

PEMBUATAN APLIKASI BERBASIS API GPT UNTUK PENYESUAIAN TATA BAHASA DAN GAYA PENULISAN AKADEMIK SKRIPSI MAHASISWA FTI UNIBBA

Asep Ridwan Supriatna¹, Mohammad Bayu Anggara²

^{1,2} Program Studi Teknik Informatika

Fakultas Teknologi Informasi, Universitas Bale Bandung

Email : asepridwns@gmail.com

ABSTRAK: Perkembangan teknologi *Artificial Intelligence* (AI) telah membawa dampak besar dalam berbagai sektor, termasuk pendidikan tinggi. Salah satu teknologi yang menonjol adalah model *Natural Language Processing* (NLP) seperti ChatGPT, Gemini, dan DeepSeek. Masalah yang dihadapi oleh mahasiswa FTI UNIBBA adalah kesulitan menyesuaikan tata bahasa dan gaya penulisan akademik skripsi sesuai pedoman. Penelitian ini bertujuan untuk merancang dan membangun aplikasi berbasis *Application Programming Interface* (API) kecerdasan buatan yang membantu mahasiswa dalam menyesuaikan tata bahasa dan gaya penulisan akademik skripsi secara otomatis dan efisien. Aplikasi ini mengintegrasikan tiga model AI, yaitu ChatGPT, Gemini, dan DeepSeek ke dalam satu sistem, sehingga pengguna dapat memilih model yang paling sesuai. Metode pengembangan yang digunakan adalah *SDLC Waterfall*, melalui tahapan analisis, desain, pengkodean, pengujian, dan pemeliharaan. Teknologi yang diterapkan mencakup Next.js berbasis TypeScript untuk membangun antarmuka web interaktif, serta Firebase untuk autentikasi dan pengelolaan basis data. Interaksi dilakukan melalui antarmuka chat, tempat mahasiswa memasukkan teks skripsi, memilih subbab, dan memilih model AI untuk menyesuaikan teks sesuai pedoman penulisan FTI UNIBBA. Sistem ini dirancang untuk meningkatkan efisiensi, konsistensi, dan kualitas penulisan skripsi mahasiswa. Hasil dari penelitian ini adalah aplikasi bernama PORSI (Penyesuaian Format Skripsi), yang dirancang untuk membantu mahasiswa dalam menyesuaikan tata bahasa dan gaya penulisan akademik skripsi secara otomatis. Berdasarkan hasil pengujian, ChatGPT terbukti sebagai model paling unggul dengan skor koreksi tertinggi dan waktu respons tercepat, sehingga dipilih sebagai model default dalam sistem PORSI. Aplikasi mampu meningkatkan efisiensi penulisan, mengurangi kesalahan tata bahasa, serta berkontribusi dalam penerapan AI untuk mendukung kualitas pendidikan tinggi di lingkungan FTI UNIBBA.

Kata kunci: AI, API, ChatGPT, Penulisan Skripsi, Tata Bahasa Akademik.

ABSTRACT: The development of *Artificial Intelligence* (AI) has brought significant impacts across various sectors, including higher education. One of the prominent technologies is *Natural Language Processing* (NLP) models such as ChatGPT, Gemini, and DeepSeek. Students at FTI UNIBBA often face difficulties in adjusting the grammar and academic writing style of their theses to comply with the required guidelines. This research aims to design and develop an *Artificial Intelligence* (AI)-based *Application Programming Interface* (API) application that assists students in automatically and efficiently adjusting the grammar and academic writing style of their theses. The

application integrates three AI models ChatGPT, Gemini, and DeepSeek into a single system, allowing users to select the most suitable model. The development method employed is the SDLC Waterfall, consisting of analysis, design, coding, testing, and maintenance stages. The technologies applied include Next.js with TypeScript to build an interactive web interface and Firebase for authentication and database management. Interaction is carried out through a chat interface where students input thesis text, select subchapters, and choose the AI model to adjust the text according to FTI UNIBBA writing guidelines. The system is designed to improve efficiency, consistency, and the overall quality of students' academic writing. The result of this research is an application called PORSI (Thesis Format Adjustment), designed to help students adjust grammar and academic writing style in their theses automatically. Based on the evaluation results, ChatGPT proved to be the most superior model, achieving the highest correction score and the fastest response time, thus selected as the default model in PORSI. The application enhances writing efficiency, reduces grammatical errors, and contributes to the implementation of AI in supporting the quality of higher education at FTI UNIBBA.

Keywords: AI, API, ChatGPT, Academic Writing Style, Thesis Writing.

PENDAHULUAN

Perkembangan teknologi *Artificial Intelligence* (AI) telah membawa transformasi signifikan dalam berbagai bidang, termasuk pendidikan. Salah satu inovasi terbaru dalam bidang AI adalah teknologi ChatGPT, yaitu model kecerdasan buatan (*Artificial Intelligence*) yang dikembangkan oleh OpenAI dan berfokus pada pemrosesan bahasa alami (*Natural Language Processing*) seperti terjemahan otomatis, analisis sentimen, serta identifikasi entitas. Seiring kemajuan teknologinya, ChatGPT kini digunakan dalam berbagai aplikasi, termasuk chatbot interaktif, sistem penulisan otomatis, hingga pembuatan konten visual (Rachbini et al., 2023). Pemanfaatan teknologi berbasis API kecerdasan buatan seperti ChatGPT, DeepSeek, dan Google Gemini telah membuka peluang baru dalam meningkatkan efisiensi dan kualitas penulisan dokumen akademik, termasuk skripsi, sehingga integrasinya ke dalam aplikasi khusus untuk menyesuaikan tata bahasa dan gaya penulisan akademik menjadi semakin relevan di era digital. Fakultas Teknologi Informasi (FTI) Universitas Bale Bandung (UNIBBA), yang berdiri sejak tahun 2008 di bawah naungan Yayasan Pendidikan Bale Bandung (YPBB), memiliki dua program studi yaitu Teknik Informatika dan Sistem Informasi, serta telah menyediakan buku pedoman penulisan skripsi untuk mendukung kegiatan akademik. Namun, masih banyak mahasiswa yang kesulitan dalam menyesuaikan tata bahasa dan gaya penulisan akademik sesuai standar yang

ditetapkan, menunjukkan bahwa keberadaan buku pedoman saja belum cukup sehingga dibutuhkan solusi berbasis teknologi yang mampu membantu mahasiswa secara otomatis dan efisien. Penulisan skripsi sendiri menjadi tantangan karena memerlukan pemahaman mendalam mengenai tata bahasa, istilah akademik, dan konsistensi penerapan kaidah ilmiah, sedangkan belum tersedia sistem yang dapat melakukan koreksi otomatis di lingkungan FTI UNIBBA. Akibatnya, mahasiswa sering kali melakukan revisi berulang sebelum naskah skripsinya memenuhi standar akademik yang ditetapkan. Oleh karena itu, diperlukan aplikasi berbasis AI yang dapat mengoreksi kalimat tidak formal, memperbaiki struktur kurang logis, serta memastikan konsistensi penulisan akademik guna mengurangi beban kerja mahasiswa dan dosen serta meningkatkan kualitas penulisan. Beberapa penelitian terdahulu telah membuktikan potensi API kecerdasan buatan dalam pengolahan teks, seperti Aryabimo et al. (2024) yang menunjukkan peningkatan interaksi pelanggan melalui API ChatGPT pada website restoran, Saputra et al. (2024) yang memanfaatkan API ChatGPT untuk merangkum teks panjang secara efisien, dan Hariyadi et al. (2024) yang menggunakan API Gemini untuk menghasilkan deskripsi karya otomatis pada aplikasi pameran. Namun, belum ada penelitian yang secara khusus mengembangkan aplikasi penyesuaian tata bahasa dan gaya penulisan akademik skripsi.

