

# Penerapan Algoritma *Naive Bayes* pada Arsip Surat Kantor Kecamatan Motoling Barat

<sup>1</sup>Frendy Rikal Gerung, <sup>2</sup>Glenn D. P. Maramis, <sup>3</sup>Efraim R. S. Moningkey

<sup>1, 2, 3</sup>Universitas Negeri Manado  
Manado, Indonesia

<sup>1</sup>20210118@unima.ac.id, <sup>2</sup>gmaramis@unima.ac.id, <sup>3</sup>efraimmoningkey@unima.ac.id

\*Penulis Korespondensi

Diajukan : 16/05/2025

Diterima : 19/05/2025

Dipublikasi : 23/05/2025

## ABSTRAK

Transformasi digital dalam pengelolaan arsip menjadi kebutuhan di instansi pemerintahan, termasuk Kantor Kecamatan Motoling Barat, akibat keterbatasan sistem manual yang berisiko terhadap efisiensi, akurasi, dan kepatuhan terhadap regulasi. Tujuan penelitian ini adalah untuk membuat sistem klasifikasi surat masuk yang menggunakan teknik *text mining* yang berbasis pada algoritma *Naive Bayes* untuk otomatisasi proses pengarsipan. Metodologi pengembangan sistem menggunakan kerangka kerja *Extreme Programming (XP)*, dengan penerapan teknologi *React JS* untuk antarmuka pengguna, *Django* sebagai *backend*, dan *MySQL* untuk manajemen basis data. Model klasifikasi dilatih dengan 219 data surat masuk yang telah melalui tahap *preprocessing* dan tokenisasi menggunakan *TF-IDF*. Dari rangkaian pengujian yang dilakukan, model berhasil mencatat akurasi sebesar 74,4%, meski terlihat fluktuasi nilai *precision* dan *recall* pada tiap kategori surat. Kategori 'Surat Keputusan' menunjukkan *precision* tinggi namun *recall* rendah, sedangkan 'Surat Edaran' memiliki *precision* terendah. Selain itu, pengujian sistem secara keseluruhan melalui metode *black-box* membuktikan bahwa seluruh fitur utama seperti *login*, manajemen surat, klasifikasi otomatis, dan pengelolaan pengguna berfungsi dengan baik. Sistem yang dikembangkan dinilai responsif, mudah digunakan, dan mampu meningkatkan efisiensi pengelolaan arsip. Temuan ini menunjukkan bahwa penerapan algoritma *Naive Bayes* dalam sistem informasi arsip digital memiliki potensi besar dalam mendukung reformasi birokrasi dan transformasi digital di sektor publik. Ke depan, pengembangan lebih lanjut disarankan dengan perluasan dataset dan eksplorasi metode klasifikasi lain seperti *SVM* atau *ensemble learning*.

**Kata Kunci:** *Algoritma Naive Bayes; Extreme Programming; Klasifikasi Dokumen; Sistem Arsip Digital; Text Mining*

## I. PENDAHULUAN

Berbagai sektor telah berubah menjadi digital karena kemajuan teknologi informasi. Ini termasuk mengubah cara pemerintah menyimpan surat. Sistem pengarsipan manual yang masih digunakan di banyak kantor, seperti di Kantor Kecamatan Motoling Barat, menghadapi tantangan seperti inefisiensi pencarian dokumen, risiko kehilangan arsip, dan ketidaksesuaian dengan regulasi pemerintah yang mewajibkan digitalisasi (Lolytasari and Dirsanala 2023; Triana et al. 2024). Sistem pengarsipan elektronik sangat penting untuk meningkatkan transparansi dan akuntabilitas sesuai dengan Peraturan Presiden No. 95 Tahun 2018 dan Peraturan ANRI No. 6 Tahun 2021 (Arsip Nasional Republik Indonesia 2021; Pemerintah Pusat 2018).

Penelitian ini mengusulkan solusi berbasis *text mining* dengan algoritma *Naive Bayes* untuk mengotomatisasi klasifikasi surat masuk. Algoritma *Naive Bayes* ini dipilih karena kemampuannya dalam pengolahan teks dan efisiensi komputasi (Dewi 2021; Farhan, Triase, and Harahap 2023; M. Afriansyah et al. 2024).

Pengembangan sistem ini menggunakan kerangka kerja *Extreme Programming (XP)* untuk memastikan fleksibilitas dan adaptabilitas selama proses pengembangan (Hijriani et al. 2020). *XP* dipilih karena kemampuannya dalam mengakomodasi perubahan kebutuhan pengguna secara iteratif (Szabó and Hercegfi 2023). Dari sisi teknis, sistem ini dibangun dengan *React JS* sebagai *frontend* dan *Django* sebagai *backend*, mengikuti tren pengembangan aplikasi web modern yang efisien (Nurhasanah and Rusdan 2020).

Selain aspek teknis, penelitian ini juga mempertimbangkan dampak sosial dan regulasi. Penerapan sistem pengarsipan digital telah terbukti meningkatkan efisiensi administratif (Fauzan, Putri, and Endiyanti 2024). Selain itu, integrasi sistem ini dengan kebijakan pemerintah, seperti aplikasi Srikandi (Sistem Informasi Kearsipan Dinamis Terintegrasi), memperkuat kepatuhan terhadap standar kearsipan nasional (Ramudin, Ramli, and Rosmana 2024).

Akibatnya, penelitian ini tidak hanya membantu Kantor Kecamatan Motoling Barat dengan solusi praktis, tetapi juga membantu mengembangkan ilmu pengetahuan di bidang *text mining* dan manajemen sistem informasi. Hasil pengujian menunjukkan akurasi klasifikasi sebesar 74,4%; ada ruang untuk peningkatan melalui teknik *preprocessing* dan pengoptimalan data pelatihan (Hasibuan and Aziz 2022; Maramis 2023; Resti et al. 2024). Di masa mendatang, sistem ini bisa ditingkatkan lagi dengan menyertakan metode *machine learning* tambahan, misalnya *Support Vector Machine (SVM)* atau *Random Forest* agar kemampuan klasifikasinya semakin optimal. (Prayogo, Fauziah, and Winarsih 2023).

