

# Implementasi Gemini Api untuk Penyederhanaan Kalimat Marketing Berbasis Web Menggunakan REACTJS dan FASTAPI

<sup>1\*</sup>Farhan Maulana, <sup>2</sup>Johan Harlan, <sup>3</sup>Cahyawati Diah Kusumarini

<sup>1,2,3</sup>Informatika, Fakultas Teknologi Industri, Universitas Gunadarma, Indonesia

\*Korespondensi : [farhanslow33@gmail.com](mailto:farhanslow33@gmail.com)

Submit : 11 Mei 2026 | Diterima : 04 Jun 2026 | Terbit : 09 Jun 2026

## ABSTRACT

*The rapid growth of digital marketing has increased the demand for promotional content that is both informative and persuasive. However, many marketing texts still contain long and complex sentences, as well as excessive technical terminology, which can reduce the effectiveness of message delivery to target audiences. This study aims to develop a web-based application for automatically simplifying marketing texts using the Gemini API. The system was developed following the Software Development Life Cycle (SDLC) Waterfall model, utilizing ReactJS for the frontend and FastAPI for the backend. The application is designed to identify several sentence complexity indicators, including passive sentences, compound sentences, complex words, and marketing jargon. The simplification results are presented in a more readable format, accompanied by readability scores, text statistics, and a summary of modifications. System evaluation was conducted using Black Box Testing to ensure that all functionalities operated according to requirements, while the System Usability Scale (SUS) was employed to assess the system's usability. The testing results demonstrated that all application features functioned properly across 15 testing scenarios. Furthermore, usability testing involving 50 respondents produced an average SUS score of 90.85, which falls within the Excellent, Grade A, and Acceptable categories. These findings indicate that the application effectively assists users in producing promotional texts that are clearer, more effective, and easier to understand. Therefore, the developed system has significant potential to support digital marketing communication needs, particularly for businesses, small and medium-sized enterprises (SMEs), and content creators in creating more communicative promotional content.*

**Keywords:** Black Box Testing, Gemini API, Readability, Sentence Simplification, Waterfall Model.

## ABSTRAK

Perkembangan pemasaran digital mendorong penggunaan konten promosi yang informatif dan persuasif. Namun, banyak teks marketing masih menggunakan kalimat panjang, kompleks, dan dipenuhi istilah teknis sehingga menurunkan efektivitas penyampaian pesan kepada audiens. Penelitian ini bertujuan mengembangkan aplikasi berbasis web untuk menyederhanakan teks marketing secara otomatis menggunakan *Gemini API*. Sistem dibangun menggunakan metode *Software Development Life Cycle* (SDLC) model *Waterfall* dengan teknologi ReactJS pada sisi *frontend* dan FastAPI pada sisi *backend*. Aplikasi dirancang untuk mendeteksi beberapa indikator kompleksitas kalimat, yaitu kalimat pasif, kalimat majemuk, kata kompleks, dan jargon marketing. Hasil penyederhanaan ditampilkan dalam bentuk teks yang lebih mudah dipahami disertai skor keterbacaan, statistik teks, dan ringkasan perubahan. Pengujian sistem dilakukan menggunakan metode *Black Box Testing* untuk memastikan seluruh fungsi berjalan sesuai kebutuhan serta metode *System Usability Scale* (SUS) untuk mengukur tingkat kegunaan sistem. Hasil pengujian menunjukkan seluruh fitur aplikasi berjalan dengan baik berdasarkan 15 skenario pengujian. Selain itu, pengujian SUS terhadap 50 responden memperoleh skor rata-rata 90,85 yang termasuk kategori *Excellent*, *Grade A*, dan *Acceptable*. Hasil tersebut menunjukkan bahwa aplikasi mampu membantu pengguna menghasilkan teks promosi yang lebih jelas, efektif, dan mudah dipahami. Dengan demikian, sistem yang dikembangkan memiliki potensi untuk mendukung kebutuhan komunikasi pemasaran digital, khususnya bagi pelaku bisnis, UMKM, dan *content creator* dalam menyusun konten promosi yang lebih komunikatif.

**Kata Kunci:** Penyederhanaan kalimat, Gemini API, keterbacaan, *Waterfall*, *Black Box Testing*

## PENDAHULUAN

Perkembangan teknologi informasi telah mengubah pola komunikasi pemasaran menjadi lebih cepat dan berbasis digital. Dalam pemasaran digital, penggunaan bahasa yang efektif menjadi faktor penting untuk menarik perhatian konsumen dan meningkatkan keberhasilan penyampaian pesan. Teks promosi yang terlalu panjang, kompleks, dan dipenuhi istilah teknis sering kali menyebabkan audiens kesulitan memahami informasi yang disampaikan. Kondisi tersebut dapat menurunkan efektivitas komunikasi serta mengurangi daya tarik suatu produk atau layanan. Keterbacaan menjadi salah satu aspek penting dalam penyusunan konten marketing. Tingkat keterbacaan dipengaruhi oleh struktur kalimat, panjang kalimat, dan pemilihan kosakata yang digunakan dalam suatu teks. Semakin sederhana struktur bahasa yang digunakan, maka semakin mudah pesan diterima oleh pembaca.

