

Transformer-Based Digital Learning Assistant with Model Context Protocol (MCP) Integration for Vocational High School Students

Fikri Ramadhani¹ , Geovanne Farel^{1*} 

¹ Universitas Negeri Padang, Padang, Indonesia

*Corresponding Author: geovannefarell@ft.unp.ac.id

Abstract - The implementation of the Transformer architecture in digital learning assistants still faces a major challenge: artificial intelligence models have not been integrated with local school data sources, resulting in interactions that tend to be generic and non-contextual. This study aims to design and develop a Transformer-based digital learning assistant integrated with the Model Context Protocol (MCP) for the Informatics subject at the Vocational High School (SMK) level. The MCP serves as a bridge enabling the chatbot to securely access local databases—including learning modules, assignments, and student information—and produce personalized responses. The system was developed following the Waterfall method, utilizing the Laravel framework, MySQL database, and the OpenAI API for the language model. System evaluation was conducted through two approaches: (1) structural testing using the white-box testing method and cyclomatic complexity metrics on 49 functional forms covering admin, teacher, and student actors; and (2) quality validation by three expert practitioners using a practicality assessment instrument. The white-box testing results showed a 100% success rate—all execution paths in every form were successfully traversed without any logic errors. Meanwhile, expert validation categorized the application as “Very Practical,” indicating that it is suitable for use as a learning medium. The integration of MCP successfully realized a digital assistant capable of leveraging local data to provide relevant and adaptive interactions, thereby addressing the need for personalized learning at the vocational high school level.

Keywords - Digital Assistant, Transformer Architecture, Artificial Intelligence, Adaptive Learning, Learning Management System.

I. PENDAHULUAN

Perkembangan teknologi kecerdasan buatan (Artificial Intelligence/AI), khususnya pada bidang pemrosesan bahasa alami (Natural Language Processing/NLP), telah membawa perubahan besar dalam berbagai sektor, termasuk pendidikan [1],[2]. Salah satu terobosan paling signifikan adalah arsitektur Transformer yang diperkenalkan oleh Vaswani et al. (2017) melalui mekanisme *self-attention* [3]. Arsitektur ini menjadi fondasi model bahasa besar (Large Language Models/LLM) seperti GPT, BERT, dan T5, yang mampu memahami konteks, menghasilkan respons koheren, serta menjalankan berbagai tugas bahasa secara otomatis [4], [5].

DOI: <https://doi.org/10.24036/voteteknika.v14i2.135559>

Received : 2025-08-14
Revised : 2026-05-11
Accepted : 2026-05-12
Published : 2026-06-26



For all articles published in VOTETEKNIKA
<https://ejournal.unp.ac.id/index.php/voteknika>, © copyright is retained by the authors. This is an open-access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.

Keunggulan Transformer dibandingkan pendekatan sebelumnya seperti Recurrent Neural Network (RNN) dan Convolutional Neural Network (CNN) terletak pada kemampuannya memproses seluruh urutan input secara paralel, sehingga lebih efisien dalam pelatihan dan lebih akurat dalam memahami konteks jangka panjang [6]. Hal ini membuatnya sangat relevan untuk pengembangan sistem interaktif seperti asisten digital, yang membutuhkan pemahaman mendalam terhadap interaksi pengguna [7].

Meskipun LLM memiliki kapabilitas tinggi, keterbatasan tetap muncul ketika model tidak memiliki akses ke data terkini atau informasi spesifik yang tersimpan dalam sistem lokal, seperti database siswa, modul pembelajaran, atau riwayat aktivitas belajar [8]. Akibatnya, respons yang dihasilkan sering kali bersifat umum dan kurang relevan dengan konteks pengguna. Untuk mengatasi hal tersebut, diperlukan mekanisme integrasi antara LLM dan sumber data eksternal secara terstruktur.

Model Context Protocol (MCP) hadir sebagai solusi untuk integrasi ini. Dikembangkan oleh Anthropic, MCP memungkinkan model bahasa berinteraksi langsung dengan layanan eksternal seperti basis data, API, atau alat bantu digital [9]. Dengan MCP, model dapat membentuk kueri secara otomatis, mengambil data dari sumber lokal seperti MySQL, dan menghasilkan respons yang kontekstual, akurat, serta personal. Integrasi ini memungkinkan asisten digital tidak

hanya mengandalkan pengetahuan umum hasil pelatihan, tetapi juga memberikan jawaban yang sesuai dengan datanya di sistem, seperti status tugas, progres belajar, dan profil siswa [10].

Dalam konteks pendidikan di Indonesia, kebutuhan akan asisten belajar yang personal dan adaptif semakin tinggi, terlebih di sekolah menengah kejuruan (SMK) yang menuntut keterampilan teknis dan manajemen belajar mandiri [11]. Namun, penelitian dan implementasi asisten digital berbasis LLM di lingkungan SMK masih sangat terbatas. Studi-studi sebelumnya cenderung berfokus pada chatbot generik untuk tanya jawab materi tanpa memanfaatkan data spesifik siswa sebagai dasar personalisasi. Lebih jauh, pemanfaatan MCP untuk mengintegrasikan LLM dengan basis data sekolah guna menciptakan asisten pembelajaran yang adaptif dan berbasis data riil siswa SMK khususnya pada mata pelajaran Informatika belum menjadi perhatian utama. Kesenjangan inilah yang mendasari dilakukannya penelitian ini. Guna mengisi kesenjangan tersebut, penelitian ini memberikan kontribusi sebagai berikut:

1. Merancang arsitektur dan mengimplementasikan asisten digital pembelajaran berbasis Transformer yang terintegrasi dengan MCP, yang secara spesifik diadaptasi untuk kebutuhan pembelajaran Informatika di jenjang SMK.
2. Membuktikan secara praktis bahwa integrasi MCP memungkinkan *chatbot* untuk mengakses dan memanfaatkan data lokal akademik seperti modul, tugas, dan profil siswa sehingga mampu menghasilkan interaksi yang bersifat personal, adaptif, dan kontekstual.
3. Menyediakan metodologi evaluasi yang komprehensif melalui *white-box testing* dengan metrik *cyclomatic complexity* dan validasi ahli praktikalitas, yang dapat menjadi acuan bagi pengembangan sistem serupa di konteks pendidikan vokasi.

