

PERANCANGAN FITUR ANOTASI PADA CHATGPT MENGUNAKAN EKSTENSI *BROWSER*

Elza Satria Bhima Sakti¹, Dini Rohmayani²

^{1,2}Teknik Informatika, Politeknik TEDC Bandung

Email: bhima.learn@gmail.com¹, dinirohmayani@poltektedc.ac.id²

DOI: <https://doi.org/10.58217/ipsikom.v13i2.105>

ABSTRAK

The rapid advancement of large language models such as ChatGPT has revolutionized text-based interactions, yet lacks built-in annotation mechanisms for marking, storing, and revisiting critical conversation points. This study designs and implements a browser extension using ReactJS and Tailwind CSS with local storage via IndexedDB, following a five-stage waterfall model encompassing requirements analysis, system design, implementation, testing (black-box and user acceptance), and maintenance. The extension supports CRUD operations on annotations, automatic navigation to annotated text positions, and JSON import/export without disrupting ChatGPT's original interface. Testing demonstrates full compliance with functional specifications, while a UAT with 11 participants reports average satisfaction scores ≥ 4.55 for usability, UI coherence, and review efficiency, confirming enhanced productivity and learning experience. In conclusion, the extension effectively facilitates critical thinking and collaboration in web-based learning, with future work exploring cross-device synchronization, real-time collaboration, and AI-driven annotation suggestions to further optimize user engagement.

Keywords: ChatGPT; Anotasi Digital; Ekstensi Browser; IndexedDB; ReactJS;

PENDAHULUAN

Perkembangan pesat teknologi kecerdasan buatan (*artificial intelligence/AI*), khususnya dalam bidang pemrosesan bahasa alami (*natural language processing/NLP*), telah mengubah cara interaksi manusia dengan mesin. Berbagai platform berbasis teks kini mampu memahami konteks, menjawab pertanyaan, serta menghasilkan konten secara otomatis (Khurana et al., 2023). Salah satu terobosan terpenting adalah kemunculan model bahasa besar (*large language models/LLM*) seperti ChatGPT, yang dikembangkan oleh OpenAI dan telah menunjukkan kemampuan generatif yang unggul (Jim et al., 2024)

ChatGPT telah diadopsi luas dalam pendidikan, penelitian, dan produktivitas profesional. Dalam konteks akademik, mahasiswa dan peneliti memanfaatkannya untuk mencari referensi, menyusun naskah, hingga brainstorming ide karena kemampuannya memberikan respons yang relevan dan mudah dipahami (Jo, 2024; Mai et al., 2024). Namun, meskipun kaya fitur interaksi, ChatGPT masih belum menyediakan mekanisme anotasi bawaan yang memadai untuk menandai, menyimpan, dan

merefleksikan poin-poin penting sepanjang percakapan (Sumon et al., 2024).

Anotasi digital penambahan catatan, penandaan teks, dan penyusunan ulang potongan informasi adalah komponen krusial dalam mendukung berpikir kritis, pencarian informasi dan proses pembelajaran (Abbas et al., 2024). Dengan membuat catatan langsung pada materi yang sedang diakses, pengguna dapat lebih mudah mengingat, membandingkan, merefleksikan, dan menyusun ulang pengetahuan sesuai kebutuhan pribadi. Selain itu, anotasi juga berfungsi sebagai sarana untuk mengeksternalisasi pemikiran, sehingga mendukung kolaborasi, diskusi, dan pemecahan masalah secara lebih sistematis. Tanpa adanya fitur anotasi yang memadai, proses menyerap dan mengelola informasi bisa menjadi terputus-putus, tidak efisien, dan kurang mendalam. Oleh karena itu, integrasi kemampuan anotasi dalam platform digital semakin menjadi kebutuhan penting di zaman saat ini.

