SMART SHOPPING TROLLEY WITH BILLING USING ARDUINO UNO

PROJECT REPORT

SOLVIX

Team Name : Solvix

Team Member :

AROKIYA NITHISH. J

ISHWARYA. M

(Vel Tech University)

Domain : Smart Automation

Theme : Smart Retail / Intelligent Shopping

Date : 28/05/2025

ABSTRACT

In todays fast-paced world, consumers demand quick and convenient shopping experiences. Traditional shopping methods, especially in supermarket and retail outlets, often involve long checkout queues and time-consuming billing process. To overcome these issues and improve customer satisfaction, technology-driven solutions have emerged. One such innovation is the Smart system using RFID .This system aims to streamline the shopping process by automating product identification and billing. By integrating an RFID reader with a microcontroller, each product can be represented by an RFID card. When the customer scans the card, the system identifies the item instantly and updates the status on an LCD display. This goal is eliminating the manual billing, reduce human error, and improve efficiency. This innovative model offers an initial step towards smart retail solutions and lays the foundation for more complex inventory and payment system in the future.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF ABBREVIATIONS

S.NO	SYMBOL	ABBREVITIONS
1	RFID	Radio Frequency Identification
2	LCD	Liquid Crystal Display
3	I2C	Inter-Integrated Circuit

CHAPTER I

INTRODUCTION

In this project, titled **“Smart Shopping Trolley with Billing Using Arduino UNO”** we aim to revolutionize the traditional shopping experience by integrating automation directly into the shopping process. The project is designed to minimize the time spent by customers at billing counters in supermarkets by enabling automatic billing within the trolley itself. This innovative approach enhances convenience, reduces manpower requirements, and simplifies the overall shopping workflow. The system is built using an Arduino microcontroller, an RFID reader, multiple RFID cards (tags) representing different products, a 16x2 LCD display, and a battery module. Each product is associated with a unique RFID tag. As the customer places an item into the trolley, they scan its RFID card using the reader. Upon successful scanning, the system displays the product name and price on the LCD display and stores the information internally. The process is repeated for up to four products, each with its own RFID card. After scanning the fourth product, the Arduino automatically calculates the total bill by summing the prices of all scanned products. This total amount is then displayed on the LCD for the customer to review. After a set duration of five minutes, the system resets itself, clearing all previously stored data and preparing the trolley for the next customer. This smart trolley system not only addresses long queues at billing counters but also supports a contactless shopping experience, which is especially important in today’s health-conscious environment. By integrating RFID and microcontroller technology, the project offers a cost-effective and scalable solution for modern retail environments.

.

CHAPTER II

EXISTING SYSTEM

In the current supermarket and retail store environment, the shopping and billing process is mostly manual. Customers use a standard shopping trolley to collect items, and once they are done shopping, they queue up at billing counters. Each item is scanned manually by a cashier using a barcode scanner, and the bill is calculated through a computer-based point-of-sale (POS) system. While this process is functional, it is time-consuming and inefficient, especially during peak shopping hours. The manual billing system often results in long queues at checkout counters, causing frustration and delays for customers. Additionally, the need for staff to scan each item manually increases labor costs for the supermarket. Errors in scanning or entering item quantities can also lead to incorrect billing, customer dissatisfaction, and revenue loss. This traditional system depends heavily on human intervention and lacks real-time automation that could significantly enhance customer service. Some stores have tried to improve this system using barcode-based self-checkout machines. However, these systems still require the customer to manually scan each product at a separate kiosk, which only partially solves the queue problem. Moreover, barcode scanners are line-of-sight devices that require proper positioning of the barcode in front of the scanner, which can be tedious and time-consuming. These systems also tend to be expensive and require maintenance and supervision

CHAPTER III

PROBLEM IDENTIFICATION

What challenges do supermarkets face due to manual billing processes, and how can these challenges be overcome to improve the efficiency and convenience of the shopping experience?