Berdasarkan uraian di atas, penelitian ini bertujuan untuk:

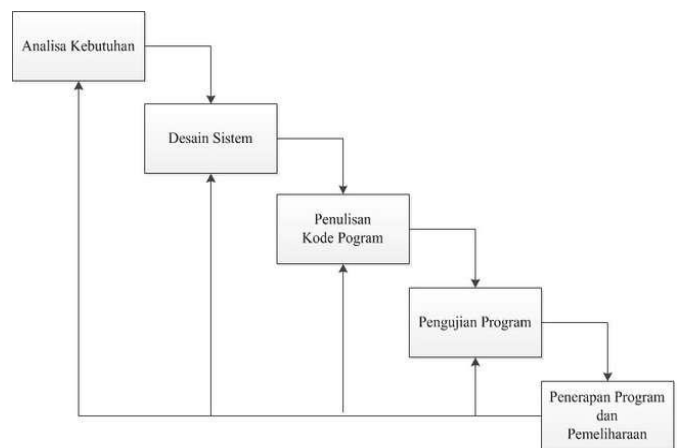
1. Merancang dan mengimplementasikan asisten digital berbasis arsitektur Transformer dengan integrasi MCP untuk mendukung proses pembelajaran mata pelajaran Informatika di tingkat SMK. Sistem ini dibangun menggunakan metode Waterfall, framework Laravel, basis data MySQL, dan API OpenAI.
2. Mengukur kualitas fungsional sistem melalui pengujian *white-box testing* dengan metrik *cyclomatic complexity* pada seluruh form fungsi utama, serta mengevaluasi tingkat kepraktisan aplikasi melalui validasi oleh para ahli.
3. Mendeskripsikan capaian integrasi MCP dalam menyediakan respons yang kontekstual, personal, dan adaptif berdasarkan data lokal yang tersimpan di sistem, sebagai bentuk inovasi penerapan AI dalam mendukung pembelajaran di SMK.

II. METODE PENGEMBANGAN

Penelitian ini menggunakan metode Waterfall sebagai kerangka kerja dalam pengembangan perangkat lunak [12]. Metode Waterfall merupakan pendekatan linear dan berurutan (sequential) pada siklus hidup pengembangan sistem, yang terdiri atas beberapa tahapan utama, yaitu: analisis kebutuhan,

perancangan sistem, implementasi, pengujian, serta pemeliharaan [13].

Pemilihan metode ini didasarkan pada karakteristiknya yang sistematis dan terstruktur, sehingga setiap tahap dapat diselesaikan secara menyeluruh sebelum melanjutkan ke tahap berikutnya. Pendekatan ini meminimalkan terjadinya pengulangan pekerjaan dan memudahkan proses pengendalian proyek [14]. Dengan demikian, kualitas hasil pengembangan perangkat lunak dapat lebih terjamin karena setiap tahapan telah diverifikasi dan divalidasi sebelum melanjutkan ke tahap selanjutnya [15]. Alur proses metode Waterfall yang digunakan dalam penelitian ini ditunjukkan pada Gambar 1.



Gambar 1. Metode Pengembangan Waterfall

A. Analisis Kebutuhan

Tahap ini bertujuan mengidentifikasi dan mendefinisikan kebutuhan sistem secara menyeluruh, baik dari sisi pengguna maupun kebutuhan teknis. Kegiatan pada tahap ini meliputi:

1. Melakukan observasi dan wawancara dengan guru untuk mengetahui masalah dan kebutuhan pembelajaran.
2. Menganalisis sistem yang sudah berjalan dan mengidentifikasi kelemahan atau kekurangannya.
3. Menentukan kebutuhan fungsional (fitur yang harus ada, seperti chatbot, modul pembelajaran, manajemen tugas, dan analisis kinerja siswa) serta kebutuhan nonfungsional (keamanan, performa, ketersediaan).
4. Hasil dari tahap ini adalah dokumen spesifikasi kebutuhan perangkat lunak (Software Requirement Specification/SRS) yang menjadi acuan pada tahap berikutnya.

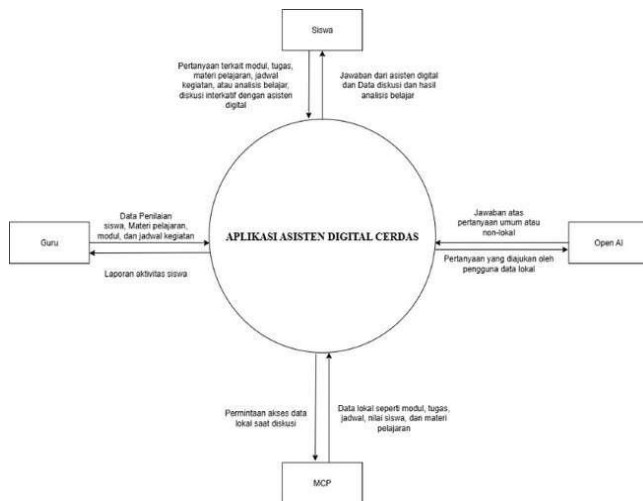
B. Perancangan Sistem

Setelah tahap analisis kebutuhan dilakukan berdasarkan hasil wawancara dan analisis sistem yang ada, langkah selanjutnya adalah tahap perancangan sistem. Perancangan sistem yang baik berperan penting dalam menciptakan komunikasi yang efektif antara pengguna dan sistem. Pada tahap ini dilakukan langkah-langkah sebagai berikut:

