

PENGEMBANGAN FITUR Pencarian Informasi Obat Berbasis AI PADA APLIKASI POTIO

Oleh:

Allycia Joshin^{1*}, Moh. Sofyan S. Thayf^{2*}, Zaenab Pontoh³

^{1,2,3}Sistem Informasi, STMIK KHARISMA Makassar

e-mail: ¹allyciajoshin_22@kharisma.ac.id, ²sofyan.thayf@kharisma.ac.id*,
³zaenabp@kharisma.ac.id

Abstrak: Banyak pengguna sering melupakan jadwal minum obat, sehingga dibutuhkan solusi yang efektif untuk meningkatkan kepatuhan. Potio hadir sebagai aplikasi alarm berbasis mobile yang berfungsi sebagai pengingat minum obat, dengan fitur utama berupa pencatatan alarm, riwayat minum obat, penggantian tema, dan pencarian informasi obat. Dalam pengembangannya, aplikasi ini menggunakan database MySQL untuk menyimpan hasil inputan data alarm dan informasi obat. Namun, dengan meningkatnya jumlah data yang perlu dikelola, peneliti menghadapi tantangan dalam pengelolaan data, karena jumlah obat yang sangat banyak perlu diinput secara manual, sehingga membutuhkan waktu lebih lama dan terdapat potensi kesalahan dalam penginputan data. Oleh karena itu, dilakukan pengembangan fitur pencarian informasi obat pada aplikasi Potio menggunakan salah satu model, yaitu Gemini AI, dengan cara mengambil kunci Application Programming Interface untuk diimplementasikan ke dalam fitur pencarian informasi obat, diikuti dengan teknik prompt untuk memberikan output secara efektif. Hasil pengujian menggunakan Blackbox Testing dengan test case menunjukkan bahwa Potio berhasil memanfaatkan Gemini AI untuk fitur pencarian informasi obat, sehingga tidak perlu menginput informasi obat secara manual ke dalam database MySQL. Penelitian ini menunjukkan bahwa adanya pemanfaatan AI dapat meningkatkan efisiensi dalam pengelolaan data informasi obat.

Kata kunci: artificial intelligence, gemini AI, API, blackbox testing, pencarian informasi obat.

Abstract: Many patients often forgot their medication schedules, highlighting the need for an effective solution to improve adherence. Potio was a mobile-based alarm application designed as a medication reminder, featuring alarm scheduling, medication history, theme customization, and medicine information search. In its development, this application used a MySQL database to store the results of inputting alarm data and medicine information. However, with the increasing amount of data that needed to be managed, researchers faced challenges in data management, because the very large number of medicines needed to be inputted manually, so it took longer and there was the potential for errors in data input. Therefore, the development of a medicine information search feature in the Potio application was carried out using one of the models, namely Gemini AI, by obtaining an Application Programming Interface key to be implemented into the medicine information search feature, followed by a prompt technique to provide effective output. The results of testing using Blackbox Testing with test cases showed that Potio successfully utilized Gemini AI for the medicine information search feature, eliminating the need for manual input of medicine information into the MySQL database. This indicated AI adoption can improve efficiency in data management.

Keywords: artificial intelligence, gemini AI, API, blackbox testing, medicine information search feature.

* Corresponding author : Moh. Sofyan S. Thayf (sofyan.thayf@kharisma.ac.id)

1. PENDAHULUAN

Potio adalah aplikasi alarm berbasis *mobile* yang berfungsi sebagai pengingat minum obat bagi pengguna yang sering melupakan jadwal obatnya. *Main feature* pada *software* ini adalah pencatatan alarm yang terdapat pada halaman alarm, di mana pengguna dapat menginput nama obat, mengatur waktu, hari, dan memilih nada dering sesuai keinginan. Selain pencatatan alarm, terdapat juga beberapa lainnya, seperti fitur riwayat minum obat yang menampilkan *history* dari alarm yang telah di atur pada tambah alarm sebelumnya di halaman riwayat, penggantian tema gelap atau terang yang dapat dipilih sesuai keinginan, dan pencarian informasi obat yang menampilkan informasi obat, seperti deskripsi, dosis, serta aturan pakai setiap obat. Untuk fitur tema dan pencarian informasi obat terdapat pada halaman pengaturan. Bagi yang belum pernah menggunakannya, pengguna wajib mendaftar akun dengan memasukkan *username*, alamat Gmail, serta *password* terlebih dahulu untuk mengakses aplikasi tersebut.

