

Web-Based Café Hopper Platform

Nur Raudah Ezatul Md Sarif¹, Suhaimi Abd Ishak^{1*}

¹ *Fakulti Sains Komputer dan Teknologi Maklumat,*

Universiti Tun Hussein Onn Malaysia, Parit Raja, Batu Pahat, 86400, MALAYSIA

*Corresponding Author: suhaimiabd@uthm.edu.my

DOI: <https://doi.org/10.30880/aitcs.2025.06.02.013>

Article Info

Received: 13 June 2025

Accepted: 3 November 2025

Available online: 30 November 2025

Keywords

Cafe hopper, Web-based platform,
Discovery Tool, and Booking
Management

Abstract

The Café Hopper Platform is a web-based solution designed to enhance the café discovery and booking experience for both customers and café owners. Developed using Agile methodologies, the platform offers a centralized hub for locating cafés that align with diverse consumer preferences, such as pet-friendliness or Wi-Fi availability, while promoting visibility for small, local establishments. The front end is built with HTML, CSS, and JavaScript to ensure an intuitive user interface, while the back end leverages PHP and MySQL for secure data management, process integration, and user authentication. Key features include personalized café search, real-time booking, a payment gateway, and a unique review system. Integrated APIs support location-based search and advanced filtering, and the responsive design ensures usability across devices. Café owners can manage profiles, update information, and interact with customer feedback to boost engagement and reputation. Developed in a prototype scope using Visual Studio Code and XAMPP for the local environment, the platform's modular architecture enhances scalability and maintainability. By streamlining the café search process and offering robust tools for customer-business interaction, the Café Hopper Platform fosters a thriving café culture and supports the growth of independent café entrepreneurs.

1. Introduction

Over the last few years, café culture has occupied an important place in the modern metropolitan experience, with cafes evolving into more than just a place for getting an espresso shot. Today cafes are an important part of our lives, as they offer unique environments that satisfy the need for communication, work, and relaxation. Cafes are popular because of first and secondly, their differentiation and setting, which may be from fashionable brunch to professional coffee. Thus, more and more people are searching for cafes to their liking – whether they have something to do with the decor, food, or place. Perhaps the right café is hard to discover despite the presence of a myriad number of options. This is more so where certain aspects of the hotel such as places for pets, internet connectivity, or vegetarian meals.

Visiting a few websites, social network accounts, and reviews is often required to finally choose a café. This may take a lot of time and may not work as intended. Another problem some café consumers experience when searching for an ideal café to meet their requirements is sorting information that may be disorganized. Moreover, new clients are hard to come by for tiny community cafes given the competition between big brands and the popular national chains. A more complete and efficient solution that satisfies the requirements of the café operators and patrons can span this market hole.

The web-based Café Hopper Platform wants to fill this void and give café lovers a central place no matter where they are. The platform maintains a list of cafés with complete detailed information about the café, including ratings, photos, menu, and other features that allow users to browse cafés based on their preferences. Besides making café hopping easier, the website benefits local companies by bringing attention to them and developing a culture of people who like coffee. The obvious and now more straightforward way to view the café as a separate category with its unique style and concepts and to discover the gems that may have remained unnoticed by many is having access to the website that can help sort through reviews of the latest café experiences by users [1].

For easy understanding, this paper is developed into the following major sections. In the subsequent section 2, a literature review is conducted to provide a rationale for establishing the proposed system. Section three describes the methodology used to design and implement the system and the techniques used. Section 4 discusses the system analysis and design and the technical and functional perspective of the instituted platform. The last section, 5, presents the actual work of the system, addressing implementation and testing issues to prove its effectiveness. Finally, in the previous part, the conclusion summarizes the whole study and provides a possibility of future improvements to the system.

2. Literature Review

Café industry is a fast-growing industry across the globe, and cafes are part of socialising, hangouts, and working. What used to be considered a trend, cafe-hopping, which is the practice of patronizing more than one café during one visit, is now cultural. After the COVID-19 pandemic, this shift has been progressing. Coffee shop operators can ascertain this by heavily investing in drive-throughs and mobile orders to capture a change in people's preferences for convenience [2].

Web-based systems can be implemented on the World Wide Web and accessed through any device's web browser and internet connection. Web-based systems can be implemented without installing software and applications, making them easy for humans to work with. Web-based is a program using web technologies and is usually accessed on a server [3]. Web-based has to work in many economic sectors, including education, retailing, and others, by providing many benefits to many. It might assist them in increasing the size of the business and offering services to international clientele by incorporating Web-based technologies into their system.

The following identifies some potential of the web-based system to the community and industries. Today, web-based apps are a significant and still growing class of business apps that dramatically transform the creation, deployment, utilization, and administration of enterprise-grade solutions [4]. The educational, healthcare, and retail sectors are the primary users of web-based systems in their daily operations to enhance convenience.

The reason for selecting the Café Hopper scenario is that users mainly utilise this system to search for cafes. Café Hopper is useful for the continuously increasing demand for convenience by having a digital environment where users can easily search for cafes. It also allows people to share their experiences of hopping from cafe to cafe with other people. The web application is a good way to connect café lovers with their favourite places, as the trend of café hopping rises with the help of constantly developing digitalisation in the hospitality industry.

By analysing existing systems, the features implemented in the proposed solution are observed. Café search and geographical positioning is a component of Google Maps, which helps users quickly locate cafés near them, and with user reviews, comes convenient ways to plan one's route around the place. Zomato, on the other hand, focuses on restaurant aggregation that offers comprehensive information about the restaurant menus, customer ratings, web reviews, and dining experience, which makes it one of the favourite platforms among users for restaurant discovery. On the other hand, Eatigo targets restaurant bookings and customers willing to dine during a particular time when they are eligible for discounts. Each supplies substantial functionalities; however, none of the systems offers the essentials a café-hopper would want —recommended, luckily, morphed filters and community engagement. These limitations bring about the necessity of developing a dedicated solution such as the Web-Based Café Hopper Platform. Table 1 shows the comparison of the features of the existing system and the proposed system.

Table 1 Comparison with the Existing Systems

Features/System	Google Maps	Zomato	Eatigo	WebBasedCafé Hopper Platform
Café Discovery	√	√	√	√
Advanced Filters	X	√	X	√
User Reviews and Rating	√	√	X	√
Table Booking	X	X	√	√
Route Planning	√	X	X	√

Community Engagement	X	√	X	√
Photo Sharing	√	√	X	√
Mobile Notifications	X	X	√	√
Focus on Café Culture	X	√	X	√

3. Methodology

The Agile model was chosen for this project, which is a managed project that is flexible for collaboration with well-defined goals to move in stages. Agile means managing a project flexibly for collaboration with well-defined goals to move in stages. It focuses on providing consumers with small, functional software solutions and continually refining those items with consumer feedback. Agile encourages flexibility in planning, unlike the typical waterfall models, where planning is a one-time affair. This characteristic aligns well with the requirements of the Web-Based Café Hopper Platform. The Agile approach enables teams to deliver incremental value while continuously improving the end-user experience [5]. Figure 1 illustrates the Agile model used in this project, showing how iterative development and use feedback are incorporated at every stage.

Agile development is driven by the 12 principles of the Agile Manifesto. This system values the client and embraces continuous change, requiring the delivery of a working product at every stage. It promotes incremental sprint cycles, each delivering a functional software segment. This allows constant refinement and adjustment based on stakeholder feedback, creating a feedback loop to ensure the platform adapts effectively to user needs and experiences.

