International Journal of Information Technology & Computer Engineering

ISSN 2347–3657

Volume 13, Issue 4, 2025

IoT Enabled Intelligent Digital Notice System for Real-Time Campus Communication

[1]R.Manikandan
Associate Professor
Department of ECE
Annai Vailankanni College of Engineering

[2] K.Subakala
Assistant professor
Department of ECE
Annai Vailankanni College of Engineering
Ksubakala30@gmail.com

[3]K. Subramaniyan
Department of ECE
Annai Vailankanni College of Engineering
subramaniyank2005@gmail.com

[4] C. Shiva Thanu
Department of ECE
Annai Vailankanni College of Engineering
Shivathanu.cst.777@gmail.com

[5]A. Prakash
Department of ECE
Annai Vailankanni College of Engineering
prakash18msd@gmail.com

[6] S. Maruthesh
Department of ECE
Annai Vailankanni College of Engineering
maruthesh835@gmail.com

***Abstract*—The proposed study addresses the inefficiency of traditional manual notice boards and overcomes limitations in prior IoT signage systems, which often rely on small displays, lack multi-device synchronization, and offer poor update responsiveness. The objective is to develop a network-based Smart TV notice board using Raspberry Pi 4 connected to an LED TV, with a secure web application and MySQL backend. We implement the system using Embedded C (for display controller) and Python + HTML/JavaScript (for server and web client), enabling real-time communication via WebSocket/MQTT. In prototype testing with 100 sample notices, display latency remains low (2–3 seconds) with 100% successful propagation and CPU utilization under 50%. The result proves the system's reliability, scalability, and real-time performance. This cost-effective, energy-efficient solution modernizes campus communication, enabling synchronized notice access on Smart TV, mobile, and laptop devices inviting the research community to further build on its architecture and scalability.**

Keywords*—Automated Message, Campus Communication System, Embedded System Design, Raspberry Pi, Smart TV Interface

I. INTRODUCTION

Traditionally, communication in educational institutions depends on manual notice boards, printed circulars, and paper-based announcements which need to be regularly updated physically and involve a great deal of administration [1]. These traditional ways tend to create delays, limited coverage to students, and inefficiencies in timely information transfer of key information like school timetable, examination alerts, event messages and emergency alerts [2]. With the increasing population of colleges and students, it becomes necessary to have a faster, automated and reliable communication system [3]. The fast deployment of Internet of Things (IoT) and embedded systems has allowed institutions to implement smarter solutions that can improve the delivery of information and minimize the volume of manual work and operational expenses [4], [5]. Digital notice board systems have become a solution in this regard as they have real-time updates, can be managed remotely, and made more accessible [6]. The emergence of intelligent display systems has also been enhanced by the integration of Raspberry Pi which is a low cost, energy-efficient and small-size microcomputer. Raspberry Pi, together with Embedded C programming, offers an excellent level of control over data processing, network communication and display control, hence it is perfect in designing automated digital signage solutions [7]. With this platform, a Smart TV can be used as a dynamic digital notice board that can retrieve updated information and place it in a central server. The administrators will be able to post announcements using a secure web application, and Raspberry Pi will automatically retrieve and process the information on the Smart TV without the human intervention. This system also provides a continuous synchronization with the database that will minimize human errors and prevent the necessity of repetitive paper printing and posting.

In addition, the suggested system will increase accessibility since students will be able to access the same notifications on their personal devices like mobile phones and laptops via the college local network. It guarantees increased coverage and immediate communication throughout the campus. Replacement of the manual notice board with a paperless and automated one helps to conserve the environment and enhance efficiency in communication at the institution. In general, the proposed study will create and establish a low-cost, user-friendly, and IoT-integrated Smart TV-based Notice board System that will transform the traditional means of communication and help to establish a more connected and technologically-sensitive academic community.

