Journal Collabits
Vol. 3 No. 2 May 2026 : 151-169
E-ISSN : 3046-6709, P-ISSN : 3062-8601
https://publikasi.mercubuana.ac.id/index.php/collabits

Online Food Ordering App

Rafif Syari Hidayah[1*], Rehan Saputra[2], Krisna Septiawan[3], Muhammad Aditya Ramadhan[4], Misni[5]
[1,2,3,4,5] Information System, Universitas Mercu Buana, Indonesia
*Corresponding Author: rafanime3@gmail.com

Abstract - Online food ordering systems have become an important component of digital commerce because users increasingly expect fast, convenient, and reliable transaction processes. However, system development should not focus only on functional ordering features; it should also consider usability, transaction security, and the ability to transform transaction records into information for future analytical use. This study aims to design and develop a web-based Online Food Ordering Application using the Software Development Life Cycle (SDLC) Waterfall approach. User and seller requirements are translated into major functions including registration and authentication, food and beverage browsing, shopping cart management, checkout, payment, order history, product management, and order confirmation. A conceptual framework is introduced to explain the relationships among user needs, system design, usability, security, transaction efficiency, and the analytical utilization of transaction data. The application was implemented using PHP with HTML and CSS for the interface and a database for operational records. Functional validation was conducted using black-box testing on the documented login, menu, and payment scenarios. All documented scenarios produced the expected outputs and were categorized as passed. The results show that the prototype supports the basic food ordering workflow. Nevertheless, the present evaluation is limited to functional testing and does not yet provide quantitative usability measurements, comprehensive security testing, transaction-time benchmarking, or experimental evaluation of recommender and Business Intelligence modules. The stored transaction and order-history data are therefore positioned as a foundation for future behavioral analytics, Business Intelligence dashboards, and personalized food recommendation development.

Keywords:
Online Food Ordering;
User Experience;
Transaction Security;
Business Intelligence;
Recommender System;

Article History:
Received: 10-03-2025
Revised: 24-04-2026
Accepted: 02-05-2026

Article DOI: 10.22441/collabits.v3i2.31236

INTRODUCTION

The rapid growth of digital technology has changed the way consumers obtain products and services, including how food is selected, ordered, paid for, and delivered. Online food ordering platforms reduce the need for direct interaction with sellers by allowing consumers to browse menus, place orders, and manage transaction information through a digital interface. Earlier work on online food delivery indicates that perceived ease of use, time-saving orientation, convenience, privacy, and security are important considerations in users' behavioral intention to use such services [1]. More recent reviews show that online food delivery research has expanded from simple website ordering toward mobile applications, online-to-offline platforms, service quality, customer behavior, and data-driven platform management [6].

From the software-development perspective, web-based food ordering systems commonly provide authentication, menu management, shopping-cart functions, checkout, payment, and transaction history. Afiksih [2] demonstrated that a web-based canteen ordering application could simplify menu access and reduce conventional queue-based ordering, while Inayati et al. [3] emphasized the role of web-based ordering in supporting ordering and promotion activities. Aldisa [4] further showed how website-based food promotion and

ordering can be developed through Rapid Application Development, whereas Huan et al. [5] employed the Waterfall model to develop a web-based food ordering system with structured analysis, design, implementation, and testing phases.

Although these studies provide useful foundations, a gap remains between conventional functional system development and the broader requirements of contemporary digital business platforms. Many food-ordering development studies primarily demonstrate whether functions such as login, menu selection, checkout, and payment can operate. The relationships among user requirements, system design, usability, transaction security, and transaction efficiency are not always made explicit. In addition, transaction data are frequently treated only as historical records, even though they can become a valuable source for behavioral analytics, Business Intelligence, and recommendation systems.

The user experience dimension is particularly important because the quality of the digital interface can influence satisfaction and subsequent behavior. Lin, Au, and Baum [7] found that the service quality of online food delivery mobile applications has spillover effects on application satisfaction, food satisfaction, and repurchase intention. This evidence indicates that the ordering interface should be designed not merely as a collection of functions but as an integrated interaction flow that helps users complete tasks clearly and efficiently.

Another development opportunity lies in the use of transaction data for personalization. A recent systematic review of food recommender systems shows that food recommendation research uses content-based filtering, machine-learning approaches, user information, food attributes, and interaction data to generate recommendations [8]. Zhang et al. [9] also demonstrate that dynamic user behavior can be incorporated into sequence-based personalized food recommendation. At the platform level, Wang, Tao, and Zhang [10] show that a multi-objective hierarchical recommender deployed in a large food-delivery marketplace can support multiple business objectives. These developments suggest that even a basic ordering application should be designed with structured transaction records that can later support intelligent analytics.

Transaction security is equally important because online ordering involves user credentials, payment-related information, and transaction records. Research on mobile payment has highlighted the continuing relevance of privacy, stakeholder concerns, and post-adoption behavior [11], while a systematic review of financial technology adoption identifies trust and security as central factors in digital financial services [12]. Therefore, authentication, input handling, authorization, session management, and payment validation should be considered part of a secure transaction architecture rather than optional technical additions.

Based on these issues, the research problem addressed in this study is how a web-based online food ordering system can translate user and seller needs into an integrated design that supports usability, security, and transaction efficiency while generating structured data that can become a foundation for Business Intelligence and personalized recommendation development.

The study addresses the following research questions:

- RQ1. How can user and seller needs be translated into the design of a web-based food ordering system that supports usability, security, and transaction efficiency?
- RQ2. How can transaction and order-history data generated by the system provide a foundation for future Business Intelligence, behavioral analytics, and food recommendation functions?

Accordingly, the objectives of this study are to design and implement a web-based online food ordering application based on the identified functional requirements, evaluate the documented major functions using black-box testing, and formulate a conceptual framework

connecting user needs, system design, usability, security, transaction efficiency, and analytical data utilization. The contribution of this study is not the proposal of a new recommendation algorithm. Rather, it extends a conventional software-development project by positioning operational transaction data as an analytical asset that can support future Business Intelligence and recommender-system development.

LITERATURE REVIEW

Online Food Ordering Systems

Online food ordering systems integrate customer access, menu information, order processing, payment-related activities, and transaction records within a digital environment. The system developed by Afiksih [2], for example, used PHP and MySQL to support a web-based canteen ordering process, while Huan et al. [5] used a Waterfall development model to structure analysis, design, implementation, and testing. These studies illustrate that the operational foundation of an online ordering system is formed by clearly defined actors, database entities, transaction workflows, and user interfaces.

A broader systematic synthesis by Shankar et al. [6] shows that online food delivery research spans website, mobile application, online-to-offline, and other technology contexts. The literature also uses established behavioral theories such as the Technology Acceptance Model, Theory of Planned Behavior, and UTAUT to explain user adoption. This development indicates that online food ordering should be considered simultaneously as a software system, a user interaction environment, and a digital business platform.

User Needs, Usability, and User Experience

User needs form the starting point of requirements analysis. For an online food ordering application, users generally require understandable navigation, accurate menu information, a manageable shopping cart, a clear checkout process, payment input, and access to historical transactions. Sellers require product management, order monitoring, order confirmation, and status updating. Translating these needs into interface and workflow design is essential because technical functionality alone does not guarantee an acceptable user experience.