1. Merancang Diagram konteks

Diagram konteks (*context diagram*) merupakan model yang menggambarkan hubungan antara sistem dan entitas eksternal secara menyeluruh. Diagram ini menunjukkan bagaimana data diterima dari entitas luar, diproses di dalam sistem, dan menghasilkan keluaran yang dikirim kembali kepada entitas tersebut. Fungsi utama diagram konteks

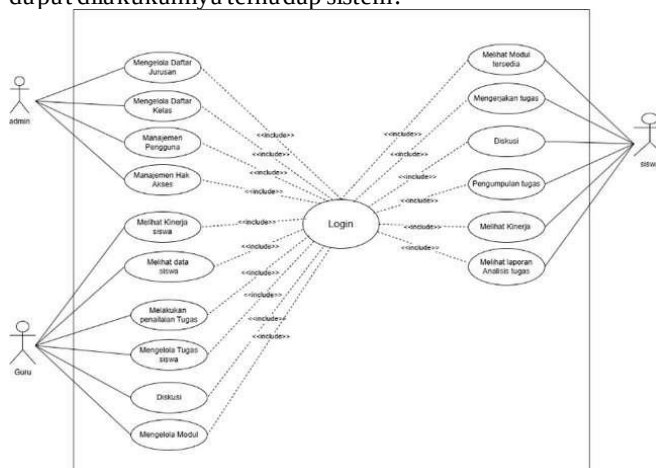
adalah memberikan gambaran ringkas dan jelas mengenai aliran data masuk dan keluar sistem.



Gambar 2. Konteks Diagram

2. Merancang Use Case Diagram

Use case diagram merupakan gambaran scenario dari interaksi antara pengguna dengan sistem. Use case diagram menggambarkan hubungan antara actor dan kegiatan yang dapat dilakukannya terhadap sistem.



Gambar 3. Use Case Diagram

3. Implementasi

Tahap implementasi dilakukan dengan membangun sistem sesuai desain yang telah disusun. Beberapa aspek teknis yang direalisasikan pada tahap ini dijelaskan sebagai berikut:

a. Spesifikasi Model Transformer

Chatbot AI memanfaatkan model berbasis arsitektur Transformer yang disediakan oleh OpenAI melalui integrasi API, sehingga tidak memerlukan pelatihan model dari awal. Model yang digunakan adalah GPT-4o-mini, yang dipanggil melalui endpoint API OpenAI dengan parameter utama sebagai berikut: suhu (*temperature*) diatur pada rentang 0,3–0,5 agar respons tetap relevan secara faktual tetapi tidak terlalu deterministik; *max_tokens* dibatasi maksimal 1024 untuk efisiensi; serta *top_p* diatur pada 0,9. Pemilihan

model ini didasarkan pada kemampuannya dalam memahami konteks pendidikan, menghasilkan respons koheren, serta mendukung pemanggilan fungsi (*function calling*) yang diperlukan untuk integrasi MCP.

b. Detail Desain Integrasi Model Context Protocol (MCP)

Integrasi MCP dirancang sebagai lapisan tengah (*middleware*) antara chatbot dan basis data lokal MySQL. Mekanisme kerjanya adalah sebagai berikut:

- 1) Ketika pengguna mengirimkan pesan, sistem terlebih dahulu mendeteksi topik pertanyaan (modul, tugas, jadwal, atau mata pelajaran) melalui komponen *McpTopikService*.
- 2) Berdasarkan topik yang terdeteksi, sistem menyusun aturan pesan (system prompt) yang membatasi konteks AI pada domain pendidikan Informatika SMK.
- 3) Apabila pertanyaan memerlukan data dari basis data lokal, MCP secara otomatis membentuk kueri SQL yang sesuai (misalnya `SELECT * FROM moduls WHERE ...`), mengeksekusinya melalui koneksi aman ke MySQL, dan mengambil hasilnya.
- 4) Hasil kueri tersebut kemudian diformat dan disisipkan ke dalam konteks percakapan sebagai tool output, sehingga model dapat merangkai jawaban akhir yang bersifat personal dan berbasis data aktual.
- 5) Data siswa yang diakses terbatas pada informasi yang relevan dengan pertanyaan, seperti status tugas, progres belajar, dan daftar modul yang telah dipelajari. Data sensitif seperti kata sandi tidak pernah disertakan dalam konteks yang dikirim ke model.

c. Mekanisme Keamanan dan Otorisasi Data

Untuk menjaga kerahasiaan dan integritas data siswa, sistem menerapkan sejumlah mekanisme keamanan:

- 1) Autentikasi dan otorisasi berbasis peran (RBAC): Setiap pengguna (admin, guru, siswa) hanya dapat mengakses data sesuai dengan hak aksesnya. Implementasi menggunakan Laravel middleware untuk memverifikasi peran sebelum setiap permintaan diproses.
- 2) Enkripsi data: Kata sandi pengguna dienkripsi menggunakan algoritma bcrypt. Komunikasi antara klien dan server dilindungi melalui protokol HTTPS.
- 3) Perlindungan terhadap serangan injeksi SQL: Seluruh kueri ke basis data menggunakan prepared statements dan parameter binding yang disediakan oleh Laravel Eloquent ORM, sehingga input pengguna tidak pernah langsung disisipkan ke dalam string kueri.
- 4) Pembatasan akses API MCP: Koneksi MCP ke basis data lokal hanya dapat dilakukan oleh komponen internal sistem; tidak ada akses langsung dari sisi klien. API key OpenAI disimpan dalam variabel lingkungan (environment variable) dan tidak diekspos ke kode sumber publik.
- 5) Logging dan audit: Seluruh akses ke data siswa melalui MCP dicatat dalam log aktivitas untuk keperluan audit dan deteksi anomali.

