ISI Journal of **Information Systems and Informatics**

Web Information System for E-Sport Arena Community with OWASP-Based Cybersecurity Using XP Method (Case Study: Esport Arena Community)

Afgha Jameel Aufan[1], Purwadi[2], Argiyan Dwi Pritama[3]

[1]Information Technology Study Program, Amikom University Purwokerto, Indonesia
[2]Master of Computer Science, Amikom University Purwokerto, Indonesia
[3]Department of Information Systems, Amikom University Purwokerto, Indonesia
Email: 22sa31a040@mhs.amikompurwokerto[1], purwadi@amikompurwokerto.ac.id[2],
argiyandwi@amikompurwokerto.ac.id[3]

Abstract. The rapid development of the e-sports ecosystem in Purwokerto has encouraged the emergence of digital communities such as the Esport Arena Community. However, the management of member data, event information, and merchandise transactions is still carried out manually through social media, resulting in inefficiency and limited-service reliability. This study aims to develop a web-based information system that integrates member management, jersey evolution documentation, and secure online merchandise transactions. The system was developed using the Extreme Programming (XP) method, which supports iterative development and continuous refinement. Security measures were implemented based on OWASP Top 10 recommendations, including prepared statements, input validation, CSRF protection, and Role-Based Access Control (RBAC). System evaluation using the System Usability Scale (SUS) produced a score of 88, categorized as Excellent, indicating high user satisfaction and strong usability performance. The results demonstrate that the system operates securely, reliably, and effectively improves operational efficiency for the Esport Arena Community.

Keywords: e-sports community, web information system, Extreme Programming, OWASP Top 10, cybersecurity, usability

1. INTRODUCTION

The rapid growth of information technology in Banyumas, especially in Purwokerto as an educational and digital hub, has encouraged the formation of various online communities, including local e-sports groups. One of the most active is the Esport Arena Community, which routinely organizes tournaments, training sessions, and produces community merchandise. However, the management of these activities is still handled manually through social media, resulting in limited functionality and security risks [1].

E-sports continues to grow rapidly in Indonesia, supported by national organizations and increasing youth participation. This development creates opportunities for communities to monetize their activities through structured tournaments and merchandise sales, but requires a reliable and secure information system—something that social media platforms cannot provide [2].

Security remains a major challenge in modern web systems. Common vulnerabilities such as SQL Injection (SQLi) and Cross-Site Scripting (XSS) remain among the most frequently exploited attack vectors worldwide, making the adoption of OWASP Top 10 guidelines essential in system development [3]. Without proper implementation of security controls, user data, transaction information, and administrative processes may be exposed to cyberattacks.

Despite the availability of various e-commerce and community-management platforms, no existing solution integrates community profiles, event registration, jersey evolution documentation, merchandise sales, and OWASP-based security in a single system, especially for local e-sports communities such as the Esport Arena Community. This gap underscores the need for a secure, integrated web information system tailored to community operations [4].

To address these issues, this study proposes the development of a Web Information System for the Esport Arena Community using the Extreme Programming (XP) method. XP emphasizes short iterations, continuous testing, and active stakeholder involvement, making it suitable for systems requiring flexibility and rapid refinement. By integrating

XP with OWASP Top 10 security principles, the system is expected to support operational activities while ensuring secure and reliable transaction management [5].

2.   RESEARCH METHOD

Several methods were used in this study. The following sections describe the system development method, security method, system testing method, and system evaluation approach.

2.1 System Development Method

The system was developed using the Extreme Programming (XP) method because it is iterative and highly adaptive to changes in user requirements. The XP development cycle was executed in weekly iterations, consisting of planning, design, coding, testing, release, and feedback. Each iteration produced incremental improvements based on user stories from the Esport Arena Community, which is in line with previous studies that successfully applied XP in web-based information systems [6]. Several XP practices were applied, including simple design, pair programming, refactoring, test-driven development (TDD), continuous integration, and small releases. These practices ensured that new requirements could be incorporated efficiently while maintaining code quality, similar to other implementations of XP in academic and organizational information systems [7].

Several XP practices were employed throughout the development, including:

1) Simple Design: Focus on the essential features and avoid unnecessary complexity.
2) Pair Programming: Promoted collaboration between developers, ensuring code quality and reducing defects.
3) Refactoring: Continuous improvement of the system's codebase, enhancing maintainability.
4) Test-Driven Development (TDD): Ensured code quality through automated testing, validating system functionality with each iteration.
5) Continuous Integration: Facilitated frequent integration of new code, maintaining a stable version of the system.
6) Small Releases: Encouraged frequent, manageable releases of the system to gather user feedback quickly.

2.2 Security Method

Security development followed the OWASP Top 10 guidelines to prevent common web vulnerabilities such as SQL Injection, Cross-Site Scripting (XSS), and insecure authentication. OWASP provides structured recommendations for secure input handling, authentication, authorization, and configuration hardening, which are widely adopted in web security engineering and vulnerability assessment [8].

In this study, secure coding techniques were implemented using password hashing with password_hash() (bcrypt), server-side input validation and sanitization, and prepared statements to prevent SQL injection. Role-Based Access Control (RBAC) was applied to limit access to administrative features, and CSRF tokens were integrated into critical forms to prevent unauthorized requests. Security scans were also performed using OWASP ZAP to identify potential weaknesses such as SQLi, XSS, CSRF, and authentication vulnerabilities, following established OWASP-based testing practices [9]. Security Measures Implemented as follow.

1) Password hashing using bcrypt

   Passwords were securely hashed using the password_hash() function with the bcrypt algorithm, ensuring that sensitive user data remained protected.

```
$passwordHash = password_hash($password, PASSWORD_BCRYPT);
```

2) Input validation and sanitization

   All user inputs were sanitized and validated to prevent malicious code injection and XSS attacks.

```
$clean_input     =     htmlspecialchars(strip_tags($input),
ENT_QUOTES, 'UTF-8');
```

3) SQL Injection prevention using prepared statements

   Prepared statements were used to prevent SQL Injection, ensuring safe interaction with the database.

```
$stmt = $db->prepare("SELECT * FROM users WHERE email = ?");
$stmt->execute([$email]);
```

Role-Based Access Control (RBAC) was applied to restrict admin privileges. CSRF tokens were added to critical POST requests. Security scanning was performed using OWASP ZAP 2.14 to detect vulnerabilities such as SQLi, XSS, CSRF, and authentication weaknesses [10].

2.3 System Evaluation

Usability evaluation of the E-Sport Arena Community platform was conducted using the System Usability Scale (SUS) with ten respondents. Based on the standardized SUS scoring procedure, the system achieved a final score of 88, which is categorized as "Excellent" and corresponds to a Grade A usability rating. This result indicates that users perceive the platform as highly usable—easy to learn, efficient to navigate, and comfortable for completing essential tasks such as browsing products and making purchases. The SUS method was selected because it is a simple yet robust measurement tool widely adopted in evaluating web applications and online platforms [11].

