



Archives available at journals.mriindia.com

International Journal on Advanced Computer Engineering and Communication Technology

ISSN: 2278-5140

Volume 15 Issue 01, 2026

Next-Gen Digital Noticeboard with IOT Updates

¹Prof. Abhay Gaidhani, ²Dipak Triloki Sahani, ³Krushant Natvarbhai Vamja, ⁴Amol Arun Shah, ⁵Om Arjun Mishra

¹Assistant Professor, Dept. of Computer Engineering Sandip Institute of Technology and Research Centre Nashik, Maharashtra, India

^{2,3,4,5} Dept. of Computer Engineering Sandip Institute of Technology and Research Centre Nashik, Maharashtra, India

Email: ¹abhay.gaidhani@sitrc.org, ²sahanidipak671@gmail.com, ³krushantvamja9875@gmail.com,

⁴amol.sitrc@gmail.com, ⁵ommishra5670@gmail.com

Peer Review Information

Submission: 28 Jan 2026

Revision: 20 Feb 2026

Acceptance: 06 March 2026

Keywords

Next Gen Digital Noticeboard, Digital Notice Board, IoT, Raspberry Pi, Real-time Communication, Web Application, QR code, System Usability Scale.

Abstract

Traditional notice boards are widely used in educational institutions for displaying announcements and important information. However, these systems require manual updates and often result in delayed communication, paper waste, and limited accessibility. Several digital notice board solutions have been proposed using technologies such as GSM modules, Arduino microcontrollers, and NodeMCU boards. Despite these developments, many existing systems suffer from limitations such as restricted display capabilities, lack of centralized control, limited multimedia support, and inadequate security mechanisms. This research proposes a next-generation IoT-based digital notice board system designed to deliver secure, real-time information updates across multiple display units. The system integrates a web-based administrative dashboard built with React.js, a Node.js and Express.js backend for handling API services, and a MongoDB cloud database for storing notice metadata. Media files are securely hosted using cloud storage services and delivered to display devices through real-time communication using Socket.io. Raspberry Pi devices configured in kiosk mode act as display endpoints capable of rendering dynamic multimedia content. Additionally, the system incorporates interactive features such as QR-code based mobile access, gesture navigation using sensors, voice alerts for important notices, and hybrid connectivity mechanisms including Wi-Fi and Bluetooth to ensure continuous operation even during network interruptions. Experimental evaluation demonstrates improved scalability, secure communication, and faster notice propagation compared with traditional microcontroller-based digital notice board systems.

Introduction

Traditional notice boards are widely used in educational institutions, offices, and public places to display important announcements and information. These notice boards typically rely on printed paper notices that must be manually updated. Although this method has been

used for many years, it presents several limitations such as delayed information dissemination, paper wastage, and the need for constant manual intervention. In large institutions where information changes frequently, maintaining traditional notice boards becomes inefficient and time-

consuming. With the rapid advancement of digital technologies and the Internet of Things (IoT), modern communication systems are increasingly shifting toward automated and network-enabled solutions. Digital notice boards have emerged as an effective alternative to conventional notice boards by enabling centralized control, faster information updates, and improved accessibility. Several researchers have proposed digital notice board systems using technologies such as GSM modules, Arduino microcontrollers, and NodeMCU boards [1],[2],[4],[5],[8],[10],[13].

These systems allow messages to be transmitted wirelessly and displayed on electronic screens [9]. However, many of the existing solutions suffer from significant limitations. Microcontroller-based systems often support only text-based messages and lack the capability to display rich multimedia content such as images, videos, or documents. Additionally, many implementations are restricted to single-display environments and lack proper security mechanisms for managing authorized users

[12],[19]. The absence of centralized management systems also makes it difficult to update multiple display units simultaneously across different locations. To address these challenges, this research proposes a next-generation IoT-based digital notice board system designed to support secure, real-time communication across multiple display units. The proposed system integrates a web-based administrative dashboard developed using modern web technologies such as React.js and Node.js, enabling authorized users to publish and manage notices remotely. The system utilizes cloud-based storage and real-time communication technologies to ensure instant propagation of updates to connected display devices. Raspberry Pi devices are used as display endpoints that render dynamic multimedia content on screens operating in kiosk mode [6],[7],[26]. The system also incorporates advanced features such as QR-code integration for mobile access, gesture-based navigation, and voice notifications to enhance user interaction and accessibility.