4. Pengujian

Tahap ini bertujuan untuk memastikan seluruh komponen sistem berjalan sesuai dengan spesifikasi yang telah ditetapkan. Pengujian dilakukan melalui dua pendekatan.

a. Pengujian Fungsional (White-Box Testing)

Pengujian fungsional dilakukan dengan metode white-box testing menggunakan metrik cyclomatic complexity. Teknik ini memungkinkan pemeriksaan menyeluruh terhadap struktur internal dan alur logika kode program. Sebanyak 49 form fungsional utama diuji, mencakup modul autentikasi, manajemen modul pembelajaran, manajemen tugas, interaksi chatbot, analisis kinerja siswa, serta pengelolaan data master (jurusan, kelas, dan mata pelajaran) untuk tiga aktor: admin, guru, dan siswa. Setiap jalur eksekusi pada setiap form diverifikasi untuk memastikan tidak terdapat kesalahan logika, *dead code*, maupun jalur yang tidak tercapai. Hasil pengujian menunjukkan bahwa seluruh jalur eksekusi berhasil dijalankan tanpa ditemukan kesalahan logika (tingkat keberhasilan 100%).

b. Validasi Ahli

Selain pengujian fungsional, dilakukan validasi oleh 3 orang praktisi/programmer yang berpengalaman di bidang pengembangan perangkat lunak. Instrumen yang digunakan berupa lembar penilaian praktikalitas yang mencakup aspek fungsionalitas, kemudahan navigasi, kejelasan informasi, dan relevansi konten. Hasil validasi menempatkan aplikasi pada kategori “Sangat Praktis”, yang menunjukkan bahwa sistem layak digunakan sebagai media pembelajaran.

c. Penggunaan Sistem oleh Guru

Sebagai indikator operasional, sistem telah digunakan di lingkungan SMK oleh 8 orang guru yang telah memasukkan data 41 siswa ke dalam aplikasi. Penggunaan ini menunjukkan bahwa sistem dapat berfungsi dalam konteks nyata, meskipun tidak dilakukan pengukuran usability secara terstruktur terhadap pengguna akhir. Data penggunaan ini melengkapi bukti bahwa sistem siap dioperasikan di lingkungan sekolah.

5. Pemeliharaan

Tahap pemeliharaan dilakukan setelah sistem diimplementasikan dan digunakan secara aktif oleh guru dan siswa. Kegiatan pemeliharaan meliputi pemantauan kinerja sistem, perbaikan bug atau kesalahan yang ditemukan, serta pembaruan fitur sesuai kebutuhan pengguna. Pemeliharaan juga mencakup penyesuaian integrasi Model Context Protocol (MCP) dan API OpenAI apabila terjadi perubahan spesifikasi atau pembaruan layanan dari penyedia. Selain itu, dilakukan pembaruan basis data modul dan materi pembelajaran secara berkala agar informasi yang disajikan tetap akurat dan relevan dengan kurikulum yang berlaku.

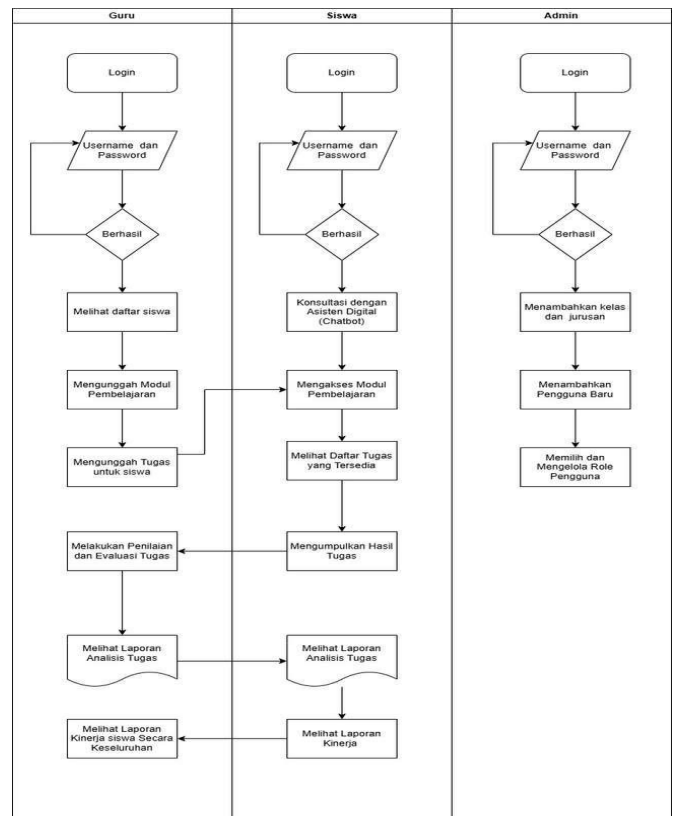
III. HASIL DAN PEMBAHASAN

A. Analisis sistem yang diusulkan

Analisis ini digunakan untuk mengidentifikasi kebutuhan yang akan dikembangkan pada sistem yang diusulkan.

Berdasarkan flowmap, alur kerja sistem dapat dijelaskan sebagai berikut:

1. Guru melakukan login, mengakses daftar siswa, mengunggah modul dan tugas, melakukan penilaian, serta memantau laporan analisis dan kinerja siswa.
2. Siswa melakukan login, berkonsultasi dengan asisten digital, mengakses modul pembelajaran, melihat serta mengumpulkan tugas, dan meninjau laporan analisis maupun kinerja.
3. Admin melakukan login, menambahkan kelas dan jurusan, mendaftarkan pengguna baru, serta mengatur role atau hak akses pengguna.

