**SEMINAR PROPOSAL**

**KLASIFIKASI TINGKAT KEMATANGAN SISTEM PEMERINTAHAN BERBASIS ELEKTRONIK (SPBE) DI KOTA KENDARI MENGGUNAKAN METODE *NAÏVE BAYES***

**Diajukan Untuk Memenuhi**

**Salah Satu Syarat Memperoleh Gelar Sarjana Teknik**

****

**MIFTAHUL JANNAH SALAM**

**E1E1 20 033**

**JURUSAN TEKNIK INFORMATIKA**

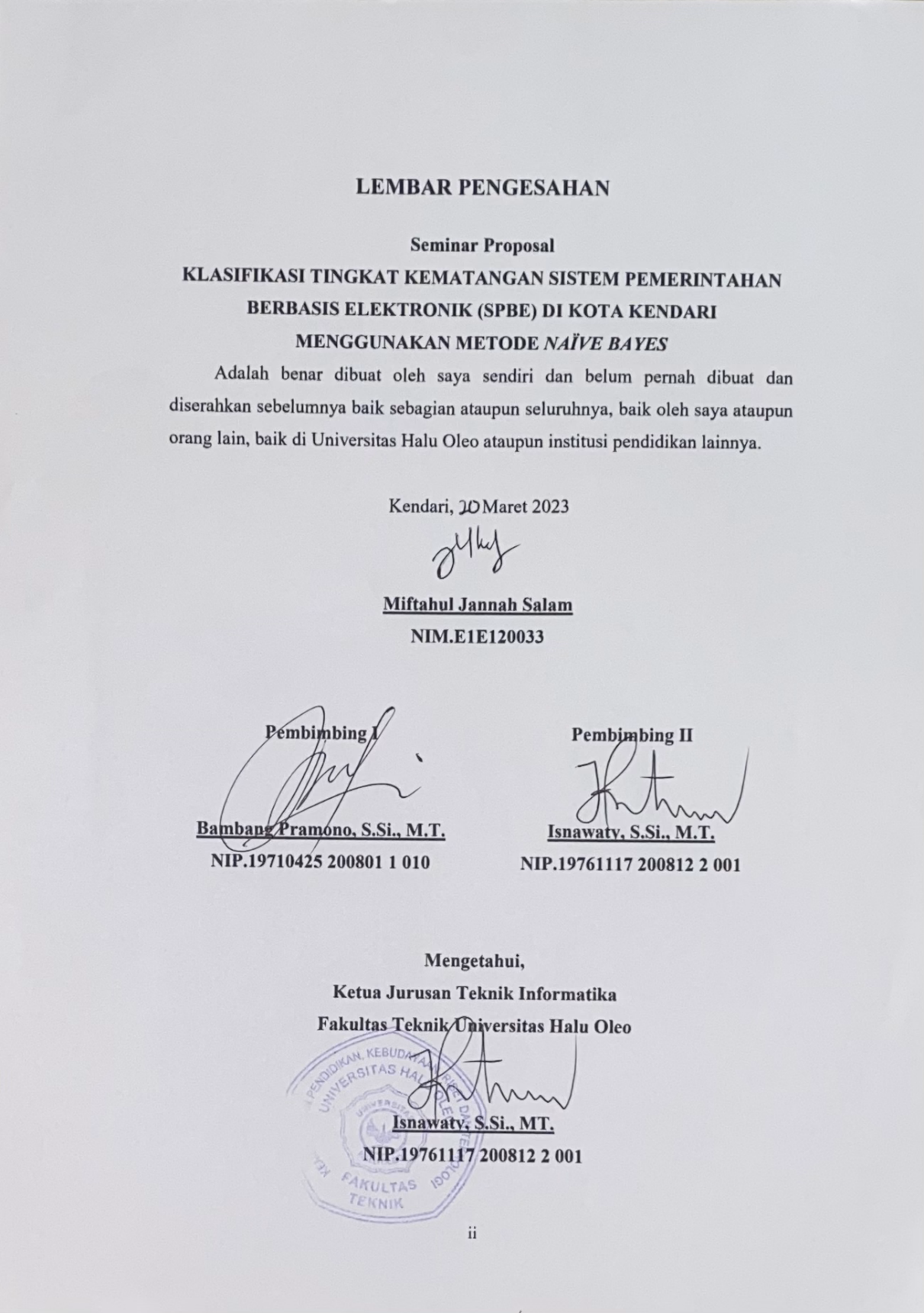
**FAKULTAS TEKNIK**

**UNIVERSITAS HALU OLEO**

**KENDARI**

**2024**

# LEMBAR PENGESAHAN



# INTISARI

**Miftahul Jannah Salam**, E1E120033

**KLASIFIKASI TINGKAT KEMATANGAN SISTEM PEMERINTAHAN BERBASIS ELEKTRONIK (SPBE) DI KOTA KENDARI MENGGUNAKAN METODE *NAÏVE BAYES***

Proposal, Fakultas Teknik, 2024

**Kata Kunci** – Klasifikasi, Tingkat Kematangan Sistem, SPBE, Naïve Bayes, *Website*

Sistem pemerintahan berbasis elektronik (SPBE) telah menjadi salah satu inovasi penting dalam meningkatkan efisiensi dan transparansi dalam administrasi pemerintahan diberbagai kota di seluruh dunia. Pemerintah Kota Kendari telah menerapkan SPBE agar memudahkan dalam proses pelayanan publik. Tidak hanya penerapan SPBE, peningkatan kualitas dan kematangan SPBE juga terus dilakukan. Kematangan SPBE memiliki peran penting dalam meningkatkan efisiensi, transparansi, dan responsivitas pemerintah dalam memberikan layanan kepada masyarakat. Banyaknya faktor yang perlu dinilai dalam mengukur tingkat kematangan SPBE, mengakibatkan penilaian secara manual kurang efektif dan efisien apabila dilakukan.

Menerapkan sistem yang bisa digunakan dalam proses klasifikasi tingkat kematangan SPBE menjadi solusi yang bisa diterapkan untuk memudahkan proses penentuan tingkat kematangan SPBE. Sistem yang dikembangkan berbasis *website* dapat membantu untuk menilai tingkat kematangan SPBE di Kota Kendari. Penelitian ini bertujuan untuk mengkaji tingkat kematangan sistem pemerintahan berbasis elektronik di Kota Kendari menggunakan klasifikasi dengan metode *Naive Bayes*. Metode *Naive Bayes* dipilih karena kemampuannya dalam mengklasifikasikan data berdasarkan probabilitas dan kecocokan dengan dataset. Proses klasifikasi dilakukan berdasarkan beberapa atribut, yaitu kebijakan, layanan, manajemen, dan tata kelola. Data yang digunakan dalam penelitian ini diperoleh dari Dinas Komunikasi dan Informasi Kota Kendari.

Penelitian ini mengambil langkah-langkah pengujian *confusion matrix* untuk menguji tingkat akurasi metode yang digunakan dan pengujian *black box* untuk menguji fungsionalitas fitur-fitur yang ada pada sistem perangkat lunak berbasis *website*.

# *ABSTRACT*

**Miftahul Jannah Salam, E1E120033**

***CLASSIFICATION OF THE MATURITY LEVEL OF ELECTRONIC GOVERNMENT SYSTEMS (EGS) IN KENDARI CITY USING THE NAÏVE BAYES METHOD***

*Proposal, Faculty of Engineering, 2024*

***Keywords*** *– Classification, IT Maturity,* SPBE*,* Naïve Bayes*, Website*

*The electronic government system (e-Government) has become a significant innovation in enhancing efficiency and transparency in governmental administration across various cities worldwide. The Kendari City Government has implemented an e-Government system to facilitate public service processes. Not only the implementation of the e-Government system, but also the improvement of its quality and maturity, continues to be pursued. The maturity of the e-Government system plays a crucial role in enhancing government efficiency, transparency, and responsiveness in providing services to the public. The multitude of factors that need to be evaluated in measuring the maturity level of the e-Government system results in manual assessment being less effective and efficient when performed.*

*Implementing a system that can be used in classifying the maturity level of the e-Government system offers a solution that can simplify the process of determining the maturity level of the e-Government system. The developed system, based on a website, can assist in assessing the maturity level of the e-Government system in Kendari City. This research aims to examine the maturity level of the electronic government system in Kendari City using classification with the Naive Bayes method. The Naive Bayes method is chosen for its ability to classify data based on probabilities and compatibility with the dataset. The classification process is based on several attributes, namely policies, services, management, and governance. The data used in this research is obtained from the Kendari City Department of Communication and Information.*

*This study employs testing steps such as confusion matrix testing to assess the accuracy level of the method used and black box testing to examine the functionality of features present in the website-based software system.*

# DAFTAR ISI

[LEMBAR PENGESAHAN ii](#_Toc165018740)

[INTISARI iii](#_Toc165018741)

[*ABSTRACT* iv](#_Toc165018742)

[DAFTAR ISI v](#_Toc165018743)

[DAFTAR GAMBAR vii](#_Toc165018744)

[DAFTAR TABEL vii](#_Toc165018745)

[BAB I PENDAHULUAN 9](#_Toc165018746)

[1.1 Latar Belakang 9](#_Toc165018747)

[1.2 Rumusan Masalah 12](#_Toc165018748)

[1.3 Batasan Masalah 12](#_Toc165018749)

[1.4 Tujuan Penelitian 12](#_Toc165018750)

[1.5 Manfaat Penelitian 12](#_Toc165018751)

[1.6 Sistematika Penulisan 13](#_Toc165018752)

[1.7 Tinjauan Pustaka 13](#_Toc165018753)

[BAB II LANDASAN TEORI 20](#_Toc165018754)

[2.1 Sistem Pemerintahan Berbasis Elektronik (SPBE) 20](#_Toc165018755)

[2.2 Kematangan Sistem Pemerintahan Berbasis Elektronik 20](#_Toc165018756)

[2.3 Klasifikasi 24](#_Toc165018757)

[2.4 Metode *Naïve Bayes* 24](#_Toc165018758)

[2.5 Hypertext Preprocessor (PHP) 38](#_Toc165018760)

[2.6 Laravel 39](#_Toc165018761)

[2.7 XAMPP 39](#_Toc165018762)

[2.8 MYSQL 39](#_Toc165018763)

[2.9 *Website* 40](#_Toc165018764)

[2.10 HTML 5 40](#_Toc165018765)

[2.11 *Cascading Style Sheet 3* (CSS) 41](#_Toc165018766)

[2.12 Visual Studio Code 41](#_Toc165018767)

[2.13 *Unified Modeling Language* (UML) 42](#_Toc165018768)

[2.14 RUP (*Rational Unified Process*) 47](#_Toc165018770)

[2.15 Pengujian *Black Box* 48](#_Toc165018771)

[2.16 *Confusion Matrix* 49](#_Toc165018772)

[BAB III 51](#_Toc165018773)

[METODOLOGI PENELITIAN 51](#_Toc165018774)

[3. 1 Metode Pengumpulan Data 51](#_Toc165018775)

[3. 2 Metode Pengembangan Sistem 51](#_Toc165018776)

[3. 3 Waktu dan Tempat Penelitian 52](#_Toc165018777)

[2.4 Analisis Kebutuhan Sistem 53](#_Toc165018780)

[2.5 Analisis Peracangan Sistem 55](#_Toc165018783)

[2.6 Analisis Perancangan *User* *Interface* 69](#_Toc165018786)

[2.7 Skenario Pengujian 73](#_Toc165018789)

[DAFTAR PUSTAKA 76](#_Toc165018792)

# DAFTAR GAMBAR

[Gambar 3. 1 Diagram *Flowchart* 56](#_Toc165018805)

[Gambar 3. 2 *Use Case* *Diagram* 57](#_Toc165018806)

[Gambar 3. 3 *Activity Diagram Login* 58](#_Toc165018807)

[Gambar 3. 4 *Activity* *Diagram* Tambah Data SPBE 58](#_Toc165018808)

[Gambar 3. 5 *Activity* *Diagram* Edit Data SPBE 59](#_Toc165018809)

[Gambar 3. 6 *Activity* *Diagram* Hapus Data SPBE 60](#_Toc165018810)

[Gambar 3. 7 *Activity* *Diagram* Tambah Data *Training* 60](#_Toc165018811)

[Gambar 3. 8 *Activity* *Diagram* Edit Data *Training* 61](#_Toc165018812)

[Gambar 3. 9 *Activity* *Diagram* Hapus Data *Training* 62](#_Toc165018813)

[Gambar 3. 10 *Activity* *Diagram* Hasil Klasifikasi Tingkat Kematangan SPBE 62](#_Toc165018814)

[Gambar 3. 11 *Sequence Diagram Login* 63](#_Toc165018815)

[Gambar 3. 12 *Sequence Diagram* Tambah Data Latih 64](#_Toc165018816)

[Gambar 3. 13 *Sequence Diagram Update* Data Latih 64](#_Toc165018817)

[Gambar 3. 14 *Sequence Diagram* Hapus Data SPBE 65](#_Toc165018818)

[Gambar 3. 15 *Sequence Diagram* Tambah Data Latih 66](#_Toc165018819)

[Gambar 3. 16 *Sequence Diagram* *Update* Data Latih 66](#_Toc165018820)

[Gambar 3. 17 *Sequence Diagram* Hapus Data Latih 67](#_Toc165018821)

[Gambar 3. 18 *Sequence Diagram* Halaman Klasifikasi 68](#_Toc165018822)

[Gambar 3. 19 *Class Diagram* 68](#_Toc165018823)

[Gambar 3. 20 Halaman *Login* 69](#_Toc165018824)

[Gambar 3. 21 *Dashboard* Admin 69](#_Toc165018825)

[Gambar 3. 22 Menu Data SPBE 70](#_Toc165018826)

[Gambar 3. 23 *Form Input* Data SPBE 70](#_Toc165018827)

[Gambar 3. 24 Data *Training* 71](#_Toc165018828)

[Gambar 3. 25 *Form Input* Data *Training* 71](#_Toc165018829)

[Gambar 3. 26 Hasil Klasifikasi 72](#_Toc165018830)

[Gambar 3. 27 Hasil Klasifikasi 72](#_Toc165018831)

# DAFTAR TABEL

[Tabel 1. 1 Tinjauan Pustaka 14](#_Toc165018880)

[Tabel 1. 2 Tinjauan Pustaka (Lanjutan) 15](#_Toc165018881)

[Tabel 1. 3 Tinjauan Pustaka (Lanjutan) 16](#_Toc165018882)

[Tabel 1. 4 Tinjauan Pustaka (Lanjutan) 17](#_Toc165018883)

[Tabel 1. 5 Tinjauan Pustaka (Lanjutan) 18](#_Toc165018884)

[Tabel 1. 6 Tinjauan Pustaka (Lanjutan) 19](#_Toc165018885)

[Tabel 2. 1 Domain Kebijakan SPBE 21](#_Toc165018886)

[Tabel 2. 2 Domain Tata Kelola SPBE 22](#_Toc165018887)

[Tabel 2. 3 Domain Manajemen SPBE 22](#_Toc165018888)

[Tabel 2. 4 Domain Manajemen SPBE (Lanjutan) 23](#_Toc165018889)

[Tabel 2. 5 Domain Layanan SPBE 23](#_Toc165018890)

[Tabel 2. 6 Predikat Penilaian 24](#_Toc165018891)

[Tabel 2. 7 Data Penilaian Aplikasi SPBE 26](#_Toc165018892)

[Tabel 2. 8 *Preprocessing* Data 26](#_Toc165018893)

[Tabel 2. 9 *Preprocessing* Data (Lanjutan) 27](#_Toc165018894)

[Tabel 2. 10 Data *Testing* 37](#_Toc165018895)

[Tabel 2. 11 Data *Preprocessing* 37](#_Toc165018896)

[Tabel 2. 12 Simbol *Use Case Diagram* 42](#_Toc165018897)

[Tabel 2. 13 Simbol *Use Case Diagram* (Lanjutan) 43](#_Toc165018898)

[Tabel 2. 14 Simbol *Activity Diagram* 43](#_Toc165018899)

[Tabel 2. 15 Simbol *Activity Diagram* (Lanjutan) 44](#_Toc165018900)

[Tabel 2. 16 Simbol *Class Diagram* 44](#_Toc165018901)

[Tabel 2. 17 Simbol *Class Diagram* (Lanjutan) 45](#_Toc165018902)

[Tabel 2. 18 Simbol *Sequence Diagram* 46](#_Toc165018903)

[Tabel 2. 19 Simbol *Sequence Diagram* (Lanjutan) 47](#_Toc165018904)

[Tabel 2. 20 *Confusion Matrix* 49](#_Toc165018905)

[Tabel 3. 1 Waktu Penelitian 53](#_Toc165018906)

[Tabel 3. 2 Perangkat Keras 54](#_Toc165018907)

[Tabel 3. 3 Perangkat Lunak 55](#_Toc165018908)

[Tabel 3. 4 Contoh Tabel *Confusion Matrix* 73](#_Toc165018909)

[Tabel 3. 5 Pengujian *Black Box* 74](#_Toc165018910)

[Tabel 3. 6 Pengujian *Black Box* (Lanjutan) 75](#_Toc165018911)

# BAB I PENDAHULUAN

## Latar Belakang

Perkembangan teknologi informasi yang cukup pesat saat ini menyebabkan berbagai bidang kehidupan masyarakat mengalami perubahan. Salah satunya adalah dibidang pelayanan publik. Pelayanan publik merupakan segala bentuk pelayanan yang diberikan oleh pemerintah baik yang diselenggarakan oleh lembaga pemerintah itu sendiri ataupun oleh lembaga non-pemerintah untuk memenuhi kebutuhan masyarakat maupun pelaksanaan ketentuan yang telah ditetapkan dengan segala sarana dan perlengkapannya melalui prosedur kerja tertentu guna memberikan jasa pelayanan dalam bentuk barang dan jasa (Suryantoro dan Kusdyana, 2020). Dalam menghadapi era globalisasi dan perkembangan teknologi informasi, menyebabkan munculnya pertimbangan bagi pemerintah untuk meningkatkan kualitas pelayanan publik dengan memanfaatkan teknologi informasi. Tidak hanya itu, banyaknya tuntutan masyarakat, seperti aksesibilitas pelayanan publik yang mudah, cepat, dan efisien menjadi pemicu utama bagi pemerintah untuk menetapkan strategi dalam mengoptimalkan pelaksanaan layanan publik.

Berdasarkan hal tersebut, dibutuhkan sistem informasi pemerintahan yang dapat menjadi solusi yang tepat untuk meningkatkan efektivitas, efisiensi, dan aksesibilitas dalam pelaksanaan pelayanan public, yaitu *e-governement* atau Sistem Pemerintahan Berbasis Elektronik (SPBE). Sistem Pemerintahan Berbasis Elektronik (SPBE) adalah penyelenggaraan pemerintahan yang memanfaatkan teknologi informasi dan komunikasi untuk memberikan layanan kepada pengguna SPBE (Natuna, 2022). Berdasarkan Peraturan Presiden Republik Indonesia Nomor 95 Tahun 2018 yang menyebutkan bahwa semua instansi pemerintah harus menggunakan Sistem Pemerintahan Berbasis Elektronik (SPBE) sebagai bentuk integrasi *e-government* (Rusdy dan Flambonita, 2023).

Pemerintah Kota Kendari, Sulawesi Tenggara telah menerapkan Sistem Pemerintahan Berbasis Elektronik pada berbagai bidang pelayanan publik. Dinas

Komunikasi dan Informasi Kota Kendari telah mengembangkan beberapa aplikasi SPBE, seperti LAIKA, Kendari Preneur, Jaga Kendari, E-Katalog Kendari, dan sebagainya. Tidak hanya melakukan penerapan saja, namun, peningkatan kematangan dari SPBE juga harus terus dilakukan agar sistem bisa lebih optimal selama proses penggunaannya. Penilaian tingkat kematangan SPBE dinilai berdasarkan berbagai aspek yang telah ditentukan. Berdasarkan hasil keputusan Menteri Pendayagunaan Aparatur Negara dan Reformasi Birokrasi Republik Indonesia Nomor 13 Tahun 2024 mengenai evaluasi tingkat kematangan pada instansi pusat dan pemerintah daerah melalui pemantauan penilaian yang dilakukan oleh Dinas Komunikasi dan Informasi Kota Kendari tiap tahunnya, Kota Kendari memperoleh nilai 2,37 dengan predikat cukup pada tahun 2020, tahun 2021 dengan nilai 1,79 predikat kurang, tahun 2022 dengan nilai 2,02 predikat cukup, dan tahun 2023 dengan nilai 2,42 predikat cukup. Tingkat kematangan tersebut tentunya masih perlu ditingkatkan lagi dengan lebih mengoptimalkan Sistem Pemerintahan Berbasis Elektronik (SPBE) yang ada di Kota Kendari agar bisa mencapai predikat memuaskan.

