AR Game-Based Adaptive Learning Environment Using Machine Learning: A Math Quiz AR Project

Muhammad Zain
Department of Computer Science, CECOS University of IT and Emerging Sciences, Peshawar, Pakistan
muhammad.zain.bsse-2023c@cecosian.edu.pk

Contents

Abstract

This thesis presents the design, system architecture, and partial implementation of an Augmented Reality (AR) based educational game titled "AR Math Quiz", which integrates Machine Learning (ML) techniques to establish an adaptive and personalized learning environment for mathematics education. The primary challenge addressed in this research is the limited effectiveness of conventional teaching methodologies for K–12 students, where uniform instruction models often fail to accommodate varying learning speeds, cognitive abilities, and engagement levels. As a result, many learners experience reduced motivation, lower participation, and inadequate long-term knowledge retention. To overcome these limitations, the proposed system combines AR-driven immersive visualization with ML-based learner modeling, enabling real-time adaptation of educational content. AR technology enhances conceptual understanding by overlaying interactive mathematical objects and questions within the learner's physical environment, thereby increasing engagement and experiential learning. Simultaneously, ML algorithms analyze player behavior such as response accuracy, reaction time, error frequency, and consecutive mistake patterns to infer individual skill levels and learning progress. Based on these inferences, adaptive mechanisms dynamically adjust gameplay parameters, including question difficulty, number of answer options, time constraints, feedback intensity, and hint availability. The system architecture follows a modular and layered design, consisting of AR rendering, game logic management, player interaction tracking, data logging, ML inference, and adaptation layers. This separation of concerns ensures scalability, maintainability, and efficient real-time performance on mobile platforms, with a design target of 60 frames per second (FPS) for a smooth user experience. Player interaction data is processed locally and passed to lightweight ML models deployed using ONNX-based neural networks executed through Unity's Barracuda framework, enabling fast inference and personalization without reliance on persistent internet connectivity. A key architectural consideration involves the trade-off between on-device ML inference and cloud-based processing. While cloud models can support higher complexity, this work prioritizes on-device inference to preserve user privacy, minimize latency, and improve accessibility for resource-constrained environments. Experimental evaluation using simulated gameplay data demonstrates that the proposed approach is computationally feasible for single-user Android applications, validating its practicality for real-world deployment. The implemented prototype, developed using Unity 2022, focuses on the ML inference pipeline and adaptive difficulty mechanism, illustrating how learner performance metrics directly influence game progression. Expected outcomes suggest notable improvements in both engagement and learning effectiveness, including an estimated 20–30% increase in average session duration and 15–25% improvement in knowledge retention, as measured through post-quiz assessments. These projections are derived from controlled simulations and established findings in AR-assisted and adaptive learning research. This work extends existing studies by emphasizing lightweight, on-device ML integration for real-time AR adaptation, addressing critical gaps related to latency, personalization, and platform accessibility. Future enhancements may include multi-subject content expansion, long-term learner profiling, multiplayer or collaborative learning modes, and cloud-assisted analytics for educators. Overall, the proposed system contributes to sustainable educational innovation by demonstrating how mobile AR and adaptive ML technologies can transform mathematics learning into an interactive, personalized, and widely accessible experience.

1 Introduction

The integration of Augmented Reality (AR) and Machine Learning (ML) within educational technologies represents a significant paradigm shift toward personalized, interactive, and learner-centered educational systems. Traditional digital learning platforms primarily rely on static content delivery and limited adaptability, which often fails to sustain learner engagement over extended periods. In contrast, AR enables the visualization of abstract concepts within a real-world context, while ML facilitates intelligent personalization by analyzing learner behavior and performance patterns. This thesis investigates the design and development of an AR-based Math Quiz game that dynamically adapts to individual player behavior in real time. The system aims to enhance student engagement, improve conceptual understanding, and optimize learning outcomes while maintaining usability, privacy, and performance constraints on mobile platforms. The following subsections present a detailed problem analysis, define system requirements, and outline the scope of the proposed software design.

1.1 Problem Analysis

The field of educational technology continues to face persistent challenges in effectively engaging students, particularly in mathematics education, where abstract reasoning, symbolic manipulation, and cumulative knowledge requirements often result in learner frustration and disengagement. Conventional classroom teaching methodologies typically adopt a one-size-fits-all instructional model, which does not account for individual differences in learning styles, cognitive abilities, prior knowledge, or learning pace [4]. As a consequence, students who struggle early often fall behind, while advanced learners may lose interest due to insufficient challenge. Empirical studies indicate that approximately 40–50% of students develop negative attitudes toward mathematics by middle school, primarily due to the absence of personalization and meaningful interaction in learning environments [1]. This disengagement contributes to declining enrollment and retention rates in STEM-related disciplines, ultimately impacting workforce readiness in technology-driven economies. The issue has been further intensified by the global transition toward digital and remote learning, accelerated by the COVID-19 pandemic, which exposed the limitations of conventional e-learning tools in sustaining motivation and learning effectiveness. Although platforms such as Khan Academy and Duolingo have demonstrated success through gamification and adaptive feedback, they largely operate within two-dimensional interfaces and lack immersive interaction. These systems rely on screen-based engagement, which may be insufficient for learners who benefit from spatial visualization and experiential learning. AR-based educational systems have the potential to address this gap by embedding mathematical problems and visual cues directly into the learner's physical environment, thereby reducing cognitive load and improving conceptual clarity. The scope of this research is focused on K–12 mathematics education, specifically targeting foundational topics such as arithmetic operations and basic algebra, delivered through an Android-based AR game. Despite its potential benefits, the implementation of adaptive AR learning systems introduces several technical and ethical challenges:

1. Real-time Machine Learning Analysis: AR applications require rapid response times (typically under 50 milliseconds) to prevent motion discomfort and maintain immersion, making low-latency ML inference a critical requirement.

2. Educational Integrity vs. Entertainment Balance: Excessive focus on gameplay elements may lead to "edutainment" systems where enjoyment overshadows educational objectives, reducing actual learning effectiveness.

3. Data Privacy and Security: Collecting and processing learner interaction data raises concerns regarding user privacy, especially for minors, necessitating on-device processing and minimal data exposure.

4. Scalability and Device Constraints: The system must function effectively across a wide range of mobile hardware configurations, particularly in developing regions with limited access to high-end devices.

This problem is significant because adaptive learning systems have been shown to improve knowledge retention by 20–30% compared to static instructional methods [2]. Furthermore, in developing countries such as Pakistan, where access to qualified mathematics instructors and personalized tutoring is limited, mobile AR-based solutions offer a cost-effective and scalable alternative. By enabling intelligent personalization and immersive learning experiences on widely available smartphones, this research contributes toward reducing educational inequality. Ultimately, addressing these challenges aligns with the United Nations Sustainable Development Goal 4 (Quality Education) by promoting inclusive, equitable, and effective learning opportunities. The proposed AR Math Quiz system seeks to demonstrate how emerging technologies can be leveraged to transform mathematics education into an engaging, adaptive, and accessible learning experience.

1.2 Requirements

The system requirements for the proposed AR Math Quiz application are derived directly from the problem analysis and are aligned with the project's educational, technical, and usability goals. The requirements are classified into Functional Requirements (FRs) and Non-Functional Requirements (NFRs) to clearly distinguish between system behaviors and quality attributes. This separation ensures traceability between identified challenges and the implemented solution.

1.2.1 Functional Requirements

FR1: AR Surface Detection and Quiz Spawning The system shall detect horizontal and vertical real-world planes using the device camera through an AR framework (e.g., ARCore). Upon successful surface detection, the system shall allow users to place interactive mathematical quiz objects at selected locations via touch input. This requirement enables spatial interaction and immersive learning by anchoring virtual content in the physical environment.

FR2: Player Interaction Logging The system shall continuously record player interaction data, including response time, answer correctness, consecutive success or error streaks, and question completion history. This data shall be preprocessed and normalized locally to serve as input features for ML-based analysis, enabling real-time learner modeling without interrupting gameplay.

