TALA-KALAKAL: A Study on the Enhancement of Digital Inventory Management System for Sari-Sari Stores

Final Laboratory Project
Submitted to the Faculty of the
Department of Computer Studies
Cavite State University
Bacoor City Campus
Bacoor, Cavite

In Partial Fulfillment of the
requirements for Course of
DCIT50: Object Oriented Programming

BALDO, NEMELYN O.

MACAWILI, ADRIAN KERBY C.

MAGDAMIT, MARY STRELLINE R.

MATAVIA, ANGELO LUIGI G.

ORTIGA, JESSAMAY O.

PESIGAN, JAMES CARLO R.

QUELANG, LIEZL S.

January 2025

DCIT50FL2025-02

TALA-KALAKAL: A Study on the Enhancement of Digital Inventory Management for Sari-Sari Stores

BALDO, NEMELYN O.

MACAWILI, ADRIAN KERBY C.

MAGDAMIT, MARY STRELLINE R.

MATAVIA, ANGELO LUIGI G.

ORTIGA, JESSAMAY O.

PESIGAN, JAMES CARLO R.

QUELANG, LIEZL S.

A final project manuscript submitted to the faculty of the Department of Computer Studies, Cavite State University – Bacoor City Campus, City of Bacoor, Cavite in partial fulfilment of the requirements for Course of DCIT50: Object Oriented Programming with Abstract No. BSCS-02 Prepared under the supervision of Mr. Julios M. Mojas.

Abstract

Sari-sari stores played an essential role in the Philippine economy, providing local communities with necessary goods. Sari-sari store owners encountered challenges such as disjointed supply chains and inefficient inventory management. Decisions relied on intuition rather than accurate data, leading to inefficiencies and missed opportunities. These challenges hindered growth and profitability of sari-sari stores, complicating adaptation to modern business demands. Advancements in digital tools showed potential in addressing challenges by streamlining processes and enhancing inventory tracking. Digital tools offered opportunities to bridge gaps between traditional practices and modern technological innovations for small and medium-sized enterprises (SMEs), including sari-sari stores. Research focused on enhancing a digital inventory management system specifically designed for sari-sari stores, with features tailored to meet operational needs.

Adoption of digital technologies enabled more effective inventory management, reduced operational inefficiencies, improved customer satisfaction, and enhanced overall operational efficiency. Digital tools provided better control over stock levels, ensured product availability when needed, and minimized losses from overstocking or spoilage. Research highlighted the broader potential of technology to transform small business operations.

A developmental research design was employed, focusing on software development to create an enhanced digital inventory management system for sari-sari stores. Findings contributed to a deeper understanding of how digital technology supported small businesses in growing and strengthening economic roles within local communities.

Keywords: (Sari-sari stores; digital inventory management; SMEs; inventory tracking; digital technology; economic growth; operational efficiency)

INTRODUCTION

1.1 Background

Sari-sari stores served as vital components of Filipino communities by providing access to daily necessities. Operational challenges in inventory management hindered efficiency and profitability. Traditional methods, such as manual recording, were time-consuming and prone to inaccuracies, leading to stockouts, wasted capital, and potential legal issues. Poor inventory management often resulted in expired products being solds unknowingly (Condura, 2020), which created further complications (Wasp Barcode Technologies, 2022).

Digital inventory management systems addressed these challenges by streamlining operations, automating stock tracking, and enabling real-time monitoring of stock levels. Automation reduced errors, ensured availability of high-demand products, and minimized losses from overstocking or spoilage. Effective inventory management improved operational efficiency and supported sustainability in business operations. Research emphasized the importance of adopting digital inventory systems specifically designed for sari-sari stores to enhance competitiveness. Modern inventory practices strengthened economic roles within local communities while helping businesses adapt to evolving market demands.

1.2 Problem

Current inventory management practices among Sari-sari stores are inefficient and often lead to operational challenges. The research seeks to address the following issues:

1. Sari-sari store owners often rely on traditional methods like recording inventory in notebooks or relying on memory which are time-consuming and prone to errors.

2. Many Sari-sari stores struggle to monitor stock levels accurately leading to frequent issues such as running out of high-demand items (stockouts) or accumulating excess inventory.
3. Inventory losses often go undetected due to poor tracking which can result from theft, damage, or discrepancies in stock records.
4. Store owners often fail to track the expiration dates of products leading to wasted capital from disposing of expired goods or unknowingly selling them which risks customer dissatisfaction and potential legal issues.
5. Significant time and effort are wasted on manual inventory tasks which reduces the attention available for customer service and other important business activities.

1.3 Objectives

The study aims to enhance the digital inventory management of Sari-sari stores. The following objectives guide the direction of this research:

1. Developed a user-friendly digital inventory management system specifically designed to address the needs of Sari-sari store owners.
2. Evaluate the impact of the digital inventory management system on key performance indicators, including inventory accuracy, sales efficiency, waste reduction, and profitability of Sari-sari stores.
3. Assess the usability, accessibility, and overall acceptability of the developed system among Sari-sari store owners to ensure its practicality and effectiveness in real-world operations.

