

**ShopEase: An Indoor Navigation and Product Locator System with Real-Time
Pricing and Dynamic Aisle Mapping for Supermarkets**

**A Capstone Project
Presented to the Faculty of
Information and Communications Technology Program
STI College Novaliches**

**In Partial Fulfilment
of the Requirements for the Degree
Bachelor of Science in Computer Engineering**

Shanna May Alasaas

Jasper M. Bacarisa

Angelo Miguel R. Cua

Shaine Faye G. Salting

October 26, 2025

ENDORSEMENT FORM FOR ORAL DEFENSE

TITLE OF RESEARCH: **ShopEase: An Indoor Navigation and Product Locator System with Real-Time Pricing and Dynamic Aisle Mapping for Supermarkets**

NAME OF PROPONENTS: Shanna May Alasaas
Jasper M. Bacarisa
Angelo Miguel R. Cua
Shaine Faye G. Salting

In Partial Fulfilment of the Requirements
for the degree Bachelor of Science in Computer Engineering
has been examined and is recommended for Oral Defense.

ENDORSED BY:

Karen Cristy A. Cifra
Thesis Adviser

APPROVED FOR ORAL DEFENSE:

Karen Cristy A. Cifra
Thesis Coordinator

NOTED BY:

Noriel C. Domondon
Program Head

October 26, 2025

APPROVAL SHEET

This thesis titled **ShopEase: An Indoor Navigation and Product Locator System with Real-Time Pricing and Dynamic Aisle Mapping for Supermarkets**, prepared and submitted by **Shanna May Alasaas; Jasper M. Bacarisa; Angelo Miguel R. Cua; and Shaine Faye G. Salting**, in partial fulfillment of the requirements for the degree of Bachelor of Science in Computer Engineering, has been examined and is recommended for acceptance and approval.

Karen Cristy A. Cifra
Thesis Adviser

Accepted and approved by the Thesis Review Panel
in partial fulfillment of the requirements for the degree of
Bachelor of Science in Computer Engineering

Dr. Harold Lucero
Panel Member

Earlgel Kenneth Guardian
Panel Member

John Robert Soriano
Lead Panelist

APPROVED:

Karen Cristy A. Cifra
Thesis Coordinator

Noriel C. Domondon
Program Head

October 26, 2025

ACKNOWLEDGEMENTS

The researchers would like to express their deepest gratitude to **God**, whose guidance, strength, and wisdom made it possible to accomplish this study.

To our **Thesis Coordinator, Ms. Karen Cristy A. Cifra**, for providing direction, encouragement, and constructive feedback throughout the entire thesis process, ensuring that the study met the required standards.

To our **Thesis Advisers, Mr. Brandon Lleno** during the Methods of Research, **Mr. Earlgel Kenneth Guardian** during Capstone 1, and **Ms. Karen Cristy A. Cifra** during Capstone 2, for their constant guidance, expertise, and patience which greatly strengthened the content and overall quality of this research.

To our **Program Head, Mr. Noriel Domondon**, for his continuous support, leadership, and for fostering an environment that promotes academic excellence.

To the faculty members of the **Information and Communications Technology**, for their insightful comments, recommendations, and professional input that significantly improved the study.

To our **Parents and Guardians**, for their unwavering support, love, and encouragement, and for providing the motivation and resources that made this journey possible.

To our **Friends and Inspirations**, who offered help, strength, and encouragement during challenging moments, inspiring us to persevere.

Finally, to everyone who contributed in any way, through time, technical assistance, moral support, or shared knowledge that the researchers express their sincere gratitude. This work would not have been possible without your collective support and inspiration.

ABSTRACT

Title of research: **ShopEase: An Indoor Navigation and Product Locator System with Real-Time Pricing and Dynamic Aisle Mapping for Supermarkets**

Researchers: **Shanna May Alasaas
Jasper M. Bacarisa
Angelo Miguel R. Cua
Shaine Faye G. Salting**

Degree: **Bachelor of Science in Computer Engineering**

Date of Completion: **December 2025**

Keywords: **ShopEase, Product Locator, Real-time Pricing, Dynamic Aisle Mapping, Supermarket Navigation**

Supermarkets are usually troubled with aiding customers to find their products, confirm their prices, and control their costs effectively. Conventional systems such as aisle marking, employee support, and price scanners are restricted, and thus a lot of time is lost, and satisfaction is lessened. In order to overcome these problems, this paper worked on ShopEase: An Indoor Navigation and Product Locator System with Real-Time Pricing and Dynamic Aisle Mapping for Supermarkets to solve the problem of improving the efficiency of shopping and assisting store operations.

ShopEase is a product that combines a Raspberry Pi, a touchscreen, a barcode scanner, and an uninterrupted power supply and is backed by a microcontroller with inherent Wi-Fi and Bluetooth functionality. It utilizes signal-based indoor positioning to give general directions within the store, real-time product search, price, and costs. An administrator dashboard gives the staff to update information and control operations. The project was developed in the form of design, integration, and testing phases with the help of the Agile approach.

Assessment proved that ShopEase was beneficial to direct customers and enhanced administrative processes. Minor delays in navigation were not problematic because they are the natural outcomes of data processing. All in all, the system was trustworthy and viable, as it provided reliable navigation, pricing transparency, and operational support to modernize the experience of retail shopping and improve customer experience.

TABLE OF CONTENTS

	Page
Title Page	i
Endorsement Form for Proposal Defense	ii
Approval Sheet	iii
Acknowledgments	iv
Abstract	v
Table of Contents	vi
List of Tables	viii
List of Figures	x
List of Appendices	xi
Introduction	1
Background of the problem	2
Overview of the current state of technology	3
Objectives of the study	4
Scope and limitations of the study	5
Review of Related Literature/Studies/Systems	7
Review of Related Systems	7
Review of Related Literature	10
Technical Background	13
Overview of the Proposed System	13
System Components	14
Block Diagram	18
Methodology	19
Requirements Analysis	19

Requirements Documentation	19
Design of Software, System, Product, and/or Processes	22
Development and Testing	24
Testing Procedures	26
Description of Prototype	27
Implementation Plan	34
Implementation Results	35
Results and Discussions	36
Summary and Discussion	51
Conclusion and Recommendations	53
Conclusion	53
Recommendations	54
References	56
Appendices	58
Calendar of Activities	59
Actual Thesis Expenses	62
User's Manual	65
Admin's Manual	70
Device Assembly Guide	74
Relevant Source Code	77
Curriculum Vitae of Researchers	100

LIST OF TABLES

Table No.	Table Name	Page
1.0	Software Components	14
1.1	Hardware Components	15
1.2	Hardware Requirements	20
1.3	Software Requirements	21
1.4	Age Group of Respondents	37
1.5	Gender of Respondents	37
1.6	Frequency of Supermarket Shopping	37
1.7	Usual Supermarket Visited	38
1.8	Reasons/Challenges Encountered in Large Supermarkets	38
1.9	Top Priorities When Shopping at a Supermarket	39
2.0	Asking staff for directions	39
2.1	Difficulty in Finding Items in the Store (Supermarket)	40
2.2	Keeping Track of Expenses	40
2.3	Leaving the Store Without Buying an Item Because It Could Not Be Found	40
2.4	Familiarity with Smart Carts	41
2.5	Interest in Using ShopEase Smart Cart	41
2.6	Most Useful Features of ShopEase	42
2.7	Comfort Level in Using a Touchscreen Device While Shopping	42
2.8	Belief That ShopEase Could Reduce Shopping Time	43
2.9	Additional Features Respondents Would Like ShopEase to Have	43
3.0	Concerns About Using a Digital Shopping Assistant Cart	44
3.1	Willingness to Recommend ShopEase to Others	45
3.2	Likelihood of Using ShopEase if Available in Local Supermarkets	45
3.3	Aspect of Shopping Experience That ShopEase Would Improve the Most	45
3.4	Age Group of Respondents.	46
3.5	Gender of Respondents.	47
3.6	Ease of Using ShopEase in Locating Products Inside the Supermarket.	47
3.7	Accuracy of Product Location Displayed on the Map.	47

3.8	User Satisfaction with the Real-Time Pricing Feature.	48
3.9	System Responsiveness When Scanning Barcodes.	48
4.0	Clarity and Understandability of the User Interface.	48
4.1	Perception on ShopEase in Making Shopping Faster and More Convenient.	49
4.2	Likelihood of Using ShopEase Again If Implemented in a Real Supermarket	49
4.3	Most Useful Feature of ShopEase as Identified by Respondents	49
4.4	Device Parts Encountered with Issues	50
4.5	Gantt Chart of Activities during Methods of Research	60
4.6	Gantt Chart of Activities during Capstone Design 1 and 2	61
4.7	Actual Thesis Expenses	63

LIST OF FIGURES

Figure No	Figure Name	Page
1.0	ShopEase Core Components Block Diagram	18
1.1	Agile Methodology Process	24
1.2	3D Model of ShopEase Device Back-View & Scanner Front-View	28
1.3	3D Model of ShopEase Device Front-View & Scanner Back-View	28
1.4	3D Model of ShopEase Device & Scanner Top-View	29
1.5	3D Model of the ShopEase Hardware Setup Diagram	29
1.6	3D Model of ShopEase Device Hardware Components	29
1.7	3D Model of the ShopEase Hardware Interfaces and Connectivity	30
1.8	User Search Page	30
1.9	Additional Features of User Interface	30
2.0	Admin Management Page	31
2.1	Password Reset Request	31
2.2	Admin's Log, Aisle/Product Lists and Admin Form	32
2.3	User's Interface Flowchart	32
2.4	Admin Dashboard Flowchart	33
2.5	ShopEase' Implementation Plan	34
2.6	ShopEase User Manual: Cover and Introduction Pages	66
2.7	ShopEase User Manual: Interface Overview and Starting Screen	66
2.8	ShopEase User Manual: Home Page and Categories Sidebar Interface	67
2.9	ShopEase User Manual: Product Details and My Cart Location Interface	67
3.0	ShopEase User Manual: Search Bar and Cart Icon Interface	68
3.1	ShopEase User Manual: Tutorial and Done Shopping Pages	68
3.2	ShopEase User Manual: Troubleshooting Guide and Thank You Page.	69
3.3	ShopEase Admin's Manual: Cover and Introduction Pages.	71
3.4	ShopEase Admin's Manual: Login and Home Dashboard Pages.	71
3.5	ShopEase Admin's Manual: Aisle and Product Management Pages.	72
3.6	ShopEase Admin's Manual: Product Lists and Admin Administration Pages.	72
3.7	ShopEase Admin's Manual: Password Management and Activity Logs Pages.	73

3.8	ShopEase Admin's Manual: Frequently Asked Questions (FAQs) and Thank You Pages.	73
3.9	ShopEase Device Setup Guide: Cover and Package Content Pages.	75
4.0	ShopEase Device Setup Guide: Device Parts and Overview Pages.	75
4.1	ShopEase Device Setup Guide: Sales Operation and Troubleshooting Guide Pages.	76

LIST OF APPENDICES

Appendix	Page
A. Calendar of Activities	59
B. Actual Thesis Expenses	62
C. User Manual	65
D. Admin's Manual	70
E. Device Assembly Guide	74
F. Relevant Source Code	77
G. Curriculum Vitae of Researchers	100

INTRODUCTION

As the number of shoppers increases and daily routines become more demanding, the need for a faster and more convenient shopping experience grows. Supermarkets and hypermarkets provide security, comfort, and a wide variety of products in one place. However, finding products can be time-consuming and frustrating, especially in large stores with constantly changing layouts. Shoppers often spend unnecessary time searching for products or asking staff for assistance, leading to delays and a less satisfying experience.

According to Nicasio (2020), one of the biggest frustrations for customers is struggling to find an item, which can result in dissatisfaction and even lost sales for stores. Customers want to locate products quickly, compare prices, stay within budget, and complete their shopping efficiently. While technology has transformed retail experiences globally, many supermarkets in the Philippines have yet to adopt solutions that streamline in-store navigation and product tracking.

One promising development is the smart shopping cart, which combines indoor navigation, price transparency, and interactive assistance. This technology provides customers with on-screen directions to products, up-to-date pricing, and expense tracking, making shopping smoother and more efficient (Retail TouchPoints, 2023). Studies also show that smart cart systems reduce the time spent searching for items and help shoppers make more informed decisions (Shopic, 2023).

Beyond enhancing customer convenience, smart carts benefit store management by optimizing product placement, updating pricing in real time, and improving inventory tracking (Kushwaha & Agrawal, 2020). As digital transformation continues, the adoption of smart-cart technology could set a new standard for supermarkets in the Philippines.

This study examines the impact of smart shopping carts on customer satisfaction, shopping efficiency, and retail operations. By exploring how indoor navigation, price transparency, and digital integration improve the supermarket experience, this research provides insights into how technology can reshape retail practices and meet the evolving needs of consumers.

Background of the Problem

Shoppers in large supermarkets across the Philippines often struggle to find products quickly, leading to frustration and wasted time. Store layouts frequently change due to restocking or promotions, making navigation difficult. Without a reliable guide, customers are left to search manually, slowing down their shopping and causing dissatisfaction. Many also rely on store staff for help, interrupting both their own experience and the employees' workflow.

Aside from navigation issues, comparing product prices is also challenging. The absence of real-time and transparent pricing prevents shoppers from making informed decisions. Those with limited time or mobility face greater inconvenience, making shopping more discouraging, especially since most local supermarkets still lack digital systems that assist in product location, price checking, and expense tracking.

A study in Metro Manila found that supermarket factors such as layout, accessibility, and staff communication strongly influence customer satisfaction and loyalty (Depusoy, Nartea, Del Rosario, & de Mesa, 2021). Inefficient product placement and unclear navigation were key issues, showing the need for smarter store designs and digital guidance systems that enhance convenience and overall service quality.

Globally, the market for smart shopping cart technologies is expanding. In 2022, it was valued at around ₱79.6 billion and is expected to grow further by 2030 (ASDR Reports, 2023). This shows the increasing demand for intelligent retail systems that combine navigation, real-time pricing, and customer support that technologies already used in some Asian supermarkets to deliver faster and more efficient shopping experiences.

According to Roe, Spanaki, Ioannou, Zamani, and Giannakis (2022), IoT-based retail systems help stores analyze customer behavior, improve inventory management, and optimize product placement. Similarly, Credence Research (2023) reported that smart shopping carts reduce checkout time, improve budgeting accuracy, and enhance customer satisfaction. For the Philippine retail industry, adopting such technology could make shopping more efficient, informed, and enjoyable while reducing the workload of store staff.

Overview of the Current State of the Technology

Large supermarket chains cater to a vast customer base seeking both a diverse product selection and a streamlined shopping experience. Despite the use of digital signage, basic store apps, and traditional aisle markers, the current in-store navigation and checkout systems still present challenges for customers. Shoppers often spend significant time searching for products or waiting in checkout lines, leading to inefficiencies, particularly in high-traffic locations. These issues have existed for years despite repeated attempts to simplify the shopping process, which shows that earlier solutions have not fully addressed customer needs.

Some advancements, such as barcode scanning devices and mobile store maps, have been introduced to improve the shopping experience, but they often fail to meet the specific requirements of medium-to-large stores. For instance, while clip-on smart devices for shopping carts are available in international markets, they typically offer limited navigation, minimal price updates, and lack budget management features. Similarly, AI-driven cashierless stores demonstrate new ways to streamline shopping, but these remain costly, sensor-heavy, and difficult to scale for multi-aisle supermarkets. Indoor navigation technologies have also been explored, yet they remain experimental due to the challenges of mapping dynamic retail environment.

With these gaps in mind, ShopEase aims to provide a more comprehensive and practical solution. By integrating indoor navigation, real-time pricing, and a barcode scanning feature for expense tracking, it offers an all-in-one platform tailored to the needs of large supermarkets. This approach not only addresses the long-standing inefficiencies faced by both shoppers and retailers but also sets a foundation for modernizing retail practices in the Philippine context.

Objectives of the Study

The general objective of this study is to design and implement a supermarket navigation and product locator system that enhances the shopping experience through product information, dynamic aisle mapping, and barcode-enabled expense tracking.

Specifically, the study aims to:

1. Create an indoor navigation feature that assists shoppers in locating specific products and guides them to the appropriate aisle.
2. Develop a user-friendly touchscreen interface for each pushcart to allow convenient product searches and navigation assistance;
3. Integrate a barcode scanning system that enables customers to check real-time product pricing, add desired items through the device, and edit item quantities on screen to track total expenses;
4. Develop a functionality that allows the store administrators to update the product location in real time and the information on navigation to be up-to-date.
5. Develop an online platform for store administrators to efficiently manage product information, locations, and pricing; and
6. Ensure effective system operation by utilizing Wi-Fi to synchronize data between the product database, administrative interface, and pushcart displays.

Scope and limitations of the study

Scope of the Study

This study focuses on the design and development of ShopEase, a supermarket indoor navigation and product locator system. Specifically, the scope covers the following:

- **Indoor Navigation and Aisle Mapping** – Provides turn-by-turn directions to guide shoppers to specific products within the supermarket.
- **User-Friendly Touchscreen Interface** – Allows convenient product searches, manual item addition to a virtual cart, quantity adjustments, and monitoring of total expenses for better budgeting.
- **Barcode Scanning Feature** – Enables customers to check real-time product pricing information.
- **Administrator Web Platform** – Allows store administrators to manage product data, update product placements, and configure aisle mapping.
- **Automatic Database Synchronization** – Ensures that product and pricing information remain accurate and up to date by connecting directly with the store's pricing database.

ShopEase is intended for use in medium-to-large-scale supermarkets and hypermarkets. The development and testing of this study were conducted during Academic Year 2025–2026.

Limitations of the Study

The system's functionality is subject to certain limitations beyond the control of the researchers. Specifically, the limitations include:

- **Dependence on Wi-Fi Connectivity** – ShopEase relies on stable internet access for real-time updates on product locations; performance may be affected by poor signal or network downtime.
- **Manual Input Requirement** – Expense tracking and quantity adjustments must be entered manually through the touchscreen interface.
- **Accessibility Concerns** – The touchscreen interface may present challenges for users with limited digital experience or visual impairments, possibly requiring staff assistance.

- **Device Maintenance Needs** – The hardware may require periodic maintenance to ensure durability and prevent issues caused by frequent handling or tampering.
- **Battery Dependence** – Touchscreen units on pushcarts must be regularly charged to maintain continuous operation throughout the day.
- **Navigation Accuracy** – Changes in store layout or temporary connectivity issues may affect the precision of aisle mapping and product navigation.
- **System Integration** – ShopEase is not designed as a self-checkout or inventory management system; all purchases must still be completed through the store's existing cashier counters.

REVIEW OF RELATED SYSTEMS AND LITERATURE

Review of Related Systems (Foreign)

Raspberry Pi has become a common choice as a local web server in embedded and IoT environments. These devices are widely recognized for hosting web services, processing sensor data locally, and reducing dependency on remote cloud infrastructure. Such practicality makes them ideal for constrained environments where low cost and flexibility are essential, a trend observed by Mathe et al. (2024) in their evaluation of alternatives to PC-based XAMPP setups.

Scalability also plays a major role in system deployment, as clusters of Raspberry Pis can function together like miniature data centers. By enabling load balance, redundancy, and distributed hosting, organizations are able to extend system reliability beyond a single unit. This principle has been tested in web applications and IoT services where several Raspberry Pis operated cooperatively, a practice that Farrel et al. (2023) considers a practical method for institutions needing low-cost yet expandable server infrastructures.

Industrial applications reinforce this adaptability, particularly within Industry 4.0 environments where Pis act as gateways for SCADA systems, sensors, and communication protocols. In these scenarios, they not only collect data but also deliver dashboards and localized processing to reduce cloud dependence. Ali (2025) demonstrates that this hybrid role of computation and connectivity positions Raspberry Pi as more than a supplementary device, but as a viable backbone for industrial-grade systems.

The decision between local and cloud hosting further highlights the Pi's flexibility. Systems hosted locally on Raspberry Pi reduce latency, maintain offline functionality, and lessen reliance on constant internet access, qualities that make it suitable for retail and public kiosks. Yet, these strengths must be balanced with additional precautions, such as SSL encryption and automated backups, to address vulnerabilities that may arise in small-scale hardware hosting (Developer Communities, 2023).

Raspberry Pi's value extends beyond hosting by enabling more intelligent computing tasks, such as federated learning in IoT networks. The FedIoT framework illustrates how Pis can process sensitive data locally, contributing to global training models without exposing

private information. Zhang, He, Ma, Gao, and Avestimehr (2021) argue that this distributed intelligence underscores the potential of Raspberry Pi to serve not only as a lightweight server but also as an active learning node in advanced computational systems.

Review of Related Systems (Local)

In the Philippines, efforts to enhance shopping and retail experiences through technology have steadily gained attention. One example is the mobile-based indoor navigation system for malls, which was introduced to help users locate stores more efficiently (De Guzman & Tan, 2019). Their project showed how digital maps could reduce the time spent searching for shops and improve overall customer satisfaction. This reflects the growing demand for digital solutions that simplify navigation in commercial spaces and create a more efficient retail environment.

Building on this, barcode scanning systems have been tested in local retail stores to provide real-time product pricing, showing how transparency can support informed decision-making. Santos and Dela Cruz (2020) noted that the introduction of barcode technology also reduced complaints about outdated or misleading price tags. Their findings stress the importance of keeping pricing information accurate and accessible to shoppers. By addressing one of the most common issues in traditional shopping, barcode systems paved the way for other tools that integrate pricing with convenience.

Further innovations came with the development of touchscreen kiosk systems for grocery stores, as described by Lopez, Cruz, and Mendoza (2021). Their study emphasized user-friendliness and accessibility, noting that even first-time users found the kiosks simple to operate. This kind of design showed that interactive technologies can appeal to a wide range of customers when presented in an intuitive manner. The presence of kiosks in stores demonstrated that digital platforms could extend beyond pricing to provide storewide assistance.

Alongside kiosk-based systems, researchers have also turned to low-cost hardware to improve retail operations. Garcia and Ramos (2022) investigated the use of Raspberry Pi as a local server for small businesses, where it successfully processed transactions and managed product data over a LAN without relying on continuous internet. Their findings revealed that this approach minimized costs while maintaining reliable access to information. Such examples suggest that affordable devices can deliver stable retail solutions while supporting connectivity in localized settings.

Commercial supermarkets have also experimented with in-store technologies, most notably the introduction of “smart carts” in several AllDay branches (BusinessWorld, 2022). These carts automatically scanned items placed inside and displayed both the price and total amount as shoppers moved through the aisles. Customers could then proceed to check out by scanning a generated QR code, which sped up the payment process. This approach showed how local supermarkets were adopting automation to merge convenience with customer engagement on the sales floor.

Broader adoption of digital retail solutions has been explored by Villanueva (2023), who examined Philippine supermarkets integrating product locators and automated pricing systems. The study found that these solutions enhanced customer satisfaction while also reducing inefficiencies in store management. Retailers benefited from smoother operations, while shoppers gained faster access to information about product availability. These results indicate that technological integration is gradually shaping consumer expectations in the local retail sector.

Research at De La Salle University investigated the potential of Smart Shelf Systems using embedded sensors and cameras to track product interactions (De La Salle University, 2024). The system recorded how customers picked up, returned, or compared items, providing valuable data on shopping behavior. Such insights enabled better shelves

organization, inventory tracking, and marketing strategies based on shopper activity. This study highlights how advanced monitoring tools can complement existing retail technologies, pointing toward a more data-driven approach to managing customer needs.

Review of Related Literature (Foreign)

Shopping convenience has a big effect on how customers behave. Studies show that making it easy for people to shop is important for keeping them as customers. Morris (2020) found that 62% of shoppers leave their carts behind if they face problems while shopping. This shows how even small issues can interrupt the shopping process and make people unhappy. If customers cannot find what they need quickly or have trouble moving around big stores, they might leave without buying anything. This proves how important it is to make shopping simple and smooth to keep customers satisfied.

According to Nicasio (2021), making shopping simple is very important to keep customers loyal. He explains that fixing problems and making the process easier can make shopping more enjoyable. This matters because people today want quick and efficient service. When customers can find what they need easily, they feel happier and less stressed. Helping them locate items quickly also builds better relationships between the store and the shoppers. A smooth shopping experience makes customers want to return and recommend the store to others.

