

SMART LEARN HUB: OFFLINE LOCAL WI-FI BASED DIGITAL LEARNING SYSTEM USING RASPBERRY PI

Dr. Anwarul Siddique¹, Ayesha Khan Pathan², Rachana Bankar³, Mahi Bomble⁴, Bushra Hussain⁵ Nishtha Kale⁶

¹²³⁴⁵⁶Department of Computer Science & Engineering
Anjuman College of Engineering & Technology, Nagpur, Maharashtra, India

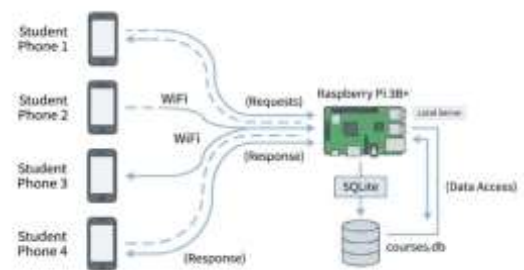
ABSTRACT

Smart Learn Hub is a Raspberry Pi based offline educational system developed for providing easy and efficient access to digital learning resources without internet connectivity. In this system, various educational contents such as videos, notes, PDFs, and quizzes are available under a single platform. The students can connect to the local server through Wi-Fi and access the study materials anytime within the network range. The system stores educational content along with basic assessment features to evaluate student performance, and offline tracking of progress is possible for the users. The best feature is that it does not require internet, reducing dependency and cost. What the user has to do is connect their device to the local Wi-Fi network and access the platform through a browser interface. If they want to study, they can select their subject, view content, and attempt quizzes based on the topic. The system is designed to be simple and user friendly so that even users with basic technical knowledge can operate it easily. The ideology behind such a system is to provide affordable digital education and reduce the gap caused by lack of internet connectivity in rural and underserved areas.

INTRODUCTION

Smart Learn Hub is a Raspberry Pi-based offline educational server — a self-contained, Wi-Fi broadcasting device that eliminates the need for internet infrastructure to deliver quality digital education to rural and semi-urban government schools across India. Unlike traditional e-learning platforms that rely heavily on stable broadband connectivity, cloud-based hosting, and recurring data subscriptions, Smart Learn Hub is a complete physical infrastructure solution. It is compact, portable, and energy-efficient — essentially a miniature server that fits in the palm of

a hand. Schools can simply power it on, and within moments, it creates a fully functional digital classroom environment.



Smart Learn Hub goes beyond being just a content storage device. It intelligently manages the entire learning ecosystem offline. It delivers structured educational content such as videos, interactive lessons, PDFs, and assessments directly to connected users. Students can engage with quizzes, assignments, and self-paced modules, while the system automatically evaluates performance in real time. Although many of these schools are equipped with digital infrastructure such as interactive e-boards, desktop computers, and PCs through government initiatives, the lack of connectivity leaves these resources underutilized and restricts them to minimal offline use.

LITERATURE REVIEW

Smart Learn Hub can be preloaded with curriculum-aligned educational content, including state board syllabi, materials and skill-based learning modules. It also supports periodic updates via USB or offline sync methods, ensuring that content remains relevant and up to date without requiring live internet access.

Table 1. Summary of Related Literature

SR. NO	NAME OF AUTHOR	YEAR OF PUBLICATION	TITLE OF PAPER
1.	PRITPAL SINGH	DECEMBER 2021	Provisioning of Educational Content to Rural Schools in PNG using RACHEL
2.	HIRAK GHAEEL	JANUARY 2021	RASPBERRY PIE
3.	Manjunath H H ECE department	2019	Smart Classroom using Raspberry Pi
4.	Dhanush K	MAY 2024	A Raspberry PI-Based Smart System

A. Drawbacks of Existing Systems

Existing offline educational systems and Raspberry Pi-based solutions face several limitations [1]–[3]. Content updates are difficult, and reliance on a single device increases the risk of system failure. These systems offer limited storage and interactivity compared to online platforms.

