

# Membakeries: AI-Powered Custom Cake Ordering System

Nurulaina Nisa Sahanuan<sup>1</sup>, Mohd Hamdi Irwan Hamzah<sup>1\*</sup>

<sup>1</sup> *Fakulti Sains Komputer dan Teknologi Maklumat,*

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

\*Corresponding Author: [hamdi@uthm.edu.my](mailto:hamdi@uthm.edu.my)

DOI: <https://doi.org/10.30880/aitcs.2025.06.02.055>

## Article Info

Received: 15 June 2025

Accepted: 20 November 2025

Available online: 30 November 2025

## Keywords

Online ordering system, Cake ordering system, Web-based, Mobile application, AI-Image generator

## Abstract

The rapid adoption of digital technology has transformed customer communication, prompting businesses to embrace online platforms for improved efficiency and accessibility. Membakeries, a neighbourhood bakery specializing in personalized cakes and desserts, faces challenges with its manual ordering process, leading to inefficiencies, limited customer reach, and potential errors. This project aims to develop a robust online ordering system comprising a web-based platform and a mobile application. Using the Scrum Agile methodology, iterative development and continuous feedback were implemented across initiation, planning, implementation, and refinement phases. The web platform was built with the Laravel Framework using the Model-View-Controller (MVC) pattern and PHP, while the mobile application employed Flutter in Android Studio. Modules include catalogue management, order processing, secure payments, and real-time order tracking. The resulting system enhances operational efficiency, ensures scalability, and elevates customer satisfaction, positioning Membakeries competitively in the online bakery market with sophisticated customization and seamless user experiences.

## 1. Introduction

Membakeries is a charming local bakery located in Kulai that specializes in a variety of cakes, pastries and desserts. The bakery is known for its beautiful crafted cakes that can be customized and requested by customers especially for special occasions like birthday and weddings. The current ordering process for this bakery is using the manual system whereas the customers have to either walk-in into the bakery or contact the admin through WhatsApp. Customers would create order and even make some cake customizations towards the admin. The admin would then write them down on some notes or save it in their Telegram Files to manage the order. There is no software system for them to manage and store the data of their orders. All of the information about the order quantity, delivery dates, and customers' information are stored in the Telegram Files. Fig.1 shows the current manual system of Membakeries.

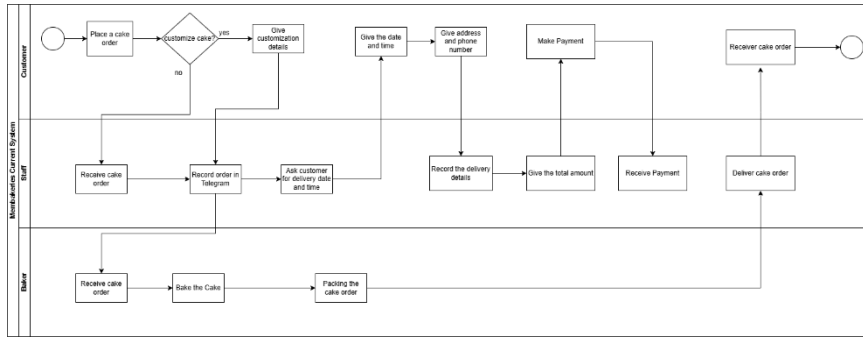


Fig. 1 Swimlane diagram of the Membakeries current system

Despite its growing popularity in the local community, Membakeries does not have any online system for customers to browse the menu and place orders anytime they want. Customers need to either visit the bakery or call the admins for special order. The absence of a system limits its reach especially for customers who might prefer the convenience of browsing the menu and placing orders online. It is not only inconvenience for the customers but also the admins or sellers to keep up the records of the orders. Having the orders physically written on documents would make it easily goes missing. The sales report, customers’ requested orders and delivery schedules must be disorganized due to the absence of a system.

The “Membakeries: AI-Powered Custom Cake Ordering System” is proposed and developed to address the issues coming from this bakery. This system will allow customers to easily browse the menu and place orders online from the comfort of their homes. Moreover, it will have the cake customization with AI image generator that help them to choose their own designs. With its user-friendly and well-organized design, this system would attract more customer to buy cakes or desserts from this Membakeries and increase the sales for brand visibility and growth of the Membakeries.

This system is targeted for the customers who lived quite far from the bakery’s location and want to create an online order that is accessible anytime. Key stakeholders involved in the study include Membakeries’ owner and staff, who will oversee system administration and order management. Their feedback on usability, functionality and operational needs will be instrumental in assessing the system’s success. The other stakeholders would be the customers to test and give feedback in using the online ordering dynamic web-based system of Membakeries.

The system will feature essential modules to streamline the ordering process, including login page, manage account, manage catalogue, manage order, manage cake customization, manage payment, manage delivery and manage report as shown in Table 1. Notifications for order confirmations and pick-up reminders will be automated via email and SMS. Through these features, the Membakeries online ordering system aims to provide customers with a convenient, seamless and efficient way to order their favourite cakes and desserts, enhancing the overall customer experience and expanding the bakery’s market reach.

