Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 10, Issue 1, Januari 2026
DOI : https://doi.org/10.33395/sinkron.v10i1.15600
e-ISSN : 2541-2019
p-ISSN : 2541-044X

Implementing a Payment Gateway in the Mount Slamet Hiking Ticketing System

Faishal Bahy, Dani Arifudin, M. Syaiful Amin
Information Technologi, Faculty of Computer Science, Amikom Purwokerto University
faishalbahy@email.com, daniarif@amikompurwokerto.ac.id, syaifulamin@amikompurwokerto.ac.id

Submitted :Nov 20, 2025 | **Accepted** : Dec 19, 2025 | **Published** : Jan 04, 2026

Abstract: Background—Manual booking at Indonesian hiking basecamps often leads to quota mismatches, delayed payment verification, and operational errors under concurrent reservations. **Objective—**This study develops and evaluates a Mount Slamet web-based e-ticketing system integrating Midtrans to ensure consistent payment-state transitions and real-time quota enforcement via signature-verified, idempotent webhooks. **Methods—**Using Agile–Scrum and business-process modeling, we implemented a user booking flow (route selection, identity input, payment initiation, e-ticket issuance) and an admin dashboard for monitoring and check-in. We performed end-to-end black-box and cross-browser validation. Technical evaluation measured webhook propagation (n = 30), response time of core endpoints (n = 50 per operation), and quota-synchronization accuracy during 20 parallel booking attempts. Usability was assessed with the System Usability Scale (SUS; n = 13). **Results—**Webhook updates propagated in 2.41 s on average (SD 0.68; min 1.32; max 3.79) with zero duplicate or missed events. Quota synchronization achieved 100% accuracy, and quotas were restored within <4 s after expired/canceled transactions. Mean endpoint response times stayed below 250 ms, and all core scenarios worked across major browsers. The system obtained a SUS score of 75.0, indicating acceptable (OK–Good) usability. **Conclusion—**The proposed event-driven architecture improves transactional reliability and reduces manual verification workload, offering a practical blueprint for quota-restricted hiking or tourism services; future work includes 500–1,000 user load testing, QR-based gate scanning, and multi-site/multi-gateway deployment studies.

Keywords: e-ticketing, information system, Midtrans, payment gateway, System Usability Scale (SUS), webhooks

INTRODUCTION

Mount Slamet is an active stratovolcano and the highest peak in Central Java (≈3,428 m above sea level). As an active mountain destination, hiking operations require careful control of quota, safety screening, and hiker verification, particularly during periods of increased visitation (Global Volcanism Program, 2025). In basecamp operations, quota enforcement is not only an administrative requirement but also a practical mechanism to reduce congestion, manage risk, and maintain orderly service flow when demand fluctuates (Banyumas Ekspres, 2025).

Despite the growing demand for hiking permits, many basecamps in Indonesia still rely on manual workflows such as on-site registration, chat-based ordering, bank-transfer confirmation, and manual gate validation. These practices frequently create long queues, uncertainty about remaining quota, delayed payment verification, and inconsistent records across booking, payment, and check-in stages. Such operational friction can degrade the visitor experience and increase the burden on basecamp staff during peak demand.

Prior domestic studies report that digitizing booking processes can improve transparency and accelerate service delivery in hiking and tourism environments (Christi, Putra, & Hanggara, 2023; Dwi Jayanto, Kumalasari Niswatin, & Kasih, 2021; Wiyono & Fachrie, 2024). However, many implementations emphasize interface digitization and administrative convenience while leaving payment confirmation reliability and quota synchronization under concurrent requests insufficiently addressed.

International studies similarly indicate that electronic ticketing improves data accuracy, reduces human error, and streamlines validation processes in service operations (Lübeck, Wittmann, & Flores Battistella, 2012; Subramanya, Kermanshachi, Pamidimukkala, & Loganathan, 2023). In tourism and digital services, adoption is strongly influenced by perceived usefulness, ease of use, and trust, making usability a key determinant of acceptance (Islam, 2023). Usability-centered evaluations further suggest that user satisfaction depends on interface

*name of corresponding author

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 10, Issue 1, Januari 2026
DOI : https://doi.org/10.33395/sinkron.v10i1.15600
e-ISSN : 2541-2019
p-ISSN : 2541-044X

clarity and perceived efficiency, especially during confirmation and validation steps where errors can disrupt transactions (Arijaya et al., 2020; Ferreira, Brändle, Widhalm, & Olaverri-Monreal, 2018).

Nevertheless, standardized usability evidence—particularly using the System Usability Scale (SUS)—remains limited in hiking e-ticketing contexts. From a systems perspective, payment confirmation in modern web applications is commonly propagated through asynchronous callbacks (webhooks). Webhooks may be delivered more than once due to retries, timeouts, or unstable networks, which can complicate transactional consistency.

Without explicit safeguards, webhook-driven updates can trigger duplicate state transitions, inconsistent transaction records, or incorrect quota deductions. Event-driven reliability principles emphasize verifiable event sources and idempotent processing to prevent duplicate updates and maintain consistent state propagation (Kreps, Narkhede, & Rao, 2011; Velepucha & Flores, 2023). In a payment workflow, signature verification strengthens integrity by ensuring incoming events originate from the payment provider and have not been tampered with, while idempotent handling ensures repeated delivery of the same event does not create conflicting transaction states (Lazzari & Farias, 2023).

Based on the above, three research gaps are identified. First, prior studies on tourism e-ticketing and hiking booking systems rarely implement signature-verified and idempotent webhook workflows for payment confirmation, even though such mechanisms are central to reliable payment-state transitions in asynchronous environments. Second, although usability is widely acknowledged as an adoption factor, few hiking or tourism ticketing systems report standardized usability outcomes such as SUS, limiting comparability and evidence of user acceptance. Third, existing research seldom integrates real-time quota management, payment-state consistency through event-driven webhook processing, and formal usability evaluation within a single solution tailored to quota-restricted mountain-hiking operations.

This study addresses these gaps by developing a web-based hiking e-ticketing system for Mount Slamet that integrates a payment gateway, enforces real-time quota synchronization during concurrent booking attempts, and maintains consistent transaction states through signature-verified, idempotent webhook handling. The system issues unique ticket identifiers after successful payment to support end-to-end traceability across booking, payment, and check-in. Accordingly, this paper is guided by the following research question: How can a web-based hiking ticketing system maintain payment-state consistency and real-time quota accuracy under concurrent booking attempts using signature-verified, idempotent webhook processing?