They often require basic technical knowledge, which can be challenging for users with low digital literacy, and lack adequate maintenance and curriculum alignment [1]. Additionally, hardware constraints such as microSD-based storage, limited processing power,

overheating, and compatibility issues due to ARM architecture restrict performance and scalability [2], [3].

PROPOSED WORK

A. Problem statement

India has over 1.2 lakh government schools with little to no internet connectivity, which severely limits the adoption of modern digital learning solutions and access to online platforms, cloud resources, and real-time

Educational tools. Although many of these schools are equipped with digital infrastructure such as interactive e-boards, desktop computers, and PCs through government initiatives, the lack of connectivity leaves these resources underutilized and restricts them to minimal offline use. As a result, digital learning remains inaccessible in practice despite the availability of hardware.

B. System Architecture

Offline-First Architecture: Design ensures zero-internet dependency and fully self-contained operation
Modular Development Approach: Phase 1 (Hardware Setup) → Phase 2 (Backend Development) → Phase 3 (Frontend Portal) → Phase 4 (Testing) → Phase 5 (Deployment)

Hardware Validation: Confirmed Raspberry Pi 3B+ as minimum spec; stress-tested for 20+ concurrent user handling.

Iterative Testing Methodology: Local network testing → Progressive load testing → Real world school environment trials (Phase 2)

SDG-Centric Design: All development decisions prioritize accessibility, cost efficiency, and equity alignment

OBJECTIVES

- To configure a Raspberry Pi 3B+ as a local server and Wi-Fi hotspot for offline access.
- To develop a web-based learning portal accessible through standard browsers.
- To store and deliver educational content (videos, notes, quizzes) locally.
- To implement a backend system using Flask/Node.js and SQLite for data management.
- To design a teacher dashboard for

monitoring student performance and activity.

- To evaluate system performance in a real-time environment

METHODOLOGY

A. Raspberry Pi Setup

The development process began with the physical setup of the Raspberry Pi, which serves as the central processing unit of the system. The setup was performed using a monitor, keyboard, and mouse to allow direct interaction with the device during installation and configuration. The Raspberry Pi OS was installed on a microSD card, and the system was booted with a graphical interface to facilitate easier setup and troubleshooting.

Initial system updates and software installations were carried out to prepare the environment for further configuration. This approach ensured better control over system settings and simplified the setup process, especially during early-stage development.

B. Network Configuration

In this phase, the Raspberry Pi was configured as a wireless access point using RaspAP, a web-based network management tool. RaspAP enables the device to broadcast a Wi-Fi network that users can connect to via smartphones laptops or smartboards. All connected users are redirected to a local IP address hosting the learning portal, creating a self-contained offline network. This ensures that all communication and content delivery occur locally without requiring internet connectivity.

C. Frontend Development

Once the network infrastructure was established, the development of the user interface was undertaken. The frontend was designed using HTML, CSS, and JavaScript, with a focus on simplicity and accessibility. The interface allows users to navigate through subjects, access or upload study materials, and attempt or upload quizzes in an intuitive manner.

D. Backend Development

The backend of the system was developed using the Flask framework in Python. It is responsible for handling user requests, managing content delivery, and processing quiz interactions. The backend communicates with an SQLite database, which stores user activity data and quiz results locally.

Since the system operates within a closed network, the backend ensures fast response times and maintains data privacy. The lightweight nature of Flask makes it well-suited for deployment on resource-constrained hardware such as the Raspberry Pi.

E. Integration and Deployment

In the final stage, all system components were integrated, linking the frontend, backend, and network configuration to ensure seamless operation.

Automation was implemented to enable the system to start essential services on boot, allowing it to function as a plug-and-play solution that automatically broadcasts the Wi-Fi network and hosts the learning portal.

The system was deployed in a controlled environment and tested with multiple users, demonstrating stable performance and effective content delivery.