The success of Amazon Go shows how technology can make shopping easier and faster. Research on Amazon Go shows that its cashier-free system helps customers spend less time waiting, making shopping quicker and more efficient (Samrat et al., 2023). Although supermarkets in the Philippines have not yet adopted this kind of technology, adding features to help customers find products more easily could bring local stores closer to this level. This shows how technology can make shopping more convenient and improve the traditional shopping experience.

As mentioned by Ryder (2023), in-store shopping is still a key part of the customer experience. He points out that today's shoppers want both speedy and personalized service.

Retailers need to find ways to offer both, so customers can easily find what they need while also feeling supported with a personal touch. Improving navigation and helping customers locate products faster does not mean losing that personal experience; instead, it can create a more customized shopping journey by meeting each customer's needs better. By using technology, stores can give help, letting customers make smarter choices. This mix of speed and personal service can increase customer satisfaction and loyalty, benefiting both the shoppers and the store.

Advancements in grocery store technology, like self-checkout systems, have reduced the time customers spend in stores. IT Retail (2023) asserts how these innovations give customers more control over their shopping, making the experience faster and more convenient. Similarly, technology that helps shoppers find products more easily can boost this sense of control, allowing them to shop with more independence and efficiency. This not only makes the shopping process quicker but also improves overall satisfaction by cutting down on the time spent looking for products.

Review of Related Literature (Local)

The integration of smart shopping technologies has been transformative in enhancing the grocery shopping experience. AllDay Supermarket pioneered the use of smart shopping carts in the Philippines, enabling customers to scan items, monitor their total expenses in real time, and complete transactions via QR code without visiting checkout counters. This innovative approach aligns with the global trend toward contactless shopping, providing consumers with more streamlined and hygienic experiences (Malapo, 2022).

The COVID-19 pandemic significantly accelerated the adoption of such technologies, emphasizing hygiene and convenience as critical factors in shopping. Ta-Asan (2024) reported that in 2023, digital payments constituted 53% of retail transactions in the Philippines, demonstrating a growing reliance on technology to simplify shopping. This shift reflects a broader effort by retailers to create efficient and connected shopping environments that cater to evolving consumer needs.

Supermarkets in the Philippines have increasingly adopted digital tools to improve customer satisfaction. Royandoyan (2022) highlighted that malls now integrate shopping, entertainment, and dining to attract tech-savvy customers. Features like real-time price updates and interactive displays help consumers access information quickly, meeting their expectations for personalized and efficient shopping experiences.

Keenan (2024) emphasized the importance of advanced shopping cart systems in transforming retail. Technologies such as inventory tracking, automated discounts, and personalized recommendations contribute to operational efficiency and customer satisfaction. These features not only enhance the shopping experience but also address modern consumer demands for tailored services.

The integration of mobile apps and cloud-based inventory systems further bridges the gap between physical and online shopping. According to Coghill (2022), tools such as digital coupons, synchronized inventories, and online ordering ensure customers have access to product availability and pricing. These features foster customer loyalty by making the shopping process more transparent and convenient.

Despite these advancements, the adoption of smart shopping systems remains limited across Philippine supermarket chains. This gap presents an opportunity to develop innovative solutions, such as product locators and dynamic aisle mapping, to enhance in-store navigation and streamline the shopping experience. By building upon existing technologies, supermarkets can meet the increasing demand for efficient, personalized, and technology-driven retail experiences.

TECHNICAL BACKGROUND

Overview of the Proposed System

Supermarkets continue to grow and produce variety, making it important to improve how customers navigate stores and manage their shopping. While some supermarkets use digital price checkers and self-checkout systems, these tools focus on payment convenience rather than helping customers find products or track their expenses effectively.

To address these gaps, ShopEase is designed as a smart shopping assistant that enhances the way customers shop. The system consists of a touchscreen display attached to shopping carts, providing shoppers with guidance to locate product's aisle quickly. Through indoor navigation and aisle mapping, ShopEase guides users to the specific aisle where a product is located, reducing search time and enhancing overall shopping efficiency.

Another key function of ShopEase is its barcode scanner, which allows customers to scan products as they shop. This enables them to check prices instantly, add or remove products from their cart digitally, and track their total spending in real time. With this feature, shoppers can stay within budget without needing to do manual calculations.

To store administrators, ShopEase provides a web-based management tool that makes product information and pricing right. The system also gives updates hence the shoppers can get the right aisle location and price information each time they use this platform. The system is also Wi-Fi enabled and therefore maintains data consistency across devices to ensure smooth running of the system.

These features make ShopEase take the in-store shopping experience to a more organized, efficient and stress-free experience. It is not only useful as it allows shoppers to locate products and control costs but also enables store operations such as enhance inventory accuracy and customer service. Therefore, the supermarkets are able to proceed to a smarter and more customer-conducive shopping environment.

System Components

Software Components

The software components of this project are essential for making the system work. Each software is chosen to ensure the user interface, hardware, and processes work together efficiently. The table below shows the software used and its purposes.

Software Components	Description
Frontend Development: HTML, CSS, JavaScript 	These are used to design and develop the user interface displayed on the touchscreen. HTML structures the content, CSS styles the interface, and JavaScript adds interactivity.
Backend Development: PHP, Node.js 	PHP is used for handling the backend, like managing the database, processing forms, and displaying data on the website. Node.js is good for real-time updates and handling many requests at once, like live tracking of products.
Backend Development: Python 	Python is used in the backend to handle data processing and compute the mapping routes. It ensures smooth communication between the database and the interface, providing efficient mapping functions.
Backend Development: C++ 	C++ was used in setting up the ESP32 to provide better performance, response, and full control over hardware functions such as Wi-Fi and communication with the backend.

Integrated Development Environment (IDE): Visual Studio Code  Visual Studio Code	Visual Studio Code (VS Code) is easy to use, fast, and supports many programming languages like PHP and JavaScript.
Database Management: 	Apache serves as the local web server that allows us to run and test our system. MySQL handles all data storage, including product details, prices, and cart information. To manage and organize our database easily, we used phpMyAdmin , which provides a user-friendly interface for creating tables, running queries, and monitoring data.

Table 1.0. Software Components

Hardware Components

The hardware parts listed below are important for the system to work. Each part has a specific job that helps the system run properly and lets the user interact with it. The table explains each hardware part and its use.

Hardware Components	Description
	Raspberry Pi is the main part of the ShopEase, acting as its main processing unit and connecting all the features together. It handles data to map or locate the store aisles, helping shoppers find their products easily. By linking to the store's database through Wi-Fi, and other software features of the device to work together, it keeps everything running

	smoothly, making the shopping experience more efficient and user-friendly.
IPS DSI 800×480 Pixel Screen	The IPS DSI Display serves as the main interactive interface that improves the shopping experience by showing product information and location. It acts as the visual part of the system, displaying navigation routes, product prices, and aisle maps to help customers navigate the store. The screen also helps the shoppers to customize the products in their cart in real-time.
Pi5 UPS 5v Li-Battery	This is a dependable power source for the Raspberry Pi in the project. Its main job is to keep the system running smoothly, even during power surges or short outages. This is to ensure that the device and its features continue to work without interruption. It also adds portability, allowing the smart cart to operate without needing to be plugged into a fixed power source, which improves the overall user experience.
ESP 32	The ESP32 is used in ShopEase as a fixed wireless anchor that helps determine the shopper's indoor location. It broadcasts a unique Wi-Fi signal from each aisle, which the Raspberry Pi detects to estimate distance and position. Its low cost, built-in Wi-Fi, and energy efficiency make it ideal for creating a reliable indoor mapping network.

Barcode Scanner 	<p>The barcode scanner is an added feature helping the shoppers with budget management. By letting customers scan items as they add them to their carts, automatically updating the system with the product's price, quantity, and total cost. This allows shoppers to keep track of their spending and stick to their budget throughout their shopping trip. The scanner also instantly verifies product details, reducing the need for manual price checks or assistance from store staff.</p>
--	--

Table 1.1. Hardware Components.

Block Diagram

This block diagram illustrates the core components of the ShopEase system and their interactions. Each block represents a key function or hardware component that contributes to the overall operation of the system, enabling users to navigate the supermarket, check prices, and manage their shopping experience efficiently.

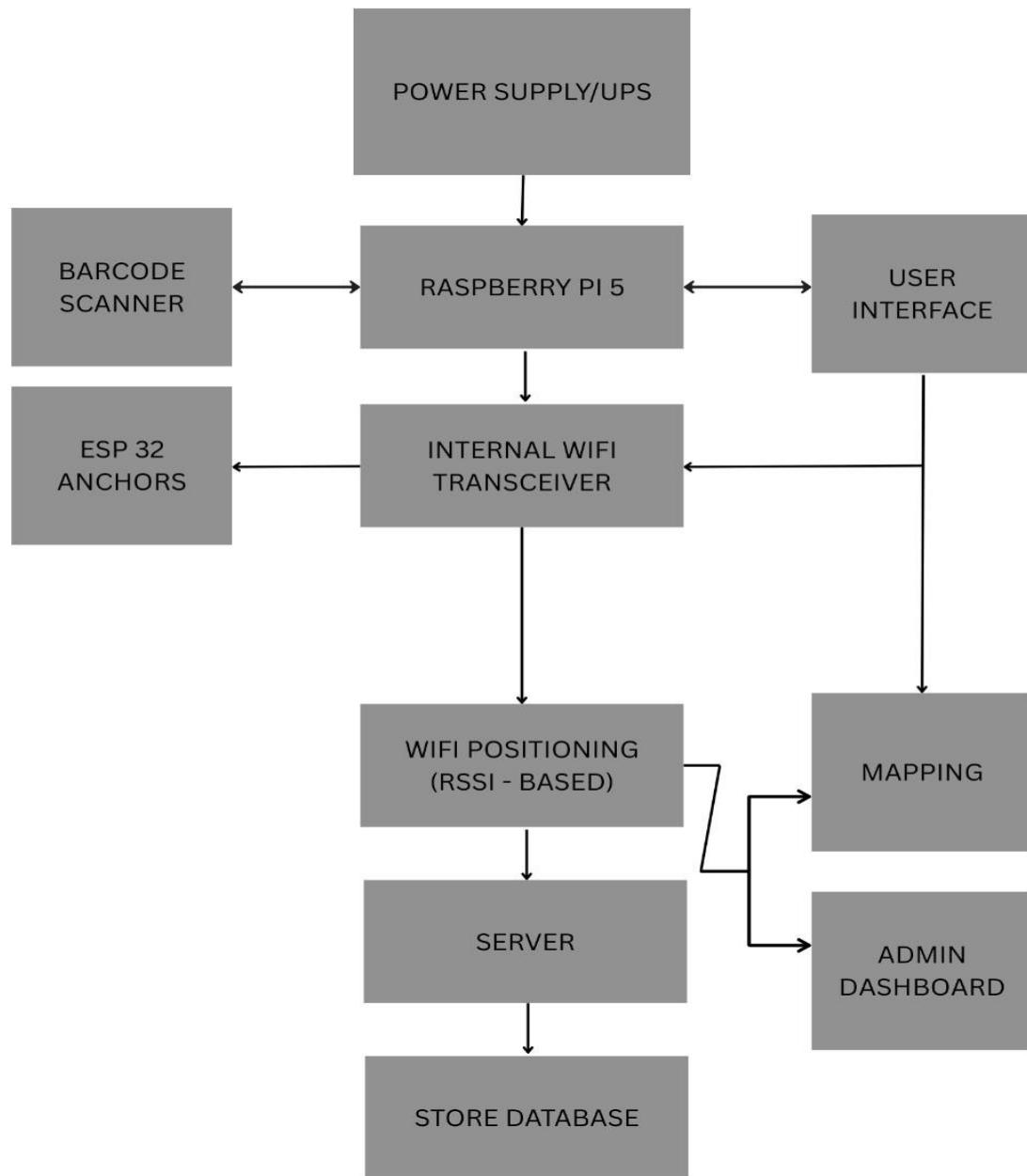


Figure 1.0. ShopEase' Core Components Block Diagram

METHODOLOGY

Requirements Analysis

The project is designed for use in medium to large-format retail environments such as supermarkets, hypermarkets, and superstores. Its primary users are supermarket customers who require efficient in-store navigation, product information, and budget assistance. The system is also intended to support store administrators, managers, and owners by simplifying inventory visibility, updating product placement, and improving customer service.

ShopEase project solves the most important problems of the modern world of retail, including the inability to find what one needs, time-consuming service, and the absence of real-time prices. ShopEase allows shoppers to navigate the aisles with a smart cart-mounted tablet device, scan barcodes to see product prices immediately, and monitor overall cart value to ensure they do not exceed their budget with an inner mapping feature. In the meantime, a database system will be available at the back end to enable store administrators to have an opportunity to maintain product information, store locations, and inventory information in real time.

Designed for continuous use during store operating hours, ShopEase synchronizes data through Wi-Fi and communicates with ESP32 anchors placed in each aisle for indoor positioning. The project aims to reduce shopping time, enhance customer satisfaction, and streamline store operations through automation and digital integration.

Requirements Documentation

The following tables outline the functional and non-functional requirements of the ShopEase device. These are divided into hardware and software components and represent the minimum capabilities that must be fulfilled.

Hardware	Status
Functional Requirements	
1. The device must allow customers to scan product barcodes and update cart contents.	Completed
2. The device must support real-time cart total tracking to aid shopper budgeting.	Completed
3. The device must synchronize data with the database via Wi-Fi.	Completed
4. The device must support user interaction through responsive touchscreen input.	Completed
5. The device must provide responsive navigation assistance using an ESP32-based indoor positioning system to guide users within the store.	Completed
6. The device must display product's aisle location and price data through a touchscreen interface.	Completed
Non-Functional Requirements	
7. The device's internal system must operate efficiently with minimal lag.	Completed
8. The device must maintain stable operation during store hours without overheating or crashing.	Completed
9. The touchscreen and housing must be durable and tamper-resistant for high-traffic environments.	Completed
10. The power system must support extended usage and prevent data loss during minor outages.	Completed
11. The hardware must support modular maintenance for easy repair or component replacement.	Completed

Table 1.2. Hardware Requirements

The succeeding tables outline the functional and non-functional requirements for the ShopEase software, which includes both the cart-mounted interface and the administrator's web-based platform. These requirements define the key features and performance standards the software must achieve to ensure seamless shopping and management experience.

Software	Status
Functional Requirements	
1. The UI must allow users to search for products by name or category.	Completed
2. The system must display the aisle number and route to the selected product.	Completed
3. The application must calculate and display the total value of selected items.	Completed
4. The system must allow barcode input to show price and product info.	Completed
5. The system must include instructional/tutorial features for user guidance.	Completed
6. The admin dashboard must allow real-time product data updates.	Completed
7. Admins must have tiered access control for database security within the XAMPP and WSS HTTPS server environment.	Completed
8. The main admin has full control, while managers and inventory staff can edit products and aisles but have no access to account management.	Completed
9. The system must include logging and activity tracking for admin and user actions.	Completed
Non-Functional Requirements	
10. The UI must be user-friendly, intuitive, and accessible to first-time users.	Completed
11. The database must be secure, encrypted, and access controlled.	Completed
12. The system must update product and cart data in real-time with minimal delay.	Completed
13. The system must be scalable to handle multiple users and devices simultaneously.	Completed

Table 1.3. Software Requirements

Design of Software, System, Product, and/or Processes

ShopEase is the supermarket cart system which is designed with three key aspects. Software, system and product design, all which are implemented to enhance the overall customer shopping experience. The system will address some of the challenges that the shoppers encounter which include inability to locate specific products, insufficient information and wastage of time in moving around the store. ShopEase is efficient and convenient in the way it locates products and reduces the time spent in trying to find the information by incorporating intelligent technology. All these aspects are backed up by a dynamic development process that is managed by the Agile approach that encourages iterative development and quick testing. This practice enabled the team to change according to feedback and incorporate the challenges such as user navigation, system responsiveness and functional reliability during the project lifecycle.

- The **Software Design** of ShopEase includes two main components: the customer-facing cart display and the admin dashboard. It centers on a simple and user-friendly interface that enables shoppers to search for specific products or browse by category. The system retrieves data from a store's centralized product database and displays important information such as item name, price, and location within the store. The interface is optimized for touch input and uses large buttons, intuitive icons, and a minimal layout to make it easy for shoppers of all ages. The system includes three major modules: the Search Module, which allows users to enter item names or categories; the Display Module, which shows item details and price on the screen; and the Navigation Module, which provides directional assistance or maps to guide users to the correct aisle or section. The admin dashboard enables authorized personnel to update product data, manage item locations, and monitor the system using role-based access control. The design follows modern usability standards like ISO/IEC 25010:2011, prioritizing security, maintainability, and user experience. According to Smart Insights, 97% of consumers abandon purchases due to poor usability, while Forrester Research shows that better user interfaces can boost conversion rates by 200% and satisfaction by 400%. These insights highlight the importance of our user-centered, intuitive design for both effectiveness and long-term success.

- The **System Design** integrates both hardware and software components into one unified device mounted on a supermarket pushcart. At the core of the system is the Raspberry Pi, which serves as the main processing unit that gathers location data from ESP32 modules positioned in store corners and communicates with the store's database via Wi-Fi. As users scan products using the barcode scanner, the system instantly updates their digital shopping list with item details and running totals. The system also includes basic indoor positioning or directional features that help shoppers navigate within the store. The entire system is powered by a rechargeable battery pack and is securely installed on the pushcart handle for easy access and durability. This design ensures that all components work together to support the main features of the product while being practical for everyday supermarket use.
- The **Product Design** of ShopEase is made to be compact, durable, and easy to use. The touchscreen display is placed in a convenient position where shoppers can easily interact with it without interrupting their shopping flow. The housing is built from lightweight but sturdy material to protect the internal components and ensure long-term use. Following IP-rated design principles, the housing is sealed to resist dust, liquid spills, and frequent handling. All ports and seams are reinforced to reduce the risk of water loss or contamination. Internally, wiring and electronic components are insulated and tamper-resistant to minimize electrical hazards and ensure reliable performance throughout extended use. The device is powered by a rechargeable battery pack, securely enclosed to prevent overheating or short circuits, always ensuring shopper safety. The user interface is clean and minimal to avoid confusion, and the entire design allows for quick maintenance or replacement if needed. The product aims to blend with the natural behavior of shoppers, acting like a smart guide as they move through the supermarket.

Development and Testing



Figure 1.1. Agile Methodology Process

The development of ShopEase follows the Agile methodology, a process model characterized by its iterative, flexible, and user-focused approach. Agile is particularly suited to the nature of the ShopEase system, as it supports ongoing refinement in response to user feedback, evolving requirements, and technological changes. This methodology enables the development team to deliver a functional, user-centric solution in manageable increments throughout the project lifecycle.

- **Planning:** The project begins with a comprehensive planning phase, during which the scope and core objectives of ShopEase are defined. This includes identifying essential system features such as product search, aisle navigation, and the development of an administrative dashboard for managing product availability and location data. Tasks are broken down into smaller units and prioritized based on importance and logical dependencies, establishing a structured roadmap to guide development activities.

- **Design:** The design phase focuses on developing detailed layouts and structural blueprints for both the user-facing interface and the administrative dashboard. The customer interface is designed for ease of use, allowing shoppers to search for items, view their availability, and locate them within store aisles. Simultaneously, the design of the admin dashboard prioritizes clarity and security, incorporating role-based access controls to ensure that only authorized personnel can update product records and assign aisle locations. Usability, accessibility, and system responsiveness are primary design considerations.
- **Development:** Development proceeds in accordance with the established design specifications. The front-end interface is implemented to facilitate product search and location tracking, while the back-end system supports data management and functionality integration. The admin dashboard is also developed during this phase, allowing authorized users to manage product entries, assign item locations, and update availability status. Although the system does not track inventory quantities, the admin panel enables manual control of whether an item is marked as “Available” or “Not Available”.
- **Testing:** Once key components were developed, the system underwent thorough testing to verify its functionality and performance. Testing covered both customer-facing features and the administrative interface, ensuring that product search, location display, and availability updates worked as intended. The evaluation was conducted through a mock-up floor planning setup in the school lobby, simulating an actual supermarket environment to observe navigation and system behavior. During this phase, ShopEase’s smart pushcart devices that equipped with Raspberry Pi units, 7-inch touchscreens, and barcode scanners that were tested to ensure stable operation and responsiveness. The development team verified that all features met the expected performance standards, including secure synchronization with the Wi-Fi-connected database and the role-based web dashboard.

The software aspect of ShopEase was also evaluated by a Software Quality Assurance specialist, who assessed the system’s usability, responsiveness, and overall functionality. This evaluation confirmed that the software met quality standards and operated efficiently under normal conditions. Additionally, an Industrial Designer examined the device’s design, confirming its functionality and providing recommendations for further improvement. Any identified issues were addressed promptly to enhance system stability and user experience.

- **Deployment:** After successful testing, ShopEase will be deployed in a real-world store environment. Customers access product search and locator features via interactive pushcart-mounted devices, while store administrators utilize the admin dashboard for product data management. A Certified Systems Engineer (Project Lead) and a Network Security Specialist (CISSP) oversee deployment, ensuring proper system configuration and secure access protocols. Deployment activities include device setup, database synchronization, role-based access control implementation, initial user training, and system performance monitoring to ensure smooth and secure operation.
- **Review:** Post-deployment feedback is gathered from end-users to assess system effectiveness and usability. This feedback informs future updates, with the development team making necessary refinements based on observed performance and user suggestions. The review phase is essential to Agile, as it ensures the continuous enhancement of the system in alignment with user expectations and operational needs.

Testing Procedures

To ensure that ShopEase meets its intended functionality, reliability, and performance goals, the system is subjected to several structured testing methods. These are designed to validate both individual components and the system as a whole under realistic conditions:

- **Unit Testing:** Individual software features such as the product search, aisle locator, and administrative dashboard were tested independently to verify proper functionality. Hardware components, including the pushcart-mounted display and barcode scanner, also underwent isolated testing to ensure reliability. This phase was conducted by the development team to confirm that each module and device worked correctly when tested separately.
- **Integration Testing:** After successful unit testing, the software and hardware components were combined and tested as a single unified system to ensure seamless interaction. This included validating the product lookup, testing the functionality of the aisle guidance, and reviewing administrative updates within the fully integrated ShopEase platform. Integration testing confirmed that all components coordinated effectively to deliver a stable and consistent user experience.

- **Performance Testing:** The system's responsiveness and stability were evaluated under various conditions, including periods of high user activity and simultaneous administrative operations. This phase aimed to identify potential performance limitations and confirm that ShopEase could maintain reliable operation during extended use. The development team monitored performance metrics and applied necessary optimizations to improve responsiveness, speed, and overall efficiency.

Description of Prototype

The smart pushcart device, which includes a Raspberry Pi, a 7" touchscreen, and a barcode scanner, must be correctly connected to a stable power supply and configured to communicate with the central system via Wi-Fi.

- Users (shoppers) interact with the device through the touchscreen interface to search for products, check availability, and receive aisle navigation guidance.
- The web-based admin dashboard requires internet access and login credentials, allowing authorized store personnel to update product listings, mark item availability, and manage aisle location data securely.
- The dashboard supports role-based access, ensuring that only authorized users can modify system records.

The following figures illustrate the structural and functional design of the ShopEase system. These visual representations provide a comprehensive understanding of the system's hardware layout, software flow, and overall interaction between components.