Penelitian ini menawarkan kebaruan dengan mengintegrasikan tiga API kecerdasan buatan sekaligus, yaitu ChatGPT, Gemini, dan DeepSeek, dalam pengembangan aplikasi berbasis metode *Waterfall* yang terdiri dari tahapan analisis, desain, pengkodean, pengujian, dan pemeliharaan. Framework Next.js berbasis TypeScript dipilih karena kemampuannya dalam membangun aplikasi full-stack berperforma tinggi melalui server-side rendering serta integrasi API yang efisien. Dengan demikian, penelitian ini diharapkan dapat meningkatkan efisiensi dan kualitas penulisan skripsi mahasiswa, serta mengurangi waktu yang dibutuhkan dalam proses revisi.

METODA

Penelitian ini menggunakan metode pengembangan *Waterfall* yang terdiri dari tahapan pengumpulan data, analisis, desain, pengkodean, dan pengujian. Metode ini dipilih karena bersifat sistematis dan berurutan sehingga setiap tahap menjadi dasar bagi tahap berikutnya.

1. Identifikasi Masalah

Menghimpun informasi awal mengenai kendala penyesuaian tata bahasa, gaya akademik, struktur dokumen, tata letak, serta sitasi melalui observasi proses pengecekan skripsi dan wawancara dengan pimpinan fakultas serta mahasiswa tingkat akhir.

2. Pengumpulan Data

Tahapan ini dilakukan untuk memperoleh informasi dan kebutuhan sistem yang akan dibangun. Data dikumpulkan melalui observasi langsung terhadap proses pengecekan tata bahasa dan gaya penulisan akademik skripsi di FTI UNIBBA, wawancara dengan pimpinan fakultas dan mahasiswa guna mengetahui kendala serta fitur yang diharapkan, serta studi pustaka dari jurnal, buku, dan artikel ilmiah terkait penerapan teknologi kecerdasan buatan, pemanfaatan API dalam pemrosesan teks, serta metode pengembangan perangkat lunak *Waterfall*.

3. Analisis

Pada tahap ini dilakukan analisis menyeluruh terhadap kebutuhan perangkat lunak, meliputi analisis fungsional dan nonfungsional. Analisis fungsional mencakup identifikasi fitur utama

seperti koreksi otomatis tata bahasa, penyesuaian gaya penulisan sesuai pedoman akademik, pemilihan model AI, serta penyimpanan hasil koreksi. Sementara itu, analisis nonfungsional mencakup aspek performa, keamanan, dan kemudahan penggunaan. Hasil analisis ini menjadi dasar dalam merancang arsitektur sistem dan antarmuka pengguna.

4. Desain

Tahap desain dilakukan dengan menggunakan pendekatan *Unified Modeling Language* (UML) yang terdiri dari *Use Case Diagram*, *Activity Diagram*, dan *Sequence Diagram* untuk memodelkan interaksi antara pengguna dan sistem. Desain basis data menggunakan pendekatan NoSQL berbasis Firebase Firestore, sedangkan desain antarmuka pengguna dibuat menggunakan Figma dengan memperhatikan kemudahan navigasi dan konsistensi tampilan. Desain ini bertujuan untuk menggambarkan alur kerja sistem serta hubungan antar komponen secara logis dan terstruktur.

5. Pengkodean

Tahap pengkodean merupakan implementasi dari desain sistem ke dalam bentuk aplikasi berbasis web menggunakan framework Next.js dan bahasa pemrograman TypeScript. Sistem ini terintegrasi dengan tiga API kecerdasan buatan, yaitu ChatGPT, Gemini, dan DeepSeek, untuk memberikan hasil koreksi yang bervariasi dan komprehensif. Selain itu, layanan Firebase digunakan untuk autentikasi pengguna, penyimpanan data, dan pengelolaan basis data secara real-time.

6. Pengujian

Tahap pengujian dilakukan menggunakan metode Black Box Testing untuk memastikan seluruh fitur berjalan sesuai dengan kebutuhan yang telah didefinisikan. Pengujian meliputi fungsi login, input dan koreksi teks, tampilan hasil koreksi, serta pengelolaan data pengguna dan pedoman skripsi. Selain itu, dilakukan pengujian komparatif terhadap ketiga model AI (GPT-4, Gemini, dan DeepSeek) dengan input yang sama untuk menilai kualitas dan konsistensi hasil koreksi yang dihasilkan oleh masing-masing model.

HASIL DAN PEMBAHASAN

1. Analisis

Analisis dilakukan sebagai langkah awal dalam penelitian untuk mengidentifikasi kebutuhan-kebutuhan yang diperlukan dalam perancangan aplikasi

a. Analisis Software

Tabel 1 Kebutuhan Software Minimum

Sistem Operasi	Windows 10
Code Editor	Visual Studio Code
Web Framework	Next.js
Database	Firebase
Perancangan User Interface	Figma
Perancangan UML	Draw.io
Web Browser	Microsoft Edge

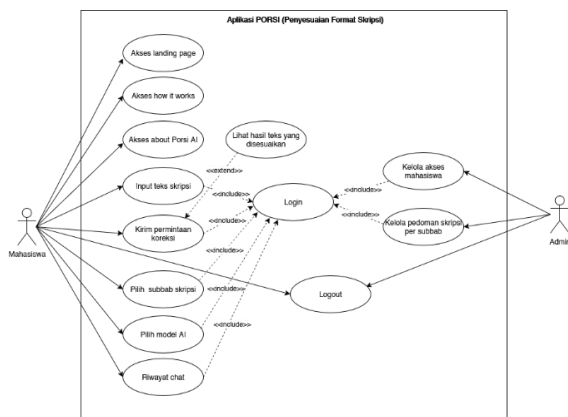
b. Analisis Pengguna

Mahasiswa dapat menginput teks skripsi, memilih bab atau subbab yang ingin disesuaikan, memilih model API kecerdasan buatan (ChatGPT, Gemini, atau DeepSeek), serta melihat hasil penyesuaian yang dihasilkan oleh sistem secara otomatis. Sementara itu, admin fakultas memiliki peran dalam pengelolaan sistem, meliputi pengaturan akses pengguna, pembaruan pedoman penulisan skripsi per subbab, serta pengelolaan data referensi yang digunakan sebagai acuan sistem.

2. Desain

A. Use Case Diagram

Pada rancangan *Use Case Diagram* bertujuan untuk menggambarkan interaksi antara aktor dengan aplikasi penyesuaian tata bahasa dan gaya penulisan akademik skripsi berbasis web. Diagram ini memperlihatkan hubungan antara mahasiswa dan admin dengan fitur-fitur utama.