Perkembangan teknologi *Artificial Intelligence* (AI), khususnya pada bidang *Natural Language Processing* (NLP), membuka peluang untuk melakukan penyederhanaan teks secara otomatis. Salah satu teknologi yang banyak digunakan saat ini adalah *Large Language Model* (LLM). Model ini mampu memahami konteks bahasa dan menghasilkan teks yang lebih natural serta mudah dipahami. Salah satu LLM yang berkembang pesat adalah Gemini AI yang dikembangkan oleh Google. Melalui *Gemini API*, proses penyederhanaan teks dapat dilakukan secara otomatis dengan mempertahankan makna dan tujuan komunikasi dari teks asli.

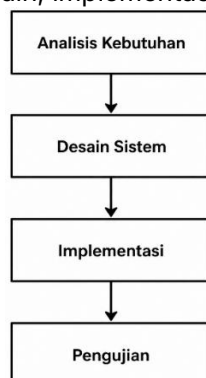
Beberapa penelitian sebelumnya telah memanfaatkan teknologi berbasis LLM dalam pengolahan teks. Penelitian oleh (Saputra et al., 2024) menggunakan *ChatGPT API* untuk membangun aplikasi peringkasan teks berbasis web. Penelitian lain oleh (Cahaya Kamilla et al., 2024) memanfaatkan *Gemini API* untuk menghasilkan deskripsi produk otomatis pada aplikasi kasir mobile. Selain itu, (Hariyadi et al., 2024) menerapkan *Gemini API* pada aplikasi pameran berbasis web untuk menghasilkan deskripsi karya secara otomatis. Namun, penelitian terkait penyederhanaan teks marketing berbasis web yang dilengkapi pengukuran keterbacaan masih terbatas.

Berdasarkan permasalahan tersebut, penelitian ini mengembangkan aplikasi berbasis web untuk menyederhanakan teks marketing secara otomatis menggunakan *Gemini API*. Sistem dirancang untuk mendeteksi kalimat pasif, kalimat majemuk, kata kompleks, dan jargon marketing, kemudian mengubahnya menjadi kalimat yang lebih sederhana dan mudah dipahami. Selain menghasilkan teks yang telah disederhanakan, aplikasi juga menampilkan skor keterbacaan sebagai indikator tingkat kemudahan teks untuk dipahami pengguna.

Pengembangan sistem menggunakan metode *Software Development Life Cycle* (SDLC) model *Waterfall* karena memiliki tahapan yang sistematis dan terstruktur. Teknologi ReactJS digunakan pada sisi *frontend*, sedangkan FastAPI digunakan untuk membangun layanan *backend*. Pengujian dilakukan menggunakan metode *Black Box Testing* untuk memastikan fungsi sistem berjalan sesuai kebutuhan serta *System Usability Scale* (SUS) untuk mengukur tingkat kegunaan aplikasi berdasarkan pengalaman pengguna.

## METODE PENELITIAN

Penelitian ini menggunakan metode *Software Development Life Cycle* (SDLC) model *Waterfall*. Model ini dipilih karena memiliki tahapan pengembangan yang sistematis dan terstruktur sehingga memudahkan proses pembangunan aplikasi. Tahapan utama dalam metode *Waterfall* meliputi analisis, desain, implementasi, dan pengujian.



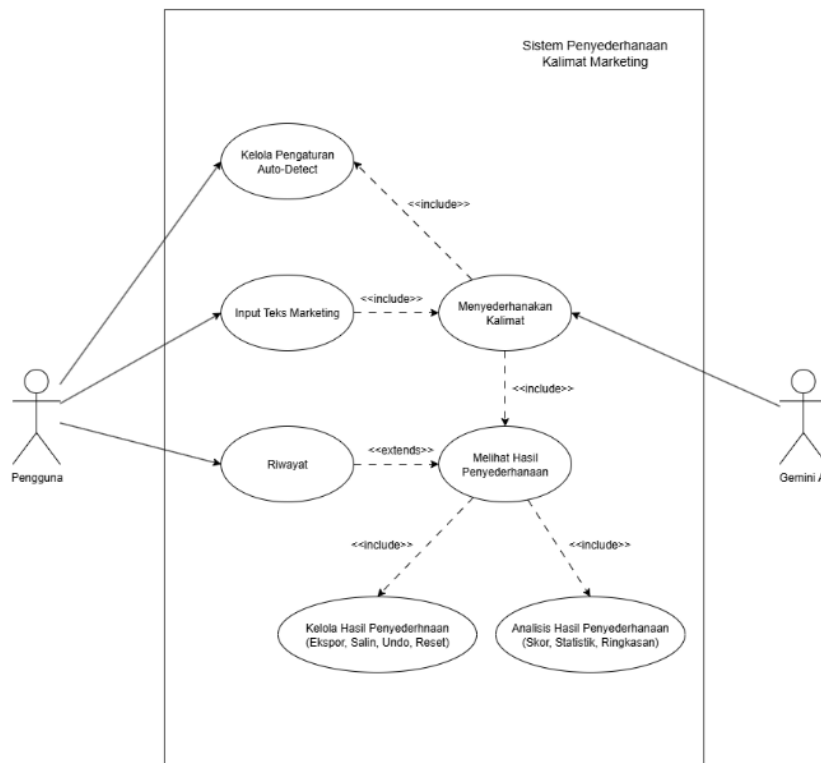
**Gambar 1. Alur Penelitian**

### Analisis kebutuhan

Analisis kebutuhan dilaksanakan guna mengidentifikasi layanan utama yang akan ditawarkan oleh situs web penyederhanaan kalimat pemasaran, sekaligus menentukan perangkat pendukung yang diperlukan untuk proses pengembangannya. Kebutuhan fungsional meliputi fitur input teks marketing, deteksi otomatis kalimat kompleks, proses penyederhanaan menggunakan *Gemini API*, tampilan hasil penyederhanaan, skor keterbacaan, statistik teks, serta penyimpanan riwayat hasil penyederhanaan. Selain itu, sistem menyediakan fitur pengaturan untuk mengaktifkan atau menonaktifkan jenis deteksi tertentu sesuai kebutuhan pengguna. Kebutuhan non-fungsional meliputi spesifikasi hardware dan software yang dibutuhkan untuk membangun serta mengoperasikan sistem.