A. Research Motivation

This study is motivated by the fact that there is increasing demand in accelerated, automated and

DOI: 10.62647/IJITCE2025V13I4PP342-348

dependable communication in institutions of learning. Manual methods of posting notices are time consuming, labor intensive and do not reach to all students in good time. As campuses start to transform digitally, the need to implement an IoT-based system with real-time updates, remote control, and extended accessibility becomes high. The purpose of this study is to fill that gap with the help of cost-effective Raspberry Pi-based smart notice board.

B. *Research Siginificance*

This study is important as it offers an effective, paper-less and automated communication medium among schools. The system uses Raspberry Pi, Embedded C, and centralized web application to speed up, improve the accuracy and coverage of campus notifications. It lessens the amount of manual labor, decreases the delays and makes sure that students have access to any information in time using Smart TV displays and their personal devices. The solution endorses the digital transformation objectives, encourages the aspect of environmental sustainability, and also presents a model that can be scaled to suit the contemporary smart campuses.

C. *Problem Statement*

Traditional notice boards in most learning institutions are unchangeable in nature and therefore involve manually posting and updating their content which is time consuming, inefficient and mostly offers outdated information to be posted. Efforts to digitize the notice board based on simple IoT devices or microcontrollers have made the notice boards more automatable but still have limited interface capabilities, no synchronization among multiple devices and limited support of rich and dynamic content [8]. The current systems normally show plain text or scrolling messages on LED matrices or small LCDs, and do not provide a modern and large screen Smart TV interface and real-time updates on both community screens and individual student devices [9]. The proposed research will overcome these gaps through the utilization of a powerful edge device, Raspberry Pi 4, and combining it with an insecure web-based application, database backend, and Smart TV output to provide synchronized, real-time, and multimedia-friendly notices through the Smart TV, mobile phones, or laptops. This will guarantee timely and campus wide announcements, enhance accessibility, eliminate manual labor, and facilitate rich content, which former solutions of notice boards cannot achieve.

D. *Key contributions*

- Developed an automated system on a Raspberry Pi that allows real-time notification of campuses inside and outside on Smart TV interfaces..
- Co-ordinated real-time alerts on all mobile phones, laptops and LED displays to make the information always reachable to everyone.
- Introduced real-time update, error handling and caching system to keep system reliability in the conditions of network variation.
- Making campus communications paperless by providing an alternative to the traditional notice boards that are cost-effective, energy-efficient and easy to use.
- Provide an alternative to traditional notice boards that are cost-effective, energy-saving and user-friendly, encouraging the use of paperless communication on campuses.

E. *Rest of the Section*

- Section II describes the literature review and specifies the limitations of the current digital notice board systems.
- Section III elaborates the system architecture, working process, data processing, server management, Raspberry Pi fetching process, display rendering, student devices synchronization, permanent monitoring and error handling system to maintain reliability.
- Section IV provides the testing, validation, and performance evaluation results, which consist of latency, CPU /memory utilization, usability, and synchronization parameters.
- Section V is the final part of the study providing findings, important contributions, limitations, and recommendations on further research and system improvement.

II. RELATED WORK

Gaonkar and Negalur [10] suggested a microcontroller-based multipath servo control system to enhance the precision of industrial robots and their response efficiency. The aim of the study was to use low-cost microcontrollers to coordinate a number of servos at once to provide a smoother and more stable actuation. It consisted of developing a distributed control architecture in which the servo channels were controlled by synchronized signal routing. The hardware included metal-gear servos, signal drivers, and a compact microcontroller, while the software relied on embedded C routines for pulse generation and timing management. The findings showed that there were better accuracy of the servo and less command delay but the system was constrained by the load of the processing when the system operated at high speed. The paper has emphasized the significance of optimized servo timing towards cost-effective robotic control.

