Mindoro State University
College of Computer Studies

"CafeEase: A Comprehensive Coffee Shop Management System for Café De Vito with QR-Based Ordering, Real-time Notifications, and Integrated Inventory Control"

A non-thesis Project
Presented to the Faculty of the
College of Computer Studies
MINDORO STATE UNIVERSITY
Calapan City Campus
Masipit, Calapan City, Oriental Mindoro

In Partial Fulfillment
of the Requirements for the Course of
APPLICATION DEVELOPMENT AND EMERGING TECHNOLOGY

by:

Calsena, Leoven Q.
Del Rosario, John Paul M.
Herrera, Adley O.
BSIT III-F1

October 2023

Mindoro State University
College of Computer Studies

TABLE OF CONTENTS

LIST OF FIGURES

Mindoro State University
College of Computer Studies

LIST OF TABLES

CHAPTER I
INTRODUCTION

In this chapter, researchers will thoroughly examine the fundamental aspects of the project, thereby establishing a solid basis for the system that lies ahead.

Project Context

In response to the dynamic landscape of the contemporary food and beverage industry, this project emerges as a strategic response to the evolving demands of the market. With the help of digital solutions and smart technologies, the "CafeEase" initiative seeks to transform the traditional coffee shop model completely. Based on a synthesis of relevant research findings and field advancements, this attempt aims to match the resulting system with state-of-the-art industry standards.

The foundational research of Singh et al. (2022), who examined the deployment of a Smart QR-based Restaurant Dine-in System, is largely responsible for the inspiration behind this project. The "CafeEase" concept is based on research into QR code technology as a means of facilitating efficient ordering procedures.

Simultaneously, the analysis and design ideas presented by Sidabutar et al. (2023) in their investigation of a Web-based Information System for Coffeeshop Management using Design Thinking are incorporated into the project's process. Using design thinking is essential to creating a user-centric interface that makes it easy and pleasant for customers and staff to interact with the coffee shop.

Demartini et al. (2018) and Manjula et al. (2021) have provided insights into the larger context of digitalization in the food business that has a substantial impact on the strategic positioning of the "CafeEase" system. With a specific emphasis on digital transformation to address operational difficulties and capitalize on future opportunities within the food retail sector, the project is strategically linked with recognized market trends.

The project's promotional strategy is informed by strategic insights from Ahamat et al. (2022) regarding innovation marketing in the food and beverage industry. The "CafeEase" system uses new technology to meet market demands and make a significant contribution to the continuous story of innovation in the larger food and beverage industry.

Furthermore, the study conducted by Daradkeh and colleagues (2023) provides a critical viewpoint, focusing on the enhancement of digital presence as a means of enhancing consumer value in fast-food outlets. "CafeEase" is positioned to improve coffee businesses' online visibility by providing a cutting-edge system that streamlines order tracking and creates a direct line of communication between customers and staff.

In conclusion, "CafeEase" fills a distinct niche at the intersection of technical innovation and the changing needs of the food and beverage business. The system intends to be a pioneering solution that enhances the customer experience, streamlines operational processes, and contributes significantly to the ongoing digital transformation in the field of coffee shop management by synthesizing insights from current research.

Objectives

The study aims to design, develop, and implement a Comprehensive Coffee Shop Management. The system aims to enhance the experience for customers and streamline the

workflow for staff, ultimately contributing to the overall success and efficiency of the coffee shop.

Specifically, this study aims to:

- Develop a user-friendly and intuitive interface for the Coffee Shop Web Kiosk, ensuring accessibility for customers of varying technological proficiency.

- Implement a QR code-based ordering system that enables customers to conveniently place orders from their tables, differentiating between take-out and dine-in orders.

- Integrate secure and efficient payment options, supporting both online payment transactions and over-the-counter payments, ensuring a seamless and flexible payment process.

- Establish a real-time notification system that alerts the cashier or on-duty personnel when a customer places an order, along with details such as table number and order items.

- Develop a customer-centric order tracking feature, allowing customers to monitor the status of their

orders in real-time, from preparation to pick up or service.

- Streamline order fulfillment processes, differentiating between over-the-counter payments where customers pick up orders at the counter, and online payments where the staff serves the prepared order directly to the table.

- Implement a dual QR code system, generating unique QR codes for take-out and dine-in orders to ensure a smooth and accurate ordering process.

- Integrate an inventory management system to track and manage the availability of menu items, ensuring efficient stock management and preventing discrepancies between online orders and actual inventory.

- Develop comprehensive user training materials and documentation to facilitate the smooth adoption of the Coffee Shop Web Kiosk system by both customers and staff.

- Implement robust security measures to protect customer data and financial information, ensuring compliance with data protection regulations and standards.

Scope and Limitations

The Coffee Shop Web Kiosk project seeks to improve the customer experience by offering a simple platform for QR-based ordering and real-time order tracking. Customers can scan unique QR codes at each table for take-out or dine-in services, while the cashier receives immediate notifications for fast order processing. The system accepts both online and in-person payments, and it has integrated inventory management to keep track of stock levels. However, constraints include the need for a stable internet connection, hardware compatibility with QR-capable devices, and the possible difficulties of total security. Customization beyond established features may necessitate extra effort, and third-party integration is contingent on API compatibility. Legal and regulatory compliance are critical, with regional variations necessitating care during deployment.

Definition of terms

To enhance clarity and facilitate comprehension, the following terminology is conceptually and operationally elucidated:

CafeEase - The overarching initiative seeks to revolutionize traditional coffee shop models through digital solutions and smart technologies, primarily focusing on a comprehensive Coffee Shop Web Kiosk system.

Smart QR-based Restaurant Dine-in System - The system inspired by Singh et al. (2022) utilizes QR code technology for efficient order placement, forming the foundation for CafeEase's QR-based ordering feature.

Web-based Information System for Coffeeshop Management - The conceptual framework proposed by Sidabutar et al. (2023), influencing the design and user-centric interface of CafeEase through the incorporation of design thinking principles.

Innovation Marketing - The promotional strategy informed by Ahamat et al. (2022), guides the strategic deployment of CafeEase by leveraging new technology to meet market

demands and contribute to ongoing innovation in the food and beverage industry.

Enhancement of Digital Presence - Informed by Daradkeh and colleagues (2023), the focus is on improving online visibility for coffee businesses through CafeEase, facilitating order tracking and direct communication between customers and staff.

Comprehensive Coffee Shop Management System - The overarching goal of the study, encompassing the development and implementation of CafeEase, is to enhance customer experience, streamline operational processes, and contribute to the digital transformation of coffee shop management.

User-Friendly Interface - A key objective focusing on the development of an intuitive and accessible interface for the Coffee Shop Web Kiosk to accommodate customers with varying technological proficiency.

QR Code-Based Ordering System - A core feature enabling customers to conveniently place orders from their tables, with a differentiation between take-out and dine-in orders.

Real-Time Notification System - A system component that alerts the cashier or on-duty personnel instantly when a customer places an order, providing essential details such as table number and order items.

