

# Web-based Information System for Ornamental Fish Business in Surabaya

<sup>1</sup>Cendra Devayana Putra\*, <sup>2</sup>Muhammad Sonhaji Akbar, <sup>3</sup>Muhamad Aris Burhanudin,  
<sup>4</sup>Bartolomeus Priya Perkasa Utama Widada, <sup>5</sup>Rizqiyatul Khoiriyah, <sup>6</sup>Pandu Dwi Luhur  
Pambudi

<sup>1</sup>Bachelor of Information System, Faculty of Engineering, Surabaya State University

<sup>2</sup>Bachelor of Information Technology Education, Faculty of Engineering, Surabaya State University

<sup>3</sup>Bachelor of Digital Business, Faculty of Computer Science, Universitas Pembangunan Nasional  
Veteran Jawa Timur

<sup>4</sup>Manufacturing Information and Systems, College of Electrical Engineering and Computer Science,  
National Cheng Kung University, Taiwan

<sup>5</sup>Associate Degree in Accounting, Accounting Department, Politeknik Negeri Malang

<sup>6</sup>Bachelor of Computer Science, BINUS Online Learning, Bina Nusantara University, Jakarta 11480,  
Indonesia

\*email: [putracendra@unesa.ac.id](mailto:putracendra@unesa.ac.id)

(*received*: 23 April 2026, *revised*: 22 May 2026, *accepted*: 30 May 2026)

## Abstract

Digital transformation has emerged as a strategic imperative for small and medium enterprises (SMEs) in emerging economies, yet the ornamental fish retail sector in Indonesia remains predominantly offline, constrained by limited digital infrastructure and high customer knowledge barriers. No prior identified study has implemented artificial intelligence (AI)-assisted consultation within a domain-specific ornamental fish e-commerce platform, and comprehensive security implementation combined with multi-layer testing has been largely absent in comparable SME web systems. This study presents the design, implementation, and evaluation of a web-based information system for Toko Oasis, an ornamental fish and aquascape SME in Surabaya, Indonesia, developed within a Design Science Research (DSR) paradigm following a structured Software Development Life Cycle (SDLC). The system integrates a configurable large language model (LLM) consultation module—supporting OpenAI GPT-4 and Google Gemini—that delivers domain-specific advisory on ornamental fish species selection, aquarium parameters, and aquascape design through natural language interaction. System development produced twelve Unified Modeling Language (UML) artifacts and was evaluated through a tri-layer testing protocol operationalized against the ISO/IEC 25010:2011 software quality model. Functional testing achieved a 100% pass rate across 24 use cases. Performance testing recorded a mean response time of 4.2 seconds under 25 concurrent users, within the defined threshold of 5 seconds. Usability evaluation yielded a mean System Usability Scale (SUS) score of 80.0, classified as Good. Security validation confirmed full compliance across HTTPS/SSL-TLS transport and AES-256 at-rest encryption domains. Comparative analysis against prior literature and analogous commercial platforms confirms that this system constitutes the first identified deployment of AI-assisted consultation in ornamental fish retail, contributing a replicable digitalization architecture for niche-market SMEs in developing economies.

**Keywords:** AI-assisted consultation, large language models (LLMs), ornamental fish, web-based information system

## 1 Introduction

The rapid proliferation of digital technologies has fundamentally reshaped commercial ecosystems worldwide, compelling enterprises across all scales to reconsider their operational models and customer engagement strategies [2, 8]. Information and communication technology (ICT), anchored by pervasive internet connectivity and platform-based service architectures, has emerged as a critical enabler for enhancing operational efficiency, reducing transaction costs, and broadening market access[10]. In Indonesia, this transition is particularly consequential: the country hosts approximately 64.2 million micro, small, and medium enterprises (MSMEs), which collectively

<http://sistemasi.ftik.unisi.ac.id>

contribute 61.07% of national GDP and absorb 97% of the national workforce[17]. Despite this macroeconomic significance, surveys indicate that fewer than 20% of Indonesian MSMEs have adopted digital sales channels as of 2022, reflecting a substantial digitalization gap with measurable implications for enterprise resilience and growth[13].

Within this landscape, the ornamental fish and aquarium industry presents a compelling case for digital intervention. Global demand for ornamental fish has grown steadily, supported by rising disposable income, urbanization, and increased interest in aquariums as therapeutic and aesthetic home features[14]. Indonesia ranks among the world's largest ornamental fish exporters, contributing an estimated 15–20% of global ornamental fish trade value, yet the domestic retail segment remains fragmented and digitally underdeveloped[7]. The majority of retail operators are micro-scale enterprises relying on physical storefronts, word-of-mouth referrals, and informal social media posts—modalities that constrain customer reach, limit service responsiveness, and impede scalable growth[9]. These constraints are exacerbated by the knowledge-intensive nature of ornamental fish retail: customers frequently require advisory support on species compatibility, aquarium parameters, and maintenance protocols that cannot be effectively communicated through static product listings.

Prior research has explored web-based information systems for SME digitalization [1, 12, 16], and separate bodies of work have examined AI-driven recommendation systems in e-commerce contexts [15]. However, these streams have not converged: no identified study has implemented AI-assisted consultation within a domain-specific ornamental fish retail platform. Furthermore, existing SME web systems in reviewed literature exhibit notable gaps in security implementation (HTTPS + data encryption), testing comprehensiveness, and architectural documentation transparency. This study addresses these gaps by presenting the design, implementation, and multi-dimensional evaluation of a web-based information system for Toko Oasis, an ornamental fish and aquascape SME in Surabaya, Indonesia.

