Philippine Copyright © 2024

by the Author
and the
College of Engineering, Department of Computer Engineering
Polytechnic University of the Philippines

All rights reserved. Portions of this manuscript may be reproduced with proper referencing and due acknowledgment of the author.

POLYTECHNIC UNIVERSITY OF THE PHILIPPINES

Multi-Grain Rice Dispensing Machine with Short Messaging System Notification

A Thesis
Presented to the Faculty of Computer Engineering
Polytechnic University of the Philippines
Sta.Mesa, Manila

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

by

Almazan, Gabrielle Angelo
Batralo, Gabrielle
Bumatay, Xypher Kelly
Quisquino, Dann Joseph

July 2024

CERTIFICATION

This thesis, *MULTI-GRAIN RICE DISPENSING MACHINE WITH SHORT MESSAGING SYSTEM NOTIFICATION* prepared and submitted by Gabrielle Angelo Almazan, Gabrielle E. Batralo, Xypher Kelly D. Bumatay, and Dann Joseph R. Quisquino in partial fulfillment of the requirements for the degree, Bachelor of Science in Computer Engineering has been examined and recommended for Oral Examination.

Evaluation Committee

ENGR. PEDRITO M. TENERIFE JR.
Chairman

ENGR. ROLITO L. MAHAGUAY
Member

ENGR. JOHN R. DELA CRUZ
Member

DR. ANTONIO Y. VELASCO
Member

APPROVAL

Approved by the panel on Oral Examination on June 14, 2024 with the grade of **PASSED**.

ENGR. JULIAN LORICO JR.
Lead Panel

ENGR. JOSHUA BENJAMIN RODRIGUEZ
Panel Member

ENGR. JAN REUELLE TEÑA
Panel Member

Accepted in partial fulfilment of the requirements for the degree Bachelor of Science in Computer Engineering.

DR. REMEDIOS G. ADO
Dean, College of Engineering

Acknowledgements

The researchers would like to express their gratitude to the following individuals and families for their invaluable support throughout this research journey.

The researchers would like to thank their thesis advisers, Engr. Dela Cruz and Engr. Velasco. Their knowledge, motivation, and guidance helped the researchers carry out their research work. The researchers offer their sincere appreciation for the learning opportunities provided by the advisors.

To the researchers' parents and family, Almazan-Careng Family, Batralo Family, Bumatay Family, and Quisquino Family, who have continuously supported them financially, emotionally, and spiritually so that they are able to accomplish this research. An extended gratitude for the families who sheltered the researchers during their development phase up until their defense.

To the Department of Science and Technology (DOST) for providing financial assistance for the development of the research in support of the undergraduate scholars.

To the researcher’s friends and batchmates, especially Danniel P. Martinez, who helped the researchers during the development and consultation of the software.

To Prof. Mon Ferdinand Prado, who helped the researchers during the early stages of development and Engr. Danilo Ramos Cabatan, who helped with the validation of the structure of the machine.

CERTIFICATION OF ORIGINALITY

This is to certify that the research work presented in this thesis *MULTI-GRAIN RICE DISPENSING MACHINE WITH SHORT MESSAGING SYSTEM NOTIFICATION* for the degree Bachelor of Science in Computer Engineering at the Polytechnic University of the Philippines embodies the result of original and scholarly work carried out by the undersigned. This thesis does not contain words or ideas taken from published sources or written works that have been accepted as basis for the award of a degree from any other higher education institution, except where proper referencing and acknowledgment were made.

ALMAZAN, GABRIELLE ANGELO
Researcher

BATRALO, GABRIELLE E.
Researcher

BUMATAY, XYPHER KELLY D.
Researcher

QUISQUINO, DANN JOSEPH R.
Researcher

ABSTRACT

Title	:	Multi-Grain Rice Dispensing Machine with Short Messaging System Notification
Name of Researchers	:	Gabrielle Angelo Almazan, Gabrielle E. Batralo, Xypher Kelly D. Bumatay, Dann Joseph R. Quisquino
Degree	:	Bachelor of Science in Computer Engineering
Institution	:	Polytechnic University of the Philippines
Year	:	2024
Advisers	:	Engr. John R. Dela Cruz, Dr. Antonio Y. Velasco

This study developed and evaluated a Multi-Grain Rice Dispensing Machine with a Short Messaging System Notification, named Ricemate, to improve rice dispensing in commercial settings. The machine features a touchscreen interface for selecting various rice types, precise dispensing via load sensors, and real-time environmental monitoring with temperature and humidity sensors. It also includes an SMS notification system for operational updates.Structural assessments confirmed its robustness and adherence to ISO standards. Testing showed minimal errors in dispensing weights, indicating high reliability. While user feedback suggested improvements for the software interface, overall performance was positively received. Future enhancements include additional payment options, automated plastic bag dispensing, and deployment in high-traffic areas to maximize utility and profitability.

Keywords: Automated Rice Dispensing, Multi-Grain Selection, Operational Efficiency, User-Friendly Interface, Short Messaging System

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Chapter 1

THE PROBLEM AND ITS SETTING

1.1 Introduction

Entrepreneurship has increasingly become a critical driver of economic development, prompting a shift from traditional employment towards self-employment and business ownership. This trend reflects a broader recognition of entrepreneurship as a means of achieving financial independence and mitigating poverty. As noted by Cudia, Rivera, and Tullao Jr. (2019), entrepreneurial activity significantly enhances the likelihood of escaping poverty and maintaining a stable economic position above the poverty line. In the Philippines, this shift is particularly evident, with a growing number of individuals pursuing entrepreneurial ventures as a strategic alternative to conventional employment.

The ‘Tugon ng Masa’ survey conducted by OCTA Research in October 2022 illustrates this trend, indicating that 78 percent of Filipinos express a preference for starting their own businesses rather than working as employees. The primary motivations for this inclination include the desire for greater autonomy, the absence of hierarchical oversight, and the potential for higher income (Concepcion, 2023). These findings suggest a cultural shift towards entrepreneurship as a preferred pathway to economic self-sufficiency and personal empowerment.

Within this entrepreneurial landscape, the deployment of dispensing machines represents a promising business opportunity, particularly for products with high consumer demand. Dispensing machines offer automated, efficient, and contactless transactions, aligning with contemporary consumer preferences for convenience. According to projections by 6W Research, the dispensing machine market in the Philippines is expected to experience substantial growth from 2022 to 2028, driven by increasing demand for automated retail solutions.

Rice, a staple commodity in the Philippines, presents a significant opportunity within this market. Traditional rice retailing practices are often manual and expose the product to environmental elements, potentially compromising its quality. The need for modernization in this sector has led to the development of automated rice dispensing machines. For example, Cristi et al. (2017) developed a rice dispensing machine, although it was limited to dispensing a single type of rice. This development underscores the potential for innovation in rice retailing, particularly in enhancing automation and product preservation.

The current market conditions present a compelling case for further advancement in rice dispensing technologies. Addressing the limitations of existing systems, such as the lack of multi-grain dispensing capabilities and the exposure of rice to air, could significantly improve the efficiency and hygiene of rice retailing. Innovations in this area, including the integration of advanced features like real-time monitoring and environmental controls, have the potential to elevate the functionality of rice dispensing machines. Such advancements would not only offer a viable business opportunity for entrepreneurs but also contribute to the broader modernization of the agricultural retail sector.

The increasing focus on entrepreneurship in the Philippines, coupled with the growth of the dispensing machine market, presents significant opportunities for innovation in rice retailing. By addressing existing challenges and leveraging technological advancements, entrepreneurs can contribute to the development of more efficient, hygienic, and consumer-friendly rice dispensing solutions, thereby enhancing both business viability and consumer satisfaction in this essential sector.

The "Multi-Grain Rice Dispensing Machine with Short Messaging System Notification" is designed to address the limitations associated with rice grain storage, dispensing, and availability in commercial settings. This automated system aims to provide an efficient and hygienic method for dispensing multiple types of rice, ensuring precise measurement and minimizing waste. The machine is equipped with a casing that facilitates easy monitoring of grain levels and quality, thereby enhancing operational efficiency.

The proposed solution eliminates the manual processes typically involved in rice retailing, such as opening and closing containers, weighing rice, and direct seller-customer interaction. The machine incorporates several key features, including a touchscreen interface for user input, load sensors to ensure accurate dispensing, humidity sensors to monitor moisture levels, and a GSM notifier to alert the owner regarding the status and condition of the rice. The payment system, which includes both bill and coin acceptors, is integrated to streamline the transaction process.

This automated dispensing machine is intended to benefit both entrepreneurs and consumers by providing a practical solution for rice distribution within specific communities. The development of this machine draws on the established principles of dispensing and vending machines, aiming to support small-scale business ventures and improve access to rice in local markets. The machine's potential impact extends to individuals, communities, and the broader economy, particularly in the Philippines, a major rice-producing country (De Castro et al., 2020; Marinelli et al., 2020).

1.2 Theoretical Framework

A. Finite State Machines (FSMs) and Theory of Automata

This study is anchored on concept of finite state machines (FSM), which refers to a set of rules that guide how something behaves. It is a flowchart that shows different steps or states that something can be inputted, and it changes from one state to another based on the input it receives. They are abstract machines that follow a predetermined sequence of operations. The finite state machine is a conceptual model under the theory of automata. The word automata comes from the Greek word for "self-making," which suggests that they can automatically process the development of a particular work (Kumar, 2021).

One application of automata is in dispensing machines. When coins are inserted into a dispensing machine, the automata inside the machine will automatically process your payment and dispense the item you selected. Kumar (2021) also stated that the automata principle is also widely used in many programs that are based on the idea of finite state machines (FSMs). FSMs are a type of automata that can be used to model the behavior of systems with a limited number of states.

B. Base Framework from Related Studies

In the study “Design of an ARM-based Automatic Rice-Selling Machine for Cafeterias” (Kang et al., 2016), an input, process, and output (IPO) model is used to propose the general structure for their proposal system.

The input, process, and output (IPO) model is a widely used framework in computer science and systems analysis. It breaks down a system into three components: input, processing, and output. The input is the data provided to the system; the process involves manipulating and analyzing the input; and the output is the result or outcome of the processing. The IPO model helps in designing efficient systems by ensuring that the right inputs are processed correctly to produce the desired outputs (Dennis et al., 2021).

However, the components and design are different for developing an automatic rice-selling machine. Their study included an ultrasonic sensor, pressure sensor, optoelectronic switch, and keyboard that can be understood by an Advanced RISC Machines (ARM) microprocessor. The

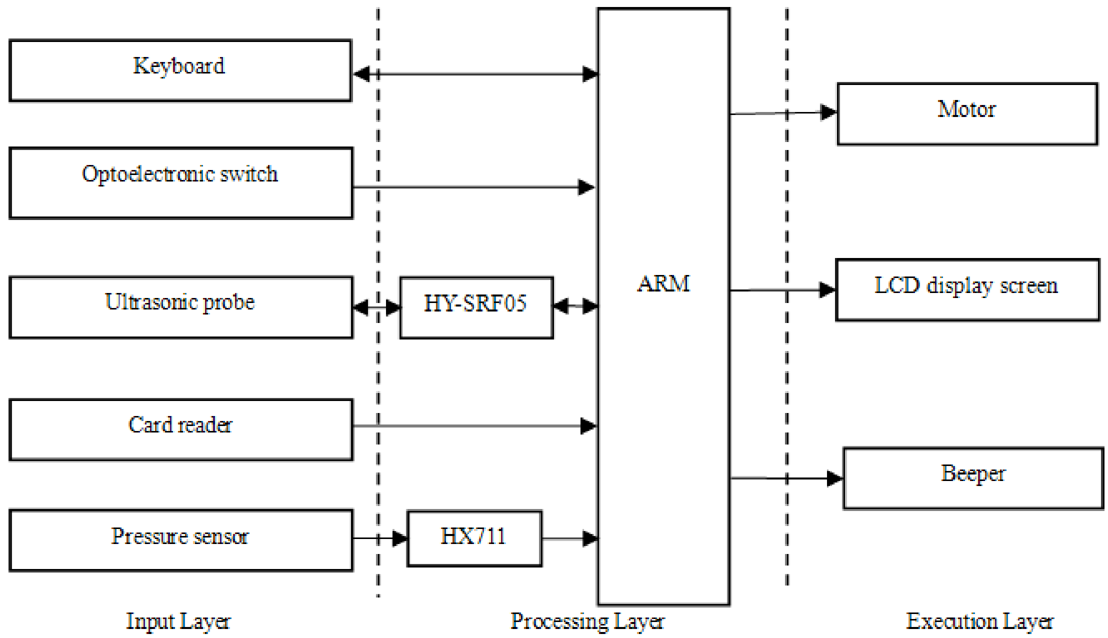


Figure 1.1. Integral Structure Framework of Related Studies

Figure 1 shows an illustration showing the layered structure and system design of the afforementioned research; the diagram is used with the layers of Input, Process, and Execution. In regards to this study of “Multi-Grain Rice Dispensing Machine with Short Messaging System Notification'', the research of Kang et al. (2016) provides a suitable IPO model guide since it utilizes the concept and functions of an automated dispensing machine. Their study are modified and adapted to suit different types of food dispensing machines, such as those for selling snacks, drinks, or other types of food.

1.3 Conceptual Framework

A. Base Diagram Model (IPO)

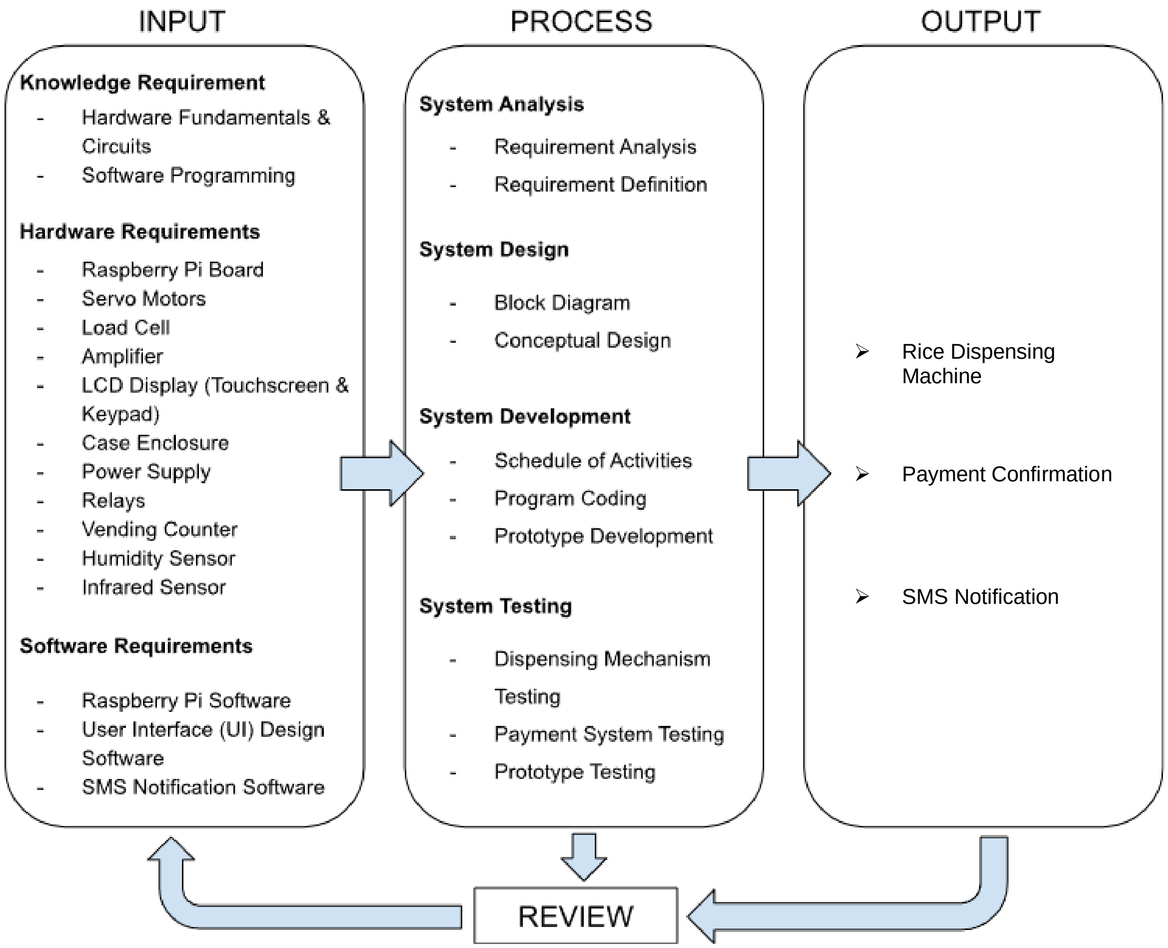


Figure 1.2. Conceptual Framework

Figure 2 is a conceptual diagram showing the design of the prototype's input, process, and output (IPO). The Input part of the diagram consists of the knowledge, software, and hardware requirements of the system. The three requirements will undergo several processes that are indicated in the center diagram. The processes are for the system implementation of requirements analysis, design, development, and testing. Thus, producing the research outputs that result in the development of a "Multi-Grain Rice Dispensing Machine with Short Messaging System Notification".

The Review block is an act of ascertaining whether the completed system is acceptable as described in the research study. If not, then the input is to be adjusted and/or the process must be re-modified or repeated.

B. Components Used

The Multi-Grain Rice Dispensing Machine with Short Messaging System Notification have various parts such as a sturdy cabinet, a dispensing mechanism for each type of rice, a control system, a touchscreen interface, and a payment system. The processes involved in making the machine includes designing the cabinet and dispensing mechanism, programming the control system, developing the touchscreen interface, installing the payment system, and testing the machine to ensure it functions properly.

Creating the "Multi-Grain Rice Dispensing Machine with Short Messaging System Notification" using Raspberry Pi would require a blend of hardware and software components. The **Raspberry Pi 4 Model B 2GB** could serve as the primary microcontroller for this project, as these boards can manage the operations of a dispensing machine and are also simple to program.

Servo Motors are used to control the dispensing machine's dispensing mechanism, with the number of motors required depending on the number of different types of rice grains the machine is designed to dispense. A **load cell**, a type of sensor used to measure weight, measures the quantity of dispensed rice. The **HX711 Amplifier**, a precision 24-bit analog-to-digital converter (ADC) designed for weigh scales and industrial control applications, which then interfaces with the load cell.

The **DHT22 temperature and humidity sensor** is used to read the container temperature and moisture content of the rice. This helps ensure the rice is not too dry or too wet. An Infrared (IR) Sensor is implemented to detect the presence of plastic packaging in the dispenser and catch the dispensed rice during the operation. This ensures that the rice is automatically prepared for the customers. A **7 Inch Touch-Screen LCD** is used to provide information to the user, such as the type and quantity of rice being dispensed, and is used for user input, allowing the user to select the type and quantity of rice they want.

The **Global System for Mobile Communication (GSM)** module sends Short Message Service (SMS) notifications to users when their rice is ready. This is a convenient way for the user to be alerted when their rice is ready, and it also helps to ensure that the rice is not left in the machine for too long. A suitable **power supply** is implemented to power the Raspberry Pi board and other components. **Relays** is used to control the power for the motors and other components. Various electronic components, such as **resistors, capacitors, diodes, and wires**, are required for the circuit connections. A suitable enclosure is made to house all the components, which could be custom-made or repurposed as based on existing dispensing machines.

On the software side, the **Raspberry Pi OS** is integrated to program the Raspberry Pi board. The program controls the motors, read from the load cell, handle user input, and display information on the LCD screen. This is a complex project that requires a good understanding of both hardware and software engineering, and safety guidelines should always be followed when working with electronic components.

1.4 Statement of the Problem

This study aims to develop a multi-grain rice dispensing machine automation system with an SMS notification and dispensing system that can streamline the process while providing versatility in handling different grain varieties.

The Multi-Grain Dispensing Machine with Short Messaging System Notification addresses the problems of:

1. What are the stages in the development of a Multi-Grain Dispensing Machine with Short Messaging System Notification?
2. What are the competitive advantages between the proposed study and the existing system in terms of:
   2.1 Structural Design; and
   2.2 System Integrations?
3. Which features need to be integrated into the dispensing machine to address the problem in terms of:
   3.1 Multiple Grain Options;
   3.2 User-friendly Interface; and
   3.3 Payment and Security?
4. How efficient is the system in terms of its capability with regards to:
   4.1 Dispensing Automation;
   4.2 Measurement in kilograms; and
   4.3 Sustainability?

1.5 Scope and Limitations of the Study

The Multi-Grain Rice Dispensing Machine with Short Messaging System Notification has an automated solution for the storage and dispensing of various rice types. The machine is designed to enhance convenience and efficiency by minimizing manual handling, ensuring accurate dispensing, and accommodating multiple rice varieties within separate compartments. It is equipped with a touchscreen interface, allowing users to select the desired rice type and quantity, with transactions limited to three kilograms per purchase. Payment is facilitated through integrated bill and coin acceptors, accepting Philippine peso coins in denominations of 1, 5, 10, and 20 pesos, and bills in denominations of 20, 50, and 100 pesos. The system is restricted to accepting only these specified denominations due to their relevance in typical rice transactions.

The machine's construction utilizes stainless steel, aluminum casing, acrylic glass, and composite panels, providing durability and enabling users to visually monitor rice levels without opening the containers. Additionally, rubberized seals and humidity sensors are incorporated to maintain the freshness and quality of the stored rice. Each container is designed to store a maximum of 12 kilograms of rice, and the machine is equipped with a temperature and humidity sensor positioned in the upper section of the storage compartment.

This system is limited to three varieties of rice and does not support the storage or dispensing of liquid substances. The machine must be deployed in locations with reliable cellular signals to support the Short Messaging System (SMS) notifications, which alert the owner to the machine's status. It is also restricted to shaded areas and locations near the owner for maintenance and monitoring purposes.

The dispensing machine requires a continuous power supply and is not equipped with a battery backup. In the event of a power outage, any ongoing transaction will be terminated, and any inserted money will be voided. The machine will resume operation once power is restored but will not continue the interrupted transaction.

For maintenance, the machine requires periodic self-cleaning, and the plastic bags used in the packaging system must be refilled by the owner due to their limited capacity. The payment system is configured to reject any currency or objects not conforming to the accepted denominations, with excess payments being rendered void. The coin acceptance mechanism accommodates coins with superficial markings, dirt, and minor bends, provided they do not exceed 3 mm in thickness. The machine's design exclusively supports Philippine peso coins and bills, rendering foreign currencies and non-coin objects incompatible. The machine must be placed in an area with at least 2G data capability to ensure the proper functioning of SMS notifications to the owner.

1.6 Significance of the Study

The study's significance spans various stakeholders, including shop vendors, the community, grain producers, technological innovators, and policymakers. It explores how the machine can reduce labor costs, increase efficiency, enhance customer satisfaction, and contribute to sustainability by reducing food waste. Additionally, it highlights the benefits of improved access to fresh grains, new market opportunities for producers, insights for tech development, and evidence to support informed policy decisions.

Shop vendors are the main beneficiaries of the study, which reveals how this technology reduces labor costs, increases efficiency due to its automation, and enhances customer satisfaction. It could also provide insights into how vendors can effectively implement and manage such technology.

The **community** plays a vital role in showing how this technology contributes to sustainability goals by reducing food waste. It could also demonstrate how it improves access to fresh grains, particularly in areas where such access is currently limited, contributing to food security and community health.

Grain producers are the main sources of rice production, showing how this technology opens up new markets, distribution channels, and mediums for grain producers to sell their products, potentially increasing demand.

Technological innovators refer to people who could provide insights into customer preferences and operational challenges that inform the design of future iterations of this technology, contributing to its ongoing improvement and adaptation.

Policymakers inform policy decisions related to food distribution and waste reduction. It provides evidence to support policies that encourages such technologies, contributing to broader sustainability and food security goals.

1.7 Definition of Terms

This section defines key terms essential for understanding the components and functionalities of the rice dispensing machine. It clarifies the roles of various parts such as the actuator, dispenser, and microcontroller, as well as features like multi-grain capability and the user interface. Additionally, it explains the importance of sensors in the machine's operation and the use of Short Message Service (SMS) for communication.

The ***actuator*** is essential component responsible for controlling the flow of rice during dispensing, translating electrical signals into physical movements within the machine.

A ***dispenser*** makes an integral part of the machine designed to accurately measure and release the desired quantity of rice chosen by the user.

The ***machine*** pertains to the prototype, representing the comprehensive assembly of components and mechanisms working synergistically to fulfill the machine's purpose.

A ***microcontroller*** is a small-scale computing device (Raspberry Pi) embedded within the machine, orchestrating the operation of various components.

The term ***multi-Grain*** refers as a feature allowing the machine to store and dispense several types of rice simultaneously.

Multiple ***sensors*** are devices integral to the machine's operation, detecting changes in temperature and humidity conditions, external shock or vibrations, and presence of plastic bags.

Short Messaging Service (SMS) refers to the machines communication protocol utilized for sending text message notifications to the machine owner.

The ***User Interface (UI)*** is a facilitating interaction between users and the machine, typically through transactions with the machine.

Chapter 2

REVIEW OF LITERATURE AND STUDIES

2.1 Agriculture and Technological Trends

Automation in agriculture has been facilitated by technological advancements, such as robotics, artificial intelligence, and machine learning. These technologies have been used to develop automated systems that can perform tasks traditionally done by humans, often with greater speed and accuracy (Farooq et al., 2020).

In the context of rice production, automation has been used at various stages of the production process. Automated rice planting machines can plant rice seeds at a consistent depth and spacing, resulting in uniform growth and increased yield (Jing et al., 2020). Automated rice harvesting machines can harvest rice faster and more accurately than workers, reducing waste and increasing yield (Farooq et al., 2020).

Automation in agriculture has revolutionized the sector, leading to increased productivity, efficiency, and profitability. Automation has been integrated into various agricultural processes, including planting, harvesting, sorting, packaging, and dispensing (Munz & Schuele, 2022). This literature review focuses on the role of automation in agriculture, particularly in the context of a Multi-Grain Rice Dispensing Machine with Short Messaging System Notification.

However, the implementation of automation in agriculture also presents challenges. These include the high cost of automated systems, the need for skilled workers to operate and maintain these systems, and the potential for job displacement (Munz & Schuele, 2022). Furthermore, there are concerns about the environmental impact of automation, as automated systems often require significant amounts of energy to operate (Farooq et al., 2020).

Automation plays a crucial role in agriculture, offering potential benefits in increased productivity, efficiency, and profitability. However, these benefits must be balanced against the challenges and potential drawbacks of automation. The development of a Multi-Grain Rice Dispensing Machine with Short Messaging System Notification represents an interesting application of automation in agriculture, and further research is needed to assess its feasibility and potential impact.

2.2 Dispensing Machines

The use of dispensing machines has gained significant popularity in various industries, providing convenient and accessible solutions for consumers. This comprehensive literature review explores the existing research and literature related to automated dispensing machines and their impact on convenience, accessibility, and customer satisfaction when purchasing grains, specific rice, and other multi-grain options.

Dispensing machines have become an integral part of modern society, revolutionizing how various products are accessed and purchased. These self-service machines offer convenience and accessibility, allowing individuals to satisfy their immediate needs without the need for traditional retail settings. From snacks and beverages to toiletries and electronics, dispensing machines have expanded their reach to encompass a wide range of goods.

2.3 Related Similar Machines

The study titled Automatic Medicine Vending Machine (Bhande, Tambe, et al., 2021) summarizes a dispensing machine designed to provide medicine 24/7. The machine stores various types of medicine, including painkillers, cold medicine, fever reducers, and more. It utilizes components such as the Atmega16 controller, ESP8266 WiFi module, DC motor, IR sensor, and motor driver IC (L293D) to operate. Users can place orders through a webpage, select the desired medicine and quantity, and make payments. The order is then communicated to the controller via the ESP8266 module. This automatic medicine dispensing machine aims to meet the needs of remote and rural areas where access to medical stores is limited.

The presence of rice dispensing machines has emerged as a game-changing development in rice distribution, providing an innovative and efficient method of accessing rice grains. These machines, specifically designed to dispense rice grains in measured quantities, have grown in significance and importance in modern food systems.

A study titled Precision Rice Vending Machine by Using Multiple Load Cell and IoT Based (Yuliandoko, Panduardi, et al., 2023) focuses on enhancing the accuracy and efficiency of a Rice Dispensing Machine through IoT integration, incorporating load cells, and a precise rice dispensing mechanism. This study helps people track the donated and provided rice in real-time for Zakat, Infaq, and Sadaqah (ZIS) funding. ZIS funding is an Islamic practice where people donate to address economic and societal issues. The analysis demonstrates the accuracy of the machine and the efficacy of website monitoring. This novel approach to ZIS management has the potential to benefit society while also streamlining the distribution of funds more efficiently and reliably.

2.4 Machine Demands and Retails

The advent of automated dispensing machines has significantly transformed the retail landscape, offering consumers a convenient and efficient way to purchase a wide range of products. The integration of technology into dispensing machines, such as the use of Raspberry Pi microcontrollers, has further enhanced their appeal. Raspberry Pi-based dispensing machines can offer a range of advanced features, such as touchscreen interfaces, cashless payment options, and real-time inventory tracking (Liu et al., 2021). These features can improve the user experience, increasing customer satisfaction and loyalty.

Rice retail enterprises are prevalent in the Philippines since rice is a basic diet for the consuming populace. The constant demand for grain indicates that commerce is profitable. Young rice retail establishments frequently use paper for inventory tracking and transaction documentation (De Castro et al., 2020). Multi-grain rice, which is a variation of different types of rice grains, has gained popularity due to its numerous health benefits. This literature review explores the importance and benefits of multi-grain rice, particularly in the context of an automated multi-grain rice dispensing machine.

The increasing demand for multi-grain rice presents an opportunity for the development of an automated multi-grain rice dispensing machine. Such a machine could provide consumers with a convenient and efficient way to purchase different types of multi-grain rice. This could potentially increase the accessibility of multi-grain rice, contributing to improved dietary diversity and health outcomes (Carcea, 2021).

However, challenges are also associated with the production and consumption of multi-grain rice. These include the need for more complex farming practices, the higher cost of multi-grain rice compared to single-grain rice, and consumer preferences for the taste and texture of single-grain rice (Maitra et al., 2021).

2.5 Dispensing Machine Benefits

Multi-grain rice is a rich source of dietary fiber, vitamins, minerals, and other essential nutrients (Carcea, 2021). The consumption of multi-grain rice has been associated with a reduced risk of chronic diseases such as heart disease, diabetes, and certain types of cancer (Schwingshackl et al., 2017). Furthermore, multi-grain rice has a low glycemic index, making it suitable for individuals with diabetes (Carcea, 2021).

In addition to its health benefits, multi-grain rice also has potential environmental benefits. Growing different types of rice in the same field, a practice known as intercropping, can increase biodiversity, improve soil health, and reduce the need for chemical fertilizers and pesticides (Maitra et al., 2021). This makes multi-grain rice a more sustainable choice compared to single-grain rice.

The concept of a Multi-Grain Rice Dispensing Machine with Short Messaging System Notification fits into this broader trend of automation in agriculture. Such a machine could dispense different types of multi-grain rice, providing consumers with a convenient and efficient way to purchase rice. This could potentially increase the accessibility of multi-grain rice, which has been associated with various health benefits (Bouchard et al., 2022).

The rise in the variety of snack foods has created a higher demand for appealing snack food packaging. To effectively capture consumers' attention, it is crucial to meet their packaging preferences, thereby influencing their purchasing decisions (Wang et al., 2023). However, the utilization of packaging solutions to minimize food waste faces challenges, such as public criticism of plastic packaging. Furthermore, there is evidence suggesting that some consumers have reservations about packaging technologies aimed at reducing food waste and lack sufficient knowledge regarding these technologies (Brennan et al., 2023). Brennan et al.'s (2023) research indicate that raising consumer awareness and understanding of food waste as an environmental concern can be combined with continuous advancements in packaging innovations that reduce food waste without solely relying on consumer demand for sustainable food packaging.

2.6 Customer Insights

Marinelli et al. (2020) suggest that utilizing customer insights can enhance the performance of automated retail environments for companies. The integration of non-traditional retail channels, including automatic dispensing machines (AVMs), introduces fresh hurdles in acquiring customer knowledge, which poses challenges for researchers and industry professionals. Within the realm of knowledge management, analyzing customer behavior sparks intense and ongoing discussions. Consequently, companies can leverage customer behavior analysis to enhance the context of their AVMs. Nevertheless, further exploration of the intricate relationship between customer behavior and the AVM's context is necessary, especially considering the evolving patterns of consumer consumption.