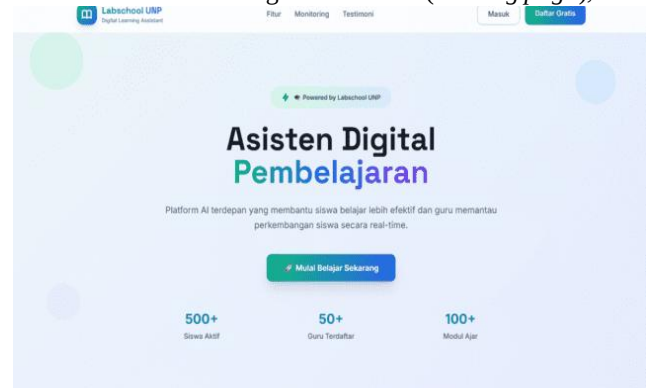


Gambar 4. Flow Map Sistem Diusulkan

B. Hasil Rancangan Sistem

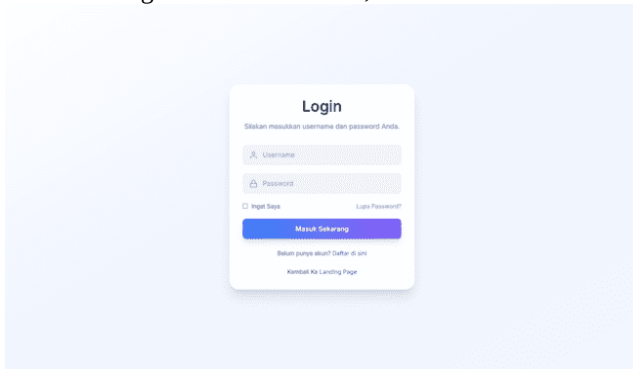
Implementasi sistem menghasilkan aplikasi asisten digital berbasis web yang terdiri atas sejumlah halaman utama sesuai dengan kebutuhan pengguna. Antarmuka yang dikembangkan mencakup:

1. Halaman Beranda sebagai titik masuk (*landing page*),



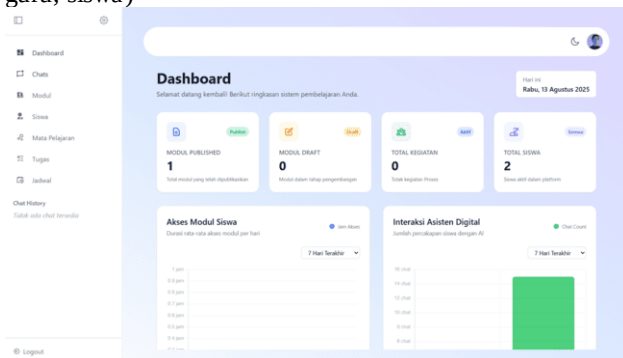
Gambar 5. Halaman Beranda

2. Halaman Login untuk autentikasi,



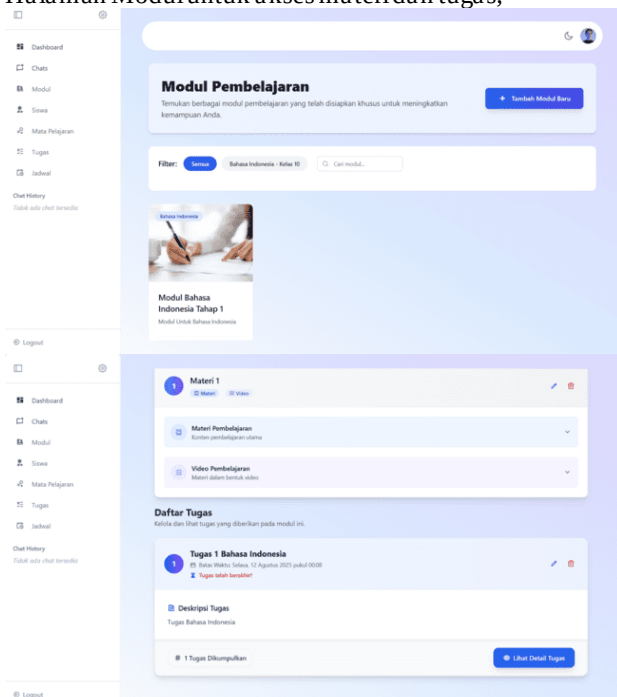
Gambar6. Halaman Login

3. Dashboard yang menyesuaikan peran pengguna (admin, guru, siswa)



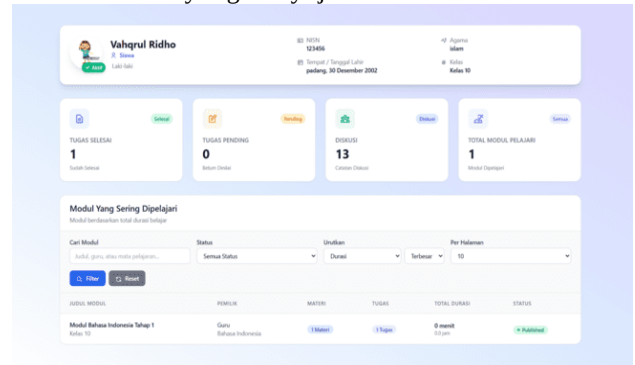
Gambar 7. Halaman Dasboard

4. Halaman Modul untuk akses materi dan tugas,



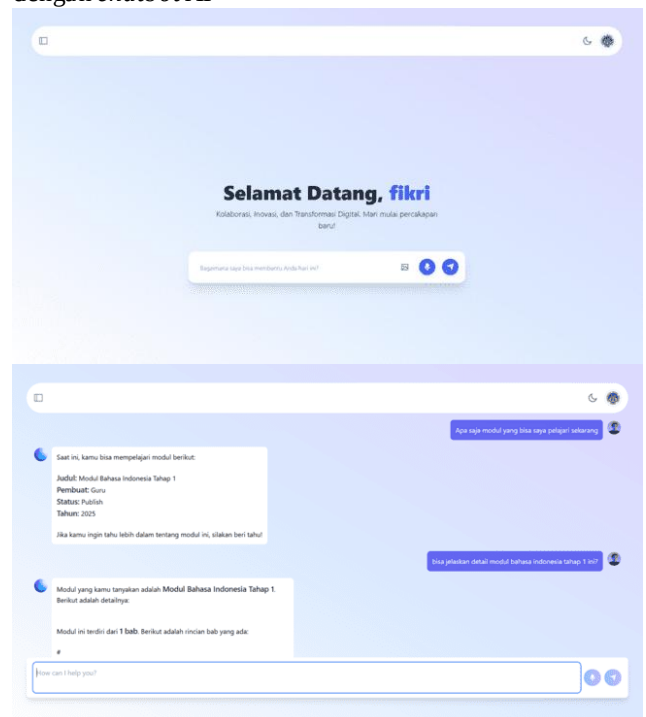
Gambar8. Halaman Modul

5. Halaman Siswa yang menyajikan data dan analisis individu



Gambar9. Halaman Siswa

6. Halaman Diskusi sebagai ruang interaksi dengan chatbot AI



Gambar 10. Halaman Diskusi

Seluruh antarmuka dirancang dengan prinsip *user-friendly* dan responsif, mengikuti *wireframe* yang telah ditetapkan pada tahap perancangan.

C. Hasil Pengujian Sistem

1. Pengujian Fungsional dengan White-Box Testing

Pengujian fungsional dilakukan terhadap 49 form/modul utama yang mencakup seluruh aktor (admin, guru, siswa) dengan metode *white-box testing* dan metrik *cyclomatic complexity* ($V(G) = E - N + 2$). Setiap fungsi diperiksa struktur internalnya untuk memastikan seluruh jalur logika tereksekusi tanpa kesalahan. Salah satu contoh pengujian adalah fungsi Login, yang memiliki *flowgraph* dengan 4 *node* dan 3 *edge*, sehingga $V(G) = 3 - 4 + 2 = 1$.

I. LoginController.php

Nama File : LoginController.php

Deskripsi :

Berikut merupakan codingan dan flowgraph yang di peroleh

1. Function Login

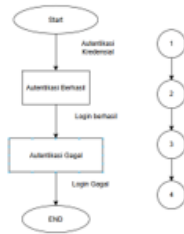
```
public function login($username, $password): RedirectResponse
{
    $request = $this->getRequest();
    $username = $request->get('username');
    $password = $request->get('password');

    return $this->redirectToRoute('dashboard', ['username' => $username]);
}

public function authenticate(): void
{
    $this->redirectToRoute('login');

    if ($this->isAuthenticated()) {
        $this->redirectToRoute('dashboard');
    } else {
        $this->redirectToRoute('login');
    }
}

public function logout(): void
{
    $this->redirectToRoute('login');
}
```



Menghitung cyclomatic complexity:

$$V(G) = E - N + 2$$

$$V(G) = 3 - 4 + 2 = 1$$

Ket :

- E = 3 (jumlah Panah)
- N = 4 (jumlah Node)

Berdasarkan perhitungan cyclomatic complexity untuk function destroy didapatkan 1 independent path dari potongan code pada gambar [0.0] seperti yang terlihat pada tabel [0.0]

Path	Jalur	Skenario	Hasil
1	1-2-3-4	1. Start: Permintaan untuk login. 2. Autentikasi: Sistem mencocokkan kredensial (username & password) ke database. 3. Autentikasi Berhasil: Kredensial cocok, sistem membuat sesi/login. 4. Login Berhasil: Pengguna diarahkan ke halaman dashboard. 5. End: Proses selesai.	

Gambar 11. Gambar Node Login

Hasil perhitungan menunjukkan bahwa fungsi Login memiliki 1 jalur independen. Pengujian pada jalur tersebut (kredensial valid → autentikasi berhasil → arahkan ke dashboard) berhasil dijalankan tanpa *error* atau *dead code*. Prosedur yang sama diterapkan pada seluruh 49 form. Secara keseluruhan, seluruh jalur eksekusi pada setiap form berhasil dijalankan (tingkat keberhasilan 100%) dan tidak ditemukan kesalahan logika. Hasil ini mengonfirmasi bahwa sistem telah memenuhi spesifikasi fungsional.

2. Validasi Ahli (Praktisi)

Validasi kualitas sistem dilakukan oleh 3 orang praktisi/programmer berpengalaman di bidang pengembangan perangkat lunak. Instrumen yang digunakan adalah lembar penilaian praktikalitas yang mencakup empat aspek: (1) fungsionalitas, (2) kemudahan navigasi, (3) kejelasan informasi, dan (4) relevansi konten. Hasil penilaian dari ketiga praktisi menunjukkan bahwa aplikasi memperoleh rata-rata skor pada kategori “Sangat Praktis”. Dengan demikian, dari sudut pandang profesional, sistem dinilai layak digunakan sebagai media pembelajaran. Validasi ini juga secara khusus mengapresiasi kemampuan *chatbot* dalam memberikan jawaban yang sesuai dengan isi modul, yang menunjukkan berfungsinya integrasi MCP.