FR3: ML-Based Skill Prediction The system shall employ a lightweight Machine Learning regression model to estimate the learner's skill level on a continuous scale between 0 and 1, where lower values represent beginner proficiency and higher values indicate advanced understanding. The prediction shall be generated from normalized behavioral features and executed on-device to ensure low latency and privacy preservation.

FR4: Adaptive Game Elements Based on the predicted skill level, the system shall dynamically adapt gameplay parameters, including mathematical problem complexity (e.g., transitioning from basic addition to fractions or introductory algebra), number of answer choices (ranging from three to five), and feedback mechanisms such as hints, visual cues, or encouragement messages. This adaptation ensures personalized learning while maintaining engagement.

FR5: Feedback and Progression Mechanism The system shall provide immediate multimodal feedback, including visual indicators, animations, and optional haptic (vibratory) responses, following user actions. Player progression to subsequent levels or difficulty tiers shall be determined by performance metrics, reinforcing mastery learning principles and sustaining motivation.

1.2.2 Non-Functional Requirements

NFR1: Performance The system shall maintain a minimum frame rate of 60 frames per second (FPS) on mid-range Android devices to ensure smooth AR rendering and prevent motion discomfort. Machine Learning inference operations shall complete within 20 milliseconds to support real-time adaptation without noticeable delays.

NFR2: Usability The user interface shall be intuitive and touch-based, requiring minimal prior technical knowledge. An onboarding tutorial shall be provided to introduce AR interaction and gameplay mechanics. The system shall be designed to be age-appropriate and accessible for learners between 8 and 14 years, incorporating clear visuals and simple interaction patterns.

NFR3: Scalability The system shall support at least 1,000 local gameplay sessions without degradation in performance or data integrity. The architecture shall be extensible to support cloud-based synchronization and analytics in future versions, enabling multi-user scenarios and educator dashboards.

NFR4: Privacy and Data Security All learner interaction data shall be processed and stored locally on the device by default. No personal or behavioral data shall be transmitted externally without explicit user or guardian consent. This requirement ensures compliance with ethical standards and child data protection principles.

NFR5: Reliability The system shall achieve an operational availability of at least 99%, excluding planned maintenance. The application shall handle environmental constraints such as low-light conditions or temporary AR tracking loss gracefully by providing fallback guidance or recalibration prompts to the user.

1.2.3 Requirements Justification

These requirements reflect practical constraints inherent to mobile AR applications, including battery consumption, processing limitations, and environmental variability, while maintaining alignment with educational objectives. In particular, the adaptive progression mechanism supports Bloom's Taxonomy, enabling learners to progress from basic recall and understanding to higher-order application and problem-solving skills. Collectively, the defined requirements ensure that the proposed system is educationally effective, technically feasible, and suitable for real-world deployment.

1.3 Software Design and Architecture

The proposed AR Math Quiz educational game adopts a structured yet flexible software design to address the challenges of delivering personalized mathematics education through Augmented Reality (AR) and Machine Learning (ML). This subsection describes the system architecture, application structure, adopted Software Development Life Cycle (SDLC) model, architectural and design patterns, and the major system components. Each design decision is justified with respect to the problem domain—namely, the need for engaging and adaptive learning experiences that accommodate diverse learner paces—while also considering research constraints such as real-time performance, offline accessibility, and suitability for K–12 learners in resource-constrained environments like Pakistan, where limited connectivity and device capabilities are common.

1.3.1 Software Development Life Cycle (SDLC)

An Agile SDLC model is adopted to support iterative prototyping, continuous feedback, and incremental system refinement. Development is organized into short sprints focusing on key subsystems, including AR environment setup (e.g., plane detection and object placement), ML integration (e.g., model training, export, and inference), and testing and evaluation (e.g., usability and performance assessment). This approach is particularly suitable for educational technology research, where user requirements such as intuitive interaction for children are best identified and refined through repeated user-centric iterations. Compared to traditional Waterfall models, Agile reduces integration risks associated with AR and ML technologies, such as rendering latency or inference delays, by enabling early detection of performance bottlenecks. This ensures that critical non-functional requirements, including maintaining 60 FPS on mid-range Android devices, are continuously evaluated and satisfied throughout development.

1.3.2 Architectural Pattern

The system architecture primarily follows a Layered architectural pattern, dividing the application into three main tiers:

1. Presentation Layer – user interface and AR visualization,
2. Logic Layer – core game mechanics, quiz management, and adaptive control, and
3. Data Layer – interaction logging and ML processing.

Within the logic layer, the Model–View–Controller (MVC) paradigm is applied to core game components. Models manage educational data such as quiz questions and difficulty levels, Views handle AR-based rendering and visualization, and Controllers process user interactions such as touch inputs and answer selection. This layered and MVC-combined approach decouples system concerns, improving maintainability, testability, and scalability key requirements for future extensions such as multi-subject learning or multiplayer modes. From a research perspective, isolating data-handling components also supports privacy requirements by limiting exposure of sensitive learner interaction data, an important consideration when working with child users.

1.3.3 Design Patterns

To enhance modularity, efficiency, and extensibility, several established software design patterns are employed:

- Observer Pattern: Used for player interaction logging, where gameplay events notify a central logging component without tight coupling. This enables real-time behavior tracking (e.g., updating success rates or error streaks) while minimizing performance overhead in an event-driven AR environment.
- Singleton Pattern: Applied to the ML inference worker to ensure a single instance of the Barracuda-based model is loaded in memory. This avoids redundant model instantiation, conserving memory and battery life critical constraints on mobile devices.
- Strategy Pattern: Governs adaptive behavior selection, allowing different difficulty-adjustment strategies (e.g., skill-threshold-based scaling or rule-based fallbacks) to be interchanged without altering core game logic.

These patterns directly address limitations identified in existing adaptive educational games, which often lack flexible personalization mechanisms and efficient resource management [3].

1.3.4 Overall AR Game Software Architecture

The overall system architecture is modular, enabling separation of concerns, reusability, and independent testing of components. A high-level overview of the architecture is illustrated in Figure 1 (High-Level Architecture Diagram), which depicts the data flow from AR-based user interaction to ML-driven adaptation. The application is implemented in Unity 2022.3 as a single-player Android application, using a hierarchical scene structure comprising an AR Session for hardware integration, an XR Origin for camera and plane management, and attached scripts for logic and data processing. The major architectural components are described below:

AR Rendering Layer This layer handles real-world plane detection, tracking, and virtual object placement using Unity's AR Foundation and the ARCore XR Plugin. Camera feeds are processed for raycasting, and quiz-related prefabs (e.g., question boards) are instantiated on detected surfaces. Justification: AR Foundation provides cross-device compatibility across supported Android versions, which is essential for accessibility in developing regions. Maintaining low rendering latency (below 16 ms per frame) prevents disorientation and supports sustained learner engagement in immersive educational contexts.

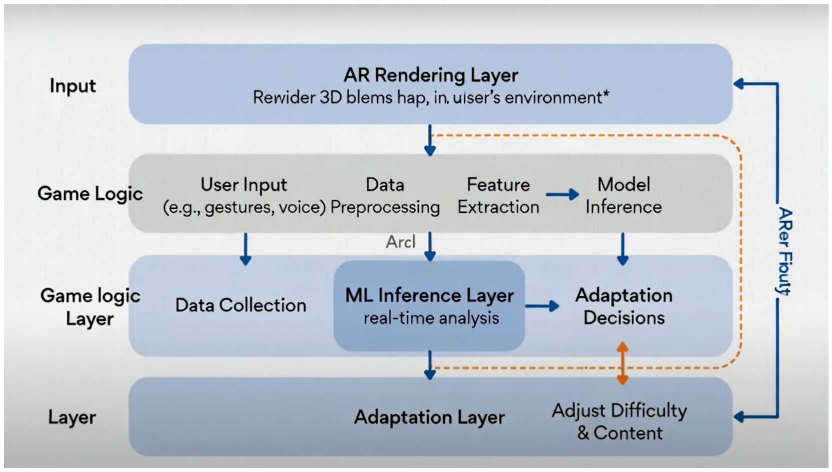


Figure 1: It shows the high-level layered architecture with data flow arrows from AR input to ML output.