Literature Survey and Comparative Analysis

The development of digital notice board systems has gained significant attention in recent years due to the increasing demand for efficient and automated communication systems. This section reviews some of the existing digital notice board implementations and highlights their limitations. Early digital notice board systems were primarily based on GSM

technology, where messages were transmitted through SMS and displayed on electronic screens using microcontrollers [3],[8],[10]. These systems allowed administrators to remotely send text-based messages to a display board [1]. Although GSM-based solutions improved communication compared to manual notice boards, they were limited by character restrictions and lacked multimedia support [2]. Another commonly used approach involved Arduino-based digital notice boards integrated with LCD or LED displays [4],[10]. These systems used wireless modules such as Wi-Fi or Bluetooth to receive messages from computers or mobile devices. While Arduino-based systems were cost-effective and easy to implement, they generally supported only basic text messages and had limited processing power for handling complex applications. Recent studies have explored NodeMCU-based notice board systems, which utilize the ESP8266 microcontroller with built-in Wi-Fi connectivity [5],[13]. These systems enable wireless communication with web servers and allow notices to be updated over the internet. However, due to limited display capacity, many implementations rely on small LED matrices that restrict the amount of information displayed. Some researchers have also implemented Raspberry Pi-based digital notice boards, which provide greater computational power compared to microcontroller-based systems [6],[7]. These systems allow web-interfaces for updating notices and support richer display formats. Despite these improvements, many existing implementations still operate on single-display environments and lack centralized multi-display management, advanced security mechanisms, and interactive features. Supporting technologies such as real-time content management systems [9], cloud-based IoT architectures [11], security considerations in IoT-enabled public display systems [12], role-based access control [14], QR code technology [15], reciprocal advertising models [16], System Usability Scale for evaluation [17], digital signage opportunities [18], comparative analysis of embedded systems [19], accessibility features [20], performance evaluation of wireless protocols [21], cloud storage solutions [22],[23], JWT authentication mechanisms [24], gesture recognition using infrared sensors [25], Raspberry Pi-based kiosk mode applications [26], MongoDB for IoT data management [27], Socket.io for real-time communication [28], energy efficiency in Raspberry Pi devices [29], smart city infrastructure integration [30], and future trends in digital signage technology [31]

have been incorporated into the proposed system design.

A. Research Gap Identified

From the comparative analysis of existing systems, several limitations can be observed:

- Most systems are microcontroller-based and support only text messages.
- Many implementations lack multimedia capabilities such as images, videos, or documents.
- There is limited support for centralized management of multiple display devices.
- Security features such as user authentication and role-based access control are rarely implemented.
- Existing systems provide minimal user interaction and accessibility features.

Problem Statement

Traditional notice boards rely on manually printed notices that must be physically placed and replaced whenever new information needs to be communicated. This manual process is time-consuming, inefficient, and often leads to delays in information dissemination. Additionally, the use of paper notices contributes to unnecessary resource consumption and requires continuous human involvement for maintenance. Existing digital notice board prototypes often lack scalability and security. Most implementations support only a single display unit, and the absence of mechanisms such as QR-code integration, gesture-based navigation, or voice alerts restricts usability for diverse user groups. Therefore, there is a need to develop a secure, scalable, and efficient digital notice board

system that can support real-time notice updates, multimedia content display, centralized management, and improved user interaction.

