**KLASIFIKASI PENDIVISIAN ORGANISASI PANKER**

**MENGGUNAKAN 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 23 Agustus 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. Ahmad Lubis Ghozali, S.Kom.,M.Kom | Ketua Penguji | ……………… | .....………… |
| 1. Eka Ismantohadi., S.Kom., M.Eng | Sekretaris Penguji | ……………… | …...………. |
| 1. Alifia Puspaningrum, S.Pd.,M.Kom | Anggota | ……………… | ……..……. |

Indramayu, 2019

Ketua Jurusan Teknik Informatika

**Iryanto, S.Si., M.Si**

**NIP. 199008012019031014**

# 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 ekstrakulikuler pecinta alam yang mempunyai hubugan 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 Survival, Search And Rescue (SAR), Rock Climbing(RC).Dalam pembagian divisi pada organisasi panker masih menggunakan cara pembagian divsi secara manual dan sistem tunjuk, Tentu ini akan berpotensi mengurangi bakat yang dimiliki oleh calon pengurus untuk menyalurkan ilmu secara efektif kepada generasi penerus selanjutnya. Dalam sistem tunjuk ini akan berakibat pada penurunan kualitas pengajaran dibidang materi dikarenakan cara pengajaran kurang begitu baik dikarenakan pengaruh pembagian divisi pada awal pembagian. Berdasarkan analis tersebut maka dibuatlah aplikasi klasifikasi pendivisian yang berjudul “Klasifikasi Pendivisian Organisasi PANKER menggunakan metode naive bayes” agar memudahkan anggota pengurus untuk menentukan penempatan divisi yanng cocok untuk calon pengurus dengan melihat aspek nilai terbaik dengan metode perhitungan naive bayes. Sistem 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) Sekolah Menengah Atas Negeri 1 Sukagumiwang untuk mengklasifikasikan calon siswa dan menentukan rekomendasi divisi yang cocok untuk mereka. Hasil kuisioner diperoleh yaitu sangat baik dengan nilai presentase 47%, baik 47%, cukup 6%, dan kurang 0%.

Kata kunci : Divisi, Naive Bayes, Pecinta alam*.*

# *ABSTRACT*

*Lovers Nature Negeri 1 Kertasemaya (PANKER) is a nature lovers extracurricular organization that has a coordinative relationship with the Intra-School Organization (OSIS) located at State High School 1 Sukagumiwang. Panker is an alaman lover organization that has 3 divisions in the field of alaman lover namely Survival, Search and Rescue (SAR), Rock Climbing (RC). In the division of divisions in panker organizations still use the method of division divsi manually and the pointing system, this course will has the potential to reduce the talent possessed by prospective administrators to effectively channel knowledge to the next generation. In this pointing system will result in a decrease in the quality of teaching in the field of material due to the way the teaching is not so good due to the influence of the division division at the beginning of the division. Based on these analysts, a classification application was made entitled "PANKER Organizational Classification using Naive Bayes" to make it easier for members of the board to determine the placement of suitable divisions for prospective administrators by looking at the best value aspects with the Naive Bayes calculation method. The system was built using the PHP programming language, PhpMyadmin database with xampp server. The result obtained is a classification application that can be used by the work council or the management committee of the state-of-the-art 1 macemaya (PANKER) State High School 1 Sukagumiwang to classify prospective students and determine recommendations division that suits them. The questionnaire results obtained are very good with a percentage value of 47%, good 47%, enough 6%, and less 0%.*

*Keywords : Final task, Information Systems, Web, Laravel, Applications*

# MOTTO

“Kalau Ingin Tetap Hidup Jangan Lupa Bernapas”

# KATA PENGANTAR

Assalamu’alaikum Warahmatullahi Wabarakatuh.

Puji syukur penulis panjakan 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 ditentukkan. 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](#_Toc21565173)

[PERNYATAAN KEASLIAN iii](#_Toc21565174)

[ABSTRAK iv](#_Toc21565175)

[*ABSTRACT* v](#_Toc21565176)

[MOTTO vi](#_Toc21565177)

[KATA PENGANTAR vii](#_Toc21565178)

[DAFTAR ISI ix](#_Toc21565179)

[DAFTAR TABEL xiv](#_Toc21565180)

[DAFTAR GAMBAR xvi](#_Toc21565181)

[DAFTAR LAMPIRAN xviii](#_Toc21565182)

[BAB 1 PENDAHULUAN 1](#_Toc21565183)

[1.1 Latar Belakang Masalah 1](#_Toc21565184)

[1.2 Rumusan Masalah 2](#_Toc21565185)

[1.3 Batasan Masalah 2](#_Toc21565186)

[1.4 Tujuan Penelitian 2](#_Toc21565187)

[1.5 Manfaat Penelitian 2](#_Toc21565188)

[1.6 Sisitematis Penulisan Laporan 3](#_Toc21565189)

[BAB II LANDASAN TEORI 4](#_Toc21565190)

[2.1 Naive Bayes 4](#_Toc21565191)

[2.2 Teorema Naive Bayes 4](#_Toc21565192)

[2.3 XAMPP 5](#_Toc21565193)

[2.4 PhpMyadmin 5](#_Toc21565194)

[2.5 PHP 6](#_Toc21565195)

[2.6 Pengenalan PHP 6](#_Toc21565196)

[2.7 Basis Data 7](#_Toc21565197)

[2.8 Website 9](#_Toc21565198)

[2.9 HTML(*Hyper Text Markup Language)* 9](#_Toc21565199)

[2.9.1 Definisi HTML(*Hyper Text Markup Language)* 9](#_Toc21565200)

[2.9.2 Pengenalan dasar HTML (*Hyper Text Markup Language*) 10](#_Toc21565201)

[2.10 MySql 11](#_Toc21565202)

[2.11 UML (*Unfied Modeling Language*) 12](#_Toc21565203)

[2.12 Laravel 13](#_Toc21565204)

[2.13 Composer 15](#_Toc21565205)

[2.14 Black Box Testing 15](#_Toc21565206)

[2.15 Bootstrap 16](#_Toc21565207)

[2.2.1 Pengertian Bootstrap 16](#_Toc21565208)

[2.2.2 Komponen Bootstrap 16](#_Toc21565209)

[2.16 Google Chrome 21](#_Toc21565210)

[2.17 Javascript 21](#_Toc21565211)

[BAB III METODOLOGI PENELITIAN 23](#_Toc21565212)

[3.1 Metodologi Penelitian 23](#_Toc21565213)

[3.2 Metode Pengumpulan Data 24](#_Toc21565214)

[3.2.1 Wawancara 24](#_Toc21565215)

[3.2.2 Observasi 25](#_Toc21565216)

[3.2.3 Studi Pustaka 25](#_Toc21565217)

[3.3 Analis Kebutuhan Sistem 25](#_Toc21565218)

[3.3.1 Kebutuhan Hardware 26](#_Toc21565219)

[3.3.2 Kebutuhan Software 26](#_Toc21565220)

[3.4 Perancangan Sistem 26](#_Toc21565221)

[3.4.1 Use Case Diagram 26](#_Toc21565222)

[3.4.2 Activity Diagram 27](#_Toc21565223)

[3.4.2.1 Activity Diagram Login 27](#_Toc21565224)

[3.4.2.2 Activity Diagram Pelatih *Import* Data Training 28](#_Toc21565225)

[3.4.2.3 Activity Diagram Pelatih Input Klasifikasi Testing 29](#_Toc21565226)

[3.4.2.4 Activity Diagram Pelatih Tambah Pengurus 30](#_Toc21565227)

[3.4.2.5 Activity Diagram Pelatih Tambah Angkatan Pengurus 31](#_Toc21565228)

[3.4.2.6 Activity Diagram Pelatih Export Pdf Klasifikasi 32](#_Toc21565229)

[3.4.2.7 Activity Diagram Pengurus Tambah Siswa 33](#_Toc21565230)

[3.4.3 *Class* Diagram 34](#_Toc21565231)

[3.4.4 *Entity Relationship Diagram* (ERD) 33](#_Toc21565232)

[3.4.5 *Flowchart* Aplikasi 34](#_Toc21565233)

[3.4.6 Perancangan Awal Antar Muka 37](#_Toc21565234)

[3.4.7 Perancangan Database 46](#_Toc21565235)

[3.4.7.1 Tabel users 46](#_Toc21565236)

[3.4.7.2 Tabel training 46](#_Toc21565237)

[3.4.7.3 Tabel testing 47](#_Toc21565238)

[3.4.7.4 Tabel siswa 47](#_Toc21565239)

[3.4.7.5 Tabel roles 48](#_Toc21565240)

[3.4.7.6 Tabel pengurus 48](#_Toc21565241)

[3.4.7.7 Tabel kepengurusan 49](#_Toc21565242)

[3.4.7.8 Tabel keorganisasian 49](#_Toc21565243)

[3.4.7.9 Tabel Kelas 49](#_Toc21565244)

[3.4.7.10 Tabel angkatan\_pengurus 50](#_Toc21565245)

[BAB IV HASIL DAN PEMBAHASAN 51](#_Toc21565246)

[4.1 Hasil Penelitian 51](#_Toc21565247)

[4.2 Hasil Pustaka 51](#_Toc21565248)

[4.3 Implementasi Desain Antar Muka 51](#_Toc21565249)

[4.3.1 Implementasi Desain Antar Muka 51](#_Toc21565250)

[4.3.2 Implementasi Hasil Tampilan Intro Aplikasi 51](#_Toc21565251)

[4.3.3 Implementasi Hasil Tampilan Login 52](#_Toc21565252)

[4.3.4 Implementasi Hasil Tampilan Data Training 52](#_Toc21565253)

[4.3.5 Implementasi Hasil Tampilan Data Testing 53](#_Toc21565254)

[4.3.6 Implementasi Form Testing Klasifikasi 53](#_Toc21565255)

[4.3.7 Implementasi Hasil Tampilan Modal View Testing 54](#_Toc21565256)

[4.3.8 Implementasi Hasil Tampilan Export Pdf 55](#_Toc21565257)

[4.4 Implementasi Database 55](#_Toc21565258)

[4.4.1 Tabel users 56](#_Toc21565259)

[4.4.2 Tabel training 56](#_Toc21565260)

[4.4.3 Tabel testing 57](#_Toc21565261)

[4.5 Tahap Pengujian Sistem 60](#_Toc21565262)

[4.5.1 User Melakukan Login 60](#_Toc21565263)

[4.5.2 User Memilih Menu Training 61](#_Toc21565264)

[4.5.3 User Menginport Data Training 62](#_Toc21565265)

[4.5.4 User Memilih Menu Testing 63](#_Toc21565266)

[4.5.5 User Melakukan Klasifikasi Calon Pengurus 63](#_Toc21565267)

[4.5.6 Pelatih Melihat Data Testing 64](#_Toc21565268)

[4.5.7 Pelatih Melihat Hasil Data Testing 64](#_Toc21565269)

[4.5.8 Pelatih Mengexport Pdf Hasil Testing 65](#_Toc21565270)

[4.6 Hasil Pengujian Kuisioner 65](#_Toc21565271)

[4.7 Hasil Jawaban Kuisioner 67](#_Toc21565272)

[4.7.1 Rekap Pertanyaan Kuisioner Pelatih 69](#_Toc21565273)

[4.7.2 Rekap Kuisioner Pengurus 72](#_Toc21565274)

[4.7.3 Rekap Pertanyaan Kuisioner Siswa 74](#_Toc21565275)

[4.8 Rekapitulasi Hasil Perhitungan Semua Kuisioner 77](#_Toc21565276)

[4.9 Hasil Pengujian *Black-Box* 77](#_Toc21565277)

[4.10 Penjelasan Dataset 79](#_Toc21565278)

[4.11 Kelebihan dan Kekurangan 81](#_Toc21565279)

[BAB V PENUTUP 83](#_Toc21565280)

[5.1 Kesimpulan 83](#_Toc21565281)

[5.2 Saran 83](#_Toc21565282)

[DAFTAR PUSTAKA 85](#_Toc21565283)

[LAMPIRAN 87](#_Toc21565284)

# DAFTAR TABEL

[Tabel 2. 1 Simbol-Simbol Entitas 8](#_Toc20255656)

[Tabel 3. 1 Kebutuhan Hardware 26](#_Toc21561728)

[Tabel 3. 2 Kebutuhan Software 26](#_Toc21561729)

[Tabel 3. 3 Tabel users 46](#_Toc21561730)

[Tabel 3. 4 Tabel training 46](#_Toc21561731)

[Tabel 3. 5 Tabel testing 47](#_Toc21561732)

[Tabel 3. 6 Tabel siswa 47](#_Toc21561733)

[Tabel 3. 7 Tabel roles 48](#_Toc21561734)

[Tabel 3. 8 Tabel pengurus 48](#_Toc21561735)

[Tabel 3. 9 Tabel kepengurusan 49](#_Toc21561736)

[Tabel 3. 10 Tabel keorganisasian 49](#_Toc21561737)

[Tabel 3. 11 Tabel kelas 50](#_Toc21561738)

[Tabel 3. 12 Tabel angkatan\_pengurus 50](#_Toc21561739)

[Tabel 4. 1 Tabel users 56](#_Toc21561706)

[Tabel 4. 2 Tabel training 56](#_Toc21561707)

[Tabel 4. 3 Tabel testing 57](#_Toc21561708)

[Tabel 4. 4 Tabel siswa 57](#_Toc21561709)

[Tabel 4. 5 Tabel roles 58](#_Toc21561710)

[Tabel 4. 6 Tabel pengurus 58](#_Toc21561711)

[Tabel 4. 7 Tabel kepengurusan 58](#_Toc21561712)

[Tabel 4. 8 Tabel keorganisasi 59](#_Toc21561713)

[Tabel 4. 9 Tabel kelas 59](#_Toc21561714)

[Tabel 4. 10 Tabel angkatan\_pengurus 59](#_Toc21561715)

[Tabel 4. 11 Kuisioner untuk Pelatih 65](#_Toc21561716)

[Tabel 4. 12 Kuisioner untuk Pengurus 66](#_Toc21561717)

[Tabel 4. 13 Kuisioner untuk Siswa 66](#_Toc21561718)

[Tabel 4. 14 Hasil Jawaban Kuisioner Pelatih 67](#_Toc21561719)

[Tabel 4. 15 Hasil Jawaban Kuisioner Pengurus 67](#_Toc21561720)

[Tabel 4. 16 Hasil Jawaban Kuisioner Siswa 68](#_Toc21561721)

[Tabel 4. 17 Hasil Jawaban Kuisioner Keseluruhan 68](#_Toc21561722)

[Tabel 4. 18 Rekap Hasil Jawaban Kuisioner Pelatih 69](#_Toc21561723)

[Tabel 4. 19 Rekap Hasil Kuisioner Pengurus 72](#_Toc21561724)

[Tabel 4. 20 Rekap Hasil Kuisioner Siswa 74](#_Toc21561725)

[Tabel 4. 21 Hasil Uji Coba Data Testing Model 79](#_Toc21561726)

[Tabel 4. 22 Uji Coba Data Testing Real 80](#_Toc21561727)

# DAFTAR GAMBAR

[Gambar 2. 1 Tampilan XAMPP Control Panel 5](#_Toc20268641)

[Gambar 2. 2 Tampilan Php Myadmin 6](#_Toc20268642)

[Gambar 2. 3 Logo Google Chrome 21](#_Toc20268643)

[Gambar 3. 1 Gambar Metodologi Penelitian 23](#_Toc20268611)

[Gambar 3. 2 Use case 27](#_Toc20268612)

[Gambar 3. 3 Activity Diagram Login User 28](#_Toc20268613)

[Gambar 3. 4 Activity Diagram Import Data Training 28](#_Toc20268614)

[Gambar 3. 5 Activity Diagram Klasifikasi Testing 29](#_Toc20268615)

[Gambar 3. 6 Activity Diagram Klasifikasi Testing 30](#_Toc20268616)

[Gambar 3. 7 Activity Diagram Klasifikasi Testing 31](#_Toc20268617)

[Gambar 3. 8 Activity Diagram Export Pdf 32](#_Toc20268618)

[Gambar 3. 9 Activity Diagram Tambah siswa 33](#_Toc20268619)

[Gambar 3. 10 Class Diagram 34](#_Toc20268620)

[Gambar 3. 11 Entity Relationship Diagram (ERD) 33](#_Toc20268621)

[Gambar 3. 12 Gambar FLowchart Pelatih 34](#_Toc20268622)

[Gambar 3. 13 FLowchart Pengurus 35](#_Toc20268623)

[Gambar 3. 14 Flowchart Siswa 36](#_Toc20268624)

[Gambar 3. 15 Flowchart Sistem 37](#_Toc20268625)

[Gambar 3. 16 Rancangan Halaman Awal 38](#_Toc20268626)

[Gambar 3. 17 Rancangan Halaman Login 38](#_Toc20268627)

[Gambar 3. 18 Rancangan Halaman Tampil Data Training 39](#_Toc20268628)

[Gambar 3. 19 Rancangan Halaman Tampil Data Testing 40](#_Toc20268629)

[Gambar 3. 20 Rancangan Halaman View Data Testing 40](#_Toc20268630)

[Gambar 3. 21 Rancangan Halaman Pelatih Tambah Pengurus 41](#_Toc20268631)

[Gambar 3. 22 Rancangan Halaman Pelatih Tambah Angkatan Pengurus 41](#_Toc20268632)

[Gambar 3. 23 Rancangan Halaman Pelatih Data Master Kelas 42](#_Toc20268633)

[Gambar 3. 24 Rancangan Halaman Pelatih Data Master Keorganisasian 42](#_Toc20268634)

[Gambar 3. 25 Rancangan Halaman Pelatih Data Master Kepengurusan 43](#_Toc20268635)

[Gambar 3. 26 Rancangan Halaman Pelatih Tampil Export Pdf 43](#_Toc20268636)

[Gambar 3. 27 Rancangan Halaman Pengurus Data Siswa 44](#_Toc20268637)

[Gambar 3. 28 Rancangan Halaman Pengurus Tambah Akun 44](#_Toc20268638)

[Gambar 3. 29 Rancangan Halaman Siswa 45](#_Toc20268639)

[Gambar 3. 30 Rancangan Halaman Siswa Pilih Minat 45](#_Toc20268640)

[Gambar 4. 1 Tampilan Halaman Awal 52](#_Toc21561740)

[Gambar 4. 2 Tampilan Halaman Login 52](#_Toc21561741)

[Gambar 4. 3 Tampilan Tabel Training 53](#_Toc21561742)

[Gambar 4. 4 Tampilan Halaman Testing 53](#_Toc21561743)

[Gambar 4. 5 Tampilan Form Testing 54](#_Toc21561744)

[Gambar 4. 6Tampilan Tabel Testing 54](#_Toc21561745)

[Gambar 4. 7 Tampilan Modal View Testing 54](#_Toc21561746)

[Gambar 4. 8 Tampilan Tabel Testing 55](#_Toc21561747)

[Gambar 4. 9 Tampilan Export Pdf 55](#_Toc21561748)

[Gambar 4. 10 Chart Presentase Data Model 80](#_Toc21561749)

[Gambar 4. 11 Chart Presentase Data Real 81](#_Toc21561750)

# DAFTAR LAMPIRAN

Lampiran 1 Hasil Pengumpulan Data Training

Lampiran 2 Kode Program

Lampiran 3 Biodata Penulis

# BAB 1 PENDAHULUAN

## 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 yang akan berpengaruh kepada kualitas organisasi dari penempatan anggota pengurus pada divisi yang akan berpengaruh pada kurikulum yang disampaikan kepada calon pengurus akan menjadi penerus menggantikan pengurus yang lama.

Organisasi PANKER (Pecinta Alam Negeri Kertasemaya) 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.

**Algoritma Naive Bayes** merupakan sebuah metoda klasifikasi menggunakan metode probabilitas dan statistik yang 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 Naïve Bayes Classifier ini adalah asumsi yg sangat kuat (naïf) akan independensi dari masing-masing kondisi / kejadian. Naive Bayes Classifier bekerja sangat baik dibanding dengan model classifier lainnya (Anonim, 2017). Hal ini dibuktikan pada jurnal Xhemali, Daniela, Chris J. Hinde, and Roger G. Stone. “Naive Bayes vs. decision trees vs. neural networks in the classification of training web pages.” (2009), mengatakan bahwa “Naïve Bayes Classifier memiliki tingkat akurasi yg lebih baik dibanding model classifier lainnya” (Xhemali, 2009).

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.

## Rumusan Masalah

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

1. Bagaimana hasil dari pembuatan website klasifikasi pendivisian pada organisasi PANKER menggunakan metode naive bayes ?
2. Bagaimana sistem ini memberikan kemudahan pada pengurus, siswa dan pelatih dalam perekomendasian lebih mudah ?
3. Bagaimana membuat aplikasi pendivisian organisasi panker menggunakan metode naive bayes ini dengan alur pendivisian organisasi PANKER ?

## 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 organisasi PANKER.
2. Sistem ini dibangun menggunakan teknologi *web-based* dengan tools PHP, MYSQL, APACHE dan *software* pendukunglainnya.
3. Perancangan sistem menggunakan metode waterfall.

## 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 menggunakan bahasa pemrograman *php* menggunakan *framework* laravel berdasarkan rancangan yang sudah dibuat.

## Manfaat Penelitian

Dengan adanya penelitian ini diharapkan dapat memberikan manfaat yaitu:

1. Memudahka calon pengurus untuk mengetahui rekomendasi dari nilai-nilai dan pilihan yang dipilih untuk mendapatkan hasil direkomendasi divisi.
2. Membantu pengurus dan pelatih untuk mengklasifikasikan tiap-tiap calon pengurus .
3. Mempersingkat waktu dan mempermudah kerja pengurus dan pelatih.

## Sisitematis 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 Penelitan 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

## Naive Bayes

Naïve Bayes merupakan sebuah pengklasifikasian probabilistik sederhana yang menghitung sekumpulan probabilitas dengan menjumlahkan frekuensi dan kombinasi nilai dari dataset yang diberikan. Algoritma mengunakan 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 dikemukan 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 paremeter 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).

## 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).

|  |  |
| --- | --- |
| C:\Users\Khairul Gaming\AppData\Local\Microsoft\Windows\INetCache\Content.MSO\97A05F43.tmp | (2.1) |

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)

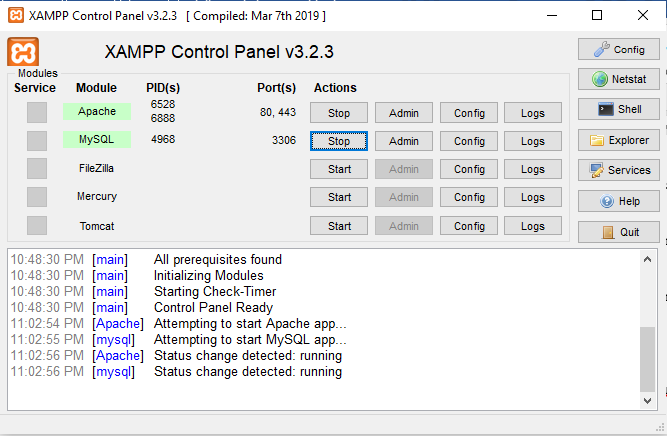
P(X|H) : Probabilitas X berdasarkan kondisi pada hipotesis H

P(X) : Probabilitas X

## XAMPP

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

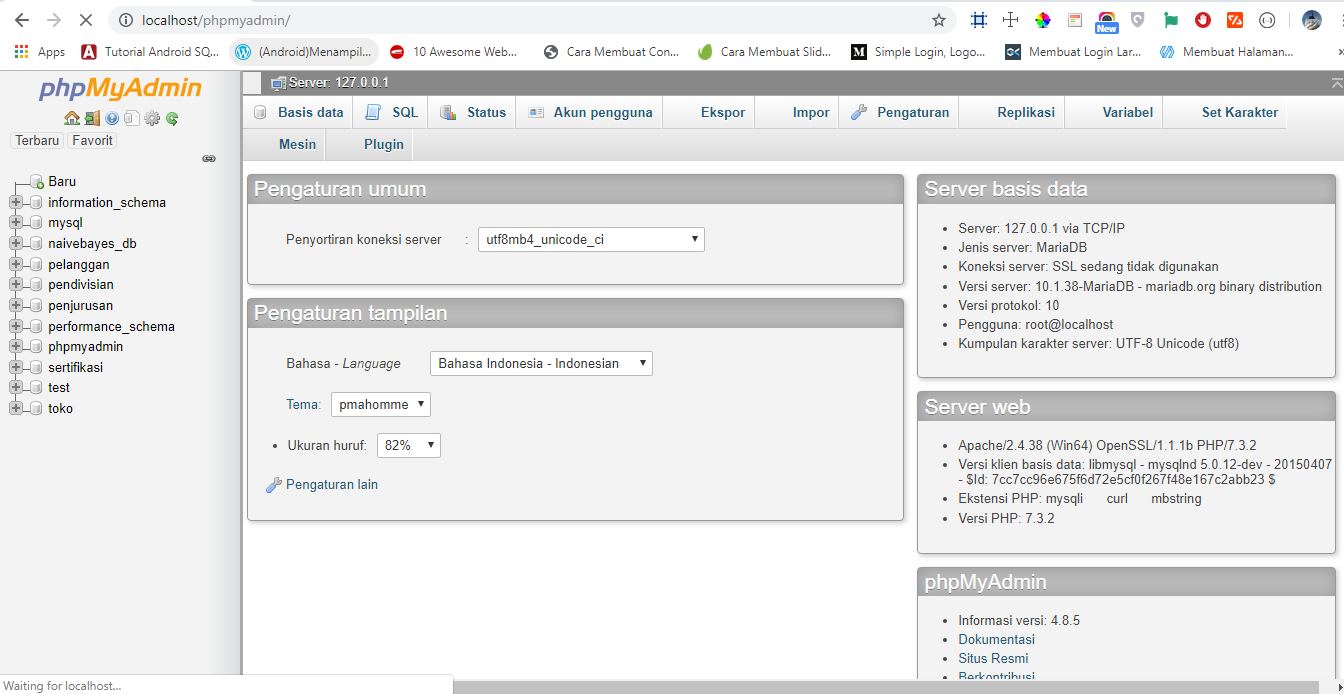
XAMPP memiliki kelebihan untuk bisa berperan sebagai server web Apache dalam melakukan simulasi pengembangan web. Tool pengembangan web ini mendukung teknologi web populer seperti PHP, MySql dan Perl. (Tumanggor Ronaldi. 2015)



Gambar 2. 1 Tampilan XAMPP Control Panel

## PhpMyadmin

phpMyAdmin adalah tools yang dapat digunakan dengan mudah untuk memanajemen database MySQL secara visual dan Server MySQl, sehingga kita tidak perlu lagi harus menulis query SQL setiap akan melakukan perintah operasi database”. Tools ini cukup populer, Anda dapat mendapatkan fasilitas ini ketika menginstal paket triad phpMyAdmin, karena termasuk dalam xampp yang sudah di instal. Menurut Nugroho (2013:71)



Gambar 2. 2 Tampilan Php Myadmin

## PHP

PHP berasal dari kata “*Hypertext Prepocessor*”, yaitu bahasa universal untuk penangangan 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 komunikas dengan database, mengumpulkan dan mengelompokkan informasi, kemudian menampilkannya pada saat ada tamu website memerlukannya menampilkan informasi sesuai permintaan user.