Lin et al. [7] examined online food delivery mobile application service quality and demonstrated that application service-quality dimensions can influence application satisfaction and subsequent repurchase intention. In the present study, usability is therefore treated as an explicit design consideration. However, because the source project documents only functional black-box testing, the current study does not claim a statistically measured usability score. Quantitative usability evaluation using instruments such as the System Usability Scale or User Experience Questionnaire is proposed for future work.

Transaction Security and Digital Payment

Security is an important design dimension because the system stores user credentials and supports payment-related activities. Behera and Kumra [11] show that mobile-payment research increasingly considers different stakeholders, privacy concerns, and post-adoption behavior. Jafri et al. [12] further identify trust and security as important factors in the adoption of digital financial technology. These findings are relevant to food ordering systems because payment and authentication processes directly affect the reliability of the transaction flow.

In this study, security is represented at the functional-design level through authentication and payment validation. Nevertheless, a secure production implementation requires stronger controls than functional validation alone. Recommended controls include password hashing, parameterized database queries, server-side input validation, role-based authorization, secure

session management, HTTPS communication, protection against cross-site request forgery, and vulnerability testing. These controls are identified as development requirements and future evaluation areas; the current black-box results do not constitute comprehensive cybersecurity evidence.

Recommender Systems and Behavioral Analytics

Recommender systems use available user, item, and interaction information to reduce information overload and provide more relevant choices. Bondevik et al. [8] reviewed food recommender systems and found that content-based filtering and machine-learning approaches are widely used, while data availability and evaluation methods remain important methodological considerations. Zhang et al. [9] show that sequence-based recommendation can capture dynamic user behavior when user-item interactions change over time.

For online food ordering applications, historical transactions can provide a basic interaction dataset. Repeated orders can reveal menu preferences, transaction frequency, combinations of items, and temporal patterns. Wang et al. [10] demonstrate the strategic importance of recommender systems in a multi-sided food-delivery marketplace by integrating multiple platform objectives. The present application does not yet implement a recommender algorithm, but its transaction-history function provides a starting point for the future construction of user-item interaction data.

Business Intelligence for Digital Food Ordering

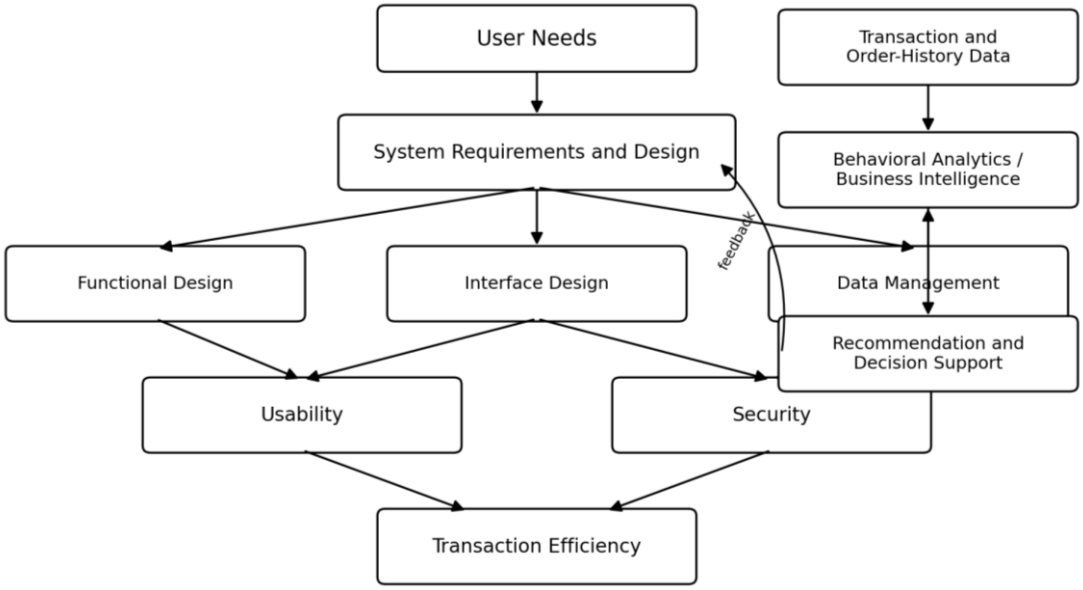
Business Intelligence converts structured operational data into information that can support monitoring and decision making. In an online food ordering context, transaction data can be aggregated into indicators such as frequently ordered menu items, transaction volume by period, repeat-order frequency, payment-method distribution, and order-status patterns. These indicators can assist sellers in menu planning, promotional decisions, customer engagement, and operational monitoring.

The analytical component proposed in this study is therefore a Business Intelligence-ready extension rather than a fully implemented dashboard. The current system stores the operational records needed for ordering and order history. A subsequent development phase can extract, transform, and aggregate those records to build dashboards and recommendation models. Making this distinction prevents the study from claiming analytical performance that has not yet been experimentally evaluated.

Conceptual Framework

The conceptual framework explains how user needs are translated into system requirements and how the resulting design relates to usability, security, transaction efficiency, and analytical data utilization. User needs guide the functional design, interface design, and data-management structure. These design components support usability and security, which together contribute conceptually to a more integrated transaction process. At the same time, transaction and order-history data can be processed through behavioral analytics and Business Intelligence to support recommendations and decision making. Analytical insights can subsequently provide feedback for future system-design improvements.

Figure 1. Conceptual Framework of the Online Food Ordering System



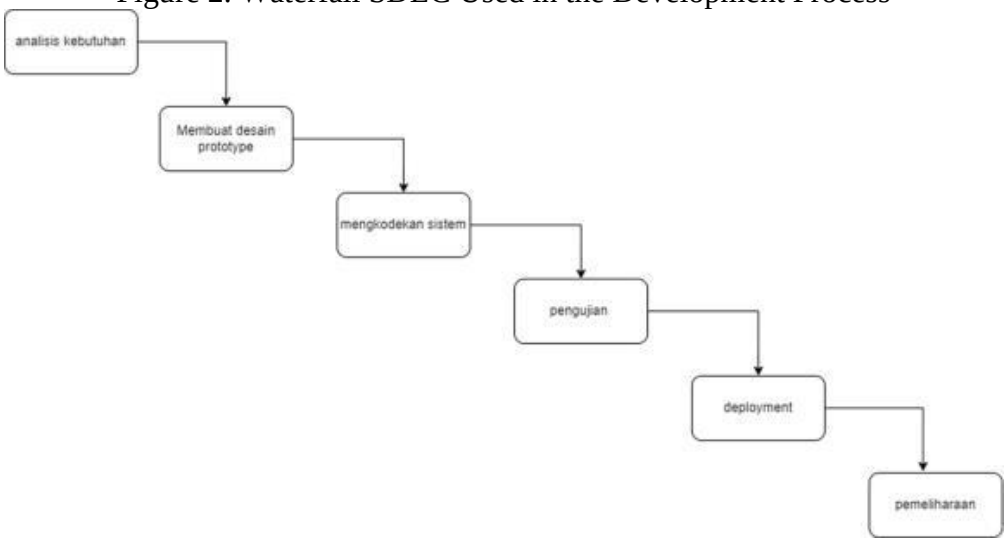
The arrows in Figure 1 represent conceptual design relationships rather than statistically tested causal effects. The current study validates documented system functions using black-box testing; empirical measurement of usability, security performance, transaction time, recommendation accuracy, and Business Intelligence effectiveness is left for future research.