Order Tracking Feature - A customer-centric component allowing real-time monitoring of order status, from preparation to pick up or service.

Dual QR Code System - The implementation of unique QR codes for take-out and dine-in orders, ensuring accuracy and efficiency in the ordering process.

Inventory Management System - A crucial element integrated into CafeEase to track and manage the availability of menu items, preventing discrepancies between online orders and actual inventory.

User Training Materials and Documentation - Comprehensive resources developed to facilitate the smooth adoption of the Coffee Shop Web Kiosk system by both customers and staff.

Security Measures - Robust features implemented to protect customer data and financial information, ensuring

compliance with data protection regulations and industry standards.

CHAPTER II
REQUIREMENTS SPECIFICATION

This chapter outlines the specific requirements, features, and functionalities significant to the researchers' system development.

Hardware Requirements

Hardware Requirements refer to the representation of the hardware used by the system. Table 1 below presents the hardware requirements to be used by the project.

Table 1. Hardware Requirements

Hardware	Functions	Specifications		Unit
		Minimum	Recommended	
Processor (CPU)	It is responsible for executing instructions, performing calculations, managing data, and coordinating the activities of other hardware components to ensure the proper functioning of our computer system.	Intel Core i3	Intel Core i5 or higher version	1
Storage (SSD)	Store and manage system data. Improve data access efficiency.	256GB	512GB to 1TB or larger	GB
Memory (RAM)	Provides smoother	8GB	16GB to 32GB	GB

	performance in handling system data.		
Network Connectivity	Modem for internet access. Stable and low-latency connection for a fast and reliable network.	35Mbps	25Mbps to 50Mbps	Mbps
POS Terminal	Wired connection to the local network. Reliable and low-latency communication with the server.	Wired connection to local network	Wired connection to local network	1
Mobile Devices	Test and ensure system responsiveness and compatibility on iOS and Android devices.	OS version: iOS and Android 4.0 (API 14) and above	OS version: iOS and Android 4.0 (API 14) and above	No particular number

Software Requirements

Software Specifications refer to the representation of the software used by the system. Table 2 below presents the software specifications to be used by the project.

Table 2. Software Requirements

Software	Description
Visual Studio Code	A free and powerful lightweight code editor providing support to the researchers' system by conducting development

	operations like debugging, task running, and version control. The required version is 1.83.1 or higher.
ReactJS	A popular JavaScript library for building user interfaces. React is a crucial tool for developing the user interface of the researchers' system. Version 17 is the required version to develop the researchers' system.
Node.js	Version 18 or higher is required for creating a high-performance and real-time application.
MongoDB	Version 6.0 or higher is required for storing and managing the system's structured data.
Web browsers	Samples are Google Chrome, Firefox, Microsoft Edge, etc. These are used to enable the system's access to necessary web-based content and manage interactions with online services and resources.
Windows 10 and Windows 11	These are versions of the Windows operating system that will be used for the researchers' system.
Laragon	A powerful development environment and server stack for building and managing the researchers' system. Version 7 to 8.2 is required.

Functional Requirements

This part enumerates the operations and features that the system must perform. Table 3 represents the descriptions of data requirements, process requirements, and output requirements.

Table 3. Functional Requirements

ID No.	Requirement Description
User Authentication and Authorization	
1.1	Customer Login: Users should be able to create accounts and log in securely. Differentiate between customer and staff logins.
1.2	Staff Authentication: Staff roles include Teller/Cashier and Crew.
1.3	Authorization Levels: Define access levels for staff (e.g., cashier, manager, inventory manager).
QR Code-Based Ordering System	
1.4	Dynamic QR Code Generation: Each table, both for take-out and dine-in, should have a unique QR code leading to the order page.
1.5	Order Placement: Customers can browse the menu, add items to the cart, and place orders via the QR code.
1.6	Payment Options: Support both online payment and over-the-counter payment.
Order Processing and Notification	
1.7	Real-time Notification: Cashiers receive instant notifications for new orders with details, including table numbers and items ordered.
1.8	Order Tracking: Customers should be able to track the status of their order (e.g., ordered, in preparation, ready for pickup).
Order Pickup and Serving	
1.9	Over-the-Counter Pickup: For over-the-counter payments, customers receive a notification when the order is ready, and they pick it up at the designated counter.
2.0	Table Service for Online Payments: For online payments, staff (crew) receive a notification, and they serve the order to the customer at the specified table.
User Interface and Experience (UI/UX)	
2.1	Intuitive Design: Ensure a user-friendly interface for

	both customers and staff.
2.2	Responsive Design: Support a responsive design for various devices (desktop, tablet, mobile).
Security Measures	
2.3	Data Encryption: Encrypt sensitive data such as user credentials and payment information.
2.4	Access Control: Implement access controls to restrict unauthorized access.
System Configuration and Deployment	
2.5	Compatibility: Ensure compatibility with popular web browsers.
2.6	Scalability: Design the system to be scalable for potential future expansions.

Non-Functional Requirements

These are requirements that pertain to behavior properties that a system must have. It defines how a system is supposed to be or its system properties. It contains the following:

Operational Requirement

Table 4 represents the requirement description that will specify the operating environment(s) in which the system must perform and how these might change over time.

Table 4. Operational Requirement

ID No.	Requirement Description
1.1	The system should be able to operate seamlessly on various web browsers and devices, including desktops, tablets, and smartphones.
1.2	The system should be able to provide clear and user-friendly documentation or guidance to help users navigate and utilize its features effectively.
1.3	The system must log user activities, including document uploads and checks for auditing and monitoring purposes.
1.4	The system must adhere to data privacy regulations and maintain user data confidentiality.

1.5	The system must be able to enhance the user-friendliness of the system's interface, making it more accessible and appealing to a broad range of users.
1.6	The system's code and interface should comply with the web development standards and best practices to ensure consistent operation across various platforms.

Performance Requirement

Table 5 represents the requirement description that will emphasize the response time, capacity, and reliability of the system.

Table 5. Performance Requirement

ID No.	Requirement Description
1.7	The system should support a minimum of 100 simultaneous users during peak hours.
1.8	The web application must load within 3 seconds on standard internet connections.
1.9	Response time for order processing and status updates should be under 2 seconds.

Security Requirement

Table 6 represents the requirement description that will address issues with security, such as who has access to the system's data and must have the ability to protect data from disruption or data loss.

Table 6. Security Requirement

ID No.	Requirement Description
2.0	All communication between the web kiosk and the server must be encrypted using HTTPS.
2.1	User authentication and authorization must be implemented securely, following industry best practices.
2.2	Personal and payment information must be stored securely, adhering to relevant data protection regulations.

Chapter III
Design and Development Methodologies

This chapter outlines the specific requirements, features, and functionalities significant to the researchers' system development.

