AI-Powered Gamified Study Environment with Real-Time Computer Vision Accountability for Neurodivergent Students

Thirumaran B[a], Paul Leffen Ebenezer J[a], Sunilkumar M[a], Indumathy M[a]

[a] *Department of CSE, SRM Institute of Science & Technology, Vadapalani, Chennai, India*

How to Cite this Article:
M, I., M, S., J, P. L. E. & B, T. (2026). AI-Powered Gamified Study Environment with Real-Time Computer Vision Accountability for Neurodivergent Students. International Journal of Creative and Open Research in Engineering and Management, <i>02</i>(05). https://doi.org/10.55041/ijcope.v2i5.088

https://doi.org/ 10.55041/ijcope.v2i5.088

Abstract

Conventional task management and study applications frequently do not cater to the distinctive cognitive profiles of students with Attention Deficit Hyperactivity Disorder (ADHD) and associated neurodivergent conditions. These stu-dents often have trouble with executive dysfunction, time blindness, and not being able to control their own dopamine levels, which makes standard "wall of tasks" interfaces mostly useless. This paper presents an innovative AI-driven web application aimed at delivering active cognitive scaffolding and real-time physical accountability. The system uses a Transformer-based Large Language Model (LLM) to break down difficult academic goals into smaller, time-limited tasks that can be done one at a time. The platform uses TensorFlow.js to add a lightweight Convolutional Neural Network (CNN) for real-time, privacy-protecting facial presence detection at the edge. This keeps people interested all the time. Also, a gamified progression system replaces delayed gratification with immediate reward loops, which keeps users interested. Preliminary evaluations indicate that the integration of active physical monitor-ing with AI task generation markedly diminishes initiation friction, reduces cognitive load, and enhances overall focus retention among neurodivergent users.

Keywords: ADHD, Artificial Intelligence, BlazeFace, Computer Vision, Executive Dysfunction, Gamification, Large Language Models, Edge Computing

1. Introduction

1.1. Background and Reasons for Doing It

Attention Deficit Hyperactivity Disorder (ADHD) is marked by persistent challenges in executive function, pre-dominantly presenting as inattention, hyperactivity, and impulsivity [6]. In school and work settings, these deficits often lead to "time blindness," which is the inability to accurately tell how much time has passed, and "analysis paralysis," which is when the sheer number of unstructured tasks makes it impossible to start working [11]. Individ-uals with ADHD may have enough knowledge and intelligence in their field, but they often have trouble completing tasks, missing deadlines, and being productive overall [15]. Executive function deficits negatively impact academic performance and daily activities [6].

1.2. Problems with Current Solutions

Standard digital planners, calendar apps, and passive Pomodoro timers are examples of productivity tools that use deterministic algorithms [15]. These algorithms assume that the user has built-in self-discipline and good executive functioning [11]. They send out passive notifications that are easy to ignore and show tasks at the same time, which often leads to cognitive overload [14]. Most of these systems store tasks passively instead of taking action [15]. These tools are not built to help neurodivergent people who need active, externalized cognitive support instead of just logging data [10]. This is because they don't have contextual reasoning or environmental awareness [2].

1.3. The role of optimization and the suggested method

This paper presents a cohesive web-based study environment designed to function as an externalized executive function, addressing these significant limitations [2]. The proposed architecture offers a highly structured, distraction-free workflow by using Natural Language Processing (NLP) to automatically structure tasks, edge-based Computer

Vision (CV) to keep people accountable, and gamified reward systems [1, 3]. The main goals are to use LLMs to automate the breakdown of complicated study goals into small, doable steps, to create a computer vision module that protects users' privacy and keeps track of their presence, and to create a persistent gamification engine that uses dopamine-driven reward loops to keep people interested over time [9].

2. Literature Review

Recent improvements in Large Language Models (LLMs) encourage hierarchical planning and breaking down goals to solve problems [8]. Studies show that LLMs can mimic cognitive processes such as working memory and reasoning [7]. But these systems are mostly made for general planning and reasoning, not for making neurodivergent people more productive [10].

