Vol.XX, No. X XXXX 2026 | DOI: 10.33480/pilar.vXXiX.xxxx

DESIGN OF A WEB-BASED EVENT ORGANIZER INFORMATION SYSTEM USING RAPID APPLICATION DEVELOPMENT

Lucky Ardi Adytia[1]*; Ruslan[2]; Luky Kartika Anggraeni[3]; Ami Rahmawati[4]

Information Systems[1,2,3]
Universitas Nusa Mandiri, Jakarta, Indonesia[1,2,3]
http://www.nusamandiri.ac.id[1,2,3]
Luckyardi19@gmail.com[1]*; ucanbita@gmail.com[2]; anggraenilukykartika62@gmail.com[3];
ami.amv@nusamandiri.ac.id[4]
(*) Corresponding Author



The creation is distributed under the Creative Commons Attribution-NonCommercial 4.0 International License.

Abstract*—Event organizer services require integrated management of service information, bookings, payments, and event execution to reduce fragmented records and improve customer access. This study aims to design and implement a web-based event organizer information system as the initial digital product of CigombongEO for CV Naraya. Data were collected through direct observation of the service workflow, interviews with the business owner and operational administrator, and literature review. The system was developed using Rapid Application Development through requirements planning, user design, construction, and cutover. The implementation supports service information, customer accounts, bookings, digital payments, invoices, itineraries, digital invitations, customer support, ratings and reviews, sales reports, role-based access, and audit logs. Black-box testing covered 24 scenarios consisting of 10 customer, 9 administrator, and 5 owner scenarios; all scenarios produced the expected results. Lighthouse testing produced scores of 99 for Performance, 89 for Accessibility, 100 for Best Practices, and 100 for SEO, with First Contentful Paint and Largest Contentful Paint of 0.7 seconds. An initial Pentest Tools scan classified the overall security risk as medium, with no critical or high findings, one medium finding, five low findings, and two informational findings. The resulting system centralizes service and transaction management while maintaining differentiated access for customers, administrators, and owners.*

Keywords: *digital payment, event organizer, information system, rapid application development, web application.*

Abstrak—Layanan event organizer membutuhkan pengelolaan informasi layanan, pemesanan, pembayaran, dan pelaksanaan acara yang terintegrasi agar pencatatan tidak tersebar dan akses pelanggan menjadi lebih mudah. Penelitian ini bertujuan merancang dan mengimplementasikan sistem informasi event organizer berbasis website sebagai produk digital awal CigombongEO untuk CV Naraya. Data diperoleh melalui observasi langsung terhadap alur pelayanan, wawancara dengan pemilik bisnis dan admin operasional, serta studi pustaka. Pengembangan sistem menggunakan Rapid Application Development melalui tahapan requirements planning, user design, construction, dan cutover. Implementasi mencakup informasi layanan, akun pelanggan, pemesanan, pembayaran digital, invoice, itinerary, undangan digital, bantuan pelanggan, rating dan review, laporan penjualan, pengelolaan hak akses, serta audit log. Black-box testing mencakup 24 skenario yang terdiri dari 10 skenario pelanggan, 9 skenario admin, dan 5 skenario owner; seluruh skenario menghasilkan keluaran sesuai harapan. Pengujian Lighthouse menghasilkan nilai Performance 99, Accessibility 89, Best Practices 100, dan SEO 100, dengan First Contentful Paint dan Largest Contentful Paint sebesar 0,7 detik. Pemindaian awal Pentest Tools mengklasifikasikan risiko keamanan keseluruhan pada tingkat medium tanpa temuan critical dan high, satu temuan medium, lima low, serta dua informational. Sistem yang dihasilkan memusatkan pengelolaan layanan dan transaksi dengan tetap membedakan hak akses pelanggan, admin, dan owner.

Kata Kunci: *aplikasi web, event organizer, pembayaran digital, rapid application development, sistem informasi.*

INTRODUCTION

Digital transformation has changed the way service businesses present information, record transactions, and communicate with customers. In

an information system, software is not merely a data repository; it connects users, procedures, and organizational data so that operational activities can be executed and traced in a structured way (Jamas et al., 2022). Web-based systems are particularly relevant for service businesses because users can access information and transaction functions through a browser without installing a dedicated application, while organizations can centralize data processing and reporting. The integration of web services with other internet-based functions also creates opportunities to connect transactions with digital payment mechanisms and supporting services (Sari et al., 2022).

