



Implementasi Struktur Data *Stack* pada Sistem *Recently Played Music Streaming* Berbasis Prinsip Lifo

Tia Risky Yasmin Saketang¹, Gus Rosauli Pandiangan², Frengky Alfredo Matondang³, Rizki Imanuell Situmorang⁴

^{1,2,3,4} Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Negeri Medan

E-mail: tyasmin488@gmail.com¹, rosapandiangan71@gmail.com², fr4nkelblue.689@gmail.com³, rizkicrew223@gmail.com⁴

Article Info

Article history:

Received September 25, 2025

Revised September 29, 2025

Accepted October 02, 2025

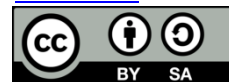
Keywords:

Stack Data Structure, LIFO Principle, Recently Played, Music Streaming, User Experience

ABSTRACT

The evolution of music streaming platforms demands efficient data management for the playback history ('Recently Played') feature to enhance the user experience. This study aims to implement a stack data structure based on the LIFO (Last-In-First-Out) principle in a 'Recently Played' system to create an optimal solution for managing historical data. Using a design and build research methodology, a prototype music player application was developed with a client-side architecture, implementing a stack for the history feature and a Doubly Linked List for the playlist. The results demonstrate the successful implementation of a functional application where the stack-based "Back to Last Played" feature operates responsively and efficiently. The study concludes that the application of the stack data structure is a superior approach for managing dynamic historical data, significantly improving the performance, efficiency, and user experience in music streaming applications.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



Article Info

Article history:

Received September 25, 2025

Revised September 29, 2025

Accepted October 02, 2025

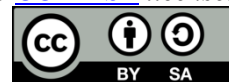
Kata Kunci :

Struktur Data Tumpukan, Prinsip LIFO, Riwayat Pemutaran, Streaming Musik, Pengalaman Pengguna

ABSTRAK

Perkembangan platform *streaming* musik menuntut manajemen data yang efisien untuk fitur riwayat pemutaran (*Recently Played*) guna meningkatkan pengalaman pengguna. Penelitian ini bertujuan untuk mengimplementasikan struktur data *stack* berbasis prinsip LIFO (*Last-In-First-Out*) pada sistem *Recently Played* untuk menciptakan solusi pengelolaan data historis yang optimal. Menggunakan metode penelitian rancang bangun, sebuah prototipe aplikasi pemutar musik dikembangkan dengan arsitektur sisi klien, mengimplementasikan *stack* untuk fitur riwayat dan *Doubly Linked List* untuk daftar putar. Hasil penelitian menunjukkan keberhasilan implementasi sebuah aplikasi fungsional di mana fitur "Back to Last Played" yang berbasis *stack* bekerja secara responsif dan efisien. Kesimpulan dari penelitian ini menegaskan bahwa penerapan struktur data *stack* merupakan pendekatan yang superior untuk mengelola data historis dinamis, yang secara signifikan meningkatkan kinerja, efisiensi, dan pengalaman pengguna pada aplikasi *streaming* musik.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



Corresponding Author:

Tia Risky Yasmin Saketang
Universitas Negeri Medan

E-mail: tyasmin488@gmail.com



PENDAHULUAN

Kemajuan teknologi digital telah secara mendalam mengubah drastis industri musik, mengubah pola konsumsi musik dari format fisik ke format digital melalui layanan streaming. Platform streaming musik seperti Spotify kini menjadi elemen penting dalam kehidupan sehari-hari generasi milenial dan Gen-Z dalam mengakses berbagai konten musik (Netti & Irwansyah, 2020). Perkembangan pesat layanan ini telah menciptakan kebutuhan akan sistem manajemen data yang efektif, terutama dalam mengelola riwayat pemutaran pengguna (Ruddin et al., 2022). Fitur “Recently Played” merupakan salah satu elemen penting dalam pengalaman pengguna aplikasi streaming musik. Penilaian pengalaman pengguna pada aplikasi semacam ini menunjukkan bahwa kemudahan akses terhadap riwayat pemutaran secara signifikan mempengaruhi tingkat kepuasan pengguna (Angela et al., 2022).

Dalam konteks pengembangan fitur “Recently Played”, struktur data tumpukan (stack) dengan prinsip LIFO (Last-In-First-Out) menawarkan solusi yang elegan untuk mengelola data riwayat pemutaran. Prinsip LIFO secara alami sesuai dengan karakteristik fitur riwayat pemutaran, di mana lagu-lagu yang paling baru diputar menjadi prioritas utama bagi pengguna (Erkamim dkk., 2024). Implementasi tumpukan memungkinkan operasi efisien melalui metode push untuk menambahkan lagu baru dan metode pop untuk menghapus lagu tertua saat kapasitas penyimpanan terpenuhi (Musiafa, 2022). Pendekatan ini memastikan akses cepat ke konten terbaru sementara data lama secara otomatis digantikan.

