



International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

BOOKSHELF AUTOMATION SYSTEM

M Vignesh Kumar¹, P Prakash², S Ashok³, V Bharath⁴, T Surenbabu⁵

^{1,2} Assistant Professor, Department of Electrical and Electronics Engineering, P.A. College of Engineering and Technology, Pollachi, Tamil Nadu, India.

^{2,3,4} UG Student, Department of Electrical and Electronics Engineering, P.A. College of Engineering and Technology, Pollachi, Tamil Nadu, India.

DOI : <https://doi.org/10.55248/gengpi.07.1026.3282>

ABSTRACT :

Finding a particular book in a large collection is slow and error-prone, and books returned to the wrong shelf are usually noticed only when another reader fails to find them. This paper presents a Smart Bookshelf Automation System built around an Arduino UNO and an Arduino Nano together with RFID technology. Each book carries a unique RFID tag linked to a Book ID. When the user enters a Book ID on a 4×4 keypad, the Arduino UNO matches it against the stored Book ID data and lights the LED of the correct shelf, while a 16×2 LCD displays status messages such as Book Found, the shelf number and Invalid ID. When a book is returned, the user scans it with the RFID reader and an LED on the shelf indicates its place. In parallel, an RFID reader connected to the Arduino Nano checks the tags on the shelf, and if a book is detected in a position that does not match its expected tag, the Nano sounds a buzzer to alert the user. Using two controllers keeps book searching and misplacement monitoring independent of each other. In testing, both functions operated concurrently without interference. The system reduces manual searching time and human error, and is a simple, reliable and cost-effective solution for small and medium-sized libraries and personal book collections.

Keywords: Arduino UNO, Arduino Nano, RFID, RC522, Smart Bookshelf, Library Automation, Misplacement Detection.

1. INTRODUCTION

Libraries and personal book collections often hold hundreds or thousands of books spread over several racks and shelves. As a collection grows, finding one title by hand becomes time-consuming and error-prone, particularly when books have been returned to the wrong place. Printed catalogues and barcode-based systems record which books exist and who has borrowed them, but the user must still search the racks shelf by shelf to find the book itself.

Radio Frequency Identification (RFID) is a reliable and low-cost way to identify and track physical objects automatically. When each book carries a unique tag linked to its Book ID, its position on the shelf can be tracked without manual effort. Combined with a micro-controller such as the Arduino UNO, this makes it possible to build an automated book-locating system that guides the user directly to the right shelf.

This work presents a Smart Bookshelf Automation System that lets a user search for a book by entering its Book ID on a keypad. The matching shelf is indicated by an LED, and an LCD gives real-time feedback. A separate misplacement detection function uses an RFID reader and an Arduino Nano to sound a buzzer when a book is in the wrong position. The main objectives are: (i) to guide the user to the shelf of a book from its Book ID using an Arduino UNO, keypad, LEDs and LCD; (ii) to detect misplaced books with an RFID reader and an Arduino Nano and alert the user with a buzzer; and (iii) to provide a simple, reliable and cost-effective solution for small to medium-sized book collections.

2. LITERATURE SURVEY

Library automation uses electronic systems for book identification, inventory management, issue and return, and tracking. RFID-based library management systems have been widely reported [1], [4], [6], [8], and the use of RFID in libraries is reviewed in [10]. Unlike barcodes, RFID tags are read without line-of-sight contact, so fixed readers on shelves or gates can support automatic check-in and check-out and inventory auditing. The underlying technology is described in [11]–[14].

Low-cost embedded platforms have been used for such systems. Arduino-based smart shelf monitoring for inventory management [2], automated shelf management using embedded systems [3] and an Arduino Uno-based RFID library logging system [7] show that inexpensive controllers can read RFID modules such as the RC522 and drive peripherals such as keypads, LCDs and alerts. A book locator using RFID and IoT has also been proposed [5], and [9] surveys IoT architectures, technologies and applications.

Existing systems still have gaps. Most focus on inventory-level tracking, that is, whether a book is in the library or issued, rather than shelf-level guidance. Those that do guide the user often need several RFID readers, Wi-Fi modules or central servers, which raises cost and complexity. Real-time misplacement alerts are rarely provided as a standalone feature in low-cost systems, and single-controller designs can become a bottleneck when

handling user input, location indication and misplacement detection together. The proposed system addresses these gaps with two controllers: an Arduino UNO for Book ID-based location guidance and an Arduino Nano dedicated to misplacement detection.