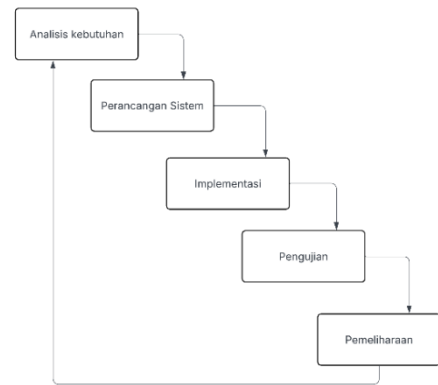
Penulis mengumpulkan data melalui kuesioner pada 10 Februari 2025 kepada

sejumlah pengguna aktif ChatGPT. Hasilnya menunjukkan bahwa, responden merasa membutuhkan penandaan dan penyimpanan bagian penting percakapan secara instan dan dilengkapi fitur tambahan lainnya seperti pengelompokan dalam kategori tertentu, peninjauan ulang, dan pencarian. Ketiadaan fitur anotasi langsung di antarmuka ChatGPT menyebabkan pengguna sering mencatat secara manual ke aplikasi lain.

Berdasarkan temuan tersebut, penelitian ini merancang ekstensi *browser* yang mengintegrasikan fitur anotasi langsung ke halaman ChatGPT. Tujuan utamanya adalah memudahkan pengguna untuk menandai, mencatat, menyimpan, dan meninjau kembali bagian percakapan penting tanpa meninggalkan antarmuka ChatGPT, sehingga mendukung produktivitas, dokumentasi, dan pengelolaan informasi yang lebih terstruktur.

METODOLOGI PENELITIAN

Penelitian ini menggunakan model pengembangan perangkat lunak *waterfall*, yang terdiri dari lima tahap terstruktur dan berurutan, dimulai dari analisis kebutuhan, perancangan sistem, pelaksanaan, pengujian, dan pemeliharaan. Model ini dipilih karena memberikan kejelasan tahapan dan dokumentasi yang sistematis, sehingga cocok untuk pengembangan sistem seperti ekstensi *browser* yang memiliki ruang lingkup dan tujuan yang cukup jelas dari awal (Abdul et al., 2024). Selain itu, model ini mendorong penyelesaian menyeluruh pada setiap tahap sebelum melanjutkan ke tahap berikutnya, meminimalkan kemungkinan perubahan mendadak yang dapat memengaruhi struktur sistem (Abdul et al., 2024).



Gambar 1. Metode *Waterfall*

1. Analisis Kebutuhan

Analisis kebutuhan adalah tahap awal dalam siklus hidup pengembangan perangkat lunak yang bertujuan menggali apa saja yang dibutuhkan pemangku kepentingan dan merumuskan spesifikasi fungsional sistem secara sistematis (Rosado da Cruz & Cruz, 2025)).

Tahap analisis kebutuhan, penulis mengumpulkan data lewat kuesioner kepada pengguna aktif ChatGPT untuk mengidentifikasi kesulitan mereka dalam menandai dan mengelola informasi penting selama percakapan. Temuan menunjukkan bahwa banyak pengguna terpaksa menyalin teks ke aplikasi lain karena aplikasi utama tidak menyediakan fitur yang memadai untuk mengelola atau memproses teks secara langsung. Hal ini menyebabkan pengalaman pengguna menjadi kurang praktis dan memperlambat alur kerja mereka. Dengan begitu kebutuhan akan fitur anotasi langsung di halaman menjadi krusial dalam meningkatkan pengalaman pengguna dalam konteks pencarian informasi dan pembelajaran berbasis *web*.

2. Perancangan Sistem

Perancangan sistem mencakup pembuatan model arsitektur perangkat lunak, alur logika, dan antarmuka pengguna guna menerjemahkan kebutuhan ke dalam desain teknis (Gooding et al., 2023)

Tahap perancangan sistem, penulis menetapkan struktur teknis dan logika

sistem. Disusun flowmap untuk memvisualisasikan alur kerja ekstensi, mulai dari interaksi awal pengguna hingga penyimpanan anotasi. Selain itu, dibuat *use case* diagram untuk menggambarkan peran pengguna dan hubungan mereka dengan fitur-fitur seperti menandai, menyimpan, mengedit, menghapus, mencatat, mencari, dan meninjau kembali. Perancangan juga mencakup struktur penyimpanan data menggunakan IndexedDB. Basis data dirancang dengan satu *object store* untuk menyimpan anotasi pengguna secara mandiri, sesuai rekomendasi praktik terbaik penyimpanan sisi klien.