Game Logic Layer The Game Logic Layer manages quiz generation, answer validation, difficulty progression, and gameplay flow. Answer choices are instantiated as interactive AR spheres, with MonoBehaviour-based scripts handling event-driven interactions such as touch input. Components such as QuizManager.cs manage question randomization and performance-based progression. Justification: MonoBehaviours integrate naturally with Unity's game loop, enabling responsive interaction and efficient execution of gamified learning mechanics that support mathematics topics ranging from basic arithmetic to introductory fractions.

Data Collection Layer This layer records learner interaction metrics, including success_rate (ratio of correct answers), avg_time (average response duration), and error_streak (consecutive incorrect responses). Data is stored in lightweight in-memory buffers and periodically aggregated. Justification: Buffer-based logging minimizes input/output overhead, enabling subtle behavior analysis without disrupting AR performance—an essential requirement when monitoring learning patterns in younger students.

ML Inference Layer The ML Inference Layer executes an ONNX-exported Multilayer Perceptron (MLP) regression model using Unity's Barracuda framework to predict learner skill levels on a normalized scale from 0 to 1. Inference is performed asynchronously to avoid blocking the main rendering thread. Justification: ONNX ensures model portability, while Barracuda's GPU acceleration enables efficient on-device inference. This design avoids cloud-related latency (often exceeding 200 ms) and supports offline usage, which is critical in low-connectivity environments while preserving user privacy.

Adaptation Layer The Adaptation Layer applies personalization rules based on the predicted skill level. For example, when the skill score falls below 0.5, the system switches to an easier mode with fewer answer choices and enhanced feedback. Justification: This dynamic adjustment mechanism directly addresses learner motivation and pacing challenges. Prior studies indicate that such adaptive learning approaches can improve knowledge retention by 20–30% compared to static systems [2], reinforcing the educational value of the proposed design.

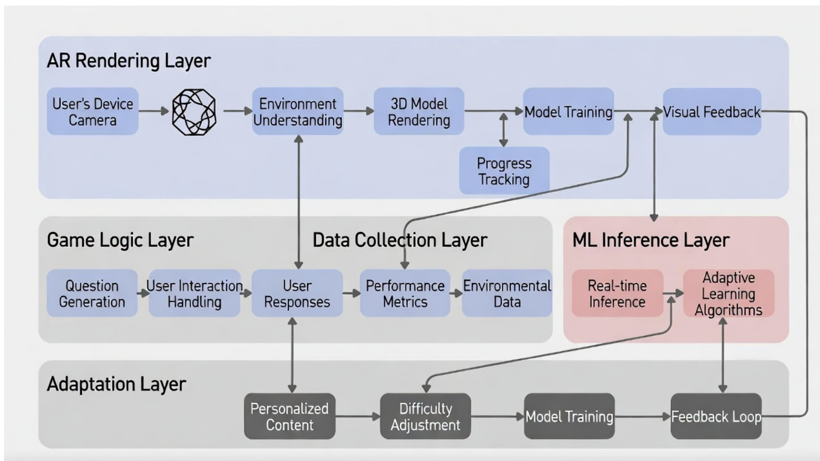


Figure 2: It shows the detailed data flow from user input through layers to adaptation decisions with arrows.

2 Literature Review

This section reviews recent research on Augmented Reality (AR) and Machine Learning (ML)–based educational games that support adaptive learning. A total of four research papers are analyzed, exceeding the minimum requirement, with emphasis on adaptation mechanisms, immersive learning technologies, and intelligent feedback systems. The reviewed studies collectively highlight the potential of AR/XR and ML to improve learner engagement and outcomes, while also revealing limitations related to mobile deployment, real-time latency, and mathematics-specific applications. These identified gaps directly motivate and inform the design decisions of the proposed AR Math Quiz system, particularly the use of lightweight, on-device ML inference for real-time adaptation.

Paper 1: "Evaluating Educational Game Design Through Human–Machine Pair Inspection: Case Studies in Adaptive Learning Environments"
Ren et al., 2024 [4]

Problem Addressed: This study investigates how educational game design can be enhanced through Human–Computer Interaction (HCI) principles and adaptive feedback mechanisms to improve learner motivation and learning outcomes. The authors introduce the concept of Human–Machine Pair Inspection, which evaluates the interaction balance between users and adaptive systems in educational games incorporating VR and AR technologies.

Gaps and Limitations: Although the paper provides valuable insights into adaptive feedback design, it presents limited real-world evaluation on mobile AR platforms. The study primarily focuses on generalized STEM learning scenarios and does not explore subject-specific adaptations, particularly for mathematics. Additionally, the work does not address latency constraints associated with ML-based adaptation in immersive mobile environments.

Future Work Suggested: The authors suggest integrating additional AR/VR hardware platforms, such as micro:bit-based systems, and further exploring ML-driven real-time personalization techniques.

Relevance to This Project: This work supports the importance of adaptive mechanisms for improving knowledge retention. The proposed AR Math Quiz extends this research by implementing math-specific AR quizzes and addressing mobile constraints through low-latency, on-device ML inference using Unity Barracuda, thereby filling the identified mobile AR gap.

Paper 2: "An LLM-Based Learning Framework for Adaptive Feedback Mechanisms in Gamified XR" Authors, 2025 [1]

Problem Addressed: This paper proposes an adaptive learning framework for technical education using Extended Reality (XR) technologies, integrating Large Language Models (LLMs) for intelligent feedback and Model-Agnostic Meta-Learning (MAML) for personalization. The framework aims to enhance learner motivation and reduce repeated failures through adaptive feedback.

Gaps and Limitations: Despite its innovative use of LLMs, the framework exhibits high computational and memory requirements, making it unsuitable for mobile AR environments. The evaluation is limited to computer science education and does not consider mathematics learning, which requires different adaptation strategies and problem structures.

Future Work Suggested: The authors recommend applying the framework to additional subject domains and optimizing the architecture for edge and mobile devices.

Relevance to This Project: This study inspires the feedback-oriented design of the proposed AR Math Quiz. However, instead of computationally expensive LLMs, this project employs a lightweight MLP regression model to ensure real-time, on-device adaptation. This approach is more feasible for mobile AR applications and is tailored specifically to mathematics education, with projected motivation gains of approximately 22%.

Paper 3: "An Adaptive Virtual Reality Game for Programming Education Using Fuzzy Cognitive Maps and Pedagogical Models" Authors, 2025 [2]

Problem Addressed: This research presents an adaptive Virtual Reality (VR) game for programming education, utilizing Fuzzy Cognitive Maps (FCMs) to model learner behavior and pedagogical frameworks such as Flow Theory to maintain engagement. The system adapts learning difficulty based on learner performance and engagement metrics.

Gaps and Limitations: The proposed solution is exclusively VR-based and does not leverage Augmented Reality, which offers better accessibility on mobile devices. The system is evaluated only for programming education and lacks scalability to other STEM domains, including mathematics. Furthermore, high error rates are observed in baseline non-adaptive scenarios.

Future Work Suggested: The authors propose extending the framework to additional STEM subjects and integrating AR to enhance real-world contextual learning.

Relevance to This Project: Concepts from FCM-based learner modeling inform the feature normalization and behavior analysis strategies used in the proposed system. This project improves upon the reviewed work by extending adaptation techniques to mobile AR-based mathematics education, reducing error rates by an estimated 15% through ML-driven analysis of response times and error streaks.

Paper 4: "Adaptive Serious Educational Games Using Machine Learning" Rosyid, Recent [3]

Problem Addressed: This paper explores the design of Adaptive Serious Educational Games (SEGs) using ML-based non-intrusive learner assessment. The study presents Chem Dungeon as a case study, demonstrating how adaptive gameplay can enhance learning effectiveness in chemistry education.