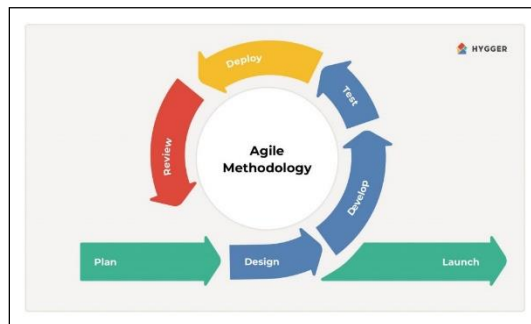


Figure 1 Agile Model

Table 2 System Development Workflow

Phase	Task	Output
Planning	<ul style="list-style-type: none"> Determine objectives, problem statement, and scope of the system Collect requirements from stakeholders 	<ul style="list-style-type: none"> Objective Problem Statement Scope Project Proposal
Analysis	<ul style="list-style-type: none"> Identify user requirements and system requirement Identify existing system Identify the comparison between the existing system and the proposed system 	<ul style="list-style-type: none"> Comparison table of the system Context Diagram DFD ERD

Design	<ul style="list-style-type: none"> • Design system and entity relationship • Design database schema • Design user interface 	<ul style="list-style-type: none"> • System architecture • Database schema • UI
Implementation	<ul style="list-style-type: none"> • Develop frontend and backend • Develop features 	<ul style="list-style-type: none"> • Functional User Interface • Complete system feature
Testing	<ul style="list-style-type: none"> • Perform unit testing and system testing • Identify bugs in the system 	<ul style="list-style-type: none"> • Test results • Fix bugs
Deployment and Maintenance	<ul style="list-style-type: none"> • Deploy the system for end users • Track system performance • Improve the system based on user feedback 	<ul style="list-style-type: none"> • Deployed system • System monitor reports • Updated features

3.1 System Requirement

System requirement analysis plays a vital role in assessing the suitability of the proposed Web-Based Café Hopper Platform to customers and stakeholders. This section identifies and documents functional and nonfunctional requirements necessary for the platform's development. By evaluating these requirements, the project guarantees that the system has desired characteristics like easy café search and reservation, besides compliance with performance, security, and usability standards. This analysis is a strong basis for designing and building a highly reliable and growing platform. Table 3 and Table 4 show the functional and non-functional requirements.

Table 3 *Functional Requirements*

Module	Description
1. User Registration and Login	<ul style="list-style-type: none"> • User may register and log in to the system by using email and password • The system will be directly on the homepage after successfully accessing the login page
2. Profile Management	<ul style="list-style-type: none"> • Users can update their profiles • Café owners can update their profiles with information such as opening and closing hours, pictures and contact details.
3. Café Search and Filtering	<ul style="list-style-type: none"> • Users can search cafes based on preferences, ratings and location
4. Booking System	<ul style="list-style-type: none"> • Users can make, view, or cancel reservations based on the café's availability
5. Reviews and Rating	<ul style="list-style-type: none"> • Users can leave reviews and ratings for cafes after the visit

6. Admin Management	<ul style="list-style-type: none"> Admin can manage user’s accounts, review café verification requests and monitor system performance
7. Analytics Reporting	<ul style="list-style-type: none"> Café owners can view analytics such as ratings, reviews and booking statistics to improve their business strategies.

Table 4 Non-Functional Requirements

No.	Requirement	Description
1.	Usability	The platform must be user-friendly and responsive design for both mobile and desktop devices.
2.	Performance	The system can handle many concurrent users without performance degradation.
3.	Scalability	The platform should be scalable to accommodate the future growth of cafes.
4.	Security	Every piece of data that belongs to a user has to be encrypted, and the operation has to follow legislation for data protection

4. Analysis and Design

The analysis and design of the Web-Based Café Hopper Platform are the main objectives of this phase. The system development process is divided into more manageable, smaller sub-activities to ensure the platform achieves its overall goals. During this stage, critical structural diagrams, such as flowcharts, entity relationship diagrams (ERD), data flow diagrams (DFD), and a context diagram, are created. For improved planning and execution, these tools aid in visualizing data flow, system operations, and component interactions.

4.1 Context Diagram

The context diagram gives a high-level summary of the communication between the Web-Based Café Hopper Platform and its users. Displaying the inputs that users provide and the outputs that the system produces demonstrates the system's limits. The external entities that interact with the platform and the information flow between them are identified in this figure. The Web-Based Café Hopper Platform's context diagram is shown in figure 2.

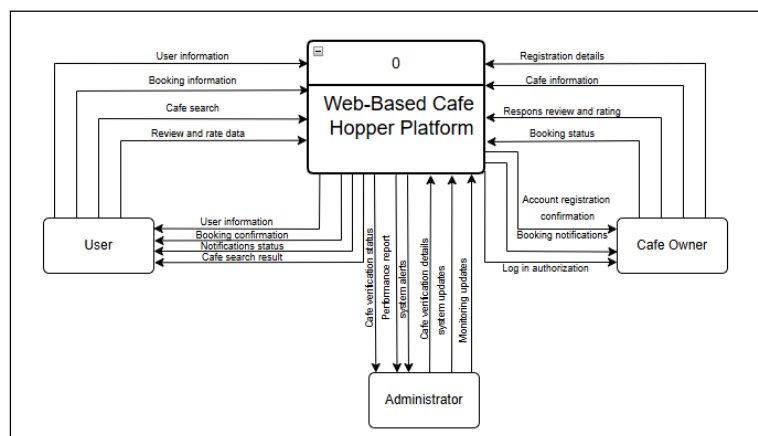


Figure 2 Context Diagram

4.2 Data Flow Diagram Level 0

An illustration of data movement through a system is called a data flow diagram (DFD). It demonstrates how incoming data from outside sources is processed and converted into output that may be saved in a data repository or transferred to another organization [5]. The inputs and outputs connected to every process and entity in the system are shown graphically in a DFD. A high-level overview of the created system's primary activities, data flows, external entities, and data storage is provided by the Level 0 Data Flow Diagram (DFD 0), which is shown in Figure 3.

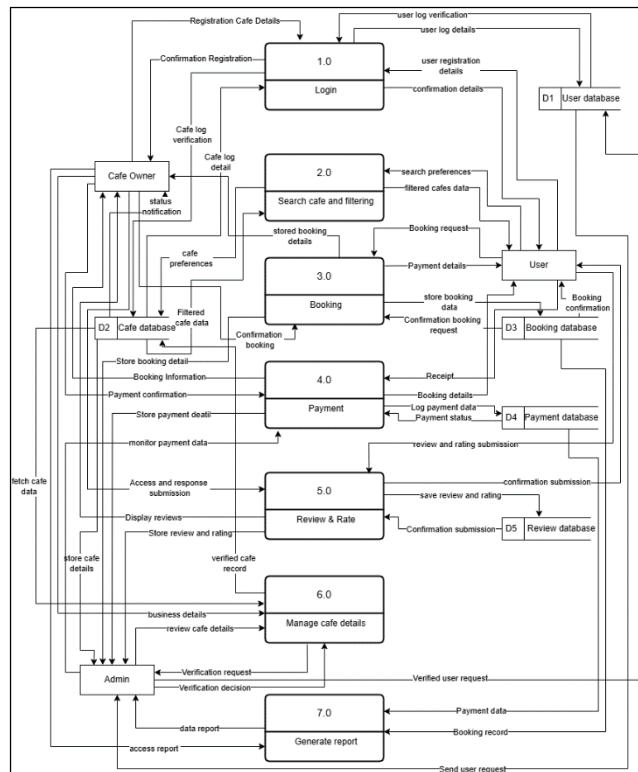


Figure 3 Data Flow Diagram Level 0

4.3 Entity Relationship Diagram

A structural diagram used in database architecture is called an Entity Relationship Diagram (ERD), often called an ER Diagram or an ER model [6]. It illustrates essential information, such as the main entities inside a system's scope and the connections between them, using a variety of symbols and connectors. The ERD for the developed system is shown in figure 4.

