

PENERAPAN ALGORITMA NAIVE BAYES DALAM KLASIFIKASI PRIORITAS PERMASALAHAN OPERASIONAL BERBASIS WEB DI PT VUTEQ INDONESIA

Ilham Avin Pratama^{1*}, Nadya Safitri²

^{1,2} Program Studi Teknik Informatika, Fakultas Informatika dan Desain, Universitas Bina Insani
Jl. Siliwangi No. 6, Sepanjang Jaya, Rawalumbu, Kota Bekasi, Jawa Barat 17114

Keywords:

Naïve Bayes, Text
Classification, Operational
Problem, Priority
Classification, Web
Application.

Correspondent Email:

ilhamavin2@gmail.com

Abstrak. Pengelolaan laporan permasalahan operasional di PT Vuteq Indonesia masih dilakukan secara manual sehingga penentuan prioritas penanganan belum berlangsung secara konsisten. Penelitian ini bertujuan mengembangkan aplikasi berbasis web yang mampu mengklasifikasikan prioritas permasalahan operasional menggunakan algoritma *Naïve Bayes*. Tahapan klasifikasi meliputi *case folding*, *tokenizing*, *stopword removal*, *stemming*, transformasi *TF-IDF*, dan klasifikasi menggunakan *Multinomial Naïve Bayes*. Sistem dikembangkan dengan metode *Agile Development* menggunakan *Laravel* dan *MySQL*. Pengujian dilakukan melalui *Black Box Testing* dan *User Acceptance Testing* (UAT). Hasil pengujian menunjukkan seluruh fungsi sistem berjalan sesuai kebutuhan, sedangkan UAT memperoleh nilai 85,73% yang termasuk kategori sangat layak. Sistem yang dikembangkan mampu membantu proses penentuan prioritas laporan operasional secara lebih terstruktur dan konsisten.



Copyright © 2026 The Author(s),
Published by Jurnal Informatika dan
Teknik Elektro Terapan. This is an
open-access article distributed under
the terms of the Creative Commons
Attribution-NonCommercial 4.0
International License (CC BY-NC
4.0).

Abstract. Operational problem reports at PT Vuteq Indonesia were still managed manually, causing inconsistent priority determination. This study developed a web-based application to classify operational problem priorities using the *Naïve Bayes* algorithm. The classification process included *case folding*, *tokenization*, *stopword removal*, *stemming*, *TF-IDF* transformation, and *Multinomial Naïve Bayes* classification. The system was developed using the *Agile Development* approach with *Laravel* and *MySQL*. Functional testing employed *Black Box Testing*, while user acceptance was evaluated using *User Acceptance Testing* (UAT). The results showed that all system functions operated properly, and the UAT score reached 85.73%, indicating that the system was highly feasible. The proposed system can support more structured and consistent operational priority classification.

1. PENDAHULUAN

PT Vuteq Indonesia merupakan perusahaan manufaktur yang memproduksi berbagai komponen otomotif untuk memenuhi kebutuhan industri kendaraan. Dalam menjalankan proses bisnisnya, perusahaan menghasilkan berbagai laporan permasalahan operasional yang berasal dari aktivitas produksi maupun kegiatan pendukung lainnya. Informasi

yang terdapat pada laporan tersebut menjadi dasar dalam menentukan langkah penanganan sehingga proses identifikasi dan penentuan prioritas perlu dilakukan secara tepat agar permasalahan yang bersifat mendesak dapat segera ditindaklanjuti.

Perkembangan teknologi *Artificial Intelligence (AI)*, khususnya pada bidang *machine learning*, telah mendorong

pemanfaatan berbagai metode klasifikasi untuk mendukung proses pengambilan keputusan berbasis data. Salah satu metode yang banyak digunakan adalah algoritma *Naïve Bayes* karena memiliki mekanisme klasifikasi berbasis probabilitas yang sederhana, proses komputasi yang relatif cepat, serta mampu memberikan performa yang baik pada data berbentuk teks. Penelitian oleh Chen et al. (2021) menunjukkan bahwa algoritma *Naïve Bayes* mampu meningkatkan proses klasifikasi data risiko melalui pendekatan probabilistik sehingga menghasilkan model yang akurat untuk mengelompokkan data ke dalam kategori tertentu [1].

Pemanfaatan algoritma *Naïve Bayes* juga telah diterapkan pada berbagai penelitian klasifikasi dokumen dan teks. Penelitian oleh Gan et al. (2021) mengembangkan metode *Hidden Naïve Bayes* untuk meningkatkan akurasi klasifikasi dokumen dengan mempertimbangkan hubungan antarfitur pada data teks [2]. Selanjutnya, penelitian oleh Liu (2023) menunjukkan bahwa kombinasi *Naïve Bayes* dengan *feature engineering* dan teknik *ensemble* mampu meningkatkan performa klasifikasi dibandingkan pendekatan konvensional [3]. Sementara itu, penelitian oleh Naulak dan Jindal (2022) menjelaskan bahwa penggunaan representasi fitur seperti *TF-IDF* dan *n-gram* memberikan pengaruh terhadap peningkatan hasil klasifikasi dokumen berbasis teks [4].