## II. STUDI LITERATUR

Penelitian Zhang et al. (2024) merancang sistem klasifikasi otomatis dokumen reformasi pendidikan berbasis *Naive Bayes* dengan antarmuka yang responsif dan *pipeline preprocessing text* yang komprehensif. Sistem ini diuji pada korpus dokumen resmi dan mencapai tingkat akurasi rata-rata di atas 78%, sekaligus mengurangi beban klasifikasi manual bagi tenaga ahli Pendidikan

Dalam studi Aubaid et al. (2024), penulis membandingkan metode *Naive Bayes* tradisional dengan pendekatan hibrida *W2vRule* pada arsip elektronik skala besar. Hasilnya, *Naive Bayes* menunjukkan performa yang stabil dengan akurasi sekitar 75%, meski sedikit tertinggal dibanding *W2vRule*, namun tetap unggul dalam hal kecepatan pelatihan dan konsumsi memori

Luo. (2021) meneliti penerapan *Naive Bayes* untuk klasifikasi dokumen ilmiah berbahasa Inggris. Dengan teknik seleksi fitur berbasis  $\chi^2$  dan *TF-IDF*, model *Naive Bayes* mencapai akurasi hingga 82% pada dataset jurnal penelitian, menegaskan efektivitas algoritma ini dalam domain korpus formal

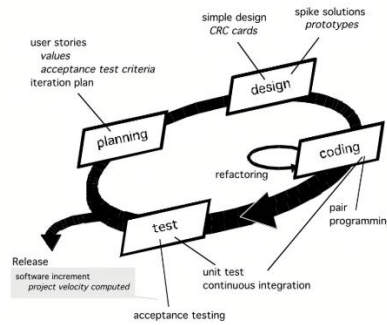
Pada konteks berita daring, studi Kashid et al. (2025) menguji *Naive Bayes* untuk mengategorikan artikel *real-time* ke dalam topik olahraga, politik, dan teknologi. Model ini berhasil mengklasifikasikan lebih dari 5.000 artikel dengan akurasi rata-rata 80%, menunjukkan potensi *Naive Bayes* dalam pengolahan *data streaming*

Sementara itu, Taha et al. (2024) membandingkan berbagai teknik klasifikasi teks termasuk *Naive Bayes*, *SVM*, dan *ensemble learning*. *Naive Bayes* disorot karena kesederhanaannya dan kemampuannya memberikan probabilitas keluaran yang berguna untuk sistem hibrida, meski performanya lebih rendah pada korpus dengan distribusi kata yang sangat tidak seimbang

Kelima penelitian ini membuktikan bahwa algoritma *Naive Bayes* tetap relevan dan efisien untuk klasifikasi teks, khususnya dalam aplikasi pengarsipan dan pengelolaan dokumen digital; sekaligus menunjukkan peluang peningkatan melalui pendekatan hibrida atau optimasi *preprocessing*.

## III. METODE

Penelitian ini menggunakan pendekatan pengembangan perangkat lunak *Extreme Programming (XP)* sebagai metodologi utama. *XP* dipilih karena kemampuannya dalam menangani perubahan kebutuhan secara adaptif, meningkatkan kualitas sistem melalui iterasi singkat, dan memperkuat komunikasi antara pengembang dan pengguna (Pramudya et al. 2025). Proses Metode *Extreme Programming* bisa dilihat pada Gambar 1.

Gambar 1. Tahapan Metode *Extreme Programming*

### Tahap *Planning* (Perencanaan)

Perencanaan diawali dengan proses mendengarkan atau melakukan wawancara untuk menggali hal-hal yang dibutuhkan oleh pelanggan. Namun, kebutuhan yang diperoleh dari hasil wawancara tersebut bersifat dinamis, sehingga dapat berubah sewaktu-waktu. Setelah itu, pelanggan akan menetapkan kebutuhan mana yang menjadi prioritas. Berdasarkan prioritas tersebut, pengembang kemudian menentukan jadwal dan biaya pengembangan.

### Tahap *Design* (Perancangan)

Temuan analisis kebutuhan yang diperoleh pada fase perencanaan berperan sebagai panduan utama dalam merumuskan setiap langkah perancangan penelitian ini. Pada tahap ini, tujuan utama adalah membangun pemodelan proses bisnis dan struktur data yang terorganisir dengan menggunakan bahasa pemodelan yang terpadu *UML (Unified Modeling Language)*. Metode ini digunakan untuk menjelaskan jalan untuk mengembangkan sistem yang sesuai dengan kebutuhan pengguna.

Dalam proses perancangan, digunakan beberapa jenis diagram *UML* untuk merepresentasikan sistem secara visual. *Activity diagram* menunjukkan alur proses untuk fitur utama sistem, seperti *login*, pengelolaan surat masuk dan keluar, klasifikasi otomatis, dan manajemen pengguna. *Use case diagram* menunjukkan hubungan antara sistem dan aktor (pengguna).

Selain pemodelan proses, dilakukan juga perancangan basis data yang mencakup struktur tabel-tabel penyimpanan untuk arsip surat masuk, surat keluar, dan data pengguna. Dengan menggunakan framework *Django* sebagai *backend* dan sistem manajemen basis data *MySQL*, basis data ini dirancang untuk mendukung integrasi sistem. Dengan pendekatan ini, diharapkan desain sistem yang dihasilkan bersifat terstruktur, logis, dan mudah diimplementasikan sesuai kebutuhan pengelolaan arsip di Kantor Kecamatan Motoling Barat.

### Tahap *Coding* (Pengkodean)

Tahap ini melibatkan implementasi sistem berdasarkan rancangan yang telah dibuat. Pengembangan antarmuka dilakukan dengan *React JS*, sedangkan *Django* digunakan untuk mengelola logika bisnis dan integrasi dengan basis data (Gupta 2021; Moningkey, Triyono, and Priyanto 2022; Shah 2021). Model klasifikasi surat masuk dibangun menggunakan algoritma *Multinomial Naïve Bayes* yang diimplementasikan dengan pustaka *Scikit-learn* pada bahasa pemrograman *Python*.

### Tahap *Testing* (Pengujian)

Pengujian sistem dilakukan untuk mengevaluasi keakuratan model klasifikasi serta memastikan seluruh fitur dalam sistem beroperasi selaras dengan tuntutan pengguna. Proses pengujian mencakup evaluasi terhadap algoritma klasifikasi yang digunakan serta pengujian fungsionalitas sistem secara keseluruhan.

## IV. HASIL DAN PEMBAHASAN

### Tahapan *Planning* (Perencanaan)

Proses perencanaan menjadi fondasi utama dalam pengembangan sistem pengelolaan arsip surat yang modern dan terintegrasi. Melalui observasi, wawancara, dan studi mendalam, dilakukan identifikasi masalah, analisis kebutuhan, serta penentuan fitur dan teknologi yang digunakan. Tujuannya adalah menghadirkan solusi digital yang efisien dan sesuai regulasi untuk meningkatkan pelayanan di Kantor Kecamatan Motoling Barat.

Kerentanan tinggi pada prosedur konvensional, mulai dari potensi *human error* hingga risiko hilangnya informasi, ditambah dengan kesulitan dalam memantau jejak data dan ketidakpatuhan terhadap regulasi yang menuntut sistem digital, menjadi kendala utama. Proses kerja yang panjang dan berulang menyebabkan inefisiensi, sementara penggunaan kertas berlebihan mengakibatkan pemborosan biaya dan dampak negatif terhadap lingkungan.