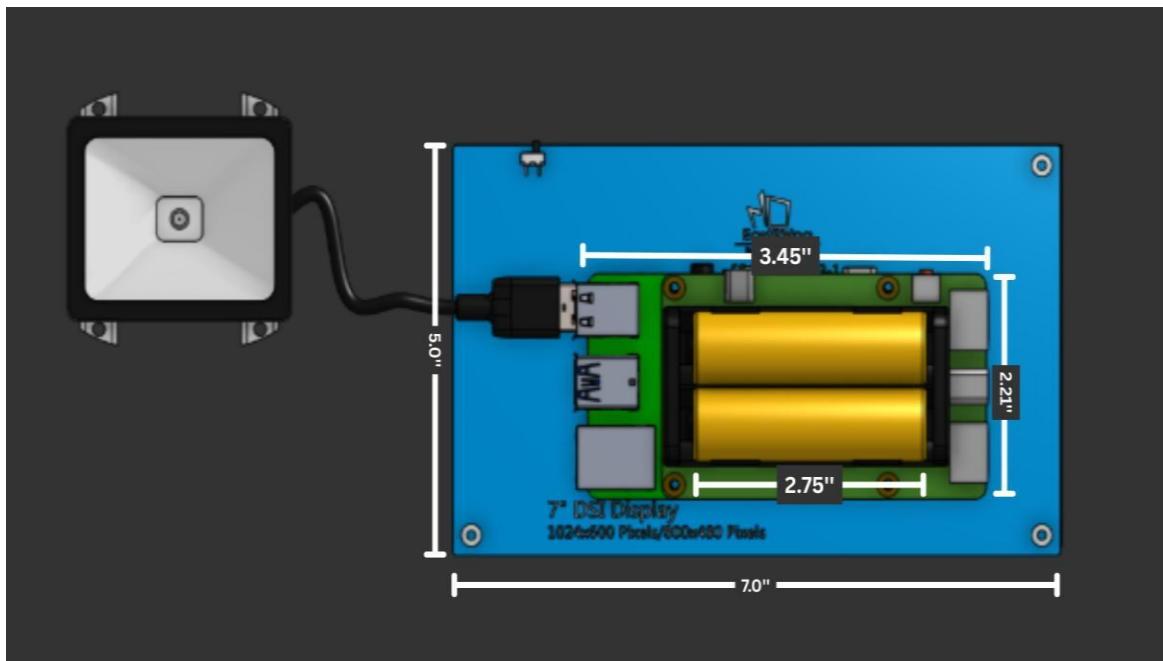


Figure 1.2. 3D Model of ShopEase Device Back-View & Scanner Front-View.

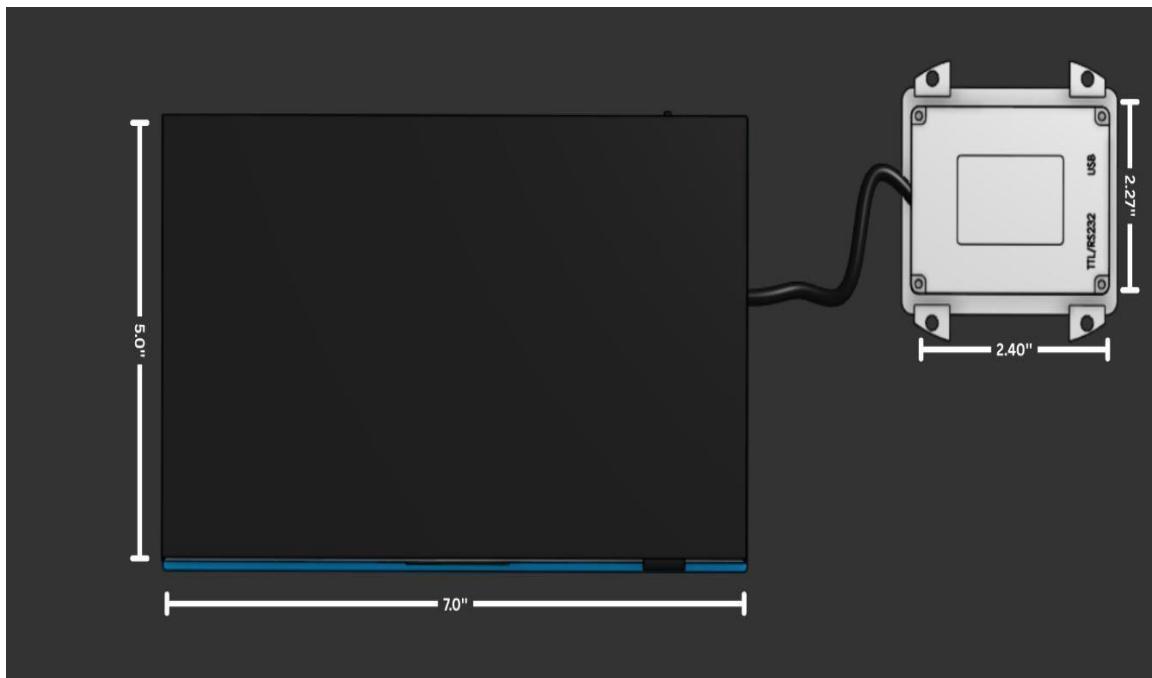


Figure 1.3. 3D Model of ShopEase Device Front-View & Scanner Back-View.

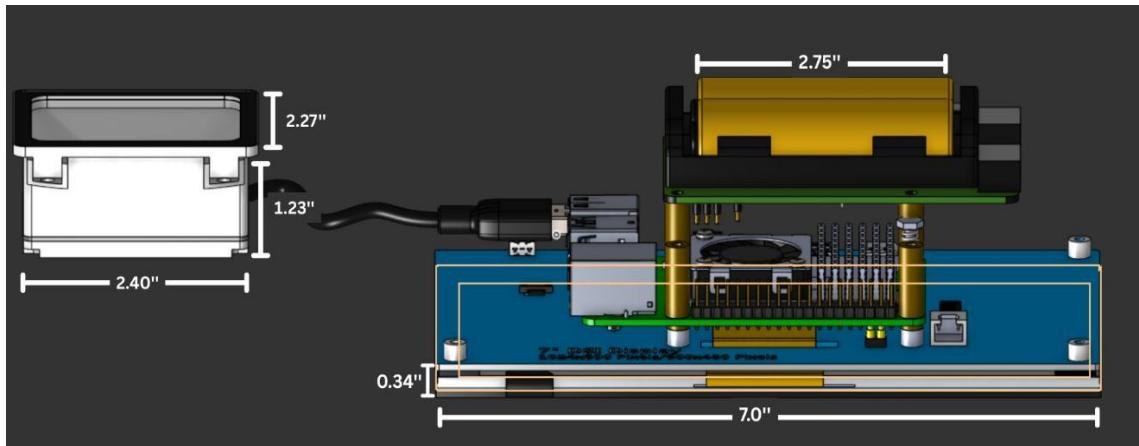


Figure 1.4. 3D Model of ShopEase Device & Scanner Top-View.

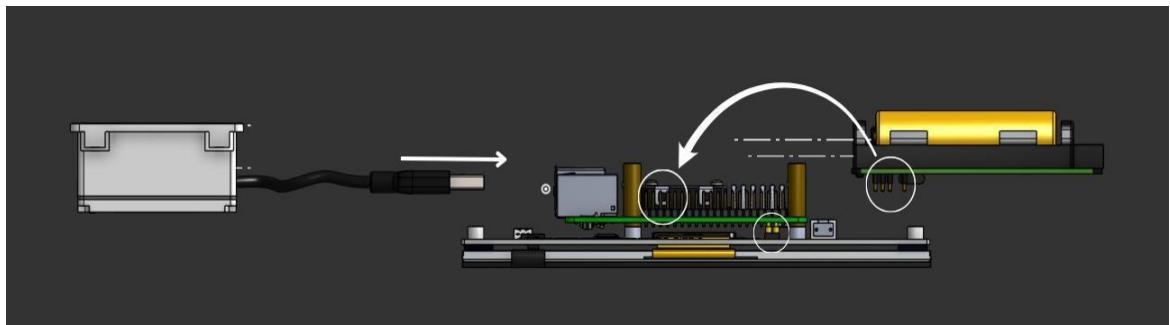


Figure 1.5. 3D Model of the ShopEase Hardware Setup Diagram.

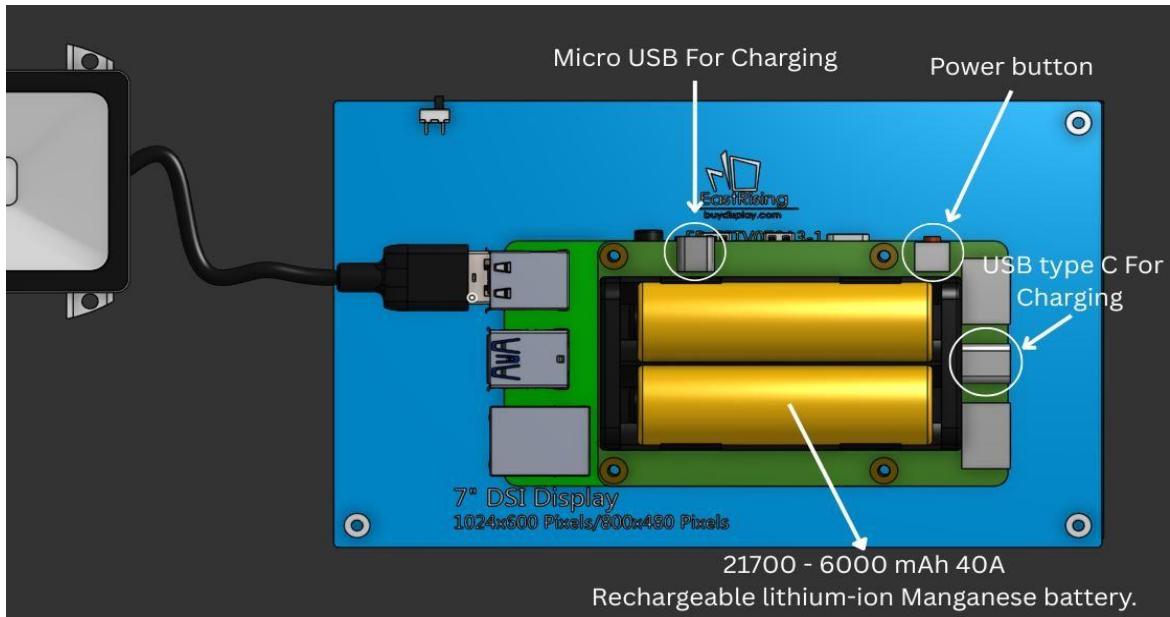


Figure 1.6. 3D Model of ShopEase Device Hardware Components.

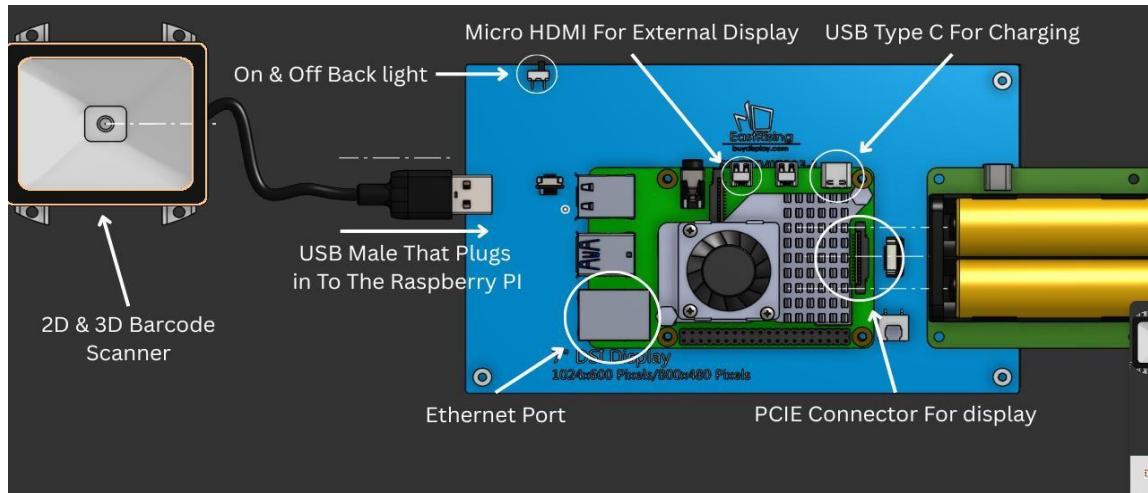


Figure 1.7. 3D Model of the ShopEase Hardware Interfaces and Connectivity.

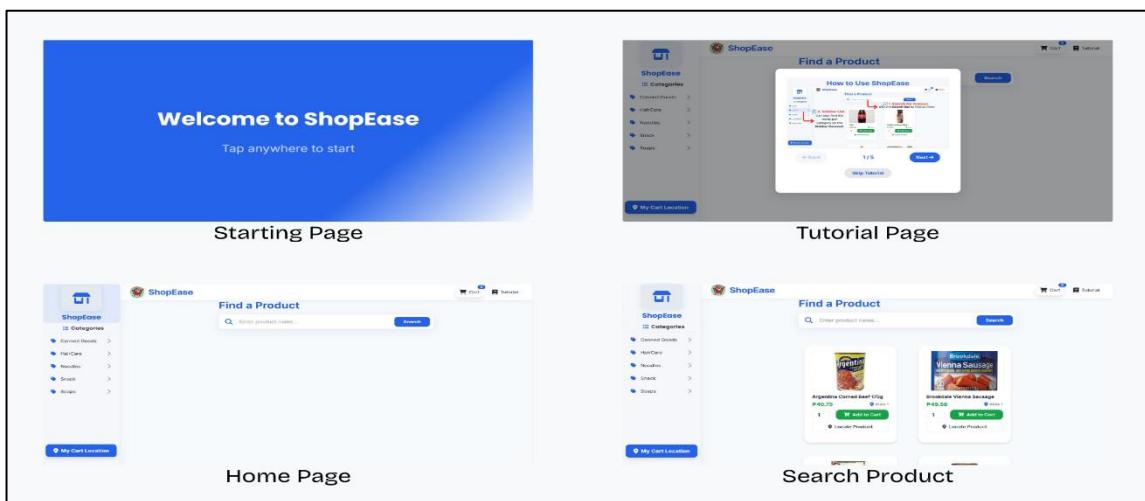


Figure 1.8. User Search Page.

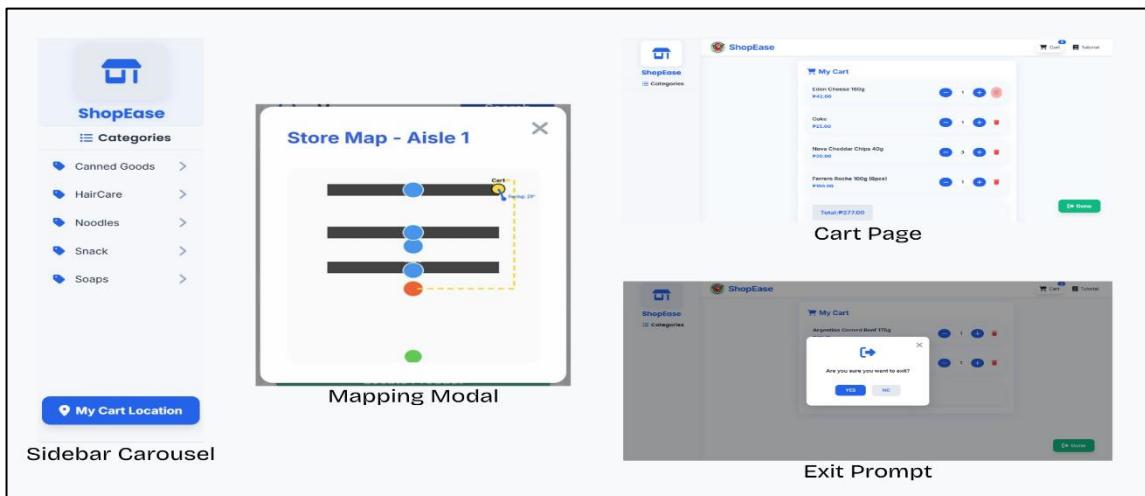


Figure 1.9. Additional Features of User Interface.

Login Page

Home Page

Aisle Management Page

Product Management Page

Figure 2.0. Admin Management Page.

Password Reset Request

Password Reset Request (All Request)

Password Reset Request (Pending Request/s)

Figure 2.1. Password Reset Request.

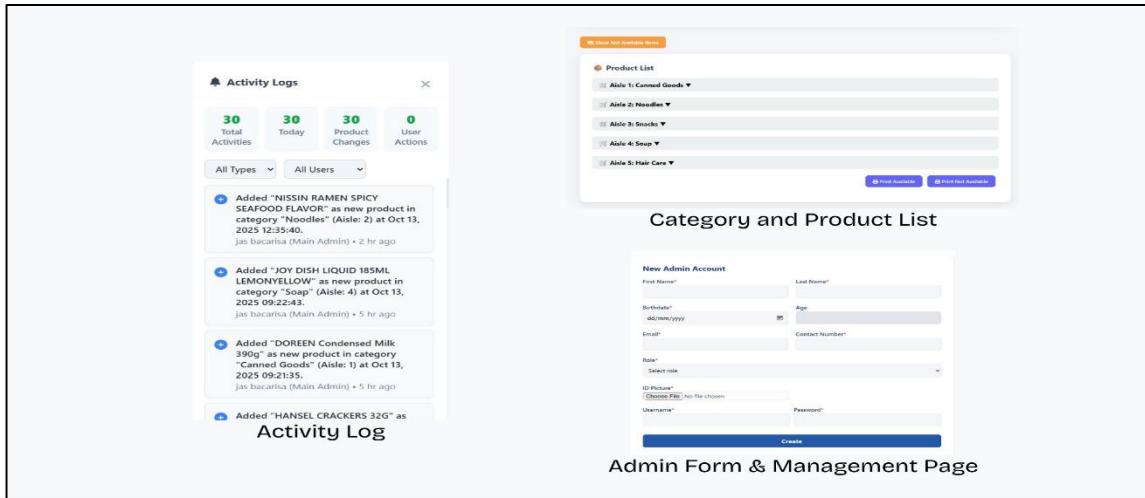


Figure 2.2. Admin's Log, Aisle/Product Lists and Admin Form.

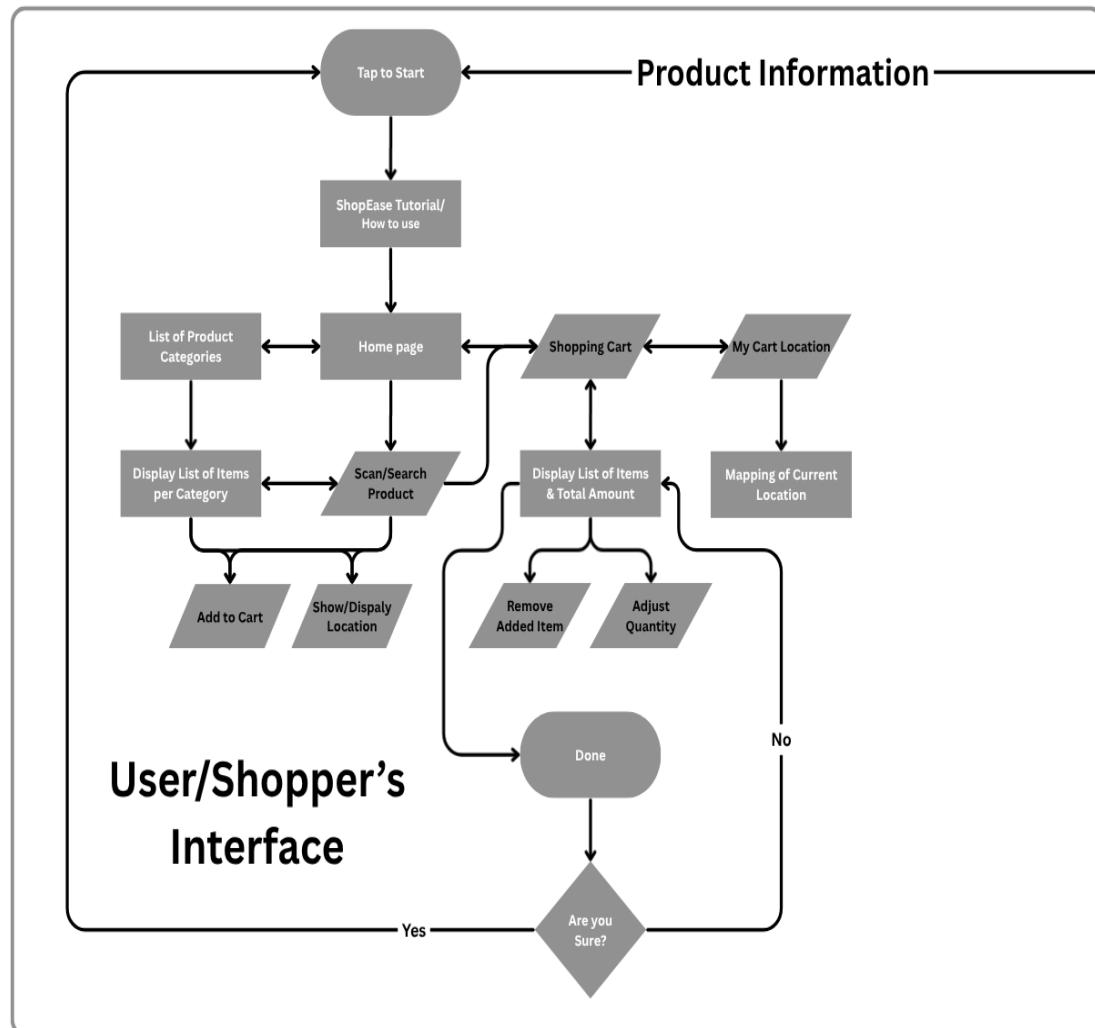


Figure 2.3. User's Interface Flowchart

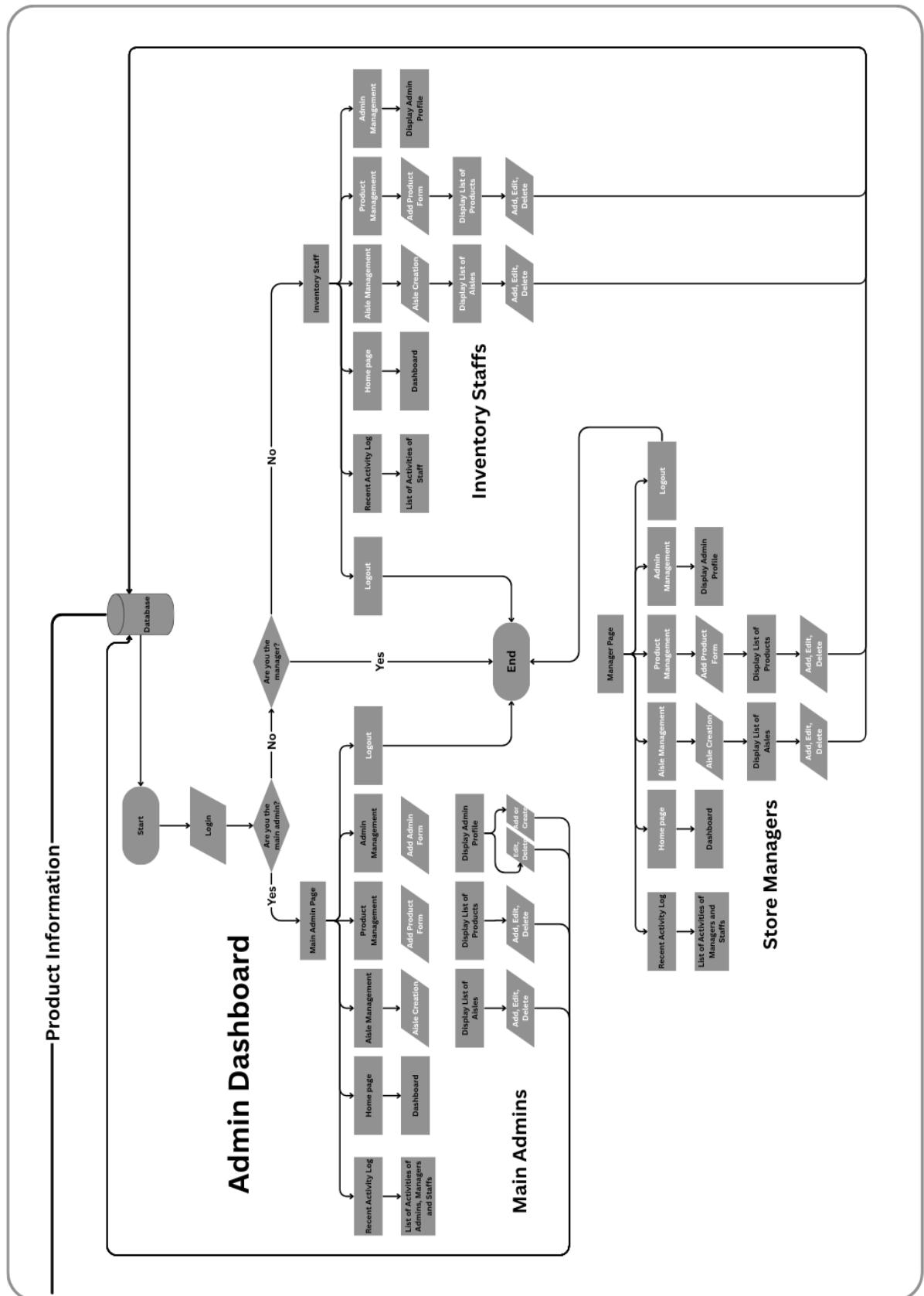


Figure 2.4. Admin Dashboard Flowchart

Implementation Plan

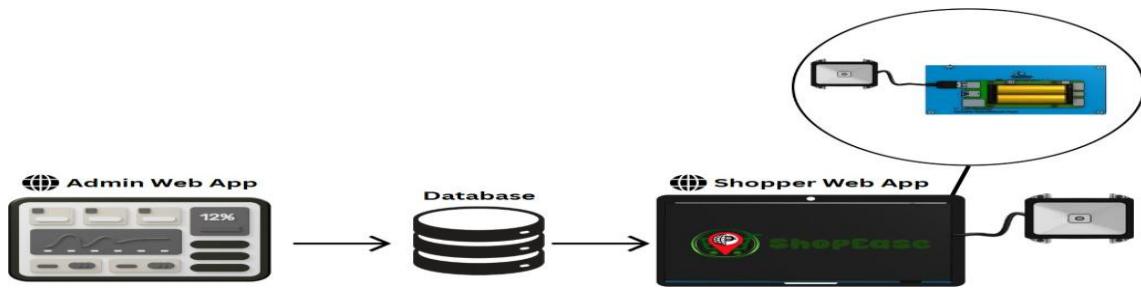


Figure 2.5. *ShopEase' Implementation Plan.*

The diagram above illustrates how the ShopEase system was implemented for testing in a mock-up supermarket setup. The hardware components, including the Raspberry Pi 5, IPS touchscreen display, barcode scanner, and Pi5 UPS battery, were assembled and mounted on a pushcart. ESP32 modules were strategically placed in each aisle of the mock-up environment to act as wireless anchors, enabling indoor navigation. The barcode scanner was configured to instantly retrieve price and product details from the local database, while the touchscreen interface displayed navigation routes, cart totals, and product information.