## 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  ?> |

## Basis Data

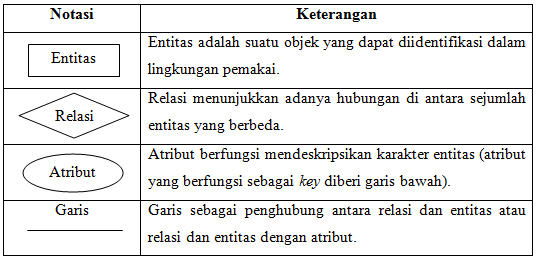
Basis data (database) adalah kumpulan data yang disimpan secara sistematis di dalam komputer yang dapat diolah atau dimanipulasi menggunakan perangkat lunak (program aplikasi) untuk menghasilkan informasi. Pendefinisian basis data meliputi spesifikasi berupa tipe data, struktur data dan juga batasan-batasan 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, menghidari 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.

Tabel 2. 1 Simbol-Simbol Entitas



1. **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.

1. **Data value**

Data value merupakan suatu informasi atau data aktual yang disimpan pada tiap data, elemen, atau atribut. Atribut nama pegawai menunjukan 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.

## 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.

## 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 tegantung pada aplikasi tertentu, karena dokumen HTMLdapat 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.

## Definisi 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.

## 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.

1. **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 tumpah tindih

1. **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> |

## 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.

1. Kelebihan MySQL

Adapun kelebihan MySQl dalam penggunaanya dalam database adalah:

1. Free atau gratis sehingga MySQL dapat dengan mudah untuk mendapatkannya.
2. MySQl stabil dan tangguh dalam pengoperasiannya
3. MySQl mempunyai sistem keamanan yang cukup baik
4. Sangat mendukung transaksi dan mempunyai banyak dukungan dari komunitas
5. Sangat fleksibel dengan berbagai macam program
6. Perkembangan dari MySQl sangat cepat
7. Kelemahan MySQL

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

1. 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.
2. Data yang dapat ditangani belum besar dan belum mendukung widowing function.

## 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 Squence (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*).

## 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 :

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

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

1. Waktu yang dibutuhkan untuk mengembangan projek website dengan menggunakan *framework*ini menjadi lebih cepat.
2. 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.

## 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).

## 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?

## Bootstrap

## Pengertian 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).

## Komponen Bootstrap

1. Dropdown

Dropdown merupakan menu-menu yang biasanya ditampilkan dalam bentuk list atau daftar bisa dibuat interaktif dengan Toggleable dropdown JavaScript plugin,

menu kontekstual untuk menampilkan link dalam format daftar. Hal ini dapat dibuat interaktif dengan dropdown Plugin JavaScript. Untuk menggunakan dropdown, hanya menyisipkan menu dropdown dalam kelas .dropdown

1. Button Groups

Button groups memungkinkan beberapa tombol untuk ditumpuk selaras bersamasama pada satu baris. Hal ini berguna ketika anda ingin menempatkan item seperti tombol alignment bersama-sama. Anda dapat menambahkan pilihan JavaScript radio dan checkbox style dengan Bootstrap Plugin Button.

1. Button Dropdown

Pada komponen ini akan dibahas tentang cara menambahkan menu button dropdown dengan menggunakan kelas Bootstrap. Untuk menambahkan dropdown ke tombol, cukup membungkus tombol dan menu dropdown di .btn-group. Anda juga dapat menggunakan <span class = "caret"> </ span> yang berfungsi sebagai indikator bahwa tombol adalah sebuah menu dropdown

1. Navs dan Navbar

Bootstrap menyediakan beberapa pilihan yang berbeda untuk elemen styling navigasi. Semua elemen navigasi dari berbagi markup yang sama dan kelas dasarnya .nav. Bootstrap juga menyediakan kelas helper, untuk berbagi markup dan kondisi. Kelas Swap modifier digunakan untuk memindahkan style masing-masing.

1. Tabel Navigasi atau Tabs

Untuk membuat menu navigasi tab: - Mulailah dengan dasar membuat daftar tak berurut dengan class dasar .nav - Tambahkan class .nav-tab.

b. Navbar

Navbar adalah fitur bagus dan merupakan salah satu ciri yang menonjol dari situs Bootstrap. Navbars merupakan komponen meta responsif yang berfungsi sebagai header navigasi untuk aplikasi atau situs anda. Pada intinya, navbar termasuk styling untuk nama situs dan navigasi dasar.

1. Standar Navbar

Untuk menambah standart navbar :

* + - * 1. Tambah kelas .navbar,.navbar-default ke tag <nav>.

b. Tambahkan role = "navigasi" pada elemen di atas, untuk membantu dengan aksesibilitas.

c. Tambah kelas header .navbar-header untuk elemen <div>. Sertakan elemen <a> dengan class navbar-brand. Hal ini akan memberikan teks ukuran yang sedikit lebih besar. - Untuk menambahkan link ke navbar, cukup tambahkan daftar unordered dengan kelas .nav, .navbar-nav.

1. Responsif Navba

Untuk menambahkan fitur responsif terhadap navbar, konten yang anda butuhkan untuk disisipkan dalam <div> dengan class .collapse, .navbar-collapse. Collapse terhubung oleh sebuah tombol yang memiliki kelas .navbar-toggle dan kemudian memiliki dua data-elemen. yang pertama, data-toggle digunakan untuk memberitahu JavaScript apa yang harus dilakukan dengan tombol. dan yang kedua, data-target, mengindikasikan elemen untuk beralih. Membuat tombol dengan 3 ikon

menggunakan class .icon-bar. Ini akan mengaktifkan unsur-unsur yang berada di .nav-collapse <div>. Agar fitur ini bekerja, Anda perlu menyertakan Bootstrap collapse plugin.

1. Breadcrumbs

Breadcrumbs adalah cara yang tepat untuk menampilkan informasi hirarki untuk sebuah situs. Dalam kasus blog, breadcrumbs dapat menunjukkan tanggal penerbitan, kategori, atau tag. Mereka menunjukkan lokasi halaman saat ini dalam hirarki navigasi.

1. Paginaion

Pagination adalah sebuah teknik untuk lebih mempermudah user agar tidak scroll terlalu jauh saat inign menampilkan data dalam bentuk yang ratusan hingga ribuan data. Dengan Menggunakan Pagination, Maka Akan terbentuk Halamanhalaman yang berurutan untuk menampilkan data-data tersebut

1. Label dan Badges
2. LABEL

Label yang besar untuk menawarkan jumlah, tips, atau markup lainnya untuk halaman. Dengan menggunakan class

1. BADGES

badges mirip dengan label, perbedaan utamanya adalah bahwa sudut budges lebih bulat. badges terutama digunakan untuk menyorot item baru atau yang belum dibaca. Untuk menggunakan budges hanya menambahkan <span class = "badge"> pada link, Bootstrap navs, dan banyak lagi.

1. Thumbnails Alert

Banyak situs membutuhkan cara untuk layout gambar, video, teks, dll dalam box, Bootstrap memiliki cara mudah untuk melakukan hal ini dengan thumbnail. Untuk membuat thumbnail menggunakan Bootstrap: • Tambahkan tag <a> dengan kelas .thumbnail di sekitar link gambar. • Hal ini menambah empat piksel padding dan border gambar abu-abu. • Pada hover, sebuah cahaya menguraikan gambar animasi

1. Progress Bar

Tujuan dari progress bar adalah untuk menunjukkan bahwa aset memuat, dalam proses, atau bahwa ada tindakan yang terjadi mengenai elemen pada halaman. Progress bar menggunakan CSS3 transisi dan animasi untuk mencapai beberapa efeknya. Fitur-fitur ini tidak didukung di Internet Explorer 9 dan di bawah atau versi Firefox yang lebih tua. Opera 12 tidak mendukung animasi.

1. Tambahkan <div> dengan class. Progress
2. Selanjutnya, di dalam di atas <div>, tambahkan kosong <div> dengan class .progress-bar.
3. Tambahkan atribut style dengan lebar dinyatakan sebagai persentase. Katakanlah misalnya, style = "60%"; menunjukkan bahwa progress bar berada di 60%.
4. Media Object dan Misc
5. MEDIA OBJECT

Media object adalah style objek abstrak untuk membangun berbagai jenis komponen (seperti komentar blog, Tweets, dll) yang menampilkan gambar kiri atau kanan-blok di samping konten tekstual. Tujuan dari objek media adalah membuat kode untuk mengembangkan blok informasi yang lebih pendek.

Tujuan dari objek media (markup ringan, extendability mudah) dicapai dengan menerapkan kelas untuk beberapa markup sederhana. Ada dua bentuk kelas pada objek media yaitu:

- .media: Kelas ini memungkinkan untuk menampilkan objek media (gambar, video, audio) ke kiri atau kanan blok konten.

- .media-list: Jika Anda sedang mempersiapkan list di mana item akan menjadi bagian dari list unordered (tak berurutan), gunakan kelas ini. .media-list berguna untuk komentar panjang atau daftar artikel

ii. MISC

Pada komponen ini terdapat beberapa jenis class dalam penggunaannya: seperti well, optional classes, close icon dan helper classes.

- Well Gunakan Well sebagai efek sederhana pada sebuah elemen untuk memberikan efek masukan pada tampilan.

|  |
| --- |
| <div class="well">  ....  </div> |

- Optional Kelas optional adalah pengaturan Kontrol padding dan sudut bulat dengan dua kelas pilihan pada pengubah. lihat contoh penggunaanya berikut Pada skala besar :

|  |
| --- |
| <div class="well well-large">  ....  </div> |

Pada Skala kecil:

|  |
| --- |
| <div class="well well-small">  ....  </div> |

- Close icon

Gunakan close icon generik untuk menolak konten seperti kata modal dan peringatan. contoh penggunaannya :

|  |
| --- |
| <button class="close">&times;</button> |

perangkat iOS memerlukan sebuah href="#" untuk me-klik peristiwa, jika Anda lebih suka menggunakan link.

|  |
| --- |
| <a class="close" href="#">&times;</a> |

- Helper class kelas ini difokuskan untuk menampilkan bantuan atau memberi solusi dari kelemahan terkecil

## 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 empunya 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).



Gambar 2. 3 Logo Google Chrome

## 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*).

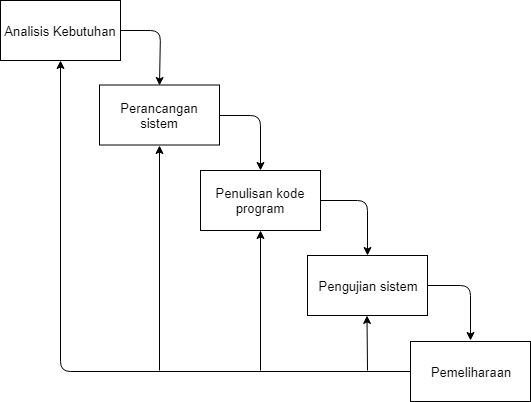
Untuk menjalankan Javascript hanya membutuhkan aplikasi *text editor* sebagai contoh sublime, atom, dreamweaver dan web browser. Javascript memiliki fitur high level *programming language*, *client side* dan berorientasi objek (Rendra H., 2018).

# BAB III METODOLOGI PENELITIAN

## 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,2016).

Langkah – langkah pembuatan dapat ditunjukan pada Gambar berikut:



Gambar 3. 1 Gambar Metodologi Penelitian

1. **Analisi Kebutuhan**

Sistem yang digunakan untuk mengklasifikasikan divisi organisasi PANKER pada Sekolah Menengah 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.

1. **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

1. **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.

1. **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.

1. **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.

## Metode Pengumpulan Data

## Wawancara

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

1. 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..

1. 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.

1. Data Parameter.

Dalam pengambilan data didapatkan 6 atribute praktek yang dibutuhkan untuk menjadi kerangka naive bayes data yaitu gunung hutan (gh), pertolongan pertama gawat darurat (ppgd), search and rescue (sar), ilmu medan peta kompas (impk), *repling*, sebrang kering. Untuk label minatnya didapatkan 3 data yaitu *survival*, sar, *rock climbing.* Dalam nilai parameter juga ada 3 yaitu rendah, sedang, tinggi untuk batas masing-masing parameter nilai yaitu tinggi dari 100-84, sedang 83-45, dan rendah 44-0.

## 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.

## 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 Sukaguwiwang.

## Analis Kebutuhan Sistem

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

## Kebutuhan Hardware

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

Tabel 3. 1 Kebutuhan Hardware

|  |  |
| --- | --- |
| Jenis *Harware* | Kebutuhan *Hardware* |
| Processor | Intel Core i3 |
| RAM | 2 GB atau lebih |
| *Hardisk* | 180 GB atau lebih |

## Kebutuhan Software

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

Tabel 3. 2 Kebutuhan Software

|  |  |
| --- | --- |
| Jenis *Software* | Kebutuhan *Software* |
| Sistem Operasi | Windows 7 Profesional |
| Desain Aplikasi | CorelDraw X8 |
| Bahasa *Scripting* | 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 |

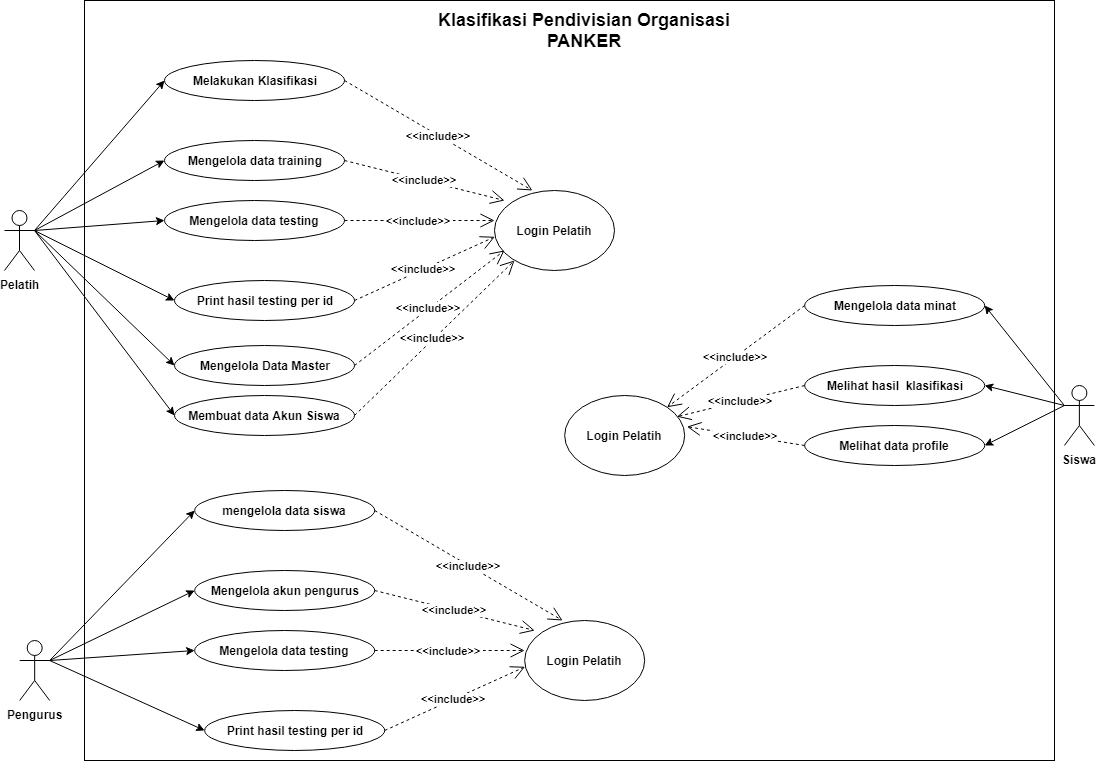
## Perancangan Sistem

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

## Use Case Diagram

Pada use case diagram yang dibuat oleh penulis dalam pembuatan aplikasi klasifikasi pendivisian organisasi panker menggunkan metode naive bayes ini memiliki 3 aktor yaitu pelatih, pengurus, siswa. Aktor pelatih ini menginputkan data training yang akan di jadikan proses sampel untuk naive bayes, aktor pelatih ini juga menginputkan data testing yang akan diproses sehingga menghasilkan output akhir yang menentukan rekomendasi calon pengurus masuk pada divisi yang direkomendasikan oleh sistem. Pelatih ini juga bisa melakukan print pdf dari hasil proses klasifikasi. Aktor pengurus bertugas untuk menambahkan siswa dan akun siswa. Aktor siswa menginputkan minatnya yang akan selanjunya akan diproses oleh pelatih.

Berikut adalah use case diagram dari aplikasi Klasifikasi Pendivisian Organisasi Panker Menggunkan Metode Naive Bayes tedapat pada gambar dibawah ini.



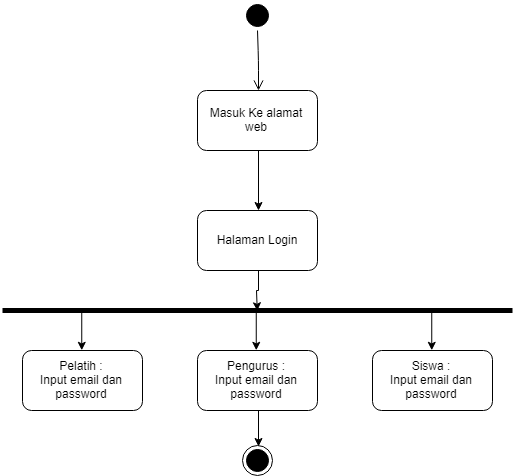
Gambar 3. 2 Use case

## Activity Diagram

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

## Activity Diagram Login

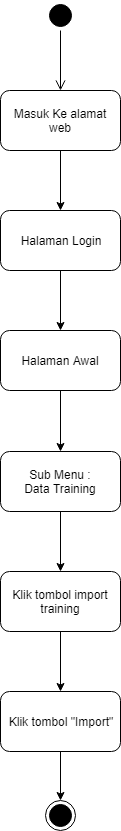
*User* harus melakukan login sesuai data akun masing masing user dan akan di alihkan ke halamana masing-masing untuk dapat menggunakan fitur-fitur yang ada pada sistem.Terdapat 3 *user* yaitu pelatih, pengurus, siswa. Rancangan activity diagram login user terdapat pada Gambar dibawah ini.



Gambar 3. 3 Activity Diagram Login User

## Activity Diagram Pelatih *Import* Data Training

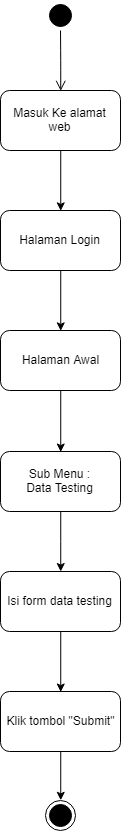
Pelatih Meng*import* data – data training yang akan di olah oleh sistem untuk mencari probabilitas dari sistem klasifikasi sehingga mendapatkan hasil training dari pengolahan data testing oleh sistem. Rancangan activity pelatih diagram import data training terdapat pada Gambar dibawah ini.



Gambar 3. 4 Activity Diagram Import Data Training

## Activity Diagram Pelatih Input Klasifikasi Testing

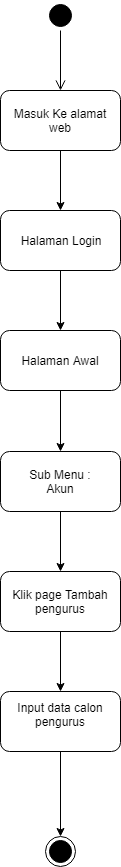
Pelatih akan melakukan klasifikasi data setelah siswa menginputkan data minatnya setelah itu akan di proses oleh pelatih dengan meginputkan data-data nilai untuk dicari rekomendasi divisi yang cocok untuk siswa tersebut. Rancangan activity diagram pelatih input klasifikasi testing terdapat pada Gambar dibawah ini.



Gambar 3. 5 Activity Diagram Klasifikasi Testing

## Activity Diagram Pelatih Tambah Pengurus

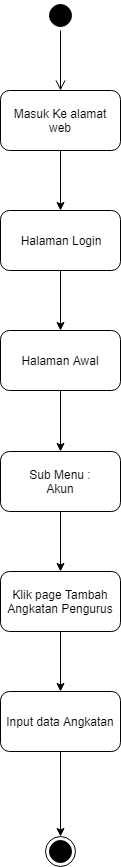
Pelatih menambahkan pengurus dari siswa yang sudah diklasifikasikan rekomendasi divisi yag didapatnya, selanjutnya siswa akan naik menjadi pengurus selajutnya menggantikan pengurus yang lama. Rancangan activity diagram pelatih tambah pengurus terdapat pada Gambar dibawah ini.



Gambar 3. 6 Activity Diagram Klasifikasi Testing

## Activity Diagram Pelatih Tambah Angkatan Pengurus

Pelatih menambahakan angkatan pengurus untuk membedakan pengurus sekarang dengan pengurus selanjutnya atau seterusnya**.** Rancangan activity diagram pelatih tambah angkatan pengurus terdapat pada Gambar dibawah ini.

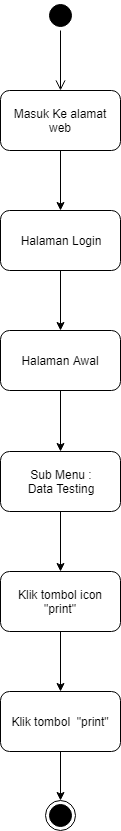


Gambar 3. 7 Activity Diagram Klasifikasi Testing

## Activity Diagram Pelatih Export Pdf Klasifikasi

Setelah data melakukan klasifikasi data akan dimuncul pada viw tabel testing

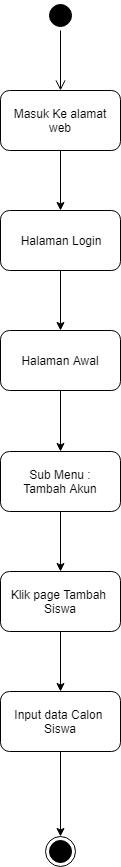
dan export ke pdf untuk hasil klasifikasi per id. Activity diagram Export Pdf Klasifikasi terdapat pada Gambar dibawah ini.