3. RESULTS AND DISCUSSION

3.1 E-Commerce Process Analysis

The analysis of the Jersey E-commerce process on the Esport Arena Community Website system. Explanation of Figure 1 os E-Commerce Business Flow, this flow shows the integration of processes between users, e-commerce systems, ordering modules, payment processes, stock management, and data storage in databases [12]. Overall, the business process is divided into five main components that coordinate with each other, namely Customer, E-Commerce Website, Order & Payment System, Stock & Delivery, and Database.
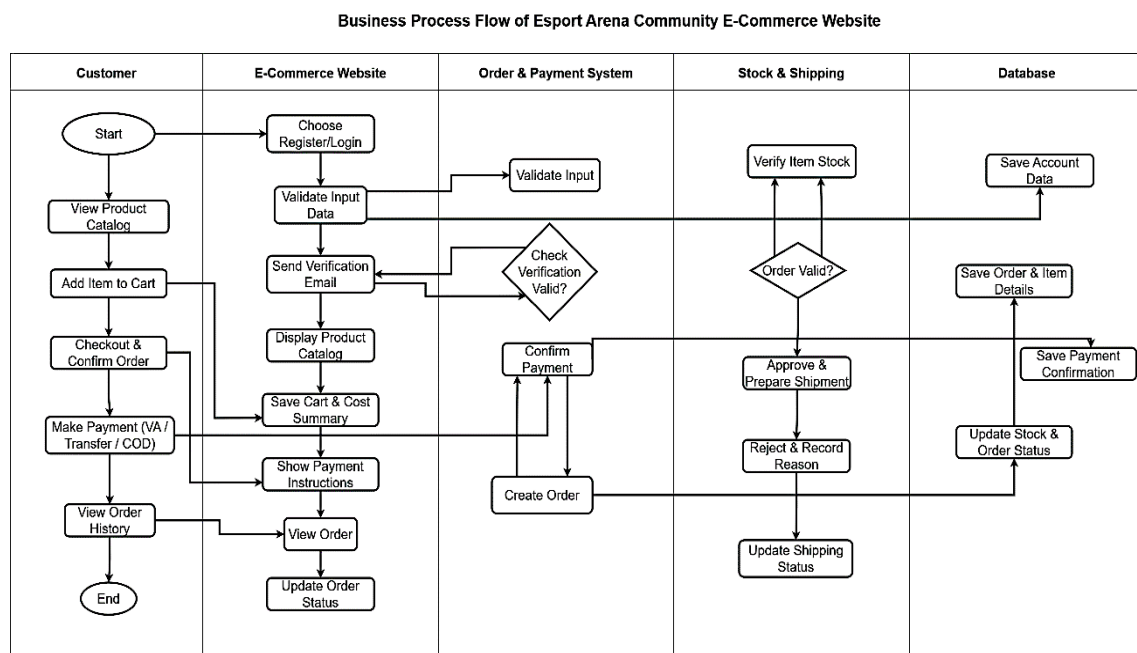
Figure 1. E-Commerce Website Business Flow

At the Customer stage, users begin the process by registering or logging in to gain access to the available features. Once verified, customers can browse products, add items to their cart, proceed to checkout, and complete payments—activities that initiate the internal transaction workflow [13]. The E-Commerce Website section manages user–system interactions such as input validation, verification e-mail handling, catalog display, and payment instruction generation. It also synchronizes order statuses based on responses from the payment module [14].

The Order & Payment System processes core transactional activities including storing new account data, validating verification links, and processing uploaded payment receipts. After successful validation, the system generates order records and detailed order items for the fulfillment pipeline [15]. The Stock & Delivery stage ensures availability of ordered products. The system verifies stock, approves or rejects orders, initiates packaging, and coordinates shipping activities. The order status is then updated to ensure customers can track delivery progress [16]. All processes are recorded in the Database component, which stores user profiles, orders, payment records, inventory data, and shipping logs. Structured data management ensures consistency, auditability, and integrity across the system [17]. The business flow in Figure 1 illustrates an integrated e-commerce process that connects user interaction, transaction processing, inventory management, and delivery operations into a secure and efficient workflow.

3.2 System Architecture

The system architecture in this study adopts the three-tier architecture concept, which separates the application into three main layers—Presentation Layer, Application Layer, and Data Layer. This architecture improves system organization, maintainability, and scalability, and aligns with modern web-application design practices [18].

1) Presentation Layer

This layer functions as the user interface that directly interacts with end-users. In the E-Sport Arena e-commerce system, it includes the homepage, product catalog, product detail pages, shopping cart, checkout interface, payment proof upload page, and the admin dashboard. The presentation layer is designed to be fully responsive, ensuring

accessibility across multiple device types and providing an optimized user experience [19].

2)   Application Layer

This layer handles the core business logic, including authentication, cart management, payment calculation, order validation, payment verification, stock management, and shipping status updates. The application layer serves as the bridge between the user interface and the database, ensuring that all system processes are executed accurately based on user requests. The separation of business logic in this layer is a fundamental principle in modern web-system design [20].

3)   Data Layer

This layer consists of a MySQL relational database that stores all system data, including user accounts, products, orders, order items, payments, and shipping histories. The relational model supports data integrity through well-structured tables and relationships, ensuring consistent and reliable transaction processing within the e-commerce system [21].
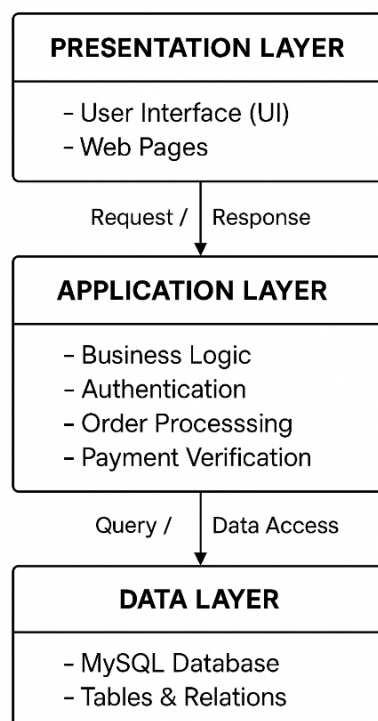
Figure 4. System Architecture

The system architecture was designed using a three-tier architecture, which consists of the Presentation Layer, Application Layer, and Data Layer. The Presentation Layer serves as the user interface that manages all interactions with the system. The Application Layer handles core business logic, including authentication, transaction processing, and order management. Meanwhile, the Data Layer is responsible for storing and managing all information in the MySQL database.

3.3 System Requirements Analysis

System requirements analysis was conducted to identify the main services and quality specifications that must be met in order for the E-Sport Arena Community e-commerce information system to function optimally. These requirements were formulated based on the results of business flow analysis, user characteristics, and the iterative development principles applied in the Extreme Programming (XP) method. In general, system requirements are divided into two categories, namely functional requirements that describe the main features that must be available, and non-functional requirements that explain the quality standards that must be met by the system [22].

3.3.1 Functional Requirements

Functional requirements are core functions that must be provided in order for e-commerce business processes to run smoothly. The identification of functional requirements follows the standard practices of software engineering, where each required function is mapped to ensure correct system behavior and alignment with business goals [23]. Details of these requirements are presented in Table 1.

Table 1. Functional Requirements

No	Requirement	Description	Priority
1	User registration	Users can register using email and password	H
2	User login	Users can log in after account verification	H
3	Email verification	System sends verification link	H
4	Product catalog	System displays available jerseys	H
5	Shopping cart	Users can add and manage items	H
6	Checkout	Displays order summary and total cost	H

No	Requirement	Description	iority
7	Payment instructions	Provides QRIS/Bank/VA payment info	H
8	Upload proof of payment	Users upload payment receipt	H
9	Payment validation	Admin/system validates payment	H
10	Product management	Admin add/update/delete products	H
11	Stock verification	System checks availability	H
12	Order status update	Auto status: pending→processing→sent→done	H
13	Order history	Users can view transaction records	M
14	Transaction recording	Orders, payments stored in DB	H
15	Admin dashboard	Admin monitors operations	H

3.2.2 Non-Functional Requirements

Non-functional requirements describe system quality standards, covering aspects of security, performance, reliability, and ease of use. These requirements are important for maintaining transaction stability and security, especially since the system processes personal data and payments [24]. Details of non-functional requirements are presented in Table 2.