Table 1 The list of modules of Membakeries: AI-Powered Custom Cake Ordering System

No.	Modules	Description	User
1.	Login	Log into the system by using email and password	Admin
		Enter personal information to register new account for new user	Customer
2.	Manage Catalogue	Add, view, delete and edit the products in the catalogue	Admin
3.	Manage Order	Browse and choose the product from catalogue and select quantity.	Customer
4.	Manage Cake Customization	Choose the size, flavor and design of the cake. The users can either put an image for the cake customization or use the AI image generator prepared in this module.	Customer

**Table 1** *Continued*

5.	Manage Delivery	Choose the date of delivery available in the slot. View delivery status  Set the available slot for cake booking and delivery per day Update delivery status	Customer  Admin
6.	Manage Payment	Choose type of payment Complete the payment	Customer
7.	Manage Report	Choose to view the sales report	Admin

This report would consist of numerous parts and chapters. The first chapter would outline the project's background, objectives, problem statement, scope, expected outcomes and project significant. The second chapter would explain the related terms of the Membakeries such as online ordering system and AI-image generator along with the technology used. Chapter 2 would also compare the three existing system with the Membakeries system. Next, Chapter 3 explain deeply into the system development technique and methodology from initiation phase, planning, implementation, review to release phase. The fourth chapter covers system requirements analysis, design, database design and interface design technicalities. The fifth chapter covers the implementation and testing, and sixth chapter covers overall conclusions.

## 2. Related Work

This section describes the related terms to the Membakeries AI-Powered Custom Cake Ordering System and the comparison between the existing system with the proposed system.

### 2.1 Online Ordering System

Online ordering system, also known as OOS, is a particularly sophisticated organizational-technical system that makes it easier to create, amend, send, manage, access and deliver the orders [9]. Numerous opportunities are emerging on the internet as a result of the quick rise in internet usage and related technologies. The internet has made it possible for a large number of enterprises to operate comfortably. An online system for ordering food is one of the enterprises that the internet brought about. Nowadays, a lot of business prioritize quick preparation and speedy delivery of orders. Most of these delivery orders were recently placed over the phone, either through a mobile application or website. [3]

### 2.2 Artificial Intelligence (AI) Image Generator for Cake Customization

Deep generative model-based text-to-image creation systems have gained popularity as tools for producing digital artwork and images [4]. Given an input prompt in spoken language, these generative systems are able to synthesis digital images of high aesthetic quality. A vast array of text-to-image creation systems are now open source as executable notebooks, and an ecosystem of tools and resources centered around this process has developed online. [6]

In the context of the proposed system, this AI image generator will be used in cake customization module for the customers to use it as a tool for them to describe their personalized cake designs. The provided AI image generator in the system will help generate the description of the cakes to three different images of the customer to choose. The chosen image will then be updated in the database for the order list and details.

### 2.3 Comparison with The Existing System

In compared to the current systems utilized by rivals like Sorella Café, Mamasab Bakery and Moonlight Cake House, the suggested Membakeries online ordering system offers a number of improvements. Membakeries will offer a specialized web-based platform with mobile optimization, in contrast to Sorella Café, which mostly depends on social media channels like Instagram and WhatsApp for orders. Customers may browse, configure and

place orders directly through the system using this method, which guarantees a smooth and expert user experience.

Although Mamasab Bakery uses an online purchasing platform, there aren't many possibilities for cake and dessert. The Membakeries system, on the other hand, has sophisticated customization tools, such as AI-powered image generator, that let users see and alter cake design instantly. This cutting-edge feature offers a notable advantage in user engagement and satisfaction.

Moonlight Cake House prioritizes providing a comprehensive product selection and easy-to-use navigation on its web and mobile platforms. With the addition of improved capabilities like real-time order monitoring and integrated inventory management for administrators, Membakeries seeks to match this functionality. These characteristics guarantee precise product availability for clients and effective order processing.

All things considered, the suggested Membakeries system sets itself apart by combining state-of-the-art customizing technology, reliable back-end management and contemporary, customer-focused interface, all of which solve shortcomings seen in its rivals' current systems. Table 2 shows the comparisons between existing systems and proposed systems.

**Table 2** Comparison between existing systems and proposed system

Features	Sorella Cafe	Mamasab Bakery	Moonlight Cake House	Membakeries
Register and Login Modules	√	√	X	√
Menu and Browsing Cakes	√	√	√	√
Cake Customization Form	X	X	X	√
Payment Option	√	X	X	√
Delivery and Pick-up Option	√	X	X	√
Admin cake catalogue management	√	√	√	√
Admin configuration module	√	√	√	√

### 3. Methodology/Framework

A foundation or framework for structuring, planning and carrying out the processes need in creating an information system is the Software Development Life Cycle [7] Variations on the SDLC have been created as a result of the disparity in system requirements, among many other variances. With the recent development of web-based technologies, it is crucial to ascertain which of them is most appropriate before starting a project [7]. The SDLC comprises discrete stages intended to provide software engineers with a well-defined objective for their work. It seeks to assist developers and other project personnel in developing a system that surpasses customer expectations and satisfies all technical and user requirements. Simple Software Development Life Cycle model has five phases which is analysis phase, design phase, implementation phase, testing phase and evaluation phase.

