

# Analisis dan Implementasi Arsitektur Three-Tier pada Sistem E-Ticketing Konser Berbasis Web

Riski Dewanti<sup>1\*</sup>, Kholiq Nur Cahyo<sup>2</sup>, Ivan Yosa Andrianto<sup>3</sup>, Radja Zain Nuur Rofiif<sup>4</sup>, Salsa Sabila<sup>5</sup>,  
Agustina Srirahayu<sup>6</sup>

<sup>1</sup>Teknik informatika

Universitas Duta Bangsa

<sup>1\*</sup>240103176@udb.ac.id (penulis  
korespondensi)

<sup>2</sup>Teknik informatika

Universitas Duta Bangsa

<sup>2</sup>240103164@udb.ac.id

<sup>3</sup>Teknik informatika

Universitas Duta Bangsa

<sup>3</sup>240103165@udb.ac.id

<sup>4</sup>Teknik informatika

Universitas Duta Bangsa

<sup>4</sup>240103174@udb.ac.id

<sup>5</sup>Teknik informatika

Universitas Duta Bangsa

<sup>5</sup>240103177@udb.ac.id

<sup>6</sup>Teknik informatika

Universitas Duta Bangsa

<sup>6</sup>agustina@udb.ac.id

**Abstrak**— Perkembangan teknologi informasi mendorong perubahan pada proses penyelenggaraan konser, termasuk dalam pengelolaan penjualan tiket yang kini banyak dilakukan secara digital. Kondisi tersebut mendorong perlunya sistem e-ticketing yang stabil dan mampu mengelola alur transaksi data secara aman dan terstruktur. Namun, sistem dengan arsitektur tradisional yang menyatu sering kali mengalami kendala berupa kegagalan sistem saat terjadi lonjakan pengunjung (ticket war). Penelitian ini bertujuan untuk menganalisis sekaligus mengimplementasikan arsitektur Three-Tier pada sistem e-ticketing konser berbasis web untuk mengatasi masalah tersebut. Pengembangan sistem dilakukan menggunakan metode Waterfall yang meliputi tahap analisis kebutuhan, perancangan, implementasi, dan pengujian. Sistem dikembangkan menggunakan HTML, CSS, JavaScript, PHP Native, serta MySQL dengan menerapkan pemisahan fungsi ke dalam Presentation Layer (tampilan), Application Layer (logika program), dan Database Layer (basis data). Hasil penelitian menunjukkan bahwa penerapan arsitektur Three-Tier mampu memisahkan fungsi kerja sistem secara jelas, sehingga proses pengelolaan data transaksi menjadi lebih terstruktur dan sistem lebih mudah untuk dikembangkan atau diperbaiki di masa mendatang tanpa risiko merusak basis data.

**Kata kunci**— e-ticketing, konser, Three-Tier, Waterfall, web

**Abstract**— The rapid advancement of information technology has transformed concert management, including ticket sales, which are now increasingly conducted through digital platforms. This development highlights the need for a stable e-ticketing system capable of managing data transactions securely and structurally. However, traditional single-structured systems often face system crashes during high-traffic bottlenecks (ticket wars). This study aims to analyze and implement the Three-Tier architecture in a web-based concert e-ticketing system to address these structural issues. The system was developed using the Waterfall method, which consists of requirement analysis, system design, implementation, and testing. HTML, CSS, JavaScript, PHP Native, and MySQL were used as the main technologies, while the Three-Tier architecture was implemented by separating the system into the Presentation Layer, Application Layer, and Database Layer. The results indicate that the implementation of the Three-Tier architecture successfully separates system functions, making transaction data management more structured and the system easier to maintain or update in the future without risking database corruption.

**Keywords**— e-ticketing, concert, Three-Tier, Waterfall, web

## I. PENDAHULUAN

Digitalisasi telah mengubah berbagai aktivitas masyarakat, termasuk penyelenggaraan konser musik. Informasi mengenai konser yang sebelumnya disebarkan melalui media cetak atau penjualan tiket secara langsung kini beralih ke platform berbasis web. Perubahan tersebut mendorong penggunaan sistem e-ticketing yang memudahkan penyelenggara dalam mengelola data pemesanan secara digital, sementara calon penonton dapat membeli tiket tanpa harus datang

ke lokasi penjualan. Selain memberikan kemudahan dalam proses pemesanan, sistem e-ticketing juga membantu penyimpanan dan pengelolaan data tiket secara lebih rapi sehingga informasi transaksi dapat diakses dengan lebih mudah [1].