The remainder of this paper presents related work, the proposed system design and implementation, evaluation results, and implications for scalable quota-restricted tourism services. Table 1 summarizes the positioning of this study relative to typical prior works.

Table 1 Positioning of this study against prior works.

Aspect	Prior works (typical focus)	This study
Primary scope	Hiking/tourism booking digitization	Hiking e-ticketing with payment-state reliability and quota enforcement
Payment confirmation	Basic integration; limited handling of retries/duplicates	Signature-verified, idempotent webhook processing for consistent payment states
Quota management	Basic quota deduction	Real-time quota synchronization under concurrent booking attempts
Usability evidence	Informal/limited measurement	Standardized SUS evaluation with technical metrics

LITERATURE REVIEW

Prior studies on e-ticketing for tourism destinations and hiking services report that shifting from manual registration to web- or app-based booking improves information transparency, accelerates service delivery, and reduces recording errors. Implementations in regional tourism contexts emphasize online reservations and more timely availability updates (Wiyono & Fachrie, 2024). In mountain-hiking settings, web-based e-ticketing designs highlight more orderly booking and ticket validation, while cross-browser compatibility is identified as a practical factor for acceptance (Christi et al., 2023). Related queue-management approaches at other tourist sites similarly aim to prevent bottlenecks at entry points (Dwi Jayanto et al., 2021).

On mobile platforms, Android-based modeling of hiking e-ticketing systems reinforces operational requirements such as quota handling, identity recording, and ticket validation at arrival (Suwarno, Hamimi, & Edora, 2023). However, many published implementations primarily emphasize interface digitization and administrative convenience. As a result, payment-state transition reliability and quota synchronization under concurrent requests are often not discussed in detail.

*name of corresponding author

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 10, Issue 1, Januari 2026
DOI : https://doi.org/10.33395/sinkron.v10i1.15600
e-ISSN : 2541-2019
p-ISSN : 2541-044X

Payment gateway integration broadens cashless channels (virtual accounts, cards, QRIS, and e-wallets) and enables automated status notifications through webhooks, which can accelerate confirmation and simplify reconciliation. MSME case studies report reliable verification and practical integration into web applications (Fatman et al., 2023; Nasri et al., 2021). In quota-restricted hiking services, payment states (e.g., pending, settlement, expire, cancel) can be mapped to quota reservation, deduction, restoration, and ticket-ID issuance for gate validation. Transactional booking design references using structured flow modeling (e.g., DFD) remain relevant for organizing state transitions and audit trails in service systems (Muliadi, Andriani, & Irawan, 2020).

From a reliability and security perspective, webhook delivery may occur more than once due to retries, timeouts, or unstable networks. Without explicit safeguards, repeated events can trigger duplicate state transitions, inconsistent transaction records, or incorrect quota deductions. Event-driven reliability principles emphasize verifiable event sources and idempotent processing so that repeated delivery of the same event results in a single consistent update (Kreps, Narkhede, & Rao, 2011; Velepucha & Flores, 2023). In practical payment workflows, signature verification functions as an integrity control by ensuring incoming events originate from the payment provider and have not been tampered with, while idempotent handling prevents duplicate deliveries from creating conflicting transaction states and quota counters (Lazzari & Farias, 2023).

In development practice, Agile (Scrum/ASD) is widely used for transactional web systems because it supports iterative delivery and requirement changes (Nova, Widodo, & Warsito, 2022). Evidence across inventory and transactional domains reports improved scope control and measurable sprint outputs (Badiwibowo Atim, 2024; Hendra, Wahyuningsih, & Mahendrasusila, 2024; Widiarta, Mulyanto, & Sutrianto, 2023), including real-time status services such as bus scheduling and allocation (Lubis, Zakir, & Sembiring, 2025). While some hiking-related work chooses RAD for rapid prototyping and early validation (Aldho Febrian Harahap et al., 2024), evaluation practices in API-driven development commonly include black-box testing and user-satisfaction surveys to ensure consistent behavior across devices and browsers (Sulistyorini, Sova, & Ramadhan, 2022). This evaluation pattern is pertinent for assessing core functions—from booking and payment to ticket-ID issuance and gate-side ticket validation.

Usability evidence is commonly reported using the System Usability Scale (SUS), a standardized 10-item instrument that produces a 0–100 score and remains reliable for small samples (Kurniawan, Nofriadi, & Nata, 2022). SUS scores are commonly interpreted using acceptability ranges, grade scales, and adjective ratings to improve comparability across studies (Kurniawan et al., 2022; Yoga Pudya Ardhana, 2022). Accordingly, Table 2 summarizes representative prior works and clarifies the gap addressed by integrating quota enforcement, payment-state reliability (signature-verified idempotent webhooks), and standardized usability evaluation in one hiking e-ticketing solution.

Table 2 Summary of prior studies and identified gaps (concise).

Prior study (year)	**Context / system type**	**Payment integration (reported)**	**Quota management (reported)**	**Reported limitation (from scope)**	**Gap addressed by this study**
Wiyono & Fachrie (2024)	Web booking (tourism)	Not specified	Reservation and availability	Limited focus on payment reliability and concurrency	Adds payment-state consistency and quota handling under concurrency, with SUS
Christi et al. (2023)	Web e-ticketing (hiking)	Not specified	Not specified	Focus on booking and validation; webhook reliability not discussed	Adds signature-verified and idempotent webhook processing, quota enforcement, and SUS
Dwi Jayanto et al. (2021)	Queue/service flow (tourism)	Not specified	Not specified	Focus on bottleneck reduction, not payment-state consistency	Adds end-to-end coupling across booking, payment, ticketing, and check-in
Suwarno et al. (2023)	Android modeling (hiking)	Not specified	Quota, identity, and validation	Limited detail on payment event reliability	Adds provider-verified webhook events, idempotency, and concurrency control

*name of corresponding author

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 10, Issue 1, Januari 2026
DOI : https://doi.org/10.33395/sinkron.v10i1.15600
e-ISSN : 2541-2019
p-ISSN : 2541-044X

METHOD

Research Procedure

This study followed an empirical software-engineering procedure that began with on-site observation of basecamp operations to identify operational issues and constraints. Functional requirements were elicited and formalized into business processes and UML artifacts to ensure traceability from operational rules to implementation. The system architecture was then designed with an event-driven payment integration and idempotent webhook handling to maintain consistent payment-state transitions and quota integrity.