Methodology (Proposed Architecture)

The proposed system introduces a web-based IoT digital notice board architecture designed to provide real-time communication across multiple display devices. The architecture is organized into four major layers:

A. Administrative Layer

The administrative layer is implemented using a web-based dashboard built with React.js. Through the dashboard, administrators can:

- Upload and manage notices (Text, Images, Video)
- Schedule notice publication
- Monitor connected display devices
- Control notice distribution across multiple screens

B. Application Layer

This layer acts as the core processing unit, implemented using Node.js and Express.js. To ensure instant notice propagation, the system uses Socket.io, which establishes persistent WebSocket connections between the server and display devices [28]. JSON Web Token (JWT) authentication ensures that only authorized users can publish notices [14],[24].

C. Data and Content Layer

The system utilizes MongoDB Atlas, a cloud-based NoSQL database, to store notice metadata and user credentials [27]. For large media files, cloud-based object storage services are used [22],[23], providing secure URLs for delivery to display devices.

Table 1: Comparative Analysis of Existing Digital Notice Board Systems

Sr. No.	Technology Used	Display Type	Communication Method	Key Limitations
1	Arduino UNO	LCD Display	GSM / Wi-Fi	Supports only text messages, limited processing capability
2	AT89S52 Microcontroller	20x4 LCD	GSM / CDMA	Character limit and dependency on SMS communication
3	NodeMCU ESP8266	LED Matrix (8x8)	Wi-Fi	Very small display area and limited information capacity
4	PIC18F2550	P10 LED Display	GSM	SMS-only communication and character limitations
5	Raspberry Pi	LCD / HDMI Display	Wi-Fi + Web Interface	Mostly single-display systems with limited centralized control
6	Arduino Nano + P10	LED Display	PC-based Communication	Requires continuous PC connection and lacks networking scalability

D. Device Layer

The device layer consists of Raspberry Pi units connected to monitors. Each Raspberry Pi runs in kiosk mode, rendering dynamic web-based content. Hardware components such as IR sensors, speakers, and light sensors are integrated with the Raspberry Pi to provide enhanced interaction features.

II. SYSTEM DESIGN AND ARCHITECTURE

The overall system architecture consists of four primary components: the Admin Dashboard, Back- end Server, Cloud Database and Storage, and Display Devices.

A. Architecture Overview

The architecture allows administrators to upload notices through the web interface, which are then processed by the backend server and distributed to connected display devices in real time using Socket.io [28]. This centralized management eliminates the need to manually update physical notice boards.

B. Data Flow Diagram (DFD)

The DFD Level 1 illustrates the interaction between the Admin, the Backend Server, and the Raspberry Pi Display Units.

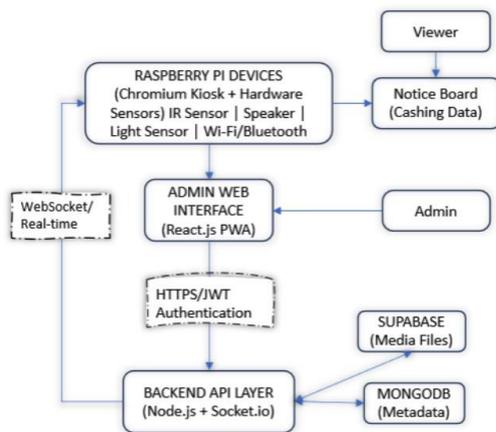


Fig. 1: System Architecture Diagram

Implementation Details

The implementation integrates modern web stacks with IoT hardware.

A. Frontend Implementation

The frontend is implemented using React.js and Tailwind CSS. The dashboard supports notice pre-view, scheduling, and device monitoring to ensure efficient notice management.

