

# Implementasi Website Pemesanan Tiket Konser Online dengan Framework Next.js

Rizky Yuniz Teresya<sup>\*#1</sup>, M Lazuardi Al Ghiffary<sup>#2</sup>, Indyah Hartami Santi<sup>#3</sup>

<sup>#123</sup>Teknik Informatika, Universitas Islam Balitar, Blitar, Indonesia

Korespondensi author <sup>\*</sup>yt3312616@gmail.com

## Info Artikel

**Diajukan:** 20 Mei 2025

**Diterima:** 3 Juli 2025

**Diterbitkan:** 9 Juli 2025

### Keywords:

Website; Concert Ticket; Next.js;  
Firebase; Waterfall Method;  
Concert Ticket Booking System

### Kata Kunci:

Website; Event Konser; Next.js;  
Firebase; Metode Waterfall;  
Sistem Pemesanan Tiket Konser



Lisensi: cc-by-sa

Copyright © 2025 Rizky Yuniz Teresya, M  
Lazuardi Al Ghiffary, Indyah Hartami Santi

## Abstract

*The rapid development of information technology has driven digital transformation in various aspects of life, including the ticket booking system for music concerts. Conventional ticketing systems still face several issues such as long queues, the risk of losing physical tickets, and illegal ticket reselling. This study, titled "Implementation of an Online Concert Ticket Booking Website Using the Next.js Framework", aims to design a web-based ticket booking system to improve transaction efficiency and security. The website is developed using Next.js, a modern framework that supports high performance and server-side rendering (SSR), and Firebase, a real-time and secure backend service. The development process follows the Waterfall method, starting from requirement analysis, system design, implementation, testing, to maintenance. The system design is modeled using Unified Modeling Language (UML) to provide a structured and detailed overview of system workflows. The result is a website that features digital ticket booking, integration with online payment methods, and concert data management by the admin. This system is expected to minimize transaction errors, enhance user convenience, and offer a modern solution for concert ticket management.*

## Abstrak

*Perkembangan teknologi informasi telah mendorong digitalisasi dalam berbagai aspek kehidupan, termasuk dalam sistem pemesanan tiket konser. Sistem konvensional memiliki berbagai kelemahan, seperti antrean panjang, risiko kehilangan tiket, dan maraknya penjualan ilegal. Penelitian ini berjudul "Implementasi Website Pemesanan Tiket Konser Online dengan Framework Next.js", yang bertujuan untuk merancang sistem pemesanan tiket berbasis web guna meningkatkan efisiensi dan keamanan dalam proses transaksi. Pengembangan website menggunakan framework Next.js karena mendukung performa tinggi dan rendering sisi server (SSR), serta Firebase sebagai layanan backend real-time yang aman dan terintegrasi. Metode pengembangan yang digunakan adalah Waterfall, dimulai dari analisis kebutuhan, perancangan, implementasi, pengujian, hingga pemeliharaan. Desain sistem digambarkan menggunakan Unified Modeling Language (UML) sebagai alat bantu pemodelan yang terstruktur. Hasil dari penelitian ini berupa sebuah website yang menyediakan fitur pemesanan tiket digital, integrasi dengan sistem pembayaran online, serta pengelolaan data konser oleh admin. Sistem ini diharapkan mampu meminimalkan kesalahan transaksi, meningkatkan kenyamanan pengguna, dan memberikan solusi modern dalam manajemen tiket konser.*

### Cara mensitasi artikel:

R. Y. Teresya, M. L. A. Ghiffary, I. H. Santi. "Implementasi Website Pemesanan Tiket Konser Online dengan Framework Next.js." *Jurnal Teknologi Informasi: Teori, Konsep, dan Implementasi (JTI-TKI)*, vol. 15, no. 2, pp.58-64, Oktober 2025, <https://doi.org/10.36382/jti-iki.v16i2.577>

## PENDAHULUAN

Perkembangan teknologi informasi yang pesat telah mendorong perubahan signifikan dalam berbagai aspek kehidupan manusia, termasuk dalam hal pemesanan tiket untuk berbagai jenis acara, khususnya konser musik. Seiring dengan meningkatnya popularitas *event* hiburan seperti konser artis lokal hingga internasional, kebutuhan akan sistem pemesanan tiket yang cepat, aman, dan efisien semakin meningkat [1]. Di sisi lain, sistem pemesanan tiket konvensional masih menyimpan berbagai kelemahan, seperti antrean panjang, risiko kehilangan tiket fisik, hingga praktik penjualan tiket ilegal yang merugikan banyak pihak[2].

Untuk mengatasi permasalahan tersebut, pemesanan tiket konser kini mulai beralih ke sistem berbasis digital melalui platform *website*. *Website* merupakan sekumpulan tampilan halaman yang sering dipakai untuk memberikan pesan atau keterangan berupa gambar, teks, suara, atau perpaduan dari beberapa komponen, yang tampilannya

bisa statis atau dinamis berubah-ubah yang merangkai suatu tampilan *website* yang saling berhubungan dengan jaringan-jaringan yang lainnya [3]. Dengan memanfaatkan *website*, proses event manajemen pemesanan tiket konser menjadi lebih praktis, efisien, serta dapat diakses kapan saja dan di mana saja oleh pengguna[4].