Event organizer businesses depend on accurate information about service packages, schedules, customer requirements, venues, additional services, payments, and event execution. Previous studies have shown that web-based event organizer systems can improve the availability of service information and make booking activities more structured. Rizal et al. (2022) developed a web-based event organizer service ordering system using Rapid Application Development (RAD), while Priyardana (2025) reported a web-based system that supported centralized service and booking management. Pratama and Syastra (2021) also emphasized the role of web applications in managing event organizer information, and H et al. (2021) demonstrated the need to digitize event organizer processes through application-based services. These studies indicate that digitalization can reduce dependence on fragmented manual processes, although the scope of functionality varies between implementations.

Several supporting processes are also important in contemporary event services. Arfian et al. (2022) designed a digital invitation application that allowed users to choose designs, order services, conduct transactions, and distribute invitation links online. Faturahman et al. (2022) integrated service ordering with a payment gateway and communication services, showing that transaction and notification functions can be connected to a service application. Payment gateway selection itself requires consideration of payment methods, cost, security, and integration characteristics (Siahaan & Sianturi, 2024). The growing use of QR-based payments also reinforces the need to maintain clear transaction status and secure records in applications that process customer payments (Febriana et al., 2025).

The research object in this study is CV Naraya, an event organizer business used as the research site and initial implementation partner for CigombongEO. Before implementation, information delivery and service bookings still relied on direct communication or private messages. Customer data, package choices, event schedules, customer requirements, and booking statuses were therefore not managed in a single integrated system. Customers also had to contact the provider to obtain detailed information about packages, prices, facilities, and schedule availability. This situation created the risk of recording errors, schedule conflicts, delayed confirmation, and difficulty tracing booking history. The problem was not limited to the public-facing booking process because payment verification, event rundown or itinerary preparation, digital invitations, customer assistance, and sales reporting also needed connected records.

The solution proposed in this study is a web-based information system that integrates customer and operational workflows while separating access rights according to user roles. Web application development requires systematic software engineering so that requirements, design, implementation, and testing remain connected (Bakri et al., 2024). A service application also needs functional testing to verify that user inputs, system processes, and outputs behave as expected; black-box testing is suitable for this purpose because it focuses on externally observable application behavior (Rahma et al., 2025). Security must also be treated as an ongoing concern, especially when digital transactions are involved, because transaction security and server configuration affect user trust and system reliability (Kristanty, 2024).

CigombongEO is positioned as a digital startup that provides system analysis, implementation, training, maintenance, and customization for event organizer businesses. CV Naraya acts as the initial implementation partner rather than as another name for the startup. This distinction is relevant because the business model can later target micro and small event organizers, venue managers, and supporting event-service providers that need digitalization without maintaining an internal technology team. Business Model Canvas can be used to map the relationship between customer segments, value propositions, channels, activities, resources, partners, costs, and revenue streams in a startup context (Nuratriningrum & Parameswari, 2026), while a pitch deck can communicate the problem, solution, target market, product, and business model to stakeholders (Syafril et al., 2025). The journal article, however, focuses on the implemented information system and its evaluated results.

The novelty of the developed system lies in combining functions that are often studied separately into one operational flow for an event organizer. The application integrates public service information, customer registration and login, booking, digital payment, invoice generation, itinerary management, digital invitations, customer

support, ratings and reviews, service-data management, sales reports, user and role management, and audit logs. Unlike studies that focus primarily on booking or a single supporting feature, the implementation also introduces an Owner role with read-only monitoring access in addition to Customer and Admin roles, and it distinguishes payment gateway transactions from transactions that require administrator verification. The research objective is therefore to design, implement, and evaluate an integrated web-based event organizer information system that addresses the operational problems identified at CV Naraya and can serve as the initial validated product of CigombongEO.

