Journal of Hypermedia & Technology-Enhanced Learning
Vol. 4, No. 2, June (2026), Pages 182–201
e-ISSN: 2985–9166 | 10.58536/j-hytel.240

RESEARCH ARTICLE

SiKurban: A Web-Based One-Time-Use QR Code System Using LCG for Real-Time Duplicate-Claim Prevention in Qurban Meat Distribution

Farras Fathin Naufal[1] *, Dedy Irfan[2], Delvi Asmara[2]

[1] Department of Informatics Education, Faculty of Engineering, Universitas Negeri Padang, Padang, **Indonesia**
[2] Faculty of Engineering, Universitas Negeri Padang, Padang, **Indonesia**

**Corresponding Author:* *fnfarras@gmail.com*

This article contributes to:

ABSTRACT

Paper coupons used for qurban (Islamic animal sacrifice) meat distribution are vulnerable to loss, physical damage, forgery, and repeated redemption. At Surau Kumango Pasa Gadang in Padang, West Sumatra, the 2025 process involved 448 coupons, 56 participants, eight animals, and 20 committee members. This study designed and functionally evaluated SiKurban, a web application that replaced paper-only verification with one-time-use QR codes and real-time duplicate-claim prevention. Development followed the Waterfall model using Laravel and MySQL. Initial token values were generated with a Linear Congruential Generator (LCG) using the Numerical Recipes parameters ($a = 1,664,525$; $c = 1,013,904,223$; $m = 2^{32}$), checked for uniqueness in the database, and signed with HMAC-SHA256. During scanning, the application verified the token and its status, then committed the distribution record, the scan log, and the irreversible transition to "used" within a single locked database transaction. Developer-executed black-box testing on 1 July 2026 covered 33 scenarios (28 positive and five negative or edge cases). All scenarios produced the specified outputs, including rejection of reused and unregistered codes. The findings demonstrated developer-verified functional conformance of this single-site prototype under the specified test conditions; they do not establish independent validation or field effectiveness. The 13 non-functional requirements, usability, user acceptance, security resistance, scalability, cost, and performance under unstable connectivity remain to be evaluated by committee members or independent testers during live or high-fidelity distribution.

KEYWORDS

Qurban meat distribution; one-time-use QR code; Linear Congruential Generator; duplicate-claim prevention; web information system.

Received: May. 10, 2026; *Revised:* Jun. 24, 2026; *Accepted:* Jul. 25, 2026; *Published Online:* Jul. 27, 2026

How to Cite: Naufal, F. F., Irfan, D., & Asmara, D. (2026). SiKurban: A Web-Based One-Time-Use QR Code System Using LCG for Real-Time Duplicate-Claim Prevention in Qurban Meat Distribution. *Journal of Hypermedia & Technology-Enhanced Learning, 4*(2), 182–201. https://doi.org/10.58536/j-hytel.240

Published by Sagamedia Teknologi Nusantara

1. INTRODUCTION

Qurban (ritual animal sacrifice) has both spiritual and socio-economic functions, and its social value depends on orderly delivery to intended recipients. Community-based qurban programs in Indonesia continue to expand and diversify, increasing the need for transparent recording, entitlement determination, and distribution control [1], [2]. Operational studies have reported that manual qurban management can produce inaccurate recipient records, duplicate coupons, unequal distribution, and limited financial transparency [3]. Packaging and collection processes must also manage queue efficiency and hygiene, which become more difficult when volunteer committees verify claims under time pressure [4]. These risks underscore the need for a system that links recipient records, entitlements, coupon status, and transaction logs.

A similar situation occurs at Surau Kumango Pasa Gadang, located in Pasa Gadang Sub-district, South Padang District, Padang City, West Sumatra, which serves as the center of qurban implementation for the surrounding community every year. Table 1 summarizes the field data collected from the 2025 qurban implementation at this site.

Table 1. Baseline qurban distribution data at Surau Kumango Pasa Gadang (2025)

No.	Aspect	Description
1	Number of qurban participant groups	8 groups
2	Number of qurban participants	56 people (7 per group)
3	Number of sacrificial animals	8 animals
4	Total small coupons managed	448 coupons
5	Number of qurban committee members	20 people

As shown in Table 1, the 2025 qurban implementation involved eight participant groups, 56 participants, eight sacrificial animals, 448 small coupons, and 20 committee members. Processing 448 coupons within a concentrated distribution workflow created a substantial manual verification burden and opportunities for recording errors, although this study did not quantify workload or error rates. Recipients were also classified into two entitlement categories: Category 1 for participants who were also recipients and Category 2 for recipients only. The paper process could not apply these distinctions automatically, so committee members had to determine and verify entitlements manually, particularly when a single person held multiple roles.

Paper coupons are susceptible to water damage, tearing, loss, illegibility, and forgery [5]. Under queue pressure, a committee may also be unable to consistently verify each coupon, leading to invalid or repeated claims [6]. At the study site, the volume of 448 coupons increased exposure to these failure modes. The same control problem is recognized in electronic-coupon systems: a centrally issued voucher can be redeemed repeatedly unless its state is checked and irreversibly changed immediately after a valid transaction [7].

Several studies have proposed digital alternatives to paper coupons. Laksono [8] designed a web-based digital qurban coupon verification system that applies AES encryption as an additional security layer to prevent forgery and misuse. In a related direction, the Ministry of Administrative and Bureaucratic Reform of the Republic of Indonesia, together with the government of the Republic of Korea, developed Verifiable Credentials, a digital identity standard that allows a person to be verified instantly through a QR code on a smartphone without disclosing their full personal data [9]; this privacy-preserving verification principle is relevant to the design of a qurban distribution system that must protect recipients' personal information. Beyond documenting ownership, digital distribution systems also depend on a reliable mechanism to generate unique identifiers for every recipient; the Linear Congruential Generator (LCG) is a lightweight pseudo-random number generator that has been widely used in various computing applications for this purpose [10].

Real-world implementations further confirm the benefit of digitalization. Takyudin [11] reported that digitalizing qurban management at Masjid Al-Jauhar accelerated administrative processes by up to 60%, increased transparency through digital reporting, and raised congregational participation by 21.43%. This pattern is consistent with a broader trend across Islamic philanthropic institutions in the region, where fintech-enabled collection and distribution platforms are reported to improve the efficiency, accountability, and transparency of managing charitable funds such as zakat [12]. Erwansyah et al. [13] implemented a barcode-based coupon system at Musala An-Nur, North Sumatra, during Eid al-Adha 1446 H/2025, in which scanning a coupon automatically moved a recipient's status from "not yet collected" to "collected," enabling the committee to identify unclaimed shares in real time. Likewise, Lewenusa [6] applied a barcode-based queue and anti-duplication system at Musala Nurul Iman, South Jakarta, in which a coupon scanned more than once triggered a warning that the recipient had already collected their share.

A specific design gap nevertheless remained. The barcode systems reported by Lewenusa [6] and Erwansyah et al. [13] retained printed paper as the physical carrier, so damage or loss remained possible even though verification was digital. Five other qurban information-system studies addressed online coupons, stock reduction, recipient records, administration, or end-to-end animal management [14], [15], [16], [17], [18]. Among these five studies, none reported the combined use of deterministic identifier generation, database uniqueness validation, a keyed token digest, one-time-use state transitions, real-time duplicate-claim rejection, and category-based entitlement within one platform.