The software side of ShopEase was also integrated during this phase, ensuring seamless coordination between the system's components. The user interface was developed using HTML, CSS, and JavaScript to maintain simplicity and responsiveness, while PHP and Node.js handled backend operations to provide secure and communication with the database. To support location processing and data handling, Python was utilized in the backend to analyze the fetched MAC addresses and compute the device's current location within the supermarket. Meanwhile, C++ was used in setting up the ESP32, which continuously broadcasts its MAC address via Bluetooth. The Raspberry Pi then detects and fetches these broadcasts and transmits the data to the backend for processing, allowing synchronization between hardware and software components. The admin dashboard, designed for store personnel, enabled managers to update product locations, adjust prices, and monitor system activity through tiered access.

After integration, the system underwent functionality and performance testing to verify barcode scanning accuracy, responsiveness of the navigation module, and synchronization speed between the admin dashboard and customer-facing devices. Once validated, the

system was deployed in the mock-up supermarket setup for one full day of testing. Participants acted as shoppers using ShopEase-equipped carts, while the team monitored performance and usability. The testing phase concluded with evaluation and refinement, where participant feedback, system logs, and team observations were analyzed to identify areas for improvement.

Implementation Results

A mock-up supermarket was used to test the ShopEase system to test its functionality in a real shopping situation. The implementation was in terms of the accuracy of navigations, reliability of the barcode scanners, responsiveness of the system, and the feedback to users. Various product searches were done in the course of testing, and the system was able to direct the users to the right aisles efficiently. Simulated congestion of Wi-Fi led to minor delays (1-2 seconds) but no aborted navigations.

The Barcode reader was tested on different products in different categories, and the rate of accuracy was 100% of the real-time price information retrieval at an average response time of 0.5 seconds. This was more efficient and quicker than the normal price-checking models. Responsiveness wise, the system registered an average latency of 0.18 seconds to 0.23 milliseconds during database to system synchronization, and the touchscreen was also quick in responding to each tap at 0.8 seconds, making the system accessible to the user. The battery pack was also designed to last the device 24 active hours, and it was possible to power the device continuously during testing.

There was also the collection of user feedback, with the majority of the respondents giving the system a high rating in terms of helping them find products faster and monitor expenses in real time. The most useful features listed by the participants were the accuracy of navigation and monitoring the budget, and some people noted that they could better improve the system, including the increased screen size and the option to use voice guidance with older users. Administratively, changes that were done in the dashboard were mirrored in a few seconds in the cart devices. Another observation made by the development team was that the activity logs and tiered access enhanced ease of use and manipulation in the process of testing.

RESULTS AND DISCUSSIONS

This chapter presents, analyzes, and interprets the data collected from the questionnaire responses of the participants. It highlights the feedback and insights provided by users and shoppers regarding the functionality, design, and usability of ShopEase. The discussions that follow aim to provide a deeper understanding of how the system meets its objectives and how user perceptions reflect its overall effectiveness.

Sources and Methods of Collecting Data

The questionnaire was distributed online through Google Forms to gather feedback from users and shoppers on the functionality and usability of the ShopEase system. Respondents were informed about the purpose of the study and guided on how to answer the questions. A total of 100 participants completed the survey, and the data collected were analyzed statistically to ensure accurate and reliable results. In addition, another survey was conducted specifically for the user testing of the ShopEase device to evaluate its performance and ease of use.

Statistical Treatment of Data

The study used the following statistical tools and techniques:

Percentage

The responses of the participants were analyzed using percentage to determine the distribution of their feedback and evaluation results. This method was used to assess the level of user satisfaction and the overall system performance of ShopEase based on the conducted testing and evaluation. The percentage was computed using the formula:

$$P = \frac{f \times 100}{N}$$

P = Percentage

F = Frequency

N = Total Number of Respondents

Shopper/User Data Analysis.

SECTION 1.0: RESPONDENT PROFILE

Age	Frequency	Percentage (%)
Below 18 years old	4	4%
18 – 24 years old	47	47%
25 – 34 years old	13	13%
35 – 44 years old	16	16%
45 years old and above	20	20%
Total	100	100%

Table 1.4. Age Group of Respondents.

The results show that most respondents are aged 18–24 years old (47%), followed by 25–34 years old (13%), 35–44 years old (16%), and 45 years old and above (20%). Only 4% are below 18.

Gender	Frequency	Percentage (%)
Male	39	39%
Female	61	61%
Total	100	100%

Table 1.5. Gender of Respondents.

The results show that 61% of respondents were female, while 39% were male. This means that women made up the majority of the participants, which is expected as grocery shopping is often done by female household members.

Variables	Frequency	Percentage (%)
Once a month	23	23%
Once a week	32	32%
Twice a week	18	18%
Thrice a week	5	5%
Occasionally	23	23%
Total	100	100%

Table 1.6. Frequency of Supermarket Shopping.

The results reveal that 32% of respondents shop once a week, 23% once a month, 18% twice a week, and 5% thrice a week. Another 23% said they shop occasionally.

Stores	Frequency	Percentage (%)
SM Supermarket	28	28%
SM Hypermarket	15	15%
Savemore	19	19%
Landers Superstore	3	3%
S&R Membership Shopping	1	1%
Puregold	34	34%
Total	100	100%

Table 1.7. Usual Supermarket Visited.

According to the results, 34% of respondents usually shop at Puregold, followed by 28% at SM Supermarket, 19% at Savemore, and 15% at SM Hypermarket. Smaller groups shop at Landers (3%) and S&R (1%).

Variables	Ranking	Percentage (%)
Lack of product category labeling per aisle	3	35%
Frequent changes in product location	4	30%
Difficulty comparing prices	1	52%
Difficulty keeping track of budget/expenses	2	49%
Confusing store layout	5	24%

Table 1.8. Reasons/Challenges Encountered in Large Supermarkets.

The data shows that the top challenge (Rank 1) is difficulty comparing prices (52%), followed by difficulty keeping track of budget/expenses (49%, Rank 2), and lack of aisle category labeling per aisle (35%, Rank 3).

SECTION 2.0: SHOPPING BEHAVIOR AND CHALLENGES

Variables	Ranking	Percentage (%)
Low Prices	1	65%
Finding products easily	2	55%
Tracking budget and expenses	4	51%
Checking product availability	3	53%
Price Comparison	5	41%

Table 1.9. Top Priorities When Shopping at a Supermarket.

The results reveal that the top priority (Rank 1) is low prices (65%), followed by finding products easily (55%, Rank 2), and checking product availability (53%, Rank 3).

Variables	Frequency	Percentage (%)
Always	13	13%
Sometimes	52	52%
Rarely	30	30%
Never	5	5%
Total	100	100%

Table 2.0. Asking staff for directions.

The data shows that 52% of respondents answered Sometimes, 30% answered Rarely, 13% answered Always, and only 5% answered Never. This means that the majority of shoppers at least sometimes rely on staff to locate products.

Variables	Frequency	Percentage (%)
Always	10	10%
Sometimes	61	61%
Rarely	28	28%
Never	1	1%
Total	100	100%

Table 2.1. Difficulty in Finding Items in the Store (Supermarket).

The results reveal that 61% of respondents answered Sometimes, 10% answered Always, 28% answered Rarely, and 1% answered Never.

Variables	Frequency	Percentage (%)
Always	30	30%
Sometimes	52	52%
Rarely	15	15%
Never	3	3%
Total	100	100%

Table 2.2. Keeping Track of Expenses.

The survey shows that 52% of respondents answered Sometimes, 30% answered Always, 15% answered Rarely, and 3% answered Never.

Variables	Frequency	Percentage (%)
Always	5	5%
Sometimes	45	45%
Rarely	42	42%
Never	8	8%
Total	100	100%

Table 2.3. Leaving the Store Without Buying an Item Because It Could Not Be Found.

The results indicate that 45% of respondents answered Sometimes, 42% answered Rarely, 5% answered Always, and 8% answered Never.

SECTION 3.0: FEEDBACK ON SHOPEASE CONCEPT

Variables	Frequency	Percentage (%)
Always	12	12%
Sometimes	19	19%
Rarely	9	9%
Never	60	60%
Total	100	100%

Table 2.4. Familiarity with Smart Carts.

The results show that 60% of respondents answered Never, meaning they have never seen or used smart carts in supermarkets. Meanwhile, 19% answered Sometimes, 12% answered Always, and 9% answered Rarely.

Variables	Frequency	Percentage (%)
Very Likely	54	54%
Likely	25	25%
Neutral	17	17%
Unlikely	2	2%
Very Unlikely	2	2%
Total	100	100%

Table 2.5. Interest in Using ShopEase Smart Cart.

The survey reveals that 54% of respondents answered Very Likely, 25% answered Likely, and 17% answered Neutral. Only 2% answered Unlikely, and another 2% answered Very Unlikely.

Variables	Ranking	Percentage (%)
GPS-like navigation to find items	4	56%

Real-time price display	1	76%
Virtual cart with total budget	3	65%
Barcode scanner for checking prices	2	70%
Touchscreen product search	5	51%
Directions to fastest route through store	6	46%

Table 2.6. Most Useful Features of ShopEase.

The results indicate that the top three features are real-time price display (76%), barcode scanner for checking prices (70%), and virtual cart with budget tracking (65%).

Variables	Frequency	Percentage (%)
Very Likely	57	57%
Likely	32	32%
Neutral	11	11%
Unlikely	0	0%
Very Unlikely	0	0%
Total	100	100%

Table 2.7. Comfort Level in Using a Touchscreen Device While Shopping.

The results show that 57% of respondents answered Very Likely, 32% answered Likely, and 11% answered Neutral. No respondents answered Unlikely or Very Unlikely.

Variables	Frequency	Percentage (%)
Very Likely	52	52%
Likely	35	35%
Neutral	13	13%

Unlikely	0	0%
Very Unlikely	0	0%
Total	100	100%

Table 2.8. Belief That ShopEase Could Reduce Shopping Time.

The survey shows that 52% of respondents answered Very Likely, 35% answered Likely, and 13% answered Neutral. No respondents answered Unlikely or Very Unlikely. This means that out of 100 respondents believe ShopEase would likely help reduce their shopping time, which proves that its efficiency feature is strongly relevant to shoppers.

SECTION 4.0: SUGGESTIONS AND EXPECTATIONS

Variables	Frequency	Percentage (%)
Integration with digital payment/e-wallets	28	28%
Voice-assisted navigation and commands	4	4%
Shopping list import	16	16%
Nutritional or dietary information for scanned products	16	16%
New arrival product notification	7	7%
Queue monitoring (shows shortest cashier/checkout line)	16	16%
Emergency assistance button	13	13%

Total	100	100%
--------------	------------	-------------

Table 2.9. Additional Features Respondents Would Like ShopEase to Have.

The results show that 28% of respondents selected Integration with digital payment or e-wallets, followed by 16% each for Shopping list import, Nutritional or dietary information display, and Queue monitoring. In addition, 13% answered Emergency assistance button, 7% chose new arrival alerts, and 4% answered Voice-assisted navigation.

Variables	Frequency	Percentage (%)
Device might be difficult to use for first-time users	25	25%
Reliability of Wi-Fi / internet connection	23	23%
Possibility of technical issues (errors, bugs, crashes)	43	43%
Device durability or battery life	9	9%
Total	100	100%

Table 3.0. Concerns About Using a Digital Shopping Assistant Cart.

The survey reveals that the highest concern among respondents was Possibility of technical issues (43%), followed by Device difficulty for first-time users (25%). Meanwhile, 23% selected Reliability of Wi-Fi or internet connection, and 9% chose Device durability or battery life.

Variables	Frequency	Percentage (%)
Very Likely	57	57%
Likely	31	31%

Neutral	12	12%
Unlikely	0	0%
Very Unlikely	0	0%
Total	100	100%

Table 3.1. Willingness to Recommend ShopEase to Others.

The results show that 57% of respondents answered Very Likely, 31% answered Likely, and 12% answered Neutral. No respondents answered Unlikely or Very Unlikely.

Variables	Frequency	Percentage (%)
Very Likely	60	60%
Likely	31	31%
Neutral	6	6%
Unlikely	1	1%
Very Unlikely	2	2%
Total	100	100%

Table 3.2. Likelihood of Using ShopEase if Available in Local Supermarkets.

The survey shows that 60% of respondents answered Very Likely, 31% answered Likely, and 6% answered Neutral. Only 1% answered Unlikely and 2% answered Very Unlikely.

Variables	Frequency	Percentage (%)
Reducing the time spent finding products	15	15%
Making prices clearer and easier to track	12	12%
Helping me stay within budget	5	5%

Making the overall shopping process convenient	6	6%
All of the above	62	62%
Total	100	100%

Table 3.3. Aspect of Shopping Experience That ShopEase Would Improve the Most

The results reveal that 62% of respondents answered All of the above, indicating that they believe ShopEase would improve multiple aspects of their shopping experience.

User Experience Testing Data Analysis.

SECTION 1.0: RESPONDENT PROFILE

Age	Frequency	Percentage (%)
Below 18 years old	1	6.3%
18 – 24 years old	4	25%
25 – 34 years old	4	25%
35 – 44 years old	4	25%
45 years old and above	3	18.8%
Total	100	100%

Table 3.4. Age Group of Respondents.

The results show that most respondents are aged 18–24 years old (25%), followed by 25–34 years old (25%), 35–44 years old (25%), and 45 years old and above (18.8%). Only 6.3% are below 18.

Gender	Frequency	Percentage (%)
Male	8	50%
Female	8	50%
Total	100	100%

Table 3.5. Gender of Respondents.

The data also shows that 50% of respondents are female and 50% are male, which indicates a balanced gender distribution among the participants.

SECTION 2.0: TESTING FEEDBACK

Variables	Frequency	Percentage (%)
Very Easy	5	31.3%
Easy	10	62.5%
Neutral	1	6.3%
Difficult	0	0%
Very Difficult	0	0%
Total	16	100%

Table 3.6. Ease of Using ShopEase in Locating Products Inside the Supermarket.

The data shows that 62.5% of respondents found ShopEase easy to use, while 31.3% rated it very easy, and only 6.2% were neutral.

Variables	Frequency	Percentage (%)
Very Accurate	7	43.8%
Accurate	9	56.3%
Neutral	0	%
Inaccurate	0	%
Very Inaccurate	0	%
Total	16	100%

Table 3.7. Accuracy of Product Location Displayed on the Map.

The results reveal that 56.3% rated the system as accurate, while 43.8% rated it as very accurate in locating products.

Variables	Frequency	Percentage (%)
Very Satisfied	12	75%
Satisfied	4	25%
Neutral	0	0%
Unsatisfied	0	0%

Very Unsatisfied	0	0%
Total	16	100%

Table 3.8. User Satisfaction with the Real-Time Pricing Feature.

The data shows that 75% of respondents were very satisfied, and 25% were satisfied with the real-time pricing feature.

Variables	Frequency	Percentage (%)
Very Responsive	12	75%
Responsive	4	25%
Neutral	0	0%
Slow	0	0%
Very Slow	0	0%
Total	16	100%

Table 3.9. System Responsiveness When Scanning Barcodes.

The results show that 75% rated the barcode scanner as very responsive, while 25% said it was responsive.

Variables	Frequency	Percentage (%)
Very Clear	8	50%
Clear	8	50%
Neutral	0	0%
Confusing	0	0%
Very Confusing	0	0%
Total	16	100%

Table 4.0. Clarity and Understandability of the User Interface.

The data shows that 50% of respondents rated the user interface as very clear, and 50% as clear.

Variables	Frequency	Percentage (%)
Strongly Agree	14	87.5%

Agree	2	12.5%
Neutral	0	0%
Disagree	0	0%
Strongly Disagree	0	0%
Total	16	100%

Table 4.1. Perception on ShopEase in Making Shopping Faster and More Convenient.

The results reveal that 87.5% of respondents strongly agree, and 12.5% agree that ShopEase makes shopping more convenient and timesaving.

Variables	Frequency	Percentage (%)
Very Likely	13	81.3%
Likely	3	18.8%
Neutral	0	0%
Unlikely	0	0%
Very Unlikely	0	0%
Total	16	100%

Table 4.2. Likelihood of Using ShopEase Again If Implemented in a Real Supermarket.

The data shows that 81.3% of respondents are very likely to use ShopEase again if implemented in a real supermarket, while 18.8% are likely to use it.

Variables	Ranking	Percentage (%)
Product Details Display	4	43.8%
Real-time Pricing	1	81.3%
Barcode Scanner	2	62.5%
Can See Total Amount	3	56.3%
Easy Navigation System	3	56.3%

Table 4.3. Most Useful Feature of ShopEase as Identified by Respondents.

The results reveal that 81.3% of respondents found the real-time pricing feature the most useful, followed by 62.5% who selected the barcode scanner, and 56.3% who appreciated both the total amount display and easy navigation system. Meanwhile, 43.8% found the product details display helpful.

Variables	Frequency	Percentage (%)
Touchscreen Display	5	%
Barcode Scanner	0	%
Locator/Map	4	%
None	7	%
Total	16	100%

Table 4.4. Device Parts Encountered with Issues.

The findings indicate that 43.8% of respondents reported having no issues with the ShopEase device, while 31.3% experienced difficulties with the touchscreen display, 25% with the locator/map, and none with the barcode scanner.

Another result indicated that most users had no further suggestions for improvement, with responses such as “None,” “N/A,” or “None at the moment” making up the majority. However, a few participants suggested enhancements like adjusting screen sizes, improving the device’s appearance, adding a self-checkout option, and integrating a live chat feature.

SUMMARY AND DISCUSSION

Based on the survey results, most respondents were aged 18–24 (47%), followed by those 45 and above (20%), showing that ShopEase appeals to both young adults and older shoppers. Female respondents (61%) made up the majority, reflecting their common role in grocery shopping. Moreover, 55% shop weekly or more often, indicating frequent encounters with shopping challenges.

Findings show that many respondents face supermarket difficulties with 61% “Sometimes” and 10% “Always” struggled to find items, while 50% sometimes left without buying due to unavailable products. The main issues were price comparison (52%), expense tracking (49%), and changing product locations (30%), emphasizing the value of ShopEase’s locator, real-time pricing, and budgeting features.

Most participants gave positive feedback, with 54% saying they were “Very Likely” and 25% saying they were “Likely” to use ShopEase. Top-rated features were real-time price display (76%), barcode scanning (70%), and virtual cart with budget tracking (65%), showing that shoppers value price awareness and expense management. For future improvements, suggestions included digital payment (28%), shopping list import (16%), and queue monitoring (16%). While some cited technical (43%) and usability (25%) issues, 57% were very likely and 31% likely to recommend ShopEase, and 62% believed it would enhance shopping efficiency and budgeting.

In conclusion, the survey confirms the need for ShopEase as a practical solution to common shopping challenges. A follow-up user testing survey was also conducted to gather feedback from users who tested the prototype, assessing its practical performance and how effectively it addressed the issues identified in the initial survey.

Prototype Evaluation Survey

The second survey involved 10 participants who personally tested the ShopEase prototype in a controlled environment. The group was balanced in terms of gender (50% male, 50% female) and age, with participants ranging from 18 to 45 years old to ensure diverse perspectives. The evaluation focused on system functionality, design, usability, and overall user experience.

Results show that most participants found the system easy to use (80%) and the interface clear and user-friendly (90%). Respondents noted that product location and price updates were the most helpful features, significantly reducing the time spent searching for items. In terms of performance, 8 out of 10 participants rated the system as “Very Effective” in assisting their shopping navigation.

However, minor issues were observed, such as occasional lag in barcode scanning and brief delays in map updates due to Wi-Fi signal strength. Despite these, participants expressed overall satisfaction, with 90% stating they would use ShopEase if implemented in supermarkets, and all respondents agreeing that it could improve the shopping experience.

In summary, the second survey supports the effectiveness of ShopEase as a user-friendly and reliable system. Feedback from prototype testing confirms that the system’s core functions—navigation, price tracking, and indoor mapping—are valuable and practical for supermarket use. These findings strengthen the conclusion that ShopEase addresses real shopping challenges and has strong potential for adoption and improvement in future version.

CONCLUSION

Conclusion

The testing and evaluation of ShopEase demonstrated its effectiveness as a supermarket navigation and product locator system designed to enhance customer convenience and support store operations. The system's indoor navigation feature successfully assisted shoppers in locating specific products and guided them accurately to their corresponding aisles. Through its dynamic aisle mapping and turn-by-turn guidance, users experienced reduced search time and improved efficiency in navigating the store layout, fulfilling the study's objective of providing a dependable navigation function within the supermarket.

Building on this, the developed touchscreen interface proved to be easy to use, interactive, and user-friendly. Customers were able to perform product searches, view information, and navigate the supermarket with ease. The responsiveness and simplicity of the touchscreen display contributed to a smooth and convenient shopping experience, satisfying the objective of creating an accessible and functional interface suitable for shoppers of varying levels of digital literacy.

Complementing the touchscreen interface, the integration of the barcode scanning feature operated effectively and allowed users to retrieve accurate product prices without delay. It also enabled the addition and modification of items in the virtual cart, supporting expense tracking and quantity adjustments. The system maintained consistent synchronization with the database, ensuring that pricing and cart information remained accurate and updated throughout the testing process.

In addition to the user-oriented functions, the administrative features of ShopEase were also found to be effective. The functionality provided to store administrators for updating product locations in a synchronized manner was successfully implemented. The administrative tools allowed for immediate modification of product placements, with all changes automatically reflected on the customer's navigation interface. This ensured that the aisle mapping and product data remained accurate, meeting the objective of maintaining dependable information updates within the system.

Furthermore, the developed online platform for store administrators served as a centralized interface for managing product information, pricing, and location details. It simplified the process of updating data and organizing inventory, allowing administrators to efficiently oversee supermarket operations. The results confirmed that this component fulfilled its purpose of improving data management and enhancing operational efficiency on the administrative side.

Lastly, the system demonstrated stable performance through Wi-Fi synchronization among the product database, administrative dashboard, and pushcart touchscreen displays. Although brief connectivity interruptions occasionally affected the update speed, ShopEase consistently maintained accurate and synchronized data once the network connection was restored. Overall, the system met all six objectives of the study, proving that ShopEase is a functional and innovative solution capable of improving the shopping experience, optimizing store operations, and advancing the integration of technology in modern supermarket environments.

Recommendations

For further improvement, it is recommended that ShopEase undergo extended testing across multiple supermarkets to validate scalability and adaptability in different store layouts. The system should also be tested under peak shopping conditions to evaluate stability in high-traffic environments.