Penelitian tingkat kematangan SPBE dilakukan tiap tahunnya sekaligus sebagai bahan evaluasi agar SPBE dapat lebih dioptimalkan. Dalam penilaian tingkat kematangan SPBE, mengharuskan tim penilai menganalisis data dan mengelompokkannya berdasarkan aspek-aspek yang telah ditentukan. Berdasarkan hasil penilaian SPBE tiap tahunnya, Kota Kendari masih memperoleh predikat tingkat kematangan di bawah Baik. Salah satu langkah yang dapat digunakan untuk membantu menentukan tingkat kematangan SPBE adalah memprediksi tingkat kematangan tiap SPBE Kota Kendari sebelum memasuki masa evaluasi penilaian pusat. Hal ini akan membantu Dinas Komunikasi dan Informasi Kota Kendari untuk mengetahui perkiraan penilaian tingkat kematangan SPBE lebih awal sebelum penilaian resmi dikeluarkan oleh pemerintah pusat sehingga dapat ditentukan solusi lebih lanjut untuk meningkatkan penilaian kematangan SPBE. Jika SPBE yang akan ditentukan tingkat kematangannya semakin banyak, hal ini akan mengakibatkan kurang efisiennya waktu yang digunakan. Oleh karena itu, dibutuhkan sistem terkomputerisasi yang dapat membantu dalam mengolah dan mengelompokkan data penilaian tingkat kematangan Sistem Pemerintahan Berbasis Elektronik (SPBE) di Kota Kendari secara cepat dan akurat.

Klasifikasi adalah proses pengelompokkan objek atau data ke dalam kategori-kategori atau kelas-kelas berdasarkan karakteristik atau atribut tertentu yang dimiliki oleh objek tersebut. Klasifikasi adalah suatu fungsionalitas *machine learning* yang menghasilkan model untuk memprediksi kelas atau kategori dari objek - objek di dalam basis data (Heliyanti Susana, 2022). Tujuan dari klasifikasi adalah mengelompokkan data yang beragam menjadi kategori-kategori yang lebih terstruktur, memudahkan proses analisis, dan pengambilan keputusan. Salah satu metode dari klasifikasi yang paling sering digunakan adalah metode *Naïve Bayes*.

*Naive Bayes* adalah salah satu metode klasifikasi sederhana berdasarkan probabilitas yang dapat menghitung seluruh kemungkinan dari setiap label berdasarkan data *training* yang ada. Suatu algoritma memanfaatkan teorema *Bayes* dan memperkirakan seluruh atribut yang akan menjadi variabel penentuan kelas. Keuntungan penggunaan metode ini adalah hanya membutuhkan jumlah data pelatihan (*training data*) yang kecil untuk menentukan parameter atau variabel yang diperlukan dalam proses pengklasifikasiannya (Rachman dan Handayani, 2021). *Naive Bayes* dipilih sebagai metode klasifikasi karena kemampuannya dalam mengatasi data dengan beberapa fitur atau atribut, sehingga diharapkan dapat memberikan kontribusi signifikan dalam meningkatkan efektivitas dalam pengklasifikasian sistem pemerintahan berbasis elektronik di Kota Kendari.

Berdasarkan latar belakang masalah yang telah diuraikan, maka dilakukan penelitian dengan judul **“Klasifikasi Tingkat Kematangan Sistem Pemerintahan Berbasis Elektronik (SPBE) Di Kota Kendari Menggunakan Metode *Naïve Bayes*”**. Hasil dari penelitian ini diharapkan dapat memberikan pemahaman yang lebih dalam tentang tingkat kematangan sistem pemerintahan berbasis elektronik di Kota Kendari dan memberikan kemudahan bagi pengguna dalam mengklasifikasikan tingkat kematangan sistem sehingga lebih efektif dan efisien dalam prosesnya.

## Rumusan Masalah

Berdasarkan latar belakang yang telah diuraikan, rumusan masalah dari penelitian ini adalah bagaimana merancang dan membangun sebuah sistem untuk mengklasifikasikan tingkat kematangan Sistem Pemerintahan Berbasis Elektronik (SPBE) di Kota Kendari menggunakan metode *Naïve Bayes*?

## Batasan Masalah

Hal-hal yang menjadi batasan masalah dalam penelitian ini adalah sebagai berikut.

1. Penelitian ini dilakukan di Kantor Dinas Komunikasi dan Informasi Kota Kendari, Sulawesi Tenggara.
2. Klasifikasi Tingkat kematangan Sistem Pemerintahan Berbasis Elektronik (SPBE) dilakukan menggunakan metode *Naïve Bayes.*
3. Aplikasi yang dirancang dan dibangun berbasis *website* karena penggunaan sistem hanya dilakukan ketika menentukan tingkat kematangan Sistem Pemerintahan Berbasis Elektronik (SPBE).

## Tujuan Penelitian

Tujuan yang diharapkan pada pembuatan sistem klasifikasi tingkat kematangan Sistem Pemerintahan Berbasis Elektronik (SPBE) dengan menggunakan metode *Naïve Bayes* adalah sebagai berikut.

1. Membangun sistem klasifikasi tingkat kematangan Sistem Pemerintahan Berbasis Elektronik (SPBE) berbasis *website*.
2. Menerapkan metode *Naïve Bayes* dalam klasifikasi tingkat kematangan Sistem Pemerintahan Berbasis Elektronik (SPBE).

## Manfaat Penelitian

Manfaat dari pembuatan sistem klasifikasi tingkat kematangan Sistem Pemerintahan Berbasis Elektronik (SPBE) dengan menggunakan metode *Naïve Bayes* adalah sebagai berikut.

1. Dapat menghasilkan sebuah sistem yang mampu mengklasifikasikan tingkat kematangan Sistem Pemerintahan Berbasis Elektronik (SPBE) di Kota Kendari.
2. Dapat mengklasifikasikan data dengan nilai akurasi tinggi menggunakan metode *Naïve Bayes*.
3. Dapat mempermudah karyawan di Dinas Komunikasi dan Informasi Kota Kendari dalam menentukan tingkat kematangan Sistem Pemerintahan Berbasis Elektronik (SPBE) di Kota Kendari.

## Sistematika Penulisan

Sistematika penulisan tugas akhir ini digunakan untuk menjelaskan penulisan perbab sebagai berikut.

**BAB I PENDAHULUAN**

Bagian ini berisi pendahuluan yang terdiri dari latar belakang, rumusan masalah, tujuan penelitian, manfaat penelitian, sistematika penulisan dan tinjauan pustaka.

**BAB II LANDASAN TEORI**

Bab ini membahas teori-teori penunjang yang berhubungan dengan penelitian berupa teori dari metode yang digunakan.

**BAB III METODE PENELITIAN**

Bagian ini menyediakan langkah-langkah pengumpulan data, prosedur pengembangan perangkat lunak yang dilakukan dalam penelitian menggunakan bentuk kalimat pasif yang meliputi prosedur dan pengumpulan data, prosedur pengembangan perangkat lunak dan waktu serta tempat penelitian.

## Tinjauan Pustaka

Penyusunan proposal penelitian ini menggunakan beberapa jurnal penelitian terdahulu sebagai referensi yang membahas mengenai klasifikasi menggunakan metode *Naïve Bayes*. Uraian penelitian dapat dilihat pada Tabel 1.1.

**Tabel 1. 1 Tinjauan Pustaka**

|  |  |  |
| --- | --- | --- |
| **No.** | **Judul** | **Uraian Penelitian** |
| 1. | Klasifikasi Algoritma *Naive Bayes* Dalam Memprediksi Tingkat Kelancaran Pembayaran Sewa Teras UMKM | Penelitian ini dilakukan dengan menggunakan metode *Naïve Bayes* dalam mengklasifikasikan 504 data berdasarkan 13 atribut yang diklasifikasikan ke dalam 6 atribut, yaitu *area manager*, kota, jenis kelamin penyewa, rata-rata umur penyewa, dan status pembayaran. Proses pengklasifikasian yang dilakukan adalah pengumpulan data, data *cleaning*, penerapan metode *Naive Bayes*, pengaplikasian model, *output* hasil, dan evaluasi. Setelah diuji menggunakan metode *Naïve Bayes*, diperoleh nilai *accuracy* 81.81% (Rachman dan Handayani, 2021). |
| 2. | Analisis Algoritma *Naive Bayes Classifier* (NBC) Pada Klasifikasi Tingkat Minat Barang Di Toko Violet Cell | Metode *Naïve Bayes* digunakan dalam klasifikasi tingkat minat paket kuota internet. Data yang digunakan adalah data kuantitatif penjualan selama dua bulan terakhir yang kemudian diklasifikasikan. Proses klasifikasi dilakukan dengan tahapan pengumpulan data, *preprocessing* data berupa pemberian label pada data dan *cleaning* untuk menghapus kolom data yang tidak dibutuhkan, dan dilakukan penerapan algoritma Naive Bayes. Hasil 23 barang memiliki kategori “minat rendah”, 5 barang dengan kategori “minat sedang”, dan 1 barang memiliki kategori “minta tinggi”. Tingkat akurasi yang diperoleh mengunakan metode ini adalah 82,76% (Sigid Widodo dkk., 2023). |

**Tabel 1. 2 Tinjauan Pustaka (Lanjutan)**

|  |  |  |
| --- | --- | --- |
| 3. | Implementasi Metode *Naïve Bayes* Dalam Penentuan Tingkat Kesejahteraan Keluarga | Metode *Naïve Bayes* digunakan untuk mengolah data kesejahteraan penduduk dikarenakan seringkali ditemukan kesalahan dalam penentuan penerima bantuan yang dilakukan secara manual. Tahapan yang dilakukan adalah memasukkan variabel data input sesuai dengan ketentuan. Kemudian, sistem menghitung *class* probabilitas yang telah ditentukan kemudian mencari *conditional* probabilitas berdasarkan pada variabel dengan *class* yang sesuai. Selanjutnya, sistem melakukan perkalian hasil dari keseluruhan *conditional* probabilitas pada setiap variabel input sesuai dengan *class* masing- masing dengan nilai dari *class* probabilitas. Sistem akan menampilkan hasil prediksi dari perhitungan menggunakan data yang diperoleh dari Badan Pusat Statistik Kota Kediri yang kemudian akan diklasifikasikan ke dalam empat kelas, yaitu tingkat kesejahteraan dibawah 10%, tingkat kesejahteraan dengan rentang 11% sampai dengan 20%, tingkat kesejahteraan dengan rentang 21% sampai dengan 30% dan tingkat kesejahteraan dengan rentang 31% sampai dengan 40%. Tingkat akurasi yang diperoleh ketika menggunakan metode ini adalah 87% (Hakim dan Irawati, 2022). |
| 4. | Penggunaan Metode *Naïve Bayes* Dalam Mengukur Tingkat | Klasifikasi tingkat kepuasan pengguna digunakan metode *Naïve Bayes* untuk mengolah data yang diperoleh dari kuisioner. Atribut yang digunakan antara lain kualitas konten, relevan, |

**Tabel 1. 3 Tinjauan Pustaka (Lanjutan)**

|  |  |  |
| --- | --- | --- |
|  | Kepuasan Pengguna Terhadap *Online System* Universitas Advent Indonesia | privasi, mudah dioperasikan, kecepatan, daya tarik visual, kelengkapan secara *online*, dan *customer service*. Hasil pengujian menggunakan metode *Naïve Bayes* adalah 81.3%, pengujian kedua dengan 80% data *training* 20% data *testing* mendapatkan nilai akurasi 80%, pengujian ketiga dengan *cross-validation* mendapatkan nlai akurasi sebesar 78.7 dan pengujian keempat dengan 66% data *training* dan 33% data *testing* mendapatkan nilai akurasi 68.6% (Samuel dan Dewi, 2019). |
| 5. | Penerapan Algoritma Klasterisasi dan Klasifikasi pada Tingkat Kepentingan Sistem Pembelajaran di Universitas Terbuka | Penelitian menerapkan beberapa metode pada proses klasifikasi, yaitu *Naïve Bayes*, k-NN, dan C4.5. Tahapan yang dilakukan antara lain, *bussiness understanding*, *data understanding*, *data preparation*, *modelling*, dan *evaluation*. Berdasarkan ketiga metode tersebut, *Naïve Bayes* memperoleh nilai akurasi paling tinggi yaitu 72,70% dengan nilai AUC sebesar 0.499. sedangkan metode lainnya, yaitu k-NN memperoleh tingkat akurasi 71,62% sebesar 0,438 dan algoritma C4.5 memperoleh nilai akurasi sebesar 68,92% dengan nilai AUC sebesar 0,450 (Hidayati dkk., 2020). |
| 6. | Implementasi Machine Learning dengan Algoritma Naive Bayes Terhadap Sistem | Penelitian dilakukan karena tidak efektif dan efisiennya pemberkasan kepegawaian di BKPSDM Kota Palembang. Klasifikasi ini dilakukan dengan menggunakan metode *Naïve Bayes*. Tahapan klasifikasi adalah *input* data, *text* |

**Tabel 1. 4 Tinjauan Pustaka (Lanjutan)**

|  |  |  |
| --- | --- | --- |
|  | Informasi Pelayanan Pemberkasan Kepegawaian di BKPSDM Kota Palembang | *preprocessing*, *words weighting*, proses klasifikasi, dan *ouput* data. Data diklasifikasikan ke dalam 6 kategori, yaitu pengajuan kenaikan pangkat, pengajuan mutasi, usulan pensiun, permohonan tugas belajar dan diklat, perizinan cuti, dan peremajaan data. Tingkat akurasi yang diperoleh menggunakan metode ini adalah 95,83% (Algiffary, 2023). |
| 7. | *The Classification of Phishing Websites using Naïve Bayes Classifier Algorithm* | Peneliti menggunakan metode *Naïve Bayes Classifier.* Tujuan dari penelitian ini adalah bagaimana mengetahui data yang diuji melalui data pelatihan akan menghasilkan data baru, terutama untuk item dan menghasilkan nilai dari algoritma *Naive Bayes*. Tahapan klasifikasi dilakukan dengan *training* data, *preprocessing* data, *testing* data, proses klasifikasi, dan *ouput* hasil klasifikasi. Hasil yang diperoleh dengan nilai akurasi rata-rata sebesar 92,98% dengan TP *Rate* sebesar 0,930%, FP *Rate* sebesar 0,076%, Presisi sebesar 0,930%, *Recall* sebesar 0,930%, dan *F-measure* sebesar 0,930% (Roni Anagora dkk., 2022). |
| 8. | Klasifikasi Tingkat Kematangan Buah Pisang Kepok Menggunakan Algoritma *Naive Bayes* | Data yang ada akan dianalisis menggunakan algoritma *Naive Bayes*. *Naive Bayes* merupakan salah satu metode pada *probabilistic reasoning*. Algoritma *Naive Bayes* bertujuan untuk melakukan klasifikasi data pada kelas tertentu, kemudian pola tersebut dapat digunakan untuk memperkirakan pisang yang diproses, sehingga |

**Tabel 1. 5 Tinjauan Pustaka (Lanjutan)**

|  |  |  |
| --- | --- | --- |
|  |  | toko pisang tersebut bisa mengambil keputusan menerima atau menolak pisang tersebut. Atribut yang digunakan adalah biji cabai, intensitas cahaya, suhu, dan warna. Hasil dari penelitian ini diperoleh tingkat akurasi sebesar 81% (Hakim dkk., 2022). |
| 9. | Penerapan Model Klasifikasi Metode *Naive Bayes* Terhadap Penggunaan Akses Internet | Penelitian ini menggunakan metode *Naïve Bayes*. Tujuan penelitian dalam paper ini adalah menerapkan algoritma *Naïve Bayes* untuk menganalisis hak akses internet siswa, dari penerapan metode tersebut dapat dilihat akurasi kemudian dapat menganalisis pemakaian berdasarkan umur. Tahapan yang dilakukan adalah pengumpulan data, data *cleaning*, data *transformation*, penerapan *Naive Bayes*, dan *evaluation*. Hasil dari penelitian ini memiliki akurasi sebesar 89.83% Hasil Prediksi Ya dan ternyata Ya sebanyak 34. Hasil Prediksi Ya dan ternyata Tidak sebanyak 6. Hasil Prediksi tidak dan ternyata Ya sebanyak 0. Hasil Prediksi tidak dan ternyata tidak sebanyak 19. hasil dari prediksi dengan uji 59 data baru maka mendapatkan hasil ya sebanyak 40 siswa dan tidak ada 19 siswa (Heliyanti Susana, 2022). |

**Tabel 1. 6 Tinjauan Pustaka (Lanjutan)**

|  |  |  |
| --- | --- | --- |
| 10. | Klasifikasi Keputusan Investasi Di Masa Pandemi Covid-19 Dengan Menggunakan *Naive Bayes* | Penelitian ini menggunakan metode *Naïve Bayes*. Klasifikasi dilakukan ke dalam beberapa kelas tertentu. Hasil penelitian menunjukkan bahwa produk investasi yang menjadi mayoritas pilihan responden pada era pandemi adalah emas dan reksadana. Hasil analisis dengan menggunakan *Naïve Bayes* terhadap 6 variabel posterior memberikan beberapa deskripsi, diantaranya nilai probabilitas prior untuk tidak menambah investasi lebih besar dari menambah investasi. Kelompok investor yang menambah investasi memiliki pola pilihan instrumen investasi yaitu reksadana, emas, deposito dan saham (Rinestu dkk., 2022). |

Dengan demikian, berdasarkan penjelasan mengenai beberapa penelitian terdahulu terkait Metode *Naïve Bayes* yang sudah berhasil diimplementasikan dalam beberapa kasus, maka akan dilakukan penelitian lebih lanjut lagi mengenai Metode *Naïve Bayes* dengan judul Klasifikasi Tingkat Kematangan Sistem Pemerintahan Berbasis Elektronik (SPBE) Di Kota Kendari Menggunakan Metode *Naïve Bayes*.

# BAB II LANDASAN TEORI

## Sistem Pemerintahan Berbasis Elektronik (SPBE)

Sistem Pemerintahan Berbasis Elektronik (SPBE) atau juga disebut *i-Government* (*Integrated Government*) adalah sebuah pemanfaatan teknologi informasi yang diterapkan di sistem pemerintahan sebagai bentuk penyelenggaraan dan peningkatan pelayanan bagi masyarakat. SPBE dilaksanakan dengan prinsip efektivitas, keterpaduan, kesinambungan, efisiensi, akuntabilitas, interoperabilitas, dan keamanan (Taqiya dkk., 2020). Sistem pemerintahan berbasis elektronik adalah konsep di mana pemerintah menggunakan teknologi informasi dan komunikasi (TIK) untuk mengelola proses administrasi publik, mengambil keputusan, dan berinteraksi dengan masyarakat serta sektor lainnya. Ini melibatkan integrasi teknologi dalam berbagai aspek pemerintahan, mulai dari penyediaan layanan publik online hingga pengelolaan data dan komunikasi internal.

Sistem pemerintahan berbasis elektronik bertujuan untuk meningkatkan efisiensi, responsivitas, dan kualitas layanan pemerintah, serta memperkuat hubungan antara pemerintah dan masyarakat dalam era digital yang semakin terkoneksi. Implementasi yang sukses memerlukan integrasi teknologi yang cermat, manajemen perubahan yang efektif, serta komitmen untuk menjaga keamanan dan privasi data. Saat ini, Pemerintah Indonesia menggunakan SPBE dengan tujuan mengukur kebijakan TIK dalam *e-government*. Penerapan SPBE di pemerintahan dan Badan Usaha Milik Negara (BUMN) sudah diatur dalam Peraturan Presiden Nomor 95 Tahun 2018 (Putri dkk., 2023).

## Kematangan Sistem Pemerintahan Berbasis Elektronik

Kematangan sistem pemerintahan berbasis elektronik mengacu pada tingkat kemajuan dan kesiapan suatu pemerintahan dalam mengimplementasikan Teknologi Informasi dan Komunikasi (TIK) secara efektif dalam operasi dan layanan publiknya. Hal ini mencakup berbagai aspek, mulai dari infrastruktur teknologi yang tersedia hingga kesiapan organisasional dan budaya untuk

mengadopsi dan memanfaatkan teknologi secara optimal. *E-Government Maturity Model* (eMM) merupakan model yang dikembangkan dengan tujuan mengukur kondisi *e-government* dalam kondisi matang. Kematangan *e-government* dapat diukur dari beberapa hal yaitu teknologi, operasional organisasi, kemampuan sumber daya dan proses dari organisasi itu sendiri (Bouty dkk., 2019).

Sistem pemerintahan yang matang secara elektronik ditandai oleh integrasi yang solid dari teknologi dalam penyediaan layanan publik, manajemen data yang efisien, keamanan informasi yang terjamin, transparansi yang tinggi, partisipasi masyarakat yang aktif, serta kemampuan untuk terus beradaptasi dengan perkembangan teknologi dan kebutuhan masyarakat. Kematangan tersebut juga mencakup kemampuan pemerintah untuk mengelola risiko teknologi dan mencapai hasil yang diharapkan dalam hal efisiensi, efektivitas, dan pemberdayaan masyarakat. Kematangan SPBE ditentukan berdasarkan beberapa domain. Berikut domain-domain yang digunakan sebagai faktor penilaian tingkat kematangan SPBE (Hidayah dan Almadani, 2022).