Potio dapat diunduh melalui *website* resmi dengan alamat <https://potio.my.id/> dan saat ini telah digunakan sebanyak 57 pengguna. Dalam pengembangannya, aplikasi ini menggunakan *database* MySQL untuk menyimpan hasil inputan data alarm dan informasi obat. Namun, dengan meningkatnya jumlah data yang perlu dikelola, peneliti menghadapi tantangan dalam pengelolaan data, karena jumlah obat yang sangat banyak perlu diinput secara manual, sehingga membutuhkan waktu lebih lama dan terdapat potensi kesalahan dalam penginputan data. Oleh karena itu, dibutuhkan pemanfaatan teknologi yang dapat mendukung pengembangan fitur dan peningkatan kualitas layanan aplikasi, salah satunya dengan menggunakan *artificial intelligence* (AI).

AI merupakan teknologi yang telah berkembang pesat dan memiliki pengaruh signifikan pada berbagai kehidupan masyarakat, bahkan mengubah banyak aspek dalam kehidupan manusia [1]–[3]. Hal ini karena AI telah banyak membawa dampak pada berbagai sektor, seperti finansial, kesehatan, manufaktur, dan sebagainya [4], [5]. Perkembangan AI membuat berbagai proses menjadi lebih efisien, serta tugas-tugas yang saat ini terlihat kompleks dapat diselesaikan dengan lebih cepat dan akurat [6].

Berdasarkan uraian di atas, peneliti melakukan survei pada 30 pengguna yang merupakan *user* sama dari survei *closed testing* pada Google Play sebelumnya. Hasil dari survei menjelaskan kebanyakan *user* menginginkan penambahan data, karena masih banyak yang belum tersedia dalam fitur tersebut, yang mengakibatkan pengguna tidak dapat menemukan deskripsi obat yang dikonsumsi. Oleh karena itu, tujuan penelitian ini adalah mengembangkan fitur pencarian informasi obat pada aplikasi Potio dengan menggunakan AI.

Model AI yang dipilih merupakan Gemini AI, karena memiliki banyak model yang disediakan oleh pihak Google AI Studio dan dapat digunakan secara gratis. Model ini diimplementasikan dengan cara mengambil kunci *Application Programming Interface* (API) Gemini AI, kemudian diimplementasikan pada fitur pencarian informasi obat aplikasi ini, diikuti dengan *Few-Shot prompt* di dalam pengembangannya. *Few-Shot prompt* merupakan cara di mana AI menyelesaikan suatu tugas dengan diberikan beberapa contoh pada *prompt*-nya [7].

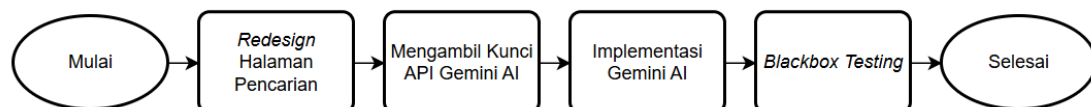
Penelitian sebelumnya yang telah dilakukan terkait pengembangan berbasis AI dengan menggunakan Gemini AI, memiliki perbedaan dengan penelitian ini. Salah satunya adalah “Penerapan Gemini AI Dalam Pembuatan Deskripsi Produk *E-Commerce*” [8] yang melakukan penerapan menggunakan Gemini AI untuk proses *generate* deskripsi pada sebuah produk *e-commerce* berbasis *website*, sedangkan penelitian ini adalah mengembangkan fitur pencarian informasi obat pada aplikasi Potio berbasis *mobile*. Terdapat juga hasil penelitian berjudul “Implementasi *Large Language Models* Gemini Pada Pengembangan Aplikasi Chatbot Berbasis Android” [9], melakukan pengembangan aplikasi Chatbot yang berbasis *mobile* dengan mengimplementasikan kunci API Gemini AI pada aplikasi tersebut, sedangkan penelitian ini menggunakan kunci API Gemini AI untuk diimplementasikan ke dalam aplikasi Potio pada bagian fitur pencarian informasi obat. Selanjutnya, penelitian yang berjudul “Penerapan API Gemini dalam Layanan Peminjaman Novel Online pada *Website* Cozybook” [4], menjelaskan bagaimana *website* Cozybook dikembangkan dengan mengimplementasikan kunci API dari Gemini AI untuk memudahkan pengguna mencari dan membaca novel. Perbedaannya adalah penelitian ini mengimplementasikan API Gemini AI untuk fitur pencarian informasi obat pada aplikasi Potio berbasis *mobile*. Terakhir, hasil penelitian yang dilakukan oleh Makhulu Relebogile Langa dan Michael Nthabiseng Moeti berjudul “*Enhancing Online Class Engagement through Generative AI: A Research using Microsoft Graph and Gemini APIs*” [10], menjelaskan tentang pemanfaatan Microsoft Graph API untuk mengumpulkan transkrip atau rekaman kuliah daring dan Gemini AI untuk menghasilkan pertanyaan berbasis materi yang diajarkan. Perbedaan dengan penelitian ini adalah hanya menggunakan API dari Gemini AI untuk menghasilkan data informasi obat.