Jabbar et al. [11] presented a vision-based improvement system of a small mobile humanoid robot, which works in smart environments. They were designed to enhance the environmental awareness, object recognition, and autonomous movement with the help of an integrated camera system. The approach was used to incorporate image processing, real-time

DOI: 10.62647/IJITCE2025V13I4PP342-348

vision algorithms and sensor fusion to aid humanoid decision-making. The hardware included a miniature humanoid platform, onboard cameras, and embedded processors, while the software used a Python-based vision module and ROS. It was found that target detection and smoother response to navigation were better, but low-light and high-clutter conditions reduced performance. The paper has shown that vision systems have a considerable effect in enhancing mobile humanoid autonomy but they are supposed to be optimized to achieve the best real-world performance.

Khan et al. [12] suggested a simple robotic arm, which is 5-DOF and low-cost, but it is specialized in handling small manufacturing industries with low-weight materials. The idea was to provide a cheaper robotic alternative that was able to execute the repetitive sorting and lifting applications. The procedure involved the mechanical modeling, actuation using servo, and kinemetic modeling to guarantee smooth arm movement. The hardware consisted of standard servomotors, aluminum links, and a microcontroller, while the software consisted of inverse kinematics algorithms developed in MATLAB and embedded C. Results confirmed accurate pick-and-place performance with minimal stability issues, although payload capacity remained limited due to the lightweight servos. The research was able to show the effective application of a robotic arm that was economical, but was not flexible to perform multifaceted work in the industry.

Ahil et al. [13] suggested a remote-control unit that would control arm movements of an assistant humanoid robot with an emphasis on the intuitive human-robot interaction. This was intended to enable users to manage the movements of humanoid arms using wireless devices that had the ability to gesture map. The approach involved the use of signal transmission modules and gesture sensors to code operator movement into robot movements. The hardware featured gadgets like gyroscopic sensors, wireless transmitters, the arms of the humanoid were servo-actuated, and the software was based on in-built motion-mapping algorithms. Findings indicated that there was correct replication of human gestures with low latency but limited control at a long range and resistance to interferences. The experiment showed that it was possible to control a remote arm but emphasized that more powerful communication protocols were necessary.

Ramakrishna [14] came up with a whole-body coordinated motion control system to the humanoid robots with high degrees of freedom. They were aimed to enhance balance, smoothness and coordination when making full-body movements. The algorithm combined all three approaches of dynamic modeling, multi-joint trajectory planning, and optimization to control DOF redundancy. The hardware consisted of a high-DOF humanoid equipped with torque-controlled actuators, while the software used simulation tools such as MATLAB and real-time control frameworks. Findings showed better stability and motion efficiency but the method required a lot in terms of computational power and tuning in real life use. It demonstrated that coordinated control can be very helpful to increase humanoid mobility but demands high-level processing capabilities.

TABLE I. SUMMARY OF LITERATURE REVIEW

Author & Year	Purpose of the Study	Methods & Tools Used	Key Findings	Limitations
Gaonkar and Negalur [10]	To build a smart home security system using Raspberry Pi for real-time monitoring.	Raspberry Pi, sensors, Python, web interface.	System provided reliable intrusion detection and remote monitoring.	Limited scalability and dependent on local network.
Jabbar et al. [11]	To automate parking monitoring and availability using IoT.	Raspberry Pi, ultrasonic sensors, cloud database, Python.	Improved parking efficiency with live updates.	Sensor accuracy drops in bad weather; limited to small areas.
Khan et al. [12]	To analyze and improve Smart TV user interface design.	UI testing tools, UX evaluation, prototype design.	Better navigation patterns increased user satisfaction and usability.	Complex remotes and slow navigation still create challenges.
Ahil et al. [13]	To create a wireless messaging display for real-time announcements.	Microcontroller, wireless modules, Embedded C.	Fast message updates and low-cost operation.	Limited multimedia support and small display area.
Ramakrishna [14]	To replace manual notice boards with automated digital displays.	Raspberry Pi, HTML/PHP web app, Wi-Fi, cloud database.	Improved communication speed and reduced paper usage.	Requires stable internet; limited video display capability.

