**Software Requirements**

**Specification**

**for**

**Math Dog Game**

**Version 1**

**Prepared by Team 2 of Software Engineering Class**

**California State University, San Bernardino10-30-23**

**Table of Contents**

1. **Introduction 4**
   1. Purpose 4
   2. Document Conventions 4
   3. Intended Audience and Reading Suggestions **4**
   4. Product Scope 5
   5. References 5
2. **Overall Description 5**
   1. Product Perspective 5
   2. Product Functions 5
   3. User Classes and Characteristics 6
   4. Operating Environment 6
   5. Design and Implementation Constraints 6
   6. User documentation 6
   7. Assumptions and Dependencies 7
3. **External Interface Requirements 7**
   1. User Interfaces 7
      1. Main Screen/Home Screen: 7

Figure 01: Loading Screen/Initial Screen for math game mobile app for kids 8

* + 1. Gameplay: 8

Figure 02: Game play Screen 8

* + 1. Game over: 8
  1. Hardware Interfaces 10
  2. Software Interfaces 10
  3. Communications Interfaces 10

1. **System Features 11**
   1. Title/Menu Screen 11
      1. Description and Priority 11
      2. Stimulus/Response Sequences 11
      3. Functional Requirements: 11
   2. Generate random numbers and validate the answers 11
      1. Description and Priority: 11
      2. Stimulus/Response Sequences: 12
      3. Functional Requirements: 12
2. **Other Nonfunctional Requirements 12**
   1. Performance Requirements 12
   2. Safety Requirement 13
   3. Security Requirements 13
   4. Software Quality Attributes 13
   5. Business Rules 14
3. **Other Requirements 14**
   1. Storage solution 14
   2. Design 14
   3. Task 14
   4. Cohesiveness 15

**Appendix: Analysis Models 15**

A.UML Use Case Diagram 15

B.UML Class Diagram 15

C. UML Sequence Diagram 15

# 1. Introduction

### 1.1 Purpose

The "Math Dog Game" app is a captivating and interactive gaming experience designed with the Unity engine. It offers a fun way for children to improve their math skills, focusing on counting and subtraction. This game not only entertains but also provides an engaging and challenging educational opportunity for kids.

### 1.2 Document Conventions

The document was developed using the IEEE’s Software Requirement Specification.

### 1.3 Intended Audience and Reading Suggestions

### The Software Requirements Specification (SRS) serves as a means for users to confirm that the developed game aligns with the initial concept. To gain a comprehensive understanding of the project, refer to the project description in Part 1. For an in-depth explanation of the game's mechanics and their relationship to the characters, consult the System Features section in Part 2. To explore details about the game's interface and navigation through the front-end menus, check out the External Interface Requirements in Part 3. Part 4 outlines the technical requirements and performance standards the project must adhere to, which are categorized as Nonfunctional Requirements.

### 1.4 Product Scope

### The "Math Dog Game" is an educational tool that effectively imparts counting and subtraction concepts to children. Its diverse range of game modes ensures that players of all skill levels can grasp the underlying principles. Moreover, this game provides an enjoyable learning experience for kids, allowing them to explore the concept of subtraction through various interactive activities with different objects. Additionally, hints are integrated into the game to assist users in comprehending the concepts and successfully completing the levels.

### 1.5 References

* **Software Engineering: A Practitioner's Approach**

<https://ebookcentral.proquest.com/lib/csusb/detail.action?docID=6328275&pq-origsite=primo>

* **GitHub page: Math game mobile app for kids**

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

* **IEEE Template** for System Requirement Specification Documents:

<https://goo.gl/nsUFwy>

* **Math Kids – Add, Subtract, Count app.**

[Math](https://apps.apple.com/us/app/math-kids-add-subtract-count/id1272098657) [Kids](https://apps.apple.com/us/app/math-kids-add-subtract-count/id1272098657) [-](https://apps.apple.com/us/app/math-kids-add-subtract-count/id1272098657) [Add,](https://apps.apple.com/us/app/math-kids-add-subtract-count/id1272098657) [Subtract,](https://apps.apple.com/us/app/math-kids-add-subtract-count/id1272098657) [Count](https://apps.apple.com/us/app/math-kids-add-subtract-count/id1272098657) [on](https://apps.apple.com/us/app/math-kids-add-subtract-count/id1272098657) [the](https://apps.apple.com/us/app/math-kids-add-subtract-count/id1272098657) [App Store](https://apps.apple.com/us/app/math-kids-add-subtract-count/id1272098657) [(apple.com)](https://apps.apple.com/us/app/math-kids-add-subtract-count/id1272098657)

* **NuGet** is a package manager designed to enable developers to share reusable code.

<https://www.nuget.org/>

* **ChatGPT**

[**https://chat.openai.com/auth/login**](https://chat.openai.com/auth/login)

# 2. Overall Description

### 2.1 Product Perspective

### The game aims to closely replicate the original children's math app available on the Play Store. Its core objective is to simplify and enhance the learning experience of subtraction for kids, making the mathematical concept both more accessible and enjoyable for them.

### 2.2 Product Functions

* Assisting children in acquiring fundamental subtraction skills through the inclusion of subtraction games with varying levels of difficulty. High-quality cartoon animations have been integrated strategically to enhance the effectiveness of a child's learning and enjoyment. The following provides an overview of the key features incorporated in the game, categorized based on their respective functions **.**
* **Title / Menu Screen**: This is the application’s initial viewable screen, which includes buttons for play and settings.
* **Dog Character**: A dog character is created to help engage children.
* **Generating random numbers and validating them**: Various numbers are randomly generated, and the correct answers are provided for the subtraction games
* **Generate questions and validate the answers with reactions:** Puzzle challenges are created, and the dog character's responses are based on the user's accuracy in providing correct or incorrect answers.
* **Kid learning progression track**: With each consecutive puzzle a child solves, they progress to the next level within the game, allowing for the monitoring of their performance development.

### 2.3 User Classes and Characteristics

### The control system is deliberately made intuitive, ensuring that gameplay is accessible to all users. Thus, the overall user experience should not significantly create a division among players. Nevertheless, like many games, there is a basic distinction between hardcore players and casual players. In this game, the hardcore players are clearly defined as primary school children.

### 2.4 Operating Environment

### This application is set to be released on both the Play Store for Android and the App Store for iOS. There are no current plans to develop a web-based platform for it.

### 2.5 Design and Implementation Constraints

### The child's math game is exceptionally minimalistic in both its functionality and user interface. Currently, there are no observed limitations or constraints for the developers.

### 2.6 User documentation

### Despite the simplicity and minimalistic design of this app, end users will encounter minimal challenges. Complex operations that could potentially lead to compatibility issues have not been integrated. This app is specifically designed for young children who are beginning their educational journey, ensuring a seamless and trouble-free experience for them.

### 2.7 Assumptions and Dependencies

# We will utilize Unity 2D for handling the graphics. Following the testing phase, we will determine the minimum system requirements and the earliest supported version of the Android platform before launching the game to the public. Currently, there are no identified dependencies for this application, as it is a standalone application without any complex operations.

# 3. External Interface Requirements

### This section of the SRS outlines the system's interface specifications, encompassing requirements related to users, hardware, software, and communication interfaces.

### 3.1 User Interfaces

#### 3.1.1 Main Screen/Home Screen:

This screen is the initial loading screen for “math dog game”.

A screenshot of a game

Description automatically generated

Figure 01: Loading Screen/Initial Screen for math game mobile app for kids

**3.1.2 Gameplay:**

A screenshot of a game

Description automatically generated

### Figure 02: Game play Screen

#### 3.1.3 Game over:

Whenever an appropriate task is incomplete, the game over screen will appear and  
 prompt the user to continue the game.