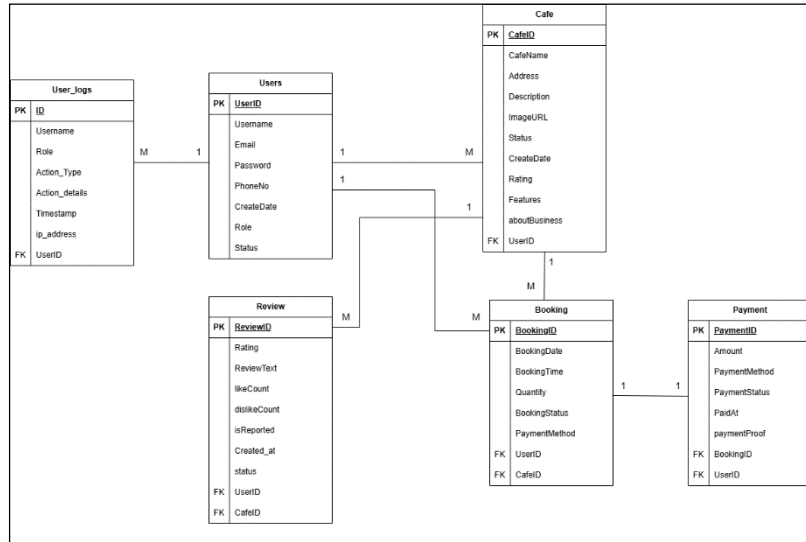


Figure 4 Entity Relationship Diagram

4.4 Flowchart

A flowchart is a graphical tool that shows the order of steps, decision points, and information flow inside a system [7]. Common flowchart symbols include rectangles for processes, diamonds for decisions, ovals for start and finish points, and arrows to indicate flow direction. The designed system's flowchart diagram for admin is shown in figure 5 while for user flowchart in figure 6 and figure 7 for café owner's flowchart.

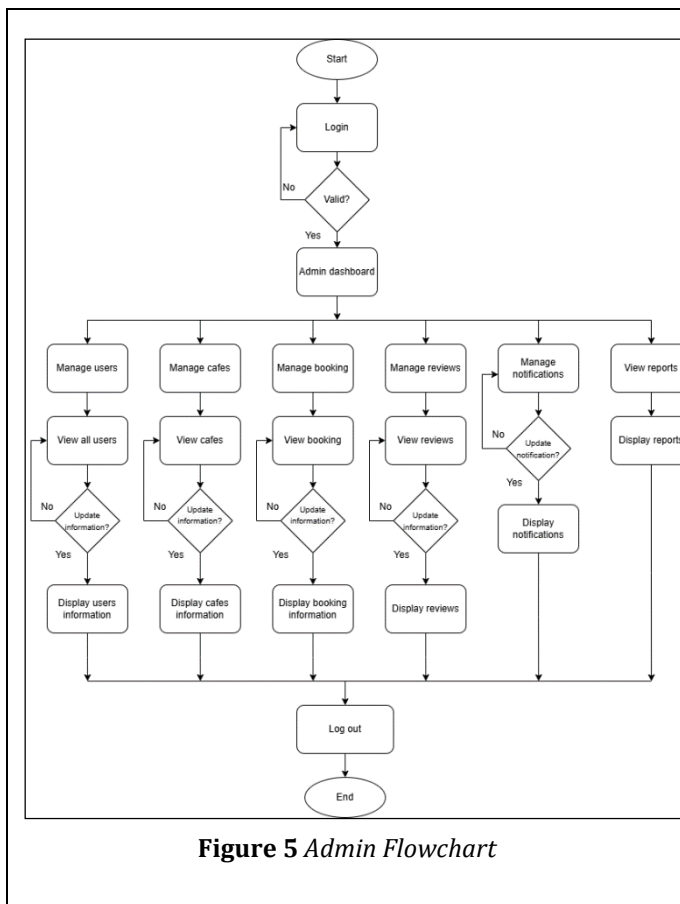


Figure 5 Admin Flowchart

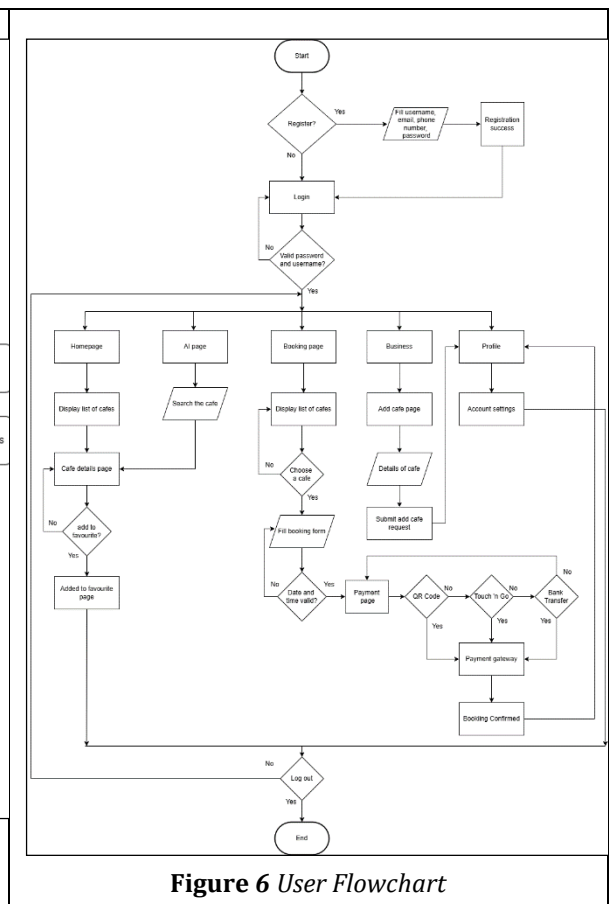


Figure 6 User Flowchart

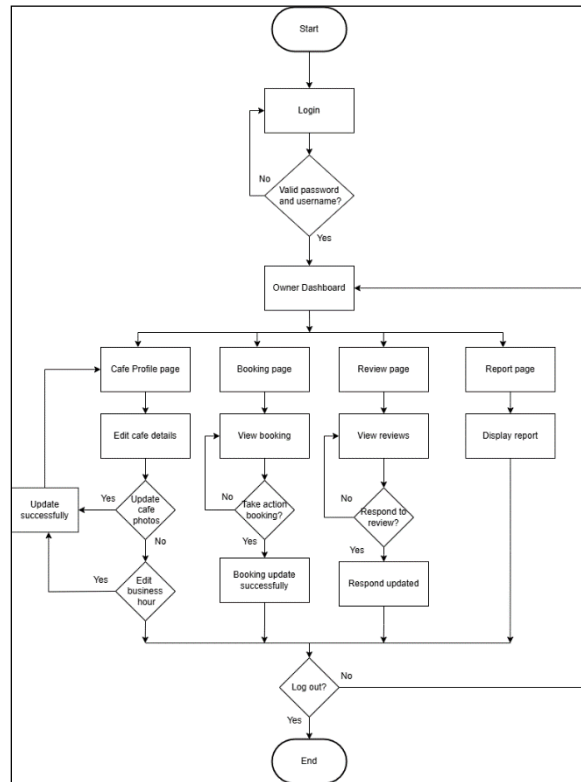


Figure 7 Café Owner Flowchart

4.5 Database Design

A methodical approach to planning, modeling, and developing a framework for effectively and permanently storing and managing data is database design [8]. This methodical procedure makes effective data retrieval and manipulation possible, which also reduces redundancy and guarantees data integrity. Transforming real-world requirements into a logical and physical data model that supports an application's data demands, preserves data integrity, and maximises performance is the core purpose of database design. The database structure for the web-based café hopper platform is shown in Figure 8.

Table	Action	Rows	Type	Collation	Size	Overhead
admin	★ Browse Structure Search Insert Empty Drop	1	InnoDB	utf8mb4_general_ci	32.0 KiB	-
booking	★ Browse Structure Search Insert Empty Drop	62	InnoDB	utf8mb4_general_ci	48.0 KiB	-
cafe	★ Browse Structure Search Insert Empty Drop	12	InnoDB	utf8mb4_general_ci	32.0 KiB	-
cafehours	★ Browse Structure Search Insert Empty Drop	84	InnoDB	utf8mb4_general_ci	32.0 KiB	-
cafeimages	★ Browse Structure Search Insert Empty Drop	32	InnoDB	utf8mb4_general_ci	32.0 KiB	-
cafeowner	★ Browse Structure Search Insert Empty Drop	13	InnoDB	utf8mb4_general_ci	48.0 KiB	-
cafevideos	★ Browse Structure Search Insert Empty Drop	14	InnoDB	utf8mb4_general_ci	32.0 KiB	-
notifications	★ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_general_ci	32.0 KiB	-
payment	★ Browse Structure Search Insert Empty Drop	23	InnoDB	utf8mb4_general_ci	48.0 KiB	-
review	★ Browse Structure Search Insert Empty Drop	1	InnoDB	utf8mb4_general_ci	48.0 KiB	-
reviewreplies	★ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_general_ci	48.0 KiB	-
saved_cafes	★ Browse Structure Search Insert Empty Drop	1	InnoDB	utf8mb4_general_ci	48.0 KiB	-
users	★ Browse Structure Search Insert Empty Drop	6	InnoDB	utf8mb4_general_ci	48.0 KiB	-
user_logs	★ Browse Structure Search Insert Empty Drop	117	InnoDB	utf8mb4_general_ci	32.0 KiB	-
14 table(s)	Sum	366	InnoDB	utf8mb4_general_ci	560 KiB	0 B

Figure 8 Database Schema

5. Result and Discussion

This section is left for the implementation of the system and user testing that was conducted for the Web-Based Café Hopper Platform to targeted users.