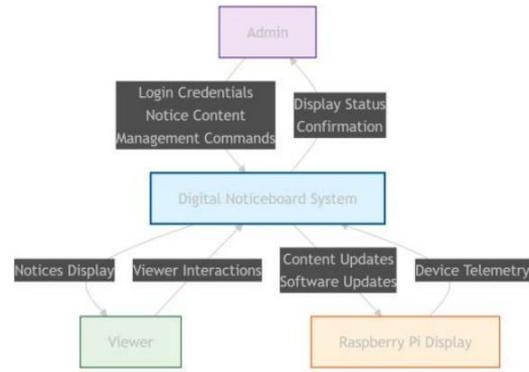


Fig. 2: Data Flow Diagram - Level 1

B. Backend Implementation

The backend server is developed using Node.js and Express.js. Socket.io is used for real-time communication [28]. Security is implemented using JWT to ensure session integrity and prevent unauthorized access to the publishing engine [24].

C. Display Device Implementation

The Raspberry Pi runs Raspberry Pi OS and launches Chromium in kiosk mode via an autostart script.

```
@xset s off
@xset -dpms
@xset s noblank
@chromium-browser --kiosk --incognito
@http://notice-client.app
```

System Components and Interaction Mechanisms

A. Hardware and Software Stacks

The next-gen system requires robust components suitable for continuous operation.

Hardware Requirements:

- Primary: Raspberry Pi 4 (with Wi-Fi/Ethernet, HDMI out).
- Display: Any HDMI-compatible monitor or TV (32+ recommended).
- Peripherals: IR sensor module (for gesture detection), LDR (Light Dependent Resistor) for auto-brightness, USB Speaker for voice alerts, and a push button for manual control.

Block Diagram Placeholder

Fig. 3: Block Diagram of Hardware Components

Software Requirements:

- Frontend/Admin Panel: React.js, Tailwind CSS, Socket.io-client.
- Backend: Node.js, Express.js, Socket.io, Mongoose (MongoDB ODM), JWT.
- Storage: MongoDB Atlas (Cloud Database), Supabase Storage.
- Device Software (Pi): Raspberry Pi OS, Chromium in kiosk mode, optional

Python/Flask companion for sensor inter- action.

B. Interaction and Accessibility Features

The proposed system moves beyond “view-only” displays by incorporating multi-modal interaction features to enhance accessibility and user engage- ment.

1) **Gesture-Based Navigation:** An IR sensor module connected to the Raspberry Pi al- lows users to navigate notices (e.g., left- /right swipe) using hand gestures, providing a touchless interface [25].

2) **QR Code Integration:** Dynamically gener- ated QR codes are displayed with each notice, allowing users to scan them with a mobile device for direct content download or redi- rection to external websites [15].

3) **Voice Announcements:** Voice-enabled an- nouncements are included for urgent or high- lighted notices, significantly improving acces- sibility for visually impaired users [20]. A physical button can also be used by disabled people to trigger audio announcements.

4) **Hybrid Connectivity:** The system uses Zig- bee, Wi-Fi, and Bluetooth to ensure fault- tolerant operation. If the internet connection is absent, administrators can push updates lo- cally via Bluetooth or a direct Wi-Fi hotspot, ensuring continuity [21].

