IoT Based Smart Trolley For Autobilling Using Arduino

Tanzila Sabir Patel, Omera Musahaf Ahemad Inamdar, Avanti Santosh Londhe, Parth Sunil Gigoo, Madhavi M. Netke

Abstract

In today's world, technologies in many fields such as artificial intelligence, machine learning, virtual reality, touch marketing, Internet of Things are changing day by day. The main purpose of this article is to focus on the customer's needs and goals because in the real world, time is more important for everyone. But people spend a lot of time in shopping malls. For example, in shopping malls, the amount of goods purchased by customers is used by shopping carts. After shopping, they may encounter problems such as long checkout lines and not knowing how to calculate payments. Therefore, taking these problems into consideration, we can find a solution called “IoT based smart business using BOLT”. Our IoT devices include barcode scanner, LCD monitor (5.5-8 inches) and Bolt ESP8266 (WI FI module). A rough explanation of the process is that when the customer drops off the product and puts it in the cart, the barcode scanner scans the product barcode, price and revenue and displays it on the digital panel. After the customer completes the shopping, the invoice is sent to the cash register. It will save time and inform customers in advance the calculation of what to buy.

Introduction

In the ever-evolving landscape of technology, the Internet of Things (IoT) has emerged as a revolutionary concept, connecting physical devices and enabling them to communicate, collect data, and perform tasks autonomously. One of the exciting applications of IoT technology is the development of a "Smart Trolley" designed to enhance the shopping experience for consumers and streamline the billing process for retailers. This project introduces the concept of an IoT-based Smart Trolley, empowered by Arduino, which aims to make the shopping experience more convenient and efficient while simultaneously simplifying the checkout and billing procedures in retail stores. Traditional grocery shopping has been a time-consuming task, often leading to long queues at the billing counters. The Smart Trolley project leverages the power of IoT to address this issue. By integrating sensors, RFID technology and user experience, smart cars can identify products and add them to the shopping cart. It also allows shoppers to view items in their shopping cart, track their spending, and receive instant digital invoices, eliminating the need, cost, and long waiting time for user manuals.
The main goal of this project is to show how IoT technology can be used together with Arduino microcontrollers to increase sales, make it easier for people to use and make it useful for retailers. By integrating RFID tags, weight sensors, and user interfaces, smart cars can seamlessly perform automated inventory, weighing, and payment processes. The system is also designed to reduce human error in billing and improve inventory management.

In this project, we will delve into the design and implementation of an IoT-based smart trolley system, discussing the required hardware and software components, how it works, and the benefits it provides to buyers and retailers. As IoT technology continues to transform many industries, the smart trolley project is a great example of how it can transform business and lead to greater connectivity and profitability in the market.

Existing System

The system must be reliable when scanning the product and respond accurately to the transaction, with all details sent to the online database. We have prepared a smart cart that allows customers to review products from the cart and complete the payment process. The customer must pick up the car, then scan their card and continue scanning the item. If he wants to remove an item from the cart, he has to copy it again. All items can go to order after checked. The customer then needs to scan his card to deduct the fee from the card. There is also the feature of checking the balance on the card. Customers can also view payment details online on the store's website.

The manager or store owner can also use the system with the help of the card given to him by the owner. Institutions allow the use of authorized key cards for transportation of customers. Only the owner can charge the customer's card, write the details on the RFID tag, add the customer to the system, give the card to the customer and store the information online.

The requisition process is an attempt to automate the cost process. A store that will benefit customers and buyers in many ways.
Methodology