1.4 Significance

The significance of the study lay in the potential to empower sari-sari store owners with essential tools and knowledge to enhance business operations. By providing a practical digital solution, the research

supported the growth and sustainability of sari-sari stores, benefiting owners and local communities. The following were identified as the beneficiaries of the study:

1. **Sari-Sari Store Owners -** Main beneficiaries included sari-sari store owners who gained access to a digital tool that increased profitability, decreased waste, and improved inventory management. Enhanced operations and increased business growth resulted from using the system to make data-driven decisions.

2. **Small and Medium-Sized Enterprises (SMEs) -** The study served as a model for similar SMEs, demonstrating how technology could be applied effectively to address operational challenges, enabling growth and sustainability in other industries.

3. **Customers -** Improved service quality, such as better product availability and timely restocking of necessary items, allowed customers of sari-sari stores to benefit indirectly and achieve greater satisfaction.

METHODOLOGY

2.1 Research Design

A developmental research design was employed in the study, specifically focusing on software development to create an enhanced digital inventory management system for sari-sari stores. The **ADDIE** method was utilized for the development and refinement of the system based on user feedback and testing.

2.2 Participants/Respondents

The target users involved in evaluating the development of the system included sari-sari store owners, operators, and industry experts in inventory management. Insights and experiences from sari-sari store owners, operators, and industry experts were crucial for ensuring that the system met the practical needs of small retail businesses.

2.3 Procedures

The development process consists of several key phases:

1. **Analysis** (Needs Assessment) - Based on the previous related studies, surveys and interviews were conducted to identify the inventory management needs and challenges faced by Sari-sari store owners and operators.
2. **Design** (Mockups and System Architecture) - Based on the analysis, mockups and system architecture were created to visualize the proposed system's layout and functionality.
3. **Development** (Prototyping and Coding) - The system was developed using Java as the primary programming language, applying object-oriented programming approaches to the system. A

prototype was created to demonstrate the system's functionalities, allowing for initial user interactions.

4. **Implementation** (Deployment and User Testing) - The prototype was deployed, followed by user testing with Sari-sari store owners and operators to gather feedback on usability, functionality, and overall user experience.
5. **Evaluation -** Usability testing was conducted with target users to assess their experience and satisfaction with the system.

2.4 Tools and Technologies

The study utilized appropriate tools and technologies to ensure the development of an efficient digital inventory management system. Java served as the programming language, while Eclipse and NetBeans functioned as integrated development environments (IDEs). The program design employed an object-oriented programming (OOP) approach, which provided a structured and modular framework for coding. The OOP approach allowed the breakdown of complex problems into smaller, manageable components, each encapsulating data and behavior. Object-oriented programming promoted code reusability, maintainability, and scalability, enabling easier extension or modification of the application when necessary.

RESULTS

The researcher successfully developed Tala-Kalakal, an enhanced digital inventory management system for Sari-sari stores. Evaluation results indicated significant improvements in inventory tracking accuracy, reduced manual errors, and fewer stockouts and overstocking incidents. Sari-sari store owners reported increased efficiency in inventory management, with quicker updates and report generation. User feedback highlighted the system's user-friendly interface and practical features tailored to small retail businesses. Real-time inventory monitoring facilitated informed decision-making regarding restocking and sales trends. Overall, Tala-Kalakal effectively addressed operational challenges, enhancing business operations and customer satisfaction while contributing to the profitability and sustainability of sari-sari stores.

3.1 Development Output

The following content outlines the system's features, functionalities, and user interface, demonstrating how the application addressed operational challenges identified during the study. The "Add Item" feature allowed addition of new items to the inventory with key details, including name, quantity, category, and expiration date. The "Remove Item" feature enabled deletion of items from the inventory, which proved useful for removing discontinued, expired, or irrelevant items, thereby maintaining a clean and focused inventory consisting only of active items. The "View Items" feature provided a detailed overview of all inventory items. The "Update Item Quantity" feature permitted modification of stock quantities in the inventory, ensuring accurate tracking of current inventory levels based on new shipments, sales, or losses.

Below are the screenshots of the system prototype:

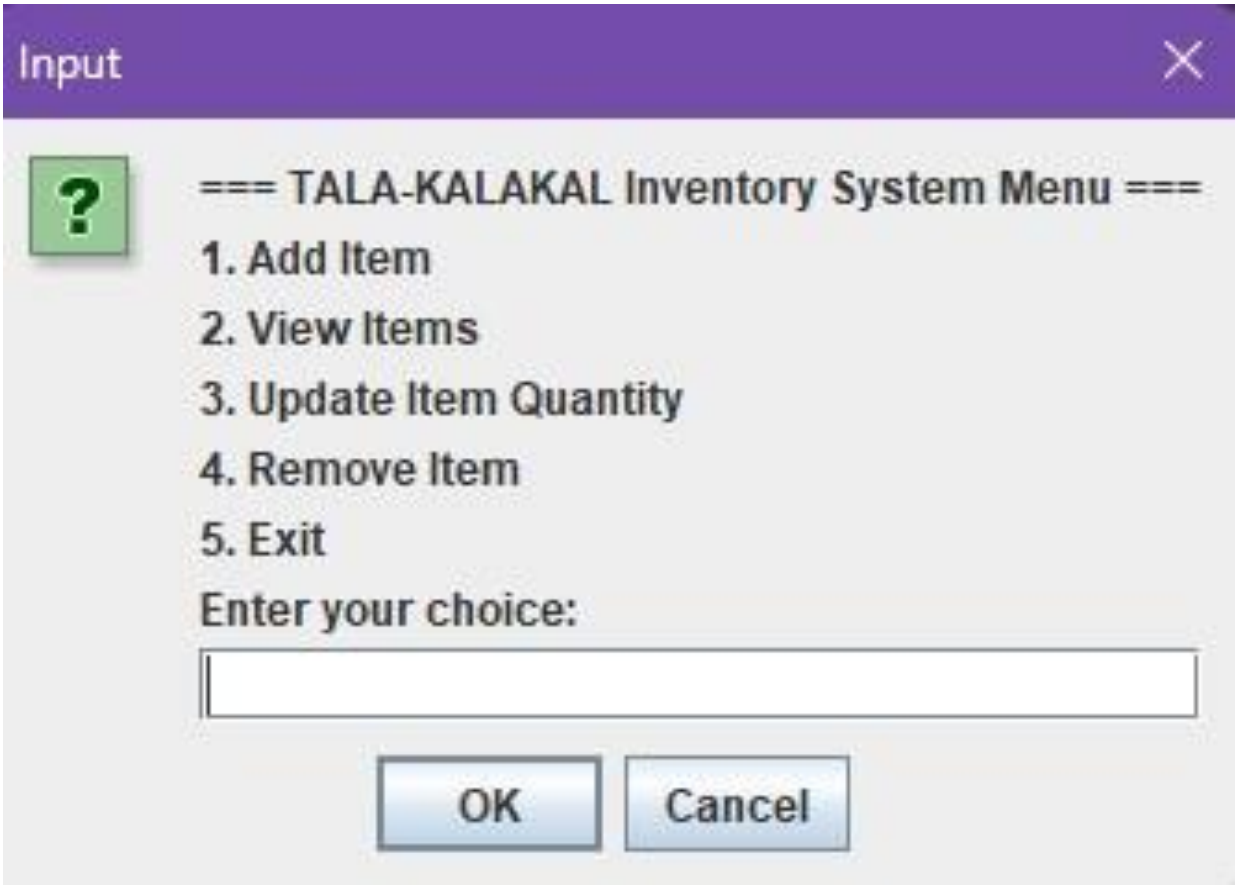

Figure 1. Tala – kalakal: - Main Menu

The main menu serves as the starting point for navigation, providing access to different sections or functionalities**.**

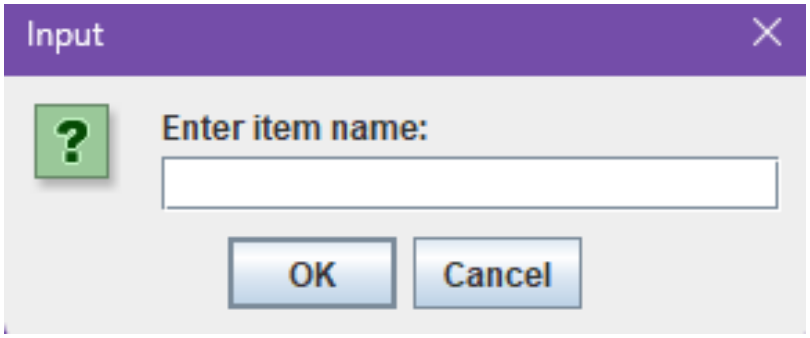

Figure 2. Tala – kalakal: Enter item

The name helped in identifying the product, making tracking, management, and differentiation from other items easier.

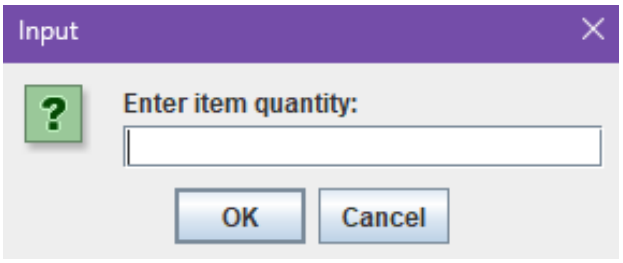

Figure 3. Tala – kalakal: Enter quantity

Inventory levels were tracked accurately by specifying the number of units of a particular item. Sufficient stock was ensured to meet demand without overstocking, resulting in unnecessary costs.

