Gambar 3.4.



Gambar 3.4 *Activy* Diagram Import Data Training

3. Activity Diagram Input Klasifikasi Testing

Mahasiswa yang telah memiliki akun dapat menggunakan fitur dalam sistem salah untuk pengajuan proposal tugas akhir dengan mengisi form pengajuan sidang serta mengunggah *file* proposal tugas akhir. Rancangan activity diagram pengajuan sidang tugas akhir terdapat pada Gambar 3.5.

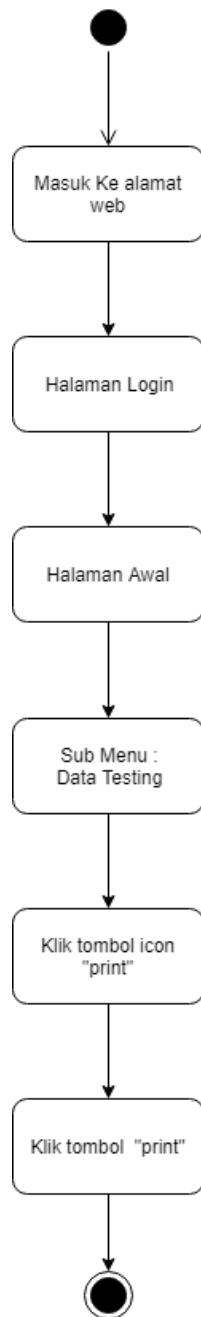


Gambar 3.5 Activity Diagram klasifikasi testing

4. Activity Diagram Export Pdf Klasifikasi

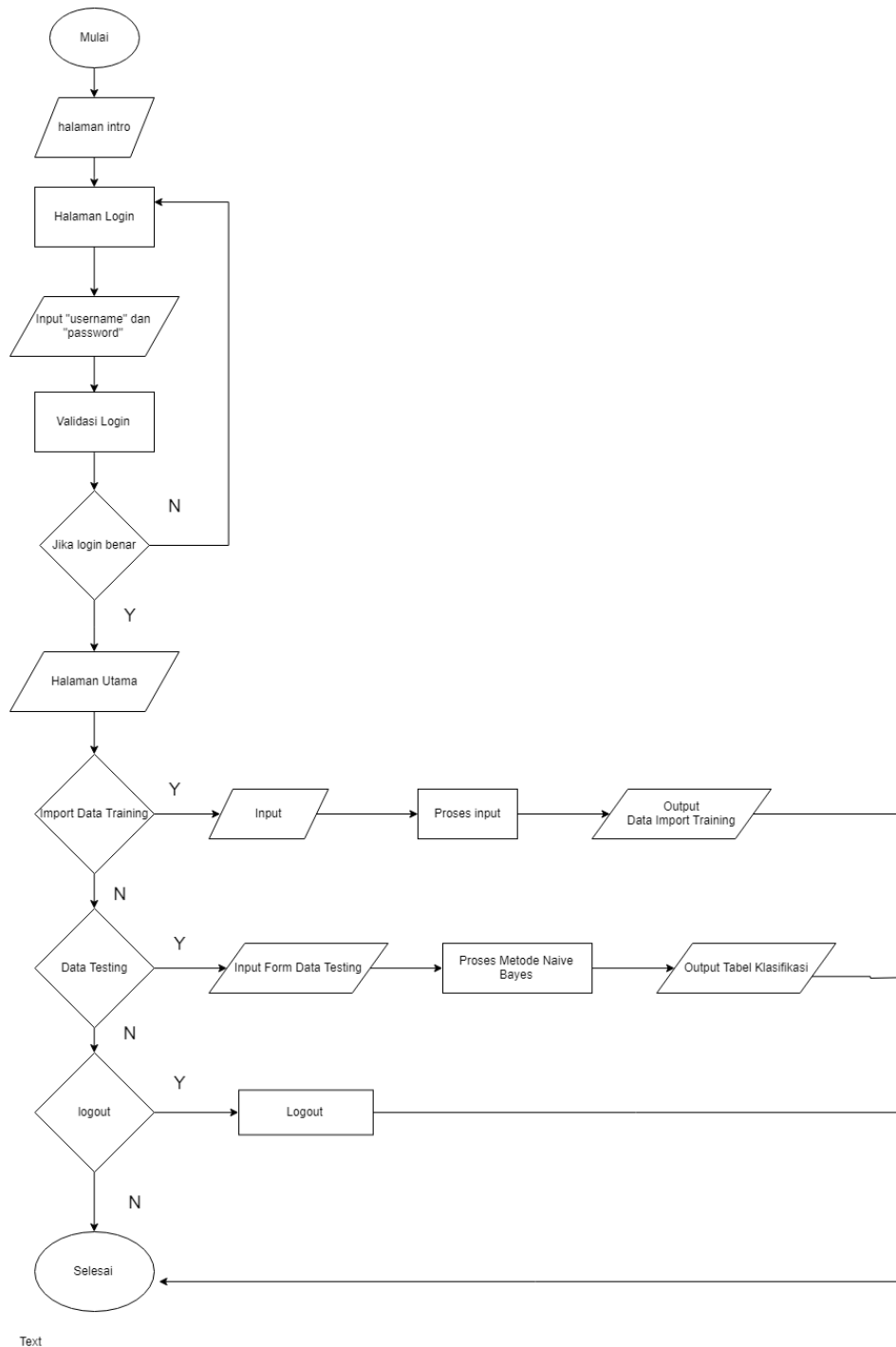
Setelah data melakukan klasifikasi data akan dimunculkan pada view tabel testing

dan export ke pdf untuk hasil klasifikasi per id activity diagram Export Pdf Klasifikasi terdapat pada Gambar 3.6.



Gambar 3.6 Activity Diagram *Export Pdf*

3.4.3 Flowcart Aplikasi



Gambar 3.7 Gambar *Flowcart* aplikasi

3.4.4 Perancangan Awal Antar Muka

Beberapa rancangan desain sistem antar muka dari aplikasi klasifikasi ini yang akan dijelaskan pada sub bab berikut.

A. Rancangan Halaman Awal

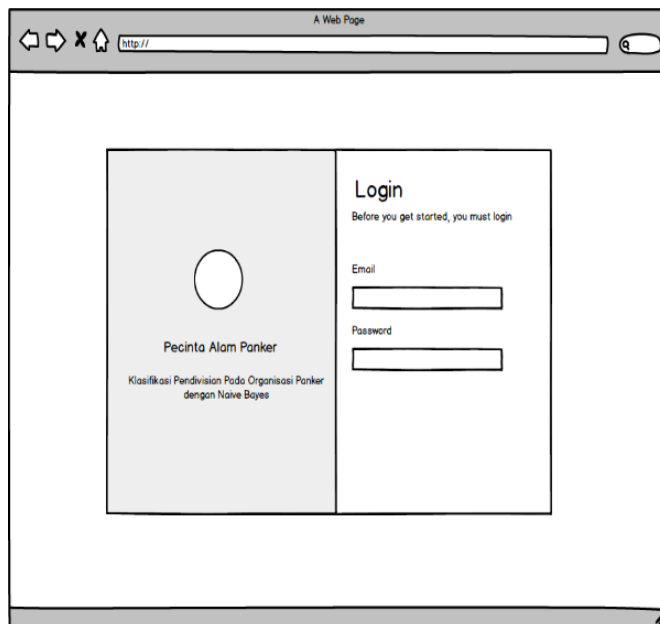
Halaman awal merupakan halaman yang pertama kali ditampilkan ketika *user* mengakses url. Adapun hasil rancangan halaman awal aplikasi terdapat pada Gambar 3.8.



Gambar 3.8 Rancangan Halaman Awal

B. Rancangan Halaman Login

Halaman login merupakan halaman yang digunakan ketika *user* akan melakukan login ke dalam aplikasi. Adapun hasil rancangan halaman login aplikasi terdapat pada Gambar 3.9.