5.1 Implementation of Module

The last stage is implementation and testing, during which the system is put through a thorough testing process to confirm that it works as intended. User authentication is the purpose of the Registration and Login module. While current users use the login function to access their existing accounts, new users use the registration option to create an account. Figure 9 and figure 11 (b) illustrate the code and user interface of the registration process, which includes four required fields: email, username, password, and phone number. Figure 10 presents the code for the login process, while figure 11 (a) displays the corresponding login interface. For the login process, users are only required to provide their username and password.

```

<div class="signup-container">
  <!-- Logo -->
  

  <h2>Create Your Account</h2>

  <?php if (isset($error)): ?>
    <div class="error"><?php echo $error; ?></div>
  <?php endif; ?>

  <form action="sign_up.php" method="POST">
    <input type="email" name="email" placeholder="Email" required>
    <input type="text" name="username" placeholder="Username" required>
    <input type="password" name="password" placeholder="Password" required>
    <input type="tel" name="phoneNo" placeholder="Phone Number" required>
    <button type="submit">Sign Up</button>
  </form>

  <p>Already have an account? <a href="login.php">Login</a></p>
</div>

```

Figure 9 Coding for registration

```

<div class="login-container">
  <!-- Logo -->
  

  <h2>Welcome Back to Cafe Hopper</h2>

  <?php if (isset($error)): ?>
    <div class="error"><?php echo htmlspecialchars($error); ?></div>
  <?php endif; ?>

  <form action="login.php" method="POST">
    <input type="text" name="username" placeholder="Username" required />
    <input type="password" name="password" placeholder="Password" required />
    <button type="submit">Login</button>
  </form>

  <form action="login.php" method="POST">
    <button type="submit" name="login_guest">Login as Guest</button>
  </form>

  <p>Don't have an account? <a href="sign_up.php">Sign up</a></p>
</div>

```

Figure 10 Coding for login

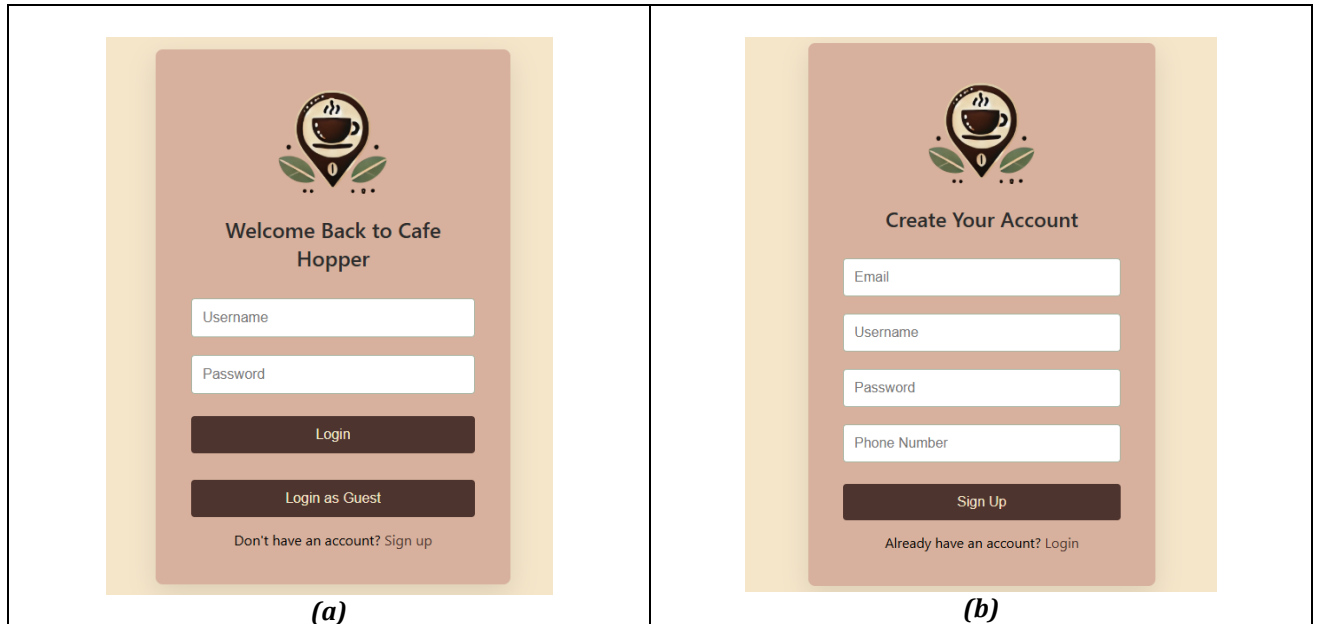


Figure 11 (a) Login page interface; (b) Registration page interface

The Web-Based Café Hopper platform displays all available cafés, allowing users to browse and book tables easily. As shown code in figure 12, the list of cafés becomes visible once users have successfully logged into the platform like figure 13. Users can view details such as the café's address, rating, features, operating hours, and a booking button for reservations.



Figure 12 Coding for the homepage

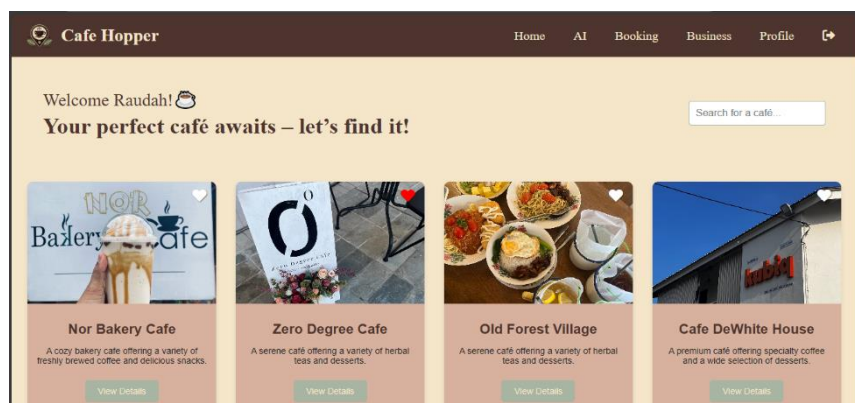


Figure 13 Homepage interface

In the booking module, users are required to complete a form with the booking date, time, quantity, and any additional notes. Upon submission, they are redirected to the payment page. Figure 14 shows the code for the booking page, while figure 15 displays the booking form interface.

```
<div class="container">
  <h2>Reserving Your Spot at <?php echo htmlspecialchars($cafe['cafeName']); ?> !</h2>
  <p><?php echo htmlspecialchars($cafe['description']); ?></p>
  

  <form action="process_booking.php" method="POST">
    <input type="hidden" name="cafe_id" value="<?php echo $cafeID ?>">

    <label for="booking_date">Select Date:</label>
    <input type="date" id="booking_date" name="booking_date" required>

    <label for="booking_time">Select Time:</label>
    <input type="time" id="booking_time" name="booking_time" required step="60">

    <label for="quantity">Number of Guests:</label>
    <input type="number" id="quantity" name="quantity" min="1" max="20" required>

    <label for="notes">Additional Notes (optional):</label>
    <textarea id="notes" name="notes" placeholder="Any specific request?"></textarea>

    <button type="submit">Confirm Booking</button>
  </form>
</div>
```