Dalam konteks konser, *event* merupakan sarana untuk mengumpulkan orang dalam satu kegiatan hiburan dengan tujuan komersial maupun promosi budaya [5]. Oleh karena itu, sistem manajemen *event* yang terintegrasi dengan teknologi informasi memiliki nilai strategis dalam mendukung penyelenggaraan acara. Dalam praktiknya, masyarakat saat ini lebih memilih metode daring (*online*) untuk melakukan aktivitas pembelian maupun pendaftaran karena dinilai lebih cepat dan praktis. Fenomena ini semakin kuat pasca-pandemi, di mana kebiasaan digitalisasi menjadi solusi utama dalam berbagai sektor, termasuk industri hiburan [6].

Sistem Pemesanan Tiket Konser adalah suatu platform yang dirancang untuk mempermudah pengguna dalam

memperoleh tiket konser via situs web, sehingga pengguna dapat dengan mudah mencari dan membeli tiket konser[7]. Sistem pemesanan tiket secara *online* dianggap mampu mengurangi kesalahan transaksi dan meningkatkan kenyamanan pengunjung dalam mengakses tiket konser [8].

Merespons permasalahan dan peluang tersebut, penulis mengimplementasikan sebuah *website* pemesanan tiket konser *online* menggunakan *framework* *Next.js* dan *Firebase* sebagai basis data.

*Next.js* adalah *framework* *React* yang skalabel dan berkinerja tinggi untuk pengembangan web modern, menyediakan berbagai fitur seperti *rendering* hibrida, *prefetching* rute, optimisasi gambar otomatis, dan internasionalisasi secara langsung[9]. *Framework* *Next.js* dipilih karena kemampuannya dalam membangun aplikasi web modern yang mendukung *rendering* sisi server (SSR) dan optimalisasi performa[10]. Sedangkan *Firebase* adalah platform *Backend-as-a-Service* (BaaS) yang dikembangkan oleh Google untuk mempermudah pengembangan aplikasi web dan *mobile*[11]. *Firebase* digunakan sebagai solusi *real-time database* yang memungkinkan integrasi cepat dan aman antara data tiket, pengguna, dan transaksi.

Penelitian ini menggunakan metode rekayasa perangkat lunak, dengan pendekatan metode *waterfall* sebagai landasan dalam tahapan perancangan sistem, mulai dari analisis kebutuhan, desain sistem, implementasi, pengujian, hingga pemeliharaan. Setiap tahap dilakukan secara sistematis untuk menghasilkan aplikasi yang tidak hanya fungsional tetapi juga dapat meningkatkan efisiensi manajemen tiket oleh penyelenggara konser serta memberikan kenyamanan dan transparansi bagi pengunjung. Dalam implementasinya, *website* dirancang menggunakan pemodelan *Unified Modeling Language* (UML) untuk menggambarkan alur sistem secara detail, mulai dari *use case*, *activity diagram*, hingga *sequence diagram*, sehingga pengembangan dapat dilakukan secara terstruktur dan sesuai kebutuhan pengguna.

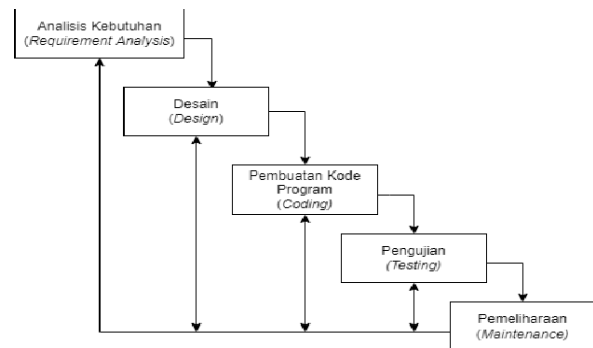
Dengan hadirnya sistem ini, diharapkan proses pemesanan tiket konser menjadi lebih modern, minim kesalahan, serta memberikan solusi dari permasalahan pemesanan secara manual yang masih banyak terjadi. Selain itu, sistem ini juga memberikan fitur tiket *online* sebagai tiket digital yang dapat dipindai saat masuk *venue*, serta menyimpan data secara otomatis di *cloud* melalui *Firebase* untuk efisiensi dan keamanan yang lebih tinggi.

## METODE

Penelitian ini menerapkan metode *waterfall* dalam pengembangan *website* pembelian tiket konser. Model air terjun (*Waterfall Model*) adalah pendekatan klasik dalam pengembangan perangkat lunak yang menggambarkan metode pengembangan linier dan berurutan. Ini terdiri dari lima hingga tujuh fase, setiap fase didefinisikan oleh tugas dan tujuan yang berbeda, di mana keseluruhan fase

menggambarkan siklus hidup perangkat lunak hingga pengirimannya. Setelah fase selesai, langkah pengembangan selanjutnya mengikuti dan hasil dari fase sebelumnya mengalir ke fase berikutnya [12].

