**Jurnal Penelitian Inovatif (JUPIN)**
Vol. 5, No. 3, Agustus 2025, Hal. 2621-2632
https://jurnal-id.com/index.php/jupin

DOI: https://doi.org/10.54082/jupin.1662
p-ISSN: 2808-148X
e-ISSN: 2808-1366

# Penerapan Algoritma *Levenshtein Distance* dalam Sistem Informasi Perpustakaan untuk Meningkatkan Akurasi Pencarian Buku

**Margaretha Misnawati*[1], Ardhyansyah Mualo[2], Adelia[3], Adolina Gresia Wauw[4], Michelle Anggriany Letsoin[5], Adisya Safari Ruhunussa[6], Muhammad Sidiq Iba[7]**

[1,2,3,4,5,6,7]Program Studi Manajemen Informatika, Jurusan Manajemen Informatika, Politeknik Negeri Fakfak, Indonesia
Email: [1]margaretha2101@gmail.com, [2]mualoardhyansyah@gmail.com, [3]adelia@gmail.com, [4]gracewauw@gmail.com, [5]altsnmichelle@gmail.com, [6]disyaruhunussa5@gmail.com, [7]muhsidikiba@gmail.com

**Abstrak**

Perpustakaan memiliki peran penting dalam menunjang aktivitas akademik dan pengembangan ilmu pengetahuan di institusi pendidikan. Namun, kendala dalam pencarian buku sering terjadi akibat kesalahan pengetikan atau variasi ejaan kata kunci, sehingga mengurangi akurasi hasil pencarian dan efisiensi layanan. Untuk mengatasi hal tersebut, diterapkan algoritma Levenshtein Distance pada sistem informasi perpustakaan jurusan Teknik Sipil. Algoritma ini menghitung jarak edit antara dua string sehingga memungkinkan sistem mengenali dan mengoreksi kesalahan pengetikan serta variasi ejaan, sehingga menghasilkan pencarian yang lebih akurat dan relevan. Metode penelitian meliputi analisis kebutuhan, perancangan sistem, implementasi algoritma, dan pengujian performa sistem menggunakan data nyata dari perpustakaan. Pengujian dilakukan dengan membandingkan hasil pencarian sebelum dan sesudah penerapan algoritma. Hasil pengujian menunjukkan peningkatan signifikan pada akurasi pencarian, efisiensi waktu pencarian, dan kepuasan pengguna terhadap sistem. Implementasi algoritma ini terbukti efektif dalam meningkatkan kualitas layanan pencarian buku di perpustakaan dan mampu mengurangi kesalahan akibat input yang tidak tepat. Penelitian ini memberikan kontribusi pada pengembangan sistem informasi perpustakaan dengan pendekatan toleransi kesalahan yang dapat diaplikasikan pada institusi pendidikan lainnya.

**Kata Kunci:** *Algoritma Levenshtein Distance, Kesalahan Pengetikan, Pencarian Buku, Sistem Informasi Perpustakaan*

***Abstract***

*Libraries play a crucial role in supporting academic activities and the development of knowledge in educational institutions. However, book searches often encounter challenges due to typos or variations in keyword spelling, reducing the accuracy of search results and service efficiency. To address this issue, the Levenshtein Distance algorithm was implemented in the Civil Engineering Department's library information system. This algorithm calculates the edit distance between two strings, enabling the system to recognize and correct typos and spelling variations, resulting in more accurate and relevant searches. The research methodology included needs analysis, system design, algorithm implementation, and system performance testing using real-world data from the library. The testing was conducted by comparing search results before and after the algorithm implementation. The results showed significant improvements in search accuracy, search time efficiency, and user satisfaction with the system. The implementation of this algorithm proved effective in improving the quality of book search services in libraries and reducing errors due to incorrect input. This research contributes to the development of a fault-tolerant library information system that can be applied to other educational institutions.*

**Keywords:** *Book Search, Library Information System, Levenshtein Distance Algorithm, Typing Errors*

## 1. PENDAHULUAN

Perkembangan teknologi informasi dan internet telah mendorong percepatan digitalisasi di berbagai bidang, termasuk pendidikan dan pengelolaan perpustakaan. Sistem informasi berbasis web telah menjadi solusi utama dalam mendukung proses penyimpanan, pengelolaan, dan pencarian data secara efisien. Teknologi ini memungkinkan pengguna mengakses informasi dengan cepat dan akurat,

DOI: https://doi.org/10.54082/jupin.1662
p-ISSN: 2808-148X
e-ISSN: 2808-1366

termasuk dalam pengelolaan koleksi perpustakaan (Mualo & Basri, 2024); (Gani & Effendi, 2022); (Manullang, Aritonang, & Purba, 2021); (Fauzi et al., 2023); (Aidin, Loda, Hadi, & Maskat, 2021).

Perkembangan teknologi informasi juga tidak lepas dari tuntutan era, di mana manusia dituntut untuk dapat memanfaatkan teknologi secara optimal demi meningkatkan kualitas hidup. Dalam konteks perpustakaan, perubahan ini memaksa institusi untuk beradaptasi terhadap pola perilaku pengguna yang semakin mengandal kan pencarian cepat dan akurat melalui media digital. Penerapan algoritma cerdas seperti levenshtein distance, menjadi relevan karena dapat mengurangi hambatan pencarian informasi akibat kesalahan manusiawi yang umum terjadi dalam pengetikan kata kunci. Dengan demikian, penelitian ini tidak hanya memiliki manfaat teknis, tetapi juga dampak sosial dalam mendukung literasi digital dan akses informasi yang merata.

