

Analisis Perbandingan Pengujian Respons Gemini AI dan Blackbox AI Menggunakan Metode Blackbox Testing

1st Muh.Avisenna Kusumadani

S1 Rekayasa Perangkat Lunak

Fakultas Informatika

avisennakusumadani@student.telkomuniversity.ac.id

Abstrak — Perkembangan Kecerdasan buatan (AI) sangat pesat yang menimbulkan berbagai tantangan dalam hal pengujian perangkat lunak, sebab hasil kecerdasan buatan(AI) harus dipastikan memiliki kualitas, akurasi dan keandalan atas hasil yang perintah yang diberikan khususnya pada aspek pemrograman dan perbaikan kode. Penelitian ini bertujuan untuk membandingkan antara Gemini AI dan Blackbox AI melalui metode pengujian Blackbox testing dan evaluasi pengguna menggunakan metode pendekatan Technology Acceptance Model (TAM). Pengujian yang dilakukan akan memastikan bahwa kedua AI memiliki keluaran terbaik untuk permasalahan pemrograman dan perbaikan kode. Pengujian dilakukan dengan lima skenario test case dengan pendekatan Software Testing Life Cycle (STLC) dan pengujian sendiri menggunakan Sonarqube sebagai alat bantu dalam analisis dan pengujian. Hasil pengujian Blackbox testing menunjukkan bahwa Blackbox AI lebih cocok digunakan sebab Blackbox AI, Blackbox AI memiliki potensi lebih unggul karena maintainability yang lebih baik dan jumlah code smell yang lebih sedikit, sehingga memudahkan proses perawatan dan pengembangan berkelanjutan. akan tetapi jika kebutuhan pengguna berdasarkan efisiensi dan keberhasilan menyelesaikan permasalahan secara cepat tanpa memerlukan penjelasan mendalam, maka Gemini AI sangat cocok untuk digunakan.

Kata kunci : Blackbox Testing, Gemini AI, Blackbox AI, SonarQube, Software Testing Life Cycle (STLC), Technology Acceptance Model (TAM).

I. PENDAHULUAN

Dalam beberapa tahun terakhir, perkembangan teknologi terutama kecerdasan buatan (AI) mengalami kemajuan pesat dan berdampak pada berbagai industri. Perkembangan ini dilandasi adanya pembaharuan teknologi setiap tahunnya. Di Indonesia sendiri, teknologi telah menyebar ke hampir seluruh penduduk. Pada tahun 2022 di Indonesia sendiri, hampir keseluruhan masyarakat menggunakan teknologi dalam kehidupan sehari-harinya. Sebanyak 277,7 juta orang

di Indonesia menggunakan smartphone, dengan 370,1 juta pengguna terkoneksi. Selain itu, ada 204,7 juta pengguna internet dan 191,4 juta pengguna media sosial aktif[1].

Data tersebut menunjukkan bahwa perkembangan teknologi di Indonesia setiap tahunnya mengalami kenaikan. Adanya peningkatan ini membawa tantangan baru terutama dalam Bidang Software Engineering. Seiring dengan pesatnya kemajuan teknologi, bagian pengujian adalah salah satu tantangan yang muncul dalam bidang rekayasa perangkat lunak. Hal ini dapat dilihat dari pesatnya perkembangan kecerdasan buatan (AI). Tantangan dalam pengujian perangkat lunak meningkat seiring dengan kemajuan teknologi, terutama dalam memastikan keandalan, akurasi, dan konsistensi sistem berbasis AI. Perkembangan AI yang semakin pesat disebabkan oleh sifat AI yang dinamis, tidak deterministik, dan mampu belajar dari data, sehingga hasil yang di dapat bisa berubah seiring waktu. Akibatnya, pengujian tidak hanya memastikan bahwa input sesuai dengan output tertentu tetapi juga harus mempertimbangkan hal-hal seperti bias data, kejelasan keputusan, dan kemungkinan perilaku yang tidak terduga muncul.