System Design

Database Design

The essential components of Café De Vito's advanced coffee shop management system, CaféEase, are simplified and organized by using MongoDB's capabilities. Although MongoDB is a NoSQL database system, the paper's relational database design provides an organized method of organizing crucial data regarding Café De Vito's products and customer orders. MongoDB collections act as tables in this design, providing a logical and orderly schema that supports effective data recording, retrieval, and relationships between orders and products. Through making use of MongoDB's scalability and flexibility, CaféEase establishes a foundation that fits in with the continually evolving operations of Café De Vito, offering a resilient and adaptable kiosk web application solution. The MongoDB environment's focus on meaningful associations and data integrity has been shown by the usage

of foreign keys, normalization guidelines, and careful data representation.

An important component of the CaféEase database structure, the "Products" table, as shown in Table 00, provides the foundation for the comprehensive operation of Café De Vito's choice of coffee products. With an auto-incremented "ProductID," every product is individually identifiable and provides a unique reference to simplify handling of data. The "Name" and "Category" components add important details about each product, which further improves the table's information. This design helps the effective structure and categorization of Café De Vito's wide range of coffee options by facilitating the organized recording of information and enabling fast and effective data retrieval. With this setup, CaféEase ensures that the "Products" table turns into an important instrument for maintaining the database accessible and well-organized, which is essential to the kiosk web application's success.

Table 7. Fields for Products Table

Field Name	Data Type	Size	Default	Description
ProductID	INTEGER	N/A	N/A	Unique identifier for each product.
Category	VARCHAR	Variable	N/A	Category to which the

				product belongs.
Name	VARCHAR	Variable	N/A	Name of the product.

As the main focus for keeping track of and organizing customer orders, the "Orders" table, as shown in Table 00, plays a vital role in CaféEase. As the primary key, "OrderID" guarantees that every order entry in the database has a distinct identity. Precise order retrieval and management are made possible by this essential component. Boolean values like "IsPaid," "IsReady," "InProgress," "IsCanceled," and "IsDelivered" offer easy-to-see indicators of different stages in the order lifecycle, allowing you to efficiently keep an eye on every order. These fields help with decision-making in the day-to-day operations of the coffee shop and provide real-time order tracking. Furthermore, the addition of timestamps, namely "CreatedAt" and "UpdatedAt," stores important historical data and provides a historical view of the creation and most recent updates of each order. To record order histories, this chronological data is quite helpful, enabling Café De Vito to keep an extensive and well-organized record of all of its interactions with customers.

Table 8. Fields for Orders Table

Field Name	Data Type	Size	Default	Description
OrderID	INTEGER	N/A	N/A	Unique identifier for each order.
Number	INTEGER	N/A	N/A	Order number.
IsPaid	BOOLEAN	N/A	false	Indicates whether the order is paid.
IsReady	BOOLEAN	N/A	true	Indicates whether the order is ready.
InProgress	BOOLEAN	N/A	false	Indicates whether the order is in progress.
IsCanceled	BOOLEAN	N/A	true	Indicates whether the order is canceled.
delivered	BOOLEAN	N/A	false	Indicates whether the order is delivered.
OrderType	VARCHAR	Variable	N/A	Type of the order (e.g., "Eat in").
PaymentType	VARCHAR	Variable	N/A	Type of payment for the order (e.g., "Pay here").
TaxPrice	DECIMAL	Variable	N/A	Price of tax for the order.
TotalPrice	DECIMAL	Variable	N/A	The total price of the order.
CreatedAt	TIMESTAMP	N/A	Current timestamp	Timestamp when the order was created.
UpdatedAt	TIMESTAMP	N/A	Current timestamp	Timestamp when the order was last updated.

The "OrderItems" table in the CaféEase database plays an important role in providing relationships between items and customer orders, establishing a many-to-many relationship that accurately reflects the complexity of everyday scenarios. For each order item, the primary key "OrderItemID" serves as a unique identifier, maintaining data integrity and distinct identity. Foreign keys "OrderID" and "ProductID," which maintain relationships with the corresponding "Orders" and "Products" tables, are necessary for maintaining meaningful relationships. In addition to keeping referential integrity, these foreign key relationships facilitate fast and simple traversal throughout linked data entities. Moreover, the "OrderItems" table's "Name," "Price," and "Quantity" fields provide full details about every product that can be found in an order. This data provides a well-organized and unified foundation that improves the overall operation of CaféEase and is needed for accurate order computations.

Table 9. Fields for OrderItems Table

Field Name	Data Type	Size	Default	Description
OrderItemID	INTEGER	N/A	N/A	Unique identifier for each order item.
OrderID	INTEGER	N/A	N/A	Foreign key referencing the

				Order table.
ProductID	INTEGER	N/A	N/A	Foreign key referencing the Product table.
Name	VARCHAR	Variable	N/A	Name of the product in the order item.
Price	DECIMAL	Variable	N/A	Price of the product in the order item.
Quantity	INTEGER	N/A	N/A	Quantity of the product in the order item.

Architectural Diagram/Block Diagram

An architectural model showing how important parts work together to ensure CafeEase operates efficiently is shown in Figure 1. The Customer Mobile App, which lets users browse the menu, place orders, and securely pay, is where the system starts. Financial transactions run smoothly thanks to the Payment Gateway. Customers and cafe employees use mobile devices (hardware) and point-of-sale (POS) devices, respectively, to process orders. The Employee Dashboard gives employees a consolidated interface with order management and real-time analytics capabilities. APIs make it possible for the Employee Dashboard, Customer Mobile App, and other system elements to communicate with one another seamlessly. using the Gateway for Payments.

Real-time updates are sent to staff and customers by a notification service. Data is processed by an analytics engine to provide intelligent analytics. Important data is stored on the Database Server, and External Integrations connect CafeEase to the cafe's inventory management system. The Reporting Module gathers information to create in-depth reports. From order placement to inventory control and analytics, this integrated design improves the cafe's operations and guarantees a smooth and effective coffee shop management experience.
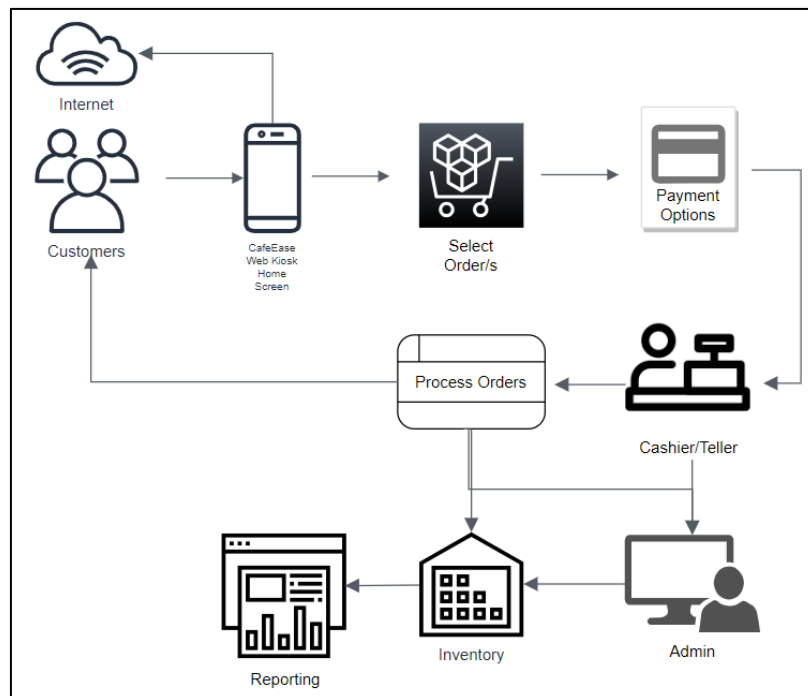

Figure 1. Architectural Diagram/Block Diagram

Data Flow Diagram Level 0

External entities like "Customer Mobile" and "Admin" in this Level 0 Data Flow Diagram, as shown in Figure 2, send requests to the central process "CafeEase." The "Customer Mobile" entity might ask for order placement, menu viewing, or payment processing services, while the "Admin" entity might ask for inventory management, report access, or notification handling.