1. Domain Kebijakan SPBE

**Tabel 2. 1 Domain Kebijakan SPBE**

|  |  |
| --- | --- |
| **Domain 1 - Kebijakan SPBE** | |
| **Aspek 1 - Kebijakan Internal Tata Kelola SPBE** | |
| Indikator 1 | Kebijakan Internal Arsitektur SPBE Instansi Pusat/Pemerintah |
| Indikator 2 | Kebijakan Internal Peta Rencana SPBE Instansi Pusat/Pemerintah |
| Indikator 3 | Kebijakan Internal Manajemen Data |
| Indikator 4 | Kebijakan Internal Pembangunan Aplikasi SPBE |
| Indikator 5 | Kebijakan Internal Layanan Pusat Data |
| Indikator 6 | Kebijakan Internal Layanan Jaringan Intra Instansi |
| Indikator 7 | Kebijakan Internal Penggunaan Sistem Penghubung Layanan |
| Indikator 8 | Kebijakan Internal Manajemen Keamanan Informasi |
| Indikator 9 | Kebijakan Internal Audit TIK |
| Indikator 10 | Kebijakan Internal Tim Koordinasi SPBE Instansi |

1. Domain Tata Kelola SPBE

**Tabel 2. 2 Domain Tata Kelola SPBE**

|  |  |
| --- | --- |
| **Domain 2 - Tata Kelola SPBE** | |
| **Aspek 1 - Perencanaan Strategis** | |
| Indikator 1 | Arsitektur SPBE Instansi Pusat/Pemerintah Daerah |
| Indikator 2 | Peta Rencana SPBE Instansi Pusat/Pemerintah Daerah |
| Indikator 3 | Rencana dan Anggaran SPBE |
| Indikator 4 | Inovasi Proses Bisnis SPBE |
| **Aspek 2 -** **Teknologi Informasi dan Komunikas** | |
| Indikator 1 | Pembangunan Aplikasi SPBE |
| Indikator 2 | Layanan Pusat Data |
| **Aspek 3 - Penyelenggaraan SPBE** | |
| Indikator 1 | Layanan Jaringan Intra Instansi Pusat/Pemerintah Daerah |
| Indikator 2 | Penggunaan Sistem Penghubung Layanan Instansi |
| Indikator 3 | Tim Koordinasi SPBE Instansi Pusat/Pemerintah Daerah |
| Indikator 4 | Kolaborasi Penerapan SPBE |

1. Domain Manajemen SPBE

**Tabel 2. 3 Domain Manajemen SPBE**

|  |  |
| --- | --- |
| **Domain 3 - Manajemen SPBE** | |
| **Aspek 1 - Penerapan Manajemen SPBE** | |
| Indikator 1 | Penerapan Manajemen Risiko SPBE |
| Indikator 2 | Penerapan Manajemen Keamanan Informasi |
| Indikator 3 | Penerapan Manajemen Data |
| Indikator 4 | Penerapan Manajemen Aset TIK |
| Indikator 5 | Penerapan Kompetensi Sumber Daya Manusia |
| Indikator 6 | Penerapan Manajemen Pengetahuan |
| Indikator 7 | Penerapan Manajemen Perubahan |
| Indikator 8 | Penerapan Manajemen Layanan SPBE |

**Tabel 2. 4 Domain Manajemen SPBE (Lanjutan)**

|  |  |
| --- | --- |
| **Aspek 2 -** **Audit TIK** | |
| Indikator 1 | Pelaksanaan Audit Infrastruktur SPBE |
| Indikator 2 | Pelaksanaan Audit Aplikasi SPBE |
| Indikator 3 | Pelaksanaan Audit Keamanan SPBE |

1. Domain Layanan SPBE

**Tabel 2. 5 Domain Layanan SPBE**

|  |  |
| --- | --- |
| **Domain 3 - Layanan SPBE** | |
| **Aspek 1 - Layanan Administrasi Pemerintahan Berbasis Elektronik** | |
| Indikator 1 | Layanan Perencanaan |
| Indikator 2 | Layanan Penganggaran |
| Indikator 3 | Layanan Keuangan |
| Indikator 4 | Layanan Pengadaan Barang dan Jasa |
| Indikator 5 | Layanan Kepegawaian |
| Indikator 6 | Layanan Kearsipan Dinamis |
| Indikator 7 | Layanan Pengelolaan Barang Milik Negara / Daerah |
| Indikator 8 | Layanan Pengawasan Internal Pemerintah |
| Indikator 9 | Layanan Akuntabilitas Kinerja Organisasi |
| Indikator 10 | Layanan Kinerja Pegawai |
| **Aspek 2 -** **Layanan Publik Berbasis Elektronik** | |
| Indikator 1 | Layanan Pengaduan Pelayanan Publik |
| Indikator 2 | Layanan Data Terbuka |
| Indikator 3 | Layanan Jaringan Dokumentasi dan Informasi Hukum (JDIH) |
| Indikator 4 | Layanan Publik Sektor 1 |
| Indikator 5 | Layanan Publik Sektor 2 |
| Indikator 6 | Layanan Publik Sektor 3 |

Berdasarkan indikator penilaian yang telah ditentukan, penilaian tingkat kematangan SPBE akan dikelompokkan ke dalam beberapa predikat penilaian yang dapat dilihat pada Tabel 2.6 berikut.

**Tabel 2. 6 Predikat Penilaian**

|  |  |  |
| --- | --- | --- |
| **No** | **Nilai Indeks** | **Predikat** |
|  | 4,2 – 5,0 | Memuaskan |
|  | 3,5 – < 4,2 | Sangat Baik |
|  | 2,6 – < 3,5 | Baik |
|  | 1,8 – < 2,6 | Cukup |
|  | < 1,8 | Kurang |

## Klasifikasi

*Machine learning* mempunyai salah satu teknik yaitu teknik klasifikasi yang berguna untuk mengklasifikasikan suatu nilai dari target variabel kategori. Klasifikasi adalah suatu fungsionalitas *machine learning* yang menghasilkan model untuk memprediksi kelas atau kategori dari objek - objek di dalam basis data. Klasifikasi merupakan proses yang terdiri dari dua tahap, yaitu tahap pembelajaran dan tahap pengklasifikasian. Pada tahap pembelajaran, sebuah algoritma klasifikasi akan membangun sebuah model klasifikasi dengan cara menganalisis *training data* (Heliyanti Susana, 2022).

## Metode *Naïve Bayes*

*Naive Bayes* adalah suatu klasifikasi kemungkinan sederhana yang dapat menghitung seluruh kemungkinan dengan menggabungkan sejumlah kombinasi dan frekuensi suatu nilai dari basis data yang didapatkan. Suatu algoritma memanfaatkan *Teorema Bayes* dan memperkirakan seluruh atribut yang bebas dan saling lepas yang dapat diberikan oleh suatu nilai pada kelas variabel. *Naïve Bayes* adalah klasifikasi dengan suatu metode kemungkinan dan perhitugan yang ditemukan oleh seseorang ilmuwan dari Inggris yaitu Thomas Bayes yang menghasilkan prediksi peluang yang akan datang berdasarkan suatu pengalaman sebelumnya (Rachman dan Handayani, 2021).

*Naïve Bay*es mengategorikan dengan metode probabilitas dan statistik. Jadi, didalam penerapannya, algoritma *Naive Bayes Classifier* tidak ada hubungan antara satu atribut dengan atribut lain, atau dengan kata lain satu atribut tidak berpengaruh dengan atribut yang lain, sekalipun mungkin atribut tersebut saling berhubungan (Sigid Widodo dkk., 2023). Ciri utama dari *Naïve Bayes* *Classifier* adalah asumsi yang sangat kuat (*naïf*) tentang independensi setiap kondisi/peristiwa. Algoritma ini mengasumsikan bahwa atribut objek adalah independen. Probabilitas yang terlibat dalam menghasilkan perkiraan akhir dihitung sebagai kuantitas frekuensi dari tabel hasil keputusan (Rayuwati dkk., 2022).

Tahapan yang dilakukan dalam proses klasifikasi menggunakan metode *Naive Bayes* adalah sebagai berikut (Martantoh dan Yanih, 2022).

* 1. Melakukan pengumpulan data set yang telah memiliki label sebelumnya.
  2. Membagi data set menjadi dua bagian, yaitu data *training* dan data *testing*. Data *training* digunakan untuk melatih model algoritma dari *Naive Bayes* sedangkan data *testing* adalah data yang digunakan untuk menguji model dari *Naive Bayes*.
  3. Menentukan atribut yang akan digunakan dalam proses klasifikasi. Atribut digunakan agar dapat membedakan kelas-kelas dari setiap data yang ada.
  4. Menghitung probabilitas untuk setiap kelas menggunakan persamaan *Naive Bayes*. Bentuk umum atau persamaan *Theorema Bayes* adalah sebagai berikut (Rinestu dkk., 2022).

(2.1)

* 1. Melakukan pemilihan kelas untuk masing-masing data pada data *testing* berdasarkan probabilitas yang telah dihitung.
  2. Tahap akhir melakukan evaluasi atau pengujian metode menggunakan *confusion matrix*.

### Contoh Perhitungan *Naïve Bayes*

Metode *Naive Bayes* yang digunakan pada penelitian klasifikasi tingkat kematangan SPBE dapat dilihat prosesnya pada contoh berikut. Data yang akan digunakan dapat dilihat pada Tabel 2.6.

**Tabel 2. 7 Data Penilaian Aplikasi SPBE**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No** | **Nama** | **Kebijakan** | **Tata Kelola** | **Manajemen** | **Layanan** | **Keterangan** |
| 1 | PPID Kota Kendari | 5 | 3 | 3 | 5 | Memuaskan |
| 2 | Si Asik Sehat | 3 | 2 | 1 | 3 | Cukup |
| 3 | Si Mantap | 1 | 2 | 3 | 2 | Kurang |
| 4 | SI-TPPNS | 1 | 2 | 3 | 2 | Cukup |
| 5 | Laika | 3 | 3 | 2 | 2 | Cukup |
| 6 | Kendari Preneur | 2 | 1 | 2 | 2 | Kurang |
| 7 | Jaga Kendari | 1 | 2 | 2 | 1 | Kurang |
| 8 | E-Katalog Kendari | 2 | 2 | 2 | 2 | Cukup |
| 9 | DINKES | 4 | 4 | 5 | 3 | Sangat baik |
| 10 | DISDUKCAPIL | 4 | 2 | 3 | 4 | Baik |

Berdasarkan data pada Tabel 2.6 dapat dilakukan beberapa tahapan berikut untuk mengklasifikasikan data.

1. *Preprocessing* Data

Tahap ini dilakukan dengan penyamaan skala data pada setiap atirbut. Pada contoh kasus ini, nilai kebijakan, tata kelola, manajemen, dan layanan dibagi menjadi 5, yaitu <1.80, 1.80-<2.60, 2.60-<3.50, 3.50-<4.20, dan 4.20-5.00.

**Tabel 2. 8 *Preprocessing* Data**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No** | **Nama** | **Kebijakan** | **Tata Kelola** | **Manajemen** | **Layanan** | **Keterangan** |
| 1 | PPID Kota Kendari | 4.20-5.00 | 2.60-<3.50 | 2.60-<3.50 | 4.20-5.00 | Memuaskan |
| 2 | Si Asik Sehat | 2.60-<3.50 | 1.80-<2.60 | <1.80 | 2.60-<3.50 | Cukup |
| 3 | Si Mantap | <1.80 | 1.80-<2.60 | 2.60-<3.50 | 1.80-<2.60 | Kurang |
| 4 | SI-TPPNS | <1.80 | 1.80-<2.60 | 2.60-<3.50 | 1.80-<2.60 | Cukup |

**Tabel 2. 9 *Preprocessing* Data (Lanjutan)**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 5 | Laika | 2.60-<3.50 | 2.60-<3.50 | 1.80-<2.60 | 1.80-<2.60 | Cukup |
| 6 | Kendari Preneur | 1.80-<2.60 | <1.80 | 1.80-<2.60 | 1.80-<2.60 | Kurang |
| 7 | Jaga Kendari | <1.80 | 1.80-<2.60 | 1.80-<2.60 | <1.80 | Kurang |
| 8 | E-Katalog Kendari | 1.80-<2.60 | 1.80-<2.60 | 1.80-<2.60 | 1.80-<2.60 | Cukup |
| 9 | DINKES | 3.50-<4.20 | 3.50-<4.20 | 4.20-5.00 | 2.60-<3.50 | Sangat baik |
| 10 | DISDUKCAPIL | 3.50-<4.20 | 1.80-<2.60 | 2.60-<3.50 | 3.50-<4.20 | Baik |

1. Menghitung Probabilitas Kelas

Contoh kasus ini menggunakan 5 kelas, yaitu kurang, cukup, baik, sangat baik, dan memuaskan.

1. Kelas Kurang (K)
2. Kelas Cukup (C)
3. Kelas Baik (B)
4. Kelas Sangat Baik (SB)
5. Kelas Memuaskan (M)
6. Menghitung Probabilitas Fitur

Berikut adalah menghitung probabilitas fitur atau atribut terhadap kelas yang digunakan pada contoh kasus.

1. Kebijakan (D1)

**Kelas kurang :**

(2.2)

**Kelas cukup :**

(2.3)

**Kelas baik :**

*P* (2.4)

**Kelas sangat baik :**

(2.5)

**Kelas memuaskan :**

(2.6)

1. Tata Kelola (D2)

**Kelas kurang :**

(2.7)

**Kelas cukup :**

(2.8)

**Kelas baik :**

*P* (2.9)

**Kelas sangat baik :**

(2.10)

**Kelas memuaskan :**

(2.11)

1. Manajemen (D3)

**Kelas kurang :**

(2.12)

**Kelas cukup :**

(2.13)

**Kelas baik :**

*P* (2.14)

**Kelas sangat baik :**

(2.15)

**Kelas memuaskan :**

(2.16)

1. Layanan (D4)

**Kelas kurang :**

(2.17)

**Kelas cukup :**

(2.18)

**Kelas baik :**

*P* (2.19)

**Kelas sangat baik :**

(2.20)

**Kelas memuaskan :**

(2.21)

1. Klasifikasi Tingkat Kematangan

Data *testing* yang digunakan pada kasus dapat dilihat pada Tabel 2.9 berikut.

**Tabel 2. 10 Data *Testing***

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **Nama** | **Kebijakan** | **Tata Kelola** | **Manajemen** | **Layanan** |
| 1 | PPID Kota Kendari | 5 | 3 | 3 | 5 |

Data *testing* akan melewati *preprocessing* data yang dapat dilihat pada Tabel 2.10 berikut.

**Tabel 2. 11 Data *Preprocessing***

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **Nama** | **Kebijakan** | **Tata Kelola** | **Manajemen** | **Layanan** |
| 1 | PPID Kota Kendari | 4.20-5.00 | 2.60-<3.50 | 2.60-<3.50 | 4.20-5.00 |

Berikut probabilitas aplikasi SPBE PPID Kota Kendari terhadap kelas yang telah ditentukan.

* + 1. Probabilitas Kematangan Kurang

= 0

* + 1. Probabilitas Kematangan Cukup
    2. Probabilitas Kematangan Baik
    3. Probabilitas Kematangan Sangat Baik
    4. Probabilitas Kematangan Memuaskan

= 0.000512

Berdasarkan perhitungan yang telah dilakukan, nilai probabilitas aplikasi PPID Kota Kendari terhadap setiap kelas dapat diketahui bahwa kelas Memuaskan memiliki nilai probabilitas paling tinggi. Oleh karena itu, aplikasi PPID Kota Kendari dikelompokkan ke dalam kelas Memuaskan.

## Hypertext Preprocessor (PHP)

PHP adalah bahasa *scripting web* HTML *embedded*. Artinya kode PHP dapat disisipkan ke dalam HTML halaman Web. Ketika sebuah halaman PHP diakses, kode PHP dibaca atau "diurai" oleh *server* (Tamburaka dkk., 2022). PHP atau kependekan dari *Hypertext Preprocessor* adalah salah satu bahasa pemrograman *open source* yang sangat cocok atau dikhususkan untuk pengembangan web dan dapat ditanamkan pada sebuah *script* HTML. Bahasa PHP dapat dikatakan menggambarkan beberapa bahasa pemograman seperti C, Java, dan Perl serta mudah untuk dipelajari (Ronaldo dan Pasha, 2021). PHP merupakan bahasa *server-side scripting* yang menyatu dengan HTML untuk membuat halaman web yang dinamis. Maksud dari *server-side scripting* adalah sintaks dan perintah-perintah yang diberikan akan sepenuhnya akan dijalankan dalam sistem (Destriana dkk., 2020).

## Laravel

Laravel merupakan *framework* *opensource* PHP berbasis *website* gratis yang dibuat seseorang bernama Taylor Otwell yang digunakan untuk mengembangkan aplikasi berbasis website menggunakan modul MVC atau pola arsitektur pengembangan *software*. Pengembangan sistem dari fitur Laravel adalah modul yang dapat disesuaikan memperkenalkan sebuah metode yang berbeda untuk mengakses *database* relasional, utilitas yang membantu penyaluran pelayanan aplikasi serta kemudahan untuk pemeliharaan (Bayu dkk., 2024).

## XAMPP

XAMPP adalah paket program *web* lengkap yang dapat dipakai untuk belajar pemograman web, khususnya PHP dan MySQL. Fungsinya adalah sebagai *server* yang berdiri sendiri (*localhos*t), yang terdiri atas program Apache HTTP *Server*, MySQL *database*, dan penerjemah bahasa yang ditulis dengan bahasa pemrograman PHP. Bagian penting dari XAMPP yang diasa digunakan adalah sebagai berikut (Anggraini dkk., 2020).

* 1. Htdoc adalah *folder* tempat meletakkan berkas-berkas yang akan dijalankan, seperti berkas PHP, HTML dan *script* lain.
  2. Phpmyadmin merupakan bagian untuk mengelola basis data MySQL yang terdapat dikomputer. Untuk membukanya, buka *browser* lalu mengetik alamat http://localhost/phpmyadmin, maka akan muncul halaman phpmyadmin.
  3. Kontrol Panel yang berfungsi untuk mengelola layanan (*service*) XAMPP. Seperti menghentikan (*stop*) layanan, ataupun memulai (*start*).

## MYSQL

MySQL merupakan basis data yang bersifat *open source* sehingga banyak digunakan untuk media. Walaupun gratis, MySQL tetap berkualitas dan sudah cukup memberikan *performance* yang memadai. Penggunaan PHP MyAdmin lebih mudah digunakan karena menggunakan *interface* yang lebih mudah dipahami (Ronaldo dan Pasha, 2021). MySQL adalah multiuser *database* yang menggunakan bahasa *Structured Query Language* (SQL). MySQL dalam operasi *client server* melibatkan server daemon MySQL disisi server dan berbagai macam program serta *library* yang berjalan disisi *client*. MySQL mampu menangani data yang cukup besar. Perusahaan yang mengembangkan MySQL yaitu TEX, mengaku mampu menyimpan data lebih dari 40 *database*, 10.000 tabel, dan sekitar 7.000.000 baris totalnya kurang lebih 100 *Gigabyte data* (Destriana dkk., 2020).

## *Website*

*Website* atau situs bisa diartikan menjadi formasi halaman-halaman yang dipakai untuk menampilkan info teks, gambar membisu atau gerak, animasi, bunyi, serta atau campuran dari semuanya itu baik yang bersifat tetap juga berfungsi yang menghasilkan satu rangkaian bangunan yang saling terkait dimana masing-masing dihubungkan dengan jaringan (Wulandari dan Nurmiati, 2022). Prosesnya dilakukan oleh komponen yang terdapat didalam aplikasi *browser* yang biasa disebut web engine semua dokumen web ditampilkan dengan cara diterjemahkan. Beberapa contoh web *browser* yang populer saat ini adalah Google Chrome, Mozilla Firefox, dan Opera (Ronaldo dan Pasha, 2021).

## HTML 5

HTML merupakan singkatan *Hypertext Markup Language* yaitu bahasa standar web yang dikelola penggunaanya oleh W3C (*World Wide Web Consortium*) berupa tag-tag yang menyusun setiap elemen dari *website*. HTML berperan sebagai peyusun struktur halaman website yang menempatkan setiap elemen *website layout* yang diinginkan. HTML biasanya disimpan dalam sebuah *file* berekstensi .html. Ketika mengetikkan skrip HTML dapat menggunakan *text editor* seperti Notepad sebagai bentuk paling sederhana atau *text editor* khusus yang dapat mengenali setiap unsur skrip HTML dan menampilkannya dengan warna yang berbeda sehingga mudah di baca, seperti Notepad++, Sublime Text dan masih banyak lagi aplikasi lain yang sejenisnya (Permatasari dan Suhendi, 2020).

## *Cascading Style Sheet 3* (CSS)

CSS adalah singkatan dari *Cascading Style Sheet* yaitu dokumen *web* yang berfungsi mengatur elemen HTML dengan berbagai properti yang tersedia sehingga dapat tampil dengan berbagai gaya yang diinginkan. Sebagian orang menganggap CSS bukan termasuk salah satu bahasa pemrograman karena memang strukturnya yang sederhana, hanya berupa kumpulan-kumpulan aturan yang mengatur style elemen HTML. Cara kerja CSS dalam memodifikasi HTML dengan memilih elemen HTML yang akan diatur kemudian memberikan properti yang sesuai dengan tampilan yang diinginkan. Dalam memberikan aturan pada elemen HTML, skrip CSS terdiri atas 3 bagian yaitu selector untuk memilih elemen yang akan diberi aturan, properti yang merupakan aturan yang diberikan dan *value* sebagai nilai dari aturan yang diberikan (Permatasari dan Suhendi, 2020).