MATERIALS AND METHODS

This research used an applied system-development design focused on solving the operational problems identified at CV Naraya. The research data came from the actual service process and from the application project used in the final implementation. Primary data were obtained through observation and interviews, while supporting concepts and comparisons were obtained through literature review. Observation covered the sequence through which prospective customers obtained service information, selected event types and packages, determined venues and additional services, submitted booking data, made payments, prepared event itineraries, used digital invitations, and received service support. The observation also examined the administrative side of booking management, payment verification, content management, and reporting.

Interviews were conducted with Anisa Wichita as the business owner and Aditya Ramdhani as the operational administrator who understood the daily booking process. The interviews were used to identify operational obstacles, required data, feature priorities, and the expected service flow. The information obtained was translated into functional requirements for Customer, Admin, and Owner actors. Literature review was then used to compare the proposed design with previous event organizer systems, digital invitation systems, payment integration research, application testing, and related digital-business concepts.

The software was developed using the Rapid Application Development approach because the system required iterative refinement based on user feedback and the need to validate working features quickly. RAD has previously been applied to web-based event organizer service development and is appropriate when requirements need to be translated into functional prototypes and refined through repeated interaction with users (Rizal et al., 2022). The process was implemented through four stages. Requirements Planning identified business problems, actor needs, required data, payment processes, and information outputs. User Design translated those requirements into process models, use cases, activity flows, database relationships, and user-interface mockups. Construction implemented the agreed modules and integrated the application with the database and payment services. Cutover evaluated the completed functions, corrected identified problems, and prepared the application for use.

The final implementation specification recorded in the research uses CodeIgniter 3, PHP 8.2, and MySQL/MariaDB, with phpMyAdmin used for database administration. The application is deployed as a web system and integrates digital payment services for supported payment methods. The system was evaluated using three complementary approaches. Functional testing used black-box testing across Customer, Admin, and Owner features. Performance testing used Lighthouse in Google Chrome to measure Performance, Accessibility, Best Practices, SEO, and page-load metrics. Initial security testing used the Website Vulnerability Scanner from Pentest Tools to identify the overall risk level and reported findings. Testing results were interpreted as evidence of whether the implemented functions matched the requirements and as indicators of areas that still require improvement.

RESULTS AND DISCUSSION

Requirements Analysis

The requirements analysis identified two different target perspectives. From the operational perspective, the application is used by three actors: Customer, Admin, and Owner. Customers interact with service information and transaction functions; Admin users manage operational data and validate processes that require administrative action; and Owner users monitor business information without being given the same modification privileges as administrators. From the startup perspective, CigombongEO targets event organizers, venue managers, and supporting event-service providers that require an integrated system but may not have an internal development team. This separation clarifies that the application's direct users are not identical to the broader business customers of the startup.

Customer requirements include account registration, login, profile management, booking management, payment, invoice access, itinerary management, digital invitation management, customer support, and rating and review submission. Admin requirements include login, booking management, payment monitoring and verification, event-type/venue/additional-service management, itinerary and digital invitation

management, support-ticket handling, user and role management, website-content management, and sales-report access. Owner requirements are deliberately narrower: login, booking monitoring, payment-status monitoring, sales-report access, and rating/review monitoring. The Owner role therefore functions as a supervisory role rather than an operational editing role.

Nonfunctional requirements include browser compatibility, authenticated access, role-based authorization, input validation, database consistency, and dependency on an internet connection and valid third-party payment configuration. These requirements are important because the system handles both public service information and internal business data. Role separation also reduces the risk that monitoring users accidentally alter transactional records. The implemented structure reflects the principle that business information should be accessible according to responsibility while sensitive modifications remain controlled.

Table 1 summarizes the final functional scope according to the three application actors.