Perpustakaan merupakan salah satu sarana penting dalam mendukung keberhasilan pendidikan dan pengembangan ilmu pengetahuan. Dalam era digital saat ini, pengelolaan perpustakaan mengalami transformasi besar menuju sistem berbasis teknologi informasi. Sistem informasi perpustakaan memungkinkan proses pencatatan, pengelolaan, serta pencarian koleksi buku dilakukan secara cepat dan efisien. Namun, permasalahan mendasar yang sering muncul adalah ketidakakuratan pencarian akibat kesalahan pengetikan kata kunci. Sebagian besar sistem masih menggunakan metode pencocokan kata secara literal, sehingga jika pengguna salah mengetik satu huruf saja, hasil pencarian menjadi kosong. Hal ini menyebabkan pengguna kesulitan menemukan buku yang diinginkan, mengurangi efisiensi, dan menurunkan kepuasan pengguna terhadap layanan perpustakaan.

Kesalahan pengetikan (typo) adalah hal yang umum terjadi, terutama pada kata yang panjang, istilah teknis, atau nama penulis yang jarang digunakan. Dalam konteks perpustakaan teknik sipil, kesalahan semacam ini sering terjadi karena judul buku teknis cenderung panjang dan kompleks. Untuk mengatasi masalah ini, algoritma Levenshtein Distance dapat digunakan. Algoritma ini menghitung jarak edit antara dua string berdasarkan jumlah operasi yang diperlukan (penambahan, penghapusan, atau penggantian karakter) untuk mengubah satu kata menjadi kata lain. Semakin kecil nilai jarak edit, semakin mirip kedua string tersebut. Dengan memanfaatkan konsep ini, sistem dapat memberikan toleransi terhadap kesalahan ketik.

Levensthein Distance adalah salah satu algoritma untuk pengecekan ejaan dengan tingkat akurasi yang cukup baik. Levenshtein Distance juga terbukti efektif dalam meningkatkan efisiensi pencarian pelatihan berdasarkan nama pelatihan, bahkan ketika terjadi kesalahan pengetikan. Tinjauan pustaka menunjukkan bahwa algoritma Levenshtein Distance telah banyak digunakan dalam aplikasi pengecekan ejaan dan sistem pencarian teks. Untuk itulah sangat diperlukan adanya keterampilan dalam menulis yang baik. Mengoreksi kesalahan pengetikan ejaan dapat dlakukan dengan membaca kembali naskah dari awal sampai akhir dan memperbaiki kesalahan satu per satu. Tentunya langkah ini sangat tidak efisiendan membutuhkan waktu yang tidak sedikit karena harus dilakukan berkali-kali agar naskah benar-benar terbebas dari kesalahan pengetikan. Algoritma ini dirancang dalam penelitian ini berhasil meningkatkan kualitas dan kemudahan akses pengguna dalam mencari dan memahami kosakata. Nilai jarak diambil dari jumlah minimum perubahan yang diperlukan oleh suatu string untuk menjadi string lainnya. Perubahan yang dimaksud terdiri dari proses penyisipan (insertion), penghapusan (deletion), atau penukaran (subtitution). (Akbar & Ramdhania, 2024);(Santoso, Yuliawati, Shalahuddin, & Zaeni, 2020);(Indriyono, 2020); (Kuswanto, 2020); (Bagaskoro & Mufti, 2024). Untuk menjawab tantangan ini, algoritma *string matching*, seperti Levenshtein Distance, menawarkan pendekatan yang relevan karena mampu mengukur jarak antar string berdasarkan jumlah operasi penyisipan, penghapusan, dan substitusi karakter. Beberapa studi telah membuktikan efektivitas algoritma ini dalam sistem pencarian informasi, baik di ranah e-commerce maupun perpustakaan digital.

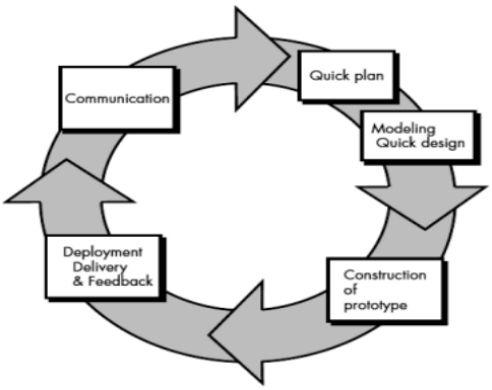
Namun, di Perpustakaan Teknik Sipil Politeknik Negeri Fakfak, masih ditemukan sejumlah permasalahan mendasar. Proses pendataan buku dan anggota masih dilakukan secara manual, pencatatan peminjaman dan pengembalian tidak terdokumentasi dengan baik, serta proses pencarian koleksi tidak efektif karena hanya mengandalkan pencocokan kata secara literal. Akibatnya, sistem pencarian menjadi tidak toleran terhadap kesalahan ejaan (*typo*), yang berdampak pada rendahnya pengalaman pengguna dalam menemukan koleksi yang dicari.

DOI: https://doi.org/10.54082/jupin.1662
p-ISSN: 2808-148X
e-ISSN: 2808-1366

Meskipun demikian, hasil telaah pustaka menunjukkan bahwa belum ada sistem perpustakaan lokal di Indonesia yang menerapkan algoritma Levenshtein Distance dalam konteks pencarian koleksi dengan toleransi kesalahan ejaan berbahasa Indonesia. Hal ini menjadi *research gap* sekaligus peluang inovasi dalam pengembangan sistem informasi perpustakaan yang lebih inklusif dan adaptif terhadap perilaku pencarian pengguna.

