

Development of Web-Based Orcakes Patisserie Online Ordering and Class Registration System

Puteri Nur Fathimah Radzin¹, Firkhan Ali Hamid Ali^{1*}

¹ *Fakulti Sains Komputer dan Teknologi Maklumat,
Universiti Tun Hussein Onn Malaysia, Parit Raja, Batu Pahat, 86400, MALAYSIA*

*Corresponding Author: firkhan@uthm.edu.my
DOI: <https://doi.org/10.30880/aitcs.2026.07.01.028>

Article Info

Received: 13 June 2025
Accepted: 18 June 2026
Available online: 30 June 2026

Keywords

Online Ordering System, Class
Registration System, Bakery
Management, User Interface Design

Abstract

The Orcakes Patisserie Online Ordering and Class Registration System is a complete web-based platform developed to streamline order management and class registration processes. To guarantee flexibility and efficiency, the system uses an object-oriented approach to divide its essential functions—secure order placing, payment processing, real-time class registration, and product browsing—into modular and reusable components. The development approach uses a prototyping model to continually align with user expectations and UML diagrams to represent system activities. Robust organizing of data is achieved with MySQL, which supports features like a comprehensive product catalog, user authentication, real-time alerts, and an easy-to-use admin interface for scheduling classes, managing inventory, and communicating with clients. The platform simplifies administrative duties, improves the client experience, and lowers mistakes by centralizing operations. Its safe and scalable architecture assures long-term dependability and usability, meeting Orcakes Patisserie's changing requirements while providing a smooth and effective user experience for both administrators and clients.

1. Introduction

As this digital transformation is happening, small and medium-sized enterprises (SMEs) get ever more pressured to adopt integrated online platforms so as to compete and keep on top of evolving customer expectations [1]. Orcakes Patisserie is a home bakery in Malaysia, which, however, employs mainly non-integrated manual methods; that means orders and class registrations are handled through, mainly, social media and messaging platforms. Such decentralization has crippled operations with inefficiencies of miscommunication regarding orders, delays, and heavier administrative duties to name a few [2].

The fundamental research problem tackled in this study falls in the lack of a structured and unified order and enrollment management system that reduces service reliability and business scalability. The present study is going to design and implement a web-based platform for the consolidation of the order and registration functions into one streamlined platform. By allowing users to choose products, make payments, and enroll through digital interfaces, this system is to improve the accuracy of transactions, convenience to the end user, and the efficiency of operating procedures [1][3].

It is also of relevance to small bakeries planning to venture into e-commerce, since it provides a practical framework for digital adoption tailored to artisanal food businesses. A further benefit of the proposed solution is its ability to integrate online ordering with ancillary services such as a register for baking classes, serving to enhance business diversification and customer engagement within the post-pandemic scenario [3].

1.1 Problem Statement

Orcakes Patisserie relies on WhatsApp for manual order management, which is inefficient and prone to frequent errors. This traditional approach often leads to misplaced, delayed, or incorrect orders, resulting in dissatisfied customers and potential revenue loss. The lack of a streamlined and automated system for handling orders undermines the bakery's ability to deliver a reliable and efficient customer experience.

Additionally, the absence of an integrated platform for managing orders, inventory, and class registrations creates significant operational challenges. Employees are overworked with manual tasks, leading to delays and inefficiencies in daily operations. This lack of coordination hampers the bakery's ability to function smoothly and negatively impacts overall productivity and service quality.

The bakery also struggles with limited visibility of its products and classes. While social media platforms play a key role in its marketing strategy, they do not provide a well-structured or comprehensive display of available offerings. This reduces customer engagement and makes it harder for the bakery to promote special discounts, baking classes, or unique products effectively. Addressing these visibility issues is crucial for attracting and retaining clients.

1.2 Objective

The objectives of the system are to:

1. Design Orcakes Patisserie Online Ordering and Class Registration System using object-oriented approach.
2. Develop an online ordering and class registration system using web-based application.
3. Test the Orcakes Patisserie Online Ordering and Class Registration System through user acceptance testing.

1.3 Scope

The system will be used by two types of users: the administrator and the client of the Orcakes Patisserie Online Ordering and Class Registration System. The system will have 6 modules: User Registration, Profile, Class Registration, Orders, Transaction Payment and Generate Report. The details of the modules are shown in Table 1.

Table 1 Modules system for Orcakes Patisserie Online Ordering and Class Registration System

Modules	Functionalities
User Registration	By entering the required information, users will be able set up accounts on the system. After a successful registration, users are sent to the login page. The system enables logout capability, validates user credentials, and, following successful login, reroutes users to role-specific dashboards
Profile	Users may browse their profiles and change personal information like name and contact details.
Class Registration	Enable clients to register, check available class schedules, and get confirmations. Allow administrators to maintain and update class schedules, assuring users availability and validity.
Orders	Offer clients the option to browse items, add them in their shopping cart, and make safe payments. Give administrators the ability to efficiently monitor, adjust, and track client orders.

Table 2 *Modules system for Orcakes Patisserie Online Ordering and Class Registration System(cont.)*

Transaction Payment	Give clients the option to safely pay for orders and class registrations while give administrators access to the system's transaction history.
Generate Report	Give administrators the ability to create reports on user activity, sales, and inventory.

2. Literature Review

Through digital transformation, the Orcakes Patisserie Online Ordering and Class Registration System seeks to modernize bakery operations by dealing with inefficiencies in traditional methods. The research emphasizes how class registration systems and online ordering platforms improve client satisfaction, lower manual mistake rates, and expedite processes [4]. Clients can easily register for classes, place orders, and browse product catalogs and class schedules from their smartphones thanks to these platforms[5]. Implementation of these systems is difficult, though, especially for small bakeries that do not have the resources or expertise required to set them into operation.

Modern bakery management systems require technologies like cloud computing, e-commerce integration, content management systems (CMS), database management systems (DBMS), web-based platforms, and user-friendly interfaces. Web-based solutions that are based on frameworks such as Vue.js or React guarantee responsiveness and accessibility. While CMS technologies enable non-technical users to handle changes, e-commerce features offer smooth transactions and tailored suggestions. Cloud integration enables safe, remote administration, while DBMS options like MySQL guarantee data dependability. Customer happiness and usability are both increased by well-designed user interfaces.

As shown by Kenny Hills Bakers, Magnolia Bakery, and Hogan Bakery, the shortcomings of current bakery systems highlight the importance of an integrated solution. These bakeries excel in online ordering and user interface design, but they lack sophisticated analytics, inventory management tools, and class registration systems. In order to satisfy changing client and company demands, Orcakes Patisserie has the opportunity to create a comprehensive system that incorporates online purchasing, class registration, and administrative features.

2.1 Comparison with the Existing System

When comparing the capabilities of existing bakery platforms, such as Kenny Hills Bakers, Magnolia Bakery, and Hogan Bakery, it is clear that while all three support basic order management and have visually appealing user interfaces, they fall short of providing more advanced or integrated functionalities such as class registration and reporting features.

Kenny Hills Bakers, for example, provides a streamlined online buying experience with a mobile-friendly design and a well chosen product inventory. However, it lacks class scheduling and individual dashboards for users. Magnolia Bakery prioritizes branding and product diversity, but its digital platform is largely designed for location-based ordering and lacks backend statistics or scheduling assistance. Similarly, Hogan Bakery offers responsive ordering capabilities but has minimal backend capacity for company insights or client involvement other than product sales.

As shown in Table 2, none of these systems provide class registration, which is a critical feature for bakeries that provide educational services or customer engagement via workshops. Furthermore, none of them provide automated reporting and analytics, which may greatly improve decision-making for small organizations. In contrast, the Orcakes Patisserie system combines both capabilities, allowing customers to register for courses and administrators to compile information on sales, inventory, and user activity. This distinguishes Orcakes not just as a digital storefront, but also as a full bakery management software [1][3][5].

When comparing the proposed system to the existing ones, it is clear that the Orcakes solution fills gaps in business information, service diversity, and operational efficiency, making it more appropriate for bakeries seeking long-term growth and digital transformation.