3. Implementasi

Implementasi adalah proses menerjemahkan desain ke dalam kode dengan memilih teknologi dan strategi integrasi yang tepat untuk memastikan performa dan kompatibilitas terjaga (Jin et al., 2025)

Tahap implementasi, penulis memulai dengan pembangunan ekstensi yang berjalan pada mesin Blink, yaitu engine *browser* yang digunakan oleh Google Chrome, Microsoft Edge, dan Opera. Ekstensi ini dirancang agar langsung menyisipkan fungsionalitas anotasi ke halaman ChatGPT tanpa mengganggu fungsi asli ChatGPT. Ketika pengguna memilih bagian teks dari percakapan, ekstensi akan menampilkan tombol interaktif untuk menambahkan anotasi dan juga dapat digunakan untuk penampilan panel daftar anotasi.

Pengembangan antarmuka pengguna, React dipilih karena modularitas dan kemampuannya memperbaiki antarmuka secara efisien, Tailwind CSS digunakan untuk mempercepat pembuatan UI responsif tanpa menulis banyak CSS manual (Sekhar Emmanni, 2023).

Data anotasi disimpan menggunakan IndexedDB, yaitu sistem basis data berbasis objek di sisi klien (*browser*). IndexedDB dipilih karena mampu menyimpan data dalam jumlah besar secara lokal, mendukung operasi *create, read, update, delete* (CRUD), dan berjalan mandiri tanpa server eksternal. Dengan penyimpanan ini, data tetap konsisten sepanjang sesi *browser* dan menjaga privasi pengguna (Sinha, n.d.).

4. Pengujian

Tahapan pengujian dilakukan untuk menjamin bahwa sistem bekerja dengan baik dan sesuai kebutuhan pengguna. Dua metode digunakan:

- a. *Blackbox Testing* dilakukan tanpa melihat kode internal, hanya menguji keluaran dari setiap aksi pengguna untuk memastikan logika sesuai dengan desain (Zulkarnaini et al., 2023). Penulis menguji fitur-fitur seperti menandai, menyimpan, mengedit, menghapus, dan meninjau ulang anotasi, diuji secara sistematis berdasarkan skenario penggunaan. Hasil dari pengujian ini menunjukkan apakah logika sistem sesuai dengan yang dirancang dan apakah ada kesalahan fungsi yang muncul.
- b. *User Acceptance Testing* (UAT) dilakukan dengan melibatkan pengguna untuk menilai aspek pengalaman pengguna secara keseluruhan (Lobchuk et al., 2023). Penulis meminta pengguna untuk mencoba fitur secara langsung dan memberikan umpan balik mengenai fitur, desain, kepuasan pengguna dan nilai tambah dari fitur anotasi ini dalam melakukan kegiatan pencarian dan manajemen informasi.

5. Pemeliharaan

Pemeliharaan adalah kegiatan berkelanjutan untuk memastikan sistem tetap andal, aman, dan kompatibel dengan perubahan lingkungan atau kebutuhan pengguna (Lavazza et al., 2024).

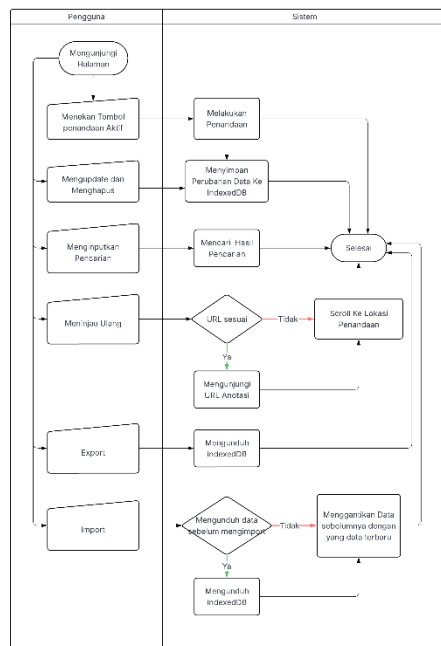
Tahap pemeliharaan, penulis melakukan pemantauan performa, responsivitas, serta kompatibilitas dengan perubahan pada struktur halaman ChatGPT atau pembaruan versi *browser*. Sistem akan diperbarui secara berkala berdasarkan umpan balik pengguna dan perubahan teknis pada *platform* yang digunakan. Perbaikan *bug* dan peningkatan fitur dilakukan secara iteratif, dengan tujuan menjaga keberfungsian dan relevansi ekstensi dalam jangka panjang.