Fitur utama yang dikembangkan mencakup klasifikasi surat otomatis menggunakan algoritma *Naïve Bayes*, penyimpanan dan pengelolaan arsip digital, serta pencarian cepat dengan akses terbatas berdasarkan peran pengguna. Fitur-fitur ini dimaksudkan untuk meningkatkan produktivitas dan keamanan arsip surat.

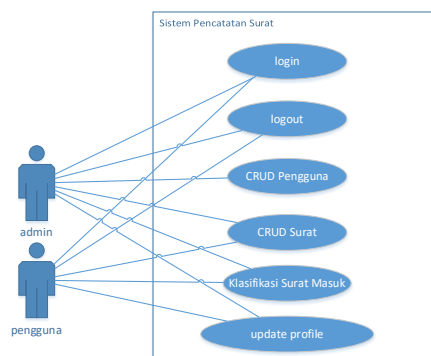
Teknologi yang digunakan meliputi algoritma *Naïve Bayes* untuk klasifikasi surat, *React JS* sebagai *frontend*, *Django* untuk *backend*, dan *MySQL* sebagai basis data. *Visual Studio Code* dipilih sebagai alat pengembangan utama. Untuk mendukung proses ini, digunakan *Windows 10* sebagai sistem operasi, browser *Chrome* untuk pengujian, serta perangkat keras seperti laptop dan perlengkapan pendukung lainnya guna memastikan kelancaran pengembangan sistem.

### Tahan Design (Perancangan)

Perancangan komponen sistem penting, seperti *Usecase Diagram*, *Activity Diagram*, dan *Database Notation*, dilakukan pada tahap ini.

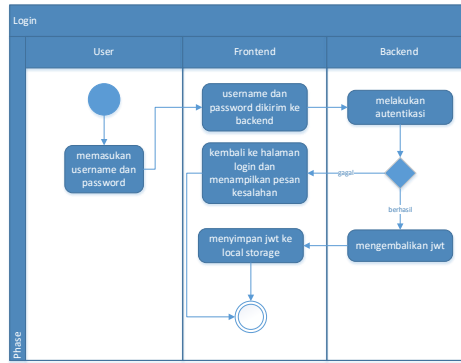
Dua aktor utama dalam sistem ini adalah admin dan pengguna. Admin bertanggung jawab untuk mengelola sistem, seperti mengelola pengguna, mengatur akses, dan mengawasi aktivitas. Pengguna adalah staf yang menggunakan sistem untuk keperluan pengarsipan dan pencarian surat. Dalam sistem ini, tiap aktor diberikan tingkat izin yang berlainan, disesuaikan dengan peran spesifik yang mereka emban.

Fitur utama dalam sistem meliputi: *Login & Logout* untuk autentikasi pengguna, *CRUD Pengguna & Surat* untuk pengelolaan data surat masuk dan keluar, *Klasifikasi Jenis Surat* guna mengategorikan surat secara otomatis, serta *Update Profil* yang memungkinkan pengguna memperbarui informasi mereka. Struktur *usecase* terlihat seperti pada Gambar 2.



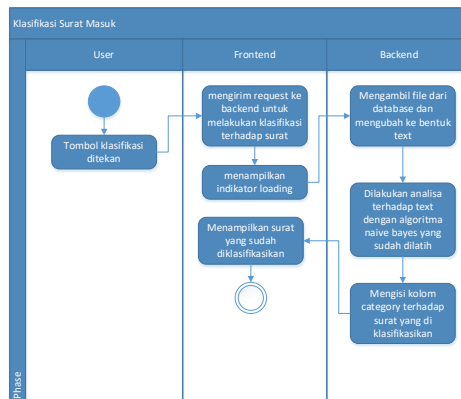
Gambar 2. Use Case Diagram Sistem Pencatatan Surat Digital

Pada Gambar 3 terlihat diagram autentikasi pengguna. Proses diawali ketika pengguna memasukkan *username* dan *password*. *Frontend* kemudian mengirimkan kredensial tersebut ke *backend* untuk diverifikasi. Jika valid, *backend* menghasilkan *JWT (JSON Web Token)* yang disimpan di *local storage frontend* untuk otorisasi permintaan selanjutnya. Jika gagal, pengguna diarahkan kembali ke halaman *login* dengan pesan kesalahan.



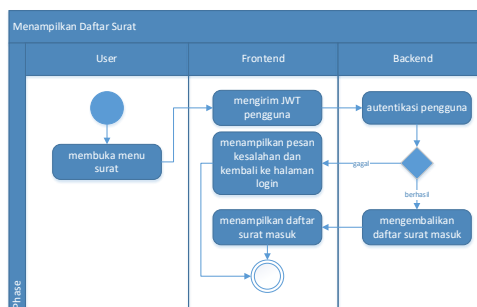
Gambar 3. Activity Diagram Login

Pada Gambar 4 terlihat diagram klasifikasi surat. Alur dimulai saat pengguna menekan tombol klasifikasi. Frontend mengirim permintaan ke *backend* sambil menampilkan indikator loading. *Backend* memprosesnya dengan mengambil data surat, mengubahnya ke bentuk teks, lalu menganalisis menggunakan algoritma *Naive Bayes* yang telah dilatih. Hasil klasifikasi kemudian dikembalikan ke *frontend* untuk ditampilkan kepada pengguna.



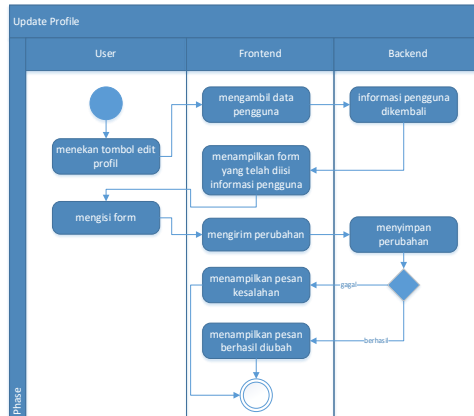
Gambar 4. Activity Diagram Klasifikasi Surat Masuk

Pada Gambar 5 terlihat diagram yang menunjukkan alur kerja saat pengguna ingin melihat daftar surat masuk. Pengguna membuka menu surat, *frontend* mengirimkan *JWT* ke *backend* untuk autentikasi. Jika gagal, ditampilkan pesan kesalahan. Jika berhasil, *backend* mengirimkan daftar surat yang kemudian ditampilkan oleh *frontend*.