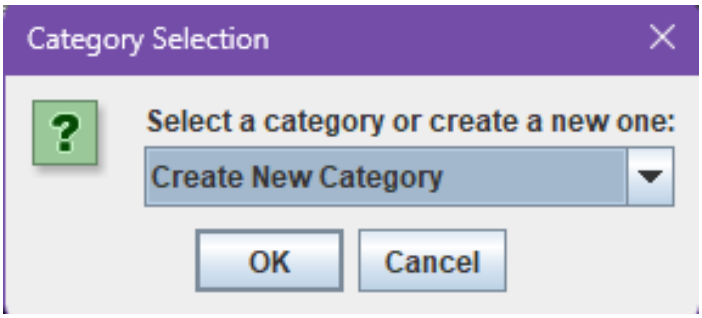

Figure 4. Tala – kalakal: Selecting category

Crucial for organizing products in a way that enhances customer experience, improves searchability, and optimizes inventory management.

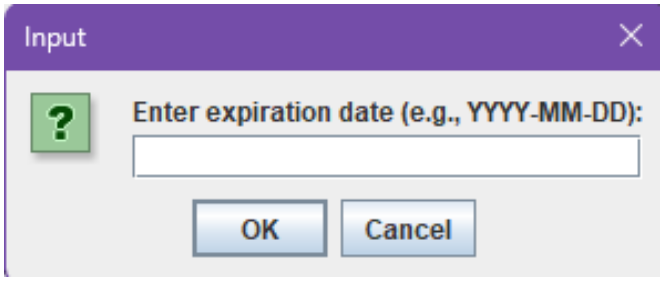

Figure 5. Tala – kalakal: Enter Expiration Date

Entering an expiration date for a product involves specifying the last date by which the product should be consumed or used to ensure its quality and safety.

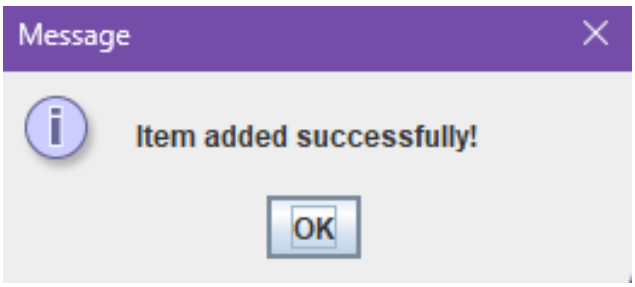

Figure 6. Tala – kalakal: Added Successfully

The message confirms that the addition of a new product to the inventory has been completed successfully

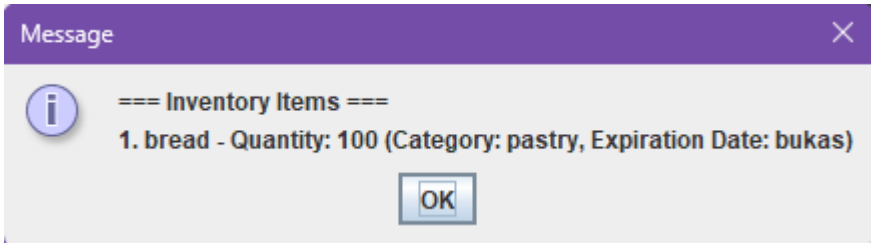

Figure 7. Tala – kalakal: Inventory Items

Inventory items referred to tangible goods that were purchased or produced for resale purposes.

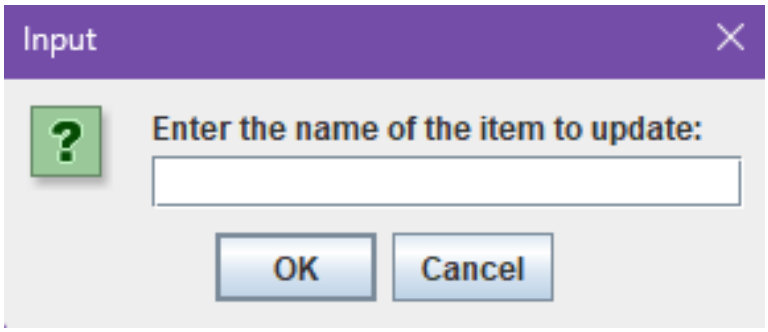

Figure 8. Tala – kalakal: Item Update

Include changes to product details, pricing, quantity, or status within an inventory management system.

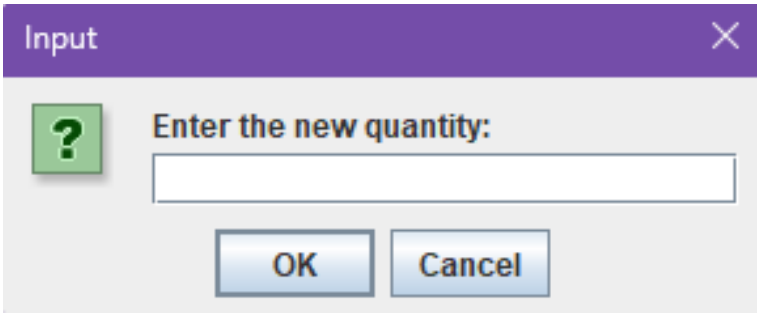

Figure 9. Tala – kalakal: New Quantity

Occurred due to various reasons such as inventory counts, sales, returns, or other transactions that affected the stock level.