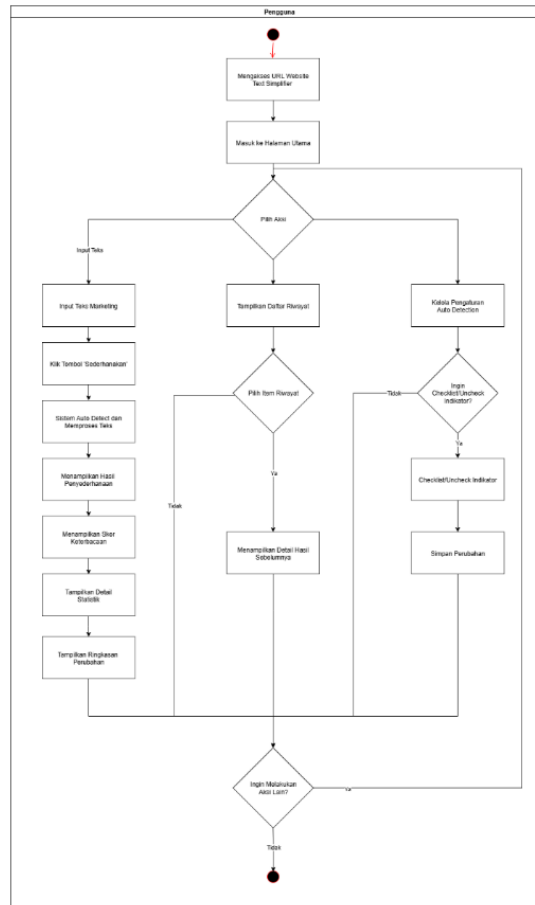
### Desain Sistem

Desain aplikasi merupakan tahapan krusial dalam pengembangan sistem yang bertujuan menyajikan gambaran komprehensif serta struktur arsitektural sistem yang akan dibangun, meliputi pemodelan dengan *Unified Modelling Language (UML)*, struktur navigasi, dan desain antarmuka pengguna. Tahap desain dilakukan untuk menggambarkan struktur dan alur kerja sistem sebelum proses implementasi. Desain sistem meliputi struktur navigasi, *Use Case Diagram*, dan *Activity Diagram*.



Gambar 2. Use Case Diagram

*Use Case Diagram* digunakan untuk menggambarkan interaksi antara pengguna dengan sistem. Pengguna dapat memasukkan teks marketing, mengatur fitur *auto-detection*, melihat hasil penyederhanaan, dan mengakses riwayat penyederhanaan. Sistem kemudian memproses teks menggunakan *Gemini API* dan menampilkan hasil penyederhanaan beserta skor keterbacaan.



**Gambar 3. Activity Diagram**

*Activity Diagram* menggambarkan alur proses aplikasi dimulai dari input teks, proses deteksi kalimat kompleks, penyederhanaan teks, hingga penampilan hasil akhir kepada pengguna.

### Implementasi Sistem

Implementasi sistem dilakukan menggunakan ReactJS pada sisi *frontend* dan FastAPI pada sisi *backend*. ReactJS digunakan untuk membangun antarmuka pengguna yang interaktif dan responsif, sedangkan FastAPI digunakan untuk mengelola komunikasi antara sistem dan *Gemini API*. Proses penyederhanaan dimulai ketika pengguna memasukkan teks marketing ke dalam sistem. Selanjutnya, sistem melakukan deteksi terhadap indikator kompleksitas kalimat, seperti kalimat pasif, kalimat majemuk, kata kompleks, dan jargon marketing. Teks kemudian dikirim ke *Gemini API* untuk diproses menjadi kalimat yang lebih sederhana dan mudah dipahami. Hasil penyederhanaan ditampilkan dalam bentuk teks sederhana disertai skor keterbacaan, detail statistik, dan ringkasan perubahan. Data riwayat penyederhanaan disimpan menggunakan *localStorage* agar dapat diakses kembali oleh pengguna.

### Pengujian Sistem

Pengujian sistem dilakukan menggunakan metode *Black Box Testing* dan *System Usability Scale (SUS)*. *Black Box Testing* digunakan untuk memastikan seluruh fungsi sistem berjalan sesuai kebutuhan. Pengujian dilakukan terhadap fitur input teks, proses penyederhanaan, tampilan hasil, pengaturan sistem, dan penyimpanan riwayat. Selanjutnya, pengujian SUS dilakukan dengan melibatkan 50 responden untuk mengevaluasi tingkat kemudahan penggunaan aplikasi. Hasil pengujian dihitung menggunakan skala SUS dan diinterpretasikan berdasarkan kategori *Acceptability*, *Grade*, dan *Adjective Rating*.