Meskipun secara teori struktur data stack menawarkan pendekatan ideal untuk pengelolaan data historis, masih terdapat celah signifikan akibat kurangnya penelitian implementasi rinci dalam lingkungan streaming musik (Marlia dkk., 2024). Analisis fitur musik dan pola popularitas lagu di Spotify menunjukkan kompleksitas dalam mengelola data audio, yang memerlukan pemilihan struktur data yang sesuai. Studi ini bertujuan untuk menerapkan struktur data tumpukan pada sistem “Recently Played” di platform streaming musik. Pentingnya penelitian ini didasarkan pada kebutuhan akan solusi optimal untuk mengelola data historis dinamis, dengan mempertimbangkan batasan memori dan efisiensi akses sebagai faktor kunci. Melalui penerapan tumpukan, diharapkan pengelolaan riwayat pemutaran yang lebih efektif dapat dicapai, dengan kompleksitas waktu konstan pada operasi dasar, sehingga meningkatkan pengalaman pengguna secara keseluruhan dalam aplikasi streaming musik dan memberikan nilai praktis bagi pengembang dalam menentukan struktur data yang tepat untuk fitur berbasis riwayat pengguna.

METODE PENELITIAN

Metode penelitian yang diterapkan adalah rancang bangun (*design and build*), yang dilaksanakan melalui tiga tahapan sistematis: identifikasi kebutuhan, perancangan sistem, dan implementasi. Pendekatan ini relevan karena tujuan akhir penelitian adalah untuk menghasilkan sebuah produk perangkat lunak yang fungsional dan dapat divalidasi

Tahap Identifikasi Kebutuhan

Tahap awal penelitian ini berfokus pada analisis masalah dan penentuan kebutuhan sistem secara menyeluruh. Proses ini dimulai dengan mengidentifikasi adanya kebutuhan akan sebuah pemutar musik berbasis web yang ringan, mandiri (*standalone*), dan tidak bergantung



pada server untuk menjalankan fungsi utamanya. Dari analisis tersebut, dirumuskan serangkaian kebutuhan fungsional yang menjadi acuan, meliputi kemampuan untuk memutar (*play*), menjeda (*pause*), menavigasi lagu (*next & previous*), menampilkan daftar putar, serta menyediakan fungsi untuk menambah dan menghapus lagu. Sebagai fitur pembeda, aplikasi ini juga diharuskan memiliki kemampuan untuk kembali ke lagu yang terakhir diputar. Selain itu, ditetapkan pula kebutuhan non-fungsional yang menekankan pada antarmuka yang responsif, kinerja cepat, dan pengalaman pengguna yang intuitif.

Tahap Perancangan Sistem

Pada tahap perancangan, semua kebutuhan yang telah diidentifikasi diterjemahkan menjadi sebuah cetak biru (*blueprint*) teknis. Sistem dirancang dengan arsitektur sisi klien murni, di mana HTML5, CSS3, dan JavaScript menjadi fondasi utamanya. Perancangan antarmuka (UI/UX) menghasilkan tata letak sederhana yang terdiri dari area kontrol, daftar putar, dan formulir input dengan tema gelap untuk estetika modern. Keputusan perancangan yang paling krusial terletak pada pemilihan struktur data. Untuk mengelola daftar putar, dipilih Doubly Linked List karena efisiensi waktu konstan yang ditawarkannya untuk operasi navigasi dan penghapusan. Sementara itu, untuk fitur riwayat pemutaran, dirancang penggunaan struktur data Stack yang sifatnya *Last-In, First-Out* (LIFO), sangat sesuai untuk fungsi "kembali ke lagu sebelumnya".

Parameter Analisis Komparatif

Tahap implementasi merupakan proses realisasi dari cetak biru perancangan menjadi kode program yang fungsional. Proses ini dimulai dengan membangun struktur atau kerangka halaman web menggunakan HTML5, termasuk elemen `<audio>` sebagai pemutar inti. Selanjutnya, rancangan visual diwujudkan menggunakan CSS3 untuk menerapkan tema gelap, tata letak, dan efek transisi. Bagian inti dari tahap ini adalah implementasi logika menggunakan JavaScript. Di sini, kelas `SongNode`, `Playlist`, dan `History` dibuat untuk merealisasikan struktur data yang telah dirancang. Fungsi-fungsi untuk memanipulasi DOM ditulis untuk memastikan antarmuka dapat diperbarui secara dinamis. Terakhir, *event listener* ditambahkan pada setiap elemen interaktif untuk menghubungkan aksi pengguna dengan fungsi yang relevan, sehingga menciptakan aplikasi yang responsif dan fungsional secara penuh.

