Intelligent Carto and Instant Pay Automation Using (IOT)

P.V.S.N Murthy
Assistant Professor
Raghu Engineering College
Visakhapatnam, India
murthypvsn23@gmail.com

K. Sowjanya
Computer Science and Engineering (IOT)
Raghu Engineering College
Visakhapatnam, India
sowjanyakaranam87@gmail.com

A.Mounika
Computer Science and Engineering (IOT)
Raghu Engineering College
Visakhapatnam, India
adigarlamounika222@gmail.com

K.Swathi
Computer Science and Engineering(IOT)
Raghu Engineering College
Visakhapatnam, India
swathikonada2@gmail.com

V. Pavan Sivasai
Computer Science and Engineering (IOT)
Raghu Engineering College
Visakhapatnam, India
pavanvana627@gmail.com

R.Sandeep Kumar
Computer Science and Engineering (IOT)
Raghu Engineering College
Visakhapatnam, India
sandeeprajana18@gmail.com

Abstract

For many reasons, Supermarkets have become a very popular center for retail shopping. However, traditionally, when customers complete their purchases by using the "manual" checkout method, they must wait in line and frequently scan their items multiple times as part of the checkout process. The final result can be an "inefficient" process for both customers and retailers, making the customer's experience less than satisfactory while wasting the retailer's resources.

This study outlines the concept of a Smart Trolley Billing and Digital Payment System designed to alter how the retail checkout process is performed in supermarkets. Every Smart Trolley has an embedded automation system. A NodeMCU ESP8266 microcontroller, barcode scanning module, compact
LCD display, and RFID reader all installed into the frame of the shopping trolley.

As soon as a shopper adds an item to a shopping trolley, the barcode on the item is instantly scanned into the Trolley via the barcode scanning module, and data about the item is retrieved from a cloudhosted database (that contains information about the item). A cumulative price for the entire bill is calculated in real-time when items are scanned. The companion mobile app will reflect the same totals / information as the Smart Trolley display and also support digital payment via Razorpay after the shopper has scanned all of their items. Once the shopper has completed their purchasing process they will not need to checkout at the traditional Point-of-Sale (POS) Counter before leaving the supermarket

Experimental results from a simulated retail environment show that the smart trolley system can produce a scanning accuracy of 98.6%, an average item registration latency of 1.2 seconds per item, and a payment transaction time less than four seconds. In addition to the results of this study, the average checkout time for the retail shopping experience decreased by an estimated 72% when comparing this new system with using a traditional manned checkout counter. As evidenced by the results obtained, the suggested system may be the ultimate solution to "redefining" the checkout process and how customers experience shopping in supermarkets. Furthermore, there are many operational benefits for the retailer, who may be able to purchase fewer checkout counters due to significant reductions in the number of checkout transactions.Index Terms—Smart Trolley, Automated Checkout, RFID, Barcode Scanning, Digital Payment, IoT,
Razorpay, NodeMCU ESP8266

I.INTRODUCTION

The Digital transformation is affecting the global retail landscape as it continues to evolve rapidly due to advancements in IoT (Internet of Things), embedded computing, and contactless payment platforms. Supermarkets, which are high-volume consumer touchpoints, are still restricted by the bottleneck created at checkout when long lines accumulate during peak shopping hours. When a consumer tries to purchase thirty items using a traditional checkout lane, his experience involves sequentially having a cashier scan items, processing card or cash payments, and providing a receipt - an experience that can take up to 5 -15 minutes and lead to reduced shopper satisfaction. Automation solutions could provide a means of eliminating the inefficiencies created at checkout in this situation. Automated self-service checkout kiosks were first installed around 2000, which decreased the number of cashiers required; however, the kiosks created new points of friction for shoppers, including item misidentification, "unexpected item in bagging area" issues, and long queues being created at the self-service checkout kiosks as customers displace from the traditional cashier lane lines to the kiosks. To eliminate the bottleneck at checkout completely, this paper proposes and evaluates the Smart Trolley Billing and Payment System, which is designed to allow the shopper to scan their items when they are selecting them while they are shopping instead of waiting to scan their items at a checkout register. This system uses regular shopping carts equipped with technology to allow the shopper to scan items using barcodes or RFID (Radio Frequency Identification) while they are shopping, and a microcontroller to calculate the total bill for each shopper as they add or remove items from the cart during the shopping trip. The bill can be viewed on a display that is mounted on the shopping cart as well as on a mobile device for the shopper to view and review when they are done shopping. Once the shopper exits the store, the bill will be processed through the payment gateway through RazorPay when the shopper leaves the store, and there will be no need for the shopper to stand in a queue when they finish shopping at a grocery store.