Namun, penerapan sistem e-ticketing untuk konser musik memiliki tantangan teknis yang cukup besar, terutama ketika jumlah pembeli meningkat drastis dalam waktu bersamaan (ticket war). Proses ini membutuhkan sistem yang stabil

untuk menangani alur data yang padat, mulai dari pengecekan sisa kuota, proses pembayaran, hingga pengiriman e-ticket secara otomatis. Jika sistem dibangun menggunakan arsitektur tradisional di mana semua komponen program menyatu, beban tinggi pada halaman utama web dapat langsung mengganggu kinerja basis data. Hal itu beresiko menyebabkan kegagalan sistem atau error pada data transaksi [2]. Oleh karena itu, fokus utama dalam pengembangan sistem ini bukan lagi sekedar membuat fitur website yang lengkap, melainkan bagaimana merancang arsitektur perangkat lunak yang dapat memisahkan fungsi kerja sistem agar stabil.

Berbagai penelitian telah mengembangkan sistem e-ticketing untuk mendukung pengelolaan kegiatan secara digital. Salah satu penelitian mengembangkan aplikasi e-ticketing untuk kegiatan webinar dan workshop sehingga proses pendaftaran peserta dan distribusi tiket dapat dilakukan secara daring [1]. Penelitian lain mengintegrasikan payment gateway untuk mendukung proses pembayaran tiket secara online [3], sedangkan teknologi QR Code dimanfaatkan sebagai media verifikasi tiket pada sistem event organizer [2]. Meskipun penelitian-penelitian tersebut berhasil meningkatkan efisiensi, sebagian besar masih berfokus pada pembuatan fitur atau modul aplikasinya saja. Pembahasan mengenai bagaimana struktur arsitektur sistem yang kuat untuk menahan beban transaksi tiket konser masih belum banyak dilakukan.

Selain pengembangan sistem e-ticketing, beberapa penelitian juga telah menerapkan arsitektur Three-Tier pada berbagai aplikasi berbasis web. Salah satu penelitian menerapkan arsitektur tersebut pada aplikasi ticketing layanan helpdesk ATM dengan memisahkan antarmuka pengguna, proses bisnis, dan basis data ke dalam lapisan yang berbeda [4]. Pendekatan serupa digunakan pada pembangunan web service perpustakaan untuk menghasilkan struktur sistem yang lebih terorganisasi [5]. Penelitian lain juga menunjukkan bahwa penerapan arsitektur Three-Tier pada platform pembelajaran berbasis web dapat mempermudah pengelolaan layanan karena

setiap komponen sistem memiliki fungsi yang jelas [6].

Berdasarkan penelitian-penelitian tersebut, penggabungan antara sistem e-ticketing konser dan arsitektur Three-Tier belum banyak dianalisis dari segi efisiensi strukturnya. Kebaruan yang ditawarkan dalam penelitian ini bukan hanya sekedar menerapkan sistem pada domain konser, melainkan melakukan analisis tentang bagaimana pemisahan tugas antara Presentation Layer (tampilan), Application Layer (logika program), dan Database Layer (basis data) dapat bekerja secara optimal pada sistem yang padat transaksi. Oleh karena itu, penelitian ini bertujuan untuk menganalisis dan mengimplementasikan arsitektur Three-Tier menggunakan metode Waterfall. Melalui pendekatan ini, diharapkan dapat dibuktikan secara terstruktur bagaimana pemisahan lapisan tersebut dapat mempermudah pengelolaan data transaksi dan proses pemeliharaan sistem e-ticketing konser berbasis web.

## II. METODOLOGI PENELITIAN

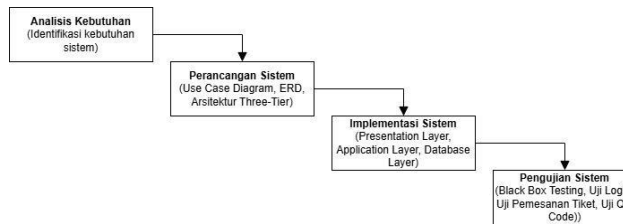
### 2.1 Metode Pengembangan Sistem

Penelitian ini menggunakan metode Waterfall sebagai metode pengembangan sistem. Metode Waterfall dipilih karena proses pengembangannya dilakukan secara bertahap, dimulai dari analisis kebutuhan, perancangan sistem, implementasi, hingga pengujian. Setiap tahapan harus diselesaikan terlebih dulu sebelum melanjutkan ke tahap berikutnya sehingga proses dapat dilakukan secara lebih terstruktur [7].