Pendekatan ini dipilih karena bersifat sistematis dan terstruktur, yang meliputi beberapa proses diantaranya analisis kebutuhan, desain, pembuatan kode, pengujian, dan pemeliharaan[13]. Tahapan dari metode *waterfall* seperti gambar 1.



Gambar. 1 Metode Waterfall

Berikut tahapan metode *waterfall* untuk pengembangan *website* pemesanan tiket konser *online* menggunakan *framework* *Next.js*.

### A. Analisa kebutuhan

Pada tahap ini dilakukan identifikasi terhadap seluruh kebutuhan sistem baik dari sisi pengguna maupun penyelenggara konser. Kebutuhan yang dikumpulkan meliputi kemampuan pengguna untuk melihat daftar konser, melakukan pemesanan tiket secara *online*, melakukan pembayaran, serta fitur *login* dan registrasi. Sementara dari sisi admin, sistem harus menyediakan fitur untuk mengelola data konser, memantau transaksi, dan menghasilkan laporan. Kebutuhan ini juga mencakup aspek teknis seperti responsifitas *website*, keamanan data pengguna, serta integrasi dengan layanan pihak ketiga seperti API pembayaran.

Dalam penelitian ini, perancangan dan pengembangan sistem dilakukan menggunakan *framework* *Next.js* sebagai *frontend* dan *Firebase* sebagai *backend*, yang memberikan keunggulan dalam hal kecepatan, skalabilitas, dan kemudahan integrasi dengan layanan modern berbasis *cloud*. Pendekatan ini membedakan penelitian ini dari penelitian terdahulu yang berjudul "*Sistem Transaksi untuk Penjualan Tiket Event Musik Berbasis Web*"[14], di mana sistem dibangun menggunakan bahasa pemrograman PHP dan database MySQL. Selain perbedaan teknologi, sistem dalam penelitian terdahulu tersebut masih memiliki fitur yang terbatas, seperti tidak adanya integrasi pembayaran *online* dan belum didukung oleh arsitektur modern berbasis *framework*. Oleh karena itu, sistem yang dikembangkan dalam penelitian ini dirancang lebih kompleks, modern, dan efisien untuk menjawab kebutuhan pengguna masa kini.

### B. Desain

Setelah kebutuhan dianalisis, tahap berikutnya adalah merancang sistem berdasarkan kebutuhan tersebut. Proses desain mencakup pembuatan struktur antarmuka pengguna (UI) serta pemodelan sistem menggunakan *Unified Modeling Language* (UML). Beberapa diagram yang digunakan meliputi use case diagram untuk menggambarkan interaksi antara pengguna dan sistem, *activity diagram* untuk menjelaskan alur proses pemesanan tiket. Selain itu, dilakukan perancangan arsitektur sistem berbasis Next.js sebagai *frontend* dan Firebase sebagai *backend*.

### C. Pembuatan Kode program

Tahap implementasi merupakan proses mengubah hasil desain ke dalam bentuk kode program nyata. *Website* dibangun menggunakan framework Next.js untuk sisi *frontend* karena mendukung performa tinggi, *server-side rendering*, serta struktur pengembangan modern. Firebase digunakan untuk menangani autentikasi pengguna, penyimpanan data (menggunakan Firestore), serta pengelolaan *database* secara *real-time*.

### D. Pengujian

Pada tahap pengujian, website diuji oleh seorang *validator* yang merupakan ahli di bidang teknologi informasi. Pengujian difokuskan pada fungsi utama seperti *login*, pemesanan tiket, pengelolaan konser, dan tampilan antarmuka. Hasil pengujian menunjukkan bahwa *website* berjalan dengan baik dan sesuai kebutuhan, namun *validator* memberikan beberapa saran perbaikan seperti penambahan notifikasi pemesanan, riwayat transaksi pengguna, serta peningkatan validasi keamanan.

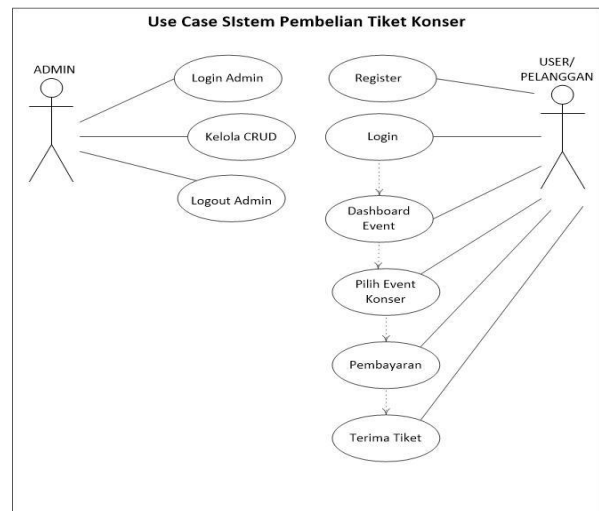
### E. Pemeliharaan

Tahap terakhir adalah pemeliharaan, Proses ini mencakup perbaikan apabila ditemukan bug atau kesalahan saat penggunaan, pembaruan sistem apabila terjadi perubahan kebutuhan dari pengguna atau penyelenggara konser, serta peningkatan performa dan keamanan sistem.