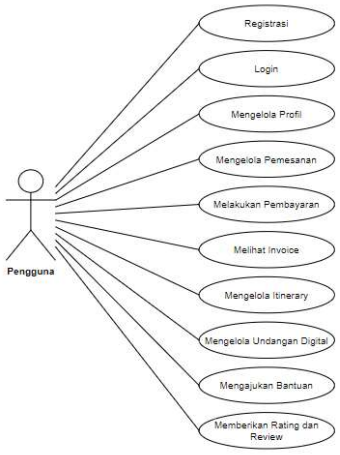
Actor	Main Functional Scope	No.
Customer	Registration; login; profile; booking; payment; invoice; itinerary; digital invitation; support; rating/review	10
Admin	Login; booking; payment verification; services; itinerary/invitation; support; users/roles; content; sales report	9
Owner	Login; booking monitoring; payment monitoring; sales report; rating/review	5

Source: (Research Results, 2026)

Table 1. Functional Scope by Actor

System Design

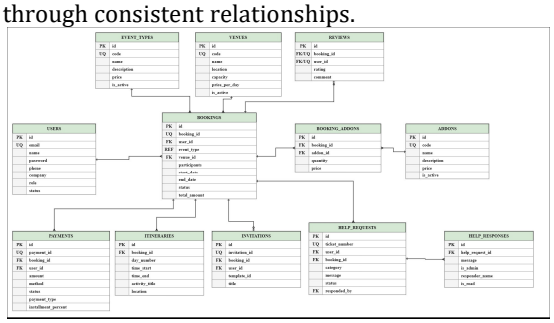
The system design translates the three actor groups into use cases and supporting activity flows. Figure 1 presents the Customer use case diagram, which represents the most extensive interaction path because customers initiate registrations, bookings, payments, itinerary activities, digital invitations, support requests, and reviews. Administrative and owner functions are aligned with the same business objects, allowing one booking record to be traced from customer creation through payment and subsequent operational handling. This arrangement is more comprehensive than an information-only website because transaction and post-booking processes remain connected to the original customer record.



Source: (Research Results, 2026)

Figure 1. Customer Use Case Diagram

The database design is centered on users and bookings. A booking is associated with a user, event type, venue, additional services, payment records, itinerary data, digital invitation data, support tickets, and reviews. Figure 2 summarizes the main entity relationships. The booking entity therefore acts as the operational reference connecting customer identity, selected services, schedule information, and subsequent transaction records. This structure reduces the need to maintain disconnected records for each process and supports reporting because related records can be traced through consistent relationships.



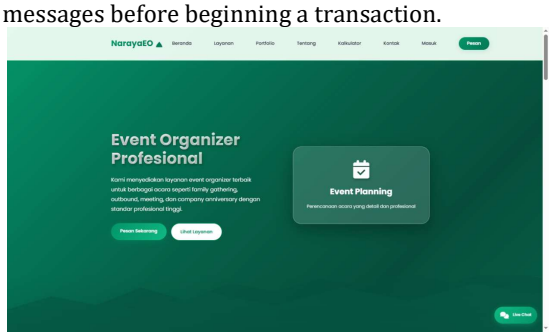
Source: (Research Results, 2026)

Figure 2. Entity Relationship Diagram

The application also supports role and audit information so that system access and important activities can be identified. This is particularly relevant for administrative functions such as payment verification and booking-status updates. Previous event organizer studies have demonstrated the usefulness of centralized booking data (Rizal et al., 2022; Priyardana, 2025), while the present implementation extends the flow by linking booking data with payment, event-preparation information, customer-support records, and monitoring access. The integration is intended to make operational information easier to trace from one system instead of requiring separate communication channels or spreadsheets.

Implementation

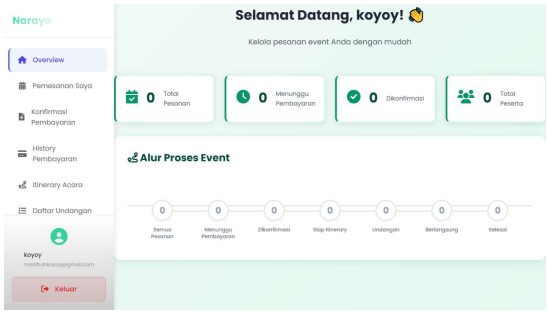
The implemented website uses the CigombongEO system as a technology product while the public interface shown during initial implementation adopts the NarayaEO identity because CV Naraya is the first implementation partner. Figure 3 shows the implemented public home page. The page provides navigation to services, portfolio information, company information, a calculator, contact access, login, and booking. This public layer is intended to reduce the need for prospective customers to repeatedly ask for basic service information through private messages before beginning a transaction.