Vending machines and dispensers have become increasingly popular in various settings as they provide convenient and easy access to goods and services. User satisfaction is considered a crucial factor in the design and success of vending machines and dispensers. This research aimed to investigate the existing literature related to user satisfaction with vending machines and dispensers.

2.7 User Satisfactions

Several studies have investigated user satisfaction with vending machines and dispensers. According to Mostowfi & Naeini (2015) on their study of using QUIS (The Questionnaire for User Interaction Satisfaction) as a measurement tool for user satisfaction evaluation, QUIS can be used to measure user satisfaction with vending machines. The study found that the user satisfaction and experience of the use of a particular vending machine were high overall. Similarly, Lee (2022) on "Consumers' experiences, opinions, attitudes, satisfaction, dissatisfaction, and complaining behavior with vending machines" found that vending machine users have a high involvement but a low level of satisfaction with vending machine services, suggesting that there may be areas for improvement in vending machine design and function.

Other studies have investigated the design and location of vending machines. A study of reliable location design of unmanned vending machines based on customer satisfaction by Wang & Yao (2021) proposed a method to calculate customer satisfaction based on vending machines' locations. The study proposes a two-stage approach to mining customer preferences and behaviors. In the first stage, the study proposes a multi-dimensional measurement to analyze customers' preferences and satisfaction based on behavioral information. In the second stage, the study applies a clustering method to analyze the candidate points based on their location and customer preferences.

Raposo, et al. (2018) focused on vending machines and university students' consumption trends, it presents that vending machines are a significant source of food for university students and suggested that vending machines offer healthier food options.

Discussions about the satisfaction of users on having multiple choices (like in vending) in terms of selling various products all in one machine. In the study by Liu, et al. (2018) on “User Intent, Behaviour, and Perceived Satisfaction in Product Search”, participants' user intent, behavior, and perceived satisfaction were investigated in product search. The study found that users may have different intentions and behaviors when searching for products, but overall, perceived satisfaction is positively related to search success and the level of perceived usefulness of the search engine.

Another study by Zhao, et al. (2021) about the impact of pricing and product information on consumer buying behavior with customer satisfaction in a mediating role. Their research investigated the effects of pricing and product information on consumer buying behavior and to explain consumer buying behavior with customer satisfaction as a mediator. The results indicate that customers may have different behavior when choosing between high-priced and low-priced products, but perceived satisfaction is positively related to satisfaction with the product.

In regard to user interface satisfaction on other display machines, Henderson, et al. (2023) studied the usability and user experience of various self-service technologies, including vending machines and parking meters, and proposed design guidelines to improve user experience and satisfaction. The results of their study show that many people can vividly recall a negative user experience with at least one self-service technology.

Meanwhile, Gunay and Erbug (2015) identified commonalities among user experience studies and shared the results of a study that elicited positive user experiences around coffee vending machines and other self-service kiosks. The results of the study showed that users of self-service kiosks prefer those that provide quality, speed, security, and ease of use, and highlighted the importance of physical appearance and touch points in design.

Overall, user satisfaction is an essential factor in the design and success of vending machines and dispensers. The studies reviewed in this research suggest that while vending machines and dispensers are convenient, they often fail to meet user expectations in terms of design, function, and food options. Future research should focus on the design and improvement of vending machines and dispensers to better meet user needs and preferences and provide an overall positive user experience.

2.8 Dispensing Storage, Microcontroller, and User Interface

A dispensing machine is an automated device that provides various consumer products to customers, including valuable items such as diamonds and platinum jewelry, without the need for human assistance. Customers can access the products by inserting currency or credit into the machine and following simple steps. A control system, consisting of a main module, a payment system, a user interface system, a product extraction system, and a communication system, is installed on the machine. These elements collaborate to ensure a smooth operation and customer satisfaction (Sibanda et. al, 2020).

Storing rice in a container ensures that the rice is in excellent condition. A study titled "Changes in Physicochemical, Cooking and Sensory Characteristics of Rice Shifted from Low-temperature Storage" by An, Zhou, and Zhang (2018) found that rice stored in a container exposed to 35 degrees Celsius has increased hardness, greater water uptake, and reduced adhesiveness when cooked compared to stored rice exposed to a 20 degree Celsius environment. According to Du et al. (2022), the best and most ideal temperature and relative humidity for rice storage are 20 °C and 40% humidity. These findings suggest that temperature and environmental humidity are essential factors influencing physicochemical characteristics and nutritional changes, providing a theoretical basis for the early detection of rice mildew during storage.

The Raspberry Pi microcontroller is a quick tool for developing small sensor-based projects (Ismailov & Jo'rayev, 2022). These microcontrollers are programmed using the Raspberry Pi Integrated Development Environment. The Raspberry Pi Model B 4 GB is a powerful microcontroller board designed for advanced electronics projects, with more memory, more digital and analog input/output pins, and enhanced capabilities for complex applications.

In accessing the user interface (UI), the automated machine includes a touchscreen interface. The ability to individually control RGB pixels is the key feature of this display, resulting in significantly higher resolution than black and white displays. The module also includes a pre-installed Resistive Touch Screen, which allows the user to detect finger presses anywhere on the screen (Desai, S., Potdar, G., Jangle, S., & Desai, P. V. J., 2021).

2.9 Actuator and Sensor Components

The servo motor, which operates in a closed feedback loop, is a motorized mechanism that includes circuitry that allows positional information to be transmitted from the motor back to the control circuit within the servo motor itself (Ariawan, Santyadiputra, & Sutaya 2019). This will serve to control the flow of the rice when dispensing.

A load cell can be defined as a specialized transducer employed to generate an electrical output signal that exhibits a direct correlation with the applied force being evaluated. This transducer is an instrumental device capable of quantifying strain by converting mechanical force into electrical energy. Alongside the load cell is the HX711 IC amplifier that reads the load cell's resistance changes (Newton, 2022).

Detecting an object in the packaging system is essential to prevent dispensing rice without the plastic that will carry the rice. The IR Sensor acts as an obstacle detector by emitting infrared waves that interact with objects and cause them to reflect the signal to the sensor. The wavelength range of this sensor is 700nm to 1mm. The output of the IR Sensor varies depending on the infrared rays it receives (Dhanuja, R., Farhana, F., & Savitha, G., 2018). With an IR sensor, the machine ensures that it will only dispense rice if there is plastic present in the bin.

Maintaining the excellent quality of rice while it is stored is an essential part of the automated machine. Humidity sensors measure and monitor the amount of water in the surrounding air (Shivananju, B. N., Hoh, H. Y., Yu, W., & Bao, Q., 2019).

Notification is vital in alerting the owner of the automated machine when anomalies occur during the operation. The machine will send messages using the GSM module when container humidity and rice insufficiency are present. The GSM Module is a versatile device capable of operating on the quad-band GSM/GPRS network frequencies (850/900/1800/1900 MHz). It provides not only internet connectivity but also facilitates voice communication (when connected to a microphone and small loudspeaker) and SMS functionality (Surana, H., Jha, S. K., Agarwal, H., Goyal, P., & Dangi, P., 2021).

The addition of SMS notification and packaging bag systems represents a further innovation in dispensing machine technology. SMS notifications can provide consumers with real-time updates on their purchases, enhancing the transparency and reliability of the dispensing process (Liu et al., 2021). However, due to the Coronavirus disease (COVID-19) pandemic, consumer behaviors have changed, with consumer satisfaction influencing purchasing initiatives and decision-making online (Rangaswamy et al., 2022).

2.10 Synthesis of the Reviewed Literature and Studies

The review of literature on agricultural automation, with a focus on rice dispensing technologies, reveals both commonalities and differences among studies. A key similarity across the research is the emphasis on the transformative impact of automation in agriculture. Studies by Farooq et al. (2020), Jing et al. (2020), and Munz & Schuele (2022) consistently highlight how automation enhances productivity, efficiency, and accuracy in processes such as planting, harvesting, and dispensing. The integration of advanced technologies like robotics, AI, and IoT is a recurring theme, with many studies underscoring their role in optimizing operations.

However, the studies also reveal significant differences, particularly in terms of implementation challenges and context-specific applications. For example, Munz & Schuele (2022) discuss the high costs and skilled labor requirements of agricultural automation, while De Castro et al. (2020) focus on the challenges faced by small-scale rice retailers in the Philippines. Farooq et al. (2020) raise concerns about the environmental impact of automation, which is less emphasized in studies prioritizing technological efficiency. The types of automated machines studied also vary widely, from medicine vending machines to IoT-integrated rice dispensing systems (Yuliandoko, Panduardi, et al., 2023), highlighting the need for tailored solutions.

Unique innovations and contextual applications set certain studies apart. For instance, Yuliandoko, Panduardi, et al. (2023) introduced a rice vending machine incorporating IoT for monitoring Zakat, Infaq, and Sadaqah (ZIS) donations, merging technology with cultural practices. Additionally, Liu et al. (2021) and Surana et al. (2021) discuss the integration of SMS notifications and environmental sensors in dispensing machines, enhancing real-time communication and product quality.

In conclusion, while the literature underscores the broad applicability and benefits of automation in agriculture and dispensing technologies, it also highlights distinct challenges and innovations unique to specific contexts. These insights suggest a dynamic and evolving field, advocating for continued research and development tailored to diverse needs and conditions.

Chapter 3

METHODOLOGY

3.1 Research Design

The researchers used a combination of two research designs: (1) Quantitative Developmental research and (2) Applied Research Design.

A quantitative developmental analysis research method allowed the researchers to track the changes and progress in the performance and completion of the rice dispensing machine with SMS notification and dispensing system over time. This helps to determine if the machine is improving or if there are any other areas that needs improvement (Huizenga, 2022).

According to Kumar (2023), an applied research design systematically addresses practical challenges with the aim of developing solutions that have real-world relevance and applicability. In the case of the Multi-Grain Rice Dispensing Machine with Short Messaging System Notification, this research design ensures that the developed technology directly meets the identified needs for efficient rice storage, dispensing, and monitoring. By focusing on actionable outcomes and iterative refinement, the approach facilitates the creation of a functional prototype that can be piloted and adjusted based on stakeholder feedback. This method emphasizes the practical implications of research findings, leading to the development of a technology that is not only effective but also usable and impactful in real-world settings.

To investigate the potential capability of a Multi-Grain Rice Dispensing Machine with Short Messaging System Notification, this research design allowed them to manipulate an independent variable (Multi-Grain Rice Dispensing Machine with Short Messaging System Notification) and measure the effects of this manipulation on a dependent variable (traditional rice dispensers, dispensing machines). This helped the researchers determine whether the machine has a positive impact on its capabilities and whether it is a viable product.

The researchers also used this method to give a vision of what the machine would be like. This is done to provide information about the machine's features, such as the types of rice it can dispense, and the payment options it offers. They also provided information about the research and development factors that went into creating the machine.

3.2 Flowchart of Research Design/Process Flowchart

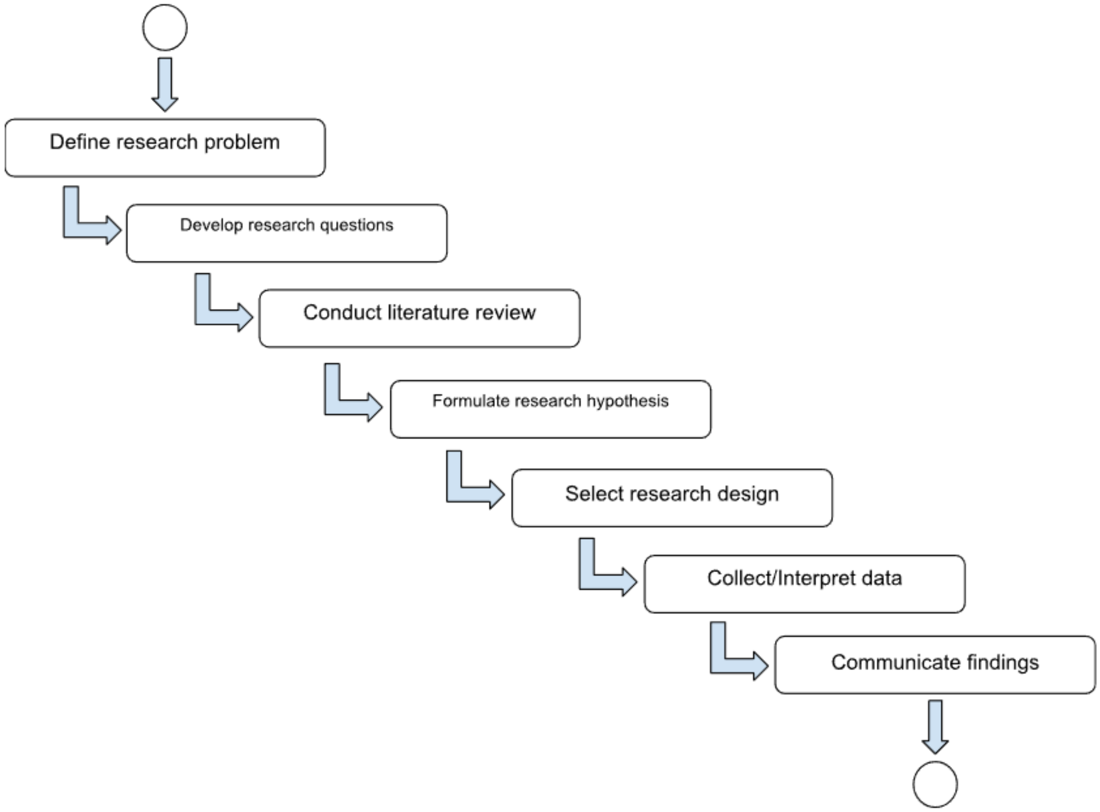


Figure 3.1. Waterfall Process Diagram

Figure 3 shows a waterfall implementation of the machine as a composite topic. To ensure the quality and validity of the research findings, it is important to follow a systematic and rigorous approach. This includes defining the research problem, developing research questions, conducting a literature review, formulating a research hypothesis, selecting a research design, collecting data, and communicating the findings. These steps are followed to present the researchers with an increase in the chances of delivering research findings that are used to inform decision-making (Khan, et al. 2016).

3.4 Research Instruments

The researchers focused on documentation analysis and design experimentation as based upon the research design of this study, and a rigorous data-gathering procedure ensures the results of the study to be accurate and reliable. Following the general data-gathering procedure, the researchers implemented:

Survey Questionnaire. This serves as a primary source of data, the questions are relayed to the respondents via both Google Forms and physical questionnaire sheets as mediums. Data gathered from the collected information from the individuals are transmutted and analyzed using Microsoft Excel and Word.

Observation. Through observation and objective analysis of raw results from several testing phases of the machine, the researchers will be able to recognize specific areas to be added, improved, or removed wether it be physical components of the machine or software-related such as code and script.

Online research. The researchers utilized reliable online searching platforms mainly academic papers to gather information with regards to rice, dispensing machines, and the like. This research instrument is a principal source of related literature gathered by the researchers to strengthen the structure of the study.

3.4 Statistical Treatment

In the study on Multi-Grain Rice Vending Machine with Short Messaging System Notification, the researchers used the following statistical methods to analyze data:

- **Descriptive statistics:** This involves summarizing and presenting the collected data using measures such as mean, median, mode, standard deviation, and range. Descriptive statistics is used to describe the characteristics of the variables involved in the study, such as the amount of rice dispensed, time it takes to tare, etc.

  Mean: The mean, or average, is calculated by summing all the data points and dividing by the number of observations:

$$Mean = \frac{Sum\ of\ Observations}{Total\ Number\ of\ Observations}\ ;\ \bar{x} = \frac{\sum x}{N}$$

Wherein:

$\sum$ = *denotes the summation or addition*

N = *denotes the total number of given observations*

X = *denotes the given observations*

  Median: The median represents the middle value in a dataset when the data points are arranged in ascending or descending order. For datasets with an odd number of observations, the median is the middle value. For even-numbered datasets, the median is the average of the two central values. The formula for the median is:

If 'n' is odd: $$Median = \left(\frac{n+1}{2}\right)^{th}\ term$$

If ‘n’ is even: $Median = \frac{\left(\frac{n}{2}\right)^{th} term + \left(\frac{n}{2}+1\right)^{th} term}{2}$

Mode: The mode is the value that appears most frequently in a dataset. Unlike the mean and median, the mode can have more than one value if multiple values have the same highest frequency. The mode is determined by identifying the value(s) that occur most frequently in the dataset.

- **Mean Absolute Percentage Error (MAPE):** This is a statistical measure that determines how accurate the predictions or forecasts are in relation to the actual values. It expresses the error as a percentage, making it easier to interpret and communicate the model's accuracy. It works best if there are no extremes to the data (and no zeros).

$$MAPE = \left(\frac{1}{N}\sum ABS\ Error\ per\ Actual\right) \times 100$$

(MAPE) Wherein:

N = *total number of observations or data points*

$\sum$ = *summation across all observations*

$ABS\ Error\ per\ Actual\ \left(\frac{Error}{Actual}\right)$ = *absolute error as proportion of actual value*

100 = *converts mean error into percentage*

- **Weighted Mean:** This is a statistical measure that provides a more accurate representation of the central tendency of a data set when different values contribute unequally to the overall average. The weighted mean considers the varying importance or frequency of each value, making it a more reliable metric in contexts where such differences matter.

- The general weighted mean equation calculates a Likert scale where 25 respondents rate a survey questionnaire.

$$\overline{X}_\omega = \frac{\sum_{i=1}^{n} \omega_i x_i}{\sum_{i=1}^{n} \omega_i}$$

(Multiple Respondents) Wherein:

$\overline{\boldsymbol{X}}_{\boldsymbol{\omega}}$ *= is the weighted mean*

$\boldsymbol{X_i}$ *= represents the frequency of input of the users*

$\boldsymbol{\omega_i}$ *= represents the rating per category*

$\boldsymbol{n}$ *= is the number of data points*

- The weighted overall score equation involves only one person (civil engineer) in rating a Likert scale survey questionnaire.

$$Overall\ Score = \frac{\sum_{i=1}^{n} Ri}{n}$$

(Single Respondent) Wherein:

$\sum_{i=1}^{n} Ri$ *= sum of all the individual ratings*

Ri *= represents each individual rating*

n *= total number of ratings or questions*

3.5 Design Project Flow

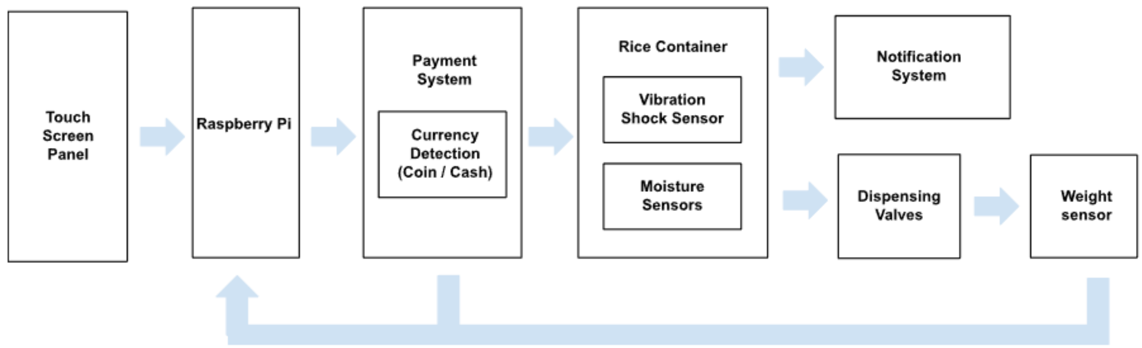


Figure 3.2. Block Diagram (Components)

Figure 4 depicts the flow and the relationship of components within the system. The power source provides power to the power supply, which powers various components such as the payment system and Raspberry Pi. The payment system with currency detection communicates with the Raspberry Pi. The touch screen panel acts as the input and output device and communicates with the Raspberry Pi.

The rice container includes features such as a vibration shock sensor, temperature/ humidity sensors, and sealed containers which communicate with the Raspberry Pi. The notification system collects information from the rice containers and sends it as a text message to a dedicated cellular number. Dispensing valves are controlled by the Raspberry Pi to dispense and control the flow of rice according to the weight ordered. A dedicated weight sensor will measure the dispensed rice to confirm that the dispensed rice is equal to the weight ordered.

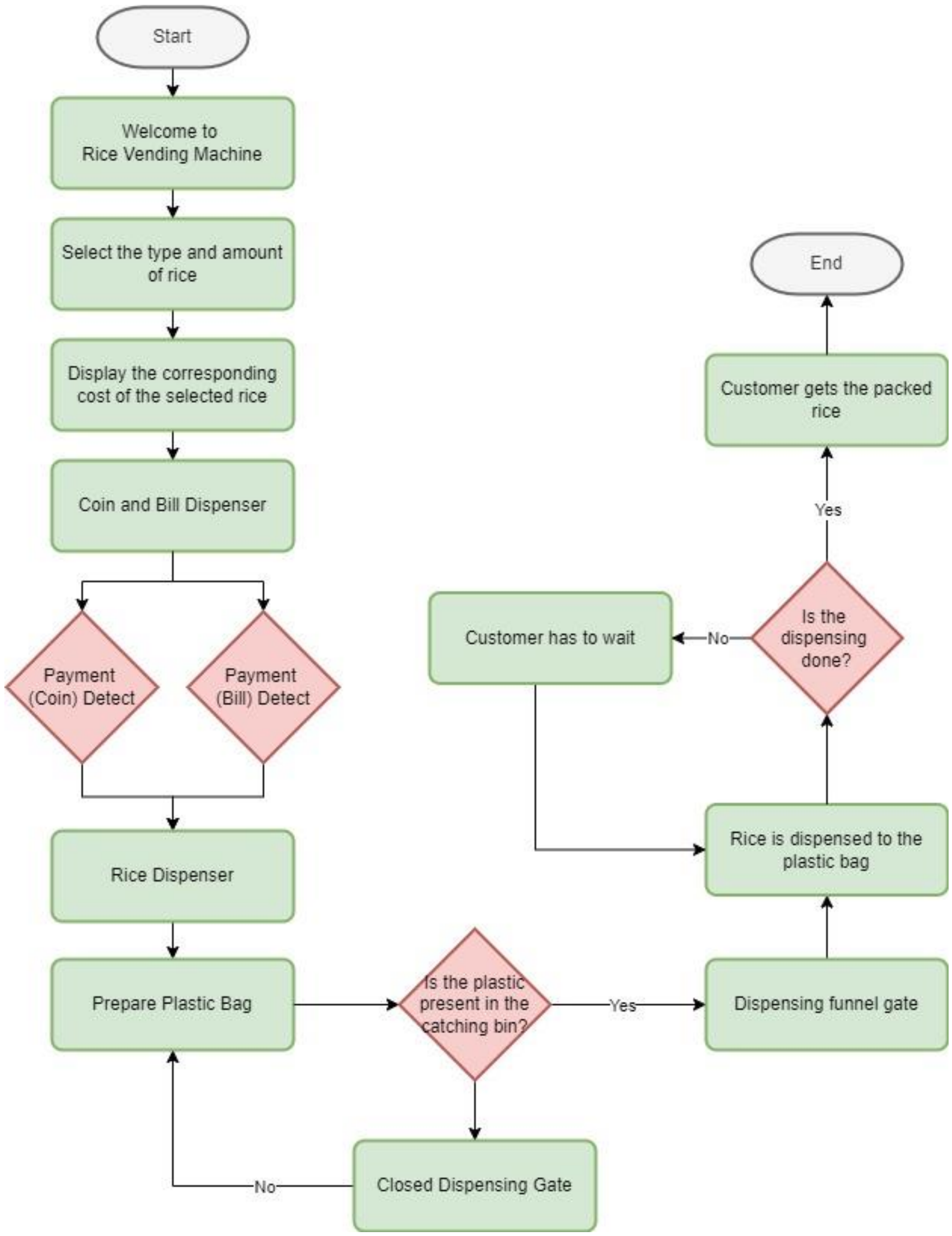


Figure 3.3. System Flowchart

Figure 5 above shows an illustration in regards to the system flowchart of the automated rice dispensing machine and the process of connecting individual sections into one, such as the Payment and SMS System implementation.

3.6 Components of the Proposed System

The Multi-Grain Rice Dispensing Machine with Short Messaging System Notification integrates multiple hardware and software components to ensure its functionality and user convenience. Below is a detailed description of each primary component:

A. Cabinet - The machine features a cabinet designed to house all the mechanical and electronic parts securely. The cabinet is constructed from materials like stainless steel, aluminum casing, acrylic glass, and composite panels. These materials enable individuals to visually inspect rice levels and types without opening the container while ensuring the structure's durability and security.

B. Dispensing Mechanism - The dispensing mechanism consists of servo motors controlling the release of different rice types. The number of servo motors corresponds to the variety of rice the machine can dispense. Each servo motor is calibrated to ensure precise measurement and dispensing, aided by a load cell and the HX711 Amplifier, a 24-bit analog-to-digital converter for accurate weight measurement.

C. Control System - A Raspberry Pi 4 Model B 2GB serves as the primary microcontroller, managing the machine's operations. The control system ensures synchronization between user commands, dispensing actions, and notifications.

D. Touchscreen Interface - A 7-inch touchscreen LCD serves as the user interface, displaying information and allowing user input. It enables users to select the type and quantity of rice and provides real-time feedback during the transaction.

E. Payment System - The payment system comprises coin and bill acceptors configured to accept specific denominations of Philippine peso coins and bills. This system ensures secure and straightforward transactions, which rejects invalid or excessive denominations.

F. Sensor Systems - DHT22 sensor serves to monitor temperature and humidity within the rice containers to monitor optimal storage conditions. While the Infrared (IR) sensor detects the presence of packaging and ensures that dispensed rice is correctly collected.

G. GSM Module - The GSM module sends SMS notifications to the seller for information gathering and enhancing user convenience. A text message is automatically implemented to: Humidity sensor - extreme increase or decrease of temperature; and Vibration sensor - detects shocks or physical tampering on the machine which then sends a notification warning message to the seller and the developers. Reaching the threshold value of the rice containers also notifies the seller that the containers need to be refilled.

H. Power Supply and Relays - A suitable power supply powers the Raspberry Pi and other components, while relays control the power distribution to motors and sensors.

I. Additional Electronic Components - Various electronic components, such as resistors, capacitors, diodes, and wiring, are essential for circuit connections and system functionality.

3.7 Multiple Constraints

Design constraints refer to the limitations or requirements that a design solution must meet in order to achieve success for the system. This study provides a comprehensive examination of the technical constraints that occurred during the development of the machine. To conceptualize the three potential designs, three distinct factors—Costing, Algorithm, and Development—were evenly allocated, each accounting for one hundred percent. These considerations were utilized to assess which of the following designs would be the most suitable or appropriate for the study. The Cell-Index Factor for the different designs was calculated using the table below to set priority based on different variables. The Total Score

Function functions as a comprehensive indication for evaluating the acceptability of the design in the study.

Table 3.1. Scale System and Total Acceptability Score

Scale for Cell-Index Factor	
5	91-100
4	81-90
3	71-80
2	51-70
1	0-50

Total Score Function	
4.1 - 5.0	Highly Acceptable
4	Acceptable
3	Neutral
2	Not Acceptable
1	Reject

Table 1 serves as the standard by which the design's acceptability is determined. The project prioritized a careful balance between cost-effectiveness and efficiency, leading to the development of three distinct designs, each tailored for specific functions. This approach underscores the recognition that making a substantial investment is crucial to achieving optimal performance and overall effectiveness.

Design A placed a strong emphasis on efficiency while carefully managing costs. Components were selected with precision to maintain a low total system cost while ensuring they meet the required performance standards. This approach involves leveraging cost-effective commercially available hardware components, open-source software tools, and optimization strategies to attain desired outcomes without significantly inflating the budget.

Design B focused on cost reduction, even at the expense of some efficiency. This may involve using basic or outdated hardware, streamlined algorithms, or reduced processing capability. The goal is to explore the feasibility of developing a functional electronic components sorting system with visual flaw detection while adhering to stringent financial constraints and acknowledging the trade-offs between cost and efficiency.

Design C strived to showcase a cutting-edge system that prioritizes efficacy over cost. The system incorporates top-of-the-line hardware components, sophisticated image processing algorithms, and state-of-the-art technology to achieve exceptional accuracy, speed, and dependability. While the cost may be higher, the primary emphasis is on pushing the boundaries of technological feasibility within the context of the project.

3.8 Multiple Constraint Data

A. Design Constraints

Table 3.2. Design A (Costing)

Design A		Costing (25%)			
Components	Descriptions	Weight	Cell-Index Factor	Score	Score Function
		a	*b*	*c=a*b*	*d= c*25%*
Microcontroller	Raspberry Pi 4	0.2	4	0.8	0.2
Programming Language	Python	0.01	5	0.05	0.0125
GSM Module	SIM900A	0.025	3	0.075	0.01875
IR Sensor	ARD IR Proximity Sensor	0.01	3	0.03	0.0075
Servo Motor	TowerPro MG958	0.075	3	0.225	0.05625
Bill Acceptor	TB74 Bill acceptor	0.125	4	0.5	0.125
Coin Acceptor	Allan Universal Coin Acceptor	0.05	5	0.25	0.0625
Temperature and Humidity Sensor	DHT22 Temperature and Humidity Sensor	0.013	5	0.065	0.01625
Weight Sensor	Load Sensor and HX711 Amplifier	0.05	4	0.2	0.05
Display	Waveshare 7 inch Touch Capacitive Screen	0.125	4	0.5	0.125
Structure	Stainless Steel + Composite Panel	0.317	3	0.951	0.23775
		1			0.9115

Table 3.3. Design A (Quality and Development)

Quality (25%)				Development (50%)			
Weight	**Cell-Index Factor**	**Score**	**Score Function**	**Weight**	**Cell-Index Factor**	**Score**	**Score Function**
a	***b***	***c=a*b***	***d= c*25%***	***a***	***b***	***c=a*b***	***d= c*50%***
0.3	5	1.5	0.375	0.35	5	1.75	0.875
0.025	5	0.125	0.03125	0.25	5	1.25	0.625
0.05	3	0.15	0.0375	0.05	4	0.2	0.1
0.03	4	0.12	0.03	0.05	3	0.15	0.075
0.05	4	0.2	0.05	0.05	4	0.2	0.1
0.1	5	0.5	0.125	0.025	5	0.125	0.0625
0.1	5	0.5	0.125	0.025	4	0.1	0.05
0.06	3	0.18	0.045	0.05	4	0.2	0.1
0.06	3	0.18	0.045	0.05	5	0.25	0.125
0.1	4	0.4	0.1	0.075	4	0.3	0.15
0.125	4	0.5	0.125	0.025	5	0.125	0.0625
1			1.08875	1			2.325

Table 3.4. Design B (Costing)

Design B		Costing (25%)			
Components	Descriptions	Weight	Cell-Index Factor	Score	Score Function
		a	*b*	*c=a*b*	*d= c*25%*
Microcontroller	Orange Pi Zero 2	0.2	4	0.8	0.2
Programming Language	Python	0.01	5	0.05	0.0125
GSM Module	SIM800F	0.025	3	0.075	0.01875
IR Sensor	Sharp GP2Y0A21YK0F	0.01	4	0.04	0.01
Servo Motor	MG996R	0.075	5	0.375	0.09375
Bill Acceptor	MEI Cashflow SC	0.125	3	0.375	0.09375
Coin Acceptor	FAS International CoinMech 5	0.05	4	0.2	0.05
Temperature and Humidity Sensor	HDC 1080	0.013	5	0.065	0.01625
Weight Sensor	Phidgets Force Sensor	0.05	4	0.2	0.05
Display	Adafruit PiTFT 3.5”	0.125	5	0.625	0.15625
Structure	Woodl + Wood	0.317	4	1.268	0.317
		1			1.01825

Table 3.5. Design B (Quality and Development)

Quality (25%)				Development (50%)			
Weight	Cell-Index Factor	Score	Score Function	Weight	Cell-Index Factor	Score	Score Function
a	*b*	*c=a*b*	*d= c*25%*	*a*	*b*	*c=a*b*	*d= c*50%*
0.3	3	0.9	0.225	0.35	3	1.05	0.525
0.025	5	0.125	0.03125	0.25	5	1.25	0.625
0.05	2	0.1	0.025	0.05	2	0.1	0.05
0.03	2	0.06	0.015	0.05	3	0.15	0.075
0.05	2	0.1	0.025	0.05	3	0.15	0.075
0.1	1	0.1	0.025	0.025	2	0.05	0.025
0.1	2	0.2	0.05	0.025	3	0.075	0.0375
0.06	2	0.12	0.03	0.05	2	0.1	0.05
0.06	0	0	0	0.05	2	0.1	0.05
0.1	2	0.2	0.05	0.075	5	0.375	0.1875
0.125	2	0.25	0.0625	0.025	5	0.125	0.0625
1			0.53875	1			1.7625

Table 3.6. Design C (Costing)

Design C		Costing (25%)			
Components	Descriptions	Weight	Cell-Index Factor	Score	Score Function
		a	*b*	*c=a*b*	*d= c*25%*
Microcontroller	Odroid N2+	0.2	2	0.4	0.1
Programming Language	Python	0.01	5	0.05	0.0125
GSM Module	Quectel L765S	0.025	3	0.075	0.01875
IR Sensor	VL6180X Time-of-Flight Sensor	0.01	2	0.02	0.005
Servo Motor	Futaba S3114	0.075	2	0.15	0.0375
Bill Acceptor	CashCode SM100	0.125	2	0.25	0.0625
Coin Acceptor	CoinCo Genesis G-10	0.05	2	0.1	0.025
Temperature and Humidity Sensor	SHT31-D	0.013	3	0.039	0.00975
Weight Sensor	Honeywall FelxiForce Sensors	0.05	2	0.1	0.025
Display	Raspberry Pi Official Touchscreen	0.125	2	0.25	0.0625
Structure	Aluminum + Glass	0.317	1	0.317	0.07925
		1			0.43775

Table 3.7. Design C (Quality and Development)

Quality (25%)				Development (50%)			
Weight	**Cell-Index Factor**	**Score**	**Score Function**	**Weight**	**Cell-Index Factor**	**Score**	**Score Function**
a	***b***	***c=a*b***	***d= c*25%***	***a***	***b***	***c=a*b***	***d= c*50%***
0.3	5	1.5	0.375	0.35	3	1.05	0.525
0.025	5	0.125	0.03125	0.25	5	1.25	0.625
0.05	4	0.2	0.05	0.05	1	0.05	0.025
0.03	5	0.15	0.0375	0.05	2	0.1	0.05
0.05	4	0.2	0.05	0.05	4	0.2	0.1
0.1	4	0.4	0.1	0.025	3	0.075	0.0375
0.1	5	0.5	0.125	0.025	2	0.05	0.025
0.06	4	0.24	0.06	0.05	4	0.2	0.1
0.06	5	0.3	0.075	0.05	3	0.15	0.075
0.1	4	0.4	0.1	0.075	5	0.375	0.1875
0.125	4	0.5	0.125	0.025	2	0.05	0.025
1			1.12875	1			1.775

B. Level of Acceptance

Table 3.8. Total Level of Acceptance towards Designs A, B, and C

Level of Acceptance	Design A	**4.32525**
	Design B	3.3195
	Design C	3.3415

Table 8 provides the Total Score Function, displays the ranges that decide whether the design is highly acceptable or rejected. The Cost, Algorithm, and Development of Design B are categorized as neutral. Meanwhile, Design C is classified as acceptable. Design A is considered to be highly acceptable. Hence, Design A is the most suitable or compatible option for the Design Project.