SOLUTION:

The conventional billing system in supermarkets relies heavily on manual scanning of products by cashiers, which leads to several significant challenges. These include long waiting times for customers, especially during peak hours, increased labor costs for store management, and the risk of human errors such as incorrect product scanning or billing mistakes. To overcome these issues, the project proposes the design and implementation of a Smart Shopping Trolley with Billing using Arduino. This system leverages RFID technology to automate the billing process directly at the point of shopping. Each product is assigned a unique RFID tag, and the trolley is equipped with an RFID reader connected to an Arduino microcontroller. When a customer places a product into the trolley, they scan its RFID tag, which immediately displays the product details and price on an LCD screen. The Arduino stores the price and keeps a running total of all scanned items. After scanning all the products, the total amount is automatically calculated and displayed, eliminating the need to wait in checkout lines. This approach not only speeds up the checkout process but also reduces the dependency on manual labor, minimizes billing errors, and provides a seamless, user-friendly shopping experience.

CHAPTER IV

PROPOSED SYSTEM

The proposed system is a Smart Shopping Trolley with Billing Using Arduino that aims to automate and speed up the billing process in supermarkets. Each product is tagged with a unique RFID card, and the trolley is equipped with an RFID reader connected to an Arduino microcontroller. When a product is placed in the trolley, its RFID tag is scanned, and the product name and price are immediately shown on an LCD screen. The system stores each product's price and keeps a running total of the items scanned. Customers can scan up to four RFID cards, with a delay of five minutes between scans to allow time for shopping. Once the fourth product is scanned, the system automatically calculates the total bill and displays the final amount on the screen. After completing the billing process, the trolley resets after a fixed time, clearing the stored data and making it ready for the next user. The entire system is powered by a battery, making the trolley portable and easy to use anywhere inside the store. This smart trolley reduces the need for customers to wait in long queues at billing counters, lowers human errors during checkout, and improves the overall shopping experience by providing real-time billing updates. The use of affordable Arduino and RFID components also makes this system cost-effective and scalable for widespread adoption in retail stores. In the future, the system can be enhanced with additional features like mobile app integration or inventory management.

CHAPTER V

BLOCK DIAGRAM / DESIGN

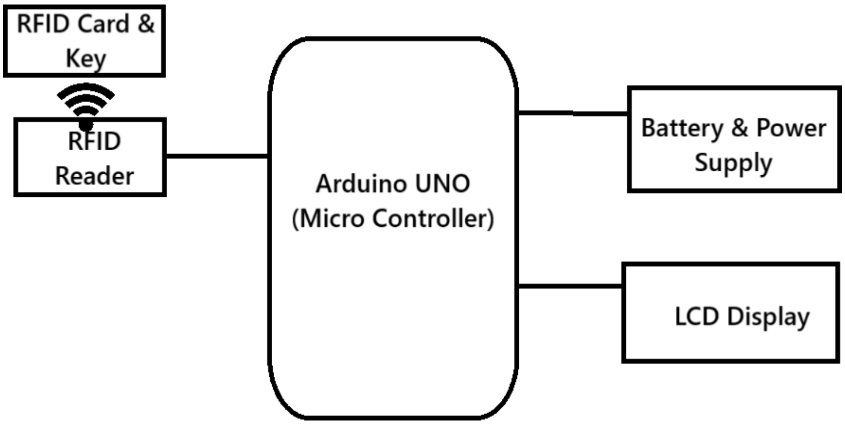

Fig 5.1 Block Diagram of Smart Shopping Trolley with Billing

CHAPTER VI

MATERIALS

1. **Arduino Uno :** The Arduino Uno is a popular microcontroller development board that serves as an excellent starting point for electronics and programming enthusiasts.

Fig 6.1

2. **Jumper Wires :** Jumper wires are essential components in electronics prototyping. Jumper wires, also known as DuPont wires, are electrical wires with connector pins at each end. They allow you to connect two points in a circuit without soldering.

Fig 6.2

3. **Battery :** A 3.7V Lithium ion battery can be a versatile power source, The Arduino Uno documentation states that the board can operate on an external supply of 6v volts as series of battery. However, if supplied with less than 7V, the 5V pin may provide less than five volts, and the board could become unstable.

Fig 6.3

4. **Battery Chargering System :** A 5V Battery Charger takes 5V input (usually from USB) and charges a 3.7V Lithium-ion/Li-Po battery safely. Input Voltage: Accepts 5V (from USB, adapter, etc.). ..