This study addressed the identified gap by designing SiKurban for Surau Kumango Pasa Gadang. Its contribution comprised five integrated controls: (1) an LCG to allocate deterministic initial identifiers, with each output checked against the database before issuance; (2) a QR code whose status could transition from available to used only once; (3) real-time rejection and logging of repeated redemption attempts; (4) automatic application of recipient-category entitlements; and (5) integration of data collection, token generation, QR-code issuance, scanning, distribution recording, and reporting. The LCG was not treated as a cryptographic random-number generator; uniqueness, integrity, and redemption control were implemented as separate safeguards.

The objectives of this study were to design and build a web-based qurban information system that replaces the conventional paper-coupon system using LCG-generated one-time-use QR codes; to implement real-time anti-duplication verification that automatically prevents duplicate meat collection; to design a recipient-classification feature so that entitlement corresponds to each recipient's status and role; and to evaluate the functional correctness of the developed system using black-box testing. Accordingly, this study was guided by four research questions: (RQ1) How can a digital qurban information system be designed to replace the conventional paper-coupon system using an LCG-based unique-code-generation algorithm? (RQ2) How can a one-time-use QR-code mechanism with real-time anti-duplication verification be implemented so that every code can be used only once? (RQ3) How can a recipient-classification feature be designed so that meat entitlement corresponds to each recipient's category? (RQ4) What are the functional testing results of the developed system, as measured by black-box testing?

2. Methods

2.1. Research Design

This study used an applied research and development (R&D) design to build a web-based information system, following the Waterfall software development model [19], as illustrated in Figure 1. The Waterfall model was selected because it suits a project whose specifications were clear and well-defined from the outset, following a linear sequence in which each stage is completed before the next begins [19]. The five stages were: Requirements, in which system needs were identified through consultation with the users, i.e., the management and committee of Surau Kumango Pasa Gadang; Design, in which the requirements were translated into a software and database architecture; Implementation, in which the design was converted into a running web application; Verification, in which the integrated system was tested to ensure it functioned as required; and Maintenance, in which the system was monitored and refined after being put into use.

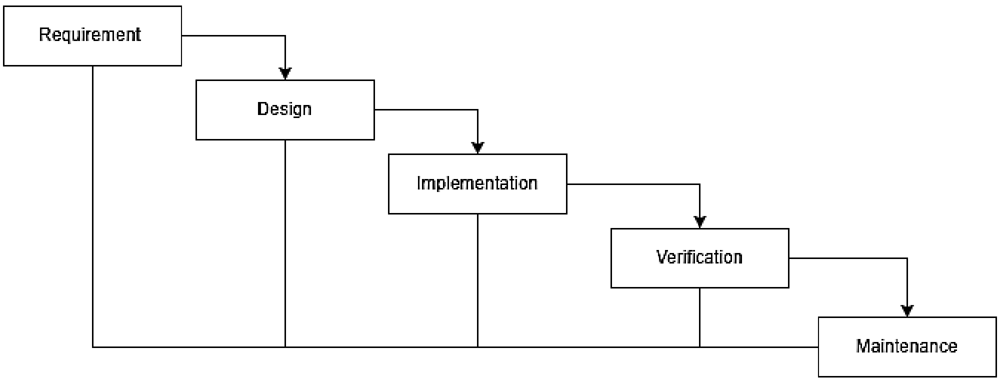


Figure 1. The Waterfall software development model applied in this study, adapted from Saravanos and Curinga [19].

2.2. Study Site and Stakeholders

The 2025 qurban event at Surau Kumango Pasa Gadang, Pasa Gadang Sub-district, South Padang District, Padang City, West Sumatra, Indonesia, supplied the manual-process baseline and the counts reported in Table 1. The same anonymized 2025 records were reused during the 2026 prototype-development and functional-testing cycle; these labels refer to one baseline event followed by system development, not two separate qurban events with identical counts. Field observation and informal interviews were conducted with the surau management and qurban committee to obtain a first-hand account of the manual distribution process, which involved six categories of actors under the existing manual system: the surau management, the head of the qurban committee, the data-collection committee, the distribution committee, qurban participants, and meat recipients. Based on the analysis of this existing process, the proposed system redefined the actors into five system roles with role-based access control: Admin, Data-Collection Committee, Distribution Committee, Surau Management, and Meat Recipient.

2.3. Development Instruments

SiKurban was developed as a server-rendered web application. The back end used the Laravel PHP framework with a Model-View-Controller (MVC) architecture, chosen for its mature ecosystem, broad community support, and built-in facilities for authentication, session management, data validation, and database migration [20], [21], which were managed through the Composer dependency manager [22]. The front end combined HTML5, CSS3, JavaScript, and the Bootstrap 5 framework to produce a responsive interface usable on both laptop and smartphone screens, complemented by the Chart.js library for interactive dashboards. MySQL served as the relational database management system. QR-code images were generated with the SimpleSoftwareIO/simple-qrcode library, camera-based scanning on the distribution-committee side used the jsQR library, and printable reports were produced with the Barryvdh/Laravel-DomPDF package. XAMPP and Visual Studio Code were used as the local development environment.

2.4. System Development Procedure

2.4.1. Requirements Analysis

Based on the field observation and interviews, 28 functional requirements and 13 non-functional requirements were identified. The functional requirements covered role-based login, user- and period-management, participant/group/animal data management, recipient-category management, LCG-based unique-code generation, QR-code creation and printing/downloading, QR-code scanning, real-time verification, one-time-use enforcement, hand-over confirmation, scan logging, monitoring dashboards, distribution reporting, National Identity Number (NIK)-based login for recipients, blank/emergency coupon issuance, a meat-portion calculator, bulk account generation, and an offline-ready scanning mode. The non-functional requirements addressed usability, performance, reliability, security, data integrity, anti-duplication control, browser compatibility, availability, backup and recovery, maintainability, auditability, privacy, and scalability. The 13 non-functional requirements were design targets only; the present evaluation did not operationalize or test them as acceptance criteria.

2.4.2. System Design

The system architecture was modeled using the Unified Modeling Language (UML), a widely used approach for visually representing a system's actors, workflows, and class structure [23]. The model consisted of a use-case diagram involving five actors: Admin, Data Collection Committee, Distribution Committee, Surau Manager, and Recipient (Figure 2). It also included seven activity diagrams representing the Login, Manage Recipient Data, Manage Recipient Groups, Generate QR Code, Real-Time Verification through QR Code Scanning, Monitoring Dashboard, and Print Report or View Distribution Report workflows. In addition, a class diagram was developed comprising 14 interconnected classes.

The database was subsequently designed as an entity-relationship model containing 14 entities: Role, User, Qurban Period, Sacrifice Group, Qurban Participant, Sacrificial Animal, Recipient Group, Recipient, LCG Config, QR Code, Distribution, Log Scan, Report, and Coupon Bank (Figure 3). This approach is commonly used to identify an organization's data requirements before physical implementation [24]. The model was implemented as 14 relational tables in MySQL: roles, users, periode_kurbans, kelompok_kurbans, peserta_kurbans, hewan_kurbans, golongan_penerimas, penerimas, lcg_configs, qr_codes, distribusi, log_scans, laporans, and kupon_blankos. These tables were connected through one-to-many and one-to-zero-or-one relationships, ensuring that each QR code, distribution record, and scan attempt could be traced to the corresponding recipient, generating user, and qurban period.

The design emphasized modularity, traceability, and role-based control. Each actor was assigned access only to functions relevant to their responsibilities, thereby reducing the risk of unauthorized data modification and minimizing overlap between data collection, verification, distribution, and reporting activities. Recipient and distribution data were linked through unique identifiers, while QR code generation and scan logs provided a verifiable record of each transaction. This structure allowed the system to distinguish between valid, previously used, expired, and unregistered QR codes during real-time verification.