Penelitian ini akan membahas perkembangan AI dan bagaimana aplikasi AI dapat melakukan kegiatan manusia di bidang software engineering, khususnya mengembangkan aplikasi. Penelitian ini juga akan menguji apakah AI memberikan jawaban atas pertanyaan yang berkaitan dengan software engineering. Pengujian sendiri merupakan proses menganalisis elemen perangkat lunak dalam mengidentifikasi perbedaan kesalahan elemen perangkat lunak[2]. Menurut Roger S.Pressman Pengujian adalah serangkaian tindakan yang direncanakan dan dilakukan secara teratur[3]. Dengan melakukan pengujian, penguji dapat menyadari kualitas perangkat lunak dapat diketahui dari hasil pengujian yang dilakukan. Segala upaya dilakukan untuk mengukur dan memprediksi kesalahan yang mungkin terlewatkan jika tes telah lulus[4].

Pengujian dilakukan untuk memastikan bahwa input menghasilkan output yang berfungsi penuh atas sistem, sehingga sistem dianggap layak digunakan dan menciptakan kerangka kerja baru yang lebih sistematis. oleh karena itu, pengujian ini memberikan panduan kepada

pengembang dan penguji untuk merancang skenario pengujian yang lebih komprehensif dan relevan dengan kebutuhan pengguna. Penelitian yang dilakukan ini menggunakan metode pengujian Blackbox testing. Pengujian Blackbox akan memeriksa hasil aplikasi menggunakan data uji atau masukan untuk memastikan bahwa fungsionalitas aplikasi memenuhi persyaratan[5].

II. KAJIAN TEORI

A. Pengujian

Proses yang dikenal sebagai pengujian dilakukan untuk memastikan bahwa sistem berfungsi sesuai harapan dan menemukan kesalahan perangkat lunak[6]. Pengujian perangkat lunak juga dapat diartikan sebagai cara untuk mendapatkan informasi mengenai kualitas dari perangkat lunak yang sedang diuji[7]. Dalam pengembangan perangkat lunak, pengujian memegang kunci penting dalam menjamin keberhasilan pengembangan perangkat lunak dan kualitas produk. Pressman (2010) menyatakan bahwa ada tiga tujuan penting untuk pengujian perangkat lunak: (1) pengujian dilakukan dengan tujuan untuk menemukan kesalahan; (2) kesuksesan pengujian bergantung pada kemampuan untuk menemukan kesalahan yang belum ditemukan sebelumnya; dan (3) kasus uji yang baik memiliki kemungkinan yang tinggi untuk menemukan kesalahan yang belum ditemukan sebelumnya[8].

B. Gemini AI

Gemini merupakan AI yang dikembangkan oleh perusahaan Google dengan teknologi canggih hingga saat ini. Gemini awalnya diperkenalkan oleh Google DeepMind. Gemini merupakan model yang menggunakan multimoda sehingga dapat memproses dan menggabungkan berbagai jenis informasi, termasuk teks, kode, audio, gambar, dan video. Gemini diluncurkan pada tanggal 6 Desember 2023, Gemini merupakan lompatan maju yang signifikan dari pendahulunya, Bard, dan menandai komitmen Google untuk menciptakan AI yang bermanfaat bagi semua orang[9].

C. Blackbox AI

Blackbox AI adalah AI yang dirancang untuk membantu menulis kode dengan lebih cepat dan akurat[10]. Keunggulan dari Blackbox AI sendiri adalah membantu mengurangi waktu pengodean, karena memberi potongan kode dan pelengkapan otomatis yang relevan, terutama untuk pekerjaan berulang[11].

D. Software Testing Life Cycle (STLC)

Proses pengujian software Life Cycle (STLC) yang dilaksanakan secara sistematis dan terorganisir. STLC memiliki tahap yaitu, Kriteria Masuk (ketentuan yang harus dimiliki sebelum pengujian), Kriteria Keluar (ketentuan yang harus diselesaikan sebelum pengujian selesai), dan Aktivitas. STLC memiliki enam alur tahap, yaitu analisis persyaratan, perencanaan tes, pengembangan kasus, penetapan lingkungan, pelaksanaan tes, dan penutup[12].