A working prototype was implemented and used as the empirical object for evaluation. The prototype was evaluated using end-to-end black-box functional scenarios, cross-browser compatibility checks, technical measurements of webhook latency and quota synchronization under concurrent access, and a usability assessment using the System Usability Scale (SUS). This procedure positions the system not only as a software artifact but also as a validated instrument for analyzing transactional reliability, data consistency, and user experience in an operational setting.

System Development Method

Agile–Scrum was selected because its iterative and adaptive nature aligns with systems that require continuous refinement during integration with external services such as payment gateways. Prior studies indicate that Scrum is widely used for web-based information systems and is effective for handling dynamic requirements, user feedback, and rapid prototyping (Anwar, Kurniawan, Rahman, & Ani, 2020; Nova, Widodo, & Warsito, 2022). In this study, the Agile approach enabled progressive validation of quota rules, payment flows, and user interactions through short sprints, ensuring that each increment was reviewed and refined before advancing.

Iterative development also supported the stabilization of the event-driven payment integration and idempotent state handling, which are critical for maintaining quota and transaction consistency under concurrent bookings (Badiwibowo Atim, 2024; Lubis et al., 2025; Widiarta et al., 2023).

Data Sources

Functional requirements were elicited via observation of basecamp operations and structured discussions with the operator. Test data included booking scenarios across trail, date, and party-size variations; transaction logs in a sandbox environment (Snap initiation; pending/settlement/expire/cancel statuses; webhook notifications); and a device/browser matrix for cross-browser validation (Chrome, Edge, Firefox, and Safari on iOS where applicable). Design artifacts—such as the Use Case diagram and the Midtrans integration flow—were compiled as references for implementation and testing (Lubis et al., 2025).

For usability evaluation (SUS), the target sample size was 12–15 participants representing prospective hikers who had hiked or intended to hike Mount Slamet and were familiar with smartphones and web applications. The realized sample consisted of 13 respondents. Testing was conducted on participants' personal devices in a test class under a shared network.

System Architecture

The system adopts a layered web architecture combined with an event-driven payment integration to maintain consistency among reservation records, payment states, and real-time quota availability. As shown in Fig. 1, users access a React (Vite) front end that communicates with a Node.js (Express) REST API for authentication, booking creation, and quota validation. Transactional data are persisted in a MySQL database accessed through Prisma ORM.

Reservation requests are handled synchronously through REST endpoints to validate trail availability, time-slot closures, and quota constraints before a provisional booking (pending) is created. For payment initiation, the backend requests a Snap token and returns it to the front end so the user can open the Snap payment UI. Payment outcomes are delivered asynchronously via Midtrans webhook notifications (settlement/expire/cancel) sent to the /payments/notify endpoint, which was exposed to Midtrans during local testing using an Ngrok public URL.

To ensure reliability under retries or duplicate deliveries, each webhook notification is validated using signature verification and processed using idempotent rules so that repeated events produce a single consistent state transition. Quota consistency is enforced by coupling payment states with quota counters: a pending transaction temporarily reserves quota, a settlement event finalizes quota deduction and enables ticket issuance (ticket_id), and an expiry/cancellation event restores quota to prevent over-allocation. The internal workflow and data structures are documented using UML artifacts to maintain traceability from business rules to implementation.

*name of corresponding author

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 10, Issue 1, Januari 2026
DOI : https://doi.org/10.33395/sinkron.v10i1.15600
e-ISSN : 2541-2019
p-ISSN : 2541-044X

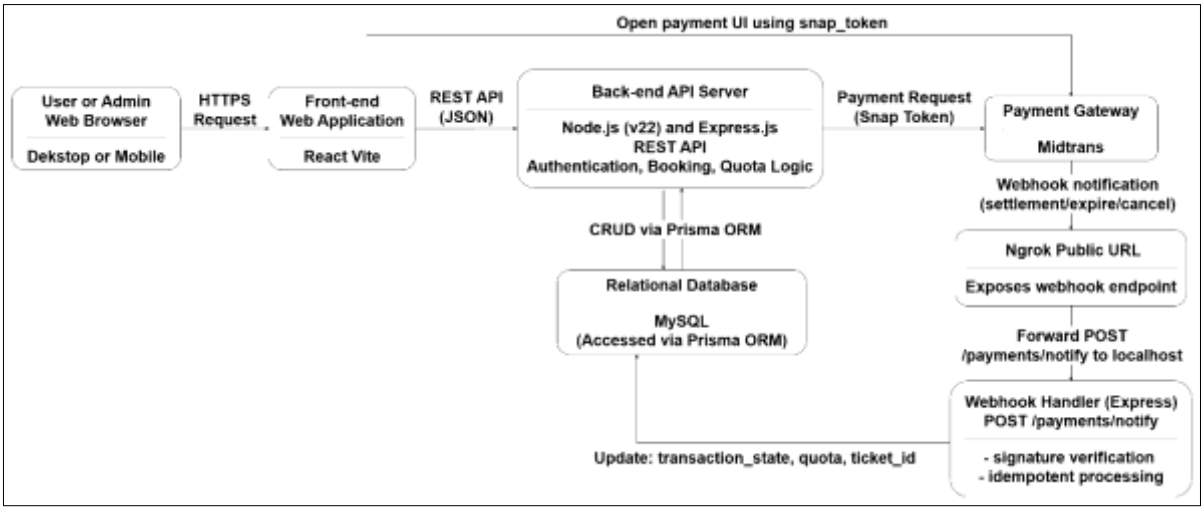

Fig. 1 Full system architecture and event-driven payment notification flow (Midtrans Snap and webhook).

Implementation Environment

All implementation and evaluation activities were conducted in a local development environment (localhost) on a Windows 10 workstation equipped with an AMD Ryzen 5 5500U CPU and 8 GB RAM. The backend service was implemented using Express.js on Node.js v22.14.0. Data persistence used MySQL Community Server 8.4.3 with Prisma ORM v6.16.2.

Payment processing and webhook notifications were tested using the Midtrans Snap sandbox environment, with the webhook endpoint exposed via an Ngrok public URL. Performance measurements and functional observations were recorded using Google Chrome (desktop) and Chrome DevTools (Console/Network). The backend API and database were executed locally on the same machine using a localhost connection.

Evaluation Protocols

Functional testing was conducted using end-to-end black-box scenarios covering the critical user journey: booking creation, payment initiation via Midtrans Snap, e-ticket issuance, and ticket validation (check-in). A scenario was considered passed if the expected outputs were produced without UI/server errors and behavior remained consistent across the tested browsers. Technical evaluation focused on webhook processing latency, API responsiveness, and quota synchronization under concurrent access.

