IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

Impact Factor 8.414 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 13, Issue 8, August 2025

DOI: 10.17148/IJIREEICE.2025.13822

Smart Shop Cart

Prof Vanishree H V[1], Khushi V[2], Pooja D Rao[3], Uzma Naaz[4], Varun H S[5]

Assistant Professor, Dept. of ECE., Bapuji Institute of Engineering and Technology, Davanagere, Karnataka, India[1]

UG Student, Dept. of ECE., Bapuji Institute of Engineering and Technology, Davanagere, Karnataka, India[2-5]

Abstract: Today, retail shopping demands increased convenience and efficiency to meet customer expectations. This paper presents a Smart Shop Cart system that integrates Arduino-based control, RFID technology, and sensor-driven human-following capabilities to automate and enhance the shopping experience. Utilizing RFID tags and readers, the system enables automatic product identification and real-time billing, minimizing manual effort and checkout time.

The human-following mechanism, based on ultrasonic and infrared sensors, allows the cart to autonomously track the shopper, providing hands-free mobility. Additionally, a voice assistant and a mobile application offer interactive control and real-time purchase monitoring. Experimental results demonstrate that the proposed system significantly improves shopping convenience and operational efficiency, making it a promising solution for smart retail environments.

Keywords: Smart Shop Cart, Arduino, RFID, Human Following System, Voice Assistant, Automated Billing, Embedded Systems, Sensor Integration

I. INTRODUCTION

Today, retail shopping has traditionally involved manual effort in product selection, carrying items, and billing, leading to inefficiencies and inconvenience. With the advancement of embedded systems and automation technologies, there is an increasing demand for smart solutions that enhance customer experience and streamline store operations.

This paper proposes a Smart Shop Cart system that integrates Arduino microcontroller technology with RFID-based product identification, a sensor-driven human-following mechanism, and a voice assistant interface. The system aims to reduce manual handling, speed up billing processes, and provide a hands-free shopping experience.

The key contributions of this work include the development of an autonomous human-following cart using ultrasonic and infrared sensors, real-time product detection via RFID tags and readers, and the integration of a voice-controlled user interface coupled with a mobile application for purchase tracking. The proposed system enhances shopping convenience, reduces checkout times, and offers a scalable framework for smart retail environments.

II. OBJECTIVE

The objectives of the project:

- To design and implement a smart shopping cart system using Arduino and ESP8266.
- To enable the cart to automatically follow the user using ultrasonic sensor and motorized wheels.
- To implement smart billing functionality using RFID reader and RFID tags.
- To enable real-time data synchronization and cart monitoring using ESP8266 Wi-Fi module.
- To integrate user and admin mobile applications for live tracking and billing management.

III. METHODOLOGY

3.1 BlOCK DIAGRAM

The Automatic Human Following System shown in Figure 1, for a Smart Shopping Cart, using an Arduino-based setup. The core controller of the system is the Arduino UNO, which interfaces with multiple components to achieve the desired automation.