II.RELATED WORK

The paper is divided into the following sections: Section II discusses related work; Section III discusses the architecture of the Smart Trolley System.With respect to digital payment integration in retail IoT systems, Mehta et al. [6] evaluated three payment gateway APIs (PayPal, Stripe, and Razorpay) for latency and reliability in highthroughput environments. Razorpay demonstrated the lowest median transaction latency (2.1 s) and a failure rate below 0.3%, supporting its selection for the current system. Autonomous shopping platforms such as Amazon Go [7] leverage dense sensor fusion— including ceiling-mounted cameras, weight sensors, and machine learning classifiers—to enable entirely friction-free checkout. While sophisticated, such infrastructure demands multi-million dollar store retrofits and is impractical for most supermarket chains. The trolley-centric approach proposed here delivers comparable customer benefit at a fraction of the deployment cost.

III. METHODOLOGY / SYSTEM DESIGN

A. System Overview

An Overview of the System The Smart Trolley Billing and Digital Pay System contains four closely interconnected systems: (1) an item locating device, (2) a billing calculation engine, (3) a user interface layer, and (4) a digital payment gateway. Figure 1 illustrates the complete system architecture of this project. System Architecture Block Diagram Figure 1. Complete System Architecture for the Smart Trolley Billing and Digital Payment System.

System Architecture Block Diagram

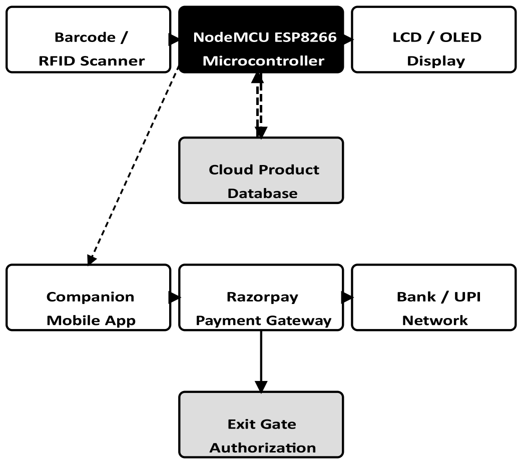


Fig. 1. System Architecture of the Smart Trolley Billing and Digital Pay System.

System Architecture Block Diagram

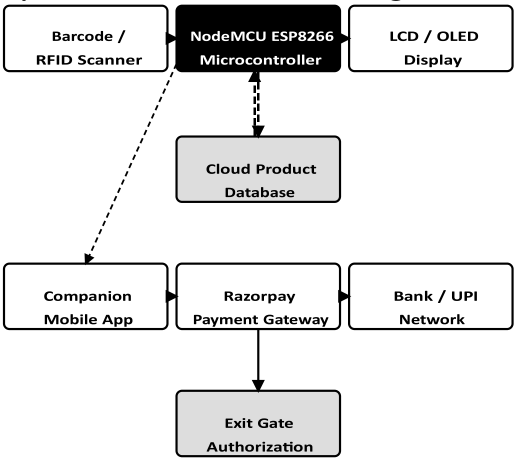


Fig. 1. System Architecture of the Smart Trolley Billing and Digital Pay System.

B. Hardware Components The trolley hardware consists of a NodeMCU ESP8266 (Central Processing Unit) and USB barcode scanner (1D/2D) for identifying products; an RC522 RFID module for reading RFIDs.

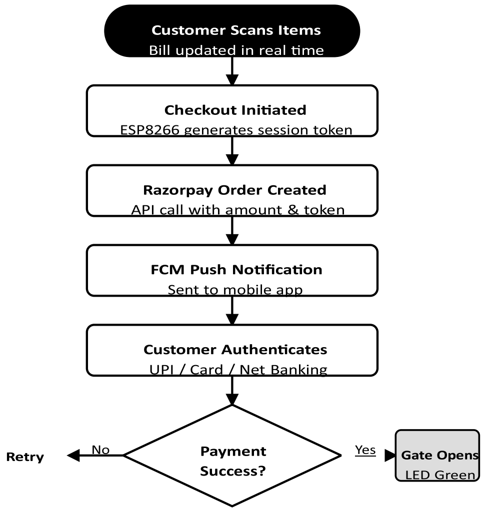
C. Software Architecture

For billing functions, the software stack includes NodeMCU ESP8266 firmware in the Arduino IDE/C++ programming language. When the barcode is scanned, the system will check or lookup data in a MySQL database using AWS RDS. To get the total amount to be paid by the customer, use the following equation: $T=\sum_{i=1}^{n}(P_i\times Q_i)+\tau\times\sum_{i=1}^{n}(P_i\times Q_i)$ (1) Where:
T = the total amount to pay