Figure 2. Use-case diagram of SiKurban, showing five actors: Admin, Data-Collection Committee, Distribution Committee, Surau Management, and Meat Recipient.

The separation of recipient data, QR code records, distribution transactions, and scan logs also supported auditability. Every verification attempt could be recorded with its time, status, related recipient, and responsible user. Consequently, administrators and committee members could review distribution progress, identify duplicate or failed scans, verify the remaining allocation, and generate reports without manually reconciling paper coupons. The design therefore provided a consistent data flow from recipient registration and QR code generation to verification, distribution, monitoring, and reporting

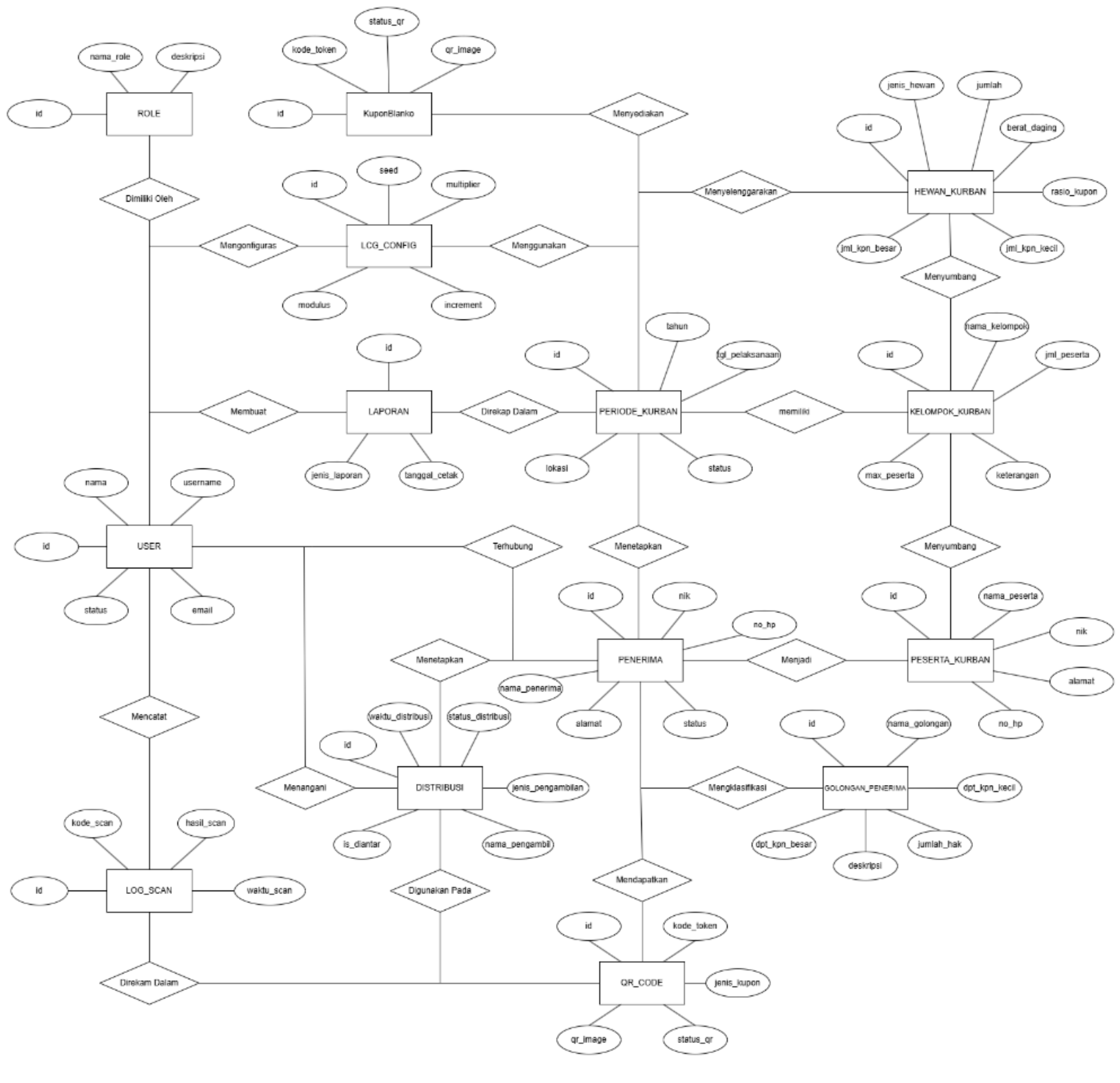


Figure 3. Entity–relationship diagram of the SiKurban database, comprising 14 interconnected entities.

2.4.3. Unique Token Generation Using a Linear Congruential Generator

The core novelty of SiKurban is the generation of a controlled, non-sequential initial code for every QR token using a Linear Congruential Generator, implemented in an LcgService class. The LCG recursive formula used is shown in Equation (1).

$$Xn + 1 = (a \, . Xn + c) \bmod m \qquad (1)$$

where Xn is the current seed value, a is the multiplier, c is the increment, and m is the modulus. Table 2 lists the parameter values used in SiKurban. The multiplier and increment in Table 2 are the

Numerical Recipes constants. For a mixed LCG with modulus 2^{32}, the Hull–Dobell full-period conditions require an odd increment and a multiplier for which (a − 1) is divisible by 4 [10].

Table 2. LCG parameters used in the SiKurban token generator

Parameter	Symbol	Value
Multiplier	a	1,664,525
Increment	c	1,013,904,223
Modulus	m	2^{32} = 4,294,967,296
Initial seed (example)	X0	Stored per qurban period in lcg_configs

With state preserved between calls, the sequence therefore traverses 2^{32} states before repeating. Full period does not imply cryptographic unpredictability, so SiKurban used the LCG only to allocate identifiers and applied database and keyed-digest controls separately. Starting from X_0 = 3,975,605,359, the first iteration was X_1 = (1,664,525 × 3,975,605,359 + 1,013,904,223) mod 4,294,967,296 = 3,187,945,218. Its base-36 representation (1GQ0Q9U) produced the token QR-2025-1GQ0Q9U. Table 3 reports the ten sequential iterations recalculated from the stated seed and parameters. For a continuously persisted sequence within a period, the full-period configuration structurally prevents a repeated state before 2^{32} iterations. This within-period guarantee does not exclude overlapping numeric suffixes after a seed reset or across independently seeded periods; the period prefix and global database uniqueness check therefore remain necessary.

Table 3. First ten LCG iterations and the resulting QR-code tokens

n	Xn (decimal)	Base-36 code	QR token
1	3,187,945,218	1GQ0Q9U	QR-2025-1GQ0Q9U
2	613,556,857	A5ANI1	QR-2025-A5ANI1
3	2,942,822,788	1CO2WLG	QR-2025-1CO2WLG
4	1,503,946,515	OVESK3	QR-2025-OVESK3
5	1,538,572,630	PG0Y8M	QR-2025-PG0Y8M
6	3,406,497,981	1KC52NX	QR-2025-1KC52NX
7	3,421,418,232	1KL0V7C	QR-2025-1KL0V7C
8	756,341,239	CIB0MV	QR-2025-CIB0MV
9	511,012,586	8G8RU2	QR-2025-8G8RU2
10	1,735,446,849	SP8NCX	QR-2025-SP8NCX

Note. The values form a deterministic sequence generated from the stated seed; they are not cryptographic random values. Each token was also checked against the database and assigned an HMAC-SHA256 digest before issuance (Sections 2.4.3–2.4.4). For this revision, rows 1–10 were recomputed sequentially from X_0 using both Python integer arithmetic and JavaScript BigInt, and the decimal and base-36 outputs agreed. Table 3 is a formula-derived verification table, not a dump of production qr_codes records.