3. Penggunaan Sistem oleh Guru

Sebagai bukti operasional awal, sistem telah digunakan oleh 8 orang guru yang melakukan input data 41 siswa ke dalam aplikasi. Guru memanfaatkan fitur unggah modul, pemberian tugas, dan pemantauan aktivitas belajar siswa tanpa kendala teknis berarti. Meskipun tidak dilakukan pengukuran *usability* secara terstruktur terhadap pengguna akhir, penggunaan nyata ini menjadi indikator

bahwa sistem dapat beroperasi di lingkungan sekolah dan diterima oleh pengguna.

4. Pengukuran Waktu Respons Chatbot

Untuk melengkapi evaluasi performa, dilakukan pengukuran waktu respons *chatbot* dari sisi klien melalui browser *developer tools* (tab *Network*). Pengukuran dilakukan pada dua kelompok respons berdasarkan ukuran data yang diterima: 4.5 kB (respons tanpa data MCP) dan 6.8 kB (respons dengan data MCP). Masing-masing kelompok dicatat sebanyak 11 dan 5 sampel. Hasilnya dirangkum pada Tabel 1.

Tabel 1
Waktu Respons Chatbot

Kelompok	Rata-rata (ms)	Median	Min (ms)	Maks (ms)
4.5kb (Tanpa MCP)	230,1	230,1	200	278
6.8 kb (Dengan MCP)	216,4	216	208	227

Data menunjukkan bahwa waktu respons kedua kelompok berada di bawah 280 ms, dan akses tambahan ke basis data lokal melalui MCP tidak memperlambat interaksi secara signifikan. Sebaliknya, respons dengan MCP tercatat sedikit lebih cepat, mengindikasikan efisiensi kueri lokal dan ringannya data yang diambil. Dari perspektif pengguna, seluruh interaksi *chatbot* berlangsung dalam rentang waktu yang nyaman dan responsif.

D. Pembahasan

Hasil pengujian *white-box* pada 49 form menunjukkan bahwa seluruh jalur eksekusi berhasil dijalankan tanpa ditemukan kesalahan logika, *dead code*, maupun *unreachable path*. Temuan ini mengonfirmasi bahwa implementasi perangkat lunak, termasuk integrasi *Model Context Protocol* (MCP), telah sesuai dengan rancangan dan memiliki kualitas kode yang baik. Integrasi MCP sendiri terbukti menjadi komponen kunci dalam menghadirkan respons yang kontekstual dan personal.

Ketika siswa mengajukan pertanyaan terkait tugas, MCP secara otomatis membentuk kueri ke basis data lokal berdasarkan ID siswa, mengambil data yang relevan, dan mengembalikan jawaban yang spesifik—bukan sekadar informasi generik. Validasi oleh tiga praktisi memperkuat temuan ini, di mana *chatbot* dinilai mampu memberikan jawaban yang konsisten dengan isi modul, menandakan bahwa integrasi MCP berfungsi secara efektif. Dari sisi adaptivitas, sistem merekam aktivitas belajar secara otomatis, mencakup durasi akses modul, progres tugas, dan frekuensi belajar.

Data tersebut ditampilkan dalam bentuk grafik dan laporan yang memungkinkan guru mengidentifikasi siswa yang memerlukan perhatian lebih serta menyesuaikan strategi pengajaran. Mekanisme ini, meskipun belum diukur melalui survei *usability*, telah terverifikasi secara fungsional dan digunakan langsung oleh 8 orang guru terhadap 41 siswa. Dalam aspek keamanan, sistem menerapkan perlindungan berlapis, meliputi autentikasi berbasis peran (*Role-Based Access Control*), penggunaan *prepared statements* untuk mencegah *SQL injection*, enkripsi kata sandi dengan *bcrypt*,

serta penyimpanan kunci API OpenAI pada variabel lingkungan yang tidak terpapar ke klien.

Pengujian *white-box* tidak menemukan jalur logika yang memungkinkan eskalasi hak akses maupun kebocoran data, sehingga integrasi MCP dapat dinyatakan aman secara fungsional. Sementara itu, pengukuran waktu respons *chatbot* melalui *browser developer tools* (Tabel 1) menunjukkan bahwa integrasi MCP tidak menambah latensi yang signifikan. Rata-rata waktu respons kelompok tanpa MCP sebesar 230,1 ms dan kelompok dengan MCP sebesar 216,4 ms, keduanya jauh di bawah ambang batas kenyamanan pengguna untuk sistem percakapan (< 2 detik).

Dengan demikian, penambahan akses data lokal tidak mengorbankan performa interaksi. Meskipun hasil yang diperoleh cukup menjanjikan, penelitian ini memiliki keterbatasan yang perlu dicatat. Akurasi jawaban *chatbot* belum diukur menggunakan metrik kuantitatif seperti ROUGE, BLEU, atau penilaian ahli berbasis rubrik, sehingga kualitas konten respons secara objektif belum terkuantifikasi. Selain itu, evaluasi pengguna akhir masih terbatas pada 8 guru dan 41 siswa tanpa menerapkan instrumen *usability* terstandar misalnya *System Usability Scale* (SUS) maupun teknik *sampling* yang ketat.

Oleh karena itu, penelitian selanjutnya disarankan untuk melaksanakan uji akurasi jawaban terhadap konten modul, menyelenggarakan survei *usability* pada sampel yang lebih besar, serta mengukur performa sistem dalam kondisi beban banyak pengguna secara simultan.