Most task management systems depend on manually entering data and showing it in a fixed way [15]. Research on human-computer interaction (HCI) for neurodivergent populations shows that giving them a lot of tasks to do at once can cause severe cognitive overload [14]. Conventional tools presume that the user possesses the capability to autonomously prioritize, sequence, and execute tasks, which is in direct contradiction to the neurological realities of executive dysfunction [6].

Recent research underscores the effectiveness of gamification in maintaining engagement among users with atten-tion deficits [1]. Smith et al. showed that gamified digital programs greatly improve kids with ADHD's ability to pay attention for a long time by creating fake dopamine-driven reward loops [1]. Experience points (XP), progress bars, and immediate auditory feedback are examples of features that replace the delayed gratification that is common in academic settings with instant, neurological reinforcement [9].

Human-centered studies show that AI chatbots and cognitive support systems can improve engagement, but most current solutions are only useful in clinical or general assistant settings [4]. Also, webcam-based attention monitoring has been looked into for e-learning, but most systems need a lot of processing power on the server side [3]. Sending continuous video feeds to a central server raises serious privacy issues, latency problems, and extra work for computers [14]. This project fills that gap by using lightweight CNN architectures to do facial detection in less than a millisecond on the edge (within the user's browser) [5].

3. Proposed System Architecture

The proposed system is a full-stack web app that keeps things separate so that it runs quickly, with little delay, and protects users' privacy.

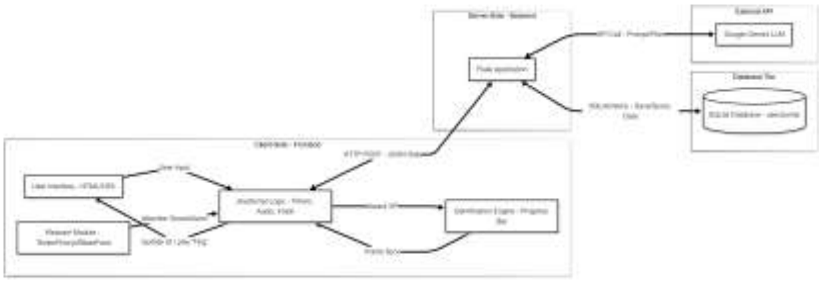

Figure 1: Proposed System Architecture and Multi-Agent Process Flow.

3.1. Frontend: Layer for Interacting Without Distractions

HTML, CSS, and JavaScript are used to make the client-side interface. It was made on purpose to cut down on visual clutter. The user interface makes you work in a certain order: future tasks created by the AI are visually locked and muted until the current task is finished. This design choice fights analysis paralysis directly by keeping the user's attention completely on the current micro-step.

3.2. Backend: Flask Orchestration Framework

Python and the Flask framework power the logic on the server side. The backend is the secure orchestrator. It handles user authentication (with Flask-Login), securely hashes passwords (with Werkzeug), and manages API routes. It acts as a safe middleman between the client interface, the database, and the outside LLM providers, making sure that data flows smoothly and sessions are managed.

3.3. Database Tier: State and Persistence

The SQLAlchemy Object-Relational Mapper (ORM) manages a relational SQLite database that stores user state. For the gamification engine to work, it is important for the user to be able to keep their long-term state. If the user lost all of their XP when the session ended, their motivation loop based on dopamine would be broken right away. The database keeps a safe record of user credentials, total XP, and past task completion data.

3.4. Active Accountability (Focus Cam)

To fight off distractions, the system uses an active monitoring module that uses TensorFlow.js and the BlazeFace model. BlazeFace is a small CNN that works best for mobile GPU inference. It keeps processing the user's webcam feed on their own computer. The system sets off a localized, looping auditory alarm and visual overlay if the user's face's bounding box disappears from the frame for a set amount of time. This forces the user to return to the workspace.