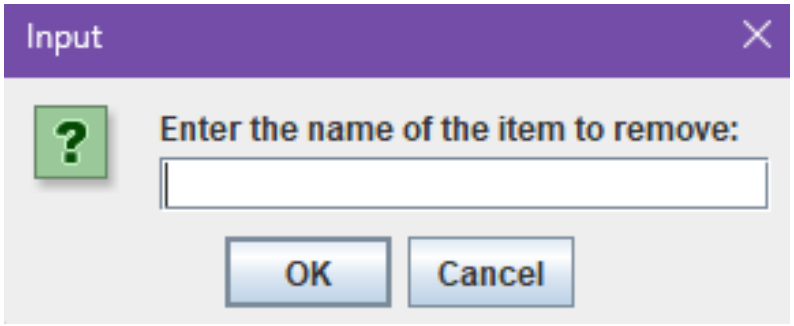

Figure 10. Tala – kalakal: Item Remove

Inventory items were removed from the system when no longer stocked, discontinued, or when inventory records needed to be cleaned up

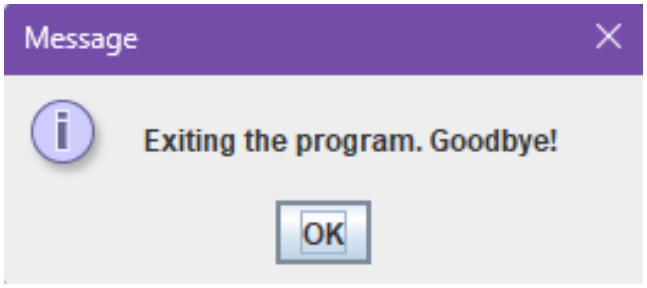

Figure 11. Tala – kalakal: Exit Program

The exit command was used to stop the program from running. The process involved closing all active tasks and freeing up system resources. The program's functionality was ceased, and control was returned to the operating system.

3.2 Evaluate Findings

The effectiveness of the digital inventory management system was evaluated using measurable criteria such as stockout rate, inventory turnover ratio, sales growth, consumer satisfaction, time spent on inventory management, and accuracy of inventory data. Related studies indicated that providing real-time data on inventory levels drastically reduced stockouts and improved purchasing decisions (Fast.com, 2023). Furthermore, stock levels were optimized based on sales trends due to improved analytics-based decision-making capabilities, leading to higher profitability (Qashier, 2023). Effective methods for inventory management reduced waste and increased customer retention by ensuring that items in demand were available at all times (Course Hero, 2023). Moreover, technological advances automated inventory processes, reduced manual errors, and allowed a stronger focus on customer service and sales (Subscribed, 2023). By comparing pre- and post-data, clear evidence are provided regarding the system's impact on inventory management practices, sales performance, and overall business efficiency, ultimately contributing to the sustainability and expansion of sari-sari stores within the community.

3.2 Data Presentation

The table presents a summary of the key metrics used to evaluate the effectiveness of the system implementation. The metrics have been measured before and after the system implementation.

METRIC	PRE- IMPLEMENTATION	POST-IMPLEMENTATION
Inventory Accuracy	65	95
Reduction in Stockouts	50	90
Reduction in Spoilage	40	85
User Satisfaction with Usability	60	92

Table 1. Evaluation of the pre-implementation and post-implementation

DISCUSSION

6.1 Interpret Findings

The findings of the study show how using a digital inventory management system can greatly improve sari-sari stores. By moving from old methods to a digital solution, store owners can work more efficiently. Automating stock tracking and supply orders helps prevent running out of popular items or having too much stock, which are common problems for these stores. This change not only makes managing inventory easier but also allows owners to spend more time on customer service and growing their business. Additionally, the study suggests that digital tools enable sari-sari store owners to make informed decisions based on data, leading to higher profits and less waste.

6.2 Comparison

When comparing the results of the research with existing literature, clear challenges faced by sari-sari stores emerged, such as poor inventory management and dependence on manual processes (Wasp Barcode Technologies, 2022; Condura, 2020). Other studies demonstrated that using technology for inventory management led to better outcomes for small businesses. For example, effective inventory tracking systems improved product availability and customer satisfaction (Subscribed.fyi, 2024). Findings supported existing ideas, showing that digital inventory management solved current problems and facilitated sustainable growth for sari-sari stores in a competitive market.