## Visual Studio Code

Visual Studio Code adalah kode editor sumber yang dikembangkan oleh Microsoft untuk Windows, Linux, dan macOS. Ini termasuk dukungan untuk *debugging*, kontrol git yang tertanam dan GitHub, penyorotan sintaksis, penyelesaian kode cerdas, snippet, dan *refactoring* kode. Ini sangat dapat disesuaikan, memungkinkan pengguna untuk mengubah tema, pintasan keyboard, preferensi, dan menginstal ekstensi yang menambah fungsionalitas tambahan (Agustini dan Kurniawan, 2019). Visual Code memudahkan dalam penulisan code yang mendukung beberapa jenis pemrograman, seperti C++, C#, Java, Python, PHP, GO. Visual Code memiliki kemampuan untuk mengidentifikasi jenis bahasa pemrograman yang digunakan dan memberi variasi warna sesuai dengan fungsi dalam rangkaian kode tersebut. Visual Studio Code juga telah terintegrasi ke Github. Selain itu fitur lainnya adalah kemampuan untuk menambah ekstensi dimana para pengembang dapat menambah ekstensi untuk menambah fitur yang tidak ada di Visual Studio Code (Ramdhan dan Nufriana, 2019).

## *Unified Modeling Language* (UML)

UML (*Unified Modeling Language*) merupakan pengganti dari metode analisis berorientasi object dan design berorientasi object (OOAD&D/*Object Oriented Analysis and Design*) yang dimunculkan sekitar akhir tahun 80-an dan awal tahun 90-an. UML merupakan gabungan dari metode Booch, Rumbaugh (OMT) dan Jacobson. Tetapi, UML mencakup lebih luas daripada OOAD. Pada pertengahan saat pengembangan UML, dilakukan standarisasi proses dengan OMG (Object Management Group) dengan harapan UML bakal menjadi bahasa standar pemodelan pada masa yang akan datang (Rahmatuloh dan Revanda, 2022).

### Diagram Dalam UML

1. *Use Case Diagram*

Diagram *use case* menggambarkan alur kerja sistem dengan cara yang sangat sederhana, fungsi utama dari sistem dan berbagai jenis pengguna yang akan berinteraksi dengan sistem (Destriana et al. 2020). *Use Case Diagram* merupakan diagram yang harus dibuat pertama kali saat pemodelan pemrograman perangkat lunak berorientasi objek dilakukan. *Use Case Diagram* akan menggambarka apa yang dikerjakan oleh actor (Jevrian Efendi, 2021).

**Tabel 2. 12 Simbol *Use Case Diagram***

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Simbol** | **Nama** | **Keterangan** |
| 1 |  | *Actor* | Memiliki peran orang, sistem yang lain, atau alat ketika berkomunikasi dengan *use case.* |
| 2 |  | *Dependency* | Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (*independent*) akan mempengaruhi elemen yang bergantung padanya (*independent*). |

**Tabel 2. 13 Simbol *Use Case Diagram* (Lanjutan)**

|  |  |  |  |
| --- | --- | --- | --- |
| 3 |  | *Generalization* | Menunjukkan spesialisasi *actor* untuk dapat berpartisipasi dengan *use case*. |
| 4 |  | *Include* | Menunjukkan bahwa suatu *use cse* seluruhnya merupakan fungsionalitas dari *use case* lainnya*.* |
| 5 |  | *Extend* | Menunjukkan bahwa suatu *use case* merupakan tambahan fungsional dari *use case* lainnya jika suatu kondisi terpenuhi. |

Sumber : (Saputra dan Fahrizal, 2021)

1. *Activity* *Diagram*

*Activity diagram* menggambarkan berbagai aliran aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing aliran berawal, *decision* yang mungkin terjadi dan bagaimana mereka berakhir. *Activity diagram* juga dapat menggambarkan proses pararel yang mungkin terjadi pada beberapa eksekusi (Kurniawan dan Syarifuddin, 2020).

**Tabel 2. 14 Simbol *Activity Diagram***

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Simbol** | **Nama** | **Keterangan** |
| 1 |  | *Activity* | Memperlihatkan bagaimana masing-masing kelas antarmuka saling berintraksi satu sama lain. |
| 2 |  | *Action* | State dari sistem yang mencerminkan eksekusi dari suatu aksi. |
| 3 |  | *Initial Node* | Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status alwal. |

**Tabel 2. 15** **Simbol *Activity Diagram* (Lanjutan)**

|  |  |  |  |
| --- | --- | --- | --- |
| 4 |  | *Activity Final Node* | Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir. |
| 5 |  | *Fork Node* | Satu aliran yang pada tahap tertentu berbuah menjadi beberapa aliran. |

Sumber : (Saputra dan Fahrizal, 2021)

1. *Class Diagram*

*Class diagram* merupakan diagram yang menunjukkan *class-class* yang ada di sistem dan hubungannya secara logik (Jevrian Efendi 2021). *Class diagram* menjadi model yang menggambarkan struktur dan deskripsi *class* serta dapat menghubungkan antara *class* yang lain. *Class diagram* menjelaskan model yang digunakan dalam perancangan atribut dan fungsi-fungsi yang akan digunakan untuk membangun sistem baru (Anggraini dkk., 2020).

**Tabel 2. 16 Simbol *Class Diagram***

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Simbol** | **Nama** | **Keterangan** |
| 1 |  | *Generalization* | Hubungan dimana objek anak (*descendent*) beerbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (*ancestor*). |
| 2 |  | *Nary Association* | Upaya untuk menghindari asosiasi dengan atribut serta operasi yang sama. |
| 3 |  | *Class* | Himpunan dari objek-objek ya g berbagi atribut srta operasi ang sama. |

**Tabel 2. 17 Simbol *Class Diagram* (Lanjutan)**

|  |  |  |  |
| --- | --- | --- | --- |
| 4 |  | *Collaboration* | Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu *actor*. |
| 5 |  | *Realization* | Operasi yang benar-benar dilakukan suatu objek. |
| 6 |  | *dependency* | Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri. |
| 7 |  | *Association* | Apa yang meghubungkan antara objek satu dengan objek lainnya. |
| 8 | 0..1 | Nilai Kardinalitas | Kosong atau satu. |
| 9 | 0..\* | Nilai Kardinalitas | Lebih dari sama dengan kosong. |
| 10 | 0..n | Nilai Kardinalitas | Lebih dari sama dengan n, dimana n lebih dari satu. |
| 11 | 1 | Nilai Kardinalitas | Hanya satu. |
| 12 | 1..\* | Nilai Kardinalitas | Lebih dari sama dengan satu. |
| 13 | 1..n | Nilai Kardinalitas | Lebih dari sama dengan satu dimana n lebih dari satu. |
| 14 | \* | Nilai Kardinalitas | Banyak atau *Many.* |
| 15 | n | Nilai Kardinalitas | Hanya N, dimana N lebih dari satu. |
| 16 | n..\* | Nilai Kardinalitas | Lebih dari sama dengan N dimana N lebih dari satu. |

Sumber : (Jevrian Efendi, 2021)

1. *Sequence Diagram*

*Sequence diagram* adalah interaksi dari objek yang disusun dalam suatu urutan waktu atau kejadian tertentu dalam suatu proses, dapat digambarkan dengan *sequence diagram* (Jevrian Efendi 2021). Sequence diagram terdiri antara dimensi vertikal (waktu) dan dimensi horizontal (objek-objek yang terkait) (Kurniawan dan Syarifuddin, 2020).

**Tabel 2. 18 Simbol *Sequence Diagram***

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Simbol** | **Nama** | **Keterangan** |
| 1. |  | Aktor | Orang proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi yang akan dibuat itu sendiri. |
| 2. |  | Garis hidup/ *lifeline* | Menyatakan kehidupan suatu objek |
| 3. | Nama objek : nama kelas | Objek | Menyatakan objek yang berinteraksi pesan |
| 4. |  | Waktu aktif | Menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu aktif ini adalah sebuah tahapan. |
| 5. |  | Pesan tipe *create* | Menyatakan suatu objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat |

**Tabel 2. 19 Simbol *Sequence Diagram* (Lanjutan)**

|  |  |  |  |
| --- | --- | --- | --- |
| 6. |  | Pesan tipe *call* | Menyatakan suatu objek memanggil opeasi/metode yang ada pada objek lain atau dirinya sendiri |
| 7. |  | Pesan tipe *send* | Menyatakan bahwa suatu objek mengirim data / masukkan / informasi ke objek lainnya, arah panah mengarah pada objek yang dikirimi. |
| 8. |  | Pesan tipe *return* | Menyatakan bahwa suatu objek yang telah menjalankan suatu operasi atau metode menghasilkan suatu kembalian ke objek tertentu, |

Sumber : (Saputra dan Fahrizal, 2021)

## RUP (*Rational Unified Process*)

Metode RUP merupakan metode pembangunan perangkat lunak yang *iterative* dan *incremental* serta berfokus pada arsitektur. Metode RUP dapat menangani resiko yang berhubungan dengan pengembangan kebutuhan sistem berdasarkan perubahan yang diinginkan oleh klien. Untuk mengurangi resiko tersebut dilakukan dengan pengujian pada setiap akhir tahapan RUP, sehingga akan mudah melakukan perubahan sebelum mencapai tahap akhir. Metode RUP juga mengutamakan kepuasan pengguna sehingga lebih sering melakukan interaksi dengan pengguna (Perwitasari dkk., 2020).

Dalam metode *Rational Unified Process* terdapat empat tahapan sebagai berikut (Perwitasari dkk., 2020).

1. ***Inception* (Permulaan)**

Tahap ini lebih pada memodelkan proses bisnis yang dibutuhkan (*business modeling*), mendefinisikan kebutuhan akan sistem yang akan dibuat (*requirement*), serta analisis dan desain yang meliputi analisis sistem *existing*, perumusan sistem target, penentuan arsitektur global target, identifikasi kebutuhan, perumusan persyaratan (fungsional, performansi, keamanan, GUI, dll), perumusan kebutuhan pengujian (level unit, integrasi, sistem, performansi, fungsionalitas, keamanan, dll), UML diagram, dan pembuatan dokumentasi.

1. ***Elaboration* (Perencanaan)**

Tahap ini lebih pada analisis dan desain sistem serta implementasi sistem yang fokus pada purwarupa sistem (*prototype*). Tahap ini juga dapat mendeteksi apakah arsitektur sistem yang dinginkan dapat dibuat atau tidak. Melakukan analisis dan desain sistem serta implementasi sistem yang berfokus pada sistem *prototype*.

1. ***Construction* (Konstruksi)**

Tahap ini lebih pada implementasi dan pengujian sistem yang fokus pada implementasi perangkat lunak pada kode program. Pada tahap awal *construction*, ada baiknya dilakukan pemeriksaan ulang hasil analisis dan desain, terutama desain pada *sequence diagram*, *class diagram*, *component*, dan *deployment*. Apabila desain yang dibuat telah sesuai dengan analisis sistem, maka implementasi dengan bahasa pemrogramanan tertentu dapat dilakukan.

1. ***Transition* (Transisi)**

Tahap ini lebih pada *deployment* atau instalasi sistem agar dapat dimengerti oleh *user*. Aktivitas pada tahap ini termasuk pelatihan pengguna, pemeliharaan, dan pengujian sistem apakah sudah memenuhi harapan pengguna.

## Pengujian *Black Box*

*Black box testing* merupakan pengujian kualitas perangkat lunak yang berfokus pada fungsionalitas perangkat lunak. Pengujian *black box* bertujuan untuk menemukan fungsi yang tidak benar, kesalahan antarmuka, kesalahan pada struktur data, kesalahan performansi, kesalahan inisialisasi, dan terminasi. Pengujian *Black Box* bertumpu pada memastikan tiap proses sudah berfungsi sesuai dengan kebutuhan yang diharapkan. Penguji dapat menartikan himpunan kondisi masukan dan menjalankan pengujian pada pengkhususan fungsi dari sistem. Sehingga pengujian merupakan suatu cara pelaksanaan program yang bertujuan menemukan kesalahan atau *error* kemudian memperbaikinya sehingga sistem dapat dikatakan layak untuk digunakan (Yahya Dwi dan Muna Wardah, 2021).

## *Confusion Matrix*

*Confusion matrix* adalah alat ukur berbentuk matrix yang digunakan untuk mendapatkan jumlah ketepatan klasifikasi terhadap kelas dengan algoritma yang dipakai. *Confussion matrix* memiliki empat istilah yang digunakan untuk menunjukkan hasil klasifikasi, yaitu *True Positive* (TP), *True Negative* (TN), *False Positive* (FP), dan *False Negative* (FN). Istilah-istilah tersebut akan digunakan dalam menghitung nilai *accuracy, precision*, *recall,* dan *F-measure* dalam menentukan tingkat keakuratan metode yang digunakan. Semakin tinggi nilai TP dan TN maka semakin tinggi tingkat klasifikasi berdasarkan hasil dari *accuracy, precision*, *recall,* dan *F-measure* (Qadrini dkk., 2021).

**Tabel 2. 20 *Confusion Matrix***

|  |  |  |  |
| --- | --- | --- | --- |
|  | | **Aktual** | |
| **Benar** | **Salah** |
| **Prediksi** | **Benar** | TP | FN |
| **Salah** | FP | TN |

**Keterangan:**

1. *True Positive* (TP): Jumlah data yang bernilai positif dan diprediksi benar sebagai positif.
2. *False Positive* (FP): Jumlah data yang bernilai negatif tetapi diprediksi sebagai positif.
3. *False Negative* (FN): Jumlah data yang bernilai positif tetapi diprediksi sebagai negatif.
4. *True Negative* (TN): Jumlah data yang bernilai negatif dan diprediksi benar sebagai negatif.

Berikut persamaan yang digunakan dalam menghitung *accuracy, precision*, *recall,* dan *F-measure* pada *confussion matrix* (Rahayu dk., 2021).

1. *Accuracy* digunakan untuk menghitung sebarapa baik metode yang digunakan untuk membuat prediksi yang *true* dari total semua prediksi yang dilakukan.

(2.22)

1. *Precision* digunakan untuk menghitung seberapa baik metode yang digunakan untuk membuat prediksi yang *true positive* dari total semua prediksi positif yang dilakukan oleh metode.

(2.23)

1. *Recall* digunakan untuk menghitung seberapa baik metode yang digunakan untuk membuat prediksi *true positive* dengan benar.

(2.24)

1. *F-measure* digunakan untuk mengukur akurasi model dengan menggunakan nilai *precision* dan *recall*.

(2.25)

# BAB III

# METODOLOGI PENELITIAN

## Metode Pengumpulan Data

Pengumpulan data dalam menyusun tugas akhir ini dilakukan menggunakan beberapa metode. Adapun metode yang digunakan adalah sebagai berikut.

1. Studi Literatur

Studi literatur adalah serangkaian kegiatan yang digunakan untuk mencari dan mengolah informasi yang menjadi referensi dalam proses penelitian. Metode ini sangat penting karena membantu peneliti dalam memahami toeri dan menemukan informasi penting mengenai topik penelitian yang akan dilakukan, yaitu mengenai Sistem Pemerintahan Berbasis Elektronik (SPBE) dan metod *Naïve Bayes*. Sumber informasi bisa diperoleh dari jurnal, buku, *paper*, skripsi, dan sebagainya.

1. Wawancara

Wawancara adalah proses komunikasi langsung yang dilakukan antara peneliti dan narasumber, yaitu karyawan Dinas Komunikasi dan Informasi Kota Kendari di bidang *E-Government* yang menangani perihal Sistem Pemerintahan Berbasis Elektronik (SPBE) di Kota Kendari. Tujuan dari wawancara ini adalah untuk mengetahui hal-hal apa saja yang menjadi faktor dalam penentuan tingkat kematangan SPBE di Kota Kendari.

## Metode Pengembangan Sistem

Metode pengemangan sistem yang digunakan dalam penelitian tugas akhir ini adalah metode *Rational Unified Process* (RUP). Metode ini merupakan metode pengembangan sistem yang iteratif, terstruktur, lebih efisien, dan menghasilkan sistem yang dapat disesuaikan dengan kebutuhan yang diinginkan. Metode RUP terdiri dari beberapa tahap berikut.

1. *Inception* (Permulaan)

Pada tahap *inception*, dilakukan pemodelan proses bisnis, tujuan, dan kebutuhan sistem yang akan dibuat, seperti biaya, waktu, kebutuhan, dan resiko. Tahap ini bisa didukung dengan melakukan pegkajian mengenai sistem pemerintahan berbasis elektronik dan implementasi metode *Naïve Bayes* pada sistem yang akan dibangun.

1. *Elaboration* (Perluasan/Perencanaan)

Tahap *elaboration* adalah tahap dimana merancang arsitektur sistem dan merencanakan pendekatan pembangunan sistem secara lebih detail. Setelah melakukan analisis pada tahap *inception*, hasil analisis tersebut akan digunakan untuk mendesain arsitektur sistem pada tahap ini. Aktivitas yang dilakukan berupa desain *user* *interface*, desain *database*, dan pemodelan diagram UML (*Unified Modelling Language*) yang akan digunakan untuk mengembangkan sistem.

1. *Construction* (Konstruksi)

Tahap *construction* adalah tahap untuk mengimplementasikan rancangan dan rencana yang telah dibuat pada tahap sebelumnya. Namun, sebelum itu, dilakukan pemeriksaan ulang kesesuaian hasil analisis dan desain terhadap permintaan. Sistem mulai dibangun dan dikembangkan menggunakan dengan memperhatikan fitur-fitur sistem. Pembangunan sistem melibatkan metode *Naïve Bayes* dalam proses klasifikasi data.

1. *Transition* (Transisi)

Tahap *transition* adalah tahap dimana akan dilakukan uji coba (*testing*) sistem yang telah dibangun dan dikembangkan apakah sudah sesuai dengan permintaan dalam mengklasifikasikan tingkat kematangan sistem pemerintahan berbasis elektronik. Pengujian sistem akan dilakukan menggunakan pengujian *black box*.

## Waktu dan Tempat Penelitian

### Waktu Penelitian

Pelaksanaan penelitian tugas akhir akan dilakukan mulai bulan April sampai dengan Juli 2024. Rincian kegiatan dan waktu penelitian dapat dilihat pada Tabel 3.1.

**Tabel 3. 1 Waktu Penelitian**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| No | Uraian | Waktu (2024) | | | | | | | | | | | | | | | |
| April | | | | Mei | | | | Juni | | | | Juli | | | |
| 1 | 2 | 3 | 4 | 1 | 2 | 3 | 4 | 1 | 2 | 3 | 4 | 1 | 2 | 3 | 4 |
| 1. | *Inception* |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 2. | *Elaboration* |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 3. | *Construction* |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 4. | *Transition* |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

### Tempat Penelitian

Penelitian tugas akhir ini akan dilakukan di Laboratorium *Software Engineering*, Jurusan Teknik Informatika, Fakultas Teknik, Universitas Halu Oleo dan Dinas Komunikasi dan Informasi Kota Kendari.

## Analisis Kebutuhan Sistem

Analisis kebutuhan sistem adalah analisis yang dilakukan untuk menentukan komponen-komponen kebutuhan yang akan diimplementasikan dalam proses pembangunan sistem. Tujuan utama dari analisis sistem adalah memahami masalah atau kebutuhan bisnis yang ada kemudian merumuskan solusi yang sesuai untuk memenuhi kebutuhan tersebut. Tahap ini meliputi kebutuhan fungsional berupa *input* dan *output* yang akan dihasilkan oleh sistem, serta kebutuhan proses yang akan mengolah *input* menjadi *output* yang diinginkan. Tidak hanya itu, kebutuhan non fungsional juga termasuk dalam analisis kebutuhan sistem berupa analisis perangkat keras dan perangkat lunak yang akan digunakan dalam proses pengembangan sistem.

### Analisis Kebutuhan Fungsional

Analisis kebutuhan fungsional adalah *input* dan *output* yang akan dihasilkan oleh sistem, serta kebutuhan proses yang akan mengolah *input* menjadi *output* yang diinginkan.

1. Analisis kebutuhan *input* dari sistem ini berupa informasi data sistem pemerintahan berbasis elektronik dan hasil penilaian dari sistem tersebut yang telah dilakukan oleh tim penilai dari Dinas Komunikasi dan Informasi Kota Kendari.
2. Analisis kebutuhan proses pada sistem ini adalah proses pembuatan sistem untuk mengolah data *input* menggunakan bahasa pemrograman PHP Laravel.
3. Analisis kebutuhan *output* dari sistem ini adalah *user* dapat melihat hasil klasifikasi tingkat kematangan SPBE.

### Analisis Kebutuhan Non Fungsional

Analisis kebutuhan non fungsional adalah analisis yang dilakukan untuk mengetahui spesifikasi sumber daya yang dibutuhkan dalam membangun aplikasi. Analisis ini dibagi dalam dua tahap, yaitu analisis kebutuhan perangkat keras (*hardware*) dan analisis kebutuhan perangkat lunak (*software*).

