Indonesian Journal of Multidisciplinary on Social and Technology
Homepage: https://journal.ilmudata.co.id/index.php/ijmst
Vol. 4 No. 3 (2026) pp: 1803-1813
P-ISSN: 2986-6790, e-ISSN: 2986-6782

Smart Searching Pada Agenda Dan Perpustakaan Berbasis Web Menggunakan Algoritma Trie Dan Jaro-Winkler Distance

Nurendro Hardjo Judanto. R[1], Asrul Sani[2], Bambang Pramono[3]
[1,2,3] Informatika, Teknik, Universitas Halu Oleo
[1]nurendro.47@gmail.com, [2]sani.asrul@uho.ac.id * , [3]bambang_pramono@uho.ac.id *

Abstrak
The management of activity agendas and libraries at the Southeast Sulawesi High Prosecutor's Office faces challenges because data recording and searching are not fully integrated, while direct text-matching-based searches are intolerant of typos, spelling differences, or incomplete keywords. This study aims to design and build SIGAP (Agenda and Library Information System), a web application with a smart searching mechanism that combines the Trie algorithm for autosuggestion and Jaro-Winkler Distance for fuzzy search. The system was developed using the Rational Unified Process through the inception, elaboration, construction, and transition phases, with Laravel, PHP, Tailwind CSS, JavaScript, Python, MySQL, and a hybrid search service as the main components. The test data consisted of 500 book data and 21 agenda data. Evaluation was carried out through speed testing, white box, precision-recall, and User Acceptance Testing. The results showed that the average search time decreased from 25.0 seconds in manual searches to 239.1 milliseconds or approximately 0.24 seconds in the hybrid system, equivalent to an acceleration of approximately 104.5 times. All 17 white-box cases passed, while the evaluation of 12 queries yielded 100% recall and 89.41% precision. User acceptance testing on 10 respondents yielded an average of 86.76%, categorized as excellent. These results indicate that the combination of Trie and Jaro-Winkler Distance improves speed, relevance, tolerance to input errors, and user acceptance in agenda and library data searches.

Keywords : Smart Searching, Trie, Jaro-Winkler Distance, Autosuggestion, Fuzzy Search

Abstrak
Pengelolaan agenda kegiatan dan perpustakaan di Kejaksaan Tinggi Sulawesi Tenggara menghadapi kendala karena pencatatan dan pencarian data belum sepenuhnya terintegrasi, sedangkan pencarian berbasis pencocokan teks langsung tidak toleran terhadap kesalahan pengetikan, perbedaan ejaan, atau kata kunci yang tidak lengkap. Penelitian ini bertujuan merancang dan membangun SIGAP (Sistem Informasi Agenda dan Perpustakaan), yaitu aplikasi web dengan mekanisme smart searching yang mengombinasikan algoritma Trie untuk autosuggestion dan Jaro-Winkler Distance untuk fuzzy search. Sistem dikembangkan menggunakan Rational Unified Process melalui fase inception, elaboration, construction, dan transition, dengan Laravel, PHP, Tailwind CSS, JavaScript, Python, MySQL, serta layanan pencarian hibrida sebagai komponen utama. Data uji terdiri atas 500 data buku dan 21 data agenda. Evaluasi dilakukan melalui pengujian kecepatan, white box, precision-recall, dan User Acceptance Testing. Hasil menunjukkan waktu pencarian rata-rata turun dari 25,0 detik pada pencarian manual menjadi 239,1 milidetik atau sekitar 0,24 detik pada sistem hibrida, setara percepatan sekitar 104,5 kali. Seluruh 17 kasus white box lulus, sedangkan evaluasi terhadap 12 kueri menghasilkan Recall 100% dan Precision 89,41%. Pengujian penerimaan pengguna terhadap 10 responden memperoleh rata-rata 86,76% dengan kategori sangat baik. Hasil tersebut menunjukkan bahwa kombinasi Trie dan Jaro-Winkler Distance mampu meningkatkan kecepatan, relevansi, toleransi terhadap kesalahan input, dan penerimaan pengguna dalam pencarian data agenda dan perpustakaan.

Kata Kunci: *Smart Searching, Trie, Jaro-Winkler Distance, Autosuggestion, Fuzzy Search*

1. Pendahuluan

Transformasi digital pada instansi pemerintah menuntut proses administrasi yang cepat, konsisten, dan mudah ditelusuri. Ketika data masih tersebar pada media manual atau aplikasi yang tidak saling terhubung, organisasi menghadapi risiko keterlambatan, duplikasi, ketidaksesuaian informasi, dan meningkatnya waktu yang dibutuhkan untuk menemukan kembali dokumen. Kajian mengenai e-government menunjukkan bahwa pemanfaatan teknologi harus diikuti kesesuaian organisasi, penerimaan pengguna, dan kesiapan proses agar manfaat digitalisasi benar-benar tercapai (Alfiani et al., 2024). Dalam konteks administrasi publik Indonesia, teknologi informasi juga dipandang sebagai instrumen penting untuk meningkatkan efektivitas layanan, akurasi data, dan produktivitas aparatur (Agni Afdilati, 2025). Sistem administrasi berbasis *web* relevan karena memungkinkan penyimpanan terpusat dan akses melalui jaringan internal tanpa bergantung pada satu perangkat tertentu (Nita et al., 2025).
Kejaksaan Tinggi Sulawesi Tenggara memiliki aktivitas administrasi yang meliputi agenda kegiatan, koordinasi antarbagian, penjadwalan rapat, pengelolaan koleksi buku, peminjaman, dan pengembalian. Berdasarkan observasi dan wawancara dengan staf persuratan serta perpustakaan, proses pencatatan dan pencarian data belum sepenuhnya terintegrasi. Agenda dapat dicatat pada media yang berbeda, sedangkan pencarian buku masih bergantung pada

pencocokan teks langsung. Kondisi tersebut menyulitkan pengguna ketika kata kunci tidak lengkap, terdapat kesalahan ketik, atau pengguna hanya mengingat sebagian judul. Permasalahan menjadi lebih nyata ketika volume data bertambah karena pengguna harus melakukan pencarian berulang atau menelusuri daftar secara manual (Sugiarso et al., 2025).

Pengelolaan agenda berbasis *web* telah digunakan untuk meningkatkan keteraturan pencatatan kegiatan dan memudahkan penelusuran kembali data. Menunjukkan bahwa sistem agenda terkomputerisasi dapat membantu pengelolaan kegiatan secara lebih terstruktur. Pada sisi perpustakaan, sumber daya informasi memiliki hubungan erat dengan produktivitas riset dan pekerjaan pengguna (Scoulas & De Groote, 2024), sedangkan implementasi perpustakaan berbasis *web* dapat meningkatkan efisiensi layanan peminjaman dan akses koleksi (Asari et al., 2025). Namun, penyediaan antarmuka digital saja belum menyelesaikan masalah temu kembali apabila mekanisme pencarian masih menuntut kecocokan *string* secara eksak.