The "CafeEase" process receives these requests and processes the data, doing tasks including order processing, analytics, and inventory management. After processing, answers are returned to the two external parties, giving the "Admin" reports, inventory updates, and other desired data, and the "Customer Mobile" order confirmations, notifications, and other pertinent information.

This bidirectional flow highlights how the CafeEase system is responsive and engaging, meeting the demands of both administrators and customers.
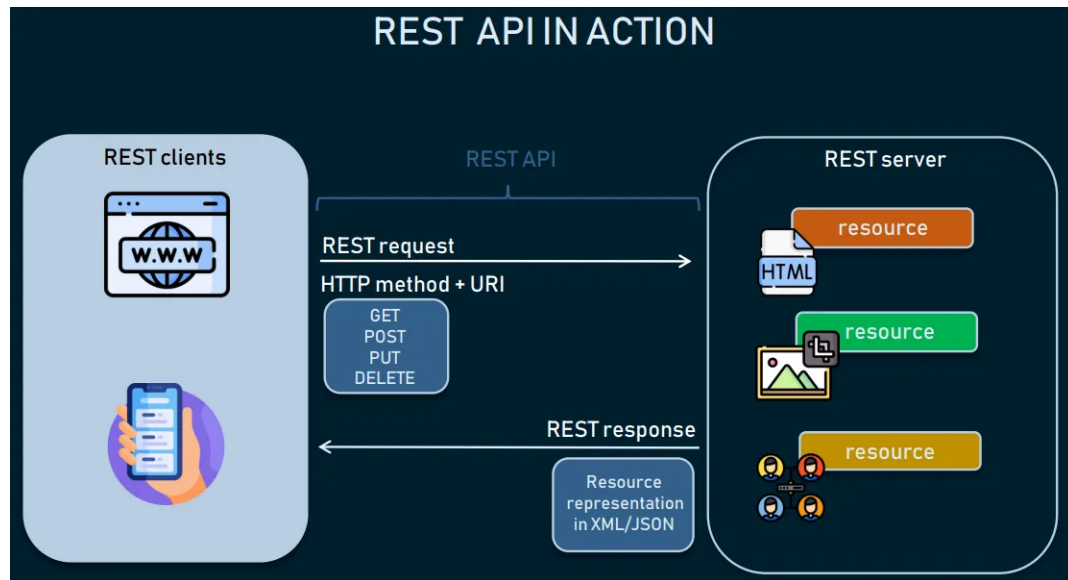
Figure 3. Data Flow Diagram Level 0

UML Use-Case Diagram

The main exchanges between the system and its users can be seen in the UML Use-Case Diagram for CafeEase. The three main players that have been identified are "Customer," "Employee," and "System Admin." The main use case, "Order Processing System," encompasses the essential features of handling orders from customers. Employees may handle orders and inventory control, while customers can browse the menu, place orders using QR codes, and make payments. The system's configuration is the responsibility of the "System Admin". The interaction between these actors

and use cases is depicted in the diagram, highlighting the systemic approach taken by CafeEase. The relationship "<>" suggests that the "Order Processing System" use case is an essential element that includes a range of interactions with customers and employees. An overview of the system's features and user interactions in a coffee shop scenario is shown in this Use-Case Diagram.
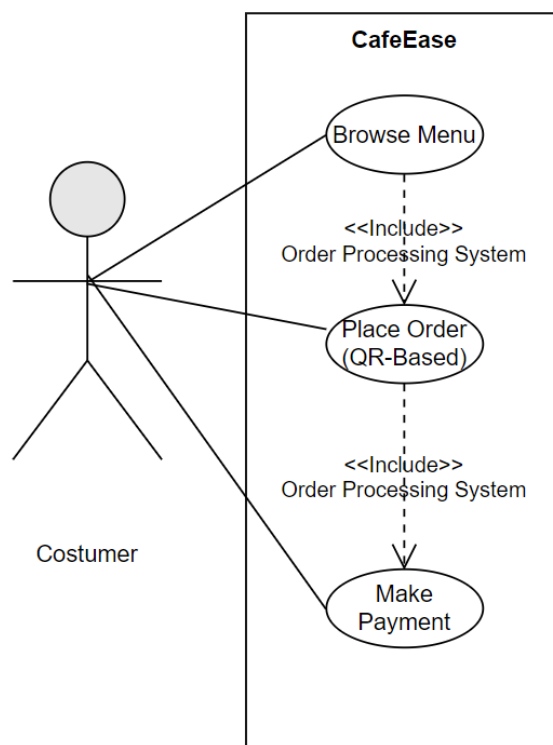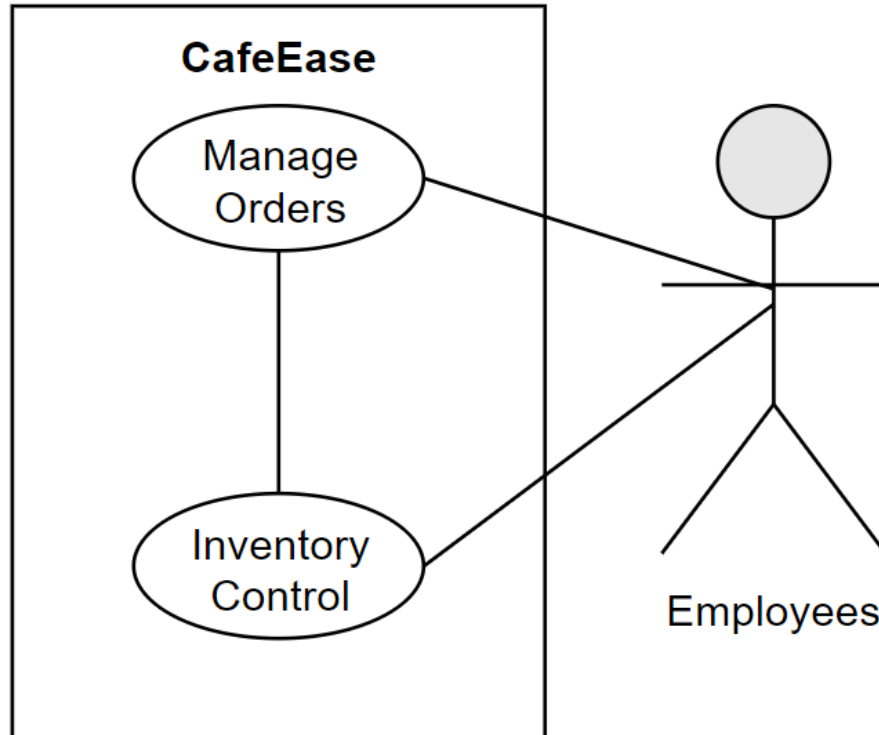

