

Design and Development of *Pakai Lagi*: Buyer-Centered Interface for Second-Hand Marketplace

Dadan Rahmat^{1*}, Hapini Awang², Laili Wahyunita³, Sintia Pebrianti⁴

^{1,4} Information Technology Education, Faculty of Teacher Training and Education,
Universitas Muhammadiyah Sukabumi, Indonesia

² Institute for Advanced and Smart Digital Opportunities, School of Computing,
Universiti Utara Malaysia

³Department of Sharia Economic Law (Muamalah), Faculty of Sharia, State Islamic University (UIN)
Palangkaraya, Indonesia

ABSTRACT

Open Access

Type: Research Paper

Article:

Received: April 12, 2026

Revised: May 16, 2026

Accepted: August 13, 2026

Issued: October 30, 2026

© Rahmat et al. (2026).



This is an open-access article
under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license

***Correspondence Address:**

dadanrahmat@ummi.ac.id

The purchase of second-hand goods has become a popular trend in Indonesia due to social, economic and environmental factors. Second-hand goods offer sustainable solutions for waste reduction while providing economic value. However, transaction uncertainty and perceived quality risk remain significant barriers in online transactions. This study presents the design, development, and functional evaluation of *Pakai Lagi*, a localized second-hand marketplace built on client-server architecture. The system utilizes CodeIgniter 3 and MySQL for data administration and backend services, paired with a cross-platform Flutter mobile client for buyers. Development focuses on buyer-centric modules including product discovery, quality and condition displays, seller profiles and instant peer-to-peer communication to mitigate information asymmetry. Empirical evaluation was conducted using the black-box testing method across 14 functional areas comprising 49 test cases. The results demonstrate a functional feasibility for core operational modules. Certain peripheral presentation sub-components, specifically chat layout and complex filtering controls were noted as being under development for future iteration. Ultimately, this study confirms the practical feasibility of a lightweight, low-overhead backend-to-mobile infrastructure for sustainable community-driven e-commerce.

Keywords: *Second-hand Purchased, Online Marketplace, Buying and Selling Application, Website, Black-box Testing*

1. INTRODUCTION

The development of information technology has advanced substantially and has transformed various aspects of human life including work and social interactions [1], [2]. Among the most prominent manifestations of this technology are websites and mobile applications which function as key tools for communication and business in the current digital era [3]. These technologies support diverse purposes including buying and selling activities through electronic commerce (e-commerce). The development of web and mobile technologies has expanded opportunities for e-commerce by enabling consumers to discover products, access product information and communicate with sellers with greater convenience and transparency, without depending solely on physical marketplace locations.

This advancement is particularly relevant to second-hand commerce, which has become increasingly popular in various parts of the world, including Indonesia [3]. This consumer trend is shaped by combination of social, economic and environmental factors [4]. Economically, second-hand goods offer significant value as they are generally available at more affordable prices while maintaining adequate quality [5]. Environmentally, products that remain in usable condition represent an effective alternative for reducing the accumulation of waste by circulating goods rather than discarding them [6]. Beyond economic and environmental utility, second-hand goods possess distinct aesthetic and historical value, which fosters creativity by encouraging the reuse, recycling or modification of existing products to generate new artistic or functional value [7]. The availability and accessibility of these goods, ranging from clothing and electronics to household appliances, have increased substantially alongside advancements in internet technology and mobile connectivity [8], [9], [10]. Consequently, online marketplaces and buying and selling applications have further facilitated consumers' access to second-hand products based on their specific needs [4], [11].

However, purchasing second-hand products through digital platforms involves particular information challenges and perceived

risks [6]. The research in consumer behavior examined second-hand products may differ significantly from new products in condition, quality, potential hidden defect and previous use [5], [12], as well as social risks concerning peer judgement or perceived status loss [2]. Buyers cannot directly inspect these characteristics before completing a transaction and therefore depend heavily on information provided through the marketplace. Product description, photographs, condition information, price and seller profiles are consequently important for helping buyers evaluate available products [2], [10], [13]. Empirical studies indicate that negative quality perceptions often stemming from slightly faded colors, small stains or minor physical wear can hinder purchase adoption unless perceived risks are adequately minimized [14]. Therefore, establishing buyer trust requires dedicated platform features that maximize transparency, including structure product description, pictures, clear condition disclosures, pricing visibility and seller credibility. In addition, direct communication between buyers and sellers provides a vital opportunity for buyers to obtain further information and verification before making a transaction [15], [16].