Sistem temu kembali informasi yang efektif perlu menempatkan relevansi dan efisiensi sebagai sasaran utama. Penelitian terdahulu menunjukkan bahwa kualitas mekanisme pencarian memengaruhi kemampuan pengguna menemukan dokumen secara tepat, terutama ketika koleksi heterogen dan *query* bervariasi (honkapohja et al., 2023; Mhawi et al., 2022). Perkembangan pencarian cerdas pada perpustakaan digital juga menekankan pentingnya rekomendasi, pemeringkatan, dan toleransi terhadap variasi bahasa agar pengguna tidak dibatasi oleh bentuk penulisan yang benar-benar identik (Shamsitdinova M et al., 2024). Karena itu, penelitian ini menggunakan konsep *smart searching* yang mengombinasikan pencarian berbasis awalan dan pencocokan *fuzzy*.

Algoritma *Trie* dipilih untuk mendukung autosuggestion. *Trie* menyimpan karakter secara hierarkis sehingga pencarian *prefix* dapat dilakukan dengan menelusuri jalur karakter yang diketik pengguna. Pada kondisi ideal, waktu penelusuran *prefix* bergantung pada panjang input, bukan pada banyaknya seluruh kata yang tersimpan. Penelitian mengenai mini *search engine* menunjukkan bahwa *Trie* sesuai untuk *autocomplete* dan pencarian berbasis awalan secara *real time* (Shamitha, 2025). Kajian struktur Multiset-*Trie* juga memperlihatkan bahwa representasi pohon berbasis elemen dapat mendukung operasi pencarian terstruktur secara efisien (Akulich et al., 2023). Dalam SIGAP, *Trie* digunakan untuk membatasi ruang kandidat sekaligus memberikan saran kata ketika pengguna mulai memasukkan karakter.

Trie tidak cukup ketika pengguna menulis kata yang salah sejak awal, menghilangkan karakter, atau menukar posisi huruf. Untuk menangani kondisi tersebut, penelitian ini menerapkan *Jaro-Winkler Distance* sebagai pengukur kemiripan *string*. *Jaro-Winkler* memberikan skor antara 0 dan 1 serta menambahkan bobot pada kesamaan *prefix*, sehingga sesuai untuk kasus salah ketik pada nama, judul, atau istilah pendek. Penerapan *fuzzy string matching* pada chatbot berbahasa Indonesia menunjukkan bahwa mekanisme kemiripan dapat membantu mengenali variasi input (Suwarningsih & Nuryani, 2024). Metode *Jaro-Winkler* juga telah diterapkan pada kamus ilmiah (Rasna et al., 2025), koreksi ejaan (Ahamed et al., 2021), penilaian teks dengan *typo* bahasa Indonesia (Jayadianti et al., 2023), serta deteksi kemiripan teks hibrida (Yulianto & Nurhasanah, 2021). Kajian record matching terkini semakin menegaskan perlunya keseimbangan antara akurasi dan biaya komputasi pada pencocokan *fuzzy real time* (Rozinek et al., 2025).

Celah penelitian terletak pada integrasi dua kebutuhan yang selama ini sering dibahas terpisah. Sebagian studi berfokus pada digitalisasi agenda atau perpustakaan, sedangkan studi lain menguji satu algoritma pencarian pada domain tertentu. Masih terbatas penelitian yang mengintegrasikan agenda dan perpustakaan dalam satu aplikasi *web* pemerintah, menggunakan *Trie* sebagai lapisan autosuggestion dan *Jaro-Winkler Distance* sebagai lapisan *fuzzy search*, lalu mengevaluasinya secara bersamaan dari sisi kecepatan, logika internal, *precision-recall*, serta penerimaan pengguna. Penelitian ini mengisi celah tersebut melalui SIGAP, yaitu Sistem Informasi Agenda dan Perpustakaan dengan arsitektur pencarian hibrida dan pengujian pada 500 data buku serta 21 data agenda.

Kontribusi penelitian ini terdiri atas tiga aspek. Pertama, penelitian menghasilkan sistem *web* terintegrasi dengan tiga peran pengguna, yaitu Super Admin, Admin/Staff, dan Pegawai. Kedua, penelitian membangun alur pencarian bertingkat yang mengutamakan kecocokan *prefix* melalui *Trie* dan menggunakan *Jaro-Winkler Distance* untuk menangani kesalahan penulisan dengan *threshold* 0,85. Ketiga, penelitian menyajikan evaluasi kuantitatif yang membandingkan pencarian manual dan sistem hibrida, mengukur *precision-recall* pada 12 *query*, memeriksa 17 jalur *white box*, serta menilai penerimaan 10 responden melalui *User Acceptance Testing*. Tujuan penelitian adalah mengetahui sejauh mana kombinasi kedua algoritma dapat meningkatkan kecepatan, ketepatan, dan kemudahan pencarian agenda serta buku di Kejaksaan Tinggi Sulawesi Tenggara.

2. Metode Penelitian

2.1. Desain Penelitian dan Sumber Data

Penelitian menggunakan pendekatan rekayasa perangkat lunak dengan studi kasus pada pengelolaan agenda dan perpustakaan di Kejaksaan Tinggi Sulawesi Tenggara. Data kebutuhan diperoleh melalui tinjauan pustaka, observasi proses kerja, wawancara dengan staf persuratan dan perpustakaan, serta dokumentasi data agenda, buku, peminjaman, pengembalian, dan pengguna. Observasi difokuskan pada alur pencatatan dan pencarian yang

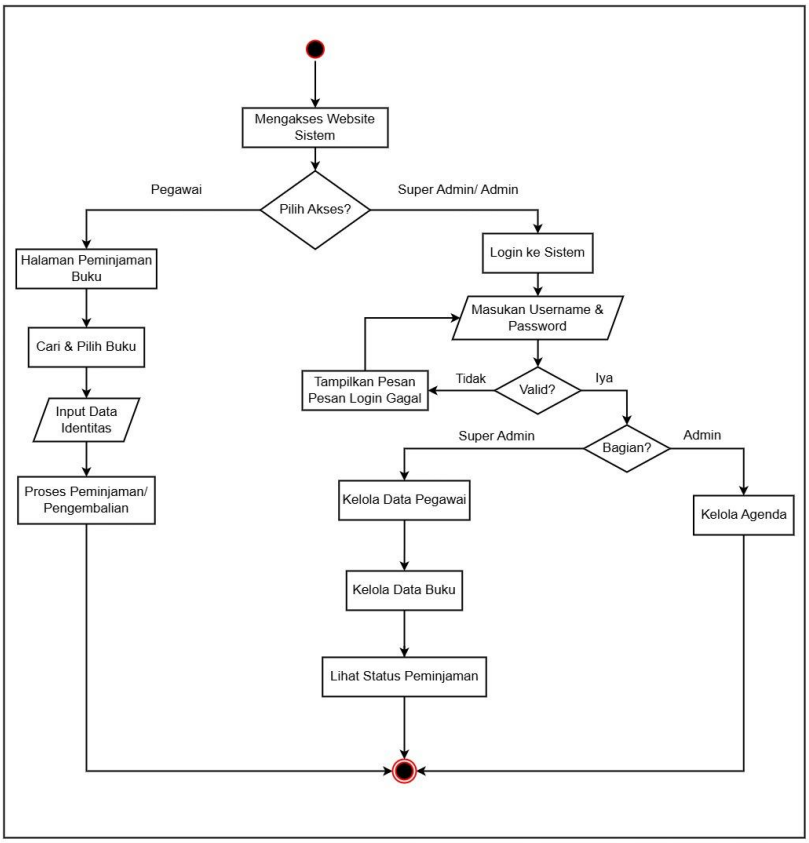
DOI: https://doi.org/10.69693/ijmst.v4i2.12274
Lisensi: Creative Commons Attribution 4.0 International (CC BY 4.0)

berjalan, sedangkan wawancara digunakan untuk mengidentifikasi kesulitan pengguna ketika data hanya dapat ditemukan melalui pencocokan teks langsung. Dokumentasi dipakai untuk memetakan struktur data dan menyiapkan bahan implementasi serta pengujian.