Pemilihan metode Waterfall disesuaikan dengan karakteristik penelitian, yaitu kebutuhan sistem telah ditentukan sejak awal sehingga proses pengembangan dapat direncanakan secara sistematis. Melalui pendekatan ini, pengembangan sistem dilakukan mulai dari identifikasi kebutuhan pengguna, penyusunan rancangan sistem, implementasi fitur, hingga pengujian untuk memastikan setiap fungsi berjalan sesuai tujuan penelitian [8].

Tahapan Waterfall yang diterapkan meliputi analisis kebutuhan, perancangan sistem, implementasi sistem, dan pengujian sistem.

Urutan proses tersebut ditunjukkan pada Gambar 1.



Gambar 1. Tahapan Penelitian Metode Waterfall

## 2.2 Tahapan Penelitian

Pengembangan sistem e-ticketing konser berbasis web dilakukan mengikuti tahapan pada metode Waterfall. Setiap tahapan memiliki peran yang saling berkaitan sehingga hasil dari satu proses menjadi dasar bagi proses berikutnya. Rangkaian tahapan penelitian dijelaskan sebagai berikut.

### 2.2.1 Analisis Kebutuhan

Tahap analisis kebutuhan berfokus pada identifikasi kebutuhan pengguna dan admin terhadap sistem e-ticketing konser berbasis web. Proses ini bertujuan mengetahui fitur-fitur yang diperlukan agar sistem mampu mendukung aktivitas pengguna maupun admin sesuai kebutuhan. Hasil identifikasi tersebut selanjutnya menjadi acuan dalam proses perancangan dan pengembangan sistem.

#### a. Tabel Kebutuhan Pengguna

| Kebutuhan Pengguna          | Deskripsi                                                        |
|-----------------------------|------------------------------------------------------------------|
| Melihat Daftar Konser       | Pengguna dapat melihat konser yang tersedia                      |
| Melihat Detail Konser       | Pengguna dapat melihat informasi lengkap konser                  |
| Memesan Tiket               | Pengguna dapat mengisi data memilih jenis dan jumlah tiket       |
| Melakukan Checkout          | Pengguna dapat mengisi data pemesanan                            |
| Mengupload Bukti Pembayaran | Pengguna dapat mengirim bukti transfer                           |
| Menerima E-tiket            | Pengguna menerima tiket lewat email                              |
| Mengecheck Tiket            | Pengguna dapat melakukan pengecekan tiket melalui menu cek tiket |

#### b. Tabel Kebutuhan Admin

| Kebutuhan Admin | Deskripsi             |
|-----------------|-----------------------|
| Login           | Admin masuk ke sistem |

Mengelola event

Mengelola pesanan

Verifikasi pembayaran

Mengirim e-ticket

Melihat dashboard

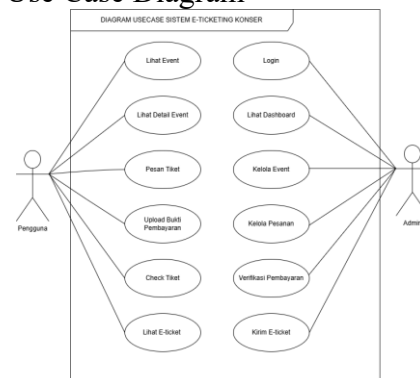
Menambah, mengubah, dan menghapus event  
Melihat data pemesanan tiket

Menyetujui atau menolak pembayaran  
Mengirim tiket ke email pengguna  
Melihat ringkasan data sistem

### 2.2.2 Perancangan Sistem

Tahap perancangan sistem dilakukan berdasarkan hasil analisis kebutuhan yang telah diperoleh sebelumnya. Pada tahap ini disusun rancangan yang menggambarkan alur kerja sistem, hubungan antar data, serta pembagian struktur aplikasi yang akan diterapkan pada sistem e-ticketing konser berbasis web [9]. Perancangan meliputi use case diagram, flowchart, Entity Relationship Diagram (ERD), dan diagram arsitektur Three-Tier.