Gambar 3.9 Rancangan Halaman Login

Gambar 3.11 Rancangan Halaman Tampil Data Testing

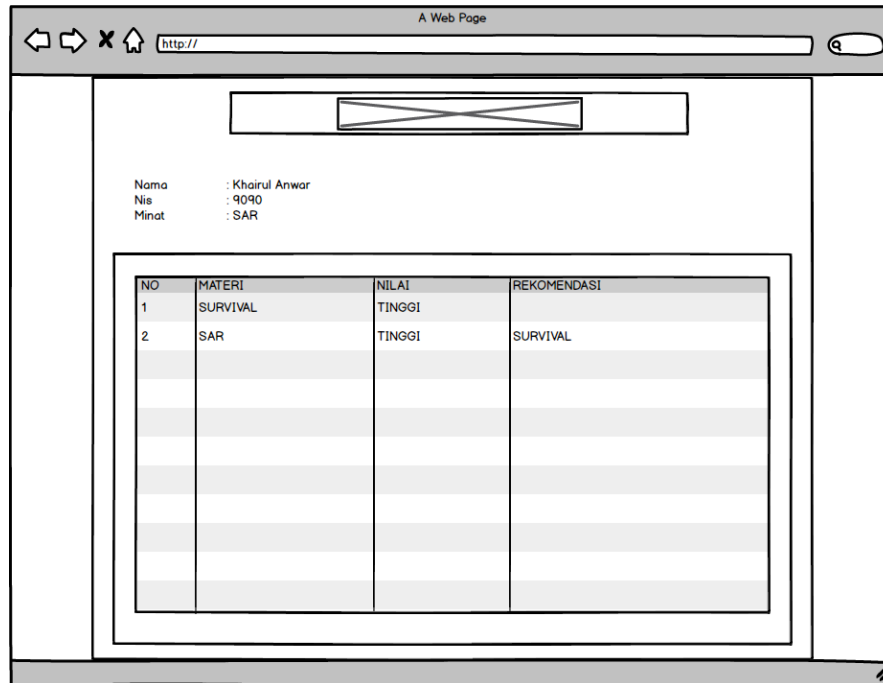
E. Rancangan View Data Tesing

Halaman view data testing merupakan suatu modal dimana user dapat melihat seluruh data dari hasil proses naive bayes secara per-id nya masing masing. Adapun rancangan halaman tambah data terdapat pada Gambar 3.12.

Gambar 3.12 Rancangan Halaman Tambah Data

F. Rancangan Halaman Tampil Export PDF

Halaman tampil export Pdf bertujuan untuk mencetak hasil dari masing masing data yang sudah di proses oleh naive bayes yang selanjutnya akan dijadikan untuk diterbitkan laporan oleh organisasi untuk dilihat hasilnya oleh calon pengurus. Adapun rancangan halaman tampil data pengajuan TA terdapat pada Gambar 3.13.



Gambar 3.13 Rancangan Halaman Tampil Data Export Pdf Per id

3.4.5 Perancangan Database

Berikut ini merupakan rancangan tabel – tabel dalam database aplikasi Klasifikasi Pendivisian Organisasi Panker dengan Metode Naive Bayes.

A. Tabel Users

Tabel ini digunakan untuk menyimpan user untuk login.dapat dilihat pada Tabel berikut.

Tabel 3.3 Tabel users

No.	Nama Kolom	Tipe Data	Keterangan
1	Id	bigint10)	Menyimpan id (PK)
2	Name	varchar(191)	Menyimpan nama
3	Email	varchar(191)	Menyimpan username

4	Password	varchar(191)	Menyimpan password
5	remember_token	varchar(100)	Menyimpan token
6	created_at	Timestamp	Menyimpan waktu <i>insert</i> data
7	updated_at	Timestamp	Menyimpan waktu <i>update</i> data

B. Tabel training

Tabel ini digunakan untuk menyimpan hasil import data dan mengambil data untuk diproses oleh naive bayes ,tabel data_dosen dapat dilihat pada tabel berikut.

Tabel 3.4 Tabel training

No	Nama Kolom	Tipe Data	Keterangan
1	Id	bigint (20)	Menyimpan id (PK)
2	Nis	int(11)	Menyimpan nis
3	Nama	varchar(50)	Menyimpan nama
4	nilai_gh	tinyint(4)	Menyimpan nilai gh
5	nilai_ppgd	tinyint(4)	Menyimpan nilai ppgd
6	nilai_sar	tinyint(4)	Menyimpan nilai sar
7	nilai_impk	tinyint(4)	Menyimpan nilai impk
8	nilai_repling	tinyint(4)	Menyimpan nilai repling
9	nilai_sebrang_kering	tinyint(4)	Menyimpan nilai sebrang kering
10	Minat	tinyint(4)	Menyimpan nilai minat
11	Divisi	tinyint(4)	Menyimpan nilai divisi
12	created_at	Timestamp	Menyimpan waktu <i>insert</i> data
13	updated_at	Timestamp	Menyimpan waktu <i>update</i> data

C. Tabel testing

Tabel ini digunakan untuk menyimpan data dari hasil proses naive bayes dan mengambil data testing untuk di export pdf. Daftar kolom – kolom tabel data_mahasiswa dapat dilihat pada tabel berikut.

Tabel 3.5 Tabel testing

No	Nama Kolom	Tipe Data	Keterangan
1	Id	bigint (20)	Menyimpan id (PK)
2	nis_test	int(11)	Menyimpan nis
3	nama_test	varchar(50)	Menyimpan nama
4	n_gh_test	tinyint(4)	Menyimpan nilai gh
5	n_ppgd_test	tinyint(4)	Menyimpan nilai ppgd
6	n_sar_test	tinyint(4)	Menyimpan nilai sar
7	n_impk_test	tinyint(4)	Menyimpan nilai impk
8	n_repling_test	tinyint(4)	Menyimpan nilai repling
9	n_sebrang_kering_test	tinyint(4)	Menyimpan nilai sebrang kering
10	minat_test	tinyint(4)	Menyimpan nilai minat
11	divisi_test	tinyint(4)	Menyimpan nilai divisi
12	created_at	Timestamp	Menyimpan waktu <i>insert</i> data
13	updated_at	Timestamp	Menyimpan waktu <i>update</i> data

BAB IV

HASIL DAN PEMBAHASAN

4.1 Hasil Penelitian

Hasil Penelitian ini menghasilkan sebuah aplikasi klasifikasi pendivisian organisasi panker menggunakan metode naive bayes. Aplikasi ini digunakan untuk memudahkan pengurus untuk menentukan rekomendasi bagi calon pengurus untuk mereka masuk ke divisi pada oraganisai panker untuk meminimalisirkan pengurus salah penempatan tanggung jawab pada divisi tersebut.

4.2 Hasil Pustaka

Hasil Studi pustaka ini penulis mencari informasi untuk bisa membuat aplikasi klasifikasi pendividsian organisasi panker menggunakan refresnsi dari internet dan jurnal penelitian.

4.3 Implemntasi Desain Antar Muka

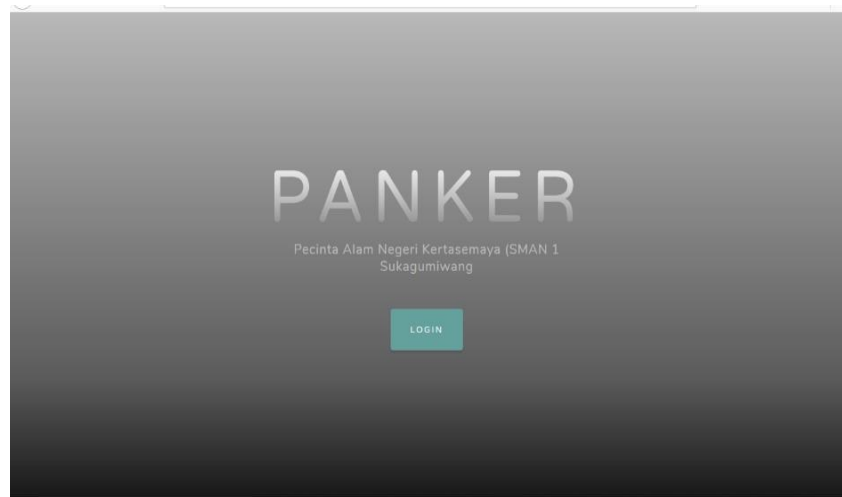
Tahap implemtasi merupakan tahap penerjemah perancangan berdasarkan hasil analisis kedalam suatu bahasa pemrograman yang digunakan serta penerapan perangkat lunak yang dibangun pada lingkungan yang sesungguhnya. Tujuan implemtasi untuk menerapkan perancangan yang telah dilakukan terhadap aplikasi, sehingga pengguna dapat memberi, memasukan, demi berkembangnya aplikasi yang telah dibangun. Adapun pembahasan implementasi desain berikut:

4.3.1 Implementasi Desain Antar Muka

Tahap implemtasi merupakan tahap penerjemah perancangan berdasarkan hasil analisis kedalam suatu bahasa pemrograman yang digunakan serta penerapan perangkat lunak yang dibangun pada lingkungan yang sesungguhnya. Tujuan implemtasi untuk menerapkan perancangan yang telah dilakukan terhadap aplikasi, sehingga pengguna dapat memberi, memasukan, demi berkembangnya aplikasi yang telah dibangun. Adapun pembahasan implementasi desain berikut:

4.3.2 Implementasi Hasil Tampilan Intro Aplikasi

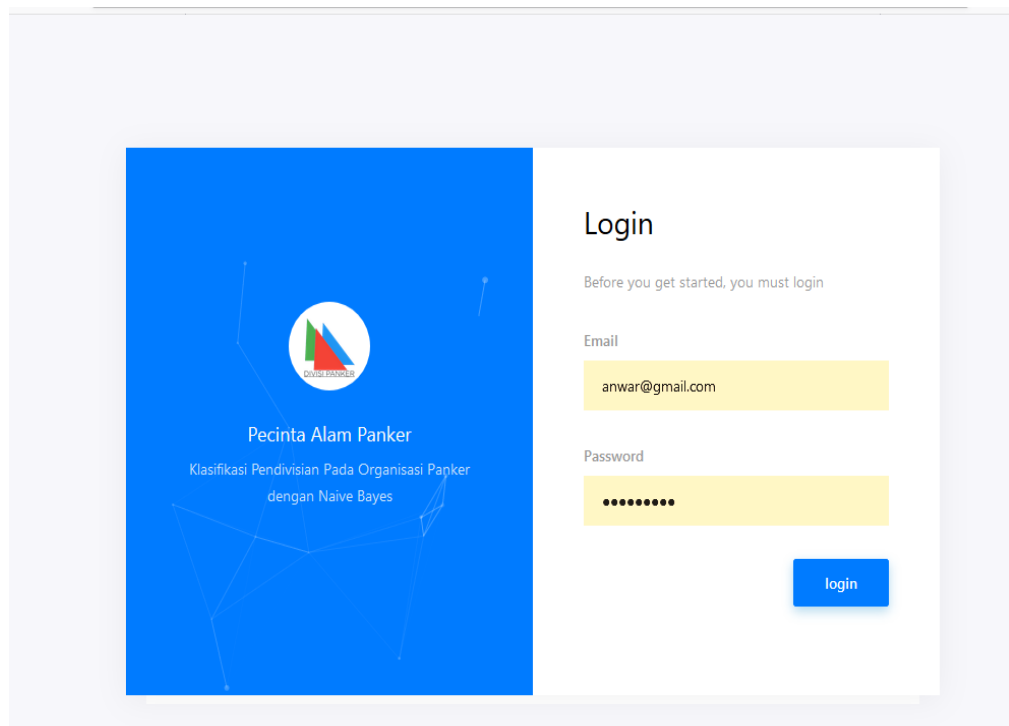
Halaman intro ini merupakan halaman awal sebelum menuju halaman login, halaman ini diakses ketika user mengunjungi url apliaksi tersebut.



Gambar 4.1 Tampilan Halaman Intro

4.3.3 Implementasi Hasil Tampilan Login

Halaman Login ini merupakan halaman ketika user akan masuk ke dalam aplikasi untuk melakukan semua aktifitas



Gambar 4.2 Tampilan Halaman Login

4.3.4 Implementasi Hasil Tampilan Data Training

Halaman Tampilan data Training merupakan halaman yang berisikan data-data training yang akan digunakan untuk pengolahan naive bayes.

Hover Data Table

Import Training

No	GH	PPGD	SAR	IMPK	Repling	SK	Minat	Divisi
1	sedang	sedang	sedang	tinggi	tinggi	tinggi	SURVIVAL	SURVIVAL
2	tinggi	tinggi	tinggi	tinggi	tinggi	tinggi	SURVIVAL	SURVIVAL
3	sedang	tinggi	rendah	tinggi	sedang	tinggi	SURVIVAL	SURVIVAL
4	rendah	rendah	sedang	sedang	tinggi	rendah	SURVIVAL	RC
5	rendah	rendah	rendah	rendah	rendah	sedang	SURVIVAL	SURVIVAL
6	rendah	sedang	rendah	tinggi	sedang	tinggi	SURVIVAL	SURVIVAL
7	rendah	rendah	sedang	sedang	tinggi	tinggi	SURVIVAL	SURVIVAL
8	tinggi	sedang	sedang	rendah	sedang	tinggi	SURVIVAL	SAR
9	sedang	sedang	rendah	sedang	sedang	sedang	SURVIVAL	SURVIVAL
10	sedang	sedang	tinggi	sedang	tinggi	tinggi	SURVIVAL	SURVIVAL
No	GH	PPGD	SAR	IMPK	Repling	SK	Minat	Divisi

Showing 1 to 10 of 90 entries

Previous 1 2 3 4 5 ... 9 Next

Gambar 4.3 Tampilan Tabel Training

4.3.5 Implementasi Hasil Tampilan Data Testing

Halaman Testing ini merupakan halaman kumpulan data-data hasil proses klasifikasi pendivisian menentukan rekomendasi calon pengurus.

DIVISI PANKER

Dashboard Version 2.0

Home - Dashboard

Menu

- Data Training
- Data Testing

DIVISI SURVIVAL [TRAINING] 30

DIVISI SAR [TRAINING] 34

DIVISI RC [TRAINING] 26

DIVISI SURVIVAL [TESTING] 1

DIVISI SAR [TESTING] 1

DIVISI RC [TESTING] 0

Horizontal Form

Nama

NIS

Nilai Gunung Hutan

Nilai PPGD

Nilai SAR

Nilai IMPK

Nilai Repling

Gambar 4.4 Tampilan Halaman Testing

4.3.6 Implementasi Form Testing Klasifikasi

Halaman form testing klasifikasi ini merupakan halaman untuk memasukan data klasifikasi yang selanjutnya datanya akan disimpan pada tabel data testing.

Horizontal Form

Nama

NIS

Nilai Gunung Hutan

Nilai PPGD

Nilai SAR

Nilai IMPK

Nilai Repling

Nilai Seebrang Kering

minat

Submit







Gambar 4.5 Tampilan Form Tetsing

4.3.7 Implementasi Hasil Tampilan Modal View Testing

Halaman Modal View Testing ini merupakan halaman rincian secara lengkap berisikan keterangan dan nilai hasil proses pendivisain calon pengurus.

Hover Data Table

Show entries Search:

No	nama	nis	[label] Divisi	action
1	khairul	9898	Survival	  
2	khairul anwar	221290	SAR	  
No	nama	nis	[label] Divisi	action

Showing 1 to 2 of 2 entries

Previous **1** Next

Gambar 4.6 Tampilan Tabel Testing

Gambar 4.7 Tampilan Modal View Testing

4.3.8 Implementasi Hasil Tampilan Export Pdf

Halaman Expot Pdf ini merupakan halaman untuk digunakan print out data untuk keperluan dokumentasi pengurus untuk pengumuman hasil rekomendasi proses klasifikasi pendivisian.

No	nama	nis	[label] Divisi	action
1	khairul	9898	Survival	
2	khairul anwar	221290	SAR	

Gambar 4.8 Tampilan Tabel Testing

NO	MATERI	NILAI	MINAT	REKOMENDASI
1	Gumung Hutan	tinggi	RC	SURVIVAL
2	PPGD	tinggi		
3	SAR	sedang		
4	Ilmu Medan Peta Kompas	sedang		
5	Repling	sedang		

Gambar 4.9 Tampilan Export Pdf

4.4 Implementasi Database

Database dalam pembuatan aplikasi klasifikasi pendivisian organisasi panker menggunakan naive bayes ini terdiri dari 3 tabel yaitu tabel users, tabel testing, dan yang terakhir adalah tabel training.

4.4.1 Tabel users

Tabel ini digunakan untuk menyimpan data akun user dan data tabel users ini usah ada dalam menggunakan fitur seeder laravel. Adapun dari pembuatan tabel ini bisa dilihat pada gambar dan keterangan berikut.

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra
1	id 🗝️	bigint(20)		UNSIGNED	Tidak	Tidak ada		AUTO_INCREMENT
2	name	varchar(50)	utf8mb4_unicode_ci		Tidak	Tidak ada		
3	email 📧	varchar(191)	utf8mb4_unicode_ci		Tidak	Tidak ada		
4	email_verified_at	timestamp			Ya	NULL		
5	password	varchar(191)	utf8mb4_unicode_ci		Tidak	Tidak ada		
6	remember_token	varchar(100)	utf8mb4_unicode_ci		Ya	NULL		
7	created_at	timestamp			Ya	NULL		
8	updated_at	timestamp			Ya	NULL		

Gambar 4.10 Gambar Tabel users

4.4.2 Tabel Training

Tabel ini digunakan untuk menyimpan data training dan data tabel training ini digunakan untuk sampel naive bayes. Adapun pembuatan tabel ini bisa dilihat pada gambar dan keterangan berikut.