Gambar 5. Activity Diagram Menampilkan Daftar Surat

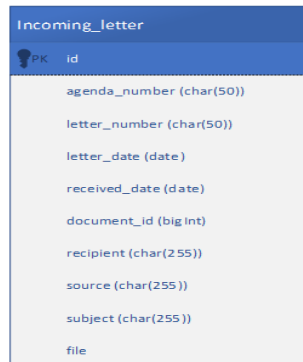
Pada Gambar 6 terlihat diagram pembaruan profil. *Frontend* terlebih dahulu mengambil data profil pengguna dari *backend* dan menampilkannya dalam formulir. Setelah pengguna mengedit informasi, perubahan dikirim ke *backend*. Jika berhasil, data diperbarui dan *frontend* menampilkan konfirmasi sukses. Jika gagal, sistem memberikan notifikasi *error*.



Gambar 6. Activity Diagram Update Profil

Operasi lain seperti *logout* mengikuti pola sederhana: *frontend* menghapus *JWT* dari penyimpanan lokal dan mengarahkan pengguna ke halaman *login*. Demikian pula operasi CRUD lainnya (tambah, edit, hapus) menerapkan logika serupa dengan penyesuaian sesuai kebutuhan fungsionalitas.

Desain database yang diberi nama *incoming\_letter* untuk surat masuk ditampilkan pada Gambar 7 yang mencakup berbagai kolom penting. Kolom *id* berfungsi sebagai *primary key* untuk mengidentifikasi setiap surat secara unik, sementara *agenda\_number* dan *letter\_number* mencatat nomor agenda dan nomor surat. Terdapat juga kolom *letter\_date* dan *received\_date* untuk menandai tanggal surat dibuat dan tanggal diterima, serta *recipient* dan *source* yang menunjukkan penerima dan pengirim surat. Kolom *subject* digunakan untuk mencatat perihal surat, dan file menyimpan *path* file surat tersebut.



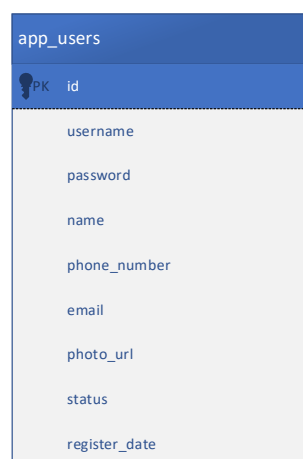
Gambar 7. Database Notation Surat Masuk

Untuk surat keluar, desain tabelnya terlihat pada Gambar 8 dan tabel diberi nama "*outgoing\_letter*" dengan struktur yang mirip namun disesuaikan untuk kebutuhan pengiriman. Kolom *id* tetap berperan sebagai *primary key*, diikuti oleh *agenda\_number* dan *letter\_number* sebagai pencatat nomor penting. *Letter\_date* menandakan tanggal surat dibuat, sementara *destination* menunjukkan kepada siapa surat ditujukan. *Subject* menjelaskan perihal surat, dan *file* sebagai *path* dokumen digitalnya.



Gambar 8. Database Notation Surat Keluar

Sementara itu, tabel pengguna diberi nama "*app\_users*" dan dirancang untuk mengelola data pengguna aplikasi yang terlihat pada Gambar 9. Kolom *id* menjadi *primary key*, sedangkan *username* dan *password* digunakan untuk proses *login* dengan kata sandi yang dienkripsi. Atribut *name* mencatat nama lengkap pengguna, *phone\_number* dan *email* menyimpan informasi kontak, serta *photo\_url* menyimpan tautan foto profil. Status menunjukkan apakah pengguna aktif atau tidak, dan *register\_date* mencatat kapan pengguna mendaftar.



Gambar 9. Database Notation Pengguna

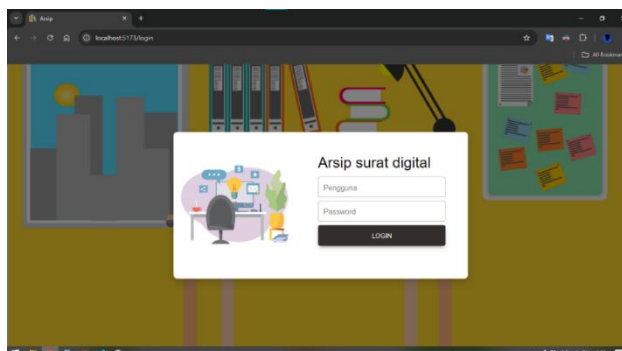
### Tahapan Coding (Pengkodean)

Tahap pengkodean adalah fase krusial di mana konsep dan desain diwujudkan menjadi sistem nyata dan fungsional. Di fase ini, seluruh komponen rancangan mulai dari *use case diagram* dan *activity diagram* hingga skema basis data diubah dan diimplementasikan langsung ke dalam kode pemrograman. Penelitian ini menggunakan *React JS* untuk *frontend* yang dinamis dan responsif, serta *Django* untuk *backend* yang menangani logika aplikasi, manajemen *database*, dan keamanan. Hasilnya adalah sistem pencatatan surat digital siap uji, mendukung transformasi digital Kantor Camat Motoling Barat.

Aplikasi pengarsipan surat di Kantor Kecamatan Motoling Barat ini terdiri dari beberapa halaman yang dirancang untuk memudahkan pengelolaan dokumen. Setiap halaman memiliki fungsi spesifik guna mendukung proses klasifikasi dan penyimpanan surat secara efisien.

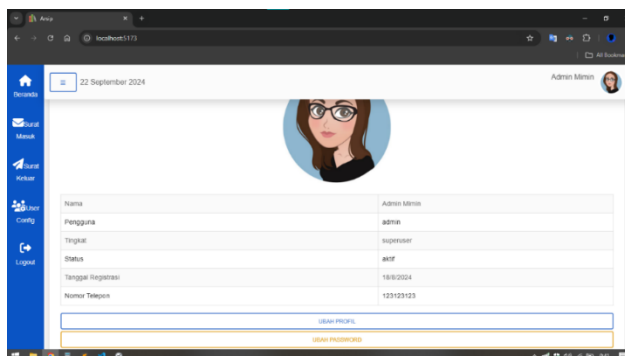
Sistem arsip surat digital diawali dengan halaman login yang didesain sederhana namun fungsional. Pengguna akan menemukan dua field input utama untuk *username* dan *password*, dilengkapi tombol *LOGIN* berwarna mencolok untuk memulai akses. Latar belakang dengan

ilustrasi abstrak memberikan kesan profesional sekaligus modern seperti yang terlihat pada Gambar 10. Halaman ini berfungsi sebagai gerbang keamanan pertama yang memastikan hanya personel terotorisasi yang dapat mengakses sistem arsip digital.