E. Blackbox Testing

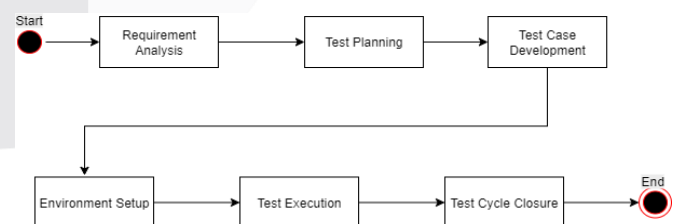
Blackbox testing adalah pengujian yang berfokus untuk mengetahui fungsional pada sebuah perangkat lunak dengan memberi masukan dan melihat keluaran apakah telah sesuai dengan masukan yang diberikan[13]. Proses pengujian Blackbox berdasarkan dari skenario yang telah disiapkan, skenario ini terdiri dari kasus uji, ekspektasi pengujian, dan hasil akhir pengujian[14].

F. Sonarqube

Sonarqube merupakan platform open source yang dikembangkan oleh perusahaan sonarSource. Aplikasi sonarqube digunakan sebagai alat inspeksi pada suatu kualitas kode aplikasi. sonarqube sendiri telah didukung dengan 30+ bahasa pemrograman yang terbagi ke dalam beberapa versi aplikasi, antara lain community edition sebanyak 17 bahasa pemrograman, developer edition sebanyak 24 bahasa pemrograman dan yang terakhir Enterprise dan Data Center Edition sebanyak 29 bahasa pemrograman[15].

III. METODE

GAMBAR 1
(Diagram Desain Perancangan Sistem)



Pertama, analisis kebutuhan dilakukan untuk memastikan apa yang dapat diuji dan mengidentifikasi risiko. Selanjutnya, Test planning untuk membuat rencana pengujian. Ini termasuk menentukan ruang lingkup, metode, tugas tim, jadwal, dan standar masuk dan keluar. Tahap ini menghasilkan dokumen yang berfungsi sebagai Rencana Tes. Setelah itu, membangun skenario dan prosedur uji sesuai dengan persyaratan, termasuk menyiapkan data uji dan hasil yang diharapkan, dan membuat skrip otomatisasi. Setelah test case sudah siap,

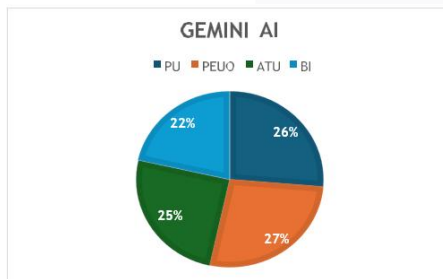
berlanjut ke pengaturan lingkungan; ini berarti menyiapkan lingkungan pengujian, untuk memastikan stabilitas. Tahap berikutnya adalah menjalankan pengujian, yang berarti menjalankan pengujian, mencatat hasil, dan melaporkan kesalahan atau kesalahan. Untuk memastikan bahwa perbaikan tidak menimbulkan masalah baru, proses ini juga melibatkan pengujian kembali. Terakhir, dilakukan dengan mengevaluasi hasil, membandingkannya dengan kriteria keluar, membuat laporan ringkasan pengujian, dan menyimpan catatan tentang pelajaran yang dipelajari untuk perbaikan di masa mendatang.

IV. HASIL DAN PEMBAHASAN

A. Test Execution Evaluasi Pengguna menggunakan Metode TAM

Evaluasi dilakukan oleh 20 Responden mahasiswa Telkom University khususnya Mahasiswa Rekayasa Perangkat Lunak yang sudah mengambil matakuliah Konstruksi Perangkat Lunak dari Angkatan 2021 hingga 2023. Perceived Usefulness (PU), Perceived Ease of Use (PEOU), Attitude toward Using (ATU), dan Behavioral Intention to Use (BI) adalah semua elemen yang terlibat dalam penilaian ini melalui kuesioner TAM. Tujuan Evaluasi ini untuk mendukung hasil dari pengujian Blackbox yang bersifat objektif dalam aspek teknis, sedangkan evaluasi TAM akan memberikan sudut pandang yang subjektif dari pengguna.

