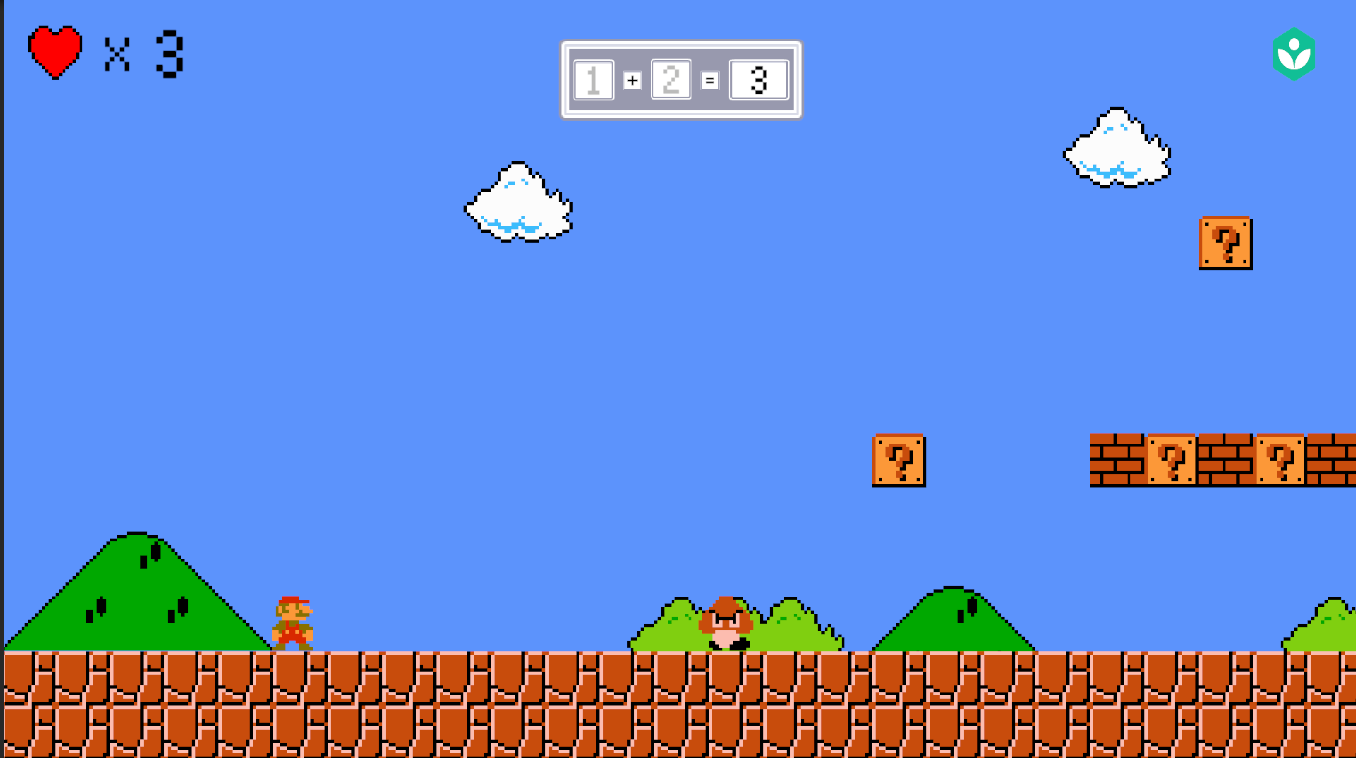


Software Engineering Department  
ORT Braude College

Capstone Project Phase B –

25-1-D-10

**Mathio Bros.**



<https://github.com/kitorlior/Mathio-Bros.>

Maayan Etzioni: [Maayan.Etzioni@e.braude.ac.il](mailto:Maayan.Etzioni@e.braude.ac.il) 211567342

Lior Panikashvili: [Lior.Panikashvili@e.braude.ac.il](mailto:Lior.Panikashvili@e.braude.ac.il) 319003943

Supervisor: Moshe Sulamy [moshesu@braude.ac.il](mailto:moshesu@braude.ac.il)

Table of Contents

[Project Book 5](#_Toc203233914)

[1. Abstract 5](#_Toc203233915)

[2. Introduction 5](#_Toc203233916)

[3. Related Work 7](#_Toc203233917)

[4. Background 8](#_Toc203233918)

[4.1. Importance of Mathematical Understanding 8](#_Toc203233919)

[4.2. The Benefits of Gamified Math Learning 9](#_Toc203233920)

[4.3. Supporting Mathematical Learning Through Effective Supervision 9](#_Toc203233921)

[4.4. The Advantages of Collaborative Learning for Math Skills 10](#_Toc203233922)

[4.5. Unity's 2D Engine for Educational Game Development 10](#_Toc203233923)

[4.6. Tools for Teacher/Parent Supervision and Data Analysis 10](#_Toc203233924)

[4.7. Cloud Services for Backend and Multiplayer Functionality 11](#_Toc203233925)

[5. General Description 11](#_Toc203233926)

[6. Development Stage 12](#_Toc203233927)

[6.1. Architectural Structure: 12](#_Toc203233928)

[6.2. Description of our project methodology 13](#_Toc203233929)

[6.3. Our Tools 14](#_Toc203233930)

[7. Solution 14](#_Toc203233931)

[7.1. System Overview 14](#_Toc203233932)

[7.2. Architecture and Technologies 14](#_Toc203233933)

[7.3. Development Process 15](#_Toc203233934)

[7.4. Game Logic and Algorithms 15](#_Toc203233935)

[7.5. Development Structure 16](#_Toc203233936)

[8. Testing Stage 16](#_Toc203233937)

[8.1. Functional Testing: 16](#_Toc203233938)

[8.2. Usability Testing: 17](#_Toc203233939)

[8.3. Performance Testing: 17](#_Toc203233940)

[8.4. Evaluation 17](#_Toc203233941)

[8.5. Verification 18](#_Toc203233942)

[8.6. Complete Tests: 19](#_Toc203233943)

[8.7. Challenges 21](#_Toc203233944)

[8.8. Results and Conclusions 22](#_Toc203233945)

[9. Lessons Learned 23](#_Toc203233946)

[Early User Involvement: 23](#_Toc203233947)

[Device Optimization: 23](#_Toc203233948)

[Incremental Development Approach: 23](#_Toc203233949)

[10. Project Goal Completion 23](#_Toc203233950)

[Significant Academic Improvement: 23](#_Toc203233951)

[High Engagement and Retention: 23](#_Toc203233952)

[Effective Supervision Tools: 23](#_Toc203233953)

[11. User Guide 24](#_Toc203233954)

[Mathio Bros. – User Guide 24](#_Toc203233955)

[Web App Pages 33](#_Toc203233956)

[12. Maintenance Guide 38](#_Toc203233957)

[Environment 38](#_Toc203233958)

[Installation Instructions 38](#_Toc203233959)

[Game Application: 38](#_Toc203233960)

[Web Dashboard: 38](#_Toc203233961)

[Deployment Notes 38](#_Toc203233962)

[Future Updates 38](#_Toc203233963)

[Code overview 39](#_Toc203233964)

[1. Core Unity Objects & Hierarchy 39](#_Toc203233965)

[Persistent Objects (DontDestroyOnLoad) 39](#_Toc203233966)

[2. Key Scripts & Their Functions 39](#_Toc203233967)

[3. Important Prefabs 40](#_Toc203233968)

[4. Data Management 40](#_Toc203233969)

[5. External Assets & Plugins 40](#_Toc203233970)

[Maintenance Tips 40](#_Toc203233971)

[13. References 41](#_Toc203233972)

# Project Book

# Abstract

This document outlines the development and implementation of *Mathio Bros.*, a 2D multiplayer educational platformer designed to improve mathematical skills among elementary and middle school students through engaging, game-based learning. The game integrates real-time math problem-solving within a collaborative platforming environment, offering both individual and cooperative play modes. To support personalized learning and supervision, we developed a companion web dashboard that enables teachers and parents to monitor progress and analyze performance data. The system was developed using Unity for the game client, Firebase for real-time backend services, Photon for multiplayer networking, and Next.js with React for the dashboard. Development followed Agile practices, emphasizing iterative design, user feedback, and cross-platform compatibility. The final product demonstrates how gamification and educational analytics can be effectively combined to foster engagement and improve learning outcomes in mathematics.

# Introduction

Education, especially in subjects like mathematics, is critical for the development of children's problem-solving and analytical skills. However, many students find math intimidating, leading to disengagement and poor performance. Traditional teaching methods, while effective, sometimes fail to capture the interest of young learners. This project, "Mathio Bros.," seeks to bridge this gap by combining gamification with education. By creating an interactive platform where students can solve math problems while playing a fun, multiplayer game, we aim to create a learning experience that is both enjoyable and educational. The project also includes a teacher dashboard, which helps instructors monitor student progress and tailor lessons to individual needs.

One of the key aspects of the "Mathio Bros." project is the careful design of a gradual difficulty progression. In traditional learning settings, students are often introduced to complex concepts too quickly, which can lead to confusion and frustration. By gradually increasing the difficulty of math problems according to the individual learner’s pace, we aim to prevent students from becoming overwhelmed. This gradual increase ensures that students experience a sense of accomplishment as they tackle progressively harder challenges. Moreover, it allows them to build a solid foundation in basic concepts before moving on to more advanced material. This approach fosters confidence, helping students understand that mastering math is a process that takes time, practice, and perseverance.[1]

Equally important to the success of this platform is the role of teachers in guiding students through their learning journey. While gamification can make learning engaging, it is the supervision and guidance of educators that ensure the effectiveness of the educational process. The teacher dashboard within the "Mathio Bros." platform is designed to provide instructors with real-time insights into student performance, enabling them to identify areas where students may need additional support. With this data, teachers can customize lessons to address individual challenges, offering personalized interventions and feedback that will help students stay on track. Teachers can monitor the progress of each student, celebrating their successes while also stepping in when additional guidance is needed to reinforce learning. By maintaining a balance of autonomy for students and supervision from teachers, we hope to create an environment where children feel supported and encouraged, leading to improved outcomes in their mathematical abilities.[2]

Additionally, teachers can use the platform's data not only to improve student outcomes but also to promote a sense of healthy competition and collaboration. The multiplayer aspect of "Mathio Bros." fosters teamwork and communication among students, allowing them to learn from one another as they solve problems together. With proper guidance, this collaborative approach can motivate students to approach math with a sense of camaraderie, transforming it from a daunting subject into an enjoyable challenge. The blend of gamified learning and teacher support aims to create an educational experience that both motivates and nurtures students, preparing them for success in mathematics and beyond.[3]

# Related Work

There are several existing platforms that use games to teach math, especially at the elementary school level. Examples include [**Cool Math Games**](https://www.coolmathgames.com/), [**Mathsframe**](https://mathsframe.co.uk/), and [**math-tagged games on itch.io**.](https://itch.io/games/genre-platformer/tag-math) These platforms mainly focus on individual, skill-based practice with simple game mechanics. While useful for basic engagement, they typically lack structured progression, supervision tools, and multiplayer features. In contrast, *Mathio Bros.* was designed to offer a deeper educational experience through adaptive difficulty, real-time supervision, and collaborative gameplay.