The SDLC model consists of five phases. As shown in Table 3.1, each phase has its tasks and deliverables that must be produced. In addition, the deliverables were accomplished within the given timeframes.

**Table 3** *Software Development Life Cycle (SDLC) tasks and deliverables*

Phase	Task	Output
Analysis Phase	<ul style="list-style-type: none"> <li>Identifying stakeholders, problem definition, and defining the project scope</li> <li>Interview with the stakeholders.</li> <li>Analysis of user, hardware, and software requirements.</li> </ul>	<ul style="list-style-type: none"> <li>User requirements</li> <li>Gantt Chart for the project’s timeline</li> <li>Hardware and software requirements</li> <li>Documentation of Chapter 1 to Chapter 3</li> </ul>
Design Phase	<ul style="list-style-type: none"> <li>Design the wireframe of the system</li> <li>Design database</li> <li>Design a user-friendly UI/UX</li> </ul>	<ul style="list-style-type: none"> <li>Object-oriented approach selected</li> <li>Wireframe of the system</li> <li>High-fidelity of UI/UX design</li> <li>Documentation on design</li> </ul>
Implementation Phase	<ul style="list-style-type: none"> <li>Development of core modules such as login, admin dashboard, and cake catalogue.</li> <li>Development of cake customization manage payment and shopping cart.</li> <li>Development of delivery module and report module, along with the SMS or email notification.</li> <li>Further improvements from the feedback of stakeholders.</li> </ul>	<ul style="list-style-type: none"> <li>Login, admin dashboard, and cake catalogue module</li> <li>Cake customization, payment, and shopping cart.</li> <li>Delivery, reports management, and notifications.</li> <li>Stakeholder’s feedback.</li> </ul>
Testing Phase	<ul style="list-style-type: none"> <li>Involvement of stakeholders and users for User Acceptance Testing (UAT)</li> <li>Fixing the defects found</li> </ul>	<ul style="list-style-type: none"> <li>A UAT report with comments and fixed problems</li> <li>A completed system that is prepared for public release</li> </ul>
Evaluation Phase	<ul style="list-style-type: none"> <li>Launch the system</li> <li>Making further improvement</li> <li>Project presentation</li> </ul>	<ul style="list-style-type: none"> <li>A closure report</li> <li>Final project documentation</li> <li>A fully functional system.</li> </ul>

## 4. Analysis and Design

This section describes the analysis and design of the Membakeries AI-Powered Custom Cake Ordering System from system requirement analysis, use case diagram, class diagram, database design and lastly the interface design.

### 4.1 Requirements Analysis

System requirement analysis is a declaration of functionality needed by a system in order to meet the needs of the user. To accomplish the goals of this project, the customers formulate both general and specific requirements. What the customer specifically wants and how they want it should be made clear in the statement. The system of requirements includes user, functional, non-functional, and system requirements.

#### 4.1.1 Functional Requirements

Functional requirements are a software system's input, behaviour, and outputs that enable it to complete a task [5]. Functional requirements, which define system behaviour, impact system development. Functional need enables system operation and the failure to meet the functional requirements will render the system unworkable. Table 4 shows the lists of Membakeries' modules and description.

**Table 4** *System's module and descriptions*

No.	Module	Description
1.	User Login and Registration	<ul style="list-style-type: none"> <li>Implement a secured login system with role-based access.</li> </ul>
2.	Catalogue Management	<ul style="list-style-type: none"> <li>Allow the admin to manage the catalogue by adding, deleting, and editing the details of the item in the catalogue.</li> </ul>
3.	Order Management	<ul style="list-style-type: none"> <li>Ensuring the accuracy in handling customer's orders.</li> </ul>
4.	Cake Customization Module	<ul style="list-style-type: none"> <li>Provides the AI-Image generator as an option to upload the design reference image.</li> <li>Ensuring the accuracy in handling the customer's customization orders.</li> </ul>
5.	Payment Management	<ul style="list-style-type: none"> <li>Develop an intuitive interface for customers to independently proceed with payments</li> <li>Implement a secured payment system.</li> </ul>
6.	Delivery Management	<ul style="list-style-type: none"> <li>Provide the admin to manage the schedule slots for delivery.</li> <li>Provide the customer with a user-friendly delivery interface.</li> </ul>
7.	Sales, Product and Order Reporting	<ul style="list-style-type: none"> <li>Provides the admin with the feature of generating report by week and allow the admin to choose type of reports.</li> </ul>

#### 4.1.2 Non-Functional Requirements

Non-functional requirements are important for describing the system's attributes and restrictions [10]. Typically, non-functional requirements pertain to the system's external behaviours, encompassing critical elements such as usability, security, performance, and overall quality. This investigation encompasses operational, performance, security, and usability aspects. Table 4.5 shows the Membakeries' non-functional requirements.