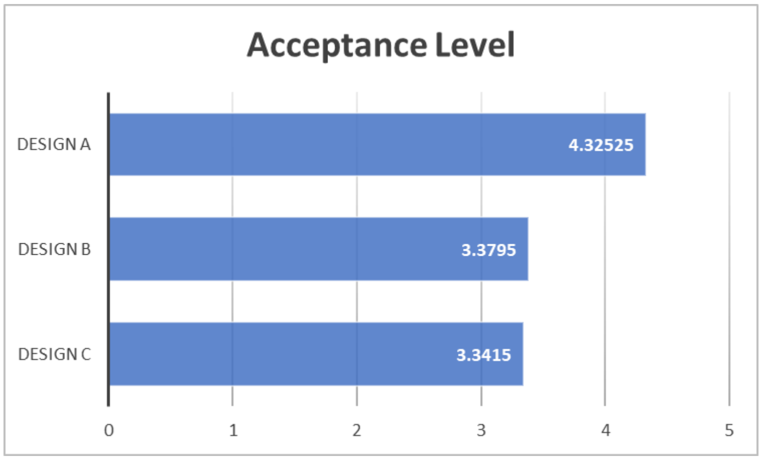


Figure 3.4. Total Level of Acceptance towards Designs A, B, and C

Figure 6 shows the level of acceptance for designs A, B, and C. It is determined by the total score function, which considers different factors such as cost, algorithm, and development.

C. Multiple Constraint Comparison

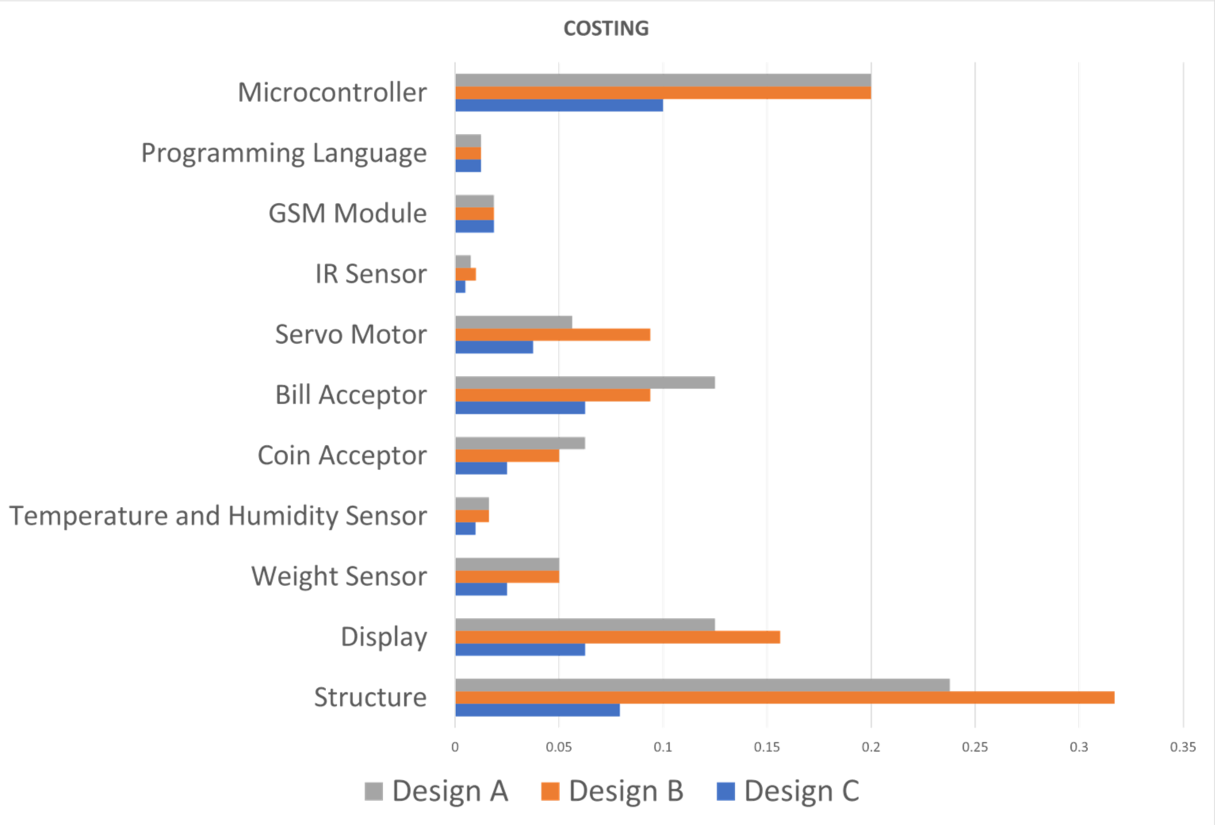


Figure 3.5. Cost Comparison of Designs A, B, and C

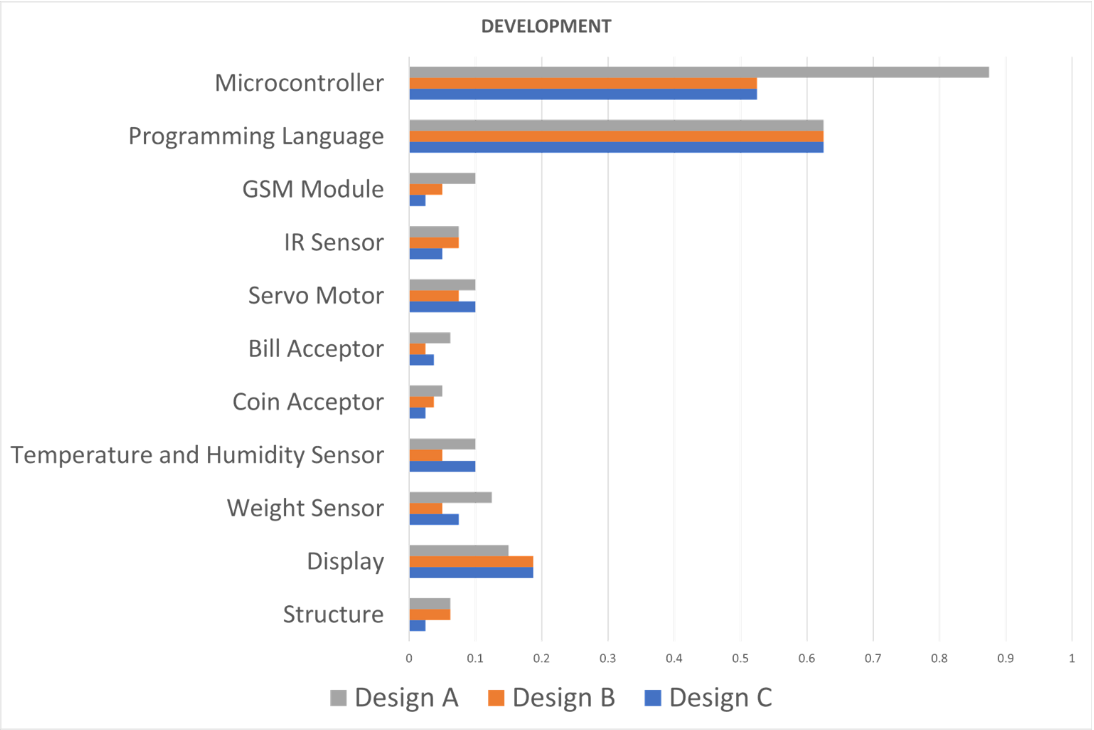


Figure 3.6. Quality Comparison of Designs A, B, and C

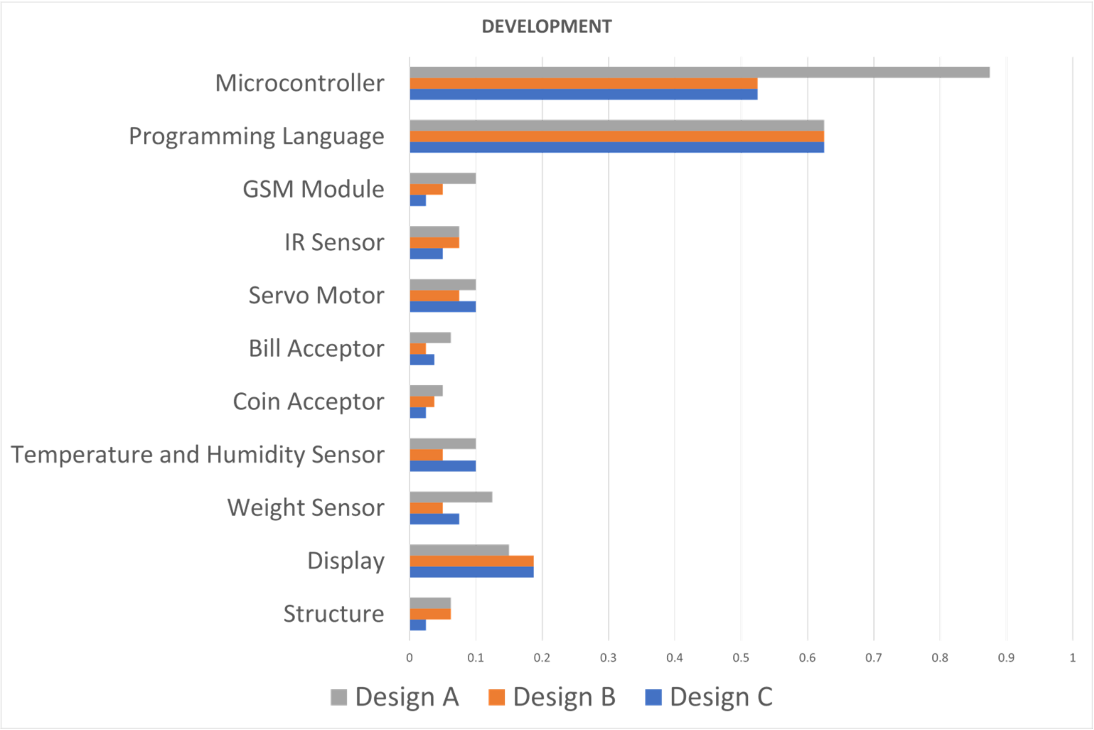


Figure 3.7. Development Comparison of Designs A, B, and C

D. Total Score of Factors of Designs A, B, and C.

Table 3.9. Total Score of Factors on Costing, Quality, and Development for Designs

Total Score of Factors			
	Costing (25%)	**Quality (25%)**	**Development (50%)**
Design A	0.9115	1.08875	2.325
Design B	1.01825	0.59875	1.7625
Design C	0.43775	1.12875	1.775

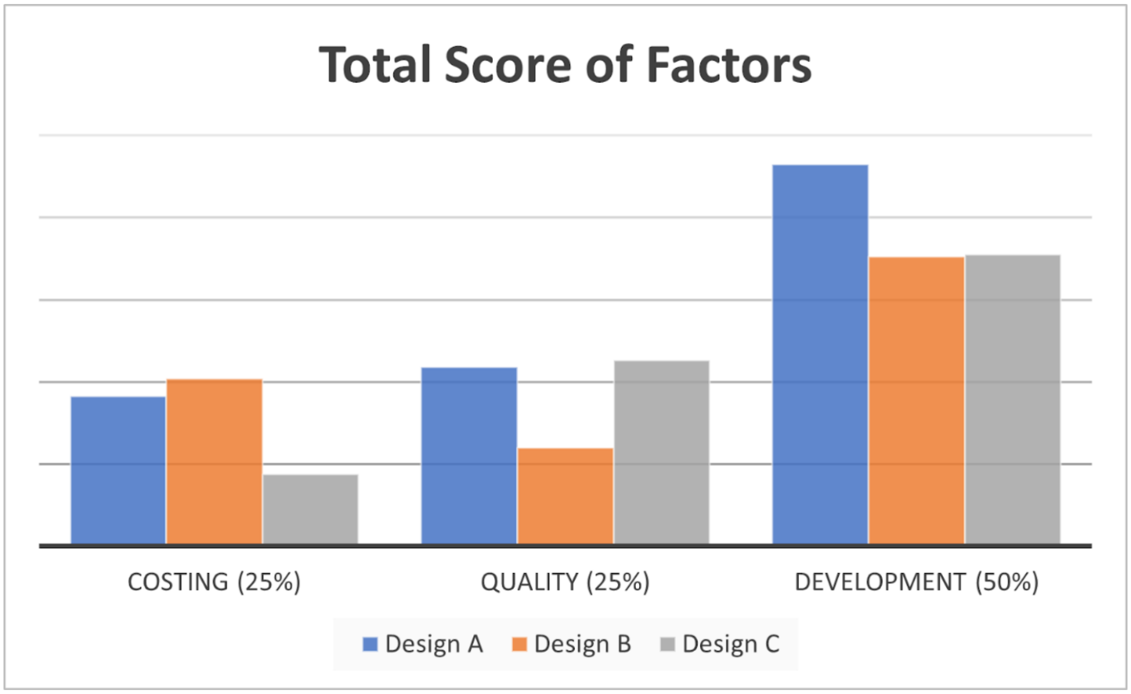


Figure 3.8. Total Score of Factors on Costing, Quality, and Development for Design

E. Narrative Reports

Design A

Design A stands out as the most reliable option among the three, leveraging a set of high-quality components to ensure optimal performance. The microcontroller of choice is the Raspberry Pi 4, offering robust computing power and versatility for software development in Python. The Aideepen IR Proximity Sensor enhances precision in object detection, a critical aspect for accurate dispensing operations. Other key components include the TowerPro MG958 Servo Motor, TB74 Bill Acceptor, and Allan Universal Coin Acceptor, all contributing to the overall reliability of the system. The inclusion of the DHT22 Temperature and Humidity Sensor, Load Sensor with HX711 Amplifier, and Waveshare 7-inch Touch Capacitive Screen further enhances the functionality and user interface of the dispensing machine. The use of Stainless Steel and Composite Panel for the structure ensures durability. While Design A may come at a slightly higher cost, its superior reliability makes it the recommended choice for a Multi-Grain Rice Dispensing Machine with a Short Messaging System Notification.

Design B

Design B offers a cost-effective alternative, albeit with certain compromises in terms of performance and reliability. The microcontroller, an Orange Pi Zero 2, is selected for its compact size and affordability, making it suitable for basic applications; however, it lacks the power and I/O options of the Raspberry Pi 4. While Python remains the chosen programming language for accessibility, the SIM800F GSM Module is more budget-friendly but supports only GSM/GPRS, limiting its functionality compared to SIM900A. The use of the Sharp GP2Y0A21YK0F IR Sensor and MG996R Servo Motor provides a balance between cost and basic functionality. The HDC 1080 Temperature and Humidity Sensor, Phidgets Force Sensor for weight measurement, and Adafruit PiTFT 3.5” Display contribute to satisfactory performance within a more constrained budget. However, the reliance on a Wood structure may compromise long-term durability compared to the Stainless Steel and Composite Panel structure of Design A. Design B is recommended for scenarios where budget constraints take precedence over advanced features and enhanced reliability.

Design C

Design C emerges as a high-performance solution, incorporating advanced features and capabilities. The microcontroller of choice, the Odroid N2+, stands out with a powerful CPU and GPU, surpassing the performance of the Raspberry Pi 4. Python remains the programming language, ensuring a familiar development environment. The Quectel L765S GSM Module provides advanced connectivity options with LTE Cat M1 and NB-IoT support, although it comes at a higher cost and requires specific network infrastructure. The use of the VL6180X Time-of-Flight Sensor for IR detection ensures greater accuracy, particularly in varying lighting conditions, albeit at a higher cost. The Futaba S3114 Servo Motor offers increased power and speed, enhancing the overall efficiency of the system. Advanced features in the CashCode SM100 Bill Acceptor and CoinCo Genesis G-10 Coin Acceptor, such as multi-currency support and high-security counterfeit detection, contribute to the reliability of the system. The SHT31-D Temperature and Humidity Sensor and Honeywell FlexiForce Sensors for weight measurement provide high accuracy and stability. The Raspberry Pi Official Touchscreen Display and an Aluminum + Glass structure add a touch of sophistication but come at an elevated cost. While Design C excels in terms of performance and features, its higher overall cost may be a consideration, making it a suitable choice for scenarios where budget constraints are less stringent, and advanced capabilities are paramount.

Best Design

Design A stands out as the superior choice among the three options. It achieves a commendable balance between reliability, performance, and cost-effectiveness. The use of the Raspberry Pi 4 as the microcontroller ensures robust computing power and versatile software development capabilities in Python. Key components, including the Aideepen IR Proximity Sensor, TowerPro MG958 Servo Motor, TB74 Bill Acceptor, and Allan Universal Coin Acceptor, collectively contribute to a high level of dependability in dispensing operations.

Furthermore, the incorporation of the DHT22 Temperature and Humidity Sensor, Load Sensor with HX711 Amplifier, and Waveshare 7-inch Touch Capacitive Screen enhances functionality and user interface without compromising reliability. The Stainless Steel and Composite Panel structure adds durability to the design. Despite a slightly higher cost, Design A's superior reliability, comprehensive feature set, and balanced cost-performance ratio make it the optimal choice for a Multi-Grain Rice Dispensing Machine, particularly in scenarios prioritizing reliability and long-term performance.

Chapter 4

RESULTS & DISCUSSION

4.1 Stages of Machine Development

The implementation of the Multi-Grain Rice Dispensing Machine involves several stages:

I. Design and Fabrication

Designing the cabinet and dispensing mechanism to meet specifications for durability and efficiency. The dispensing mechanism is tailored to handle multiple rice types and ensure accurate dispensing. The skeletal framework of the machine is designed to demonstrate the development stages. It includes:

Cabinet Structure: Housing for all components and user interface.

Dispensing Mechanism: Controlled by servo motors for accurate dispensing.

Control System: Raspberry Pi interfaced with sensors and motors.

User Interface: 7-inch touchscreen LCD for user interaction.

Payment System: Coin and bill acceptors for secure transactions.

Notification System: GSM module for sending SMS notifications.

Power Supply and Relays: Ensuring stable power distribution.

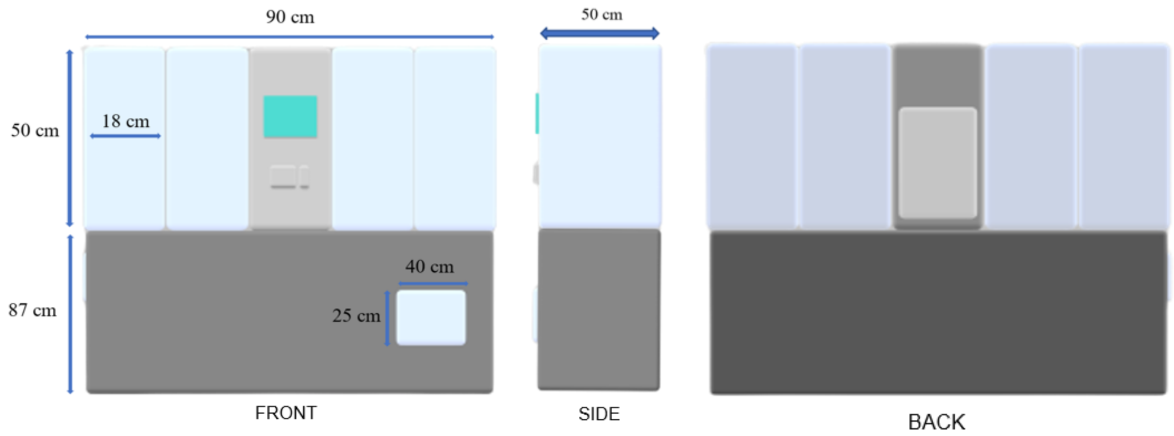


Figure 4.1. Early Model Design Proposal:

(left) Front-View; (middle) Side-View; (right) Back-View

Figure 11 presents the very first 3D model design proposal of the Multi-Grain Rice Dispensing Machine. The model includes detailed measurements, providing a comprehensive view of the machine's dimensions and spatial layout. The design features the proposed locations for key components such as the rice containers, dispensing mechanisms, touchscreen interface, and payment system. This initial model served as a blueprint for subsequent design refinements and iterations.

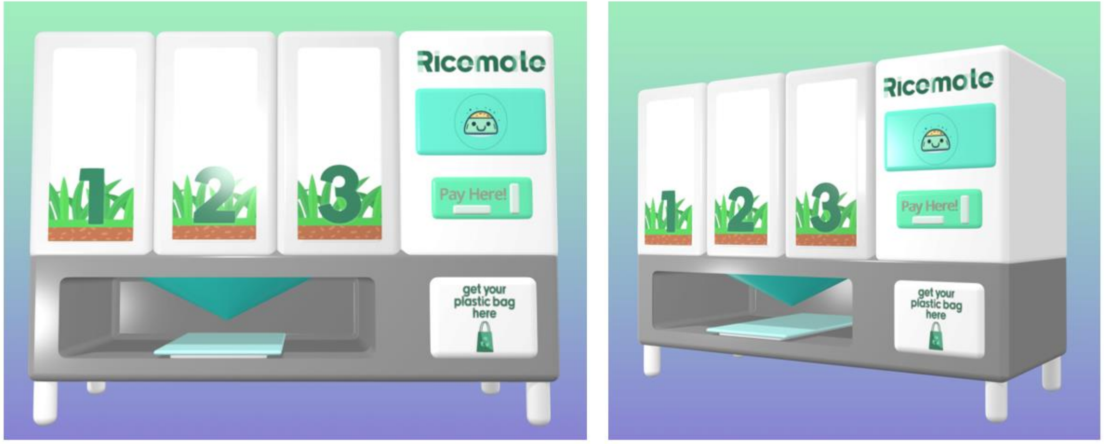


Figure 4.2. Final Model Design:

(left) Front Capture; (right) Side Capture

Figure 12 illustrates the final 3D model design of the Multi-Grain Rice Dispensing Machine. Significant improvements and refinements from the initial proposal are evident, including optimized component placement, enhanced aesthetic appeal, and improved ergonomic design. The final model incorporates user feedback and technical adjustments to ensure functionality and user convenience. It provides a precise and polished visualization of the machine's intended appearance and configuration.

II. Programming and Integration

Programming the control system using Raspberry Pi OS and Python's framework *PyQT* to manage motor operations, sensor readings, user inputs, and information display on the touchscreen. The payment system is integrated for secure transactions, and the GSM module is configured for SMS notifications.

III. Testing and Calibration

The system undergoes rigorous testing to ensure all components function correctly. Calibration of the servo motors and load cell is performed to guarantee accurate rice dispensing.

Figure 4.3. Machine Structure During Development Process:

(left) General View; (right) Containers and Dispenser

Figure 13 captures the Multi-Grain Rice Dispensing Machine during the mid-development process. It showcases the partially assembled machine, highlighting the integration of major components such as the rice containers, dispensing mechanisms, and the mounting framework for the touchscreen and payment systems. This image provides insight into the assembly stages and the structural framework that supports the machine's operational components.

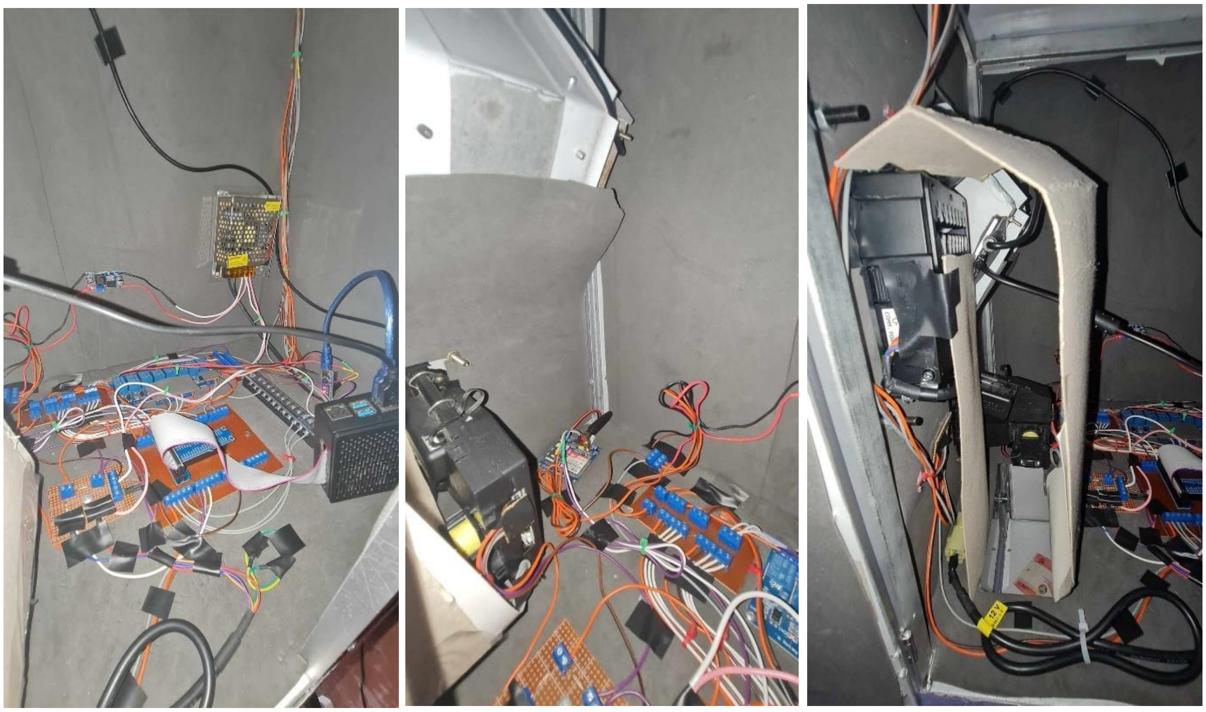


Figure 4.4. Interior Material and Circuitry Connections:

(left) General View; (middle) Top View; (right) Bill Acceptor

Figure 14 shows a look inside the machine's interior, focusing on the circuitry connections and the arrangement of internal materials. Revealing the wiring, circuit boards, sensors, and other electronic components essential for the machine's operation. The image illustrates how the various hardware elements are interconnected, showcasing the machine's assembly.

IV. Deployment Considerations

The machine is designed to store a maximum of 12 kilograms of rice per container and dispense up to 3 kilograms per transaction. It must be placed in areas with strong cellular signals for reliable GSM module operation and near its owners for maintenance and refilling.

Figure 4.5. Final Physical Design and Structure of Machine:

(left) Front Capture; (right) Side Capture

Figure 15 displays the final physical design and structure of the Multi-Grain Rice Dispensing Machine. It presents the completed machine in its operational form, ready for deployment. The image highlights the sleek exterior, user-friendly touchscreen interface, and the accessible payment system. This final design represents the culmination of the design, development, and testing processes, embodying the project's goals of functionality, efficiency, and user satisfaction.

4.2 Significant Differences Between Existing System And Proposed Study

This section discusses the significant difference between the existing system and proposed study in terms of – Structural Design, and System Integrations.

i. Structural Design of Existing Systems

This section compares the structural designs of existing rice dispensing machines with the proposed study. It examines their distinct approaches and emphasizes the implications for durability, cost-effectiveness, maintenance, and overall performance in rice dispensing applications.

The materials used in constructing the machine from *Precision Rice Vending Machine by Using Multiple Load Cell and IoT Based* are plate iron to handle up to 200 kilograms of weight and stainless steel for its rice holding tube to prevent rust. Plate iron is suited for heavy-duty structures where support and stability are critical requirements, as it is much thicker than metal sheets (Jon, 2023). Stainless steel is an ideal component for a dispensing machine due to its exceptional corrosion resistance and low maintenance (Powers, 2024). The Precision Rice Vending Machine utilizes plate iron for its structural stability and stainless steel for its corrosion resistance and low maintenance, ensuring durability and reliability in heavy-duty use.

In *the Design of an ARM-based Automatic Rice-Selling Machine for Cafeterias*, the rice storage is made of austenitic stainless steel (0Cr19Ni9) for its upper part and stainless steel (0Cr25Ni20) for its lower part. The austenitic stainless steel is known for its toughness, weldability, and corrosion resistance. Austenitic stainless steel's corrosion resistance is evident in oxidizing and reducing environments (Civmats Co.). The study also utilizes 0Cr25Ni20 stainless steel, which is noted for its excellent forming and welding properties and its robust corrosion and oxidation resistance. It also maintains toughness at low temperatures, is easy to fabricate, and has an attractive appearance. (Bebon International, 2011). In designing an ARM-based automatic rice-selling machine, using the stated materials highlights a strategic choice to leverage the material's strengths in toughness, weldability, and corrosion resistance, ensuring durability and aesthetic appeal in various operational environments.

ii. Structural Design of Proposed Study

The proposed machine, Ricemate, is assembled using galvanized steel as its frame, aluminum composite panels and aluminum stucco sheets as its cover and partitioner, and acrylic glass for observing the rice grains. Galvanized steel provides extended durability and corrosion protection due to its zinc-iron coating, resulting in a lifespan of over 50 years in average environments and over 20 years even in severe water exposure conditions. It is also inexpensive, ready to use upon delivery, and requires no maintenance, reducing overall costs and increasing the machine's reliability (Marshall, 2019).

Aluminum composite panels are utilized in the development due to their versatility. They offer durability, corrosion resistance, non-combustible qualities, lightweight construction, and high strength, making them suitable for constructing the prototype and ensuring its longevity and safety (The benefits & uses of ACP - Mulford Plastics NZ). Aluminum stucco sheets are well-suited for constructing the machine prototype due to their corrosion resistance, better thermal insulation properties, long service life with no maintenance, and easy fabrication, ensuring durability upon deployment (Aluminum-Stucco-Sheet, 2020).

Acrylic glass is an excellent choice for the machine's window due to its high transparency. It allows 92% of white light to pass through, ensuring clear visibility of the rice.

Acrylic glass resists temperature changes and impact, making it durable and less likely to warp or break under varying conditions. Additionally, acrylic glass's chemical resistance and ease of sanitation make it suitable for maintaining hygiene in rice dispensing applications.