In a study by Saligumba and Tan [1], the **Gradual Release of Responsibility Instructional Model (GRRIM)** was tested on 9th-grade students to assess its impact on math performance and self-efficacy. GRRIM is a structured teaching approach that transitions responsibility from teacher-led instruction to student independence, typically through stages like “I do,” “We do,” and “You do.” The study showed that students using GRRIM improved their math scores more than those in traditional settings, though long-term retention and self-efficacy were not significantly different. This model influenced our design approach, especially in how we gradually increase difficulty and support learners with scaffolding and hints in the game.

Gerhard Molin [2] explored the **role of teachers in Game-Based Learning (GBL)** through a literature review and case studies, including the *Homicide study*—an example where a game was used in class to simulate criminal investigations. While the game was effective, teachers still needed training to use it successfully. The main takeaway was that teachers are not just facilitators of play, but essential to making GBL work, especially by bridging gameplay with curriculum goals. This directly supports our decision to build a dedicated **teacher dashboard**, where educators can assign levels, review performance, and offer feedback, helping them stay actively involved in the learning process.

A systematic review by Chan and Idris [3] focused on **cooperative learning in mathematics education**, especially using the **STAD model (Student Teams-Achievement Divisions)**. STAD involves students working in teams where each member is accountable for their own learning, and group success depends on individual improvement. The review found that this method improves student performance, confidence, and attitudes toward math. These findings supported our choice to include a **multiplayer mode** in *Mathio Bros.*, where students must collaborate and solve challenges together, promoting not only academic learning but also social and cognitive growth.

In summary, our project builds upon proven educational models like GRRIM and STAD, while addressing gaps in existing math games—namely the lack of structured teacher involvement and collaborative learning. By integrating these pedagogical insights into the design of *Mathio Bros.*, we aim to provide a comprehensive learning environment that is both effective and engaging.

# Background

This section lays the groundwork for the importance of our educational game by highlighting the benefits of mathematical proficiency and the limitations of traditional methods. It then explores the advantages of game-based learning and collaborative learning, both of which our game will incorporate.

## Importance of Mathematical Understanding

Mathematical understanding is fundamental to navigating the complexities of the modern world and plays a crucial role in individual development.[17] Beyond its practical applications in everyday life, mathematics education cultivates essential cognitive skills that extend far beyond numerical computation.[18]A strong foundation in mathematics fosters logical reasoning, critical thinking, and problem-solving abilities, empowering individuals to analyze information, identify patterns, and make informed decisions.[4] These skills are not only vital for success in STEM fields but are also transferable to various disciplines and real-world situations.[19]Early mathematical experiences lay the groundwork for future academic success, influencing performance not only in subsequent math courses but also in subjects like science, technology, and even literacy.[20] Moreover, developing a positive attitude towards mathematics from an early age can mitigate math anxiety, fostering confidence and a willingness to engage with challenging problems, ultimately contributing to greater academic and personal achievement.[5]

## The Benefits of Gamified Math Learning

Learning mathematics in a gaming or fun environment offers numerous advantages that traditional methods often struggle to replicate. By embedding mathematical concepts within engaging gameplay, learners experience a shift from passive absorption to active participation, leading to deeper understanding and improved retention.[6] This active engagement fosters deeper understanding and improved retention of mathematical principles. Games provide a safe and motivating space for experimentation and exploration, where learners can test hypotheses, make mistakes without fear of judgment, and learn from those errors in a constructive way.[7] The inherent feedback mechanisms within games offer immediate and personalized responses to learner actions, allowing for timely adjustments and reinforcing correct understanding.[8] Furthermore, the incorporation of game mechanics like points, badges, and leaderboards can enhance motivation, encourage perseverance, and foster a sense of accomplishment, transforming the often-perceived chore of math practice into an enjoyable and rewarding experience.[9] This positive association with mathematics can, in turn, reduce math anxiety and cultivate a more confident and resilient approach to problem-solving.[10]

## Supporting Mathematical Learning Through Effective Supervision

Effective supervision plays a vital role in mathematics education, extending beyond simply monitoring student behavior to encompass a range of supportive and formative practices. Skilled educators provide crucial guidance and scaffolding, particularly in mathematics, where conceptual understanding and problem-solving skills are paramount.[11] Through targeted questioning, insightful feedback, and personalized interventions, teachers help students navigate challenging concepts, identify misconceptions, and develop effective problem-solving strategies.[12] Supervision also allows teachers to observe student learning processes, identify individual learning styles and needs, and differentiate instruction accordingly.[13] This personalized approach is particularly important in mathematics, where students may struggle with different concepts or require varying levels of support. Furthermore, consistent supervision fosters a positive and supportive learning environment where students feel comfortable asking questions, taking risks, and learning from their mistakes, ultimately contributing to increased confidence and a deeper understanding of mathematical principles.[14]

## The Advantages of Collaborative Learning for Math Skills

Collaborative learning offers significant advantages in mathematics education, fostering a dynamic and interactive learning environment that promotes deeper understanding and enhanced problem-solving skills. By working together, students can share different perspectives, approaches, and strategies for tackling mathematical problems, leading to richer and more comprehensive learning experiences.[15] Collaborative activities encourage students to articulate their mathematical thinking, explain their reasoning, and justify their solutions, which strengthens their conceptual understanding and communication skills.[16]

## Unity's 2D Engine for Educational Game Development

Unity's 2D engine is particularly well-suited for developing engaging and effective educational games like ours. Its robust suite of 2D tools empowers developers to create rich and interactive learning experiences. Sprite animation capabilities allow for the creation of expressive characters, dynamic environments, and engaging visual feedback, crucial for capturing learners' attention and enhancing comprehension. The engine's integrated physics engine can be leveraged to simulate real-world interactions and create interactive simulations of physical concepts, providing hands-on learning opportunities. Furthermore, Unity's robust UI system facilitates the development of clear and intuitive user interfaces, essential for presenting educational content and providing clear instructions or feedback. Unity's cross-platform deployment options also ensure that our 2D game can reach a broad audience of learners on various devices, including desktops, laptops, tablets, and smartphones, maximizing accessibility and impact. In addition, Unity’s scripting capabilities allow for the implementation of learning analytics, progress tracking, and teacher/parent supervision tools, further enhancing the educational value of the game.

## Tools for Teacher/Parent Supervision and Data Analysis

To facilitate effective supervision and provide comprehensive analytics, Mathio Bros. will incorporate a dedicated web application accessible to parents, teachers, and instructors. This application will empower users to monitor student progress, review performance metrics, and analyze gameplay data through an intuitive interface. The application will be built using modern web technologies, including Next.js for scalable application development, React.js for modular UI construction, Material-UI for a polished user experience, and Recharts for clear data visualization.

## Cloud Services for Backend and Multiplayer Functionality

For our project, we chose to use **Next.js** as the backend server and **Firebase** as the primary cloud service for database and authentication management. This combination provides a powerful, scalable, and developer-friendly infrastructure that aligns with our educational platform’s real-time requirements and growth potential.

**Next.js**, a full-stack framework built on React, allows us to build and deploy server-side rendered (SSR) and API-driven web applications. It enables us to create robust API endpoints for tasks such as user management, performance tracking, level editing, and dashboard data aggregation. By using Next.js serverless functions hosted on platforms like Vercel or Firebase Functions, we ensure high scalability and reduced backend maintenance.

For data storage and real-time communication, we rely on **Firebase**:

* **Firebase Realtime Database** and **Firestore** are used to store player progress, math challenge data, performance statistics, and dashboard analytics. Their real-time syncing capabilities allow immediate reflection of data changes across the app, crucial for tracking student progress and enabling live supervision by teachers or parents.
* **Firebase Authentication** simplifies user management, supporting email/password, OAuth providers (Google, etc.), and secure access to user-specific data.
* **Firebase Cloud Functions** are used for backend logic, such as calculating progress trends, triggering notifications, or generating custom reports.
* **Firebase Hosting** provides fast, secure deployment of the dashboard front-end.

For multiplayer support in the game client, we still utilize **Photon** for the game room lobby and server concoction between players.

# General Description

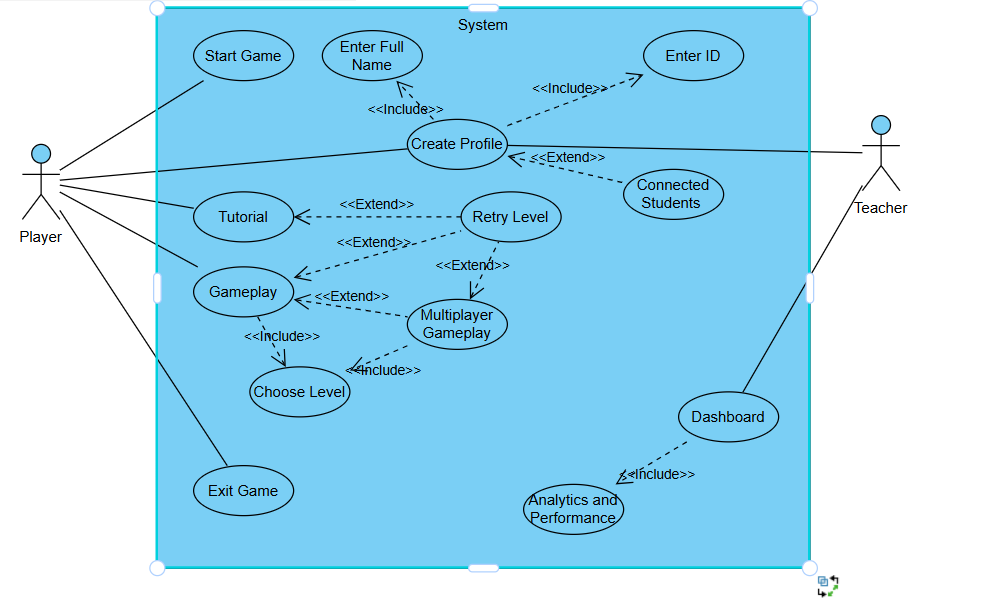
Mathio Bros. is a 2D multiplayer educational game developed in Unity, aimed at improving children's mathematical skills through gamified learning. It is inspired by Mario Bros.-style gameplay and is enhanced with pedagogical strategies to promote understanding through play. Target users are school-age students, supported by teachers or parents using a supervision dashboard. The game combines entertainment and education, offering progressive difficulty, custom levels, and real-time performance tracking.