**Table 5** *System's non-functional requirements*

No.	Requirement	Description
1	Performance	The system should be able to function on any web browser and any mobile device.
2	Operational	The system should be available for use even after business hours.
3	Security	The system should be securely storing user login credentials, especially those of administrators using encryption techniques
4	Cultural and Political	The system should be with local regulations and international regulations regarding data privacy

5	Compatibility	The system should be compatible with a wide range of modern web browsers and new mobile devices.
---	---------------	--

### 4.1.3 User Requirement Analysis

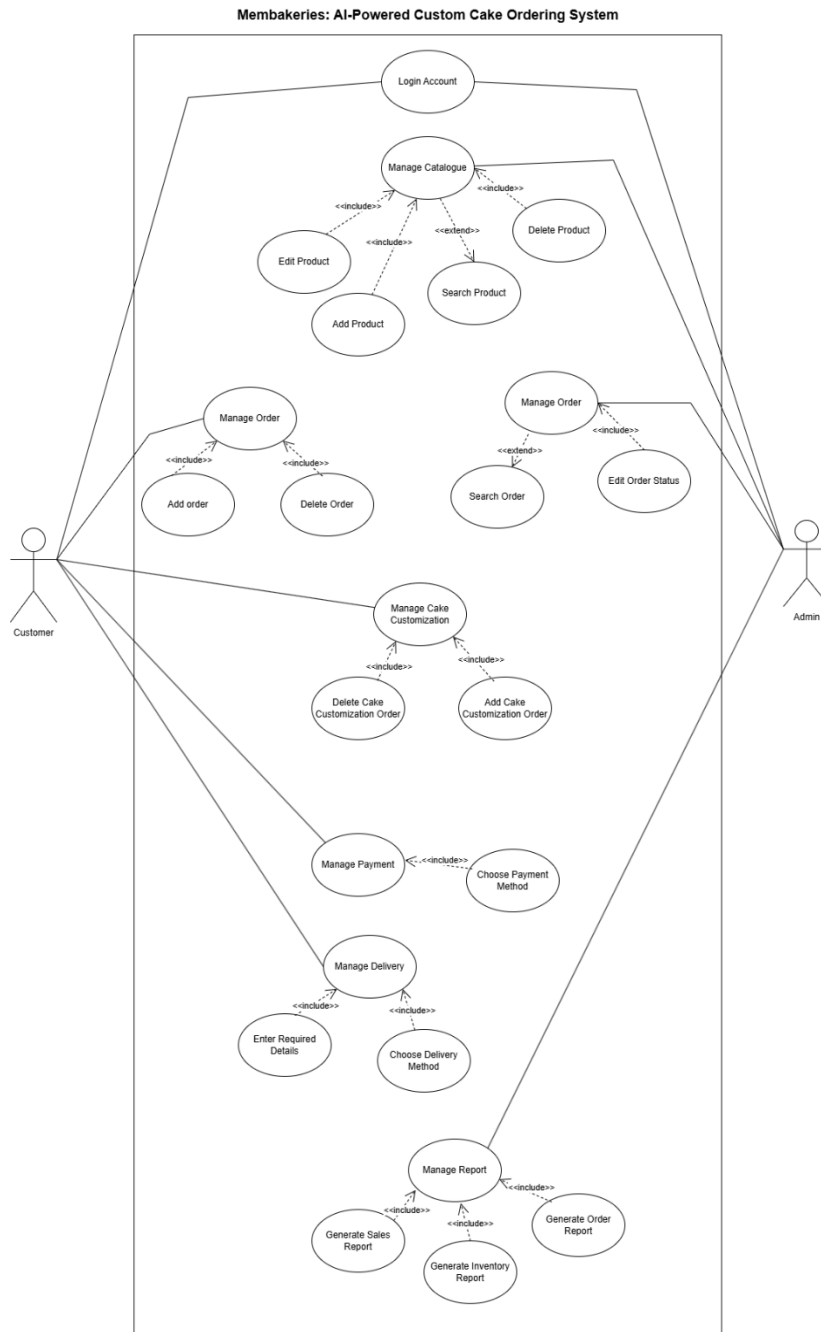
When building a unique technological system, it is critical to meet the demands and expectations of the users to ensure that the product is accepted and utilized by future users [1]. It involves gathering, documenting, and prioritizing the requirements that users have for a certain system or software program in a methodical manner. The purpose of user requirement analysis is to bridge the communication gap between developers and users, guaranteeing that the finished product meets the user's needs and solves the problems. Table 6 highlights the user requirements and the users consist of admin and customer.

**Table 6** *User requirements*

No.	User Requirements
1	Admin and customers must have a registered account to be successfully logged in with the right email and password.
2	Customers who do not have any account yet must register to the system.
3	Customers should be able to browse the menu, add item to the cart and place order, including the ability to do customization order with the feature of AI-Image generator.
4	Customers should be able to view their cart, add and delete item from the cart and check out the item from the cart.
5	Customers should be able to choose payment and delivery method.
6	Admin should be able to manage the product catalogue by adding, deleting, and editing the details of the product.
7	Admin should be able to schedule the number of slots per day for delivery.
8	Admin should be able to view the order list made by the customers.
9	Admin should be able to view all type of reports generated by the system.