B. Visualisasi Hasil TAM (Gemini AI)



GAMBAR 1
(Visualisasi Hasil TAM Gemini AI)

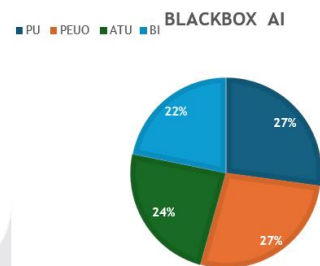
Hasil data diatas diambil dari nilai rata rata setiap pertanyaan yang berisi nilai Utama TAM. Nilai rata rata ini kemudian digambarkan menjadi bentuk diagram lingkaran untuk memudahkan visualisasi hasil evaluasi pengguna yang menggunakan metode TAM.

Dari data tersebut dibuktikan bahwa nilai PEOU memiliki nilai yang tertinggi dari aspek lainnya, yaitu sebesar 27%, hal ini menunjukkan bahwa pengguna merasa bahwa aspek PEOU pada Gemini AI khususnya dari segi User interface dan segi aksesibilitas sistem memiliki kemudahan yang dapat digunakan oleh pengguna. Selanjutnya untuk aspek tertinggi kedua setelah PEOU, yaitu PU. PU memiliki persentase

sebesar 26% yang berjarak 1% dari PEOU. Aspek ini menunjukkan adanya manfaat yang dirasakan secara langsung oleh pengguna dalam membantu memahami konsep pemrograman. Aspek ATU sendiri memiliki persentase sebesar 25% yang menjadikanya tertinggi ketiga dari aspek sebelumnya. Hal ini menunjukkan bahwa aspek ATU diterima baik oleh pengguna dalam penggunaan AI tersebut, meskipun belum sepenuhnya yakin tentang efektivitas AI tersebut dalam jangka panjang. Sementara itu, untuk BI menjadi nilai terendah dari beberapa nilai yang ada dengan presentase sebesar 22%. Nilai ini menunjukkan bahwa pengguna AI ini secara konsisten masih rendah, banyak faktor yang mempengaruhi misalnya kendala teknis selama proses penggunaan atau akses pengguna AI yang memiliki limit harian.

Secara keseluruhan, hasil ini memperlihatkan bahwa variabel PEOU menjadi variabel tertinggi di susul oleh PU dan ATU. Namun, untuk variabel BI masih menunjukkan presentase yang rendah, dimana hal ini menandakan bahwa meskipun pengguna telah merasakan manfaatnya, tidak semua dari pengguna secara otomatis memiliki niat untuk menggunakan AI secara berkelanjutan dan konsisten.

C. Visualisasi Hasil TAM (Blackbox AI)



GAMBAR 2
(Visualisasi Hasil TAM Blackbox AI)

Hasil data diatas diambil dari nilai rata rata setiap pertanyaan yang berisi nilai Utama TAM. Nilai rata rata ini kemudian digambarkan menjadi bentuk diagram lingkaran untuk memudahkan visualisasi hasil evaluasi pengguna yang menggunakan metode TAM.

Dari data tersebut dibuktikan bahwa nilai PU dan PEOU memiliki nilai yang sama sebesar 27%, hal ini menunjukkan bahwa pengguna merasa bahwa aspek PU memberikan manfaat dalam membantu memahami konsep pemrograman sedangkan untuk PEOU, nilai tersebut menunjukkan bahwa Blackbox AI mudah untuk dioperasikan baik dari segi User interface dan segi aksesibilitas sistem. Selanjutnya untuk aspek ATU sendiri memiliki persentase sebesar 24% yang menunjukkan

adanya penerimaan terhadap penggunaan AI ini, meskipun belum sepenuhnya yakin tentang efektivitas AI tersebut dalam jangka panjang. Sementara itu, untuk BI menjadi nilai terendah dari beberapa nilai yang ada dengan presentase sebesar 22%. Nilai ini menunjukkan bahwa pengguna AI ini secara konsisten masih rendah, banyak faktor yang mempengaruhi misalnya kendala teknis selama proses penggunaan atau akses pengguna AI yang memiliki limit harian.