These conditions indicate the need for digital marketplace that not only provides product listing but also supports structured discovery, access to detailed condition information and interaction between buyers and sellers. In the context of second-hand commerce, such functions can help organize the information required by buyers while providing sellers with a dynamic medium for presenting their products [7], [17]. Based on these needs, the *Pakai Lagi* platform was designed and developed as a dedicated marketplace for second-hand products that provides facilities for both sellers and buyers.

Prior research examining second-hand e-commerce, mobile application and system architecture has yielded important empirical and technical insights. The implementation of web-based e-commerce systems represents a widely adopted approach to facilitating product transactions and expanding access to second-hand products. From an architectural perspective, the study of "Software Architecture Using a Decoupling, Layered and Web Restful Service Approach" has demonstrated that isolating frontend interfaces from backend data

services yields significant advantages in system maintainability, modularity, and resource efficiency under concurrent workloads compared to tightly coupled monolithic architectures [18]. Furthermore, other research on backend REST APIs utilizing enterprise systems, as in [17], has confirmed that server-side API endpoints can successfully handle item listing management and product organization. Other studies have constructed second-hand trading platforms [3] by dividing responsibilities between mobile clients such as Android or Flutter and web-administration dashboards, which verified functional validity in improving user accessibility, data organization, and real-time data synchronization via REST APIs.

Despite the development of second-hand marketplace systems through web-based as in [2], [13], mobile-based and user-oriented approaches [9], implementations in the existing literature remain predominantly limited. Many campus- and community-based preloved platforms are constrained by single-platform designs. For instance, web-based preloved marketplaces successfully automate catalog browsing, search filtering, and administration, yet studies concluded that web-only accessibility constrains immediate, on-the-go user transactions [9], [15]. Furthermore, existing implementations differ widely in the target users, platform characteristics, and supported marketplace functions.

In particular, the combination of buyer-oriented product discovery, detailed product-condition information, seller-store access, favorites, and cross-platform buyer-seller communication across both web and mobile environments provides a specific implementation context for second-hand transactions [17]. Therefore, this study focuses on the design and development of the *Pakai Lagi* platform to address these integrated information and interaction needs in second-hand product transactions. The contribution of this study is explicitly positioned as a contextual and implementation-oriented contribution rather than as a new software framework or algorithm.

This manuscript presents the original results of the authors' development of the *Pakai Lagi* marketplace, showcasing the complete technical system design alongside the

functional validation of its buyer-centered features.

2. METHODS

The development of the *Pakai Lagi* marketplace platform follows the Agile Software Development Life Cycle (Agile SDLC) framework. This iterative approach was adopted to ensure architectural flexibility, enabling the dual-platform environment, comprising a centralized web-based administration backend and a dedicated mobile buyer client, to be engineered and refined through successive sprint cycles. In this study, the development is structured into five distinct phases as follows in **Figure 1**.

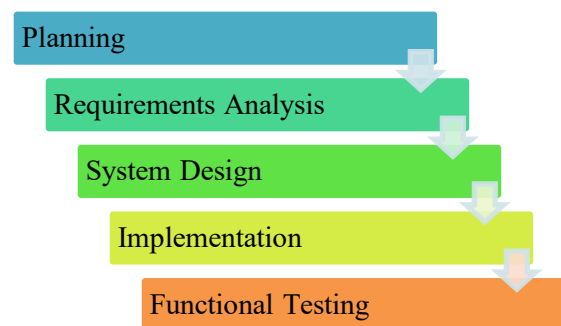


Figure 1. Development structured flow

- a. Planning
A decoupled client-server architecture was defined comprising a mobile buyer client using Flutter and a web-based administrative backend using CodeIgniter 3, PHP, and MySQL.
- b. Requirement analysis
Requirements were derived from a literature synthesis on information asymmetry in second-hand markets, resulting in 14 functional modules.
- c. System design
System design was modeled using flowcharts, an Entity Relationship Diagram (ERD), and Unified Modeling Language (UML) object models.
- d. Implementation
The mobile application was developed using the Flutter framework, while CodeIgniter 3 was utilized for backend and API development, and MySQL was employed for data management.

- e. Functional testing
The functional modules were evaluated using the Black-Box testing method. This evaluation method was intentionally designated to verify the functional feasibility of the implemented features rather than establish comprehensive usability, user satisfaction, performance, scalability, accessibility or security.