Table 3 Comparison System

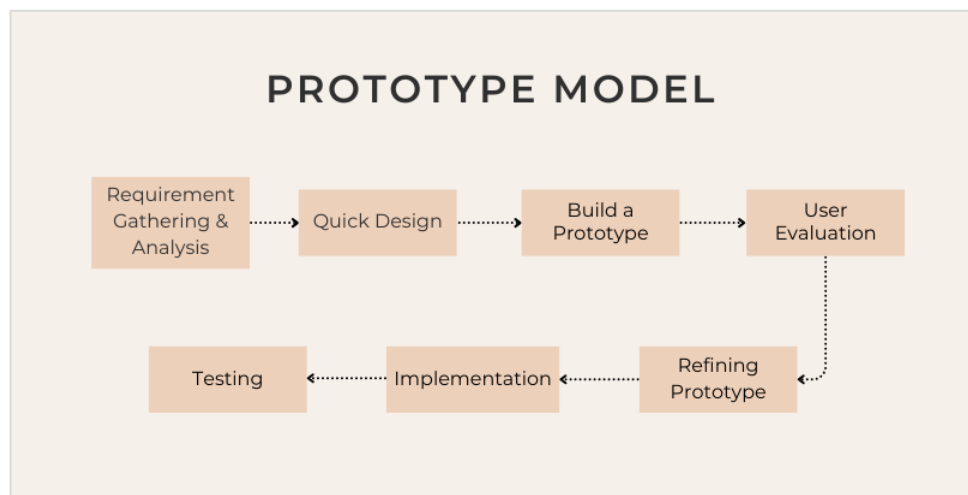
FEATURE	KENNY HILLS BAKERS	MAGNOLIA BAKERY	HOGAN BAKERY	ORCAKES PATISSERIE ONLINE ORDERING AND CLASS REGISTRATION SYSTEM
Order Management	✓	✓	✓	✓
Class Registration	x	x	x	✓
User Interface	✓	✓	✓	✓
Reporting and Insights	x	x	x	✓

3. Methodology

A methodology offers an organized set of tools methods, and guidelines to direct a project's design, implementation, and administration. It assures task consistency and facilitates the effective accomplishment of project objectives. The Prototyping Model has been chosen for this project because it supports user interaction at every step and enables iterative development[6],[9]. This strategy reduces risks and enhances the system's conformity to stakeholder expectations [7],[9].

The Prototyping Model includes creating a prototype, or early version of the system, which serves as the basis for obtaining user and stakeholder input. The system is iteratively refined based on feedback until the final product is completed. By addressing possible problems early in the development process, this approach improves system usability and minimizes misconceptions [8],[9].

However, the Prototyping Model's iterative nature might make managing modifications and avoiding scope creep more difficult. Requirement gathering, rapid design, prototype construction, user assessment, refinement, and execution are the six main phases of this process. Figure 1 shows the phases in the Prototype Model.

**Figure 1** Phases in Prototype Model

3.1 System Development Workflow

There are total of seven phases that are involved in this prototyping model. Table 3 shows the system development activities and the deliverables for each phase

Table 4 *Software Development Phases Activities*

Phase	Task	Output
Requirement Gathering and Analysis	- Meetings with Orcakes Patisserie staff and customers - Brainstorming and requirement analysis - Identifying manual order processing inefficiencies	- Preliminary requirement specification document - Identified functional and non-functional requirements
Quick Design	- Creating system wireframes and flow diagrams using tools like Lucidchart and Figma - Drafting database structure and interface designs	- Basic database structure - Wireframes - System flow diagrams
Build a Prototype	Developing a functional prototype including class registration module, online ordering, and product catalog browsing	- Prototype system showcasing essential features - Source files and documentation
User Evaluation	- Conducting usability testing sessions with stakeholders - Gathering feedback via surveys and interviews - Observing user interactions	- User evaluation report - Identified usability issues and improvement areas
Table 3 <i>Software Development Phases Activities(cont.)</i>		
Refining Prototype	- Improving functionality and fixing bugs based on feedback - Enhancing user interface and usability - Conducting repeated testing	- Updated prototype - Revised feature and design documentation
Implementation	- Finalizing the system for deployment - Providing user manuals and training to bakery staff - Deploying the system in a live environment	- Fully implemented system - Instructional materials - Maintenance and observation plans
Testing	- Conducting user acceptance testing (UAT) to ensure all requirements are met - Identifying and resolving any issues before deployment	- Detailed test reports - Final approval for system launch

4. Analysis and Design

Analyzing a system involves evaluating its current state, identifying problems, and establishing its goals. In order to make sure the Orcakes Patisserie Online Ordering and Class Registration System achieves its objectives, this section focuses on evaluating and creating it. In order to comprehend user demands and limitations, it starts with describing the functional and non-functional requirements for the system. UML specifications serve as the foundation for technical implementation by offering visual representations of system processes and interactions. To ensure smooth operation and an intuitive interface, this section also includes database and interface designs, providing a solid basis for a reliable and user-focused system.

4.1 Functional and Non-Functional Requirements

Functional requirements define the specific functionalities and capabilities that a system, software, or product must perform. They outline what the system should do, detailing the essential tasks, operations, and interactions it needs to support to meet user needs and organizational objectives. Table 4 shows the modules and functionalities of the proposed system.

Table 5 *Functional Requirement of the system*

No	Modules	Functionalities
1	User Registration	By entering the required information, users will be able set up accounts on the system. After a successful registration, users are sent to the login page. The system enables logout capability, validates user credentials, and, following successful login, reroutes users to role-specific dashboards
2	Profile	Users may browse their profiles and change personal information like name and contact details.
3	Class Registration	Enable clients to register, check available class schedules, and get confirmations. Allow administrators to maintain and update class schedules, assuring users availability and validity.
4	Orders	Offer clients the option to browse items, add them in their shopping cart, and make safe payments. Give administrators the ability to efficiently monitor, adjust, and track client orders.
5	Transaction Payment	Give clients the option to safely pay for orders and class registrations while give administrators access to the system's transaction history.
6	Generate Report	Give administrators the ability to create reports on user activity, sales, and inventory.

Non-functional requirements, on the other hand, include the standards that specify how the system should behave and operate, not just its particular features. In order to ensure that the system satisfies user expectations, regulatory compliance, and organizational demands while addressing elements beyond basic functionality, non-functional requirements offer the quality criteria by which the system performance is assessed. The non-functional needs of the suggested system are displayed in Table 5 below.

Table 6 *Non-Functional Requirement of the system*

No	Requirements	Descriptions
1	Operational	The system should be accessible by all and work effectively on every major web browsers and devices.
2	Performance	Fast response times and uninterrupted support for up to 100 concurrent users are requirements for the system.
3	Safety	Secure authentication and encryption should be used for securing all user information and transactions.
4	Accessibility	To handle users at any time without downtime, the system must be accessible 24 hours a day.

4.2 User Requirement Analysis

Table 7 *User Requirements of the system*

No.	User Requirements
1.	Clients should to be able to safely register and log in.
2.	Clients must be able to explore and easily sign up for classes.
3.	Clients should to be able to access and modify their profiles, including their personal data.
4.	It should be possible for administrators to view and modify class schedules.
5.	Order history and class registrations must be shown on the user's profile.
6.	Administrators should to be able to keep an eye on and control user transactions, activities, and performance data.

4.3 Use Case Diagram

In the Unified Modelling Language (UML), a use case diagram is a graphic representation that shows how a system communicates with its users or other external parties. It displays the different features or activities that a system does in response to certain requests made by actors, which might be people, systems, or other entities. The use case diagram may be seen in the figure below.