Gambar 3. 8 Activity Diagram Export Pdf

## Activity Diagram Pengurus Tambah Siswa

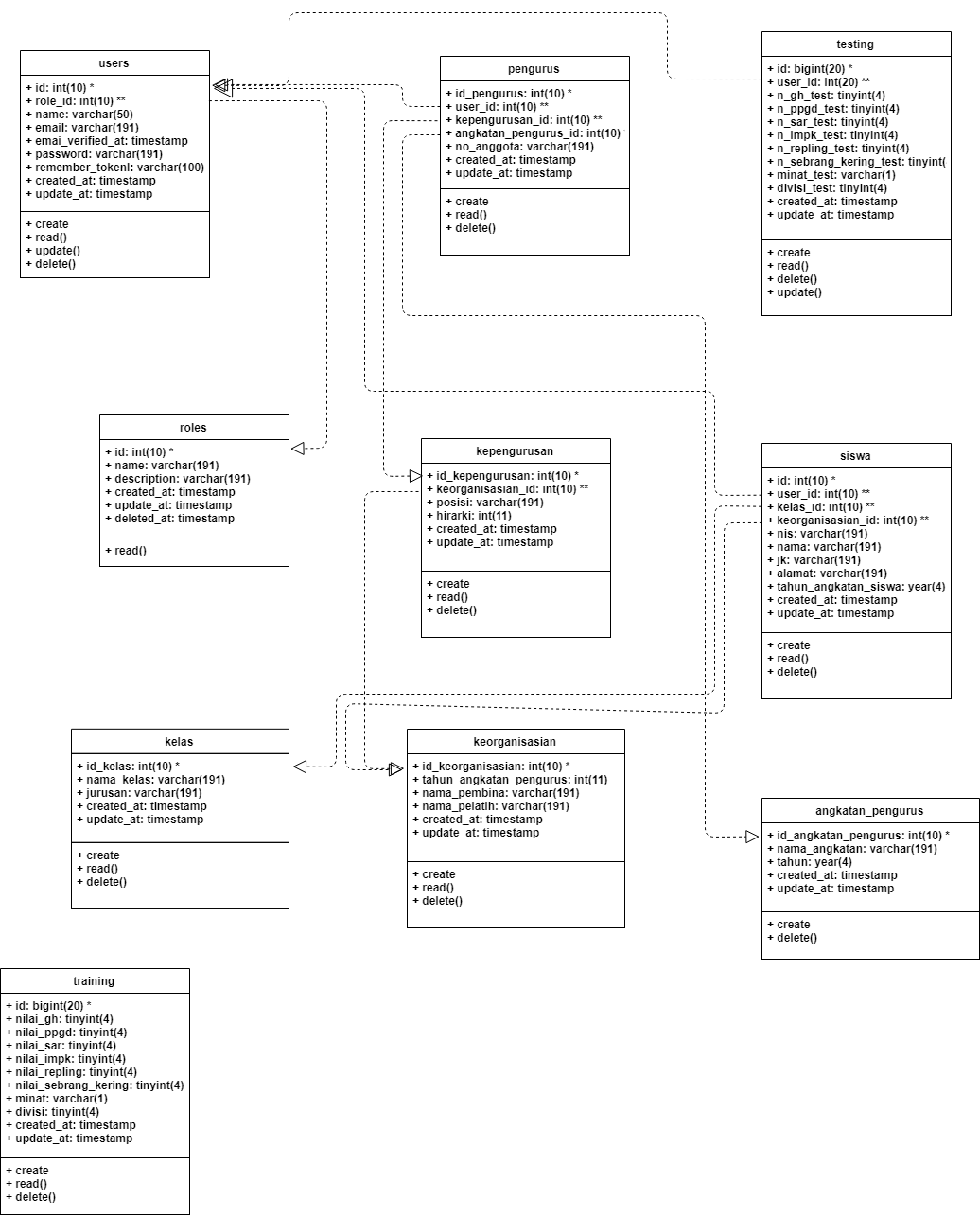
Pengurus menambahkan calon siswa untuk menggantikan angkatan yang sebekumnya sudah menjadi pengurus dan sudah di klasifikasi divisi oleh pengurus terdahulu. Activity diagram Export Pdf Klasifikasi terdapat pada Gambar dibawah ini.



Gambar 3. 9 Activity Diagram Tambah siswa

## *Class* Diagram

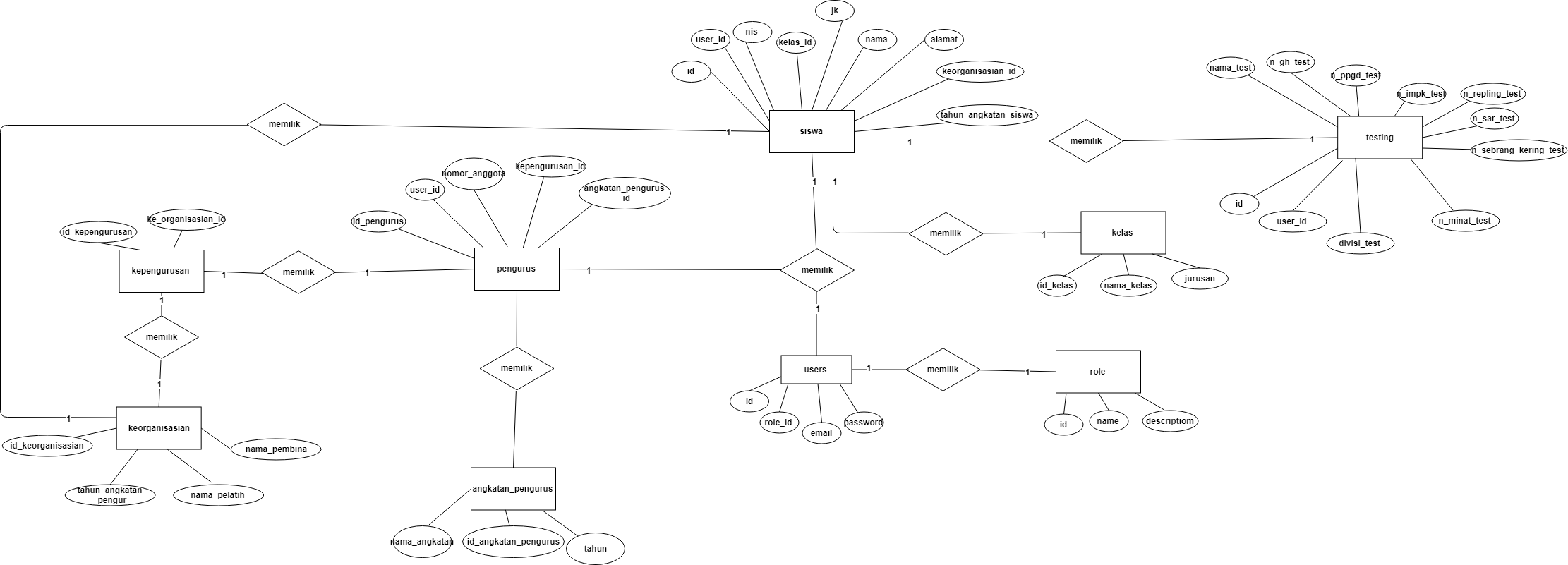
*Class diagram* menunjukan interaksi antar kelas dalam sistem serta memperlihatkan struktur dari sistem berupa kelas, atribut, dan relasi antar kelas. Adapun *class diagram* dari aplikasi ini terdapat pada Gambar ini.



Gambar 3. 10 Class Diagram

## *Entity Relationship Diagram* (ERD)

*Entity Relationship Diagram* (ERD)menunjukan struktur tabel dan memperlihatkan relasi antar tabel dalam *database* yang digunakan dalam pembuatan aplikasi sistem informasi tugas akhir ini. Adapun class diagram dari aplikasi ini terdapat pada Gambar dibawah ini.

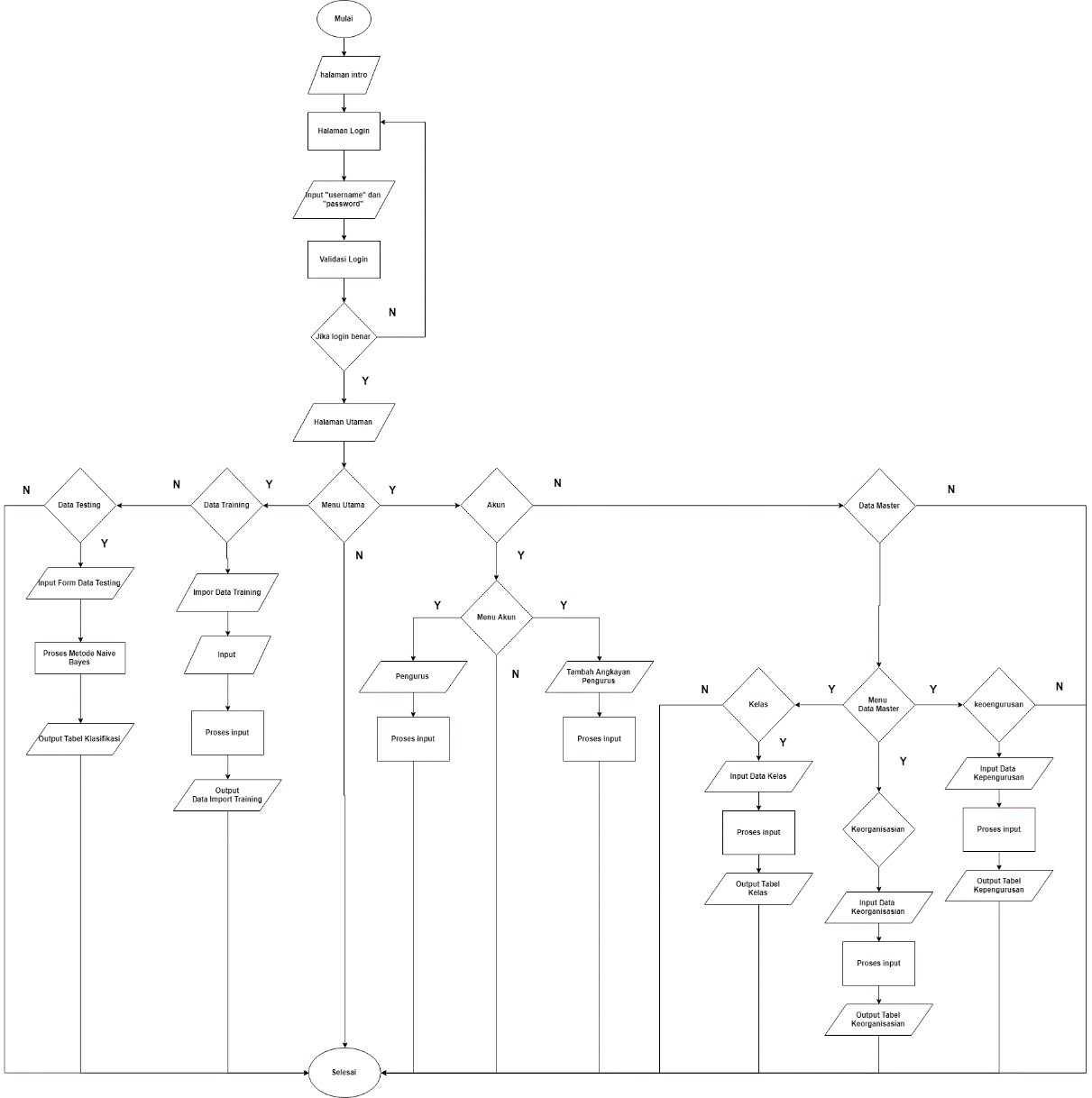


Gambar 3. 11 Entity Relationship Diagram (ERD)

## *Flowchart* Aplikasi

1. Flowchart Pelatih

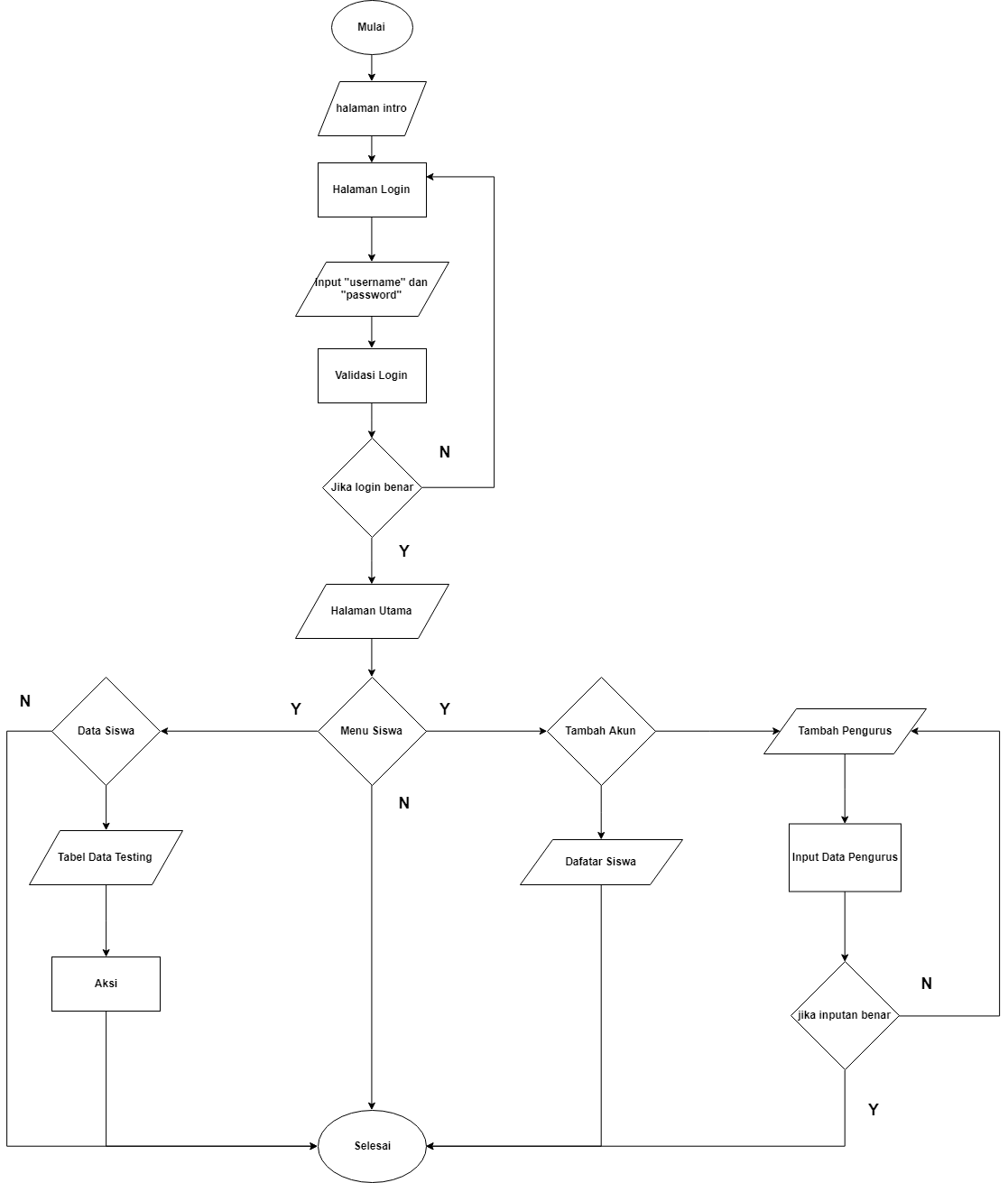
Rancangan *flowchart* Pelatih memiliki 4 menu utama yaitu menu data training untuk menginputkan data training sebagai rancangan data model naive bayes, menu data testing untuk mengklasifikasikan siswa untuk mencari rekomendasi minat dan data-data siswa yang sudah diklasifikasi oleh pelatih, menu akun yaitu untuk menanmbahkan pengurus yang lama dengan pengurus yang baru dengan memilih siswa yang akan di angkat menjadi pengurus, menu data master digunakan untuk menambahkan data kelas, kepengurusan. Adapun rancangan bisa dilihat pada gambar dibawah ini.



Gambar 3. 12 Gambar FLowchart Pelatih

1. ***Flowchart* Pengurus**

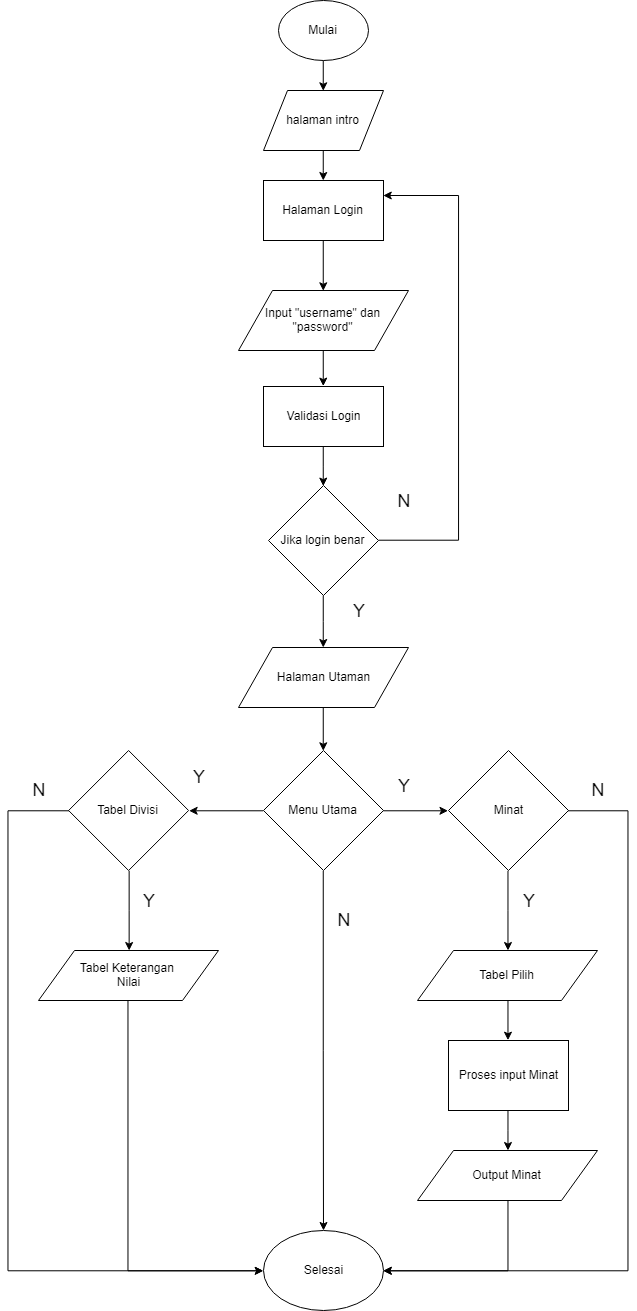
Rancangan *flowchart* pengurus memiliki 3 menu yaitu menu data siswa untuk melihat data-data siswa yang sudah diklasifikasikan oleh pelatih, menu tambah akun yaitu menu yang berisikan data-data siswa yang sudah dibuatkan akun siswa dan tambah akun untuk menambahkan akun siswa baru sekaligus menambahkan biodata siswa tersebut. Adapun rancangan bisa dilihat pada gambar dibawah ini.



Gambar 3. 13 FLowchart Pengurus

1. ***Flowchart* Siswa**

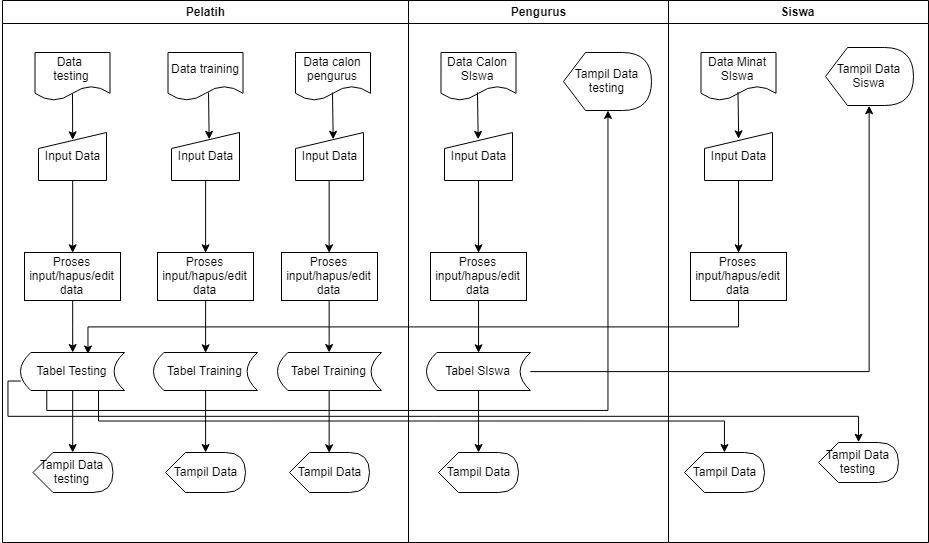
Rancangan *flowchart* meliki 1 halama yang menampilkan biodata siwa dan kererangan minat.dihalaman awal tersebut terdapat 2 paging berisikan page tabel keterangan niali-nilai setiap atribute dan page pilih minat siswa tersebut yang merupakan pilihan untuk diklasifikasikan oleh pelatih. Adapun rancangan bisa dilihat pada gambar dibawah ini.



Gambar 3. 14 Flowchart Siswa

1. ***Flowcart* Sistem**

*Flowchart system* dari pembuatan aplikasi klasifikasi pendivisian organisasi PANKER , yaitu suatu bagan yang memperlihatkan urutan *procedure* dan proses dari beberapa *filed* di dalam beberapa fitur tertentu. Adapun *flowchart system* dari pembuatan aplikasi ini dapat dilihat pada dibawah ini.



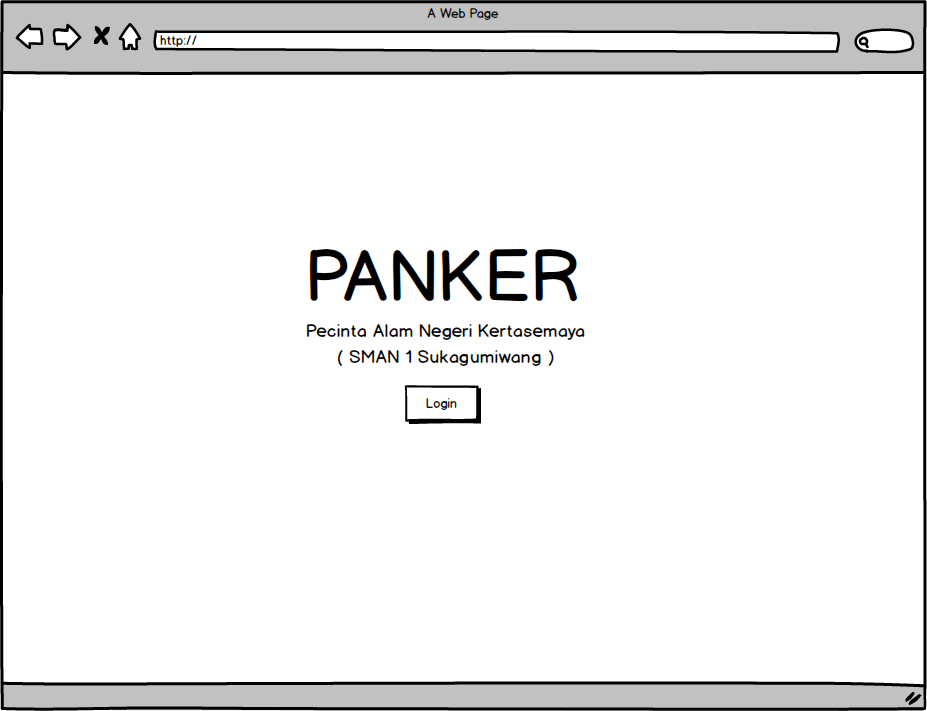
Gambar 3. 15 Flowchart Sistem

## Perancangan Awal Antar Muka

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

1. **Rancangan Halaman Awal**

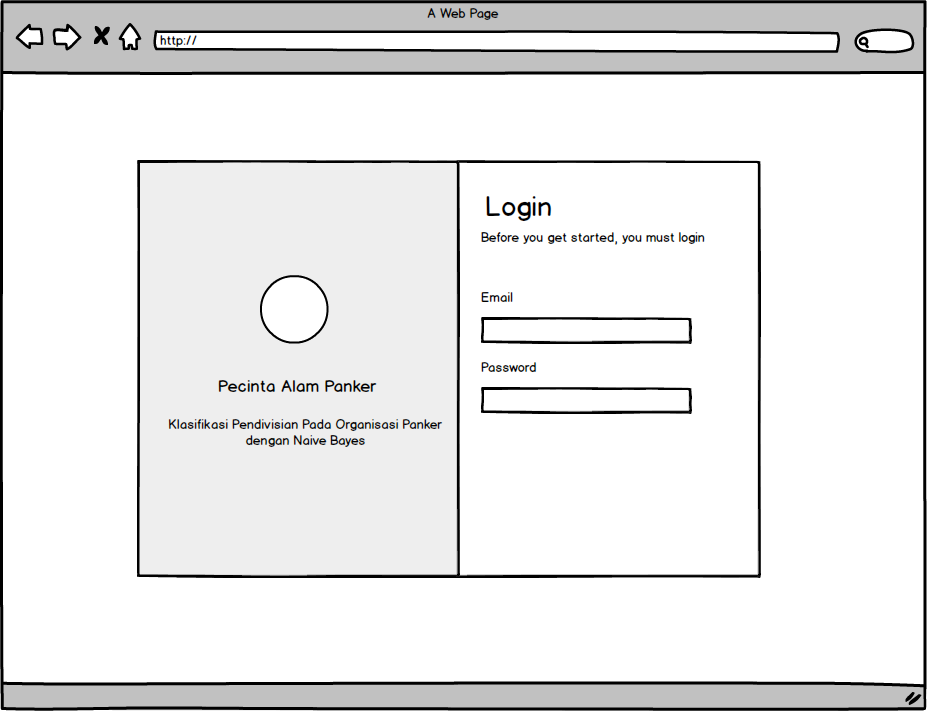
Halaman awal merupakan halaman yang pertama kali ditampilkan ketika *user* mengakses url. Adapun hasil rancangan halaman awal aplikasi terdapat pada Gambar berikut.