### F. Unified Modeling Language (UML)

*Unified Modeling Language* (UML) adalah salah satu standar bahasa yang banyak digunakan di dunia industri untuk mendefinisikan requirement, membuat analisis dan desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek[15].

1) *Use Case Diagram*: *Use case* menggambarkan interaksi antara pengguna dan sistem dalam proses pembelian tiket konser secara *online*. *Use case* ini mencakup aktivitas dari *user/pelanggan* dan *admin*. Alur *website use case diagram* pembelian tiket konser digambarkan pada Gambar 2.



Gambar. 2 Use Case Diagram

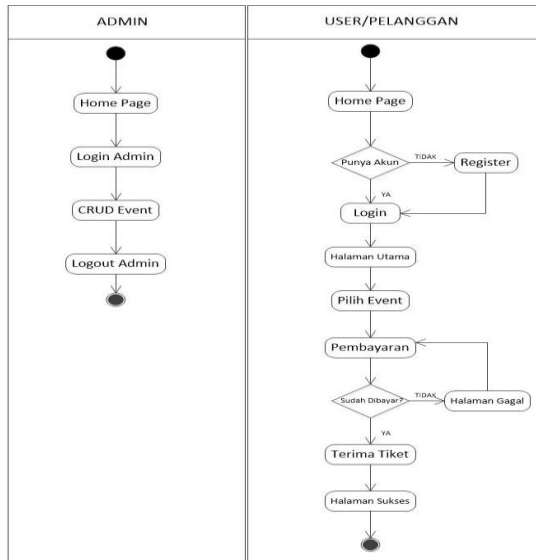
- Admin - Mengelola sistem dan konten konser, *Use Case* untuk Admin: Admin mengakses sistem dengan menggunakan kredensial khusus (Masuk Admin). Admin memiliki kemampuan untuk membuat, melihat, memperbarui, dan menghapus rincian acara konser (CRUD). Setelah menyelesaikan tugasnya, Admin keluar dari sistem (Keluar Admin).

- User/Pelanggan, Memanfaatkan sistem untuk membeli tiket konser. *Use Case* untuk User/Pelanggan: Pelanggan membuat akun baru jika mereka belum memiliki akun (Daftar). Setelah akun siap, pelanggan masuk ke sistem dengan akun yang sudah ada (Masuk). Pelanggan menelusuri daftar acara konser yang ada (Lihat Acara) dan kemudian memilih konser yang ingin dihadiri (Pilih Acara). Setelah memilih acara, pelanggan menyelesaikan pembayaran untuk tiket yang diinginkan (Pembayaran), dan tiket akan diterima setelah pembayaran berhasil (Terima Tiket).

2) *Diagram activity*: *Diagram activity* yang menunjukkan aliran proses dari sistem pembelian tiket konser dibagi menjadi dua alur utama, yaitu Admin dan User (Pelanggan). Gambar Alur *diagram activity* sebagai mana terlihat di Gambar 3.

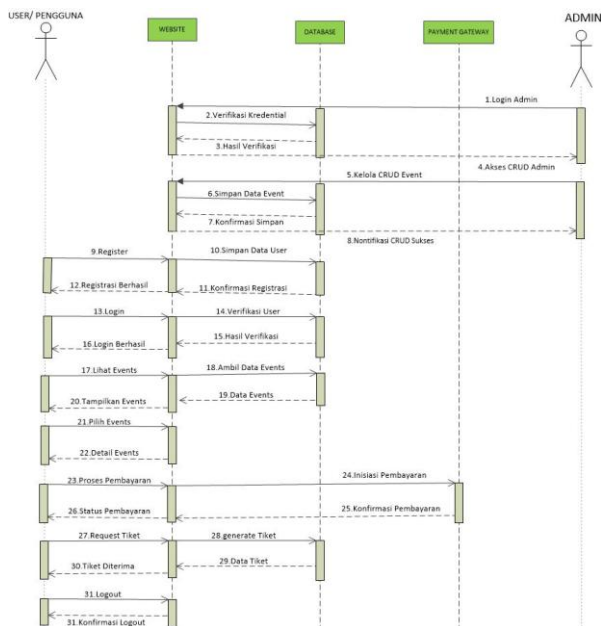
- Alur Admin dimulai dari Start Point, kemudian Admin mengunjungi beranda situs web (*Home Page*) dan melakukan *Login Admin*. Setelah berhasil masuk, Admin melakukan *CRUD Events*. Setelah semua tugas selesai, Admin melakukan *Logout Admin*, dan diakhiri dengan *End Point*.
- Alur User/Pelanggan dimulai dari *Start Point*, kemudian User mengunjungi beranda situs web (*Home Page*). Di tahap Memiliki Akun?, jika Ya, lanjut ke *Login*, jika Tidak, ke Registrasi, lalu diarahkan untuk *Login*. Setelah masuk, User melihat Halaman Utama, melakukan *Pilih Event*, dan Pembayaran. Di tahap Sudah Dibayar?, jika Ya, lanjut

ke Terima Tiket, jika Tidak, diarahkan ke halaman gagal dan dapat mencoba pembayaran lagi. Setelah tiket diterima, *User* menuju ke Halaman Sukses, dan diakhiri dengan *End Point*.