3. MODELING AND ANALYSIS

3.1 Block Diagram

Figure 1 shows the block diagram of the proposed system. It has two paths that work independently. In the book location path, the Book ID entered on the keypad goes to the Arduino UNO, which drives the shelf LEDs and the LCD. In the misplacement detection path, the RFID reader sends the tag it reads to the Arduino Nano, which drives the buzzer. A DC supply powers the two controllers and the modules.

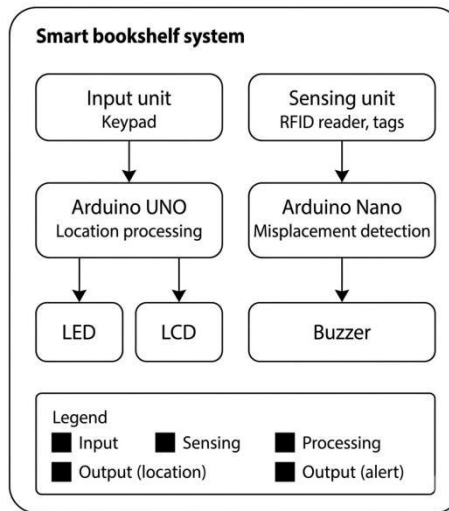


Figure 1: Block Diagram (System Architecture)

3.2 Hardware Description

The prototype uses two controllers. The circuit diagram is given in Figure 2 and the pin assignment is summarised in Table 1. The hardware consists of the following units.

- **Arduino UNO:** An open-source board based on the ATmega328P with 14 digital I/O pins (6 PWM), 6 analog inputs and a 16 MHz resonator. It reads the Book ID from the keypad, compares it with the Book ID data stored in its program memory and drives the shelf LEDs and the LCD.
- **Arduino Nano:** A compact board built around the same ATmega328P chip. It is dedicated to misplacement detection: it receives the tag data from the RFID reader, compares it with the expected tag and activates the buzzer on a mismatch.
- **RFID reader and tags:** An RC522 (MFRC522) reader communicates over SPI with passive tags fixed to the books, without line-of-sight contact. Each tag stores a unique identifier.
- **4×4 matrix keypad:** Its 16 keys are arranged in 4 rows and 4 columns and use 8 digital pins of the Arduino UNO. It is used to enter the numeric Book ID, which is confirmed with the '#' key.
- **16×2 LCD:** It shows two rows of 16 characters and is connected to the Arduino UNO either in parallel or through an I2C adapter. It displays messages such as Enter Book ID, Book Found, the shelf number and Invalid ID.
- **LED indicators:** One LED is placed at each shelf and connected to a digital output pin through a current-limiting resistor. It is switched on when the entered Book ID is matched.
- **Buzzer:** A piezoelectric buzzer connected to a digital output pin of the Arduino Nano. It sounds when a book is detected in the wrong position.

Table 1: Arduino pin assignment

S. No.	Module	Arduino Pin(s)	Function
1	Keypad rows	UNO: D9, D8, D7, D6	Row scanning of keypad
2	Keypad columns	UNO: A0, A1, A2, A3	Column scanning of keypad
3	16×2 LCD	UNO: D12, D11, D5, D4, D3, D2	Status display (RS, E, D4–D7)
4	Shelf LEDs (1–5)	UNO: A4, A5, D13, D10, A6	Shelf indication
5	RFID reader (RC522)	Nano: D10 (SS), D9 (RST), SPI	Tag detection
6	Buzzer	Nano: D8	Misplacement alert