## HASIL DAN PEMBAHASAN

Penelitian ini menghasilkan aplikasi penyederhanaan kalimat marketing berbasis web dengan memanfaatkan *Gemini API* sebagai inti proses penyederhanaan teks. Pengembangan sistem dilakukan menggunakan metode *Software Development Life Cycle (SDLC)* model

*Waterfall* yang terdiri dari tahapan analisis kebutuhan, perancangan sistem, implementasi, pengujian, dan pemeliharaan. Pemilihan metode *Waterfall* dilakukan karena tahapan pengembangan sistem pada penelitian ini bersifat terstruktur dan memiliki kebutuhan sistem yang telah ditentukan sejak awal.

### Analisis Kebutuhan

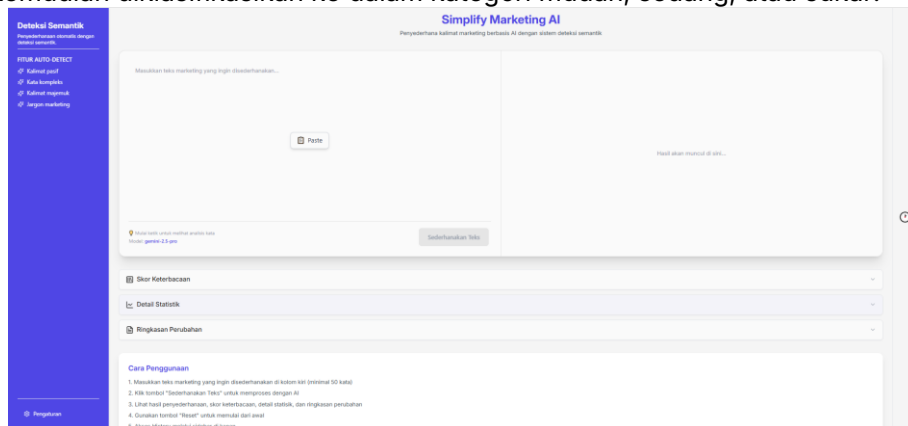
Tahap analisis kebutuhan dilakukan dengan mengidentifikasi permasalahan utama pada kalimat marketing yang cenderung panjang, kompleks, dan mengandung istilah teknis sehingga menurunkan tingkat keterbacaan informasi promosi. Berdasarkan hasil analisis, sistem dirancang untuk mampu mendeteksi kalimat pasif, kalimat majemuk, kata kompleks, serta jargon marketing secara otomatis. Selain itu, sistem juga dirancang agar mampu memberikan hasil penyederhanaan teks yang tetap mempertahankan makna utama dari kalimat asli.

### Tahap Perencanaan

Pada tahap perancangan, aplikasi dibangun menggunakan *ReactJS* pada sisi *frontend* dan *FastAPI* pada sisi *backend*. Arsitektur sistem dirancang agar proses komunikasi antara antarmuka pengguna dan layanan *Gemini API* dapat berjalan secara efisien melalui mekanisme *Application Programming Interface (API)*. Struktur navigasi sistem dibuat sederhana sehingga pengguna dapat melakukan proses penyederhanaan kalimat, melihat riwayat penggunaan, serta mengatur preferensi aplikasi dengan lebih mudah.

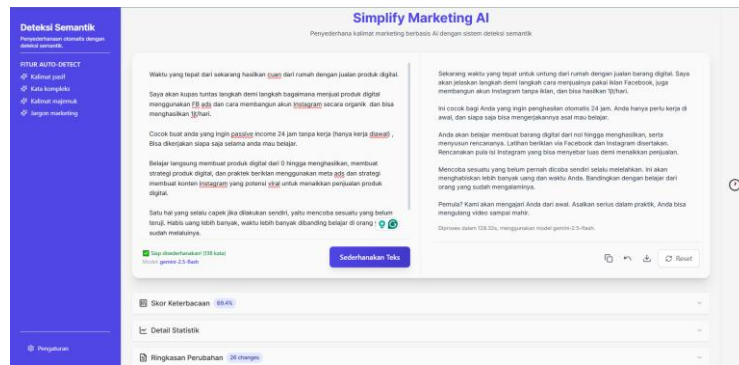
### Hasil Implementasi

Tahap implementasi menghasilkan aplikasi yang mampu memproses input kalimat marketing dan menampilkan hasil penyederhanaan dalam bentuk teks yang lebih sederhana dan mudah dipahami. Sistem juga menampilkan skor keterbacaan berdasarkan indikator panjang kalimat, jumlah kata kompleks, penggunaan kalimat pasif, dan kalimat majemuk. Hasil analisis tersebut kemudian diklasifikasikan ke dalam kategori mudah, sedang, atau sukar.