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra
1	id 🗝️	bigint(20)		UNSIGNED	Tidak	Tidak ada		AUTO_INCREMENT
2	nilai_gh	tinyint(4)			Tidak	Tidak ada		
3	nilai_ppgd	tinyint(4)			Tidak	Tidak ada		
4	nilai_sar	tinyint(4)			Tidak	Tidak ada		
5	nilai_impk	tinyint(4)			Tidak	Tidak ada		
6	nilai_repling	tinyint(4)			Tidak	Tidak ada		
7	nilai_sebrang_kering	tinyint(4)			Tidak	Tidak ada		
8	minat	varchar(1)	utf8mb4_unicode_ci		Tidak	Tidak ada		
9	divisi	tinyint(4)			Tidak	Tidak ada		
10	created_at	timestamp			Ya	NULL		
11	updated_at	timestamp			Ya	NULL		

Gambar 4.11 Gambar Tabel training

4.4.3 Tabel Testing

Tabel ini digunakan untuk menyimpan data testing dari pengolahan naive bayes. Adapun pembuatan tabel ini bisa dilihat pada gambar dan keterangan berikut.

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra
1	id 	bigint(20)		UNSIGNED	Tidak	Tidak ada		AUTO_INCREMENT
2	nis_test	int(11)			Ya	NULL		
3	nama_test	varchar(50)	utf8mb4_unicode_ci		Ya	NULL		
4	n_gh_test	tinyint(4)			Tidak	Tidak ada		
5	n_ppgd_test	tinyint(4)			Tidak	Tidak ada		
6	n_sar_test	tinyint(4)			Tidak	Tidak ada		
7	n_impk_test	tinyint(4)			Tidak	Tidak ada		
8	n_repling_test	tinyint(4)			Tidak	Tidak ada		
9	n_sebrang_kering_test	tinyint(4)			Tidak	Tidak ada		
10	minat_test	varchar(1)	utf8mb4_unicode_ci		Tidak	Tidak ada		
11	divisi_test	tinyint(4)			Tidak	Tidak ada		
12	created_at	timestamp			Ya	NULL		
13	updated_at	timestamp			Ya	NULL		

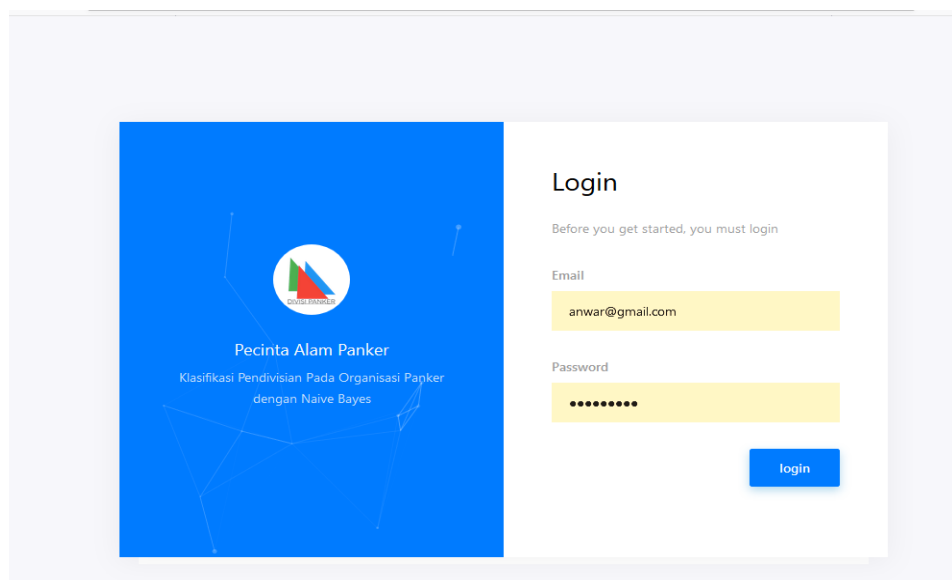
Gambar 4.12 Gambar Tabel testing

4.5 Tahap Pengujian Sistem

Tahap Pengujian sistem ini merupakan sebuah pengujian sistem aplikasi kemudian akan dijelaskan prosedur dan hasilnya sebagai berikut:

4.5.1 User Melakukan Login

Pengurus organisasi melakukan login mengisi username dan password yang sudah ada default aplikasinya, kemudian pengurus login sebagai user.



Pecinta Alam Panker
Klasifikasi Pendivisian Pada Organisasi Panker dengan Naive Bayes

Login

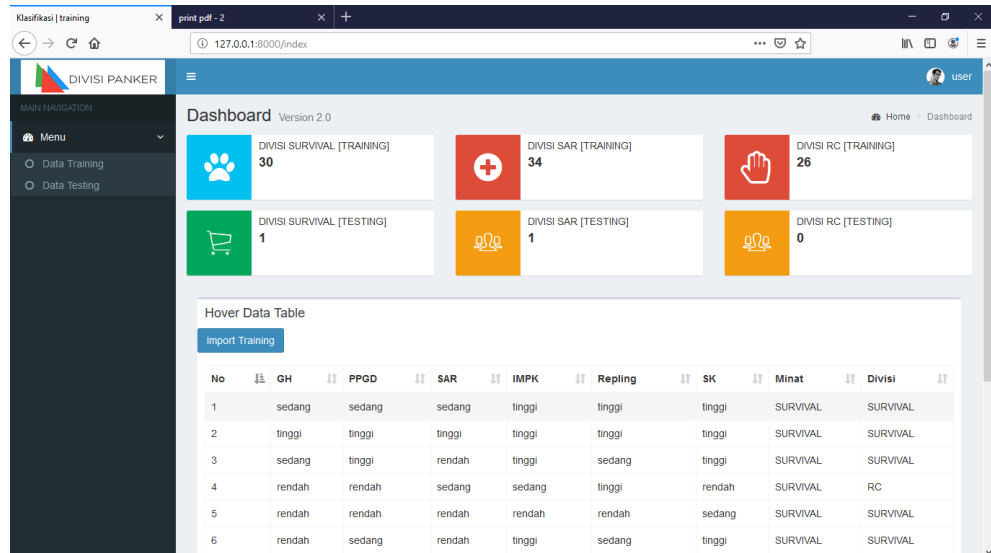
Before you get started, you must login

Email
anwar@gmail.com

Password
.....

login

Gambar 4.13 Tampilan Halaman Login



Gambar 4.14 Gambar Setelah Berhasil Melakukan Login

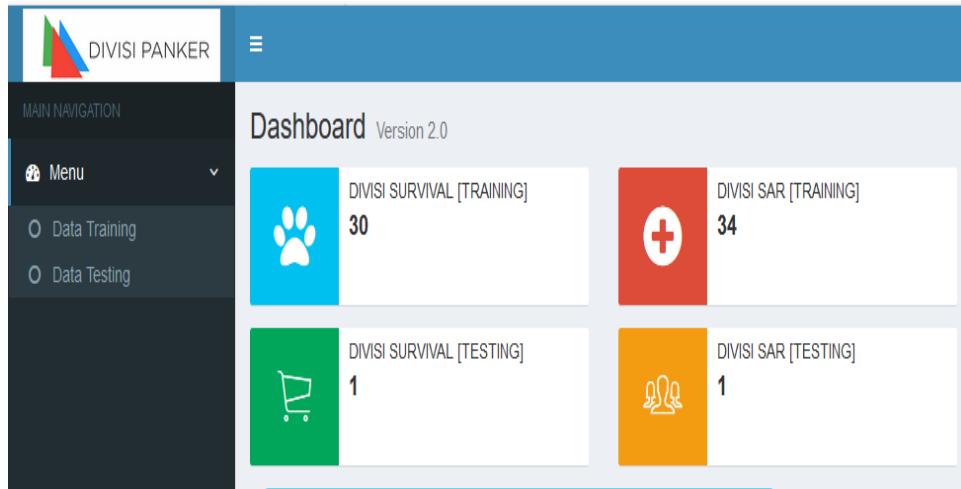
Jika user salah mengisi form *username* atau *password* maka akan user tidak bisa melakukan login akan di kembali ke halaman login untuk mengisi form login dengan benar dan akan muncul peringatan , bisa dilihat pada gambar berikut:

The image shows two versions of the login page. The top version is the standard login form with fields for 'Username' and 'Password', and a 'login' button. The bottom version shows the same form but with an error message 'username atau password salah!' displayed in a red box above the 'Username' field, indicating a failed login attempt.