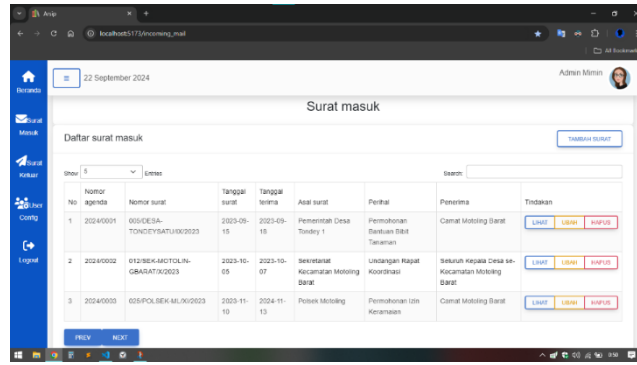
Gambar 10. Halaman *Login*

Setelah berhasil masuk, pengguna akan disambut oleh *dashboard* personal yang menampilkan informasi profil lengkap. Bagian ini memperlihatkan foto profil, nama lengkap, *username*, level akses, status keaktifan akun, tanggal registrasi, dan kontak telepon. Dua tombol aksi utama tersedia di sini: 'Ubah Profil' untuk memperbarui data pribadi dan 'Ubah *Password*' untuk mengatur ulang kredensial login. Tampilan *dashboard* yang terorganisir ini memudahkan pengguna memahami informasi akun mereka sekilas. Gambar 11 menampilkan visualisasi antarmuka *dashboard*.



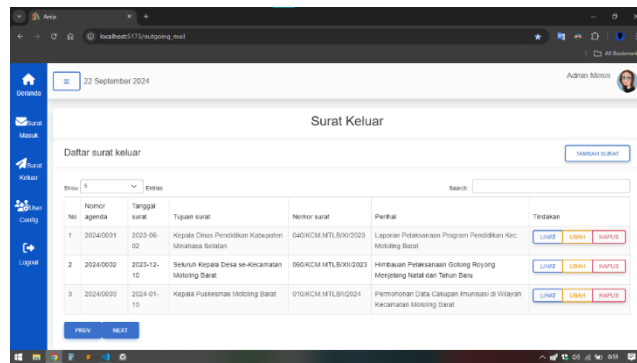
Gambar 11. Halaman Antarmuka *Dashboard*

Pada Gambar 12 terlihat daftar surat masuk yang menyajikan tabel komprehensif berisi semua dokumen yang diterima. Setiap entri mencatat detail penting mulai dari nomor agenda, nomor surat, tanggal surat, hingga asal dan perihal surat. Kolom aksi menyediakan tiga opsi: *preview* untuk melihat detail, *edit* untuk modifikasi data, dan *delete* untuk menghapus entri. Fitur unggulan meliputi tombol '+' *Tambah Surat* untuk input dokumen baru dan *search bar* untuk pencarian cepat melalui seluruh arsip.



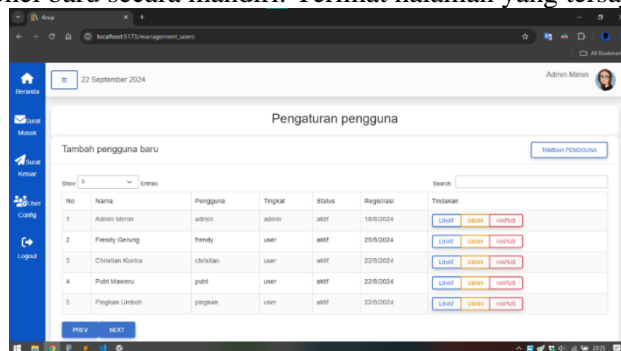
Gambar 12. Halaman untuk Daftar Surat Masuk

Layaknya tampilan pada surat masuk, modul surat keluar di Gambar 13 menyuguhkan seluruh kumpulan dokumen yang telah dikirim. Informasi kunci seperti nomor agenda, tanggal kirim, tujuan, dan ringkasan isi tersaji rapi dalam format tabel. Desain antarmuka konsisten dengan modul lain, mempertahankan tombol aksi dan fitur pencarian. Penambahan dokumen baru dapat dilakukan melalui tombol 'TAMBAH SURAT' yang terletak di bagian atas tabel.



Gambar 13. Halaman Daftar Surat Keluar

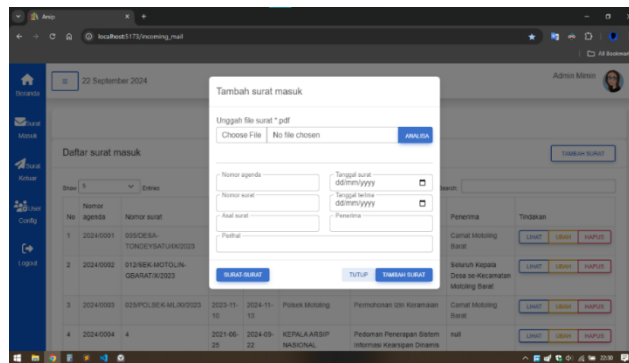
Halaman administrasi ini diperuntukkan bagi admin sistem untuk mengelola semua akun pengguna. Tabel berisi data penting seperti nama lengkap, *username*, *role* (admin/user), status, dan tanggal registrasi. Hak akses khusus memungkinkan admin untuk melihat detail, mengedit, atau menghapus akun melalui ikon-ikon aksi. Tombol 'Tambah Pengguna' yang mencolok memudahkan proses registrasi personel baru secara mandiri. Terlihat halaman yang tersaji pada Gambar 14.



Gambar 14. Halaman Pengaturan Pengguna

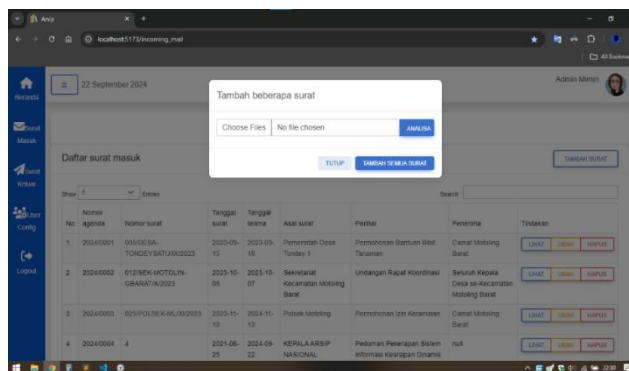
Pada Gambar 15 terlihat antarmuka unggah dokumen tunggal menawarkan proses *input* yang terstruktur. Pengguna dapat *upload file* melalui *drag-and-drop* atau browser file, kemudian mengekstrak data otomatis dengan tombol ANALISA. Formulir isian muncul setelah ekstraksi, memungkinkan verifikasi dan penyempurnaan data sebelum disimpan melalui tombol 'TAMBAH SURAT'. Opsi navigasi fleksibel disediakan melalui tombol 'SURAT-SURAT' (untuk

beralih ke mode *batch*) dan 'TUTUP' (untuk keluar dari form).