3. RESULTS AND DISCUSSION

3.1 Planning

The planning phase yielded a clearly defined architectural blueprint for a localized second-hand marketplace in Sukabumi. The primary output was a decoupled, dual-platform architecture that separates the mobile buyer client from the web-based administrative backend. This architectural decision explicitly reflects the distinction between transactional users (buyers) and ecosystem moderators (administrators), which is central to managing information asymmetry and maintaining trust in community-driven markets. Key planning outputs include:

1. Mobile buyer interface configuration based on the Flutter framework, restricted to core buyer-centric modules (catalog discovery, wear-and-tear condition display, favorite watchlists, and peer-to-peer messaging) to optimize on-the-go access to product-condition information.
2. Backend and administrative panel configuration based on CodeIgniter 3, PHP, and MySQL, designated as the central node for authentication, input filtering, and transaction-related logic over relational tables for users, products, transactions, and community forum boards.
3. Integration-layer protocol specifying that all client-server communication must occur through stateless HTTP API gateways, ensuring a clear separation of concern and supporting maintainability in resource-constrained deployments.

These planning artifacts established the structural boundaries within which all subsequent requirements, design models, and implementation decisions were developed.

3.2 Requirement Analysis

The requirement analysis phase produced a theoretically grounded set of functional modules derived from a synthesis of digital commerce literature on information asymmetry and perceived risk in used-goods markets. The main output was a catalog of fourteen functional modules that collectively address key sources of transaction uncertainty:

1. Account authentication (login)
2. User registration
3. Registration OTP
4. Password recovery
5. Password recovery OTP
6. New password configuration
7. Homepage dashboard
8. P2P communication (chat)
9. Product catalog
10. Product detail
11. Favorite management
12. Profile settings
13. System notification
14. Seller store profile

These modules operationalize the conceptual distinction between buyers and administrators and explicitly target information transparency and risk reduction. For example, the inclusion of dedicated product detail (with wear-and-tear condition fields), P2P communication, and seller store profile modules directly responds to literature indicating that uncertainty about product quality and seller credibility are primary barriers in second-hand transactions. The requirement set thus serves as a functional baseline that aligns system capabilities with the study's objective of reducing perceived transaction risk through clearer product information and more direct buyer-seller interaction.

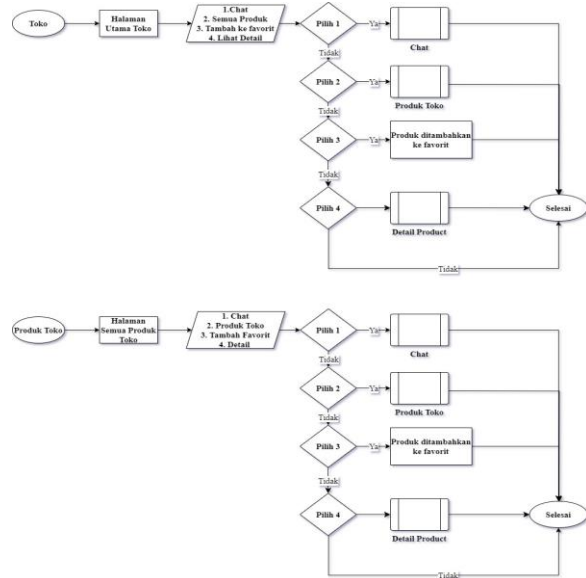


Figure 2. The platform's brief flowchart

3.3 System Design

The system design phase translated the planning architecture and functional requirements into formal visual models that guide implementation and support traceability [1]. The primary outputs were operational flowcharts documenting step-by-step workflows for landing-page catalog navigation, secure user registration, password-recovery sequences, and conversational chat routing paths, which operationalize the buyer journey and administrative processes and ensure that critical paths for product discovery, authentication, and communication are explicitly modeled [2], [7].