6.3 Strengths and Limitations

One of the main strengths of this study is its focus on user-centered design, which ensures that the system developed meets the specific needs of sari-sari store owners. By involving participants in the evaluation process, the research gathers valuable feedback that helps improve the system. However, there are some limitations to consider. The study mainly focuses on a specific group of sari-sari store owners,

which may limit how applicable the findings are to other small retail businesses. Additionally, while the initial results are promising, more research is needed to understand the long-term effects on profitability and operational efficiency.

6.4 Recommendations

To optimize the advantages of digital inventory management systems for sari-sari stores, continual training and assistance were provided to owners and employees to enable effective utilization of the system and facilitate adaptation to new technologies. Research explored advanced technologies, such as artificial intelligence and machine learning, which offered valuable insights into customer behavior and inventory trends. Additionally, features like expiration date notifications, item search capabilities, items arrangement by category, and the generation of detailed inventory reports were enhanced to improve operational efficiency. Collaboration with local suppliers was established to further enhance supply chain efficiency while strengthening community ties, creating a more integrated and responsive business environment for sari-sari stores.

CONCLUSION

The study aimed to enhance digital inventory management and address operational problems faced by sari-sari stores, including manual inventory management, inaccurate stock tracking, unaccounted losses, and operational inefficiencies. A developmental research design focused on software development was utilized to create a user-friendly and cost-effective system specifically designed for the needs of sari-sari stores.

Findings contributed to a deeper understanding of how digital technologies transformed the operations of small and medium-sized enterprises (SMEs). Practical tools were provided to improve business efficiency and strengthen the role of micro-enterprises in the local economy.

In summary, the developed system highlighted the potential of technology to support small businesses in adapting to the digital age. Sustainable growth and innovation were encouraged, emphasizing the importance of digital solutions in improving economic progress and enhancing the livelihoods of small business owners in the Philippines.

REFERENCES

Condura. (2020). *Inventory management practices of sari-sari stores in Malolos City, Bulacan and its relationship with business performance*.

FAST. (2024, June 14). Keeping up with customers: How retailers can avoid stockouts. *FAST Blog*.

FAST. (2024, July 1). Smart business moves: ways to make sari-sari stores more profitable. *FAST Blog*.

Perez, J. (2022). *Optimizing inventory management for sari-sari stores: A study in Malolos City*.

Subscribed.fyi. (2024, October 11). Inventory management in a sari-sari store: Practical insights. *Subscribed.fyi Blog*.

Wasp Barcode Technologies. (2022). *The importance of inventory management for small businesses*.

4S Vlogs. (2019). First-in-first-out strategy for stock level optimization in sari-sari stores.

Qashier. (2023, December 18). Managing inventory: Best practices for mini-marts to boost sales.

Creswell, J. W. (2014). *Research Design: Qualitative, Quantitative, and Mixed Methods Approaches*. SAGE Publications. This book provides comprehensive guidance on conducting needs assessments and qualitative research methods, including surveys and interviews.

Patton, M. Q. (2008). *Utilization-Focused Evaluation*. SAGE Publications. This resource discusses how to assess needs effectively and gather relevant data from stakeholders.

Sommerville, I. (2011). *Software Engineering* (9th ed.). Addison-Wesley. A comprehensive overview of software development processes, including prototyping methods and coding practices.

Beck, K., & Andres, C. (2005). *Extreme Programming Explained: Embrace Change* (2nd ed.). Addison-Wesley. This book introduces agile development methodologies that emphasize iterative development and user feedback.

Rubin, J., & Chisnell, D. (2008). *Handbook of Usability Testing: How to Plan, Design, and Conduct Effective Tests* (2nd ed.). Wiley. A practical guide on conducting user testing to gather feedback on usability during the implementation phase.

Nielsen, J., & Norman, D. A. (2013). *Designing User Experience: The Interaction Design Foundation*. Retrieved from Interaction Design Foundation. This article provides insights into user experience design principles relevant to testing and implementation.

International Organization for Standardization (ISO). (2018). *ISO 9241-11: Ergonomics of human-system interaction — Part 11: Usability: Definitions and concepts*. ISO. This standard outlines the principles of usability evaluation and measurement.

Dumas, J. S., & Redish, J. C. (1999). *A Practical Guide to Usability Testing*. Greenwood Publishing Group. This guide offers practical advice on usability testing methods that can be applied during the evaluation phase.

Ngubane, N. & Mayekiso, S. (2015) Inventory Management Systems used by Manufacturing Small Medium and Micro Enterprises in Cape Town Factors Influencing the Effectiveness of Inventory Management in Manufacturing SMEs.

APPENDICES

Below is the flowchart of the system.