Ricemate's construction leverages the strengths of galvanized steel, aluminum composite panels, aluminum stucco sheets, and acrylic glass. These materials collectively ensure the machine's durability, reliability, safety, and hygiene, making it a robust and efficient solution for rice dispensing.

iii. Difference of Structural Designs of Existing and Proposed Systems

The significant difference in the structural designs between the existing rice dispensing machines and the proposed Ricemate lies in the choice and combination of materials. While existing machines primarily rely on plate iron and stainless steel for durability and corrosion resistance, Ricemate employs a more diverse range of materials, including galvanized steel for the frame, aluminum composite panels, aluminum stucco sheets for covering and partitioning, and acrylic glass for observation. This strategic selection enhances Ricemate's durability, cost-effectiveness, ease of maintenance, and overall performance, making it a more efficient and reliable solution for rice dispensing.

iv. Material Quality and Structure Survey

In line with ISO standards and standard engineering practices for assessing material quality and design structure, the researchers devised a comprehensive survey questionnaire to evaluate the structural integrity and functionality of the machine. The survey was written to cover critical aspects of the machine's construction and performance. A licensed professional civil engineer “**Danilo Ramos Cabatan**” was invited to provide expert ratings and feedback on various criteria, ensuring an objective assessment based on industry standards.

The survey consisted of the following sections, each with specific criteria evaluated on a **scale of 1 to 5 (1 for poor, 5 for excellent)** (Hong et al, 2014). **Engineer Cabatan** was provided with detailed information about the materials used, including a list of materials, photographs, and videos of the machine.

I. Frame Construction: The use of galvanized steel was deemed appropriate, and the frame design was highly rated for its structural integrity. The minor deduction in fabrication quality suggests a need for further scrutiny of welding and connection techniques.

II. Cladding Materials: The suitability of aluminum composite panels and stucco sheets was affirmed, with secure attachment methods and significant structural contributions. Testing for environmental suitability received a slightly lower rating, indicating a need for comprehensive material testing.

III. Overall Functionality: The machine excelled in accessibility, user interaction, and operational safety, reflecting a user-centric design approach.

This detailed evaluation provided by a licensed civil engineer, based on a structured survey adhering to ISO and engineering standards, underscores the robustness and effectiveness of the machine while identifying areas for further refinement.

Table 4.1. Total Score for Structural Integrity and Functionality

I. Frame Construction (Galvanized Steel)		
Criterion	**Description**	**Rating (1-5)**
a. Material Selection	Is galvanized steel appropriate for the anticipated load and environment?	**4**
b. Frame Design	Is the frame design structurally sound for the weight of rice and machine components? Consider factors like beam sizes, connection points, and overall stability.	**5**
c. Fabrication Quality	Are the welds, connections, and overall fabrication of the frame free from defects and meet industry standards?	**4**
d. Load Testing	Has the frame undergone any load testing to verify its ability to withstand anticipated weight and stresses? (Consider static and dynamic loads)	**5**
II. Cladding Materials		
a. Suitability for Purpose	Are aluminum composite panels and aluminum stucco sheets suitable for the intended function of the cladding (protection, aesthetics)?	**5**
b. Structural Contribution	Do the cladding materials contribute significantly to the overall structural integrity of the machine, or are they primarily for aesthetics?	**4**
c. Attachment Method	Is the method of attaching the cladding to the frame secure and able to withstand potential forces (e.g., user interaction, accidental bumps)?	**5**
d. Material Testing	Have the cladding materials been tested for their suitability in the intended environment (e.g., moisture resistance, impact resistance)?	**4**
III. Overall Functionality		
a. Accessibility	Does the design allow for easy access to internal components for maintenance and refilling?	**5**
b. User Interaction	Is the machine stable and user-friendly during rice dispensing operations?	**5**
c. Operational Safety	Does the overall design minimize potential safety hazards (e.g., sharp edges, tipping hazards)?	**5**
	Rating Score:	**51 / 55**

Table 10 shows the given survey with the corresponding ratings based from the civil engineer. Since the ratings are on a Likert scale from 1 to 5, with 1 being the lowest and 5 being the highest, a mean rating of 4.63 is quite high. It shows that, on average, the civil engineer gave ratings that are closer to 5 (high satisfaction or agreement) than to 1 (low satisfaction or disagreement). The result suggests that the civil engineer generally had a very positive response to the questions. The average rating of 4.63 reflects a high level of satisfaction, agreement, or positive evaluation regarding the aspects covered by the questions.

$$Weighted\ Score = \frac{\sum_{i=1}^{(\#\ of\ Questions)} Ri}{Total\ Number\ of\ Questions}$$

$$Weighted\ Score = \frac{\sum_{i=1}^{11} Ri}{11}$$

$$Weighted\ Score = \frac{R1 + R2 + R3 + \cdots + R11}{11} = \frac{51}{11} \approx \mathbf{4.63}$$

A weighted overall score equation is used as it effectively summarizes the data by providing the average rating across all questions. This allows for a clear and concise interpretation of the civil engineer's overall satisfaction or agreement with the evaluated criteria (Chen & Singh, 2001). The weighted mean of 4.63 indicates the average rating across all 11 questions. This value is calculated by summing all the ratings and dividing it by the number of questions.

4.3 System Integrations

This section compares the system integrations of existing rice dispensing machines with the proposed machine. It examines their distinct approaches, focusing on materials, specifications, and circuit connections and design, and emphasizes their implications for functionality, compatibility, and overall performance in rice dispensing applications.

i. System Integration of Existing Systems

The "Precision Rice Vending Machine by Using Multiple Load Cell and IoT Based" explores an innovative system designed to improve the distribution of Zakat, Infaq, and Sadaqah (ZIS) during the COVID-19 pandemic. This research addresses the challenge of distributing essential aid, particularly rice, in a manner that minimizes human interaction to reduce the risk of virus transmission. The solution integrates several advanced technologies, primarily focusing on the Internet of Things (IoT), load cells, and RFID technology.

Central to this system is the Internet of Things (IoT), which enables machine-to-machine communication and real-time monitoring through a web interface. The IoT integration ensures that data on rice distribution is accurately recorded and can be monitored remotely, which is crucial for maintaining transparency and accountability in ZIS management.

Load cells play a critical role in the system by ensuring precise measurement of the rice dispensed. The vending machine incorporates multiple load cells to achieve high accuracy. This high level of precision is evidenced by the machine's Mean Absolute Percentage Error (MAPE), which is 2% for 1000g and 3% for 2000g quantities, indicating an accurate dispensing system.

RFID technology is utilized for user identification and access control. Each user is provided with an RFID card, which they scan to access the rice vending machine. This system ensures that only authorized individuals can use the machine, facilitating organized and fair distribution of rice. The RFID system is integrated with the machine's control system, ensuring seamless operation and accurate tracking of each transaction.

A robust web-based monitoring system complements these technologies, providing real-time data on transactions, rice stock levels, and recipient information. This system enables administrators to monitor the number of ZIS recipients, manage rice quotas, and track the total amount of rice dispensed. The web interface ensures that all activities are transparent and can be audited, enhancing trust in the system.

The integration of IoT, load cells, RFID technology, and a comprehensive web-based monitoring system improves accuracy and efficiency and ensures safety by reducing human contact. This innovative approach to ZIS distribution could serve as a model for similar initiatives in other regions, highlighting the potential of technology to address complex social challenges.

A related study, "Design of an ARM-based Automatic Rice-Selling Machine for Cafeterias," presents a comprehensive system integrating various components to enhance the efficiency and hygiene of rice selling in institutional cafeterias. The proposed system integrates multiple hardware and software elements structured into three layers: input, processing, and execution.

The input layer includes devices such as a 4 x 4 keyboard, an optoelectronic switch, an ultrasonic probe, and a pressure sensor, which facilitate user interaction and precise measurement of rice. These inputs are processed by an Advanced RISC Machines (ARM) microprocessor that performs analog-to-digital conversion and signal amplification. The execution layer translates these processed signals into actions, controlling the motor's operation, displaying relevant information, and managing alarms.

The hardware design leverages an ARM Cortex-M3 microprocessor and various sensors to ensure accurate rice dispensing and card payment processing. The mechanical design encompasses a rice storage bin, thermal insulation, and a conveyor system driven by a stepper motor, ensuring efficient rice scattering and delivery. The system employs fuzzy control algorithms to manage the stepper motor's speed, enhancing the machine's precision and reliability.

This integration of advanced microprocessor control, precise measurement, and user-friendly interface significantly improves the efficiency, hygiene, and consistency of rice sold in cafeterias. Both studies highlight the critical role of technology in optimizing rice distribution, whether in the context of aid distribution during a pandemic or improving operational efficiency in institutional settings.

ii. System Integration of Proposed System

The materials used in constructing the Multi-Grain Rice Dispensing Machine are chosen in regards to durability, efficiency, and safety. A comprehensive table with names, pictures, and specifications of each component is provided to offer a detailed understanding of the materials used.

Table 4.2. Main Components and Materials Used

COMPONENT	MAIN MATERIALS	IMAGE
Cabinet	• Stainless Steel • Aluminum	
Dispensing Mechanism	• MG958 Digital Servo Motor 15Kg	
Primary Microcontroller	• Raspberry Pi 4 Model B (2GB)	

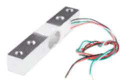
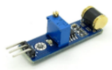
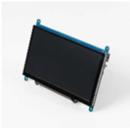
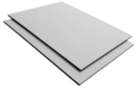
Continuation of Table 4.2		
Load Cell	• Load Sensor + 5Kg Load Cell	
Amplifier	• HX711 Amplifier	
Sensor for Temperature/ Humidity/ Vibration	• DHT22 Temperature/Humidity Sensor • 801S Vibration Shock Sensor Module	 
Packaging Detection Sensor	• ARD-Infrared (IR) Sensor	
User Interface	• Rasberry Pi 7in LEDTouch Screen	
Notification System	• GSM Module Sim900A	
Power Supply	• 12V 5A Power Supply + Cord	
Circuit Components	• Resistors, Capacitors, Diodes	
Seals	• Rubberized Seals	
Payment System	• Alan Universal Coin Slot • TB74 Pulse Bill Acceptor	 
Display Window	• Acrylic Glass • Composite Panels	

Table 11 shows the main components of the machine, the proposed Multi-Grain Rice Dispensing Machine with Short Messaging System Notification aims to revolutionize rice transactions by automating dispensing, providing consumers with at least three rice options, and eliminating the need for direct seller-consumer interaction. This solution leverages various integrated components seamlessly to ensure efficient operation, enhanced security, and optimal user experience.

At the heart of the system is the Raspberry Pi 4, which serves as the central processing unit. It orchestrates the interactions between all components. It runs the software that manages user inputs, processes data from various sensors, controls the dispensing mechanism, and communicates with external systems via the GSM module. The power supply is designed to meet each component's voltage and current requirements, with a buck converter ensuring a five-volt stable power delivery across the system.

The GSM module is crucial for enabling remote monitoring and notification. Connected to the Raspberry Pi via UART, this module sends real-time notifications about the system status to the seller, allowing for efficient remote management. Three servo motors are integral to the dispensing mechanism. Each motor precisely controls the release of rice from different compartments, ensuring accurate dispensing based on user selection. The touchscreen panel provides an intuitive and interactive way for users to select the variety of rice and the desired quantity. It is also a way for the seller to update product details and drain the containers via Administrator mode. The touchscreen displays a user-friendly graphical interface to guide users through the selection and payment process. It enhances the overall user experience by making the machine easy to operate and reducing the need for physical buttons, thus streamlining the design and functionality of the rice dispensing machine.

Temperature sensors within the rice storage compartments continuously monitor rice conditions. Data from these sensors helps maintain optimal storage conditions, ensuring rice quality. The inclusion of a vibration sensor further enhances security. The vibration sensor detects the impact of hard material if someone tries to break the machine, triggering a text message via the GSM module notifying the seller. The IR sensor ensures that rice is only dispensed when the plastic container is correctly positioned, preventing spills and waste.

Admin feedback and interaction are managed through an LED and a push button. The LED provides visual feedback on the machine's status when draining, while the push button allows users to initiate the draining process at the administrator menu. The load cell, paired with an amplifier, accurately measures the amount of rice dispensed, ensuring it matches the user's selection. For payment handling, the machine is equipped with coin and bill acceptors that validate payments and send confirmation signals to the Raspberry Pi, which then authorizes dispensing. This integration of payment mechanisms streamlines the transaction process and enhances the system's security and reliability.

Table 12 shows the integration of other minor components that results in the development of the system. The Raspberry Pi 4's central role ensures smooth operation and coordination among all parts, while the GSM module provides critical remote monitoring capabilities. The servo motors, relay module, and sensors work in tandem to facilitate precise and secure rice dispensing. The proposed machine significantly enhances operational efficiency and customer satisfaction by automating the transaction process and minimizing the need for direct seller involvement, allowing sellers to focus on other tasks within their store.

Table 4.3. Minor Components and Other Materials Used

Raspberry Pi 4 USB-C Charger	Proto Board Large	Videoke Button
Raspberry Pi 4 Heatsink	Soldering Lead	Plastic Small
Raspberry Pi 4 Case w/ Fan	TUDP 20 Guage Wire	Plastic Medium
Raspberry PiT-Cobbler GPIO Breakout	TB 3813P (Terminal Block 3P)	Metal Bowl 20cm
DC-DC Buck Converter	TB 3812P (Terminal Block 2P)	Metal Hook
Micro SD Card 128GB	Terminal Block (12 lines)	Aluminum Stucco Sheet
MicroHDMI Cable 1.5m	Header Pins	Cabinet Lock
Soldering Set + PCB Set	Lead	Rubber Feet
Proto Board Medium	EVA Foam	Resistors, Capacitors, Diodes, and Wires

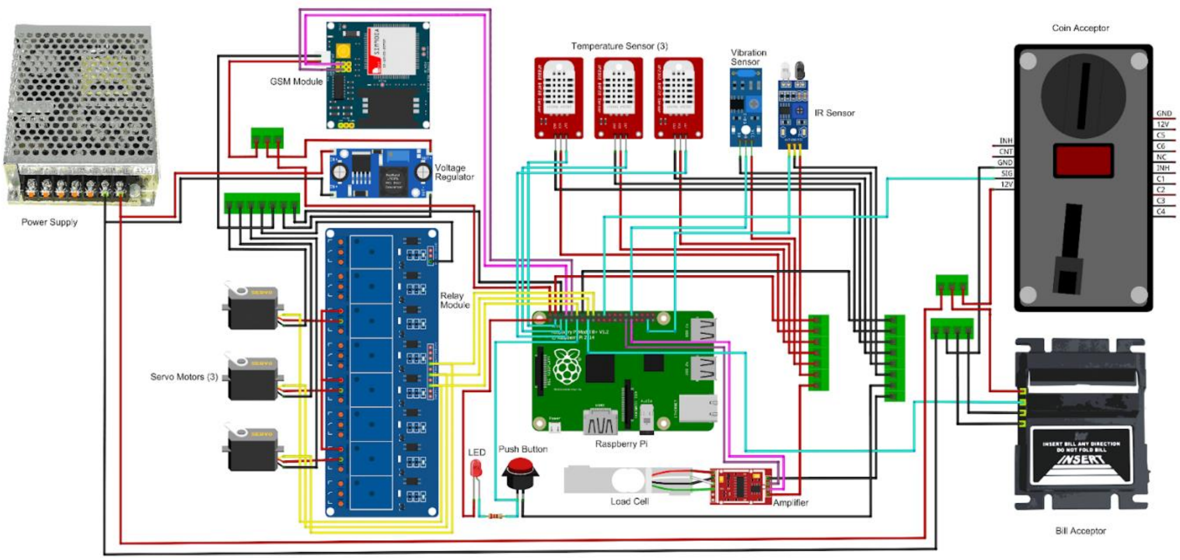


Figure 4.6. Circuit Wiring Diagram

Wiring Legend:

Black (Ground):	Represents all ground connections, establishing a common return path for electric current.
Red (Current):	Signifies power supply lines to all components, ensuring they receive necessary current.
Red (Servo Motors):	Dedicated control signals for operating servo motors.
Cyan (Sensors):	Data lines for transmitting sensor information to the Raspberry Pi.
Pink (TX Wires):	Transmit lines used in serial communication.
Purple (RX Wires):	Receive lines used in serial communication.

Figure 16 represents the circuitry of a dispensing machine, showcasing the interconnections and functionalities of various critical components. Materials in the image shows the power supply, GSM module, buck-converter, servo motors (3 pieces), relay module, temperature sensor (3 pieces), vibration sensor, IR sensor, raspberry pi, and LED, a push button, load cell, amplifier, coin acceptor, and bill acceptor. This detailed schematic is designed to provide an in-depth understanding of the machine's electrical and electronic framework.

iii. Difference of System Integrations of Existing and Proposed System

The proposed Multi-Grain Rice Dispensing Machine with Short Messaging System Notification introduces several significant advancements compared to existing rice dispensing systems. Both systems aim to automate rice distribution and minimize human interaction; however, the proposed system integrates a more sophisticated array of technologies to enhance user experience, security, and operational efficiency.

A key difference lies in the central processing unit. The proposed system uses a Raspberry Pi 4, a powerful and versatile microcomputer that manages interactions between all components, processes sensor data, controls the dispensing mechanism, and communicates with external systems via a GSM module. This integration allows real-time system status notifications to the seller, enhancing remote management capabilities. In contrast, existing systems, such as the one described in "Design of an ARM-based Automatic Rice-Selling Machine for Cafeterias," utilize an ARM Cortex-M3 microprocessor, which, while effective, does not emphasize real-time remote notifications and management to the same extent.

The proposed system's dispensing mechanism is another area of significant improvement. It employs three servo motors to control the release of rice from different compartments, allowing consumers to choose from multiple rice varieties. This multi-grain option enhances user satisfaction by offering greater choice, which is not highlighted in existing systems. Additionally, the proposed machine incorporates a touchscreen panel, providing a user-friendly graphical interface that simplifies the selection and payment process, thereby improving the overall user experience.

The proposed system further enhances security and quality control by including temperature and vibration sensors. Temperature sensors ensure that rice is stored under optimal conditions, maintaining its quality. The vibration sensor adds a layer of security by detecting tampering attempts, which triggers a notification to the seller via the GSM module. This proactive approach to security and quality control is more advanced than the systems described in existing research, which primarily focus on basic dispensing accuracy and user identification through RFID technology.

Integrating a comprehensive web-based monitoring system in existing systems facilitates real-time data on transactions and stock levels, enhancing transparency and accountability. However, the proposed system goes a step further by incorporating a GSM module for immediate status updates, ensuring that any issues such as tampering or low stock levels are promptly addressed.

Payment handling in the proposed system also sees significant advancements, including coin and bill acceptors, which streamline the transaction process and enhance security. This is in contrast to the systems in existing research, which may not focus as extensively on diverse and secure payment methods.

The Multi-Grain Rice Dispensing Machine introduced in this study utilizes the Raspberry Pi 4 to enhance processing and communication capabilities. It incorporates a touchscreen interface to facilitate user interaction, while advanced sensors ensure quality assurance and system security. Integrated payment mechanisms further streamline transactions, marking a notable advancement over traditional rice dispensing systems. These innovations not only enhance the core function of rice distribution but also expand user options, optimize operational efficiency, and bolster overall system integrity and quality control measures.

4.4 Integrated Features on the Dispensing Machine

This section discusses the features integrated into the dispensing machine to address the problem in terms of – Multiple Grain Options, User-friendly Interface, and Payment and Security.

i. Multiple Grain Options

The machine's design must effectively handle and dispense a variety of rice and other grain types, including whole wheat, brown rice, and millet. Cuneio (2020) emphasizes that a stable and reliable dispensing mechanism is essential for accurately dispensing the correct amounts of each grain variety. Sahu (2021) and Song (2023) advocate for the inclusion of acrylic glass or transparent panels to allow users to visually inspect the available grain types. Implementing transparent panels in the machine design enables users to verify the grain type and quality before making a selection. This transparency enhances user confidence, minimizes errors, and fosters trust. For example, Sood and Aggarwhal (2021) of the World Food Programme developed the Annapurti (GrainATM), which features modular units with clear sections that enable users to see the grains inside.

Touchscreens are crucial for user interaction with the dispensing machine, providing a simple and intuitive interface for selecting grain types and quantities. The GrainATM in Gurugram, India, exemplifies the effective use of touchscreens to facilitate a seamless and efficient user experience.

The ability to dispense multiple grain types—such as rice, wheat, and millet—is a vital feature for a versatile dispensing machine. Users can select their preferred grain type from the touchscreen interface, accommodating diverse dietary preferences and optimizing the machine's utility in various settings, from public distribution systems to retail environments. The GrainATM models effectively demonstrate this capability by allowing users to choose and dispense different grains as needed.

By incorporating these features, the machine can meet the needs of a wide range of users, ensuring both convenience and reliability. The use of acrylic glass for visual inspection, combined with touchscreen interfaces for easy navigation and secure transactions, creates a comprehensive solution that aligns with modern standards of efficiency and user satisfaction.

ii. User-Friendly Interface

A survey was conducted to gather user feedback on the machine's effectiveness and trend towards its user interface and use. Participants provided insights on their satisfaction with the machine's performance, ease of use, and the convenience of SMS notifications. In this section of the paper, the researchers present the findings of a survey conducted to evaluate the characteristics of software usability, focusing on Functional Suitability, Performance Efficiency, Interaction Capability, Reliability, and Maintainability.

A sample of 25 participants were selected to provide feedback on software usability. Participants were asked to rate various aspects of software performance using a Likert scale ranging from 0 to 4, with 0 indicating strong disagreement and 4 indicating strong agreement. The data collected were analyzed to assess the extent to which the software met the criteria outlined for each characteristic. The survey results are presented in a table below, highlighting the overall effectiveness and user acceptance of the machine.

Table 4.4. Total Percentage of Survey for Functional Suitability

Functional Suitability				
Questions	**1**	**2**	**3**	**4**
The software performs all the tasks and functions I need it to do for my work.	0	1	5	19
The software offers all the features and functionalities advertised or documented.	0	0	5	20
The software provides the necessary tools and options to efficiently complete my tasks	0	0	6	19
The outputs generated by the software are accurate and consistent with expectations.	0	2	3	20
There are minimal workarounds needed to complete tasks using the software.	0	1	8	16
Weighted Mean				**3.72**

Table 14 discusses functional suitability, which refers to the software's ability to meet user needs and perform required tasks efficiently, was assessed using a weighted mean calculation. The result, a weighted mean of 3.72, indicates that respondents generally agree the software adequately fulfills their work requirements.

Table 4.5. Total Percentage of Survey for Performance Efficiency

Performance Efficiency				
Questions	**1**	**2**	**3**	**4**
The software responds quickly and performs tasks without significant delays.	0	1	11	13
The software utilizes system resources (memory, CPU) efficiently.	0	0	4	21
The software maintains performance even under heavy workloads.	0	1	8	16
Startup times and loading times for the software are reasonable.	0	0	9	16
The software operates smoothly without frequent freezes or crashes.	0	2	8	15
Weighted Mean				**3.62**

Table 15 discusses performance efficiency, it was evaluated using a weighted mean score of 3.62, focusing on factors such as responsiveness, resource utilization, and stability under varying workloads. Participants expressed moderate satisfaction with the software's performance, noting its effectiveness under heavy workloads and efficient system resource use, contributing to a smooth user experience.

Table 4.6. Total Percentage of Survey for Interaction Capability

Interaction Capability				
Questions	**1**	**2**	**3**	**4**
The user interface of the software is easy to understand and navigate.	0	3	4	18
The software provides clear and concise instructions and error messages.	0	1	8	16
The software offers user-friendly customization options for interface preferences.	0	0	9	16
The software provides helpful feedback during interaction to guide user actions.	0	2	6	17
The software utilizes consistent design principles throughout the interface.	0	1	8	16
Weighted Mean				**3.61**

Table 16 discusses the software's interaction capability, assessed with a weighted mean score of 3.61, shows positive user feedback in key areas. Users find the interface easy to navigate, appreciate clear instructions and error messages, and value customizable options. They also receive helpful feedback during interactions, enhancing user confidence. Consistent design principles throughout the interface contribute to a cohesive and user-friendly experience.

Table 4.7. Total Percentage of Survey for Reliability

Reliability				
Questions	**1**	**2**	**3**	**4**
The software utilizes consistent design principles throughout the interface.	0	1	8	16
The software experiences minimal downtime or unexpected crashes.	0	2	9	14
Data entered or stored in the software is accurate and remains secure.	0	1	9	15
The software recovers gracefully from errors without significant data loss.	0	2	7	16
The software performs reliably over extended periods of use without degradation in performance.	0	1	10	14
Weighted Mean	**3.54**			

Table 17 discusses machine reliability, with a weighted mean score of 3.54, the software's reliability shows strong performance in key areas. Users noted the consistent application of design principles, which fosters a stable and predictable user experience. Reports of minimal downtime and rare, unexpected crashes highlight the software's dependable nature. Data accuracy and security are well-maintained, reinforcing user trust in the system's integrity. Additionally, the software recovers gracefully from errors, effectively minimizing data loss. Over prolonged use, the software continues to perform reliably without any noticeable degradation. Overall, these results indicate that the software reliably meets user expectations, demonstrating stability, data integrity, robust error recovery, and consistent long-term performance.

Table 4.8. Total Percentage of Survey for Maintainability

Maintainability				
Questions	**1**	**2**	**3**	**4**
I feel confident that using this software will not cause any data loss or system crashes.	0	1	10	14
The software has adequate safeguards in place to prevent unauthorized access and malicious attacks.	0	0	12	13
The software provides clear warnings and error messages to help me avoid making mistakes with potentially negative consequences.	0	0	10	15
I trust the software to handle sensitive information securely and reliably.	0	2	3	20
The software operates consistently without unexpected behavior that could compromise safety.	0	0	9	16
Weighted Mean	**3.6**			

Table 18 discusses maintainability, which assesses the software's ease of maintenance, data security, and handling of sensitive information, received a favorable evaluation from participants. With a weighted mean score of 3.6, users expressed confidence in the software's ability to prevent data loss and system crashes. This score reflects a positive perception of the software's maintainability, indicating that users trust its reliability and effectiveness in safeguarding data and ensuring stable operation.

Table 4.9. Total Rating per Characteristic and Overall Percentage

Characteristics	1	2	3	4
Functional Suitability	0%	3.20%	21.60%	75.20%
Performance Efficiency	0%	3%	32%	64.80%
Interaction Capability	0%	6.40%	25.60%	68%
Reliability	0%	5.60%	34.40%	60%
Maintainability	0%	2.40%	35.20%	62.40%
	0%	**4.16%**	**29.76%**	**66.08%**

Table 20 shows the survey findings highlight positive perceptions of software usability across various characteristics. While there are areas for improvement, particularly in performance efficiency and interaction capability, the results indicate that the surveyed software generally meets user expectations in terms of functional suitability, reliability, and maintainability.

iii. Payment and Security

For a rice dispensing machine that accepts both coins and bills, the coin (Alan Universal Coin Slot) and bill (TB74 Pulse Bill Acceptor) acceptors play vital roles in facilitating smooth transactions. The system is tailored to accept legal tender Philippine peso coins in denominations of 1, 5, 10, and 20, as well as bills in denominations of 20, 50, and 100 pesos, aligning with the common denominations used by customers for purchasing rice. The acceptors are designed to accommodate coins with superficial markings or dirt, as long as they meet the specified thickness criteria of not exceeding 3 mm. The integration of coin and bill acceptors in the rice vending machine not only streamlines the payment process but also enhances customer satisfaction, operational efficiency, and security, making it a beneficial feature for both users and operators (Ismail et al., 2020).

In regard to security, apart from the external case, type of material used, and emergency interactions from the machine. A vibration sensor (801S Vibration Shock Sensor Module) is integrated inside the interior circuit and hardware. Sudden shocks or extreme damage received by the machine will be detected by the vibration which will then prompt the GSM Module to prompt a warning text message to the seller and the developers. A vibration sensor is a device designed to detect vibrations and convert them into electrical signals. This sensor, often referred to as a casing measurement, typically employs seismic sensor transducers to measure both speed and acceleration. Speed is measured using a velocity probe, while acceleration is gauged with an acceleration sensor probe (Gunawan et al., 2021)

4.5 Efficiency of the system

This section discusses the efficiency of the system in terms of its capability with regards to – Dispensing Automation, Measurement in Kilograms, and Sustainability.

i. Dispensing Automation and Measurement in Kilograms

The system's efficiency in dispensing automation and measurement in kilograms is assessed by considering the speed and accuracy of rice dispensing. Detailed tests involve measuring the time it takes to dispense specific rice quantities, evaluating load cell precision, and assessing SMS notification reliability. The resulting data is presented in performance metric tables. In the development of the Multi-Grain Rice Dispensing Machine, extensive testing was conducted to evaluate the accuracy and efficiency of the dispensing mechanism across three different containers.

The data testing for Containers 1, 2, and 3 of the Multi-Grain Rice Dispensing Machine involved conducting 20 trials, divided into three sections based on different target weights: 1000 grams, 2000 grams, and 3000 grams. Each trial measured the actual weight dispensed, the target weight, the error, ABS, the time it took to tare, and the total time it took to dispense. The Mean Absolute Percentage Error (MAPE) was calculated for each target weight category to assess the accuracy of the dispensing mechanism. For each trial, the following parameters were measured:

Target Weight: This is the pre-set weight value that the dispensing mechanism attempts to achieve. It is used as a benchmark for evaluating the system's performance. For the purposes of this analysis, target weights of 1000 g, 2000 g, and 3000 g were used.

(N) No. of Trials: This indicates the number of trials conducted to evaluate the performance of the dispensing mechanism. In each set of trials, the actual weight dispensed is recorded and analyzed.

Actual Weight: is the weight measured after the dispensing mechanism has completed a trial. It is compared to the target weight to assess accuracy.

Error: is the difference between the Actual Weight and the Target Weight. It indicates how much the dispensed weight deviates from the desired weight. It is calculated as:

$$Error = Actual\ Weight - Target\ Weight$$

ABS Error (Absolute Error): is the absolute value of the Error. It shows the magnitude of the deviation without considering its direction (over or under the target).

ABS Error per Actual: is the absolute error expressed as a proportion of the actual weight. This metric normalizes the error relative to the actual weight, providing a sense of proportional accuracy. It is calculated as:

$$ABS\ Error\ per\ Actual = \frac{ABS\ Error}{Actual\ Weight}$$

∑ (Summation of ABS Error per Actual): this is the total sum of the ABS Error per Actual values over all trials. It provides a cumulative measure of the proportional errors across all trials.

MAPE (Mean Absolute Percentage Error): is the average of the ABS Error per Actual values, expressed as a percentage. It is calculated as:

$$MAPE = \left(\frac{1}{N}\sum ABS\ Error\ per\ Actual\right) \times 100$$

ii. Data Testing for Machine Function (Container 1)

Table 21 shows the target weight of 1000 g for the first container, the researchers conducted 20 trials to evaluate the accuracy and performance of the dispensing mechanism. The actual weights measured during these trials showed a range of minor deviations from the target weight. The absolute errors per trial, calculated as |Actual - Target| / Target, were used to determine the Mean Absolute Percentage Error (MAPE).

The errors in the measurements varied from -2.2 grams to 22.8 grams, indicating an absolute error range between 0.002 grams and 0.023 grams. These deviations show that while most trials remained fairly close to the target weight, a few instances exhibited larger discrepancies. This suggests that although the system generally performs well, there are occasional outliers that need to be addressed.

The time required to tare the system ranged from 2.51 seconds to 9.08 seconds. In contrast, the dispensing times spanned from 20.38 seconds to 26.79 seconds. These variations indicate differences in the efficiency and speed of the system's operations, with some instances taking significantly longer than others.