The approach is important to develop an IoT based automatic billing smart cart using Arduino. The project should begin by defining requirements and objectives. Then carefully search and select the hardware and software tools needed to ensure compatibility. The architecture should establish the data flow and communication protocol of the system. Next comes hardware integration, where selected components (such as RFID scanners, load sensors, displays, and IoT modules) are connected to the Arduino board and pre-tested. Then, using the RFID system, the tags are read and the products are identified, and the RFID tags are signed for the products and prices. Integrate load sensors to recognize added and removed items and adjust items in the cart accordingly. Create IoT connections by configuring IoT modules to connect to selected platforms and write code to send product data to the cloud. Create a backend system to interact with the IoT platform, create an API endpoint for resource storage and billing. Create a database to store product information and pricing. Whether it is a mobile or web application, the user interface is designed, built and connected to the backend for real data analysis. Thorough testing and debugging is crucial to ensure smooth functionality. After successful evaluation, the system is deployed in the desired environment and users are provided with training and support. Maintenance, updates, information security, compatibility, evaluation and feedback are constant considerations for success. Careful planning and testing is important, and working with professionals can help achieve the desired and efficient performance.
Implementation

Complete IoT based smart cart for automatic billing using Arduino including step by step process. Below is detailed information about usage:

1. Hardware Assembly:
   - Physically connect the Arduino board, RFID scanner, load sensor, display and IoT module according to the data and pinout of the selected product.

2. Arduino code for RFID integration:
   - Write the Arduino code to interface with the RFID scanner. Program the Arduino to read RFID tags and identify objects. Create files or data models to represent RFID tags for products and prices.

3. Load Sensor Logic:
   - Implements code to detect additions and subtractions using load sensors. Update the load according to sensor data.

4. IoT Connectivity Settings:
   - Configure an IoT module (such as ESP8266 or ESP32) to connect to the device of your choice via WiFi or other optional IoT connectivity platforms (such as ThingSpeak, AWS IoT, Google Cloud IoT).

5. Arduino code for IoT data transfer:
   - Write Arduino code to transfer (add, remove) data to the IoT platform. Use appropriate libraries and techniques for communication.

6. Cloud backend development:
   - Build backend systems using programming languages such as Node.js or Python. The backend must interact with the IoT platform and provide an API end for data collection and billing.

7. Database settings:
   - Set up a database (such as MySQL, MongoDB) to store product data, price and cart history. Make sure the backend and IoT platform can access and update the data.

8. User Interface Development:
   - Design and build user interface mobile or web applications so customers can check out products, view the shopping cart, and check prices instantly. Combine the interface with the backend for data retrieval.

9. Testing and debugging:
   - Challenging testing of the entire system, from RFID scanning to IoT data transmission and billing. Identify and fix any errors or problems that may occur during testing. Do a check to make sure the system can handle more than one car at a time.

10. Deployment:
    - Use the system in a real environment such as a retail store. Make sure all components are installed correctly and connected to the Internet.
11. Training and user support:
- Teach store staff and customers how to use the smart car. To provide support to users in resolving questions or problems that may arise during operation.

12. Maintenance and updates:
- Item library and prices are updated regularly to stay current. Monitor the IoT infrastructure and perform necessary maintenance to ensure continuous operation of the system.

13. Data Security and Compliance:
- Use security measures to protect customers and ensure compliance with data privacy laws.

14. Analysis and Feedback:
- Collect feedback from store owners, employees and customers to proactively identify areas for improvement and future development. Use this feedback to refine and improve the system over time.
By following this step-by-step plan, you can create and deploy an IoT-based automatic billing cart using Arduino and make it work at the same time. You did a good job and achieved your goals.

**System Architecture**
Conclusion

Smart shopping carts are leading to new trends in supermarkets. The system makes it easier for customers to shop. The main incentives of the current system and the proposed system are to reduce customers' shopping time, avoid spending more time at the checkout, and ensure that customers know the exact price at the time of purchase. The system also reduces sales staff at the cash register. We can show the ESP8266 bolt tool in this cart because it can be used to easily monitor and control your system compared to other IoT devices. Bolt is a popular IoT platform. Bolt uses no transmissions or equipment as it sends data to other devices on WiFi networks. This shopping cart system makes the customer's job more efficient and increases customer satisfaction.

Reference


5. Shyamambika, N. and Thillaiarasu, N., 2016, January. A survey on acquiring integrity of shared data with effective user termination in the cloud. In Intelligent Systems and Control (ISCO), 2016 10th International Conference on (pp. 1-5). IEEE