Figure 14 Coding for booking table in café

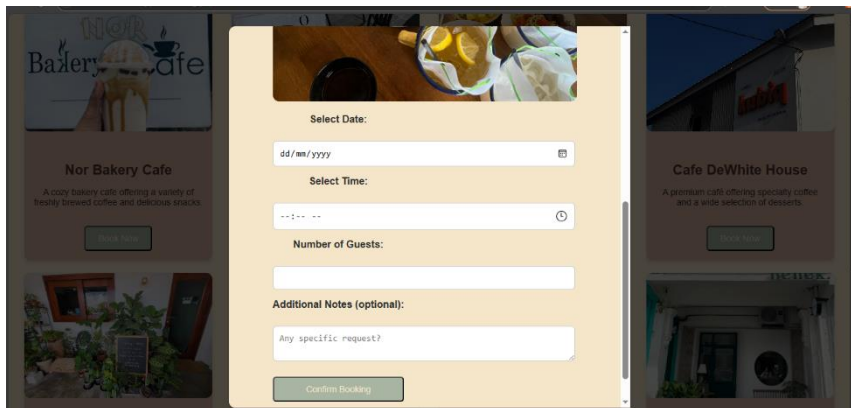


Figure 15 Booking page interface

Next, after completing the booking, users are directed to the payment page, where they can choose from several payment options such as QR Code, Touch 'n Go, and Bank Transfer. The code implementation is shown in figure 16 (a), while figure 17 (a) also displays the interface of the payment options. Once a payment method is selected, users are redirected to the payment gateway, as illustrated in figure 17 (b). The gateway displays the payment details and prompts users to upload proof of payment before clicking the 'I've Paid' button to complete the process.

```
<head><title>Choose Payment Method</title></head>
<body>
  <h2>Payment for Booking ID: <?php echo htmlspecialchars($bookingID); ?></h2>
  <form action="gateway.php" method="POST">
    <input type="hidden" name="booking_id" value="<?php echo htmlspecialchars($bookingID); ?>">

    <label>Select Payment Method:</label><br>
    <input type="radio" name="method" value="qr" required> QR Code<br>
    <input type="radio" name="method" value="tng"> Touch 'n Go<br>
    <input type="radio" name="method" value="bank"> Bank Transfer<br><br>

    <button type="submit">Proceed to Payment</button>
  </form>
</body>
```

(a)

```

<body>
<div class="container">
<h2>Pay with <?php echo strtoupper(htmlspecialchars($method)); ?></h2>
<p class="amount">Total: <strong>RM <?php echo number_format($amount, 2); ?></strong></p>

<?php if ($method == 'qr'): ?>
<p class="instructions">Scan this QR to pay:</p>

<?php elseif ($method == 'tng'): ?>
<p class="instructions">Scan this QR to pay via Touch 'n Go:</p>

<?php else: ?>
<p class="instructions">Transfer to RHB Bank:</p>
<p class="amount"><strong>1 51021 0042707 7</strong> <br> (Raudah Ezatul)</p>
<?php endif; ?>

<form action="confirm_payment.php" method="POST" enctype="multipart/form-data">
<input type="hidden" name="booking_id" value="<?php echo htmlspecialchars($bookingID); ?>">
<input type="hidden" name="method" value="<?php echo htmlspecialchars($method); ?>">

<label for="proof">Upload Payment Proof (required):</label><br>
<input type="file" name="proof" id="proof" required><br><br>

<button type="submit" class="btn-confirm">I've Paid (Confirm)</button>
</form>

```

(b)

Figure 16 (a) Coding for payment options; (b) Coding for payment gateway

(a)

(b)

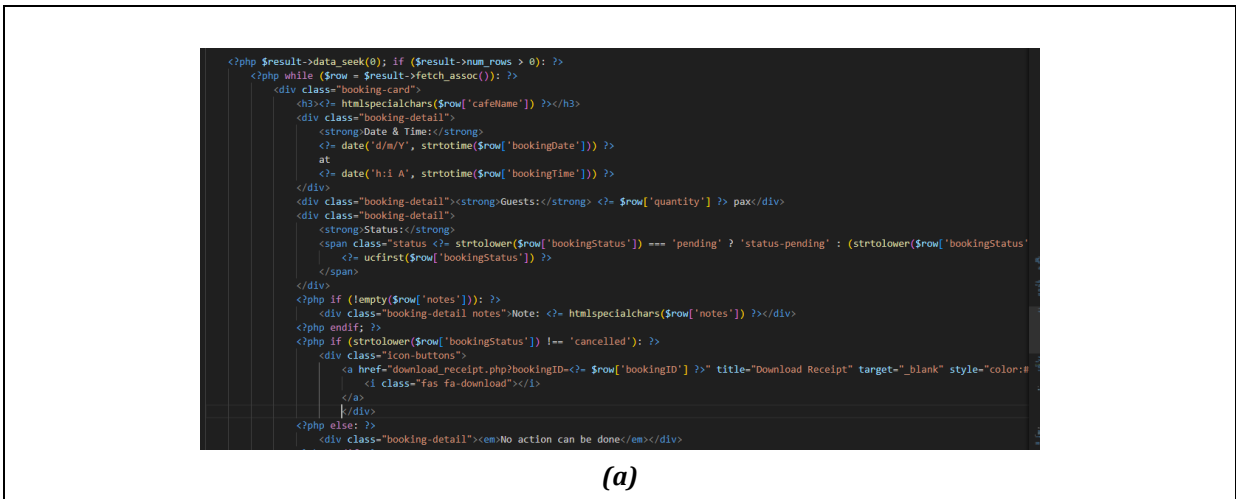
Figure 17 (a) Interface for payment options; (b) Interface for payment gateway

If a user wants to become a café owner, they can click on the 'Business' page, which redirects them to the 'Add Café' form. The form requires users to provide details such as the café name, address, description, features, business information, an image, a video, and the operating hours. Once completed, users can click the 'Request to Add Café' button to submit the form for admin review. The admin will then either approve or reject the café request. Figure 18 (a) shows the code for the 'Add Café' form, while Figure 18 (b) displays the form interface.



Figure 18 (a) Coding for add cafe; (b) Interface for add café

Next is the Booking History section, which displays a calendar highlighting all past and upcoming bookings. Below the calendar, bookings are also presented in a list view, where users can apply filters to search specific records. Each booking is displayed in a card format containing relevant details. Users can also download a receipt as proof of booking, as shown in Figure 19 (c), which illustrates the PDF receipt. Figure 19 (a) shows the code for the Booking History feature, while Figure 19(b) displays its user interface.