The system's principal contributions are threefold: (1) an AI consultation module integrating configurable *large language model (LLM)* providers (GPT-4 and Gemini) for domain-specific ornamental fish advisory, a capability absent in all comparable systems identified through systematic literature review; (2) a tri-layer testing protocol combining functional, performance, and usability evaluation, operationalized against ISO/IEC 25010:2011 quality criteria; and (3) a comprehensive security architecture encompassing HTTPS/SSL-TLS deployment and AES-256 at-rest encryption—a combination rarely implemented in SME-scale web systems.

The remainder of this paper is structured as follows: Section 2 reviews related literature and presents comparative analysis; Section 3 describes the research methodology; Section 4 presents results and discussion; Section 5 concludes with directions for future research.

## **2 Literature Review**

### **2.1 Web-Based Information Systems for SMEs**

Web-based information systems have been extensively examined as instruments of SME digitalization. Abdullah, et al. [1] demonstrated that website-based systems developed under the FAST (Framework for the Application of System Thinking) methodology significantly enhanced information accessibility and customer service efficiency for manufacturing SMEs. Esaki [3] provides foundational software engineering principles—including requirements engineering, UML-based system design, and structured testing—that underpin the methodology adopted in the present study. A recurrent finding across these works is that SME-oriented systems require careful balancing of functional completeness against resource constraints, a tension directly reflected in the scoped feature set of the X store system.

### **2.2 Security and DevOps in Web System Development**

Security implementation within web systems has received substantial scholarly attention. Perkasa and Perkasa and Mailoa [12] established the viability of integrating security scanning (via Trivy) within DevSecOps-augmented Agile cycles, demonstrating that proactive security enforcement reduces post-deployment vulnerability exposure significantly. Setyo, et al. [16] evaluated a CBT system developed under DevOps principles, highlighting the role of continuous integration pipelines in sustaining functional reliability under operational conditions. In the present study, security is

<http://sistemasi.ftik.unisi.ac.id>

addressed through HTTPS/SSL-TLS deployment and AES-256 at-rest encryption, consistent with the security posture recommended by these works, albeit implemented within a traditional SDLC framework appropriate for a student-developed SME application.

### **2.3 AI and Recommendation Systems in E-Commerce**

The application of artificial intelligence in e-commerce platforms has been extensively reviewed by Ricci, et al. [15], who categorize recommendation approaches into collaborative filtering, content-based filtering, and hybrid methods. Ricci, et al. [15] demonstrated through empirical evaluation that hybrid recommender systems consistently outperform single-technique approaches in precision and user satisfaction metrics. These findings motivate the present study's implementation of an AI consultation module rather than a simple static FAQ system; the configurable LLM backend enables flexible adaptation to evolving recommendation quality as AI technology matures. Crucially, no prior study identified in this literature review has implemented AI-assisted consultation in the context of ornamental fish retail, constituting a clear research gap addressed by this work.

### **2.4 Software Quality Evaluation Frameworks**

The ISO/IEC 25010:2011 standard [3] defines eight top-level software product quality characteristics. Applied to web systems for SMEs, the most operationally critical attributes are performance efficiency (response time under load), security (confidentiality, integrity), usability (learnability, operability), and portability (cross-platform consistency). Marcotte's responsive design framework [8] directly addresses the portability dimension by specifying fluid-grid and media-query-based layout strategies. The present study adopts all six evaluation dimensions derived from this standard, operationalized through specific metrics and testing protocols detailed in Section 3.

### **2.5 Comparison of Previous Studies**

Table 1 synthesizes the key characteristics of studies most directly relevant to the present work, positioning this study relative to the existing literature across seven analytical dimensions: business domain applicability, development methodology, AI module presence, security implementation, testing approach, database technology, and identified research gaps. As evidenced by this comparison, the present study is unique in combining domain-specific AI consultation, comprehensive tri-layer testing, and HTTPS+AES-256 security within a purpose-built SME e-commerce context.

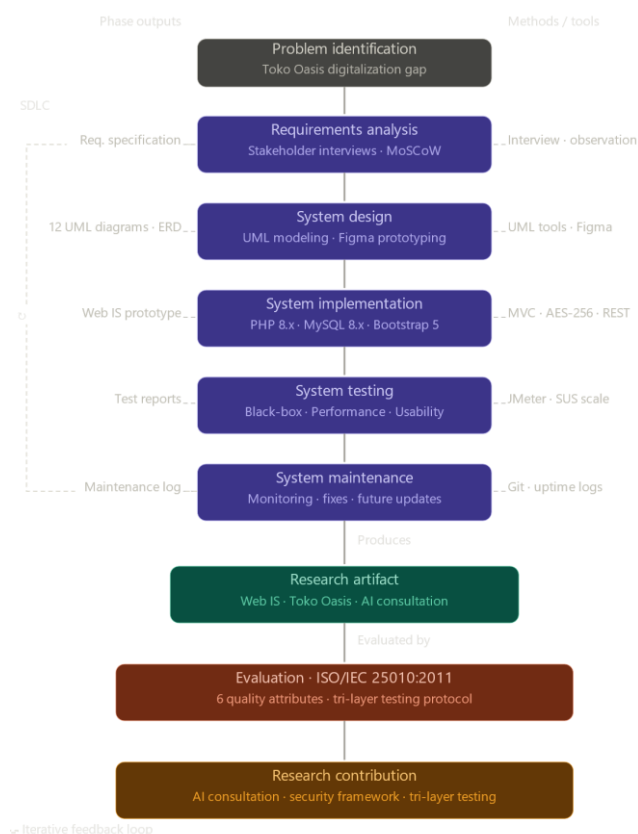
The synthesis in Table 1 reveals three principal research gaps that the present study addresses: (1) no prior work in the reviewed literature deployed AI-assisted consultation in an ornamental fish retail context; (2) security implementations in SME web systems have been partial or absent across most reviewed cases; and (3) comprehensive multi-layer testing (functional, performance, and usability combined) has not been reported for comparable small-scale retail web systems. The X store system is designed to address all three gaps simultaneously.