METHODOLOGY

Software Development Method

The system was developed using an SDLC Waterfall approach. Based on the submitted manuscript, the development activities include requirements analysis, prototype or system design, coding, testing, deployment, and maintenance [5]. The sequential structure is appropriate for a project in which the primary user and seller functions are identified before implementation and represented using flowcharts and Unified Modeling Language diagrams.

Figure 2. Waterfall SDLC Used in the Development Process

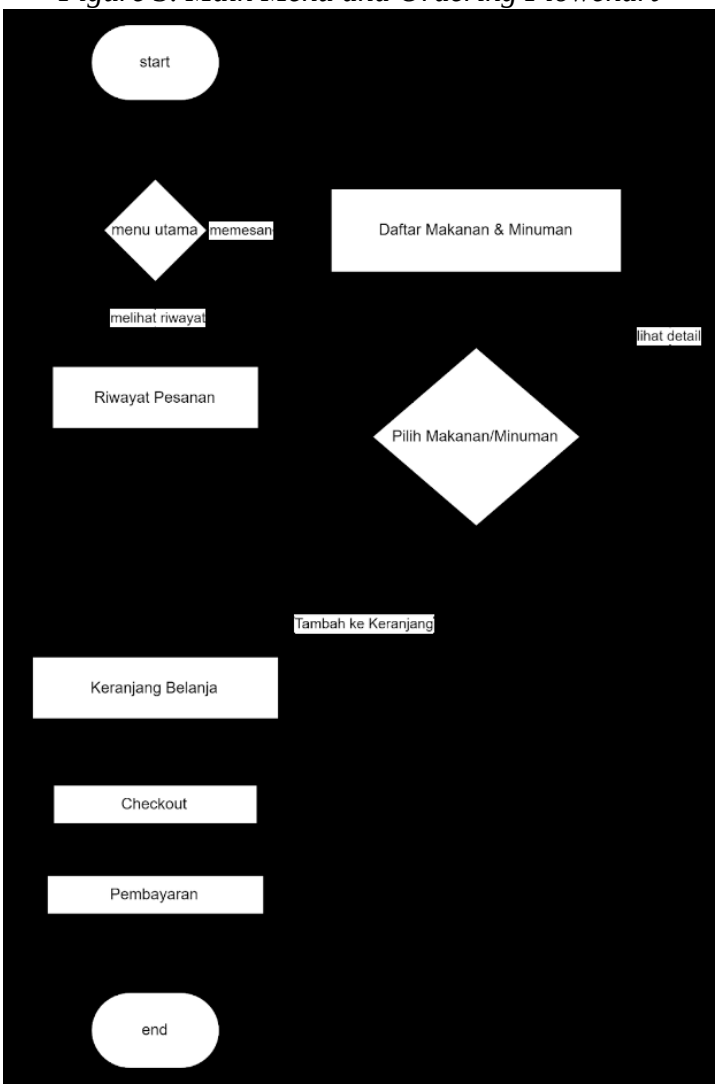


During requirements analysis, the main user functions were identified as registration, login, viewing food and drink menus, adding products to the shopping cart, checkout, payment, and order-history access. Seller functions include authentication, adding new menu items, updating or deleting existing items, viewing customer orders, confirming orders, and updating

order status. These requirements were then translated into navigation, use-case, activity, and class diagrams.

Main Transaction Flow

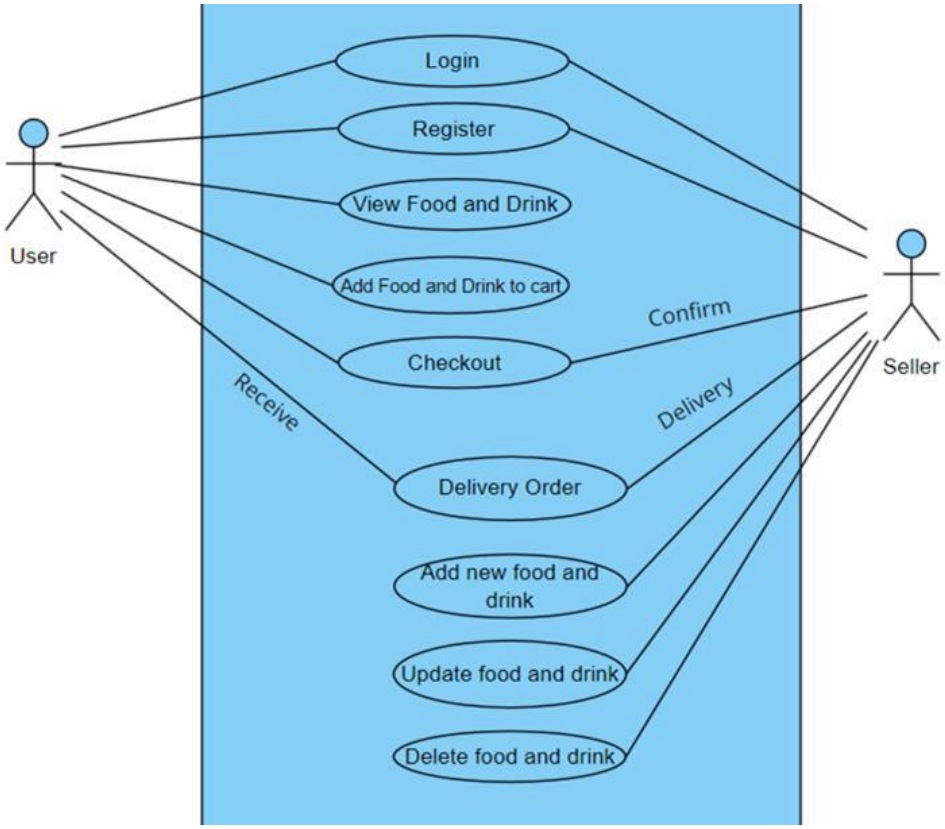
Figure 3. Main Menu and Ordering Flowchart



The main ordering flow begins after a user accesses the system. The user can view the food and beverage list, select products, add them to the shopping cart, review the cart, proceed to checkout, and enter payment information. The flowchart also includes order-history access, which enables previous transaction records to be viewed. This sequence constitutes the principal user transaction path evaluated by the system design.