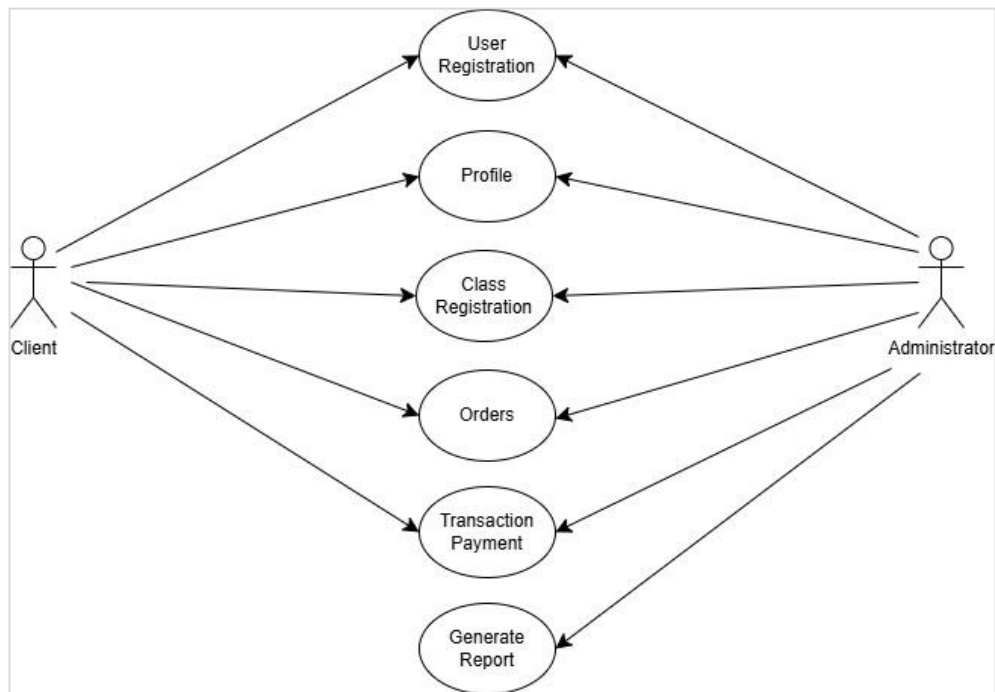


Figure 2 Use Case Diagram

4.4 Class Diagram

In Unified Modelling Language (UML), a class diagram is a graphic representation that shows the linkages and organization of a system. Classes, their properties, methods, and the relationships between other classes are all displayed. The suggested system's domain class diagram is displayed in Figure 3.

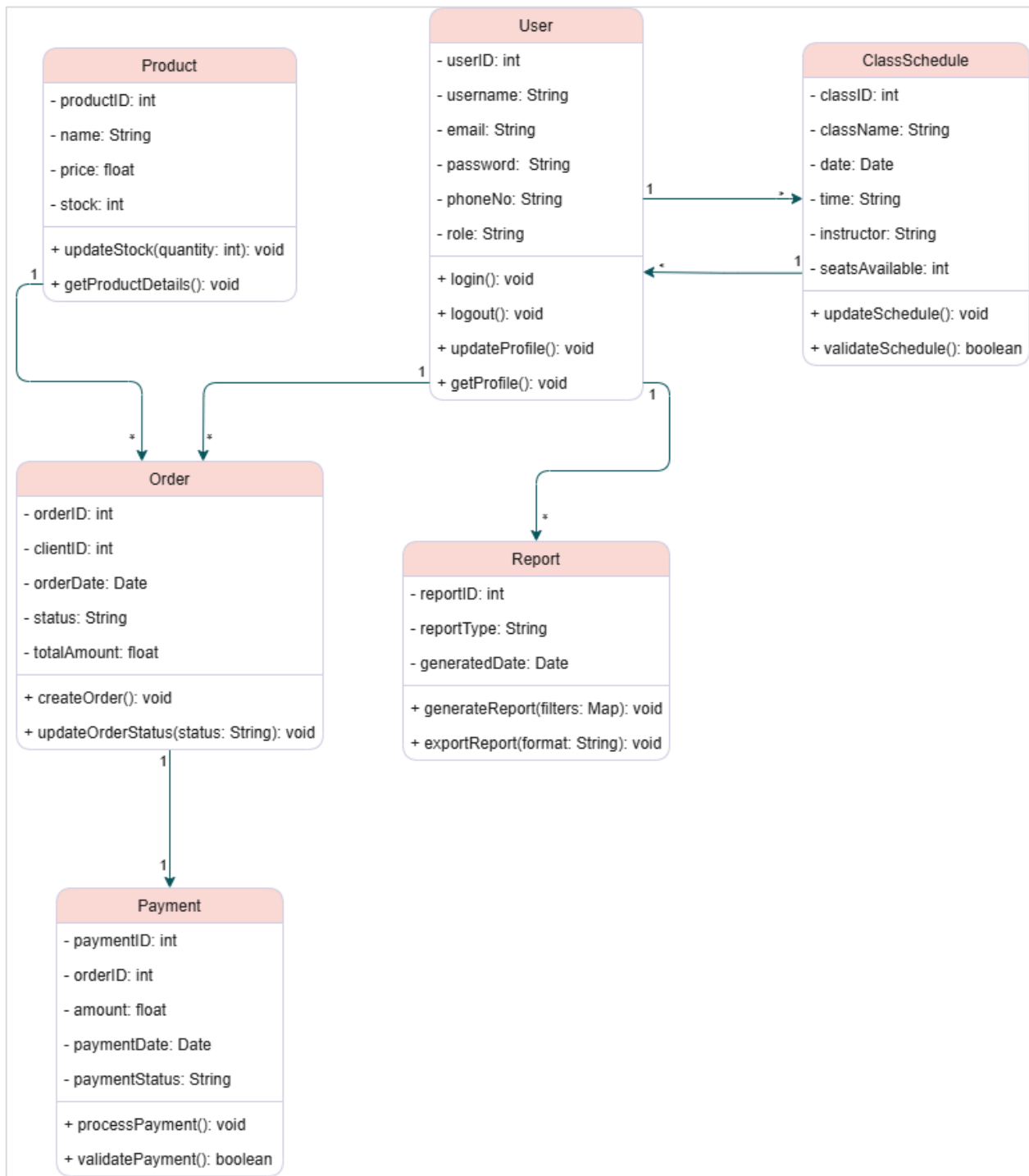


Figure 3 Class Diagram of the system

4.5 Interface Design

Interface design involves creating visually and functionally cohesive systems that facilitate user interaction with the system. Interface design of the proposed system shown in the Figure 4 to Figure 14.