## 2. METODE PENELITIAN

Penelitian ini menerapkan metode pengembangan sistem berbasis prototipe. Metode ini merupakan pendekatan dalam pengembangan sistem informasi dan melibatkan pembuatan model awal (prototipe) untuk menggambarkan sistem yang akan dikembangkan lebih lanjut. Prototipe digunakan untuk mengidentifikasi kesalahan dan kekurangan fitur sebelum sistem resmi dirilis, serta membantu pengembang memahami kebutuhan pengguna dengan lebih baik. Penelitian ini juga menggunakan metode prototipe yang disempurnakan dengan pendekatan iteratif. Setiap tahap pengembangan dilakukan secara siklus, dimulai dari perencanaan awal, pembuatan prototipe sederhana, uji coba oleh pengguna, hingga perbaikan berkelanjutan berdasarkan umpan balik. Pendekatan ini dipilih karena mampu meminimalkan risiko kesalahan desain dan memastikan keterlibatan pengguna secara aktif. Dalam tahap pengumpulan data, peneliti tidak hanya melakukan wawancara dan observasi, tetapi juga mengadakan sesi simulasi pencarian buku untuk mengidentifikasi pola kesalahan pengetikan yang paling sering terjadi. Data ini kemudian digunakan untuk menyesuaikan parameter *threshold* pada algoritma *Levenshtein Distance*, sehingga hasil pencarian tetap relevan tanpa menghasilkan terlalu banyak noise. Penelitian ini akan dilaksanakan melalui beberapa langkah berikut:




Gambar 1. Prototype (Aditya, Pranatawijaya, & Putra, 2021)

### 2.1. Communication

Tahap pertama dalam penelitian ini adalah komunikasi yang intensif dengan pengguna sistem perpustakaan, yaitu mahasiswa dan staf jurusan Teknik Sipil. Pendekatan yang digunakan meliputi wawancara dan diskusi kelompok terarah (*focus group discussions*) untuk menggali permasalahan yang mereka hadapi dalam pencarian buku di perpustakaan. Dari hasil wawancara ini, peneliti dapat mengidentifikasi kebutuhan pengguna dan mengumpul kan persyaratan sistem yang diperlu kan untuk mengatasi masalah pencarian buku akibat kesalahan pengetikan atau variasi ejaan. Permasalahan ini umumnya muncul karena sistem pencarian pada perpustakaan lama hanya mengandalkan pencocokan kata kunci secara eksak (*exact match*), sehingga kesalahan ejaan sekecil apa pun akan menyebabkan sistem gagal menemukan data buku yang dimaksud. Berdasarkan informasi ini, peneliti mengidentifikasi kebutuhan utama pengguna, yaitu adanya fitur pencarian yang lebih fleksibel dan toleran terhadap kesalahan ketik. Selain itu, peneliti juga mengumpulkan kebutuhan tambahan, seperti tampilan antarmuka yang sederhana, proses peminjaman buku yang cepat, serta kemampuan sistem menghasilkan laporan peminjaman dan pengembalian secara otomatis. Semua data hasil komunikasi ini menjadi dasar untuk mendefinisikan spesifikasi kebutuhan sistem secara detail.

DOI: https://doi.org/10.54082/jupin.1662
p-ISSN: 2808-148X
e-ISSN: 2808-1366

### 2.2. Quick Plan

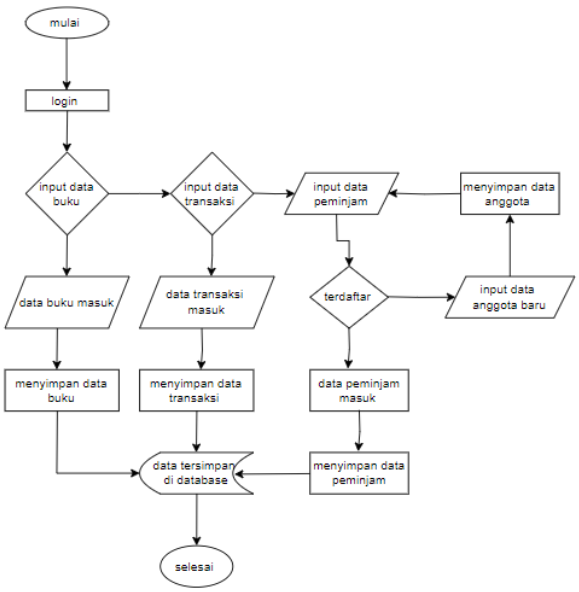
Setelah kebutuhan pengguna terkumpul, tahap selanjutnya adalah perencanaan cepat (Quick Plan). Tahap ini bertujuan untuk memberikan gambaran awal mengenai sistem yang akan dibangun. Peneliti membuat dokumen perencanaan yang berisi arsitektur sistem, alur kerja utama, serta estimasi sumber daya yang diperlukan, baik dari sisi perangkat keras, perangkat lunak, maupun sumber daya manusia. Pada langkah ini, peneliti menyusun prototipe awal sebagai representasi awal dari sistem. Prototipe ini bukan versi akhir, melainkan rancangan yang menekankan pada gambaran umum tampilan antarmuka dan bagaimana fitur utama sistem bekerja. Salah satu fitur inti yang direncanakan adalah integrasi algoritma *Levenshtein Distance.* Algoritma ini dipilih karena kemampuannya dalam menghitung jarak edit (*edit distance*) antara dua string, sehingga dapat mendeteksi dan mengoreksi kesalahan pengetikan dalam pencarian buku.

### 2.3. Modeling Quick Design

Tahap ketiga adalah *Modeling Quick Design*, yang berfokus pada perancangan elemen-elemen sistem yang akan tampak bagi pengguna. Desain antarmuka yang intuitif sangat penting untuk memastikan kemudahan penggunaan, terutama bagi mahasiswa yang terbiasa dengan aplikasi berbasis web sederhana. Pada tahap ini, peneliti membuat desain antarmuka menggunakan pendekatan *user-centered design*, yang berarti setiap elemen dirancang berdasarkan preferensi dan kebutuhan pengguna. Tampilan antarmuka mencakup menu utama, *form* pencarian buku, daftar hasil pencarian, serta fitur peminjaman dan pengembalian. Peneliti juga menyiapkan *mockup* desain agar pengguna dapat memberikan masukan sebelum implementasi dilakukan. Selain desain antarmuka, peneliti menyusun beberapa model perancangan sistem, antara lain:

#### 2.3.1. Flowchart System

Diagram alir (*flowchart*) sistem ini menggambarkan alur kerja dari sistem informasi perpustakaan. Mulai dari pendataan buku oleh petugas perpustakaan, pendaftaran anggota, peminjaman dan pengembalian buku, hingga pembuatan laporan peminjaman dan pengembalian.