Gambar 15. Halaman Tambah Surat Masuk

Pada Gambar 16 terlihat fitur khusus untuk kebutuhan masal. Sistem menyediakan mode unggah multi-dokumen sekaligus. Pengguna dapat memilih beberapa file kemudian melakukan analisa otomatis untuk semua dokumen. Proses finalisasi dilakukan melalui tombol 'TAMBAH SEMUA SURAT' setelah verifikasi data. Desain *pop-up* yang digunakan mempertahankan konsistensi visual dengan tetap memungkinkan akses cepat ke fungsi penutupan melalui tombol 'TUTUP'.



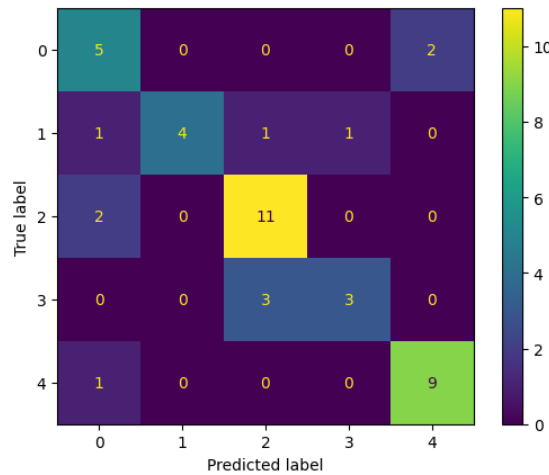
Gambar 16. Halaman Unggah Dokumen Masal

**Tahap Testing (Pengujian)**

Pada pengujian ini, dilakukan dua jenis uji: pertama, uji model klasifikasi surat masuk, dan kedua, uji semua fitur sistem menggunakan metode *blackbox*.

Penelitian ini menguji model klasifikasi untuk mengkategorikan surat masuk secara otomatis ke dalam lima kategori: Surat Edaran, Surat Keputusan, Surat Permohonan, Surat Tugas, dan Surat Undangan. Kelima kategori dipetakan ke label numerik untuk mempermudah klasifikasi menggunakan algoritma *Naive Bayes*. Data diperoleh dari Kantor Camat Motoling Barat, terdiri dari 219 surat yang terbagi ke dalam lima kategori tersebut.

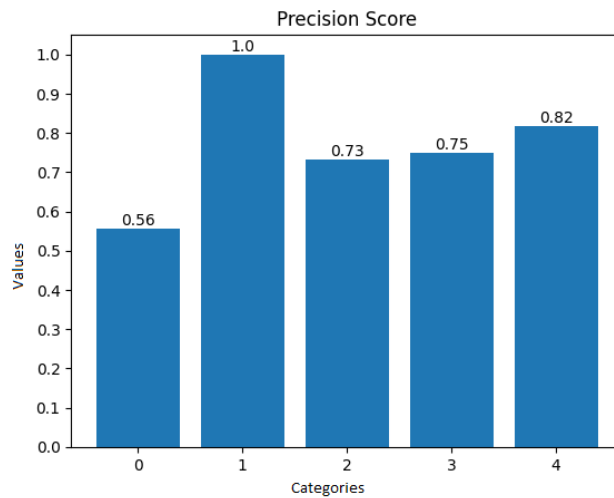
Seperti yang ditunjukkan pada Gambar 17, evaluasi kinerja model klasifikasi dilakukan dengan menggunakan *confusion matrix* untuk membandingkan label aktual dengan label prediksi. Sumbu vertikal menunjukkan label aktual (*True label*), sedangkan sumbu horizontal menunjukkan label prediksi (*Predicted label*). Nilai pada setiap sel menggambarkan jumlah data dalam kombinasi label aktual dan prediksi. Misalnya, sel pada baris ke-3 dan kolom ke-3 menunjukkan 11 data surat yang benar diklasifikasikan sebagai 'Surat Permohonan' (label 2), sementara sel pada baris ke-1 dan kolom ke-0 menunjukkan 5 data surat 'Surat Keputusan' (label 1) yang salah diprediksi sebagai 'Surat Edaran' (label 0). Perhitungan metrik evaluasi seperti *accuracy*, *precision*, *recall*, dan *F1 score* didasarkan pada *confusion matrix* ini.



Gambar 17. Confusion Matrix

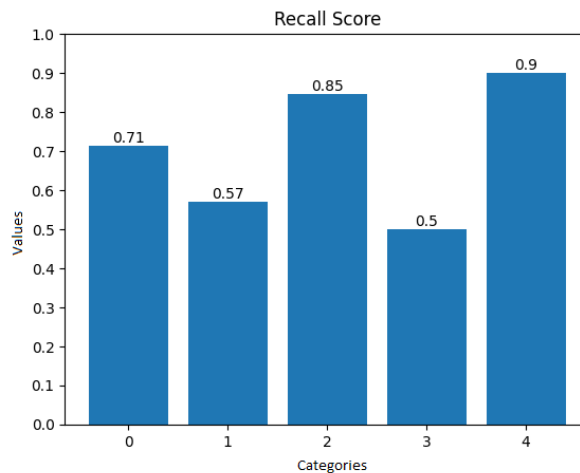
Model klasifikasi ini mencapai akurasi sebesar 74.4%, yang berarti 74.4% dari seluruh data uji berhasil diklasifikasikan dengan benar. Meskipun nilai ini menunjukkan kemampuan model yang cukup baik, masih ada ruang untuk peningkatan, terutama dalam hal kualitas data latih, pemilihan algoritma, dan teknik *preprocessing*.

Pada Gambar 18 terlihat presisi (*Precision Score*), yang mengukur keakuratan prediksi model untuk setiap kategori, menunjukkan variasi performa. Kategori 'Surat Keputusan' (1) memiliki presisi sempurna (1.0), artinya semua prediksinya akurat. Namun, 'Surat Edaran' (0) hanya mencapai presisi 0.56, menunjukkan bahwa 44% prediksinya salah.