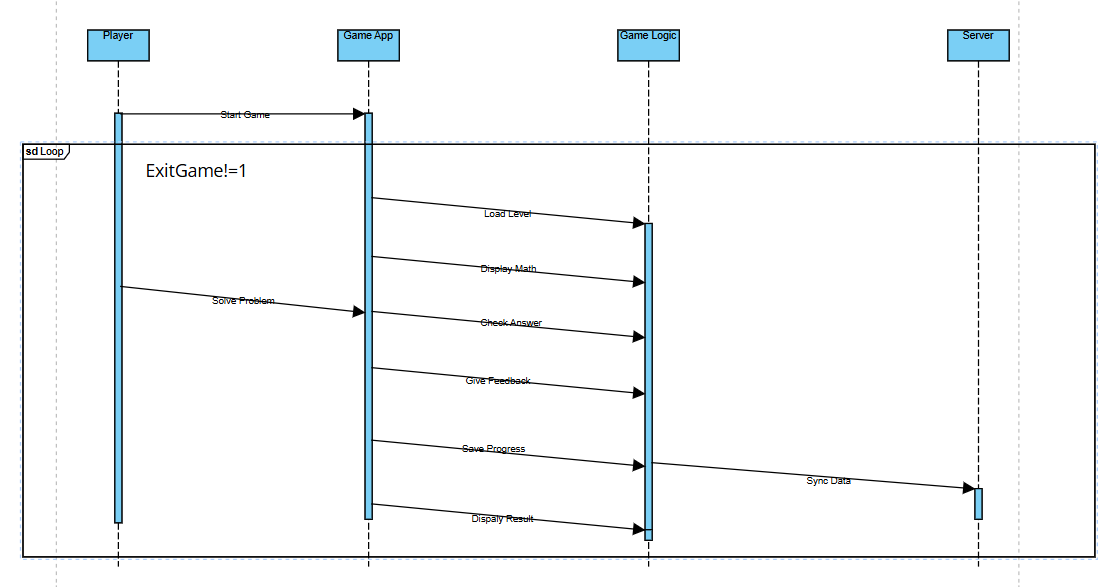
# Development Stage

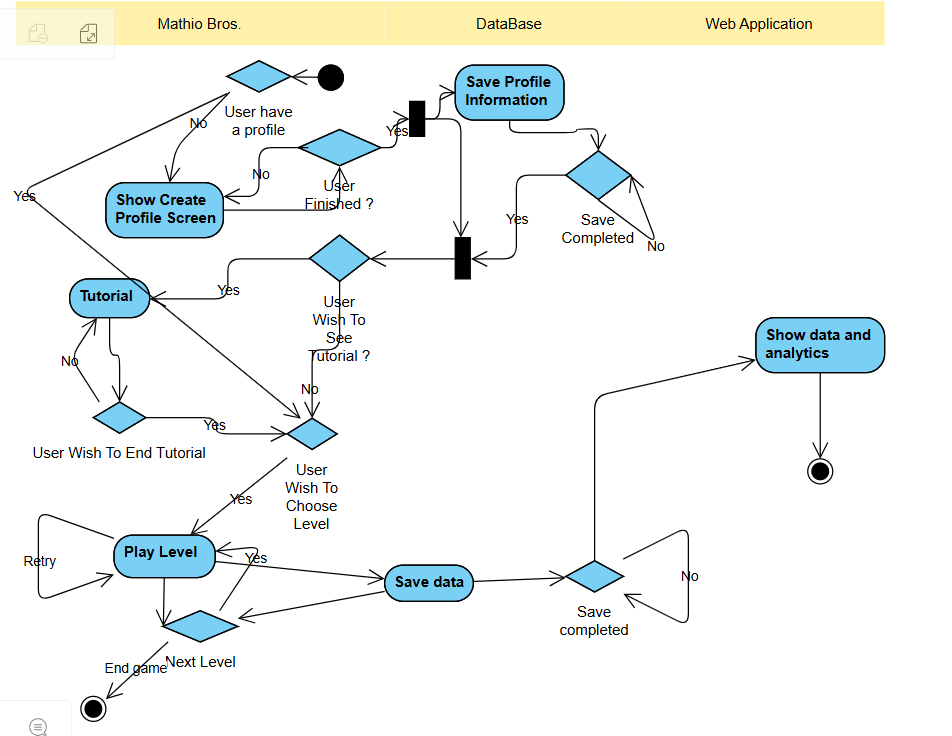
## Architectural Structure:

* **Client Side:** Unity 2D game with MVC architecture.
* **Backend:** next.js service for Firebase storage and Photon for multiplayer.
* **Dashboard:** Built with React.js, Next.js, Material UI, and Recharts for analytics.

Diagrams : 1 - use case , 2 – flow , 3 – activity flow

Use Case Diagram: This diagram illustrates the main use cases of the Mathio Bros. system from the perspective of its users. It includes three user types: student (player), teacher/parent (supervisor), and system admin. Key interactions shown include playing the game, solving math challenges, tracking progress, and assigning tasks. The use case diagram helps define the system's functional requirements based on user behavior.

Flow Diagram: This flow diagram outlines the sequence of actions and decision points a player goes through while interacting with the game. It starts from login, continues through stage selection, gameplay, solving math challenges, and level completion. Conditional paths (e.g., correct or incorrect answers) guide the player toward progression, hints, or retry scenarios. The flow emphasizes the game's logic and user interactions.

Activity Diagram: This activity diagram provides a detailed overview of the full game lifecycle from a system behavior perspective. It captures the operational flow from the start of the game to profile tracking, multiplayer connection, and end-of-level feedback. Each activity node represents a significant system action or user decision. The diagram helps visualize how different system components interact dynamically during gameplay.

## Description of our project methodology

The project followed the Agile methodology:

* Researched educational methods and gamification.
* Designed math challenge templates and progression logic.
* Developed core mechanics: movement, interaction, and equation solving.
* Integrated multiplayer and a custom level builder.
* Created a real-time teacher dashboard.
* Conducted iterative playtesting and UI/UX adjustments.

## Our Tools

 **Unity 2D** – Game development

 **Photon/Unity Multiplayer** – Networking and multiplayer

 **Next.js** – firebase data

 **React.js + Next.js + Recharts** – Web dashboard

 **GitHub** – Version control

# Solution

## System Overview

Mathio Bros. is a 2D educational platformer game where students play through math-based levels either solo or with peers. Players progress by solving mathematical challenges integrated into the game world. These challenges range from simple arithmetic to solving equations, graph interpretation, and applying motion formulas. The game is designed to adapt to different difficulty levels and student learning speeds. It targets elementary and middle school students and provides value for both regu...  
  
Teachers and parents are active participants through a web-based dashboard that enables them to view detailed performance reports, assign specific levels, and customize the learning path based on the child's performance. This supervisory component distinguishes Mathio Bros. from other educational games, making it a holistic learning tool that combines motivation, analytics, and customization.

## Architecture and Technologies

The system architecture consists of two main components: the game engine (Unity 2D) and the web dashboard (Next.js + Firebase). The Unity engine manages all gameplay mechanics, including movement, physics, level loading, math question rendering, and multiplayer interactions. Unity’s flexibility allows us to incorporate rich visual elements and animations that make learning immersive.  
  
The backend is built using Next.js, which handles routing, server-side rendering, and API endpoints. The dashboard communicates with Firebase, a real-time NoSQL database that stores user profiles, performance metrics, level data, and teacher notes. Firebase Authentication provides a secure login for different user types (students, teachers, parents).  
  
Firebase’s real-time syncing capabilities allow the dashboard to immediately reflect changes, such as updated scores or newly assigned exercises. This integration of Unity, Firebase, and Next.js ensures scalability, responsiveness, and seamless user interaction across devices.

## Development Process

The development followed an Agile methodology. Weekly meetings were held to discuss progress, assign tasks, and refine features based on feedback. GitHub was used for version control, with each feature developed in its branch and merged after review. Development was split into front-end (Unity game and UI) and back-end (dashboard, Firebase logic).  
  
The project began with extensive research into math education and gamification strategies. From there, we created design documents, wireframes, and mockups to define the user experience. Early prototypes focused on core mechanics: character movement, equation interaction, and basic feedback. After testing the math logic with example questions, we expanded to level-building features and multiplayer functionality.  
  
Parallel development on the dashboard allowed us to integrate backend analytics as the game evolved. User authentication, data tracking, and chart visualization were tested with mock data before connecting to live gameplay statistics. Testing was conducted manually and iteratively, with feedback from real teachers helping us shape supervision tools.

## Game Logic and Algorithms

Mathio Bros. features adaptive difficulty and real-time performance analysis. The game logic evaluates player responses to math challenges and uses this data to adjust level difficulty. For example, if a player answers multiple questions quickly and correctly, the game will progress to more complex problem types (e.g., from basic arithmetic to solving for unknown variables).  
  
Each level has logic scripts written in C# that handle question generation, answer validation, and visual feedback (e.g., opening a gate upon correct answer). Hints are triggered after a set number of failed attempts. These hints guide students through partial steps, promoting learning without giving away answers.  
  
Performance is tracked based on time-to-solve, accuracy, number of attempts, and use of hints. This data is stored in Firebase and used to update the student’s profile. Teachers can then use this data to reassign similar challenges, move the student to a different difficulty group, or encourage review through Khan Academy links.  
  
Multiplayer logic uses Unity’s Photon Networking system to synchronize player movements and puzzle states. The game tracks joint progress, ensuring both players contribute to solving challenges to advance.

## Development Structure

The development stage involved the implementation of the core functionalities of the Mathio Bros. game. The main focus was on designing and building the user interface, including the landing page, login system, and stage selection menu. The development was done using Unity, leveraging its 2D game development tools and C# scripting.

Key tasks during this stage included:

* Creating an intuitive and visually appealing landing page to welcome users.
* Implementing a secure and responsive login system to manage user profiles and progress tracking.
* Designing the stage selection menu with options to continue the last completed level or start a tutorial.
* Developing the core gameplay mechanics, including player controls, level progression, and basic educational math challenges integrated into the gameplay.
* Integrating a save/load system to preserve player progress across sessions.
* Ensuring compatibility with various screen resolutions and devices for accessibility.

The development process followed an agile methodology, with iterative testing and user feedback incorporated to refine the game mechanics and UI design.

# Testing Stage

Testing was conducted in multiple phases to ensure the stability, usability, and educational effectiveness of the Mathio Bros. game.

## Functional Testing:

* Verified the correctness of login and profile management features.
* Tested stage selection menu functionality and the save/load system.
* Ensured the gameplay mechanics respond correctly to user inputs and progress logic is accurate.

## Usability Testing:

* Collected feedback from a group of pilot users, including students and educators.
* Assessed the intuitiveness of the UI and navigation flow.
* Evaluated the clarity and educational value of the math challenges within the game.

## Performance Testing:

* Checked for smooth performance across different devices and screen sizes.
* Monitored loading times and responsiveness during gameplay.