Dari studi yang telah dilakukan terhadap penelitian sebelumnya, menjelaskan bahwa pemanfaatan Gemini AI masih berfokus pada konteks selain aplikasi kesehatan. Oleh karena itu, kontribusi ilmiah dari penelitian ini adalah menghadirkan pendekatan baru dalam pemanfaatan Gemini AI untuk mengembangkan fitur pencarian informasi obat pada aplikasi *mobile*, yaitu Potio.

2. METODE PENELITIAN

2.1. Tahapan Penelitian

Ada beberapa tahapan yang dilakukan dalam penelitian ini, di antaranya sebagai berikut.



Gambar 1. Tahapan Penelitian

a. Identifikasi Masalah

Peneliti mencari tahu permasalahan dari aplikasi Potio, dengan melihat dari berbagai fitur. Permasalahan yang ditemukan ada pada fitur pencarian informasi obat. Hal ini karena peneliti menghadapi tantangan dalam pengelolaan data, karena jumlah obat yang sangat banyak perlu diinput secara manual, sehingga membutuhkan waktu lebih lama dan terdapat potensi kesalahan dalam penginputan data. Setelah melakukan survei pada 30 jumlah pengguna, hasil menjelaskan kebanyakan *user* menginginkan penambahan data, karena masih banyak yang belum tersedia dalam fitur tersebut, yang mengakibatkan pengguna tidak dapat menemukan deskripsi obat yang dikonsumsi.

b. *Redesign* Halaman Pencarian

Pada tahap ini, peneliti melakukan *redesign* di Android Studio sebagai bagian dari pengembangan fitur pencarian informasi obat tersebut. Melalui tahapan ini, peneliti merancang ulang *user interface* (UI) pada bagian fitur pencarian informasi obat di Android Studio untuk diimplementasikan dengan API Gemini AI.

c. Mengambil Kunci API Gemini AI

Setelah melakukan *redesign*, peneliti mengambil kunci API Gemini AI di *website* Google AI Studio yang tersedia gratis oleh Google dengan alamat <https://aistudio.google.com/>. Untuk mengambil kunci tersebut, peneliti masuk ke dalam *website*-nya terlebih dahulu, lalu *me-request* API agar kuncinya *di-generate* oleh pihak Google AI Studio.

d. Implementasi API Gemini AI

Setelah tahapan sebelumnya, peneliti mengimplementasikan kunci API ke dalam fitur pencarian informasi obat informasi obat yang UI-nya telah di *redesign* sebelumnya. Teknik yang digunakan berupa *prompting*, di mana peneliti melakukan *coding* di Android Studio pada halaman pencarian informasi obat untuk mengatur informasi obat yang diberikan Gemini AI. Dengan *prompting*, *user* dapat menginput nama obat, Gemini AI hanya akan menampilkan hasil berupa deskripsi, dosis, kandungan, indikasi, dan kontradiksi secara konsisten. Peneliti juga mengatur dari segi kebahasaan informasi yang ditampilkan, sehingga Gemini AI hanya akan menampilkan informasi obat berbahasa Indonesia, mengingat aplikasi Potio menggunakan Bahasa Indonesia. Selain itu, peneliti menetapkan batasan, di mana ketika pengguna memasukkan kata di luar nama obat, AI tidak akan menampilkan informasi tersebut. Peneliti juga membuat kelas baru untuk menghubungkan Gemini AI dengan Potio melalui kunci API yang telah diambil sebelumnya, serta mengatur model Gemini AI yang dipilih, yaitu Gemini 2.5 Flash.

e. *Blackbox Testing*

Peneliti melakukan *Blackbox Testing* dengan menggunakan *test case* untuk menguji fungsionalitas fitur pencarian informasi obat yang berbasis AI. *Blackbox Testing* merupakan metode pengujian yang berfokus pada fungsionalitas *software*, tanpa melihat struktur kode yang ingin diuji [11]. Alasan pemilihan *Blackbox Testing* adalah peneliti ingin berfokus pada hasil pengembangan fitur pencarian informasi obat berbasis AI yang telah dilakukan, sehingga metode pengujian ini cocok untuk menguraikan hasil dilakukannya pengembangan pada aplikasi Potio.

2.2. Pengumpulan Data

a. Studi literatur.

Peneliti mencari penelitian terkait yang mendukung dari berbagai sumber literatur, yang akan dijadikan sebagai acuan penelitian ini.

b. Mengambil sumber data.