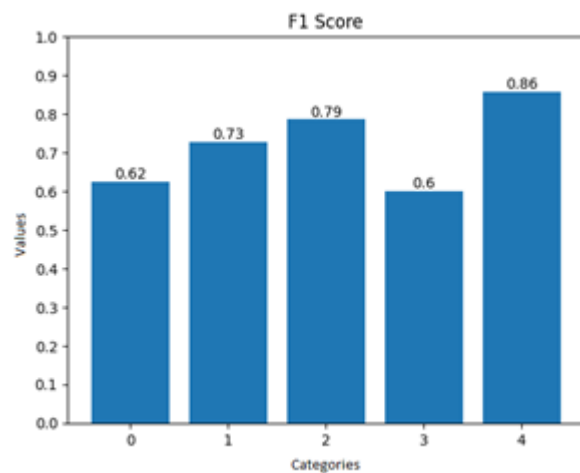
Gambar 18. Precision Score

Sementara itu, *Recall Score* terlihat pada Gambar 19 yang mengukur kemampuan model dalam mendeteksi data yang relevan, mencatat nilai tertinggi pada 'Surat Undangan' (4) dengan 0.9, tetapi terendah pada 'Surat Keputusan' (1) dengan 0.57.



Gambar 19. Recall Score

*F1-Score*, yang berfungsi sebagai harmonisasi antara *precision* dan *recall*, menunjukkan seberapa baik performa model. Kategori dengan *precision* dan *recall* tinggi (seperti 'Surat Undangan') secara alami memiliki *F1-Score* yang baik, sedangkan kategori dengan ketimpangan antara *precision* dan *recall* (seperti 'Surat Keputusan' yang presisinya 1.0 tetapi *recall*-nya 0.57) akan memiliki *F1-Score* yang lebih rendah. Hasil ini menunjukkan bahwa meskipun model cukup akurat dalam memprediksi beberapa kategori (*high precision*), kemampuannya untuk menemukan semua kasus yang relevan (*recall*) masih perlu ditingkatkan, terutama untuk kategori 'Surat Edaran' dan 'Surat Keputusan'. Diagram batang dari *F1-Score* terlihat pada Gambar 20.



Gambar 20. *F1 Score*

Model klasifikasi berhasil mengkategorikan surat dengan akurasi 74,4%, namun performanya bervariasi antar kategori. Kategori seperti 'Surat Keputusan' memiliki *precision* tinggi namun *recall* rendah, sedangkan 'Surat Edaran' menunjukkan *precision* terendah. Model cukup menjanjikan, tetapi masih perlu perbaikan pada data latih, algoritma, dan *preprocessing* untuk meningkatkan performa secara menyeluruh.

Pengujian komprehensif diperlukan untuk memastikan sistem pencatatan surat digital berfungsi optimal. Salah satu metode yang digunakan adalah *black-box testing*, yang mengevaluasi fungsionalitas sistem dari perspektif pengguna tanpa mempertimbangkan aspek teknis internal.

Sebagaimana ditunjukkan pada Tabel 1, berikut merupakan hasil uji coba *black-box* yang dirancang untuk mengevaluasi berbagai skenario penggunaan sistem.

Tabel 1. Table Pengujian Blackbox Sistem

No	Modul	Test Case	Hasil yang Diharapkan	Status
1	<i>Login</i>	Kredensial valid	Akses berhasil ke beranda	Berhasil
2		Kredensial tidak valid	Menampilkan pesan <i>error</i>	Berhasil
3		Field kosong	Peringatan field wajib diisi	Berhasil
4	Beranda	Tampilan profil	Informasi profil lengkap	Berhasil
5	Surat Masuk	Tampilan daftar	Kolom lengkap + fitur pencarian & tambah data	Berhasil
6		CRUD surat masuk	Operasi create/read/update/delete berfungsi	Berhasil
7		Otomatisasi ( <i>single/batch</i> )	Scan dan input otomatis berhasil	Berhasil
8	Surat Keluar	Tampilan daftar	Kolom lengkap + fitur pencarian & tambah data	Berhasil
9		CRUD surat keluar	Operasi create/read/update/delete berfungsi	Berhasil
10		Otomatisasi ( <i>single/batch</i> )	Scan dan input otomatis berhasil	Berhasil
11	Manajemen Pengguna	CRUD pengguna	Operasi tambah/ubah/hapus berfungsi	Berhasil
12	<i>Logout</i>	Proses <i>logout</i>	Sesi berakhir & kembali ke halaman login	Berhasil

## V. KESIMPULAN

Studi ini menunjukkan bahwa *text mining* yang menggunakan algoritma *Naive Bayes* dapat membantu mengotomatisasi klasifikasi surat masuk di Kantor Kecamatan Motoling Barat. Dengan akurasi sebesar 74,4%, sistem yang dikembangkan berhasil mengkategorikan surat secara cukup akurat ke dalam lima jenis, meskipun masih terdapat variasi performa antar kategori. Pengujian menggunakan *confusion matrix* menunjukkan bahwa model memiliki presisi tinggi pada beberapa kategori, namun *recall* yang rendah pada kategori lainnya menunjukkan perlunya peningkatan lebih lanjut dalam teknik *preprocessing* dan pemodelan. Dari hasil pengujian kotak hitam atau *blackbox* terungkap bahwa setiap modul, mulai dari proses otentikasi pengguna, pengelolaan surat-menyurat, klasifikasi otomatis, hingga administrasi pengguna, berfungsi sempurna sesuai rancangan, sekaligus memberikan dorongan nyata pada produktivitas kerja. Dengan dukungan arsitektur berbasis *React JS* dan *Django*, sistem ini tidak hanya unggul dari segi performa klasifikasi, tetapi juga responsif dan ramah pengguna. Penelitian ini membuka peluang untuk pengembangan lebih lanjut melalui pemanfaatan algoritma lain seperti SVM atau *ensemble learning*, serta perluasan dataset untuk meningkatkan akurasi dan generalisasi model. Dengan demikian, sistem ini berpotensi besar mendukung transformasi digital kearsipan di lingkungan pemerintahan.

## VI. REFERENSI

- Arsip Nasional Republik Indonesia. 2021. *Peraturan Arsip Nasional Republik Indonesia Nomor 6 Tahun 2021 Tentang Pengelolaan Arsip Elektronik*. Vol. 1.
- Aubaid, Asmaa M., Alok Mishra, and Atul Mishra. 2024. "Machine Learning and Rule-Based Embedding Techniques for Classifying Text Documents." *International*