HASIL DAN PEMBAHASAN

Hasil Penelitian

Penelitian ini berhasil menghasilkan sebuah prototipe aplikasi pemutar musik interaktif berbasis web yang sepenuhnya fungsional, sesuai dengan seluruh tujuan yang ditetapkan dalam tahap perancangan. Hasil akhir adalah produk perangkat lunak mandiri yang berjalan efisien di sisi klien. Aplikasi ini memiliki antarmuka pengguna yang bersih dan dilengkapi dengan semua fitur fungsional yang telah diidentifikasi. Sistem kontrol pemutaran (putar, jeda, berikutnya, sebelumnya) beroperasi dengan responsif.

Manajemen daftar putar yang dinamis memungkinkan pengguna untuk menambah lagu baru melalui formulir dan menghapus lagu dari daftar secara instan. Selain itu, fitur unik "Back to Last Played" yang dirancang khusus juga berhasil diimplementasikan dan berfungsi dengan



benar. Pengujian fungsional yang dilakukan pada setiap fitur menunjukkan bahwa seluruh komponen aplikasi berjalan sesuai harapan tanpa kesalahan.

Pembahasan

Keberhasilan dan kinerja aplikasi ini dapat dianalisis melalui efektivitas keputusan yang diambil pada tahap perancangan, terutama terkait arsitektur dan struktur data. Pemilihan arsitektur sisi klien terbukti menjadi fondasi utama untuk pengalaman pengguna yang sangat responsif. Karena semua logika dan manajemen data ditangani langsung di peramban, tidak ada latensi jaringan (*network latency*) untuk fungsi-fungsi inti. Hal ini membuat aplikasi terasa instan dan memenuhi kebutuhan non-fungsional akan kinerja yang tinggi. Secara teknis, implementasi Doubly Linked List untuk daftar putar adalah keputusan yang paling berdampak pada efisiensi. Struktur data ini memungkinkan operasi navigasi (*next* dan *previous*) dieksekusi dalam waktu konstan ($O(1)$), yang sangat ideal untuk fungsi utama pemutar musik. Jika dibandingkan dengan struktur data alternatif seperti array, penghapusan item dari tengah daftar jauh lebih efisien karena tidak memerlukan pergeseran elemen lain. Di sisi lain, penggunaan Stack untuk fitur riwayat adalah solusi yang elegan. Sifatnya yang *Last-In, First-Out* (LIFO) secara inheren cocok dengan logika "kembali ke lagu terakhir", membuat implementasi fitur tersebut menjadi sederhana dan sangat efisien. Secara keseluruhan, hasil penelitian ini menunjukkan bahwa keberhasilan sebuah produk perangkat lunak sangat bergantung pada keselarasan antara masalah yang ingin diselesaikan dengan pemilihan arsitektur dan struktur data yang tepat.

KESIMPULAN

Berdasarkan implementasi dan analisis yang telah dilakukan, penelitian ini berhasil membuktikan bahwa struktur data stack dengan prinsip LIFO (*Last-In-First-Out*) merupakan solusi yang sangat efektif dan efisien untuk sistem riwayat pemutaran (*Recently Played*) pada aplikasi musik. Melalui metode rancang bangun, prototipe aplikasi yang dihasilkan mampu menjalankan semua fungsi yang dirancang, dengan fitur "Back to Last Played" berbasis *stack* yang bekerja secara responsif dan intuitif. Keberhasilan ini menegaskan bahwa pemilihan struktur data yang tepat, dalam hal ini *stack* untuk data historis dan *Doubly Linked List* untuk daftar putar, secara langsung berkontribusi pada peningkatan kinerja dan pengalaman pengguna. Dengan demikian, penelitian ini menunjukkan bahwa penerapan *stack* adalah pendekatan optimal untuk mengelola data riwayat yang dinamis dalam lingkungan *streaming* musik, menawarkan solusi yang elegan dengan kompleksitas waktu konstan untuk operasi dasar.