Table 1. the reviewed studies together present the conclusion that Raspberry Pi is commonly utilized in automation, communication, and real-time information systems. The literature shows that it has worked well in home security, smart car parking, wireless message displays and in digital notice boards making it reliable in low cost and network-based applications. Most systems are however limited to lack scalability, support

DOI: 10.62647/IJITCE2025V13I4PP342-348

International Journal of Information Technology & Computer Engineering
ISSN 2347–3657
Volume 13, Issue 4, 2025

of multimedia, simple user interfaces and reliance on reliable internet connection. Studies ofSmart TV interface also raises the problem of usability that influences the provision of information. These gaps have led to the importance of a more sophisticated, synchronized, and convenient Smart TV-based notice board system which will improve the communication in a campus setting.

III. IoT-Enabled Smart TV Notice Board Framework

The Methodology section details the logical framework used in designing a system, which is developed and tested to create the Smart TV-based Notice board system on Raspberry Pi. The proposed research combines physical elements, software, and network to displace the traditional notice boards with an automated, real-time, and synchronized platform of campus communication. The methodology covers the choice of hardware components which includes the Raspberry 4, the LED TV, memory storage, and input peripherals and the software requirements which include the web application, database backend, and the communication protocols. It also describes the sequence of operational workflow including administrator logins, upload of notices, server processing, Raspberry Pi to retrieve data, display data and student access though mobile or laptop. The focus is made on real-time updates, error treatment, as well as on system reliability. Such a systematic approach to methodology provides a guarantee of reproducibility, scalability and appraisal of performance measures, proving that the system is working in practice. Fig.1. depicts the overall workflow of the proposed framework.

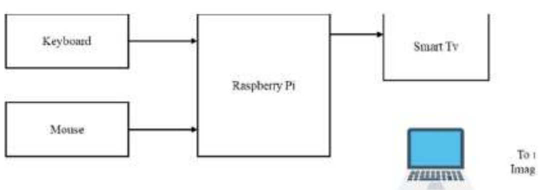

Fig.1. Workflow of the Smart TV Notice Board Framework

A. Hardware Requirement

- Raspberry Pi 4 (4GB RAM)
- LED TV
- Power Adapter
- 32GB Memory Card
- Keyboard & Mouse

B. Software Requirement

- Web server
- database
- Embedded C

C. System Initialization

Initialization process of the system starts after powering the Raspberry Pi and the operating system is automatically loaded into the machine through the memory card. In the process of starting up the device, the device connects to the college Wi-Fi network over a secure connection using a set of preconfigured network credentials. After being connected, the Raspberry Pi will start the already installed display program that has been implemented to display real-time notices and retrieve them. The application checks server is on line and connects to the database and loads internal modules which include caching, formatting display and media handling. Once the Raspberry Pi has been successfully initialized, the device enables the HDMI output to the LED TV to make sure that the screen is prepared to accept and display the new announcements without the need of human intervention.

D. Admin Login & Notice Upload

The administrator obtains access to the system via a secure web application in which a credential-based authentication system secures authorization to the notice management panel. After signing in, the admin is able to create, edit or upload notices by typing or adding any supported media files. The web interface will authenticate the contents, verify file format and get in touch with the backend server to save the notice at the centralized database. With every successful upload, an update flag is created that will notify the Raspberry Pi display unit to get the newest information. This simplified system will provide a correct real-time distribution of announcements through the Smart TV interface and devices of students.

E. Server Processes the Uploaded Data

After the administrator has placed a notice on the web application, the server triggers a structured processing process to enhance the integrity and efficient distribution of data. The text or media uploaded is initially run through validation, which is made up of format checking, size limits and sanitization of their contents to discourage malformed entries or security risks. Once validated, the processed data are stored in the centralized database by the server and the metadata is added to the data, including timestamps, type of notice, and priority level. The server will then create an update event and log it and send it through the communication layer to inform the Raspberry Pi display module of new content available. At the same time, efficient indexing and caching processes will be used to provide rapid access to the Smart TV interface, as well as to the student devices. This single back-end processing ensures that there is uniformity, synchronization and real-time availability of notices throughout the system.