Gambar 4.15 Ketika user salah memasukan *username* / *password*

4.5.2 User Memilih Menu Training

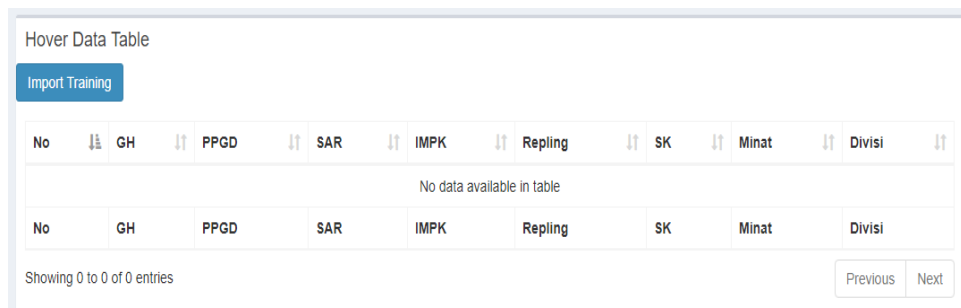
User memilih menu training untuk melihat data-data training hasil dari import data excel training. dikarenakan data banyak hanya ditampilkan beberapa tab.



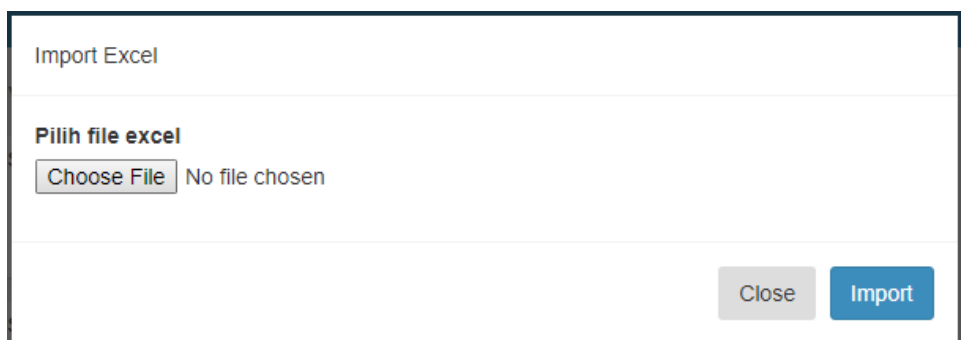
Gambar 4.16 Tampilan Menu dan Sub-menu

4.5.3 User Mengimport Data Training

User menginputkan data training dengan mengimportkan data excel yang berisikan data data training untuk di olah sebagai pengolahan naive bayes untuk mencari probabilitas dalam penginputan data testing.



Gambar 4.17 Gambar Import Data Training



Gambar 4.18 Gambar Pilih Data Excel

Hover Data Table

Data Training Berhasil Diimport!

Import Training

No	GH	PPGD	SAR	IMPK	Replng	SK	Minat	Divisi
1	sedang	sedang	sedang	tinggi	tinggi	tinggi	SURVIVAL	SURVIVAL
2	tinggi	tinggi	tinggi	tinggi	tinggi	tinggi	SURVIVAL	SURVIVAL
3	sedang	tinggi	rendah	tinggi	sedang	tinggi	SURVIVAL	SURVIVAL
4	rendah	rendah	sedang	sedang	tinggi	rendah	SURVIVAL	RC
5	rendah	rendah	rendah	rendah	rendah	sedang	SURVIVAL	SURVIVAL
6	rendah	sedang	rendah	tinggi	sedang	tinggi	SURVIVAL	SURVIVAL
7	rendah	rendah	sedang	sedang	tinggi	tinggi	SURVIVAL	SURVIVAL
8	tinggi	sedang	sedang	rendah	sedang	tinggi	SURVIVAL	SAR
9	sedang	sedang	rendah	sedang	sedang	sedang	SURVIVAL	SURVIVAL
10	sedang	sedang	tinggi	sedang	tinggi	tinggi	SURVIVAL	SURVIVAL
No	GH	PPGD	SAR	IMPK	Replng	SK	Minat	Divisi

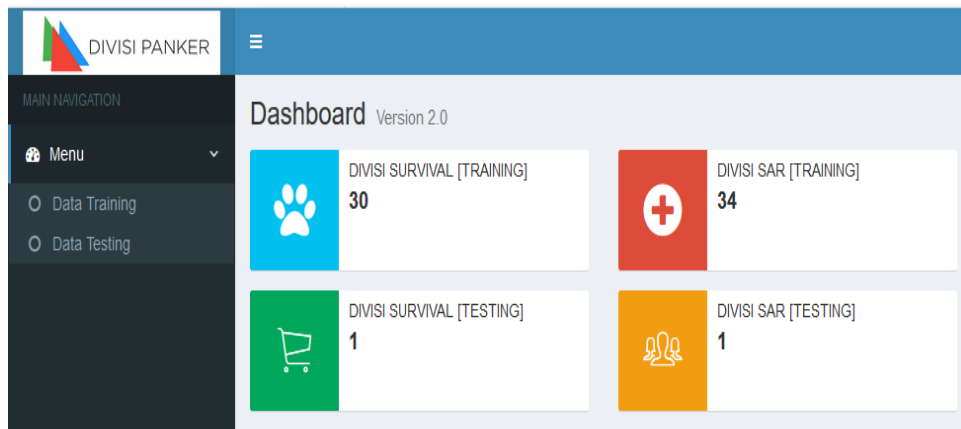
Showing 1 to 10 of 45 entries

Previous 1 2 3 4 5 Next

Gambar 4.19 Gambar Status Berhasil Import

4.5.4 User Memilih Menu Testing

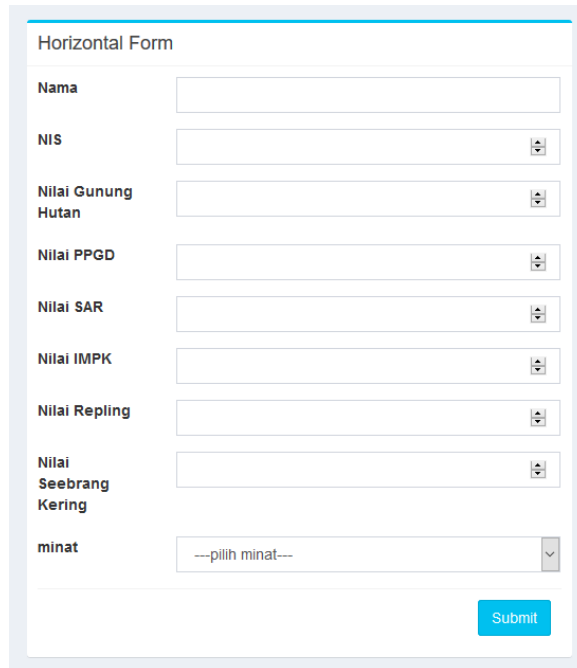
User memilih menu tesing untuk melakukan pengujian data testing, dalam halaman menu testing terdapat sebuah form untuk mengklasifikasikan pengujian data testing.



Gambar 4.20 Tampilan Menu dan sub-menu

4.5.5 User Melakukan Klasifikasi Calon Pengurus

Setelah user memilih menu testing maka user mengklasifikasikan data testing calon pengurus untuk menentukan rekomendasi calon pengurus ke divisi yang direkomendasikan.



Horizontal Form

Nama

NIS

Nilai Gunung Hutan

Nilai PPGD

Nilai SAR

Nilai IMPK

Nilai Repling

Nilai Seebrang Kering

minat

Gambar 4.21 form Klasifikasi Calon Pengurus

4.5.6 User Melihat Data Testing

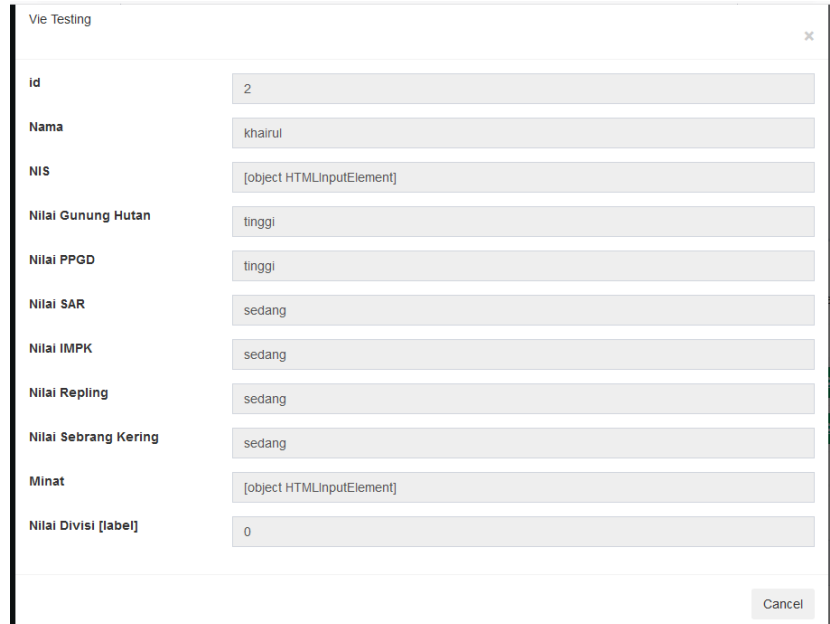
User melihat tabel yang berisikan data data hasil dari klasifikasi yang berbentuk data testing yang ditampilkan nama dan label nya .tabel dapat dilihat pada gambar berikut:

Show	10	entries	Search:	<input type="text"/>
No	nama	nis	[label] Divisi	action
1	Khairu Anwar	321094	Survival	<input type="button" value="delete"/> <input type="button" value="view"/> <input type="button" value="print"/>
2	Hegariyadi Zdelvanza	321093	SAR	<input type="button" value="delete"/> <input type="button" value="view"/> <input type="button" value="print"/>
3	Sulastri	321092	RC	<input type="button" value="delete"/> <input type="button" value="view"/> <input type="button" value="print"/>
4	Eva Elisa	321091	SAR	<input type="button" value="delete"/> <input type="button" value="view"/> <input type="button" value="print"/>
5	Indah Nurhidayah	321090	SAR	<input type="button" value="delete"/> <input type="button" value="view"/> <input type="button" value="print"/>
6	Alifah Qoulan Tsabita	321089	SAR	<input type="button" value="delete"/> <input type="button" value="view"/> <input type="button" value="print"/>
7	Sopiah	321088	SAR	<input type="button" value="delete"/> <input type="button" value="view"/> <input type="button" value="print"/>
8	Vivi Indriani	321087	RC	<input type="button" value="delete"/> <input type="button" value="view"/> <input type="button" value="print"/>
9	Efi Fitriani	321086	SAR	<input type="button" value="delete"/> <input type="button" value="view"/> <input type="button" value="print"/>
10	Dina Amalia	321085	SAR	<input type="button" value="delete"/> <input type="button" value="view"/> <input type="button" value="print"/>
No	nama	nis	[label] Divisi	action

Gambar 4.22 Tabel Testing

4.5.7 User Melihat Hasil Data Testing

User melihat hasil testing klasifikasi dan melihat rincian secara detail nilai nilai sesuai dengan atribut dan label yang dimasukkan melalui proses klasifikasi.



The 'View Testing' modal window displays the following data:

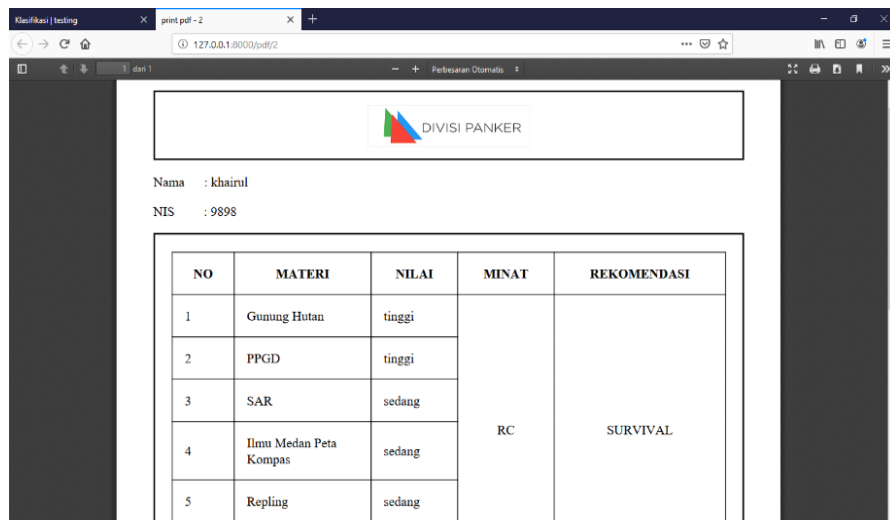
Attribute	Value
id	2
Nama	khairul
NIS	[object HTMLInputElement]
Nilai Gunung Hutan	tinggi
Nilai PPGD	tinggi
Nilai SAR	sedang
Nilai IMPK	sedang
Nilai Repling	sedang
Nilai Sebrang Kering	sedang
Minat	[object HTMLInputElement]
Nilai Divisi [label]	0

Cancel

Gambar 4.23 View Modal Testing

4.5.8 User Mengexport Pdf Hasil Testing

User mengekspor data-data dari hasil dari klasifikasi data testing yang disimpan dalam tabel dalam bentuk pdf yang selanjutnya akan diprint sebagai dokumentasi untuk pengurus dan pelaporan hasil rekomendasi ke pada calon pengurus.



The PDF document displays the following table:

NO	MATERI	NILAI	MINAT	REKOMENDASI
1	Gunung Hutan	tinggi	RC	SURVIVAL
2	PPGD	tinggi		
3	SAR	sedang		
4	Ilmu Medan Peta Kompas	sedang		
5	Repling	sedang		

Gambar 4.24 Tampilan Export Testing Pdf

4.6 Hasil Pengujian *Black-Box*

No	Desripsi Pengujian	Prosedur Pengujian	Hasil Yang diharapkan	Hasil Pengujian	Kesimpulan
User Aplikasi Klasifikasi Pendivisian Organisasi Panker					
1	user melakukan login	Masukan url -Masukan Username dan Password -klik login	User masuk halaman aplikasi	User memasuki tampilahan awal halaman training	Sesuai yang diharapkan
2	User memilih menu testing	-klik menu Pilih sub menu testing	User berada di halaman testing	User masuk halaman testing	Sesuai yang diharapkan
3	User menginputkan form data klasifikasi	-isi data form klasifikasi	-data masuk di tabel testing	-data masuk di tabel testing	Sesuai yang diharapkan
4	User melakukan inport data training	-klik tombol “import training”	-data testing masuk pada tabel testing	-data muncul pada tabel testing	Sesuai yang harapkan
5	User melakukan export pdf	Klik icon”print”	Masuk halaman Pdf	Masuk pada halaman web	Sesuai yang diharapkan
6	User melakukan logout	-Klik username -klik tombol logout	User keluar dari halaman aplikasi	User keluar dari halaman aplikasi	Sesuai yang diharapkan

4.7 Kelebihan dan Kekurangan

A) Kelebihan

Adapun kelebihan dari aplikasi ini adalah:

- a) Aplikasi ini membantu pengurus organisasi untuk merekomendasikan divisi sesuai dengan nilai nilai atribut yang dimasukkan di aplikasi
- b) Aplikasi ini mempermudah dokumentasi pendivisian calon pengurus
- c) Aplikasi ini mempermudah penyampaian informasi ke calon pengurus

B) Kekurangan

Adapun kekurangan dari aplikasi ini adalah:

- a) Aplikasi ini hanya terdapat satu user
- b) Aplikasi ini masih belum memiliki fitur lengkap
- c) Penyampaian data pada aplikasi ini masih sederhana

BAB V

PENUTUP

5.1 Kesimpulan

Setelah penulis menyelesaikan laporan ini, maka penulis telah mendapatkan banyak hal-hal yang bermanfaat yang terkait dengan pengerjaan Aplikasi Klasifikasi Pendivisian Organisasi Panker Menggunakan Naive Bayes, dengan begitu penulis dapat menyimpulkan sebagai berikut:

1. Klasifikasi Pendivisian Organisasi Panker Menggunakan Naive Bayes bisa digunakan sebagai media pengkelompokan divisi di Sekretariat Pecinta Alam PANKER.
2. Aplikasi Klasifikasi Pendivisian Organisasi Panker Menggunakan Naive Bayes dapat memudahkan Pengurus skeretariat khususnya bagian Kedivisian untuk memperoses pengkelompokan divisi.
3. Aplikasi Klasifikasi Pendivisian Organisasi Panker Menggunakan Naive Bayes ini menampilkan hasil dari pemrosesan data testing dan mempunyai salah satu fitur export pdf untuk mengetahui hasil yang lebih jelas.

5.2 Saran

Klasifikasi Pendivisian Organisasi Panker Menggunakan Naive Bayes ini tidak jauh dari kekeurangan dan kelemahan pada sistem yang dibuat. Berdasarkan kesimpulan dari pembahasan maka terdapat beberapa saran yang harus diperhatikan untuk merubah menjadi baik, meliputi :

1. Aplikasi dapat dikembangkan agar bisa digunakan mencangkup beberapa fitur yang bisa ditambahkan atau di sempurnakan.
2. Rancangan tampilan dapat dikembangkan lebih menarik untuk memudahkan dalam penggunaan.

Menambahkan panduan penggunaan aplikasi untuk memudahkan pengurus yang baru untuk menggunakan aplikasi ini.