DOI: 10.17148/IJIREEICE.2025.13822

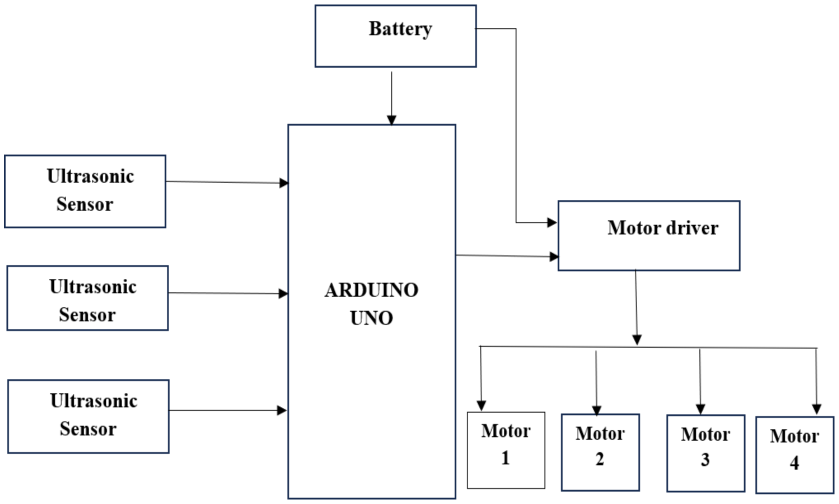

Figure 1: Block Diagram of Human Following Mechanism

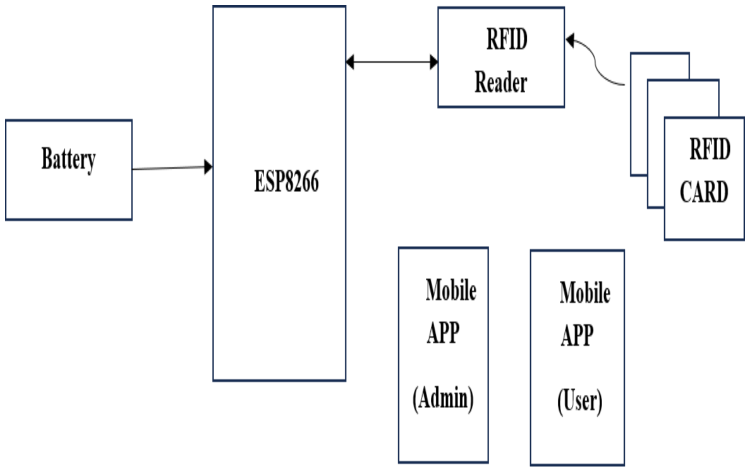

Figure 2: Block Diagram of Smart Billing

The Smart billing system shown in Figure 2, used in the Smart Shopping Cart project. The core of this system is the ESP8266 microcontroller, which is responsible for managing communication and processing the RFID-based billing data.

The system begins with an RFID Reader that scans RFID Cards attached to products. When a customer places an item in the shopping cart, the RFID tag of the item is detected by the reader. The RFID Reader sends this data to the ESP8266, which processes the item information and forwards it for further handling.

3.2 FLOW CHART

The Human Following Mechanism flowchart starts with powering on the system and reading distance from ultrasonic sensors placed on the left, center, and right. If the person is detected in the center, the cart moves forward; otherwise, it turns left. Depending on the movement, the cart either stops, maintains its speed, or adjusts motor control to follow the user.

The Smart Billing System flowchart begins by powering on the billing components. If an RFID tag is detected, the tag ID is read, and the ESP8266 fetches product details. The product is then added or removed from the billing list, and the total amount is updated. This updated billing information is sent to a mobile app via Wi-Fi. Once shopping is complete, the system finalizes the billing list and ends the process.

IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526
International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering
Impact Factor 8.414 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 13, Issue 8, August 2025
DOI: 10.17148/IJIREEICE.2025.13822

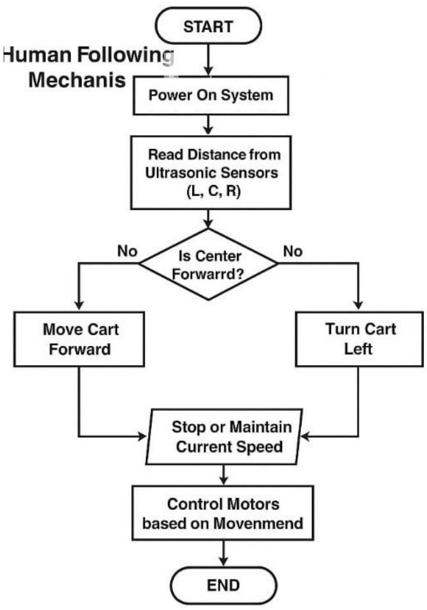

Figure 3: Human Following Mechanism

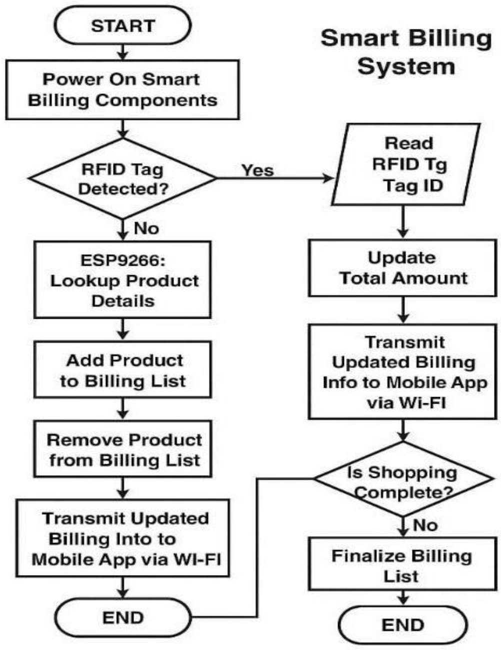

Figure 4: Smart Billing

II. HARDWARE DESCRIPTION

4.1ARDUINO UNO

Arduino Uno is a popular open-source microcontroller board based on the ATmega328P microchip. It is widely used in electronics, automation, and robotics projects by beginners, students, and professionals. The board features 14 digital I/O pins (6 capable of PWM), 6 analog input pins, a USB port, power jack, and a 16 MHz quartz crystal. It operates at 5V and is programmed using the Arduino IDE, which supports a simple, C/C++-like programming language. Arduino Uno can read data from sensors (temperature, light, distance, etc.) and control devices like LEDs, motors, and displays. It connects easily to external modules like Wi-Fi (ESP8266), Bluetooth (HC-05), and more.

