Penerapan *Test-Driven Development* dalam Pembangunan *Backend REST* API Chatbot AI *Showroom* Kendaraan

Mochammad Aldi Priyanta
Sistem Informasi, Fakultas Teknologi Informasi, Universitas Merdeka Malang
e-mail: priyantaldi123@gmail.com[1]

Kata Kunci:

Showroom Mobil Bekas
Test-Driven Development
Cerebras AI
Express.js
System Usability Scale

ABSTRAK

Penelitian ini bertujuan untuk mengembangkan sistem informasi website Showroom Arjuna Motor Purbalingga yang terintegrasi dengan asisten virtual berbasis Cerebras AI, serta mengevaluasi keandalan *backend* dan tingkat keterpakaian (*usability*) sistem secara terukur. Metode pengembangan *backend* Express.js menerapkan pendekatan *Test-Driven Development* (TDD) menggunakan *framework* pengujian Jest dan basis data relasional SQLite, sedangkan antarmuka *frontend* dibangun menggunakan Vue.js 3 berbasis Vite. Pengujian sistem dilakukan secara komprehensif melalui pengujian terautomasi Jest, pengujian fungsionalitas *Black Box Testing*, serta pengujian *usability* menggunakan metode *System Usability Scale* (SUS) terhadap 15 responden. Hasil pengujian unit berbasis TDD menunjukkan tingkat keberhasilan 100% PASS dari 11 skenario pengujian pada Jest, dan pengujian *Black Box* menghasilkan status valid untuk seluruh fungsi utama sistem. Sementara itu, pengujian keterpakaian menggunakan SUS menghasilkan skor rata-rata sebesar 59,67 yang masuk dalam kategori Cukup (OK / Marginal Low). Berdasarkan hasil penelitian, penerapan TDD berhasil menjamin stabilitas dan keandalan fungsionalitas *backend*, sedangkan antarmuka sistem telah dapat beroperasi dengan baik namun membutuhkan penyempurnaan alur navigasi pada pengembangan selanjutnya.

Keyword:

Used Car Showroom
Test-Driven Development
Cerebras AI
Express.js
System Usability Scale

ABSTRACT

This study aims to develop the Arjuna Motor Purbalingga Showroom website integrated with a Cerebras AI-based virtual assistant, as well as to evaluate the backend reliability and system usability in a measurable manner. The Express.js backend development applies the Test-Driven Development (TDD) approach using the Jest testing framework and SQLite relational database, while the frontend interface is built using Vue.js 3 with Vite. System evaluation was conducted comprehensively through automated Jest testing, Black Box functionality testing, and usability testing using the System Usability Scale (SUS) method with 15 respondents. The TDD-based unit test results showed a 100% PASS success rate across 11 test scenarios in Jest, and Black Box testing yielded a valid status for all core system functions. Meanwhile, usability evaluation using SUS resulted in an average score of 59.67, falling into the OK / Marginal Low category. Based on the findings, TDD implementation successfully guaranteed backend functional stability, while the system interface operates effectively but requires navigation flow refinements in future developments.

Copyright © SENASIF 2026
ISSN: 2598-0076

I. PENDAHULUAN

Perkembangan teknologi informasi telah mendorong akselerasi transformasi digital pada berbagai sektor bisnis dan perdagangan secara fundamental [9]. Pada lanskap pemasaran modern, pengelolaan platform *website* memegang peranan krusial sebagai media publikasi, komunikasi, dan penyampaian informasi interaktif kepada calon konsumen [6]. Terlebih dalam industri niaga otomotif, penyampaian profil produk dan keterbukaan informasi unit secara digital menjadi strategi penting untuk menjangkau pasar yang lebih luas [7]. Namun, pada operasional showroom konvensional, proses konsultasi ketersediaan unit dan rincian spesifikasi kendaraan kerap terkendala oleh keterbatasan jam operasional staf pemasaran.