### 4.2 Use Case Diagram

The different sorts of system users and their interactions can be shown in a use case diagram. This diagram is often used alongside a written use case and other diagrams [2]. The analysis included creating a use case graphic to highlight system operations. Membakeries system requirements are defined, clarified, and structured using system analysis. In Unified Modelling Language (UML), use case diagrams model real-world objects and systems. The admin and customer who execute use cases are the actors or users. The use case diagram depicts the admin and customers. The first actor which is the admin interacts with the login account, and focuses on managing the catalogue and report modules. Meanwhile, the next actor which is the customer who engages often, will log in or register the account, place an order, customize the cake, make a payment, and choose the delivery slots. Fig. 2 shows the use case diagram of Membakeries system.



**Fig. 2** Use case diagram

### 4.2 Class Diagram

Based on Fig. 3, this class diagram shows how customers, administrators, and the system's main functions interact in the Membakeries system. Customer and admin accounts are managed by the User class. CartItem objects, which stand in for products from the Product catalogue, are added by customers using a Cart. The Order class handles orders by integrating with OrderItem, Payment, and Delivery to process and complete orders. The Customization class manages cake customization choices. Using the Admin classes, administrators manage products, delivery schedule, and produce reports. From product selection to delivery, this system guarantees efficient procedures.

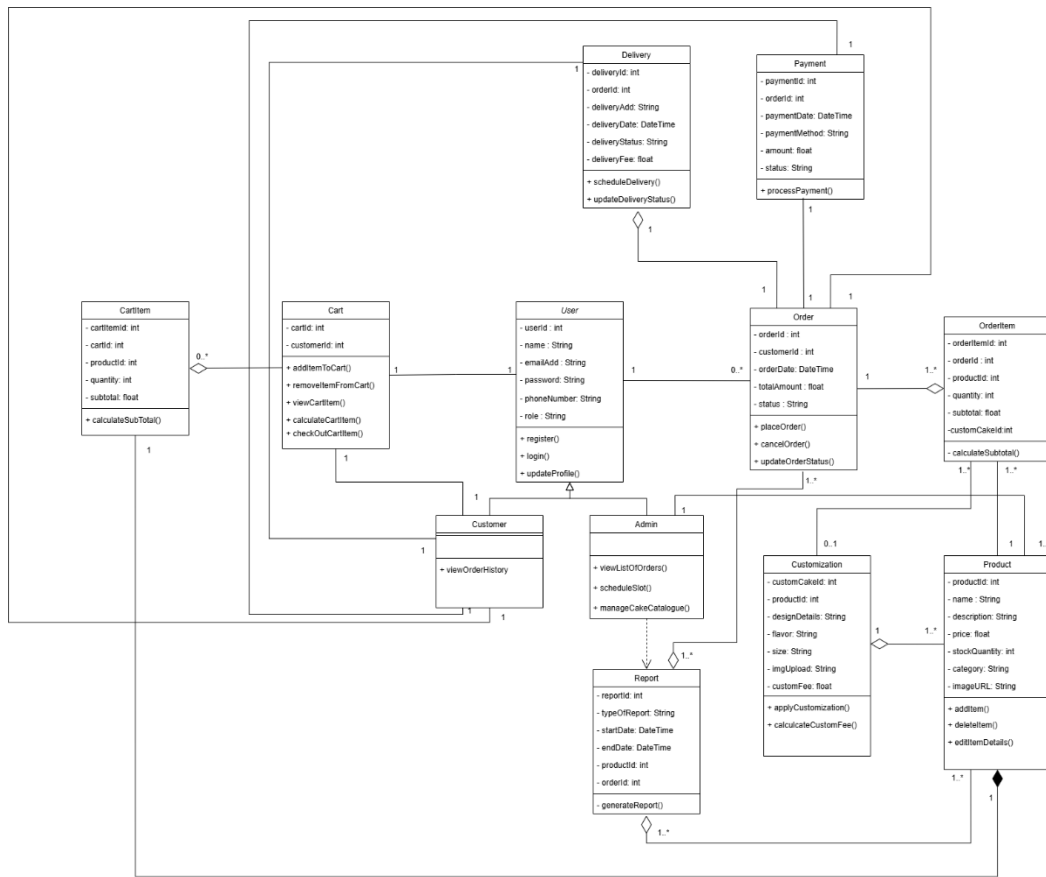


Fig. 3 Class diagram

### 4.4 Database Design

A database design is a blueprint for the organization and partitioning of data that is present in every database utilized by the system. To generate relationship schemas from the ground up, the database architecture has been formulated. Schema table is the organization and the structure of how the database in the system is constructed in the table in which each of the table has attributes that defined by the schema. The following table show all schema tables of Membakeries AI-Powered Cake Custom Ordering System database.