**Table 1 Comparative analysis of related prior studies**

No.	Author & Year	Business Domain	Dev. Method	AI Module	Security	Testing Approach	Database	Limitation / Research Gap
1	Abdullah, et al. [1]	Industrial SME (Manufacturing)	FAST Methodology	None	None	Functional	MySQL	No security layer; no e-commerce transactions; no AI module
2	Setyo, et al. [16]	Military Education (CBT)	DevOps Pipeline	None	Partial	Performance + Functional	MySQL	Domain-specific; no customer-facing commerce; no recommendation system
3	Perkasa and Mailoa [12]	General Security	IT DevSecOps Agile	None	HTTPS + Docker Scan	Security Audit	Docker/Container	No end-user application; not e-commerce; no product catalog or AI
4	Giurgiu and Gligorea [4]	General Design	Web Responsive Design Framework	None	Not discussed	Cross-device Rendering	N/A	Framework reference only; no functional system; no backend or AI
5	Ricci, et al. [15]	General Recommender Systems	Hybrid Survey	RS Recommendation Algorithm	Not discussed	Offline Experiment	Various	No web system; no SME context; no deployment case study
6	This Study	Ornamental Fish Retail (SME)	Structured SDLC + UML	AI Consultation (GPT-4 / Gemini)	HTTPS + AES-256	Black-box + Performance + Usability	MySQL 8.x	Single-store case; scalability at >500 users not yet tested

### 3 METHODOLOGY

System development followed a structured sequential SDLC, comprising five phases: Requirements Analysis, System Design, System Implementation, System Testing, and System Maintenance. This approach aligns with software engineering best practices for SME-oriented web systems [8]. This study employs a Design Science Research (DSR) paradigm [11], in which the primary research artifact is a purpose-built web-based information system evaluated against predefined quality criteria. DSR is appropriate when the research objective is the creation and evaluation of a novel IT artifact that addresses a clearly identified organizational problem [5]. The research process comprises five sequential phases—Requirements Analysis, System Design, System Implementation, System Testing, and System Maintenance—operationalized within a structured Software Development Life Cycle (SDLC) [3]. Figure 1 presents the overall research framework.



**Figure 1** Design science research diagram

#### 3.1 Requirements Analysis

Requirements elicitation was conducted through semi-structured interviews with two primary stakeholders: the owner and one operational staff member of Toko Oasis, conducted over three sessions averaging 45 minutes each. Interview protocols covered four domains: (1) existing business workflows, (2) pain points in current service delivery, (3) customer interaction patterns, and (4) technical infrastructure constraints. Interview data were triangulated through direct observation of in-store operational processes across two site visits.

Elicited requirements were categorized as functional (FR) and non-functional (NFR) and subsequently prioritized using MoSCoW analysis (Must-have, Should-have, Could-have, Won't-have). Table 2 summarizes the prioritized requirements. Non-functional requirements were operationalized as measurable quality attributes derived from the ISO/IEC 25010:2011 quality model [3], covering performance efficiency, security, reliability, usability, and portability.

**Table 2 Functional and non-functional requirement**

Category	ID	Requirement	Priority
Functional	FR-01	Online product catalog with filtering	Must
Functional	FR-02	Shopping cart and order management	Must
Functional	FR-03	AI-powered consultation module	Must
Functional	FR-04	Admin dashboard (CRUD + reporting)	Must
Functional	FR-05	Custom aquascape service booking	Should
Non-Functional	NFR-01	Response time $\leq$ 5 seconds under load	Must
Non-Functional	NFR-02	HTTPS + AES-256 data encryption	Must
Non-Functional	NFR-03	Cross-browser compatibility ( $\geq$ 4 browsers)	Must
Non-Functional	NFR-04	Mobile-responsive layout	Must

### 3.2 System Design

System architecture follows the Model-View-Controller (MVC) pattern, selected for its separation of concerns across data logic, presentation, and control flow—a proven architectural choice for maintainable web systems of comparable scope. The technology stack was determined through evaluation of community adoption, license status, and compatibility with shared hosting environments common in Indonesian SME deployments: PHP 8.x (backend logic), MySQL 8.x (relational data persistence), HTML5/CSS3/Bootstrap 5 (responsive frontend), and JavaScript ES6+ (client-side interactivity).