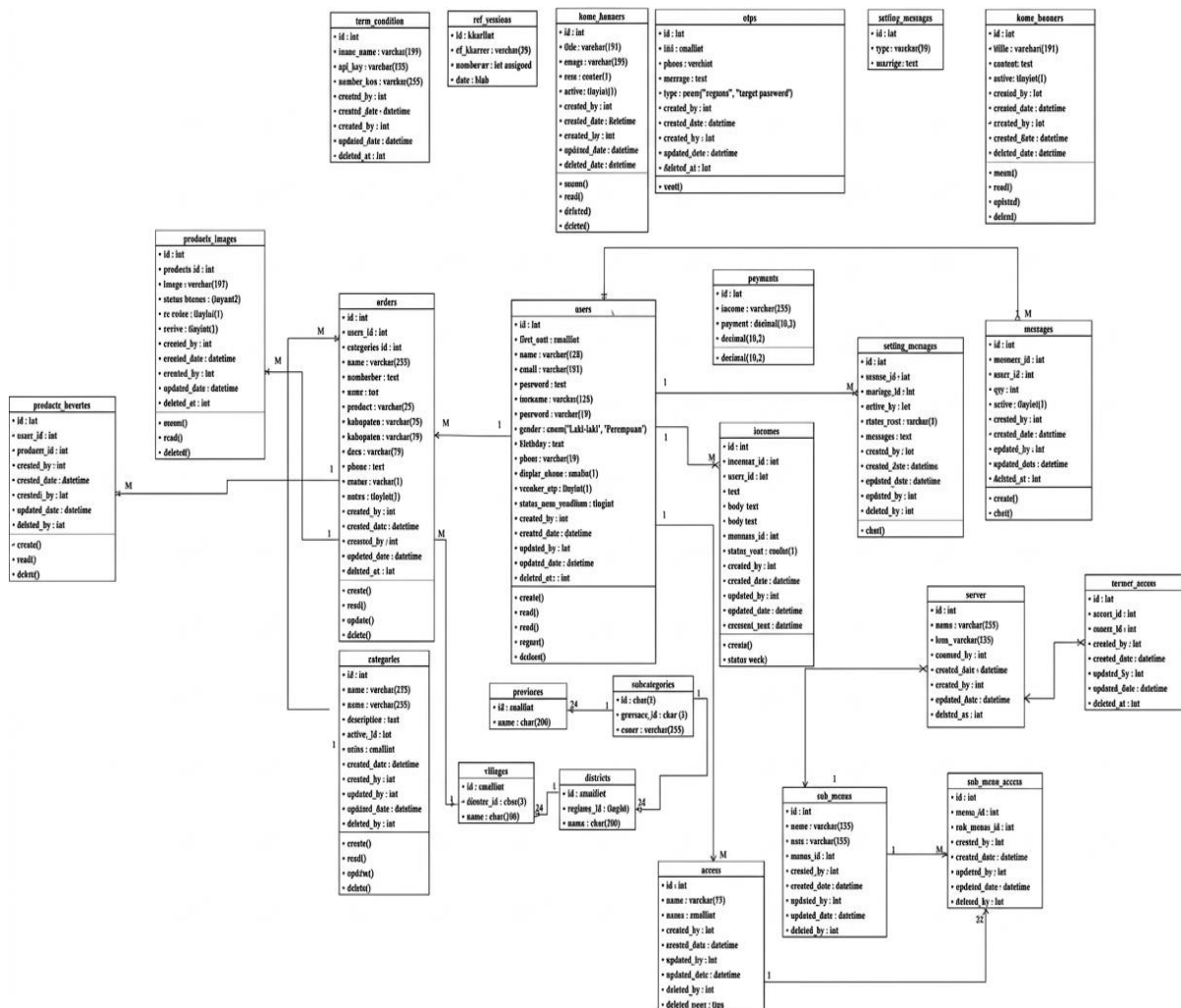


Figure 3 The platform's class diagram

Concurrently UML models, including Class Diagrams present in **Figure 3** specifying visibility modifiers, operations, and attributes, and Use Case and Activity Diagrams modeling user interactions and process flows, were also developed to provide a blueprint for the system's structural organization and to link

functional requirements to concrete interaction patterns and control flows [19]. Furthermore, an Entity-Relationship Diagram (ERD) was produced to define relational foreign-key constraints across core tables (users, products, transactions, forum boards) as present in **Figure 4**.