Sistem diuji menggunakan 500 data buku dan 21 data agenda. Pengguna dibagi menjadi tiga peran. Super Admin mengelola akun, buku, transaksi peminjaman dan pengembalian, statistik, log, laporan, serta pengaturan sistem. Admin/Staff mengelola agenda dan melakukan pencarian agenda. Pegawai mengakses daftar buku, melakukan pencarian, peminjaman, dan pengembalian. Pemisahan hak akses tersebut mempertahankan konteks operasional masing-masing pengguna sekaligus menyediakan satu sumber data terpusat.

2.2. Pengembangan Sistem

Pengembangan sistem mengikuti *Rational Unified Process* (RUP), yaitu proses iteratif yang membagi pengembangan ke dalam fase *inception*, *elaboration*, *construction*, dan *transition*. Pendekatan RUP dipilih karena mendukung pengembangan bertahap dari identifikasi kebutuhan sampai evaluasi sistem dan telah digunakan pada pengembangan aplikasi *web* yang menuntut arsitektur terstruktur (Firmansyah et al., 2024). Pada fase *inception* dilakukan identifikasi masalah, aktor, ruang lingkup, dan tujuan pencarian cerdas. Fase *elaboration* memuat analisis kebutuhan, perancangan basis data MySQL, pemodelan UML, serta rancangan mekanisme *Trie* dan *Jaro-Winkler Distance*. Fase *construction* mengimplementasikan sistem menggunakan Laravel, PHP, Tailwind CSS, JavaScript, Python, dan MySQL. Fase *transition* mencakup *deployment*, pengujian teknis, *User Acceptance Testing*, dan penyempurnaan berdasarkan umpan balik pengguna.



Gambar 1. Alur Kerja Sistem Secara Umum SIGAP

Arsitektur SIGAP memisahkan antarmuka aplikasi, basis data, dan layanan pencarian. Aplikasi Laravel menerima *query* pengguna dan mengambil indeks agenda atau buku dari basis data. *Query* beserta indeks dikirim ke *Search_Service.py* yang melakukan pembersihan input, tokenisasi, pembentukan *Trie*, perhitungan skor kemiripan, *filtering* berdasarkan *threshold*, ranking, dan *fallback search*. Pemisahan layanan ini menjaga modularitas karena perubahan nilai ambang, bobot *field*, atau logika pencarian dapat dilakukan tanpa mengubah seluruh komponen aplikasi.

2.3. Mekanisme Pencarian Hibrida

Trie menyimpan setiap kata sebagai rangkaian node karakter dari *root* sampai penanda akhir kata. Ketika pengguna mengetik sebuah *prefix*, sistem menelusuri node sesuai urutan karakter. Jika *prefix* ditemukan, penelusuran *Depth-First Search* mengumpulkan kata-kata yang berada di bawah node tersebut sebagai kandidat autosuggestion. Sebagai contoh, data AGEN, AGENDA, ARSIP, SURAT, SURPAT, SURATMU, dan SALINAN menghasilkan

DOI: https://doi.org/10.69693/ijmst.v4i2.12274
Lisensi: Creative Commons Attribution 4.0 International (CC BY 4.0)

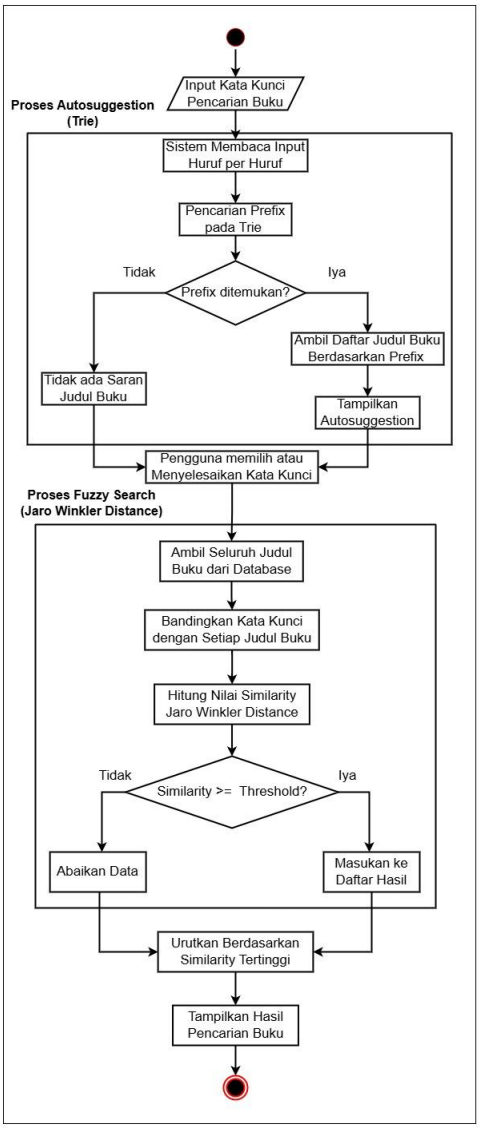
saran AGEN dan AGENDA untuk *prefix* "AGE", sedangkan *prefix* "SUR" menghasilkan SURAT, SURATMU, dan SURPAT. Kompleksitas penelusuran *prefix* dinyatakan *O(m)*, dengan m sebagai panjang *prefix*.
Apabila *query* tidak identik atau mengandung kesalahan, sistem menghitung *Jaro-Winkler Distance*. Nilai *Jaro* dihitung berdasarkan jumlah karakter cocok, panjang kedua *string*, dan transposisi. Setelah itu, penyesuaian *Winkler* menambahkan bonus untuk *prefix* yang sama sampai maksimal empat karakter. Bentuk perhitungannya ditunjukkan pada Persamaan (1) dan Persamaan (2).