Pada hal ini, sumber datanya adalah hasil dari pengujian *Blackbox Testing*.

2.3. Metode Pengujian

Peneliti melakukan *Blackbox Testing* dengan *test case* untuk menguji fungsionalitas fitur pencarian informasi obat yang berbasis AI. Adapun pengujian ini dilakukan untuk mengecek apakah fitur pencarian yang telah diimplementasi menggunakan AI dapat berjalan dengan lancar atau terdapat eror selama dijalankan.

3. HASIL DAN PEMBAHASAN

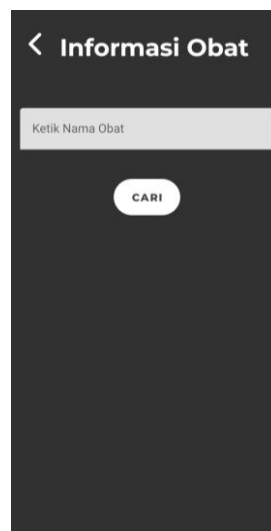
3.1. *Redesign* Halaman Pencarian

Dalam tahapan *redesign* ini, peneliti melakukan sedikit perubahan pada kelas `fragment_pengaturan.xml` untuk mengubah kata "Pencarian" di bawah tombol pencarian menjadi "Informasi Obat", agar memperjelas bahwa fitur pencarian ini berfungsi untuk menampilkan informasi obat. Perubahan tampilan ini dapat dilihat pada Gambar 2.



Gambar 2. Tampilan Halaman Pengaturan Sesudah *Redesign*

Selanjutnya, peneliti melakukan *redesign* pada kelas `activity_pencarian.xml` dan melakukan *coding* pada kelas `Pencarian.java` untuk menyesuaikan fungsionalitas *layout*-nya dan membuat tampilannya menjadi lebih sederhana, di mana *search bar* dan informasi obat berada pada halaman yang sama, serta tambahan tombol “Cari” untuk menampilkan informasi obat seperti pada Gambar 3.

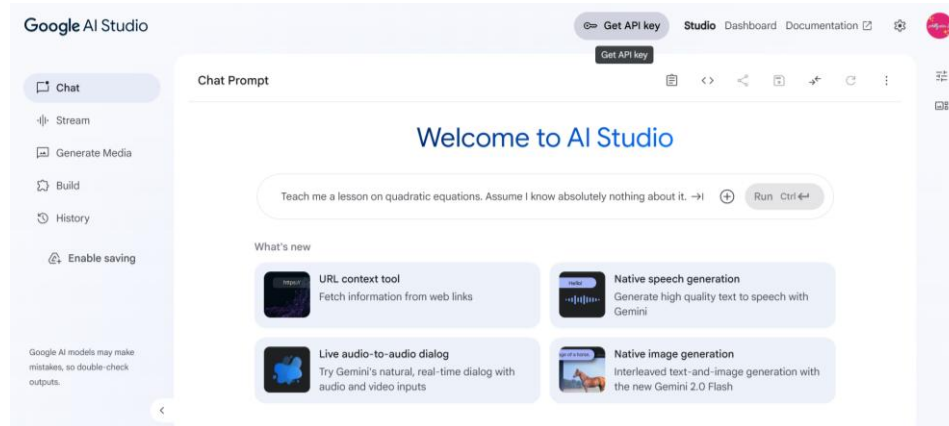


Gambar 3. Tampilan Halaman Pencarian Informasi Obat Sesudah *Redesign*

Oleh karena itu, fitur pencarian informasi obat pada aplikasi Potio tidak perlu terhubung lagi dengan *database* MySQL untuk menampilkan informasi obat seperti sebelumnya.

