International Journal of Engineering Trends and Applications (IJETA) – Volume 13 Issue 3, May-Jun 2026

RESEARCH ARTICLE OPEN ACCESS

Think Lab: A Flutter-Based Serious-Gaming Platform for Cognitive Bias Awareness and Critical Thinking Training

Prof. Wrushab Sirsat[1], Shruti Marne[2], Akshada More[3], Sanket Thorat[4], Prasad Patil[5]

Asst. Prof[1], UG Student[2], UG Student[3], UG Student[4], UG Student[5]

Department of Information Technology, Zeal College of Engineering, Pune, India

Abstract

Cognitive biases systematic deviations from rational judgment affect how individuals interpret evidence, choose actions, and evaluate online information. Conventional bias awareness training is often delivered as passive reading or one-time instruction, which can limit retention and real-world transfer of knowledge. This paper presents ThinkLab, a cross-platform serious-gaming application developed using Flutter (Android, iOS, and Web) that operationalizes cognitive bias education as a short, repeatable daily practice. ThinkLab offers multiple micro games targeting bias recognition, argument quality, and inference under time constraints while providing immediate feedback and explanations to support reflection. The platform integrates Firebase Authentication and Cloud Firestore to manage user identity, persist game progress (XP, levels, streaks, and best scores), record game history events, and maintain leaderboards. We detail the system architecture, design of the critical thinking game suite (Mind Detective, Social Feed, Story Mode, Debate Arena, Logic Lab, and Escape Room), and content model used to represent bias clues, scenarios, and difficulty-based unlocking. To enable publication-ready evaluation, we propose a pilot methodology that combines pre/post questionnaires on bias awareness and critical thinking self-efficacy, task-based performance measures, and usage analytics derived from app telemetry. The proposed approach demonstrates how a mobile-first, analytics-ready game platform can provide scalable and engaging training for bias awareness and critical reasoning.

Keywords: *Cognitive Bias, Critical Thinking, EdTech, Flutter, Firebase, Micro-learning, Serious Games.*

I. INTRODUCTION

Humans often rely on mental shortcuts, known as heuristics, to make decisions quickly. Although these shortcuts can be efficient, they frequently produce systematic errors known as cognitive biases. Examples include confirmation bias (favoring evidence that supports existing beliefs), authority bias (overweighting the opinion of perceived experts), framing effects (different choices depending on question wording), and bandwagon effects (following majority behavior). In modern information environments, especially social media, these biases can amplify misinformation, polarization, and poor decision making outcomes.

Bias-awareness initiatives exist in education and professional training, yet many interventions are not designed for repeated practice. Learners may understand definitions but fail to identify biases in the context. Serious games and microlearning address this gap by offering short sessions, active decision-making, immediate feedback, and a progression loop that encourages revisiting concepts. ThinkLab is built as a 'mental gym': a set of small games that can be played daily to strengthen critical thinking and bias recognition through practice.

This paper presents the design, implementation, and evaluation methodology of ThinkLab. The application targets diverse users—students, working professionals, and general learners—who seek to improve their analytical reasoning skills without requiring long training sessions. The remainder of this paper is organized as follows: Section II states the problem, Section III outlines the objectives, Section IV describes the contributions, Section V surveys related work, Section VI covers the methodology, Sections VII–XIII present the technical implementation, Section XIV describes the evaluation design, Section XV discusses the findings, and Section XVI concludes.

II. PROBLEM STATEMENT

This project addresses the challenge of delivering reliable, engaging, and scalable bias recognition practice to a wide audience. Traditional approaches, such as classroom lectures, reading materials, or one-time workshops, suffer from low retention and poor transfer to real-world scenarios. The core challenge is to create an intervention that simultaneously satisfies five key requirements.

Accessibility: The application must run on common devices (mobile and web) with a low setup burden, ensuring a broad reach regardless of device preference. Active Learning: Tasks must require recognition and reasoning, not only passive memorization, so that learners engage with bias identification in realistic contexts. Immediate Feedback: The system must provide instant explanations to reinforce correct reasoning and correct misconceptions before they are consolidated. Motivation: The platform must encourage repetition via extrinsic and intrinsic motivators including experience points (XP), level progression, daily streaks, and competitive leaderboards. Persistence and Security: All user progress and game history must be stored securely with authentication and appropriate access control to protect personal data.

III. OBJECTIVES

The objectives of ThinkLab are as follows:

1) Build a cross-platform application that provides daily training for cognitive skills and is deployable on Android, iOS, and the web from a single codebase.