Webhook latency was measured across 30 Midtrans settlement events in the sandbox environment. Time t0 was recorded when the webhook HTTP POST request was received at the /payments/notify handler (server log timestamp), and time t1 was recorded after the corresponding Payment/Booking update was committed in the database; webhook latency was defined as (t1 − t0). API response time was measured using Chrome DevTools (Network) for 50 requests per endpoint in the booking flow (/quota/check, /booking/create, and /ticket/{id}) under normal operating conditions.

For concurrency validation, quota synchronization was tested using controlled bursts of 20 parallel booking attempts targeting the same slot. The objective was to verify that the system prevents over-allocation and correctly restores quota after expire/cancel events. (Kalau jurnal kamu menuntut definisi kuantitatif, tambahkan 1 kalimat: “quota-synchronization accuracy was defined as zero over-allocation events and correct quota restoration in all expire/cancel cases.”)

Usability Evaluation (SUS) Participants and Scoring

After completing the “booking → payment → view ticket” task, participants filled out the SUS questionnaire consisting of 10 items on a 5-point Likert scale. SUS scoring followed the standard procedure: odd-numbered items were transformed to (response − 1) and even-numbered items to (5 − response); the item scores were summed and multiplied by 2.5 to obtain a 0–100 SUS score (Kurniawan et al., 2022; Yoga Pudya Ardhana, 2022). The system was considered usable if the mean SUS score was ≥ 68.

The usability evaluation involved 13 respondents recruited using purposive sampling from the researchers’ network to represent prospective hikers familiar with web-based applications and online transactions. Prior hiking experience was recorded as a binary variable (yes/no); in this study, 9 participants reported prior hiking experience and 4 intended to hike Mount Slamet. Demographic variables such as age and gender were not collected, which is noted as a limitation when interpreting the generalizability of the SUS findings.

*name of corresponding author

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 10, Issue 1, Januari 2026
DOI : https://doi.org/10.33395/sinkron.v10i1.15600
e-ISSN : 2541-2019
p-ISSN : 2541-044X

RESULT

Overview of Results

This section reports the evaluation results of the proposed e-ticketing system in terms of functional correctness, usability, and technical performance. End-to-end black-box testing confirmed that core scenarios—booking, payment, ticket issuance, and check-in—executed correctly without UI/server errors. Usability testing using the System Usability Scale (SUS) produced a mean score of 75.0, indicating acceptable perceived usability. From a technical standpoint, the payment notification pipeline processed Midtrans settlement events with a mean webhook update time of 2.41 s (n = 30), while core API endpoints maintained mean response times below 250 ms (n = 50 per endpoint). Quota synchronization tests under concurrent booking attempts showed no over-allocation and correct quota restoration for expire/cancel events.

Use Case Diagram

The Use Case diagram summarizes interactions between the actors and the system's primary functions. As shown in Fig. 2, the main actors are Visitor/User, Basecamp Admin/Staff, and Midtrans. Visitors can register, sign in, browse hiking routes, manage group members, create bookings, and complete payments through Midtrans. Admins manage route data, hiking quotas, ticket prices, date closures, and payment validation.

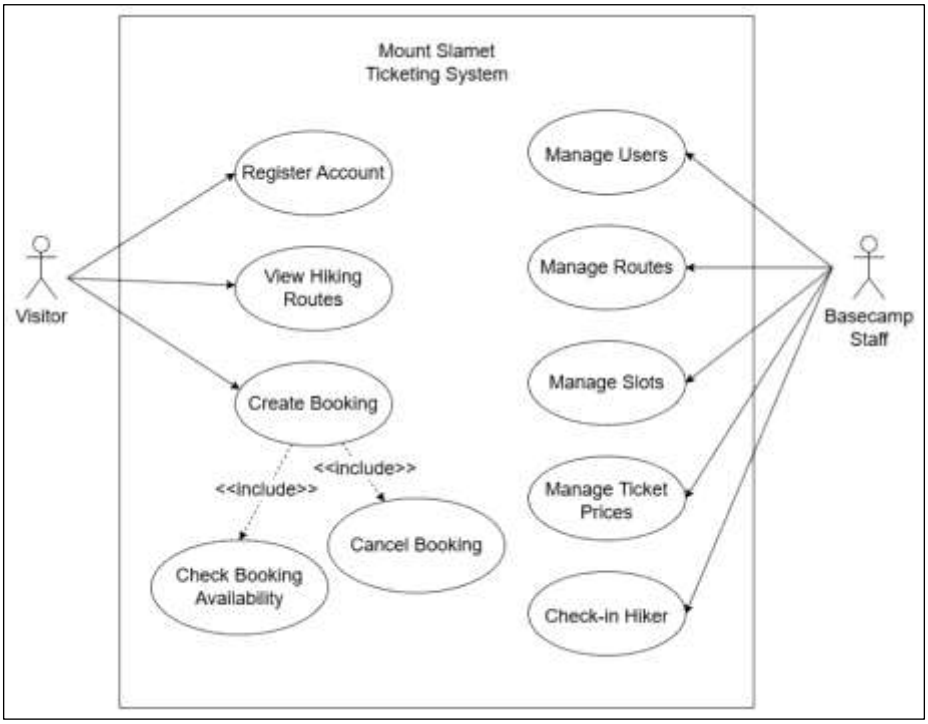

Fig. 2 Use Case diagram of the Mount Slamet Hiking Ticketing System.

Event-Driven Payment and Quota Consistency

Fig. 3 illustrates the payment-state transitions and their coupling to quota updates in an event-driven workflow. After a booking is created, the backend generates a Midtrans Snap token and the front end opens the Snap payment UI for the user. Transaction outcomes are finalized through asynchronous Midtrans notifications (e.g., settlement, expire, cancel) handled by the server, rather than relying on client-side status assumptions. The webhook handler validates event authenticity (signature verification) and applies idempotent processing so that repeated notifications lead to a single effective state transition. Settlement finalizes quota deduction and triggers ticket identifier issuance, while expire/cancel restores quota to prevent over-allocation.