The LCG was retained in this prototype to provide a deterministic, reproducible identifier sequence with explicit state tracking, not because it offers stronger security or simpler collision handling than a cryptographically secure pseudorandom number generator (CSPRNG). Laravel/PHP can instead use CSPRNG-backed values, such as random_bytes()-derived identifiers or UUIDs, without maintaining deterministic LCG state. Such alternatives should be evaluated for production deployment when unpredictability is the primary requirement; this study does not claim that the LCG is superior to them.

Every generated value was validated against all existing tokens in the qr_codes table before being accepted; if a collision occurred, the generator iterated again with the updated seed, up to a maximum of 1,000 attempts, inside a database transaction protected by row-level locking (lockForUpdate()) to prevent two concurrent generation requests on the same period from producing a duplicate token.

2.4.4. Token Security

For token integrity checking, the application generated an HMAC-SHA256 digest of the token, qurban period identifier, and coupon type, using the Laravel application key (APP_KEY) as the secret [25]. The digest was stored with the token and recomputed during scanning; hash_equals() provided a constant-time comparison. This mechanism did not make the LCG sequence unpredictable and was evaluated only through the functional scenarios reported in this study, not through adversarial penetration testing.

2.4.5. One-Time-Use Verification Flow

When a distribution-committee member scanned a QR code, the system read the token and queried its status in the database in real time. Four conditions were distinguished: (i) a registered token with an active validity period and status "not yet used," for which the system displayed the recipient's data and offered hand-over confirmation; (ii) a registered token already marked "used," for which the system rejected the scan, displayed the time and committee member of the earlier valid transaction, and logged the attempt as a duplication; (iii) a registered but expired token, for which the scan was rejected and the committee was directed to request a re-generated code; and (iv) an unregistered token, for which the scan was rejected and logged as an anomaly. Upon a valid hand-over confirmation, the status changed to "used," the creation of the distribution record, and the scan-log entry were executed together in a single atomic database transaction protected by row-level locking, so that the QR-code status and the distribution record could never become inconsistent, even under near-simultaneous scans from two devices.

2.5. Testing and Data Analysis Technique

System verification used developer-executed black-box testing, in which the running application received specified inputs and its outputs were compared with expected results without using internal code structure as the test oracle [26]. Scenarios were derived from the 28 functional requirements and covered all five roles, as well as negative or edge conditions central to one-time-use verification, including repeated and unregistered tokens. A scenario passed when the observed output matched the predefined expected output; failed scenarios were corrected and retested. The 33 scenarios were executed on 1 July 2026. This procedure assessed functional conformance only; it did not measure usability, throughput, security resistance, user acceptance, or live-field effectiveness. All scenarios were designed and executed by the developer; no committee member or independent tester reran them. Accordingly, the 100% result is reported as developer-verified functional conformance, not user-acceptance testing or field validation.

3. RESULTS

3.1. System Implementation Overview

Following the design in Section 2.4.2, the Laravel migration created all 14 database tables, and the system was implemented with five role-based interfaces: Admin, Data-Collection Committee, Distribution Committee, Surau Management, and a QR-code portal for Meat Recipients accessed through a National Identity Number (NIK) login rather than a conventional account, since most recipients do not have an institutional email address. The public landing page and login page used an emerald-green, dark-mode visual theme intended to evoke an Islamic identity. They displayed four live statistics — the number of qurban groups, participants, generated QR tokens, and registered recipients — retrieved directly from the database. Figure 4 shows the Distribution-Committee dashboard used on the day of distribution, which summarized overall progress (total, collected, and not-yet-collected recipients) together with QR-code integrity indicators (valid, used, and invalid/duplicate scan attempts) to support real-time field decision-making.

In the current prototype, the recipient's NIK also served as the default credential. Because an NIK may be known to relatives or local officials or exposed through data leakage, NIK-only authentication is a current security limitation. Production deployment should require a forced credential change at first login or a separate one-time-password factor and should minimize the personal status information displayed before stronger authentication is completed.

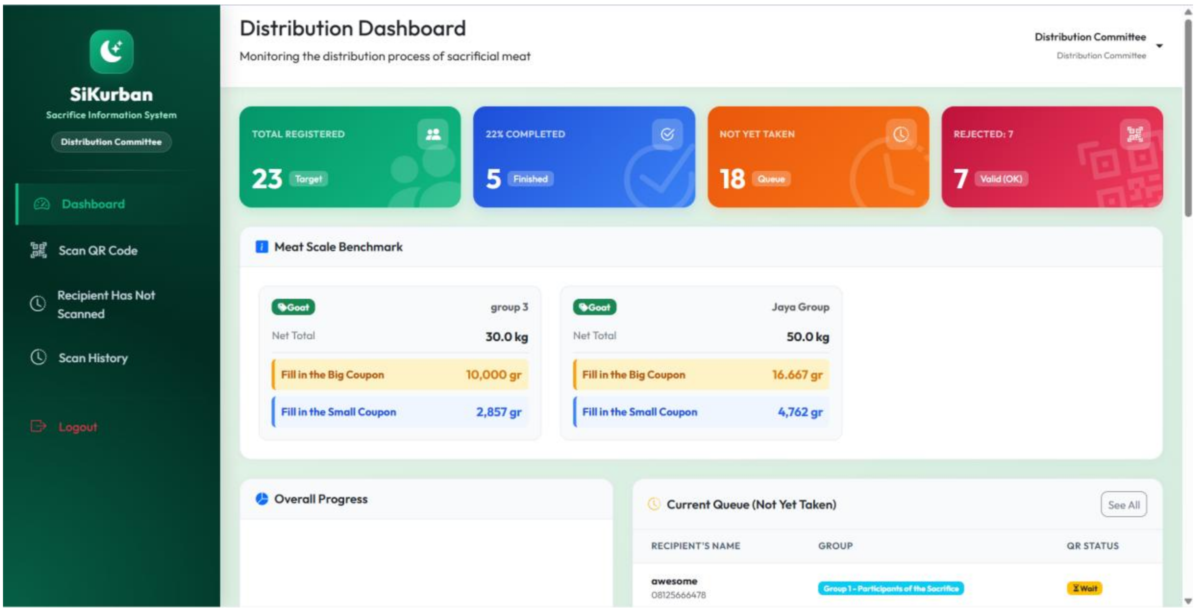


Figure 4. Distribution-Committee dashboard, showing real-time recipient progress and QR-code integrity indicators.

3.2. LCG-Based Unique Token Generation

The LcgService class implemented generate(), which performed Equation (1); generateUniqueToken(), which generated and checked identifiers inside a locked database transaction; and generateSignature(), which produced the HMAC-SHA256 digest described in Section 2.4.4 (Figure 5). Recalculating the first 10 outputs from the documented seed reproduced the corrected values in Table 3. The developer reported no duplicate in the 448-token prototype generation run; no independent export of qr_codes rows or generation logs was available for audit.