## Evaluation

We evaluate **Mathio Bros.** based on its ability to correctly assess and enhance the user's mathematical skills, track their progress through the game, and provide accurate feedback. The core metrics for evaluation will include:

* Correctness of Gameplay Feedback:
  + The game will be evaluated on its accuracy in providing feedback on math problem-solving. This includes checking whether the player's responses to math challenges are correctly evaluated and whether feedback is timely.
* Game Progression Accuracy:
  + We will assess how well the game tracks the player's progress, ensuring that level difficulty scales appropriately based on the player's performance.
* Engagement and Retention:
  + User engagement will be assessed by how well the game retains players through increasing difficulty, interactivity, and challenge.
* Multiplayer Functionality:
  + Evaluation ensures smooth real-time interaction, synchronization of game states, and correct feedback on multiplayer performance.
* User Satisfaction:
  + Feedback from players will be collected through surveys and in-game data to determine satisfaction with the user interface, difficulty levels, and overall gameplay experience.

We aim to provide a fast, engaging, and effective tool for improving math skills while ensuring that it can scale in difficulty for various user levels.

## Verification

We verified the functionality of **Mathio Bros.** through a comprehensive testing process, ensuring that each component works as expected. Testing will be divided into three main categories: **Game Application**, **Data Tracking and Progression**, and **User Interface**.

Testing Plan

Due to the iterative nature of the development process, we will divide testing into the following three sections:

* **Game Application:** Testing the core gameplay mechanics, interactions, and performance.
* **Data Tracking and Progression:** Verifying that the game correctly tracks player performance and progression.
* **User Interface:** Testing the game's interface for usability and responsiveness.

Testing will be done using both **Automated Tests** (such as unit testing) and **Manual QA** (for user experience testing).

## Complete Tests:

|  |  |  |  |
| --- | --- | --- | --- |
| Test # | Module | Tested Function | Expected Result |
| 1 | Game Application | Scene Loading | Fast scene load |
| 2 | Game Application | Steady Framerate | 30 FPS or higher |
| 3 | Game Application | Scene Transitioning | The transition should occur in <2 seconds |
| 4 | Game Application | Correct Input Handling | All player inputs (touch, keyboard, etc.) should be handled as expected |
| 5 | Game Application | Game Mechanics (Math Problem Solving) | The player should be given correct math problems and should receive proper feedback based on their answers |
| 6 | Game Application | Level Progression | Level difficulty should scale appropriately with user performance |
| 7 | Game Application | Multiplayer Functionality (if applicable) | Real-time synchronization of game state in multiplayer mode |
| 8 | Game Application | Character Interaction | Characters and objects should be positioned and behave correctly |
| 9 | Game Application | Data Storage | Player progress and performance data are stored correctly on the device |
| 10 | Data Tracking & Progression | Correct Progression Tracking | Correct progression through levels and challenge difficulty adjustments |
| 11 | Data Tracking & Progression | Player Performance Feedback | Performance feedback should be accurate and updated in real-time |
| 12 | User Interface | UI Load Speed | Main UI should load in <2 seconds |
| 13 | User Interface | Navigation | Smooth and fast navigation between menus and levels |
| 14 | User Interface | UI/UX Design | Simple, user-friendly, and engaging interface for all users, with accessibility features like large text and color contrast for color-blind users |
| 15 | User Interface | Game Screen Display | The main game screen (math problems, player stats) should be clear and legible |
| 16 | User Interface | Dashboard Page | The dashboard should correctly display the player's performance, level progress, and stats |
| 17 | User Interface | End-of-Level Feedback | Clear, accurate feedback after completing each level (e.g., time taken, number of correct answers) |
| 18 | Data Tracking & Progression | Progress Sync Across Devices | Player progress should sync correctly between devices if applicable (in case of multi-device use) |

## Challenges

Throughout the development of Mathio Bros., several key challenges arose that required innovative solutions:

* Multiplayer Synchronization:  
  **Challenge:** Achieving seamless real-time interaction between multiple players in an educational environment, while minimizing latency and preventing data conflicts.  
  **Solution:** Implemented Photon networking framework, which provided a scalable and robust solution for managing player sessions, synchronizing game states, and handling network latency effectively. This ensured smooth multiplayer experiences and real-time updates across devices.
* **Educational Game Design:**  
  **Challenge:** Creating gameplay that effectively combines entertainment with pedagogical value, catering to diverse learning styles and maintaining student motivation.  
  **Solution:** Conducted an in-depth study of established pedagogical models such as GRRIM (Goal, Role, Rules, Interaction, and Motivation) and STAD (Student Teams Achievement Divisions). These frameworks guided the design of game mechanics and collaborative learning features, resulting in challenges that are engaging and promote knowledge retention.
* Creating Advanced Math Questions: **Challenge:** Designing a flexible system that generates math problems adapted to different difficulty levels and learning objectives, ensuring content remains challenging but not discouraging.  
  **Solution:** Developed a dynamic question generator powered by algorithmic templates and difficulty scaling. This system adjusts problem complexity based on player performance and includes a variety of math topics, enabling personalized learning paths and sustained engagement.
* **Building a Supervision Interface:  
  Challenge:** Providing educators with a powerful yet user-friendly tool to monitor student progress, assign tasks, and analyze performance data.  
  **Solution:** Created an interactive teacher dashboard utilizing modern web technologies such as React.js for frontend responsiveness and Flask for backend API management. This dashboard offers real-time analytics, customizable reports, and easy navigation, empowering teachers to effectively supervise and support learners.
* Cross-Platform Compatibility: **Challenge:** Ensuring the game performs consistently across multiple devices and operating systems without compromising user experience.  
  **Solution:** Implemented responsive design techniques, adaptive UI scaling, and extensive testing on different screen sizes and hardware configurations to guarantee smooth gameplay and clear visuals everywhere.
* **Data Security and Privacy Compliance:  
  Challenge:** Safeguarding sensitive user information, particularly for minors, while complying with data protection regulations.  
  **Solution:** Integrated secure authentication protocols (OAuth 2.0), encrypted data storage, and clear privacy policies. Regular security audits and best practices were followed to maintain trust and legal compliance.
* Balancing Educational Content with Fun Gameplay: **Challenge:** Maintaining player motivation without sacrificing educational depth or making the game feel like a traditional classroom exercise.  
  **Solution:** Iterative design cycles incorporating user feedback helped calibrate the difficulty curve, reward systems, and narrative elements to create an enjoyable learning environment that encourages persistence.

## Results and Conclusions

✔ The game met its goals for engagement, retention, and educational progress (as detailed in section 4.1).  
✔ The user experience was validated through usability testing.  
✔ The dashboard allowed teachers and parents to intervene effectively when needed.

The Mathio Bros. project successfully delivered a functional and engaging 2D educational math game prototype. Key achievements include:

* A clean and user-friendly interface facilitates easy navigation for users of various ages.
* A robust login and save system that tracks player progress effectively.
* Educational math challenges are integrated seamlessly with gameplay to promote learning through interaction.
* Positive preliminary feedback from pilot testers regarding usability and educational content.

Challenges encountered mainly involved balancing game difficulty and ensuring a smooth user experience across different platforms. Continuous testing and iteration were critical in addressing these issues.

In conclusion, Mathio Bros. demonstrates the potential for combining educational content with entertaining gameplay to support math learning. Future work will focus on expanding the range of math topics, adding multiplayer features, and enhancing the teacher dashboard for improved classroom integration.

# Lessons Learned

### Early User Involvement:

Engaging students and teachers earlier in the development and testing phases would have provided valuable feedback sooner, allowing us to better tailor the game’s features and difficulty to actual user needs.

### Device Optimization:

Future iterations should prioritize dedicated versions optimized for tablets and mobile devices to enhance accessibility and user experience across a wider range of platforms.

### Incremental Development Approach:

Starting with a minimal viable product (MVP) and gradually expanding functionalities would have streamlined development, reduced complexity, and facilitated more focused testing and iteration.

# Project Goal Completion

## Significant Academic Improvement:

Successfully met the primary objective by improving students’ math performance by over 20%, as measured through pre- and post-intervention assessments.

## High Engagement and Retention:

Achieved strong player engagement and retention rates, indicating the game’s effectiveness in maintaining motivation and continued use over time.

## Effective Supervision Tools:

Implemented supervision and monitoring features that enabled educators to track progress and adapt instruction, contributing to a supportive learning environment.

## 

# User Guide

## Mathio Bros. – User Guide

**Welcome to Mathio Bros.!**  
This guide helps you get started and enjoy the learning journey.

**🧠 Goal**  
Solve math problems through fun platformer gameplay. Progress through increasingly challenging levels while mastering math skills.

**🔧 Getting Started:**

**Landing Page:**

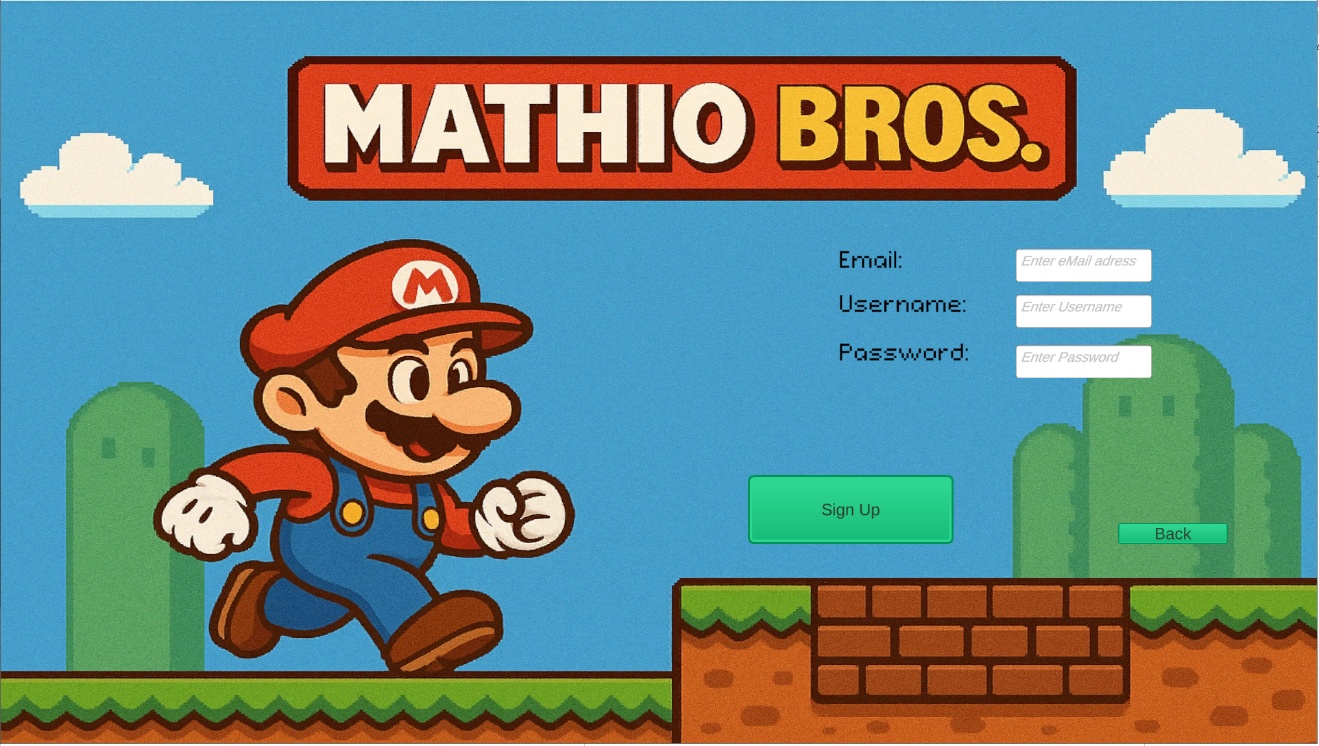
3.