- Journal of System Assurance Engineering and Management* 15(12):5637–52. doi: 10.1007/s13198-024-02555-w.
- Dewi, Findra Kartika Sari. 2021. “Klasifikasi Berita Menggunakan Metode Multinomial Naïve Bayes.” *SCAN - Jurnal Teknologi Informasi Dan Komunikasi* 16(3):1–8. doi: 10.33005/scan.v16i3.2870.
- Farhan, Farhan, Triase Triase, and Aninda Muliani Harahap. 2023. “Penggunaan Algoritma Naive Bayes Dalam Text Mining Untuk Klasifikasi Pasal UU ITE.” *J-SISKO TECH (Jurnal Teknologi Sistem Informasi Dan Sistem Komputer TGD)* 6(2):314. doi: 10.53513/jsk.v6i2.7896.
- Fauzan, Irsyad Nur, Marsya Sabrina Putri, and Bakhitah Kurnia Endiyanti. 2024. “Pengaruh Efektivitas Sistem Pengarsipan Digital Terhadap Produktivitas Karyawan Di Lingkungan Perkantoran.” *Komunika : Jurnal Ilmiah Komunikasi* 2(1):67–78. doi: 10.70437/komunika.v2i1.881.
- Gupta, Priyanshu. 2021. “Django and React Integration: Session Auth, CORS.” Retrieved April 16, 2025 (<https://priyanshuguptaofficial.medium.com/django-and-react-integration-b712321a5232>).
- Hasibuan, Muhammad Said, and RZ Abdul Aziz. 2022. “Detection of Learning Styles with Prior Knowledge Data Using the SVM, K-NN and Naïve Bayes Algorithms.” *Jurnal Infotel* 14(3):209–13. doi: 10.20895/infotel.v14i3.788.
- Hijriani, Astria, Jannati Asri Safitri, Raden Irwan Adi Pribadi, and Rico Andrian. 2020. “Pengembangan Sistem Informasi Manajemen Supplier Dan Barang Dengan Extreme Programming.” *Jurnal Teknik Informatika Dan Sistem Informasi* 6(1):28–43. doi: 10.28932/jutisi.v6i1.2132.
- Kashid, Dhruv S., Janhavi D. Patil, and Amar Buchade. 2025. “Live News Classification Using Naive Bayes Classifier.” *Cureus Journal of Computer Science*. doi: 10.7759/s44389-024-01030-8.
- Lolytasari, Lolytasari, and Ardelia Dirsanalala. 2023. “Digitalisasi Arsip Dalam Mendukung Pelaksanaan E-Government.” *Shaut Al-Maktabah : Jurnal Perpustakaan, Arsip Dan Dokumentasi* 15(1):18–30. doi: 10.37108/shaut.v15i1.963.
- Luo, Xiaoyu. 2021. “Efficient English Text Classification Using Selected Machine Learning Techniques.” *Alexandria Engineering Journal* 60(3):3401–9. doi: 10.1016/j.aej.2021.02.009.
- M. Afriansyah, Joni Saputra, Valian Yoga Pudya Ardhana, and Yuan Sa’adati. 2024. “Algoritma Naive Bayes Yang Efisien Untuk Klasifikasi Buah Pisang Raja Berdasarkan Fitur Warna.” *Journal of Information Systems Management and Digital Business* 1(2):236–48. doi: 10.59407/jismdb.v1i2.438.
- Maramis, Glenn D. P. 2023. “Evaluating Learning Management Systems in University: Evidence in University in North Sulawesi.” *International Journal of Information Technology and Education* 2(2):103–18. doi: 10.62711/ijite.v2i2.109.
- Moningkey, Efraim R. S., Mochamad Bruri Triyono, and Priyanto. 2022. “Mokobang SDGP Website Analysis and Design.” *International Journal of Information Technology and Education* 1(2):86–90. doi: 10.62711/ijite.v1i2.45.
- Nurhasanah, Siti, and Muchamad Rusdan. 2020. “Development of Front End on Tour and Travel Applications Using Python and Django Framework in PT. Industri Telekomunikasi Indonesia.” *Journal of Informatics and Communication Technology (JICT)* 2(1):31–38. doi: 10.52661/j\_ict.v2i1.42.
- Pemerintah Pusat. 2018. *Peraturan Presiden Nomor 95 Tahun 2018 Tentang Sistem Pemerintahan Berbasis Elektronik*.
- Pramudya, Bintang, Dinda Chesar Putri Ramadhani, Hilda Nuzulul Mujaddidah, and Risqy

- Siwi Pradini. 2025. "Implementation of Extreme Programming (XP) in the Development of Dental Clinic Information Systems." *Journal of Enhanced Studies in Informatics and Computer Applications* 2(1):20–28. doi: 10.47794/jesica.v2i1.22.
- Prayogo, Anang, Fauziah Fauziah, and Winarsih Winarsih. 2023. "PERBANDINGAN ALGORITMA NAÏVE BAYES DAN K-NEAREST NEIGHBOR PADA KLASIFIKASI JUDUL ARTIKEL PADA JURNAL ILMIAH." *JUPI (Jurnal Ilmiah Penelitian Dan Pembelajaran Informatika)* 8(4):1327–38. doi: 10.29100/jupi.v8i4.4141.
- Ramudin, Riko Priyatmo, Ahmad M. Ramli, and Hilman Rosmana. 2024. "Kepastian Hukum Penerapan Aplikasi Sistem Informasi Kearsipan Dinamis Terintegrasi (SRIKANDI)." *Khazanah: Jurnal Pengembangan Kearsipan* 17(2):142. doi: 10.22146/khazanah.93331.
- Resti, Yulia, Ning Eliyati, Mau'izatil Rahmayani, Des Alwine Zayanti, Endang Sri Kresnawati, Endro Setyo Cahyono, and Irsyadi Yani. 2024. "Ensemble of Naive Bayes, Decision Tree, and Random Forest to Predict Air Quality." *IAES International Journal of Artificial Intelligence* 13(3):3039–51. doi: 10.11591/ijai.v13.i3.pp3039-3051.
- Shah, Nakul. 2021. "React Native and Django for Beginners." Retrieved April 16, 2025 (<https://crowdbotics.com/posts/blog/react-native-django-for-beginners/>).
- Szabó, Bálint, and Károly Hercegf. 2023. "User-Centered Approaches in Software Development Processes: Qualitative Research into the Practice of Hungarian Companies." *Journal of Software: Evolution and Process* 35(2):1–28. doi: 10.1002/smr.2501.
- Taha, Kamal, Paul D. Yoo, Chan Yeun, Dirar Homouz, and Aya Taha. 2024. "A Comprehensive Survey of Text Classification Techniques and Their Research Applications: Observational and Experimental Insights." *Computer Science Review* 54:100664. doi: 10.1016/j.cosrev.2024.100664.
- Triana, Apri, Halimi, Syarif Hidayatullah, and Zikri Muhammad Iqbal. 2024. "Characteristics of Digital And Manual Archive Management." *JUPI (Jurnal Ilmu Perpustakaan Dan Informasi)* 9(2):194. doi: 10.30829/jupi.v9i2.21450.
- Zhang, Peng, Zifan Ma, Zeyuan Ren, Hongxiang Wang, Chuankai Zhang, Qing Wan, and Dongxue Sun. 2024. "Design of an Automatic Classification System for Educational Reform Documents Based on Naive Bayes Algorithm." *Mathematics* 12(8):1127. doi: 10.3390/math12081127.