
**SISTEM OTOMATISASI *FORMATTING* DOKUMEN SKRIPSI MENGGUNAKAN
METODE *RULE-BASED PROCESSING* SESUAI PEDOMAN PENULISAN
PROGRAM STUDI INFORMATIKA
UNIVERSITAS MUHAMMADIYAH MAKASSAR**

Muhammad Farid Nurzikri¹, Fahrim Irfhamna Rachman², Lukman³

¹²³Program Studi Informatika, Fakultas Teknik,
Universitas Muhammadiyah Makassar

¹105841111020@student.unismuh.ac.id, ²fachrim141020@unismuh.ac.id,

³lukman@unismuh.ac.id

ABSTRACT

The process of compiling a thesis frequently encounters technical formatting problems, including incorrect margin settings, inappropriate font types and sizes, inconsistent line spacing, improper chapter numbering, and non-compliant citation formats. These recurring issues consume time and increase the revision burden for both students and supervisors. This study develops DocuForge, a web-based thesis formatting automation system using the Rule-Based Processing method for the Informatics Study Program at Universitas Muhammadiyah Makassar. The system architecture adopts a Router–Service–Rule pattern that integrates document parsing via python-docx, IF–THEN rule validation, automatic formatting correction, and PDF report generation using fpdf2. The rules implemented cover eight formatting aspects: margin, font, line spacing, paragraph alignment, chapter heading structure, subchapter headings, cover page layout, and paragraph indentation. Testing was conducted using black box testing on a thesis document containing various formatting violations. Results show that the system successfully detected all explicit formatting violations and automatically corrected 1,126 non-compliant text segments, 65 paragraph spacing errors, 59 misaligned paragraphs, 12 cover elements, 5 chapter headings, and 28 subchapters. The Rule-Based Processing approach proved highly effective for deterministic and numerical formatting attributes. The system significantly reduces the manual revision burden, enabling students to focus more on research substance rather than technical formatting issues.

Keywords: thesis formatting automation, rule-based processing, document validation, DocuForge, python-docx

ABSTRAK

Proses penyusunan skripsi seringkali menemui kendala teknis terkait formatting dokumen, seperti kesalahan pengaturan margin, jenis dan ukuran font, spasi antarbaris, penomoran bab, serta format sitasi. Permasalahan yang berulang ini menyita waktu dan meningkatkan beban revisi bagi mahasiswa maupun dosen pembimbing. Penelitian ini mengembangkan DocuForge, sebuah system otomatisasi formatting skripsi berbasis web menggunakan metode Rule-Based Processing untuk Program Studi Informatika Universitas Muhammadiyah Makassar.

Arsitektur sistem mengadopsi pola Router–Service–Rule yang mengintegrasikan parsing dokumen melalui python-docx, validasi aturan IF–THEN, koreksi formatting otomatis, serta pembuatan laporan PDF menggunakan fpdf2. Aturan yang diimplementasikan mencakup delapan aspek formatting: margin, font, spasi baris, alignment paragraf, struktur heading BAB, heading subbab, tata letak halaman cover, dan indentasi paragraf. Pengujian dilakukan menggunakan pendekatan black box testing terhadap dokumen skripsi yang mengandung berbagai pelanggaran formatting. Hasil menunjukkan sistem berhasil mendeteksi seluruh pelanggaran formatting yang bersifat eksplisit dan secara otomatis memperbaiki 1.126 segmen teks tidak sesuai, 65 kesalahan spasi paragraf, 59 paragraf dengan alignment keliru, 12 elemen halaman cover, 5 judul BAB, dan 28 subbab. Pendekatan Rule-Based Processing terbukti sangat efektif untuk atribut formatting yang bersifat deterministik dan numerik. Sistem ini secara signifikan mengurangi beban revisi manual, sehingga mahasiswa dapat lebih fokus pada substansi penelitian.

Kata Kunci: otomatisasi formatting skripsi, rule-based processing, validasi dokumen, DocuForge, python-docx.

A. Pendahuluan

Skripsi merupakan karya ilmiah wajib yang harus disusun oleh mahasiswa tingkat akhir sebagai salah satu syarat memperoleh gelar Sarjana Strata Satu (S1). Penyusunan skripsi tidak hanya bertujuan untuk mengukur penguasaan mahasiswa terhadap bidang keilmuannya, tetapi juga melatih kemampuan berpikir ilmiah, analisis masalah, serta penyajian hasil penelitian secara sistematis sesuai kaidah akademik yang berlaku (Suwita, 2020).