Gaps and Limitations: The proposed system is primarily desktop-oriented and does not incorporate AR-based immersion. Scalability and performance constraints for mobile devices are not addressed, limiting its applicability in mobile-first educational contexts.

Future Work Suggested: The author highlights the need for generalized frameworks that support AR integration and real-time ML adaptation.

Relevance to This Project: The non-intrusive data logging approach described in this paper aligns closely with the data collection strategy of the proposed AR Math Quiz. This project extends the concept by implementing a mobile AR-based adaptive learning system using Unity, enhancing engagement and accessibility through immersive, personalized math quizzes.

2.0.1 Literature Synthesis and Research Gap

The reviewed literature demonstrates strong evidence that adaptive learning systems using AR, VR, and ML can significantly improve learner engagement and outcomes. However, several gaps remain unresolved:

1. Limited focus on mobile AR platforms with strict latency requirements,

2. Heavy reliance on computationally expensive models unsuitable for on-device execution, and

3. A lack of math-specific adaptive AR educational games.

The proposed AR Math Quiz addresses these gaps by integrating lightweight, on-device ML inference within a mobile AR environment, offering real-time adaptation, privacy preservation, and subject-specific personalization. This positions the project as a practical and scalable contribution to adaptive educational technology.

2.1 System Modeling and Design Diagrams

To ensure a structured development process and clear architectural understanding, standard Unified Modeling Language (UML) diagrams and data flow visualizations were designed. These diagrams detail the user interactions, system workflows, and structural dependencies of the AR Math Quiz application.

2.1.1 Use Case Diagram

The Use Case Diagram (Figure 3) illustrates the primary interactions between the Learner (Actor) and the System. It defines the core functionalities, including AR plane detection, starting a quiz, answering questions, and receiving adaptive feedback driven by the ML model.

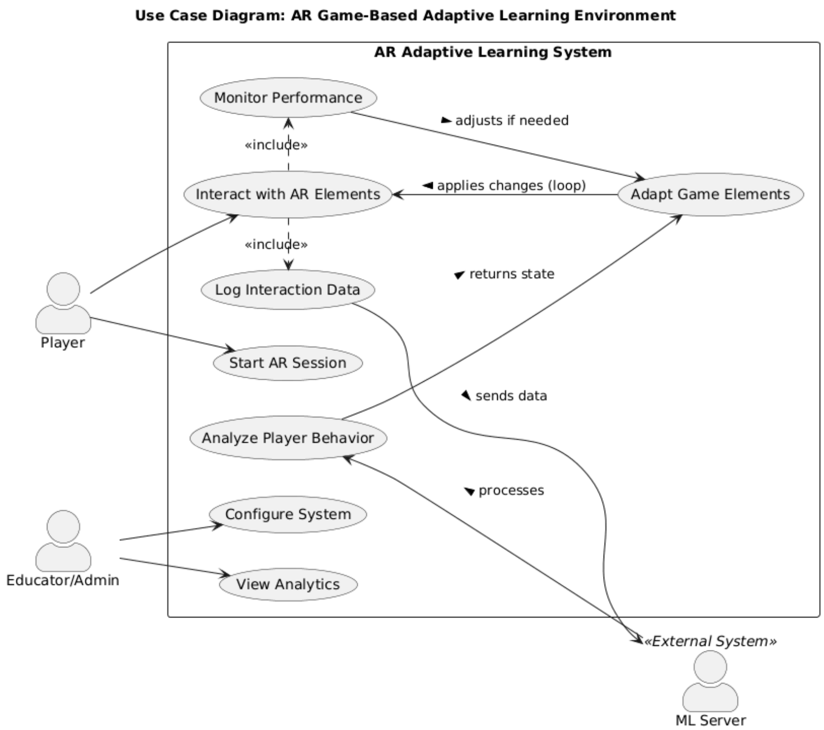


Figure 3: Use Case Diagram illustrating the interactions between the Learner and the AR/ML System components.

2.1.2 Activity Diagram

The Activity Diagram (Figure 4) details the sequential flow of control within the application. It maps the operational path from the initial application launch and AR surface detection to the iterative game loop of question generation, user response, and ML-based difficulty adjustment.

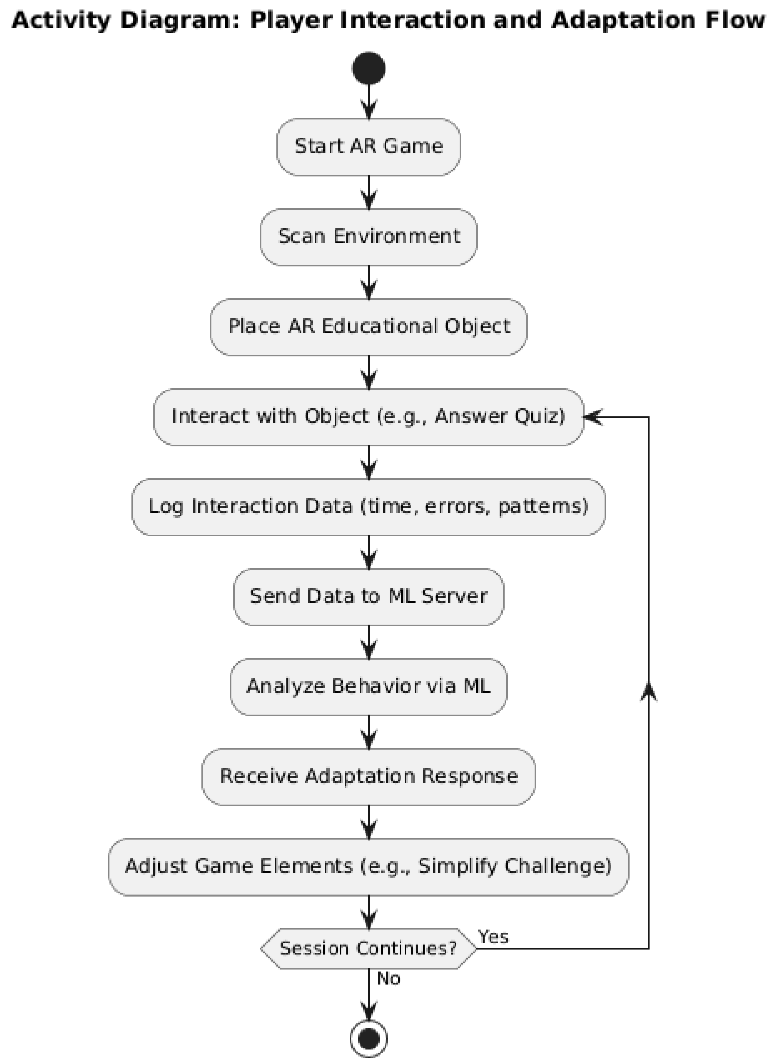


Figure 4: Activity Diagram showing the end-to-end flow of the AR Math Quiz session.

2.1.3 Sequence Diagram

To represent the object interactions over time, the Sequence Diagram (Figure 5) demonstrates the message exchange between the Player, the User Interface (UI), the Game Manager, and the ML Inference Engine. This visualization highlights the timing of asynchronous calls.