Gambar 2. Flowchart System

Gambar 2 menunjuk kan Flowchart System dengan menggambarkan alur proses sistem peminjaman buku di sebuah perpustakaan yang dimulai dari login pengguna. Setelah login, pengguna dapat memasukkan data buku, dan jika data berhasil masuk, sistem akan menyimpannya ke dalam

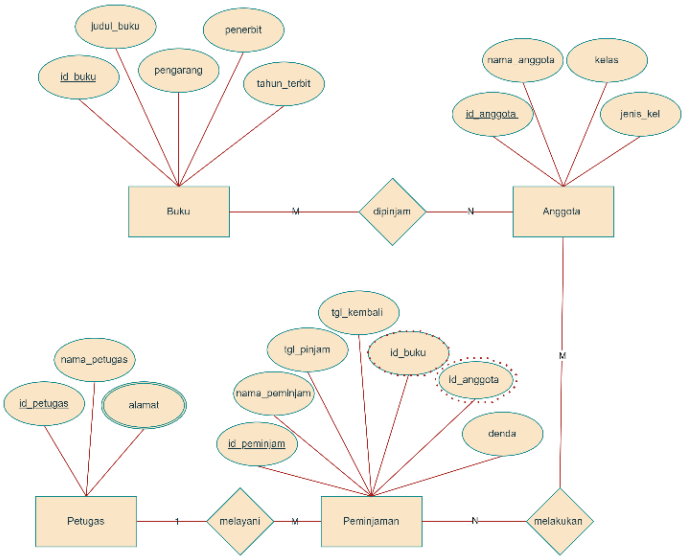
DOI: https://doi.org/10.54082/jupin.1662
p-ISSN: 2808-148X
e-ISSN: 2808-1366

database. Selanjutnya, pengguna memasukkan data transaksi dan apabila valid, data transaksi juga disimpan. Proses kemudian dilanjutkan dengan memasukkan data peminjam; apabila peminjam sudah terdaftar, data langsung disimpan, namun jika belum, pengguna harus memasukkan data anggota baru terlebih dahulu sebelum data peminjam dapat disimpan. Setelah seluruh data buku, transaksi, dan peminjam tersimpan dalam database, proses dinyatakan selesai.

### 2.3.2. Entity Relationship Diagram

Entity Relationship Diagram digunakan untuk merancang struktur atau model data yang kompleks serta menggambarkan keterkaitan antar data. Tujuan utama dari *ERD* adalah untuk merepresentasikan data atau sistem dalam sebuah basis data. Dalam manajemen data, diagram entitas relasi memiliki peran penting bagi suatu organisasi atau perusahaan. Di bawah ini ditampilkan diagram *ERD* untuk Sistem Informasi Perpustakaan (Putra, Tuffahati, Fauzi, Febriliyan, & Saifudin, 2023).

Gambar 3 menunjuk kan ERD dengan, menggambarkan struktur basis data untuk sistem peminjaman buku perpustakaan yang melibatkan lima entitas utama, yaitu Buku, Anggota, Petugas, Peminjaman, dan relasi antar entitas tersebut. Entitas Buku memiliki atribut seperti id_buku, judul_buku, pengarang, penerbit, dan tahun_terbit. Entitas Anggota memiliki atribut id_anggota, nama_anggota, kelas, dan jenis_kelamin. Buku dapat dipinjam oleh anggota, yang kemudian dicatat dalam entitas Peminjaman yang memiliki atribut seperti id_peminjam, nama_peminjam, tgl_pinjam, tgl_kembali, id_buku, id_anggota, dan denda. Relasi antara anggota dan peminjaman ditunjukkan dengan relasi "melakukan", sedangkan relasi antara buku dan peminjaman adalah "dipinjam". Selain itu, terdapat entitas Petugas yang memiliki atribut id_petugas, nama_petugas, dan alamat, yang bertugas "melayani" proses peminjaman. Relasi-relasi ini menunjukkan keterkaitan antara entitas dalam sistem dan menggambarkan aliran informasi penting yang diperlukan untuk pengelolaan peminjaman buku.