Gambar 3. 16 Rancangan Halaman Awal

1. **Rancangan Halaman Login**

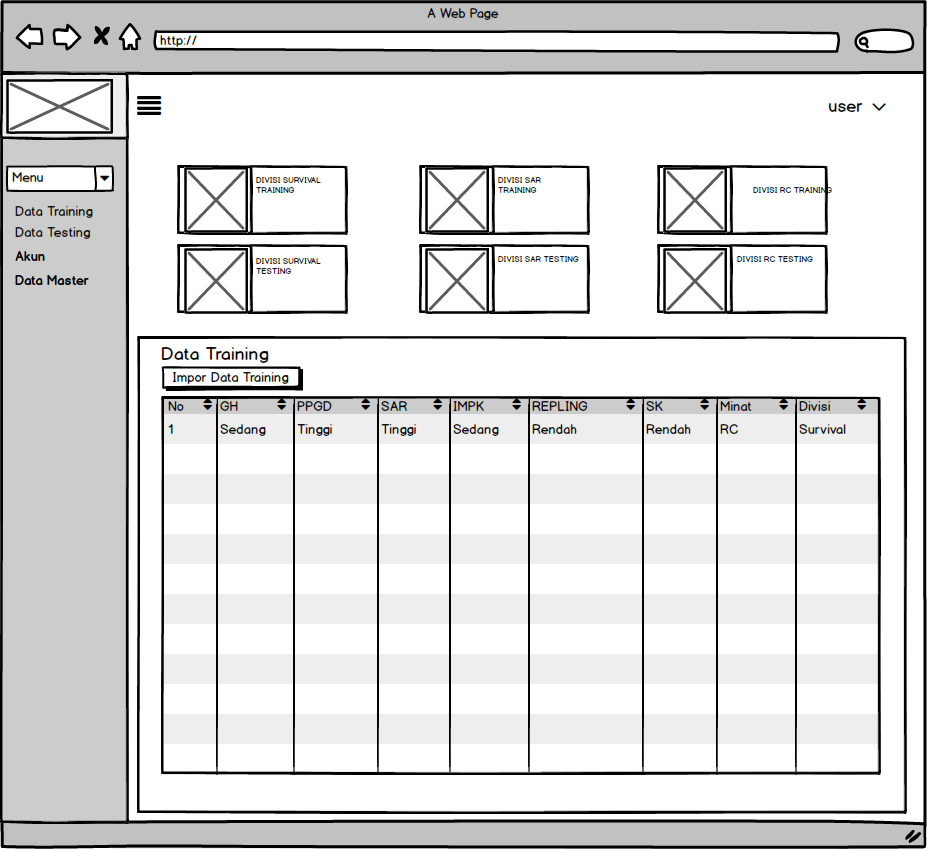
Halaman login merupakan halaman yang digunakan ketika *user* akan melakukan login ke dalam aplikasi sesuai dengan role masing-masing jika role pelatih makan akan dialihkan pada halaman pelatih jika rolenya sebagai pengurus makan akan dialihkan ke halaman pengurus dan jika role sebagai siswa makan akan di alihkan ke halaman siswa . Adapun rancangan seperti gambar dibawah ini.



Gambar 3. 17 Rancangan Halaman Login

1. **Rancangan Halaman Pelatih Tampil Data Training**

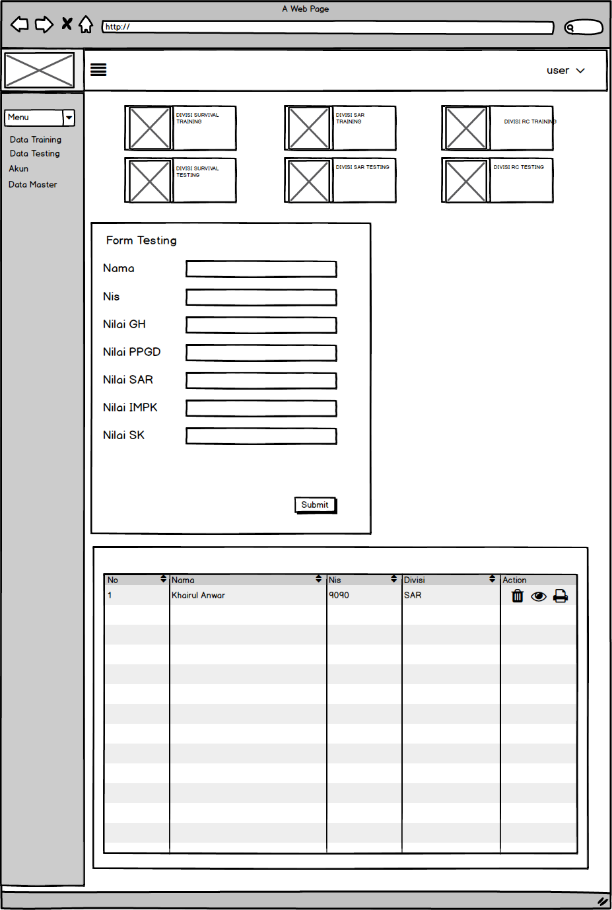
Halaman tampil data testing merupakan halaman utama dari *user* setelah melakukan proses *login*. Halaman Tampil data taining merupakan suatu halaman yang menampilkan data – data yang akan di proses oleh metode naive bayes Adapun rancangan seperti gambar dibawah ini.



Gambar 3. 18 Rancangan Halaman Tampil Data Training

1. **Rancangan Halaman Pelatih Tampil Data Testing**

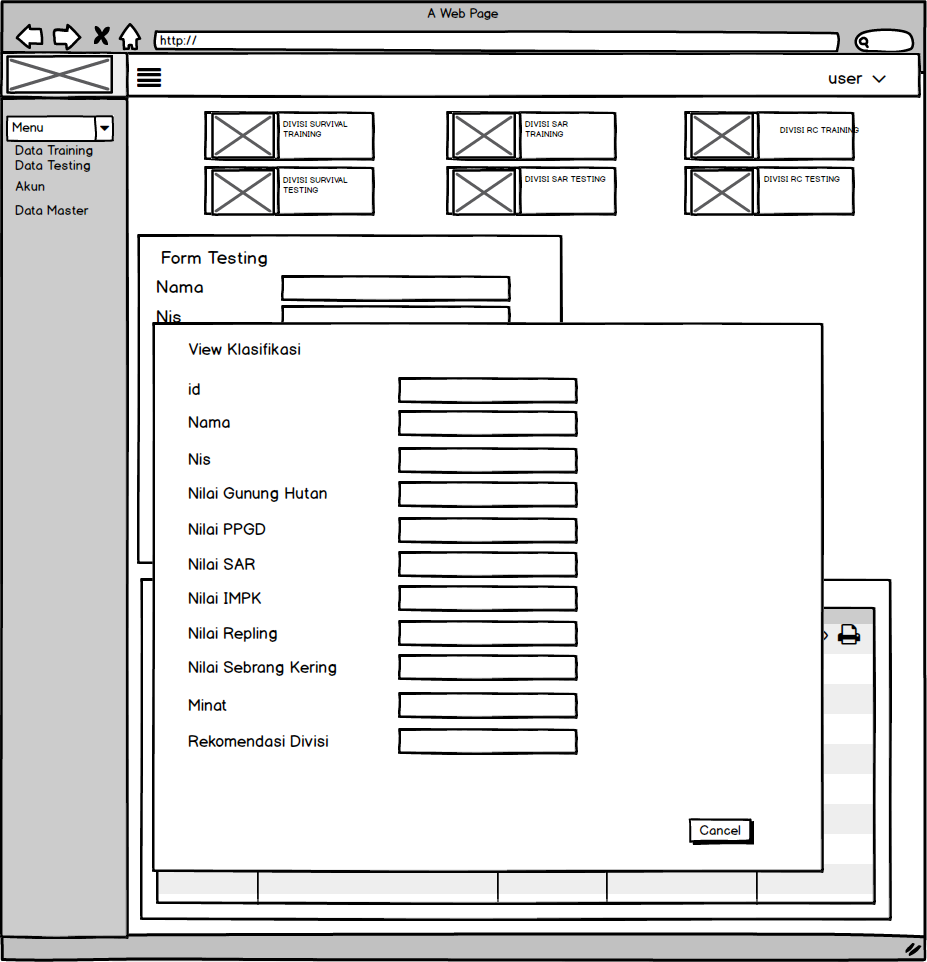
Halaman tampil data Testing merupakan suatu halaman yang menampilkan hasil pengolahan metode naive bayes yang di lakukan oleh *user* untuk mendapatkan hasil rekomendasi divisi dari nilai nilai atribute yang dimasukan oleh user, selanjutnya hasil akan ditampilkan kedalam tabel berikut dengan *action*. Adapun rancangan seperti gambar dibawah ini..



Gambar 3. 19 Rancangan Halaman Tampil Data Testing

1. **Rancangan Pelatih View Data Testing**

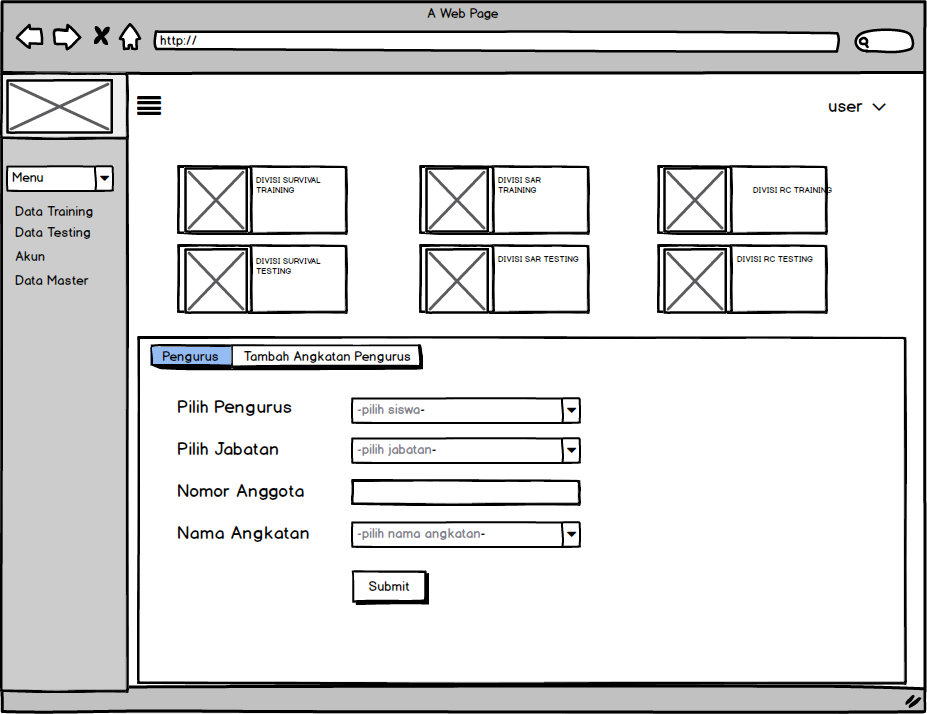
Halaman view data testing merupakan suatu modal dimana user dapat melihat seluruh data dari hasil proses naive bayes secara per-id nya masing masing. Adapun rancangan seperti gambar dibawah ini.



Gambar 3. 20 Rancangan Halaman View Data Testing

1. **Rancangan Halaman Pelatih Tambah Pengurus**

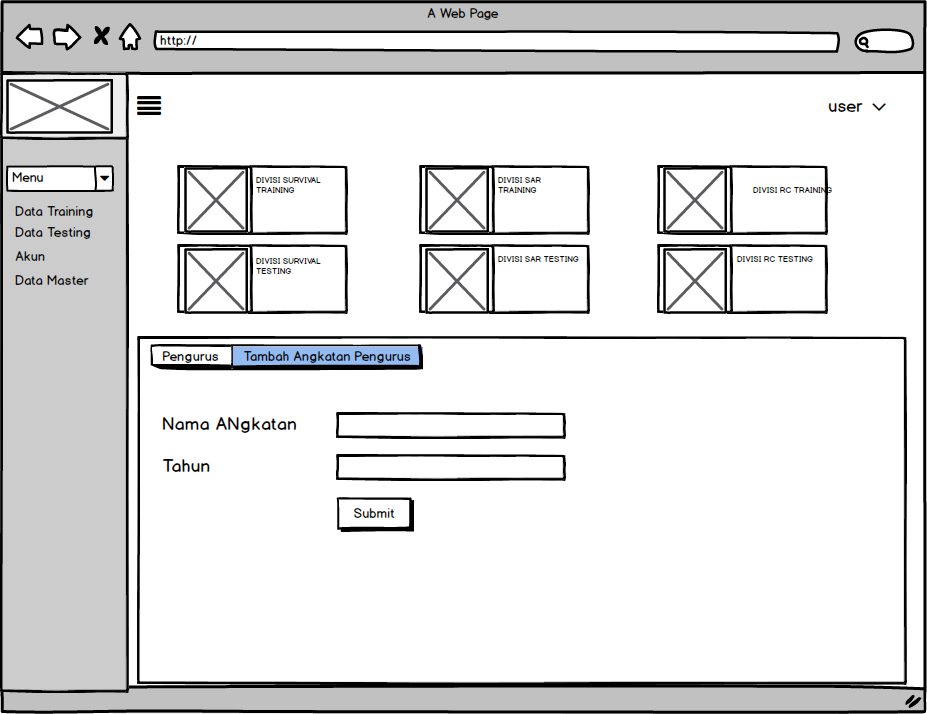
Halaman Pelatih Tambah pengurus merupakan suatu halaman yang menampilakn fitur penambahan pengurus yang diambil dari data-data data siswa yang sudah melakukan klasifikasi pendivisian oleh pelatih . Adapun rancangan seperti gambar dibawah ini.



Gambar 3. 21 Rancangan Halaman Pelatih Tambah Pengurus

1. **Rancangan Halaman Pelatih Tambah Angkatan Pengurus**

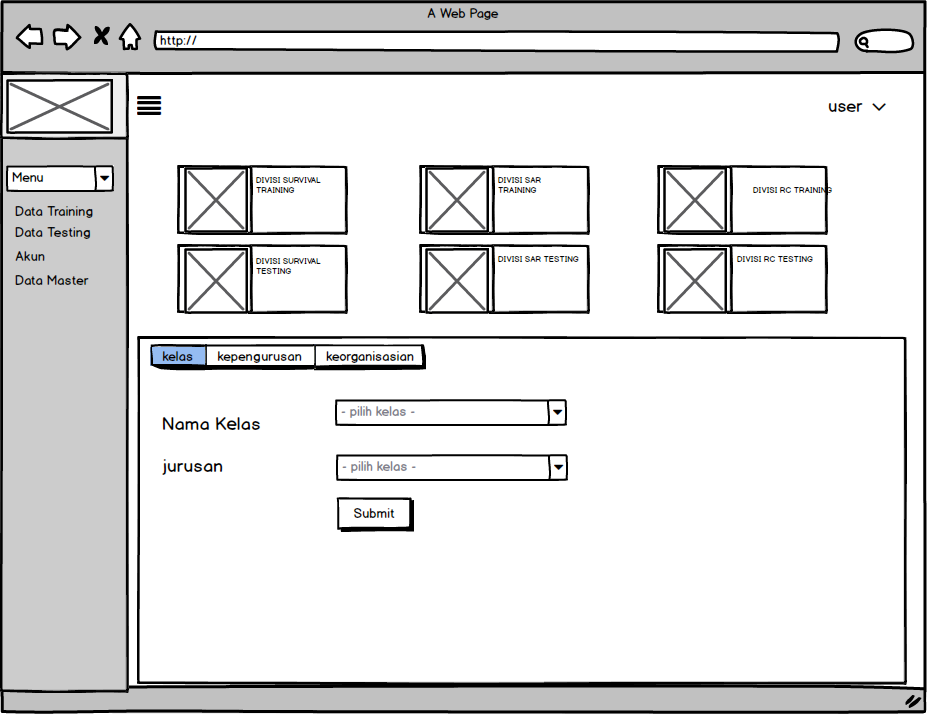
Halaman pelatih tambah angkatan berisikan tentang fitur pelatih untuk menambahkan angkatan baru dari pengurus baru. Adapun rancangan seperti gambar dibawah ini.



Gambar 3. 22 Rancangan Halaman Pelatih Tambah Angkatan Pengurus

1. **Rancangan Halaman Pelatih Data Master Kelas**

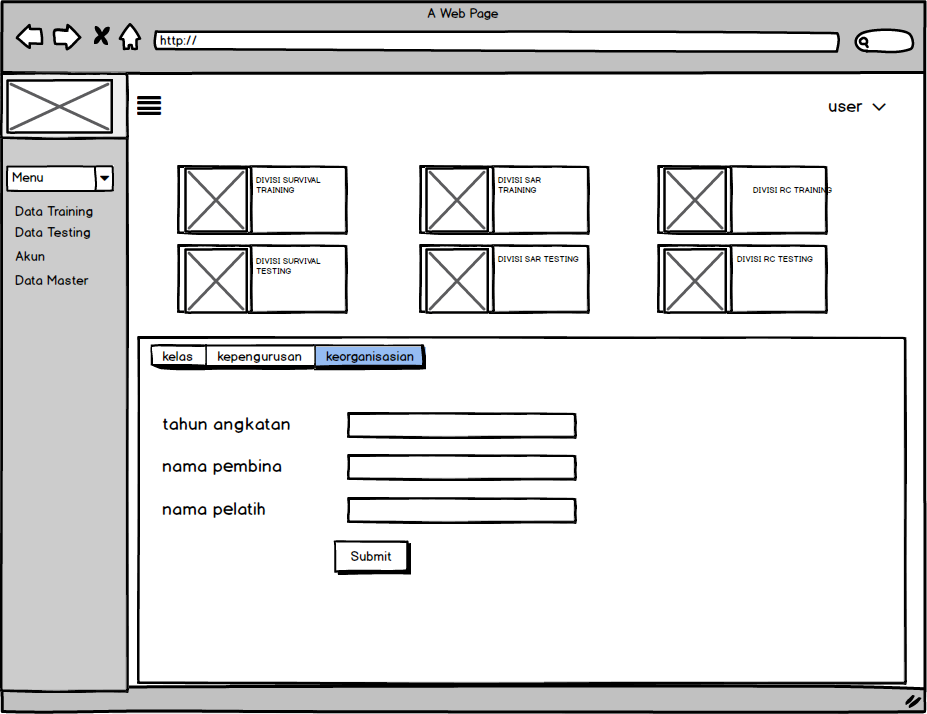
Halaman data master kelas merupakan halaman untuk menambahkan kelas dan jurusan pada input data siswa baru. Adapun rancangan seperti gambar dibawah ini.



Gambar 3. 23 Rancangan Halaman Pelatih Data Master Kelas

1. **Rancangan Halaman Pelatih Data Master Keorganisasian**

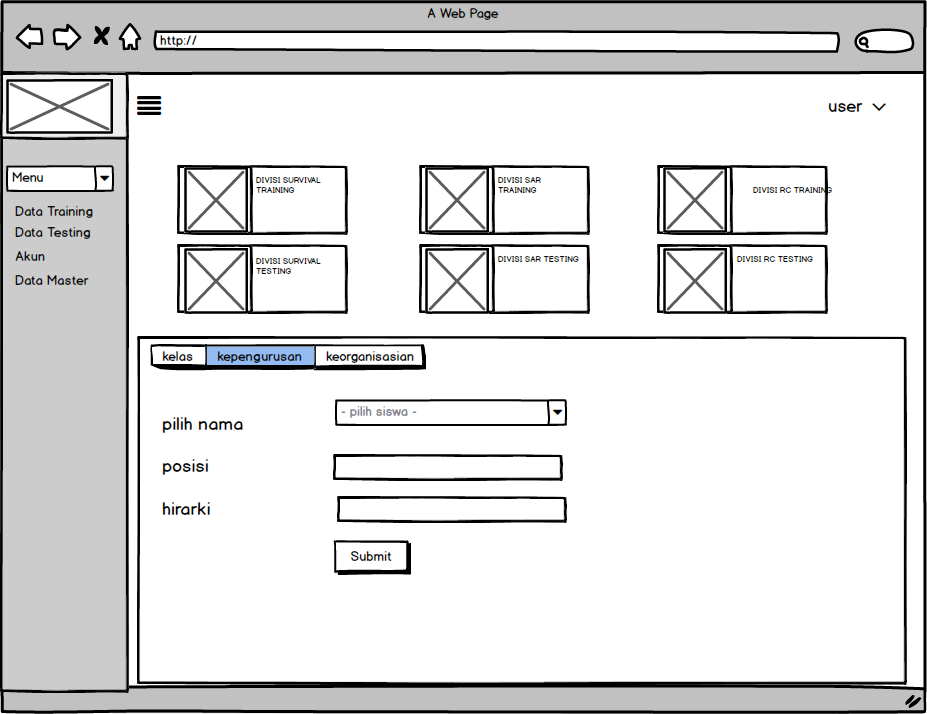
Halaman data master keorganisasian merupakan halamn yang menambahakan tahun angkatan dan nama pembina dan pelatih . adapun rancangan seperti gambar dibawah ini.