Table 2. Non-Functional Requirements

| No	Requirement	Description
1	Security	Implements OWASP (SQLi, XSS, CSRF), password hashing
2	Performance	Fast response time for login, checkout, payment
3	Usability	UI is simple, intuitive, mobile-responsive
4	Reliability	Ensures data consistency and error minimization
5	Availability	Accessible during peak hours with minimal downtime
6	Scalability	Supports increasing users and feature expansion
7	Maintainability	Modular code structure for easier updates
8	Session Timeout	Automatic logout after inactivity
9	Password Strength Policy	Minimum length + mix of characters

No	Requirement	Description
10	Audit Logging	Records critical user and admin actions

3.4 System Design

The system design phase transforms the requirements analysis into a technical blueprint that guides the development of the E-Sport Arena Community e-commerce system. This phase follows Extreme Programming (XP) design principles, which emphasize simplicity, clarity, and iterative refinement throughout the development cycle [25]. The system adopts a three-tier architecture consisting of the Presentation Layer, Application Layer, and Data Layer. The Presentation Layer handles user interaction, the Application Layer manages business logic and transaction workflows, while the Data Layer stores and organizes information using a MySQL relational database. This layered architecture enhances modularity, maintainability, and scalability, enabling new features to be added without affecting existing components [26].

3.5 Use Case Diagram

The use case diagram illustrates how two main actors—Customer and Admin—interact with System for Esport Arena, as shown in Figure 5. The Customer uses the system to perform essential shopping activities such as creating an account, logging in, browsing available products, managing their shopping cart, completing the checkout process, uploading payment proof, and monitoring the status of their orders from start to finish.
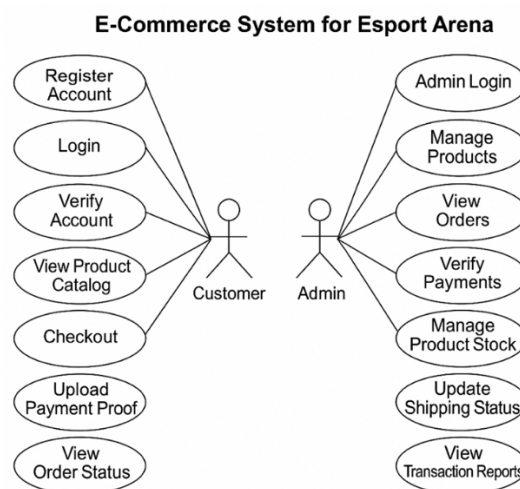

Figure 5. Use Case Diagram

Meanwhile, the Admin is responsible for handling the system's operational tasks. Through the admin dashboard, the admin can log in securely, manage product information and stock availability, review incoming orders, validate customer payments, update shipping progress, and access transaction reports. This diagram shows how both users and administrators work within the system to support smooth, secure, and well-organized e-commerce operations.

3.6 Class Diagram

The class diagram illustrates the main database entities used in the Esport Arena e-commerce system and the relationships between them, as shown in Figure 6. The users table stores customer account and profile information, which is linked to orders, cart, and payments. The jersey_category and jersey_product tables manage product classification and jersey details, including size, stock, weight, and active status. Each order is recorded in the orders table and detailed in order_details, which stores the items purchased, quantity, price, and shipping status. The payments table records payment method, proof of payment, amount, and verification status. These relationships support consistent transaction recording and ensure that customer data, product information, and payment records are integrated in a structured way.
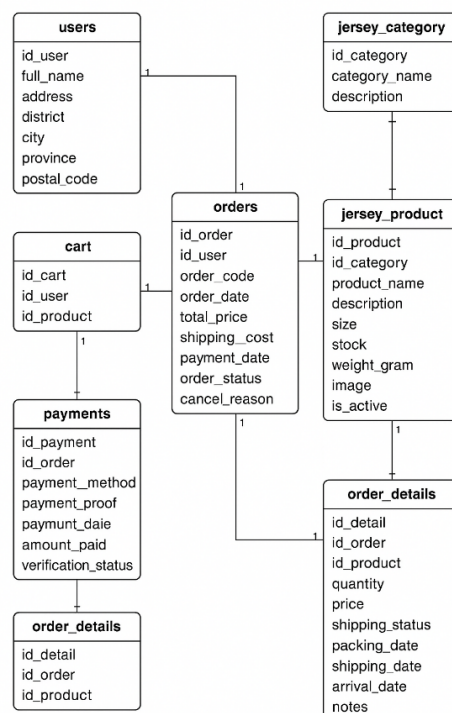

Figure 6. Class Diagram

3.7 Sequence Diagram

Sequence diagrams illustrate the flow of interactions between users and system components during key processes. These diagrams help visualize how requests are handled step-by-step and ensure that the system logic is clearly defined and consistent. Login sequence describes the authentication process when a user attempts to log into the system, as shown in Figure 7. The user submits their email and password, which are forwarded to the authentication controller. The system then validates the credentials, checks account activation status, and returns a success response if the data is correct. If validation fails, the system responds with an error message. This mechanism ensures secure and controlled access to the platform.
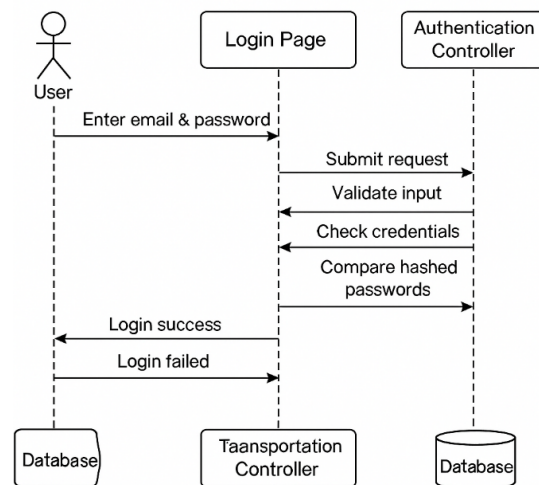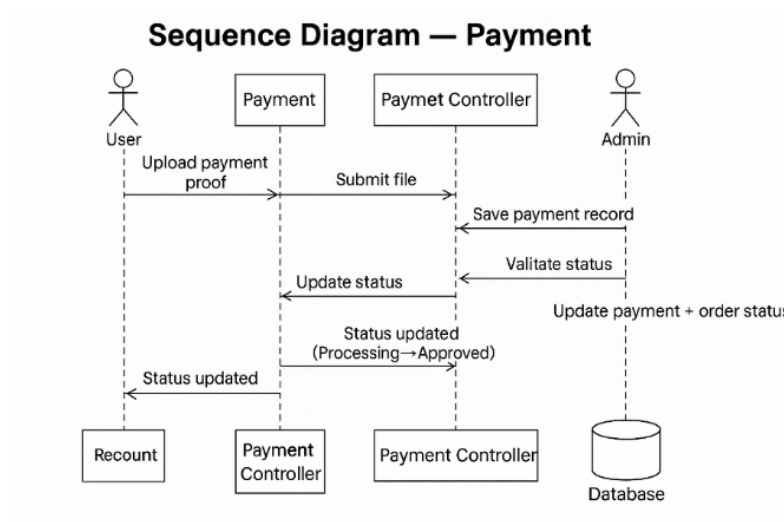