*name of corresponding author

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 10, Issue 1, Januari 2026
DOI : https://doi.org/10.33395/sinkron.v10i1.15600
e-ISSN : 2541-2019
p-ISSN : 2541-044X

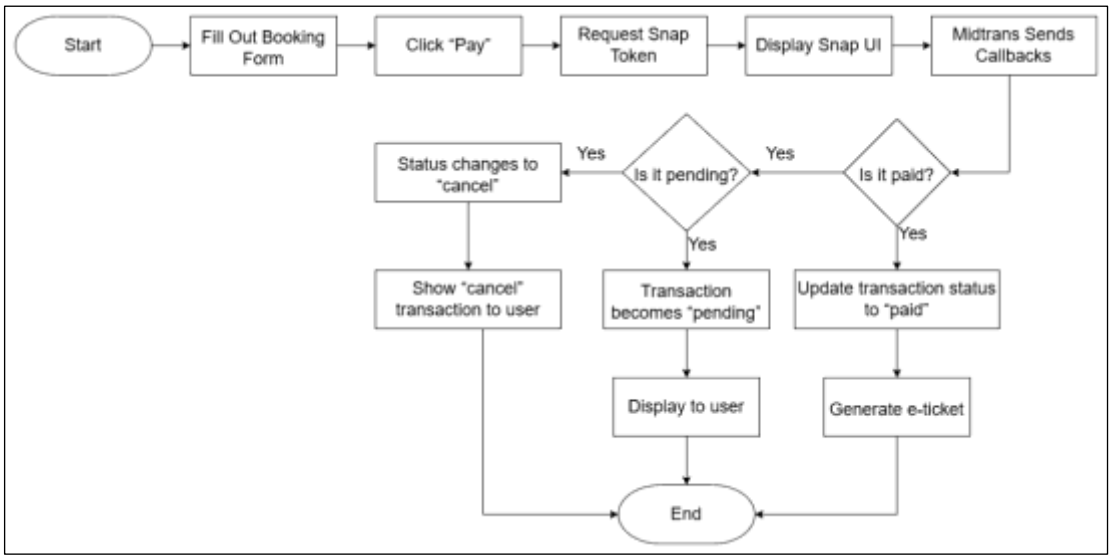

Fig. 3 Event-driven payment and quota consistency for the Mount Slamet Hiking Ticketing System.

Black-box Testing

The release-candidate build was tested using end-to-end black-box scenarios covering the critical user journey. All evaluated scenarios produced the expected outputs and system states, including booking creation, payment initiation, ticket issuance after settlement, and ticket validation at check-in. Table 3 summarizes representative functional test results for key modules. These results indicate that the system meets the functional requirements for online management of Mount Slamet hiking-ticket bookings.

Table 3
Functional test results (black-box testing).

Function Tested	Test Scenario	Expected Result	Actual Result	Status
Login	Enter valid email and password	Successfully reaches dashboard	Outcome aligned with expected behavior	Pass
Ticket Booking	Input ticket quantity and date	Data saved and redirected to Midtrans	Outcome aligned with expected behavior	Pass
Midtrans Payment	Complete transaction and receive webhook	Transaction status = PAID	Outcome aligned with expected behavior	Pass
Ticket PDF Generation	System generates PDF ticket after successful payment	Available when transaction status = PAID	Outcome aligned with expected behavior	Pass
Ticket Check-in	Verify ticket with status PAID	Ticket valid and allowed to enter	Outcome aligned with expected behavior	Pass

Evaluate System Usability Scale (SUS)

Usability was assessed using SUS after participants completed the "booking → payment → view ticket" task on the same build. Thirteen respondents completed the 10-item questionnaire, and SUS scores were computed following the standard transformation and scaling procedure. As shown in Fig. 4, individual SUS scores ranged from 67.5 to 82.5 with a mean of 75.0, exceeding the commonly used benchmark of 68. Using adjective-rating interpretations, a mean SUS score of 75.0 corresponds to a "Good" usability level. This score also falls within the acceptable range and corresponds to approximately a grade of C to B depending on the adopted SUS grading scale.

*name of corresponding author

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 10, Issue 1, Januari 2026
DOI : https://doi.org/10.33395/sinkron.v10i1.15600
e-ISSN : 2541-2019
p-ISSN : 2541-044X

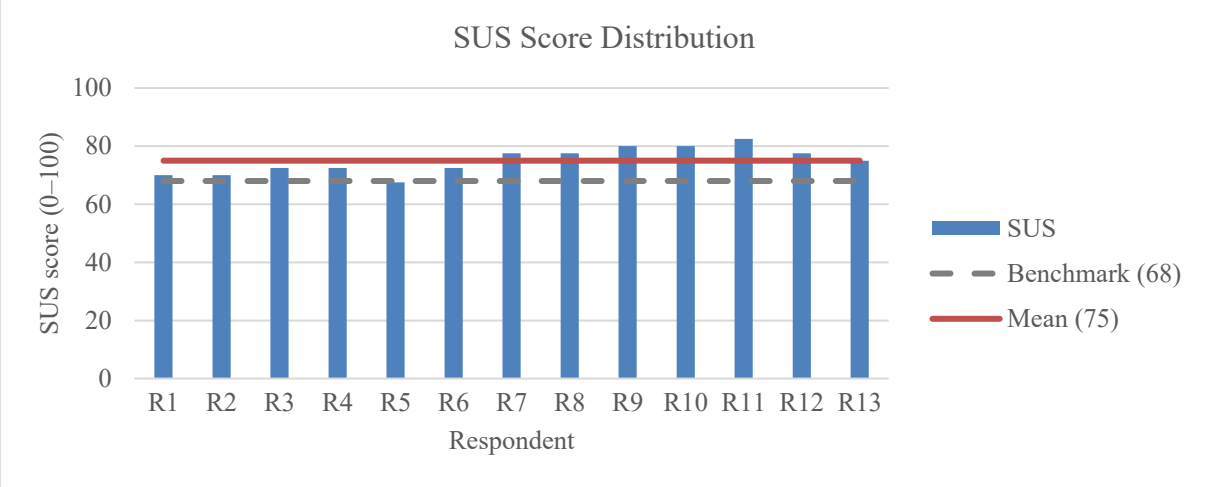

Fig. 4 SUS score distribution (n = 13).

Webhook Propagation Performance

To evaluate the reliability of the asynchronous payment flow, we measured the propagation time between a Midtrans settlement event and the corresponding update in the Payment and Booking tables. Thirty settlement events were observed in the sandbox environment. As shown in Fig. 5, webhook propagation time averaged 2.41 s (SD = 0.68 s), with a minimum of 1.32 s and a maximum of 3.79 s. No duplicate or missed events were recorded across the 30 events, supporting the effectiveness of idempotent webhook handling.