Dalam praktiknya, banyak mahasiswa mengalami kendala pada aspek teknis penulisan, terutama dalam menyesuaikan format dokumen skripsi dengan pedoman yang berlaku. Permasalahan yang sering muncul meliputi kesalahan pengaturan margin, jenis dan ukuran font, spasi antarbaris, sistem penomoran bab dan subbab, hingga format sitasi dan daftar pustaka. Skripsi yang tidak sesuai pedoman format umumnya dikembalikan untuk direvisi, sehingga menyebabkan penundaan penyelesaian studi (Suwita, 2020). Castrena, Cholissodin, dan Prakoso (2021) menegaskan bahwa pemeriksaan manual dokumen memiliki keterbatasan dari sisi efisiensi dan konsistensi, terutama ketika dilakukan dalam skala besar.

Selain itu, penyusunan sitasi dan daftar pustaka secara manual membutuhkan ketelitian tinggi. Saputra, Wiarsih, dan Yuniyanti (2025) menyatakan bahwa ketidaksesuaian format sitasi dapat mengurangi kredibilitas karya ilmiah. Permasalahan ini menegaskan perlunya sistem yang mampu membantu

standarisasi format dokumen secara otomatis. Metode Rule-Based Processing dipandang tepat untuk diterapkan karena bekerja berdasarkan aturan-aturan yang jelas dan terstruktur dalam bentuk if-then rules. Aesyi dan Cahyo (2023) menyatakan bahwa metode rule-based merupakan pendekatan sederhana namun efektif untuk mendeteksi ketidaksesuaian dan melakukan penyesuaian data berdasarkan aturan yang telah ditetapkan.

Berdasarkan permasalahan tersebut, penelitian ini berfokus pada pengembangan Sistem Otomatisasi Formatting Dokumen Skripsi Menggunakan Metode Rule-Based Processing yang diberi nama DocuForge. Sistem ini diharapkan mampu meminimalkan kesalahan formatting, meningkatkan efisiensi penyusunan skripsi, serta memungkinkan mahasiswa untuk lebih fokus pada substansi penelitian. Tujuan penelitian ini adalah: (1) merancang sistem otomatisasi formatting dokumen skripsi sesuai pedoman Program Studi Informatika Universitas Muhammadiyah Makassar; (2) menerapkan metode Rule-Based Processing untuk memvalidasi dan menyesuaikan aturan-aturan penulisan skripsi secara otomatis.

B. Metode Penelitian

1. Alat dan Bahan

Penelitian ini dilaksanakan di Laboratorium Teknik Informatika Universitas Muhammadiyah Makassar pada bulan Oktober hingga Desember 2025. Perangkat yang digunakan meliputi laptop Lenovo IdeaPad Slim 3 15AMN8, bahasa pemrograman Python, editor Visual Studio Code, berbagai library Python (python-docx, FastAPI, Uvicorn, fpdf2), serta basis data MySQL. Data uji yang digunakan adalah dokumen skripsi mahasiswa Program Studi Informatika UNISMUH Makassar berformat DOCX.

2. Perancangan Sistem

Sistem DocuForge dikembangkan menggunakan arsitektur berbasis web dengan pemisahan komponen backend dan frontend. Backend dibangun menggunakan framework FastAPI yang dijalankan melalui ASGI server Uvicorn, sedangkan antarmuka pengguna dikembangkan menggunakan HTML, CSS, dan JavaScript. Struktur implementasi mengadopsi pola Router-Service-Rule sebagai adaptasi dari Model View Controller (MVC) guna mendukung pemrosesan dokumen berbasis aturan.

Alur kerja sistem dimulai ketika mahasiswa mengunggah file skripsi berformat DOCX. Sistem kemudian melakukan parsing dokumen menggunakan pustaka python-docx untuk mengekstraksi atribut formatting seperti margin, font, spasi, alignment, dan struktur heading. Hasil ekstraksi selanjutnya divalidasi menggunakan mekanisme rule-based processing. Jika ditemukan ketidaksesuaian, sistem secara otomatis menerapkan proses auto-formatting. Keluaran akhir berupa dokumen hasil perbaikan dan laporan validasi dalam format PDF yang dapat diunduh pengguna.