Source: (Research Results, 2026)

Figure 3. Public Home Page

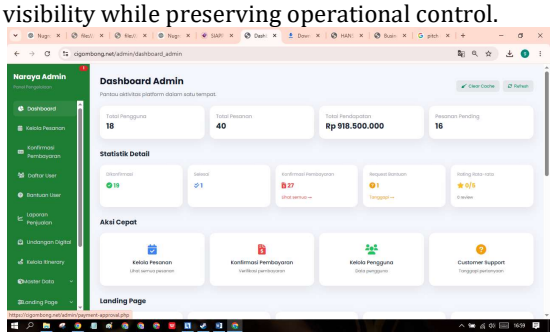
After authentication, customers access a dashboard that summarizes bookings and the progression of event-related activities. Figure 4 shows the implemented Customer dashboard, which presents total bookings, payments awaiting completion, confirmed bookings, participant totals, and a process flow from booking through itinerary, invitation, event execution, and completion. The dashboard also provides direct access to payment history, itinerary, invitations, and profile functions. This organization is consistent with the requirement that customers should be able to trace their own transactions instead of depending on manual status confirmation.



Source: (Research Results, 2026)

Figure 4. Customer Dashboard

The Admin dashboard in Figure 5 functions as the operational control point. It summarizes users, bookings, revenue, payment conditions, support requests, ratings, and recent activities. Administrative menus connect to booking management, payment verification, service master data, support handling, user and role management, content management, and reports. The Owner role receives monitoring access to selected business information without inheriting the same editing privileges. This differentiation is important because the system is intended not only to execute transactions but also to provide management visibility while preserving operational control.



Source: (Research Results, 2026)

Figure 5. Admin Dashboard

One of the most important revisions during system alignment concerned the payment flow. The final application distinguishes the type of payment from the payment method. A customer first selects the booking with an outstanding balance, determines whether the payment is full or staged according to the available transaction flow, and then selects the payment method. Transactions handled through the payment gateway follow the response returned by the payment service. Transactions that require payment proof can remain pending until an Admin reviews the evidence and confirms or cancels the transaction. This design avoids the inaccurate assumption that every transaction is automatically confirmed by a payment gateway. The approach is consistent with research emphasizing the importance of integrating payment data, payment status, and transaction records into web-based service systems (Faturahman et al., 2022) and with payment-gateway studies that highlight the need to consider method characteristics and integration behavior (Siahaan & Sianturi, 2024).

Digital invitations and itinerary functions are linked to booking records so that post-booking preparation is not separated from transaction data. This differs from a standalone digital invitation application such as the type studied by Arfian et al. (2022), because the invitation is treated as one part of an event-service workflow rather than as an isolated product. The application also provides support tickets and rating/review functions, allowing customer questions and post-service feedback to remain associated with user and booking information. These additions support a broader interpretation of event organizer digitalization in which service information, transaction processing, event preparation, and feedback are part of one traceable process.

Testing and Evaluation

Functional testing was performed against the finalized implementation and the aligned use cases,

mockups, and program behavior. A total of 24 black-box scenarios were executed: 10 Customer scenarios, 9 Admin scenarios, and 5 Owner scenarios. All scenarios produced the expected results and were recorded as valid, resulting in a 100% functional success rate for the tested cases. The Customer scenarios covered registration, login, profile management, booking, payment, invoice access, itinerary, digital invitations, support requests, and ratings/reviews. Admin testing covered authentication, booking management, payment monitoring and verification, service data, itinerary/invitations, customer support, user/role management, web content, and sales reports. Owner testing verified authenticated read-only access to bookings, payments, sales reports, and ratings/reviews.