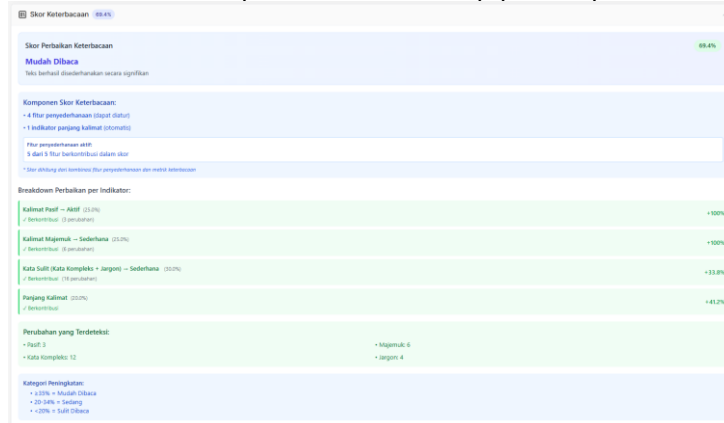
Gambar 4. Tampilan Halaman Utama

Halaman utama aplikasi dirancang sebagai pusat interaksi pengguna dalam melakukan penyederhanaan teks *marketing*. Antarmuka dibagi menjadi dua bagian utama, yaitu kolom input di sisi kiri dan kolom output di sisi kanan. Kolom input digunakan untuk memasukkan teks promosi yang akan diproses, baik melalui pengetikan langsung maupun *paste* teks, dengan ketentuan minimal 50 kata agar proses penyederhanaan dapat dijalankan secara optimal. Setelah proses dilakukan, hasil penyederhanaan akan ditampilkan pada kolom output secara terstruktur dan mudah dibaca. Di bagian bawah halaman terdapat tiga panel informasi yang dapat diperluas, yaitu Skor Keterbacaan, Detail Statistik, dan Ringkasan Perubahan, yang berfungsi memberikan analisis tambahan terhadap hasil penyederhanaan. Selain itu, sisi kiri halaman memuat menu *auto detection* untuk memilih jenis transformasi teks yang diaktifkan, serta tombol pengaturan untuk mengakses konfigurasi lanjutan sistem.



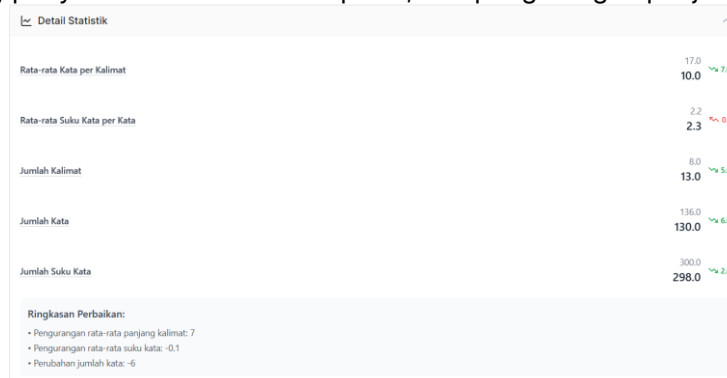
Gambar 5. Tampilan Halaman Setelah Penyederhanaan

Sistem menampilkan hasil penyederhanaan teks *marketing* berdasarkan konfigurasi *auto detection* yang dipilih pengguna dengan menggunakan data promosi nyata sebagai bahan pengujian agar hasil evaluasi lebih representatif terhadap praktik pemasaran digital.



Gambar 6. Panel Skor Keterbacaan

Panel skor keterbacaan menampilkan tingkat peningkatan keterbacaan teks sebesar 69,4% dengan kategori “Mudah Dibaca” berdasarkan transformasi kalimat pasif, pemecahan kalimat majemuk, penyederhanaan kata kompleks, dan pengurangan panjang kalimat.



Gambar 7. Panel Detail Statistik

Panel Detail Statistik menyajikan analisis kuantitatif terhadap perubahan karakteristik teks sebelum dan sesudah proses penyederhanaan. Informasi yang ditampilkan meliputi jumlah kata, jumlah kalimat, jumlah suku kata, rata-rata kata per kalimat, dan rata-rata suku kata per kata. Hasil pengujian menunjukkan bahwa jumlah kata berkurang dari 136 menjadi 130 kata, yang menandakan teks menjadi lebih ringkas tanpa menghilangkan makna utama. Jumlah kalimat meningkat dari 8 menjadi 13 kalimat akibat proses pemecahan kalimat majemuk menjadi struktur yang lebih sederhana. Rata-rata kata per kalimat juga menurun dari 17 menjadi 10 kata, sehingga tingkat keterbacaan teks meningkat dan isi promosi menjadi lebih mudah dipahami. Sementara itu, perubahan rata-rata suku kata per kata relatif kecil, yaitu dari 2,2 menjadi 2,3, yang menunjukkan bahwa penyederhanaan lebih berfokus pada struktur kalimat dibanding

perubahan kosakata secara ekstrem.

### Hasil Uji Coba

Pengujian sistem dilakukan menggunakan metode *Black Box Testing* untuk memastikan seluruh fungsi aplikasi berjalan sesuai kebutuhan. Pengujian dilakukan pada 15 skenario yang mencakup proses input teks, penyederhanaan kalimat, penghitungan skor keterbacaan, penyimpanan riwayat, serta pengaturan sistem.