2) Implement a critical thinking game suite that covers multiple interaction patterns including recognition tasks, selection from options, and narrative choices with consequences.

3) Integration of cloud persistence for identity management, progress tracking, leaderboards, and analytics-ready game history.

4) Provide an evaluation methodology suitable for an academic pilot study, including measurable pre- and post-learning outcomes and engagement analytics.

IV. CONTRIBUTIONS

This study makes four primary contributions to the fields of serious games and cognitive training.

First, a modular Flutter implementation of a cognitive training platform with a consistent design language and fully reusable widget components, enabling future extensions with minimal engineering effort.

Second, a critical-thinking game suite that includes line-by-line bias identification (Mind Detective) and scenario-based bias reflection (Bias Detective), both grounded in the debiasing literature.

Third, a Firebase-backed persistence model using Firebase Authentication and Cloud Firestore for secure user profiles, leaderboards, and append-only game history events that support analytics pipelines.

Fourth, a practical evaluation plan with measurable metrics for assessing both learning outcomes (accuracy and recall) and engagement (streaks, retention, and session frequency) suitable for IRB-approved pilot studies.

V. RELATED WORK

Debiasing research consistently shows that awareness alone is insufficient for durable change [1]. Repeated practice with corrective feedback is essential for overcoming entrenched heuristics [2]. Microlearning, which distributes practice across short sessions, supports the spacing effect and reduces cognitive overload compared to massed training [3].

Serious games have demonstrated improved engagement by turning learning tasks into challenges with reward loops [4]. In the domain of human-computer interaction, immediate feedback and progressive difficulty are known to help users remain in a flow state, thereby reducing dropout rates [5]. ThinkLab combines all these principles through timed challenges, instant explanations, and level-based unlocking of most difficult modules.

From an engineering perspective, Flutter enables a single codebase for mobile and web deployment [6], whearas Firebase provides scalable managed services for authentication and NoSQL storage [7]. This combination supports rapid prototyping and deployment of educational applications, making it an appropriate technology stack for this project.

Prior work on mobile-based cognitive training applications, such as cognitive rehabilitation tools for clinical populations, has demonstrated feasibility but has limited focus on bias-specific training for general audiences [8]. ThinkLab fills this gap by targeting everyday reasoning rather than clinical rehabilitation approaches.

VI. METHODOLOGY

ThinkLab follows a client-first architecture organized into five conceptual layers, each with clear responsibilities and boundaries. The design prioritizes the separation of concerns, allowing individual screens and modules to be developed and tested independently.

A. Presentation Layer

Flutter screens handle all user-facing interactions including authentication, home navigation, game play, profile management, and leaderboard display. Each screen is a self-contained widget tree that reads state from the Provider layer and dispatches updates upon completion.

B. State Layer

A Provider-based AuthProvider holds the current login state and UserModel object. It exposes reactive streams to screens and provides the updateProgress() method, which applies XP gains, recalculates levels, updates streaks, and persists the changes to Firestore.

C. Navigation Layer

The go_router package provides declarative route definitions that, enable consistent navigation on both mobile and web platforms. Each game and screen is exposed as a named route, supporting direct deep links and browser-standard back-button behavior on web builds.

D. Persistence Lifecycle

The application lifecycle is as follows : (i) the user signs up or signs in; (ii) the app loads the user profile from Firestore; (iii) the user plays games; (iv) on game completion, the app updates XP, level, and streak fields and writes a game-history event document; and (v) leaderboard screens read aggregated ranking fields for display.

VII. SYSTEM ARCHITECTURE

The ThinkLab system architecture comprises five distinct modules arranged in a layered dependency structure. Higher layers depend on lower layers; lower layers have no upward dependencies, ensuring testability and maintainability. Fig. 1 illustrates the module relationships.

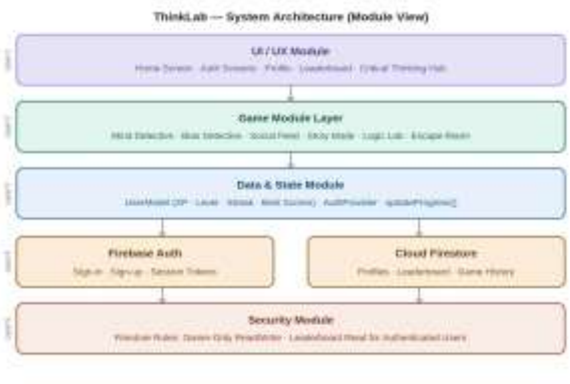

Fig. 1. ThinkLab System Architecture — Five-Layer Module View

The UI/UX Module (Layer 1) contains all Flutter screens and navigation entry points. It communicates exclusively downward to the Game Module Layer (Layer 2), which houses each game as a self-contained screen with its own scoring logic, countdown timers, and end state handling. Shared widget components - gradient buttons, game cards, and progress chips are defined once and reused across all game screens.