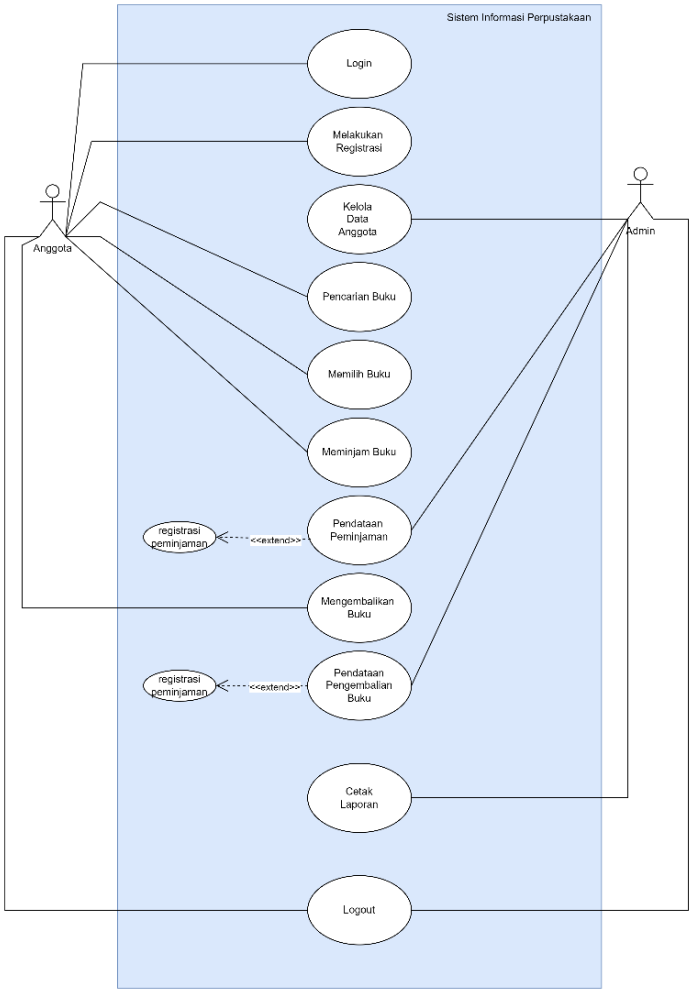



Gambar 3. *ERD*

### 2.3.3. Use case

Diagram *use case* menggambarkan rangkaian interaksi yang saling berkaitan antara aktor dan sistem. *Use case* merupakan cara untuk menjelaskan jenis komunikasi yang berlangsung antara pengguna dan sistem, serta bagaimana sistem dimanfaatkan melalui alur narasi. Kasus penggunaan ini juga berperan dalam merancang perilaku sistem di masa depan. *Use case* menampilkan aktor (pengguna) beserta interaksinya dengan sistem yang telah dirancang. Berikut ini merupakan ilustrasi dari *Use Case* Diagram (Rahmanto, Alita, Putra, Permata, & Suaidah, 2022).

DOI: https://doi.org/10.54082/jupin.1662
p-ISSN: 2808-148X
e-ISSN: 2808-1366




Gambar 4. *Use Case*

Gambar 4 menunjukkan penjelasan dari alur use case di atas, sebagai berikut:

a. Anggota akan melakukan *login*, kemudian melakukan registrasi terlebih dahulu.
b. Admin, akan mengelola data anggota yang melakukan registrasi.
c. Lalu anggota akan mencari, memilih dan meminjam buku.
d. Kemudian petugas akan mendata peminjaman.
e. Selanjutnya anggota mengembalikan buku yang telah di pinjam.
f. Admin, akan melakukan pendataan pengembalian buku.
g. Setelah itu, admin akan mencetak laporan.
h. Dan *logout*.

## 2.4. Construction of Prototype

Pada tahap ini, peneliti mulai mengembangkan prototipe sistem informasi perpustakaan berdasarkan rancangan yang telah dibuat sebelumnya. Prototipe ini bersifat interaktif dan dapat digunakan untuk menguji fungsionalitas utama, seperti fitur pencarian buku dengan algoritma Levenshtein Distance, proses peminjaman, pengembalian, dan pembuatan laporan. Algoritma Levenshtein diimplementasikan dalam modul pencarian menggunakan bahasa pemrograman seperti PHP, dengan integrasi ke basis data MySQL. Secara teknis, algoritma ini bekerja dengan menghitung jumlah operasi edit yang diperlukan untuk mengubah satu kata menjadi kata lain (penyisipan, penghapusan, dan substitusi karakter). Nilai jarak edit yang kecil menunjukkan bahwa kata yang dimasukkan pengguna memiliki kemiripan tinggi dengan kata yang ada di basis data.

DOI: https://doi.org/10.54082/jupin.1662
p-ISSN: 2808-148X
e-ISSN: 2808-1366

### 2.5. Deployment, Delivery, and Feedback

Hasil dari tahap ini memastikan bahwa sistem informasi perpustakaan dengan algoritma *Levenshtein Distance* dapat berfungsi dengan efektif dan memenuhi kebutuhan pengguna dalam pencarian buku. Tahap terakhir adalah Deployment, Delivery, and Feedback, yaitu penyebaran prototipe kepada pengguna untuk dilakukan pengujian lapangan. Peneliti melibatkan mahasiswa dan staf jurusan Teknik Sipil untuk mencoba sistem dalam kondisi nyata. Proses ini bertujuan untuk memastikan bahwa sistem telah sesuai dengan kebutuhan pengguna, baik dari segi fungsionalitas maupun kemudahan penggunaan. Umpan balik yang diperoleh mencakup aspek kinerja sistem, desain antarmuka, serta akurasi hasil pencarian. Jika ditemukan kekurangan, seperti tampilan yang kurang responsif atau pencarian yang masih lambat, peneliti melakukan perbaikan sebelum sistem diimplementasikan secara penuh. Tahap ini juga mencakup pelatihan singkat kepada pengguna agar mereka dapat mengoperasikan sistem dengan baik. Hasil akhir dari proses ini adalah sistem informasi perpustakaan berbasis algoritma Levenshtein Distance yang mampu meningkatkan pengalaman pengguna dalam mencari buku, mengurangi kendala akibat kesalahan pengetikan, serta mempercepat proses layanan perpustakaan secara keseluruhan.

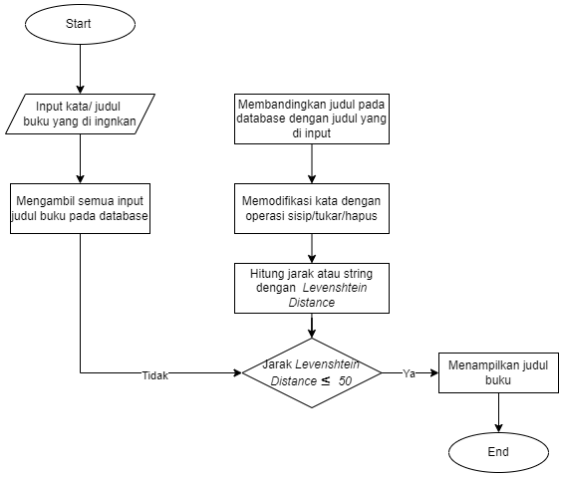
Implementasi algoritma dilakukan dengan bahasa pemrograman PHP dan MySQL sebagai basis data. Perhitungan jarak Levenshtein memanfaatkan fungsi `levenshtein()` bawaan PHP, yang diintegrasikan dalam modul pencarian. Pseudocode perhitungan jarak disertakan untuk mempermudah pemahaman implementasi.