**3.1.3.1 Invoke restart level:**

When invoked, the current game   
 level restart

##### 3.1.3.2 Invoke level map window.

When invoked, the game is closed and the “level map” screen appears

A screenshot of a game

Description automatically generated

Figure 03: Game over Screen for math kid game for kids.

**3.1.4. Level Complete:**

A screenshot of a video game

Description automatically generated

Figure 04: level completed Screen for math kid game for kids.

**3.1.4.1 Invoke Restart Level**

* When invoked, the current game level restarts

**3.1.4.2. Invoke Next Level**

* When this button is invoked, the next level starts

**3.2 Hardware Interfaces**

The Math kid Game for kids’ mobile app may interact with the following hardware interfaces:

##### 3.2.1 Touchscreen Interface

**Description**: The app relies on the device's touchscreen interface for user interactions, such as tapping buttons, entering answers, and navigating menus.

**Functionality:** The app should respond to touchscreen gestures, including taps, swipes, and multi-touch interactions.

##### 3.2.2 Speaker and Audio Interface

Description: The app uses the device's speaker interface to sounds, music.

Functionality: The app should play audio, instructions, and background music through the device's speaker.

**3.3 Software Interfaces**

##### 3.3.1 Unity

The application will use the Unity game engine for the user interface of the Connectome application. While the application will not contain any game components, the framework makes the interface with the Math kid Game easy. A 32-bit personal version of the unity game engine is used for the project. None of the components of the professional version should be needed for the application. Unity version 5.5 is used for the development of the project.

##### 3.3.2 Visual Studio

A streamlined code editor, Visual Studio Code supports development activities like task execution, debugging, version management. It seeks to offer only the tools a developer needs for a brief code-build-debug cycle and leaves more complicated processes to IDEs with more features, like Visual Studio IDE.

### 3.4 Communications Interfaces

# As there is no requirement for network activity, no interface requirements are necessary.

# 4 System Features

**4.1 Title/Menu Screen**

##### 4.1.1 Description and Priority

#### The title screen serves as the initial point of entry for players into the game. It provides players with options to either initiate the game or adjust the settings. Given that the home/menu screen acts as the central hub for all project activities, it is an essential inclusion.

#### 4.1.2 Stimulus/Response Sequences

**Step 1:** The player will launch the game from their portable device.

**Step 2:** The start screen loads and appears prompting the player to start the game.

**Step 3:** The player presses the button and next the actual game begins.

#### 4.1.3 Functional Requirements:

**RFQ-1:** The home / menu screen must load and appear every time the game is launched.

**RFQ-2:** If the player quits the game during any stage of a level, they must be returned to the main screen.

**RFQ-3:** If the player presses the exit button, the game will end and return the player to the mobile device’s regular interface.

**RFQ-4**: If the player completes the game, it will take them to the Level complete screen and then the player can replay the level or can go to the next level.

**4.2 Generate random numbers and validate the answers.**

**4.2.1 Description and Priority:**

#### After transitioning from the menu screen, the player proceeds to the Adding Puzzle screen, which features subtraction questions. Here, the player is tasked with selecting the correct answer and dragging it to the appropriate location. If the answer is accurate, a dog animation will be displayed to indicate correctness; otherwise, a prompt informs the player of an incorrect response. Upon providing the correct answer, the game advances to the next level

#### 4.2.2 Stimulus/Response Sequences:

**Step 1**: The player navigates to the game play screen from the main screen.

**Step 2**: In the screen we can see a dog and monster and some random numbers along with accurate number for response.

**Step 3**: A Subtraction question will be generated with random numbers and displayed on the screen.

**Step 4**: In the answer board random numbers are generated, including the correct answer.

**Step 5**: Once the answer is selected from the board, a dog animation will pop up reacting to whether the answer was correct or incorrect.

#### 4.2.3 Functional Requirements:

**RFQ-1**: The dog should appear on the screen and indicate the answer is correct if the player selects the correct answer from the random numbers.