Vol. 4, No. 2, June (2026), Pages 182–201 | e-ISSN: 2985–9166

Within a continuously persisted period sequence, the full-period configuration structurally prevents a repeated state before 2^{32} iterations; this is not a probabilistic 'negligible collision' claim. Across seed resets or independently seeded periods, suffix overlap is not excluded, so the period prefix and global database check guard against resets, imported records, and concurrent requests.

```php
// LcgService.php — Fungsi inti LCG
// Rumus: Xn+1 = (a × Xn + c) mod m
public function generate(int $seed, int $a, int $c, int $m): int
{
    return (($a * $seed) + $c) % $m;
}

// Membangkitkan token unik dengan collision detection
public function generateUniqueToken(int $periodeId): string
{
    return DB::transaction(function () use ($periodeId) {
        // Kunci baris lcg_configs agar aman dari race condition
        $config = LcgConfig::where('periode_id', $periodeId)
                      ->lockForUpdate()->first();

        $seed       = (int) $config->seed;        // 3.975.605.359
        $multiplier = (int) $config->multiplier;  // 1.664.525
        $increment  = (int) $config->increment;   // 1.013.904.223
        $modulus    = (int) $config->modulus;     // 4.294.967.296

        for ($i = 0; $i < 1000; $i++) {
            // Hitung bilangan acak berikutnya dengan rumus LCG
            $nextSeed = $this->generate($seed, $multiplier, $increment, $modulus);

            // Format token: QR-[TAHUN]-[BASE36]
            $tokenStr = 'QR-' . date('Y') . '-'
                      . strtoupper(base_convert($nextSeed, 10, 36));

            // Collision detection: cek keunikan token di basis data
            if (!QrCode::where('kode_token', $tokenStr)->exists()) {
                $config->update(['seed' => $nextSeed]); // perbarui seed
                return $tokenStr;  // token unik ditemukan
            }
            $seed = $nextSeed; // collision → lanjut iterasi
        }
        throw new RuntimeException('Gagal generate token unik.');
    });
}

// HMAC-SHA256 signature untuk integritas token
public function generateSignature(
    string $token, int $periodeId, string $jenisKupon
): string {
    // Payload: token|periodeId|jenisKupon
    $payload = $token . '|' . $periodeId . '|' . $jenisKupon;
    return hash_hmac('sha256', $payload, config('app.key'));
}
```

Figure 5. Excerpt of the LcgService.php implementation, showing the LCG calculation and unique-token generation logic.

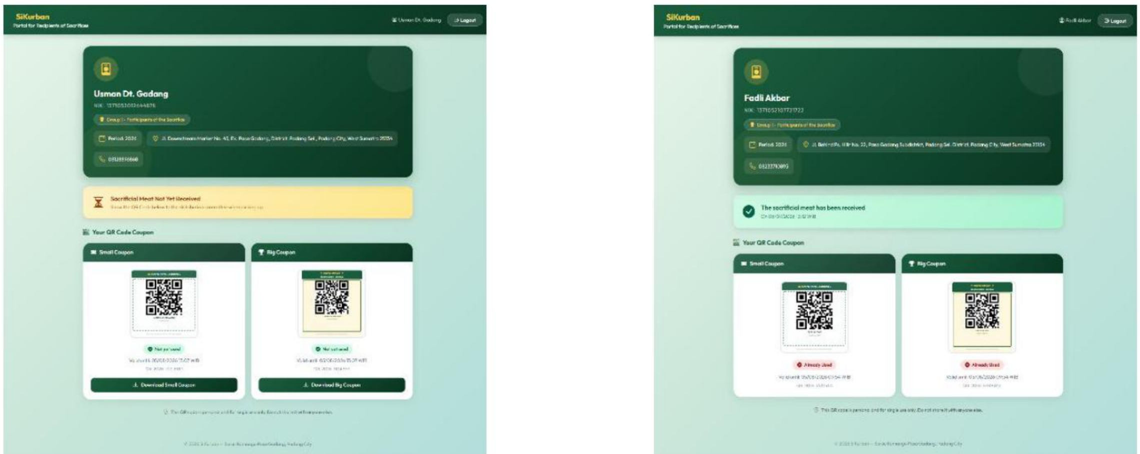
3.3. One-Time-Use QR Verification Outcomes

Table 4 summarizes the four verification conditions implemented in Section 2.4.5, along with their corresponding system responses observed during system testing.

Table 4. Real-time QR-code verification conditions and system responses

Token condition	System response	Outcome
Registered, active, "not yet used"	Display recipient data; allow hand-over confirmation	Accepted
Registered, status "used"	Reject the scan; show the time and the committee member from the earlier transaction; log as a duplication.	Rejected
Registered, expired	Reject the scan; direct the committee to request a regenerated code	Rejected
Unregistered token	Reject the scan; log as an anomaly	Rejected

Figure 6 illustrates the recipient-facing evidence of the one-time-use mechanism, comparing the QR-code portal of a recipient before collection, showing a yellow "meat not yet received" banner and a "not yet used" badge on each coupon, with the same portal after a distribution-committee member confirmed hand-over, showing the banner and badges permanently switched to "meat already received"/"already used."



(a) (b)

Figure 6. Recipient QR-code portal: (a) before collection, showing coupons marked "not yet used"; (b) after distribution-committee confirmation, showing coupons marked "already used."

Figure 7 shows the outcome when an already-used QR code was scanned a second time: the scanner interface displayed a warning that the code had already been collected, together with the timestamp of the original, valid transaction, while the internal scan counter confirmed that the attempt was logged but not treated as a valid collection.

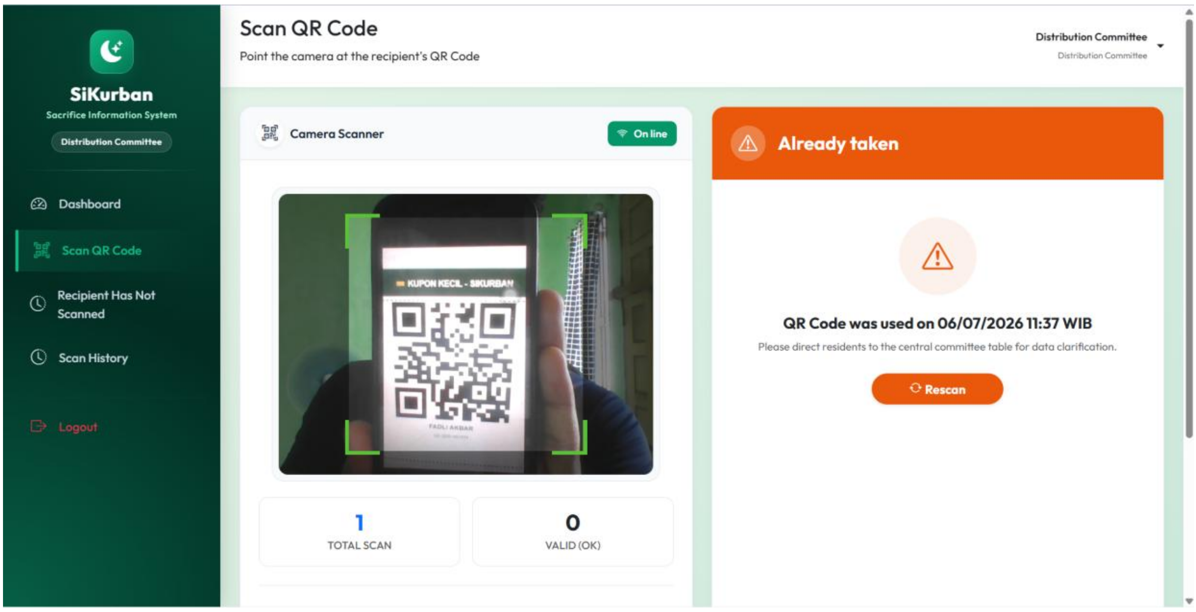


Figure 7. Scanner interface response when an already-used QR code was scanned a second time, showing rejection of the duplicate attempt.