Untuk mengatasi kendala interaksi tersebut, adopsi teknologi *chatbot* berbasis kecerdasan buatan (*Artificial Intelligence* / AI) menjadi solusi yang efektif [1]. Perkembangan mutakhir pada model pemrosesan bahasa alami atau *Large Language Model* (LLM) memungkinkan asisten virtual memahami konteks percakapan pengguna secara luwes dan generatif [3]. Kendati demikian, penerapan model bahasa generatif pada skenario layanan tanya-jawab pelanggan (*customer support* dan FAQ) memiliki tantangan teknis terkait keakuratan informasi [4]. Tanpa batasan konteks data yang ketat, model AI rentan menghasilkan informasi yang tidak sesuai dengan stok riil kendaraan yang tersedia di showroom [4].

Oleh karena itu, keandalan lapisan arsitektur sisi belakang (*backend*) memegang peranan vital sebagai pengendali utama yang menghubungkan masukan pengguna, basis data inventaris, dan mesin inferensi AI. Pembangunan modul *backend* menuntut penerapan disiplin rekayasa perangkat lunak yang ketat [10]. Penelitian mengenai kematangan otomatisasi pengujian menunjukkan bahwa verifikasi kode program secara terautomasi mampu meningkatkan kualitas produk perangkat lunak dan menekan cacat logika secara signifikan [10]. Pendekatan *Test-Driven Development* (TDD) menjadi metodologi yang relevan untuk memastikan setiap *endpoint* API dan *guardrails* data teruji melalui skenario uji (*test-first*) sebelum diimplementasikan ke lingkungan produksi [10].

Di samping itu, performa inferensi AI sangat bergantung pada efisiensi infrastruktur komputasi yang digunakan. Pemanfaatan teknologi akselerasi *Wafer-Scale Integration* dari Cerebras AI menawarkan kecepatan pemrosesan inferensi bahasa alami yang tinggi untuk mendukung waktu respon percakapan yang instan [5].

Berdasarkan latar belakang tersebut, penelitian ini bertujuan untuk merancang dan membangun arsitektur *backend REST API* berbasis Express.js dan SQLite untuk website Showroom Arjuna Motor Purbalingga yang terintegrasi dengan Cerebras AI menggunakan pendekatan *Test-Driven Development* (TDD). Keandalan fungsionalitas sistem diuji melalui pengujian unit Jest terautomasi [10] dan *Black Box Testing*, serta dievaluasi tingkat keterpakaiannya oleh pengguna akhir menggunakan instrumen *System Usability Scale* (SUS) [2].

Copyright © SENASIF 2026
ISSN: 2598-0076

II. METODE

Penelitian ini menggunakan pendekatan kuantitatif dalam kerangka rekayasa perangkat lunak berbasis pengujian terautomasi [10]. Alur pengembangan *backend* REST API menerapkan siklus *Test-Driven Development* (TDD) yang terdiri dari tiga tahapan berulang:

1. *Fase Red*: Pengembang merancang skenario pengujian unit (*test suite*) menggunakan *framework* Jest dan pustaka Supertest untuk memvalidasi masukan *endpoint* sebelum kodenya dibuat, sehingga tes menghasilkan status gagal (*FAIL*).
2. *Fase Green*: Pengembang menuliskan kode pengendali (*controller*) Express.js dan kueri SQLite3 seminimal mungkin hingga pengujian Jest berhasil lolos (*PASS*).
3. *Fase Refactor*: Struktur kode dioptimasi (*clean code*) tanpa merusak fungsi yang telah lolos uji untuk menjaga pemeliharaan jangka panjang [10].

Arsitektur aplikasi web dibangun secara terpisah antara lapisan *frontend* Vue.js 3 berbasis Vite dan lapisan *backend* Express.js [8]. Layanan kecerdasan buatan memanfaatkan API Cerebras AI yang mengandalkan keunggulan kecepatan inferensi perangkat keras *Wafer-Scale* [5]. Untuk menjamin akurasi jawaban, *backend* mengimplementasikan metode penginjeksian data terverifikasi (*credible context injection*) ke dalam *system prompt* [11], di mana sistem hanya mengekstraksi data mobil dengan status aktif (is_available = 1) dari basis data SQLite sebelum meneruskan kueri ke Cerebras AI.