IV. KESIMPULAN

Berdasarkan hasil pengujian dan validasi yang telah dilakukan, pengembangan asisten digital berbasis arsitektur Transformer dengan integrasi Model Context Protocol (MCP) telah berhasil mencapai tujuan yang ditetapkan. Pengujian *white-box* pada 49 form fungsional membuktikan bahwa seluruh jalur eksekusi berjalan tanpa kesalahan logika, yang menandakan bahwa implementasi sistem telah memenuhi spesifikasi fungsional secara penuh. Validasi oleh tiga praktisi menempatkan aplikasi pada kategori “Sangat Praktis”, menegaskan kelayakan sistem sebagai media pembelajaran. Pengukuran waktu respons *chatbot* menunjukkan bahwa integrasi MCP tidak menimbulkan latensi tambahan yang berarti—rata-rata respons berada di bawah 250 ms, jauh di bawah ambang batas kenyamanan pengguna. Mekanisme perekaman aktivitas belajar secara otomatis memberi dasar bagi adaptivitas pembelajaran, sementara penerapan perlindungan berlapis RBAC, prepared statements, enkripsi bcrypt, dan isolasi kunci API memastikan keamanan integrasi MCP. Meskipun demikian, penelitian ini memiliki keterbatasan, terutama belum dilakukannya pengukuran akurasi jawaban *chatbot* secara kuantitatif serta belum diterapkannya instrumen *usability* terstandar pada sampel pengguna yang lebih luas. Oleh karena itu, pengembangan selanjutnya diarahkan pada: (1) pengukuran akurasi respons menggunakan metrik evaluasi bahasa, (2) pelaksanaan survei *usability* dengan instrumen baku pada populasi siswa dan guru yang lebih besar, serta (3) pengujian performa sistem dalam kondisi beban banyak pengguna secara simultan. Dengan penyempurnaan tersebut, asisten digital ini berpotensi menjadi platform pembelajaran yang interaktif, adaptif, dan aman, serta

mampu mendukung transformasi digital di lingkungan pendidikan kejuruan.

DAFTAR PUSTAKA

- [1] N. Chalesa Fitriani, S. Rafi, and A. Zaki Yamani, “DIGITALISASI LEGAL DRAFTING MELALUI ARTIFICIAL INTELLIGENCE: PELUANG DAN TANTANGAN MASA DEPAN DOKUMEN HUKUM DI INDONESIA,” 2025, [Online]. Available: <https://shariajournal.com/index.php/IJJEL/>
- [2] IMANUEL SETYO BUDI, PUTRAYASA, and WISUDARIANI, “PERAN DAN TANTANGAN PENGGUNAAN ARTIFICIAL INTELLIGENCE DALAM INOVASI PENGEMBANGAN KURIKULUM PEMBELAJARAN BAHASA INDONESIA MASA DEPAN,” Dec. 2024.
- [3] A. Vaswani *et al.*, “Attention Is All You Need.”
- [4] G. Setiawan Nurohim, B. Al Amin, and H. Ary Setya di, “Effectiveness And Efficiency Of LLM Models Vs Traditional Machine Learning In Sentiment Analysis Of Indonesian Language Product Reviews,” 2025.
- [5] R. Aulia, A. Ruslani, and M. D. Fauzan, “Pengembangan Model Transformer untuk Analisis Sentimen dan Ringkasan Ulasan E-commerce Berbahasa Indonesia,” vol. 4, no. 1, 2025.
- [6] K. Shim and W. Sung, “A Comparison of Transformer, Convolutional, and Recurrent Neural Networks on Phoneme Recognition,” Oct. 2022, [Online]. Available: <http://arxiv.org/abs/2210.00367>
- [7] A. Zein, “DAMPAK PENGGUNAAN CHATGPT PADA DUNIA PENDIDIKAN,” *JITU: Jurnal Informatika Utama*, vol. 1, no. 2, 2023, doi: 10.55903/jitu.v1i2.151.
- [8] Uban Subandi and Supardi US, “Integrasi Teknologi AI dalam Pembelajaran STEM di Sekolah Menengah: Perspektif Personalisasi, Tantangan, dan Implikasi,” *Tahun*, vol. 6, pp. 89–104, 2024, doi: 10.62383/bilangan.v2i6.320.
- [9] A. Brasoveanu, M. Moodie, and R. Agrawal, “Making REST APIs Agent-Ready: From OpenAPI to Model Context Protocol Servers for Tool-Augmented LLMs,” in *CEUR Workshop Proceedings*, CEUR-WS, 2020, pp. 1–9. doi: 10.1145/nnnnnnnn.nnnnnnnn.
- [10] C.-M. Chan *et al.*, “RQ-RAG: Learning to Refine Queries for Retrieval Augmented Generation,” Mar. 2024, [Online]. Available: <http://arxiv.org/abs/2404.00610>
- [11] Siti Zubaidah, “KETERAMPILAN ABAD KE-21: KETERAMPILAN YANG DIAJARKAN MELALUI PEMBELAJARAN,” Dec. 2016.
- [12] A. Abdul Wahid Sekolah Tinggi Manajemen Informatika dan Komputer Sumedang, “Analisis Metode Waterfall Untuk Pengembangan Sistem Informasi,” 2020. [Online]. Available: <https://www.researchgate.net/publication/346397070>
- [13] A. Z. D. Nur Adiya, D. L. Anggraeni, and Ilham Albana, “Analisa Perbandingan Penggunaan Metodologi Pengembangan Perangkat Lunak (Waterfall, Prototype, Iterative, Spiral, Rapid Application Development (RAD)),” *Merkurius : Jurnal Riset Sistem Informasi dan*

Teknik Informatika, vol. 2, no. 4, pp. 122–134, Jun. 2024, doi: 10.61132/mercurius.v2i4.148.

- [14] F. Marwati and S. Farizy, “PERANCANGAN SISTEM INFORMASI PENGAJUAN CUTI KARYAWAN MENGGUNAKAN METODE WATERFALL BERBASIS WEB,” 2025.
- [15] A. G. Topandi and D. A. Andriati, “EVALUASI KINERJA KARYAWAN DAN SISTEM PENGGAJIAN BERBASIS WEBSITE MENGGUNAKAN METODE WATERFALL,” 2025, doi: 10.58290/jukomtek.v.