**RFQ-2**: When the player chooses the correct answer, a dog animation needs to be played.

**RFQ-3**: When the player chooses the wrong response, it should indicate the answer is incorrect.

**RFQ-4**: By clicking on the pause menu window button displayed at the top left corner, the player can go back to the screen and choose the game options like back to home screen or settings.

# 5 Other Nonfunctional Requirements

### 5.1 Performance Requirements

### Given the capabilities of modern smartphones and Android operating systems, performance issues should not be a concern. However, phones with lower-end hardware may encounter some operational slowdowns. Regardless of the hardware, the game is meticulously crafted to ensure an enjoyable experience on all Android phones. The game's functionality will remain straightforward and user-friendly. To prevent system slowdowns, the graphics will be kept relatively simple, avoiding complexity.

### 5.2 Safety Requirement

The Kid Math Game with Dog is designed to operate without causing any disruptions or damage to other applications installed on the player's phone. Furthermore, it will not lead to overheating, safeguarding the phone's internal components. This precaution is taken to ensure the player's safety. Math Game with dog should not be played when the player's attention is split between several things.

### 5.3 Security Requirements

### As the Kid Math Game with Dog does not require any personal information from the player, it poses no risk of compromising such data. The game does not necessitate player authentication, and simply downloading the application grants access to playing the subtraction games. Consequently, the game is open to anyone who can access the player's phone. It is the player's responsibility to ensure that their phone is not accessible to unauthorized individuals, as anyone who has access to the player's phone can play the math game.

### 5.4 Software Quality Attributes

The Kid Math Game features a dog ensures responsive and reliable feedback to the player's actions, prioritizing dependability, and accuracy. The game is designed to display the outcomes of a player's actions to others in a matter of milliseconds, not delayed by ten seconds. The Kid Math Game with a dog is both flexible and adaptable, automatically saving the player's progress at the completion of each level. This feature allows players to resume the game from a logical starting point, even if their phone runs out of power during gameplay.

The Kid Math Game featuring a dog prioritizes responsive and dependable feedback for the player's actions, emphasizing accuracy. The game is designed to promptly display the results of a player's actions to others, with minimal delays, ensuring outcomes are visible within milliseconds, not ten seconds later. Moreover, the Kid Math Game with a dog is known for its flexibility and adaptability, automatically saving the player's progress upon completing each level. This feature allows players to resume the game from a logical starting point, even if their phone's power is depleted during gameplay. Furthermore, players are expected to become proficient in using the newly introduced commands in each level, as well as those introduced in previous levels, by the time they finish that level.

**5.5 Business Rules**

**N/A**

1. **Other Requirements**
   1. **Storage solution**

We explored two possible game data storage systems: utilizing a file system or an integrated database. Ultimately, we opted to leverage Azure App Service as our backend technology because of its robust support for databases and its user-friendly tools for data retrieval and storage. Through the Azure App Service class, we have a straightforward means to create, modify, and oversee Azure App Service. Additionally, with a database and regular game data saving in place, even if the application were to unexpectedly terminate, data loss is prevented.

* 1. **Design**

Upon installation of the game, a database is established, and game content is integrated. When the game is initiated, it retrieves the existing game state, encompassing the current task and associated path. If there is no existing game state, the current task is automatically set to the initial task from the newly retrieved route sourced from the database.

* 1. **Task**

The initial user interface screen for the user is a welcoming screen displaying a text greeting and a button to proceed to the next activity. When the user clicks the "next task" button, the subsequent task in the route is fetched from the database. Upon completing a task, the application automatically retrieves the following one. If the user exits the game and returns, the welcome page will reappear, and the route and task information will be retrieved from the database. If the user selects the "next task" button, the application will load the task they were working on before it was terminated.

