International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 10 Issue: 10 | Oct - 2026 SJIF Rating: 8.659 ISSN: 2582-3930

Design of an AIOT Based Personalized Cognitive and Memory Enhancement System for Children with Special Needs

Priyanka Ashok Swami[1], Prof. Swati Deshmukh[2], Dr. Ashwini Bade[3], Dr. Prabhat Pallav[4]

1 PG Student, Department of Electronics and Telecommunication Engineering
2 PG Coordinator, Department of Electronics and Telecommunication Engineering
3 Assistant Professor, Department of Electronics and Telecommunication Engineering
4 Head of Department, Department of Electronics and Telecommunication Engineering
Siddhant College of Engineering, Sudhambare, Talegaon - Chakan Highway, Sudumbre, Maharashtra 410501, India

***

Abstract - This paper presents the design of a low-cost Artificial Intelligence of Things platform for personalized cognitive and memory training for children with special needs. The system separates deterministic physical interaction from multimedia presentation and adaptive analytics. An ESP32-S3 console provides four large colored inputs, a Start or Hint control, visual feedback and audio cues, while a Python application manages visual, working, spatial and associative memory activities. Trial accuracy, response latency, sequence span, attempts, hints and delayed recall are stored locally. A Random Forest model estimates success probability for candidate activities, and transparent guardrails constrain recommendations when learner history is limited. The proposed architecture supports reproducible evaluation, educator oversight and privacy-preserving local operation. It is intended for assistive training and progress monitoring rather than clinical diagnosis.

Key Words: AIoT, cognitive training, memory assessment, ESP32-S3, adaptive learning, Random Forest, spaced repetition

1. INTRODUCTION

Children with special needs may benefit from training activities that use clear prompts, repeatable interaction and immediate feedback. Conventional worksheets and fixed digital games often provide limited control over difficulty and produce little objective evidence about response timing, error patterns or retention. An adaptive platform can address these limitations by recording trial-level behavior and selecting subsequent activities from the learner's recent performance.

The proposed system combines an embedded interaction console with an edge computer that presents multimedia tasks, stores observations and runs the personalization model. The design focuses on four memory domains: visual recall, working-memory sequences, spatial patterns and object associations. The same physical controls support all four domains, reducing interaction complexity while allowing the software to vary task demands.

The contribution is a complete, explainable architecture for closed-loop memory training. The system links accessible hardware, structured telemetry, a local database, supervised prediction and rule-based safety constraints. It also defines an evaluation protocol that can test usability and learning outcomes without making unsupported diagnostic claims.

2. RELATED WORK

Working memory includes separable verbal and visuospatial processes, so a single total score cannot fully describe performance [1], [2]. Sequence span, spatial reproduction and associative recall provide complementary observations when task conditions remain consistent. Spaced practice also improves long-term retention compared with repeated practice concentrated in one session [3]. These findings motivate domain-specific activities and scheduled delayed-recall trials.

Adaptive learning systems commonly adjust task difficulty from observed performance. In a child-facing assistive system, however, personalization must remain interpretable and must avoid abrupt changes based on sparse data. The proposed design therefore combines a predictive model with explicit rules for cold start, confidence thresholds, maximum difficulty changes and educator override. This hybrid approach keeps the recommendation process inspectable.

3. SYSTEM ARCHITECTURE

The architecture contains four layers. The device layer captures button events and provides light or sound feedback. A serial protocol transfers timestamped events to the application layer. The application presents stimuli, scores responses and writes structured records to SQLite. The intelligence layer constructs learner-history features and scores candidate next activities. The dashboard exposes domain trends and recommended actions to the educator.



Fig -1: Proposed hardware and edge AI architecture

3.1 Embedded Interaction Unit

The console uses an ESP32-S3 development board, four large colored push buttons arranged as a two-by-two grid, a dedicated Start or Hint input, WS2812B indicators and a buzzer. Firmware performs debouncing, event timestamping, output control and serial packet handling. The embedded unit does not make learning decisions, which keeps its behavior deterministic and simplifies verification.

3.2 Application and Data Layer

A Python dashboard presents stimuli and instructions, receives newline-delimited JSON events through USB serial and calculates trial outcomes. SQLite tables retain learner identity, session metadata, activity configuration, individual responses and review schedules. Local operation reduces dependence on Internet connectivity and limits unnecessary exposure of child data.

4. PERSONALIZATION METHOD

Each completed trial produces a feature vector containing memory type, difficulty, sequence length, stimulus duration, recent accuracy, recent valid response latency, hint rate, days since the previous review and the latest delayed-recall score. Features use only previously completed trials so that target information cannot leak into the prediction.

A Random Forest classifier estimates the probability that the learner will complete each candidate activity successfully. Candidate generation filters activities by domain, review status and educator settings. The decision layer then selects a task near a configurable success range. A very easy prediction permits a small difficulty increase, a low prediction triggers an easier task or additional support, and uncertain cases retain the current level.