Sign Up for Orcakes Patisserie

username

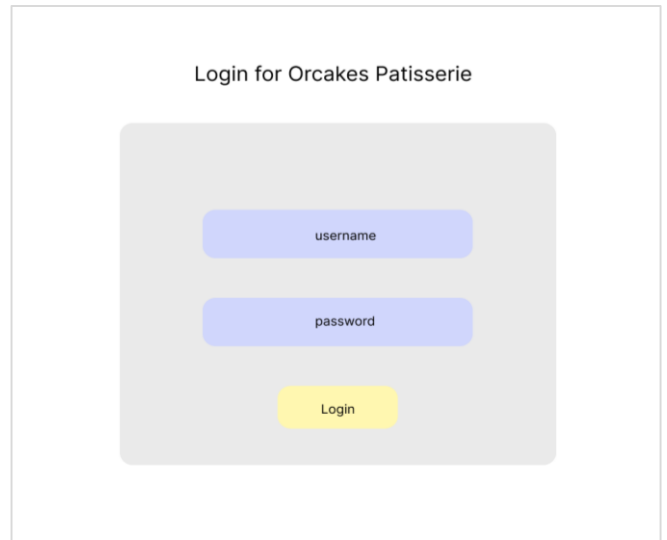
email

password

password confirmation

Create an Account

Fig. 4 Sign up Interface



Login for Orcakes Patisserie

username

password

Login

Fig.5 Login Interface



ORCAKES PATISSERIE

Shop Class Registration Order History Profile

IMAGE



ORCAKES PATISSERIE

Image Username

Username

Email

Full Name

Phone No

Address

Update

Fig. 6 Profile Interface

Fig.7 Homepage Interface

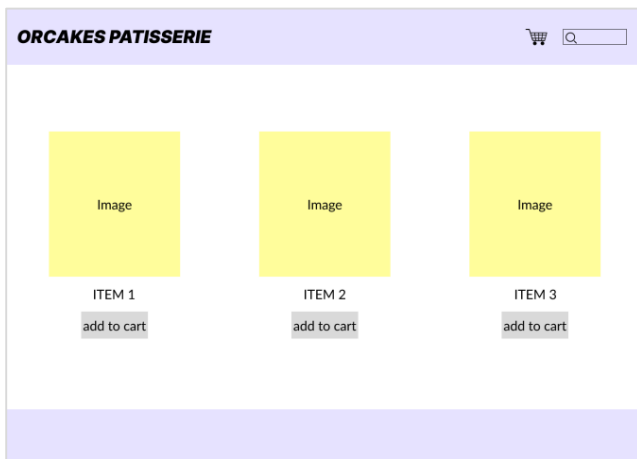


Fig. 8 Menu Interface

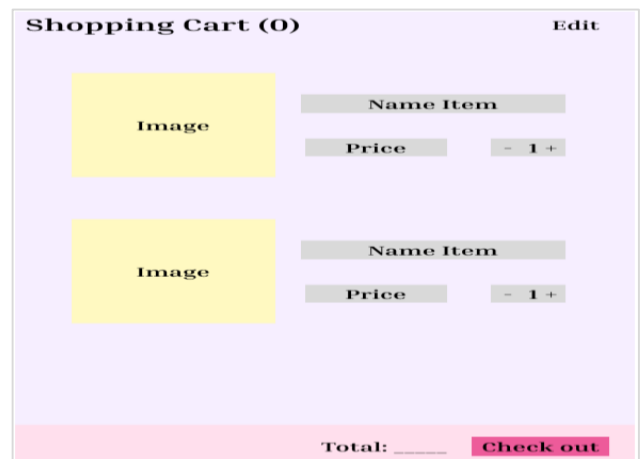


Fig. 9 Cart Interface



Fig. 10 Class Registration Interface



Fig. 11 View Order Interface

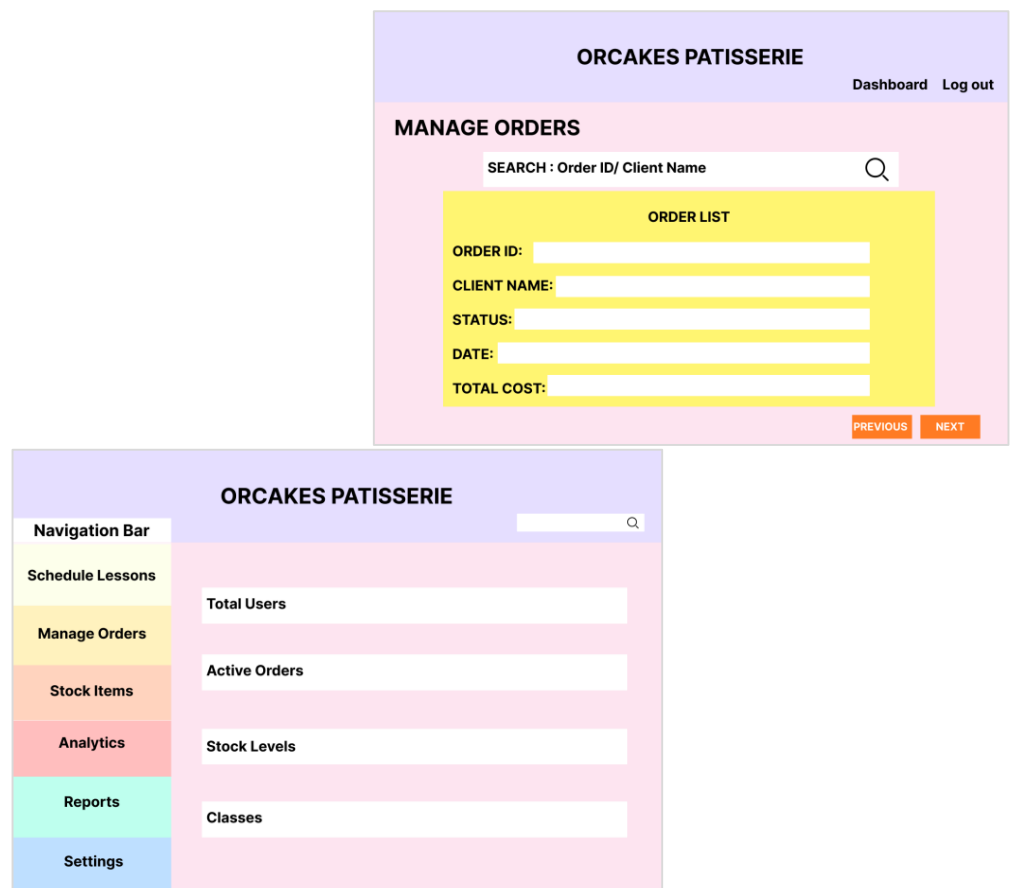


Fig. 12 Admin Dashboard Interface

Fig. 13 Manage Order Interface



Fig. 14 Manage Class Interface

5. Implementation and Testing

The Orcakes Patisserie Online Ordering and Class Registration System's installation had three major life-cycle steps. The installation was meant to initially support three types of users: the administrator, staff and the client. Thus the development utilized the object-oriented design and web-based application methodology. MySQL was used for the back-end so as to harness the strong database features of the system. Major components were installed like Orders, Class Registration, User Registration, Profile, and Transaction Payment. Secure payment gateway implementation is recognized as critical in online food ordering systems to foster customer trust and seamless operations[3],[5]. Also, there was a reporting module in the system. The interface is shown in Figs. 4 through 12.

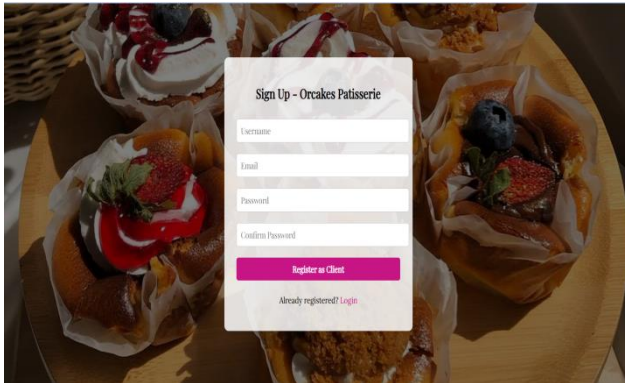


Fig. 15 Sign Up Interface

```

<body>
<div class="container">
<h2>Sign Up - Orcakes Patisserie</h2>
<form action="signup.php" method="post">
  <input type="text" name="username" placeholder="Username" class="input-field" required<br>
  <input type="email" name="email" placeholder="Email" class="input-field" required<br>
  <input type="password" name="password" placeholder="Password" class="input-field" required<br>
  <input type="password" name="confirm_password" placeholder="Confirm Password" class="input-field" required<br>
  <button type="submit" class="submit-btn">Register as Client</button><br>
</form>
<p>Already registered? <a href="login.php">Login</a></p>
</div>
</body>
</html>

```

Fig. 16 Sign up Code Segment

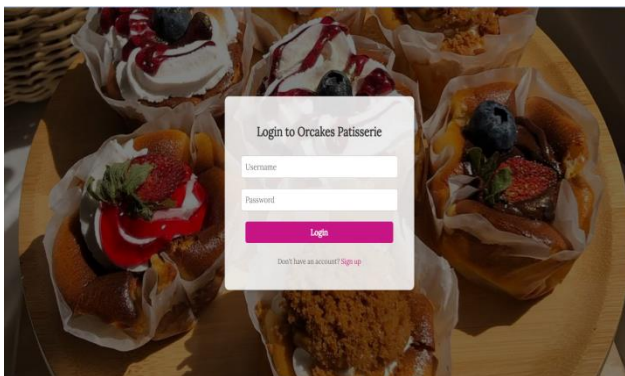


Fig. 17 Login Interface

```

<body>
<div class="container">
<h2>Login to Orcakes Patisserie</h2>
<form class="login-form" action="login.php" method="post">
  <input type="text" name="username" placeholder="Username" class="input-field" required<br>
  <input type="password" name="password" placeholder="Password" class="input-field" required<br>
  <button type="submit" class="submit-btn">Login</button><br>
</form>
<p>Don't have an account? <a href="signup.php">Sign up</a></p>
</div>
</body>
</html>