DAFTAR PUSTAKA

- Arif. 2018. Pengertian composer. <http://www.kursuswebsite.org/pengertian-composer/>. Diakses pada tanggal 21 Mei 2019.
- Dsn, Cahayahttps. 2018.Apa itu naive bayes. http://cahyadsn.phpindonesia.id/extra/naive_bayes.php . Diakses pada tanggal 27 April 2019
- Daniel, 2016. Pengertian Xampp dan Fungsinya. <http://www.kursuswebsite.org/pengertian-xampp-dan-fungsi-nya/>. Diakses pada tanggal. Diakses pada tanggal 08 Agustus 2019.
- Fadul, Fadullah. 2018. Belajar Bootstrap Untuk Pemula. Apa itu Bootstrap ?.
<https://www.apacara.com>. Diakses pada tanggal 04 Mei 2019.
- Febiyan, Arya. 2014. Apa itu composer. <https://www.dumetschool.com/blog/Apa-Itu-Composer>. Diakses pada tanggal 21 Mei 2019.
- H, Rendra. 2018. Javascript adalah : Pengertian Javascript dan Dasar Penggunaan Javascript. <https://www.webhostingterbaik.id>. Dikakses pada tanggal 03 April 2019.
- Halwa, RF. 2018. Apa itu CSS (Cascading Style Sheets) ?.<https://www.apacara.com/tutorial/apa/apa-itu-css-cascading-style-sheets.html>.Diakses pada tanggal 21 Mei 2019.
- Ham, Hanry. Kelebihan Menggunakan Laravel Web Development. <http://socs.binus.ac.id/2018/12/13/kelebihan-menggunakan-laravel-web-development/>. Diakses pada tanggal 18 Juli 2019.
- Harefa, Efriaman. 2016. Pengertian Website. <https://www.citratek.co.id>. Diakses pada tanggal 02 Maret 2019.
- N,Sora.2014.Pengertian Basis Data dan Sistem Basis Data <http://www.Pengertianku.net>. Diakses pada tanggal 09 Maret 2019.
- Osd, Dimas. 2015.Pengertian SI (Sistem Informasi). <https://www.kompasiana.com>. Diakses pada tanggal 13 Maret 2019.
- Rahardi,Wisa.2014.Mengenal Html <https://www.blogooblok.com/2014/04/belajar->

html-mengenal-tag-elemen-atribut.html. Diakses pada tanggal 02 maret 2019

Ritonga, Pahmi. 2015. apa itu uml? <https://bangpahmi.com/pengertian-unified-modeling-language-uml-dan-modelnya-menurut-pakar-dan-ahli/>. Di akses pada tanggal 27 Mei 2019 Pahmi Ritonga — April 25, 2015

Wajib Anda Ketahui. <https://www.nesabamedia.com/pengertian-php-dan-fungsinya/>. Diakses pada tanggal 21 Mei 2019.

Syafitri Irmayani. 2018. Pengertian PHP Beserta Fungsi dan Sejarah PHP yang Wajib Anda Ketahui. <https://www.nesabamedia.com/pengertian-php-dan-fungsinya/>. Diakses pada tanggal 21 Mei 2019.

Winarso, Bambang. 2016. Apa Itu Google Chrome. <https://dailysocial.id/post>. Diakses pada tanggal 27 April 2019.

LAMPIRAN

LAMPIRAN 1 HASIL PENGUMPULAN DATA TRAINING								
Data Training Organisasi Panker								
Keterangan =		Kode Nilai **		Kode Minat Divisi*		Kode Status Nilai *100		
		Nilai Rendah = 0		Survival = 0		Nilai <=44 (Nilai Rendah)		
		Nilai Sedang = 1		SAR = 1		Nilai <=84 (Nilai Sedang)		
		Nilai Tinggi = 2		Rc = 2		Nilai > 84 (Nilai Tinggi)		
N	Nilai	Nilai	Nilai	Nilai	Nilai	Nilai	Mina	Divis
0.	Gunu ng Hutan **	Penolong an Pertama Gawat Darurat* *	SAR **	IMPK **	Repling **	Sebara ng Kering **	t*	i*
1	2	2	1	1	0	0	0	0
2	2	2	1	1	1	0	0	0
3	2	2	1	1	0	1	0	0
4	2	2	2	2	1	1	0	1
5	2	2	0	0	0	0	0	0
6	2	2	2	2	1	1	0	1
7	0	0	1	1	0	0	0	1
8	1	1	2	2	1	0	0	1
9	1	1	2	2	0	1	0	1
10	0	1	2	1	0	0	0	1
11	2	1	0	0	1	1	0	0
12	1	0	0	0	0	0	0	0
13	2	0	2	1	0	0	0	1

14	0	1	2	0	0	0	0	1
15	2	1	2	2	1	1	0	1
16	2	2	2	2	1	1	1	1
17	1	1	2	2	0	0	1	1
18	0	0	2	2	1	1	1	1
19	1	1	1	1	0	0	1	0
20	2	1	2	1	1	1	1	0
21	2	2	2	2	0	0	1	1
22	0	0	2	2	0	0	1	1
23	1	1	2	1	0	1	1	1
24	1	2	2	2	0	1	1	1
25	2	2	2	1	1	1	1	0
26	2	2	2	2	2	2	1	2
27	0	0	1	1	1	0	1	1
28	0	1	1	2	1	1	1	1
29	0	0	0	1	1	1	1	2
30	1	0	2	2	1	1	1	1
31	0	0	0	0	2	2	2	2
32	2	2	2	2	2	2	2	1
33	0	0	0	0	2	2	2	2
34	0	0	1	1	2	2	2	2
35	0	0	0	0	1	1	2	2
36	1	1	2	1	2	2	2	2
37	1	1	0	1	1	1	2	1
38	2	1	1	0	1	2	2	2
39	0	0	1	1	2	2	2	2
40	0	1	0	2	1	2	2	2
41	0	0	0	0	0	1	2	2
42	0	0	1	1	2	0	2	1

43	1	2	0	2	1	2	2	2
44	2	2	2	2	2	2	2	1
45	1	1	1	2	2	2	2	2

LAMPIRAN 2 KODE PROGRAM

Kode Program *Import data Training*

```
public function import_excel(Request $request)
{
    // validasi
    $this->validate($request, [
        'file' => 'required|mimes:csv,xls,xlsx' ]);
    // menangkap file excel
    // $file =$request->file('file')->store('excel-files');
    $file = $request->file('file');
    // membuat nama file unik
    $nama_file = rand().$file->getClientOriginalName();
    // upload ke folder file_siswa di dalam folder public
    $file->move('file_training',$nama_file);
    // import data
    // dd($file);

    Excel::import(new
    TrainingImport,public_path('/file_training/'.$nama_file));
    // notifikasi dengan session
    Session::flash('sukses','Data Training Berhasil Diimport!');

    // alihkan halaman kembali
    return redirect('/index');
}
```


Kode Program Klasifikasi data

```
public function store(Request $request)

{
    $messages = [
        'required' => ' wajib diisi ',
        'min' => ':attribute harus diisi minimal :min
karakter ya ',
        'max' => ':attribute harus diisi maksimal :max
karakter ya ',
    ];
    $this->validate($request, [
        'nama'=> 'required',
        'nis_test'=>'required',
        'gh'=>'required',
        'ppgd'=>'required',
        'sar'=>'required',
        'impk'=>'required',
        'repling'=>'required',
        'sk'=>'required',
        'minat'=>'required',
    ], $messages);
    //memasukan data ke variable array
    $data[0]= $request->get('gh');
    $data[1]= $request->get('ppgd');
    $data[2]= $request->get('sar');
    $data[3]= $request->get('impk');
    $data[4]= $request->get('repling');
    $data[5]= $request->get('sk');
    $minat= $request->get('minat');
    $atribut = $this->kat_atribut($data);
```

```

        $data_nb = array(  'gh'=>$atribut[0],
                           'ppgd'=>$atribut[1],
                           'sar'=>$atribut[2],
                           'impk'=>$atribut[3],
                           'repling'=>$atribut[4],
                           'sk'=>$atribut[5],
                           'minat'=>$minat);

        $hasil = $this->nb($data_nb);

//insert data nilai yang sudah dikonver menjadi sedang
renda atau tinggi ke database dalam bentuk integer
        $dttesting= new testing();
        $dttesting->nama_test = $request->get('nama');
        $dttesting->nis_test = $request->get('nis_test');
        $dttesting->n_gh_test = $atribut[0];
        $dttesting->n_ppgd_test = $atribut[1];
        $dttesting->n_sar_test = $atribut[2];
        $dttesting->n_impk_test = $atribut[3];
        $dttesting->n_repling_test = $atribut[4];
        $dttesting->n_sebrang_kering_test = $atribut[5];
        $dttesting->minat_test = $minat;
        $dttesting->divisi_test = $hasil['hasil'];
        //      dd($dttesting);
        $dttesting->save();

        return redirect()->route('testing')->with('alert-
success', 'data berhasil dimasukan');
    }

    //mengkonvert dari nilai 0,1,2 yang mewakili rendah
    sedang dan tinggi

    private function kat_atribut($data){