Figure 7. Login Sequence Diagram

This sequence diagram illustrates in Figure 8 is the checkout workflow. The customer begins by confirming the items in the shopping cart. The system retrieves product data, calculates the total cost, validates the shipping address, and generates an order record. After verification, the system forwards the process to the payment module. This structured sequence ensures accuracy in order generation and smooth transition to the payment stage. Figure 9 explains the payment confirmation process. The user uploads proof of payment, which is received by the payment controller and stored in the database. The system then notifies the admin to validate the payment. Once approved, the order status is updated to "Paid" or "Processing." This interaction ensures secure payment verification and accurate transaction tracking throughout the order cycle.
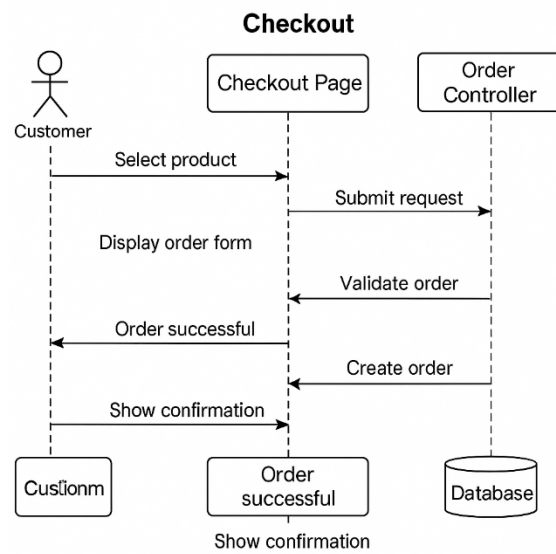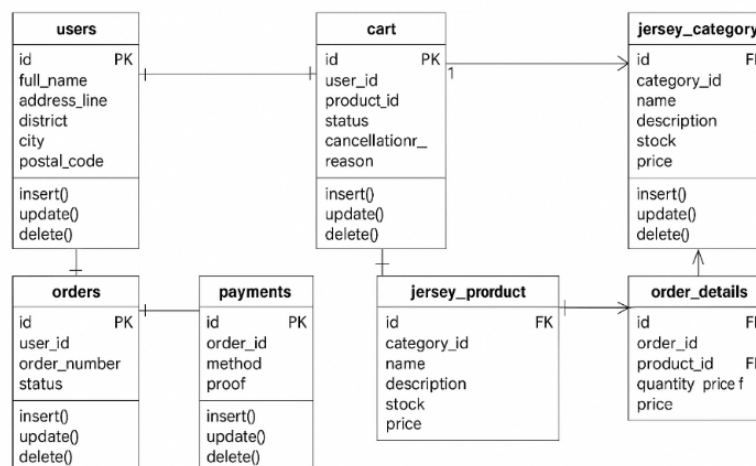
Figure 8. Checkout Sequence Diagram

Figure 9. Payment Sequence Diagram

3.8 Database Design

The database structure of this system is illustrated through an Entity–Relationship Diagram (ERD), which provides an overview of how data is organized and connected within the Esport Arena e-commerce platform. The ERD is composed of six core entities—users, cart, orders, payments, jersey_product, and order_details—each representing a key part of the system's operational workflow.

Every entity contains attributes that support essential activities such as managing user profiles, storing product information, processing transactions, verifying payments, and handling order fulfillment. The connections between entities reflect typical e-commerce behavior: a single user may create multiple orders, each order may include several order items, and every product is associated with a specific category. This database design helps maintain clear data flow across the system, ensures the accuracy of relationships between records, and supports efficient data retrieval. By structuring the system in this way, the application becomes easier to maintain, more reliable, and capable of supporting future feature expansion without disrupting existing components.


Figure 10. ERD System

3.9 Interface Design

The interface design for the E-Sport Arena Community web system was developed to ensure clarity, ease of use, and consistent navigation across all modules. The design follows modern UI/UX principles, including simplicity, visual hierarchy, and consistent component placement. Only key interface pages are presented in this section to maintain article conciseness, while the complete UI collection is provided in the Appendix as recommended by the reviewer.

1) Customer Home Page Display

Figure 11 shows the design of the customer home page, which serves as the main entry point after the user successfully logs in. This page provides access to essential features such as tournament information, recent event listings, merchandise navigation, and user

profile controls. The interface uses a dark color theme to increase visual comfort and highlight interactive elements, while the layout follows a structured grid arrangement to ensure the content is easy to read and well organized. Key actions, including viewing tournament schedules and navigating between menus, are placed in the upper area to support fast access and intuitive user interaction. This design approach improves usability, clarity, and responsiveness for customer browsing activities. Figure 12 shows the registration page used by users to create new accounts in the system. This page contains several input fields, including username, email, password, and password confirmation. The layout is designed to be simple and easy to understand, helping users complete the registration process efficiently. The form also includes eye-icons to toggle password visibility, improving convenience and reducing input errors. Overall, this interface prioritizes clarity and usability to support a smooth onboarding experience for new users.
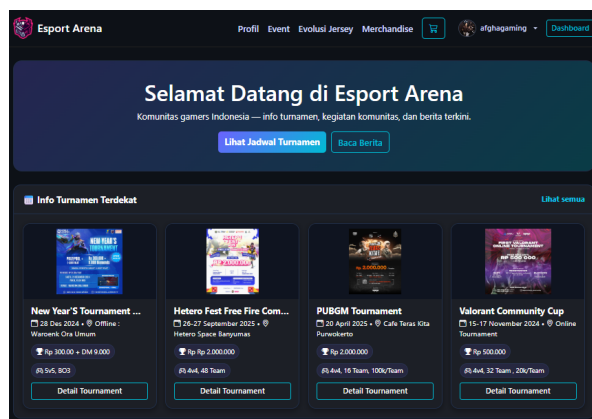