DAFTAR PUSTAKA

Angela, Halim, F., Pramana, T. A., & Simanjuntak, A. (2022). User experience evaluation on music streaming applications with UEQ method. *JURTEKSI (Jurnal Teknologi dan Sistem Informasi)*, VIII(2), 209–218. <https://doi.org/10.33330/jurteksi.v8i2.1323>



- Erkamim, M., Abdurrohman, I., Yuliyanti, S., Karim, R., Rahman, A., Admira, T. M. A., & Ridwan, A. (2024). Buku ajar algoritma dan struktur data. PT. Sonpedia Publishing Indonesia.
- Firmansyah, M. I., Rohman, R. S., & Marsusanti, E. (2023). Penerapan algoritma klastering K-Means untuk fitur atribut pada layanan streaming musik Spotify. *Indonesian Journal Computer Science*, 2(2), 58-66.
- Ginting, S. H. N., Effendi, H., Kumar, S., Marsisno, W., Sitanggang, Y. R. U., Anwar, K., Santuari, N. P. L., Setyowibowo, S., Sigar, T. R., Atho'llah, I., Setyantoro, D., & Smrti, N. N. E. (2023). Pengantar struktur data. PT. Mifandi Mandiri Digital.
- Kementerian Pendidikan, Kebudayaan, Riset, dan Teknologi. (2021). *Informatika untuk SMA Kelas X*. Pusat Kurikulum dan Perbukuan.
- Marlia, S., Setiawan, K., & Juliane, C. (2024). Analisis fitur musik dan tren popularitas lagu di Spotify menggunakan K-Means dan CRISP-DM. *SISTEMASI: Jurnal Sistem Informasi*, 13(2), 595–607.
- Musiafa, Z. (2022). Buku ajar struktur data dan implementasi algoritma (SDIA): Bahasa pemrograman Python Java C C++.
- Netti, S. Y. M., & Irwansyah. (t.t.). *Spotify: Aplikasi music streaming untuk generasi milenial*. Universitas Pelita Harapan, Universitas Indonesia.
- Putri, G. M., Pradja, K. A. D., Azizi, M. B. M., Nurwahi, P., Perdana, A. S., & Munawir. (2024). Implementasi stack dan array pada pengurutan lagu dengan metode selection sort. *Jurnal Teknologi Dan Sistem Informasi Bisnis*, 6(2), 286-296. <https://doi.org/10.47233/jteksis.v6i2.1192>
- Ruddin, I., Santoso, H., & Indrajit, R. E. (2022). Digitalisasi musik industri: Bagaimana teknologi informasi mempengaruhi industri musik di Indonesia. *Jurnal Pendidikan Sains dan Komputer*, 2(1). <https://doi.org/10.47709/jpsk.v2i1.1395>
- Safwandi. (t.t.). *Struktur data*. Bahan Ajar. Fakultas Teknik, Universitas Malikussaleh.
- Santoso, J. T. (t.t.). Prevent database transmission of the about The Notis environment to receive. *YR/NSAN PRIMA AGIS TEKNIK*.
- Gunawan, P., & Voutama, A. (2025). *Rancang bangun website e-commerce "One Step Store" menggunakan teknologi MERN Stack dengan metode waterfall*. *Jurnal Sistem Informasi dan Teknologi*, 8(2).
- Mushthofa, Wahyono, Asfarian, A., Ramadhan, D. A., Putro, H. P., Wisnubhadra, I., Saputra, B., & Pratiwi, H. (2021). *Informatika untuk SMA Kelas X*. Pusat Kurikulum dan Perbukuan, Badan Penelitian dan Pengembangan dan Perbukuan, Kementerian Pendidikan, Kebudayaan, Riset, dan Teknologi.



- Musyaffa, A. I., Zulfa, M. I., & Alim, M. S. (2024). *Rancang bangun PureCompute platform e-commerce untuk belanja laptop berbasis website*. SINTA Jurnal Sistem Informasi dan Teknologi Komputasi, 1(1), 21–29.
- Santoso, J. T. (2021). *Struktur data dan algoritma (Bagian 1)*. Yayasan Prima Agus Teknik.
- Setiyawan, R. D., Hermawan, D., Abdillah, A. F., Mujayanah, A., & Vindua, R. (2024). *Penggunaan struktur data stack dalam pemrograman C++ dengan pendekatan array dan linked list*. JUTECH (Journal Education and Technology), 5(2).
- Wijaya, G., Permana, S. W., Bagaskara, N. N., & Santi, I. H. (2025). *Rancang bangun sistem marketplace toko karya sembako dengan stack teknologi MERN*. JITU: Journal Informatic Technology and Communication, 9(1), 71-82.