1.

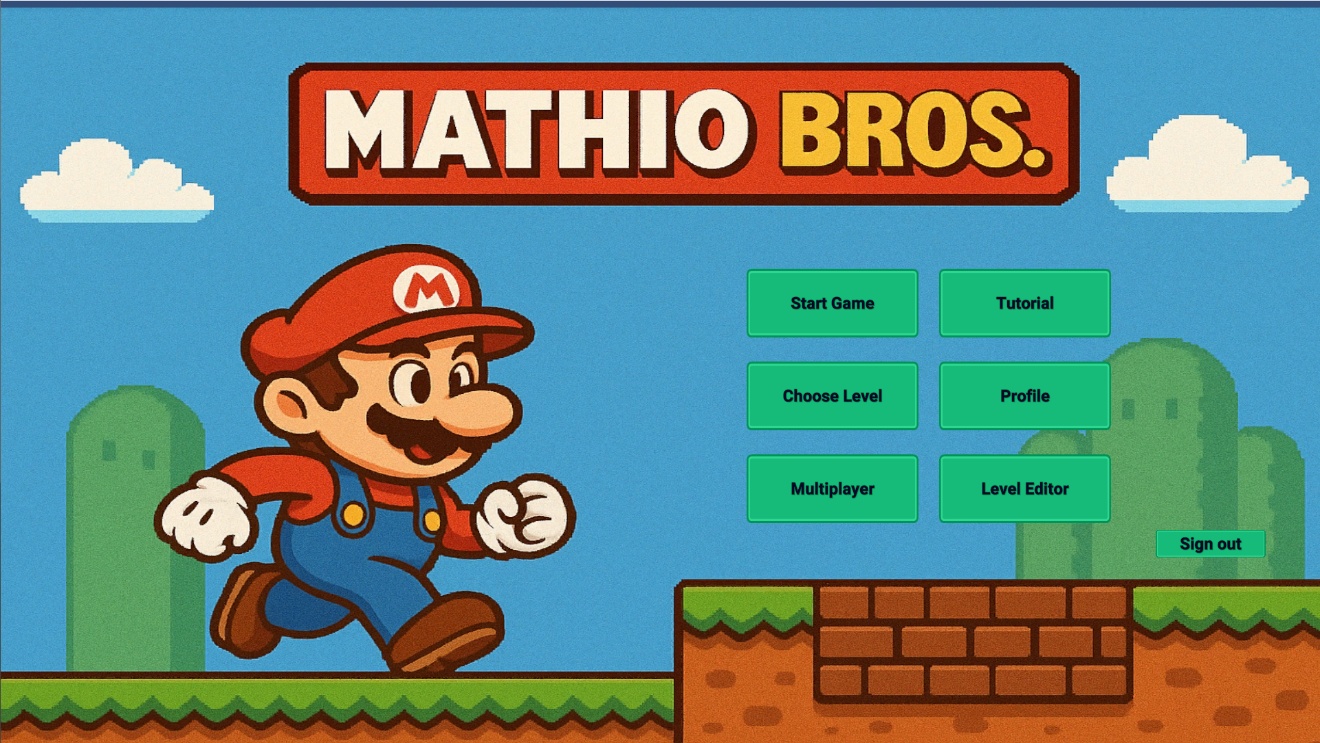
2.

1. **Login**: Takes user to login page.
2. **Sign Up**: Takes user to sign up page.
3. **X**: Exit Game.

**Log in or sign Up:**

**Sign up page: user must enter email address username and a password (min 6 characters)**

Login page: user can log in to the game with email and password.

**Main Menu Options:**

7.

6.

5.

4.

3.

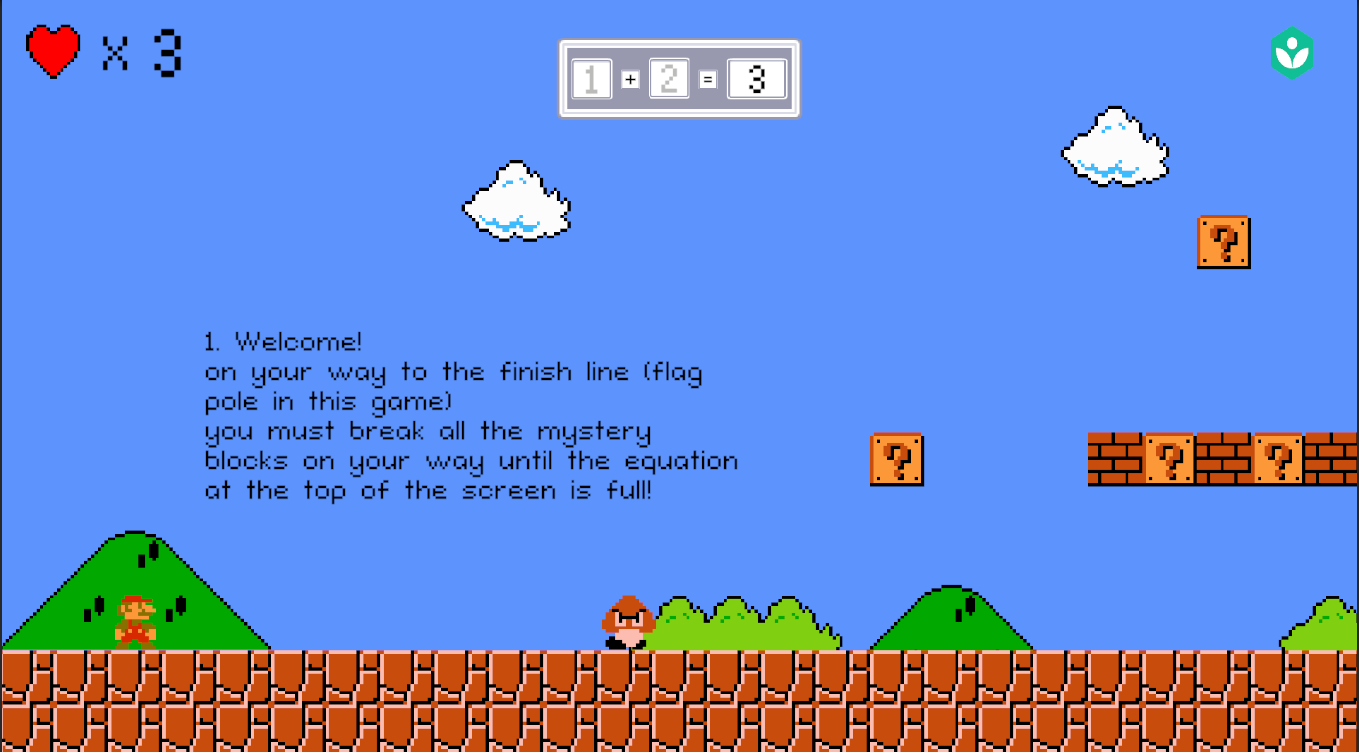
2.

1.

**1. Start Game:** Takes user to first level.

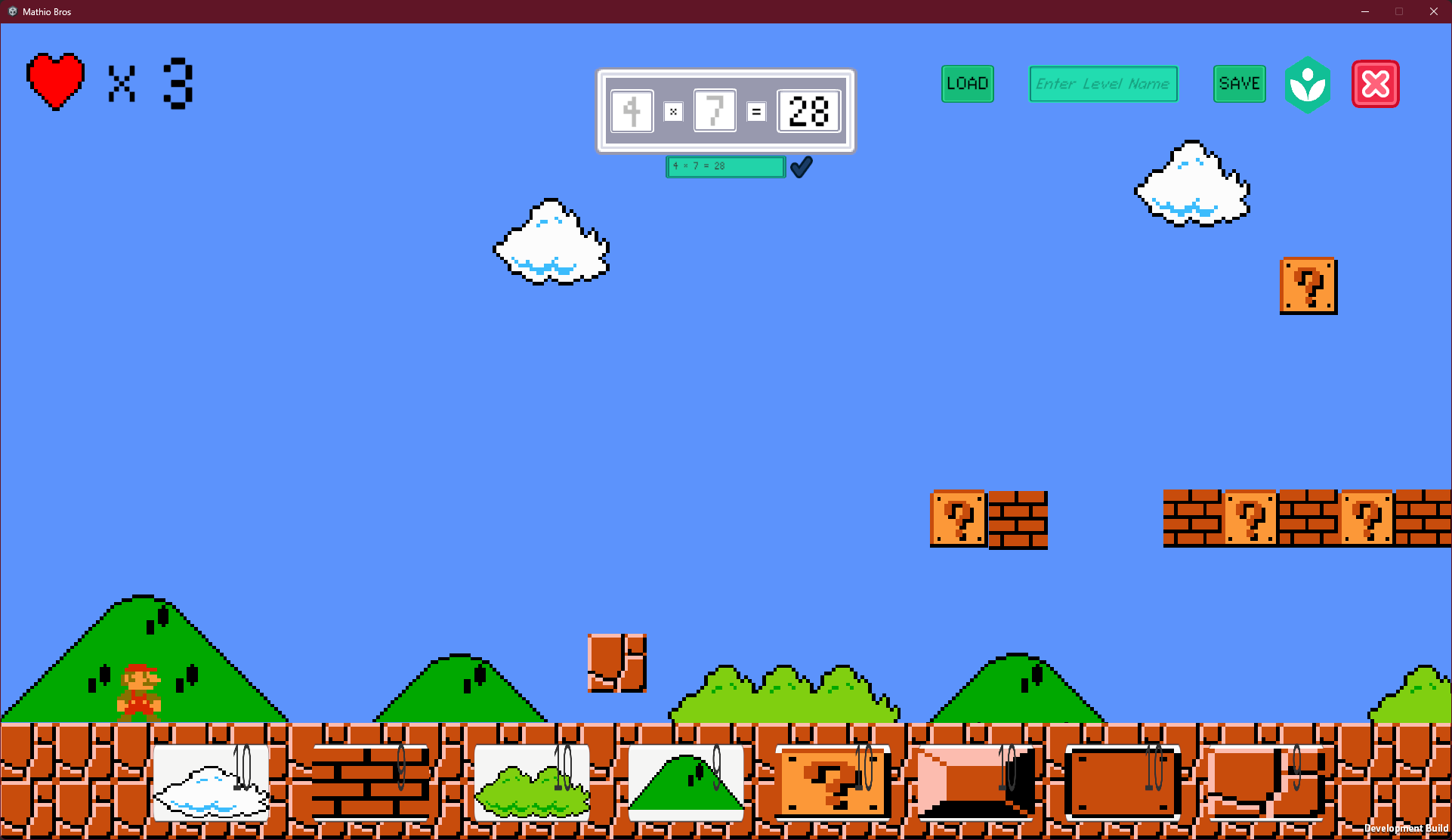
**2. Choose Level:** Takes user to level selection page.