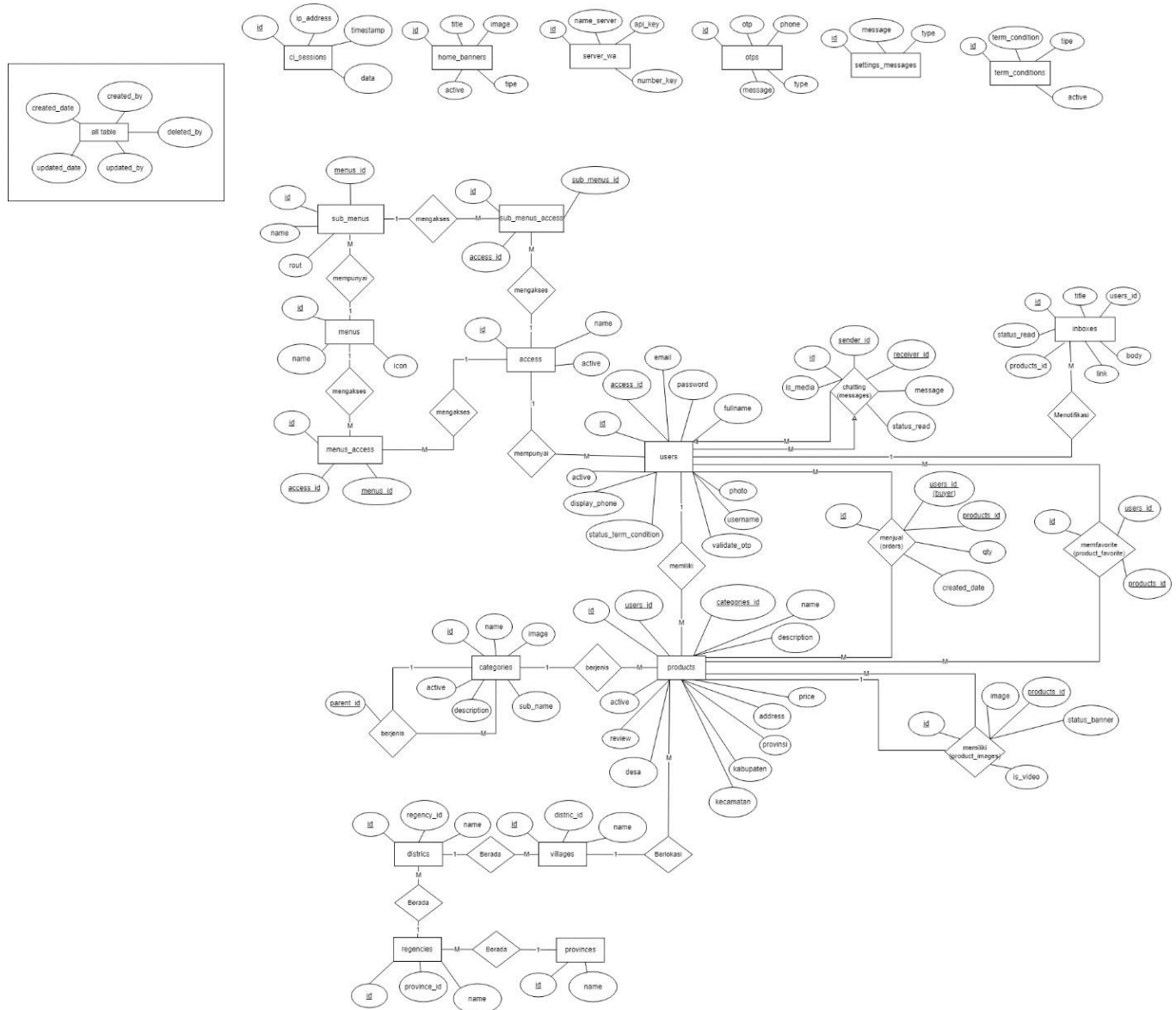


Figure 4. The platform's Entity Relationship Diagram (ERD)

This model formalized data integrity rules and supporting consistent enforcement of data relationships across the ecosystem [20]. Together, these design artifacts reduce implementation ambiguity and provide explicit references for expected workflows, data structures, and interaction patterns, directly supporting the study's aim of reducing

transaction uncertainty through clearer information architecture [1], [7].

3.4 Implementation

During the implementation phase, the blueprints from the design phase were translated into a fully functional software prototype. The mobile buyer client was coded using the Flutter framework, leveraging its

responsive widget system to deliver a consistent interface for product discovery and real-time interaction as in [3].

```
class _MahasiswaScreenState extends State<MahasiswaScreen> {
  final String apiUrl = "http://10.0.2.2/api_ci/index.php/mhscontroller";

  Future<List<dynamic>> fetchMahasiswa() async {
    final response = await http.get(Uri.parse(apiUrl));

    if (response.statusCode == 200) {
      final jsonResponse = json.decode(response.body);
      return jsonResponse['data'];
    } else {
      throw Exception('Gagal memuat data mahasiswa');
    }
  }
}
```

Figure 5. Flutter data retrieval routine

On the server side, the backend and API endpoints were engineered using CodeIgniter 3 and PHP. This lightweight MVC framework allowed for rapid API routing, secure request handling and smooth execution of business logic without the overhead of heavy enterprise environments as in [1], [20].

```
<?php
defined('BASEPATH') OR exit('No direct script access allowed');

class MhsController extends CI_Controller {

  public function __construct() {
    parent::__construct();
    $this->load->model('Mhs_model');
  }

  public function index() {
    $data = $this->Mhs_model->getAll();
    $this->output
      ->set_content_type('application/json')
      ->set_output(json_encode([
        'status' => true,
        'data' => $data
      ]));
  }
}
```

Figure 6. Backend API endpoint implementation in CodeIgniter 3

Data persistence was managed entirely through a MySQL database, where tables were indexed to support efficient query execution for catalog filtering, user authentication and message storage.