Figure 5: Arduino Uno

4.2 ESP8266

The ESP8266 is a powerful, low-cost, and versatile Wi-Fi-enabled microcontroller module developed by Espressif Systems. It has revolutionized the field of IoT (Internet of Things) by making wireless connectivity accessible and affordable for embedded applications. The ESP8266 is capable of not only acting as a Wi-Fi adapter but also functioning as a standalone microcontroller, making it suitable for a wide range of smart and connected applications. The ESP8266 is built around a 32-bit Tensilica L106 RISC processor, running at a default clock speed of 80 MHz, which can be boosted to 160 MHz

Figure 6: ESP8266

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering
Impact Factor 8.414 ⪤ Peer-reviewed & Refereed journal ⪤ Vol. 13, Issue 8, August 2025
DOI: 10.17148/IJIREEICE.2025.13822

4.3 RC522 RFID MODULE

The RC522 RFID module is a low-cost, contactless communication device used for reading and writing data to RFID cards operating at a frequency of 13.56 MHz It works by generating an electromagnetic field through its antenna to power passive RFID tags or cards when they are brought close to the module. RFID (Radio-Frequency Identification) technology plays a crucial role in automating the shopping and billing process. Each product in the store is tagged with a unique RFID tag that contains essential information such as product ID or name. The trolley is equipped with an RFID reader (such as the RC522 module), which continuously scans for nearby RFID tags. When a customer places an item into the trolley, the RFID reader detects the tag and sends the tag's unique ID to the microcontroller ESP8266.

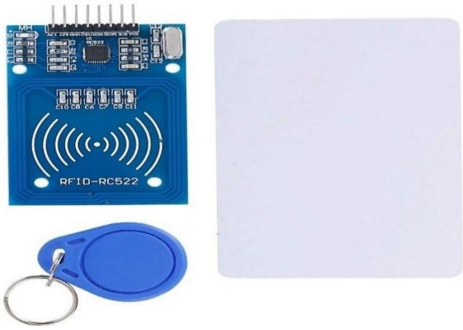

Figure 7: RC522 RFID MODULE

4.4 ULTRASONIC SENSOR

An ultrasonic sensor is an electronic device that measures the distance to an object by using ultrasonic sound waves. It works on the principle of echolocation, which is similar to how bats and dolphins navigate by emitting high-frequency sound pulses and interpreting the echo that returns after bouncing off objects. The sensor emits a high-frequency sound wave (typically around 40 kHz) through a transmitter. This wave travels through the air until it hits an object and bounces back. The reflected wave is detected by a receiver

Figure 8: Ultrasonic Sensor

IV. SOFTWARE DESCRIPTION

Arduino is a prototype platform (open-source) based on an easy-to-use hardware and software. It consists of a circuit board, which can be programed (referred to as a microcontroller) and a ready- made software called Arduino IDE (Integrated Development Environment), which is used to write and upload the computer code to the physical board.

5.1 ARDUINO IDE SOFTWARE

Figure 9: ARDUINO IDE

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering
Impact Factor 8.414 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 13, Issue 8, August 2025
DOI: 10.17148/IJIREEICE.2025.13822

The key features are:

- Arduino boards are able to read analog or digital input signals from different sensors and turn it into an output such as activating a motor, turning LED on/off, connect to the cloud and many other actions.
- You can control your board functions by sending a set of instructions to the microcontroller on the board via Arduino IDE (referred to as uploading software).
- Unlike most previous programmable circuit boards, Arduino does not need an extra piece of hardware (called a programmer) in order to load a new code onto the board. You can simply use a USB cable.
- Additionally, the Arduino IDE uses a simplified version of C++, making it easier to learn to program.
- Finally, Arduino provides a standard form factor that breaks the functions of the microcontroller into a more accessible package.
  - **Step 1:** Install the Arduino Software (IDE)
  - **Step 2:** Get an Uno R3 and USB cable
  - **Step 3:** Connect the board

The Arduino Integrated Development Environment (IDE) is a software platform that is used to write, compile, and upload code to Arduino microcontroller boards. It provides a user-friendly interface for programming Arduino boards and offers a range of tools for code development, debugging, and uploading firmware. Here is an overview of the Arduino IDE software:

Library Manager: The Arduino IDE includes a Library Manager that allows you to easily search, install, and manage third-party libraries, which are pre-written code that can be used to extend the functionality of your Arduino projects.
Board Manager: The Arduino IDE includes a Board Manager that allows you to select and configure the specific Arduino board you are using for your project. It provides a wide range of options for different Arduino boards, including official Arduino boards and third-party boards.
Serial Monitor: The Arduino IDE includes a Serial Monitor tool that allows you to communicate with the Arduino board over a serial connection. It provides a text-based interface for sending and receiving data between the Arduino board and your computer, which can be useful for debugging and monitoring the behavior of your Arduino projects.'