To improve accessibility, features such as voice assistance, larger font options, and adjustable screen brightness may be added, particularly to support elderly and visually impaired customers. Optimizing Wi-Fi connectivity through stronger anchors or backup offline modes is also advised to minimize latency during network congestion.

For administrators, adding predictive analytics and reporting tools could improve decision-making by providing insights on customer behavior and popular products. Developing a mobile version of ShopEase would also let customers pre-plan shopping lists and sync them with in-store devices.

Lastly, future researchers may consider integrating cashless payments AI-driven recommendations, to further enhance the system's functionality and usefulness.

REFERENCES

Ali, S. A., et al. (2025). *Raspberry Pi in Industry 4.0: A comprehensive review of applications*. Preprints, 2025, 1–21. <https://doi.org/10.20944/preprints202501.1234.v1>

BusinessWorld. (2022, September 8). *AllDay introduces smart carts for experiential shopping*. BusinessWorld Online. <https://www.bworldonline.com/corporate/2022/09/08/474561/allday-introduces-smart-carts-for-experiential-shopping/>

Caper. (2023). *Caper smart cart*. Retrieved from <https://www.caper.ai/caper-smart-cart>

Capterra. (2025). *XAMPP reviews 2025: Verified reviews, pros & cons*. Capterra. Retrieved from <https://www.capterra.com/p/246810/XAMPP/reviews>

Coghill, D. (2022, December 24). *Filling the shopping cart with good tidings* | Inquirer Opinion. INQUIRER.net. <https://opinion.inquirer.net/159788/filling-the-shopping-cart-with-good-tidings>

De Guzman, R., & Tan, M. (2019). *Mobile-based indoor navigation system for malls*. *Journal of Philippine Information Technology*, 14(2), 45–52.

De La Salle University. (2024). *Smart shelf systems: Enhancing retail with sensor-based monitoring*. *Philippine Journal of Computing and Retail Innovation*, 19(1), 77–89.

Farrel, G. E., et al. (2023). *Scalable edge computing cluster using a set of Raspberry Pis*. Proceedings of the 2023 ACM Symposium on Edge Computing, 112–123. <https://doi.org/10.1145/3578358.3580000>

Garcia, M., & Ramos, L. (2022). *Raspberry Pi-based local server system for small retail businesses*. *Philippine Journal of Computing and Information Technology*, 17(2), 45–56.

Geo-Admin. (2024, September 13). *Industrial Solutions - Geoplan PH: Distributor & Systems Integrator* Geoplan PH: Distributor & Systems Integrator since 2011 RTLS | IoT | GIS | AV | IT | Managed Services. Geoplan PH: Distributor & Systems Integrator Since 2011. <https://geoplanph.com/>

Gupta, V. (2023, November 16). *10 Best Programming Languages for Raspberry Pi - AlmaBetter*. AlmaBetter. <https://www.almabetter.com/bytes/articles/best-programming-languages-for-raspberry-pi>

Henry, L. (2024, June 4). *5 Top Grocery Store Technology Innovations To Implement*. Itretail.com; POS Nation. <https://www.itretail.com/blog/grocery-store-technology-innovations#:~:text=Self%2DCheckout%20Systems>

Installers PH. (2024). *Price Checker and Verifier Machines*. Retrieved from <https://installersph.com>.

Intelloger (2024). Indoor Navigation System & Tracking Solution : Precision Guidance. <https://www.intelloger.com/indoor-navigation-system/>

Jaycon-Dev. (2024, November 8). 15 Innovative Raspberry Pi Use Cases: What You Can Do with Raspberry Pi. Jaycon. <https://www.jaycon.com/15-innovative-ways-to-use-raspberry-pi/>

Kaur, R., & Singh, J. (2021). Comparative study of local server deployment using Raspberry Pi versus cloud-based hosting for IoT applications. International Journal of Advanced Computer Science and Applications, 12(9), 245–252. <https://doi.org/10.14569/IJACSA.2021.0120931>

Keenan, M. (2024, September 26). Ecommerce Shopping Cart: Definition, Options, and How To Choose (2024). Shopify. <https://www.shopify.com/ph/blog/shopping-cart>

Krify. (2020). Which one to choose from XAMPP vs WAMP? Krify Blog. Retrieved from <https://krify.co/xampp-vs-wamp-which-one-to-choose-from-xampp-vs-wamp>

Kushwaha, T., & Agrawal, M. (2020). Digital retailing: How digital innovations impact customer satisfaction and operational efficiency in supermarkets. Journal of Retail Technology, 45(2), 102–115.

Lopez, J., Cruz, P., & Mendoza, R. (2021). Development of touchscreen kiosk systems for grocery stores. Philippine Journal of Computing and Innovation, 17(1), 60–69.

Makerlab Electronics. (2024). Interactive Displays for Retail Systems. Retrieved from <https://www.makerlab-electronics.com>.

Malapo, J. (2022, March 1). AllDay introduces Philippines' first supermarket 'smart carts' BusinessWorld Online. <https://www.bworldonline.com/spotlight/2022/03/01/432935/allday-introduces-philippines-first-supermarket-smart-carts/>

APPENDICES

APPENDIX A. CALENDAR OF ACTIVITIES

Gantt Chart of Activities (Methods of Research)

Legend: On-going Completed

ACTIVITY	AUGUST	SEPTEMBER	OCTOBER	NOVEMBER	DECEMBER	JANUARY	FEBRUARY	MARCH	APRIL	MAY
CORE ACTIVITIES										
Group Meetings		■	■	■	■	■	■	■	■	
Data Gathering			■	■						
Project Consultation		■	■	■	■	■	■	■	■	
Title Defense				■	■					
Chapter 1-3 Paper				■	■	■	■	■	■	
HARDWARE DEV.										
Miniature Making					■	■	■	■	■	
Component Selection				■	■	■	■			
Prototype Development				■	■	■	■	■	■	
SOFTWARE DEV.										
Design Analysis				■	■	■	■			
Wireframing					■	■	■	■	■	
Coding				■	■	■	■	■	■	
Debugging							■	■	■	

Table 4.5. Gantt Chart of Activities during Methods of Research.

Gantt Chart of Activities (Capstone Design 1 and 2)

Legend: On-going Completed

ACTIVITY	JANUARY	FEBRUARY	MARCH	APRIL	MAY	JUNE	JULY	AUGUST	SEPT	OCTOBER
PLANNING										
Group Meetings		■	■	■	■	■	■	■	■	
Data Gathering			■	■	■	■	■	■	■	
Project Consultation		■	■	■	■	■	■	■	■	
Component Purchasing		■	■	■	■	■	■	■	■	
DESIGNING										
Wireframing			■	■	■	■	■	■	■	
UI/Architecture Design			■	■	■	■	■	■	■	
3D Modeling				■	■	■	■	■	■	
DEVELOPING										
Prototyping			■	■	■	■	■	■	■	
Front End Development				■	■	■	■	■	■	
Back-End Development			■	■	■	■	■	■	■	
Integration				■	■	■	■	■	■	
TESTING										
Functionality Testing					■	■	■	■	■	
DEPLOYING										
Initial MVP Deployment					■	■	■	■	■	
REVIEWING										
Iteration						■	■	■	■	
Bug Fixing						■	■	■	■	

Table 4.6. Gantt Chart of Activities during Capstone Design 1 and 2.

APPENDIX B. ACTUAL THESIS EXPENSES

THESIS EXPENSES

Quantity	Specifics	Approximate Cost	Actual Cost
1	Raspberry Pi 5	₱ 7,000.00	₱ 6,075.00
1	IPS DSI 800×480 Pixel Screen	₱ 3,000.00	₱ 2,270.00
1	Pi5 UPS 5v Li-Battery	₱ 1,500.00	₱ 1,455.00
1	Barcode Scanner	₱ 1,000.00	₱ 848.00
1	ESP Acrylic Case	₱ 500.00	₱ 312.00
1	Metal Casing	₱ 1,000.00	₱ 981.00
5	ESP 32	₱ 1,000.00	₱ 900.00
5	Case Brackets	₱ 200.00	₱ 104.00
1 pack	Wire	₱ 200.00	₱ 141.00
1 pack	Screws	₱ 500.00	₱ 305.00
-	3D Printed Case	₱ 2,500.00	₱ 3,071.69
TOTAL THESIS EXPENSES		₱ 18,400.00	₱ 16,462.69

Table 4.7. Actual Thesis Expenses.

Prepared by:

Shanna May Alasaas

Jasper M. Bacarisa

Angelo Miguel R. Cua

Shaine Faye G. Salting

Noted by:

Ms. Sharon P. Alasaas

Mrs. Rexie M. Bacarisa

Mrs. Michelle R. Cua

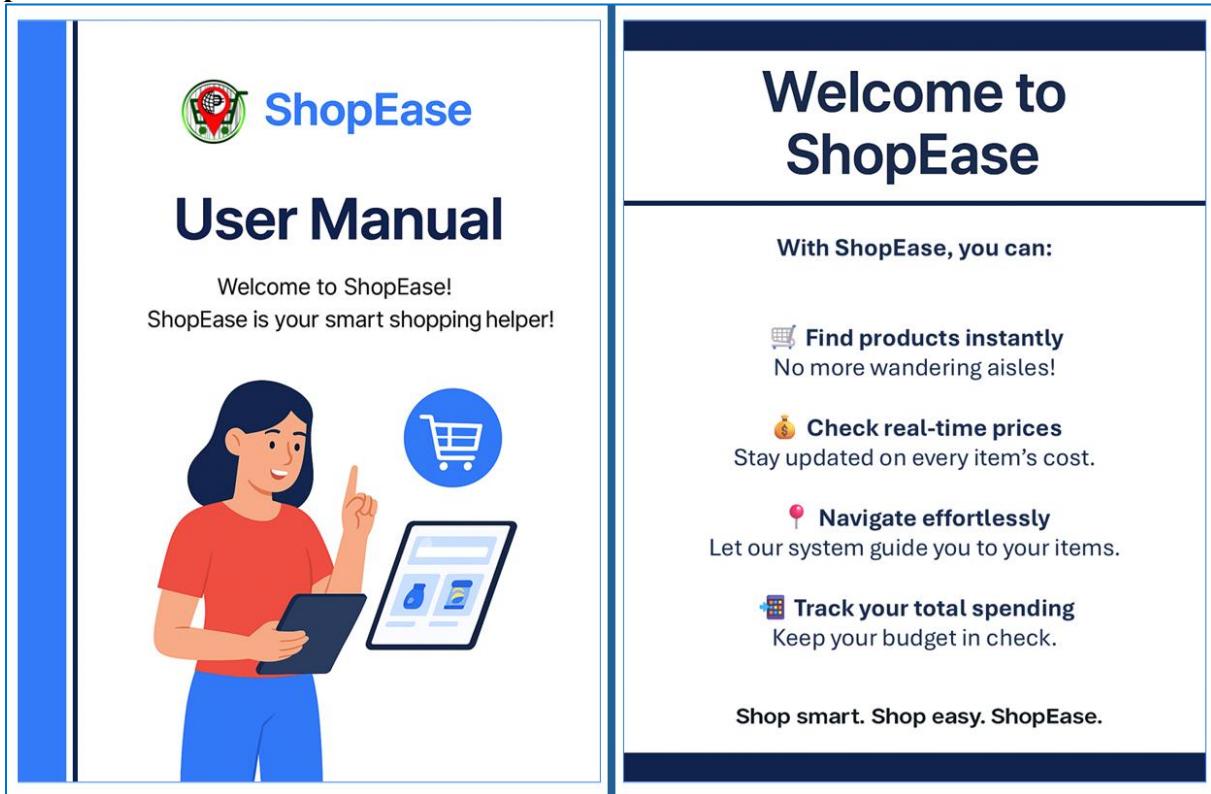
Mrs. Emily G. Salting

Approved by:

Ms. Karen Cristy A. Cifra

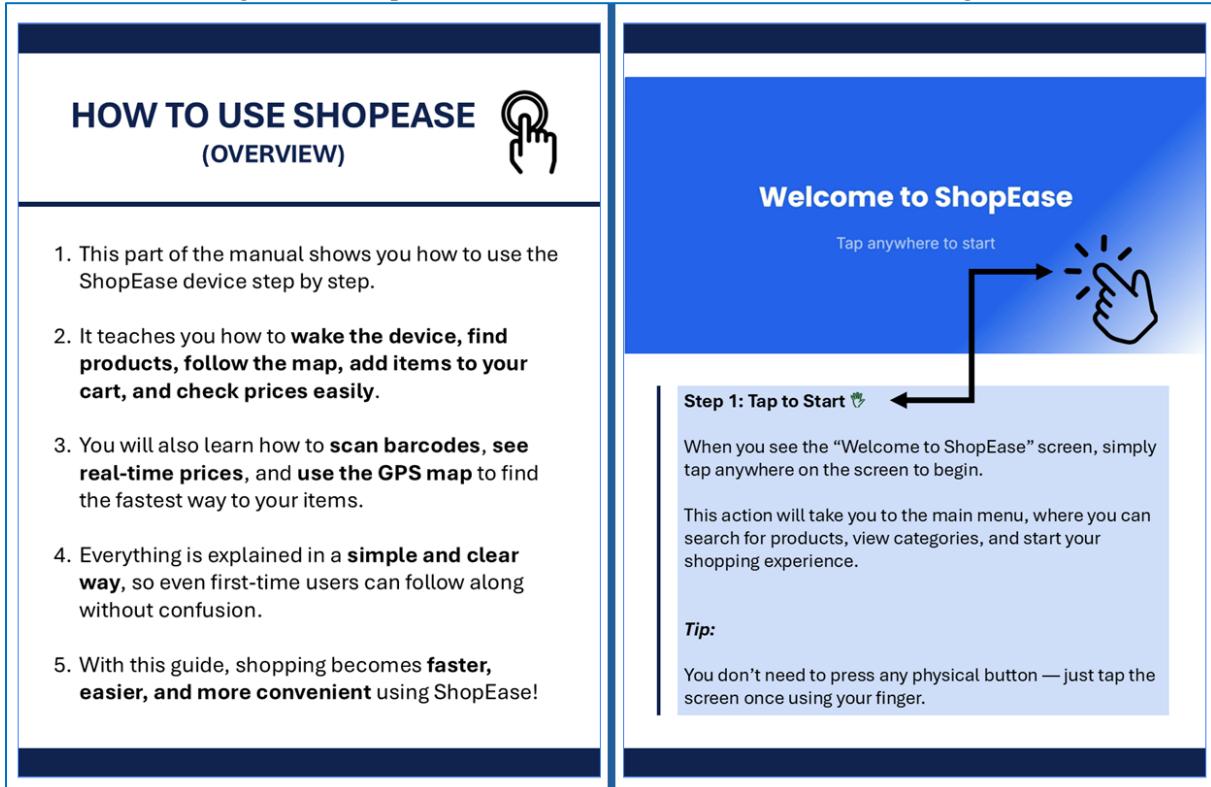
APPENDIX C. USER'S MANUAL

Appendix C. User Manual



The image shows the cover and the first page of the ShopEase User Manual. The cover features the ShopEase logo (a green circle with a white globe and a blue location pin) and the text "ShopEase User Manual". Below this, it says "Welcome to ShopEase! ShopEase is your smart shopping helper!" and features an illustration of a woman in a red shirt and blue pants holding a tablet, pointing at a blue shopping cart icon. The introduction page has a dark blue header with the text "Welcome to ShopEase". Below this, it says "With ShopEase, you can:" followed by four bullet points: "Find products instantly" (with a magnifying glass icon), "Check real-time prices" (with a price tag icon), "Navigate effortlessly" (with a map pin icon), and "Track your total spending" (with a bar chart icon). At the bottom, it says "Shop smart. Shop easy. ShopEase."

Figure 2.6. ShopEase User Manual: Cover and Introduction Pages.



The image shows two pages from the ShopEase User Manual. The left page is titled "HOW TO USE SHOPEASE (OVERVIEW)" and lists five steps: 1. Shows how to use the device step by step. 2. Teaches how to wake the device, find products, follow the map, add items to your cart, and check prices easily. 3. Explains how to scan barcodes, see real-time prices, and use the GPS map. 4. States that everything is explained in a simple and clear way. 5. Notes that shopping becomes faster, easier, and more convenient using ShopEase! The right page is titled "Welcome to ShopEase" and shows a hand tapping the screen with the text "Tap anywhere to start". A callout box titled "Step 1: Tap to Start" provides instructions: "When you see the 'Welcome to ShopEase' screen, simply tap anywhere on the screen to begin." It also includes a "Tip:" section stating "You don't need to press any physical button — just tap the screen once using your finger."

Figure 2.7. ShopEase User Manual: Interface Overview and Starting Screen.

Step 2: Home Page

After tapping the screen, you'll see the ShopEase Home Page. Here, you can start exploring and searching for the products you need.

What You'll See:

- Search Bar (Center):** Type the name of the product you're looking for, then tap Search.
- Categories (Left Side):** You can also browse items by category such as Dairy, Drinks, Snack, Chocolates, and Body Soap.
- My Cart Location (Bottom Left):** Tap this to view or locate your cart anytime.
- Cart Icon (Top Right):** Shows the number of items you've added.
- Tutorial Button (Top Right):** Need help? Tap this for a step-by-step guide on using ShopEase.

Tip:
You can always return to this home page anytime by tapping the ShopEase logo at the top.

Step 3: Categories Sidebar

On the left side of the screen, you'll see the Categories Sidebar. This section helps you browse different types of products available in the supermarket.

How to Use It:

- Tap any category name (like *Dairy, Drinks, Snack, Chocolates, or Body Soap*).
- Once tapped, the screen will show all the items under that category.
- You can scroll up or down to explore more products within the list.
- If you see something you like, just tap the item to view its price, details, and location in the store.

Tip:
The Categories Sidebar is perfect when you just want to look around or don't have a specific item in mind yet.

It's like walking through supermarket aisles — but easier and faster!

Figure 2.8. ShopEase User Manual: Home Page and Categories Sidebar Interface.

PRODUCT DETAILS

This **Quantity button** lets you edit the number of items you want to add to your cart. Simply tap the number and change it to how many you wish to buy.

A confirmation message will then appear, saying that your chosen product has been successfully added to your virtual cart.

"Locate Product" Icon

After setting the quantity, tap "Add to Cart" if you want to include it in your shopping list.

Sample visualization when using the Locate Product feature, showing a directional line and your distance from the item.

Store Map - Aisle 1

You can tap the "Locate Product" button to see a real-time store map that guides you with a directional line straight to the product's exact location.

Step 4: My Cart Location

At the bottom left corner of the screen, you'll see the My Cart Location button. When you tap this, ShopEase will show your current location inside the supermarket.

How It Works:

- The map will display your exact position in the store — for example, *near the entrance, aisle 1, aisle 2, or any section*.
- The map is based on the real layout of the supermarket, so you can easily understand where you are.
- This helps you find your way faster and know how close you are to your desired items.

Figure 2.9. ShopEase User Manual: Product Details and My Cart Location Interface.

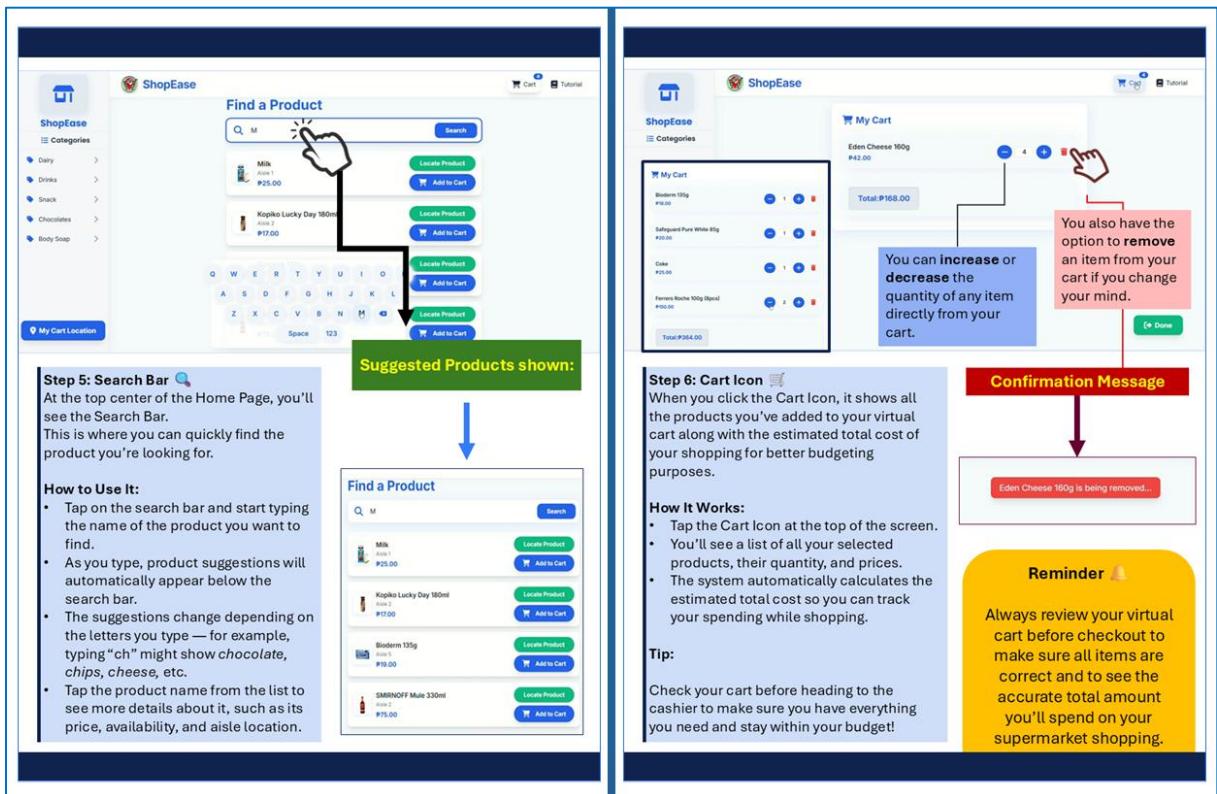


Figure 3.0. ShopEase User Manual: Search Bar and Cart Icon Interface

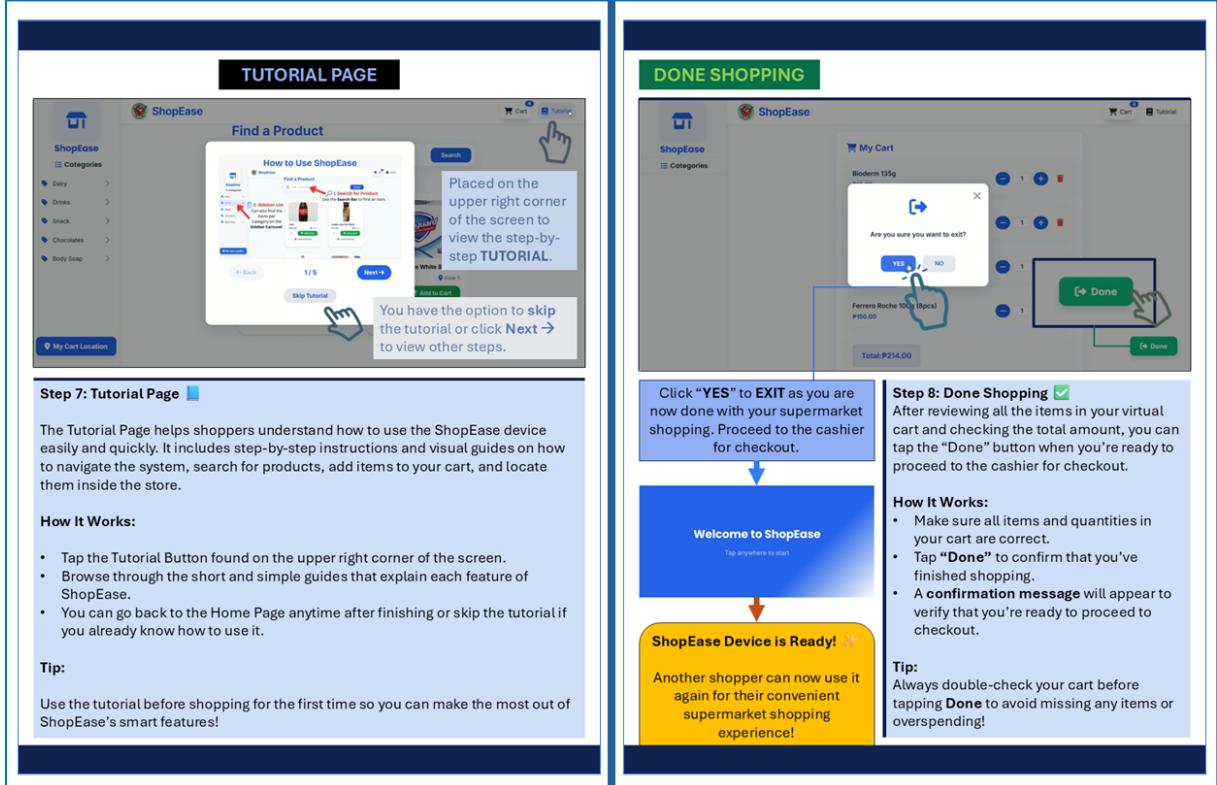


Figure 3.1. ShopEase User Manual: Tutorial and Done Shopping Pages.