$$d_j = \frac{1}{3}\left(\frac{m}{|s_1|} + \frac{m}{|s_2|} + \frac{m-t}{m}\right) \quad (1)$$

$$d_w = d_j + (l \cdot p \cdot 1 - d_j)) \quad (2)$$

Pada persamaan *Jaro-Winkler*, variabel m menunjukkan jumlah karakter yang cocok antara dua *string*, yaitu karakter yang memiliki nilai sama dan berada dalam rentang posisi tertentu. Variabel t menyatakan setengah dari jumlah transposisi, yaitu karakter yang cocok tetapi berada pada urutan atau posisi yang berbeda. Variabel l menunjukkan panjang *prefix* yang sama pada awal kedua *string* dengan batas maksimum empat karakter, sedangkan p merupakan faktor skala yang memberikan bobot tambahan terhadap kesamaan *prefix*. Pada sistem ini, nilai faktor skala ditetapkan sebesar 0,1. Sistem juga menerapkan *threshold* sebesar 0,85 sebagai batas minimum penerimaan kandidat. Kandidat dengan skor di bawah 0,85 ditolak, sedangkan kandidat dengan skor sama dengan atau lebih besar dari 0,85 dipertahankan dan diurutkan berdasarkan nilai *similarity* dari yang tertinggi ke terendah.
Contoh perhitungan manual menunjukkan bahwa *string* AGENA dibandingkan dengan AGENDA menghasilkan skor *similarity* sebesar 0,967, meskipun terdapat satu karakter yang hilang. *String* PEMINJMAN dibandingkan dengan PEMINJAMAN memperoleh skor 0,958 karena terdapat penghilangan satu karakter. Sementara itu, *string* SUART dibandingkan dengan SURAT menghasilkan skor 0,947 meskipun terjadi perubahan urutan karakter. Ketiga contoh tersebut memperoleh skor di atas *threshold* 0,85 sehingga tetap dianggap relevan oleh sistem. Hasil ini menunjukkan bahwa algoritma *Jaro-Winkler* mampu menangani beberapa bentuk kesalahan pengetikan, seperti penghilangan karakter, pertukaran posisi karakter, dan variasi kecil dalam susunan kata.
Dalam implementasinya, algoritma *Trie* dan *Jaro-Winkler* ditempatkan dalam satu alur pencarian hibrida. *Trie* digunakan untuk melakukan pencarian berbasis *prefix* secara cepat dan memberikan prioritas tambahan kepada kandidat yang memiliki kecocokan awalan secara eksak dengan input pengguna. Setelah kandidat diperoleh, *Jaro-Winkler* menghitung tingkat kemiripan antara input dan setiap kandidat, terutama ketika terdapat *typo*, karakter yang hilang, atau karakter yang tertukar. Kandidat yang memenuhi *threshold* 0,85 kemudian diurutkan berdasarkan skor *similarity* tertinggi sehingga hasil yang paling mirip memperoleh posisi teratas dalam ranking pencarian.
Apabila hasil utama tidak memenuhi syarat atau belum menghasilkan kandidat yang memadai, sistem menjalankan mekanisme *fallback* berbasis *substring*. Mekanisme ini mencari keberadaan bagian kata dari input pengguna di dalam data kandidat agar sistem tetap dapat memberikan alternatif hasil. Dengan demikian, kombinasi *Trie*, *Jaro-Winkler*, dan *fallback substring* membentuk strategi pencarian bertingkat. *Trie* menangani kecocokan *prefix*, *Jaro-Winkler* memberikan toleransi terhadap kesalahan pengetikan, sedangkan *substring fallback* menjadi mekanisme alternatif ketika dua tahap sebelumnya belum memberikan hasil yang sesuai. Strategi ini membantu menjaga ketepatan ranking sekaligus meningkatkan toleransi sistem terhadap variasi input pengguna.

DOI: https://doi.org/10.69693/ijmst.v4i2.12274
Lisensi: Creative Commons Attribution 4.0 International (CC BY 4.0)



Gambar 2. Alur pencarian hibrida *Trie* dan *Jaro-Winkler Distance*

2.4. Rancangan Pengujian

Evaluasi dilakukan pada empat dimensi. Pertama, pengujian kecepatan membandingkan pencarian manual dengan sistem hibrida. Pada metode manual, satu pengguna menelusuri lima target agenda atau buku menggunakan scrolling dan stopwatch. Pada sistem hibrida, lima *query* diuji masing-masing tiga kali untuk memperoleh waktu respons API yang stabil, termasuk dua *query* dengan kesalahan pengetikan.

Pengujian *white box* memeriksa 17 kasus pada Controller, integrasi *Service* dan Python, *Trie*, *Jaro-Winkler Similarity*, *filtering* dan ranking, *fallback* dan *result processing*, serta logging. Ketiga, pengujian temu kembali menggunakan 12 *query* yang mewakili pencarian eksak, multikata, dan *typo* pada seluruh 21 agenda serta 500 buku. Ground truth ditentukan secara manual, lalu hasil sistem diklasifikasikan menjadi *true positive* (TP), *false positive* (FP), dan *false negative* (FN). *Precision* dan *Recall* dihitung menggunakan Persamaan (3) dan (4).

$$Precision = \frac{TP}{TP + FP} \; X \; 100\% \tag{3}$$

$$Recall = \frac{TP}{TP + FN} \; X \; 100\% \tag{4}$$

User Acceptance Testing melibatkan 10 responden: 2 Super Admin, 3 Admin/Staff, dan 5 Pegawai. Setiap kelompok menyelesaikan skenario tugas sesuai hak akses kemudian menilai 30 pernyataan menggunakan skala Likert lima tingkat. UAT digunakan untuk melihat penerimaan fungsi, kemudahan penggunaan, dan kesesuaian sistem dari perspektif pengguna; pendekatan ini lazim digunakan untuk mengevaluasi kelayakan sistem berbasis

DOI: https://doi.org/10.69693/ijmst.v4i2.12274
Lisensi: Creative Commons Attribution 4.0 International (CC BY 4.0)

web (Yakub et al., 2024). Nilai 81-100% dikategorikan sangat baik, 61-80% baik, 41-60% cukup, 21-40% kurang baik, dan 0-20% tidak baik.