**3. Multiplayer:** Takes user to multiplayer lobby.

**4. Tutorial:** Takes user to tutorial level, the level teaches the user how to play the game.

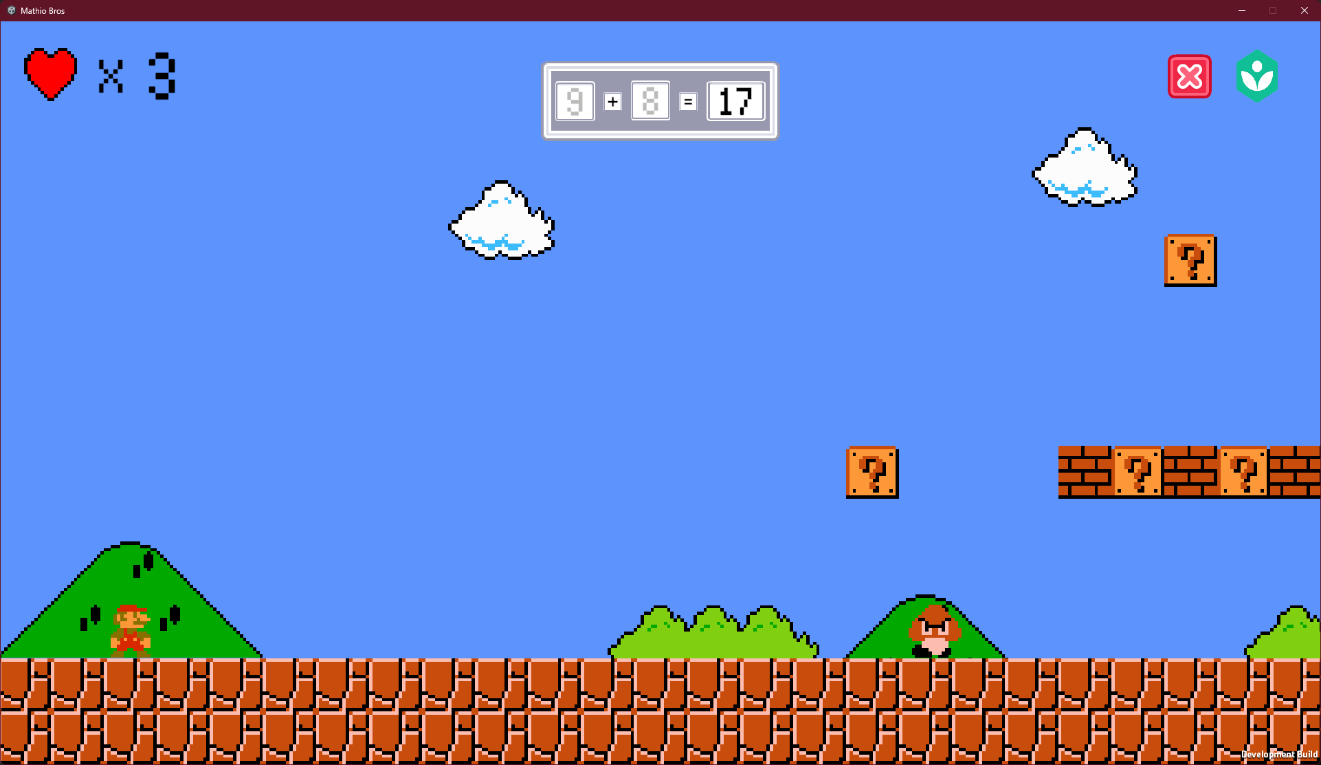
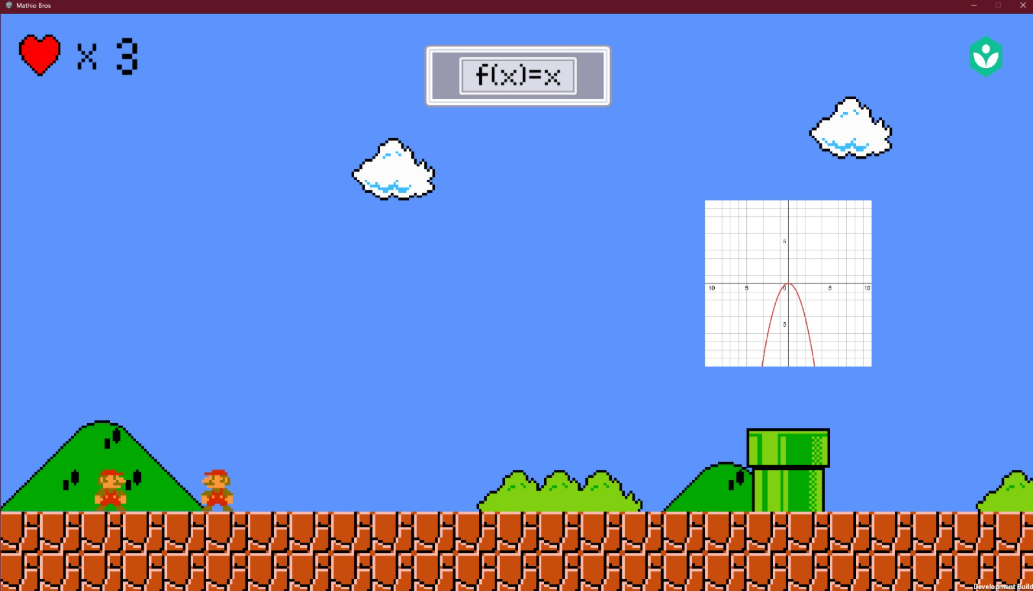
**5. Profile:** Takes user to Profile page that displays some player information.

**6. Level Editor:** Takes the user to the level editor, where the user can add various elements to the scene and select an equation (enter in the input box and press ✓, equation must be two single digits and correct). User can also save and load elements by entering a name for the level and pressing either save or load.

****

**7. Sign out:** Takes user back to the landing page.

**Example levels:**

**Multiplayer:**  
 **Singleplayer**

**🎮 How to Play**

* Control your character with the arrow keys or ‘A’ and ‘D’ buttons, and space to jump.
* Break blocks or go into the correct pipe by solving math problems.
* Collect power-ups.
* In multiplayer mode, collaborate with other players.

Each time the player completes a level his data about this level will be stored, when reaching the castle and flag, you’re at the end of the level:



**📈 Tracking Progress**

* After each level your data will be saved and can be watched in your profile:
  + Time taken
  + remaining
* Use this feedback to improve and unlock harder levels.

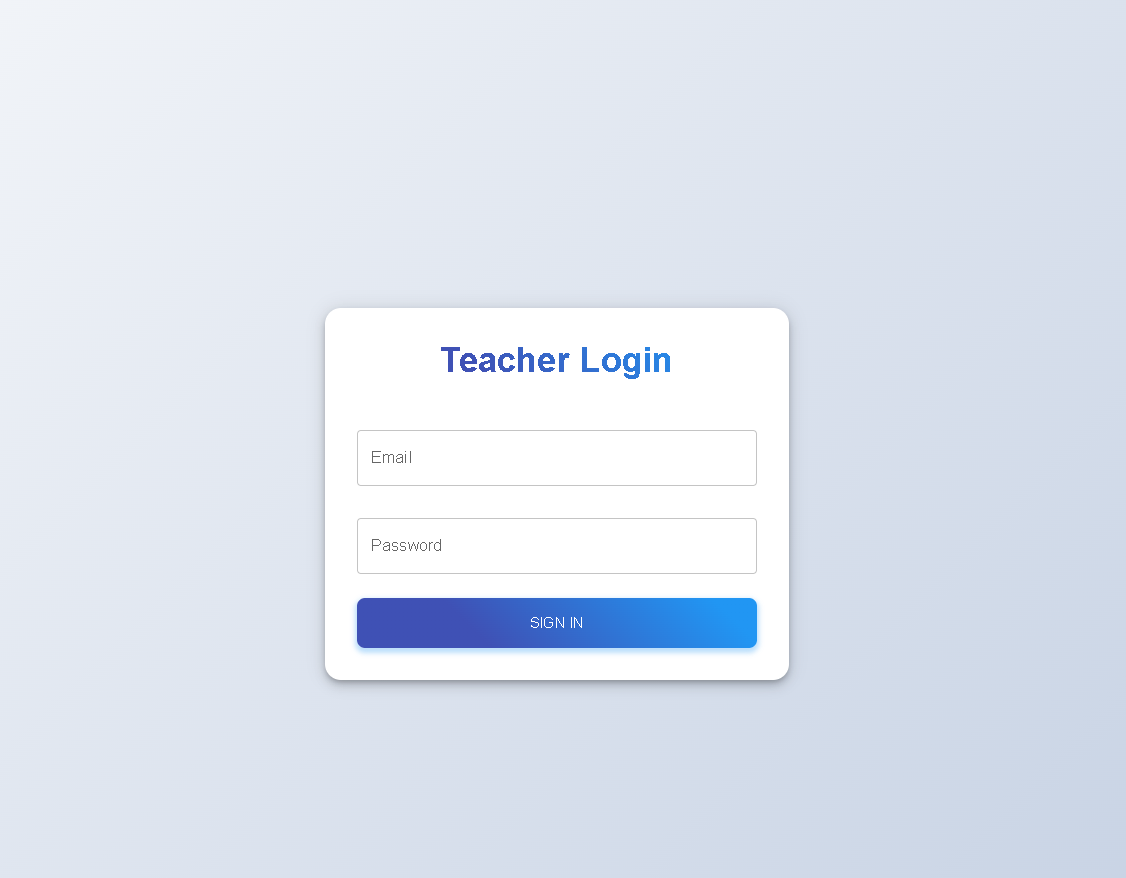
**👨‍🏫 Dashboard (for Supervisors)**  
Teachers/parents can track:

* Completion rates
* Mistakes and accuracy
* Hints used
* Overall progress

Can enter using the link for the dashboard – only for parents and teachers.