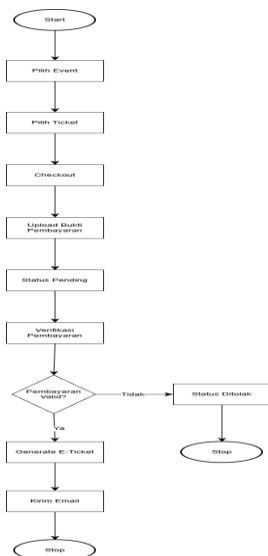
#### a. Use Case Diagram



Gambar 2. Use Case Diagram Sistem E-Ticketing

Use case diagram pada Gambar 2 menggambarkan interaksi antara pengguna maupun admin dengan sistem e-ticketing konser berbasis web. Diagram ini memperlihatkan fungsi-fungsi yang dapat diakses oleh masing-masing aktor sesuai dengan kebutuhan yang telah diidentifikasi pada tahap analisis.

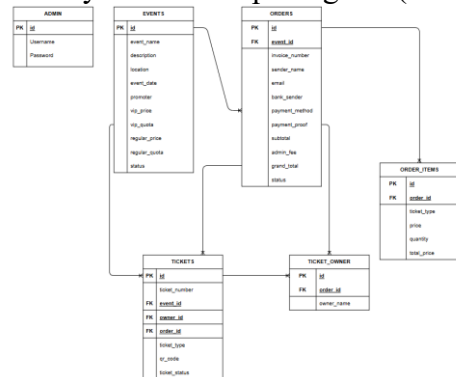
#### b. Flowchart Sistem



Gambar 3. Flowchart Alur Pemesanan Tiket

Flowchart pada Gambar 3 menunjukkan alur proses pemesanan tiket dalam sistem e-ticketing konser berbasis web. Proses dimulai ketika pengguna memilih event dan jenis tiket, kemudian dilanjutkan dengan pembayaran, verifikasi oleh admin, hingga e-ticket berhasil diterima oleh pengguna.

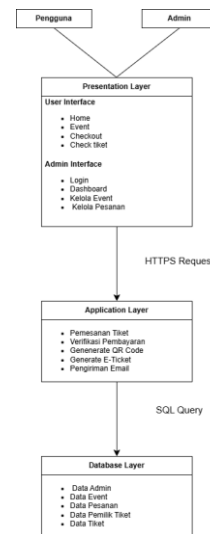
#### c. Entity Relationship Diagram (ERD)



Gambar 4. Entity Relationship Diagram (ERD) Sistem

Entity Relationship Diagram (ERD) pada Gambar 4 menggambarkan hubungan antar entitas yang digunakan dalam sistem e-ticketing konser berbasis web. Penyusunan ERD didasarkan pada kebutuhan data yang telah diidentifikasi sebelumnya sehingga hubungan antar entitas, seperti data event, pesanan, pemilik tiket, dan tiket, dapat dirancang secara terstruktur sebagai dasar pembangunan basis data.

#### d. Diagram Arsitektur Three-Tier



Gambar 5. Diagram Arsitektur Three-Tier Sistem

Diagram arsitektur Three-Tier pada Gambar 5 menunjukkan pembagian sistem ke dalam tiga lapisan utama, yaitu Presentation Layer, Application Layer, dan Database Layer. Pembagian tersebut bertujuan memisahkan proses antarmuka pengguna, logika aplikasi, serta pengelolaan basis data sehingga setiap lapisan dapat menjalankan fungsinya secara terpisah.

#### 2.2.3 Implementasi Sistem

Tahap implementasi merupakan proses pembangunan sistem berdasarkan hasil analisis kebutuhan dan perancangan yang telah disusun sebelumnya. Pengembangan aplikasi menggunakan HTML, CSS, JavaScript, PHP Native, dan MySQL sebagai teknologi utama. Pada tahap ini arsitektur Three-Tier mulai diterapkan melalui pemisahan Presentation Layer, Application Layer, dan Database Layer. Seluruh fitur yang telah dirancang kemudian diintegrasikan sehingga sistem dapat digunakan sesuai kebutuhan pengguna maupun admin.

#### 2.2.4 Pengujian Sistem

Tahap pengujian dilakukan untuk memastikan setiap fitur pada sistem e-ticketing konser berbasis web berfungsi sesuai dengan kebutuhan yang telah ditetapkan. Pengujian menggunakan metode Black Box Testing dengan teknik Equivalence Partitioning karena berfokus pada kesesuaian antara masukan dan keluaran sistem.

tanpa memperhatikan struktur kode program [10]. Pengujian mencakup beberapa fitur utama, antara lain pemesanan tiket, unggah bukti pembayaran, verifikasi pembayaran, pengecekan tiket, serta pengiriman e-ticket.