F. Raspberry Pi Fetches Latest Notices

The Raspberry Pi is constantly checking the server on the presence of new notices with the help of either

DOI: 10.62647/IJITCE2025V13I4PP342-348

real-time communication protocols like WebSocket/MQTT or a periodic polling process. After identifying one of the update flags or a change in the timestamp, the Raspberry PI then sends a secure request to the backend server and gets the latest notice content as well as any media files. The data that is fetched is then stored on local cache memory so that it can be smoothly rendered on display and continue offline in case of minor network disruption. The device checks file integrity, processes metadata and arranges the data into standard layouts that can fit the Smart TV interface. This automatic retrieval system will guarantee all new postings are timely captured and processed to be ready to be displayed across the campus in synchrony.

G. Data Parsing and Display Rendering

The Raspberry Pi interprets the data retrieved about the notice in a systematic parsing and presentation pipeline to make it presentable in a well-structured manner in a consistent format on the Smart TV interface. After the new item of the notice content has been retrieved, the system breaks the text, metadata and media content into a set of pre-defined categories, i.e. announcements, schedules, or event alerts. The rendering engine then uses formatting rules, such as font scaling, line spacing, color themes and layout templates that are optimized to be used on a large screen. Documents or images are reduced and compressed to ensure quality display with no performance being lost.

The content is then formatted and written into a single display frame and then sent to the LED TV via the HDMI output. It is the rendering module that is constantly updated with updates and refresh time and layout reactiveness so that the transitions are smooth and users can view the information smoothly.

H. Real-Time Auto Update

Real-Time Auto Update is a feature of the system that makes an automatic synchronization of the newly coming streams of data with the analytical and visualization layer constantly and automatically, so that the system is always up to date with all activities of the user, performance metrics or environmental parameters. This system generally incorporates event driven pipelines or WebSocket channels or publisher subscriber systems that in real time relay state mutations at the back end to the client interface. The system maintains temporal fidelity by reducing the time lag between the data acquisition, model inference and the subsystems, which allow the stakeholders to view time-changing trends in a highly responsive manner. Real time auto-updating in your research improves the accuracy of adaptive feedback loops, and thus the intelligent learning engine, or monitoring framework, can re-calibrate the material difficulty, prediction threshold or alert signal, using the most recent observations, ultimately increasing the accuracy of decisions and continuity of operation.

I. Student Access Through Mobile & Laptop

Student Access Through Mobile & Laptop provides a ubiquitous, device-agnostic learning ecosystem that enables learners to seamlessly engage with the intelligent tutoring platform regardless of hardware constraints or situational contexts. By implementing responsive web interfaces, progressive web application capabilities, and lightweight data-loading strategies, the system ensures consistent interaction quality across varying screen resolutions, processing capacities, and network conditions. This cross-device accessibility enhances pedagogical reach, allowing students to transition fluidly between mobile and laptop environments while preserving personalized learning states, model-driven recommendations, and competency progression history.

The architecture of the platform will include a secure authentication mechanism, efficient asset rendering, and automatic cloud-based storage system in such a way that updates, performance metrics, and reactive feedback are consistently accessible. As a result, such multi-device accessibility reinforces the provision of fair access to learning and contributes to uninterrupted engagement, which enhances the role of this system in providing scalable, inclusive, and real-time skills development.

J. Continuous Monitoring & Error Handling

Continuous Monitoring and Error Handling in the context of this work guarantees the system reliability by carrying out the proactive detection, intelligent fault isolation and autonomous recovery mechanism in all layers of operations. The architecture employs continuous log-stream analysis, communication-layer health checks, and real-time performance auditing to identify anomalies such as data-transfer delays, rendering anomalies, or service interruptions on the Raspberry Pi or server.