Following the deployment of the backend operations and client-side protocols, the final system output successfully rendered the intended interfaces. The visual results of the implemented mobile buyer client are presented in Figure 7 and Figure 8.

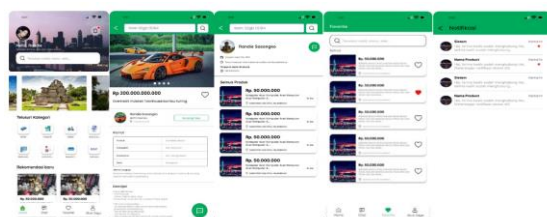


Figure 7. Overview of the platform on mobile display

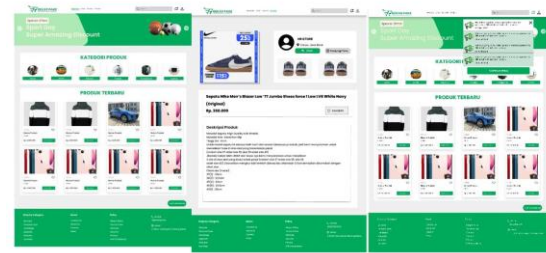


Figure 8. Overview of the platform on website

3.5 Functional Testing

In alignment with the research boundaries, the developed features of the *Pakai Lagi* platform were empirically evaluated using the black-box testing method. Each of the 14 functional modules defined during the requirement analysis phase was subjected to rigorous test cases to verify actual system responses against predefined expected outcomes. A total of 49 black-box test cases were executed across the fourteen functional modules, with particular emphasis on components directly related to information transparency and risk reduction such as product catalog, product details, peer-to-peer communication chat and seller store profiles. As documented across table, the empirical results yielded a successful validation rate, with every primary test case achieving a fully valid conclusion. The system successfully managed baseline operations, proving that the underlying data pipelines, security protocols and clients-server communication channels function reliably under structured test flows.

Table 1. Summary of the black-box testing results

T R	Module	T C	Result Conclusion	Summary &
1	Account Authentication	5	Validates credentials and empty parameters; blocks unauthorized entries and updates home dashboard. Result: Valid	
2	User Registration	10	Enforces data constraints and duplicate phone tracking; successfully triggers OTP dispatcher. Result: Valid	
3	Registration OTP	3	Rejects token mismatches and verification parsing errors; transitions user state inside MySQL database. Result: Valid	
4	Password Recovery	3	Validates active database records; blocks missing or unregistered numbers from recovery workflows. Result: Valid	

T R	Module	T C	Result Conclusion	Summary &
5	Password Recovery OTP	4	Maps token arrays and session lifetimes; blocks expired authorization tokens from triggering resets. Result: Valid	
6	New Password Configuration	5	Evaluates field matching constraints; encrypts database rows and securely updates the user status. Result: Valid	
7	Homepage Dashboard	6	Banner sliders and category loops load perfectly; query engine routes keywords (e.g., "bola") smoothly. Result: Valid	
8	P2P Communication (Chat)	3	Live text messaging pipeline operates reliably; interface alignment and emoji layouts are under active optimization. Result: Valid*	
9	Product Catalog	2	Basic image/text blocks load successfully; right-side price and category multi-criteria sorting pane is under development. Result: Valid*	
10	Product Detail	1	Renders wear-and-tear info and pricing details; functional link redirects buyers directly to the chat module. Result: Valid	
11	Favorite Management	1	Love toggle assignments register inside the database immediately; updates page watchlists automatically. Result: Valid	
12	Profile Settings	4	Restricts alphabetical entries within mobile number fields; safely updates records inside the database. Result: Valid	
13	System Notification	1	Displays transactional alert feeds; read/unread status visibility flags update correctly inside relational tables. Result: Valid	
14	Seller Store Profile	1	Maps targeted vendor metrics accurately and correctly isolates independent store catalog views. Result: Valid	
Total Test Case		49	All functional validation success rate achieved	

*status: as expected, but still under development.