```

Fig. 18 Login Code Segment

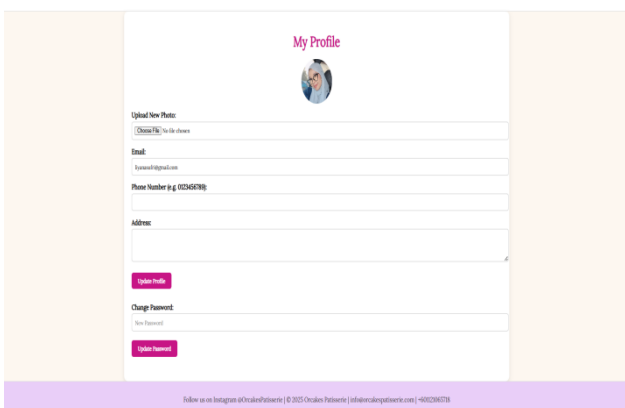


Fig.19 Profile Interface

```

<main>
<div class="profile-container">
  <h1>My Profile</h1>
  <div class="section">
    <form method="POST" enctype="multipart/form-data">
      
      <label for="profileImage">Upload New Photo</label>
      <input type="file" name="profileImage" accept="image/*" />
      <label for="email">Email</label>
      <input type="email" name="email" value="{$email}" />
      <label for="phone">Phone Number (e.g. 0123456789)</label>
      <input type="text" id="phone" name="phone" pattern="[0-9]{10-11}" title="Enter a valid Malaysian phone number (e.g., 0123456789)" required />
      <label for="address">Address</label>
      <textarea name="address" rows="3" />
      <button type="submit">Update Profile</button>
    </form>
  </div>
  <div class="section">
    <form method="POST">
      <label for="password">Change Password</label>
      <input type="password" name="newPassword" placeholder="New Password" required />
      <button type="submit">Update Password</button>
    </form>
  </div>
</div>

```

Fig. 20 Profile Code Segment

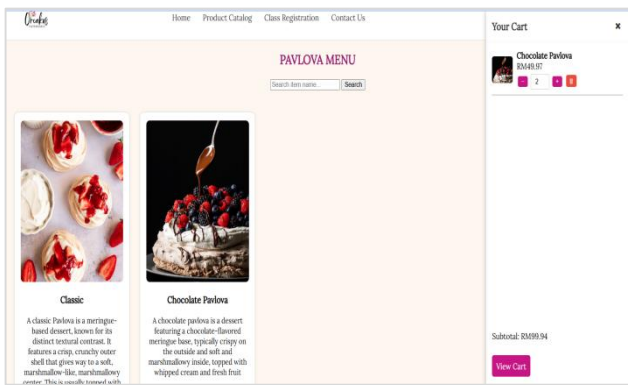


Fig. 21 Menu Interface

```

<div class="item-grid">
  <div class="item">
    
      <div class="item-name">
        <h3>{{ $item->title }}</h3>
      </div>
      <div class="item-description">
        <p>{{ $item->description }}</p>
      </div>
      <div class="item-price">
        <p>{{ $item->price }}</p>
      </div>
      <div class="item-quantity">
        <div class="quantity">
          <input type="text" value="1" />
        </div>
        <div class="add-to-cart">
          <button type="button" value="Add to Cart" />
        </div>
      </div>
    </div>
  </div>
</div>

```

Fig. 22 Menu Code Segment

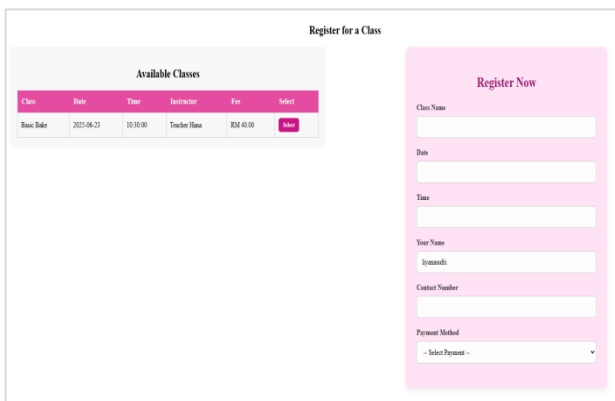


Fig. 23 Class Registration Interface

```

<div class="form-section">
  <div class="register-form">
    <div class="form-group">
      <input type="text" name="class_id" id="class_id" />
      <label class="name-label">
        <input type="text" id="class_name" />
      </label>
      <input type="text" id="class_date" />
      <input type="text" id="class_time" />
      <input type="text" name="client_name" value="" />
      <input type="text" name="contact_number" />
      <input type="text" name="payment_method" />
      <input type="text" name="fee" />
      <input type="text" name="contact_email" />
      <input type="text" name="password" />
      <input type="text" name="confirm_password" />
      <input type="button" value="Register" />
    </div>
  </div>
</div>

```

Fig. 24 Class Registration Code Segment

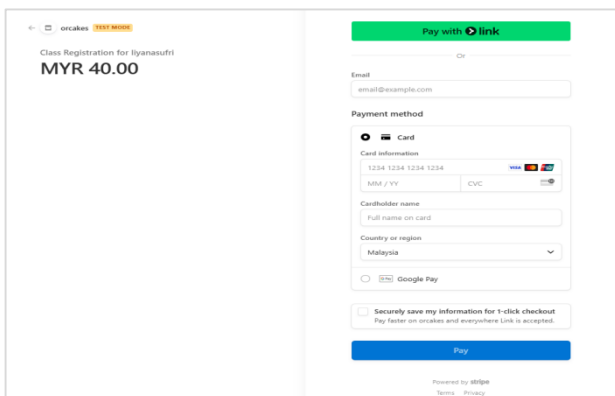


Fig. 25 Manage Payment Interface

```

<div class="stripe-checkout">
  <div class="stripe-form">
    <div class="stripe-card">
      <input type="text" name="card_number" />
      <input type="text" name="card_expiry" />
      <input type="text" name="card_cvc" />
      <input type="text" name="cardholder_name" />
      <input type="text" name="cardholder_email" />
      <input type="button" value="Pay with Stripe" />
    </div>
  </div>
</div>