The Data and State Module (Layer 3) maintains the UserModel data structure and AuthProvider reactive state object. It mediates all read and write operations between the presentation layers and cloud services. The Cloud Services Module (Layer 4) consists of Firebase Authentication for session management and Cloud Firestore for the document storage. The Security Module (Layer 5) defines Firestore security rules that enforce access control at the database level, which is independent of the application code.

VIII. DATA MODEL

ThinkLab uses three Firestore collections, each designed to serve a distinct purpose within the application. Fig. 2 illustrates the schema and access semantics for each collection.

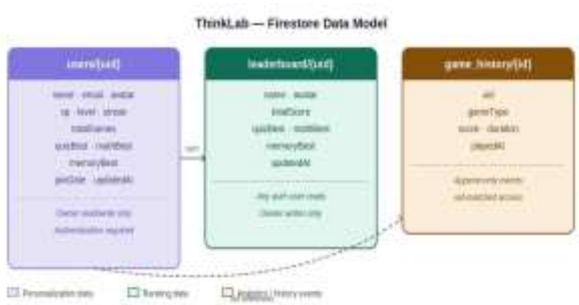

Fig. 2. ThinkLab Firestore Data Model — Three-Collection Schema

The users/{uid} collection stores all personalization fields: name, email, avatar URL, current XP, computed level, active streak length, total games played, per-game best scores (quiz, math, memory), and timestamps for account creation and last update. Access is strictly limited to the authenticated owner.

The leaderboard/{uid} collection stores the subset of user data required for ranking: display name, avatar, total score, per-game bests, and last update time. Any authenticated user may read leaderboard documents, but only the owning user may write to their entry.

The game_history collection uses auto-generated document IDs and stores append-only event records: uid, game type, score, duration in seconds, and the timestamp of play. This append-only design makes the collection suitable for downstream analytics without requiring modification of existing records.

IX. CRITICAL THINKING GAME SUITE

The critical thinking suite comprises six games, organized by cognitive skill target and unlocking tier. Games 1 and 2 are available from the beginning, games 3 and 4 are unlocked at intermediate levels, games 5 and 6 are reserved for advanced users.

A. Mind Detective (Bias Detection in Dialogue)

The player is presented with a multi-line conversation or text excerpt. Each line is rendered as an interactive element. Player must identify lines containing biased reasoning by tapping them. Scoring awards full credit for perfectly identified bias sets and partial credit for partially correct selections. Upon completion, the app reveals bias labels, explanations for each flagged line, and highlights missed biased lines in the review panel. The performance metrics are precision (correct taps/all taps) and recall (correct taps/total biased lines).

B. Bias Detective (Scenario Quiz)

The player receives a scenario drawn from a curated bank along with multiple-choice options. A countdown timer creates a time pressure analogous to real-world rapid judgment situations. If the player selects an incorrect option, the app displays the identified bias type and a detailed explanation of why the correct reasoning applies to that situation. The response time and accuracy were recorded for analysis.

C. Social Feed Simulator

A simulated social media feed presents posts with varying credibility signals. Players can choose to like, skip, or report each post. The game reveals how repeated actions create filter bubbles and echo chambers by progressively narrowing the diversity of the content shown. Reflection prompts appeared after each round to promote metacognitive awareness.

D. Story Mode

Players navigates branching narrative scenarios in which each decision has visible cognitive consequences. After each choice, the game labels the thinking type used (e.g., analytical, intuitive, biased) and shows how the narrative would have unfolded with an alternative approach to the decision.

E. Debate Arena (Level-Gated)

Players evaluate arguments on contested topics and, identify logical fallacies and evidence quality. This module emphasizes structured reasoning and requires prior familiarity with the bias types introduced in earlier games.

F. Logic Lab and Escape Room (Advanced)

Logic Lab presents inference-from-evidence puzzles in which players must reach conclusions using only the provided data. The Escape Room combines multiple reasoning skills in a multi-step problem-solving scenario with a narrative frame, unlocked only at the highest user level.