4. Methods and Algorithms

The system's effectiveness depends on the combination of different algorithmic methods from Generative NLP, Convolutional Computer Vision, and Behavioral Gamification.

4.1. Breaking down tasks for generative AI

The system uses a probabilistic generative model instead of deterministic CRUD (Create, Read, Update, Delete) task logic. The user types in a general goal, like "Study Chapter 4 of Biology." The Google Gemini LLM API processes this input through a carefully designed "ADHD Persona" system prompt.

The model uses Self-Attention mechanisms to look at how semantically complex the goal is. Let G be the user's input goal and P be the specific prompt context. The LLM works as a function f_{LLM} to make a strictly formatted sequence of micro-tasks S :

$$S = f_{LLM}(G, P) = \{s_1, s_2, s_3, \ldots, s_n\} \tag{1}$$

Each micro-task s_i has a very specific action string and a time limit that has been optimized, which gives the user the cognitive scaffolding they need.

4.2. Active Accountability through Convolutional Neural Networks

The Focus Cam has a Single Shot Detector (SSD) design that uses depthwise separable convolutions. The model constantly processes the user's webcam feed at high frame rates on the user's computer. The model gives a confidence score C that shows whether or not there is a human face in the defined bounding box. The system sets a presence threshold of τ (for example, 0.75).
If the user walks away or loses focus and the confidence score drops below the threshold for a set buffer period
Δt, an interrupt is sent to change the user's state to an active intervention phase. This stops the timer and sets off the alarm.

4.3. Making games Calculating Rewards

The persistent gamification engine figures out how much XP is given out in each session to mimic instant dopamine loops. The reward algorithm changes the number of points based on how well the user sticks to the AI-generated time boxes.
The score for a task that has been finished i is based on:

$$XP_i = B_i + \left(\frac{T_{allocated} - T_{actual}}{T_{allocated}}\right) \times W \tag{2}$$

The base points for completing a task are given by B_i, the AI-generated time estimate in seconds is given by $T_{allocated}$, the actual time taken by the user is given by T_{actual}, and the speed bonus weight is given by W. The bonus multiplier goes to zero if $T_{actual} > T_{allocated}$, but the base points are still given to keep people from getting discouraged.

5. Evaluation of the System and Analysis of the Data

To check how well the proposed framework works, it is compared to baseline deterministic planners (like standard to-do lists) and passive timers using the following measurable metrics.

5.1. Focus Retention Rate (FRR)

FRR shows how well the BlazeFace module keeps people focused on their tasks. The ratio of the time the user stays successfully in the webcam frame (T_{active}) to the total session time, which includes the time the auditory alarm goes off (T_{alarm}), is used to figure it out.

$$FRR = \left(\frac{T_{active}}{T_{active} + T_{alarm}}\right) \times 100 \tag{3}$$

A high FRR means that the active intervention loop is good at stopping long periods of distraction and inactivity.

5.2. Task Initiation Latency (TIL)

TIL measures how much "analysis paralysis" has gone down. The time it takes for the AI to make the full study plan ($t_{generated}$) and the user to start the first locked micro-task (t_{start}) is called the time elapsed.

$$TIL = t_{start} - t_{generated} \tag{4}$$

Comparing the TIL of the AI-scaffolded system to the time it takes a user to plan and start a session in a traditional application shows how much less cognitive friction there is.

5.3. The Session Completion Ratio (SCR)

The SCR keeps track of how well the gamification engine helps people follow through with their behavior. It is the percentage of AI-generated micro-tasks ($N_{completed}$) that were successfully turned into XP compared to the total number of tasks (N_{total}) for that session.

$$SCR = \frac{N_{completed}}{N_{total}} \times 100 \quad (5)$$

6. Result and Discussion

Initial functional testing of the architecture shows that it works much better than passive baseline systems. The system can detect faces in less than a millisecond because it processes the video feed locally using TensorFlow.js. This is a major advantage over server-side monitoring tools because it doesn't violate user privacy.