Table 4.10. Dispensing Mechanism - Container 1 (1000 g)

CONTAINER 1						
Target Weight: 1000 g						
No. of Trials	**Actual weight**	**Error**	**ABS Error**	**ABS Error per Actual**	**Time it takes to tare**	**Time it takes to dispense**
1	1004.7	4.7	4.7	0.005	8.27	25.69
2	1018.4	18.4	18.4	0.018	3.73	21.55
3	1020.2	20.2	20.2	0.020	2.77	20.88
4	1006.9	6.9	6.9	0.007	4.48	22.02
5	1020.1	20.1	20.1	0.020	8.28	26.79
6	1000.4	0.4	0.4	0.000	9.08	26.34
7	1020.2	20.2	20.2	0.020	5.52	23.23
8	1004.6	4.6	4.6	0.005	3.22	21.32
9	1022.8	22.8	22.8	0.023	2.98	21.59
10	1002.6	2.6	2.6	0.003	6.38	23.89
11	1006.9	6.9	6.9	0.007	3.79	21.82
12	1010.4	10.4	10.4	0.010	2.63	20.55
13	1018.4	18.4	18.4	0.018	3.95	23.3
14	1000.4	0.4	0.4	0.000	6.06	24.11
15	1014.4	14.4	14.4	0.014	2.51	20.38
16	997.8	-2.2	2.2	0.002	4.15	22.23
17	1021.2	21.2	21.2	0.021	5.4	23.15
18	1019.2	19.2	19.2	0.019	6.11	24.19
19	1002.4	2.4	2.4	0.002	3.55	21.79
20	1013.2	13.2	13.2	0.013	4.86	24.28
∑				**0.230**		
N				**20**		
MAPE				**1.148**		

Table 4.11. Dispensing Mechanism - Container 1 (2000 g)

CONTAINER 1						
Target Weight: 2000 g						
No. of Trials	**Actual weight**	**Error**	**ABS Error**	**ABS Error per Actual**	**Time it takes to tare**	**Time it takes to dispense**
1	2005.3	5.3	5.3	0.003	3.76	38.55
2	2002.6	2.6	2.6	0.001	9.42	44.26
3	2019.5	19.5	19.5	0.010	6.37	41.12
4	2017.1	17.1	17.1	0.009	3.98	39.71
5	2002.7	2.7	2.7	0.001	3.75	37.95
6	2004.4	4.4	4.4	0.002	3.8	38.5
7	2008.6	8.6	8.6	0.004	4.49	39.25
8	2010.8	10.8	10.8	0.005	3.62	36.94
9	2000	0	0	0.000	11.8	46.58
10	2004.8	4.8	4.8	0.002	9.1	44.3
11	2011.9	11.9	11.9	0.006	10.38	45.9
12	1997.5	-2.5	2.5	0.001	5.05	39.06
13	2014.4	14.4	14.4	0.007	3.43	39.31
14	2013.2	13.2	13.2	0.007	5.79	41.53
15	2012.2	12.2	12.2	0.006	8.25	44.18
16	2017.1	17.1	17.1	0.009	5.67	41.17
17	2009	9	9	0.005	5.57	40.56
18	2013.5	13.5	13.5	0.007	5.23	41.32
19	2019	19	19	0.010	5.82	39.69
20	2003.8	3.8	3.8	0.002	3.49	36.02
$\sum$				**0.096**		
N				**20**		
MAPE				**0.481**		

Table 22 shows the target weight of 2000 g for the first container. The absolute errors per actual weight ranged from -2.5 grams to 19.5 grams, resulting in an absolute error range of 0.001 to 0.010. This data illustrates that the system generally performs reliably, with most measurements closely aligning with the target weights. However, there are occasional minor discrepancies that highlight areas for potential improvement.

The taring times varied from 3.43 seconds to 11.80 seconds, while the dispensing times ranged between 36.02 seconds and 46.58 seconds. These metrics reflect a consistent operation cycle, indicating that the system operates within a predictable timeframe, albeit with some variation in the time taken for different tasks.

The summation of the absolute errors per actual weight was 0.096, leading to a Mean Absolute Percentage Error (MAPE) of 0.481%. This low MAPE value suggests a high level of accuracy and reliability in dispensing weights close to the target values. Overall, the system demonstrates strong performance, with minimal deviations from the desired weights, ensuring precise and dependable operations.

Table 4.12. Dispensing Mechanism - Container 1 (3000 g)

CONTAINER 1						
Target Weight: 3000 g						
No. of Trials	Actual weight	Error	ABS Error	ABS Error per Actual	Time it takes to tare	Time it takes to dispense
1	3017.6	17.6	17.6	0.006	3.42	55.73
2	3024	24	24	0.008	3.56	56.09
3	3004.3	4.3	4.3	0.001	10.49	1:01.14
4	3074.5	74.5	74.5	0.025	11.97	1:04.47
5	2998	-2	2	0.001	10.86	1:01.97
6	2992.6	-7.4	7.4	0.002	10.06	59.72
7	3004.6	4.6	4.6	0.002	4.57	56.43
8	3000.1	0.1	0.1	0.000	7.61	59.97
9	3000	0	0	0.000	12.86	1:04.01
10	3005.3	5.3	5.3	0.002	4.72	56.09
11	3000.6	0.6	0.6	0.000	8.57	1:00.71
12	3011.2	11.2	11.2	0.004	4.78	56.59
13	3011.5	11.5	11.5	0.004	5.65	57.62
14	3005.2	5.2	5.2	0.002	14.5	1:01.20
15	3018.5	18.5	18.5	0.006	5.18	57.19
16	2999.2	-0.8	0.8	0.000	11.9	1:03.83
17	3017.9	17.9	17.9	0.006	8.77	1:02.95
18	3016.1	16.1	16.1	0.005	9.4	1:02.44
19	3007.1	7.1	7.1	0.002	4.97	57.45
20	3004.1	4.1	4.1	0.001	6.35	58.22
∑				0.078		
N				20		
MAPE				0.388		

Table 23 shows the target weight of 3000 g for the first container. The absolute errors per actual weight varied from -7.4 grams to 74.5 grams, resulting in an absolute error range of 0.000 to 0.025. While most trials maintained close proximity to the target weight, there were a few instances with larger deviations. This indicates that the system generally performs well, but there are occasional outliers that cause significant errors.

The time required to tare the system ranged from 3.42 seconds to 14.50 seconds. Dispensing times spanned between 55.73 seconds and 64.47 seconds. These measurements suggest that the system operates with consistent timing, although there is some variation in the duration needed for taring and dispensing tasks.

The summation of the absolute errors per actual weight was 0.078, resulting in a Mean Absolute Percentage Error (MAPE) of 0.388%. This low MAPE value underscores the system's strong performance and high accuracy in dispensing weights close to the target values. Overall, the system demonstrates a high level of precision, with minimal deviations, ensuring reliable and accurate operations.

Container 1 Analysis

Across the three target weights (1000 g, 2000 g, and 3000 g), the dispensing mechanism of Container 1 demonstrated a high degree of accuracy and consistency. The MAPE values of 1.148%, 0.481%, and 0.388% for the respective target weights indicate a reliable performance with minor deviations. The time metrics for taring and dispensing remained within acceptable ranges, reflecting the system's efficiency. This comprehensive evaluation confirms the robustness of the dispensing mechanism, with a slight room for improvement in minimizing the occasional higher discrepancies observed in some trials.

iii. Data Testing for Machine Function (Container 2)

Table 4.13. Dispensing Mechanism - Container 2 (1000 g)

CONTAINER 2						
Target Weight: 1000 g						
No. of Trials	**Actual weight**	**Error**	**ABS Error**	**ABS Error per Actual**	**Time it takes to tare**	**Time it takes to dispense**
1	1012.1	12.1	12.1	0.012	8.08	32.65
2	1002	2	2	0.002	2.83	29.22
3	994.6	-5.4	5.4	0.005	8.94	31.82
4	1016.8	16.8	16.8	0.017	6.59	31.41
5	1020.4	20.4	20.4	0.020	4.76	29.17
6	1005.7	5.7	5.7	0.006	3.78	28.72
7	1017.5	17.5	17.5	0.018	3.23	28.28
8	1004.8	4.8	4.8	0.005	5.73	32.11
9	996.6	-3.4	3.4	0.003	5.65	31.04
10	1000.9	0.9	0.9	0.001	5.93	30.69
11	1006.6	6.6	6.6	0.007	8.48	33.25
12	1012.2	12.2	12.2	0.012	5.72	30.53
13	1006.3	6.3	6.3	0.006	3.17	28.86
14	1011.8	11.8	11.8	0.012	8.96	32.68
15	1000.8	0.8	0.8	0.001	2.96	29.69
16	997.2	-2.8	2.8	0.003	2.84	30.26
17	1009.3	9.3	9.3	0.009	4.28	29.35
18	1007.4	7.4	7.4	0.007	4.21	30.1
19	1012.8	12.8	12.8	0.013	8.44	31.67
20	1006.1	6.1	6.1	0.006	5.97	32.25
∑				**0.165**		
N				**20**		
MAPE				**0.825**		

Table 24 shows the target weight of 1000 g for the second container. The errors in the measurements varied from -5.4 grams to 20.4 grams, resulting in an absolute error range of 0.001 to 0.020. While the majority of the trials stayed relatively close to the target weight, a few instances recorded higher discrepancies. This pattern indicates that the system generally performs well, with most measurements aligning closely with the intended weights, though occasional larger deviations do occur.

The time required to tare the system ranged from 2.83 seconds to 8.96 seconds, while the dispensing times varied between 28.28 seconds and 33.25 seconds. These figures reflect a consistent operational cycle, with some variation in the time taken for taring and dispensing tasks. Overall, the system shows a predictable performance in terms of timing, ensuring efficient operations within a defined range.

The summation of the absolute errors per actual weight was 0.165, leading to a Mean Absolute Percentage Error (MAPE) of 0.825%. This MAPE value suggests a relatively high level of accuracy, with minor deviations in the dispensed weights. Overall, the system demonstrates strong performance, maintaining a high degree of precision and reliability in its operations.

Table 4.14. Dispensing Mechanism - Container 2 (2000 g)

CONTAINER 2						
Target Weight: 2000 g						
No. of Trials	**Actual weight**	**Error**	**ABS Error**	**ABS Error per Actual**	**Time it takes to tare**	**Time it takes to dispense**
1	2008.8	8.8	8.8	0.004	8.84	58.11
2	2010.3	10.3	10.3	0.005	7.18	57.39
3	2005.4	5.4	5.4	0.003	7.31	56.81
4	2003.5	3.5	3.5	0.002	5.14	55.7
5	2019.5	19.5	19.5	0.010	5.34	56.34
6	2012.6	12.6	12.6	0.006	4.6	58.97
7	1994.1	-5.9	5.9	0.003	3.09	54.51
8	2007.9	7.9	7.9	0.004	5.2	55.95
9	2015.6	15.6	15.6	0.008	4.73	56.51
10	2005.1	5.1	5.1	0.003	3.99	57.38
11	1996.5	-3.5	3.5	0.002	3.09	52.02
12	2014.6	14.6	14.6	0.007	5.85	59.2
13	2008.7	8.7	8.7	0.004	3.29	52.94
14	2010.9	10.9	10.9	0.005	5.26	55.46
15	2004.8	4.8	4.8	0.002	5.06	58.38
16	2013.6	13.6	13.6	0.007	4.41	57.62
17	2002.3	2.3	2.3	0.001	2.52	50.62
18	2011.1	11.1	11.1	0.006	4.56	55.49
19	2006.4	6.4	6.4	0.003	5.79	57.46
20	2016.2	16.2	16.2	0.008	7.3	59.08
$\sum$				**0.093**		
N				**20**		
MAPE				**0.467**		

Table 25 shows the target weight of 2000 g for the second container. The absolute errors per actual weight ranged from -5.9 grams to 19.5 grams, resulting in an absolute error range of 0.001 to 0.010. This data indicates a generally reliable performance, with most trials staying close to the target weight. However, occasional minor discrepancies were observed, suggesting areas for potential improvement.

The time required to tare the system ranged from 2.52 seconds to 8.84 seconds. Dispensing times varied between 50.62 seconds and 59.20 seconds. These metrics reflect a consistent operational cycle, indicating that the system operates predictably and efficiently within a defined range of times.

The summation of the absolute errors per actual weight was 0.093, leading to a Mean Absolute Percentage Error (MAPE) of 0.467%. This lower MAPE value underscores the system's high level of accuracy and reliability in dispensing weights close to the target values. Overall, the system demonstrates strong performance, maintaining precision and dependability in its operations.

Table 4.15. Dispensing Mechanism - Container 2 (3000 g)

CONTAINER 2						
Target Weight: 3000 g						
No. of Trials	**Actual weight**	**Error**	**ABS Error**	**ABS Error per Actual**	**Time it takes to tare**	**Time it takes to dispense**
1	3006.2	6.2	6.2	0.002	4.27	1:13.57
2	2993.6	-6.4	6.4	0.002	7.76	1:16.28
3	3030.5	30.5	30.5	0.010	22.53	1:33.30
4	2995.2	-4.8	4.8	0.002	4.65	1:13.21
5	3006.3	6.3	6.3	0.002	18.04	1:27.05
6	3000.8	0.8	0.8	0.000	4.82	1:14.56
7	3000.1	0.1	0.1	0.000	4.86	1:13.67
8	2998.3	-1.7	1.7	0.001	11.24	1:21.96
9	3004.5	4.5	4.5	0.002	9.5	1:18.19
10	3002.5	2.5	2.5	0.001	4.94	1:12.21
11	3010.9	10.9	10.9	0.004	3.88	1:15.70
12	3011.3	11.3	11.3	0.004	7.54	1:18.32
13	3009.6	9.6	9.6	0.003	3.97	1:11.86
14	2996.2	-3.8	3.8	0.001	3.96	1:11.51
15	3009.7	9.7	9.7	0.003	11.12	1:20.10
16	3000	0	0	0.000	1.037	1:18.33
17	3011.2	11.2	11.2	0.004	4.41	1:12.40
18	3008.4	8.4	8.4	0.003	6.28	1:12.95
19	2996.8	-3.2	3.2	0.001	10.49	1:18.29
20	3014.1	14.1	14.1	0.005	9.32	1:18.15
∑				**0.049**		
N				**20**		
MAPE				**0.243**		

Table 26 shows the target weight of 3000 g for the second container. The absolute errors per actual weight ranged from -6.4 grams to 30.5 grams, resulting in an absolute error range of 0.000 to 0.010. While most trials maintained close proximity to the target weight, a few instances exhibited larger deviations. This indicates that the system generally performs reliably, though there are occasional outliers that result in greater errors.

The time required to tare the system ranged from 3.88 seconds to 22.53 seconds, while the dispensing times varied between 1:11.51 and 1:33.30. These metrics suggest that the system operates with consistent timing, albeit with some variation in the duration needed for taring and dispensing tasks. The predictable nature of these timings reflects the system's operational consistency.

The summation of the absolute errors per actual weight was 0.049, leading to a Mean Absolute Percentage Error (MAPE) of 0.243%. This low MAPE value underscores the system's strong performance and high accuracy in dispensing weights close to the target values. Overall, the system demonstrates a high level of precision, with minimal deviations, ensuring reliable and accurate operations.

Container 2 Analysis

Across the three target weights (1000 g, 2000 g, and 3000 g), the dispensing mechanism of Container 2 demonstrated a high degree of accuracy and consistency. The MAPE values of 0.825%, 0.467%, and 0.243% for the respective target weights indicate a reliable performance with minor deviations. The time metrics for taring and dispensing remained within acceptable ranges, reflecting the system's efficiency. This comprehensive evaluation confirms the robustness of the dispensing mechanism, with a slight room for improvement in minimizing the occasional higher discrepancies observed in some trials.

iv. Data Testing for Machine Function (Container 3)

Table 4.16. Dispensing Mechanism - Container 3 (1000 g)

CONTAINER 3						
Target Weight: 1000 g						
No. of Trials	**Actual weight**	**Error**	**ABS Error**	**ABS Error per Actual**	**Time it takes to tare**	**Time it takes to dispense**
1	1012.8	12.8	12.8	0.013	6.36	26.17
2	1019.1	19.1	19.1	0.019	11.59	33.02
3	1001.8	1.8	1.8	0.002	4.4	24.9
4	994.8	-5.2	5.2	0.005	2.83	23.67
5	1000.6	0.6	0.6	0.001	5.89	26.32
6	1002.2	2.2	2.2	0.002	3.42	25.3
7	996.7	-3.3	3.3	0.003	6.15	26.07
8	1001.3	1.3	1.3	0.001	3.81	23.83
9	1003	3	3	0.003	3.35	24.47
10	995.2	-4.8	4.8	0.005	2.96	22.77
11	1011.2	11.2	11.2	0.011	10.07	30.57
12	1002.6	2.6	2.6	0.003	3.61	23.44
13	1010.2	10.2	10.2	0.010	10.07	31.04
14	998.2	-1.8	1.8	0.002	3.93	24.52
15	1000.2	0.2	0.2	0.000	3.42	24.42
16	997.2	-2.8	2.8	0.003	9.41	29.46
17	1004.3	4.3	4.3	0.004	4.32	25.03
18	998.3	-1.7	1.7	0.002	9.95	30.44
19	1002.8	2.8	2.8	0.003	3.61	24.92
20	1012.2	12.2	12.2	0.012	9.03	29.87
∑				**0.104**		
N				**20**		
MAPE				**0.520**		

Table 27 shows the target weight of 1000 g for the third container. The errors in the measurements varied from -5.2 grams to 19.1 grams, resulting in an absolute error range of 0.001 to 0.019. These deviations indicate that while the majority of trials remained close to the target weight, a few instances recorded higher discrepancies. This pattern suggests that the system generally performs well, maintaining accuracy in most cases, but occasionally exhibits larger errors that need to be addressed.

The time required to tare the system ranged from 2.83 seconds to 11.59 seconds, while the dispensing times varied between 22.77 seconds and 33.02 seconds. These figures reflect a consistent operational cycle, with some variation in the time taken for taring and dispensing tasks. Overall, the system demonstrates predictable performance in terms of timing, ensuring efficient operations within a defined range.

The summation of the absolute errors per actual weight was 0.104, resulting in a Mean Absolute Percentage Error (MAPE) of 0.520%. This MAPE value suggests a relatively high level of accuracy, with minor deviations in the dispensed weights. Overall, the system demonstrates strong performance, maintaining a high degree of precision and reliability in its operations.

Table 4.17. Dispensing Mechanism - Container 3 (2000 g)

CONTAINER 3						
Target Weight: 2000 g						
No. of Trials	Actual weight	Error	ABS Error	ABS Error per Actual	Time it takes to tare	Time it takes to dispense
1	2013.3	13.3	13.3	0.007	3.95	44.29
2	2014.2	14.2	14.2	0.007	6.55	46.43
3	2009.9	9.9	9.9	0.005	9.74	49.72
4	2016.4	16.4	16.4	0.008	7.85	50.18
5	2018.1	18.1	18.1	0.009	12.43	55.64
6	2013.1	13.1	13.1	0.007	9.35	52.32
7	2001.2	01.2	1.2	0.001	9.49	50
8	2014.9	14.9	14.9	0.007	4.25	45.21
9	2008.5	8.5	8.5	0.004	4.4	46.25
10	2004.4	4.4	4.4	0.002	7.45	47.3
11	2009.9	9.9	9.9	0.005	5.5	48.04
12	2007.6	7.6	7.6	0.004	9.42	51.75
13	2006	6	6	0.003	4.71	45.52
14	2001.2	1.2	1.2	0.001	7.39	48.03
15	2008.3	8.3	8.3	0.004	9.17	49.74
16	2003.1	3.1	3.1	0.002	7.9	48.46
17	2016.6	16.6	16.6	0.008	4.84	47.47
18	2017.2	17.2	17.2	0.009	7.72	49.93
19	2016.5	16.5	16.5	0.008	6.97	48.35
20	2020.7	20.7	20.7	0.010	4.13	45.95
∑				**0.111**		
N				**20**		
MAPE				**0.553**		

Table 28 shows the target weight of 2000 g for the third container. The absolute errors per actual weight ranged from 1.2 grams to 20.7 grams, resulting in an absolute error range of 0.001 to 0.010. This data illustrates a generally reliable performance, with most measurements closely aligning with the target weights. However, occasional minor discrepancies were observed, suggesting some variability in the system's accuracy.

The time required to tare the system ranged from 3.95 seconds to 12.43 seconds. Dispensing times varied between 44.29 seconds and 55.64 seconds. These metrics indicate a consistent operation cycle, reflecting the system's ability to perform tasks within predictable timeframes despite some variation in duration.

The summation of the absolute errors per actual weight was 0.111, resulting in a Mean Absolute Percentage Error (MAPE) of 0.553%. This MAPE value indicates a high level of accuracy and reliability in dispensing weights close to the target values. Overall, the system demonstrates strong performance, maintaining precise and dependable operations with minimal deviations from the desired weights.

Table 4.18. Dispensing Mechanism - Container 3 (3000 g)

CONTAINER 3						
Target Weight: 3000 g						
No. of Trials	Actual weight	Error	ABS Error	ABS Error per Actual	Time it takes to weigh	Time it takes to dispense
1	3006.6	6.6	6.6	0.002	12.41	1:12.73
2	3022.9	22.9	22.9	0.008	12.42	1:14.14
3	2029	-971	-971	0.324	14.64	1:16.14
4	3019.2	19.2	19.2	0.006	3.29	1:04.46
5	3000.5	0.5	0.5	0.000	9.46	1:10.27
6	3001.3	1.3	1.3	0.000	10.95	1:11.10
7	3017.5	17.5	17.5	0.006	10.04	1:13.49
8	3007.5	7.5	7.5	0.003	6.68	1:08.50
9	3002.5	2.5	2.5	0.001	6.68	1:08.32
10	3015.4	15.4	15.4	0.005	9.27	1:13.72
11	3011.1	11.1	11.1	0.004	3.94	1:06.73
12	3003.4	3.4	3.4	0.001	7.34	1:07.81
13	2999.6	-0.4	0.4	0.000	15.31	1:16.46
14	3000.2	0.2	0.2	0.000	3.74	1:05.64
15	3008.2	8.2	8.2	0.003	4.92	1:08.38
16	3009.9	9.9	9.9	0.003	4.44	1:07.75
17	3012.2	12.2	12.2	0.004	12.89	1:14.22
18	2998.5	-1.5	1.5	0.001	5.12	1:06.15
19	3011.2	11.2	11.2	0.004	15.82	1:16.97
20	3001.3	1.3	1.3	0.000	8.05	1:08.50
∑				0.375		
N				20		
MAPE				1.873		

Table 29 shows the target weight of 3000 g for the third container.The absolute errors per actual weight ranged from -971 grams to 22.9 grams, resulting in an absolute error range of 0.000 to 0.324. While most trials maintained close proximity to the target weight, a few instances exhibited larger deviations. This indicates that the system generally performs well, with the majority of measurements aligning closely with the intended weights. However, the presence of significant outliers suggests areas where improvements could enhance overall accuracy.

The time required to tare the system ranged from 3.29 seconds to 15.82 seconds, while the dispensing times varied between 1 minute and 4.46 seconds to 1 minute and 16.97 seconds. These metrics indicate consistent operational timing, highlighting the system's ability to maintain predictable performance despite varying durations for taring and dispensing tasks.

The summation of the absolute errors per actual weight was 0.375, resulting in a Mean Absolute Percentage Error (MAPE) of 1.873%. This MAPE value indicates overall strong performance with high accuracy in dispensing weights close to the target values. However, the presence of a few outliers significantly impacted the overall error rate, suggesting potential areas for refinement to further enhance accuracy and reliability in the system's operations.

Container 3 Analysis

Across the three target weights (1000 g, 2000 g, and 3000 g), the dispensing mechanism of Container 3 demonstrated a high degree of accuracy and consistency. The MAPE values of 0.520%, 0.553%, and 1.873% for the respective target weights indicate a reliable performance with minor deviations. The time metrics for taring and dispensing remained within acceptable ranges, reflecting the system's efficiency. This comprehensive evaluation confirms the robustness of the dispensing mechanism, with a slight room for improvement in minimizing the occasional higher discrepancies observed in some trials.

4.6 Sustainability

This section evaluates sustainability through rigorous testing of temperature and humidity sensors for reliability and environmental adaptability. It also examines structural design and system integrations based on ISO standards and national codes, ensuring robustness, safety, and efficiency in compliance with industry benchmarks.

i. Data Testing for Temperature and Humidity Sensor

In this section, a detailed methodology and results for testing of the DHT22 temperature and humidity sensors. Each sensor was deployed in one of the three containers, and data were collected on an hourly basis.

For each of the three tests, data were recorded and analyzed. The temperature data were measured in degrees Celsius (°C), and the humidity data were recorded in relative humidity percentage (RH%). By conducting these tests, the researchers aimed to ensure that the DHT22 sensors provide accurate and consistent readings under different conditions, thereby validating their use for monitoring the internal environment of the containers.

The results of these tests are crucial for understanding the reliability of the DHT22 sensors in various scenarios and ensuring that they can be trusted for precise temperature and humidity monitoring in the context of the containers used in the study.

ii. Correlations of DHT22 Sensor vs Digital Reader

This test was conducted on May 21, 2024, to evaluate the accuracy of the DHT22 sensors compared to a high-precision digital reader. An FY-12 Digital Temperature and Humidity Meter with LCD is used as the reference device. The FY-12 meter has a temperature accuracy of ±1% and a humidity accuracy of ±5%.

The testing involved simultaneous readings from the DHT22 sensors and the FY-12 meter across all three containers. Data was collected every hour over a 12-hour period, and the correlation between the DHT22 sensor readings and the digital reader was analyzed to determine the sensors' reliability and precision.

Table 4.19. DHT Sensor vs Digital Reader (Container 1)

May 21, 2024				
CONTAINER 1				
Time	DHT		READER	
	C°	RH%	C°	RH%
9:00 AM	35.3	65.8	36.8	63
10:00 AM	36.4	58.5	35.9	62
11:00 AM	36.3	57.2	36.1	63
12:00 PM	37	55.8	36.8	62
1:00 PM	36.2	57.1	36.1	62
2:00 PM	34.5	62.6	34	65
3:00 PM	35.1	61.4	35.4	62
4:00 PM	35.9	58.8	35.8	61
5:00 PM	35.7	58.2	35.2	59
6:00 PM	34.1	62	34.6	61
7:00 PM	33.1	65.8	34.1	64
8:00 PM	32.7	68.9	33.1	69
9:00 PM	33.4	67.8	33.3	71

Table 30 shows the comparison of temperature and humidity between the DHT sensor and digital reader for the first container. Throughout the day, the DHT22 sensor provided consistent temperature readings with slight fluctuations. In the morning, these readings tended to be slightly lower compared to the temperatures recorded by the digital reader, but they showed closer alignment during the afternoon. The DHT22 sensor recorded temperatures ranging from a minimum of 32.7°C at 8:00 PM to a maximum of 37°C at 12:00 PM. In contrast, the digital reader captured temperatures ranging from a minimum of 33.1°C at 8:00 PM to a maximum of 36.8°C at 9:00 AM and again at 12:00 PM.

The humidity readings from the DHT22 sensor exhibited higher variability compared to the temperature data. Similar to the temperature readings, the DHT22 sensor generally reported lower humidity levels than the digital reader in the morning and late evening but showed closer alignment during the afternoon. Humidity levels measured by the DHT22 sensor ranged from a minimum of 55.8% at 12:00 PM to a maximum of 68.9% at 8:00 PM. In comparison, the digital reader recorded humidity levels ranging from a minimum of 59% at 5:00 PM to a maximum of 71% at 9:00 PM. These observations indicate varying humidity conditions throughout the day, with both sensors showing distinct patterns in their readings.

Table 4.20. DHT Sensor vs Digital Reader (Container 2)

May 21, 2024				
CONTAINER 2				
Time	DHT		READER	
	C°	RH%	C°	RH%
9:00 AM	35.4	59.3	36.9	62
10:00 AM	36.8	55.1	35.9	62
11:00 AM	36.4	53.6	36	62
12:00 PM	37.1	53.1	36.6	61
1:00 PM	36.5	55.2	36.1	62
2:00 PM	34.7	60.3	34.1	65
3:00 PM	35.3	58.7	35.5	62
4:00 PM	36.1	55.4	35.9	61
5:00 PM	35.8	54.3	35.4	59
6:00 PM	34.6	58.2	34.7	61
7:00 PM	33.4	62.2	34	64
8:00 PM	33	66.6	33.6	69
9:00 PM	33.6	64.6	33.3	71

Table 31 shows the comparison of temperature and humidity between the DHT sensor and digital reader for the second container. Throughout the day, the DHT22 sensor consistently provided temperature readings with slight fluctuations. Generally, in the morning, these readings were slightly lower compared to those recorded by the digital reader but showed closer alignment in the afternoon. The DHT22 sensor recorded temperatures ranging from a minimum of 33°C at 8:00 PM to a maximum of 37.1°C at 12:00 PM. In contrast, the digital reader captured temperatures ranging from a minimum of 33.3°C at 9:00 PM to a maximum of 36.9°C at 9:00 AM.

The humidity readings from the DHT22 sensor exhibited higher variability compared to the temperature data. Similar to the temperature readings, the DHT22 sensor generally reported lower humidity levels than the digital reader in the morning and late evening but showed closer alignment during the afternoon. Humidity levels measured by the DHT22 sensor ranged from a minimum of 53.1% at 12:00 PM to a maximum of 66.6% at 8:00 PM. On the other hand, the digital reader recorded humidity levels ranging from a minimum of 59% at 5:00 PM to a maximum of 71% at 9:00 PM. These observations highlight the varying humidity conditions throughout the day, with both sensors showing distinct patterns in their measurements.

Table 4.21. DHT Sensor vs Digital Reader (Container 3)

May 21, 2024				
CONTAINER 3				
Time	**DHT**		**READER**	
	C°	**RH%**	**C°**	**RH%**
9:00 AM	35.6	59.7	36.9	61
10:00 AM	37.4	54.4	36	62
11:00 AM	37.3	52.9	36	61
12:00 PM	37.5	53.1	36.6	60
1:00 PM	37.1	54.7	36.2	61
2:00 PM	35	58.9	34.2	65
3:00 PM	36	56.1	35.6	61
4:00 PM	37	54.5	35.9	61
5:00 PM	36.4	53.2	35.4	59
6:00 PM	35.1	57.3	34.6	61
7:00 PM	33.9	61.5	33.9	64
8:00 PM	33.5	65	33.8	69
9:00 PM	34.1	65.4	33.3	71

Table 32 shows the comparison of temperature and humidity between the DHT sensor and digital reader for the third container. The temperature readings from the DHT22 sensor generally followed a similar trend as those from the digital reader but consistently showed higher values during the day. The most notable difference occurred at 12:00 PM, where the DHT22 sensor recorded 37.5°C compared to the digital reader's 36.6°C. Throughout the day, the DHT22 sensor recorded temperatures ranging from 33.5°C at 8:00 PM to 37.5°C at 12:00 PM. In comparison, the digital reader's temperatures ranged from 33.3°C at 9:00 PM to 36.9°C at 9:00 AM.

The humidity readings from the DHT22 sensor displayed more variability and occasional deviations compared to the temperature data. Generally, the DHT22 sensor recorded lower humidity levels in the morning and afternoon but showed closer alignment with the digital reader in the evening. Humidity levels measured by the DHT22 sensor ranged from 52.9% at 11:00 AM to 65.4% at 9:00 PM. Meanwhile, the digital reader recorded humidity levels ranging from 59% at 5:00 PM to 71% at 9:00 PM. These observations highlight the varying humidity conditions throughout the day, with the DHT22 sensor sometimes reporting lower levels than the digital reader, particularly earlier in the day, but showing convergence in the evening hours.

The overall results indicate that the DHT22 sensors provide reliable and consistent temperature measurements that closely correlate with the digital reader across all three containers. However, the humidity measurements from the DHT22 sensors showed greater variability and occasional deviations from the digital reader. These discrepancies were more pronounced during the day, with the DHT22 sensors generally recording lower humidity levels than the digital reader.

iii. Correlations of DHT22 Sensor vs Rice Container Amount

Conducted on May 21, 2024, this test aimed to understand how the amount of rice in each container influences the temperature and humidity readings of the DHT22 sensors. Each container was filled to different levels: Container 1 was full (12 kg), Container 2 was half-full (6 kg), and Container 3 was at threshold level (3 kg).

Hourly readings over a 12-hour period of temperature and humidity were taken from the DHT22 sensors placed in each container. The data collected were analyzed to identify any correlations between the amount of rice in the containers and the sensor readings, providing insights into how varying rice levels affect the internal environment.

Table 4.22. DHT Sensor vs Level of Rice (kilogram)

May 21, 2024						
Time	CONTAINER 1 - FULL (12kg)		CONTAINER 2 - HALF (6 kg)		CONTAINER 3 - THRESHOLD (3 kg)	
	C°	RH%	C°	RH%	C°	RH%
9:00 AM	35.3	65.8	35.4	59.3	35.6	59.7
10:00 AM	36.4	58.5	36.8	55.1	37.4	54.4
11:00 AM	36.3	57.2	36.4	53.6	37.3	52.9
12:00 PM	37	55.8	37.1	53.1	37.5	53.1
1:00 PM	36.2	57.1	36.5	55.2	37.1	54.7
2:00 PM	34.5	62.6	34.7	60.3	35	58.9
3:00 PM	35.1	61.4	35.3	58.7	36	56.1
4:00 PM	35.9	58.8	36.1	55.4	37	54.5
5:00 PM	35.7	58.2	35.8	54.3	36.4	53.2
6:00 PM	34.1	62	34.6	58.2	35.1	57.3
7:00 PM	33.1	65.8	33.4	62.2	33.9	61.5
8:00 PM	32.7	68.9	33	66.6	33.5	65
9:00 PM	33.4	67.8	33.6	64.6	34.1	65.4

Table 33 shows the differences of temperature and humidity all three containers, each with certain amounts of stored rice. Across all containers, the highest temperatures were observed around midday, with Container 3 (threshold) recording the peak temperature of 37.5°C at 12:00 PM. Containers 1 (full) and 2 (half-full) displayed slightly lower maximum temperatures of 37.0°C and 37.1°C, respectively, at the same time. The temperature data suggest that the amount of rice in the container has minimal effect on the peak temperatures recorded by the DHT22 sensors. However, the empty container tended to heat up slightly more than the full and half-full containers throughout the observation period.