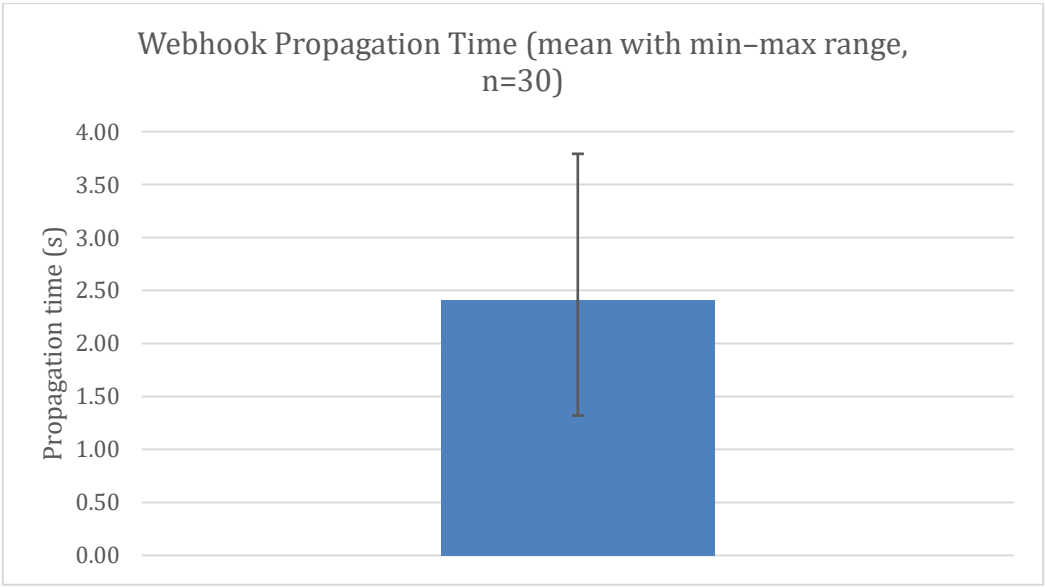

Fig. 5 Webhook propagation time (mean with min–max range, n = 30 settlement events). Mean = 2.41 s; SD = 0.68 s; min = 1.32 s; max = 3.79 s; duplicate = 0; missed = 0.

Application Response Time

The response time for core endpoints was measured using browser developer tools and API logs under normal network conditions (20–25 Mbps). Three operations were evaluated: quota checking (/quota/check), booking submission (/booking/create), and ticket retrieval (/ticket/{id}), with 50 requests per endpoint. As shown in Fig. 6, mean response times were 142±21 ms, 231±34 ms, and 187±27 ms, respectively, indicating consistent responsiveness for user-facing interactions under the tested conditions. The corresponding medians were 138 ms, 224 ms, and 181 ms, and all mean response times were below the 250 ms responsiveness target. Fig. 6 summarizes response-time variability using mean ± SD across 50 requests per endpoint; medians are also reported to provide a robust central tendency measure under potential skew.

*name of corresponding author

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 10, Issue 1, Januari 2026
DOI : https://doi.org/10.33395/sinkron.v10i1.15600
e-ISSN : 2541-2019
p-ISSN : 2541-044X

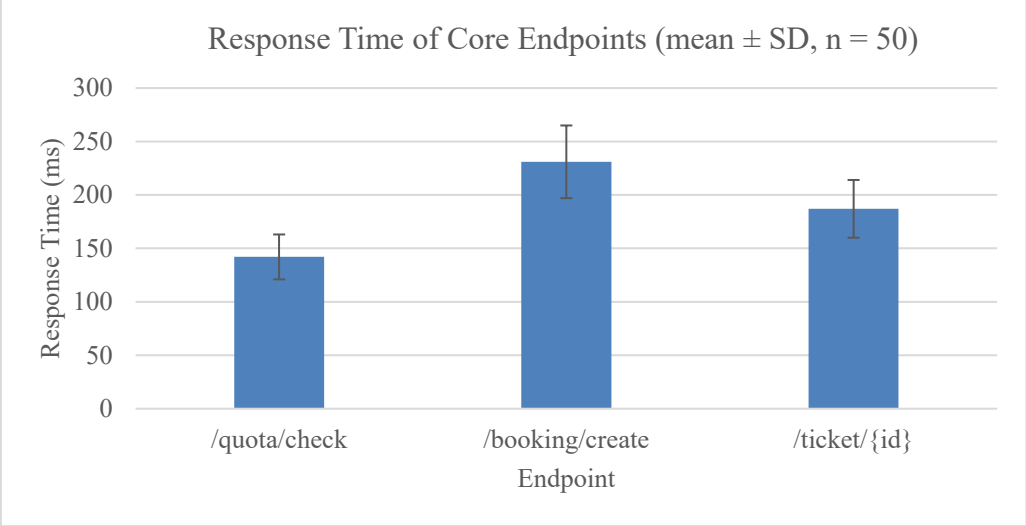

Fig. 6 Response time of core endpoints (mean ± SD, n = 50 per endpoint).

Quota Synchronization Accuracy

Quota consistency was evaluated using controlled bursts of 20 parallel booking attempts targeting the same slot. Across the tested scenarios, the system prevented over-allocation in all runs and restored quota correctly after expire/cancel events. Table 4 summarizes the outcomes across the concurrency scenarios. Overall, the quota-synchronization test produced a 0% error rate under the tested conditions.

Table 4
Quota consistency test under concurrent booking attempts.

Scenario	Runs	Consistent Results	Error Rate
Parallel 20-user booking	20	20	0%
Settlement → quota decrement	30	30	0%
Expire → quota restoration	15	15	0%

Manual vs. System Error Rate

Operator logs and interviews indicated that the manual process typically generated an average of four recording errors per week, with occasional overbooking incidents and payment verification delays. In contrast, no recording errors were observed across 40 test transactions using the proposed system in the evaluation environment. Table 5 compares error patterns between the manual process and the proposed system. This comparison should be interpreted as context-specific operational evidence rather than a generalized population estimate.

Table 5
Table 5. Manual vs. system error rate comparison.

Metric	Manual Process	Proposed System
Recording errors	4 errors/week (avg)	0 errors
Overbooking incidents	1-2 per month	0 incidents
Payment verification delay	10-30 min	2.41 s (mean webhook propagation)

DISCUSSION

The evaluation results provide evidence on functional correctness, transactional consistency, usability, and the reliability of the event-driven payment integration. End-to-end black-box testing confirmed that the core scenarios—booking, payment transition handling, ticket issuance, and check-in—executed correctly across the tested devices and browsers. Within the tested environment, these results support the feasibility of deploying an integrated booking–payment–validation workflow for quota-restricted hiking services.