```

Fig. 26 Manage Payment Code Segment

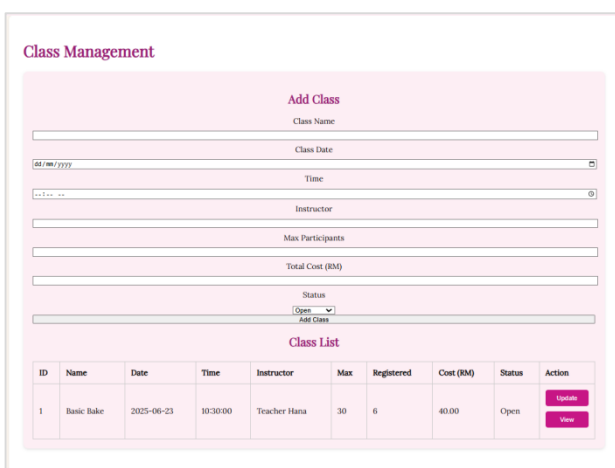


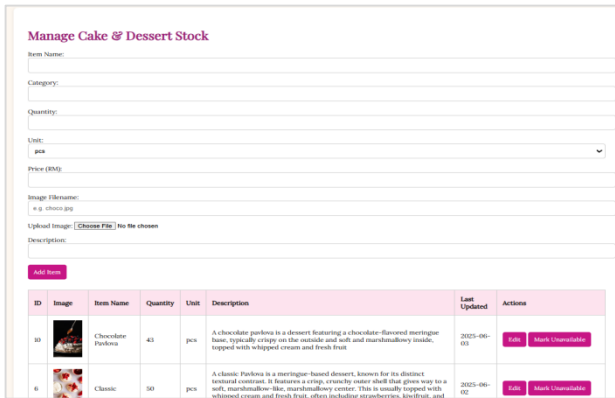
Fig. 27 Manage Class Interface

```

<div class="manage-class">
  <div class="add-class">
    <div class="form-group">
      <input type="text" name="class_name" />
      <input type="text" name="class_date" />
      <input type="text" name="class_time" />
      <input type="text" name="instructor" />
      <input type="text" name="max_participants" />
      <input type="text" name="total_cost" />
      <input type="button" value="Add Class" />
    </div>
  </div>
  <div class="class-list">
    <table>
      <thead>
        <tr>
          <th>ID</th>
          <th>Name</th>
          <th>Date</th>
          <th>Time</th>
          <th>Instructor</th>
          <th>Max</th>
          <th>Registered</th>
          <th>Cost (RM)</th>
          <th>Status</th>
          <th>Action</th>
        </tr>
      </thead>
      <tbody>
        <tr>
          <td>1</td>
          <td>Basic Bake</td>
          <td>2025-06-23</td>
          <td>10:30:00</td>
          <td>Teacher Hana</td>
          <td>30</td>
          <td>6</td>
          <td>80.00</td>
          <td>Open</td>
          <td><button type="button" value="Update" /><button type="button" value="View" /></td>
        </tr>
      </tbody>
    </table>
  </div>
</div>

```

Fig. 28 Manage Class Code Segment



Manage Cake & Dessert Stock

Item Name:

Category:

Quantity:

Unit:

Price (RM):

Image Filename:

Upload Image: No file chosen

Description:

ID	Image	Item Name	Quantity	Unit	Description	Last Updated	Actions
10		Chocolate Pavlova	45	pcs	A chocolate pavlova is a dessert featuring a chocolate-flavored meringue base, typically crispy on the outside and soft and marshmallowy inside, topped with whipped cream and fresh fruit.	2025-06-10	Edit Stock Unavailable
6		Classic	50	pcs	A classic Pavlova is a meringue-based dessert, known for its distinct funnelled crown. It features a crisp, crumbly outer shell that gives way to a soft, marshmallow-like, marshmallowy center. This is usually topped with whipped cream and fresh fruit, often including strawberries, kiwifruit, and	2025-06-10	Edit Stock Unavailable

Fig. 29 Manage Stock Interface

```

<div class="content">
<h1>Manage cake & Dessert Stock</h1>

<!-- Add form -->
<form method="post" action="stock.php" enctype="multipart/form-data">
  <div class="form-group">
    <label>Item Name</label>
    <input type="text" name="itemname" required>
  </div>

  <div class="form-group">
    <label>Category</label>
    <input type="text" name="category" required>
  </div>

  <div class="form-group">
    <label>Quantity</label>
    <input type="number" name="quantity" required>
  </div>

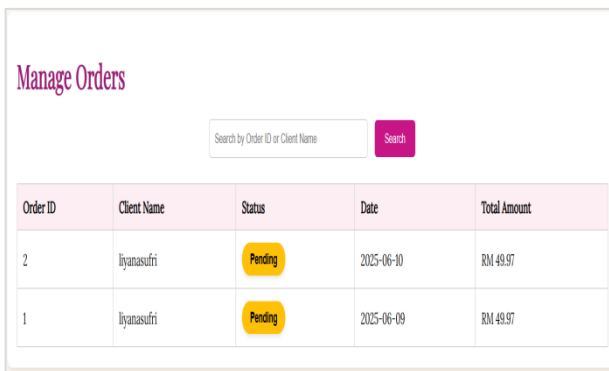
  <div class="form-group">
    <label>Unit</label>
    <select name="unit"><option>pcs</option><option>slices</option><option>boxes</option></select>
  </div>

  <div class="form-group">
    <label>Price (RM)</label>
    <input type="number" step="0.01" name="price" required>
  </div>

  <div class="form-group">
    <label>Image Filename</label>
    <input type="text" name="imagefilename" placeholder="e.g. choco.jpg">
  </div>
</form>

```

Fig. 30 Manage Stock Code Segment



Manage Orders

Search by Order ID or Client Name

Order ID	Client Name	Status	Date	Total Amount
2	Ilyanasufri	Pending	2025-06-10	RM 49.97
1	Ilyanasufri	Pending	2025-06-09	RM 49.97

Fig. 31 Manage Orders Interface

```

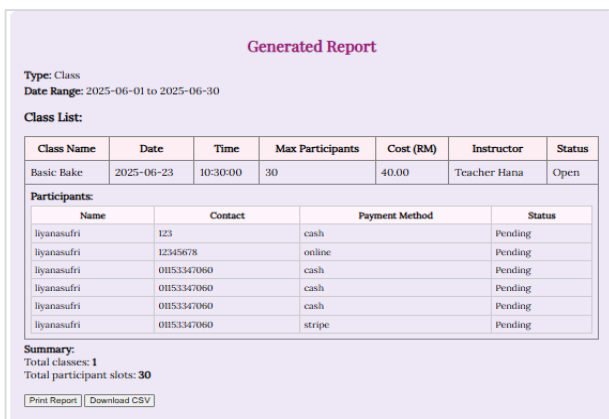
<!-- Main Content -->
<div class="content">
<h1>Manage Orders</h1>

<div class="search-container">
  <input type="text" id="searchInput" placeholder="Search by Order ID or Client Name">
  <button onclick="searchOrders()">Search</button>
</div>

<table id="orderTable">
  <thead>
    <tr>
      <th>Order ID</th>
      <th>Client Name</th>
      <th>Status</th>
      <th>Date</th>
      <th>Total Amount</th>
    </tr>
  </thead>
  <tbody id="orderTableBody">
    <!-- Will be filled by fetch_admin_orders.php using AJAX -->
  </tbody>
</table>

```

Fig. 32 Manage Orders Code Segment



Generated Report

Type: Class

Date Range: 2025-06-01 to 2025-06-30

Class List:

Class Name	Date	Time	Max Participants	Cost (RM)	Instructor	Status
Basic Bake	2025-06-23	10:30:00	30	40.00	Teacher Hana	Open

Participants:

Name	Contact	Payment Method	Status
Ilyanasufri	123	cash	Pending
Ilyanasufri	12345678	online	Pending
Ilyanasufri	01153347060	cash	Pending
Ilyanasufri	01153347060	cash	Pending
Ilyanasufri	01153347060	cash	Pending
Ilyanasufri	01153347060	stripe	Pending

Summary:
Total classes: 1
Total participant slots: 30

Fig. 33 Report Interface

```

<div class="main-content">
  <h1>Welcome to Orca's Patisserie</h1>
  <p>Generate reports for user activity, sales, and inventory.</p>
  <p>Use the form below to generate a report.</p>
</div>

<div class="content">
  <h2>Generate Report</h2>
  <form method="post" action="report.php">
    <div class="form-group">
      <label for="reportType">Select Report Type</label>
      <select id="reportType" name="reportType" required>
        <option value="">Select --</option>
        <option value="userActivity">User Activity</option>
        <option value="sales">Sales</option>
        <option value="inventory">Inventory</option>
        <option value="staff">Staff</option>
        <option value="class">Class</option>
      </select>
    </div>

    <div class="form-group">
      <label for="dateFrom">Date From</label>
      <input type="date" id="dateFrom" name="dateFrom" required>
    </div>

    <div class="form-group">
      <label for="dateTo">Date To</label>
      <input type="date" id="dateTo" name="dateTo" required>
    </div>

    <div class="form-actions">
      <button type="submit" name="generate">Generate Report</button>
    </div>
  </form>

```

Fig. 34 Report Code Segment

5.1 Test Case

Test cases are divided into 9 categories, one for each module in the system. The capabilities tested derive from the functional criteria enumerated in chapter 4, making sure there is a thorough inquisitive confirmation of every need mentioned. This systemic attitude enables easier monitoring for the realization of each condition. Tables 7 to 15 contain results and full test details.