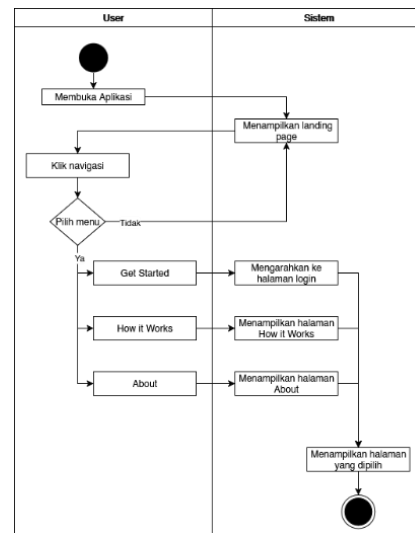
Gambar 1 Use Case Diagram

B. Activity Diagram

Berikut adalah *Activity Diagram* pada aplikasi penyesuaian tata bahasa dan gaya penulisan akademik skripsi berbasis API GPT:

b. Activity Diagram akses landing page

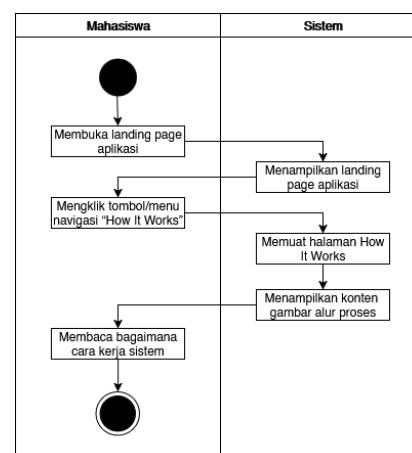
Setelah aplikasi terbuka, sistem secara otomatis menampilkan halaman landing page yang berisi informasi umum mengenai aplikasi. Selanjutnya, pengguna dapat melakukan interaksi awal berupa memilih salah satu menu navigasi yang tersedia.



Gambar 2 Activity Diagram akses landing page

a. Activity Diagram akses how it works

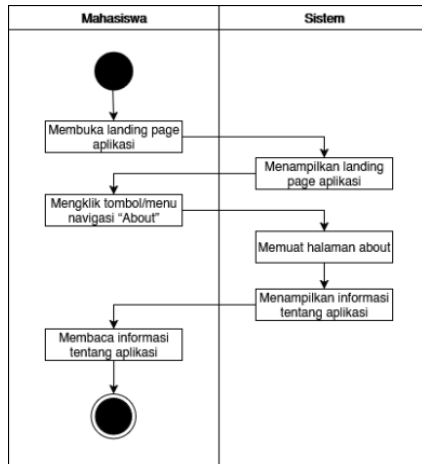
Setelah aplikasi berhasil dimuat, sistem menampilkan halaman utama atau landing page yang berisi informasi umum terkait aplikasi. Selanjutnya, pengguna melakukan interaksi dengan memilih menu navigasi yang tersedia, dalam hal ini menu “how it works”.



Gambar 3 Activity Diagram akses how it works

b. *Activity Diagram* akses about

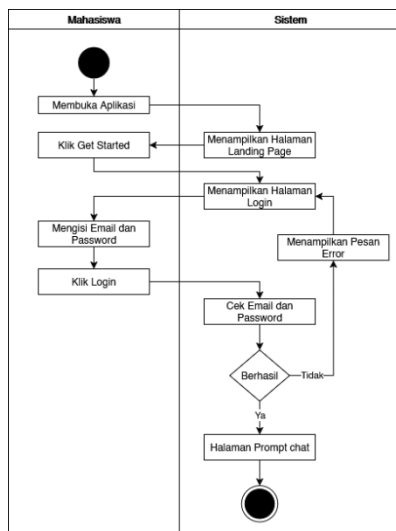
Setelah halaman landing page berhasil ditampilkan oleh sistem, pengguna dapat memilih menu navigasi yang tersedia. Salah satu pilihan navigasi adalah menu “about” yang biasanya berada di bagian atas atau samping halaman.



Gambar 4 Activity Diagram akses about

c. *Activity Diagram* login mahasiswa

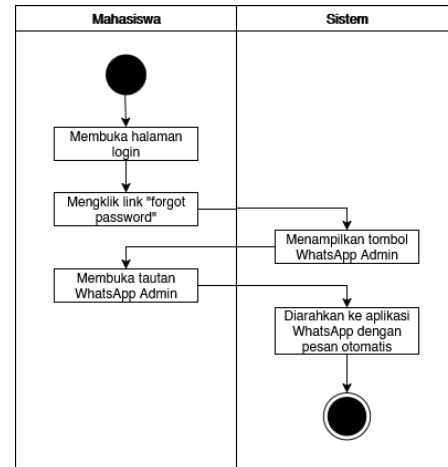
Mahasiswa sebagai pengguna akan membuka aplikasi penyesuaian tata bahasa dan gaya penulisan akademik skripsi dan mengakses halaman login. Selanjutnya, mahasiswa mengisi email dan password, lalu menekan tombol login. Sistem akan memproses dan melakukan validasi terhadap data login.



Gambar 5 Activity Diagram login mahasiswa

d. *Activity Diagram* forgot password

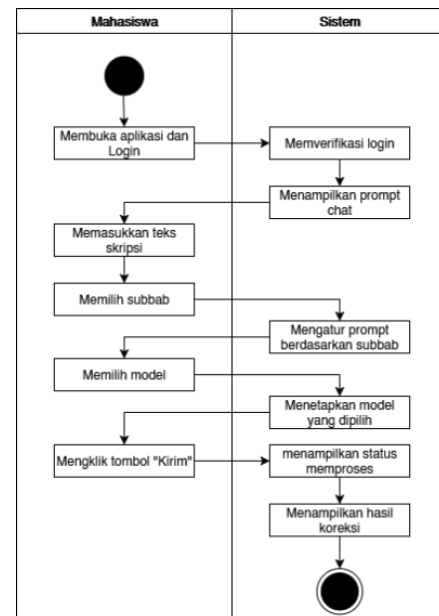
Proses dimulai saat mahasiswa membuka halaman login aplikasi. Jika mahasiswa tidak dapat mengingat kata sandinya, ia dapat mengklik link “forgot password” yang tersedia.



Gambar 6 Activity Diagram forgot password

e. *Activity Diagram* koreksi teks

Tahapan yang dilakukan oleh mahasiswa mengirimkan permintaan ke sistem. Selanjutnya, sistem akan memproses permintaan tersebut dengan menyesuaikan teks berdasarkan pedoman yang telah ditentukan dan menampilkan hasil koreksi kepada mahasiswa.