X. ROUTING AND NAVIGATION

All critical thinking games are exposed as named routes through the go_router package. The route hierarchy follows the pattern /critical/hub for the game selection screen and /critical/{game-name} for individual games (e.g., /critical/mind-detective, /critical/logic-lab, /critical/escape-room).

This design offers three practical benefits. Direct navigation allows the hub screen to push any game by name without knowledge of the widget constructors. Browser URL updates mean that web users see meaningful bookmarkable URLs that change as they navigate between games. Future deep linking enables sharing a specific game module directly via a URL, supporting referral flows and researcher-directed study participation.

XI. IMPLEMENTATION DETAILS

The implementation uses the Flutter stable channel with Dart for all the modules. The key engineering decisions are as follows:

State management is handled exclusively through the Provider, a lightweight reactive state library. The AuthProvider is injected at the application root, so all screens can read the current user state without prop drilling.

The reusable UI components included gradient-filled primary buttons, outline secondary buttons, game-card widgets with title, icon, and progress chip, and difficulty selector widgets that visually represented three tiers (Beginner, Intermediate, Advanced). Consistency across screens is enforced by sourcing all colors and text styles from a central AppTheme class.

Timing mechanics use Dart's Timer class wrapped in a StatefulWidget mixin to manage countdown timers independently of the game logic. Progress indicators are updated on each tick without rebuilding the full widget tree.

The level and unlock system derives the user's level from the accumulated XP using a predefined threshold table. Advanced game modules check the current level against the minimum threshold before rendering navigation options, showing them as locked with a visual indicator when the threshold is not met.

Firestore writes are batched where possible and always occur at stable completion points (game end or explicit save action) rather than at every user interaction, minimizing unnecessary network traffic and Firestore operation costs.

XII. SECURITY AND PRIVACY

ThinkLab uses Firebase Authentication as the sole gateway for personalized data. All screens that display or modify user data require an active authenticated session; unauthenticated users are redirected to the sign-in screen by AuthProvider.

The Firestore security rules enforce three access policies. For users/{uid}, only authenticated users whose UID match the document ID can read or write. For leaderboard/{uid}, any authenticated user can read all entries, but only the owner can write to their own entry. For game_history documents, the document's uid field must match the requesting user's UID for all operations, preventing cross-user data access, even if a document ID is known.

For research deployments, ThinkLab supports anonymized operations by replacing personal names and email fields with participant codes at the time of account creation. This approach satisfies the typical IRB requirements for minimizing personal data collection while retaining the ability to link gameplay events to participant records.

XIII. TESTING AND VALIDATION

A layered testing strategy is recommended for ThinkLab prior to the deployment of research.

Functional tests verified that each named route opened the correct game screen and that back navigation returned to the hub without state corruption. These tests used Flutter's integration test framework with a real Firebase Emulator for Firestore interactions.

Data integrity tests verify that Firestore writes on game completion contain correct field values, that XP calculations respect the level threshold table, and that the leaderboard entries are updated atomically with user profile documents.

Usability checks assessed the readability of game instructions, sufficiency of tap-target sizes for mobile interaction (minimum 48×48 logical pixels per Material Design guidelines), and clarity of countdown timer visuals across a range of screen sizes from small phones to desktop browsers.

Performance checks measure animation frame rates and list rendering smoothness on representative low-end Android devices, ensuring that the application remains usable on devices with limited GPU and memory resources.

XIV. EVALUATION DESIGN

The pilot evaluation was designed to assess two primary outcomes: learning gains in bias awareness and critical thinking self-efficacy, and user engagement with the platform over a seven-day period.

A. Participants and Duration

A sample of N = 20–50 participants drawn from university student populations or professional communities will use ThinkLab for seven consecutive days. The recruitment criteria excluded participants with prior formal training in cognitive bias theory to avoid ceiling effects.

B. Instruments

A pre-test questionnaire administered before the first use measured baseline bias awareness (10 scenario-based items) and critical thinking self-efficacy (8 Likert-scale items adapted from established instruments [9]). The same questionnaire was administered as a post-test on Day 8. Additional post-test items assessed perceived usefulness, enjoyment, and intention to continue using the application.