HASIL DAN PEMBAHASAN

Bagian ini menyajikan hasil implementasi fitur anotasi pada ChatGPT melalui ekstensi *browser*, yang dirancang untuk menjawab kebutuhan pengguna berdasarkan hasil kuesioner dan analisis awal. Fitur ini difokuskan pada kemudahan dalam menandai, mencatat, dan mengelola informasi penting tanpa meninggalkan halaman. Pembahasan mencakup perancangan sistem, implementasi, serta pengujian.

1. Perancangan Sistem

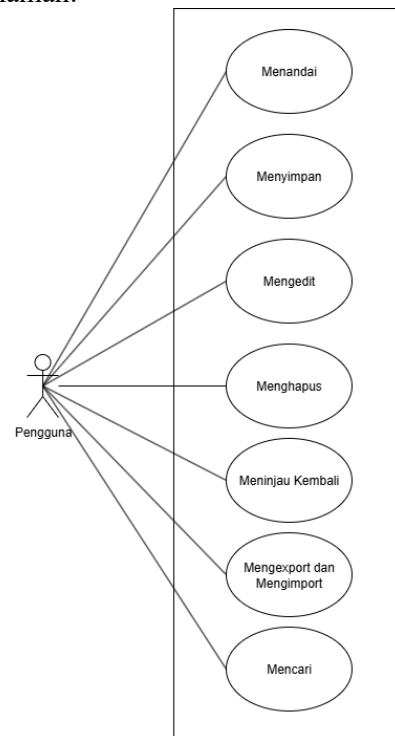
Hasil dari tahap perancangan sistem direpresentasikan melalui pembuatan flowmap sistem, diagram *use case*, dan struktur tabel basis data objek.



Gambar 2. Flowmap fitur anotasi

Flowmap pada Gambar 2 menunjukkan alur kerja sistem. Proses dimulai saat pengguna membuka halaman, lalu sistem secara otomatis menyuntikkan UI *overlay* ke dalam DOM percakapan. Setelah itu, data dimuat dari IndexedDB, dan penandaan diterapkan ke DOM sesuai dengan URL yang cocok. Pengguna dapat memulai proses penandaan dengan menekan tombol khusus. Tindakan ini akan memicu penandaan, menyimpan perubahan ke IndexedDB, serta memperbarui atau menghapus data yang diperlukan. Sistem juga menyediakan fitur

pencarian untuk memudahkan pengguna mencari dan melihat hasil penandaan. Selain itu, tersedia fungsi ekspor dan impor untuk menyimpan atau memuat data. Saat data diimpor, sistem akan menggantikan data lama dengan data terbaru. Terakhir, ada fitur peninjauan ulang, jika digunakan pengguna akan meninjau bagian yang ditandai secara otomatis tidak ada batasan apabila anotasi didalam halaman atau diluar halaman.



Gambar 3. Use case fitur anotasi

Use case pada gambar 3 menunjukkan aktor utamanya adalah pengguna yang berinteraksi langsung dengan tujuh fungsi inti, dimulai dari menandai, menyimpan, mengedit, menghapus, meninjau kembali, mengekspor/mengimpor, dan mencari anotasi. Diagram ini merepresentasikan kebutuhan utama pengguna dalam mencatat dan mengelola informasi penting selama sesi percakapan.