As shown in Table 7, all test cases related to the Sign-Up functionality were successfully passed. The system allowed both administrator and client users to register by entering valid credentials and redirected them appropriately to the login page. This confirms that the user registration module meets the functional requirements defined in SRS_REQ_100.

Table 8 Test Case for Sign Up

Software Requirement Specification	Test Case ID	Test Case Description	Test Result (Pass/Fail)
SRS_REQ_100	TEST_100	Sign Up	

SRS_REQ_100	TEST_101	The administrator and client can create new accounts by entering their username, email, and confirming their password.	Pass
SRS_REQ_100	TEST_102	After successful sign up, the system should redirect the administrator and client to the login page.	Pass

As indicated in Table 8, the login module correctly evaluated user credentials and led users to the appropriate role-based dashboard after a successful login. Furthermore, users may safely log out and stop sessions. These results demonstrate that the system meets all login-related standards under SRS_REQ_200-SRS_REQ_203, including secure authentication and session management.

Table 9 *Test Case for Login*

Software Requirement Specification	Test Case ID	Test Case Description	Test Result (Pass/Fail)
SRS_REQ_200	TEST_200	Login	
SRS_REQ_201	TEST_201	The system will check user credentials before granting access.	Pass
SRS_REQ_202	TEST_202	After a successful login, the system will provide a redirect to the user's designated homepage or dashboard based on their responsibilities.	Pass
SRS_REQ_203	TEST_203	After a session, the user will be able to log out of the system.	Pass

Table 9 demonstrates that the profile management module satisfied all of the test requirements. Users, including administrators, staff, and clients, may see and edit their personal information, such as name, phone number, and profile photo. This is consistent with the expectations indicated in SRS_REQ_300-SRS_REQ_302, which confirm the system's capabilities for user customization and data management.

Table 10 *Test Case for Manage Profile*

Software Requirement Specification	Test Case ID	Test Case Description	Test Result (Pass/Fail)
SRS_REQ_300	TEST_300	Manage Profile	
SRS_REQ_301	TEST_301	Administrator, staff and the client will be able to manage their profile through the system, including updating their name, phone number, and profile picture.	Pass
SRS_REQ_302	TEST_302	Administrator, staff and the client will be able to access their profiles on the system.	Pass

The system's order feature passed all four test scenarios, as shown in Table 10. Clients were able to add things to their carts and successfully check out, while administrators and staff could track order status in real time and view detailed order information. The successful outcomes for SRS_REQ_400-SRS_REQ_404 show that the system supports smooth order processing and real-time communication between users and back-end personnel.

Table 11 *Test Case for Order Module*

Software Requirement Specification	Test Case ID	Test Case Description	Test Result (Pass/Fail)
SRS_REQ_400	TEST_400	Order Module	
SRS_REQ_401	TEST_401	The client may check out after adding things to their cart, and the admin or staff dashboard will show the successful placement of their purchase.	Pass

SRS_REQ_402	TEST_402	A list of the client previous orders, complete with exact data, will appear when they browse their order history.	Pass
SRS_REQ_403	TEST_403	The status of an order is updated in real-time and appears immediately on the client side when an administrator or staff member modifies it (for example, from Pending to Done).	Pass
SRS_REQ_404	TEST_404	All order details, including goods, quantity, custom remarks, and client information, are completely shown when an administrator or staff member reads detailed order information.	Pass

As shown in Table 11, the payment module successfully updated the order status after checkout and provided a confirmation page to users. This test validates SRS_REQ_500-SRS_REQ_501, demonstrating that the system safely and reliably supports both Stripe and Cash on Delivery options.

Table 12 *Test Case for Payment Module*

Software Requirement Specification	Test Case ID	Test Case Description	Test Result (Pass/Fail)
SRS_REQ_500	TEST_500	Payment Module	
SRS_REQ_501	TEST_501	The order status is updated and a confirmation screen appears once you choose to pay using Stripe or Cash on Delivery.	Pass

The class registration module passed all tests, as shown in Table 12. Clients may register for available classes and receive real-time updates when administrators added or updated class listings. This demonstrates compliance with SRS_REQ_600-SRS_REQ_603, demonstrating the system's capacity to dynamically manage educational programs and keep participants informed.

Table 13 *Test Case for Class Registration*

Software Requirement Specification	Test Case ID	Test Case Description	Test Result (Pass/Fail)
SRS_REQ_600	TEST_600	Class Registration	
SRS_REQ_601	TEST_601	Once a client selects and registers for an available class, a registration confirmation appears, and the administrator or staff can see the participant list.	Pass
SRS_REQ_602	TEST_602	The new class instantly shows up on the client's class registration page once an administrator or staff member adds a new class listing with its date, time, and capacity.	Pass
SRS_REQ_603	TEST_603	Clients are instantly updated when an administrator modifies an existing class's information, such as its date or capacity.	Pass

Table 13 shows how stock items added or changed by administrators or personnel appeared instantly in the catalog for clients. These findings meet SRS_REQ_700-SRS_REQ_702, indicating that inventory changes are updated in real time and the catalog is still correct.

Table 14 *Test Case for Manage Stocks*

Software Requirement Specification	Test Case ID	Test Case Description	Test Result (Pass/Fail)
SRS_REQ_700	TEST_700	Manage Stocks	
SRS_REQ_701	TEST_701	A new stocks is instantly visible for clients to see and purchase when an administrator or staff adds it to the catalog.	Pass
SRS_REQ_702	TEST_702	A product's stock amount is updated and reflected instantly when an administrator or staff member makes changes.	Pass

Table 14 confirmed the system's capacity to create thorough reports. Administrators have access to accurate daily and monthly information on sales, inventory, user activity, and class enrollment. This validates that SRS_REQ_800-SRS_REQ_801 has been met and that the reporting module facilitates operational decision-making.

Table 15 *Test Case for Generate Reports*

Software Requirement Specification	Test Case ID	Test Case Description	Test Result (Pass/Fail)
SRS_REQ_800	TEST_800	Generate Reports	
SRS_REQ_801	TEST_801	Table reports appear as expected when the administrator sees the daily or monthly report of sales, inventory, user activity, and class registrations.	Pass

As seen in Table 15, the user management capability worked as intended. Administrators may examine all registered users and deactivate them as needed. Clients could no longer access the system once it was disabled, which met the conditions outlined in SRS_REQ_900-SRS_REQ_902.

Table 16 *Test Case for User Management*

Software Requirement Specification	Test Case ID	Test Case Description	Test Result (Pass/Fail)
SRS_REQ_900	TEST_900	User Management	
SRS_REQ_901	TEST_901	A detailed list of both client and staff users appears when the administrator checks the list of all registered users.	Pass
SRS_REQ_902	TEST_902	When a client account is deactivated by a admin, the user may no longer log in and the account is listed as inactive.	Pass

5.2 Overall Test Case Result

A comprehensive web-based tool created to optimize bakery operations is the Orcakes Patisserie Online Ordering and Class Registration System. In addition to centralizing and organizing all operational data, it streamlines the

ordering and class registration processes. Clients can register and manage their orders and class reservations with ease thanks to this system, which also helps staff fulfill orders efficiently and gives administrators the ability to effortlessly supervise everything. It greatly improves customer satisfaction and administrative efficiency with features including user registration, class registration, online ordering with customization, secure payment processing, and thorough reporting. The ultimate goal of this system is to modernize bakery operations by improving internal management and the convenience of the client experience.