Gambar 3. 24 Rancangan Halaman Pelatih Data Master Keorganisasian

1. **Rancangan Halaman Pelatih Data Master Kepengurusan**

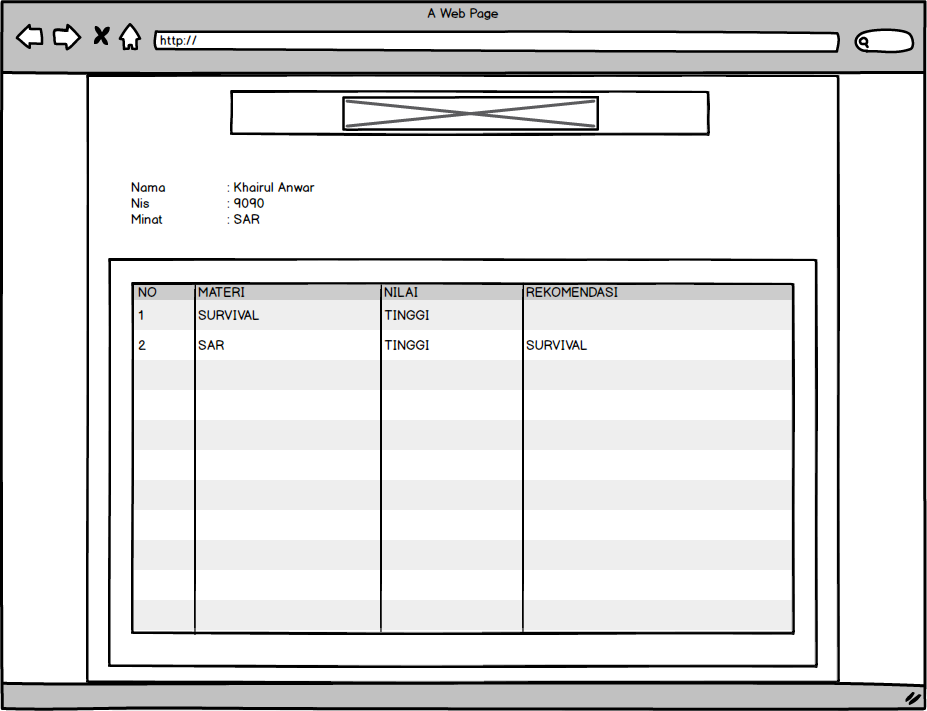
Rancangan data master kepengurusan merupakan halaman yang berisikan inputan untuk menambahkan status pengurus pada angkatan organisasi. Adapun rancangan seperti gambar dibawah ini



Gambar 3. 25 Rancangan Halaman Pelatih Data Master Kepengurusan

1. **Rancangan Halaman Pelatih Tampil Export PDF**

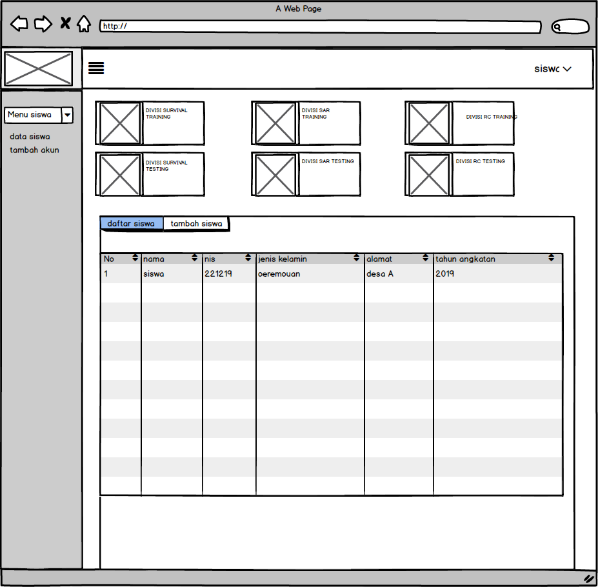
Halaman tampil export Pdf bertujuan untuk mencetak hasil dari masing masing data yang sudah di proses oleh naive bayes yang selanjutnya akan dijadikan untuk diterbitkan laporan oleh organisasi untuk dilihat hasilnya oleh calon pengurus. Adapun rancangan halaman tampil data pengajuan TA terdapat pada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



Gambar 3. 26 Rancangan Halaman Pelatih Tampil Export Pdf

1. **Rancangan Halaman Pengurus Data Siswa**

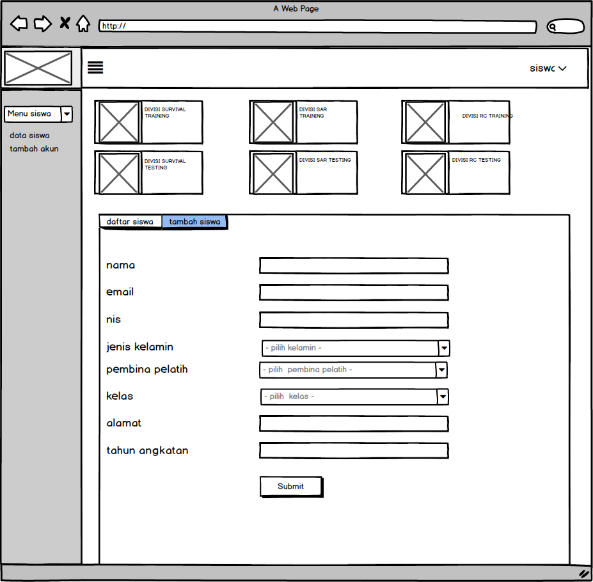
Rancang halman pengurus data siswa erupakan rancangan halaman yang berisikan data data siswa yang sudah ditambahkan maupun baru ditambahkan dari daftar siswa dan tambah siswa untuk menambahkan siswa. Adapun rancangan seperti gambar dibawah ini.



Gambar 3. 27 Rancangan Halaman Pengurus Data Siswa

1. **Rancangan Halaman Pengurus Tambah Akun**

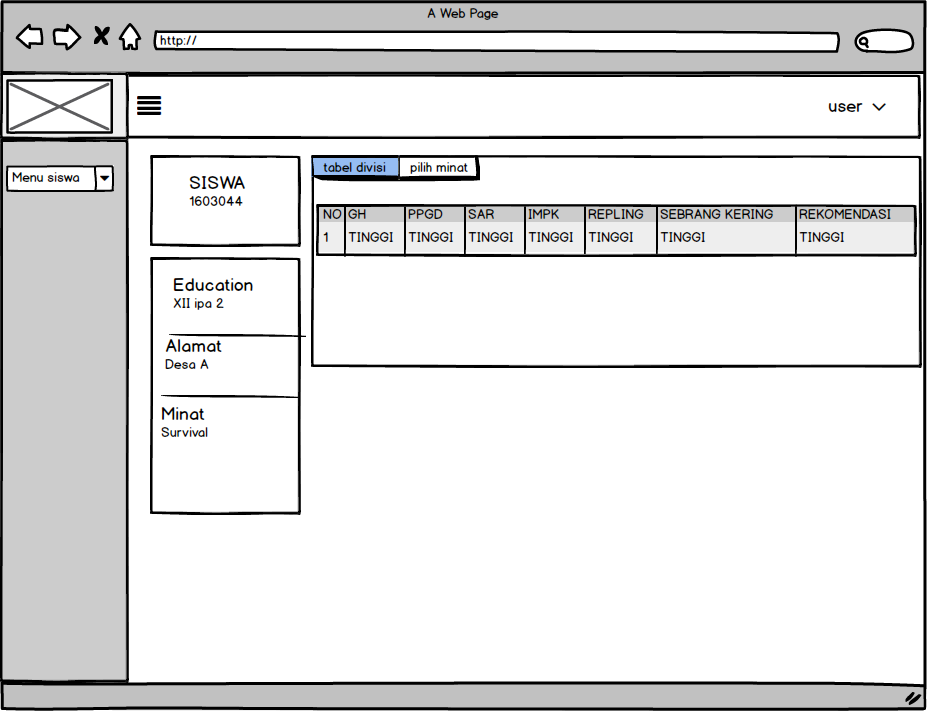
Halaman Tambah aku merupakn halaman untuk menambahkan data-data sekaligus membuat akun siswa yang selanjutnya akan dipakai oleh siswa untuk menginputkan minat dan mengetahui hasil klasifikasi dari semua data data tersebut. Adapun rancangan seperti gambar dibawah ini.



Gambar 3. 28 Rancangan Halaman Pengurus Tambah Akun

1. **Rancangan Halaman Siswa**

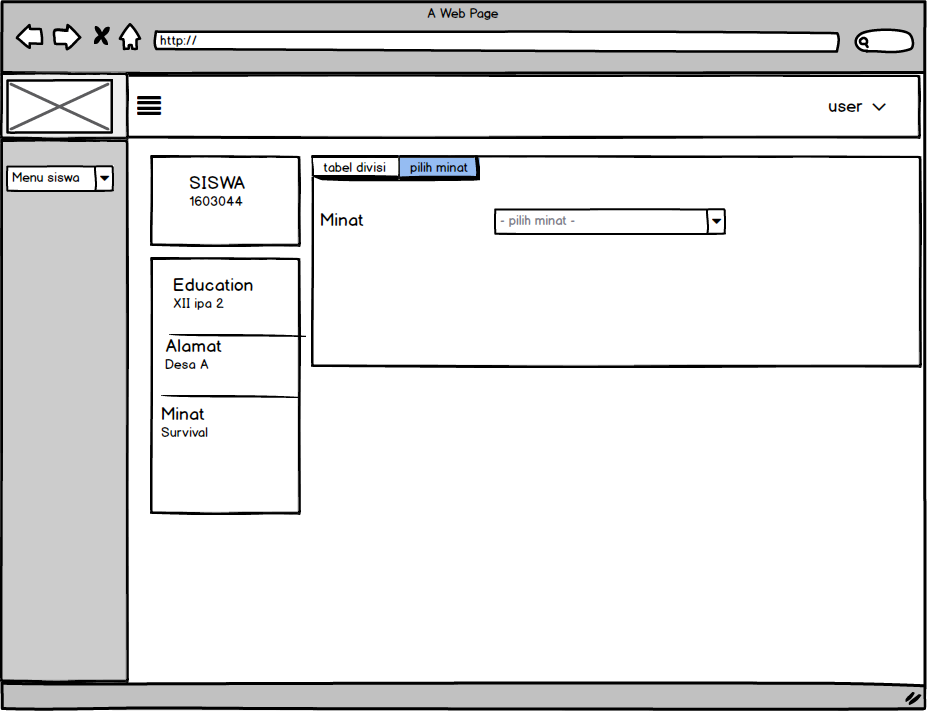
Racangan halaman siswa merupakan halaman yang berisikan data data siswa seperti nama nim, kelas, alamat, minat dan tabel divsi yang berisikan data-data keterangan nilai peminatan. Adapun rancangan seperti gambar dibawah ini.



Gambar 3. 29 Rancangan Halaman Siswa

1. **Rancnagan Halaman Siswa Pilih Minat**

Rancangan halaman siswa pilih minat merupakan halaman untuk siswa menginputkan data minatnya yang akan selanjutnya diproses oleh peatih untuk mengetahui hasil rekomendasi yang cocok untuk siswa tersebut. Adapu rancangan halaman seperti gambar dibawah ini.



Gambar 3. 30 Rancangan Halaman Siswa Pilih Minat

## Perancangan Database

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

## 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) |
|  | role\_id | int(10) | Menyimpan id role |
| 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 *insert* data |
| 7 | updated\_at | Timestamp | Menyimpan waktu *update* data |

## 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 | nilai\_gh | tinyint(4) | Menyimpan nilai gh |
| 3 | nilai\_ppgd | tinyint(4) | Menyimpan nilai ppgd |
| 4 | nilai\_sar | tinyint(4) | Menyimpan nilai sar |
| 5 | nilai\_impk | tinyint(4) | Menyimpan nilai impk |
| 6 | nilai\_repling | tinyint(4) | Menyimpan nilai repling |
| 7 | nilai\_sebrang\_kering | tinyint(4) | Menyimpan nilai sebrang kering |
| 8 | Minat | tinyint(4) | Menyimpan nilai minat |
| 9 | Divisi | tinyint(4) | Menyimpan nilai divisi |
| 10 | created\_at | Timestamp | Menyimpan waktu *insert* data |
| 11 | updated\_at | Timestamp | Menyimpan waktu *update* data |

## 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 | user\_id | int(11) | Menyimpan user id |
| 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 *insert* data |
| 13 | updated\_at | Timestamp | Menyimpan waktu *update* data |

## Tabel siswa

Tabel siswa merupakan tabel untuk menyimpan data-data siswa sekaligus sebagai. Adapun rancangan tabel bisa dilihat pada tabel dibawah ini.

Tabel 3. 6 Tabel siswa

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Nama Kolom** | **Tipe Data** | **Keterangan** |
| 1 | id | int(10) | menyimpan id (PK) |
| 2 | user\_id | int(10) | menyimpan user\_id (PK) |
| 3 | kelas\_id | int(10) | menyimpan kelas\_id (PK) |
| 4 | keorganisasian\_id | int(10) | menyimpan keorganisasian\_id |
| 5 | nis | varchar(191) | menyimpan nis |
| 6 | nama | varchar(191) | menyimpan nama |
| 7 | jk | varchar(191) | menyimpan jenis kelamin |
| 8 | alamat | varchar(191) | menyimpan alamat |
| 9 | tahun\_angkatan\_siswa | year(4) | menyimpan tahun angkatan siswa |
| 10 | created\_at | timestamp | Menyimpan waktu *insert* data |
| 11 | update\_at | timestamp | Menyimpan waktu *update* data |

## Tabel roles

Rancangan tabel roles merupakan tabel yang akan menyimpan roles dari akun akun yang user. Adapun rancangan tabel bisa dilihat pada tabel dibawah ini.

Tabel 3. 7 Tabel roles

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Nama Kolom** | **Tipe Data** | **Keterangan** |
| 1 | id | int(10) | menyimpan id (PK) |
| 2 | name | varchar(191) | menyimpan name role |
| 3 | desciption | varchar(191) | menyimpan desciption role |
| 4 | created\_at | timestamp | Menyimpan waktu *insert* data |
| 5 | update\_at | timestamp | Menyimpan waktu *update* data |
| 6 | deleted\_at | timestamp | Menyimpan waktu *delete* data |

## Tabel pengurus

Rancnagan tabel pengurus meruppakan tabel untuk menyimpan data pengurus . adapun rancangn tabel bisa dilihat pada tabel dibawah ini.

Tabel 3. 8 Tabel pengurus

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Nama Kolom** | **Tipe Data** | **Keterangan** |
| 1 | id | int(10) | menyimpan id (PK) |
| 2 | user\_id | int(10) | menyimpan user\_id (PK) |
| 3 | kelas\_id | int(10) | menyimpan kelas\_id (PK) |
| 4 | keorganisasian\_id | int(10) | menyimpan keorganisasian\_id |
| 5 | nis | varchar(191) | menyimpan nis |
| 6 | nama | varchar(191) | menyimpan nama |
| 7 | jk | varchar(191) | menyimpan jenis kelamin |
| 8 | alamat | varchar(191) | menyimpan alamat |
| 9 | tahun\_angkatan\_siswa | year(4) | menyimpan tahun angkatan siswa |
| 10 | created\_at | timestamp | Menyimpan waktu *insert* data |
| 11 | update\_at | timestamp | Menyimpan waktu *update* data |

## Tabel kepengurusan

Rancangan tabel kepengurusan merupakan rancangan tabel untuk meyimpan kepengurusan dan mengambil data keorganisasian\_id pada tabel keorganisasian. Adapun rancangan tabel bisa dilihat pada tabel dibawah ini.

Tabel 3. 9 Tabel kepengurusan

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Nama Kolom** | **Tipe Data** | **Keterangan** |
| 1 | id\_kepengurusan | int(10) | menyimpan id kepengurusan |
| 2 | keorganisasian\_id | int(10) | menyimpan keorganisasian id |
| 3 | posisi | varchar(191) | menyimpan posisi |
| 4 | hirarki | int(11) | menyimpan hirarki |
| 5 | created\_at | timestamp | Menyimpan waktu *insert* data |
| 6 | updated\_at | timestamp | Menyimpan waktu *update* data |

## Tabel keorganisasian

Rancangan tabel keorganisasian merupakan tabel untuk menyimpan data keorganisasian yang akan di relasi dengan tabel kepengurusan.adapun rancangan tabel bisa dilihat pada tabel dibawah ini.

Tabel 3. 10 Tabel keorganisasian

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Nama Kolom** | **Tipe Data** | **Keterangan** |
| 1 | id\_keorganisasian | int(10) | menyimpan id keorganisasian |
| 2 | tahun\_angkatan\_pengurus | int(10) | menyimpan tahun angkatan pengurus |
| 3 | nama\_pembina | varchar(191) | menyimpan nama pembina |
| 4 | nama\_pelatih | varchar(191) | menyimpan nama pelatih |
| 5 | created\_at | timestamp | Menyimpan waktu *insert* data |
| 6 | updated\_at | timestamp | Menyimpan waktu *update* data |

## Tabel Kelas

Rancagan tabel kelas merupakan rancangan tabek yang menyimpan data kelas yang akan di relasasikan ke kelas id pada tabel siswa. Adapun rancangan tabek bisa dilihat pada tabel dibawah ini

Tabel 3. 11 Tabel kelas

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Nama Kolom** | **Tipe Data** | **Keterangan** |
| 1 | id\_kelas | int(10) | menyimpan id kelas |
| 2 | nama\_kelas | varchar(191) | menyimpan nama kelas |
| 3 | jurusan | varchar(191) | menyimpan jurusan |
| 4 | created\_at | timestamp | Menyimpan waktu *insert* data |
| 5 | updated\_at | timestamp | Menyimpan waktu *update* data |

## Tabel angkatan\_pengurus

Rancangan tabel kepengurusan merupakan rancangan tabel yang akan menyimpan data-data seperti id\_angkatan\_pengurus, nama\_angkatan, dan tahun. Adapun rancnagan tabel bida dilihat pada tabel dibawah ini.

Tabel 3. 12 Tabel angkatan\_pengurus

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Nama Kolom** | **Tipe Data** | **Keterangan** |
| 1 | id\_angkatan\_pengurus | int(10) | menyimpan id angkatan pengurus |
| 2 | nama\_angkatan | varchar(191) | menyimpan nama angkatan pengurus |
| 3 | tahun | year(4) | menyimpan tahun angkatan pengurus |
| 4 | created\_at | timestamp | Menyimpan waktu *insert* data |
| 5 | updated\_at | timestamp | Menyimpan waktu *update* data |

# BAB IV HASIL DAN PEMBAHASAN

## Hasil Penelitian

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

## Hasil Pustaka

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

## Implementasi Desain Antar Muka

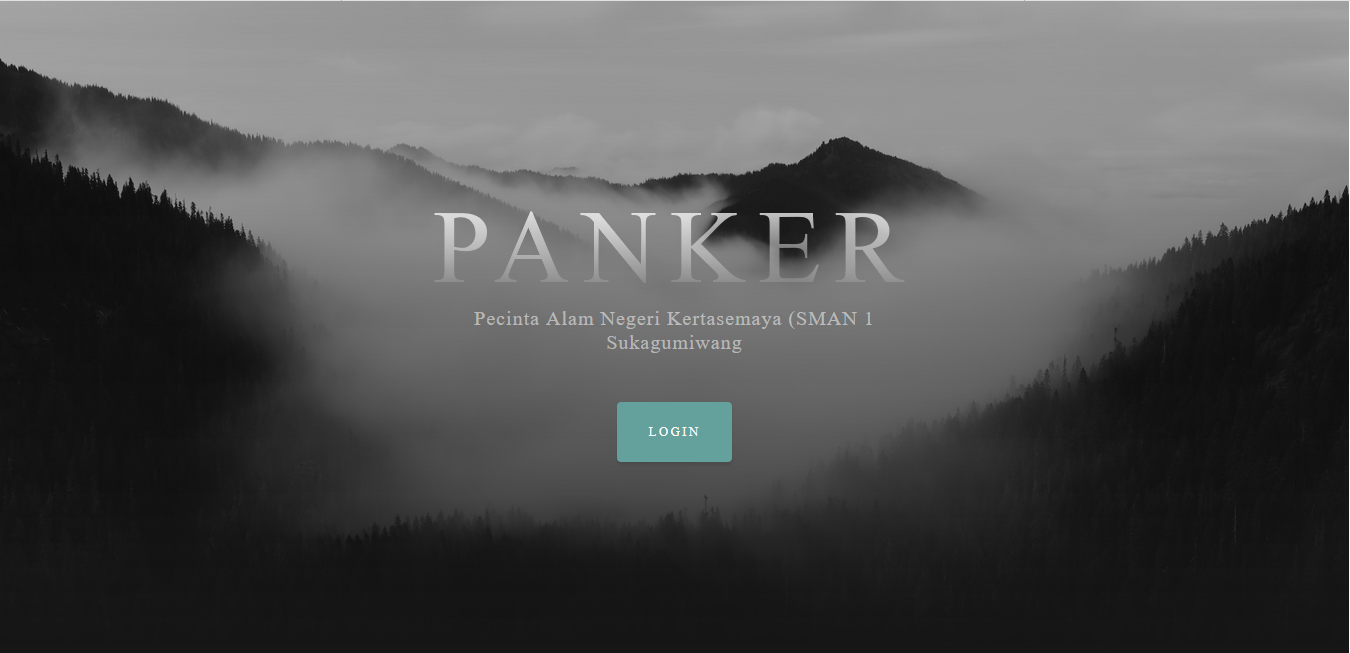
Tahap implemtasi merupakan tahap penerjemah perancangan berdasarkan hasil analisis kedalam suatu bahasa pemrograman yang digunakan serta penerapan perangkat lunak yang dibangun pada lingkungan yang sesungguhnya. Tujuan implemtasi untuk menerapkan perancangan yang telah dilakukan terhadap aplikasi, sehingga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

## 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 berkembanganya aplikasi yang telah dibangun. Adapun pembahasan implementasi desain berikut:

## Implementasi Hasil Tampilan Intro Aplikasi

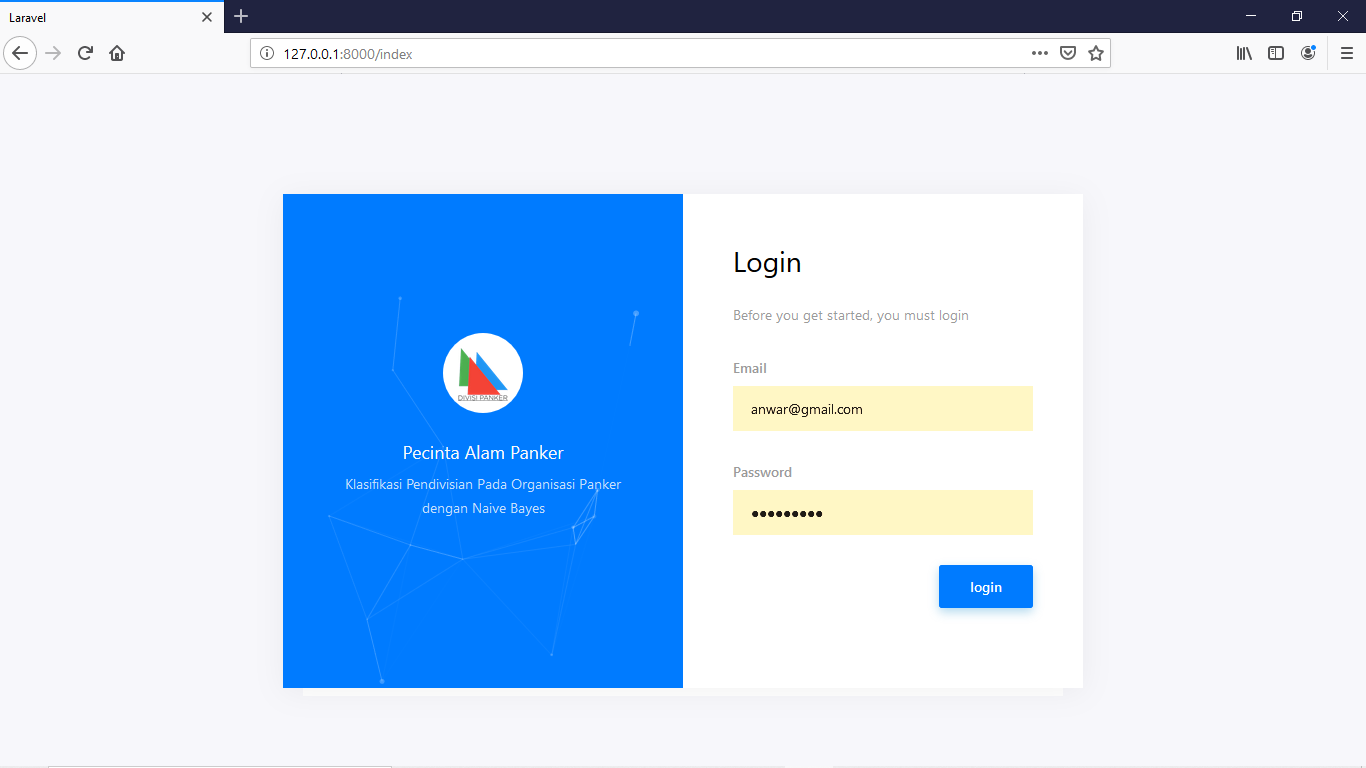
Halaman intro ini merupakan halaman awal sebelum menuju halaman login, halaman ini diakses ketika user mengunjungi url apliaksi tersebut.