Meskipun berbagai penelitian tersebut menunjukkan keberhasilan penerapan *Naïve Bayes* pada klasifikasi teks, implementasinya dalam proses penentuan prioritas permasalahan operasional pada perusahaan manufaktur masih relatif terbatas. Sebagian besar penelitian lebih berfokus pada klasifikasi sentimen, klasifikasi dokumen, maupun prediksi penjualan, sedangkan penelitian yang secara khusus mengelompokkan laporan operasional berdasarkan tingkat prioritas masih belum banyak ditemukan. Selain itu, hasil klasifikasi pada penelitian sebelumnya umumnya hanya menghasilkan kategori data tanpa menyediakan informasi yang dapat dimanfaatkan sebagai dasar penentuan tindakan penanganan [5].

Berdasarkan hasil observasi selama pelaksanaan kerja praktik di PT Vuteq Indonesia, proses penentuan prioritas laporan permasalahan operasional masih dilakukan

secara manual berdasarkan penilaian masing-masing pihak. Kondisi tersebut berpotensi menimbulkan perbedaan penilaian terhadap tingkat urgensi suatu permasalahan sehingga laporan yang memiliki dampak besar tidak selalu memperoleh penanganan terlebih dahulu. Selain itu, meningkatnya jumlah laporan operasional menyebabkan proses identifikasi menjadi kurang akurat apabila dilakukan tanpa dukungan sistem yang mampu mengelompokkan laporan secara otomatis.

Oleh karena itu, penelitian ini mengembangkan aplikasi berbasis *web* yang menerapkan algoritma *Naïve Bayes* untuk mengklasifikasikan prioritas permasalahan operasional ke dalam kategori *urgent*, *middle*, dan *low*. Sebelum proses klasifikasi dilakukan, data laporan diproses melalui tahapan *case folding*, *tokenizing*, *stopword removal*, *stemming*, dan pembobotan *TF-IDF* untuk meningkatkan kualitas representasi data. Sistem yang dikembangkan diharapkan mampu membantu perusahaan dalam menentukan prioritas penanganan secara lebih objektif, konsisten, serta mendukung proses pengambilan keputusan berdasarkan informasi yang dihasilkan dari hasil klasifikasi.

2. TINJAUAN PUSTAKA

2.1. *Naïve Bayes*

Naïve Bayes merupakan salah satu algoritma *machine learning* yang digunakan untuk menyelesaikan permasalahan klasifikasi berdasarkan pendekatan probabilistik. Menurut penelitian Gan et al. (2021), metode ini memanfaatkan Teorema *Bayes* untuk menghitung peluang suatu data termasuk ke dalam kelas tertentu berdasarkan informasi yang dimiliki oleh data tersebut [2]. Pendekatan ini banyak diterapkan pada proses klasifikasi teks karena memiliki proses komputasi yang sederhana namun tetap mampu menghasilkan performa yang baik pada data berdimensi tinggi.

Pada klasifikasi teks, setiap dokumen dipandang sebagai sekumpulan kata yang mewakili informasi tertentu. Algoritma kemudian menghitung probabilitas setiap kategori berdasarkan kemunculan kata-kata pada data latih. Kategori dengan nilai probabilitas terbesar dipilih sebagai hasil klasifikasi.

Dalam implementasi klasifikasi teks, persamaan tersebut disederhanakan menggunakan asumsi bahwa setiap fitur atau kata bersifat independen terhadap fitur lainnya pada kategori yang sama. Menurut Hofmeyr et al. (2025), asumsi independensi tersebut membuat proses perhitungan menjadi lebih akurat tanpa mengurangi kemampuan metode dalam menyelesaikan berbagai kasus klasifikasi dokumen [6].

Lebih lanjut, Chen et al. (2021) menjelaskan bahwa *Naïve Bayes* memiliki beberapa varian, seperti *Gaussian Naïve Bayes*, *Bernoulli Naïve Bayes*, dan *Multinomial Naïve Bayes*, yang digunakan sesuai karakteristik data. Pada penelitian ini digunakan *Multinomial Naïve Bayes* karena sesuai untuk mengolah data teks yang telah direpresentasikan menggunakan pembobotan *TF-IDF* [2].