1. Kebutuhan Perangkat Keras (*Hardware*)

Analisis kebutuhan perangkta keras dilakukan untuk menentukan spesifikasi perangkat keras yang akan digunakan dalam membangun sistem. Perangkat keras yang digunakan dapat dilihat pada Tabel 3.2 berikut.

**Tabel 3. 2 Perangkat Keras**

|  |  |  |
| --- | --- | --- |
| **No** | **Nama Perangkat** | **Spesifikasi** |
| 1 | Laptop | *Lenovo IdeaPad* S340-14IML |
| 2 | *Processor* | *Intel(R) Core(TM)* i7-10510U |
| 3 | Monitor | 14 *Inch* |
| 4 | Memori | 8 *GB* |
| 5 | *Harddisk* | 512 *GB SSD* |

1. Kebutuhan Perangkat Lunak (*Software*)

Perangkat lunak yang dibutuhkan dalam membangun sistem dapat dilihat pada Tabel 3.3.

**Tabel 3. 3 Perangkat Lunak**

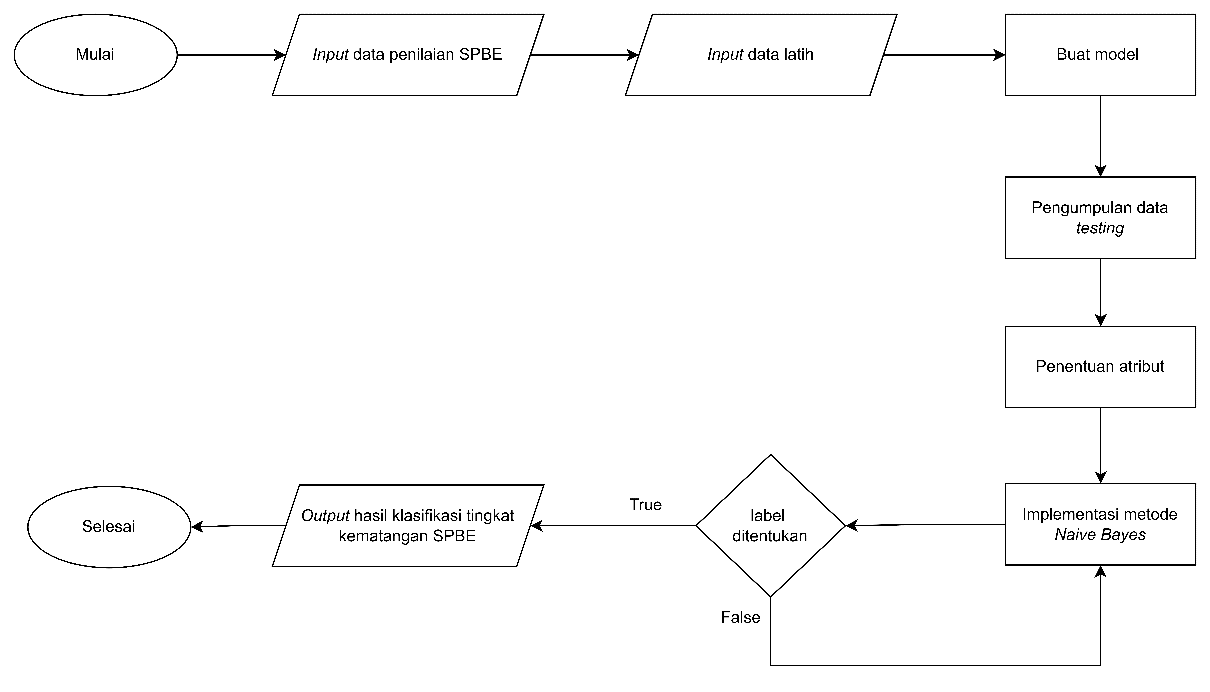
|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Nama** | **Fungsi** | **Spesifikasi** |
| 1 | *Windows* | Sistem operasi | *Windows 11 Home* |
| 2 | PHP | Bahasa Pemrograman | Versi 8.1.10 |
| 3 | *Laravel* | *Framework* PHP | Versi 10.42.0 |
| 4 | *Xampp* | *Integrated Development Environment* | Versi 2022.2 |
| 5 | *Visual Studio Code* | *Integrated Development Environment* | Versi 1.86.0 |
| 6 | *Mysql* | *Database Management System* | Versi 6.0.0 |
| 7 | *Apache* | *Web Server* | *Apache* 2.4.39 |
| 9 | *Chrome* | *Web Browser* | Versi 121.0.6167 |

## Analisis Peracangan Sistem

Analisis perancangan sistem merupakan analisis yang meliputi perancangan *flowchart*, perancangan UML (*Unified Modeling Language*), dan perancangan *user interface* sistem.

### Perancangan *Flowchart*

Perancangan *flowchart* adalah perancangan alur kerja sistem ketika menerapkan metode *Naïve Bayes* dalam kasus klasifikasi tingkat kematangan SPBE berbasis *web*. Perancangan dilakukan setelah melakukan analisis kebutuhan sistem. Alur kerja sistem dapat dilihat pada Gambar 3.1 berikut.



**Gambar 3. 1 Diagram *Flowchart***

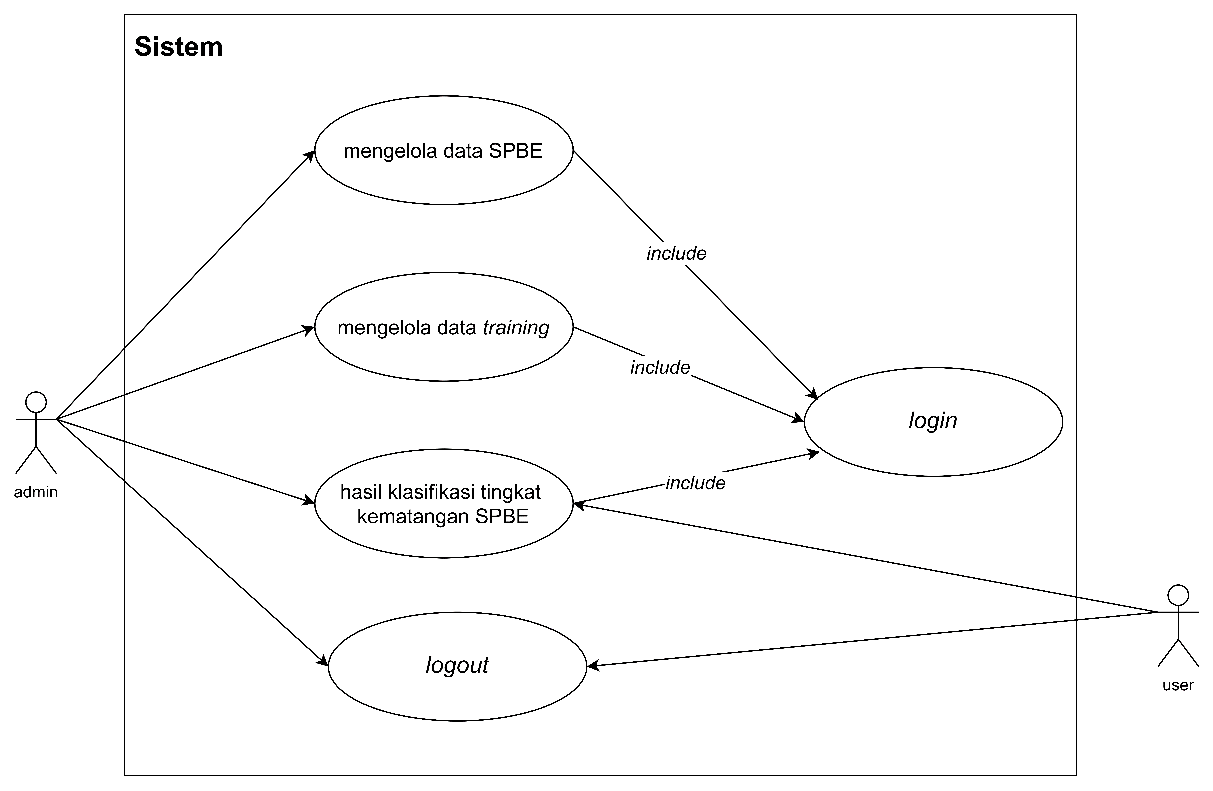
Alur kerja diagram *flowchart* sistem yang akan dibangun berdasarkan Gambar 3.1 adalah sebagai berikut.

1. Tahap pertama, *user* menginputkan data penilaian SPBE.
2. Tahap kedua, *user* menginputkan data latih yang akan digunakan untuk melatih model.
3. Tahap ketiga, melakukan pengumpulan data *testing* yang akan digunakan untuk menguji model yang akan digunakan.
4. Kemudian, melakukan penentuan atribut yang akan digunakan pada proses perhitungan *Naive Bayes*.
5. Selanjutnya, implementasi perhitungan *Naive Bayes* dalam mencari nilai probabilitas tingkat kematangan SPBE.
6. Tahap akhir, menampilkan hasil klasifikasi tingkat kematangan SPBE dengan label kurang, cukup, baik, sangat baik, dan memuaskan.

### Perancangan UML (*Unified Modeling Language*)

1. *Use Case Diagram*

*Use case diagram* adalah gambaran interaksi antara *user* dan sistem. *Use case diagram* pada sistem klasifikasi tingkat kematangan SPBE yang akan dibangun dapat dilihat pada Gambar 3.2 berikut.



**Gambar 3. 2 *Use Case* *Diagram***

Deskrispi interaksi yang dilakukan oleh *user* terhadap sistem pada Gambar 3.2 adalah sebagai berikut.

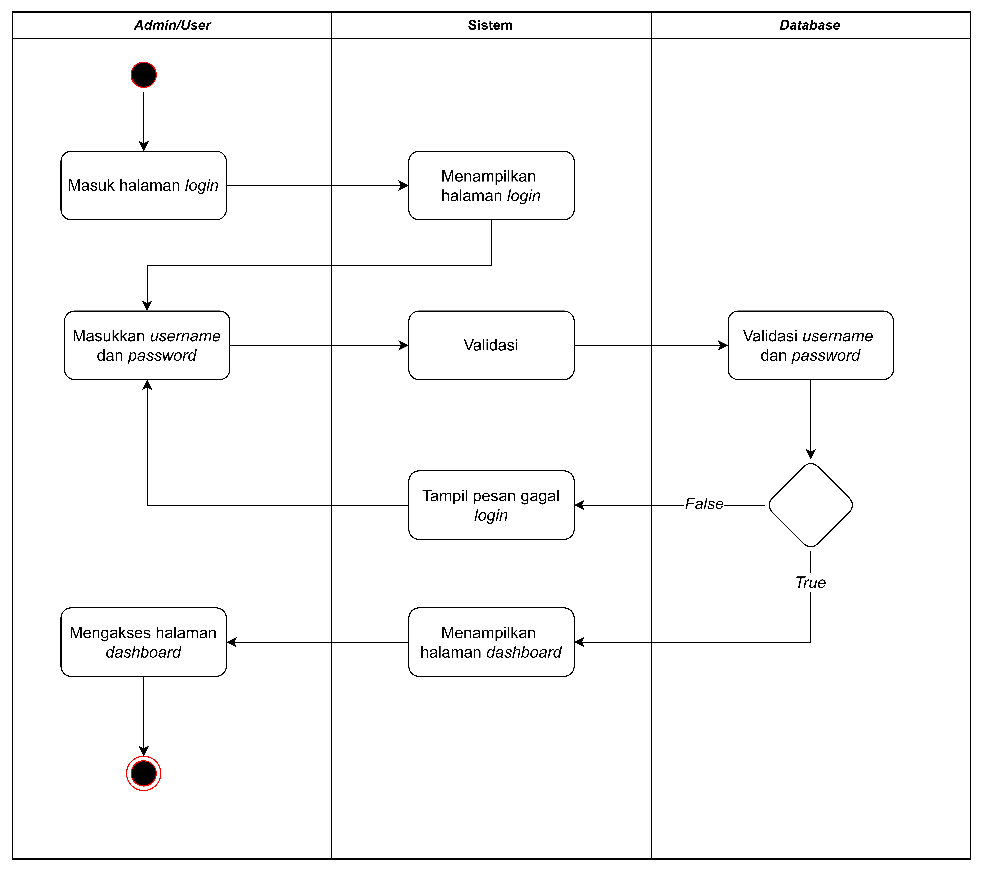
* + 1. Admin dan *user* terlebih dahulu melakukan *login* dengan memasukkan *username* dan *password* yang sesuai.
    2. *User* hanya dapat melihat hasil klasifikasi tingkat kematangan SPBE.
    3. Admin dapat melakukan *input* data SPBE, mengelola data SPBE, mengelola data *training*, dan melihat hasil klasifikasi tingkat kematangan SPBE.
    4. Admin dan *user* dapat melakukan *logout* untuk keluar dari sistem.

1. *Activity Diagram*

*Activity diagram* adalah diagram alir yang berisi aktivitas-aktivitas yang dilakukan oleh *user* pada sistem.

1. *Activity diagram login*

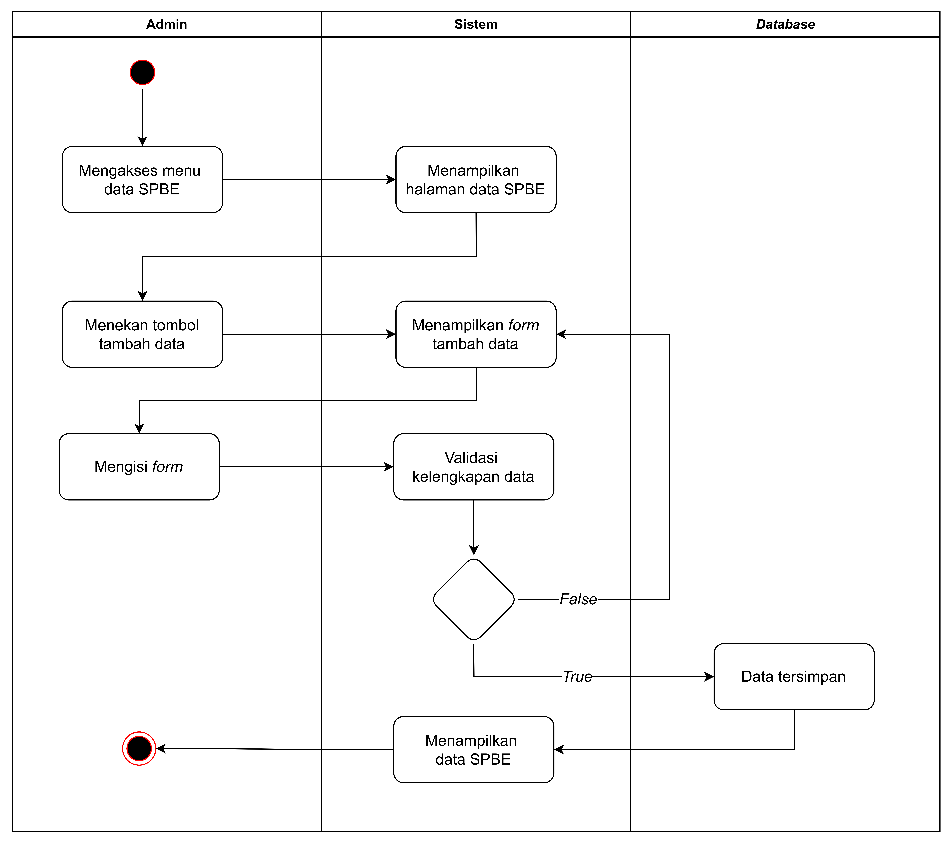
Halaman *dashboard* dapat diakses ketika admin atau *user* melakukan *login* terlebih dahulu dengan memasukkan *username* dan *password*. Kemudian, sistem akan melakukan proses validasi *username* dan *password* yang tersimpan di *database*. Ketika *invalid*, maka akan ditampilkan pesan gagal *login*. Namun, jika *username* dan *password* benar maka halaman *dashboard* akan ditampilkan. *Diagram activity login* dapat dilihat pada Gambar 3.3 berikut.



**Gambar 3. 3 *Activity Diagram Login***

1. *Activity diagram* tambah data SPBE

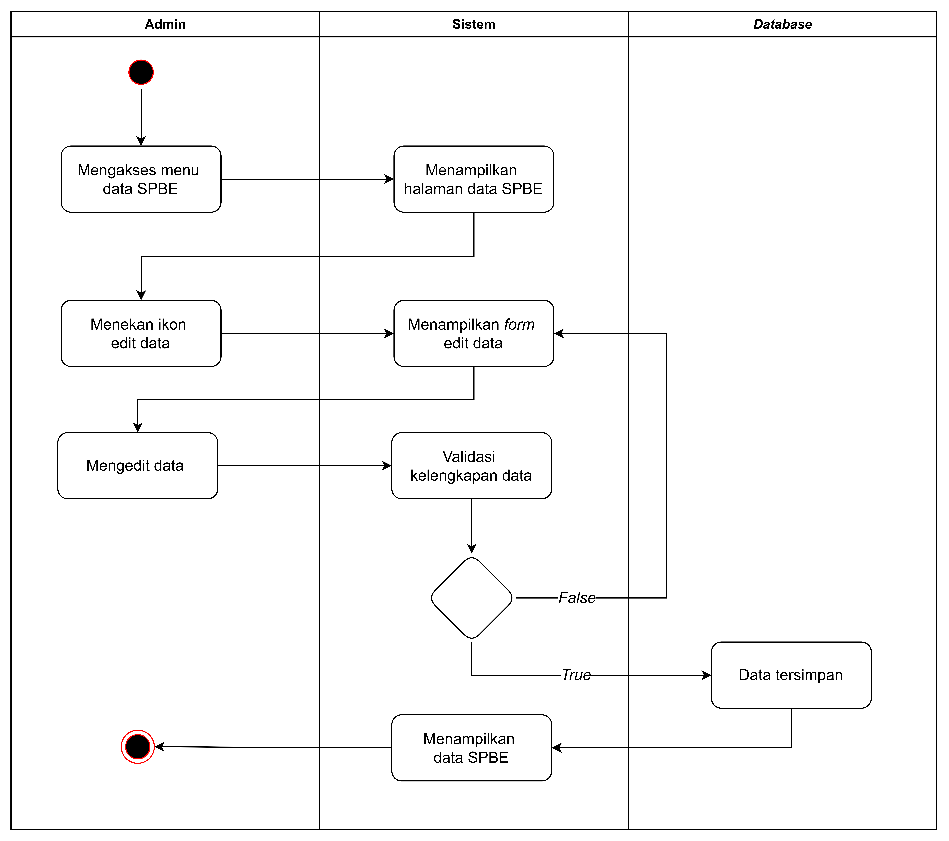
Penambahan data SPBE dapat dilakukan dengan mengakses menu data SPBE kemudian menekan tombol tambah data. Admin memasukkan informasi data secara lengkap lalu tekan tombol simpan. Setalah itu, data disimpan di *database* dan ditampilkan di halaman data SPBE. *Activity* tambah data SPBE dapat dilihat pada Gambar 3.4 berikut.



**Gambar 3. 4 *Activity* *Diagram* Tambah Data SPBE**

1. *Activity diagram* edit data SPBE

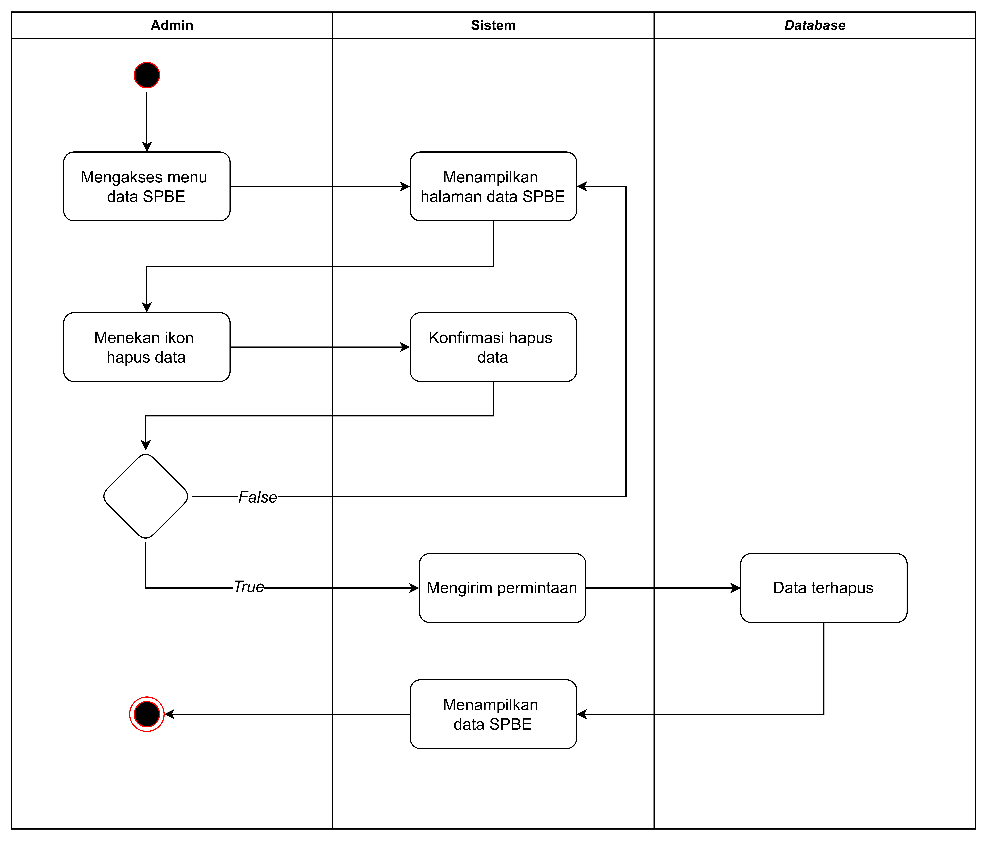
Pengeditan atau *update* data SPBE dapat dilakukan dengan mengakses menu data SPBE kemudian menekan ikon edit data. Admin mengedit informasi data secara lengkap lalu tekan tombol simpan. Setalah itu, data ter-*update* di *database* dan ditampilkan di halaman data SPBE. *Activity* edit data SPBE dapat dilihat pada Gambar 3.5 berikut.