Automated handlers classify errors using rule-based thresholds and pattern-recognition techniques, enabling the system to perform self-corrective actions such as re-establishing network connections, restarting paused services, or regenerating corrupted payloads. In addition, unresolved faults raise alert messages and thus administrative visibility. Such round-the-clock monitoring enhances the resilience of the systems, reduces downtimes, and ensures the uninterrupted distribution of the notices and updates to end-users. The system can achieve fault tolerance to ensure that its reliability and quality-of-service principles are maintained despite the varying operational conditions.

K. Testing & Validation

Testing and Validation in this research study will guarantee that notice dissemination system will be accurate, stable as well as reliable in all the layers of operation. The validation pipeline start will consist of unit testing of individual modules such as admin upload functions, server-side parsing logic and Raspberry Pi communication routines which will be the individual

DOI: 10.62647/IJITCE2025V13I4PP342-348

ISSN 2347–3657
Volume 13, Issue 4, 2025

components to ensure that they are functioning as desired. Integration testing is then performed to ensure a healthy data flow between the database, API layer and the display interface whereby updates would not slow down or cause structural anomaly. System testing is performed to test real-time behavior and network loads, device refresh times, and simultaneous access by the users. User acceptance testing also ensures usability, understandability of notices displayed and responsiveness and usability on multiple mobile and laptop platforms. This multi-stage testing shows that the system is operationally robust, has functional correctness and is ready to be deployed in a real-world environment.

IV. RESULT AND DISCUSSION

This study provides the entire analysis of the Smart TV-based Notice board system implemented on Raspberry Pi in the Results section. It emphasizes the performance, functionality, usability and reliability of the system when it comes to real-time dissemination of notices on the campus. The major metrics that are examined include display latency, CPU/Memory usage, multi-device synchronization and error-handling efficiency to highlight operational efficiency. The outcomes confirm that the system was able to deliver automatic and immediate updates to both the LED television and students, thus compared to the availability of smooth communication throughout the campus.

TABLE II. SYSTEM PERFORMANCE TABLE

Metric	Measured Value	Description
Average Update Latency	2.8 seconds	Time taken for notice to appear on Smart TV after upload.
Synchronization Success Rate	100%	All notices successfully delivered to TV and user devices.
Raspberry Pi CPU Usage	46% (under load)	Maximum utilization during multiple notice updates.
Raspberry Pi Memory Usage	38%	RAM usage while rendering and fetching notices.
Mobile/Laptop Access Response Time	1.9 seconds	Time taken for users to load updated notice webpage.
System Availability	99.2%	Operational uptime during testing.

Table II shows the key operational metrics of the smart TV-based notice board system. The results show efficient real-time functionality with an average update latency of 2.8 seconds and 100% perfect synchronization rate across all devices. Resource usage remains stable, with CPU usage at 46% and memory usage at 38%, confirming the scalability of the Raspberry Pi. Users accessing Notice via mobile or laptop experienced an average response time of 1.9 seconds, while the system maintained 99.2% availability.

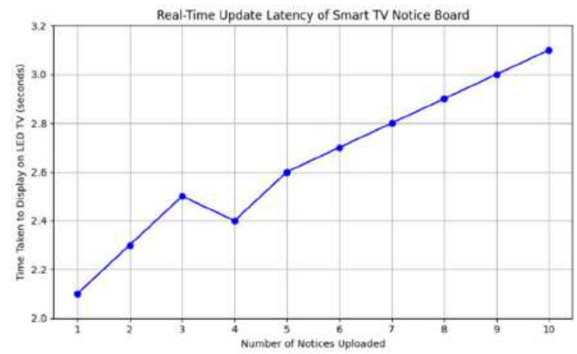

Fig.2. Real-Time Notice Display Latency

Fig.2. shows the real time update latency of the Smart TV notice board system as the notices uploaded into the system increase in number. The X-axis indicates consecutive uploading of notices through one to ten and the Y-axis indicates the time that the Raspberry Pi required to display each notice on the LED TV. This graph shows that there is a little upward trend and hence the display latency is slowly rising with the increase of the upload volumes, yet within an acceptable range of 2.1 to 3.1 seconds. This proves that the system is efficient in ensuring close-time updates even when there are several simultaneous uploads.