2.2. TF-IDF

Term Frequency-Inverse Document Frequency (TF-IDF) merupakan teknik pembobotan kata yang digunakan untuk mengubah data teks menjadi representasi numerik sebelum diproses oleh algoritma klasifikasi. Pembobotan ini bertujuan untuk mengukur tingkat kepentingan suatu kata dalam sebuah dokumen dengan mempertimbangkan frekuensi kemunculan kata pada dokumen tersebut serta penyebarannya pada seluruh kumpulan dokumen. Dengan pendekatan tersebut, kata yang memiliki informasi lebih spesifik akan memperoleh bobot yang lebih tinggi dibandingkan kata yang sering muncul pada hampir semua dokumen [3].

Penggunaan *TF-IDF* banyak diterapkan pada proses klasifikasi teks karena mampu meningkatkan kualitas representasi fitur yang digunakan oleh algoritma *machine learning*. Menurut penelitian Naulak dan Jindal (2022), kombinasi antara teknik pembobotan *TF-IDF* dan algoritma *Naïve Bayes* mampu menghasilkan performa klasifikasi yang lebih baik dibandingkan penggunaan frekuensi kata secara langsung, terutama pada data berbentuk dokumen teks [4].

Dalam penelitian ini, proses pembobotan *TF-IDF* dilakukan setelah tahapan *case folding*, *tokenizing*, *stopword removal*, dan *stemming*. Hasil pembobotan tersebut kemudian digunakan sebagai masukan bagi algoritma *Multinomial Naïve Bayes* untuk membentuk

model klasifikasi prioritas permasalahan operasional. Representasi numerik yang dihasilkan oleh *TF-IDF* membantu algoritma dalam membedakan karakteristik setiap laporan sehingga proses klasifikasi dapat dilakukan dengan lebih akurat [5].

2.3. Agile Development

Model pengembangan sistem yang diterapkan pada penelitian ini adalah *Agile Development*. Menurut penelitian Waja et al. (2021), *Agile* merupakan pendekatan pengembangan perangkat lunak yang dilakukan secara bertahap (*iterative*) dan berulang (*incremental*), sehingga setiap proses dapat dievaluasi dan diperbaiki sesuai kebutuhan pengguna [7]. Pendekatan ini memungkinkan pengembang melakukan penyesuaian terhadap perubahan kebutuhan tanpa harus mengulang keseluruhan proses pengembangan.

Karakteristik utama *Agile* adalah membagi proses pengembangan menjadi beberapa siklus kerja yang lebih kecil sehingga setiap tahap menghasilkan peningkatan terhadap sistem yang sedang dibangun. Melalui pendekatan tersebut, pengembang dapat melakukan pengujian, evaluasi, dan penyempurnaan secara berkelanjutan sebelum sistem digunakan oleh pengguna.

Menurut Ibarra et al. (2024), *Agile* memiliki berbagai kerangka kerja seperti *Scrum*, *Extreme Programming (XP)*, dan *Kanban* yang dapat dipilih sesuai kebutuhan proyek. Setiap kerangka kerja memiliki mekanisme pengembangan yang berbeda, namun tetap berorientasi pada kolaborasi, komunikasi, serta kemampuan beradaptasi terhadap perubahan selama proses pengembangan berlangsung [8].

Selain itu, penelitian Wang et al. (2024) menjelaskan bahwa keberhasilan implementasi *Agile* tidak hanya dipengaruhi oleh proses pengembangan perangkat lunak, tetapi juga oleh koordinasi antaranggota tim, pertukaran pengetahuan, dan konsistensi penerapan praktik pengembangan pada tingkat organisasi [9]. Oleh karena itu, *Agile* dipilih dalam penelitian ini karena mampu mendukung proses pengembangan sistem klasifikasi prioritas secara bertahap, fleksibel, serta memudahkan penyempurnaan sistem berdasarkan hasil evaluasi pada setiap iterasi.

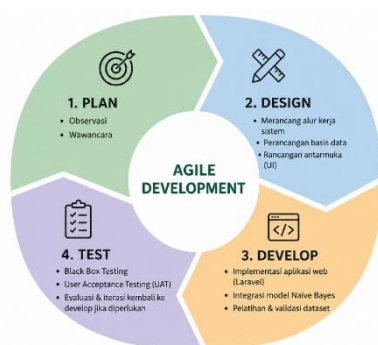
3. METODE PENELITIAN

Penelitian ini menggunakan pendekatan *Agile Development* sebagai metode pengembangan sistem, sedangkan proses klasifikasi memanfaatkan algoritma *Multinomial Naïve Bayes*. *Agile* dipilih karena memungkinkan pengembangan dilakukan secara bertahap melalui proses perencanaan, implementasi, pengujian, dan evaluasi secara berulang sehingga sistem dapat disesuaikan dengan kebutuhan pengguna selama proses pengembangan berlangsung [7].