### III. HASIL DAN PEMBAHASAN

#### 3.1 Hasil Implementasi Arsitektur Three-Tier

Pengembangan sistem menghasilkan aplikasi e-ticketing konser berbasis web yang menerapkan arsitektur Three-Tier sebagai dasar pengelolaan aplikasi. Sistem dibangun menggunakan HTML, CSS, JavaScript, PHP Native, dan MySQL, kemudian dijalankan pada lingkungan lokal sebagai media implementasi dan pengujian.

Penerapan arsitektur Three-Tier membagi sistem ke dalam tiga lapisan, yaitu Presentation Layer, Application Layer, dan Database Layer. Pembagian tersebut bertujuan memisahkan fungsi antarmuka, logika aplikasi, serta pengelolaan basis data sehingga setiap lapisan memiliki tanggung jawab yang berbeda.

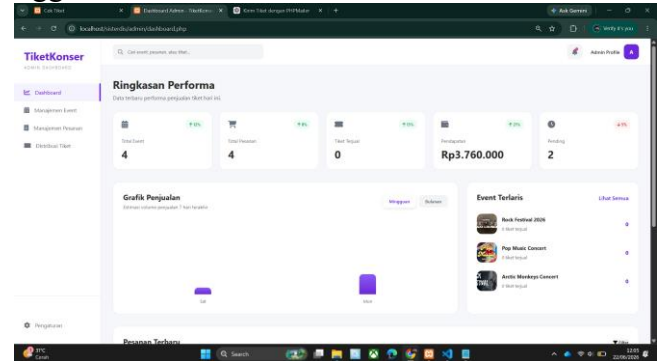
Presentation Layer berperan sebagai antarmuka yang digunakan pengguna saat mengakses sistem melalui browser. Pada lapisan ini pengguna dapat melihat informasi konser, melakukan pencarian, serta melanjutkan proses pemesanan tiket. Tampilan antarmuka dikembangkan menggunakan HTML, CSS, dan JavaScript sehingga dapat memberikan pengalaman penggunaan yang lebih mudah.

Application Layer menangani seluruh proses bisnis yang terdapat pada sistem, seperti pengolahan data pengguna, pemesanan tiket, pengelolaan data konser, serta komunikasi dengan basis data. Seluruh proses tersebut dibangun menggunakan PHP Native sehingga data yang diterima dari antarmuka dapat diproses sebelum disimpan ataupun ditampilkan kembali kepada pengguna.

Database Layer berfungsi sebagai tempat penyimpanan seluruh data yang digunakan dalam sistem, meliputi data konser, pengguna, transaksi, dan informasi pendukung lainnya. Penyimpanan data dilakukan menggunakan MySQL dengan struktur tabel yang telah

dirancang berdasarkan Entity Relationship Diagram (ERD).

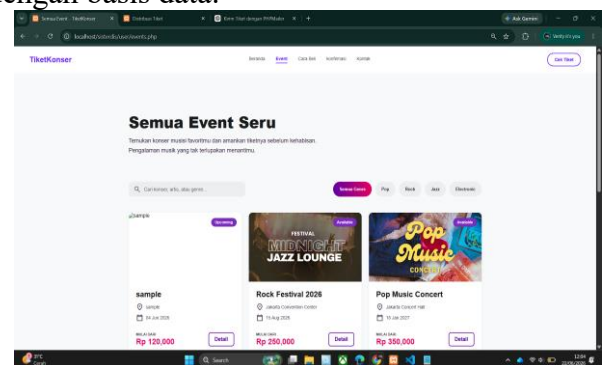
Hasil implementasi sistem ditunjukkan melalui tampilan dashboard admin dan dashboard pengguna.



Gambar 6. Antarmuka Halaman Utama Admin

Dashboard admin pada Gambar 6 merupakan bagian dari Presentation Layer yang digunakan untuk mengelola aktivitas utama pada sistem. Melalui halaman ini administrator dapat mengelola data konser, melakukan verifikasi pembayaran, memantau jumlah tiket yang terjual, serta melihat ringkasan transaksi.

Meskipun seluruh data ditampilkan melalui dashboard admin, proses pengolahan tetap dilakukan oleh Application Layer. Setelah proses selesai, data akan disimpan atau diperbarui pada Database Layer sesuai kebutuhan. Mekanisme tersebut menjaga pemisahan fungsi antar lapisan sehingga antarmuka tidak berhubungan langsung dengan basis data.