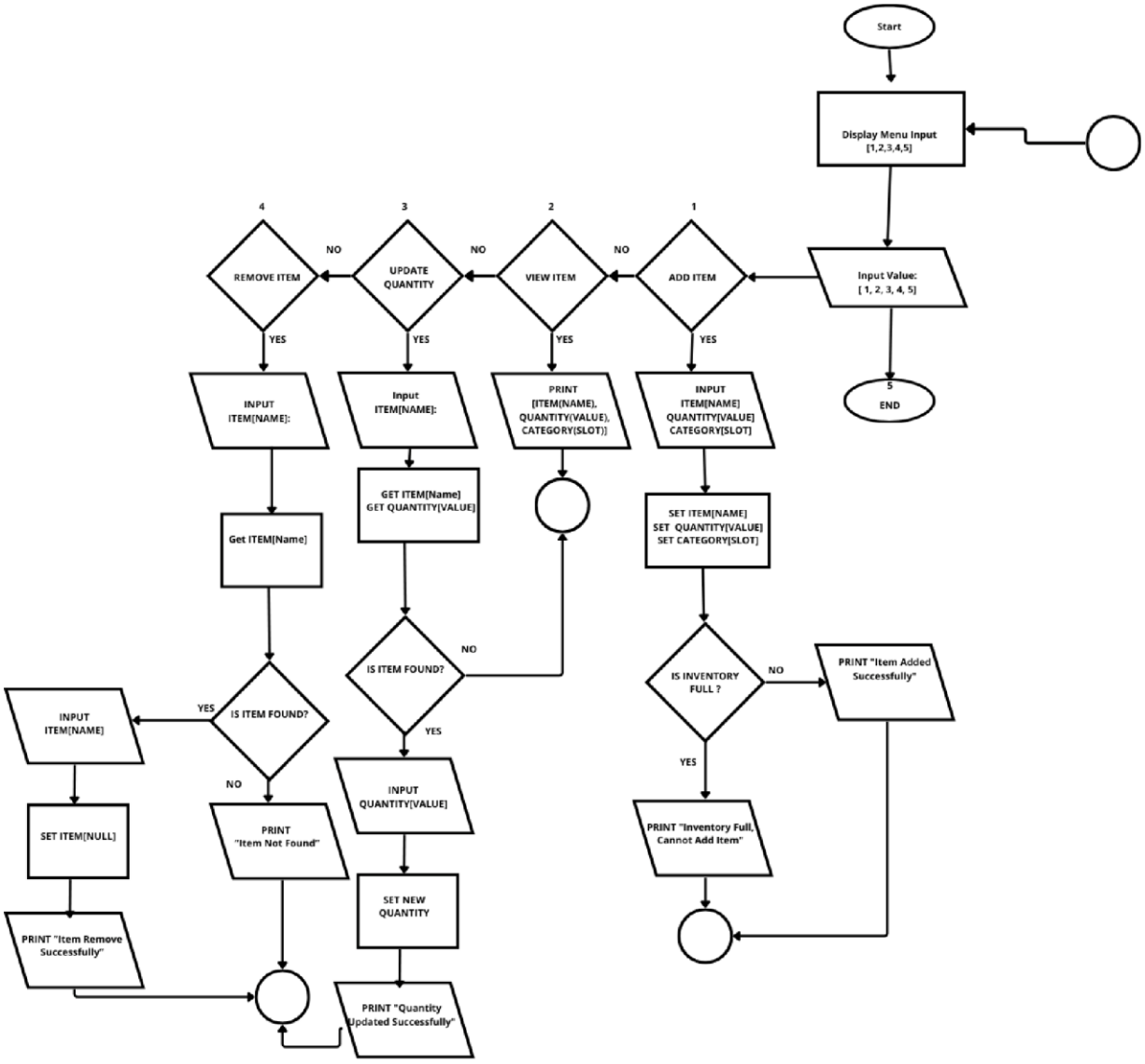

Figure 12. Flowchart of the system

SOURCE CODE

InventorySystem Class

```
package finalproj;

import javax.swing.JOptionPane;

public class InventorySystem {

 public static void main(String[] args) {

 Inventory inventory = new Inventory(1000); // Inventory capacity of 100 items  int choice;

 do {

 String menu = """

 === TALA-KALAKAL Inventory System Menu ===  1. Add Item

 2. View Items

 3. Update Item Quantity

 4. Remove Item

 5. Exit

 Enter your choice:""";

 String input = JOptionPane.showInputDialog(menu);

 if (input == null) break; // Exit if user cancels

 try {

 choice = Integer.parseInt(input);

 } catch (NumberFormatException e) {

 JOptionPane.showMessageDialog(null, "Invalid input! Please enter a number.");  continue;
```

```
}

switch (choice) {

case 1 -> {

String name = JOptionPane.showInputDialog("Enter item name:");  if (name == null) continue;

String quantityInput = JOptionPane.showInputDialog("Enter item quantity:");

if (quantityInput == null) continue;

int quantity;

try {

quantity = Integer.parseInt(quantityInput);

} catch (NumberFormatException e) {

JOptionPane.showMessageDialog(null, "Invalid quantity! Please enter a valid number.");  continue;

}

String[] categories = inventory.getCategories();

String category = (String) JOptionPane.showInputDialog(

null,

"Select a category or create a new one:",

"Category Selection",

JOptionPane.QUESTION_MESSAGE,

null,

categories,

categories[0]
```

```
);

if (category == null) continue;

if (category.equals("Create New Category")) {

category = JOptionPane.showInputDialog("Enter the new category name:"); if (category == null || category.isEmpty()) {

JOptionPane.showMessageDialog(null, "No category created. Item not added."); continue;

}

inventory.addCategory(category);

}

String expirationDate = JOptionPane.showInputDialog("Enter expiration date (e.g., YYYY MM-DD):");

if (inventory.addItem(name, quantity, category, expirationDate)) { JOptionPane.showMessageDialog(null, "Item added successfully!"); } else {

JOptionPane.showMessageDialog(null, "Inventory is full. Cannot add more items."); }

}

case 2 -> {

String items = inventory.viewItems();

JOptionPane.showMessageDialog(null, items);

}

case 3 -> {

String name = JOptionPane.showInputDialog("Enter the name of the item to update:"); if (name == null) continue;

String quantityInput = JOptionPane.showInputDialog("Enter the new quantity:"); if (quantityInput == null) continue;
```

```
try {

int newQuantity = Integer.parseInt(quantityInput);

if (inventory.updateItem(name, newQuantity)) {

JOptionPane.showMessageDialog(null, "Item quantity updated successfully!"); } else {

JOptionPane.showMessageDialog(null, "Item not found in the inventory."); }

} catch (NumberFormatException e) {

JOptionPane.showMessageDialog(null, "Invalid quantity! Please enter a valid number."); }

}

case 4 -> {

String name = JOptionPane.showInputDialog("Enter the name of the item to remove:"); if (name ==
null) continue;

if (inventory.removeItem(name)) {

JOptionPane.showMessageDialog(null, "Item removed successfully!"); } else {

JOptionPane.showMessageDialog(null, "Item not found in the inventory."); }