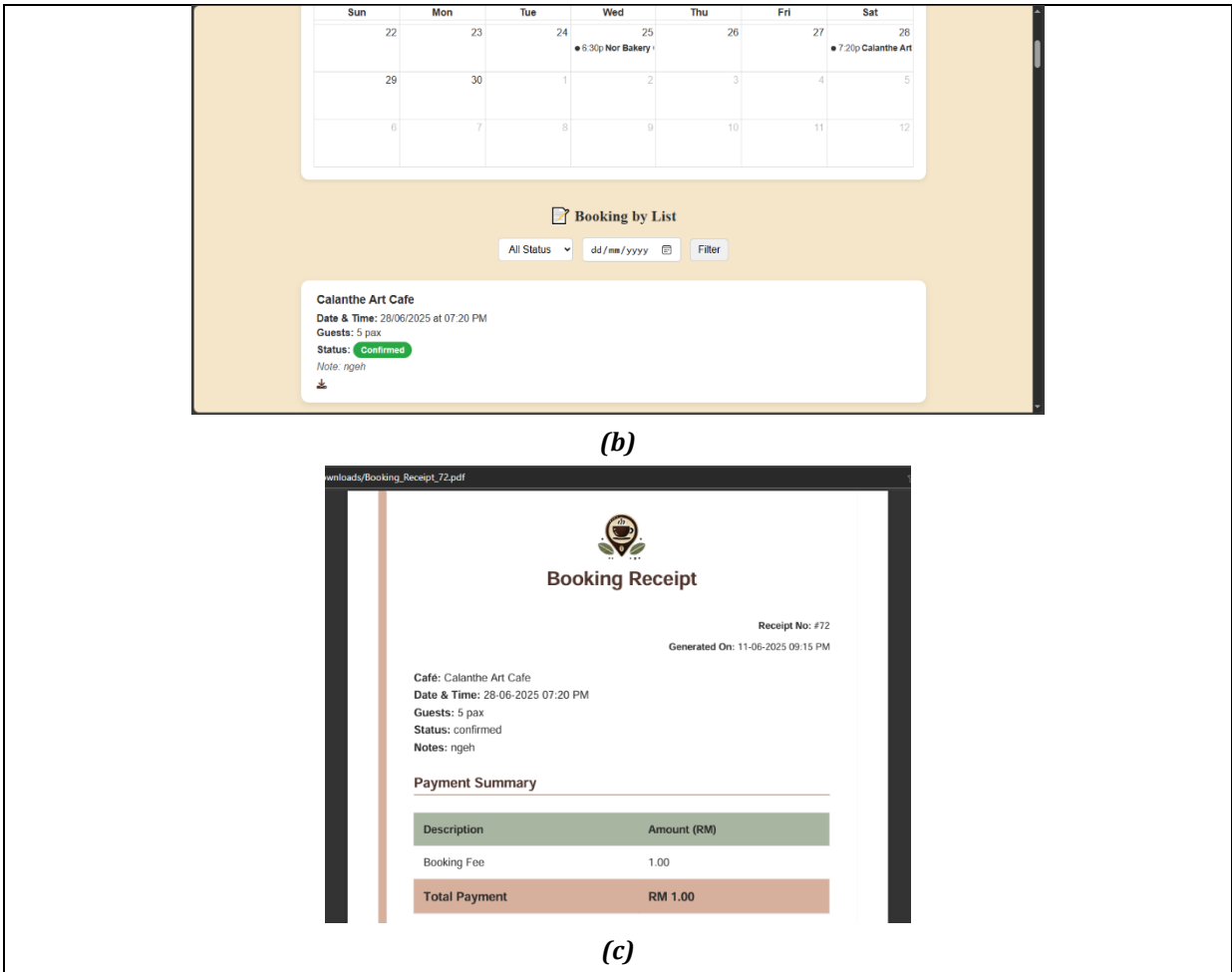


Figure 19 (a) Coding for booking history; (b) Interface for booking history; (c) PDF of receipt

Next, the admin manages the entire system through the admin panel, which provides control over users, cafés, bookings, reviews, and report generation. Figure 20(a) shows the code for the admin dashboard, while figure 20(b) displays its user interface. From the dashboard, the admin can perform various actions such as viewing, editing, and deleting records.



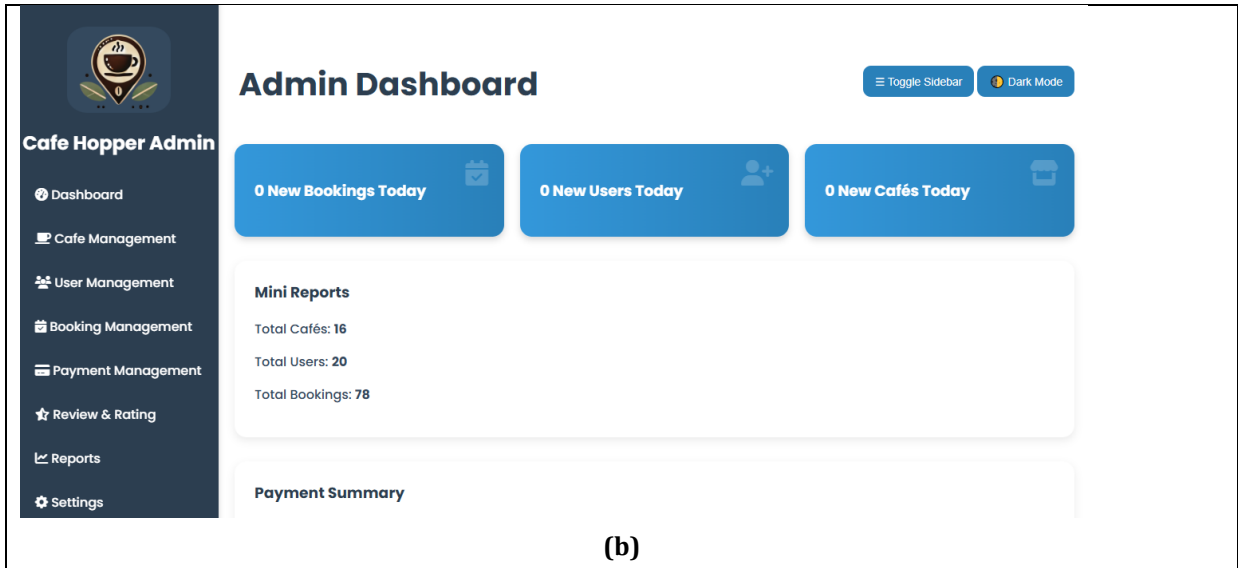
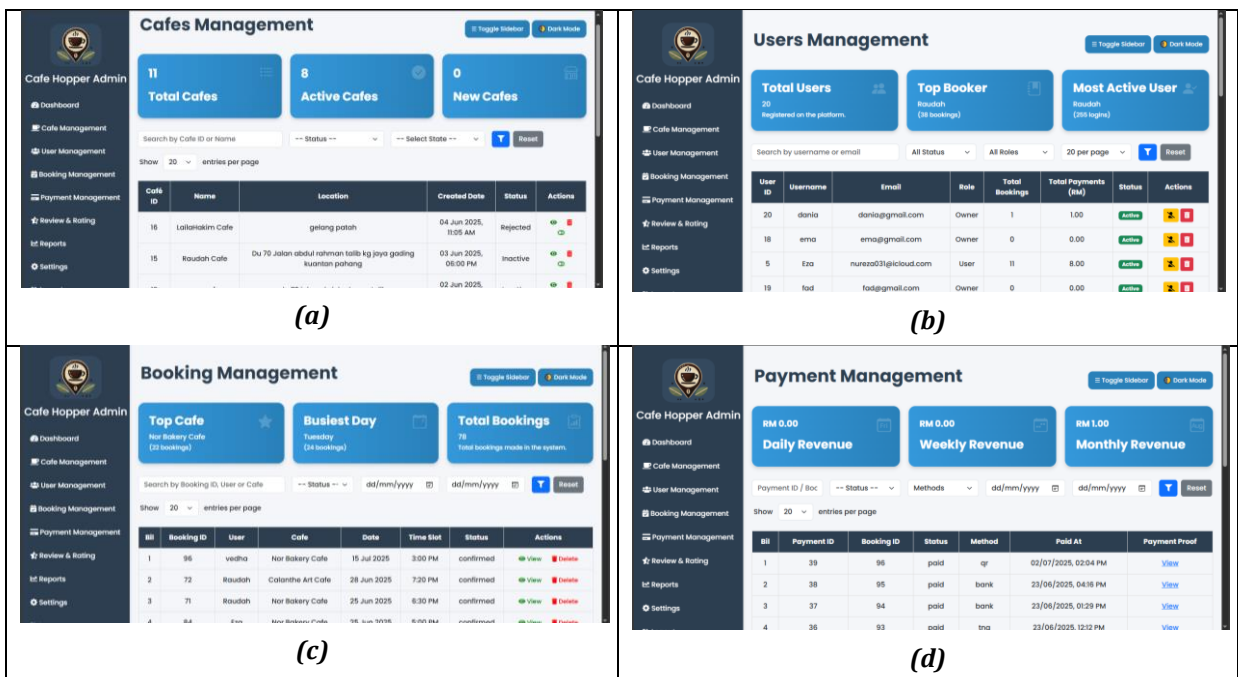


Figure 20 (a) Coding for admin dashboard; (b) Interface of admin dashboard

Next, the admin is provided with various management interfaces to ensure smooth platform operations. As shown in Figure 21(a), the café management interface allows the admin to add, update, enable, or disable cafes, making it easy to control which cafes are visible to users. In Figure 21(b), the user management interface enables the admin to perform actions such as activating, reactivating, or deleting user accounts, helping to maintain user control and platform security. Figure 21(c) presents the booking management section, where the admin can view and manage all user bookings, track booking details, and handle any related issues. Lastly, Figure 21(d) displays the payment management interface, which allows the admin to monitor payment status, verify transactions, and ensure that all completed bookings are properly recorded. In Figure 21(e), the review and rating interface is shown, where the admin can monitor user feedback and take action on inappropriate or spam reviews. Lastly Figure 21(f) shows the activity reports interface, which provides summarized analytics and system usage reports to help the admin evaluate platform performance and user behavior trends.