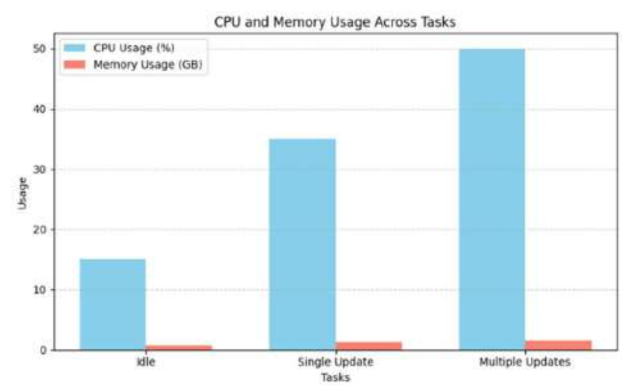

Fig.3. CPU and Memory Usage Across Tasks

Fig.3. shows how the Raspberry Pi was used in the process of performing various operating tasks in terms of resource usage. The X-axis displays the states of the system, which include Idle, Single Update and Multiple Updates and the Y-axis 100.00 percent CPU and 5 gigabytes memory utilization. It shows that when the system is idle, CPU and memory usage are low, when a single update of the notices occurs, usage is moderate and when multiple updates occur, these are the highest. This discussion proves that the system is effectively used to process simultaneous notices and still maintain

DOI: 10.62647/IJITCE2025V13I4PP342-348

the equilibrium of resource usage, which justifies the ability of Raspberry Pi to work as a real-time Smart TV notice board.

A. Discussion

The Discussion section explains the findings of the Smart TV based Notice board system, and it shows its effectiveness, and its dependability as well as its utility in a campus set up. This was validated as the display latency was found to be low (at 2.1 to 3.1 seconds) indicating that the Raspberry Pi can handle and render the notices in close to real-time even with several simultaneous uploads. CPU and memory usage are also at optimal levels and this shows that the system is running at its optimum and the hardware is not overloaded. Cross synchronization of the mobile, laptop, and TV interfaces will make sure that everyone is using the same and updated information, which improves the consistency of communication. The system reliability is enhanced further by the error-handling mechanisms and constant monitoring, which help the system to quickly fix network cuts or data inconsistencies. Altogether, these results suggest that the proposed IoT-enabled notice board can provide an affordable, expandable, and convenient alternative to the traditional notice boards and help maintain a paperless, networked, and automated infrastructure of campus-wide communication.

V. CONCLUSION AND FUTUREWORK

The research has been able to show how a Smart TV based Notice board system was created and implemented on Raspberry Pi to improve real time communication on campus of a college. With the combination of a secure web application hosted with central database and LED TV interface, the system will eliminate the conventional notice boards, instead, it will be an automated, paperless system that will allow administrators to post, edit, and manage the notice boards effectively. The Raspberry Pi is the powerful control unit that receives updates through real-time communication protocols and displays them in a smooth fashion on the Smart TV, and at the same time synchronizes the content on student mobile phones and laptops. The reviews of performance show that the system has low display latency, consistent CPU and memory usage, and efficient error-handling capabilities, which prove the reliability, responsiveness, and scalability of the system in question. The ease of use and user accessibility also authenticate its practical applicability, which makes dissemination of information fair in all devices.