The System Usability Scale (SUS) yielded a mean score of 75.0 (n = 13), which exceeds the commonly used benchmark of 68 and indicates acceptable usability. Participant feedback suggested that the booking-to-payment flow was easy to follow, likely due to consistent layouts and clear step-by-step progression. However, feedback also revealed usability gaps, particularly the lack of prominent error notifications for expired or failed payments

*name of corresponding author

e-ISSN : 2541-2019
p-ISSN : 2541-044X

and the discoverability of the e-ticket download action. These issues indicate that usability improvements should prioritize error-state visibility and the placement of critical post-payment actions.

Compared with prior e-ticketing studies, the findings align with the general conclusion that web-based booking improves information clarity and operational efficiency. The hiking context, however, imposes stricter operational constraints because quota enforcement and payment-state accuracy directly affect both service capacity and gate-side validation. In this study, coupling payment states to quota counters and processing Midtrans notifications through signature verification and idempotent handling reduced the risk of duplicate state transitions and inconsistent quota updates. While this does not guarantee correctness under all real-world conditions, it demonstrates that event-driven reliability mechanisms can function as practical safeguards in quota-restricted outdoor tourism settings.

Technical limitations

Several constraints should be considered when interpreting these findings. First, payment evaluation was conducted in a sandbox environment, which may not fully reflect production conditions such as banking delays, operational monitoring practices, and more variable callback retry patterns. Second, response-time measurements assumed stable network conditions (20–25 Mbps), and therefore do not represent the latency and packet-loss characteristics that may occur in mountainous areas. Third, concurrency was tested in controlled bursts rather than under sustained peak-load conditions, and the study did not include formal stress/load testing. Future validation should include broader network variability, higher concurrency, and failure-mode scenarios such as delayed webhooks, repeated notifications, and partial downstream failures.

Scalability potential and architectural implications

The system can be extended to multi-basecamp or multi-destination deployment by introducing tenant-aware configuration (e.g., destination identifiers, per-location quota rules, pricing policies, and closure calendars) and partitioning quota and transaction records by destination. At larger scale, the event-driven approach can be strengthened using established reliability patterns, including durable audit/event logging, controlled retries with backoff, and dead-letter handling for repeatedly failing events. If throughput and operational complexity increase, a message broker can decouple webhook ingestion from downstream quota updates and reporting, while preserving the same state-machine semantics and consistency constraints. This evolution would also support better observability and operational incident handling without changing the core business rules.

Overall, these findings indicate that event-driven payment integration can support payment-state consistency and quota integrity in a quota-restricted ticketing workflow. At the same time, production-scale validation remains necessary to confirm robustness under diverse networks, higher concurrency, and real operational incident conditions.

CONCLUSION

This study developed a web-based e-ticketing system for Mount Slamet that integrates real-time quota management with a payment gateway using signature-verified and idempotent webhook processing. Functional testing, cross-browser validation, and usability evaluation indicate that the system can support an end-to-end workflow from booking creation to payment confirmation and e-ticket issuance within the tested environment. The usability assessment produced a mean SUS score of 75.0, which suggests acceptable perceived usability for the evaluated booking–payment flow.

The contributions of this work are threefold. First, it presents an architectural model for event-driven payment integration in quota-restricted tourism services, demonstrating how signature verification and idempotent webhook handling can reduce the risk of inconsistent transaction states under duplicate event delivery. Second, it proposes a domain-oriented linkage between quota reservation, settlement-based quota finalization, and automated quota restoration on expire/cancel events, which is often under-specified in prior hiking or tourism ticketing implementations. Third, it adds standardized usability evidence (SUS) to support comparability and provide an initial acceptance signal for hiking e-ticketing contexts.

From a practical perspective, automating payment-state verification and coupling payment outcomes to quota counters can reduce manual confirmation steps and improve the consistency of quota information displayed to users. These properties can help reduce administrative friction at basecamps and support more predictable capacity planning, especially during peak-demand periods. However, the operational impact should be interpreted as context-specific evidence from the tested setup rather than a generalized production guarantee.

Future research should include controlled load testing (e.g., 500–1,000 concurrent users) to assess throughput, latency, and failure behavior under peak demand. Production-grade validation is also needed to evaluate unstable networks, real banking delays, refund/void flows, and webhook retry patterns under operational monitoring. In addition, QR-based gate scanning can be integrated and evaluated to strengthen check-in speed and auditability.

*name of corresponding author

Finally, comparative studies across multiple payment providers (e.g., Midtrans, Xendit, and DOKU) can examine differences in webhook latency, retry policies, and consistency characteristics in multi-provider deployments.

ACKNOWLEDGMENT

The success of this research was made possible by the support of many parties. The author(s) express profound gratitude to[Dani Arifudin and M. Syaiful Amin for guidance, support, and constructive feedback throughout the study and the preparation of this article. We also thank Amikom Purwokerto University, Faculty of Computer Science the internal examination/review team, and the Mount Slamet basecamp operators for their technical and operational assistance. The author(s) appreciate the contributions of others who cannot be named individually.

REFERENCES

Aldho Febrian Harahap, M., Parga Zen, B., Ahmad Firdaus, E., & Yunita Kisworini, R. (2024). Metode rapid application development pada penerapan pemesanan tiket online Gunung Slamet berbasis website. *JURNAL INFORMATIKA*, *3*(2), 20–27. https://doi.org/10.57094/ji.v3i2.2064

Anwar, K., Kurniawan, L. D., Rahman, M. I., & Ani, N. (2020). Aplikasi marketplace penyewaan lapangan olahraga dari berbagai cabang dengan metode agile development. *Jurnal Sisfokom (Sistem Informasi Dan Komputer)*, *9*(2), 264–274. https://doi.org/10.32736/sisfokom.v9i2.905

Arijaya, I. G. N. P., Pradnyana, I. M. A., Wirawan, I. M. A., Suwendra, I. W., Nugraha, I. G. P., & Suparta, I. N. (2020). Usability Testing in Tourism Object Management System. *Advances in Social Science, Education and Humanities Research*, 139–144. Paris: Atlantis Press. https://doi.org/10.2991/assehr.k.200115.023

Badiwibowo Atim, S. (2024). Permodelan sistem informasi penjualan barang berbasis website menggunakan metode agile. *Journal of Artificial Intelligence and Technology Information (JAITI)*, *2*(1), 14–25. https://doi.org/10.58602/jaiti.v2i1.104