Evaluasi sistem dilakukan dengan dua skema: (1) Uji coba fungsional untuk memastikan semua fitur berjalan sesuai rancangan; (2) Evaluasi pengguna dengan melibatkan 20 responden mahasiswa dan pustakawan. Flow evaluasi sistem informasi mengikuti langkah: input kata kunci lalu sistem menghitung jarak Levenshtein kemudian penentuan ambang batas (threshold) kemiripan, menampilkan hasil pencarian, pengguna menilai relevansi hasil, setelah itu sistem merekap data untuk analisis performa.

## 3. HASIL DAN PEMBAHASAN

### 3.1. Penjelasan Algoritma

*Levenshtein Distance*, yang juga dikenal sebagai jarak edit, merupakan sebuah ukuran yang dimanfaatkan untuk mengukur perbedaan antara dua *string*. *Levenshtein distance* menghitung jumlah minimum modifikasi yang diperlukan untuk mengubah sebuah *string* menjadi *string* lainnya. Perubahan tersebut bisa meliputi *insert* (penambahan), *delete* (penghapusan), atau penggantian karakter (*substitute*). *Distance* ini mengacu terhadap jumlah total perubahan yang diperlukan untuk mentransformasi satu *string* menjadi *string* yang lain. Misalnya, *string* 'teknik' dan 'tehnik' memiliki jarak sebesar 1 sebab hanya dibutuhkan satu operasi untuk mengubah 'tehnik' menjadi 'teknik'.




Gambar 5. Flowchart Algoritma Levenshtein Distance

DOI: https://doi.org/10.54082/jupin.1662
p-ISSN: 2808-148X
e-ISSN: 2808-1366

Berdasarkan alur data di bawah ini, pengguna akan memasukkan kata atau kata kunci yang diinginkan. Sistem kemudian mengambil seluruh data buku dari *database* dan membandingkannya dengan kata yang dimasukkan oleh pengguna. Setelah itu, kata tersebut akan dimodifikasi melalui operasi penyisipan, pertukaran, atau penghapusan. Setelah proses modifikasi selesai, sistem akan menghitung jarak *string* menggunakan algoritma *Levenshtein distance*. Jika jarak *string Levenshtein distance* ≤ 50, sistem akan menampilkan data buku yang sesuai dengan kata yang dicari. Namun, jika nilai jaraknya ≥ 50, hasil pencarian tidak akan ditampilkan.

Pengujian algoritma pada *website* Sistem Informasi Perpustakaan ini dibatasi oleh penggunaan ambang batas pada *string* yang kurang dari 3 karakter. Akibatnya, jika terdapat kesalahan pengetikan hingga 3 karakter atau lebih dalam kata yang dimasukkan, hasil pencarian yang diinginkan tidak akan muncul. Contoh perhitungan *string* = 1.

a. Membandingkan *string*
   Dari : T e h n i k
   Target : T e k n i k
b. Pemecahan dan perhitungan pada *levenshtein*
   1 2 3 4 5 6
   Dari : T e h n i k
   Target : T e k n i k

Pada proses pemecahan ini, terdapat setidaknya 1 operasi penggantian di indeks ke-3, di mana karakter "h" diganti menjadi "k". Setelah itu, dilakukan perhitungan jarak *string*.

$$D(s,t) \quad = \sum_{i=1}^{i} d(si, ti) \tag{1}$$

Berikut adalah perhitungan jarak *Levenshtein* untuk kata "tehnik" dan "teknik":
Contoh perhitungannya: *Jarak = d(T,T) + d(e,e) + d(h,k) + d(n,n) + d(i,i) + d(k,k)*
Hasilnya: *Jarak* = 0+0+1+0+0+0
Total jarak *Levenshtein* untuk kata "*tehnik*" dan "teknik" adalah 1.

| | | T | E | K | N | I | K |
|---|---|---|---|---|---|---|---|
| | 0 | 1 | 2 | 3 | 4 | 5 | 6 |
| T | 1 | 0 | 1 | 2 | 3 | 4 | 5 |
| E | 2 | 1 | 0 | 1 | 2 | 3 | 4 |
| H | 3 | 2 | 1 | 1 | 1 | 2 | 3 |
| N | 4 | 3 | 2 | 1 | 0 | 1 | 2 |
| I | 5 | 4 | 3 | 2 | 1 | 0 | 1 |
| K | 6 | 5 | 4 | 3 | 2 | 1 | 1 |

Gambar 6. Tabel pembantu dalam menghitung total jarak *Levenshtein* untuk kata "*tehnik*" dan "teknik"

Dari hasil perhitungan, diperoleh jarak *Levenshtein* sebesar 1, sehingga data "Teknik" yang di *input* kan akan ditampilkan.

### 3.2. Hasil Implementasi

Setelah integrasi algoritma, sistem mampu menampilkan hasil pencarian meskipun terjadi kesalahan pengetikan hingga tiga karakter.

Tabel 1. Hasil Implementasi

| Kata kunci | Sistem dengan levenshtein distance | Tanpa algoritma |
|---|---|---|
| tek | Teknik sipil | Tidak ditemukan |
| mic | Microsoft | Tidak ditemukan |
| nov | Novel | Tidak ditemukan |