- i. **users** (id (PK), name, phoneNo, email, email\_verified\_at, password, role, rememberToken, created\_at)
- ii. **categories** (id(PK), name, created\_at)
- iii. **items** (id (PK), name, description, price, image\_path, created\_at, category\_id(FK))
- iv. **orders** (id (PK), created\_at, total, status, delivery\_id (FK), payment\_id (FK), user\_id (FK))
- v. **orders\_items** (id (PK), quantity, price, subtotal, created\_at, order\_id (FK), items\_id (FK))
- vi. **cart** (id (PK), created\_at, user\_id (FK))
- vii. **cart\_items** (id (PK), quantity, subtotal, custom\_details, cart\_id (FK), items\_id (FK))
- viii. **payments** (id (PK), total, method, created\_at, stripe\_id, user\_id (FK))
- ix. **deliveries** (id (PK), method, address, delivery\_datetime, user\_id (FK))

### 4.5 System Interface

The Membakeries interface design places a high value on ease of use, responsiveness, and simplicity in order to satisfy administrators and users alike. With its well-organized menus, dashboards, and visual aids, the web-based platform's design ensures smooth navigation for handling orders, reports, and catalogue. With a visually appealing catalogue display, efficient order placing, and obvious payment and delivery choices, the mobile application's design prioritizes user involvement. Fig. 6 to Fig. 14 shows some of the Membakeries system interface for both web-based and mobile applications.

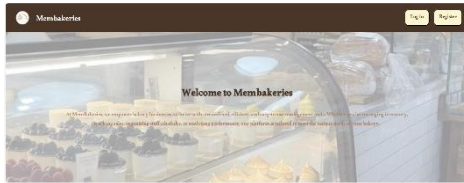


Fig. 6 Home page for admin

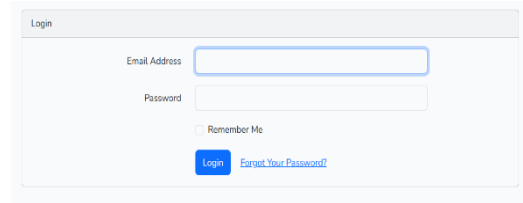


Fig. 7 Admin login interface



Fig. 8 Admin manage product interface

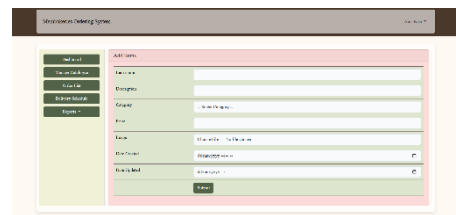


Fig. 9 Admin add new product interface

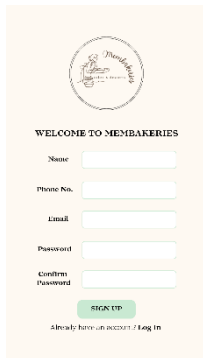


Fig. 10 Customer register interface



Fig. 11 Customer login interface

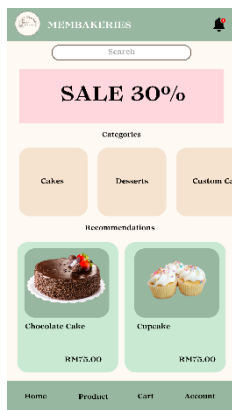


Fig. 13 Customer home page interface



Fig. 14 Customer product catalogue interface



## 5.1.2 Implementation of Manage Catalogue Module

This section presents the implementation of the Manage Catalogue module for the Admin side of the Membakeries: AI-powered Custom Cake Ordering System. Built using the Laravel framework, this module enables administrators to efficiently manage the catalogue of cake products, including adding new items, updating existing ones, and removing outdated entries. Features such as category assignment, price management, stock status, and image uploads are integrated to ensure a comprehensive and user-friendly experience. The module interacts with the system's database to reflect real-time changes, ensuring that customer-facing listings are always up to date. Fig. 18 shows the interface of the manage catalogue module, and Fig. 19 shows the code segment.



Fig. 18 Interface of Admin's manage catalogue module

```

<pre>
<code>
</code>
</pre>

```

Fig. 19 Code segment of Admin's manage catalogue module

## 5.1.3 Implementation of Manage Order Module

Based on Fig. 20 and Fig. 22, on the Admin side, developed using Laravel, the system provides a comprehensive interface to view and manage all customer orders. Admins can access a list of incoming and past orders. A key feature of this module is the ability for Admins to update the status of each order through a simple dropdown interface. This status update is reflected in real-time, enabling efficient order tracking and communication.

On the Customer side, from Fig. 21 and Fig. 23, implemented using Flutter, users can browse the product catalogue and add items to their cart. The cart system allows customers to select quantities, view total pricing, and modify or remove items before proceeding to checkout. Once the customer confirms the order and completes payment or selects cash-on-delivery, the order is submitted and stored in the backend via the PHP API. Customers can also view the status of their placed orders from within the app, ensuring transparency and engagement.