3.4. Recipient-Category Entitlement Outcomes

The implemented classification rule assigned Category 1 when a person was registered both as a qurban participant and a meat recipient, and Category 2 when the person was registered only as a recipient. In the tested workflow, the selected category was stored with the recipient record, carried into QR-token issuance, and used to display the corresponding entitlement before hand-over confirmation. Thus, a participant-recipient record followed the Category 1 rule, whereas a recipient-only record followed the Category 2 rule. The nine Data-Collection Committee scenarios summarized in Table 5 included recipient, category, and QR-generation functions and matched the developer-defined expected outputs. These observations demonstrate implemented rule execution only; allocation accuracy, fairness, and committee usability were not independently evaluated.

3.5. Developer-Executed Functional Testing Results

Black-box testing covered 33 scenarios across the five user roles: 28 positive scenarios for specified functions and five negative or edge scenarios for invalid login, duplicate QR issuance, reuse of a consumed token, an unregistered token, and deletion of recipient data with an existing distribution history. The developer recorded that all 33 scenarios produced the predefined expected outputs. This 100% scenario-pass rate denotes developer-verified functional conformance rather than independent validation or user-acceptance testing. Table 5 reports the module-level results.

Table 5. Summary of developer-executed black-box testing results by module

Module	Scenarios	Passed	Failed	Success rate
Authentication (login/logout, all roles)	4	4	0	100%
Admin module (accounts, periods)	4	4	0	100%
Data-Collection Committee module (participants, groups, animals, recipients, categories, QR generation)	9	9	0	100%
Distribution Committee module (scanning, verification, hand-over confirmation)	6	6	0	100%
Surau Management module (monitoring, reporting)	4	4	0	100%
Meat Recipient portal	1	1	0	100%
Negative / edge-case scenarios (anti-duplication, invalid input)	5	5	0	100%
Total	33	33	0	100%

4. Discussion

All 33 black-box scenarios produced their predefined outputs, including rejection of reused and unregistered tokens. This result supports functional correctness for the specified cases in the development test environment; it does not demonstrate defect-free operation, scalability, usability, or field effectiveness. The result is consistent with barcode-based qurban studies in which a database-checked status helped identify completed and repeated collections [6], [13]. SiKurban extended that design by allowing smartphone access to the QR portal (Figure 6) while retaining a printable fallback. Consequently, dependence on paper was reduced but not eliminated. Printed codes also remain susceptible to copying or substitution, which has motivated tamper-resistant QR-code research beyond the qurban domain [27].

Compared with the five related qurban systems reviewed in the Introduction [14], [15], [16], [17], [18], SiKurban integrated identifier allocation, database collision handling, a keyed digest, one-time-use status control, and category-based entitlement. Prior applications of LCGs have emphasized their computational simplicity in examination randomization, order-code generation, quiz applications, and hybrid pseudo-random generators [28], [29], [30], [31]. The present study used that simplicity for identifier allocation rather than claiming cryptographic randomness. Table 6 consolidates this comparison and distinguishes features not reported in the cited publications from features demonstrated by SiKurban.

Table 6. Feature comparison of related qurban distribution systems and SiKurban

Study/system	Identifier or code method	Physical carrier	One-time-use control	Real-time verification	Category entitlement
Lewenusa [6]	Barcode; generation method NR	Printed coupon	Duplicate-scan control reported	Scan/status checking reported	NR
Erwansyah et al. [13]	Barcode; generation method NR	Printed coupon	Status changed after collection	Collected/uncollected status updated	NR
Heldiansyah et al. [14]	NR	Web-based distribution records	NR	NR	NR
Ruhyana et al. [15]	NR	Digital management system	NR	NR	NR
Andika and Haryono [16]	NR	Digital management system	NR	NR	NR
Maulindar et al. [17]	NR	Web-based management system	NR	NR	NR
Azzahra and Wicaksono [18]	NR	Web-based animal-data management	NR	NR	NR
SiKurban	LCG state to base-36; DB uniqueness check; HMAC	Digital QR portal with printable fallback	Atomic available-to-used transition	Locked transaction and scan log	Category 1 and Category 2

Note. NR = not reported in the cited publication. NR does not establish that a feature was absent; the comparison is limited to features explicitly described in [6], [13]–[18].

The technical role of the LCG also requires careful interpretation. Full-period conditions describe cycle length [32], whereas the statistical and lattice properties of congruential generators require separate evaluation [33], [34]. Accordingly, SiKurban did not rely on the LCG for secrecy. HMAC supplied a keyed integrity mechanism [35], while the database transaction and row lock were intended to serialize competing redemption updates; transaction-isolation research shows why these concurrency assumptions must be made explicit and tested [36].

QR-code security also extends beyond duplicate status. QR codes are untrusted input carriers, and cryptographic protection can introduce a security–usability trade-off by requiring denser payloads and more scanning constraints [37]. SiKurban's functional tests did not evaluate malicious payloads, copied valid code, key compromise, replay across periods, or signature-handling errors. These cases should be included in a future threat model and independent security assessment. Recipient authentication introduces an additional present risk because the prototype uses NIK as the default credential. NIK-only access should not be treated as adequate production authentication; a forced first-login credential change or OTP-based second factor should be implemented and verified before live deployment.

The evaluation boundary is equally important. Information-system success includes system quality, information quality, use, user satisfaction, and net benefits, not only functional output [38]. Intended users conduct user acceptance testing in a realistic setting to determine whether software meets operational needs [39]. Thus, the present 100% test result should be read as a scenario-pass rate, and the next evaluation should compare SiKurban with the paper process using completion time, duplicate or invalid-claim handling, correction frequency, queue duration, and user-reported usability.

Finally, smartphone access does not by itself ensure equitable use. Digital-divide research distinguishes device or Internet access from differences in skills, use, and beneficial outcomes [40]. Retaining printable or assisted-verification options is therefore an inclusion safeguard, not merely a temporary implementation detail.

Operationally, a browser-based scanner can be deployed on ordinary smartphones without dedicated scanning hardware. Its reliance on real-time database checks, however, makes connectivity a material operating condition. The offline-ready requirement should therefore be validated through interruption, synchronization, and conflict-recovery tests before live multi-device deployment.

This study had eight principal limitations. First, it concerned one site and one qurban period, so transferability to larger or multi-location distributions is unknown. Second, testing was executed by the developer rather than independent testers or committee users. Third, throughput, concurrent multi-device scans, and unstable-network recovery were not stress-tested. Fourth, the study did not conduct penetration testing, a key management review, or adversarial analysis of token guessing and QR-code substitution. Fifth, it did not measure task time, error reduction, usability, satisfaction, adoption, distribution outcomes, or cost relative to the paper process. Sixth, the 13 non-functional requirements remained untested design targets. Seventh, NIK-only recipient authentication is a present vulnerability and is not suitable for production deployment without stronger credentials. Eighth, the zero-duplicate generation result relied on developer-observed application output; the underlying database rows and generation logs were not independently archived or audited.

Future work should prioritize independent user-acceptance testing with committee members during a live or high-fidelity distribution exercise, followed by load, concurrency, offline synchronization, and security testing. Before any live deployment, NIK-only recipient authentication should be replaced by a forced first-login credential change or an OTP factor, and administrative accounts should use two-factor authentication. Subsequent product extensions may include consent-based WhatsApp or SMS delivery, multi-location coordination, and cross-period analytics. Any digital-only workflow should retain an accessible fallback for recipients who lack compatible devices, connectivity, or digital skills.