3. Hasil dan Pembahasan

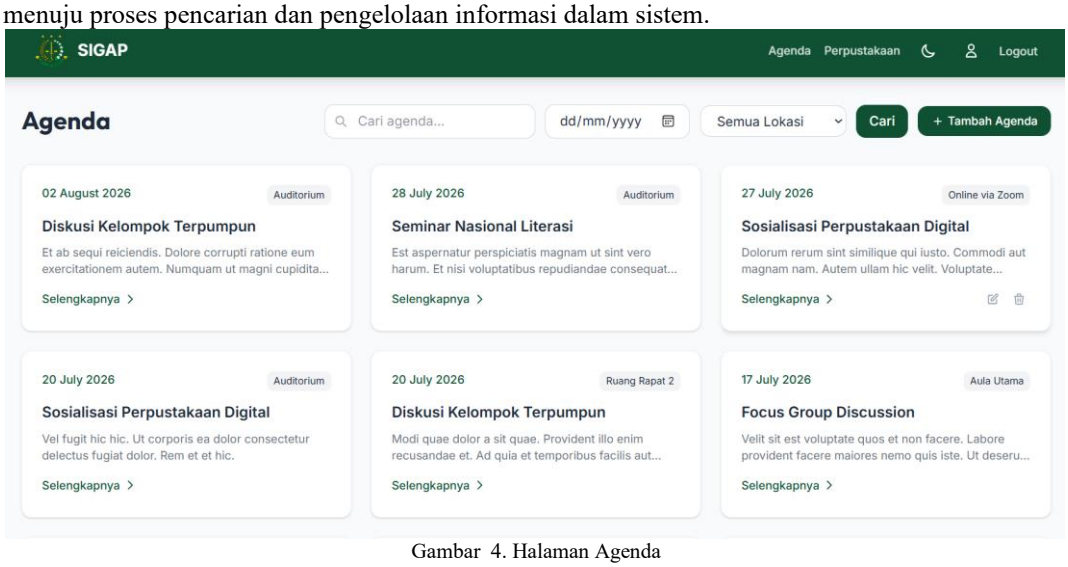
3.1. Implementasi SIGAP

Implementasi menghasilkan SIGAP (Sistem Informasi Agenda dan Perpustakaan), sebuah aplikasi *web* yang mengintegrasikan pengelolaan agenda kegiatan dan layanan perpustakaan dalam satu platform. Sistem menggunakan Laravel dan PHP pada lapisan aplikasi, Tailwind CSS serta JavaScript pada antarmuka, MySQL sebagai basis data, dan Python sebagai *engine* pencarian. Integrasi ini memungkinkan seluruh indeks buku dan agenda diproses oleh layanan pencarian yang sama, tetapi hasil tetap didekorasi sesuai tipe data dan hak akses pengguna.



Gambar 3. Halaman Utama SIGAP

Halaman utama Sistem Informasi Agenda dan Perpustakaan (SIGAP) merupakan halaman awal yang berfungsi sebagai pusat akses bagi pengguna untuk menggunakan seluruh fitur utama sistem berbasis *web* milik Kejaksaan Tinggi Sulawesi Tenggara. Pada halaman ini terdapat menu navigasi Agenda dan Perpustakaan, serta dua tombol utama, yaitu Cari Agenda dan Cari Buku, yang memudahkan pengguna menuju fitur pencarian sesuai kebutuhan. Tampilan halaman dirancang sederhana dan responsif agar pengguna dapat memahami fungsi utama sistem sejak pertama kali mengaksesnya. Melalui fitur *smart search*ing, pengguna dapat mencari informasi agenda kegiatan dan koleksi buku dengan lebih cepat, termasuk ketika terjadi kesalahan pengetikan atau variasi kata kunci. Dengan demikian, halaman utama SIGAP tidak hanya berperan sebagai tampilan pembuka, tetapi juga sebagai pintu masuk utama menuju proses pencarian dan pengelolaan informasi dalam sistem.

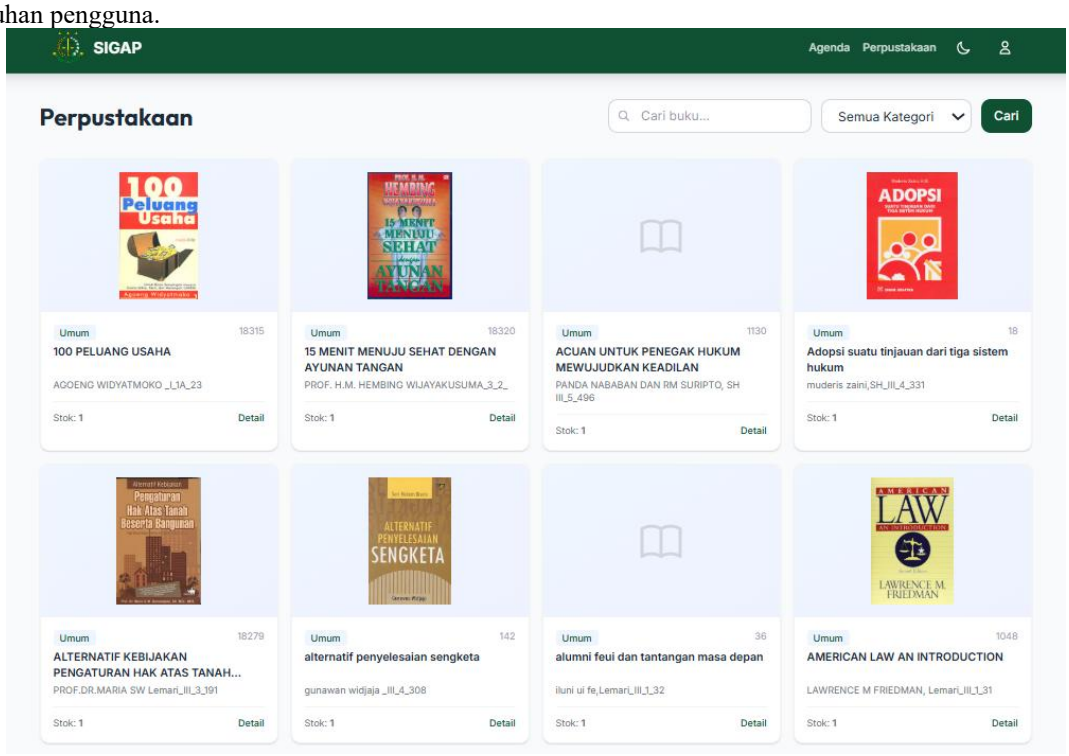


Gambar 4. Halaman Agenda

Halaman Agenda pada sistem SIGAP berfungsi untuk menampilkan dan mengelola informasi kegiatan secara terstruktur. Setiap agenda disajikan dalam bentuk kartu yang memuat informasi penting, seperti tanggal pelaksanaan, judul agenda, lokasi kegiatan, serta deskripsi singkat. Pengguna dapat memanfaatkan kolom pencarian untuk menemukan agenda berdasarkan kata kunci tertentu, serta menggunakan filter tanggal dan lokasi

DOI: https://doi.org/10.69693/ijmst.v4i2.12274
Lisensi: Creative Commons Attribution 4.0 International (CC BY 4.0)

untuk mempersempit hasil pencarian. Selain itu, tombol Selengkapnya memungkinkan pengguna melihat informasi agenda secara lebih rinci. Bagi pengguna yang memiliki hak akses sebagai administrator, tersedia fitur Tambah Agenda, Edit, dan Hapus untuk melakukan pengelolaan data secara langsung. Dengan fitur tersebut, proses pencarian dan pengelolaan agenda menjadi lebih cepat, terorganisasi, dan mudah digunakan sesuai kebutuhan pengguna.



Gambar 5. Halaman Daftar Buku

Pada halaman perpustakaan, pengguna dapat melihat koleksi, kategori, stok, dan detail buku, kemudian memasukkan kata kunci pada kolom pencarian. Desain berbasis *web* mendukung penggunaan melalui jaringan internal instansi dan sejalan dengan temuan bahwa digitalisasi perpustakaan berbasis *web* dapat mempercepat layanan peminjaman dan akses koleksi (Wijaya et al., 2022).

3.2. Verifikasi Algoritma *Trie* dan *Jaro-Winkler Distance*

Verifikasi manual menunjukkan bahwa *Trie* dan *Jaro-Winkler Distance* memberikan keluaran sesuai skenario yang dirancang. *Trie* berhasil membentuk jalur node dan menampilkan seluruh kandidat dengan *prefix* yang sama. Untuk *prefix* "AGE", sistem memperoleh AGEN dan AGENDA; untuk "SUR", sistem memperoleh SURAT, SURATMU, dan SURPAT. Pada pencarian *fuzzy*, kesalahan penghilangan huruf maupun pertukaran posisi tetap menghasilkan skor tinggi. Rangkuman contoh ditampilkan pada Tabel 1.