Figure 11. Customer Home Page Display
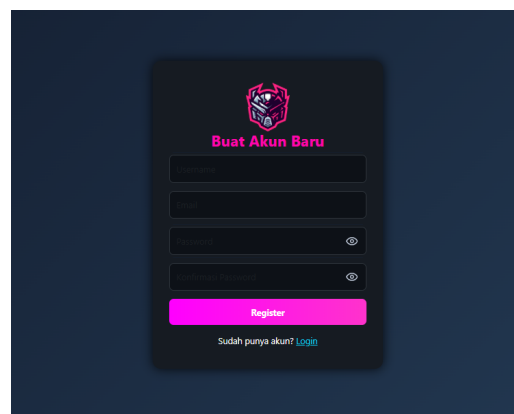
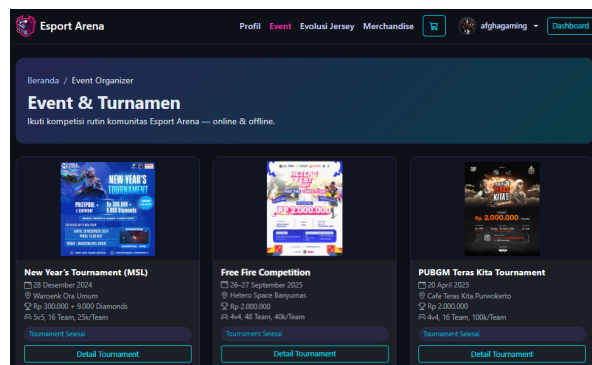**Figure 12.** Registration Page Display
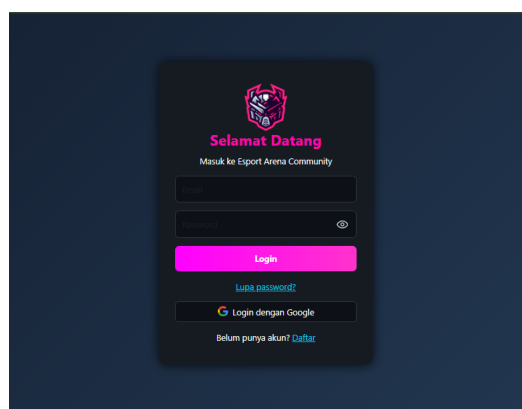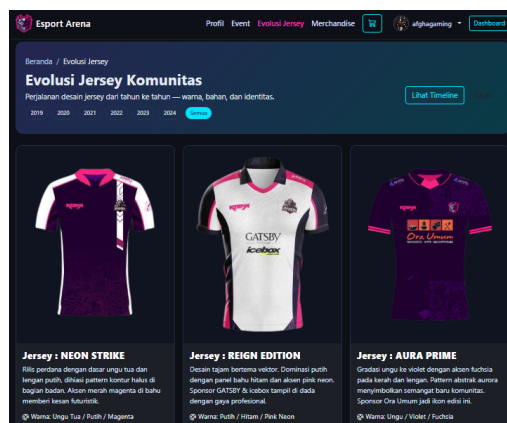

Figure 13. Login Page Display

Figure 13 shows the login page that users interact with to access the system. The interface is designed to be simple and focused, providing only the essential input fields email and password—to streamline the authentication process. The page also includes additional options such as password visibility toggles and an alternative login method via Google to enhance usability and reduce login friction. Figure 14 shows the community profile page, which functions as an information center for users. On this page, the system presents details about the Esport Arena Community, including its background, identity, programs, and organizational structure. The layout combines text and visual elements to ensure that information is easy to read, well-organized, and visually engaging.
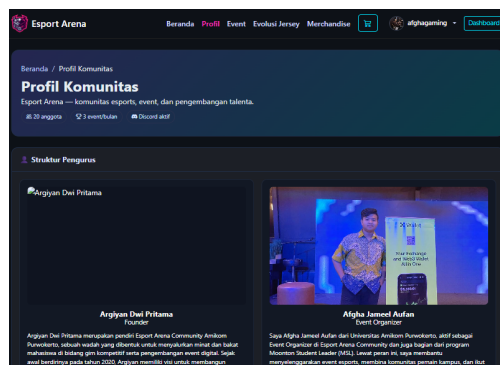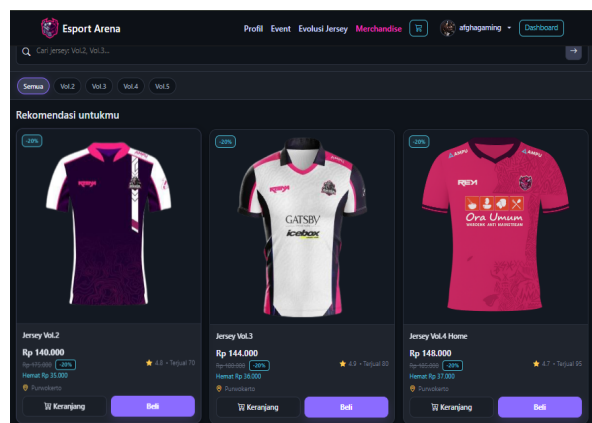

Figure 14. Profile Page Display

Figure 15. Event Page Display

Figure 16. Jersey Evolution Page

Figure 17. Shop Page Display

Figure 15 illustrates the event page of the system, which is used to display a list of tournaments organized by the Esport Arena community. At the top of the interface, the page displays a header containing the event title and a short description that informs users about the category and purpose of the tournaments. Each tournament card includes key information such as the event title, date, registration status, and a button

to access detailed information. This structure helps users browse tournaments efficiently and supports the community's competitive activities by providing clear and accessible event listings. Figure 16 explains the Jersey Evolution Page Display, which shows the progression of official community jersey designs over several years. At the top of the page, users can see a title and brief introduction explaining the purpose of the jersey evolution feature. Each jersey card includes an image, name, and short description that highlights the design concept and release year. This page helps users understand the branding development of the Esport Arena community and serves as a visual archive of previous jersey generations, making it easier for members to appreciate the history and design identity of the community. Figure 17 illustrates the shop page interface used by customers to browse and purchase official Esport Arena merchandise. The page presents a structured product catalog containing jersey images, prices, and quick purchase actions, supported by a search bar and category filter that help users find merchandise based on edition or jersey type, ensuring a smooth and efficient shopping experience.
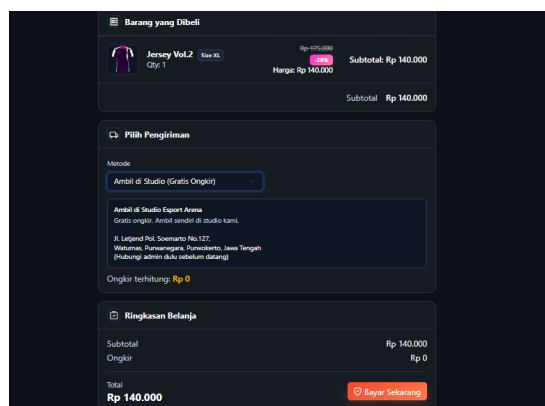
Figure 18. Checkout Page Display
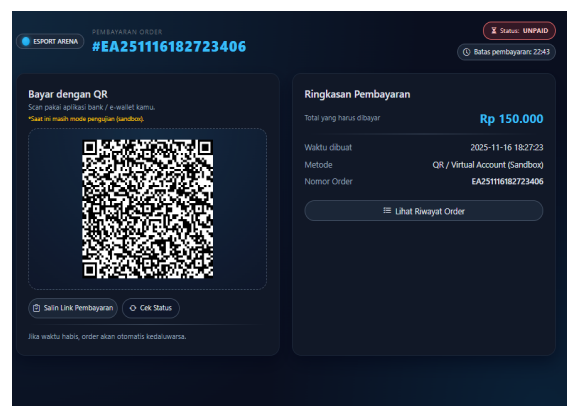
Figure 19. Payment Page Display

Figure 18 shows the checkout page, where customers review their order before completing the purchase. The page displays product details, shipping information, delivery address, and a summary of total costs. At the bottom, a confirmation button is provided to finalize the transaction, ensuring customers can easily verify all order components prior to payment. The layout is designed to maintain clarity, reduce errors, and improve the overall checkout experience. Figure 19 shows the payment page, that customers use to finalize their transactions after completing checkout. The page is divided into two main sections. On the left side, a QRIS code is provided so users can make instant payments using banking applications or digital wallets that support QR

scanning. On the right side, the system displays a detailed payment summary, including the total amount, payment deadline, and order information. This layout helps users clearly understand the required payment steps and ensures that the transaction process can be completed efficiently.