Signal	Purpose
Accuracy	Measures correct responses within a trial
Latency	Measures time from prompt to valid input
Span	Records the longest correctly recalled sequence
Hints	Indicates the assistance required
Delayed recall	Measures retention after a scheduled interval

Table -1: Core learner telemetry

4.1 Cold Start and Guardrails

New learners initially receive short, low-difficulty activities across all four domains. Until sufficient history exists, a deterministic rule set controls progression. The model cannot change difficulty by more than one level, cannot repeatedly select the same activity, and must prioritize reviews that are due. Educators may lock a domain, repeat an activity or override a recommendation.

4.2 Spaced Repetition

Correct delayed recall increases the interval before the next review, while an incorrect response shortens it. The scheduler records the due date and the evidence used for the interval change. This design allows retention to influence future selection without hiding the decision behind the machine-learning model.



Fig -2: Closed-loop execution and feedback flow

5. IMPLEMENTATION WORKFLOW

A session begins when the educator selects a learner. The application loads the learner profile, identifies due reviews and requests a recommendation. It then sends stimulus and feedback commands to the controller. During the response window, the console reports validated button events with millisecond timestamps. The application scores the response, updates the database and displays concise feedback before the next trial.

The communication protocol uses small JSON records with message type, trial identifier, input code and timestamp fields. Explicit trial identifiers prevent delayed events from contaminating the next activity. The application logs malformed packets and can continue in a keyboard simulation mode when hardware is unavailable, supporting repeatable software testing.

6. EVALUATION FRAMEWORK

The proposed evaluation separates engineering verification from a later learning study. Engineering tests verify input debounce, event ordering, serial recovery, stimulus timing, scoring rules, database integrity and deterministic fallback behavior. A pilot usability assessment can measure completion rate, invalid inputs, hint use and educator intervention. Any study involving children requires institutional approval, informed consent from guardians and assent appropriate to the participant.

Metric	Interpretation
Task accuracy	Immediate performance at the assigned difficulty
Response latency	Efficiency of a valid response
Maximum span	Highest sequence length recalled correctly
Review success	Retention at the scheduled revisit
Recommendation acceptance	Educator agreement with suggested tasks

Table -2: Proposed evaluation measures

A controlled study may compare adaptive training with fixed-difficulty training over the same number of sessions. The primary analysis should examine change from baseline in domain-specific accuracy and delayed recall. Secondary measures may include latency, maximum span, dropout, hint use and usability. Participant characteristics, session count and missing-data handling must be reported before interpreting any effect.

7. PRIVACY ETHICS AND LIMITATIONS

The system stores only the identifiers and performance records required for training and analysis. Access controls, encrypted backups and a defined retention period should protect the database. Reports should use pseudonymous participant codes. Educators and guardians must understand that the model recommends activities and does not diagnose a disability or replace professional assessment.

The current work defines a prototype architecture and evaluation plan. It does not report clinical effectiveness. Performance may vary with motor ability, attention, sensory preferences and familiarity with the interface. A small pilot cannot establish generalization across diagnoses or age groups, and model explanations based on feature importance require cautious interpretation.

8. CONCLUSION

The proposed AIoT platform integrates accessible physical interaction, local multimedia activities, structured telemetry and explainable adaptive selection. Its separation of embedded control from edge analytics simplifies implementation and supports privacy-preserving operation. The next stage is to complete hardware integration, verify all scoring and communication paths, and conduct an ethically approved pilot using the defined baseline and delayed-recall measures.

ACKNOWLEDGEMENT

The authors acknowledge the guidance and facilities provided by the Department of Electronics and Telecommunication Engineering, Siddhant College of Engineering, Sudhambare.

REFERENCES

[1] A. Baddeley, Working Memory, Thought, and Action. Oxford University Press, 2007.

[2] A. Baddeley, 'The episodic buffer: a new component of working memory?' Trends in Cognitive Sciences, vol. 4, no. 11, pp. 417-423, 2000.

[3] N. J. Cepeda, H. Pashler, E. Vul, J. T. Wixted, and D. Rohrer, 'Distributed practice in verbal recall tasks: a review and quantitative synthesis,' Psychological Bulletin, vol. 132, no. 3, pp. 354-380, 2006.

[4] L. Breiman, 'Random forests,' Machine Learning, vol. 45, pp. 5-32, 2001.

[5] Espressif Systems, ESP32-S3 Series Datasheet, 2024.

[6] Python Software Foundation, Python 3 Documentation, 2026.

[7] SQLite Consortium, SQLite Documentation, 2026.

[8] World Health Organization and UNICEF, Global Report on Assistive Technology. Geneva: WHO, 2022.