3. Pembentukan Rule Base

Pembentukan rule base dilakukan dengan menurunkan aturan-aturan dari Pedoman Penulisan Skripsi Program Studi Informatika Universitas Muhammadiyah Makassar. Aturan-aturan tersebut diformalkan ke dalam bentuk if-then rules agar dapat dieksekusi dalam sistem Python. Aspek formatting yang dijadikan dasar pembentukan rule base meliputi: (1) ukuran kertas A4 dengan margin atas dan kiri 4 cm, serta margin bawah dan kanan 3 cm; (2) jenis huruf Times New Roman ukuran 12 pt untuk teks utama; (3) spasi antarbaris 1,5; (4) paragraf baru menjorok 1,27 cm dari margin kiri dengan teks rata kiri-kanan (justify); (5) penomoran halaman awal menggunakan angka Romawi kecil di tengah bawah, dan Bab I ke atas menggunakan angka Arab di kanan atas; (6) judul BAB menggunakan huruf kapital, rata tengah, dan tebal; serta (7) daftar pustaka menggunakan APA Style.

4. Teknik Pengujian

Pengujian sistem menggunakan pendekatan black box testing, yaitu menguji fungsi sistem berdasarkan masukan dan keluaran yang dihasilkan tanpa melihat kode internal. Dokumen skripsi mahasiswa dijadikan masukan untuk diuji oleh sistem. Hasil validasi sistem kemudian dibandingkan dengan pemeriksaan manual untuk memastikan keakuratan. Pengujian dilakukan secara bertahap mulai dari pengujian unit (fungsi pemeriksa margin, font, spasi), integrasi (keseluruhan proses pemeriksaan), hingga pengujian dokumen skripsi lengkap.

C. Hasil Penelitian dan Pembahasan

1. Implementasi Sistem

Sistem DocuForge berhasil diimplementasikan sebagai aplikasi web yang mengintegrasikan tiga komponen utama. Komponen Router bertugas menangani

request pengguna melalui endpoint REST API serta mengatur komunikasi antara frontend dan backend. Komponen Service mengelola logika bisnis, termasuk orkestrasi proses parsing, validasi, dan auto-formatting dokumen. Komponen Rule mengimplementasikan aturan IF–THEN berdasarkan pedoman penulisan skripsi untuk mengevaluasi dan menentukan tindakan korektif terhadap dokumen. Tabel 1 menyajikan komponen utama sistem beserta teknologi dan fungsinya masing-masing.

Tabel 1 Komponen Utama Sistem DocuForge

No	Komponen	Teknologi	Fungsi
1	Backend API	FastAPI + Uvicorn	Menyediakan endpoint REST API untuk validasi dan formatting
2	Document Parser	python-docx	Membaca dan mengekstraksi atribut dokumen DOCX
3	Rule Validator	Python (Rule-Based)	Memvalidasi dokumen berdasarkan aturan IF- THEN
4	Document Formatter	python-docx	Mengubah format dokumen secara otomatis sesuai aturan
5	Report Generator	fpdf2	Menghasilkan laporan PDF hasil validasi dan formatting
6	Frontend	HTML/CSS/JS	Antarmuka pengguna berbasis web

2. Aturan Formatting yang Diimplementasikan

Metode rule-based processing diimplementasikan melalui seperangkat aturan IF–THEN yang disusun berdasarkan Pedoman Penulisan Skripsi. Setiap aturan merepresentasikan satu aspek formatting yang harus dipenuhi dokumen. Untuk meningkatkan akurasi validasi dan mengurangi kemungkinan false positive, sistem menerapkan nilai toleransi pada beberapa parameter: toleransi margin $\pm 0,2$ cm, toleransi ukuran font $\pm 0,5$ pt, dan toleransi spasi $\pm 0,1$. Tabel 2 menyajikan seluruh aturan formatting yang diimplementasikan.

Tabel 2 Aturan Formatting

No	Aspek	Aturan (Pedoman)	Implementasi IF-THEN
1	Margin	Atas & Kiri: 4cm, Bawah & Kanan: 3cm	IF margin $\neq$ expected (± 0.2 cm) THEN ERROR
2	Font	Times New Roman 12pt, warna hitam	IF font $\neq$ "Times New Roman" OR size $\neq$ 12pt THEN ERROR
3	Spasi	1.5 (line spacing)	IF line spacing $\neq$ 1.5 (± 0.1) THEN ERROR
4	Alignment	Rata kiri-kanan (justify)	IF alignment $\neq$ "justify" THEN WARNING
5	Heading BAB	HURUF KAPITAL, bold, rata tengah	IF NOT (uppercase AND bold AND center) THEN ERROR
6	Subbab	Kapital di awal kata, bold, rata kiri	IF NOT bold THEN WARNING
7	Cover	Center, spasi 1, tanpa indent, bold	IF NOT (center AND spacing=1.0) THEN WARNING
8	Indentasi	Paragraf indent 1.27cm	IF first line indent $\neq$ 1.27cm THEN WARNING