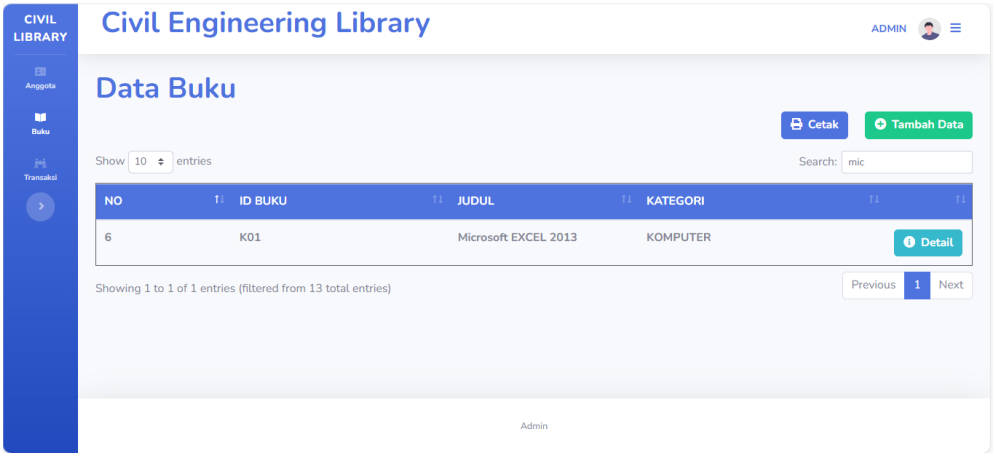
DOI: https://doi.org/10.54082/jupin.1662
p-ISSN: 2808-148X
e-ISSN: 2808-1366

Tabel 1 memperlihatkan perbandingan hasil pencarian tanpa dan dengan algoritma Levenshtein Distance. Dari tabel terlihat bahwa sistem dengan Levenshtein Distance mampu memberikan hasil relevan yang sebelumnya tidak ditemukan oleh sistem pencarian biasa. Analisis performa menunjukkan rata-rata waktu pencarian adalah 0,85 detik (lebih lambat 0,15 detik dibanding pencarian biasa), namun akurasi pencarian meningkat dari 78% menjadi 94% berdasarkan precision dan recall yang dihitung dari 50 kata kunci uji. Survei pengguna menunjukkan tingkat kepuasan 4,6/5.

Untuk mengimplementasikan algoritma *levenshtein distance* pada pencarian, ketik kata yang ingin di cari seperti kata "pa" maka semua yang ada kata "pa" nya semua akan di munculkan/di tampilkan.

Gambar 7. Halaman pencarian data buku

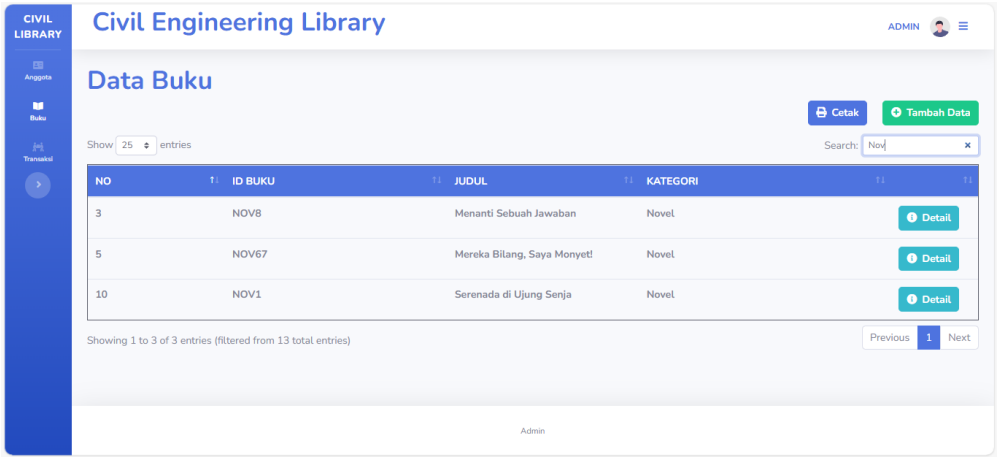
Berikut ini pencarian kata "mic" maka kata yang di maksudkan akan di tampilan seperti di bawah ini. Karena kata yang di maksudkan yaitu judul buku, "*Microsoft* EXCEL 2013".




Gambar 8. Halaman pencarian judul buku

Adapun pada menu pencarian di bawah ini, jika ingin mencari kategori tertentu pun bisa. Seperti kata "Nov" karena yang di maksudkan adalah "Novel" maka, kategori tersebut akan di tampilkan.

DOI: https://doi.org/10.54082/jupin.1662
p-ISSN: 2808-148X
e-ISSN: 2808-1366




Gambar 9. Halaman pencarian kategori buku

### 3.3. Perbandingan dengan Literatur

Hasil ini sejalan dengan penelitian Akbar & Ramdhania (2024) dan Bagaskoro & Mufti (2024), yang melaporkan peningkatan akurasi pencarian teks setelah penerapan Levenshtein Distance. Namun, metode ini belum seoptimal cosine similarity atau TF-IDF untuk pencarian berbasis konteks, sehingga dapat menjadi arah pengembangan selanjutnya. Selain itu, hasil pengujian ini menunjukkan potensi penerapan algoritma Levenshtein Distance tidak hanya terbatas pada pencarian buku di perpustakaan, tetapi juga dapat digunakan dalam sistem pencarian pada platform e-learning, basis data penelitian, dan aplikasi komersial seperti e-commerce. Kelebihan algoritma ini adalah kemampuannya mengakomodasi kesalahan pengetikan yang umum dilakukan oleh pengguna, sehingga mengurangi frustasi pengguna akibat hasil pencarian yang kosong atau tidak relevan. Namun, perlu dicatat bahwa efektivitas algoritma ini sangat dipengaruhi oleh penentuan ambang batas (threshold) yang tepat. Ambang batas yang terlalu rendah dapat menyebabkan hasil pencarian yang terlalu sedikit, sedangkan ambang batas yang terlalu tinggi berpotensi menghasilkan terlalu banyak hasil yang kurang relevan. Dengan demikian, penyesuaian parameter ini menjadi aspek penting dalam pengembangan sistem di masa depan. Dari perspektif pengguna, integrasi antarmuka pencarian yang intuitif dengan algoritma ini dapat memberikan pengalaman yang lebih baik, mempercepat proses pencarian, dan meningkatkan kepuasan secara keseluruhan.