5. CONCLUSION

This study designed and functionally evaluated SiKurban, a Laravel–MySQL web application for qurban meat distribution at Surau Kumango Pasa Gadang. The system combined deterministic LCG-based identifier allocation, database uniqueness checks, an HMAC-SHA256 digest, atomic one-time-use status transitions, and category-based entitlement. Recalculation corrected the documented ten-value LCG sequence, while all 33 developer-executed black-box scenarios produced their predefined outputs, including rejection of reused and unregistered tokens. These results establish developer-verified functional conformance under the specified conditions, not independent validation, user acceptance, non-functional quality, security resistance, or effectiveness in live distribution. SiKurban should therefore be regarded as a promising single-site prototype rather than a demonstrated field-effective alternative. Broader deployment requires independent user acceptance, comparative field and cost evidence, non-functional evaluation, scalability, offline recovery, and security testing before broader deployment.

DECLARATIONS

Author Contributions

Farras Fathin Naufal: Conceptualization, Methodology, Software, Investigation, Data Curation, Visualization, Writing – Original Draft. **Dedy Irfan:** Conceptualization, Supervision, Validation, Writing – Review & Editing. **Delvi Asmara:** Validation, Writing – Review & Editing. All authors have read and approved the final version of the manuscript.

Acknowledgments

The authors would like to thank the management and qurban committee of Surau Kumango Pasa Gadang, Padang, West Sumatra, for their cooperation during the requirements analysis and provision of operational context, and the Department of Informatics Engineering Education, Universitas Negeri Padang, for their academic support.

Ethical Approval

No clinical trials or experimental interventions on human or animal subjects were conducted in this study. The research relied solely on field observations, informal interviews with qurban committee members for requirements gathering, and anonymized operational data from system testing. Consequently, formal ethical clearance was deemed unnecessary.

Informed Consent

Verbal informed consent was obtained from all qurban committee members who participated in the requirements-analysis interviews and system demonstrations. Personal data of qurban meat recipients (e.g., name and National Identity Number/NIK) used to demonstrate the system were either anonymized or used with the consent of the individuals concerned, and are not disclosed in this manuscript beyond illustrative first names already shown in the source thesis.

Funding

This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors. It was conducted as part of the first author's undergraduate thesis at Universitas Negeri Padang.

Data Availability Statement

The data supporting this study's findings are not publicly available. Summary data are included in the manuscript, and more detailed information can be requested from the corresponding author under strict confidentiality agreements.

Competing Interests

The authors declare that they have no competing interests related to the content of this article.

Generative AI and AI-Assisted Technologies Statement

During the preparation of this manuscript, the authors used AI (Claude) to assist with summarizing data, as explained in the manuscript. The authors reviewed, verified, and edited all AI-assisted content, including all citations and numerical values against the original thesis, and take full responsibility for the accuracy and integrity of the published work.

REFERENCES

[1] M. S. Apriantoro, R. A. Zaky, and V. D. Febrianti, "Participatory planning assistance for the qurban savings and joint-purchase program to increase interest in qurban during the pandemic," *Jurnal Pengabdian kepada Masyarakat*, vol. 2, no. 1, pp. 29–39, 2022, doi: 10.35870/ajad.v2i1.27.

[2] Y. Purwanto, A. N. Fitriani, Syachrial, and A. Aziz, "The influence of innovation in sacrificial worship services and digital marketing on millennial and Gen Z donation tendencies," *Journal of Islamic Philanthropy & Social Finance*, vol. 6, no. 1, pp. 83–93, 2024, doi: 10.24191/JIPSF/v6n12024.

[3] B. Ardianto, Jaroji, and M. A. Subandri, "Development of the SIMASJID application using the extreme programming method," *Jurnal Penelitian Rumpun Ilmu Teknik*, vol. 3, no. 3, pp. 187–202, 2024, doi: 10.55606/juprit.v3i3.4249.

[4] S. N. Wahab, N. M. Sayuti, M. L. Mohaini, and H. Ismail, "Examining efficiency and hygiene in qurban meat packaging and distribution," *Environment-Behaviour Proceedings Journal*, vol. 10, no. 33, pp. 29–31, 2025, doi: 10.21834/e-bpj.v10i33.7276.

[5] D. Effendi, R. P. Dhaniawaty, M. Fitriawati, D. Cancerina, and A. Ramadhan, "Design and development of a qurban service information system using the research and development method," *Journal of Engineering Science and Technology*, vol. 19, no. 5, pp. 1588–1599, 2024.

[6] I. Lewenusa, "Implementation of a barcode-based queue system for qurban meat distribution at Musala Nurul Iman, South Jakarta," *Jurnal Serina Abdimas*, vol. 2, no. 3, pp. 1084–1091, 2024, doi: 10.24912/jsa.v2i3.32080.

[7] J. Han, Y. Son, and H. Eom, "A secure e-coupon service based on blockchain systems," *IEEE Access*, vol. 10, pp. 21836–21846, 2022, doi: 10.1109/ACCESS.2022.3152765.

[8] A. M. Laksono, "Development of a digital qurban meat coupon authenticity verification system using AES encryption," undergraduate thesis, Universitas Muhammadiyah Yogyakarta, Yogyakarta, Indonesia, 2025

[9] Ministry of Administrative and Bureaucratic Reform of the Republic of Indonesia, "Addressing the risk of data loss and leakage, the governments of Indonesia and the Republic of Korea develop verifiable credentials," 2025. [Online]. Available: https://www.menpan.go.id/site/berita-terkini/atasi-risiko-kehilangan-dan-kebocoran-data-pemerintah-indonesia-dan-republik-korea-kembangkan-verifiable-credentials. [Accessed: Jul. 20, 2026]

[10] O. K. Durrani, M. S. Ali, D. S. Makandar, H. T. P., G. Bano, and D. Begum, "Performance analysis of linear congruential random generator algorithms using Python and Java languages," *Journal of Advances in Mathematics and Computer Science*, vol. 40, no. 2, pp. 40–52, 2025, doi: 10.9734/jamcs/2025/v40i21968

[11] Takyudin, "Optimizing digital-based qurban management through community service at Masjid Al-Jauhar," *Jurnal Pengabdian kepada Masyarakat*, vol. 5, no. 1, pp. 27–31, 2025

[12] W. N. A. Wan Mohamed Salleh, S. Z. Abdul Rasid, and R. Basiruddin, "Optimising digital technology in managing zakat," *International Journal of Academic Research in Business and Social Sciences*, vol. 12, no. 8, pp. 698–711, 2022, doi: 10.6007/IJARBSS/v12-i8/14355

[13] K. Erwansyah, M. Zunaidi, F. Taufik, Tugiono, and Hafizah, "Implementation of a barcode system for Eid al-Adha 1446 H qurban meat distribution coupons at Musholla An-Nur, Marindal I Village, Patumbak, Deli Serdang," *Jurnal Abdimas TGD*, vol. 5, no. 2, pp. 196–204, 2025

[14] Heldiansyah, N. Shintia, Rustaniah, and M. Salim, "A web-based meat distribution information system model for qurban worship committees," *Jurnal Ilmiah Teknik Informatika dan Sistem Informasi*, vol. 11, no. 1, p. 241, 2022, doi: 10.35889/jutisi.v11i1.801.

[15] N. Ruhyana, A. O. Sari, T. Mardiana, A. Bayhaqy, A. A. Riyadi, and Setiaji, "Design and development of a qurban distribution management information system," *Inti Nusa Mandiri*, vol. 20, no. 1, pp. 11–17, 2025, doi: 10.33480/inti.v20i1.6945.

[16] R. Andika and K. Haryono, "Qurban management information system to improve the administrative governance of qurban implementation," *Prosiding Automata*, vol. 3, no. 2, 2022. [Online]. Available: https://journal.uii.ac.id/AUTOMATA/article/view/24230. [Accessed: Jul. 20, 2026].