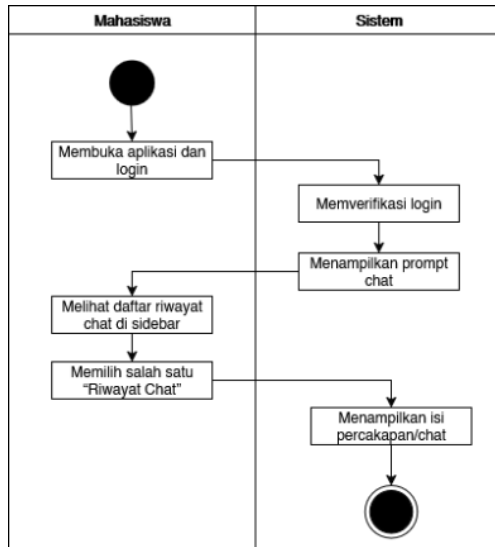


Gambar 7 Activity Diagram koreksi teks

f. *Activity Diagram* riwayat chat

Menunjukkan langkah-langkah yang dilakukan pengguna untuk mengakses riwayat

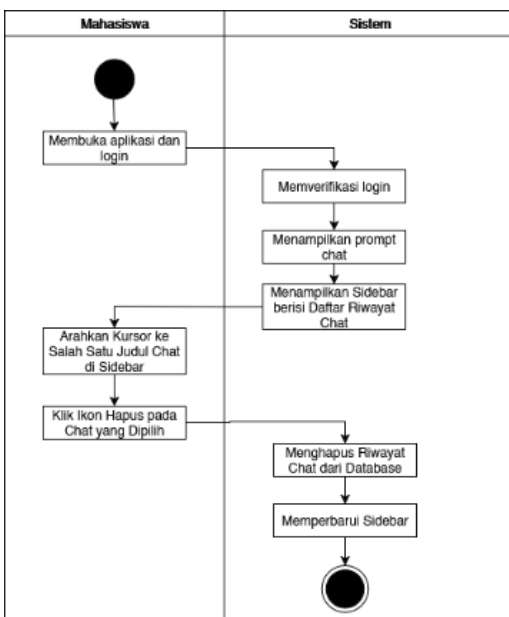
percakapan yang pernah dilakukan di dalam aplikasi. Diagram ini menggambarkan alur aktivitas mulai dari pengguna membuka aplikasi, melakukan login, hingga sistem menampilkan daftar riwayat chat pada tampilan sidebar.



Gambar 8 Activity Diagram riwayat chat

g. Activity Diagram hapus riwayat chat

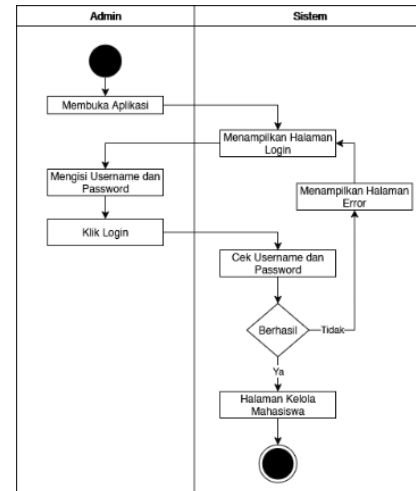
Menunjukkan proses langkah demi langkah dalam prosedur menghapus riwayat percakapan di dalam sistem. Diagram ini menggambarkan alur aktivitas mulai dari pengguna memilih menu riwayat, memilih percakapan yang ingin dihapus, hingga sistem menghapus data tersebut dari database.



Gambar 9 Activity Diagram hapus riwayat chat

h. Activity Diagram login admin

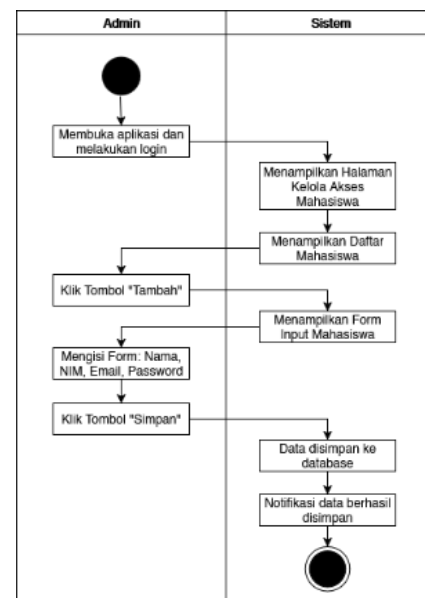
Merupakan visualisasi alur proses autentikasi yang dilakukan oleh Admin saat mengakses sistem.



Gambar 10 Activity Diagram login admin

i. Activity Diagram tambah data mahasiswa

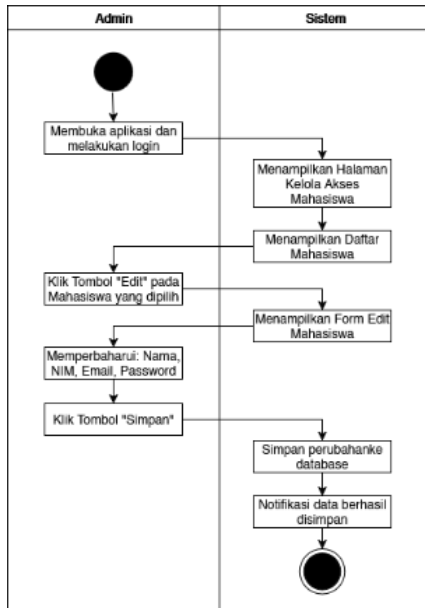
Menggambarkan alur proses yang dilakukan oleh admin saat ingin menambahkan data mahasiswa ke dalam sistem. Diagram ini dimulai dari proses login admin, navigasi ke menu pengguna, pemilihan tombol tambah data, pengisian form data mahasiswa, hingga penyimpanan data ke dalam sistem dan pemberian notifikasi bahwa data berhasil disimpan.



Gambar 11 Activity Diagram tambah data mahasiswa

j. *Activity Diagram* edit data mahasiswa

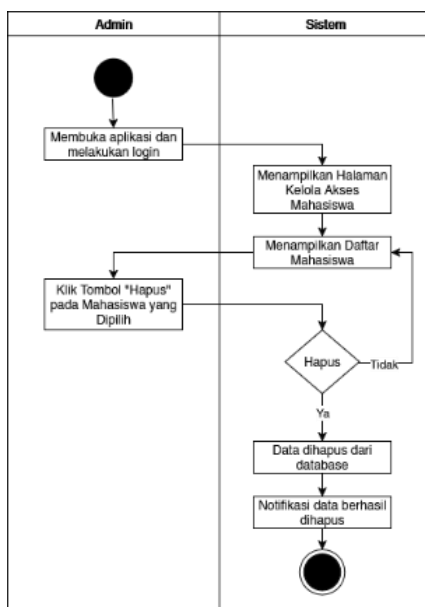
Menjelaskan alur aktivitas admin dalam melakukan proses pengeditan data mahasiswa di dalam sistem. Diagram ini dimulai dari admin yang melakukan login, kemudian diarahkan ke dashboard, hingga proses penyimpanan data yang telah diedit.



Gambar 12 *Activity Diagram* edit data mahasiswa

k. *Activity Diagram* hapus data mahasiswa

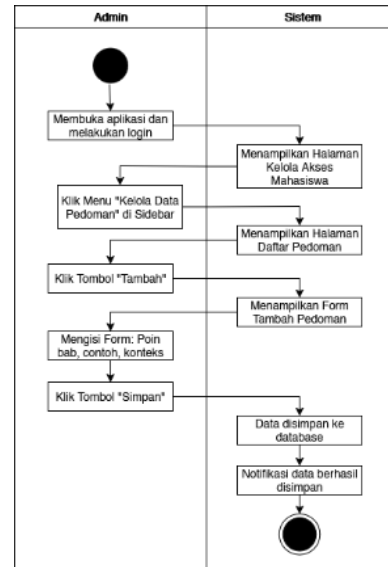
Menggambarkan alur aktivitas admin ketika melakukan proses penghapusan data mahasiswa dari sistem.