Use Case Design

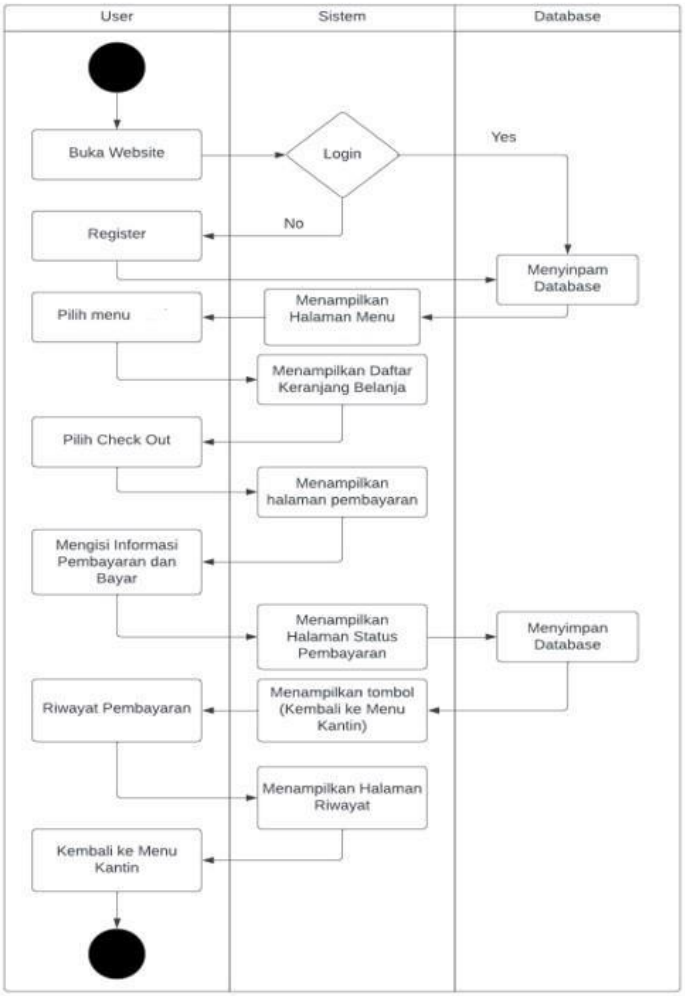
Figure 4. Use Case Diagram



The use-case model contains two main actors: User and Seller. The User can register or log in, view food and beverage items, add items to the cart, and continue to checkout. The Seller can register or log in, manage food and beverage records, confirm user orders, and process delivery-related actions. The diagram establishes role separation at the design level, which is important for both usability and authorization planning.

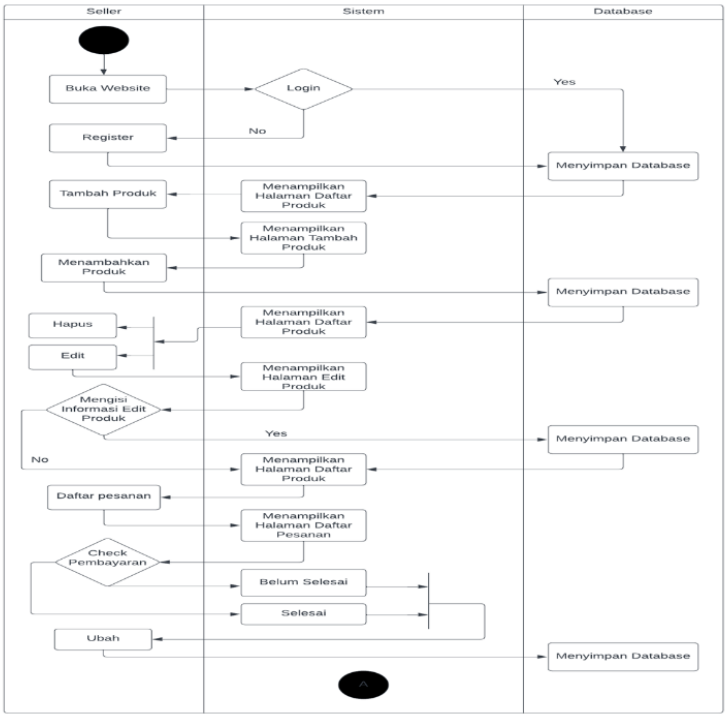
Activity Diagrams

Figure 5. User Activity Diagram



The user activity diagram describes the sequence from opening the website and authentication through menu selection, shopping-cart review, checkout, payment input, payment-status display, and transaction-history access. The system and database swimlanes illustrate that user actions depend on application processing and data storage rather than occurring as isolated interface events.

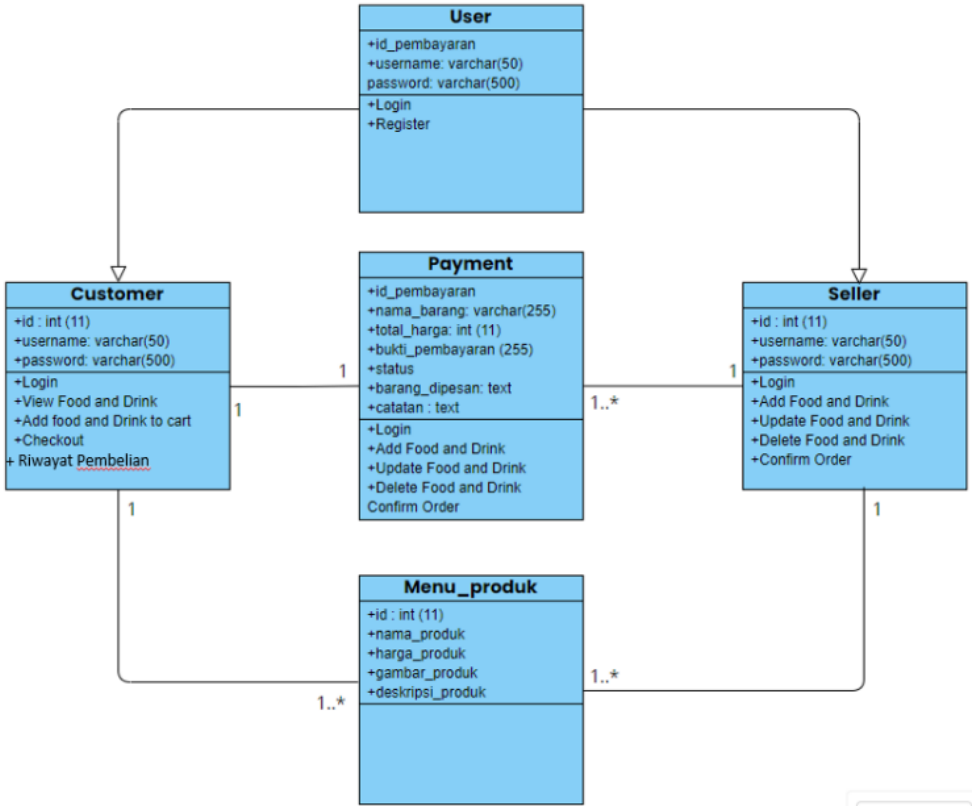
Figure 6. Seller Activity Diagram



The seller activity diagram represents seller authentication, product addition and editing, product list access, order monitoring, order confirmation, and status updates. This flow supports the operational side of the ordering system by connecting menu administration and incoming customer transactions.

Class and Data Structure

Figure 7. Class Diagram



The class diagram in the submitted system includes the User, Customer, Seller, Payment, and Menu_product classes. The structure shows that authentication and user roles are connected with product and payment-related data. Although the diagram is relatively simple, it provides the data foundation for storing user, menu, and transaction information. The same records can later be transformed into analytical datasets for product-performance monitoring and recommendation development.