[17] J. Maulindar, Y. Bahtiar, T. M. Diva, and S. Umar, "Web-based qurban management system at Masjid Al-Falah, Kebak Wirun, Mojolaban," *Jurnal Manajemen Informatika & Sistem Informasi*, vol. 5, pp. 140–150, 2022

[18] S. A. Azzahra and A. Wicaksono, "Digital transformation in web-based qurban animal data management at Masjid Baiturahhman," *Jurnal Sains dan Teknologi*, vol. 3, no. 1, pp. 45–54, 2024, doi: 10.58169/saintek.v3i1.542.

[19] A. Saravanos and M. X. Curinga, "Simulating the software development lifecycle: The waterfall model," *Applied System Innovation*, vol. 6, no. 6, Art. no. 108, 2023, doi: 10.3390/asi6060108.

[20] Z. Subecz, "Web development with the Laravel framework," *Gradus*, vol. 8, no. 1, pp. 211–218, 2021, doi: 10.47833/2021.1.csc.006.

[21] T. S. Huynh, D. T. Tran, Q. H. Vu, and L. A. T. Nguyen, "Design and implementation of a web application based on MVC Laravel architecture," *European Journal of Electrical Engineering and Computer Science*, vol. 6, no. 4, pp. 37–42, 2021, doi: 10.24018/ejece.2021.6.4.448.

[22] O. Ekhlakov, "PHP: Methodology for configuring third-party Composer packages," *The American Journal of Engineering and Technology*, vol. 7, no. 6, pp. 56–65, 2025, doi: 10.37547/tajet/volume07issue06-06

[23] N. D. Kulkarni and S. Bansal, "System analysis to UML—Restaurant case study," *Journal of Artificial Intelligence & Cloud Computing*, vol. 1, no. 1, 2022, doi: 10.47363/JAICC/2022(1)196.

[24] S. M. Pulungan, R. Febrianti, T. Lestari, N. Gurning, and N. Fitriana, "Analysis of the entity-relationship diagram technique in database design," *Jurnal Ekonomi Manajemen dan Bisnis*, vol. 1, no. 2, pp. 98–102, 2023, doi: 10.47233/jemb.v1i2.533.

[25] M. Rana, A. Pandey, A. Mishra, and V. Kandu, "Enhancing data security: A comprehensive study on the efficacy of JSON Web Token and HMAC-SHA-256 algorithm for web application security," *International Journal of Recent Innovation and Trends in Computing and Communication*, vol. 11, no. 9, pp. 4409–4416, 2023, doi: 10.17762/ijritcc.v11i9.9930.

[26] Z. M. Alshamaa and N. N. Saleem, "Black-box software testing techniques: A literature review," *Passer Journal of Basic and Applied Sciences*, vol. 7, no. 2, pp. 1001–1011, 2025, doi: 10.24271/PSR.2025.508216.1963.

[27] S. A. Alsuhibany, "Innovative QR code system for tamper-proof generation and fraud-resistant verification," *Sensors*, vol. 25, no. 13, Art. no. 3855, 2025, doi: 10.3390/s25133855.

[28] A. Prakarsa, A. A. Sunarto, and Prajoko, "An online high-school examination randomization model using the linear congruential generator and Fisher–Yates methods," *Jurnal Ilmiah Komputer*, vol. 16, no. 2, pp. 133–142, 2020, doi: 10.35889/progresif.v16i2.519.

[29] R. D. Tjahjono, N. A. Prasita, and A. A. Widodo, "Implementation of a unique nominal-transfer code using the linear congruential generator method for order deposits," *Jurnal Informatika Merdeka Pasuruan*, vol. 1, no. 3, pp. 39–52, 2016.

[30] T. M. Zulfikar and L. Tanti, "Design of a TOEFL simulation quiz application using the linear congruential generator algorithm on Android," *Jurnal Info Digit*, vol. 2, no. 1, pp. 18–31, 2024.

[31] B. A. Hameedi, A. A. Hattab, and M. M. Laftah, "A pseudorandom number generator based on a new hybrid LFSR and LCG algorithm," *Iraqi Journal of Science*, vol. 63, no. 5, pp. 2230–2242, 2022, doi: 10.24996/ijs.2022.63.5.35.

[32] T. E. Hull and A. R. Dobell, "Random number generators," *SIAM Review*, vol. 4, no. 3, pp. 230–254, 1962, doi: 10.1137/1004061.

[33] P. L'Ecuyer, "Tables of linear congruential generators of different sizes and good lattice structure," *Mathematics of Computation*, vol. 68, no. 225, pp. 249–260, 1999, doi: 10.1090/S0025-5718-99-00996-5.

[34] S. K. Park and K. W. Miller, "Random number generators: Good ones are hard to find," *Communications of the ACM*, vol. 31, no. 10, pp. 1192–1201, 1988, doi: 10.1145/63039.63042.

[35] M. Bellare, R. Canetti, and H. Krawczyk, "Keying hash functions for message authentication," in *Advances in Cryptology—CRYPTO '96*, Berlin, Germany: Springer, 1996, pp. 1–15, doi: 10.1007/3-540-68697-5_1.

[36] H. Berenson, P. Bernstein, J. Gray, J. Melton, E. O'Neil, and P. O'Neil, "A critique of ANSI SQL isolation levels," in *Proceedings of the 1995 ACM SIGMOD International Conference on Management of Data*, 1995, pp. 1–10, doi: 10.1145/223784.223785.

[37] R. Focardi, F. L. Luccio, and H. A. M. Wahsheh, "Usable security for QR codes," *Journal of Information Security and Applications*, vol. 48, Art. no. 102369, 2019, doi: 10.1016/j.jisa.2019.102369.

[38] W. H. DeLone and E. R. McLean, "The DeLone and McLean model of information systems success: A ten-year update," *Journal of Management Information Systems*, vol. 19, no. 4, pp. 9–30, 2003, doi: 10.1080/07421222.2003.11045748.

[39] I. Otaduy and O. Diaz, "User acceptance testing for Agile-developed web-based applications: Empowering customers through wikis and mind maps," *Journal of Systems and Software*, vol. 133, pp. 212–229, 2017, doi: 10.1016/j.jss.2017.01.002.

[40] A. Scheerder, A. van Deursen, and J. van Dijk, "Determinants of Internet skills, uses, and outcomes: A systematic review of the second- and third-level digital divide," *Telematics and Informatics*, vol. 34, no. 8, pp. 1607–1624, 2017, doi: 10.1016/j.tele.2017.07.007.

AUTHOR BIOGRAPHIES

Farras Fathin Naufal is an undergraduate student in the Informatics Engineering Education Program at the Faculty of Engineering, Universitas Negeri Padang, Indonesia. His academic interests lie in web development and artificial intelligence, with a focus on applying these technologies to solve practical problems in digital system design. In this study, he contributed to the development of the qurban information system, particularly in the design and implementation of a QR-code-based distribution mechanism and a real-time verification feature to prevent duplicate scans and ensure accurate, traceable hand-over of qurban meat to recipients.

Dedy Irfan is a lecturer and researcher in the Department of Informatics Engineering Education, Faculty of Engineering, Universitas Negeri Padang (UNP). His academic expertise spans information systems, educational technology, and web-based application development. Driven by a commitment to advancing modern education, his research primarily focuses on applied computing, technology-enhanced learning, and system development within the context of engineering education.

Delvi Asmara is an academic and researcher in the Department of Electronic Engineering at Universitas Negeri Padang (UNP), dedicated to bridging the intersection of education and information technology. With an academic background that blends pedagogy and computer science up to the doctoral level, his work focuses on teaching, authoring reference books, and conducting applied research in information systems and network security. For him, educating goes beyond transmitting coding theories; it is about shaping an adaptive, innovative young generation equipped to create future digital solutions.