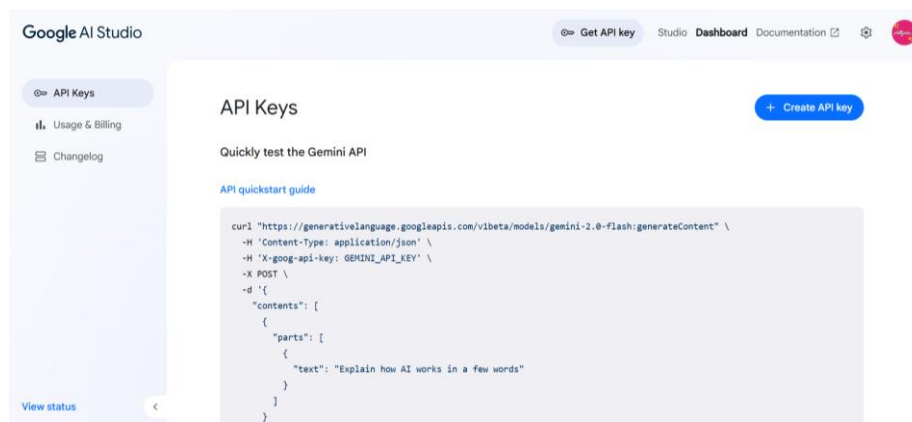
3.2. Mengambil Kunci API Gemini AI

Setelah melakukan *redesign*, peneliti mengambil kunci API Gemini AI di *website* Google AI Studio yang tersedia gratis oleh Google dengan alamat <https://aistudio.google.com/>. Untuk mengambil kunci tersebut, peneliti masuk ke dalam *website*-nya terlebih dahulu, di mana tampilan awalnya seperti pada Gambar 4.



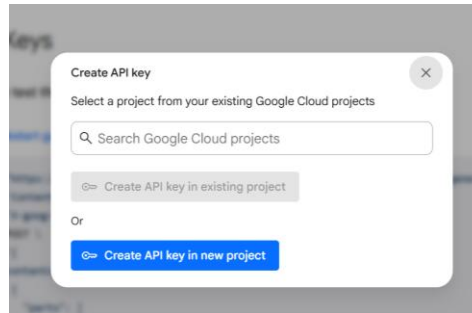
Gambar 4. Tampilan Halaman Google AI Studio

Pada bagian kanan atas, terdapat tombol “Get API Key”, di mana ketika peneliti menekannya, maka halaman depan berpindah ke halaman berikut, seperti pada Gambar 5.



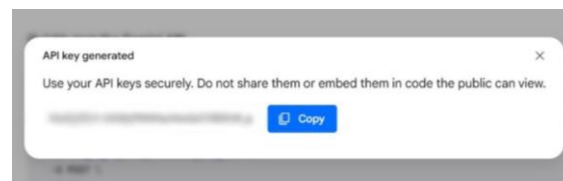
Gambar 5. Tampilan Halaman Get API Key

Di dalam halaman Get API Key, terdapat tombol biru Create API Key pada kanan atas. Untuk mendapatkan kunci API, klik tombol tersebut, kemudian pilih tombol Create API key in new project, seperti pada Gambar 6.



Gambar 6. Create API Key

Setelah klik tombol tersebut, tunggu beberapa detik selama kunci API di-*generate* secara otomatis oleh Google AI Studio hingga selesai. Jika berhasil ter-*generate*, kunci API diberikan dan siap di-*copy* untuk diimplementasikan seperti pada Gambar 7.



Gambar 7. Kunci API Gemini AI

3.3. Implementasi Gemini AI

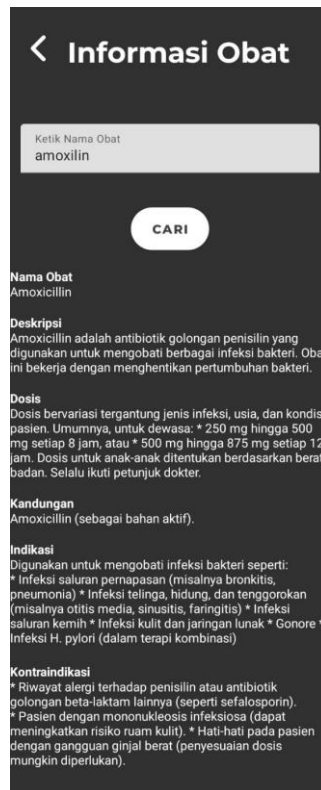
Dalam Android Studio, peneliti membuat beberapa kelas baru untuk mengimplementasikan kunci API yang telah diambil sebelumnya, yaitu kelas `BuildConfig.java` dan `GeminiPro.java`. Selain kelas baru, peneliti juga menambahkan *prompt* pada kelas `Pencarian.java`.

Pada kelas `BuildConfig.java`, peneliti memasukkan kunci API Gemini AI yang telah diambil pada tahapan sebelumnya. Dalam kelas `GeminiPro.java`, peneliti mengirimkan *query* ke model Gemini AI, yaitu Gemini 2.5 Flash untuk mendapatkan respons berupa teks dari model tersebut. Dalam hal ini, informasi obat yang ditampilkan otomatis berformat teks. Kelas ini juga berfungsi menghubungkan kunci API yang dimasukkan ke dalam `BuildConfig.java`