IMPLEMENTATION

A. System Architecture Implementation

The system was implemented using a layered architecture comprising hardware, server, and client layers. The Raspberry Pi acts as the central unit, functioning as both a local server and wireless access point. A 16 GB microSD card serves as the primary storage, hosting the operating system, application, database, and educational content, making the system fully self-contained.

B. Network and Connectivity

The networking functionality was implemented using RaspAP, which enables the Raspberry Pi to operate as a wireless access point. The system broadcasts a dedicated Wi-Fi network, allowing multiple users to connect simultaneously. All user requests are routed to a local IP address hosting the web application, ensuring that all interactions occur within the local network. This approach guarantees uninterrupted access to educational content without requiring internet connectivity.

C. Frontend Implementation

The frontend was developed using HTML, CSS, and JavaScript, providing a simple and responsive interface accessible through any web browser. It enables users to navigate content, view materials, and attempt quizzes with real-time feedback.

The design is optimized for low-end devices, ensuring accessibility in resource-constrained environments.

D. Backend Implementation

The backend was developed using Python with the Flask framework, enabling efficient routing, request handling, and content delivery. Educational content and user interactions, including quiz submissions, are processed dynamically through defined endpoints. An SQLite database is used for local storage of quiz data and user activity, ensuring fast access and maintaining data privacy without external transmission.

E. Content Management and Delivery

Educational content was preloaded onto the system in the form of compressed video files, PDF notes, and structured quiz data. The backend serves this content efficiently based on user requests.

The content is organized in a hierarchical structure, allowing users to easily navigate subjects, topics, and learning materials. This structured approach improves usability and ensures systematic learning.

F. System Operation Workflow

The operational flow begins when the Raspberry Pi is powered on, initiating the network and server services automatically. Users connect to the broadcasted Wi-Fi network and access the learning portal through a browser. Once connected, users can browse content, view learning materials, and attempt quizzes. The backend processes all interactions and stores relevant data locally. The system operates entirely within the local network, ensuring fast response times and consistent performance.

G. Performance Considerations

The implementation was optimized to handle multiple concurrent users within the limitations of the Raspberry Pi hardware. Lightweight frameworks, efficient data storage, and minimal external dependencies were used to ensure smooth performance.

Testing confirmed that the system can support classroom-level usage with stable operation and acceptable response times.

SAMPLE OUTPUT:



Fig 1 : Connecting to SmartLearn_Box

Users can connect to this network without internet access, enabling entry into the offline learning environment.

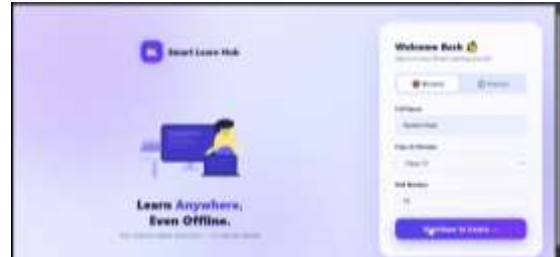


Fig 2:

Login Page

The login interface allows users to enter basic details such as name, class, and roll number. The system supports both student and teacher modes for role-based access.



Fig 3:

Student Dashboard

The dashboard provides an overview of user activity, including completed chapters, quizzes taken, and performance metrics. It also displays available subjects for learning.



Fig 4:

The teacher dashboard provides an overview of student performance, including progress tracking, quiz results, and student details for effective monitoring.

APPLICATIONS

- It can be used in rural and remote schools & colleges.
- Works without continuous internet connectivity.
- Reduces digital divide between urban and rural students.
- Portable and can be used in multiple locations

- It supports multiple users at the same time.
- It can store large amount of educational content locally
- Customizable for different subjects and syllabus.
- Useful for NGOs and government education programs.
- Promotes digital literacy in offline environments.
- Scalable system for a large number of schools.

CONCLUSION

The Smart Learn Hub demonstrates that quality digital education can be effectively delivered without relying on internet connectivity. By utilizing a low-cost device such as the Raspberry Pi, the system provides a practical and affordable solution for schools in rural and underdeveloped areas. It enables access to structured learning content, including videos, notes, and quizzes, allowing students to learn anytime and within the local network environment.