Data penelitian berupa laporan permasalahan operasional yang terdiri atas judul laporan, deskripsi permasalahan, divisi, lokasi, dan dampak permasalahan. *Dataset* yang digunakan telah diberi label prioritas *urgent*, *middle*, dan *low* sebagai data latih untuk membangun model klasifikasi. Seluruh proses pengolahan data dilakukan menggunakan bahasa pemrograman *Python* dengan pustaka *Scikit-learn*, sedangkan implementasi sistem dikembangkan menggunakan *framework Laravel*.

3.1. Tahapan Penelitian

Tahapan penelitian terdiri atas pengumpulan data, *preprocessing*, pelatihan model, implementasi sistem, dan pengujian. Seluruh proses dilakukan secara berurutan hingga sistem mampu memberikan hasil klasifikasi prioritas secara otomatis.



Gambar 1. Alur Tahapan Penelitian

Sumber: Diadaptasi Dari [10]

3.2. Pengumpulan Data

Data penelitian diperoleh dari laporan permasalahan operasional PT Vuteq Indonesia yang terdiri atas judul laporan, deskripsi masalah, lokasi, divisi, dampak, kategori prioritas, dan rekomendasi penanganan.

Dataset tersebut digunakan sebagai data latih dalam pembentukan model klasifikasi.

3.3. Preprocessing Data

Sebelum dilakukan proses pelatihan model, data teks terlebih dahulu melalui tahapan *preprocessing* untuk meningkatkan kualitas data. Tahapan *preprocessing* terdiri atas: *Case Folding*, *Tokenizing*, *Stopword Removal*, dan *Stemming*. Tahapan tersebut bertujuan menghilangkan kata yang tidak relevan serta menyeragamkan bentuk kata sehingga proses klasifikasi menjadi lebih optimal [6].

Tahapan	Hasil
Input Asli	"Mesin Produksi Mengalami Overheating Sehingga Produksi Berhenti Total"
Case Folding	"mesin produksi mengalami overheating sehingga produksi berhenti total"
Tokenizing	["mesin", "produksi", "mengalami", "overheating", "sehingga", "produksi", "berhenti", "total"]
Stopword Removal	["mesin", "produksi", "mengalami", "overheating", "produksi", "berhenti", "total"]
Stemming	["mesin", "produksi", "alami", "overheating", "produksi", "henti", "total"]

Gambar 2. Tahap Preprocessing Text

Sumber: Hasil Penelitian (2026)

3.4. Pembobotan TF-IDF

Setelah *preprocessing* selesai, setiap dokumen direpresentasikan menggunakan metode *Term Frequency-Inverse Document Frequency (TF-IDF)*. Pembobotan ini digunakan untuk memberikan nilai penting pada setiap kata berdasarkan frekuensi kemunculannya dalam dokumen sehingga dapat meningkatkan kualitas fitur yang digunakan pada proses klasifikasi [3]

$$TFIDF = TF \times IDF$$

Sumber: Liu et al.[3]

3.5. Klasifikasi Menggunakan Naïve Bayes

Model klasifikasi dibangun menggunakan algoritma *Naïve Bayes* yang tersedia pada pustaka *Scikit-learn*. Algoritma ini menghitung probabilitas setiap kelas berdasarkan distribusi kata pada data latih, kemudian menentukan kategori dengan probabilitas terbesar sebagai hasil klasifikasi [1].

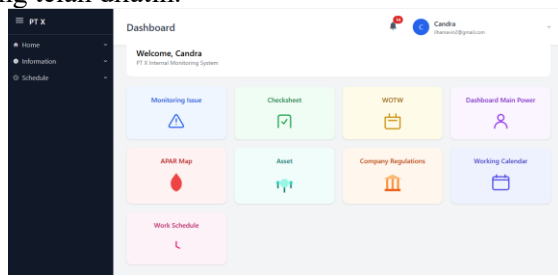
$$P(C|X) = P(C) \times P(x_1|C) \times \dots \times P(x_n|C)$$

Sumber: Chen et al. [2]

3.6. Implementasi Sistem

Model klasifikasi diintegrasikan ke dalam aplikasi berbasis *web* menggunakan *framework Laravel*. Data laporan yang dimasukkan pengguna akan diproses secara otomatis mulai dari *preprocessing*, pembobotan *TF-IDF*,

hingga proses klasifikasi menggunakan model yang telah dilatih.



Gambar 3. Tampilan Halaman Utama

Sumber: Hasil Penelitian (2026)

3.7. Pengujian Sistem

Pengujian dilakukan menggunakan dua metode, yaitu *Black Box Testing* untuk mengevaluasi fungsi sistem serta *User Acceptance Testing (UAT)* untuk mengetahui tingkat penerimaan pengguna terhadap aplikasi yang dikembangkan. Pengujian *UAT* menggunakan skala *Likert* lima tingkat untuk memperoleh nilai persentase kepuasan pengguna [11].