Testing Method

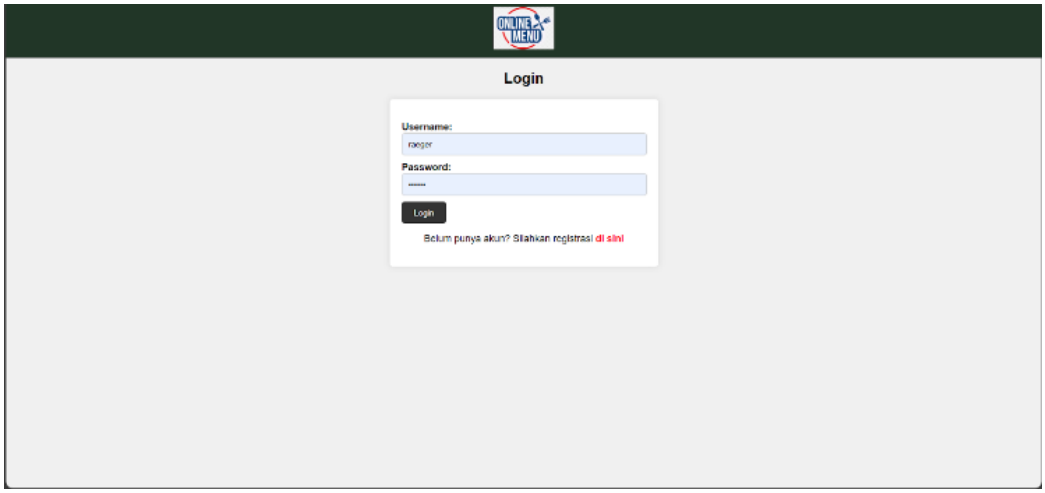
Functional verification uses black-box testing, which evaluates whether a function produces the expected output from a user-facing perspective without requiring the tester to analyze the internal code structure. The documented test table in the submitted manuscript covers login, menu-page access, and payment input. Therefore, the findings reported in this revision are limited to those documented scenarios and do not extend the evidence beyond the original test results.

RESULTS AND DISCUSSION

User Interface Implementation

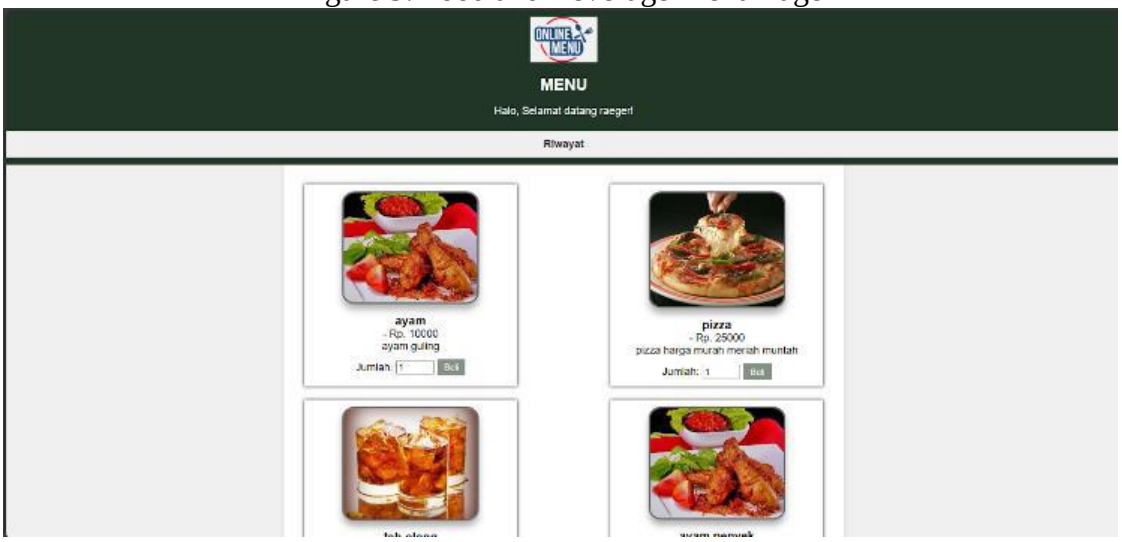
The application was implemented as a web-based system using PHP, with HTML and CSS used for the user interface. The submitted manuscript documents login, menu, order-history, payment, seller product-list, and order-list pages. These interfaces demonstrate the translation of the design diagrams into operational screens.

Figure 8. Login Page



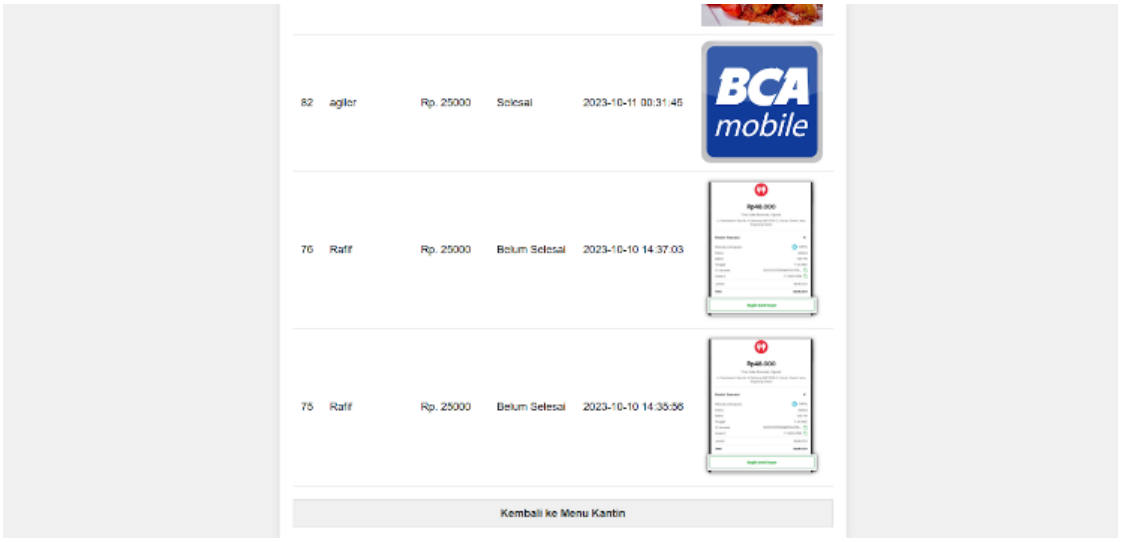
The login page provides the authentication entry point for existing users. Authentication is necessary before access to protected user functions. From a security perspective, this interface should be supported by secure password storage and server-side validation in a production deployment.