Fig 6.4

5. **LCD Display with I2C Module :** The 16x2 LCD with I2C module allows you to display text (2 lines × 16 characters) using only 2 Arduino pins (SDA and SCL), thanks to the I2C interface. I2C Module (typically using the PCF8574 IC) converts I2C signals into parallel signals needed by the LCD.

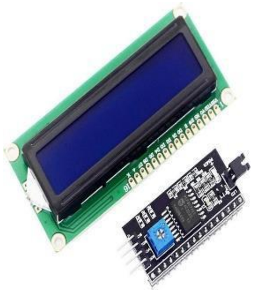

Fig 6.5

6. **MFRC522 RFID Reader Module :** The MFRC522 is a 13.56 MHz RFID reader module used to read and write data from RFID cards or tags via SPI communication with Arduino.Its Emits electromagnetic waves to detect RFID cards/tags nearby. Communication: When a tag comes close, it exchanges data using RF field modulation.

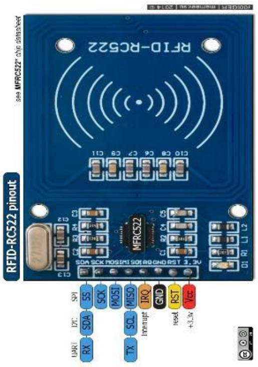

Fig 6.6

7. **RFID Card :** The MIFARE RFID card is a passive tag that stores a Unique ID (UID) and data blocks, which can be read or written by an RFID reader like the MFRC522. When Near Reader: The card uses radio frequency (RF) communication to send its UID to the reader.

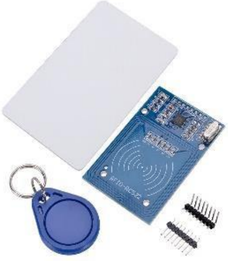

Fig 6.7

8. **PVC Plastic Enclosure Box :** A PVC plastic box is a durable enclosure used to house and protect electronic circuits like Arduino boards, sensors, batteries, and other components from dust, moisture, and physical damage**.**

Fig 6.8

CHAPTER VII

WORKING

The Smart Shopping Trolley with Billing system is designed to simplify the shopping process by allowing customers to manage their billing directly through the trolley. The system uses an Arduino microcontroller connected to an RFID reader, LCD display, and a battery for power. As customers shop, they scan each item using the RFID reader attached to the trolley. Each item in the store has a unique RFID card. When a customer scans the RFID card of a product, the RFID reader reads the unique ID and sends it to the Arduino. The Arduino is pre-programmed with the details of the RFID cards, such as the product name and price. Once a card is scanned, the corresponding product name and price are displayed on the LCD screen attached to the trolley .After displaying the product information, the system adds the price to a running total, which is stored in the Arduino’s memory. The customer is required to scan items one after another with a delay of five minutes between each scan. This delay simulates the real-time shopping process and allows the customer to take their time selecting products. The system allows up to four products to be scanned in a single shopping session. Once the fourth product is scanned, the Arduino calculates the final bill by summing the prices of all the scanned products. This total amount is displayed on the LCD screen, giving the customer a complete and real-time view of their shopping cost. Finally, the system is designed to reset automatically after a fixed duration, clearing all the stored data from the previous session. This feature ensures that the trolley is ready for use by the next customer without manual intervention.