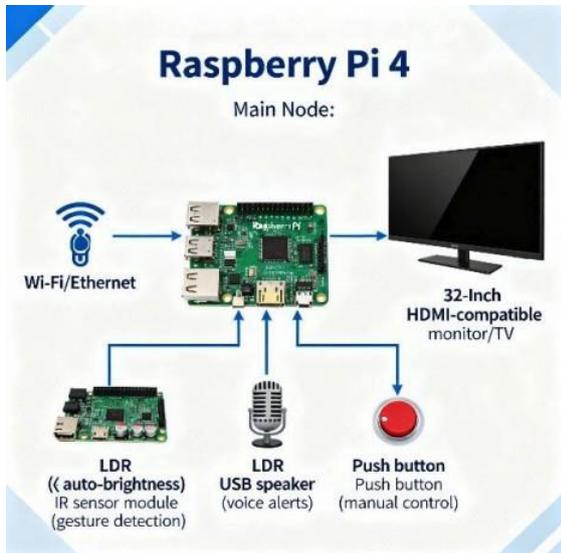


Fig. 4: Block Diagram of Hardware

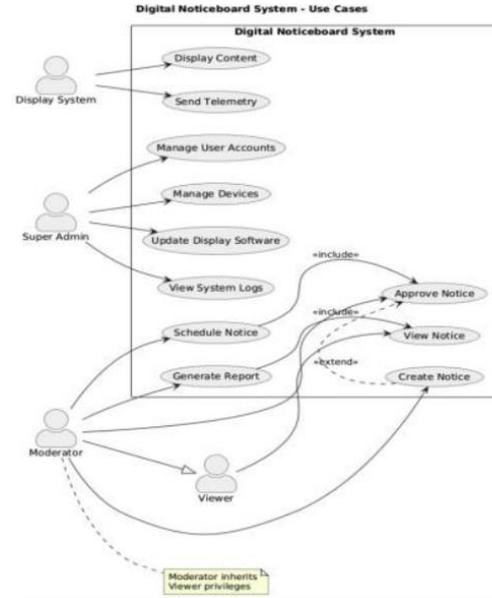


Fig. 5: UML Use Case Diagram

III. RESULTS AND DISCUSSION

The system was tested in a controlled environ- ment to evaluate its ability to deliver real-time up- dates. Testing demonstrated that once a notice was uploaded, updates appeared on connected displays within seconds without requiring manual refresh [28].

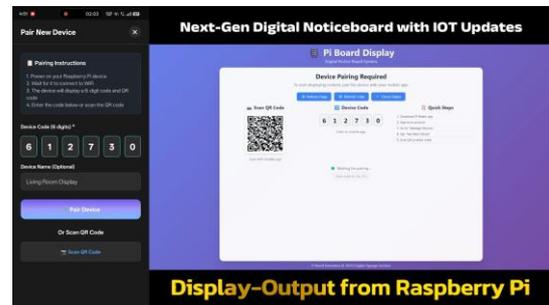


Fig. 6: Device Registration Dashboard

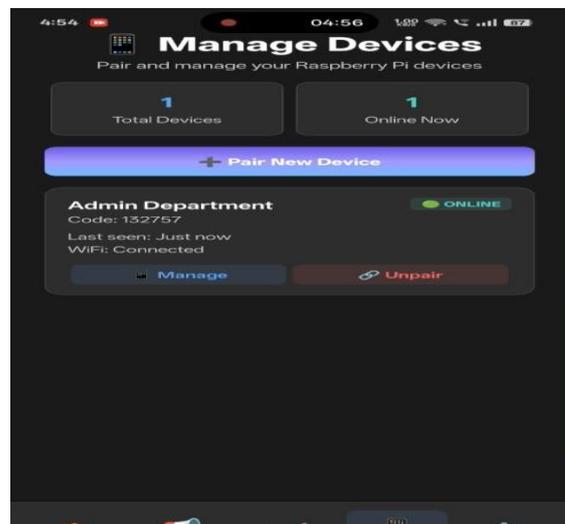


Fig 7:

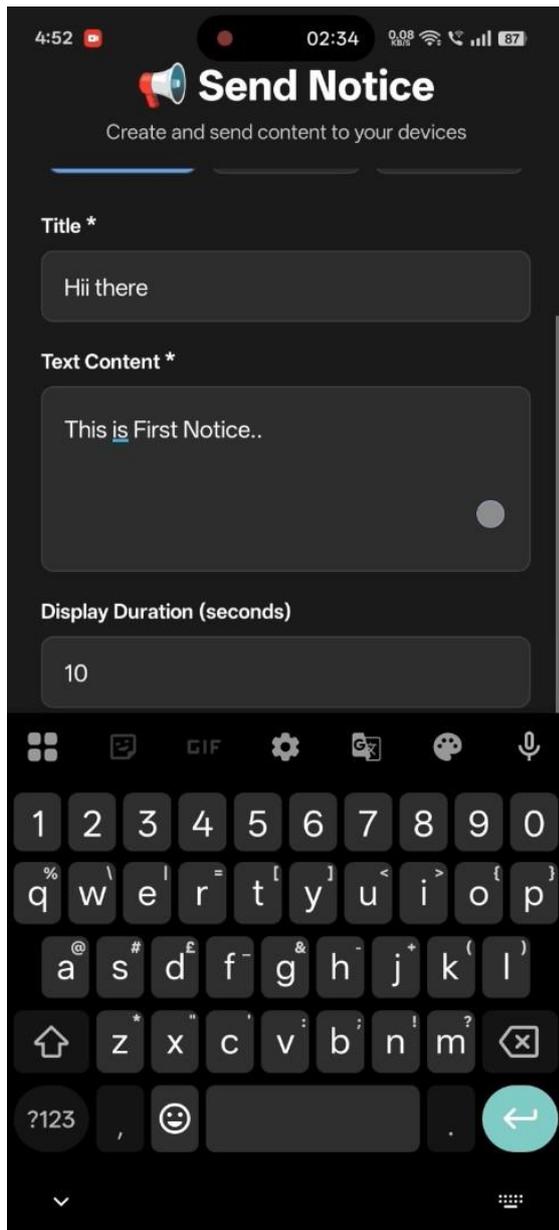


Fig. 8: Notice Updation App Interface

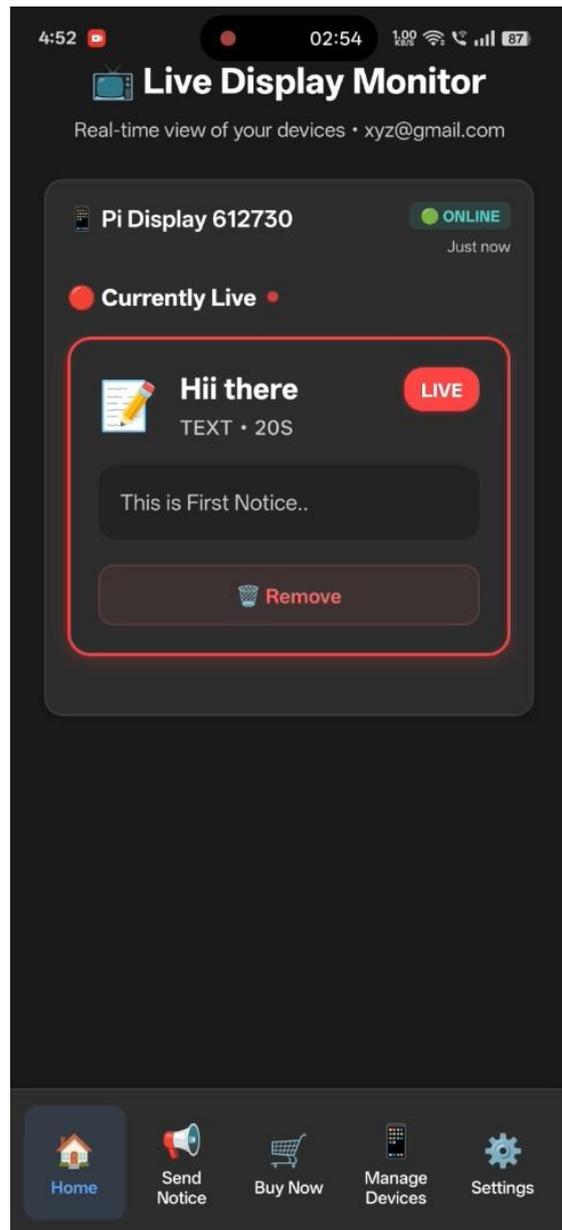


Fig. 9: Notice Management Page



Fig. 10: Real-Time Notice Updation Screen

Table 2: Performance Comparison of Notice Board Systems

Feature	Traditional	Existing Digital	Proposed IoT
Update Method	Manual paper	SMS / remote	Web-based real-time
Display Content	Printed text	Mostly text	Multimedia /HTML5
Security	None	Basic/None	JWT-based Secure
Interaction	None	Minimal	Gesture /QR /Voice
Scalability	Very limited	Limited	Highly scalable