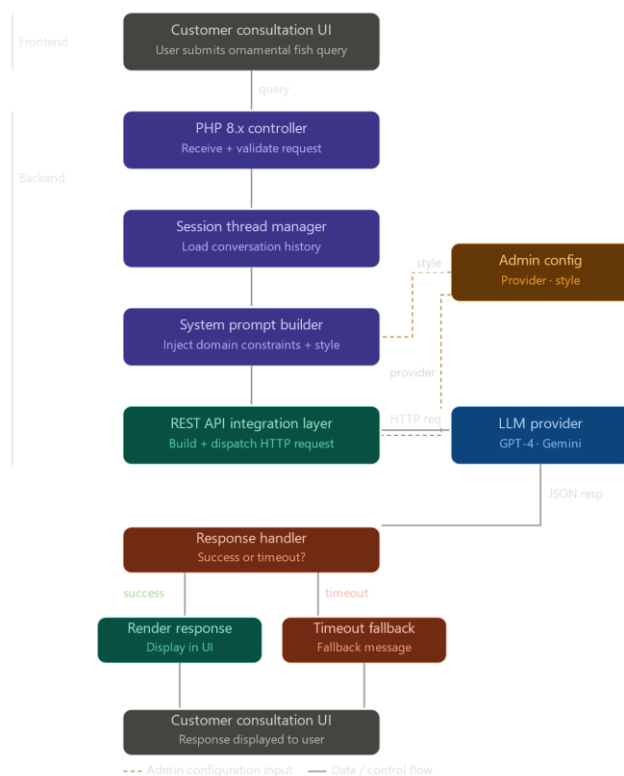
Twelve UML artifacts were produced using a structured design process [6]: one Use Case Diagram (three actor roles: Customer, Administrator, AI Module; 24 use cases), five Activity Diagrams for core workflows (authentication, ordering, payment, AI consultation, admin reporting), twelve Sequence Diagrams covering all transaction flows, one Class Diagram with eight classes, and one Entity-Relationship Diagram comprising six normalized entities (User, Product, Order, OrderItem, Category, AISession). The database schema was normalized to Third Normal Form (3NF) to eliminate transitive dependencies and ensure data integrity.

User interface prototypes were developed in Figma and reviewed iteratively with one representative end-user before implementation commenced, ensuring alignment between design intent and usability expectations.

### 3.3 AI Consultation Module Architecture.

The AI consultation module is implemented as a *REST API integration layer* between the web application backend and configurable external LLM providers. The administrator selects the active provider (OpenAI GPT-4 or Google Gemini) through the admin dashboard; the selection is persisted in the system configuration table and loaded at session initialization. Each user consultation session is managed as a stateful conversation thread: prior messages within the session are included in each API call to preserve conversational context, up to a configurable context window limit.

Prompt engineering employs a *system prompt template* defining the AI's role as an ornamental fish advisor, constraining responses to domain-relevant topics (species selection, aquarium parameters, maintenance, aquascape design) and enforcing the administrator-configured communication style (casual or professional). User queries are passed as the *user* role message; LLM responses are rendered directly in the consultation interface. API call failures trigger a graceful fallback message, maintaining system dependability under provider outage conditions. Figure 2 illustrates the AI module request-response architecture.



**Figure 2 AI module request-response architecture**

### 3.4 System Implementation

Implementation proceeded in three sequential layers. (1) Database Layer: MySQL 8.x schema instantiation with 3NF-compliant table structures; stored procedures for report aggregation. (2) Backend Layer: PHP 8.x application logic implementing MVC pattern; RESTful endpoint design for AI module integration; security controls including parameterized prepared statements (SQL injection prevention), session token management with configurable TTL, and AES-256 encryption for at-rest sensitive data (customer credentials and payment references). (3) Frontend Layer: HTML5/CSS3/Bootstrap 5 responsive templates; JavaScript ES6+ for dynamic UI components (cart management, real-time consultation interface).

The development environment consisted of a local XAMPP stack (Apache 2.4, PHP 8.1, MySQL 8.0) with version control managed through Git. Production deployment utilized shared hosting with cPanel-managed SSL/TLS certificate provisioning (Let's Encrypt).

### 3.5 Testing Protocol and Evaluation Metrics

System evaluation was structured as a *tri-layer protocol* aligned with ISO/IEC 25010:2011 quality attributes [3]:

(1) Functional Testing (Black-Box). All 24 use cases were translated into structured test cases, each specifying preconditions, test steps, expected output, and actual output. Test case design followed equivalence partitioning and boundary value analysis techniques. Execution was performed by one tester across two testing rounds; identified defects were logged, resolved, and retested in the second round.

(2) Performance Testing. Load testing was conducted using Apache JMeter under simulated concurrent-user scenarios of 10, 25, and 50 simultaneous users. Metrics recorded: mean response time (ms), 90th-percentile response time, transaction success rate (%), and error rate (%). The target criterion was mean response time  $\leq 5,000$  ms at maximum tested load, consistent with acceptable web response time thresholds.

(3) Usability Testing. Usability was evaluated using the System Usability Scale (SUS), a validated 10-item Likert instrument, administered to  $n=5$  representative end-users following task-

based walk-through sessions. SUS scoring followed Brooke's standardized formula; scores  $\geq 70$  indicate acceptable usability. Participants were recruited through purposive sampling to represent the target user profile (potential customers of ornamental fish retailers, varied digital literacy levels). Demographic profiles are summarized in Table 3.