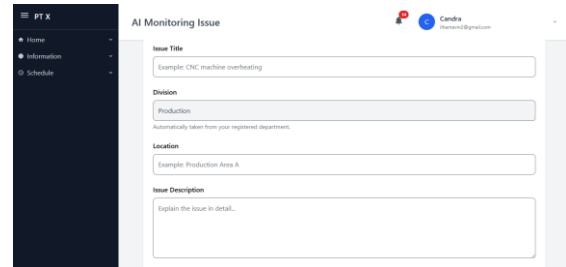
4. HASIL DAN PEMBAHASAN

4.1. Hasil Pengembangan Sistem

Sistem dikembangkan menggunakan *framework Laravel* sebagai aplikasi berbasis *web*, sedangkan model klasifikasi dibangun menggunakan algoritma *Multinomial Naïve Bayes* melalui pustaka *Scikit-learn*. Proses klasifikasi dilakukan secara otomatis setelah pengguna menginput data laporan yang terdiri atas judul, deskripsi, lokasi, divisi, dan dampak permasalahan. Integrasi antara aplikasi *web* dan model *machine learning* memungkinkan proses klasifikasi dilakukan secara langsung tanpa memerlukan analisis manual oleh pengguna [12].

Pada tahap implementasi, aplikasi *web* berperan sebagai antarmuka yang digunakan pengguna untuk memasukkan data laporan, sedangkan model klasifikasi bertugas mengolah data tersebut melalui tahapan *preprocessing*, pembobotan *TF-IDF*, dan proses prediksi menggunakan model yang telah dilatih sebelumnya. Hasil klasifikasi kemudian dikirim kembali ke aplikasi untuk ditampilkan bersama rekomendasi tindakan sesuai kategori prioritas yang dihasilkan. Menurut penelitian Setiawan et al. (2025), integrasi antara aplikasi berbasis *web* dan model *Naïve Bayes* dapat membantu proses pengolahan data menjadi lebih cepat,

terstruktur, serta mendukung pengambilan keputusan berdasarkan hasil klasifikasi secara otomatis [12].



Gambar 4. Tampilan Input Laporan

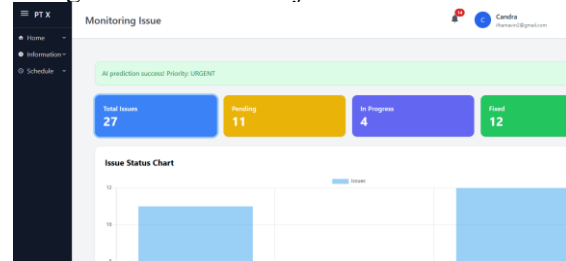
Sumber: Hasil Penelitian (2026)

4.2. Proses Klasifikasi

Data laporan yang diterima sistem terlebih dahulu diproses menggunakan tahapan *case folding*, *tokenizing*, *stopword removal*, dan *stemming*. Hasil *preprocessing* kemudian diubah menjadi representasi numerik menggunakan metode *TF-IDF* sebelum diproses oleh model *Multinomial Naïve Bayes* untuk menentukan kategori prioritas. Pendekatan tersebut mampu mengurangi kata yang tidak relevan sehingga proses klasifikasi menjadi lebih representatif terhadap isi laporan [13].

4.3. Hasil Klasifikasi

Model menghasilkan tiga kategori prioritas, yaitu *Urgent*, *Middle*, dan *Low*. Kategori ditentukan berdasarkan nilai probabilitas terbesar yang dihitung dari setiap kata pada laporan operasional. Selain menampilkan kategori prioritas, sistem juga memberikan rekomendasi penanganan sesuai dengan tingkat prioritas hasil klasifikasi sehingga pengguna memperoleh informasi yang dapat digunakan sebagai dasar tindak lanjut.



Gambar 5. Tampilan Hasil Klasifikasi

Sumber: Hasil Penelitian (2026)

4.4. Pengujian Sistem

Pengujian dilakukan menggunakan metode *Black Box Testing* untuk memastikan setiap fungsi sistem berjalan sesuai kebutuhan.

Seluruh fitur utama, seperti *login*, *input* laporan, *preprocessing*, klasifikasi, penyimpanan data, dan riwayat laporan, berhasil dijalankan sesuai skenario pengujian tanpa ditemukan kesalahan fungsional. Hasil tersebut menunjukkan bahwa sistem telah memenuhi kebutuhan fungsional yang dirancang [14].