**Gambar 3. 5 *Activity* *Diagram* Edit Data SPBE**

1. *Activity diagram* hapus data SPBE

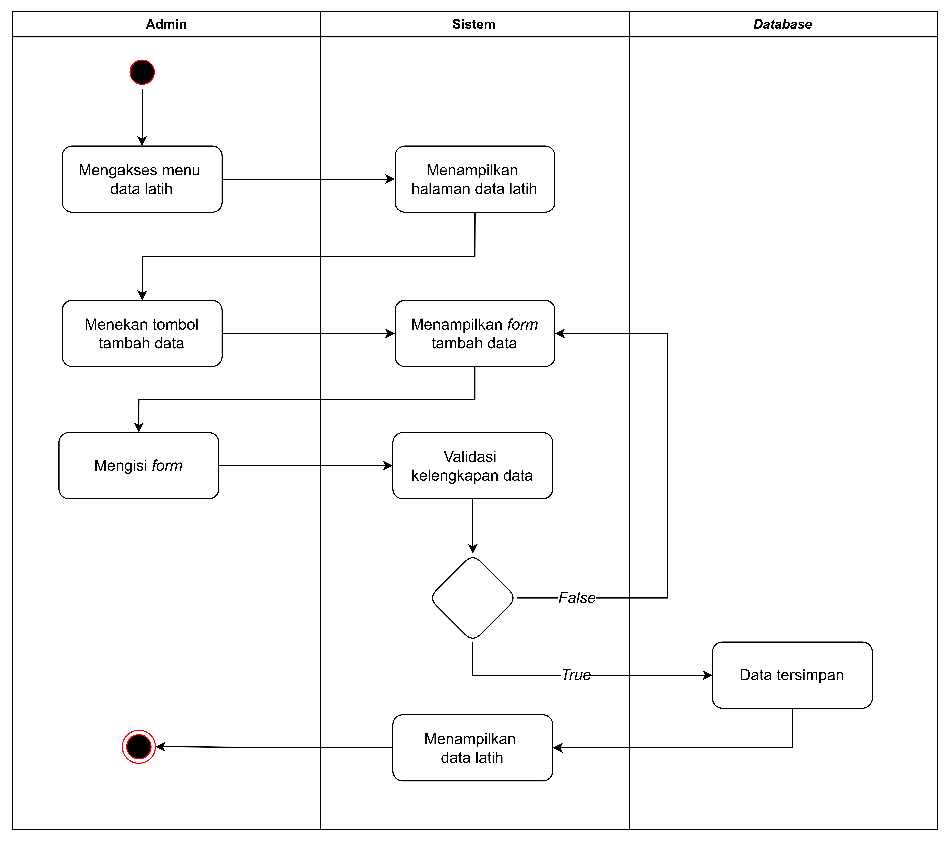
Penghapusan data SPBE dapat dilakukan oleh admin dengan mengakses menu data SPBE kemudian menekan ikon *delete*. Setalah itu, data akan dihapus di *database* dan di halaman data SPBE. *Activity* hapus data SPBE dapat dilihat pada Gambar 3.6 berikut.



**Gambar 3. 6 *Activity* *Diagram* Hapus Data SPBE**

1. *Activity diagram* tambah data *training*

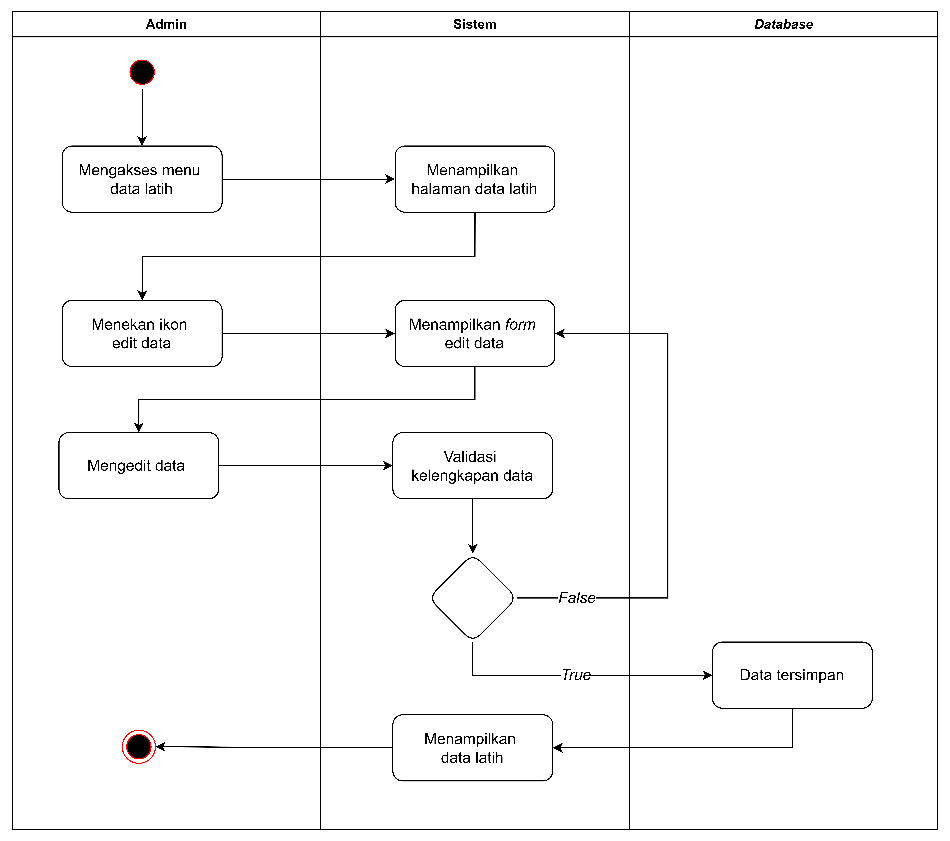
Penambahan data *training* dapat dilakukan dengan mengakses menu data *training* kemudian menekan tombol tambah data. Admin memasukkan informasi data secara lengkap lalu menekan tombol simpan. Setalah itu, data disimpan di *database* dan ditampilkan di halaman data *training*. *Activity* tambah data *training* dapat dilihat pada Gambar 3.7 berikut.



**Gambar 3. 7 *Activity* *Diagram* Tambah Data *Training***

1. *Activity diagram* edit data *training*

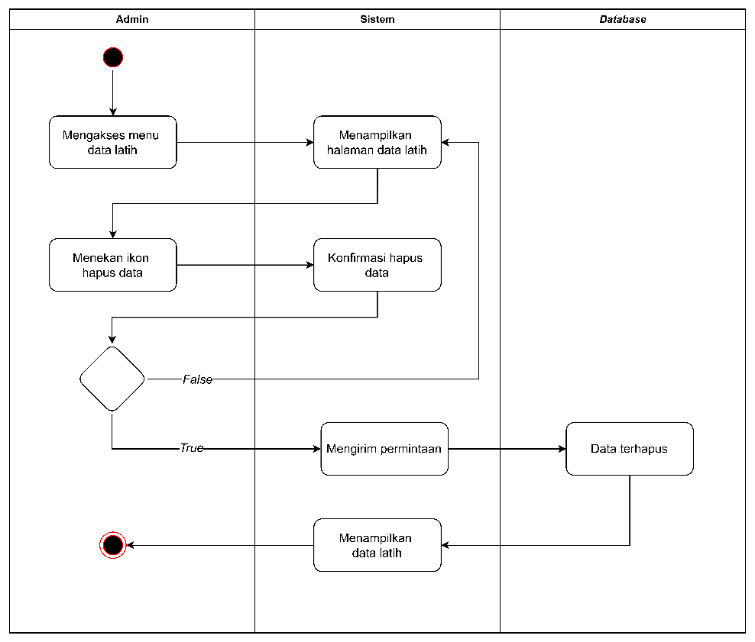
Pengeditan atau *update* data *training* dapat dilakukan dengan mengakses menu data *training* kemudian menekan ikon edit data. Admin mengedit informasi data secara lengkap lalu menekan tombol simpan. Setalah itu, data ter-*update* di *database* dan ditampilkan di halaman data *training*. *Activity* edit data *training* dapat dilihat pada Gambar 3.8 berikut.



**Gambar 3. 8 *Activity* *Diagram* Edit Data *Training***

1. *Activity diagram* hapus data *training*

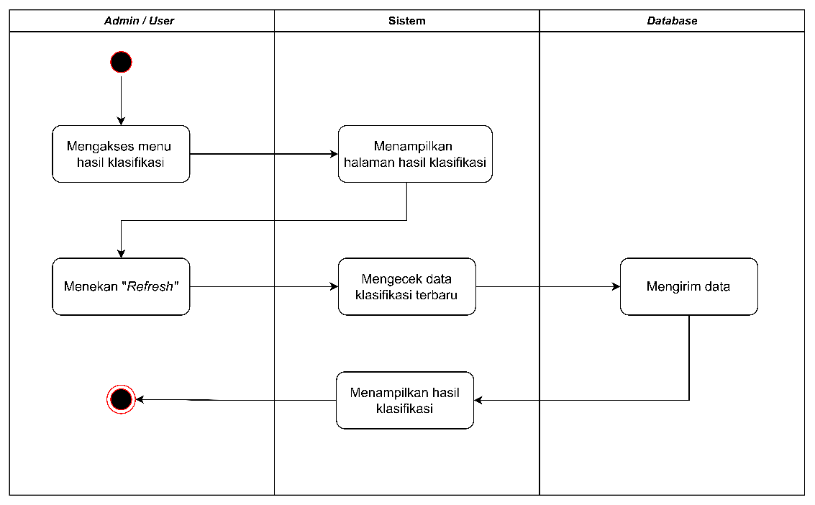
Penghapusan data *training* dapat dilakukan oleh admin dengan mengakses menu data *training* kemudian menekan ikon *delete*. Setalah itu, data akan dihapus di *database* dan di halaman data *training*. *Activity* hapus data *training* dapat dilihat pada Gambar 3.6 berikut.



**Gambar 3. 9 *Activity* *Diagram* Hapus Data *Training***

1. *Activity diagram* hasil klasifikasi

Hasil klasifikasi tingkat kematangan SPBE dapat dilihat oleh admin atau *user* dengan mengakses halaman hasil klasifikasi. Kemudian, untuk mengetahui hasil klasifikasi terbaru apabila ada penambahan data SPBE, maka admin atau *user* dapat menekan tombol *refresh* untuk memuat ulang hasil klasifikasi. *Activity* hasil klasifikasi dapat dilihat pada Gambar 3.10 berikut.



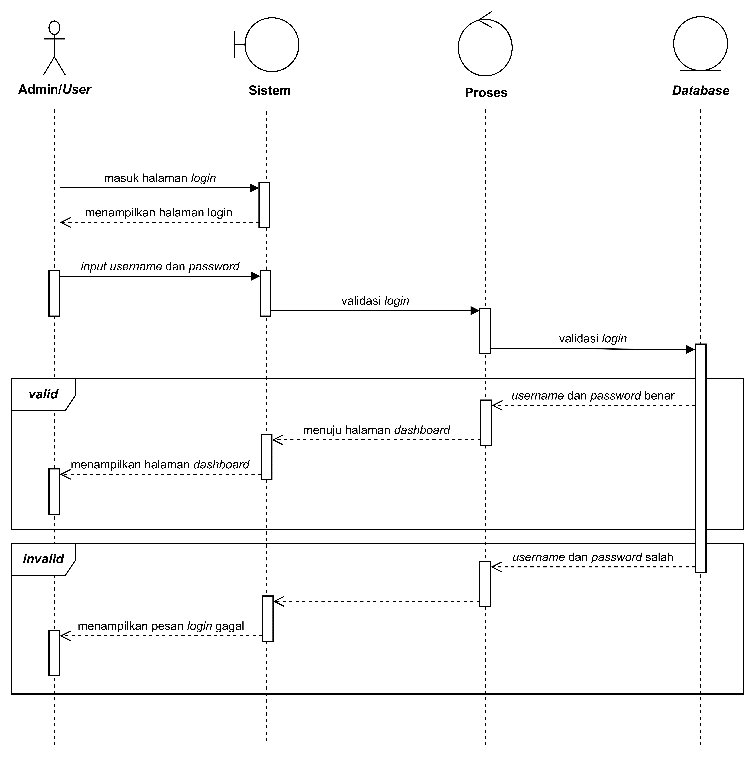
**Gambar 3. 10 *Activity* *Diagram* Hasil Klasifikasi Tingkat Kematangan SPBE**

1. *Sequence Diagram*

*Sequence diagram* adalah diagram urutan yang digunakan untuk menjelaskan interaksi antar objek-objek yang ada pada sistem secara terperinci berdasarkan urutan waktu.

1. *Login*

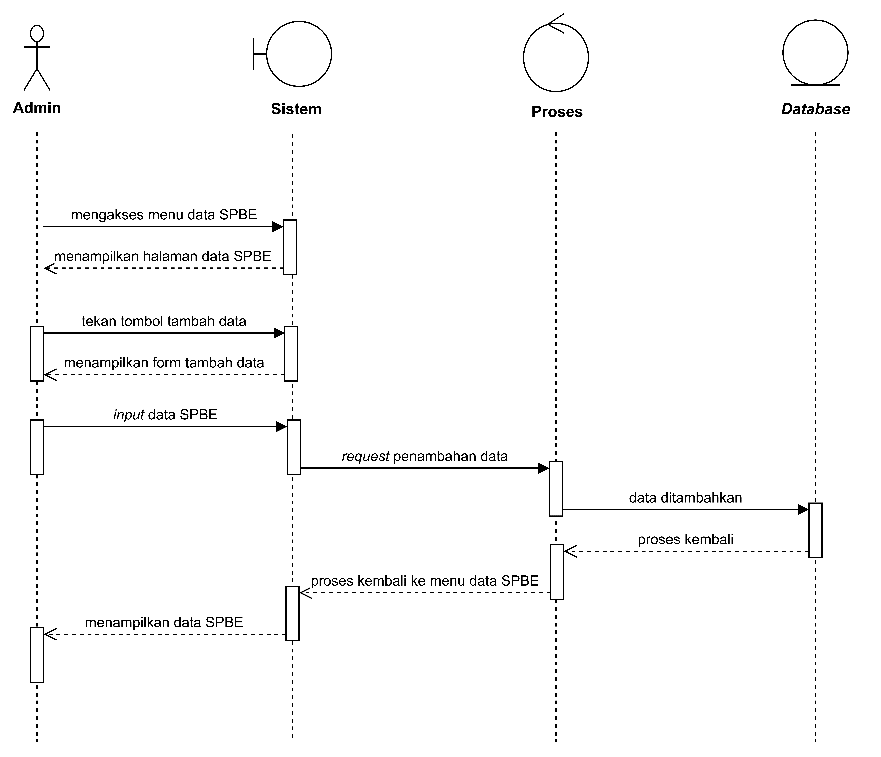
Admin atau *user* mengakses halaman *login* kemudian memasukkan *username* dan *password*. Sistem akan melakukan validasi kesesuaian data di *database*. Jika *username* dan *password* valid, maka halaman *dashboard* akan ditampilkan. Namun, jika *invalid*, sistem akan menampilkan pesan gagal *login*. *Sequence* diagram *login* dapat dilihat pada Gambar 3.11 berikut.



**Gambar 3. 11 *Sequence Diagram Login***

1. Tambah data SPBE

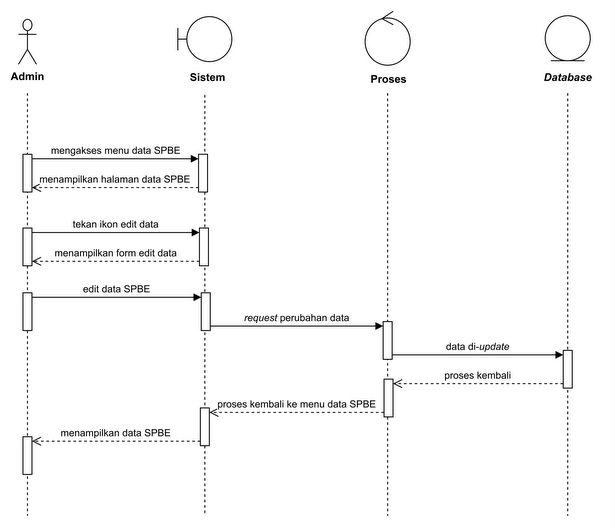
Admin dapat menambahkan data SPBE dengan menekan tombol tambah data dan *form* tambah data akan ditampilkan oleh sistem. Admin mengisi data SPBE, kemudian sistem akan mengirim permintaan penambahan data sehingga data akan ditambahkan di *database*. Setelah itu, data SPBE akan ditampilkan. *Sequence* diagram tambah data SPBE dapat dilihat pada Gambar 3.12 berikut.



**Gambar 3. 12 *Sequence Diagram* Tambah Data Latih**

1. *Update* data SPBE

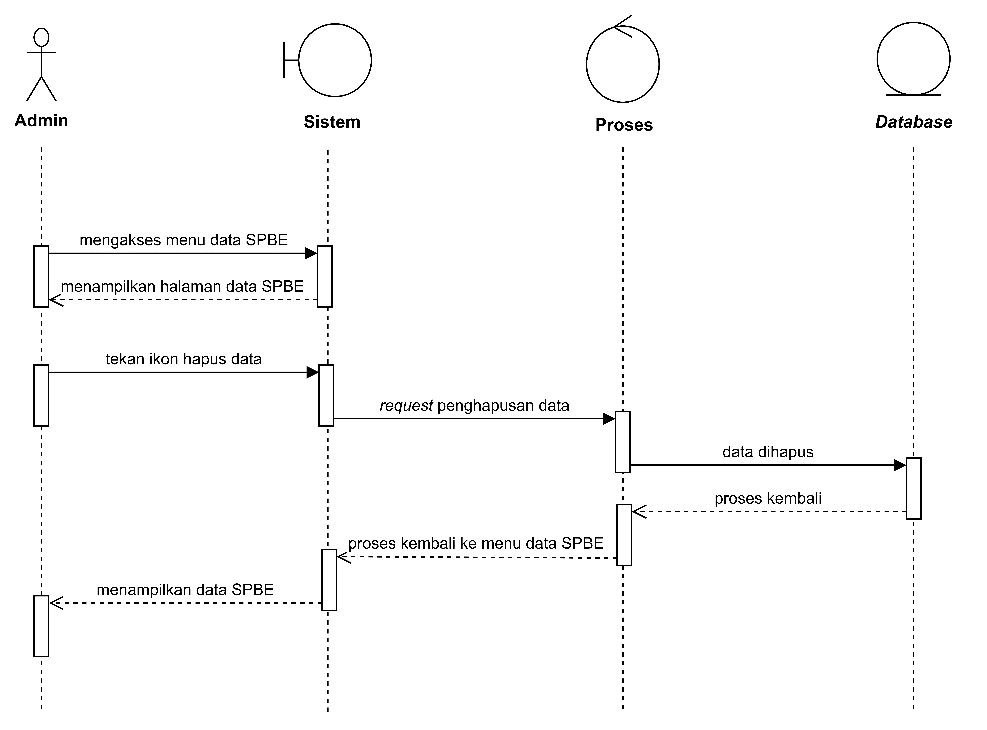
Admin dapat melakukan perubahan data SPBE dengan menekan ikon edit data. Sistem akan menampilkan *form* edit lalu admin akan melakukan perubahan data. Sistem akan mengirim permintaan perubahan data dan data di *database* akan di-*update* kemudian ditampilkan kepada admin. *Sequence* diagram *update* data SPBE dapat dilihat pada Gambar 3.13 berikut.