The image shows two pages from the ShopEase User Manual. The left page is a 'Troubleshooting Guide' with a table mapping common problems to solutions. The right page is a 'Thank You' page with a message of appreciation.

PROBLEM	WHAT YOU CAN DO
1. Product not showing after searching	Check your spelling or try a shorter word (example: type "soap" instead of "body soap").
2. Barcode not scanning	Wipe the barcode to make sure it's clean and try again. Make sure the scanner light is on.
3. Cart not updating	Wait for a few seconds or tap the Cart icon again to refresh it.
4. Item not locating correctly	Try pressing Locate Product again or ask for assistance from a store staff.

**THANK YOU FOR USING
ShopEase!**

We hope ShopEase made your shopping faster, easier, and more enjoyable.

Shop smart. Shop easy. ShopEase.

Figure 3.2. *ShopEase User Manual: Troubleshooting Guide and Thank You Page.*

APPENDIX D. ADMIN'S MANUAL

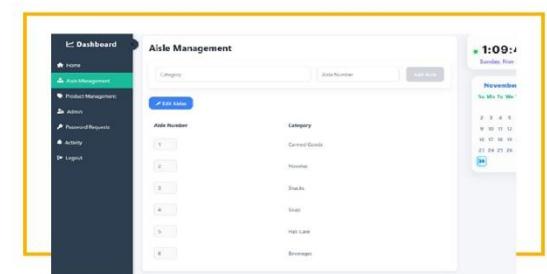
Appendix D. Admin's Manual

The image shows the cover and the first page of the ShopEase Admin's Manual. The cover features a blue and yellow chevron pattern at the top and bottom, with the MITHI Co. logo in the center. Below it is a photograph of a computer monitor displaying a 'Welcome to ShopEase' screen. The title 'ADMIN'S MANUAL' is prominently displayed in a large blue font. The introduction page follows, featuring the same chevron pattern and the ShopEase logo. It includes sections for PURPOSE, SCOPE, INTENDED USERS, REQUIREMENTS, and NOTE, along with a brief description of the manual's purpose.

Figure 3.3. ShopEase Admin's Manual: Cover and Introduction Pages.

The image shows the Login Page and the Home Dashboard Page of the ShopEase Admin's Manual. The Login Page (labeled O1) shows a green background with a white login form containing fields for Username and Password, and a 'Login' button. Below the form, the text 'STEP 1: LOG IN' is displayed. The Home Dashboard Page (labeled O2) shows a dark dashboard with various metrics and links. It includes sections for Total Products, Store Aisle, Active Admins, and Pending Actions. Below the dashboard, the text 'STEP 2: HOME DASHBOARD' is displayed. Both pages include detailed descriptions of their respective features and purposes.

Figure 3.4. ShopEase Admin's Manual: Login and Home Dashboard Pages.



STEP 3: AISLE MANAGEMENT

Aisle Management Page

The Aisle Management Page allows administrators to add, view, and update the aisles inside the supermarket. Each aisle has a number and a specific product category. This page helps maintain an organized store layout for easier product locating.

Purpose

The purpose of this page is to let admins manage all store aisles by assigning correct categories to each aisle. A well-organized aisle list helps the ShopEase system guide users accurately during in-store navigation.

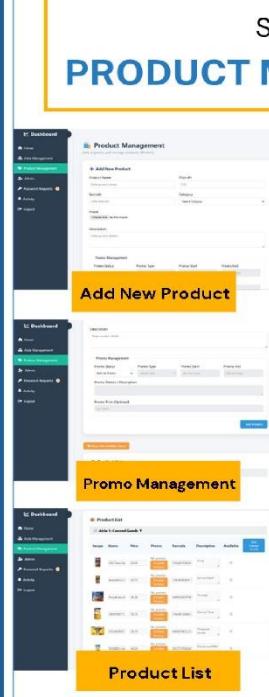
How to Add a New Aisle

- Type the category name in the Category field.
- Type the aisle number in the Aisle Number field.
- Click the Add Aisle button.
- The new aisle will appear in the list below.

How to Edit Existing Aisles

- Click the Edit Aisles button.
- Update the aisle number or category as needed.
- Save the changes.

03



STEP 4: PRODUCT MANAGEMENT

ADD NEW PRODUCT

- Product Name** – Enter the name of the item.
- Price** – Set the price.
- Barcode** – Input the product barcode.
- Category** – Choose the correct category.
- Image** – Upload a product picture.
- Description** – Add short product details.

PROMO MANAGEMENT

- Promo Status** – Select if the product is on promo or not.
- Promo Type** – Choose the promo type.
- Promo Start / End** – Set promo dates.
- Promo Details** – Give a short explanation of the promo.
- Promo Price (Optional)** – Enter the special promo price.
- Click Add Product to save the new item.

SHOW NOT AVAILABLE ITEMS

This button displays products that are out of stock or marked unavailable.

PRODUCT LIST

The Product List shows all items grouped by category. Each product displays:

- Name
- Category
- Price
- Image
- Availability
- Edit

This helps admins quickly review and update product data.

04

Figure 3.5. ShopEase Admin's Manual: Aisle and Product Management Pages.



STEP 5: PRINT PRODUCT LIST

Print Product Lists

The Print Product Lists section allows admins to generate a printed copy of product information. This feature is useful for inventory checking, manual counting, or store reporting.

PRINT AVAILABLE PRODUCTS

This button prints a list of all products that are currently available in the store. The printed list includes **Product Name**, **Category**, **Barcode**, **Availability Status**, **Price**, and **Date Available**.

Purpose: Helps admins check active items during inventory or when updating the shelves.

PRINT NOT AVAILABLE PRODUCTS

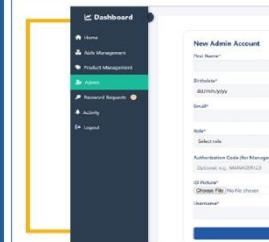
This button prints a list of all products that are marked as Not Available or Out of Stock. It shows **Product Name**, **Category**, **Last Known Price**, **Barcode**, **Availability Status**, **Price**, and **Date Available**.

Purpose: Useful for monitoring out-of-stock items, restocking needs, and tracking items that require updates.

NOTES

- Printing is optional but recommended for physical inventory checks.
- Ensure the printer is connected and working before printing.
- Printed lists can be used during audits, stock reviews, or product verification.

05



STEP 6: ADMIN PAGE

Admin Page

Purpose

The Admin Page allows the Full-Access Admin to create new administrative accounts in the ShopEase system. Only authorized Full-Access Admins can add new users. This ensures that the system remains secure and that only approved admins can log in and manage store data.

Fields and Requirements

- First Name** – Enter the admin's first name.
- Last Name** – Enter the admin's last name.
- Birthday** – Select the admin's date of birth using the calendar dropdown.
- Role** – Automatically calculated after selecting a birthday (if system-supported).
- Email** – Input a valid email address.
- Contact Number** – Enter a working mobile number.
- Role** – Select the admin's role (e.g., Full-Access Admin, Admin, Standard Admin, Limited Access).
- ID Picture** – Upload a photo of the person for identification.
- Username** – Create a unique username for login.
- Password** – Set a secure password for the new admin.

How to Create a New Admin Account

- Fill in the personal information fields (First Name, Last Name, Birthday, Age).
- Enter the Email and Contact Number.
- Choose the appropriate role from the Role dropdown.
- Upload a valid ID Picture.
- Set the Username and Password for the new admin.
- Click Create to save the new admin account.

If the required fields are complete, the new admin will be added to the system and can now log in using their assigned credentials.

06

Figure 3.6. ShopEase Admin's Manual: Product Lists and Admin Administration Pages.

STEP 7: PASSWORD REQUESTS

How to Manage Password Reset Requests

View All Requests

- When you open the page, the "All Requests" table shows every password reset request made by users.
- Each row displays the user's information, request date and time, IP address, status and available actions.

View Pending Requests Only

- Click the "Pending Requests" tab to filter only requests that still need admin action.
- Use this view to easily identify users who are still waiting for their password reset.

Check Request Details

For each request, review:

- User** – Name and email of the requesting user.
- Requested** – Date and time the request was made.
- IP Address** – The IP used when sending the request.
- Status** – "Pending" or "Handled".
- Handler** – The admin who has already processed the request (if handled).

Reset a User's Password

- Login a request with **Pending** status.
- Click the **Reset Password** button.
- The system will automatically generate and apply a new password for the user.
- Provide the new password to the user securely.

Mark a Request as Handled

- After resetting the user's password (or after sending an reset is needed), click **Mark Handled**.
- The status will change to **Handled**, and your name will appear under the handler column.
- This helps track which admin processed the request.

Return to Dashboard

Click **Back to Dashboard** at the top-left corner to exit the password request page.

STEP 8: ACTIVITY LOGS

The Activity Page displays all actions performed inside the Admin System. It records updates, changes, and tasks done by different admins. This page helps maintain transparency and ensures that all system activities can be monitored easily.

PURPOSE

- To track recent actions within the system
- To provide a record of product updates, user actions, and admin tasks
- To help identify errors, unauthorized changes, or missing updates
- To support accountability among admin users

ACTIVITY SUMMARY

The top section shows quick counters such as:

- Total Activities** – Total actions recorded in the system
- Today** – Number of actions done for the current day
- Product Changes** – Updates related to products
- User Actions** – Activities done by admin users

FILTERS

The page includes dropdowns that allow admins to filter logs by:

- Activity Type** (e.g., product update, promo change, account edit)
- User** (which admin performed the action)

ACTIVITY LIST

A list of activities is displayed, showing:

- Action Description (e.g., "Updated product: Argentina Corned Beef 175g")
- User who performed the action
- Timestamp (e.g., "1 min ago")
- Activity icon that indicates the type of action performed

GENERAL USAGE

- Monitor real-time updates in the system
- Review and verify recent changes
- See which admin performed a specific action
- Assume security and accountability
- Track product, promo, or data adjustments

LOGGING OUT

After checking/managing activities, admins can select **Logout** from the navigation menu. Once logged out, the system will return to the **Login Page** for security.

Figure 3.7. ShopEase Admin's Manual: Password Management and Activity Logs Pages.

FREQUENTLY ASKED QUESTIONS (FAQs)

- What should I do if I cannot log in to the Admin Panel?**
Check if your username and password are correct. Make sure your internet connection is stable. If the issue continues, contact the system administrator to verify your account.
- What if I forget my admin password?**
Use the password reset process in the system. If needed, another authorized admin can reset your password through the Password Requests page.
- How do I update a product's information?**
Go to the Product Management page, search for the product, and click the edit button. Update the necessary fields and save your changes.
- How can I remove a product from the system?**
Open the Product Management page, find the product, and choose the delete option. Confirm the action to remove the product.
- How do I add a new aisle?**
Go to the Aisle Management page, click "Add New Aisle," fill in the required details, and save it.
- What should I do if product stock is not updating?**
Make sure you are editing the correct product and that the changes were successfully saved. If the issue continues, report it to the system developer or support team.
- How do I know if changes were made by another admin?**
Check the Activity Logs page. It records all actions made by every admin in the system.
- Is it safe to use the Admin Panel on any device?**
Use only secure and trusted devices. Avoid public computers or shared networks to protect system data.
- What happens if two admins work at the same time?**
The system supports multiple admins, but it is recommended to communicate with your team to avoid editing the same product or data at the same time.
- Who should I contact if I encounter system errors?**
Report errors or unusual behavior to the system developer or the assigned support person immediately.

THANK YOU

THANK YOU FOR TAKING THE TIME TO READ AND USE THIS ADMIN MANUAL. YOUR DEDICATION TO MAINTAINING AND IMPROVING OUR SYSTEM IS TRULY APPRECIATED.

ADMIN'S MANUAL

Figure 3.8. ShopEase Admin's Manual: Frequently Asked Questions (FAQs) and Thank You Pages.

APPENDIX E. DEVICE ASSEMBLY GUIDE

Appendix E. Device Assembly Guide

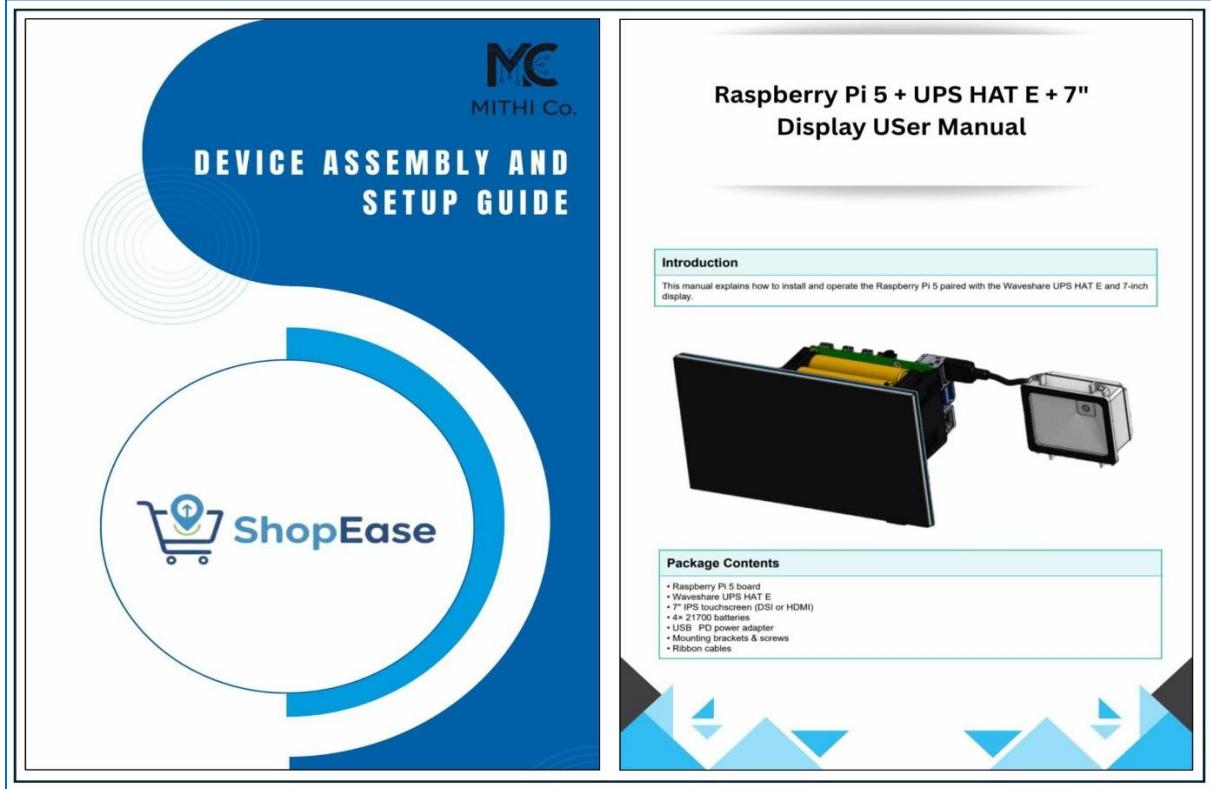


Figure 3.9. *ShopEase Device Setup Guide: Cover and Package Content Pages.*

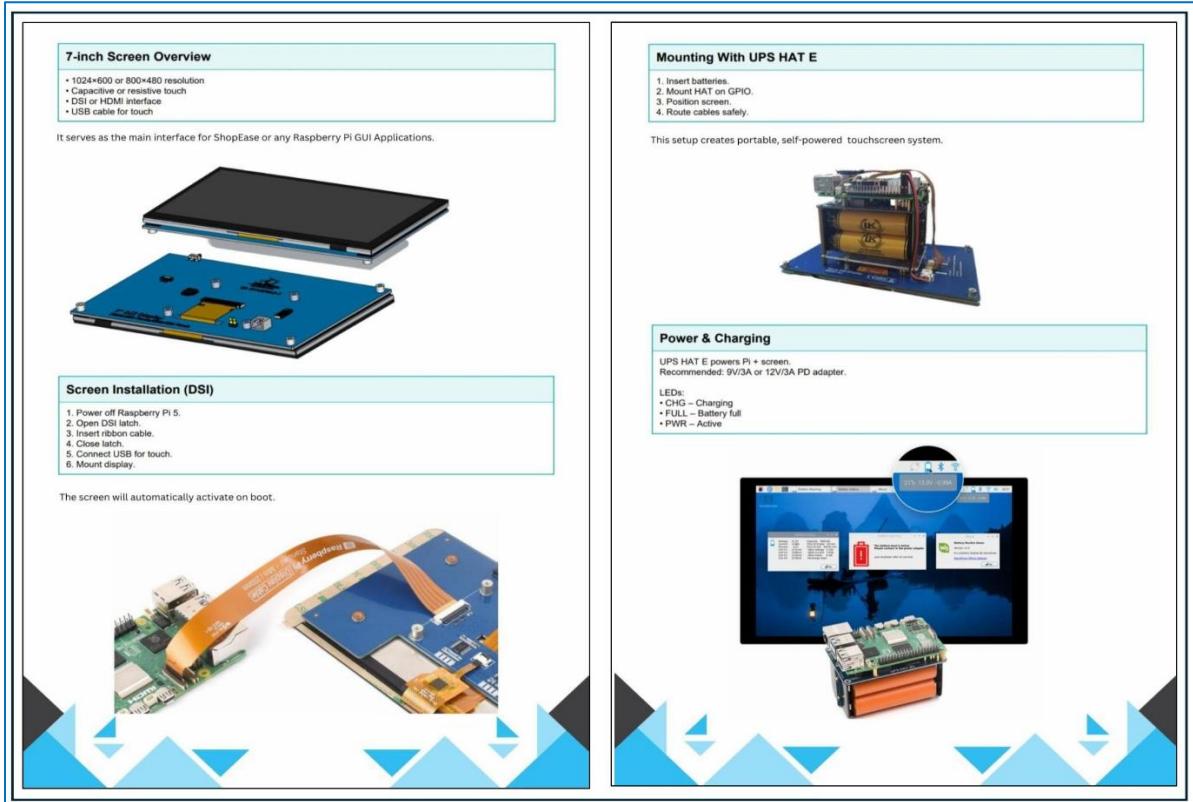


Figure 4.0. *ShopEase Device Setup Guide: Device Parts and Overview Pages.*

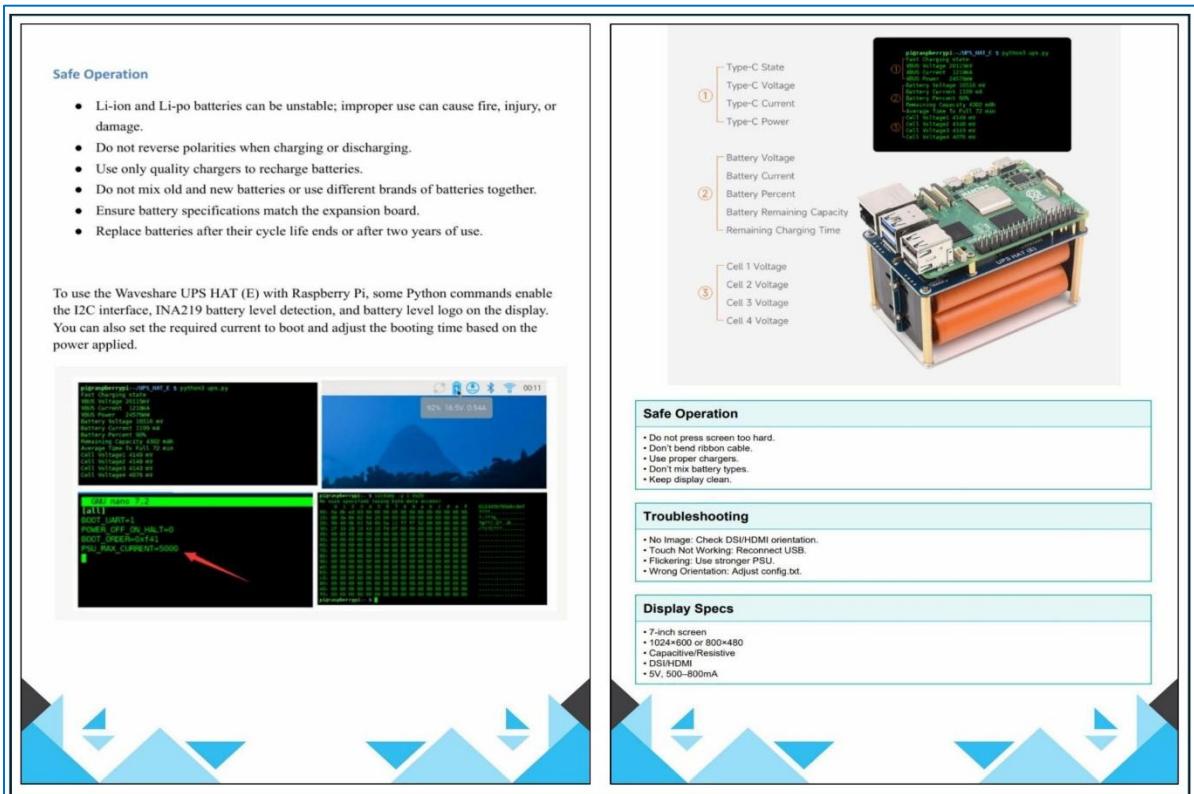


Figure 4.1. ShopEase Device Setup Guide: Sales Operation and Troubleshooting Guide Pages.