MODEL / PROTOTYPE

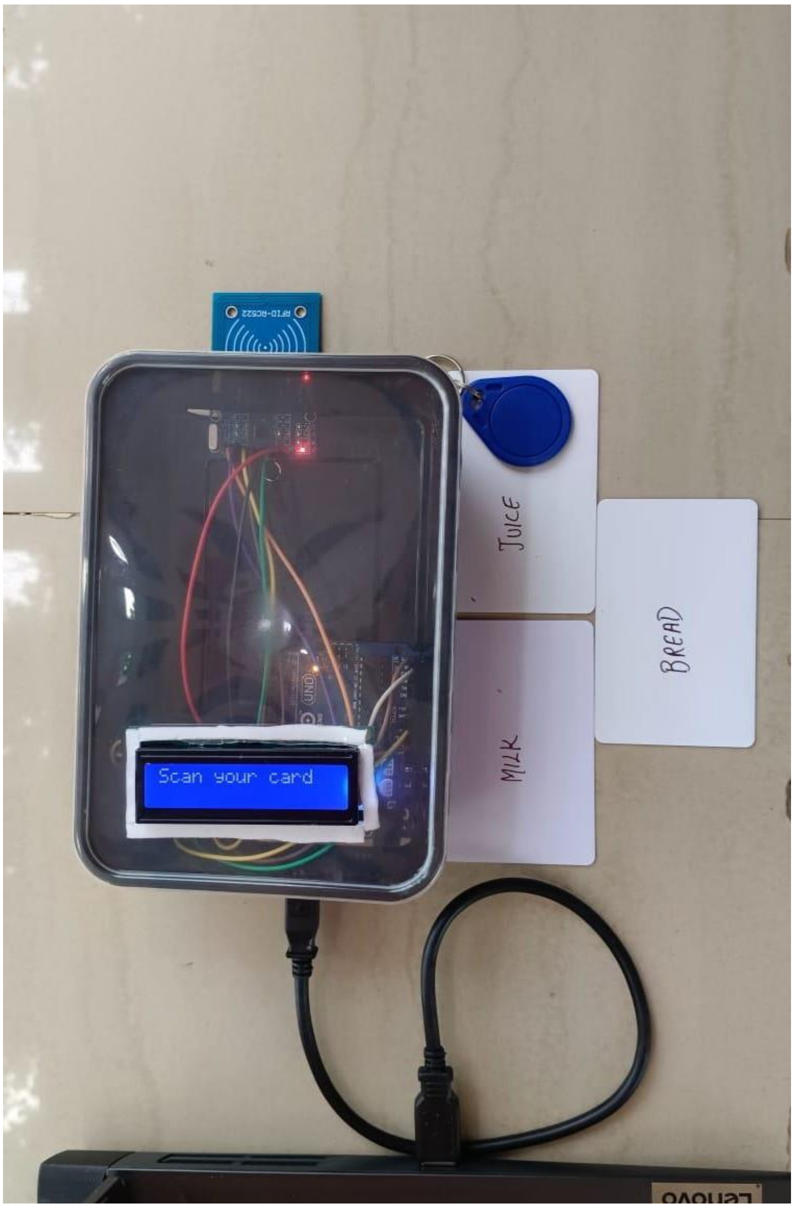

Fig7.1 Model of Smart Shopping Trolley with Billing Using Arduino

CHAPTER VIII

SOFTWARE DETAILS

The software component of the Smart Shopping Trolley system is centered around the Arduino Integrated Development Environment (IDE), which is used for writing, compiling, and uploading the control program to the Arduino microcontroller. The Arduino IDE supports programming in C/C++, providing a flexible and efficient platform for embedded system development. The program running on the Arduino is designed to manage multiple tasks simultaneously, such as reading RFID tags, displaying product details on the LCD, storing product prices, calculating totals, and managing timing for scanning delays and system resets. The core of the software consists of the RFID communication protocol, LCD display control, and logic for price accumulation. When an RFID tag is scanned, the Arduino communicates with the RFID reader using the SPI (Serial Peripheral Interface) protocol to read the unique identifier from the tag. The software then compares this ID against a predefined database stored within the Arduino's memory, which contains product names and prices mapped to their RFID codes. In summary, the software integrates RFID reading, data processing, display management, and timing controls within a compact Arduino program. This approach provides a seamless and automated billing experience, reducing the need for manual operations and improving customer convenience in supermarkets.

CHAPTER IX

SOURCE CODE

Check Your RFID Card UIDs Number :

```
#include <SPI.h>

#include <MFRC522.h>


#define SS_PIN 10

#define RST_PIN 9


MFRC522 rfid(SS_PIN, RST_PIN);


void setup() {

  Serial.begin(9600);

  SPI.begin();

  rfid.PCD_Init();

  Serial.println("Scan your RFID key...");

}


void loop() {

  if  (!rfid.PICC_IsNewCardPresent()  ||  !rfid.PICC_ReadCardSerial())  return;
  Serial.print("UID:  ");
  for (byte i = 0; i < rfid.uid.size; i++) {
    if (rfid.uid.uidByte[i] < 0x10) Serial.print("0");
    Serial.print(rfid.uid.uidByte[i], HEX);
    if (i != rfid.uid.size - 1) Serial.print(" ");
  }
  Serial.println();
  delay(2000);
  rfid.PICC_HaltA();
  rfid.PCD_StopCrypto1();
}
```