Evaluasi sistem dilakukan melalui tiga tingkatan: pengujian unit TDD Jest terautomasi [10], pengujian fungsional *Black Box Testing*, dan pengujian keterpakaian antarmuka (*usability testing*) terhadap 15 responden untuk mengukur kualitas layanan dan kepuasan pengguna [2].

Pengujian keterpakaian dianalisis menggunakan kuesioner *System Usability Scale* (SUS) dengan 10 butir instrumen berskala Likert 1–5. Skor kontribusi butir bernomor ganjil (P_1, P_3, P_5, P_7, P_9) dihitung menggunakan Persamaan (1):

$$R_i = X_i - 1 \tag{1}$$

Sedangkan untuk pernyataan bernomor genap ($P_2, P_4, P_6, P_8, P_{10}$), skor kontribusi dihitung menggunakan formula pada Persamaan (2):

$$R_i = 5 - X_i \tag{2}$$

Di mana R_i adalah skor kontribusi butir ke-i dan X_i adalah nilai jawaban responden pada butir ke-i. Skor SUS individual (S_j) untuk masing-masing responden ke-j diperoleh dari perkalian total skor kontribusi 10 instrumen dengan konstanta 2,5 sebagaimana dirumuskan pada Persamaan (3):

$$S_j = \left(\sum_{i=1}^{10} R_{ij}\right) \times 2{,}5 \tag{3}$$

Nilai rata-rata skor SUS keseluruhan ($\bar{X}_{\text{SUS}}$) dari total $N = 15$ responden dihitung menggunakan formula rata-rata hitung pada Persamaan (4):

Copyright © SENASIF 2026

ISSN: 2598-0076

$$\bar{X}_{\text{SUS}} = \frac{\sum_{j=1}^{N} S_j}{N} \quad (4)$$

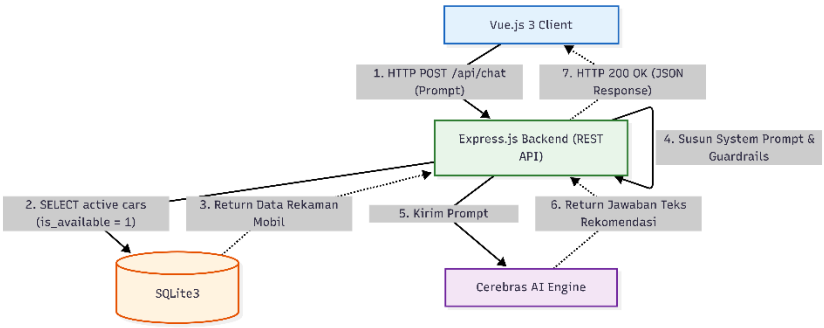
Hasil skor rata-rata SUS kemudian diinterpretasikan dan dipetakan ke dalam kriteria Acceptability Ranges, Grade Scale, dan Adjective Ratings berdasarkan standar evaluasi kepuasan pengguna [2].

III. HASIL DAN PEMBAHASAN

Pengembangan arsitektur *backend* dan integrasi sistem website Showroom Arjuna Motor Purbalingga menghasilkan evaluasi teknis serta analisis keterpakaian sebagai berikut.

Penerapan Siklus TDD pada Backend REST API

Pembangunan *endpoint* /api/chat menerapkan TDD secara konsisten. Pengujian unit Jest dirancang untuk memverifikasi validasi muatan (*payload*), kueri filter inventaris aktif SQLite, serta pemanggilan API Cerebras. Berkat pemanfaatan akselerasi inferensi Cerebras AI [5], respons teks rekomendasi dapat dihasilkan secara cepat dan akurat sesuai pembatasan konteks inventaris yang dikirimkan [4], [11].



Gambar 1. Alur Komunikasi Stateless Backend REST API dengan SQLite dan Cerebras AI

Hasil Pengujian Unit TDD dan Black Box Testing

Eksekusi pengujian unit otomatisasi Jest terhadap modul autentikasi JWT, manipulasi basis data inventaris, dan pemrosesan *prompt* AI disajikan pada Tabel 1.