Figure 4. UML Use-Case Diagram of Users

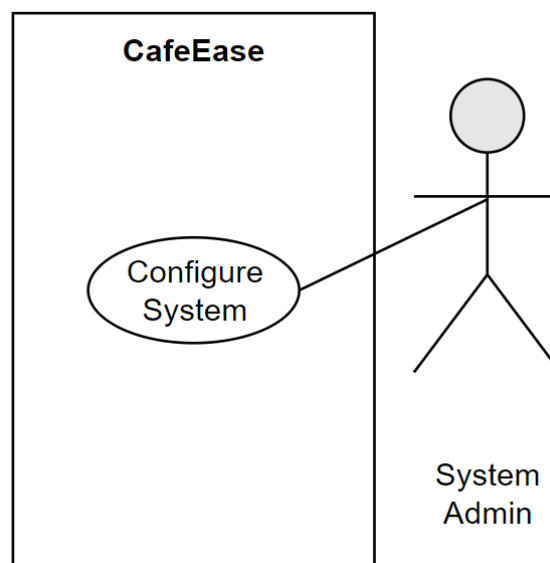**Figure 5. UML Use-Case Diagram of Employees**

Figure 6. UML Use-Case Diagram for System Admin

Figure 7. General UML Use-Case Diagram
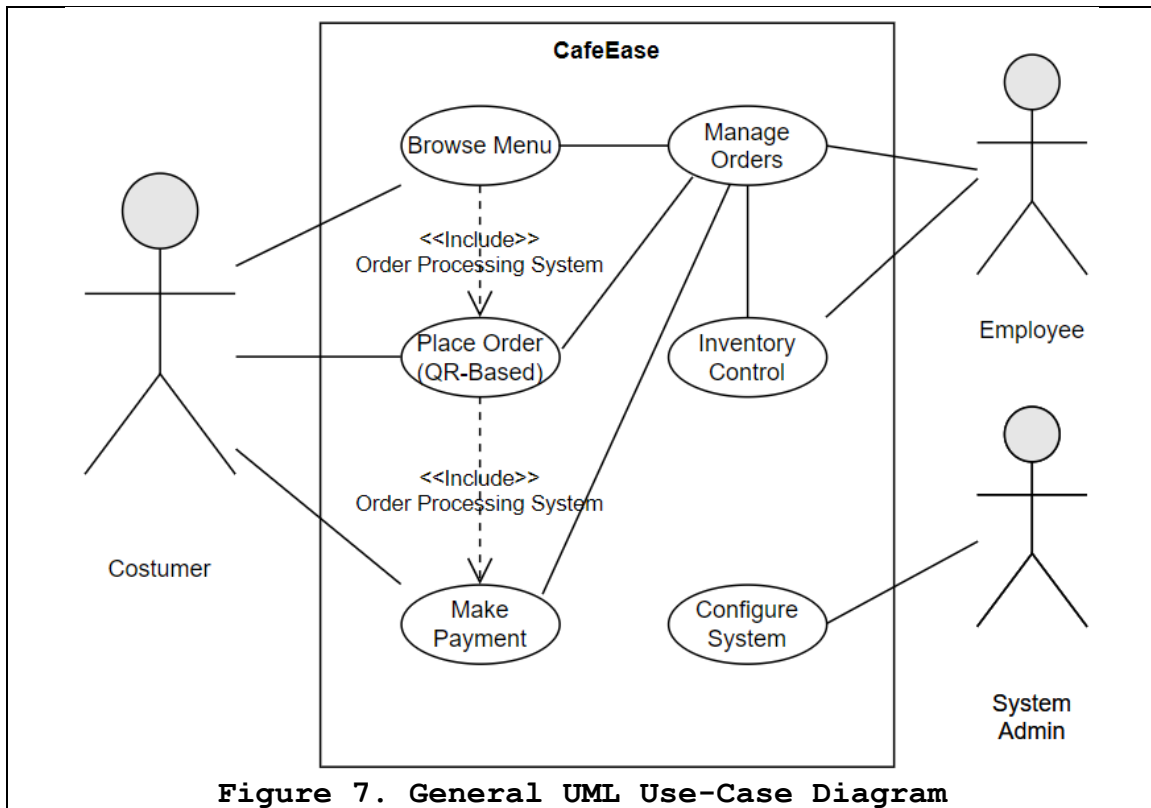
Sample Mock-up

A sample mock-up is a visual representation of a website after it is built. It consists of visuals that show how the website should look and its function. It is used to refine the design, identify potential problems, and ensure that the system meets the user's needs and expectations. Below is the system user's interface:

Figure 8. Home Page

Figure 9. Order Type Page

Figure 10. Menu Page
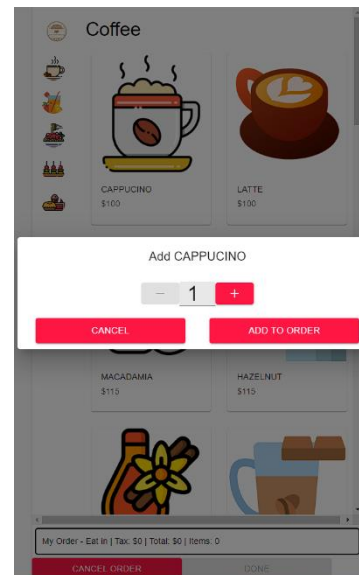
Figure 11. Adding Order/s Modal
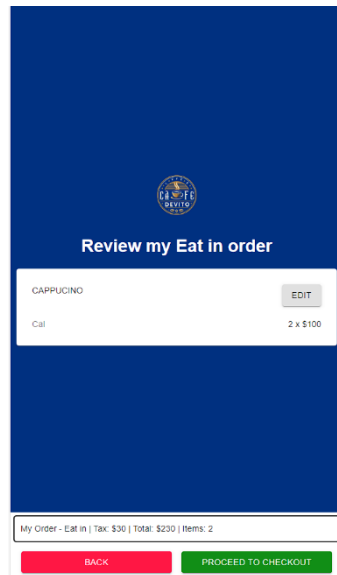
Figure 12. Review Order/s Screen