Banyumas Ekspres. (2025, July 26). Jumlah pendaki Gunung Slamet via Bambangan diprediksi membludak di bulan Agustus. https://banyumasekspres.id/jumlah-pendaki-gunung-slamet-via-bambangan-diprediksi-membludak-di-bulan-agustus

Christi, M., Putra, W. H. N., & Hanggara, B. T. (2023). Rancang bangun sistem informasi dan pelayanan e-ticket (booking online) pada wisata pendakian Gunung Budheg Tulungagung menggunakan website dengan framework Laravel. *JPTIIK*, *7*(1), 83–91. https://j-ptiik.ub.ac.id/index.php/j-ptiik/article/view/12105

Dwi Jayanto, A., Kumalasari Niswatin, R., & Kasih, P. (2021). Sistem informasi dan pelayanan e-tiket berbasis website menggunakan algoritma FIFO pada kawasan wisata Trenggalek. *Seminar Nasional Inovasi Teknologi*, *5*(1), 207–212. https://doi.org/10.29407/inotek.v5i1.949

Fatman, Y., Khoirun Nafisah, N., & Bendoro Jembar Pambudi, P. (2023). Implementasi payment gateway dengan menggunakan Midtrans pada website UMKM Geberco. *Jurnal KomtekInfo*, *10*(2), 64–72. https://doi.org/10.35134/komtekinfo.v10i2.364

Ferreira, A., Brändle, N., Widhalm, P., & Olaverri-Monreal, C. (2018). Assessment of trip validation interfaces for smartphone-based travel surveys. *Transportation Research Procedia*, *32*, 126–134. Vienna: Elsevier B.V. https://doi.org/10.1016/j.trpro.2018.10.025

Global Volcanism Program. (2025). Slamet (Indonesia). https://doi.org/10.5479/si.GVP.VOTW5-2025.5.3

Hendra, Wahyuningsih, Y., & Mahendrasusila, F. (2024). Rancang bangun sistem proses transaksi perusahaan berbasis website dengan metode agile development. *Jurnal Pengembangan Riset Dan Observasi Rekayasa Sistem Komputer (PROSISKO)*, *11*(1), 10–19.

Islam, M. N. (2023). Factors affecting adoption of self-service E-ticketing technology: A study on heritage sites in Bangladesh. *Heliyon*, *9*(3). https://doi.org/10.1016/j.heliyon.2023.e14691

Kreps, J., Narkhede, N., & Rao, J. (2011). Kafka: A distributed messaging system for log processing. *Proceedings of the NetDB'11 Workshop*, 1–7. Athens: Association for Computing Machinery (ACM). https://doi.org/10.1145/1993130.1993140

Kurniawan, E., Nofriadi, & Nata, A. (2022). Penerapan system usability scale (SUS) dalam pengukuran kebergunaan website program studi di STMIK Royal. *Journal of Science and Social Research*, (1), 43–49. https://doi.org/10.54314/jssr.v5i1.817

Lazzari, L., & Farias, K. (2023). Event-driven architecture and REST architectural style: An exploratory study on modularity. *Journal of Applied Research and Technology*, *21*(3), 338–351. https://doi.org/10.22201/icat.24486736e.2023.21.3.1764

Lübeck, R. M., Wittmann, M. L., & Flores Battistella, L. (2012). Electronic ticketing system as a process of innovation. *Journal of Technology Management & Innovation*, *7*(1), 18–30. https://doi.org/10.4067/S0718-27242012000100002

Lubis, R. A., Zakir, A., & Sembiring, B. O. (2025). Pemesanan e-tiket bus menggunakan metode agile pada CV Final Transport. *EXPLORE*, *15*(1), 40–47.

*name of corresponding author

e-ISSN : 2541-2019
p-ISSN : 2541-044X

Muliadi, M., Andriani, M., & Irawan, H. (2020). Perancangan sistem informasi pemesanan kamar hotel berbasis website (web) menggunakan data flow diagram (DFD). *JISI: Jurnal Integrasi Sistem Industri*, *7*(2), 111–122. https://doi.org/10.24853/jisi.7.2.111-122

Nasri, E., Nuryani, E., & Agustiawan, N. (2021). Rancangan e-commerce dan payment API Midtrans untuk produk UMKM (studi kasus pada Rumah Kemasan). *Jurnal Sistem Informasi Dan Informatika (SIMIKA)*, *4*(2), 131–146. https://doi.org/10.47080/simika.v4i2.1366

Nova, S. H., Widodo, A. P., & Warsito, B. (2022). Analisis metode agile pada pengembangan sistem informasi berbasis website: Systematic literature review. *Techno.Com*, *21*(1), 139–148. https://doi.org/10.33633/tc.v21i1.5659

Subramanya, K., Kermanshachi, S., Pamidimukkala, A., & Loganathan, K. (2023). An assessment of the impacts of electronic ticketing on inspections and material procurement for highway construction. *Frontiers in Built Environment*, *9*. https://doi.org/10.3389/fbuil.2023.1182609

Sulistyorini, T., Sova, E., & Ramadhan, R. (2022). Pemantauan kasus penyebaran COVID-19 berbasis website menggunakan framework React JS dan API. *JUKIM*, *1*(4), 1–13. https://doi.org/10.56127/jukim.v1i04.137

Suwarno, A., Hamimi, R., & Edora. (2023). Pemodelan aplikasi pemesanan e-tiket pendakian gunung berbasis Android. *Jurnal Ilmiah Teknologi - Informasi & Sains (TEKNOIS)*, *13*(1), 133–145. https://doi.org/10.36350/jbs.v13i1

Widiarta, I. M., Mulyanto, Y., & Sutrianto, A. (2023). Rancang bangun sistem informasi inventory menggunakan metode agile software development (studi kasus Toko Nada). *Digital Transformation Technology*, *3*(1), 133–143. https://doi.org/10.47709/digitech.v3i1.2549

Wiyono, H., & Fachrie, M. (2024). Perancangan aplikasi pemesanan tiket wisata di Kecamatan Donorojo menggunakan metode payment gateway. *Jurnal Indonesia : Manajemen Informatika Dan Komunikasi*, *5*(1), 574–586. https://doi.org/10.35870/jimik.v5i1.522

Yoga Pudya Ardhana, V. (2022). Evaluasi usability e-learning Universitas Qamarul Huda menggunakan system usability scale (SUS). *Journal of Informatics, Electrical and Electronics Engineering*, *2*(1), 5–11. https://doi.org/10.47065/jieee.v2i1.430

*name of corresponding author