Figure 9. Food and Beverage Menu Page



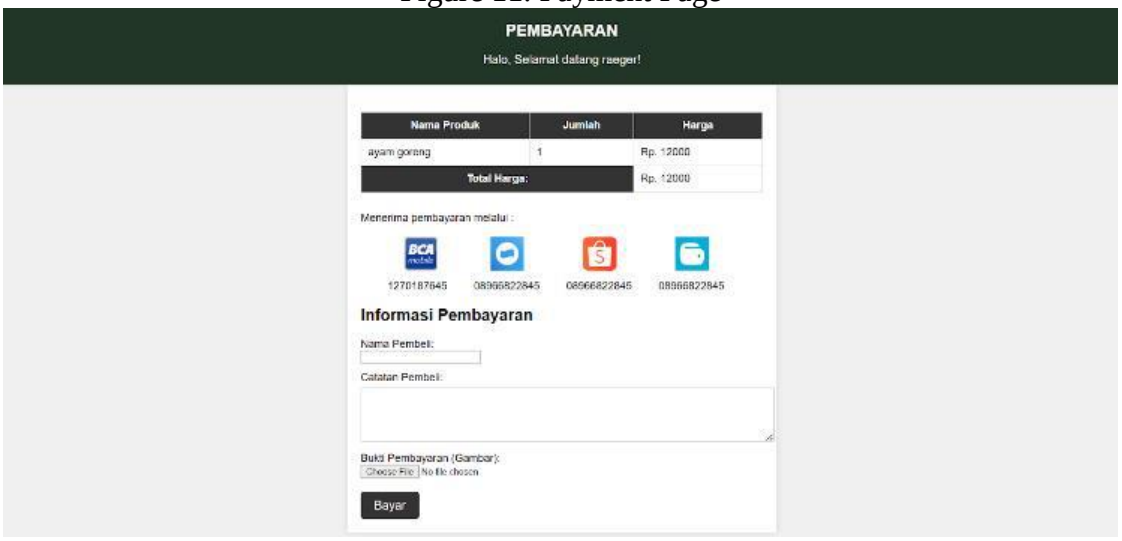
The menu page presents available food and beverage products with visual and price information. This interface responds directly to the user requirement for clear product access and provides the starting point for product selection and cart creation.

Figure 10. Order History Page



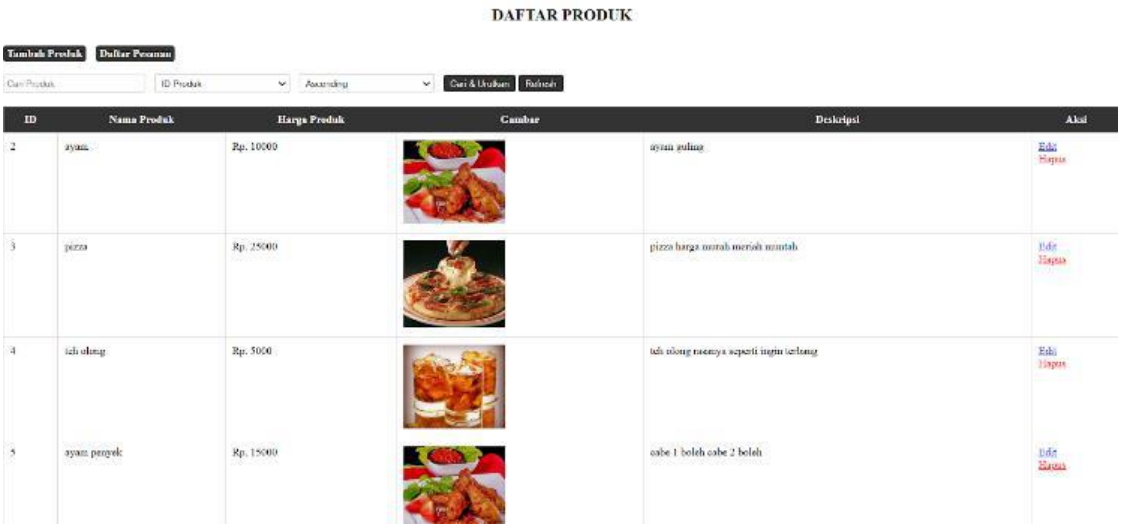
The order-history page records previous transactions and their status-related information. In the current system, this function supports transaction traceability. From an analytical perspective, the same records can later become a source for repeat-order analysis, product-frequency analysis, and personalized recommendations.

Figure 11. Payment Page



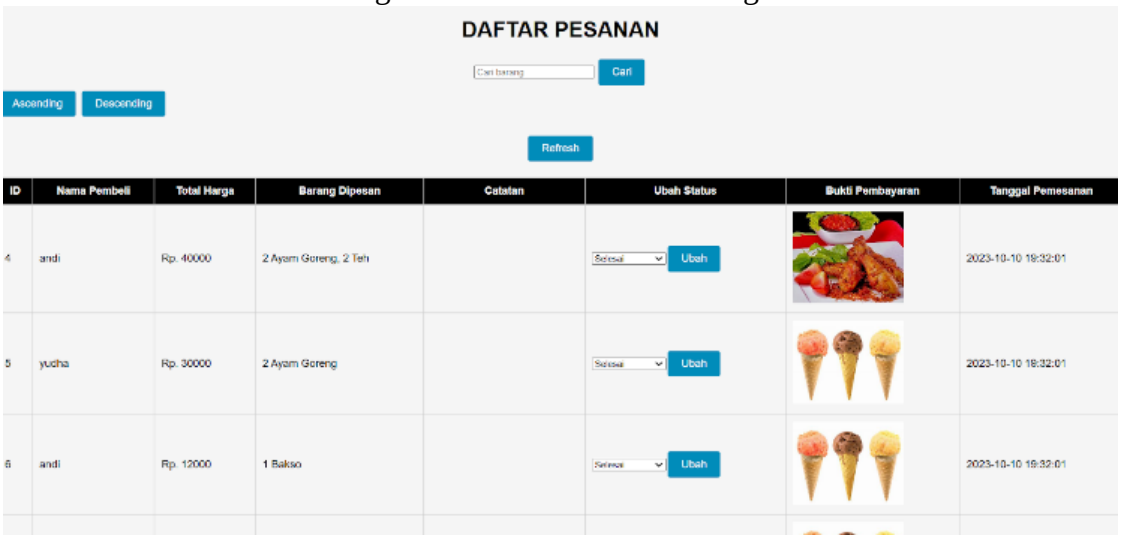
The payment page displays order information and provides fields required to continue the payment process. The original manuscript indicates that the system validates payment before the order is processed. However, the present test evidence is functional rather than security-oriented; therefore, no claim is made that the payment mechanism has undergone comprehensive security certification or penetration testing.

Figure 12. Seller Product List Page



The seller product-list page allows menu information to be managed. This supports data quality because menu availability, price, description, and related product information must be maintained by the seller before they can be presented to users.

Figure 13. Seller Order List Page



The seller order-list page supports operational monitoring of incoming transactions and status updating. Integrating order monitoring with product management connects customer-facing ordering functions with seller-facing transaction processing.

Implementation Logic

The submitted manuscript also documents source-code fragments for order-list processing, login, registration, and main-menu display. These fragments demonstrate how the web interface is connected with PHP logic and database operations. In the revised discussion, the code figures are treated as implementation evidence rather than as the main research result, because the research contribution is better explained through system behavior, design relationships, and the implications of the transaction data.