Actor	Scenarios	Valid	Success
Customer	10	10	100%
Admin	9	9	100%
Owner	5	5	100%
Total	24	24	100%

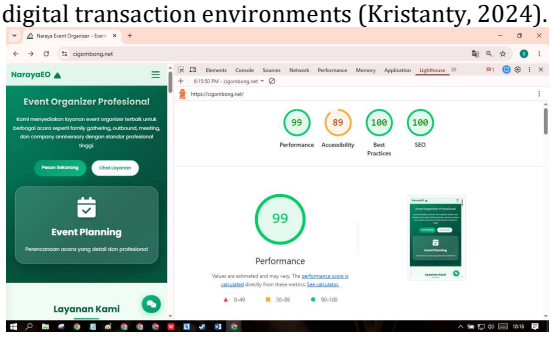
Source: (Research Results, 2026)

Table 2. Black-Box Testing Summary

The black-box results are summarized in Table 2. The test design included both normal and invalid-input conditions where relevant, such as incorrect credentials, invalid or incomplete data, duplicate data, unauthorized changes, and payment-status handling. The payment scenarios were specifically aligned with the implemented program: gateway transactions depend on gateway responses, while transactions that require manual evidence can be verified by an Admin. This alignment is important because a test case should evaluate the behavior that actually exists in the application rather than a conceptual flow that has not been implemented. The successful results indicate that the tested features behave consistently with the final functional requirements. Black-box testing is appropriate for this type of validation because it evaluates observable input-output behavior without requiring the tester to inspect internal code structures (Rahma et al., 2025).

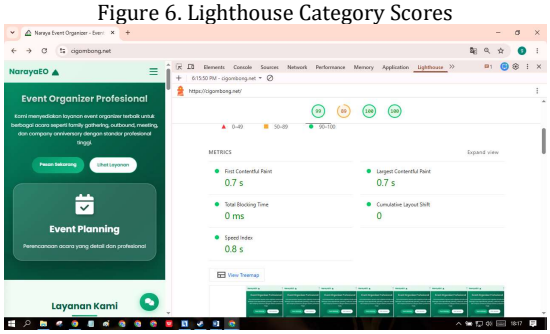
Performance testing using Lighthouse produced a Performance score of 99, an Accessibility score of 89, a Best Practices score of 100, and an SEO score of 100. The detailed metrics were First Contentful Paint of 0.7 seconds, Largest Contentful Paint of 0.7 seconds, Total Blocking Time of 0 milliseconds, Speed Index of 0.8 seconds, and Cumulative Layout Shift of 0. These measurements indicate that the tested home page displayed initial and major content quickly, experienced no meaningful blocking time in the measurement, and maintained stable layout positioning during loading. The lower Accessibility score, compared with the other categories, shows that interface accessibility still requires attention. The final research document identifies contrast, form labels, interactive-element naming, keyboard navigation, and information structure as areas that should be reviewed further.

Initial security assessment was conducted with the Website Vulnerability Scanner from Pentest Tools against the implemented website. The scan executed 40 checks in 26 seconds and reported no critical findings, no high findings, one medium finding, five low findings, and two informational findings, resulting in an overall medium-risk classification. The medium finding related to the PHP version used by the server, indicating that server software and configuration should be reviewed and updated to a stable version that includes relevant security fixes. The scan was a lightweight assessment and did not comprehensively test several critical vulnerability classes such as SQL Injection, Cross-Site Scripting, Command Injection, and XML External Entity attacks. Consequently, the result must be interpreted as an initial security indication rather than proof that the website is free from all vulnerabilities. This interpretation is consistent with the need for continued security attention in digital transaction environments (Kristanty, 2024).



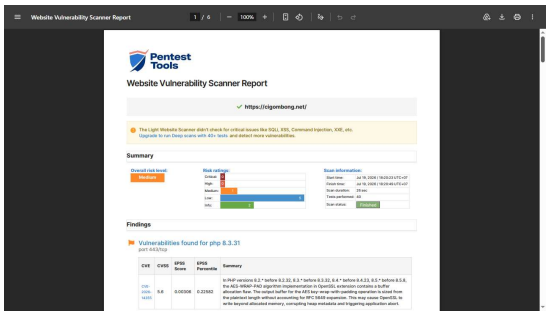
Source: (Research Results, 2026)

Figure 6. Lighthouse Category Scores



Source: (Research Results, 2026)

Figure 7. Lighthouse Performance Metrics



Source: (Research Results, 2026)

Figure 8. Initial Security Scan

Figures 6–8 combine the performance and security evidence recorded in the research. Taken together, the test results show a useful distinction between functional readiness and areas for continuing improvement. Functionally, all 24 black-box scenarios were valid. Performance was strong on the tested page, with near-perfect Lighthouse categories except Accessibility. Security, however, still requires server updates and deeper testing. This balance is important in evaluating an operational web application because functional correctness alone does not guarantee accessibility or security quality. The evaluation therefore provides a practical baseline for the next development iteration rather than treating deployment as the end of the software lifecycle.