Humidity readings exhibited more variation across the different containers. The full container (Container 1) consistently recorded higher humidity levels compared to the half-full and empty containers. For instance, at 9:00 AM, the full container recorded a humidity level of 65.8%, while the half-full and threshold containers recorded 59.3% and 59.7%, respectively, at the same time. Throughout the day, the full container maintained higher humidity levels, reaching a peak of 68.9% at 8:00 PM, whereas the half-full and threshold containers recorded 66.6% and 65.0%, respectively, at the same hour. This pattern indicates that the presence of rice, which can absorb moisture, likely contributes to maintaining higher humidity levels in the full container compared to containers with less or no rice.

The data from this testing phase indicates that the amount of rice in the containers has a noticeable impact on the humidity readings but a minimal effect on the temperature readings recorded by the DHT22 sensors. The full container consistently showed higher humidity levels, suggesting that the rice helps retain moisture within the container. The temperature data across all containers followed a similar trend, with minor variations indicating that the amount of rice did not significantly influence temperature fluctuations within the testing period.

iv. Correlations of DHT22 Sensor on Sunny vs Rainy Weather

This test was conducted over two separate days to assess the performance of the DHT22 sensors under different weather conditions. The first set of data was collected on May 21, 2024, during sunny weather, and the second set on May 27, 2024, during rainy weather.

For each day, the DHT22 sensors in all three containers recorded temperature and humidity data every hour for a 10-hour period. This allowed for a comparative analysis of sensor performance and environmental readings under varying atmospheric conditions, highlighting the sensors' responsiveness to changes in external weather.

Table 4.23. DHT Sensor vs Weather Climate - Container 1

CONTAINER 1				
Time	May 21 - Sunny		May 27 - Rainy	
	C°	RH%	C°	RH%
11:00 AM	36.3	57.2	31	75.9
12:00 PM	37	55.8	32.3	70.6
1:00 PM	36.2	57.1	32.9	69
2:00 PM	34.5	62.6	33.6	67.7
3:00 PM	35.1	61.4	33.5	67.9
4:00 PM	35.9	58.8	32.9	68.9
5:00 PM	35.7	58.2	33.2	68.4
6:00 PM	34.1	62	31.5	71.8
7:00 PM	33.1	65.8	31.3	72.7
8:00 PM	32.7	68.9	31.7	70.8
9:00 PM	33.4	67.8	31.6	73.7

Table 34 shows the comparison of temperature and humidity for the first container between a sunny and rainy day. During sunny weather, the highest temperature recorded was 37.0°C at 12:00 PM, gradually decreasing to 32.7°C by 8:00 PM. In contrast, on the rainy day, temperature values were lower, with the highest reading of 33.6°C at 2:00 PM and subsequent decreases to 31.0°C by 11:00 AM. These observations suggest that the container's internal temperature is generally higher on sunny days compared to rainy days, illustrating the impact of external weather conditions on internal temperature fluctuations.

On the sunny day, humidity levels ranged from 55.8% at 12:00 PM to 68.9% at 8:00 PM. In contrast, the rainy day exhibited consistently higher humidity readings, ranging from 67.7% at 2:00 PM to 75.9% at 11:00 AM. This significant increase in humidity during rainy weather indicates the sensor's sensitivity to external moisture conditions, highlighting the higher ambient humidity levels experienced during rainfall. These fluctuations underscore how external environmental factors influence the moisture content within the container, affecting the humidity levels observed throughout the day.

Table 4.24. DHT Sensor vs Weather Climate - Container 2

CONTAINER 2				
Time	**May 21 - Sunny**		**May 27 - Rainy**	
	C°	**RH%**	**C°**	**RH%**
11:00 AM	36.4	53.6	30.9	73.5
12:00 PM	37.1	53.1	32.2	69.9
1:00 PM	36.5	55.2	33.5	67.5
2:00 PM	34.7	60.3	33.8	64.9
3:00 PM	35.3	58.7	33.4	66.1
4:00 PM	36.1	55.4	32.9	67.4
5:00 PM	35.8	54.3	33.2	66.8
6:00 PM	34.6	58.2	31.7	70.2
7:00 PM	33.4	62.2	31.4	70.9
8:00 PM	33	66.6	32	69.5
9:00 PM	33.6	64.6	31.8	70.9

Table 35 shows the comparison of temperature and humidity for the second container between a sunny and rainy day. During the sunny day, the highest temperature recorded inside the container was 37.1°C at 12:00 PM, gradually decreasing to 33.0°C by 8:00 PM. Conversely, on the rainy day, temperature values were consistently lower, with the highest reading of 33.8°C at 2:00 PM and the lowest reading of 30.9°C at 11:00 AM. These observations indicate that temperatures inside the container were higher during sunny weather compared to rainy weather, showcasing the sensor's ability to reflect external temperature changes effectively.

On sunny days, humidity levels inside the container were generally lower, ranging from 53.1% at 12:00 PM to 66.6% at 8:00 PM. In contrast, the rainy day exhibited higher humidity levels, ranging from 64.9% at 2:00 PM to 73.5% at 11:00 AM. The significant increase in humidity on the rainy day demonstrates the sensor's sensitivity to external humidity conditions, highlighting the higher ambient moisture levels experienced during rainfall. These variations underscore how external weather conditions influence the moisture content within the container, affecting the humidity levels observed throughout the day.

Table 4.25. DHT Sensor vs Weather Climate - Container 3

CONTAINER 3				
Time	May 21 - Sunny		May 27 - Rainy	
	C°	RH%	C°	RH%
11:00 AM	37.3	52.9	31.5	73.6
12:00 PM	37.5	53.1	32.6	69.5
1:00 PM	37.1	54.7	33.5	66.4
2:00 PM	35	58.9	34.5	63.1
3:00 PM	36	56.1	34	65
4:00 PM	37	54.5	33.5	66.3
5:00 PM	36.4	53.2	33.8	65.7
6:00 PM	35.1	57.3	32.4	72.8
7:00 PM	33.9	61.5	32.1	72.9
8:00 PM	33.5	65	32.6	72.7
9:00 PM	34.1	65.4	32.5	73.7

Table 36 shows the comparison of temperature and humidity for the third container between a sunny and rainy day. During the sunny day, the highest temperature recorded inside the container was 37.5°C at 12:00 PM, gradually decreasing to 33.5°C by 8:00 PM. In contrast, on the rainy day, temperatures were consistently lower, peaking at 34.5°C at 2:00 PM and dropping to 31.5°C at 11:00 AM. These observations clearly indicate that the container's internal temperature was higher during sunny weather compared to rainy weather, showcasing the sensor's capability to accurately reflect external temperature variations throughout the day.

On sunny days, humidity levels inside the container were generally lower, ranging from 52.9% at 11:00 AM to 65.4% at 9:00 PM. In contrast, the rainy day exhibited higher humidity levels, ranging from 63.1% at 2:00 PM to 73.7% at 9:00 PM. The significant increase in humidity on the rainy day illustrates the sensor's sensitivity to external humidity conditions, highlighting the higher ambient moisture levels experienced during rainfall. These fluctuations underscore how external weather conditions directly impact the moisture content within the container, influencing the observed humidity levels throughout the day.

The data from all three containers show consistent patterns in temperature and humidity variations under different weather conditions. The DHT22 sensors accurately detected higher temperatures and lower humidity levels on sunny days and lower temperatures and higher humidity levels on rainy days. These results affirm the sensors' reliability and responsiveness to environmental changes, providing precise monitoring of storage conditions

4.7 ISO Standards and National Standard Codes

In designing the Multi-Grain Rice Dispensing Machine, a focus on structural design and system integrations based on ISO Standards and National Standard Codes from the Philippines ensures robustness, safety, and compliance.

Frame Construction: The frame of the machine should be constructed in accordance with ISO standards such as ISO 2107 which provides International temper designation systems for wrought aluminum alloys, providing guidelines for designations and identification of aluminum alloys (Samoilova & Zamyatina, 2005). The frame must adhere to the National Structural Code of the Philippines for durability and safety, considering factors like wind loads and seismic considerations.

Cladding Materials: The cladding materials used should meet standards such as EN 15643 and ISO 15392 for sustainability in the urban tropics. Adhering to these standards ensures that the materials are environmentally friendly, durable, and suitable for the local climate conditions in the Philippines (Mathern et al., 2020).

Overall Functionality: Integrating the overall functionality of the Multi-Grain Rice Dispensing Machine must align with ISO 9001 standards for quality management systems (Martínez-Costa, 2009). Ensuring the machine's performance aligns with ISO 21001 for education organizations guarantees operational efficiency and quality standards (Kovalenko, 2020).

By incorporating these standards into the structural design and system integrations of the Multi-Grain Rice Dispensing Machine, it will not only meet international quality benchmarks but also cater to the specific needs and regulations of the Philippines' national standards, ensuring a reliable and compliant product.

Chapter 5

SUMMARY OF FINDINGS, CONCLUSIONS, AND RECOMMENDATIONS

Summary of Findings

The development process of the Multi-Grain Dispensing Machine was divided into several key stages: Design and Fabrication, Programming and Integration, and Testing and Calibration. During these stages, the machine's cabinet and dispensing mechanism were designed, the control system was programmed using Raspberry Pi OS and PyQT, and testing was performed to ensure the machine's functionality, accuracy, and reliability. The final design included an aluminum cabinet structure, a touchscreen interface, a payment system, and an SMS notification system.

The proposed Multi-Grain Dispensing Machine showcased improvements in structural design compared to existing systems. The machine was constructed with stainless steel and composite panels, ensuring compliance with ISO standards and national structural codes. This design not only enhances the machine's longevity but also ensures user safety and reliability.

The integration of advanced components, such as the Raspberry Pi 4, DHT22 sensors for temperature and humidity monitoring, and servo motors for precise dispensing, provided a competitive edge. These integrations resulted in a more efficient and user-friendly system, capable of maintaining optimal storage conditions and delivering accurate rice measurements.

The machine was designed to dispense three various types of rice, with separate containers and servo motors for each grain type. This feature addressed the need for versatility in rice dispensing, accommodating multiple rice grain varieties.

The machine included a 7-inch touchscreen LCD interface, allowing users to easily select the type and quantity of rice. Despite positive feedback on design consistency and error recovery, some areas for improvement were identified, particularly in navigation efficiency.

A secure payment system was integrated, accepting specific denominations of Philippine peso coins and bills. The system was designed to reject invalid payments and provided secure transactions, contributing to the machine's overall reliability.

The system demonstrated high efficiency in dispensing automation. The servo motors and load cells provided precise control over the amount of rice dispensed, with a Mean Absolute Percentage Error (MAPE) of 0.520% for 1000g, 0.553% for 2000g, and 1.873% for 3000g, indicating minimal deviations from target weights.

The machine was able to accurately measure and dispense rice in the desired quantities, supported by the load cell and HX711 amplifier, which ensured reliable performance in various weight settings.

The machine's design and sensor integrations ensured sustainable operation by maintaining optimal storage conditions for rice. The use of humidity and temperature sensors effectively prevented spoilage, contributing to the machine's long-term sustainability.

These findings highlight the successful development and competitive advantages of the Multi-Grain Rice Dispensing Machine with Short Messaging System Notification, addressing key problems related to structural design, system integration, feature integration, and operational efficiency.

Conclusion

The Multi-Grain Rice Dispensing Machine with Short Messaging System Notification demonstrated effective performance in automated rice dispensing. The machine accurately dispensed rice with minimal deviation, and its environmental sensors reliably monitored storage conditions. Adherence to ISO and national standards ensured the machine's structural integrity and safety. User feedback indicated overall satisfaction with the machine's functionality,although improvements to the software interface are recommended to enhance user experience. Future development should prioritize software optimization to maximize the machine's potential.

Recommendations

For future development, several enhancements are suggested to refine and validate the concept of the Multi-Grain Rice Dispensing Machine. Implementing these recommendations could further demonstrate the machine's potential efficiency and usability:

- Incorporating additional payment options and the capability to dispense change would increase its appeal to a broader customer base.
- Automating the dispensing of plastic bags could streamline operations and reduce material costs.
- Enhancing the machine's communication capabilities by enabling real-time sales data and analytics to be sent to sellers via text message could provide valuable insights for inventory and marketing strategies.
- Improving the dispensing speed and integrating a robust power backup system would enhance the machine's commercial viability, ensuring consistent operation even during power outages.
- Deploying the machine in high-traffic commercial areas such as malls could maximize its usage and profitability.

References

An, Y., Zhou, X., & Zhang, Y. (2018, January). Changes in Physicochemical, Cooking and Sensory Characteristics of Rice Shifted from Low-temperature Storage. Research Gate. Retrieved from https://www.researchgate.net/publication/326121753_Changes_in_Physicochemical_Cooking_and_Sensory_Characteristics_of_Rice_Shifted_from_Low-temperature_Storage

Ariawan, K. U., Santyadiputra, G. S., & Sutaya, I. W. (2019). Design of Hexapod Robot Movement based on Arduino Mega 2560. Journal of Physics: Conference Series, 1165, 012011. https://doi.org/10.1088/1742-6596/1165/1/012011

Bhange, S., Niphade, K., Pachorkar, T., & Pansare, A. (2021). Automatic medicine vending machine. Int J Adv Res Electron Commun Eng (IJARECE), 4(3), 703-705.

Bebon International. (2011). GB 0Cr25Ni20 stainless steel coil. Stainless Steel Coil supplier-Retrieved from https://www.bebonchina.com/ss/Industry-news/GB-0Cr25Ni20-Stainless-Steel-Coil.html

Bouchard, J., Malalgoda, M., Storsley, J., Malunga, L., Netticadan, T., & Thandapilly, SJ. (2022). Health Benefits of Cereal Grain- and Pulse-Derived Proteins. Molecules. Jun 10;27(12):3746. doi: 10.3390/molecules27123746. PMID: 35744874; PMCID: PMC9229611.

Brennan, L., Caroline, F., Eva, L.. Jenkins, B. S., Michaela, J., Eloise, F., Lukas, P., Sophie, L., Simon, L., & Karli, V. (2023). Consumer Perceptions of Food Packaging in Its Role in Fighting Food Waste. Sustainability 15, no. 3: 1917. https://doi.org/10.3390/su15031917

Carcea, M. (2021). Value of Wholegrain Rice in a Healthy Human Nutrition. Agriculture 11, no. 8: 720. https://doi.org/10.3390/agriculture11080720

Civmats Co., L. (n.d.). Austenitic stainless steel-introduction, applications, Data Sheet. civmats.com. https://civmats.com/grades/GRADES-Austenite.HTML

Cudia, C., Rivera, J., & Tullao, T. (2019). Alleviating poverty in the Philippines through entrepreneurship. DLSU Business and Economics Review, 28 (3), 121-130. Retrieved from https://animorepository.dlsu.edu.ph/faculty_research/1138

De Castro, C. J., Felix, V. R., Adante, P. M., & Young, M. N. (2020). Implementation of Computer-Based Information System on Rice Retailing Business using MS Access. 2020 IEEE 7th International Conference on Engineering Technologies and Applied Sciences (ICETAS), Kuala Lumpur, Malaysia, 2020, pp. 1-4, doi: 10.1109/ICETAS51660.2020.9484255.

Dennis, A., Wixom, B., & Roth, R. (2021). Systems Analysis and Design, 8th Edition. ISBN: 978-1-119-80379-9

Desai, S., Potdar, G., Jangle, S., & Desai, P. V. J. (2021). Smart Restaurant Menu Ordering System Using Arduino Uno. Int. J. Adv. Res. Sci. Commun. Technol, 6(1), 661-666.

Dhanuja, R., Farhana, F., & Savitha, G. (2018). Smart blind stick using Arduino. International Research Journal of Engineering and Technology (IRJET), 5(03).

Du, J., Lin, Y., Gao, Y., Tian, Y., Zhang, J., & Fang, G. (2022, January 11). Nutritional changes and early warning of moldy rice under different relative humidity and storage temperature. MDPI. https://www.mdpi.com/2304-8158/11/2/185

Ek, K., Mathern, A., Rempling, R., Karlsson, M., Brinkhoff, P., Norin, M., Lindberg, J., & Rosén, L. (2020). A harmonized method for automatable life cycle sustainability performance assessment and comparison of civil engineering works design concepts. IOP Conference Series: Earth and Environmental Science, Volume 588, 1.15 – 1.19. DOI 10.1088/1755-1315/588/5/052023

Farooq, M. S., Shamyla, R., Adnan, A., Tariq, U., & Yousaf, B. Z. (2020). Role of IoT Technology in Agriculture: A Systematic Literature Review. Electronics 9, no. 2: 319. https://doi.org/10.3390/electronics9020319

Gunawan, Gunawan & Nasution, Benny & Zarlis, M & Sari, Marliana & Lubis, Arif & Solikhun,. (2021). Design of Earthquake Early Warning System Based on Internet of Thing. Journal of Physics: Conference Series. 1830. 10.1088/1742-6596/1830/1/012010.

Günay, A., & Erbuğ, Ç. (2015). Eliciting positive user experiences with self-service kiosks: pursuing possibilities. Behaviour & Information Technology, 34(1), 81–93. https://doi.org/10.1080/0144929X.2014.937459

Henderson, H., Grace, K., Gulbransen-Diaz, N., Klaassens, B., Leong, T. W., & Tomitsch, M. (2023). From Parking Meters to Vending Machines: A Study of Usability Issues in Self-Service Technologies. International Journal of Human–Computer Interaction, 1–15. https://doi.org/10.1080/10447318.2023.2212228

Hong, Yi & Davis, Brad & Marron, J.S. & Kwitt, Roland & Singh, Nikhil & Kimbell, Julia & Pitkin, Elizabeth & Superfine, Richard & Davis, Stephanie & Zdanski, Carlton & Niethammer, Marc. (2014). Statistical Atlas Construction via Weighted Functional Boxplots. Medical image analysis. 18. 684-698. 10.1016/j.media.2014.03.001.

Huizenga, S. & Ivanova, D. (2022). THE URBAN LAB Imaginative Work in the city. Rahmawan-Huizenga and Ivanova 2022 - International Journal of Urban and Regional Research. International Journal of Urban and Regional Research. 10.1111/1468-2427.13123.

Ismail, Edzwan & Abd Gani, Shamsul Fakhar & Zainal Abidin, Amar Faiz & Ibrahim, Tengku & Semail, Saifuddin. (2020). Improvement Development of Bill Acceptor for Laundry Application with Internet-of-Thing as Notification. IOP Conference Series: Materials Science and Engineering. 846. 012006. 10.1088/1757-899X/846/1/012006.

Jie, H., Xiwen, L., Zhigang, Z., Pei, W., Jing, H., Binbin, Y., Fan, D., & Qiming Z. (2020). Positioning correction method for rice transplanters based on the attitude of the implement. Computers and Electronics in Agriculture. Volume 176. 105598. ISSN 0168-1699. https://doi.org/10.1016/j.compag.2020.105598.

Jon. (2023, December 8). Sheet Metal vs. Plate Metal: Which to choose?. Thin Metal Sales. https://www.thinmetalsales.com/blog/sheet-metal-vs-plate-metal-which-to-choose/

Khan, O., Stolte, T., Creazza, A., & Hansen, Z. (2016). Integrating product design into the supply chain. Cogent Engineering. 3. 1210478. 10.1080/23311916.2016.1210478.

Kovalenko, S, Romelashvili, O., Zborovska, T., & Blagun, O. (2020). General aspects of introduction of management systems in educational organizations in pursuance of ISO 21001:2018. Quality, standardization and certification in pharmacy

Kumar, Anil & M.C.S.Geetha, Dr & R, Dr. (2023). Research Methodology. LAP LAMBERT Academic PublishingISBN: 978-620-6-78341-1

Kumar, K. (2021). Design of vending machine through implementation of visual automata simulator and finite state machine. International Journal of Research in Circuits, Devices and Systems 2021; 2(2): 60-64

Lee, D. H. (2022). Consumers' Experiences, Opinions, Attitudes, Satisfaction, Dissatisfaction, and Complaining Behavior with Vending Machines. Vol. 16 (2003): The Journal of Consumer Satisfaction, Dissatisfaction and Complaining Behavior

Liu, L., Cui, J. Huan, Y., Zou, Z., Hu, X. & Zheng, L. (2021). A Design of Smart Unmanned Vending Machine for New Retail Based on Binocular Camera and Machine Vision. IEEE Consumer Electronics Magazine. PP. 1-1. 10.1109/MCE.2021.3060722.

Maitra, S., Akbar, H.., Marian, B., Milan, S., Peter, O., Harun, G., Koushik, B., Tanmoy, S., Preetha, B., & Jnana, B. P. (2021). Intercropping—A Low Input Agricultural Strategy for Food and Environmental Security. Agronomy 11, no. 2: 343. https://doi.org/10.3390/agronomy11020343

Mao Chen and Jaswinder Pal Singh. (2001). Computing and using reputations for internet ratings. In Proceedings of the 3rd ACM conference on Electronic Commerce (EC '01). Association for Computing Machinery, New York, NY, USA, 154–162. https://doi.org/10.1145/501158.501175

Marinelli, L. & Fiano, F. & Gregori, G. & Daniele, L. (2020). Food purchasing behavior at automatic vending machines: the role of planograms and shopping time. British Food Journal. 10.1108/BFJ-02-2020-0107.

Marshall. (2019, November 1). Galvanized steel: Types, uses, benefits. National Material Company - Steel Processing Facilities. https://www.nationalmaterial.com/galvanized-steel-types-uses-benefits/

Martínez-Costa, Micaela & Choi, Thomas & Martinez, Jose & Martínez-Lorente, Angel. (2009). ISO 9000/1994, ISO 9001/2000 and TQM: the performance debate revisited. Journal of Operations Management - J OPER MANAG. 27. 495-511. 10.1016/j.jom.2009.04.002.

Mohapatra, P & Sahu, B. (2022). Designer Rice in Quest of High Grain Filling. 10.1007/978-3-030-67897-5_13.

Munz, J., and Schuele, H. (2022). Influencing the Success of Precision Farming Technology Adoption—A Model-Based Investigation of Economic Success Factors in Small-Scale Agriculture. Agriculture 12, no. 11: 1773. https://doi.org/10.3390/agriculture12111773

Mulford Plastics. (n.d.). The benefits & uses of ACP - Mulford Plastics NZ. Retrieved from https://www.mulfordplastics.co.nz/blog/the-benefits-of-aluminium-composite-panels/

Naeini, H. & Mostowfi, S. (2015). Using QUIS as a Measurement Tool for User Satisfaction Evaluation (Case Study: Vending Machine). 2015. 14-23. 10.5923/j.ijis.20150501.03.

Powers, L. (2024, February 26). Stainless Steel Characteristics & Properties. Eagle Stainless. https://eagletube.com/about-us/news/stainless-steel-characteristics/

Rangaswamy, E., Nawaz, N. & Changzhuang, Z. (2022). The impact of digital technology on changing consumer behaviours with special reference to the home furnishing sector in Singapore. Humanit Soc Sci Commun 9, 83. https://doi.org/10.1057/s41599-022-01102-x

Raposo, A., Perez, E., Sanjuan, E., Saavedra, P., Millan, R., Carrascosa, C. (2018). Vending machines and university students' consumption trends. Journal of Food & Nutrition Research, 2018, Vol 57, Issue 3, p295

Samoilova, O.V., Zamyatina, O.V. (2005). Activity and standards of ISO and IEC in the field of corrosion and corrosion protection. Prot Met 41, 177–186 . https://doi.org/10.1007/s11124-005-0025-0

Schwingshackl, L., Schwedhelm, C., Hoffmann, G., Lampousi, A. M., Knüppel, S., Iqbal, K., Bechthold, A., Schlesinger, S., & Boeing, H. (2017). Food groups and risk of all-cause mortality: a systematic review and meta-analysis of prospective studies. Am J Clin Nutr. 2017 Jun;105(6):1462-1473. doi: 10.3945/ajcn.117.153148. Epub 2017 Apr 26. PMID: 28446499.

Sheth Insulations PVT. LTD. (2020, June). Aluminum-Stucco-Sheet.

Shivananju, B. N., Hoh, H. Y., Yu, W., & Bao, Q. (2019). Optical biochemical sensors based on 2D materials. Fundamentals and Sensing Applications of 2D Materials, 379–406. https://doi.org/10.1016/b978-0-08-102577-2.00010-5

Sibanda, V., Munetsi, L., Mpofu, K., Murena, E., & Trimble, J. (2020). Design of a high-tech vending machine. Procedia CIRP, 91, 678–683. doi:10.1016/j.procir.2020.04.133

Song, W. Wan, Y. Zhou, B. Kuang, F. Xiong W. Liao J.; Zhu, D. & Zhang S. (2023). DESIGN AND EXPERIMENT OF COMBINED CAVITY-TYPE PRECISION HOLE-DROP SEED-METERING DEVICE FOR RICE. INMATEH - Agricultural Engineering, 2023, Vol 71, Issue 3, p25. 10.35633/inmateh-71-02

Sood, A. & Aggarwhal, N. (2021). Annapurti - GrainATM. World Food Programme (WFP): Innnovation Accelerator. Retrieved from innovation.wfp.org/project/annapurti-grainatm

Su, Ning & He, Jiyin & Liu, Yiqun & Zhang, Min & Ma, Shaoping. (2018). User Intent, Behaviour, and Perceived Satisfaction in Product Search. WSDM '18: Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining. 547-555. 10.1145/3159652.3159714.

Surana, H., Jha, S. K., Agarwal, H., Goyal, P., & Dangi, P. (2021). Smoke detector system with GSM module. International Journal of Technical Research & Science, 18–21. https://doi.org/10.30780/ijtrs.v06.i08.004

Vending Australia. (2020). Vending Machine Parts: A Closer Look into a Vending Machine. https://www.tcnvending.com.au/blog/vending-machine-parts-closer-look/#:~:text=Manufacturers%20use%20galvanized%20steel%20so,rust%20reduces%20the%20material’s%20mass.

Wang, H., Ab Gani, M. A., & Liu, C. (2023). Impact of Snack Food Packaging Design Characteristics on Consumer Purchase Decisions. SAGE Open, 13(2). https://doi.org/10.1177/21582440231167109

Wang, M., Yao, J. A reliable location design of unmanned vending machines based on customer satisfaction. Electron Commer Res 23, 541–575 (2023). https://doi.org/10.1007/s10660-021-09479-8

Yuliandoko, H., Panduardi, F., Lusi, N. & Wibowo, S. "Precision Rice Vending Machine by Using Multiple Load Cell and IoT Based," 2023 International Conference on Computer Science, Information Technology and Engineering (ICCoSITE), Jakarta, Indonesia, 2023, pp. 296-300, doi: 10.1109/ICCoSITE57641.2023.10127836.

Zhao, Hui-Liang & Yao, Xuemei & Liu, Zhenghong & Yang, Qin. (2021). Impact of Pricing and Product Information on Consumer Buying Behavior With Customer Satisfaction in a Mediating Role. Frontiers in Psychology. 12. 10.3389/fpsyg.2021.720151.

Zhiliang, K., Lijia, X., Chen, L., Tangui, H., & Nocklos, T. (2016). Design and Implementation of an Automatic Rice-Selling Machine Based on Fuzzy Control. Journal of Engineering Science and Technology Review 9, no. 1 (2016): 95-101.

APPENDICES

Appendix A. Cost-Benefit Analysis

Table A.1. Overall List of Materials and Cost

Components Cost			
Materials	**Quantity**	**Price**	**Total**
Raspberry Pi 4 Model B (2GB)	1	3624	3624
MG958 Digital Servo Motor 15Kg	3	533	1599
HX711 Load Sensor + 5Kg Load Cell	1	205	205
DHT22 Temperature/Humidity Sensor	3	200	600
ARD-Infrared (IR) Sensor	1	57	57
801S Vibration Shock Sensor	1	158	158
Raspberry Pi 7in LEDTouch Screen	1	2078	2078
GSM Module Sim900A	1	349	349
12V 5A Power Supply + Cord	1	545	545
Relay Module and Channel	8	21.25	170
Alan Universal Coin Slot	1	280	280
TB74 Pulse Bill Acceptor	1	3591	3591
Raspberry Pi 4 USB-C Charger	1	229	229
Raspberry PiT-Cobbler GPIO Breakout	1	250	250
DC-DC Buck Converter	1	50	50
Micro SD Card 128GB	1	609	609
Proto Board	1	192	192
20AWG 50 Meters 5 Color Wire	1	449	449
TB 3813P (Terminal Block 3P)	12	21	252
TB 3812P (Terminal Block 2P)	20	22	440
Terminal Block (12 lines)	1	85	85
Header Pins	1	107	107
Videoke Button	1	17	17

Structure Cost			
Materials	**Quantity**	**Price**	**Total**
Stainless Steel Cabinet	1	1500	1500
Acrylic Glass Panels	3	223	669
Composite Panels	3	400	1200
Metal Bowl 20cm	1	98	98
Metal Hook	1	26	26
Cabinet Lock	2	39	78
Rubber Feet	4	13	52
Labor Cost			
Labor Cost	1	2500	2500
Total Machine Cost			**22,059**

Table A.2. Utility Cost of the Machine

Utility Cost			
Electricity	30	6.714	201.42

Table A.3. Return of Investment

Return on Investment	
Machine Cost	₱22,059
Utility Cost (Montly Recurring)	₱201.42
Cost per Kilo	₱60
Selling price per Kilo	₱70
Average monthly sales	810 kg

Return of Investment for Machine Cost

$$\frac{\text{Machine cost}}{\text{Selling Price per kilo} - \text{Cost per kilo}} \times \text{Average monthly rice sales}$$

$$\frac{₱22{,}059}{₱70/\text{kg} - ₱60/\text{kg}} \times 810\ \text{kg}$$

37.77 months (approximately 3.15 years)

Break-even for Utility

$$\frac{\text{Monthly Recurring Fees}}{\text{Selling Price per Kilo} - \text{Cost per Kilo}}$$

$$\frac{₱201.42}{₱70/\text{kg} - ₱60/\text{kg}}$$

402.84 kg

Depreciation Expense

$$\frac{\text{Machine Cost} - \text{Salvage Value}}{\text{Useful Life}}$$

$$(\frac{₱\,22{,}059 - ₱\,12{,}000}{10\ \text{years}})$$

₱ 1,005.90

Appendix B. Survey Questionnaire for User Interface and Ease of Use

Functional Suitability

This characteristic represents the degree to which a product or system provides functions that meet stated and implied needs when used under specified conditions.

The software performs all the tasks and functions I need it to do for my work. *
(Ginagampanan ng software ang lahat ng tungkulin na kailangan ko para sa aking pakinabang)

	1	2	3	4	
Strong Disagree (Lubos na hindi sumasang-ayon)	○	○	○	○	Strongly Agree (Lubos na sumasangayon)

The software offers all the features and functionalities advertised or documented. *
(Inaalok ng software ang lahat ng functionalities nito na ayon sa naidokumento dito)

	1	2	3	4	
Strong Disagree (Lubos na hindi sumasang-ayon)	○	○	○	○	Strongly Agree (Lubos na sumasangayon)

The software provides the necessary tools and options to efficiently complete my tasks *
(Ang software ay nagbibigay ng mga kinakailangang tool at opsyon upang mahusay na makumpleto ang aking mga gawain)

	1	2	3	4	
Strong Disagree (Lubos na hindi sumasang-ayon)	○	○	○	○	Strongly Agree (Lubos na sumasangayon)

The outputs generated by the software are accurate and consistent with expectations. *
(Ang mga output na ginawa ng software ay tumpak at tugma sa mga inaasahan)

	1	2	3	4	
Strong Disagree (Lubos na hindi sumasang-ayon)	○	○	○	○	Strongly Agree (Lubos na sumasangayon)

There are minimal workarounds needed to complete tasks using the software. *
(Kaunti lamang ang mga workarounds na kailangan upang makumpleto ang mga gawain gamit ang software)

	1	2	3	4	
Strong Disagree (Lubos na hindi sumasang-ayon)	○	○	○	○	Strongly Agree (Lubos na sumasangayon)

Performance Efficiency

This characteristic represents the degree to which a product performs its functions within specified time and throughput parameters and is efficient in the use of resources (such as CPU, memory, storage, network devices, energy, materials...) under specified conditions.