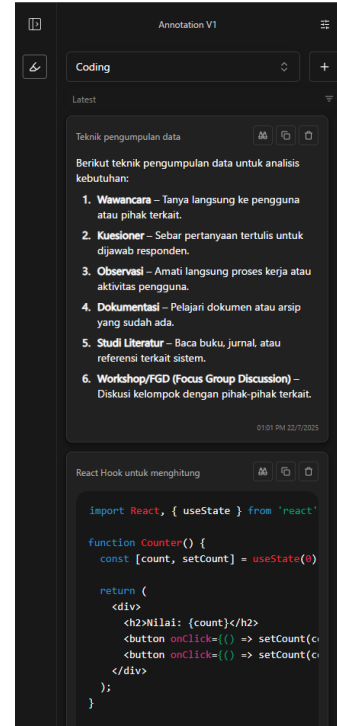
Tabel 1. Struktur basis data objek anotasi

Key	Tipe Data	Deskripsi
id	String	Id unik untuk setiap anotasi
label	String	Anotasi/Catatan/Label yang diberikan oleh pengguna.
url	String	Alamat halaman ChatGPT tempat anotasi dibuat.
snapshot	String	Cuplikan HTML dari jawaban ChatGPT yang digunakan untuk pemformatan
times tamp	Date/String	Waktu anotasi dibuat

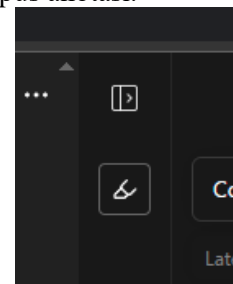
Struktur basis data objek anotasi pada tabel 1 menunjukkan implementasi menggunakan IndexedDB, dengan struktur data berbentuk objek JSON yang terorganisasi dalam sistem *folder*. Setiap *folder* memiliki kumpulan anotasi yang disimpan dalam array, di mana setiap anotasi mencakup informasi id, label, url, *snapshot*, *timestamp*, dan *position*. Struktur ini dirancang untuk memungkinkan pencatatan yang fleksibel, menyimpan beserta dengan pemformatannya, dan pencarian yang efisien terhadap data yang telah ditandai.

2. Implementasi

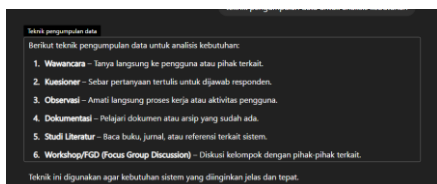
Antarmuka pengguna (*user interface/UI*) dirancang menggunakan ReactJS untuk menciptakan komponen dinamis dan Tailwind CSS untuk *styling* yang efisien. Tampilan antarmuka dibangun secara ringan, fungsional, dan tidak mengganggu tampilan asli ChatGPT.

**Gambar 4.** Tampilan panel anotasi

Tampilan Panel anotasi pada Gambar 4 merupakan komponen utama antarmuka yang menampilkan daftar catatan yang telah disimpan oleh pengguna. Panel ini dirancang agar muncul di sisi kanan layar tanpa mengganggu tampilan utama ChatGPT. Setiap entri/anotasi menampilkan label, kutipan konten yang ditampilkan sesuai dengan format ChatGPT, serta tombol peninjauan kembali anotasi yang jika ditekan akan melakukan *scroll* secara otomatis ke posisi anotasi yang terdapat pada halaman dan apabila anotasi yang tersimpan diluar halaman maka akan dilakukan peninjauan ke alamat URL atau alamat halaman tersebut kemudian *scroll* otomatis, terdapat salin teks anotasi untuk menyimpannya ke *clipboard*, dan juga terdapat hapus anotasi.

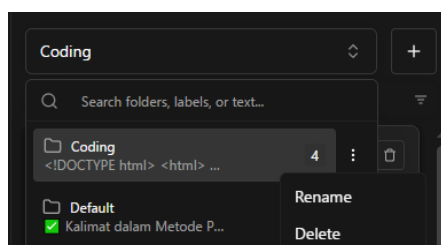
**Gambar 5.** Tampilan tombol

Tampilan Tombol pada Gambar 5 terdiri dari dua tombol, tombol pembuka daftar anotasi yang berfungsi untuk menampilkan panel anotasi dan tombol pengaktifan anotasi memungkinkan pengguna untuk mengaktifkan atau menonaktifkan mode anotasi. Kedua tombol ini diletakkan secara minimalis di pojok kiri panel untuk menjaga estetika antarmuka.