Figure 20 shows the order history page, which provides customers with a complete list of transactions they have made within the system. This page presents essential information such as order number, order date, total payment, and current order status. The status labels—such as Paid, Unpaid, Processing, or Completed—allow users to easily monitor the progress of each order. This interface helps customers track past purchases and verify payment or delivery statuses efficiently. Figure 21 illustrates the user dashboard page, which serves as the main control center after customers successfully log in. The dashboard provides a clear summary of essential information, including the total number of orders, pending orders, completed payments, and overall transaction costs. This summary section helps users quickly understand the status of their shopping activities. Additionally, the page offers easy access to profile details, order history, and other navigation menus, allowing users to manage their account and transactions efficiently.

Figure 20. Order History Page Display

Figure 21. User Dashboard Page

Figure 22 explains the customer tracking page that allows users to monitor the real-time progress of their orders. At the top of the interface, a status navigation bar visually represents each stage of the delivery process—including Packing, Shipped, In Transit, and Delivered—so users can easily understand the current position of their order. The page also provides detailed information such as shipping address, courier data, item summary, and total payment, helping customers verify order accuracy and track delivery more

transparently. Figure 23 shows the customer profile page, which is used to manage personal information and shipping details. The page contains a structured form with fields such as username, full name, email address, phone number, province, city, district, sub-district, postal code, and full address. The layout is designed to support clear data entry and easy updates, ensuring customers can maintain accurate information for order processing and delivery.
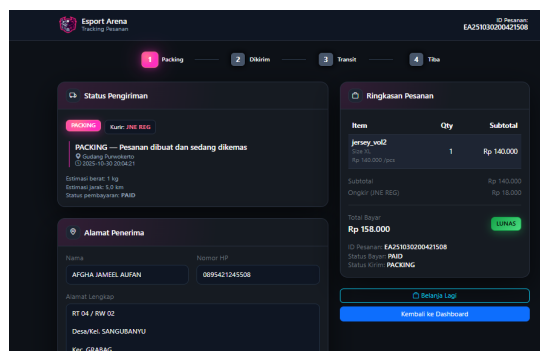
Figure 22. Customer Tracking Page
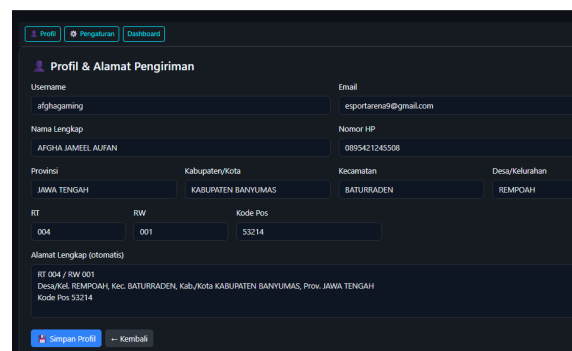
Figure 23. Customer Profile Page

Figure 24. Admin Profile Page
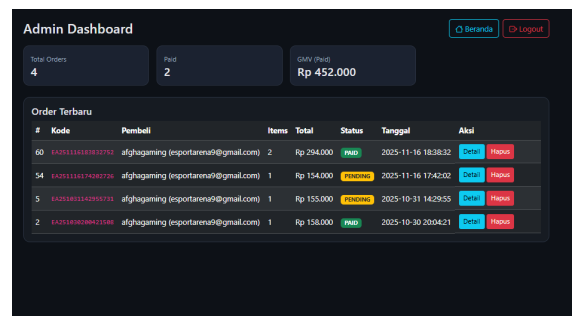
Figure 25. Admin Dashboard Page

Figure 24 shows the admin profile page, which is used to manage account information and security settings. On the left side, the page displays the admin's profile panel containing the profile photo, username, email, phone number, and other d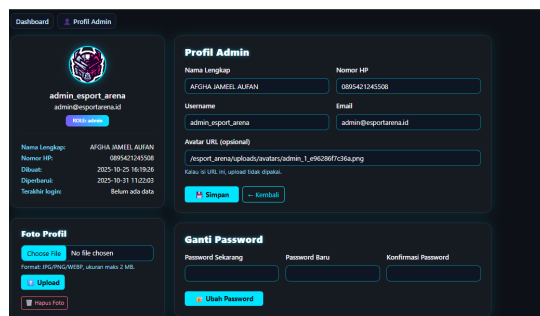etails such as account creation date and access level. On the right side, there are input fields for updating profile information and a dedicated section for changing the password, ensuring that administrators can manage their credentials securely and efficiently. Figure 25 explains the admin dashboard page, which serves as the central control panel for administrators in managing the e-commerce system. At the top of the dashboard, a set of summary cards provides key metrics such as total orders, pending orders, and total revenue. Below the summary section, a detailed table lists recent transactions, including order codes, customer email addresses, order totals, payment statuses, dates, and action

buttons for viewing, verifying, or updating each order. This dashboard layout helps administrators monitor system activity efficiently and manage operations quickly. Figure 26 shows the completed order page that appears after the transaction and delivery process have been fully finalized. At the top of the interface, the system presents key information such as the order number, completion date, and the final status labeled "Completed." The order summary panel provides complete delivery information including the recipient's name, phone number, full address, and detailed cost breakdown consisting of the subtotal, shipping cost, and total payment. This page ensures that customers can clearly verify the final results of their order, improving transparency and user experience during post-purchase activities.
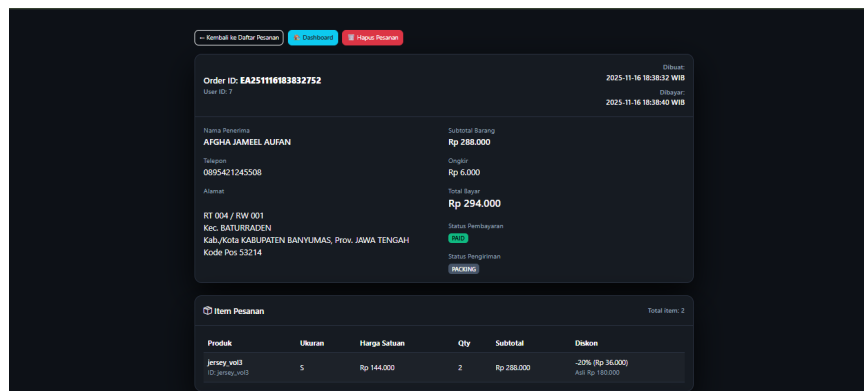
Figure 26. Completed Order Page Display

4.   Usability Evaluation (System Usability Scale – SUS)

Usability evaluation of the system was conducted using the System Usability Scale (SUS), a standardized instrument consisting of 10 items designed to measure users' perceived usability. In this study, the SUS questionnaire was adapted so that only two items represent negative statements, while the remaining items capture positive perceptions of usability. This adjustment is appropriate because the tested e-commerce system already demonstrates a high level of usability and does not require excessive negative phrasing. All items were rated using a 1–5 Likert scale, where 1 means "Strongly Disagree" and 5 means "Strongly Agree." The complete list of questions used in the evaluation is presented in Table 3.