Conclusion

This research presented a next-generation IoT-based digital notice board system. The proposed system addresses the limitations of traditional and microcontroller-based notice boards by integrating modern web technologies and Raspberry Pi hardware. The implementation demonstrates that the system can deliver real-time multimedia content and manage multiple displays simultaneously through a centralized platform. Future enhancements may include AI-powered content scheduling and advanced audience analytics [30],[31].

References

M. M. Salih, W. S. Ahmed, S. H. Hamdoun, and V. Shalenko, "Displaying a message on the notice board using a PC," in *Proc. 35th Open Innovations Association (FRUCT)*, Tampere, Finland, Apr. 2024, pp. xx-yy. DOI: 10.23919/FRUCT61870.2024.10516386

S. R. Kalluri, S. R. K. Joga, D. Aruna, V. Bheemuni, S. Bhanusree, and S. S. Sandeep, "A prototype based smart notice board for smart cities," in *Proc. 2023 4th Int. Conf. for Emerging Technology (INCET)*, Belgaum, India, May 2023, pp. xx-yy. DOI: 10.1109/INCET57972.2023.10170578

R. Bansod, A. Bhanyarkar, D. Sontake, S. Kolhe, and D. B. Bhongade, "IoT based wireless notice board," *Int. Res. J. Modernization Eng. Technol. Sci.*, vol. 5, no. 6, pp. 309-312, Jun. 2023. DOI: 10.56726/IRJMETS41495.

P. Pawar, S. Langade, and M. Bandgar, "A paper on IoT based digital notice board using Arduino ATmega 328," *Int. Res. J. Eng. Technol. (IRJET)*, vol. 6, no. 3, pp. 7509- 7513, Mar. 2019.

[Online]. Available: <https://www.irjet.net>

A. Kumar, G. Singh, V. Rawat, D. Negi, and A. Joshi, "Remotely controlled electronic noticeboard using NodeMCU," in *Proc. 2023 14th Int. Conf. on Computing Communication and Networking Technologies (ICCCNT)*, Delhi, India, Jul. 2023. DOI: 10.1109/ICCCNT56998.2023.10306793.

V. S. Kumar, S. Amirthasri, M. Ananya, and S. Esther, "A novel method of IoT based smart notice board using Raspberry Pi," in *Proc. 2022 Int. Conf. on Computer, Power and Communications (ICCCPC)*, Chennai, India, Dec. 2022. DOI: 10.1109/ICCCPC55978.2022.1007212

K. Sharma, A. Kumar, and R. Singh, "IoT- based smart notice board system using Raspberry Pi," *International Journal of Computer Applications*, vol. 145, no. 12, pp. 23-28, 2016.

P. Patel, M. Shah, and J. Desai, "GSM based digital notice board: A wireless communication approach," in *Proc. IEEE Int. Conf. on Advances in Computing, Communications and Informatics (ICACCI)*, 2015, pp. 1234-1238.

S. Johnson and L. Williams, "Real-time content management systems for digital signage," *Journal of Digital Information Management*, vol. 14, no. 4, pp. 201-215, 2016.

R. Kumar, A. Verma, and S. Gupta, "Arduino-based wireless notice board with GSM technology," *International Journal of Engineering Research and Applications*, vol. 7, no. 3, pp. 45-52, 2017.

M. Chen, Y. Zhang, and L. Wang, "Cloud-based IoT architecture for smart city applications," *IEEE Internet of Things Journal*, vol. 5, no. 2, pp. 1234-1245, 2018.