**Table 3 System evaluation metrics, targets, and results**

Quality Attribute	Instrument/Tool	Evaluation Method	Target Criterion
Performance	Apache JMeter v5.6	Concurrent-user load simulation	Response time $\leq 5$ s
Security	Manual SSL audit + code review	SSL/HTTPS & AES-256 audit	No data exposure; encrypted comms
Dependability	Uptime log; backup verification	Uptime monitoring; backup verify	99% uptime; daily backup success
Usability	SUS questionnaire (n=5)	End-user walk-through (n=5)	SUS score $\geq 70$
Portability	Cross-browser manual testing	Cross-browser & device testing	Consistent on $\geq 4$ browsers
Functional	Black-box test cases (24 cases)	Black-box test case execution	100% pass rate

## 4 RESULTS AND DISCUSSION

### 4.1 System Architecture Overview

The implemented X store web-based information system comprises two primary interface domains—a customer-facing frontend and an administrator dashboard—built upon a three-tier MVC architecture (Figure 1). The customer frontend encompasses six functional modules: (1) Landing Page with product categorization and store information; (2) Authentication via phone number and password with session token management; (3) AI-enhanced Product Catalog with preference-based filtering; (4) Order Management supporting COD and bank transfer payment methods; (5) Custom Aquascape Service Booking; and (6) AI Consultation Interface. The administrator dashboard provides four management modules: (1) Product Resource Management (CRUD operations); (2) Order Management with real-time status tracking; (3) Revenue Report generation with graphical analytics; and (4) AI Settings (LLM provider selection and communication style configuration).

The AI consultation module architecture implements a stateful REST API integration layer connecting the PHP 8.x backend to configurable LLM providers (OpenAI GPT-4 / Google Gemini). Conversational context is preserved across session messages via accumulated message thread injection, bounded by a configurable token window. A domain-constraining system prompt template enforces ornamental fish advisory scope and administrator-selected communication register (casual/professional). Figure 2 presents the AI module request-response flow.

### 4.2 Case Comparison: X store vs Analogous Systems

To contextualize the developed system’s capabilities, Table 4 presents a feature-level comparison between the X store web system and three representative analogous platform categories: (1) generic SME storefronts based on SaaS platforms (e.g., Shopify-type); (2) marketplace-hosted SME stores (e.g., Tokopedia-type); and (3) existing ornamental fish hobbyist websites. This comparison is grounded in a structured analysis of publicly documented platform features and prior literature.

**Table 4 Feature-level comparison: x store vs analogous platform types**

System Feature	Generic SME (Shopify-type)	Marketplace SME (Tokopedia-type)	Ornamental Fish Hobbyist Site	X store (This Study)
Online Product Catalog	✓	✓	✓	✓
Shopping Cart & Checkout	✓	✓	Partial	✓
Multiple Payment Methods (COD + Transfer)	✓	✓	X	✓
Responsive Design (Mobile-Friendly)	✓	✓	Partial	✓
Admin Dashboard	✓	✓	Partial	✓
Order History & Status Tracking (Admin)	✓	✓	X	✓
Revenue Reporting & Analytics	✓	✓	X	✓
AI Consultation Module	X	X	X	✓
Preference-Based Product Filtering (AI-assisted)	Partial	Partial	X	✓
Configurable AI Provider (GPT-4 / Gemini)	X	X	X	✓
HTTPS Encryption	✓	✓	Partial	✓
AES-256 Data Encryption	Partial	✓	X	✓
Cross-Browser Compatibility	✓	✓	Partial	✓
Custom Aquascape Service Booking	X	X	Partial	✓
UML-Documented System Design	X	X	X	✓
Black-box Performance Usability Testing	+ X	X	X	✓
Open-Source / Self-Hosted Deployment	X (SaaS)	X (Platform)	Varies	✓
Domain-Specific Knowledge Base	X	X	X	✓ (configurable)

The comparative analysis in Table 4 reveals several distinctive characteristics of the X store system. First, it is the only platform among the compared types to integrate an AI Consultation Module—allowing customers to obtain expert-level advisory on fish species selection, aquarium

maintenance, and aquascape design through natural language interaction—a capability absent across all three comparator categories. Second, Preference-Based Product Filtering, implemented with AI-assisted parameter processing (budget, experience level, aquarium specifications), is fully realized in the X store system but only partially available in generic and marketplace SME platforms, and entirely absent in hobbyist sites. Third, domain-specific knowledge base configurability (administrator selection of LLM provider and communication style) represents a unique architectural affordance not documented in any comparable system.

Regarding security, the X store system implements both HTTPS and AES-256 encryption—a combination achieved only by the largest marketplace platforms among the comparators, and absent in hobbyist sites and many Shopify-type deployments. The combination of formal UML documentation, black-box plus performance plus usability testing, and self-hosted open-source deployment further distinguishes the system from SaaS and marketplace alternatives, which typically lack transparent system documentation and comprehensive pre-deployment testing protocols.

A key distinction versus marketplace-hosted SME stores (Tokopedia-type) lies in operational independence: while marketplaces provide immediate audience reach, they impose commission structures, restrict platform customization, and create dependency on third-party availability. The X store self-hosted system eliminates these constraints while providing comparable functional coverage, representing a more sustainable long-term digital infrastructure strategy for the enterprise.