Tabel 1. Rekapitulasi Hasil Pengujian Unit Terautomasi Berbasis TDD Menggunakan Jest

No	Modul	Skenario Pengujian	Hasil	Status
1.	Autentikasi Admin API	Verifikasi login valid, penolakan password salah, dan pengujian token autentikasi.	3 *passed*	Valid
2.	Kelola Mobil & Stok API	Validasi query tambah data, ubah detail, hapus unit serta toggle status stok pada SQLite	5 *passsed*	Valid
3.	Integrasi Prompt AI Service	Pengujian penyusunan system prompt, pencetakan konteks data stok aktif dan konektivitas REST API ke Cerebras API	3 *passed*	Valid
Total			11 *passed*	Valid

Copyright © SENASIF 2026
ISSN: 2598-0076

Tingkat kelulusan 100% *Passed* pada Tabel 1 membuktikan bahwa praktik penulisan kode berbasis TDD mampu menghasilkan struktur fungsi yang stabil dan bebas kesalahan logika mendasar [10]. Pengujian fungsionalitas sistem dari sisi pengguna dilakukan menggunakan *Black Box Testing* sebagaimana tercantum pada Tabel 2.

Tabel 2. Hasil Pengujian Fungsionalitas Sistem Melalui Black Box Testing

No	Fitur / Fungsi	Skenario Pengujian	***Input***	***Output***	**Hasil**	**Status**
1.	Autentikasi Admin	Login dengan password salah.	Username benar, Password salah.	Sistem menolak login dan menampilkan pesan peringatan.	Sesuai	Valid
2.	Autentikasi Admin	Login dengan data benar.	Username & Password benar.	Sistem memverifikasi kredensial dan mengarahkan ke Dashboard Admin.	Sesuai	Valid
3.	Kelola Mobil	Menambah data mobil baru.	Form spesifikasi mobil & unggah gambar.	Data tersimpan ke SQLite dan muncul di halaman katalog.	Sesuai	Valid
4.	Kelola Stok	Mengubah status ketersediaan.	Mengubah toggle dari tersedia ke terjual.	Status stok di SQLite terbarui dan unit terjual disembunyikan dari konteks AI.	Sesuai	Valid
5.	Katalog Customer	Filter & Pencarian Kendaraan	Memilih kategori Hatchback.	Menampilkan daftar kendaraan yang sesuai kriteria filter.	Sesuai	Valid
6.	ChatbotAI	Menanyakan stok kendaran.	Apakah Honda Brio 2020 masih ada?	AI merespon dengan megonfirmasi ketersediaan unit beserta harga resmi.	Sesuai	Valid
7.	Chatbot AI	Menanyakan unit di luar stok.	Apakah ada mobil Fortuner?	AI menjawab dengan sopan dan lemah lembut bahwa unit belum tersedia di stok showroom.	Sesuai	Valid

Seluruh fungsi yang diuji pada Tabel 2 beroperasi sesuai dengan ekspektasi rancangan tanpa adanya malfungsi fungsionalitas.

Evaluasi Keterpakaian Sistem (System Usability Scale)

Pengukuran keterpakaian (*usability*) dan kepuasan pengguna dilakukan melalui pengisian kuesioner SUS oleh 15 responden [2]. Matriks skor respon disajikan pada Tabel 3.