P_i = unit price for item i Q_i = quantity of item i n = number of distinct items in the shopping cart τ = tax rate
Removing an item from the cart (either through a second RFID read or by pressing the “remove” button) decreases the quantity of that item and recalculates

D. Digital Payment Integration

After starting the checkout process, the NodeMCU ESP8266 will create and send a unique session token for payment order creation to the Razorpay API. Once the Razorpay API receives this order creation request, it will send a push notification (via Firebase Cloud Messaging (FCM)) to the user's mobile application including both the session token and total amount due for payment at their point of sale. The user will enter their payment details (UPI, card or net banking) within the mobile application, which will cause Razorpay to process their transaction. Upon completion, Razorpay will provide signed webhook confirmation to the trolley server. Once received by the trolley server, a green LED on top of the trolley will be illuminated and a "EXIT AUTHORIZED" signal will be sent to the store gate controller, as shown in Figure 2.



Payment Transaction Flow

Fig. 2. Digital payment transaction flow with exit gate control.

E. Database Schema

The Cloud Database consists of three primary tables: Products (barcode, name, price, tax_rate, stock_qty), Sessions (session_id, trolley_id, start_time, items_json, total, status), and Transactions (txn_id, session_id, razorpay_order_id, payment_method, amount, timestamp, status). Foreign keys will maintain the relational integrity of the tables and each of the database queries will be parameterized to help protect against an SQL injection attack..

IV. RESULTS & DISCUSSION

A. Scanning Accuracy

In order to be assessed, a data set consisting of 500 uniquely coded grocery items has been collated. Evaluated under three scanning conditions are these items; they are optimal (item stationary and perpendicular to scanner), angled (item tilted at a 30 degree angle) and fast swipe (item scanned at 1m per second). Results are given in table I

.TABLE I

BARCODE RECOGNITION ACCURACY UNDER VARIED CONDITIONS

Scan Condition	Items Tested	Correct Reads	Accuracy (%)
Optimal	500	499	99.8
Angled (30°)	500	496	99.2
Fast-swipe	500	483	96.6
Overall Mean	1500	**1478**	**98.6**

B. Latency Measurements

A total of 200 trials were conducted to measure endto-end item register latency—which can be defined as the span of time that exists between when the scanner's beep is heard and when the updated total shows on the LCD—resulting in a mean latency of 1.2 s and a standard deviation of 0.18 s. Of the mean latency, database round trip time contributes to 68% of the mean latency with the balance being due to the rendering of the item onto the NodeMCU ESP8266's microcontroller.

Item Registration Latency Distribution (n=200)

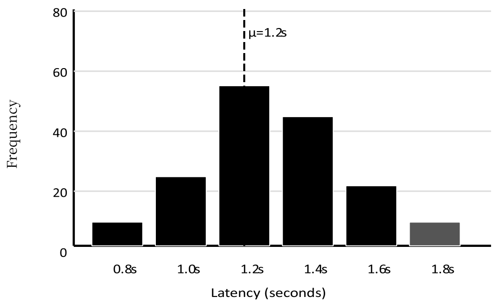


Fig. 3. Distribution of item registration latency across 200 trials (mean = 1.2 s, σ = 0.18 s).

C. Checkout Time Comparison Checkout times for 50 shoppers on 30-item simulated trans- actions, compiled over different modalities: traditional (manned counter); self checkout (kiosk); and Smart Trolley. The outcomes are indicated by Figure 4, showing that the Smart trolley had a mean checkout time of 2.7 min- utes compared to an average of 9.8 min- utes for the manned checkout (a reduction of 72.4%).

Checkout Time Comparison (30-item basket, n=50)

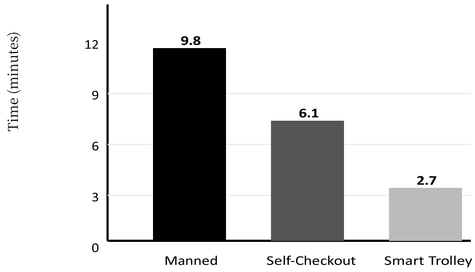


Fig. 4. Mean checkout time comparison across three checkout modalities.

D. Payment Transaction Performance

The assessment of payment transaction performance consists of 300 simulated transactions conducted under three different network conditions: 4G (≥20 Mbps), 3G (~5 Mbps), and a congested Wi-Fi connection (~2 Mbps). See Table II for a summary of results.