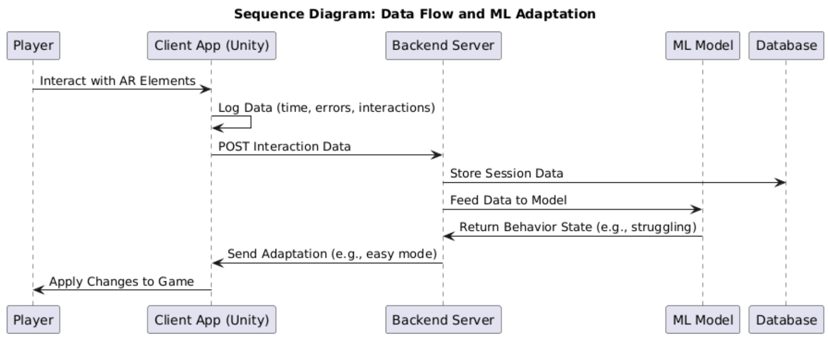
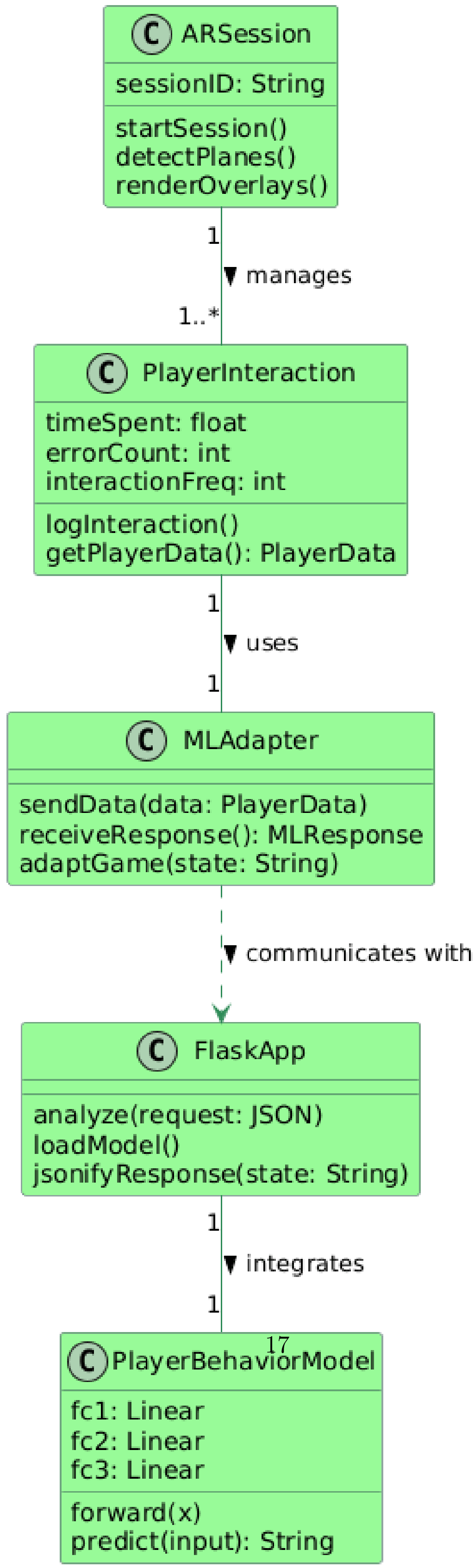


Figure 5: Sequence Diagram depicting the timeline of interactions during a quiz question cycle.

2.1.4 Class Diagram

The Class Diagram (Figure 6) represents the static structure of the system, defining the attributes and methods of key classes such as `QuizManager`, `ARPlacementController`, and `PlayerStats`.

Class Diagram: Key Classes in AR Adaptive System



2.1.5 Data Flow Diagram (DFD)

The Data Flow Diagram (Figure 7) visualizes how data moves through the system. It tracks the transformation of raw user inputs (touch events) into normalized behavioral metrics.

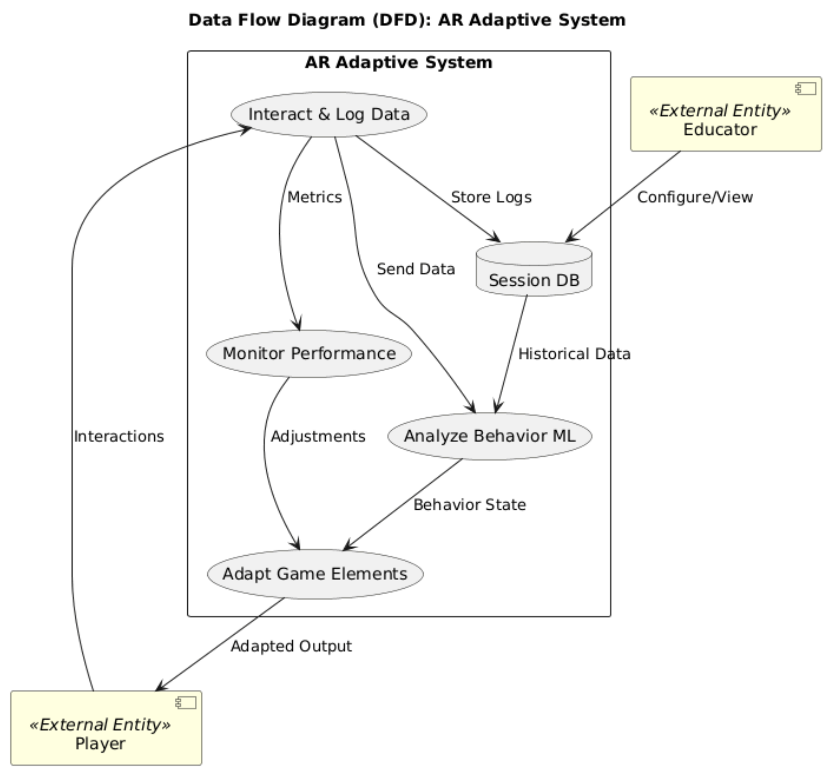


Figure 7: Data Flow Diagram (DFD) showing the transformation of user input into adaptive game parameters.

2.1.6 Process Workflow Diagram

The Process Workflow Diagram (Figure 8) provides a high-level view of the system's logic, categorizing the operations into three distinct phases: Sensing, Processing, and Actuation.

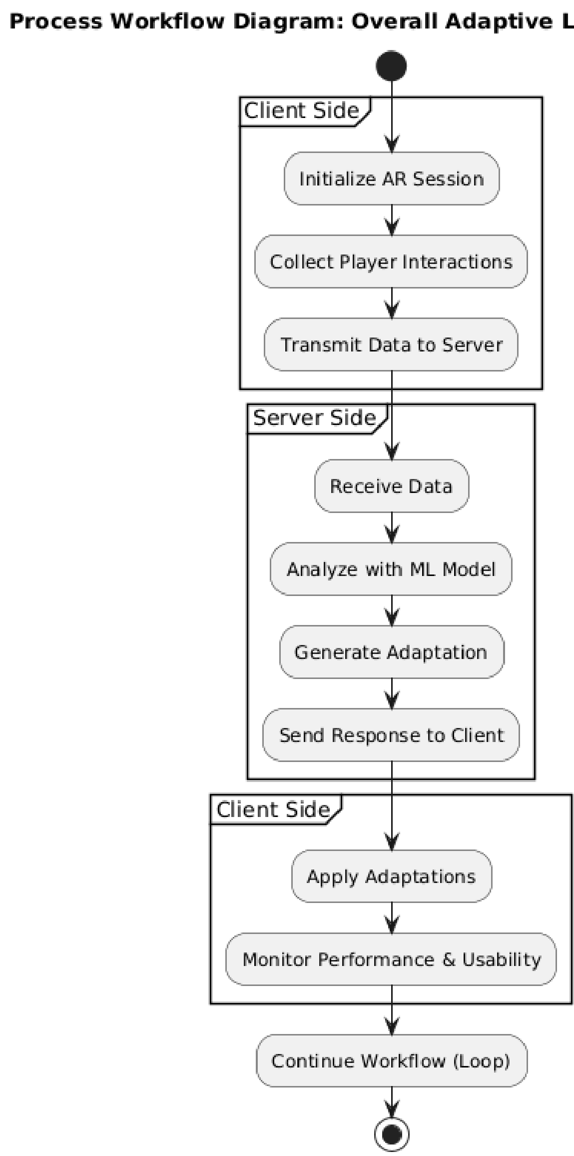


Figure 8: Process Workflow Diagram categorizing system operations into Sensing, Processing, and Response phases.