3. Pengujian Sistem

Pengujian difokuskan pada dua aspek utama: kemampuan sistem dalam melakukan validasi awal (error detection capability) dan kemampuan melakukan perbaikan otomatis (auto-formatting effectiveness). Dokumen uji yang digunakan adalah "FARIDD SAMPLE.docx" yang dipilih karena mengandung berbagai variasi kesalahan formatting yang umum ditemukan dalam penyusunan skripsi, mulai dari kesalahan teknis seperti jenis font yang tidak sesuai, spasi tidak konsisten, dan alignment yang masih rata kiri, hingga kesalahan struktural seperti format heading BAB yang tidak sesuai standar.

a. Kondisi Sebelum Formatting Pada tahap validasi awal, sistem membandingkan atribut formatting aktual dokumen terhadap aturan IF-THEN yang telah didefinisikan. Hasil validasi menunjukkan sejumlah error dan warning yang signifikan. Sistem mendeteksi penggunaan jenis font Arial pada sebagian besar isi dokumen (padahal pedoman mensyaratkan Times New Roman 12 pt), dengan 1.126 bagian teks teridentifikasi tidak sesuai standar font. Selain itu, spasi

antarbaris terdeteksi bernilai 0,1 (single spacing tidak standar), sebagian besar paragraf menggunakan perataan kiri, judul BAB tidak dicetak tebal dan tidak rata tengah, serta seluruh elemen cover menggunakan alignment left dan spasi yang tidak sesuai ketentuan. Kondisi ini memperlihatkan inkonsistensi formatting yang bersifat sistematis pada hampir seluruh bagian dokumen, menegaskan bahwa proses formatting manual sangat rentan terhadap inkonsistensi.

b. Perubahan Formatting oleh Sistem

Setelah tahap validasi selesai, sistem menjalankan mekanisme auto-formatting secara otomatis dan sistematis terhadap setiap elemen yang teridentifikasi tidak sesuai. Perubahan yang berhasil dilakukan sistem disajikan pada Tabel 3.

Tabel 3 Perubahan Formatting yang Dilakukan Sistem

No	Tipe Perubahan	Jumlah	Keterangan
1	Font	1.126 segmen	Diubah ke Times New Roman
2	Spasi Paragraf	65 paragraf	Diatur ke spasi 1,5
3	Alignment	59 paragraf	Diubah ke rata kiri-kanan (justify)
4	Cover Page	12 elemen	Center, bold, spasi 1,0
5	Heading BAB	5 judul	sesuai pedoman Uppercase, bold, center, page break
6	Subbab	28 subbab	Format bertingkat
7	Indentasi	17 paragraf	sesuai pedoman Disesuaikan setelah subbab
8	Margin	Seluruh halaman	Atas & kiri 4cm, bawah & kanan 3cm

Seluruh perubahan dilakukan berdasarkan parameter deterministik yang telah ditentukan, termasuk toleransi margin dan ukuran font untuk menghindari false positive akibat perbedaan presisi unit. Proses ini berlangsung dalam waktu yang relatif singkat dibandingkan dengan proses manual yang biasanya memerlukan pemeriksaan satu per satu pada setiap elemen dokumen.

c. Hasil Validasi Setelah Formatting

Setelah auto-formatting selesai, sistem kembali melakukan proses validasi untuk mengukur tingkat efektivitas perbaikan. Hasil menunjukkan bahwa mayoritas

error kritis berhasil diperbaiki. Judul BAB telah diformat sesuai pedoman dengan huruf kapital penuh, dicetak tebal, dan rata tengah. Paragraf isi telah menggunakan font Times New Roman ukuran 12 pt serta alignment rata kiri-kanan. Seluruh elemen teks pada halaman cover telah diformat menjadi rata tengah, dicetak tebal, serta menggunakan spasi tunggal sesuai pedoman. Sisa warning yang masih muncul sebagian besar berkaitan dengan elemen yang memerlukan interpretasi kontekstual lebih lanjut, seperti variasi pola heading yang kompleks.