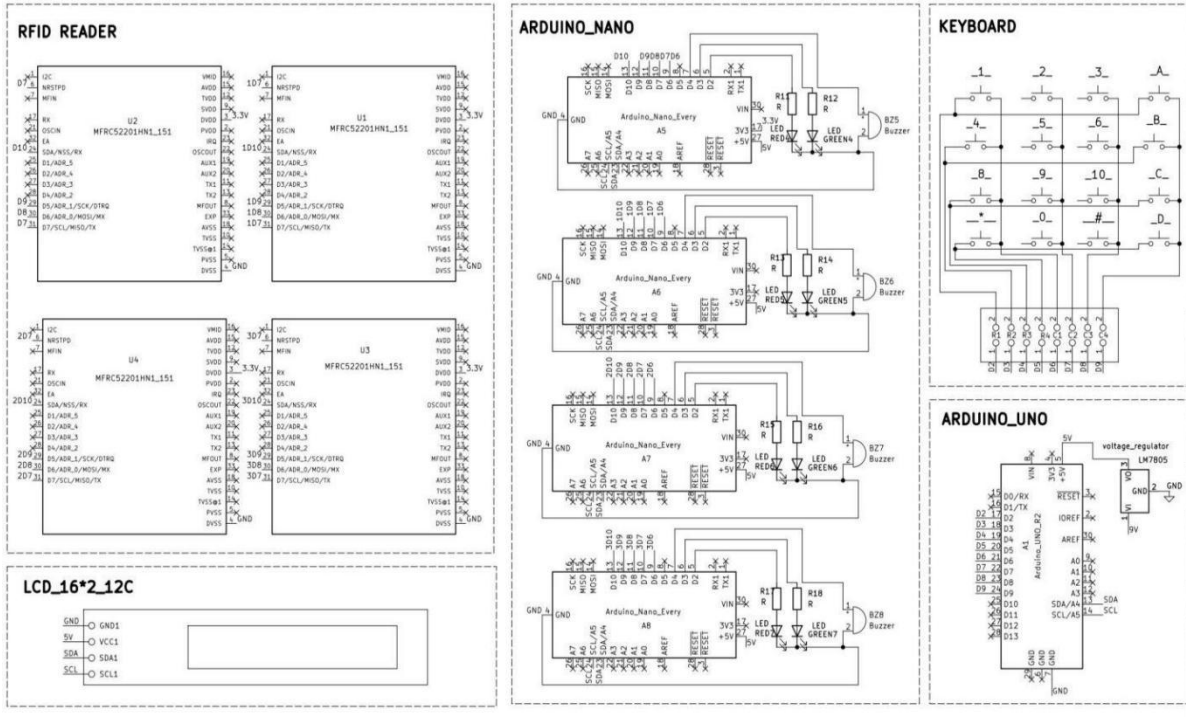


Figure 2: Circuit Diagram

3.3 Software Description

The Arduino IDE is used to write, compile and upload the program to both boards. The Keypad.h library scans the keypad, the LiquidCrystal library controls the LCD and MFRC522.h communicates with the RFID reader over SPI. The Serial Monitor was used to debug the Book ID matching and to verify tag readings.

The system has no external database. The Book ID-to-shelf mapping is stored in the program memory of the Arduino UNO as parallel arrays that link each Book ID to its LED pin. When a Book ID is entered, the program searches these arrays and returns the shelf number. This keeps the system self-contained, although the list has to be reprogrammed when new books are added. The software flow of the two functions is shown in Figure 3.

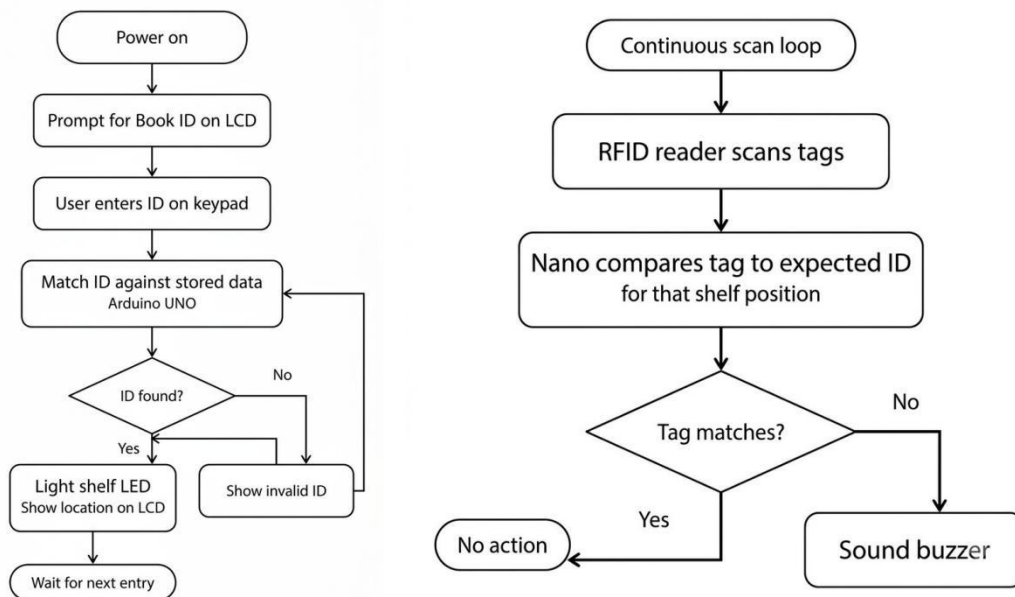


Figure 3: Flowcharts of (a) system workflow and (b) misplacement detection logic

3.4 Working Principle

On power-up the Arduino UNO initialises the keypad, LCD and LEDs, and the LCD asks for a Book ID. The user types the ID, which is echoed on the LCD, and presses '#' to confirm. The UNO compares it with the stored Book ID data. If a match is found, the LED of the corresponding shelf lights up and the LCD shows Book Found with the shelf number; otherwise the LCD shows Invalid ID and asks the user to enter the ID again. When a book is returned, the user scans it with the RFID reader and the LED on the shelf indicates where it belongs.