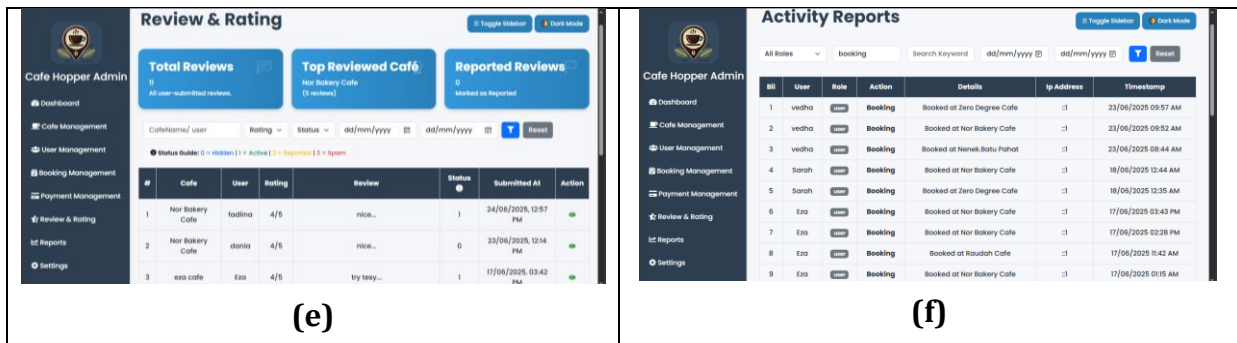


Figure 21 (a) Interface of café management; (b) Interface of users management; (c) Interface of booking management; (d) Interface of payment management; (e) Interface of review and rating management; (f) Interface of activity reports

5.2 Functional Testing System

Testing is done in this stage to assess each module's functioning. The tests are conducted using the User Acceptance Testing (UAT) methodology. Table 5 provides a summary of each module's functional testing outcomes.

Table 5 Functional Test

No	Function Testing	Expected Result	Result
1.	Registration and Login Module		
	System displays registration and Login form	Users can see the registration and login form	Success
	Login with correct credentials	Redirect to homepage/dashboard	Success
	Login with incorrect credentials	Error message shown	Success
2.	Profile Management		
	Update profile details	Profile updated successfully	Success
	Change password	New password updated	Success
3.	Café Search and Filter		
	Search for café by name	Matching cafes displayed	Success
	Search cafés by rating	Filtered results displayed	Unsuccessful
4.	Booking System		
	Search cafes by place	Filtered results displayed	Unsuccessful
	Book a café table with valid details	Booking saved and redirect to payment page	Success
5.	Try booking outside operating hours	Warning message shown	Success
	Review and Rating		
6.	Submit review and rating after booking	Review can be access to user	Success
	Admin Management		
7.	Approve/Reject café requests	Café status updated	Success
	View, edit and change status user accounts	User records updated successfully	Success
Report Analytic			
	Generate booking and user logs reports	Report generated with correct data	Success
	Export reports as PDF	File downloaded in PDF	Unsuccessful

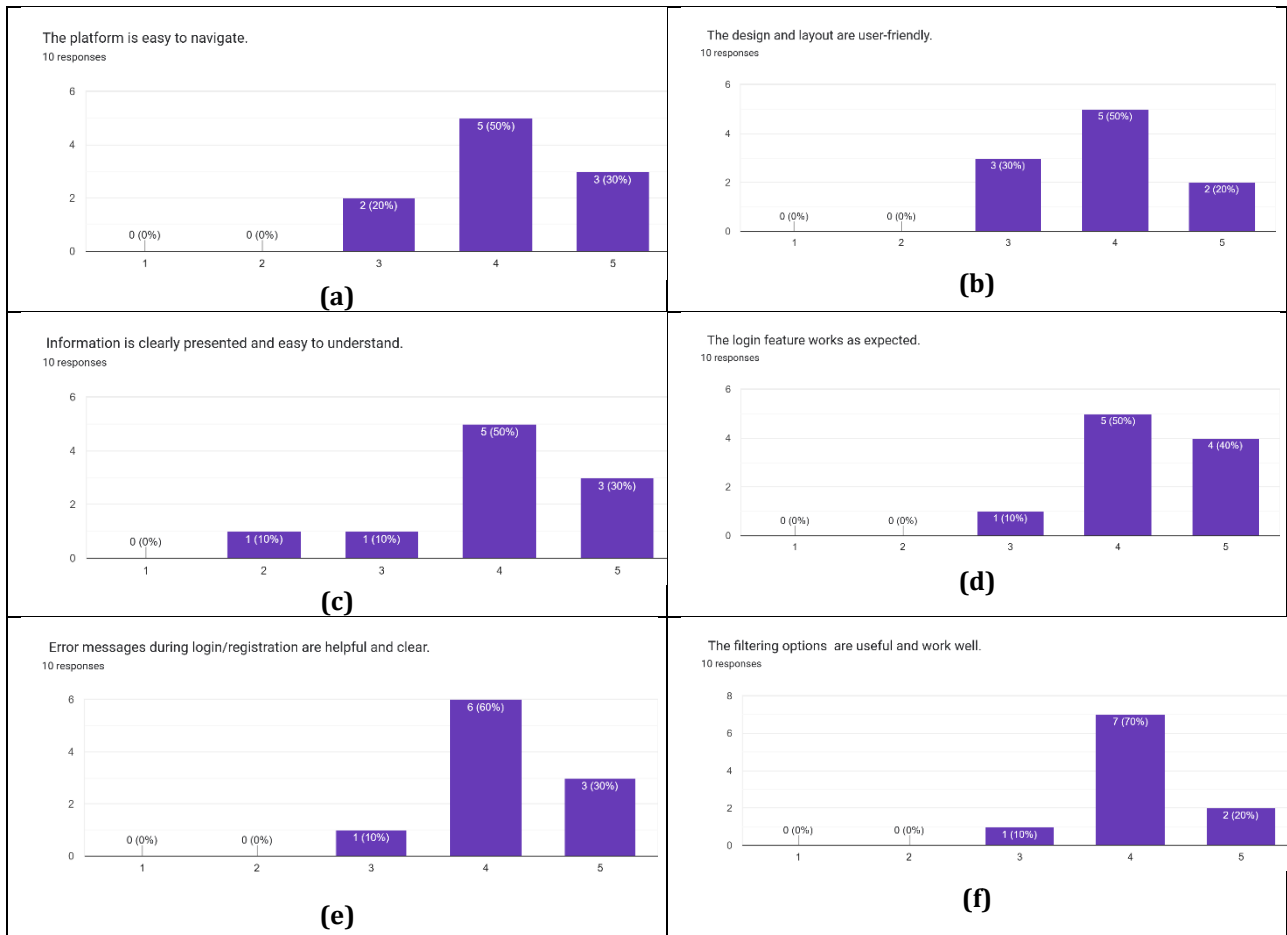
Overall, every module and function in this system performed as expected. Table 6 presents the comprehensive results for all modules.