Main Code for Smart Shopping Trolley System :

```
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
#include <SPI.h>
#include <MFRC522.h>

#define RST_PIN 9
#define SS_PIN 10
```

```
MFRC522 rfid(SS_PIN, RST_PIN);

LiquidCrystal_I2C lcd(0x27, 16, 2);


// Define RFID card UIDs

String cardUIDs[4] = {

"EF 69 35 1E",  // Milk

  "F3 AB DD E2",  // Bread

  "63 97 60 E4",  // Juice

  "97 8B 7A 00"    // Checkout

};


String productNames[3] = {"Milk", "Bread", "Juice"};

int productPrices[3] = {30, 20, 25};

bool productScanned[3] = {false, false, false};

int totalAmount = 0;

unsigned long lastScanTime = 0;

bool checkoutDone = false;


void setup() {

  Serial.begin(9600);

  SPI.begin();

  rfid.PCD_Init();
  lcd.init();

  lcd.backlight();

  lcd.setCursor(0, 0);

  lcd.print("Scan your card");

}


void loop() {

  if  (!rfid.PICC_IsNewCardPresent()  ||  !rfid.PICC_ReadCardSerial())  return;


  // Read UID

  String scannedUID = "";

  for (byte i = 0; i < rfid.uid.size; i++) {

    if (rfid.uid.uidByte[i] < 0x10) scannedUID += "0";

    scannedUID += String(rfid.uid.uidByte[i], HEX);

    if (i < rfid.uid.size - 1) scannedUID += " ";

  }

  scannedUID.toUpperCase();

  Serial.println("Scanned UID: " + scannedUID);


  // Check if it's the checkout card

  if (scannedUID == cardUIDs[3] && !checkoutDone) {

    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("Checkout Done");
    lcd.setCursor(0, 1);
    lcd.print("Total: Rs." + String(totalAmount));
    checkoutDone = true;
    lastScanTime = millis();
  }
  // Check if it's a product card
  else if (!checkoutDone) {
    for (int i = 0; i < 3; i++) {
      if (scannedUID == cardUIDs[i]) {
        if (!productScanned[i]) {
        productScanned[i] = true;
        totalAmount += productPrices[i];
        lcd.clear();
          lcd.setCursor(0, 0);
          lcd.print(productNames[i]);
          lcd.setCursor(0, 1);
          lcd.print("Rs." + String(productPrices[i]));
        } else {
          lcd.clear();
          lcd.setCursor(0, 0);
```

```
          lcd.print("Already added:");
          lcd.setCursor(0, 1);
          lcd.print(productNames[i]);
        }
        lastScanTime = millis();
        break;
      }
    }
  }

  delay(1000); // 1 second delay after each scan
  rfid.PICC_HaltA();
  rfid.PCD_StopCrypto1();

  // Reset system after 2 seconds of checkout
  if (checkoutDone && millis() - lastScanTime >= 2000) {
    resetSystem();
  }
}

void resetSystem() {
  totalAmount = 0;
  for (int i = 0; i < 3; i++) productScanned[i] = false;
  checkoutDone = false;
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("Restarted...");
  delay(1000);
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("Scan your card");
}
```

CHAPTER X

HARDWARE DISCUSSION

The Smart Shopping Trolley system uses an Arduino Uno as the main controller, which manages all hardware components and processes the data from the RFID reader. The RFID reader module scans RFID cards attached to products, each containing unique identification and price information. This eliminates the need for manual barcode scanning and speeds up the checkout process. An LCD display (16x2) is connected to the Arduino to show the product name and price immediately after each scan, along with the total amount after all items are scanned. The system uses a rechargeable battery for power, making the trolley portable and easy to use anywhere in the store. The program running on the Arduino continuously reads RFID tags via SPI communication, matches the tag IDs with stored product details, and updates the display accordingly. A timing function ensures a 5-minute delay between scans, allowing customers time to add products. After four products are scanned, the total is calculated and shown on the LCD. The system then resets automatically to prepare for the next user. This combination of hardware and software enables fast, accurate billing directly on the trolley, reducing manual effort and errors, and enhancing the overall shopping experience.