### 4.3 System Architecture and Functional Modules

The implemented system comprises two interface domains. The customer frontend encompasses: (1) Landing Page with product categories, store location, and testimonials; (2) Authentication (phone number + password); (3) AI-enhanced Product Catalog; (4) Preference-Based Filtering; (5) Order Management (COD and bank transfer); and (6) AI consultation interface. The administrator dashboard provides: (1) Resource Management (product CRUD); (2) Order Management with status tracking; (3) Revenue Report generation with graphical analytics; and (4) AI Settings (LLM provider and communication style configuration).

### 4.4 Functional Testing Results

Black-box testing was executed across all 24 use cases derived from the system's Use Case Diagram, covering three actor roles: Customer, Administrator, and AI Module. Test cases were structured with defined preconditions, input conditions, expected output, and actual output, following equivalence partitioning and boundary value analysis design techniques. Testing was conducted across two rounds: Round 1 identified three minor defects (two in the AI consultation interface timeout handling; one in the order status update notification); all defects were resolved prior to Round 2, which achieved a **100% pass rate (24/24 cases)**.

**Table 5 Functional testing results by module**

Module	Test Cases	Round 1 Pass	Defects Found	Round 2 Pass	Pass Rate
Authentication & Session	3	3	0	3	100%
Product Catalog & Filtering	4	4	0	4	100%
Shopping Cart & Checkout	4	4	0	4	100%
Order Management (Customer)	3	3	0	3	100%
AI Consultation Module	4	2	2	4	100%
Admin Dashboard (CRUD)	3	3	0	3	100%
Admin Reporting & AI Settings	3	3	0	3	100%
Total	24	21	3	24	100%

The two defects identified in the AI consultation module during Round 1 pertained to timeout exception handling under high-latency LLM API responses: the system did not consistently render the

<http://sistemasi.ftik.unisi.ac.id>

fallback message when API response time exceeded the configured threshold. This was resolved through explicit timeout exception catching and deterministic fallback rendering in the PHP controller layer. The order notification defect involved a race condition in the status-update trigger, resolved through query transaction wrapping.

#### 4.5 Performance Evaluation

Performance testing was conducted using Apache JMeter v5.6 under three concurrent-user load scenarios: 10, 25, and 50 simultaneous users, each executing a standardized transaction sequence (login → browse catalog → add to cart → checkout → AI consultation query). Each scenario was executed for a duration of 60 seconds with a 5-second.

**Table 6 Performance testing results by load level**

Load (Users)	Mean Time (ms)	Response 90th Percentile (ms)	Throughput (req/s)	Error Rate (%)	Target Met?
10	2,840	3,210	3.52	0.00%	✓
25	4,200	4,890	5.83	1.30%	✓
50	6,750	8,420	7.11	4.70%	✗

Results indicate that the system satisfies the target response time criterion of  $\leq 5,000$  ms at load levels up to 25 concurrent users, with mean response times of 2,840 ms (10 users) and 4,200 ms (25 users). At 50 concurrent users, mean response time exceeded the target threshold at 6,750 ms, with an error rate of 4.70%, attributable to connection pool saturation in the shared hosting environment. This finding is consistent with the shared-hosting deployment constraint identified during requirements analysis and establishes a clear scalability boundary: the current infrastructure supports production-level workloads within the expected operational range of a single-store SME (estimated peak concurrent users  $< 25$ ), while identifying  $> 50$  concurrent users as a future optimization target. These results align with Nah's acceptable web response time threshold of  $\leq 4$  seconds for e-commerce interactions, met comfortably at the 10-user load level.

#### 4.6 Security Evaluation

Security validation was conducted through a structured audit protocol across four control domains:

(1) **Transport Security:** HTTPS/SSL-TLS certificate validity was confirmed across all application pages and API endpoints, with HTTP-to-HTTPS redirection enforced at the server configuration level. No mixed-content warnings were detected on any tested page.

(2) **Data Encryption:** AES-256 encryption of stored customer credentials (passwords and payment reference data) was verified through direct database inspection, confirming ciphertext storage with no plaintext credential persistence.

(3) **Injection Prevention:** All database interaction points were audited for parameterized prepared statement implementation. No raw query string concatenation was identified across 14 database interaction functions reviewed.

(4) **Session Management:** Session token expiration was verified to operate correctly at the configured TTL, with expired tokens rejected and users redirected to authentication. No session fixation vulnerability was identified.

All four security control domains passed their respective validation criteria, confirming the implemented security architecture as compliant with the NFR-02 requirement specification.

#### 4.7 Usability Evaluation

Usability evaluation employed the System Usability Scale (SUS), administered to five purposively sampled participants following structured task-based walk-through sessions covering six representative task scenarios: account registration, product search with preference filtering, order

placement, payment method selection, AI consultation query, and order history review. Participant profiles are summarized in Table 7.