The system successfully addresses the challenge of limited connectivity by creating a self-contained offline learning platform. It helps bridge the digital divide by making educational resources accessible to students who otherwise lack internet access. Additionally, the platform is designed to be user-friendly, ensuring that teachers can operate it easily without requiring advanced technical skills.

The Smart Learn Hub also supports scalable deployment and can be implemented in real classroom environments with minimal infrastructure requirements. Overall, the system proves to be an effective, reliable, and sustainable solution for improving digital education accessibility and enhancing student learning outcomes in connectivity-constrained regions.

FUTURE SCOPE

1. Enhanced Hardware Performance:

Future improvements can include upgrading from Raspberry Pi 3 Model B+ to higher models (like Pi 4/5) for better speed, storage handling, and support for more concurrent users.

2. Integration with Internet (Hybrid Mode):

The system can be extended to support both offline and online modes, allowing periodic content

updates and access to cloud resources when internet becomes available.

3. Mobile Application Development:

A dedicated mobile app can be developed to provide a more optimized and user-friendly experience compared to browser-based access.

4. Advanced Analytics Dashboard:

Future versions can include detailed analytics for teachers, such as student progress tracking, performance reports, and learning behavior analysis.

5. AI-Based Personalized Learning:

Integration of Artificial Intelligence can help in recommending personalized content based on student performance and learning patterns.

6. Content Expansion and Localization:

The system can be enhanced by adding multilingual support and aligning content with different state board curricula across India. centralized monitoring.

7. Improved Security Features:

Future versions can include user authentication, encrypted data transfer, and role-based access control for better security.

8. Scalability for Large Institutions:

The system can be expanded using multiple interconnected devices (cluster setup) to support larger schools or colleges.

9. Integration with Smart Classroom Tools:

The system can be integrated with smart boards, IoT devices, and other classroom technologies to enhance interactive learning.

ACKNOWLEDGMENT

The authors express profound gratitude to their guide, Dr. Anwarul Siddique for his valuable guidance and encouragement throughout this project. They also thank Prof. Nazish Khan (Head of Department) and Prof. Dr. K.S. Zakiuddin (Principal), Anjuman College of Engineering & Technology, Nagpur, for providing necessary facilities. Thanks, are also extended to all faculty, staff, and family members who contributed directly or indirectly to the successful completion of this work.

REFERENCES

- [1] P. Singh, "Provisioning of Educational Content to Rural Schools in PNG using RACHEL," 2019. [Online]. Available: https://www.researchgate.net/publication/335421122_Provisioning_of_Educational_Content_to_Rural_Schools_in_PNG_using_RACHEL
- [2] H. Ghael, "A Review Paper on Raspberry Pi and its Applications," 2021. [Online]. Available: https://www.researchgate.net/publication/348993428_A_Review_Paper_on_Raspberry_Pi_and_its_Applications
- [3] H. Manjunath, "Smart Classroom using Raspberry Pi," *International Journal of Engineering Research & Technology*, 2019. [Online]. Available: <https://www.ijert.org/smart-classroom-using-raspberry-pi>
- [4] "RaspAP Project," *International Journal of Future Management Research*, 2024. [Online]. Available: <https://www.ijfmr.com/research-paper.php?id=20248>
- [5] Raspberry Pi Foundation, "Raspberry Pi Official Website." [Online]. Available: <https://www.raspberrypi.org>
- [6] Pallets Projects, "Flask Documentation." [Online]. Available: <https://flask.palletsprojects.com>
- [7] SQLite Consortium, "SQLite Documentation." [Online]. Available: <https://www.sqlite.org>
- [8] RaspAP, "RaspAP Project." [Online]. Available: <https://raspap.com>
- [9] Khan Academy, "Khan Academy." [Online]. Available: <https://www.khanacademy.org>
- [10] National Programme on Technology Enhanced Learning (NPTEL), "NPTEL." [Online]. Available: <https://nptel.ac.in>