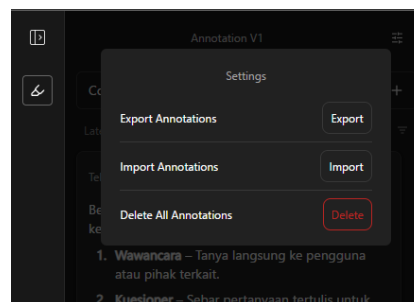
Gambar 6. Tampilan penandaan

Tampilan penandaan pada Gambar 6 menampilkan sebuah kotak yang berisikan bagian penting dari percakapan yang telah ditandai dan dibagikan kiri atas kotak terdapat catatan atau anotasi. Penandaan ini otomatis akan muncul saat pengguna berada pada halaman yang terdapat penandaan.



Gambar 7. Tampilan Sistem penyimpanan

Tampilan sistem penyimpanan pada Gambar 7 ditampilkan dalam bentuk manajemen data dengan sistem *folder*, yang memungkinkan pengguna menambahkan, mengubah nama, dan menghapus *folder* anotasi. Setiap *folder* berisi daftar anotasi yang tersimpan secara lokal menggunakan IndexedDB. Untuk membantu menemukan anotasi tertentu, sistem dilengkapi fitur pencarian berdasarkan kata kunci yang dicocokkan dengan nama *folder*, label, atau isi anotasi. Fitur ini mempermudah pengguna dalam meninjau kembali informasi penting yang telah disimpan, tanpa perlu membuka seluruh daftar anotasi secara manual.



Gambar 8. Tampilan pengaturan

Tampilan pengaturan pada Gambar 8 menyediakan tiga fungsi utama, yaitu *import*, *export*, dan *delete all*. Fitur *export* memungkinkan pengguna mengunduh seluruh data anotasi dalam format *file* JSON sebagai cadangan atau untuk dipindahkan ke perangkat lain. Fitur *import* digunakan untuk memuat kembali data anotasi yang berasal dari *file* yang telah diekspor sebelumnya. Sedangkan *delete all* berfungsi untuk menghapus seluruh data anotasi secara permanen dari sistem. Ketiga opsi ini dirancang sederhana dan mudah diakses agar pengguna dapat mengelola datanya secara mandiri, fleksibel, dan aman.

3. Pengujian

Pengujian dilakukan untuk memastikan fitur anotasi bekerja sesuai kebutuhan pengguna dan bebas dari kesalahan fungsi. Hasil pengujian dibagi menjadi dua metode yaitu, *Blackbox Testing* dan *User Acceptance Testing* (UAT).

Tabel 2. Hasil pengujian *blackbox*

No	Skenario	Hasil yang Diharapkan	Hasil Pengujian
1	Buka panel anotasi atau menekan tombol aksi (<i>scroll</i> /salin/hapus)	Panel tampil di sebelah kanan, daftar anotasi muncul, aksi tombol dapat dijalankan	Berhasil dengan Gambar 4
2	Tekan tombol munculkan panel atau aktif/nonaktifkan	Panel terbuka atau mode aktif/nonaktif sesuai tombol	Berhasil dengan Gambar 5
3	Pengguna buka halaman yang teranotasi	Kotak anotasi otomatis muncul dengan label dan bagian ditandai	Berhasil dengan Gambar 6
4	Pengguna ubah/hapus/tambah <i>folder</i>	Perubahan <i>Folder</i> diperbarui atau hasil pencarian	Berhasil dengan Gambar 7

	atau cari anotasi	tampil sesuai kata kunci	
5	Pengguna tekan <i>import/export/delete all</i> di pengaturan	Data diekspor (JSON), diimpor, atau semua anotasi terhapus sesuai aksi	Berhasil dengan Gambar 8

Hasil pengujian pada Tabel 2 menunjukkan bahwa seluruh fungsi utama pada fitur anotasi ekstensi ChatGPT berjalan sesuai dengan yang diharapkan. Setiap skenario yang diuji, mulai dari pemunculan panel anotasi, navigasi otomatis ke posisi anotasi, penyalinan teks, penghapusan catatan, penandaan otomatis saat halaman dikunjungi kembali, hingga fitur ekspor dan impor data anotasi, berhasil dijalankan tanpa ditemukan kesalahan fungsi. Hal ini mengindikasikan bahwa implementasi antarmuka serta manajemen data lokal telah sesuai dengan perancangan dan mampu mendukung pengalaman pengguna secara optimal.