APPENDIX F. RELEVANT SOURCE CODES

Shopper's Main Interface

```
<?php session_start();
$conn = new mysqli('localhost', 'root', '', 'ecommerce_db');
if ($conn->connect_error) {die("Connection failed: " . $conn->connect_error);
}

$categories = [];
$cat_query = $conn->query("SELECT category, aisle FROM aisles ORDER BY category ASC");
if ($cat_query) {
    while ($row = $cat_query->fetch_assoc()) $categories[] = $row;
}
$items_by_category = [];
$item_query = $conn->query("SELECT category, name FROM products ORDER BY category, name ASC");
if ($item_query) {
    while ($row = $item_query->fetch_assoc()) {
        $items_by_category[$row['category']][] = $row['name'];
    }
}
usort($categories, function($a, $b) { return $a['aisle'] <=> $b['aisle']; });
?>

<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1, maximum-scale=1, user-scalable=yes">
<title>ShopEase - Supermarket Product Locator</title>
<link href="https://fonts.googleapis.com/css2?family=Inter:wght@400;600;700&family=Poppins:wght@400;600;700&display=swap" rel="stylesheet">
<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.4.2/css/all.min.css"/>
<script src="https://cdn.tailwindcss.com"></script>
<style>
body, html { height: 100vh; width: 100vw; background: #f7fafc; }
.sidebar { position: fixed; top:0; left:0; height:100vh; width:240px; background:#f5f7fa; border-right:1px solid #e5eaf1; z-index:40; display:flex; flex-direction:column; }
.maincontent { margin-left:240px; min-height:100vh; }
.sidebar-logo { display:flex; flex-direction:column; align-items:center; gap:.5em; padding:2.2rem 1.4rem 1.2rem 1.4rem; border-bottom:1px solid #e5eaf1; }
.sidebar-logo img { width:40px; height:40px; object-fit:contain; border-radius:16px; box-shadow:0 4px 16px #2563eb22; background:#e5eaf1; }
.sidebar-header { font-family:'Poppins',sans-serif; font-weight:700; font-size:1.12rem;
```

```

color:#2563eb; padding:0.75rem 1.4rem 0.5rem 2.1rem; border-bottom:1px solid #e5eaf1;}
.sidebar-list { flex:1; overflow-y:auto; padding:0.3rem 0.7rem 1.5rem 0.2rem;
background:none;}
.product-card { transition:box-shadow .15s; }
.product-card:hover { box-shadow:0 8px 32px 0 rgba(37,99,235,0.14); transform:translateY(-2px) scale(1.01);}
.start-screen { position:fixed; inset:0; z-index:50; display:flex; flex-direction:column; justify-content:center; align-items:center; background:linear-gradient(135deg, #2563eb 70%, #fff 100%);}
</style>
</head>

<body class="bg-neutral min-h-screen flex flex-col font-sans text-dark relative">
<aside id="sidebar" class="sidebar">
    <div class="sidebar-logo cursor-pointer px-3 py-2 rounded-primary flex items-center gap-2 text-dark" onclick="showTab('home')">
        <i class="fas fa-store-alt" style="font-size:2.8rem;color:#2563eb;background:#e5eaf1;border-radius:16px;padding:0.65em 0.79em;box-shadow:0 4px 16px #2563eb22;"></i>
        <span class="brand text-xl font-bold text-dark">ShopEase</span>
    </div>
    <div class="sidebar-header">Categories</div>
    <div class="sidebar-list">
        <ul id="category-list">
            <?php
            usort($categories, function($a, $b) {
                return strcasecmp($a['category'], $b['category']);
            });
            foreach ($categories as $cat):
                $cat_name = $cat['category'];
                $cat_id = 'cat_'. md5($cat_name);
                $cat_items = $items_by_category[$cat_name] ?? [];
            ?>
            <li class="category-li" tabindex="0" data-category="<?= htmlspecialchars($cat_name) ?>" data-cat-id="<?= $cat_id ?>">
                <span><?= htmlspecialchars($cat_name) ?></span>
            </li>
            <ul class="category-items-list" id="<?= $cat_id ?>" style="display:none">
                <?php foreach($cat_items as $item): ?>
                    <li tabindex="0"><?= htmlspecialchars($item) ?></li>
                <?php endforeach; ?>
            </ul>
            <?php endforeach; ?>
            <?php if(empty($categories)): ?>
                <li class="text-gray-400"><i class="fas fa-exclamation-circle"></i> No categories found.</li>
            <?php endif; ?>
        </ul>
    </div>
</aside>
<main class="flex-grow p-4">
    <?php if(!empty($products)): ?>
        <div class="grid grid-cols-2 gap-4">
            <?php foreach($products as $product): ?>
                <div class="product-card">
                    <img alt="{$product['image']}" class="w-full h-40 object-cover" />
                    <div class="p-2">
                        <strong>{$product['name']}</strong>
                        <p>{$product['description']}</p>
                        <div class="flex justify-between w-full">
                            <span>{$product['price']}</span>
                            <span>{$product['rating']}</span>
                        </div>
                    </div>
                </div>
            <?php endforeach; ?>
        </div>
    <?php else: ?>
        <div class="text-center p-8">
            <img alt="No products found" class="w-40" />
            <p>No products found. Please refine your search or try a different category.</p>
        </div>
    <?php endif; ?>
</main>
<script>
    // Sidebar logic
    const sidebar = document.querySelector('.sidebar');
    const sidebarLogo = sidebar.querySelector('.sidebar-logo');
    const sidebarHeader = sidebar.querySelector('.sidebar-header');
    const sidebarList = sidebar.querySelector('.sidebar-list');

    sidebarLogo.addEventListener('click', () => {
        sidebar.classList.toggle('open');
        sidebarHeader.classList.toggle('open');
        sidebarList.classList.toggle('open');
    });

    sidebarHeader.addEventListener('click', () => {
        sidebar.classList.toggle('open');
        sidebarHeader.classList.toggle('open');
        sidebarList.classList.toggle('open');
    });

    sidebarList.addEventListener('click', () => {
        sidebar.classList.toggle('open');
        sidebarHeader.classList.toggle('open');
        sidebarList.classList.toggle('open');
    });

    // Product card logic
    const productCards = document.querySelectorAll('.product-card');
    const productCard = document.querySelector('.product-card');
    const productImage = productCard.querySelector('img');
    const productName = productCard.querySelector('strong');
    const productDescription = productCard.querySelector('p');
    const productPrice = productCard.querySelector('span');
    const productRating = productCard.querySelector('span');

    productImage.addEventListener('click', () => {
        productCard.classList.add('is-expanded');
    });

    productImage.addEventListener('mouseleave', () => {
        productCard.classList.remove('is-expanded');
    });

    productCard.addEventListener('click', () => {
        productCard.classList.add('is-expanded');
    });

    productCard.addEventListener('mouseleave', () => {
        productCard.classList.remove('is-expanded');
    });

    // Category logic
    const categoryList = document.querySelector('#category-list');
    const categoryLi = categoryList.querySelector('li');
    const categorySpan = categoryLi.querySelector('span');
    const categoryUl = categoryList.querySelector('ul');
    const categoryItemsList = categoryList.querySelector('ul');
    const categoryItemLi = categoryItemsList.querySelector('li');
    const categoryItemSpan = categoryItemLi.querySelector('span');

    categorySpan.addEventListener('click', () => {
        categoryList.classList.add('is-expanded');
    });

    categorySpan.addEventListener('mouseleave', () => {
        categoryList.classList.remove('is-expanded');
    });

    categoryItemSpan.addEventListener('click', () => {
        categoryList.classList.add('is-expanded');
    });

    categoryItemSpan.addEventListener('mouseleave', () => {
        categoryList.classList.remove('is-expanded');
    });

    // Main content logic
    const mainContent = document.querySelector('main');
    const mainGrid = mainContent.querySelector('.grid');
    const mainGridItem = mainGrid.querySelector('div');
    const mainGridImage = mainGridItem.querySelector('img');
    const mainGridName = mainGridItem.querySelector('strong');
    const mainGridDescription = mainGridItem.querySelector('p');
    const mainGridPrice = mainGridItem.querySelector('span');
    const mainGridRating = mainGridItem.querySelector('span');

    mainGridImage.addEventListener('click', () => {
        mainGridItem.classList.add('is-expanded');
    });

    mainGridImage.addEventListener('mouseleave', () => {
        mainGridItem.classList.remove('is-expanded');
    });

    mainGridName.addEventListener('click', () => {
        mainGridItem.classList.add('is-expanded');
    });

    mainGridName.addEventListener('mouseleave', () => {
        mainGridItem.classList.remove('is-expanded');
    });

    mainGridDescription.addEventListener('click', () => {
        mainGridItem.classList.add('is-expanded');
    });

    mainGridDescription.addEventListener('mouseleave', () => {
        mainGridItem.classList.remove('is-expanded');
    });

    mainGridPrice.addEventListener('click', () => {
        mainGridItem.classList.add('is-expanded');
    });

    mainGridPrice.addEventListener('mouseleave', () => {
        mainGridItem.classList.remove('is-expanded');
    });

    mainGridRating.addEventListener('click', () => {
        mainGridItem.classList.add('is-expanded');
    });

    mainGridRating.addEventListener('mouseleave', () => {
        mainGridItem.classList.remove('is-expanded');
    });

    // Footer logic
    const footer = document.querySelector('footer');
    const footerText = footer.querySelector('p');

    footerText.addEventListener('click', () => {
        window.location.href = 'https://www.shop-ease.com';
    });
</script>

```

```

        </ul>

        <button id="showCartLocationBtn" onclick="showCartLocation()" class="fixed bottom-8 left-2 px-6 py-3 bg-blue-600 text-white font-bold rounded-xl shadow-lg hidden">
            <i class="fas fa-map-marker-alt text-xl"></i> My Cart Location
        </button>
    </div>
</aside>
<div id="maincontent" class="maincontent flex-grow flex flex-col">
    <div id="startScreen" class="start-screen" onclick="startShopEase()">
        <h1 class="font-poppins font-extrabold">Welcome to ShopEase</h1>
        <p class="font-sans">Tap anywhere to start</p>
    </div>

    <nav class="sticky top-0 w-full z-30 bg-white shadow-nav flex items-center px-4 py-2">
        <div class="flex items-center gap-3 flex-1">
            
            <span class="font-extrabold text-2xl md:text-3xl text-accent tracking-tight font-poppins">ShopEase</span>
        </div>

        <div class="flex gap-1 items-center">
            <button onclick="showTab('cart')" id="nav-cart" class="relative px-3 py-2 rounded-primary flex items-center gap-2 text-dark">
                <i class="fas fa-shopping-cart text-xl"></i>
                <span class="hidden sm:inline font-medium">Cart</span>
                <span id="cart-badge" class="absolute -top-2 -right-2 bg-accent text-white rounded-primary px-2 text-xs font-bold shadow border-2 border-white transition-all">0</span>
            </button>

            <button onclick="openTutorial()" id="nav-tutorial" class="px-3 py-2 rounded-primary flex items-center gap-2 text-dark">
                <i class="fas fa-book text-xl"></i>
                <span class="hidden sm:inline font-medium">Tutorial</span>
            </button>
        </div>
    </nav>

    <div id="adding-msg" style="display:none; position:fixed; top:300px; left:50%; transform:translateX(-50%); background:#2563eb; color:#fff; padding:14px 30px; border-radius:12px; font-weight:600;">
        Item Added to Cart...
    </div>
    <div id="home" class="tab-content w-full max-w-2xl mx-auto">
        <h1 class="text-3xl md:text-4xl font-extrabold text-accent mb-4 font-poppins">Find a Product</h1>

```

```

<div class="flex flex-col gap-6">
  <div class="mx-auto w-full">
    <div class="relative flex flex-col sm:flex-row items-center justify-center gap-3 sm:gap-0">
      <i class="fas fa-search absolute left-5 top-1/2 -translate-y-1/2 text-accent text-2xl"></i>
      <input type="text" id="search" placeholder="Enter product name..." class="w-full py-4 pl-16 pr-28 rounded-primary bg-white shadow-card border border-subtle text-lg font-medium" autocomplete="off" />
      <button onclick="closeKeyboard()" class="absolute right-3 top-1/2 -translate-y-1/2 px-7 py-2 bg-accent text-white rounded-primary">Search</button>
    </div>
  </div>
  <div id="search-results-container" style="display:none;"></div>
  <div id="category-products" class="mt-6 grid grid-cols-1 sm:grid-cols-2 gap-x-8 gap-y-10"></div>
</div>
<section id="cart" class="tab-content w-full max-w-2xl mx-auto hidden">
  <div class="cart bg-white rounded-primary shadow-card p-7 mt-5">
    <h2 class="text-2xl font-bold text-accent mb-6 flex items-center gap-2 font-poppins">
      <i class="fas fa-shopping-cart text-2xl"></i> My Cart
    </h2>
    <ul class="cart-items flex flex-col gap-6 mb-6" id="cart-items"></ul>
    <li class="cart-item flex items-center justify-between bg-neutral rounded-primary px-4 py-4 shadow-card transition mb-2">
      <div class="total flex justify-between items-center bg-subtle rounded-secondary py-4 px-6 font-bold text-accent text-xl shadow-inner border border-neutral-200">
        <span>Total:</span> <span id="cart-total">₱0.00</span>
      </div>
      <button id="exitBtn" class="exit-btn fixed bottom-8 right-5 px-7 py-3 bg-accentgreen text-white font-bold rounded-primary">
        <i class="fas fa-sign-out-alt text-xl"></i>Done
      </button>
    <div id="exitPopup" class="fixed inset-0 flex items-center justify-center bg-black/40 z-50 hidden">
      <div class="bg-white rounded-2xl shadow-modal p-10 w-[92vw] max-w-sm modal-fade flex flex-col items-center gap-7 relative">
        <button class="absolute top-4 right-5 text-gray-400 hover:text-accent text-3xl font-bold" onclick="closeTab('exitPopup')">
          <i class="fas fa-times"></i>
        </button>
        <i class="fas fa-sign-out-alt text-5xl text-accent mb-2"></i>
      </div>
    </div>
  </div>
</section>

```

```
<p class="font-semibold text-lg text-dark mb-4 text-center">Are you sure you  
want to exit?</p>
```

```
<div class="flex gap-5">  
    <button id="yesExit" class="px-8 py-2.5 bg-accent text-white rounded-  
primary font-semibold shadow hover:bg-accent/90">YES</button>  
    <button id="noExit" class="px-8 py-2.5 bg-subtle text-accent rounded-  
primary font-semibold shadow">NO</button>  
</div>  
</div>  
</li>  
</div>  
</section>  
</div>  
</body>  
</html>
```

Category Product Listing API

```
<?php

$conn = new mysqli('localhost', 'root', '', 'ecommerce_db');
if ($conn->connect_error) die("Connection failed");
$category = isset($_GET['category']) ? $conn->real_escape_string($_GET['category']) : "";
if (!$category) exit;

$res = $conn->query("SELECT p.*, a.aisle FROM products p LEFT JOIN aisles a ON
p.category = a.category WHERE p.category='$category' ORDER BY p.name ASC");

if ($res && $res->num_rows > 0) {
    while ($prod = $res->fetch_assoc()) {

        $outOfStock = (isset($prod['stock']) && $prod['stock'] <= 0);
        $usePromo = (isset($prod['promo_status']) && $prod['promo_status'] === 'on' &&
!empty($prod['promo_price']) && is_numeric($prod['promo_price']));
        $promoPrice = $usePromo ? $prod['promo_price'] : null;
        $useDiscount = !$usePromo && isset($prod['discount_price']) &&
$prod['discount_price'] > 0 && $prod['discount_price'] < $prod['price'];
        $discountPrice = $useDiscount ? $prod['discount_price'] : null;

        $aisle = !empty($prod['aisle']) ? htmlspecialchars($prod['aisle']) : "";

        $badge = "";
        if ($outOfStock) {
            $badge = '<span class="absolute top-5 left-5 bg-red-500 text-white text-xs font-
semibold rounded-full px-4 py-1 shadow z-20">Out of Stock</span>';
        } elseif ($usePromo) {
            $badge = '<span class="absolute top-5 left-5 bg-red-500 text-white text-xs font-
semibold rounded-full px-4 py-1 shadow z-20">Promo!</span>';
        } elseif ($useDiscount) {
            $badge = '<span class="absolute top-5 left-5 bg-yellow-500 text-white text-xs font-
semibold rounded-full px-4 py-1 shadow z-20">Discount!</span>';
        }

        if ($usePromo) {
            $priceHtml = '<span class="text-green-600 font-bold text-lg mr-1">₱' .
number_format($promoPrice, 2) . '</span><span class="line-through text-gray-400 text-base font-semibold">₱' . number_format($prod['price'], 2) . '</span>';
            $cartPrice = $promoPrice;
        } elseif ($useDiscount) {
            $priceHtml = '<span class="text-green-600 font-bold text-lg mr-1">₱' .
number_format($discountPrice, 2) . '</span><span class="line-through text-gray-400 text-base font-semibold">₱' . number_format($prod['price'], 2) . '</span>';
            $cartPrice = $discountPrice;
        } else {
    
```

```

$priceHtml = '<span class="text-green-600 font-bold text-lg">₱' .  

number_format($prod['price'],2) . '</span>';
$cartPrice = $prod['price'];
}  

?>
<div class="shop-card bg-white rounded-[22px] shadow-md border border-[#f2f4f8] flex flex-col items-stretch min-h-[390px] max-w-xs mx-auto p-0 relative transition hover:shadow-lg animate-fadeIn" style="margin-bottom: 0.75rem;">  

<?= $badge ?>  

<div class="w-full flex items-center justify-center pt-5 px-5">  

<?php if (!empty($prod['image'])): ?>  

" class="w-full aspect-[1.1/1] object-cover rounded-[18px]  
border border-[#f3f4f7] bg-white max-h-40" />  

<?php else: ?>  

<div class="w-full aspect-[1.1/1] flex items-center justify-center rounded-[18px]  
border border-[#f3f4f7] bg-[#f8fafc] text-gray-400 font-semibold text-base max-h-40">No  
Image</div>  

<?php endif; ?>  

</div>  

<div class="flex-1 flex flex-col px-6 pb-5 pt-3">  

<div class="font-bold text-base text-gray-900 mb-1"><?=  
htmlspecialchars($prod['name']) ?></div>  

<div class="flex items-center justify-between mb-2">  

<?= $priceHtml ?>  

<span class="flex items-center gap-1 text-[#6b7280] text-sm font-medium ml-2">  

<i class="fas fa-map-marker-alt text-base text-accent"></i> Aisle <?= $aisle ?>  

</span>  

</div>  

<div class="product-actions mt-auto">  

<div class="action-row">  

<div class="qty-control">  

<button type="button" class="qty-btn minus-btn"><i class="fas fa-minus"></i></button>  

<input type="text" class="qty-num" value="1" readonly>  

<button type="button" class="qty-btn plus-btn"><i class="fas fa-plus"></i></button>  

</div>  

<button class="addtocart-btn px-3 py-2 bg-green-600 text-white rounded-[14px]  
font-semibold shadow hover:bg-green-700 active:scale-95 transition-all"  

onclick="addProductToCart(this,<?=  
htmlspecialchars(addslashes($prod['name'])) ?>,<?= $cartPrice ?>,<?= $outOfStock ? 0 : 1

```

```

?>)"
    <?= $outOfStock ? 'disabled' : " ?>>
        <i class="fas fa-cart-plus"></i> Add
    </button>
</div>

<button class="locate-btn w-full px-0 py-2 border border-[#e5e7eb] text-[#374151] rounded-[13px] font-semibold flex items-center justify-center gap-2 bg-white hover:bg-[#f8fafc] transition"
    onclick="showProductLocation('Aisle <?= $aisle ?>')"
        <i class="fas fa-map-marker-alt"></i> Locate Product
    </button>
</div>
</div>
</div>
<?php
    }
} else {
    echo '<div class="text-center text-gray-400 py-10">No products found in this category.</div>';
}
?>

```

Product Search & Barcode Lookup API

```
<?php
$conn = new mysqli('localhost', 'root', '', 'ecommerce_db');
$search = $_GET['search'] ?? "";

if ($search) {
    $stmt = $conn->prepare("SELECT * FROM products WHERE (barcode = ? OR name LIKE ?) AND available = 1");
    $searchLike = "%" . $search . "%";
    $stmt->bind_param("ss", $search, $searchLike);
    $stmt->execute();
    $result = $stmt->get_result();

    if ($result->num_rows > 0) {

        while ($row = $result->fetch_assoc()) {

            $usePromo = (isset($row['promo_status']) && $row['promo_status'] === 'on' &&
!empty($row['promo_price']) && is_numeric($row['promo_price']));
            $promoPrice = $usePromo ? $row['promo_price'] : null;

            $useDiscount = !$usePromo && isset($row['discount_price']) &&
                $row['discount_price'] > 0 &&
                $row['discount_price'] < $row['price'];
            $discountPrice = $useDiscount ? $row['discount_price'] : null;

            $outOfStock = (isset($row['stock']) && $row['stock'] <= 0);

            $badge = "";
            if ($usePromo) {
                $badge = '<div class="mvp-promo-badge"><span class="mvp-promo-star">' . $row['barcode'] . '</span><span class="mvp-promo-text">PROMO</span></div>';
            } elseif ($useDiscount) {
                $badge = '<div class="mvp-promo-badge mvp-discount-badge"><span class="mvp-promo-star">' . $row['barcode'] . '</span><span class="mvp-promo-text">DISCOUNT</span></div>';
            } elseif ($outOfStock) {
                $badge = '<div class="mvp-promo-badge mvp-oos-badge"><span class="mvp-promo-star">' . $row['barcode'] . '</span><span class="mvp-promo-text">OUT OF STOCK</span></div>';
            }

            if ($usePromo) {
                $priceHtml = '<span class="text-green-600 font-bold text-lg mr-1">' . $row['name'] . '</span><span class="line-through text-gray-400 text-base font-semibold ml-1">' . $row['price'] . '</span>';
            }
        }
    }
}
```

```

    $cartPrice = $promoPrice;
} elseif ($useDiscount)
    $priceHtml = '<span class="text-green-600 font-bold text-lg mr-1">₱' .
number_format($discountPrice,2) . '</span><span class="line-through text-gray-400 text-base font-semibold ml-1">₱' . number_format($row['price'],2) . '</span>';
    $cartPrice = $discountPrice;
} else {
    $priceHtml = '<span class="text-[#2563eb] font-bold text-lg">₱' .
number_format($row['price'],2) . '</span>';
    $cartPrice = $row['price'];
}

$aisle_number = (isset($row['aisle']) && trim($row['aisle']) !== "") ? $row['aisle'] : null;

if ($aisle_number) {
    $aisle = "Aisle " . htmlspecialchars($aisle_number, ENT_QUOTES);

    $viewLocationButton =
        '<button class="view-location-btn inline-block font-bold text-sm px-5 py-2.5 mt-2 rounded-lg bg-[#2563eb] text-white shadow hover:bg-[#397cf7] transition-all duration-200 focus:outline-none"
            onclick="showProductLocation(\"' . addslashes($aisle) . '\")"
            <i class="fas fa-map-marker-alt mr-2"></i> Locate
        </button>';
} else {
    $aisle = "<span class='aisle-info' style='background:#fcc;'>No aisle set</span>";
    $viewLocationButton = "";
}

$imgSrc = !empty($row['image']) ? htmlspecialchars($row['image'], ENT_QUOTES) : 'ASSETS/placeholder.png';

echo "<div class='search-result-card-wrapper' style='position:relative; margin-top:40px; margin-bottom:15px;'> .
    $badge .
    "<div class='product-card flex flex-col sm:flex-row items-center justify-between bg-white rounded-xl shadow-lg px-4 py-3 gap-4 sm:gap-0' style='min-width:350px;'>

        <div class='flex flex-row items-center gap-3 w-full sm:w-auto'>
            <img src='$imgSrc' class='product-img' id='prodimg-' . $row['id'] . '' alt=''" .
htmlspecialchars($row['name'], ENT_QUOTES) . '' style='width:54px; height:54px; object-fit:cover; border-radius:10px; box-shadow:0 2px 8px #2563eb22;' loading='lazy'>

            <div>
                <div class='font-semibold text-lg text-gray-800'>" .
htmlspecialchars($row['name'], ENT_QUOTES) . "</div>
                <div class='text-sm text-gray-500'>" . $aisle . "</div>

```

```

        <div class='font-bold mt-1' style='font-size:1.08em;'>" . $priceHtml .
    "</div>
        $viewLocationButton
    </div>
</div>

<div class='flex flex-col gap-2 min-w-[140px] w-full sm:w-auto'>
    <div class='qty-control flex items-center justify-between bg-gray-100 rounded-lg p-1 mb-1'>
        <button type='button' class='qty-btn minus-btn px-2 py-1 bg-[#2563eb] rounded-md shadow-sm hover:bg-[#397cf7] active:bg-[#1d4ed8]' data-action='decrease'>
            <i class='fas fa-minus text-white'></i>
        </button>

        <input type='text' value='1' class='qty-num w-12 text-center font-bold bg-transparent border-none focus:outline-none' readonly>

        <button type='button' class='qty-btn plus-btn px-2 py-1 bg-[#2563eb] rounded-md shadow-sm hover:bg-[#397cf7] active:bg-[#1d4ed8]' data-action='increase'>
            <i class='fas fa-plus text-white'></i>
        </button>
    </div>

    <button type='button'
        class='add-to-cart-btn fixed-size-btn flex items-center justify-center gap-1 font-bold text-base rounded-full bg-[#10b981] text-white shadow-lg transition-all duration-200 hover:bg-[#13cc91] hover:shadow-xl active:scale-95 active:shadow focus:outline-none'>
        onclick=\"$addProductToCart(this, \" . htmlspecialchars($row['name'], ENT_QUOTES) . \" , $cartPrice, true)\"
            \" . ($outOfStock ? 'disabled' : '') . \">
        <span class='icon-wrap flex items-center justify-center'>
            <i class='fas fa-shopping-cart text-lg'></i>
            <i class='fas fa-check text-lg absolute left-0 opacity-0'></i>
        </span>
        <span>ADD</span>
    </button>
</div>

</div>
</div>";
    }
} else {
    echo "<div class='error-message text-center text-red-500 font-semibold mt-4'>Item not found</div>";
}
?>