Gambar 7. Antarmuka Halaman Utama Pengguna

Berbeda dengan dashboard admin, dashboard pengguna pada Gambar 7 dirancang sebagai media bagi pengguna untuk memperoleh informasi konser dan melakukan pemesanan tiket. Halaman ini menyediakan fitur pencarian,

informasi detail konser, serta proses pembelian tiket yang dapat diakses melalui browser.

Setiap aktivitas yang dilakukan pengguna akan diproses terlebih dahulu oleh Application Layer sebelum data dikirim ke Database Layer ataupun ditampilkan kembali pada halaman web. Dengan mekanisme tersebut, Presentation Layer hanya berfungsi sebagai media interaksi, sedangkan logika aplikasi dan pengelolaan data tetap berada pada lapisan yang sesuai. Pembagian ini membuat pengembangan maupun pemeliharaan sistem menjadi lebih mudah apabila di kemudian hari diperlukan perubahan pada salah satu bagian aplikasi.

### 3.2 Hasil Pengujian Sistem

Pengujian sistem dilakukan menggunakan metode Black Box Testing untuk memastikan seluruh fungsi pada sistem berjalan sesuai kebutuhan pengguna maupun admin. Pengujian ini difokuskan pada pengujian fungsional dengan membagi skenario data valid dan skenario data tidak valid pada fitur-fitur utama sistem.

| Fitur                                   | Hasil yang Diharapkan                                               | Hasil Pengujian     |
|-----------------------------------------|---------------------------------------------------------------------|---------------------|
| Melihat daftar & detail konser          | Sistem menampilkan informasi lengkap konser                         | Berhasil            |
| Pemesanan Tiket (Input Valid)           | Sistem memproses pesanan dan menyimpan data ke basis data           | Berhasil            |
| Pemesanan Tiket (Input Minus)           | Sistem menolak pesanan dan menampilkan pesan peringatan input salah | Berhasil<br>Menolak |
| Checkout                                | Sistem mengarahkan ke halaman pembayaran                            | Berhasil            |
| Upload Bukti Pembayaran (Format Gambar) | File berhasil diunggah ke sistem untuk diverifikasi admin           | Berhasil            |
| Upload Bukti Pembayaran (Format PDF)    | Sistem menolak dokumen dan menampilkan pesan kesalahan format file  | Berhasil<br>Menolak |
| Verifikasi Pembayaran                   | Status pembayaran otomatis diperbarui                               | Berhasil            |
| Pengiriman e-ticket                     | Sistem mengirimkan pesan kode tiket otomatis ke email pengguna      | Berhasil            |
| Cek Tiket (Kode Valid)                  | Sistem menampilkan status tiket "Aktif/Belum Digunakan"             | Berhasil            |
| Cek Tiket (Kode Salah/Palsu)            | Sistem menampilkan pesan bahwa kode tiket tidak ditemukan           | Berhasil<br>Menolak |

Tabel 1. Hasil Pengujian Sistem

Berdasarkan hasil pengujian, seluruh fitur berhasil dijalankan sesuai skenario yang telah ditentukan tanpa ditemukan kesalahan fungsional. Sistem tidak hanya mampu

memproses data yang benar, tetapi juga berhasil memvalidasi dan menolak input yang salah. Hasil tersebut menunjukkan bahwa fungsi-fungsi utama pada sistem telah bekerja sesuai dengan kebutuhan yang dirumuskan pada tahap analisis.

### 3.3 Pembahasan

Penerapan arsitektur Three-Tier pada sistem e-ticketing konser berbasis web menunjukkan bahwa setiap lapisan telah menjalankan fungsi sesuai perannya. Hasil tersebut dapat dilihat pada Tabel 2 yang memperlihatkan implementasi Presentation Layer, Application Layer, dan Database Layer pada sistem yang dikembangkan.

| Layer              | Implementasi pada Sistem     | Analisis                                                                                                                  |
|--------------------|------------------------------|---------------------------------------------------------------------------------------------------------------------------|
| Presentation Layer | Dashboard pengguna dan admin | Menyediakan antarmuka bagi pengguna dan admin untuk berinteraksi dengan sistem tanpa mengakses basis data secara langsung |
| Application Layer  | PHP Native                   | Mengelola proses bisnis seperti pemesanan tiket, verifikasi pembayaran, dan pengiriman e-ticket                           |
| Database Layer     | MySQL                        | Menyimpan data sistem seperti data event, pengguna, dan transaksi                                                         |

Tabel 2. Analisis Penerapan Arsitektur Three-Tier

Presentation Layer diimplementasikan melalui dashboard pengguna dan dashboard admin sebagai media interaksi dengan sistem. Seluruh aktivitas pengguna, seperti melihat informasi konser, melakukan pemesanan tiket, maupun mengelola data konser, dilakukan melalui antarmuka tanpa berhubungan langsung dengan basis data. Pemisahan ini membuat perubahan pada tampilan aplikasi dapat dilakukan tanpa memengaruhi logika aplikasi maupun struktur basis data.