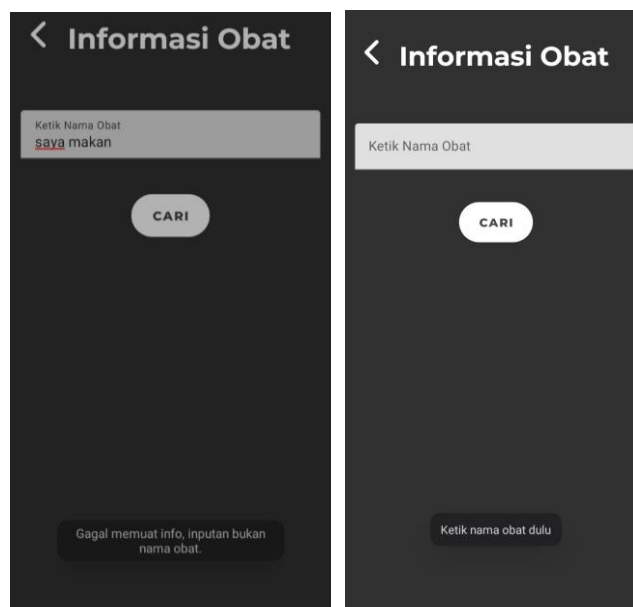
Untuk kelas `Pencarian.java`, peneliti memberikan *prompt* (*Few-shot*) berupa konsistensi *output* nama obat, deskripsi, dosis, kandungan, indikasi, serta kontraindikasi. Selain itu, peneliti juga menambahkan *prompt* agar Gemini AI hanya akan menampilkan semua informasi obat hanya dalam bahasa Indonesia, dengan memberikan contoh kalimat *prompt* :

- "Saya hanya ingin Anda menjawab jika saya memberikan nama obat."
- "Jika saya menuliskan nama obat, tolong tampilkan hanya dalam bahasa Indonesia informasi berikut"
- "Secara ringkas dan terstruktur: Nama Obat, Deskripsi, Dosis, Kandungan, Indikasi, dan Kontraindikasi."
- "Jangan menjawab jika input saya bukan nama obat. Nama obat:"

Dengan ini, setiap kali pengguna menginput nama obat, Gemini AI memberikan *output* berupa nama obat, deskripsi, dosis, kandungan, indikasi, dan kontraindikasi dalam bahasa Indonesia secara konsisten yang bisa dilihat pada Gambar 8. Selain itu, Gemini AI tidak melayani inputan di luar nama obat maupun inputan kosong seperti pada Gambar 9.

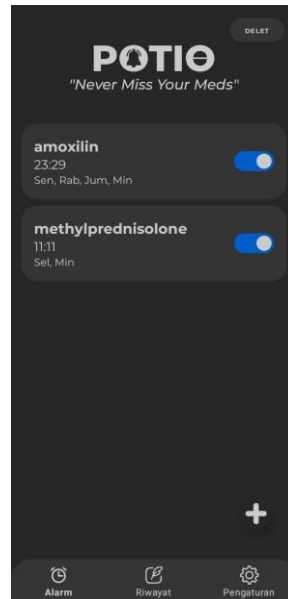


Gambar 8. Tampilan Halaman Pencarian Informasi Obat Implementasi Gemini AI



Gambar 9. Gemini AI Tidak Memberikan Informasi Di Luar Nama Obat dan Inputan Kosong

Tidak hanya mengetik di *search bar*, peneliti juga menambahkan kelas AlarmAdapter.java yang berfungsi sebagai penghubung halaman alarm dan pencarian informasi obat. Dengan adanya kelas tersebut, pengguna dapat mencari informasi obat hanya dengan mengklik nama obat dari alarm yang telah disetel di halaman alarm, seperti pada Gambar 10. Ketika nama obat ditekan, maka secara otomatis berpindah ke halaman pencarian dengan Gemini AI secara bersamaan meng-*generate* untuk menampilkan informasi obat berupa nama obat, deskripsi, dosis, kandungan, indikasi, dan kontraindikasi (Gambar 8).



Gambar 10. Halaman Alarm

3.4. Blackbox Testing

Salah satu pengujian yang dapat dilakukan, yaitu *Blackbox Testing* dengan *test case*. Untuk Potio, peneliti menguraikan skenario (*test case*) pengujian berikut.

1. Konsistensi informasi obat untuk memberikan *output* nama obat, deskripsi, dosis, kandungan, indikasi, dan kontraindikasi.
2. Konsistensi bahasa, di mana AI hanya memberikan *output* informasi obat berbahasa Indonesia.
3. Konsistensi validasi inputan, di mana AI hanya memberikan *output* informasi apabila pengguna menginput nama obat. Inputan yang tidak relevan atau di luar nama obat, tidak di tampilkan informasinya oleh AI dan muncul pesan "Gagal memuat info, inputan bukan nama obat".
4. Konsistensi validasi inputan, di mana AI hanya memberikan *output* informasi apabila pengguna menginput nama obat. Inputan yang kosong, tidak di tampilkan informasinya oleh AI dan muncul pesan "Ketik nama obat dulu".
5. Ketika alarm telah disetel, maka nama obat, waktu, dan hari muncul sebagai sebuah *list* di halaman alarm. Apabila pengguna memencet nama obat tersebut, navigasi harus

berfungsi dengan semestinya, yaitu halaman alarm berpindah ke halaman pencarian dan Gemini AI dapat menampilkan informasi nama obat, deskripsi, dosis, kandungan, indikasi, dan kontraindikasi.