Table 6 Functional test result

No	Module Testing	Result	(%)
1.	Registration and Login Module	Successful	100
2.	Profile Management	Successful	100
3.	Café Search and Filter	Successful	60
4.	Booking System	Successful	100
5.	Review and Rating	Successful	100
6.	Admin Management	Successful	100
7.	Report Analytics	Successful	70

5.3 User Acceptance Testing

The system's ability to satisfy user requirements and expectations was assessed using User Acceptance Testing (UAT). The goal of this phase was to evaluate how satisfied users were overall with the design and functionality of the system. Stakeholders and chosen users took part in the testing process to assess different system features. Figure 22, which displays user comments and testing process insights, summarizes the input gathered. According to the findings, ten users reported being satisfied with the system's functionality, performance, and general use throughout the assessment phase.



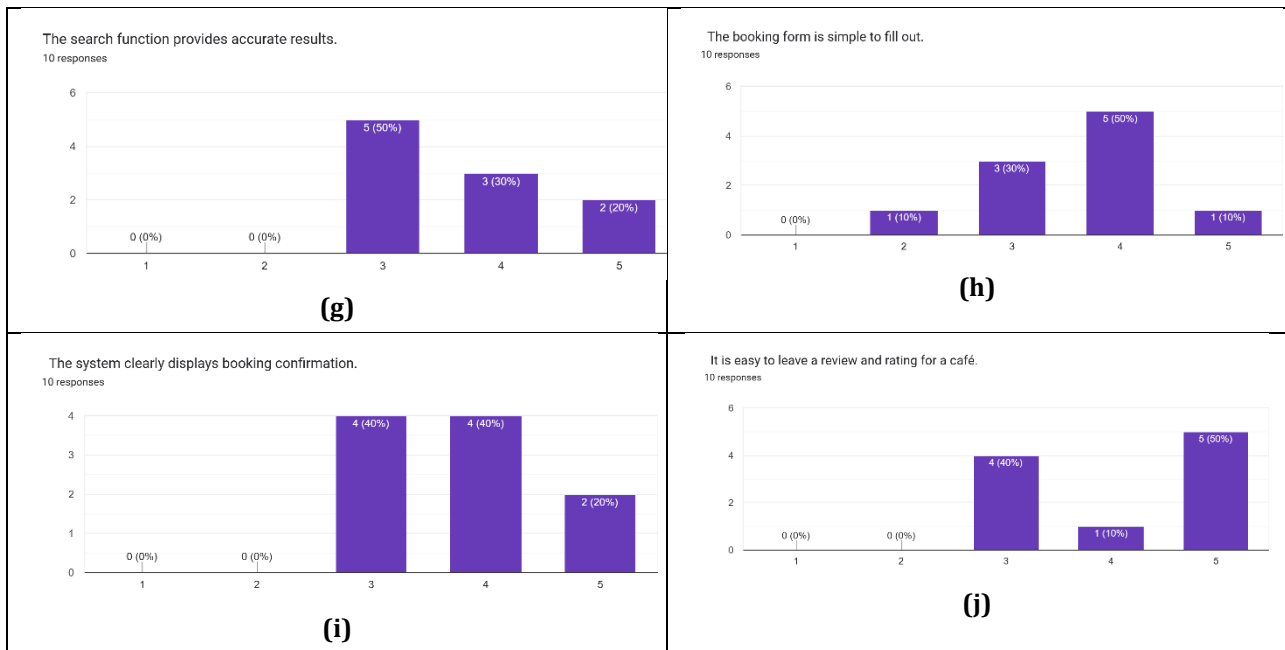


Figure 22 User Acceptance Testing Result

6. Conclusion

In conclusion, the Web-Based Café Hopper Platform serves to assist café hoppers in finding new and relevant cafés while at the same time increasing the independent cafés' prominence. Ease of use, efficiency, clear and understandable interface, efficient and fast work with large amounts of user, café owner, and administrator information. These include café search and filtering, café booking mechanisms, payments, café review, and café administration options that enhance consumer and café owner satisfaction. The system design here features well-coordinated data flows, while data-sharing processes between different entities are also effectively communicated throughout the platform. Employing the relational schemas for the database tables and developing user-friendly interfaces is sufficient to make the platform easily understandable to as many people as possible. In the end, the layouts and capabilities of the platform are primarily designed to enhance the qualitative interaction between the cafe-lovers and the cafe-owners.

6.1 Advantages

Both customers and café proprietors may benefit from the Web-Based Café Hopper System in a number of ways. Through an easy and user-friendly interface, users can easily find cafés, explore their features, browse business hours, and make reservations. Users may also evaluate their favourite cafés, manage their profiles, and keep track of their booking history on the portal. The system offers strong management features for administrators, such as analytics reporting, café approval, user control, and system monitoring. The solution makes independent cafés more visible and streamlines the decision-making process for patrons by centralising all pertinent data.

6.2 Limitation

The system still has many drawbacks in spite of its effectiveness. The accuracy of booking timeframes may be impacted by the incomplete implementation of real-time availability updates. For more individualised suggestions, the search and filtering features might be enhanced. Furthermore, the system does not presently integrate with a real payment gateway; instead, it employs a simulated booking and payment procedure. Additionally, the platform is exclusively web-based and does not have a mobile version, which may restrict mobile users' access.

6.3 Future Works

Future developments might greatly increase the system's efficacy and usefulness. Real-time transactions would be possible by integration with a trustworthy and safe online payment gateway. One way to increase accessibility and user engagement is to create a mobile application version. Implementing sophisticated search filters, tracking availability in real-time, making AI-based café suggestions, and accommodating multilingual customers are

further possible capabilities. To facilitate better business choices, administrative features can also be extended to incorporate advanced analytics and automated reporting.

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

The authors confirm contribution to the paper as follows: **study conception and design:** Nur Raudah Ezatul Md Sarif, Suhaimi Abd Ishak; **data collection:** Nur Raudah Ezatul Md Sarif; **analysis and interpretation of results:** Nur Raudah Ezatul Md Sarif, Suhaimi Abd Ishak; **draft manuscript preparation:** Nur Raudah Ezatul Md Sarif. All authors reviewed the results and approved the final version of the manuscript.

References

- [1] The Star. (2024, May 25). Café-hopping culture is a lifestyle now. *The Star*. Retrieved from <https://www.thestar.com.my/metro/metro-news/2024/05/25/caf-hopping-culture-a-lifestyle-now>
- [2] Schaefer, M. (2021, June 17). Smartphones and contactless service drive coffee shops' post-COVID evolution. *Nation's Restaurant News*. Retrieved from <https://www.nrn.com/beverage-trends/smartphones-contactless-servicedrive-coffee-shops-post-covid-evolution>
- [3] Sturm, R., Pollard, C., & Craig, J. (2015). Introduction to hybrid IT and cloud computing. In *Application performance management* (pp. 101-118). Elsevier. <https://doi.org/10.1016/B978-0-12-804018-8.00007-3>
- [4] Sturm, R., Pollard, C., & Craig, J. (2017). Managing web-based applications. In R. Sturm, C. Pollard, & J. Craig, *Application performance management (APM) in the digital enterprise* (1st ed., pp. 157-180). Elsevier. <https://doi.org/10.1016/B978-0-12-804018-8.00007-3>
- [5] E. C. Daraojimba, C. N. Nwasike, A. O. Adegbite, C. A. Ezeigweneme, and J. O. Gidiagba, "Comprehensive Review of Agile Methodologies in Project Management," *Comput. Sci. & IT Res. J.**, vol. 5, no. 1, pp. 190–218, Jan. 2024. doi:[10.51594/csitrj.v5i.717](https://doi.org/10.51594/csitrj.v5i.717)
- [6] Davis, W. S. (2019). Data flow diagrams. In *The Information System Consultant's Handbook* (pp. 175-188). CRC Press.
- [7] Frantiska, Jr, J., & Frantiska, J. (2018). Entity-relationship diagrams. *Visualization Tools for Learning Environment Development*, 21-30.
- [8] Weng, W. (2024). Flowchart. In *A Beginner's Guide to Informatics and Artificial Intelligence: A Short Course Towards Practical Problem Solving* (pp. 13-20). Singapore: Springer Nature Singapore
- [9] R. Elmasri and S. B. Navathe, *Fundamentals of Database Systems*, 7th ed. Pearson, 2017.