Secara keseluruhan, hasil ini memperlihatkan bahwa variabel PU dan PEOU merupakan bagian yang paling dominan terhadap persepsi responden. Dengan kata lain, responden merasakan kebermanfaatan secara langsung dan kemudahan dalam penggunaan AI ini. Namun, untuk variabel ATU dan BI masih menunjukkan presentase yang rendah, dimana hal ini menandakan bahwa meskipun pengguna telah merasakan manfaatnya, tidak semua dari pengguna secara otomatis memiliki niat untuk

D. Perbandingan Hasil TAM Blackbox AI & Gemini AI

TABEL 1
(PERBANDINGAN HASIL TAM BLACKBOX AI & GEMINI AI)

Aspek TAM	Blackbox AI	Gemini AI	Keterangan
Perceived Usefulness (PU)	27 %	26%	Blackbox unggul,karena sistem Blackbox AI dinilai dapat menyelesaikan tugas dengan cepat menurut pengalaman dari pengguna baik secara waktu penyelesaian tugas hingga output kode yang dihasilkan dibandingkan menggunakan Gemini AI.
Perceived Easeof Use (PEUO)	27%	27%	Kedua AI sama , karena kedua AI tersebut dinilai mudah digunakan baik dari Blackbox AI maupun Gemini AI
Atitude Toward Using (ATU)	24%	25%	Gemini lebih unggul,karena menurut pengguna, gemini memberikan kejelasan hasil dengan menyertakan penjelasan logis terhadap kode program yang

			dihasilkan
Behavioral Intention to Use (BI)	22%	22%	Kedua AI seimbang,karena kedua AI dinilai menunjukkan tingkat penerimaan oleh pengguna yang tinggi .

Berdasarkan hasil tabel diatas, Blackbox AI menunjukkan keunggulan dibandingkan Gemini. Hal ini dapat dilihat melalui skor tertinggi pada variabel Perceived Usefulness (PU) sebesar 27 % dibandingkan Gemini AI yang hanya memperoleh 26%. Keunggulan PU sendiri di pengaruhi oleh sistem Blackbox AI yang dinilai dapat menyelesaikan tugas dengan cepat menurut pengalaman dari pengguna baik secara waktu penyelesaian tugas hingga output kode yang dihasilkan dibandingkan menggunakan Gemini AI. Gemini AI sendiri hanya unggul tipis pada variabel Attitude Toward Using (ATU) dengan skor 25%, dibandingkan Blackbox AI yang hanya memperoleh 24%.

E. Hasil Pengujian Blackbox Testing (Blackbox AI)

TABEL 2
(Hasil Pengujian Blackbox Testing (Blackbox AI))

Test Case ID	Expected result	Keterangan	Status Pengujian
TC-01	Kode menggunakan enum atau switch sesuai jumlah hari.	Kode program tidak memiliki Enum.	Failed
TC-02	Deklarasi dan pemanfaatan enum.	Kode Program mendeklarasikan variabel Enum.	Pased
TC-03	Kode dapat dimodifikasi dan di Run tanpa Bug.	Kode modifikasi menghasilkan Bug.	Failed
TC-04	Kode tidak memiliki bug saat dijalankan.	Kode memiliki Bug, ketika di test menggunakan SonarQube.	Failed
TC-05	Blackbox AI menjelaskan kesalahan dan memperbaiknya serta tidak menghasilkan program yang memiliki bug.	Blackbox memberikan solusi perbaikan, akan tetapi kode yang diperbaiki memiliki bug.	Failed

F. Hasil Pengujian Blackbox Testing (Gemini AI)

TABEL 3
(Hasil Pengujian Blackbox Testing (Gemini AI))

Test Case ID	Expected result	Keterangan	Status Pengujian
TC-01	Kode menggunakan enum atau switch sesuai jumlah hari.	Kode program tidak memiliki Enum.	Passed
TC-02	Deklarasi dan pemanfaatan enum.	Kode Program mendeklarasikan variabel Enum.	Pased
TC-03	Kode dapat dimodifikasi dan di Run tanpa Bug.	Kode modifikasi menghasilkan Bug.	Failed
TC-04	Kode tidak memiliki bug saat dijalankan.	Kode memiliki Bug, ketika di test menggunakan SonarQube.	Failed
TC-05	Gemini AI menjelaskan kesalahan dan memperbaikinya serta tidak menghasilkan program yang memiliki bug.	Blackbox memberikan solusi perbaikan, akan tetapi kode yang diperbaiki memiliki bug.	Failed