Gambar 3. Diagram Activity

3) *Diagram Sequence*: *Diagram sequence* menunjukkan interaksi dan pertukaran pesan antar objek dalam sistem pembelian tiket konser secara berurutan. Aktor dan objek utama dalam sistem terdiri dari: *User* sebagai pelanggan yang akan membeli tiket konser, *Website* sebagai antarmuka sistem yang diakses oleh *user* dan *admin*, *Database* sebagai penyimpanan data sistem, *Payment Gateway* sebagai layanan pembayaran pihak ketiga, dan *Admin* sebagai pengelola sistem. Gambar 4 adalah Interaksi *Diagram Sequence*.



Gambar 4. Diagram Sequen

- Proses Admin dimulai dengan *Login Admin* (1-4), di mana Admin masuk ke sistem, kredensialnya diverifikasi, dan diberikan akses ke fungsi CRUD. Selanjutnya, dalam proses *Create/Update Event* (5-8), Admin membuat atau memperbarui *event* konser, data disimpan di *database*, dan konfirmasi diberikan.
- Proses *User* dimulai dengan *Register* (9-12), di mana *User* mendaftar akun baru, data disimpan di *database*, dan konfirmasi registrasi diberikan. Setelah itu, proses dilanjutkan dengan *Login* (13-16), di mana *User* masuk ke sistem, kredensialnya diverifikasi, dan akses diberikan. Setelah berhasil *login*, *User* melihat daftar *event* konser yang tersedia (*View Events* 17-20), lalu *User* memilih *event* konser spesifik untuk dihadiri (*Select Event* 21-22). Proses selanjutnya adalah *Payment Process* (23-26), di mana *User* melakukan pembayaran melalui *payment gateway* dan menerima konfirmasi pembayaran. Setelah pembayaran berhasil, sistem melanjutkan ke proses *Issue Ticket* (27-30), yaitu sistem mengeluarkan tiket untuk *user*. Terakhir, *User* keluar dari sistem (*Logout* 31-32).

## HASIL DAN PEMBAHASAN

### A. Data Pemesanan Tiket Konser

*Dataset* yang disusun dan digunakan dalam sistem ini adalah dituliskan pada Tabel 1.

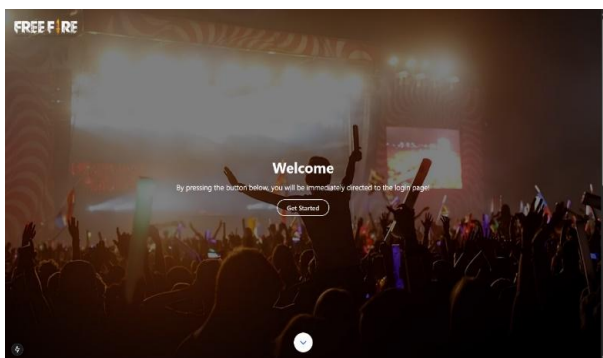
Tabel 1. Daftar Pemesanan Tiket Konser Online

ID Pembelian	Nama Pembeli	Jenis Tiket	Jumlah Tiket
TKT001	Rizky Maulana	VIP	2
TKT002	Siti Nurhaliza	Reguler	1
TKT003	Andi Saputra	VVIP	3
TKT004	Dina Lestari	Reguler	2
TKT005	Budi Hartono	VIP	1
TKT006	Fajar Pratama	VVIP	1
TKT007	Lestari Wulan	Reguler	2
TKT008	Rangga Okta	VIP	3
TKT009	Intan Putri	Reguler	1
TKT010	Ahmad Fauzi	VVIP	2

Berdasarkan data pembelian tiket konser *online* pada tabel 1 yang terdiri dari 10 transaksi, terlihat bahwa sistem ini mampu mengakomodasi berbagai jenis tiket seperti Reguler, VIP, dan VVIP, yang masing-masing memiliki segmentasi harga dan fasilitas berbeda. Dalam data tersebut, tercatat bahwa pembeli berasal dari beragam nama dan kemungkinan latar belakang, menunjukkan bahwa sistem ini dapat diakses secara luas oleh berbagai kalangan.