Tabel 3. Hasil pengujian User Acceptance Testing (UAT)

No.	Aspek yang Dinilai	Rata-rata
1	Familiaritas dan kemudahan fitur menandai, mencatat, dan menyimpan	4,73
2	Efektivitas peninjauan otomatis dalam menghemat waktu dan energi	4,73
3	Kesesuaian desain UI dengan tampilan asli ChatGPT	4,73
4	Kesesuaian pemformatan anotasi (warna, gaya teks, markdown, dll.) dengan gaya ChatGPT	4,64
5	Kejelasan navigasi dalam aplikasi	4,64
6	Efektivitas fitur dalam mendukung aktivitas dokumentasi	4,55
7	Potensi peningkatan produktivitas dari penggunaan fitur	4,73

Hasil *User Acceptance Test* (UAT) pada Tabel 3 menunjukkan bahwa fitur anotasi pada ekstensi ChatGPT mendapatkan respons sangat positif dari 11 responden. Penilaian dilakukan menggunakan skala Likert 1–5, di mana skor 1 berarti sangat tidak setuju dan skor 5 berarti sangat setuju. Aspek dengan nilai rata-

rata tertinggi 4,73 mencakup kemudahan penggunaan fitur anotasi, efisiensi peninjauan otomatis, serta kesesuaian desain UI dengan tampilan asli ChatGPT. Aspek lain seperti pemformatan anotasi dan navigasi aplikasi juga mendapat skor tinggi 4,64, sedangkan efektivitas dalam mendukung dokumentasi dan peningkatan produktivitas masing-masing mencatat skor rata-rata di atas 4,5.

KESIMPULAN

Berdasarkan hasil implementasi dan pengujian, ekstensi browser yang mengintegrasikan fitur anotasi langsung pada antarmuka ChatGPT berhasil menjawab kebutuhan pengguna akan mekanisme penandaan, penyimpanan, dan peninjauan kembali poin-poin penting tanpa meninggalkan jendela percakapan. Melalui desain sistem berbasis IndexedDB dan antarmuka modular menggunakan ReactJS serta Tailwind CSS, pengguna dapat melakukan operasi CRUD pada anotasi secara instan, meninjau kembali ke konteks asli, serta mengeksport atau mengimpor data anotasi dalam format JSON. Uji *Blackbox* menunjukkan bahwa semua fungsi mulai dari penandaan teks, pemunculan panel anotasi, hingga peninjauan otomatis berjalan sesuai rencana tanpa gangguan pada tampilan utama ChatGPT. Sementara itu, UAT dengan 11 responden memberikan skor kepuasan tinggi (rata-rata $\geq 4,55$) pada aspek kemudahan penggunaan, keselarasan UI, dan efisiensi peninjauan ulang, menegaskan bahwa fitur ini secara signifikan meningkatkan produktivitas dan pengalaman belajar berbasis web.

Lebih lanjut, penelitian ini menekankan pentingnya anotasi digital dalam mendukung produktivitas, dengan ekstensi sebagai solusi praktis untuk mengatasi keterbatasan antarmuka ChatGPT. Integrasi fitur anotasi langsung ini tidak hanya mempermudah dokumentasi percakapan, tetapi juga membuka peluang pengembangan lanjutan seperti sinkronisasi lintas perangkat, kolaborasi *real-time*, dan analitik penggunaan anotasi. Ke depan, penelitian lanjutan dapat mengeksplorasi adopsi model proses pengembangan agile untuk merespons kebutuhan pengguna

secara lebih dinamis serta integrasi kecerdasan buatan untuk merekomendasikan label atau kategori anotasi secara otomatis.