Table 17 Overall Test Cases

Test CASE ID	Total Test Cases	Total Passes
TEST_100	2	2
TEST_200	3	3
TEST_300	2	2
TEST_400	4	4
TEST_500	1	1
TEST_600	3	3
TEST_700	2	2
TEST_800	1	1
TEST_900	2	2

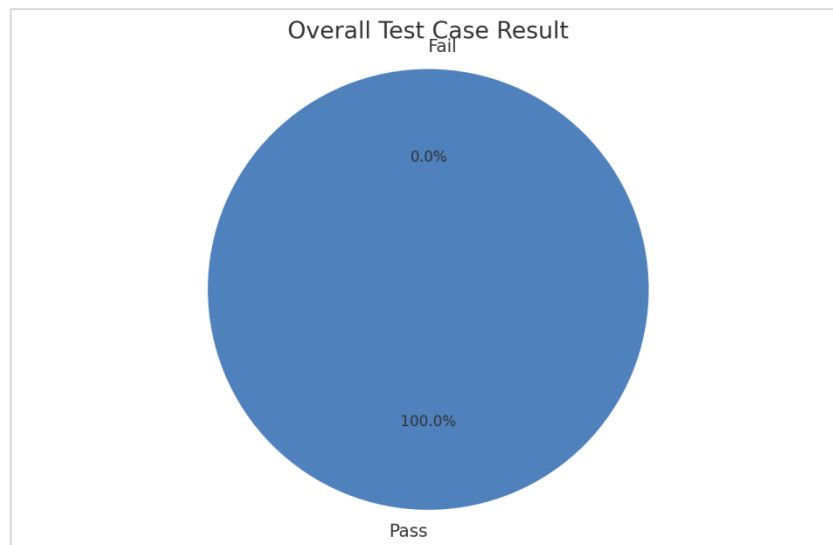


Fig. 35 Overall Test Result

5.3 User Acceptance Testing

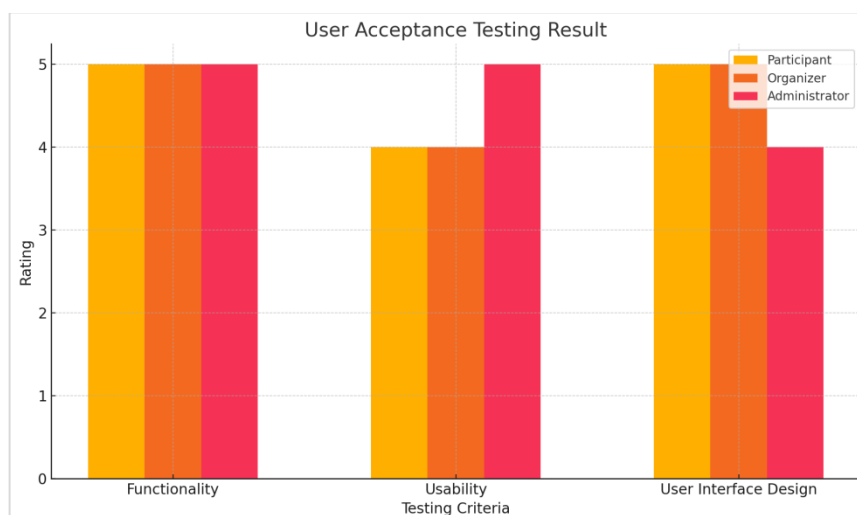


Fig. 36 User Acceptance Testing Result

The user acceptability testing results for three categories—Usability, Functionality, and User Interface Design assessed by three user roles Administrator, Staff, and Client are displayed in a bar chart. Functionality received a perfect score of five from all roles, meaning the system effectively satisfies its stated functional criteria. Clients and staff gave the system a usability score of 4, while administrators gave it a slightly higher rating of 5, indicating that it provides a generally user-friendly experience with just minimal adjustments required for end-user interactions. Clients and staff gave the system's visual and interactive features a high rating of five stars for user interface design, while administrators gave it a slightly lower rating of four. Overall, the findings demonstrate the system's excellent functional performance and favorable user acceptance, with potential for interface improvement to better support administrative tasks.

6. Conclusion

In the first place, the main goal of the Orcakes Patisserie Online Ordering and Class Registration System was achieved, i.e., digitization and streamlining of the bakery order and class registration processes. By having the six interlinked modules developed, i.e., user registration, profile management, order placement, class scheduling, payment processing, and report generation, the system thus enhanced its operational efficiency and offered the customer better services. Considerable testing was carried out to verify the system's functionality, after which user acceptance testing revealed that users were highly satisfied in terms of usability, functionality, and interface design.

Yet, the system has its drawbacks. The present payment system is quite secure but limited only to a Stripe payment gateway and Cash on Delivery, which may not be to the liking of every customer. Another limitation is that the system does not have features like inventory auto-restocking alerts, management of promotional campaigns, and a customer review system. Having such features built into the system might further boost customer retention.

In the future, the possible improvements may consist of the inclusion of a loyalty rewards module, payment gateway options (FPX, e-wallets, and others) with an expansion, and AI sales analytics that provide a deeper layer of insight into business decisions. It may also improve user access and convenience on the go by implementing a mobile app. All these enhancements will support the long-term digital transformation and scalability agenda of the bakery.

Hence, all in all, the system is a fit to bring in an initial step towards digitizing Orcakes Patisserie while punting and solving key operational challenges, thus setting the platform for further innovations in artisanal bakery industries.

Acknowledgement

The authors would like to thank the Faculty of Computer Science and Information Technology, Universiti Tun Hussein Onn Malaysia for its support.

Conflict of Interest

Authors declare that there is no conflict of interests regarding the publication of the paper.

Author Contribution

This journal requires that all authors take public responsibility for the content of the work submitted for review. The contributions of all authors must be described in the following manner:

*The authors confirm contribution to the paper as follows: **study conception and design:** Puteri Nur Fathimah Binti Radzin, Firkhan Ali Bin Hamid Ali; **data collection:** Puteri Nur Fathimah Binti Radzin; **analysis and interpretation of results:** Puteri Nur Fathimah Binti Radzin, Firkhan Ali Bin Hamid Ali; **draft manuscript preparation:** Puteri Nur Fathimah Binti Radzin, Firkhan Ali Bin Hamid Ali. All authors reviewed the results and approved the final version of the manuscript.*

An author name can appear multiple times, and each author name must appear at least once. For single authors, use the following wording:

The author confirms sole responsibility for the following: study conception and design, data collection, analysis and interpretation of results, and manuscript preparation.

References

- [1] Chayanis, C. (2021). THE FACTORS AFFECTING ONLINE PURCHASE INTENTION OF BAKERY PRODUCTS IN THAILAND (Doctoral dissertation, Mahidol University).
- [2] A. R. Agusti and W. I. Pasar V, "Online Ordering System on Bakery Website," BIOS Journal, vol. 9, no. 2, pp. 75–82, 2022.
- [3] Mozakrah. (2024). How to Get IGCSE Results: A Guide for Students & Parents. <https://mozakrah.com/en/blogs-details/student-registration-for-online-courses-a-comprehensive-guide-to-maximizing-benefits>
- [4] Dummer, J. (2024). What's On Trend In Baking For 2024: Final Proof - Bakers Journal. Bakers Journal. <https://www.bakersjournal.com/whats-on-trend-in-baking-for-2024-final-proof/>
- [5] IRJMETs, "Digitization of Bakery Products: A Consumer Behavior Study," International Research Journal of Modernization in Engineering Technology and Science, vol. 5, no. 4, pp. 42–48, 2023.
- [6] Confianz Global, "The Benefits of the Prototyping Model in Software Development," Confianz Tech Blog, Aug. 2023. [Online]. Available: <https://blog.confianz.it.com/the-benefits-of-the-prototyping-model>
- [7] Sommerville, I. (2015). Software Engineering (10th ed.). Pearson Education.
- [8] Boehm, B. W. (1986). A Spiral Model of Software Development and Enhancement. ACM SIGSOFT Software Engineering Notes, 11(4), 14-24. <https://doi.org/10.1145/12944.12948>
- [9] R. Budde and H. Züllighoven, "The Role of Prototyping in Software Development," Empirical Software Engineering, vol. 5, no. 1, pp. 45–65, 1990.