4. Analisis Hasil Pengujian

Berdasarkan hasil pengujian, sistem menunjukkan tingkat efektivitas yang tinggi dalam mendeteksi dan memperbaiki ketidaksesuaian formatting dokumen skripsi. Pada tahap validasi awal, sistem mampu mengidentifikasi seluruh pelanggaran formatting yang bersifat eksplisit dan terdefinisi dalam aturan, mencakup kesalahan pada jenis dan ukuran font, spasi antarbaris, alignment paragraf, struktur heading BAB, format subbab, serta tata letak halaman cover. Kemampuan deteksi ini menunjukkan bahwa mekanisme rule-based processing bekerja secara komprehensif dan konsisten dalam mengevaluasi setiap elemen dokumen berdasarkan parameter yang telah ditentukan.

Perbaikan paling signifikan terjadi pada aspek deterministik seperti font, margin, spasi, dan alignment, yang secara langsung dapat diukur dan dibandingkan dengan standar pedoman. Hal ini menunjukkan bahwa pendekatan berbasis aturan IF-THEN sangat efektif dalam menangani atribut formatting yang memiliki nilai absolut dan tidak memerlukan interpretasi kontekstual. Selama aturan dapat direpresentasikan dalam bentuk kondisi logis yang jelas, sistem mampu mengeksekusi tindakan korektif secara akurat dan konsisten.

Dari perspektif efisiensi, sistem secara signifikan mengurangi beban manual dalam proses revisi teknis. Jika formatting dilakukan secara manual, pengguna harus memeriksa dan menyesuaikan setiap paragraf, heading, dan elemen dokumen satu per satu, yang berpotensi memakan waktu lama dan rentan terhadap inkonsistensi. Dengan sistem otomatisasi, sebagian besar koreksi dapat dilakukan dalam satu kali eksekusi. Hal ini sejalan dengan temuan Castrena et al. (2021) yang menyatakan bahwa pemeriksaan manual memiliki keterbatasan dari sisi efisiensi dan konsistensi, terutama dalam skala besar.

Namun demikian, hasil pengujian juga memperlihatkan keterbatasan pendekatan rule-based dalam menangani aspek yang memerlukan interpretasi kontekstual, seperti struktur halaman cover dan hierarki heading bertingkat yang kompleks. Deteksi berbasis pola tidak selalu mampu memahami makna semantik teks, sehingga beberapa warning masih tersisa setelah auto-formatting. Selain itu, sistem belum sepenuhnya menangani validasi elemen non-teks seperti tabel dan gambar. Untuk aspek kontekstual, diperlukan pengembangan lebih lanjut melalui pendekatan hybrid yang mengintegrasikan teknik pembelajaran mesin atau pemrosesan bahasa alami.

Arsitektur berbasis web serta konfigurasi multi-program studi memberikan fleksibilitas tambahan dalam penerapan sistem. Aturan formatting dapat dikonfigurasi ulang sesuai pedoman yang berbeda tanpa mengubah struktur utama sistem, sehingga berpotensi diterapkan pada konteks akademik yang lebih luas. Hal ini menunjukkan bahwa sistem yang dikembangkan tidak hanya menjawab kebutuhan Program Studi Informatika UNISMUH Makassar, tetapi juga dapat diadaptasi untuk institusi lain.

D. Kesimpulan

Penelitian ini berhasil mengembangkan sistem DocuForge, yaitu sistem otomatisasi formatting dokumen skripsi berbasis web menggunakan metode Rule-Based Processing sesuai Pedoman Penulisan Skripsi Program Studi Informatika Universitas Muhammadiyah Makassar. Sistem mengadopsi arsitektur Router–Service–Rule yang mengintegrasikan parsing dokumen, validasi aturan IF–THEN, auto-formatting otomatis, dan pembuatan laporan PDF. Hasil pengujian menggunakan black box testing menunjukkan bahwa sistem berhasil mendeteksi seluruh pelanggaran formatting yang bersifat eksplisit dan secara otomatis memperbaiki 1.126 segmen teks tidak sesuai, 65 kesalahan spasi, 59 paragraf dengan alignment keliru, 12 elemen halaman cover, 5 judul BAB, dan 28 subbab. Pendekatan Rule-Based Processing terbukti sangat efektif untuk atribut formatting yang bersifat deterministik dan numerik, dengan tingkat keberhasilan mendekati sempurna. Sistem ini secara signifikan mengurangi beban revisi teknis, sehingga mahasiswa dapat lebih fokus pada substansi penelitian. Untuk pengembangan lebih lanjut, sistem disarankan untuk meningkatkan mekanisme deteksi cover berbasis

struktur halaman, mengintegrasikan pendekatan hybrid dengan machine learning untuk akurasi deteksi heading dan struktur kontekstual, menambahkan modul validasi tabel dan gambar, mengembangkan fitur batch processing untuk dokumen majemuk, serta memperluas pengujian ke berbagai program studi guna menguji reliabilitas dan generalisasi sistem.