### B. Tampilan website

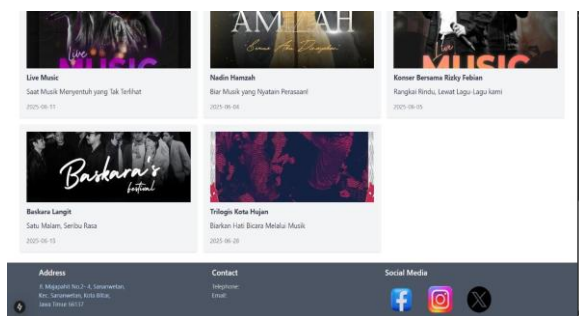
1) *Tampilan halaman utama*: Fitur halaman utama web yang telah dihasilkan seperti terlihat pada Gambar 5.



Gambar. 5 Halaman Utama

Tampilan halaman utama memiliki desain menarik dengan latar suasana konser malam hari, menampilkan logo "Free Fire", teks sambutan "Welcome", serta penjelasan singkat bahwa pengguna akan diarahkan ke halaman *login*. Fokus utama terletak pada tombol besar "Get Started" sebagai *call-to-action* untuk melanjutkan ke proses *login*. Desainnya sederhana namun efektif. Tepat di bawah teks tersebut terdapat tombol besar bertuliskan "Get Started" (Mulai) yang berfungsi sebagai *call-to-action* utama untuk melanjutkan ke proses *login*. Secara keseluruhan, desain ini sederhana namun efektif, dengan fokus yang jelas pada tombol "Get Started" untuk mengarahkan pengguna.

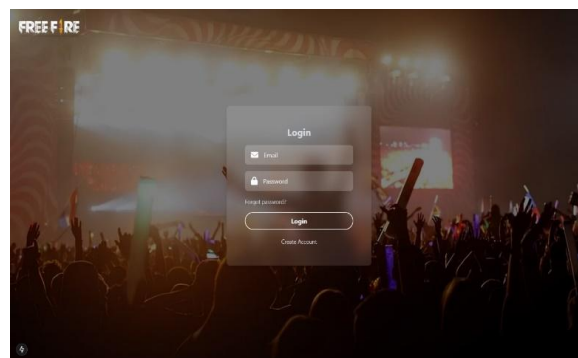
2) *Tampilan Section Bawah Home Page*: Fitur *section Home Page* yang telah dihasilkan seperti terlihat pada Gambar 6.



Gambar. 6 Halaman Section Home Page

Tampilan *event list* menampilkan daftar acara secara visual dengan poster, disertai informasi lengkap seperti nama acara, tanggal, waktu, dan lokasi. Halaman ini juga dilengkapi dengan informasi kontak, alamat penyelenggara, dan ikon media sosial di bagian bawah. Desainnya rapi dan informatif, memudahkan pengunjung mengetahui detail acara dan cara menghubungi penyelenggara.

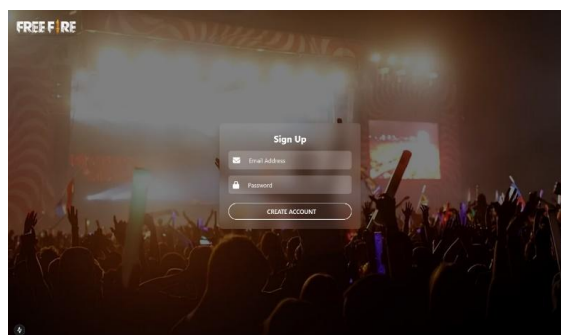
3) *Tampilan Halaman Login*: Fitur halaman *login* web yang telah dihasilkan seperti terlihat pada Gambar 7.



Gambar. 7 Halaman Login

Halaman *login* berfungsi untuk mengautentikasi pengguna dengan meminta *email* dan kata sandi. Dilengkapi fitur "Forgot Password" untuk pemulihan akun dan tombol "Create Account" bagi pengguna baru. Desainnya modern dan intuitif, memudahkan akses ke platform secara aman dan cepat.

4) *Tampilan halaman register*: Fitur halaman *login* web yang telah dihasilkan seperti terlihat pada Gambar 8.

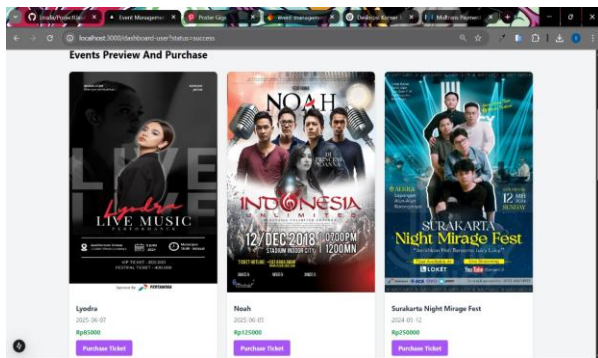


Gambar. 8 Halaman Register

Halaman ini berfungsi sebagai antarmuka pendaftaran (*register*) dengan *form* isian *email* dan *password*. Terdapat tombol "Create Account" untuk mengirim data dan membuat akun baru. Desainnya sederhana dan mudah dipahami, dengan latar belakang konser yang dinamis untuk menarik perhatian dan meningkatkan pengalaman pengguna.