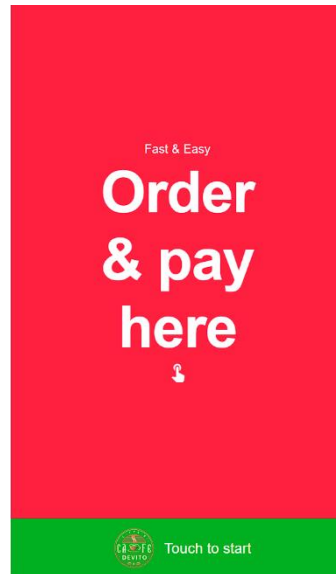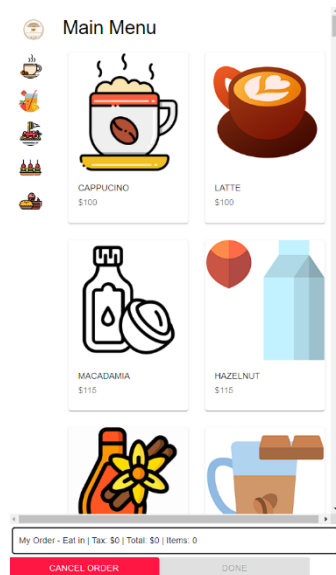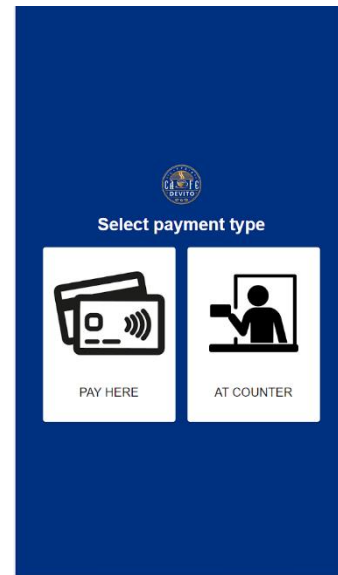**Figure 13. Select Payment Screen**

Figure 14. End Page

Development Method

The development of "CafeEase: A Comprehensive Coffee Shop Management System for Café De Vito with QR-Based Ordering, Real-time Notifications, and Integrated Inventory Control" will adhere to the System Development Life Cycle (SDLC) to ensure the development of an effective and functioning system. The development method for this study will be the Agile software development methodology because it works very well with small groups, timeframes, and budgets. It is a methodology that anticipates the need for flexibility and applies a level of pragmatism to the delivery of the finished product. Another good reason is the great deal of flexibility that it gives, especially with constantly changing requirements.
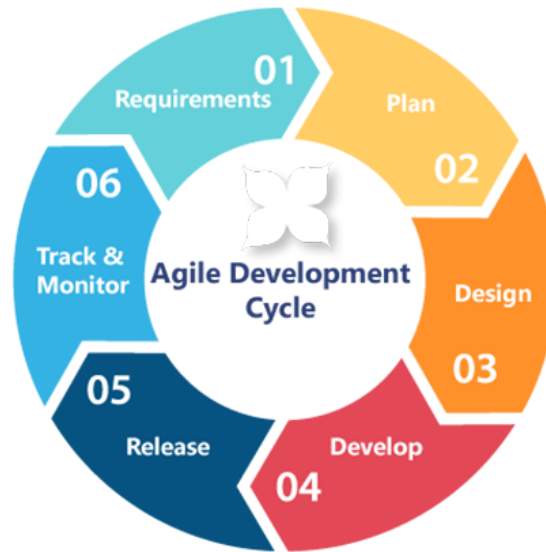
Figure 15. Agile Model

The researchers followed the following phases:

1. **Requirements Gathering.** This phase consists of brainstorming and conceptualizing the project as well as communicating with the chosen client and identifying the specific features that are needed for the software to be developed. This also includes analyzing and further refining the gathered requirements needed for developing the proposed project. It also consists of conducting a project assessment to determine the time and resources required for the given project.

2. **Planning.** During this phase, researchers conduct interviews and observations to address the identified problem. They establish the project's objectives and plans, forming the

groundwork for generating the desired system output by the end of the study.

3. **Designing.** The design phase consists of understanding the requirements gathered in the first phase. Then introduce and discuss the essential tools for the given project – the programming languages, libraries, and frameworks. This phase also includes constructing and developing a prototype to show the expected user interface and to demonstrate the main functions.

4. **Developing.** This phase consists of developing the project itself. The given project is delivered in different stages with each stage designed to improve the current version of the project. Each stage or cycle includes testing and includes many changes designed to provide improved functionality and new features compared to the last stage.

5. **Release/Testing.** At this phase, the project becomes available to the client and the developers must conduct tests to ensure that the software is functioning as expected. In this test, the researchers used the method of self-testing.

6. **Track and Monitor.** During this phase, the software is fully deployed and available to the client. The developers must now maintain the software and provide ongoing support to keep the system functioning smoothly. This stage may also

include delivering updates to the software to fix issues and add new functionality.

Gantt Chart

In this section, a Gantt Chart is presented to show the plans and schedules of the project timeline. All the development stages up to the completion of the project were documented in this chart. This helps the researchers to know the deadlines needed to accomplish and show breakthroughs in various tasks.

Table 10 shows the whole process of developing CafeEase. It displayed the various tasks and marks as completed oats certain date. Researchers will be kept informed of the progress of the development which will help them not to miss out on steps and differentiate tasks from the amount of time took to complete them.

Table 10. Gantt Chart

Task Name	Task Date																			
	Oct				Nov				Dec				Jan				Feb			
	Week 1	Week 2	Week 3	Week 4	Week 1	Week 2	Week 3	Week 4	Week 1	Week 2	Week 3	Week 4	Week 1	Week 2	Week 3	Week 4	Week 1	Week 2	Week 3	Week 4
1. Planning	■	■	■	■	■															
1.1 Conduct an interview	■	■																		
1.2 Define project objectives		■	■	■																
1.3 Define project plan			■	■																
1.4 Approval of project plan					■															
2. Requirements Gathering						■	■	■												

Mindoro State University
College of Computer Studies

Task																		
2.1 Data Collection						🟥	🟥	🟥										
2.2 Functional						🟥	🟥	🟥										
2.3 Non-Functional						🟥	🟥	🟥										
3. Design								🟥	🟥	🟥	🟥	🟥	🟥				
3.1 Frontend Software Design								🟥	🟥	🟥	🟥	🟥	🟥				
4. Development								🟥	🟥	🟥	🟥	🟥	🟥				
4.1 Back-end coding								🟥	🟥	🟥	🟥	🟥	🟥				
5. Testing													🟥	🟥	🟥			
5.1 Functionality testing													🟥	🟥	🟥			
5.2 User interface testing													🟥	🟥				
6. Implementation														🟥	🟥			
7. Maintenance															🟥	🟥	🟥	🟥
7.1 Project monitoring															🟥	🟥	🟥	🟥
7.2 Resolve system errors															🟥	🟥	🟥	🟥

Legend: 🟥 - Completed/ Done

Testing and Evaluation

This section refers to the testing and evaluation phase of the software development process. During this phase, developers are required to conduct tests on their system to determine its capabilities and limitations. This will allow them to identify any issues or potential problems before the production and deployment stages. The tests should include all the requirements outlined in the Requirements Phase, such as design, performance, supportability, etc. The results of these tests will be evaluated to assess the progress of the system and ensure

it meets the requirements of the project. The developers considered the following:

1. **Unit Testing**- a type of software testing where individual units or components of a software system are tested. This type of testing is usually done by the developers, as it requires detailed knowledge of the internal structure of the system. Unit Testing is designed to test individual functions, modules, and features of the system, to ensure that they all behave as expected.