DAFTAR PUSTAKA

- Aesyi, U. S., & Cahyo, P. W. (2023). Cuscoma: Platform peningkatan penjualan produk berdasarkan analisis komentar pelanggan di marketplace Shopee menggunakan metode rule-based. *Jurnal Sains dan Informatika*, 9(1), 1–8. <https://doi.org/10.34128/jsi.v9i1.539>
- Barus, G. (2022). Mengulik tiga faktor pendukung percepatan penulisan skripsi mahasiswa. *Scholaria: Jurnal Pendidikan Dan Kebudayaan*, 12(2), 96–108. <https://doi.org/10.24246/j.js.2022.v12.i2.p96-108>
- Castrena, A. W., Cholissodin, I., & Prakoso, B. S. (2021). Pengembangan perangkat lunak untuk otomatisasi pengecekan format penulisan dokumen jurnal skripsi menggunakan fitur tag pada XML. *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, 5(4), 1596–1604.
- Diono, D., Simanjuntak, G. L. M., Toar, H., Gozali, M. S., & Jefiza, A. (2022). Sistem semi otomasi pada proses tinning pin lampu di PT. Excelitas Technologies Batam. *Jurnal Integrasi*, 14(1), 52–60.
- Ghozali, M. I., Riadi, A. A., Putra, D. A., & Sugiharto, W. H. (2024). Pengembangan sistem sortir otomatis untuk jeruk citrus: Integrasi teknologi sensor dan algoritma rule-based. *Jurnal Teknologi dan Sistem*, 4, 241–248.
- Makmun, A. H., Devilishanti, T., Yuliana, D., & Hasni, J. (2023). Pelatihan formatting karya ilmiah kepada mahasiswa Universiti Sains Islam Malaysia. *Indonesian Journal Of Community Service*, 3(1), 24–29.
- Napitupulu, R., Mauludin, R., Gianadevi, M., & Rumambi, H. (2023). Aplikasi konversi dokumen gambar menjadi teks menggunakan model Waterfall. *Jurnal Informatika dan Rekayasa Perangkat Lunak*, 5(2), 78–86.
- Prasetyo, D., Dima, A., Gelong, M., Hipir, M., & Adoe, V. (2023). Otomatisasi sistem pengarsipan surat pada Program Studi Ilmu Lingkungan Pascasarjana

-
- Universitas Nusa Cendana berbasis web. Jurnal Pengabdian Kepada Masyarakat, 3(1), 23–30.
- Sabila, F. H., & Rakawitan, F. (2023). Persiapan peralatan dan dokumen sebelum melaksanakan aktivitas bongkar curah kering. Student Scientific Creativity Journal, 1(5), 509–519. <https://doi.org/10.55606/sscj-amik.v1i5.4393>
- Santoso, H. (2020). Ekstraksi informasi berbasis rule dalam pengolahan dokumen digital. Jurnal Ilmu Komputer dan Informatika, 6(1), 45–53.
- Saputra, W. J., Wiarsih, S., & Yuniyanti, B. (2025). Pemanfaatan merger web sebagai standar baku penulisan sitasi dan daftar pustaka. J-Symbol: Jurnal Magister Pendidikan Bahasa dan Sastra Indonesia, 13(1), 604–612.
- Soyusiawaty, D., & Putra, F. G. (2023). Pengembangan chatbot untuk layanan pimpinan daerah Muhammadiyah Kota Yogyakarta menggunakan metode rule-based. Kesatria: Jurnal Penerapan Sistem Informasi (Komputer dan Manajemen), 4(2), 354–363.
- Suwita, F. S. (2020). Pengembangan sistem informasi tugas akhir dan skripsi (SIMITA) di Universitas Komputer Indonesia. Jurnal Teknologi dan Informasi, 10(1), 71–82.
- Syazali, M., & Erfan, M. (2022). Keterampilan menulis mahasiswa PGSD melalui implementasi template makalah sebagai pedoman penyusunan. Research and Development Journal Of Education, 8(1), 35–45.
- Zaiha, F. H. (2025). Identifikasi faktor penghambat mahasiswa tingkat akhir dalam menyelesaikan tugas akhir di Program Studi Pendidikan Jasmani Universitas Muhammadiyah Kotabumi. Griya Cendikia, 10(1), 447–456.