The software responds quickly and performs tasks without significant delays. *
(Ang software ay tumutugon nang mabilis at gumaganap ng mga gawain nang walang makabuluhang mga pagkaantala.)

	1	2	3	4	
Strong Disagree (Lubos na hindi sumasang-ayon)	○	○	○	○	Strongly Agree (Lubos na sumasangayon)

The software utilizes system resources (memory, CPU) efficiently. *
(Ang software ay mahusay na gumagamit ng mga mapagkukunan ng sistema (memorya, CPU).)

	1	2	3	4	
Strong Disagree (Lubos na hindi sumasang-ayon)	○	○	○	○	Strongly Agree (Lubos na sumasangayon)

The software maintains performance even under heavy workloads. *
(Ang software ay nagpapanatili ng pagganap kahit sa ilalim ng mabigat na mga workload.)

	1	2	3	4	
Strong Disagree (Lubos na hindi sumasang-ayon)	○	○	○	○	Strongly Agree (Lubos na sumasangayon)

Startup times and loading times for the software are reasonable. *
(Ang oras ng pagsisimula at pag-load ng software ay makatwiran.)

	1	2	3	4	
Strong Disagree (Lubos na hindi sumasang-ayon)	○	○	○	○	Strongly Agree (Lubos na sumasangayon)

The software operates smoothly without frequent freezes or crashes. *
(Ang software ay gumagana nang maayos nang walang madalas na pagfreeze at pagcrash.)

	1	2	3	4	
Strong Disagree (Lubos na hindi sumasang-ayon)	○	○	○	○	Strongly Agree (Lubos na sumasangayon)

Interaction Capability

Degree to which a product or system can be interacted with by specified users to exchange information ia the user interfaceto complete specific tasks in a variety of contexts of use.

The user interface of the software is easy to understand and navigate. *
(Ang user interface ng software ay madaling maunawaan at i-navigate.)

	1	2	3	4	
Strong Disagree (Lubos na hindi sumasang-ayon)	○	○	○	○	Strongly Agree (Lubos na sumasangayon)

The software provides clear and concise instructions and error messages. *
(Ang software ay nagbibigay ng malinaw at maigsi na mga instruksyon at mensahe ng error.)

	1	2	3	4	
Strong Disagree (Lubos na hindi sumasang-ayon)	○	○	○	○	Strongly Agree (Lubos na sumasangayon)

The software offers user-friendly customization options for interface preferences. *
(Ang software ay nag-aalok ng mga user-friendly na opsyon sa pagpapasadya para sa mga kagustuhan sa interface.)

	1	2	3	4	
Strong Disagree (Lubos na hindi sumasang-ayon)	○	○	○	○	Strongly Agree (Lubos na sumasangayon)

The software provides helpful feedback during interaction to guide user actions. *
(Ang software ay nagbibigay ng kapaki-pakinabang na feedback sa panahon ng pakikipag-ugnayan upang gabayan ang mga aksyon ng gumagamit.)

	1	2	3	4	
Strong Disagree (Lubos na hindi sumasang-ayon)	○	○	○	○	Strongly Agree (Lubos na sumasangayon)

The software utilizes consistent design principles throughout the interface. *
(Ang software ay gumagamit ng pare-parehong mga prinsipyo ng disenyo sa buong interface.)

	1	2	3	4	
Strong Disagree (Lubos na hindi sumasang-ayon)	○	○	○	○	Strongly Agree (Lubos na sumasangayon)

Reliability

Degree to which a system, product or component performs specified functions under specified conditions for a specified period of time.

The software utilizes consistent design principles throughout the interface. *
(Ang software ay gumagana nang tuloy-tuloy at mapagkakatiwalaan para sa pang-araw-araw na mga gawain.)

	1	2	3	4	
Strong Disagree (Lubos na hindi sumasang-ayon)	○	○	○	○	Strongly Agree (Lubos na sumasangayon)

The software experiences minimal downtime or unexpected crashes. *
(Ang software ay may kaunting downtime o hindi inaasahang pagbagsak.)

	1	2	3	4	
Strong Disagree (Lubos na hindi sumasang-ayon)	○	○	○	○	Strongly Agree (Lubos na sumasangayon)

Data entered or stored in the software is accurate and remains secure. *
(Ang data na ipinasok o nakaimbak sa software ay tumpak at nananatiling ligtas.)

	1	2	3	4	
Strong Disagree (Lubos na hindi sumasang-ayon)	○	○	○	○	Strongly Agree (Lubos na sumasangayon)

The software recovers gracefully from errors without significant data loss. *
(Ang software ay nakaka-recover nang maayos mula sa mga error nang walang makabuluhang pagkawala ng data.)

	1	2	3	4	
Strong Disagree (Lubos na hindi sumasang-ayon)	○	○	○	○	Strongly Agree (Lubos na sumasangayon)

The software performs reliably over extended periods of use without degradation in performance. *
(Ang software ay maaasahang gumagana sa mahabang panahon nang hindi bumababa ang pagganap.)

	1	2	3	4	
Strong Disagree (Lubos na hindi sumasang-ayon)	○	○	○	○	Strongly Agree (Lubos na sumasangayon)

Maintainability

This characteristic represents the degree of effectiveness and efficiency with which a product or system can be modified to improve it, correct it or adapt it to changes in environment, and in requirements.

I feel confident that using this software will not cause any data loss or system crashes. *
(Ako ay may kumpiyansa na ang paggamit ng software na ito ay hindi magdudulot ng anumang pagkawala ng data o pagcrash ng sistema.)

	1	2	3	4	
Strong Disagree (Lubos na hindi sumasang-ayon)	○	○	○	○	Strongly Agree (Lubos na sumasangayon)

The software has adequate safeguards in place to prevent unauthorized access and malicious attacks. (Ang software ay may sapat na mga proteksyon upang maiwasan ang hindi awtorisadong access at mapanirang mga pag-atake.) *

	1	2	3	4	
Strong Disagree (Lubos na hindi sumasang-ayon)	○	○	○	○	Strongly Agree (Lubos na sumasangayon)

The software provides clear warnings and error messages to help me avoid making mistakes with potentially negative consequences. *
(Ang software ay nagbibigay ng malinaw na babala upang tulungan akong maiwasan ang pagkakamali na maaaring magdulot ng negatibong epekto.)

	1	2	3	4	
Strong Disagree (Lubos na hindi sumasang-ayon)	○	○	○	○	Strongly Agree (Lubos na sumasangayon)

I trust the software to handle sensitive information securely and reliably. *
(Tiwala ako sa software na pangalagaan ang sensitibong impormasyon nang ligtas at maasahan.)

	1	2	3	4	
Strong Disagree (Lubos na hindi sumasang-ayon)	○	○	○	○	Strongly Agree (Lubos na sumasangayon)

The software operates consistently without unexpected behavior that could compromise safety. *
(Ang software ay kumikilos nang tuloy-tuloy nang walang inaasahang mga pag-uugali na maaaring magdulot ng panganib sa kaligtasan.)

	1	2	3	4	
Strong Disagree (Lubos na hindi sumasang-ayon)	○	○	○	○	Strongly Agree (Lubos na sumasangayon)

Appendix C. Survey Questionnaire for Material Quality and Design Structure

Republic of the Philippines
POLYTECHNIC UNIVERSITY OF THE PHILIPPINES
Office of the Vice President for Academic Affairs
College of Engineering
COMPUTER ENGINEERING (CPE) DEPARTMENT

QUESTIONNAIRE:

Evaluator: DANILO R. CABATAN **Date:**

I. Frame Construction (Galvanized Steel)

Criterion	Description	Rating (1-5)	Comments
a. Material Selection	Is galvanized steel appropriate for the anticipated load and environment?	4	
b. Frame Design	Is the frame design structurally sound for the weight of rice and machine components? Consider factors like beam sizes, connection points, and overall stability.	5	
c. Fabrication Quality	Are the welds, connections, and overall fabrication of the frame free from defects and meet industry standards?	4	
d. Load Testing	Has the frame undergone any load testing to verify its ability to withstand anticipated weight and stresses? (Consider static and dynamic loads)	5	

II. Cladding Materials

Criterion	Description	Rating (1-5)	Comments
a. Suitability for Purpose	Are aluminum composite panels and aluminum stucco sheets suitable for the intended function of the cladding (protection, aesthetics)?	5	
b. Structural Contribution	Do the cladding materials contribute significantly to the overall structural integrity of the machine, or are they primarily for aesthetics?	4	
c. Attachment Method	Is the method of attaching the cladding to the frame secure and able to withstand potential forces (e.g., user interaction, accidental bumps)?	5	
d. Material Testing	Have the cladding materials been tested for their suitability in the intended environment (e.g., moisture resistance, impact resistance)?	4	

III. Overall Functionality

Criterion	Description	Rating (1-5)	Comments
a. Accessibility	Does the design allow for easy access to internal components for maintenance and refilling?	5	
b. User Interaction	Is the machine stable and user-friendly during rice dispensing operations?	5	
c. Operational Safety	Does the overall design minimize potential safety hazards (e.g., sharp edges, tipping hazards)?	5	

Appendix D. User Manual (for Seller and Costumers) – Trifold Brochure

FOR THE SELLER

Turn On the Machine
- Seller must first plug the machine on a nearby power source. The machine will automatically turn on after plugging.

For Refilling
- Unlock the container shutters on top, then open to check or refill rice. Each container has a maximum capacity to hold 12 kilograms of rice.

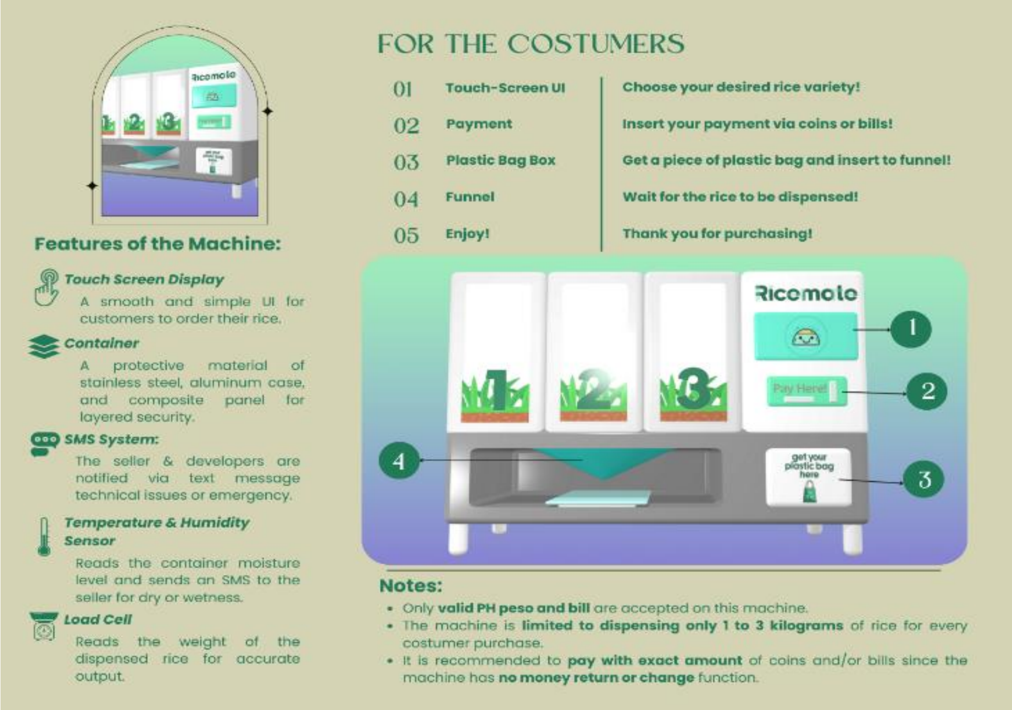
Accessing the Cash Box
- The cash box is situated on the lower right corner of the machine. Seller must first unlock the box using a physical key to open the register.

For Dispensing Errors or Releasing All Rice
- In case of dispensing malfunctions, an emergency button is situated on top with green color. Pressing it will release all rice to the dispensing funnel. **SELLER MUST FIRST ACCESS ADMIN MODE OF MACHINE VIA PANEL TO ACTIVATE BUTTON.**

Ricemate

A Multi-Grain Rice Dispensing Machine

USER MANUAL



Features of the Machine:

Touch Screen Display
A smooth and simple UI for customers to order their rice.

Container
A protective material of stainless steel, aluminum case, and composite panel for layered security.

SMS System:
The seller & developers are notified via text message technical issues or emergency.

Temperature & Humidity Sensor
Reads the container moisture level and sends an SMS to the seller for dry or wetness.

Load Cell
Reads the weight of the dispensed rice for accurate output.

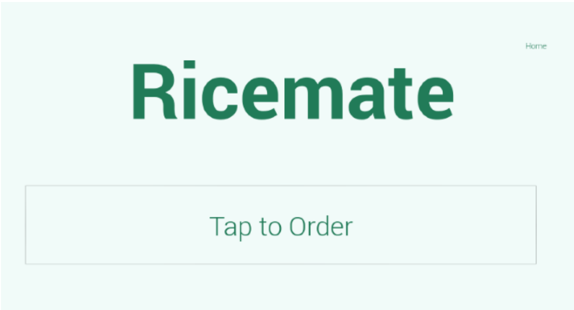
FOR THE COSTUMERS

01	**Touch-Screen UI**	**Choose your desired rice variety!**
02	**Payment**	**Insert your payment via coins or bills!**
03	**Plastic Bag Box**	**Get a piece of plastic bag and insert to funnel!**
04	**Funnel**	**Wait for the rice to be dispensed!**
05	**Enjoy!**	**Thank you for purchasing!**

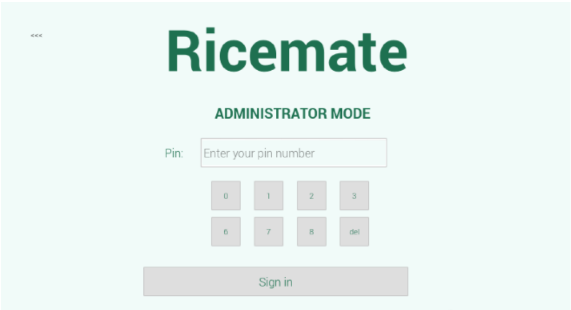
Notes:
- Only **valid PH peso and bill** are accepted on this machine.
- The machine is **limited to dispensing only 1 to 3 kilograms** of rice for every costumer purchase.
- It is recommended to **pay with exact amount** of coins and/or bills since the machine has **no money return or change** function.

Appendix E. Deployed Instructional Manual - Poster

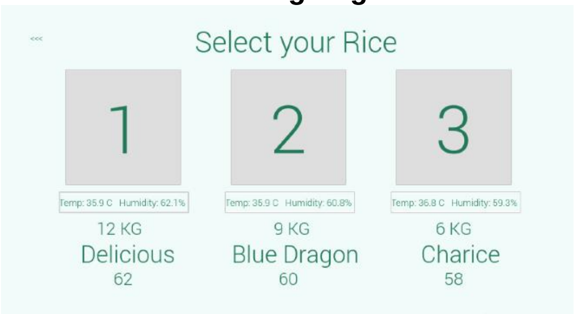
Appendix F. Program Screenshots



Landing Page

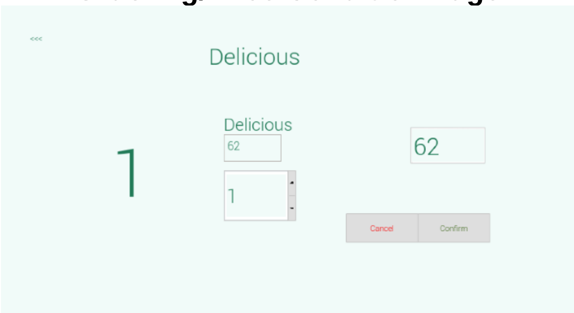


Administrator Page

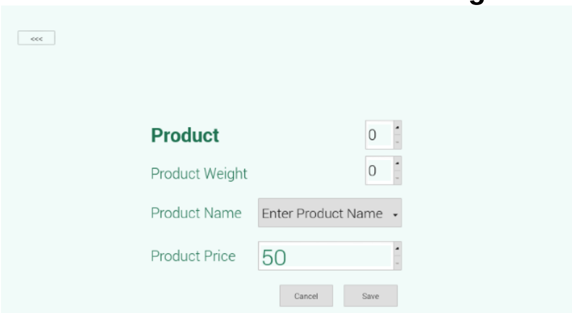


Ordering/ Rice Condition Page

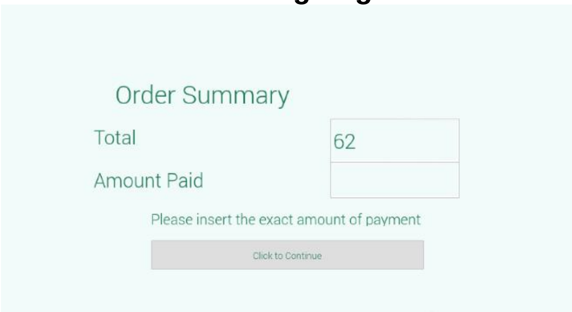
Administrator Main Menu Page



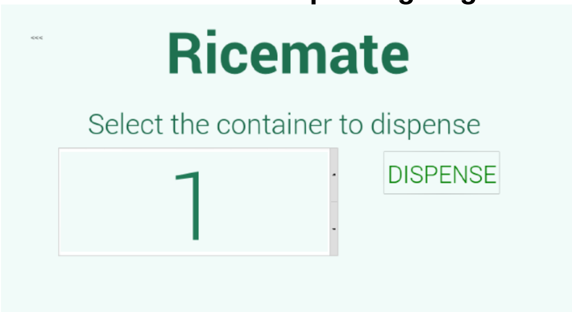
Ordering Page



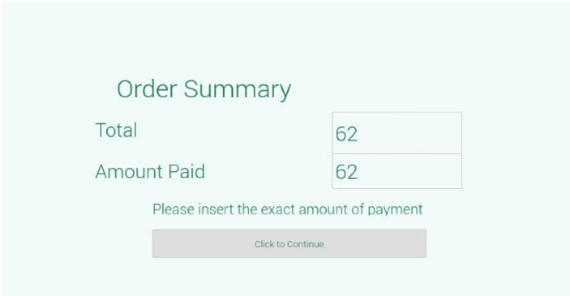
Product Detail Updating Page



Payment Page

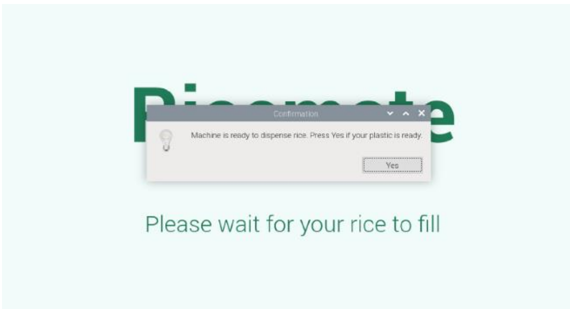


Manual Dispensing Page

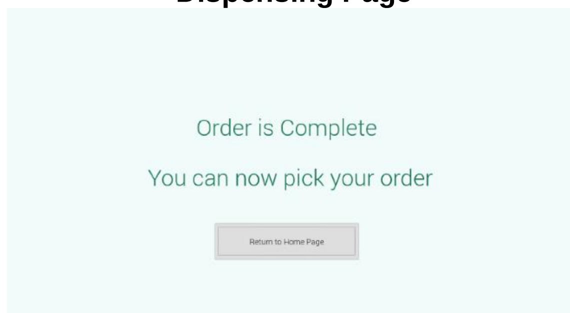


Payment Confirmation Page

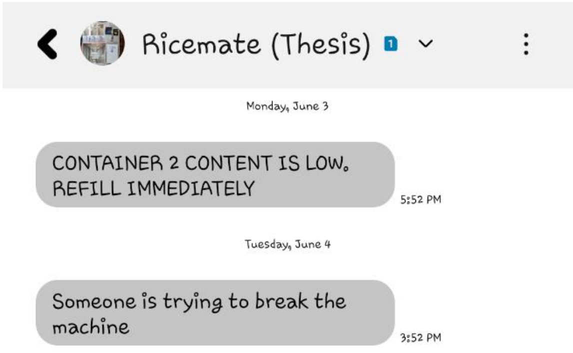
Dispensing Page



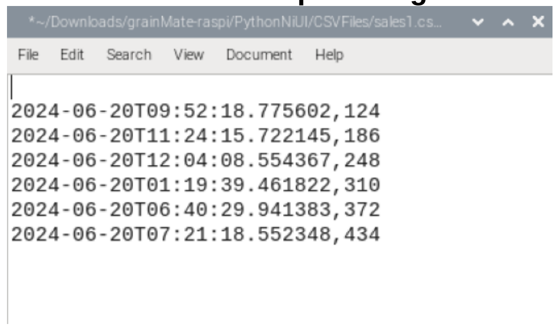
Plastic Presence Confirmation



Order Complete Page

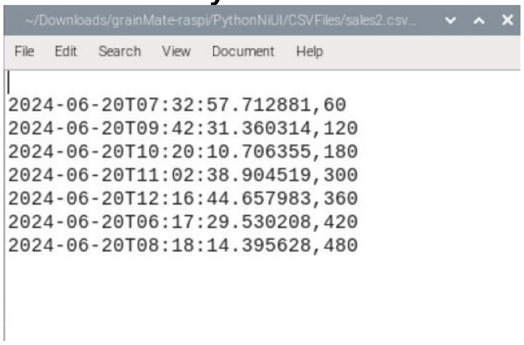


Security Breach Notification



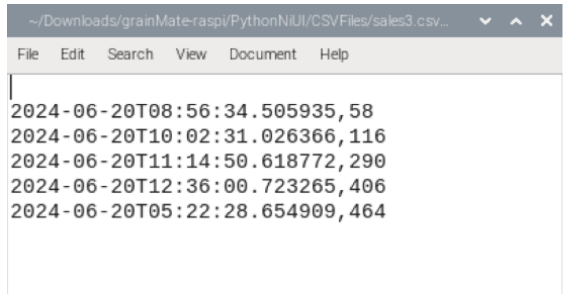

```
2024-06-20T09:52:18.775602,124
2024-06-20T11:24:15.722145,186
2024-06-20T12:04:08.554367,248
2024-06-20T01:19:39.461822,310
2024-06-20T06:40:29.941383,372
2024-06-20T07:21:18.552348,434
```

Sales Summary of Container 1



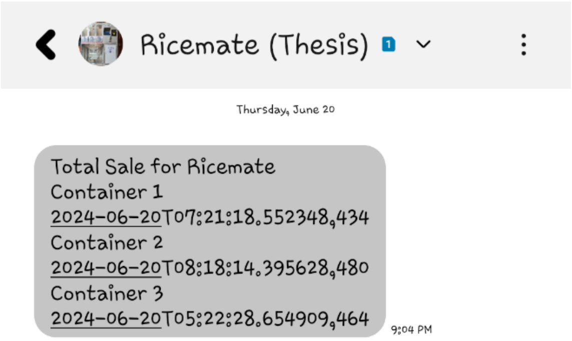

```
2024-06-20T07:32:57.712881,60
2024-06-20T09:42:31.360314,120
2024-06-20T10:20:10.706355,180
2024-06-20T11:02:38.904519,300
2024-06-20T12:16:44.657983,360
2024-06-20T06:17:29.530208,420
2024-06-20T08:18:14.395628,480
```

Sales Summary of Container 2




```
2024-06-20T08:56:34.505935,58
2024-06-20T10:02:31.026366,116
2024-06-20T11:14:50.618772,290
2024-06-20T12:36:00.723265,406
2024-06-20T05:22:28.654909,464
```

Sales Summary of Container 3



Sales Summary Notification

Appendix G. Source Code

```
def vibrationSensing(self):
    print("Vending machine monitor started.")
    self.vibration_detected = multiprocessing.Value('i', 0)
    self.vibration_process =
multiprocessing.Process(target=read_vibration,
args=(self.vibration_detected,))
        # Start the vibration sensor reading process
    self.vibration_process.daemon = True
    self.vibration_process.start()
        try:
        while True:
        # Check if vibration is detected
        if self.vibration_detected.value == 1:
        print("Vibration detected!")
        gsmNotificationForVibration()
        # Add your code to take action when vibration is
detected
        # For demonstration, let's print a message
        time.sleep(1)  # Delay to avoid repeated action
        else:
        # Your code for touchscreen interaction can be placed
here
        # For demonstration, let's print a message
        print("Touchscreen panel is idle.")
        time.sleep(1)  # Adjust the delay as needed

        except KeyboardInterrupt:
      print("\nExiting...")
        finally:
      #self.vibration_process.terminate()
      print("thread stop")
      GPIO.cleanup()

def displayHomePage(self):

        self.read_vibration_sensing =
multiprocessing.Process(target=self.vibrationSensing)
        self.read_vibration_sensing.start()

def verifyAccount(self):
        password = self.adminPasswordLineEdit2.text()
        with open('CSVFiles/password.csv', 'r') as csv:
        data = [[x.strip() for x in line.strip().split(',')] for line in
csv.readlines()][-1]
        pin = data[0]
        if password == pin:
      print("Logged In")
      self.setAdminMainMenu()

        else:
      invalidMessage = QMessageBox(self)
      invalidMessage.setWindowTitle("Invalid Input")
      invalidMessage.setText("You entered an invalid account
username or password")
      invalidMessage.setStandardButtons(QMessageBox.Ok)
      invalidMessage.setIcon(QMessageBox.Warning)
        asdf = invalidMessage.exec()
      self.adminPasswordLineEdit2.clear()
      print("Invalid username or password")
def setOrderPage(self): #MainMenue page
        try:
       self.read_vibration_sensing.terminate()
        except Exception as e:
        print(e)

with open('CSVFiles/product1Weight.csv', 'r') as csv:
        data = [[x.strip() for x in line.strip().split(',')] for line in
csv.readlines()][-1]
        loadProductWeight1 = data[0]
        self.orderWeight.setText(loadProductWeight1 + " KG")

        with open('CSVFiles/product2Weight.csv', 'r') as csv:
        data = [[x.strip() for x in line.strip().split(',')] for line in
csv.readlines()][-1]
        loadProductWeight2 = data[0]
        self.orderWeight2.setText(loadProductWeight2 + " KG")

        with open('CSVFiles/product3Weight.csv', 'r') as csv:
        data = [[x.strip() for x in line.strip().split(',')] for line in
csv.readlines()][-1]
        loadProductWeight3 = data[0]
        self.orderWeight3.setText(loadProductWeight3 + " KG")

def coins(self):
try:
        setup()
        self.loop()
        except KeyboardInterrupt:
        GPIO.cleanup()
def loop(self):
        while True:
        self.totalPayAmount = self.pulse_count_bill +
self.pulse_count_coin
        # BILL PHASE
        if GPIO.input(self.bill_pin) == GPIO.LOW:
                if time.time() - self.start_time_bill >
self.debounce_delay:
                self.pulse_count_bill += 10
                self.start_time_bill = time.time()

            # Print total amount in real-time
                print("Total Amount Inserted:",
self.totalPayAmount)

            QtWidgets.QApplication.processEvents()

        self.amountCoinPaidTextEdit.setText(str(self.totalPayA
mount))

        # COIN PHASE
        if GPIO.input(self.coin_pin) == GPIO.LOW:
                if time.time() - self.start_time_coin >
self.debounce_delay:
                self.pulse_count_coin += 1
                self.start_time_coin = time.time()

            # Print total amount in real-time
                print("Total Amount
Inserted:",self.totalPayAmount)

            QtWidgets.QApplication.processEvents()

        self.amountCoinPaidTextEdit.setText(str(self.totalPayA
mount))

        # CHECK FOR "YES" COMMAND

      if int(self.textEdit.text()) == int(self.totalPayAmount):
        print("Total Amount Inserted (Bill + Coin):",
self.totalPayAmount)
        print("Tapos na ey")

        QtWidgets.QApplication.processEvents()

        self.amountCoinPaidTextEdit.setText(str(self.totalPayA
mount))
        self.pulse_count_bill = 0
        self.pulse_count_coin = 0
                # You may choose to remove or reduce this
delay
        time.sleep(1)
        break
        else:
                pass
def verifyInputPrice(self):
        toPayAmount = self.sender()
        toPayAmount = self.textEdit.text()
        PayedAmount = self.sender()
        PayedAmount = self.amountCoinPaidTextEdit.text()
        if toPayAmount == PayedAmount:
    self.riceDispensing()
        else:
    print("Failed to Pay amount")
def dispensingMessage(self):
        with open('CSVFiles/product1Weight.csv', 'r') as csv:
```

```
            data = [[x.strip() for x in line.strip().split(',')] for line in
csv.readlines()][-1]
        loadProductWeight1 = data[0]
        print(loadProductWeight1 + " KG")

            with open('CSVFiles/product2Weight.csv', 'r') as csv:
            data = [[x.strip() for x in line.strip().split(',')] for line in
csv.readlines()][-1]
        loadProductWeight2 = data[0]
        print(loadProductWeight2 + " KG")
with open('CSVFiles/product3Weight.csv', 'r') as csv:
            data = [[x.strip() for x in line.strip().split(',')] for line in
csv.readlines()][-1]
        loadProductWeight3 = data[0]
        print(loadProductWeight3 + "KG")            reminder =
QMessageBox(self)
      reminder.setWindowTitle("Confirmation")
      reminder.setText("Machine is ready to dispense rice. Press Yes
if your plastic is ready.")
      reminder.setStandardButtons(QMessageBox.Yes)
      reminder.setIcon(QMessageBox.Information)
            button = reminder.exec()
            if button == QMessageBox.Yes:
          print("Yes button clicked!")
            # Assuming self.selectContainerSpinBox is a valid
QSpinBox object.
if self.selectedProduct == 1:  # Dispense 1st Container
            readWeight1 = int(loadProductWeight1)
            print(f"Selected product: {self.selectedProduct}")
            amountSpinBox = self.productSpinBox1.value()
            totalAmountSpinBox = int(amountSpinBox)
            print(f"Total amount selected: {totalAmountSpinBox}")
            print("Dispensing from the 1st container")
            if totalAmountSpinBox == 1:
              if readWeight1 == 12:
                dispense_Runner(840, 18)
              #elif readWeight1 == 8:
                #dispense_Runner(830, 18)
              #elif readWeight1 == 5:
                #dispense_Runner(830, 18)
              else:
                dispense_Runner(840, 18)
              self.thankYouCard()
            elif totalAmountSpinBox == 2:
              if readWeight1 == 12:
                dispense_Runner(1750, 18)
              #elif readWeight1 == 10:
                #dispense_Runner(1750, 18)
              #elif readWeight1 == 8:
                #dispense_Runner(1750, 18)
              #elif readWeight1 == 6:
                #dispense_Runner(1750, 18)
              else:
                dispense_Runner(1750, 18)
              self.thankYouCard()
            elif totalAmountSpinBox == 3:
              print(f"Dispensing 3 units with load weight
{readWeight1}")
              if readWeight1 == 7:
                dispense_Runner(2850, 18)
              #elif readWeight1 == 6:
                #control_servo(3, 16.75)
              #elif readWeight1 == 5:
                #control_servo(3, 16.75)
              else:
                dispense_Runner(2850, 18)
              self.thankYouCard()
            else:
              print("Invalid amount for 1st container")

            self.weightSubtracter1()
elif self.selectedProduct == 2:  # Dispense 2nd Container
            print(f"Selected product: {self.selectedProduct}")
            amountSpinBox = self.productSpinBox2.value()
            totalAmountSpinBox = int(amountSpinBox)
            print(f"Total amount selected: {totalAmountSpinBox}")
            readWeight2 = int(loadProductWeight2)
            print(readWeight2)
            print("Dispensing from the 2nd container")
            if totalAmountSpinBox == 1:
              if readWeight2 == 12:
                dispense_Runner(900, 23)
              #elif readWeight2 == 7:
                #dispense_Runner(850, 23)
              #elif readWeight2 == 6:
                #dispense_Runner(850, 23
              #elif readWeight2 == 5:
                #dispense_Runner(900, 23)
              #elif readWeight2 == 4:
                #dispense_Runner(870, 23)
              else:
                dispense_Runner(880, 23)
              self.thankYouCard()
            elif totalAmountSpinBox == 2:
              print(f"Dispensing 2 units with load weight
{readWeight2}")
              if readWeight2 == 12:
                dispense_Runner(1800, 23)
              #elif readWeight2 == 7:
                #dispense_Runner(1820, 23)
              #elif readWeight2 == 5:
                #dispense_Runner(1830, 23)
              else:
                dispense_Runner(1810, 23)
              self.thankYouCard()
            elif totalAmountSpinBox == 3:
              print(f"Dispensing 3 units with load weight
{readWeight2}")
              if readWeight2 == 12:
                dispense_Runner(2880, 23)
              #elif readWeight2 == 9:
                #dispense_Runner(2880, 23)
              #elif readWeight2 == 6:
                #dispense_Runner(2880, 23)
              else:
                dispense_Runner(2880, 23)
              self.thankYouCard()
            else:
              print("Invalid amount for 2nd container")
            self.weightSubtracter2()

            elif self.selectedProduct == 3:  # Dispense 3rd
Container
            print(f"Selected product: {self.selectedProduct}")


amountSpinBox = self.productSpinBox3.value()
            totalAmountSpinBox = int(amountSpinBox)
            print(f"Total amount selected: {totalAmountSpinBox}")
            readWeight3 = int(loadProductWeight3)
            print(readWeight3)
            print("Dispensing from the 3rd container")
            if totalAmountSpinBox == 1:
              if readWeight3 == 12:
                dispense_Runner(860, 24)
              #elif readWeight3 == 5:
                #dispense_Runner(850, 24)
              #elif readWeight2 == 6:
                #dispense_Runner(870, 23)
              else:
                dispense_Runner(860, 24)

              self.thankYouCard()
            elif totalAmountSpinBox == 2:
              dispense_Runner(1780, 24)
              self.thankYouCard()
            elif totalAmountSpinBox == 3:
              print(f"Dispensing 3 units with load weight
{readWeight3}")
              if readWeight3 == 12:
                dispense_Runner(2890, 24)
              else:
                dispense_Runner(2890, 24)
              self.thankYouCard()
            else:
              print("Invalid amount for 3rd container")
            self.weightSubtracter3()
            else:
      print("Invalid product selected")
def productRiceLabel(self): #Calls UpdateProductFunctions in every
spinbox
            productLabel= self.sender()
            productLabel =
self.updateProductName1LineEdit.currentText()
```

```
                productWeight = self.sender()
                productWeight = self.spinBox_2.value()


                #Apply try and except for error handling
        print("productLabel = ", productLabel)
        print("productWeight = ", productWeight)
                if self.spinBox.value() == 1:
                if productLabel == None:
            self.getWeight1FromAdmin()
                else:
            self.getWeight1FromAdmin()
            self.getItemInProduct1DetailCSV()

                elif self.spinBox.value() == 2:
                if productLabel == None:
            self.getWeight2FromAdmin()
                else:
            self.getWeight2FromAdmin()
            self.getItemInProduct2DetailCSV()

                elif self.spinBox.value() == 3: # TO FOLLOW
                if productLabel == None:
            self.getWeight3FromAdmin()
                else:
            self.getWeight3FromAdmin()
            self.getItemInProduct3DetailCSV()
def gsmNotificationForRiceAlert1(self):
        print("ALERT!")
                # Define the serial port for the GSM module
                gsm_port = "/dev/ttyS0"
                # Initialize the serial connection with GSM module
                ser = serial.Serial(gsm_port, 9600, timeout=1)
                number = "+639294697705"
                message = "RICE CONDITION IN CONTAINER 1 IS
BAD"
        ser.write("AT+CMGF=1\r".encode())  # Set the GSM Module in
Text Mode
                time.sleep(1)
        ser.write(('AT+CMGS="' + number + '"\r').encode())  # Set the
recipient number
                print(number)
                time.sleep(1)
                ser.write(( message + chr(26)).encode())  # Write the
message
        print(message)
                time.sleep(1)

                def gsmNotificationForRiceAlert2(self):
        print("ALERT!")
                # Define the serial port for the GSM module
                gsm_port = "/dev/ttyS0"
                # Initialize the serial connection with GSM module
                ser = serial.Serial(gsm_port, 9600, timeout=1)
                number = "+639294697705"
                message = "RICE CONDITION IN CONTAINER 2 IS
BAD"
        ser.write("AT+CMGF=1\r".encode())  # Set the GSM Module in
Text Mode
                time.sleep(1)
        ser.write(('AT+CMGS="' + number + '"\r').encode())  # Set the
recipient number
                print(number)
                time.sleep(1)
                ser.write(( message + chr(26)).encode())  # Write the
message
        print(message)
                time.sleep(1)

                def gsmNotificationForRiceAlert3(self):
        print("ALERT!")
                # Define the serial port for the GSM module
                gsm_port = "/dev/ttyS0"
                # Initialize the serial connection with GSM module
                ser = serial.Serial(gsm_port, 9600, timeout=1)
                number = "+639294697705"
                message = "RICE CONDITION IN CONTAINER 3 IS
BAD"
        ser.write("AT+CMGF=1\r".encode())  # Set the GSM Module in
Text Mode
                time.sleep(1)
        ser.write(('AT+CMGS="' + number + '"\r').encode())  # Set the
recipient number
                print(number)
                time.sleep(1)
                ser.write(( message + chr(26)).encode())  # Write the
message
        print(message)
                time.sleep(1)
def setup_dht_sensor(self, pin):
                try:
            GPIO.setmode(GPIO.BCM)
            dht_device = adafruit_dht.DHT22(pin, use_pulseio=False)
                return dht_device
                except Exception as e:
            print(f"Failed to initialize DHT22 sensor on pin {pin}: {e}")
                return None

                def riceConditionNotification(self):
                dht_device = self.setup_dht_sensor(board.D2)  #
Replace with the BCM pin number

                if dht_device is None:
                return

                try:
            temperature_c = dht_device.temperature
                humidity = dht_device.humidity
            riceStatusText = "Temp: {:.1f} C    Humidity:
{}%".format(temperature_c, humidity)
            self.temperatureSensor.setText(riceStatusText)
                if temperature_c > 40 and humidity > 80:
              print("Bad Condition")
              self.gsmNotificationForRiceAlert1()
                except RuntimeError as err:
            print(err.args[0])

                def riceConditionNotification2(self):
                dht_device = self.setup_dht_sensor(board.D3)  #
Replace with the BCM pin number

                if dht_device is None:
                return

                try:
            temperature_c = dht_device.temperature
                humidity = dht_device.humidity
            riceStatusText = "Temp: {:.1f} C    Humidity:
{}%".format(temperature_c, humidity)
            self.temperatureSensor2.setText(riceStatusText)
                if temperature_c > 40 and humidity > 80:
              print("Bad Condition")
              self.gsmNotificationForRiceAlert2()
                except RuntimeError as err:
            print(err.args[0])

                def riceConditionNotification3(self):
                dht_device = self.setup_dht_sensor(board.D4)  #
Replace with the BCM pin number

                if dht_device is None:
                return

                try:
            temperature_c = dht_device.temperature
                humidity = dht_device.humidity
            riceStatusText = "Temp: {:.1f} C    Humidity:
{}%".format(temperature_c, humidity)
            self.temperatureSensor3.setText(riceStatusText)
                if temperature_c > 40 and humidity > 80:
              print("Bad Condition")
              self.gsmNotificationForRiceAlert3()
                except RuntimeError as err:
            print(err.args[0])
```