**Gambar 3. 13 *Sequence Diagram Update* Data Latih**

1. Hapus data SPBE

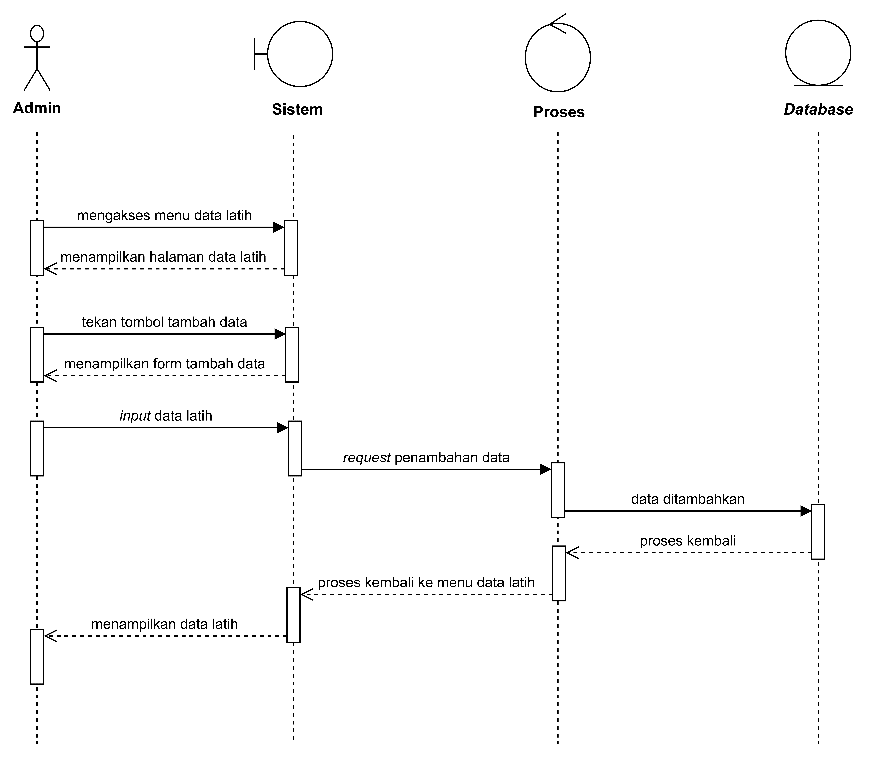
Admin dapat melakukan penghapusan data dengan menekan ikon *delete*. Sistem akan mengirim permintaan penghapusan data. Kemudian, data akan dihapus di *database*. *Sequence* diagram hapus data SPBE dapat dilihat pada Gambar 3.14 berikut.



**Gambar 3. 14 *Sequence Diagram* Hapus Data SPBE**

1. Tambah data latih

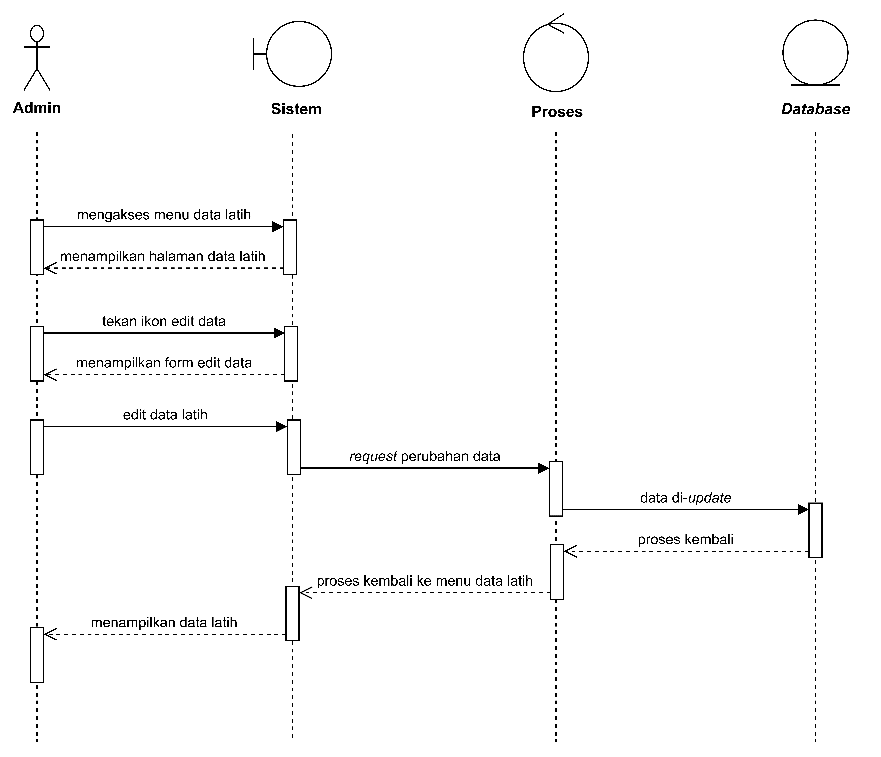
Admin dapat menambahkan data latih dengan menekan tombol tambah data dan *form* tambah data akan ditampilkan oleh sistem. Admin mengisi data latih, kemudian sistem akan mengirim permintaan penambahan data sehingga data akan ditambahkan di *database*. Setelah itu, data latih akan ditampilkan. *Sequence* diagram tambah data latih dapat dilihat pada Gambar 3.15 berikut.



**Gambar 3. 15 *Sequence Diagram* Tambah Data Latih**

1. *Update* data latih

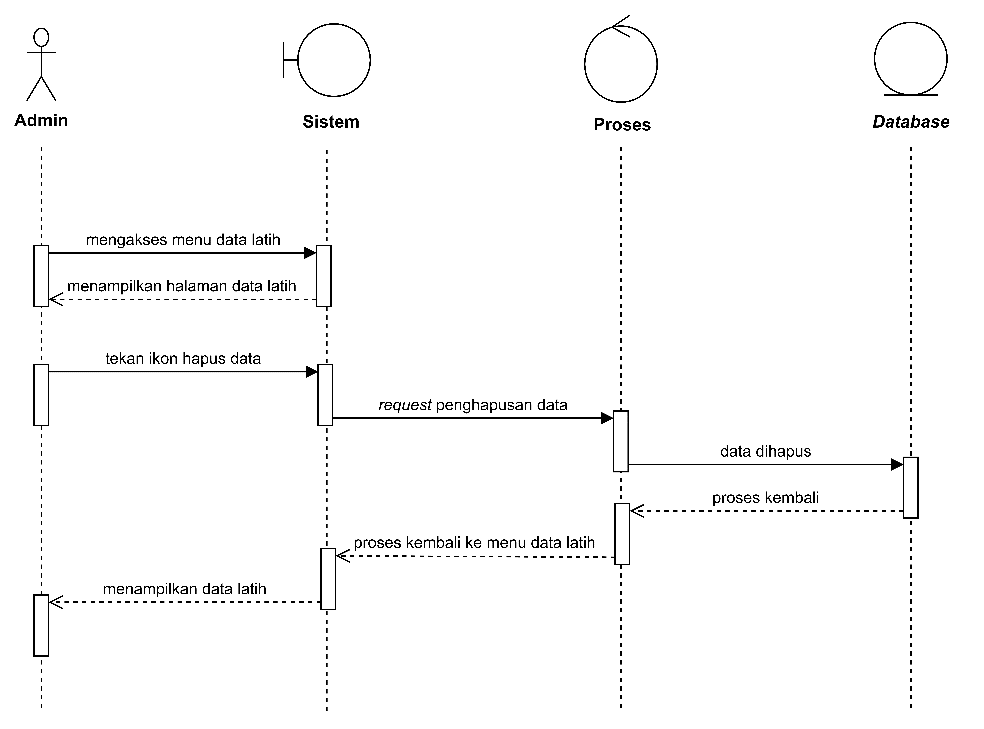
Admin dapat melakukan perubahan data latih dengan menekan ikon edit data. Sistem akan menampilkan *form* edit lalu admin akan melakukan perubahan data. Sistem akan mengirim permintaan perubahan data dan data di *database* akan di-*update* kemudian ditampilkan kepada admin. *Sequence* diagram *update* data latih dapat dilihat pada Gambar 3.16 berikut.



**Gambar 3. 16 *Sequence Diagram* *Update* Data Latih**

1. Hapus data latih

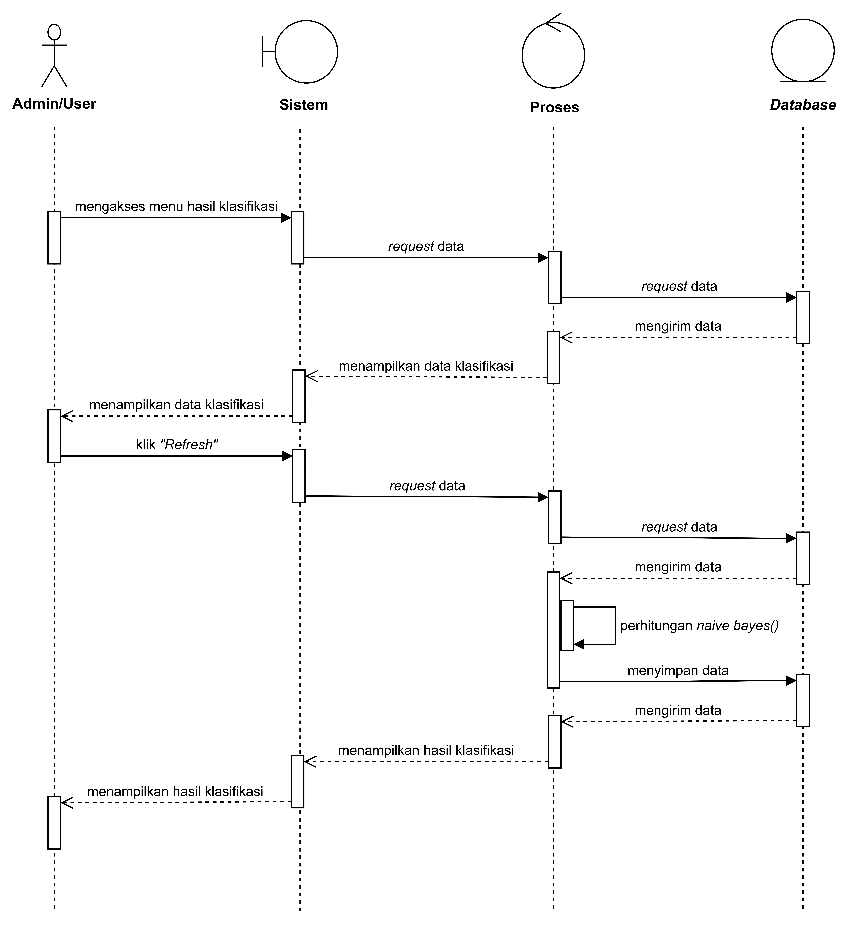
Admin dapat melakukan penghapusan data dengan menekan ikon *delete*. Sistem akan mengirim permintaan penghapusan data. Kemudian, data akan dihapus di *database*. *Sequence* diagram hapus data latih dapat dilihat pada Gambar 3.17 berikut.



**Gambar 3. 17 *Sequence Diagram* Hapus Data Latih**

1. Klasifikasi tingkat kematangan SPBE

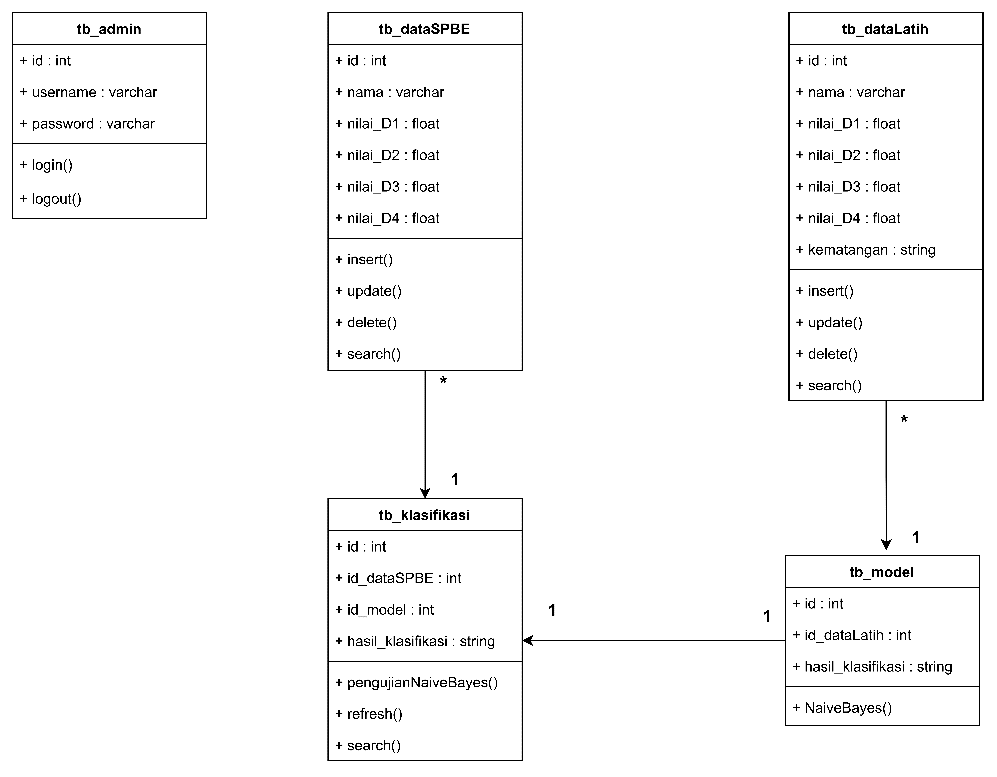
Admin atau *user* dapat melihat hasil klasifikasi dengan mengakses menu hasil klasifikasi. Sistem akan menampilkan data klasifikasi kepada admin atau *user.* Kemudian, untuk melihat hasil klasifikasi terbaru, admin atau *user* menekan tombol *refresh*. Sistem akan mengirim permintaan data klasifikasi terbaru yang sudah melalui perhitungan Naïve Bayes. Setelah tersimpan di *database*, hasil klasifikasi terbaru akan ditampilkan. *Sequence* diagram hasil klasifikasi dapat dilihat pada Gambar 3.18 berikut.



**Gambar 3. 18 *Sequence Diagram* Halaman Klasifikasi**

1. *Class Diagram*

*Class diagram* adalah diagram UML yang menunjukkan hubungan antar kelas-kelas dalam sistem. Berdasarkan Gambar 3.19, sistem yang dibangun memiliki 5 kelas, yaitu tb\_admin, tb\_dataSPBE, tb\_dataLatih, tb\_model, dan tb\_klasifikasi. Kelas-kelas tersebut saling berelasi satu sama lain.



**Gambar 3. 19 *Class Diagram***

## Analisis Perancangan *User* *Interface*

Analisis perancangan *user interface* adalah perancangan desain awal sistem yang akan berinteraksi langsung dengan pengguna. Perancangan ini berfokus pada tampilan sistem yang akan dibangun.

### Admin (Penilai Kematangan SPBE)

1. *Login*

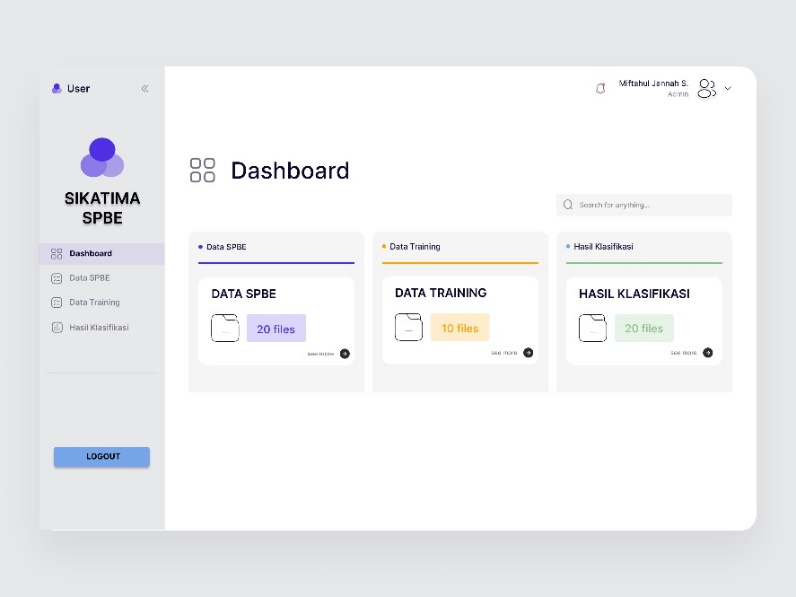
Halaman ini digunakan ketika admin atau *user* melakukan *login* ke dalam sistem. Tampilan halaman ini dapat dilihat pada Gambar 3.20 berikut.



**Gambar 3. 20 Halaman *Login***

1. *Dashboard* Admin

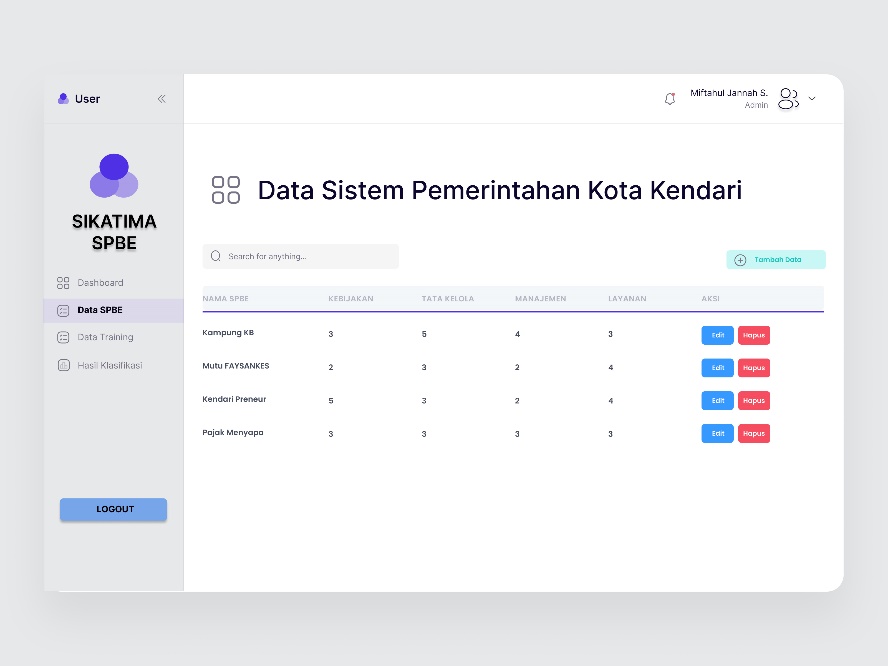
Halaman ini adalah tampilan awal ketika admin berhasil *login* ke dalam sistem. Tampilan halaman ini dapat dilihat pada Gambar 3.21 berikut.



**Gambar 3. 21 *Dashboard* Admin**

1. Menu Data SPBE

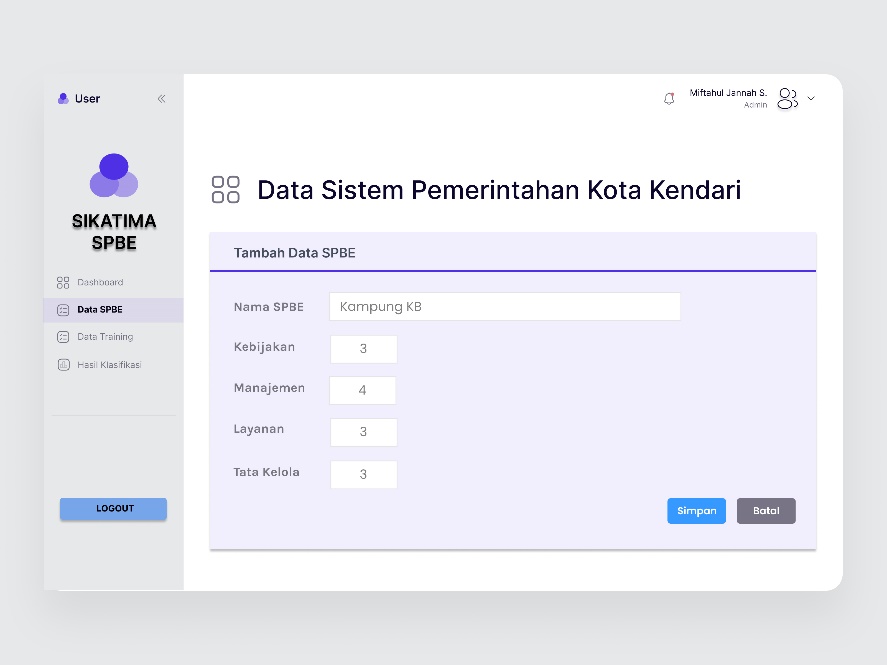
Halaman ini adalah halaman yang menampilkan data SPBE yang akan digunakan sebagai data *testing* nantinya untuk menguji model *Naïve Bayes*. Tampilan halaman ini dapat dilihat pada Gambar 3.22 berikut.