2. **Component Testing**- involves testing the individual components of the system (modules, classes, objects, and programs) in isolation, without integrating them with other components. This helps to identify any defects or bugs in the individual components before they are integrated into the system. It also helps to identify any unexpected interactions or dependencies between components that could cause problems or errors in the system.

3. **System Testing**- a type of software testing that evaluates the entire system or application and its components to verify that all individual modules are working properly and that data is transferred accurately between modules and the entire system. System testing is meant to ensure that the system meets its requirements, performs as expected, and functions correctly in its intended environment. It is an

overall test of the system and its components, and it is typically done after unit and integration testing.

4. **Unit Acceptance Testing**- a process to validate if the unit (which could be a software, product, or service) meets the requirements of the end-users and clients. During unit acceptance testing, end-users and clients interact with the unit and provide feedback on its features, usability, and performance. Based on their feedback, they either accept or reject the unit. It is an important step in the development process as it ensures that the unit meets the user's expectations.

CHAPTER IV
DEVELOPMENT, TESTING AND EVALUATION RESULT

This chapter deals with the presentation of the system output and testing results of the system.

Presentation of the System Output

A. **Home Page/Screen**

Figure 15 shows the Home Page/Screen of the System. This is the starting point for the users on how to order in the shop.
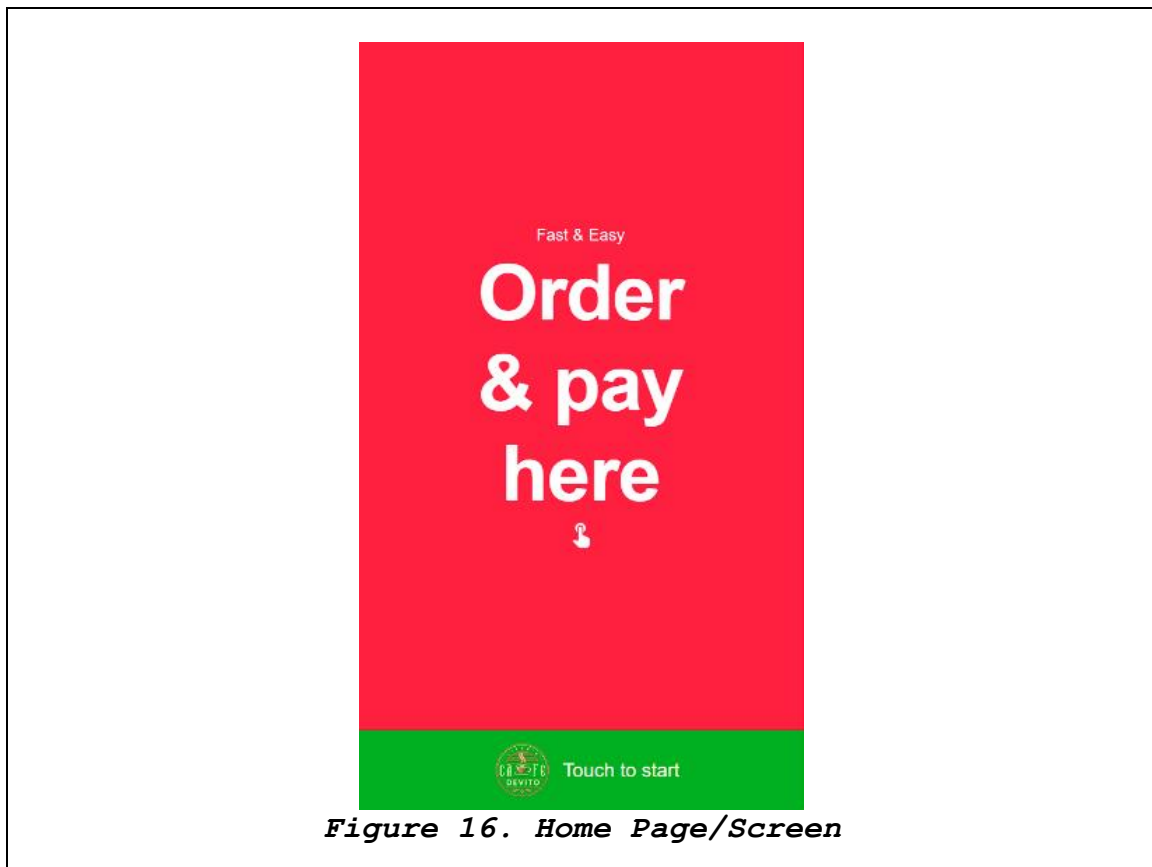

Figure 16. Home Page/Screen

B. **Order Type Screen**

Figure 16 shows the Order Type Screen of the System. In this screen, users have two options whether to "Eat In" or "Take Out" his/her order/s.

Figure 17. Order Type Screen

C. **Menu Page/Screen**

Figure 17 shows the Menu Page/Screen of the System. This screen shows all the products of the Café with their corresponding category.
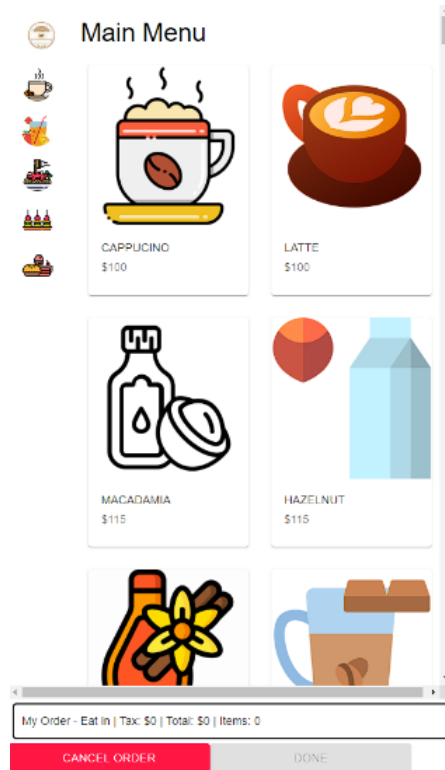
Figure 18. Menu Page/Screen

D. **Adding Order/s Modal**

Figure 18 shows the addition of order/s Modal. In this screen, after selecting the product, users can manage or decide to add more of that selected product and the system will count it.
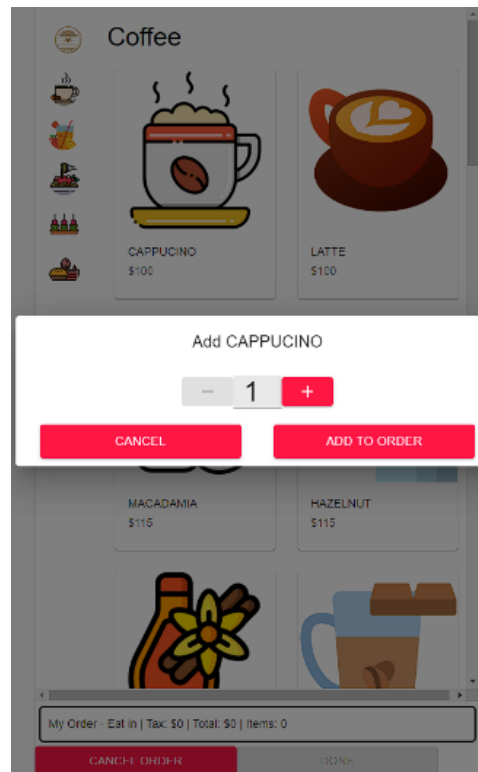
Figure 19. Adding Order/s Modal