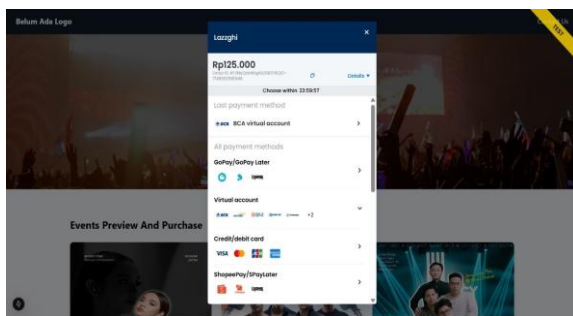
5) *Tampilan halaman event preview and purchase*: Fitur halaman *event preview and purchase* web yang telah dihasilkan seperti terlihat pada Gambar 9. Halaman *event preview* dan *purchase* menampilkan poster konser, tanggal pelaksanaan, dan harga tiket. Pengguna dapat melihat detail acara, lalu menekan tombol "Purchase Ticket" untuk diarahkan langsung ke halaman *payment gateway* guna menyelesaikan pembayaran.





Gambar. 9 Halaman Event Preview

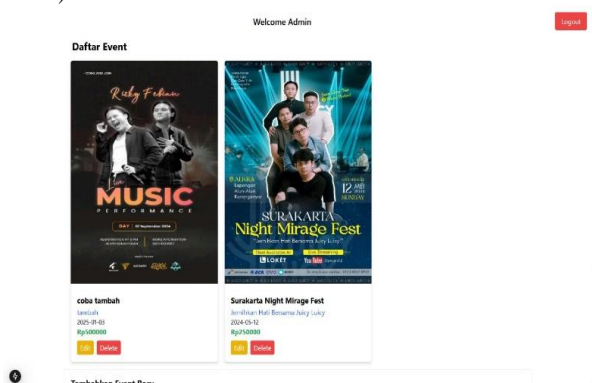
6) Tampilan *payment gateway*: Fitur halaman *payment gateway* yang telah dihasilkan seperti terlihat pada Gambar 10.



Gambar. 10 Halaman Payment Gateway

Halaman *payment gateway* digunakan untuk memfasilitasi proses pembayaran tiket konser secara *online*. Pengguna dapat memilih berbagai metode pembayaran yang tersedia, seperti *BCA Virtual Account*, kartu kredit/debit, dan metode pembayaran lainnya yang didukung oleh sistem. Setelah memilih metode yang diinginkan, pengguna akan diarahkan ke proses konfirmasi dan penyelesaian transaksi secara aman. Halaman ini dirancang untuk memberikan kemudahan, keamanan, dan kecepatan dalam melakukan pembayaran, sehingga memastikan tiket dapat dibeli tanpa hambatan.

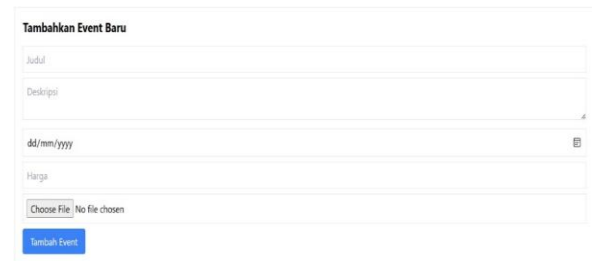
7) Halaman *dashboard* admin



Gambar. 11 Halaman Dashboard Admin

Halaman *dashboard* admin digunakan untuk mengelola daftar acara secara efisien. Admin dapat melihat acara dalam bentuk kartu berisi poster dan deskripsi singkat, serta menggunakan tombol "*Edit*" untuk mengubah data atau "*Delete*" untuk menghapus acara.

8) Halaman *tambah event baru*: Fitur halaman *payment gateway* yang telah dihasilkan seperti terlihat pada Gambar



Gambar. 12 Halaman Tambah Event

Halaman *dashboard* admin menyediakan *form* untuk menambahkan *event* baru dengan kolom judul, deskripsi, tanggal, harga, dan fitur unggah file. Setelah mengisi data, admin dapat menyimpan *event* dengan tombol "*Tambah Event*" untuk memudahkan pengelolaan acara.

## KESIMPULAN DAN SARAN

Berdasarkan hasil implementasi dan pembahasan, dapat disimpulkan bahwa pengembangan sistem pemesanan tiket konser berbasis *website* dengan menggunakan *framework* Next.js dan Firebase dapat menjadi solusi efektif untuk mengatasi permasalahan dalam sistem konvensional. *Website* ini mampu memfasilitasi pengguna dalam melihat informasi konser, melakukan pemesanan tiket secara *online*, serta menyelesaikan pembayaran melalui berbagai metode digital. Di sisi lain, admin dapat mengelola data konser, memantau transaksi, dan mengelola laporan secara efisien.