Berdasarkan skenario tersebut, maka dibuat uraian *test case* yang ditunjukkan pada Tabel 1.

Tabel 1. Tabel *Test Case*

ID	Test Case	Hasil yang Diharapkan	Hasil Pengujian
1.	Memberikan <i>output</i> nama obat, deskripsi, dosis, kandungan, indikasi, dan kontraindikasi secara konsisten.	Gemini AI meng- <i>generate</i> untuk memberikan <i>output</i> nama obat, deskripsi, dosis, kandungan, indikasi, dan kontraindikasi secara konsisten pada halaman pencarian informasi obat.	Berhasil memberikan <i>output</i> nama obat, deskripsi, dosis, kandungan, indikasi, dan kontraindikasi secara konsisten pada halaman pencarian informasi obat.
2.	Memberikan <i>output</i> informasi obat dalam bahasa Indonesia.	Gemini AI meng- <i>generate</i> untuk memberikan <i>output</i> informasi obat dalam bahasa Indonesia.	Berhasil memberikan <i>output</i> informasi obat dalam bahasa Indonesia.
3.	Memberikan <i>output</i> informasi apabila pengguna menginput nama obat.	Gemini AI tidak memberikan <i>output</i> apapun, melainkan muncul pesan "Gagal memuat info, inputan bukan nama obat".	Berhasil. Gemini AI tidak memberikan <i>output</i> , melainkan muncul pesan "Gagal memuat info, inputan bukan nama obat".
4.	Memberikan <i>output</i> informasi apabila pengguna menginput nama obat.	Gemini AI tidak memberikan <i>output</i> apapun, melainkan muncul pesan "Ketik nama obat dulu".	Berhasil. Gemini AI tidak memberikan <i>output</i> , melainkan muncul pesan "Ketik nama obat dulu".
5.	Navigasi dari halaman alarm ke pencarian informasi obat saat mengklik nama obat dari alarm yang diset dan dan Gemini AI menampilkan informasi nama obat, deskripsi, dosis, kandungan, indikasi, dan kontraindikasi.	Halaman alarm pindah ke pencarian informasi obat saat mengklik nama obat dari alarm yang diset dan dan Gemini AI menampilkan informasi nama obat, deskripsi, dosis, kandungan, indikasi, dan kontraindikasi.	Halaman alarm berhasil pindah ke pencarian informasi obat saat mengklik nama obat dari alarm yang diset dan Gemini AI berhasil menampilkan informasi nama obat, deskripsi, dosis, kandungan, indikasi, dan kontraindikasi.

Berdasarkan Tabel 1, seluruh skenario memberikan hasil pengujian sesuai yang diharapkan peneliti. Oleh karena itu, penelitian tentang pengembangan fitur pencarian informasi obat berbasis AI pada aplikasi Potio, berhasil dikembangkan menggunakan model Gemini AI. Dengan begitu, AI dapat terlihat perannya dalam menyelesaikan tugas yang kompleks secara lebih cepat, di mana peneliti tidak perlu lagi untuk menginput data informasi obat secara manual ke dalam *database* MySQL.

Meskipun kontribusi ilmiah dari penelitian ini adalah menghadirkan pendekatan baru dalam pemanfaatan Gemini AI untuk mengembangkan fitur pencarian informasi obat pada Potio yang merupakan aplikasi *mobile*, penelitian ini masih memiliki keterbatasan. Model

Gemini AI yang digunakan, yaitu Gemini 2.5 Flash hanya mendukung hasil *output* berupa teks.