Gambar 4. 1 Tampilan Halaman Awal

## Implementasi Hasil Tampilan Login

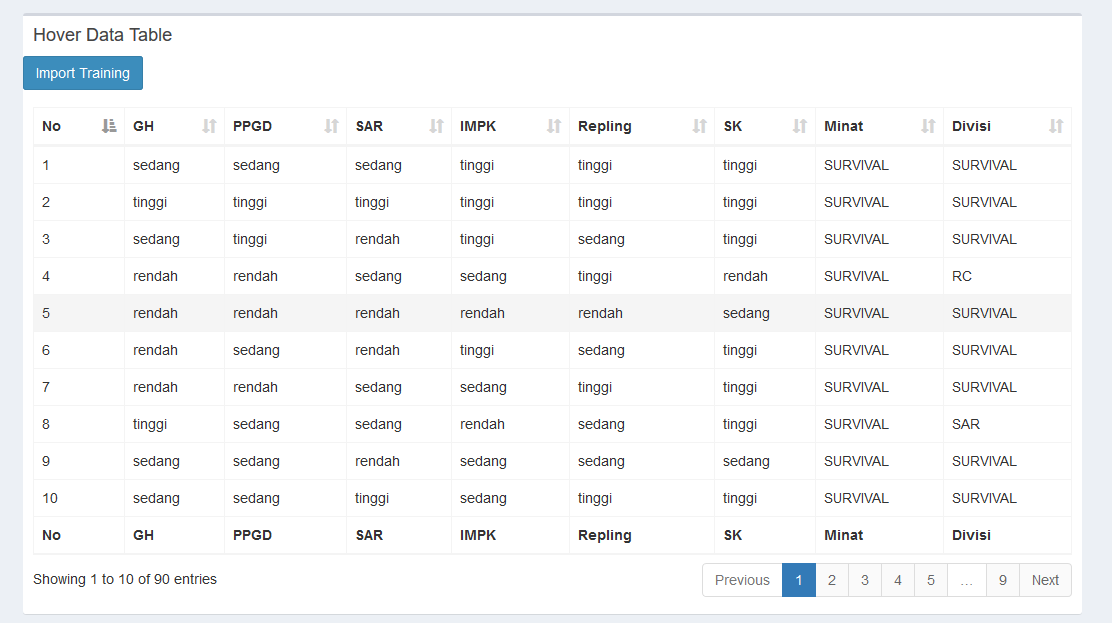
Halaman Login ini merupakan halaman ketika user akan masuk ke dalam aplikasi untuk melakukan semua aktifitas



Gambar 4. 2 Tampilan Halaman Login

## Implementasi Hasil Tampilan Data Training

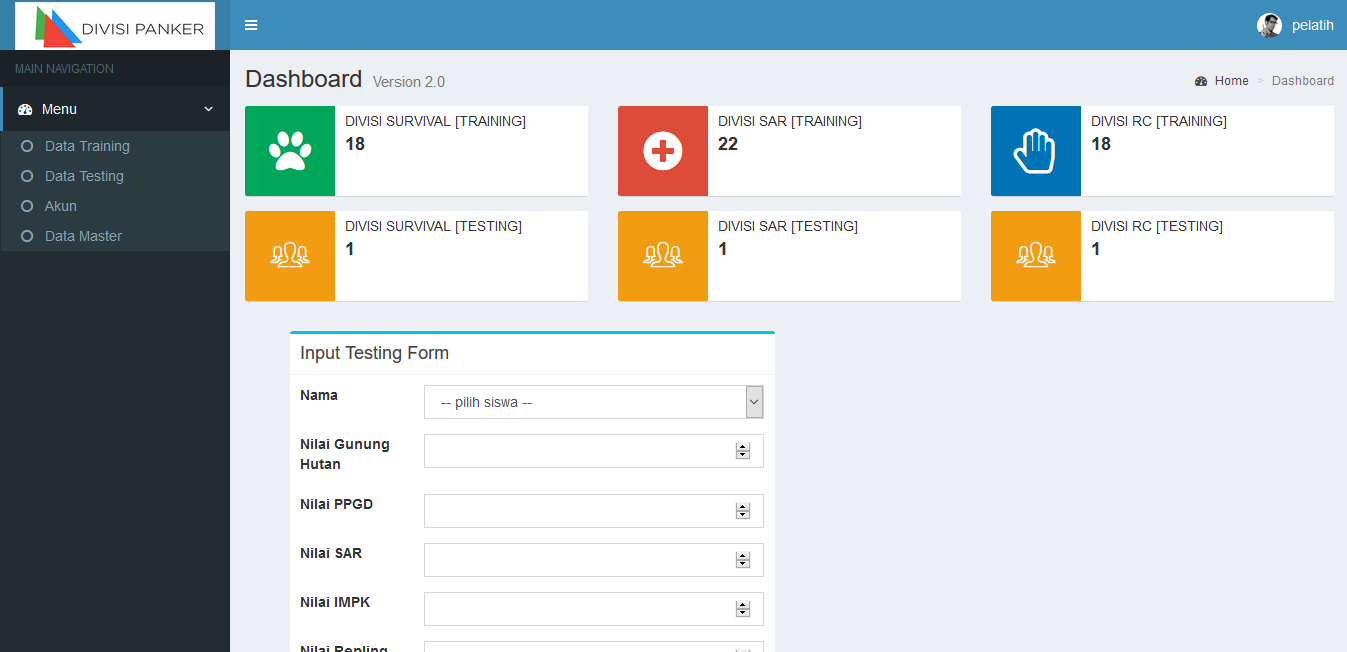
Halaman Tampilan data Training merupakan halaman yang berisikan data-data training yang akan digunkan untuk pengolahan naive bayes.



Gambar 4. 3 Tampilan Tabel Training

## Implementasi Hasil Tampilan Data Testing

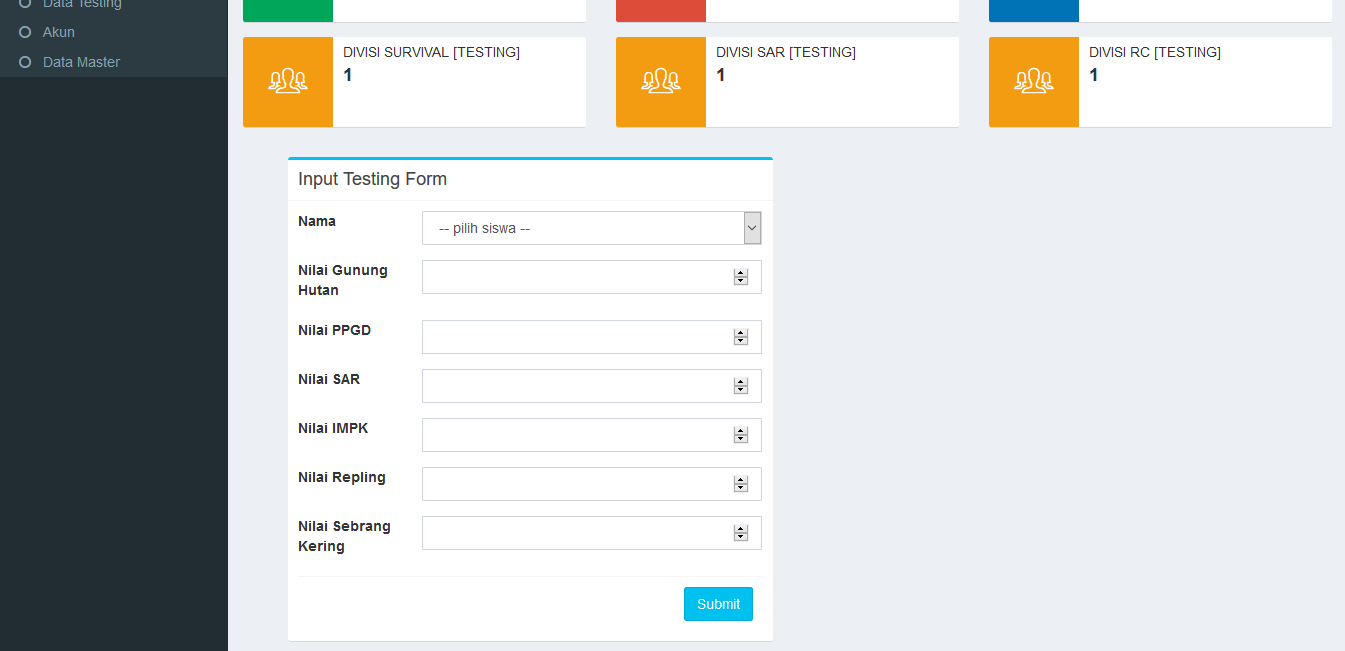
Halaman Testing ini merupakan halaman kumpulan data-data hasil proses klasifikasi pendivisian menentukan rekomendasi calon pengurus.



Gambar 4. 4 Tampilan Halaman Testing

## Implementasi Form Testing Klasifikasi

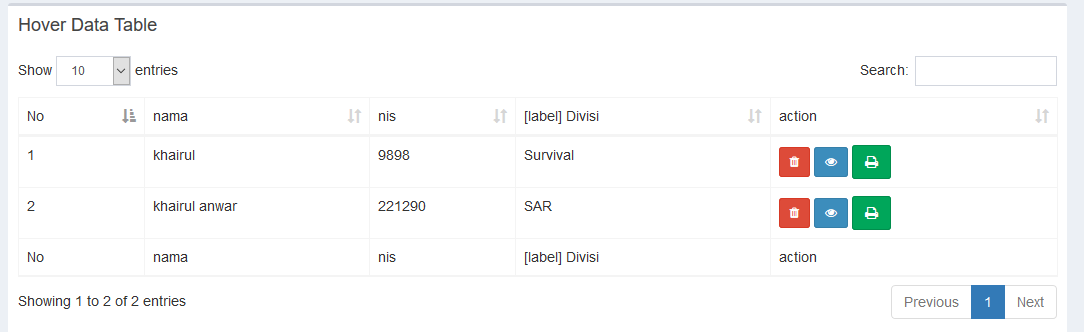
Halaman form testing klasifikasi ini merupakan halaman untuk memasukan data klasifikasi yang selanjutnya datanya akan disimpan pada tabel data testing.



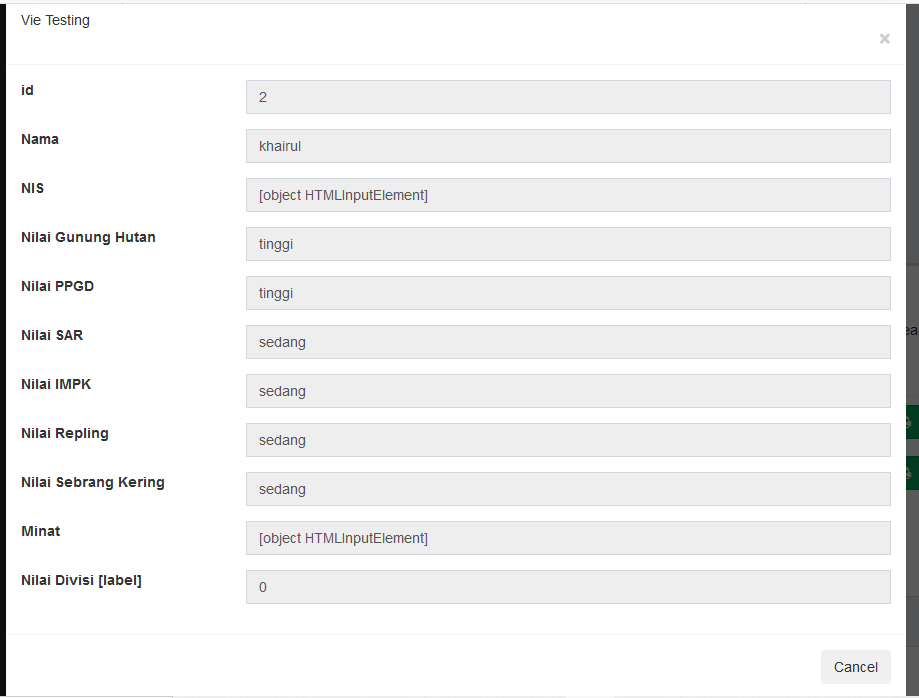
Gambar 4. 5 Tampilan Form Testing

## Implementasi Hasil Tampilan Modal View Testing

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



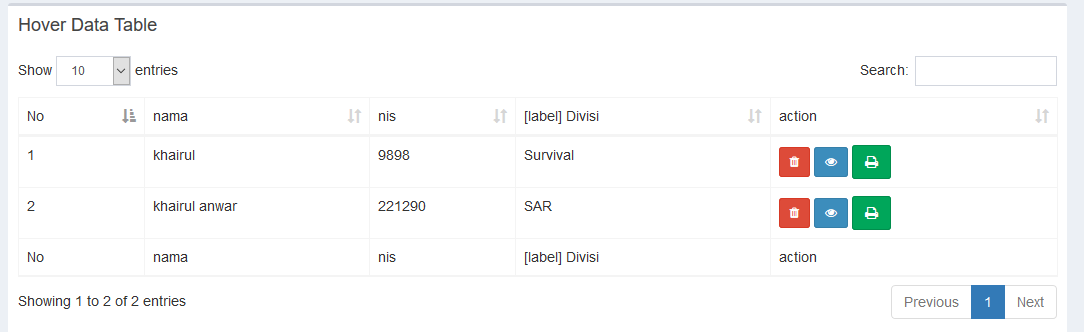
Gambar 4. 6Tampilan Tabel Testing



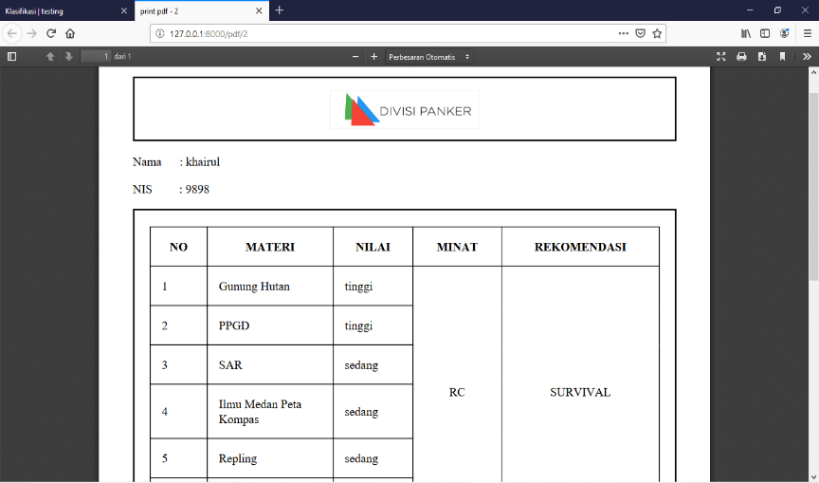
Gambar 4. 7 Tampilan Modal View Testing

## Implementasi Hasil Tampilan Export Pdf

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



Gambar 4. 8 Tampilan Tabel Testing



Gambar 4. 9 Tampilan Export Pdf

## 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.

## Tabel users

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

Tabel 4. 1 Tabel users

****

## Tabel training

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

Tabel 4. 2 Tabel training

****

## Tabel testing

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

Tabel 4. 3 Tabel testing



1. **Tabel siswa**

Tabel ini digunakan untuk meyimpan data siswa. Adapun pembuatan tabel ini bisa dilihat pada gambar dan keterangan berikut.

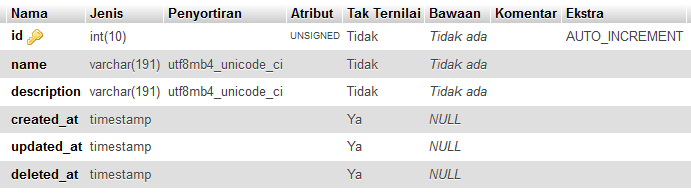
Tabel 4. 4 Tabel siswa

****

1. **Tabel roles**

Tabel roles merupakn tabel untuk meyimpan data role dan deskripsi dari masing masung role. Adapun pembuatan tabel ini bisa dilihat pada gambar dan keterangan berikut.

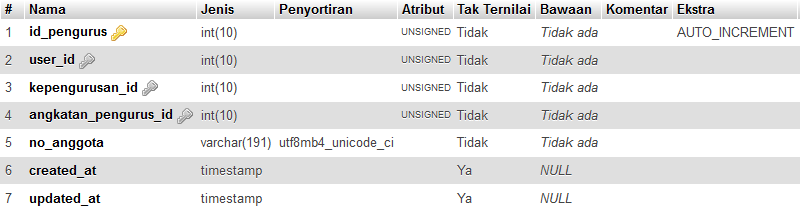
Tabel 4. 5 Tabel roles

****

1. **Tabel pengurus**

Tabel pengurus untuk menyimpan data-data pengursu. Adapun pembuatan tabel ini bisa dilihat pada gambar dan keterangan berikut.

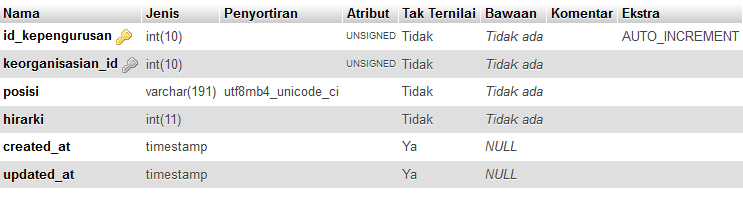
Tabel 4. 6 Tabel pengurus

****

1. **Tabel kepengurusan**

Tabel kepengurusan merupakan tabel untuk menyimpan data-data kepengurusan. Adapun pembuatan tabel ini bisa dilihat pada gambar dan keterangan berikut.

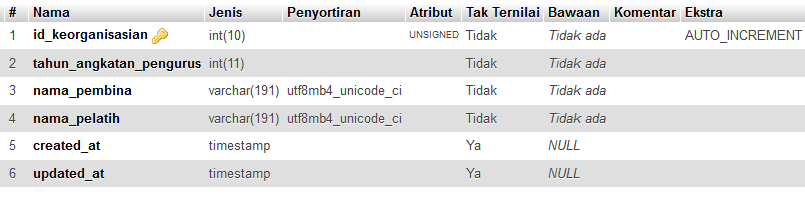
Tabel 4. 7 Tabel kepengurusan

****

1. **Tabel keorganisasian**

Tabel keorganisasian merupakan tabel untuk menyimpan data organisasi dari masing-masing angkan sesuai dengan tahun keorrganisasian. Adapun pembuatan tabel ini bisa dilihat pada gambar dan keterangan berikut.

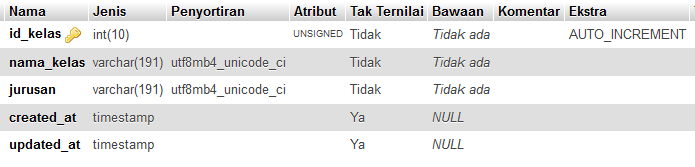
Tabel 4. 8 Tabel keorganisasi

****

1. **Tabel Kelas**

Tabel kelas merupakan tabel untuk menyimpan data kelas beserta jurusannya. Adapun pembuatan tabel ini bisa dilihat pada gambar dan keterangan berikut.

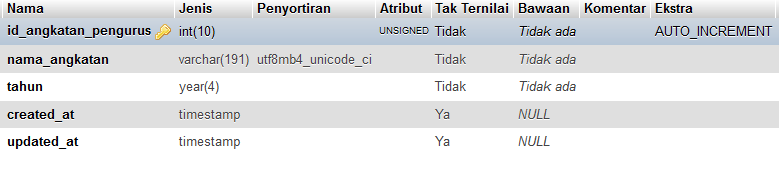
Tabel 4. 9 Tabel kelas

****

1. **Tabel angkatan\_pengurus**

Tabel angkatan\_pengurus merupakan tabel untuk menyimpan data angkatan kepengurun . Adapun pembuatan tabel ini bisa dilihat pada gambar dan keterangan berikut.

Tabel 4. 10 Tabel angkatan\_pengurus

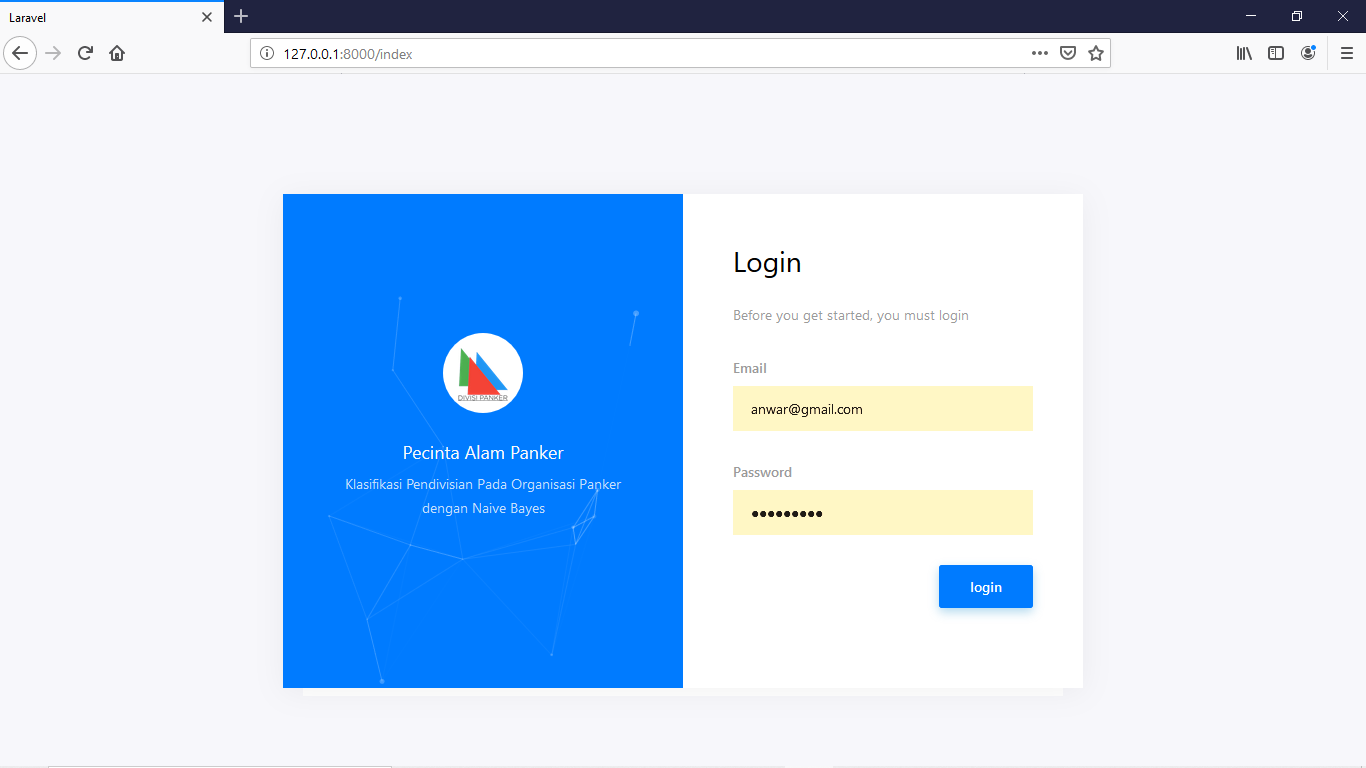
****

## Tahap Pengujian Sistem

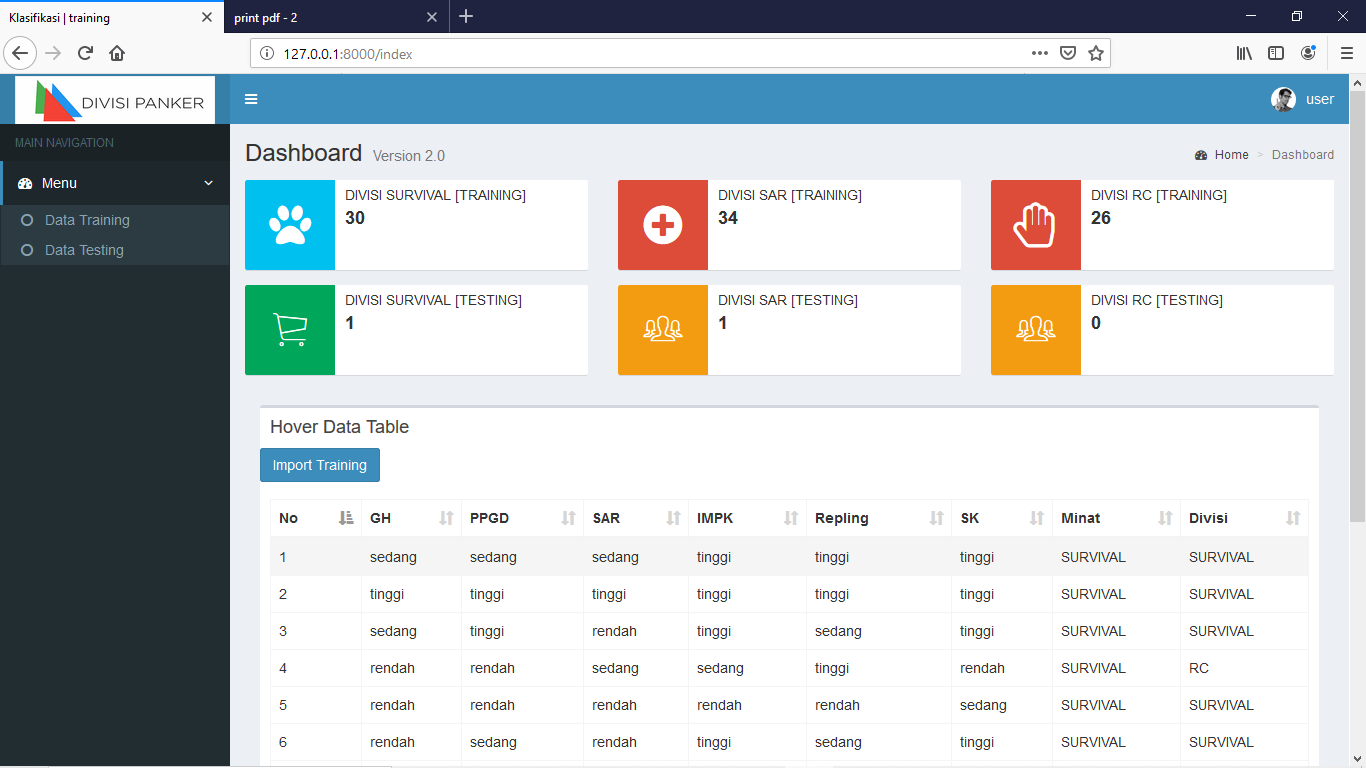
Tahap Pengujian sistem ini merupakan sebuah pengujian sistem aplikasi kemudian akan dijelaskan prosedur dan hasillnya sebagai berikut:

## User Melakukan Login

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

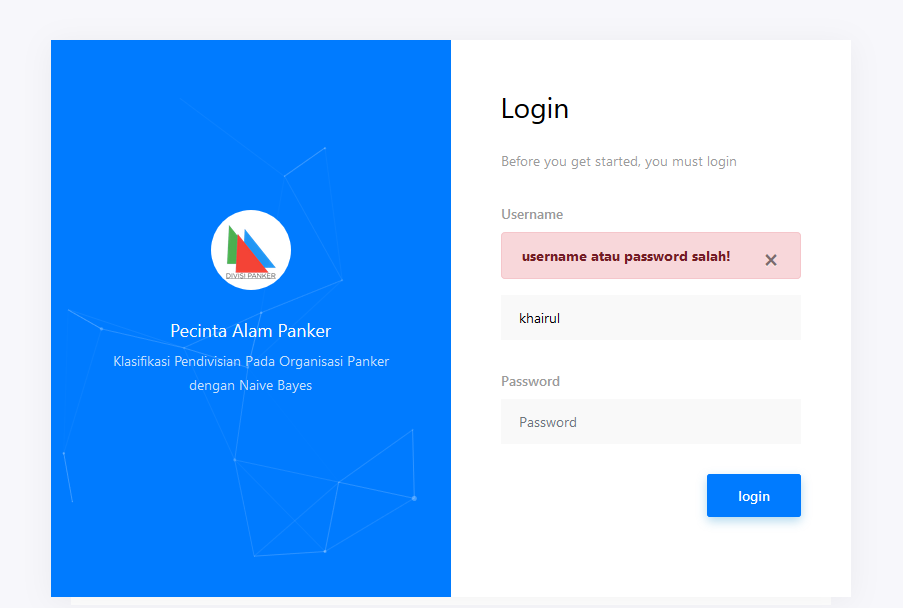
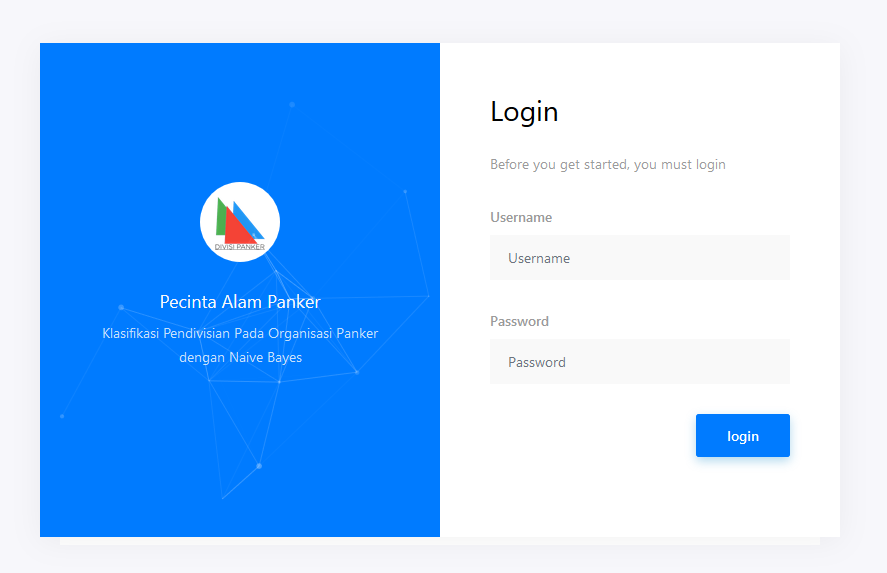


Gambar 4.13 Tampilan Halaman Login



Gambar 4.14 Gambar Setelah Berhasil Melakukan Login

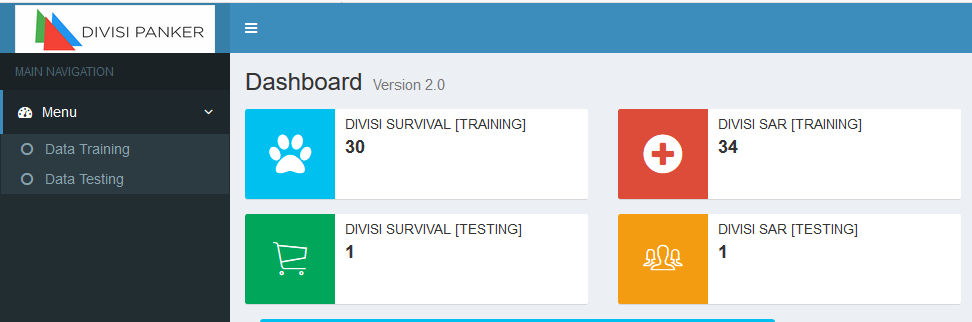
Jika user salah mengisi form *username* atau *password* maka akan user tidak bisa melakukan login akan di kembali ke halaman login untuk mengisi form login dengan benar dan akan muncul peringatan , bisa dilihat pada gambar berikut:



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

## User Memilih Menu Training

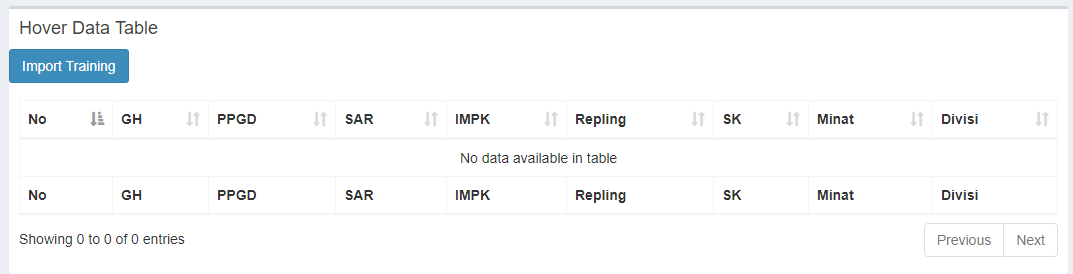
User memilih menu training untuk melihat data-data training hasil dari import data excel training.dikarenakan data banyak hanya ditampilkan beberapa tab.



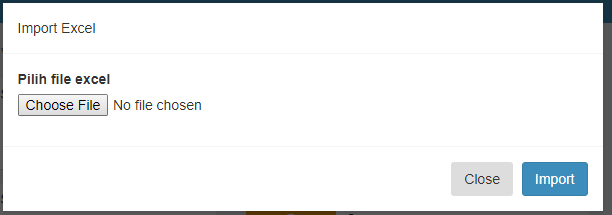
Gambar 4.16 Tampilan Menu dan Sub-menu

## User Menginport Data Training

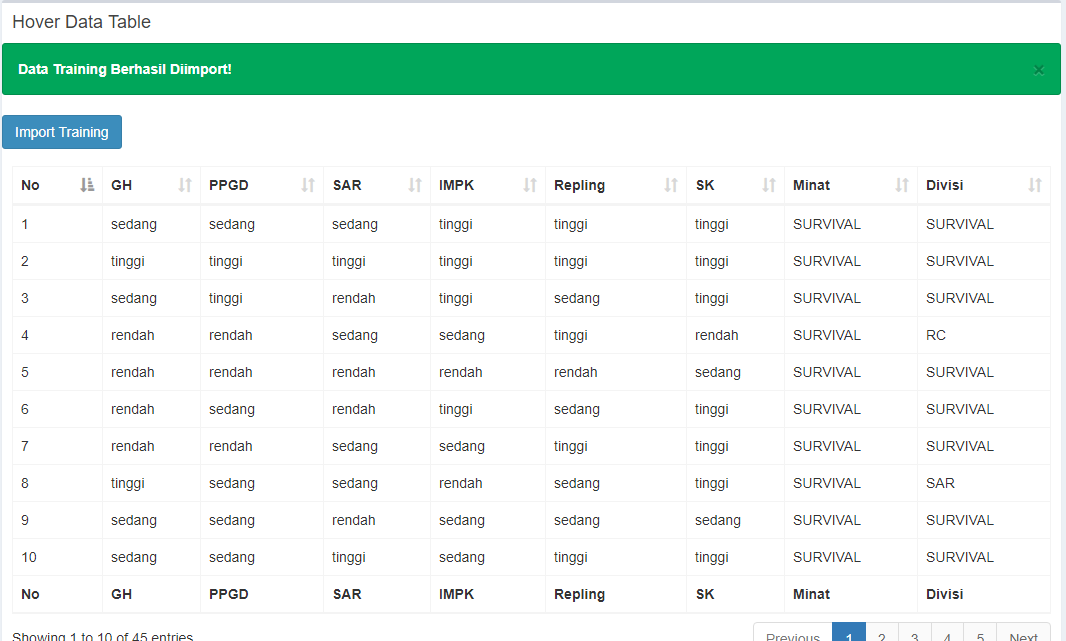
User menginputkan data training dengan mengimportkan data excel yang berisikan data data training untuk di olah sebagai pengolahan naive bayes untuk mencari probabilitas dalam penginputan data testing.



Gambar 4.17 Gambar Import Data Training



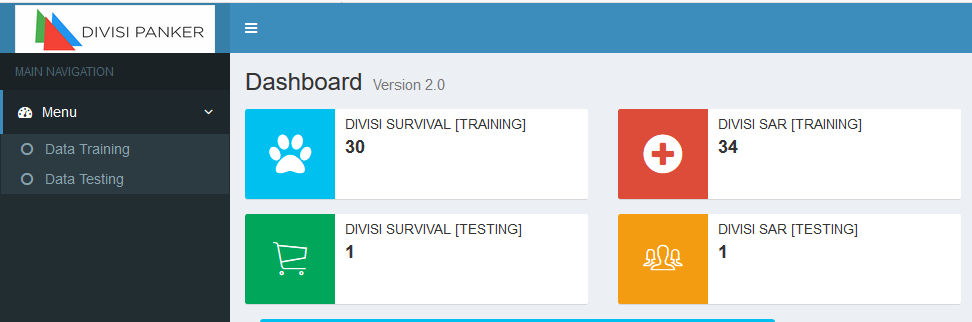
Gambar 4.18 Gambar Pilih Data Excel



Gambar 4.19 Gambar Status Berhasil Import

## User Memilih Menu Testing

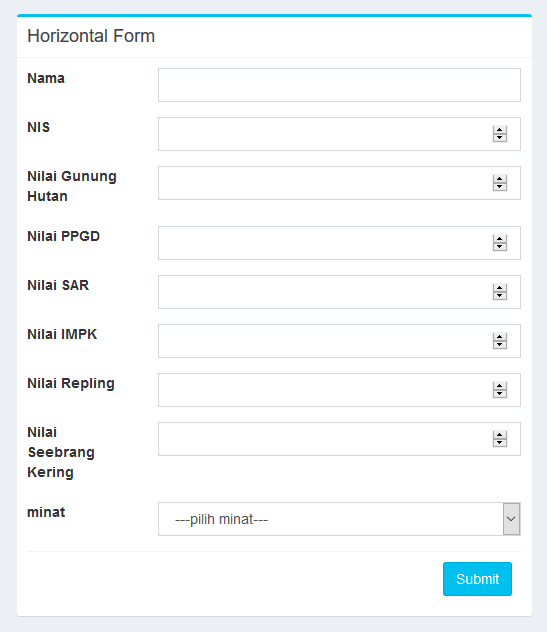
User memilih menu tesing untuk melakukan pengujian data testing, dalam halaman menu testing terdapat sebuah form untuk mengklasifikasikan pengujian data testing.



Gambar 4.20 Tampilan Menu dan sub-menu

## User Melakukan Klasifikasi Calon Pengurus

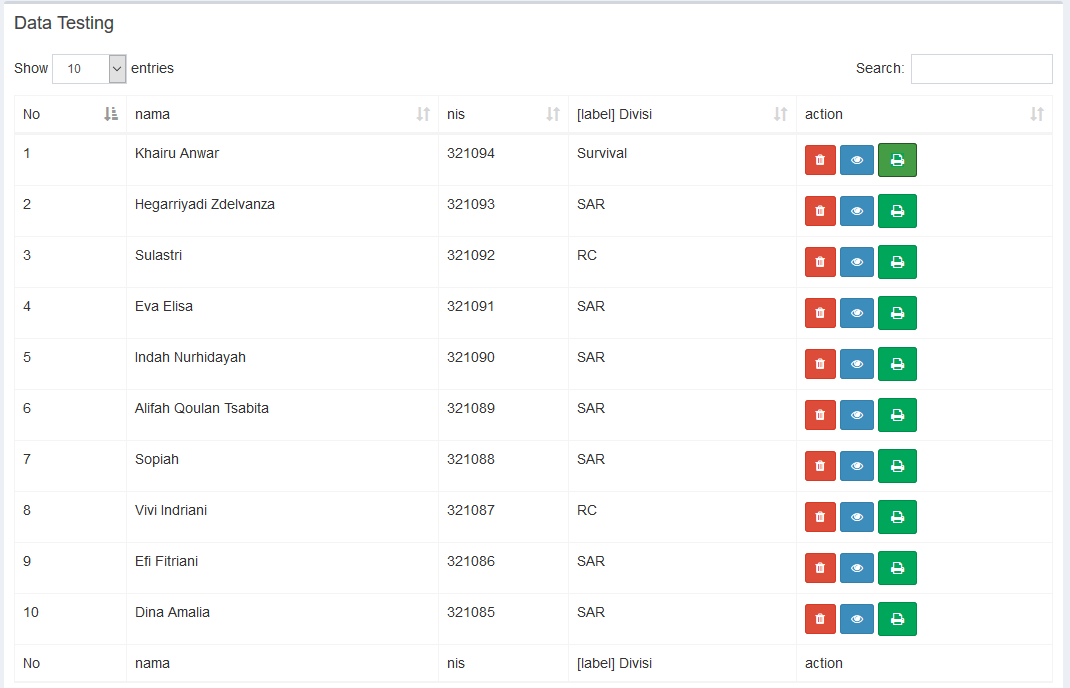
Setelah user memilih menu testing makan user mengklasifikasikan data testing calon pengurus untuk menentukan rekomendasikan calon pengurus ke divisi yang direkomendasikan.



Gambar 4.21 form Klasifikasi Calon Pengurus

## Pelatih Melihat Data Testing

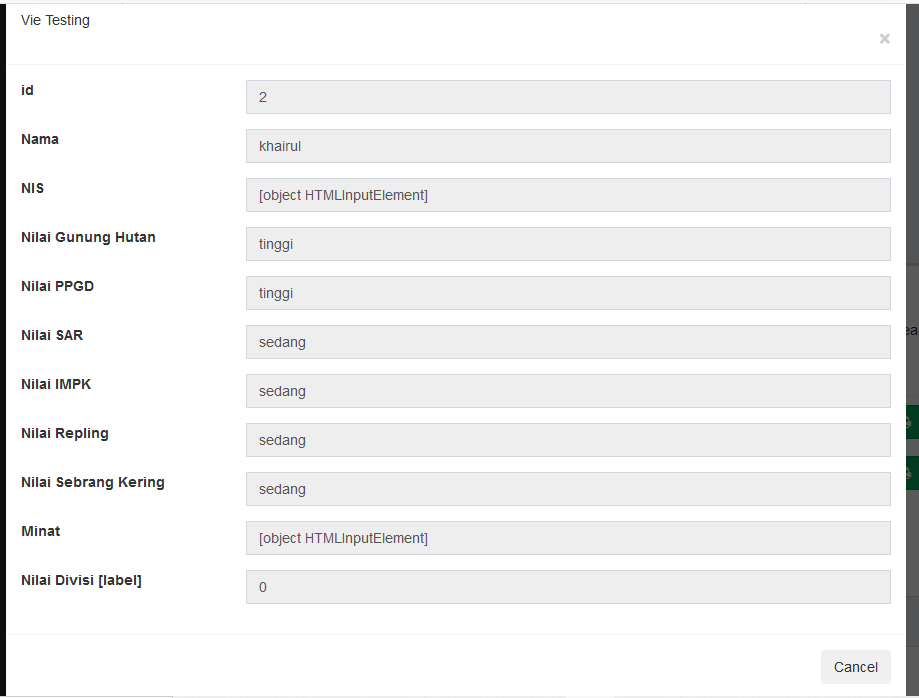
User melihat tabel yang berisikan data data hasil dari klasifikasi yang berbentuk data testing yang ditampilkan nama dan label nya .tabel dapat dilihat pada gambar berikut:



Gambar 4.22 Tabel Testing

## Pelatih Melihat Hasil Data Testing

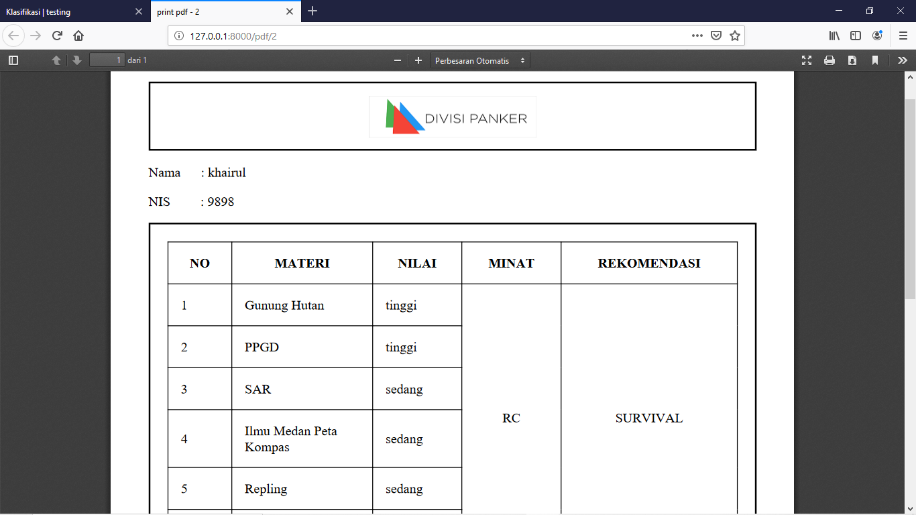
User melihat hasil testing klasifikasi dan melihat rincian secara detai nilai nilai sesuai dengan atribute dan label yang dimasukan melalui proses klasifikasi.



Gambar 4.23 View Modal Testing

## Pelatih Mengexport Pdf Hasil Testing

User mengeksport 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.



Gambar 4.24 Tampilan Export Testing Pdf

## Hasil Pengujian Kuisioner

Kuisioner tersebut terdiri dari 6 buah pertanyaan yang diberikan kepada 1 orang pelatih, 13 orang penguji sebagai pengurus, dan 15 orang penguji sebagai siswa dengan pertanyaan yang berbeda. Berikut adalah rekapitulasi pengujian kuisioner dari masing masing penguji.

**Keterangan:**

**A : Sangat Baik**  **C** **:** **Cukup**

**B : Baik D : Kurang**

Tabel 4. 11 Kuisioner untuk Pelatih

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **Pertanyaan** | **Penilaian** | | | |
| **A** | **B** | **C** | **D** |
| 1. | Bagaimana pendapat anda, apakah *website* ini cukup mudah digunakan? |  |  |  |  |
| 2. | Bagaimana pendapat anda tentang desain antarmuka dari *website* ini? |  |  |  |  |
| 3. | Apakah *website* ini dapat membantu anda dalam melakukan klasifikasi pendivisian ? |  |  |  |  |
| 4. | Apakah semua fungsi pada *website* ini untuk pelatih dapat berjalan dengan baik? |  |  |  |  |
| 5. | Apakah aplikasi ini memudahkan bagi anda sebagai pelatih? |  |  |  |  |
| 6. | Apakah sistem ini mudah untuk dipahami? |  |  |  |  |

Tabel 4. 12 Kuisioner untuk Pengurus

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **Pertanyaan** | **Penilaian** | | | |
| **A** | **B** | **C** | **D** |
| 1. | Bagaimana pendapat anda, apakah *website* ini cukup mudah digunakan? |  |  |  |  |
| 2. | Bagaimana pendapat anda tentang desain antarmuka dari *website* ini? |  |  |  |  |
| 3. | Bagaimana pendapat anda, apakah form input data siswa mudah digunakan? |  |  |  |  |
| 4. | Apakah semua fungsi pada *website* ini untuk pengurus dapat berjalan dengan baik? |  |  |  |  |
| 5. | Apakah *website* ini memudahkan bagi anda sebagai pengurus? |  |  |  |  |
| 6. | Apakah sistem ini mudah untuk dipahami? |  |  |  |  |

Tabel 4. 13 Kuisioner untuk Siswa

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **Pertanyaan** | **Penilaian** | | | |
| **A** | **B** | **C** | **D** |
| 1. | Bagaimana pendapat anda, apakah *website* ini cukup mudah digunakan? |  |  |  |  |
| 2. | Bagaimana pendapat anda tentang desain antarmuka dari *website* ini? |  |  |  |  |
| 3. | Apakah *website* ini dapat membantu anda untuk melihat hasil rekomendasi divisi ? |  |  |  |  |
| 4. | Apakah semua fungsi pada website ini untuk siswa dapat berjalan dengan baik? |  |  |  |  |
| 5. | Apakah *website* ini memudahkan bagi anda sebagai siswa? |  |  |  |  |
| 6. | Apakah sistem ini mudah untuk dipahami? |  |  |  |  |

## Hasil Jawaban Kuisioner

Tabel 4. 14 Hasil Jawaban Kuisioner Pelatih

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| No | Nama | Jawaban | | | | | |
| 1 | 2 | 3 | 4 | 5 | 6 |
| 1 | Carudi | B | A | C | B | B | A |

Tabel 4. 15 Hasil Jawaban Kuisioner Pengurus

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| No | Nama | Jawaban | | | | | |
| 1 | 2 | 3 | 4 | 5 | 6 |
| 1 | Nugraha Dwi Putra | A | A | A | A | A | A |
| 2 | Muhammad Syifa Rasiman | B | A | B | B | B | A |
| 3 | Erlangga | A | B | B | B | A | A |
| 4 | Kurnia Lailatul Fazriyah | A | B | A | B | A | A |
| 5 | Dilah Nadilah | B | B | A | B | A | B |
| 6 | Ainun Novi yansah | A | B | B | B | A | B |
| 7 | Nur Dwi Haryanti | B | C | B | B | A | A |
| 8 | Putri Widia Astuti | A | B | B | B | A | B |
| 9 | Andika Wiguna | B | C | B | B | A | B |
| 10 | Aminudin | A | A | A | B | A | B |
| 11 | Dedy Octavian | A | B | A | A | B | A |
| 12 | Nezla Rustianingsih | B | B | B | B | A | B |
| 13 | Amelia Syamsa Mayangsari | B | B | B | B | A | B |

Tabel 4. 16 Hasil Jawaban Kuisioner Siswa

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| No | Nama | Jawaban | | | | | |
| 1 | 2 | 3 | 4 | 5 | 6 |
| 1 | Andini | A | B | A | B | A | B |
| 2 | Triana Lasifa | B | B | A | A | A | A |
| 3 | Efi Safitri | B | B | A | A | B | A |
| 4 | Hegariyadi Zoelvanzah | B | B | B | B | C | B |
| 5 | Vivi Indriyani | B | A | A | B | B | B |
| 6 | Piya Apriyanti | B | A | A | A | B | A |
| 7 | Eva Elisa | B | A | A | B | A | B |
| 8 | Sopiyah | A | B | A | B | A | B |
| 9 | Sulastri | B | C | A | A | A | A |
| 10 | Alifah | B | C | B | B | B | B |
| 11 | Arifah Qoulan Tsabita | A | A | B | A | A | A |
| 12 | Indah Nur Hidayah | B | A | A | C | B | C |
| 13 | Aminudin | A | A | A | B | A | B |
| 14 | Jacob Nawawi | A | A | A | A | A | A |
| 15 | Dina Amaliyah | B | A | A | C | C | A |

Tabel 4. 17 Hasil Jawaban Kuisioner Keseluruhan

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| No | Nama | jawaban | | | | | |
| 1 | 2 | 3 | 4 | 5 | 6 |
| 1 | Carudi | B | A | C | B | B | A |
| 2 | Nugraha Dwi Putra | A | A | A | A | A | A |
| 3 | Muhammad Syifa Rasiman | B | A | B | B | B | A |
| 4 | Erlangga | A | B | B | B | A | A |
| 5 | Kurnia Lailatul Fazriyah | A | B | A | B | A | A |
| 6 | Dilah Nadilah | B | B | A | B | A | B |
| 7 | Ainun Novi yansah | A | B | B | B | A | B |
| 8 | Nur Dwi Haryanti | B | C | B | B | A | A |
| 9 | Putri Widia Astuti | A | B | B | B | A | B |
| 10 | Andika Wiguna | B | C | B | B | A | B |
| 11 | Aminudin | A | A | A | B | A | B |
| 12 | Dedy Octavian | A | B | A | A | B | A |
| 13 | Nezla Rustianingsih | B | B | B | B | A | B |
| 14 | Amelia Syamsa Mayangsari | B | B | B | B | A | B |
| 15 | Andini | A | B | A | B | A | B |
| 16 | Triana Lasifa | B | B | A | A | A | A |
| 17 | Efi Safitri | B | B | A | A | B | A |
| 18 | Hegariyadi Zoelvanzah | B | B | B | B | C | B |
| 19 | Vivi Indriyani | B | A | A | B | B | B |
| 20 | Piya Apriyanti | B | A | A | A | B | A |
| 21 | Eva Elisa | B | A | A | B | A | B |
| 22 | Sopiyah | A | B | A | B | A | B |
| 23 | Sulastri | B | C | A | A | A | A |
| 24 | Alifah | B | C | B | B | B | B |
| 25 | Arifah Qoulan Tsabita | A | A | B | A | A | A |
| 26 | Indah Nur Hidayah | B | A | A | C | B | C |
| 27 | Aminudin | A | A | A | B | A | B |
| 28 | Jacob Nawawi | A | A | A | A | A | A |
| 29 | Dina Amaliyah | B | A | A | C | C | A |

## Rekap Pertanyaan Kuisioner Pelatih

Tabel 4. 18 Rekap Hasil Jawaban Kuisioner Pelatih

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Sangat Baik** | **Baik** | **Cukup** | **Kurang** |
| 1 | 0 | 1 | 0 | 0 |
| 2 | 1 | 0 | 0 | 0 |
| 3 | 0 | 0 | 1 | 0 |
| 4 | 0 | 1 | 0 | 0 |
| 5 | 0 | 1 | 0 | 0 |
| 6 | 1 | 0 | 0 | 0 |

Dari hasil jawaban pengujian dengan kuisioner tersebut akan dihitung dengan menggunakan skala likert sebagai berikut:

A (Sangat baik) = 4

B (Baik) = 3

C (Cukup) = 2

D (Kurang) = 1

Skor tertinggi = 4 x 1= 4

Skor terendah = 1 x 1 = 1

Skor kriteria = Jumlah responden x Skor nilai (4.1)

Rating Scale

76% - 100% = Sangat baik

50% - 75% = Baik

26% - 50% = Cukup

0% - 25% = Kurang

Presentase persetujuan

*p* = x 100% (4.2)

1. Pertanyaan Pertama

Sangat baik = *p* = x 100% = 0%

Baik = *p* = x 100% = 75%

Cukup = *p* = x 100% = 0%

Kurang = *p* = x 100% = 0% +

Jumlah = 75%

Dari hasil pengujian dengan kuisioner untuk pertanyaan pertama diperoleh presentase 75% dan pertanyaan tersebut masuk ke dalam kategori sangat baik

1. Pertanyaan Kedua

Sangat baik = *p* = x 100% = 100 %

Baik = *p* = x 100% = 0%

Cukup = *p* = x 100% = 0%

Kurang = *p* = x 100% = 0% +

Jumlah = 100%

Dari hasil pengujian dengan kuisioner untuk pertanyaan pertama diperoleh presentase 100% dan pertanyaan tersebut masuk ke dalam kategori sangat baik

1. Pertanyaan Ketiga

Sangat baik = *p* = x 100% = 0%

Baik = *p* = x 100% = 0%

Cukup = *p* = x 100% = 50%

Kurang = *p* = x 100% = 0% +

Jumlah = 50%

Dari hasil pengujian dengan kuisioner untuk pertanyaan pertama diperoleh presentase 50% dan pertanyaan tersebut masuk ke dalam kategori sangat baik.