Gambar 13 *Activity Diagram* hapus data mahasiswa

l. *Activity Diagram* tambah data pedoman

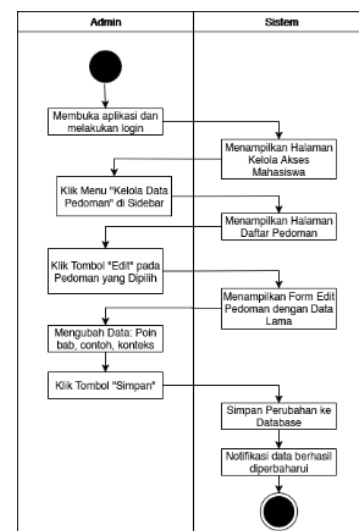
menggambarkan alur aktivitas yang dilakukan oleh admin dalam menambahkan data pedoman ke dalam sistem. Diagram ini memvisualisasikan langkah-langkah proses secara bertahap, mulai dari login hingga penyimpanan data ke dalam database.



Gambar 14 *Activity Diagram* tambah data pedoman

m. *Activity Diagram* edit data pedoman

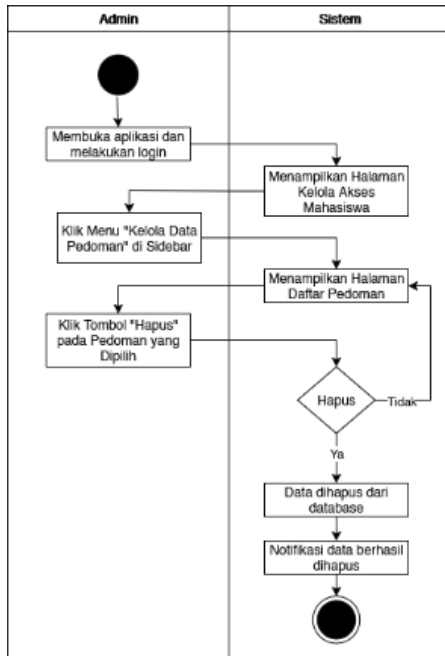
Menggambarkan alur aktivitas yang dilakukan oleh admin saat mengedit data pedoman yang sudah tersedia dalam sistem. Proses ini mencakup langkah-langkah dari login hingga sistem memberikan notifikasi bahwa data berhasil diedit.



Gambar 15 *Activity Diagram* edit data pedoman

n. Activity Diagram hapus data pedoman

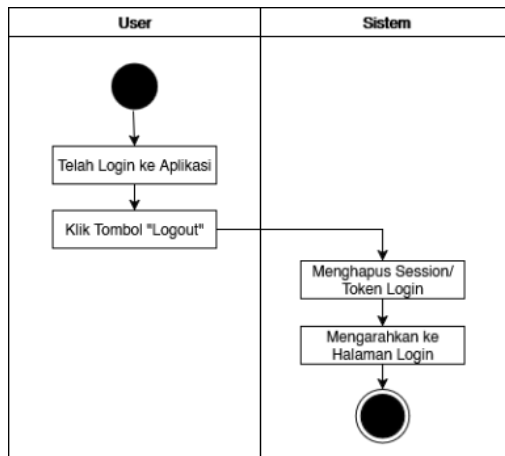
Menggambarkan alur aktivitas yang dilakukan oleh admin ketika ingin menghapus data pedoman dari sistem. Diagram ini menunjukkan proses interaksi antara admin dengan sistem mulai dari login hingga data berhasil dihapus.



Gambar 16 Activity Diagram hapus data pedoman

o. Activity Diagram logout

Menggambarkan proses langkah demi langkah dalam prosedur keluar dari sistem.

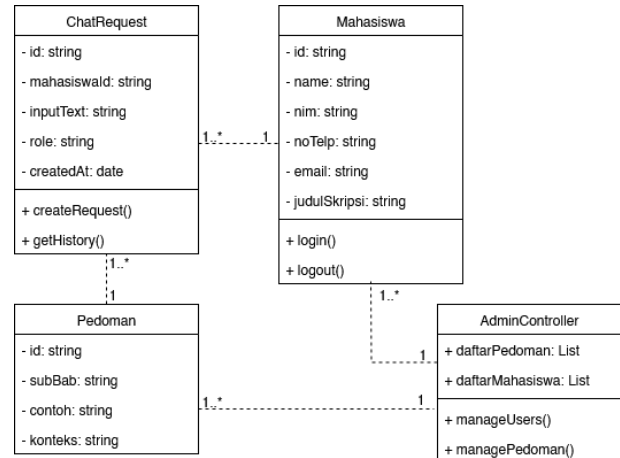


Gambar 17 Activity Diagram logout

A. Class Diagram

Class diagram adalah diagram yang menggambarkan struktur sebuah sistem melalui

definisi kelas-kelas yang digunakan untuk membangun sistem, termasuk atribut, method, serta relasi antar kelas seperti asosiasi, generalisasi, dan dependensi. Sistem terdiri dari beberapa kelas utama yaitu Mahasiswa, ChatRequest, Pedoman, dan AdminController. Kelas Mahasiswa mewakili pengguna sistem yang dapat mengirim permintaan koreksi melalui kelas ChatRequest.



Gambar 18 Class diagram

3. Pengkodean

Tahap implementasi aplikasi berbasis web dilakukan menggunakan Next.js, API kecerdasan buatan (ChatGPT, Gemini dan DeepSeek) serta didukung oleh Firebase. Next.js digunakan untuk membangun antarmuka pengguna dan mengelola rute aplikasi secara efisien. API GPT (API ChatGPT) diintegrasikan sebagai API utama (default), API kecerdasan buatan alternatif, yaitu Gemini dan DeepSeek, yang dapat dipilih pengguna sesuai kebutuhan untuk hasil koreksi yang bervariasi. Firebase digunakan sebagai backend layanan cloud, meliputi autentikasi pengguna, penyimpanan data, serta pengelolaan basis data secara real-time melalui Firestore.

a. Login admin

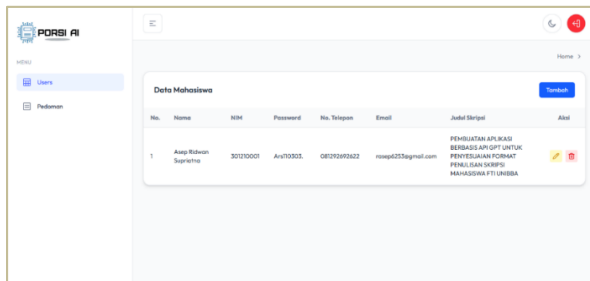
Sebelum mengakses halaman pengelolaan data pada aplikasi PORSI, admin diwajibkan untuk melakukan login terlebih dahulu menggunakan username dan password yang telah ditentukan.