```

```

for($i=0; $i<count($data); $i++){
switch ($data[$i]) {
case ($data[$i] <= 44):
$x[$i] = 0;
break;
case ($data[$i] <= 84):
$x[$i] = 1;
break;
case ($data[$i] > 84):
$x[$i] = 2;
break;
default:
$x[$i] = "data tidak ada!";
break;
}
}
return $x;
}

private function nb($data){

    // memasukan data yang akan di proses ke dalam
variable baru

        $gh = $data['gh'];
        $ppgd = $data['ppgd'];
        $n_sar = $data['sar'];
        $impk = $data['impk'];

        $repling = $data['repling'];
        $sk = $data['sk'];

        $minat = $data['minat'];
        $table_data = training::get();
        $tot_row = $table_data->count();
        $survival = training::where('divisi',0);

```

```

    $survival_row = $survival->count();
    $sar = training::where('divisi',1);
    $sar_row = $sar->count();
    $rc = training::where('divisi',2);
    $rc_row = $rc->count();

    //mengambil jumlah data baris berdasarkan attribute
    dan label

    //cnth : total attribute gh sedang dengan
    label Survival dibagi keseluruhan jumlah baris
    // dari label survival

    $tot_gh_survival = training::where('nilai_gh',$gh)-
    >where('divisi','0')->count();
    $p_gh_survival = $tot_gh_survival / $survival_row;
    $tot_gh_sar =
    training::where('nilai_gh',$gh)->where('divisi',1)-
    >count();
    $p_gh_sar = $tot_gh_sar / $sar_row;
    $tot_gh_rc = training::where('nilai_gh',$gh)-
    >where('divisi',2)->count();
    $p_gh_rc = $tot_gh_rc / $rc_row;

    $tot_ppgd_survival =
    training::where('nilai_ppgd',$ppgd)-
    >where('divisi',0)->count();
    $p_ppgd_survival = $tot_ppgd_survival /
    $survival_row;
    $tot_ppgd_sar = training::where('nilai_ppgd',$ppgd)-
    >where('divisi',1)->count();
    $p_ppgd_sar = $tot_ppgd_sar / $sar_row;

```

```

$tot_ppgd_rc = training::where('nilai_ppgd',$ppgd) -
>where('divisi',2)->count();
$p_ppgd_rc = $tot_ppgd_rc / $rc_row;
$tot_sar_survival =
training::where('nilai_sar',$n_sar) -
>where('divisi',0)->count();
$p_sar_survival = $tot_sar_survival / $survival_row;
$tot_sar_sar = training::where('nilai_sar',$n_sar) -
>where('divisi',1)->count();
$p_sar_sar = $tot_sar_sar / $sar_row;
$tot_sar_rc = training::where('nilai_sar',$n_sar) -
>where('divisi',2)->count();
$p_sar_rc = $tot_sar_rc / $rc_row;

$tot_impk_survival =
training::where('nilai_impk',$impk) -
>where('divisi',0)->count();
$p_impk_survival = $tot_impk_survival /
$survival_row;
$tot_impk_sar = training::where('nilai_impk',$impk) -
>where('divisi',1)->count();
$p_impk_sar = $tot_impk_sar / $sar_row;
$tot_impk_rc = training::where('nilai_impk',$impk) -
>where('divisi',2)->count();
$p_impk_rc = $tot_impk_rc / $rc_row;

$tot_repling_survival =
training::where('nilai_repling',$repling) -
>where('divisi',0)->count();
$p_repling_survival = $tot_repling_survival /
$survival_row;

```

```

$tot_repling_sar =
training::where('nilai_repling',$repling)-
>where('divisi',1)->count();
$sp_repling_sar = $tot_repling_sar / $sar_row;
$tot_repling_rc =
training::where('nilai_repling',$repling)-
>where('divisi',2)->count();
$sp_repling_rc = $tot_repling_rc / $rc_row;

$tot_sk_survival =
training::where('nilai_sebrang_kering',$sk)-
>where('divisi',0)->count();
$sp_sk_survival = $tot_gh_survival / $survival_row;
$tot_sk_sar =
training::where('nilai_sebrang_kering',$sk)-
>where('divisi',1)->count();
$sp_sk_sar = $tot_sk_sar / $sar_row;
$tot_sk_rc =
training::where('nilai_sebrang_kering',$sk)-
>where('divisi',2)->count();
$sp_sk_rc = $tot_sk_rc / $rc_row;

$tot_minat_survival = training::where('minat',$minat)-
>where('divisi',0)->count();
$sp_minat_survival = $tot_minat_survival /
$survival_row;
$tot_minat_sar = training::where('minat',$minat)-
>where('divisi',1)->count();
$sp_minat_sar = $tot_minat_sar / $sar_row;
$tot_minat_rc =
training::where('minat',$minat)->where('divisi',2)-
>count();

```

```

$sp_minat_rc = $tot_minat_rc / $rc_row;

        //seluruh hasil dari pembagian diatas dengan
        label yang sama dikalikan

$x_survival = $p_gh_survival * $p_ppgd_survival *
$p_sar_survival * $p_impk_survival *
$p_repling_survival * $p_sk_survival *
$p_minat_survival;
        $x_sar          = $p_gh_sar * $p_ppgd_sar *
        $p_sar_sar * $p_impk_sar * $p_repling_sar * $p_sk_sar
        * $p_minat_sar;
        $x_rc           = $p_gh_rc * $p_ppgd_rc *
        $p_sar_rc * $p_impk_rc * $p_repling_rc * $p_sk_rc *
        $p_minat_rc;

        //jumlah baris yang dimiliki label dibagi
        total keseluruhan baris
        // cnth: jumlah baris survival 34 dibagi
        keseluruhan baris yang totalnya 90
$b_survival      = $survival_row / $tot_row;
$b_sar           = $sar_row / $tot_row;
$b_rc           = $rc_row / $tot_row;

        // hasil dari perkalian semua atribut
        // dengan label yg sama dikalikan dengan hasil
        dari
        // pembagian total baris label dan total
        keseluruhan baris data

$p_survival = $x_survival * $b_survival;
$p_sar = $x_sar * $b_sar;

```

```
$p_rc = $x_rc * $b_rc;

    // menampilkan perbandingan hasil perkalian di
    atas
    $data['survival'] = $p_survival;
    $data['sar'] = $p_sar;
    $data['rc'] = $p_rc;

    //menampilkan hasil hasil terbesar dari
    perbandingan
    if(($p_rc < $p_survival) && ($p_survival > $p_sar)){
        $data['hasil'] = 0;
    }else if(($p_rc < $p_sar) &&
    ($p_sar > $p_survival)){
        $data['hasil'] = 1;
    }else if(($p_sar < $p_rc) &&
    ($p_rc > $p_survival)){
        $data['hasil'] = 2;
    }else{
        $data['hasil'] =
    "Error";
    }
    return $data;
}
```


LAMPIRAN 3 BIODATA PENULIS

BIODATA PENULIS TUGAS AKHIR

Nama : Khairul Anwar
Nomor Induk Mahasiswa : 1603044
Jenis Kelamin : Laki-Laki
Tempat, Tanggal Lahir : Indramayu, 22 Desember 1997
Status : Belum Menikah
Agama : Islam
Asal Ijazah Sekolah : Nama Sekolah Kota Sekolah Tahun Ijazah
SD : -SDN Kliwed II -Indramayu -2010
SMP : -SMPN 1 Kertasemaya -Indramayu -2013
SMA : -SMAN 1 Sukagumiwang -Indramayu -2016

Program Studi/Jurusan : Teknik Informatika
Alamat Rumah Asal : Desa Kliwed Blok Sondol Rt 02 Rw 01
Kec.Ketasemaya, Kab. Indramayu, 45274.
No. Handphone : 08996367455
Email : anwaroye67@gmail.com
Nama Orang Tua : Dulkayi - Wastinih
Pekerjaan Orang Tua : Petani
Judul Tugas Akhir : Klasifikasi Pendivisian Organisasi Panker
Menggunakan Metode Naive Bayes
Pembimbing : 1. Eka Ismantohadi, S.Kom., M.Eng
2. Muhammad Anis Al Hilmi, S.Si.,M.T

Indramayu, 18 Agustus 2019
Penulis,

Khairul Anwar

NIM.1603044