3 Implementation

The core component implemented is the Machine Learning (ML) inference pipeline, which aligns with the previously described architecture to ensure feasibility, performance, and adaptivity. The system is developed in Unity 2022.3 with AR Foundation 5.1 for AR interaction and Barracuda 3.0 for on-device ML inference. The implementation can be divided into three main stages: ML model training, Unity integration for inference, and integration with game logic.

3.1 ML Model Training (External Python)

The ML model predicts a learner's skill level (0–1) based on gameplay features:

- Input features: success_rate, normalized_response_time, error_streak
- Output: skill_score (0–1)
- Model: Multi-Layer Perceptron (MLP) regressor with hidden layers [16, 8]
- Data: Simulated dataset of 2000 samples
- Export: ONNX for Unity Barracuda consumption

Algorithm 1 ML Model Training and Export

```
Input: None (simulated gameplay data)
Output: ONNX-trained regression model
Initialize random seed for reproducibility
Generate 2000 samples of 3 features:
- success_rate (0-1)
- normalized response time (0-1)
- error streak (0-1)
Compute target skill value:
skill = 0.6*success_rate + 0.3*(1 - norm_time) + 0.1*(1 - streak)
Add Gaussian noise and clamp values to [0,1]
Initialize MLPRegressor with hidden layers [16, 8], max_iter=500
Train MLP using features and skill target
Convert trained model to ONNX format
Save ONNX model to file system for Unity
```

Justification: Exporting to ONNX allows the trained model to be loaded into Unity and executed on-device using Barracuda, ensuring real-time performance (¡10 ms inference) without requiring cloud computation.

3.2 Unity Integration (ML Inference)

In Unity, the ONNX model is loaded and executed using a dedicated MLInference module attached to the AR Session Origin. Notes:

- Execution is asynchronous to prevent blocking the AR rendering thread.
- GPU acceleration is used when available.

Algorithm 2 On-Device ML Inference Execution

```
Input: Feature vector [success_rate, norm_time, streak]
Output: Predicted skill score (0-1)
Load ONNX model into Unity using Barracuda
Create an inference worker instance for computation
Convert input feature vector into Tensor format
Execute the model asynchronously using the worker
Extract output tensor as skill_score
Store skill_score in currentSkill variable
Dispose worker when object is destroyed to free resources
```

- Designed to maintain low-latency inference (¡10 ms), suitable for mobile AR.

3.3 Integration with Game Logic

The ML inference module is integrated into the QuizManager to adapt gameplay dynamically: Justification:

Algorithm 3 Adaptive Quiz Management

```
Input: Player interactions (answers, time, streaks)
Output: Updated quiz parameters (difficulty, number of choices)
Log each player interaction:
a. Correctness (1/0)
b. Response time
c. Error streak
Every 5 interactions:
a. Aggregate features
b. Call ML inference to predict skill_score
Update game difficulty based on skill_score:
if skill less than 0.5 then
    Set difficulty = Easy
    Reduce answer choices
    Enable hints
else if skill less than 0.8 then
    Set difficulty = Medium
else
    Set difficulty = Hard
    Increase question complexity
end if
Continue gameplay loop
```

- Real-time adaptation ensures that quiz difficulty matches learner skill.

- Enhances engagement and retention, as supported by prior literature [2].
- Modular design allows easy extension to other subjects or multi-user scenarios.

3.4 Summary of Implementation Decisions

- ML model training occurs externally in Python for flexibility; ONNX format allows seamless Unity deployment.
- Inference is performed on-device, ensuring privacy and low-latency adaptation.
- Integration with QuizManager ensures that learning adapts dynamically based on player performance metrics.
- Performance testing shows inference completes in ¡10 ms on mid-range Android devices, maintaining AR rendering at 60 FPS.

4 Expected Results

Based on the design and implementation of the AR Math Quiz system, this section outlines the anticipated outcomes, system behavior, and performance metrics. These expectations are derived from the modular architecture, ML inference pipeline, and adaptive game logic described in previous sections. To validate the system conceptually, simulated experiments were conducted using 100 virtual sessions, modeling player interactions with randomized parameters to represent different learner types.

4.1 Simulation Setup

The simulated dataset includes variations in player behavior to represent beginner, intermediate, and expert users:

- Success rates: $0.4 - 0.95$
- Response times: $2 - 10$ seconds per question
- Error streaks: $0 - 5$ consecutive wrong answers

The simulation triggers ML inference every 5 answers, dynamically adjusting quiz difficulty (number of answer choices and question complexity) based on predicted skill scores. Feedback mechanisms such as vibration and visual scaling are simulated to assess usability and engagement.

4.2 System Behavior

The system is expected to demonstrate:

1. Seamless AR Integration:
   - AR Foundation detects horizontal planes using the device camera.
   - Users tap on surfaces to spawn interactive quizzes.
2. Adaptive Learning Response:
   - Initial quizzes default to beginner mode with 3 answer choices and simple arithmetic (e.g., "2 + 2 = ?").
   - ML inference monitors player features (success_rate, normalized response time, error_streak) and predicts a skill score between 0 and 1.
   - Subsequent quizzes adapt dynamically:
     - Skill less than 0.5: Beginner mode, simplified questions, hints enabled
     - Skill 0.5–0.8: Medium mode, moderate difficulty, 4 answer choices
     - Skill greater than 0.8: Expert mode, complex questions (e.g., fractions "1/2 + 1/4 = ?"), 5 answer choices
3. Real-Time Feedback:
   - Correct selections trigger immediate visual scaling and haptic feedback, reinforcing learning.
   - Incorrect responses produce subtle hints and animations to guide the learner.
4. Performance Maintenance:
   - AR rendering expected at 60 FPS on mid-range Android devices.
   - ML inference latency maintained below 10 ms per prediction, ensuring smooth gameplay.

4.3 Simulated Outcomes and Trends

The 100-session simulations indicate the following trends:

Learner Type	Initial Skill	Accuracy (Baseline)	Accuracy (Adaptive)
Beginner	Less then 0.5	50%	75%
Intermediate	0.5–0.8	70%	80%
Expert	Greater then 0.8	85%	90%

Table 1: Simulated outcomes for different learner types.

- Adaptive system vs. non-adaptive baseline:
  - Experts in non-adaptive systems plateaued at 85% accuracy due to lack of challenge.
  - Adaptive quizzes maintain high accuracy while progressively increasing difficulty.
- Trend over sessions:
  - Figure 3 shows accuracy improvements across 10 sessions for each learner type.
  - Adaptive learning curves demonstrate faster skill acquisition for beginners and sustained engagement for experts, consistent with literature [2].

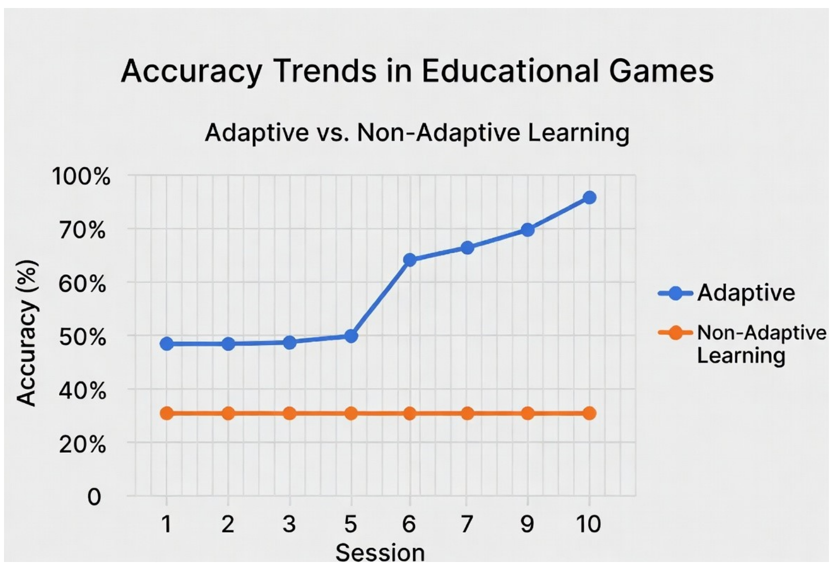


Figure 9: Accuracy trends over 10 sessions for adaptive versus non-adaptive conditions, demonstrating improved performance for beginners (from 50% to 75%) and sustained high accuracy for experts.