G. Keterkaitan Evaluasi TAM dan Hasil Pengujian Blackbox Testing

Metode Technology Acceptance Model (TAM) digunakan untuk melakukan evaluasi sistem, sedangkan metode pengujian Blackbox digunakan untuk melakukan pengujian sistem. Evaluasi menunjukkan bahwa terdapat perbedaan antara hasil berdasarkan persepsi pengguna dengan kinerja sistem. Berdasarkan hasil evaluasi TAM, Blackbox AI memperoleh nilai lebih tinggi dalam hal kemudahan penggunaan dan kegunaan, hal ini menandakan bahwa sistem Blackbox AI lebih disukai oleh pengguna dalam aspek antarmuka dan interaksi. Namun sebaliknya, berdasarkan analisis pengujian Blackbox Testing menggunakan SonarQube, Gemini AI menunjukkan hasil yang lebih unggul, terutama untuk konsistensi keluaran, efisien algoritma, serta kualitas struktur kode. dari hasil ini membuktikan bahwa meskipun Gemini AI memiliki tingkat penerimaan pengguna yang lebih rendah dibanding Blackbox AI, namun secara performa dan sistem, Gemini lebih akurat dan dapat diandalkan dalam waktu jangka panjang. Namun, berdasarkan hasil Blackbox Testing yang dikombinasikan dengan analisis teknis menggunakan

SonarQube, sistem Gemini AI menunjukkan hasil teknis yang lebih unggul, terutama dalam hal konsistensi keluaran, efisiensi algoritma, serta kualitas struktur kode. Hasil ini mengindikasikan bahwa meskipun sistem Gemini AI memiliki tingkat penerimaan pengguna yang lebih rendah dibanding Blackbox AI, namun secara teknis sistem ini lebih stabil, akurat, dan dapat diandalkan dalam jangka panjang.

H. Perbandingan Hasil Pengujian Blackbox AI dan Gemini AI

TABEL 4
(Perbandingan Hasil Pengujian Blackbox AI dan Gemini AI)

Aspek Pengujian	Blackbox AI	Gemini AI	Keterangan
Keberhasilan Test Case	1 Test case berhasil	2 Test Case berhasil	Gemini AI lebih unggul dari Blackbox AI, karena jumlah test yang berhasil lebih banyak, dimana ada 5 test case yang diujikan
Software Quality (Maintability dan Reliability)	3 temuan dan 5 temuan	6 temuan dan 4 temuan	Blackbox AI lebih unggul dari aspek maintability karena menghasilkan temuan yang sedikit, sedangkan Gemini lebih unggul dalam aspek Reliability karena sedikit temuan
Code smell	3 temuan	5 temuan	Blackbox AI lebih unggul dalam code smell karena kode memiliki struktur yang lebih bersih, mudah dipahami, dan lebih sederhana untuk dirawat.
Bug	5 temuan	5 temuan	Kedua AI memiliki bug yang sama.
Kemudahan Pengguna (TAM)	27%	26%	Blackbox AI lebih unggul karena pada variabel PU memberikan dasar yang kuat bahwa Blackbox AI lebih unggul secara keseluruhan dalam hal penerimaan pengguna berdasarkan model TAM.

Berdasarkan hasil Tabel Perbandingan Pengujian Blackbox AI dan Gemini AI menggunakan metode Blackbox Testing. Tabel tersebut menunjukkan bahwa Berdasarkan

hasil pengujian terhadap Blackbox AI dan Gemini AI pada lima aspek penilaian, diperoleh bahwa Gemini AI memiliki keunggulan pada aspek keberhasilan test case dengan menyelesaikan 2 dari 5 test case, sedangkan Blackbox AI hanya 1 test case saja. Pada aspek Software Quality, Blackbox AI unggul pada indikator Maintainability karena hanya 3 temuan dibandingkan Gemini AI yang memiliki 6 temuan. Sebaliknya, Gemini AI unggul pada Reliability dengan 4 temuan, lebih sedikit dibandingkan Blackbox AI.