Application Layer diimplementasikan menggunakan PHP Native yang menangani seluruh proses bisnis, mulai dari pemesanan tiket, verifikasi pembayaran, hingga pengiriman e-ticket. Seluruh data yang diterima dari antarmuka diproses terlebih dahulu pada lapisan ini sebelum diteruskan ke basis data. Pendekatan tersebut membuat alur pengolahan data menjadi lebih

terstruktur karena setiap proses dipusatkan pada satu lapisan.

Database Layer menggunakan MySQL sebagai media penyimpanan data konser, pengguna, dan transaksi. Seluruh proses penyimpanan maupun pengambilan data dilakukan melalui Application Layer sehingga basis data tidak diakses secara langsung oleh pengguna. Mekanisme ini membantu menjaga konsistensi pengelolaan data sekaligus mempermudah pemeliharaan apabila terjadi perubahan pada struktur basis data.

Untuk membuktikan penerapan arsitektur Three-Tier secara objektif, dilakukan analisis perbandingan dengan arsitektur tradisional di mana seluruh komponen program menyatu berdasarkan karakteristik sistem e-ticketing konser yang telah dikembangkan. Perbandingan ini ditinjau dari beberapa aspek penting yang dijelaskan pada Tabel 3.

| Aspek                                 | Arsitektur Tradisional (Menyatu)                                                                                                            | Arsitektur Three-Tier (Sistem yang Dibuat)                                                                                                      |
|---------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------|
| Kemudahan Perawatan (Maintainability) | Sulit, karena kode program tampilan (HTML) dan database (MySQL) menyatu, sehingga perubahan kecil bisa merusak bagian program lain.         | Mudah, karena kode tampilan (Presentation) dan database (Database) terpisah total, sehingga perubahan tampilan tidak mengganggu data.           |
| Keamanan Data Transaksi               | Lebih rentan, karena pengguna dari halaman web bisa memicu kesalahan atau mengakses langsung ke bagian basis data.                          | Lebih aman, karena seluruh akses ke basis data wajib melewati Application Layer (PHP Native) sebagai filter keamanan.                           |
| Penanganan Trafik (Ticket War)        | Beban kerja terpusat pada satu tempat, sehingga jika halaman utama crash akibat lonjakan pengunjung, seluruh transaksi langsung mati total. | Beban kerja terbagi, jika halaman utama (Presentation) melambat karena trafik padat, proses pengolahan data di Application Layer tetap terjaga. |

Tabel 3. Pembuktian Penerapan Arsitektur Three-Tier

Berdasarkan analisis perbandingan pada Tabel 3, pemisahan menjadi tiga lapisan fungsional ini tidak sekadar merapikan susunan kode program, melainkan secara efektif meningkatkan keamanan data transaksi dan mempermudah proses pemeliharaan sistem.

Hasil penelitian ini juga sejalan dengan penelitian-penelitian terdahulu yang menunjukkan bahwa arsitektur Three-Tier mampu menghasilkan struktur aplikasi yang

lebih terorganisasi melalui pembagian fungsi pada setiap lapisan. Pada sistem e-ticketing konser yang dikembangkan, konsep tersebut berhasil diterapkan dengan baik pada proses pemesanan tiket, verifikasi pembayaran, serta pengelolaan data, sehingga setiap komponen sistem dapat menjalankan fungsinya sesuai dengan tanggung jawab masing-masing.

#### IV. KESIMPULAN

Penelitian ini berhasil mengimplementasikan dan menganalisis arsitektur Three-Tier pada sistem e-ticketing konser berbasis web dengan membagi fungsi aplikasi ke dalam Presentation Layer, Application Layer, dan Database Layer. Berdasarkan hasil analisis perbandingan structural, penerapan arsitektur ini terbukti secara objektif memberikan keunggulan yang nyata dibandingkan dengan arsitektur tradisional yang menyatu. Keunggulan tersebut ditunjukkan pada aspek kemudahan pemeliharaan (maintainability), di mana perubahan pada halaman antarmuka tidak mengganggu tata kelola data, serta aspek keamanan yang mewajibkan seluruh akses transaksi melewati filter logika program di lapisan aplikasi sebelum masuk ke basis data.