Discussion

Compared with prior research, the main contribution of this implementation is the breadth of integration within a single event organizer workflow. Rizal et al. (2022) and Priyardana (2025) demonstrate the value of web-based systems for event service information and booking. The present system retains those core functions but also connects payment status, invoice records, itinerary preparation, digital invitations, customer support, reviews, reporting, user roles, and audit logs. Pratama and Syastra (2021) and H et al. (2021) similarly support the relevance of web or application-based event organizer digitalization, while this study places the functions into an implementation that is directly validated with an initial partner.

The payment and invitation modules also show how specialized features from previous studies can be absorbed into a wider operational model. Arfian et al. (2022) focused on digital invitation ordering and distribution, whereas the current application links the invitation to an existing booking. Faturahman et al. (2022) demonstrated the benefit of connecting service ordering with payment integration; the present implementation likewise integrates payments but preserves a separate administrative verification path for transactions that require evidence. QR-based and gateway payments require consistent transaction status and records, a concern also emphasized in payment-related literature (Febriana et al., 2025; Siahaan & Sianturi, 2024).

The system architecture and testing also support maintainable operational use. Wistika et al. (2023) showed the relevance of web frameworks for service-ordering applications, while Bakri et al. (2024) emphasized structured software engineering for efficient development. In this study, RAD supported iterative alignment between requirements, mockups, activity flows, implementation, and testing. The final revision process was especially important for ensuring that design artifacts no longer described features absent from the program. For example, the login mockup was aligned with the implemented credential flow, payment diagrams were adjusted to represent both gateway and administrator-verified transactions, and black-box scenarios were reconciled with the actual 24 test cases. This reduces the gap between documentation and software behavior.

From a business perspective, CigombongEO is not positioned as a replacement for CV Naraya. It is the technology vendor and digital startup concept, while CV Naraya is the research site and initial implementation partner. This positioning allows the validated product to become a reference for future implementation in other micro and small event organizer businesses, venue managers, or supporting event-service providers. The business-model discussion in the source research maps value through implementation, training, maintenance, and customization, consistent with the role of Business Model Canvas in clarifying customer and value relationships (Nuratriningrum & Parameswari, 2026). The pitch-deck output supports communication of the problem, solution, target market, product, and business model to stakeholders (Syafril et al., 2025), but the technical evaluation in this article remains limited to the implemented application and its recorded tests.

The results also reveal limitations. The Lighthouse evaluation represents the tested page and environment rather than every possible device, browser, network condition, or application page. The Accessibility score of 89 indicates that usability for diverse users should not be considered complete. The security scan is intentionally preliminary and does not replace comprehensive penetration testing. The system also depends on third-party payment configuration and internet connectivity. Future development should therefore prioritize accessibility improvements, updated PHP and server configuration, regular backups, deeper security testing, and automated notifications for booking, payment, itinerary, invitation, and support-status changes. These improvements follow directly from the evaluation findings rather than from unverified assumptions about the system.

CONCLUSION

The research produced an integrated web-based event organizer information system for the initial implementation at CV Naraya and as the first validated digital product of CigombongEO. The system centralizes service information, customer accounts, bookings, digital payments, invoices, itineraries, digital invitations, customer support, ratings and reviews, service data, sales reports, user and role management, and audit logs. Access is separated among Customer, Admin, and Owner roles so that transaction execution, operational management, and supervisory monitoring can be handled according to responsibility. Functional evaluation recorded a 100% success rate across 24 black-box scenarios, consisting of 10 Customer, 9 Admin, and 5 Owner scenarios. Lighthouse testing showed strong performance with scores of 99 for Performance, 89 for Accessibility, and 100 for both Best Practices and SEO, while the detailed load metrics showed fast and stable rendering in the tested condition. Initial security scanning classified the overall risk as medium because one medium finding remained even though no critical or high findings were reported. Therefore, the system can support more centralized, documented, and traceable service and transaction management, but accessibility refinement, server updates, backups, deeper security assessment, and automated notifications remain important directions for further development.