Pada aspek Code Smell, Blackbox AI menunjukkan kualitas lebih baik dengan 3 temuan, sedangkan Gemini AI memiliki 5 temuan. Hal ini membuktikan bahwa kode Blackbox AI memiliki struktur yang lebih bersih, mudah dipahami, dan lebih sederhana untuk dirawat. Di aspek Bug, kedua AI menunjukkan hasil yang sama dengan masing-masing 5 temuan. Sedangkan, pada aspek Kemudahan Pengguna (TAM), Blackbox AI unggul dibandingkan Gemini AI, menunjukkan penerimaan pengguna terhadap Blackbox AI sedikit lebih tinggi.

Apabila dilihat secara perspektif jangka panjang, Blackbox AI memiliki potensi lebih unggul karena maintainability yang lebih baik dan jumlah code smell yang lebih sedikit, sehingga memudahkan proses perawatan dan pengembangan berkelanjutan. Meskipun Gemini AI unggul dalam keberhasilan eksekusi test case dan aspek reliability. Meskipun Gemini AI unggul dalam keberhasilan eksekusi test case dan aspek reliability, hal tersebut memiliki sifat berjangka pendek karena masalah reliabilitas relatif lebih mudah. Sementara itu, kelemahan pada maintainability dan code smell pada Gemini AI menyebabkan beban teknis dimasa depan.

V. KESIMPULAN

Dari Hasil pengujian yang telah dilakukan terhadap kedua AI, yaitu Gemini AI dan Blackbox AI. Dapat disimpulkan bahwa Blackbox AI memiliki potensi lebih unggul karena maintainability yang lebih baik dan jumlah code smell yang lebih sedikit, sehingga memudahkan proses perawatan dan pengembangan berkelanjutan. Meskipun Gemini AI unggul dalam keberhasilan eksekusi test case dan aspek reliability. Meskipun Gemini AI unggul dalam keberhasilan eksekusi test case dan aspek reliability, hal tersebut memiliki sifat berjangka pendek karena masalah reliabilitas relatif lebih mudah. Sementara itu, kelemahan pada maintainability dan code smell pada Gemini AI menyebabkan beban teknis dimasa depan.

Oleh karena itu secara keseluruhan, jika kebutuhan pengguna berdasarkan kinerja dan performa untuk jangka panjang, maka Blackbox AI lebih cocok digunakan sebab output yang dihasilkan lebih konsisten dan mudah untuk dirawat, akan tetapi jika kebutuhan pengguna berdasarkan efisiensi dan keberhasilan menyelesaikan permasalahan secara cepat tanpa memerlukan penjelasan mendalam, maka Gemini AI sangat cocok untuk digunakan..