Tabel 1. Black Box Testing

Kelas Pengujian	Aspek Pengujian	Jenis Pengujian	Berhasil
Halaman Login	Validasi login pengguna dan autentikasi akun	Black Box Testing	Berhasil
Halaman Input Laporan	Input data laporan operasional dan validasi form laporan	Black Box Testing	Berhasil
Sistem Klasifikasi Prioritas	Preprocessing teks, klasifikasi prioritas Naïve Bayes, dan rekomendasi tindakan	Black Box Testing	Berhasil
Halaman Hasil Klasifikasi	Menampilkan hasil kategori prioritas laporan	Black Box Testing	Berhasil
Halaman Riwayat Laporan	Menampilkan data riwayat laporan operasional	Black Box Testing	Berhasil

Kelas Pengujian	Aspek Pengujian	Jenis Pengujian	Berhasil
Database Sistem	Penyimpanan data laporan dan hasil klasifikasi ke database	Black Box Testing	Berhasil
Sistem Logout	Proses keluar akun dari sistem	Black Box Testing	Berhasil

Sumber: Hasil Penelitian (2026)

4.5. User Acceptance Testing

Evaluasi penerimaan pengguna dilakukan menggunakan *User Acceptance Testing (UAT)* dengan skala *Likert* terhadap 20 responden. Penilaian mencakup aspek kemudahan penggunaan, tampilan antarmuka, dan keandalan sistem. Hasil pengujian memperoleh nilai sebesar 85,73%, yang termasuk dalam kategori sangat layak, sehingga menunjukkan bahwa sistem dapat diterima dan digunakan untuk membantu pengelolaan laporan permasalahan operasional [15].

Tabel 2. UAT Aspek Kemudahan Penggunaan Sistem

No	KEMUDAHAN PENGGUNAAN SISTEM	PENILAIAN				
		SS	S	N	TS	STS
1	Sistem mudah digunakan	14	6	0	0	0
2	Menu pada sistem mudah dipahami	7	11	2	0	0
3	Proses input laporan mudah dilakukan	6	14	0	0	0

No	KEMUDAHAN PENGGUNAAN SISTEM	PENILAIAN				
		SS	S	N	TS	STS
4	Sistem membantu pengelolaan laporan operasional	12	3	5	0	0
5	Fitur sistem berjalan dengan baik	7	10	3	0	0
6	Sistem mudah dipelajari	8	11	1	0	0
7	Sistem tidak membingungkan saat digunakan	9	7	4	0	0
8	Respon sistem cukup cepat	11	8	1	0	0
9	Sistem membantu menentukan prioritas laporan	6	12	2	0	0
10	Sistem nyaman digunakan	10	10	0	0	0

Sumber: Hasil Penelitian (2026)

Sebagai contoh, perhitungan skor pada pernyataan pertama adalah sebagai berikut.

$$\text{Skor} = (14 \times 5) + (6 \times 4) + (0 \times 3) + (0 \times 2) + (0 \times 1) = 94$$

Perhitungan yang sama dilakukan pada seluruh 10 pernyataan sehingga diperoleh total skor sebesar:

$$94 + 85 + 86 + 87 + 84 + 87 + 85 + 90 + 84 + 90 = 872$$

Skor maksimum dihitung menggunakan rumus:

$$\text{Skor Maksimum} = \text{Jumlah Responden} \times \text{Jumlah Pernyataan} \times \text{Skor Tertinggi}$$

$$20 \times 10 \times 5 = 1000$$

Selanjutnya, persentase penilaian dihitung menggunakan rumus:

$$\text{Persentase} = (872 / 1000) \times 100\% = 87,2\%$$

Tabel 3. UAT Aspek Tampilan Sistem (UI)

No	TAMPILAN SISTEM (UI)	PENILAIAN				
		SS	S	N	TS	STS
1	Tampilan sistem terlihat menarik	9	11	0	0	0
2	Warna sistem nyaman dilihat	6	12	2	0	0
3	Tulisan pada sistem mudah dibaca	8	10	2	0	0
4	Tata letak menu terlihat rapi	2	7	11	0	0
5	Navigasi sistem mudah digunakan	6	11	3	0	0
6	Informasi pada sistem mudah dipahami	11	8	1	0	0
7	Tombol dan fitur mudah ditemukan	4	14	2	0	0
8	Tampilan hasil klasifikasi jelas	4	12	4	0	0
9	Desain sistem terlihat konsisten	7	11	2	0	0
10	Tampilan sistem membuat pengguna nyaman	4	12	4	0	0

Sumber: Hasil Penelitian (2026)

Perhitungan pada aspek tampilan sistem dilakukan menggunakan metode yang sama dengan aspek sebelumnya, yaitu menjumlahkan seluruh skor hasil perkalian bobot skala *Likert* pada setiap pernyataan.