C. Performance Metrics

The Mind Detective metrics include precision (correct taps/all taps), recall (correct taps/total biased lines in the scenario), and completion time per scenario. The Bias Detective metrics include per-trial accuracy, mean response time, and a frequency table of the most commonly missed bias categories. These metrics were extracted directly from the game_history collection.

D. Engagement Analytics

Engagement is measured by sessions per day, streak length at study end, and Day-1 to Day-7 retention rate. These metrics are derived from game_history timestamps and user profile streak fields, without requiring additional instrumentation.

E. Statistical Analysis

Paired t-tests (or Wilcoxon signed-rank tests if normality assumptions are violated) will compare pre- and post-test questionnaire scores. Descriptive statistics and regression analyses will examine relationships between engagement metrics and learning gains, controlling for prior digital literacy level.

XV. DISCUSSION

The ThinkLab design addresses a recognized gap in the bias-training literature: the need for a scalable, engaging, and repeatedly-playable intervention that can be delivered on personal devices without instructor facilitation. By grounding game mechanics in established principles from debiasing research (repeated practice, immediate corrective feedback) and engagement design (XP, streaks, progressive difficulty), the platform is positioned to produce measurable improvements in bias recognition over short study periods.

The choice of Flutter as the implementation framework directly serves the accessibility objective. A single codebase compiled to Android, iOS, and web eliminates platform fragmentation as a barrier to participation. Firebase Authentication and Firestore provide a managed, standards-compliant backend that scales to research sample sizes without operational overhead.

A key limitation of the current implementation is the reliance on a fixed scenario bank for Bias Detective and Mind Detective. As participants progress, they may encounter repeated scenarios, reducing ecological validity. Future iterations should implement an adaptive content selection algorithm that draws scenarios matched to each user's demonstrated weakness areas, as identified from their game_history records.

A second limitation is the short evaluation window. Seven days is sufficient to detect initial learning gains but cannot address long-term retention or transfer to real-world decision-making contexts. Longitudinal studies with follow-up assessments at one month and three months would provide more robust evidence of intervention effectiveness.

XVI. CONCLUSION

This paper presented ThinkLab, a Flutter-based serious-gaming platform designed for daily cognitive bias awareness and critical thinking training. The platform implements a five-layer modular architecture, six game types targeting different reasoning skills, a Firebase-backed persistence model, and a practical pilot evaluation design.

ThinkLab contributes a working proof-of-concept demonstrating that micro-game-based bias training can be delivered at scale on consumer devices without instructor facilitation. The platform's open analytics architecture—grounded in append-only game history events—makes it suitable for integration with research data pipelines.

Future work will focus on adaptive difficulty and content selection driven by individual performance analytics, longitudinal evaluation studies extending beyond the initial seven-day pilot window, institutional deployment and integration with learning management systems, and expanded game types addressing additional cognitive biases and reasoning fallacies not covered in the current suite.

ThinkLab's modular, cross-platform implementation makes it a practical foundation for future research and development in the intersection of serious games, mobile EdTech, and cognitive psychology.

ACKNOWLEDGMENT

The authors thank all pilot study participants for their time and feedback. No external funding was received for this work.

REFERENCES

[1] D. Kahneman, Thinking, Fast and Slow. New York: Farrar, Straus and Giroux, 2011.

[2] C. R. Sunstein and R. Thaler, Nudge: Improving Decisions About Health, Wealth, and Happiness. New Haven: Yale University Press, 2008.

[3] A. Janszky, “Micro-learning in higher education: A systematic review,” Educational Technology Research and Development, vol. 69, no. 4, pp. 1815–1836, 2021.

[4] J. P. Gee, What Video Games Have to Teach Us About Learning and Literacy. New York: Palgrave Macmillan, 2003.

[5] M. Csikszentmihalyi, Flow: The Psychology of Optimal Experience. New York: Harper & Row, 1990.

[6] Google LLC, “Flutter – Build apps for any screen,” 2023. [Online]. Available: https://flutter.dev

[7] Google LLC, “Firebase Documentation,” 2023. [Online]. Available: https://firebase.google.com/docs

[8] T. Hardy, "Mobile applications for cognitive rehabilitation: A scoping review," Journal of NeuroEngineering and Rehabilitation, vol. 18, no. 1, pp. 1–14, 2021.

[9] P. A. Facione, "Critical thinking: A statement of expert consensus for purposes of educational assessment and instruction," American Philosophical Association, 1990.

[10] A. Halpern, Thought and Knowledge: An Introduction to Critical Thinking, 5th ed. New York: Psychology Press, 2013.