Independently of this, the Arduino Nano continuously polls the RFID reader. Each shelf position has an expected tag. If the detected tag matches the expected tag, no action is taken. If a tag belonging to a different position is found, the book is treated as misplaced and the Nano activates the buzzer for a predefined time. This logic only detects that a misplacement has occurred; where the book belongs is obtained when the user later searches for its Book ID on the keypad. Because the two functions run on separate controllers, neither blocks the other.

4. RESULT AND DISCUSSION

The prototype was built with an Arduino UNO, an Arduino Nano, an RC522 reader with tags fixed on sample books, a 4×4 keypad, a 16×2 LCD, shelf LEDs and a piezo buzzer, as shown in Figure 4. The Book ID data for each shelf position was pre-loaded into the Arduino UNO, and the system was tested under normal indoor lighting and typical shelf spacing.

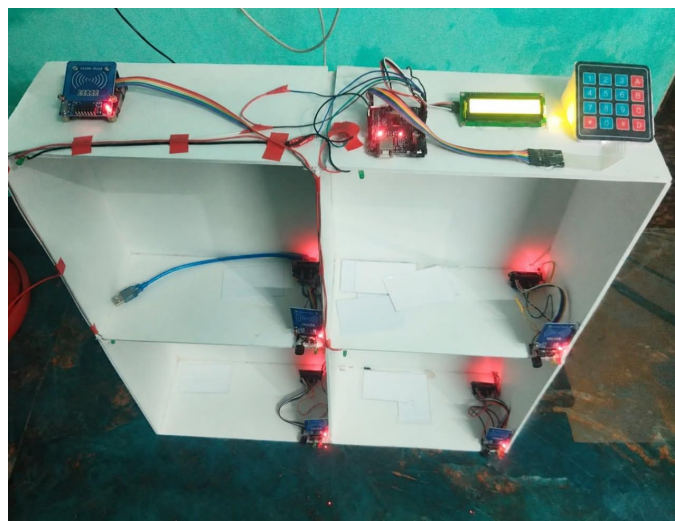


Figure 4: Hardware Setup of the Prototype

- **Book location guidance:** When a valid Book ID was entered, the LCD showed a confirmation with the shelf location and the correct LED lit (Figure 5). The time from confirming the entry to LED activation was near-instantaneous, well within a fraction of a second, because the lookup is a simple search through the stored data.
- **Invalid entry:** An unrecognised Book ID produced an error message on the LCD and a prompt to re-enter the ID, and no LED was switched on.
- **Misplacement detection:** When a tagged book was deliberately placed in a wrong shelf position, the reader detected the mismatch and the Nano sounded the buzzer within a short and consistent delay.
- **Simultaneous operation:** Location guidance on the UNO and misplacement detection on the Nano ran at the same time without interference, which supports the dual-controller design.

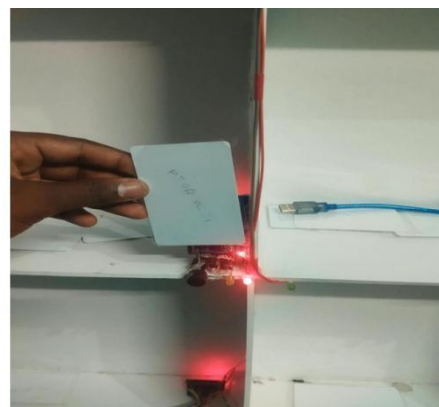


Figure 5: (a) LCD showing book location, (b) LED indication on the correct shelf, (c) buzzer alert on book misplacement

Compared with manual or catalogue-based searching, the system reduces the time and effort needed to find a book because the LED guides the user directly to the shelf. Misplacement is detected as soon as it happens instead of when another user fails to find the book. The system uses low-cost, readily available parts, and the keypad and LCD interface needs very little training. The estimated cost of the prototype is given in Table 2; the total of ₹6,400 makes the system economically viable for small and medium-sized libraries and personal collections.