Pemanfaatan Next.js sebagai *frontend* memberikan kecepatan dan responsivitas, sementara integrasi dengan Firebase mendukung penyimpanan data secara *real-time* dan aman. Pendekatan pengembangan menggunakan metode *Waterfall* serta pemodelan menggunakan UML turut memastikan bahwa proses pengembangan dilakukan secara sistematis dan terstruktur. Secara keseluruhan, sistem ini meningkatkan efisiensi, kenyamanan, dan transparansi dalam manajemen tiket konser

Untuk pengembangan lebih lanjut sistem dapat ditingkatkan dengan penambahan fitur scan QR code untuk mempercepat validasi tiket dan autentikasi dua faktor (2FA) guna meningkatkan keamanan. Selain itu, tampilan antarmuka perlu disempurnakan agar lebih rapi, interaktif, dan *user-friendly*. Evaluasi kepuasan pengguna juga disarankan melalui uji coba lebih luas dan pengumpulan umpan balik langsung.

## UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih yang sebesar-besarnya kepada JTI-TKI atas dukungan dan kesempatan yang telah diberikan selama proses penyusunan jurnal ini. Penulis juga sangat berterima kasih kepada para dosen pembimbing yang telah memberikan bimbingan, arahan, dan motivasi yang sangat berarti. Selain itu, penulis mengapresiasi pihak penyedia platform yang telah memfasilitasi pengunggahan jurnal ini sehingga dapat diakses dengan mudah oleh pembaca. Semoga kerja sama dan dukungan dari semua pihak terus berlanjut demi kemajuan ilmu pengetahuan dan teknologi.

## REFERENSI

- [1] D. Afriyandi, R. Yuliana, And D. Widhi Yamin, "Perancangan Sistem Informasi Pemesanan Tiket Konser Dengan Metode Waterfall Berbasis Web," *Spectrum: Multidisciplinary Journal*, Vol. Vol. 2, No.1, Januari 2025, Jan. 2025.
- [2] R. I. Hermanto And Malabay, "Rancang Bangun Aplikasi Pemesanan Tiket Event Di Kota Jakarta Berbasis Website," *Ikraith-Informatika*, Vol. Vol 5 No 3, Nov. 2021.
- [3] H. Bekti, *Mahir Membuat Website Dengan Adobe Dreamweaver Cs6, Css, Dan JQuery*. Yogyakarta: Andi, 2015.
- [4] M. A. Ardiansa And T. Haryanti, "Pengembangan Sistem Informasi Penjualan Tiket Konser Online Eventku.Id Berbasis Website Php," *Jurnal Ilmiah Computing Insight*, Vol. Vol.3 No.2, 2021.
- [5] Riadi, *Manajemen Event*. 2019.
- [6] M. A. Nugroho And M. Fachrie, "Pengembangan Aplikasi Penjualan Tiket Acara Berbasis Website Dan Mobile," *Jurnal Indonesia : Manajemen Informatika Dan Komunikasi*, Vol. Vol 5 No 1, Januari (2024), Jan. 2024.
- [7] I. S. L. Siantar, And A. Voutama, "Perancangan Dan Pengembangan Sistem Informasi Aplikasi Berbasis Website Pemesanan Tiket Konser Dengan Menerapkan Pemodelan Unified Modeling Language," *Jati (Jurnal Mahasiswa Teknik Informatika)*, Vol. Vol. 8 No. 6, Desember 2024.
- [8] R. Sari And A. Kurniawan, "Sistem Informasi Pemesanan Tiket Konser Berbasis Web.," *Jurnal Teknologi Dan Sistem Komputer*, 2021.
- [9] M. Riva, *Real-World Next.js: Build Scalable, High-Performance, And Modern Web Applications Using Next.js, The React Framework For Production*. Packt Publishing, 2022.
- [10] A. Y. Prasetyo And G. Swalaganata, "Sistem Informasi Manajemen Ticketing Event Dengan Payment Gateway Berbasis Node.js Dan Next.js," *Journal Of Information System And Application Development*, Sep. 2024.
- [11] F. P. Rosyana, P. Onno W, And A. A. Rz, *Membangun Aplikasi Berbasis Android*. Penerbit Andi, 2021.
- [12] F. N. Hasanah And R. Sri Untari, *Rekayasa Perangkat Lunak*. Umsida Press, 2020.
- [13] F. Wahyuni, "Perancangan Sistem Informasi Kas Berbasis Web Dengan Menggunakan Metode Waterfall," *Methomika: Jurnal Manajemen Informatika & Komputerisasi Akuntansi*, Vol. Vol. 7 No. 1, Apr. 2023, Doi: <https://doi.org/10.46880/Jmika.Vol7no1.Pp138-143>.
- [14] I. W. Rahayu, A. N. Andini, And S. M. I. Putri, "Sistem Transaksi Untuk Penjualan Tiket Event Musik Berbasis Web," *Jurnal Teknik Informatika*, Vol. Vol. 16, No. 1, Jan. 2024.
- [15] Rosa A.S And M. Shalahuddin, *Rekayasa Perangkat Lunak: Terstruktur Dan Berorientasi Objek*. Informatika, 2016.