1. Pertanyaan Keempat

Sangat baik = *p* = x 100% = 0%

Baik = *p* = x 100% = 75%

Cukup = *p* = x 100% = 0%

Kurang = *p* = x 100% = 0 % +

Jumlah = 75%

Dari hasil pengujian dengan kuisioner untuk pertanyaan pertama diperoleh presentase 75% dan pertanyaan tersebut masuk ke dalam kategori sangat baik.

1. Pertanyaan Kelima

Sangat baik = *p* = x 100% = 0%

Baik = *p* = x 100% = 75%

Cukup = *p* = x 100% = 0%

Kurang = *p* = x 100% = 0% +

Jumlah = 75%

Dari hasil pengujian dengan kuisioner untuk pertanyaan pertama diperoleh presentase 96,14% dan pertanyaan tersebut masuk ke dalam kategori sangat baik.

1. Pertanyaan Keenam

Sangat baik = *p* = x 100% = 100%

Baik = *p* = x 100% = 0%

Cukup = *p* = x 100% = 0%

Kurang = *p* = x 100% = 0% +

Jumlah = 100%

Dari hasil pengujian dengan kuisioner untuk pertanyaan pertama diperoleh presentase 100% dan pertanyaan tersebut masuk ke dalam kategori sangat baik.

Dari hasil pengujian dengan kuesioner untuk semua pertanyaan dapat diperoleh rata – rata presentase seperti berikut :

75% + 100% + 50%+ 75% + 75% + 100% = 79.16%

6

## Rekap Kuisioner Pengurus

Tabel 4. 19 Rekap Hasil Kuisioner Pengurus

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Sangat Baik** | **Baik** | **Cukup** | **Kurang** |
| 1 | 7 | 6 | 0 | 0 |
| 2 | 3 | 8 | 2 | 0 |
| 3 | 5 | 8 | 0 | 0 |
| 4 | 2 | 11 | 0 | 0 |
| 5 | 11 | 2 | 0 | 0 |
| 6 | 6 | 7 | 0 | 0 |

Dari hasil jawaban pengujian dengan kuisioner tersebut akan dihitung dengan menggunakan skala likert sebagai berikut:

A (Sangat baik) = 4

B (Baik) = 3

C (Cukup) = 2

D (Kurang) = 1

Skor tertinggi = 4 x 13 = 52

Skor terendah = 1 x 13 = 13

Skor kriteria = Jumlah responden x Skor nilai (4.3)

Rating Scale

76% - 100% = Sangat baik

50% - 75% = Baik

26% - 50% = Cukup

0% - 25% = Kurang

Presentase persetujuan

*p* = x 100% (4.4)

1. Pertanyaan Pertama

Sangat baik = *p* = x 100% = 53,84%

Baik = *p* = x 100% = 34.16,15%

Cukup = *p* = x 100% = 7,69%

Kurang = *p* = x 100% = 0% +

Jumlah = 95,63%

Dari hasil pengujian dengan kuisioner untuk pertanyaan pertama diperoleh presentase 95,63% dan pertanyaan tersebut masuk ke dalam kategori sangat baik

1. Pertanyaan Kedua

Sangat baik = *p* = x 100% = 23,07 %

Baik = *p* = x 100% = 46,15%

Cukup = *p* = x 100% = 7,69%

Kurang = *p* = x 100% = 0% +

Jumlah = 76,91%

Dari hasil pengujian dengan kuisioner untuk pertanyaan pertama diperoleh presentase 76,91%dan pertanyaan tersebut masuk ke dalam kategori sangat baik

1. Pertanyaan Ketiga

Sangat baik = *p* = x 100% = 38,62%

Baik = *p* = x 100% = 46,15%

Cukup = *p* = x 100% = 0%

Kurang = *p* = x 100% = 0% +

Jumlah = 84.77%

Dari hasil pengujian dengan kuisioner untuk pertanyaan pertama diperoleh presentase 84.77% dan pertanyaan tersebut masuk ke dalam kategori sangat baik.

1. Pertanyaan Keempat

Sangat baik = *p* = x 100% = 15,38%

Baik = *p* = x 100% = 63,46%

Cukup = *p* = x 100% = 0%

Kurang = *p* = x 100% = 0 % +

Jumlah = 78,84%

Dari hasil pengujian dengan kuisioner untuk pertanyaan pertama diperoleh presentase 78,84% dan pertanyaan tersebut masuk ke dalam kategori sangat baik.

1. Pertanyaan Kelima

Sangat baik = *p* = x 100% = 84,61%

Baik = *p* = x 100% = 11,53%

Cukup = *p* = x 100% = 0%

Kurang = *p* = x 100% = 0% +

Jumlah = 96,14%

Dari hasil pengujian dengan kuisioner untuk pertanyaan pertama diperoleh presentase 96,14% dan pertanyaan tersebut masuk ke dalam kategori sangat baik.

1. Pertanyaan Keenam

Sangat baik = *p* = x 100% =46,15%

Baik = *p* = x 100% = 40.38%

Cukup = *p* = x 100% = 0%

Kurang = *p* = x 100% = 0% +

Jumlah = 86.53%

Dari hasil pengujian dengan kuisioner untuk pertanyaan pertama diperoleh presentase 86.53% dan pertanyaan tersebut masuk ke dalam kategori sangat baik.

Dari hasil pengujian dengan kuesioner untuk semua pertanyaan dapat diperoleh rata – rata presentase seperti berikut :

95,63% + 76,91% + 84,77%+78,84% + 96,14% + 86,53% = 86.47%

6

## Rekap Pertanyaan Kuisioner Siswa

Tabel 4. 20 Rekap Hasil Kuisioner Siswa

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Sangat Baik** | **Baik** | **Cukup** | **Kurang** |
| 1 | 5 | 10 | 0 | 0 |
| 2 | 8 | 5 | 2 | 0 |
| 3 | 12 | 3 | 0 | 0 |
| 4 | 6 | 7 | 2 | 0 |
| 5 | 8 | 5 | 2 | 0 |
| 6 | 7 | 7 | 1 | 0 |

Dari hasil jawaban pengujian dengan kuisioner tersebut akan dihitung dengan menggunakan skala likert sebagai berikut:

A (Sangat baik) = 4

B (Baik) = 3

C (Cukup) = 2

D (Kurang) = 1

Skor tertinggi = 4 x 15 = 60

Skor terendah = 1 x 15 = 15

Skor kriteria = Jumlah responden x Skor nilai (4.5)

Rating Scale

76% - 100% = Sangat baik

50% - 75% = Baik

26% - 50% = Cukup

0% - 25% = Kurang

Presentase persetujuan

*p* = x 100% (4.6)

1. Pertanyaan Pertama

Sangat baik = *p* = x 100% = 33,33%

Baik = *p* = x 100% = 50%

Cukup = *p* = x 100% = 0%

Kurang = *p* = x 100% = 0% +

Jumlah = 83,33%

Dari hasil pengujian dengan kuisioner untuk pertanyaan pertama diperoleh presentase 83,33% dan pertanyaan tersebut masuk ke dalam kategori sangat baik

1. Pertanyaan Kedua

Sangat baik = *p* = x 100% = 53,33%

Baik = *p* = x 100% = 25%

Cukup = *p* = x 100% = 6,66%

Kurang = *p* = x 100% = 0% +

Jumlah = 84,99%

Dari hasil pengujian dengan kuisioner untuk pertanyaan pertama diperoleh presentase 84,99% dan pertanyaan tersebut masuk ke dalam kategori sangat baik

1. Pertanyaan Ketiga

Sangat baik = *p* = x 100% = 80%

Baik = *p* = x 100% = 15%

Cukup = *p* = x 100% = 0%

Kurang = *p* = x 100% = 0% +

Jumlah = 95%

Dari hasil pengujian dengan kuisioner untuk pertanyaan pertama diperoleh presentase 95% dan pertanyaan tersebut masuk ke dalam kategori sangat baik.

1. Pertanyaan Keempat

Sangat baik = *p* = x 100% = 40%

Baik = *p* = x 100% = 35%

Cukup = *p* = x 100% = 6,66%

Kurang = *p* = x 100% = 0 % +

Jumlah = 81,66%

Dari hasil pengujian dengan kuisioner untuk pertanyaan pertama diperoleh presentase 81,66% dan pertanyaan tersebut masuk ke dalam kategori sangat baik.

1. Pertanyaan Kelima

Sangat baik = *p* = x 100% = 53,33%

Baik = *p* = x 100% = 25%

Cukup = *p* = x 100% = 6,66%

Kurang = *p* = x 100% = 0% +

Jumlah = 84,99%

Dari hasil pengujian dengan kuisioner untuk pertanyaan pertama diperoleh presentase 84,99% dan pertanyaan tersebut masuk ke dalam kategori sangat baik.

1. Pertanyaan Keenam

Sangat baik = *p* = x 100% =46,66%

Baik = *p* = x 100% = 35%

Cukup = *p* = x 100% = 3,3%

Kurang = *p* = x 100% = 0% +

Jumlah = 84,96%

Dari hasil pengujian dengan kuisioner untuk pertanyaan pertama diperoleh presentase 84,96% dan pertanyaan tersebut masuk ke dalam kategori sangat baik.

Dari hasil pengujian dengan kuesioner untuk semua pertanyaan dapat diperoleh rata – rata presentase seperti berikut :

83,33% + 84,99% + 95% + 81,66% + 84,99% + 84,96% = 85.82%

6

## Rekapitulasi Hasil Perhitungan Semua Kuisioner

Dari hasil pengujian dengan kuesioner untuk semua pertanyaan pengujian dari pelatih, pengurus, dan siswa dapat diperoleh rata – rata presentase seperti berikut :

79.16% + 86.47% + 85,82% = 83.81%

3

Dari hasil perhitungan jawaban kuesioner dapat disimpulkan bahwa rata-rata presentase dengan jumlah rata – rata presentase adalah 83.81%. Sistem ini secara keseluruhan masuk kedalam kategori sangat baik untuk gunakan

## Hasil Pengujian *Black-Box*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **Descripsi Pengujian** | **Prosedur Pengujian** | **Hasil Yang diharpakan** | **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 |

## Penjelasan Dataset

1. **Hasil Uji Coba Testing Model**

Uji coba data dilakukan data dilakukan dengan 23 data sampel data yang dimasukana untuk melihat data dan ke akurasian.data bisa dilihat pada tabel dibawah ini.

Tabel 4. 21 Hasil Uji Coba Data Testing Model

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| no | nilai\_gh | nilai\_ppgd | nilai\_sar | nilai\_impk | nilai\_repling | nilai\_sebrang\_kering | minat | hasil | hasil (y/n) |
| 1 | 0 | 1 | 2 | 2 | 1 | 1 | 2 | 1 | n |
| 2 | 2 | 1 | 2 | 2 | 0 | 0 | 0 | 1 | y |
| 3 | 0 | 0 | 2 | 2 | 1 | 1 | 1 | 1 | y |
| 4 | 2 | 1 | 2 | 1 | 2 | 0 | 0 | 0 | y |
| 5 | 0 | 1 | 2 | 1 | 1 | 1 | 2 | 2 | n |
| 6 | 1 | 1 | 2 | 2 | 2 | 2 | 2 | 1 | n |
| 7 | 1 | 2 | 2 | 2 | 0 | 0 | 2 | 1 | y |
| 8 | 2 | 1 | 2 | 2 | 1 | 1 | 1 | 1 | y |
| 9 | 2 | 2 | 2 | 1 | 2 | 2 | 2 | 2 | n |
| 10 | 2 | 2 | 2 | 1 | 1 | 1 | 1 | 0 | y |
| 11 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 0 | y |
| 12 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | y |
| 13 | 2 | 1 | 2 | 2 | 1 | 2 | 0 | 0 | y |
| 14 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 2 | y |
| 15 | 1 | 2 | 2 | 2 | 1 | 1 | 0 | 0 | n |
| 16 | 1 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | y |
| 17 | 2 | 2 | 2 | 2 | 2 | 2 | 1 | 0 | y |
| 18 | 2 | 2 | 2 | 0 | 0 | 0 | 0 | 0 | y |
| 19 | 2 | 2 | 1 | 1 | 2 | 2 | 0 | 0 | y |
| 20 | 0 | 0 | 0 | 0 | 1 | 1 | 2 | 2 | y |
| 21 | 2 | 0 | 2 | 0 | 0 | 0 | 1 | 1 | n |
| 22 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | n |
| 23 | 1 | 2 | 2 | 1 | 0 | 1 | 1 | 1 | y |

Hasil dari tabel diatas menunjukan ke akurasian sekitar 70% dengan data tebakan tepat sebanyak 16 data dan data tebakan salah sebanyak 7 dengan total seluruh datannya yaitu 23 data uji.untuk presentase data bisa dilihat pada chart dibawah ini.

Gambar 4. 10 Chart Presentase Data Model

1. **Hasil Uji Coba Testing Sebenarnya**

Uji coba data testing ini dilakukan dengan menginput kan 13 data yang diambil dari data pengurus untuk melihat hasil tingkat ke akurasian data model yang digunakan. Data bisa dilihat pada tabel di bawah ini.

Tabel 4. 22 Uji Coba Data Testing Real

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| no | nilai\_gh | nilai\_ppgd | nilai\_sar | nilai\_impk | nilai\_repling | nilai\_sebrang\_kering | minat | tebakan | hasil |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | y |
| 2 | 1 | 1 | 1 | 2 | 1 | 1 | 0 | 1 | y |
| 3 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | y |
| 4 | 1 | 1 | 1 | 1 | 1 | 1 | 2 | 2 | y |
| 5 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | y |
| 6 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | y |
| 7 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | n |
| 8 | 1 | 1 | 1 | 2 | 1 | 1 | 1 | 1 | y |
| 9 | 1 | 1 | 1 | 1 | 1 | 1 | 2 | 1 | n |
| 10 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | n |
| 11 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 2 | y |
| 12 | 1 | 1 | 1 | 1 | 1 | 1 | 2 | 2 | y |
| 13 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 2 | y |

Hasil dari tabel di atas menunjukan ke akurasian di dapatkan dari 13 data yaitu sekitar 77% dengan data tebakan dan hasil yang sama yaitu 10 data tebakan tepat dan 3 data tebakan meleset.

Gambar 4. 11 Chart Presentase Data Real

## Kelebihan dan Kekurangan

1. **Kelebihan**

Adapun kelebihan dari aplikasi ini adalah:

1. Aplikasi ini membantu pengurus organisasi untuk merekomendasikan divisi sesuai dengan nilai nilai atribute yang dimasukan di aplikasi
2. Aplikasi ini mempermudah dokementasi pendivisian calon pengurus
3. Aplikasi ini mempermudah penyampaian informasi ke calon pengurus
4. **Kekurangan**

Adapun kekurangan dari aplikasi ini adalah:

1. Aplikasi ini menggunkan data training dengan jumlah yang masih sedikit
2. Apliaksi ini masih belum memiliki fitur lengkap
3. Penyampaian data pada aplikasi ini masih sederhana

# BAB V PENUTUP

## 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, aplikasi ini mempunyai nilai ke akurasian sebesar 77% dan nilai hasil kuisioner 83.81% .
2. Aplikasi Klasifikasi Pendivisian Organisasi Panker Menggunakan Naive Bayes dapat memudahkan Pelatih,Pengurus skeretariat khususnya bagian Kedivisian dan siswa untuk memperoses pengelompokan divisi. Dalam sekala presentaste aplikasi ini memudahkan adalah sekitar 65% sangat mudahkan, 28% memudahkan, 0% kurang memudahkan
3. Aplikasi Klasifikasi Pendivisian Organisasi Panker Menggunakan Naive Bayes ini menggunkan metode penelitian *waterfall* dengan diawali analis kebutuhan organisasi tersebut dengan aplikasi yang akan dibuat selanjutnya menuju ke perancnagan sistem, tahap ini akan di rancang dari tahap yang sebelumnya dan dilanjutkan dengan penulisan kode program untuk membuat desain sistem web yang akan dibuat selanjutnya menuju ke pengujian sistem diproses lagi ketahap akhir yaitu pemeliharaan. Wawancara pengumpulan data bertujuan untuk mengambil data testing dan data training.

## Saran

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

1. Aplikasi dapat dikembangkan agar bisa digunkan mencangkup beberapa fitur yang bisa ditambahkan atau di sempurnakan.
2. Rancangan tampilan dapat dikembangkan lebih menarik untuk memudahkan dalam penggunaan.
3. Data training harus seimbang bila untuk mendapatkan ke akurasian yang tepat

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

# DAFTAR PUSTAKA

Arif. 2018. Pengertian composer. http://www.kursuswebsite.org/pengertian-Com

poser/. Diakses pada tanggal 21 Mei 2019.

Bustami., (2013), Penerapan Algoritma Naive Bayes Untuk Mengklasifikasi Data

Nasabah Asuransi, TECHSI : Jurnal Penelitian Teknik Informatika, Vol. 3, No.2, Hal. 127-146.

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.kursus*website*.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.Pengertian

ku.net. Diakses pada tanggal 09 Maret 2019.

Osd, Dimas. 2015.Pengertian SI (Sistem Informasi)*.* https://www.kompasiana.com.

Diakses pada tanggal 13 Maret 2019.

Patil, T. R., Sherekar, M. S., (2013), Performance Analysis of Naive Bayes and J48

Classification Algorithm for Data Classification, International Journal of Computer Science and Applications, Vol. 6, No. 2, Hal 256-261.

Rahardi,Wisa.2014.Mengenal Html https://www.blogooblok.com/2014/04/belajar-

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

Ridwan, M., Suyono, H., Sarosa, M., (2013), Penerapan Data Mining untuk

Evaluasi Kinerja Akademik Mahasiswa Menggunakan Algoritma Naive Bayes Classifier, Jurnal EECCIS, Vol 1, No. 7, Hal. 59-64.

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.

Xhemali, Daniela, Hinde, C.J., and Stone, R.G. (2009) Naive Bayes vs. Decision

Trees vs. Neural Networks in the Classification of Training Web Pages

# 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) | | |
|  | | | | | | | | | | | |
| **N0.** | **Nilai Gunung  Hutan \*\*** | **Nilai Penolongan Pertama Gawat Darurat\*\*** | | **Nilai SAR\*\*** | **Nilai IMPK\*\*** | | **Nilai Repling\*\*** | **Nilai Sebarang Kering\*\*** | | **Minat\*** | **Divisi\*** |
| 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 |
| 46 | 0 | 0 | | 0 | 0 | | 0 | 0 | | 0 | 0 |
| 47 | 0 | 0 | | 0 | 0 | | 0 | 0 | | 0 | 0 |
| 48 | 0 | 0 | | 0 | 0 | | 0 | 0 | | 0 | 1 |
| 49 | 0 | 0 | | 0 | 0 | | 0 | 0 | | 0 | 1 |
| 50 | 0 | 0 | | 0 | 0 | | 0 | 0 | | 0 | 2 |
| 51 | 0 | 0 | | 0 | 0 | | 0 | 0 | | 0 | 2 |
| 52 | 1 | 1 | | 1 | 1 | | 1 | 1 | | 1 | 2 |
| 53 | 2 | 1 | | 2 | 1 | | 2 | 1 | | 2 | 1 |
| 54 | 0 | 0 | | 0 | 0 | | 2 | 2 | | 2 | 2 |
| 55 | 0 | 0 | | 0 | 1 | | 1 | 2 | | 2 | 1 |
| 56 | 1 | 1 | | 1 | 1 | | 1 | 1 | | 1 | 0 |
| 57 | 1 | 1 | | 1 | 1 | | 1 | 1 | | 1 | 1 |
| 58 | 2 | 2 | | 1 | 1 | | 1 | 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);  $dttesting->save();  return redirect()->route('testing')->with('alert-success', 'data berhasil dimasukan');  }  //mengkonvert dari nilai 0,1,2 yang mewakili rendah sendang dan tinggi    private function kat\_atribut($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();    //mengabil jumlah data baris berdasarkan atribute dan label    //cnth : total atribute 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();  $p\_repling\_sar = $tot\_repling\_sar / $sar\_row;  $tot\_repling\_rc = training::where('nilai\_repling',$repling)->where('divisi',2)->count();  $p\_repling\_rc = $tot\_repling\_rc / $rc\_row;    $tot\_sk\_survival = training::where('nilai\_sebrang\_kering',$sk)->where('divisi',0)->count();  $p\_sk\_survival = $tot\_gh\_survival / $survival\_row;  $tot\_sk\_sar = training::where('nilai\_sebrang\_kering',$sk)->where('divisi',1)->count();  $p\_sk\_sar = $tot\_sk\_sar / $sar\_row;  $tot\_sk\_rc = training::where('nilai\_sebrang\_kering',$sk)->where('divisi',2)->count();  $p\_sk\_rc = $tot\_sk\_rc / $rc\_row;    $tot\_minat\_survival = training::where('minat',$minat)->where('divisi',0)->count();  $p\_minat\_survival = $tot\_minat\_survival / $survival\_row;  $tot\_minat\_sar = training::where('minat',$minat)->where('divisi',1)->count();  $p\_minat\_sar = $tot\_minat\_sar / $sar\_row;  $tot\_minat\_rc = training::where('minat',$minat)->where('divisi',2)->count();  $p\_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 atribute  // dengan label yg sama dikalikan dengan hasil dari  // pembagian total baris label dan total kesulurahn batis 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 : anwaroye68@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**