REFERENSI

- [1] G. Santoso, J. Setiawan, and A. Sulaiman, "Development of OpenAI API Based Chatbot to Improve User Interaction on the JBMSantoso, G., Setiawan, J., & Sulaiman, A. (2023). Development of OpenAI API Based Chatbot to Improve User Interaction on the JBMS Website. G-Tech: Jurnal Teknologi Terapan, 7(4)," *G-Tech J. Teknol. Terap.*, vol. 7, no. 4, pp. 1606–1615, Oct. 2023, doi: 10.33379/gtech.v7i4.3301.
- [2] B. Huda and A. Lia Hananto, "Penerapan Software Testing Life Cycle Pada Pengujian Otomatisasi Platform Dzikra," *CSRID J.*, vol. 15, no. 1, pp. 1–11, 2023, [Online]. Available: <https://www.doi.org/10.22303/csr.id.15.1.2023.01-11>
- [3] M. F. Ahmadi and S. R. C. Nursari, "Pengujian Kondisi Struktur Kontrol pada Website (Studi Kasus: Erigo Official)," *J. Informatics Adv. Comput.*, vol. 3, no. 2, pp. 108–109, 2022.
- [4] M. Nurudin, W. Jayanti, R. D. Saputro, M. P. Saputra, and Y. Yulianti, "Pengujian Black Box pada Aplikasi Penjualan Berbasis Web Menggunakan Teknik Boundary Value Analysis," *J. Inform. Univ. Pamulang*, vol. 4, no. 4, p. 143, 2019, doi: 10.32493/informatika.v4i4.3841.
- [5] M. Mintarsih, "Pengujian Black Box Dengan Teknik Transition Pada Sistem Informasi Perpustakaan Berbasis Web Dengan Metode Waterfall Pada SMC Foundation," *J. Teknol. Dan Sist. Inf. Bisnis*, vol. 5, no. 1, pp. 33–35, 2023, doi: 10.47233/jteksis.v5i1.727.
- [6] Nabil Fahlevi Abdi and S.R. Candra Nursari, "Pengujian black box pada Website dengan Metode Robustness Testing (Studi kasus : Eiger Adventure)," *J. Informatics Adv. Comput.*, vol. 3, no. 2, pp. 93–96, 2022.
- [7] A. Maulana, A. Kurniawan, W. Keumala, V. R. Sukma, and A. Saifudin, "Pengujian Black Box pada Aplikasi Penjualan Berbasis Web Menggunakan Metode Equivalents Partitions (Studi Kasus: PT Arap Store)," *J. Teknol. Sist. Inf. dan Apl.*, vol. 3, no. 1, p. 50, 2020, doi: 10.32493/jtsi.v3i1.4307.
- [8] H. Sulistyanto, "Urgensi Pengujian Pada Kemajemukan Perangkat Lunak Dalam Multi Perspektif," *Komuniti J. Komun. dan Teknol. Inf.*, vol. 6, no. 1, pp. 65–74, 2017, [Online]. Available: <https://journals.ums.ac.id/index.php/komuniti/article/view/2944>
- [9] A. Pande, R. Patil, R. Mukkemwar, R. Panchal, and S. Bhoite, "Comprehensive Study of Google Gemini and Text Generating Models: Understanding Capabilities and Performance," *15th Int. Conf. Adv. Comput. Control. Telecommun. Technol. ACT 2024*, vol. 2, no. November, pp. 856–863, 2024.
- [10] I. Lasari and J. N. Utamajaya, "Evaluasi Efektivitas Blackbox . ai dalam Perkembangan Website Menggunakan Metode Systematic Literature Review," vol. 3, no. April, pp. 47–56, 2025.
- [11] Y. Qianyi, "Systematic Evaluation of AI-Generated Python Code : A Comparative Study across Progressive Programming Tasks," pp. 1–13, 2024.
- [12] I. R. Dhaifullah, M. Muttanifudin H, A. Ananda Salsabila, and M. Ainul Yaqin, "Survei Teknik Pengujian Software," *J. Autom. Comput. Inf. Syst.*, vol. 2, no. 1, pp. 31–38, 2022, doi: 10.47134/jacis.v2i1.42.

- [13] Y. Mad Cani and A. Ali Ridha, "Penguujian Black Box Testing Pada Sistem Pendukung Keputusan Penerima Beasiswa di SMK Tarbiyatul Ulum Karawang," *J. Ilm. Wahana Pendidik.*, vol. 9, no. 9, pp. 754–760, 2023, [Online]. Available: <https://doi.org/10.5281/zenodo.8084698>
- [14] A. Setiawan *et al.*, "Black Box Testing Dengan Teknik State Transition Testing Pada Inventori Alat-Alat Medis," *J. Sains dan Teknol.*, vol. 2, no. 2, pp. 151–158, 2022, doi: 10.47233/jsit.v2i3.218.
- [15] D. Novita and F. Helena, "Analisis Kepuasan Pengguna Aplikasi Traveloka Menggunakan Metode Technology Acceptance Model (TAM) Dan End-User Computing Satisfaction (EUCS)," *J. Teknol. Sist. Inf.*, vol. 2, no. 1, pp. 22–37, 2021, doi: 10.35957/jtsi.v2i1.846.