Figure 14. Order List Processing Code

daftar_pesanan.php > ...

```php
<?php
session_start();

// Periksa apakah session seller_id telah diatur
if (!isset($_SESSION['seller_id'])) {
    header("Location: umbkantin_login_seller.php");
    exit;
}

$conn = mysqli_connect("localhost", "root", "", "kantin");

if (!$conn) {
    die("Connection failed: " . mysqli_connect_error());
}

// Set jumlah item per halaman
$items_per_page = 5;

// Tentukan halaman saat ini
$current_page = isset($_GET['page']) ? $_GET['page'] : 1;

// Pencarian
$search = isset($_GET['search']) ? $_GET['search'] : '';

// Urutan tabel (default: ascending)
$order_by = isset($_GET['order_by']) ? $_GET['order_by'] : 'ASC';

// Tombol Refresh
if (isset($_POST['refresh'])) {
    header("Location: daftar_pesanan.php");
    exit;
}

// Query untuk mengambil data pesanan dengan pembagian halaman
$start_from = ($current_page - 1) * $items_per_page;

// Query untuk menghitung total item yang cocok dengan kriteria pencarian
$count_query = "SELECT COUNT(*) AS total_items FROM pembayaran
                WHERE nama_barang LIKE '%$search%'
```

Figure 15. Login Processing Code

login.php > ...

```php
<?php
session_start();
if(isset($_SESSION['login'])){
    header('Location: index.php');
    exit;
}
require 'functions.php';

if(isset($_POST['login'])){
    $username = $_POST['username'];
    $password = $_POST['password'];

    $result = mysqli_query($conn, "SELECT * FROM user WHERE username = '$username'");
    if(mysqli_num_rows($result) == 1){
        $_SESSION['login'] = true;

        $row = mysqli_fetch_assoc($result);
        if(password_verify($password, $row['password'])){
            $_SESSION['username'] = $username; // Menyimpan username ke dalam sesi
            header("Location: index.php");
            exit;
        }
    }
    $error = true;
}

?>
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Kantin Halaman Login</title>
    <style>
        /* CSS */
        body {
            font-family: Arial, sans-serif;
            background-color: #f0f0f0;
            margin: 0;
            padding: 0;
```

Figure 16. Registration Processing Code

registrasi.php > ...

```php
<?php
require 'functions.php';

if(isset($_POST['register'])){

    if(registrasi($_POST) > 0){

            echo "
            <script>
                alert('user berhasil ditambahkan');
            </script>
        ";
    } else {
       echo mysqli_error($conn);
    }
    }
?>

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Halaman Registrasi</title>
    <style>

        body {
            font-family: Arial, sans-serif;
            background-color: #f0f0f0;
            margin: 0;
            padding: 0;
        }

        .navbar {
            background-color: #333;
            color: white;
            padding: 10px 0;
            text-align: center;
        }

```

Figure 17. Main Menu Processing Code

index.php > ...

```php
<html lang="en">
<body>
<header>
        <p>Halo, Selamat datang <?php echo $_SESSION['username']; ?>!</p>
    <?php endif; ?>
    <a href="riwayat_pembayaran.php">Riwayat</a>
</header>

    <div class="container">
        <div class="product-container">
            <?php foreach ($menu_produk as $produk) { ?>
                <div class="product">
                    <img src="img/<?php echo $produk['gambar_produk']; ?>" alt="<?php echo $produk['nama_produk']; ?>" width="200">
                    <hr>
                    <strong><?php echo $produk['nama_produk']; ?></strong> - Rp. <?php echo $produk['harga_produk']; ?>
                    <br>
                    <?php echo $produk['deskripsi_produk']; ?>
                    <hr>
                    <form action="index.php?action=add&barang=<?php echo $produk['id_produk']; ?>&nama=<?php echo $produk['nama_produk']; ?>&harga=<?php echo $
                        Jumlah: <input type="number" name="jumlah" value="1" min="1">
                        <input type="submit" value="Beli">
                    </form>
                </div>
            <?php } ?>
    </div>
    <h2>Keranjang Belanja</h2>
    <form action="pembayaran.php" method="post">
        <table>
            <tr>
                <th>Nama Barang</th>
                <th>Jumlah</th>
                <th>Harga Persatu Barang</th>
                <th>Kurangi</th>
            </tr>
            <?php
            if (isset($_SESSION['keranjang'])) {
                foreach ($_SESSION['keranjang'] as $id_produk => $barang) {
                    echo "<tr>";
                    echo "<td>{$barang['nama']}</td>";
                    echo "<td>{$barang['jumlah']}</td>";
```

Ln 1, Col 1 Spaces: 4 UTF-8 CRLF

Black-Box Testing Results

Black-box testing was used to determine whether the documented functions generated the expected outputs. Table 1 reproduces the functional evidence available in the submitted manuscript while standardizing the wording for clarity.

Table 1. Black-Box Testing Results for the Documented User Functions

Tested Feature	Testing Scenario	Expected / Observed Result	Conclusion
Login	Enter a registered username and password	A successful login message is displayed and the user is redirected to the menu page.	Passed
Menu Page	Access the menu after authentication	Food and beverage menu data and order-history access are displayed.	Passed
Payment	Click the payment button and continue to payment input	The system provides the required payment data-entry fields.	Passed

All three documented scenarios were categorized as passed. These results demonstrate that the tested functions behave according to their expected functional flow. Nevertheless, functional success must be interpreted within the scope of the test. The current evidence does not measure how quickly users complete tasks, how users perceive the interface, whether the system resists security attacks, or whether analytical and recommendation modules improve decision quality.

User Needs, Usability, and Transaction Efficiency

The implemented functions can be mapped directly to the user needs identified during requirements analysis. Menu browsing addresses the need to identify available products; the shopping cart allows selected products to be reviewed; checkout organizes the transition from selection to transaction; payment supports transaction completion; and order history provides traceability. Seller product management and order monitoring support the operational requirements on the supply side of the system.

This mapping supports the first research question by showing how user and seller needs are translated into a concrete transaction architecture. The sequence from authentication to menu selection, cart, checkout, and payment reduces fragmentation by placing major activities within one web-based workflow. However, the manuscript does not contain task-completion-time measurements or standardized usability questionnaires. Transaction efficiency and usability are therefore interpreted as design objectives supported by workflow integration, not as statistically proven performance advantages.

Future empirical testing can strengthen this relationship. Usability can be measured using the System Usability Scale or User Experience Questionnaire, while transaction efficiency can be operationalized through task-completion time, number of interaction steps, transaction-success rate, and error frequency. Such measurements would allow the conceptual relationships in Figure 1 to be evaluated quantitatively rather than only discussed as design logic.

Security Implications

Authentication and payment are the two system areas most directly associated with security. The current implementation demonstrates that users can log in and reach the payment input process, but black-box functional testing does not verify confidentiality, integrity, access control, or resistance to malicious input. The security literature reviewed in [11] and [12] shows why trust, privacy, and security must be treated as central elements of digital transaction systems.

For future development, the login and registration modules should use secure password hashing and parameterized queries, while protected pages should enforce role-based authorization and secure session handling. Payment-related communication should use encrypted transport, server-side validation, and appropriate integration with a trusted payment

gateway. Vulnerability scanning and penetration testing should then be conducted separately from functional black-box testing. This security roadmap aligns the existing application with the reviewer's request without overstating the security evidence currently available.