Sementara itu, hasil pengujian menggunakan Black Box Testing membuktikan bahwa seluruh fitur utama pada sistem dapat berfungsi dengan baik. Sistem tidak hanya sukses memproses skenario data yang valid, tetapi juga terbukti melalui pengujian mampu menolak masukan data yang salah seperti jumlah tiket bernilai minus, format filter non-gambar pada bukti pembayaran, dan kode tiket palsu untuk menjaga validasi transaksi. Dengan demikian, penerapan arsitektur Three-Tier terbukti efektif dalam membangun sistem e-ticketing konser yang terstruktur, aman, dan mudah dikembangkan di masa mendatang.

#### UCAPAN TERIMA KASIH

Puji syukur kami panjatkan kepada Tuhan Yang Maha Esa karena atas rahmat dan karunia-Nya, jurnal ini dapat diselesaikan dengan baik. Kami mengucapkan terima kasih kepada dosen pembimbing yang telah memberikan arahan, masukan, dan bimbingan selama proses

penyusunan penelitian ini, serta kepada pihak-pihak yang telah membantu dalam pengumpulan data, penyusunan, dan penyelesaian jurnal ini. Tidak lupa, kami juga berterima kasih kepada teman-teman dan semua pihak yang telah memberikan dukungan, semangat, serta bantuan sehingga penelitian ini dapat terselesaikan dengan baik.

#### REFERENSI

- [1] F. R. Daffa Arif Pardila, "Rancang Bangun Aplikasi E-Ticketing Acara Webinar/Workshop Berbasis Web pada Yayasan Literasi Sains Indonesia," *Jurnal SIKOM (Sistem Informasi Komputer)*, vol. 1, 2024.
- [2] M. ., A. R. D. D. R. Muhammad Fitriani Shousyade Almadina, "Implementasi Teknologi QR Code dalam Sistem E-Ticketing pada Event Organizer," *JITET (Jurnal Informatika dan Teknik Elektro Terapan)*, vol. 13, no. 2830-7062, 2025.
- [3] S. R. N. H. V. V. R. R. Arinda Inansya Nariswari, "Rancang Bangun Sistem E-Ticket Berbasis Payment Gateway," *JSil | Jurnal Sistem Informasi*, vol. 10, no. 2581-2181, 2023.
- [4] K. K. R. I. Nurul Chafid, "Analisis dan Perancangan Aplikasi Ticketing pada Layanan Helpdesk ATM dengan Menggunakan Arsitektur 3 Tier," *Jurnal Satya Informatika*, vol. 2, 2023.
- [5] H. W. Cahyo Prihantoro, "Perancangan Client Server Three Tier Pada Pembangunan Web Service Anggota Perpustakaan Universitas Muhammadiyah Bengkulu," *JTIS*, vol. 2, no. 2614 – 3070, 2019.
- [6] Ade Ismail, "High Availability Moodle dengan Load Balancer pada Three-tier Architecture," *Jurnal Minfo Polgan*, vol. 12, no. 2797-3298, 2023.
- [7] B. F. Muhammad Yusuf, "Sistem Informasi Penjualan Kartu dan Voucher Internet Berbasis Web Menggunakan Metode Waterfall," *BULLETIN OF COMPUTER SCIENCE RESEARCH*, vol. 4, no. 2774-3639, 2024.
- [8] D. H. Temi Ardiansah, "Penerapan Metode Waterfall pada Aplikasi Reservasi Lapangan Futsal Berbasis Web," vol. 1, no. 2962-0635, 2023.
- [9] D. N. Dwi Andini Putri, "Pengembangan Sistem Informasi Pendaftaran Peserta Seminar Berbasis Web Menggunakan Metode Prototype," *IJCIT (Indonesian Journal on Computer and Information Technology)*, vol. 2, 2025.
- [10] M. A. F. A. W. I. S. A. M. I. K. Deny Setiawan, "Pengujian Black Box pada Aplikasi Perpustakaan Berbasis Web Menggunakan Teknik Equivalence Partitioning," *Jurnal Teknologi Sistem Informasi dan Aplikasi*, vol. 3, no. 2654-3788, 2021.