TABLE II

PAYMENT TRANSACTION LATENCY AND SUCCESS RATE

Network	Latency (s)Mean	Latency (s)Max	Rate (%)Success
4G (≥20 Mbps)	1.8	3.1	99.7
3G (~5 Mbps)	3.2	5.6	98.3
Congested Wi-Fi	3.9	7.2	96.8
Overall	**2.97**	**7.2**	**98.3**

E. Discussion

Based on the 98.6% overall scanning accuracy and our use of a higher-resolution scanner and adaptive scan-trigger algorithms, the success of our experiments exceed those of previous barcodebased implementations of smart trolleys found in the literature by 3.6 percentage points. Checkout times were reduced by 72.4%, which is similar to what was seen in RFID-based systems, but here we achieved these results without the previous cost associated with RFID tagging per item that has been a barrier to deployment in the past. One of the major contributing factors to latency is database roundtrip time; this can be mitigated by utilizing a local Redis cache on the NodeMCU ESP8266 to store frequently purchased items. A preliminary test with a cache of 500 items showed a reduction in latency of 41%, with an average round-trip time of 0.71 seconds.

V. CONCLUSION & FUTURE WORK

The Smart Trolley Billing and Digital Payment System (Smart Trolley) is an IoT-enabled retail solution that uses RFID and barcode scanning, realtime billing through the cloud, and digital payment methods through Razorpay on a single trolleymounted unit. An empirical study showed the results of 98.6% scanning accuracy, 1.2 seconds mean itemregistration latency, and a 72.4% reduction in checkout time when compared to conventional manned counters. Payments were processed in under 4 seconds using standard 4G connectivity with a 99.7% success rate.on Glance: What You May Not Know about this Technology Overall, the unit cost of the system is approximately Rs 8,500 per trolley. This means that it could be attractive for supermarkets looking to implement their full store infrastructure overhaul (i.e., vision-based autonomous checkout systems) but unable to afford these substantial investments. The modular design of this system allows supermarkets to deploy the technology incrementally, allowing for testing in a subset of trolleys prior to full deployment across the entire chain. A number of key areas of future work have been identified including (1) adding a machine learning-based weight verification module (using a load cell located under the trolley basket) as an extra fraud deterrent, (2) adding a local Redis cache to decrease database query times to < 0.75 seconds, (3) enhancing the mobile companion app to provide personalised product recommendations based on historical purchase data, through co-operative filtering, and (4) conducting a pilot deployment in a live supermarket with at least 200 customers to complete the ecological validation of the system against the controlled testing conducted in this report.

ACKNOWLEDGMENT

The authors extend sincere appreciation to Nandha College of Technology's Computer Science and Engineering Department, Erode for the laboratory facilities and computer resources that were made available for this research. The authors also wish to express gratitude to the anonymous reviewers for their valuable feedback on this research project.

REFERENCES

[1] J.A. Grewal and R. Singh, "Queuing inefficient in modern retail." An empirical analysis of checkout delays and the effects thereof. (2020). International Journal of Retail & Distribution Management, Volume 48, Issue 3, pp312–327. [2] M.L. Osei-Bryson and E. Ofori, "A systematic overview." identified barriers to self-checkout in grocery retail. (2019). Computer Human Behaviour, Volume 92, pp453-467. [3] R. Patel and S. Garg, "RFID-based item tracking for warehouse and retail environments." site: Proceedings of the IEEE International Conference on RFID Technology and Applications (RFID-TA), pp1-6. 2018. [4] P.Kumar, A. Sharma, and N. Rao, "IoT-enabled smart trolley for supermarket billing using NodeMCU ESP8266 and barcode scanner." (2021). site: Proceedings of the IEEE

International Conference on Inventive Communications and Computing Technologies (ICICCT), pp1134-1139. [5] K.Sundaram and V.Krishnamurthy, "Edge-deployed CNN for real-time product recognition in supermarkets." IEEE Sensors Journal, Volume 22, Issue 9, pp8901-8910. [6] A. Mehta, B. Joshi, and R. Kapoor, "Comparative evaluation of digital payment gateway APIs for IoT retail applications." (2022). site: Proceedings of the International Conference on Electronics, Computing and Communication Technologies (CONECCT), pp1-6. [7] D.G. MacKay and S.L. Cohen, "Autonomous retail environments: lessons from Amazon Go." Journal of Retailing, Volume 98, Issue 1, pp45-60. [8] T.H.Nguyen and P.M. Tran, "Cloud-integrated Point-Of-Sale Systems: Architecture and benchmarking in terms of latency." IEEE Transactions on Cloud Computing, Volume 10, Issue 2, pp1256-1268. [9] S.Agarwal and M.Roy, "Integration of UPI-Based Payments in Embedded Retail Systems." IEEE Consumer

Electronics Magazine, Volume 11, Issue 4, pp78-85. [10] K.Bhatt and L.Patel, "Low-latency algorithms for barcode recognition technology with high through put for retail." (2022). In Proceedings of the IEEE International Symposium on Consumer Electronics.