Gambar 19 Halaman login Admin

b. Data mahasiswa

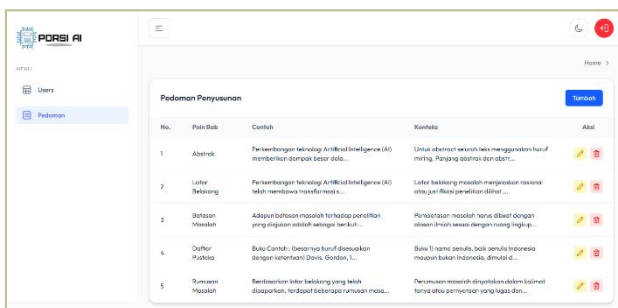
Halaman data mahasiswa merupakan fitur yang digunakan oleh admin untuk mengelola akun mahasiswa yang terdaftar dalam sistem. Admin dapat melihat daftar lengkap mahasiswa beserta informasi penting seperti nama, NIM, password, nomor telepon, email, dan judul skripsi yang sedang dikerjakan. Admin juga memiliki kemampuan untuk menambahkan data, mengubah data, serta menghapus data.



Gambar 20 Halaman kelola data mahasiswa

c. Pedoman

Halaman pedoman merupakan fitur yang digunakan oleh admin untuk mengelola pedoman penulisan skripsi berdasarkan masing-masing poin bab. Setiap pedoman dapat disesuaikan dan diperbarui sesuai dengan standar penulisan yang berlaku di institusi.

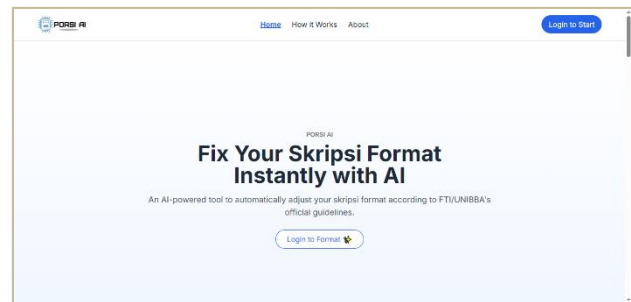


Gambar 21 Halaman kelola pedoman skripsi

d. Landing page

Halaman utama PORSI AI merupakan tampilan awal dari aplikasi yang dirancang untuk

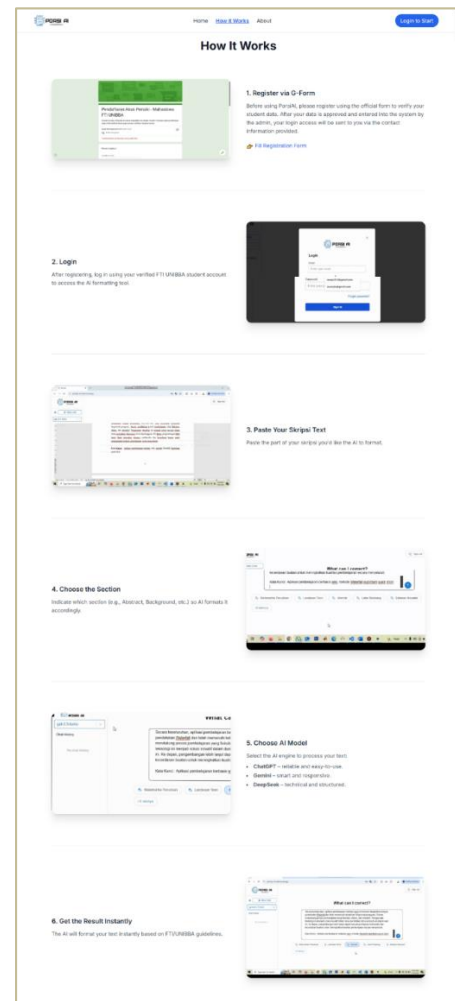
memperkenalkan layanan utama kepada pengguna. Pengguna mendapatkan gambaran umum mengenai fungsi dan manfaat aplikasi.



Gambar 22 Landing Page

e. How it works

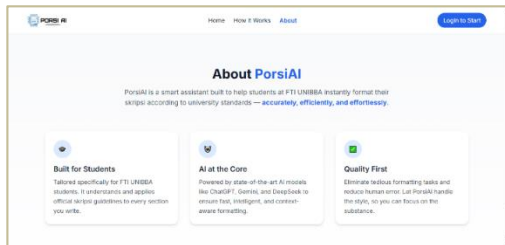
Halaman "how it works" (cara penggunaan) dari aplikasi Porsi AI, yang berfungsi untuk memberikan panduan langkah demi langkah kepada pengguna mengenai cara menggunakan sistem secara efektif.



Gambar 23 Halaman how it works

f. *About*

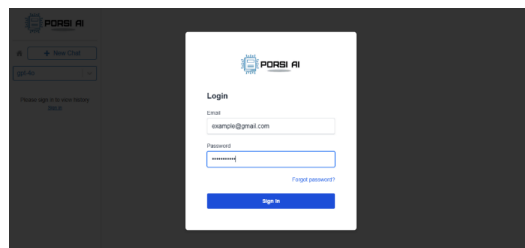
Halaman *about* pada aplikasi PORSI AI berfungsi untuk memperkenalkan tujuan utama dan keunggulan aplikasi kepada pengguna (mahasiswa).



Gambar 24 Halaman about

g. *Login Mahasiswa*

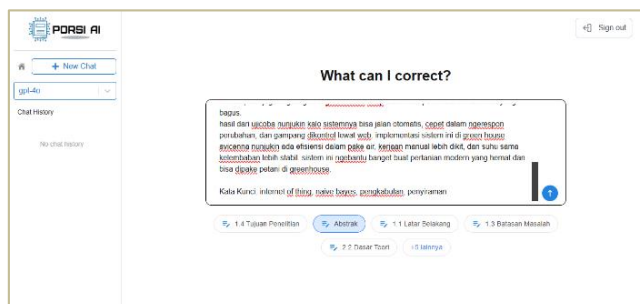
Halaman login mahasiswa merupakan pintu masuk utama bagi pengguna untuk mengakses fitur-fitur yang tersedia dalam aplikasi PORSI AI. Pada halaman ini, mahasiswa diminta untuk memasukkan *email* dan *password* yang telah didaftarkan.



Gambar 25 Halaman login mahasiswa

h. *Prompt chat*

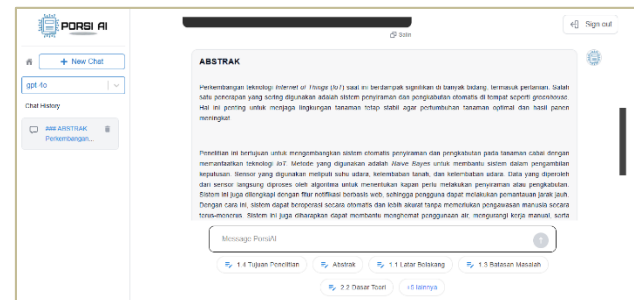
Halaman *prompt chat* memungkinkan mahasiswa untuk berinteraksi dengan sistem berbasis AI dalam proses penyesuaian tata bahasa dan gaya penulisan akademik skripsi. Melalui halaman ini, mahasiswa dapat memasukkan pertanyaan atau instruksi terkait penulisan, dan sistem akan merespons secara otomatis sesuai pedoman yang berlaku.