DAFTAR PUSTAKA

- Abbas, M., Jam, F. A., & Khan, T. I. (2024). Is it harmful or helpful? Examining the causes and consequences of generative AI usage among university students. *International Journal of Educational Technology in Higher Education*, 21(1). <https://doi.org/10.1186/s41239-024-00444-7>
- Abdul, R., Khan, H., Gopal, P. R., Yogesh, P. B., Nitin, P. V., Sahebrao, K. S., Sharad, P. N., & Santosh, P. V. (2024). The Waterfall Model-Software Engineering. In *International Journal of Research Publication and Reviews Journal homepage: www.ijrpr.com* (Vol. 5). www.ijrpr.com
- Gooding, S., Werner, L., C, V., & Research, G. (2023). *A Pilot Study on Annotation Interfaces for Summary Comparisons*. <https://www.anthropic.com/>
- Jim, J. R., Talukder, M. A. R., Malakar, P., Kabir, M. M., Nur, K., & Mridha, M. F. (2024). Recent advancements and challenges of NLP-based sentiment analysis: A state-of-the-art review. *Natural Language Processing Journal*, 6, 100059. <https://doi.org/10.1016/j.nlp.2024.100059>
- Jin, B., Li, H., & Zou, Y. (2025). Impact of extensions on browser performance: An empirical study on google chrome. *Empirical Software Engineering*, 30(4), 103. <https://doi.org/10.1007/s10664-025-10633-1>
- Jo, H. (2024). From concerns to benefits: a comprehensive study of ChatGPT usage in education. *International Journal of Educational Technology in Higher Education*, 21(1). <https://doi.org/10.1186/s41239-024-00471-4>
- Khurana, D., Koli, A., Khatter, K., & Singh, S. (2023). Natural language processing: state of the art, current trends and challenges. *Multimedia Tools and Applications*, 82(3), 3713–3744. <https://doi.org/10.1007/s11042-022-13428-4>
- Lavazza, L., Locoro, A., & Meli, R. (2024). Software Development and Maintenance Effort Estimation Using Function Points and Simpler Functional Measures. *Software*, 3(4), 442–472. <https://doi.org/10.3390/software3040022>
- Lobchuk, M., Hoplock, L., Harder, N., Friesen, M., Rempel, J., & Bathi, P. R. (2023). Usability Testing of a Web-Based Empathy Training Portal: Mixed Methods Study. *JMIR Formative Research*, 7. <https://doi.org/10.2196/41222>
- Mai, D. T. T., Da, C. Van, & Hanh, N. Van. (2024). The use of ChatGPT in teaching and learning: a systematic review through SWOT analysis approach. In *Frontiers in Education* (Vol. 9). Frontiers Media SA. <https://doi.org/10.3389/feduc.2024.1328769>
- Rosado da Cruz, A. M., & Cruz, E. F. (2025). Machine Learning Techniques for Requirements Engineering: A Comprehensive Literature Review. *Software*, 4(3), 14. <https://doi.org/10.3390/software4030014>
- Sekhar Emmanni, P. (2023). Comparative Analysis of Angular, React, and Vue.js in Single Page Application Development. *International Journal of Science and Research (IJSR)*, 12(6), 2971–2974. <https://doi.org/10.21275/sr24401230015>
- Sinha, A. R. (n.d.). *Navigating Client-Side Storage in Modern Web Applications: Mechanisms, Best Practices, and Future Directions*. www.ijfmr.com
- Sumon, R. I., Uddin, S. M. I., Akter, S., Mozumder, M. A. I., Khan, M. O., &

Kim, H. C. (2024). Natural Language Processing Influence on Digital Socialization and Linguistic Interactions in the Integration of the Metaverse in Regular Social Life. In *Electronics (Switzerland)* (Vol. 13, Issue 7). Multidisciplinary Digital Publishing Institute (MDPI).

<https://doi.org/10.3390/electronics13071331>

Zulkarnaini, Firdhayanti, A., Taufik, T., & Bachry, B. (2023). User Acceptance Testing through Blackbox Evaluation for Corn Distribution Information System. *Bit-Tech*, 6(2), 208–215. <https://doi.org/10.32877/bt.v6i2.1065>