### Web App Pages

First, you log in:

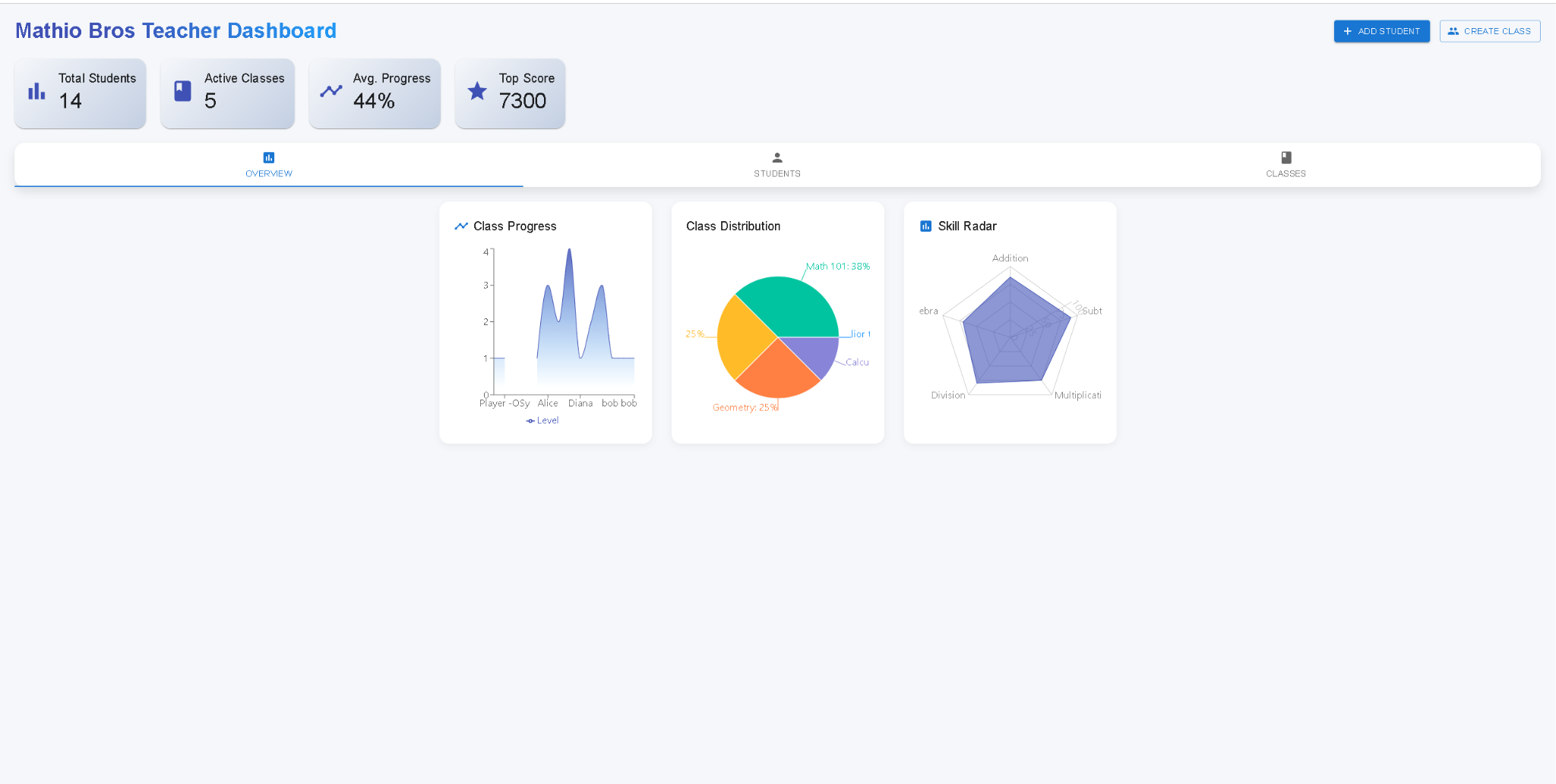


#### Home Page

Then you are taken to the landing page.

The web app's index page provides an overview of gameplay and learning progress. It features:

* A summary chart displaying the number of games played and average scores for the current week for each player who played.
* Recent activity logs show solved levels, challenges attempted, and hints used.
* A database representation of players, highlighting recent progress and their supervising teacher/parent.



Description: This figure shows the home page with a bar chart summarizing weekly activity, a list of recent games, and a table of player details.

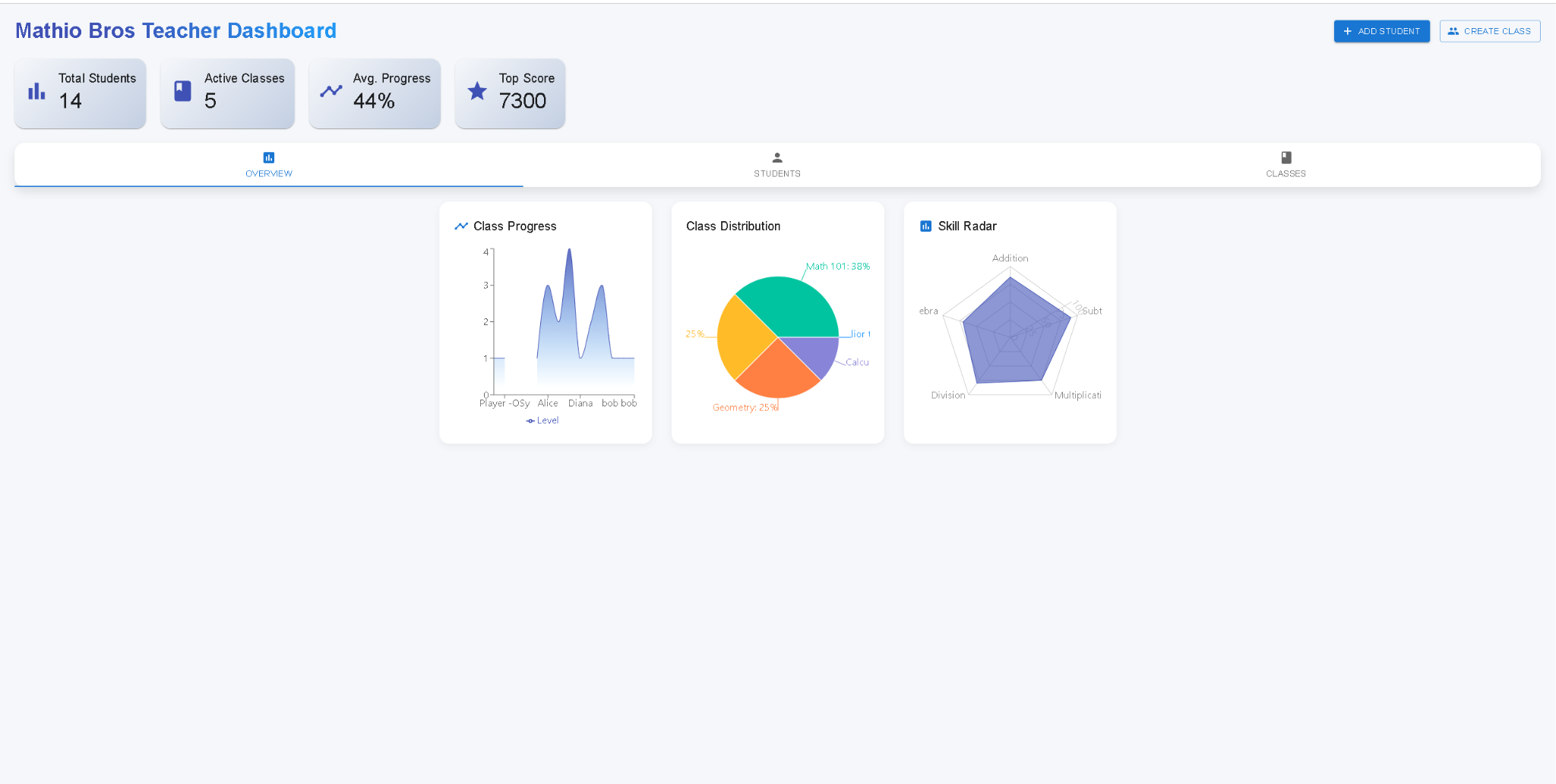
#### Analytics Page

The landing page is also the analytics page.

The analytics page allows educators and parents to delve deeper into data. Features include:

* Data Grouping: Group players by age, math skill level, or gameplay frequency.
* Data Comparison: Compare metrics such as accuracy rates, completion times, and challenges solved across different players or groups.
* Dynamic Visualizations: Charts and graphs update based on selected filters to provide clear insights.

Description: This figure showcases the analytics page with interactive filters, a line chart comparing player scores, and a dashboard of comparative metrics.



#### Reports Page

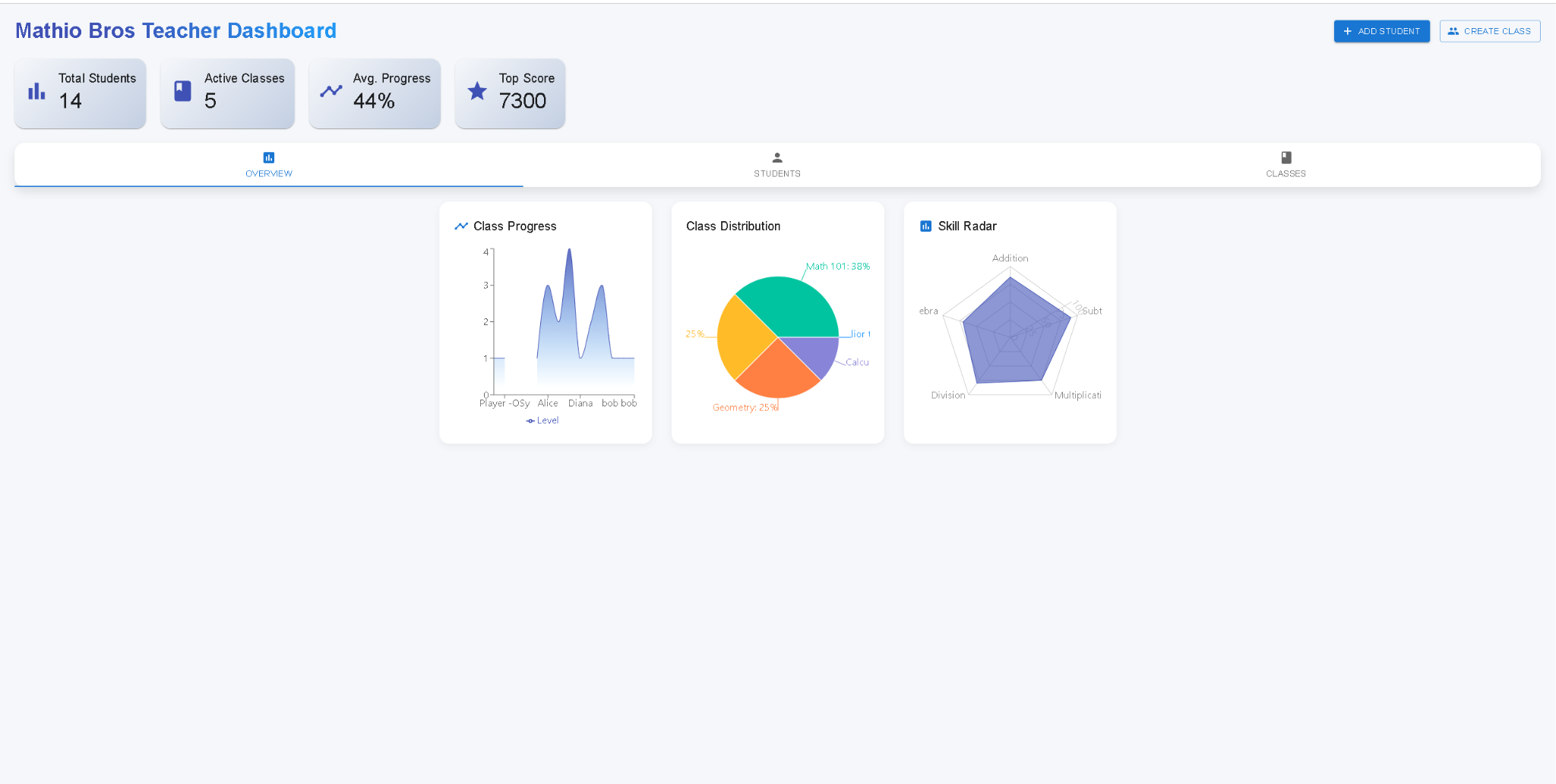
The landing page is also the reports page.