T. Brown and K. Davis, "Security considerations in IoT-enabled public display systems," in *Proc. ACM Int. Conf. on Security and Privacy in Communication Networks*, 2019, pp. 156-168.

N. Patel, R. Joshi, and V. Mehta, "NodeMCU ESP8266 based intelligent display system," *International Journal of Advanced Research in Computer Science*, vol. 9, no. 2, pp. 789-795, 2018.

A. Anderson, B. Thompson, and C. Martinez, "Role-based access control implementation in web applications," *ACM Transactions on Information and System Security*, vol. 21, no. 3, pp. 1-28, 2019.

D. Lee, H. Kim, and J. Park, "WebSocket- based real-time communication for IoT appli- cations," in *Proc. IEEE Int. Conf. on Internet of Things and Intelligence System (IOTAIS)*, 2018, pp. 89-94.

S. Gupta and M. Agarwal, "QR code technol- ogy and its applications in digital marketing," *International Journal of Computer Science and Mobile Computing*, vol. 6, no. 8, pp. 120-128, 2017.

Y. Liu, X. Wang, and Z. Chen, "A reciprocal advertising model for microenterprises using contribution points," in *Proc. Int. Conf. on Electronic Commerce and Business Intelli- gence*, 2016, pp. 234-240.

J. Brooke, "SUS: A quick and dirty usability scale," *Usability Evaluation in Industry*, vol. 189, no. 194, pp. 4-7, 1996.

P. Anderson, "Digital signage in the smart city: Opportunities and challenges," *Smart Cities*, vol. 2, no. 1, pp. 45-62, 2019.

R. White, T. Black, and S. Green, "Com- parative analysis of embedded systems for digital display applications," *IEEE Embedded Systems Letters*, vol. 11, no. 2, pp. 45-48,2019.

M. Jones and L. Smith, "Accessibility features in public information systems," *Universal Ac- cess in the Information Society*, vol. 18, no. 3, pp. 567-580, 2019.

K. Patel, N. Shah, and R. Kumar, "Perfor- mance evaluation of wireless protocols for IoT applications," *Journal of Network and Computer Applications*, vol. 142, pp. 76-89,2019.

A. Garcia, B. Rodriguez, and C. Lopez, "Cloud storage solutions for multimedia con- tent

delivery," in *Proc. IEEE Int. Conf. on Cloud Computing Technology and Science*, 2018, pp. 123-130.

D. Wilson, E. Taylor, and F. Harris, "JWT authentication mechanisms for secure web services," *IEEE Security & Privacy*, vol. 17, no. 4, pp. 34-42, 2019.

H. Zhang, I. Li, and J. Wu, "Gesture recogni- tion systems using infrared sensors," *Sensors*, vol. 18, no. 11, p. 3721, 2018.

S. Kumar, T. Sharma, and U. Verma, "Rasp- berry Pi-based kiosk mode applications for institutional use," *International Journal of Computer Applications*, vol. 175, no. 15, pp. 1-7, 2020.

M. Brown, N. Davis, and O. Evans, "Mon- goDB and NoSQL databases for IoT data management," in *Proc. IEEE Int. Conf. on Big Data*, 2019, pp. 2345-2352.

P. Miller, Q. Anderson, and R. Scott, "Socket.io for real-time bidirectional event- based communication," in *Proc. Int. Conf. on Web Engineering*, 2018, pp. 156-169.

V. Singh, W. Kumar, and X. Gupta, "Energy efficiency in Raspberry Pi-based IoT devices," *IEEE Access*, vol. 7, pp. 89345-89358, 2019.

Y. Chen, Z. Wang, and A. Liu, "Smart city infrastructure integration: Challenges and so- lutions," *IEEE Transactions on Industrial In- formatics*, vol. 15, no. 6, pp. 3285-3294,2019.

B. Thompson, C. White, and D. Johnson, "Future trends in digital signage technology," *IEEE Pervasive Computing*, vol. 19, no. 1, pp. 74-82, 2020.