4. KESIMPULAN

Berdasarkan hasil penelitian dan pengujian yang telah dilakukan, dapat disimpulkan bahwa :

1. Fitur pencarian informasi obat berbasis AI pada aplikasi Potio berhasil dikembangkan dengan memanfaatkan Gemini AI, sehingga peneliti tidak perlu lagi menginput informasi obat secara manual ke dalam *database* MySQL.
2. Hasil pengujian menggunakan *Blackbox Testing* menunjukkan bahwa aplikasi berhasil menampilkan informasi obat secara konsisten dalam bahasa Indonesia, mencakup nama obat, deskripsi, dosis, kandungan, indikasi, dan kontraindikasi.
3. Keterbatasan penelitian ini adalah model AI hanya mendukung hasil *output* berupa teks. Sebagai tindak lanjut, peneliti menyarankan agar ke depannya fitur ini dapat dikembangkan dengan melihat keterbatasan penelitian saat ini, seperti menggunakan model AI lain yang lebih akurat, relevan, dan mendukung berbagai inputan, misalnya AI yang bisa memberikan *output* dari inputan gambar produk obat dan memberikan *output* dari inputan suara pengguna.
4. Peneliti juga menyarankan untuk memperluas cakupan penggunaan aplikasi yang mendukung kebutuhan lembaga atau layanan kesehatan berskala besar, seperti rumah sakit, klinik, maupun apotek.

DAFTAR PUSTAKA

- [1] P. Pramana, C. W. Priastuty, P. Utari, R. A. Aziz, and E. Purwati, "Beradaptasi Dengan Perubahan Teknologi: Kecerdasan Buatan Dan Evolusi Komunikasi Interpersonal," *J. Ilm. Din. Sos.*, vol. 7, no. 2, pp. 214–225, 2023, doi: 10.38043/jids.v7i2.4909.
- [2] E. A. S. Mohamed, M. E. Osman, and B. A. Mohamed, "The Impact of Artificial Intelligence on Social Media Content," *J. Soc. Sci.*, vol. 20, no. 1, pp. 12–16, 2024, doi: 10.3844/jssp.2024.12.16.
- [3] S. Rifky, "Dampak Penggunaan Artificial Intelligence Bagi Pendidikan Tinggi," *Indones. J. Multidiscip. Soc. Technol.*, vol. 2, no. 1, pp. 37–42, 2024, doi: 10.31004/ijmst.v2i1.287.
- [4] S. Aprilia, R. Agustin, V. H. Pranatawijaya, and N. N. K. Sari, "Penerapan Api Gemini Dalam Layanan Peminjaman Novel Online Pada Website Cozybook," *J. Inform. dan Tek. Elektro Terap.*, vol. 12, no. 3, pp. 1929–1936, 2024, doi: 10.23960/jitet.v12i3.4508.
- [5] Siti Masrichah, "Ancaman Dan Peluang Artificial Intelligence (AI)," *Khatulistiwa J. Pendidik. dan Sos. Hum.*, vol. 3, no. 3, pp. 83–101, 2023, doi: 10.55606/khatulistiwa.v3i3.1860.
- [6] F. Morandín-Ahuerma, "What is Artificial Intelligence?," *Int. J. Res. Publ. Rev.*, vol. 3, no. 12, pp. 1947–1951, 2022, doi: 10.55248/gengpi.2022.31261.

- [7] P. M. Katikireddi and S. Jaini, "In Generative AI : Zero-Shot and Few-Shot," *Int. J. Sci. Res. Comput. Sci. Eng. Inf. Technol.*, vol. 8, no. 1, pp. 391–397, 2022, doi: <https://doi.org/10.32628/CSEIT2390668>.
- [8] A. Nazarius, F. Saputra, N. Noor Kamala sari, and V. Handrianus Pranatawijaya, "Penerapan Gemini AI Dalam Pembuatan Deskripsi Produk E-Commerce," *JATI (Jurnal Mhs. Tek. Inform.*, vol. 8, no. 3, pp. 3721–3725, 2024, doi: [10.36040/jati.v8i3.9670](https://doi.org/10.36040/jati.v8i3.9670).
- [9] N. Rachmat and D. P. Kesuma, "Implementasi Large Language Models Gemini Pada Pengembangan Aplikasi Chatbot Berbasis Android," *J. Ilmu Komput.*, vol. 4, no. 1, p. 2024, 2024, doi: [10.31314/juik.v4i1.2831](https://doi.org/10.31314/juik.v4i1.2831).
- [10] M. R. Langa and M. N. Moeti, "Enhancing Online Class Engagement through Generative AI: A Research using Microsoft Graph and Gemini APIs," *Int. Conf. Intell. Innov. Comput. Appl.*, no. November, pp. 67–78, 2024, doi: <https://doi.org/10.59200/ICONIC.2024.009>.
- [11] D. Triady, Izmy Alwiah Musdar, and Hendra Surasa, "Pengujian Blackbox Pada Website Worker'S Menggunakan Metode Equivalence Partitioning," *KHARISMA Tech*, vol. 18, no. 1, pp. 84–98, 2023, doi: [10.55645/kharismatech.v18i1.292](https://doi.org/10.55645/kharismatech.v18i1.292).