Gambar 26 Halaman *prompt chat*

i. *Chat response*

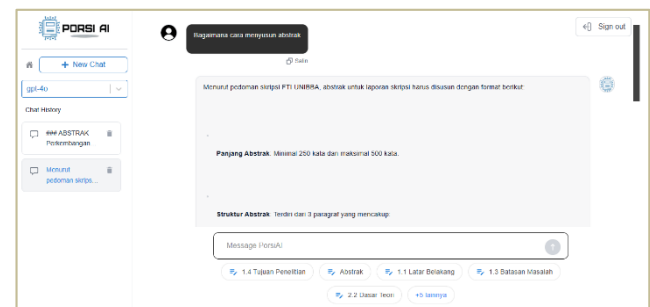
Halaman *chat response* merupakan tampilan yang menampilkan hasil tanggapan atau penyesuaian teks yang dihasilkan oleh sistem AI berdasarkan input yang sebelumnya dikirimkan oleh pengguna. Fitur ini memungkinkan interaksi langsung, sehingga pengguna dapat melihat hasil perbaikan tata bahasa dan gaya penulisan akademik secara instan serta melakukan penyesuaian lanjutan bila diperlukan.



Gambar 27 Halaman *chat response*

j. *Pertanyaan terkait skripsi*

Pertanyaan terkait skripsi merupakan tampilan yang menampilkan interaksi pengguna dengan sistem PORSI AI ketika mengajukan pertanyaan seputar penulisan skripsi.



Gambar 28 Pertanyaan terkait skripsi

4. *Pengujian*

Tahap berikutnya setelah proses perancangan dan pengembangan adalah melakukan pengujian terhadap aplikasi. Melalui pengujian, dapat diketahui apakah aplikasi telah memenuhi kebutuhan pengguna, apakah terdapat bug atau kesalahan dalam sistem, serta bagaimana performa aplikasi ketika digunakan dalam berbagai skenario.

Pada tahap ini, dilakukan pengujian menggunakan metode *Black Box Testing* guna memastikan bahwa fungsi-fungsi dalam aplikasi telah berjalan sesuai dengan perancangan, setiap fitur diuji untuk memastikan bahwa hasil

yang ditampilkan sesuai dengan kebutuhan pengguna dan sesuai dengan spesifikasi yang telah ditetapkan pada tahap perancangan.

a. User

Tabel 2 Pengujian *Black Box* untuk User

No.	Item	Skenario Pengujian	Hasil yang Diharapkan	Hasil Pengujian
1	Login	Mahasiswa mengisi email dan password	Mahasiswa dapat masuk ke dalam sistem PORSI	SESUAI
2	Input Teks Skripsi	Mahasiswa mengetik teks skripsi pada kolom input editor	Sistem menerima dan menyimpan teks sementara	SESUAI
3	Pilih Subbab Skripsi	Mahasiswa memilih salah satu subbab (misal: Latar Belakang)	Sistem merekam pilihan subbab sebagai parameter koreksi	SESUAI
4	Pilih Model AI	Mahasiswa memilih salah satu model AI yang tersedia	Sistem menampilkan pilihan dan menyimpan model yang dipilih	SESUAI
5	Kirim Permintaan Koreksi	Mahasiswa menekan tombol kirim untuk mengirim teks ke AI	Sistem mengirimkan permintaan ke API dan menampilkan hasil penyesuaian	SESUAI
6	Lihat Hasil Penyesuaian	Mahasiswa melihat hasil	Mahasiswa dapat	SESUAI

b. Admin

Tabel 3 Pengujian *Black Box* untuk Admin

No.	Item	Skenario Pengujian	Hasil yang Diharapkan	Hasil Pengujian
1	Login	Admin mengisi email dan password	Admin berhasil masuk ke sistem dan diarahkan ke halaman dashboard	KURANG SESUAI
2	Kelola Data Mahasiswa	Admin menambah, mengedit, atau menghapus data mahasiswa	Data mahasiswa berhasil ditambahkan, diperbarui, atau dihapus sesuai aksi	SESUAI

3	Kelola Pedoman	Admin menambah, mengedit, atau menghapus pedoman skripsi	Data pedoman berhasil dikelola dan tampil di sistem	SESUAI
4	Validasi Akses	Admin mengatur hak akses mahasiswa dalam sistem	Mahasiswa hanya dapat mengakses fitur yang diizinkan sesuai dengan peran	SESUAI

Berdasarkan pengujian aplikasi PORSI AI telah berfungsi sesuai dengan hasil analisis dan perancangan sistem. Seluruh fitur yang dikembangkan mampu berjalan dengan baik dan sesuai dengan tujuan fungsional yang telah ditetapkan sebelumnya.

SIMPULAN DAN SARAN

Berdasarkan hasil penelitian serta proses analisis, perancangan, implementasi, dan pengujian sistem yang telah dilakukan, dapat disimpulkan bahwa aplikasi penyesuaian tata bahasa dan gaya penulisan akademik berbasis API GPT berhasil memenuhi tujuan pengembangan. Integrasi tiga model kecerdasan buatan ChatGPT, Gemini, dan DeepSeek memberikan fleksibilitas bagi pengguna dalam memilih model koreksi yang paling sesuai, dengan hasil pengujian yang menunjukkan bahwa ChatGPT unggul dalam kecepatan dan konsistensi. Antarmuka chat interaktif serta fitur pemilihan sub-bab dan pedoman penulisan terbukti memudahkan mahasiswa dalam memperoleh koreksi teks yang relevan dan sesuai format akademik. Pengujian black box menunjukkan seluruh fitur berjalan dengan baik, sementara validasi pengguna mengindikasikan bahwa aplikasi ini layak digunakan sebagai alat bantu percepatan proses revisi skripsi. Selain itu, penerapan metode SDLC *Waterfall* memberikan alur pengembangan sistem yang runtut dan terstruktur, sehingga menghasilkan aplikasi yang stabil dan berkualitas.

Untuk pengembangan lebih lanjut, aplikasi ini masih memiliki ruang peningkatan agar lebih optimal dalam mendukung kebutuhan penulisan akademik. Pertama, penambahan fitur unggah

dokumen skripsi secara utuh diperlukan agar sistem dapat mendeteksi struktur dokumen secara otomatis tanpa input per bab. Kedua, penyediaan rekomendasi otomatis atas hasil koreksi dari ketiga model AI akan mengurangi kebutuhan perbandingan manual oleh pengguna.

Selain itu, integrasi fitur pengecekan plagiarisme dan validasi sitasi otomatis sangat disarankan agar aplikasi mampu memberikan dukungan yang lebih komprehensif, tidak hanya dari segi tata bahasa tetapi juga dari aspek orisinalitas dan kelengkapan referensi. Fitur pelatihan singkat atau tutorial dalam aplikasi juga diperlukan untuk meningkatkan pemahaman pengguna dalam menggunakan sistem secara optimal. Selanjutnya, pengembangan sistem rekomendasi model AI berdasarkan jenis kesalahan atau konteks sub-bab dapat meningkatkan efisiensi proses revisi.