Tabel 1. Verifikasi contoh *Jaro-Winkler Distance*

No	Input	Target	Jenis kesalahan	Skor	Keputusan
1	AGENA	AGENDA	Penghilangan huruf	0,967	Diterima
2	PEMINJMAN	PEMINJAMAN	Penghilangan/transposisi	0,958	Diterima
3	SUART	SURAT	Transposisi	0,947	Diterima

Skor 0,967 pada AGENA-AGENDA dan 0,958 pada PEMINJMAN-PEMINJAMAN melampaui *threshold* 0,85 sehingga data tetap ditampilkan. Kasus SUART-SURAT memperoleh 0,947 dan menjadi rekomendasi utama dibandingkan kandidat lain. Hasil ini konsisten dengan literatur yang menunjukkan bahwa *Jaro-Winkler* berguna untuk koreksi ejaan dan variasi teks karena memberikan bobot tambahan pada kesamaan karakter awal (Halim et al., 2022). Penggunaan metode kemiripan pada domain kamus dan chatbot juga menunjukkan bahwa toleransi terhadap variasi input dapat meningkatkan peluang sistem mengenali maksud pengguna (Hawa et al., 2023).

Mekanisme hibrida pada Gambar 2 memperlihatkan dua lapisan yang saling melengkapi. *Trie* bekerja lebih dahulu sebagai jalur cepat untuk *prefix* yang benar dan sekaligus membatasi kandidat. *Jaro-Winkler* kemudian menilai kemiripan ketika kata kunci tidak identik. Pendekatan tersebut menghindari ketergantungan pada satu jenis pencocokan. Pengguna yang mengetik secara benar memperoleh respons autosuggestion, sedangkan pengguna yang melakukan *typo* tetap memiliki peluang menemukan data relevan.

DOI: https://doi.org/10.69693/ijmst.v4i2.12274
Lisensi: Creative Commons Attribution 4.0 International (CC BY 4.0)

3.3. Kinerja Kecepatan Pencarian

Pengujian kecepatan menunjukkan perbedaan yang besar antara pencarian manual dan sistem hibrida. Pada lima skenario manual, waktu pencarian berkisar antara 19 sampai 32 detik dengan rata-rata 25,0 detik. Waktu tersebut dipengaruhi posisi data dalam daftar dan banyaknya elemen yang harus dipindai secara visual. Rincian hasil disajikan pada Tabel 2.

Tabel 2. Hasil Pencarian Manual

No	Kategori	Target pencarian	Waktu
1	Agenda	Rapat Koordinasi Pejabat	28 detik
2	Buku	Hukum Perdata Islam di Indonesia	32 detik
3	Agenda	Sosialisasi Peraturan Baru	22 detik
4	Buku	Pengantar Studi Hukum Perdata	24 detik
5	Agenda	Rapat Evaluasi Kinerja	19 detik
		Rata-rata	25,0 detik

Pada sistem hibrida, lima *query* diuji masing-masing tiga kali. Waktu rata-rata per *query* berkisar antara 185,3 milidetik dan 305,0 milidetik. Rata-rata keseluruhan adalah 239,1 milidetik atau sekitar 0,24 detik. Dua skenario *typo*, yaitu "hukum pradata" dan "rapat koordinsi", tetap menghasilkan dokumen relevan dengan rata-rata 264,0 milidetik dan 196,7 milidetik. Rincian pengujian ditunjukkan pada Tabel 3.

Tabel 3. Waktu Respons Sistem Hibrida

No	Kategori	Kata kunci	Uji 1	Uji 2	Uji 3	Rata-rata (ms)
1	Agenda	rapat koordinasi	245	231	258	244,7
2	Buku	hukum perdata	189	175	192	185,3
3	Agenda	sosialisasi peraturan	312	298	305	305,0
4	Buku	hukum pradata (*typo*)	267	254	271	264,0
5	Agenda	rapat koordinsi (*typo*)	198	203	189	196,7
		Rata-rata				239,1

Perbandingan langsung menghasilkan percepatan sekitar 104,5 kali, dihitung dari 25.000 milidetik dibagi 239,1 milidetik. Hasil ini bukan berarti semua kondisi operasional selalu menghasilkan rasio yang identik karena performa dapat dipengaruhi ukuran data, spesifikasi server, koneksi, cache, dan beban pengguna. Namun, pada lingkungan uji penelitian, penurunan dari 25,0 detik menjadi 0,24 detik menunjukkan manfaat praktis yang jelas. Temuan ini mendukung studi mengenai *Trie* pada mini *search engine* yang menekankan kesesuaian struktur *prefix* untuk *autocomplete* cepat, sekaligus sejalan dengan kebutuhan sistem temu kembali yang mengutamakan efisiensi dan relevansi (Didin, 2021).

Tabel 4. Ringkasan peningkatan kecepatan

Metode	Waktu rata-rata	Peningkatan
Manual scrolling	25.000 ms	-
Algoritma hibrida	239,1 ms	104,5× lebih cepat

3.4. Hasil Pengujian *White box*

Pengujian *white box* terhadap 17 kasus menghasilkan tingkat kelulusan 100%. Seluruh kasus pada Controller, *Service* dan integrasi Python, algoritma *Trie*, *Jaro-Winkler Similarity*, *filtering* dan ranking, *fallback* dan *result processing*, serta logging berhasil memberikan keluaran sesuai rancangan. Rangkuman hasil disajikan pada Tabel 4.

Tabel 5. Rekapitulasi Pengujian *White box*

Komponen	Jumlah kasus	Lulus	Persentase
Controller	2	2	100%
Service dan integrasi Python	3	3	100%
Algoritma *Trie*	2	2	100%
Jaro-Winkler Similarity	3	3	100%
Filtering dan ranking	3	3	100%
Fallback dan *result processing*	3	3	100%
Logging	1	1	100%

DOI: https://doi.org/10.69693/ijmst.v4i2.12274
Lisensi: Creative Commons Attribution 4.0 International (CC BY 4.0)

Komponen	Jumlah kasus	Lulus	Persentase
Total	17	17	100%

Kelulusan 100% menunjukkan bahwa jalur logika yang diuji berfungsi secara konsisten pada skenario penelitian. Validasi request mampu menolak *query* kosong, indeks agenda dan buku dapat diambil dari basis data, layanan Python menerima parameter, *Trie* membentuk struktur dan menelusuri *prefix*, *Jaro-Winkler* menghitung *similarity*, *threshold* memisahkan kandidat diterima dan ditolak, hasil diurutkan berdasarkan skor, *fallback* dapat dijalankan, dan log pencarian tersimpan. Hasil ini penting karena kecepatan pencarian tanpa integritas alur internal dapat menghasilkan respons yang cepat tetapi tidak dapat dipercaya.