Table 3. SUS Questionnaire System

No	SUS Questions	Scale
1	I think I would use this system frequently.	1–5

Vol. 7, No. 4, December 2025

ISI *Journal of*
Information Systems and Informatics

Published By
Asosiasi Doktor
Sistem Informasi Indonesia

2	I find this e-commerce system easy to use.	1–5
3	I feel confident using this system.	1–5
4	I think this system is unnecessarily complex.	1–5
5	I think the features in this system are well integrated.	1–5
6	I believe most people would learn to use this system very quickly.	1–5
7	I feel the system's navigation is clear and understandable.	1–5
8	I think I would need technical support to use this system.	1–5
9	I find the system functions to be consistent throughout the pages.	1–5
10	I feel the system is easy to learn, even for new users.	1–5

The items listed in Table 3 represent the adapted SUS questionnaire used to evaluate user perceptions of the system's usability. Items 1, 2, 3, 5, 6, 7, 9, and 10 are positive items that measure ease of use, clarity, learnability, and user confidence. Items 4 and 8 serve as negative control items, intended to identify potential usability barriers such as unnecessary complexity or dependence on external assistance. Each respondent rated the items on a 1–5 scale, and the scores were processed using the standard SUS scoring method, where positive items are scored as (Rating – 1) and negative items are scored as (5 – Rating). This approach produces a validated usability score on a 0–100 scale, enabling objective interpretation according to industry standards. After distributing the adapted SUS questionnaire consisting of 10 items—eight positive statements and two negative control items—ten respondents provided usability ratings for the e-commerce system on a 1–5 Likert scale. Each response was scored using the standard SUS calculation method, where positive items are converted using *(Rating – 1)* and negative items using *(5 – Rating)*. The converted values were then summed and multiplied by 2.5 to produce a final SUS score for each respondent on a 0–100 scale. Table Y presents the SUS score results from all participants.

Table 4. SUS Respondent System

Respondent	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10	SUS Score
R1	4	4	4	2	4	4	4	2	4	4	82.5
R2	5	4	5	1	5	5	5	1	5	4	90.0
R3	4	4	4	2	4	4	4	2	4	4	82.5

Respondent	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10	SUS Score
R4	5	5	5	1	5	5	5	1	5	5	95.0
R5	4	4	4	2	4	4	4	2	4	4	82.5
R6	5	5	5	1	5	5	5	1	5	5	95.0
R7	4	4	4	2	4	4	4	2	4	4	82.5
R8	5	4	5	1	5	5	5	1	5	4	90.0
R9	4	4	4	2	4	4	4	2	4	4	82.5
R10	5	5	5	1	5	5	5	1	5	5	95.0
Average											**88.0**

The results in Table 4 show that all respondents provided high usability ratings for the system, resulting in an average SUS score of 88.0**.** According to the standard SUS interpretation scale, a score above 80 falls within the "Excellent" category and corresponds to a Grade A usability rating. This indicates that the users perceive the system as intuitive, efficient, and highly comfortable to use. The distribution of scores across respondents is consistent, with all values falling within the 82.5–95.0 range, showing that the usability experience is stable and not significantly affected by individual differences. The low disagreement on negative items (Q4 and Q8) indicates that users did not perceive unnecessary complexity or the need for technical assistance when using the system. These findings confirm that the system successfully meets usability expectations, offering smooth navigation, well-integrated functionality, and an interface design that supports fast task completion for both new and experienced users.

3.10   Discussion

The results of this study provide valuable insights into the design, functionality, and usability of the Esport Arena Community e-commerce system. The analysis covered multiple aspects of the system, from the business flow and system architecture to usability evaluation using the System Usability Scale (SUS). These results contribute to understanding how the system meets user needs, supports business processes, and delivers an intuitive user experience.

The integration of key business components—Customer, E-Commerce Website, Order & Payment System, Stock & Delivery, and Database—forms a seamless and efficient e-commerce workflow. This well-structured flow, depicted in Figure 1, illustrates how each component interacts to ensure smooth transactions. The system starts with customer registration and progresses through product selection, payment processing, and order fulfillment, ensuring all data is properly stored in the database for easy retrieval. This comprehensive process aligns with best practices in e-commerce systems, emphasizing reliability, security, and data integrity across all stages of interaction.

The adoption of a three-tier architecture (Presentation, Application, and Data Layers) ensures the system is modular, maintainable, and scalable. By separating the user interface, business logic, and data storage, the architecture enhances the system's flexibility, allowing for future updates and expansions without disrupting existing operations. Moreover, the layered structure allows for easier debugging, testing, and maintenance, ensuring long-term stability. The presentation layer's responsiveness across devices further enhances user experience, aligning with modern design principles.

The analysis of functional and non-functional requirements revealed the core services and quality standards essential for the system's success. Functional requirements, such as user registration, payment processing, product catalog management, and order tracking, are fundamental to the smooth operation of the e-commerce platform. These features, outlined in Table 1, support all key business operations and ensure that users can perform essential tasks with ease.

In terms of non-functional requirements, the system excels in security, performance, and usability. The use of OWASP security measures (e.g., SQL injection prevention, password hashing, and CSRF protection) ensures that the platform adheres to modern web security standards, reducing the risk of common vulnerabilities. Performance requirements, including fast response times during checkout and payment processes, are met, contributing to a seamless user experience. The system's reliability and scalability also ensure that it can handle increasing user loads and adapt to future feature expansions, making it a robust solution for long-term use.

The usability evaluation, conducted using the System Usability Scale (SUS), demonstrated that the system provides an exceptional user experience. The average SUS score of 88.0 falls into the "Excellent" category, indicating that users perceive the system as intuitive, efficient, and easy to use. This result is consistent with the system's design principles, which prioritize simplicity, clarity, and ease of navigation.

The positive ratings across all respondents reflect a strong consensus that the system is user-friendly. Users reported confidence in navigating the platform, with a majority agreeing that the system was easy to learn and that the features were well integrated. The minimal disagreement on negative statements (such as complexity and the need for technical support) further confirms that the system is well-optimized for a smooth user experience. These findings underscore the effectiveness of the iterative development approach used in this study, where user feedback was continuously incorporated to enhance the usability of the system.

Moreover, the system's design, including intuitive layout, clear navigation, and responsive interface, contributed to a seamless user experience. The ease of completing tasks such as account creation, browsing products, and processing payments ensures that users, both new and experienced, can navigate the platform with minimal friction. The consistency of scores across different respondents further validates the system's reliability in delivering a user-friendly experience to a broad audience.

Security remains a crucial aspect of any e-commerce platform, especially one that handles sensitive data such as payment information and user profiles. This system adheres to the OWASP Top 10 security guidelines, implementing robust measures such as password hashing, input validation, prepared statements to prevent SQL injection, and the use of CSRF tokens to protect against unauthorized requests. These security practices, combined with regular security testing using tools like OWASP ZAP, contribute to the system's resilience against common vulnerabilities, ensuring a safe environment for users to interact with the platform.

In terms of performance, the system demonstrates fast response times for critical operations, such as login, checkout, and payment processing, which are essential for maintaining a positive user experience. The system's ability to handle these tasks

efficiently without significant delays contributes to the overall usability and satisfaction of the users.

While the system has shown strong results in terms of usability and functionality, there are areas where further improvements can be made. For instance, incorporating additional personalization features, such as personalized product recommendations based on user behavior, could enhance the shopping experience. Additionally, expanding payment options to include more localized payment methods could increase accessibility for a broader range of users. Furthermore, integrating more advanced analytics tools to track user interactions and optimize system performance could provide valuable insights for future iterations.

4   CONCLUSION

The development of this e-commerce information system provides a seamless, integrated platform that streamlines both merchandise ordering and community engagement for Esport Arena users. Customers can easily browse product details, place orders, complete payments, and track delivery status directly through the website, eliminating the need for manual interaction with administrators. For administrators, the system offers centralized tools to efficiently manage product information, validate payments, and oversee ongoing transactions via a dedicated dashboard. Functional and usability evaluations, including the System Usability Scale (SUS) assessment, confirm that all core features are fully operational and the system provides an excellent user experience. These findings demonstrate that the system effectively meets the operational needs of the Esport Arena Community and is well-positioned to support future feature expansions, ensuring long-term scalability and adaptability.