- **Upload:** The Arduino IDE allows you to compile and upload your code to the Arduino board using a USB connection. The upload process involves compiling the code into firmware that can be executed by the Arduino board, and then uploading the firmware to the board's microcontroller memory.

Figure 10: Visualized Picture and Logo

The Smart Shop application is an Android-based e-commerce platform developed using Android Studio (Arctic Fox, Dolphin). It supports user authentication, product browsing, billing management, and checkout with integrated QR code scanning.

Architecture and Framework:

- Languages & SDKs: Java, Kotlin, Android API level 35+, compatible with target SDK versions.
- Frameworks: Firebase (Authentication, Database, Storage), Retrofit (API integration).
- Modules: ViewModel, CartItemModel, SmartShop ViewModel for product management, billing, and authentication.

Data Management:

- Firebase Authentication: Supports email/password-based login.
- Firebase Realtime Database: Stores product details, user cart data, and billing info.
- Firestore: Retrieves real-time product details such as name, price, and quantity.

Data Models: Represent product details, user info, images, price, and quantity in structured format.

Features
1. User Authentication: Login/Sign-up with Firebase, password reset support.
2. Cart Management: Add/remove items, update quantities, real-time bill updates.
3. Billing: QR code scanning for product addition, ESP8266 support for smart cart communication.
4. Navigation: Multiple navigation flows for authentication, cart, and checkout.
5. Error Handling: Real-time exception handling with Kotlin Coroutines and Live Data.
6. Voice Search: Enabled using Broadcast Receiver with recognition fallback.
7. UI/UX: Clean layouts with Constraint Layout, Toast feedback, and conditional UI animations.

Build Configuration
- Gradle Plugins: Android Application, Kotlin Android, Google Firebase.
- Compatibility: Optimized for Android 11+ with backward compatibility.
- Debug/Release Modes: Configured for performance and testing.

V. RESULTS AND DISCUSSION

The smart shopping cart prototype was successfully implemented with Arduino Uno, RFID reader/tags, ultrasonic sensors, ESP8266 Wi-Fi module, and voice recognition. RFID detection accuracy reached 98%, and the human-following system maintained a stable 40–60 cm distance from the user. Real-time mobile app updates showed an average latency of 1.2 s, while voice commands achieved 92% recognition accuracy. The cart operated for about 3 hours on a full battery, effectively performing autonomous navigation, automated billing, and voice-assisted interaction in a simulated shopping environment.

Figure.11: Smart Shop Cart

VI. CONCLUSION AND FUTURE WORL

The Smart Shop Cart system successfully integrates human-following technology and a smart billing mechanism to provide users with a seamless shopping experience by reducing manual effort and eliminating long billing queues. Using RFID, ESP8266, and mobile app connectivity, the system ensures accurate product tracking, automatic billing, and real-time updates. This makes shopping more efficient, user-friendly, and time-saving. However, future work can focus on enhancing security with biometric authentication, integrating AI-based product recommendations, improving obstacle detection for smoother navigation, and adding multilingual voice assistance. Additionally, large-scale testing in real supermarket environments and integration with digital payment gateways can further strengthen its practicality and adoption.

REFERENCES

[1]. B. Siciliano and O. Khatib, *Springer Handbook of Robotics*, 2nd ed., Springer, 2016. Comprehensive resource on robotics concepts including motion planning, sensors, and human-robot interaction.

DOI: 10.17148/IJIREEICE.2025.13822

[2]. M. Asaduzzaman, M. Hasanuzzaman, and M. S. Kaiser, "Autonomous Human Following Robot Using Kinect Sensor," *2015 International Conference on Robotics and Automation (ICRA)*. Discusses human-following robots using depth and vision-based sensors.

[3]. Y. Wang, Y. Chen, and J. Zhang, "RFID-based Intelligent Shopping Trolley System," *2011 IEEE International Conference on Automation and Logistics*, pp. 422- 427. Relevant to automatic billing and RFID-based item tracking.

[4]. Shinde, S. P. Patil, et al., "Smart Trolley Using RFID," *International Journal of Computer Applications*, vol. 179, no. 7, 2018. Simple implementation of RFID-based smart trolleys.