CHAPTER XI

ADVANTAGES AND DISADVANTAGES

ADVANTAGES:

The Smart Shopping Trolley system offers numerous advantages that can transform the traditional shopping experience. First and foremost, it significantly reduces checkout time by allowing customers to scan products as they shop, eliminating long queues at billing counters. This real-time scanning also reduces errors often caused by manual barcode scanning, such as missed items or incorrect prices, leading to more accurate billing. Using RFID technology enhances convenience because it does not require a direct line of sight like barcode scanners, making it faster and more reliable in reading product data. The LCD display provides immediate feedback to the shopper, showing product names, prices, and cumulative totals, which helps users keep track of their spending. Furthermore, the portable, battery-operated design ensures the trolley can be used anywhere within the store without dependency on wired power sources. This system also helps reduce staffing requirements for billing, potentially lowering operational costs for supermarkets. Additionally, it can improve customer satisfaction by providing a seamless and modern shopping experience, appealing especially to tech-savvy shoppers**.**

DISADVANTAGES:

Despite these benefits, the Smart Shopping Trolley system has some challenges and limitations. One major drawback is the cost and effort required to tag every product with RFID cards, which can be expensive and time-consuming to implement across all items in a supermarket. RFID technology is also susceptible to interference from metal objects or electromagnetic signals, which might occasionally cause scanning errors or missed reads. The system's current design supports scanning only a limited number of items (four in this project), which restricts its usability for larger shopping trips unless the hardware and software are expanded. The imposed 5-minute interval between scans might frustrate customers who want to add products more quickly, potentially slowing down the shopping process. Additionally, the system depends heavily on proper maintenance; RFID tags can be damaged or lost, requiring replacement. The trolley's hardware components, like the battery and LCD, also need regular checks to ensure consistent performance. Lastly, software bugs or glitches in the Arduino program could cause the system to freeze or malfunction, necessitating technical support and increasing maintenance efforts.

CHAPTER XII

CONCLUSION

In conclusion, the Smart Shopping Trolley with Billing system is a significant step forward in modernizing the retail shopping experience by incorporating advanced technologies like RFID and microcontroller programming. This project successfully demonstrates how automation can simplify and speed up the billing process, reducing the dependency on manual labor and minimizing human errors such as incorrect product scanning and miscalculated bills. By allowing customers to scan products in real-time and view prices and totals immediately on the LCD display, the system enhances transparency and empowers shoppers to keep better track of their expenditures. Moreover, the automatic reset feature after a fixed period ensures that the trolley is always ready for the next customer, promoting efficiency and usability. While there are challenges related to the cost of implementing RFID tags on all products and the limitations in the number of items scanned per session, these issues can be addressed through future enhancements and scalability options. Overall, this project highlights the potential of combining hardware and software technologies to improve retail operations, reduce checkout times, and elevate customer satisfaction. As supermarkets continue to evolve, integrating smart solutions like this trolley can pave the way for more intelligent, seamless, and convenient shopping environments worldwide.

REFERENCES

1. Machhirke, K., Goche, P., Rathod, R., Petkar, R. and Golait, M., 2017. A new technology of smart shopping cart using RFID and ZigBee. International Journal on Recent and Innovation Trends in Computing and Communication, 5(2), pp.256-259.
2. Chandrasekar, P. and Sangeetha, T., 2014, February. Smart shopping cart with automatic billing system through RFID and ZigBee. In International Conference on Information Communication and Embedded Systems (ICICES2014) (pp. 1-4). IEEE.
3. Ali, Z. and Sonkusare, R., 2014, August. RFID based smart shopping: an overview. In 2014 international conference on advances in communication and computing technologies (ICACACT 2014) (pp. 1-3). IEEE.
4. Shahroz, M., Mushtaq, M.F., Ahmad, M., Ullah, S., Mehmood, A. and Choi, G.S., 2020. IoT-based smart shopping cart using radio frequency identification. Ieee Access, 8, pp.68426-68438.
5. https://youtu.be/7zR-OuZuCUM?si=c2kP76Z5N3Yyzy42