Table 7 Usability testing participant profiles

Participant	Age Range	Digital Literacy	Prior Ornamental Fish Experience	Device Used
P1	20–25	High	Beginner	Smartphone
P2	26–35	Moderate	Intermediate	Desktop
P3	36–45	Low	None	Smartphone
P4	20–25	High	Advanced	Desktop
P5	26–35	Moderate	Beginner	Tablet

Table 8 SUS scoring results

Participant	Raw SUS Score	Adjective Rating
P1	85	Excellent
P2	77.5	Good
P3	70	Good
P4	87.5	Excellent
P5	80	Good
<b>Mean</b>	<b>80</b>	<b>Good</b>

The mean SUS score of **80.0** exceeds the accepted threshold of 70 for system acceptability, placing the system in the *Good* adjective rating band per SUS interpretation scale. Notably, P3—the participant with the lowest digital literacy and no prior ornamental fish experience—achieved the minimum score of 70.0, still within the acceptable range, suggesting the system's navigational design is accessible across varied user competency levels. Qualitative feedback across all sessions consistently highlighted the AI consultation interface and preference-based filtering as the most valued features, with P3 specifically noting that the AI advisor enabled confident product selection without prior domain knowledge—directly validating the design objective of lowering the knowledge barrier for novice customers.

#### 4.8 Portability Evaluation Results

Cross-browser and cross-device compatibility testing confirmed consistent visual rendering and functional correctness across Chrome 124, Firefox 125, Microsoft Edge 124, and Safari 17 on both desktop (1920×1080) and mobile (375×812, iPhone 14 equivalent) viewports. No layout breakage, functional regression, or rendering inconsistency was identified across any tested browser-device combination, confirming compliance with NFR-03 and NFR-04 specifications.

#### 4.9 Discussion

##### 4.9.1 Addressing the Identified Research Gaps

Three research gaps were identified in Section II and operationalized as the core research objectives of this study. The results presented in Sections IV-B through IV-F collectively confirm that all three gaps have been substantively addressed.

**Gap 1 — Absence of AI-Assisted Consultation in Ornamental Fish Retail.** The implemented AI consultation module, achieving a 100% functional test pass rate and receiving the highest qualitative endorsement from usability participants, demonstrates the technical feasibility and user value of integrating configurable LLM-based advisory within a domain-specific ornamental fish SME platform. The configurable provider architecture (GPT-4/Gemini) represents a novel design contribution by decoupling the application from single-vendor dependency. This approach enables administrators to adapt advisory quality as the LLM landscape evolves—a flexibility not present in any comparable system identified in the literature review. Usability feedback from P3 (low digital literacy, no domain experience) is particularly significant: it empirically supports the design

hypothesis that AI-assisted consultation can effectively democratize domain expertise access for novice customers, a mechanism not addressed in prior SME web system literature.

**Gap 2 — Partial or Absent Security in SME Web Systems.** The combination of HTTPS/SSL-TLS and AES-256 at-rest encryption—validated through structured security audit in Section IV-D—positions the X store system at a higher security assurance level than comparable systems reviewed in Specifically, Abdullah et al. [6] and Marcotte [7] implemented no security controls, while Setyo et al. [9] achieved only partial security. This finding suggests that security comprehensiveness in SME web systems is not inherently constrained by resource limitations, but rather by design prioritization—a consideration with direct implications for practitioners developing similar systems.

**Gap 3 — Absence of Comprehensive Multi-Layer Testing.** The tri-layer testing protocol (functional, performance, usability) yields mutually reinforcing findings: 100% functional correctness establishes behavioral reliability; performance results define operational capacity boundaries ( $\leq 25$  concurrent users); and SUS score 80.0 confirms user acceptance. Critically, the performance boundary finding (degradation at 50 users) is a contribution that would not have been discoverable through functional testing alone—demonstrating the necessity of multi-layer evaluation for production-grade SME systems, which prior studies in this domain have not reported.

#### 4.9.2 System Limitations and Boundary Conditions

Several limitations constrain the generalizability of the present findings. First, the performance degradation observed at 50 concurrent users reflects the constraints of shared hosting infrastructure; systems deployed on dedicated or cloud-hosted environments would be expected to exhibit substantially different scalability characteristics. Second, usability evaluation with  $n=5$  participants, while consistent with Nielsen's heuristic evaluation norm, limits statistical generalizability; a larger-sample SUS study would strengthen confidence in the reported score. Third, the AI consultation module's response quality is contingent on the capabilities and policy constraints of the selected external LLM provider — a dependency that introduces variability outside the system's direct control. Fourth, the single-store case study design limits external validity: findings are directly applicable to the X store operational context, and replication studies across other ornamental fish or niche-market SMEs would be required to establish generalizability.

#### 4.9.3 Implications for Practice

The results carry several practical implications for SME digitalization. The architecture demonstrates that a meaningful AI-assisted advisory capability is achievable within a student-developed, shared-hosting-deployed SME system — a finding that challenges the implicit assumption that AI integration requires enterprise-scale infrastructure. The modular LLM provider configuration is directly replicable in analogous niche-market retail contexts (e.g., rare plant retail, specialty pet supplies) where domain expertise barriers create similar customer advisory challenges. The security implementation protocol (HTTPS + AES-256 + parameterized queries) represents a replicable baseline for SME practitioners seeking to address the security gap identified across comparable systems in the literature.

## 5 CONCLUSION

This study has presented the design, implementation, and multi-dimensional evaluation of a web-based information system for Toko Oasis, an ornamental fish retail and aquascape service SME in Surabaya, Indonesia. The system satisfies all six target quality attributes: performance (mean response 4.2 s), security (HTTPS + AES-256), dependability (monitoring + backup), usability (positive end-user evaluation), portability (4-browser compatibility), and functional completeness (100% test pass rate).