Copyright © SENASIF 2026

ISSN: 2598-0076

Tabel 3. Matriks Skor Respon 15 Responden pada Instrumen Kuesioner SUS

P	R1	R2	R3	R4	R5	R6	R7	R8	R9	R10	R11	R12	R13	R14	R15
1	3	5	4	5	3	2	4	3	5	3	3	3	3	3	3
2	3	1	2	1	1	3	3	4	2	3	2	3	3	3	3
3	4	5	4	5	3	4	3	2	5	3	3	3	3	3	3
4	2	1	2	1	2	2	1	3	4	3	3	3	3	3	3
5	4	5	5	5	5	3	2	3	4	3	3	3	3	3	3
6	4	5	5	5	2	4	4	4	3	3	3	3	3	3	3
7	2	2	1	1	3	2	3	2	5	3	2	3	4	3	4
8	2	1	1	2	4	3	3	5	2	3	2	3	2	2	2
9	4	5	5	5	4	4	2	5	3	5	3	3	4	4	4
10	2	3	1	3	2	3	2	3	2	2	3	3	2	3	3
Skor Mentah	24	31	28	29	27	20	21	16	29	23	21	20	24	22	23
Skor SUS Individual	60	77.5	70	72.5	67.5	50	52.5	40	72.5	57.5	52.5	50	60	55	57.5

Rincian nilai rata-rata tiap butir instrumen pertanyaan kuesioner SUS ditampilkan pada Tabel 4.

Tabel 4. Rata-Rata Skor Tanggapan Responden per Butir Pertanyaan SUS

NO	Pertanyaan	Rata-Rata Skor
1.	Saya ingin menggunakan aplikasi ini secara rutin	3.53
2.	Aplikasi ini terasa rumit	2.47
3.	Aplikasi ini mudah digunakan	3.53
4.	Saya butuh bantuan teknis untuk menggunakan aplikasi ini	2.4
5.	Fitur-fitur dalam aplikasi terintegrasi dengan baik	3.6
6.	Ada banyak ketidakkonsistenan dalam aplikasi	3.6
7.	Kebanyakan orang akan cepat belajar menggunakan aplikasi ini	2.67
8.	Aplikasi ini membingungkan	2.47
9.	Saya merasa percaya diri menggunakan aplikasi ini	3.93
10.	Saya perlu belajar banyak hal sebellum bisa menggunakan aplikasi	2.47

Copyright © SENASIF 2026
ISSN: 2598-0076

Berdasarkan Persamaan (4), rata-rata skor SUS keseluruhan dikalkulasikan sebagai berikut:

$$\bar{X}_{\text{SUS}} = \frac{60{,}0 + 77{,}5 + 70{,}0 + 72{,}5 + 67{,}5 + 50{,}0 + 52{,}5 + 40{,}0 + 72{,}5 + 57{,}5 + 52{,}5 + 50{,}0 + 60{,}0 + 55{,}0 + 57{,}5}{15}$$

$$= \frac{895{,}0}{15} = 59{,}67$$

Pembahasan Temuan Penelitian

Skor rata-rata SUS sebesar 59,67 menempatkan sistem pada kategori *Marginal Low* (*Grade Scale* D, predikat "Cukup / OK"). Nilai positif yang dominan muncul pada butir ke-9 (skor 3,93) dan ke-5 (skor 3,60), yang menandakan bahwa integrasi *chatbot* AI memberikan rasa percaya diri dan kemudahan dalam menelusuri katalog showroom.

Namun, perolehan nilai pada butir ke-6 (skor 3,60 pada pernyataan inkonsistensi) mengindikasikan adanya kendala ergonomis visual pada antarmuka *mobile*, seperti *widget chatbot* yang sempat menutupi sebagian tombol aksi katalog. Meskipun antarmuka masih membutuhkan perbaikan tata letak navigasi, arsitektur *backend* yang dibangun melalui metodologi TDD telah membuktikan keandalan logika 100% sehingga siap menopang perbaikan antarmuka pada siklus berikutnya [2], [10].

IV. SIMPULAN

Metode *Test-Driven Development* (TDD) menggunakan *framework* pengujian Jest berhasil diterapkan dalam pembangunan arsitektur *backend REST API* untuk website Showroom Arjuna Motor Purbalingga. Pendekatan TDD melalui tahapan *Red, Green,* dan *Refactor* berhasil menghasilkan kode *backend* yang stabil dan modular dengan tingkat kelulusan pengujian unit terautomasi sebesar 100% pada 11 skenario uji logika kritis, termasuk integrasi layanan inferensi Cerebras AI dan manipulasi basis data SQLite. Pengujian fungsionalitas menggunakan *Black Box Testing* mengonfirmasi bahwa seluruh fungsi antarmuka berjalan valid sesuai spesifikasi rancangan. Evaluasi keterpakaian menggunakan instrumen *System Usability Scale* menghasilkan nilai rata-rata sebesar 59,67 yang masuk dalam kategori Cukup (*OK / Marginal Low*), menunjukkan bahwa sistem dapat diterima dan berfungsi dengan baik, dengan saran pengembangan lanjutan yang berfokus pada penyempurnaan responsivitas tata letak antarmuka pengguna.