E. **Review Order/s Page/Screen**

Figure 19 shows the Reviewing of Order/s Page. On this page, the system will show all the orders of the customer. The user can also edit his/her order. The user also can view the total number of orders and the total amount to be paid.
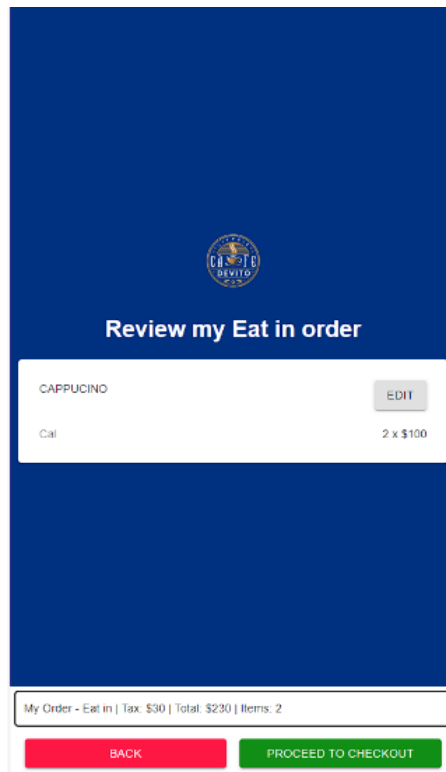
Figure 20. Review Order/s Screen/Page

F. **Payment Options Screen**

Figure 20 shows the Payment Options Screen. The users can decide whether payment will be made in a kiosk or at the counter.
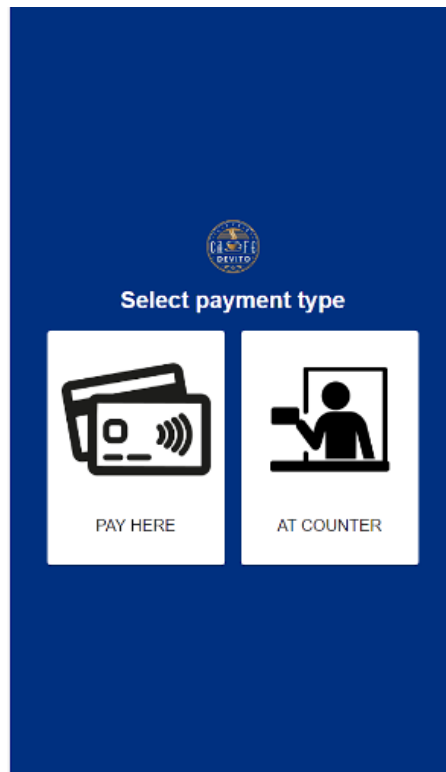
Figure 21. Payment Options Screen

G. **End Page**

Figure 21 shows the last page or interface of the system. Users can view his/her order number.

Figure 22. End Page

CHAPTER V
CONCLUSION AND RECOMMENDATION

CONCLUSION

The development of the comprehensive Coffee Shop Management System, CafeEase, specifically designed for Café De Vito, is a big step in the direction of modernizing and optimizing coffee ordering and administration procedures. The comprehensive Software Requirements Specification (SRS), which incorporates real-time notifications, QR-based ordering, and a strong inventory control system, captures the idea for an effective and customer-focused platform. This effort meets the unique demands of Café De Vito in improving customer experience and operational efficiency, in addition to being in line with current technological developments in the hospitality sector.

After a careful examination of both functional and non-functional requirements, the development team may move on with the implementation phase with a strong base. CafeEase is ready to transform the cafe's operations by following these guidelines, guaranteeing a smooth and enjoyable experience for both patrons and employees.

RECOMMENDATION

The effective implementation of essential recommendations is vital to the successful implementation of CafeEase, the comprehensive Coffee Shop Management System created specifically for Café De Vito. Encouraging open communication between developers and cafe stakeholders through a collaborative development strategy will guarantee that CafeEase easily integrates with the unique operational intricacies of Café De Vito. To ensure a seamless transition for cafe employees, reduce interruptions, and maximize the system's integration into everyday operations, it is imperative to consider a comprehensive user training and onboarding program. Comprehensive testing, including functional, security, and stress testing, will ensure that CafeEase is dependable and resilient in a variety of scenarios at every level of development. Protecting client data and financial activities requires a constant commitment to data security procedures, including frequent audits and compliance checks. Anticipating future expansion through proactive scalability planning will future-proof CafeEase, allowing it to handle rising numbers without sacrificing functionality. Creating a way for users to

submit feedback—both from patrons and cafe employees—will yield insightful information for continuous enhancements, guaranteeing that CafeEase develops in step with Café De Vito's changing requirements and continues to be a pillar of both customer pleasure and operational effectiveness.

REFERENCES

1. Singh, R., Sonje, R., Salkar, S., & Jadhav, A. (2022). Smart QR-based Restaurant Dine-in System with Sales Analysis. In ITM Web of Conferences (Vol. 44, p. 03014). EDP Sciences.

2. Sidabutar, G. D. G., Seah, J. A., & Singgalen, Y. A. (2023). Analysis and Design of Web-based Information System for Coffeeshop Management using Design Thinking Methodology: Case of Kopi KurangLebih. Journal of Information Systems and Informatics, 5(1), 217-231.

3. Demartini, M., Pinna, C., Tonelli, F., Terzi, S., Sansone, C., & Testa, C. (2018). Food industry digitalization: from challenges and trends to opportunities and solutions. IFAC-PapersOnLine, 51(11), 1371-1378.

4. Manjula, S., Balaji, P., Deepa, N., & Vidhyavathi, A. (2021). Digital Transformation on Food Retail Industries-A Review. Asian Journal of Agricultural Extension, Economics & Sociology, 39(11), 88-101.

5. AHAMAT, A., ALI, M. S. S., AZAMI, M. A., Prasad, N. V., dan Teknousahawanan, F. P. T., & Dhabi, A. (2022).

Innovation Marketing from the Perspective of New Technologies in the Food and Beverage Industry.

6. Daradkeh, F. M., Hassan, T. H., Palei, T., Helal, M. Y., Mabrouk, S., Saleh, M. I., ... & Elshawarbi, N. N. (2023). Enhancing Digital Presence for Maximizing Customer Value in Fast-Food Restaurants. Sustainability, 15(7), 5690.