In the assessment of the current prototype, certain cases within the chat page and product catalog modules were designated with an “as expected, but still under development” status. As in [2], [15], this classification denotes that the foundational logic and backend data mechanisms are mature; the core components compile correctly, messages route successfully and data fields render accurately as intended. However, this status indicates that the peripheral presentation

layers and granular data-sorting interfaces are currently being optimized for an enhanced user experience in subsequent deployment phases. Specifically, while the essential communication lines for peer-to-peer chat logs, initial-character lookups, product listings, and localized screen redirects operate reliably, their front-end execution such as layout alignment templates, emoji formatting, and precise low-to-high price categorization fields, are preserved as active areas for incremental refinement. By documenting these valid outcomes alongside these strategic development indicators, this study establishes a verified functional baseline for the platform, demonstrating that the lightweight architecture is operationally viable for localized consumer-to-consumer ecosystems while providing clear, predefined paths for future interface adjustments [2], [15], [21].

CONCLUSION

This study designed and developed the *Pakai Lagi* platform to facilitate second-hand product transaction. Developed using a decoupled client-server architecture utilizing the Flutter framework for the mobile buyer interface, CodeIgniter 3 for the administrative backend, and MySQL for database management, the system provides a lightweight alternative suitable for community-driven deployments.

The empirical evaluation, conducted through 49 black-box test cases across 14 core functional modules demonstrated a 100% functional validation success rate. All 49 test cases achieved a valid conclusion, indicating that the implemented core functions operated consistently according to the predefined expected requirements. However, certain sub-functions, specifically advanced layouts within the peer-to-peer chat module and the granular price/category product filtering controls, were classified as under development during testing highlighting the need for incremental interface refinement in subsequent development cycles.

Nevertheless, this development demonstrates the implementation of a buyer interface that supports structured product discovery, immediate access to detailed product information, and active buyer-seller interaction within a decentralized second-hand marketplace. Therefore, further empirical

evaluation involving field-based usability testing and technical system performance measurement is highly recommended to assess the platform beyond its baseline functional performance.

DECLARATIONS

Author Contributions: The first author conceptualized the study, developed the methodology, conducted the system analysis, designed and implemented the backend and API endpoints using CodeIgniter 3, and prepared the original draft; the second author supervised the research, validated the findings, reviewed and edited the manuscript, and approved the final version; the third author contributed to data curation and assisted in the system design modeling; the fourth author developed the Flutter mobile buyer client interface, conducted the functional black-box testing, and analyzed the testing results.

Funding: This research received no external funding from any public, commercial, or not-for-profit funding agency. The authors independently covered all costs related to the design, development, and testing of the Pakai Lagi platform.

Conflict of Interest: The authors declare no conflict of interest. None of the authors has any financial, commercial, or personal relationship that could have inappropriately influenced the design, development, evaluation, or reporting of this study.

Generative Artificial Intelligence Use Statement: No Generative AI or Large Language Model was used to generate the scientific content, research data, analysis, interpretation, or conclusions of this manuscript.

Ethics Approval and Informed Consent: Not applicable. This study focused on system design, development, and functional black-box testing, and did not involve human participants, personal data collection, or animal subjects; therefore, ethics approval and informed consent were not required.

Data Availability: The data and source code supporting the findings of this study are

available from the corresponding author upon reasonable request.

REFERENCES

- [1] Silfia, Anjelina, Ardiyansyah, and Y. Mustika, "Implementation of a Web-Based Sales Information System and Stock Management at TB Solusi Mitra Bangunan," *Journal of Artificial Intelligence and Engineering Applications*, vol. 5, no. 3, Jun. 2026, [Online]. Available: <https://ioinformatic.org/>
- [2] A. Mahgfiro *et al.*, "Design of a Preloved Marketplace for University Student," *Jurnal Kecerdasan Buatan dan Teknologi Informasi*, vol. 5, no. 2, pp. 306–313, May 2026, doi: 10.69916/JKBTI.V5I2.489.
- [3] A. N. Kusuma and A. F. Waluyo, "Web and Mobile-Based Online Watch Auction Application Using Waterfall Method," *Journal of Scientific Research, Education, and Technology (JSRET)*, vol. 5, no. 1, pp. 801–819, Mar. 2026, doi: 10.58526/JSRET.V5I1.1061.
- [4] S. Zeyu and E. A. Sundararajan, "Web-Based Second-Hand Marketplace," Selangor, 2025.
- [5] U. Vicamara, A. Santoso, and R. Riawan, "Thrift shopping intention: Understanding determinant of second-hand apparel shopping behavior," *Jurnal Ekonomi dan Bisnis*, vol. 26, no. 2, pp. 393–412, Nov. 2023, doi: 10.24914/JEB.V26I2.8276.
- [6] Z. Xinyang, "Project Management of the Creation of an Online Platform for the Exchange and Sale of Used iItems 'EcoChange Haven,'" Lutsk, 2024.
- [7] Z. Cao, "Design and Implementation of the Lut Second-Hand Goods Trading Platform," Lappeenranta, 2024.
- [8] A. N. Arif, "Sistem Informasi Donasi Berbasis Mobile Android pada Lembaga Dana Sosial Al-Jihad (DASA) melalui Integrasi Codeigniter dan Flutter," *Simtek: jurnal sistem informasi dan teknik komputer*, vol. 9, no. 2, pp. 80–88, Oct. 2024, doi: 10.51876/SIMTEK.V9I2.304.