**Tabel 1. Hasil Uji Coba *BlackBox Testing***

No	Skenario Uji	Input	Proses yang Diharapkan	Output yang Diharapkan	Hasil
1	Akses Halaman Utama	Tautan aplikasi: <a href="https://simplify-ai-six.vercel.app/">https://simplify-ai-six.vercel.app/</a>	Sistem menampilkan halaman utama	Halaman utama muncul sesuai dengan konten aplikasi	Berhasil
2	Mengakses Menu Pengaturan	Menekan ikon pengaturan	Sistem membuka menu pengaturan	Menu pengaturan tampil di layar	Berhasil
3	Mengaktifkan dan Menonaktifkan Opsi <i>Auto Detection</i>	Mengubah status <i>checkbox Auto Detection</i>	Sistem memperbarui pengaturan dan memastikan minimal satu opsi tetap aktif	Pengaturan tersimpan dengan minimal satu opsi aktif	Berhasil
4	Validasi Input Teks Marketing Kurang dari 50 Kata	Memasukkan teks kurang dari 50 kata	Sistem melakukan validasi dan menonaktifkan tombol "Sederhanakan"	Pesan validasi muncul dan tombol tetap nonaktif	Berhasil
5	Input Teks Marketing Minimal 50 Kata	Memasukkan teks dengan panjang $\geq$ 50 kata	Sistem menerima input dan mengaktifkan tombol "Sederhanakan"	Tombol "Sederhanakan" aktif dan siap digunakan	Berhasil
6	Proses Penyederhanaan Teks	Menekan tombol "Sederhanakan" setelah memasukkan teks valid	Sistem memproses teks sesuai pengaturan dan mengirimkan hasil	Proses penyederhanaan berjalan hingga selesai	Berhasil
7	Menampilkan Hasil Penyederhanaan pada Kolom Output	Teks telah diproses	Sistem menampilkan hasil penyederhanaan	Hasil penyederhanaan tampil pada kolom output	Berhasil
8	Membuka Panel Skor Keterbacaan	Menekan panel Skor Keterbacaan	Sistem menampilkan nilai keterbacaan sebelum dan sesudah penyederhanaan	Data skor keterbacaan muncul sesuai perhitungan	Berhasil
9	Membuka Panel Detail Statistik	Menekan panel Detail Statistik	Sistem menampilkan detail statistik teks sebelum dan sesudah	Data statistik tampil sesuai perhitungan	Berhasil

No	Skenario Uji	Input	Proses yang Diharapkan	Output yang Diharapkan	Hasil
10	Membuka Panel Ringkasan Perubahan	Menekan panel Ringkasan Perubahan	Sistem menampilkan ringkasan perubahan hasil penyederhanaan	Ringkasan perubahan muncul pada panel	Berhasil
11	Menampilkan Jumlah dan Jenis Transformasi serta <i>Highlight</i> Perubahan	Teks telah diproses	Sistem menampilkan jumlah, jenis, dan <i>highlight</i> perubahan	Informasi transformasi tampil pada panel ringkasan	Berhasil
12	Menggunakan Fitur Salin, Unduh, dan <i>Reset</i>	Menekan tombol sesuai fitur	Sistem menjalankan fungsi sesuai perintah pengguna	Hasil berhasil disalin, diunduh, dikembalikan, atau dihapus	Berhasil
13	Mengakses Menu Riwayat	Menekan ikon riwayat	Sistem menampilkan daftar riwayat penyederhanaan	Panel riwayat tampil di sisi kanan layar	Berhasil
14	Menyimpan dan Memuat Riwayat Penyederhanaan	Menyederhanakan teks dan memilih data riwayat	Sistem menyimpan hasil ke riwayat dan memuat ulang data yang dipilih	Hasil sebelumnya dapat dimuat kembali	Berhasil
15	Menghapus Seluruh Riwayat Penyederhanaan	Menekan tombol "Hapus Semua Riwayat"	Sistem menghapus seluruh data riwayat	Riwayat kosong setelah dihapus	Berhasil

Hasil pengujian menunjukkan bahwa seluruh fitur sistem berjalan sesuai kebutuhan. Pengujian *Black Box Testing* yang dilakukan pada berbagai skenario menunjukkan bahwa fungsi input teks, proses penyederhanaan, pengaturan *auto detection*, riwayat, dan tampilan hasil dapat beroperasi dengan baik tanpa ditemukan kesalahan fungsional. Selain pengujian fungsional, evaluasi kegunaan sistem dilakukan menggunakan metode *System Usability Scale* (SUS) terhadap 50 responden. Hasil evaluasi memperoleh skor rata-rata sebesar 90,85 yang termasuk kategori *Excellent*, *Grade A*, dan *Acceptable*. Nilai tersebut menunjukkan bahwa sistem memiliki tingkat kemudahan penggunaan yang sangat baik serta mampu memberikan pengalaman pengguna yang positif.

Selain pengujian fungsional, evaluasi kegunaan sistem dilakukan menggunakan metode *System Usability Scale* (SUS) terhadap 50 responden. Hasil evaluasi memperoleh skor rata-rata sebesar 90,85 yang termasuk kategori *Excellent*, *Grade A*, dan *Acceptable*. Nilai tersebut menunjukkan bahwa sistem memiliki tingkat kemudahan penggunaan yang sangat baik serta mampu memberikan pengalaman pengguna yang positif.