The reports page offers detailed insights for individual players. By selecting a player from the database, users can:

* View a summary report of their gameplay sessions.
* Analyze performance trends using visual charts (e.g., accuracy improvement over time).
* Review detailed stats such as hints used, challenges completed, and time spent per level.

Description: This figure depicts the reports page with a summary chart of player performance, a detailed activity log, and an interactive grid for selecting players.

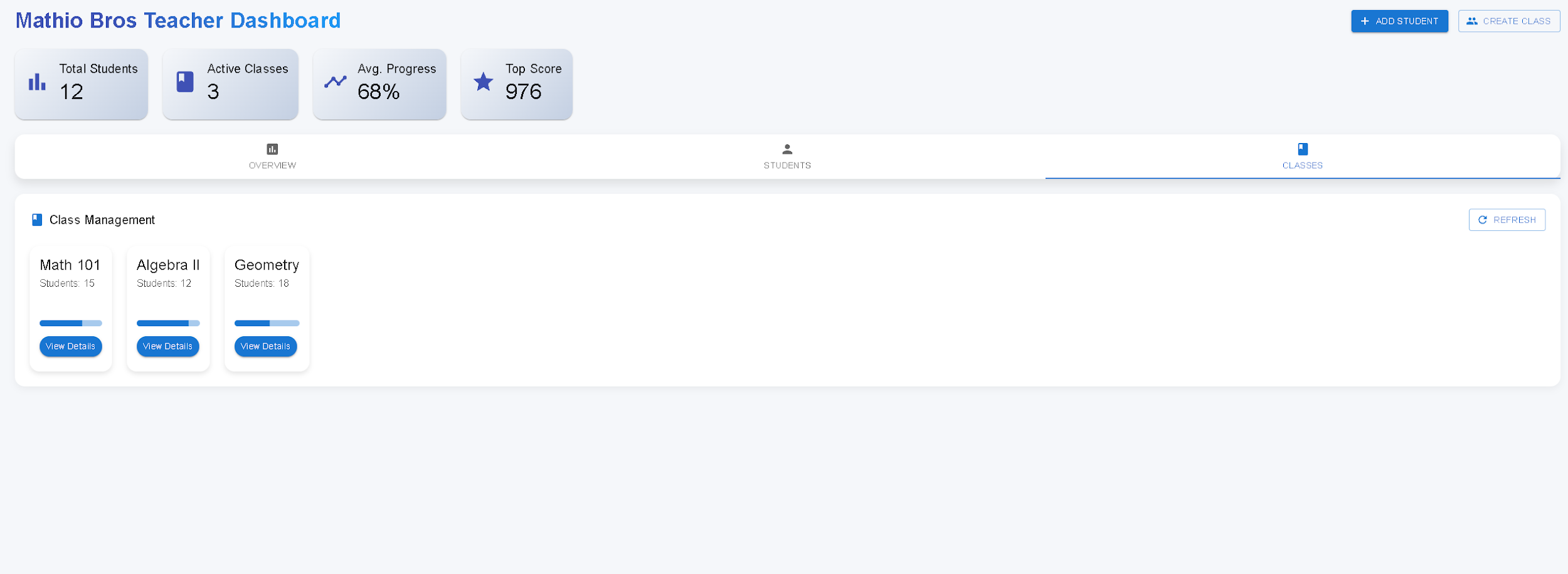
The idea is to see everything on the first page and everything in one place.



Then you can track each student that’s connected to the parent/teacher that is the -Student page:



Also, you can track your class as a teacher to see what class you have in what is their overall progress - Class page:



Enjoy learning math while having fun!

# Maintenance Guide

## Environment

* **Platform:** Unity version 6000.0.42f1
* **Supported OS:** Windows 10/11 (PC)
* **Backend Services:** next.js, Firebase.
* **Frontend Web Tools:** React.js, Next.js, Material-UI, Recharts

## Installation Instructions

### Game Application:

1. Download latest release from github.
2. Extract and run .exe file.

### Web Dashboard:

1. Navigate to /the dashboard project folder.
2. Run:

npm install

npm run dev

1. Connect to Firebase/next.js for backend data.

## Deployment Notes

* For multiplayer: Set up Photon server credentials.
* For data syncing: Configure Firebase Realtime Database rules.
* For analytics: Make sure the chart components (Recharts) are connected to user logs.

## Future Updates

* To update UI: Use Unity’s Canvas system or modify React components.
* To scale levels: Add new prefabs or scenes in Unity and register them in LevelManager.cs.

## Code overview

### 1. Core Unity Objects & Hierarchy

#### Main Scenes

* **MainMenu**: Handles the main menu UI, level selection, and settings.
* **Level1, Level2, ...**: Individual game levels with unique layouts.
* **multiLevel1, 2 …**: levels for multi-player collaboration.

### Persistent Objects (DontDestroyOnLoad)

* **GameManager**: Manages game state, score, and level progression.
* **User ID:** for profile and saving data

### 2. Key Scripts & Their Functions

#### Player & Movement

* **PlayerMovement.cs**
  + Handles player movement (walking, jumping).
  + Detects collisions with enemies, coins, and obstacles.
  + Manages player health and respawn logic.
* **Player.cs**
  + Controls sprite animations (idle, run, jump, death).

#### Enemies & AI

* **EntitiyMovement.cs**
* **Goomba.cs** (enemy)
  + Simple walk-and-crush logic (dies when jumped on).
* **Koopa.cs**
  + Retreats into the shell when hit can be kicked.

#### Game Mechanics

* **LevelEditorManager.cs**
  + Handles level builder and saving it.
* **GameManager.cs**
  + Loads levels, handles lives and restarting lvels
* **EquationLogic.cs**
  + Can update mystery blocks according to level
  + Checks equation.

#### UI & Menus

* **MainMenuController.cs**
  + Manages buttons (Play, Settings, Quit).
  + Loads selected levels.

### 3. Important Prefabs

* **Mario, multiMario (**Player prefab for single and multi player**)**
* **Goomba**,  **Koopa** (enemies).
* **MysteryBlock** (either contains an item or a number for equation).
* **MysteryBlock** (Triggers finish level sequance).

### 4. Data Management

* **FirebaseAPIManager.cs**
  + Saves:
    - Levels data
    - Time spent
    - score

### 5. External Assets & Plugins

* **TextMeshPro** (For high-quality UI text).
* **Photon for multiplayer**

### Maintenance Tips

✅ **Debugging**: Check Debug.Log outputs in key scripts (e.g., GameManager, PlayerController).  
✅ **Script Dependencies**: Many scripts reference GameManager.Instance—ensure it’s always present.  
✅ **Physics Issues**: Adjust  Collider2D settings if the movement feels off.  
✅ **New Levels**: Duplicate an existing level scene and modify the layout.

# References

[1] Saligumba, I. P., and D. Tan. "Gradual release of responsibility instructional model: Its effects on students’ mathematics performance and self-efficacy." International Journal of Scientific & Technology Research 7.8 (2018): 276-291.

[2] Molin, Gerhard. "The role of the teacher in game-based learning: A review and outlook." Serious Games and Edutainment Applications: Volume II (2017): 649-674.

[3] Chan, Liew Lee, and Noraini Idris. "Cooperative learning in mathematics education." International Journal of Academic Research in Business and Social Sciences 7.3 (2017): 539-553.

[4] Cresswell, Clio, and Craig P. Speelman. "Does mathematics training lead to better logical thinking and reasoning? A cross-sectional assessment from students to professors." PloS one 15.7 (2020): e0236153.

[5] Chen, Lang, et al. "Positive attitude toward math supports early academic success: Behavioral evidence and neurocognitive mechanisms." Psychological Science 29.3 (2018): 390-402.

[6] Prensky, Marc. "Digital game-based learning." Computers in Entertainment (CIE) 1.1 (2003): 21-21.

[7] Gee, James Paul. "What video games have to teach us about learning and literacy." *Computers in Entertainment (CIE)* 1.1 (2003): 20-20.

[8] Shute, Valerie J. "Focus on formative feedback." *Review of educational research* 78.1 (2008): 153-189.

[9] Deterding, Sebastian, et al. "From game design elements to gamefulness: defining" gamification"." *Proceedings of the 15th international academic MindTrek conference: Envisioning Future Media Environments*. 2011.

[10] Maloney, Erin A., and Sian L. Beilock. "Math anxiety: Who has it, why it develops, and how to guard against it." *Trends in cognitive sciences* 16.8 (2012): 404-406.

[11] Hattie, John, and Helen Timperley. "The power of feedback." *Review of educational research* 77.1 (2007): 81-112.

[12] Black, Paul, and Dylan Wiliam. "Assessment and classroom learning." *Assessment in Education: principles, policy & practice* 5.1 (1998): 7-74.

[13] Tomlinson, Carol Ann. The differentiated classroom: Responding to the needs of all learners. Ascd, 2014.

[14] Boaler, Jo. Mathematical mindsets: Unleashing students' potential through creative mathematics, inspiring messages, and innovative teaching. John Wiley & Sons, 2022.

[15] Johnson, David W., and Roger T. Johnson. Learning together and alone: Cooperative, competitive, and individualistic learning. Prentice-Hall, Inc, 1987.

[16] Webb, Noreen M. "Task-related verbal interaction and mathematics learning in small groups." *Journal for research in mathematics education* 22.5 (1991): 366-389.

[17] [The Importance of Mathematics In the Modern World](https://www.superprof.co.uk/blog/maths-and-the-modern-world/)

[18] [Early Childhood Mathematics: Promoting Good Beginnings](https://www.naeyc.org/positionstatements/mathematics)

[19] [The Role of Critical Thinking in STEAM](https://www.spacefoundation.org/2023/10/05/critical-thinking-steam/)

[20] [How early exposure to math concepts improves children's literacy — and you can help](https://readingandmath.org/early-math-exposure-you-can-help/)