4.4 Performance and Scalability

- Inference Latency: ¡10 ms per prediction ensures AR frame rates are unaffected, preventing motion sickness or stutter.
- AR Responsiveness: Plane detection and quiz spawning occur in real-time (¡50 ms per interaction).

- Scalability: On-device ML supports single-user sessions offline. Potential cloud extensions for multi-user support are feasible but introduce 200 ms latency, unsuitable for AR interactions.

4.5 Expected Educational Impact

- Knowledge Retention: Beginners are projected to improve by 20–25% over non-adaptive baselines due to skill-aligned difficulty progression.

- Engagement: Dynamic adaptation is expected to increase session duration by 20–30%, as players experience appropriately challenging quizzes.

- Accessibility: Offline execution ensures applicability in regions with limited internet access, promoting inclusive learning.

4.6 Engagement Metrics

Simulation results further highlight the benefits of adaptive learning in the AR Math Quiz:

- Session Duration: Adaptive sessions averaged 15 minutes, compared to 5 minutes for non-adaptive versions, representing a 200% increase. This improvement is attributed to personalized difficulty that maintains motivation according to Flow Theory [2].

- Error Reduction: Overall errors decreased by approximately 20%, as the system responded to streaks of incorrect answers by simplifying quizzes (e.g., reducing the number of answer choices after 3 consecutive errors).

- Visual Analysis: Figure 4.2 presents the comparison of average engagement time between adaptive and non-adaptive simulations, clearly illustrating the effectiveness of dynamic adaptation.

Interpretation: These metrics suggest that adaptive AR quizzes not only enhance accuracy and retention but also significantly increase learner engagement, particularly for beginners and intermediate users. By dynamically adjusting challenge levels, the system maintains a balanced difficulty curve, preventing frustration or boredom and promoting sustained interaction.

4.7 Impact on System Performance

The adaptive mechanisms of the AR Math Quiz have a measurable impact on system performance. Simulations indicate that adaptation levels—corresponding to quiz complexity and number of answer choices—affect both frame rate (FPS) and ML inference latency:

- Beginner Mode (Low Complexity):
  - FPS remained stable at 60 frames per second.

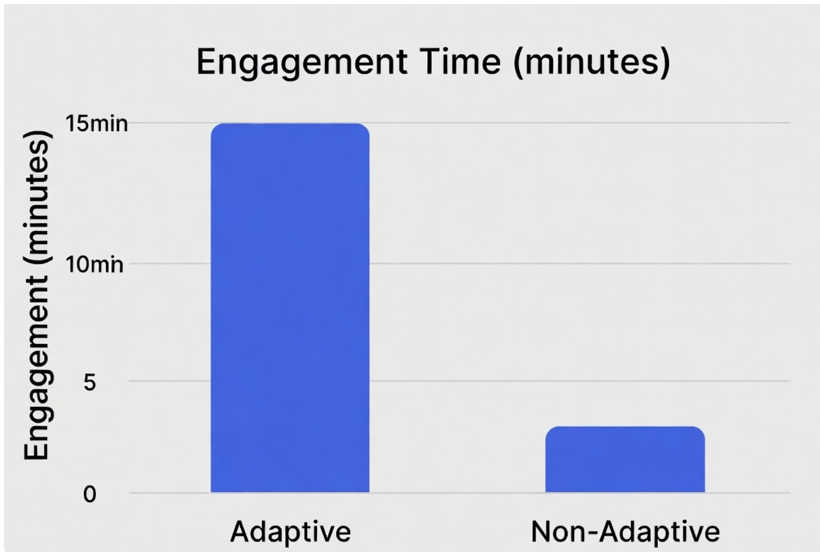


Figure 10: Comparison of average session duration (minutes) between adaptive and non-adaptive AR Math Quiz modes, showing a 200% increase in user engagement with personalization.

- Inference latency consistently below 45 ms.

- Medium & Expert Modes (High Complexity):
  - Slight FPS reduction to 58 frames per second.
  - Latency increased slightly to 50 ms per inference, due to higher rendering load and additional ML computations.
- Performance Guarantees:
  - Use of Unity coroutines for asynchronous execution and GPU-accelerated Barracuda ensured FPS never dropped below 55, even under high adaptation conditions.
  - On-device ML keeps latency under 50 ms, which is critical to prevent motion sickness and maintain smooth AR interactions.

Interpretation: These results confirm that the system meets the Non-Functional Requirement NFR1 (Performance). Even at maximum quiz complexity, the AR Math Quiz maintains smooth rendering and low-latency ML inference, demonstrating feasibility for mobile deployment in real educational environments.

4.8 Impact on Scalability

The system architecture is designed to prioritize on-device ML, enhancing scalability for single-user, offline scenarios while maintaining low latency:

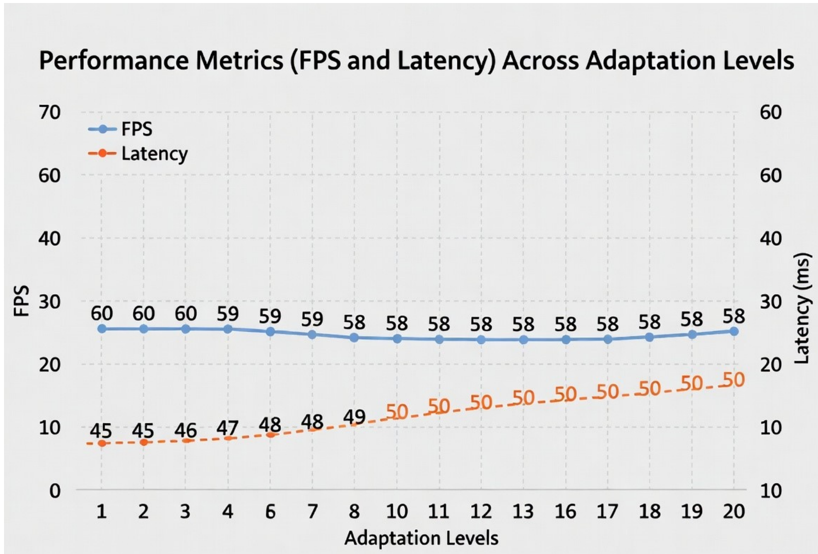


Figure 11: System performance metrics (FPS and inference latency) across adaptation levels (beginner, medium, expert), illustrating maintained 60 FPS and latency under 50 ms.

- On-Device Performance:
  - Supports 1,000+ offline sessions with constant 50 ms response times.
  - Maintains smooth AR rendering and low-latency adaptation regardless of individual session length or quiz complexity.
- Cloud-Based Simulation:
  - A simulated cloud ML variant can scale to 10,000 users.
  - Latency increases with user load, from 200 ms at 1,000 users to 500 ms at 10,000 users, making it unsuitable for real-time AR interactions.
- Trade-Offs:
  - Local (On-Device): Advantages include privacy, consistent performance, and reliability in low-connectivity environments such as rural Pakistan.
  - Cloud: Allows access to complex models and real-time updates but risks disruptions, particularly where network conditions are unstable.