```

Admin Dashboard & Management Panel

```
/* === ACCESS & PERMISSION SYSTEM (role/request module access) === */
if (
    $_SERVER['REQUEST_METHOD'] === 'POST' &&
    isset($_POST['action']) &&
    $_POST['action'] === 'request_module_access'
) {
    if ($current_role === 'Inventory Staff') {
        $module = $_POST['module'] ?? '';
        $staff_id = $_SESSION['admin_id'];

        $check_stmt = $conn->prepare(
            "SELECT id FROM access_requests
             WHERE staff_id = ? AND requested_module = ? AND status = 'pending'"
        );
        $check_stmt->bind_param("is", $staff_id, $module);
        $check_stmt->execute();
        $check_stmt->store_result();

        if ($check_stmt->num_rows == 0) {
            $stmt = $conn->prepare(
                "INSERT INTO access_requests (staff_id, requested_module)
                 VALUES (?, ?)"
            );
            $stmt->bind_param("is", $staff_id, $module);
            $stmt->execute();
            $stmt->close();
        }

        $check_stmt->close();
        echo json_encode(['success' => true]);
    }
    exit;
}

if (
    $_SERVER['REQUEST_METHOD'] === 'POST' &&
    isset($_POST['action']) &&
    $_POST['action'] === 'handle_access_request'
) {
    if ($current_role === 'Manager' || $is_main_admin) {
        $request_id = intval($_POST['request_id']);
        $decision = $_POST['decision'] === 'approve' ? 'approved' : 'denied';
        $handler_id = $_SESSION['admin_id'];

        $stmt = $conn->prepare("
```

```

        UPDATE access_requests
        SET status = ?, handled_by = ?, handled_at = NOW()
        WHERE id = ? AND status = 'pending'
    ");
    $stmt->bind_param("sii", $decision, $handler_id, $request_id);
    $stmt->execute();
    $stmt->close();

    echo json_encode(['success' => true]);
}
exit;
}

/* === PHYSICAL AISLE & CATEGORY MAPPING === */
if ($_SERVER['REQUEST_METHOD'] === 'POST' && isset($_POST['add_aisle'])) {
    $category = ucfirst(trim($_POST['category']));
    $aisle   = ucfirst(trim($_POST['aisle']));

    $check = $conn->query("
        SELECT * FROM aisles
        WHERE category = '$category' OR aisle = '$aisle'
    ");

    if ($check->num_rows == 0) {
        $conn->query("
            INSERT INTO aisles (category, aisle)
            VALUES ('$category', '$aisle')
        ");
        log_activity($conn, "Added Aisle Number $aisle with category $category");
    }
}

header("Location: dashboard.php");
exit();
}

/* === ADVANCED PRODUCT PROMO/PRICING LOGIC === */
if (isset($_POST['update_product'])) {
    $promo_status = $_POST['promo_status'];
    $promo_type  = $_POST['promo_type'] ?? '';
    $promo_details = $_POST['promo_details'] ?? '';
    $promo_price  = !empty($_POST['promo_price']) ? floatval($_POST['promo_price']) : null;

    // Save these values alongside standard product fields
    // (SQL update logic here)
}

```

```

/* === GRANULAR ACTIVITY LOGGING === */
log_activity($conn, "Updated product: $name (ID: $id)");

Admin Login & Account Recovery

if (
    $_SERVER["REQUEST_METHOD"] == "POST" &&
    isset($_POST['action']) &&
    $_POST['action'] === 'forgot_password'
) {
    $stmt = $conn->prepare(
        "SELECT id, username, email, contact
         FROM admins
        WHERE username = ? OR email = ?
        LIMIT 1
    ");
    $stmt->bind_param("ss", $identifier, $identifier);
    $stmt->execute();
    $user = $stmt->get_result()->fetch_assoc();

    $stmt = $conn->prepare(
        "INSERT INTO password_reset_requests (username, email, requested_at, requester_ip)
         VALUES (?, ?, ?, ?)
    ");
    $stmt->bind_param("ssss", $u_username, $u_email, $requested_at, $ip);
    $stmt->execute();
    $stmt->close();

    // Notify all Main Admins
    $admins_res = $conn->query(
        "SELECT email, contact FROM admins
         WHERE role = 'Main Admin'
    ");
}

while ($admin = $admins_res->fetch_assoc()) {
    @mail($admin['email'], $subject, $message_body, $headers);

    if (!empty($admin['contact'])) {
        $admin_sms = "ShopEase: Password reset requested for user {$u_username}.";
        file_put_contents(
            'sms_log.txt',
            date('Y-m-d H:i:s') .
            " - Admin notification SMS to: {$admin['contact']}, Message: $admin_sms\n",
            FILE_APPEND
        );
    }
}

```

```

echo json_encode(['success' => true]);
exit;
}

```

Admin Logout Handler

```

session_start();
session_destroy();
header("Location: login.php");
exit();

```

Product Promo Status Update API

```

session_start();
if (!isset($_SESSION['admin_id'])) exit;

$id = isset($_POST['id']) ? intval($_POST['id']) : 0;
$status = $_POST['promo_status'] ?? "";

if ($status === 'off') {
    $stmt = $conn->prepare("
        UPDATE products
        SET promo_status='off', promo_type=NULL, promo_start=NULL,
            promo_end=NULL, promo_details=NULL, promo_price=NULL
        WHERE id=?"
    );
    $stmt->bind_param("i", $id);
    $stmt->execute();
    $stmt->close();
}

}elseif ($status === 'on') {
// Demo-only logic
$stmt = $conn->prepare("
    UPDATE products
    SET promo_status='on', promo_type='discount_percent',
        promo_details='Default 10% off', promo_price=price*0.9
    WHERE id=?"
);
$stmt->bind_param("i", $id);
$stmt->execute();
$stmt->close();
}

```

Activity Log System Tied to Admin Actions (Granular and Exportable)

```
if ($current_user_role === 'Inventory Staff') {  
    $where_clauses[] = "l.admin_id = ?";  
    $params[] = $current_user_id;  
    $types .= 'i';  
    $filter_user_id = $current_user_id;  
}  
  
if (!empty($filter_user_id)) {  
    // Add WHERE admin_id filter  
}  
  
if (!empty($filter_type)) {  
    // Add WHERE l.activity LIKE filter  
}  
  
if (!empty($filter_date)) {  
    // Add WHERE created_at DATE filter  
}
```

Rasberry Pi Mapping Code

```
#!/usr/bin/env python3
import asyncio
from bleak import BleakScanner
import requests
import time
import numpy as np
from scipy.optimize import least_squares
from collections import deque

# === MAP & BEACON SETTINGS ===
MAP_SIZE = 300 # 3m x 3m (in centimeters)
BEACONS = {
    "BL": ("F4:65:0B:55:07:36", (0, 0)),      # Back Left (0,0)
    "BR": ("EC:E3:34:BF:99:16", (MAP_SIZE, 0)), # Back Right (300,0)
    "FR": ("EC:E3:34:BE:87:E2", (MAP_SIZE, MAP_SIZE)), # Front Right (300,300)
    "FL": ("F4:65:0B:E8:EA:EE", (0, MAP_SIZE)), # Front Left (0,300)
}
SERVER_URL = "http://192.168.8.141:5050/cart_location"
# === RSSI TO DISTANCE SETTINGS (calibrate for your environment!) ===
TX_POWER = -59 # Typical default for 1m; calibrate for your beacons!
N = 3          # Environmental factor; adjust for your store/room

# === SMOOTHING BUFFER SETTINGS ===
BUFFER_SECONDS = 2.0 # Buffer time (seconds) for smoothing
SCAN_INTERVAL = 0.5 # How often to scan (seconds)
BUFFER_LEN = max(1, int(BUFFER_SECONDS / SCAN_INTERVAL))
position_buffer = deque(maxlen=BUFFER_LEN)

# === KALMAN FILTER PARAMETERS ===
KALMAN_DT = SCAN_INTERVAL
KALMAN_PROCESS_VAR = 1.0
KALMAN_MEAS_VAR = 800.0 # increase if environment is noisy (metal, multipath)

# === MOVEMENT DETECTION PARAMETERS ===
SPEED_THRESHOLD_CM_PER_S = 10.0 # speed > this => moving (10 cm/s)
DEBOUNCE_SECONDS = 0.8         # require state stable for this duration before flipping

def rssi_to_distance(rssi, tx_power=TX_POWER, n=N):
    # Log-distance path loss model
    meters = 10 ** ((tx_power - rssi) / (10 * n))
    return meters * 100 # convert to cm

def least_squares_trilateration(beacon_positions, distances):
    def fun(X, positions, dists):
        return [np.linalg.norm(X - np.array(pos)) - dist for pos, dist in zip(positions, dists)]
```

```

x0 = np.array([MAP_SIZE / 2, MAP_SIZE / 2])
res = least_squares(fun, x0, args=(beacon_positions, distances))
return float(res.x[0]), float(res.x[1])

# --- Simple 2D Kalman filter (constant velocity) ---
class KalmanFilter2D:
    """
    State vector: [x, y, vx, vy]^T
    Constant-velocity model. Use step((x,y), dt) to update.
    """

    def __init__(self, dt=0.5, process_var=1.0, meas_var=400.0):
        self.dt = float(dt)
        self.F = np.array([
            [1, 0, self.dt, 0],
            [0, 1, 0, self.dt],
            [0, 0, 1, 0],
            [0, 0, 0, 1]
        ], dtype=float)

        # We observe position x,y only
        self.H = np.array([
            [1, 0, 0, 0],
            [0, 1, 0, 0]
        ], dtype=float)

        # Covariances
        self.Q = process_var * np.eye(4) # process noise
        self.R = meas_var * np.eye(2) # measurement noise

        # Initial state and covariance
        self.x = np.zeros((4,1), dtype=float)
        self.P = np.eye(4) * 500.0

        self._initialized = False

    def set_dt(self, dt):
        self.dt = float(dt)
        self.F = np.array([
            [1, 0, self.dt, 0],
            [0, 1, 0, self.dt],
            [0, 0, 1, 0],
            [0, 0, 0, 1]
        ], dtype=float)

    def initialize(self, x, y):
        self.x = np.array([[x], [y], [0.0], [0.0]], dtype=float)

```

```

        self.P = np.eye(4) * 50.0
        self._initialized = True

    def predict(self):
        self.x = self.F @ self.x
        self.P = self.F @ self.P @ self.F.T + self.Q

    def update(self, meas):
        z = np.array([[meas[0]], [meas[1]]], dtype=float)
        S = self.H @ self.P @ self.H.T + self.R
        K = self.P @ self.H.T @ np.linalg.inv(S)
        y = z - (self.H @ self.x)
        self.x = self.x + K @ y
        I = np.eye(self.P.shape[0])
        self.P = (I - K @ self.H) @ self.P

    def step(self, meas, dt=None):
        if dt is not None:
            self.set_dt(dt)
        if not self._initialized:
            self.initialize(meas[0], meas[1])
            return (self.x[0,0], self.x[1,0])
        self.predict()
        self.update(meas)
        return (self.x[0,0], self.x[1,0])

# Instantiate Kalman filter
kf      = KalmanFilter2D(dt=KALMAN_DT,      process_var=KALMAN_PROCESS_VAR,
meas_var=KALMAN_MEAS_VAR)

async def scan_and_trilaterate():
    print(f"BLE trilateration starting on a {MAP_SIZE}x{MAP_SIZE}cm map...")
    print("Beacons (MAC: Position):")
    for name, (mac, pos) in BEACONS.items():
        print(f" {name}: {mac} -> {pos}")
    print(f'Buffer seconds: {BUFFER_SECONDS} | Scan interval: {SCAN_INTERVAL} | Buffer
length: {BUFFER_LEN}')

    last_state = "stopped"
    state_change_time = None
    last_time = time.time()

    while True:
        try:
            devices = await BleakScanner.discover(timeout=SCAN_INTERVAL)
        except Exception as e:
            print("BLE scan failed:", e)

```

```

await asyncio.sleep(SCAN_INTERVAL)
continue

# Build rssi_dict in a robust, version-independent way
rssi_dict = {}
for entry in devices:
    # BleakScanner.discover may return either:
    # - BLEDevice instances (with .rssi attribute) OR
    # - tuples (BLEDevice, AdvertisementData)
    dev = None
    adv = None
    if isinstance(entry, tuple) or (hasattr(entry, "__len__") and not isinstance(entry, type) and len(entry) == 2):
        try:
            dev, adv = entry[0], entry[1]
        except Exception:
            dev = entry
            adv = getattr(entry, "advertisement_data", None)
    else:
        dev = entry
        # advertisement_data attribute exists in some Bleak versions
        adv = getattr(dev, "advertisement_data", None)

    # Extract address
    addr = (getattr(dev, "address", None) or "").upper()

    # Prefer RSSI from AdvertisementData if present
    rssi = None
    if adv is not None:
        rssi = getattr(adv, "rssi", None)
    # Fallback to BLEDevice.rssi
    if rssi is None:
        rssi = getattr(dev, "rssi", None)
    # Some platforms put RSSI in metadata dict
    if rssi is None:
        md = getattr(dev, "metadata", None)
        if isinstance(md, dict) and "rssi" in md:
            rssi = md.get("rssi")

    if not addr:
        continue
    # match against known beacons
    for name, (mac, pos) in BEACONS.items():
        if addr == mac:
            if rssi is None:
                # ignore if no RSSI available
                continue

```

```

# basic sanity check
if abs(rssi) > 120:
    continue
rssi_dict[name] = rssi

print("\n--- BLE Scan ---")
print("Detected RSSIs:", rssi_dict)

if len(rssi_dict) >= 3:
    # Use up to four strongest beacons
    picked = sorted(rssi_dict.keys(), key=lambda k: rssi_dict[k], reverse=True)[:4]
    beacon_positions = [BEACONS[name][1] for name in picked]
    distances = [rssi_to_distance(rssi_dict[name]) for name in picked]
    print("Using beacons:", picked)
    print("Distances (cm):", [f"{d:.2f}" for d in distances])

try:
    x, y = least_squares_trilateration(beacon_positions, distances)
    # Clamp to map boundaries
    x = max(0.0, min(MAP_SIZE, float(x)))
    y = max(0.0, min(MAP_SIZE, float(y)))

    # Short buffer smoothing
    position_buffer.append((x, y))
    avg_x = float(sum(p[0] for p in position_buffer)) / len(position_buffer))
    avg_y = float(sum(p[1] for p in position_buffer)) / len(position_buffer))

    # Feed into Kalman filter
    filtered_x, filtered_y = kf.step((avg_x, avg_y), dt=SCAN_INTERVAL)

    # Estimate speed from Kalman state (vx, vy in cm/s)
    vx = float(kf.x[2, 0])
    vy = float(kf.x[3, 0])
    speed = (vx*vx + vy*vy)**0.5

    # Movement state with debounce
    now = time.time()
    desired_state = "moving" if speed >= SPEED_THRESHOLD_CM_PER_S else "stopped"
    if desired_state != last_state:
        if state_change_time is None:
            state_change_time = now
        elif (now - state_change_time) >= DEBOUNCE_SECONDS:
            last_state = desired_state
            state_change_time = None
    else:
        state_change_time = None

```

```

payload = {
    "x": float(filtered_x),
    "y": float(filtered_y),
    "timestamp": now,
    "moving": True if last_state == "moving" else False,
    "speed_cm_s": float(speed)
}

try:
    resp = requests.post(SERVER_URL, json=payload, timeout=1.5)
    print("Server response:", getattr(resp, "status_code", None), getattr(resp, "text", ""))
except Exception as e:
    print("Failed to send to server:", e)

print(f"Raw:({{x:.1f}},{ {y:.1f}}) avg:({{avg_x:.1f}},{ {avg_y:.1f}})
      filt:({{filtered_x:.1f}},{ {filtered_y:.1f}}) speed: {speed:.1f}cm/s state: {{last_state}}")

except Exception as e:
    print("Trilateration / processing failed:", e)
else:
    print(f"Not enough beacons detected. Found: {{rss_i_dict}}")

await asyncio.sleep(SCAN_INTERVAL)

if __name__ == "__main__":
    asyncio.run(scan_and_trilaterate())

```

APPENDIX G. CURRICULUM VITAE OF RESEARCHERS

Curriculum Vitae of
SHANNA MAY ALASAAS

Sauyo, Quezon City
alasaasshanna@gmail.com
+63-915-594-2490



EDUCATIONAL BACKGROUND

Level	Inclusive Dates	Name of school/ Institution
Tertiary	September 2022 – July 2026	STI College Novaliches
Senior High	June 2020 – June 2022	STI College Novaliches
High School	June 2016 – March 2020	San Bartolome High School
Elementary	June 2009 – March 2016	Benito R. Villar Memorial School

PROFESSIONAL OR VOLUNTEER EXPERIENCE

Inclusive Dates	Nature of Experience/ Job Title	Name and Address of Company or Organization
2025	Barista	Wealth Tea - Crossroad Tandang Sora, Sangandaan

AFFILIATIONS

Inclusive Dates	Name of Organization	Position
2025 – 2026	Student Executive Council	President
2024 – 2025	Student Executive Council	ICT Representative
2024 – 2025	Association of Computer Engineering Students (ACES)	Member
2023 – 2024	Association of Computer Engineering Students (ACES)	Member
2022 – 2023	Association of Computer Engineering Students (ACES)	Member

SKILLS

SKILLS	Level of Competency	Date Acquired
Basic Electronics and Circuit Design	Intermediate	2023
Basic Web Development (HTML, CSS, JavaScript)	Proficient	2020
Database Management (MySQL or similar)	Intermediate	2024
PCB Design and Soldering	Proficient	2023
Documentation and Report Writing	Advanced	2020

TRAININGS, SEMINARS OR WORKSHOP ATTENDED

Inclusive Dates	Title of Training, Seminar or Workshop
July 2025	Next-Gen AI in Action: Transforming Academe and Engineering Workflows
June 2025	The Future of Development: Will Vibe Coding Replace Traditional Programming
June 2025	10 Hours BOSH Course for SO1 with 2 hours Training of Trainer
April 2025	AI and prompt Engineering in Educational Settings
March 2025	Building Cloud-Native Telecom Services: The Role of SDN and NFV in Mobile and Broadband Networks
March 2025	Training and Education in Medical Imaging for AI in PACS: Equipping the Next Generation
March 2025	Hybrid AGV-drone using YOLOv8 and Arduino-Raspberry Pi for defect detection and structural health monitoring in built infrastructure
March 2025	Introduction to Cybersecurity: Securing the Digital World of Connectivity

Curriculum Vitae of
JASFER M. BACARISA

Gulod, Quezon City
bacarisajas@gmail.com
+63-916-7979-873



EDUCATIONAL BACKGROUND

Level	Inclusive Dates	Name of school/ Institution
Tertiary	September 2022 – July 2026	STI College Novaliches
Senior High	June 2019 – July 2021	Lagro High School
High School	June 2015 – April 2019	North Fairview High School
Elementary	June 2009 – March 2015	Rosa L. Susano Novaliches Elementary School

PROFESSIONAL OR VOLUNTEER EXPERIENCE

Inclusive Dates	Nature of Experience/ Job Title	Name and Address of Company or Organization
April 2022 – June 2024	Guest Experience Leader/Service Crew	McDonald's Philippines
August 2024 – Present	Customer Service Agent/Team Support	CONCENTRIX Philippines

AFFILIATIONS

Inclusive Dates	Name of Organization	Position
July 2024 – Present	SKEP-SLEC (Youth Organization)	Secretary
May 2021 – July 2024	SKEP-SLEC (Youth Organization)	Vice President
September 2024 – September 2025	Association of Computer Engineering Students (ACES)	Secretary

SKILLS

SKILLS	Level of Competency	Date Acquired
System Analyst	Satisfactory	2025
Documentation	Intermediate	2020
Office 365 (MS Word, PPT, Excel, etc.)	Satisfactory	2022
Computer and Printer	Satisfactory	2022
Literate/Troubleshooting		
Database Management (MySQL or similar)	Intermediate	2025

TRAININGS, SEMINARS OR WORKSHOP ATTENDED

Inclusive Dates	Title of Training, Seminar or Workshop
June 2025	10 Hours BOSH Course for SO1 with 2 hours Training of Trainer
May 2025	Finalist of QC START-UP Student Competition 2025
December 2024	Cyber Security Fundamentals: Understanding the Risks and Building Effective Defense
July 2023	Philippine Diocesan Youth Camp & Workshop 2023

Curriculum Vitae of
ANGELO MIGUEL R. CUA

Novaliches Bayan, Quezon City
angelo@gmail.com
+63-993-9842-272



EDUCATIONAL BACKGROUND

Level	Inclusive Dates	Name of school/ Institution
Tertiary	September 2022 – July 2026	STI College Novaliches FEATI University
Senior High School	June 2017 – June 2019 June 2011 – March 2017	Manila Central University Tandang Sora Integrated School
Elementary	June 2004 – March 2011	Philippine Cultural College

PROFESSIONAL OR VOLUNTEER EXPERIENCE

Inclusive Dates	Nature of Experience/ Job Title	Name and Address of Company or Organization
2018 – Present	Freelance Computer Technician	2018 – Present

AFFILIATIONS

Inclusive Dates	Name of Organization	Position
2024 – 2025	Association of Computer Engineering Students (ACES)	Member
2023 – 2024	Association of Computer Engineering Students (ACES)	Member
2022 – 2023	Association of Computer Engineering Students (ACES)	Member

SKILLS

SKILLS	Level of Competency	Date Acquired
Basic Electronics and Circuit Design	Advanced	2023
Basic Web Development (HTML, CSS, JavaScript)	Proficient	2020
Database Management (MySQL or similar)	Intermediate	2024
PCB Design and Soldering	Proficient	2023
Documentation and Report Writing	Advanced	2020

TRAININGS, SEMINARS OR WORKSHOP ATTENDED

Inclusive Dates	Title of Training, Seminar or Workshop
July 2025	Next-Gen AI in Action: Transforming Academe and Engineering Workflows
June 2025	The Future of Development: Will Vibe Coding Replace Traditional Programming
June 2025	10 Hours BOSH Course for SO1 with 2 hours Training of Trainer
April 2025	AI and prompt Engineering in Educational Settings
March 2025	Building Cloud-Native Telecom Services: The Role of SDN and NFV in Mobile and Broadband Networks
March 2025	Training and Education in Medical Imaging for AI in PACS: Equipping the Next Generation
March 2025	Hybrid AGV-drone using YOLOv8 and Arduino-Raspberry Pi for defect detection and structural health monitoring in built infrastructure

Curriculum Vitae of
SHAINE FAYE G. SALTING

Nagkaisang Nayon, Quezon City
Shainefsalting@gmail.com
+63-947-6376-022



EDUCATIONAL BACKGROUND

Level	Inclusive Dates	Name of school/ Institution
Tertiary	September 2022 – July 2026	STI College Novaliches
Senior High	June 2020 – June 2022	STI College Novaliches
High School	June 2016 – March 2020	San Bartolome High School
Elementary	June 2009 – March 2016	San Bartolome Elementary School

PROFESSIONAL OR VOLUNTEER EXPERIENCE

Inclusive Dates	Nature of Experience/ Job Title	Name and Address of Company or Organization
2024 – Present	Logistics and Social Com	Parokya ng Banal na Sakramento, QC
2023 – Present	Semi-Virtual Assistant	Printing Works and Services, QC
2023 – 2024	Assistant Manager	Source VHub, QC

AFFILIATIONS

Inclusive Dates	Name of Organization	Position
July 2025 – June 2026	Student Executive Council	Vice President
Sept 2024 – June 2025	Association of Computer Engineering Students (ACES)	Vice President
Sept 2023 – July 2024	Association of Computer Engineering Students (ACES)	2 nd Year Representative

SKILLS

SKILLS	Level of Competency	Date Acquired
Computer and Printer	Satisfactory	January 2024
Troubleshooting	Satisfactory	October 2023
Java Fundamentals, OOP	Satisfactory	January 2023
Java Fundamentals, PLD	Satisfactory	August 2020
Microsoft 365	Mastered	

TRAININGS, SEMINARS OR WORKSHOP ATTENDED

Inclusive Dates	Title of Training, Seminar or Workshop
June 2025	ICpEP, Blockchain is not just Crypto
June 2025	The Future of Development: Will Vibe Coding Replace Traditional Programming
June 2025	10 Hours BOSH Course for SO1 with 2 hours Training of Trainer
Dec 2024	Cybersecurity Fundamentals: Understanding the Risks and Building Effective Defenses
Feb 2021	Webinar in Marketing Trends and Issues