## 4. KESIMPULAN

Penelitian ini menunjukkan bahwa penerapan algoritma Levenshtein Distance pada sistem informasi perpustakaan mampu meningkatkan akurasi pencarian buku meskipun terjadi kesalahan pengetikan. Sistem berhasil meningkatkan precision dan recall, serta mendapatkan umpan balik positif dari pengguna. Potensi pengembangan ke depan meliputi integrasi metode machine learning untuk adaptasi ambang batas kesalahan, serta kombinasi dengan metode pencarian lain seperti TF-IDF untuk meningkatkan relevansi hasil berbasis konteks. Lebih jauh lagi, implementasi algoritma ini di lingkungan perpustakaan dapat menjadi model bagi pengembangan sistem serupa di berbagai institusi pendidikan dan organisasi. Dengan menggabungkan algoritma Levenshtein Distance dan metode pencarian berbasis konteks seperti TF-IDF atau cosine similarity, diharapkan sistem pencarian dapat memahami maksud pengguna secara lebih mendalam. Pengembangan di masa depan juga dapat memanfaatkan pembelajaran mesin (machine learning) untuk menyesuaikan parameter pencarian secara dinamis berdasarkan perilaku pengguna. Selain itu, studi lanjutan dapat mengeksplorasi integrasi algoritma ini dengan sistem rekomendasi buku untuk memberikan saran bacaan yang relevan meskipun pengguna tidak mengetahui judul pasti yang dicari. Penerapan ini tidak hanya bermanfaat untuk meningkatkan efisiensi pencarian, tetapi juga untuk memperluas akses informasi dan mendukung pembelajaran.

DOI: https://doi.org/10.54082/jupin.1662
p-ISSN: 2808-148X
e-ISSN: 2808-1366

## DAFTAR PUSTAKA

Aditya, R., Pranatawijaya, V. H., & Putra, P. B. A. A. (2021). Rancang Bangun Aplikasi Monitoring Kegiatan Menggunakan Metode Prototype. *Journal of Information Technology and Computer Science*, *1*(1), 47–57.

Aidin, B. S., Loda, D. Y. L., Hadi, M., & Maskat, S. (2021). *Invasi Media Massa*. Media Nusa Creative (MNC Publishing).

Akbar, I. F., & Ramdhania, K. F. (2024). Algoritma Levenshtein Distance sebagai Solusi Efisiensi Pencarian pada Training Registration System. *Journal of Students 'Research in Computer Science*, *5*(2), 147–154.

Bagaskoro, M. A., & Mufti, M. (2024). PENERAPAN ALGORITMA LEVENSHTEIN DISTANCE, N-GRAM DAN SYNONYM REPLACEMENT PADA SISTEM PENILAIAN ESAI OTOMATIS DI UNIVERSITAS BUDI LUHUR. In *Prosiding Seminar Nasional Mahasiswa Fakultas Teknologi Informasi (SENAFTI)* (Vol. 3, pp. 451–460).

Fauzi, A. A., Kom, S., Kom, M., Budi Harto, S. E., Mm, P. I. A., Mulyanto, M. E., … Kom, S. (2023). *Pemanfaatan Teknologi Informasi di Berbagai Sektor Pada Masa Society 5.0*. PT. Sonpedia Publishing Indonesia.

Gani, A. G., & Effendi, M. R. (2022). Perancangan Sistem Informasi Absen Siswa pada SMA Islamic School Berbasis Web. *JSI (Jurnal Sistem Informasi) Universitas Suryadarma*, *9*(2), 49–60.

Indriyono, B. V. (2020). Kombinasi damerau levenshtein dan jaro-winkler distance untuk koreksi kata bahasa inggris. *Jurnal Teknik Informatika Dan Sistem Informasi*, *6*(2).

Kuswanto, W. (2020). Implementasi Algoritma Levenshtein Distance Dengan Restful Web Service Pada Kata Bahasa Indonesia Ke Bahasa Jawa Berbasis Web. *JUSTINDO (Jurnal Sistem Dan Teknologi Informasi Indonesia)*, *5*(2), 78–85.

Manullang, A. H., Aritonang, M., & Purba, M. J. (2021). Sistem Informasi Bimbingan Belajar Number One Medan Berbasis Web. *TAMIKA: Jurnal Tugas Akhir Manajemen Informatika & Komputerisasi Akuntansi*, *1*(1), 44–49.

Mualo, A., & Basri, H. (2024). Rancang Bangun Sistem Absensi Terintegrasi Mahasiswa Jurusan Manajemen Informatika Politeknik Negeri Fakfak. *Jurnal Informasi, Sains Dan Teknologi*, *7*(1), 211–225.

Putra, A. A., Tuffahati, D. A., Fauzi, H. S., Febriliyan, I. H., & Saifudin, A. (2023). Analisis dan Perancangan Sistem Informasi Perpustakaan SMK Pesat Bogor. *JRIIN: Jurnal Riset Informatika Dan Inovasi*, *1*(2), 341–346.

Rahmanto, Y., Alita, D., Putra, A. D., Permata, P., & Suaidah, S. (2022). Penerapan Sistem Informasi Perpustakaan Berbasis Web Pada Smk Nurul Huda Pringsewu. *Journal of Social Sciences and Technology for Community Service (JSSTCS)*, *3*(2), 151–159.

Santoso, P., Yuliawati, P., Shalahuddin, R., & Zaeni, I. A. E. (2020). Penghapusan kolom dan baris pertama pada matriks distance untuk optimasi spell checker damerau-levenshtein distance. *Sains, Apl. Komputasi Dan Teknol. Inf*, *2*(2), 57–63.

**Jurnal Penelitian Inovatif (JUPIN)**
Vol. 5, No. 3, Agustus 2025, Hal. 2621-2632
https://jurnal-id.com/index.php/jupin
DOI: https://doi.org/10.54082/jupin.1662
p-ISSN: 2808-148X
e-ISSN: 2808-1366

# Halaman Ini Dikosongkan