REFERENSI

Arfian, A. B., Immasari, I. R., & Rini, A. S. (2022). Perancangan aplikasi undangan digital berbasis website menggunakan CodeIgniter 4. Jurnal Manajemen Informatika Jayakarta, 2, 1–14.

Bakri, S. N., Irwan, M., Nasution, P., Islam, U., & Sumatera, N. (2024). Penerapan metodologi rekayasa perangkat lunak untuk efisiensi pengembangan sistem. JSITIK, 3(1), 53–66.

Faturahman, E. T., Hayuhardhika, W., Putra, N., & Purnomo, W. (2022). Pembangunan sistem informasi pemesanan jasa foto berbasis web menggunakan REST API pada Heroe Photography. Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer, 6(12), 5693–5702.

Febriana, A. I., S, A. E., & Harimurti, F. (2025). Studi literatur: Penerapan QRIS sebagai payment gateway dalam mendukung efisiensi sistem informasi akuntansi pada UMKM. Jurnal Riset Rumpun Ilmu Ekonomi, 4, 73–88.

H, H., Sarjan, M., & Muammar. (2021). Sistem informasi event organizer berbasis Android. Peqguruang Conference Series, 3.

Jamas, A. R., Anggraeni, D., Mubarak, M. A. N., & Aribowo, D. (2022). Perangkat lunak komputer. Jurnal Sains dan Teknologi, 2(2).

Kristanty, D. N. (2024). Tren dan tantangan keamanan bertransaksi dengan QRIS dalam era transformasi sistem pembayaran digital. Jurnal Syntax Admiration, 5(10), 3923–3933.

Nuratriningrum, A., & Parameswari, R. (2026). Pelatihan kewirausahaan tentang Business Model Canvas bagi siswa sekolah menengah kejuruan. 6(1), 89–102. https://doi.org/10.31253/ad.v6i1.4313

Pratama, M. I., & Syastra, M. T. (2021). Analisis dan perancangan sistem informasi event organizer berbasis web di Antique Salon dan Pelaminan. Jurnal Comasie, 3.

Priyardana, R. (2025). Sistem informasi event organizer berbasis website pada Seven Organizer dengan metode Waterfall. Jurnal Ilmu Komputer dan Informatika, 1(3), 72–78.

Rahma, N., Zahra, D., Putri, A. W., Azzahra, K. S., & Widhiwipati, D. R. (2025). Pengujian pada website SmartPetsCare untuk layanan. Jurnal Mahasiswa Teknik Informatika, 9(1), 378–383.

Rizal, K., Alkhalifi, Y., Fibriany, F. W., & Rachmawaty, Z. (2022). Rancang bangun sistem informasi pemesanan jasa event organizer berbasis website menggunakan metode RAD. Jurnal Teknik Komputer AMIK BSI, 8(1). https://doi.org/10.31294/jtk.v4i2

Sari, I. P., Batubara, I. H., Basri, M., & Hazidar, A. H. (2022). Implementasi Internet of Things berbasis website dalam pemesanan jasa rumah service teknisi komputer dan jaringan komputer. Jurnal Ilmu Bersama, 0–6.

Siahaan, R. A., & Sianturi, R. A. (2024). Analisis perbandingan payment gateway untuk sistem berbasis aplikasi. JTTIK, 11(2). https://doi.org/10.25126/jtiik.20241127680

Syafril, Sofia, Mohdari, Fahmi, M., & Fadilah, N. R. (2025). Pelatihan kewirausahaan: Membuat proposal bisnis dan pitch deck untuk menarik minat investor. Jurnal Inovasi Pengabdian Masyarakat, 2(3), 221–230.

Wistika, K. P., Pramana, D., & Setiasih, N. W. (2023). Sistem informasi pemesanan jasa fotografi pada Julian Photography menggunakan framework Laravel. Jurnal Informasi dan Teknologi, 5(1), 240–249. https://doi.org/10.37034/jidt.v5i1.301.