DAFTAR RUJUKAN

[1] E. Adamopoulou and L. Moussiades, “An Overview of Chatbot Technology,” in Artificial Intelligence Applications and Innovations, I. Maglogiannis, L. Iliadis, and E. Pimenidis, Eds., Cham: Springer International Publishing, 2020, pp. 373–383. doi: 10.1007/978-3-030-49186-4_31.

Copyright © SENASIF 2026

ISSN: 2598-0076

[2] A. Adellia and L. A. Abdillah, “Analisis Kualitas Layanan Website E-Commerce Bukalapak Terhadap Kepuasan Pengguna Mahasiswa Universitas Bina Darma Menggunakan Metode Webqual 4.0,” Journal of Software Engineering Ampera, vol. 1, no. 3, pp. 144–159, Oct. 2020, doi: 10.51519/journalsea.v1i3.52.

[3] S. K. Dam, C. S. Hong, Y. Qiao, and C. Zhang, “A Complete Survey on LLM-based AI Chatbots,” Nov. 18, 2024, arXiv: arXiv:2406.16937. doi: 10.48550/arXiv.2406.16937.

[4] F. Khennouche, Y. Elmir, N. Djebari, Y. Himeur, and A. Amira, “Revolutionizing Customer Interactions: Insights and Challenges in Deploying ChatGPT and Generative Chatbots for FAQs,” Nov. 16, 2023, arXiv: arXiv:2311.09976. doi: 10.48550/arXiv.2311.09976.

[5] Y. Kundu et al., “A Comparison of the Cerebras Wafer-Scale Integration Technology with Nvidia GPU-based Systems for Artificial Intelligence,” Mar. 11, 2025, arXiv: arXiv:2503.11698. doi: 10.48550/arXiv.2503.11698.

[6] M. Nuh, “PENYULUHAN MENGELOLA WEBSITE SEBAGAI MEDIA PUBLIKASI, KOMUNIKASI DAN INFORMASI PADA PESANTREN HIDAYATULLAH JONGGOL,” JURNAL PEDES - PENGABDIAN BIDANG DESAIN, vol. 2, no. 1, pp. 110–117, Sep. 2022.

[7] N. Palaniswamy, “Social Media Marketing (SMM) A Strategic Tool for Developing Business for Tourism Companies,” arXiv.org. Accessed: May 18, 2026. [Online]. Available: https://arxiv.org/abs/2107.03895v1

[8] G. Pantelakis, P. Papadopoulos, N. Kourtellis, and E. P. Markatos, “Measuring the (Over)use of Service Workers for In-Page Push Advertising Purposes,” vol. 13210, 2022, pp. 426–438. doi: 10.1007/978-3-030-98785-5_19.

[9] G. Vial, “Understanding digital transformation: A review and a research agenda,” in Managing Digital Transformation, Routledge, 2021.

[10] Y. Wang, M. Mäntylä, Z. Liu, and J. Markkula, “Test Automation Maturity Improves Product Quality -- Quantitative Study of Open Source Projects Using Continuous Integration,” Feb. 07, 2022, arXiv: arXiv:2202.04068. doi: 10.48550/arXiv.2202.04068.

[11] Z. Xiao, Q. V. Liao, M. X. Zhou, T. Grandison, and Y. Li, “Powering an AI Chatbot with Expert Sourcing to Support Credible Health Information Access,” Jan. 25, 2023, arXiv: arXiv:2301.10710. doi: 10.48550/arXiv.2301.10710.

Copyright © SENASIF 2026
ISSN: 2598-0076