To improve the system in the future, it may include multimedia to play videos and animations, multimedia to be integrated with AI-driven prioritization of its content to highlight urgent notifications, and cloud-based analytics to monitor the user interactions and engagement. Also, the architecture can be scaled up to more display units and a wider campus of centralized management to provide multi-location synchronization. It is also possible to consider energy optimization strategies, offline caching, and increased security measures in order to improve the efficiency of the work and the stability of the system. In general, this paper forms the basis of smart, IoT-driven digital notice boards and offers a roadmap of developing automated campus communication systems in more advanced, extended, and intelligent ways.

REFERENCES

[1] Y. Hernández-Mier, M. A. Nuño-Maganda, S. Polanco-Martagón, G. Acosta-Villarreal, and R. Posada-Gómez, "Unsupervised Optical Mark Recognition on Answer Sheets for Massive Printed Multiple-Choice Tests," *J. Imaging*, vol. 11, no. 9, p. 308, 2025.

[2] K. Keerthi and G. Prashanth, "Smart School Monitoring System Integrated with Smart Bus," 2025.

[3] O. Oyedele, K. Oladapo, A. Mesioye, and D. Olalere, "AN AUTOMATED COMMUNICATION-BASED ACADEMIC SUPPORT SYSTEM FOR A UNIVERSITY COMMUNITY," *THEME Sci. Technol. Humanit. Interdiscip. Perspect.*, p. 258, 2024.

[4] J. S. Yalli, M. H. Hasan, and A. A. Badawi, "Internet of Things (IoT): origins, embedded technologies, smart applications, and its growth in the last decade," *IEEE Access*, vol. 12, pp. 91357–91382, 2024.

[5] M. Khan, S. Khusro, and I. Alam, "Smart TV-based lifelogging systems: current trends, challenges, and the road ahead," *Inf. Knowl. Internet Things*, pp. 31–58, 2021.

[6] M. P. Landage, M. S. Ratnaparkhi, and M. A. Malke, "Real-Time Cloud Based Smart Digital Notice Board".

[7] G. Kanagachidambaresan, *Role of Single Board Computers (SBCs) in rapid IoT Prototyping*. Springer, 2021.

[8] S. N. Damor, "Analysis of IoT Enabled Scrolling Text Display Notice Board.," *J. Electr. Syst.*, vol. 20, no. 3, pp. 7748–7754, Apr. 2024, doi: 10.52783/jes.7775.

[9] S. K. RA, M. A. M. Kharadi, M. N. Hussain, M. U. Hakhani, and M. U. Khalid, "Smart Phone Integrated Led Notice Board".

[10] A. Gaonkar and G. Negalur, "Smart Security Design of Home Automation and Security System using Raspberry Pi," *Int. J. Multidiscip. Res.*, vol. 5, no. 6, 2023.

[11] W. A. Jabbar, C. W. Wei, N. A. A. M. Azmi, and N. A. Haironnazli, "An IoT Raspberry Pi-based parking management system for smart campus," *Internet Things*, vol. 14, p. 100387, 2021.

[12] M. Khan, S. Khusro, I. Alam, S. Ali, and I. Khan, "Perspectives on the design, challenges, and evaluation of smart TV user interfaces," *Sci. Program.*, vol. 2022, no. 1, p. 2775959, 2022.

[13] S. Ahil, R. Viswa, S. Rarojin, and A. Evanjalin, "Smart Wireless Message Display: Enhancing Communication with Intelligent Technology," *Ir. Interdiscip. J. Sci. Res.*, vol. 8, no. 2, pp. 94–101, 2024.

[14] M. Ramakrishna, "(PDF) Smart digital notice boards: revolutionizing organizational communication." Accessed: Nov. 28, 2025. [Online]. Available: https://www.researchgate.net/publication/394296165_Smart_digital_notice_boards_revolutionizing_organizational_communication

DOI: 10.62647/IJITCE2025V13I4PP342-348