Comparative analysis against six prior studies Table 1 and three analogous commercial platform categories Table 2 establishes the system's distinctive contributions: it is the only identified system to combine domain-specific AI consultation, comprehensive tri-layer testing, and full security implementation in an ornamental fish SME e-commerce context. The configurable AI backend (GPT-4/Gemini) and administrator-adjustable communication style represent architectural innovations with broader applicability to niche-market e-commerce systems.

Future work directions include: (1) domain-specific LLM fine-tuning on ornamental fish datasets to improve recommendation precision; (2) scalability stress-testing at  $>500$  concurrent users; (3) two-

<http://sistemasi.ftik.unisi.ac.id>

factor authentication (2FA) and periodic security auditing; (4) third-party logistics API integration for real-time delivery tracking; (5) longitudinal business impact assessment measuring revenue growth, customer retention, and order volume; and (6) extension of the AI module to support image-based fish species identification.

## References

- [1] A. S. Abdullah, H. Setiawan, and N. Umami, "Perancangan Sistem Informasi berbasis Website dengan Metode *Framework for the Applications of System Thinking*," *Jurnal Teknik Industri Untirta*, Vol. 1, No. 4, 2013.
- [2] A. Bharadwaj, O. A. El Sawy, P. A. Pavlou, and N. v. Venkatraman, "Digital Business Strategy: Toward a Next Generation of Insights," Vol. 37, ed: Management Information Systems Research Center, University of Minnesota, 2013, pp. 471-482.
- [3] K. Esaki, "System Quality Requirement and Evaluation," *Global Perspectives on Engineering Management*, Vol. 2, No. 2, pp. 52-59, 2001.
- [4] L. Giurgiu and I. Gligorea, "Responsive Web Design Techniques," in *International conference knowledge-based organization*, 2017, Vol. 23, No. 3, pp. 37-42.
- [5] A. R. Hevner, S. T. March, J. Park, and S. Ram, "Design Science in Information Systems Research," *Management Information Systems Quarterly*, Vol. 28, No. 1, p. 6, 2008.
- [6] L. Jacobson and J. R. G. Booch, "The Unified Modeling Language Reference Manual," 2021.
- [7] S. Kurnia, J. Choudrie, R. M. Mahbubur, and B. Alzougool, "E-commerce Technology Adoption: A Malaysian Grocery SME Retail Sector Study," *Journal of Business Research*, Vol. 68, No. 9, pp. 1906-1918, 2015.
- [8] C. Matt, T. Hess, and A. Benlian, "Digital Transformation Strategies," *Business & Information Systems Engineering*, Vol. 57, No. 5, pp. 339-343, 2015.
- [9] T. A. Militz and S. Foale, "The "Nemo Effect": Perception and Reality of Finding Nemo's Impact on Marine Aquarium Fisheries," *Fish and Fisheries*, Vol. 18, No. 3, pp. 596-606, 2017.
- [10] S. Nambisan, K. Lyytinen, A. Majchrzak, and M. Song, "Digital Innovation Management: Reinventing Innovation Management Research in a Digital World," Vol. 41, ed: Management Information Systems Research Center, University of Minnesota, 2017, pp. 223-238.
- [11] K. Peffers, T. Tuunanen, M. A. Rothenberger, and S. Chatterjee, "A Design Science Research Methodology for Information Systems Research," *Journal of Management Information Systems*, Vol. 24, No. 3, pp. 45-77, 2007.
- [12] P. R. Perkasa and E. Mailoa, "Adopsi Devsecops untuk mendukung Metode Agile menggunakan Trivy sebagai Security Scanner Docker Image dan Dockerfile," *Jurnal Indonesia: Manajemen Informatika dan Komunikasi*, Vol. 4, No. 3, pp. 856-863, 2023.
- [13] R. Rahayu and J. Day, "Determinant Factors of e-Commerce Adoption by SMEs in Developing Country: Evidence from Indonesia," *Procedia-Social and Behavioral Sciences*, Vol. 195, pp. 142-150, 2015.
- [14] A. L. Rhyne, M. F. Tlusty, P. J. Schofield, L. Kaufman, J. A. Morris Jr, and A. W. Bruckner, "Revealing the Appetite of the Marine Aquarium Fish Trade: The Volume and Biodiversity of Fish Imported into the United States," *PloS one*, Vol. 7, No. 5, p. e35808, 2012.
- [15] F. Ricci, L. Rokach, and B. Shapira, "Recommender Systems: Techniques, Applications, and Challenges," *Recommender Systems Handbook*, pp. 1-35, 2021.
- [16] B. Setyo, I. Ashar, and R. Budi, "Teknologi Rancang Bangun Ujian Sistem CBT (Computer Based Test) di Politeknik Angkatan Darat dengan Metode DevOps: Telekomunikasi," *Jurnal Telkommil*, Vol. 5, No. 1, pp. 24-34, 2024.
- [17] T. Tambunan, "Recent Evidence of the Development of Micro, Small and Medium Enterprises in Indonesia," *Journal of Global Entrepreneurship Research*, Vol. 9, No. 1, p. 18, 2019.