Dari sisi keamanan, implementasi fitur pemulihan kata sandi, autentikasi dua langkah, dan CAPTCHA perlu dipertimbangkan untuk mencegah penyalahgunaan akses. Terakhir, penambahan fitur deteksi otomatis terhadap kesalahan teknis format seperti margin, spasi paragraf, ukuran kertas, dan jenis huruf akan semakin mempermudah mahasiswa dalam memastikan kesesuaian dengan standar penulisan akademik.

PUSTAKA ACUAN

- Arshad Busro Cahyono, S., Firliana, R., Najibuloh Muzzaki, M., Sari Wardani, A., Iqbal Khalid, M., Wicak Milbar Gamas, A., & Setiawan, H. (2022). *Bulletin of Information Technology (BIT) Rancangan Pembuatan Api Website Data Tanaman Obat Dan Langka Kabupaten Kediri*. 3(4), 255–260. <https://doi.org/10.47065/bit.v3i1>
- Aryabimo, A. R., Bernady, D., Sari, N. N. K., & Pranatawijaya, V. H. (2024). IMPLEMENTASI API CHAT GPT PADA APLIKASI RESTORAN BERBASIS WEBSITE. *Jurnal Informatika Dan Teknik Elektro Terapan*, 12(3). <https://doi.org/10.23960/jitet.v12i3.4408>
- DeepSeek-AI, :, Bi, X., Chen, D., Chen, G., Chen, S., Dai, D., Deng, C., Ding, H., Dong, K., Du, Q., Fu, Z., Gao, H., Gao, K., Gao, W., Ge, R., Guan, K., Guo, D., Guo, J., ... Zou, Y. (2024). *DeepSeek LLM: Scaling Open-Source Language Models with Longtermism*. <http://arxiv.org/abs/2401.02954>
- Direktorat Pembelajaran dan Kemahasiswaan. (2024). Panduan penggunaan Generative Artificial Intelligence (GenAI) pada pembelajaran di perguruan tinggi. *Direktorat Jenderal Pendidikan Tinggi, Riset, Dan Teknologi, Kementerian Pendidikan, Kebudayaan, Riset, Dan Teknologi*.
- Draw.io. (n.d.). *About draw.io*. Retrieved June 16, 2025, from <https://www.drawio.com/about>
- Ekin, S. (2023). *Prompt Engineering For ChatGPT: A Quick Guide To Techniques, Tips, And Best Practices*. <https://doi.org/10.36227/techrxiv.22683919.v1>
- Eriana, E. S., & Zein, A. (2023). *ARTIFICIAL INTELLIGENCE (AI) PENERBIT CV. EUREKA MEDIA AKSARA*.
- Fenton, S. (2018). *Pro TypeScript*. In *Pro TypeScript*. Apress. <https://doi.org/10.1007/978-1-4842-3249-1>
- Figma. (n.d.). *Figma Newsroom*. Retrieved June 16, 2025, from <https://www.figma.com/newsroom/>
- Firebase Google. (n.d.). *Documentation Firestore*. Retrieved May 9, 2025, from <https://firebase.google.com/docs/firestore?hl=id>
- Fitriyanto, A., & Fitriani, A. S. (2024). Aplikasi Penjualan Tas Di Indonesia Berbasis Web Menggunakan Metode Waterfall. *Indonesian Journal of Applied Technology*, 1(2), 32. <https://doi.org/10.47134/ijat.v1i2.3046>
- Google Cloud. (n.d.). *Model Bahasa Besar yang didukung oleh AI Google kelas dunia*.
- Hariyadi, Noor Kamala Sari, N., Irawan, J., & Handrianus Pranatawijaya, V. (2024). Implementasi Gemini API untuk Generatif Teks Deskripsi Karya Otomatis dalam Aplikasi Pameran Berbasis Web dengan Metode Waterfall. In *Jurnal Teknologi Informasi dan Terapan (J-TIT)* (Vol. 11, Issue 1). <https://doi.org/10.25047/jtit.v11i1.356>
- Hasanah, F. N., & Untari, R. S. (2020). *BUKU AJAR REKAYASA PERANGKAT LUNAK Diterbitkan oleh UMSIDA PRESS UNIVERSITAS MUHAMMADIYAH SIDOARJO 2020*.
- Hausawi, Y. M., & Mayron, L. M. (2013). Towards usable and secure *Natural Language*

- Processing systems. *Communications in Computer and Information Science*, 373(PART I), 109–113. https://doi.org/10.1007/978-3-642-39473-7_23
- Maulida, R., Kunci-ChatGPT, K., & Box, B. (2024). *Komparasi Respons ChatGPT dan Gemini terhadap Command Pattern Identik dengan Metode Black Box*.
- Microsoft. (n.d.-a). *TypeScript is JavaScript with syntax for types*. Retrieved May 9, 2025, from <https://www.typescriptlang.org/>
- Microsoft. (n.d.-b). *visualstudio microsoft*. Retrieved June 16, 2025, from <https://visualstudio.microsoft.com/>
- Rachbini, W., Evi, T., & Suyanto. (2023). *Pengenalan ChatGPT Tips dan Trik bagi Pemula*.
- Raiaan, M. A. K., Mukta, M. S. H., Fatema, K., Fahad, N. M., Sakib, S., Mim, M. M. J., Ahmad, J., Ali, M. E., & Azam, S. (2024). A Review on Large Language Models: Architectures, Applications, Taxonomies, Open Issues and Challenges. *IEEE Access*, 12, 26839–26874. <https://doi.org/10.1109/ACCESS.2024.3365742>
- Ramanda Kalawa Putri, M., Fajari, R., Handrianus Pranatawijaya, V., & Noor Kamala Sari, N. (2024). PENERAPAN API PADA DASHBOARD ADMIN SALON BERBASIS WEBSITE. In *Jurnal Mahasiswa Teknik Informatika* (Vol. 8, Issue 3).
- Rosa, A. S., & Shalahuddin, M. (2016). *Rekayasa Perangkat lunak*. In *Rekayasa Perangkat Lunak*.
- 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>
- Schulhoff, S. (2024). *Struktur prompt dasar*. https://Learnprompting.Org/Id/Docs/Basics/Prompt_structure.
- Sumirat, L. P., Cahyono, D., Kristyawan, Y., & Kacung, S. (2023). *DASAR-DASAR Rekayasa Perangkat Lunak*. www.madzamedia.co.id
- Uminingsih, Nur Ichsanudin, M., & Yusuf, M. (2022). *PENGUJIAN FUNGSIONAL PERANGKAT LUNAK SISTEM INFORMASI PERPUSTAKAAN DENGAN METODE BLACK BOX TESTING BAGI PEMULA*. 1(2), 1–8. <https://doi.org/10.55123>
- Vercel. (n.d.). *Next.js documentation*.
- Yadav, S., Jindal, R., & Anand, S. (2020). A Study on Comparison of UML and ER Diagram. *International Research Journal of Engineering and Technology*. www.irjet.net
- Yosli, R. (2021). Meningkatkan Kapasitas Hosting, Mengelola Content Management System Untuk Kenyamanan Memakai Website Berbayar. *JAVIT: Jurnal Vokasi Informatika*, 31–37. <https://doi.org/10.24036/javit.v1i2.6>