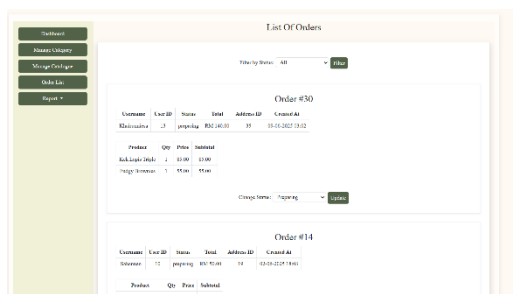


Fig. 20 Interface of Admin's manage order

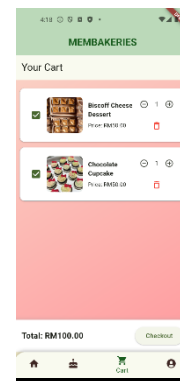


Fig. 21 Interface of Customer's Manage Order







**Table 7** Test cases and results

Test Case ID	Description	Expected Result	Actual Result	Pass/ Fail
<b>TC-01: Login Account</b>				
TC-01-001	To check whether the admin can log in with email and password	The admin should be able to log in with their email and password	The admin has successfully logged in with their email and password	Pass
TC-01-002	To check whether the admin can register a new account	The admin should be able to register a new account	The admin has successfully registered a new account.	Pass
TC-01-003	To check whether the customer can log in with their email and password	The customer should be able to log in with their email and password	The customer has successfully logged in with their email and password.	Pass
TC-01-004	To check whether the customer can register a new account	The customer should be able to register a new account	The customer has successfully registered a new account.	Pass
TC-01-005	To check whether the system shows an error for an incorrect email and password	The system should show an error message for an incorrect email and password	The system successfully shows an error message for an incorrect email and password	pass
TC-01-006	To check whether the system shows the error when the customer's confirmed password does not match the password for registration	The system should show an error message for an unmatched confirmed password and password for registration.	The system successfully shows an error message for an unmatched confirmed password and password for registration.	Pass
<b>TC-02: Manage Catalogue</b>				
TC-02-001	To check whether the admin can view a list of items on the system.	The admin should be able to view the list of items on the system	The admin can view the list of items on the system	Pass
TC-02-002	To check whether the admin can search for a specific item on the system	The admin should be able to search for a specific item on the system	The admin can search for a specific item on the system	Pass
TC-02-003	To check whether the admin can add a new item to the system	The admin should be able to add a new item to the system	The admin has successfully added a new item to the system	Pass
TC-02-004	To check whether the admin can edit the existing item on the system	The admin should be able to edit the existing item on the system	The admin has successfully edited the existing item on the system	Pass
TC-02-005	To check whether the admin can delete an item from the system	The admin should be able to delete an item from the system	The admin has successfully deleted an item from the system	Pass
TC-02-006	To verify whether the system displays an	The system should be able to show an	The system shows the error message when the admin	Pass

TC-02-007	error message when the admin fails to provide the required details. To check whether the system shows an error message when the admin adds duplicate items.	error message when the admin fails to provide the required details The system should be able to show an error message when the admin adds duplicate items.	fails to provide the required details. The system shows the error message when the admin adds duplicate items.	Pass
<b>TC-03: Manage Order</b>				
TC-03-001	To check whether the admin can see the list of orders made by the customers	The admin should be able to view the list of orders made by the customers	The admin has successfully viewed the list of orders made by the customers	Pass
TC-03-002	To check whether the admin can change the order status.	The admin should be able to change the order status	The admin has successfully changed the order status	Pass
TC-03-003	To check whether the customer can add items to the cart	The customer should be able to add items to the cart	The customer has successfully added items to the cart	Pass
TC-03-004	To check whether the customer can delete an item from the cart	The customer should be able to delete an item from the cart	The customer has successfully deleted an item from the cart	Pass
TC-03-005	To check whether the customer can check out the item from the cart	The customer should be able to check out the item from the cart	The customer has successfully checked out the item from the cart	Pass
<b>TC-04: Manage Cake Customization</b>				
TC-04-001	To check whether the customer can view the cake customization form.	The customer should be able to view the cake customization form	The customer has successfully viewed the cake customization form	Pass
TC-04-002	To check whether the customer can enter the required details	The customer should be able to enter the required details	The customer has successfully entered the required details	Pass
TC-04-003	To check whether the customer can upload an image to the system	The customer should be able to upload an image to the system	The customer has successfully uploaded an image to the system	Pass
TC-04-004	To check whether the customer can use the AI-image generator feature on the system	The customer should be able to use the AI-image generator feature on the system	The customer has successfully used the AI-image generator feature on the system	Pass
TC-04-005	To check whether the system shows an error message when the customer fails to provide the required information.	The system should be able to show an error message when the customer fails to provide the required details.	The system shows an error message when the customer fails to provide the required details.	Pass
<b>TC-05: Manage Payment</b>				
TC-05-001	To check whether the customer can choose the payment method	The customer should be able to	The customer has successfully chosen the payment method	Pass