**Tabel 2. Hasil Uji *System Usability Scale* (SUS)**

No	Responden	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10
1	Alif Ainul	4	1	5	1	5	1	4	1	5	2
2	Jonathan	5	1	5	2	5	2	4	1	4	2
3	Rafli	5	1	5	1	5	1	5	1	5	1
4	Rilo Prianoko	5	1	5	1	5	1	5	1	5	1
5	Wardiansyah Fauzi Abdillah	5	1	5	1	5	1	5	1	5	1
6	Aprilia Anggraini	5	1	5	5	5	1	5	1	5	3
7	Rasya Rachman Hakim	5	1	5	1	5	1	5	1	5	1
8	Alvina	5	1	5	1	5	1	5	1	5	1

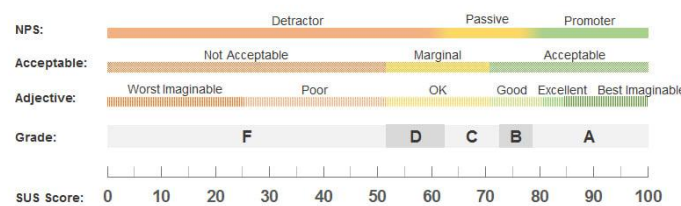
No	Responden	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10
9	Ananda	4	1	5	1	5	1	5	1	4	1
10	Ferdian	5	2	5	1	5	1	5	1	5	1
...	...	...	...	...	...	...	...	...	...	...	...
50	Irsyad	4	2	4	3	5	2	4	2	4	3

Pada metode *System Usability Scale* (SUS), setiap jawaban responden terlebih dahulu dikonversi menjadi skor usability. Pernyataan positif dihitung dengan mengurangi skor jawaban dengan nilai 1, sedangkan pernyataan negatif dihitung menggunakan rumus 5 dikurangi skor jawaban. Seluruh skor hasil konversi kemudian dijumlahkan dan dikalikan dengan faktor 2,5 untuk menghasilkan nilai SUS dalam rentang 0–100. Nilai tersebut digunakan sebagai indikator tingkat usability website secara keseluruhan.

**Tabel 3. Hasil Pengolahan Data SUS**

Skor Asli										Jumlah	Nilai (Jumlah x 2.5)
Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10		
4	1	5	1	5	1	4	1	5	2	37	92,5
5	1	5	2	5	2	4	1	4	2	35	87,5
5	1	5	1	5	1	5	1	5	1	40	100
5	1	5	1	5	1	5	1	5	1	40	100
5	1	5	1	5	1	5	1	5	1	40	100
5	1	5	5	5	1	5	1	5	3	34	85
5	1	5	1	5	1	5	1	5	1	40	100
5	1	5	1	5	1	5	1	5	1	40	100
4	1	5	1	5	1	5	1	4	1	38	95
5	2	5	1	5	1	5	1	5	1	39	97,5
..	..	..	..	..	..	..	..	..	..	..	..
..	..	..	..	..	..	..	..	..	..	..	..
4	2	4	3	5	2	4	2	4	3	29	72,5
<b>Skor Rata-rata (Hasil Akhir)</b>											<b>90,85</b>

Hasil pengolahan kuesioner terhadap 50 responden menunjukkan rata-rata skor SUS sebesar 90,85. Nilai ini kemudian diinterpretasikan menggunakan beberapa kategori penilaian SUS, meliputi *Percentile*, *Grade*, *Adjective*, *Acceptability*, dan *Net Promoter Score* (NPS), untuk mengevaluasi kualitas usability sistem secara lebih komprehensif. Posisi nilai rata-rata skor SUS digambarkan menggunakan garis putus-putus yang dapat dilihat pada Gambar 12.



**Gambar 8. Hasil Skor Interpretasi**

Berdasarkan interpretasi SUS, skor 90,85 berada di atas rata-rata global SUS sebesar 68 sehingga termasuk kategori di atas rata-rata pada aspek *Percentile*. Pada kategori *Grade*, nilai tersebut masuk ke peringkat A karena melampaui ambang batas minimal 80. Dari aspek *Adjective*, skor tersebut termasuk kategori *Excellent*, sedangkan pada aspek *Acceptability* tergolong *Acceptable*. Selain itu, berdasarkan kategori *Net Promoter Score* (NPS), skor tersebut termasuk *Promoter*, yang menunjukkan bahwa mayoritas pengguna memiliki kecenderungan tinggi untuk merekomendasikan website kepada pengguna lain. Hasil ini menunjukkan bahwa website memiliki tingkat usability yang sangat baik serta mampu memberikan pengalaman penggunaan yang positif.

Berdasarkan hasil implementasi dan pengujian, aplikasi yang dikembangkan mampu membantu pengguna menghasilkan teks promosi yang lebih sederhana, jelas, dan mudah dipahami. Integrasi Gemini API terbukti efektif dalam mendukung proses penyederhanaan kalimat secara

otomatis, sementara fitur evaluasi keterbacaan memberikan informasi tambahan yang membantu pengguna memahami kualitas hasil penyederhanaan teks.