REFERENCES

[1]   P. D. K. Sujata, S. B. Sardjono, and S. P. Hendratno, "Esport ecosystem, financial behavior, and carbon emissions in Indonesian urban area," in *Proc. 4th Asia Pacific Information Technology Conf.*, New York, USA, 2022, pp. 123–130, doi: 10.1145/3512353.3512371.

[2] S. A. Pradana, R. M. Pikahulan, M. A. Alvian, and S. Adriana, "Regulation of esports in the context of employment in Indonesia," *Amsir Law Journal*, vol. 4, no. 1, pp. 15–31, Oct. 2022, doi: 10.36746/alj.v4i1.98.

[3] S. D. Hilda, N. Heryana, and A. A. Ridha, "Website security analysis for Curug Village Government using the Open Web Application Security Project (OWASP)," *Jurnal Informatika dan Teknik Elektro Terapan*, vol. 12, no. 3S1, Oct. 2024, doi: 10.23960/jitet.v12i3S1.5236.

[4] M. Syarifudin, L. Widyawati, and O. Asroni, "Web security vulnerability analysis and mitigation based on OWASP Top 10," *Journal of Artificial Intelligence and Engineering Applications*, vol. 4, no. 3, pp. 1829–1834, Jun. 2025, doi: 10.59934/jaiea.v4i3.1029.

[5] I. Purnama, "Clinical information system using Extreme Programming method," *International Journal of Science, Technology & Management*, vol. 4, no. 5, pp. 1229–1235, Sep. 2023, doi: 10.46729/ijstm.v4i5.931.

[6] O. Fenardi and F. S. Lee, "Aplikasi akademik berbasis website menggunakan metode Extreme Programming pada SMAN 1 Belinyu," *Jurnal Teknologi dan Sistem Informasi Bisnis*, vol. 5, no. 4, pp. 440–447, Oct. 2023, doi: 10.47233/jteksis.v5i4.843.

[7] S. D. Pohan and I. Firdaus, "Implementation of Extreme Programming method in the development of Pekanbaru community training information system," *Cyberspace: Jurnal Pendidikan Teknologi Informasi*, vol. 6, no. 1, pp. 20–29, Mar. 2022, doi: 10.22373/cj.v6i1.11851.

[8] D. Priyawati, S. Rokhmah, and I. C. Utomo, "Website vulnerability testing and analysis of website applications using OWASP," *International Journal of Computer and Information System*, vol. 3, no. 3, pp. 142–147, Aug. 2022, doi: 10.29040/ijcis.v3i3.90.

[9] O. Iparraguirre-Villanueva, F. Sierra-Linan, and M. Cabanillas-Carbonell, "Location-based mobile application for blood donor search," *International Journal of Advanced Computer Science and Applications*, vol. 13, no. 4, pp. 1–7, 2022, doi: 10.14569/IJACSA.2022.0130418.

[10] F. P. E. Putra et al., "Systematic literature review: Security gap detection on websites using OWASP ZAP," *Brilliance: Research of Artificial Intelligence*, vol. 4, no. 1, pp. 348–355, Jul. 2024, doi: 10.47709/brilliance.v4i1.4227.

[11] O. Suria, "A statistical analysis of System Usability Scale (SUS) evaluations in online learning platforms," *Journal of Information Systems and Informatics*, vol. 6, no. 2, pp. 992–1007, Jun. 2024, doi: 10.51519/journalisi.v6i2.750.

[12]   P. Palee, P. Wannapiroon, and P. Nilsook, "The architecture of intelligent career prediction system based on cognitive technology," *International Journal of Advanced Computer Science and Applications*, vol. 11, no. 12, pp. 1–8, 2020, doi: 10.14569/IJACSA.2020.0111214.

[13]   M. L. Sheng, N. Natalia, and C. Y. Hsieh, "Reconceptualizing value creation: Exploring the role of goal congruence in the co-creation process," *Journal of Retailing and Consumer Services*, vol. 66, p. 102947, May 2022, doi: 10.1016/j.jretconser.2022.102947.

[14]   T. Carsault, J. Nika, P. Esling, and G. Assayag, "Combining real-time extraction and prediction of musical chord progressions," *Electronics*, vol. 10, no. 21, p. 2634, Oct. 2021, doi: 10.3390/electronics10212634.

[15]   M. A. Hassan, Z. Shukur, and M. K. Hasan, "An efficient secure electronic payment system for e-commerce," *Computers*, vol. 9, no. 3, p. 66, Aug. 2020, doi: 10.3390/computers9030066.

[16]   A. Karalko et al., "In vivo contrast imaging of rat heart with carbon dioxide foam," *Sensors*, vol. 22, no. 14, p. 5124, Jul. 2022, doi: 10.3390/s22145124.

[17]   Y. Luo, S. Duan, and X. Xu, "FPGA-PRO: A defense framework against crosstalk-induced secret leakage in FPGA," *ACM Transactions on Design Automation of Electronic Systems*, vol. 27, no. 3, pp. 1–31, May 2022, doi: 10.1145/3491214.

[18]   Y. Wang et al., "Computer prediction of seawater sensor parameters in the Central Arctic region based on hybrid machine-learning algorithms," *IEEE Access*, vol. 8, pp. 213783–213798, 2020, doi: 10.1109/ACCESS.2020.3038570.

[19]   K. Chattrairat, W. Wongseree, and A. Leelasantitham, "Comparisons of machine-learning methods for statistical downscaling," *Journal of Web Engineering*, vol. 20, no. 5, pp. 1–12, Jul. 2021, doi: 10.13052/jwe1540-9589.2057.

[20]   Z. Zhang and L. Dai, "Effects of synaptic pruning on phase synchronization in chimera states of neural networks," *Applied Sciences*, vol. 12, no. 4, p. 1942, Feb. 2022, doi: 10.3390/app12041942.

[21]   A. Pradhan et al., "Biosensors as nano-analytical tools for COVID-19 detection," *Sensors*, vol. 21, no. 23, p. 7823, Nov. 2021, doi: 10.3390/s21237823.

[22]   J. Qin, L. Liu, H. Shen, and D. Hu, "Uniform pooling for graph networks," *Applied Sciences*, vol. 10, no. 18, p. 6287, Sep. 2020, doi: 10.3390/app10186287.

[23]   C. Wiecher et al., "Model-based analysis and specification of functional requirements and tests for complex automotive systems," *Systems Engineering*, vol. 27, no. 4, pp. 728–744, Jul. 2024, doi: 10.1002/sys.21748.

[24] K. M. Habibullah, G. Gay, and J. Horkoff, "Non-functional requirements for machine learning," *Requirements Engineering*, vol. 28, no. 2, pp. 283–316, Jun. 2023, doi: 10.1007/s00766-022-00395-3.

[25] Q. Li and Z. Song, "Ensemble-learning-based prediction of steel bridge deck defect condition," *Applied Sciences*, vol. 12, no. 11, p. 5442, May 2022, doi: 10.3390/app12115442.

[26] G. A. Stelea, L. Sangeorzan, and N. Enache-David, "When cybersecurity meets accessibility: A holistic development architecture for inclusive cyber-secure web applications," *Future Internet*, vol. 17, no. 2, p. 67, Feb. 2025, doi: 10.3390/fi17020067.