3.5. *Precision* dan *Recall*

Evaluasi *precision-recall* dilakukan pada 12 *query* yang mencakup pencarian agenda dan buku, termasuk kesalahan ketik seperti "kompetisi beraklak", "angaran reviu", "biaya plaksanaan", "laporan kenerja", dan "surat kterangan skit". Dari 76 data relevan pada ground truth, sistem menemukan seluruh 76 data sehingga FN bernilai 0. Sistem menampilkan total 85 hasil, terdiri atas 76 TP dan 9 FP. Dengan demikian, *Recall* mencapai 100% dan *Precision* 89,41%. Tabel 5 menunjukkan hasil per *query*.

Tabel 6. Hasil Pengujian *Precision* dan *Recall*

No	*Query* uji	Kategori	Relevan	Ditemukan	TP	FP	FN
1	undangan pelatihan	Agenda	4	5	4	1	0
2	kompetisi beraklak (*typo*)	Buku	12	14	12	2	0
3	permohonan pensiun	Agenda	3	3	3	0	0
4	angaran reviu (*typo*)	Buku	8	8	8	0	0
5	biaya plaksanaan (*typo*)	Agenda	4	4	4	0	0
6	magang universitas	Buku	12	12	12	0	0
7	Teknis	Agenda	3	4	3	1	0
8	laporan kenerja (*typo*)	Buku	5	5	5	0	0
9	pendapat mutasi	Agenda	2	3	2	1	0
10	Kejaksaan tinggi	Buku	10	12	10	2	0
11	surat kterangan skit (*typo*)	Agenda	4	5	4	1	0
12	Halu Oleo	Buku	9	10	9	1	0
	Total		76	85	76	9	0

Recall 100% berarti tidak ada data relevan yang terlewat pada 12 *query* uji. Hal ini sangat penting pada konteks administrasi karena kegagalan menemukan agenda atau buku yang sebenarnya tersedia dapat memaksa pengguna melakukan pencarian ulang. *Precision* 89,41% menunjukkan bahwa mayoritas hasil yang ditampilkan relevan, meskipun terdapat beberapa *false positive*. Trade-off tersebut wajar pada *fuzzy search*: *threshold* yang terlalu tinggi dapat meningkatkan *precision* tetapi berisiko menurunkan *recall*, sedangkan *threshold* terlalu rendah dapat memperluas hasil sekaligus meningkatkan *false positive*. Penggunaan *threshold* 0,85 pada penelitian ini menghasilkan keseimbangan yang memprioritaskan tidak terlewatnya data relevan.

Hasil tersebut juga memperlihatkan peran ranking. Meskipun beberapa *query* menghasilkan satu atau dua *false positive*, skor kemiripan dan bonus *Trie* menempatkan kecocokan yang lebih kuat pada urutan atas. Kajian *precision* dan *recall* pada sistem temu kembali menegaskan bahwa kedua metrik perlu dibaca bersama karena *precision* saja tidak menunjukkan kemampuan menemukan seluruh data relevan, sedangkan *recall* saja tidak mengukur kebersihan hasil (Auwiyah & Erlianti, 2025). Pada penelitian ini, kombinasi *Recall* 100% dan *Precision* 89,41% menunjukkan kemampuan temu kembali yang kuat pada dataset uji.

3.6. *User Acceptance Testing*

User Acceptance Testing melibatkan 10 responden sesuai tiga kelompok peran. Dua responden Super Admin memberikan nilai 86,67%, tiga responden Admin/Staff memberikan 86,00%, dan lima responden Pegawai memberikan 87,60%. Rata-rata keseluruhan adalah 86,76%, yang termasuk kategori sangat baik. Ringkasan hasil ditampilkan pada Tabel 6.

Tabel 7. Ringkasan *User Acceptance Testing*

Peran	Responden	Nilai	Kategori
Super Admin	2	86,67%	Sangat Baik
Admin/Staff	3	86,00%	Sangat Baik
Pegawai	5	87,60%	Sangat Baik
Rata-rata	10	86,76%	Sangat Baik

DOI: https://doi.org/10.69693/ijmst.v4i2.12274
Lisensi: Creative Commons Attribution 4.0 International (CC BY 4.0)

Kelompok Pegawai memperoleh nilai tertinggi, menunjukkan bahwa fungsi pencarian buku, autosuggestion, *fuzzy search*, peminjaman, dan pengembalian dapat digunakan dengan baik oleh pengguna akhir. Nilai Super Admin dan Admin/Staff juga berada pada kategori sangat baik, sehingga fungsi pengelolaan dan pencarian dinilai layak. Hasil UAT melengkapi pengujian teknis: sistem tidak hanya cepat dan akurat pada metrik komputasional, tetapi juga dapat diterima oleh pengguna yang menjalankan skenario nyata sesuai peran mereka.

Secara keseluruhan, hasil penelitian memperlihatkan hubungan yang konsisten antara rancangan algoritma dan hasil evaluasi. *Trie* menyediakan jalur pencarian cepat berbasis *prefix*, *Jaro-Winkler* menangani variasi penulisan, *threshold* dan ranking menjaga relevansi, sedangkan *fallback* mempertahankan ketersediaan hasil. Pendekatan hibrida sejalan dengan penelitian yang menggabungkan algoritma kemiripan untuk meningkatkan ketahanan pencocokan teks dan dengan kajian record matching *real time* yang menekankan optimasi antara kualitas pencocokan serta biaya komputasi. Namun, generalisasi hasil masih dibatasi oleh satu instansi, ukuran dataset 521 entri utama, 12 *query* evaluasi *precision-recall*, dan 10 responden UAT. Pengujian pada data lebih besar, variasi *query* yang lebih luas, serta lebih banyak pengguna diperlukan untuk menilai stabilitas performa pada skala berbeda.

4. Kesimpulan

Penelitian ini berhasil menghasilkan SIGAP, yaitu sistem informasi agenda dan perpustakaan berbasis *web* yang mengintegrasikan 21 data agenda dan 500 data buku pada satu platform dengan tiga peran pengguna. Fitur *smart searching* mengombinasikan *Trie* untuk autosuggestion berbasis *prefix* dan *Jaro-Winkler Distance* untuk fuzzy search dengan *threshold* 0,85. Hasil pengujian menunjukkan bahwa rancangan tersebut meningkatkan efisiensi pencarian secara nyata pada lingkungan uji, waktu rata-rata turun dari 25,0 detik pada metode manual menjadi 239,1 milidetik atau sekitar 0,24 detik, setara percepatan sekitar 104,5 kali. Seluruh 17 kasus *white box* dinyatakan lulus. Evaluasi 12 *query* menghasilkan *Recall* 100% dan *Precision* 89,41%, sehingga seluruh data relevan pada ground truth berhasil ditemukan dengan sebagian kecil false positive. *User Acceptance Testing* terhadap 10 responden memperoleh rata-rata 86,76% dan termasuk kategori sangat baik. Dengan demikian, kombinasi *Trie* dan *Jaro-Winkler Distance* layak digunakan untuk mendukung pencarian agenda dan buku yang cepat, toleran terhadap kesalahan penulisan, dan relevan bagi pengguna. Pengembangan berikutnya perlu menguji dataset yang lebih besar, lebih banyak variasi *query* dan responden, melakukan optimasi indeks untuk skala tinggi, serta memperkuat aspek keamanan, integrasi antarsistem, dan akses melalui perangkat bergerak.