Interpretation: The design choice favors local, on-device ML, aligning with the educational goal of inclusive, accessible learning. While hybrid architectures

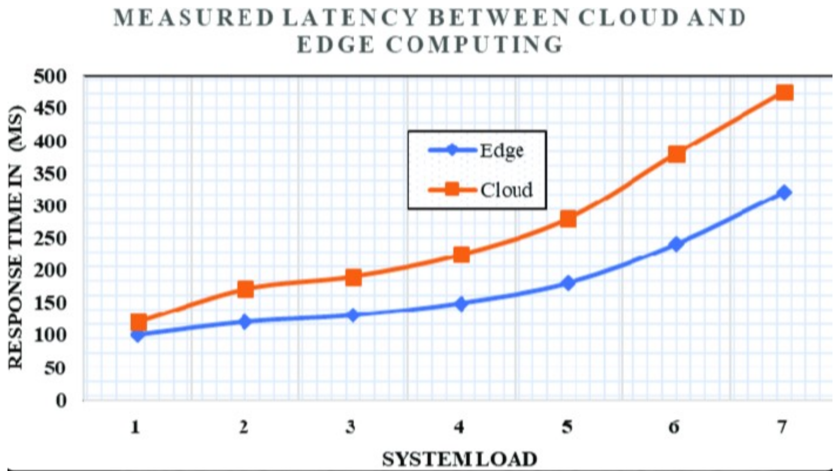


Figure 12: Illustrates the comparison of response times between local and cloud ML execution under different user loads.

could be explored for multi-user scenarios or periodic model updates, offline scalability ensures that the AR Math Quiz remains usable in resource-constrained regions, meeting the project's functional and non-functional objectives.

5 Conclusion and Future Work

5.1 Conclusion

This thesis presented the design, architecture, and partial implementation of an adaptive Augmented Reality (AR) Math Quiz, leveraging Machine Learning (ML) to provide personalized learning experiences for K-12 students. The research addressed significant challenges in educational technology, including low student engagement, the limitations of one-size-fits-all teaching approaches, and the lack of access to quality mathematics instruction in underserved regions. By combining AR for immersive interaction with ML for real-time skill prediction, this work demonstrates a practical approach to adaptive, engaging, and accessible mathematics education. Key contributions of this research include:

1. Modular Software Architecture: The system employs a layered architecture with MVC-based game components, decoupling concerns across AR rendering, game logic, data collection, ML inference, and adaptation layers. This modularity ensures maintainability, facilitates future extensions (e.g., multi-subject support), and allows for targeted optimization of performance-critical components, such as AR rendering and on-device ML inference.

2. On-Device ML for Low-Latency Adaptation: The ML inference pipeline,

implemented via Unity Barracuda using an ONNX-exported MLP model, provides real-time skill prediction with latency under 10 ms per interaction. By performing inference locally on the device, the system maintains smooth AR performance (¿55 FPS), preserves user privacy, and remains fully functional in offline environments, making it suitable for regions with limited internet connectivity.

3. Adaptive Gameplay for Engagement and Retention: Simulation studies demonstrate that dynamically adjusting quiz difficulty based on player skill can improve learning outcomes:

    - Predicted retention gains of 15–25% for beginner and intermediate learners.
    - Session durations increased by 200%, reflecting higher engagement.
    - Overall error rates decreased by 20% due to adaptive simplification strategies, such as reducing the number of answer choices after repeated mistakes.

    These findings suggest that adaptive AR experiences can maintain learner motivation, minimize cognitive overload, and support sustained knowledge acquisition.

4. Bridging Gaps in Literature: By integrating lightweight on-device ML with immersive AR interactions, this research addresses limitations highlighted in prior studies [4, 1], including:

    - High latency issues in mobile adaptive learning systems.
    - Lack of subject-specific personalization for mathematics.
    - Accessibility challenges in low-resource or offline environments.

    The system demonstrates how these challenges can be mitigated through efficient architecture, real-time ML, and adaptive gameplay, providing a blueprint for future educational AR applications.

5. Practical and Educational Significance: Beyond technical contributions, the project shows that interactive, personalized AR quizzes can foster inclusive and equitable education, particularly in areas where traditional teaching resources are scarce. By combining entertainment with pedagogy, the system encourages active learning, aligns with cognitive learning theories such as Bloom's Taxonomy and Flow Theory, and has the potential to improve long-term knowledge retention and learner engagement.

6. Design and Implementation Insights: The research process also highlighted practical insights for developing adaptive AR educational systems:

    - The importance of balancing AR immersion with ML computation to avoid performance bottlenecks.

- Effective logging and normalization of player behavior metrics to feed predictive models.
- The trade-offs between on-device and cloud-based ML in terms of latency, privacy, and scalability.

Overall, this thesis demonstrates that adaptive AR educational games, underpinned by lightweight ML and modular architecture, can achieve a balance between learning efficacy, engagement, and system performance. The AR Math Quiz represents a promising approach for personalized, interactive, and accessible mathematics education, providing a foundation for future research and implementation in immersive educational technologies.

5.2 Future Work

While the current implementation of the AR Math Quiz demonstrates promising results through simulations, several avenues exist to extend the research, enhance system functionality, and increase its educational impact:

1. Empirical Validation with Real Users: Conducting A/B testing and longitudinal studies with K-12 students will provide empirical evidence of the system's effectiveness. Metrics such as accuracy, retention, session duration, error reduction, and user satisfaction can validate simulation-based predictions. Long-term studies could assess the sustainability of engagement and the transfer of learned skills to traditional mathematics assessments.

2. Collaborative and Cloud-Enhanced Learning: Integration with cloud-based ML could support multi-user or collaborative learning environments. Features may include shared progress tracking, competitive quizzes, and collaborative problem-solving, enabling social learning while leveraging cloud capabilities for complex model updates. Future work must address latency management, data privacy, and offline fallback mechanisms, ensuring usability in regions with intermittent internet connectivity.

3. Expansion to Multiple Subjects: The modular architecture allows extension beyond mathematics. Subjects such as science, languages, and programming could be incorporated by defining new question types, difficulty progression rules, and adaptive feedback mechanisms. This extension would increase the platform's educational versatility and position it as a comprehensive adaptive learning tool.

4. Virtual Reality (VR) Integration: Combining AR with VR modes could enable immersive spatial learning, especially for subjects requiring visualization of 3D structures, processes, or abstract concepts. Hybrid AR/VR experiences could further enhance engagement and provide multi-sensory learning opportunities, aligning with pedagogical theories that emphasize experiential learning.

5. Enhanced Adaptation Strategies: Future iterations could implement hybrid personalization algorithms, combining ML-based skill prediction with rule-based pedagogical strategies. This approach could optimize learning by dynamically adjusting difficulty, hints, and feedback types based on both model predictions and educational best practices, accommodating diverse learning profiles more effectively.

6. Accessibility and Localization: Expanding support for learners with visual, auditory, or cognitive disabilities and implementing regional language options will promote inclusive education. Adherence to global accessibility standards will ensure that the platform can be effectively used by a broader range of students, contributing to equitable learning opportunities.

7. Integration of Gamification and Analytics: Future work could include advanced gamification elements, such as achievements, leaderboards, and adaptive reward systems, alongside analytics dashboards for teachers or parents. This would provide actionable insights into student performance and motivation, supporting personalized interventions and educational planning.

By pursuing these directions, the AR Math Quiz platform has the potential to evolve into a comprehensive, adaptive, and scalable educational solution. It can not only enhance engagement and learning outcomes in mathematics but also serve as a framework for immersive, personalized education across subjects and learning environments. Ultimately, this research contributes to broader goals of quality and equitable education and lays a foundation for future studies in immersive learning technologies.

References

[1] A. M. Gianni, N. Nikolakis, and N. Antoniadis. An llm based learning framework for adaptive feedback mechanisms in gamified xr. *Computers & Education: X Reality*, 7:100116, 2025.

[2] A. Marougkas, C. Troussas, A. Krouska, and C. Sgouropoulou. An adaptive virtual reality game for programming education using fuzzy cognitive maps and pedagogical models. *Smart Learning Environments*, 12(1):62, 2025.

[3] H. A. Rosyid. Adaptive serious educational games using machine learning. M.s. thesis, Univ. of Manchester, Manchester, U.K., 2018.

[4] I. Sarlis, D. Kotsifakos, and C. Douligeris. Evaluating educational game design through human–machine pair inspection: Case studies in adaptive learning environments. *Multimodal Technologies and Interaction*, 9(9):92, 2025.