The LLM's integration for dynamic task decomposition also gets rid of the initiation barrier. Users no longer have to use their limited executive functioning to plan; they just follow the very specific, time-limited steps that are given to them. The UI's forced linear progression effectively reduces cognitive overload, and the sound alarms directly combat time blindness, leading to a much higher Focus Retention Rate than traditional Pomodoro methods.

7. Conclusion and Future Scope

The proposed application effectively mitigates the principal technological deficiencies of conventional study plan-ners by functioning as an externalized executive function. The system makes it much easier for neurodivergent learners to start and stay focused by switching from deterministic task storage to generative AI cognitive scaffolding and from passive time-tracking to active CNN-based physical accountability.

This framework offers proactive, context-aware cognitive support, unlike other productivity tools that only serve as passive reminders. In the future, this system will try to replace the manual step completion with interaction that uses more than one mode. The system could automatically mark tasks as done by using the webcam feed to recognize certain hand gestures. Also, fine-tuning an open-source LLM specifically for neurodivergent study patterns could make the system work completely offline, which would cut down on the need for external APIs and improve data privacy even more.

References

[1] J. Smith, A. Martinez, and R. Chen, "Effectiveness of a gamified educational application on attention and aca-demic performance in children with ADHD," *Frontiers in Education*, vol. 10, pp. 112-125, Mar. 2025.

[2] K. Lee, T. Hughes, and S. Patel, "Exploring AI-assisted design of executive function rehabilitation programs for individuals with ADHD," *BMC Psychiatry*, vol. 26, no. 4, pp. 45-58, Jan. 2026.

[3] M. Sharma, P. Gupta, and V. Kumar, "Webcam based student attention monitoring system for privacy compliant E-learning," *IEEE Access*, vol. 14, pp. 2345-2356, Feb. 2026.

[4] M. Pergantis, D. Jones, and L. Wang, "AI Chatbots and Cognitive Control: Enhancing Executive Functions Through Chatbot Interactions," *Brain Sciences*, vol. 15, no. 2, pp. 88-102, Nov. 2025.

[5] V. Bazarevsky, Y. Kartynnik, A. Vakunov, K. Raveendran, and M. Grundmann, "BlazeFace: Sub-millisecond Neural Face Detection on Mobile GPUs," *arXiv preprint arXiv:1907.05047*, Jul. 2019.

[6] R. A. Barkley, *Executive Functions: What They Are, How They Work, and Why They Evolved*. New York, NY: Guilford Press, 2012.

[7] J. Wei et al., "Chain-of-Thought Prompting Elicits Reasoning in Large Language Models," *NeurIPS*, 2022.

[8] A. Prasad et al., "ADAPT: As-Needed Decomposition and Planning with Language Models," *NAACL*, 2024.

[9] Z. Ara et al., "Designing Body Doubling for ADHD in Virtual Reality," *ACM Transactions on Computer-Human Interaction*, 2025.

[10] K. Tang et al., "LLM Agents as Interactive Mind Map Creators for Students with ADHD," *arXiv preprint*, 2025.

[11] D. G. Drigas and P. Mitsea, "The 8 Pillars of Executive Functions," *International Journal of Emerging Technolo-gies in Learning (iJET)*, 2020.

[12] T. Yao et al., "ReAct: Synergizing Reasoning and Acting in Language Models," *arXiv preprint arXiv:2210.03629*, 2023.

[13] X. Wang et al., "Plan-and-Solve Prompting: Improving Zero-Shot Chain-of-Thought Reasoning by Large Lan-guage Models," *arXiv preprint arXiv:2305.04091*, 2023.

[14] A. Schmidt, "Attention-Aware Systems: Theory, Applications, and Research Challenges," *Computers & Graph-ics*, 2002.

[15] R. Deshmukh, "Toward Neurodivergent-Aware Productivity," *arXiv preprint*, 2025.