- [9] I. M. Safitri and D. Kurniadi, "Rancang Bangun Aplikasi Jual-Beli Barang Secondhand di Komunitas UNP (SecondApps) Berbasis Android," 2024.
- [10] S. Che, "Developing a Second-Hand Online Shopping Application with Java and Android," *Vaasa*, Mar. 2021.
- [11] B. A. Pratama, U. Proboyekti, and K. Wijana, "Penerapan Metode User Centered Design (UCD) Dalam Pembangunan Layanan Online Jual Beli Barang Bekas," *Jurnal Terapan Teknologi Informasi*, vol. 4, no. 1, pp. 33–43, Jul. 2021, doi: 10.21460/jutei.2020.41.192.
- [12] N. A. Mobarak, N. A. Amin, A. Abdel-Mohssen, A. Sharif, Y. T. Halim, and K. Brickman, "Second glance: exploring consumer shifts to thrift shopping and perception of second-hand fashion," *Future Business Journal* 2025 11:1, vol. 11, no. 1, pp. 88–, Apr. 2025, doi: 10.1186/S43093-025-00481-2.
- [13] Kelvin, "Rancang Bangun Aplikasi Web Jual Beli Barang Bekas untuk Mahasiswa di Lingkungan Kampus," *Ikraith Informatika*, vol. 9, no. 3, Nov. 2025.
- [14] D. Puspita-Sari, P. Adi-Wicaksono, N. Rahmadi, M. Surya-Perdhana, and O. Ayuning-Nareswari, "Understanding Consumer Fashion Circular Behavior Regarding Second Hand Clothings in Indonesia: An Extended Theory of Planned Behavior and Value Belief Norm Theory Perspective," *Journal of Industrial Engineering and Management*, vol. 18, no. 3, pp. 495–526, 2025, doi: 10.3926/jiem.6412.
- [15] F. Ahmed, N. Kumar Jha, F. Guided, and K. Chitra, "Design and Development of a Localized E-Commerce Solution for Students focussing on Economical Sharing," Nov. 2024, Accessed: Aug. 12, 2026. [Online]. Available: <https://arxiv.org/pdf/2411.11527>
- [16] S. R. Nuraida, A. Lattu, and M. A. Permana, "Perancangan UI Aplikasi Mobile Untuk Penjual Barang Bekas Dengan Metode Design Thinking," *JIPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)*, vol. 11, no. 2, pp. 1145–1159, May 2026, doi: 10.29100/JIPI.V11I2.8296.
- [17] T. Prilsafira, Y. Novaria Kunang, and M. H. Putra, "Rest API Backend Aplikasi E-Commerce Secondhand Menggunakan Framework Spring Boot," *Positif: Jurnal Sistem dan Teknologi Informasi*, Feb. 2023.
- [18] S.-P. Ruben, R.-A. Eduardo, R.-A. Aldo, and A.-G. Elizabeth, "Software Architecture Using a Decoupling, Layered and Web Restful Service Approach," *J. Theor. Appl. Inf. Technol.*, vol. 98, no. 7, p. 7, Apr. 2020, [Online]. Available: www.jatit.org
- [19] I. Setiawan and Santoso, "Rancang Bangun Aplikasi Jual Beli Barang Bekas Online," *Jurnal Teknik Informatika*, vol. 16, no. 1, pp. 1–7, Jan. 2024.
- [20] M. Siahaan and T. Lim, "Perancangan Sistem Informasi Penjualan Barang Bekas Berbasis Website dengan Metode Extreme Programming Design and build a Website-Based Second-Hand Sales with Extreme Programming Method," *Journal of Computing Engineering, System and Science) e-ISSN*, vol. 9, no. 1, pp. 39–50, 2024, [Online]. Available: www.jurnal.unimed.ac.id
- [21] H. Kapoh, O. E. Melo, and A. A. Kimbal, "Black Box Testing in Web-based Applications: Case Study-Remedial Application at Manado State Polytechnic," 2021.