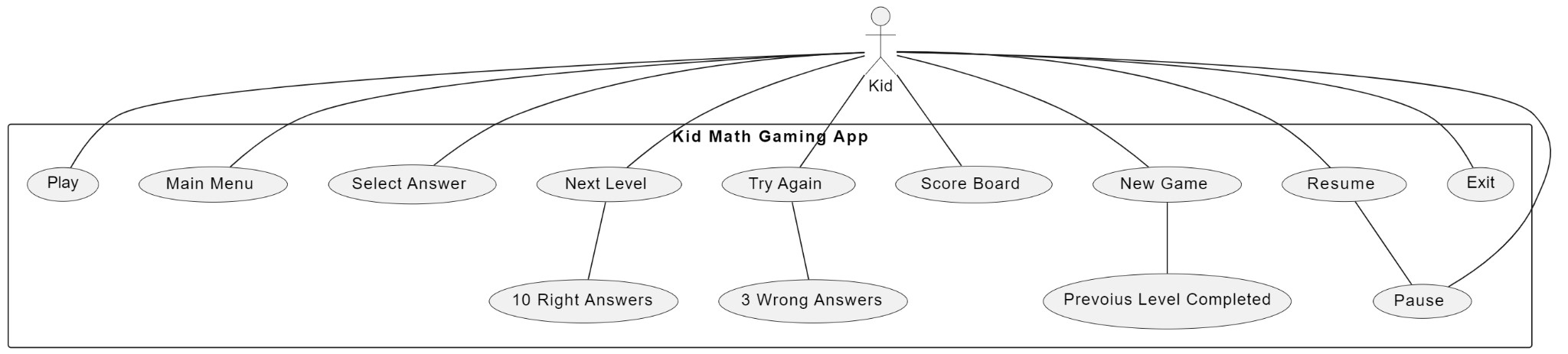
### Cohesiveness

High cohesion enhances reusability, reliability, and robustness within the project. Given that it's a gaming project, the various modules, classes, and files are interdependent as it cannot be developed in isolation. Access to both the real character's location and the positions of colliding objects is crucial to detect collisions with boundaries, walls, or obstacles. The nature of game physics necessitates that each element in the game interacts with the data and attributes of other objects, resulting in a high degree of cohesion throughout the game code. This level of cohesiveness exceeds that commonly found in typical software projects.

# Appendix: Analysis Models

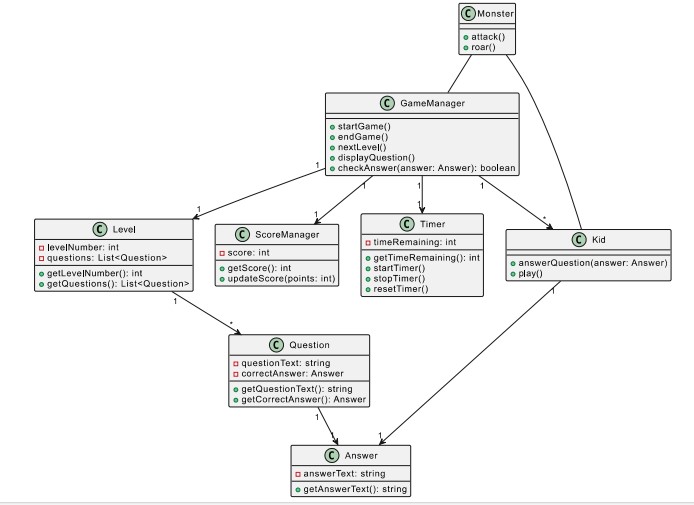
**A . UML Use Case Diagram**

Use case diagrams are the diagrams that are used to show the relation between actors and their interactions. A use case diagram shows various use cases and different types of users the system has.



## B. UML Class Diagram:

UML diagram describes a system by visualizing the different types of objects within a system and the kinds of static relationships that exist among them. It also illustrates the method operations and attributes of the classes.



## C. UML Sequence Diagram:

UML Sequence diagrams illustrate the sequences of messages between objects in an interaction. A sequence diagram consists of a group of objects that are represented by lifelines and the messages that they exchange over time during interaction.