LOAD SENSOR READER

```
import RPi.GPIO as GPIO
from hx711 import HX711
from time import sleep

def get_load(known_weight_grams):
   GPIO.setmode(GPIO.BCM)
                hx = HX711(dout_pin=21, pd_sck_pin=20)
```

```
        err = hx.zero()

        reading = hx.get_raw_data_mean()

        reading = hx.get_data_mean()
        value =int(known_weight_grams)

        ratio = reading / value
  hx.set_scale_ratio(ratio)

        load = int(hx.get_weight_mean(20))
        sleep(1)

        return load
```

FLUSH BUTTON

```
import time
import datetime
import RPi.GPIO as GPIO
from gpiozero.pins.pigpio import PiGPIOFactory
from gpiozero import AngularServo
import gpiozero
from time import sleep
import sys

def container(servoPin):
        factory = PiGPIOFactory()
        servo = AngularServo(servoPin, min_angle=0,
max_angle=180, min_pulse_width=0.0005,
max_pulse_width=0.0025, pin_factory=factory)
        servo_relay = gpiozero.OutputDevice(servoPin,
active_high=True, initial_value=False)
        servo.angle = 0

        flushBtn = 27  # GPIO pin number on Raspberry Pi
  GPIO.setmode(GPIO.BCM)
  GPIO.setup(flushBtn, GPIO.IN)

        try:
        timer = 100
        elapsedTime = 0
    servo_relay.on()
        while elapsedTime <= timer:
        btnState = GPIO.input(flushBtn)
        if btnState:
      print("on")
      servo.angle = 0
      sleep(0.1)
        else:
      elapsedTime += 1
      print(f"{elapsedTime:.2f}")
        if elapsedTime == timer:
        print("TAPOS NA")
        servo_relay.off()
        break
        else:
        print("off")
        sleep(0.1)
        servo.angle = 180

        except Exception as e:
    print(f"An error occurred: {e}")

        finally:
    GPIO.cleanup()
    print("GPIO cleanup done")
```

GSM NOTIFIER

```
import serial
import time

def gsmNotificationForThreshold1():
  print("THRESHOLD MESSAGE 1")
        try:
        # Define the serial port for the GSM module
        gsm_port = "/dev/ttyS0"
        # Initialize the serial connection with GSM module
        ser = serial.Serial(gsm_port, 9600, timeout=1)
        numbers = ["+639062957043", "+639294697705"]


message = "CONTAINER 1 CONTENT IS LOW. REFILL
IMMEDIATELY"

        for number in numbers:
        # Set the recipient number
    ser.write(('AT+CMGS="' + number + '"\r').encode())
    time.sleep(1)

        # Write the message
    ser.write((message + chr(26)).encode())
    print(f"Message sent to {number}")
    time.sleep(5)

        ser.close()
        except Exception as e:
  print(f"An error occurred: {e}")

def gsmNotificationForThreshold2():
  print("THRESHOLD MESSAGE 2")
        try:
        # Define the serial port for the GSM module
        gsm_port = "/dev/ttyS0"
        # Initialize the serial connection with GSM module
        ser = serial.Serial(gsm_port, 9600, timeout=1)
        numbers = ["+639062957043", "+639294697705"]
        message = "CONTAINER 2 CONTENT IS LOW.
REFILL IMMEDIATELY"

        for number in numbers:
        # Set the recipient number
    ser.write(('AT+CMGS="' + number + '"\r').encode())
    time.sleep(1)

        # Write the message
    ser.write((message + chr(26)).encode())
    print(f"Message sent to {number}")
    time.sleep(5)

        ser.close()
        except Exception as e:
  print(f"An error occurred: {e}")

def gsmNotificationForThreshold3():
  print("THRESHOLD MESSAGE 3")
        try:
        # Define the serial port for the GSM module
        gsm_port = "/dev/ttyS0"
        # Initialize the serial connection with GSM module
        ser = serial.Serial(gsm_port, 9600, timeout=1)
        numbers = ["+639062957043", "+639294697705"]
        message = "CONTAINER 3 CONTENT IS LOW.
REFILL IMMEDIATELY"

        for number in numbers:
        # Set the recipient number
    ser.write(('AT+CMGS="' + number + '"\r').encode())
    time.sleep(1)

        # Write the message
    ser.write((message + chr(26)).encode())
    print(f"Message sent to {number}")
    time.sleep(5)

        ser.close()
        except Exception as e:
  print(f"An error occurred: {e}")

def gsmNotificationForVibration():
  print("Someone is trying to break the machine")
        try:
        # Define the serial port for the GSM module
        gsm_port = "/dev/ttyS0"
        # Initialize the serial connection with GSM module
        ser = serial.Serial(gsm_port, 9600, timeout=1)

        # Phone numbers to send the message to
        numbers = ["+639062957043", "+639294697705"]
        message = "Someone is trying to break the machine"

        # Set the GSM Module in Text Mode
  ser.write("AT+CMGF=1\r".encode())
        time.sleep(1)

        for number in numbers:
        # Set the recipient number
```

```
        ser.write(('AT+CMGS="' + number + '"\r').encode())
        time.sleep(1)

             # Write the message
        ser.write((message + chr(26)).encode())
        print(f"Message sent to {number}")
        time.sleep(5)  # Ensure enough time for the message to be
sent

             # Close the serial connection
             ser.close()
             except Exception as e:
     print(f"An error occurred: {e}")

#gsmNotificationForThreshold2()
LOAD-SERVO MULTIPROCESSING
import multiprocessing
import serial
import time
from gpiozero.pins.pigpio import PiGPIOFactory
from gpiozero import AngularServo
import gpiozero
import RPi.GPIO as GPIO
from time import sleep
ser = serial.Serial('/dev/ttyUSB0', 9600, timeout=1)

def dispense_Runner(weightThreshold, pin):
             try:
             # Initialize the PiGPIOFactory to handle GPIO
communication via the pigpio daemon
             factory = PiGPIOFactory()
             except IOError as e:
             # Handle the case where the connection to the pigpio
daemon fails
     print(f"Failed to connect to PiGPIOFactory: {e}")
             return

             # Start the monitoring process with the shared factory
instance
             process1 =
multiprocessing.Process(target=monitor_load,
args=(weightThreshold, pin, factory))
             process1.start()
             process1.join()

def monitor_load(weightThreshold, pin, factory):
             #threshold = weightThreshold
   send_reset_command()
             while True:
     print("checking weight")
             weight = get_weight()
             if weight < weightThreshold:
        command_servo(True, pin, factory)
             else:
        print(f"{weight} g")
        command_servo(False, pin, factory)
             break

def command_servo(servoBool, pin, factory):
             servoPin = pin
             # Create an AngularServo object with the shared factory
instance
             servo = AngularServo(servoPin, min_angle=0,
max_angle=180, min_pulse_width=0.0005,
max_pulse_width=0.0025, pin_factory=factory)
             # Create a relay object to control the servo power
             servo_relay = gpiozero.OutputDevice(servoPin,
active_high=True, initial_value=False)
             servo_relay.on()

             if servoBool:
     print("open servo")
             servo.angle = 0
     time.sleep(0.2)
             else:
             servo.angle = 180
             time.sleep(1)
             servo.angle = 0
     servo_relay.off()
VIBRATION SENSING
import RPi.GPIO as GPIO
import time
import multiprocessing
GPIO.setmode(GPIO.BCM)
vibration_pin = 16  # Adjust this to the actual GPIO pin you're using
GPIO.setup(vibration_pin, GPIO.IN)

# Global variable to store vibration status
vibration_detected = multiprocessing.Value('i', 0)

def read_vibration(vibration_detected):
             while True:
     vibration_value = GPIO.input(vibration_pin)
     vibration_detected.value = vibration_value
     time.sleep(0.1)  # Adjust the delay as needed
BILL-COIN CODE
def setup():
             GPIO.setmode(GPIO.BCM)
             GPIO.setup(bill_pin, GPIO.IN)
             GPIO.setup(coin_pin, GPIO.IN)
             print("Insert Bill and Coin. Type 'yes' when done")

def loop():
             global pulse_count_bill, pulse_count_coin,
start_time_bill, start_time_coin

  # BILL PHASE
             if GPIO.input(bill_pin) == GPIO.LOW:
             if time.time() - start_time_bill > debounce_delay:
             pulse_count_bill += 10
             start_time_bill = time.time()

       # Print total amount in real-time
             print("Total Amount Inserted:", pulse_count_bill +
pulse_count_coin)

             # COIN PHASE
             if GPIO.input(coin_pin) == GPIO.LOW:
             if time.time() - start_time_coin > debounce_delay:
             pulse_count_coin += 1
             start_time_coin = time.time()

       # Print total amount in real-time
             print("Total Amount Inserted:", pulse_count_bill +
pulse_count_coin)

             # CHECK FOR "YES" COMMAND
             if input_available():
             user_input = input().strip()
             if user_input == "yes":
             print("Total Amount Inserted (Bill + Coin):",
pulse_count_bill + pulse_count_coin)
             print("Tapos na ey")
             pulse_count_bill = 0
             pulse_count_coin = 0
             # You may choose to remove or reduce this delay
             time.sleep(1)

IR Code
import RPi.GPIO as GPIO
import time

# GPIO setup
def IR_setup():
             GPIO.setwarnings(False)
             GPIO.setmode(GPIO.BCM)
             GPIO.setup(sensor_pin, GPIO.IN)
             try:
             while sensor_pin != False:
             if GPIO.input(sensor_pin):
             print("no object")
             time.sleep(0.5)
             else:
             # If an object is detected
             print("object detected")
             time.sleep(0.1)
             break
             #straight to dispense. wait 2 secs before dispensing

  except Exception as e:
             print(e)
             GPIO.cleanup()

# Update Sales Notification
```

```
import schedule
from datetime import datetime, timedelta

if self.selectedProduct == 1:
        sales_file = 'CSVFiles/sales1.csv'
        notification_function = gsmNotificationForSales1
    elif self.selectedProduct == 2:
        sales_file = 'CSVFiles/sales2.csv'
        notification_function = gsmNotificationForSales2
    else:
        sales_file = 'CSVFiles/sales3.csv'
        notification_function = gsmNotificationForSales3

    # Initialize totalPrice from the previous sales file if it exists
    try:
        with open(sales_file, 'r') as file:
            reader = csv.reader(file)
            next(reader)  # Skip the header
            for row in reader:
                if len(row) >= 2:
                    self.totalPrice = int(row[1])  # Read the previous total
weight
    except FileNotFoundError:
        # If sales file does not exist, initialize totalPrice to 0
        self.totalPrice = 0

    # Read the price from the QLineEdit
    addedPrice = self.textEdit.text()  # Assuming self.textEdit is an
instance of QLineEdit
    try:
        addedPrice = int(addedPrice)
    except ValueError:
        raise ValueError(f"Cannot convert {addedPrice} to int")

    self.totalPrice += addedPrice  # Update totalPrice

    # Write the updated total with timestamp to the sales file
    with open(sales_file, 'a', newline='') as file:
        writer = csv.writer(file)
        writer.writerow([datetime.now().isoformat(), self.totalPrice])

    # Call the notification function
    notification_function(self.totalPrice)

  def aggregate_sales(self, period='day'):
    current_time = datetime.now()
    sales_file = f'CSVFiles/sales{self.selectedProduct}.csv'
    total_sales = 0

    try:
        with open(sales_file, 'r') as file:
            reader = csv.reader(file)
            next(reader)  # Skip the header
            for row in reader:
                if len(row) >= 2:
                    timestamp, price = row
                    timestamp = datetime.fromisoformat(timestamp)
                    price = int(price)

                    if period == 'day' and (current_time - timestamp).days
< 1:
                        total_sales += price
                    elif period == 'week' and (current_time -
timestamp).days < 7:
                        total_sales += price
                    elif period == 'month' and (current_time -
timestamp).days < 30:
                        total_sales += price

    except FileNotFoundError:
        print("No sales data found.")

    return total_sales

  def save_aggregated_sales(self):
    for product in range(1, 4):
        self.selectedProduct = product
        daily_sales = self.aggregate_sales('day')
        weekly_sales = self.aggregate_sales('week')
        monthly_sales = self.aggregate_sales('month')

            with open(f'CSVFiles/aggregated_sales_{product}.csv', 'w',
newline='') as file:
                writer = csv.writer(file)
                writer.writerow(["Period", "Total Sales"])
                writer.writerow(["Daily", daily_sales])
                writer.writerow(["Weekly", weekly_sales])
                writer.writerow(["Monthly", monthly_sales])

            # Send notifications based on the selected product
            if product == 1:
                gsmNotificationForSales1({"Daily": daily_sales, "Weekly":
weekly_sales, "Monthly": monthly_sales})
            elif product == 2:
                gsmNotificationForSales2({"Daily": daily_sales, "Weekly":
weekly_sales, "Monthly": monthly_sales})
            else:
                gsmNotificationForSales3({"Daily": daily_sales, "Weekly":
weekly_sales, "Monthly": monthly_sales})

  def schedule_tasks(self):
    # Schedule end of day task

schedule.every().day.at("21:00").do(self.save_aggregated_sales)

    # Schedule end of week task (Sunday at 21:00)

schedule.every().sunday.at("21:00").do(self.save_aggregated_sales)

    # Schedule end of month task (last day of month at 21:00)
    def is_last_day_of_month():
        tomorrow = datetime.now() + timedelta(days=1)
        return tomorrow.day == 1

    def end_of_month_task():
        if is_last_day_of_month():
            self.save_aggregated_sales()

    schedule.every().day.at("21:00").do(end_of_month_task)

    # Start the schedule in a separate thread or in the main loop
    threading.Thread(target=self.run_scheduler).start()

  def run_scheduler(self):
    while True:
        schedule.run_pending()
        time.sleep(1)  # Check every second

#  Send Update Sales Notification SMS

def gsmNotificationForSales1(totalSold):
  print("Container 1 Confirmaton of Transaction")
  try:
    # Define the serial port for the GSM module
    gsm_port = "/dev/ttyS0"
    # Initialize the serial connection with GSM module
    ser = serial.Serial(gsm_port, 9600, timeout=1)

    # Phone numbers to send the message to
    numbers = ["+639062957043", "+639234522684"]
    message = "Container 1 Confirmaton of Transaction", totalSold

    # Set the GSM Module in Text Mode
    ser.write("AT+CMGF=1\r".encode())
    time.sleep(1)

    for number in numbers:
        # Set the recipient number
        ser.write(('AT+CMGS="' + number + '"\r').encode())
        time.sleep(1)

        # Write the message
        ser.write((message + chr(26)).encode())
        print(f"Message sent to {number}")
        time.sleep(5)  # Ensure enough time for the message to be
sent

    # Close the serial connection
    ser.close()
  except Exception as e:
    print(f"An error occurred: {e}")

def gsmNotificationForSales2(totalSold):
```

```
    print("Container 2 Confirmaton of Transaction")
    try:
        # Define the serial port for the GSM module
        gsm_port = "/dev/ttyS0"
        # Initialize the serial connection with GSM module
        ser = serial.Serial(gsm_port, 9600, timeout=1)

        # Phone numbers to send the message to
        numbers = ["+639062957043", "+639234522684"]
        message = "Container 2 Confirmaton of Transaction", totalSold

        # Set the GSM Module in Text Mode
        ser.write("AT+CMGF=1\r".encode())
        time.sleep(1)

        for number in numbers:
            # Set the recipient number
            ser.write(('AT+CMGS="' + number + '"\r').encode())
            time.sleep(1)

            # Write the message
            ser.write((message + chr(26)).encode())
            print(f"Message sent to {number}")
            time.sleep(5)  # Ensure enough time for the message to be
sent

        # Close the serial connection
        ser.close()
    except Exception as e:
        print(f"An error occurred: {e}")
```

```
def gsmNotificationForSales3(totalSold):
    print("Container 3 Confirmaton of Transaction")
    try:
        # Define the serial port for the GSM module
        gsm_port = "/dev/ttyS0"
        # Initialize the serial connection with GSM module
        ser = serial.Serial(gsm_port, 9600, timeout=1)

        # Phone numbers to send the message to
        numbers = ["+639062957043", "+639234522684"]
        message = "Container 3 Confirmaton of Transaction", totalSold

        # Set the GSM Module in Text Mode
        ser.write("AT+CMGF=1\r".encode())
        time.sleep(1)

        for number in numbers:
            # Set the recipient number
            ser.write(('AT+CMGS="' + number + '"\r').encode())
            time.sleep(1)

            # Write the message
            ser.write((message + chr(26)).encode())
            print(f"Message sent to {number}")
            time.sleep(5)  # Ensure enough time for the message to be
sent

        # Close the serial connection
        ser.close()
    except Exception as e:
        print(f"An error occurred: {e}")
```

Appendix H. Curriculum Vitae

Almazan, Gabrielle Angelo

Blk 11 Lot 34 Ph 5 Graciela St., Villa De Primarosa

Buhay na Tubig, Imus City, Cavite 4103

angelo.almazan0409@gmail.com

Personal Information

Age:	22
Date of Birth:	April 9, 2002
Place of Birth:	Sta. Cruz, Manila
Height:	5'7
Weight:	85 kg
Religion:	Born-Again Christian
Father's Name:	
Mother's Name:	Rochelle Almazan

Education

Elementary	St. Martin Montesorri School (2009-2014)
High School	Blanco Family Academy (2014-2016) University of Perpetual Help – Molino Campus (2016-2018)
Senior High School	Unida Christian Colleges (2018-2020)
College	Polytechnic University of the Philippines - Sta. Mesa (2020-Present)

Work Experience

CMS Team Manager Intern at Truly
(OJT - 2022)

Desktop Engineer Intern at IntouchCX
(OJT - 2023)

Batralo, Gabrielle E.

1755 E Manuel Hizon St., Barangay 337,

Sta. Cruz, Metro Manila, 1003

09062957043

batralogab28@gmail.com

Personal Information

Age:	22
Date of Birth:	February 20, 2002
Place of Birth:	Pasay City, Manila
Height:	5'0
Weight:	52 kg
Religion:	Roman Catholic
Father's Name:	Cesar U. Batralo
Mother's Name:	Preciosa T. Escudero

Education

Elementary	Canossa College San Pablo City (2008-2014)
High School	Canossa College San Pablo City (2014-2018)
Senior High School	Polytechnic University of the Philippines - Sta. Mesa (2018-2020)
College	Polytechnic University of the Philippines - Sta. Mesa (2020-Present)

Work Experience

Developer at Hue Apparel Trading
(On-The-Job Trainee, 2022)

AI Talkbot Support Intern at Wiz AI
(On-The-Job Trainee, 2023)

Bumatay, Xypher Kelly D.

1601, 1016 Duhat, Santa Mesa, Manila

09666448570

kellybumatay5@gmail.com

Personal Information

Age:	21
Date of Birth:	September 12, 2002
Place of Birth:	Bangued, Abra
Height:	5’8
Weight:	68 kg
Religion:	Agnostic
Father’s Name:	Jayson B. Bumatay
Mother’s Name:	Valin D. Bumatay

Education

Elementary	Villaviciosa Central School (2008-2014)
High School	Aurora National High School (2014-2018)
Senior High School	Aurora National Science High School (2018-2020)
College	Polytechnic University of the Philippines - Sta. Mesa (2020-Present)

Work Experience

Information Technology Researcher | Verafede Inc.
(On-The-Job Trainee, 2022)

Designer/Analyst | DOST PES-ITD
(On-The-Job Trainee, 2023)

Quisquino, Dann Joseph R.

Blk 1 Lot 5 Loresville dr., Lores Farm Subd. Brgy.

San Roque, Rizal

09234522684

d4nnjoseph@gmail.com

Personal Information

Age:	22
Date of Birth:	February 11, 2002
Place of Birth:	Antipolo City, Rizal
Height:	5’9
Weight:	73 kg
Religion:	Born Again Christian
Father’s Name:	Dante D. Quisquino
Mother’s Name:	Leonila R. Quisquino

Education

Elementary	Juan Sumulong Elementary School (2008-2014)
High School	San Jose National High School (2014-2018)
Senior High School	Our Lady of Fatima University - Antipolo Campus (2018-2020)
College	Polytechnic University of the Philippines - Sta. Mesa (2020-Present)

Work Experience

Web Developer Intern at Visible Teams Opc (On-The-Job Trainee, 2022)

Desktop Engineer Intern at IntouchCX (On-The-Job Trainee, 2023)

Dela Cruz, John R.
Grace Park, Caloocan City
jrdelacruz@pup.edu.ph

EDUCATIONAL BACKGROUND

Graduate Studies

2024 – present	Batangas State University – The National Engineering University Alangilan, Batangas City Doctor of Philosophy in Electronics Engineering (PhD ECE)
2021 – 2024	University of the Philippines Diliman Diliman, Quezon City Doctor of Philosophy in Electrical and Electronics Engineering (PhD EEE) *Completed 15 academic units*
2016 – 2020	De La Salle University Taft Avenue, Manila Master of Science in Electronics and Communications Engineering (MSECE) *ERDT Scholar*

Undergraduate Studies

2008 – 2013	Polytechnic University of the Philippines Sta. Mesa, Manila Bachelor of Science in Electronics and Communications Engineering (BSECE) *RA 7687 Scholar*

PROFESSIONAL EXPERIENCE

2024 – Present	Technological Institute of the Philippines – Quezon City Aurora Boulevard, Quezon City Part-time Lecturer
2019 – Present	Department of Transportation Ortigas Ave., Mandaluyong City

	Engineer IV, NSCR-EX Project (2022 – present) Engineer III, NSCR-EX Project (2019 – 2021)
2017 – Present	Polytechnic University of the Philippines Sta. Mesa, Manila Associate Professorial Lecturer II (2024 – present) Associate Professorial Lecturer I (2023 – 2024) Assistant Professorial Lecturer IV (2020 – 2023) Instructor III, Part-time (2019 – 2020) Instructor I, Part-time (2017 – 2019)
2018	Ateneo de Manila University Loyola Heights, Quezon City Lecturer
2018	FEU Institute of the Philippines Morayta, Manila Lecturer IV

AWARDS AND RECOGNITIONS

2023 Outstanding Electronics Engineer, conferred by Institute of Electronics Engineers of the Philippines, 24 November 2023

2023 Outstanding IECEP Rizalian Award, conferred by Institute of Electronics Engineers of the Philippines – Rizal Chapter, 23 September 2023

Associate ASEAN Engineer, conferred by ASEAN Federation of Engineering Organizations, 22 March 2019

Appendix H. Technology Transfer

TECHNOLOGY TRANSFER AGREEMENT

KNOW ALL MEN BY THESE PRESENTS:

This **TECHNOLOGY TRANSFER AGREEMENT** made and executed in the City of Quezon, Philippines, by Xypher Kelly D. Bumatay, Gabrielle Angelo Almazan, Gabrielle E. Batralo, Dann Joseph R. Quisquino of College of Engineering, Computer Engineering Department, PUP Sta. Mesa, Manila herein called the **TECHNOLOGY PROVIDER/S;**

\- favor of -

MRS. JOCELYN EDORIA, A sole proprietor, hereinafter called the **TECHNOLOGY RECIPIENT**.

WITHNESSETH:

That the **TECHNOLOGY PROVIDER/S** is/are the absolute owner of that certain prototype "Multi-Grain Rice Dispensing Machine with Small Messaging System Notification" more particularly described as follows:

This project is designed as a dispenser for multiple (3) varieties of rice, along with an interactive payment system that includes a coin and bill slot, sensors that detects the temperature, humidity, vibration, and weight of each rice container, and a module for sending message updates to the seller.

That, for and in consideration of PUP College of Engineering Research mandate, the said **TECHNOLOGY PROVIDER/S** hereby **cedes, transfer, and conveys** unto the said **TECHNOLOGY RECIPIENT/S** the technology described above, free and clear of all liens and encumbrances.

That the **TECHNOLOGY RECIPIENT/S** shall be responsible for the monitoring, maintenance and control of the said technology.

ACCEPTANCE

That the **TECHNOLOGY RECIPIENT/S** does hereby accepts the foregoing technology from the TECHNOLOGY **PROVIDER/S**.

IN WITNESS HEREOF, the parties hereto have hereunto set their hands, on this ____ day of _______, 20___, in the City/Municipality of ______________________.

TECHNOLOGY PROVIDERS **TECHNOLOGY RECIPIENT/S**

By:

Xypher Kelly D. Bumatay
Proponent

Gabrielle Angelo Almazan
Proponent

Gabrielle E. Batralo
Proponent

Dann Joseph R. Quisquino
Proponent

By:

Mrs. Jocelyn Edoria
Representative

Signed in the Presence of

Remedios G. Ado
Dr. Remedios G. Ado
Witness

Engr. John R. Dela Cruz
Witness

ACKNOWLEDGMENT

REPUBLIC OF THE PHILIPPINES)
CITY OF CITY OF MANILA) S. S.

In the City of CITY OF MANILA, this _______ day of AUG 22 2024, 20__, personally appeared before me the following persons:

Known to me and to me known to the same persons who executed the foregoing instrument and acknowledged the same to be her free and voluntary act and deed.

Notary Public

ATTY. JOEL E. PANER
NOTARY PUBLIC COMMISSION NO. 2024-014
Issued on 2-16-2024 Until Dec. 31, 2025 / Manila
UNIT 208 TMR II TAFT AVE., MALATE, MANILA
ROLL NO. 44009 * IBP LIFETIME NO. 2022 / 15-12-00
PTR No.1557911 / 01-15-2024 / City of Manila / TIN 104063310
MCLE Compliance No. VII-0007120 / 04-14-2025

Doc No. 422
Page No. 84
Book No. XII
Series of 2024