Table 2: Cost estimation of the prototype

S. No.	Component	Qty	Unit Price (₹)	Total (₹)
1	Arduino Nano	4	300	1200
2	Arduino UNO	1	550	550
3	RFID module	4	260	1040
4	Red LED	10	5	50
5	Green LED	10	5	50
6	Buzzer	4	150	600
7	4×4 keypad	1	250	250
8	I2C LCD display	1	250	250
9	9 V adapter	1	200	200
10	5 V regulator	1	30	30
11	Switch	1	30	30
12	Sheet 1×10	1	1500	1500
13	Jumper wires M-M	1	150	150
14	Jumper wires M-F	1	150	150
15	Jumper wires F-F	1	150	150
16	PCB board THT	1	200	200
Total				6,400

5. CONCLUSION

The Smart Bookshelf Automation System was designed and implemented using an Arduino UNO and an Arduino Nano with RFID, a keypad, an LCD, LED indicators and a buzzer. The user enters a Book ID, the Arduino UNO lights the LED of the matching shelf and the LCD gives status feedback, while the RFID reader and the Arduino Nano independently watch for misplaced books and sound a buzzer. Testing showed that both functions worked reliably and at the same time, which validates the dual-controller approach. The system reduces manual search time and human error at low cost.

The prototype uses a fixed Book ID list stored in the Arduino UNO and detects that a misplacement has occurred without locating the book. Future work includes storing book records on an SD card, EEPROM or external database, adding Wi-Fi or Bluetooth for remote search through a mobile app, using multiple RFID readers to find the exact shelf of a misplaced book, automatic registration of new books, voice-based search, and a networked multi-node design for larger libraries.

6. REFERENCES

1. Yusuf, M., Akintunde, A., Habeeb, S. and Quadri, A. (2023) "Radio Frequency Identification (RFID) Based Library Management System," International Journal of Technology Systems, Vol. 8, No. 2, pp. 21–35.
2. Sharma, S. and Verma, A. (2021) "Arduino Based Smart Shelf Monitoring System for Inventory Management," International Journal of Innovative Technology and Exploring Engineering, Vol. 10, No. 6, pp. 88–93.
3. Kumar, N. and Reddy, K.V. (2020) "Design and Implementation of Automated Shelf Management System Using Embedded Systems," Journal of Embedded Systems and Applications, Vol. 5, No. 2, pp. 45–52.
4. Zulkiply, M.A.Z. and Mohd Noh, F.H. (2020) "Library Book Search and Management System Using RFID Technology," Progress in Engineering Application and Technology, Vol. 1, No. 1, pp. 303–316.
5. Gupta, R. and Sharma, P. (2019) "Smart Book Locator System Using RFID and IoT," International Journal of Computer Applications, Vol. 178, No. 32, pp. 20–24.
6. Bansal, A. and Kumar, S. (2018) "RFID Based Library Management System Using Arduino," International Journal of Engineering Research and Technology, Vol. 7, No. 5, pp. 112–116.
7. Morallo, N.T. (2017) "Library Logging System Design Based on Radio Frequency Identification Interface using Arduino Uno Board as Platform," JPAIR Multidisciplinary Research, Vol. 30, No. 1, pp. 103–114.
8. Hassan, A.S. and Hussein, R.T. (2017) "Simulation of Radio Frequency Identification Based Library Management System," Journal of Engineering and Sustainable Development, Vol. 21, No. 4, pp. 104–113.
9. Patel, K. and Patel, H. (2016) "Internet of Things: Survey on Architecture, Enabling Technologies, Applications and Challenges," International Journal of Engineering Science and Computing, Vol. 6, No. 5, pp. 6122–6131.
10. Sabit, H., Alwadi, A., Kilby, J. and Gawanmeh, A. (2016) "Radio Frequency Identification Technology in Libraries," International Journal of Engineering Research & Science, Vol. 2, No. 11, pp. 65–77.
11. Chawla, V. and Ha, D.S. (2007) "An Overview of Passive RFID," IEEE Communications Magazine, Vol. 45, No. 9, pp. 11–17.

-
12. Want, R. (2011) "Near Field Communication," IEEE Pervasive Computing, Vol. 10, No. 3, pp. 4–7.
 13. Want, R. (2006) "An Introduction to RFID Technology," IEEE Pervasive Computing, Vol. 5, No. 1, pp. 25–33.
 14. Landt, J. (2005) "The History of RFID," IEEE Potentials, Vol. 24, No. 4, pp. 8–11.
 15. Rao, K.V.S., Nikitin, P.V. and Lam, S.F. (2005) "Antenna Design for UHF RFID Tags: A Review and a Practical Application," IEEE Transactions on Antennas and Propagation, Vol. 53, No. 12, pp. 3870–3876.