Business Intelligence and Behavioral Analytics Implications

The second research question concerns how the generated data can support future Business Intelligence and recommendation functions. The most important existing asset is the order-history data. Each completed or recorded transaction can potentially be transformed into an analytical record containing user, product, quantity, value, payment, status, and time attributes, depending on the fields implemented in the database. Aggregating these records can produce descriptive indicators such as popular products, transaction volume, repeat-order frequency, product demand by period, and payment-method use.

These indicators would allow a seller to move from passive transaction storage to data-driven monitoring. A simple Business Intelligence dashboard could present total orders, total sales value if transaction amounts are stored reliably, best-selling products, orders by status, and repeat-purchase patterns. Because the current study does not implement or test such a dashboard, these indicators are presented as a development pathway rather than as completed results.

Behavioral analytics can also prepare data for a recommender system. A user-item matrix can be constructed from order history, where interactions represent purchases or order frequency. Content-based techniques can incorporate menu attributes, collaborative approaches can use patterns among users, and hybrid models can combine both sources [8]. Sequence-based approaches can further consider the order in which user preferences change over time [9]. For a larger marketplace, multi-objective recommendation can balance consumer relevance with broader platform goals [10]. These possibilities demonstrate how the existing system architecture can be extended toward intelligent digital business functionality that is more closely aligned with the scope of Journal Collabits.

Research Contribution and Limitations

The revised study makes three main contributions. First, it restructures a basic food-ordering development project around an explicit problem statement and two research questions. Second, it introduces a conceptual framework connecting user needs, system design, usability, security, and transaction efficiency. Third, it interprets transaction and order-history data as a foundation for Business Intelligence, behavioral analytics, and future recommendation functions rather than treating the data only as historical records.

The limitations are equally important. Only the functional scenarios documented in the submitted manuscript are used as empirical test evidence. No standardized usability survey, load test, task-time comparison, vulnerability assessment, penetration test, recommendation experiment, or Business Intelligence dashboard evaluation was available in the source manuscript. Accordingly, the discussion distinguishes existing results from proposed future extensions and avoids claiming effectiveness beyond the available evidence.

CONCLUSION

This study designed and developed a web-based Online Food Ordering Application using an SDLC Waterfall approach. The system translates user and seller requirements into integrated functions that include registration and authentication, menu browsing, shopping-cart management, checkout, payment, order history, product management, and order monitoring. The design is documented through a main-menu flowchart, use-case diagram,

user and seller activity diagrams, a class diagram, interface screenshots, and implementation-code examples.

The documented black-box testing results show that the login, menu-page, and payment scenarios produced the expected functional outputs and were categorized as passed. These results answer the first research objective at the functional level: the prototype can support the basic ordering workflow that was designed. However, the results should not be interpreted as proof that the application has superior usability, complete cybersecurity, or faster transaction performance because those dimensions were not measured quantitatively in the submitted test evidence.

The conceptual framework developed in this revision clarifies how user needs lead to functional, interface, and data-management design and how these elements relate to usability, security, and transaction efficiency. The framework also extends the system beyond transactional functionality by positioning transaction and order-history data as a potential analytical asset. These records can support descriptive Business Intelligence indicators, behavioral analytics, and the future development of personalized food recommendations.

The study therefore contributes an analytics-ready perspective to a conventional web-based ordering application. Future work should measure usability using validated instruments, evaluate transaction efficiency through task-time and success indicators, conduct vulnerability and penetration testing, implement a Business Intelligence dashboard, and compare recommender-system approaches using historical user-item interactions. These extensions would provide stronger empirical evidence regarding the usability, security, analytical capability, and intelligent decision-support value of the Online Food Ordering Application.

REFERENCES

[1] T. C. Lau and D. N. C. Yat, “Online Food Delivery Services: Making Food Delivery the New Normal,” Journal of Marketing Advances and Practices, vol. 1, no. 1, 2019.

[2] M. Afiksih, “Perancangan Aplikasi Pemesanan Makanan Berbasis Web di Kantin PT. Pegadaian Kanwil I Medan,” Journal of Computer Science and Informatics Engineering (CoSIE), vol. 1, no. 2, pp. 66–77, 2022, doi: 10.55537/cosie.v1i2.61.

[3] I. Inayati, M. N. Hidayatulloh, and M. Kamisutara, “Aplikasi Pemesanan Makanan Berbasis Web (Studi Kasus: RM Lesehan Berkah Ilaahi Gresik),” e-Jurnal NARODROID, vol. 1, no. 2, 2015.

[4] R. T. Aldisa, “Penerapan Metode RAD (Rapid Application Development) Pada Sistem Informasi Promosi dan Pemesanan Makanan Berbasis Website Studi Kasus Restoran Waroenk Anak Kuliahan,” Building of Informatics, Technology and Science (BITS), vol. 3, no. 3, pp. 446–452, 2021, doi: 10.47065/bits.v3i3.1137.

[5] K. J. Huan, H. Mahdin, and M. S. Arshad, “Design and Development of Food Ordering System on Web Based Technology,” Applied Information Technology and Computer Science, vol. 4, no. 1, pp. 446–465, 2023, doi: 10.30880/aitcs.2023.04.01.026.

[6] A. Shankar, C. Jebarajakirthy, P. Nayal, H. I. Maseeh, A. Kumar, and A. Sivapalan, “Online food delivery: A systematic synthesis of literature and a framework development,” International Journal of Hospitality Management, vol. 104, art. 103240, 2022, doi: 10.1016/j.ijhm.2022.103240.

[7] P. M. C. Lin, W. C. W. Au, and T. Baum, “Service quality of online food delivery mobile application: an examination of the spillover effects of mobile app satisfaction,” International Journal of Contemporary Hospitality Management, vol. 36, no. 3, pp. 906–926, 2024, doi: 10.1108/IJCHM-09-2022-1103.

[8] J. N. Bondevik, K. E. Bennin, Ö. Babur, and C. Ersch, "A systematic review on food recommender systems," Expert Systems with Applications, vol. 238, art. 122166, 2024, doi: 10.1016/j.eswa.2023.122166.
[9] J. Zhang, Z. Wang, W. Liu, X. Liu, and Q. Zheng, "A unified approach to designing sequence-based personalized food recommendation systems: tackling dynamic user behaviors," International Journal of Machine Learning and Cybernetics, vol. 14, pp. 2903–2912, 2023, doi: 10.1007/s13042-023-01808-7.
[10] Y. Wang, L. Tao, and X. X. Zhang, "Recommending for a Multi-Sided Marketplace: A Multi-Objective Hierarchical Approach," Marketing Science, vol. 44, no. 1, pp. 1–29, 2025, doi: 10.1287/mksc.2022.0238.
[11] C. K. Behera and R. Kumra, "Two decades of mobile payment research: A systematic review using the TCCM approach," International Journal of Consumer Studies, vol. 48, no. 1, art. e13003, 2024, doi: 10.1111/ijcs.13003.
[12] J. A. Jafri, S. I. Mohd Amin, A. Abdul Rahman, and S. Mohd Nor, "A systematic literature review of the role of trust and security on Fintech adoption in banking," Heliyon, vol. 10, no. 1, art. e22980, 2024, doi: 10.1016/j.heliyon.2023.e22980.