TC-05-002	To check whether the system can provide the Stripe Card Payment.	choose the payment method The system should be able to provide the Stripe Card Payment	The system has successfully provided the Stripe Card Payment	Pass
TC-05-003	To check whether the system can accept the payment made by the customer	The system should be able to accept the payment made by the customer	The system has successfully accepted the payment made by the customer	Pass
TC-05-004	To check whether the system can generate the order invoice once the payment is successful	The system should be able to generate the order invoice once the payment is successful	The system has successfully generated the order invoice once the payment is successful	pass
<b>TC-06: Manage Delivery</b>				
TC-06-001	To check whether the customer can choose the delivery method	The customer should be able to choose the delivery method	The customer has successfully chosen the delivery method	Pass
TC-06-002	To check whether the customer can enter the required details for delivery	The customer should be able to enter the required details for delivery	The customer has successfully entered the required details for delivery	Pass
TC-06-003	To check whether the customer can pick the date and time for delivery and pickup	The customer should be able to pick the date and time for delivery and pickup	The customer has successfully picked the date and time for delivery and pickup	Pass
TC-06-004	To check whether the system shows an error message when the customer chooses a time outside of working hours	The system should show an error message when the customer chooses a time outside of working hours	The system has successfully shown an error message when the customer chooses a time outside of working hours	Pass
TC-06-005	To check whether the system shows an error message when the customer fails to provide the required details.	The system should show an error message when the customer fails to provide the required details	The system has successfully shown an error message when the customer fails to provide the required details	Pass
<b>TC-07: Manage Report</b>				
TC-07-001	To check whether the admin can view all types of reports	The admin should be able to view all types of reports	The admin has successfully viewed all types of reports	Pass
TC-07-002	To check whether the admin can choose the From Date and To Date on the system	The admin should be able to choose the From Date and To Date on the system	The admin has successfully chosen the From Date and To Date on the system	Pass
TC-07-003	To check whether the system can generate the report based on the date and time given by the admin	The system should be able to generate the report based on the date and time given by the admin	The system has successfully generated the report based on the date and time given by the admin	Pass



## 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:** Nurulaina Nisa Sahanuan, Mohd Hamdi Irwan Hamzah; **data collection:** Mohd Hamdi Irwan Hamzah; **analysis and interpretation of results:** Nurulaina Nisa Sahanuan, Mohd Hamdi Irwan Hamzah; **draft manuscript preparation:** Mohd Hamdi Irwan Hamzah. 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] Böhm, S. (2022). Do you know your user group? Why it is essential to put your user requirements analysis on a broad database. *Universal Access in the Information Society*, 909–926. doi:<https://doi.org/10.1007/s10209-021-00805-3>
- [2] Case, B. (2015). *Use Case Diagram*. Retrieved from kelas-karyawan-bali.kurikulum: <http://kelas-karyawan-bali.kurikulum.org/IT/en/2420-2301/use-case-diagrams-4557-kelas-karyawan-bali-kurikulum.html>
- [3] Chavan, V., Chavan, Jadhav, P., Jadhav, Korade, S., Korade, & Teli, P., Teli. (2015). *Implementing customizable online food ordering system using web based application* [Journal-article]. *IJISET - International Journal of Innovative Science, Engineering & Technology*, Vol. 2(Issue 4), 722–723. [https://ijiset.com/vol2/v2s4/IJISET\\_V2\\_I4\\_112.pdf](https://ijiset.com/vol2/v2s4/IJISET_V2_I4_112.pdf)
- [4] Crowson, K., Biderman, S., Kornis, D., Stander, D., Hallahan, E., Castricato, L., & Raff, E. (2022). *VQGAN-CLIP: Open Domain Image Generation and Editing with Natural Language Guidance*. arXiv (Cornell University). <https://doi.org/10.48550/arxiv.2204.08583>
- [5] Martin, M. (2023).. *What is a Functional Requirement in Software Engineering?* Guru99. <https://www.guru99.com/functional-requirement-specification-example.html>
- [6] Oppenlaender, J. (2022). The Creativity of Text-to-Image Generation. *ACM*, 192-202. <https://doi.org/10.1145/3569219.3569352>
- [7] Opralova, A. (2015). *SELECTING a DEVELOPMENT APPROACH*. *www.academia.edu*. [https://www.academia.edu/13239574/SELECTING\\_A\\_DEVELOPMENT\\_APPROACH](https://www.academia.edu/13239574/SELECTING_A_DEVELOPMENT_APPROACH)
- [8] Rouse, M.(2015). *System Requirements*. Retrieved from Techopedia.com: <https://www.techopedia.com/definition/4371/system-requirements#:~:text=Hardware%20system%20requirements%20often%20specify,peripherals%2C%20i%20any%2C%20needed>
- [9] Sahu, R. (2019). Project management for implementing an online-ordering system. *Ekonomika i Finansy (Ukraine)*, (12), 4-9.
- [10] Shreda, Q., & A. Hanani, A. (19 January, 2021). Identifying Non-functional Requirements from Unconstrained Documents using Natural Language Processing and Machine Learning Approaches. *IEEE Access*. doi:10.1109/ACCESS.2021.3052921
- [11] Yadav, N., Rajpoot, D. S., Dhakad, S. K., (2019). *LARAVEL: A PHP Framework for E-Commerce Website*. 2019 Fifth International Conference on Image Information Processing (ICIIP), 503 [https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=8985771&casa\\_token=FBeY1RPIUqUAAAAA:iUL86VtQDUJtGIvNOAO-JiD1tonUPIHrevEt2yuSULLUcf3fw7SDi5H5xAHtFFdsxNjNAv-hY&tag=1](https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=8985771&casa_token=FBeY1RPIUqUAAAAA:iUL86VtQDUJtGIvNOAO-JiD1tonUPIHrevEt2yuSULLUcf3fw7SDi5H5xAHtFFdsxNjNAv-hY&tag=1)