}

case 5 -> {

JOptionPane.showMessageDialog(null, "Exiting the program. Goodbye!"); System.exit(0);

}

default -> JOptionPane.showMessageDialog(null, "Invalid choice. Please try again."); }

} while (true);

}
```

```
}
```

Inventory Class

```
package finalproj;

class Inventory {
 private Item[] items;
 private String[] categories;
 private int itemCount;
 private int categoryCount;
 public Inventory(int capacity) {
 items = new Item[capacity];
 categories = new String[capacity];
 itemCount = 0;
 categoryCount = 0;
 }
 public boolean addCategory(String category) {
 if (categoryCount >= categories.length) {
 return false; // Categories array is full
 }
 for (int i = 0; i < categoryCount; i++) {
 if (categories[i].equalsIgnoreCase(category)) {
 return true; // Category already exists
```

```
}

}

categories[categoryCount++] = category;

return true;

}

public String[] getCategories() {

String[] existingCategories = new String[categoryCount + 1];  System.arraycopy(categories, 0, existingCategories, 0, categoryCount);

existingCategories[categoryCount] = "Create New Category"; // Add option for creating a new category

return existingCategories;

}

public boolean addItem(String name, int quantity, String category, String expirationDate) {  if (itemCount >= items.length) {

return false; // Inventory is full

}

items[itemCount++] = new Item(name, quantity, category, expirationDate);  return true;

}

public String viewItems() {

if (itemCount == 0) {

return "The inventory is empty.";

}
```

```
StringBuilder inventoryList = new StringBuilder("=== Inventory Items ===\n");  for (int i = 0; i < itemCount; i++) {

inventoryList.append(String.format("%d. %s - Quantity: %d (Category: %s, Expiration Date: %s)%n",

i + 1, items[i].getName(), items[i].getQuantity(), items[i].getCategory(), items[i].getExpirationDate()));

}

return inventoryList.toString();

}

public boolean updateItem(String name, int newQuantity) {

for (int i = 0; i < itemCount; i++) {

if (items[i].getName().equalsIgnoreCase(name)) {

items[i].setQuantity(newQuantity);

return true; // Item found and updated

}

}

return false; // Item not found

}

public boolean removeItem(String name) {  for (int i = 0; i < itemCount; i++) {

if (items[i].getName().equalsIgnoreCase(name)) {  // Shift items to fill the gap

for (int j = i; j < itemCount - 1; j++) {  items[j] = items[j + 1];

}
```

```
items[--itemCount] = null; // Clear the last item  return true; // Item removed

}

}

return false; // Item not found

}

}
```

Item Class

```
package finalproj;

class Item {

private String name;

private int quantity;

private String category;

private String expirationDate;

public Item(String name, int quantity, String category, String expirationDate) {  this.name = name;

this.quantity = quantity;

this.category = category;

this.expirationDate = expirationDate;

}

public String getName() {

return name;

}
```

```
public int getQuantity() {

return quantity;

}

public void setQuantity(int quantity) {

this.quantity = quantity;

}

public String getCategory() {

return category;

}

public String getExpirationDate() {

return expirationDate;

}

}
```