**Gambar 3. 22 Menu Data SPBE**

1. *Form Input* Data SPBE

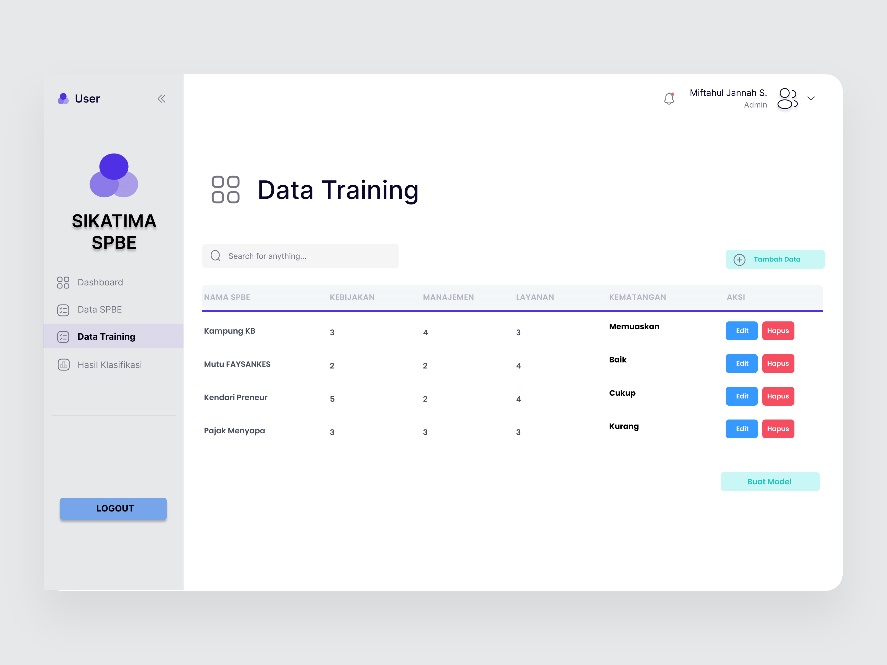
Tampilan *form input* yang digunakan admin untuk menambahkan data SPBE dapat dilihat pada Gambar 3.23.



**Gambar 3. 23 *Form Input* Data SPBE**

1. Menu Data *Training*

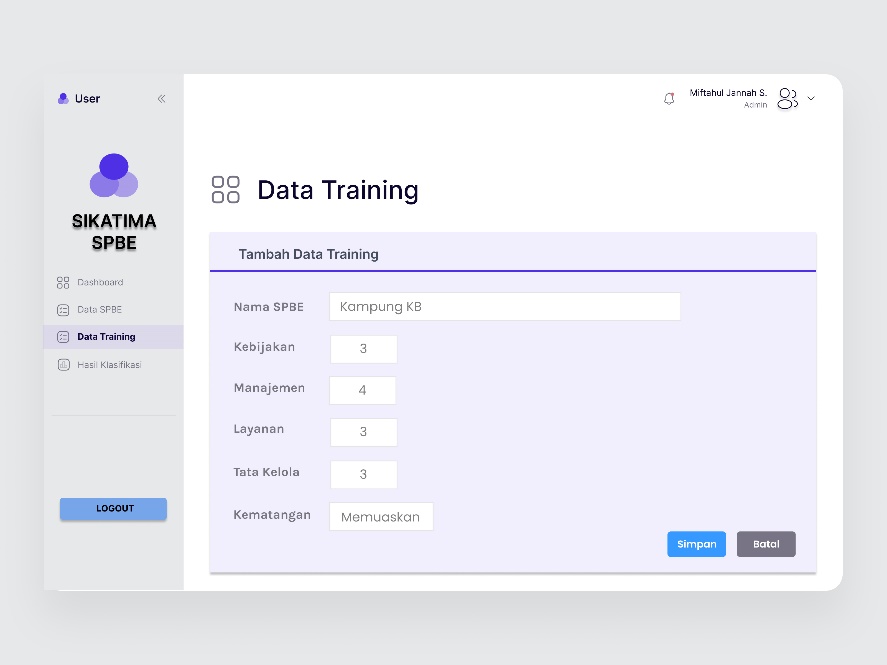
Halaman ini adalah halaman yang menampilkan menu data latih. Tampilan halaman ini dapat dilihat pada Gambar 3.24 berikut.



**Gambar 3. 24 Data *Training***

1. *Form Input* Data *Training*

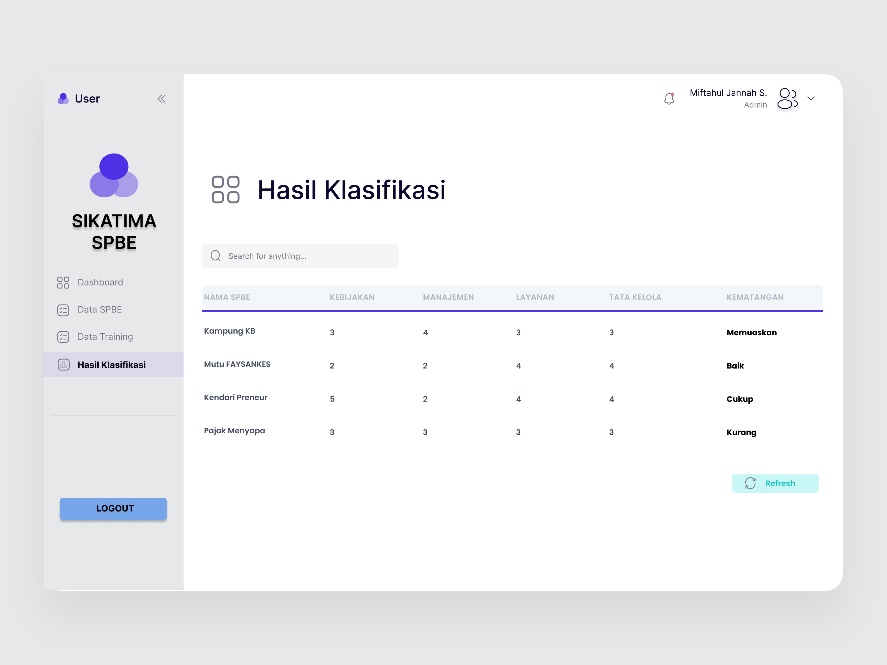
Tampilan *form input* yang digunakan admin untuk menambahkan data *training* dapat dilihat pada Gambar 3.25



**Gambar 3. 25 *Form Input* Data *Training***

1. Menu Hasil Klasifikasi

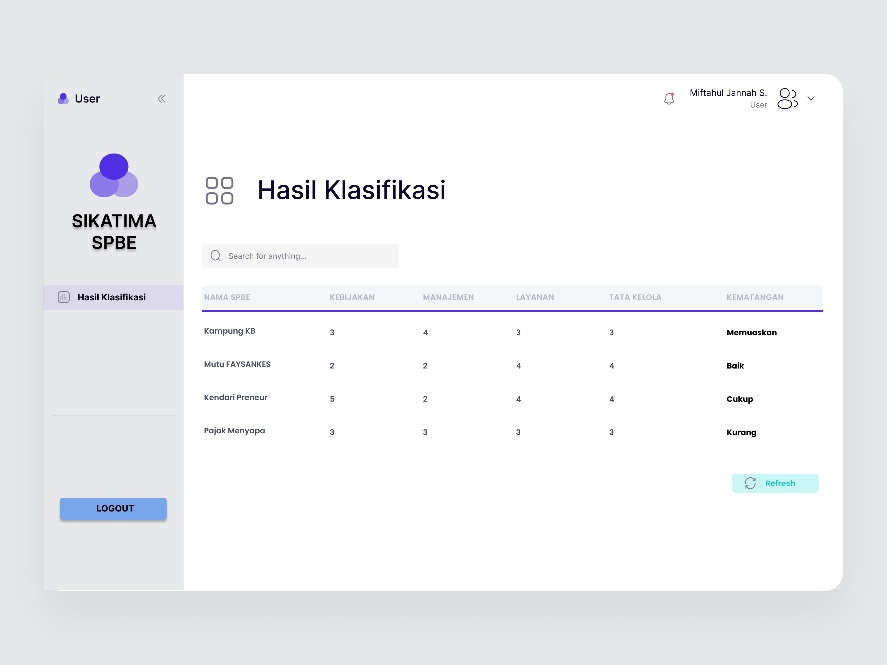
Tampilan halaman hasil klasifikasi tingkat kematangan SPBE dapat dilihat pada Gambar 3.26 berikut.



**Gambar 3. 26 Hasil Klasifikasi**

### *User* (Karyawan)

Ketika *user* berhasil *login* ke dalam sistem, halaman yang akan ditampilkan adalah halaman hasil klasifikasi tingkat kematangan SPBE. Tampilan halaman ini dapat dilihat pada Gambar 3.27 berikut.



**Gambar 3. 27 Hasil Klasifikasi**

## Skenario Pengujian

### *Confusion Matrix*

*Confusion matrix* digunakan untuk menguji performa model dengan membandingkan hasil prediksi dan nilai aktual.

**Tabel 3. 4 Contoh Tabel *Confusion Matrix***

|  |  |  |  |
| --- | --- | --- | --- |
|  | | **Aktual** | |
| **Benar** | **Salah** |
| **Prediksi** | **Benar** | 25 | 15 |
| **Salah** | 15 | 20 |

1. *Accuracy*

(3.1)

1. *Precision*

(3.2)

1. *Recall*

(3.3)

1. *F-measure*

(3.4)

Berdasarkan hasil perhitungan menggunakan *confusion matrix*, diperoleh nilai *accuracy* 53%, nilai *precision* 62,5%, nilai *recall* 55,5%, dan nilai *f-measure* 58,7%.

### *Black Box*

Pengujian *black box* adalah pengujian yang dilakukan untuk menguji spesifikasi fungsional dari sistem. Pengujian ini berfokus pada *input* yang dilakukan dan *output* yang dihasilkan oleh sistem sudah sesuai dengan apa yang diharapkan. Pengujian ini bisa digunakan sebagai bahan evaluasi fungsi dari sistem yang dibangun. Adapun rencana pengujian *black box* yang dilakukan dapat dilihat pada Tabel 3.5 berikut.

**Tabel 3. 5 Pengujian *Black Box***

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Halaman yang Diuji** | **Aksi yang Dilakukan** | **Reaksi Sistem yang Diharapkan** |
|  | Halaman *login* | *User* memasukkan *username* dan *password* | *User* terverifikasi sebagai pengguna sistem |
|  | Halaman *dashboard* | *User* mengakses halaman *dashboard* | Halaman dashboard ditampilkan |
|  | Halaman *login* | *User* memasukkan *username* dan *password* | *User* terverifikasi sebagai pengguna sistem |
|  | Halaman *dashboard* | *User* mengakses halaman *dashboard* | Halaman dashboard ditampilkan |
|  | Halaman data SPBE | *User* mengakses halaman data SPBE | Halaman data SPBE ditampilkan |
|  | Tambah data SPBE | *User* menekan tombol tambah data | Menampilkan *form* tambah data SPBE |
|  | Edit data SPBE | *User* menekan ikon edit data | Menampilkan *form* edit data SPBE |

**Tabel 3. 6** **Pengujian *Black Box* (Lanjutan)**

|  |  |  |  |
| --- | --- | --- | --- |
|  | Hapus data SPBE | *User* menekan ikon hapus data | Data SPBE terhapus |
|  | Halaman data *training* | *User* mengakses halaman data *training* | Halaman data *training* ditampilkan |
|  | Tambah data *training* | *User* menekan tombol tambah data | Menampilkan *form* tambah data *training* |
|  | Edit data *training* | *User* menekan ikon edit data | Menampilkan *form* edit data *training* |
|  | Edit data *training* | *User* menekan ikon edit data | Menampilkan *form* edit data *training* |
|  | Halaman hasil klasifikasi | *User* mengakses halaman hasil klasifikasi | Menampilkan halaman hasil klasifikasi |
|  | *Refresh* hasil klasifikasi | *User* menekan tombol *refresh* | Menampilkan hasil klasifikasi terbaru |
|  | *Logout* | *User* menekan tombol *logout* | *User* keluar dari sistem |

# DAFTAR PUSTAKA

Agustini, Kurniawan, W.J., 2019, Sistem E-Learning Do’a dan Iqro’ dalam Peningkatan Proses Pembelajaran pada TK Amal Ikhlas, *Jurnal Mahasiswa Aplikasi Teknologi Komputer dan Informasi*, 1, 3, 154–159.

Algiffary, A., 2023, Implementasi Machine Learning dengan Algoritma Naive Bayes Terhadap Sistem Informasi Pelayanan Pemberkasan Kepegawaian di BKPSDM Kota Palembang, *Indonesian Journal of Computer Science*, 12, 3, 1272–1281.

Anggraini, Y., Pasha, D., Damayanti, D., Setiawan, A., 2020, Sistem Informasi Penjualan Sepeda Berbasis Web Menggunakan Framework Codeigniter, *Jurnal Teknologi dan Sistem Informasi*, 1, 2, 64–70.

Bayu, M., Adhalik, A., Nisa, F.H., Muh, L., Jaya, G., 2024, Lokasi Industri Manufaktur Sulawesi Tenggara Berbasis Website, 9, 1, 47–58.

Bouty, A.A., Koniyo, M.H., Novian, D., 2019, The Evaluation of Electronic Based Government System Using E-Government Maturity Model (Case in Government of Gorontalo City), *Penelitian Komunikasi dan Opini Publik*, 23, 1, 16–24.

Destriana, R., Taufiq, R., Suryana, B.E., 2020, Rancang Bangun Sistem Informasi Document Managemen System pada LKP ITC-PCB Berbasis Web Menggunakan UML dan PHP, *Jurnal Inovasi Informatika*, 5, 1, 64–71.

Hakim, Z., Rahayu, S., Irawati, K., 2022, Klasifikasi Tingkat Kematangan Buah Pisang Kepok Menggunakan Algoritma Naive Bayes, *Academic Journal of Computer Science Research*, 4, 1, 8–11.

Heliyanti Susana, 2022, Penerapan Model Klasifikasi Metode Naive Bayes Terhadap Penggunaan Akses Internet, *Jurnal Riset Sistem Informasi dan Teknologi Informasi (JURSISTEKNI)*, 4, 1, 1–8.

Hidayah, E.S., Almadani, M., 2022, Analisis Tingkat Kematangan Sistem Pemerintahan Berbasis Elektronik (SPBE) pada Pemerintah Provinsi Sulawesi Selatan, *Jurnal Teknologi dan Komunikasi Pemerintahan*, 4, 2, 49–67.

Hidayati, N., Rizmayanti, A.I., Dewi, C.B.S., Fatmasari, R., Gata, W., 2020, Penerapan Algoritma Klasterisasi dan Klasifikasi pada Tingkat Kepentingan Sistem Pembelajaran di Universitas Terbuka, *Swabumi*, 8, 2, 134–142.

Jevrian Efendi, S.M., 2021, Perancangan Sistem Informasi Pelayanan Di Kantor Desa Ranah Baru Berbasis Web, *Jurnal Intra Tech*, 5, 2, 79–90.

Kurniawan, t bayu, Syarifuddin, 2020, Perancangan Sistem Aplikasi Pemesanan Makanan dan Minuman Pada Cafetaria No Caffe di TAnjung Balai Karimun Menggunakan Bahasa Pemrograman PHP dan MySQL, *Jurnal Tikar*, 1, 2, 192–206.

Martantoh, E., Yanih, N., 2022, Implementasi Metode Naïve Bayes Untuk Klasifikasi Karakteristik Kepribadiaan Siswa Di Sekolah MTS Darussa’adah Menggunakan Php Mysql, *Jurnal Teknologi Sistem Informasi*, 3, 2, 166–175.

Meylia Susiana Dewi Putri, Rudy A.G. Gultom, Achmad Farid Wadjidi, 2023,

Analisis Portabilitas Instrumen Pengukuran Sistem Pemerintahan Berbasis Elektronik (Spbe, *Jurnal Informatika Teknologi dan Sains*, 5, 1, 75–80.

Natuna, D.K. dan I.K., 2022, Arsitektur Tata Kelola Dan Manajemen SPBE, *Dinas Komunikasi Dan Informatika Kabupaten Natuna*, p. 4, Natuna.

Permatasari, A., Suhendi, S., 2020, Rancang Bangun Sistem Informasi Pengelolaan Talent Film berbasis Aplikasi Web, *Jurnal Informatika Terpadu*, 6, 1, 29–37.

Perwitasari, R., Afawani, R., Anjarwani, S.E., 2020, Penerapan Metode Rational Unified Process (RUP) Dalam Pengembangan Sistem Informasi Medical Check Up Pada Citra Medical Centre, *Jurnal Teknologi Informasi, Komputer, dan Aplikasinya (JTIKA )*, 2, 1, 76–88.

Qadrini, L., Sepperwali, A., Aina, A., 2021, Decision Tree Dan Adaboost Pada Klasifikasi Penerima Program Bantuan Sosial, *Jurnal Inovasi Penelitian*, 2, 7, 1959–1966.

Rachman, R., Handayani, R.N., 2021, Klasifikasi Algoritma Naive Bayes Dalam Memprediksi Tingkat Kelancaran Pembayaran Sewa Teras UMKM, *Jurnal Informatika*, 8, 2, 111–122.

Rahayu, W.I., Prianto, C., Novia, E.A., 2021, Perbandingan Algoritma K-Means Dan Naïve Bayes Untuk Memprediksi Prioritas Pembayaran Tagihan Rumah Sakit Berdasarkan Tingkat Kepentingan Pada Pt. Pertamina (Persero), *Jurnal Teknik Informatika*, 13, 2, 1–8.

Rahmatuloh, M., Revanda, M.R., 2022, Rancang Bangun Sistem Informasi Jasa Pengiriman Barang Pada PT. Haluan Indah Transporindo Berbasis Web, *Jurnal Teknik Informatika*, 14, 1, 54–59.

Ramdhan, N.A., Nufriana, D.A., 2019, Rancang Bangun Dan Implementasi Sistem Informasi Skripsi Oline Berbasis WEB, *Jurnal Ilmiah Intech : Information Technology Journal of UMUS*, 1, 02, 1–12.

Rayuwati, Husna Gemasih, Irma Nizar, 2022, Implementasi Algoritma Naive Bayes Untuk Memprediksi Tingkat Penyebaran Covid, *Jural Riset Rumpun Ilmu Teknik*, 1, 1, 38–46.

Rinestu, M., Made Indra, I.P., Marsanto, B., Trisakti, S., 2022, Classification Of Investment Decisions During Covid-19 Pandemic Using Naive Bayes Klasifikasi Keputusan Investasi Di Masa Pandemi Covid-19 Dengan Menggunakan Naive Bayes, *Management Studies and Entrepreneurship Journal*, 3, 4, 1784–1796.

Ronaldo, M., Pasha, D., 2021, Sistem Informasi Pengelolaan Data Santri Pondok Pesantren an-Ahl Berbasis Website, *Telefortech*, 2, 1, 17–20.

Roni Anagora, R.A., Rudini, R., Rohmat Taufiq, R.T., Ahmad Dedi Jubaedi, A.D.J., Rio Wirawan, R.W., Arman Syah Putra, 2022, The Classification of Phishing Websites using Naive Bayes Classifier Algorithm, *International Journal of Science, Technology & Management*, 3, 2, 553–562.

Rusdy, imam R., Flambonita, S., 2023, Penerapan Sistem Pemerintahan Berbasis Elektronik (Spbe) Di Pemerintah Daerah Untuk Mewujudkan Good Governance, *LexLata: Jurnal Ilmiah Ilmu Hukum*, 05, 02, 218–239.

Samuel, Y.T., DEwi, K., 2019, Penggunaan Metode NAÏVE BAYES Dalam Mengukur Tingkat Kepuasan Pengguna Terhadap Online System Universitas Advent Indonesia, *TeIKa*, 9, 02, 147–153.

Saputra, A.K., Fahrizal, M., 2021, Rancang Bangun Berbasis Web Crm (Customer Relationship Management) Berbasis Web Studi Kasus Pt Budi Berlian Motor Hajimena Bandar Lampung, *Portaldata.org*, 17, 1, 1–31.

Sigid Widodo, A.Z.M., Pandu Kusuma, A., Dwi Puspitasari, W., 2023, Analisis Algoritma Naive Bayes Classifier (Nbc) Pada Klasifikasi Tingkat Minat Barang Di Toko Violet Cell, *JATI (Jurnal Mahasiswa Teknik Informatika)*, 7, 1, 87–94.

Suryantoro, B., Kusdyana, Y., 2020, Analisis Kualitas Pelayanan Publik Pada Politeknik Pelayaran Surabaya, *Jurnal Baruna Horizon*, 3, 2, 223–229.

Tamburaka, I., Edwin, R.S., Fitriah, F., Pramono, B., Sarita, I., 2022, Peningkatan Pemasaran Industri Paving Block Berbasis Online di Masa Pandemi Covid 19 Di Kecamatan Kendari Kota Kendari Provinsi Sulawesi Tenggara, *Dinamisia : Jurnal Pengabdian Kepada Masyarakat*, 6, 4, 897–904.

Taqiya, N.A., Mukaromah, S., Pratama, A., 2020, Analisis Tingkat Kematangan Spbe Di Dinas Komunikasi Dan Informatika Jawa Timur, *SCAN - Jurnal Teknologi Informasi dan Komunikasi*, 15, 1, 22–33.

Wulandari, T., Nurmiati, S., 2022, Rancang Bangun Sistem Pemesanan Wedding Organizer Menggunakan Metode Rad di Shofia Ahmad Wedding, *Jurnal Rekasaya Informasi*, 11, 69, 79–85.

Yahya Dwi, W., Muna Wardah, A., 2021, Pengujian Blackbox Sistem Informasi Penilaian Kinerja Karyawan Pt Inka (Persero) Berbasis Equivalence Partitions Blackbox Testing of Pt Inka (Persero) Employee Performance Assessment Information System Based on Equivalence Partitions, *Jurnal Digital Teknologi Informasi*, 4, 1, 22–26.