### KESIMPULAN

Berdasarkan hasil penelitian dan pengujian yang telah dilakukan, dapat disimpulkan bahwa aplikasi penyederhanaan kalimat *marketing* berbasis web berhasil dikembangkan dengan memanfaatkan Gemini API menggunakan ReactJS pada sisi *frontend* dan FastAPI pada sisi *backend*. Sistem mampu melakukan penyederhanaan kalimat secara otomatis melalui deteksi terhadap kalimat pasif, kata kompleks, kalimat majemuk, dan jargon *marketing*, sehingga menghasilkan teks yang lebih mudah dipahami tanpa menghilangkan makna utama. Aplikasi juga berhasil menampilkan skor keterbacaan berdasarkan indikator yang telah ditentukan, sehingga pengguna dapat mengetahui tingkat perbaikan antara teks asli dan hasil penyederhanaan. Pendekatan ini membantu pengguna dalam menghasilkan konten promosi yang lebih jelas, efektif, dan komunikatif. Hasil pengujian *Black Box Testing* menunjukkan bahwa seluruh fungsi sistem berjalan sesuai kebutuhan yang telah dirancang. Selain itu, pengujian *System Usability Scale* (SUS) yang melibatkan 50 responden memperoleh skor rata-rata 90,85 dengan kategori *Excellent*, *Grade A*, dan *Acceptable*. Hasil tersebut menunjukkan bahwa aplikasi memiliki tingkat kegunaan yang sangat baik serta mampu memberikan pengalaman penggunaan yang positif bagi pengguna. Berdasarkan hasil penelitian ini, disarankan agar pengembangan selanjutnya menggunakan metode pengukuran keterbacaan yang lebih komprehensif dan sesuai dengan karakteristik bahasa Indonesia, meningkatkan fleksibilitas sistem sehingga mampu menyederhanakan teks dengan jumlah kata di bawah 50, serta memanfaatkan model bahasa yang lebih spesifik disertai integrasi database guna menyimpan dan meningkatkan kualitas hasil penyederhanaan teks secara berkelanjutan.

### REFERENSI

- Cahya Kamilla, A., Priyani, N., Handrianus Pranatawijaya, V., Noor Kamala Sari, N., Yos Sudarso, J., Jekan Raya, K., Palangka Raya, K., & Tengah, K. (2024). Pengembangan Aplikasi Kasir Mobile Yang Efisien: *JATI (Jurnal Mahasiswa Teknik Informatika)*, 8(4), 5966–5971. <https://doi.org/10.36040/JATI.V8I4.9829>
- Hariyadi, H., Irawan, J., Pranatawijaya, V. H., & Sari, N. N. K. (2024). Implementasi Gemini API dalam Generatif Teks Deskripsi Karya pada Pengembangan Aplikasi Pameran Karya Berbasis Web. *Jurnal Teknologi Informasi Dan Terapan*, 11(1), 21–26. <https://doi.org/10.25047/JTIT.V11I1.356>
- Homepage, J., Yani, R., Nazhifah, I., & Pradika, M. I. (2025). System Usability Scale in Information System Application Development Using Systematic Mapping Study. *IJATIS: Indonesian Journal of Applied Technology and Innovation Science*, 2(2), 130–140. <https://doi.org/10.57152/IJATIS.V2I2.2275>
- Khairunnisa, C. M. (2022). Pemasaran Digital sebagai Strategi Pemasaran: Conceptual Paper. *JAMIN: Jurnal Aplikasi Manajemen Dan Inovasi Bisnis*, 5(1), 98–102. <https://doi.org/10.47201/JAMIN.V5I1.109>
- Kumar, P. (2024). Large language models (LLMs): survey, technical frameworks, and future challenges. *Artificial Intelligence Review 2024* 57:10, 57(10), 260–. <https://doi.org/10.1007/S10462-024-10888-Y>
- Kusuma, I. D., Yuhana, U. L., & Informatika, T. (2024). 1, 2, 3. 10(1), 122–127.
- Le, N. Q. K., & Chua, M. C. H. (2026). Natural language processing. *Methods*, 245(November), 53–54. <https://doi.org/10.1016/j.ymeth.2025.11.004>
- Mabotha, E., Mabunda, N. E., & Ali, A. (2024). Performance Evaluation of a Dynamic RESTful API Using FastAPI, Docker and Nginx. *IEEE Global Energy Conference 2024, GEC 2024*, 174–181. <https://doi.org/10.1109/GEC61857.2024.10881712>
- Maylawati, D. S. Adillah, Kumar, Y. J., Kasmin, F. B., & Ramdhani, M. A. (2024). Readability Evaluation Metrics for Indonesian Automatic Text Summarization: A Systematic Review. *Journal of Engineering Science and Technology Review*, 17(5), 199–210. <https://doi.org/10.25103/jestr.175.25>
- North, K., Ranasinghe, T., Shardlow, M., & Zampieri, M. (2024). Deep learning approaches to lexical simplification: A survey. *Journal of Intelligent Information Systems 2024* 63:1, 63(1), 111–134. <https://doi.org/10.1007/S10844-024-00882-9>
- Phatak, A., Savage, D. W., Ohle, R., Smith, J., & Mago, V. (2022). Medical Text Simplification

- Using Reinforcement Learning (TESLEA): Deep Learning-Based Text Simplification Approach. *JMIR Medical Informatics*, 10(11), e38095. <https://doi.org/10.2196/38095>
- Saputra, S. D., Syahrohim, I., Handrianus Pranatawijaya, V., Noor, N., & Sari, K. (2024). Implementasi Api ChatGPT Summarizer Berbasis Website. *Jurnal Informatika Dan Teknik Elektro Terapan*, 12(3), 2830–7062. <https://doi.org/10.23960/JITET.V12I3.4313>
- Sitio, H. J. S., Christovita, I., Ahmad, R. K., & Setiawan, Y. (2023). Web-Based Application Development for Training Data Management Using REACTJS. *Indonesian Journal of Multidisciplinary Science*, 2(6), 2573–2588. <https://doi.org/10.55324/IJOMS.V2I6.461>
- Sun, T. X., Liu, X. Y., Qiu, X. P., & Huang, X. J. (2022). Paradigm Shift in Natural Language Processing. *Machine Intelligence Research 2022 19:3*, 19(3), 169–183. <https://doi.org/10.1007/S11633-022-1331-6>