Total skor yang diperoleh adalah:

$$89 + 84 + 86 + 71 + 83 + 90 + 82 + 80 + 85 + 80 = 830$$

Skor maksimum dihitung sebagai berikut:

$$20 \times 10 \times 5 = 1000$$

Persentase penilaian dihitung menggunakan rumus:

$$\text{Persentase} = (830 / 1000) \times 100\% = 83,0\%$$

Tabel 4. UAT Aspek Keandalan Sistem

No	KEANDALAN SISTEM	PENILAIAN				
		SS	S	N	T S	S T S
1	Sistem dapat digunakan tanpa mengalami <i>crash</i>	11	8	1	0	0
2	Sistem tetap berjalan stabil saat digunakan	8	12	0	0	0
3	Proses <i>login</i> berjalan dengan baik	8	10	2	0	0
4	Data laporan tersimpan dengan benar	11	7	2	0	0
5	Sistem dapat menampilkan data laporan dengan lengkap	8	11	1	0	0
6	Sistem dapat memproses laporan tanpa <i>error</i>	6	12	2	0	0
7	Perpindahan halaman pada sistem berjalan lancar	9	8	3	0	0

No	KEANDALAN SISTEM	PENILAIAN				
		SS	S	N	T S	S T S
8	Sistem mampu menampilkan hasil proses dengan baik	9	8	3	0	0
9	Sistem memberikan respon yang konsisten	5	12	3	0	0
10	Sistem dapat digunakan sesuai fungsi yang diharapkan	13	6	1	0	0

Sumber: Hasil Penelitian (2026)

Perhitungan pada aspek keandalan sistem dilakukan dengan cara yang sama, yaitu menjumlahkan seluruh skor hasil perkalian bobot jawaban responden pada setiap pernyataan.

Total skor yang diperoleh adalah:

$$90 + 88 + 86 + 89 + 87 + 84 + 86 + 86 + 82 + 92 = 870$$

Skor maksimum dihitung menggunakan rumus:

$$20 \times 10 \times 5 = 1000$$

Persentase penilaian diperoleh sebagai berikut:

$$\text{Persentase} = (870 / 1000) \times 100\% = 87,0\%$$

Nilai keseluruhan *User Acceptance Testing (UAT)* diperoleh dari akumulasi skor pada ketiga aspek pengujian, yaitu kemudahan penggunaan, tampilan sistem, dan keandalan sistem.

$$\text{Total Skor} = 872 + 830 + 870 = 2572$$

Skor ideal dihitung sebagai berikut:

$$\text{Skor Ideal} = \text{Jumlah Responden} \times \text{Jumlah Pernyataan} \times \text{Skor Tertinggi}$$

$$20 \times 30 \times 5 = 3000$$

Selanjutnya, persentase tingkat penerimaan pengguna dihitung menggunakan rumus:

$$Y = (2572 / 3000) \times 100\% = 85,73\%$$

Berdasarkan hasil pengujian, aspek kemudahan penggunaan memperoleh nilai 87,20%, yang menunjukkan bahwa pengguna dapat mengoperasikan sistem dengan mudah, mulai dari proses input laporan hingga memperoleh hasil klasifikasi prioritas. Aspek tampilan sistem memperoleh nilai 83,00%,

yang mengindikasikan bahwa antarmuka aplikasi dinilai cukup jelas, mudah dipahami, dan mendukung aktivitas pengguna, meskipun masih terdapat peluang untuk meningkatkan desain visual maupun penyajian informasi agar pengalaman pengguna menjadi lebih baik.

Sementara itu, aspek keandalan sistem memperoleh nilai 87,00%, yang menunjukkan bahwa sistem mampu menjalankan fungsi-fungsi utama secara stabil, termasuk proses *login*, penyimpanan data, klasifikasi prioritas, dan penampilan hasil tanpa mengalami kendala yang berarti selama pengujian. Dengan nilai rata-rata keseluruhan sebesar 85,73%, sistem termasuk dalam kategori sangat layak, sehingga dapat dinyatakan telah memenuhi kebutuhan pengguna sebagai media pengelolaan dan klasifikasi prioritas permasalahan operasional [16].

5. KESIMPULAN

- a. Sistem berbasis *web* yang dikembangkan mampu membantu proses identifikasi laporan permasalahan operasional berdasarkan data yang diinput pengguna, meliputi judul, deskripsi, lokasi, dampak, dan divisi terkait. Proses tersebut mendukung pengelolaan laporan secara lebih terstruktur dibandingkan proses manual.
- b. Penerapan algoritma *Naïve Bayes* yang didukung tahapan *preprocessing* teks dan pembobotan *TF-IDF* berhasil mengklasifikasikan laporan ke dalam kategori *urgent*, *middle*, dan *low*. Berdasarkan hasil *Black Box Testing*, seluruh fungsi sistem berjalan sesuai kebutuhan, sedangkan hasil *User Acceptance Testing (UAT)* memperoleh nilai 85,73%, sehingga sistem termasuk dalam kategori sangat layak untuk digunakan.
- c. Sistem mampu memberikan rekomendasi tindakan berdasarkan hasil klasifikasi serta menyediakan fitur pengelolaan laporan, riwayat pelaporan, dan pemantauan status penanganan. Fitur-fitur tersebut diharapkan dapat membantu perusahaan dalam menentukan prioritas penanganan permasalahan operasional secara lebih konsisten dan mendukung proses pengambilan keputusan.

UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada Universitas Bina Insani atas dukungan selama pelaksanaan penelitian. Ucapan terima kasih juga disampaikan kepada PT Vuteq Indonesia yang telah memberikan kesempatan, dukungan, dan fasilitas selama proses penelitian. Penulis juga menyampaikan apresiasi kepada dosen pembimbing serta seluruh pihak yang telah memberikan arahan, masukan, dan bantuan sehingga penelitian ini dapat diselesaikan dengan baik.

DAFTAR PUSTAKA

- [1] H. Chen, S. Hu, R. Hua, and X. Zhao, "Improved naive Bayes classification algorithm for traffic risk management," vol. 6, 2021.
- [2] S. Gan, S. Shao, L. Chen, L. Yu, and L. Jiang, "Adapting Hidden Naive Bayes for Text Classification," pp. 1–14, 2021.
- [3] D. Liu, "Improvement of Naive Bayes Text Classifier Based on Ensemble Technology and Feature Engineering," no. Iciaai, 2023.
- [4] C. Naulak and R. Jindal, "A comparative study of Naive Bayes Classifiers with improved technique on Text Classification," pp. 0–8, 2022, doi: 10.36227/techrxiv.19918360.v1.
- [5] A. F. E. Naïve and J. B. Barbosa, "Classification Using Naïve Bayes," pp. 9–18, 2022.
- [6] D. P. Hofmeyr, F. Kamper, and M. C. Melonas, "Optimal Projections for Classification with Naïve Bayes," pp. 1–28, 2025.
- [7] G. Waja, J. Shah, and P. Nanavati, "Agile software development," vol. 5, no. 12, pp. 73–78, 2021.
- [8] J. G. R. Ibarra, G. Borrego, and R. R. Palacio, "Early Estimation in Agile Software Development Projects : A Systematic Mapping Study," pp. 1–25, 2024.
- [9] X. Wang, M. Marchesi, and P. G. Eds, *Agile Processes in Software*. 2024.
- [10] D. Rifandi, I. Ratna, I. Astutik, N. Ariyanti, and A. Eviyanti, "Design of a Web-Based Workshop Reservation System for Sumber Rejeki Jaya Motor Using the Laravel Framework with the Agile Development Method [Rancangan Sistem Reservasi Bengkel Sumber Rejeki Jaya Motor Berbasis Web Menggunakan Framework Laravel dengan Me," pp. 1–19, 2025.
- [11] H. K. Saputra, A. Hadi, and D. Novaliendry, "Penerapan User Acceptance Testing (UAT) pada Pengujian Sistem Informasi Manajemen Operasional Studio Foto Berbasis Client-Server 1*," vol. 5, no. 2, pp. 2396–2410, 2026.

- [12] R. Setiawan, B. Priyatna, E. Novalia, and B. Huda, "Klasifikasi Tingkat Penjualan Produk pada Toko Jati Karebet Menggunakan Algoritma Naïve Bayes," vol. 15, no. 2, pp. 297–303, 2025.
- [13] N. B. Putri and A. W. Wijayanto, "Analisis Komparasi Algoritma Klasifikasi Data Mining Dalam Klasifikasi Website Phishing Comparative Analysis Of Data Mining Classification Algorithm In Phishing Website Classification," vol. 11, no. 28, pp. 59–66, 2022, doi: 10.34010/komputika.v11i1.4350.
- [14] J. David, W. Panjaitan, A. Muhammad, and A. Suprianto, "Journal of Computer Networks , Architecture and High Performance Computing Web-Based Attendance Information System At Diskominfoantik Bekasi District With Prototype Method Journal of Computer Networks , Architecture and High Performance Computing," vol. 7, no. 3, pp. 1021–1031, 2025.
- [15] B. Simamora, "Skala Likert , Bias Penggunaan dan Jalan," vol. 12, no. 1, pp. 84–93, 2022.
- [16] N. Safriadi and A. Perwitasari, "ADAPTASI FLEKSIBILITAS SISTEM INFORMASI MANAJEMEN RUMAH SUSUN SEDERHANA SEWA," vol. 11, no. 3, 2023.