Reference

Agni Afdilati. (2025). Peran Teknologi Informasi dalam Meningkatkan Efektivitas Administrasi Publik di Indonesia. *Jurnal Ilmu Komunikasi, Administrasi Publik Dan Kebijakan Negara*, *2*(3), 140–155. https://doi.org/10.62383/komunikasi.v2i3.491

Ahamed, I., Jahan, M., Tasnim, Z., Karim, T., Reza, S. M. S., & Hossain, D. A. (2021). Spell corrector for bangla language using norvig's algorithm and jaro-winkler distance. *Bulletin of Electrical Engineering and Informatics*, *10*(4), 1997–2005. https://doi.org/10.11591/EEI.V10I4.2410

Akulich, M., Savnik, I., Krnc, M., & Škrekovski, R. (2023). Multiset-Trie Data Structure. *Algorithms*, *16*(3), 1–28. https://doi.org/10.3390/a16030170

Alfiani, H., Kurnia Aditya, S., Lusa, S., Indra Sensuse, D., Adi Wibowo Putro, P., & Indriasari, S. (2024). *E-Government Issues in Developing Countries Using TOE and UTAUT Frameworks: A Systematic Review*. https://doi.org/10.30589/pgr

Asari, F. L., Meimaharini, R. S., & Khotimah, T. (2025). Implementasi Sistem Perpustakaan Berbasis Web untuk Meningkatkan Efisiensi Layanan Peminjaman dan Pengguna. *Bit-Tech*, *7*(3), 770–778. https://doi.org/10.32877/bt.v7i3.2185

Auwiyah, W., & Erlianti, G. (2025). Recall and Precision Temu Kembali Informasi Pada Repository Universitas Negeri Padang. *INNOVATIVE: Journal Of Social Science Research*, *5*, 1796–1804.

Didin, J. (2021). Implementasi Algoritma Trie pada Aplikasi Pencarian Kontak Telepon dengan Autocomplete. *Jurnal Multimedia Dan Teknologi Informasi*, *3*(1), 42–47.

Halim, J. T., Hiryanto, L., & Lewenusa, I. (2022). Jurnal Ilmu Komputer dan Sistem Informasi PENDETEKSIAN KESALAHAN KETIK DENGAN DAMERAU-LEVENSHTEIN DISTANCE DAN TRIE. *Jurnal Ilmu Komputer Dan Sistem Informasi*, *10*(2), 1–6. https://kvzaytsev.github.io/trie-visualizer/.

Hawa, S., Shareef, M., Sulaiman, R. A., Samad, A., & Basari, H. (2023). The Hybrid Jaro-Winkler and Manhattan Distance using Dissimilarity Measure for Test Case Prioritization Approach. *IJACSA) International Journal of Advanced Computer Science and Applications*, *14*(11), 2023. www.ijacsa.thesai.org

honkapohja, A., Thaisen, J., & Nøklestad, A. (2023). A search tool based on language modelling developed for The Index of Middle English Prose. *Open Research Europe*, *3*(197), 1–18. https://doi.org/10.12688/openreseurope.16590.1

Jayadianti, H., Santosa, B., Cahyaning, J., Saifullah, S., & Drezewski, R. (2023). Essay auto-scoring using N-Gram and Jaro Winkler based Indonesian Typos. *MATRIK : Jurnal Manajemen, Teknik Informatika Dan Rekayasa Komputer*, *22*(2), 325–338. https://doi.org/10.30812/matrik.v22i2.2473

Mhawi, D. N., Oleiwi, H. W., Saeed, N. H., & Al-Taie, H. L. (2022). An Efficient Information Retrieval System Using Evolutionary Algorithms. *Network*, *2*(4), 583–605. https://doi.org/10.3390/network2040034

Nita, Y. P., Isma Nurmalia, & Rabini Sayyidati. (2025). Web-Based Administrative Completeness Information System At Badan Pusat Statistik Of Tanah Laut Regency. *Engineering: Journal of Mechatronics and Education*, *2*(1), 1–11.

Rasna, Dewi, N. W. E. R., Bani, A. U., Lamsir, S., & Nugroho, F. (2025). Application Dictionary of Scientific Plants and Animals Android-Based Algorithm Using Jaro Winkler Distance. *International Journal of Engineering, Science and Information Technology*, *5*(2), 314–319.

DOI: https://doi.org/10.69693/ijmst.v4i2.12274
Lisensi: Creative Commons Attribution 4.0 International (CC BY 4.0)

Rozinek, O., Marek, J., Panuš, J., & Mareš, J. (2025). Real-Time Fuzzy Record-Matching Similarity Metric and Optimal Q-Gram Filter. *Algorithms*, *18*(3). https://doi.org/10.3390/a18030150

Scoulas, J., & De Groote, S. (2024). Use and Importance of Library Resources to Support Faculty Research and Productivity. *College & Research Libraries*, *85*(5), 673–694. https://doi.org/10.5860/crl.85.5.669

Shamitha, K. (2025). Mini Search Engine using Python and Trie Data Structure. *International Journal for Research in Applied Science and Engineering Technology*, *13*(7), 1722–1726. https://doi.org/10.22214/ijraset.2025.73263

Shamsitdinova M, Khashimova D, Niyazova N, Nasirova U, & Khikmatov N. (2024). Harnessing AI for Enhanced Searching in Digital Libraries: Transforming Research Practices. *Indian Journal of Information Sources and Services*, *14*(3), 102–109. https://doi.org/10.51983/ijiss-2024.14.3.14

Sugiarso, T., Sulistyowati, Y., & Widayati, T. (2025). PERANCANGAN SISTEM PENCATATAN AGENDA BERBASIS WEB UNTUK OPTIMALISASI PENGELOLAAN KEGIATAN PADA UNIVERSITAS RESPATI INDONESIA. *Jurnal Teknologi Informasi*, *11*(1), 9–18.

Suwarningsih, W., & Nuryani. (2024). Generate fuzzy string-matching to build self attention on Indonesian medical-chatbot. *International Journal of Electrical and Computer Engineering*, *14*(1), 819–829. https://doi.org/10.11591/ijece.v14i1.pp819-829

Wijaya, D. P., Murti, L. D., & Rachman, M. R. (2022). Recall dan Precision pada Online Public Access Catalog (OPAC) Dinas Arsip dan Perpustakaan Kota Bandung. *VISI PUSTAKA*, *24*(1), 81–91.

Yakub, H., Daniawan, B., Wijaya, A., & Damayanti, L. (2024). Sistem Informasi E-Commerce Berbasis Website Dengan Metode Pengujian User Acceptance Testing. *JSITIK: Jurnal Sistem Informasi Dan Teknologi Informasi Komputer*, *2*(2), 113–127.

Yulianto, M. A., & Nurhasanah, N. (2021). The Hybrid of Jaro-Winkler and Rabin-Karp Algorithm in Detecting Indonesian Text Similarity. *Jurnal Online Informatika*, *6*(1), 88–95. https://doi.org/10.15575/join.v6i1.640

DOI: https://doi.org/10.69693/ijmst.v4i2.12274
Lisensi: Creative Commons Attribution 4.0 International (CC BY 4.0)